User’s Guide and Reference for DB2 Data

Version 13 Release 1
## Contents

### About this document
- Who should use this document
- Releases of DB2 supported
- National characters

### Summary of changes

### Part 1. Working with FM/DB2

#### Chapter 1. FM/DB2 overview
- What you can do with FM/DB2
- Templates
  - What is a template?
  - Where you can use templates

#### Chapter 2. Getting started with FM/DB2
- Starting and exiting FM/DB2
- TSO region size
- Starting FM/DB2
- Checking your FM/DB2 version
- Checking which COBOL compiler you’re using
- Exiting from FM/DB2
- Selecting the DB2 subsystem when more than one is available
  - Selecting the DB2 subsystem with the DB2 SSID field
  - Selecting the DB2 subsystem by entering the SSID command
  - Using the DB2 Subsystem Selection menu
- Using the FM/DB2 interface
  - Navigating in FM/DB2
  - FM/DB2 panel features
  - Screen size
  - Scrollable input and display fields for long names
  - Selecting options on FM/DB2 panels
  - Specifying a DB2 object name
  - Specifying the S Q L I D
  - Specifying a data set and a member name
  - Canceling long-running DB2 queries
- Getting help
  - Accessing Tutorial Help
  - Displaying field-level help information
  - Displaying the expanded form of error messages
- Setting default processing options
  - System Options (option 0)
  - Print settings (option 0.1)
  - FM/DB2 system options (option 0.2)
  - Job card specifications (option 0.3)
  - Compiler language selection (option 0.4)

#### Chapter 3. Working with templates
- Associating a template with a DB2 object
- Information stored in a template
- Printing a template
- Using templates with non-DB2 data
  - How FM/DB2 compiles a copybook into a template
  - Support for COBOL data description entries
  - Support for HLASM data description entries
  - Support for PL/I data description entries
  - Support for variable-length arrays
- Editing a template
  - Adding a description to a template
  - Selecting rows
  - Holding columns (when scrolling left or right)
  - Changing the sequence of displayed or printed data

### Utilities (option 3)
- Print utility options (option 3.1)
- Copy utility options (option 3.3)
- Object List Options (option 3.4)
- Export utility options (option 3.7)
- DB2 LOAD utility options (option 3.L)
- DB2 Utility LISTDEF options (option 3.UL)
- DB2 Utility OPTIONS options (option 3.UO)
- DB2 Utility TEMPLATE options (option 3.UT)
- DB2 Utility TEMPLATE options (option 3.U.U)
- Displaying or editing DBCS characters
- SAF-rule controlled auditing
  - Determining when SAF-rule controlled auditing is in effect
  - SAF rules for auditing
  - SAF-rule controlled auditing not in effect
  - SAF-rule controlled auditing in effect, no auditing
  - SAF-rule controlled auditing in effect, auditing possible

### Chapter 3. Working with templates
- COBOL compiler specifications (option 0.5)
- HLASM compiler specifications (option 0.6)
- PL/I compiler specifications (option 0.7)
- Temporary Data Set Allocations (option 0.8)
- Output Data Set Allocations (option 0.9)
- Trace options (option 0.10)
- ISPF settings (option 0.11)
- View and Edit options (options 1 and 2)
- Utilities (option 3)
- Print utility options (option 3.1)
- Copy utility options (option 3.3)
- Object List Options (option 3.4)
- Export utility options (option 3.7)
- DB2 LOAD utility options (option 3.L)
- DB2 Utility LISTDEF options (option 3.UL)
- DB2 Utility OPTIONS options (option 3.UO)
- DB2 Utility TEMPLATE options (option 3.UT)
- DB2 Utility TEMPLATE options (option 3.U.U)
- Displaying or editing DBCS characters
- SAF-rule controlled auditing
  - Determining when SAF-rule controlled auditing is in effect
  - SAF rules for auditing
  - SAF-rule controlled auditing not in effect
  - SAF-rule controlled auditing in effect, no auditing
  - SAF-rule controlled auditing in effect, auditing possible
### Chapter 4. Viewing and changing DB2 data

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 editor sessions</td>
<td>91</td>
</tr>
<tr>
<td>Starting an FM/DB2 editor session</td>
<td>93</td>
</tr>
<tr>
<td>Specifying the editor session mode: &quot;normal mode&quot; or &quot;large mode&quot;</td>
<td>93</td>
</tr>
<tr>
<td>Specifying the start position for your FM/DB2 editor session</td>
<td>95</td>
</tr>
<tr>
<td>Setting options for the current FM/DB2 editor session</td>
<td>95</td>
</tr>
<tr>
<td>Validating and saving changes with an FM/DB2 editor session</td>
<td>96</td>
</tr>
<tr>
<td>Ending an FM/DB2 editor session</td>
<td>101</td>
</tr>
<tr>
<td>Starting another FM/DB2 editor session</td>
<td>103</td>
</tr>
<tr>
<td>Specifying the DB2 object you want to view</td>
<td>103</td>
</tr>
<tr>
<td>Restricting the data that FM/DB2 loads from DB2</td>
<td>105</td>
</tr>
<tr>
<td>Storage considerations</td>
<td>105</td>
</tr>
<tr>
<td>Viewing large tables or views</td>
<td>106</td>
</tr>
<tr>
<td>Viewing data in an FM/DB2 editor session</td>
<td>108</td>
</tr>
<tr>
<td>Count of rows fetched in an FM/DB2 editor session</td>
<td>108</td>
</tr>
<tr>
<td>Locating a row</td>
<td>108</td>
</tr>
<tr>
<td>Locating a column</td>
<td>109</td>
</tr>
<tr>
<td>Finding a string</td>
<td>112</td>
</tr>
<tr>
<td>Finding data within specific columns</td>
<td>113</td>
</tr>
<tr>
<td>Finding a string within specific portions of the data</td>
<td>115</td>
</tr>
<tr>
<td>Finding data within excluded or non-excluded rows</td>
<td>115</td>
</tr>
<tr>
<td>Finding data within a label range</td>
<td>115</td>
</tr>
<tr>
<td>Searching numeric columns</td>
<td>117</td>
</tr>
<tr>
<td>Changing the view of displayed data</td>
<td>117</td>
</tr>
<tr>
<td>Handling special data</td>
<td>131</td>
</tr>
<tr>
<td>DATE and TIME columns</td>
<td>131</td>
</tr>
<tr>
<td>Large numeric fields</td>
<td>131</td>
</tr>
<tr>
<td>Working with large tables</td>
<td>132</td>
</tr>
<tr>
<td>Null values and VARCHAR processing</td>
<td>136</td>
</tr>
<tr>
<td>Unicode data</td>
<td>137</td>
</tr>
<tr>
<td>Changing data</td>
<td>138</td>
</tr>
<tr>
<td>Installation settings that affect CAPS</td>
<td>138</td>
</tr>
<tr>
<td>Changing data by overtyping</td>
<td>139</td>
</tr>
<tr>
<td>Changing data with the CHANGE command</td>
<td>140</td>
</tr>
<tr>
<td>Changing data using prefix commands</td>
<td>144</td>
</tr>
<tr>
<td>Converting data to uppercase</td>
<td>146</td>
</tr>
<tr>
<td>&quot;Read-only&quot; views, non-updateable columns</td>
<td>147</td>
</tr>
<tr>
<td>Deleting rows</td>
<td>148</td>
</tr>
<tr>
<td>Deleting rows within excluded or non-excluded rows</td>
<td>148</td>
</tr>
<tr>
<td>Deleting rows within a label range</td>
<td>149</td>
</tr>
<tr>
<td>Editing related tables</td>
<td>149</td>
</tr>
<tr>
<td>Listing related tables</td>
<td>150</td>
</tr>
<tr>
<td>Self-referencing constraints</td>
<td>151</td>
</tr>
<tr>
<td>Referential integrity errors</td>
<td>151</td>
</tr>
<tr>
<td>Starting an Edit session of a related table</td>
<td>152</td>
</tr>
<tr>
<td>Differences when you edit a related table</td>
<td>152</td>
</tr>
<tr>
<td>Differences when saving or canceling while editing a related table</td>
<td>153</td>
</tr>
<tr>
<td>Navigating between Edit sessions</td>
<td>153</td>
</tr>
<tr>
<td>Deleting a primary key</td>
<td>153</td>
</tr>
<tr>
<td>Inserting a new foreign key</td>
<td>154</td>
</tr>
<tr>
<td>Updating a primary key</td>
<td>154</td>
</tr>
<tr>
<td>Adding a primary key</td>
<td>155</td>
</tr>
<tr>
<td>Issuing DB2 commands from within an FM/DB2 session</td>
<td>156</td>
</tr>
<tr>
<td>Issuing DB2 commands from the Enter and Execute DB2 Commands panel</td>
<td>156</td>
</tr>
<tr>
<td>Issuing DB2 commands from the command line</td>
<td>156</td>
</tr>
<tr>
<td>Handling errors during an Edit session</td>
<td>157</td>
</tr>
<tr>
<td>Determining why an error occurred</td>
<td>158</td>
</tr>
<tr>
<td>&quot;Duplicate Row&quot; errors</td>
<td>158</td>
</tr>
<tr>
<td>&quot;No Primary Key&quot; errors</td>
<td>158</td>
</tr>
<tr>
<td>&quot;Delete Restrict Rule&quot; errors</td>
<td>159</td>
</tr>
<tr>
<td>&quot;Check Constraint&quot; errors</td>
<td>159</td>
</tr>
<tr>
<td>&quot;Row Not Found&quot; errors</td>
<td>159</td>
</tr>
<tr>
<td>Deadlocks or timeouts</td>
<td>160</td>
</tr>
<tr>
<td>Example of correcting an error during an Edit session</td>
<td>160</td>
</tr>
<tr>
<td>Chapter 5. Creating and dropping DB2 objects</td>
<td>163</td>
</tr>
<tr>
<td>Working with the Create and Drop panels</td>
<td>164</td>
</tr>
<tr>
<td>SQL Keywords</td>
<td>165</td>
</tr>
<tr>
<td>Creating a database</td>
<td>173</td>
</tr>
<tr>
<td>Creating a table space</td>
<td>174</td>
</tr>
<tr>
<td>Table Space Type (Create Table Space)</td>
<td>174</td>
</tr>
<tr>
<td>Table Space Allocation (Create Table Space)</td>
<td>174</td>
</tr>
<tr>
<td>Data Storage (Create Table Space)</td>
<td>175</td>
</tr>
<tr>
<td>Locking (Create Table Space)</td>
<td>175</td>
</tr>
<tr>
<td>Data Sharing (Create Table Space)</td>
<td>175</td>
</tr>
<tr>
<td>Partitions (Create Table Space)</td>
<td>175</td>
</tr>
<tr>
<td>Creating a table</td>
<td>176</td>
</tr>
<tr>
<td>Creating a table using a model</td>
<td>176</td>
</tr>
<tr>
<td>Creating a table without using a model</td>
<td>176</td>
</tr>
<tr>
<td>Table Creation options</td>
<td>179</td>
</tr>
<tr>
<td>1. Columns (Create Table)</td>
<td>180</td>
</tr>
<tr>
<td>2. Nulls and default values (Create Table)</td>
<td>180</td>
</tr>
<tr>
<td>3. Options (Create Table)</td>
<td>180</td>
</tr>
<tr>
<td>4. Unique constraints (Create Table)</td>
<td>180</td>
</tr>
<tr>
<td>5. Referential constraints (Create Table)</td>
<td>181</td>
</tr>
<tr>
<td>6. Check constraints (Create Table)</td>
<td>182</td>
</tr>
<tr>
<td>7. Procedure exits (Create Table)</td>
<td>183</td>
</tr>
<tr>
<td>8. Generate values (Create Table)</td>
<td>183</td>
</tr>
<tr>
<td>Creating a view</td>
<td>183</td>
</tr>
<tr>
<td>Creating an alias</td>
<td>183</td>
</tr>
<tr>
<td>Creating an index</td>
<td>184</td>
</tr>
<tr>
<td>Index type (Create Index)</td>
<td>184</td>
</tr>
<tr>
<td>Index Column Selection (Create Index)</td>
<td>185</td>
</tr>
<tr>
<td>Allocation (Create Index)</td>
<td>185</td>
</tr>
<tr>
<td>Index (Create Index)</td>
<td>185</td>
</tr>
<tr>
<td>Index Partitions (Create Index)</td>
<td>185</td>
</tr>
<tr>
<td>Partition Values Option (Create Index)</td>
<td>185</td>
</tr>
<tr>
<td>Creating a synonym</td>
<td>185</td>
</tr>
</tbody>
</table>
Creating a distinct type.............. 186
Creating a function.............. 186
Parameters (Create Function)...... 186
Returned Data Type (Create Function).... 187
Option List 1/2 (Create Function)..... 187
Option List 2/2 (Create Function)..... 187
Creating a procedure.............. 187
Procedure Parameters (Create Procedure)...... 187
Parameter Types (Create Procedure)..... 187
Option List 1/2 (Create Procedure)..... 188
Option List 2/2 (Create Procedure)..... 188
Creating a trigger.............. 188
Details (Create Trigger)........... 188
Search Conditions (Create Trigger)...... 188
SQL Statement (Create Trigger)...... 188
Creating an auxiliary table........... 189
Dropping DB2 objects.............. 189

Chapter 6. Populating a DB2 table with data........ 191
Editing the template.............. 192

Chapter 7. Copying data........ 193
Copying data from one DB2 object to another........ 193
Specifying details for the “From” table........ 194
Specifying details for the “To” table........ 196
Copying data from a VSAM or QSAM file........ 198
When (and when not) to use the Import utility........ 199
Specifying details for the “From” data set........ 200
Specifying details for the “To” table........ 201
Using a REXX procedure with the Import utility........ 203
General information about REXX........ 204
Enhancing the Import utility using a REXX procedure........ 211
Copying data to a VSAM or QSAM file........ 221
Specifying details for the “From” table........ 222
Specifying details for the “To” data set........ 224
Copying data from an FM/DB2 editor session, or from an SQL statement........ 229
Scrambling data........ 232
When can you scramble data?........ 232
Scrambling rules........ 232
How do I...?........ 233
How do I copy data from one DB2 table to another on the same DB2 system?........ 233
How do I copy data from one DB2 table to another on a different (connectable) DB2 system?........ 233
How do I copy data from one DB2 table to another on a different (non-connectable) DB2 system?........ 233
How do I copy and reorder selected data from a DB2 table to a sequential file?........ 234
How do I define a copybook that describes data exported to a sequential file?........ 234
How do I define a copybook that describes data exported to a sequential file?........ 234

Chapter 8. Working with lists of DB2 objects........ 241
Displaying or printing a list of objects........ 242
Working with object list panels........ 244
Limiting the data displayed........ 245
Locating the data displayed........ 245
Locating a row or column in a list of DB2 objects........ 250
Finding a string in a list of DB2 objects........ 250
Using the line command area (Cmd)........ 250
Sorting the displayed data........ 257
Reverse engineering........ 257
Performance tips........ 259
Considerations........ 259

Chapter 9. Printing........ 261
Printing the contents of a DB2 table........ 261
Formatting the print output........ 262
Running Print in batch........ 263
Printing rows from a DB2 table........ 264
Looking at the print output from your FM/DB2 session........ 266
Printing an audit trail report........ 267
Audit Trail Report........ 269

Chapter 10. Managing DB2 privileges........ 271
Using the Manage DB2 Privileges panel........ 271
Displaying privileges........ 273
Granting privileges........ 273
Revoking privileges........ 275

Chapter 11. Generating batch JCL for DB2 utility jobs........ 277
Setting options for DB2 utilities........ 278
Generating a LISTDEF statement in the DB2 utility job........ 280
Generating a DB2 utility job........ 280
Working with LISTDEF utility panels........ 282
COPY utility........ 283
Default values........ 284
COPY (table spaces)........ 284
COPY (index spaces)........ 285
LOAD utility........ 286
Input data........ 287
Loading data using LOAD utility control statements........ 287
Loading data described by a copybook or template........ 289
REBUILD utility........ 292
REBUILD utility (indexes)........ 292
REBUILD utility (indexes for table spaces)........ 293
RECOVER utility........ 294
RECOVER (table spaces)........ 295
RECOVER (index spaces)........ 297
REORG utility........ 298
REORG (table spaces)........ 298
REORG (indexes)........ 300
RUNSTATS utility........ 302
RUNSTATS (table spaces)........ 303
RUNSTATS (indexes)........ 304
Chapter 12. Working with SQL statements

Should you use basic or advanced SQL prototyping? 317
Using basic SQL prototyping 317
Specifying a predicate 320
Example of using basic SELECT prototyping 321
Using advanced SQL prototyping 325
Tabbing and color-coding 327
Using the advanced SELECT prototyping panels to build a SELECT statement 328
Changing the SQL statement 333
Handling repeatable items in the syntax 341
Executing a SELECT statement after you have built it using prototyping 350
Entering, executing, and explaining SQL statements 351
Entering SQL statements 351
Executing SQL statements 352
Explaining SQL... 353
Editing and executing SQL statements from a data set 354
Managing tables used by SQL explain 355
Exiting SQL prototyping 355

Chapter 13. Interfacing with DB2 Interactive

Part 2. FM/DB2 reference

Chapter 14. DB2 data types

Large objects (LOBs) 363

Chapter 15. FM/DB2 panels and fields

Scrollable fields on FM/DB2 panels 365
Action bar pull-down menu 365
Additional LISTDEF Statements panel 367
Additional TEMPLATE Statements panel 368
Advanced SELECT Prototyping panel 368
Alter Database panel 371
Alter External Scalar Function panel 372
Alter Index panel 373
Alter Stored Procedure panel 374
Alter Table panel 375
Alter Table - ADD CHECK Constraint panel 376
Alter Table - ADD Column panel 377
Alter Table - ADD FOREIGN KEY Constraint panel 378
Alter Table - ADD PRIMARY KEY panel 379
Alter Table - AUDIT panel 380
Alter Table - DATA CAPTURE panel 381
Alter Table - DROP CHECK Constraint panel 382
Alter Table - DROP FOREIGN KEY Constraint panel 383
Alter Table - DROP PRIMARY KEY panel 384
Alter Table - RESTRICT ON DROP panel 385
Alter Table - VALIDPROC panel 386
Alter Table Space panel 387
Application Packages panel 387
Application Plans panel 389
Basic SELECT Prototyping panel 391
Bind Application Plan panel 396
Bind Package panel 396
Children of Table panel 396
Collections panel 396
Column Attributes panel (DATE) 400
Column Attributes panel (numeric) 402
Column Attributes panel (TIME) 405
Column Attributes panel (TIMESTAMP) 406
Column Distribution panel 407
Column Part Statistics panel 407
Column Selection/Edit panel 407
Columns panel 411
Columns in Index panel 413
Columns in Table panel 413
Comment panels 413
Compiler Language Selection panel 413
Constraint Columns for panel 415
Copy Options panel 415
Copy Utility ("From") panel 418
COPY Utility (Index Spaces) panel 421
COPY Utility (Index Spaces) with LISTDEF panel 423
COPY Utility ("To") panel 424
COPY Utility (Table Spaces) panel 427
COPY Utility (Table Spaces) with LISTDEF panel 429
Create Alias panel 430
Create Auxiliary Table panel 431
Create Database panel 432
Create Distinct Type panel 433
Create Function: Option List (1/2) panel 435
Create Function: Option List (2/2) panel 436
Create Function panel 437
Create Function: Parameter Types panel 438
Create Function: Parameters panel 439
Create Function: Returned Data Type panel 441
Create Index panel 442
Create Index: Allocation panel 444
Create Index: Column Selection panel 446
Create Index: Options panel 447
Create Index: Partitions panel 448
Create Index: Partition Values panel 450
Create Index: Type panel 451
Create Procedure panel 452
Create Procedure: Option List (1/2) panel 453
Create Procedure: Option List (2/2) panel 455
Create Procedure: Parameter Types panel 456
Create Procedure: Parameters panel 457
Create Synonym panel 458
Create Table panel 461
Create Table Space panel 462
Create Table Space: Allocation panel 463
Create Table Space: Data Sharing Options panel 465
Create Table Space: Data Storage Options panel 466
Create Table Space: Define Partitions panel 467
Create Table Space: Locking Options panel 469
Create Table Space: Type panel 470
Create Table: Column Check Constraints panel 471
Create Table: Column Referential Constraints panel 473
Create Table: Columns panel 474
Create Table: Generate Values panel 476
Create Table: Model Load panel 477
Create Table: Nulls and Default Values panel 479
Create Table: Options panel 481
Create Table: Procedure Exits panel 483
Create Table: Table Check Constraints panel 484
Create Table: Table Referential Constraint panel 485
Create Table: Table Referential Constraints panel 486
Create Table: Unique Constraints panel 487
Create Trigger panel 488
Create Trigger: Details panel 490
Create Trigger: Search Condition panel 491
Create Trigger: SQL statement panel 492
Create View panel 493
Create View Source Statements panel 495
Data Create Utility panel 495
Database Request Modules panel 498
Database Structure panel 500
Databases panel 500
DB2 Browse panel 501
DB2 Edit panel 505
DB2 Location Selection panel 509
DB2 Object Functions panel 510
DB2 Save Error Action panel 511
DB2 Subsystem Selection panel 512
DB2 Template Description panel 514
DB2 Utilities panel 515
DB2 View panel 518
DB2I Primary Option Menu panel 521
Details panels 522
Display Row panel 523
Distinct Types panel 523
Drop panels 524
Drop Confirmation panel 525
Edit/Execute SQL (Data Set) panel 526
Editor Options (1 of 8) panel 528
Editor Options (2 of 8) panel 531
Editor Options (3 of 8) panel 535
Editor Options (4 of 8) panel 540
Editor Options (5 of 8) panel 543
Editor Options (6 of 8) panel 545
Editor Options (7 of 8) panel 548
Editor Options (8 of 8) panel 551
Enter and Execute DB2 Commands panel 554
Enter, Execute and Explain SQL Statements panel 555
EXPLAIN Interpretation panel 556
Explain Utilities panel 556
Export Options (1 of 3) panel 558
Export Options (2 of 3) panel 561
Export Options (3 of 3) panel 563
Export “To” panel 564
Export Utility panel 567
Field Selection/Edit panel 570
FM/DB2 Release News panel 572
FM/DB2 Object List Options (1 of 2) panel 572
FM/DB2 Object List Options (2 of 2) panel 575
FM/DB2 Systems Options (1 of 4) panel 577
FM/DB2 Systems Options (2 of 4) panel 578
FM/DB2 Systems Options (3 of 4) panel 580
FM/DB2 Systems Options (4 of 4) panel 582
Foreign Keys for Table panel 583
Free Application Plan panel 583
Free Package panel 583
From Column Mapping panel 583
Functions panel 584
Generate SQL From DB2 Catalog panel 585
Grant privileges panels 587
Import Options panel 588
Import Utility (“From”) panel 591
Import Utility (“To”) panel 592
Index Parts panel 595
Indexes panel 595
Indexes and Columns for panel 597
ISPF Primary Option panel 597
ISPF Settings panel 598
List Column panel 598
List Table panel 599
LISTCAT Output Display panel 599
LISTDEF Options panel 599
LOAD from panel 601
LOAD Utility panel 603
LOAD Utility panel using DB2 templates 604
LOAD Utility - Using Templates panel 605
LOAD Utility Options panel 607
Manage DB2 Privileges panel 609
Object List Line Commands panel 610
Object List Utility panel 610
OPTIONS panel 612
Package Dependencies panel 614
Package List panel 614
Parents of Table panel 614
Partitioned Table Statistics panel 615
Plan Dependencies panel 615
Plan Enabled/Disabled Connections panel 615
Plan Table Rows panel 615
Primary Key for Table panel 617
Primary Option Menu panel 617
Print Audit Trail panel 618
Print Utility Options panel 619
Print Utility panel 621
Privilege details panels 624
Privileges panels 624
Rebind Application Plan panel 625
Rebind Package panel 627
REBUILD (Indexes for Table Spaces) panel 628
REBUILD (Indexes for Table Spaces) with LISTDEF panel 629
REBUILD Utility (Indexes) panel 630
REBUILD (Indexes) with LISTDEF panel 632
Record Type Selection panel 633
RECOVER Utility (Indexes Spaces) panel 634
RECOVER Utility (Indexes Spaces) with LISTDEF panel 635
RECOVER Utility (Table Spaces) panel 636
RECOVER Utility (Table Spaces) with LISTDEF panel 637
Recovery Information panel 638
Related Tables panel 639
Remote DB2 Location Selection panel 641
REORG Utility (Indexes) panel 642
RIGHT primary command 773
RP primary command 774
RUNTEMP primary command 774
SAVE primary command 775
SAVEAS primary command 775
SHADOW primary command 775
SHOW primary command 776
SHOWCOB primary command 776
SORT primary command 776
SQL primary command 777
SQLID primary command 778
TEDIT primary command 779
TOP primary command 779
TP primary command 779
TYPE primary command 780
UNDO primary command 780
UP primary command 780
VER primary command 781
VIEW primary command 782
XMLBRWS primary command 782
XMLEDIT primary command 783
XMLLEN primary command 784
XMLVIEW primary command 785
ZOOM primary command 785

Chapter 17. FM/DB2 functions 787
How to use this reference chapter 787
General tips about performance when you use File Manager functions 788
File Manager DB2 batch functions 788
DBC (Copy) batch command 788
DBI (Import) batch command 789
DBP (Print) batch command 790
DBX (Export) batch command 799
D2G (Data Generate) batch command 800
D2TP (Template create/update) batch command 812
Specifying a DB2 object name 815
Specifying the template for a DB2 object 815

Specifying the template/copybook for a data set 816

Part 3. Appendixes 817

Appendix. FM/DB2 messages 819

Support resources and problem solving information 941
Searching IBM support Web sites for a solution 941
Searching the information center 941
Searching product support documents 941
IBM Support Assistant 942
Obtaining fixes 942
Receiving support updates through e-mail notification 943
Receiving support updates through RSS feeds 944
If you need to contact IBM Software Support 944
Determining the business impact 945
Describing problems and gathering information 945
Submitting problems 946

Notices 947
Programming interface information 949
Trademarks 949

Bibliography 951
File Manager publications 951
Related publications for Problem Determination Tools 951
Related publications for COBOL 951
Related publications for PL/I 951
Related publications for DB2 951
Related publications for z/OS 951

Index 953
About this document

This document provides guidance and reference information for users of File Manager for z/OS DB2 Component.

In the rest of this document, the term FM/DB2 refers to File Manager for z/OS DB2 Component.

This document consists of three parts:

Part 1. Working with FM/DB2

Describes the concepts inherent within FM/DB2, provides step-by-step instructions on how to perform tasks using the FM/DB2 panels under ISPF. The information is presented in a task-based format, with panel and command definitions provided only when they are relevant to the current task. For a complete description of a particular panel or command, you should use the appropriate chapter in Part 2 FM/DB2 Reference.

Users who are new to FM/DB2 should find that Part 1 presents the tasks in logical groupings, helping you to use the guide as a learning tool, as well as enabling you to quickly find the specific task instructions you need.

Part 2. FM/DB2 reference

Provides a lookup reference for FM/DB2 panels and commands. The information is presented in two chapters:

- **FM/DB2 panels and fields**: Lists the FM/DB2 panels, alphabetically by panel title. For each listed panel, cross-references are provided to one or more “parent” panels (the panels from which you can reach the panel in question) and also, if applicable, cross-references to any “child” panels (the panels you can reach from this panel). For several of the panels, there is a definition for each field in the panel including, where applicable, the value ranges that are valid for each field.

- **FM/DB2 commands**: Lists the primary commands you can use with FM/DB2, including a description of their syntax and usage.

Both new and experienced users of FM/DB2 can use Part 2 to quickly look up such things as the exact syntax of a command, or the acceptable values for a field in a panel.

Part 3. Appendixes

The Appendix is made up of the File Manager DB2® Messages, and provides the full text explanation of all FM/DB2 batch error messages. The support information tells you about IBM® Web sites that can help you answer questions and solve problems, followed by notices, bibliography, glossary and index.

File Manager for z/OS contains four “flavors” of File Manager, for working in different environments or with different data formats:

- **DB2 Component** (described in this document) for working with DB2 data
- **File Manager** “base” for working with QSAM data sets, VSAM data sets or PDS members
- **IMS Component** for working with IMS™ data
- **FM/CICS** for working with CICS® resources.
This document describes only the **DB2 Component**. For more information about the File Manager “base” and IMS Component, see these books:

- **File Manager User’s Guide and Reference** SC19-4119
- **File Manager User’s Guide and Reference for IMS Data** SC19-4121
- **File Manager User’s Guide and Reference for CICS** SC19-4122

### Who should use this document

This document is for three kinds of DB2 users:

- Application programmers who need to test and debug programs
- Service support personnel who need to analyze and fix problems
- System administrators and system operators who need to do routine tasks such as moving large amounts of data

This document assumes that you are familiar with DB2 and with using ISPF.

To use FM/DB2 functions in batch jobs, you must already be familiar with JCL.

### Releases of DB2 supported

FM/DB2 supports:

- DB2 Version 9
- DB2 Version 10
- DB2 Version 11

Where variations exist within FM/DB2 depending on the version of DB2 you are using, these variations are highlighted in this document by indicators that identify the start and end of a section of text. For example:

**Note:** The examples shown in this document were produced using DB2 Version 9. If you are using DB2 Version 10 or 11, you may find minor variations in the appearance of FM/DB2 panels.

### National characters

File Manager uses the national characters shown in [Table 1](#).

**Table 1. National characters**

<table>
<thead>
<tr>
<th>Character</th>
<th>Hexadecimal value</th>
<th>Displayed as, in code pages 37 and 500</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dollar sign</td>
<td>X’5B’</td>
<td>$</td>
</tr>
<tr>
<td>Pound sign</td>
<td>X’7B’</td>
<td>#</td>
</tr>
<tr>
<td>At sign</td>
<td>X’7C’</td>
<td>@</td>
</tr>
</tbody>
</table>
National characters

Table 1. National characters (continued)

<table>
<thead>
<tr>
<th>Character</th>
<th>Hexadecimal value</th>
<th>Displayed as, in code pages 37 and 500</th>
</tr>
</thead>
</table>

Note:
1. The dollar sign ($) and the pound sign (#) have special syntactical meaning in File Manager syntax.
2. The at sign (@) is the default value used for the “Null column input indicator”.
3. In countries using code pages other than 37 and 500:
   a. The dollar sign ($), pound sign (#), and at sign (@) as represented on terminal keyboards might generate a different hexadecimal representation, and this might cause an error or unwanted results. For example, in some countries the $ character might generate a X'4A'.
   b. The hexadecimal values in Table 1 on page xii might display as different characters to those shown.

When you enter File Manager commands in batch or online, use the keyboard characters that correspond to the hexadecimal values shown in Table 1 on page xii.

Prerequisite and related information

FM/DB2 documentation supports the following tasks:

Evaluating FM/DB2

[File Manager Fact Sheet] provides an overview of the product to help the business professional decide if FM/DB2 has the functions needed.

Planning for, installing, customizing, and maintaining FM/DB2

Refer first to [File Manager Customization Guide], which might then refer you to the [File Manager Program Directory] (included with the product tape) for some information.

Using FM/DB2

This document, the [File Manager User's Guide and Reference for DB2 Data], is a guide to using FM/DB2. When using FM/DB2 under ISPF, you can also refer to the online help.

For information about related products, see the “Bibliography” on page 951.

How to read syntax diagrams

The syntactical structure of commands described in this document is shown by means of syntax diagrams.

[Figure 1 on page xiv] shows a sample syntax diagram that includes the various notations used to indicate such things as whether:

- An item is a keyword or a variable.
- An item is required or optional.
- A choice is available.
- A default applies if you do not specify a value.
- You can repeat an item.
Prerequisite and related information

Syntax

---

**Order of reading**

Read the syntax diagrams from left to right, from top to bottom, following the path of the line.

The ►► symbol indicates the beginning of a statement.

The —— symbol indicates that a statement is continued on the next line.

The —— symbol indicates that a statement is continued from the previous line.

The —— symbol indicates the end of a statement.

**Keywords**

Keywords appear in uppercase letters.

---

Sometimes you only need to type the first few letters of a keyword; the required part of the keyword appears in uppercase letters.
In this example, you could type "KEY", "KEYW", "KEYWO", "KEYWOR" or "KEYWORD".
The abbreviated or whole keyword you enter must be spelled exactly as shown.

**Variables**
Variables appear in lowercase letters. They represent user-supplied names or values.

```
>> required_variable
```

**Required items**
Required items appear on the horizontal line (the main path).

```
>> COMMAND_NAME—required_variable
```

**Optional items**
Optional items appear below the main path.

```
>> OPTIONAL_KEYWORD=variable
```

**Choice of items**
If you can choose from two or more items, they appear vertically, in a stack.

If you *must* choose one of the items, one item of the stack appears on the main path.

```
>> required_choice1
   required_choice2
   required_choice3
```

If choosing one of the items is optional, the entire stack appears below the main path.

```
>> optional_choice1
   optional_choice2
```

If a default value applies when you do not choose any of the items, the default value appears above the main path.

```
>> DEFAULT_KEYWORD
   KEYWORD1
   KEYWORD2
```
Prerequisite and related information

Repeatable items

An arrow returning to the left above the main line indicates an item that can be repeated.

If you need to specify a separator character (such as a comma) between repeatable items, the line with the arrow returning to the left shows the separator character you must specify.

Fragments

Where it makes the syntax diagram easier to read, a section or fragment of the syntax is sometimes shown separately.

How to send your comments to IBM

We appreciate your input on this publication. Feel free to comment on the clarity, accuracy, and completeness of the information or give us any other feedback that you might have.

Use one of the following methods to send us your comments:
1. Send an email to comments@us.ibm.com
2. Use the form on the Web at:
   www.ibm.com/software/ad/rcf/
3. Mail the comments to the following address:

   IBM Corporation
   DTX/E269
   555 Bailey Avenue
   San Jose, CA
   95141-1003
   U.S.A.

Include the following information:

• Your name and address
Prerequisite and related information

- Your email address
- Your telephone or fax number
- The publication title and order number:
  - FM V13R1 User’s Guide and Reference for DB2 Data
  - SC19-4120-00
- The topic and page number related to your comment
- The text of your comment.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

IBM or any other organizations will only use the personal information that you supply to contact you about the issues that you submit.

If you have a technical problem

Do not use the feedback methods listed above. Instead, do one of the following:
- Contact your IBM service representative
- Call IBM technical support
- Visit the IBM support portal at http://www.ibm.com/systems/z/support/
Prerequisite and related information
Summary of changes

This edition of the document provides information applicable to File Manager Version 13 Release 1. Here are the major changes to this document from the previous edition, for Version 12 Release 1, SC19-3675-01.

- The enhanced section “TSO region size” on page 7.
- An added comment on row size (“Choosing the appropriate editor mode” on page 94).
- The new section “Viewing and changing LOB data and XML documents” on page 97
- The new ability to show the start and end position of the character data displayed in a row. For more information see the Column field description in “SNGL display format” on page 121 and Show char position range field in “Editor Options (3 of 8) panel” on page 535.
- The enhanced section “Using basic SQL prototyping” on page 317.
- The new field Disable input delimiter in “Editor Options (5 of 8) panel” on page 543.
- The new panel “Editor Options (8 of 8) panel” on page 551.
- The new field Additional options in “Set COBOL Processing Options panel” on page 662, “Set HLASM Processing Options panel” on page 663, and “Set PL/I Processing Options panel” on page 667.
- New ONLINE and BATCH options in the “EXPORT primary command” on page 745.
- New “FMAP primary command” on page 752.
- Many new messages.

Changes are indicated by a “l” changebar in the left margin of the page.

The source files for this document were migrated from one source type to another. This has resulted in minor formatting adjustments and other changes. These changes, and other small changes such as minor editorial clarifications, are not listed. Many are tagged with the changebar.
Part 1. Working with FM/DB2
Chapter 1. FM/DB2 overview

FM/DB2 provides a comprehensive, user-friendly set of tools for working with DB2 data. These tools include the familiar view, edit, copy and print utilities found in ISPF, enhanced to meet the needs of application developers.

This chapter provides an overview of the types of actions you can perform with FM/DB2 and detailed concept information about templates, a means of viewing DB2 data in FM/DB2.

What you can do with FM/DB2

FM/DB2 is a powerful set of utility functions for editing, browsing, printing, copying, and maintaining DB2 data. For a description of the DB2 data types that FM/DB2 supports, see Chapter 14, “DB2 data types,” on page 361.

It also provides utilities for listing DB2 objects, managing DB2 privileges, generating JCL to run DB2 standalone utilities, exporting and importing DB2 tables to or from QSAM or VSAM data sets, creating data to populate DB2 tables, and prototyping SQL SELECT statements.

FM/DB2 is an ISPF application program. It uses panels that you can use to select options and to specify parameters, commands and program function (F) keys to simplify requests for common functions, and full-screen format for information display and editing. If data does not fit on the screen, ISPF provides scrolling in all four directions.

Templates

FM/DB2 uses templates to provide a formatted view of your data, enabling you to view, edit, and manipulate data according to the columns and data types in the table you are working with.

What is a template?

An FM/DB2 template is a collection of information that you can use to select and format tables and columns in a DB2 object. If you use an FM/DB2 function that interfaces with non-DB2 data, the corresponding template describes the records and fields in the data set.

When you associate a template with data, you can:

Manipulate columns

You can select which columns you want to work with, adjust the order in which they are shown, and modify the column headings.

You can select to show leading zeros for numeric columns. (By default, leading zeros for numeric columns are suppressed. For example, 00057 is displayed as 57.)

Map data

If you are copying, importing, exporting, or loading data, you can “map” how data is transferred from the source to the destination.
Templates

Select rows
You can specify row selection criteria to restrict the data to the rows you want to work with.

Create data
If you are creating a new table, or adding new columns when moving data, you can specify create attributes that FM/DB2 uses to initialize column values.

If you use an FM/DB2 function that uses a template for a DB2 object, you can select one of the following options:

- Specify the name of an existing template
- Use FM/DB2 to generate a template (using information from the DB2 catalog for the DB2 object)
- Use the template that was last used with the DB2 object
- Use FM/DB2 to generate a template and immediately save the template using a name you specify

Related tasks
Chapter 3, “Working with templates,” on page 49

Where you can use templates
You can use templates to view, browse, edit, print, copy, import, export, create, or load data. The following table lists the panels where you can use templates:

<table>
<thead>
<tr>
<th>Task</th>
<th>Panel</th>
</tr>
</thead>
<tbody>
<tr>
<td>Browse data</td>
<td>View (option B)</td>
</tr>
<tr>
<td>View data</td>
<td>View (option 1)</td>
</tr>
<tr>
<td>Edit data</td>
<td>Edit (option 2)</td>
</tr>
<tr>
<td>Print data</td>
<td>Print (option 3.1)</td>
</tr>
<tr>
<td>Copy data</td>
<td>Copy (option 3.3)</td>
</tr>
<tr>
<td>Import data</td>
<td>Import (option 3.6)</td>
</tr>
<tr>
<td>Export data</td>
<td>Export (option 3.7)</td>
</tr>
<tr>
<td>Create data</td>
<td>Create (option 3.8)</td>
</tr>
<tr>
<td>Load data</td>
<td>Utilities (option 3.9), DB2 LOAD utility</td>
</tr>
<tr>
<td>Unload data</td>
<td>Utilities (option 3.9), DB2 UNLOAD utilities</td>
</tr>
</tbody>
</table>

Note:
1. You can use a template or a copybook with these functions. See “Using templates with non-DB2 data” on page 52

Related tasks
Chapter 4, “Viewing and changing DB2 data,” on page 91
“Printing the contents of a DB2 table” on page 261
“Copying data from one DB2 object to another” on page 193
“Copying data from a VSAM or OSAM file” on page 198
“Copying data to a VSAM or OSAM file” on page 221
Chapter 6, “Populating a DB2 table with data,” on page 191
“LOAD utility” on page 286
“UNLOAD (tables)” on page 306
Templates

“UNLOAD (table spaces)” on page 312

Related references
“DB2 Browse panel” on page 501
“DB2 View panel” on page 518
“DB2 Edit panel” on page 505
“Print Utility panel” on page 621
“Copy Utility (“From”) panel” on page 418
“Import Utility (“From”) panel” on page 591
“Export Utility panel” on page 567
“Data Create Utility panel” on page 495
“LOAD Utility panel” on page 603
“UNLOAD Utility (Tables) “From” panel” on page 718
“UNLOAD Utility (Tables) with LISTDEF panel” on page 719
“UNLOAD Utility (Table Spaces) panel” on page 716
“UNLOAD Utility (Table Spaces) with LISTDEF panel” on page 717
Templates
Chapter 2. Getting started with FM/DB2

This chapter provides answers to questions such as: “How do I enter and exit the application?”, “How do I move around the application?”, “How do I get help?”, and “What do I need to do before I start working?”.

Related tasks

- Starting and exiting FM/DB2
- Selecting the DB2 subsystem when more than one is available
- Using the FM/DB2 interface
- Getting help
- Setting default processing options

Starting and exiting FM/DB2

FM/DB2 is an ISPF-developed application and is usually run from within an ISPF session. Exiting from FM/DB2 returns you to your ISPF session.

Related tasks

- Starting FM/DB2
- Exiting from FM/DB2

TSO region size

The minimum supported TSO region size for FM/DB2 is 16MB. While FM/DB2 can operate with region sizes in the range 4MB - 16MB, these region size specifications restrict the amount of “below the line” storage. If you encounter storage-related abends running with TSO region sizes less than 16 MB, attributable to “below the line” storage shortages, the remedy is to increase the TSO region size.

When processing large DB2 objects, significant amounts of 31-bit storage may be required.

For most applications a TSO region size of 32MB is a suitable starting point, this should be increased if there is a need to process large DB2 objects.

You should not attempt to use the GEN prefix command with a TSO region size less than 16 MB.

Related tasks

- Storage considerations
- Viewing large tables or views

Starting FM/DB2

How you start FM/DB2 depends on how it has been installed on your system. Typically, FM/DB2 is installed as an option on your z/OS Primary Option Menu:
Starting and exiting FM/DB2

1. Select the FM/DB2 option from the z/OS Primary Option Menu.

   **Note:** If FM/DB2 is not an option on your menu, ask your systems administrator to explain the startup process used at your site.

   When you start FM/DB2, the FM/DB2 Primary Option Menu is shown, with the current version information displayed in a pop-up message box.

2. Press Enter to close the version window.

---

![Figure 2. z/OS Primary Option Menu panel showing FM/DB2 option](image)

To start FM/DB2:

1. Select the FM/DB2 option from the z/OS Primary Option Menu.

   **Note:** If FM/DB2 is not an option on your menu, ask your systems administrator to explain the startup process used at your site.

   When you start FM/DB2, the FM/DB2 Primary Option Menu is shown, with the current version information displayed in a pop-up message box.

![Figure 3. FM/DB2 Primary Option Menu panel with version information](image)

2. Press Enter to close the version window.
Checking your FM/DB2 version

To display the full version information for FM/DB2, enter VER on the command line of any panel. The current FM/DB2 version number and the PTF number for each FM/DB2 component is displayed in a window. The pop-up also indicates whether or not FM/DB2 is APF-authorized.

An alternative way to display version information is to select Help > 7. About from the Action Bar on a panel. The current FM/DB2 version number and the PTF number of the DB2 component is displayed. The pop-up panel also shows the copyright information and any notes from IBM that are shipped with the product.

Figure 4. FM/DB2 Primary Option Menu panel with PTF information
### Starting and exiting FM/DB2

**FM/DB2 (DF72) Primary Option Menu**

- **0** Settings Set processing options
- **1** View View DB2 object

**System ID:** FMD2

<table>
<thead>
<tr>
<th>IBM File Manager for z/OS Version x Release x</th>
</tr>
</thead>
<tbody>
<tr>
<td>DB2 Component</td>
</tr>
<tr>
<td>PTF level: xxxxxx</td>
</tr>
</tbody>
</table>

**Note to US Government Users**

(C) Copyright IBM Corporation xxxx, xxxx - All rights reserved

Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract with IBM Corp.

F1=Help F2=Split F3=Exit F9=Swap F12=Cancel

**Command ==> ABOUT**

F1=Help F2=Split F3=Exit F4=Expand F7=Backward F8=Forward

F9=Swap F10=Left F11=Right F12=Cancel

**Related references**

- "VER primary command" on page 781
- "Action bar pull-down menu" on page 365

### Checking which COBOL compiler you're using

FM/DB2 uses the currently loaded COBOL compiler to create a template from a COBOL copybook.

To display details about the current COBOL compiler, enter SHOWCOB on the Command line of any panel. FM/DB2 displays details about the current COBOL compiler in a window.

**Related topics**

- "SHOWCOB primary command" on page 776

### Exiting from FM/DB2

You can exit from FM/DB2 from the Primary Options Menu panel in any of the following ways:

- Press the Exit function key (F3).
- Enter X (or EXIT or END) on the command line.
- Select Process> Exit FM/DB2 from the Action Bar.

To exit the application from a panel within FM/DB2:

- Enter =X on the command line.

**Related references**

- "END primary command" on page 740
- "Action bar pull-down menu" on page 365
Selecting the DB2 subsystem when more than one is available

On most panels, FM/DB2 displays the currently connected DB2 subsystem immediately to the right of the product name. For instance, in Figure 5 on page 12 “FM/DB2 (DFG2)” (in the top lefthand corner) shows that the current DB2 subsystem is DFA2.

If your system contains only one active DB2 subsystem, FM/DB2 automatically connects to that subsystem without you having to select it.

If you are working in a system where there is more than one active DB2 subsystem, you can select the subsystem to which you want FM/DB2 to connect by:

- Entering a value in the DB2 SSID field on the Primary Option Menu, or
- Entering the SSID command in some FM/DB2 panels.

Note:
1. FM/DB2 can only run on a system that has at least one active DB2 subsystem that uses DB2 Version 9, DB2 Version 10, or DB2 Version 11.
2. The first time you use FM/DB2, if your system contains multiple active DB2 subsystems, FM/DB2 immediately displays the DB2 Subsystem Selection panel from which you can select the DB2 subsystem you want. Otherwise, if there is only one active DB2 system in your environment, the Primary Option Menu is displayed with the DB2 subsystem ID already entered in the DB2 SSID field without you having to select it.

Related tasks
- “Selecting the DB2 subsystem with the DB2 SSID field”
- “Selecting the DB2 subsystem by entering the SSID command” on page 13

Related references
- “Primary Option Menu panel” on page 617
- “DB2 Subsystem Selection panel” on page 512

Selecting the DB2 subsystem with the DB2 SSID field

When you start FM/DB2, the Primary Option Menu panel is displayed. You can also display this panel by exiting completely from any FM/DB2 panel; that is, by pressing the Exit function key (F3) the required number of times until FM/DB2 returns you to the Primary Option Menu panel.
Selecting the DB2 subsystem when more than one is available

The **DB2 SSID** field on the Primary Option Menu shows the ID of the DB2 subsystem to which FM/DB2 is currently connected. (The DB2 subsystem is also shown in the top lefthand corner of the panel.) In the example shown in Figure 5, FM/DB2 is connected to the DFA2 subsystem.

To select a DB2 subsystem using the **DB2 SSID** field, you can:
- Overtype the ID of DB2 subsystem currently shown in the **DB2 SSID** field with the ID of the active DB2 subsystem you want, and press Enter. For example, in Figure 5 to change from the current DB2 subsystem to which FM/DB2 is connected (DFG2) to the DB2 subsystem DFB2, overtype DFA2 with DFB2 and press Enter.
- Enter an asterisk (*), or blanks, in the **DB2 SSID** field and press Enter. FM/DB2 displays the DB2 Subsystem Selection panel showing a list of all the DB2 subsystems in your environment from which you can select the DB2 subsystem you want.
- Enter a partial ID, starting or ending with an asterisk (*), or enclosed within two asterisks, in the **DB2 SSID** field and press Enter. FM/DB2 displays the DB2 Subsystem Selection panel showing a restricted list of DB2 subsystems from which you can select the DB2 subsystem you want.

The following examples demonstrate how you can use the (*) wildcard if specifying a partial DB2 subsystem ID:

- **DB*** Lists all DB2 subsystems whose ID starts with “DB”
- **TST** Lists all DB2 subsystems whose ID ends with “TST”
- **2*” Lists all DB2 subsystems whose ID contains “2”

**Note:** To change the DB2 subsystem from a panel other than the Primary Option Menu panel, use the SSID command.

**Related tasks**
- [“Using the DB2 Subsystem Selection menu” on page 14](#)
- [“Selecting the DB2 subsystem by entering the SSID command” on page 13](#)
Selecting the DB2 subsystem when more than one is available

Related references
“Primary Option Menu panel” on page 617
“DB2 Subsystem Selection panel” on page 512

Selecting the DB2 subsystem by entering the SSID command

You can change the currently connected DB2 subsystem by using the SSID command (or its synonym, DB2SYS), followed by the name of the DB2 subsystem to which you want to connect.

Note:
• At the time of installing FM/DB2, the name of this command (or its synonym) may have been customized to something other than SSID (or DB2SYS).

You can verify what the name of the command is by referring to the appropriate Tutorial Help panel:
1. From the Primary Option Menu panel, press the Help function key (F1).
   FM/DB2 displays the “FM/DB2 Tutorial - Table of Contents” panel.
2. Select item D (DB2 SSID).
   FM/DB2 displays the “FM/DB2 Tutorial - DB2 Subsystem Selection” panel.
   This panel shows the name of the command (and its synonym) as set when FM/DB2 was installed.
• You can abbreviate the command name to as few as 4 characters.
• In this document, this command is referred to as the SSID command. If some other name was specified at the time of installing FM/DB2 in your environment, substitute that name wherever reference is made to the SSID command in this document.

For example, to change to the DB27 subsystem, enter the command:
SSID DB27

When you use the SSID command, be aware of the following restrictions:
• The name you specify must be an active DB2 subsystem.
• The name you specify must not contain an asterisk (*) or wildcard.
• You can only use the SSID command from certain panels:
  – Menu panels
  – Most function entry panels
  – Option panels (selected by option 0)
• Once an FM/DB2 dialog has started, you cannot use the SSID command to change the currently connected DB2 subsystem. If you attempt to do so, FM/DB2 issues a “Command not active” message.

To display a list of all the DB2 subsystems in your environment, use the DB2 SSID field on the Primary Option Menu panel.

Related tasks
“Using the DB2 Subsystem Selection menu” on page 14
“Selecting the DB2 subsystem with the DB2 SSID field” on page 11
“Accessing Tutorial Help” on page 34

Related references
“Primary Option Menu panel” on page 617
Using the DB2 Subsystem Selection menu

You use the DB2 Subsystem Selection menu to select the DB2 subsystem to which you want FM/DB2 to connect. It shows a list of the DB2 subsystems in your environment and is displayed:

- The first time you use FM/DB2 (providing your system contains multiple active DB2 subsystems),
- If you enter any of the following in the **DB2 SSID** field on the Primary Option Menu:
  - An asterisk (*)
  - Spaces
  - A partial ID, starting or ending with an asterisk (*), or both

If you enter a partial ID in the **DB2 SSID** field on the Primary Option Menu, the list of DB2 subsystems is restricted accordingly.

**Related references**

- “Primary Option Menu panel” on page 617
- “DB2 Subsystem Selection panel” on page 512

Selecting a subsystem

To select a DB2 subsystem from the DB2 Subsystem Selection menu:

1. Enter S (or /) in the **Sel** column for the DB2 subsystem you want.

   The status of the subsystem you select must be "ACTIVE" or "GROUP", and the subsystem must have been defined in the FM/DB2 installation module. For DB2 systems that do not meet these criteria, the selection field is disabled.

2. Press Enter.

   FM/DB2 displays the Primary Option Menu. The ID of the subsystem you selected is displayed in the **DB2 SSID** field showing that FM/DB2 is now connected to that DB2 subsystem.

**Related references**

- “Primary Option Menu panel” on page 617
- “DB2 Subsystem Selection panel” on page 512

Refreshing the list of DB2 subsystems

To refresh the list of DB2 subsystems on the DB2 Subsystem Selection menu, use the Refresh (F5) function key or enter the primary command REFRESH.

The refreshed list of DB2 subsystems reflects any changes that have occurred since you last displayed the list (for example, the status of each DB2 subsystem).

**Related references**

- “REFRESH primary command” on page 771
- “DB2 Subsystem Selection panel” on page 512

Restricting the list of DB2 subsystems

You can restrict the list of DB2 subsystems shown on the DB2 Subsystem Selection menu by using the SHOW primary command or the Show function key (F6).

Either of the following commands lists all DB2 subsystems:

- SHOW *
- SHOW
Selecting the DB2 subsystem when more than one is available

The following examples demonstrate how you can use the SHOW primary command to restrict the list of DB2 subsystems shown on the DB2 Subsystem Selection menu using the asterisk (*) wildcard and a partial DB2 subsystem ID:

**SHOW MM**
Restricts the list to DB2 subsystems with an ID starting with “MM”

**SHOW *PBJ**
Restricts the list to DB2 subsystems with an ID ending with “PBJ”

**SHOW *V**
Restricts the list to DB2 subsystems with an ID containing “V”

The Show function key (F6) works in the same way as the SHOW command, except that it uses the contents of the command line as the selection criteria. For example:

- If the command line is blank and you press the Show function key (F6), FM/DB2 lists all DB2 subsystems. (You would get the same result if the command line contained an asterisk (*)).
- If you type DBT* on the command line and press the Show function key (F6), FM/DB2 lists all DB2 subsystems whose ID begins with “DBT”.
- If you type *26 on the command line and press the Show function key (F6), FM/DB2 lists all DB2 subsystems whose ID ends with “26”.
- If you type *AB* on the command line and press the Show function key (F6), FM/DB2 lists all DB2 subsystems whose ID contains “AB”.

Related references

- “DB2 Subsystem Selection panel” on page 512
- “SHOW primary command” on page 776

Using the FM/DB2 interface

The FM/DB2 interface is based upon the ISPF model. Tasks are performed by processing a function, together with its applicable parameters. Panels provide a user-friendly way of selecting a function and supplying the parameter information. Some functions can be processed in your choice of "foreground" or "batch" mode. In foreground mode, the selected function is processed immediately and any results are returned to you on screen or directed to print outputs, as dictated by your default settings. In batch mode, JCL is generated from the panel information, and then presented to you for editing. You can modify this code and submit it to a processing queue.

This section provides instructions on how to navigate to panels within FM/DB2 and how to specify common panel information.

Navigating in FM/DB2

You navigate around FM/DB2 panels in the same way as any other panels under ISPF. That is, you launch processing panels from menu panels, by entering the appropriate menu option number on the command line. FM/DB2 uses a Primary Options Menu panel to provide access to processing panels that are related to a particular function or group of functions. In some cases, an option on the Primary Options menu leads to another menu panel, from which you can select the required processing panel.

To access a panel, choose one of the following methods:
Using the FM/DB2 interface

- Type the menu option number on the Command line and press Enter. Repeat for each nested menu until the required processing panel is displayed.
- From the Primary Option Menu, type the complete menu path to the required panel, separating each menu level with a period, then press Enter. For example, entering 3.4 takes you directly to the Object List Utility panel.
- Use "point-and-shoot" for fields on menu panels. To use this facility, place the cursor on the name of the menu option that is required, and press Enter.

**Note:** ISPF provides an "=" parameter that you can specify before an option selection. For example, =3.4. When supported, the "=" parameter enables direct navigation to the specified menu option from another unrelated option within an ISPF application.

The "=" parameter is not supported by FM/DB2.

To exit from a panel, choose one of the following methods:
- Press the Exit function key (F3).
- Type X on the command line and press Enter.
- Type END on the command line and press Enter.

Many FM/DB2 panels have more information than can be seen in a single screen, particularly when you are running in 80x24 display. In this case, a "More" indicator is displayed near the top right corner of the panel with a plus sign (+), a minus sign (-), or both.

![FM/DB2 Tutorial - Ed/Br Options](image)

**Figure 6. Example panel showing "More" message**

A plus sign (+) adjacent to the word "More" indicates more information is available on a following panel. Press the NxtPage function key (F11) to display the next information panel.

A minus sign (-) adjacent to the word "More" indicates more information is available on a previous panel. Press the PrvPage function key (F10) to display the previous information panel.

To scroll down the panel, choose one of these methods:
- Press the Forward function key (F8). This scrolls down one screenful of lines.
- Enter FORWARD or DOWN on the command line. These commands scroll down one screenful of lines.
- Enter BOTTOM on the command line. This takes you to the bottom of the panel.

To scroll up the panel, choose one of these methods:
- Press the Backward function key (F7). This scrolls up one screenful of lines.
- Enter BACKWARD or UP on the command line. These commands scroll up one screenful of lines.
Using the FM/DB2 interface

- Enter **TOP** on the command line. This takes you to the beginning of the panel.

Related references

"Scrolling" on page 125
"BACKWARD primary command" on page 728
"BOTTOM primary command" on page 729
"DOWN primary command" on page 738
"END primary command" on page 740
"FORWARD primary command" on page 753
"TOP primary command" on page 779
"UP primary command" on page 780
"Screen size" on page 19

FM/DB2 panel features

This section describes the general features of FM/DB2 panels.

**Figure 7** shows a typical FM/DB2 panel.

![Typical FM/DB2 panel](image_url)

In general, the following lines are displayed on FM/DB2 panels:

**A Action bar**

Provides access to pull-down menus that give you a fast way to move around the product.

**Note:** The action bar is not displayed if you select Start DB2 Interactive (option 5).

**B Panel title**

Identifies the function being carried out.

**C Row count information(editor sessions only)**

Shows row number (of top row currently displayed) and total number of rows fetched for the current FM/DB2 editor session.

**D Panel body**

Shows information pertaining to the panel.
Using the FM/DB2 interface

**Command line**

Use to enter a command or, on a menu, to enter either a command or an option. The command can be any ISPF command or valid FM/DB2 primary command for the FM/DB2 function being carried out.

To change the position of the command line (the bottom or top of the panel, see the ISPF User’s Guide.

**Function keys**

Displays the settings for the function keys (also known as “Programmable Function” or “PF” keys). For most FM/DB2 panels, these settings are the ISPF function key values. However, for some FM/DB2 panels, these settings are preset to correspond with the needed values for the FM/DB2 function.

For information about default function key values, and how to define your own function key values, see the ISPF User’s Guide.

**Selecting an item from the action bar**

To select an item from the action bar:

1. Move the cursor to the item on the action bar; for example, Options (see Figure 8).
2. Press Enter. FM/DB2 displays a pull-down menu with choices relating to the action bar item you selected.
3. Select a choice from the pull-down menu. Either:
   - Type the pull-down menu selection number (for example, 6 for Print settings), and press Enter, or
   - Move the cursor with the Up or Down arrow keys to the required choice, and press Enter.

An asterisk in front of a choice on a menu indicates that the choice is not available for the current panel.

![Related references](#)
Screen size

FM/DB2 is designed to handle screen sizes larger than 24 x 80 characters.

For information about ISPF settings affecting terminal characteristics, see the ISPF User’s Guide.

Scrollable input and display fields for long names

FM/DB2 uses scrollable fields on FM/DB2 panels to allow you to input and view values longer than the field length.

"There are two types of scrollable fields on FM/DB2 panels:

**Input fields**

The fields on FM/DB2 panels where you can enter information, such as the name of an owner (or creator), the name of a DB2 object, or the name of a DB2 column.

**Display fields**

The fields on FM/DB2 panels that display information as a result of some action you have taken.

"The following two sections describe how to work with both types of scrollable fields."

**Working with scrollable input fields**

On panels where you can enter the name of:

- The owner (or creator),
- A DB2 object, or
- A column

FM/DB2 provides scrollable input fields that allow you to enter values longer than the length of the displayed entry field.

You can recognise a scrollable field by the presence of one of the following adjacent to the end of the entry field:

+ Plus sign. Indicates the field is scrollable to the right
- Minus sign. Indicates the field is scrollable to the left
+- Plus sign and minus sign. Indicates the field is scrollable to the right or left

If the length of the input field displayed on the panel is insufficient for the value you want to enter, perform one of these actions:

- Progressively scroll to the right as you enter the value with the Right function key (F11) or by entering the RIGHT primary command (with the cursor positioned within the entry field),
- Press the Expand function key (F4), or enter the EXPAND primary command, to display a window that allows you to enter the full length of the value

Figure 9 on page 20 shows the format of the DB2 View panel. The “+” symbol adjacent to the Owner and the Name input fields indicates that both are scrollable fields.
Scrollable input and display fields for long names

The Owner input field displayed on the panel allows for up to 16 characters to be entered. To enter the name of an owner that is longer than 16 characters, you must utilize the scrolling or expand capabilities of the Owner input field.

For example, to enter the 21-character name “HANSCHRISTIANANDERSON”, you can perform either of these actions:

- Type the first 16 characters of the name (HANSCHRISTIANAND) in the Owner field, use the left arrow key (洽) to position the cursor back in the Owner input field, press the Right function key (F11) to scroll right, and then type the remainder of the name (ERSON) in the Owner field,

- Press the Expand function key (F4), or enter the EXPAND primary command, to display a window that allows you to enter the full name, and then press the Exit function key (F3) to return to the DB2 View panel.

Figure 10 on page 21 shows the window for the Owner field with the full name entered.
Scrollable input and display fields for long names

When you press the Exit function key (F3) to return to the DB2 View panel, the **Owner** input field shows the first 16 characters of the name you entered:

To see the rest of the name in the **Owner** input field, scroll right by either pressing the Right function key (F11) or entering the RIGHT primary command (ensuring the cursor is positioned within the **Owner** input field):

**Deleting data in scrollable input fields:** If you want to delete all the data in a field that is displayed as a scrollable input field, take care to ensure that you do, in fact, delete all the data (and not just the displayed data).

If you use the field delete key combination to delete data, FM/DB2 deletes only the data that is currently displayed. If the field contains more data than is displayed, the undisplayed data is left undeleted.
Scrollable input and display fields for long names

It is recommended that, before deleting data in a field that is displayed as a scrollable field, you first display the entire contents of the field by pressing the Expand function key (F4) or entering the EXPAND primary command. You can then use the field delete key combination to delete all data in the field at once.

**Working with scrollable display fields**

Many FM/DB2 panels show data displayed in scrollable fields. FM/DB2 uses such fields where the length of the field containing the data to be displayed is longer than the field displayed on the panel.

You can recognise a scrollable field by the presence of one of the following adjacent to the end of the display field:

- **Plus sign**. Indicates the field is scrollable to the right
- **Minus sign**. Indicates the field is scrollable to the left
- **Plus sign and minus sign**. Indicates the field is scrollable to the right or left

If the data to be displayed is longer than the length of the display field on the panel, you can view the rest of the data by performing either of these actions:

- Progressively scrolling to the right by pressing the Right function key (F11) or by entering the RIGHT primary command (with the cursor positioned within the display field),
- Pressing the Expand function key (F4) or entering the EXPAND primary command to display a window that allows you to view the entire contents of the field.

**Figure 11 on page 23** shows the Table Details panel which includes a number of scrollable display fields (those with a “+” symbol adjacent to the end of the field: Table owner, Table name, Created by, Created, and Altered). Scrollable display fields indicate that, possibly, not all of the data held in the field is being displayed.

**Note**: The presence of a “+” symbol indicates that the field is scrollable and may contain more data than is currently being displayed. It does not necessarily mean that more data exists than is currently being shown.
### Scrollable input and display fields for long names

#### Figure 11. Table Details panel: example showing scrollable fields

For example, to see more of the data in the `Created` display field (in the bottom righthand part of the Table Details panel), you can perform either of these actions:

- Scroll right by either pressing the Right function key (F11) or entering the `RIGHT` primary command (ensuring the cursor is positioned within the `Created` field).

The `Created` display field now shows more data:

```
... Data capture . . . : No Created . . . . . . : 1.40.08.268623 -
```

Note that the scroll symbol has changed from a "+" to a ".", indicating that, in this case, you have scrolled to end of the field.

- Press the Expand function key (F4) or enter the EXPAND primary command to display a window that shows the entire contents of the `Created` display field:
Selecting options on FM/DB2 panels

There are several FM/DB2 panels, such as the DB2 View panel shown in Figure 13, where you can select one or more options for the FM/DB2 function you are using.

The options on FM/DB2 panels are displayed in two different formats:

- **A** A list of numbered options in the form of a menu.
  - To select one of these options, type the option number in the entry field provided. For example, in the DB2 View panel shown in Figure 13 to select Previous, type 2.

- **B** One or more options that you can select or deselect.
Scrollable input and display fields for long names

To select one of these options for the current session of the FM/DB2 function only, type a “/” in the entry field provided. To deselect the option, type a space in the entry field.

If the panel shows you can select the option by entering a “/” or an “A” (such as the DB2 View panel):

– To select an option for the current session of the FM/DB2 function only, type a “/”.
– To retain this option as selected from one session of the FM/DB2 function to the next, type an “A”. Use this selection character if you want FM/DB2 to always select this option before continuing processing the FM/DB2 function.

For example, if you always want to edit the template before viewing data, select Edit template on the DB2 View panel using the selection character “A”.

You must select all the required options before you press Enter.

Related references

“DB2 View panel” on page 518

Specifying a DB2 object name

When you use any of the FM/DB2 functions listed below, you specify the name of the DB2 table or view you want to work with in the same way.

• View (option 1)
• Edit (option 2)
• Print (option 3.1)
• Copy (option 3.3)
• Import (option 3.6)
• Export (option 3.7)
• Create (option 3.8)
• Prototyping: Basic (option 4.1)
• Browse (B command on Primary Option Menu panel, or B prefix command against an object in a list of DB2 objects)

Figure 14 on page 26 shows a typical panel containing the entry fields for specifying a DB2 object,
The panels for each of the preceding FM/DB2 functions contain the following entry fields, all of which are optional apart from **Name**:

### Location

The location of the table or view.

If the table or view that you want to work with is at your current location, leave this field blank. Otherwise, type:
- The full name of the remote location
- An asterisk (*) to display all available remote locations, or
- A pattern using wildcard characters

If you specify an asterisk on its own or a pattern, FM/DB2 displays a selection list showing the locations that match the input specification. To select the location you want, type S against the required entry and press Enter. The location selection list is not available unless access to certain DB2 catalog tables has been granted to FM/DB2 users. When the access is not available, you can still access a remote DB2 system by specifying the exact location name.

When you enter a value in this field, synonym object names are not recognized and are not included in the selection list.

It is also possible to specify the location of a DB2 system using:
- The DB2 subsystem ID
- A location nickname defined for the DB2 system

**Note:** In what follows, the use of either the DB2 subsystem ID, or a location nickname, is referred to as an *alternative location reference*.

In order to be able to use these alternatives, the systems programmer must have defined the location value and location nickname in the FMN2SSDM macro definition for each DB2 system (FMN2POPT module). You can check whether a location and location nickname have been defined by selecting **About DB2** from the Help pull-down menu.
If a location value has been specified, you can use the DB2 subsystem ID (for example, DFE2) as an alternative to the actual location name. If location and location nickname values have been specified, you can use the location nickname (for example, V9DEV) as an alternative to the actual location name.

You cannot use wildcards when specifying an alternative location reference. When you specify an alternative location reference, the actual location appears in any location reference displayed by FM/DB2; that is, in the DB2 object name shown in an FM/DB2 editor session. Similarly, any DB2 object name specified in a DB2 batch job is the actual location name. Specifying an alternative location reference in an FM/DB2 batch job is not supported.

You can only access remote locations that have been defined to your local DB2 subsystem.

**Owner**

The authorization ID of the owner of the table, view, or alias.

If you leave this field blank, FM/DB2 uses the current SQL ID. Otherwise, type the full name of the owner, an asterisk (*) to match any owner, or a pattern using wildcard characters.

To specify a synonym in the object name field, either leave this field blank, or specify a generic name that includes the current SQLID, or specify the current SQLID. Synonyms are not recognized or included on the table selection list if any of the table space, database, or location fields are non-blank.

**Note:** DB2 alias names are recognised. DB2 synonyms are not recognised.

**Name**

The name of the DB2 object.

This entry field is mandatory. Type the full name of the DB2 object, an asterisk (*) to display the names of all available objects, or a pattern using wildcard characters.
Scrollable input and display fields for long names

To specify a synonym in this field, type the name of the synonym and ensure that the owner field is either blank, a generic name that includes the current SQLID, or the current SQLID. Synonyms are not recognized or included in the table selection list if any of the table space, database, or location fields are non-blank.

**Database**
The name of the database to which you want to restrict your selection.

Leave this field blank if you do not want to confine the selection to specific databases. Otherwise, type the full name of the database, an asterisk (*) to match any database, or a pattern using wildcard characters.

When you enter a value in this field, synonym names are not recognized or included in the selection list.

**Table space**
The name of the table space to which you want to restrict your selection.

Leave this field blank if you do not want to confine the selection to specific table spaces. Otherwise, type the full name of the table space, an asterisk (*) to match any table space, or a pattern using wildcard characters.

When you enter a value in this field, synonym names are not recognized or included in the selection list.

If you specify an asterisk (*) or a pattern in one or more of the preceding entry fields (Owner, Name, Database, Table space), FM/DB2 displays a selection list showing the DB2 tables and views that match all the selection criteria entered in these fields. The selection list may also include synonyms entries under circumstances discussed above.

To select the table, view, or synonym you want, type $ in the selection column next to the required entry.

For example, say you specified __DB2 in the Owner entry field, *EMP* in the Name entry field, and FM* in the Database entry field. FM/DB2 displays a selection list of DB2 tables and views where:

- The name of the owner is five characters long, with “DB2” as the last three characters,
- The name of the DB2 table or view contains the characters “EMP”, and
- The name of the database starts with “FM”.

To select, say, the view VEMP, type $ in the selection column next to that entry.

Related tasks
- "Using an asterisk (*) or a pattern in entry fields"

Related references
- "Table/View/Alias Selection panel" on page 703
- "DB2 Location Selection panel" on page 509

**Using an asterisk (*) or a pattern in entry fields**
For entry fields on FM/DB2 panels where you can enter an asterisk (*) or wildcards, you can use:

- An asterisk (*) on its own to match all values, or
- A pattern using wildcard characters to select all objects that match the pattern.
In either case, FM/DB2 displays a selection list. To select the listed item you want, type \texttt{S} against the required entry and press Enter.

You can specify a pattern using the following characters:

**asterisk (*) in combination with one or more other wildcard characters**

- Represents a string of zero or more characters. Acts the same as a percent sign (%).

**percent sign (%)**

- Represents a string of zero or more characters. Acts the same as an asterisk (*).

**underscore (_)**

- Represents any single character, including a trailing blank.

**Other characters**

- Any other characters that are valid for the entry field.

The following examples show how various patterns work if used to select entries from the following list:

- AGE
- AGREE
- GUM
- IGLOO
- MAN
- MANE
- MANY
- MANAGE
- MANAGES

**Pattern Entries selected**

<table>
<thead>
<tr>
<th>Pattern</th>
<th>Entries selected</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>AGE</em></td>
<td>AGE, MANAGE, and MANAGES</td>
</tr>
<tr>
<td>MAN_</td>
<td>MAN, MANE, and MANY</td>
</tr>
<tr>
<td>MANAGE%</td>
<td>MANAGE and MANAGES</td>
</tr>
<tr>
<td>_G*</td>
<td>AGE, AGREE, and IGLOO</td>
</tr>
<tr>
<td>%G__</td>
<td>AGE, MANAGE, and MANAGES</td>
</tr>
</tbody>
</table>

**Specifying the SQLID**

You can specify the SQLID to be used as part of the connection to DB2 by typing the required value in the SQL ID field on the FM/DB2 Primary Option Menu panel. The default is your TSO logonid.

You cannot change the SQLID value until FM/DB2 is successfully connected to a DB2 system.

You require authorization to change the SQLID value to something other than your TSO logonid. See your DB2 systems administrator if you require this capability.

You can also change the SQLID value dynamically by using the SQLID primary command.

You can determine the current value for the SQLID by selecting "About DB2" from the Help pull-down menu.
Specifying a data set and a member name

Some FM/DB2 panels require you to specify the data set that you wish to use for the current process. If the data set is a PDS, a CA-Panvalet library, or a library accessed using the Library Management System Exit, you also must enter the member name. These panels are collectively referred to as Entry panels.

To specify your data set on any entry panel:

1. In the **Data set name** field, enter a fully-qualified data set name or a name pattern. The name may include a member name or name pattern in parenthesis. If the member is specified here, the associated Member field must be empty.

   **Note:** In the case of a copybook, this can be the name of a CA-Panvalet library or the name of a library accessed using the Library Management System Exit.

   If you do not enclose the data set name in quotes, then the TSO prefix for your user ID is used as the high-level qualifier.

   If you enter a pattern, when the Entry panel is processed FM/DB2 displays a list of the data sets that match the pattern.

   Patterns can contain a percent sign (%) to represent a single character, and an asterisk (*) to represent any number of characters within one qualifier. Two asterisks (**) represent any number of characters within any number of qualifiers.

   For performance reasons, you should qualify the data set name as much as you can.

2. If you specified the name of a partitioned data set (PDS), the name of a CA-Panvalet library, or the name of a library accessed using the Library Management System Exit, in the **Data set name** field (without including a member name or name pattern in parenthesis), then enter a name or name pattern in the **Member** field.

   A member name pattern can consist of any characters that are valid in a member name and the following two special pattern characters:

   **asterisk (*)**
   
   Represents any number of characters. As many asterisks as required can appear anywhere in a member name. For example, if you enter *d*, a list of all members in the data set whose name contains "d" is displayed.

   **percent sign (%)**
   
   A place-holding character representing a single character. As many percent symbols as necessary may appear anywhere in a member name. For example, if you enter %%%, a list of all members in the data set whose name is four characters in length is displayed.

   If you perform either of these actions:

   - Do not include a member name or name pattern in parenthesis in the **Data set name field**, and leave the **Member** field blank, or you specify a pattern in the **Member** field,
   - Include a member name pattern in parenthesis in the **Data set name field**, and leave the **Member** field blank,
Scrollable input and display fields for long names

FM/DB2 displays a member name list when the Entry panel is processed and the data set has been specified or selected.

3. For data sets which have not been catalogued, enter the serial number of the volume which contains the data set in the **Volume serial** field.

4. Complete the remaining panel-specific fields, and then press Enter to process the panel function.
   - If both the data set name and the member name have been fully specified, the panel action will proceed to its next stage.
   - If you entered a pattern for the data set name or member name, the Data Set Selection panel is displayed, followed by the Member Selection panel, as needed. When you have made your selections from these panels, the Entry panel action will proceed to its next stage.

5. When the Data Set Selection panel is displayed, select your data set by entering an **S** in the **Sel** field adjacent to the required data set. You can only select one name from this list.

6. When the Member Selection panel is displayed, select your members by typing an **S** in the **Sel** field next to each member, and then pressing Enter.
   - You can select as many names as required from this list. When the panel is processed, the first member on the list is used in the relevant function. When the function is completed, the next member in list is used. For example, if you are selecting several members to be edited, the first member is displayed in the Edit Session panel. When you exit this panel, the next member is displayed, and so on, until all selected members have been processed.

**Examples of data set name patterns**

The examples below assume that the following data sets exist under the USERID high-level qualifier:

- USERID.FMN.DATA
- USERID.FMN.TEMPLATES
- USERID.COBO.COPY
- USERID.COBO.SOURCE
- USERID.PLICOPY
- USERID.PLISOURCE
- USERID.MISC.DATA.BACKUP
- USERID.WORK

**Example 1**

Entering the following pattern (with a single asterisk):

'USERID.*'

displays the only item with exactly one level of qualifier after USERID:

USERID.WORK

**Example 2**

Entering the following pattern (with two asterisks):

'USERID.*.*'

displays the complete list of data sets with the USERID high-level qualifier.

**Example 3**

Entering the following pattern:

'USERID.*.%%%'
 displays the following list (containing those data sets with exactly three qualifiers, whose third-level qualifier contains exactly four characters):
USERID.FMN.DATA
USERID.COBOL.COPY
USERID.WRI.COPY

Examples of specifying a member name or pattern

The examples below assume that the following members exist within the data set named, FMNUSER.DATA:
DATA1
FMNCDATA
FMNCTAM
FMNCTEM
FMNCTMP
NEWDATA
NEWSTUFF
TEMPA
TEMPB

Example 1

Entering the following details on an Entry panel:
displays the entire member name list:

```
: 
  Data set name . . . . FMNUSER.DATA
  Member . . . . . . . . . . . . . . . . . . .
: 
```

Example 2

Entering the following details on an Entry panel:
displays the member name list:

```
: 
  Data set name . . . . FMNUSER.DATA
  Member . . . . . . . . . . . . . . . . . . FMNCT*
: 
```

Example 3

Entering the following details on an Entry panel:
Scannable input and display fields for long names

<table>
<thead>
<tr>
<th>Data set name</th>
<th>FMNUSER.DATA(NEWDATA)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Member</td>
<td></td>
</tr>
</tbody>
</table>

displays details for the member NEWDATA in data set FMNUSER.DATA.

**Example 4**

Entering the following details on an Entry panel:
displays the member name list:

<table>
<thead>
<tr>
<th>Data set name</th>
<th>FMNUSER.DATA(TEM*)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Member</td>
<td></td>
</tr>
</tbody>
</table>

TEMPA
TEMPB

**Example 5**

Entering the following details on an Entry panel:
displays the entire member name list:

<table>
<thead>
<tr>
<th>Data set name</th>
<th>FMNUSER.DATA(*)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Member</td>
<td></td>
</tr>
</tbody>
</table>

DATA1
FMNCDATA
FMNCTAM
FMNCTEM
FMNCTMP
NEWDATA
NEWSTUFF
TEMPA
TEMPB

Related references

- “Copy Utility ("From") panel” on page 418
- “Copy Utility ("To") panel” on page 424
- “Data Create Utility panel” on page 495
- “DB2 View panel” on page 518
- “DB2 Browse panel” on page 501
- “DB2 Edit panel” on page 505
- “Export "To" panel” on page 564
- “Export Utility panel” on page 567
- “Import Utility ("To") panel” on page 592
- “Import Utility ("From") panel” on page 591
- “LOAD from panel” on page 601
- “LOAD Utility - Using Templates panel” on page 605
- “UNLOAD Utility (Tables) "To" panel” on page 718
- “Print Utility panel” on page 621
Canceling long-running DB2 queries

On occasions, when you are working with very large amounts of DB2 data, you may experience long response times associated with the FM/DB2 query that FM/DB2 is processing.

When you are using one of the following FM/DB2 utilities, you can use the attention interrupt key to cancel the query.

- View
- Browse
- Edit
- Print
- Copy
- Import
- Export
- Create
- SQL

Pressing the attention interrupt key when DB2 is processing cancels the current DB2 function. Any uncommitted changes made since the start of the unit of work are lost.

Getting help

FM/DB2 offers two levels of help information: Tutorial Help and, for input fields (and some output fields), field-level help information. Tutorial Help is a context-sensitive help system that provides detailed explanations of the FM/DB2 panels and the processes in which they are used. The field-level help information provides a description of the current field. In addition, all error messages have an associated full text explanation, which can help you to determine the cause of a problem.

This section explains how to find the help information that you need.

Related tasks

- "Accessing Tutorial Help"
- "Displaying field-level help information" on page 36
- "Displaying the expanded form of error messages" on page 37

Accessing Tutorial Help

The Tutorial Help system is both context-sensitive and structurally organised. You can get directly to the information you need, using the context-sensitive access method or you can enter the Help system at a specific location and navigate to any topic information within the structure.

To access a context-sensitive Tutorial Help page:

1. Place your cursor on the Command Line or anywhere in the panel that is outside of a field entry line.
2. Press the Help function key (F1). The Tutorial Help page associated with the current panel is displayed.
   When you access the Tutorial Help in this way, the first page that is displayed may be a main topic page, with a menu listing the associated sub-topics, or it may be a sub-topic page, depending upon the context from which you started.
3. Navigate through the Tutorial Help pages, to find the information you require (see below for a list of navigation commands).
Getting help

4. When you have finished, press the Exit function key (F3) to return to your starting panel.

The Tutorial Help system is organised with a Table of Contents structure that is based upon the FM/DB2 Primary Options menu. Each entry within the Table of Contents leads to a main topic with a number of associated sub-topics. In addition, there is an index that lists selected topics from the Tutorial Help, alphabetically by subject.

To choose your starting point in Tutorial Help:
1. Select the Help pull-down menu from a panel Action Bar.
2. Enter the option number for the Tutorial Help entry point that you require. These are:

   1. **Help for help**
      Displays the Help panel for the Tutorial Help system.

   2. **Extended help**
      Displays the Tutorial Help panel associated with the current FM/DB2 panel (equivalent to pressing F1 from the FM/DB2 panel). When on the Primary Options menu, this is the Tutorial Help Table of Contents panel.

   3. **Keys help**
      Displays a panel that provides help for the Function Keys that are active on the current FM/DB2 panel.

   4. **Help index A-M**
      Displays the Help Index for subjects starting with "A" to "M".

   5. **Help index N-Z**
      Displays the Help Index for subjects starting with "N" to "Z".

   6. **Tutorial**
      Displays the Tutorial Help Table of Contents panel.

   7. **About**
      Displays the FM/DB2 version, PTF level, and copyright information in a window.

   8. **About DB2**
      Displays the DB2 subsystem, DB2 version and release information, FM/DB2 plan names, and audit options in a window.

      **Note:** For more information about audit options, see “SAF-rule controlled auditing” on page 47.

   9. **News about FM/DB2**
      Displays a panel providing general information about the current FM/DB2 version/release.

To navigate Tutorial Help, enter one of the following commands in the Command line on any Tutorial page:

**BACK or B**
To back up to the previously viewed page.

**SKIP or S**
To skip the current topic and go on to the next topic.
Getting help

UP or U
To display a higher level list of topics.

TOC or T
To display the Table of Contents.

INDEX or I
To display the Tutorial Index. Once the Index has been displayed, use the Right function key (F11) and the Left function key (F10) to scroll through the list, then position your cursor on a subject and press Enter to display the topic.

Alternatively, you may use the following keys whenever you are in the tutorial:

Enter (Ctrl)
To display the next sequential page within a topic.

Help (F1)
To display the Help Tutorial page within the Tutorial Help system (that is, the page that describes how to use Tutorial Help).

End (F3)
To terminate the tutorial.

Up (F7)
To display a higher level list of topics (instead of typing UP ).

Down (F8)
To skip to the next topic (instead of typing SKIP ).

Right (F11)
To display the next sequential page within a topic (instead of pressing Enter).

Left (F10)
To display the previous sequential page within a topic (instead of typing BACK ).

Note: The listed keys are the default key mappings. As you can customize key mappings, they may be different on your system.

Related references

[Primary Option Menu panel” on page 617]

Displaying field-level help information
To display the field-level help information for a field on an FM/DB2 panel:
1. Place your cursor on the field entry line.
2. Press the Help function key (F1). The field-level help information is displayed in a pop-up window.
   If further information is needed, press the Extended Help function key (F5) to access the Tutorial Help page for the current panel.
   Pressing Help while the field information window is displayed takes you to the Help Tutorial page within the Tutorial Help system (that is, the page that describes how to use Tutorial Help).
3. When you have finished reading the message or the relevant Tutorial Help page, press the Exit function key (F3) to return to your starting panel.
Getting help

For example, Figure 15 shows the message seen when F1 is pressed from the PAGESIZE field in the Set Print Processing Options panel.

Displaying the expanded form of error messages

If an error occurs when FM/DB2 attempts to process a panel, a short text message displays in the upper right corner of the screen. Figure 16 shows an example where FM/DB2 cannot find the DB2 object specified.

Figure 16. DB2 View showing short text message
While this message is displayed, pressing the Help function key (F1) displays the expanded text of the error message at the bottom of the screen as shown in Figure 17.

Some messages, like the one shown in Figure 17 have "Press Help for more information." in the text of the long message. In this case, you can press the Help function key (F1) (again) when the long message is displayed to show a help panel with additional information like that shown in Figure 18 on page 39.

Figure 17. DB2 View showing expanded message

Some messages, like the one shown in Figure 17 have "Press Help for more information." in the text of the long message. In this case, you can press the Help function key (F1) (again) when the long message is displayed to show a help panel with additional information like that shown in Figure 18 on page 39.

Figure 17. DB2 View showing expanded message

Some messages, like the one shown in Figure 17 have "Press Help for more information." in the text of the long message. In this case, you can press the Help function key (F1) (again) when the long message is displayed to show a help panel with additional information like that shown in Figure 18 on page 39.
Setting default processing options

Many of the processing operations performed by File Manager utilize default values that can be set from within the FM/DB2 application. By adjusting these values, you can customize FM/DB2 so that its behavior is best suited to your needs.

When you first use FM/DB2, you will probably find that the default values for processing options (as set at the time of installation) are sufficient. As you become more familiar with working with FM/DB2, you may choose to change some of these options to better suit your needs.

You can update these default values by accessing the relevant processing option panel (for example, the FM/DB2 System Options panel, Editor Options panels, and so on). Changes are saved in your ISPF user profile for future FM/DB2 sessions. They remain in effect until you change the option again.

To access a processing option panel, use either of the following methods:
• From the FM/DB2 Primary Option Menu panel:
  1. Select Option 0 (Settings). The Set Processing Options panel is displayed.
Setting default processing options

2. Select the required processing options type from the menu.

   For example, to display the current FM/DB2 system options, enter 0.2 on
   the command line and press Enter to display the FM/DB2 System Options (1
   of 3) panel.

   **Note:** You can navigate directly to the options panel you want by entering
   the path shown on the Set Processing Options panel (such as 0.2, 3.7, and
   so on). Alternatively, for the system and utility options, you can enter 0 or 3
   to first display the Set System Processing Options or Set Utility Processing
   Options panels respectively from which can then select the option you want.

- From any FM/DB2 panel, use the **Options** pull-down menu to select the
  required processing options type:

```
<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>Set Processing Options</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>System</td>
<td>System options</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>View</td>
<td>View options</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Edit</td>
<td>Edit options</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Utilities</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3.3</td>
<td>Copy</td>
<td>Copy utility options</td>
<td></td>
</tr>
<tr>
<td>3.4</td>
<td>Object List</td>
<td>Object list utility options</td>
<td></td>
</tr>
<tr>
<td>3.7</td>
<td>Export</td>
<td>Export utility options</td>
<td></td>
</tr>
<tr>
<td>3.L</td>
<td>Load</td>
<td>DB2 LOAD utility options</td>
<td></td>
</tr>
<tr>
<td>3.LUL</td>
<td>LISTDEF</td>
<td>DB2 Utility LISTDEF options</td>
<td></td>
</tr>
<tr>
<td>3.U0</td>
<td>OPTIONS</td>
<td>DB2 Utility OPTIONS options</td>
<td></td>
</tr>
<tr>
<td>3.U1</td>
<td>TEMPLATE</td>
<td>DB2 Utility TEMPLATE options</td>
<td></td>
</tr>
<tr>
<td>3.UU</td>
<td>Unload</td>
<td>DB2 Unload utility options</td>
<td></td>
</tr>
</tbody>
</table>
```

**Figure 19. Set Processing Options menu**

- From any FM/DB2 panel, use the **Options** pull-down menu to select the
  required processing options type:

```
| FM/DB2 ( | | | |
| 0 Setti | 1 View | 2 Edit | 3 Utili |
| 3. Job card specifications | 4 Compiler language selection | 5 COBOL compiler specifications | 6 HLAASM compiler specifications |
| 4 SQL | 5 DB2I | 6 Comma | 7 PL/I compiler specifications |
| 8. Temporary Data Set Allocations | 9. Output Data Set Allocations |
| X Exit | 10. ISPF settings | 11. Editor options | 12. Copy utility options |
| 17. DB2 Utility OPTIONS options | 18. DB2 Utility TEMPLATE options | 19. DB2 Unload utility options |
```

**Figure 20. Options pull-down menu**
Setting default processing options

When you select an option processing type, File Manager displays a panel listing those options and their current values. The following notes apply to all of these option panels:

- To change the value of an option, overtype its current value.
- To save the options you have changed and exit the panel, press the Exit function key (F3).
- To exit a panel without saving changes, press the Cancel function key (F12).
- The options on these panels only affect the behavior of FM/DB2 panels.

Related tasks

- "Selecting an item from the action bar" on page 18

Related references

- "Set Processing Options panel" on page 671
- "Set System Processing Options panel" on page 673
- "Set Utility Options panel" on page 674

System Options (option 0)

System options displays the Set System Processing Options panel from which you can select and display subsequent system options panels.

Related references

- "Set System Processing Options panel" on page 673
- "Set Print Processing Options panel" on page 668
- "FM/DB2 Systems Options (1 of 4) panel" on page 577
- "Set Batch Job Card Information panel" on page 661
- "Compiler Language Selection panel" on page 413
- "Set COBOL Processing Options panel" on page 662
- "Set PL/I Processing Options panel" on page 667
- "Set Temporary Data Set Allocation Options panel" on page 676
- "Set Output Data Set Allocation Options panel" on page 666
- "Set Trace options panel" on page 677
- "ISPF Settings panel" on page 598

Print settings (option 0.1)

The Print settings options control such things as the destination of your print output and the page format to use.

Related tasks

- "Printing the contents of a DB2 table" on page 261

Related references

- "Set Print Processing Options panel" on page 668

FM/DB2 system options (option 0.2)

The FM/DB2 system options control such things as the translation of DB2 object names and input SQL statements to uppercase, the use of double quotes for SQL identifiers, and the external format for TIME columns.

Related references

- "FM/DB2 Systems Options (1 of 4) panel" on page 577
- "FM/DB2 Systems Options (2 of 4) panel" on page 578
- "FM/DB2 Systems Options (3 of 4) panel" on page 580
Setting default processing options

Job card specifications (option 0.3)

On the Set Batch Job Card Information panel, you can specify batch job card information to be used for generating batch job submission JCL. The Batch Job Card Information panel provides a number of blank lines (Batch Submission Job Statement Information) where you can enter this information.

FM/DB2 assumes any non-blank lines define a JCL job card and copies the information from these lines into generated JCL. If all the lines are blank, the JCL generation process generates a basic job card using information from system variables.

Related references
“Set Batch Job Card Information panel” on page 661

Compiler language selection (option 0.4)

The Compiler language selection options allow you to select which compiler is used to generate templates from the COBOL or PL/I copybook for FM/DB2 import and export utilities.

Related tasks
“How FM/DB2 compiles a copybook into a template” on page 53
“Copying data from a VSAM or QSAM file” on page 198
“Copying data to a VSAM or QSAM file” on page 221

Related references
“Compiler Language Selection panel” on page 413

COBOL compiler specifications (option 0.5)

The COBOL compiler specifications control various options used by the COBOL compiler to generate templates from COBOL copybooks used by the FM/DB2 import and export utilities.

Related tasks
“How FM/DB2 compiles a copybook into a template” on page 53
“Copying data from a VSAM or QSAM file” on page 198
“Copying data to a VSAM or QSAM file” on page 221

Related references
“Set COBOL Processing Options panel” on page 662

HLASM compiler specifications (option 0.6)

The HLASM compiler specifications control various options used by the HLASM compiler to generate templates from HLASM copybooks used by the FM/DB2 import and export utilities.

Related tasks
“How FM/DB2 compiles a copybook into a template” on page 53
“Copying data from a VSAM or QSAM file” on page 198
“Copying data to a VSAM or QSAM file” on page 221

Related references
“Set HLASM Processing Options panel” on page 663
Setting default processing options

PL/I compiler specifications (option 0.7)
The PL/I compiler specifications control various options used by the PL/I compiler to generate templates from PL/I copybooks used by the FM/DB2 import and export utilities.

Related tasks
- “How FM/DB2 compiles a copybook into a template” on page 53
- “Copying data from a VSAM or QSAM file” on page 198
- “Copying data to a VSAM or QSAM file” on page 221

Related references
- “Set PL/I Processing Options panel” on page 667

Temporary Data Set Allocations (option 0.8)
The temporary data set allocations allow you to specify UNIT, DATA CLASS, STORAGE CLASS, and MANAGEMENT CLASS names for the allocation of a temporary data set.

Related references
- “Set Temporary Data Set Allocation Options panel” on page 676

Output Data Set Allocations (option 0.9)
The output data set allocations allow you to specify UNIT, DATA CLASS, STORAGE CLASS, and MANAGEMENT CLASS names for the allocation of an output data set.

Related references
- “Set Output Data Set Allocation Options panel” on page 666

Trace options (option 0.10)
The Trace options displays the Set Trace options panel.

Related references
- “Set Trace options panel” on page 677

ISPF settings (option 0.11)
The ISPF settings option displays the ISPF Settings panel.

Related references
- “ISPF Settings panel” on page 598

View and Edit options (options 1 and 2)
The View and Edit options (both these options lead you to the first of the Editor Options panels) control various options used to format the display of data when viewing or editing.

**Note:** If you are viewing or editing data, and you use the Options pull-down menu to change these options, then your changes do not take effect until your next FM/DB2 editor session. To temporarily change the behavior for the current FM/DB2 editor session only, use the Edit options option on the DB2 View or DB2 Edit panels.

Related tasks
Utilities (option 3)
Utilities displays the Set Utility Processing Options panel from which you can select and display subsequent utility options panels.

Related references
- “Set Utility Options panel” on page 674
- “Copy Options panel” on page 415
- “Export Options (1 of 3) panel” on page 558
- “LOAD Utility Options panel” on page 607
- “LISTDEF Options panel” on page 599
- “OPTIONS Options panel” on page 612
- “TEMPLATE Options (1 of 2) panel” on page 707
- “UNLOAD Options panel” on page 713

Print utility options (option 3.1)
The Print utility options control such things as print mode (table or single, null column indicator character, batch execution, uncommitted read, and hexadecimal representation.

Related tasks
- “Printing the contents of a DB2 table” on page 261

Related references
- “Print Utility Options panel” on page 619

Copy utility options (option 3.3)
The Copy utility options control such things as concurrency, table locking, and duplicate key processing when copying data between DB2 tables.

Related tasks
- “Copying data from one DB2 object to another” on page 193

Related references
- “Copy Options panel” on page 415

Object List Options (option 3.4)
The object list utility options affect the behavior of catalog table displays produced in the following utility functions:
- Object List utility (3.4)
- Manage DB2 Privileges (3.5)
- Explain Utilities (4.5)

Related references
- “FM/DB2 Object List Options (1 of 2) panel” on page 572
- “FM/DB2 Object List Options (2 of 2) panel” on page 575
Export utility options (option 3.7)
The Export utility options allow you to tailor the way in which the Export utility operates.

Related tasks
“Copying data to a VSAM or QSAM file” on page 221

Related references
“Export Options (1 of 3) panel” on page 558
“Export Options (2 of 3) panel” on page 561
“Export Options (3 of 3) panel” on page 563

DB2 LOAD utility options (option 3.L)
The Load utility options control the way in which FM/DB2 generates JCL for the DB2 LOAD utility.

Related tasks
“LOAD utility” on page 286

Related references
“LOAD Utility Options panel” on page 607

DB2 Utility LISTDEF options (option 3.UL)
The Listdef utility options displays the LISTDEF Options panel.

Related tasks
“Setting options for DB2 utilities” on page 278

Related references
“LISTDEF Options panel” on page 599

DB2 Utility OPTIONS options (option 3.UO)
The Option utility options displays the OPTIONS Options panel.

Related tasks
“Setting options for DB2 utilities” on page 278

Related references
“OPTIONS Options panel” on page 612

DB2 Utility TEMPLATE options (option 3.UT)
The Template utility options displays the TEMPLATE Options panel.

Related tasks
“Setting options for DB2 utilities” on page 278

Related references
“TEMPLATE Options (1 of 2) panel” on page 707
“TEMPLATE Options (2 of 2) panel” on page 708

DB2 Unload utility options (option 3.UU)
The Unload utility options displays the Unload Options panel.
Setting default processing options

The Unload Utility options apply when you use the LOAD utility in Utilities (3.9) to transfer data.

Related tasks

- "Setting options for DB2 utilities" on page 278
- "UNLOAD utility" on page 306

Related references

- "UNLOAD Options panel" on page 713

Displaying or editing DBCS characters

If you want to display or edit DBCS characters in FM/DB2 and you have a DBCS capable terminal, you must ensure that your ISPF settings specify a terminal type that supports these characters.

To specify the Terminal Type in ISPF:

1. Exit from FM/DB2 and return to the z/OS Primary Options Menu panel.
   Although you can change the ISPF Terminal Type setting while FM/DB2 is running, the change will not be reflected until the next time that FM/DB2 is invoked.
2. Select option 0. Settings. The ISPF Settings panel is displayed.

3. Select either 3277KN or 3278KN from the list of Terminal Types. (You may need to scroll down this panel to find the list of Terminal Types.)
4. Press Exit (F3) to save your selection and return to the z/OS Primary Options Menu panel.
5. Restart FM/DB2.

Related tasks

- "Starting FM/DB2" on page 7
- "Exiting from FM/DB2" on page 10
SAF-rule controlled auditing

The way in which auditing is performed during an FM/DB2 session may be determined by SAF rules, depending on the auditing requirement of your installation.

Determining when SAF-rule controlled auditing is in effect

You can determine if SAF-rule controlled auditing is in effect by selecting "About DB2" on the Help pull-down menu:

- When SAF-rule controlled auditing is effect, the Auditing line of the "About DB2" pop-up panel shows "SAF-RULE CONTROLLED".
- When SAF-rule controlled auditing is active and no auditing is to occur, the message indicates that auditing is prohibited.
- When SAF-rule controlled auditing is active and auditing might occur, the message states only that SAF-rule controlled auditing is active.

Note: The final decision as to whether to audit or not depends on the FM/DB2 function and the appropriate SAF XFACILIT rules defined for the DB2 object names, SQL statements, or DB2 commands that are accessed by the FM/DB2 function.

SAF rules for auditing

When auditing is determined by SAF rules:

- Auditing can be specified independently for each DB2 system accessed by FM/DB2, to either your audit log data set, to SMF, or to both destinations.
- All FM/DB2 functions that allow the execution of SQL statements are subject to audit, with the exception of SQL statements used to access the DB2 catalog tables for the internal processes of FM/DB2.
- It determines if and where the Create audit trail (*) option is displayed on FM/DB2 panels. "(*)" is appended to the Create audit trail option to show auditing is under SAF control.

For select statements applying to a DB2 object, SAF rules can be specified at a DB2 object name level allowing audit for READ, UPDATE, or both types of operations. An UPDATE operation includes any SQL statement that changes DB2 data such as INSERT, DELETE, and UPDATE.

SAF-rule controlled auditing not in effect

When SAF-rule controlled auditing is not in effect (that is, auditing is controlled by the settings in the FMN2POPT), the Create audit trail option is displayed on the first global editor options panel. The value shown reflects the current behaviour.

When you display the first local editor options panel, the Create audit trail option is not displayed since it appears on the editor function entry panel.

SAF-rule controlled auditing in effect, no auditing

When SAF-rule controlled auditing is in effect, and no auditing is to occur for the DB2 system, the Create audit trail (*) option is not displayed on either the editor function entry panel, or on the first editor options panel.
SAF-rule controlled auditing in effect, auditing possible

When SAF-rule controlled auditing is in effect, and auditing is to occur for the DB2 system, the actual decision on whether or not to create an audit log depends on the DB2 object name being processed.

Therefore, when you press Enter on the editor function entry panel, a check is made to see if auditing should occur for the current editor type (browse or edit) and the DB2 object name.

If the Create audit trail (*) option is set appropriately (that is, it matches the auditing requirement as specified in the relevant SAF rule), processing continues, the data is displayed, and audit records are written, or not, depending on the option selection.

If you have not selected the Create audit trail (*) option and auditing is required, the function entry panel is re-displayed with a message to indicate that auditing is required. You will not be able to process the object until the audit option is selected.

If you have selected the Create audit trail (*) option and auditing is not required, the function entry panel is re-displayed with a message to indicate that auditing is not required. You will not be able to process the object until the audit option is un-selected.

Note that both the previous scenarios assume that you do not have audit control access for the DB2 object in question. If you do have control level audit access for the DB2 object, the current value of the Create audit trail (*) option determines whether or not audit records are written.

Related references
- “Editor Options (1 of 8) panel” on page 528
- “DB2 Edit panel” on page 505
- “DB2 View panel” on page 518
Chapter 3. Working with templates

FM/DB2 uses templates to provide a formatted view of your data, enabling you to view, edit, and manipulate data according to the columns and data types in the table you are working with.

Initially, FM/DB2 generates a template from the DB2 catalog table. After you have saved the template, you can then use FM/DB2 to enhance it.

The following sections describe how you use templates with FM/DB2.

**Associating a template with a DB2 object**

To use an existing template with an FM/DB2 panel, you supply the name of the template in the template **Data set name** entry field (and optionally the template **Member** entry field) then, in the **Template usage** entry field, select option 1 (Above).

If you have previously used a template with the input DB2 object, you can select option 2 (Previous) instead. The Previous option instructs FM/DB2 to ignore the contents of the Template fields, and instead use the template most recently associated with the input data set. If you want to edit the template before using it, select the **Edit template** field.

To generate a template for the specified DB2 object, select option 3 (Generate from table). This option instructs FM/DB2 to ignore the contents of the Template fields and use information from the DB2 catalog table to generate a template for the DB2 object. This is the default setting.

To generate a template for the specified DB2 object and immediately save the template, enter the name you want FM/DB2 to use for the template in the template **Data set name** entry field (and optionally the template **Member** entry field) then, in the **Template usage** entry field, select option 4 (Generate/Replace).

Related tasks

- “Editing a template” on page 57

Related references

- “Copy Utility ("From") panel” on page 418
- “Copy Utility ("To") panel” on page 424
- “Data Create Utility panel” on page 495
- “DB2 Browse panel” on page 501

© Copyright IBM Corp. 2001, 2013 49
**Information stored in a template**

The following section describes the information that a template can contain, and where and how FM/DB2 uses the information.

**Note:** In this chapter, “transferring data” refers to using any of the following FM/DB2 functions:
- Copy (option 3.3)
- Import (option 3.6)
- Export (option 3.7)
- Utilities (option 3.9), DB2 LOAD utility, DB2 UNLOAD utility

**Row selection criteria**
Determines which rows are to be selected.

Rows that do not match the row selection criteria are excluded when comparing, copying, or printing.

**Column selection**
Determines which columns are displayed (if you are browsing or editing) or printed.

Except for when you are exporting data, this information is ignored when transferring or creating DB2 objects. (When you are copying, importing, exporting, or loading data, if you want to exclude some data in the input from being transferred to the output, you need to “map” the “To” template without those fields.)

**Column sequence**
Overrides the default order in which columns are displayed (if you are browsing or editing) or printed. By default, columns are displayed or printed in the order in which the columns are defined in the object in the DB2 catalog (the COLNO column of SYSIBM.SYSCOLUMNS).

This information is ignored when transferring or creating DB2 objects. (When copying, importing, exporting, or loading data, if you want to exclude some data in the input from being transferred to the output, you need to “map” the “To” template without those fields.)

**Column headings**
Overrides the default column headings when browsing, editing or printing. (The default column headings are the column names defined in the DB2 catalog.)

**Column fixed attributes**
The column name and data type as defined in the DB2 catalog on which the template was based.
Column use attributes
Consist of two attributes:

Output width
Defines the number of positions allocated to the column if you are browsing, editing or printing.

The default width depends on the data type and relates to the “external” format for the data. For example, the default width for a SMALLINT column is 6 bytes (the usual external display size).

If a numeric field contains a valid numeric value, but the value is too large to fit in the output width, then, if you are browsing or editing, the value is truncated and highlighted.

Leading zeros (numeric columns only)
Defines whether or not the column value is shown with leading zeros if you are browsing, editing or printing.

Column create attributes
Defines the value to which the column is initialized if you are transferring data, or creating a table.

If you are transferring data, FM/DB2 uses this information in the “To” template. In a “From” template, this information is ignored.

Column mapping
Defines which columns in the “From” template map to columns in the “To” template if you are transferring data.

If you are transferring data, FM/DB2 uses this information in the “To” template. In a “From” template, this information is ignored.

Scrambling options
Determine how the contents of the column are scrambled (if at all) during a copy, import, or export process.

Note:
1. When copying, FM/DB2 uses this information in the “From” template. In a “To” template, this information is ignored. However, when comparing, the selection information in both the “Old” and “New” templates is used.

Related tasks
"Copying data from one DB2 object to another" on page 193
"Copying data from a VSAM or QSAM file" on page 198
"Copying data to a VSAM or QSAM file" on page 221
"LOAD utility" on page 286
"UNLOAD utility" on page 306
"Mapping data" on page 80

Related references
Chapter 14, “DB2 data types,” on page 361
"Copy Utility ("From") panel" on page 418
"Import Utility ("From") panel" on page 591
"Export Utility panel" on page 562
"Data Create Utility panel" on page 495
"DB2 Utilities panel" on page 515
Printing a template

To print a template, use the TP primary command.

Related tasks

"Looking at the print output from your FM/DB2 session" on page 266

Related references

"TP primary command" on page 779
"PB primary command" on page 766

Using templates with non-DB2 data

Some FM/DB2 functions interface with non-DB2 data:

- You can retrieve non-DB2 data with these FM/DB2 functions:
  - Import (option 3.6) — for retrieving non-DB2 data from a VSAM or QSAM data set.
  - The DB2 LOAD utility in Utilities (option 3.9) — for retrieving data from a sequential data set.

- You can write data to non-DB2 files with these FM/DB2 functions:
  - Export (option 3.7) — for writing data to a VSAM data set, a sequential data set, or a member of a partitioned data set.
  - The DB2 UNLOAD utility in Utilities (option 3.9) — for unloading data to a sequential data set.

Because this is non-DB2 data, FM/DB2 is not able to create a template from the DB2 catalog table. Instead, you have to provide a source for the template. The source is either a (non-DB2) template or a copybook. This describes the records and fields in the data set or member. If you specify a copybook, FM/DB2 compiles it into a template. You can save it and reuse it with any application data set that has the same record structure.

A copybook is a data set containing COBOL, HLASM, or PL/I source statements that describe the record structure of an application data set:

**COBOL data description entries**

Each level-01 group item in the copybook describes a record type in the application data set; elementary items in the group describe fields in the record type.

For example, the following entries describe the record structure of an application data set that contains two record types, ORDER and ITEM:

```
01 ORDERS.
   02 ORDER-ID           PIC X(5).
   02 CUSTOMER-ID        PIC X(5).
   02 ORDER-DATE.
      03 ORDER-YEAR      PIC X(4).
      03 ORDER-MONTH     PIC X(2).
      03 ORDER-DAY       PIC X(2).

01 ITEM.
   02 PRODUCT-ID         PIC X(9).
   02 QUANTITY           PIC 9(4) BINARY.
   02 UNIT-COST          PIC 9(8) BINARY.
```

ORDER records contain five fields: ORDER-ID, CUSTOMER-ID, ORDER-YEAR, ORDER-MONTH and ORDER-DAY. ITEM records contain three fields: PRODUCT-ID, QUANTITY and UNIT-COST.
Using templates with non-DB2 data

**HLASM DSECT definitions**
HLASM copybooks are similar to COBOL copybooks, in that the major structures (DSECT names) describe record types and elementary names describe fields:

ORDER DESCT  
ORDERID DS CL9  
CUSTOMERID DS CL5  
ORDERDATE DS 0CL8  
ORDERYEAR DS CL4  
ORDERMONTH DS CL2  
ORDERDAY DS CL2  
ITEM DESCT  
PRODUCTID DS CL9  
QUANTITY DS H  
UNITCOST DS F

**PL/I DECLARE statements**
Similar to COBOL copybooks, major structures (level-1 names) describe record types, and elementary names describe fields:

```
DECLARE 1 ORDER,  
    2 ORDER_ID CHAR(9),  
    2 CUSTOMER_ID CHAR(5),  
    2 ORDER_DATE,  
    3 ORDER_YEAR CHAR(4),  
    3 ORDER_MONTH CHAR(2),  
    3 ORDER_DAY CHAR(2);  

DECLARE 1 ITEM,  
    2 PRODUCT_ID CHAR(9),  
    2 QUANTITY   BIN(15),  
    2 UNIT_COST  BIN(31);
```

A copybook can contain the entire source of a COBOL, HLASM, or PL/I program, or only COBOL data description entries, HLASM DSECT definitions, or PL/I DECLARE statements. FM/DB2 ignores everything in the copybook except for COBOL data description entries, HLASM DSECT definitions, or PL/I DECLARE statements.

FM/DB2 supports copybooks stored in any of the following:
- PDS
- PDSE
- CA-Panvalet library
- Library accessed using the Library Management System Exit

**Note:**
1. Library members may not be packed by ISPF.
2. For details about enabling File Manager to use a CA-Panvalet library, or a library accessed using the Library Management System Exit, see the [File Manager Customization Guide](#).

**Related tasks**
- "How FM/DB2 compiles a copybook into a template"

**How FM/DB2 compiles a copybook into a template**
If you specify a copybook, FM/DB2:
1. Checks (if the compiler language selected is not HLASM) whether the copybook is a complete source program, or contains only COBOL data declarations or PL/I DECLARE statements. If the copybook is not a complete program, FM/DB2 includes the copybook in a shell program.
Using templates with non-DB2 data

2. Calls the selected compiler to check the syntax of the copybook, and to produce an ADATA file.

   You can use the Settings: Compiler language selection panel (option 0.0.4) to specify whether FM/DB2 calls the COBOL, HLASM, or PL/I compiler.

   For COBOL and PL/I copybooks, you can select the Auto detect option on the Compiler Language Selection panel to instruct File Manager to detect the language of the copybook and call the appropriate compiler. In this case, FM/DB2 analyzes the source to determine whether the language of the source is COBOL or PL/I and creates a template based on the result.

   To instruct File Manager to process an Assembler copybook, you must set the language option on the Compiler Language Selection panel to HLASM. Once you have set the option to HLASM, every function that supports copybooks can be used with an Assembler copybook. Use the HLASM compiler specifications to provide extra SYSLIB and change ALIGN and DBCS processing options.

   You can specify the acceptable return code level for the compilation in the compiler specifications panel for the selected language.

   If the compilation produces a return code with a value less than or equal to the specified maximum, FM/DB2 processes the information in the ADATA file to create a template.

   If the compilation produces a return code with a value greater than the specified maximum, FM/DB2 displays a pop-up menu. From the pop-up menu, you can:

   • View the compilation listing using Print Browse (option 3.11).
   • Cancel the template creation process.
   • Try the compilation again. Before you select this option, first view the compilation listing and correct any errors in the copybook. While you are viewing the compilation listing, you can use the ISPF split screen facility to swap to another ISPF session and use your editor to correct the errors in the copybook.
   • Ignore the errors and proceed with creating the template. This option is only available if FM/DB2 is able to create a template. Some compilation errors, such as warnings, might have no effect on the creation of the template. If you are unsure, view the compilation listing. If the compilation error is too severe, FM/DB2 cannot create a template.
   • Try the compiler for the other language. The option is not available if the HLASM compiler language has been selected.

Related tasks

“Compiler language selection (option 0.4)” on page 42
“Compiler Language Selection panel” on page 413

Related references

“Set COBOL Processing Options panel” on page 662
“Set PL/I Processing Options panel” on page 667
“Set HLASM Processing Options panel” on page 663

Support for COBOL data description entries

In the COBOL data description entries specified in a copybook, FM/DB2 supports:

• ASSIGN clause
• OCCURS clause (including OCCURS DEPENDING ON)
• PICTURE clause (except for numeric editing)
• REDEFINES clause
Using templates with non-DB2 data

- **RENAMES clause**
- **USAGE clause**
- **VALUE clause**

FM/DB2 ignores:
- **BLANK WHEN ZERO clause**
- **DATE FORMAT clause**
- **EXTERNAL clause**
- **GLOBAL clause**
- **JUSTIFIED clause**
- Numeric editing in the PICTURE clause
- **SYNCHRONIZED clause**

FM/DB2 supports all data item level numbers except 88 (condition-names).

**Support for HLASM data description entries**

In the HLASM data description entries specified in a copybook, FM/DB2 produces a record, or segment, for each DSECT it finds with corresponding DS/DC fields.

For more information, see the section "Using HLASM copybooks with File Manager" in the base *File Manager User's Guide and Reference*.

**Support for PL/I data description entries**

In the PL/I data description entries specified in a copybook, FM/DB2 supports:
- PL/I data attributes VARYING and VARYINGZ for single and double-byte character strings.
- PL/I bit variables including the VARYING attribute.
- Binary, decimal, floating point and numeric picture data (excluding symbols).
- REFER variables for array extents. Both upper and lower bound array elements may be declared for each dimension. The number of extents for each dimension is always:
  
  upper bound - lower bound + 1

*Note:* A zero extent is not allowed for a PL/I dimension. If a bound is not specified PL/I assumes a value of 1.

- REFER variable to declare the length of a string variable, character (AN), varying-length character (VC) or zero-terminated character string (ZC) in bytes.
- FM/DB2 also supports PL/I double-byte character variables (DB), varying-length double-byte character (VD) and zero-terminated double-byte characters (Z2) as byte pairs. FM/DB2 also supports bit strings (BT) and varying bit string (VB) variables.
- REFER variables can be declared as integer, fixed binary, packed decimal, floating point, or character string variables.
- UNION declarations at any level which can have varying length (REFER string length or REFER array extents).

*Note:* UNIONs can contain overlays of different data types, which may require template workbench record selection criteria or SHOW control if data is not displayable.

FM/DB2 ignores:
- **INIT data values**
Using templates with non-DB2 data

- REFER start expression, for example,
  
  \text{VAR(, expr Refer abc, ) or CHAR(expr Refer abc).}

- EXTERNAL attributes
- DEFINED structure overlays
- LIKE structure attributes

Note: In general, FM/DB2 treats Numeric Edited fields as alphanumeric. A special code, ZA, is used for unsupported zoned decimals.

Support for variable-length arrays

A variable-length array is defined in a COBOL copybook by an OCCURS DEPENDING ON (ODO) clause, and in a PL/I copybook by a dimension attribute where the lower or upper bounds (or both) are defined by REFER options.

The following considerations apply if you are using a template that contains variable-length arrays:

- A record structure can contain multiple variable-length arrays; however, the fields that define the size of the arrays (the ODO or refer “objects”) must all be in the record before the first variable-length array item.
- If any of the following are true:
  - (COBOL copybook only.) The object is out of the range specified by the ODO clause.
  - The record length does not match the calculated length of a record with the number of array items specified by the object.

then, if you are editing the data set, the prefix area contains the description \text{=LGTH}, indicating that the record was not selected because of incorrect length.

- If you change the value of an object field, then the number of items in the array expands or contracts automatically.

New items are inserted at the end of the current array (and before any fields that follow the array); numeric items are initialized to zero, and alphanumeric items are initialized to blanks.

Similarly, items are deleted from the end of the array, and any fields subsequent to the end of the table are not affected.

For PL/I, the number of items in an array is determined by subtracting the lower bound of the dimension attribute from the upper bound and then adding one. If both the lower and upper bounds are specified using REFER options, changing the value of either of the refer object fields inserts or deletes array items accordingly.

- (COBOL copybook only.) You cannot change the object field to a value outside the range specified by the ODO clause. (FM/DB2 supports an ODO lower value of 0.)
- (COBOL copybook only.) Inserting a new record results in a record with the minimum number of array items specified by the ODO definition. For example:
  - Given the following ODO clause:
    \text{ODO-ONE OCCURS 0 TO 10 TIMES DEPENDING ON ODO-CNTRL-ONE.}

    if you insert a new record, the ODO-CNTRL-ONE field is initialized to zero, with no ODO-ONE array items in the new record.
  - Given the following ODO clause:
    \text{ODO-ONE OCCURS 5 TO 7 TIMES DEPENDING ON ODO-CNTRL-ONE.}
Using templates with non-DB2 data

ODO-CNTRL-ONE is initialized to 5, with five ODO-ONE array items in the new record.

• (PL/I copybook only.) Inserting a new record results in a record with zero array items. Upper and lower bound refer object fields are initialized to zero.

Related tasks
“Copying data from a VSAM or QSAM file” on page 198
“LOAD utility” on page 286
“Copying data to a VSAM or QSAM file” on page 221

Editing a template

FM/DB2 always uses a template to display DB2 data.

The discussion in this section provides examples based on editing a template while you are browsing the sample data shown in Figure 125 on page 690.

You can use either of the following methods to edit a template:

• On an entry panel where you can use templates:
  1. Specify the template you want FM/DB2 to use (by selecting the appropriate Template usage option and, if required, specifying the name of the template).
  2. Select the Edit template field.
  3. Press Enter.

FM/DB2 presents the template for editing, then runs the function after you have finished editing.

If you edit a template in this way, you can change all parts of the template (including row selection criteria).

• If you are using Browse (option B), View (option 1), or Edit (option 2) to browse, view, or edit data, you can use the TEDIT primary command.

Note:

1. You can only use the TEDIT primary command from within an FM/DB2 editor session (if you have the data displayed). You cannot use TEDIT from either the DB2 View or DB2 Edit function panels.

2. If you use the TEDIT primary command, you cannot change the Order, A/D, or Row Selection Criteria input fields.

The reason you cannot alter these parts of the template is that an FM/DB2 editor session is in progress and the session can contain uncommitted changes to data. Altering the preceding parts of the template might require data to be retrieved from DB2 using a new (and possibly different) SQL statement.

When you edit a template (using either of the preceding methods), FM/DB2 displays the Column Selection/Edit panel.

When you have finished editing the template using the Column Selection/Edit panel, you can:

• Use the edited template “on the fly” without saving it by pressing the RunTemp function key (F6),

• Optionally save the edited template before returning to the FM/DB2 function panel by pressing the Exit function key (F3), or
Editing a template

- Return to the FM/DB2 function panel without saving the template by pressing the Cancel function key (F12).

Related tasks
  - “Where you can use templates” on page 4
  - Chapter 4, “Viewing and changing DB2 data,” on page 91

Related references
  - “TEDIT primary command” on page 779
  - “Column Selection/Edit panel” on page 407

Adding a description to a template

When you are creating or modifying a template, you can issue the DESCRIBE primary command to add a description to the template.

To add or modify a description for a template:
1. Display the Column Selection/Edit panel for the template concerned by:
   - Selecting the Edit template field on an entry panel where you can use templates, or
   - Issuing the TEDIT primary command from within an FM/DB2 editor session that is using the template to display the data
2. Enter the DESCRIBE primary command.
   - FM/DB2 displays the DB2 Template Description panel.
3. Enter or modify the description for the template in the Description entry field.
4. Press the Exit function key (F3) to return to the Column Selection/Edit panel.

Related references
  - “DESCRIBE primary command” on page 738
  - “TEDIT primary command” on page 779
  - “Column Selection/Edit panel” on page 407
  - “DB2 Template Description panel” on page 514

Selecting rows

Before you view or edit data, you can select the rows you want to view by editing a template and specifying row selection criteria.

To specify row selection criteria, you must first display the Column Selection/Edit panel by selecting the Edit template field on an entry panel where you can use templates. (If you use the TEDIT primary command to display the Column Selection/Edit panel, you cannot change the row selection criteria.)

You can specify row selection criteria:
- By column, or
- In freeform style

The following sections describe these two methods.

For a full-screen edit, use the SQL primary command or press the SQL function key (F4).
Selecting rows

Row selection by column

To enter row selection criteria by column, type 1 on the command line on the Column Selection/Edit panel and press Enter. FM/DB2 displays the Row Selection Criteria panel.

On the line for each column for which you want to specify selection criteria, you specify an operator (like = or >) in the Op column and, in most case, a value in the Value column.

For example, to select only employees with a surname of SMITH from a table of employees:

1. On the line for the column LASTNAME, type = in the Op column and SMITH (optionally enclosed in single quotes) in the Value column:

2. Press the Exit function key (F3).

   FM/DB2 displays the Column Selection/Edit panel showing the generated row selection criteria (WHERE clause) in the Row Selection Criteria field at the top of the panel:

3. Press Exit function key (F3)

   FM/DB2 displays the selected data:

If you now want to select, say, employees with a surname of SMITH or JONES, you need to modify the row selection criteria. To do this, you must return to the Row Selection Criteria panel.
Selecting rows

1. From the panel of displayed data, press the Exit function key (F3) to return to the DB2 Edit entry panel.
2. Select the Edit template field and press the Exit function key (F3) to display the Column Selection/Edit panel.

You will notice on this panel that the Row Selection Criteria field (showing your previous WHERE "LASTNAME" = 'SMITH' clause) is protected. This is because row selection criteria entered by column or as a free-form expression are not interchangeable. You can change row selection criteria that has been entered by column, by returning to the Row Selection Criteria panel. You can convert row selection criteria entered by column to a free-form expression by issuing the SQL command on the Column Selection/Edit panel, and saving the data. The conversion removes the specification by column. You cannot convert a free-form expression to a row selection criteria entered by column.

3. Enter 1 on the command line to display the Row Selection Criteria panel.
4. To select more than one value for the same column (as in this case, where you want to select two values, SMITH and JONES), there are two methods you can use:

   a. Use multiple lines for the column:
      i. Type the R (Repeat) prefix command in the Cmd column to repeat the line for the column name concerned
      ii. On the repeated line (or lines) for the column name, change the connector (in the Con column) to “OR” by overtyping the AND with 0 or OR
      iii. Type the value in the Value column

   For our example, the selection lines now look like this:

   ```
   Cmd Con Column name Op Value
   AND LASTNAME = SMITH
   OR LASTNAME = JONES
   ```

   b. Use the IN operator:
      i. On the line for LASTNAME, change the operator (from =) to IN. The IN operator lets you specify a list of values you want to select.
      ii. In the value column, type the values you want to select. For alphanumeric columns, when you specify a list of values, each value must be enclosed in quotes and separated by a comma.

   For our example, the selection line now look like this:

   ```
   Cmd Con Column name Op Value
   AND LASTNAME IN 'SMITH','JONES'
   ```

5. Press the Exit function key (F3) twice to display the selected data. Whichever method you used to specify row selection criteria for SMITH or JONES, the same results are achieved:
Selecting rows

Continuing with this same example, if you now want to select employees with a surname of SMITH or JONES who are in department D21, you again need to modify the row selection criteria.

This time, in order to avoid any ambiguity, you may need to use parentheses to define the sequence of evaluation for the resultant WHERE clause. On the Row Selection Criteria panel, you can use the ( and ) columns to enclose groups of expressions in parentheses.

For our example, the selection lines may now look like this:

```
<table>
<thead>
<tr>
<th>Cmd</th>
<th>Con</th>
<th>( Column name</th>
<th>Op</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Column name</td>
<td>Op</td>
<td>Value</td>
</tr>
<tr>
<td></td>
<td></td>
<td>FIRSTNAME</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MIDINIT</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AND</td>
<td>(</td>
<td>LASTNAME</td>
<td>=</td>
<td>SMITH</td>
</tr>
<tr>
<td>OR</td>
<td>LASTNAME</td>
<td>=</td>
<td>JONES</td>
<td></td>
</tr>
<tr>
<td>AND</td>
<td>WORKDEPT</td>
<td>=</td>
<td>D21</td>
<td></td>
</tr>
</tbody>
</table>
```

The selected data now appears as:

```
000000 **** Top of data ****
000001 000210 WILLIAM< T JONES< D11 0942 11.04.1979
000002 000250 DANIEL< S SMITH< D21 0961 30.10.1969
000003 000300 PHILIP< X SMITH< E11 2095 19.06.1972
000004 **** End of data ****
```

Toggling the display (80-character screens only)

On wide screens, FM/DB2 displays both the Column Name and the Data type(length) details for each column on the Row Selection Criteria panel. However, for 80-character screens, by default FM/DB2 displays only the Column Name details.
Selecting rows

To toggle between the **Column Name** and the **Data type**(length) details appearing, press the PF11 function key (the cursor must not be positioned on a **Value** input field).

**Verifying the format of DATE, TIME, and TIMESTAMP columns**

If the row selection criteria you are specifying includes a DATE, TIME, or TIMESTAMP column, the value you specify (in the Value entry field) for that column must be in the correct format.

If you are not sure of the format you should use:
1. Type the required operator (such as “=” or “<”) in the **Op** entry field
2. Type “=” in the Value entry field
3. Press Enter

FM/DB2 displays the field in the format required, showing the current value.

For example, if you enter “=” in the Value entry field for a TIMESTAMP(4) column, FM/DB2 returns the current date and time in the format required for a TIMESTAMP(4) column:

Note that the last two digits are removed. TIMESTAMP(4) means four digits of precision.

Using the same format, you can now overtype the value that FM/DB2 returns with the value you want to use as part of your row selection criteria.

Related tasks

- Scrolling the Row Selection Criteria field on page 64
- Expanding the Row Selection Criteria field on page 65

Related references

- Column Selection/Edit panel on page 407
- Row Selection Criteria panel on page 648

**Row selection criteria in freeform style**

To enter freeform row selection criteria, you use the **Row Selection Criteria** entry field at the top of the Column Selection/Edit panel. The entry field is protected and you cannot update it if:

- It already contains a WHERE clause which was generated by column, or
- You have used the TEDIT command to display the Column Selection/Edit panel.

In this entry field, enter any valid SQL WHERE clause to specify selection criteria to select the data you want to view or edit.

For example, if you have two columns, WORKDEPT and LASTNAME, and you want to display only details of people in department A00 whose family name starts in the range “A” to “M”:  

```sql
Cmd Con { Column name Op Value }
<-> - <--------1--------2--------3-----> <- <---+-+---1--------2--------->
*** **** Top of data ****
;
<table>
<thead>
<tr>
<th>AND - TSTAMP</th>
<th>=</th>
<th>2004-09-24-09.19.36.7749</th>
</tr>
</thead>
</table>
```
1. Enter the following selection criteria:

   WORKDEPT = 'A00' AND LASTNAME BETWEEN 'A%' AND 'M%'

   Alternatively, you can use the FM/DB2 reference numbers as abbreviations for
   the column names. For example, you can write the preceding expression as:
   
   #5 = 'A00' AND #4 BETWEEN 'A%' AND 'M%'

   If you use abbreviations for column names, FM/DB2 expands them to the full
   column name when you press Enter. If you omit the keyword WHERE,
   FM/DB2 inserts it when you press Enter.

   The Column Selection/Edit panel now looks like this:

   ![Primary Column Selection/Edit panel showing row selection criteria](image)

   2. Press the Exit function key (F3).
   
   FM/DB2 displays the selected rows:
Scrolling the Row Selection Criteria field

To scroll the Row Selection Criteria entry field on the Column Selection/Edit panel, you can use the:
• Right function key (F11) and Left function key (F10), or
• RIGHT and LEFT primary commands

In either case, you must position the cursor on the Row Selection Criteria entry field before pressing either function key, or pressing Enter to issue either primary command.

The scroll indicator (+ or -) to the right of the expression indicates which directions you can scroll:
+ Entry field scrollable to the right
- Entry field scrollable to the left

To control how far you scroll, enter a scroll amount in the Scroll field:

Scroll amount

Scroll...

MAX To the first or last position of the expression
PAGE One viewable width of data at a time
HALF Half a viewable width of data at a time
DATA One character less than a viewable width of data at a time
Selecting rows

CSR  When scrolling to the right, FM/DB2 positions the field with the character at the cursor being moved to the leftmost position in the viewable expression.

When scrolling to the left, FM/DB2 positions the field with the character at the cursor being moved to the rightmost position in the viewable expression.

nnnn  nnnn characters at a time

To change the scroll amount, enter a new value into the Scroll field. If you enter MAX, FM/DB2 restores the scroll amount to its previous value after the operation. If you enter any other scroll amount, it remains in effect until you change it again.

Related tasks
"Row selection by column“ on page 59
"Row selection criteria in freeform style“ on page 62

Related references
"RIGHT primary command“ on page 773
"LEFT primary command“ on page 757

Expanding the Row Selection Criteria field

You can expand the Row Selection Criteria entry field by:

- Entering the command SQL, or pressing the SQL function key (F4), to display an ISPF edit session containing the SQL expression
- Using the EXPAND command:
  1. Type EXPAND on the command line
  2. Position the cursor on the Row Selection Criteria entry field
  3. Press Enter.
     FM/DB2 displays an expanded entry panel that allows you to enter row selection criteria up to 32,767 bytes long.

Related tasks
"Row selection by column“ on page 59
"Row selection criteria in freeform style“ on page 62
"Using the SQL Edit session“

Related references
"Column Selection/Edit panel“ on page 407
"EXPAND primary command“ on page 745

Using the SQL Edit session

To start an SQL Edit session, use the SQL primary command. The Edit session contains the SQL statement that FM/DB2 is using to retrieve data from DB2.

You can use the SQL Edit session to change the row selection criteria (WHERE clause), before retrieving data from DB2.

If you use the SQL primary command from within an editor session, FM/DB2 displays a panel such as that shown in Figure 24 on page 66.
If you are editing a template (before displaying data) and you use the SQL primary command (or press the SQL function key (F4)), FM/DB2 displays a panel such as that shown in Figure 25.

You can use the ISPF editor commands CREATE and REPLACE to take a copy of the SQL statement into another, permanent, data set.

Column details for the DB2 object might be shown as note lines (=NOTE=) at the top of the ISPF edit session. These are only displayed if the WHERE clause can be changed.

Related references

"SQL primary command" on page 778
Making changes

The note lines (=NOTE=) at the top of the SQL Edit session indicate whether you can make changes to the SQL statement. When data is displayed in an FM/DB2 editor session, you cannot change the SQL statement without first ending the current editor session. To interactively test and change SQL statements, use Prototyping: Enter (option 4.3), or use the Re-edit template option on the DB2 View or DB2 Edit panel.

You can use FM/DB2 reference numbers as abbreviations for the column names.

When you exit the SQL Edit session by pressing the Exit function key (F3), FM/DB2 converts FM/DB2 reference numbers to the corresponding DB2 column name, and validates the statement. If the validation fails, FM/DB2 redisplayed the SQL statement with an error message.

Note:

1. You cannot change any part of the SQL statement before the WHERE keyword. This includes the list of columns in the SELECT clause (see the following note item).

2. All columns in the table are included in the SELECT clause (following the SELECT keyword) of the SQL statement, even if you have edited the template to deselect some columns. If you deselect a column, it causes FM/DB2 not to display that column; it does not affect the SQL command used by FM/DB2 to retrieve the data from DB2.

3. The maximum length for the WHERE clause that can be specified in the ISPF editor session displayed by the SQL command is 32767 characters.

Related tasks

- Ending an FM/DB2 editor session” on page 101
- “Entering, executing, and explaining SQL statements” on page 351
- “Selecting rows” on page 58
- “Selecting columns” on page 68

Related references

- “DB2 View panel” on page 518
- “DB2 Edit panel” on page 505
- “SQL primary command” on page 778

Saving changes

You can only save any changes you make in the SQL Edit session if the WHERE clause can be changed, and the resultant SQL statement is acceptable to DB2.

When you have made the required changes, press the Exit function key (F3) to redisplay the Column Selection/Edit panel. The row selection criteria section of the panel shows the changes you made to the WHERE clause in the Edit session.

To return to the Column Selection/Edit panel without saving your changes, press the Cancel function key (F12).

Related references

- “Column Selection/Edit panel” on page 407
Selecting rows

Long WHERE clauses
You can use the FM/DB2 SQL Edit session to enter long WHERE clauses (subject to any limits imposed by your host DB2 system). See the DB2 for z/OS SQL Reference.

If the SQL expression is too large to be displayed in the Row Selection Criteria section of the Column Selection/Edit panel, FM/DB2 displays as much of the expression as possible. In this situation, if you make any changes to the expression on the Column Selection/Edit panel, FM/DB2 displays an ISPF Edit session containing the complete SQL expression.

The maximum length for the WHERE clause that can be specified in the ISPF editor session displayed by the SQL command is 32767 characters.

Related references
Column Selection/Edit panel on page 407

Manipulating columns

FM/DB2 offers a number of ways to manipulate columns within rows. Using the Column Selection/Edit panel, you can:

- Select columns
- Change the order in which columns are displayed
- Change column headings
- Access column attributes panels to change individual column attributes

Related tasks
Selecting columns
Changing the sequence of displayed or printed data
Changing the column heading
Changing column attributes

Related references
Column Selection/Edit panel on page 407

Selecting columns

To select which columns are displayed in an FM/DB2 editor session, use the Column Selection/Edit panel. Selected columns are displayed with an “S” in the S column. To toggle between selecting and deselecting a column, enter S in the Cmd column of the Column Selection/Edit panel.

By default, FM/DB2 selects all of the columns in the DB2 object for display, in the order that they are defined to DB2. The initial display shows all columns as selected, with the exception of any ROWID column.

Note: Deselecting a column does not alter the SQL statement that FM/DB2 uses to retrieve data from DB2. In an FM/DB2 editor session, FM/DB2 always generates an SQL statement that specifies every column in the DB2 object.

Toggling the selection status of more than one column at a time
You can toggle between selecting or deselecting more than one column at a time by entering one of the following selection commands:

SS Selects or deselects a block of columns.
**Manipulating columns**

Enter SS twice: first in the **Cmd** column at the start of the block you want to select or deselect, and again at the end of the block. FM/DB2 toggles the selection status of each column in the selected block.

**S** Selects or deselects *n* columns.

Enter **S** in the **Cmd** field of the first column you want to select or deselect. FM/DB2 toggles the selection status of each column, starting from the first column selected for *n* columns.

**S** Selects or deselects a block of columns, starting from the first column selected and then all subsequent columns.

Enter **S** in the **Cmd** field of the first column you want to select or deselect. FM/DB2 toggles the selection status of each column, starting from the first column selected to the last column.

**Note:** If every column in a template is deselected, FM/DB2 selects every column in the template for processing.

**Related tasks**

"**ROWID columns**" on page 76

**Related references**

"**Column Selection/Edit panel**" on page 407

**Holding columns (when scrolling left or right)**

You can hold columns so that, if you scroll left or right in TABL display format, those columns are always displayed on the left of the panel (regardless of how far you scroll). To hold a column, enter **H** in the **Cmd** column of the Column Selection/Edit panel. To free the column, enter **H** again in the **Cmd** column.

If you hold a column, “**H**” is displayed in the **H** column on the Column Selection/Edit panel, and “**1**” is displayed in the **Seq** column to show it is the first held column. If you hold a second column, “**2**” is displayed in the **Seq** column to show it is the second held column, and so on.

**Note:**

1. The sequence numbers shown in the **Seq** column for held columns are independent of any sequence number you might enter to change the sequence of displayed columns. Held columns are always displayed as the leftmost columns on the panel (in the order that you held them when editing the template) and take priority (as far as the left-to-right sequence in which columns are displayed is concerned) over other columns.

2. If you hold a number of columns, there might be little or no room left on the display for non-held columns. This is a particular problem if using an 80-byte display. In this situation you might not be able to scroll right.

**Toggling the hold status of more than one column at a time**

You can toggle between holding or freeing more than one column at a time by entering one of the following hold commands in the **Cmd** column of the Column Selection/Edit panel:

**HH** Holds or frees a block of columns.

Enter **HH** twice: first in the **Cmd** column at the start of the block of columns you want to hold or free, and again at the end of the block. FM/DB2 toggles the hold status of each column in the selected block.
Manipulating columns

**Hn**
Holds or frees n columns.
Enter Hn in the Cmd field of the first column you want to hold or free.
FM/DB2 toggles the hold status of each column, starting from the first column selected for n columns.

Related tasks
- "Changing the sequence of displayed or printed data"

Related references
- "Column Selection/Edit panel" on page 407

**Changing the sequence of displayed or printed data**
You can edit a template to change:
- the order in which columns are displayed or printed
- the sequence in which rows are displayed or printed

Related tasks
- "Changing the sequence of displayed or printed columns"
- "Changing the sequence of displayed or printed rows" on page 71

**Changing the sequence of displayed or printed columns**
By default, FM/DB2 displays or prints columns in the sequence they are defined in the DB2 catalog. This is the same sequence that the column names are displayed in the Column Selection/Edit panel.

To change the order in which columns are displayed or printed, use the Seq field on the Column Selection/Edit panel to specify an alternative order.

**Note:** The Seq field on the Column Selection/Edit panel affects only the sequence in which FM/DB2 displays or prints columns. It has no effect on the sequence in which columns are exported when you use the Export utility function (3.7).

For example, on the Column Selection/Edit panel shown in Figure 26 on page 71, to make the JOB column be shown first, then the HIREDATE column (followed by the remaining columns):
1. In the Seq column against the JOB column, type 1.
2. In the Seq column against the HIREDATE column, type 2.

The Column Selection/Edit panel now looks like this:
3. Press the RunTemp function key (F6).

The data is displayed showing the revised order of the columns:

---

**Related tasks**

- "Copying data to a VSAM or QSAM file" on page 221

**Related references**

- "Column Selection/Edit panel" on page 407

**Changing the sequence of displayed or printed rows**

To change the order in which rows are displayed or printed, use the Order field and, optionally, the A/D field on the Column Selection/Edit panel. For the most significant column, type “1” in the Order field; for the next most significant
Manipulating columns

column, type “2” in the Order field, and so on. If you use the Order field, rows are displayed in ascending sequence by default. To display the rows in descending sequence, type “D” in the A/D field.

If you use either the Order or A/D fields on the Column Selection/Edit panel, FM/DB2 adds an ORDER BY clause to the SQL SELECT statement used to retrieve the data from DB2.

For example, on the Column Selection/Edit panel shown in Figure 27, the Order and A/D fields specify rows to be shown in the sequence LASTNAME (ascending) within WORKDEPT (descending).

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>Column Selection/Edit</td>
<td>Line 1 of 16</td>
<td></td>
</tr>
<tr>
<td>TABLE FMNUSER.EMP</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Row Selection Criteria ---- (Use SQL/PF4 for full screen edit) ----</td>
<td></td>
<td>+</td>
<td></td>
</tr>
<tr>
<td>1 Sel:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cmd</td>
<td>Seq</td>
<td>SHE</td>
<td>CL#</td>
</tr>
<tr>
<td>----</td>
<td>----</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>S 1</td>
<td>EMPNO</td>
<td>CHARACTER(6)</td>
<td>None</td>
</tr>
<tr>
<td>S 2</td>
<td>FIRSTNAME</td>
<td>VARCHAR(12)</td>
<td>None</td>
</tr>
<tr>
<td>S 3</td>
<td>MIDINIT</td>
<td>CHARACTER(1)</td>
<td>None</td>
</tr>
<tr>
<td>S 4</td>
<td>LASTNAME</td>
<td>VARCHAR(15)</td>
<td>None</td>
</tr>
<tr>
<td>S 5</td>
<td>WORKDEPT</td>
<td>CHARACTER(3)</td>
<td>Y</td>
</tr>
<tr>
<td>S 6</td>
<td>PHONENO</td>
<td>CHARACTER(4)</td>
<td>Y</td>
</tr>
<tr>
<td>S 7</td>
<td>HIREDATE</td>
<td>DATE</td>
<td>Y</td>
</tr>
<tr>
<td>S 8</td>
<td>JOB</td>
<td>CHARACTER(8)</td>
<td>Y</td>
</tr>
<tr>
<td>S 9</td>
<td>EDLEVEL</td>
<td>SMALLINT</td>
<td>Y</td>
</tr>
<tr>
<td>S 10</td>
<td>SEX</td>
<td>CHARACTER(1)</td>
<td>Y</td>
</tr>
<tr>
<td>S 11</td>
<td>BIRTHDATE</td>
<td>DATE</td>
<td>Y</td>
</tr>
<tr>
<td>S 12</td>
<td>SALARY</td>
<td>DECIMAL(9,2)</td>
<td>Y</td>
</tr>
</tbody>
</table>

Figure 27. Primary Column Selection/Edit panel showing row sequence selection

When the data is displayed, it appears in the revised sequence:
Manipulating columns

Changing column attributes

To change the attribute of a column, use the Column Selection/Edit panel and enter $E$ in the **Cmd** field against the column. You can enter $E$ against as many columns as you want. For each column you select, FM/DB2 displays a Column Attributes panel.

The attributes you can change are:

- Column heading
- Output width
- Whether to show leading zeros (numeric columns)
- Values to use for creating new columns

**Note:**

1. For numeric, DATE, TIME, and TIMESTAMP columns, you can change the column heading as for an alphanumeric column.
2. For DATE, TIME, and TIMESTAMP columns, you can change the output width as for an alphanumeric column.
3. For very long columns, change the display width to a smaller value to see more data on the display.

The attributes are used to determine how to display, print, or create a column.

**Related references**

- "Changing the attributes for an alphanumeric column" on page 74
- "Changing the attributes for a numeric column" on page 75

Related references

- "Column Selection/Edit panel" on page 407
- "Column Attributes panel (alphanumeric)" on page 397
- "Column Attributes panel (numeric)" on page 402
Changing the attributes for an alphanumeric column

If you enter E in the **Cmd** field for an alphanumeric column, the alphanumeric Column Attributes panel is displayed. You use this panel to change attributes for an alphanumeric column, or to set the pattern if you are creating alphanumeric columns.

**Changing the column heading:** By default, FM/DB2 use the column name (as defined in the DB2 catalog) as the column heading. However, you can change this by editing the template and specifying an alternative column heading.

To change the heading for a column, enter the new heading into the **Heading** entry field.

The column heading might change the width of the display column. For example, in Figure 28 on page 73, to see how the display changes if you change the heading for the MIDINIT column to the longer heading, “MIDDLE INITIAL”:

1. Enter the TEDIT primary command.
   
   FM/DB2 displays the Column Selection Edit panel.

2. On the line for the MIDINIT column, enter E in the **Cmd** field.

   FM/DB2 displays the Column Attributes panel for the MIDINIT column.

3. Type the new heading, **MIDDLE INITIAL**, in the **Heading** entry field.

4. Press the Exit function key (F3).

   FM/DB2 displays the Column Selection/Edit panel.

5. Press the Exit function key (F3) again.

   FM/DB2 displays the Template Save panel.

6. Press the RunTemp function key (F6).

   FM/DB2 displays the data with the new heading.

   Figure 29 on page 75 shows how the change of heading has changed the width of the third column from what it was when originally displayed in Figure 28 on page 73.
Changing the output width: To change the output width, enter the new width into the Output width entry field.

The output width is the number of character positions used by view, edit, browse, and print panels to show the column in TABL display or print format. The default output width is the maximum of the number of characters needed to show the column heading (or column name, if no heading is specified), and the number of characters needed to show the value of the column, as determined from the column definition in the copybook.

The minimum width is six characters. The maximum is 30 or the column width + 10 (whichever is the greater).

Related references

“Column Attributes panel (alphanumeric)” on page 397

Changing the attributes for a numeric column

If you enter E in the Cmd field for a numeric column, the numeric Column Attributes panel is displayed. You use this panel to change attributes for a numeric column, or to set the pattern if you are creating numeric columns.

Changing the column heading: For numeric columns, you change the column heading as for an alphanumeric column.

Related tasks

“Changing the attributes for an alphanumeric column” on page 74

Changing the output width: To change the output width, enter the new width into the Output width column. The output width is the number of character positions used by view, edit, browse, and print panels to show the column in TABL display or print format. The default output width is the maximum of the number of characters needed to show the column heading (or column name, if no heading is specified), and the number of characters needed to show the value of the column, as determined from the column definition in the copybook, including a sign character and decimal point.
Manipulating columns

The allowable widths are 6 to 32 characters.

**Showing leading zeros:** To show leading zeros, enter **YES** into the **Leading zeros** column. The default setting is **NO** (suppress leading zeros). Figure 30 shows the leading zeros shown for SALARY (but not for the other two numeric columns shown).

### Related references

"Column Attributes panel (numeric)" on page 402

### ROWID columns

You can use ROWID columns to create a unique index for tables that have no obvious unique index column or columns.

After a row has been inserted, you cannot update the data in a ROWID column. You can only specify a ROWID value on insert operations for ROWID columns defined as GENERATED BY DEFAULT. See the *DB2 for z/OS SQL Reference*.

When FM/DB2 generates the template for a table containing a ROWID column, the ROWID column is deselected and is not initially displayed in an FM/DB2 editor session.

### Specifying scrambling options

Scrambling data allows you to create test data based on production (or "live") data, but with the ability to change the values of certain columns.

When you use the Copy Utility (option 3.3), Import Utility (option 3.6), or Export Utility (option 3.7), you can choose to scramble some or all of the columns being copied. You specify the columns you want scrambled by marking those columns for scrambling in the output template.

You mark a column for scrambling by setting the scrambling options for that column in the template.
Specifying scrambling options

You can specify or change scrambling options on the Column Attributes panel. The panel comes in two flavors: one for alphanumeric columns and the other for numeric columns.

“Selecting a column for specifying scrambling options” on page 232
“Scrambling data” on page 397
“Column Attributes panel (alphanumeric)” on page 397
“Column Attributes panel (numeric)” on page 402
“Value List Edit panel” on page 722
“Scramble Exit Specification panel” on page 658

Selecting a column for specifying scrambling options

You select a column from the Column Selection/Edit panel. To select a column for specifying scrambling options, enter E in the Cmd field adjacent to the required column. You can enter E against as many columns as you like. For each column you select, File Manager displays a Column Attributes panel.

After you have updated the scrambling options for a column (or any of the other column attributes), when you return to the Column Selection/Edit panel you will see an "E" for that column in the E column (under the "SHE" heading) indicating that the attributes for that column have changed.

The scrambling options you can specify are:

- Scramble type
- Value option
- Value input column
- Value output column
- Minimum and maximum values for a numeric range
- Value data set name

The scrambling options determine how the input data is scrambled (if at all) when it is copied, imported or exported.

Related topics
“Specifying the scramble type”
“Specifying the value option” on page 78
“Specifying value input and output columns” on page 79
“Specifying range values” on page 79
“Specifying a value data set name” on page 79
“Specifying and editing a value list” on page 80

Specifying the scramble type

The Scramble Type option determines how data is scrambled when it is copied, imported, or exported.

Set the scramble type to one of these values:

Blank Data is not scrambled.

1 ("Random")
Data scrambled to produce a random output value on each invocation.

2 ("Repeatable")
Data scrambled to produce the same output value on each invocation.

3 ("Translate")
Data scrambled using input and output values held in the value data set (specified in the Dsn field).
Specifying scrambling options

This option requires you to select the Value option with an input and output column (In and Out) and value data set name (Dsn). All records in the translate data set are loaded into memory for the copy operation. The input and output columns provide the location of the input and output field values as stored in the data set. Their lengths are determined by the respective input and output columns that have been mapped for the copy, import, or export operation. If a matching input column value is found, then the corresponding output column value is obtained from the matching record. If no match is found, then the column is initialized to zero, or the value determined by the data create attributes.

4 ("Exit")

Data scrambled using a scrambling exit (as specified on a separate panel). For details about writing a scrambling exit, see the File Manager Customization Guide.

Related topics
- Specifying the value option
- Specifying value input and out columns on page 79
- Specifying range values on page 79
- Specifying a value data set name on page 79

Specifying the value option

The Value option and corresponding fields control the output of the scrambling process.

The way in which you select, or specify, this option differs according to whether the associated column is alphanumeric or numeric:

- For alphanumeric column, do one of these:
  - Deselect the Value option by entering a blank.
  - Select the Value option by entering a "/".
- For numeric columns, do one of these:
  - Deselect the Value option by entering a blank.
  - Select a range by entering "1".
  - Select the Value option by entering "2".

Value option selected ("/")

Selecting this option allows you to provide:

- A translate data set. This applies when you select Scramble Type 3 ("Translate") with input and output columns and a value data set.
- A lookup data set. This applies when you select a Scramble Type of 1 ("Random") or 2 ("Repeatable") along with a value data set name. The data set is loaded into memory and the output value is randomly or repeatably selected from one of the loaded records. The output column determines the location of the value that is selected for this column.
- A value list. This applies when you select a Scramble Type of 1 ("Random") or 2 ("Repeatable") without a value data set name and allows you to enter the selection values for this column on the panel displayed when you hit Enter. One of the values you provide is randomly or repeatably selected during a copy, import, or export operation. The values you enter are stored in the template. Value lists are described in more detail later.

Value option deselected (blank)
Specifying scrambling options

Deselecting this option scrambles the input column to produce a random or repeatable output value. Scramble Type must be set to 1 ("Random") or 2 ("Repeatable").

**Specifying value input and out columns**
The input column field (In) defines the start location of the input field value on the value data set and is used when the translate process is run during a copy, import, or export operation to match the input column with a value on the value data set. The length of the field is set to the length of the input column that is mapped to this column during the copy, import, or export process.

**Note:** This value is only required when you select Scramble Type 3 ("Translate").

The output column field Out defines the start location of the output field value on the value data set and is used in these ways during a copy operation:
- **Translate.** If an input column value is matched on the value data set, then the corresponding output value is used.
- **Random** or **Repeatable.** The input value is used to randomly, or repeatably, select an output value from the value data set.

**Note:** This value is required when you select Scramble Type 3 ("Translate").

If you select Scramble Type 1 ("Random") or 2 ("Repeatable"), and you have also selected the Value option, then the output column defaults to 1 if you have specified a value data set name (Dsn).

**Specifying range values**
When you are specifying scramble options for a numeric column, you can specify a range of values from which the resultant scrambled value is to be selected.

You specify the minimum value of the range in the Min field and the maximum value of the range in the Max field.

You can only select the range option with random and repeatable scrambling. The output value is randomly or repeatably selected from a number within the specified range.

**Specifying a value data set name**
The output data set name field Dsn field defines the value data set. It can be any catalogued sequential, partitioned or VSAM data set containing data that is used to determine the output column value during a copy, import, or export process.

If you select Scramble Type 3 ("Translate"), then the data set must contain the input and output values in the locations provided in the input and output columns.

If you select Scramble Type 1 ("Random") or 2 ("Repeatable"), then the data set must contain the output value in the locations provided in the output column.

**Note:** This field is required when you select Scramble Type 3 ("Translate"). If you select Scramble Type 1 ("Random") or 2 ("Repeatable"), and also select the Value option, then you can optionally provide a data set name. If you leave this field blank having selected the Value option, you are prompted to enter the value list to be stored in the template.

Chapter 3. Working with templates 79
Specifying scrambling options

Specifying and editing a value list
A value list is a list of values which is used in conjunction with the scrambling options to select a value with which a column is populated during a copy, import, or export operation. When the output column is being populated, one of the values in this list is selected. Scrambling options combine the originating value with the random or repeatable seed to select a value from the list.

For random and repeatable scrambling, you can specify a value list inline by leaving the value data set field (Dsn) blank and selecting the Value option.

To edit the related value list, select the Value option and press Enter. FM/DB2 displays the Value List Edit panel.

The values you provide must be valid for the corresponding data type. Deleting a value list turns off the value list option on the previous panel.

To enter hexadecimal values, use the X'hhhh' format:

x'C1C2C3'

To provide leading blanks or a blank value, use a quoted string:

'SMITH'

Blank value lines (no quotes) are ignored.

Use the FIND and LOCATE commands to bring a value containing or starting with a given string to the current line.

You can use prefix commands to copy, insert, move, repeat or delete lines.

To save the values, press F3. To ignore changes, press Cancel.

Related topics

"Value List Edit panel" on page 722

Mapping data

Mapping tells FM/DB2 how you want columns transferred from an old table to a new table. You can set up mapping so that, in the extreme case, a column is copied to a column of a different name, with a different length, and with a different, but compatible, data type. Most mappings are more straightforward.

Other types of mapping supported by FM/DB2 include:

• Mapping the columns of one table to another.
  You can use this type of mapping with the FM/DB2 Copy utility.

• Mapping the columns of a table to the fields in a file.
  You can use this type of mapping with the FM/DB2 Export utility and the DB2 UNLOAD utility.

• Mapping the fields in a file to the columns of a DB2 table.
  You can use this type of mapping with the FM/DB2 Import utility and the DB2 LOAD utility.
Mapping data

Where you can map data

You can use templates to map data with the following FM/DB2 functions:

- Copy (option 3.3)
- Import (option 3.6)
- Export (option 3.7)
- Utilities (option 3.9)

Related tasks

- "Copying data from one DB2 object to another" on page 193
- "Copying data from a VSAM or QSAM file" on page 198
- "Copying data to a VSAM or QSAM file" on page 221
- "LOAD utility" on page 286

Letting FM/DB2 do the mapping

If you want, you can use FM/DB2 to set up the mappings. This happens automatically if you do not select the Edit template mapping on the Copy “To” panel. If no mapping exists in the “To” template, FM/DB2 performs a corresponding move by going through the following two steps:

1. Generates mappings based on column names. That is, columns in the “From” template are mapped to like-named columns in the “To” template. FM/DB2 provides mappings for as many columns as it can in this way.

2. For the remaining “unmatched” columns in the “From” template (if any), FM/DB2 generates mapping based on data attributes:
   - FM/DB2 compares the attributes of the first unmatched column in the “From” template with the attributes of the first unmatched column in the “To” template. If the attributes of the two columns are the same, FM/DB2 generates mapping between the two columns.
   - FM/DB2 repeats the preceding step with the next pair of unmatched columns in the two templates until either it has generated mapping for all the columns in the “From” template, or if it finds the attributes of the two columns do not match.

Related references

- "Copy Utility (“To”) panel" on page 424
- "Import Utility (“To”) panel" on page 592
- "Export “To” panel” on page 564
- "LOAD from panel" on page 601
- "UNLOAD Utility (Tables) “To” panel” on page 718

Specifying your own mapping

If you want to set up your own mappings, indicate that you want to edit the “To” template by selecting the Edit template mapping option on the “To” panel for the relevant FM/DB2 function. If you select this option, FM/DB2 displays the Template Mapping panel (see Figure 31 on page 82).
Note that, in Figure 31, there is no “From” columns against the “To” columns RID and TSTAMP. This shows that, for these two columns in the “To” template, there is no corresponding column in the “From” template. When the table is copied, FM/DB2 creates new values in RID and TSTAMP.

Related references

- “Template Mapping panel” on page 710
- “Copy Utility (“To”) panel” on page 424
- “Import Utility (“To”) panel” on page 592
- “Export “To” panel” on page 564
- “LOAD from panel” on page 601
- “UNLOAD Utility (Tables) “To” panel” on page 718

### Setting up mapping for a column

To set up or change the mapping for a column, enter S in the Cmd field against the column name. The From Column Mapping panel is displayed.
In Figure 32, the information at the top of the panel shows you the current mapping. In the figure, the **To column** is RID. There is currently no **From column** shown because there is no column in the “From” template with the same name.

To **change** the “From” column, enter **S** against the new “From” column. The information at the top of the panel changes immediately to reflect the new mapping. For example, if you enter **S** against column #6 (PHONENO), the information at the top of the panel changes:

**FM/DB2 (DFG2) From Column Mapping**

- **To DB2 object**: DSN8610.EEMP
- **From DB2 object**: DSN8610.EMP
- **To column**: #1 5R I D
- **From column**: #6 PHONENO

and the **Template Mapping panel** also changes to show that PHONENO is mapped to RID:

**FM/DB2 (DFG2) Template Mapping**

- **To FMUSER.EEMP**
- **From FMUSER.EMP**

<table>
<thead>
<tr>
<th><strong>Cl#</strong></th>
<th><strong>From Column</strong></th>
<th><strong>Data Type(length)</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>EMPNO</td>
<td>CHARACTER(6)</td>
</tr>
<tr>
<td>2</td>
<td>FIRSTNME</td>
<td>VARCHAR(12)</td>
</tr>
<tr>
<td>3</td>
<td>MIDINIT</td>
<td>CHARACTER(1)</td>
</tr>
<tr>
<td>4</td>
<td>LASTNAME</td>
<td>VARCHAR(15)</td>
</tr>
<tr>
<td>5</td>
<td>WORKDEPT</td>
<td>CHARACTER(3)</td>
</tr>
<tr>
<td>6</td>
<td>PHONENO</td>
<td>CHARACTER(4)</td>
</tr>
<tr>
<td>7</td>
<td>HIREDATE</td>
<td>DATE</td>
</tr>
<tr>
<td>8</td>
<td>JOB</td>
<td>CHARACTER(8)</td>
</tr>
<tr>
<td>9</td>
<td>EDLEVEL</td>
<td>SMALLINT</td>
</tr>
<tr>
<td>10</td>
<td>SEX</td>
<td>CHARACTER(1)</td>
</tr>
</tbody>
</table>

**Figure 32. From Column Mapping panel**

There is now an entry in the “From” column against RID.
Mapping data

To remove the current mapping for a “To” column, enter Delete “From column” on the From Column Mapping panel for that column. For example, the Template Mapping panel shown in Figure 31 on page 82 shows that the “To” column BONUS is mapped to the “From” column BONUS. To remove the mapping for BONUS (so that it is not mapped to any “From” column):

1. On the Template Mapping panel shown in Figure 31 on page 82 enter S in the Cmd field against column #13 (BONUS).

The From Column Mapping panel is displayed as shown in Figure 33.

2. Enter S against Delete “From column”.

3. Press Enter.

The “From column” mapping information (fourth line on the From Column Mapping panel) for BONUS is displayed.
Mapping data

4. Press the Exit function key (F3).

The Template Mapping panel is displayed showing the “To” column BONUS is now not mapped to any “From” column.

Note: DB2 determines the default value for an unmapped column according to the definition of the column in the DB2 catalog.

**Changing the attributes for a “To” column**

To change the attributes for a “To” column, enter E against the column name on the Template Mapping panel (Figure 31 on page 82) to display the Column Attributes panel.
Creating data for a column

If no mapping exists for a “To” column, and FM/DB2 is unable to match it to a “From”, FM/DB2 creates data for the column.

Related tasks

“Mapping data” on page 80

Mapping using different templates

The mapping relationship is stored in the “To” template. The mapping information is the column number and the column name. If you reuse a template to copy, import, export, or load data, the mapping information is checked. FM/DB2 ensures that the column in the specified position has the specified name. If this is not the case, FM/DB2 indicates that the mapping has broken down, and gives you the option of either terminating the FM/DB2 function, or updating the mapping information in the “To” template.

This means that you can use a template that is different from the “From” template you used to set up the mapping, provided that the structure of the new template does not differ from the structure of the old template. That is, no columns that have been used in the mapping are in a different position or have a different name. The new “From” template can, however, have different selection conditions.

You can also use a “To” template as a “From” template. However, in this case, the mapping information is not relevant.

Mapping contiguous columns

If you need to map contiguous columns, you might find it more convenient to use the SS, S#, or S* commands.

If you select more than one column at a time, FM/DB2 displays the From Column Mapping panel for the next selected column each time you press the Exit function key (F3). If you are viewing the From Column Mapping panel for the last selected field and press the Exit function key (F3), File Manager returns you to the Template Mapping panel.

Related tasks

“Toggling the selection status of more than one column at a time” on page 68

Related references

“From Column Mapping panel” on page 583
“Template Mapping panel” on page 710

Column mapping rules

FM/DB2 uses the rules shown in Table 2 on page 87 to determine if a given “From” column can be mapped to a given “To” column. “Yes” indicates that the mapping is valid. “No” indicates that the mapping is not valid.

Note: If mapping, you cannot select a “From” column with a data type that is incompatible with the data type of the “To” column. For example, Figure 33 on page 84 shows the two DATE “From” columns, HIREDATE and BIRTHDATE, as un-selectable for mapping to the numeric “To” column, BONUS.
### Table 2. Column mapping rules.

<table>
<thead>
<tr>
<th>“From” column</th>
<th>DATE</th>
<th>TIME</th>
<th>TIMESTAMP</th>
<th>Character</th>
<th>Numeric</th>
</tr>
</thead>
<tbody>
<tr>
<td>DATE</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>TIME</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>TIMESTAMP</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Character</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Numeric</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
</tbody>
</table>

**Notes:**

1. Conversion of TIMESTAMP data type to DATE data type results in truncation of time value. Data type conversion is not supported where the DB2 installation-defined format for dates is LOCAL.
2. Conversion of TIMESTAMP data type to TIME data type results in truncation of date and microsecond values. Data type conversion is not supported where the DB2 installation-defined format for time is LOCAL.
3. Conversion between unlike TIMESTAMP data types may result in loss of precision, or padding with zeroes in the fractional part of the second. Similarly the time zone information may be lost, or the default time zone for the current server may be added. For example:
   - `TIMESTAMP(4) -> TIMESTAMP(10)`
     - `2012-12-02-12.34.56.1234` -> `2012-12-02-12.34.56.1234567890`
   - `TIMESTAMP(4) WITH TIME ZONE -> TIMESTAMP(0)`
     - `2012-12-02-12.34.56.1234-12:00` -> `2012-12-02-12.34.56`
4. The category “character” includes CHAR and VARCHAR data types.
5. Character value must be in a valid format, depending on the DB2 installation-defined formats for date and time, otherwise the “To” column is set to the default value as specified by data create attributes. If the DB2 installation-defined format for dates or time is LOCAL, FM/DB2 is unable to validate the “From” column before inserting it into the “To” column.
6. The “From” column must consist only of numeric characters, and is treated as a numeric column.
7. The category “numeric” includes DECIMAL, NUMERIC, SMALLINT, INTEGER, REAL, DOUBLE, DOUBLE PRECISION, FLOAT, and DECFLOAT data types.

These rules describe how a validly mapped “From” column is moved to a “To” column. If the “To” column is:

**DATE** If the “From” column has a data type of TIMESTAMP, only bytes 1–4 of the “From” column are moved to the “To” column; otherwise, the full four bytes of the “From” column are moved to the “To” column.

**TIME** If the “From” column has a data type of TIMESTAMP, only bytes 5–7 of the “From” column are moved to the “To” column; otherwise, the full three bytes of the “From” column are moved to the “To” column.

**TIMESTAMP**

When the data types of the "FROM" and "TO" columns are identical all bytes of the "From" column are moved to the "To" column.

When the precision and TIME ZONE components of the "FROM" and "TO" columns are different:

- Excess digits are lost in the fraction part of the second when the precision of the "FROM" column is greater than the precision of the "TO" column.
- The fractional part of the second is padded on the right with zeroes when the precision of the "FROM" column is less than the precision of the "TO" column.
Mapping data

- TIME ZONE information is lost when the "FROM" column includes a
  TIME ZONE and the "TO" column does not.
- TIME ZONE information is added when the "FROM" column does not
  include a TIME ZONE and the "TO" column does.

Character
  The data is aligned at the leftmost character position and, if necessary,
  truncated or padded with spaces at the right.
  
  If the “From” column is a numeric column, the absolute (unsigned) value
  is used. If the column is defined as signed and the sign occupies a separate
  character, that character is not moved, and the sending item is considered
  to be one less character than the actual size.

Integer or packed
  The data is aligned on the assumed decimal point and, if necessary,
  truncated or padded with zeros. If an assumed decimal point is not
  explicitly specified in the column definition, one is assumed immediately
  to the right of the column.

  If the “From” column is numeric, the data is moved as if both the “From”
  column and “To” column were described as signed.

  If the “From” column is alphanumeric, the data is moved as if the “From”
  column were described as a signed numeric.

Internal floating-point
  A decimal point is assumed immediately to the left of the column. The
  data is aligned on the leftmost digit position following the decimal point,
  with the exponent adjusted accordingly.

External floating-point
  The data is aligned on the leftmost digit position, and the exponent
  adjusted accordingly.

Related tasks

"Changing column attributes” on page 73

Creating a copybook from an FM/DB2 Template

FM/DB2 provides a sample REXX exec (FMN2CPYB) that you can use to create a
PL/I or COBOL copybook from an FM/DB2 template. The exec is provided in
FMN2SAM1.

To run the exec:
1. Copy FMN2CPYB to a REXX library allocated to your TSO session (for
   example, SYSPROC or SYSEXEC).
2. Issue the statement:

   FMN2CPYB input_file output_file copybook_type

   Where:

   input_file
   The name of an FM/DB2 template. This is typically a member of a
   partitioned data set.

   output_file
   The name of an LRECL=80 sequential file, or member of a partitioned
Creating a copybook from an FM/DB2 Template

data set. The output file contains the copybook derived from the template. Any existing data in the sequential file or member is overwritten.

copybook_type
This parameter is optional. It specifies the type of copybook to be produced. Valid values are:
COBOL
    COBOL copybook created.
PLI
    PL/I copybook created.
Creating a copybook from an FM/DB2 Template
Chapter 4. Viewing and changing DB2 data

To view the contents of a DB2 table or view, use the FM/DB2 View Utility (option 1), the FM/DB2 Edit Utility (option 2), or the FM/DB2 Browse Utility1 (option B).

**FM/DB2 View Utility**

The FM/DB2 View Utility (option 1) allows you to view and temporarily change data, but without the ability to save any changes.

When you initiate the FM/DB2 editor with the View Utility, this is referred to as being in a "View editor session" (or just "View").

The FM/DB2 View Utility, similar to ISPF view, provides an FM/DB2 editor session that works essentially the same as the Edit Utility. It presents you with an interface that is capable of holding your temporary changes to data. This may be useful, for example, for assessing a change but without any risk of altering live data. The first time you make a change to the data in a View editor session, FM/DB2 displays a warning as shown in [Figure 36](#).

When you exit from a View editor session by pressing the Exit function key (F3), if you have made a change to the data in the FM/DB2 editor session, FM/DB2 displays a warning as shown in [Figure 37](#).

---

1. For consistency with ISPF, the FM/DB2 Browse utility does not appear on the FM/DB2 main menu. You can, however, select the FM/DB2 Browse utility by typing the B command.
Viewing and changing DB2 data

**FM/DB2 Edit Utility**

The FM/DB2 Edit Utility (option 2) allows you to view and change your data (by inserting, deleting and otherwise modifying the contents of a DB2 table or view), and save any changes you make.

When you initiate the FM/DB2 editor with the Edit Utility, this is referred to as being in an "Edit editor session" (or just "Edit").

**FM/DB2 Browse**

The FM/DB2 Browse Utility (option B) allows you to view data, but without the ability to save any changes.

When you start up the FM/DB2 editor in browse mode, this is referred to as being in a "Browse editor session" (or just "Browse").

To start up a Browse editor session, perform one of these actions:

- From the Primary Option Menu panel, enter the command:
  
  B

- When you are in an FM/DB2 editor session, enter the command:
  
  BROWSE

- From a list of DB2 objects, where it is legitimate to do so, enter the command:
  
  B

When you start up a Browse editor session, FM/DB2 displays the DB2 Browse entry panel.

Related references

- Chapter 8, "Working with lists of DB2 objects," on page 241
- "Table Browse panel" on page 689
- "Table Edit panel" on page 691

---

2. For consistency with ISPF, the FM/DB2 Browse utility does not appear on the FM/DB2 main menu. You can, however, select the FM/DB2 Browse utility by typing the B command.
These instructions are given within the context of the FM/DB2 editor. In most cases, you can perform these actions in View, Edit, or Browse in which case the term "editor" is used. Similarly, the term "FM/DB2 editor session" can indicate either a View editor session, an Edit editor session, or a Browse editor session.

**FM/DB2 editor sessions**

This topic tells you how to start and end an FM/DB2 editor session, and explains some of the possibilities, such as specifying "normal mode" or "large mode", and validating and saving changes.

**Starting an FM/DB2 editor session**

From the Primary Option Menu panel, to start your FM/DB2 editor session, select one of these:

1. **View**
   - Displays the DB2 View panel.
   - **Note:** This option does not allow you to save any changes you make to the data.

2. **Edit**
   - Displays the DB2 Edit panel.

3. **Browse**
   - Displays the DB2 Browse panel.
   - **Note:** This option does not allow you to save any changes you make to the data.

**Specifying the editor session mode: "normal mode" or "large mode"**

The FM/DB2 editor can operate in two different modes, "normal mode" and "large mode".

**Normal mode**

To operate the FM/DB2 editor session in "normal mode", enter a value other than 0, *, or ALL in the **Row count** field on the editor entry panel.

The FM/DB2 editor retrieves rows from the DB2 object and loads them into the editor until one of these conditions occurs:

- The end of the DB2 object is reached.
- The number of rows loaded into the editor equals the value specified in the **Row count** field.
- Available storage is exhausted.

For information about the amount of storage used by the FM/DB2 editor to store each row, see "Storage considerations" on page 105.
For information about limiting the number of rows loaded into an editor session, see “Restricting the rows to view by specifying row selection criteria” on page 106.

Large mode

To operate the FM/DB2 editor session in "large mode", enter 0, *, or ALL in the Row count field on the editor entry panel. The FM/DB2 editor retrieves rows from the DB2 object and loads them into the editor until one of these conditions occurs:

- The end of the DB2 object is reached.
- Approximately three screen's worth of rows has been loaded.

The number of rows displayed on the editor data panel (in tabular format) depends on the terminal emulation settings (24x80, 27x132, 40x132, and so on) and also the number of optional columns headers.

In "large mode", the FM/DB2 editor keeps only enough rows in memory to satisfy the current display requirements. You can scroll up or down as required to see all rows of the DB2 object.

In an Edit session in "large mode", the FM/DB2 editor also keeps any rows that have been changed in memory, regardless of the current display position. This can increase the storage requirement for the Edit session substantially if many rows are changed. For more information about limiting the number of rows loaded into an editor session, see “Restricting the rows to view by specifying row selection criteria” on page 106.

Choosing the appropriate editor mode

The two editor modes, “normal mode” and “large mode”, have different characteristics as outlined in Table 3. In most cases, you can use either mode to process a DB2 object.

Table 3. Characteristics of editor modes

<table>
<thead>
<tr>
<th></th>
<th>&quot;Normal mode&quot;</th>
<th>&quot;Large mode&quot;</th>
</tr>
</thead>
<tbody>
<tr>
<td>Storage usage</td>
<td>Directly dependent on number of rows loaded, processing objects with large numbers of rows requires very large TSO region size, or not possible.</td>
<td>Independent of number of rows in the DB2 object. Storage use increases as rows in the object are changed.</td>
</tr>
<tr>
<td>DB2 requirements</td>
<td>Initial data load occurs quickly for small DB2 objects, more slowly as the size of the object increases. Once loaded, scrolling within the DB2 object is very fast.</td>
<td>Initial data load occurs quickly for small DB2 objects, significantly more slowly as the size of the object increases. Once loaded, scrolling to nearby locations is fast, scrolling to distant locations may be slow.</td>
</tr>
<tr>
<td>Auto commit feature</td>
<td>Available.</td>
<td>Not available. If selected the auto commit option is disabled. The first attempt to auto commit data changes displays a warning message.</td>
</tr>
<tr>
<td>Data visibility</td>
<td>Once loaded, the data in the editor session is static and does not change, even if other processes alter the data.</td>
<td>When an insensitive cursor is used, the data displayed in the editor session is static and does not change, even if other processes alter the data. When a sensitive cursor is used, the data displayed in the editor session might change if other processes delete or update rows. Rows inserted by other processes are not visible.</td>
</tr>
</tbody>
</table>

Note: For a description of insensitive cursor and sensitive cursor, see “Editor Options (2 of 8) panel” on page 531.
### Specifying the start position for your FM/DB2 editor session

You can load rows from the DB2 object being processed starting at the first row, or you can start at some other point.

The **Start position** entry field on the editor entry panel specifies the required start position. The default is 1.

When the FM/DB2 editor is operating in "normal mode", the editor session contains rows starting at the specified start position. The total number of rows loaded is either the value specified in **Row count**, or some lesser number if the end of the DB2 object was encountered before the required number of rows were loaded. You cannot scroll up to see the rows that were skipped.

When the FM/DB2 editor is operating in "large mode", the editor session initially contains rows starting at the specified start position. You can scroll up to see rows prior to the start position, as far as the first row of the DB2 object if required.

**Usage Note**: You can use the **Start position** value to limit the rows loaded in the editor session, or to start midway through a DB2 object. For example, if you have a large table of, say, a million rows containing people's names ordered by surname and you were only interested in names starting with S, you could enter a start value of, say, 700,000 assuming you knew this is the approximate start position for surnames starting with "S".

### Setting options for the current FM/DB2 editor session

To set options for the current FM/DB2 editor session, display the first of the seven Editor Options panels by entering one of the values shown here against **Edit options** on the DB2 View or DB2 Edit panel:

- **A “/” or an “A”**
  
  Displays the first FM/DB2 Editor Options panel, prior to displaying the data for the DB2 object being processed. When the first Editor Options panel is displayed, you can navigate to one or more of the other Editor Options panels.

  **Usage Note**: Entering an "A" in an option field changes the behavior of the dialog so that the Editor Options panel is ALWAYS displayed. Use the "/" character when you only need to refer to the editor options on a single occasion. The "/" is cleared after the option is processed returning the dialog to the default behavior, which is to NOT show the various Editor Options panels.
FM/DB2 editor sessions

A number in the range 1-8
Displays the n-th FM/DB2 Editor Options panel directly where n is the value entered.

Note: Any options you set on the Editor Options panels apply for the current FM/DB2 editor session only. When you exit from the FM/DB2 editor session, the editor options revert to the global editor options.

In Edit, an exception to this is the audit option if the installation settings for the currently connected DB2 sub-system specifies AUDIT=OPTIONAL, OFF. In this case there is no difference between the audit option set using the global options, or for an individual Edit session. Further, any value specified is discarded at the end of the FM/DB2 session.

Related references
“DB2 View panel” on page 518
“DB2 Edit panel” on page 505
“Editor Options (1 of 8) panel” on page 528

Validating and saving changes with an FM/DB2 editor session
In Edit only, you can make changes to the data and save those changes to DB2.

An Edit session of a DB2 object has important differences when compared to an Edit session of a data set. These differences need to be understood to make the most effective use of an FM/DB2 Edit session.

DB2 data is accessed by calls to DB2. Some important DB2 concepts are discussed here because an understanding of these concepts is central to understanding how the FM/DB2 editor operates:

DB2 unit of work
A DB2 unit of work commences when a process initially connects to, or accesses data from, DB2. The unit of work continues until the process ends the connection or issues an explicit DB2 COMMIT or ROLLBACK statement. Between the start and end of the unit of work, the process may make many calls to DB2 to read (fetch) change (update), insert, or delete rows.

DB2 COMMIT
This is a DB2 command that is used to signal the end of the current unit of work. A DB2 commit occurs automatically when a process ends; however, a DB2 commit may also be issued to finalize any pending changes to DB2 data. Issuing a DB2 commit is an irrevocable step. After the commit, it is not possible to reverse (back out) any changes made since the start of the unit of work.

DB2 ROLLBACK
This is a DB2 command that is used to signal that all pending changes for the current unit of work are to be reversed (backed out). After the rollback is processed, the DB2 data is in the same state that it was in at the start of the unit of work, or at the last commit point.

For more information about DB2 units of work, commit, and rollback, see “DB2 Concepts (Application Processes, concurrency and recovery)” in the SQL Reference manual, for the appropriate version of DB2.
Ordering of rows within a DB2 object is not guaranteed

The relational model on which DB2 is based does not require data within a DB2 table to be in any specific order. This is not to say that data in a DB2 object is not stored in a particular order as this is certainly the case. In addition, a number of methods can be used (for example, CLUSTERING) to influence the way rows are stored in a DB2 table. In terms of the FM/DB2 editor, these considerations apply:

- Data is read from DB2 in the order DB2 delivers it to FM/DB2. Unless there is an ORDER BY clause for the data entered explicitly (on the Column Selection/Edit panel) or implicitly (for example, when processing a VIEW and the VIEW includes an ORDER BY clause), no assumptions should be made about the relative positions of any two rows.
- Moving rows from one position to another in an FM/DB2 Edit session has no effect on the way FM/DB2 stores the data in the table.
- When the editor is operating in "large mode", moving rows from one position to another is disabled.

Viewing and changing LOB data and XML documents

The FM/DB2 editor handles LOB and XML data differently to the data for other columns, primarily because of the potentially very large size for LOB and XML data.

You can use the editor options (see "Editor Options (8 of 8) panel" on page 551) to control how much (if any) of the data in LOB or XML columns is shown in the FM/DB2 editor. By default the column name for LOB or XML data is shown in the FM/DB2 editor, but no data is shown - this minimizes the use of storage by the FM/DB2 editor.

When the DB2 object being processed contains LOB columns, you can view and change the data in a single LOB column in a separate FM/DB2 editor session, initiated using the LOBBRWS, LOBEDIT or LOBVIEW editor primary commands.

When the DB2 object being processed contains XML columns, you can view and change the data in a single XML column in a separate ISPF editor session, initiated using the XMLBRWS, XMLEDIT or XMLVIEW editor primary commands.

An FM/DB2 editor session of the data in a LOB column has some differences when compared with an FM/DB2 editor session of a DB2 object, as described in Table 4. In this table, the term originating session refers to the FM/DB2 editor session of the DB2 object that contains a LOB column; LOB session refers to the FM/DB2 editor session of a LOB column, this is started from the originating session.

**Table 4. LOB edit differences**

<table>
<thead>
<tr>
<th>Action</th>
<th>Object Edit</th>
<th>LOB edit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initiation</td>
<td>From the function entry panel. EDIT, BROWSE, VIEW editor primary commands</td>
<td>Using the LOBBRWS, LOBEDIT, LOBVIEW editor primary commands from within an FM/DB2 editor session of the DB2 object containing the LOB column (Originating session).</td>
</tr>
</tbody>
</table>
### FM/DB2 editor sessions

<table>
<thead>
<tr>
<th>Action</th>
<th>Object Edit</th>
<th>LOB edit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Saving</td>
<td>Changes to rows presented to DB2 one at a time. It is possible that DB2 may reject changes to any row for many reasons.</td>
<td>Any changes made in the LOB session are not presented to DB2 as a change to the DB2 object when the LOB session ends. Instead FM/DB2 uses a LOB locator to store any changes. When a SAVE command is issued in the originating session, any rows that have pending changes to LOB columns are presented to DB2.</td>
</tr>
<tr>
<td>Commit</td>
<td>A DB2 commit may be issued when a SAVE is issued from the originating session (this depends on the editor option’s settings). A DB2 commit is issued at the end of the originating edit session.</td>
<td>No DB2 commits are issued from within a LOB session.</td>
</tr>
<tr>
<td>Data Display</td>
<td>Data is shown initially in either TABL or SNGL format, depending on the settings for the 'Initial display format' editor option. (Displayed on the Editor options (8 of 8) panel.) When data is displayed in SNGL mode, you can use the Previous and Next PFKeys (assigned to PF10 and PF11 by default) to scroll through the LOB data.</td>
<td>Data is shown initially in either TABL or SNGL format, depending on the 'LOB display format' editor option. (Displayed on the Editor options (8 of 8) panel.) When data is displayed in SNGL mode, you can use the Previous and Next PFKeys (assigned to PF10 and PF11 by default) to scroll through the LOB data.</td>
</tr>
<tr>
<td>VARCHAR</td>
<td>The &quot;varying length column&quot; editor options are shown on Editor options (5 of 8) panel. The settings for these options depend on user settings.</td>
<td>The following &quot;varying length column&quot; editor options are fixed: Remove trailing spaces: Off Show end of string: On Convert spaces to DB2 null: Off</td>
</tr>
<tr>
<td>Storage</td>
<td>As described for &quot;normal&quot; or &quot;large&quot; mode, see Table 3.</td>
<td>All of the data for the LOB column is loaded into the FM/DB2 editor. This may not be possible (insufficient storage) with small TSO region sizes and large LOB columns.</td>
</tr>
</tbody>
</table>

The following usage notes may assist in changing data within a LOB column.
- The LOB session is similar to an FM/DB2 editor session of a DB2 object containing a single VARCHAR(*n*) column, where *n* is in the range 1000 - 4000. The values for *n* are approximate and depend on the screen size and other factors.
- The data for a single LOB column is split into pieces as it is loaded in the LOB session. This is purely for convenience, since it makes displaying and changing the LOB data easier, given the limitations of a 3270 terminal.
- You can use the FIND editor command to locate specific text within the LOB session. However a string will not be found if it is split across two "rows" in the LOB session.
- The LOB session initially displays the LOB data in either SNGL or TABL mode, this is determined by the current setting of the 'LOB display format' option. The amount of data shown depends on the 'LOB piece size' option, the terminal's characteristics and and the number of characters in the LOB column's name.
- For large LOBs where the intent is to scroll through the data, consider setting the LOB display option to 'Single', combined with a LOB piece size option of...
‘Single display size’. This allows the maximum use of the available screen size to display the LOB data and the absolute character positions of data is also displayed. You can scroll through the data using the NEXT and PREVIOUS commands (PF11 and PF10).

- For edit operations, consider setting the LOB display option to ‘Table’, combined with a LOB piece size option of ‘Table display size’. In TABL display mode, all of the data for each “row” is completely visible without the need to scroll to the right.
- You can also use Single display as the starting point for edit operations. When the insert point for edit operations is found, switch to TABL display mode and issue the SP(lit) prefix command against the row of interest. This converts the row to many smaller rows that are completely visible in TABL display format.
- Trailing spaces at the end of a “row” are significant, and you should exercise caution in removing these using the editor. When FM/DB2 re-assembles the data for the LOB column, it uses the VARCHAR length for each “row” to determine how much data is added to the LOB column. The editor options for varying length columns (see “Editor Options (5 of 8) panel” on page 543) are set to prevent accidental removal of trailing spaces from a “row”, when the data in that “row” is changed.

Note that the end of the row is shown using the display string delimiter for varying length columns (see “Editor Options (5 of 8) panel” on page 543). You may explicitly set the end for a row using the input string delimiter (see “Editor Options (5 of 8) panel” on page 543).

**XML differences:** FM/DB2 uses the ISPF editor to display and process changes to data in an XML column. The File Manager and ISPF editors are designed to handle different types of data:
- The ISPF editor is primarily a text editor, and is well suited to handling XML data.
- The File Manager editor is primarily a data editor, and is well suited to handling data that has a format that can be described using a template.

An XML document cannot be described using a template.

Table 5 highlights some important differences between an FM/DB2 editor session and an ISPF editor session of an XML document.

**Table 5: XML edit differences**

<table>
<thead>
<tr>
<th>Action</th>
<th>Object Edit</th>
<th>XML edit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initiation</td>
<td>From the function entry panel. Edit, browse and view editor primary commands.</td>
<td>Using the XMLBRWS, XMLEDIT, XMLVIEW editor primary commands from within an FM/DB2 editor session of the DB2 object containing the XML column (Originating session).</td>
</tr>
<tr>
<td>Saving</td>
<td>Changes to rows presented to DB2 one at a time. It is possible that DB2 may reject changes to any row for many reasons. When a SAVE command is issued any changes made to XML columns are presented to DB2.</td>
<td>The XML document is converted into a string when the ISPF editor session ends. A LOB locator is used to store the XML document within DB2 - no attempt is made to update the DB2 object containing the XML document when the ISPF editor session ends.</td>
</tr>
<tr>
<td>Editor used</td>
<td>File Manager DB2 editor</td>
<td>ISPF editor</td>
</tr>
</tbody>
</table>
**FM/DB2 editor sessions**

### Table 5. XML edit differences (continued)

<table>
<thead>
<tr>
<th>Action</th>
<th>Object Edit</th>
<th>XML edit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Commit</td>
<td>A DB2 commit may be issued at the end of SAVE command, depending on the settings for various editor options.</td>
<td>No DB2 commits are issued from within an edit session of an XML document.</td>
</tr>
<tr>
<td>Data Display</td>
<td>Data is shown initially in either TABL or SNGL format, depending on the settings for the 'Initial display format' editor option. (Displayed on the Editor options (1) panel.)</td>
<td>There is only one display format in the ISPF editor. The data is formatted using tag and end-tag pairs prior to display.</td>
</tr>
<tr>
<td>Storage</td>
<td>As described for &quot;normal&quot; or &quot;large&quot; mode, see Table 3 on page 94.</td>
<td>All of the data for the XML document is written to a temporary data set, and then loaded into the ISPF editor. This may not be possible if there are disk space constraints, or with a small TSO region size.</td>
</tr>
</tbody>
</table>

### DB2-related considerations when changing data in an FM/DB2 Edit session

An FM/DB2 Edit session is a DB2 unit of work, commencing at the start of the Edit session and continuing until one of these events occurs:

- The Edit session ends when END is pressed.
- The Edit session ends when CANCEL is pressed.
- A DB2 commit point is issued as part of an FM/DB2 SAVE primary command.
- A DB2 commit point is issued automatically because the number of successful changes reached the **Auto-commit count** value specified on the Editor Options (6 of 8) panel.

Within an FM/DB2 Edit session, multiple changes can be made to the data; you can send these changes to DB2 for validation with the SAVE primary command. At any time, the changes can be backed out (rolled back) to the state the data was in at the start of the Edit session, as long as a DB2 commit point has not been issued. This is unlike an the edit of a data set where typically, once changes have been saved, it is not possible to return to an earlier version of the data.

Changes made during an FM/DB2 Edit session are processed one row at a time. This is unlike the edit of a data set where typically all of the changes made are written all together to the data set when the data is saved and the existing data set is completely replaced by the new one.

Changes made to the data in a DB2 object are validated by DB2 prior to the changes being accepted by DB2. This is unlike changes that can be made to a data set, where typically there is no validation of the data and issuing a SAVE or END primary command is certain to save any pending changes.

DB2 might reject changes for many reasons. These are some, but not all, of the reasons that a change may be rejected:

- Inappropriate values for the data type of a column. For example, entering non-numeric data in a numeric column, or numeric data that is out of range in a numeric column.
**FM/DB2 editor sessions**

- Violation of any check constraints applying to the DB2 object. For example, a column defined as CHAR(1) used to store the gender of a person, constrained to accept a value of only "M" or "F".
- Violation of any referential integrity rules applying to the DB2 object. For example, attempting to delete a primary key for which there are dependent rows in another table, and the DELETE rule for the constraint is RESTRICT.
- Attempts to insert a duplicate with respect to a unique index defined on the DB2 object.

Within an FM/DB2 Edit session, you can use these methods to validate, and optionally commit, pending changes to DB2:

- Issue an explicit SAVE primary command.
- Press the Enter key with no other pending changes (the auto save feature). (See the **Save data** option on “Editor Options (6 of 8) panel” on page 545.)
- Press the END key (normally PF3) to end the Edit session. (See “Ending an FM/DB2 editor session.”)

When you issue a SAVE primary command, any pending changes made are processed one row at a time. Each change results in a single INSERT, UPDATE, or DELETE SQL statement. If the change is unsuccessful, FM/DB2 marks the row in error in the Edit session. You must then correct the error before you can complete (commit) the data changes to DB2.

FM/DB2 may issue DB2 commit points automatically during the SAVE process, if a non-zero value has been specified for the **Auto-commit count**, specified on the Editor Options (6 of 8) panel. Such commit points are issued whenever there have been the specified number of successful changes; it is possible for multiple commit points to be issued during the SAVE process. The commit points are issued regardless of whether errors are encountered when processing other rows.

At the end of the SAVE process, FM/DB2 may also issue an explicit DB2 commit, depending on the setting of the **Commit when save issued** and **Commit when no save errors** options. (See “Editor Options (6 of 8) panel” on page 545.)

**Usage note:** When processing a DB2 object that is subject to constraints, frequent validation of any changes made can avoid frustration at the end of the editor session (when PF3 is pressed). You can issue the SAVE command repeatedly during an Edit session to validate the changes just made, while retaining the ability to cancel the entire Edit session. Remember that, until a DB2 commit point is issued, the entire unit of work can be rolled back. In "normal mode", you also have the option to auto-commit changes when you issue a SAVE primary command. This option is disabled in "large mode".

**Related references**

- “SAVE primary command” on page 773
- “Editor Options (6 of 8) panel” on page 545

**Ending an FM/DB2 editor session**

In View or Browse:

**To end your FM/DB2 editor session:**

Press the Exit function key (F3) or the Cancel function key (F12).

In Edit:
FM/DB2 editor sessions

To end your FM/DB2 editor session and save any changes you have made:
Enter END on the command line, or press the Exit function key (F3).

To present the changes you have made to the current table to DB2 for validation, without ending the FM/DB2 editor session:
Use the SAVE primary command.

When you use the SAVE command, FM/DB2 processes any changes made one line at a time. Each change results in either an SQL INSERT, UPDATE or DELETE operation, DB2 may reject one of more of your changes. This is unlike an editor session of a normal data set, where generally all changes are made, or no changes are made.

Note the following about the SAVE process:
• FM/DB2 issues DB2 COMMIT points according to these commit options:
  – Commit when save issued
  – Commit when no save errors
  – Auto-commit count
The settings for these options are either the global settings, or as specified for this FM/DB2 editor session only. The first two options influence whether a DB2 COMMIT is issued at the end of the SAVE process. The third option influences whether DB2 COMMIT points are issued during the SAVE process.
• Issuing a DB2 COMMIT finalizes all pending changes to the data, and is an irrevocable process. After the COMMIT is issued, you can cancel the editor session; however, any changes made since the start of the edit session are not backed out. Conversely, if no COMMIT has been issued, you can back out all changes made during the current editor session, regardless of how many SAVE commands have been issued.
• When the editor is operating in "large mode", any automatic DB2 COMMIT is disabled. Your changes are still presented to DB2 and validated; however, the final COMMIT is deferred until you end the editor session.

To end your FM/DB2 editor session without saving your changes (that is, since the previous successful commit point):
Enter CANCEL (or CAN) on the command line, or press the Cancel function key (F12). If you have made any changes to the data, FM/DB2 displays a message box asking you to confirm that you want to discard the changes.

Note:
1. If there are dependent related FM/DB2 editor sessions still active:
   • You need to end these sessions before you end the current (parent) FM/DB2 editor session (unless you cancel the session).
   • Canceling the current FM/DB2 editor session also cancels all the dependent related FM/DB2 editor sessions.

Related tasks
“Editing related tables” on page 149
“Specifying the editor session mode: "normal mode" or "large mode"” on page 93

Related references
“CANCEL primary command” on page 730
“SAVE primary command” on page 775
“Editor Options (6 of 8) panel” on page 545
Starting another FM/DB2 editor session

When you are in an FM/DB2 editor session, you can start another FM/DB2 editor session by issuing the BROWSE, EDIT, VIEW, or REDIT primary commands.

**BROWSE, EDIT, VIEW commands**

The BROWSE, EDIT, and VIEW primary commands start another FM/DB2 editor session and display the Browse, Edit, or View entry panel respectively.

The new FM/DB2 editor session is stacked on top of the current FM/DB2 editor session. If you specify a DB2 object owner and DB2 object name, the owner and name fields are pre-filled on the entry panel for the new FM/DB2 editor session.

**Note:** To start an FM/DB2 editor session in Browse mode from the Primary Option Menu panel, enter the abbreviated form of the BROWSE command, B, on the command line.

**Examples:**

- **VIEW** Starts a new FM/DB2 editor session, in View mode.
- **BRO DSN8910.EMP** Starts a new FM/DB2 editor session, in Browse mode. The owner field is set to 'DSN8910' and the name field is set to 'EMP', on the Browse entry panel for the new editor session.
- **EDIT EMP** Starts a new FM/DB2 editor session, in Edit mode. The name field is set to 'EMP', on the Edit entry panel for the new editor session.

To exit the new FM/DB2 editor session and return to the current FM/DB2 editor session, press the End or CANCEL key.

**EDIT command**

The REDIT primary command displays information about tables related by a DB2 referential integrity (RI) constraint, to the DB2 object currently being edited. The command is context and cursor sensitive.

**Related references**

- “Listing related tables” on page 150
- “BROWSE primary command” on page 729
- “EDIT primary command” on page 740
- “VIEW primary command” on page 782
- “REDIT primary command” on page 770
- “Related tables panel” on page 639

Specifying the DB2 object you want to view

This section describes how you specify the DB2 object you want to view when you use the FM/DB2 View Utility (option 1), the FM/DB2 Edit Utility (option 2), or the FM/DB2 Browse Utility (option B) to view DB2 data or, in the case of edit, to optionally change DB2 data. The descriptions apply to the DB2 View panel, DB2 Edit panel, and DB2 Browse panel respectively.

To specify the table you want to view, use the Name entry field and, optionally, the Location, Owner, Database, and Table space entry fields.
Specifying the DB2 object you want to view

To specify the template that you want FM/DB2 to use when viewing the data, use a combination of the Template and Template usage entry fields. You can:

- Specify a template that describes the format of the DB2 object:
  - Use Data set name, and optionally Member, to specify the template.
  - Select 1 (Above).
- Use the last (previously used) template:
  - Select 2 (Previous).
  Any values in the Data set name and Member fields are ignored.
- Use FM/DB2 to generate a template using the information in the DB2 catalog:
  - Select 3 (Generate from table). This is the default setting.
  Any values in the Data set name and Member fields are ignored.
- Use FM/DB2 to generate a template using the information in the DB2 catalog and store the generated template using the name you specify (if the data set already exists, FM/DB2 replaces it):
  - Use Data set name, and optionally Member, to specify the name that you want FM/DB2 to use if storing the generated template.
  - Select 4 (Generate/Replace).

You can select this option to change the options for the current editor session:

**Edit options**

Enter one of these:

A “/” or an “A”
- Displays the first FM/DB2 editor options panel, prior to displaying the data for the DB2 object being processed.

A number in the range 1–8
- Displays the nth FM/DB2 editor options panel directly where n is the value entered.

Any options that you set last only for the duration of the current Edit session, and override any conflicting global options. To permanently change the Edit options, select option 0.2.

You can select the following options by entering a “/” or an “A”:

**Edit template**
- Select this option to edit the template before editing the data.

**Re-edit template**
- Select this option to control the navigation between panels within the Edit dialog. This option controls the behavior of the F3 and F12 keys when pressed from within an Edit session.

**Create audit trail**
- Select this option to generate an audit trail report of all successful modifications to DB2 data made during the current Edit session.

**Note:** This option is not displayed if SAF-rule controlled auditing is not in effect. See “SAF-rule controlled auditing” on page 47.

**Note:** Entering an “A” in an option field changes the behavior of the dialog so that the appropriate options panel is always displayed. If you only need to refer to an
Specifying the DB2 object you want to view

options panel on one occasion, use the “/” character. The “/” is cleared after the option is processed, thereby returning the dialog to the default behavior, which is to not show the various option panels.

Related tasks
"Selecting options on FM/DB2 panels” on page 24
"Setting options for the current FM/DB2 editor session” on page 95
"Editing a template” on page 57

Related references
"DB2 Browse panel” on page 501
"DB2 Edit panel” on page 505

Restricting the data that FM/DB2 loads from DB2

When you view a DB2 table or view in an FM/DB2 editor session, FM/DB2 accesses DB2 to obtain a copy of the data in the table or view. This copy of the data is loaded into the FM/DB2 editor. In normal circumstances, the data for every column of the DB2 object is retrieved - with the exception of any LOB or XML columns (see “Viewing and changing LOB data and XML documents” on page 97 for more information). This occurs regardless of whether you have deselected one or more columns using the template editor.

However, you may find that you want to restrict the data that FM/DB2 loads, either because you are only interested in data that satisfies certain selection criteria, or simply to limit the number of rows of data that FM/DB2 retrieves.

The following sections discuss the ways in which you can restrict the data that FM/DB2 loads.

Related tasks
"Storage considerations”
"Viewing large tables or views” on page 106
"Restricting the rows to view by specifying row selection criteria” on page 106
"Restricting the rows to view with the “Row count” entry field” on page 107
"Interactively developing row selection criteria” on page 106
"Using Select Statement Edit to limit the data retrieved by FM/DB2” on page 107
"Specifying the editor session mode: “normal mode” or “large mode”” on page 93
"Specifying the start position for your FM/DB2 editor session” on page 95

Storage considerations

If you are working with large DB2 tables or views, you may need to consider how much space is required in the editor to store the copy of the DB2 object.

The FM/DB2 editor uses storage in the TSO user’s address space. The size of the address space is governed by the TSO region size, which may be limited by your systems administrator. As a guide, the amount of space needed by the FM/DB2 editor is the number of rows loaded, multiplied by the number of bytes of data in each row, multiplied by an editor overhead.

For example, to load 100,000 rows of 20 bytes into the FM/DB2 editor, requires at least 2MB of storage. The editor overhead factor is at least 1 and can be as much as 3, so up to 6MB of storage may be needed.
Restricting the data that FM/DB2 loads from DB2

To take another example, if each row occupies 1000 bytes, every 1024 rows loaded requires about 1MB of storage. Therefore to load 100,000 such rows, at least 100MB of storage is required.

When the editor is operating in "normal mode", the number of rows loaded remains fixed and can be used to estimate the total storage requirements.

When the editor is operating in "large mode", the number of rows loaded in approximately three times the number of rows visible in TABL display format. This number remains unchanged regardless of any scrolling operations. When changes are made to rows, those rows remain in the editor. Therefore you may still encounter storage problems if you perform edit operations that change large numbers of rows in the table.

When you need to change large numbers of rows, consider changing the table by direct execution of SQL statements.

Viewing large tables or views

If the DB2 table or view requires more storage than is available in the TSO region, you will have to restrict the data that FM/DB2 loads from DB2. The most convenient way to do this is to select "large mode" by specifying ALL, $ or * in the Row count field on the entry panel. Large mode may not be available if the product installer has chosen to disable it.

When processing an object containing large object columns, FM/DB2 requires at least as much memory (virtual storage) as that needed to store one row of data from the object, with every large object column at the maximum possible size for that column. This may not be possible with a small TSO region size and large object data in the megabyte (or greater) range. To process an object with large object columns, create a DB2 view of the object without the large object columns present, and then process the view.

The following sections describe other ways in which you can restrict the data that FM/DB2 loads.

Related references

Restricting the rows to view by specifying row selection criteria

When you specify row selection criteria, FM/DB2 retrieves only those rows that match your criteria from DB2. Only these rows are loaded into the DB2 editor. Specifying row selection criteria is usually the most convenient and efficient method for restricting the data that FM/DB2 loads from DB2.

This method applies to both "normal mode" and "large mode".

Related tasks

Interactively developing row selection criteria

Sometimes you may need to view or edit a DB2 table or view with which you are not familiar. You may only need to view (and possibly change) a few rows of the
Restricting the data that FM/DB2 loads from DB2

table or view. FM/DB2 View and FM/DB2 Edit provide options to assist in interactively identifying the rows of interest, and then restricting the data loaded by FM/DB2 using row selection criteria.

To use this method, you need to alter the behavior of the editor dialog by setting the Edit template and Re-edit template options (use “A”). You can then alternate between the data display and the Column Selection/Edit panels, refining the row selection criteria until you have identified exactly the rows you require.

Related tasks
- “Selecting options on FM/DB2 panels” on page 24
- “Selecting rows” on page 58

Restricting the rows to view with the “Row count” entry field
If you just want to limit the volume of data that FM/DB2 retrieves from DB2, use the Row count entry field on the DB2 View panel or the DB2 Edit panel to specify the number of rows of DB2 data you want.

You can use this entry field to select “large mode” by typing ALL, 0, or * in the field.

If the maximum number of rows was retrieved, the message “Limit of Rows Retrieved” appears immediately below the last row (in TABL display format).

You can use this entry field when viewing or editing large tables, in combination with the Start position entry field, to show small portions of the table. You can then specify row selection criteria that limit rows retrieved to those of interest.

Related references
- “DB2 View panel” on page 518
- “DB2 Edit panel” on page 505

Restricting the rows to view with the “Start position” entry field
In some situations, the rows of interest may be located together in the table, but not necessarily at the beginning. In this case, if you know the approximate starting row number, you can specify a start position value in the Start position entry field. FM/DB2 skips rows until the specified start position and then loads rows from that point.

In "normal mode", the Row count value applies from the starting row, not the beginning of the table. Once the rows have been loaded you cannot scroll up to see the skipped rows.

In "large mode", FM/DB2 loads rows from the start position until the limits discussed in “Specifying the editor session mode: "normal mode" or "large mode"” on page 93 are reached. In "large mode", you can scroll up from the start position to the first row of the table if required.

Related references
- “DB2 View panel” on page 518
- “DB2 Edit panel” on page 505

Using Select Statement Edit to limit the data retrieved by FM/DB2
You can use the “Enter, execute and explain SQL statements” function (option 4.3) to enter a SELECT statement and display the results in an FM/DB2 editor session.
Restricting the data that FM/DB2 loads from DB2

In general, if you want to view or edit a DB2 table or view, View (option 1) and Edit (option 2) are recommended. However, you can specify the columns to retrieve using option 4.3.

For example, the following statement retrieves all columns from the DSN8610.EMP table:

```
SELECT * FROM DSN8610.EMP
```

However, the following statement retrieves only the listed columns:

```
SELECT EMPNO, FIRSTNAME, MIDINIT, LASTNAME, WORKDEPT, PHONENO, HIREDATE, JOB, EDLEVEL, SEX, BIRTHDATE FROM DSN8610.EMP
```

You might use this method when editing a table with many columns, most of which are of no interest to you. Alternatively, if the DB2 object has a “wide” row, omitting long VARCHAR or VARGRAPHIC columns may dramatically reduce the amount of data retrieved by FM/DB2 and loaded into the editor.

**Note:** When you omit columns from a DB2 object, you may find that not all edit operations are possible. For example, inserting a new row will fail if any of the omitted columns is defined as NOT NULL.

Related tasks
- “View and Edit options (options 1 and 2)” on page 43
- “Storage considerations” on page 105
- “Entering, executing, and explaining SQL statements” on page 351

Related references
- “DB2 View panel” on page 518
- “DB2 Edit panel” on page 505
- “Editor Options (1 of 8) panel” on page 528

Viewing data in an FM/DB2 editor session

When you view a DB2 table or view in an FM/DB2 editor session, you can:
- Locate a specific row or column
- Search the data for a specific string or numeric value
- Select a display format for the data
- Scroll or sort the data

Count of rows fetched in an FM/DB2 editor session

When FM/DB2 first displays data in an FM/DB2 editor session, it shows the number of rows that were fetched (“nn rows fetched”). The count of rows fetched is shown in the top left corner, on either the second line (if the command line is at the bottom of the display) or third line (if the command line is at the top of the display), and persists until you press Enter or any function key.

Locating a row

To move to a particular row number, use the LOCATE primary command. In an FM/DB2 editor session, rows are numbered from 1 upwards and the row number is shown in the prefix area (TABL display mode) or in the top right-hand part of the screen (SNGL display mode). For example, to move to row 42, enter the following command:

```
LOCATE 42
```
Viewing data in an FM/DB2 editor session

In TABL display format, this command scrolls the display (either forwards or backwards) so that row number 42 is displayed at the top of the screen. In SNGL display format, the display changes so that row number 42 is displayed.

In an FM/DB2 editor session, you can also use the LOCATE primary command to move to a row to which a particular label has been assigned. For example, to move to the row labeled “.HERE”, enter the following command:

LOCATE .HERE

In an FM/DB2 editor session, if in TABL display format, this command scrolls the display (either forwards or backwards) so that the row labeled “.HERE” is at the top of the screen. In SNGL display format, the display changes so that the row, previously labeled “.HERE” in TABL display format, is displayed.

Related tasks

- “TABL display format” on page 118
- “SNGL display format” on page 121
- “Assigning labels to rows” on page 116

Related references

- “LOCATE primary command” on page 761

Locating a column

You can use the LOCATE primary command to move to a particular column. You can specify either a DB2 column number, or part or all of a column name.

You specify a DB2 column number by preceding the column number with the “#” symbol (for example, #4). DB2 column numbers are shown on the Column Selection/Edit panel, and optionally in both TABL and SNGL display formats when you view data. (To display column numbers, you must select the Show column number option on the third Editor Options panel.) For long character columns only (> 50 bytes) you can optionally locate a character position within a column.

You specify a DB2 column name by typing all or part of the column name, with or without quotation marks.

When FM/DB2 finds the specified column, it scrolls the data so that the matching column is displayed on the left of the screen (in TABL display format), or at the top of the screen (in SNGL display format). In addition, after FM/DB2 has found the specified column, in SNGL display format if you press F11 (to display the following record) or F10 (to display the preceding record), the data is scrolled to show the matching column at the top of the screen for each record displayed. If you change to TABL display format, the data is scrolled so that the matching column is displayed on the left of the screen.

Related tasks

- “TABL display format” on page 118
- “SNGL display format” on page 121
- “View and Edit options (options 1 and 2)” on page 43
- “Setting options for the current FM/DB2 editor session” on page 95

Related references

- “LOCATE primary command” on page 761
- “Column Selection/Edit panel” on page 407
Specifying the column number or column name

To locate a column, you must specify one of the following:

- a DB2 column number
- a column name, or
- part of a column name

The following command locates column 4 in the data:

```
LOCATE #4
```

The following command locates column 4 in the data, and scrolls the display so that character position 100 of that column is visible:

```
LOCATE #4 100
```

Either of the following commands locates column EMPNO in the data:

```
L EMPNO
L 'EMPNO'
```

If you specify part of column name, the LOCATE command matches any column that contains the specified string within the column name. For example,

```
L NO
```

locates both EMPNO and DEPTNO.

**Using the FLD parameter:** The FLD parameter indicates to FM/DB2 that the string following the FLD keyword is the name or part of the name of the column you want to locate. You use the FLD parameter to resolve any ambiguity that might exist between column names and other command parameters.

You must specify the FLD parameter before the string, if the search string is “NEXT”, “PREV”, “FIRST”, “LAST”, “FLD”, starts with “#”, or is a number.

For example, to locate a column name containing the # symbol such as #ITEMS, use the following command:

```
L FLD #ITEMS
```

The command L #ITEMS results in an error as FM/DB2 is expecting the # symbol to be followed by a numeric DB2 column number.

**Specifying the search direction and starting point**

By default, the LOCATE primary command searches for the next occurrence of the specified column name, starting at the column currently displayed on the left of the screen (in TABL display format), or at the top of the screen (in SNGL display format).

You can control the direction and scope of the column name search by entering one of the following optional parameters:

- NEXT
- PREV
- FIRST
- LAST

The following sections describe how you use each of the preceding parameters.
Using the NEXT parameter: NEXT is the default value. Both of the following commands search for the next column name containing the string “NO”:

```
L NEXT 'NO'
L NO NEXT
```

In TABL display format, the preceding commands start at (but exclude) the column currently displayed on the left of the screen and scan right to find the first column name that is either “NO”, or contains the string “NO” (for example, NO, EMPNO, DEPTNO and NOMINAL all contain the string “NO”). The search ends when either a match occurs, or the last column of the DB2 object is encountered. If a matching column name is found, FM/DB2 scrolls the data so that the matching column is displayed on the left of the display.

In SNGL display format, the preceding commands start at (but exclude) the column currently displayed at the top of the screen and scan down to find the first matching column name. The search ends when either a match occurs, or the last column of the DB2 object is encountered. If a matching column name is found, FM/DB2 scrolls the data so that the matching column is displayed at the top of the display.

Using the PREV parameter: Both of the following commands search backwards for the next column name containing the string “NO”:

```
L PREV NO
L NO PREV
```

In TABL display format, the preceding commands start at (but exclude) the column currently displayed on the left of the screen and scan left to find the first column name that is either “NO”, or contains the string “NO”. The search ends when either a match occurs, or the first column of the DB2 object is encountered. If a matching column name is found, FM/DB2 scrolls the data so that the matching column is displayed on the left of the display.

In SNGL display format, the preceding commands start at (but exclude) the column currently displayed at the top of the screen and scan up to find the first matching column name. The search ends when either a match occurs, or the first column of the DB2 object is encountered. If a matching column name is found, FM/DB2 scrolls the data so that the matching column is displayed at the top of the display.

Using the FIRST parameter: Both of the following commands search for the first column name containing the string “NO”:

```
L FIRST NO
L 'NO' FIRST
```

In TABL display format, the preceding commands start at the first column for the DB2 object and scan right to find the first column name that is either “NO”, or contains the string “NO”. The search ends when either a match occurs, or the last column of the DB2 object is encountered. If a matching column name is found, FM/DB2 scrolls the data so that the matching column is displayed on the left of the display.

In SNGL display format, the preceding commands start at the first column for the DB2 object and scan down to find the first matching column name. The search ends when either a match occurs, or the last column of the DB2 object is encountered. If a matching column name is found, FM/DB2 scrolls the data so that the matching column is displayed at the top of the display.
Viewing data in an FM/DB2 editor session

Using the LAST parameter: Both of the following commands search for the last column name containing the string “NO”:

L LAST NO
L NO LAST

In TABL display format, the preceding commands start at the last column for the DB2 object and scan left to find the last column name that is either “NO”, or contains the string “NO”. The search ends when either a match occurs, or the first column of the DB2 object is encountered. If a matching column name is found, FM/DB2 scrolls the data so that the matching column is displayed on the left of the display.

In SNGL display format, the preceding commands start at the last column for the DB2 object and scan up to find the first matching column name. The search ends when either a match occurs, or the first column of the DB2 object is encountered. If a matching column name is found, FM/DB2 scrolls the data so that the matching column is displayed at the top of the display.

Finding a string

To search the data for a specific string, use the FIND primary command (which you can abbreviate as F or /).

For example, to search for the next occurrence of the string “Golden” in any mix of uppercase and lowercase, enter the following on the command line:

F GOLDEN

Repeating the search

To move the cursor to the next occurrence of the search string, issue the RFIND primary command or press the RFind function key (F5).

Note: RFIND works from the cursor position. When you use the RFind function key (F5), the cursor remains in the body of the data and pressing the key moves to the next occurrence of the string. However, if you type the RFIND primary command on the command line, you need to reposition the cursor on the last occurrence of string and then press Enter in order to find the next occurrence.

To find the same string as you specified on the previous FIND primary command, specify an asterisk (*) as the search string.

Handling long strings

For character columns (CHAR, VARCHAR), you can specify a search string up to 100 characters long. However, when you are dealing with a long search string, you may find that you cannot fit the whole FIND command (including other parameters) on the command line. To overcome the problem, do one of the following to display an Extended Command Entry pop-up panel:

- Enter FX (for Find Extended)
- Enter the FIND command (or one of its abbreviations, such as F or /) with no parameters

Note: If you have previously used the FIND or CHANGE commands in the current FM/DB2 editor session, to display the Extended Command Entry pop-up panel in this way, you will need to first use the RESET command before entering the FIND command (or one of its abbreviations) without any parameters. Alternatively, enter FX.
 Viewing data in an FM/DB2 editor session

The Extended Command Entry pop-up panel contains five lines (each 50 characters long) that you can use to enter all the FIND command parameters (but not the actual keyword FIND or any of its abbreviations). FM/DB2 treats the five lines on the pop-up panel as contiguous, allowing you to specify a long search string, the DB2 column number and optionally any other parameters, using up to 250 characters in total.

**Note:** You can continue the search string (or any other parameter) from the last position on one line of the pop-up panel to the first position of the next line. However, this does not mean that you need to completely fill a line before using the next line. For example, you can specify the search string on the first of the five lines, and the DB2 column number (specified as \#n) on the next line. If you do not use all of a line, FM/DB2 treats the unused part of the line as a single space. Figure 38 shows the Extended Command Entry pop-up panel for the FIND command. In this example, FM/DB2 finds any previous occurrences of a long string (78 characters) in DB2 column 3.

**Related references**

[FIND primary command](#) on page 748

**Finding data within specific columns**

You can optionally limit the effects of primary commands to a specified column, a list of columns, or one or more ranges of columns.

You specify the columns to be searched using \#n column references. These are displayed above each column in TABL display format, or next to each column in SNGL display format. The search string is found only if it is completely contained within one of the specified columns.

**Specifying a list of columns**

When you specify a list of columns, you must either:
Viewing data in an FM/DB2 editor session

- Enclose the column references (optionally separated by commas) in parentheses, or
- Separate the column references with commas but without any intervening spaces.

These are valid lists of columns:

#17
(#17)
(#17 #22)
(#17, #22)
#17,#22

These are invalid lists of columns:

#17 #22
#17, #22

The search string is found if it is completely contained within one of the specified columns. For example, the following command excludes all records containing the characters “the” (in any mix of uppercase and lowercase) in column #5:

EXCLUDE ALL the #5

The following command finds the next record containing an uppercase letter A in column #8 or #9:

FIND C'A' #8,#9

Specifying ranges of columns

You specify a range of columns by nominating the first and last column reference of the range, separated by a hyphen, with or without intervening spaces. You can optionally enclose the range in parentheses.

These are valid ranges of columns:

#17-#19
#17 - #19
(#17-#19)
(#17 - #19)

To specify more than one range of columns:
- Enclose each column range in parentheses (optionally separated by commas), or
- Separate each column range with commas, but without any intervening spaces.

These are valid ranges of columns:

(#8-#11) (#17-#19) (#24-#25)
(#8-#11),(#17-#19),(#24-#25)
#8-#11,#17-#19,#24-#25

These are invalid ranges of columns:

#8-#11 #17-#19 #24-#25
#8-#11, #17-#19, #24-#25

Note:
1. Column ranges can overlap. For example, #2-#7, #5-#9. In this case, the result is a consolidated single column range equivalent to #2-#9.
Finding a string within specific portions of the data

When you use the FIND command to search for a string, you can limit the search to specific portions of the data you are searching:

- To limit the search for a string to only where it appears as a prefix in the data, specify the PREFIX parameter.
- To limit the search for a string to only where it appears as a suffix in the data, specify the SUFFIX parameter.
- To limit the search for a string to only where it appears as a word in the data, specify the WORD parameter.

Finding data within excluded or non-excluded rows

If your data contains excluded and non-excluded rows, you can limit the search for a string or numeric value to:

- Only non-excluded rows (parameter NX), or
- Only excluded rows (parameter EX or X)

For example, the following command finds the next non-excluded row containing the string “SCARECROW” in column number 1 or column number 7:

```
FIND SCARECROW (#1 #7) NX
```

And the following command finds the next excluded row containing the value 1024 in column number 3:

```
FIND 1024 #3 EX
```

Finding data within a label range

In an FM/DB2 editor session, you can limit the search for a string or numeric value to a range of rows, identified by a pair of labels indicating the first and last rows to be searched.

For example, the following command finds the next row containing the string “TIME” in column number 10 in the range of rows from the label .STRT to the label .FNSH:

```
FIND TIME #10 .STRT .FNSH
```

And the following command finds the next row containing the value 2840 in column number 4 or column number 6 in the range of rows from the current cursor position to the label .END:

```
FIND 2840 #4,#6 .ZCSR .END
```

Related tasks

- “Assigning labels to rows” on page 116
Assigning labels to rows

In an FM/DB2 editor session, if you use the CHANGE, DELETE, EXCLUDE, EXPORT, or FIND command in TABL display format, you can limit the rows that are searched by entering a "range" parameter. The range parameter consists of a pair of labels indicating the first and last rows to be searched. The string is found if it is contained within the specified range.

A label can be one to four alphabetic characters, beginning with a period (.). Labels starting with the letter “Z” are assigned by the editor.

You assign a label by typing the valid character string into the prefix area of the row. In the following example, the label .HERE is assigned to row number 16 and the label .THRE is assigned to row number 129:

```
00015 This is a row.
.HERE This is a row with a label.
00017 This is a row.
...
00128 This is a row.
.THRE This is a row with a label.
00130 This is a row.
...
```

After it is assigned, a label stays with the row, even if the row is moved.

To remove a label from a data row, blank out the label characters (or overtype them with a new label) then press Enter. You can unassign a label by deleting the row containing that label, or you can remove all row labels by using the RESET LABEL command.

You can overtype a label with a prefix command without unassigning the label. The label reappears at the completion of the command and remains displayed in the row prefix area.

You can move a label to another row by typing the same label on a new row.

Some labels are automatically assigned and maintained by the editor. These special labels begin with the letter “Z” (Your own labels therefore must not start with a “Z”.) Unlike other labels, these editor-assigned labels do not necessarily stay with the same row, but instead represent logical positions on the display.

Some of these editor-assigned labels are available to you. The editor assigns:

- `.ZCSR` To the row on which the cursor is currently positioned.
- `.ZFST` To the first row (relative row number 1). This can be abbreviated to `.ZF`.
- `.ZLST` To the last row. This can be abbreviated to `.ZL`.

The following FIND command finds the next row, excluded or non-excluded, containing the string `xxx`, in the column with column number 5, in the range of rows from the current cursor position to the end of data:
FIND xxx #5 .ZCSR .ZLST

The following EXCLUDE command excludes all rows, excluded or non-excluded, containing the string “ABC”, in either of the columns with column numbers 3 and 7, in the range of rows from the label .HERE to the label .THRE:

EXCLUDE ALL ABC (#3 #7) .HERE .THRE

If you use the range parameter, the two labels must be separated by a blank or comma. A single label is not allowed. The labels can be any combination of the editor-assigned labels (.ZCSR, .ZFST, .ZLST) or labels you have assigned yourself.

The label representing the smaller relative row number is used as the start of the range and the label representing the larger relative row number is used as the end of the range, regardless of the order in which they are specified.

If the range you specify includes excluded rows that are hidden from display or represented by shadow lines, those rows are not affected by the command.

Related references

“CHANGE primary command” on page 732
“DELETE primary command” on page 737
“EXCLUDE primary command” on page 741
“FIND primary command” on page 748
“RESET primary command” on page 771

Searching numeric columns

The numeric value of the search string must equal the formatted numeric value of the column. The lengths (number of digits) of the search string and of the numeric column are not significant. Similarly, the data type of the numeric column is not significant. For example, a search string of 123 matches a DECIMAL column containing 00123, or a SMALLINT column containing 123. A search string of 123 does not match a DECIMAL or SMALLINT column containing 12300 (because the numeric values are different). An unsigned value in the string is considered to be positive.

Related references

“Finding data within specific columns” on page 113

Changing the view of displayed data

You can change the view of data displayed in an FM/DB2 editor session by editing the template to:

- Show only selected columns
- Change the sequence of displayed columns
- Change column headings
- Change the width of columns
- Show leading zeros (numeric columns)

After you have edited the template, you can either save it for future use, or just use it for the current FM/DB2 editor session only.
Viewing data in an FM/DB2 editor session

To edit the template in the current FM/DB2 editor session, use the TEDIT primary command.

To edit the template from the Table View or Table Edit panel, select the Edit template option.

Related tasks
- Chapter 3, “Working with templates,” on page 49

Related references
- TEDIT primary command” on page 779

Selecting a display format
In an FM/DB2 editor session, you can view the data in the following display formats:
- Tabular format (TABL), or
- Single-row format (SNGL)

To select the display format you want, you can:
- Enter one of the following values (or just its first letter) in the Format field (in the upper right corner of the Table View or Table Edit panel):
  - TABL
  - SNGL
- Use the FORMAT primary command to toggle between TABL and SNGL display formats.
- In TABL display format, use the FS primary command to change to SNGL display format.
- In SNGL display format, use the FT primary command to change to TABL display format.
- If in TABL display format, use the FS prefix command to change to SNGL display format.

The following pages describe the TABL and SNGL display formats.

Related references
- “FORMAT primary command” on page 753
- “FS primary command” on page 754
- “FT primary command” on page 755
- “Table Edit panel” on page 691

TABL display format
In TABL display format (the default format), the rows of the table are presented formatted into columns displayed horizontally across the panel.

Figure 39 on page 119 shows an example of a view display in TABL display format.
Viewing data in an FM/DB2 editor session

Figure 39. View panel in TABL display format (with all optional lines displayed)

Figure 40 shows an example of an edit display in TABL display format.

Multiple rows are displayed on the one panel (limited by the screen size). Each column of the table is formatted according to the data type for that column. For example, numeric data types are displayed as whole or decimal numbers, and date and time fields are displayed as dates or times.

By default, the headings are the names of the columns as defined in the DB2 catalog. You can change them by editing the template.

For SELECT statement View or Edit only, the heading might be displayed as COLnnn where nnn is 001, 002, and so on. This indicates a generated column name. Both of the following examples cause FM/DB2 to generate a column name:
### Viewing data in an FM/DB2 editor session

```
SELECT 1, dbname FROM SYSIBM.SYSTABLES
SELECT CONCAT(dbname, tsname) FROM SYSIBM.SYSTABLES
```

One or more columns in the FM/DB2 editor session might be protected. This can occur if:

- The object being edited has a primary key, and the **Allow updates to primary key** option is not selected.
- The object being edited is a read-only view.
- The object being edited is the result table from an arbitrary SELECT statement (that is, select statement edit).
- The object being edited is a table, but contains columns, excluding ROWID columns, marked as not updateable in the DB2 catalog. The DB2 catalog tables fall into this category.

If you display data in TABL display format, you can scroll up, down, left, and right.

To display the data in hexadecimal format, use the HEX primary command:
```
HEX ON
```

In TABL display format, the view or edit display always shows the column names for each displayed column (or the alternative field heading if you have defined one).

Optionally, you can display up to 3 additional information lines between the column heading line and the top of data. These are (in order):

- A line containing the column number (as defined in the DB2 catalog).
- A line containing the data type for the column (and the length for non-numeric fields).
- A line showing a ruler above each column. The ruler can include offset information and, optionally, information about primary and foreign keys, and unique indexes.

You can select which, if any, of these additional information lines are displayed on the editor display either:
- For all FM/DB2 editor sessions by setting the appropriate global option using the Editor Options panel, or
- For the current FM/DB2 editor session only:
  - For View, you set the appropriate option using the View (option 1) panel.
  - For Edit, you set the appropriate option using the Edit (option 2) panel.

**Related tasks**
- Chapter 3, “Working with templates,” on page 49
- “Scrolling” on page 125
- “View and Edit options (options 1 and 2)” on page 43
- “ROWID columns” on page 76
- “Setting options for the current FM/DB2 editor session” on page 95
- “Excluding rows” on page 126

**Related references**
- “DB2 View panel” on page 518
- “DB2 Edit panel” on page 505
- “Editor Options (1 of 8) panel” on page 528
Viewing data in an FM/DB2 editor session

**SNGL display format**
In SNGL display format, only one row of the table is displayed on the panel at the one time.

Figure 41 shows an example of an edit display in SNGL display format.

The columns of the table are displayed vertically on the panel, with the column header on the left of the panel, and the data for that column on the right. Each column of the table is formatted according to the data type for that column. For example, numeric data types are displayed as whole or decimal numbers, and date and time fields are displayed as dates or times.

By default, the headings are the names of the columns as defined in the DB2 catalog. You can change them by editing the template.

For SELECT statement in an FM/DB2 editor session, the heading might be displayed as COLUMN where nnn is 001, 002, and so on. This indicates a generated column name. Both of the following examples cause FM/DB2 to generate a column name:

```
SELECT 1, dbname FROM SYSIBM.SYSTABLES
SELECT CONCAT(dbname, tsname) FROM SYSIBM.SYSTABLES
```

One or more columns in the FM/DB2 editor session might be protected. This can occur if:
- The object being edited has a primary key, and the “Allow updates to primary key” option is not selected.
- The object being edited is a read-only view.
- The object being edited is the result table from an arbitrary SELECT statement (that is, select statement edit).
Viewing data in an FM/DB2 editor session

- The object being edited is a table, but contains columns, excluding ROWID columns, marked as not updateable in the DB2 catalog. The DB2 catalog tables fall into this category.

If you display data in SNGL display format, you can:
- Scroll within the row (up and down)
- Scroll forward and backward within the table (to show the next and previous non-excluded rows)

To display the data in hexadecimal format, use the HEX primary command HEX ON.

In SNGL display format, the view or edit display always shows two columns: the name of the column, and the data for that column.

Optionally, you can display up to three additional information columns. Two of these columns, Ref and Key, are displayed to the left of the column containing the column name; the third, Type(Len) is displayed between the name and data columns. If you choose to show all information columns, the displayed columns are (in order of appearance from left to right):

- **Ref**  An optional column showing the column number (as defined in the DB2 catalog).
  You can turn this optional column on and off with the REFS primary command. The value shown can be zero if displaying the result table from an arbitrary select statement (that is, select statement edit). This indicates either that the column is generated, or that the column is not a column of the first table named in the FROM clause of the SELECT statement that produced the result table.

- **Rdf**  Unused.

- **Key**  An optional column showing information about primary keys, foreign keys, unique indexes and check constraints.
  You can turn this optional column on and off with the INDEXINF primary command.

- **Column**  A column showing the column name.
  This column may also show the start and end position of the character data displayed for a row. For this to occur:
  1. The length of the data for the column must be long enough to require two or more rows to display.
  2. The “Show char position range” editor option needs to be selected, or the CHARPOS command can be issued to turn on the character position range display. For more information, see “Editor Options (3 of 8) panel” on page 535 and CHARPOS command “CHARPOS primary command” on page 736.

- **Type(Len)**  An optional column showing the data type for the column (and the length for non-numeric fields).
  You can turn this optional column on and off with the TYPE primary command.

- **Data**  A column showing the (formatted) data for the column.
Viewing data in an FM/DB2 editor session

You can customize the view or edit display either:

- For all FM/DB2 editor sessions by setting the appropriate global option using the Editor Options panels, or
- For the current FM/DB2 editor session only:
  - For View, you set the appropriate option using the DB2 View panel.
  - For Edit, you set the appropriate option using the DB2 Edit panel.

Related tasks
Chapter 3, “Working with templates,” on page 49
“Scrolling” on page 125

Related references
“DB2 View panel” on page 518
“DB2 Edit panel” on page 505
“Editor Options (1 of 8) panel” on page 528
“HEX primary command” on page 755
“INDEXINF primary command” on page 756
“REFS primary command” on page 771
“TYPE primary command” on page 780

Zooming in to see all of a row
You can “zoom in” on a single row by pressing the Zoom function key (F2) or by using the ZOOM primary command.

In TABL display format, to display all data in a particular row without having to scroll left or right, move the cursor to the row, then press the Zoom function key (F2). FM/DB2 limits the display to just that row. In SNGL display format, just press the Zoom function key (F2).

If you zoom in on a row, only that row is displayed in “zoomed” SNGL display format. The zoomed display automatically scrolls to the field that was at the left of the panel, or, if applicable, the column where the cursor was positioned before zooming.

In “zoomed” SNGL display format, the word Zoom is displayed next to Format in the upper right corner of the panel as shown in Figure 42 on page 124.
Viewing data in an FM/DB2 editor session

If you view a record in “zoomed” SNGL display format, FM/DB2 displays all columns, regardless of whether they were selected for display in the template. FM/DB2 displays the columns in the order defined in the template. Any template re-sequencing is ignored. In this display format, you cannot navigate between rows or use the TEDIT primary command.

In TABL or normal “un-zoomed” SNGL display format, File Manager only shows the columns selected for display in the template.

For instance, in the example shown in Figure 42, the columns WORKDEPT and BIRTHDATE are shown in “zoomed” SNGL display format, even though these columns happen to be “deselected” in the template. In “un-zoomed” SNGL display format, the same row is displayed:

Figure 42. Zooming in on one row (“zoomed” SNGL display format)

If you view a record in “zoomed” SNGL display format, FM/DB2 displays all columns, regardless of whether they were selected for display in the template. FM/DB2 displays the columns in the order defined in the template. Any template re-sequencing is ignored. In this display format, you cannot navigate between rows or use the TEDIT primary command.

In TABL or normal “un-zoomed” SNGL display format, File Manager only shows the columns selected for display in the template.

For instance, in the example shown in Figure 42, the columns WORKDEPT and BIRTHDATE are shown in “zoomed” SNGL display format, even though these columns happen to be “deselected” in the template. In “un-zoomed” SNGL display format, the same row is displayed:
Viewing data in an FM/DB2 editor session

In “un-zoomed” SNGL display format, you can navigate between rows using the PREVIOUS and NEXT primary commands or the Previous function key (F10) and the Next function key (F11).

To return to the display format that was current before zooming in, enter the ZOOM primary command again, or the CANCEL, END, or EXIT command.

Related tasks

“TABL display format” on page 118
“SNGL display format” on page 121

Related references

“ZOOM primary command” on page 785
“TEDIT primary command” on page 779

Scrolling

To scroll through your data, press the following function keys:

Press this function key
   To...
Up function key (F7)
   Scroll backward (up)
Down function key (F8)
   Scroll forward (down)

In TABL display format:

Right function key (F11)
   Scroll right
Left function key (F10)
   Scroll left

In SNGL display format:

Next function key (F11)
   Go to the next row
Previous function key (F10)
   Go to the previous row

You can also scroll using the primary commands UP, DOWN, LEFT, RIGHT, TOP, BOTTOM, NEXT and PREVIOUS.

Related tasks

“TABL display format” on page 118
“SNGL display format” on page 121
“Holding columns (when scrolling left or right)” on page 69

Related references

“ZOOM primary command” on page 785
“TEDIT primary command” on page 779
“UP primary command” on page 780
“DOWN primary command” on page 738
“LEFT primary command” on page 757
“RIGHT primary command” on page 773
“TOP primary command” on page 779
“BOTTOM primary command” on page 729
“NEXT primary command” on page 765
“PREVIOUS primary command” on page 767
**Viewing data in an FM/DB2 editor session**

**Controlling how far you scroll:** To control how far you scroll when you press one of the scrolling function keys, enter a scroll amount in the Scroll field:

- **Scroll amount**
  - **PAGE** One panel at a time
  - **HALF** Half a panel at a time
  - **DATA** One line or panel column less than a screen at a time
  - **CSR** To the cursor position (if the cursor is not positioned on a row when the scroll is performed, the default scroll amount is PAGE)
  - **nnnn** A number of panel columns (if scrolling left or right) or rows (when scrolling up or down) at a time

**Note:**
1. In TABL display format, you can temporarily override the amount in the Scroll field by typing a scroll amount on the command line, then pressing a scroll function key. For example, if you enter 8 on the command line, then press the Right function key (F11), FM/DB2 scrolls right 8 panel columns.
2. In SNGL display format, you can specify the number of rows FM/DB2 scrolls forward or backward by typing a number on the command line, then pressing the Next function key (F11) or the Previous function key (F10). For example, if you enter 5 on the command line, then press the Next function key (F11), FM/DB2 scrolls forward 5 rows.
3. In TABL display format, the position that the cursor occupies after you scroll left or right depends on the data type of the DB2 column that occupies the resultant panel column:
   - If the data type of the DB2 column is non-numeric, scrolling left or right moves to the relative panel column within the DB2 column (for small columns, scrolling moves to the start of the column).
   - If the data type of the DB2 column is numeric, scrolling left or right moves to the start of the DB2 column.

**Scrolling to the first or last row or column:** To scroll to the first or last row or column in the table, type **MAX** (or **M**) on the command line, then press one of the scroll function keys. For example, typing **M** then pressing the Right function key (F11) scrolls right to the last column.

You can also scroll to the first or last row by entering **TOP** or **BOTTOM** on the command line.

**Excluding rows**

When you are viewing or editing data, you can “exclude” certain rows in the data you are viewing. These rows are referred to as excluded rows and are not displayed on your screen. The remaining rows (those that are displayed on your screen) are referred to as non-excluded rows.

You may choose to exclude certain rows to restrict the scope of the CHANGE, DELETE and FIND primary commands. When you use one of these commands with data that contains excluded rows and non-excluded rows, you can specify whether the command is to affect:
- Only non-excluded rows,
- Only excluded rows, or
- All rows
To exclude certain rows in the table you are viewing, you can use:

- The EXCLUDE primary command. For example, the following command excludes all rows with the string “SMITH” in column number 3:

```
EXCLUDE ALL SMITH #3
```

The following command excludes the next row containing the string “XXX” within either column number 20 or column number 30:

```
EXCLUDE XXX #20 #30
```

You can exclude a range of rows identified by a “from” label and a “to” label. For example, the following command excludes a range of rows, starting with the row labeled “.LABA” and ending with the row labeled “.LABB”:

```
EXCLUDE ALL .LABA .LABB
```

In an FM/DB2 editor session, you can specify an asterisk (*) as the search string to find the same string as you specified on the previous EXCLUDE primary command.

In an FM/DB2 editor session, you can limit the effect of the EXCLUDE command to specific portions of the data you are searching:

- To limit the search for a string to only where it appears as a prefix in the data, specify the PREFIX parameter.
- To limit the search for a string to only where it appears as a suffix in the data, specify the SUFFIX parameter.
- To limit the search for a string to only where it appears as a word in the data, specify the WORD parameter.

- The following prefix commands:

```
X Exclude one row.
Xn Exclude n rows.
XX Exclude a block of rows.
```

You can use the following line commands to show various rows from a block of excluded rows:

```
F Show the first row from a block of excluded rows.
Fn Show the first n rows from a block of excluded rows.
L Show the last row from a block of excluded rows.
Ln Show the last n rows from a block of excluded rows.
```

Related tasks

- [Showing or hiding excluded rows](#)

Related references

- ["EXCLUDE primary command" on page 741](#)
- ["CHANGE primary command" on page 732](#)
- ["DELETE primary command" on page 737](#)
- ["FIND primary command" on page 748](#)

**Showing or hiding excluded rows:** In TABL display format, you can choose to hide excluded rows from display, or represent them by shadow lines:

```
- - - - - - - - - - - - - - - - - - - - - - - - - - - n line(s) excluded
```

The SHADOW primary command controls whether shadow lines are shown or hidden. To turn on shadow lines for excluded rows, enter SHADOW ON. To turn off shadow lines for excluded rows, enter SHADOW OFF.

In SNGL display format, shadow lines are not displayed. If you use the Next function key (F11) or Previous function key (F10) to scroll through rows in SNGL
display format, the only indication that an excluded row has been skipped is that
the row number (shown in the upper right corner of the panel) skips accordingly.

To redisplay excluded rows, enter the following command:
RESET EXCLUDED

Related tasks
- "TABL display format" on page 118
- "SNGL display format" on page 121

Related references
- "SHADOW primary command" on page 775
- "RESET primary command" on page 771

Sorting the data
To sort the data you are viewing or editing in TABL display format, use the SORT
primary command.

You can specify up to 5 column numbers (the most significant first) and whether
you want the data sorted in ascending (the default sequence) or descending
sequence for each column number you specify.

Examples

**SORT #7 D #6 #3**

Sorts all rows on the column with column number #7 (primary sort key) in
descending order, then on the column with column number #6 (secondary
sort key) in ascending order, then on the column with column number #3
(minor sort key) in ascending order,

**SORT #6 X**

Sorts all excluded rows on the column with column number #6 in
ascending order (as if non-excluded rows did not exist).

Note: The SORT command only changes the order in which you view the data; it
has no effect on the order in which the data is stored within DB2.

Related references
- "SORT primary command" on page 777

Displaying data in hexadecimal format
To display the data you are browsing or editing in hexadecimal format, use the
HEX primary command.

For example, if your display looks something like this:

<table>
<thead>
<tr>
<th>NAME</th>
<th>CREATOR</th>
<th>TYPE</th>
<th>DBNAME</th>
<th>TSNAME</th>
<th>DBID</th>
<th>OBID</th>
</tr>
</thead>
<tbody>
<tr>
<td>#1</td>
<td>#2</td>
<td>#3</td>
<td>#4</td>
<td>#5</td>
<td>#6</td>
<td>#7</td>
</tr>
<tr>
<td>VARCHAR(18)</td>
<td>CH(8)</td>
<td>CH(1)</td>
<td>CH(8)</td>
<td>CH(8)</td>
<td>SMINT</td>
<td>SMINT</td>
</tr>
</tbody>
</table>

0000000 **** Top of data ****
000001 SYSCOPY  SYSIIBM  T  DSNDB06  SYSCOPY  00006  00046
000002 SYSFIELDS SYSIIBM  T  DSNDB06  SYSDBASE  00006  00021

then issuing the command HEX ON causes the hexadecimal representation to be
displayed:

<table>
<thead>
<tr>
<th>NAME</th>
<th>CREATOR</th>
<th>TYPE</th>
<th>DBNAME</th>
<th>TSNAME</th>
<th>DBID</th>
<th>OBID</th>
</tr>
</thead>
<tbody>
<tr>
<td>#1</td>
<td>#2</td>
<td>#3</td>
<td>#4</td>
<td>#5</td>
<td>#6</td>
<td>#7</td>
</tr>
<tr>
<td>VARCHAR(18)</td>
<td>CH(8)</td>
<td>CH(1)</td>
<td>CH(8)</td>
<td>CH(8)</td>
<td>SMINT</td>
<td>SMINT</td>
</tr>
</tbody>
</table>
Viewing data in an FM/DB2 editor session

Managing the prefix area
The prefix area displays line numbers that represent the row number in the DB2 table or view being edited.

The prefix area also doubles as the prefix command entry area. You enter prefix commands by overtyping the row number for the appropriate row.

You can use prefix commands to:
- Insert or delete rows
- Repeat (duplicate) rows
- Move rows
- Shift data
- Limit the data being edited

In an FM/DB2 editor session, to see which prefix commands are valid for the current environment, enter an "?" in the prefix area (against any listed line). FM/DB2 displays a list of the line commands you can enter, like those shown in Figure 43.

Process Options Utilities Help
FM/DB2 Available Edit Prefix Commands
More: +

Prefix commands
You can enter the following commands in the prefix area of the data display:
- A identifies the record after which data is to be moved or copied.
- B identifies the record before which data is to be moved or copied.
- C copy one record.
- Cn copy n records.
- CC copy block of records. Mark start and end of block.
- D delete one record.
- Dn delete n records.
- DD delete block of records. Mark start and end of block.
- E displays information about why DB2 rejected the change to a row
- F display the first record of a block of excluded records.

F1=Help  F2=Split  F3=Exit  F4=Retrieve  F7=Backward  F8=Forward
F9=Swap  F10=Actions  F12=Cancel

Figure 43. Using the ? prefix command to display a list of valid prefix commands

In an FM/DB2 editor session, you can choose whether a prefix area is displayed next to the data or not.
To specify the default position and display state of the prefix area for all FM/DB2 editor sessions, set these editor options on the relevant Editor Options panel (0.2):

- Display prefix area
- Display prefix area on the right
- Prefix area width (A,6,7,8,9)

To specify the default position and display state of the prefix area for the current FM/DB2 editor session, use the PREFIX primary command. For example:

- To display the prefix area on the right, enter PREFIX RIGHT.
- To display the prefix area as an 8-digit field, enter PREFIX 8.
- To display the prefix area with a width that automatically adjusts (from 6 to 9 digits wide) in order to display the record number, enter PREFIX A.
- To remove the display of the prefix area, enter PREFIX OFF.

Related references
  "PREFIX primary command" on page 767

Displaying the column number
In SNGL display format, use the REFS primary command to show or hide the column number for each displayed column:

- If the column numbers are not currently displayed, FM/DB2 displays an additional column (to the left of the column containing the column name and, if displayed, to the left of the column showing index information) showing the column number for each column.

If you enter the REFS primary command in TABL display format, the display is unaltered. However, if you change to SNGL display format, the display status of the additional column is reversed.

- If the column numbers are already displayed, the REFS primary command removes them from display.

Note: You can also display the column numbers by selecting the Show column number option (for TABL, SNGL, or both display formats).

Related references
  "REFS primary command" on page 771
  "Editor Options (3 of 8) panel" on page 535

Displaying the data type
In SNGL display format, use the TYPE primary command to show or hide the data type (and, for non-numeric columns, the length of the column) for each displayed column:

- If the data type details are not currently displayed, FM/DB2 displays an additional column (between the name and data columns) showing the data type details for each column.

If you enter the TYPE primary command in TABL display format, the display is unaltered. However, if you change to SNGL display format, the display status of the additional column is reversed.

- If the data type details are already displayed, the TYPE primary command removes them from display.

Note: You can also display the data type details by selecting the Show data type option (for TABL, SNGL, or both display formats).

Related references
  "REFS primary command" on page 771
Handling special data

The following sections describe how to work with special types of data.

**DATE and TIME columns**

DB2 accepts dates and times in a number of formats.

**DATE columns**

In an FM/DB2 editor session, the display format for DATE columns depends on the setting of the DATE parameter in the DB2 DSNHDECP installation module. For further information, see your DB2 systems administrator, the *DB2 UDB for z/OS SQL Reference*, or other DB2 manuals.

If you edit a DATE column, you can enter a date in any format recognized by DB2. FM/DB2 converts the date you enter into the appropriate display format. For example, to set the contents of a DATE column to represent September 27, 2004, you can enter `27.09.2004`.

However, you need to be careful when choosing the separator character. If you try to enter `27/09/2004`, FM/DB2 interprets “27” as a month and rejects the date as incorrect. A more subtle example is `08/05/1900` and `08.05.1900`. FM/DB2 interprets the first as 5 August, 1900, but the second as May 8, 1900.

**TIME columns**

In an FM/DB2 editor session, the display format for TIME columns depends on the setting of the *External Format for TIME Data Type Columns* System Option.

If you edit a TIME column, you can enter a time in any format recognized by DB2. FM/DB2 converts the time you enter into the appropriate display format.

If you enter times in USA format (`hh:mm AM` or `PM`), the absence of second information means that the time is stored internally as `hh:mm.00` (that is, the seconds portion is always set to zero). This can cause unexpected results when comparing different TIME columns that have the same display format but different internal representations with respect to seconds.

Related tasks

“FM/DB2 system options (option 0.2)” on page 41

Related references

“FM/DB2 Systems Options (1 of 4) panel” on page 577

**Large numeric fields**

If FM/DB2 generates a template, it sets the column width for numeric columns such that any value can be correctly displayed.

However, if as a result of editing the template, a numeric column contains a valid numeric value, but the value is too large to fit in the width allocated to the column (as defined in the template), the value is truncated. The column is highlighted to indicate that truncation has occurred.

To display the entire column without truncation, you need to edit the template, and specify an output width for the column:
Handling special data

1. Enter TEDIT on the command line to display the Column Selection/Edit panel.
2. Enter E template editor prefix command in the Prefix Command area to display the Column Attributes panel.
3. In the Output width field, enter a value that is wide enough to display the entire column.
4. Press the Exit function key (F3) key until you are back at the editor panel.

Related references
- “Column Selection/Edit panel” on page 407
- “EXCLUDE primary command” on page 741
- “FIND primary command” on page 748
- “TEDIT primary command” on page 779

Working with large tables

When operating in view mode, FM/DB2 allows you to process large DB2 tables without the need to specify a large TSO region size. This feature is not available if the product installer has chosen to disable it.

A number of options and processes associated with working with large tables are available and described in these sections:

- “SQL optimization options”
- “Large table options”
- “Data sampling” on page 133

SQL optimization options

You set the Optimization options on the Editor Options (2 of 8) panel to specify a OPTIMIZE FOR nnn ROWS clause in the SQL statement used to retrieve data from DB2.

Table 6 describes the behavior for various combinations of options:

<table>
<thead>
<tr>
<th>Use row count value</th>
<th>Row count</th>
<th>Optimize for ... Rows</th>
<th>Behavior</th>
</tr>
</thead>
<tbody>
<tr>
<td>Not selected</td>
<td>0</td>
<td>0</td>
<td>No clause</td>
</tr>
<tr>
<td>Not selected</td>
<td>0</td>
<td>nnn</td>
<td>OPTIMIZE FOR nnn ROWS appended</td>
</tr>
<tr>
<td>Not selected</td>
<td>nnn</td>
<td>0</td>
<td>No clause</td>
</tr>
<tr>
<td>Not selected</td>
<td>nnn</td>
<td>nnn</td>
<td>OPTIMIZE FOR nnn ROWS appended</td>
</tr>
<tr>
<td>Selected</td>
<td>0</td>
<td>0</td>
<td>No clause</td>
</tr>
<tr>
<td>Selected</td>
<td>0</td>
<td>nnn</td>
<td>OPTIMIZE FOR nnn ROWS appended</td>
</tr>
<tr>
<td>Selected</td>
<td>nnn</td>
<td>0</td>
<td>OPTIMIZE FOR nnn ROWS appended</td>
</tr>
<tr>
<td>Selected</td>
<td>nnn</td>
<td>nnn</td>
<td>OPTIMIZE FOR nnn ROWS appended</td>
</tr>
</tbody>
</table>

Related references
- “Editor Options (2 of 8) panel” on page 531

Large table options

You can set the Scrollable cursor type option on the Editor Options (2 of 8) panel to either 1 (Insensitive) or 2 (Sensitive static). When processing a large DB2 object in place holder mode only, FM/DB2 uses the appropriate type of scrollable cursor.
There is no simple method for you to determine which type of scrollable cursor is in use. A DBG trace shows the SQL statement used to declare the cursor and fetch the data.

Related references

”Editor Options (2 of 8) panel” on page 531

**Data sampling**

Data sampling allows you specify that a subset of the rows in a DB2 object are loaded into the editor. This can be particularly useful when you are working with very large DB2 tables and require a representative sample of the data.

The **Sampling type** option on the Editor Options (2 of 8) panel allows you to specify how you want FM/DB2 to select rows from a DB2 object in order to provide sample data:

- **Clustered sampling**
  - A specified number of rows are loaded into the editor.

- **Random sampling**
  - Each row of the object is considered for inclusion in the editor. A random number generator and a frequency are used to determine whether the row is actually loaded.

Each of these two methods of selecting sample data are discussed here in detail.

**Clustered data sampling:** To specify clustered data sampling, set the **Sampling type** option on the Editor Options (2 of 8) panel to **2. Clustered sampling**.

These points describe the behavior of the FM/DB2 editor when using clustered data sampling:
- Data sampling applies to browse, view, and edit.
- When using data sampling, the editor always loads all sampled rows into memory. Therefore large table support is NOT available when sampling data.
- The options that are pertinent to clustered sampling are:
  - **Row count** on the function entry panel.
  - **Start position** on the function entry panel.
  - **Sampling limit** on the Editor Options (2 of 8) panel.
  - **Initial skip count** on the Editor Options (2 of 8) panel.
  - **Include count** on the Editor Options (2 of 8) panel.
  - **Skip count** (final) on the Editor Options (2 of 8) panel.

The minimum of these options is applied as the sampling limit. For example, if **Row count** is 0, **Start position** is 1, **Sampling limit** is 0, **Initial skip count** is 0, **Include count** is **nnn**, and **Skip count** (final) is 0, then the minimum of these options (0) will be applied as the sampling limit. An option of 0 is treated as the presence of a **nnn** value. An option of 0, 1, or **nnn** is treated as the absence of a **nnn** value.

**Table 7. Behavior of FM/DB2 editor for options related to clustered data sampling**

<table>
<thead>
<tr>
<th>Row count</th>
<th>Start position</th>
<th>Sampling limit</th>
<th>Initial skip count</th>
<th>Include count</th>
<th>Skip count</th>
<th>Behavior</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td><strong>nnn</strong></td>
<td>0</td>
<td>All rows sampled</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td><strong>bbb</strong></td>
<td>0</td>
<td><strong>nnn</strong></td>
<td>0</td>
<td><strong>bbb</strong> rows sampled¹.</td>
</tr>
<tr>
<td><strong>rrr</strong></td>
<td>1</td>
<td>0</td>
<td>0</td>
<td><strong>nnn</strong></td>
<td>0</td>
<td><strong>rrr</strong> rows sampled¹.</td>
</tr>
<tr>
<td><strong>rrr</strong></td>
<td>1</td>
<td><strong>bbb</strong></td>
<td>0</td>
<td><strong>nnn</strong></td>
<td>0</td>
<td>The minimum of (<strong>rrr</strong>, <strong>bbb</strong>) rows are sampled¹.</td>
</tr>
<tr>
<td>0</td>
<td><strong>sss</strong></td>
<td>0</td>
<td>0</td>
<td><strong>nnn</strong></td>
<td>0</td>
<td>Rows sampled from <strong>sss</strong> until the end of table.</td>
</tr>
</tbody>
</table>

Table 7 describes the behavior of the FM/DB2 editor for various options related to clustered data sampling.
### Handling special data

<table>
<thead>
<tr>
<th>Row count</th>
<th>Start position</th>
<th>Sampling limit</th>
<th>Initial skip count</th>
<th>Include count</th>
<th>Skip count</th>
<th>Behavior</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>sss bbb</td>
<td>0</td>
<td>nnn</td>
<td>0</td>
<td></td>
<td>bbb rows sampled, commencing sss.</td>
</tr>
<tr>
<td>rrr</td>
<td>sss</td>
<td>0</td>
<td>nnn</td>
<td>0</td>
<td></td>
<td>rrr rows sampled, commencing sss.</td>
</tr>
<tr>
<td>rrr</td>
<td>sss bbb</td>
<td>0</td>
<td>nnn</td>
<td>0</td>
<td></td>
<td>The minimum of (rrr, bbb) rows are sampled, commencing sss.</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>xxx nnn yyy</td>
<td></td>
<td></td>
<td></td>
<td>Rows are sampled in groups of xxx+nmm+yyy rows. Within each group, xxx rows are skipped; nnn rows are sampled and yyy rows are skipped. The first group starts at row 1, the second group at (xxx+nmm+yyy+1). Groups are sampled until the end of the result table.</td>
</tr>
<tr>
<td>rrr</td>
<td>1</td>
<td>xxx nnn yyy</td>
<td></td>
<td></td>
<td></td>
<td>Rows are sampled in groups of xxx+nmm+yyy rows. Within each group, xxx rows are skipped; nnn rows are sampled and yyy rows are skipped. The first group starts at row 1, the second group at (xxx+nmm+yyy+1). Groups are sampled until rrr rows are loaded.</td>
</tr>
<tr>
<td>rrr</td>
<td>1</td>
<td>xxx nnn yyy</td>
<td></td>
<td></td>
<td></td>
<td>Rows are sampled in groups of xxx+nmm+yyy rows. Within each group, xxx rows are skipped; nnn rows are sampled and yyy rows are skipped. The first group starts at row 1, the second group at (xxx+nmm+yyy+1). Groups are sampled until the minimum of (rrr, bbb) rows are loaded.</td>
</tr>
<tr>
<td>rrr</td>
<td>1</td>
<td>xxx nnn yyy</td>
<td></td>
<td></td>
<td></td>
<td>Rows are sampled in groups of xxx+nmm+yyy rows. Within each group, xxx rows are skipped; nnn rows are sampled and yyy rows are skipped. The first group starts at row sss, the second group at sss+(xxx+nmm+yyy). Groups are sampled until the minimum of (rrr, rrr) rows are loaded.</td>
</tr>
</tbody>
</table>

**Note:**

1. Sampling continues until one of these conditions is met:
   - Any non-zero Sampling limit is reached.
   - Any non-zero Row count limit is reached.
   - The end of the result table is reached.

**Random data sampling:** To specify random data sampling, set the Sampling type option on the Editor Options (2 of 8) panel to 3. Random sampling.

These points describe the behavior of the FM/DB2 editor when using random data sampling:
Handling special data

- Data sampling applies to browse, view, and edit.
- When using data sampling, the editor always loads all sampled rows into memory. Therefore large table support is NOT available when sampling data.
- The options that are pertinent to random sampling are:
  - **Row count** on the function entry panel.
  - **Start position** on the function entry panel.
  - **Sampling limit** on the Editor Options (2 of 8) panel.
  - **Sampling frequency** on the Editor Options (2 of 8) panel.
  - **Sampling seed** on the Editor Options (2 of 8) panel.

Table 8 describes the behavior of the FM/DB2 editor for various options related to random data sampling.

<table>
<thead>
<tr>
<th>Row count</th>
<th>Start position</th>
<th>Sampling limit</th>
<th>Sampling frequency</th>
<th>Sample seed</th>
<th>Behavior</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0.fff</td>
<td>0</td>
<td>Rows are fetched, starting at row 1 and continuing until the end of the result table.</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>bbb</td>
<td>0.fff</td>
<td>0</td>
<td>Rows are fetched, starting at row 1 and continuing until the end of the result table, or until bbb rows have been added to the editor.</td>
</tr>
<tr>
<td>rrr</td>
<td>1</td>
<td>0</td>
<td>0.fff</td>
<td>0</td>
<td>rrr rows are fetched, starting at the first row. Some subset (approximately 0.fff x rrr) of these rows is added to the sample data set.</td>
</tr>
<tr>
<td>rrr</td>
<td>1</td>
<td>bbb</td>
<td>0.fff</td>
<td>0</td>
<td>At most rrr rows are fetched, starting at the first row. Some subset (approximately 0.fff x rrr) of these rows, but at most bbb rows, is added to the sample data set.</td>
</tr>
<tr>
<td>0</td>
<td>sss</td>
<td>0</td>
<td>0.fff</td>
<td>0</td>
<td>Rows are sampled, starting at row sss, and continuing until the end of the result table.</td>
</tr>
<tr>
<td>0</td>
<td>sss</td>
<td>bbb</td>
<td>0.fff</td>
<td>0</td>
<td>Rows are fetched, starting at row sss, and continuing until the end of the result table or until bbb rows have been added to the editor.</td>
</tr>
<tr>
<td>rrr</td>
<td>sss</td>
<td>0</td>
<td>0.fff</td>
<td>0</td>
<td>rrr rows are fetched, starting at row sss. Some subset (approximately 0.fff x rrr) of these rows is added to the sample data set.</td>
</tr>
<tr>
<td>rrr</td>
<td>sss</td>
<td>bbb</td>
<td>0.fff</td>
<td>0</td>
<td>At most rrr rows are fetched, starting at row sss. Some subset (approximately 0.fff x rrr) of these rows, but at most bbb rows, is added to the sample data set.</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0.fff</td>
<td>0</td>
<td>Rows are fetched, starting at row 1 and continuing until the end of the result table. The random number generator starts with seed yyy.</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>bbb</td>
<td>0.fff</td>
<td>0</td>
<td>Rows are fetched, starting at row 1 and continuing until the end of the result table, or until bbb rows have been added to the editor. The random number generator starts with seed yyy.</td>
</tr>
<tr>
<td>rrr</td>
<td>1</td>
<td>0</td>
<td>0.fff</td>
<td>0</td>
<td>rrr rows are fetched, starting at the first row. Some subset (approximately 0.fff x rrr) of these rows is added to the sample data set. The random number generator starts with seed yyy.</td>
</tr>
<tr>
<td>rrr</td>
<td>1</td>
<td>bbb</td>
<td>0.fff</td>
<td>0</td>
<td>At most rrr rows are fetched, starting at the first row. Some subset (approximately 0.fff x rrr) of these rows, but at most bbb rows, is added to the sample data set. The random number generator starts with seed yyy.</td>
</tr>
</tbody>
</table>
### Handling special data

Table 8. Behavior of FM/DB2 editor for options related to random data sampling (continued).

<table>
<thead>
<tr>
<th>Row count</th>
<th>Start position</th>
<th>Sampling limit</th>
<th>Sampling frequency</th>
<th>Sample seed</th>
<th>Behavior¹</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>sss</td>
<td>0</td>
<td>0.fff</td>
<td>0</td>
<td>Rows are sampled, starting at row sss and continuing until the end of the result table. The random number generator starts with seed yyy.</td>
</tr>
<tr>
<td>0</td>
<td>sss bbb</td>
<td>0</td>
<td>0.fff</td>
<td>0</td>
<td>Rows are fetched, starting at row sss and continuing until the end of the result table, or until bbb rows have been added to the editor. The random number generator starts with seed yyy.</td>
</tr>
<tr>
<td>rrr</td>
<td>sss</td>
<td>0</td>
<td>0.fff</td>
<td>0</td>
<td>rrr rows are fetched, starting at row sss. Some subset (approximately 0.fff x rrr) of these rows is added to the sample data set. The random number generator starts with seed yyy.</td>
</tr>
<tr>
<td>rrr</td>
<td>sss bbb</td>
<td>0</td>
<td>0.fff</td>
<td>0</td>
<td>At most rrr rows are fetched, starting at row sss. Some subset (approximately 0.fff x rrr) of these rows, but at most bbb rows, is added to the sample data set. The random number generator starts with seed yyy.</td>
</tr>
</tbody>
</table>

Note:

1. Sampling continues until one of the following conditions is met:
   - Any non-zero Sampling limit is reached.
   - Any non-zero Row count (fetch) limit is reached.
   - The end of the result table is reached.

   The Sampling limit sets an upper bound on the number of rows loaded into the editor. This is the number of rows in an editor session. To reach this many rows, approximately \( \frac{1}{(\text{Sampling frequency} \times \text{Sampling limit})} \) rows must be fetched. By contrast, the Row count limit sets an upper bound on the number of rows that are fetched from the object. The number of rows that are sampled is approximately Row count x Sampling frequency.

   With "small" sampling frequencies, specifying a low Row count limit may result in no rows being sampled. For example, with a Row count limit of 60, a Sampling count of 2000, and a Sampling frequency of 0.01, there is a reasonable chance that no rows are sampled.

2. A Sampling frequency of 0.fff results in, on average and for large numbers of rows sampled, 0.fff x 100% of rows being sampled. Therefore a frequency of 0.1 results in 10% of rows being sampled. For small frequencies, many rows need to be processed to find each matching row. For any particular random sample of data there is no guarantee that the number of rows in the sample will exactly reflect the sampling frequency.

   When a Sampling seed value of 0 is specified, FM/DB2 uses the fraction part of the second’s value derived from the current system clock value to initiate the random number generator. This value is accurate to the microsecond; therefore each seed has a value in the range 0-999999 inclusive. It is unlikely that two random samples generated with a sampling seed of 0 will be identical.

   When the same user-specified Sampling seed is used for the same table and other conditions, the data samples produced will be identical.

Related references
- “DB2 Browse panel” on page 501
- “DB2 Edit panel” on page 505
- “DB2 View panel” on page 518
- “Editor Options (2 of 8) panel” on page 531

### Null values and VARCHAR processing

FM/DB2 provides different options for processing null values, and varying-length columns.

#### Null column indicator character

FM/DB2 provides options that you can use to specify two null column characters:
Handling special data

Null column indicators: Input

For use when editing a null column

Null column indicators: Display

Indicates a null column when it is displayed

In an FM/DB2 editor session, entering the null column input indicator character in the first (leftmost) column of a column that permits null values indicates that the column is to have a null value. For FM/DB2 to recognize the null column input indicator character, you must type the selected character in the first position of the input field, and remove all the remaining characters.

FM/DB2 also provides an option whereby it interprets a column containing spaces as containing a null value.

Related references

“Editor Options (4 of 8) panel” on page 540

Varying-length columns

You can use FM/DB2 options to specify two string delimiter characters:

String delimiters: Input

For use when editing a varying-length column

String delimiters: Display

Indicates the end of a varying-length column when it is displayed

You can also select an option (Remove trailing spaces) to control how FM/DB2 interprets trailing spaces in a varying-length column.

The following examples show the effect of the Remove trailing spaces option, and the use of the string delimiter character (“#” in this case), if you specify a string (for a varying-length column) in an FM/DB2 editor session. In the examples, “>” and “<” indicate the start and end of the data you type.

• If the Remove trailing spaces option is selected, trailing spaces are removed:

  Example string #<
  Stored as “Example string”.

• If the Remove trailing spaces option is not selected, trailing spaces are not removed:

  Example string #<
  Stored as “Example string ”.

Related tasks

“View and Edit options (options 1 and 2)” on page 43
“Setting options for the current FM/DB2 editor session” on page 95

Related references

“Editor Options (5 of 8) panel” on page 543

Unicode data

The FM/DB2 editor handles character data stored in Unicode differently to data stored in EBCDIC or ASCII. The differences are summarized here.

FM/DB2 normally retrieves character data from DB2 with automatic conversion to the CCSID of the FM/DB2 plan, normally CCSID 37. This is US EBCDIC; other EBCDIC CCSIDs may also be used.
Handling special data

When FM/DB2 retrieves character data stored in Unicode (within DB2), this automatic data conversion does not occur. The FM/DB2 editor processes the character data internally in native Unicode format. This means that, prior to display on the terminal, the data is converted from Unicode to the CCSID of the terminal. This may result in conversion errors when a Unicode character has no corresponding code point in the CCSID of the terminal. Any characters that cannot be converted are shown on the display as periods, indicating an unconverted character. When data displayed (and possibly changed) on the terminal is processed, it is converted from the CCSID of the terminal into Unicode, prior to submission to DB2. This conversion will always be successful, since Unicode includes code points for all characters in commonly used CCSIDs.

Data corruption is possible when the displayed data includes a period indicating an unconverted Unicode character, and the data for the column is changed by overtyping. Any periods in the modified data are converted to the Unicode equivalent. Therefore, if the period represents an unconverted Unicode character, that character will be corrupted by the change. When character data is displayed in hexadecimal mode, the hexadecimal characters normally represent the encoding for the displayed character in the CCSID of the terminal. For example, ‘A’ is represented by ‘C1’x in EBCDIC. When the character data is for a Unicode-encoded column, the hexadecimal characters represent the native Unicode data, not the encoding of the displayed character in the CCSID of the terminal. For example, ‘A’ is represented by ‘41’x when Unicode data is being displayed.

You can also make changes to the data in a column stored in Unicode by turning hexadecimal display on (HEX ON command) and overtyping the hexadecimal characters directly. When you do this, you should use the Unicode representation of any character, not the terminal’s CCSID representation.

Assuming the target column is character data stored within DB2 in Unicode, the FIND, CHANGE, and EXCLUDE commands operate like this:

- Strings such as ‘ABC’ are converted to Unicode prior to any search or change occurring.
- Hexadecimal strings such as ‘414243’x are not converted to Unicode prior to any search or change occurring. Therefore F ‘ABC’ and F ‘414243’x are equivalent commands when the target column contains Unicode data, but not when the target column contains EBCDIC data.

Changing data

In an FM/DB2 editor session, you can change data by:

- Overtyping
- Using the CHANGE primary command
- Entering prefix commands in the prefix area next to each row

Related tasks

- “Installation settings that affect CAPS”
- “Changing data by overtyping” on page 139
- “Changing data with the CHANGE command” on page 140
- “Changing data using prefix commands” on page 144

Installation settings that affect CAPS

The setting of CAPS determines whether or not the characters you enter in an FM/DB2 editor session are converted to uppercase. If CAPS is set to ON, FM/DB2
translates all data in rows that are changed by overtyping from lowercase to uppercase, and translates replacement data specified in the CHANGE command from lowercase to uppercase. If CAPS is OFF, the new data you type is not converted to uppercase.

When FM/DB2 is installed, a number of customization settings affect CAPS. These settings determine:

- The initial setting for CAPS (ON or OFF)
- Whether you can change this setting by means of the edit options; that is, whether the setting is “updateable” (meaning you can change it) or “fixed” (meaning you cannot change it)
- Whether you are permitted to issue the FM/DB2 CAPS primary command

For more information about these customization settings, see the Customization Guide.

For the first and second of the above settings, you can verify their value by looking at the third Editor Options panel.

For the third of the above settings, if you try to issue the CAPS command but FM/DB2 has been customized so that the CAPS command is unavailable, FM/DB2 displays the message “The CAPS command has been disabled at installation time.”.

If the installation option for CAPS is “fixed”, each FM/DB2 editor session starts with the installation setting for CAPS (either ON or OFF) and you cannot change the setting using the edit options.

If the installation option for CAPS is “updateable”, you can change the setting (to either ON or OFF) using the edit options and each FM/DB2 editor session starts with the current CAPS setting.

If the CAPS command is disabled, and the installation option for CAPS is “fixed”:
- If the installation option for CAPS is ON, to enter lowercase data you must first issue the CASE MIXED command
- If the installation option for CAPS is OFF, to enter uppercase data you must use the Shift or Caps Lock keys

Related tasks
- “Changing data by overtyping” on page 140
- “Changing data with the CHANGE command” on page 140
- “Converting data to uppercase” on page 146

Related references
- “Editor Options (1 of 8) panel” on page 528
- “CAPS primary command” on page 730
- “CASE primary command” on page 731

Changing data by overtyping
You can change the data displayed in an FM/DB2 editor session by overtyping it with the new data.

To successfully change data by overtyping:
The new data you enter must be compatible with the data type of the column being changed. For example, you can only enter numeric values in a column that has a numeric data type.

The column must be updateable (as defined in the DB2 catalog).

If the table is subject to referential integrity, the new data must not cause a referential integrity violation.

Related tasks

- “Read-only” views, non-updateable columns” on page 147
- “Editing related tables” on page 149
- “Converting data to uppercase” on page 146

Related references

Chapter 14, “DB2 data types,” on page 361

Changing data with the CHANGE command

When you use the CHANGE command to change data, you must specify one or more column numbers to be included in the search (in the same way as you use the FIND command to find data).

For example, the following command changes all occurrences of the string “CHAIR” in all rows to the string “TABLE” in either of the columns with column numbers #5 and #6.

```
CHANGE ALL CHAIR TABLE #5,#6
```

And the following command changes the next occurrence of the value 400 in column number 3 to 500:

```
CHANGE 400 500 #3
```

To change the next occurrence of the string, press the RChange function key (F6). To find the next occurrence of the string and optionally change it, use a combination of the RFind function key (F5) and the RChange function key (F6).

To find the same string as you specified on the previous CHANGE primary command, specify an asterisk (*) as the search string.

To use the same replacement string as you specified on the previous CHANGE primary command, specify an asterisk (*) as the replacement string.

Further examples

```
CHANGE BLACK WHITE #4 #7
```

In TABL display format, changes the next occurrence of “BLACK” to “WHITE”, where “BLACK” is completely contained within either of the columns represented by column numbers 4 and 7.

In SNGL display format, changes the next occurrence of “BLACK” to “WHITE”, where “BLACK” is completely contained within either of the columns represented by column numbers 4 and 7, in the current row only.

```
CHANGE BLACK WHITE ALL #2 #8
```

In TABL display format, changes all occurrences of “BLACK” to “WHITE”, where “BLACK” is completely contained within either of the columns represented by column numbers 2 and 8, in all rows.
In SNGL display format, changes all occurrences of “BLACK” to “WHITE”, where “BLACK” is completely contained within either of the columns represented by column numbers 2 and 8, in the current row only.

Related concepts
- “Installation settings that affect CAPS” on page 138

Related tasks
- “Changing data within excluded or non-excluded rows” on page 143
- “Changing data within a label range” on page 144
- “Finding data within specific columns” on page 113

Related references
- “CHANGE primary command” on page 732
- “Table Edit panel” on page 691

“From” and “To” strings of different lengths
For character columns (CHAR, VARCHAR), you can specify “from” and “to” strings (from_string and to_string) of different lengths, subject to the following conditions:

- For fixed-length character columns, when the “to” string (to_string) is shorter than the “from” string (from_string), FM/DB2 performs the change and pads the column with spaces. When the “to” string (to_string) is longer than the “from” string (from_string), FM/DB2 performs the change only if there are enough unused characters in the column to accommodate the change.

For example, for a given row, if column #2 is a fixed-length, character column of length 8 containing “GEORGE ” (with two trailing spaces):

```
CHANGE GEORGE FRED #2
Changes “GEORGE ” (with two trailing spaces) to “FRED ” (with four trailing spaces) in that row.
```

```
CHANGE GEORGE WILLIAM #2
Changes “GEORGE ” (with two trailing spaces) to “WILLIAM ” (with one trailing space) in that row.
```

```
CHANGE GEORGE ALEXANDER #2
FM/DB2 does not perform the change to that row as there are not enough unused characters in the column to accommodate the change.
```

- For varying-length character columns, when the “to” string (to_string) is shorter than the “from” string (from_string), FM/DB2 performs the change and adjusts the length of the column downwards by the difference between the lengths of to_string and from_string. When the “to” string (to_string) is longer than the “from” string (from_string), FM/DB2 performs the change only if there are enough unused characters in the column to accommodate the change. In this case, FM/DB2 adjusts the length of the column upwards by the difference between the lengths of from_string and to_string.

Handling long strings
For character columns (CHAR, VARCHAR), you can specify “from” and “to” strings (from_string and to_string) up to 100 characters long. However, when you are dealing with long strings, you may find that you cannot fit the whole CHANGE command (including other parameters) on the command line. To overcome the problem, do one of the following to display an Extended Command Entry pop-up panel (see Figure 44 on page 142):

- Enter CX (for Change Extended)
Changing data

- Enter the CHANGE primary command (or one of its abbreviations, such as C) with no parameters

**Note:** If you have previously used the CHANGE or FIND commands in the current FM/DB2 editor session, to display the Extended Command Entry pop-up panel in this way, you will need to first use the RESET command before entering the CHANGE command (or one of its abbreviations) without any parameters. Alternatively, enter `CX`.

The Extended Command Entry pop-up panel contains five lines (each 50 characters long) that you use to enter all the CHANGE command parameters (but not the actual keyword CHANGE or any of its abbreviations). FM/DB2 treats the five lines on the pop-up panel as contiguous, allowing you to specify long “from” and “to” strings, the DB2 column number and optionally any other parameters, using up to 250 characters in total.

**Note:** You can continue a string (or any other parameter) from the last position on one line of the pop-up panel to the first position of the next line. However, this does not mean that you need to completely fill a line before using the next line. For example, you can specify the “from” string on the first of the five lines, the “to” string on the next line, and the DB2 column number (specified as `#n`) on the next line. If you do not use all of a line, FM/DB2 treats the unused part of the line as a single space.

**Figure 44** shows the Extended Command Entry pop-up panel for the CHANGE command. In this example, any previous occurrences of the string “dichlorodiphenyltrichloroethane” in DB2 columns 4 or 7, between the labels .abc and .def, are to be changed to the string “trinitrophenylmethylnitramine” for non-excluded rows only.

**Figure 45** on page 143 shows the pop-up panel for the CHANGE command where the next occurrence of a long string (84 characters) in DB2 column 5 is to be changed to another long string (70 characters).
Changing numeric columns
If you use the CHANGE command to change a numeric column, then the lengths of from_string and to_string are not significant. For example, if column #2 is a numeric column, then the following CHANGE command is permitted, because from_string and to_string are interpreted as numeric values:

```
CHANGE 3.1415 3.2 #2
```

Changing data within excluded or non-excluded rows
If your data contains excluded and non-excluded rows, you can limit the effect of the CHANGE command to:
- Only non-excluded rows (parameter NX), or
- Only excluded rows (parameter EX or X)

For example, the following command changes all occurrences of the string “GREEN”, in column number 2 or column number 5 in non-excluded rows only, to the string “BLACK”:  

```
CHANGE ALL GREEN BLACK (#2 #5) NX
```

And the following command changes the next occurrence of the value 250, in column number 2 in an excluded row, to 520:

```
CHANGE 250 520 #2 EX
```

Related tasks
"Excluding rows" on page 126

Related references
"CHANGE primary command" on page 732
Changing data

Changing data within a label range
You can limit the effect of the CHANGE command to a range of rows, identified by a pair of labels indicating the first and last rows to be searched for the data you want to change.

For example, in the range of rows from the label .LABA to the end of the data, the following command changes the next occurrence of the string “EVIL” in column number 4 to the string “GOOD”:
CHANGE EVIL GOOD #4 .LABA .ZLST

And, in the range of rows from the current cursor position to the label .LABB, the following command changes all occurrences of the value 48 in column number 3 or column number 11 to the value 148:
CHANGE ALL 48 148 #3,#11 .ZCSR .LABB

Related tasks
Assigning labels to rows on page 116

Related references
CHANGE primary command on page 732

Limiting the change to specific portions of the data
You can limit the effect of the CHANGE command to specific portions of the data you are searching:
• To limit the search for a string to only where it appears as a prefix in the data, specify the PREFIX parameter.
• To limit the search for a string to only where it appears as a suffix in the data, specify the SUFFIX parameter.
• To limit the search for a string to only where it appears as a word in the data, specify the WORD parameter.

Related references
CHANGE primary command on page 732

Changing data using prefix commands
In an FM/DB2 View or Edit editor session, you can enter prefix commands in the prefix area of one or more rows on the display. The prefix area is typically represented by a column of 6-digit numbers on the far left or far right of your display.

You can use prefix commands to:
• Insert or delete rows
• Repeat (duplicate) rows
• Move rows
• Overlay data in one or more rows with data from one or more other rows
• Shift data
• Copy or move rows to and from a clipboard
• Limit the data being edited

To set the position of the prefix area (left or right), or turn the prefix area on or off, use the PREFIX primary command.

You can use most of the prefix area commands on either a single row or a block of consecutive rows.
Changing data

- To perform an operation on a single row, you enter the appropriate command code.
- To perform an operation on a block of consecutive rows, either enter the command code preceded or followed by the number of lines, or enter the appropriate block command code at the start and end lines of the block.

Generally, you need to type over only the first 1 or 2 characters of the line number to enter a prefix command. Sometimes, however, typing a single character can be ambiguous. In the following example, it is unclear whether the intended prefix command is R to repeat line 31700, or R3 to repeat the line three times:

<table>
<thead>
<tr>
<th>Line</th>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>031600</td>
<td>R31700</td>
</tr>
<tr>
<td>031800</td>
<td></td>
</tr>
</tbody>
</table>

In such cases, the editor assumes that you have not typed a number following the prefix command (it cannot detect if you overtype a character with the same character). If you want to repeat the line three times, you can use any of the following procedures:

- Type one or more blanks following the R3:
  - R3 700
- Type R3 and press the Erase EOF key to clear the rest of the Line Command field, or press the Erase EOF key and then type R3.
- Type one or more blanks after the R but before the number such that the number when entered is different than the characters being overtyped:
  - R 3700
- Type the number before the R, ensuring that the number when entered is different than the characters being overtyped:
  - 3R1700

To clear any outstanding commands in the prefix area, use the RESET primary command.

**Note:** Moving rows in an FM/DB2 editor session does not affect the order in which the rows are stored in the underlying DB2 table.

**Overlaying data in existing rows**

You can overlay data in a row with data from another row by specifying the O or OO prefix commands.

When data is to be copied or moved by the C (copy) or M (move) prefix commands and overlaid on one or more existing rows, the O or OO (overlay) prefix commands specify the destination for the data. You can only overlay data onto character columns (CHAR and VARCHAR).

The data that is copied or moved overlays blanks in the destination rows.

If the destination column is VARCHAR:

- It must have a non-zero length.
- If the length of the data being moved or copied is longer than the destination column, only the data that can fit into the destination column is moved or copied.

When data is to be moved or copied and then overlaid on a destination:

- Where the destination is a single row:
Changing data

- The O (overlay) prefix command specifies the destination for the data. You can type a number after the O prefix command to specify the number of times that the M or C prefix command is to be performed. For example, typing the command O3 against a row causes the data to be moved or copied and then overlaid on that row and also the next two rows.

• Where the destination is a block of rows:
  - The OO (overlay, multiple-line target) prefix command specifies the first and last row of the destination for the data.

To overlay one or more single rows:
1. Type either M or C in the prefix area of the row that is to be moved or copied.
2. Type O in the prefix area of the row that the moved or copied record is to overlay.
3. Press Enter. The data being moved or copied overlays the specified row or rows.

To overlay one or more blocks of rows:
1. Type either MM or CC in the prefix area of the first and last rows of a block of rows that is to be moved or copied. You can scroll (or use FIND or LOCATE) between typing the first command and the second command, if necessary.
2. Type 00 in the prefix area of the first and last rows that the block of rows being moved or copied is to overlay. Again, you can scroll (or use FIND or LOCATE) between typing the first 00 and the second 00, if necessary.
3. Press Enter. The rows that contain the two CC or MM commands and all of the rows between them overlay the rows that contain the two OO commands and all of the rows between them.

The number of source and receiving rows need not be the same. If there are more receiving rows, the source rows are repeated until the receiving rows are gone. If there are more source rows than receiving rows, the extra source rows are ignored.

Related topics
- “Table Edit panel” on page 691

Related references
- “Table Edit panel” on page 691
- “PREFIX primary command” on page 767
- “RESET primary command” on page 771

Converting data to uppercase
In an FM/DB2 editor session, the CAPS and CASE primary commands control whether data in the data area is converted to uppercase on input (CAPS, CAPS ON or CASE UPPER), or left alone (CASE, CASE MIXED or CAPS OFF; this is the default action).

You can abbreviate the CAPS command to CAP and the CASE command to CAS.

The CAPS ON and CASE UPPER commands have slightly different effects.

CAPS or CAPS ON is similar to the ISPF editor CAPS command:
• If you overtype any data in a row, only the current column is converted to uppercase.
Changing data

• The CHANGE command treats the new (replacement) string as if it were specified in all uppercase unless:
  – The new string is enclosed in single quotation marks, prefixed by the letter C (for example, C‘New String’), or
  – The new string is specified as a hexadecimal string (for example, X'C1C2C3')
If either of the previous conditions apply, the new string is not converted to uppercase.

CAPS or CAPS ON affects only rows changed after the CAPS or CAPS ON command is entered.

CASE UPPER converts all data in any new or changed rows to uppercase, regardless of how the rows are changed. CASE UPPER affects all rows inserted or changed in the current FM/DB2 editor session, including rows inserted or changed before the CASE UPPER command is entered.

The CAPS OFF, CASE, and CASE MIXED commands have the same effect: they turn off conversion to uppercase.

Any CAPS command overrides any previously entered CASE command, and any CASE command overrides any previously entered CAPS command.

Note:
1. In the ISPF editor with CAPS ON, a line is considered changed if you type anything on the line (if you overtype a space with a space, it is considered to be a change). However, in the FM/DB2 editor, overtyping a character with the same character is not considered a change.
2. The values for CASE and CAPS are not maintained between FM/DB2 editor sessions. Each new FM/DB2 editor session begins with CAPS OFF; that is, data entered in lowercase is not converted to uppercase.

Related concepts
"Installation settings that affect CAPS" on page 138

Related tasks
"Changing data by overtyping" on page 139

Related references
"CAPS primary command" on page 730
"CASE primary command" on page 731
"Table Edit panel" on page 691
"Editor Options (1 of 8) panel" on page 528

“Read-only” views, non-updateable columns

If the object you are editing is a view, it is possible that one or more columns within the view are marked as non-updateable in the DB2 catalog. In DB2 parlance, such a view is referred to as a read-only view.

If you attempt to edit a read-only view that contains only non-updateable columns, FM/DB2 displays a Browse session (instead of an Edit session) and issues an advisory message at the start of the session. Otherwise (that is, if the read-only view contains at least one updateable column), FM/DB2 displays an FM/DB2 editor session showing the non-updateable columns as protected and issues an
Changing data

advisory message as the start of the FM/DB2 editor session. If you edit a view containing one or more non-updateable columns, not all edit operations are possible.

If you edit a table that contains one or more columns, excluding ROWID columns, that are marked as non-updateable in the DB2 catalog (such as a DB2 catalog table), FM/DB2 displays an FM/DB2 editor session showing the non-updateable columns as protected and issues an advisory message at the start of the FM/DB2 editor session. If you edit a table containing one or more non-updateable columns, excluding ROWID columns, not all edit operations are possible.

Related tasks
- "ROWID columns" on page 76

Deleting rows

In an FM/DB2 editor session, you can delete rows by using:
- The DELETE primary command
- The D, Dn, or DD prefix commands in the prefix area next to each row (TABL display only)

In TABL display format, you can restrict the rows you delete by:
- Limiting the rows that are deleted to only excluded or only non-excluded rows
- Limiting the rows that are deleted to those within a range of rows identified by a pair of labels
- Using the D, Dn, or DD prefix commands

In SNGL display format, DELETE deletes the currently displayed row only. To delete the currently displayed row, enter:
DELETE

Note:
1. In TABL display format, care should be taken when using the DELETE primary command to ensure that you only delete the rows you intend to. The command DELETE ALL (with no other parameters) deletes all rows in the table or view being edited.
2. If the table target has referential constraints, you might not be able to delete rows. For further information, see "Handling errors during an Edit session" on page 157.

Related tasks
- "Deleting rows within excluded or non-excluded rows" on page 74
- "Deleting rows within a label range" on page 149

Related references
- "DELETE primary command" on page 737
- "Table Edit panel" on page 691

Deleting rows within excluded or non-excluded rows

If your data contains excluded and non-excluded rows, you can limit the effect of the DELETE command to:
- Only non-excluded rows (parameter NX), or
- Only excluded rows (parameter EX or X)
Deleting rows

For example, the following command deletes all excluded rows:

```
DELETE ALL EX
```

And the following command deletes the first non-excluded row:

```
DELETE NX
```

Related tasks

- “Excluding rows” on page 126

Related references

- “DELETE primary command” on page 737

Deleting rows within a label range

You can limit the effect of the DELETE command to a range of rows, identified by a pair of labels indicating the first and last rows to be deleted.

For example, the following command deletes all rows from the label .LABA to the label .LABB.

```
DELETE ALL .LABA .LABB
```

Related tasks

- “Assigning labels to rows” on page 116

Related references

- “DELETE primary command” on page 737

Editing related tables

FM/DB2 provides support for editing tables that are subject to referential integrity (RI) constraints. If you are not familiar with the terms parent table, dependent table, primary key, foreign key and referential integrity, see the relevant DB2 manuals.

Note: The support provided by FM/DB2 for editing related tables applies only to DB2 tables, not to any DB2 view created on a table that is subject to RI constraints.

This section describes the behavior of the REDIT command when a DB2 error has been encountered, and the DB2 error is an RI violation. For a description of issuing the REDIT command when no RI error has occurred, see “Listing related tables” on page 150.

If FM/DB2 encounters an error when attempting to save the data changed in an Edit session and the error is due to a RI violation, it automatically determines the name of the other table (the related table) in the RI relationship. You can then start another FM/DB2 Edit session for the related table. The new Edit session (dependent session) is displayed as a new ISPF logical session, as if you had typed START on the command line and then selected FM/DB2 Edit. You can use the ISPF commands SWAP and SWAP LIST to move between different logical sessions.

FM/DB2 customizes the template for the new Edit session to make the identification of the key columns easier. If the related table is a dependent table, FM/DB2 displays only the rows with values that match the primary key in the parent table. If the related table is a parent table, you can use the IP (Insert primary key) prefix command to insert a row with a primary key matching the foreign key that caused the error.
If an Edit session has active dependent Edit sessions, you cannot save changes made in the session until you end all dependent Edit sessions. If you cancel an edit session that has active dependent Edit sessions, all dependent Edit sessions are also canceled. Any uncommitted changes in any of the dependent Edit sessions are lost.

Listing related tables
When you are in an FM/DB2 View or Edit editor session, you can issue the REDIT primary command to display a list of tables related by a DB2 referential integrity (RI) constraint to the DB2 object currently being edited.

When you enter the REDIT primary command, you can optionally supply a field number that identifies the column in the table of the DB2 object currently being edited.

The way in which FM/DB2 responds after you enter the REDIT primary command depends on the object you are editing, the position of the cursor, and whether you specify a column number:

- When the object being edited does not have RI constraints, a new FM/DB2 Edit session is stacked on top of the existing Edit session. You are presented with the DB2 Edit entry panel.
- When the object being edited does have RI constraints:
  - When the cursor is not located in the data portion of the screen, or is not located on the data for a column of the table, FM/DB2 displays a selection list showing all the DB2 objects that are directly related to the object being edited by an RI constraint. This includes tables that are parent tables in a relationship with the table being edited, and tables that are dependent tables in a relationship with the table being edited.
  - When the cursor is located on the data for a column of the table, and that column is not part of a primary, parent or foreign key, FM/DB2 displays a selection list showing all the DB2 objects that are directly related to the object being edited by an RI constraint. This includes tables that are parent tables in a relationship with the table being edited, and tables that are dependent tables in a relationship with the table being edited.
  - When the cursor is located on the data for a column of the table, and that column is part of a primary, parent, or foreign key, FM/DB2 displays a selection list showing only those DB2 objects that are directly related to the object being edited, limited to those tables where the primary, parent, or foreign key of the table being edited contains the column where the cursor is located.
  - When you have specified a field number with the REDIT primary command and the column identified by the field number is part of a primary, parent, or foreign key, FM/DB2 displays a selection list showing only those DB2 objects that are directly related to the object being edited, limited to those tables where the primary, parent, or foreign key of the table being edited contains the column identified by the field number.
  - When a field number is provided with the REDIT command and the column identified by the field number is not part of a primary, parent, or foreign key, FM/DB2 displays a selection list showing all the DB2 objects that are directly related to the object being edited by an RI constraint. This includes tables that are parent tables in a relationship with the table being edited, and tables that are dependent tables in a relationship with the table being edited.
Editing related tables

The list of related tables for an object is displayed by the FM/DB2 Object List utility on the Related tables panel. You can issue a number of prefix commands against any of the objects shown on the list. For example:

Prefix command

FM/DB2 displays...

EE or VE
A new FM/DB2 editor session for the selected object. The new FM/DB2 editor session appears in a new ISPF logical screen (as though you had split the screen and started a new FM/DB2 session).

The first panel displayed, and the rows displayed in the Edit (or View) session, depend on where you position the cursor at the time of entering the REDIT command:

• When the cursor is located on the data for a column that is part of a parent, primary, or foreign key, the new FM/DB2 editor session starts with the display of data, and contains only those rows related to the row where the cursor was located.

• When the cursor is not located on the data for a column that is part of a parent, primary, or foreign key, the new FM/DB2 editor session starts with the display of the DB2 Edit (or DB2 View) entry panel (where you can alter the object name if required), and contains all rows of the object.

E or V A new FM/DB2 Edit session stacked on top of the existing Edit session.

To see all the available commands, type “?” in the selection field (SEL) to the left of any entry and press Enter.

You can only issue the REDIT command when editing a table. The REDIT command is inoperative when editing a view.

Related references

Chapter 8, “Working with lists of DB2 objects,” on page 241
“Related tables panel” on page 639
“REDIT primary command” on page 770

Self-referencing constraints

A self-referencing constraint exists if a DB2 object is subject to a primary or foreign key relationship in which the parent table and the dependent table are the same table. If the DELETE rule for the relationship is CASCADE, the deletion or change of one row can cause a recursive deletion of other rows in the table. If FM/DB2 detects a self-referencing constraint defined on a table, it does not attempt to determine which rows are affected by a deletion or change of a primary key value. This also applies to any other tables affected by other relationships defined on the table.

Referential integrity errors

There are two types of referential integrity errors:

• The first type of referential integrity error occurs if you are editing a parent table and you either delete or update a primary key value. If the restrict rule applies to foreign key values in the dependent table, DB2 rejects the update or delete operation. To make the change to the primary key value in the parent table, you must edit the dependent table and either delete or change all rows with a matching foreign key.
The second type of referential integrity error occurs if you are editing a dependent table and you create a foreign key value for which there is no corresponding entry in the parent table. DB2 rejects the insert or update operation (because a foreign key value cannot exist without a corresponding primary key value). To change the value of the foreign key in the dependent table, you must edit the parent table and create a row with a primary key value that matches the value of the new foreign key value in the dependent table.

Related tasks
“Handling errors during an Edit session” on page 157

Starting an Edit session of a related table
To start an FM/DB2 Edit session of a related table, perform one of these actions:
- In the information panel for an RI error (accessed by typing E in the prefix area for an RI error), type \REDIT on the command line.
- Use the RE prefix command against a row marked in error if the error is an RI error.

Note: When the object being edited is a DB2 view, both the \REDIT line command and the RE prefix command are disabled.

Related references
“REDIT primary command” on page 770
“Table Edit panel” on page 691

Differences when you edit a related table
An FM/DB2 Edit session for a related table is similar to a typical Edit session, with the following exceptions:
- The new session starts with the display of data for the related table.
- You cannot change the edit options. The current global edit options apply.
- There is no opportunity to edit the template before displaying the data.
- The panel title shows “Table Edit (related)” (instead of “Table Edit”).
- The location information is replaced with either “PARNT” or “DPNDT” and the name of the other object in the relationship. “PARNT” indicates that the other table is the parent table in the relationship. “DPNDT” indicates the other table is the dependent table in the relationship.
- For a dependent table, the template includes a WHERE clause that limits the data displayed to only those rows that match the primary key value in the parent table. If the parent table is subject to a self-referencing constraint, all rows might be displayed.
- The template includes information about the order in which columns are displayed. Columns that comprise the primary or foreign key are displayed (in order) at the left (TABL display format), or top (SNGL display format), of the other object in the relationship.
- In TABL display format, columns that form part of the relationship’s key are identified with a “K” in the sixth character of the scale header line.
- If an Edit session has active dependent Edit sessions (that is, you have used either the \REDIT primary command or the RE prefix command to start another FM/DB2 edit session of a related table, and that Edit session is still active), you cannot save any changes you have made to data until you end all dependent Edit sessions.
Editing related tables

- If you cancel an Edit session that has active dependent Edit sessions, FM/DB2 also cancels all dependent Edit sessions. Any uncommitted changes in the canceled dependent Edit sessions are lost.

Related tasks
"Self-referencing constraints" on page 151

Related references
"REDIT primary command" on page 770
"Table Edit panel” on page 691

Differences when saving or canceling while editing a related table

If you have initiated one or more Edit sessions for related tables, the following restrictions apply:
- If there are active dependent Edit sessions (that is, you have used either the REDIT primary command or the RE prefix command to start another FM/DB2 Edit session for a related table, and any of those edit sessions are still active), you cannot save any changes to data until you have ended all dependent Edit sessions.
- If you cancel an Edit session that has active dependent Edit sessions, any dependent Edit sessions are also canceled. Any uncommitted changes in the canceled dependent Edit sessions are lost.

Related references
"REDIT primary command” on page 770
"Table Edit panel” on page 691

Navigating between Edit sessions

To switch between two or more FM/DB2 Edit sessions, use the ISPF commands SWAP and SWAP LIST. For more information, see the z/OS ISPF User’s Guide Vol 1.

If the table you are editing is part of a larger RI structure, you might need to start several Edit sessions of related tables to make the required changes to the original table.

Deleting a primary key

If you delete a row in a parent table, it might generate an error if there are rows in the dependent table and the delete restrict rule is specified for the relationship. The delete restrict rule prevents the deletion of the primary key value unless you first delete (or change) all the rows in the dependent table with foreign key values matching the primary key value.

In this situation:
1. Start an FM/DB2 Edit session for the dependent table by either using the RE prefix command, or REDIT primary command from the error information panel. Usually, only those rows that would be affected by the deletion of the primary key in the parent table are displayed (see following note).
2. Either delete the displayed rows (see following note), or change the values of the foreign keys to another valid value.
3. Press the Exit function key (F3) to validate and commit your changes.
4. Return to the Edit session for the primary table and proceed with deletion of the primary key value.
Editing related tables

Note:

1. In most cases, FM/DB2 shows only those rows that would be affected by the deletion of the primary key in the parent table. The exception is the situation where the parent table has a self-referencing constraint that might cause the deletion of other rows in the parent table if a primary or parent key value is deleted or changed. In this situation, FM/DB2 shows all rows and you must select the dependent rows.

2. If the parent or dependent tables are also parent or dependent tables in other relationships, changes to the primary or foreign key values can result in other errors.

Related references

“REDIT primary command” on page 770
“Table Edit panel” on page 691

Inserting a new foreign key

If you insert a row in a dependent table, it might generate an error if there is no row in the parent table with a corresponding primary key value. DB2 prevents the insertion of the foreign key value until a primary key is created in the parent table.

The parent table can be the same as the table currently being edited (a self-referencing constraint. This is indicated by the absence of an “R” in the rightmost position of the prefix area for the row in error. You can use the E prefix command to display an information panel with details of the relationship name, the parent and dependent table names, and the names of the columns in the key in both the parent and dependent tables.

In the situation where the parent and dependent tables are different:

1. Start an FM/DB2 Edit session for the parent table by either using the RE prefix command, or REDIT primary command from the error information panel.

2. Use the IP prefix command to insert a new row in the table, with a primary key that matches the foreign key value entered in the Edit session for the dependent table. You can also change an existing row so that its primary key matches the foreign key entered in the Edit session of the dependent table.

3. Press the Exit function key (F3) to validate and commit your changes.

4. Return to the original Edit session and save the new foreign key value.

Note: If the parent or dependent tables are also parent or dependent tables in other relationships, changes to the primary or foreign key values can result in other errors.

Related tasks

“Self-referencing constraints” on page 151

Related references

“REDIT primary command” on page 770
“Table Edit panel” on page 691

Updating a primary key

DB2 interprets updating a primary key as a deletion, followed by an insertion of a new row the same as the deleted row but with the new primary key value.
Editing related tables

If you update a row in a parent table, it can generate an error if there are rows in the dependent table with matching foreign key values and the delete restrict rule is specified for the relationship. The delete restrict rule prevents the deletion of the primary key value unless you first delete (or change) all the rows in the dependent table with foreign key values matching the primary key value.

In this situation:
1. Repeat the row containing the original primary key value you want to change.
   
   Note: After the repeat operation, the new row is displayed immediately below the repeated row and is intensified.
2. Change the primary key in the new row to the required value.
3. Use the SAVE primary command to verify that there are no other DB2 errors.
4. Delete the original row.
5. Start an FM/DB2 Edit session for the dependent table by either using the RE prefix command, or REDIT primary command from the error information panel. Usually, only those rows that would be affected by the deletion of the primary key in the parent table are displayed (see following note).
6. Either delete the displayed rows (see following note), or change the values of the foreign keys to another valid value.
7. Press the Exit function key (F3) to validate and commit your changes.
8. Return to the Edit session for the primary table and proceed with deletion of the primary key value.

Note:
1. In most cases, FM/DB2 shows only those rows that would be affected by the deletion of the primary key in the parent table. The exception is the situation where the parent table has a self-referencing constraint that might cause the deletion of other rows in the parent table if a primary or parent key value is deleted or changed. In this situation, FM/DB2 shows all rows and you must select the dependent rows.
2. If the parent or dependent tables are also parent or dependent tables in other relationships, changes to the primary or foreign key values can result in other errors.

Related references

- "REIT primary command" on page 770
- "SAVE primary command" on page 775
- "Table Edit panel" on page 691

Adding a primary key

You can add a row with a new primary key value to a table using one of these methods:

- You can insert a new row in the table (I prefix command) and then change the values in the new row to create the required primary key value.
- You can repeat an existing row and then change the values in the new row to create the required primary key value.

Note: If you use this method, you must change the new row (added immediately below the original row, and intensified), rather than the original row. The reason for this is that changing the original row instead of the new row results in FM/DB2 attempting to update the original row with the new primary key value.
key value. This change fails if the original row has dependent rows, and SQLCODE -531 results. FM/DB2 also reports a duplicate row error (SQLCODE -803) against the new row as it attempts to insert a new row, identical to the original row.

Related tasks
“Updating a primary key” on page 154
“Handling errors during an Edit session” on page 157

Related references
“Table Edit panel” on page 691

Issuing DB2 commands from within an FM/DB2 session

When you are working in an FM/DB2 session, you can issue a DB2 command and view the output returned by DB2.

There are two ways in which you can enter a DB2 command from within an FM/DB2 session:
• By displaying the Enter and Execute DB2 Commands panel and entering the required DB2 command (prefixed by "DB2 ") on the panel.
• From some FM/DB2 panels, entering the required DB2 command on the command line.

Issuing DB2 commands from the Enter and Execute DB2 Commands panel

To issuing DB2 commands from the Enter and Execute DB2 Commands panel:
1. Display the Enter and Execute DB2 Commands panel by performing one of these actions:
   • Select option 6 (Command) from the Primary Option Menu panel and press Enter.
   • From the Process pull-down menu on the Action bar, select option 1 (DB2 Command) and press Enter.
2. On the Enter and Execute DB2 Commands panel, under **DB2 Command**, type a hyphen (-) followed immediately by the DB2 command you want to issue.
   For example:
   -DISPLAY ARCHIVE

   **Note:** If you do not prefix the command with a hyphen, one is added by FM/DB2.
3. When you issue either the DISPLAY THREAD or DISPLAY DATABASE command, you can optionally specify the maximum number of lines to be returned in the **Line count** field.
4. Press Enter.
   FM/DB2 displays the output from the DB2 command in an ISPF View session as shown in Figure 46 on page 157

Issuing DB2 commands from the command line

You can issuing a DB2 command from the command line on these FM/DB2 panels:
• Table Browse panel
• Table Edit panel
• Table View panel
• Object list panels displayed with the Object List utility
Issuing DB2 commands from within an FM/DB2 session

- Privilege details panels
- Enter, Execute and Explain SQL Statements panel

To issue a DB2 command from the command line of an FM/DB2 panel:

1. Type DB2, followed a space, and then the DB2 command you want to issue.
   For example, to issue the DB2 command DISPLAY ARCHIVE, type:
   DB2 DISPLAY ARCHIVE

2. Press Enter.
   FM/DB2 displays the output from the DB2 command in an ISPF View session as shown in Figure 46.

Related references
- “Enter and Execute DB2 Commands panel” on page 554

Handling errors during an Edit session

FM/DB2 might report errors if it attempts to implement the changes to data that you have made in an Edit session. These errors are detected by DB2 when FM/DB2 issues INSERT, UPDATE, or DELETE SQL statements. SQL errors can arise for many reasons, including violation of uniqueness, referential integrity and check data constraints defined on the DB2 object being edited. If you are unfamiliar with these concepts, see the DB2 UDB for z/OS Administration Guide.

If FM/DB2 detects an error, it marks the row that caused the error with either “=ERR “, “=ERR R”, or the SQLCODE in the prefix area. The choice of error indicator is determined by the Show SQLCODE editor option. Rows that were deleted are reinserted at the top of the current display (in TABL display format). Save operations cannot proceed until all pending errors have been corrected. If the DB2 object you are editing is subject to many constraints, or if many changes have been made to the data without verification, it might be impossible to resolve all errors. In this situation, cancel the Edit session.

Related tasks
**Handling errors during an Edit session**

- “View and Edit options (options 1 and 2)” on page 43
- “Setting options for the current FM/DB2 editor session” on page 95
- “Ending an FM/DB2 editor session” on page 101
- “Example of correcting an error during an Edit session” on page 160

**Related references**
- “Editor Options (3 of 8) panel” on page 535

**Determining why an error occurred**

You can find more information about the error detected against a row by typing `E` in the prefix field for that row. An error information panel is displayed with an explanation of why DB2 rejected the change operation.

If the SQLCODE is displayed in the prefix area, see the *DB2 UDB for z/OS Messages and Codes* for a detailed explanation of the error.

Some errors are the result of referential integrity (RI) constraint violations. If the RI relationship applies to two tables, these errors are indicated by the presence of an “R” in the rightmost position of the prefix area.

You can use the RE prefix command against rows with RI errors that apply to two tables to start another FM/DB2 Edit session. The new session is an Edit session for the other table affected by the RI error.

**Related tasks**
- “Editing related tables” on page 149
- “Example of correcting an error during an Edit session” on page 160

**Related references**
- “Table Edit panel” on page 691

**“Duplicate Row” errors**

FM/DB2 reports a “Duplicate Row Error” if the insert or update operation would have resulted in two rows with the same value with respect to a unique index defined on the table. You can use the Primary Key and Unique Index indicators to identify columns of the table that are part of unique indexes. Note that if you used row selection criteria to limit the amount of data in your Edit session, the duplicate row might not be displayed in your Edit session.

**Related tasks**
- “Selecting rows” on page 58

**“No Primary Key” errors**

FM/DB2 reports a “No Primary Key” error if your change would have created a foreign key value for which there is no corresponding primary key in the parent table of the relationship. This is a referential integrity (RI) constraint error. The prefix area error indicator only shows an “R” in the rightmost position if the parent table is different to the table currently being edited.

You can use the foreign key indicators to identify columns of the table that are defined as foreign keys.
Handling errors during an Edit session

Note: The foreign key information is only available if the “Retrieve foreign key information when building templates” option was selected when the template for the object was built.

The error information panel for this error shows the names of the parent and dependent tables in the relationship that caused the error, and the name of the relationship. At the bottom of the panel are the names of the columns that comprise the key in both the parent and dependent tables, along with the value that caused the error.

If the parent table is different to the dependent table, you can enter the RE prefix command against the row, or enter the REDIT primary command from the error information panel, to edit the parent table of the relationship and correct the error.

Related tasks

“Editing related tables” on page 149

Related references

“FM/DB2 Systems Options (1 of 4) panel” on page 577
“REDIT primary command” on page 770

“Delete Restrict Rule” errors

FM/DB2 reports a “Delete Restrict Rule” error if your deletion or change would have deleted one or more rows in dependent tables, but the delete rule is RESTRICT. These are referential integrity (RI) constraint errors.

The error information panel for this error shows the names of the parent and dependent tables in the relationship that caused the error, and the name of the relationship. At the bottom of the panel are the names of the columns that comprise the key in both the parent and dependent tables, along with the value that caused the error.

You can enter the RE prefix command against the row, or enter the REDIT command from the error information panel, to edit the dependent table of the relationship and correct the error.

Related tasks

“Editing related tables” on page 149

Related references

“REDIT primary command” on page 770

“Check Constraint” errors

FM/DB2 reports a “Check Constraint Error” if a change violates a check constraint defined on either a column or the table. The panel shows the name of the constraint that caused the change to fail. You can press the ShowCon function key (F5), or issue the SHOWCON command, to display an ISPF Edit session showing the SQL text of the constraint.

“Row Not Found” errors

FM/DB2 reports a “Row Not Found Error” if the original row was not found when either a DELETE or UPDATE operation was attempted. This can occur because another user has changed the row since the data was originally retrieved from DB2. The recommended action in this situation is to restart the Edit session.
Handling errors during an Edit session

with a current copy of the data. This can also occur if the DB2 object being edited has a TIME data type column and the system options are set to display the data for this column in USA format.

Related tasks

“FM/DB2 system options (option 0.2)” on page 41
“DATE and TIME columns” on page 131

Related references

“FM/DB2 Systems Options (1 of 4) panel” on page 577

Deadlocks or timeouts

DB2 might encounter a deadlock or timeout condition (SQLCODE -911) if it attempts to implement changes made in an FM/DB2 Edit session. If this occurs there is a delay, followed by the display of the SQL error pop-up panel with the explanation for the -911 SQLCODE.

If you encounter this error, be aware that:

- Changes made to any row marked with -911 have not been completed.
- DB2 has rolled back the current unit of work. This means that all uncommitted changes made in the current Edit session have been lost.
- After the rollback of the current unit of work, the data shown in the FM/DB2 Edit session might not reflect the current state of the data in the DB2 table.

The recommended course of action is to cancel the current Edit session and restart with a new copy of the data. If the problem persists contact your DB2 systems administrator for assistance in determining which other DB2 task is preventing access to the DB2 object.

Related tasks

“Ending an FM/DB2 editor session” on page 101

Example of correcting an error during an Edit session

If an error occurs during an Edit session, FM/DB2 displays “=ERR”, “=ERR R”, or the SQLCODE in the prefix area of the row (or rows) in error. Figure 47 on page 161 shows an error has occurred for one row.
To determine the type of error, type E in the prefix area for the row in error and press Enter. FM/DB2 displays an error information panel that describes the error and instructions about what you can do to correct the error.

You can then take one of the following actions:

- Cancel the Edit session by pressing the Cancel function key (F12). Any uncommitted changes are lost.
- Press Enter or the Exit function key (F3) to return to the Edit session.
- Use the REDIT primary command (if the error information panel offers it as an option) to edit the related table to correct the error.
Handling errors during an Edit session

Note: If the error information panel does not offer the REDIT primary command as an option, REDIT displays an Edit session.

In the example shown in Figure 47 on page 161, the error has been caused by two rows having the same index key (rows 4 and 5 both have an EMPNO of 000050). To correct the error:
1. Press Enter to return to the Edit session.
2. Change the offending index key for row 5 to a unique value such as 000060.
3. Press the Exit function key (F3) to return to the Edit session.

Related references
- "Table Edit panel" on page 691
- "REDIT primary command" on page 770
Chapter 5. Creating and dropping DB2 objects

You can use FM/DB2 to create and drop DB2 objects without needing to know the SQL commands for these operations.

To create or drop DB2 objects, use the Objects utility (3.2).

The main panel for the Objects utility is the DB2 Object Functions panel.

To display the DB2 Object Functions panel,
1. From the Primary Option Menu panel, enter 3 (Utilities) to display the Utility Functions panel.
2. From the Utility Functions panel, enter 2 (Objects)

Now that you have the DB2 Object Functions panel displayed, you can select the DB2 object type you want to create or drop:
1. Specify the relevant number in the **Object Type** entry field. When you first display the DB2 Object Functions panel, this field is initialized to 3 (Table).
2. If you intend to drop an object but would like FM/DB2 to display a Drop Confirmation panel before it actually drops the object you have specified, ensure the **Confirm object drop** option is checked.
3. To create the specified object type, type **C** on the command line.
   To drop the specified object type, type **D** on the command line.
4. Press Enter.

Table 9 shows the DB2 object types you can create or drop.

<table>
<thead>
<tr>
<th>Object Type</th>
<th>Create</th>
<th>Drop</th>
</tr>
</thead>
<tbody>
<tr>
<td>Database</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Table space</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Table</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>View</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Alias</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Index</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Synonym</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Distinct Type</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Function</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Stored Procedure</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Trigger</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Auxiliary Table</td>
<td>Yes</td>
<td>No</td>
</tr>
</tbody>
</table>

Table 9. Supported actions for each DB2 object.
Creating and dropping DB2 objects

Table 9. Supported actions for each DB2 object (continued).

<table>
<thead>
<tr>
<th>Object Type</th>
<th>Create</th>
<th>Drop</th>
</tr>
</thead>
</table>

Notes:
1. Use table drop. Populated auxiliary tables can only be dropped by dropping their associated base table. To drop an unpopulated auxiliary table without dropping its base table:
   - Use the drop table function (by selecting Object Type 3 and entering D on the command line on the DB2 Object Functions panel.
   - Specify the auxiliary table as the table to drop.

Related references
- “Primary Option Menu panel” on page 617
- “Utility Functions panel” on page 720
- “DB2 Object Functions panel” on page 510
- “Drop Confirmation panel” on page 525

Working with the Create and Drop panels

Here are a few general points relating to the usage of the Create and Drop panels:

You cannot use wildcard characters when using the Create or Drop panels.

You can type SQL on the command line of the main Create and Drop panels and the Drop Confirmation panel to examine the SQL that would be constructed given the current panel inputs.

If you receive a validation error when inputting data on the panels, a brief error message is displayed on the top right hand corner of the panel. To obtain a more detailed explanation of the error message, press the Help function key (F1). Also, if you receive an SQL error during the Create or Drop operation, you can press the Help function key (F1) to receive a more detailed explanation. Press the Help function key (F1) again to examine the SQL statement and the cause of the error.

Some of the Create panels such as the Create Table Space panel and the Create Function panel incorporate a number of subsidiary panels to collect the Create statement inputs. Generally, only a few of these panels are required, therefore you do not need to visit all of the lower-level panels before executing the Create statement. The inputs and the panels that are required are indicated on the main panel. Generally, you can visit and revisit the subsidiary panels in any order but exceptions are indicated on the main panel. If you receive an SQL error during the create operation, you need only return to the panels that are in error to correct the errors and then try the Create operation again. When inputting your data on the subsidiary panel, you can press Enter to validate your changes, the Exit function key (F3) to save your changes and return to the main panel, or the Cancel function key (F12) to return to the main panel without saving your changes.

Many of the subsidiary panels use ISPF tables to manage object information that can have more than one occurrence, such as the columns of a table or the parameters of a function. Each line in the table contains a set of values for an occurrence of the piece of information.

When editing the ISPF tables, you can repeat, delete, and insert lines when the CMD field is displayed using the following commands:
Creating and dropping DB2 objects

I               Inserts a new (blank) line at the current position.
I(n)            Inserts n new (blank) lines at the current position.
D               Deletes the line.
D(n)            Deletes n lines, starting at the current position.
R               Repeats the line.
R(n)            Repeats the line n times.

If the Cmd field is not displayed, you can only edit some fields in the existing lines.

SQL Keywords

Table 10 through to Table 21 on page 173 show, for each of the Create panels (and, where applicable, for each subsidiary panel), the SQL keyword (or variable) that relates to each entry field on that panel.

For example, Table 10 shows that, on the main Create Database panel, the optional entry field Storage group name relates to the SQL keyword STOGROUP. When FM/DB2 builds the CREATE DATABASE statement, if you have entered a name (storage-name) in the Storage group name entry field, FM/DB2 generates the storage group part of the CREATE DATABASE statement syntax as:

STOGROUP storage-name

Note:
1. In some cases, a keyword that is valid for the connected DB2 version is not displayed on any panel because, although the keyword is supported, it is not required to build the SQL statement. The create index TYPE parameter is an example of this.
2. Some keyword values are managed entirely by FM/DB2 and are displayed only as output fields on the panels, and are included here for reference. For example, for the PART keyword, FM/DB2 generates the value set for NUMPARTS. Fields that are displayed elsewhere as input fields are not redocumented if they reappear as output fields.
3. Where possible, fields are associated with SQL keywords (rather than the values associated with those keywords). Otherwise, they are associated with the value-names shown in the relevant SQL syntax diagram. Occasionally more detailed context is given to avoid ambiguity.

<table>
<thead>
<tr>
<th>Main/subsidiary panel</th>
<th>Field</th>
<th>SQL Keyword(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main</td>
<td>(See &quot;Create Database panel&quot; on page 432.)</td>
<td></td>
</tr>
<tr>
<td>(Database) Name</td>
<td>database-name</td>
<td>STOGROUP</td>
</tr>
<tr>
<td>Storage group name</td>
<td></td>
<td>BUFFERPOOL</td>
</tr>
<tr>
<td>Buffer pool for table spaces</td>
<td></td>
<td>INDEXBP</td>
</tr>
<tr>
<td>Database usage</td>
<td>AS WORKFILE</td>
<td></td>
</tr>
<tr>
<td>Database usage</td>
<td>AS TEMP</td>
<td></td>
</tr>
<tr>
<td>Data encoding</td>
<td>CCSID</td>
<td></td>
</tr>
<tr>
<td>Member name</td>
<td>FOR MEMBER</td>
<td></td>
</tr>
</tbody>
</table>

Chapter 5. Creating and dropping DB2 objects  165
## Creating and dropping DB2 objects

### Table 11. Create Table Space

<table>
<thead>
<tr>
<th>Main/subsidiary panel</th>
<th>Field</th>
<th>SQL Keyword(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main</td>
<td>(See “Create Table Space panel” on page 462.)</td>
<td></td>
</tr>
<tr>
<td>(Table space) Name</td>
<td>table-space-name</td>
<td></td>
</tr>
<tr>
<td>Database</td>
<td>IN</td>
<td></td>
</tr>
<tr>
<td>Buffer pool</td>
<td>BUFFERPOOL</td>
<td></td>
</tr>
<tr>
<td>1. Type</td>
<td>(See “Create Table Space: Type panel” on page 470.)</td>
<td></td>
</tr>
<tr>
<td>Table Space Type</td>
<td>LARGE</td>
<td></td>
</tr>
<tr>
<td>Table Space Type</td>
<td>LOB</td>
<td></td>
</tr>
<tr>
<td>Segment size</td>
<td>SEGSIZE</td>
<td></td>
</tr>
<tr>
<td>No. of partitions</td>
<td>NUMPARTS</td>
<td></td>
</tr>
<tr>
<td>Max. partition size</td>
<td>DSSIZE</td>
<td></td>
</tr>
<tr>
<td>2. Allocation</td>
<td>(See “Create Table Space: Allocation panel” on page 463.)</td>
<td></td>
</tr>
<tr>
<td>VCAT name</td>
<td>VCAT</td>
<td></td>
</tr>
<tr>
<td>Storage group name</td>
<td>STOGROUP</td>
<td></td>
</tr>
<tr>
<td>Primary space</td>
<td>PRIQTY</td>
<td></td>
</tr>
<tr>
<td>Secondary space</td>
<td>SECQTY</td>
<td></td>
</tr>
<tr>
<td>Free pages</td>
<td>FREEPAGE</td>
<td></td>
</tr>
<tr>
<td>Free space</td>
<td>PCTFREE</td>
<td></td>
</tr>
<tr>
<td>Maximum rows per page</td>
<td>MAXROWS</td>
<td></td>
</tr>
<tr>
<td>3. Data Storage Options</td>
<td>(See “Create Table Space: Data Storage Options panel” on page 466.)</td>
<td></td>
</tr>
<tr>
<td>Use data compression</td>
<td>COMPRESS</td>
<td></td>
</tr>
<tr>
<td>Close if not in use</td>
<td>CLOSE</td>
<td></td>
</tr>
<tr>
<td>Erase data on delete</td>
<td>ERASE</td>
<td></td>
</tr>
<tr>
<td>Define data set now</td>
<td>DEFINE</td>
<td></td>
</tr>
<tr>
<td>INSERT uses clustering index</td>
<td>MEMBER CLUSTER</td>
<td></td>
</tr>
<tr>
<td>Track modified changes</td>
<td>TRACKMOD</td>
<td></td>
</tr>
<tr>
<td>Log changes to LOB columns</td>
<td>LOG</td>
<td></td>
</tr>
<tr>
<td>Data encoding</td>
<td>CCSID</td>
<td></td>
</tr>
<tr>
<td>4. Locking Options</td>
<td>(See “Create Table Space: Locking Options panel” on page 469.)</td>
<td></td>
</tr>
<tr>
<td>Locksize parameter</td>
<td>LOCKSIZE</td>
<td></td>
</tr>
<tr>
<td>Application Locking</td>
<td>LOCKMAX</td>
<td></td>
</tr>
<tr>
<td>5. Data Sharing Options</td>
<td>(See “Create Table Space: Data Sharing Options panel” on page 465.)</td>
<td></td>
</tr>
<tr>
<td>Selective Partition Locking</td>
<td>LOCKPART</td>
<td></td>
</tr>
<tr>
<td>Group Buffer Pool Usage</td>
<td>GBPCACHE</td>
<td></td>
</tr>
<tr>
<td>6. Define Partitions</td>
<td>(See “Create Table Space: Define Partitions panel” on page 467.)</td>
<td></td>
</tr>
<tr>
<td>Pt. No.</td>
<td>PART</td>
<td></td>
</tr>
<tr>
<td>VCAT name</td>
<td>VCAT</td>
<td></td>
</tr>
<tr>
<td>Storage group name</td>
<td>STOGROUP</td>
<td></td>
</tr>
<tr>
<td>Primary space</td>
<td>PRIQTY</td>
<td></td>
</tr>
<tr>
<td>Secndry space</td>
<td>SECQTY</td>
<td></td>
</tr>
<tr>
<td>Erase Data</td>
<td>ERASE</td>
<td></td>
</tr>
</tbody>
</table>
### Table 11. Create Table Space (continued)

<table>
<thead>
<tr>
<th>Main/subsiduary panel</th>
<th>Field</th>
<th>SQL Keyword(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Free pages</td>
<td>FREEPAGE</td>
<td></td>
</tr>
<tr>
<td>Free space</td>
<td>PCTFREE</td>
<td></td>
</tr>
<tr>
<td>Use Data Cmpr</td>
<td>COMPRESS</td>
<td></td>
</tr>
<tr>
<td>Group BP Caching</td>
<td>GBPCACHE</td>
<td></td>
</tr>
</tbody>
</table>

### Table 12. Create Table

<table>
<thead>
<tr>
<th>Main/subsiduary panel</th>
<th>Field</th>
<th>SQL Keyword(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main</td>
<td>(See &quot;Create Table panel&quot; on page 459)</td>
<td></td>
</tr>
<tr>
<td>(New Table) Owner</td>
<td>table-name</td>
<td></td>
</tr>
<tr>
<td>(New Table) Name</td>
<td>table-name</td>
<td></td>
</tr>
<tr>
<td>(New Table) Database</td>
<td>IN</td>
<td></td>
</tr>
<tr>
<td>(New Table) Table Space</td>
<td>IN</td>
<td></td>
</tr>
<tr>
<td>(Model Table) Owner</td>
<td>LIKE</td>
<td></td>
</tr>
<tr>
<td>(Model Table) Name</td>
<td>LIKE</td>
<td></td>
</tr>
<tr>
<td>1. Columns</td>
<td>(See &quot;Create Table: Columns panel&quot; on page 474)</td>
<td></td>
</tr>
<tr>
<td>Column Name</td>
<td>column-name</td>
<td></td>
</tr>
<tr>
<td>Data Type</td>
<td>built-in-data-type</td>
<td></td>
</tr>
<tr>
<td>Len</td>
<td>built-in-data-type</td>
<td></td>
</tr>
<tr>
<td>Pr</td>
<td>built-in-data-type</td>
<td>(numeric only)</td>
</tr>
<tr>
<td>Sc</td>
<td>built-in-data-type</td>
<td>(decimal only)</td>
</tr>
<tr>
<td>For Data</td>
<td>FOR ... DATA</td>
<td></td>
</tr>
<tr>
<td>User Defined Data Type</td>
<td>distinct-type-name</td>
<td></td>
</tr>
<tr>
<td>2. Default Values</td>
<td>(See &quot;Create Table: Nulls and Default Values panel&quot; on page 479)</td>
<td></td>
</tr>
<tr>
<td>Not Null</td>
<td>NOT NULL</td>
<td></td>
</tr>
<tr>
<td>Default Value</td>
<td>WITH DEFAULT</td>
<td></td>
</tr>
<tr>
<td>3. Options</td>
<td>(See &quot;Create Table: Options panel&quot; on page 481)</td>
<td></td>
</tr>
<tr>
<td>Editproc</td>
<td>EDITPROC</td>
<td></td>
</tr>
<tr>
<td>Validproc</td>
<td>VALIDPROC</td>
<td></td>
</tr>
<tr>
<td>OBID value</td>
<td>OBID</td>
<td></td>
</tr>
<tr>
<td>Audit Options</td>
<td>AUDIT</td>
<td></td>
</tr>
<tr>
<td>Data Storage Options</td>
<td>CCSID</td>
<td></td>
</tr>
<tr>
<td>Log data capture changes</td>
<td>DATA CAPTURE</td>
<td></td>
</tr>
<tr>
<td>Restrict drop of table</td>
<td>WITH RESTRICT ON DROP</td>
<td></td>
</tr>
<tr>
<td>4. Primary Key</td>
<td>(See &quot;Create Table: Unique Constraints panel&quot; on page 487)</td>
<td></td>
</tr>
<tr>
<td>Constraint Name</td>
<td>CONSTRAINT constraint-name PRIMARY KEY</td>
<td></td>
</tr>
<tr>
<td>Order</td>
<td>PRIMARY KEY(column-name ...)</td>
<td></td>
</tr>
<tr>
<td>5. Foreign Key</td>
<td>(See &quot;Create Table: Column Referential Constraints panel&quot; on page 473)</td>
<td></td>
</tr>
<tr>
<td>Constrnt Name</td>
<td>FOREIGN KEY</td>
<td></td>
</tr>
<tr>
<td>Owner</td>
<td>REFERENCES</td>
<td></td>
</tr>
</tbody>
</table>
# Creating and dropping DB2 objects

## Table 12. Create Table (continued)

<table>
<thead>
<tr>
<th>Main/subsiduary panel</th>
<th>Field</th>
<th>SQL Keyword(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table Name</td>
<td>REFERENCES</td>
<td></td>
</tr>
<tr>
<td>Column Name</td>
<td>REFERENCES</td>
<td></td>
</tr>
<tr>
<td>ON DLT</td>
<td>ON DELETE</td>
<td></td>
</tr>
</tbody>
</table>

6. Check Constraints  
(See “Create Table: Column Check Constraints panel” on page 471.)

<table>
<thead>
<tr>
<th>Field</th>
<th>SQL Keyword(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table Constraint</td>
<td>CONSTRAINT constraint-name CHECK</td>
</tr>
<tr>
<td>Check Condition</td>
<td>CHECK(check-condition ... )</td>
</tr>
<tr>
<td>(Constraint) Name</td>
<td>CONSTRAINT constraint-name CHECK</td>
</tr>
<tr>
<td>(Constraint) Condition</td>
<td>CHECK(check-condition ... )</td>
</tr>
</tbody>
</table>

7. Procedure Exits  
(See “Create Table: Procedure Exits panel” on page 483.)

<table>
<thead>
<tr>
<th>Field</th>
<th>SQL Keyword(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>FIELDPROC</td>
</tr>
<tr>
<td>Parameters</td>
<td>FIELDPROC</td>
</tr>
</tbody>
</table>

8. Generate Values  
(See “Create Table: Generate Values panel” on page 476.)

<table>
<thead>
<tr>
<th>Field</th>
<th>SQL Keyword(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gen (A/D)</td>
<td>GENERATE</td>
</tr>
<tr>
<td>As Idty (Y/N)</td>
<td>AS IDENTITY</td>
</tr>
<tr>
<td>Start value</td>
<td>START WITH</td>
</tr>
<tr>
<td>Incr value</td>
<td>INCREMENT BY</td>
</tr>
<tr>
<td>Caching Option</td>
<td>(NO) CACHE</td>
</tr>
</tbody>
</table>

## Table 13. Create View

<table>
<thead>
<tr>
<th>Main/subsiduary panel</th>
<th>Field</th>
<th>SQL Keyword(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main</td>
<td>(See “Create View panel” on page 493.)</td>
<td></td>
</tr>
<tr>
<td>Owner</td>
<td>view-name</td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td>view-name</td>
<td></td>
</tr>
<tr>
<td>Select statement</td>
<td>AS</td>
<td></td>
</tr>
<tr>
<td>Column names</td>
<td>column-name ...</td>
<td></td>
</tr>
<tr>
<td>With Check Option</td>
<td>WITH ... CHECK OPTION</td>
<td></td>
</tr>
</tbody>
</table>

## Table 14. Create Alias

<table>
<thead>
<tr>
<th>Main/subsiduary panel</th>
<th>Field</th>
<th>SQL Keyword(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main</td>
<td>(See “Create Alias panel” on page 430.)</td>
<td></td>
</tr>
<tr>
<td>(Alias) Owner</td>
<td>alias-name</td>
<td></td>
</tr>
<tr>
<td>(Alias) Name</td>
<td>alias-name</td>
<td></td>
</tr>
<tr>
<td>(Table or View) Location</td>
<td>FOR</td>
<td></td>
</tr>
<tr>
<td>(Table or View) Owner</td>
<td>FOR</td>
<td></td>
</tr>
<tr>
<td>(Table or View) Name</td>
<td>FOR</td>
<td></td>
</tr>
</tbody>
</table>

## Table 15. Create Index

<table>
<thead>
<tr>
<th>Main/subsiduary panel</th>
<th>Field</th>
<th>SQL Keyword(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main</td>
<td>(See “Create Index panel” on page 442.)</td>
<td></td>
</tr>
<tr>
<td>Owner</td>
<td>index-name</td>
<td></td>
</tr>
</tbody>
</table>
### Table 15. Create Index (continued)

<table>
<thead>
<tr>
<th>Main/subsidary panel</th>
<th>Field</th>
<th>SQL Keyword(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>index-name</td>
<td></td>
</tr>
<tr>
<td>Table Owner</td>
<td>ON</td>
<td></td>
</tr>
<tr>
<td>Table Name</td>
<td>ON</td>
<td></td>
</tr>
<tr>
<td>Buffer Pool</td>
<td>BUFFERPOOL</td>
<td></td>
</tr>
</tbody>
</table>

1. **Index Type** (See “Create Index: Type panel” on page 451.)
   - Uniqueness: UNIQUE WHERE NOT NULL
   - Internal format: TYPE
   - Clustering Index Option: CLUSTER
   - Piecesize: PIECESIZE
   - Multiplier: PIECESIZE

2. **Column Selection** (See “Create Index: Column Selection panel” on page 446.)
   - Order: column-name
   - (A/D): column-name

3. **Allocation** (See “Create Index: Allocation panel” on page 444.)
   - VCAT name: VCAT
   - Storage group name: STOGROUP
   - Primary space: PRIQTY
   - Secondary space: SECQTY
   - Free pages: FREEPAGE
   - Free space: PCTFREE

4. **Options** (See “Create Index: Options panel” on page 447.)
   - Close if not in use: CLOSE
   - Erase data on delete: ERASE
   - Defer building index: DEFER
   - Define data set now: DEFINE
   - Allow COPY of index: COPY
   - Group Buffer Pool Usage: GBPCACHE

5. **Partitions** (See “Create Index: Partitions panel” on page 448.)
   - Parttn. Number: PART
   - VCAT name: VCAT
   - Storage group name: STOGROUP
   - Primary space: PRIQTY
   - Secndry space: SECQTY
   - Erase Data: ERASE
   - Free pages: FREEPAGE
   - Free space: PCTFREE
   - Group BP Caching: GBPCACHE

6. **Partition Values** (See “Create Index: Partition Values panel” on page 450.)
   - Value: VALUES
## Creating and dropping DB2 objects

### Table 16. Create Synonym

<table>
<thead>
<tr>
<th>Main/subsidiary panel</th>
<th>Field</th>
<th>SQL Keyword(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main</td>
<td>(See “Create Synonym panel” on page 458.)</td>
<td></td>
</tr>
<tr>
<td>(Synonym) Name</td>
<td>synonym</td>
<td></td>
</tr>
<tr>
<td>(Table) Owner</td>
<td>FOR</td>
<td></td>
</tr>
<tr>
<td>(Table) Name</td>
<td>FOR</td>
<td></td>
</tr>
</tbody>
</table>

### Table 17. Create Distinct Type

<table>
<thead>
<tr>
<th>Main/subsidiary panel</th>
<th>Field</th>
<th>SQL Keyword(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main</td>
<td>(See “Create Distinct Type panel” on page 433.)</td>
<td></td>
</tr>
<tr>
<td>Schema</td>
<td>distinct-type-name</td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td>distinct-type-name</td>
<td></td>
</tr>
<tr>
<td>Source Type</td>
<td>AS</td>
<td></td>
</tr>
<tr>
<td>Length</td>
<td>AS</td>
<td></td>
</tr>
<tr>
<td>Scale</td>
<td>AS</td>
<td></td>
</tr>
<tr>
<td>For Data</td>
<td>FOR ... DATA</td>
<td></td>
</tr>
<tr>
<td>Encoding Method</td>
<td>CCSID</td>
<td></td>
</tr>
</tbody>
</table>

### Table 18. Create Function

<table>
<thead>
<tr>
<th>Main/subsidiary panel</th>
<th>Field</th>
<th>SQL Keyword(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main</td>
<td>(See “Create Function panel” on page 437.)</td>
<td></td>
</tr>
<tr>
<td>Schema</td>
<td>function-name</td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td>function-name</td>
<td></td>
</tr>
<tr>
<td>Unique Name</td>
<td>SPECIFIC</td>
<td></td>
</tr>
<tr>
<td>1. Parameters</td>
<td>(See “Create Function: Parameters panel” on page 439.)</td>
<td></td>
</tr>
<tr>
<td>Parameter Name</td>
<td>parameter-name</td>
<td></td>
</tr>
<tr>
<td>Data Type</td>
<td>built-in-data-type</td>
<td></td>
</tr>
<tr>
<td>Len</td>
<td>built-in-data-type</td>
<td></td>
</tr>
<tr>
<td>Pr</td>
<td>built-in-data-type</td>
<td></td>
</tr>
<tr>
<td>Sc</td>
<td>built-in-data-type</td>
<td></td>
</tr>
<tr>
<td>For Data</td>
<td>FOR ... DATA</td>
<td></td>
</tr>
<tr>
<td>User Defined Data Type</td>
<td>distinct-type-name</td>
<td></td>
</tr>
<tr>
<td>2. Parameter Types</td>
<td>(See “Create Function: Parameter Types panel” on page 438.)</td>
<td></td>
</tr>
<tr>
<td>As LOC</td>
<td>AS LOCATOR</td>
<td></td>
</tr>
<tr>
<td>Owner</td>
<td>TABLE LIKE</td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td>TABLE LIKE</td>
<td></td>
</tr>
<tr>
<td>3. Returned Data Type</td>
<td>(See “Create Function: Returned Data Type panel” on page 441.)</td>
<td></td>
</tr>
<tr>
<td>(Returns) Data Type</td>
<td>RETURNS</td>
<td></td>
</tr>
<tr>
<td>(Returns) Schema</td>
<td>RETURNS</td>
<td></td>
</tr>
<tr>
<td>(Returns) Length</td>
<td>RETURNS</td>
<td></td>
</tr>
<tr>
<td>(Returns) Scale</td>
<td>RETURNS</td>
<td></td>
</tr>
<tr>
<td>(Returns) For Data</td>
<td>RETURNS ... FOR ... DATA</td>
<td></td>
</tr>
</tbody>
</table>
### Table 18. Create Function (continued)

<table>
<thead>
<tr>
<th>Main/subsidiary panel</th>
<th>Field</th>
<th>SQL Keyword(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Returns) Encoding Method</td>
<td>RETURNS ... CCSID</td>
<td></td>
</tr>
<tr>
<td>(Cast From) Data Type</td>
<td>CAST FROM</td>
<td></td>
</tr>
<tr>
<td>(Cast From) Length</td>
<td>CAST FROM</td>
<td></td>
</tr>
<tr>
<td>(Cast From) Scale</td>
<td>CAST FROM</td>
<td></td>
</tr>
<tr>
<td>(Cast From) For Data</td>
<td>CAST FROM ... FOR ... DATA</td>
<td></td>
</tr>
<tr>
<td>(Cast From) Encoding Method</td>
<td>CAST FROM ... CCSID</td>
<td></td>
</tr>
<tr>
<td>As Locator</td>
<td>RETURNS ... AS LOCATOR</td>
<td></td>
</tr>
</tbody>
</table>

4. Option List (1/2)  
(See “Create Function: Option List (1/2) panel” on page 435.)

<table>
<thead>
<tr>
<th>Field</th>
<th>SQL Keyword(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>External Name</td>
<td>EXTERNAL NAME</td>
</tr>
<tr>
<td>Scratchpad</td>
<td>(NO) SCRATCHPAD</td>
</tr>
<tr>
<td>Package Collection</td>
<td>(NO) COLLID</td>
</tr>
<tr>
<td>Workload Manager Environment</td>
<td>WLM ENVIRONMENT</td>
</tr>
<tr>
<td>Time Limit</td>
<td>ASUTIME (NO) LIMIT</td>
</tr>
<tr>
<td>Language</td>
<td>LANGUAGE</td>
</tr>
<tr>
<td>SQL</td>
<td>(READS)(NO)(MODIFIES) (CONTAINS) SQL (DATA)</td>
</tr>
<tr>
<td>Security</td>
<td>SECURITY</td>
</tr>
<tr>
<td>Parallel</td>
<td>(DIS)ALLOW PARALLEL</td>
</tr>
<tr>
<td>Run-time Options</td>
<td>RUN OPTIONS</td>
</tr>
</tbody>
</table>

5. Option List (2/2)  
(See “Create Function: Option List (2/2) panel” on page 436.)

<table>
<thead>
<tr>
<th>Field</th>
<th>SQL Keyword(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main Program</td>
<td>PROGRAM TYPE</td>
</tr>
<tr>
<td>Stay Resident</td>
<td>STAY RESIDENT</td>
</tr>
<tr>
<td>Deterministic</td>
<td>(NOT) DETERMINISTIC</td>
</tr>
<tr>
<td>Called on null input</td>
<td>(RETURNS NULL)(CALLED) ON NULL INPUT</td>
</tr>
<tr>
<td>No external action</td>
<td>(NO) EXTERNAL ACTION</td>
</tr>
<tr>
<td>Final Call</td>
<td>(NO) FINAL CALL</td>
</tr>
<tr>
<td>DBINFO argument passed</td>
<td>(NO) DBINFO</td>
</tr>
</tbody>
</table>

### Table 19. Create Procedure

<table>
<thead>
<tr>
<th>Main/subsidiary panel</th>
<th>Field</th>
<th>SQL Keyword(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(See “Create Procedure panel” on page 452.)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Schema</td>
<td>procedure-name</td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td>procedure-name</td>
<td></td>
</tr>
<tr>
<td>Result Sets</td>
<td>DYNAMIC RESULT SET(S)</td>
<td></td>
</tr>
</tbody>
</table>

1. Parameters  
(See “Create Procedure: Parameters panel” on page 457.)

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>SQL Keyword(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Type</td>
<td>built-in-data-type</td>
</tr>
<tr>
<td>Len</td>
<td>built-in-data-type</td>
</tr>
<tr>
<td>Pr</td>
<td>built-in-data-type</td>
</tr>
<tr>
<td>Sc</td>
<td>built-in-data-type</td>
</tr>
</tbody>
</table>
### Creating and dropping DB2 objects

**Table 19. Create Procedure (continued)**

<table>
<thead>
<tr>
<th>Main/subsiduary panel</th>
<th>Field</th>
<th>SQL Keyword(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>For Data</td>
<td>FOR ... DATA</td>
<td></td>
</tr>
<tr>
<td>User Defined Data Type</td>
<td>distinct-type-name</td>
<td></td>
</tr>
</tbody>
</table>

2. Parameter Types

<table>
<thead>
<tr>
<th>Field</th>
<th>SQL Keyword(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>As LOC</td>
<td>AS LOCATOR</td>
</tr>
<tr>
<td>Owner</td>
<td>TABLE LIKE</td>
</tr>
<tr>
<td>Name</td>
<td>TABLE LIKE</td>
</tr>
<tr>
<td>(IN)(OUT)(INOUT)</td>
<td>(IN)(OUT)(INOUT)</td>
</tr>
</tbody>
</table>

4. Option List (1/2)

<table>
<thead>
<tr>
<th>Field</th>
<th>SQL Keyword(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>External Name</td>
<td>EXTERNAL NAME</td>
</tr>
<tr>
<td>Package Collection</td>
<td>(NO) COLLID</td>
</tr>
<tr>
<td>Workload Manager Environment</td>
<td>(NO) WLM ENVIRONMENT</td>
</tr>
<tr>
<td>Time Limit</td>
<td>ASUTIME (NO) LIMIT</td>
</tr>
<tr>
<td>Language</td>
<td>LANGUAGE</td>
</tr>
<tr>
<td>SQL</td>
<td>(READS)(NO)(MODIFIES) (CONTAINS) SQL (DATA)</td>
</tr>
<tr>
<td>Security</td>
<td>SECURITY</td>
</tr>
<tr>
<td>Parameter Style</td>
<td>PARAMETER STYLE</td>
</tr>
<tr>
<td>Run-time Options</td>
<td>RUN OPTIONS</td>
</tr>
</tbody>
</table>

5. Option List (2/2)

<table>
<thead>
<tr>
<th>Field</th>
<th>SQL Keyword(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main Program</td>
<td>PROGRAM TYPE</td>
</tr>
<tr>
<td>Stay Resident</td>
<td>STAY RESIDENT</td>
</tr>
<tr>
<td>Deterministic</td>
<td>(NOT) DETERMINISTIC</td>
</tr>
<tr>
<td>DBINFO argument passed</td>
<td>(NO) DBINFO</td>
</tr>
<tr>
<td>Commit on return</td>
<td>COMMIT ON RETURN</td>
</tr>
</tbody>
</table>

**Table 20. Create Trigger**

<table>
<thead>
<tr>
<th>Main/subsiduary panel</th>
<th>Field</th>
<th>SQL Keyword(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main</td>
<td>(See “Create Trigger panel” on page 488.)</td>
<td></td>
</tr>
<tr>
<td>Schema</td>
<td>trigger-name</td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td>trigger-name</td>
<td></td>
</tr>
</tbody>
</table>

1. Type

<table>
<thead>
<tr>
<th>Field</th>
<th>SQL Keyword(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Order) No cascade before</td>
<td>NO CASCADE BEFORE</td>
</tr>
<tr>
<td>(Order) After</td>
<td>AFTER</td>
</tr>
<tr>
<td>(Type) Insert</td>
<td>INSERT</td>
</tr>
<tr>
<td>(Type) Delete</td>
<td>DELETE</td>
</tr>
<tr>
<td>(Type) Update</td>
<td>UPDATE</td>
</tr>
<tr>
<td>Columns</td>
<td>UPDATE OF</td>
</tr>
<tr>
<td>(Trigger Table) Name</td>
<td>ON</td>
</tr>
<tr>
<td>(Trigger Table) Owner</td>
<td>ON</td>
</tr>
<tr>
<td>Old correlation name</td>
<td>OLD</td>
</tr>
</tbody>
</table>
Creating and dropping DB2 objects

Table 20. Create Trigger (continued)

<table>
<thead>
<tr>
<th>Main/subsiduary panel</th>
<th>Field</th>
<th>SQL Keyword(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>New correlation name</td>
<td>NEW</td>
</tr>
<tr>
<td></td>
<td>Old table identifier</td>
<td>OLD TABLE</td>
</tr>
<tr>
<td></td>
<td>New table identifier</td>
<td>NEW TABLE</td>
</tr>
<tr>
<td></td>
<td>(Executed) For each modified row</td>
<td>FOR EACH ROW</td>
</tr>
<tr>
<td></td>
<td>(Executed) Once</td>
<td>FOR EACH STATEMENT</td>
</tr>
</tbody>
</table>

2. Search Condition
(See “Create Trigger: Search Condition panel” on page 491.)

<table>
<thead>
<tr>
<th>Field</th>
<th>SQL Keyword(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Search Condition</td>
<td>WHEN</td>
</tr>
</tbody>
</table>

3. SQL Statement
(See “Create Trigger: SQL statement panel” on page 492.)

<table>
<thead>
<tr>
<th>Field</th>
<th>SQL Keyword(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Triggered SQL Statement(s)</td>
<td>BEGIN ATOMIC</td>
</tr>
</tbody>
</table>

Table 21. Create Auxiliary Table

<table>
<thead>
<tr>
<th>Main/subsiduary panel</th>
<th>Field</th>
<th>SQL Keyword(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main</td>
<td>(Auxiliary Table) Owner</td>
<td>aux-table-name</td>
</tr>
<tr>
<td></td>
<td>(Auxiliary Table) Name</td>
<td>aux-table-name</td>
</tr>
<tr>
<td></td>
<td>(Table Space) Name</td>
<td>IN</td>
</tr>
<tr>
<td></td>
<td>Database</td>
<td>IN</td>
</tr>
<tr>
<td></td>
<td>(Base Table) Owner</td>
<td>STORES</td>
</tr>
<tr>
<td></td>
<td>(Base Table) Name</td>
<td>STORES</td>
</tr>
<tr>
<td></td>
<td>(Base Table Column) Name</td>
<td>COLUMN</td>
</tr>
<tr>
<td></td>
<td>Partition</td>
<td>PART</td>
</tr>
</tbody>
</table>

Creating a database

To create a new database, perform the following on the DB2 Object Functions panel:
1. Type 1 in the Object Type entry field.
2. Type C on the command line.
3. Press Enter. FM/DB2 displays the Create Database panel.

The Create Database panel defines a DB2 database at the current server.

The only required field on the panel is the first, where you enter the name for the new database. All of the other fields on the panel are optional, although values are often specified for the second group of input fields.

The database options and member selection sections of the panel contain input fields that you can use to define a work file or temporary database.

The default data encoding scheme for z/OS systems is EBCDIC. You can specify ASCII encoding if required.

For workfile databases, you can also specify a member name (for use in a data sharing environment only).

Related references
Creating a table space

To create a new table space, perform the following on the DB2 Object Functions panel:
1. Type 2 in the **Object Type** entry field.
2. Type C on the command line.
3. Press Enter. FM/DB2 displays the Create Table Space panel.

Use the Create Table Space panel to define a simple, segmented or partitioned table space at the current server.

Related tasks
- "Table Space Type (Create Table Space)"
- "Table Space Allocation (Create Table Space)"
- "Data Storage (Create Table Space)" on page 175
- "Locking (Create Table Space)" on page 175
- "Data Sharing (Create Table Space)" on page 175
- "Partitions (Create Table Space)" on page 175

Related references
- "DB2 Object Functions panel" on page 510
- "Create Table Space panel" on page 462
- "Create Table Space: Type panel" on page 470
- "Create Table Space: Allocation panel" on page 463
- "Create Table Space: Data Storage Options panel" on page 466
- "Create Table Space: Locking Options panel" on page 469
- "Create Table Space: Data Sharing Options panel" on page 465
- "Create Table Space: Define Partitions panel" on page 467

**Table Space Type (Create Table Space)**

In DB2, table space is either non-partitioned, or partitioned. A non-partitioned table space can be segmented (a partitioned table space cannot). Non-partitioned means that there is a 1:1 relationship between the table space and the data set defined to store the data. Partition means that many data sets are used to store the data.

To make more effective use of non-partitioned table spaces, by improving the support for multiple table definitions within the one table space, use segmented table spaces.

The maximum partition size for a partitioned table space determines the amount of data that can be stored within the table space. Large values imply the creation of multi-volume table spaces capable of storing GB of data.

Related references
- "Create Table Space: Type panel" on page 470

**Table Space Allocation (Create Table Space)**

The Create Table Space: Allocation panel is optional, although most users need to specify a DB2 storage group and the size of the table space.

You have the option of defining DB2 data sets externally, or DB2 can create the data sets on behalf of you. The former option is selected by specifying a VCAT
name in the data set parameters section of the panel. The most common option is
to specify a (DB2) storage group name. These parameters are mutually exclusive.

Related references
"Create Table Space: Allocation panel” on page 463

Data Storage (Create Table Space)
Use the Create Table Space: Data Sharing Options panel to specify options that
control various aspects of data management for the function.

Related references
"Create Table Space: Data Sharing Options panel” on page 465

Locking (Create Table Space)
The locksize parameter specifies the size of locks within the table space.

You can select from six different locking parameters. Page and Row cannot be
specified for LOB table spaces. Table can only be specified for a segmented table
space.

You can also specify a maximum value for the number of locks that application
programs can hold on a table space before lock escalation occurs.

Related references
"Create Table Space: Locking Options panel” on page 469

Data Sharing (Create Table Space)
The Create Table Space: Data Sharing Options panel is optional and contains
options that control how the table space behaves in a data sharing environment. In
most cases, the default values are appropriate.

Related references
"Create Table Space: Data Sharing Options panel” on page 465

Partitions (Create Table Space)
The Create Table Space: Define Partitions panel is optional, even for the definition
of a partitioned table space. However, you might want to specify parameters for
each partition of a partitioned table space, rather than relying on the values set at
the table space level.

For a partitioned table space, there is an optional 'USING CLAUSE' for every
partition of the table space. You can define the number of partitions on the Table
Space Type panel before selecting this panel. The table at the bottom of the panel is
pre-formatted with the required number of rows (one for each partition).

Alternatively, you can select the Define Partitions panels directly. The number of
pre-formatted rows reflect the value specified for a number of partitions (the
default value is 1). You can then adjust the values for this single partition to what
is required. After this is done, you can use the R prefix command (optionally
followed by an integer) to expand the number of partitions to the required number.
The value specified for number of partitions on the Table space Type panel is
updated accordingly.
Creating and dropping DB2 objects

Note: The individual partitions of a partitioned table space are typically defined with similar parameters. It is easier for you to define the values once, then repeat them, rather than type the same information in for each row repeatedly.

The DB2 default values for Free Pages, Percent Free, Data Compression and GBPCACHE are used.

Related references
- Create Table Space: Define Partitions panel” on page 47
- Create Table Space: Type panel” on page 47

Creating a table

To create a new table, perform the following on the DB2 Object Functions panel:
1. Type 3 in the Object Type entry field.
2. Type C on the command line.
3. Press Enter. FM/DB2 displays the Create Table panel.

Use the Create Table utility to create a new DB2 table. If a table already exists that is similar to the one you want to create, you can use that table as a model for the new table.

Related tasks
- “1. Columns (Create Table)” on page 180
- “2. Nulls and default values (Create Table)” on page 180
- “3. Options (Create Table)” on page 180
- “4. Unique constraints (Create Table)” on page 180
- “5. Referential constraints (Create Table)” on page 181
- “6. Check constraints (Create Table)” on page 182
- “7. Procedure exits (Create Table)” on page 183
- “8. Generate values (Create Table)” on page 183

Related references
- “DB2 Object Functions panel” on page 510
- “Create Table panel” on page 459
- “Create Table: Columns panel” on page 474
- “Create Table: Nulls and Default Values panel” on page 479
- “Create Table: Options panel” on page 481
- “Create Table: Unique Constraints panel” on page 487
- “Create Table: Column Referential Constraints panel” on page 473
- “Create Table: Column Check Constraints panel” on page 471
- “Create Table: Procedure Exits panel” on page 483
- “Create Table: Generate Values panel” on page 476

Creating a table using a model

The Create Table utility lets you use an existing table or view as a model for the new table you want to create. You can use this modeling facility to create a table in either of these ways:
- With columns that have exactly the same name and description as the model table or view: Usage option 1.

Note: If you use Usage option 1, none of the Table Creation options related to columns are available (DB2 copies the column information about the model table...
or view when it issues the CREATE TABLE statement). However, you can still specify non-column details on the main panel and by selecting Table Creation option 3 (Table options).

- Based on the model table or view but where you want to change some of the column details: Usage option 2, Table Creation options 1–8.

To create a table with columns that have exactly the same name and description as the model table or view:

1. Specify the name of the table you want to create in New Table.
2. Specify the name of the model table or view in Model Table/View. You can optionally qualify the name of the model table or view by specifying an owner. If you do not specify a model table (or view) owner, FM/DB2 uses your current SQL ID.
3. Select Usage option 1 (Generate LIKE clause).
4. Select the Table Creation option Create the table.
5. Press Enter.

FM/DB2 generates the following SQL statement:

```
CREATE TABLE new_table_name LIKE model_table_name
```

before sending it to DB2 for execution. FM/DB2 issues a message to confirm that the generated SQL statement has been run. For further details about the LIKE clause in the CREATE TABLE statement, see the DB2 UDB for z/OS SQL Reference.

**Note:** To view the generated SQL statement, either before or after it is issued, enter SQL on the command line.

To create a table based on the model table or view, but where you want to change some of the column details:

1. Specify the name of the DB2 table you want to create in New Table.
2. Specify the name of the model DB2 table or view in Model Table/View. You can optionally qualify the name of the model table or view by specifying an owner. If you do not specify a model table (or view) owner, FM/DB2 uses your current SQL ID.
3. Select Usage option 2 (Load table information).
4. Press Enter.

FM/DB2 displays the Create Table Model Load panel.

**Note:** The Create Table Model Load panel is displayed if you:
- Initially specify, or subsequently change, the Model Table/View details,
- Select Usage option 2 (Load table information), and
- Press Enter.

Use the Create Table Model Load panel to selectively load information describing the model table or view from the DB2 catalog into the ISPF variables and tables used to generate the CREATE TABLE statement. You can also use this panel to specify whether you want this information to replace the current information, or to be added to it.

5. Select the Model Load Options you want by typing a slash (/) next to the required option. (FM/DB2 selects the first three options for you.) To deselect a model load option, type a blank in place of the slash.
Creating and dropping DB2 objects

**Note:** If no data exists in the model table or view for any of the model load options you select, FM/DB2 issues the message “Some data not loaded” when it redisplays the main panel.

When you return to the main Create Table panel, you can change or add to this information by selecting the appropriate table creation option 1–8. For details, see [“Table Creation options” on page 179](#).

**Note:** If you use a view as the model, FM/DB2 loads only the database name, table space name, encoding scheme, and column information.

6. To specify whether you want FM/DB2 to use the information about the model table or view as a refresh, or if you want FM/DB2 to add it to the current information for the table you are creating, select the appropriate **Refresh/Add** option.

7. If the model table has referential constraints, specify whether you want single-column referential constraints loaded as **column** referential constraints or **table** referential constraints by selecting one of the following options:
   - 1. Column referential constraints
   - 2. Table referential constraints

**Note:**

   a. FM/DB2 does not check for duplicate column names if multiple model tables (or views) are loaded or model table (or view) columns are added to manually-defined columns.
   b. If you select any of the key or constraint options (Primary key information, Unique key information, Referential constraints or Check constraints), the Column information option is selected automatically.
   c. FM/DB2 loads all check constraint information in the form of **table** check constraints.

8. Press Enter.

   FM/DB2 loads information into the ISPF variables and tables used to generate the CREATE TABLE statement according to the Model Load Options and Refresh/Add options you have selected and returns you to the main Create Table panel.

   **Note:** To view the generated SQL statement, type `SQL` on the command line of the main Create Table panel.

9. To specify further information for the table you want to create, select any of the table creation options 1–8 and press Enter.

   FM/DB2 validates your input and displays a panel that you use to enter information for the table creation option you have selected. For details about each table creation option, see [“Table Creation options” on page 179](#).

10. Select the Table Creation option **Create the table** on the main panel.

11. Press Enter.

   FM/DB2 generates the following SQL statement:

   ```sql
   CREATE TABLE new_table_name other_details
   ```

   where `other_details` represents SQL clauses based on the details you have selected using the model table or view and Table Creation options. FM/DB2 sends the generated SQL statement to DB2 and issues a message to confirm that it has been successfully run.

   **Note:** To view the generated SQL statement, either before or after it is issued, enter `SQL` on the command line.
Creating a table without using a model

To create a new table without using a model (for instance, if no table already exists that is similar to the one you want to create):

1. Specify the name of the DB2 table you want to create in **New Table**.
2. Leave the owner and name of the model DB2 table in **Model Table** blank.
3. Select any of the Table Creation options 1–8 and press Enter. The default setting is Option 1 (**Columns**).

   **Note:** Before you can specify any of the table creation options 2, 4, 5, 6, 7, or 8, you must create at least some of the columns for the new table (using option 1).

FM/DB2 validates your input and displays a panel that you use to enter information for the table creation option you have selected.

For information about each table creation option, see “Table Creation options.”

4. Select the Table Creation option **Create the table**
5. Press Enter.

FM/DB2 generates the following SQL statement:

```
CREATE TABLE new_table_name other_details
```

where **other_details** represents SQL clauses based on the details you have selected using the Table Creation options. FM/DB2 sends the generated SQL statement to DB2 and issues a message to confirm that it has been successfully run.

**Note:** To view the generated SQL statement, either before or after it is issued, enter SQL on the command line.

Table Creation options

Use the table creation options on the main Create Table panel to provide additional information to define the table you are creating.

If you are using a model table with Usage option 2, you can select any of the options 1–8 before using the Table Creation option **Create the table** to create the table. If you have already used the Model Load panel to provide information for the table you want to create, use the table creation options to supplement or change details you have already selected.

If you are not using a model table, you must select option 1 (**Columns**) and then optionally options 2–8, before using the Table Creation option **Create the table** to create the table.

You can revisit each of the table creation options as many times as you want to add or modify the data.

If you select one of the table creation options 1 to 8, FM/DB2 displays a panel that you use to enter information for the creation option you have selected. If you select the Table Creation option **Create the table**, FM/DB2 generates and issues a CREATE TABLE SQL statement.
Creating and dropping DB2 objects

The following pages describe the panels for each creation option.

Related references

"Create Table panel" on page 459
"Create Table: Model Load panel" on page 477

1. Columns (Create Table)

The Create Table Columns panel is displayed if you select Table Creation option 1 (Columns) on the main Create Table panel. You use this panel to define columns for the table you are creating, or to change information for columns you have already defined.

The lower section of the panel shows the columns defined for the table. (If you have not yet defined any columns, this part of the panel shows a single line of empty entry fields.) If you have used the modeling facility, the panel shows the column details for each column in the model table.

Related references

"Create Table: Columns panel" on page 474
"Create Table panel" on page 459

2. Nulls and default values (Create Table)

The Nulls and Default Values panel is displayed if you select Table Creation option 2 (Nulls/default values) on the main Create Table panel.

The Nulls and Default Values panel shows the columns defined for the table. If you have used the modeling facility, the panel shows the column details for each column in the model table. You use this panel to specify information to define the null attributes and default values for the columns of the table you are creating.

Related references

"Create Table: Nulls and Default Values panel" on page 479
"Create Table panel" on page 459

3. Options (Create Table)

The Options panel is displayed if you select Table Creation option 3 (Table options) on the main Create Table panel.

You use the Options panel to specify information to define options for the table you are creating.

Related references

"Create Table: Options panel" on page 481
"Create Table panel" on page 459

4. Unique constraints (Create Table)

The Unique Constraints panel is displayed if you select Table Creation option 4 (Unique Constraints) on the main Create Table panel.

You use the Unique Constraints panel to specify information to define a primary key and up to three additional unique keys for the table you are creating. The panel shows only those columns that are defined as NOT NULL (and therefore eligible to be selected as part of either the primary key or a unique key).
Creating and dropping DB2 objects

To define the primary key or any of the three unique keys, enter $ or a number in one of the Column Order in Key fields to add a column to the key. If you enter $, FM/DB2 generates the next highest number for that key. To deselect a column that is already selected, clear the displayed number by overtyping it with spaces.

**Note:** The actual numbers are not important; FM/DB2 uses their relative order to define the column order for the key.

You can optionally specify constraint names for the primary key and the three unique keys.

Related references

- “Create Table: Unique Constraints panel” on page 487
- “Create table panel” on page 459

5. Referential constraints (Create Table)

If you select Table Creation option 5 (Referential Constraints) on the main Create Table panel, FM/DB2 displays either the Column Referential Constraints panel or the Table Referential Constraints panel. To change from one panel to the other, press the F11 function key.

**Note:** The first time you select the Referential Constraints creation option, FM/DB2 displays the Column Referential Constraints panel. Thereafter, within the same invocation of the Create Table utility, each time you select the Referential Constraints creation option, FM/DB2 displays whichever panel (column or table) it last displayed.

**Column referential constraints and table referential constraints**

You can specify referential constraints associated with individual columns (column referential constraints), or with the entire table (table referential constraints).

- A column referential constraint relates a column (the foreign key) in the table being created (the dependent table) to an equivalent column (the parent key) in another table (the parent table), and restricts non-null values of the foreign key column to the values of the parent key column.

- A table referential constraint relates a set of columns (the foreign key) in the table being created (the dependent table) to an equivalent set of columns (the parent key) in another table (the parent table), and restricts non-null values of the foreign key to the values of the parent key.

**Note:**

1. The parent key must be a primary or unique key on the parent table.
2. If you do not specify the parent key column name (or names), the foreign key is related to the primary key of the parent table.

To specify column referential constraints, use the Column Referential Constraints panel.

The Column Referential Constraints panel shows the columns defined for the table you are creating. If you have used the modeling facility, the panel shows the column details for each column in the model table. You use this panel to specify information to define column referential constraints for the table you are creating. To define a referential constraint, you must at least specify the parent table name. To indicate a column does not have a referential constraint, leave all the fields for that column blank.
To specify table referential constraints, use the Table Referential Constraints panel. Each row of the panel defines a single table referential constraint.

You use the Table Referential Constraints panel to specify information to define table referential constraints for the table you are creating. To define a referential constraint, you must at least specify the parent table name.

The Create Table Referential Constraint panel is displayed if you enter S in the Cmd entry field.

The Create Table Referential Constraint panel shows the columns defined for the table you are creating (the dependent table). If you have used the modeling facility, the panel shows the column details for each column in the model table. You use this panel to select the dependent table columns (the foreign key) and specify the parent table columns (the parent key) to which they relate.

Related references
“Create Table panel” on page 459
“Create Table: Column Referential Constraints panel” on page 473
“Create Table: Table Referential Constraint panel” on page 485

6. Check constraints (Create Table)

If you select Table Creation option 6 (Check Constraints) on the main Create Table panel, FM/DB2 displays either the Column Check Constraints panel or the Table Check Constraints panel. To change from one panel to the other, press the F11 function key. You use these panels to specify information to define check constraints for the table you are creating.

**Note:** The first time you select the Check Constraints creation option, FM/DB2 displays the Column Check Constraints panel. Thereafter, within the same invocation of the Create Table utility, each time you select the Check Constraints creation option, FM/DB2 displays whichever panel (column or table) it last displayed.

**Column check constraints and table check constraints**
You can specify check constraints associated with individual columns (column check constraints), or with the entire table (table check constraints). Both types of check constraint produce the same result when the statement is issued, and you can specify any valid check condition as either type. However, you might prefer to use table check constraints for check conditions that would violate the SQL standard (not enforced by DB2) that a column check constraint only refers to the column on which it is defined.

**Note:**
1. If model table information is loaded into the Create Table utility, all check constraints defined on the table are loaded as table check constraints.

To specify column check constraints, use the Column Check Constraints panel.

The Column Check Constraints panel shows the columns defined for the table you are creating. If you have used the modeling facility, the panel shows the column details for each column in the model table.

To specify table check constraints, use the Table Check Constraints panel. Each row of the panel defines a single table check constraint.
You use the Table Check Constraints panel to specify an SQL check condition for each check constraint you want to define for the table you are creating.

Related references
- “Create Table panel” on page 459
- “Create Table: Column Check Constraints panel” on page 471
- “Create Table: Table Check Constraints panel” on page 484

7. Procedure exits (Create Table)

The Procedure Exits panel is displayed if you select Table Creation option 7 (Procedure exits) on the main Create Table panel.

You use the Procedure Exits panel to specify information to define procedure exits for the table you are creating.

Related references
- “Create Table: Procedure Exits panel” on page 483
- “Create Table panel” on page 459

8. Generate values (Create Table)

The Generate Values panel is displayed if you select Table Creation option 8 (Generate value) on the main Create Table panel.

You use the Generate Values panel to specify information to define generated values for identity or ROWID columns in the table you are creating.

Related references
- “Create Table: Generate Values panel” on page 476
- “Create Table panel” on page 459

Creating a view

To create a new view, perform the following on the DB2 Object Functions panel:
1. Type 4 in the Object Type entry field.
2. Type C on the command line.
3. Press Enter. FM/DB2 displays the Create View panel.

Use the Create View panel to create a new view at the current location.

Related references
- “DB2 Object Functions panel” on page 510
- “Create View panel” on page 493

Creating an alias

To create a new alias, perform the following on the DB2 Object Functions panel:
1. Type 5 in the Object Type entry field.
2. Type C on the command line.
3. Press Enter. FM/DB2 displays the Create Alias panel.

Use the Create Alias panel to create an alias for a table or a view at the correct location.
Creating and dropping DB2 objects

An alias is an alternative name for a table or view. The table or view need not exist when the alias is created. The table or view can exist at a different location to the local DB2 system.

Related references

“DB2 Object Functions panel” on page 510
“Create Alias panel” on page 430

Creating an index

To create a new index, perform the following on the DB2 Object Functions panel:
1. Type 6 in the Object Type entry field.
2. Type 6 on the command line.
3. Press Enter. FM/DB2 displays the Create Index panel.

Use the Create Index panel to create an index for a pre-existing base table.

The top section of the panel contains five input fields that define the index to be created. If you entered object name information on the DB2 object functions panel, these values are carried forward and the appropriate name fields are pre-filled.

The bottom part of the panel contains a list of options that need to be completed to define the index. There is a single selection field, in which you can type the number of the required option.

The first option is required (define the columns in the index). Typically, the third option is needed (define the space requirements for the index).

When the index has been completely specified, you select option 7 to create the index.

Related tasks

“Index type (Create Index)”
“Index Column Selection (Create Index)” on page 185
“Allocation (Create Index)” on page 185
“Index (Create Index)” on page 185
“Index Partitions (Create Index)” on page 185
“Partition Values Option (Create Index)” on page 185

Related references

“DB2 Object Functions panel” on page 510
“Create Index panel” on page 442
“Create Index: Type panel” on page 451
“Create Index: Column Selection panel” on page 446
“Create Index: Allocation panel” on page 444
“Create Index: Options panel” on page 447
“Create Index: Partitions panel” on page 448
“Create Index: Partition Values panel” on page 450

Index type (Create Index)

You can use the first group of input fields to enter uniqueness information, and the type of index.

Related references

“Create Index: Type panel” on page 451
Creating a synonym

To create a new synonym, perform the following on the DB2 Object Functions panel:

1. Type 7 in the Object Type entry field.
2. Type C on the command line.
3. Press Enter. FM/DB2 displays the Create Synonym panel.

A synonym is an alternative name for a table or view. It is used as an unqualified identifier and is specific to the SQLID that owns the synonym.
Creating and dropping DB2 objects

For more details, see the DB2 UDB for z/OS SQL Reference.

Related references
- “DB2 Object Functions panel” on page 510
- “Create Synonym panel” on page 458

Creating a distinct type

To create a new distinct type, perform the following on the DB2 Object Functions panel:
1. Type 8 in the Object Type entry field.
2. Type C on the command line.
3. Press Enter. FM/DB2 displays the Create Distinct Type panel.

Related references
- “DB2 Object Functions panel” on page 510
- “Create Distinct Type panel” on page 433

Creating a function

To create a new function, perform the following on the DB2 Object Functions panel:
1. Type 9 in the Object Type entry field.
2. Type C on the command line.
3. Press Enter. FM/DB2 displays the Create Function panel.

Use this panel to create an external scalar function.

Related tasks
- “Parameters (Create Function)”
- “Parameter Types (Create Function)”
- “Returned Data Type (Create Function)” on page 187
- “Option List 1/2 (Create Function)” on page 187
- “Option List 2/2 (Create Function)” on page 187

Related references
- “DB2 Object Functions panel” on page 510
- “Create Function panel” on page 437
- “Create Function: Parameters panel” on page 439
- “Create Function: Parameter Types panel” on page 438
- “Create Function: Returned Data Type panel” on page 441
- “Create Function: Option List (1/2) panel” on page 435
- “Create Function: Option List (2/2) panel” on page 436

Parameters (Create Function)

Use this panel to define and optionally name the parameters for the function.

Related references
- “Create Function: Parameters panel” on page 439

Parameter Types (Create Function)

Use this panel to specify additional type information for function parameters.

Related references
- “Create Function: Parameter Types panel” on page 438
Creating and dropping DB2 objects

Returned Data Type (Create Function)
Use this panel to specify the data type returned by the function.

Related references
“Create Function: Returned Data Type panel” on page 441

Option List 1/2 (Create Function)
Use this panel to specify various options to further define the function.

Related references
“Create Function: Option List (1/2) panel” on page 435

Option List 2/2 (Create Function)
This panel lets you specify additional options.

Related references
“Create Function: Option List (2/2) panel” on page 436

Creating a procedure

To create a new procedure, perform the following on the DB2 Object Functions panel:
1. Type 10 in the Object Type entry field.
2. Type C on the command line.
3. Press Enter. FM/DB2 displays the Create Procedure panel.

Related tasks
“Procedure Parameters (Create Procedure)”
“Parameter Types (Create Procedure)”
“Option List 1/2 (Create Procedure)” on page 188
“Option List 2/2 (Create Procedure)” on page 188

Related references
“DB2 Object Functions panel” on page 510
“Create Procedure panel” on page 452
“Create Procedure: Parameters panel” on page 457
“Create Procedure: Parameter Types panel” on page 456
“Create Procedure: Option List (1/2) panel” on page 453
“Create Procedure: Option List (2/2) panel” on page 455

Procedure Parameters (Create Procedure)
This panel is equivalent to the Create Function: Parameters panel.

Related references
“Create Procedure: Parameters panel” on page 457
“Create Function: Parameters panel” on page 439

Parameter Types (Create Procedure)
This panel is equivalent to the Create Function: Parameter Types panel.

Related references
“Create Procedure: Parameter Types panel” on page 456
“Create Function: Parameter Types panel” on page 438
Creating and dropping DB2 objects

**Option List 1/2 (Create Procedure)**
This panel lets you specify various options to further define the procedure.

Related references

“Create Procedure: Option List (1/2) panel” on page 453

**Option List 2/2 (Create Procedure)**
This panel lets you specify additional options.

Related references

“Create Procedure: Option List (2/2) panel” on page 455

**Creating a trigger**
To create a new trigger, perform the following on the DB2 Object Functions panel:
1. Type 11 in the **Object Type** entry field.
2. Type C on the command line.
3. Press Enter. FM/DB2 displays the Create Trigger panel.

Related tasks

“Details (Create Trigger)”
“Search Conditions (Create Trigger)”
“SQL Statement (Create Trigger)”

Related references

“DB2 Object Functions panel” on page 510
“Create Trigger panel” on page 488
“Create Trigger: Details panel” on page 490
“Create Trigger: Search Condition panel” on page 491
“Create Trigger: SQL statement panel” on page 492

**Details (Create Trigger)**
This panel lets you specify additional details required for the definition of a trigger.

Related references

“Create Trigger: Details panel” on page 490

**Search Conditions (Create Trigger)**
Use this panel to specify a search condition. The triggered SQL statements are run only if the search condition evaluates to true or is omitted.

Related references

“Create Trigger: Search Condition panel” on page 491

**SQL Statement (Create Trigger)**
Use this panel to specify the SQL to be run if the trigger is activated.

Related references

“Create Trigger: SQL statement panel” on page 492
Creating an auxiliary table

To create a new auxiliary table, perform the following on the DB2 Object Functions panel:
1. Type 12 in the Object Type entry field.
2. Type C on the command line.
3. Press Enter. FM/DB2 displays the Create Auxiliary Table panel.

Auxiliary tables are required for LOB support. An auxiliary table is needed if a table contains a LOB column. If such a column is defined, an auxiliary table needs to be defined to store the data for that column. If the table is also partitioned, there needs to be an auxiliary table for each partition.

Related references

“DB2 Object Functions panel” on page 510
“Create Auxiliary Table panel” on page 431

Dropping DB2 objects

The available drop functions are shown in Table 9 on page 163.

To drop a DB2 object, perform the following using the DB2 Object Functions panel:
1. Specify the relevant number (1–11) for the DB2 object type you want to drop in the Object Type entry field.
2. If you want FM/DB2 to display a Drop Confirmation panel before it actually drops the object you have specified, check the Confirm object drop option.
3. Type D on the command line.
4. Press Enter. FM/DB2 displays the relevant Drop panel.
5. Specify the name of the DB2 object you want to drop and press Enter.

If you have selected the Confirm object drop option, FM/DB2 displays a Drop Confirmation panel. To proceed with the drop, press Enter. To cancel the drop, press the Cancel function key (F12) or the Exit function key (F3).

If the drop operation fails, FM/DB2 displays an error message; press the Help function key (F1) for further information.

Related references

“DB2 Object Functions panel” on page 510
“Drop panels” on page 524
“Drop Confirmation panel” on page 525
Creating and dropping DB2 objects
Chapter 6. Populating a DB2 table with data

You can populate a table with data by using the FM/DB2 Create utility (3.8). You can specify the number of rows that are to be created and how each row is to be initialized. By editing a template, you can use fill characters and patterns to initialize the data and change the data create attributes for individual columns.

When fill characters and/or patterns are used to initialize a column, the resultant data must conform to any data type restrictions imposed by DB2 for that column.

During the data creation process, FM/DB2 attempts to insert each row formatted by the data create utility into the target DB2 object. The insert operation may be rejected by DB2 if uniqueness, referential integrity, or other constraints would be violated.

The data create operation can be performed in background (batch execution) or in the foreground.

The main panel for the FM/DB2 Create utility is the Data Create Utility panel.

To display the Data Create Utility panel,
1. From the Primary Option Menu panel, enter 3 (Utilities) to display the Utility Functions panel.
2. From the Utility Functions panel, enter 8 (Create)

Now that you have the Data Create Utility panel displayed, you can specify the name of the table you want to populate with data, the number of rows that are to be created, and how each row is to be initialized:
1. Identify the DB2 table you want to populate with data using the Specify the DB2 Object entry fields.
2. Use the Create Count entry field to specify the number of rows you want to create.
3. If you are using a specific template to specify the data initialization details, (see option 1 in Template usage), specify the name of the template data set.
4. Use the Processing Options to specify which template you want to use for the create process, and whether you want to edit the template before the create process.
5. Press Enter. If you selected Edit template, the Column Selection/Edit panel is displayed. Otherwise, FM/DB2 creates the data in the specified DB2 table.

Related tasks
   "Specifying a DB2 object name" on page 25
   "Specifying a data set and a member name" on page 30
   "Editing the template" on page 192

Related references
   "Primary Option Menu panel" on page 617
   "Utility Functions panel" on page 720
   "Data Create Utility panel" on page 495
Editing the template

If you selected the **Edit template** option on the Create panel, FM/DB2 displays the Column Selection/Edit panel when you press Enter.

To populate the columns for the DB2 table you have specified by editing the template:

1. Enter E against the first column in the Column Selection/Edit panel. FM/DB2 displays one of five Column Attributes panels depending on the data type of the column:
   - Alphanumeric (for CHAR, VARCHAR, GRAPHIC, and VARGRAPHIC data types)
   - Numeric (for DECIMAL or NUMERIC, SMALLINT, INTEGER, REAL, and DOUBLE data types)
   - DATE
   - TIME
   - TIMESTAMP

2. Use the Column Attributes panel to specify the *create attributes* for the column. The create attributes determine the value that FM/DB2 puts in the column when it creates data.
   - For details about how to enter the create attributes, refer to the description for the relevant Column Attributes panel.

3. Press the Exit function key (F3) to return to the Column Selection/Edit panel.

4. Repeat from Step 1 for each of the remaining columns you want to populate.

Related references:
- Chapter 14, “DB2 data types,” on page 361
- “Column Selection/Edit panel” on page 407
- “Column Attributes panel (alphanumeric)” on page 397
- “Column Attributes panel (numeric)” on page 402
- “Column Attributes panel (DATE)” on page 400
- “Column Attributes panel (TIME)” on page 405
- “Column Attributes panel (TIMESTAMP)” on page 406
Chapter 7. Copying data

FM/DB2 provides the following utilities that allow you to copy data from one place to another:

**Copy Utility (3.3)**
Copies data from one DB2 object to another DB2 object (that is accessible from within the same DB2 subsystem).

**Import Utility (3.6)**
Copies data from a VSAM or QSAM file into a DB2 table or view.

**Export Utility (3.7)**
Copies data from a DB2 table or view to a VSAM or QSAM file.

Related tasks
- "Copying data from one DB2 object to another" on page 193
- "Copying data from a VSAM or QSAM file" on page 198
- "Copying data to a VSAM or QSAM file" on page 221
- "How do I...?" on page 233

**Copying data from one DB2 object to another**

To copy data between DB2 tables, use the Copy utility function (3.3).

**Note:**
1. To copy data to a DB2 table or view from a QSAM or VSAM data set, use the Import utility function (3.6).
2. To copy data from a DB2 table or view to a QSAM or VSAM data set, use the Export utility function (3.7).

When you copy data from one DB2 object to another, you can:
- Select which rows to copy
- Reformat data during the copy by mapping columns in the input table to different columns in the output table
- Initialize new columns using user-specified values or patterns
- Limit the number of rows copied

The main panel for the Copy utility is the Copy Utility ("From") panel.

To display the Copy Utility ("From") panel,
1. From the Primary Option Menu panel, enter 3 (Utilities) to display the Utility Functions panel.
2. From the Utility Functions panel, enter 3 (Copy)

You can now specify details about:
- The source table (where the data is that you want to copy). During this step, you can also edit the source template to restrict the data that is copied (if applicable).
- The target table (where you want the data copied). During this step, you can also change the copy options, edit the template mapping for the target table, and specify if you want the copy performed in a batch job.
Specifying details for the “From” table

Specify the name of the source (“From”) table on the Copy Utility “From” panel. If the table you specify does not exist, FM/DB2 issues an error message and stops the utility.

If you do not want to copy all of the rows in the “From” table, specify the number of rows you want to copy (1–99999999) in Copy Count.

If you are using a specific template for the “From” table (see options 1 and 4 in Template usage), specify the name of the template data set and, optionally, the member name, in the From Template entry fields.

The Processing Options specify which template you want to use for the “From” table, and whether you want to edit the template before the copy process.

Template usage

The Copy Utility panel provides four options for template usage:

1. Above
   Requires that you enter the name of a template data set (and optionally a member name) in the From Template section of the panel. If you do not specify a member name, or if you specify a pattern, FM/DB2 displays a member selection list. After you have specified the member containing a template, processing continues using this template in place of the automatically-generated template for the table. (For a detailed explanation, see Template Usage option 3).

2. Previous
   Uses the last (previously used) template for this table.

3. Generate from table
   Generates a template based on the information in the DB2 catalog for the specified table. This is the default setting. If you have specified a member name in the Template From section of the panel, it is ignored. FM/DB2 generates the required template in memory.

4. Generate/Replace
   Generates a template based on the information in the DB2 catalog for the specified table, and saves this information in the member specified in the From Template section of the panel. If the member already exists, it is replaced. Any prior customization is lost.

You can select the following options by entering a “/” or an “A”:
Copying data from one DB2 object to another

**Edit template**

Use to edit the template for the “From” table before proceeding to the Copy Utility “To” panel.

**Copy “From” values to “To” panel**

Use to copy the values (DB2 object Location, Owner, Name, and template Data set name, Member) entered on the “From” panel to the corresponding fields on the “To” panel.

To proceed, press Enter.

If you selected **Edit template** (either by using a “/”, or by having previously used the “A” selection character for this option), the Column Selection/Edit panel is displayed. Otherwise, the Copy Utility “To” panel is displayed.

**Related tasks**

- “Specifying a DB2 object name” on page 25
- “Selecting options on FM/DB2 panels” on page 24
- Chapter 3, “Working with templates,” on page 49
- “Editing the template for the “From” table”
- “Specifying details for the “To” table” on page 196

**Related references**

- “Copy Utility (“From”) panel” on page 418
- “Column Selection/Edit panel” on page 407
- “Copy Utility (“To”) panel” on page 424

**Editing the template for the “From” table**

If you select the **Edit template** option on the Copy Utility “From” panel, FM/DB2 displays the Column Selection/Edit panel (Figure 49 on page 196). You can use this panel to restrict the data in the “From” table that you want copied by:

- Selecting (or deselecting) specific columns for copying,
- Specifying row selection criteria. Only rows whose contents satisfy the criteria are selected for copying.

For example, the template shown in Figure 49 on page 196 has been edited to select certain columns and only rows whose contents meet certain criteria. If you use this template as the “From” template for your Copy, the data copied is restricted to columns where WORKDEPT contains “MNT” and JOB contains “PAINTER”.


Copying data from one DB2 object to another

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>Column Selection/Edit</td>
<td>Line 1 of 16</td>
<td></td>
</tr>
</tbody>
</table>

TABLE FMMUSER.EMP

-------- Row Selection Criteria ----- (Use SQL/PF4 for full screen edit) --------
1 Sel: WHERE WORKDEPT = 'MNT' AND JOB = 'PAINTER'

Cmd Seq SHE CL# Column name Data type(length) Null Default Order A/D

**** Top of data ****

--- S 1 EMPNO CHARACTER(6) None --- -
--- S 2 FIRSTNME VARCHAR(12) None --- -
--- S 3 MIDINIT CHARACTER(1) None --- -
--- S 4 LASTNAME VARCHAR(15) None --- -
--- S 5 WORKDEPT CHARACTER(3) Y Null --- -
--- S 6 PHONENO CHARACTER(4) Y Null --- -
--- S 7 HIREDATE DATE Y Null --- -
--- S 8 JOB CHARACTER(8) Y Null --- -
--- S 9 EDLEVEL SMALLINT Y Null --- -
--- S 10 SEX CHARACTER(1) Y Null --- -
--- S 11 BIRTHDATE DATE Y Null --- -
--- S 12 SALARY DECIMAL(9,2) Y Null --- -

Command ==> ___________________________________________________ Scroll PAGE
F1=Help F2=Split F3=Exit F4=SQL F5=RFind F6=RunTemp
F7=Up F8=Down F9=Swap F10=Left F11=Right F12=Cancel

Figure 49. Primary Column Selection/Edit panel

Related tasks

- Chapter 3, “Working with templates,” on page 49

Related references

- “Copy Utility (“From”) panel” on page 418
- “Column Selection/Edit panel” on page 407
- “Copy Utility (“To”) panel” on page 424

Specifying details for the “To” table

Specify the name of the target (“To”) table on the Copy Utility “To” panel.

If you are using a specific template for the “To” table, specify the name of the template data set and, optionally, the member name, in the To Template entry fields.

The Processing Options specify which template you want to use for the “To” table, and whether you want to edit the template before the copy process.

Template usage

The Copy Utility panel provides four options for template usage:

1. Above

Requirements that you enter the name of a template data set (and optionally a member name) in the To Template section of the panel. If you do not specify a member name, or if you specify a pattern, FM/DB2 displays a member selection list. After you have specified the member containing a template, processing continues using this template in place of the automatically-generated template for the table. (For a detailed explanation, see Template Usage option 3).

2. Previous

Uses the last (previously used) template for this table.

3. Generate from table

Generates a template based on the information in the DB2 catalog.
Copying data from one DB2 object to another

for the specified table. This is the default setting. If you have specified a member name in the To Template section of the panel, it is ignored. FM/DB2 generates the required template in memory.

4. Generate/Replace
Generates a template based on the information in the DB2 catalog for the specified table, and saves this information in the member specified in the To Template section of the panel. If the member already exists, it is replaced. Any prior customization is lost.

You can select the following options by entering a “/” or an“A”:

Edit copy options
Use to edit the Copy options before copying the data.

Edit template mapping
Use to edit the template mapping for the “To” table before copying the data.

Batch execution
Builds the necessary JCL to perform the Copy utility in a batch job.

To proceed, press Enter.

If you selected Edit copy options (either by using a “/”, or by having previously used the “A” selection character for this option), the Copy Options panel is displayed for editing.

If you selected Edit template mapping (either by using a “/”, or by having previously used the “A” selection character for this option), the Template Mapping panel is displayed for editing.

If you selected Batch execution (either by using a “/”, or by having previously used the “A” selection character for this option), the generated JCL is displayed using the ISPF editor, where you can review the JCL and submit the job for batch processing.

Otherwise (if you selected none of the preceding options), FM/DB2 copies the data from the “From” table to the “To” table.

Setting options for the current Copy session
If you select Edit copy options on the Copy “To” panel FM/DB2 displays the Copy Options panel.

You can use this panel to change the options for the current Copy session.

Note: Any options you set on this panel apply for the current Copy session only. When you exit from the Copy session, the copy options revert to the global copy options.

Related references
“Copy Options panel” on page 415

Running Copy in batch
If you want to copy large DB2 tables, you can perform the copy in a batch job.

If you select the Batch execution option on the Copy Utility “From” panel, the Copy utility builds the necessary JCL to perform the copy function in a batch job.
Copying data from one DB2 object to another

The generated JCL is displayed using the ISPF editor, where you can review the JCL and submit the job for batch processing.

Related tasks

- “Specifying a DB2 object name” on page 25
- “Selecting options on FM/DB2 panels” on page 24
- “Setting options for the current Copy session” on page 197
- “Mapping data” on page 80
- “Running Copy in batch” on page 197

Related references

- “Copy Utility (”To”) panel” on page 424
- “Copy Utility (”From”) panel” on page 418
- “Copy Options panel” on page 415
- “Copy utility options (option 3.3)” on page 44

Copying data from a VSAM or QSAM file

To copy data to a DB2 table or view from a QSAM or VSAM data set, use the Import utility function (3.6).

**Note:**

1. To copy data between DB2 tables, use the Copy utility function (3.3).
2. To copy data from a DB2 table or view to a QSAM or VSAM data set, use the Export utility function (3.7).

The input to the Import utility function can be a QSAM or VSAM data set produced by:

- The FM/DB2 Export utility
- Any external source

You must specify a copybook or template that describes the record format of the data in the input QSAM or VSAM data set. If you specify a copybook, FM/DB2 compiles it to produce a template.

When you copy data to a DB2 table or view from a QSAM or VSAM data set, you can:

- Reformat data during import by mapping DB2 table columns to fields in the input QSAM or VSAM data set (as defined by a copybook or template).
- Use a REXX procedure to enhance the way in which data is selected and imported.
- Specify the number of rows to be imported, and the starting record in the import file.
- Specify whether you want to run the import in foreground or batch.
- Specify that any duplicate rows are to be updated with the values in the imported row.
- Specify how often FM/DB2 should commit changes to DB2 during the import process.

The main panel for the Import utility is the Import Utility (”From”) panel.

To display the Copy Utility (”From”) panel,

1. From the Primary Option Menu panel, enter 3 (Utilities) to display the Utility Functions panel.
Copying data from a VSAM or QSAM file

2. From the Utility Functions panel, enter 6 (Import)

You can now specify details about:
- The source QSAM or VSAM data set (where the data is that you want to copy). During this step, you can also edit the source template to restrict the data that is copied (if applicable).
- The target table (where you want the data copied). During this step, you can also:
  - Edit the template mapping for the target table
  - Specify if you want the copy performed in a batch job
  - Optionally specify the name of a REXX procedure to further enhance the way data is selected

When (and when not) to use the Import utility

The Import utility provides one method for moving DB2 data between different DB2 subsystems (that is, export on one subsystem followed by import on another). If you want to move data between tables in the same DB2 sub-system, either the Copy utility or direct execution of SQL statements is typically more convenient.

In general terms, the Import utility is suitable for inserting small to medium volumes of data into DB2 tables. Import operations are achieved by inserting each selected record in the import data set into the target table, and any update operations are achieved by generating an update statement.

All insert and update operations are logged by DB2, which can have operational and performance implications. The Import utility is not intended as a substitute for the DB2 utilities if unloading or migrating large volumes of data, and the DB2 utilities offer superior performance in these situations. Whenever possible, run import operations in batch.

The import utility provides for automatic mapping of the DB2 null indicator, and, optionally, one of the following formats can be specified corresponding to the template used:
- FM/DB2 (SQLDA) format
- DB2 unload format
- DSNTIAUL format
- User defined format (excluding Placement, User defined)

However, the template editor does not show the null indicator as a separate item in the "From" template on the Field Mapping panel. It is therefore not possible to re-map the null indicator associated with nullable columns to a column in the target table.

Related tasks
- "Compiler language selection (option 0.4)" on page 42
- "Specifying details for the "From" data set" on page 200
- "Specifying details for the "To" table" on page 201
- "Using a REXX procedure with the Import utility" on page 203

Related references
- "Primary Option Menu panel" on page 617
- "Utility Functions panel" on page 720
- "Import Utility ("From") panel" on page 591
- "Import Utility ("To") panel" on page 592
Specifying details for the “From” data set

Specify the name of the source (“From”) data set on the Import Utility “From” panel. Use a combination of the Data set name, Member, and Volume entry fields to specify the sequential data set, VSAM data set, or PDS member from which you want to import the data.

If you do not want to import all of the records from the input data set, specify the number of records you want to import (1–99999999) in Import Count.

If you are using a specific template for the “From” data set (see option 1 in Template usage), specify the name of the template data set and, optionally, the member name, in the From Copybook or Template entry fields.

The Processing Options on the Import Utility “From” panel specify template usage, and whether you want to edit the template before the import process.

Template usage

The Import Utility “From” panel provides two options for template processing:

1. **Above**
   Requires that you enter the name of a copybook or template in the From Copybook or Template section of the panel. After you have specified a name of a copybook or template, processing continues using this template in place of the previously used template (see Template usage option 2).

2. **Previous**
   Uses the last (previously used) template for this data set.

You can select the following option by entering a “/” or an “A”:

**Edit template**
Use to edit the template for the “From” data set before proceeding to the Import Utility “To” panel.

To proceed, press Enter.

If you selected **Edit template** (either by using a “/”, or by having previously used the “A” selection character for this option), the Field Selection/Edit panel is displayed.

Otherwise, the Import Utility “To” panel is displayed.

Related tasks

- “Specifying a data set and a member name” on page 30
- “Selecting options on FM/DB2 panels” on page 24
- Chapter 3, “Working with templates,” on page 49
- “Editing the template for the “From” data set”
- “Specifying details for the “To” table” on page 201

Related references

- “Import Utility ("From") panel” on page 591
- “Import Utility ("To") panel” on page 592

**Editing the template for the “From” data set**

If you select the **Edit template** option on the Import Utility “From” panel, FM/DB2 displays the Field Selection/Edit panel (see Figure 50 on page 201).
You can narrow down which records you want to process by specifying record selection criteria. Only records whose contents satisfy the criteria are selected. You specify record selection criteria using REXX comparison expressions.

**Note:** Record selection criteria is applied on the input data before any field mapping or data conversion is performed, whereas the REXX import procedure applies after any field mapping or data conversion is performed.

For example, the template shown in Figure 50 has been edited to select only records that meet certain criteria. If you use this template as the “From” template for your Import utility, the data imported is restricted to records where SALARY contains a value greater than 50 000.

If you press Enter, FM/DB2 saves the updated template and returns you to the Import Utility “From” panel.

**Note:** For more information about specifying record selection criteria, see the File Manager User’s Guide and Reference.

Related tasks

- “General information about REXX” on page 204

Related references

- “Import Utility (“From”) panel” on page 591
- “Field Selection/Edit panel” on page 570

**Specifying details for the “To” table**

Specify the name of the table to which the input data is to be imported on the Import Utility “To” panel. If the table you specify does not exist, FM/DB2 issues an error message.

If you are using a specific template for the “To” table (see options 1 and 4 in Template usage), specify the name of the template in the To Template entry fields.
Copying data from a VSAM or QSAM file

To delete all the rows in the target table before importing the data from the QSAM or VSAM data set, select the **Delete existing rows** option.

Set the **Duplicate row options** according to how you want FM/DB2 to handle any instances where the data to be imported is a “duplicate” of an existing row; that is, where the only changes to the data are in non-unique index columns:

- To ignore duplicate rows, select **1. Ignore**.
  Any duplicates that FM/DB2 detects are not updated and FM/DB2 ignores the value (if any) in the **For ... duplicates** field.
- To update duplicate rows, select **2. Update**.
  FM/DB2 updates any duplicate rows. If the number of duplicates exceeds the value specified in the **For ... duplicates** field, FM/DB2 terminates the import operation and all changes are rolled back.

To proceed, press Enter.

If you selected **Edit template mapping** (either by using a “/”, or by having previously used the “A” selection character for this option), then, before FM/DB2 imports the data, the Template Mapping panel is displayed for editing.

If you selected **Batch execution** (either by using a “/”, or by having previously used the “A” selection character for this option), the generated JCL is displayed using the ISPF editor, where you can review the JCL and submit the job for batch processing.

Otherwise, FM/DB2 imports the data from the “From” data set to the “To” DB2 table.

Related tasks

- “Specifying a DB2 object name” on page 25
- “Specifying a data set and a member name” on page 30
- “Mapping data” on page 80
- “Running Import in batch” on page 710

Related references

- “Import Utility (“To”) panel” on page 592
- “Template Mapping panel” on page 710

**Running Import in batch**

If you want to import large DB2 tables, you can perform the import in a batch job.

If you select the **Batch execution** option on the Import Utility “To” panel, the Import utility builds the necessary JCL to perform the import function in a batch job. The generated JCL is displayed using the ISPF editor, where you can review the JCL and submit the job for batch processing.

If you have one or more data sets that contain the template that describes the record structure of your input data, you can specify the data set details (or TSO ALLOC statement) in the generated JCL with the keywords TINPUT or TCIN.

Similarly, if you have one or more data sets that contain the template that describes the record structure of your output data, you can specify the data set details (or TSO ALLOC statement) in the generated JCL with the keywords TOUTPUT or TCOUT.
Using a REXX procedure with the Import utility

You can specify an existing REXX procedure, or create a new one, to further enhance the way in which data is selected and imported.

You can code any number of REXX statements and functions to manipulate the output record, select specific records, print reports and tally numeric values. If you are coding a REXX procedure, be aware that:

- The import procedure is run after input template record selection is performed. The import procedure can cause a record (which would have been selected by the template selection process, if there was one) to be deselected for copying using RETURN DROP.
- The import procedure is run after any field mapping or conversion supplied by a template has been performed.
- If the record has been reformatted as a result of the template processing, the variable INREC contains the input record value and the variable OUTREC contains the reformatted output record.
- Changes made by the import procedure to INREC are ignored. Changes made to OUTREC are used in the import process to load the DB2 table.
- If using the PRINT() function, the template is determined as follows:
  - If an input template without an output template has been specified, the input template is used.
  - If an output template is specified, and the record value specified to print is the input record, the input template is used. Otherwise the output template is used.
  - If a record cannot be matched against the selected printing template, it is not printed.

Example 1

Import records, inserting a new field after column 40 for a length of 4.
```rexx
outrec = fld(1,40)||' '||fld(41)
```

Example 2

Import records with type A and print the first 10. The type value is in column 6.
```rexx
if fld(6,1) = 'A' then do /* Type A? */
  if prtcount() <= 10 then print(inrec,'CHAR') /* Print 10 */
  RETURN /* Copy */
end
return 'DROP' /* Drop the rest */
```

Example 3

Change the output records to uppercase while importing.
```rexx
upper outrec
```

Related tasks

- "Specifying details for the “To” table" on page 201
- "Enhancing the Import utility using a REXX procedure" on page 211
General information about REXX

REXX stands for the REstructured eXtended eXecutor language. REXX is a general purpose programming language, similar to PL/I. It includes extensive parsing capabilities for character manipulation, extensive mathematical capabilities, and numerous built-in functions that perform various processing, searching, and comparison functions.

REXX programs are typically run by a language processor (interpreter). That is, the program is run statement-by-statement, without first being converted to another form (compiled). Because of the interpretive nature of REXX, FM/DB2 is able to exploit the power of REXX when processing record selection criteria.

Although REXX is a powerful and versatile language, coding selection criteria using REXX is straightforward. You do not need to know how to write REXX programs. All you need to know is how to write REXX comparison expressions. FM/DB2 takes care of turning your selection criteria into a REXX program.

Much of the information in this section has been extracted from the OS/390 TSO/E REXX Reference and OS/390 TSO/E REXX User’s Guide. For more detailed information about REXX, see those manuals.

Overview of REXX expressions

REXX expressions consist of one or more terms interspersed with zero or more operators that denote operations to be carried out on terms. Expressions are evaluated left to right, modified by parentheses and by operator precedence in the typical algebraic manner. If parentheses are encountered (other than those that identify function calls) the entire subexpression between the parentheses is evaluated immediately when the term is required. Expressions are wholly evaluated, unless an error occurs during evaluation. The REXX language uses a free format. This means you can insert extra spaces between words without causing an error.

The following list shows the terms you can use in a selection criteria expression. The operators you can use are described on the pages that follow.

Literal strings

A literal string is a sequence including any characters and delimited by the single quotation mark (’) or the double quotation mark (“). Use two consecutive double quotation marks (““) to represent a “ character within a string delimited by double quotation marks. Use two consecutive single quotation marks (‘’) to represent a ’ character within a string delimited by single quotation marks. A literal string is a constant and its contents are never modified when it is processed.

These are valid strings:

‘Fred’
"Don’t Panic!"
‘You shouldn’t’ /* Same as "You shouldn't" */

Note that a string followed immediately by a ( is considered to be the name of a function. If followed immediately by the symbol X or x it is considered to be a hexadecimal string. If followed immediately by the symbol B or b it is considered to be a binary string.

A hexadecimal string is a literal string, expressed using a hexadecimal notation of its encoding. It is any sequence of zero or more hexadecimal digits (0-9, a-f, A-F), grouped in pairs. A single leading 0 is assumed, if necessary, at the front of the string to make an even number of
hexadecimal digits. The groups of digits are optionally separated by one or more blanks, and the whole sequence is delimited by single or double quotation marks, and immediately followed by the symbol X or x. (Neither x nor X can be part of a longer symbol. A hexadecimal string is a literal string formed by packing the hexadecimal digits given. Packing the hexadecimal digits removes blanks and converts each pair of hexadecimal digits into its equivalent character, for example: 'C1'X A.

You can use hexadecimal strings to include characters in a program even if you cannot directly enter the characters themselves. These are valid hexadecimal strings:

'ABCD'x  
"1d ec f8"X  
"1 d8"x

A binary string is a literal string, expressed using a binary representation of its encoding. It is any sequence of zero or more binary digits (0 or 1) in groups of 8 (bytes) or 4 (nibbles). The first group can have fewer than four digits; in this case, up to three 0 digits are assumed to the left of the first digit, making a total of four digits. The groups of digits are optionally separated by one or more blanks, and the whole sequence is delimited by matching single or double quotation marks and immediately followed by the symbol b or B.

A binary string is a literal string formed by packing the binary digits given. If the number of binary digits is not a multiple of eight, leading zeros are added on the left to make a multiple of eight before packing. You can use binary strings to specify characters explicitly, bit by bit.

These are valid binary strings:

'11110000'b /* == 'f0'x */  
"101 1101"b /* == '5d'x */  
'1'b /* == '00000001'b and '01'x */  
'1000 10101010'b /* == '0001 0000 1010 1010'b */  
''b /* == '' */

Symbols

Character strings, without quotation marks, which are converted to uppercase. Any symbol that begins with a # is treated as a reference to a field in the record being processed, and the value of the field is used. All other symbols are treated as constants.

FM/DB2 assigns to each field defined in a template a unique field reference number. If you want to refer to a field in a selection criteria expression, you specify the field’s field reference number prefixed by #. Note that you can only refer to fields defined in the record you are currently working with.

REXX expression evaluation only processes data in the form of “typeless” character strings (typeless because they are not, as in the case of COBOL, of a particular data type such as binary, packed-decimal, and so forth). Therefore, if you refer to a numeric field, FM/DB2 converts the value of the field to a numeric character string that can be processed by REXX when evaluating the selection criteria expression. The number of integer and decimal digits in the character string is determined by the field definition in your template. For example, if you refer to a packed-decimal field with a COBOL PICTURE clause of 999V99, FM/DB2 converts the value of the field to a character string consisting of numeric digits, a period for the virtual decimal place, and, if the value of the field is
negative, a leading sign character (such as -123.45). Note that all numeric fields are treated as signed, regardless of whether the COBOL PICTURE clause contains a sign symbol.

Occasionally, you might want to evaluate the value of a numeric field without converting it to a numeric character string. To do this, prefix the field reference number by #u instead of #. This tells FM/DB2 not to convert the number to a numeric character string. For example, if to test a 2-byte binary numeric field (with a field reference number of 45) for a special value of X'FFFF', code:

```
#u45 = 'FFFF'
```

If you refer to a field in an array, you must qualify the field reference with the required number of subscripts enclosed in parentheses and separated by commas. The number of subscripts you specify must equal the number of dimensions in the array that contains the field you are referencing. In COBOL terms, there must be a subscript for each OCCURS clause in the hierarchy containing the field, including any OCCURS clause for the field itself. Each subscript you specify must be a positive integer in the range 1 to the maximum number of occurrences of the field as specified in the OCCURS clauses. If the field you refer to is in a variable-length array (specified using the OCCURS DEPENDING ON clause), ensure that you do not refer to an occurrence of the field that, for any given record, might not exist. If you do refer to a non-existent field, the selection criteria is not satisfied, and the record is not selected.

**Note:** The object of an OCCURS DEPENDING ON clause must be defined as a field in the static portion of the same record as the array (that is, the portion of the record that precedes any variable-length array defined with the OCCURS DEPENDING ON clause). If the object of the OCCURS DEPENDING ON clause is not defined in this way, you cannot refer to any fields in the array or any fields in the record that follow the array.

**Function call**
A call to REXX built-in function.

**Subexpressions**
Terms in an expression bracketed within left and right parentheses.

**Comparison operators**
REXX comparison operators compare two terms and return the value 1 if the result of the comparison is true, or 0 otherwise. Comparison operators can compare numbers or character strings. The most commonly used comparison operators are:

<table>
<thead>
<tr>
<th>Operator</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>=</td>
<td>Equal</td>
</tr>
<tr>
<td>^=, =</td>
<td>Not equal</td>
</tr>
<tr>
<td>&lt;</td>
<td>Less than</td>
</tr>
<tr>
<td>&lt;=, &lt;</td>
<td>Not less than</td>
</tr>
<tr>
<td>&gt;=, &gt;</td>
<td>Greater than</td>
</tr>
<tr>
<td>&gt;</td>
<td>Greater than</td>
</tr>
<tr>
<td>&gt;&gt;</td>
<td>Not greater than</td>
</tr>
<tr>
<td>&lt;=</td>
<td>Less than or equal to</td>
</tr>
<tr>
<td>&gt;=</td>
<td>Greater than or equal to</td>
</tr>
<tr>
<td>&gt;&lt;&gt;</td>
<td>Greater than or less than (same as not equal)</td>
</tr>
<tr>
<td>&lt;&gt;</td>
<td>Less than or greater than (same as not equal)</td>
</tr>
</tbody>
</table>
Note that the not character (¬), is synonymous with the backslash (\). You can use the two characters interchangeably.

When comparing terms using these comparison operators, if both terms in the expression are numeric, REXX performs a numeric comparison. Otherwise, both terms are treated as character strings and REXX performs character comparison. (A number in REXX is a string that contains one or more decimal digits, an optional decimal point, and an optional leading sign character. The string can contain one or more leading or trailing blanks, and the sign character can be separated from the digits by one or more blanks.)

In a character comparison, leading and trailing blanks are ignored, and the shorter string is padded with blanks on the right. Character comparisons are case-sensitive. To prevent lowercase characters being converted to uppercase, delimit character strings with quotation marks. For example, if the field #4 contains the value MixedCase both the following comparison operations would be true:

```
#4 = 'MixedCase'
#4 = ' MixedCase '  
```

but the following comparison operation would not be true:

```
#4 = MixedCase
```

In numeric comparisons, the comparison is effected by subtracting the two numbers (calculating the difference) and then comparing the result with 0. For example, the comparison operation:

```
#6 = 10
```

is identical to the operation:

```
(#6 - 10) = 0
```

In addition to these comparison operators, REXX provides a number of “strict” comparison operators that are mainly intended for use if comparing character strings. The strict comparison operators all have one of the characters defining the operator doubled, such as == (strictly equal).

The strict comparison operators are:

<table>
<thead>
<tr>
<th>Operator</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>==</td>
<td>Strictly equal</td>
</tr>
<tr>
<td>=~, ==</td>
<td>Strictly not equal</td>
</tr>
<tr>
<td>&lt;</td>
<td>Strictly less than</td>
</tr>
<tr>
<td>=~&lt;, &lt;&lt;</td>
<td>Strictly not less than</td>
</tr>
<tr>
<td>&gt;</td>
<td>Strictly greater than</td>
</tr>
<tr>
<td>=~&gt;, &gt;&gt;</td>
<td>Strictly not greater than</td>
</tr>
<tr>
<td>&lt;=</td>
<td>Strictly less than or equal to</td>
</tr>
<tr>
<td>&gt;=</td>
<td>Strictly greater than or equal to</td>
</tr>
</tbody>
</table>

If you use the == comparison operator (strictly equal), the two character strings being compared must be identical (character by character) and of the same length to be considered strictly equal. Leading and trailing blanks are significant. For example, continuing the example using field #4 that contains the value MixedCase only the first of the following comparison operations would be true:
Similarly, the strict comparison operators such as $\gg$ or $\ll$ carry out a simple character-by-character comparison, with no padding of either of the strings being compared. The comparison of the two strings is from left to right. If one string is shorter than and is a leading substring of another, then it is smaller than (less than) the other. The strict comparison operators do not attempt to perform a numeric comparison on the two terms. Do not use them to compare numeric fields.

**Arithmetic operators**

You can process numeric terms in comparison expressions using the arithmetic operators:

<table>
<thead>
<tr>
<th>Operator</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>$+$</td>
<td>Add</td>
</tr>
<tr>
<td>$-$</td>
<td>Minus</td>
</tr>
<tr>
<td>$*$</td>
<td>Multiply</td>
</tr>
<tr>
<td>$/$</td>
<td>Divide</td>
</tr>
<tr>
<td>$%$</td>
<td>Integer divide (divide and return the integer part of the result)</td>
</tr>
<tr>
<td>$// $</td>
<td>Remainder (divide and return the remainder—not modulo, because the result might be negative)</td>
</tr>
<tr>
<td>$^\star\star$</td>
<td>Power (raise a number to a whole-number power)</td>
</tr>
</tbody>
</table>

*Prefix -*  
Same as the subtraction: $0 - \text{number}$

*Prefix +*  
Same as the addition: $0 + \text{number}$

You can use these operators to produce an intermediate result that you can compare with another term. For example, given these fields:

- #6 Contains a numeric value representing an employee's annual salary.
- #15 Contains a numeric value representing his annual travel allowance.
- #23 Contains a numeric value representing his annual bonus.

you can use this comparison to select records for employees with a combined annual payment of greater than $100,000:

$$ (#6 + #15 + #23) > 100000 $$

For another example, with these fields:

- #45 Contains the number of sick days an employee is entitled to annually.
- #46 Contains the number of sick days an employee has used in the current year.

you can use this comparison to select records for employees who have used 50% or more of their sick days entitlement:

$$ (#46 / #45) \geq .5 $$

Note that in each of these examples, the arithmetic subexpression is contained in parentheses. This ensures that the entire subexpression is evaluated before the comparison operation.

The order of precedence of arithmetic operators is as follows (highest is at the top):

<table>
<thead>
<tr>
<th>Operator</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>$+$</td>
<td>$-$ $\ldots$ $\backslash$ Prefix operators</td>
</tr>
</tbody>
</table>
** Power
*/ Multiply and divide
+ - Add and subtract

For example, * (multiply) has a higher priority than + (add), therefore 3+2*5 evaluates as 13 (rather than the 25 that would result if strict left to right evaluation occurred). To force the addition to occur before the multiplication, rewrite the expression as (3+2)*5. Adding the parentheses makes the first three tokens a subexpression.

**Logical (boolean) operators**

REXX comparison expressions return a true (1) or false (0) value when processed. Logical operators combine two comparisons and return the true (1) or false (0) value depending on the results of the comparisons.

The logical operators are:

**Operator**

<table>
<thead>
<tr>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>&amp;</td>
</tr>
<tr>
<td>Returns 1 if both comparisons are true. For example:</td>
</tr>
<tr>
<td>(4 &gt; 2) &amp; (a = a)</td>
</tr>
<tr>
<td>(2 &gt; 4) &amp; (a = a)</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>Returns 1 if at least one comparison is true. For example:</td>
</tr>
<tr>
<td>(4 &gt; 2)</td>
</tr>
<tr>
<td>(2 &gt; 4)</td>
</tr>
<tr>
<td>&amp;&amp;</td>
</tr>
<tr>
<td>Returns 1 if only one comparison (but not both) is true. For example:</td>
</tr>
<tr>
<td>(4 &gt; 2) &amp;&amp; (5 = 3)</td>
</tr>
<tr>
<td>(4 &gt; 2) &amp;&amp; (5 = 5)</td>
</tr>
<tr>
<td>(2 &gt; 4) &amp;&amp; (5 = 3)</td>
</tr>
<tr>
<td>Prefix</td>
</tr>
<tr>
<td>Logical NOT</td>
</tr>
<tr>
<td>Returns the opposite response. For example:</td>
</tr>
<tr>
<td>¬ 0</td>
</tr>
<tr>
<td>¬ (4 &gt; 2)</td>
</tr>
</tbody>
</table>

Logical expressions are used in complex comparisons and can act as checkpoints to stop unwanted conditions (such as testing a field for a value of zero before using it as a divisor). If you have a series of logical expressions, for clarification, use one or more sets of parentheses to enclose each expression. For example:

(#46 = 999) | ((#45 > 0) & (#46 / #45) >= .5)

**Useful functions**

REXX provides a rich set of built-in functions, including character manipulation and conversion functions. Some of these functions might be of use when you are writing your comparison expressions. To call a function, type the function name directly followed by one or more arguments within parentheses. There can be no space between the function name and the left parenthesis. For example:

function(arguments)

A function call can contain up to 20 arguments separated by commas. Each argument can be:
Arguments

Example

Blank    function( )

Constant

function(55)

Symbol

function(#5)

Literal string

function('With a literal string')

Option recognized by function

function(option)

Another function

function(function( arguments ))

Combination of argument types

function('Literal string', #5, option)

Some of the built-in functions provided by REXX that you might find useful are:

```
ABS()
```

Returns the absolute value of a number. For example, to select records in which field #12 contains a value in the range -10 to +10, specify:

```
ABS(#12) <= 10
```

```
MAX()
```

Returns the largest number from the list specified. For example, to select records in which any of fields #10, #11, or #12 contains a value greater than 55, specify:

```
MAX(#10, #11, #12) > 55
```

```
MIN()
```

Returns the smallest number from the list specified. For example, to select records in which any of fields #10, #11, or #12 contains a value less than 0, specify:

```
MIN(#10, #11, #12) < 0
```

```
POS()
```

Copying data from a VSAM or QSAM file
Returns the position of one string, *needle*, in another, *haystack*. Returns 0 if *needle* is a null string, or is not found in *haystack*, or if *start* is greater than the length of *haystack*. By default, the search starts at the first byte of *haystack* (that is, the value of *start* is 1). You can override this by specifying *start* (which must be a positive whole number), the point at which the search starts. For example, to select records in which any character in field #22 is a blank, specify: POS(' ', #22) > 0

**SUBSTR()**

```plaintext
SUBSTR(string, n, length, pad)
```

Returns the substring of *string* that begins at the *n*th byte and is of length *length* bytes, padded with *pad* if necessary. *n* is a positive whole number. If *n* is greater than the length of *string* (in bytes), only pad characters are returned.

If you omit *length*, the rest of the string is returned. The default *pad* character is a blank.

For example, to select records in which bytes 4-6 of field #22 are the string 'NOT', specify:

```
SUBSTR(#22, 4, 3) = 'NOT'
```

**Performance tips**

Do not use EXITs in REXX

Use RETURN to terminate all your REXX processing. Do not use EXIT statements. EXIT statements in REXX used with FM/DB2 cause excessive CPU usage.

**Examples**

The following are examples of REXX comparison expressions:

**Example 1**

Select records in which the transaction date (field #14) is any date in July 2000, or the transaction value (field #27) is greater than $100,000.00

```
(#14 >= 20000701 & #14 < 20000801) | #27 > 100000.00
```

**Example 2**

Select records in which the count of credit notes (field #62) is greater than 10% of the count of invoices (field #61)

```
#62 > #61/10
```

**Example 3**

Select records in which either the transaction value (field #27) is greater than $50,000.00 and the purchase order number (field #25) starts with characters other than 'TX', or the transaction value is greater than $70,000 and the supplier number (field #23) is 984545, but not if both sets of conditions are true

```
(#27 > 50000.00 & SUBSTR(#25, 1, 2) ¬= 'TX') &
(#27 > 70000.00 & #23 = 984545)
```

**Enhancing the Import utility using a REXX procedure**

When you use the Import utility, you can supply a REXX procedure to enhance typical import FM/DB2 processing according to your own custom requirements.
Supplying a REXX procedure when using the Import utility

You use the **Use REXX proc** field on the Import “From” panel to supply the REXX procedure:

```
/* Use REXX proc member
```

If you enter an * (asterisk) as the member name, FM/DB2 displays an edit panel where you can enter a temporary procedure for one time use. Use this method only for short, ad hoc procedures that you do not want to keep for future use.

Alternatively, you can allocate an FMNEXEC DD that identifies a PDS where your saved REXX procedures reside. If you specify a member name, File Manager either edits a new member or runs with an existing member. Specifying blank or a pattern other than * displays a member selection list of the PDS allocated to FMNEXEC. Concatenated FMNEXEC data sets are not supported under ISPF.

For example, if you store your REXX procedures in a PDS called 'USERID.FMNEXEC' then, before using the **Use REXX proc** field, you must issue a TSO ALLOC command similar to this:

```
TSO ALLOC DD(FMNEXEC) DSN('USERID.FMNEXEC') SHR
```

In each case, FM/DB2 validates the REXX procedure that you supply. If errors are detected, FM/DB2 shows all detected errors in a message box and opens the procedure in an edit panel which you can use to correct it. Canceling the Edit session clears the **Use REXX proc** field and redisplay the entry panel.

How enhanced processing works

The REXX procedure that you supply is run against each input record in sequence, either until the end of the input data set is reached, or until the REXX procedure issues a RETURN STOP (or RETURN STOP IMMEDIATE) command.

FM/DB2 defines two special REXX variables, INREC and OUTREC, that you can use in the REXX procedure that you supply to perform enhanced processing. When the FM/DB2 function or panel calls the REXX procedure, the contents of each input record selected for processing are passed to the procedure in both INREC and OUTREC. When the procedure is called, the contents of INREC and OUTREC are identical.

The variable INREC is intended to be used as a reference variable (FM/DB2 ignores any changes to INREC). The variable OUTREC can be updated by the procedure, and (unless you “drop” the record from further processing, as described later in this section) when the REXX procedure has completed, is passed back for processing by the Import utility. For example, the following code changes a two-digit year in the output record to a four-digit year:

```
/* Changes a date of format MMDDYY (starting at column 1) to MMDDYYYY */
If fld(5,2,z) > 50
  outrec = fld(1,4)||'19'||fld(5)
else
  outrec = fld(1,4)||'20'||fld(5)
```

Using templates with enhanced processing: Only those records that the template selects are passed to the REXX procedure for further processing.

Regardless of whether or not the template you are using selects only certain fields, the INREC and OUTREC variables contain complete records with all fields.

Related tasks
Dropping a record from further processing: If the REXX procedure exits normally after processing a record, that record is considered to be “selected”, and the contents of OUTREC are used as the output record for importing. However, if the REXX procedure ends with a RETURN DROP (or RETURN STOP IMMEDIATE) instruction, the current record is “dropped” from further processing by FM/DB2, and is not imported.

External REXX functions
FM/DB2 provides the following external REXX functions that you can use in addition to the built-in REXX functions described in the OS/390 TSO/E REXX Reference.

CHANGE
Change a character string

CONTAINS
Check for character values

FLD
Refer to a field from the current input record

NCONTAIN
Check for numeric values

PRINT
Print a record

PRTCOUNT
Count of records printed

RECSIN
Count of records read

RECSOUT
Count of records written to a given data set

TALLY
Total a field value and report the total

TM
Test a string for a bit value

WRITE
Write a record

Note: You can only use these FM/DB2-specific REXX external functions, and the INREC and OUTREC variables, in a REXX procedure specified by the Use REXX proc field. You cannot use these functions and variables in REXX procedures outside of this FM/DB2 environment.

CHANGE:

```
  Change(string, old, new, new, count, start, length)
  Change(string, old, new, new)
```
Returns string, with up to count occurrences of the substring old changed to the substring new. The default value for count is 1. If you specify count, it must be zero or a positive whole number. If you specify zero, all occurrences of old in string are changed to new.

You can restrict the byte positions within string that are examined for substring old by specifying a starting byte position (start) and a length in bytes (length). The default value for start is 1. If you specify start, it must be a positive whole number. If start is greater than LENGTH(string), the CHANGE function has no effect. If you specify length, it must be zero or a positive whole number. If you omit length, or specify zero, the remainder of string from byte position start is examined. If length is less than LENGTH(old), the CHANGE function has no effect.

The lengths of substrings old and new can be different. If you omit old, the substring new is inserted in string at the byte position indicated by start. If you omit new, the substring old is deleted from string.

Here are some examples:

Example 1
CHANGE('abcabcabc','abc','DeF') → 'DeFabcabc'
/* 1 (default) occurrence of old changed */

Example 2
CHANGE('abcabcabc','abc','DeF',2) → 'DeFDeFabc'
/* 2 occurrences of old changed */

Example 3
CHANGE('abcabcabc','abc','DeF',0) → 'DeFDeFDeF'
/* count = 0, all occurrences of old changed */

Example 4
CHANGE('abcabcabc','abc','DeF',4) → 'abcDeFabc'
/* 1 (default) occurrences of old changed, */
/* starting at position 4 */

Example 5
CHANGE('aaaaaaaa','a','A',0,3,2) → 'aaaaAAA'
/* all occurrences of old changed, starting at */
/* position 3 for a length of 2 */

Example 6
CHANGE('abcabcabc','a',0) → 'bcbcbc'
/* new omitted, count = 0, */
/* all occurrences of old deleted */

Example 7
CHANGE('abc','def',2) → 'adefbc'
/* old omitted, new inserted, starting at */
/* position 2 */
If the haystack string contains one or more of the needle strings, CONTAINS returns 1. Otherwise, CONTAINS returns 0.

CONTAINS is case-sensitive: it only returns 1 if the haystack contains a string with the same mix of uppercase and lowercase as a needle.

Example 1

If the current input record contains “Michael”, “Mick” or “Mike” in the first ten columns, print the record.
If co(fld(1,10),'Michael','Mick','Mike') Then print(inrec, 'CHAR')

Example 2

If the current input record contains “USA”, “Australia” or “England”, drop the record from processing.
If contains(inrec,'USA','Australia','England') Then exit 'DROP'

Related tasks
“NCOUNT” on page 216

FLD:

Returns the value of a field from the current input record (INREC), starting at start_column (in bytes), of length number of bytes, interpreted according to the specified type:

B if the field is binary. If you specify B for type, length must be 2, 4, or 8.
C if the field contains characters.
P if the field is packed decimal. If you specify P for type, length must be between 1 and 16.
Z if the field is zoned decimal. If you specify Z for type, length must be between 1 and 32 or, if the field contains a separate sign character, between 1 and 33.

The default value for type is C.

If you omit length and specify type P (packed decimal), FLD attempts to determine the packed field length from the record data, and returns only that field. If you omit length for other field types, FLD returns the remainder of the record.

The FLD function is similar to the built-in REXX SUBSTR function, except that FLD interprets the “substring” according to the specified data type, and returns the value formatted appropriately. (For a numeric field, FLD returns the value with a sign, and without leading zeros.)

Example 1

If the value of the packed-decimal field that starts at byte 8 is greater than 100, do not process the current record.
if fld(8,P) > 100 then exit 'DROP'
Example 2

If the value of the 2-digit year field starting at byte 42 is greater than 60, insert the literal “19” before the year field. Otherwise, insert “20”.

```plaintext
if fld(42,2,Z) > 60 then
  outrec = fld(1,41)||'19'||fld(42)
else
  outrec = fld(1,41)||'20'||fld(42)
```

NCONTAIN:

```plaintext
NContains(number, match)
```

If the numeric value of any of the match arguments is equal to the numeric value of number, NCONTAIN returns 1. Otherwise, NCONTAIN returns 0.

Example 1

If the current record contains a packed decimal value of value of 10, 20, or 30 starting at byte 8, print the record.

```plaintext
if nco(fld(8,P),10,20,30) then print(inrec,'CHAR')
```

Example 2

If the current record contains a zoned decimal value of 11, 12, or 13 starting at byte 10, drop the record from processing.

```plaintext
if nco(fld(10,5,Z),11,12,13) then exit 'DROP'
```

Related tasks

“CONTAINS” on page 214

PRINT:

```plaintext
PRINT(record, format)
```

Prints the record string in the specified format: SNGL or TABL.

Note:
1. Only those fields that have been selected in the template are printed.
2. If the record has been reformatted by template processing, the variable INREC contains the input record value and OUTREC will contain the reformatted output record.

Example 1

Print the first hundred records.

```plaintext
if PRTCOUNT() < 100 then PRINT(inrec,'CHAR')
```

Example 2

Print the current input record in TABL format.
Copying data from a VSAM or QSAM file

rc = PRINT(inrec,'TABL')

Related tasks
Chapter 3, “Working with templates,” on page 49
"Selecting a display format” on page 118

PRTCOUNT:

PRTCOUNT()

Returns the current count of records printed.

Example

Print the first 10 input records.
if prtcnt() < 10 then print(inrec,'CHAR')

RECSIN:

RECSIN()

Returns the count of records read so far from the input data set, and passed to the REXX procedure. Only the input records that are selected by the template are passed to the REXX procedure and included in the count returned by RECSIN.

Example

Print every hundredth record.
if RECSIN()//100 = 0 then PRINT(inrec,'CHAR')

RECSOUT:

RECSOUT(ddname)

Returns the count of records so far written to the specified output data set.

The argument you can specify is:

ddname

Specifies that the count of records so far written to the data set identified by the specified ddname be returned.

The count of records written to an output data set is incremented each time a WRITE function is issued against the specified data set. In the case of the primary output data set, the count is also incremented each time a record is written to the data set by the FM/DB2 function. Unless a record is discarded using the RETURN DROP (or STOP IMMEDIATE) instruction, each record selected for processing is written to the primary output data set.

Example 1

If more than one hundred records have been written to the EXT100 file, stop FM/DB2 processing.
Copying data from a VSAM or QSAM file

rc = WRITE(EXT100)
if RECSOUT(EXT100) > 100 then exit "STOP"

Related tasks
"RETURN values" on page 220

TALLY:

Accumulates the value of the specified input record field in a TALLY register and, at the end of the Import utility, prints on SYSPRINT the TALLY register prefixed by the character string string. If the TALLY function is successful, it returns a value of 0. If it is unsuccessful, it raises the REXX syntax error condition.

The field whose value is to be accumulated starts at position start (in bytes) in the input record, and is length bytes long. If the sum of start and length is greater than LENGTH(INREC), the TALLY function returns a value of 0 without changing the TALLY register.

The data type of the field to be accumulated is specified by type. The values that can be specified for type are:

- **B** if the field is binary. If you specify B for type, length must be 2, 4, or 8. The field is assumed to be signed.
- **P** if the field is packed decimal. If you specify P for type, length must be between 1 and 16 bytes.
- **Z** if the field is zoned decimal. If you specify Z for type, length must be between 1 and 32 bytes or, if the field contains a separate sign character, between 1 and 33 bytes.

The default value for type is Z.

You can code more than one TALLY function in your procedure. FM/DB2 creates a separate TALLY register for each TALLY function with a unique combination of arguments. This means that you can accumulate a given field in more than one TALLY register by specifying a different value for string in each TALLY function.

Example

Accumulate hours recorded in personnel records depending on record type.

select;
    when(fld(1,1) = 'E') then
      rc = TALLY(15,4,B,'Sum of employee hours')
    when(fld(1,1) = 'S') then
      rc = TALLY(15,4,B,'Sum of supervisor hours')
    otherwise
      rc = TALLY(28,4,B,'Sum of manager hours')
end

TM:

Accumulates the value of the specified input record field in a TALLY register and, at the end of the Import utility, prints on SYSPRINT the TALLY register prefixed by the character string string. If the TALLY function is successful, it returns a value of 0. If it is unsuccessful, it raises the REXX syntax error condition.

The field whose value is to be accumulated starts at position start (in bytes) in the input record, and is length bytes long. If the sum of start and length is greater than LENGTH(INREC), the TALLY function returns a value of 0 without changing the TALLY register.

The data type of the field to be accumulated is specified by type. The values that can be specified for type are:

- **B** if the field is binary. If you specify B for type, length must be 2, 4, or 8. The field is assumed to be signed.
- **P** if the field is packed decimal. If you specify P for type, length must be between 1 and 16 bytes.
- **Z** if the field is zoned decimal. If you specify Z for type, length must be between 1 and 32 bytes or, if the field contains a separate sign character, between 1 and 33 bytes.

The default value for type is Z.

You can code more than one TALLY function in your procedure. FM/DB2 creates a separate TALLY register for each TALLY function with a unique combination of arguments. This means that you can accumulate a given field in more than one TALLY register by specifying a different value for string in each TALLY function.

Example

Accumulate hours recorded in personnel records depending on record type.

select;
    when(fld(1,1) = 'E') then
      rc = TALLY(15,4,B,'Sum of employee hours')
    when(fld(1,1) = 'S') then
      rc = TALLY(15,4,B,'Sum of supervisor hours')
    otherwise
      rc = TALLY(28,4,B,'Sum of manager hours')
end

TM:
Tests selected bits of a string and sets the condition code accordingly.

The length of the test is based on the length of the shorter of the two arguments, string and mask. A mask bit of one indicates that the equivalent bit in string is to be tested. If a mask bit is zero, the equivalent string bit is ignored. If the tested bits are all ones, TM returns 1. Otherwise, TM returns 0.

Example 1

Test the third byte of the input record and if the low order bit is set, overlay a hex FF into the second byte of that record.

```rxml
if TM(fld(3,1),'01'x) then do
  outrec = OVERLAY('FF'x,outrec,2)
exit
end
exit drop
```

Example 2

Test the third byte of the input record and if the high order bit is set, logically OR a hex 04 over the contents of the second byte of that record.

```rxml
if TM(fld(3,1),'10000000'b) then do
  outrec = OVERLAY(BITOR(fld(2,1),'04'x),outrec,2)
exit
end
exit drop
```

WRITE:

```
WRITE(ddname)
```

Writes a record to the specified data sets. If the WRITE function is successful, it returns a value of 0. If the WRITE function is unsuccessful, it raises the REXX syntax error condition.

The argument you can specify is:

**ddname**

Specifies a record is to be written to the data set identified by the specified ddname.

The record is written from the contents of one of the following REXX variables:

- If it has been assigned a value by the procedure, the variable OUTREC.ddname, where ddname is the ddname specified in the WRITE function
- If the variable OUTREC.ddname has not been assigned a value, or has been unassigned using the DROP instruction, the FM/DB2-defined variable, OUTREC

The length of the record written depends upon the data set attributes of the output data set. If the output data set contains variable-length records, the length of the record is determined from the length of the data in the REXX variable. If the length of the data is greater than the maximum record length specified in the data set attributes, the record is truncated. If the output data set contains fixed-length
Copying data from a VSAM or QSAM file

records, the length of the record written is the length specified in the data set attributes, truncated or padded as necessary.

If the record format of the output data set specifies that the records contain a carriage control character, depending on the output device, the first character of the record data in the REXX variable is interpreted as a carriage control character. For more information about records containing carriage control characters, see OS/390 DFSMS: Using Data Sets.

Example 1

If the current record is type 01, write it to the DD01 file.
if fld(1,2) = 01 then WRITE('DD01')

Example 1

If the current record is type 02, write it to the DD02 and DD02COPY files.
if fld(1,2) = 02 then WRITE('DD02','DD02COPY')

RETURN values
In REXX, you can use the RETURN instruction to leave a procedure. You can optionally specify a character string as a parameter on the RETURN instruction. This character string is returned to the caller of the procedure. File Manager recognizes and processes the following character strings if specified on the RETURN instruction. The character strings can be specified in uppercase, lowercase, or a mixture of both.

DROP: The character string DROP tells FM/DB2 to not write the current record to the primary output data set. Specifying DROP on the RETURN instruction results in the current record not being imported to the target table.

Here are some examples:
RETURN 'DROP'
Return 'Drop'
Return drop /* assumes no variable named 'drop' has been assigned */

STOP: The character string STOP tells FM/DB2 to stop the current FM/DB2 function after the current record has been written to the primary output data set. Specifying STOP on the RETURN instruction results in the current record being imported to the target table and the Import utility being stopped.

Here are some examples:
RETURN 'STOP'
Return 'Stop'
Return stop /* assumes no variable named 'stop' has been assigned */

STOP IMMEDIATE: The character string STOP IMMEDIATE tells FM/DB2 to stop the current FM/DB2 function without writing the current record to the primary output data set. Specifying STOP IMMEDIATE on the RETURN instruction results in the current record not being imported to the target table and the Import utility being stopped.

Here are some examples:
RETURN 'STOP IMMEDIATE'
Return 'Stop Immediate'
Return stop immediate /* assumes variables 'stop' and 'immediate' not assigned */
Copying data to a VSAM or QSAM file

To copy data from a DB2 table or view to a partitioned, sequential, or VSAM data set, use the Export utility function (3.7).

Note:
1. To copy data between DB2 tables, use the Copy utility function (3.3).
2. To copy data to a DB2 table or view from a QSAM or VSAM data set, use the Import utility function (3.6).
3. To copy data to a sequential file using the DB2 UNLOAD utility, use the DB2 Utilities option (3.9). When copying large tables, the use of the DB2 UNLOAD utility is preferable (for reasons of performance and efficiency).

The Export utility generates the export data using DB2 SQL. The data can be exported in any of these formats:
- DB2 (File Manager internal) format. This format is not the same as the data formats used by the DSNTIAUL sample job and the DB2 utilities. This is the recommended option if the exported data is imported into another DB2 table using the FM/DB2 Import utility.
- DB2 external (or unload) format. This is the same format used by the DB2 UNLOAD utility.
- DB2 DSNTIAUL unload format. This is the format used by the DB2 DSNTIAUL sample program.
- A user defined format. Note that the user defined Placement: 3. User defined is not supported by the Import utility.
- Delimited (CSV) format. The exported data format is not supported by the Import utility.
- Copybook layout format.

You can use the output data set generated by the Export Utility for input to:
1. The FM/DB2 Import utility. This utility supports data in DB2 format and copybook formats.
2. A PC-based spreadsheet application program in the form of a comma-delimited (CSV) file.
3. Other user application programs.

If you export data, you can:
- Select the format in which the data is exported.
- Select the columns to be exported by selecting the required columns in the "from" template.
- Reformat data during export by mapping table columns to fields in the output partitioned, sequential, or VSAM data set (as defined by a template).
- Create and save a template that describes the exported data.
- Select which rows to export.
- Specify the number of rows that are exported.
- Specify whether you want to run the export in foreground or batch.

To display the Export Utility panel,
1. From the Primary Option Menu panel, enter 3 (Utilities) to display the Utility Functions panel.
2. From the Utility Functions panel, enter 7 (Export)
Specifying details for the “From” table

Specify the name of the source (“From”) table on the Export Utility panel. If the table you specify does not exist, FM/DB2 issues an error message and stops the utility.

If you do not want to export all of the rows in the “From” table, specify the number of rows you want to export (1–99999999) in Export Count.

If you are using a specific template for the “From” table (see option 1 in Template usage), specify the name of the template in the From Template entry fields.

The Processing Options on the Export Utility panel specify which template you want to use for the “From” table, and whether you want to edit the template before the export process.

Template usage

The Export Utility panel provides four options for template processing:

1. Above
   Requires that you enter the name of a template data set (and optionally a member name) in the From Template section of the panel. If you do not specify a member name, or if you specify a pattern, FM/DB2 displays a member selection list. After you have specified the member containing a template, processing continues using this template in place of the automatically-generated template for the table (see Template usage option 3).

2. Previous
   Uses the last (previously used) template for this table.

3. Generate from table
   Generates a template based on the information in the DB2 catalog for the specified table. This is the default setting. If you have specified a member name in the From Template section of the panel, it is ignored. FM/DB2 generates the required template in memory.

4. Generate/Replace
   Generates a template based on the information in the DB2 catalog for the specified table, and saves this information in the member specified in the From Template section of the panel. If the member already exists, it is replaced. Any prior customization is lost.

You can select the following options by entering a “/” or an “A”:

Edit options
   Use to display the first of the Export Options panels.
Copy data to a VSAM or QSAM file

**Edit template**

Use to edit the template for the “From” table before proceeding to the Export Utility “To” panel.

To proceed, press Enter.

If you selected **Edit options** (either by using a “/”, or by having previously used the “A” selection character for this option), the first of the Export Options panels is displayed.

If you selected **Edit template** (either by using a “/”, or by having previously used the “A” selection character for this option), the Column Selection/Edit panel is displayed.

Otherwise, the Export Utility “To” panel is displayed.

**Related tasks**

- “Specifying a DB2 object name” on page 25
- “Specifying a data set and a member name” on page 30
- “Selecting options on FM/DB2 panels” on page 24

Chapter 3, “Working with templates,” on page 49
- “Editing the template for the “From” table”
- “Specifying details for the “To” data set” on page 224

**Related references**

- “Export Utility panel” on page 567

**Setting options for the current Export session**

If you select **Edit options** on the Export Utility panel (or the Export Utility “To” panel), FM/DB2 displays the first of up to three Export Options panels. The remaining panels are displayed by pressing the NxtPage function key (F11).

**Note:**

1. To navigate to the Export Options (2 of 3) panel, you must select one of the data format options 4. **User defined** or 5. **Delimited variables (CSV), and Null indicators / CSV options** on the Export Options (1 of 3) panel.
2. To navigate to the Export Options (3 of 3) panel, you must select the data format option 4. **User defined and Data type format** on the Export Options (1 of 3) panel.

Use the Export Options panels to change the options for the current Export session.

**Note:** Options apply for the current session of FM/DB2 and only revert to global settings when the session is terminated or global settings are invoked (from the pull down or from the menu).

**Related references**

- “Export Utility panel” on page 567
- “Export Options (1 of 3) panel” on page 558

**Editing the template for the “From” table**

If you select the **Edit template** option on the Export Utility panel, FM/DB2 displays the Column Selection/Edit panel. You can use this panel to restrict the data in the “From” table that you want exported by:
Selecting (or deselecting) specific columns for exporting.

Specifying row selection criteria. Only rows whose contents satisfy the criteria are selected for exporting.

Note: When you export data in FM/DB2 internal (SQLDA) format, you cannot alter the order in which the columns are exported. Data is always exported in the order that the columns are defined in the DB2 catalog (the same order in which the columns are displayed in the Column/Selection Edit panel).

The Export Utility ignores any information specified in the Seq field of the template (this information is used to change the order in which FM/DB2 displays or prints columns).

For example, the template shown in Figure 51 has been edited to select certain columns and only rows whose contents meet certain criteria. If you use this template as the “From” template for your Export, the data exported is restricted to:

- The columns FIRSTNME, LASTNAME, WORKDEPT, and JOB, but only where WORKDEPT equals “MNT” and JOB equals “PAINTER”.

**Related tasks**

Chapter 3, “Working with templates,” on page 49

“Changing the sequence of displayed or printed data” on page 70

**Related references**

“Export Utility panel” on page 567

“Column Selection/Edit panel” on page 407

**Specifying details for the “To” data set**

Type the “To” data set details. Use a combination of the Data set name, Member, and Volume entry fields to specify the partitioned, sequential, or VSAM data set to which you want to export the data. You can specify the name of an existing data set, or the name of a data set that you want the Export utility function to create.
Copying data to a VSAM or QSAM file

If you are using a specific template or copybook for the “To” data set, specify the name of the template or copybook data set and, optionally, the member name, in the To Copybook or Template entry fields.

Use the Processing Options to select a number of options for the export process.

To proceed, press Enter.

If you selected Edit options (either by using a “/”, or by having previously used the “A” selection character for this option), the second of the Export Options panels is displayed.

If you selected Edit template mapping (either by using a “/”, or by having previously used the “A” selection character for this option), the Template Mapping panel is displayed for editing.

If you selected Batch execution (either by using a “/”, or by having previously used the “A” selection character for this option), the generated JCL is displayed using the ISPF editor, where you can review the JCL and submit the job for batch processing.

If you selected options 3. Batch, using DB2 utility and Edit UNLOAD options on the Export Options - (1 of 3) panel, before the generated JCL is displayed using the ISPF editor, the UNLOAD Options panel is displayed.

Otherwise, FM/DB2 exports the data from the “From” table to the “To” partitioned, sequential, or VSAM data set.

Related tasks

"Specifying a data set and a member name” on page 30
"Mapping data” on page 80
"Exported data formats”
"Running Export in batch” on page 227
"Notes on using Export” on page 228

Related references

"Export “To” panel” on page 564
"Template Mapping panel” on page 710

Exported data formats

You can export data in four pre-defined formats:

• DB2 (File Manager internal) format
• DB2 external (or unload) format
• DB2 DSNTIAUL unload format
• Comma-delimited format (CSV output)

and two formats that require user definition:

• A user-defined format
• Copybook layout format

The output format depends on which Data format option you selected on the Export Options - (2 of 3) panel:

To export data in DB2 (File Manager internal) format, specify option 1 (FM/DB2 (SQLDA) format).
Copying data to a VSAM or QSAM file

To export data in DB2 external (or unload) format, specify option 2 (DB2 unload format).

To export data in DB2 DSNTIAUL unload format, specify option 3 (DSNTIAUL format).

To export data in user-defined format, specify option 4 (User defined).

To export data in comma-delimited format (CSV output), specify option 5 (Delimited Variables (CSV)).

The output format depends on which Template usage processing option you select.

To export data in DB2 (File Manager internal) format, specify Template usage option 3 (Generate from input) or option 4 (Generate and save) and export option 1 FM/DB2 (SQLDA) format. When you specify these options, FM/DB2 copies the output template from the input template, and exports the data in the format returned by DB2 if an SQLDA is used to access the result table. In this format, any nullable column is preceded by a 2-byte binary field containing the null indicator. If the “From” column has a null value, the null indicator field for the corresponding “To” field contains -1, otherwise it contains 0. Each decimal field in the output data set occupies 16 bytes, with the data left-aligned followed by trailing binary zeros. Any graphic or vargraphic field is followed by two bytes containing binary zeros. Use this option if you plan to import the exported data into another DB2 table using Import (option 3.6).

To export data in DB2 unload format, specify Template usage option 3 (Generate from input) or option 4 (Generate and save) and select export option 2 DB2 unload format. The exported data is in the same format as that returned by the DB2 UNLOAD utility, with the NOPAD option not specified. In this format, any nullable column is preceded by a 1-byte binary field containing the null indicator. If the “From” column has a null value, the null indicator field for the corresponding “To” field contains X’FF’, otherwise it contains 0. Any decimal field occupies the minimum number of bytes needed to represent the data as a packed decimal field. Variable length columns in the unloaded records are padded to their maximum length and the padded data fields are preceded by length fields that indicate the size of the actual data without the padding. CCSID conversion is not supported. The exported data will be in the CCSID used to bind the FM/DB2 plan.

To export data in DSNTIAUL format, specify Template usage option 3 (Generate from input) or option 4 (Generate and save), and select export option 3. DSNTIAUL format. In this format, any nullable column is followed by a 1-byte binary field containing the null indicator. If the "From" column has a null value, the null indicator field for the corresponding "To" field contains "?", otherwise it contains 0. Any decimal field occupies the minimum number of bytes needed to represent the data as a packed decimal field.

To export data in comma-delimited format (CSV output), specify Template usage option 5 (None. CSV output). If you specify this option, FM/DB2 does not generate an output template but writes the exported data as varying-length records, separated by a delimiter. For delimited format, FM/DB2 uses the Null column display indicator to indicate that data for a nullable column is the null value.
To export data in user-defined format, specify Template usage option 3 (Generate from input) or option 4 (Generate and save), and select export option 4. User defined format. For a user-defined format, FM/DB2 provides the following options for nullable columns:

- To use or not use nulls. In the latter case, there would be no fields in the exported data corresponding to the null indicator. Set Null Indicators Usage to 2. None on Export Options screen.

- To place the null indicator either before or after the data for the column to which it applies. Set Null Indicators Placement to 1. Before or 2. After on the Export Options screen.

- To use either a 1-byte (character format) or 2-byte (signed integer format) for the null indicator, set Null Indicators Type to 1. One byte or 2. Two byte on the Export Options screen. Set the value to be used in Indicator Char or Integer.

To export data in user-defined format, and to place either a 1-byte (character format) or a 2-byte (signed integer format) null indicator somewhere in the output record, but not before or after the data for the column to which it applies, select Null Indicators Placement to 3. User defined on the Export Options - (2 of 3) panel. You must provide a copybook and specify Template usage option 1 (Above) or 2 (Previous), and select Edit template mapping and map the null indicators. If the field names in the copybook are not the same as the DB2 column names, you will also need to map each DB2 column to the appropriate field in the copybook.

To export data in any other format, you must use a user-defined template. Specify Template usage option 1 (Above) to use the template you have specified in the To Copybook or Template section of the panel, or Template usage option 2 (Previous) to use the template last used for the table from which you are exporting data. If exporting columns that permit null values, FM/DB2 processes them as follows: if the field that corresponds to a nullable column is followed by an unmapped 2-byte binary field, FM/DB2 maps the null indicator for the column to this 2-byte binary field. In all other situations, FM/DB2 does not map the null indicator to any field in the output template.

Related references

“Editor Options (4 of 8) panel” on page 540

Running Export in batch

If you want to export large DB2 tables, you can perform the export in a batch job.

If you select the Batch execution option on the Export Options (1 of 3) panel, the Export utility builds the necessary JCL to perform the export function in a batch job. The generated JCL is displayed using the ISPF editor, where you can review the JCL and submit the job for batch processing. If you select the Batch, using DB2 utility option, the JCL is set up for the DB2 UNLOAD utility, and displayed for review before submission.

If you have one or more data sets that contain the template that describes the record structure of your input data, you can specify the data set details (or TSO ALLOC statement) in the generated JCL with the keywords TINPUT or TCIN.

Similarly, if you have one or more data sets that contain the template that describes the record structure of your output data, you can specify the data set details (or TSO ALLOC statement) in the generated JCL with the keywords TOUTPUT or TCOUT.
Copying data to a VSAM or QSAM file

If the export data set does not already exist, you are prompted to create it. However, you can defer the creation of the export data set until batch job execution time by selecting the **Batch data set creation** option. If you select this option, FM/DB2 creates a new sequential export data set in the JCL for the batch job.

Related references

“DBX (Export) batch command” on page 800

Notes on using Export

The Export utility function provides one method for moving DB2 data between different DB2 sub-systems (that is, to export data from one DB2 sub-system and import it to another DB2 sub-system). If you want to move data between tables in the same DB2 sub-system, you might find it more convenient to use Copy (3.3), or move the data by direct execution of SQL statements.

In general terms, the Export utility function is suitable for extracting small to medium volumes of data from DB2 tables. It is not intended to serve as a substitute for the DB2 utilities if unloading or migrating large volumes of data. The DB2 utilities offer superior performance in these situations.

It is recommended that you run the Export utility in batch whenever possible.

The Export utility provides for automatic mapping of the DB2 null indicator. However, if Export Option Data format 4. User defined Null Indicators Placement 3. User defined is selected the template editor shows the null indicator as a separate item in the “From” template on the Field Mapping panel. It is therefore possible to re-map the null indicator associated with nullable columns to a field in the export data set. If you want to rearrange the data fields in the export data set, it is recommended that you first export the data using the default mapping and then use File Manager (base) to map the data to the required format.

A sample REXX exec is provided (in SFMNSAMP) which converts a FM/DB2 template generated by FM/DB2 into a COBOL or PL/I copybook.

To convert a FM/DB2 template generated by FM/DB2 into a COBOL or PL/I copybook, you specify a TSO command like the following:

```
FMN2CPYB input output language
```

where:

input The name of a sequential file or partitioned data set and member that contains a File Manager DB2 template.

output The name of a sequential file or partitioned data set and member that will contain the copybook.

language Optional keyword that determines the type of copybook generated. Valid values are:

- COBOL
- PL/I

If not specified, a COBOL copybook is generated.
Copying data to a VSAM or QSAM file

The sample REXX must be copied to a library in your SYSPROC or SYSEXEC concatenation. Alternatively, you can specify the library name containing the sample exec in the command.

Related tasks
- “Copying data from one DB2 object to another” on page 193
- Chapter 11, “Generating batch JCL for DB2 utility jobs,” on page 277
- Chapter 3, “Working with templates,” on page 49

Copying data from an FM/DB2 editor session, or from an SQL statement

Use the EXPORT primary command to write data from the current FM/DB2 editor session to an external file. Alternatively, you can re-execute the SQL statement that generated the data for the current FM/DB2 editor session and write rows for this select statement to an external file. For the second option, you can generate a JCL deck to perform the export operation using batch processing.

Data can be exported in the same data formats as for the Export utility (3.7).

During the export process, the Export Utility panel is optionally displayed. You enter the data format for the exported data on this panel.

The Export “To” panel is also optionally displayed. The Export “To” panel enables you to:
  • Specify the name of the data set that will contain the exported data.
  • Specify the name of a data set that contains a template or copybook, that describes the exported data.
  • Specify a processing option for the template data set.
  • Specify a disposition option for the export data set.
  • View and possibly change the export options.
  • Use the template editor to display and possibly change the mapping between the source and data to be exported.

Related references
- “Export Utility panel” on page 567
- “Export “To” panel” on page 564
- “Specifying details for the “From” data set” on page 200
- “Specifying details for the “To” table” on page 196
- “EXPORT primary command” on page 745

Specifying the source for the exported data

You can specify the source for the exported data with the ALL, NX, X, or EX parameter to the EXPORT primary command.

When you specify this parameter, FM/DB2 copies all, or selected, rows in the current FM/DB2 editor session to the target data set.

If you omit this parameter, FM/DB2 re-executes the SQL statement that produced the data for the current FM/DB2 editor session. FM/DB2 exports the rows fetched by the SQL statement to the target data set, up to any row limit specified on the export options panel.

You can use the Export Count field on the Export Utility panel to limit the data loaded into the FM/DB2 editor, for large DB2 objects, or arbitrary select statements.
that generate large result tables. Issuing the EXPORT primary command without
the ALL parameter enables all rows for the DB2 object, or result table from the
arbitrary select statement, to be written to the export data set.

Related references

“EXPORT primary command” on page 745
“Export Utility panel” on page 567

Specifying the format of the exported data
You can export data in a number of data formats (see “Exported data formats” on
page 225).

You can specify the data format for the exported data as an optional parameter to
the EXPORT primary command. When specified, the parameter overrides the
current data format, as displayed on the Export Options (1 of 3) panel.

The possible parameter values are SQLDA, DB2, DSNTIAUL, USER, and CSV.

Related references

“EXPORT primary command” on page 745
“Export Options (1 of 3) panel” on page 558

Limiting the rows exported with a label range
You can limit the rows processed to only those rows within a specified label range,
rather than all rows in the FM/DB2 editor session.

Note: You can only specify a label range if you have also specified the source
parameter (ALL, NX, X, or EX).

You must specify a "from" label and a "to" label.

Related references

“Assigning labels to rows” on page 116
“EXPORT primary command” on page 745

Differences compared with the export utility
The process for exporting data with the EXPORT primary command is similar to
that for the Export utility function (3.7) with the following exceptions:

• The Batch, using DB2 UNLOAD execution option is not available when
  exporting data using the EXPORT primary command.

• The dialog commences with the optional display of the Export Options (1 of 3)
  panel. The Export Utility panel is not displayed.

• The data to be exported can be the result table for the SQL statement that
generated the current FM/DB2 editor session, or rows from the current FM/DB2
editor session. In the former case, the SQL statement can refer to a DB2 object
(for example, when entered using the Edit function), or an arbitrary SQL
statement (for example, when entered using one of the SQL options 4.1, 4.2, 4.3
or 4.4). The arbitrary SQL statement can be any valid select statement, including
joins of multiple tables, sub-selects, and so on. By contrast, the Export utility
only allows export of data from a DB2 object (for example, table, view, alias).

• When using the EXPORT primary command to export data from an arbitrary
  SELECT statement (for example, entered using one of the FM/DB2 functions 4.1,
  4.2, 4.3 or 4.4) in batch, there are some restrictions on the template mapping
  operations that are supported.
Copying data from an FM/DB2 editor session, or from an SQL statement

It is not possible to use the template used to describe the SELECT statement in the editor session in the generated JCL deck.

For export in CSV format, which cannot be described using a template, this means any changes made to the template for the editor session are discarded when exporting the data - the output will include data for every column.

For export in other supported formats:
- Use the TE command in the editor session to select or de-select the columns as required.
- Issue the EXPORT BAT command to generate a JCL deck.
- You are required to save the output template that describes the data to be exported. This is included as a parameter in the JCL deck.
- When the JCL deck is submitted for processing, FM/DB2 re-creates a temporary template for the arbitrary SQL statement specified in the JCL deck, and then uses template mapping to identify which columns should be exported - these are the columns defined in the output template.

- The export options panel includes an input field where the number of rows to be exported can be entered. This operates independently of any input field that limits the number of rows loaded into the FM/DB2 editor session. It is therefore possible to edit a few rows of a result table using FM/DB2 edit, and then export all the rows for that result table to a data set.

Related references
- “Export Options (1 of 3) panel” on page 558
- “Export Utility panel” on page 567
- “EXPORT primary command” on page 745

Ways that you can use the EXPORT primary command

Here are two ways you can use the EXPORT primary command:

- To export data from an arbitrary SQL statement to an external file:
  1. Develop the SQL SELECT statement using options 4.1, 4.2, 4.3, or 4.4. The SQL statement can be any valid SQL statement, including joins of multiple tables, sub-selects, and so on.
  2. Use the row count field to limit the rows loaded into the FM/DB2 editor, if the result table is large.
  3. Confirm that the result table is correct by examining the data in the Edit session.
  4. Optionally edit the template and de-select any columns of the result table that are not required.
  5. Issue the EXPORT command without the ALL parameter, to export all rows from the result table of the arbitrary SQL statement to an external file. You can also issue EXPORT BAT to generate a JCL deck to export the rows from the arbitrary SQL statement to an external file. If required you can take a copy of the generated JCL for later re-use.

- To export selected editor session rows to an external file:
  1. Ensure that the Use edit (instead of browse) to display results option is selected. This option is set using the FM/DB2 System Options (2 of 3) panel.
  2. Develop the SQL SELECT statement using options 4.1, 4.2, 4.3 or 4.4, or edit the required DB2 object using the FM/DB2 Edit utility.
  3. Use editor commands to isolate the required data in either excluded, not excluded rows, label ranges, or some combination of these methods.
4. Issue the EXPORT primary command with the OPT parameter and one of ALL, X, or NX. Ensure that the required data format is selected. On the Export "To" panel, enter the required export data set name and set the disposition option to MOD.

5. After the initial export of rows to the export data set, isolate additional rows to be exported, then issue EXPORT QUIET plus a combination of ALL, X, or NX and optionally a label range, to export the additional rows. The MOD setting for the export file ensures that each additional EXPORT primary command adds the newly-exported rows to the end of the export data set.

Scrambling data

Scrambling data allows you to create test data based on production (or "live") data, but with the ability to change the values of certain columns. In this way, you can avoid sensitive or confidential information appearing in test data.

When can you scramble data?

When you copy data from one place to another with either the Copy Utility (option 3.3), Import Utility (option 3.6), or Export Utility (option 3.7), you can choose to scramble some or all of the columns being copied.

For scrambling to occur during the copy, import, or export process, you must supply an output template which has mapped fields marked for scrambling.

Scrambling rules

FM/DB2 scrambles data according to these rules:

- If you do not specify any value or range options:
  - Uppercase alphabetic characters are scrambled to other uppercase alphabetic characters.
  - Lowercase alphabetic characters are scrambled to other lowercase alphabetic characters.
  - DBCS characters are scrambled to other DBCS characters.
  - Numerics are scrambled to other numerics.
  - Any other characters remain unchanged.
  - Repeatable scrambling produces unique results for numeric fields defined with Leading zeros set to YES, and all non-numeric fields.

- If you provide a value list, then the column is populated with a value from the list.

- If you provide a range, then the resultant number is in the range provided.

- If you specify a scramble type of random or repeatable and also the value option and a value data set, then the column is populated with a value from the data set.

- If you specify a scramble type of translate, then the input column value is matched and the corresponding output value is used from the value data set.

Related topics

“Specifying scrambling options” on page 76
“Column Attributes panel (alphanumeric)” on page 397
“Column Attributes panel (numeric)” on page 402
“Value List Edit panel” on page 722
“Scramble Exit Specification panel” on page 658
If you want to copy data from one place to another, but are not sure of the best way of going about it, you may find the following sections helpful.

The sections below describe typical situations that you may encounter when planning to copy data. For each scenario, step-by-step instructions are included describing how to achieve the required result.

Note: In the following examples, “source table” refers to the table from which the data is exported, and “target table” refers to the table to which the data is imported.

**How do I copy data from one DB2 table to another on the same DB2 system?**

Scenario: I have data in a DB2 table and want to copy it to another DB2 table. The source and target DB2 tables are on the same DB2 system.

Consider using the FM/DB2 Copy Utility (3.9), instead of the FM/DB2 Export Utility and the FM/DB2 Import Utility. If the source and target table have different column names, different numbers of columns, or if data type conversions are needed, you can use template re-mapping to map columns in the source table to columns in the target table. In most cases, FM/DB2 performs any data type conversions automatically.

Related tasks
- “Copying data from one DB2 object to another” on page 193
- “Column mapping rules” on page 86

**How do I copy data from one DB2 table to another on a different (connectable) DB2 system?**

Scenario: I have data in a DB2 table and want to copy it to another DB2 table. The source and target DB2 tables are on different DB2 systems. It is possible to connect to the target DB2 system from the source DB2 system by specifying a location value.

Consider using the FM/DB2 Copy Utility (3.9), instead of the FM/DB2 Export Utility and the FM/DB2 Import Utility. See “How do I copy data from one DB2 table to another on the same DB2 system?” If a large number of rows are to be transferred, some caution needs to be exercised due to network considerations.

Related tasks
- “Copying data from one DB2 object to another” on page 193
- “Column mapping rules” on page 86

**How do I copy data from one DB2 table to another on a different (non-connectable) DB2 system?**

Scenario: I have data in a DB2 table and want to copy it to another DB2 table. The source and target DB2 tables are on different DB2 systems. It is not possible to connect to the target DB2 system from the source DB2 system by specifying a location value, or I want to transfer the data out of DB2 as part of the copy process.
How do I...?

To copy the data, you need to perform two steps:

1. Use the FM/DB2 Export Utility (3.7) to export the data to a sequential file (preferably) or a VSAM file. In general, the default mapping is sufficient. To use the default mapping, specify a template data set and member name on the Export “To” panel and select Template usage option 4 (Generate and save). If the amount of data to be transferred is large (more than 10,000 rows), run the export in batch by selecting Batch execution. If the amount of data to be transferred is very large (more than 100,000–1,000,000 rows), consider using the DB2 UNLOAD utility (accessed from Utilities, 3.9) instead.

2. Use the FM/DB2 Import Utility (3.6) to load the exported data into the target DB2 table. If the target table has different column names to the source table, or if data type conversions are required, use template re-mapping to map columns in the source table to columns in the target table. In most cases, FM/DB2 performs any data type conversions automatically.

Related tasks
- “Copying data to a VSAM or QSAM file” on page 221
- “Running Export in batch” on page 227
- “Copying data from a VSAM or QSAM file” on page 198
- “UNLOAD utility” on page 306
- “Column mapping rules” on page 86

Related references
- “Export “To” panel” on page 564

How do I copy and reorder selected data from a DB2 table to a sequential file?

Scenario: I have data in a DB2 table and I want to copy it to a sequential file. I only want the data for some of the columns in the DB2 table, and I want the data for those columns reordered in the sequential data set.

The FM/DB2 Export Utility (3.7) supports the export of selected columns from a DB2 object. To export selected columns only, edit the template on the Export Utility panel and deselect the columns that are not required.

The FM/DB2 Export Utility does not support the reordering of columns in the export data set. The selected columns are always exported in the order in which they are defined in the DB2 catalog. There is a simple workaround for this. Create a view for the DB2 object to be exported, specifying the columns in the desired order. Then export data from the view, to achieve the desired result.

Related tasks
- “Copying data to a VSAM or QSAM file” on page 221
- “Mapping data” on page 80

Related references
- “Export Utility panel” on page 567

How do I define a copybook that describes data exported to a sequential file?

Scenario: I have data in a DB2 table and I've used the FM/DB2 Export Utility to copy the data to a sequential file. I need to define a copybook that describes the exported data.
You need to code the copybook in either COBOL, HLASM, or PL/I. The amount of space required in the export data set for each of the DB2 data types is summarized in the following example:

- Suppose you want to export data from the table created using the following definition:

```sql
CREATE TABLE SAMPLE.EXPORT_EX
(COL1 INTEGER NOT NULL,
 COL2 SMALLINT NOT NULL,
 COL3 FLOAT(21) NOT NULL,
 COL4 FLOAT(53) NOT NULL,
 COL5 DECIMAL(1,0) NOT NULL,
 COL6 DECIMAL(15,7) NOT NULL,
 COL7 CHAR(10) NOT NULL,
 COL8 VARCHAR(10) NOT NULL,
 COL9 DATE NOT NULL,
 COL10 TIME NOT NULL,
 COL11 TIMESTAMP NOT NULL,
 COL12 ROWID GENERATED ALWAYS NOT NULL)
```

A COBOL copybook that describes the exported data is as follows:

```cobol
01 EXPORT-EXAMPLE.
   * INTEGER DATA TYPE
   10 COL1 PIC S9(6) USAGE COMP.
   * SMALLINT DATA TYPE
   10 COL2 PIC S9(4) USAGE COMP.
   * FLOATING POINT (4 BYTES) DATA TYPE
   10 COL3 USAGE COMP-1.
   * FLOATING POINT (8 BYTES) DATA TYPE
   10 COL4 USAGE COMP-2.
   * PACKED DECIMAL DATA TYPE
   10 COL5 PIC S9(1)V USAGE COMP-3.
   10 FILLER PIC X(15).
   * PACKED DECIMAL DATA TYPE
   10 COL6 PIC S9(8)V9(7) USAGE COMP-3.
   10 FILLER PIC X(8).
   * CHARACTER DATA TYPE
   10 COL7 PIC X(10).
   * VARCHAR DATA TYPE
   10 COL8.
       49 COL8-LEN PIC S9(4) USAGE COMP.
       49 COL8-TEXT PIC X(10).
   * DATE DATA TYPEN
   10 COL9 PIC X(10).
   * TIME DATA TYPE
   10 COL10 PIC X(8).
   * TIMESTAMP DATA TYPE
   10 COL11 PIC X(26).
   * ROWID DATA TYPE
   10 COL12.
       49 COL12-LEN PIC S9(4) USAGE COMP.
       49 COL12-TEXT PIC X(40).
```

In HLASM, the copybook looks like this:

```asm
EXAMPLE DSECT
   * Integer data type
   COL1 DS F
   * Smallint data type
   COL2 DS H
   * Floating point (4 bytes) data type
   COL3 DS E
   * Floating point (8 bytes) data type
   COL4 DS D
   * Packed decimal data type (1 byte)
```
How do I...?

COL5  DS   PL1
* Packed decimal data type (8 bytes)
COL6  DS   PL8
* Character data type
COL7  DS   CL10
* VARCHAR data type
COL8_LEN DS   H
COL8_TXT DS   CL10
* DATE data type
COL9  DS   CL10
* TIME data type
COL10 DS   CL8
* TIMESTAMP data type
COL11 DS   CL26
* ROWID data type
COL12_LEN DS   H
COL12_TXT DS   CL40
  END

In PL/I, the copybook looks like this:

DCL 1 EXPORT_EXAMPLE UNALIGNED,
  /* INTEGER DATA TYPE */
  2 COL1 BIN(31,0),
  /* SMALLINT DATA TYPE */
  2 COL2 BIN(15,0),
  /* FLOATING POINT (4 BYTES) DATA TYPE */
  2 COL3 FLOAT,
  /* FLOATING POINT (8 BYTES) DATA TYPE */
  2 COL4 FLOAT(8),
  /* PACKED DECIMAL DATA TYPE */
  2 COL5 FIXED DEC(1,0),
  2 COL5_PAD CHAR(15),
  /* PACKED DECIMAL DATA TYPE */
  2 COL6 FIXED DEC(15,7),
  2 COL6_PAD CHAR(8),
  /* CHARACTER DATA TYPE */
  2 COL7 CHAR(10),
  /* VARCHAR DATA TYPE */
  2 COL8,
  3 COL8_LEN BIN(15,0),
  3 COL8_TEXT CHAR(10),
  /* DATE DATA TYPE */
  2 COL9 CHAR(10),
  /* TIME DATA TYPE */
  2 COL10 CHAR(8),
  /* TIMESTAMP DATA TYPE */
  2 COL11 CHAR(26),
  /* ROWID DATA TYPE */
  2 COL12,
  3 COL12_LEN BIN(15,0),
  3 COL12_TEXT CHAR(40);

Note the following:

**DB2 data type**
Space in exported data set.

**INTEGER**
4 bytes.

**SMALLINT**
2 bytes.

**REAL (4 bytes)**
4 bytes.
FLOAT (8 bytes)
8 bytes.

PACKED DECIMAL
16 bytes in total. The data is left-aligned. The unused bytes contain binary zeros. A DECIMAL(n,m) field occupies TRUNC(n/2) + 1 bytes within the 16-byte field.

CHAR(n)

n bytes.

VARCHAR(n)

n+2 bytes. The data is preceded by a 2-byte binary field, containing the number of characters in the VARCHAR field.

DATE 10 bytes. See note 1.

TIME 8 bytes. See note 1.

TIMESTAMP

26 bytes.

ROWID

42 bytes. Define a ROWID column as for a VARCHAR(40) column.

Note:
1. The length required for DATE and TIME fields may vary if a date or time exit is in use.

- If the DB2 table has columns that allow nulls and was created using the following definition:

```sql
CREATE TABLE SAMPLE.EXPORT_EX_NULL
(COL1 INTEGER
, COL2 SMALLINT
, COL3 FLOAT(21)
, COL4 FLOAT(53)
, COL5 DECIMAL(1,0)
, COL6 DECIMAL(15,7)
, COL7 CHAR(10)
, COL8 VARCHAR(10)
, COL9 DATE
, COL10 TIME
, COL11 TIMESTAMP
, COL12 ROWID GENERATED ALWAYS NOT NULL )
```

A COBOL copybook that describes the exported data is as follows:

```cobol
01 EXPORT-EXAMPLE2.
* INTEGER DATA TYPE WITH NULL INDICATOR
10 COL1-NULL PIC S9(4) USAGE COMP.
10 COL1 PIC S9(6) USAGE COMP.
* SMALLINT DATA TYPE WITH NULL INDICATOR
10 COL2-NULL PIC S9(4) USAGE COMP.
10 COL2 PIC S9(4) USAGE COMP.
* FLOATING POINT (4 BYTES) DATA TYPE WITH NULL INDICATOR
10 COL3-NULL PIC S9(4) USAGE COMP.
10 COL3 USAGE COMP-1.
* FLOATING POINT (8 BYTES) DATA TYPE WITH NULL INDICATOR
10 COL4-NULL PIC S9(4) USAGE COMP.
10 COL4 USAGE COMP-2.
* PACKED DECIMAL DATA TYPE WITH NULL INDICATOR
10 COL5-NULL PIC S9(4) USAGE COMP.
10 COL5 PIC S9(1)V USAGE COMP-3.
10 FILLER PIC X(15).
* PACKED DECIMAL DATA TYPE WITH NULL INDICATOR
```

How do I...?
How do I...?

In HLASM, the copybook looks like this:

```hlasm
EXAMPLE2 DSECT
  * Integer data type with null indicator
  COL1_NI DS H
  COL1 DS F
  * Smallint data type with null indicator
  COL2_NI DS H
  COL2 DS H
  * Floating point (4 bytes) data type with null indicator
  COL3_NI DS H
  COL3 DS E
  * Floating point (8 bytes) data type with null indicator
  COL4_NI DS H
  COL4 DS D
  * Packed decimal data type (1 byte) with null indicator
  COL5_NI DS H
  COL5 DS PL1
  * Packed decimal data type (8 bytes) with null indicator
  COL6_NI DS H
  COL6 DS PL8
  * Character data type with null indicator
  COL7_NI DS H
  COL7 DS CL10
  * VARCHAR data type with null indicator
  COL8_NI DS H
  COL8_LEN DS H
  COL8_TXT DS CL10
  * DATE data type with null indicator
  COL9_NI DS H
  COL9 DS CL10
  * TIME data type with null indicator
  COL10_NI DS H
  COL10 DS CL8
  * TIMESTAMP data type with null indicator
  COL11_NI DS H
  COL11 DS CL26
  * ROWID data type
  COL12 LEN DS H
  COL12_TXT DS CL40
END
```
In PL/I, the copybook looks like this:

```pli
DCL 1 EXPORT_EXAMPLE UNALIGNED,
  /* INTEGER DATA TYPE WITH NULL INDICATOR */
  2 COL1_NULL BIN(15,0),
  2 COL1 BIN(31,0),
  /* SMALLINT DATA TYPE WITH NULL INDICATOR */
  2 COL2_NULL BIN(15,0),
  2 COL2 BIN(15,0),
  /* FLOATING POINT (4 BYTES) DATA TYPE WITH NULL INDICATOR */
  2 COL3_NULL BIN(15,0),
  2 COL3 FLOAT,
  /* FLOATING POINT (8 BYTES) DATA TYPE WITH NULL INDICATOR */
  2 COL4_NULL BIN(15,0),
  2 COL4 FLOAT(8),
  /* PACKED DECIMAL DATA TYPE WITH NULL INDICATOR */
  2 COL5_NULL BIN(15,0),
  2 COL5 FIXED DEC(1,0),
  2 COL5_PAD CHAR(15),
  /* PACKED DECIMAL DATA TYPE WITH NULL INDICATOR */
  2 COL6_NULL BIN(15,0),
  2 COL6 FIXED DEC(15,7),
  2 COL6_PAD CHAR(8),
  /* CHARACTER DATA TYPE WITH NULL INDICATOR */
  2 COL7_NULL BIN(15,0),
  2 COL7 CHAR(10),
  /* VARCHAR DATA TYPE WITH NULL INDICATOR */
  2 COL8_NULL BIN(15,0),
  2 COL8,
    3 COL8_LEN BIN(15,0),
    3 COL8_TEXT CHAR(10),
  /* DATE DATA TYPE WITH NULL INDICATOR */
  2 COL9_NULL BIN(15,0),
  2 COL9 CHAR(10),
  /* TIME DATA TYPE WITH NULL INDICATOR */
  2 COL10_NULL BIN(15,0),
  2 COL10 CHAR(8),
  /* TIMESTAMP DATA TYPE WITH NULL INDICATOR */
  2 COL11_NULL BIN(15,0),
  2 COL11 CHAR(26),
  /* ROWID DATA TYPE */
  2 COL12,
    3 COL12_LEN BIN(15,0),
    3 COL12_TEXT CHAR(40);
```

The only difference between the first and second set of definitions is the addition of a 2-byte binary field immediately prior to the column definition for those columns that allow DB2 null values.

Related tasks

- "Copying data to a VSAM or QSAM file" on page 221
- "Using templates with non-DB2 data" on page 52

Related references

- "Export Utility panel" on page 567

How do I export data using a copybook that describes a different data format to that used by the FM/DB2 Export Utility?

Scenario: I have data in a DB2 table. I want to export data using a copybook that I have defined, but the copybook describes a different data format to that used by the FM/DB2 Export Utility.
How do I...?

You need to code the copybook in either COBOL, HLASM, or PL/I. When you perform the export with the FM/DB2 Export Utility (3.7), enter the name of the data set that contains the copybook in the **To Copybook or Template** section on the Export “To” panel. Set the **Template usage** processing option to 1 (Above). If you specify 4 (Generate and save), your copybook will be overwritten. If you have renamed the fields in the copybook, or if data type conversions are required, use template re-mapping to map columns in the source table to fields in the target data set.

If the DB2 data contains one or more nullable columns, you are restricted in the processing for the DB2 null indicator. If you need the DB2 null indicator in the exported data, you must include a 2-byte binary field immediately after the definition for the field in the template that corresponds to the column.

For example, for a CHAR(10) field that allows for the DB2 null value, you would code:

```
* CHARACTER DATA TYPE
    10 COL7 PIC X(10).
    10 COL7-NULL PIC S9(4) USAGE COMP.
```

In addition, the 2-byte field must be un-mapped when the export begins. In other words, you must not map a column of the table to this field using template re-mapping; if you do, the FM/DB2 Export Utility does not include the null indicator for the column in the exported data set.

If you want the DB2 null indicator mapped to a different position in the export data set, or you want the DB2 null indicator converted into a different data type, you can:

1. Export the data using either the default mapping, or using a user-defined copybook that includes fields defined for the DB2 null indicators as described above.
2. Create a template that describes the final data format.
3. Use File Manager “base” to copy the data from the export data set into the final format. You may need to use REXX to convert the DB2 null indicators (and possibly other data values) as part of the copy process.

Related tasks

- Copying data to a VSAM or QSAM file on page 221
- Using templates with non-DB2 data on page 52

Related references

- Export “To” panel on page 564
Chapter 8. Working with lists of DB2 objects

If you want to look at a group of DB2 objects, you can use the Object List utility (3.4) to list or print the objects in which you're interested.

You can use the Object List utility to specify:
- Whether you want the list of objects displayed or printed.
- The type of object you want listed or printed.
- Criteria to restrict the objects listed or printed.
- Options to apply when processing the listed objects.

To display the Object List Utility panel,
1. From the Primary Option Menu panel, enter 3 (Utilities) to display the Utility Functions panel.
2. From the Utility Functions panel, enter 4 (Object List)

Now that you have the Object List Utility panel displayed, you can start specifying the sorts of DB2 objects you want to display or print.

Note:
1. One of the options available at the time of installing FM/DB2 allows for views on DB2 catalog tables to be created for those tables accessed by FM/DB2. The set of views includes only those columns within the tables that are needed by FM/DB2.
   If this option has been used at installation, it may affect the results obtained when you use the Object List utility. When FM/DB2 accesses a view, it shows only the columns of the view, rather than the columns of the table. If there are fewer columns in the view than the underlying table, the information displayed is incomplete.
2. The system option, Show all catalog table columns, affects which columns FM/DB2 displays when you use the Object List utility.
   When you select this option, any display of catalog table information includes every available column, including any columns marked as "unused" or "internal use only" in the DB2 for z/OS SQL Reference.
   The default for this option is not selected. Selecting this option overrides any settings for the Show "unused" catalog table columns and Show "internal use only" catalog table columns options.

Related tasks
- "Working with object list panels" on page 244
- "Displaying or printing a list of objects" on page 242

Related references
- "Primary Option Menu panel" on page 617
- "Object List Utility panel" on page 610
- "Utility Functions panel" on page 720
- "Editor Options (3 of 8) panel" on page 535
Displaying or printing a list of objects

To display or print a list of objects:
1. Specify object identification criteria to identify the objects you want listed:

   **Location**
   The location of the objects you want listed.
   Leave this field blank if the objects you want listed are at your current location. Otherwise, type the full name of the remote location, an asterisk (*) to display all available remote locations, or a pattern using wildcard characters. If you specify an asterisk on its own or a pattern, FM/DB2 displays a selection list showing the locations that match the input specification.

   To select the location you want, type **S** against the required entry and press Enter.

   **Owner**
   The name of the person who created the DB2 object.
   To select objects regardless of their owner, leave this field blank. Otherwise, to restrict the list of objects by owner, type either the full name of the owner or a pattern using wildcard characters. If this entry field is not blank, FM/DB2 limits the objects listed to those matching the owner selection criteria you specify.

   **Name**
   The name of the DB2 object.
   To select objects regardless of their name, leave this field blank. Otherwise, to restrict the list of objects by name, type either the full name of the object or a pattern using wildcard characters. If this entry field is not blank, FM/DB2 limits the objects listed to those matching the name selection criteria you specify.

   **Database/collect/schema**
   The name of a database, a collection, or a schema. The name can be a full name or a pattern. This name is used to restrict the list of objects to only those objects that are in the specified database, collection, or schema. If you specify a pattern, the list of objects includes any objects that are included in a database or collection or schema whose name matches the pattern specification.

   The way in which FM/DB2 uses this field depends upon the type of object being listed as specified using the **Object Type** (see step 3 on page 243):

<table>
<thead>
<tr>
<th>Object type</th>
<th>Field use</th>
</tr>
</thead>
<tbody>
<tr>
<td>Database</td>
<td>Not allowed</td>
</tr>
<tr>
<td>Table space</td>
<td>Used as database name</td>
</tr>
<tr>
<td>Table, view, or alias</td>
<td>Used as database name</td>
</tr>
<tr>
<td>Index</td>
<td>Used as database name</td>
</tr>
<tr>
<td>Column</td>
<td>Used as database name</td>
</tr>
<tr>
<td>Synonym</td>
<td>Used as database name</td>
</tr>
<tr>
<td>Plan</td>
<td>Not allowed</td>
</tr>
</tbody>
</table>
Displaying or printing a list of objects

Package
  Used as collection name
Collection
  Not allowed
DBRM
  Not allowed
Schema
  Used as schema name
Distinct type
  Used as schema name
Function
  Used as schema name
Stored procedure
  Used as schema name
Trigger
  Used as schema name

2. Specify any additional selection criteria to further restrict the objects to be listed by coding a simple SQL SELECT “where clause” that specifies a column in the appropriate DB2 system catalog table.

Column
  Name of column in the relevant DB2 system catalog table. To list the columns in the appropriate DB2 system catalog table, enter an asterisk (*).

Operator
  A valid SQL operator.

Value
  The value to be compared with the contents of the column.

For further information about SQL statements, see the DB2 for z/OS SQL Reference.

For example, to restrict a list of columns to only those columns with a data type of DATE, specify the following values:

<table>
<thead>
<tr>
<th>Additional Selection Criteria:</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Column ........................</td>
<td>COLTYPE</td>
</tr>
<tr>
<td>Operator ......................</td>
<td>=</td>
</tr>
<tr>
<td>Value .........................</td>
<td>'DATE'</td>
</tr>
</tbody>
</table>

3. Nominate the type of object you want listed, and other options to apply to processing of the listed objects.

Object Type
  Specify the type of DB2 object that you want listed. The numeric indicator for each object type is shown on the Object List Utility panel. This input field is mandatory, but defaults to “3” (Table/view/alias).

Confirm DB2 Object Drop
  Indicates if you want FM/DB2 to display a confirmation panel each time you enter a DR(op) command against an object.

/    Display a confirmation panel if a DROP command is entered against an object. This is the default setting.

(blank)
    If a DROP command is entered against an object, issue the command without displaying a confirmation panel.

4. Specify if you want to display or print the list of objects:
   * To display a list of objects, leave the command line blank.
Displaying or printing a list of objects

- To print a list of objects, enter `P` on the command line.

5. Press Enter.

   FM/DB2 displays or prints the object list panel for the type of object you have specified.

Related tasks

"Using an asterisk (*) or a pattern in entry fields" on page 28

Related references

"Remote DB2 Location Selection panel" on page 641

Working with object list panels

After you have completed the necessary steps to display a list of objects, FM/DB2 displays the object list panel for the corresponding object type.

The following pages describe the features that are common to all object list panels. Figure 52 shows the areas of a typical object list panel.

<table>
<thead>
<tr>
<th>Process Options Utilities Help</th>
<th>A</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2) Databases</td>
<td></td>
</tr>
<tr>
<td>Location: DATABASE DATABASE</td>
<td></td>
</tr>
<tr>
<td>STORAGE GROUP NAME BUFFER</td>
<td></td>
</tr>
<tr>
<td>POOL NAME DATABASE</td>
<td></td>
</tr>
<tr>
<td>INTERNAL AUTHID DATABASE TYPE</td>
<td></td>
</tr>
<tr>
<td>THAT IDENTIFIER CREATED</td>
<td></td>
</tr>
<tr>
<td>TYPE OF DATABASE</td>
<td>B</td>
</tr>
<tr>
<td>#1---- #2++++ #3+++++++ #4++++</td>
<td></td>
</tr>
<tr>
<td>#5 #7++++ #20++++</td>
<td>D</td>
</tr>
<tr>
<td>Top of data</td>
<td></td>
</tr>
<tr>
<td>ABC KEISTE2 DDDDDDDDDDD BP0 270 KEISTE2</td>
<td></td>
</tr>
<tr>
<td>DSNB081A JCULLEN DSNB0810 BP0 259 JCULLEN</td>
<td></td>
</tr>
<tr>
<td>DSNB081P JCULLEN DSNB0810 BP0 260 JCULLEN</td>
<td></td>
</tr>
<tr>
<td>DSNATP0B JCULLEN SYSDEFLT BP0 257 JCULLEN</td>
<td></td>
</tr>
<tr>
<td>DSNDB04 SYIBM SYSDEFLT BP0 4 SYSIBM</td>
<td></td>
</tr>
<tr>
<td>DSNDB06 SYIBM SYSDEFLT BP0 6 SYSIBM</td>
<td></td>
</tr>
<tr>
<td>DSNDB07 JCULLEN SYSDEFLT BP0 7 JCULLEN WORKFILE</td>
<td></td>
</tr>
<tr>
<td>DSNRG0DB JCULLEN SYSDEFLT BP0 258 JCULLEN</td>
<td></td>
</tr>
<tr>
<td>DSNRLST JCULLEN SYSDEFLT BP0 256 JCULLEN</td>
<td></td>
</tr>
<tr>
<td>JLV53081A KEISTEN JLV530810 BP0 266 KEISTEN</td>
<td></td>
</tr>
<tr>
<td>JLV53081P KEISTEN JLV530810 BP0 267 KEISTEN</td>
<td></td>
</tr>
<tr>
<td>JOHNB081P KEISTEN JOHNB0810 BP0 265 KEISTEN</td>
<td></td>
</tr>
<tr>
<td>KPS0808A KEISTEN KPS08080 BP0 268 KEISTEN</td>
<td></td>
</tr>
<tr>
<td>KPS0808P KEISTEN KPS08080 BP0 269 KEISTEN</td>
<td></td>
</tr>
<tr>
<td>KPS0808A KEISTEN KPS08080 BP0 267 KEISTEN</td>
<td></td>
</tr>
<tr>
<td>KPS0808P KEISTEN KPS08080 BP0 268 KEISTEN</td>
<td></td>
</tr>
<tr>
<td>KPS0808A KEISTEN KPS08080 BP0 269 KEISTEN</td>
<td></td>
</tr>
<tr>
<td>KPS0808P KEISTEN KPS08080 BP0 267 KEISTEN</td>
<td></td>
</tr>
<tr>
<td>RMO133 KEISTEN SYSDEFLT BP0 261 KEISTEN</td>
<td></td>
</tr>
<tr>
<td>End of data</td>
<td></td>
</tr>
</tbody>
</table>

Figure 52. Layout of a typical object list panel

A **Action bar.** The action bar choices for the object list panels function.

B **Column heading lines.** Shows the name of the column of data shown below. A plus sign (“+”), minus sign (“-”), or both (“+-”) indicates that the column is scrollable and may contain more data than is currently displayed.
Working with object list panels

- **Filter line.** This is where you can specify a filter string for one or more columns to limit the information FM/DB2 displays.

- **Operator line.** This is where you can specify an operator for one or more columns to work in conjunction with the corresponding filter pattern.

- **Line command area.**

- **End of data marker.** This marker line indicates the end of the data returned from FM/DB2.

- **Command line.** On this line you can specify any DB2 command, ISPF command, or FM/DB2 primary command.

Related tasks

- “Displaying or printing a list of objects” on page 242
- “Working with scrollable display fields” on page 22
- “Limiting the data displayed”
- “Using the line command area (Cmd)” on page 250
- “Sorting the displayed data” on page 257

Related references

- “Collections panel” on page 396
- “Columns panel” on page 411
- “Databases panel” on page 500
- “Database Request Modules panel” on page 498
- “Distinct Types panel” on page 523
- “Functions panel” on page 584
- “Indexes panel” on page 595
- “Application Packages panel” on page 387
- “Application Plans panel” on page 389
- “Schemas panel” on page 657
- “Stored Procedures panel” on page 685
- “Synonyms panel” on page 687
- “Tables, Views and Aliases panel” on page 704
- “Table Spaces panel” on page 698
- “Triggers panel” on page 712

Limiting the data displayed

You can limit the data that FM/DB2 displays by “filtering out” the data you do not want displayed.

To filter the data, use:

- The *filter line* to specify a filter pattern for one or more columns
- And, optionally, the *operator line* to specify an operator to work in conjunction with the corresponding filter pattern.

Using the filter line

On the filter line, the asterisk (*) under a column heading marks an area where you can enter a filter pattern to limit the data displayed. You can specify filter patterns for more than one column if necessary.

For alphanumeric columns, the asterisk is displayed in the leftmost position. For numeric columns, the asterisk is displayed in the rightmost position.

In the filter pattern area, use:

- An asterisk (*) on its own to match all values, or
A filter pattern using wildcard characters to select only those objects that match the pattern.

You can specify a filter pattern using any of the following special characters:

**asterisk (**)**
- Represents a string of zero or more characters. Acts the same as a percent sign (%).

**percent sign (%)**
- Represents a string of zero or more characters. Acts the same as an asterisk (*).

**underscore (_)**
- Represents any single character, excluding a trailing blank.

When you specify a filter pattern for a column, FM/DB2 only selects objects where the contents of that column match the specified filter pattern.

**Note:** If you have also specified an operator (on the operator line) for the column, FM/DB2 uses the operator in conjunction with the filter pattern to match objects (see following section).

For alphanumeric columns, FM/DB2 matches the filter pattern you enter with the *leading* characters of the contents of the column for each row. That is, the filter pattern DEF matches 'DEF' and 'DEFG', but not 'CDEF'.

For numeric columns, FM/DB2 matches the filter pattern you enter with the *value* of the contents of the column for each row. That is, the search argument 7 matches columns containing 7 but not, for example, 70 or 17.

The following examples show how you can use filter patterns to limit the data FM/DB2 displays. (In the examples shown, it is assumed no operator has been specified in the corresponding area on the operator line.)

**Pattern filter**

<table>
<thead>
<tr>
<th>Pattern filter</th>
<th>Selects objects where column contains...</th>
</tr>
</thead>
<tbody>
<tr>
<td>*</td>
<td>Any alphanumeric string</td>
</tr>
<tr>
<td>ABC*</td>
<td>Any alphanumeric string starting with &quot;ABC&quot;</td>
</tr>
<tr>
<td><em>ABC</em></td>
<td>Any alphanumeric string containing the string &quot;ABC&quot;</td>
</tr>
<tr>
<td>A__C</td>
<td>Any four-character alphanumeric string where the first character is &quot;A&quot; and the last character is &quot;C&quot;</td>
</tr>
<tr>
<td>%BC_</td>
<td>Any alphanumeric string containing the string &quot;BC&quot; followed by one more character</td>
</tr>
<tr>
<td>123</td>
<td>For numeric columns: the value 123</td>
</tr>
<tr>
<td>23</td>
<td>For numeric columns: the value 23</td>
</tr>
</tbody>
</table>

**Using the operator line**

You use the operator line to specify an operator that FM/DB2 uses in conjunction with the filter pattern specified in the corresponding area on the filter line. You can specify operators for more than one column if necessary. Normally, you only specify an operator for a column where you have also specified a filter pattern.

To enter an operator in the operator line, type the required operator anywhere within the operator area for the required column. Processing occurs left to right. FM/DB2 only recognizes the first operator encountered and any other characters
entered in the operator field are ignored. For example, if you type "=>" in the
operator field, FM/DB2 treats it as "=" and ignores the ">".

To remove an operator, overtype the operator with a space.

You can use any of the following operators:

**Blank**  Means “like”. Any *, %, or _ characters in the corresponding filter pattern
are treated as special characters. A match occurs if the filter pattern
matches the data after considering any special characters.

! (or) ¬  Means “not like”. Any *, %, or _ characters in the corresponding filter
pattern are treated as special characters. A match occurs if the filter pattern
does not match the data after considering any special characters.

=  Means “equals”. For numeric data types, special characters in the
 corresponding filter pattern are not allowed. For non-numeric data types, *,%
and _ characters are allowed but are not treated as special characters. *
%, and _ characters match themselves only.

!=, ¬=, (or) <>  Means “not equals”. For numeric data types, special characters in the
 corresponding filter pattern are not allowed. For non-numeric data types, *,%
and _ characters are allowed but are not treated as special characters. *
%, and _ characters match themselves only.

<  Means “less than”. For numeric data types, special characters in the
 corresponding filter pattern are not allowed. For non-numeric data types, *,%
and _ characters are allowed but are not treated as special characters. *
%, and _ characters match themselves only.

<=, !>, (or) <=>  Means “less than or equal to”. For numeric data types, special characters in the
 corresponding filter pattern are not allowed. For non-numeric data types, *,%
and _ characters are allowed but are not treated as special characters. *
%, and _ characters match themselves only.

>=, !<, (or) =><  Means “greater than or equal to”. For numeric data types, special
characters in the corresponding filter pattern are not allowed. For non-numeric data types, *,%
and _ characters are allowed but are not treated as special characters. *
%, and _ characters match themselves only.

>  Means “greater than”. For numeric data types, special characters in the
 corresponding filter pattern are not allowed. For non-numeric data types, *,%
and _ characters are allowed but are not treated as special characters. *
%, and _ characters match themselves only.

You can also enter the following commands in the operator field for a column:

**X**  Excludes the column from the data display. The effect is the same as
 editing the template and de-selecting the column, followed by a re-display
 of the data.

**W**  Resets the display width for the column to the maximum value for the
 column. The command is only effective when entered against a column
 that has a non-numeric data type.

**W** **nnn**  Sets the display width for the column to **nnn** (**nnn** must be an integer)
Working with object list panels

between 6 and the maximum possible display width for the column). The command is only effective when entered against a column that has a non-numeric data type.

When you specify an operator for a column, FM/DB2 first applies the operator to the corresponding filter pattern and then only selects objects where the contents of that column match the resultant filter pattern.

The following examples show how you can use operators, in conjunction with filter patterns, to limit the data FM/DB2 displays.

**Pattern filter (operator)**

Selects objects where column...

- **ABC* (blank)**
  Contains any alphanumeric string starting with “ABC”

- **ABC* (=)**
  Contains the string “ABC***”

- **ABC* (¬)**
  Does not contain the string “ABC***”

- **A (>)**
  Contains an alphanumeric string whose first character is greater than “A”

- **A_C (¬>)**
  Contains an alphanumeric string whose first three characters are less than or equal to the string “A_C”

- **123 (=)**
  For numeric columns: contains the value 123

- **23 (<)**
  For numeric columns: contains a value less than 23

Let’s take an example. Figure 52 on page 244 shows a list of database object types. To limit the columns displayed to those where:

- The database creator is KEISTEW
- The storage group name starts with the letter K
- The internal database identifier is greater than 264

Enter the following filter patterns and operators:

1. Under the column heading, **DATABASE CREATOR**, type KEISTEW on the filter line.
2. Under the column heading, **STORAGE GROUP NAME**, type K* on the filter line.
3. Under the column heading, **INTERNAL DATABASE IDENTIFIER**, type 264 on the filter line and > on the operator line.
4. Press Enter. A list of database object types is displayed, restricted to the search criteria you specified:

<table>
<thead>
<tr>
<th>Process Options Utilities Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>-------------------------------</td>
</tr>
<tr>
<td>FM/DB2 (DFG2) Databases</td>
</tr>
<tr>
<td>Location:</td>
</tr>
<tr>
<td>DATABASE DATABASE GROUP POOL DATABASE THAT IDENTIFIER CREATED DATABASE</td>
</tr>
<tr>
<td>SEL NAME CREATOR NAME NAME IDENTIFIER CREATED DATABASE</td>
</tr>
<tr>
<td>* KEISTEW K* * 264 * *</td>
</tr>
</tbody>
</table>

**** Top of data ****

| ABC KEISTEW KEISTE2 DDDDDDDDDDD BP0 270 KEISTE2 |
| DSNDB081A JCULEN DSNB8B10 BP0 259 JCULEN |
| DSNDB081B JCULEN DSNB8B10 BP0 260 JCULEN |
| DSNATPDB JCULEN SYSDFLT BP0 257 JCULEN |
| DSNDB04 SYSBM SYSDFLT BP0 4 SYSBM |
| DSNDB06 SYSBM SYSBM BP0 6 SYSBM |
| DSNDB07 JCULEN SYSDFLT BP0 7 JCULEN WORKFILE |
| DSNRGFD8 JCULEN SYSDFLT BP0 258 JCULEN |
| DSNRLST JCULEN SYSDFLT BP0 256 JCULEN |
| JLV3DB81A KEISTEW JLV3G810 BP0 266 KEISTE2 |
| JLV3DB81P KEISTEW JLV3G810 BP0 267 KEISTE2 |
| JHNDB81P KEISTEW JHN8G810 BP0 265 KEISTE2 |
| KPSDB81A KEISTEW KPS8B810 BP0 268 KEISTE2 |
| KPSDB81B KEISTEW KPS8B810 BP0 269 KEISTE2 |
| KPSAD81A KEISTEW KPSA8G810 BP0 262 KEISTE2 |
| KPSAD81P KEISTEW KPSA8G810 BP0 263 KEISTE2 |
| RFD0133 KEISTEW SYSDFLT BP0 261 KEISTE2 |

**** End of data ****

Command ===> Scroll PAGE

F1=Help F2=Zoom F3=Exit F4=Retrieve F5=Find F6=Change
F7=Up F8=Down F9=Swap F10=Left F11=Right F12=Cancel

Figure 53. Object list: databases

Figure 54. Object List: databases (list restricted using filter patterns and operators)
Working with object list panels

Locating a row or column in a list of DB2 objects

When you display data with the Object List utility, you can use the LOCATE primary command to locate either a row or a column in a similar way to when you are viewing data with FM/DB2 View or Edit.

Related references
“Locating a row” on page 108
“Locating a column” on page 109
“LOCATE primary command” on page 761

Finding a string in a list of DB2 objects

When you display data with the Object List utility, you can use the FIND primary command to find a specific string in a similar way to when you are viewing data with FM/DB2 View or Edit.

Related references
“Finding a string” on page 112
“FIND primary command” on page 748

Using the line command area (Cmd)

In the line command area (under the heading Cmd) for a listed object, you can enter a line command or one of the general line commands (>, =, or ROW).

Line command

Table 22 shows the line commands you can use with each type of object.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
<th>Database</th>
<th>Table space</th>
<th>Table/View</th>
<th>Index</th>
<th>Column</th>
<th>Synonym</th>
<th>Plan</th>
<th>Package</th>
<th>Collection</th>
<th>DBRM</th>
<th>Schema</th>
<th>Distinct Type</th>
<th>Function</th>
<th>Procedure</th>
<th>Trigger</th>
<th>Storage Group</th>
<th>Sequence</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Alter an object</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADD</td>
<td>Add table check constraint.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ALS</td>
<td>Show aliases of an object.</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AT</td>
<td>Show auxiliary tables for LOB columns.</td>
<td>Y</td>
<td></td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>Browse the object</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BI</td>
<td>Bind the object</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BT</td>
<td>Show base table for auxiliary table.</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>Copy table</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CBI</td>
<td>Copy and bind the object</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CDI</td>
<td>Show column distribution statistics (as generated by RUNSTATS).</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CH</td>
<td>Show child (dependent) tables</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CHR</td>
<td>Show child relationships (referential constraints).</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Working with object list panels

#### Table 22. Object list line commands (continued)

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
<th>Database</th>
<th>Table space</th>
<th>Table/View</th>
<th>Index</th>
<th>Column</th>
<th>Synonym</th>
<th>Plan</th>
<th>Package</th>
<th>Collection</th>
<th>DBRM</th>
<th>Schema</th>
<th>Distinct Type</th>
<th>Function</th>
<th>Procedure</th>
<th>Trigger</th>
<th>Storage Group</th>
<th>Sequence</th>
</tr>
</thead>
<tbody>
<tr>
<td>CK</td>
<td>Show information about table check constraints.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>COL</td>
<td>Show the columns for the object.</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>COM</td>
<td>Comment on an object.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CR</td>
<td>Create an object like this object.</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>CRA</td>
<td>Create an alias for an object.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CRS</td>
<td>Create a table space in the database.</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CRT</td>
<td>Create a table in the table space.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CRX</td>
<td>Create an index.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CS</td>
<td>Describe the columns for an object (abbreviation of information provided by COL command).</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>D</td>
<td>Show the database for an object.</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DEP</td>
<td>Show the dependencies on an object.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DI</td>
<td>Display distribution statistics of a column.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DR</td>
<td>Drop the object/constraint.</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>DS</td>
<td>Display the database structure.</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DT</td>
<td>Show the source data type (columns) or distinct types (schemas).</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>E</td>
<td>Edit the object.</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EDC</td>
<td>Show enabled/disabled connections.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>F</td>
<td>Show functions.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FC</td>
<td>Show the foreign key From Column.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FK</td>
<td>Show information about the referential integrity defined for foreign keys.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FRE</td>
<td>Free the object.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>G</td>
<td>Grant privileges for the object.</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>GEN</td>
<td>Generate SQL for object from DB2 catalog.</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>H</td>
<td>Show the homonyms (like-named objects) for the object.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>I</td>
<td>Show detailed information about the object.</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
</tbody>
</table>
## Working with object list panels

**Table 22. Object list line commands (continued)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
<th>Database</th>
<th>Table/View</th>
<th>Index</th>
<th>Column</th>
<th>Synonym</th>
<th>Plan</th>
<th>Package</th>
<th>Collection</th>
<th>DBRM</th>
<th>Schema</th>
<th>Distinct Type</th>
<th>Function</th>
<th>Procedure</th>
<th>Trigger</th>
<th>Storage Group</th>
<th>Sequence</th>
</tr>
</thead>
<tbody>
<tr>
<td>ICS</td>
<td>Show the status of image copies for the object.</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>L</td>
<td>Show the collection for the object or, for tables, show the rows of this table.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LAB</td>
<td>Label the object.</td>
<td></td>
<td></td>
<td></td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LC</td>
<td>Show LISTCAT information for the data set.</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LPT</td>
<td>List the PLAN_TABLE table for the object.</td>
<td></td>
<td></td>
<td></td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M</td>
<td>Show the DBRMs for the object.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>O</td>
<td>Show related stored procedures.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>P</td>
<td>Display information about the privileges for this object.</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PA</td>
<td>Show parent tables.</td>
<td></td>
<td></td>
<td></td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PAR</td>
<td>Show parent relations (referential constraints).</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PARM</td>
<td>Show the routine parameters.</td>
<td></td>
<td></td>
<td></td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PC</td>
<td>Show the column privileges for this object.</td>
<td></td>
<td></td>
<td></td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PK</td>
<td>Show the primary key for this table.</td>
<td></td>
<td></td>
<td></td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PKG</td>
<td>Show the packages for the object.</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PKL</td>
<td>Show the package lists for the object.</td>
<td></td>
<td></td>
<td></td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PL</td>
<td>Show the plans for the object.</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PR</td>
<td>Print table.</td>
<td></td>
<td></td>
<td></td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PS</td>
<td>Display information about the privileges for the schema for this object.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PST</td>
<td>Show partition statistics.</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PT</td>
<td>Show the parts of the object.</td>
<td></td>
<td></td>
<td></td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>R</td>
<td>Revoke the privilege for this object.</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RBI</td>
<td>Rebind the object.</td>
<td></td>
<td></td>
<td></td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RDT</td>
<td>Return data type.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ROW</td>
<td>Show all the columns in this row of the result table.</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RT</td>
<td>Recover TOCOPY.</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>S</td>
<td>Show the table spaces for the object.</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Table 22. Object list line commands (continued)

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
<th>Database</th>
<th>Table/View</th>
<th>Index</th>
<th>Column</th>
<th>Synonym</th>
<th>Plan</th>
<th>Package</th>
<th>Collection</th>
<th>DBRM</th>
<th>Schema</th>
<th>Distinct</th>
<th>Function</th>
<th>Procedure</th>
<th>Trigger</th>
<th>Storage Group</th>
<th>Sequence</th>
</tr>
</thead>
<tbody>
<tr>
<td>SCH</td>
<td>Show the schema the object.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SEL</td>
<td>Build SQL SELECT statement for this object.</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SEQ</td>
<td>Show identity column information.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SQL</td>
<td>Show the SQL statements in the object.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SYM</td>
<td>Show the synonyms for the object.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>T</td>
<td>Show the tables for the object.</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TC</td>
<td>Show the foreign key To Column.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TR</td>
<td>Show triggers.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UTL</td>
<td>Run a DB2 utility job against the object.</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td></td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>V</td>
<td>View the object.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VER</td>
<td>Show the versions of the object.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VOL</td>
<td>Show volumes.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VS</td>
<td>Show how the view was created.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VW</td>
<td>Show views.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X</td>
<td>Show the indexes for the object.</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>XC</td>
<td>Show the indexes and index columns for the object.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>XPT</td>
<td>Show index parts.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Notes:**

1. Only available from a list of table constraints.
2. Only available from a list of object privileges.
3. Only available from a list of foreign keys.
4. Only available when the object is a view.
5. Only available when the object is a view or an alias, or from a list of object privileges.
6. Only available from a list of image copies.
7. The synonym name is copied to the function panel if the location is unspecified and the owner is the SQLID; otherwise, the synonym name is resolved and the resolved name is copied.

**Related tasks**

"Line command query (?)"
"Equal (=)" on page 254
"ROW" on page 255

**Line command query (?)**

If you enter ? in the line command area, FM/DB2 displays a list of the line commands you can enter for this object type. Figure 55 on page 254 shows an example of a list of valid line commands (in this case, for a synonym object type).
Working with object list panels

To select a command from the list of valid commands, enter `S` in the `Sel` column.

Related references

- “Object List Line Commands panel” on page 610
- Table 22 on page 250

**Equal (=)**

If you enter `=` in the line command area, FM/DB2 runs the previous line command.

Figure 56 on page 255 shows the use of the `=` line command. The previous line command that FM/DB2 ran was DS (to display the structure of the data base DSNRLST). The asterisk is displayed in the first position of the line command area after FM/DB2 has run the command and you have returned to the original panel. If you enter `=` in the line command area for another listed object (in this case, the data base FMN2), FM/DB2 runs the DS command for that object.
If you enter the "=" line command multiple times, as shown in Figure 57, FM/DB2 runs the line command (in this case, DS) consecutively for each of the nominated objects in the list without returning to the original panel between each execution.

<table>
<thead>
<tr>
<th>Command</th>
<th>Name</th>
<th>Owner</th>
<th>+ Group</th>
<th>Pool</th>
<th>Created DBID</th>
<th>By + T</th>
<th>E</th>
<th>Pool</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>DSNDB04</td>
<td>SYIBM</td>
<td>SYDEFLT</td>
<td>BP0</td>
<td>4 SYIBM</td>
<td>BPO</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>DSNDB06</td>
<td>SYIBM</td>
<td>SYDEFLT</td>
<td>BP0</td>
<td>6 SYIBM</td>
<td>E BPO</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>DSNDB07</td>
<td>FMUSER</td>
<td>SYDEFLT</td>
<td>BP0</td>
<td>7 FMUSER</td>
<td>W BPO</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>DSNRLST</td>
<td>FMUSER</td>
<td>SYDEFLT</td>
<td>BP0</td>
<td>256 FMUSER</td>
<td>E BPO</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>DSNRGDFB</td>
<td>FMUSER</td>
<td>SYDEFLT</td>
<td>BP0</td>
<td>257 FMUSER</td>
<td>E BPO</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>FMN2</td>
<td>FMUSER</td>
<td>SG01</td>
<td>BPO</td>
<td>258 FMUSER</td>
<td>E BPO</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>FMN2</td>
<td>FMUSER</td>
<td>SG01</td>
<td>BPO</td>
<td>259 FMUSER</td>
<td>E BPO</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>DSN8D61P</td>
<td>BRADB</td>
<td>DSN8G610</td>
<td>BPO</td>
<td>261 BRADB</td>
<td>E BPO</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>DSN8D61A</td>
<td>BRADB</td>
<td>DSN8G610</td>
<td>BPO</td>
<td>260 BRADB</td>
<td>E BPO</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>DSN8D61L</td>
<td>BRADB</td>
<td>DSN8G610</td>
<td>BPO</td>
<td>262 BRADB</td>
<td>E BPO</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 57. Using the equal (=) line command multiple times

ROW

If you enter ROW in the line command area, FM/DB2 displays the column name (and its value) as held in the DB2 catalog table for each column of the specified object.

You can enter more than one ROW line command at a time.
Working with object list panels

**Figure 58** shows the use of the ROW line command, placed in the line command area for the synonym object TCONA.

**Figure 59** shows the result. In this case, the column names are from the DB2 catalog table, SYSSYNONYMS.

You can limit the number of entries shown on the panel by entering search criteria in the areas marked by an asterisk (*) immediately below the headings.

Related tasks

"Limiting the data displayed" on page 245
Sorting the displayed data

Use the SORT primary command to sort a column in the current object list.

To sort one column (in ascending sequence):
1. Type SORT on the command line.
2. Place the cursor within the column you want to sort.
3. Press Enter.
   FM/DB2 sorts the column in ascending sequence.

To sort multiple columns, in ascending or descending sequence:
1. Type SORT on the command line.
2. Ensure the cursor is not positioned within any column.
3. Press Enter:
   FM/DB2 displays the Sort Fields panel.
   You can limit the number of entries shown on the panel by entering search criteria in the areas marked by an asterisk (*) immediately below the headings.

Related tasks
   “Limiting the data displayed” on page 245

Related references
   “Sort Fields panel” on page 679
   “SORT primary command” on page 777

Reverse engineering

Reverse engineering lets you reverse engineer the DB2 objects in your database catalog.

Reverse engineering generates the SQL statements necessary to re-create a DB2 object.

Use the GEN line command on the corresponding object list panel to reverse engineer any of the following objects:
- Database
- Table space
- Table or view
- Index
- Schema
- Distinct type
- Function
- Stored procedure

Note: The GEN line command applies to objects at the current server only. If you enter the GEN line command against an object at a remote location, the location information is discarded when the Generate SQL From DB2 Catalog panel is displayed.

To generate DDL for objects at a remote location, first connect to that location, then use the GEN line command.

Figure 60 on page 258 shows the GEN line command entered on a database object list panel:
Typical uses for reverse engineering include:

- Extracting the DDL for an object before changes are made, so that the changes are applied to the current definition and are available for fallback purposes.

- Move DB2 objects to another DB2 subsystem. By using reverse engineering (together with the DB2 table unload and load facilities), objects can be moved after a few manual modifications to the generated SQL and batch jobs.

When extracting database, table space, and table objects, all dependent objects can also be generated; this includes table spaces, tables, indexes, views, synonyms, aliases, referential constraints, table checks, and table triggers. When extracting objects in schemas, reverse engineering can extract the dependent distinct types, functions, and stored procedures. All authorizations to these objects can also be generated.

You can generate the SQL statements using a batch or online job. If you are using FM/DB2 to extract several objects from a large catalog, batch jobs are recommended.

If you use the GEN line command, FM/DB2 displays the Generate SQL From DB2 Catalog panel.

On the Generate SQL From DB2 Catalog panel, you can:

- As an option, specify new values for the:
  - Storage group (possibly using a different storage group for table spaces and index spaces).
  - Database.
  - Specify a new object owner. If specified, the new owner is used whenever an object is created.
  - Specify a new schema name (where applicable). If specified, the new schema is used whenever an object is created.
  - Specify the data set in which FM/DB2 places the generated SQL.
  - Specify whether the SQL generation runs as a batch or online job.
Reverse engineering

- Specify how often reverse engineering adds an SQL COMMIT statement to the generated SQL.
- Specify whether DB2 default parameters are removed or kept in the generated SQL.

If you specify an execution mode of BATCH, FM/DB2 generates a batch job and displays the job in an ISPF Edit session, ready for any modifications you need to make before submitting the job for execution. If you specify TSO, FM/DB2 generates the SQL statements online and displays the results.

Related tasks
“Performance tips”

Related references
“Generate SQL From DB2 Catalog panel” on page 585

Performance tips
To improve performance of the extract program, consider adding the following indexes to the DB2 catalog:

**On SYSDBAUTH**
(database,grantor,grantee)

**On SYSCHECKS**
tbowner,dbname

**On SYSRELS**
creator,tbname,relname

**On SYSRESAUTH**
(creator,dbname,relname,obtype)

**On SYSTABAUTH**
tcreator,dbname,relname,grantee,granteetype

**On SYSCOLAUTH**
creator,dbname,relname,dgranted,tgranted)

The recommended mode of operation is batch, even if only a few objects are requested. This is because the design process is based on the standard DB2 catalog indexes, which means that some parts of the process scan the catalog tables instead of doing a direct reference.

Considerations
The following considerations apply to reverse engineering:

- FM/DB2 does not extract IDCAMS DEFINE CLUSTER statements for VCAT-defined table spaces and indexes.
- The generated SQL for table spaces and indexes defined with a DSETPASS (password) contains a SPUFI comment line such as:
  -- DSETPASS XXXXXXXX

  FM/DB2 does not reveal the data set password in the catalog; FM/DB2 generates the comment line and issues a warning.
- The ability to generate actually allocated or actually used space allocations depends on information in the DB2 catalog. The actual data set sizes for table spaces or index spaces are not retrieved. Only use these options if you have recently run STOSPACE and RUNSTATS for the selected objects.
Reverse engineering
Chapter 9. Printing

FM/DB2 provides various utilities that allow you to print or view printed output:
- To print the contents of a DB2 table or view, use the Print utility function (3.1).
- To browse, and optionally purge, the print output from your current FM/DB2 editor session held in the print data set, use the Print Browse utility function (3.11).
- To print an audit trail report using the contents of the audit trail data set, use the Audit trail utility function (3.10).

Each of these utilities is described in the remainder of this chapter.

Related tasks
- “Printing the contents of a DB2 table” on page 266
- “Looking at the print output from your FM/DB2 session” on page 266
- “Printing an audit trail report” on page 267

Printing the contents of a DB2 table

To print the contents of a DB2 table or view, use the Print utility function (3.1).

When you print data, you can:
- Select which rows to print.
- Select which columns to print and the order in which they are arranged on the listing.
- Print DB2 tables or views with columns formatted according to column data type.
- Provide customized column headings.
- Limit the number of rows printed:

  **Limit the print**

  You can limit the number of rows that are printed by specifying the number of rows to print.

  To print all the rows of the DB2 object, specify 0 (zero) or ALL.

  **Select rows**

  You can select the rows you want to print by specifying row selection criteria.

To display the Print Utility panel,
1. From the Primary Option Menu panel, enter 3 (Utilities) to display the Utility Functions panel.
2. From the Utility Functions panel, enter 1 (Print)

Now that you have the Print Utility panel displayed, you can specify the name of the DB2 table or view you want to print, how many rows you want to print, and other options to control how the data is printed.

When you have finished specifying these details, press Enter.
Printing the contents of a DB2 table

If you selected Edit options (either by using a “/”, or by having previously used the “A” selection character for this option), before printing the data, the Print Utility Options panel is displayed.

If you selected Edit template (either by using a “/”, or by having previously used the “A” selection character for this option), before printing the data, the Column Selection/Edit panel is displayed.

If you selected Batch execution (either by using a “/”, or by having previously used the “A” selection character for this option), the generated JCL is displayed using the ISPF editor, where you can review the JCL and submit the job for batch processing.

Otherwise (if you selected none of the preceding options), FM/DB2 prints the data.

Formatting the print output

Formatting of the print output is controlled by the following:

Settings: Print settings (option 0.0.1)
Several processing options, that you can specify using the Set Print Processing Options panel, affect the printed output:

Output destination
Where you want the printed output sent:

SYSPRINT
Send print output to the current SYSPRINT allocation.
Typically, SYSPRINT is allocated to the terminal, making this option synonymous with TERMINAL. However, you can allocate SYSPRINT in other ways.

Terminal
Send print output to the terminal.

Data set
Output is accumulated in the print data set specified in the Data set name field. This data set can browsed using the PB command or sent to the JES spool queue for printing by issuing the PRINT command while browsing the data set. (The output is sent to the JES spool queue class specified in the Output class field.)

REXX Send print output to the REXX stem variable FILEM.
Choose this option when you want to control your printing from within a REXX program.

Record length
How many columns wide the output is

Page skip
Whether output from each function starts on a new page

Wide print
Whether the maximum print line length for print output is used

Translate non-printable chars
Whether non-printable characters are translated to blanks

Uppercase message text
Whether all message text is translated to uppercase.
Printing the contents of a DB2 table

Data set DISP
Whether the print output is appended to the existing data set (MOD) or replaces it (OLD).

Note: This option only affects output sent to the print data set specified in the PRINTDSN option.

Dump format
Which format (updown or across) is used for hexadecimal print output

Data set name
The print data set where print output is directed when the PRINTOUT print option is set to SYSOUT=

Output class
The class of the JES spool queue to be used when the PRINT command is issued while browsing the temporary print data set.

Lines per page
How many lines per page the output has

Record limits
Which part of each record to print

You can view the print data set using Print Browse (option 3.11). While you are using the Print Browse, you can use the PURGE primary command to clear the print data set. To transfer the contents of the print data set to a SYSOUT class, press the Print function key (F4).

Print Mode
This entry field on the Print Utility panel determines the format of the printed output.

Related references
"Set Print Processing Options panel" on page 668
"PURGE primary command" on page 769

Editing the template
Before you print the data, you can edit the template to specify what data is printed and how it is displayed. For example, you might only want to print certain columns, specify different headings for them, and change the sequence in which the columns are printed.

To edit the template, select the Edit template option on the Print Utility panel. The Column Selection/Edit panel is displayed.

Running Print in batch
If you want to print large DB2 tables, you can perform the print in a batch job.

If you select the Batch execution option on the Print Utility panel, the Print utility builds the necessary JCL to perform the print function in a batch job. The generated JCL is displayed using the ISPF editor, where you can review the JCL and submit the job for batch processing.

Related tasks
"Editing the template for the “From” table” on page 195
"Specifying a DB2 object name” on page 25
Printing the contents of a DB2 table

To print rows from a DB2 table, use:

- The RD ("Record Dump") primary command to print the data in dump format
- The RP ("Record Print") primary command to print the data in character format

Examples

Here is an example of RD output when the display format is SNGL:

```
<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>Print</td>
<td>Browse</td>
<td>KEISTE2.FMN.LIST</td>
<td>Top of 64</td>
</tr>
<tr>
<td>Command</td>
<td>===&gt;</td>
<td>Scroll CSR</td>
<td></td>
</tr>
<tr>
<td>--------</td>
<td>---------</td>
<td>-------------</td>
<td>--------</td>
</tr>
<tr>
<td>--------</td>
<td>---------</td>
<td>-------------</td>
<td>--------</td>
</tr>
</tbody>
</table>
```

**Figure 61. Sample printed output from RD primary command (SNGL display format)**

Here is an example of RD output when the display format is TABL:
Printing rows from a DB2 table

Here is an example of RP output when the display format is SNGL:

![Sample printed output from RD primary command (TABL display format)](image)

Here is an example of RP output when the display format is SNGL:

![Sample printed output from RP primary command (SNGL display format)](image)

Figure 62. Sample printed output from RD primary command (TABL display format)

Figure 63. Sample printed output from RP primary command (SNGL display format)

Here is an example of RP output when the display format is TABL:

![Sample printed output from RP primary command (TABL display format)](image)
Looking at the print output from your FM/DB2 session

To browse, and optionally purge, the print output from your current FM/DB2 editor session held in the print data set:

- Issue the PB primary command, or
- Use the Print Browse utility function (3.11)

You specify the print data set using the PRINTDSN entry field on the Set Print Processing Options panel.

If you use the PB primary command or the Print Browse utility when you have not created any print output, FM/DB2 displays the message Empty print data set.

The following primary commands are available with Print Browse:

- FIND
- LOCATE
- PRINT
- PURGE

To select the Print Browse utility:

1. From the Primary Option Menu panel, enter 3 (Utilities) to display the Utility Functions panel.
2. From the Utility Functions panel, enter 11 (Print browse)

Figure 65 on page 267 shows an example of some printed output displayed by Print Browse.
Looking at the print output from your FM/DB2 session

Figure 65. Print Browse: sample output

<table>
<thead>
<tr>
<th>EMPNO</th>
<th>EMPLOYEE_NAME</th>
<th>MIDDLEINIT</th>
<th>EMPLOYEE_LASTNAME</th>
<th>WORKDEPT</th>
<th>PHONE_NO</th>
<th>HIREDATE</th>
<th>JOB</th>
</tr>
</thead>
<tbody>
<tr>
<td>000010</td>
<td>CHRISTINE I. HAAS</td>
<td>A00</td>
<td>3978</td>
<td>1965-01-01</td>
<td>PRES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>000020</td>
<td>Michael L. THOMPSON</td>
<td>B01</td>
<td>3476</td>
<td>1973-10-10</td>
<td>MANAGER</td>
<td></td>
<td></td>
</tr>
<tr>
<td>000030</td>
<td>SALLY A. Kwan</td>
<td>C01</td>
<td>4738</td>
<td>1975-04-05</td>
<td>MANAGER</td>
<td></td>
<td></td>
</tr>
<tr>
<td>000050</td>
<td>JOHN B. GEYER</td>
<td>E01</td>
<td>6789</td>
<td>1949-08-17</td>
<td>MANAGER</td>
<td></td>
<td></td>
</tr>
<tr>
<td>000060</td>
<td>IRVING F. STERN</td>
<td>D11</td>
<td>6423</td>
<td>1973-09-14</td>
<td>MANAGER</td>
<td></td>
<td></td>
</tr>
<tr>
<td>000070</td>
<td>EVA D. PULASKI</td>
<td>D21</td>
<td>7831</td>
<td>1980-09-30</td>
<td>MANAGER</td>
<td></td>
<td></td>
</tr>
<tr>
<td>000090</td>
<td>NEIL N. HENDERSON</td>
<td>E11</td>
<td>5F98</td>
<td>1970-08-15</td>
<td>MANAGER</td>
<td></td>
<td></td>
</tr>
<tr>
<td>000100</td>
<td>THEODORE Q. SPENGER</td>
<td>E21</td>
<td>0972</td>
<td>1980-06-19</td>
<td>MANAGER</td>
<td></td>
<td></td>
</tr>
<tr>
<td>000110</td>
<td>VINCENZO G. LUCCHESI</td>
<td>A00</td>
<td>3490</td>
<td>1958-05-16</td>
<td>SALESREP</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Printing an audit trail report

To print an audit trail report using the contents of the audit trail data set, use the Audit trail utility function (3.10).

An audit trail data set is created if the Create an audit trail option is selected when you use:

- View (option 1)
- Edit (option 2)
- Copy (option 3.3)

**Note:** You can create an audit trail data set yourself using the batch job FMNSMF04 (supplied with FM/DB2). This job creates an audit trail data set using data from the SMF log file. For details, see the File Manager Customization Guide.

To display the Print Audit Trail panel,
1. From the Primary Option Menu panel, enter 3 (Utilities) to display the Utility Functions panel.
2. From the Utility Functions panel, enter 10 (Audit trail)

To print an audit trail report:
Utility functions: Audit trail (option 3.10)

1. Specify the name of the audit trail data set that contains the audit trail data you want to print (Data set name).
   If FM/DB2 created the audit trail data set (as a result of FM/DB2 view, edit, or copy activity), the name is in the format
   
   \( \text{prefix}.\text{FMN2AUD}.\text{ssid}.\text{Dyymmdd}.\text{Tthmmss} \)

   or

   \( \text{userid}.\text{FMN2AUD}.\text{ssid}.\text{Dyymmdd}.\text{Tthmmss} \)

   where:

   \( \text{prefix} \) is the TSO prefix of the user, if it exists.

   \( \text{userid} \) is the ISPF shared pool variable ZUSER. This is typically the same as the user's TSO logonid.

   \( \text{FMN2AUD} \) is a constant that identifies the data set as an audit trail

   \( \text{ssid} \) is the DB2 subsystem ID

   \( \text{Dyymmdd} \) is the date FM/DB2 created the audit trail data set

   \( \text{Tthmmss} \) is the time FM/DB2 created the audit trail data set

   Note: If the AUDITHLQ option is set in FMN0POPT (see File Manager Customization Guide), audit trail data sets have the name

   \( \text{audithlq}.\text{FMNLOG}.\text{Dyymmdd}.\text{Tthmmss} \).

   If you created the audit trail data set yourself, use the data set name you specified in the supplied batch job FMNSMFX. For details, see the File Manager Customization Guide.

2. To help identify your audit trail report, use the Description entry field. If the description contains imbedded spaces, you must enclose it in quotation marks. The description you enter is displayed on the Formatted Audit Event Records section of the report.

3. Select the processing options you want:

   Print only changed columns
   Limits the amount of printed lines.

   Keep data set after printing
   Whether to keep or delete the data set after it is printed.

   Browse report
   The audit trail report is displayed using Print Browse (option 3.11).

   Note: The contents of the audit trail report are transferred to a SYSOUT class as specified by the PRINTOUT print option on the Set Print Processing Options panel. To browse the audit trail report, set the PRINTOUT print option to SYSOUT=c.

   Batch execution
   Generates JCL for printing the audit trail report using a batch job. For command syntax, see "AUD (Print Audit Trail Report)" in the File Manager User's Guide and Reference.

4. Press Enter.
Audit Trail Report

The audit trail report is divided into two sections: Formatted Audit Events Records (Figure 66) and Summary Statistics (Figure 67 on page 270).

Audit Trail Report

The first part of the Formatted Event Records section shows:
- The FM/DB2 function being audited (View, Edit, or Copy).
- The name of the data set holding the audit log records.
- The report description (if specified).
- Location information for the DB2 object being manipulated.

The second part of the Formatted Event Records section shows information for each audit event (each SQL call is defined as a separate audit event):
- The audit event number.
- A date and timestamp.
- The SQL statement.
- The DB2 column name.
- The type of key (if applicable).
- The original value of the column (if the column has been changed or deleted).
- The new value of the column (if the column has been changed or inserted).

If Print only changed fields was selected on the Print Audit Trail panel, only columns where the data has changed are shown.

Every FM/DB2 view, edit, or copy of a DB2 object is recorded as a separate session. If SMF recording is used, many sessions can be logged. Summary statistics are provided for each session, not the entire period being audited.
Utility functions: Audit trail (option 3.10)

Summary Statistics

IBM File Manager for z/OS DB2 Component  Audit Trail Report
Summary Statistics

Data base opened on 2004-09-18 at 15.52.00 by userid: FMUSER
Data base closed on 2004-09-18 at 15.55.47

Total SQL statements: 1
  Rows inserted: 0
  Rows deleted: 0
  Rows replaced: 0
TOTAL Data base changes: 0
  Records bypassed: 0

Audit Trail Report Options
  Maximum Print Lines per Page: 60
  Print Only Changed Fields on an Update: NO

Figure 67. Audit trail report: Summary Statistics section

The Summary Statistics section shows:

- The times the database was opened and closed.
- The user ID.
- Counts of the number of SQL calls, inserts, deletes, and row replacements for the FM/DB2 session.
- The total number of update events.

Related tasks

- "View and Edit options (options 1 and 2)" on page 43
- "Setting options for the current FM/DB2 editor session" on page 95
- "Copy utility options (option 3.3)" on page 44
- Chapter 4, "Viewing and changing DB2 data," on page 91
- "Copying data from one DB2 object to another" on page 193
- "Looking at the print output from your FM/DB2 session" on page 266
- "Print settings (option 0.1)" on page 41

Related references

- "Editor Options (1 of 8) panel" on page 528
- "Print Audit Trail panel" on page 618
Chapter 10. Managing DB2 privileges

Use the Privileges utility function to display, grant, and revoke DB2 privileges.

To perform these functions, you use the Manage DB2 Privileges panel.

The available functions for managing DB2 privileges are shown in Table 23.

Table 23. Privilege functions

<table>
<thead>
<tr>
<th>Privilege Type</th>
<th>Grant</th>
<th>Revoke</th>
</tr>
</thead>
<tbody>
<tr>
<td>Database</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Table space</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Table/View</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Column</td>
<td>Yes</td>
<td>No1</td>
</tr>
<tr>
<td>Plan</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Package</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Collection</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Schema</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Distinct type</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Function</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Stored procedure</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Note:
1. You cannot revoke privileges on specific columns. You must revoke the appropriate table privileges.

To display the Manage DB2 Privileges panel,
1. From the Primary Option Menu panel, enter 3 (Utilities) to display the Utility Functions panel.
2. From the Utility Functions panel, enter 5 (Privileges)

Related tasks

“Displaying privileges” on page 273
“Granting privileges” on page 273
“Revoking privileges” on page 275

Related references

“Manage DB2 Privileges panel” on page 609
“Primary Option Menu panel” on page 617
“Utility Functions panel” on page 720

Using the Manage DB2 Privileges panel

The Manage DB2 Privileges panel consists of two sections:
Object Identification Criteria
Use this section of the panel to specify object identification criteria to identify the objects you want listed. FM/DB2 lists all objects, restricted by any such criteria, for which privileges exist. Input in this section depends on the **Object Type** you specify.

Location
Leave this field blank if the objects you want listed are at your current location. Otherwise, type the full name of the remote location, an asterisk (*) to display all available remote locations, or a pattern using wildcard characters. If you specify an asterisk on its own or a pattern, FM/DB2 displays a selection list showing the locations that match the input specification.

To select the location you want, type `S` against the required entry and press Enter.

**Note:** You can only specify a location if displaying privileges (command line blank).

Owner
To select objects regardless of their owner, leave this field blank. Otherwise, to restrict the list of objects by owner, type either the full name of the owner or a pattern using wildcard characters. If this entry field is not blank, FM/DB2 limits the objects listed to those matching the owner selection criteria you specify.

**Note:** You can only specify an owner for table or column object types.

Name
To select objects regardless of their name, leave this field blank. Otherwise, to restrict the list of objects by name, type either the full name of the object or a pattern using wildcard characters. If this entry field is not blank, FM/DB2 limits the objects listed to those matching the name selection criteria you specify.

In
This is an optional field you can use to qualify the following object types:

<table>
<thead>
<tr>
<th>Object type</th>
<th>Qualify by specifying the...</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table space</td>
<td>Name of the database</td>
</tr>
<tr>
<td>Column</td>
<td>Name of the table</td>
</tr>
<tr>
<td>Package</td>
<td>Name of the collection</td>
</tr>
<tr>
<td>Distinct type</td>
<td>Name of the schema</td>
</tr>
<tr>
<td>Function</td>
<td>Name of the schema</td>
</tr>
<tr>
<td>Stored procedure</td>
<td>Name of the schema</td>
</tr>
</tbody>
</table>

Object Type
Use this section to identify the type of DB2 object against which privilege operations are performed. The available functions for managing DB2 privileges are shown in Table 23 on page 271.

To select the object type you want, type the appropriate number in the **Object Type** selection field. The default value is 3 (table).

You select whether to display, grant, or revoke privileges by what, if anything, you type on the command line:
Displaying privileges

You can display privilege details for any of the object types listed on the Manage DB2 Privileges panel.

To display privilege details:
1. Type the required details (if any) in the Object Identification Criteria section.
2. Type the appropriate number in the Object Type selection field to select the object type you want, or use the default value, 3 (table).
3. Leave the command line blank.
4. Press Enter. The privilege display panel for the specified privilege types is displayed.

Granting privileges

You can grant (create new) privileges for any of the object types listed on the Manage DB2 Privileges panel.

To grant one or more privileges for an object, you can use any of the following methods:
• Method 1 (specifying the object using the grant privileges panel):
  1. On the Manage DB2 Privileges panel:
     a. Type the required details (if any) in the Object Identification Criteria section.
Granting privileges

b. Type the appropriate number in the **Object Type** selection field to select the object type you want, or use the default value, 3 (table).

2. Type 6 on the command line.

3. Press Enter.

   The grant privileges panel for the specified object type is displayed. The panel lists each privilege type, with a corresponding selection entry field, that can apply to the object type.

   On the grant privileges panel:
   a. Select the privileges you want to grant by entering “Y” or “G” in the corresponding selection fields.
   b. In the entry field (or fields) provided, specify the name of the object.
   c. In the **To** entry field, specify the SQLID of the user to whom the privileges are to be granted (the grantee).
   d. Press Enter.

- **Method 2 (specifying the object from a privilege display panel):**

  **Note:** This method only works if there are already privileges for the DB2 object. If not, FM/DB2 issues the message “No rows selected”.

1. On the Manage DB2 Privileges panel,
   a. Type the required details (if any) in the **Object Identification Criteria** section.
   b. Type the appropriate number in the **Object Type** selection field to select the object type you want, or use the default value, 3 (table).

2. Leave the command line blank.

3. Press Enter. The privilege display panel for the specified object type is displayed listing the selected objects.

4. On the privilege display panel, type 6 in the **Cmd** field against the object for which you want to grant privileges.

5. Press Enter. The grant privileges panel for the specified object is displayed.

   **Note:** If you use this method to navigate to the grant privileges panel for the specified object type, the following details are pre-entered:
   - The existing setting for each privilege type (Y, G, or blank)
   - The name details for the object
   - The SQLID of the user to whom the existing privileges are granted (the grantee)

   On the grant privileges panel:
   a. Select the privileges you want to grant by entering “Y” or “G” in the corresponding selection fields.
   b. Press Enter.

- **Method 3 (using the Object List utility):**

1. Use the utility function, Object List (option 3.4), to display a list of the type of objects you want.

2. On the object list panel for the selected object type, type 6 in the **Cmd** field against the object for which you want to grant privileges.

3. Press Enter.

   The grant privileges panel for the specified object type is displayed (as for Method 1). The panel lists each privilege type, with a corresponding selection entry field, that can apply to the object type.

4. Follow the remaining steps as for Method 1 (3a to 3d)
Granting privileges

If the grant privilege processing is successful, FM/DB2 displays a message indicating that the grant ran successfully. Otherwise, a panel is displayed showing the formatted SQL return code.

Related tasks
- “Displaying privileges” on page 273
- “Revoking privileges”
- “Displaying or printing a list of objects” on page 242

Related references
- “Manage DB2 Privileges panel” on page 609
- “Primary Option Menu panel” on page 617
- “Utility Functions panel” on page 720
- “Grant privileges panels” on page 587

Revoking privileges

You can revoke (remove existing) privileges for any of the object types listed on the Manage DB2 Privileges panel apart from column. (You cannot revoke privileges on specific columns. You must revoke the appropriate table privileges.)

To revoke one or more privileges for an object, you can use either of the following methods:

- Method 1 (specifying the object using the revoke privileges panel):
  1. On the Manage DB2 Privileges panel,
     a. Type the required details (if any) in the Object Identification Criteria section.
     b. Type the appropriate number in the Object Type selection field to select the object type you want, or use the default value, 3 (table).
  2. Type R on the command line.
  3. Press Enter.

The revoke privileges panel for the specified object type is displayed. If there is more than one type of privilege that can apply to the object type, the panel lists each privilege type with a corresponding entry field.

On the revoke privileges panel:
- Select the privileges you want to revoke by entering any character in the corresponding selection fields.

  Note: For object types where there is only one type of privilege, this step does not apply.
- In the entry field provided (for some types of privilege there are two entry fields), specify the name of the object.
- In the From entry field, specify the SQLID of the user for whom the privileges are to be revoked (the revokee).
- In the By entry field, specify the SQLID of the user who granted the privileges (the grantor), or “ALL” for all grantors.
- Press Enter.

- Method 2 (specifying the object from a privilege display panel):
  1. On the Manage DB2 Privileges panel,
     a. Type the required details (if any) in the Object Identification Criteria section.
Revoking privileges

b. Type the appropriate number in the Object Type selection field to select the object type you want, or use the default value, 3 (table).

2. Leave the command line blank.

3. Press Enter. The privilege display panel for the specified object type is displayed listing the selected objects.

4. On the privilege display panel, type R in the Cmd field against the object for which you want to revoke the privileges.

5. Press Enter. The revoke privileges panel for the specified object is displayed.

Note: If you use this method to navigate to the revoke privileges panel (for the specified object type), the following details are pre-entered:
- The existing setting for each privilege type (Y, G, or blank)
- The name details for the object
- The SQLID of the user for whom the privileges are to be revoked (the revokee) in the From entry field.

On the revoke privileges panel:

a. Select the privileges you want to revoke by entering any character in the corresponding selection fields.

   Note: For object types where there is only one type of privilege, this step does not apply.

b. In the By entry field, specify the SQLID of the user who granted the privileges (the grantor), or “ALL” for all grantors.

c. Press Enter.

If the revoke privilege processing is successful, FM/DB2 displays a message indicating that the revoke ran successfully. Otherwise, a panel is displayed showing the formatted SQL return code.

Related tasks

- Displaying privileges” on page 273
- Granting privileges” on page 273

Related references

- Manage DB2 Privileges panel” on page 609
- Primary Option Menu panel” on page 617
- Utility Functions panel” on page 720
- Revoke privileges panels” on page 647
Chapter 11. Generating batch JCL for DB2 utility jobs

To generate batch JCL to execute the DB2 utility jobs listed below, use the Utilities utility function (3.9). You do not need to know the utility control statements for these operations.

- COPY
- LOAD
- REBUILD
- RECOVER
- REORG
- RUNSTATS
- UNLOAD

For an explanation of the uses of these DB2 utility jobs, refer to the DB2 Utility Guide and Reference.

Note: To successfully execute the generated utility job, you must have the appropriate DB2 privileges on any DB2 objects that are referenced.

The main panel for the Utilities utility is the DB2 Utilities panel.

To display the DB2 Utilities panel,

1. From the Primary Option Menu panel, enter 3 (Utilities) to display the Utility Functions panel.
2. From the Utility Functions panel, enter 9 (Utilities)

Use the fields on the DB2 Utilities panel to select:

- The type of DB2 utility you want to run
- The type of DB2 object that the utility processes
- Details of the particular DB2 objects you want to run a utility on

When you press Enter, all objects at the local DB2 server are displayed, unless you entered information in the “DB2 Object Details” fields on the DB2 Utilities panel in which case the displayed rows are limited to the objects that match the information you have specified.

The JCL for the DB2 utility job is generated and presented in an ISPF edit session. You can make any required changes prior to submitting the job, or use the ISPF editor commands to save a copy of the JCL for later use.

Table 24 show which DB2 utilities you can use with which DB2 objects.

<table>
<thead>
<tr>
<th>DB2 utility</th>
<th>Use with these DB2 objects</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1. Tables</td>
</tr>
<tr>
<td>1. COPY</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Table 24. DB2 utilities you can use with DB2 objects (continued)

<table>
<thead>
<tr>
<th>DB2 utility</th>
<th>Use with these DB2 objects</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1. Tables</td>
</tr>
<tr>
<td>2. LOAD</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>(see “LOAD utility” on page 286)</td>
</tr>
<tr>
<td>3. REBUILD</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>(see “REBUILD utility (indexes)” on page 293)</td>
</tr>
<tr>
<td>4. RECOVER</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>(see “RECOVER (table spaces)” on page 295)</td>
</tr>
<tr>
<td>5. REORG</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>(see “REORG (table spaces)” on page 298)</td>
</tr>
<tr>
<td>6. RUNSTATS</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>(see “RUNSTATS (table spaces)” on page 303)</td>
</tr>
<tr>
<td>7. UNLOAD</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>(see “UNLOAD (tables)” on page 306)</td>
</tr>
</tbody>
</table>

**Related tasks**
- “COPY utility” on page 283
- “LOAD utility” on page 286
- “REBUILD utility” on page 292
- “RECOVER utility” on page 294
- “REORG utility” on page 298
- “RUNSTATS utility” on page 302
- “UNLOAD utility” on page 306
- “Generating a DB2 utility job” on page 280

**Related references**
- “DB2 Utilities panel” on page 515
- “Primary Option Menu panel” on page 617
- “Utility Functions panel” on page 720

### Setting options for DB2 utilities

You can set DB2 utility options to:
- Insert JCL DD statements for specified LISTDEF or TEMPLATE libraries
- Modify the DB2 utility functions to include DD name clauses with specified template names
- Insert LISTDEF, TEMPLATE, and OPTIONS DB2 utility statements
Generating DB2 utility jobs

To change the option settings for DB2 utilities, first select 0 (Settings) from the Primary Option menu to display the Set Processing Options panel. The new settings apply for the current subsystem only. When you reconnect to the same subsystem, your option settings are retrieved.

To reset the option settings back to the installation defaults (for the current DB2 subsystem), use the RESET primary command or the Reset function key (F6).

The Set Processing Options panel shows menu items for:

Unload utility options
The FM/DB2 Unload Utility options apply when the UNLOAD utility is used to transfer data.

Options utility options
The FM/DB2 Options utility options assist in generating DB2 OPTIONS utility control statements.

Listdef utility options
The FM/DB2 Listdef utility options assist in generating DB2 LISTDEF utility control statements and in generating JCL for utility batch jobs to access a LISTDEF library.

Template utility options
The FM/DB2 Template utility options assist in generating DB2 TEMPLATE utility control statements and in generating JCL for utility batch jobs to access a template library. Also, options are provided that allow you to specify template names for DD names clauses of various DB2 utility control statements.

For a full description of DB2 utilities, see the DB2 Utility Guide and Reference.

Before you generate batch JCL to execute a DB2 Utility, check that these options are set to the values you want.

FM/DB2 uses the above options to generate JCL that refers to the appropriate LISTDEF and TEMPLATE libraries and also, where applicable, to generate an OPTIONS, LISTDEF, or TEMPLATE utility statement in the utility batch job. If specified, FM/DB2 places the OPTIONS, LISTDEF, and TEMPLATE statements in the SYSIN deck before the utility statement that corresponds to the utility selected from the DB2 Utilities panel.

To change the option settings for a DB2 utility:
1. From either the Set Processing Options panel or the Options pull-down menu, select the menu item for the options you want to change:
   - 11 Unload
   - 12 Options
   - 13 Listdef
   - 14 Template
   FM/DB2 displays the selected options panel.
2. Change the option settings as necessary

Related tasks
"Setting default processing options" on page 39

Related references
"Set Processing Options panel" on page 671
Generating a LISTDEF statement in the DB2 utility job

The DB2 Utilities panel shows a **Specification** section and a **LISTDEF name** entry field. You can use these entry fields to specify the name of a LISTDEF.

If you select LISTDEF in the **Specification** section, you must also specify a LISTDEF name as the object of the utility.

To use a LISTDEF name as the object of the utility:
1. In the **Specification** section, select **2. LISTDEF**
2. Enter the name of the LISTDEF in **LISTDEF name**

**Note:** The LOAD syntax does not have a LISTDEF clause. If you select LOAD in the **DB2 Utility** section and LISTDEF in the **Specification** section, FM/DB2 issues an error message.

Related references

“DB2 Utilities panel” on page 515

Generating a DB2 utility job

To generate a DB2 utility job:

1. Specify the DB2 utility you want to generate:
   1. COPY
   2. LOAD
   3. REBUILD
   4. RECOVER
   5. REORG
   6. RUNSTATS
   7. UNLOAD

2. Specify the object type:
   1. Tables
   2. Table spaces
   3. Index spaces
   4. Indexes
   5. Indexes for table spaces

3. For COPY, REBUILD, RECOVER, REORG, RUNSTATS, and UNLOAD (that is, all the FM/DB2 DB2 utilities except LOAD), you can specify that a LIST clause is to be included in the JCL generated by FM/DB2 for the nominated DB2 utility.

   To generate a LIST clause in the JCL for the DB2 utility, select **2. LISTDEF**; otherwise, select **1. Object name**.

4. You now need to supply details about either the object or the LISTDEF for the DB2 utility job:
Generating DB2 utility jobs

- If you specified **1. Object name** in the Specification section (Step 3), optionally, specify details about the object by using one or more of the four filter fields:
  - **Database**
  - **Table/Index space**
  - **Owner** (Table or Index)
  - **Name** (Table or Index)

FM/DB2 validates any details you enter and carries them forward to the next panel.

For each of the preceding filter fields, you can enter:
- The full name of the item.
- An asterisk (*) to display a list of items from which you can select the one you want.
  
  If you enter an asterisk (*) in any of the four filter fields, FM/DB2 displays the respective selection list panel from which you can select a value to put into the filter field. The records displayed in these panels are restricted by any other related filter you have previously entered. Each combination of utility and DB2 object results in certain valid filter combinations. Incorrect filter combinations are rejected.
- A pattern to display a list of all items whose name matches the pattern specification.
- Blanks. No filtering effect occurs.

- If you specified **2. LISTDEF** in the Specification section (Step 3), enter the name to be used in the LIST clause in **LISTDEF name**.

5. Press Enter.

FM/DB2 displays the panel for the specified DB2 utility and object type.

If you specified **1. Object name** in the Specification section (Step 3), the panel shows the qualifying list of rows, with various input and output fields. Select the rows you want and change the values in any of the editable fields as required.

If you specified **2. LISTDEF** in the Specification section (Step 3), FM/DB2 displays an alternative panel for the DB2 utility. See "Working with LISTDEF utility panels" on page 282.

6. Press Enter.

FM/DB2 displays the generated JCL which you can edit if you need to before you submit it as a job. (The LOAD and UNLOAD utilities are exceptions to this rule, as both utilities show another series of panels, before displaying the generated JCL.) Figure 68 on page 282 shows an example of generated JCL.
7. To submit the JCL as a job stream for batch processing, use the SUBMIT command.

The following sections describe each of the DB2 utilities that FM/DB2 supports.

**Related tasks**

- "Using an asterisk (*) or a pattern in entry fields" on page 28
- "COPY utility" on page 283
- "LOAD utility" on page 286
- "REBUILD utility" on page 292
- "RECOVER utility" on page 294
- "REORG utility" on page 298
- "RUNSTATS utility" on page 302
- "UNLOAD utility" on page 306

**Working with LISTDEF utility panels**

When you specify 2. LISTDEF in the Specification section and the name of the LISTDEF in **LISTDEF name** of one of the DB2 utility panels (excluding LOAD), FM/DB2 displays an alternative panel showing the name of the LISTDEF. You use the LISTDEF form of the utility panel to specify one or more names to be used in the LIST clause in the JCL generated for the DB2 utility.

Initially the panel shows a single line containing the name you specified in the **LISTDEF name** field on the DB2 utility panel.

You can manipulate the panel rows as necessary by entering line commands in the CMD field to insert, repeat, delete, or select rows.
Generating DB2 utility jobs

An asterisk (*) in the PRC column indicates the LISTDEF in the row is selected. Use the S command to toggle unselected rows to selected and selected rows to unselected.

You cannot select a row where the LISTSDEF Name field is blank.

To display the LISTDEF Selection pop-up panel, type an asterisk (*) in the LISTSDEF Name field.
Select a LISTDEF name from the displayed list by entering an S in the Sel field.

```
FMN2PLOC Row 1 to 4 of 4
Select one from the list of the last twelve entries.

Sel LISTDEF
  NORMAL
  EXAMPLE1
  JOHNS
  MYLIST
*************** Bottom of data ***********************
```

To create a utility batch job with the selected LISTDEFS, press Enter.

For more information about the LISTDEF name, see the LIST keyword in the DB2 Utility Guide and Reference.

COPY utility

To select the COPY utility, enter 1 in the DB2 utility input field on the DB2 Utilities panel.

If you selected “Table spaces” in the Object Type section of the DB2 Utilities panel, the COPY Utility (Table spaces) panel is displayed.

If you selected “Index spaces” in the Object Type section of the DB2 Utilities panel (and did not specify name of a LISTDEF), one of two forms of the COPY Utility (Index spaces) panel is displayed. The format of the panel depends on which, if any, of the filter fields you specified on the DB2 Utilities panel.

Note: If you specified the name of a LISTDEF on the DB2 Utilities panel, FM/DB2 displays an alternative form of the COPY Utility (Table spaces) panel or COPY Utility (Index spaces) panel.

Related tasks
“Default values” on page 284
“COPY (table spaces)” on page 284
“COPY (index spaces)” on page 285

Related references
“DB2 Utilities panel” on page 515
“COPY Utility (Table Spaces) panel” on page 427
“COPY Utility (Table Spaces) with LISTDEF panel” on page 429
Default values

The default naming conventions used for image copy data sets in the generated JCL are:

- \&ZUSER, \&DBNAME, and \&PARTNO are self-explanatory.
- \&OBJNAME is either the table space name or the index space name.
- \&PRE can be:
  - F Full
  - I Incremental
  - C Change Limit. This can result in DB2 performing a full or incremental imagecopy.
- \&FMN2DTIM contains \texttt{jjjhhmm}, where with \texttt{jjj} is the Julian day, and \texttt{hhmm} is the time.

For example, OSPEED.UTILDB1.T1RIX1.F00000.D1011044

COPY (table spaces)

This option corresponds to the COPY TABLE SPACE form of the COPY utility control statement.

- If you did \textit{not} specify the name of a LISTDEF on the DB2 Utilities panel:
  - If you entered information in the Database name or Table/index space name filter fields on the DB2 Utilities panel, the displayed rows are limited to the DB2 objects that match the information you specified.
  - To select a table space from those displayed, type S in the Sel column. An asterisk (*) in the PRC column indicates selected table spaces.
  - You can modify the COPY utility statement by changing the values in the following input fields:
    - FCP
    - Change Limits
    - RPO
  - \textit{Figure 68 on page 282} shows an example of generated JCL for copying a table space.

- If you \textit{did} specify the name of a LISTDEF on the DB2 Utilities panel:
  - FM/DB2 displays an alternative form of the COPY Utility (Table Spaces) panel showing the specified LISTDEF and a number of editable fields that correspond to keywords in the COPY statement.
  - \textit{Figure 69 on page 285} shows an example of generated JCL for copying a table space where a LISTDEF has been specified.
Related references

“COPY Utility (Table Spaces) panel” on page 427
“COPY Utility (Table Spaces) with LISTDEF panel” on page 429
“DB2 Utilities panel” on page 515

COPY (index spaces)

This option corresponds to the COPY INDEXSPACE form of the COPY utility control statement.

- If you did not specify the name of a LISTDEF on the DB2 Utilities panel:
  - When you use COPY (index spaces), you can use the following combinations of filter fields on the DB2 Utilities panel:
    - Database name, or Table/index space name.
    - Database name, Table/index owner, or Table/index name.
  - If you entered information in any of the preceding filter fields on the DB2 Utilities panel, the displayed rows are limited to the DB2 objects that match the information you specified.
  - To select an index space from those displayed, type 5 in the Sel column. An asterisk (*) in the PRC column indicates selected table spaces.
  - The COPY INDEXSPACE statements generated by FM/DB2 specify the database and index space name (rather than the index owner and index name), regardless of the way you specified the index to be copied. Figure 70 on page 286 shows an example of generated JCL for copying an index space.
Generating DB2 utility jobs

If you did specify the name of a LISTDEF on the DB2 Utilities panel:

- FM/DB2 displays an alternative form of the COPY Utility (Index Spaces) panel showing the specified LISTDEF and a number of editable fields that correspond to keywords in the COPY statement.

Related references

“COPY Utility (Index Spaces) panel” on page 421
“COPY Utility (Index Spaces) with LISTDEF panel” on page 423
“DB2 Utilities panel” on page 515

LOAD utility

This option corresponds to the DB2 LOAD utility control statement.

To select the LOAD utility, on the DB2 Utilities panel:

1. Enter 2 (LOAD) in the DB2 Utility entry field.
2. Enter 1 (Tables) in the Object Type entry field.

The LOAD Utility panel is displayed.

Note: You can also use the Import (option 3.6) utility to load data from a QSAM or VSAM data set into a DB2 table, either in foreground or in batch.

Related tasks

“Copying data from a VSAM or QSAM file” on page 198
“Input data” on page 287
“Loading data using LOAD utility control statements” on page 287
“Loading data described by a copybook or template” on page 289

Related references
Input data

The FM/DB2 process that builds the LOAD utility JCL requires the input data to be in a fixed-length format, as it builds fixed-position field specification statements.

You can use the LOAD utility where the data to be loaded is:

- Described by existing LOAD utility control statements.
  
  For example, the REORG utility can be used to unload DB2 data into DB2 “unload” format by specifying the UNLOAD EXTERNAL option. The REORG utility also produces a file containing DB2 LOAD utility control statements to (re)load the data into the original tables.

- Described by a copybook or template.
  
  You must specify the name of a copybook or FM/DB2 template that describes the input data. The format of the data needs to be consistent with the requirements of the LOAD utility.

Related tasks

- “REORG utility” on page 298
- “Setting options for DB2 utilities” on page 278
- “Loading data using LOAD utility control statements” on page 289
- “Loading data described by a copybook or template” on page 289

Loading data using LOAD utility control statements

**Note:** To load data in this way, the default INDDN name (in the **INDDN template name** field on the second **TEMPLATE Options** panel) must be blank.

To load data from a previous DB2 “unload” operation:

1. In the entry fields for input data, specify the name of the data set (**Data set name**), and optionally the name of the member (**Member**), containing the data to be loaded.

2. In the entry fields for utility control statements, specify the name of the data set (**Data set name**), and optionally the name of the member (**Member**), containing the utility control statements.

3. Select option 1 (“The data is described by utility control statements”)

4. Press Enter.

   FM/DB2 constructs a batch job to run the DB2 LOAD utility job using the input details you have specified. Figure 71 on page 288 shows an example of generated JCL for loading a table using data in DB2 “unload” format.
Generating DB2 utility jobs

Figure 71. Sample JCL generated for LOAD Utility using LOAD utility control statements

Related tasks
"Loading data described by a copybook or template” on page 289

Related references
"LOAD Utility panel” on page 603
Loading data described by a copybook or template

You can use the LOAD utility to load data in a sequential data set, providing a copybook or template exists that describes the format of the input data.

In this case, you use a series of panels:

- On the LOAD Utility panel, you specify the name of the data set containing the input data, and the name of the target table.
- On the second panel, you specify the “From” copybook or template that describes the input data.
- On the third panel, you can specify the “To” template that describes the target table. Alternatively, you can use FM/DB2 to generate a template for the target table using the information in the DB2 catalog.

At this stage, you can define the mapping between fields in the input template and fields in the output template.

To load data described by a copybook or template:

1. In the entry fields for input data, specify the name of the data set (Data set name), and optionally the name of the member (Member), containing the data to be loaded.
2. In the entry fields for the target DB2 table (Table owner, Table name, and optionally Database and Table space), specify the name of the table into which the data is to be loaded. In any of these fields, you can enter either an asterisk (*) to display a selection list, or wildcards.
3. Select option 2 (“The data is described by a template or copybook”)
4. Press Enter.

If you have specified a default name for an INDDN clause (in the INDDN template name field on the second TEMPLATE Options panel), FM/DB2 displays an interim panel showing the default name (for example, LODINDDN) in the Template name field for the input data:

FM/DB2 displays a panel where you can specify the input template or copybook.
5. Specify the “From” template details. Perform one of these actions:
   • Specify a copybook or template:
     – In the entry fields for the input template or copybook, specify the name
       of the data set (Data set name), and optionally the name of the member
       (Member), for the template or copybook that describes the format of the
       input data.
     – Select 1 (Above) for the Template usage option.
   • Use the last (previously used) template:
     – Select 2 (Previous) for the Template usage option.

6. To edit the template for the input data before proceeding to the LOAD utility
   “To” template panel, select the View template option.
   You can select this option by entering either a “/” or an “A”.

Note: The template edit operations have no effect when the template is used
to generate DB2 LOAD utility control statements. If you select this option, you
cannot update and save changes to the template.

7. Press Enter.
   FM/DB2 displays a panel where you can specify the template for the target
   DB2 table.

8. Specify the “To” template details using a combination of the To Template and
    Template usage entry fields. You can:
   • Specify a template that describes the format of the input data:
     – Use Data set name, and optionally Member, to specify the template.
     – Select 1 (Above).
   • Use the last (previously used) template:
     – Select 2 (Previous).
   • Use FM/DB2 to generate a template using the information in the DB2
     catalog:
     – Select 3 (Generate from table).
   • Use FM/DB2 to generate a template using the information in the DB2
     catalog and store the generated template using the name you specify (if the
     member already exists, FM/DB2 replaces it):
     – Use Data set name, and optionally Member, to specify the name that
       you want FM/DB2 to use when storing the generated template.
     – Select 4 (Generate/Replace).

9. To change the options for the current LOAD session, select Edit load utility
    options by entering either a “/” or an “A”.
   FM/DB2 displays the Load Utility Options panel for editing.

   Note: Any options you set on this panel apply for the current LOAD session
   only. When you exit from the LOAD session, the load options revert to the
   global LOAD options.

10. To map the columns (or fields) in the “From” template to the columns in the
    “To” template, select Edit load utility options by entering either a “/” or an
    “A”. FM/DB2 displays the Template Mapping panel.

11. Press Enter.
    FM/DB2 constructs a batch job to run the DB2 LOAD utility job using the
    input details you have specified. Figure 72 on page 291 shows an example of
    generated JCL for loading a table using data described by a copybook or
Generating DB2 utility jobs

template.

Figure 72. Sample JCL generated for LOAD Utility using data described by a copybook or template (continued in next figure)

Related tasks

"Loading data using LOAD utility control statements" on page 287
"Mapping data" on page 80
"Selecting options on FM/DB2 panels" on page 24
"Editing a template" on page 57
REBUILD utility

This option corresponds to the DB2 REBUILD INDEX utility control statement.

To select the REBUILD utility, enter 3 in the DB2 Utility input field on the DB2 Utilities panel.

If you selected Table spaces in the Object Type section of the panel, the REBUILD Utility (Indexes) panel is displayed.

If you selected Indexes or Index spaces in the Object Type section of the panel, the REBUILD Utility (Indexes for Table Spaces) panel is displayed.

Note: If you specified the name of a LISTDEF on the DB2 Utilities panel, FM/DB2 displays an alternative form of the REBUILD Utility (Indexes) panel or REBUILD Utility (Indexes for Table Spaces) panel.

Related tasks
- “REBUILD utility (indexes)” on page 293
- “REBUILD utility (indexes for table spaces)” on page 293

Related references
- “DB2 Utilities panel” on page 515
- “REBUILD Utility (Indexes) panel” on page 630
- “REBUILD (Indexes) with LISTDEF panel” on page 632
- “REBUILD (Indexes for Table Spaces) panel” on page 628
- “REBUILD (Indexes for Table Spaces) with LISTDEF panel” on page 629

REBUILD utility (indexes)

To include a WORKDDN clause in the generated REBUILD statement, specify the required template name in WORKDDN template name on the second TEMPLATE Options panel.

- If you did not specify the name of a LISTDEF on the DB2 Utilities panel:
  - If you entered information in the Database name, Table/index owner, or Table/index name filter fields on the DB2 Utilities panel, the displayed rows are limited to the DB2 objects that match the information you specified.
  - To select an index from those displayed, type S in the Sel column. An asterisk (*) in the PRC column indicates selected indexes.
  - Figure 73 on page 293 shows an example of generated JCL for rebuilding indexes.

In this example, FM/DB2 has generated two JCL steps, one for each index selected from the displayed rows on the previous panel.
If you did specify the name of a LISTDEF on the DB2 Utilities panel:

- FM/DB2 displays an alternative form of the REBUILD Utility (Indexes) panel showing the specified LISTDEF and a number of editable fields that correspond to keywords in the REBUILD statement.

Related tasks

- "REBUILD utility (indexes for table spaces)"

Related references

- "DB2 Utilities panel" on page 515
- "REBUILD Utility (Indexes) panel" on page 630
- "REBUILD (Indexes) with LISTDEF panel" on page 632
- "TEMPLATE Options (2 of 2) panel" on page 708

**REBUILD utility (indexes for table spaces)**

- If you did not specify the name of a LISTDEF on the DB2 Utilities panel:
  - If you entered information in the Database name or Table/index space name filter fields on the DB2 Utilities panel, the displayed rows are limited to the DB2 objects that match the information you specified.
  - To select a table space from those displayed, type S in the Sel column. An asterisk (*) in the PRC column indicates selected table spaces.
- Figure 74 on page 294 shows an example of generated JCL for rebuilding indexes for table spaces.
Generating DB2 utility jobs

- If you did specify the name of a LISTDEF on the DB2 Utilities panel:
  - FM/DB2 displays an alternative form of the REBUILD (Indexes for Table Spaces) panel showing the specified LISTDEF and a number of editable fields that correspond to keywords in the REBUILD statement.

Related tasks

“REBUILD utility (indexes)” on page 292

Related references

“DB2 Utilities panel” on page 515
“REBUILD (Indexes for Table Spaces) panel” on page 628
“REBUILD (Indexes for Table Spaces) with LISTDEF panel” on page 629

RECOVER utility

The RECOVER utility option generates JCL to run the DB2 RECOVER utility job.

To select the RECOVER utility, enter 4 in the DB2 Utility input field on the DB2 Utilities panel.

If you selected Table spaces in the Object Type section of the panel, the RECOVER Utility (Table spaces) panel is displayed.

If you selected Index spaces in the Object Type section of the panel, the RECOVER Utility (Index spaces) panel is displayed.
Note: If you specified the name of a LISTDEF on the DB2 Utilities panel, FM/DB2 displays an alternative form of the RECOVER Utility (Table Spaces) panel or RECOVER Utility (Index Spaces) panel.

Related tasks
- “RECOVER (table spaces)”
- “RECOVER (index spaces)” on page 297

Related references
- “DB2 Utilities panel” on page 515
- “RECOVER Utility (Table Spaces) panel” on page 636
- “RECOVER Utility (Table Spaces) with LISTDEF panel” on page 637
- “RECOVER Utility (Index Spaces) panel” on page 634
- “RECOVER Utility (Index Spaces) with LISTDEF panel” on page 635

FM/DB2 only supports the following form of the RECOVER utility statement:

```
RECOVER object TOCOPY datasetname TOVOLUME CATALOG
```

**RECOVER (table spaces)**

This option corresponds to the RECOVER TABLE SPACE form of the RECOVER utility control statement.

- If you did not specify the name of a LISTDEF on the DB2 Utilities panel:
  - The information displayed in the RECOVER Utility (Table Spaces) panel includes one row for every matching entry found in the SYSCOPY catalog table. The match is performed on database name and table space name, and the type (Typ) of the entry must be either “F” (Full image copy) or “I” (Partial image copy). Only cataloged image copy data sets are supported. This means that there can be 0, 1, or many entries for any table space name.
  - If you entered information in the Database name or Table/index space name filter fields on the DB2 Utilities panel, the displayed rows are limited to the DB2 objects that match the information you specified.
  - To select a table space from those displayed, type S in the Sel column. An asterisk (*) in the PRC column indicates selected table spaces.
  - Figure 75 on page 296 shows an example of generated JCL for recovering table spaces.
Generating DB2 utility jobs

If you did specify the name of a LISTDEF on the DB2 Utilities panel:

- FM/DB2 displays an alternative form of the REBUILD Utility (Indexes) panel showing the specified LISTDEF and a number of editable fields that correspond to keywords in the RECOVER statement.

- Figure 76 shows an example of generated JCL for recovering table spaces where a LISTDEF has been specified.

Figure 75. Sample JCL generated for RECOVER Utility (table spaces)

- If you did specify the name of a LISTDEF on the DB2 Utilities panel:
  - FM/DB2 displays an alternative form of the REBUILD Utility (Indexes) panel showing the specified LISTDEF and a number of editable fields that correspond to keywords in the RECOVER statement.
  - Figure 76 shows an example of generated JCL for recovering table spaces where a LISTDEF has been specified.

Figure 76. Sample JCL generated for RECOVER Utility (table spaces) with LISTDEF specified

Related tasks
RECOVER (index spaces)

This option corresponds to the RECOVER INDEXSPACE form of the RECOVER utility control statement.

- If you did not specify the name of a LISTDEF on the DB2 Utilities panel:
  - If you entered information in the Database name or Table/index space name filter fields on the DB2 Utilities panel, the displayed rows are limited to the DB2 objects that match the information you specified.
  - To select an index space from those displayed, type S in the Sel column. An asterisk (*) in the PRC column indicates selected index spaces.
  - Figure 77 shows an example of generated JCL for recovering index spaces.

- If you did specify the name of a LISTDEF on the DB2 Utilities panel:
  - FM/DB2 displays an alternative form of the RECOVER Utility (Index Spaces) panel showing the specified LISTDEF and a number of editable fields that correspond to keywords in the RECOVER statement.

Related references

"DB2 Utilities panel” on page 515
"RECOVER Utility (Table Spaces) panel” on page 636
"RECOVER Utility (Table Spaces) with LISTDEF panel” on page 637

Figure 77. Sample JCL generated for RECOVER Utility (index spaces)

- If you did specify the name of a LISTDEF on the DB2 Utilities panel:
  - FM/DB2 displays an alternative form of the RECOVER Utility (Index Spaces) panel showing the specified LISTDEF and a number of editable fields that correspond to keywords in the RECOVER statement.
REORG utility

To select the REORG utility, enter 5 in the **DB2 Utility** input field on the DB2 Utilities panel.

If you selected Table spaces in the **Object Type** section of the panel, the REORG Utility (Table spaces) panel is displayed.

If you selected Indexes in the **Object Type** section of the panel, the REORG Utility (Indexes) panel is displayed.

**Note:** If you specified the name of a LISTDEF on the DB2 Utilities panel, FM/DB2 displays an alternative form of the REORG Utility (Table spaces) panel or REORG Utility (Indexes) panel.

**Related tasks**

"REORG (table spaces)"  
"REORG (indexes)" on page 300

**Related references**

"DB2 Utilities panel” on page 515  
"REORG Utility (Table Spaces) panel” on page 644  
"REORG Utility (Table Spaces) with LISTDEF panel” on page 646  
"REORG Utility (Indexes) panel” on page 642  
"REORG Utility (Indexes) with LISTDEF panel” on page 643

REORG (table spaces)

To include a UNLDDN clause in the generated REORG statement, specify the required template name in **UNLDDN template name** for the REORG TABLESPACE statement on the second TEMPLATE Options panel.

**Note:** The **Unload External** option on the REORG Utility (Table Spaces) panel for the selected table space must be set to “Y”.

This option corresponds to the REORG TABLE SPACE utility control statement.

- If you did *not* specify the name of a LISTDEF on the DB2 Utilities panel:
  - If you entered information in the **Database name** or **Table/index space name** filter fields on the DB2 Utilities panel, the displayed rows are limited to the DB2 objects that match the information you specified.
  - To select a table space from those displayed, type 5 in the Sel column. An asterisk (*) in the PRC column indicates selected table spaces.
  - **Figure 78 on page 299** shows an example of generated JCL for reorganizing table spaces.
If you did specify the name of a LISTDEF on the DB2 Utilities panel:
- FM/DB2 displays an alternative form of the REORG (Table Spaces) panel showing the specified LISTDEF and a number of editable fields that correspond to keywords in the REORG statement.
- Figure 79 on page 300 shows an example of generated JCL for recovering table spaces where a LISTDEF has been specified.
Generating DB2 utility jobs

File Edit Edit_Settings Menu Utilities Compilers Test Help

EDIT FMNUSER.SPFTEMP1.CNTL

File Manager/DB2 Data V13R1 User's Guide

Figure 79. Sample JCL generated for REORG Utility (table spaces) with LISTDEF specified

Related tasks

- "REORG (indexes)"
- "Setting options for DB2 utilities" on page 278

Related references

- "DB2 Utilities panel" on page 515
- "TEMPLATE Options (2 of 2) panel" on page 708
- "REORG Utility (Table Spaces) panel" on page 644
- "REORG Utility (Table Spaces) with LISTDEF panel" on page 646

REORG (indexes)

This option corresponds to the REORG INDEX utility control statement.

- If you did not specify the name of a LISTDEF on the DB2 Utilities panel:
  - If you entered information in the Database name, Table/index owner, or Table/index name filter fields on the DB2 Utilities panel, the displayed rows are limited to the DB2 objects that match the information you specified.
  - To select an index from those displayed, type $ in the Sel column. An asterisk (*) in the PRC column indicates selected indexes.
  - Figure 80 on page 301 shows an example of generated JCL for reorganizing indexes.

Generating DB2 utility jobs

File Edit Edit_Settings Menu Utilities Compilers Test Help

EDIT FMNUSER.SPFTEMP1.CNTL

File Manager/DB2 Data V13R1 User's Guide

Figure 79. Sample JCL generated for REORG Utility (table spaces) with LISTDEF specified

Related tasks

- "REORG (indexes)"
- "Setting options for DB2 utilities" on page 278

Related references

- "DB2 Utilities panel" on page 515
- "TEMPLATE Options (2 of 2) panel" on page 708
- "REORG Utility (Table Spaces) panel" on page 644
- "REORG Utility (Table Spaces) with LISTDEF panel" on page 646

REORG (indexes)

This option corresponds to the REORG INDEX utility control statement.

- If you did not specify the name of a LISTDEF on the DB2 Utilities panel:
  - If you entered information in the Database name, Table/index owner, or Table/index name filter fields on the DB2 Utilities panel, the displayed rows are limited to the DB2 objects that match the information you specified.
  - To select an index from those displayed, type $ in the Sel column. An asterisk (*) in the PRC column indicates selected indexes.
  - Figure 80 on page 301 shows an example of generated JCL for reorganizing indexes.

Generating DB2 utility jobs

File Edit Edit_Settings Menu Utilities Compilers Test Help

EDIT FMNUSER.SPFTEMP1.CNTL

File Manager/DB2 Data V13R1 User's Guide

Figure 79. Sample JCL generated for REORG Utility (table spaces) with LISTDEF specified

Related tasks

- "REORG (indexes)"
- "Setting options for DB2 utilities" on page 278

Related references

- "DB2 Utilities panel" on page 515
- "TEMPLATE Options (2 of 2) panel" on page 708
- "REORG Utility (Table Spaces) panel" on page 644
- "REORG Utility (Table Spaces) with LISTDEF panel" on page 646

REORG (indexes)

This option corresponds to the REORG INDEX utility control statement.

- If you did not specify the name of a LISTDEF on the DB2 Utilities panel:
  - If you entered information in the Database name, Table/index owner, or Table/index name filter fields on the DB2 Utilities panel, the displayed rows are limited to the DB2 objects that match the information you specified.
  - To select an index from those displayed, type $ in the Sel column. An asterisk (*) in the PRC column indicates selected indexes.
  - Figure 80 on page 301 shows an example of generated JCL for reorganizing indexes.
If you did specify the name of a LISTDEF on the DB2 Utilities panel:

- FM/DB2 displays an alternative form of the REORG (indexes) panel showing the specified LISTDEF and a number of editable fields that correspond to keywords in the REORG statement.
- Figure 81 on page 302 shows an example of generated JCL for recovering table spaces where a LISTDEF has been specified.

Figure 80. Sample JCL generated for REORG Utility (indexes)

- If you did specify the name of a LISTDEF on the DB2 Utilities panel:
  - FM/DB2 displays an alternative form of the REORG (indexes) panel showing the specified LISTDEF and a number of editable fields that correspond to keywords in the REORG statement.
  - Figure 81 on page 302 shows an example of generated JCL for recovering table spaces where a LISTDEF has been specified.
RUNSTATS utility

To select the RUNSTATS utility, enter 6 in the DB2 Utility input field on the DB2 Utilities panel.

If you selected Table spaces in the Object Type section of the panel, the Runstats Utility (Table spaces) panel is displayed.

If you selected Indexes in the Object Type section of the panel, the Runstats Utility (Indexes) panel is displayed.

Note: If you specified the name of a LISTDEF on the DB2 Utilities panel, FM/DB2 displays an alternative form of the Runstats Utility (Table spaces) panel or Runstats Utility (Indexes) panel.

Related tasks

"RUNSTATS (table spaces)" on page 303
RUNSTATS (table spaces)

This option corresponds to the RUNSTATS TABLE SPACE form of the RUNSTATS utility control statement.

- If you did not specify the name of a LISTDEF on the DB2 Utilities panel:
  - If you entered information in the Database name or Table/index space name filter fields on the DB2 Utilities panel, the displayed rows are limited to the DB2 objects that match the information you specified.
  - To select a table space from those displayed, type S in the Sel column. An asterisk (*) in the PRC column indicates selected table spaces.
  - Figure 82 shows an example of generated JCL for gathering statistics on a table space.

- If you did specify the name of a LISTDEF on the DB2 Utilities panel:
  - FM/DB2 displays an alternative form of the RUNSTATS Utility (Table Spaces) panel showing the specified LISTDEF and a number of editable fields that correspond to keywords in the RUNSTATS statement.
  - Figure 83 on page 304 shows an example of generated JCL for recovering table spaces where a LISTDEF has been specified.
Generating DB2 utility jobs

**RUNSTATS (indexes)**

This option corresponds to the RUNSTATS INDEX form of the RUNSTATS utility control statement.

- If you did not specify the name of a LISTDEF on the DB2 Utilities panel:
  - If you entered information in the **Database name, Table/index owner, or Table/index space name** filter fields on the DB2 Utilities panel, the displayed rows are limited to the DB2 objects that match the information you specified.
  - To select an index from those displayed, type **S** in the **Sel** column. An asterisk (*) in the **PRC** column indicates selected indexes.
  - **Figure 84 on page 305** shows an example of generated JCL for gathering statistics on indexes.
If you did specify the name of a LISTDEF on the DB2 Utilities panel:
- FM/DB2 displays an alternative form of the RUNSTATS Utility (Indexes) panel showing the specified LISTDEF and a number of editable fields that correspond to keywords in the RUNSTATS statement.
- Figure 85 shows an example of generated JCL for recovering table spaces where a LISTDEF has been specified.

Figure 85. Sample JCL generated for RUNSTATS (indexes) with LISTDEF specified
UNLOAD utility

The UNLOAD utility unloads data from one or more source objects to one or more sequential data sets in external format. It is the preferred method for unloading or transferring large amounts of data.

You can specify the DB2 templates that identify the data set to receive the LOAD utility control statements (PUNCHDDN clause) and the data sets into which the data is unloaded (UNLDDN clause).

To select the UNLOAD utility, enter 7 in the DB2 utility input field on the DB2 Utilities panel.

If you selected “Tables” in the Object Type section of the DB2 Utilities panel, the UNLOAD Utility (Tables) panel is displayed.

If you selected “Table spaces” in the Object Type section of the DB2 Utilities panel, the UNLOAD Utility (Table Spaces) panel is displayed.

Note: If you specified the name of a LISTDEF on the DB2 Utilities panel, FM/DB2 displays an alternative form of the UNLOAD Utility (Tables) panel or UNLOAD Utility (Table Spaces) panel.

UNLOAD (tables)

Use the UNLOAD (tables) utility to unload data into a sequential data set. You can choose to replace any existing data with the new data, or to add the new data to the data already in the data set. The UNLOAD (tables) utility provides the most efficient means of adding large amounts of data to a data set.

When you use the UNLOAD utility, you can:

- Select the columns to be unloaded by specifying the required columns in the “From” template
Generating DB2 utility jobs

- Reformat data during the unload by mapping table columns to fields in the output data set (as defined by a template)
- Create and save a File Manager "base" template that describes the unloaded data
- Select which rows to unload
- Specify the number of rows to be unloaded

To include a PUNCHDDN clause in the generated UNLOAD statement, specify the required template name in **PUNCHDDN template name** on the second TEMPLATE Options panel.

To include a UNLDDN clause in the generated UNLOAD statement, specify the required template name in **UNLDDN template name** for the UNLOAD statement on the second TEMPLATE Options panel.

The following description applies if you did not specify the name of a LISTDEF on the DB2 Utilities panel. If you did specify the name of a LISTDEF on the DB2 Utilities panel, see “UNLOAD (tables) with LISTDEF specified” on page 311.

On the UNLOAD Utility (Tables) panel:
1. The table name details you entered on the DB2 Utilities panel are copied to the UNLOAD Utility (Tables) panel. If you do not want to unload all of the rows in the source table, specify the number of rows you want to unload in the **Unload Count** field.
2. If you are using a specific template for the source table, specify the name of the template in the **From Template** entry fields.
3. In the **Processing Options**, specify the template you want FM/DB2 to use for the source table, and whether you want to edit the template before FM/DB2 builds the utility batch job.
4. To proceed, press Enter.

   If you selected the **Edit template** option, FM/DB2 displays the Column Selection/Edit panel. Use this panel to restrict the data in the source table by:
   - Selecting or deselecting specific columns for unloading
   - Specifying row selection criteria. FM/DB2 selects only the rows whose contents satisfy the criteria for unloading

For example, the template shown in Figure 86 on page 308 has been edited to select certain columns and only rows whose contents meet certain criteria. If you use this template as the “From” template for an unload, the data unloaded is restricted to:
   - The columns **FIRSTNME**, **LASTNAME**, **WORKDEPT**, and **JOB**, but only where **WORKDEPT** equals “MNT” and **JOB** equals “PAINTER”.

Chapter 11. Generating batch JCL for DB2 utility jobs  307
If you did not select the **Edit template** option (or have completed the edit of the template option), and the value of the **UNLDDN template name** field on the TEMPLATE Options (2 of 2) panel is blank, FM/DB2 displays the UNLOAD Utility “To” panel.

**Note:** If the **UNLDDN template name** field is not blank, FM/DB2 does not display the UNLOAD Utility “To” panel as the output is already defined.

5. To temporarily change the settings for the UNLOAD options that FM/DB2 uses in building the current batch job, select **Edit UNLOAD utility options** to display the UNLOAD Options panel.

If you change any of the original settings, they remain in effect only for as long as you use the DB2 Utilities UNLOAD feature in which the settings were changed. If you exit from the UNLOAD Utility (Tables) panel and return to this panel later, the settings changed in the previous UNLOAD session are unavailable.

6. On the UNLOAD Utility “To” panel, specify the sequential data set that is the target of the unload by entering the “To” data set details using the **Data set name** and, optionally, **Volume**, entry fields.

7. If you are using a specific template or copybook for the “To” data set, you can specify the name of the template or copybook data set and, optionally, the member name, in the **To Data Template or Copybook** entry fields.

The data format used by the DB2 UNLOAD utility is fixed. Therefore, if you specify a copybook or template data set, the copybook or template should reflect the DB2 UNLOAD data format to avoid data mismatch problems.

When a template is specified for the DB2 unload data set, it must be a File Manager "base" template; a FM/DB2 template cannot be used.

When the **Allow padding** option on the DB2 UNLOAD utility options panel is not selected, FM/DB2 cannot generate a template that describes the unloaded data. In this situation, all fields related to template processing are protected when the UNLOAD utility “To” panel is displayed and the template usage option is set to 5 (NONE).

8. Select the **Template usage** and **Disposition** options you want.
9. You can review the DB2 UNLOAD options by selecting the **View UNLOAD options** field. No changes can be made when the DB2 UNLOAD utility options panel is displayed. To make changes return to the DB2 UNLOAD utility “From” panel and select **Edit UNLOAD options**.

10. To edit the template mapping for the source table, select **Edit template mapping** to display the Template Mapping panel.

11. To proceed, press Enter.

   FM/DB2 builds a batch job to execute the DB2 UNLOAD utility job using the specified input details. 

   [Figure 87 on page 310](#) shows an example of the generated JCL for unloading a table object.
Generating DB2 utility jobs

Figure 87. Sample JCL generated for UNLOAD (Tables)

Related tasks
- Chapter 3, “Working with templates,” on page 49
- “Editing a template” on page 57
- “Letting FM/DB2 do the mapping” on page 81
- “Setting options for DB2 utilities” on page 278

Related references
- “Column Selection/Edit panel” on page 407
- “UNLOAD Utility (Tables) ’From’ panel” on page 718
Null indicator byte processing

The DB2 UNLOAD utility data format uses a single byte for the DB2 null value, located immediately before the data for each nullable column, in the output data set. The value in this byte is set to X'00' if the column contains a value and X'FF' when the column contains the DB2 null value. The DB2 UNLOAD utility automatically places the appropriate value in the null indicator byte in the output data set.

FM/DB2 automatically generates a File Manager "base" template when the template processing option is set to 3 or 4 on the “To” panel. This template includes fields for the null indicator bytes as appropriate.

When a copybook is used to describe the output data, FM/DB2 attempts to generate a reasonable mapping between the template for the DB2 object and the template for the output data set. The data format used by the DB2 UNLOAD utility is fixed, therefore the data format described by the copybook should reflect this format, to avoid data mismatch problems.

The DB2 null indicator byte in a user-specified copybook is successfully mapped when:

- It immediately precedes a nonvarying target field, or precedes the length field of a varying target field.
- The null indicator byte does not overlap any field that has already been mapped.
- The field name for the null indicator byte contains "NULL". In this case, FM/DB2 documents the mapping as comments in the generated JCL for the DB2 UNLOAD utility.

Other types of mapping, or null indicator byte usage are not supported, including:

- A copybook that would map a nullable DB2 column to a data field without a null indicator byte.
- A copybook that would map a not-nullable DB2 column to a data field with a null indicator byte.
- A copybook that uses null indicator fields that are not one byte in length, or are not located immediately before the data for the column to which they pertain.

Related references

UNLOAD (tables) with LISTDEF specified

If you specified the name of a LISTDEF on the DB2 Utilities panel, FM/DB2 displays an alternative form of the UNLOAD Utility (Tables) panel showing the specified LISTDEF and a number of editable fields that correspond to keywords in the UNLOAD statement.

Figure 88 on page 312 shows an example of generated JCL for unloading a table object where a LISTDEF has been specified.
Related references

"UNLOAD Utility (Tables) with LISTDEF panel" on page 719

UNLOAD (table spaces)

This option corresponds to the UNLOAD TABLESPACE form of the UNLOAD utility control statement.

When you use UNLOAD (table spaces), you can use the following combinations of filter fields on the DB2 Utilities panel:

- Database name, or Table/index space name.

If you entered information in any of the preceding filter fields on the DB2 Utilities panel, the displayed rows are limited to the DB2 objects that match the information you specified.

Figure 89 on page 313 shows an example of the UNLOAD Utility (Table Spaces) panel with a list of table spaces.
To select an index space from those displayed, type S in the Sel column. An asterisk (*) in the PRC column indicates selected table spaces.

If only a single row results from the filter values you entered, the row is marked as selected, and is shown with an asterisk (*) in the PRC field.

If you entered a table space name containing wildcards, the ISPF part of the UNLOAD Utility panel is initialized with one row for each of the table space names that match the input specification. Note that if you did not specify a database name on the DB2 utilities panel, or specified a generic name, the list may contain table spaces from more than one database.

To select a table spaces to copy, type an “S” in the SEL field. Selected table spaces have an asterisk (*) in the PRC field.

You can modify the UNLOAD utility statement by changing the value in the Part'n Range input field.

In Figure 89, the input shows that a range of partitions (from 2 to 3) is to be unloaded.

Figure 90 on page 314 shows an example of generated JCL for unloading a table object where a LISTDEF has been specified.
UNLOAD (table spaces) with LISTDEF specified

If you specified the name of a LISTDEF on the DB2 Utilities panel, FM/DB2 displays an alternative form of the UNLOAD Utility (Table Spaces) panel showing the specified LISTDEF and a number of editable fields that correspond to keywords in the UNLOAD statement.

Figure 91 on page 315 shows an example of generated JCL for unloading a table space object where a LISTDEF has been specified.
Generating DB2 utility jobs

Figure 91. Sample JCL generated for UNLOAD (Table Space) with LISTDEF specified

Related references

"UNLOAD Utility (Table Spaces) with LISTDEF panel" on page 717
Generating DB2 utility jobs
Chapter 12. Working with SQL statements

When you want to:
- Issue any dynamic SQL statement from your panel or from a data set, or
- Build and run an SQL SELECT statement interactively

you can use the SQL prototyping, execution and analysis option (4).

Note:
1. This chapter does not attempt to describe the syntax of SQL statements. For full details, see the DB2 for z/OS SQL Reference.

To display the SQL Prototyping, Execution and Analysis panel, from the Primary Option Menu panel select option 4 (SQL).

Related tasks
- “Should you use basic or advanced SQL prototyping?”
- “Using basic SQL prototyping”
- “Using advanced SQL prototyping” on page 325

Related references
- “Primary Option Menu panel” on page 617
- “SQL Prototyping, Execution and Analysis panel” on page 680

Should you use basic or advanced SQL prototyping?

If you only need to build a simple SQL SELECT statement, basic SELECT prototyping 4.1) is probably sufficient for your needs. If you need to build a more complex statement, use advanced SELECT prototyping (4.2).

Related references
- “Using basic SQL prototyping”
- “Using advanced SQL prototyping” on page 325

Using basic SQL prototyping

Basic SELECT prototyping uses two panels: you use the primary panel to specify the table (or tables) that contain the data you want to retrieve (FROM clause), and the secondary panel to specify the columns you want to show, any qualifying information (WHERE clause), and the sequence in which you want the data to be shown (ORDER BY clause).

To display the primary Basic SELECT Prototyping panel, select option 1 (Basic) on the SQL Prototyping, Execution and Analysis panel.
Using basic SQL prototyping

On this panel you can specify up to 15 tables, views, synonyms, or aliases that can be part of the SELECT statement being prototyped. If you specify more than one object, the columns for all objects specified are shown on the secondary Basic SELECT Prototyping panel when you press enter - see Figure 93 on page 319.

You cannot modify the entry-line numbers at the left of the panel (1, 2, and so on). The purpose of the entry-line number is to provide a suffix for the correlation name used to identify the DB2 object. DB2 correlation names are discussed in detail in the SQL Reference. The complete DB2 correlation name is shown in the "Tab" column on the secondary Basic SELECT Prototyping panel when you press enter. The complete correlation name will be a capital letter, followed by the entry-line number. The capital letter will usually be 'T', but may be another letter if any of the DB2 object names start with 'T'. The Basic Prototyper ensures that the letter selected for the correlation name reference is not the same as the first letter of any DB2 object name included on the primary Basic SELECT Prototyping panel.

The columns and rows included in the result table are determined by the SELECT and WHERE clauses specified on the secondary Basic SELECT Prototyping panel, which is displayed when you specify the objects and press Enter. The examples in this section use the tables DSN81010.EMP and DSN81010.DEPT.

Tables and views are specified by optionally entering values in the Location, Database, Tbl spc. (table space), and Owner fields, and entering values in the Name fields. Synonyms are specified by entering values in the Name fields. If a value is entered in the Owner field for a synonym, it must be the current SQLID. All of the fields support catalog searches by specifying a pattern in the field. After you have specified the table names on the primary Basic SELECT Prototyping panel, press Enter to display the secondary Basic SELECT Prototyping panel.
When initially displayed, the secondary Basic SELECT Prototyping panel shows all of the columns for the DB2 objects entered on the primary panel. You can scroll up or down if there are more columns than will fit on the display. The SQL statement being prototyped is shown at the top of the panel as 4 clauses (SELECT, FROM, WHERE and ORDER BY). An empty clause is indicated by a “?”.

FM/DB2 automatically updates one or more of these clauses as you enter either primary or line commands. Each of the clause fields is an ISPF expandable field. If the field is too short to display all of the clause then a “+” will be shown at the end of the field. You can position the cursor anywhere in the field and press the ISPF EXPAND key to show the complete clause in a pop-up window.

To limit the number of rows returned in the result set when the SQL statement is executed, specify the maximum number of rows in the **Row count** field.

The following primary commands are available when you use basic SELECT prototyping:

**ALL**    Use this command to select all columns for a DB2 object. You can specify an optional parameter #n to include only the columns of the nth DB2 object displayed.

**CANCEL** Use this command to return to the primary Basic SELECT Prototyping panel

**EXECUTE** Use this command to run the SQL statement being prototyped

**RESET** Use this command to abandon all changes to the SQL statement. All clauses are returned to empty status.

**SQL**    Use this command to show the complete SQL statement in an ISPF editor session

**UNDO**    Use this command to back-out the last change made.
Using basic SQL prototyping

The following line commands can be entered against any column:

- **S or /**
  You type either S or / against the entry for a column to add that column to the list of columns for the SQL statement. Data from the column will be included in the result table when the statement is executed.
  You can select multiple columns at once, however your entries are processed from top to bottom, so you might need to select columns individually if the order of columns in the result table is important.

- **A and D**
  You type either A or D against the entry for a column to add an ORDER BY clause for that column. If the specified column does not exist in the column list, the column is also added to the column list. Data from the column will be included in the result table when the statement is executed, and rows will be sorted in ascending or descending order on that data.
  Entering a A results in the ORDER BY clause including the column name without modification, this defaults to ascending (ASC keyword).
  Entering a D results in the ORDER BY clause including the column name followed by the DESC keyword, this indicates the result table should be sorted in descending order based on the data for the column.

- **U**
  You type U against the entry for a column to remove the last column reference from the list of columns in the result table. When the column is removed and the column is no longer specified in the list of columns:
  - All ORDER BY clause references to the column are also removed.
  - All predicate references to the column are also removed.
  If the column has been added to the column list multiple times, use the U command repeatedly to remove all references.

- **UO**
  You type UO against the entry for a column to remove the last ORDER BY reference.
  Removing an ORDER BY reference for the column does not remove the column from the column list, nor does it remove any predicates referencing the column from the WHERE clause.
  If the column has been added to the ORDER BY clause multiple times, use the UO command repeatedly to remove all references.

- **UW**
  You type UW against the entry for a column to remove the last predicate that references the column.
  Removing a WHERE clause reference for the column does not remove the column from the column list, nor does it remove any references for the column from the ORDER BY clause.
  If there are multiple predicates containing the column, use the UW command repeatedly to remove all references.

**Specifying a predicate**

You specify a predicate (for example EMPNO = '000010') by typing the comparison operator ("=") in the Op field, and the comparison value ("000010") in the Value field. To specify a join predicate, include the correlation reference for the table containing the join column in the value field.
Using basic SQL prototyping

For example, to specify a join between DSN81010.EMP.WORKDEPT and DSN81010.DEPT.DEPTNO:
- Type “=” in the Op field for column WORKDEPT.
- Type “T2.DEPTNO” in the Value column for WORKDEPT.

Figure 94 shows an example of specifying the join, and Figure 95 shows the result of the join predicate.

Example of using basic SELECT prototyping

We'll use an example to show how to use basic SELECT prototyping to build a SELECT statement and then run it to see the results.
Using basic SQL prototyping

Suppose you want to list details of all employees in the table DSN8610.EMP
whose:
  • Department is not “E01”, and
  • JOB is “Manager” or SALARY is at least $40,000

For each employee, you want to show the following columns from the table
DSN8610.EMP (correlation name #1):
  LASTNAME
  FIRSTNME
  WORKDEPT
  JOB
  SALARY (in ascending sequence)

and the following column from the table DSN8610.EEMP (correlation name #2):
  BONUS

You can specify details on the secondary Basic SELECT Prototyping panel for more
than one column at a time. However, for the purpose of clarity, the following steps
show the details being entered one at a time for each column, and the resultant
SELECT statement that FM/DB2 progressively builds (and displays at the top of
the panel) after you press Enter.

Step 1
Select the column, LASTNAME, by typing S in the S entry field:

FM/DB2 updates the generated SELECT statement by:
  • Adding the name of the selected column (LASTNAME) to the SELECT clause.

Step 2
Select the column, FIRSTNME, by typing S in the S entry field:

FM/DB2 updates the generated SELECT statement by:
  • Adding the name of the selected column (FIRSTNME) to the SELECT clause.
Step 3

To exclude employees in WORKDEPT “E01”:

- Select the column, WORKDEPT, by typing $ in the $ entry field,
- Type the expression operator $<> in the Op entry field, and
- Type E01 in the Value entry field:

```
$ LOp ( Tab Column Name   Data Type(length) Op Value   )
$       ___   #1 WORKDEPT   CHAR(3)      <>   e01
```

FM/DB2 updates the generated SELECT statement by:
- Adding the name of the selected column (WORKDEPT) to the SELECT clause.
- Adding a predicate, #1.WORKDEPT <> 'E01', to the WHERE clause.

```
SELECT #1.LASTNAME, #1.FIRSTNME, #1.WORKDEPT
FROM DSN8610.EMP #1, DSN8610.EEMP #2
WHERE #1.WORKDEPT <> 'E01'
ORDER BY ?
```

Step 4

To select employees with a JOB of “Manager”:

- Select the column, JOB, by typing $ in the $ entry field,
- Type an opening parenthesis, (, in the ( entry field, and
- Type MANAGER in the Value entry field:

```
$ LOp ( Tab Column Name   Data Type(length) Op Value   )
$       ___   #1 JOB       CHAR(8)      manager
```

FM/DB2 updates the generated SELECT statement by:
- Adding the name of the selected column (JOB) to the SELECT clause.
- Inserting the default logical operator AND before adding a predicate, (#1.JOB = 'MANAGER', to the WHERE clause.

```
SELECT #1.LASTNAME, #1.FIRSTNME, #1.WORKDEPT, #1.JOB
FROM DSN8610.EMP #1, DSN8610.EEMP #2
WHERE #1.WORKDEPT <> 'E01' AND (#1.JOB = 'MANAGER'
ORDER BY ?
```

Step 5

To also select employees with a SALARY of at least $40,000 (regardless of their JOB), and to specify that the results are to be shown in ascending sequence of SALARY:

- Select the column, SALARY, for ascending sequence by typing A in the $ entry field,
- Type the logical operator, OR, in the LOp entry field,
- Type the expression operator $>= in the Op entry field, and
- Type 40000 in the Value entry field.

```
```
Using basic SQL prototyping

FM/DB2 updates the generated SELECT statement by:
- Adding the name of the selected column (SALARY) to the SELECT clause.
- Adding a predicate, OR #1.SALARY >= 40000, to the WHERE clause.
- Adding the name of the selected column (SALARY), and the keyword ASC, to the ORDER BY clause.

Step 6

Select the column, BONUS (from the table DSN8610.EEMP), by typing S in the S entry field:

FM/DB2 updates the generated SELECT statement by:
- Adding the name of the selected column (BONUS) to the SELECT clause.

Figure 96 on page 325 shows how the Basic SELECT Prototyping panel is displayed if you had entered the details for all of the columns at the same time.

Note: The effect of entering details for one column at a time (as in the previous steps) can be different to when you enter details for more than one column at a time.

For example, if you enter the details as shown in Figure 96 on page 325, FIRSTNME is shown before LASTNAME in the generated SELECT statement.
Using basic SQL prototyping

Using advanced SQL prototyping

In this section, the following terms are used:

clause  A distinct part of the SELECT statement, such as a WHERE clause.

element  A single item within the syntax (such as a keyword, variable, reference to a fragment, or special character).

fragment  A portion of the SELECT statement syntax. Fragments can contain references to other fragments.

instance  A repeatable item that has been assigned a value. For example, the following GROUP BY clause contains three instances:

GROUP BY COL1, COL2, COL3

keyword  An item that must be spelled exactly as shown. Keywords (for example, FROM) are shown in uppercase in the syntax diagram displayed on the Advanced SELECT Prototyping panel.
Using advanced SQL prototyping

repeatable item
An item that can be repeated within the syntax. Repeatable items are indicated by a repeat arrow that can contain a repeat separator. For example:

```
GROUP BY column
```

variable
An element that requires you to enter a value.

Advanced SELECT prototyping lets you develop a complex SELECT statement using a recursive set of specification panels.

Each panel displays a fragment of the SELECT statement syntax, presented in the form of a syntax diagram (in a similar way to a reference manual). You build the SELECT statement by selecting the optional fragments of the syntax you want to use, discarding those that you do not need, and entering items where required. The panels guide you through the syntax and ensure that you include all the required fragments of the syntax. Each panel shows the partly-formed SELECT statement as you progressively build it.

As you build the SELECT statement, you can return to any part of the statement to:
- Change values.
- Add a fragment of syntax.
- Remove a fragment of syntax.

Note:
1. If you have a 24-line screen, to see the SQL statement as it is being built you must remove the function key lines from your display by entering PF$HOW OFF.
2. If the generated SQL statement extends over several lines, you can scroll forward or backward with the Forward function key (F8) and Backward function key (F7) respectively.

To display the Advanced SELECT Prototyping panel, select option 2 (Advanced) on the SQL Prototyping, Execution and Analysis panel.

The following primary commands are available when you use advanced SELECT prototyping:
- CANCEL
- DESELECT
- EXECUTE
- INSRPT
- NEXTRPT
- PREVRPT
- SQL

Related tasks
- “Example 2 (Changing the SQL statement)” on page 333
- “Example 3 (Adding to the SQL statement)” on page 334
- “Example 4 (Removing part of the SQL statement)” on page 340
Using advanced SQL prototyping

“Executing a SELECT statement after you have built it using prototyping” on page 350

Related references
“SQL Prototyping, Execution and Analysis panel” on page 680
“Advanced SELECT Prototyping panel” on page 368
“CANCEL primary command” on page 730
“DESELECT primary command” on page 738
“EXECUTE primary command” on page 744
“INSRPT primary command” on page 757
“NEXTRPT primary command” on page 765
“PREVRPT primary command” on page 768
“SQL primary command” on page 778

Tabbing and color-coding
Advanced SELECT prototyping uses colors to help you:

- Navigate the SELECT statement syntax.
- Identify current and non-current parts of the generated SELECT statement, particularly for fragments that contain repeatable items.

Syntax diagram color-coding
To help you navigate the syntax, advanced SELECT prototyping uses a tabbing and color-coding scheme.

You can tab to those parts of the syntax where you are able to either make a choice or enter a variable. You cannot tab to a required keyword.

Advanced SELECT prototyping uses the following system of colors for the syntax diagram:

RED References to required fragments that have not been completed, or references to optional fragments that have been selected but not completed.

WHITE Required keywords. You cannot select these, or tab to them.

GREEN In general terms, indicates a reference to a completed fragment of the syntax:
- Selected optional keywords (otherwise yellow)
- Required fragments whose subselections have been made to enable them to be resolved (otherwise red)
- Optional fragments whose subselections have been made to enable them to be resolved (otherwise yellow)

YELLOW Optional keywords and fragments that are not currently selected (green when selected and complete; red when selected and incomplete).

Generated SELECT statement color-coding
To help you recognize which part of the generated SELECT statement is current (the part that is accessible to you) and, in the case of repeatable items, which is the current repeatable item, advanced SELECT prototyping uses the following system of colors for the generated SELECT statement:
Using advanced SQL prototyping

**GREEN**
The current part of the SELECT statement and which, therefore, is accessible to you. This includes any data that is part of a current repetition.

**YELLOW**
Part of a repetition that is not current. To make a non-current (YELLOW) current (GREEN), use the PrvRpt function key (F10) or NxtRpt function key (F11) function keys.

**WHITE**
Parts of the SELECT statement that are above the level of the current fragment.

Related references

“Advanced SELECT Prototyping panel” on page 368

Using the advanced SELECT prototyping panels to build a SELECT statement

If you use advanced SELECT prototyping, you place the cursor on the part of the syntax you want and press Enter. To specify a value for a variable, you type the name in a panel. As you proceed to build the SELECT statement, FM/DB2 shows the partly-built statement within the panel you are viewing. You repeat this process until you have built the complete SELECT statement.

**Example 1 (Building a simple SQL statement)**
The following series of panels show each stage of using advanced SELECT prototyping to build an SQL statement to show all rows of the table PARTS.

1. Select option 2 (Advanced) from the SQL Prototyping and Execution panel to display the initial Advanced SELECT Prototyping panel.

2. Move the cursor to the fullselect fragment.

3. Press Enter. The syntax for the full SELECT clause is displayed.

4. Move the cursor to the subselect fragment.
5. Press Enter. The syntax for the subselect clause is displayed.

The panel now shows the partly-built SELECT statement:

```
SELECT ALL
```

This is because:

- The keyword SELECT is a required keyword for this fragment. As you have selected this fragment, FM/DB2 has entered it into the SELECT statement.
- ALL is a default keyword (and, in this case, what you want).

Notice that the `select` and `from` fragments are displayed on your screen in red indicating that the subsections relating to these fragments must be completed before these fragments can be resolved.

6. Move the cursor to the `select` fragment.
7. Press Enter. The syntax for the SELECT clause is displayed.
Using advanced SQL prototyping

8. Move the cursor to the asterisk (*).

The panel now shows the partly-built SELECT statement:

```
SELECT ALL *
```

10. Press the Exit function key (F3). The syntax for the subselect clause is displayed.

Notice that the select fragment has now changed on your screen from red to green indicating that the subsection relating to this fragment has been completed and this fragment can be resolved.

You now need to complete the subsection relating to the FROM fragment.

11. Move the cursor to the FROM fragment.
12. Press Enter. The syntax for the FROM clause is displayed.

The panel now shows the partly-built SELECT statement:

```
SELECT ALL * FROM
```

This is because the keyword FROM is a required keyword.

Again, notice that the `tabref` fragment is displayed on your screen in red, indicating that the subsection relating to this fragment is yet to completed before this fragment can be resolved.

13. Move the cursor to the `tabref` fragment.

14. Press Enter. The syntax for the table specification clause is displayed.

The panel now shows the partly-built SELECT statement:

```
SELECT ALL * FROM
```

15. Move the cursor to the `table` fragment.

16. Press Enter. A panel is displayed in which you can enter a table name.
Using advanced SQL prototyping

17. Type PARTS in the panel.

```
Process Options Utilities Help
----- ----- ----- ---------------
F P E
__  __ _____________ FreeForm Entry ___________

Enter a table name to be inserted in the SQL:
PARTS

Command ===>
F1=Help  F2=Split  F3=Exit  F7=Backward  F8=Forward
F9=Swap  F10=PrvRpt
F11=NxtRpt  F12=Cancel

SELECT ALL * FROM
```

18. Press the Exit function key (F3). The syntax for the table specification clause is displayed.

```
Process Options Utilities Help
----- ----- ----- ---------------
FL/DB2 (DFG2) Advanced SELECT Prototyping

Prototyping: table reference clause Status: statement complete

- table
  - view
    - tablocref
      - correl
    - TABLE
      - (fullselect -> correl
    - tabfunref
    - tablerel
    - joinedtable

SELECT ALL * FROM PARTS
```

Figure 97. Advanced SELECT prototyping: the completed statement (example 1)

The panel shows the completed SELECT statement:

```
SELECT ALL * FROM PARTS
```

Notice that the table fragment has now changed on your screen from red to green, indicating that the subsection relating to this fragment has been
Using advanced SQL prototyping

completed and the fragment can be resolved. Also notice that the status indicator (Status:) shows “statement complete”.

Related tasks
“Executing a SELECT statement after you have built it using prototyping” on page 350

Related references
“SQL Prototyping, Execution and Analysis panel” on page 680
“Advanced SELECT Prototyping panel” on page 368

Changing the SQL statement
At any stage of building an SQL statement, you can:
• Change part of the statement by positioning the cursor on the relevant part of the SELECT statement and pressing Enter. FM/DB2 returns you to that part of the syntax. You can then make the required change.
• Add an extra fragment to the statement by positioning the cursor on the relevant part of the SELECT statement and pressing Enter. FM/DB2 returns you to that part of the syntax. You can then add the required fragment.
• Remove a part of the statement by pressing the Deselect function key (F4).

Example 2 (Changing the SQL statement)
Having built the SQL statement in the previous exercise to show all rows of the table PARTS (see Figure 97 on page 332), suppose you want to change this to eliminate all but one of each set of duplicate rows:
1. Move the cursor under the word ALL in the generated statement.
2. Press Enter. The syntax for the SELECT clause is displayed.

3. Move the cursor to the keyword DISTINCT.
4. Press Enter. DISTINCT replaces ALL in the built SELECT statement.
The panel shows the completed SELECT statement:
SELECT DISTINCT * FROM PARTS

Related tasks
“Example 1 (Building a simple SQL statement)” on page 328
“Executing a SELECT statement after you have built it using prototyping” on page 350

Example 3 (Adding to the SQL statement)
Suppose now that you want to add a condition to the SQL statement in the
previous exercise (see Figure 98) to show only rows of the table PARTS with a part
number greater than 12456:

1. Press the Exit function key (F3) to return to the syntax for the subselect clause.
Using advanced SQL prototyping

2. Move the cursor to the where fragment.
3. Press Enter. The syntax for the WHERE clause is displayed.

```
WHERE
```

Notice that FM/DB2 has added the required keyword WHERE to the SQL statement.

4. Move the cursor to the srchcon fragment.
5. Press Enter. The syntax for the search condition clause is displayed.

```
NOT (srchcon)
```

6. Move the cursor to the first predicate fragment.
7. Press Enter. The syntax for the predicate clause is displayed.
8. Move the cursor to the **basicpred** fragment.
9. Press Enter. The syntax for the basic predicate clause is displayed.

10. Move the cursor to the **expression** fragment (to the left of the list of operators).
11. Press Enter. The syntax for the expression clause is displayed.
12. Move the cursor to the **column** fragment.

13. Press Enter. A pop-up panel is displayed in which you can enter a column name.

14. Type **PARTNO** in the pop-up panel.

15. Press the Exit function key (F3). The column name **PARTNO** is added to the **SELECT** statement.

16. Press the Exit function key (F3). The syntax for the basic predicate clause is displayed.
Using advanced SQL prototyping

17. Move the cursor to the “greater than” (>) symbol.
18. Press Enter. A “greater than” (>) symbol is displayed in the partly-built SELECT statement.
19. Move the cursor to the expression fragment (to the right of the list of operators).
20. Press Enter. The syntax for the expression clause is displayed.

21. Move the cursor to the constant fragment.
22. Press Enter. A pop-up panel is displayed in which you can enter a constant.
Using advanced SQL prototyping

23. Type 12456 in the pop-up panel.

![Image of the pop-up panel]

**SELECT DISTINCT * FROM PARTS WHERE PARTNO > 12456**

24. Press the Exit function key (F3).

The panel shows the completed SELECT statement:

![Image of the completed statement]

**SELECT DISTINCT * FROM PARTS WHERE PARTNO > 12456**

Related tasks

- “Example 2 (Changing the SQL statement)” on page 333
- “Executing a SELECT statement after you have built it using prototyping” on page 350
Using advanced SQL prototyping

Example 4 (Removing part of the SQL statement)
To remove part of a SELECT statement you have built using advanced SELECT prototyping, use the Deselect function key (F4). The effect of the Deselect function key depends on which fragment of the syntax is currently active. For more information, see the description of the DESELECT command.

Suppose that you want to remove the condition you added to the SQL statement in the previous exercise (see Figure 99 on page 339) that restricted the rows shown to those with a part number greater than 12456. This means that you need to remove the WHERE clause from the SELECT statement.

You can remove the WHERE clause by either of these methods:
- Using the Deselect function key (F4):
  1. Place the cursor on the WHERE clause in the generated SELECT statement.
  2. Press Enter to select the WHERE clause. It changes to green, and the syntax for the WHERE clause is redisplayed.
  3. Place the cursor on the command line.
  4. Press the Deselect function key (F4).
- Returning to the prototyping panel that contains the where fragment and “deselecting” it:

  Note: The following steps assume the current panel on your screen is the last panel in Example 3 as shown in Figure 99 on page 339.
  1. Press the Exit function key (F3). FM/DB2 returns you to the syntax for the basic predicate clause.
  2. Press the Exit function key (F3). FM/DB2 returns you to the syntax for the predicate clause.
  3. Press the Exit function key (F3). FM/DB2 returns you to the syntax for the search condition clause.
  4. Press the Exit function key (F3). FM/DB2 returns you to the syntax for the WHERE clause.
  5. Press the Exit function key (F3). FM/DB2 returns you to the syntax for the subselect clause. This panel contains the where fragment.
6. Move the cursor to the \texttt{where} fragment.

7. Press the Deselect function key (F4).

The panel shows the SELECT statement without the WHERE clause:

\begin{verbatim}
SELECT DISTINCT * FROM PARTS
\end{verbatim}

\begin{verbatim}
Command ===>
\end{verbatim}

\begin{verbatim}
F1=Help F2=Split F3=Exit F4=Deselect F5=InsRpt F6=Executed
F7=Backward F8=Forward F9=Swap F10=PrvRpt F11=NxtRpt F12=Cancel
\end{verbatim}

\begin{figure}
\centering
\includegraphics[width=\textwidth]{advanced.sql}
\caption{Advanced: the statement with WHERE clause removed (example 4)}
\end{figure}

\textbf{Related tasks}

“Example 3 (Adding to the SQL statement)” on page 334

“Executing a SELECT statement after you have built it using prototyping” on page 350

\textbf{Handling repeatable items in the syntax}

Some parts of the SELECT statement syntax contain repeatable items where one or more repeatable items are delimited by a repeat separator. The repeat separator for
Using advanced SQL prototyping

some repeat fragments is a comma (,), for some others it is a constant or an operand, and for others it is a space. If you specify more than one repeatable item, and the separator is constant, Advanced SELECT prototyping inserts the correct separator for you; otherwise, you must select the separator like any other element.

Let's take a simple example to show how you can build a fragment of the SELECT clause with repeatable items. Say you want to show certain information (specified in the rest of the SELECT statement) about a company's employees, grouped by their skill type, grade, and the year they started.

1. Repeat steps 1 to 5 of the first example (see “Example 1 (Building a simple SQL statement)” on page 328) so that the syntax for the subselect clause is displayed.

2. Move the cursor to the **groupby** fragment.

3. Press Enter. The syntax for the GROUP BY clause is displayed.
Notice that FM/DB2 shows the required keywords GROUP BY in the partly-built SQL clause on the panel.

You can see that the syntax for the GROUP BY clause consists of:
- The required key words GROUP BY (shown on your screen in white),
- The repeatable item `groupingexpr` (shown on your screen in red), and
- A comma (,) as the repeat separator.

4. Move the cursor to the `groupingexpr` fragment.
5. Press the InsRpt function key (F5). A pop-up panel is displayed in which you can enter a column name.
6. Type SKILLTYPE in the pop-up panel.
Using advanced SQL prototyping

7. Press the Exit function key (F3). The column name SKILLTYPE is added to the GROUP BY clause.

```
GROUP BY SKILLTYPE
```

Notice that, at this stage, FM/DB2 has not added a repeat separator to the statement. If you are only specifying one repeatable item, this is the correct syntax. However, if FM/DB2 detects that you are specifying more than one repeatable item (as in the following steps), it inserts the repeat separator.

8. Again, move the cursor to the `groupingexpr` fragment.
9. Press the InsRpt function key (F5) to display the pop-up panel.
10. Type EMPGRADE in the pop-up panel.
11. Press the Exit function key (F3). The repeat separator and column name EMPGRADE are added to the GROUP BY clause.

```
GROUP BY SKILLTYPE , EMPGRADE
```

```
Using advanced SQL prototyping

12. Repeat steps 8 on page 344 through 11 on page 344 for STARTYEAR.

The panel shows the completed GROUP BY clause:

GROUP BY SKILLTYPE, EMPGRADE, STARTYEAR

Note: If you only require one repeatable item in the fragment, after you move the cursor to the groupingexpr fragment (step 4 on page 343), you can press the InsRpt function key (F5) or Enter.

Related tasks
“Example 1 (Building a simple SQL statement)” on page 328
“Navigating through the repeatable items” on page 345
“Inserting an extra repeatable item” on page 346
“Changing a repeatable item” on page 346
“Deleting a repeatable item” on page 347
“Handling selectable repeat separators” on page 348
“Points to consider when working with repeatable items” on page 349

Navigating through the repeatable items
After you have added a number of repeatable items to a clause, you might need to modify part of the clause by inserting an extra repeatable item, or changing or deleting an existing repeatable item. In any of these cases, you need to navigate your way through the repeatable items so that the appropriate repeatable item is current.

To make a repeatable item current (and therefore change its color from yellow to green), either:

- Move the cursor to one of the elements within the repeat group in the syntax, and use PrvRpt (F10) to scroll backwards or NxtRpt (F11) to scroll forwards through the repeatable items in the generated statement, or
- Move the cursor to the repeatable item in the generated statement, and press Enter.
Using advanced SQL prototyping

**Inserting an extra repeatable item**
Suppose you need to insert an extra repeatable item (department) in the previous GROUP BY clause so that the information is grouped by their skill type, department, level, and the year they started:

1. Move the cursor to the `SKILLTYPE` repeatable item (that is to precede the new repeatable item) in the GROUP BY clause.
2. Press Enter to make `SKILLTYPE` the current repeatable item. Its color changes from yellow to green.
3. Move the cursor to the `groupingexpr` fragment.
4. Press the InsRpt function key (F5). A pop-up panel is displayed in which you can enter a column name.
5. Type `DEPT` in the pop-up panel.
6. Press the Exit function key (F3). The repeat separator and the new repeatable item, `DEPT`, are added to the GROUP BY statement after the `SKILLTYPE` repeatable item.

---

**Changing a repeatable item**
Suppose you want to change the name of the repeatable item, EMPGRADE, in the previous GROUP BY clause to EMPLEVEL:

1. Move the cursor to the `groupingexpr` fragment.
2. Press the NxtRpt function key (F11) to make EMPGRADE the current repeatable item. Its color changes from yellow to green.
3. Press Enter. A pop-up panel is displayed showing the repeatable item EMPGRADE already entered.
4. You can now edit the name of the column in the pop-up panel. Change it to EMPLEVEL.

5. Press the Exit function key (F3). EMPGRADE is replaced by EMPLEVEL in the GROUP BY statement.

Deleting a repeatable item
Suppose you want to remove the repeatable item, STARTYEAR, from the previous GROUP BY clause:
1. Move the cursor to the repeatable item, STARTYEAR
2. Press Enter to make STARTYEAR the current repeatable item. Its color changes from yellow to green.
3. Move the cursor to the groupingexpr fragment.
4. Press the Deselect function key (F4). The repeatable item, STARTYEAR (and its preceding repeat separator), is deleted from the GROUP BY clause.

Handling selectable repeat separators
Some repeatable items in the SELECT statement syntax have a repeat separator that you can select (rather than a constant such as a comma). For example, the expression clause as shown in Figure 101 uses an operator (oper) as a repeat separator:

To show how to handle selectable repeat separators, we'll use an example. To generate the following expression clause:
Using advanced SQL prototyping

\[ \text{AMT1} + \text{AMT2} - \text{AMT3} \]

where the columns \text{AMT1}, \text{AMT2}, and \text{AMT3} are three repeat items, and “+” and “-” are the two repeat separators, perform the following steps:

1. Move the cursor to the \textbf{column} element and press Enter. A pop-up panel is displayed.
2. Type \text{AMT1} and press the InsRpt function key (F5). \text{AMT1} is added to the clause:
   \[ \text{AMT1} \]
3. Move the cursor to the \textbf{oper} repeat separator and press the InsRpt function key (F5). A list of valid repeat separators is displayed.
4. Move the cursor to the “+” repeat separator and press Enter. “+” is added to the clause:
   \[ \text{AMT1} + \]
5. Press the Exit function key (F3). The syntax for the expression clause is redisplayed.
6. Move the cursor to the \textbf{column} element and press Enter. A pop-up panel is displayed.
7. Type \text{AMT2} and press the InsRpt function key (F5). \text{AMT2} is added to the clause:
   \[ \text{AMT1} + \text{AMT2} \]
8. Move the cursor to the \textbf{oper} repeat separator and press the InsRpt function key (F5). A list of valid repeat separators is displayed.
9. Move the cursor to the “-” repeat separator and press Enter. “-” is added to the clause:
   \[ \text{AMT1} + \text{AMT2} - \]
10. Press the Exit function key (F3). The syntax for the expression clause is redisplayed.
11. Move the cursor to the \textbf{column} element and press Enter. A pop-up panel is displayed.
12. Type \text{AMT3} and press the InsRpt function key (F5). \text{AMT3} is added to the clause:
    \[ \text{AMT1} + \text{AMT2} - \text{AMT3} \]

\textbf{Note:} The repeat separator is associated with the repeat item it follows. When the last repeat item is current, you cannot edit the repeat separator.

\textbf{Points to consider when working with repeatable items}

If you are working with repeatable items, keep in mind the following:

- Repeatable item commands are only effective if you move the cursor to one of the elements of the repeatable item.
- For each repeatable item in a fragment (generally none or one), only one “instance” of the item is current at a given time. Generated data for the current instance is displayed in green; generated data for any other instances is displayed in yellow.
- Press the PrvRpt (F10) and NxtRpt (F11) function keys to navigate between the instances.
- The InsRpt function key (F5) inserts a repeat instance after the active repeat instance and makes the new repeat instance active.
- If there are no instances for a repeatable item, you can create the initial instance and make it active by either:
Using advanced SQL prototyping

- Using the InsRpt function key (F5), or
- Moving the cursor to an element of the repeatable item and pressing Enter

- If there is already an instance of a repeatable item, pressing Enter with the cursor on an element of the repeatable item causes that element to be “edited”.

Executing a SELECT statement after you have built it using prototyping

When you have completed building an SQL SELECT statement, using either basic SELECT prototyping or advanced SELECT prototyping, you can run it by:

- Entering the EXECUTE primary command, or
- Pressing the Execute function key (F6)

FM/DB2 displays the results according to the Use edit (instead of browse) to display results option. If this option is selected, FM/DB2 displays the results using FM/DB2 Edit; otherwise, FM/DB2 displays the results using FM/DB2 Browse.

Figure 102 shows a results table using FM/DB2 Browse.

If there is insufficient space on the panel to show all of the columns, use the LEFT primary command to scroll left, or the RIGHT primary commands to scroll right, to view the additional columns.

Related tasks

- FM/DB2 system options (option 0.2)” on page 41

Related references

- “EXECUTE primary command” on page 744
Entering, executing, and explaining SQL statements

If you want to:
- Enter freeform SQL statements,
- Run freeform SQL statements, or
- Obtain information about an SQL statement

select option 3 (Enter) on the SQL Prototyping, Execution and Analysis panel.

When you select this option, FM/DB2 displays the Enter, Execute and Explain SQL Statements panel.

Related tasks
- “Entering SQL statements” on page 352
- “Executing SQL statements” on page 352
- “Explaining SQL” on page 353

Related references
- “SQL Prototyping, Execution and Analysis panel” on page 680
- “Enter, Execute and Explain SQL Statements panel” on page 555

Entering SQL statements

You can enter SQL statements freeform in the SQL statement entry area. If you want to enter more than one SQL statement, you must separate the statements with a semicolon (;). When the SQL statements are processed, the last column of each line is joined directly to the first column of the next line. If a token ends in the last column of a line, ensure that the first column of the next line contains a blank or a delimiter token.

For detailed information about constructing SQL statements, see the DB2 for z/OS SQL Reference.

Editing SQL

You can edit the SQL you have entered by entering the SQL primary command. The SQL you have entered is copied to a temporary data set and an ISPF edit session is started. When you complete the Edit session, the temporary data set is copied back to the entry area on the Enter, Execute and Explain SQL Statements panel.

When you edit the SQL, you can:
- Save it to a data set for later use (using the ISPF edit CREATE command).
- Retrieve SQL you have saved previously (using the ISPF edit COPY command).
- Exploit the power of the ISPF editor to construct your SQL.

Related references
- “Enter, Execute and Explain SQL Statements panel” on page 555
- “SQL primary command” on page 778

Limiting the rows loaded into the FM/DB2 editor

To limit the rows loaded into the FM/DB2 editor when the SQL statement being processed is a SELECT statement, specify the maximum number of rows in the Row count field on the Enter, Execute and Explain SQL Statements panel.

To load all rows for the SELECT statement, specify * or ALL in this field.
Scrolling the entry area

If the statement you are entering extends beyond the visible display area, you can scroll through your data, using the following function keys:

**Up function key (F7)**
- Scroll backward (up)

**Down function key (F8)**
- Scroll forward (down)

You can also scroll using the primary commands UP and DOWN.

To control how far you scroll when you press one of the scrolling function keys, enter a scroll amount in the Scroll field:

**Scroll amount**
- **PAGE**  One page of data at a time
- **HALF**  Half a page of data at a time
- **MAX**  To the top or bottom
- **DATA**  One line less than a page of data at a time
- **CSR**  To the cursor position (if the cursor is within the data, otherwise the default scroll amount, PAGE, applies)

*nnnn* *nnnn* lines at a time.

Related references

- “Enter, Execute and Explain SQL Statements panel” on page 555

Executing SQL statements

To run the SQL you have entered in the statement entry area, press Enter.

If the SQL statement returns a result table, FM/DB2 displays it in an edit or browse panel, depending on the setting of the **Use edit (instead of browse) to display results** option.

If you have entered more than one SQL statement, the statements are run in sequence, interrupted by the display of any result tables. Execution continues until:
- An error occurs,
- The CANCEL command is entered on a result table display, or
- All the statements have been run

You can use the Enter, Execute and Explain SQL Statements panel to specify one or more of the following SQL SELECT statements (without using prototyping) and then run them in sequence:

- **ALTER**
- **CREATE**
- **EXPLAIN**
- **LABEL**
- **REVOKE**
- **SET CURRENT DEGREE**
- **COMMENT ON**
- **COMMIT**
- **DELETE**
- **GRANT**
- **LOCK TABLE**
- **RENAME**
- **ROLLBACK**
- **SELECT**
- **SET CURRENT SQLID**
- **UPDATE**

Related tasks

- “Entering SQL statements” on page 351
- “FM/DB2 system options (option 0.2)” on page 41
Entering, executing, and explaining SQL statements

Related references

“Enter, Execute and Explain SQL Statements panel” on page 555

SQL cost
If any SELECT statements are run, and no errors occur, FM/DB2 displays a message after each execution showing the total SQL “cost” of all the SELECT statements.

SQL errors
If an SQL error occurs while a statement is being run or explained, a panel is displayed containing the SQL error information.

To display the full text of the SQL statement that caused the error, press the SQL function key (F5) or enter SQL on the command line.

When you return from the error panel, FM/DB2 places the cursor on the failing statement or, if a syntax error occurred, places the cursor on the token in the failing statement that DB2 identified as in error. Statements that ran successfully before the failing statement are not rolled back, and no further statements are run or explained.

Explaining SQL
To explain the SQL you have entered in the statement entry area, enter the EXPLAIN primary command. FM/DB2 creates an EXPLAIN statement for the entered SQL, and optionally displays the Plan Table Rows panel showing the rows added by DB2 as a result of explaining.

Note: For detailed information about explaining an SQL statement, see the DB2 for z/OS SQL Reference.

To help identify the results of this explanation, you can enter a number in the Query number entry field. If you leave Query number blank, FM/DB2 generates a query number for you in the format YYYYMMDDss, where ss is a sequence number.

If you have selected the Show results option, FM/DB2 attempts to display the relevant rows that DB2 has inserted in the plan table as a result of explaining the statement. If you have entered more than one SQL statement, the statements are explained in sequence, interrupted by the display of the plan table rows. Execution continues until an error occurs, the CANCEL command is entered on a plan table display, or all the statements have been explained.

You can use the Enter, Execute and Explain SQL Statements panel to explain the following SQL statements:

SELECT
INSERT
UPDATE (searched form only)
DELETE (searched form only)

The following primary commands are available with the Enter, Execute and Explain SQL Statements panel:

• DOWN
• EXECUTE
• EXPLAIN
• SQL
• UP
Related tasks
- “Entering SQL statements” on page 351
- “Executing SQL statements” on page 352
- “Additional considerations for EXPLAIN”

Related references
- “Enter, Execute and Explain SQL Statements panel” on page 555
- “Plan Table Rows panel” on page 615
- “DOWN primary command” on page 738
- “EXECUTE primary command” on page 744
- “EXPLAIN primary command” on page 745
- “SQL primary command” on page 778
- “UP primary command” on page 780

Additional considerations for EXPLAIN
The EXPLAIN primary command constructs and runs an SQL EXPLAIN statement. For the explain statement to run successfully, a plan table (called PLAN_TABLE) must exist with an owner name matching your current SQL authorization ID. DB2 places the results of the EXPLAIN statement in the plan table.

FM/DB2 also places relevant data in a statement table (called DSN_STATEMNT_TABLE) and a function table (called DSN_FUNCTION_TABLE) if they exist.

You can access the statement and function table rows related to a plan table row by using line commands.

Related tasks
- “Explaining SQL” on page 353
- “Managing tables used by SQL explain” on page 355

Related references
- “COL primary command” on page 736
- “PLAN primary command” on page 766
- “HINT primary command” on page 756

Editing and executing SQL statements from a data set
If you want to:
• Optionally initiate an ISPF Edit session on a data set or PDS member containing SQL statements
• Run the SQL statements in a data set or PDS member

select option 4 (Edit) on the SQL Prototyping, Execution and Analysis panel.

Note: You cannot use this option to create a new member.

When you select this option, FM/DB2 displays the Edit and Execute SQL Statements from a Data Set panel.

To use this panel:
1. Specify the name of the data set or PDS member containing the SQL statements you want to edit. Use a combination of the Data set name, Member, and Volume serial entry fields.
**Editing and executing SQL statements from a data set**

2. To limit the rows loaded into the FM/DB2 editor when the SQL statement being processed is a SELECT statement, specify the maximum number of rows in the **Row count** field on the Edit and Execute SQL Statements from a Data Set panel.
   
   To load all rows for the SELECT statement, specify * or ALL in this field.

3. Set the processing options.

4. Press Enter.

   If you selected the **Edit data set** processing option, the contents of the specified data set or PDS member are displayed in an ISPF Edit session. When you have completed editing the SQL statements, press the Exit function key (F3) to run the commands (the **Execute SQL from data set** processing option must be selected).

   If you selected the **Execute SQL from data set** processing option, but not the **Edit data set** processing option, FM/DB2 immediately runs the SQL commands in the specified data set or PDS member

**Note**: If an SQL error occurs, to prevent FM/DB2 executing further SQL statements, use the Cancel function key (F12) to return you to the Edit and Execute SQL Statements from a Data Set panel.

**Related tasks**

- [Specifying a data set and a member name](#) on page 30

**Related references**

- [SQL Prototyping, Execution and Analysis panel](#) on page 680
- [Edit/Execute SQL (Data Set) panel](#) on page 526

---

**Managing tables used by SQL explain**

If you want to select functions to support the use of the SQL EXPLAIN statement, select option 5 (Utility) on the SQL Prototyping, Execution and Analysis panel.

When you select this option, FM/DB2 displays the Explain Utilities panel. This panel lets you create and delete the tables used by the SQL EXPLAIN statement. For additional information about the tables used by the SQL EXPLAIN statement, see the *DB2 for z/OS SQL Reference*.

Use the Explain Utilities panel to specify the type of function you want to perform. For more details about using this panel, refer to the reference sections.

**Related tasks**

- [Entering, executing, and explaining SQL statements](#) on page 351

**Related references**

- [SQL Prototyping, Execution and Analysis panel](#) on page 680
- [Explain Utilities panel](#) on page 556

---

**Exiting SQL prototyping**

To exit SQL prototyping, issue the CANCEL primary command or press the Cancel function key (F12).

**Related references**

- [CANCEL primary command](#) on page 730
Exiting SQL prototyping
Chapter 13. Interfacing with DB2 Interactive

To interface to DB2 Interactive (DB2I), use the DB2I utility function (option 5).

To display the DB2I Primary Option Menu panel, from the Primary Option Menu panel select option 5 (DB2I).

For further information about DB2I, see the DB2 Application Programming and SQL Guide.

Related references
- “Primary Option Menu panel” on page 617
- “DB2I Primary Option Menu panel” on page 521
Interfacing with DB2 Interactive
Part 2. FM/DB2 reference
Chapter 14. DB2 data types

FM/DB2 supports the following IBM-supplied data types:

**Character strings (alphanumeric)**
- A sequence of bytes where the length of the string is the number of bytes in the sequence.
  - **CHAR**
    - Fixed-length character string.
  - **VARCHAR**
    - Varying-length character string.
  - **GRAPHIC**
    - Fixed-length graphic string.
  - **VARGRAPHIC**
    - Varying-length graphic string.
  - **BINARY**
    - Fixed-length binary string.
  - **VARBINARY**
    - Varying-length binary string.

**Numeric**
- Decimal, binary integer, and floating point data types containing a sign and a precision.
  - **DECIMAL** or **NUMERIC**
    - A packed decimal number with an implied decimal point.
  - **DECFLOAT**
    - A decimal floating-point number.
  - **SMALLINT**
    - A binary integer with a precision of 15 bits.
  - **INTEGER**
    - A binary integer with a precision of 31 bits.
  - **BIGINT**
    - A binary integer with a precision of 63 bits.
  - **REAL**
    - A single precision floating-point number (32 bits).
  - **DOUBLE, DOUBLE PRECISION, or FLOAT**
    - A double precision floating-point number (64 bits).

**Row identifier**
- Supports queries that navigate directly to a row in the table.
  - **ROWID**
    - A unique value maintained permanently by DB2.

**Datetime**
- These data types represent a point in time.
  - **DATE**
    - Internally to DB2, the date is represented as a 4 byte field, with each byte containing 2 packed decimal digits stored in the format `yyyyymmdd`.
DB2 data types

Externally, as returned by an SQL SELECT statement, the date is represented by a character field. The format of the string is dependant on the DATE parameter in the DB2 DSNHDECP installation module (the DB2 installation option specified on the installation panel DSNTP4). Table 25 shows valid DB2 date formats. The length of the date field is 10 bytes, except where the DB2 Installation Date format is LOCAL, in which case the length is installation-dependent and in the range 10 to 254 bytes. The date value must be in the range 0001-01-01 to 9999-12-31 (ISO format).

TIME

Internally to DB2 the timestamp is represented as a 7-15 byte field, with each byte containing 2 packed decimal digits. The first 4 bytes represent the date, the next 3 bytes the time, the next 0-6 bytes the microseconds, and when a TIME ZONE is present, 2 bytes containing the time zone.

Externally, as returned by an SQL select statement, the timestamp is represented by a 19-38 byte character field. The timestamp is always returned in the format yyyy-mm-dd.hh.mm.ss.sHH:MM where:

- yyyy An integer in the range 1 to 9999 representing the years.
- mm An integer in the range 1 to 12 representing the months.
- dd An integer in the range 1 to 31 representing the days. The value must be appropriate for the month, including leap years.
- hh An integer in the range 0 to 24 representing the hours.
- mm An integer in the range 0 to 59 representing the minutes.
- ss An integer in the range 0 to 59 representing the seconds.

The remaining parts of the TIMESTAMP (fractional part of the seconds value and TIME ZONE) are optional:

- nnnnnnnnnnnn An integer in the range 0 to 99999999999 representing the fractional part of the seconds. The number of digits can vary from 0-12 inclusive, depending on the precision of the TIMESTAMP.
- sHH A signed integer in the range -12 to +14 inclusive. -12 represents 12 hours behind Greenwich mean time, +14 represents 14 hours ahead.
- MM An integer in the range 00-59, representing the minutes component of the time zone. Internally to DB2, the time is represented as a 3 byte field with each byte containing 2 packed decimal digits stored in the format hh:mm:ss.

Table 25. Date and Time field formats

<table>
<thead>
<tr>
<th>Format Name</th>
<th>Abbreviation</th>
<th>Date format</th>
<th>Time format</th>
</tr>
</thead>
<tbody>
<tr>
<td>International Standard Organization</td>
<td>ISO</td>
<td>yyyy-mm-dd</td>
<td>hh:mm:ss</td>
</tr>
<tr>
<td>IBM USA standard</td>
<td>USA</td>
<td>mm/dd/yyyy</td>
<td>hh:mm AM or PM</td>
</tr>
<tr>
<td>IBM European standard</td>
<td>EUR</td>
<td>dd.mm.yyyy</td>
<td>hh:mm:ss</td>
</tr>
<tr>
<td>Japanese industrial standard</td>
<td>JIS</td>
<td>yyyy-mm-dd</td>
<td>hh:mm:ss</td>
</tr>
<tr>
<td>Installation defined</td>
<td>LOCAL</td>
<td>Installation defined format</td>
<td>Installation defined format</td>
</tr>
</tbody>
</table>

For a full description of DB2 data types, see the DB2 UDB for z/OS SQL Reference.
Large objects (LOBs)

The term large object (LOB) refers to any of the following data types:

- **CLOB**  Character large object
- **DBCLOB**  Double-byte character large object
- **BLOB**  Binary large object

FM/DB2 recognizes columns with a LOB data type but these columns do not participate in FM/DB2 functions. If displayed, the name of the LOB column is shown but the column is displayed as empty.
Chapter 15. FM/DB2 panels and fields

This section of the manual lists most of the FM/DB2 panels. For several of the panels, there is a definition for each field in the panel and, where applicable, the value ranges that are valid for each entry field.

Scrollable fields on FM/DB2 panels

Some fields on FM/DB2 panels are scrollable to allow you to input or view longer values.

The panels shown in this section were produced using DB2 Version 8. In these panels, some fields have adjacent “+” and “-” signs. These signs show that the fields are scrollable.

For further information, see “Scrollable input and display fields for long names” on page 19.

Action bar pull-down menu

The Action bar pull-down menus give you a fast way to move around the product. The panel shown here shows an example of a pull-down menu for the action bar item, Options.

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (Command</td>
<td>1. Print settings</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2. FM/DB2 system options</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3. Job card specifications</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0 Setti</td>
<td>4. Compiler language selection</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 View</td>
<td>5. COBOL compiler specifications</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2 Edit</td>
<td>6. HLA compiler specifications</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3 Utili</td>
<td>7. PL/I compiler specifications</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4 SQL</td>
<td>8. Temporary Data Set Allocations</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5 DB2I</td>
<td>9. Output Data Set Allocations</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6 Comma</td>
<td>10. Trace options</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7 Templ</td>
<td>11. ISPF settings</td>
<td></td>
<td></td>
</tr>
<tr>
<td>X Exit</td>
<td>12. Editor options</td>
<td></td>
<td></td>
</tr>
<tr>
<td>13. Print utility options</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14. Copy utility options</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15. Object list utility options</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>16. Import utility options</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17. Export utility options</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>18. DB2 LOAD utility options</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>19. DB2 Utility LISTDEF options</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20. DB2 Utility OPTIONS options</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>21. DB2 Utility TEMPLATE options</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>22. DB2 Unload utility options</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

© Copyright IBM Corp. 2001, 2013
Action bar pull-down menu

Options

Note: The Options menu is not available for the Set Processing Options panel ("Set Processing Options panel" on page 671).
The Options menu offers the following choice:
1. Print settings
2. FM/DB2 system options
3. Job card specifications
4. Compiler language selection
5. COBOL compiler specifications
6. HLASM compiler specifications
7. PL/I compiler specifications
8. Temporary Data Set Allocations
9. Output Data Set Allocations
10. Trace options
11. ISPF settings
12. Editor options
13. Print utility options
14. Copy utility options
15. Object list utility options
16. Import utility options
17. Export utility options
18. DB2 LOAD utility options
19. DB2 Utility LISTDEF options
20. DB2 Utility OPTIONS options
21. DB2 Utility TEMPLATE options
22. DB2 Unload utility options

Utilities

Note: The Utilities menu is not available for the Set Processing Options panel ("Set Processing Options panel" on page 671).
The Utilities menu offers the following choice:
1. Print
2. Objects
3. Copy
4. Object List
5. DB2 Privileges
6. Import
7. Export
8. Create
9. DB2 Utilities
10. Audit Trail
11. Print Browse

Help
The Help menu offers the following choice:
1. Help for help...
2. Extended help...
3. Keys help...
4. Help index A-M
5. Help index N-Z
6. Tutorial...
7. About...
8. About DB2...
9. News about FM/DB2...
Action bar pull-down menu

Parent panels

Pull-down menus are available from all FM/DB2 panels.

Child panels

None.

Related tasks

- “Checking your FM/DB2 version” on page 9
- “Exiting from FM/DB2” on page 10
- “Selecting an item from the action bar” on page 18

Additional LISTDEF Statements panel

You use the Additional LISTDEF Statements panel to provide additional LISTDEF statements, which are used to supplement the LISTDEF statement provided in the LISTDEF Options panel.

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DGF2)</td>
<td>Additional LISTDEF Statements</td>
<td>Global Settings</td>
<td></td>
</tr>
</tbody>
</table>

No corresponding statement is generated if its field is blank.

LISTDEF

The text that you enter here is included in a LISTDEF statement, if the Use additional statements option on the LISTDEF Options panel (“LISTDEF Options panel” on page 599) is selected.

Parent panels

- “LISTDEF Options panel” on page 599

Child panels

None.
Additional TEMPLATE Statements panel

You use the Additional TEMPLATE Statements panel to provide additional TEMPLATE statements, which are used to supplement the TEMPLATE statement provided in the TEMPLATE Options (1 of 2) panel.

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>Additional TEMPLATE Statements</td>
<td>Global Settings</td>
<td></td>
</tr>
</tbody>
</table>

No corresponding statement is generated if its field is blank.

```
TEMPLATE

------------------------------------------
<table>
<thead>
<tr>
<th>TEMPLATE</th>
</tr>
</thead>
<tbody>
<tr>
<td>------------------------------------------</td>
</tr>
<tr>
<td>------------------------------------------</td>
</tr>
<tr>
<td>------------------------------------------</td>
</tr>
</tbody>
</table>
```

`Command ===>
F1=Help  F2=Split  F3=Exit  F4=CRetriev  F6=Reset  F7=Backward
F8=Forward  F9=Swap  F10=Actions  F12=Cancel`

**TEMPLATE**

The text that you enter here is included in a TEMPLATE statement, if the Use additional statements option on the TEMPLATE Options (1 of 2) panel ("TEMPLATE Options (1 of 2) panel" on page 707) is selected.

Parent panels

- "TEMPLATE Options (1 of 2) panel" on page 707

Child panels

None.

Related tasks

- "RESET primary command" on page 771
- "Setting options for DB2 utilities" on page 278

Advanced SELECT Prototyping panel

The advanced select prototyping function is an aid to learning the syntax of the SQL SELECT statement, and a tool for prototyping complex SQL statements. Fragments of the statement syntax are displayed on the screen in a form based on the "railroad tracks" used to describe the syntax in the DB2 manuals. By
positioning the cursor on the parts of the syntax you wish to explore or use, you can build up a statement that conforms to the syntax requirements. The statement you create is displayed at the bottom of the screen, and can be executed when you have completed it.

Panel and field definitions

Prototyping
Shows the name of the syntax fragment currently displayed.

Status
The current status of the SQL statement you are building. The status indicator, in the form of a text string, applies either to the whole SQL statement (in which case, it uses the word “statement”), or to the clause within the SQL statement that you are currently prototyping (in which case, it uses the word “clause”):

- **statement incomplete**
  Only is displayed on the first Advanced SELECT Prototyping panel, and indicates that the statement generated so far is not ready for execution.

- **clause incomplete**
  Can be displayed on any panel except the first, and indicates that the clause currently being prototyped is not ready for execution (and by implication, the statement as a whole is not yet ready for execution).

- **clause complete**
  Can be displayed on any panel except the first, and indicates that the clause currently being prototyped is now complete. This indirectly implies (because otherwise the status would appear as “statement complete”) that the statement as a whole is not yet ready for execution.

- **statement complete**
  Can be displayed on any panel, and indicates that the statement as
Advanced SELECT Prototyping panel

...a whole is now complete and ready for execution (and implies that the current clause is also complete).

**Note:** Advanced SELECT prototyping is designed to help you build a SELECT statement until it is displayed to be complete. However, the “completed” status of a SELECT clause or statement does not necessarily mean that it is syntactically correct or references valid DB2 data. Only the successful execution of the SELECT statement confirms that its syntax is correct.

**View syntax notes**
Explanatory notes relating to the current syntax fragment, available on some of the advanced SELECT prototyping panels. Such panels show the text “View syntax notes” immediately below the “Prototyping:” field heading.

To view the notes, move the cursor to the “View syntax notes” text and press Enter. FM/DB2 displays a panel containing notes relevant to the current syntax fragment.

**Parent panels**
- “SQL Prototyping, Execution and Analysis panel” on page 680

**Child panels**

<table>
<thead>
<tr>
<th>To display this panel...</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>“Select Statement Browse panel” on page 659</td>
<td>Press Enter</td>
</tr>
</tbody>
</table>

**Related tasks**
- “Using advanced SQL prototyping” on page 325
- “Generated SELECT statement color-coding” on page 327

**Related references**
- “CANCEL primary command” on page 730
- “DESELECT primary command” on page 738
- “EXECUTE primary command” on page 744
- “INSRPT primary command” on page 757
- “NEXTRPT primary command” on page 765
- “PREVRPT primary command” on page 768
- “SQL primary command” on page 778
Alter Database panel

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>Alter Database</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Database name . . FMN2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Buffer pool . . (Optional: BPn,BP8Kn,BP16Kn,BP32Kn)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Index buffer pool . BP0 (Optional: BPn)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Storage group . . SG01 + (Optional, storage group name)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Parent panels
- "Databases panel" on page 500

Child panels
None.

Related tasks
**Alter External Scalar Function panel**

**Panel and field definitions**

<table>
<thead>
<tr>
<th>Process Options Utilities Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2) Alter External Scalar Function</td>
</tr>
</tbody>
</table>

**ALTER SPECIFIC FUNCTION** "SYATES"."FUNCTIO1"  

All parameters are optional

- **External name** . . . **FUNCTIO1** + (default is the function name)
- **Language** . . . . COBOL Deterministic . N (Y/N)
- **NOT NULL Call** . . . (Y/N) External action T (Y/N)
- **Final call** . . . N (Y/N) DBINFO . . . R (Y/N)
- **SQL** . . . . . . . R (C - Contain R - Read M - Mod N - No)
- **Scratch pad** . . . D (scratch pad length or 0)
- **ALLOW PARALLEL** (Y/N)
- **Collid** . . . + (collection or '-' for none)
- **WLM Environment** . . . WLM FOR DFA2 + (WLM env. name or '-' for none)
- **ASUTIME limit** . . . U (CPU Service units or 0 for no limit)
- **Stay resident** . . . N (Y/N)
- **Program type** . . S (M - Main S - Subroutine)
- **External security** . D (O - DB2 AS U - User C - Definer)
- **Run time options** . +

**Child panels**

None.

**Related tasks**

- "Functions panel" on page 584
Alter Index panel

Panel and field definitions

```
Process Options Utilities Help
________________________
FM/DB2 (DFG2) Alter Index

Index owner .... FMUSER +
Index name .... XDEPT1 +

Partition . . . .

Primary quantity . Buffer pool . . . BP0
Secondary quantity Erase rule . . . NO
Free page . . . . Close rule . . . NO
Percent free . . .
Storage group . . +
VSAM catalog . . Piece size(KB) . . 2097152
GB pool cache . . Copy allowed . . NO
Limit value . . . . +

Use the SHOW primary command to display index partition values, and the SAVE line command to copy the values to this panel.

Command =>
F1=Help F2=Split F3=Exit F4=Expand F7=Backward F8=Forward
F9=Swap F10=Left F11=Right F12=Cancel
```

Parent panels
- “Indexes panel” on page 595

Child panels
None.

Related tasks
Alter Stored Procedure panel

Alter Stored Procedure panel
Panel and field definitions

```
Process Options Utilities Help
FM/DB2 (DFG2) Alter Stored Procedure

ALTER SPECIFIC FUNCTION "SYATES"."FUNCTION1"

All parameters are optional

RESULT SETS . . . 0 (maximum number of result sets)
EXTERNAL NAME . . . TSNM9U1J + (default is the function name)
LANGUAGE . . . . . C + DETERMINISTIC . N (Y/N)
PARM STYLE . . . . D (D - DB2SQL G - General "N - w. NULLS)
SQL . . . . . . . M (C - Contain R - Read M - Mod N - No)
DBINFO . . . . . N (Y/N)
COLLID . . . . . . TSNJAR + (collection or '-' for none)
WLM ENV . . . . . WLM FOR DFA2 + (WLM env. name or '-' for none)
ASUTIME LIMIT . . 0 (CPU Service units or 0 for no limit)
STAY RESIDENT . . Y (Y/N)
PROGRAM TYPE . . S (M - Main S - Subroutine)
EXT SECURITY . . U (D - DB2 AS U - User C - Definer)
COMMIT RETURN . . N (Y/N)
RUN TIME OPTS . . POSIX(ON) +

Command ===>
F1=Help F2=Split F3=Exit F4=Expand F7=Backward F8=Forward
F9=Swap F10=Left F11=Right F12=Cancel
```

Parent panels
- "Stored Procedures panel" on page 685

Child panels
None.

Related tasks
Alter Table panel

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
</table>

FM/DB2 (DFG2)  Alter Table

Table Identification:
- Table owner ..... KPS8810 LONG
- Table name ..... VACT

Processing Options:
Select type of ALTER
- 1. AUDIT
- 2. ADD column
- 3. VALIDPROC
- 4. ADD PRIMARY KEY
- 5. ADD FOREIGN KEY constraint
- 6. DROP PRIMARY KEY
- 7. DROP FOREIGN KEY constraint
- 8. DATA CAPTURE
- 9. ADD CHECK constraint
- 10. DROP CHECK constraint
- 11. ADD RESTRICT ON DROP
- 12. DROP RESTRICT ON DROP

Command ==>
F1=Help  F2=Split  F3=Exit  F4=Expand  F7=Backward  F8=Forward
F9=Swap  F10=Left  F11=Right  F12=Cancel

Parent panels
- “Tables, Views and Aliases panel” on page 704
- “Table Check Constraints panel” on page 691

Child panels

<table>
<thead>
<tr>
<th>To display this panel...</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>“Alter Table - AUDIT panel” on page 380</td>
<td>Processing option 1</td>
</tr>
<tr>
<td>“Alter Table - ADD Column panel” on page 377</td>
<td>Processing option 2</td>
</tr>
<tr>
<td>“Alter Table - VALIDPROC panel” on page 386</td>
<td>Processing option 3</td>
</tr>
<tr>
<td>“Alter Table - ADD PRIMARY KEY panel” on page 379</td>
<td>Processing option 4</td>
</tr>
<tr>
<td>“Alter Table - ADD FOREIGN KEY Constraint panel” on page 383</td>
<td>Processing option 5</td>
</tr>
<tr>
<td>“Alter Table - DROP PRIMARY KEY panel” on page 384</td>
<td>Processing option 6</td>
</tr>
<tr>
<td>“Alter Table - DROP FOREIGN KEY Constraint panel” on page 383</td>
<td>Processing option 7</td>
</tr>
<tr>
<td>“Alter Table - DATA CAPTURE panel” on page 381</td>
<td>Processing option 8</td>
</tr>
<tr>
<td>“Alter Table - ADD CHECK Constraint panel” on page 376</td>
<td>Processing option 9</td>
</tr>
<tr>
<td>“Alter Table - DROP CHECK Constraint panel” on page 382</td>
<td>Processing option 10</td>
</tr>
<tr>
<td>“Alter Table - RESTRICT ON DROP panel” on page 385</td>
<td>Processing option 11 or 12</td>
</tr>
</tbody>
</table>

Related tasks
Alter Table - ADD CHECK Constraint panel

Panel and field definitions

Table Identification:
- Owner: KPS8810_LONG +
- Name: VACT +

Constraint:
- Name: +
- Check condition: +

Command: F1=Help F2=Split F3=Exit F4=Expand F7=Backward F8=Forward
F9=Swap F10=Left F11=Right F12=Cancel

Parent panels
- "Alter Table panel" on page 375

Child panels
- None.

Related tasks
Alter Table - ADD Column panel

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>Alter Table - ADD Column</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table Identification:
- Owner: KPS8810_LONG +
- Name: VALT

Column Details:
- Name: 
- Type: + (for example CHAR, DECIMAL, INTEGER)
- Length: 
- Precision: (optional, valid for FLOAT and DECIMAL only)
- Scale: (optional, valid for DECIMAL only)
- NOT NULL: (Yes or No)
- FOR ? DATA: - (type: B - BIT, S - SBCS, D - DBCS, blank - none)
- WITH DEFAULT: - (Yes, No or enter value below)
- Default value: +

FIELDPROC Details:
- Name: (optional)
- Parm: +

Parent panels
- "Alter Table panel" on page 375

Child panels
None.

Related tasks
Alter Table - ADD FOREIGN KEY Constraint panel

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>Alter Table - ADD FOREIGN KEY Constraint</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table Identification:
- Owner: KPS8810 LONG +
- Name: VACT +

Constraint:
- Name: VACT +

Foreign Key:
- Column name 1...
- 2...
- 3...
- 4...
- 5...
- 6...
- 7...
- 8...

REFERENCES Table Identification:
- Owner: SYSIBM
- Name: ...

ON DELETE:
- Select type of delete rule
  1. RESTRICT
  2. CASCADE
  3. SET NULL
  4. NO ACTION

Command: F1=Help  F2=Split  F3=Exit  F4=Expand  F7=Backward  F8=Forward  F9=Swap  F10=Left  F11=Right  F12=Cancel

Parent panels
- “Alter Table panel” on page 375

Child panels
- None.

Related tasks
### Alter Table - ADD PRIMARY KEY panel

#### Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (QFG2)</td>
<td>Alter Table - ADD PRIMARY KEY</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Table Identification:**
- Owner: KPS8810_LONG +
- Name: VACT +

**Primary Key:**
- Column name 1
- Column name 2
- Column name 3
- Column name 4
- Column name 5
- Column name 6
- Column name 7
- Column name 8

**Command ===>
F1=Help F2=Split F3=Exit F4=Expand F7=Backward F8=Forward
F9=Swap F10=Left F11=Right F12=Cancel

#### Parent panels
- “Alter Table panel” on page 375

#### Child panels
- None.

#### Related tasks
Alter Table - AUDIT panel

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>Alter Table - AUDIT</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table Identification:
- Owner ........ KPS8810_LONG
- Name ........ VACT

Auditing Attribute:
Select type of AUDIT
- 1. NONE
- 2. CHANGES
- 3. ALL

Command ===>
F1=Help  F2=Split  F3=Exit  F4=Expand  F7=Backward  F8=Forward
F9=Swap  F10=Left  F11=Right  F12=Cancel

Parent panels
- “Alter Table panel” on page 375

Child panels
None.

Related tasks
Alter Table - DATA CAPTURE panel

Panel and field definitions

```
Panel and field definitions

Table Identification:
Owner .......... KPS8810_LONG +
Name .......... VACT +

Additional Logging Attribute:
Select type of DATA CAPTURE
1. NONE
2. CHANGES

Command ===>
F1=Help      F2=Split      F3=Exit      F4=Expand      F7=Backward      F8=Forward
F9=Swap      F10=Left      F11=Right    F12=Cancel
```

Parent panels

- "Alter Table panel" on page 375

Child panels

None.

Related tasks
### Alter Table - DROP CHECK Constraint panel

#### Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>Alter Table - DROP CHECK Constraint</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Table Identification:**
- **Owner:** KPS8810_LONG
- **Name:** VACT

**CHECK Constraint:**
- **Name:**

**Command ===>
F1=Help  F2=Split  F3=Exit  F4=Expand  F7=Backward  F8=Forward  F9=Swap  F10=Left  F11=Right  F12=Cancel

#### Parent panels
- [“Alter Table panel” on page 375](#)
- [“Table Check Constraints panel” on page 691](#)

#### Child panels
None.

#### Related tasks
Alter Table - DROP FOREIGN KEY Constraint panel

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (QFG2)</td>
<td>Alter Table - DROP FOREIGN KEY Constraint</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table Identification:
- **Owner** ........ KPS8810 LONG +
- **Name** ........ VACT +

FOREIGN KEY Constraint:
- **Name** ........ +

Command ===> F1=Help F2=Split F3=Exit F4=Expand F7=Backward F8=Forward F9=Swap F10=Left F11=Right F12=Cancel

Parent panels
- “Alter Table panel” on page 375

Child panels
None.

Related tasks
Alter Table - DROP PRIMARY KEY panel

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>Alter Table - DROP PRIMARY KEY</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table Identification:
- Owner: KPS8810 LONG
- Name: VACT

DROP PRIMARY KEY

Press ENTER to confirm.

Command ===>

F1=Help  F2=Split  F3=Exit  F4=Expand  F7=Backward  F8=Forward
F9=Swap  F10=Left  F11=Right  F12=Cancel

Parent panels
- "Alter Table panel" on page 375

Child panels
- None.

Related tasks
Alter Table - RESTRICT ON DROP panel

Panel and field definitions

![Panel and field definitions](image)

Parent panels
- “Alter Table panel” on page 375

Child panels
None.

Related tasks
Alter Table - VALIDPROC panel

Panel and field definitions

Table Identification:

Owner ........ KPS8810_LONG +
Name .......... VACT +

Validation Procedure:
Select type of VALIDPROC
1. NULL
2. Program name . .

Parent panels
- "Alter Table panel" on page 375

Child panels
None.

Related tasks
### Alter Table Space panel

#### Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**FM/DB2 (DFG2) Alter Table Space**

- **Database name**: FMN2
- **Table space name**: SYSCOPY
- **Partition**: __________
- **Primary quantity**: __________
- **Secondary quantity**: __________
- **Free page**: __________
- **Percent free**: __________
- **Storage group**: __________
- **VSAM catalog**: __________
- **Compress**: __________
- **GBP Cache**: __________
- **Track mods**: __________
- **Buffer pool**: BP0
- **Close rule**: NO
- **Erase rule**: NO
- **Lock size**: ANY
- **Lock max**: __________
- **Lock part**: __________
- **Max rows**: 255
- **Log**: __________

Use the SHOW primary command to display table space partition values, and the SAVE line command to copy the values to this panel.

**Command ===>
F1=Help F2=Split F3=Exit F4=Expand F7=Backward F8=Forward F9=Swap F10=Left F11=Right F12=Cancel

---

**Parent panels**
- “Table Spaces panel” on page 698

**Child panels**
None.

**Related tasks**

**Application Packages panel**

You use the Application Packages panel to list packages in the DB2 catalog.

**Panel and field definitions**

The columns that are displayed include the SEL field and columns of SYSIBM.SYSPACKAGE.
Note:
1. The system option, **Show all catalog table columns**, affects which columns FM/DB2 displays when you use the Object List utility.
2. You can also customize which columns FM/DB2 displays when you use the Object List utility. For details, see the File Manager Customization Guide.

To view other columns on this panel, press the Right function key (F11).

For a complete list of columns, refer to “DB2 catalog tables” in the DB2 for z/OS SQL Reference relevant to your version of DB2.

SEL  Line command area.

For a list of the line commands you can use on this panel, see Table 22 on page 250.

Parent panels
- “Object List Utility panel” on page 610
- “Tables, Views and Aliases panel” on page 704
- “Indexes panel” on page 595
- “Application Plans panel” on page 389
- “Collections panel” on page 396
- “Stored Procedures panel” on page 685
- “Triggers panel” on page 712
- “Plan Table Rows panel” on page 615
- “Statement Table Rows panel” on page 681

Child panels

<table>
<thead>
<tr>
<th>To display this panel...</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>“Bind Package panel” on page 396</td>
<td>Line command BI or CBI</td>
</tr>
<tr>
<td>“Package Dependencies panel” on page 614</td>
<td>Line command DEP</td>
</tr>
</tbody>
</table>
Application Packages panel

To display this panel... Use/do this

“Free Package panel” on page 583 Line command FRE
“Grant privileges panels” on page 587 Line command G
“Details panels” on page 522 Line command I
“Plan Table Rows panel” on page 615 Line command LPT
“Privileges panels” on page 624 Line command P
“Package List panel” on page 614 Line command PKL
“Application Plans panel” Line command PL
“Revoke privileges panels” on page 647 Line command R
“Rebind Package panel” on page 627 Line command RBI
“Display Row panel” on page 523 Line command ROW
“Table Spaces panel” on page 698 Line command S
(ISPF panel) Line command SQL
“Tables, Views and Aliases panel” on page 704 Line command TB
“Tables, Views and Aliases panel” on page 704 Line command V
“Application Packages panel” on page 387 Line command VER
“Indexes panel” on page 595 Line command X
“Sort Fields panel” on page 679 Primary command SORT

Related tasks
• “Working with object list panels” on page 244
• “Using the line command area (Cmd)” on page 250

Related references
• “SORT primary command” on page 777

Application Plans panel

You use the Application Plans panel to list application plans in the DB2 catalog.

Panel and field definitions

The columns that are displayed include the SEL field and columns of SYSIBM.SYSPLAN.
Note:

1. The system option, **Show all catalog table columns**, affects which columns FM/DB2 displays when you use the Object List utility.

2. You can also customize which columns FM/DB2 displays when you use the Object List utility. For details, see the [File Manager Customization Guide](#).

To view other columns on this panel, press the Right function key (F11).

For a complete list of columns, refer to “DB2 catalog tables” in the [DB2 for z/OS SQL Reference](#) relevant to your version of DB2.

**SEL**  
Line command area.

For a list of the line commands you can use on this panel, see [Table 22 on page 250](#).

**Parent panels**

- “Object List Utility panel” on page 610
- “Application Packages panel” on page 387
- “Collections panel” on page 396
- “Database Request Modules panel” on page 498

**Child panels**

<table>
<thead>
<tr>
<th>To display this panel...</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>“Bind Application Plan panel” on page 396</td>
<td>Line command BI</td>
</tr>
<tr>
<td>“Free Application Plan panel” on page 583</td>
<td>Line command FRE</td>
</tr>
<tr>
<td>“Grant privileges panels” on page 587</td>
<td>Line command G</td>
</tr>
<tr>
<td>“Details panels” on page 522</td>
<td>Line command I</td>
</tr>
<tr>
<td>“Plan Table Rows panel” on page 615</td>
<td>Line command LPT</td>
</tr>
<tr>
<td>“Database Request Modules panel” on page 498</td>
<td>Line command M</td>
</tr>
<tr>
<td>“Privileges panels” on page 624</td>
<td>Line command P</td>
</tr>
</tbody>
</table>
Basic SELECT Prototyping panel

You use the Basic SELECT Prototyping (primary) panel to specify up to 15 tables to be used to construct a FROM clause as the starting point for prototyping a SELECT statement.

Panel and field definitions
Basic SELECT Prototyping panel

Owner
This field is optional. If specified, it is used to qualify the name field on the same line. If omitted, the current SQL ID is used.

Name
This field is required. It specifies the name of the table to be processed.

Location
This field is optional. If specified, it is used to qualify all the tables entered in the Owner and Name fields. If omitted, the current location is be used.

Database
This field is optional. If specified, all the tables entered in the Owner and Name fields must exist in the given database. If omitted, the tables can exist in any database at the specified or defaulted location.

Table space
This field is optional. If specified, all the tables entered in the Owner and Name fields must exist in the given table space. If omitted, the tables can exist in any table space at the specified or defaulted location, subject to any restriction imposed by the database specification.

After you have specified the table names on the primary Basic SELECT Prototyping panel, press Enter to display the secondary Basic SELECT Prototyping panel.

Row count
Enter a number in this field to control:
Basic SELECT Prototyping panel

- The number of rows returned in the result set when the SQL statement is executed. Specifying 0 results in all qualifying rows being returned. Specifying n (when n>0) limits the number of rows returned to the lesser of: n rows, or the number of rows in the result table.
- Whether or not an "OPTIMIZE FOR n ROWS" clause is added to the SQL statement prior to execution. Specifying n (when n>0) results in the addition of "OPTIMIZE FOR n ROWS" to the SQL statement prior to execution. Specifying a 0 value does not add "OPTIMIZE FOR n ROWS" to the SQL statement. Specifying a non-zero value may improve DB2’s use of resources when executing the SQL statement.

S  A selection column in which you can specify one of the following values for the column on that line:

S  Select. Adds the column name to the column list in the SELECT clause.

A  Select ascending. Adds the column name to the column list in the SELECT clause, and adds the column name with the keyword ASC to the column list in the ORDER BY clause.

D  Select descending. Adds the column name to the column list in the SELECT clause, and adds the column name with the keyword DESC to the column list in the ORDER BY clause.

LOp  Logical operator to join a new predicate to previous predicates. You can specify one of the following values for the column on that line:

   Logical operator to join a new predicate to previous predicates. You can specify one of the following values for the column on that line:

   AND (the default operator)
   OR
   NOT (equivalent to “and not”)
   NOR (equivalent to “or not”)

(  Inserts an opening parenthesis at the start of the predicate added to the WHERE clause. Use an opening parenthesis in conjunction with a closing parenthesis to ensure the predicate for this column is evaluated before other (non-parenthesized) predicates.

Tab  The correlation name of the table containing the column on this line. This is an information-only field.

Column Name  The name of the column as held in the DB2 catalog. This is an information-only field.

Data Type(length)  The DB2 data type of the column and its length, or numeric precision and scale. This is an information-only field.

Op  Expression operator. You can specify one of the following values for the column on that line:

   =  Equal. This is the default operator.
   <>  Not equal.
   >  Greater than.
   <  Less than.
   >=  Greater than or equal to.
   <=  Less than or equal to.
   IN  In set. If the value in the column is in the specified set of values
(specified in Value field), the result of the expression is true. The IN expression operator generates the predicate IN set in the WHERE clause.

NI Not in set. If the value in the column is not in the specified set of values (specified in Value field), the result of the expression is true. The NI expression operator generates the predicate NOT IN set in the WHERE clause.

LI Like pattern. If the value in the column matches the pattern (specified in Value field), the result of the expression is true. The LI expression operator generates the predicate LIKE pattern in the WHERE clause.

NL Not like pattern. If the value in the column does not match the pattern (specified in Value field), the result of the expression is true. The NL expression operator generates the predicate NOT LIKE pattern in the WHERE clause.

Value The value to be used with the expression operator (specified in the Op field).

The format of the data you enter depends on the expression operator:

- For expression operators =, <>, >, <, >=, and <=:
  - For columns with an alphanumeric data type, specify a character string, optionally enclosed in quotation marks. If you do not enclose the string in quotes, FM/DB2 automatically adds the quotation marks around the string when it adds the generated predicate to the WHEN clause.
    Examples:
    
    SMITH
    'SMITH'
    SALES REP
    'SALES REP'
  - For columns with a numeric data type, specify a numeric value.
    Examples:
    
    500
    33.75
    0
  - You can also specify a column name as a value. The column name must be qualified with its correlation name.
    Examples:
    
    #1.WORKDEPT
    #2.BONUS

Note: To enter a string that begins with a “#”, you must enclose the string in quotation marks.

- For expression operators IN and NI:
  - For columns with an alphanumeric data type, specify a list of character strings. Each character string must be enclosed in quotation marks and separated by a comma.
    Examples:
    
    'SMITH','JONES','BROWN'
    'ANALYST', 'DESIGNER', 'OPERATOR', 'MANAGER'
Basic SELECT Prototyping panel

- For columns with a numeric data type, specify a list of numeric values. Each value must be separated by a comma and, optionally, one or more spaces.
  Examples:
  101,102,103
  45.5, 50.0, 65.5
- You can also specify a list of column names. Each column name must be qualified with its correlation name and be separated by a comma.
  Examples:
  #1.ACCT,#1PROD,#2MAINT
  #1.BONUS, #2.BONUS
- For the expression operator LI:
  - Specify a pattern (as a character string), optionally enclosed in quotation marks.

  Note: You can only specify a pattern for columns with an alphanumeric data type.
  If you do not enclose the pattern in quotation marks, FM/DB2 automatically adds the quotation marks around the pattern when it adds the generated predicate to the WHEN clause.
  The pattern can contain an underscore (_) to represent any single character, or a percent sign (%) to represent a string of zero or more characters.
  All of the following examples find the string “SMITH”:  
    'SMITH'
    $MI$
    'SM%'
    $MI$
  - Specify a column name as a value. The column name must be qualified with its correlation name.
  Examples:
  #1.SEX
  #1.WORKDEPT

) Inserts a closing parenthesis at the end of the predicate added to the WHERE clause. Use in conjunction with an opening parenthesis to ensure the predicate for this column is evaluated before other (non-parenthesized) predicates.

Parent panels
- “Tables, Views and Aliases panel” on page 704
- “SQL Prototyping, Execution and Analysis panel” on page 680

Child panels
To display this panel... Use/do this
“Select Statement Browse panel” on page 659 Press Enter

Related tasks
- “Using basic SQL prototyping” on page 317
Related references

- “ALL primary command” on page 727
- “CANCEL primary command” on page 730
- “EXECUTE primary command” on page 744
- “RESET primary command” on page 771
- “SQL primary command” on page 778
- “UNDO primary command” on page 780

Bind Application Plan panel

FM/DB2 displays the Bind Application Plan panel if you use the DR line command against an item listed on the “Application Plans panel” on page 389.

Bind Package panel

FM/DB2 displays the Bind Package panel if you use either the BI or CBI line command against an item listed on the “Application Packages panel” on page 387.

Children of Table panel

FM/DB2 displays the Children of Table panel if you use the CHR line command against an item listed on the “Tables, Views and Aliases panel” on page 704.

Collections panel

You use the Collections panel list to list collections in the DB2 catalog.

Panel and field definitions

The columns that are displayed include the SEL field and columns of SYSIBM.SYSPACKAGES.
Note:
1. The system option, **Show all catalog table columns**, affects which columns FM/DB2 displays when you use the Object List utility.
2. You can also customize which columns FM/DB2 displays when you use the Object List utility. For details, see the [File Manager Customization Guide](#).

To view other columns on this panel, press the Right function key (F11).

For a complete list of columns, refer to “DB2 catalog tables” in the *DB2 for z/OS SQL Reference* relevant to your version of DB2.

**SEL** Line command area.

For a list of the line commands you can use on this panel, see [Table 22 on page 250](#).

**Parent panels**
- [“Object List Utility panel” on page 610](#)

**Child panels**

<table>
<thead>
<tr>
<th>To display this panel...</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>“Grant privileges panels” on page 587</td>
<td>Line command G</td>
</tr>
<tr>
<td>“Privileges panels” on page 624</td>
<td>Line command P</td>
</tr>
<tr>
<td>“Application Packages panel” on page 387</td>
<td>Line command PKG</td>
</tr>
<tr>
<td>“Package List panel” on page 614</td>
<td>Line command PKL</td>
</tr>
<tr>
<td>“Application Plans panel” on page 389</td>
<td>Line command PL</td>
</tr>
<tr>
<td>“Revoke privileges panels” on page 647</td>
<td>Line command R</td>
</tr>
<tr>
<td>“Display Row panel” on page 523</td>
<td>Line command ROW</td>
</tr>
<tr>
<td>(ISPF panel)</td>
<td>Line command SQL</td>
</tr>
<tr>
<td>“Sort Fields panel” on page 679</td>
<td>Primary command SORT</td>
</tr>
</tbody>
</table>

**Related tasks**
- [“Working with object list panels” on page 244](#)
- [“Using the line command area (Cmd)” on page 250](#)

**Related references**
- [“SORT primary command” on page 777](#)

### Column Attributes panel (alphanumeric)

The Column Attributes (alphanumeric) panel describes the formatting and data create attributes for an alphanumeric column.
Panel and field definitions

**Column name**
The name of the column.

**Type**
The DB2 data type of the column.

**Heading**
The alternative column heading.

**Output width**
The number of character positions used by view, edit, browse, and print functions to show the column in TABL display or print format.

**Filler**
The value placed in each byte of the column before any other operation in the construction of the column. The default value is a blank character.

- **char** This character, such as 0, is written in each byte.
- **X'cc'** This hexadecimal value, such as X'FF', is written in each byte.

**Action**
This column sets how the contents of the column are varied for each row. The default action is FX.

- **FX** Contents of the column remain fixed.
- **RA** Contents of the column are generated from characters selected at random from the specified pattern. A different set of characters is selected for each row.
- **RO** Contents of the column rolled. The specified pattern is shifted left one byte for each row until the first nonblank character of the pattern is in the first byte of the column. Then the pattern is shifted right one byte for each output row until it returns to its original position in the column. RO can only be used with a user-supplied pattern, not with an IBM-supplied pattern (see following description for Pattern).
Column Attributes panel (alphanumeric)

RP  Contents of the column are rippled. The specified pattern is shifted left one byte for each row and the truncated character is appended to the right end of the pattern.

SL  Contents of the column are shifted left for each row. The specified pattern is shifted left one character and filled to the right with a space. When all characters in the pattern have been shifted out, the original pattern is restored and the process is repeated.

SR  Contents of the column are shifted right for each row. The specified pattern is shifted right one character and filled to the left with a space. When all characters in the pattern have been shifted out, the original pattern is restored and the process is repeated.

TL  Contents of the column are truncated on the left for each row. The specified pattern is left-truncated (the leftmost character replaced with a space) one character at a time until all characters in the pattern have been truncated. Then the original pattern is restored and the process is repeated.

TR  Contents of the column are truncated on the right for each row. The specified pattern is right-truncated (the rightmost character replaced with a space) one character at a time until all characters in the pattern have been truncated. Then the original pattern is restored and the process is repeated.

WV  Contents of the column are waved. The specified pattern is shifted left one byte for each row until the first nonblank character of the pattern is in the first byte of the column. Then the original pattern is restored and the process repeated. WV can only be used with a user-supplied pattern, not with an IBM-supplied pattern (see following description for Pattern).

Start character
Starting character used when you specify an IBM-supplied pattern (AL, AN, or CO). The specified character must be one of the characters in the IBM-supplied pattern. The default start character is the first character in the IBM-supplied pattern.

Pattern
The pattern used when generating data for this column. You can specify either an IBM-supplied pattern indicator or a quoted string containing a user-supplied pattern. The IBM-supplied patterns you can specify are:

AN  Alphanumeric. Characters A–Z, 0–9, inclusive.
CO  Collating sequence. All printable characters in the range X'40'–X'F9'.

If you provide a user-supplied pattern that is longer than the column, for actions other than RA and RP, the pattern is first truncated on the right to fit the receiving column before the specified action is performed. For the RA action, characters are randomly selected from the entire user-supplied pattern. For the RP action, the entire pattern is rippled for each row before it is truncated to fit the receiving column.

If you provide a user-supplied pattern that is shorter than the column, you can specify that you want the pattern repeated to fill the column.
Repeat user pattern
Specify YES if you have provided a user-supplied pattern that is shorter than the receiving column. This option has no effect when the RA action is specified. The default value is NO.

**NO** Fill character used to pad the receiving column.

**YES** Supplied pattern repeated as many times as necessary to fill the receiving column.

**Note:** IBM-supplied patterns are always repeated in a column, as necessary.

Scrambling Options
These options control column scrambling used during copy, import, or export functions to produce test data.

**Scramble Type**
Specify one of the following values:

- **Blank** No scrambling is performed. Value or range specifications are saved but ignored for the associated function.

- **1 (Random)**
  Performs random scrambling. The same input value produces different output values on subsequent invocations.

- **2 (Repeatable)**
  Performs repeatable scrambling. The same input value produces the same output value on subsequent invocations.

- **3 (Translate)**
  Performs translation. The value data set is searched to find a matching input value. If a match is found, then the output value is taken from the output column of the matching record.

- **4 (Exit)**
  Invokes a scrambling user exit. FM/DB2 displays a panel where you can specify the user exit name and user exit parameters and options.

Parent panels
- "Column Selection/Edit panel" on page 407

Child panels
None.

Related tasks
- "Editing the template" on page 192
- "Changing column attributes" on page 73
- "Changing the attributes for an alphanumeric column" on page 74

Column Attributes panel (DATE)
The Column Attributes (DATE) panel describes the formatting and data create attributes for a DATE column.
Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>Column Attributes</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Fixed Attributes:
- **Column name**: BIRTHDATE
- **Type**: DATE
- **Format**: dd.mm.yyyy

Use Attributes:
- **Heading**
- **Output width**

Create Attributes:
- **Date value**
- **Increment**: 0

(Use = for date now, * for runtime date)

Column name
The name of the column.

Type
The DB2 data type of the column.

Format
The format in which the date is displayed or printed.

Heading
The alternative column heading.

Output width
The number of character positions used by view, edit, and print functions to show the column in TABL display or print format.

Date value
The initial value a column is to contain. Valid values are:

- **start date**
  A date in a valid DB2 or local format.
- **(asterisk)**
  The date the data is created. Not valid if the DB2 installation date format is LOCAL.
- **=**
  The current date. Not valid if the DB2 installation date format is LOCAL.

Increment
A integer that represents the number of days to add to the date each cycle. The default increment value is 0.

Parent panels
- "Column Selection/Edit panel" on page 407
The Column Attributes (numeric) panel describes the formatting and data create attributes for a numeric column.

Panel and field definitions

Child panels
None.

Related tasks
- “Editing the template” on page 192
- “Changing column attributes” on page 73
- Table 25 on page 362

Column name
The name of the column.

Type
The DB2 data type of the column.

Heading
The alternative column heading.

Output width
The number of character positions used by view, edit, and print functions to show the column in TABL display or print format.

Leading zeros
Specifies whether leading zeros are to be shown when this field is displayed or printed in SNGL or TABL display or print format. Specify YES if you want leading zeros shown. Specify NO if you want leading zeros replaced with blanks.
Column Attributes panel (numeric)

Start value
The initial value a column is to contain, before being adjusted by the specified increment. The value must be a number that the column can hold without truncation or loss of data. The value is converted to the appropriate numeric data type if required. For a column containing decimal places, the value specified cannot contain more decimal places than specified by the column definition. If you specify a negative number, the sign is honored, even if the column is unsigned. If the column is a floating-point column, the start value can be specified as a floating-point number consisting of a mantissa and an exponent (such as -1.14579E01). The mantissa consists of an optional sign (+ or -) followed by 1 to 16 digits. The mantissa can also contain a decimal point. The exponent consists of the letter E, an optional sign (+ or 0), and 1 or 2 digits.

The default start value is 0.

End value
The maximum value (if the increment is a positive number) or minimum value (if the increment is a negative number) a column is to contain. The value must be a number that the column can hold without truncation or loss of data. The value is converted to the appropriate numeric data type if required. For a column containing decimal places, the value specified cannot contain more decimal places than specified by the column definition. If you specify a negative number, the sign is honored, even if the column is unsigned. If the column is a floating-point column, the end value can be specified as a floating-point number consisting of a mantissa and an exponent (such as -1.14579E01). The mantissa consists of an optional sign (+ or -) followed by 1 to 16 digits. The mantissa can also contain a decimal point. The exponent consists of the letter E, an optional sign (+ or 0), and 1 or 2 digits.

The default end value is the largest positive or negative number the column can contain.

Increment
The positive or negative number by which the value in the column is adjusted for each row (or cycle of rows). The increment value must be a number that the column can hold without truncation or loss of data. The increment value is converted to the appropriate numeric data type if required. For a column containing decimal places, the value specified cannot contain more decimal places than specified by the column definition. If the column is a floating-point column, the increment can be specified as a floating-point number consisting of a mantissa and an exponent (such as -1.14579E01). The mantissa consists of an optional sign (+ or -) followed by 1 to 16 digits. The mantissa can also contain a decimal point. The exponent consists of the letter E, an optional sign (+ or 0), and 1 or 2 digits.

For the first row (or cycle of rows), the column is set to the start value you specify. For each subsequent row (or cycle of rows), the increment is added to the value in the column. This process continues while the calculated value in the column does not exceed the specified end value. That is, while the calculated value is less than or equal to the end value if the increment is a positive number, or greater than or equal to the end value if the increment is a negative number. When the end value is exceeded, the value in the column remains unchanged for all subsequent rows.

If the increment value is zero, the value in the column is always set to the start value.
Column Attributes panel (numeric)

The default increment is 0.

**Cycle**  The number of output rows that are generated before the increment value is applied to the column value. For example, if you specify a column start value of 100, an increment value of 10, and a cycle value of 3, the column in the first 3 rows contains 100, 110 in next 3 rows, 120 in the next 3 rows, and so on.

If cycle is zero, the value in the column is always set to the start value.

The default cycle is 1.

**Scrambling Options**

These options control column scrambling used during copy, import, or export functions to produce test data.

**Scramble Type**

Specify one of the following values:

**Blank**  No scrambling is performed. Value or range specifications are saved but ignored for the associated function.

**1 (Random)**  Performs random scrambling. The same input value produces different output values on subsequent invocations.

**2 (Repeatable)**  Performs repeatable scrambling. The same input value produces the same output value on subsequent invocations.

**3 (Translate)**  Performs translation. The value data set is searched to find a matching input value. If a match is found, then the output value is taken from the output column of the matching record.

**4 (Exit)**  Invokes a scrambling user exit. FM/DB2 displays a panel where you can specify the user exit name and user exit parameters and options.

**Note:** For numeric columns, setting **Leading zeros** to YES causes leading zeros to be scrambled when random or repeatable scrambling (**Value** option blank) has been requested.

For example:

- With **Leading zeros** set to YES, 00123 may produce 56872 (zeros changed).
- With **Leading zeros** set to NO, 00123 may produce 00343 (zeros unchanged).

For repeatable scrambling, setting **Leading zeros** to YES guarantees a unique result.

**Parent panels**

- **“Column Selection/Edit panel” on page 407**
The Column Attributes (TIME) panel describes the formatting and data create attributes for a TIME column.

**Panel and field definitions**

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DGF2)</td>
<td>Column Attributes</td>
<td>Fixed Attributes:</td>
<td></td>
</tr>
<tr>
<td>Column name COL9_TIME</td>
<td>Type TIME</td>
<td>Format hh.mm AM or PM</td>
<td></td>
</tr>
<tr>
<td>Use Attributes:</td>
<td>Heading</td>
<td>Output width</td>
<td></td>
</tr>
<tr>
<td>Create Attributes:</td>
<td>Time value</td>
<td>(Use = for time now, * for runtime time)</td>
<td></td>
</tr>
<tr>
<td>Command ===&gt;</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Column name**

The name of the column.

**Type**

The DB2 data type of the column.

**Format**

The format in which the time is displayed or printed.

**Heading**

The alternative column heading.

**Output width**

The number of character positions used by view, edit, and print functions to show the column in TABL display or print format.

**Time value**

The initial value a column is to contain. Valid values are:

- `start time`
  - A time in a valid DB2 or local format.

---

Chapter 15. FM/DB2 panels and fields 405
Column Attributes panel (TIME)

* (asterisk)
The time the data is created. Not valid if the DB2 installation time format is LOCAL.

= The current time. Not valid if the DB2 installation time format is LOCAL.

Parent panels
- "Column Selection/Edit panel" on page 407

Child panels
None.

Related tasks
- "Editing the template" on page 192
- "Changing column attributes" on page 73
- Table 25 on page 362

Column Attributes panel (TIMESTAMP)

The Column Attributes (TIMESTAMP) panel describes the formatting and data create attributes for a TIMESTAMP column.

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>Column Attributes</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Fixed Attributes:
- Column name: TSTAMP
- Type: TIMESTAMP (3) WITH TIME ZONE
- Format: CCYY-MM-DD-HH.MM.SS.NNNsHH:MM

Use Attributes:
- Heading
- Output width

Create Attributes:
- Timestamp value
  (Use = for timestamp now, * for runtime timestamp)

The example shows a column defined as TIMESTAMP(3) WITH TIME ZONE. The Type value and the Format are different for columns with different TIMESTAMP data types.

Column name
The name of the column.

Type
The DB2 data type of the column.
Column Attributes panel (TIMESTAMP)

Format
The format in which the timestamp is displayed or printed.

Heading
The alternative column heading.

Output width
The number of character positions used by view, edit, and print functions to show the column in TABL display or print format.

Timestamp value
The initial value a column is to contain. Valid values are:

- start timestamp
  A timestamp in the DB2 timestamp format.

- * (asterisk)
  The date and time the data is created.

- =
  The current date and time.

Parent panels
- “Column Selection/Edit panel”

Child panels
None.

Related tasks
- “Editing the template” on page 192
- “Changing column attributes” on page 73

Related references
- Chapter 14, “DB2 data types,” on page 361

Column Distribution panel
FM/DB2 displays the Column Distribution panel if you use the CDI line command against an item listed on the “Tables, Views and Aliases panel” on page 704.

Column Part Statistics panel
FM/DB2 displays the Column Part Statistics panel if you use the PST line command against an item listed on the “Columns panel” on page 411.

Column Selection/Edit panel
You use the Column Selection/Edit panel to edit a template.
Panel and field definitions

### Process Options Utilities Help

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>Column Selection/Edit</td>
<td>Line 1 of 16</td>
<td></td>
</tr>
</tbody>
</table>

**TABLE FMNUSER.EMP**

------- Row Selection Criteria ---- (Use SQL/PF4 for full screen edit) ------- 1 Sel:

<table>
<thead>
<tr>
<th>Cmd</th>
<th>Seq</th>
<th>SHE</th>
<th>CL#</th>
<th>Column name</th>
<th>Data type(length)</th>
<th>Null</th>
<th>Default</th>
<th>Order</th>
<th>A/D</th>
</tr>
</thead>
<tbody>
<tr>
<td>S</td>
<td>1</td>
<td>EMPNO</td>
<td>CHARACTER(6)</td>
<td>None</td>
<td>——</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>S</td>
<td>2</td>
<td>FIRSTNME</td>
<td>VARCHAR(12)</td>
<td>None</td>
<td>——</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>S</td>
<td>3</td>
<td>MIDINIT</td>
<td>CHARACTER(1)</td>
<td>None</td>
<td>——</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>S</td>
<td>4</td>
<td>LASTNAME</td>
<td>VARCHAR(15)</td>
<td>None</td>
<td>——</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>S</td>
<td>5</td>
<td>WORKDEPT</td>
<td>CHARACTER(3)</td>
<td>Y</td>
<td>Null</td>
<td>——</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>S</td>
<td>6</td>
<td>PHONENO</td>
<td>CHARACTER(4)</td>
<td>Y</td>
<td>Null</td>
<td>——</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>S</td>
<td>7</td>
<td>HIREDATE</td>
<td>DATE</td>
<td>Y</td>
<td>Null</td>
<td>——</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>S</td>
<td>8</td>
<td>JOB</td>
<td>CHARACTER(8)</td>
<td>Y</td>
<td>Null</td>
<td>——</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>S</td>
<td>9</td>
<td>EDLEVEL</td>
<td>SMALLINT</td>
<td>Y</td>
<td>Null</td>
<td>——</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>S</td>
<td>10</td>
<td>SEX</td>
<td>CHARACTER(1)</td>
<td>Y</td>
<td>Null</td>
<td>——</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>S</td>
<td>11</td>
<td>BIRTHDATE</td>
<td>DATE</td>
<td>Y</td>
<td>Null</td>
<td>——</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>S</td>
<td>12</td>
<td>SALARY</td>
<td>DECIMAL(9,2)</td>
<td>Y</td>
<td>Null</td>
<td>——</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Seq**
The display sequence number of the column (in an FM/DB2 editor session) indicating either of these:
- The sequence number you have entered to change the sequence of the displayed columns.
- The sequence number generated by FM/DB2 as a result of a hold template editor prefix command (H, HH, or Hn) being entered in the **Cmd** prefix entry area.

**S** Column selection indicator. An asterisk (*) shows that the column is selected. Only selected columns appear in an FM/DB2 editor session.

**H** Column hold indicator. An “H” shows that the column is held. Held columns appear as the leftmost columns in an FM/DB2 editor session and are unaffected by scrolling left or right.

**Object type and name**
The object type (TABLE or VIEW) and the name of the DB2 object being processed.

**Row Selection Criteria**
Field where you can enter a freeform SQL WHERE clause or expression that limits the rows selected from DB2.

If the template already contains row selection criteria, FM/DB2 shows the corresponding WHERE clause in this field.

**Cmd** Prefix command area for entering template editor prefix commands:
- **S, SS, Sn, S** Selecting (or deselecting) columns.
- **H, HH, Hn** Holding (or freeing) columns.
- **E** Changing the attributes of a column.
Column Selection/Edit panel

E  Data edit indicator. An “E” shows that the column has been previously edited to add information such as data create, column headings, or scrambling options.

Cl#  The column number as defined in the DB2 catalog. The column number reflects the DB2 column number unless either of the following are true:
• The column is a generated column
• The column is not a column of the first table named in the SELECT statement used to create the result table

If either of these conditions is true, the column has a number greater than the maximum DB2 column number, for all specified columns in the first DB2 object named in the SELECT statement.

Column name  The name of the column.

Data type(length)  The DB2 data type (and optionally the length) of the column.

Null  Shows whether the column is defined as nullable.

Y  Nullable column

(blank)  DB2 null values are not permitted

Default  Default value information for the column:

Always  Either a ROWID or identity column defined with the generated always attribute.

Default  Either a ROWID or identity column defined with the generated by default attribute.

None  There is no default value for the column.

Null  The default value is the DB2 null value.

On type  The column has a default value that depends on the data type of the column:

Data type

Default value

Numeric

0

Fixed-length character data types

Blanks

Varying-length data types

String of length 0

DATE  Current date

TIME  Current time

TIMESTAMP  Current date and time
## Column Selection/Edit panel

### On value
The column has a default value that is defined in the DEFAULTVALUE column of the SYSIBM.SYSCOLUMNS catalog table.

### SQLID
The default value is the value of the SQL authorization ID of the process at the time the default value is used.

### USER
The default value is the value of the USER special register at the time the default value is used.

### Order
An optional sequence number indicating how to order the data, based on the data values for the column, when the data is retrieved from DB2.

### A/D
Sort sequence indicator. You can only specify this if an order sequence number (Order) is also specified.
- **A** Ascending
- **D** Descending

### Parent panels
- “DB2 View panel” on page 518
- “DB2 Edit panel” on page 505
- “Print Utility panel” on page 621
- “Copy Utility (“From”) panel” on page 418
- “Export Utility panel” on page 567
- “Data Create Utility panel” on page 495

### Child panels

<table>
<thead>
<tr>
<th>To display this panel...</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>“Table View panel” on page 700</td>
<td>If parent panel is DB2 View and you press F3</td>
</tr>
<tr>
<td>“DB2 View panel” on page 518</td>
<td>If parent panel is DB2 View and you press F12</td>
</tr>
<tr>
<td>“Table Edit panel” on page 691</td>
<td>If parent panel is DB2 Edit and you press F3</td>
</tr>
<tr>
<td>“DB2 Edit panel” on page 505</td>
<td>If parent panel is DB2 Edit and you press F12</td>
</tr>
<tr>
<td>“Table Browse panel” on page 689</td>
<td>If parent panel is DB2 Browse and you press F3</td>
</tr>
<tr>
<td>“DB2 Browse panel” on page 501</td>
<td>If parent panel is DB2 Browse and you press F12</td>
</tr>
<tr>
<td>“Template Save panel” on page 712</td>
<td>Press F3</td>
</tr>
<tr>
<td>“DB2 Template Description panel” on page 514</td>
<td>Primary command DESCRIBE</td>
</tr>
<tr>
<td>“Export “To” panel” on page 564</td>
<td>If parent panel is Export Utility and you press F3</td>
</tr>
<tr>
<td>“Column Attributes panel (alphanumeric)” on page 397</td>
<td>Line command E for an alphanumeric column</td>
</tr>
<tr>
<td>“Column Attributes panel (numeric)” on page 402</td>
<td>Line command E for a numeric column</td>
</tr>
</tbody>
</table>
Columns panel

You use the Columns panel to list column object types in the DB2 catalog.

Panel and field definitions

The columns that are displayed include the SEL field and columns of SYSIBM.SYSCOLUMNS.
Note:
1. The system option, Show all catalog table columns, affects which columns FM/DB2 displays when you use the Object List utility.
2. You can also customize which columns FM/DB2 displays when you use the Object List utility. For details, see the File Manager Customization Guide.

To view other columns on this panel, press the Right function key (F11).

For a complete list of columns, refer to “DB2 catalog tables” in the DB2 for z/OS SQL Reference relevant to your version of DB2.

SEL  Line command area.

   For a list of the line commands you can use on this panel, see Table 22 on page 250.

Parent panels
- “Object List Utility panel” on page 610
- “Constraint Columns for panel” on page 415
- “Columns in Table panel” on page 413
- “Privileges panels” on page 624

Child panels

<table>
<thead>
<tr>
<th>To display this panel...</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>“Tables, Views and Aliases panel” on page 704</td>
<td>Line command AT</td>
</tr>
<tr>
<td>See “Comment panels” on page 413</td>
<td>Line command COM</td>
</tr>
<tr>
<td>“Column Distribution panel” on page 407</td>
<td>Line command DI</td>
</tr>
<tr>
<td>“Distinct Types panel” on page 523</td>
<td>Line command DT</td>
</tr>
<tr>
<td>“Grant privileges panels” on page 587</td>
<td>Line command G</td>
</tr>
<tr>
<td>“Columns panel” on page 411</td>
<td>Line command H</td>
</tr>
<tr>
<td>“Details panels” on page 522</td>
<td>Line command I</td>
</tr>
</tbody>
</table>
Columns in Index panel

FM/DB2 displays the Columns in Index panel if you use the COL line command against an item listed on the **Indexes panel** on page 595.

Columns in Table panel

FM/DB2 displays the Columns in Table panel if you use the COL line command against an item listed on:

- The **“Tables, Views and Aliases panel” on page 704**.
- The **“Table Check Constraint Dependencies panel” on page 691**.

Comment panels

These panels are:
- Comment Column panel
- Comment Distinct Type panel
- Comment Procedure panel
- Comment Specific Function panel
- Comment Table panel
- Comment Trigger panel

FM/DB2 displays one of these panels when you specify the COM line command against an object displayed in a list of DB2 objects.

Compiler Language Selection panel

You use the Compiler Language Selection panel to set the language to be used for compiling the current source member to create a template. The default language is COBOL. This option is saved in your ISPF profile and persist from one FM/DB2 session to the next.
Language Selection

The following language can be specified for the compilation of the source member to create a template:

1. **COBOL**
   - Use the COBOL compiler. This is the default setting.
2. **PL/I**
   - Use the PL/I compiler.
3. **Auto detect**
   - Determine which compiler to use
4. **HLASM**
   - Use the HLASM compiler

Processing Options

- **Override compiler options for template update**
  - When selected, overrides the compiler options found in the template being updated with the current compiler options.
  - Foreground processing always overrides the compiler options for older templates that do not contain compiler options.
  - You should select this option for batch template update of older templates to generate compiler options in the JCL.

- **Preserve copybook library**
  - Ensures that, if a copybook still exists in the library that it was previously found in and that library is in the list that the update is using, then that version of the copybook is used.
  - If you do not select this option, or the copybook no longer exists in the library it was previously found in, or that library is not in the list the update is using, then the utility searches the libraries in the order they are listed and uses the first version of the copybook that it finds.
Compiler Language Selection panel

Show copybook name with record type field name
Select this option to make the copybook name for a record layout visible during template edit and for an edit, view or browse session where a copybook or template is being used.

Note: For the editor session where the layout name is specified it is prefixed with the name of the associated copybook. This may cause the layout name to be truncated, depending on its length.

Parent panels
• “Set System Processing Options panel” on page 673

Child panels
None.

Related tasks
• “Compiler language selection (option 0.4)” on page 42
• “How FM/DB2 compiles a copybook into a template” on page 53

Constraint Columns for panel
FM/DB2 displays the Constraint Columns for panel if you use the COL line command against an item listed on the “Table Check Constraints panel” on page 691.

Copy Options panel
You use the Copy Options panel to set the global (or initial) options used by the Copy utility (3.3). These options are saved in your ISPF profile and persist from one FM/DB2 session to the next.

The global Copy options are restored at the start of each Copy session and can be temporarily changed for a copy session by selecting Edit Copy Options on the Copy “To” entry panel.
### Panel and field definitions

<table>
<thead>
<tr>
<th>Process Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>Copy Options</td>
<td>Global Settings</td>
</tr>
</tbody>
</table>

#### From Table Concurrency Option:
- Use uncommitted read
- Enter "/" to select option

#### To Table Locking Option:
- Locking
- Enter "/" to select option
  - 1. None
  - 2. Share mode
  - 3. Exclusive mode

#### Processing Options:
- Duplicate key processing
  - 1. Ignore
  - 2. Update
  - Max duplicates
  - ALL

---

Use uncommitted read

Indicates whether to enable retrieval of uncommitted data for the Copy session:

/ Retrieve data, including data not yet committed by another user or application program. The resultant SELECT statement used by FM/DB2 to retrieve table data has the suffix “WITH UR” appended. This allows reading through any locks which might be held on that table. This is the default setting.

(blank) Only retrieve committed data from tables without write locks.

Note: If FM/DB2 attempts to retrieve data that has a write lock, an SQL error can result.

Locking

FM/DB2 provides options to lock the table to prevent other users from either updating or reading data. Use these options with caution.

1. None
   - FM/DB2 does not issue an explicit table lock for the object being copied. Locking of rows and pages within the object is managed by DB2 according to FM/DB2 concurrency options and DB2 installation and object definition options. This is the default setting.

2. Share mode
   - FM/DB2 issues a LOCK TABLE IN SHARE MODE statement before reading the data from DB2. This restricts all other users to read-only operations on the object.
Copy Options panel

3. Exclusive mode
   FM/DB2 issues a LOCK TABLE IN EXCLUSIVE MODE statement before reading the data from DB2. This prevents all other users from accessing the object, unless the other process is running with an isolation level of uncommitted read.

Duplicate key processing
   You can specify what action FM/DB2 takes if it encounters a duplicate key. Set this option to one of the following:
   1. Ignore
      Existence of duplicate key ignored. This is the default.
   2. Update
      Updates duplicate rows when DB2 returns a duplicate key error (SQL error code -803) in response to an INSERT SQL statement. When this option is selected, FM/DB2 attempts to update the existing row. All columns of the table that are part of at least one unique index are used to identify the row. No comparison is made between the copied row and the existing row prior to the update operation (that is, the update is done regardless of whether the copied row and existing row are the same). When using this option, it is recommended that a DB2 table name, rather than a view name, be specified. FM/DB2 does not attempt to update existing rows in a view defined on a view.
   For _______ duplicates
      Specifies the maximum number of duplicates permitted. Valid values are:
      • A number in the range 0 to 2147483647. If FM/DB2 detects more than this number of duplicates during the copy process, the copy is canceled and all changes are rolled back.
      • ALL

Delete existing rows
   Determines whether FM/DB2 deletes all existing rows in the table before copying the data.
   / Delete all existing rows.
   (blank) Do not delete existing rows. This is the default setting.

Ignore RI/Constraint errors
   Determines what processing occurs when the Copy utility encounters an SQLCODE-530 (RI error - no primary key) or SQLCODE-545 (Constraint error).
   / The Copy process ignores the error and continues with the next row. The row is not copied or updated and is not included in the copy count.
   (blank) Copy utility canceled. This is the default setting.

Native unicode processing
   Determines how FM/DB2 processes Unicode encoded data when copying data between two Unicode encoded tables. When the option is enabled, Unicode encoded data is copied between Unicode encoded tables without the potential for character substitution. When the option is not enabled, the option has no effect; that is, the default processing is used. When either or both of the FROM or TO tables are not Unicode encoded, the option has
Copy Options panel

no effect, even if enabled. When this option and the batch execution options are enabled, "NATIVE= YES" is shown in the generated JCL.

/ Process Unicode data natively.

(Blank)

Do not Process Unicode data natively. This is the default setting.

Parent panels

- "Copy Utility ("To") panel" on page 424

Child panels

None.

Related tasks

- "Setting options for the current Copy session" on page 197
- "Copy utility options (option 3.3)" on page 44

Copy Utility ("From") panel

You use the Copy Utility ("From") panel to enter the name of an object to be copied.

Panel and field definitions

| Location       | The location name for the DB2 object. You can specify a generic name to select from a list. If you enter a value in this field, synonym object names are not recognized or included in the selection list. |
Copy Utility (“From”) panel

Database
The database name for the DB2 object. You can use this field to optionally specify a database to further define the object to be processed. If you enter a value in this field:
- The values displayed in the Owner and Name selection lists are limited to those values defined within the specified database.
- Synonym names are not recognized or included in the selection list.

Owner
The name of the owner of the object. You can specify a generic name to select from a list. To specify a synonym in the object Name field, either leave this field blank, or specify a generic name that includes the current SQLID or specify the current SQLID. Synonyms are not recognized or included on the table selection list if any of the Table space, Database, or Location fields are non-blank.
If the Translate DB2 object names option is selected, lowercase characters are converted to uppercase. If this option is not selected, lowercase characters are not converted to uppercase; in this situation, a lowercase owner name such as “sysibm” does not match “SYSIBM” in the DB2 catalog.

Table space
The table space name for the DB2 object. To further define the object to be processed, you can use this field to optionally specify a table space. If you enter a value in this field:
- The values displayed in the Owner and Name selection lists are limited to those values defined within the specified table space.
- Synonym names are not recognized, nor included in the selection list.

Name
The name of the object name to be processed. You can specify a generic name to select from a list. To specify a synonym in this field, type the name of the synonym and ensure that the Owner field is blank, contains a generic name that includes the current SQLID, or contains the current SQLID. Synonyms are not recognized or included in the table selection list if any of the Table space, Database, or Location fields are non-blank.
If the Translate DB2 object names option is selected, lowercase characters are converted to uppercase. If this option is not selected, lowercase characters are not converted to uppercase; in this situation, a lowercase owner name such as “sysibm” does not match “SYSIBM” in the DB2 catalog.

Copy count
The maximum number of rows to be copied from a DB2 object. It must be one of the following:
- A number in the range 1–99999999,
- An asterisk (*), or
- ALL. This is the default, where all rows selected from the DB2 object are copied.
When the “From” template includes a “Row Selection Criteria” (or WHERE clause), only those rows matching the selection criteria are considered eligible to be copied.

Data set name
The name of a data set that contains, or will contain, the template. It can be a fully-qualified data set name or a pattern. The name may include a
Copy Utility ("From") panel

member name or name pattern in parenthesis. If the member is specified here, the associated Member field must be empty.

Member
If you specified the name of a partitioned data set (PDS) without including a member name or name pattern in parenthesis in the Data set name field, then you can use this field to specify the member name or a member name pattern.

Template usage
The Copy Utility panel provides four options for template processing:

1. **Above**
   Requires that you enter the name of a template data set (and optionally a member name) in the Template section of the panel. If you do not specify a member name, or if you specify a pattern, FM/DB2 displays a member selection list. After you have specified the member containing a template, processing continues using this template in place of the automatically-generated template for the table (for a detailed explanation, see Template Usage option 3).

2. **Previous**
   Uses the last (previously used) template for this table.

3. **Generate from table**
   Generates a template based on the information in the DB2 catalog for the specified table. This is the default setting. If you have specified a member name in the Template section of the panel, it is ignored. FM/DB2 generates the required template in memory.

4. **Generate/Replace**
   Generates a template based on the information in the DB2 catalog for the specified table, and saves this information in the member specified in the From Template section of the panel. If the member already exists, it is replaced. Any prior customization is lost.

You can select the following option by entering a “/” or an “A”:

**Edit template**
Use to edit the template before copying the data.

**Copy panel values**
Use to copy the DB2 object name and template details to the “To” panel.

**Create audit trail**

**Note:** "(*)" is appended to the option name for SAF-controlled auditing when required by SAF rules.
Controls whether access to DB2 by FM/DB2 is written to the FM/DB2 audit log. This option only appears when SAF-rule controlled auditing is in effect.

FM/DB2 may write audit records to either SMF, or a data set.

When SAF rules are used to control FM/DB2 auditing, the decision to write audit records is based on the DB2 object name being processed, and the current userid (TSO logonid). Therefore auditing may be required when processing one DB2 object, but not another. In addition, it is possible that the auditing requirement cannot be changed for a particular DB2 object; this might be different for another DB2 object, and the requirement may be different for other TSO userids.
Copy Utility ("From") panel

If you do not have the authority to change the audit setting when processing a particular DB2 object, FM/DB2 prompts you if you enter the wrong setting. If you do have the authority to set the audit setting for a particular DB2 object, FM/DB2 honors whatever value is specified in this field.

Parent panels

- "Utility Functions panel" on page 720
- "Tables, Views and Aliases panel" on page 704

Child panels

<table>
<thead>
<tr>
<th>To display this panel</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;Column Selection/Edit panel&quot; on page 407</td>
<td>Select Edit template</td>
</tr>
<tr>
<td>&quot;Copy Utility (&quot;To&quot;) panel&quot; on page 424</td>
<td>Press Enter</td>
</tr>
</tbody>
</table>

Related concepts

- "Information stored in a template" on page 50
- "SAF-rule controlled auditing" on page 47

Related tasks

- "Selecting options on FM/DB2 panels" on page 24
- Chapter 3, "Working with templates," on page 49
- "Where you can use templates" on page 4
- "Copying data from one DB2 object to another" on page 193
- "Copying data from a VSAM or QSAM file" on page 198
- "Copying data to a VSAM or QSAM file" on page 221
- "Specifying a data set and a member name" on page 30
- "FM/DB2 system options (option 0.2)" on page 41

COPY Utility (Index Spaces) panel

You use the COPY Utility (Index Spaces) panel to generate utility control statements using the COPY INDEXSPACE form of the COPY utility.

The format of the COPY Utility (Index Spaces) panel depends on which, if any, of the filter fields you specified on the DB2 Utilities panel:

- The panel shown in Figure 103 on page 422 is displayed by default, or if you specified information in the **Database name** filter field, the **Table/index space name** filter field, or both.
- The panel shown in Figure 104 on page 422 is displayed if you specified information in the **Table/index owner** filter field, the **Table/index name** filter field, or both.

**Note:** Indexes need to have been defined with the COPY YES clause. Indexes which have not been defined with the COPY YES clause cannot be selected for image copying in the panel.
COPY Utility (Index Spaces) panel

Panel and field definitions

Sel  Selection column. To select an index space, type S in this column. Selected index spaces show an asterisk (*) in the PRC column.

PRC  Processing indicator. An asterisk (*) indicates the index space is selected.

Database Name

The name of the database in which the index space is defined.

Index Space Name

The name of the index space.

Part'n Number

The partition number for the table space. This is 0 for a non-partitioned table space. A non-zero value indicates a partition of a partitioned table space. By default, when you select a non-partitioned table space for
COPY Utility (Index Spaces) panel

processing, the utility control statements do not include the DSNUM ALL parameter. When you select a partition, the utility control statements include a DSNUM nnn clause (nnn is the partition number you selected).

Primary
The primary allocation value for the table space. See the DB2 Utility Guide and Reference for a full explanation. This value gives some indication of the size of the table space (but is not definitive).

Secondary
The secondary allocation value for the table space. See the DB2 Utility Guide and Reference for a full explanation.

Space
The space allocation for the table space, as reported by the STOSPACE utility. See the DB2 Utility Guide and Reference for a full explanation. This value gives an accurate indication of the size of the table space when the STOSPACE utility was run.

Index Creator
The name of the index creator.

Index Name
The name of the index.

Parent panels
- “DB2 Utilities panel” on page 515

Child panels
None.

Related tasks
- “COPY utility” on page 283
- “COPY (index spaces)” on page 285

COPY Utility (Index Spaces) with LISTDEF panel

You use the COPY Utility (Index Spaces) with LISTDEF panel to generates utility control statements using the COPY TABLESPACE form of the COPY utility.
Panel and field definitions

You can enter commands in this field to manipulate the panel rows.

**Cmd**
- **I(n)**: Insert (n) new row(s)
- **R(n)**: Repeat the current row (n times)
- **D(n)**: Delete (n) row(s)
- **S(n)**: Select (n) rows(s)

An asterisk (*) in the PRC column shows that the LISTDEF in the row is selected. The S command toggles the selection of rows; unselected rows become selected and selected rows become unselected.

**LISTDEF Name**
On entry to the panel, this field is set to the value in the LISTDEF name field in the DB2 Utilities panel.

**Parent panels**
- *DB2 Utilities panel* on page 515

**Child panels**
None.

**Related tasks**
- *COPY utility* on page 283
- *COPY (index spaces)* on page 285

**Copy Utility (“To”) panel**

You use the Copy Utility (“To”) panel to enter the name of the DB2 object that is the target of the copy operation. You also use this panel to specify the name of a template that is to provide the column mapping and initialization information.
Panel and field definitions

Location
The location name for the DB2 object. You can specify a generic name to select from a list. If you enter a value in this field, synonym object names are not recognized or included in the selection list.

Database
The database name for the DB2 object. You can use this field to optionally specify a database to further define the object to be processed. If you enter a value in this field:

- The values displayed in the Owner and Name selection lists are limited to those values defined within the specified database.
- Synonym names are not recognized or included in the selection list.

Owner
The name of the owner of the object. You can specify a generic name to select from a list. To specify a synonym in the object Name field, either leave this field blank, or specify a generic name that includes the current SQLID or specify the current SQLID. Synonyms are not recognized or included on the table selection list if any of the Table space, Database, or Location fields are non-blank.

If the Translate DB2 object names option is selected, lowercase characters are converted to uppercase. If this option is not selected, lowercase characters are not converted to uppercase; in this situation, a lowercase owner name such as “sysibm” does not match “SYSIBM” in the DB2 catalog.

Table space
The table space name for the DB2 object. To further define the object to be processed, you can use this field to optionally specify a table space. If you enter a value in this field:

- The values displayed in the Owner and Name selection lists are limited to those values defined within the specified table space.
- Synonym names are not recognized, nor included in the selection list.
Copy Utility ("To") panel

Name  The name of the object name to be processed. You can specify a generic name to select from a list. To specify a synonym in this field, type the name of the synonym and ensure that the Owner field is blank, contains a generic name that includes the current SQLID, or contains the current SQLID. Synonyms are not recognized or included in the table selection list if any of the Table space, Database, or Location fields are non-blank.

If the Translate DB2 object names option is selected, lowercase characters are converted to uppercase. If this option is not selected, lowercase characters are not converted to uppercase; in this situation, a lowercase owner name such as “sysibm” does not match “SYSIBM” in the DB2 catalog.

Data set name  The name of a data set that contains, or will contain, the template. It can be a fully-qualified data set name or a pattern. The name may include a member name or name pattern in parenthesis. If the member is specified here, the associated Member field must be empty.

Member  If you specified the name of a partitioned data set (PDS) without including a member name or name pattern in parenthesis in the Data set name field, then you can use this field to specify the member name or a member name pattern.

Template usage  The Copy Utility panel provides four options for template processing:

1. Above
   Requires that you enter the name of a template data set (and optionally a member name) in the Template section of the panel. If you do not specify a member name, or if you specify a pattern, FM/DB2 displays a member selection list. After you have specified the member containing a template, processing continues using this template in place of the automatically-generated template for the table (for a detailed explanation, see Template Usage option 3).

2. Previous
   Uses the last (previously used) template for this table.

3. Generate from table
   Generates a template based on the information in the DB2 catalog for the specified table. This is the default setting. If you have specified a member name in the Template section of the panel, it is ignored. FM/DB2 generates the required template in memory.

4. Generate/Replace
   Generates a template based on the information in the DB2 catalog for the specified table, and saves this information in the member specified in the To Template section of the panel. If the member already exists, it is replaced. Any prior customization is lost.

You can select the following options by entering a “/” or an “A”:

Edit copy options
   Displays the copy options panel

Edit template mapping
   Use to edit the template for the table to which you are copying the data, before copying the data.
Batch execution
Generates the JCL to perform the FM/DB2 Copy function as a batch job. You can edit the JCL and submit the job manually once the JCL has been generated.

Parent panels
- "Copy Utility (“From”) panel" on page 418

Child panels
<table>
<thead>
<tr>
<th>To display this panel...</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;Copy Options panel&quot; on page 415</td>
<td>Select Edit copy options</td>
</tr>
<tr>
<td>&quot;Template Mapping panel” on page 710</td>
<td>Select Edit template mapping</td>
</tr>
</tbody>
</table>

Related tasks
- "Specifying a data set and a member name” on page 30
- "Selecting options on FM/DB2 panels” on page 24
- "Copying data from one DB2 object to another” on page 193
- "Mapping data” on page 80
- "Copying data from one DB2 object to another” on page 193

COPY Utility (Table Spaces) panel
You use the COPY Utility (Table Spaces) panel to generate utility control statements using the COPY TABLE SPACE form of the COPY utility.

Panel and field definitions

```
<table>
<thead>
<tr>
<th>Sel</th>
<th>PRC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Selection column. To select a table space, type S in this column. Selected table spaces show an asterisk (*) in the PRC column.</td>
<td></td>
</tr>
</tbody>
</table>
```

```
<table>
<thead>
<tr>
<th>Sel</th>
<th>PRC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Selection column. To select a table space, type S in this column. Selected table spaces show an asterisk (*) in the PRC column.</td>
<td></td>
</tr>
<tr>
<td>Selection column. To select a table space, type S in this column. Selected table spaces show an asterisk (*) in the PRC column.</td>
<td></td>
</tr>
<tr>
<td>Selection column. To select a table space, type S in this column. Selected table spaces show an asterisk (*) in the PRC column.</td>
<td></td>
</tr>
<tr>
<td>Selection column. To select a table space, type S in this column. Selected table spaces show an asterisk (*) in the PRC column.</td>
<td></td>
</tr>
<tr>
<td>Selection column. To select a table space, type S in this column. Selected table spaces show an asterisk (*) in the PRC column.</td>
<td></td>
</tr>
<tr>
<td>Selection column. To select a table space, type S in this column. Selected table spaces show an asterisk (*) in the PRC column.</td>
<td></td>
</tr>
</tbody>
</table>
```

```
<table>
<thead>
<tr>
<th>Sel</th>
<th>PRC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Selection column. To select a table space, type S in this column. Selected table spaces show an asterisk (*) in the PRC column.</td>
<td></td>
</tr>
<tr>
<td>Selection column. To select a table space, type S in this column. Selected table spaces show an asterisk (*) in the PRC column.</td>
<td></td>
</tr>
<tr>
<td>Selection column. To select a table space, type S in this column. Selected table spaces show an asterisk (*) in the PRC column.</td>
<td></td>
</tr>
<tr>
<td>Selection column. To select a table space, type S in this column. Selected table spaces show an asterisk (*) in the PRC column.</td>
<td></td>
</tr>
<tr>
<td>Selection column. To select a table space, type S in this column. Selected table spaces show an asterisk (*) in the PRC column.</td>
<td></td>
</tr>
<tr>
<td>Selection column. To select a table space, type S in this column. Selected table spaces show an asterisk (*) in the PRC column.</td>
<td></td>
</tr>
</tbody>
</table>
```
### COPY Utility (Table Spaces) panel

<table>
<thead>
<tr>
<th><strong>Database Name</strong></th>
<th>The name of the database in which the table space is defined.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Table Space Name</strong></td>
<td>The name of the table space.</td>
</tr>
<tr>
<td><strong>FCP</strong></td>
<td>Full Copy flag:</td>
</tr>
<tr>
<td>Y</td>
<td>Take a full image copy. This is the default setting.</td>
</tr>
<tr>
<td>N</td>
<td>Take a partial image copy.</td>
</tr>
<tr>
<td><strong>Change Limits (One Two)</strong></td>
<td>You can specify 1 or 2 values indicating the percentage of changed pages needed to trigger a partial image copy, and a full image copy. See the DB2 Utility Guide and Reference for a detailed explanation. A value entered in either of the Change Limits fields overrides the FCP setting.</td>
</tr>
<tr>
<td><strong>RPO</strong></td>
<td>Report Only. This only takes effect if you have specified a change limit value (see Change Limits).</td>
</tr>
<tr>
<td>Y</td>
<td>Only image copy information is displayed. Image copies are not taken, only recommended. FM/DB2 generates a DD DUMMY in the JCL for the SYSCOPY DD name.</td>
</tr>
<tr>
<td>N</td>
<td>Image copy information is displayed and image copies taken. This is the default setting.</td>
</tr>
</tbody>
</table>

The remaining columns of the ISPF table are display-only fields showing information from the DB2 catalog.

| **Part'n Number** | The partition number for the table space. This is 0 for a non-partitioned table space. A non-zero value indicates a partition of a partitioned table space. By default, when you select a non-partitioned table space for processing, the utility control statements do not include the DSNUM ALL parameter. When you select a partition, the utility control statements include a DSNUM mnn clause (mnn is the partition number you selected). |
| **Number Tables** | The number of tables defined within the table space. |
| **Primary** | The primary allocation value for the table space. See the DB2 Utility Guide and Reference for a full explanation. This value gives some indication of the size of the table space (but is not definitive). |
| **Secdry** | The secondary allocation value for the table space. See the DB2 Utility Guide and Reference for a full explanation. |
| **Space** | The space allocation for the table space, as reported by the STOSPACE utility. See the DB2 Utility Guide and Reference for a full explanation. This value gives an accurate indication of the size of the table space when the STOSPACE utility was run. |

**Parent panels**
- "DB2 Utilities panel" on page 515

**Child panels**
None.
COPY Utility (Table Spaces) panel

Related tasks
- "COPY utility" on page 283

COPY Utility (Table Spaces) with LISTDEF panel

You use the COPY Utility (Table Spaces) with LISTDEF panel to generate utility control statements using the COPY TABLE SPACE form of the COPY utility, and with a LISTDEF provided.

Panel and field definitions

<table>
<thead>
<tr>
<th>Cmd</th>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>FM/DB2 (DFG2)</td>
<td>COPY Utility (Table Spaces)</td>
<td>Row 1 to 1 of 1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>P</td>
<td>R</td>
<td>Change</td>
<td></td>
</tr>
<tr>
<td>Cmd</td>
<td>LISTDEF</td>
<td>COPYDDN</td>
<td>Full Limits</td>
<td>Report</td>
</tr>
<tr>
<td>N</td>
<td>LETTER</td>
<td>Y</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>N</td>
<td>LETTER</td>
<td>Y</td>
<td>N</td>
<td>N</td>
</tr>
</tbody>
</table>

An asterisk (*) in the PRC column shows that the LISTDEF in the row is selected. The S command toggles the selection of rows; unselected rows become selected and selected rows become unselected.

LISTDEF Name

On entry to the panel, this field is set to the value in the LISTDEF name field in the DB2 Utilities panel.

Parent panels
- "DB2 Utilities panel" on page 515

Child panels

None.
Create Alias panel

You use the Create Alias panel to create an alias for a table or a view at the correct location.

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td></td>
<td>Create Alias</td>
<td></td>
</tr>
</tbody>
</table>

**Alias Details:**
- **Owner** ........ + (optional)
- **Name** ........ +

**Table or View Details:**
- **Location** ........ +
- **Owner** ........ +
- **Name** ........ +

**Owner (Alias Details)**
The owner of the alias you want to create (optional). If you leave this field blank, FM/DB2 uses the current SQL ID as a default ID. Otherwise, type the full name of the owner.

**Name Alias Details**
The name of the alias you want to create (mandatory).

**Location**
The location of the table or view for which you want to create an alias (optional). If you leave this field blank, FM/DB2 uses the location of the local DB2 system.

**Owner (Table or View Details)**
The owner of the table or view for which you want to create an alias (optional). If you leave this field blank, FM/DB2 uses the current SQL ID as a default ID. Otherwise, type the full name of the owner.

**Name (Table or View Details)**
The name of the table or view for which you want to create an alias (mandatory).

Parent panels
- “DB2 Object Functions panel” on page 510
- “Tables, Views and Aliases panel” on page 704
- “Synonyms panel” on page 687
Create Alias panel

Child panels

None.

Related tasks

- Chapter 5, “Creating and dropping DB2 objects,” on page 163
- “Creating an alias” on page 183

Create Auxiliary Table panel

You use the Create Auxiliary Table panel to create a new auxiliary table.

Panel and field definitions

<table>
<thead>
<tr>
<th>Process Options Utilities Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2) Create Auxiliary Table</td>
</tr>
</tbody>
</table>

Auxiliary table:
- Owner ........ + (optional)
- Name ................ +

Table Space:
- Name ................ (optional)

Database ........

Base Table:
- Owner ........ + (optional)
- Name ................ +

Base Table Column:
- Name ................ (optional)
- Partition ........ (optional)

Owner (Auxiliary Table)
- Specifies the owner of the auxiliary table.

Name (Auxiliary Table)
- Specifies the name of the auxiliary table.

Name (Table Space)
- Specifies the LOB table space in which the auxiliary table is created.

Database (Table Space)
- Specifies the database to which the specified table space belongs.

Owner (Base Table)
- Specifies the owner of the base table containing the LOB column that is stored in the auxiliary table.

Name (Base Table)
- Specifies the name of the base table containing the LOB column that is stored in the auxiliary table.

Name (Base Table Column)
- Specifies the name of the LOB column in the base table that is stored in the auxiliary table.
Create Auxiliary Table panel

**Partition (Base Table Column)**
For base tables defined in partitioned table spaces, specifies the partition of the base table for which this auxiliary table stores data.

**Parent panels**
- "DB2 Object Functions panel" on page 510

**Child panels**
None.

**Related tasks**
- Chapter 5, “Creating and dropping DB2 objects,” on page 163
- "Creating an auxiliary table" on page 189

Create Database panel

You use the Create Database panel to create a new database.

**Panel and field definitions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Specify the name of the DB2 database you want to create.</td>
</tr>
<tr>
<td>Storage group name</td>
<td>Names the DB2 storage group to be used, as the default group for DASD allocation for table spaces and indexes in the database.</td>
</tr>
<tr>
<td>Buffer pool for table spaces</td>
<td>Specifies the default buffer pool name to be used for table spaces created within the database.</td>
</tr>
<tr>
<td>Database Options:</td>
<td>Database Usage (only for user databases)</td>
</tr>
<tr>
<td>- User database (default)</td>
<td>1. EBCDIC (default)</td>
</tr>
<tr>
<td>- Work file database</td>
<td>2. ASCII</td>
</tr>
<tr>
<td>- Temporary database</td>
<td>3. UNICODE</td>
</tr>
<tr>
<td>Member Selection:</td>
<td>(only valid for work file databases)</td>
</tr>
<tr>
<td>Command</td>
<td>F1=Help, F2=Split, F3=Exit, F4=Expand, F7=Backward, F8=Forward, F9=Swap, F10=Left, F11=Right, F12=Cancel</td>
</tr>
</tbody>
</table>

**Note:**

432   File Manager/DB2 Data V13R1 User’s Guide
Create Database panel

8K and 16K buffer pools are not supported for work file databases.

**Buffer pool for indexes**
Specifies the default buffer pool name to be used for the indexes created within the database.

**Database Usage**
1. **User database (default)**
   Specifies the database is an ordinary database for general user tables and indexes.
2. **Work file database**
   Specifies the database is a work file database for use in a data sharing environment.
3. **Temporary database**
   Specifies the database is for declared temporary tables only.

**Data Encoding (only for user databases)**
1. **EBCDIC (default)**
   Specifies the encoding scheme.
2. **ASCII**
   Specifies the encoding scheme.
3. **UNICODE**
   Specifies the encoding scheme.

**Parent panels**
- “DB2 Object Functions panel” on page 510
- “Databases panel” on page 500

**Child panels**
None.

**Related tasks**
- Chapter 5, “Creating and dropping DB2 objects,” on page 163
- “Creating a database” on page 173

Create Distinct Type panel

You use the Create Distinct Type panel to create a new distinct type.
Create Distinct Type panel

Panel and field definitions

<table>
<thead>
<tr>
<th>Process Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>Create Distinct Type</td>
<td></td>
</tr>
</tbody>
</table>

**Distinct Type:**
- **Schema**
  - The schema name in which to create the function.
- **Name**
  - The name of the user-defined function.
- **Source Type**
  - DB2 data type in which the distinct type is based.
- **Length/Precision**
  - Data length of the distinct type. For distinct types based on the DECIMAL data type, this is the precision.
- **Scale**
  - For distinct types based on the DECIMAL data type.
- **For Data Encoding Method**
  - 1. **SBCS**
  - 2. **Mixed**
  - 3. **Bit**
  - 4. **EBCDIC**
  - 5. **ASCII**
  - 6. **UNICODE**

**Encoding Method**
- 1. **EBCDIC** (default)
  - Specifies the encoding scheme for the distinct data type.
- 2. **ASCII**
  - Specifies the encoding scheme for the distinct data type.
- 3. **UNICODE**
  - Specifies the encoding scheme for the distinct data type.

EBCDIC and ASCII specifies the encoding scheme for the distinct data type.

**Parent panels**
- “DB2 Object Functions panel” on page 510
- “Distinct Types panel” on page 523

**Child panels**

None.
Create Distinct Type panel

Related tasks
- Chapter 5, “Creating and dropping DB2 objects,” on page 163
- "Creating a distinct type" on page 186

Create Function: Option List (1/2) panel

You use the Create Function: Option List (1/2) panel to specify information to define options for the function to be created.

Panel and field definitions

<table>
<thead>
<tr>
<th>Panel and field definitions</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>External Name</strong></td>
</tr>
<tr>
<td><strong>Scratch Pad</strong></td>
</tr>
<tr>
<td><strong>Package Collection</strong></td>
</tr>
<tr>
<td><strong>WLM Environment</strong></td>
</tr>
<tr>
<td><strong>Time Limit (CPU units)</strong></td>
</tr>
<tr>
<td><strong>LANGUAGE (Required)</strong></td>
</tr>
<tr>
<td><strong>SQL</strong></td>
</tr>
</tbody>
</table>

| Command ===| F1=Help F2=Split F3=Exit F4=Expand F7=Backward F8=Forward |

- **ASSEMBLE**
- **C**
- **COBOL**
- **PLI**

- **1. Reads SQL Data**
- **2. No SQL**
- **3. Modifies SQL Data**
- **4. Contains SQL**

- **1. DB2**
- **2. User**
- **3. Definer**

- **1. Allowed**
- **2. Disallowed**
Create Function: Option List (1/2) panel

Security
   Specifies the interaction of the function with an external security product:

Parallel
   Specifies if DB2 can consider parallelism for the function:

Run-time options
   Specifies the Language Environment® run-time options to be used for the function.

Parent panels
   • “Create Function panel” on page 437

Child panels

None.

Related tasks
   • “Creating a function” on page 186

Create Function: Option List (2/2) panel

You use the Create Function: Option List (2/2) panel to further specify information to define options for the function to be created.

Panel and field definitions

You can select the following options by entering a “/”:

Main Program
   The program runs as a main routine, otherwise it runs as a subroutine.

Stay Resident
   The function load module remains in memory when the function ends.

Deterministic
   The function always returns the same result for the same set of input values.
Called on Null Input
The function is called even if one or more of the input parameters has a null value.

No external action
The function does not take any action that changes the state of an object DB2 does not manage.

Final Call
A final call is made to the function to allow it to perform any required cleanup operations.

DBINFO Argument Passed
Additional DB2 information is passed when the function is called.

Parent panels
• “Create Function panel”

Child panels
None.

Related tasks
• “Creating a function” on page 186

Create Function panel
You use the Create Function panel to create a new function.

Panel and field definitions

Create Function: Option List (2/2) panel

Called on Null Input
The function is called even if one or more of the input parameters has a null value.

No external action
The function does not take any action that changes the state of an object DB2 does not manage.

Final Call
A final call is made to the function to allow it to perform any required cleanup operations.

DBINFO Argument Passed
Additional DB2 information is passed when the function is called.

Parent panels
• “Create Function panel”

Child panels
None.

Related tasks
• “Creating a function” on page 186

Create Function panel
You use the Create Function panel to create a new function.

Panel and field definitions

Schema
The schema name in which to create the function.

Name
The name of the user-defined function.
Create Function panel

Unique Name
Specifies a unique name for the function that is used by certain SQL statements and DB2 commands.

Creation Options
Specify the number of the creation option you want to use:
1. Parameters
Displays the Create Function Parameters panel for defining parameters for the function.
2. Parameter Types
Displays the Create Function Parameter Types panel for specifying additional type information for the function parameters.
3. Returned data type
Displays the Create Function: Returned Data Type panel for specifying the data type returned by the function.
4. Option List
Displays the Create Function Option List (1/2) panel for specifying various options for further defining the function.
5. Option List (More)
Displays the Create Function Option List (2/2) panel for specifying additional options for the function.
6. Create the Function
When you have finished specifying parameter, data type, and option information as required, select this creation option to create the function.

Parent panels
- “DB2 Object Functions panel” on page 510
- “Functions panel” on page 584

Child panels

<table>
<thead>
<tr>
<th>To display this panel...</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>“Create Function: Parameters panel” on page 439</td>
<td>Creation option 1</td>
</tr>
<tr>
<td>“Create Function: Parameter Types panel”</td>
<td>Creation option 2</td>
</tr>
<tr>
<td>“Create Function: Returned Data Type panel” on page 441</td>
<td>Creation option 3</td>
</tr>
<tr>
<td>“Create Function: Option List (1/2) panel” on page 435</td>
<td>Creation option 4</td>
</tr>
<tr>
<td>“Create Function: Option List (2/2) panel” on page 436</td>
<td>Creation option 5</td>
</tr>
</tbody>
</table>

Related tasks
- Chapter 5, “Creating and dropping DB2 objects,” on page 163
- “Creating a function” on page 186

Create Function: Parameter Types panel

You use the Create Function: Parameter Types panel to specify additional type information for the function parameters.
Create Function: Parameter Types panel

Panel and field definitions

Parameter Name
As defined using option 1.

Data Type (Length)
As defined using option 1.

As LOC (Y/N)
For LOB data types, specifies whether a locator for the parameter is passed instead of the value itself:
Y Locator passed
N Value passed

Owner
Defines the parameter as a locator to a transition table.

Like Table/View Name
The owner and name specify a table or view with columns that match those of the transition table.

Parent panels
- "Create Function panel" on page 437

Child panels
None.

Related tasks
- "Creating a function" on page 186

Create Function: Parameters panel

You use the Create Function: Parameters panel to define parameters for the function.
**Panel and field definitions**

<table>
<thead>
<tr>
<th>Process Options Utilities Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2) Create Function Parameters Now 1 to 1 of 1</td>
</tr>
<tr>
<td>C M D Parameter Name + Data Type + Len Pr Sc Data Data Type +</td>
</tr>
</tbody>
</table>

******************************* Bottom of data ************************

**Cmd** This field is used for entering commands.

**Parameter Name**

As defined using option 1.

**Data Type**

This field is used for entering the data type for the column being defined.

**Len**

This field is (optionally) used for entering the length, and optionally the scale of the column. The numeric fields such as INTEGER, REAL, no length specification is required. For character fields, only a single length specification is required. For decimal fields, you can specify a length and a scale.

**DEC**

For a decimal number. The first integer is the precision number. That is, the total number of digits which can range from 1 to 31. The second integer is the scale of the number. That is the number of digits to the right of the decimal point, which can range from 0 to the precision of the number.

**FL**

For a floating-point number. If integer is between 1 and 21 inclusive, the format is single precision floating-point. If the integer is between 22 and 53 inclusive, the format is double precision floating-point.

**Pr Sc**

The scale is the total number of digits in the fractional part of the number and the precision is the total number of digits in the number.

**For Data**

The definitions for character and character large object (CLOB) allow the specification of an optional 'FOR DATA' clause. This field is used to specify the appropriate value.

**User Defined Data Type**

Specifies the data type of the column is a distinct type (a user defined-data type). The length, precision and scale of the column are respectively the length, precision and scale of the source type of the distinct type. The privilege set must implicitly or explicitly include the USAGE privilege on the distinct type.
Create Function: Returned Data Type panel

You use the Create Function: Returned Data Type panel to specify the data type returned by the function.

### Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFF2)</td>
<td>Create Function: Returned Data Type</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Returns:**
- Schema: Specifies the schema name that qualifies a user-defined data type.
- Data Type: Specifies the data type returned by the function.
- Length/Precision: Specifies the length for built-in data types. For the decimal data type this is the precision.
- Scale: For the DECIMAL data type only.
- For Data Encoding Method:
  - SBCS
  - Mixed
  - Bit
  - EBCDIC
  - ASCII
  - UNICODE

**Cast From:**
- Data Type: Specifies the data type that the function is cast from.
- Length/Precision: Specifies the length for cast types.
- Scale: For cast types.
- For Data Encoding Method:
  - SBCS
  - Mixed
  - Bit
  - EBCDIC
  - ASCII
  - UNICODE

**As Locator:**
- Enter /* to select option
- Return a locator to the value (LOB data type only)

Command ===>
- F1=Help
- F2=Split
- F3=Exit
- F4=Expand
- F5=Backward
- F8=Forward
- F9=Swap
- F10=Left
- F11=Right
- F12=Cancel

### Field definitions

**Schema**
- Specifies the schema name that qualifies a user-defined data type.

**Data Type (Returns)**
- Specifies the data type returned by the function.

**Length/Precision (Returns)**
- Specifies the length for built-in data types. For the decimal data type this is the precision.

**Scale (Returns)**
- For the DECIMAL data type only.

**For Data (Returns)**
Create Function: Returned Data Type panel

1. SBCS  
   - Only for built-in data types.
2. Mixed  
   - Only for built-in data types.
3. Bit  
   - Only for built-in data types.

Encoding Method (Returns)
   - Only for built-in data types. Specifies the actual data type returned by the function. See note below.

Data Type (Cast From)
   - Specifies the data type returned by the function.

Length/Precision (Cast From)
   - Specifies the length for built-in data types. For the decimal data type this is the precision.

Scale (Cast From)
   - For the DECIMAL data type only.

For Data (Cast From)
   - 1. SBCS  
     - Only for built-in data types.
   2. Mixed  
     - Only for built-in data types.
   3. Bit  
     - Only for built-in data types.

Encoding Method (Cast From)
   - See note below.

As Locator
   - For LOB data types, specifies that a locator for the parameter is returned instead of the value itself.

Note: If “Cast From” is not specified, then the “Returns” Encoding Method defines the data type returned by the function to the invoking statement. If “Cast From” is specified, then the “Cast From” Encoding Method is the data type returned by the function and the “Returns” Encoding Method is the data type passed to the invoking statement by DB2 (having “cast” it from the “Cast From” data type).

Parent panels
   - “Create Function panel” on page 437

Child panels

None.

Related tasks
   - “Creating a function” on page 186

Create Index panel

You use the Create Index panel to create an index for a pre-existing base table.
Panel and field definitions

Create Index panel

Owner
The owner of the index.

Name
The name of the index. This is a required field.

Table Owner
The SQLID used to create the table on which the index is being defined.

Table Name
The name of the table on which the index is being created. This is a required field. You can enter the name of the table on which the index is created. A validation check needs to be performed to verify that the table exists.

Buffer Pool
The name of the buffer pool in which the index data sets are created.

Creation Options
Specify the number of the creation option you want to use:

1. Index type
To specify information to define the type of index to be created. Displays the Create Index: Type panel.

2. Index columns
To select the columns that comprise the index. Displays the Create Index: Column Selection panel.

3. Allocation
To specify the space requirements for the index being created. Displays the Create Index: Allocation panel.

4. Index options
To specify information to define options for the index to be created. Displays the Create Index: Options panel.

5. Partitions
To specify space information for index partitions. Displays the Create Index: Partitions panel.
Create Index panel

6. **Partition values**
   To specify limit values for index entries in each partition, displays the Create Index: Partition Values panel.

7. **Create the index**
   When you have finished specifying details for the index to be created, select this creation option to create the index.

**Parent panels**
- “DB2 Object Functions panel” on page 510
- “Tables, Views and Aliases panel” on page 704
- “Indexes panel” on page 595
- “Explain Utilities panel” on page 556

**Child panels**

<table>
<thead>
<tr>
<th>To display this panel...</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>“Create Index: Type panel” on page 451</td>
<td>Creation option 1</td>
</tr>
<tr>
<td>“Create Index: Column Selection panel” on page 446</td>
<td>Creation option 2</td>
</tr>
<tr>
<td>“Create Index: Allocation panel”</td>
<td>Creation option 3</td>
</tr>
<tr>
<td>“Create Index: Options panel” on page 447</td>
<td>Creation option 4</td>
</tr>
<tr>
<td>“Create Index: Partitions panel” on page 448</td>
<td>Creation option 5</td>
</tr>
<tr>
<td>“Create Index: Partition Values panel” on page 450</td>
<td>Creation option 6</td>
</tr>
</tbody>
</table>

**Related tasks**
- ”Creating an index” on page 184

Create Index: Allocation panel

You use the Create Index: Allocation panel to specify the space requirements for the index being created.
Panel and field definitions

**VCAT name**
User-defined data sets.

**Storage group name**
DB2-defined data sets (STOGROUP name)

**Primary space**
Specifies the minimum primary space allocation for a managed DB2 data set.

**Secondary space**
Specifies the minimum secondary space allocation for a managed DB2 data set.

**Free pages**
Valid values are 0-255. For a detailed description, see *DB2 for z/OS SQL Reference*.

**Free space**
Valid values are 0-99. For a detailed description, see *File Manager User’s Guide and Reference for IMS Data*.

**Parent panels**
- “Create Index panel” on page 442

**Child panels**
None.

**Related tasks**
- “Creating an index” on page 184

---

Create Index: Allocation panel

**Data Set Parameters:**
- **VCAT name**
  - (for user-defined data sets only)
- **Storage group name**

**Space Requirements for Storage Group:**
- **Primary space**
  - 20 (in kilobytes)
- **Secondary space**
  - 10 (in kilobytes)

**Free Space Options:**
- **Free pages**
  - (0-255, default 0)
- **Free space**
  - (0-99 percent, default 10)

Command ===>

F1=Help F2=Split F3=Exit F4=Expand F7=Backward F8=Forward

F9=Swap F10=Left F11=Right F12=Cancel
Create Index: Column Selection panel

You use the Create Index: Column Selection panel to select the columns that comprise the index.

Panel and field definitions

**Owner**
The name of the owner of the new index.

**Name**
The name of the new index.

**Table owner**
The SQLID used to create the table on which the index is being defined.

**Table name**
The table on which the index is created.

The bottom half of the panel shows a table that can be scrolled in the usual ISPF manner.

**Sel**
An input selection field. This is where you identify the columns that make up the index. You can enter S to select or deselect any column. You can also type a number against a column, to indicate the ordering of the columns in the index.

**Order**
A display-only field. Possible values are spaces, indicating this column is not part of the index; or a number, indicating that this column has been selected as part of the index, and its order among the selected columns.

**(A/D)**
Indicates how the values in the column are to be indexed. If Sel is blank, but A/D is not blank, it is as if Sel contains “S”. The available options are:
- A Ascending. This is the default. If you do not specify a value, this field is filled with an “A” when the panel is redisplayed.
- D Descending

**Column Name**
The name of the column.
Create Index: Options panel

You use the Create Index: Options panel to specify information to define options for the index to be created.

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>Create Index Options</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Data Storage Options:
- Enter '/' to select option
  - Close when not in use
  - Erase data on delete
  - Defer building index
  - Define data set now
  - Allow COPY of index

Group Buffer Pool Usage:
- Write to the group buffer pool
  1. Changed
  2. All
  3. None

You can use the first group of input fields to select the appropriate option with the '/' selection character. The following section describes the options:

Close when not in use
This corresponds to the CLOSE YES/NO clause. It relates to how DB2 handles open data sets when the limit for the maximum number of open data sets is reached.

Erase data on delete
This corresponds to the ERASE YES/NO clause in the 'USING' block. ERASE YES means that the data sets associated with the partition are overwritten with binary zeros when the partition is deleted (dropped).
Create Index: Options panel

**Defer building index**
This corresponds to the DEFER YES/NO clause. The default value is DEFER NO, meaning that the index is built when the CREATE INDEX statement is executed.

**Define data set now**
This corresponds to the DEFINE YES/NO clause. The default value is DEFINE YES, meaning that the data sets used for the index are created when the CREATE INDEX statement is executed.

**Allow COPY of index**
This corresponds to the COPY YES/NO clause. The default value is COPY NO, meaning that it is not possible to use the COPY utility against the index.

The second input field on the panel contains three options relating to the use of the grouped buffer pool. You can select the required option by typing the appropriate number. This corresponds to the GBPCACHE clause in the create statement syntax.

**Parent panels**
- "Create Index panel" on page 442

**Child panels**
None.

**Related tasks**
- "Creating an index" on page 184

Create Index: Partitions panel

You use the Create Index: Partitions panel to specify space information for index partitions.
Panel and field definitions

Owner
The name of the owner of the new index.

Table owner
The SQLID used to create the table on which the index is being defined.

Name
The name of the new index.

Table name
The table on which the index is created.

Cmd
An input selection field. When the panel is first displayed, the table contains 1 row, with default values entered. To define additional partitions type 'I' in the selection field and press Enter. You can also type 'R' to repeat the current definition. Other valid selection commands are i and r followed by an integer.

Parttn. Number
A sequential number showing the number of partitions defined. This is a display only field.

VCAT Name
The VCAT name used to define user-specified data sets. This is an input field. It is mutually exclusive with the STOGROUP parameter.

STOGROUP name
The DB2 storage group name in which the index data set is defined. This is an input field. It is mutually exclusive with the VCAT parameter.

Primary Space
The primary space allocation for the partition, entered in kilobytes.

Secndry Space
The secondary space allocation for the partition, entered in kilobytes.

Erase Data?
A YES/NO value corresponding to the ERASE YES/NO clause.
Create Index: Partitions panel

Free Pages
Free pages results in DB2 leaving an empty page every n pages (n is what is specified).

% Free
The value specified here results in DB2 leaving n% (n is what is specified) of each page free.

Grp BP Caching
This corresponds to the GBPCACHE clause of the create statement (for each partition). These options only apply in a data sharing environment.

Parent panels
- "Create Index panel" on page 442

Child panels
None.

Related tasks
- "Creating an index" on page 184

Create Index: Partition Values panel

You use the Create Index: Partition Values panel to specify limit values for index entries in each partition.

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>Create Index: Partition Values</td>
<td>Row 1 to 6 of 6</td>
<td></td>
</tr>
</tbody>
</table>

New Index:
- Owner
- Name
- Table owner
- Table name

Parttn. Number Column Name Value
1 EMPNO FIRSTNAME MIDINIT
2 EMPNO FIRSTNAME MIDINIT

Owner
The name of the owner of the new index.

Name
The name of the new index.

Table owner
The SQLID used to create the table on which the index is being defined.
Create Index: Type panel

You use the Create Index: Type panel to specify information to define the type of index to be created.

Panel and field definitions

<table>
<thead>
<tr>
<th>Field</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Process Options Utilities Help</strong></td>
<td></td>
</tr>
<tr>
<td><strong>FM/DB2 (DFG2)</strong> Create Index Type</td>
<td></td>
</tr>
<tr>
<td>Index Type Options:</td>
<td></td>
</tr>
<tr>
<td>Uniqueness</td>
<td></td>
</tr>
<tr>
<td>1. Non-Unique</td>
<td></td>
</tr>
<tr>
<td>2. Unique</td>
<td></td>
</tr>
<tr>
<td>3. Unique Not Null</td>
<td></td>
</tr>
<tr>
<td>Clustering Index Option:</td>
<td></td>
</tr>
<tr>
<td>Enter ‘/’ to select option</td>
<td></td>
</tr>
<tr>
<td>/ Clustering Index (required for partitioned indexes)</td>
<td></td>
</tr>
<tr>
<td>Non-Partitioned Index:</td>
<td></td>
</tr>
<tr>
<td>Piecesize . . . Multiplier</td>
<td></td>
</tr>
<tr>
<td>1. K (1024)</td>
<td></td>
</tr>
<tr>
<td>2. M (1048576)</td>
<td></td>
</tr>
<tr>
<td>3. G (1073741824)</td>
<td></td>
</tr>
</tbody>
</table>

1. Non-Unique
   - Duplicate index values are permitted.

2. Unique
   - Distinct values for all entries in the index.
Create Index: Type panel

3. Unique Not Null
   Distinct values for all not null entries in the index.

Clustering Index Option
   Enter a "/" to specify a clustering index (CLUSTER keyword).

Piecesize
   You can specify a maximum addressability for each piece (data set of a
   non-partitioning index). The values entered must be powers of 2, and the
   valid values depend on the multiplier.
   • If the multiplier is K, valid values are 2**n between 256 and 67 108 864
     inclusive.
   • If the multiplier is M, valid values are 2**n between 1 and 66 536
     inclusive.
   • If the multiplier is G, valid values are 2**n between 1 and 64 inclusive.

Parent panels
   • “Create Index panel” on page 442

Child panels

None.

Related tasks
   • “Creating an index” on page 184

Create Procedure panel

You use the Create Procedure panel to create a new procedure.

Panel and field definitions

<table>
<thead>
<tr>
<th>Process Options Utilities Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2) Create Procedure</td>
</tr>
</tbody>
</table>

Stored Procedure:
- Schema . . . + (optional)
- Name . . . . . . . . . . . . . . . . . . . . . . . . . . . . . (optional)
- Result Sets . . . . (optional)

Stored Procedure Creation:
- Creation Options
  - 1. Parameters (optional, usually required)
  - 2. Parameter Types (optional, define parameters first)
  - 3. Option List (required)
  - 4. Option List (More) (optional)
  - 5. Create the Procedure

Command ===>
F1=Help F2=Split F3=Exit F4=Expand F7=Backward F8=Forward
F9=Swap F10=Left F11=Right F12=Cancel

Schema
   The name of the schema in which the procedure is defined.

Name
   The name of the stored procedure.
Create Procedure panel

Result Sets
Specifies the maximum number of dynamic result sets that can be returned by the stored procedure.

Creation Options
Specify the number of the creation option you want to use:

1. Parameters
   To specify information to define the parameters for a procedure to be created. Displays the Create Procedure: Parameters panel.

2. Parameter Types
   To specify information to define the parameter types in relation to tables or views, and to specify the parameter type. Displays the Create Procedure: Parameter Types panel.

3. Option List
   To specify information to define options for the procedure to be created. Displays the Create Procedure: Option List (1/2) panel.

4. Option List (More)
   To further specify information to define options for the procedure to be created. Displays the Create Procedure: Option List (2/2) panel.

5. Create the Procedure
   When you have finished specifying details for the stored procedure to be created, select this creation option to create the stored procedure.

Parent panels
- “DB2 Object Functions panel” on page 510
- “Stored Procedures panel” on page 685

Child panels

To display this panel... | Use/do this
---|---
“Create Procedure: Parameters panel” on page 457 | Creation option 1
“Create Procedure: Parameter Types panel” on page 456 | Creation option 2
“Create Procedure: Option List (1/2) panel” | Creation option 3
“Create Procedure: Option List (2/2) panel” on page 455 | Creation option 4

Related tasks

Create Procedure: Option List (1/2) panel

You use the Create Procedure: Option List (1/2) panel to specify information to define options for the procedure to be created.
Panel and field definitions

External Name
Name of the MVS load module that contains the user-written code to implement the procedure.

Package Collection
Specifies the package collection to be used when the procedure is run. If not specified, the invoking program’s package collection is used.

Workload Manager Environment
Specifies the MVS workload manager environment in which the procedure is to run.

Time Limit (CPU units)
Limit on processor service units used by a single invocation of the procedure.

Language
Application programming language of the procedure.

SQL
Specifies what type of SQL statements the procedure contains.

Security
Specifies the interaction of the procedure with an external security product.

Parameter Style
1. DB2SQL
   Additional parameters containing DB2 information are passed to the procedure.

2. General
   Only the parameters on the call statement are passed to the procedure and they cannot be null.

3. General with nulls
   A vector of null indicators is passed in addition to the parameters on the call statement.
Create Procedure: Option List (1/2) panel

Parent panels
• “Create Procedure panel” on page 452

Child panels
None.

Related tasks
• “Creating a procedure” on page 187

Create Procedure: Option List (2/2) panel

You use the Create Procedure: Option List (2/2) panel to further specify information to define options for the procedure to be created.

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>Create Procedure Option List (2/2)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Enter ‘/’ to select option
- Main Program
- Stay Resident
- Deterministic
- DBINFO argument passed
- Commit on return

Command ===> Scroll PAGE
F1=Help    F2=Split   F3=Exit    F4=Retrieve   F7=Backward   F8=Forward
F9=Swap    F10=Actions F12=Cancel

Main Program
If checked, the program runs as a main routine, otherwise it runs as a subroutine.

Stay Resident
If checked, the function load module remains in memory when the function ends.

Deterministic
If checked, indicates the function always returns the same result for the same set of input values.

DBINFO argument passed
If checked, indicates that additional DB2 information is passed when the function is called.

Commit on return
If checked, indicates that DB2 issues a commit when the stored procedure returns.
Create Procedure: Parameter Types panel

You use the Create Procedure: Parameter Types panel to specify information to define the parameter types in relation to tables or views, and to specify the parameter type.

Panel and field definitions

- **Parameter Name**
  - The name of the parameter.

- **Data Type(Length)**
  - The data type of the parameter.

- **As LOC (Y/N)**
  - For LOB data types, specifies that a locator for the parameter is passed instead of the value itself.

- **Owner**
  - Defines the parameter as a locator to a transition table.

- **Like Table/View Name**
  - The owner and name specify a table or view with columns that match those of the transition table.
Specifies whether the parameter can be used for input only, output only, or both input and output. Valid values are:

IN      Identifies the parameter as an input parameter.
OUT     Identifies the parameter as an output parameter.
INOUT   Identifies the parameter as both an input and output parameter.

Parent panels

- "Create Procedure panel" on page 452

Child panels

None.

Related tasks

- "Creating a procedure" on page 187

Create Procedure: Parameters panel

You use the Create Procedure: Parameters panel to specify information to define the parameters for a procedure to be created.

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFUZ) Create Procedure: Parameters Row 1 to 1 of 1</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<---(For Built in Data Types)----->

<DEC> For User Defined

<table>
<thead>
<tr>
<th>Cmd</th>
<th>Parameter Name</th>
<th>Data Type</th>
<th>Len</th>
<th>Pr Sc Data</th>
<th>Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>PARAM1</td>
<td>CHAR</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

****************************************************************************** Bottom of data ****************************************************************************

Command ===> Scroll PAGE

F1=Help F2=Split F3=Exit F4=Expand F5=Backward F6=Forward

F9=Swap F10=Left F11=Right F12=Cancel

Cmd     This field is used for entering commands.

Parameter Name

As defined using option 1.

Data Type

This field is used for entering the data type for the column being defined.
Create Procedure: Parameters panel

**Len**
This field is (optionally) used for entering the length, and optionally the scale of the column. The numeric fields such as INTEGER, REAL, no length specification is required. For character fields, only a single length specification is required. For decimal fields, you can specify a length and a scale.

**Dec**
For a decimal number. The first integer is the precision number. That is, the total number of digits which can range from 1 to 31. The second integer is the scale of the number. That is the number of digits to the right of the decimal point, which can range from 0 to the precision of the number.

**FL**
For a floating-point number. If integer is between 1 and 21 inclusive, the format is single precision floating-point. If the integer is between 22 and 53 inclusive, the format is double precision floating-point.

**Pr Sc**
The scale is the total number of digits in the fractional part of the number and the precision is the total number of digits in the number.

**For Data**
The definitions for character and character large object (CLOB) allow the specification of an optional 'FOR DATA' clause. This field is used to specify the appropriate value.

**User Defined Data Type**
Specifies the data type of the column is a distinct type (a user defined-data type). The length, precision and scale of the column are respectively the length, precision and scale of the source type of the distinct type. The privilege set must implicitly or explicitly include the USAGE privilege on the distinct type.

**Parent panels**
- "Create Procedure panel" on page 452

**Child panels**
None.

**Related tasks**
- "Creating a procedure" on page 187

Create Synonym panel

You use the Create Synonym panel to create a new synonym.
Create Synonym panel

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>Create Synonym</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Synonym Details:
Name ........ DBNAME +

Table or View Details:
Owner ........ +
Name ........ TABLE1 +

Command ===>
F1=Help    F2=Split   F3=Exit    F4=CRetrieve F7=Backward F8=Forward
F9=Swap    F10=Actions F12=Cancel

Name (Synonym Details)
The name of the synonym that is being created.

Owner
The owner of the table.

Name (Table or View Details)
The name of the table.

Parent panels
- "DB2 Object Functions panel" on page 510
- "Synonyms panel" on page 687

Child panels
None.

Related tasks
- "Creating a synonym" on page 185

Create Table panel

You use the Create Table panel to create a new table.
Create Table panel

Panel and field definitions

```markdown
<table>
<thead>
<tr>
<th>Process Options Utilities Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2) Create Table</td>
</tr>
</tbody>
</table>

**New Table:**
- **Owner**: + (optional)
- **Name**: TABLE1 +
- **Database**: (optional)
- **Table Space**: (optional)

**Model Table/View**: (optional)
- **Owner**: +
- **Name**: +

**Usage:**
1. Generate LIKE clause
2. Load table information

**Table Creation:**
- **Creation Options**: (Use option 1 before options 2, 4-9)
  1. **Columns**: (normally required, invalid with model LIKE)
  2. **Nulls/default values**: (optional)
  3. **Table options**: (editproc,,validproc,,more; optional)
  4. **Unique Constraints**: (primary/unique keys; optional)
  5. **Referential Constraints**: (foreign keys; optional)
  6. **Check constraints**: (optional)
  7. **Procedure exits**: (optional)
  8. **Generate values**: (optional)
  9. **Create the table**

**Command ====>**
- **F1=Help**
- **F2=Split**
- **F3=Exit**
- **F4=Expand**
- **F7=Backward**
- **F8=Forward**
- **F9=Swap**
- **F10=Left**
- **F11=Right**
- **F12=Cancel**

**Owner (New Table)**
The name of the owner of the new table.

**Name (New Table)**
The name of the new table.

**Database**
The name of the database in which the new table is to be created.

**Table Space**
The name of the table space in which the new table is to be created.

**Owner (Model Table/View)**
The name of the owner of the model table.

**Name (Model Table/View)**
The name of the model table.

**Creation Options**
Specify the number of the creation option you want to use:

1. **Columns**
   - To specify information to define the columns for a table to be created. Displays the Create Table: Columns panel.

2. **Nulls/default values**
   - To specify information to define the null attributes and default values for the columns of the table. Displays the Create Table: Nulls and Default Values panel.
```
3. Table options
To specify information to define options for the table you are creating. Displays the Create Table: Options panel.

4. Unique Constraints
To specify information to define a primary key and up to three additional unique keys for the table you are creating. Displays the Create Table: Unique Constraints panel.

5. Referential Constraints
To specify information to define one or more column referential constraints for the table. Displays the Create Table: Column Referential Constraints panel.

6. Check Constraints
To specify information to define column check constraints for the table. Displays either the Create Table: Column Check Constraints panel or the Create Table: Table Check Constraints panel.

7. Procedure exits
To specify information to define procedure exits for the table you are creating. Displays the Create Table: Procedure Exits panel.

8. Generate values
To specify information to define generated values for identity or ROWID columns in the table. Displays the Create Table: Generate Values panel.

9. Create the table
When you have finished specifying details for the table to be created, select this creation option to create the table.

Parent panels
- “DB2 Object Functions panel” on page 510
- “Table Spaces panel” on page 698
- “Tables, Views and Aliases panel” on page 704

Child panels

<table>
<thead>
<tr>
<th>To display this panel...</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>“Create Table: Columns panel” on page 474</td>
<td>Creation option 1</td>
</tr>
<tr>
<td>“Create Table: Nulls and Default Values panel” on page 479</td>
<td>Creation option 2</td>
</tr>
<tr>
<td>“Create Table: Options panel” on page 481</td>
<td>Creation option 3</td>
</tr>
<tr>
<td>“Create Table: Unique Constraints panel” on page 487</td>
<td>Creation option 4</td>
</tr>
<tr>
<td>“Create Table: Column Referential Constraints panel” on page 473</td>
<td>Creation option 5</td>
</tr>
<tr>
<td>“Create Table: Column Check Constraints panel” on page 471</td>
<td>Creation option 6</td>
</tr>
<tr>
<td>“Create Table: Procedure Exits panel” on page 483</td>
<td>Creation option 7</td>
</tr>
<tr>
<td>“Create Table: Generate Values panel” on page 476</td>
<td>Creation option 8</td>
</tr>
<tr>
<td>“Create Table: Model Load panel” on page 477</td>
<td>See note below 2</td>
</tr>
<tr>
<td>“Create Table: Columns panel” on page 474</td>
<td>See note below 1</td>
</tr>
</tbody>
</table>
Create Table panel

To display this panel... Use/do this

Notes:
1. When you select Table Creation option 5 (Referential Constraints) on the main Create Table panel, FM/DB2 displays either the Column Referential Constraints panel ("Create Table: Column Referential Constraints panel" on page 473) or the Table Referential Constraints panel ("Create Table: Table Referential Constraint panel" on page 485). To change from one panel to the other, press the F5 function key.
2. The Create Table Model Load panel is displayed when you:
   • Initially specify, or subsequently change, the Model Table/View details,
   • Select Usage option 2 (Load table information), and
   • Press Enter.

Related tasks
• "Creating a table" on page 176

Create Table Space panel

You use the Create Table Space panel to create a new table space.

Panel and field definitions

**Process Options Utilities Help**

FM/DB2 (DFG2) Create Table Space

Table Space Definition:
Name ......... (required)
Database .......... (optional)
Buffer pool ........ (optional: BPn,BP8Kn,BP16Kn,BP32Kn)

Table Space Creation:
Creation Options
1. Table space type (optional, usually needed)
2. Table space allocation (optional, usually needed)
3. Data storage options (optional)
4. Locking options (optional)
5. Data sharing options (optional)
6. Partitions (partitioned table spaces only)
7. Create the table space

Command ===>
F1=Help F2=Split F3=Exit F4=CREtiev F7=Backward F8=Forward
F9=Swap F10=Actions F12=Cancel

Name The name of the table space you want to create.
Database The name of the database in which the table space is to be created.
Buffer pool The name of the buffer pool to be used for the table space.
Creation Options Specify the number of the creation option you want to use:

1. Table space type
   To specify information to define the type of table space to be created. Displays the Create Table Space: Type panel.
Create Table Space panel

2. Table space allocation
   To specify information to define the data set and space allocation for the table space to be created. Displays the Create Table Space: Allocation panel.

3. Data storage options
   To specify information to define data storage options for the table space to be created. Displays the Create Table Space: Data Storage Options panel.

4. Locking options
   To specify information to define the locking options for the table space to be created. Displays the Create Table Space: Locking Options panel.

5. Data sharing options
   To specify information to define the data sharing options for the table space to be created. Displays the Create Table Space: Data Sharing Options panel.

6. Partitions
   To specify information to define the partitions for a partitioned table space to be created. Displays the Create Table Space: Define Partitions panel.

7. Create the table space
   When you have finished specifying details for the table space to be created, select this creation option to create the table space.

Parent panels

- “DB2 Object Functions panel” on page 510
- “Databases panel” on page 500
- “Table Spaces panel” on page 698

Child panels

To display this panel... Use/do this
---
“Create Table Space: Type panel” on page 470 Creation option 1
“Create Table Space: Allocation panel” Creation option 2
“Create Table Space: Data Storage Options panel” on page 466 Creation option 3
“Create Table Space: Locking Options panel” on page 469 Creation option 4
“Create Table Space: Data Sharing Options panel” on page 465 Creation option 5
“Create Table Space: Define Partitions panel” on page 467 Creation option 6

Related tasks

- Chapter 5, “Creating and dropping DB2 objects,” on page 163
- ”Creating a table space” on page 174

Create Table Space: Allocation panel

You use the Create Table Space: Allocation panel to specify information to define the data set and space allocation for the table space to be created.
Panel and field definitions

VCAT name
Specifies the first data set for the table space is managed by the user.

Storage group name
Specifies DB2 defines and manages the data sets for the table space.

Primary space
Specifies the minimum primary space allocation for a DB2 managed data set.

Secondary space
Specifies the minimum secondary space allocation for a DB2 managed data set.

Free pages
Specifies how often to leave a page of free space when the table space or partition is loaded or reorganized.

Free space
The total amount of unused space in a page.

Maximum rows per page
Specifies the maximum number of rows that DB2 considers placing on each data page.

Parent panels
- “Create Table Space panel” on page 462

Child panels
None.

Related tasks
- “Creating a table space” on page 174
You use the Create Table Space: Data Sharing Options panel to specify information to define the data sharing options for the table space to be created.

Panel and field definitions

You can select the following option by entering a “/” in the selection field at the top of the panel:

Selective Partition Locking
Enter ‘/’ to select option
  - Use selective partition locking

Group Buffer Pool Usage:
  Write to the Group Buffer Pool
  1. Changed
  2. All
  3. System
  4. None

You can select the following option by entering a “/” in the selection field at the top of the panel:

Selective Partition Locking
See DB2 for z/OS SQL Reference for a detailed explanation of selective partition locking, and the software prerequisites.

Group Buffer Pool Usage

1. Changed
   When there is inter-DB2 R/W interest on the table space or partition updated pages are written to the group buffer pool. When there is no inter-DB2 R/W interest, the group buffer pool is not used.

2. All
   Indicates that pages are to be stored in a cache in the group buffer pool as they are read in from DASD.

3. System
   Indicates that only changed system pages within the LOB table space are to be stored in a cache in the group buffer pool.

4. None
   Indicates that no pages are to be stored in a cache in the group buffer pool.

Parent panels

- “Create Table Space panel” on page 462

Child panels

None.
Create Table Space: Data Storage Options panel

Related tasks
- “Creating a table space” on page 174

Create Table Space: Data Storage Options panel

You use the Create Table Space: Data Storage Options panel to specify information to define data storage options for the table space to be created.

Panel and field definitions

Use data compression
This corresponds to the COMPRESS YES/NO clause. COMPRESS YES means that DB2 stores the data in compressed format.

Close when not in use
This corresponds to the CLOSE YES/NO clause. It relates to how DB2 handles open data sets when the limit for the maximum number of open data sets is reached.

Erase data on delete
This corresponds to the ERASE YES/NO clause of the 'USING' block (see DB2 for z/OS SQL Reference). ERASE YES means that DB2 data sets are overwritten with binary zeros when they are dropped (deleted).

Define dataset now
This corresponds to the DEFINE YES/NO clause. DEFINE YES means that DB2 defines the data sets for the table space when the create statement is executed. DEFINE NO means creation is delayed until data is placed in the table space.

INSERT uses clustering index
This corresponds to the MEMBER CLUSTER clause.

Track modified pages
This corresponds to the TRACKMOD YES/NO clause. It relates to a feature where changes to data pages are recorded in the table space map pages. The default setting is TRACKMOD YES. There are restrictions on this clause when LOB is specified.
Create Table Space: Data Storage Options panel

Log changes to LOB columns
This corresponds to the LOG YES/NO clause. LOG YES can only be specified for a LOB table space.

Data encoding
Covers the encoding method option. For OS/390® systems, the default format is EBCDIC.

- **EBCDIC** specifies the encoding scheme.
- **ASCII** specifies the encoding scheme.
- **UNICODE** specifies the encoding scheme.

Parent panels
- "Create Table Space panel" on page 462

Child panels
None.

Related tasks
- "Creating a table space" on page 174

Create Table Space: Define Partitions panel

You use the Create Table Space: Define Partitions panel to specify information to define the partitions for a partitioned table space to be created.

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>Create Table Space: Define Partitions</td>
<td>Row 1 to 1 of 1</td>
<td></td>
</tr>
</tbody>
</table>

New Table Space:
- Name ........
- Database ....

<table>
<thead>
<tr>
<th>Pt.</th>
<th>VCAT</th>
<th>STOGROUP</th>
<th>Primary Secndry Erase Free % Data BP Track</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cmd No.</td>
<td>Name</td>
<td>Name + Space</td>
<td>Space</td>
</tr>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Command ===> Scroll PAGE
F1=Help F2=Split F3=Exit F4=Expand F7=Backward F8=Forward
F9=Swap F10=Left F11=Right F12=Cancel

Cmd  Use the following line commands to manipulate the panel rows containing partition information:
Create Table Space: Define Partitions panel

<table>
<thead>
<tr>
<th></th>
<th>Insert one new row.</th>
</tr>
</thead>
<tbody>
<tr>
<td>In</td>
<td>Insert $n$ new rows.</td>
</tr>
<tr>
<td>R</td>
<td>Repeat the current row.</td>
</tr>
<tr>
<td>Rn</td>
<td>Repeat the current row $n$ times.</td>
</tr>
<tr>
<td>D</td>
<td>Delete one row.</td>
</tr>
<tr>
<td>Dn</td>
<td>Delete $n$ rows.</td>
</tr>
</tbody>
</table>

Pt. No.
Partition number

VCAT Name
This is used to specify user-defined data sets. It is mutually exclusive with a STOGROUP name.

STOGROUP Name
This is used to specify a (DB2) storage group name. Specifying a storage group name means that DB2 creates the data sets for the partition. It is mutually exclusive with a VCAT name.

Primary Space
This is used to specify the primary space allocation for the partition (in kilobytes).

Secondary Space
This is used to specify the secondary space allocation for the partition (in kilobytes).

Erase Data
This corresponds to the ERASE YES/NO clause in the 'USING' block. ERASE YES means that the data sets associated with the partition are overwritten with binary zeros when the partition is deleted (dropped).

Free Pages
Free pages results in DB2 leaving an empty page every $n$ pages ($n$ is what is specified).

% Free
The value specified here results in DB2 leaving $n\%$ ($n$ is what is specified) of each page free.

Use Data Cmpr?
This corresponds to the COMPRESS NO/YES clause. YES means the data is stored in compressed format.

Group BP Caching
This corresponds to the GBPCACHE clause of the create statement (for each partition). This option only applies in a data sharing environment.

TRACK MOD
Specifies whether DB2 tracks modified pages in the space map pages of the table space or partition. Do not specify TRACKMOD for a LOB table space. For a table space in a TEMP database, DB2 uses TRACKMOD NO regardless of the value specified.

Parent panels
- “Create Table Space panel” on page 462
Create Table Space: Define Partitions panel

Child panels

None.

Related tasks

- "Creating a table space" on page 174

Create Table Space: Locking Options panel

You use the Create Table Space: Locking Options panel to specify information to define the locking options for the table space to be created.

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
<th>Locking Options:</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFQ2)</td>
<td>Create Table Space: Locking Options</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Locksize parameter**

Specify the LOCKSIZE parameter:

1. **ANY**
   
   Specifies DB2 can use any lock size.

2. **Table space**
   
   Specifies DB2 can use table space lock.

3. **Table**
   
   Specifies DB2 can use table locks.

4. **Page**
   
   Specifies DB2 can use page locks.

5. **Row**
   
   Specifies DB2 can use row locks.

6. **LOB**
   
   Specifies DB2 can use LOB locks.

**Maximum Locks**

The maximum number of page, row, or LOB locks an application process can hold simultaneously in the table space. Valid values are: 0–2147483647 or “SYSTEM”. 

Chapter 15. FM/DB2 panels and fields 469
Create Table Space: Locking Options panel

Parent panels
- "Create Table Space panel" on page 462

Child panels
None.

Related tasks
- “Creating a table space” on page 174

Create Table Space: Type panel

You use the Create Table Space: Type panel to specify information to define the type of table space to be created.

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>F4=Retrieve</td>
<td>F7=Backward</td>
<td>F8=Forward</td>
<td>F9=Swap</td>
</tr>
<tr>
<td>F1=Help</td>
<td>F2=Split</td>
<td>F3=Exit</td>
<td>F10=Actions</td>
</tr>
</tbody>
</table>

Table Space Type Option

Specify the type option:

1. Normal
   Identifies the table space is a normal (non-LOB) table space.

2. Large
   Identifies that each partition of a partitioned table space has a maximum partition size of 4 GB. This relates to a deprecated SQL option, therefore it is preferable to use Max. Partition Size. If Large is specified, then No. of partitions must also be specified.

3. LOB (large object)
   Identifies the table space as LOB table space. If LOB is specified, neither Segment size or No. of partitions can be specified.

<table>
<thead>
<tr>
<th>Table Space Type Option</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Normal (default)</td>
</tr>
<tr>
<td>2. Large (not valid if a partition size is specified)</td>
</tr>
<tr>
<td>3. LOB (large object table space)</td>
</tr>
</tbody>
</table>

Select either non-partitioned or partitioned below

Non-Partitioned Table Space:
   Segment size ........ KB (4,8,...,64 for segmented table space)

Partitioned Table Space:
   No. of partitions ... (1-254 for partitioned table space)
   Max. partition size .. GB (blank,1,2,4,8,16,32,64)
LOB means Large Object. A LOB table space is used to store data that does not fit within a single table space page. Examples of LOBs might include such things as photographs, video files, or sound files.

**Segment size**
Indicates that the table space is separated and specifies how many pages are in each segment. This cannot be specified with Large, LOB or either of the partitioned table space options.

**No. of partitions**
Indicates that the table space is partitioned and specifies the number of partitions. This is required if Large is specified or if Max. partition size is specified with LOB or segment size.

**Max. partition size**
Indicates that the maximum size of each partition or for LOB table spaces, the maximum size of each data set. This requires the specification of LOB or No. of partitions. This cannot be specified with Large or Segment size.

**Parent panels**
- "Create Table Space panel" on page 462

**Child panels**
None.

**Related tasks**
- "Creating a table space” on page 174

---

**Create Table: Column Check Constraints panel**

You use the Create Table: Column Check Constraints panel to specify information to define column check constraints for the table.
Panel and field definitions

Sel  Type S in this field to display a scrollable panel that uses to specify a check condition up to the maximum length permitted by DB2. See the description for the Check Condition entry field.

Column Name  The name of the column in the table you are creating.

Data Type(Length)  The data type and length of the column in the table you are creating.

Constraint Name  The name of the constraint. This is an optional field.

Check Condition  SQL expression defining the column check constraint.

If you need to specify a check condition that is too long to fit in the entry field on the panel, enter S in the Sel field to display a panel. If a long check condition already exists and you alter the part of the check condition displayed on the Column Check Constraints panel, FM/DB2 displays the same panel that you can use to confirm the changes you have made.

Note: FM/DB2 does not validate the check condition you specify.

Parent panels
  • "Create Table panel" on page 459

Child panels
None.

Related tasks
  • "Creating a table" on page 176
Create Table: Column Referential Constraints panel

You use the Create Table: Column Referential Constraints panel to specify information to define one or more column referential constraints for the table.

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>Create Table: Column Referential Constraints</td>
<td>Row 1 to 1 of 1</td>
<td></td>
</tr>
</tbody>
</table>

Owner . . . . . . +
Name . . . . . . TAB1 +
Database . . . .
Table Space . .

Constnt <-------- Parent Table --------------- ON
columns + Name + Owner + Name + Column + DLT
COL1

******************************* Bottom of data *******************************

Command ===> Scroll PAGE
F1=Help F2=Split F3=Exit F4=Expand F5=Tabcons F7=Backward
F8=Forward F9=Swap F10=Left F11=Right F12=Cancel

Column Name
The foreign key. This is the column in the table being created that is to be related to an equivalent column (the parent key) in the parent table.

Constnt Name
Optionally use this field to specify the name of the constraint.

Owner
Optionally use this field to qualify the name of the parent table.

Name
The name of the table containing the parent key. This field is mandatory.

Column
The parent key. Specify the name of the column in the parent table to which you want the foreign key (in Column Name) related. The parent key must be a primary or unique key on the parent table. If you leave the Parent Table Column field blank, the foreign key is related to the primary key of the parent table.

ON DLT
Determines the type of ON DELETE clause, if any, that FM/DB2 generates in the CREATE TABLE statement. Valid values are:
(blank)
No ON DELETE clause generated
R ON DELETE RESTRICT
N ON DELETE NO ACTION
C ON DELETE CASCADE
S ON DELETE SET NULL
Create Table: Column Referential Constraints panel

Parent panels

- "Create Table panel" on page 459

Child panels

None.

Related tasks

- "Creating a table" on page 176

Create Table: Columns panel

You use the Create Table: Columns panel to specify information to define the columns for a table to be created.

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>Create Table: Columns</td>
<td>Row 1 to 1 of 1</td>
<td></td>
</tr>
</tbody>
</table>

Owner ....... +
Name ....... TAB1 +
Database .......
Table Space ....

<---(For Built in Data Types)----->
<DEC>

Cmd Column Name + Data Type Len Pr Sc Data Data Type +

COL1 CHAR 3

******************************* Bottom of data ********************************

Command ===> Scroll PAGE
F1=Help F2=Split F3=Exit F4=Expand F7=Backward F8=Forward
F9=Swap F10=Left F11=Right F12=Cancel

Cmd Line command area. Use the following line commands to manipulate the panel rows containing column information:

- **In** Insert $n$ new rows.
- **Rn** Repeat the current row $n$ times. (See following note.)
- **Dn** Delete $n$ rows.

Note: When you repeat a row, FM/DB2 repeats only the column information shown on this panel and the Nulls and Default Values panel. All other information is cleared.

Column Name

The name of the column.

Data Type

The data type of the column.
Create Table: Columns panel

**Len**  
The length of the column being defined. Only use this field for columns with a string data type.

**Pr, Sc**  
The precision and scale for numeric data types.

For a decimal number:
- Specify the precision (1–31) in the **Pr** field and the scale (0 to the precision of the number) in the **Sc** field.

For a single-precision floating point number:
- If you have specified a data type of FLOAT, specify the precision (1–21) in the **Pr** field and leave the **Sc** field blank.
- If you have specified a data type of REAL, leave the **Pr** and **Sc** fields blank.

For a double-precision floating point number:
- If you have specified a data type of FLOAT, specify the precision (22–53) in the **Pr** field or leave the **Pr** field blank, and leave the **Sc** field blank.
- If you have specified a data type of DOUBLE, leave the **Pr** and **Sc** fields blank.

For a binary integer, leave the **Pr** and **Sc** fields blank.

**For Data**  
The subtype to be used in the FOR subtype DATA clause. Only specify a non-blank value for character string columns with a data type of CHAR, VARCHAR, LONG VARCHAR, or CLOB. Valid values are:

- **(Blank)**  
  FOR DATA clause not generated.

- **SBCS (or S)**  
  FOR SBCS DATA clause generated.

- **MIXED (or M)**  
  FOR MIXED DATA clause generated.

- **BIT (or B)**  
  FOR BIT DATA clause generated. Do not specify BIT for a CLOB column.

**User-defined Data Type**  
Specifies the data type of the column is a distinct type (a user defined-data type). The length, precision and scale of the column (if applicable) are respectively the length, precision and scale of the source type of the distinct type.

**Parent panels**
- “Create Table panel” on page 459

**Child panels**

None.

**Related tasks**
- “Creating a table” on page 176

**Related references**
- Chapter 14, “DB2 data types,” on page 361
- “Create Table: Nulls and Default Values panel” on page 479
Create Table: Generate Values panel

You use the Create Table: Generate Values panel to specify information to define generated values for identity or ROWID columns in the table.

**Panel and field definitions**

<table>
<thead>
<tr>
<th>Process Options Utilities Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2) Create Table: Generate Values Row 1 to 1 of 1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Owner</th>
<th>Name</th>
<th>Database</th>
<th>Table Space</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>TAB1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Gen</th>
<th>As Caching</th>
<th>(A/D) Idty</th>
<th>Start Incr</th>
<th>Option</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Data Type (Length)</th>
<th>(Y/N)</th>
<th>value</th>
<th>(0=NO CACHE)</th>
</tr>
</thead>
<tbody>
<tr>
<td>COL1</td>
<td>CHAR(3)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

******************************* Bottom of Data *******************************

Command ===> Scroll PAGE

F1=Help F2=Split F3=Exit F4=Expand F7=Backward F8=Forward
F9=Swap F10=Left F11=Right F12=Cancel

**Column Name**

The name of the column in the table you are creating for which you want DB2 to create values.

**Gen (A/D)**

Determines when DB2 generates values for the column. Valid values are:

*(blank)*

Values not generated for this column.

A

Values always generated for this column when a new row is inserted into the table.

D

Values generated for this column when a new row is inserted unless a value is specified.

**As Idty (Y/N)**

Identifies whether the column is the identity column for the table (there can be only one).

N (or blank)

Column is not an identity column.

Y

Column is the identity column for the table.

**Start value**

Starting value for generating identity column values (integer).

**Incr value**

Interval between consecutive generated identity column values (non-zero integer).
Create Table: Generate Values panel

Caching Option
An integer specifying how many preallocated values of the identity column are kept in memory to improve performance. Valid values:

0 Generates a NO CACHE clause.
n Generates a NO CACHE nclause. n must be 2 or more.

Parent panels
• "Create Table panel" on page 459

Child panels
None.

Related tasks
• "Creating a table" on page 176

Create Table: Model Load panel

You use the Create Table: Model Load panel to control what information about a model table is loaded into the ISPF variables and tables used to generate the CREATE TABLE statement. The panel also allows you to specify whether the load is a refresh of the current information, or adds to it.

Panel and field definitions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Owner</td>
<td>The name of the owner of the model table.</td>
</tr>
<tr>
<td>Name</td>
<td>The name of the model table.</td>
</tr>
</tbody>
</table>

Press enter to load the selected information for the model table/view, or press PF3 (End) or PF12 (Cancel) to cancel the load.

Model Table/View:
Owner . . . FMNUSER +
Name . . . MODEL2 +

Model Load Options:
Enter '/' to load
/ Basic table information / Unique key information / Refresh/Add
/ Column information / Referential constraints 1 1. Refresh
/ Primary key information / Check constraints 2 2. Add

Load single-column referential constraints as
1 1. Column referential constraints
2 2. Table referential constraints

Command ==>
F1=Help  F2=Split  F3=Exit  F4=Expand  F7=Backward  F8=Forward  F9=Swap  F10=Left  F11=Right  F12=Cancel
Create Table: Model Load panel

**Basic table information**
Database name, table space name if explicitly defined for the model table, and all the table options values for the model table or view. (See following Note.)

**Column information**
Column names and data types, and nulls and default values specifications for the model table or view. (See following Note.)

**Primary key information**
Column order in the primary key of the model table.

**Unique key information**
Column order for the first three unique keys in the model table.

**Referential constraints**
Referential constraints for the model table.

**Check constraints**
Check constraints for the model table.

1. **(Refresh)**
Discard all current CREATE TABLE information, whether manually entered or from a previous load, and load the new selected information for the model table or view.

2. **(Add)**
Add the selected type of information for the model table or view to the current information for the new table.

**Note:** If you select the Add option, you can only select **Basic table information** and **Column information**. FM/DB2 only loads basic table information if the target field is blank.

**Load single-column referential constraints as**
If the model table has referential constraints, whether you want single-column referential constraints loaded as **column** referential constraints or **table** referential constraints.

1. **Column referential constraints**
Single-column referential constraints associated with the foreign key column.

2. **Table referential constraints**
Single-column referential constraints not associated with the foreign key column. (The associated constraint names are kept regardless of the level of the currently connected DB2 system.)

**Parent panels**
- "Create Table panel" on page 459

**Child panels**
None.

**Related tasks**
- "Creating a table" on page 176
Create Table: Nulls and Default Values panel

You use the Create Table: Nulls and Default Values panel to specify information to define the null attributes and default values for the columns of the table.

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2</td>
<td>(DFG2)</td>
<td>Create Table: Nulls and Default Values</td>
<td>Row 1 to 1 of 1</td>
</tr>
</tbody>
</table>

Owner .................................................. +
Name ............ TAB1 +
Database ..........
Table Space ....
Sel Column Name + Data Type(Length) + Nulls Default Value +
COL1 CHAR(3)

******************************************************************************* Bottom of data ******************************************************

Command ===> Scroll PAGE
F1=Help F2=Split F3=Exit F4=Expand F7=Backward F8=Forward
F9=Swap F10=Left F11=Right F12=Cancel

Owner
The name of the owner of the table.

Name
The name of the table.

Database
The name of the database in which the table is to be created.

Table Space
The name of the table space in which the table is to be created.

Sel
Type S in this field to display a panel that you use to enter a long default value (up to 512 characters) for the column. See the description for the Default Value field.

Column Name
The name of the column.

Data Type (Length)
The data type and, where applicable, the length of the column.

Nulls
Determines if the field can contain null values, and the default value (if any) to be used for the column when a row is inserted in the table without explicitly specifying a value for the column. Valid values are:

(blink)
Permit nulls and use null as the default value. No NOT NULL or WITH DEFAULT clause is generated in the CREATE TABLE statement.

Chapter 15. FM/DB2 panels and fields 479
Create Table: Nulls and Default Values panel

For non-identity columns, leaving the Nulls field blank and specifying DN are equivalent. For identity columns only, a blank value in the Nulls field is equivalent to specifying NN.

**NN** Disallow nulls, with no default value. A NOT NULL clause is generated in the CREATE TABLE statement.

**ND** Disallow nulls, and use the default value in the Default Value field or the system-defined non-null default value. A NOT NULL WITH DEFAULT clause is generated in the CREATE TABLE statement.

**NU** Disallow nulls, and use the value of the user special register at the time of insert or load as the default value. A NOT NULL WITH DEFAULT USER clause is generated in the CREATE TABLE statement.

**NS** Disallow nulls, and use the SQL authorization ID of the process at the time of insert or load as the default value. A NOT NULL WITH DEFAULT CURRENT SQLID clause is generated in the CREATE TABLE statement.

**DN** Permit nulls and use null as the default value. A DEFAULT NULL clause is generated in the CREATE TABLE statement.

**DD** Permit nulls, and use the default value in the Default Value field or the system-defined non-null default value. A WITH DEFAULT clause is generated in the CREATE TABLE statement.

**DU** Permit nulls, and use the value of the user special register at the time of insert or load as the default value. A WITH DEFAULT USER clause is generated in the CREATE TABLE statement.

**DS** Permit nulls, and use the SQL authorization ID of the process at the time of insert or load as the default value. A WITH DEFAULT CURRENT SQLID clause is generated in the CREATE TABLE statement.

**Default Value**

The value that a column contains when a row is inserted. If you specify a constant, you must specify either ND or DD in the Nulls field; otherwise, leave this field blank.

If you need to specify a default value that is too long to fit in the field on the panel, enter $ in the Sel field to display a panel. If a long default value already exists and you alter the part of the value displayed in the Default Value field, FM/DB2 displays the same panel that you can use to confirm the changes you have made.

The following points apply to the Default Value field:

- In most cases, when you specify a character string as a default value, you do not need to enclose the string in quotation marks. FM/DB2 places quotation marks around unquoted character strings when it generates the SQL.
- You must enclose a character string in quotation marks when the string contains trailing blanks.
  - Specify “Twenty characters ”, as:
    - ‘Twenty characters ’
- If the string contains quotation marks, represent each quotation mark with two quotation marks.
  - Specify “It’s Diana’s house” as:
Create Table: Nulls and Default Values panel

```
It's Diana's house
Specify “Captain Cook sailed on the 'Endeavour” as:
Captain Cook sailed on the "Endeavour"

Note: If the string starts and ends with quotation marks, place the entire string within quotation marks and represent each quotation mark contained within the string with two quotation marks.

Specify "'Harry's Practice" as:
''Harry's Practice''

• If the data type of the column is a distinct type, specify the default value only (FM/DB2 generates the required casting function for the distinct type). If the distinct type is based on a string data type, enclose the constant within quotation marks.
```

Parent panels
• "Create Table panel” on page 459

Child panels
None.

Related tasks
• "Creating a table” on page 176

Create Table: Options panel

You use the Create Table: Options panel to specify information to define options for the table you are creating.

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>Create Table: Options</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Processing Options, Internal Identifiers:
- EDITPROC . . DSNBEAE1 (optional)
- VALIDPROC . . (optional)
- OBID value . . (optional)

Audit and Data Storage Options:
- Audit Options
  - 1. None (default)
  - 2. Changes
  - 3. All
- Data Storage Options
  - 1. EBCDIC (default)
  - 2. ASCII
  - 3. UNICODE

Logging and Table Deletion Options:
- Enter '/' to select option
- Log data capture changes (default is not to log)
- Restrict drop of table (default is to allow)

Command ===>
F1=Help F2=Split F3=Exit F4=Retrieve F7=Backward F8=Forward
F9=Swap F10=Actions F12=Cancel
Create Table: Options panel

EDITPROC
If the table you are creating requires an EDITPROC clause, specify the name of the edit procedure here. The default is not to use an EDITPROC clause. An edit procedure transforms data at row level within DB2 before presentation.

VALIDPROC
If the table you are creating requires a VALIDPROC clause, specify the name of the validation procedure here. The default is not to use a VALIDPROC clause. A validation procedure validates data at row level before it is added.

OBID value
Use this field, if necessary, to specify the integer for the OBID clause for the table you are creating. DB2 assigns an (internal) object identifier value to each DB2 object. There are circumstances where it is desirable to specify the value for a DB2 object instead of using the DB2-assigned value. An example is where there are two parallel DB2 subsystems, with the same objects defined in each, and where every matching object has the same OBID value.

Audit Options
DB2 provides 3 auditing options which determine the type of access to the table that causes auditing to be performed:

1. None
   No auditing performed. This is the default setting.

2. Changes
   Auditing performed when changes are made to the table.

3. All
   Auditing performed when the table is accessed.

Data Storage Options
The encoding scheme for string data stored in the table. FM/DB2 uses the selected option when it generates the CCSID clause in the CREATE TABLE statement.

1. EBCDIC
   String data encoded using EBCDIC CCSIDs. This is the default setting.

2. ASCII
   String data encoded using ASCII CCSIDs.

3. UNICODE
   UNICODE string data encoded using UNICODE CCSIDs.

Log data capture changes
Whether additional information is logged for INSERT, UPDATE and DELETE operations.

Slash (/)
FM/DB2 generates a DATA CAPTURE CHANGES clause in the CREATE TABLE statement.

(blank)
FM/DB2 generates a DATA CAPTURE NONE clause in the CREATE TABLE statement. This is the default setting.

Restrict drop of table
Whether the table cannot be dropped. You can use this option to prevent the accidental deletion of tables.
Create Table: Options panel

Slash (/)

FM/DB2 generates a WITH RESTRICT ON DROP in the CREATE TABLE statement.

(blank)

FM/DB2 allows the table to be dropped. This is the default setting.

Parent panels

- "Create Table panel" on page 459

Child panels

None.

Related tasks

- "Creating a table" on page 176

Create Table: Procedure Exits panel

You use the Create Table: Procedure Exits panel to specify information to define procedure exits for the table you are creating.

Panel and field definitions

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Name</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>COL1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Column Name

The name of the column in the table you are creating.

Name

The program name of the field procedure exit.

Parameters

A list of parameters passed to the field procedure exit when it is called. The list must be comma-delimited and enclosed in parenthesis as required by the SQL syntax.
Create Table: Procedure Exits panel

Parent panels
- "Create Table panel" on page 459

Child panels
None.

Related tasks
- "Creating a table" on page 176

Create Table: Table Check Constraints panel

You use the Create Table: Table Check Constraints panel to specify information to define table check constraints for the table.

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>Create Table: Column Check Constraints</td>
<td>Row 1 to 1 of 1</td>
<td></td>
</tr>
</tbody>
</table>

Owner . . . . . . +
Name . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . | | |
| Name . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . | | | |
| Database . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . | | | |
| Table Space . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . | | | |
| Sel Column Name + Data Type(Length) + Constraint Name + Condition + Check |
| COL1 CHAR(3) | | | |
| ************************************************************ Bottom of data ************************************************************

Cmd
Line command area. Use the following line commands to manipulate the panel rows containing table check constraint information:

- **I**n: Insert n new rows.
- **R**epeat the current row n times.
- **D**elete n rows.
- **S**: Select a row. FM/DB2 displays a scrollable panel that you use to specify a check condition up to the maximum length allowed by DB2. See the description for the Check Condition field.

Constraint Name
The name of the constraint. This is an optional field.

Check Condition
SQL expression defining the table check constraint.
If you need to specify a check condition that is too long to fit in the field on the panel, enter S in the Cmd field to display a panel. If a long check
Create Table: Table Check Constraints panel

Condition already exists and you alter the part of the check condition displayed on the Table Check Constraints panel, FM/DB2 displays the same panel that you can use to confirm the changes you have made.

**Note:** FM/DB2 does not validate the check condition you specify.

**Parent panels**
- "Create Table panel" on page 459

**Child panels**
None.

**Related tasks**
- "Creating a table" on page 176

Create Table: Table Referential Constraint panel

You use the Create Table: Table Referential Constraint panel to specify the foreign key and parent key columns for a table referential constraint.

**Panel and field definitions**

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2</td>
<td>Create Table: Table Referential Constraint Row 1 of 1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Constraint:
- Name ............ CON1
- Parent Table Owner ..
- Parent Table Name ... EMP

Order Column Name Data Type(Length) Parent Key Column
- COL1 CHAR(3)

**************************************** Bottom of data ****************************************

Command ===> Scroll PAGE
F1=Help F2=Split F3=Exit F7=Backward F8=Forward F9=Swap F12=Cancel

**Name**  The name of the constraint.

**Parent Table Owner**  The name of the parent table owner.

**Parent Table Name**  The name of the parent table.

**Order**  Enter S or a number to add a column to the foreign key. When you enter S, FM/DB2 generates the next highest number for the key. To deselect a column that is already selected, clear the displayed number by overtyping it with spaces.
Create Table: Table Referential Constraint panel

Note: The actual numbers are not important; FM/DB2 uses their relative order to define the column order for the key.

Column Name
The name of the column in the table you are creating.

Data Type(Length)
The data type of the column in the table you are creating.

Parent Key Column
Specify the name of the column in the parent table that you want to relate to the foreign key column (Column Name) you have ordered. If you leave this field blank, FM/DB2 relates the foreign key to the primary key of the parent table.

Note: If you have selected more than one column for the foreign key (by entering S or a number in the Order field), the corresponding Parent Key Columns must all be blank or all be filled.

Parent panels
- "Create Table: Table Referential Constraints panel"

Child panels
None.

Related tasks
- "Creating a table" on page 176

Create Table: Table Referential Constraints panel

You use the Create Table: Table Referential Constraints panel to specify information to define one or more table referential constraints for the table.

Panel and field definitions
Create Table: Table Referential Constraints panel

Cmd  Line command area. Use the following line commands to manipulate the panel rows containing table referential constraint information:
- **I**  Insert \( n \) new rows.
- **R**  Repeat the current row \( n \) times.
- **D**  Delete \( n \) rows.
- **S**  Select the row and display the Referential Constraint panel.

Constrnt Name  
The name of the constraint. This is an optional field.

Foreign key columns...  
A list of the columns currently selected as the foreign key for this constraint. FM/DB2 displays only as many columns as fit in the available space.

Parent Table Owner  
Optionally use this field to qualify the name of the parent table.

Parent Table Name  
The name of the table containing the parent key. This field is mandatory.

ON DLT  
See details for the **ON DLT** field as described in "Create Table: Column Referential Constraints panel" on page 473.

Parent panels
- "Create Table panel" on page 459

Child panels

<table>
<thead>
<tr>
<th>To display this panel...</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;Create Table: Table Referential Constraint panel&quot; on page 485</td>
<td>Enter S in Cmd field</td>
</tr>
</tbody>
</table>

Related tasks
- "Creating a table" on page 176

Related references
- "Create Table: Table Referential Constraint panel" on page 485

Create Table: Unique Constraints panel

You use the Create Table: Unique Constraints panel to specify information to define a primary key and up to three additional unique keys for the table you are creating.
Create Table: Unique Constraints panel

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>Create Table: Unique Constraints</td>
<td>Row 1 to 1 of 1</td>
<td></td>
</tr>
</tbody>
</table>

Constraint Names (optional):
- Primary Key
- Unique Key 1
- Unique Key 2
- Unique Key 3

<---- Column Order in Key ---->
Primary Unique1 Unique2 Unique3 Column Name + Datatype(length) +
COL1 CHAR(3)

******************************* Bottom of data ****************************

Command ====> Scroll PAGE
F1=Help F2=Split F3=Exit F4=Expand F7=Backward F8=Forward
F9=Swap F10=Left F11=Right F12=Cancel

Primary
If not blank, shows the relative order of the column for the primary key.

Unique1
If not blank, shows the relative order of the column for the first unique key.

Unique2
If not blank, shows the relative order of the column for the second unique key.

Unique3
If not blank, shows the relative order of the column for the third unique key.

Column Name
The name of the column.

Datatype(length)
The data type and, if applicable, the length of the column.

Parent panels
- "Create Table panel" on page 459

Child panels
None.

Related tasks
- "Creating a table" on page 176

Create Trigger panel

You use the Create Trigger panel to create a new trigger.
Panel and field definitions

Create Trigger panel

Schema
Name of schema.

Name
Name of new trigger.

Creation Options
Specify the number of the creation option you want to use:

1. Details
To specify information to define the type of trigger to be created. Displays the Create Trigger: Details panel.

2. Search Conditions
To specify information to define the optional search condition to restrict the execution of the trigger. Displays the Create Trigger: Search Condition panel.

3. SQL statement
To specify SQL statements to be executed when the trigger is activated. Displays the Create Trigger: SQL statement panel.

4. Create the Trigger
When you have finished specifying details for the trigger to be created, select this creation option to create the trigger.

Parent panels
- “DB2 Object Functions panel” on page 510
- “Triggers panel” on page 712

Child panels

To display this panel... Use/do this

“Create Trigger: Details panel” on page 490 Creation option 1

“Create Trigger: Search Condition panel” on page 491 Creation option 2
Create Trigger panel

To display this panel... Use/do this

"Create Trigger: SQL statement panel" on page 492 Creation option 3

Related tasks
• "Creating a trigger" on page 188

Create Trigger: Details panel

You use the Create Trigger: Details panel to specify information to define the type of trigger to be created.

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2) Create Trigger: Details</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Trigger:
Order (required) Type (required)
- 1. No cascade before - 1. Insert
- 2. After - 2. Delete
- 3. Update

Columns (optional, separate by comma)

Trigger Table:
Owner ........ + (optional)
Name ........ +

Referencing Old:
Correlation name . + (optional)
Table identifier . + (optional, after trigger)

Referencing New:
Correlation name . + (optional)
Table identifier . + (optional, after trigger)

Executed (Required)
- 1. For each modified row
- 2. Once (After Trigger)

Command ===>
F1=Help F2=Split F3=Exit F4=Expand F7=Backward F8=Forward
F9=Swap F10=Left F11=Right F12=Cancel

No cascade before
Defines the trigger as a before trigger.

After
Defines the trigger as an after trigger.

Type

Insert
The trigger is executed when a row is inserted in the trigger table.

Delete
The trigger is executed when a row is deleted from the trigger table.

Update
The trigger is executed when a row is updated from the trigger table.
Create Trigger: Details panel

Columns
Specifies a list of columns to restrict the execution of an update trigger to updates on the specified columns.

Name
Name of the table against which an action causes the trigger to be executed.

Owner
Owner of the trigger table.

Old correlation name
Correlation name used to reference rows as they were before the triggering SQL operation.

New correlation name
Correlation name used to reference rows as they were after the triggering SQL operation.

Old table identifier
Temporary table identifier used to reference the complete set of rows as they were before the SQL operation.

New table identifier
Temporary table identifier used to reference the complete set of rows as they were after the SQL operation.

1. For each modified row
   The trigger option is executed for each modified row of the trigger table.

2. Once (After Trigger)
   The trigger action is executed only once for the triggering SQL operation.

Parent panels
- "Create Trigger panel" on page 488

Child panels
None.

Related tasks
- "Creating a trigger" on page 188

Create Trigger: Search Condition panel

You use the Create Trigger: Search Condition panel to specify information to define the optional search condition to restrict the execution of the trigger. You do not need to type the WHEN keyword and enclosing parentheses for the condition; they are added automatically if omitted.
Create Trigger: Search Condition panel

Panel and field definitions

Search Condition
Freeform entry area for the search condition. The triggered SQL statements are executed only if the search condition evaluates to true or is omitted.

Parent panels
• "Create Trigger panel" on page 488

Child panels
None.

Related tasks
• "Creating a trigger" on page 188

Create Trigger: SQL statement panel

You use the Create Trigger: SQL statement panel to specify SQL statement(s) to be executed when the trigger is activated. You must use a semicolon to separate multiple SQL statements, but you do not need to enclose the statements in BEGIN ATOMIC ... END keywords; they are added automatically if omitted.
Panel and field definitions

```
Process Options Utilities Help

FM/DB2 (DFG2) Trigger SQL statement

Triggered SQL Statement (required, separate with semicolons)

Command ===> F1=Help F2=Split F3=Exit F4=Expand F7=Backward F8=Forward
F9=Swap F10=Left F11=Right F12=Cancel
```

**Triggered SQL Statements**
Freeform entry area for the trigger SQL. Enter one or more SQL statements separated by semicolons.

**Parent panels**
- "Create Trigger panel" on page 488

**Child panels**
None.

**Related tasks**
- "Creating a trigger" on page 188

Create View panel

You use the Create View panel to create a new view.
### Panel and field definitions

**Owner**
The name of the owner of the view.

**Name**
The name of the view to be created.

**Subselect clause**
SQL subselect clause defines the contents of the view.

**Column names**
A list of column names defining the names of the view columns. Must be specified if the subselect clause result table contains duplicate or unnamed columns. If specified, must contain the same number of columns as the subselect clause result table.

**With Check Option (Optional):**
- 1. **With Cascaded Check Option**
- 2. **With Local Check Option**

**Command ===**
F1=Help  F2=Split  F3=Exit  F4=Expand  F7=Backward  F8=Forward
F9=Swap  F10=Left  F11=Right  F12=Cancel

### Owner
The name of the owner of the view.

### Name
The name of the view to be created.

### Subselect clause
SQL subselect clause defines the contents of the view.

### Column names
A list of column names defining the names of the view columns. Must be specified if the subselect clause result table contains duplicate or unnamed columns. If specified, must contain the same number of columns as the subselect clause result table.

### With Check Option
Controls whether or not inserted rows must conform to the view definition.

1. **With Cascaded Check Option**
   Generates WITH CASCADED CHECK OPTION clause.

2. **With Local Check Option**
   Generates WITH LOCAL CHECK OPTION clause.

### Parent panels
- **“DB2 Object Functions panel” on page 510**

### Child panels
None.

### Related tasks
- **“Creating a view” on page 183**
Create View Source Statements panel

Panel and field definitions

Create view keistw.v1 (col1, col2) as select tbname, tbcreator from sysibm.sysffields

Parent panels
- "Tables, Views and Aliases panel” on page 704

Child panels
None.

Related tasks

Data Create Utility panel

You use the Data Create Utility panel to populate a DB2 table with a specified number of rows of data.
### Panel and field definitions

<table>
<thead>
<tr>
<th>Location</th>
<th>Database</th>
<th>Owner</th>
<th>Name</th>
<th>Create Count</th>
<th>Template</th>
<th>Processing Options</th>
</tr>
</thead>
<tbody>
<tr>
<td>Location</td>
<td>Database</td>
<td>Owner</td>
<td>Name</td>
<td>Create Count</td>
<td>Template</td>
<td>Processing Options</td>
</tr>
<tr>
<td>Location</td>
<td>Database</td>
<td>Owner</td>
<td>Name</td>
<td>Create Count</td>
<td>Template</td>
<td>Processing Options</td>
</tr>
</tbody>
</table>

**Location**
The location name for the DB2 object. You can specify a generic name to select from a list. If you enter a value in this field, synonym object names are not recognized or included in the selection list.

**Database**
The database name for the DB2 object. You can use this field to optionally specify a database to further define the object to be processed. If you enter a value in this field:
- The values displayed in the Owner and Name selection lists are limited to those values defined within the specified database.
- Synonym names are not recognized or included in the selection list.

**Owner**
The name of the owner of the object. You can specify a generic name to select from a list. To specify a synonym in the object Name field, either leave this field blank, or specify a generic name that includes the current SQLID or specify the current SQLID. Synonyms are not recognized or included on the table selection list if any of the Table space, Database, or Location fields are non-blank.

If the Translate DB2 object names option is selected, lowercase characters are converted to uppercase. If this option is not selected, lowercase characters are not converted to uppercase; in this situation, a lowercase owner name such as “sysibm” does not match “SYSIBM” in the DB2 catalog.

**Table space**
The table space name for the DB2 object. To further define the object to be processed, you can use this field to optionally specify a table space. If you enter a value in this field:
- The values displayed in the Owner and Name selection lists are limited to those values defined within the specified table space.
- Synonym names are not recognized, nor included in the selection list.
Data Create Utility panel

Name  The name of the object name to be processed. You can specify a generic name to select from a list. To specify a synonym in this field, type the name of the synonym and ensure that the Owner field is blank, contains a generic name that includes the current SQLID, or contains the current SQLID. Synonyms are not recognized or included in the table selection list if any of the Table space, Database, or Location fields are non-blank.

If the Translate DB2 object names option is selected, lowercase characters are converted to uppercase. If this option is not selected, lowercase characters are not converted to uppercase; in this situation, a lowercase owner name such as “sysibm” does not match “SYSIBM” in the DB2 catalog.

Create Count  Specify the number of rows you want to create in the DB2 table. This is a required field and must be in the range 1–2147483647.

Data set name  If you are using a specific template to specify the data initialization details, (see option 1 in Template usage), specify the name of the template data set. It can be a fully-qualified data set name or a pattern. The name may include a member name or name pattern in parenthesis. If the member is specified here, the associated Member field must be empty.

Member  If you specified the name of a partitioned data set (PDS) without including a member name or name pattern in parenthesis in the Data set name field, then you can use this field to specify the member name or a member name pattern.

Processing Options  You can use these options to specify which template you want to use for the create process, and whether you want to edit the template before the create process.

Template usage  The Data Create Utility panel provides four options for template processing:

1. Above  Requires that you enter the name of a template data set (and optionally a member name) in the Template section of the panel. If you do not specify a member name, or if you specify a pattern, FM/DB2 displays a member selection list. After you have specified the member containing a template, processing continues using this template in place of the automatically-generated template for the table (for a detailed explanation, see Template Usage option 3).

2. Previous  Uses the last (previously used) template for this table.

3. Generate from table  Generates a template based on the information in the DB2 catalog for the specified table. This is the default setting. If you have specified a member name in the Template section of the panel, it is ignored. FM/DB2 generates the required template in memory.

4. Generate/Replace  Generates a template based on the information in the DB2
Data Create Utility panel

catalog for the specified table, and saves this information in the member specified in the **DB2 Template** section of the panel. If the member already exists, it is replaced. Any prior customization is lost.

You can select the following options by entering a “/” or an “A”:

**Edit template**
Use to edit the template before creating the data.

**Batch execution**
Builds the necessary JCL to perform the Create function in a batch job. The generated JCL is displayed using the ISPF editor, where you can review the JCL and submit the job for batch processing.

**Create audit trail**
Controls whether access to DB2 by FM/DB2 is written to the FM/DB2 audit log. This option only appears when SAF-rule controlled auditing is in effect.

FM/DB2 may write audit records to either SMF, or a data set.

When SAF rules are used to control FM/DB2 auditing, the decision to write audit records is based on the DB2 object name being processed, and the current userid (TSO logonid). Therefore auditing may be required when processing one DB2 object, but not another. In addition, it is possible that the auditing requirement cannot be changed for a particular DB2 object; this might be different for another DB2 object, and the requirement may be different for other TSO userids.

If you do not have the authority to change the audit setting when processing a particular DB2 object, FM/DB2 prompts you if you enter the wrong setting. If you do have the authority to set the audit setting for a particular DB2 object, FM/DB2 honors whatever value is specified in this field.

**Parent panels**
- [“Utility Functions panel” on page 720](#)

**Child panels**

<table>
<thead>
<tr>
<th>To display this panel...</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td><a href="#">“Column Selection/Edit panel” on page 407</a></td>
<td>Select Edit template</td>
</tr>
</tbody>
</table>

**Related tasks**
- Chapter 6, “Populating a DB2 table with data,” on page 191
- [“Specifying a data set and a member name” on page 30](#)
- [“Selecting options on FM/DB2 panels” on page 24](#)
- Chapter 3, “Working with templates,” on page 49
- [“SAF-rule controlled auditing” on page 47](#)

**Database Request Modules panel**

You use the Database Request Modules panel to list the DBRMs in the DB2 catalog.
**Panel and field definitions**

The columns that are displayed include the SEL field and columns of SYSIBM.SYSDBRM.

```
<table>
<thead>
<tr>
<th>SEL</th>
<th>NAME</th>
<th>PDSNAME</th>
<th>TIMESTEX</th>
<th>PLNAME</th>
<th>PLCREATOR</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

Note:
1. The system option, **Show all catalog table columns**, affects which columns FM/DB2 displays when you use the Object List utility.
2. You can also customize which columns FM/DB2 displays when you use the Object List utility. For details, see the **File Manager Customization Guide**.

To view other columns on this panel, press the Right function key (F11).

For a complete list of columns, refer to “DB2 catalog tables” in the **DB2 for z/OS SQL Reference** relevant to your version of DB2.

**SEL** Line command area.

For a list of the line commands you can use on this panel, see **Table 22 on page 250**.

**Parent panels**
- “Object List Utility panel” on page 610
- “Application Plans panel” on page 389

**Child panels**

<table>
<thead>
<tr>
<th>To display this panel...</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>(ISPF panel)</td>
<td>Line command B</td>
</tr>
<tr>
<td>“Details panels” on page 522</td>
<td>Line command I</td>
</tr>
<tr>
<td>“Application Plans panel” on page 389</td>
<td>Line command PL</td>
</tr>
<tr>
<td>“Display Row panel” on page 523</td>
<td>Line command ROW</td>
</tr>
<tr>
<td>“Sort Fields panel” on page 679</td>
<td>Primary command SORT</td>
</tr>
</tbody>
</table>
Database Request Modules panel

Related tasks
- "Working with object list panels" on page 244
- "Using the line command area (Cmd)" on page 250

Related references
- "SORT primary command" on page 777

Database Structure panel

FM/DB2 displays the Database Structure panel if you use the DS line command against an item listed on:
- The "Databases panel," or
- The "Table Spaces panel" on page 698

Databases panel

You use the Databases panel to list database object types in the DB2 catalog.

Panel and field definitions

The columns that are displayed include the SEL field and columns of SYSIBM.SYSDATABASE.

Note:
1. The system option, Show all catalog table columns, affects which columns FM/DB2 displays when you use the Object List utility.
2. You can also customize which columns FM/DB2 displays when you use the Object List utility. For details, see the File Manager Customization Guide.

To view other columns on this panel, press the Right function key (F11).

For a complete list of columns, refer to "DB2 catalog tables" in the DB2 for z/OS SQL Reference relevant to your version of DB2.
Databases panel

SEL  Line command area.
For a list of the line commands you can use on this panel, see Table 22 on page 250.

Parent panels

• "Object List Utility panel" on page 610
• "Table Spaces panel" on page 698
• "Tables, Views and Aliases panel" on page 704
• "Indexes panel" on page 595

Child panels

To display this panel...  Use/do this
"Alter Database panel" on page 371  Line command A
"Create Database panel" on page 432  Line command CR
"Create Table Space panel" on page 462  Line command CRS
See "Drop panels" on page 524  Line command DR
"Database Structure panel" on page 500  Line command DS
"Grant privileges panels" on page 587  Line command G
"Generate SQL From DB2 Catalog panel" on page 585  Line command GEN
"Details panels" on page 522  Line command I
"Recovery Information panel" on page 638  Line command ICS
"Privileges panels" on page 624  Line command P
"Revoke privileges panels" on page 647  Line command R
"Display Row panel" on page 523  Line command ROW
"Table Spaces panel" on page 698  Line command S
"Tables, Views and Aliases panel" on page 704  Line command T
"Indexes panel" on page 595  Line command X
"Sort Fields panel" on page 679  Primary command SORT

Related tasks

• "Working with object list panels" on page 244
• "Using the line command area (Cmd)" on page 250

Related references

• "SORT primary command" on page 777

DB2 Browse panel

You use the DB2 Browse panel to enter the name of an object to be browsed. You also use this panel to specify the name of a template that describes a logical view of the object.
DB2 Browse panel

Panel and field definitions

| Location | The location name for the DB2 object. You can specify a generic name to select from a list. If you enter a value in this field, synonym object names are not recognized or included in the selection list. |
| Database | The database name for the DB2 object. You can use this field to optionally specify a database to further define the object to be processed. If you enter a value in this field: |
| Owner | The name of the owner of the object. You can specify a generic name to select from a list. To specify a synonym in the object Name field, either leave this field blank, or specify a generic name that includes the current SQLID or specify the current SQLID. Synonyms are not recognized or included on the table selection list if any of the Table space, Database, or Location fields are non-blank. |
| Table space | The table space name for the DB2 object. To further define the object to be processed, you can use this field to optionally specify a table space. If you enter a value in this field: |

If the Translate DB2 object names option is selected, lowercase characters are converted to uppercase. If this option is not selected, lowercase characters are not converted to uppercase; in this situation, a lowercase owner name such as “sysibm” does not match “SYSIBM” in the DB2 catalog.

Template: Data set name... 'FMNUSER.TEMPLATE'
Member ....... SALARY

Processing Options:
- Template usage: Enter "/", "A"lways to select option
- 3. Above: Edit options
- 2. Previous: Edit template
- 3. Generate from table: Re-edit template
- 4. Generate/Replace

Command ===> F1=Help F2=Split F3=Exit F4=Expand F7=Backward F8=Forward F9=Swap F10=Left F11=Right F12=Cancel

FM/DB2 (DFG2)  DB2 Browse

Specify the DB2 Object:
Location . . . . Database . . (optional)
Owner . . . . . . FMNUSER + Table space . . (optional)
Name . . . . . . EMP +
Start position . . 1
Row count . . . . 100

Number of rows to browse

Template: Data set name... 'FMNUSER.TEMPLATE'
Member ....... SALARY

Processing Options:
- Template usage: Enter "/", "A"lways to select option
- 3. Above: Edit options
- 2. Previous: Edit template
- 3. Generate from table: Re-edit template
- 4. Generate/Replace

Command ===> F1=Help F2=Split F3=Exit F4=Expand F7=Backward F8=Forward F9=Swap F10=Left F11=Right F12=Cancel

FM/DB2 (DFG2)  DB2 Browse

Specify the DB2 Object:
Location . . . . Database . . (optional)
Owner . . . . . . FMNUSER + Table space . . (optional)
Name . . . . . . EMP +
Start position . . 1
Row count . . . . 100

Number of rows to browse

Template: Data set name... 'FMNUSER.TEMPLATE'
Member ....... SALARY

Processing Options:
- Template usage: Enter "/", "A"lways to select option
- 3. Above: Edit options
- 2. Previous: Edit template
- 3. Generate from table: Re-edit template
- 4. Generate/Replace

Command ===> F1=Help F2=Split F3=Exit F4=Expand F7=Backward F8=Forward F9=Swap F10=Left F11=Right F12=Cancel

Location
The location name for the DB2 object. You can specify a generic name to select from a list. If you enter a value in this field, synonym object names are not recognized or included in the selection list.

Database
The database name for the DB2 object. You can use this field to optionally specify a database to further define the object to be processed. If you enter a value in this field:
- The values displayed in the Owner and Name selection lists are limited to those values defined within the specified database.
- Synonym names are not recognized or included in the selection list.

Owner
The name of the owner of the object. You can specify a generic name to select from a list. To specify a synonym in the object Name field, either leave this field blank, or specify a generic name that includes the current SQLID or specify the current SQLID. Synonyms are not recognized or included on the table selection list if any of the Table space, Database, or Location fields are non-blank.

If the Translate DB2 object names option is selected, lowercase characters are converted to uppercase. If this option is not selected, lowercase characters are not converted to uppercase; in this situation, a lowercase owner name such as “sysibm” does not match “SYSIBM” in the DB2 catalog.

Table space
The table space name for the DB2 object. To further define the object to be processed, you can use this field to optionally specify a table space. If you enter a value in this field:
- The values displayed in the Owner and Name selection lists are limited to those values defined within the specified table space.
- Synonym names are not recognized, nor included in the selection list.
Name  The name of the object name to be processed. You can specify a generic name to select from a list. To specify a synonym in this field, type the name of the synonym and ensure that the Owner field is blank, contains a generic name that includes the current SQLID, or contains the current SQLID. Synonyms are not recognized or included in the table selection list if any of the Table space, Database, or Location fields are non-blank.

If the Translate DB2 object names option is selected, lowercase characters are converted to uppercase. If this option is not selected, lowercase characters are not converted to uppercase; in this situation, a lowercase owner name such as “sysibm” does not match “SYSIBM” in the DB2 catalog.

Start position
Indicates where FM/DB2 is to begin loading rows from the result table into the FM/DB2 editor.

If you specify a value greater than 1 (the default), FM/DB2 skips n-1 rows of the result table before loading rows into the FM/DB2 editor. If the Row count value is ALL (or 0 or *), when the data is displayed you can scroll up to see the rows that were skipped. For a fixed Row count value, you cannot scroll backwards to see any skipped rows.

Row count
Use this option to specify the maximum number of rows to be retrieved from DB2 and loaded into memory for the Browse session. You can specify a value in the range 1–999999999 or, to indicate that all rows are to be retrieved, specify ALL, 0, or *. The default value is 100.

An FM/DB2 Browse session only applies to those rows fetched from DB2 and loaded into the editor. It does not apply to any unfetched rows. This means that the FIND primary command only applies to those rows fetched from DB2.

Note: When you are browsing large tables, use the value of ALL, 0, or * with caution as it is possible to encounter storage problems. The amount of storage required by the editor depends on both the number of rows retrieved, and the row length. Use the row selection criteria section of the template to limit the number of rows retrieved.

Data set name
The name of a data set that contains, or will contain, the template. It can be a fully-qualified data set name or a pattern. The name may include a member name or name pattern in parenthesis. If the member is specified here, the associated Member field must be empty.

Member
If you specified the name of a partitioned data set (PDS) without including a member name or name pattern in parenthesis in the Data set name field, then you can use this field to specify the member name or a member name pattern.

Template usage
The DB2 Browse panel provides four options for template processing:

1. Above
Requires that you enter the name of a template data set (and optionally a member name) in the Template section of the panel. If you do not specify a member name, or if you specify a pattern, FM/DB2 displays a member selection list. After you have specified
the member containing a template, processing continues using this
template in place of the automatically-generated template for the
table (for a detailed explanation, see Template Usage option 3).

2. Previous
Uses the last (previously used) template for this table.

3. Generate from table
Generates a template based on the information in the DB2 catalog
for the specified table. This is the default setting. If you have
specified a member name in the Template section of the panel, it is
ignored. FM/DB2 generates the required template in memory.

4. Generate/Replace
Generates a template based on the information in the DB2 catalog
for the specified table, and saves this information in the member
specified in the Template section of the panel. If the member
already exists, it is replaced. Any prior customization is lost.

You can select this option to change the options for the current editor session:

Edit options
Enter one of these:

A “/” or an “A”
Displays the first FM/DB2 editor options panel, prior to displaying
the data for the DB2 object being processed.

A number in the range 1–8
Displays the nth FM/DB2 editor options panel directly where n is
the value entered.

You can select the following options by entering a “/” or an “A”:!

Edit template
Select this option to edit the template before browsing the data.

Re-edit template
Select this option to control the navigation between panels within the
Browse dialog. This option controls the behavior of the F3 and F12 keys
when pressed from within a Browse session.

Parent panels
- “Table Browse panel” on page 689 (when BROWSE primary command entered)
- “Table Edit panel” on page 691 (when BROWSE primary command entered)
- “Table View panel” on page 700 (when BROWSE primary command entered)
- “Primary Option Menu panel” on page 617 (when "B" entered on command line)
- “Tables, Views and Aliases panel” on page 704 (when B line command entered)

Child panels

<table>
<thead>
<tr>
<th>To display this panel...</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>“Table Browse panel” on page 689</td>
<td>No options selected and press Enter</td>
</tr>
<tr>
<td>“Editor Options (1 of 8) panel” on page 528</td>
<td>Select Edit options</td>
</tr>
<tr>
<td>“Column Selection/Edit panel” on page 407</td>
<td>Select Edit template or Re-edit template</td>
</tr>
</tbody>
</table>
Related tasks

- Chapter 4, “Viewing and changing DB2 data,” on page 91
- “Selecting options on FM/DB2 panels” on page 24
- “Specifying a data set and a member name” on page 30
- “Where you can use templates” on page 4
- Chapter 3, “Working with templates,” on page 49
- “View and Edit options (options 1 and 2)” on page 43

**DB2 Edit panel**

You use the DB2 Edit panel to enter the name of an object to be edited. You also use this panel to specify the name of a template that describes a logical view of the object.

**Panel and field definitions**

<table>
<thead>
<tr>
<th>Location</th>
<th>Database</th>
<th>(optional)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Location</td>
<td>Database</td>
<td>(optional)</td>
</tr>
<tr>
<td>Owner</td>
<td>FMUSER</td>
<td>+</td>
</tr>
<tr>
<td>Name</td>
<td>EMP</td>
<td></td>
</tr>
<tr>
<td>Start position</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Row count</td>
<td>100</td>
<td>Number of rows to edit</td>
</tr>
</tbody>
</table>

**Template:**
- Data set name: 'FMUSER.TEMPLATE'
- Member: SALARY

**Processing Options:**
- Enter "/", "A"lways to select option
- 1. Above
- 2. Previous
- 3. Generate from table
- 4. Generate/Replace
- Create audit trail

**Command ==>**
- F1=Help
- F2=Split
- F3=Exit
- F4=Expand
- F7=Backward
- F8=Forward
- F9=Swap
- F10=Left
- F11=Right
- F12=Cancel

**Location**

The location name for the DB2 object. You can specify a generic name to select from a list. If you enter a value in this field, synonym object names are not recognized or included in the selection list.

**Database**

The database name for the DB2 object. You can use this field to optionally specify a database to further define the object to be processed. If you enter a value in this field:

- The values displayed in the **Owner** and **Name** selection lists are limited to those values defined within the specified database.
- Synonym names are not recognized or included in the selection list.

**Owner**

The name of the owner of the object. You can specify a generic name to select from a list. To specify a synonym in the object **Name** field, either leave this field blank, or specify a generic name that includes the current
SQLID or specify the current SQLID. Synonyms are not recognized or included on the table selection list if any of the Table space, Database, or Location fields are non-blank.

If the Translate DB2 object names option is selected, lowercase characters are converted to uppercase. If this option is not selected, lowercase characters are not converted to uppercase; in this situation, a lowercase owner name such as “sysibm” does not match “SYSIBM” in the DB2 catalog.

Table space
The table space name for the DB2 object. To further define the object to be processed, you can use this field to optionally specify a table space. If you enter a value in this field:

- The values displayed in the Owner and Name selection lists are limited to those values defined within the specified table space.
- Synonym names are not recognized, nor included in the selection list.

Name
The name of the object name to be processed. You can specify a generic name to select from a list. To specify a synonym in this field, type the name of the synonym and ensure that the Owner field is blank, contains a generic name that includes the current SQLID, or contains the current SQLID. Synonyms are not recognized or included in the table selection list if any of the Table space, Database, or Location fields are non-blank.

If the Translate DB2 object names option is selected, lowercase characters are converted to uppercase. If this option is not selected, lowercase characters are not converted to uppercase; in this situation, a lowercase owner name such as “sysibm” does not match “SYSIBM” in the DB2 catalog.

Start position
Indicates where FM/DB2 is to begin loading rows from the result table into the FM/DB2 editor.

If you specify a value greater than 1 (the default), FM/DB2 skips n-1 rows of the result table before loading rows into the FM/DB2 editor. If the Row count value is ALL (or 0 or *), when the data is displayed you can scroll up to see the rows that were skipped. For a fixed Row count value, you cannot scroll backwards to see any skipped rows.

Row count
Use this option to specify the maximum number of rows to be retrieved from DB2 and loaded into memory for the editor session. You can specify a value in the range 1–999999999 or, to indicate that all rows are to be retrieved, specify ALL, 0, or *. The default value is 100.

An FM/DB2 editor session only applies to those rows fetched from DB2 and loaded into the editor. It does not apply to any unfetched rows. This means that the FIND and CHANGE primary commands only apply to those rows fetched from DB2.

Note: When you are editing large tables, use the value of 0 with caution as it is possible to encounter storage problems. The amount of storage required by the editor depends on both the number of rows retrieved, and the row length. Use the row selection criteria section of the template to limit the number of rows retrieved.

Data set name
The name of a data set that contains, or will contain, the template. It can
be a fully-qualified data set name or a pattern. The name may include a member name or name pattern in parenthesis. If the member is specified here, the associated **Member** field must be empty.

**Member**

If you specified the name of a partitioned data set (PDS) without including a member name or name pattern in parenthesis in the **Data set name** field, then you can use this field to specify the member name or a member name pattern.

**Template usage**

The DB2 Edit panel provides four options for template processing:

1. **Above**
   - Requires that you enter the name of a template data set (and optionally a member name) in the **Template** section of the panel. If you do not specify a member name, or if you specify a pattern, FM/DB2 displays a member selection list. After you have specified the member containing a template, processing continues using this template in place of the automatically-generated template for the table (for a detailed explanation, see Template Usage option 3).

2. **Previous**
   - Uses the last (previously used) template for this table.

3. **Generate from table**
   - Generates a template based on the information in the DB2 catalog for the specified table. This is the default setting. If you have specified a member name in the **Template** section of the panel, it is ignored. FM/DB2 generates the required template in memory.

4. **Generate/Replace**
   - Generates a template based on the information in the DB2 catalog for the specified table, and saves this information in the member specified in the **Template** section of the panel. If the member already exists, it is replaced. Any prior customization is lost.

**Edit options**

Select this option to change the options for the current editor session by entering one of these:

A **“/” or an “A”**
   - Displays the first FM/DB2 editor options panel, prior to displaying the data for the DB2 object being processed.

A **number in the range 1–8**
   - Displays the nth FM/DB2 editor options panel directly where n is the value entered.

Any options that you set last only for the duration of the current Edit session, and override any conflicting global options. To permanently change the Edit options, select option 0.2.

You can select the following options by entering a **“/” or an “A”**:

**Edit template**
   - Select this option to edit the template before editing the data.

**Re-edit template**
   - Select this option to control the navigation between panels within the Edit dialog. This option controls the behavior of the F3 and F12 keys when pressed from within an Edit session.
Create audit trail

Note: "(*)" is appended to the option name for SAF-controlled auditing when required by SAF rules.
Determines if FM/DB2 generates an audit trail report of all successful modifications to DB2 data made during an Edit session:

/ Audit trail report generated.

(blank) Audit trail report not generated.

Note: The ability to change this option depends on installation options within the DB2 subsystem to which FM/DB2 is connected. The Create an audit trail option can be:

- Permanently selected ("/" shown in the option). Audit trail reporting occurs for Edit processing. You cannot change the setting of the option.
- Permanently deselected (blank shown in the option). Audit trail reporting does not occur for Edit processing. You cannot change the setting of the option.
- Optional. Type a "/" in the option to generate audit reporting for all Edit processing; otherwise, type a space in the option to suppress audit trail reporting of any Edit processing.

At the end of your FM/DB2 session, the Create an audit trail option that you specify can be saved to your ISPF profile, or it can be set to blank (audit trail report not selected), depending on the installation options within the DB2 subsystem to which FM/DB2 is connected. The audit installation options for the currently connected DB2 system can be determined by selecting option 7 from the HELP pull down menu. When the auditing option is “OPTIONAL,OFF”, the Create an audit trail option is always reset to blank (not selected) at the start of an FM/DB2 session. This means that, for an audit log to be written, auditing must be explicitly selected each time a new FM/DB2 session is started. Further, when this option is selected, there is no distinction between the global setting for the audit option, and any value entered using edit options on the DB2 edit panel. For more information, see the File Manager Customization Guide or see your systems administrator.

Parent panels
- “Primary Option Menu panel” on page 617
- “Tables, Views and Aliases panel” on page 704

Child panels

<table>
<thead>
<tr>
<th>To display this panel...</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>“Table Edit panel” on page 691</td>
<td>No options selected and press Enter</td>
</tr>
<tr>
<td>“Editor Options (1 of 8) panel” on page 528</td>
<td>Select Edit options by typing a “/”, an “A”, or “1”</td>
</tr>
<tr>
<td>“Editor Options (2 of 8) panel” on page 531</td>
<td>Select Edit options by typing a “2”</td>
</tr>
<tr>
<td>“Editor Options (3 of 8) panel” on page 535</td>
<td>Select Edit options by typing a “3”</td>
</tr>
</tbody>
</table>
**DB2 Location Selection panel**

You use the DB2 Location Selection panel to display all the available remote locations. You can only access remote locations that have been defined to your local DB2 subsystem.

### Panel and field definitions

```
<table>
<thead>
<tr>
<th>Process Opti</th>
<th>DB2 Location Selection</th>
<th>Row 1 to 4 of 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>Select one of the DB2 locations from the list.</td>
<td></td>
</tr>
<tr>
<td>Specify the DB2 Location . Owner . . Name . .</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Template: Data set nam Member . .</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Processing Opti Template usa
1. Above
2. Previous
3. General
4. General

Command ===> F1=Help F2=Split F3=Exit F7=Backward F8=Forward F9=Swap F12=Cancel
```

### Parent panels

Asterisk (*) or pattern entered in **Location** field on any FM/DB2 panel (except) **Object List Utility.**
DB2 Object Functions panel

You use the DB2 Object Functions panel to create and drop DB2 objects.

Panel and field definitions

```
Process Options Utilities Help

FM/DB2 (DFG2)   DB2 Object Functions

C    Create object                  D    Drop object

Processing Options:
Object Type
2. Table space 6. Index 10. Stored procedure
3. Table 7. Synonym 11. Trigger
4. View 8. Distinct type 12. Auxiliary Table (Create)

Enter "/" to select option
  Confirm object drop
```

You can select the following option by entering a "/":

Confirm object drop
Indicates if you want FM/DB2 to display a confirmation panel each time you enter a DR(op) command against an object

Parent panels

- “Utility Functions panel” on page 720

Child panels

<table>
<thead>
<tr>
<th>To display this panel...</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>“Create Database panel” on page 432</td>
<td>Select object type Database and enter C</td>
</tr>
<tr>
<td>“Create Table Space panel” on page 462</td>
<td>Select object type Table space and enter C</td>
</tr>
<tr>
<td>“Create Table panel” on page 459</td>
<td>Select object type Table and enter C</td>
</tr>
</tbody>
</table>
### DB2 Object Functions panel

<table>
<thead>
<tr>
<th>To display this panel...</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>“Create View panel” on page 493</td>
<td>Select object type View and enter C</td>
</tr>
<tr>
<td>“Create Alias panel” on page 430</td>
<td>Select object type Alias and enter C</td>
</tr>
<tr>
<td>“Create Index panel” on page 442</td>
<td>Select object type Index and enter C</td>
</tr>
<tr>
<td>“Create Synonym panel” on page 458</td>
<td>Select object type Syntax and enter C</td>
</tr>
<tr>
<td>“Create Distinct Type panel” on page 433</td>
<td>Select object type Distinct type and enter C</td>
</tr>
<tr>
<td>“Create Function panel” on page 437</td>
<td>Select object type Function and enter C</td>
</tr>
<tr>
<td>“Create Procedure panel” on page 452</td>
<td>Select object type Stored procedure and enter C</td>
</tr>
<tr>
<td>“Create Trigger panel” on page 488</td>
<td>Select object type Trigger and enter C</td>
</tr>
<tr>
<td>“Create Auxiliary Table panel” on page 431</td>
<td>Select object type Auxiliary table and enter C</td>
</tr>
</tbody>
</table>

Select object type Database and enter D
Select object type Table space and enter D
Select object type Table and enter D
Select object type View and enter D
Select object type Alias and enter D
Select object type Index and enter D
Select object type Syntax and enter D
Select object type Distinct type and enter D
Select object type Function and enter D
Select object type Stored procedure and enter D
Select object type Trigger and enter D

### Related tasks
- Chapter 5, “Creating and dropping DB2 objects,” on page 163

---

**DB2 Save Error Action panel**

You use the DB2 Save Error Action panel to determine the type of error, and what action you can take, when an error occurs during an FM/DB2 Edit session.
DB2 Save Error Action panel

Panel and field definitions

DB2 Save Error Action panel

**DB2 Save Error Action panel**

**Panel and field definitions**

<table>
<thead>
<tr>
<th>DB2 Save Error Action</th>
<th>FM/DB2 (DFG2)</th>
<th>DB2 Save Error Action</th>
<th>Row 1 to 1 of 1</th>
</tr>
</thead>
</table>

DB2 reported a No Primary Key error while attempting to save this row.

Relationship: RPAA (See below for key column details)

Parent table: FMMUSER.ACT

Dependent table: FMMUSER.PROJACT

Explanation: The insert or update operation on this line would have resulted in a foreign key value for which there is no corresponding primary key value.

Instructions: Type REDIT on the command line to edit the parent table shown above. Press Enter or enter EXIT to return to the Edit session and correct the error. Press the CANCEL key to end the Edit session. Any changes made since the last commit point are lost.

<table>
<thead>
<tr>
<th>Parent Column Name</th>
<th>Depndnt Column Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACTNO</td>
<td>ACTNO</td>
<td>000999</td>
</tr>
</tbody>
</table>

Command ===>

- F1=Help
- F2=Split
- F3=Exit
- F7=Backward
- Scroll PAGE
- F6=RChange
- F8=Forward
- F9=Swap
- F12=Cancel

**Related tasks**

- “Example of correcting an error during an Edit session” on page 160

DB2 Subsystem Selection panel

You use the DB2 Subsystem Selection menu to select the DB2 subsystem to which you want FM/DB2 to connect.
Panel and field definitions

**Sel** Selection field. Enter S or / to select the DB2 subsystem shown on that line.

**SSID** The subsystem ID. This ID is displayed in the **DB2 SSID** field on the Primary Option Menu and shows to which DB2 subsystem FM/DB2 is currently connected.

**Status** Shows status of the DB2 subsystem:

- **ACTIVE**
  DB2 system is active and is available for connection.

- **GROUP**
  DB2 system is in a DB2 data sharing environment and is available for connection.

- **INACTIVE**
  DB2 system is not active and is not available for connection.

- **UNAVAIL.**
  FM/DB2 has not been installed on a subsystem or DB2 data sharing group, or FM/DB2 is not available for connection.

**Description** The descriptive name of the DB2 subsystem.
DB2 Subsystem Selection panel

Prefix  Command prefix for the DB2 subsystem.

Parent panels
  • “Primary Option Menu panel” on page 617

Child panels
None.

Related tasks
  • “Selecting the DB2 subsystem when more than one is available” on page 11

DB2 Template Description panel

You use the DB2 Template Description panel to add a description to the template you are creating or modifying.

Panel and field definitions

```
| Command ===|  F1=Help  F2=Split  F3=Exit  F7=Backward  F8=Forward |
|           |  F9=Swap  F12=Cancel                  |
```

Template Data set
The name of the partitioned data set where the template is stored.

DB2 Object
The name of the DB2 object from which the template was built.

Subsystem
The subsystem ID of the DB2 subsystem where the DB2 object resides.

Release
The DB2 version, release, and modification level of the DB2 subsystem on which the object resides.

Created
The date that the template was created from the copybook.

Modified
The date that the template was last modified.
The description of the template. Use this field to enter or modify the description of the template.

Parent panels

- "Column Selection/Edit panel" on page 407

Child panels

None.

Related tasks

- "Editing a template" on page 57

DB2 Utilities panel

You use the DB2 Utilities panel to generate DB2 utility jobs to copy, load, rebuild, reorganize and collect statistics on DB2 objects.

Panel and field definitions

The DB2 Utilities panel shows a list of the available utilities on the left side of the “Processing Options” section, and a list of DB2 object types on the right. Each DB2 utility applies to only certain object types. For example, the LOAD utility only applies to DB2 tables. The valid object types for each utility are shown under the heading “Valid DB2 Obj”.

The panel shows a Specification section and a LISTDEF name field. A selection of LISTDEF means that a LISTDEF name is to be specified as the object of the utility; otherwise, the object name of the type specified in object type is to be specified.

In the Specification section, you can select:
1. Object name
   The object specified in DB2 Object Details is used as the object of the utility.

2. LISTDEF
   The name specified in LISTDEF name is used as the object of the utility.

Note: The LOAD syntax does not have a LISTDEF clause. If you select LOAD in the DB2 Utility section and LISTDEF in the Specification section, FM/DB2 issues a warning message.

In the LISTDEF name section, if you have specified 2 (LISTDEF) in the Specification section, specify the name of the LISTDEF you want to use.

You specify the objects to be processed by the DB2 utility at the bottom of the panel.

Database
   The name of the database for the object.

Table/Index space
   The name of the table space or index space for the object.

Owner
   The name of the owner of the table or index for the object.

Name
   The name of the table or index for the object.

Parent panels
   • “Primary Option Menu panel” on page 617
   • “Table Spaces panel” on page 698
   • “Tables, Views and Aliases panel” on page 704
   • “Indexes panel” on page 595

Child panels

<table>
<thead>
<tr>
<th>To display this panel...</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>“COPY Utility (Table Spaces) panel” on page 427</td>
<td>DB2 utility: COPY; object type: Table space; specification: Object name</td>
</tr>
<tr>
<td>“COPY Utility (Table Spaces) with LISTDEF panel” on page 429</td>
<td>DB2 utility: COPY; object type: Table space; specification: LISTDEF</td>
</tr>
<tr>
<td>“COPY Utility (Index Spaces) panel” on page 421</td>
<td>DB2 utility: COPY; object type: Index space; specification: Object name</td>
</tr>
<tr>
<td>“COPY Utility (Index Spaces) with LISTDEF panel” on page 423</td>
<td>DB2 utility: COPY; object type: Index space; specification: LISTDEF</td>
</tr>
<tr>
<td>“LOAD Utility panel” on page 603</td>
<td>DB2 utility: LOAD; object type: Tables; specification: Object name</td>
</tr>
<tr>
<td>“LOAD Utility panel using DB2 templates” on page 604</td>
<td>DB2 utility: LOAD; object type: Tables; specification: LISTDEF</td>
</tr>
<tr>
<td>Panel Description</td>
<td>Use/do this</td>
</tr>
<tr>
<td>------------------------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>&quot;REBUILD Utility (Indexes) panel&quot; on page 630</td>
<td>DB2 utility: REBUILD; object type: Indexes; specification: Object name</td>
</tr>
<tr>
<td>&quot;REBUILD (Indexes) with LISTDEF panel&quot; on page 632</td>
<td>DB2 utility: REBUILD; object type: Indexes; specification: LISTDEF</td>
</tr>
<tr>
<td>&quot;REBUILD (Indexes for Table Spaces) panel&quot; on page 628</td>
<td>DB2 utility: REBUILD; object type: Indexes for Table space; specification: Object name</td>
</tr>
<tr>
<td>&quot;RECOVER Utility (Table Spaces) panel&quot; on page 636</td>
<td>DB2 utility: RECOVER; object type: Table space; specification: Object name</td>
</tr>
<tr>
<td>&quot;RECOVER Utility (Table Spaces) with LISTDEF panel&quot; on page 637</td>
<td>DB2 utility: RECOVER; object type: Table space; specification: LISTDEF</td>
</tr>
<tr>
<td>&quot;RECOVER Utility (Index Spaces) panel&quot; on page 634</td>
<td>DB2 utility: RECOVER; object type: Index spaces; specification: Object name</td>
</tr>
<tr>
<td>&quot;REORG Utility (Table Spaces) panel&quot; on page 644</td>
<td>DB2 utility: REORG; object type: Table space; specification: Object name</td>
</tr>
<tr>
<td>&quot;REORG Utility (Indexes) panel&quot; on page 642</td>
<td>DB2 utility: REORG; object type: Indexes; specification: Object name</td>
</tr>
<tr>
<td>&quot;REORG Utility (Indexes) with LISTDEF panel&quot; on page 643</td>
<td>DB2 utility: REORG; object type: Indexes; specification: LISTDEF</td>
</tr>
<tr>
<td>&quot;RUNSTATS Utility (Table Spaces) panel&quot; on page 654</td>
<td>DB2 utility: RUNSTATS; object type: Table space; specification: Object name</td>
</tr>
<tr>
<td>&quot;RUNSTATS Utility (Table Spaces) with LISTDEF panel&quot; on page 656</td>
<td>DB2 utility: RUNSTATS; object type: Table space; specification: LISTDEF</td>
</tr>
<tr>
<td>&quot;RUNSTATS Utility (Indexes) panel&quot; on page 652</td>
<td>DB2 utility: RUNSTATS; object type: Indexes; specification: Object name</td>
</tr>
<tr>
<td>&quot;RUNSTATS Utility (Indexes) with LISTDEF panel&quot; on page 653</td>
<td>DB2 utility: RUNSTATS; object type: Indexes; specification: LISTDEF</td>
</tr>
<tr>
<td>&quot;UNLOAD Utility (Tables) &quot;From&quot; panel&quot; on page 718</td>
<td>DB2 utility: UNLOAD; object type: Tables; specification: Object name</td>
</tr>
<tr>
<td>&quot;UNLOAD Utility (Table Spaces) panel&quot; on page 716</td>
<td>DB2 utility: UNLOAD; object type: Table spaces; specification: Object name</td>
</tr>
<tr>
<td>&quot;UNLOAD Utility (Table Spaces) with LISTDEF panel&quot; on page 717</td>
<td>DB2 utility: UNLOAD; object type: Table spaces; specification: LISTDEF</td>
</tr>
</tbody>
</table>
Related tasks

- Chapter 11, “Generating batch JCL for DB2 utility jobs,” on page 277

DB2 View panel

You use the DB2 View panel to enter the name of an object to be edited (but without the ability to save any changes you make). You also use this panel to specify the name of a template that describes a logical view of the object.

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>DB2 View</td>
<td></td>
</tr>
</tbody>
</table>

Specify the DB2 Object:
- Location (optional)
- Owner: FMNUSER (optional)
- Name: EMP
- Start position: 1
- Row count: 100

Template:
- Data set name: FMNUSER.TEMPLATE
- Member: SALARY

Processing Options:
- Template usage
- 1. Above
- 2. Previous
- 3. Generate from table
- 4. Generate/Replace
- Command ==> 

Location

The location name for the DB2 object. You can specify a generic name to select from a list. If you enter a value in this field, synonym object names are not recognized or included in the selection list.

Database

The database name for the DB2 object. You can use this field to optionally specify a database to further define the object to be processed. If you enter a value in this field:
- The values displayed in the Owner and Name selection lists are limited to those values defined within the specified database.
- Synonym names are not recognized or included in the selection list.

Owner

The name of the owner of the object. You can specify a generic name to select from a list. To specify a synonym in the object Name field, either leave this field blank, or specify a generic name that includes the current SQLID or specify the current SQLID. Synonyms are not recognized or included on the table selection list if any of the Table space, Database, or Location fields are non-blank.

If the Translate DB2 object names option is selected, lowercase characters are converted to uppercase. If this option is not selected, lowercase
DB2 View panel

characters are not converted to uppercase; in this situation, a lowercase owner name such as “sysibm” does not match “SYSIBM” in the DB2 catalog.

Table space
The table space name for the DB2 object. To further define the object to be processed, you can use this field to optionally specify a table space. If you enter a value in this field:

- The values displayed in the Owner and Name selection lists are limited to those values defined within the specified table space.
- Synonym names are not recognized, nor included in the selection list.

Name
The name of the object name to be processed. You can specify a generic name to select from a list. To specify a synonym in this field, type the name of the synonym and ensure that the Owner field is blank, contains a generic name that includes the current SQLID, or contains the current SQLID. Synonyms are not recognized or included in the table selection list if any of the Table space, Database, or Location fields are non-blank.

If the Translate DB2 object names option is selected, lowercase characters are converted to uppercase. If this option is not selected, lowercase characters are not converted to uppercase; in this situation, a lowercase owner name such as “sysibm” does not match “SYSIBM” in the DB2 catalog.

Start position
Indicates where FM/DB2 is to begin loading rows from the result table into the FM/DB2 editor.

If you specify a value greater than 1 (the default), FM/DB2 skips n-1 rows of the result table before loading rows into the FM/DB2 editor. If the Row count value is ALL (or 0 or *), when the data is displayed you can scroll up to see the rows that were skipped. For a fixed Row count value, you cannot scroll backwards to see any skipped rows.

Row count
Use this option to specify the maximum number of rows to be retrieved from DB2 and loaded into memory for the editor session. You can specify a value in the range 1–999999999 or, to indicate that all rows are to be retrieved, specify ALL, 0, or *. The default value is 100.

An FM/DB2 editor session only applies to those rows fetched from DB2 and loaded into the editor. It does not apply to any unfetched rows. This means that the FIND and CHANGE primary commands only apply to those rows fetched from DB2.

Note: When you are editing large tables, use the value of 0 with caution as it is possible to encounter storage problems. The amount of storage required by the editor depends on both the number of rows retrieved, and the row length. Use the row selection criteria section of the template to limit the number of rows retrieved.

Data set name
The name of a data set that contains, or will contain, the template. It can be a fully-qualified data set name or a pattern. The name may include a member name or name pattern in parenthesis. If the member is specified here, the associated Member field must be empty.

Member
If you specified the name of a partitioned data set (PDS) without including
DB2 View panel

A member name or name pattern in parenthesis in the Data set name field, then you can use this field to specify the member name or a member name pattern.

Template usage
The DB2 View panel provides four options for template processing:

1. **Above**
   Requires that you enter the name of a template data set (and optionally a member name) in the Template section of the panel. If you do not specify a member name, or if you specify a pattern, FM/DB2 displays a member selection list. After you have specified the member containing a template, processing continues using this template in place of the automatically-generated template for the table (for a detailed explanation, see Template Usage option 3).

2. **Previous**
   Uses the last (previously used) template for this table.

3. **Generate from table**
   Generates a template based on the information in the DB2 catalog for the specified table. This is the default setting. If you have specified a member name in the Template section of the panel, it is ignored. FM/DB2 generates the required template in memory.

4. **Generate/Replace**
   Generates a template based on the information in the DB2 catalog for the specified table, and saves this information in the member specified in the Template section of the panel. If the member already exists, it is replaced. Any prior customization is lost.

You can select this option to change the options for the current editor session:

**Edit options**
Enter one of these:

- **A “/” or an “A”**
  Displays the first FM/DB2 editor options panel, prior to displaying the data for the DB2 object being processed.

- **A number in the range 1–8**
  Displays the nth FM/DB2 editor options panel directly where n is the value entered.

You can select the following options by entering a “/” or an “A”:

**Edit template**
Select this option to edit the template before editing the data.

**Re-edit template**
Select this option to control the navigation between panels within the Edit dialog. This option controls the behavior of the F3 and F12 keys when pressed from within an Edit session.

**Create audit trail**

*Note:* "(*)" is appended to the option name for SAF-controlled auditing when required by SAF rules. Controls whether access to DB2 by FM/DB2 is written to the FM/DB2 audit log. This option only appears when SAF-rule controlled auditing is in effect.
FM/DB2 may write audit records to either SMF, or a data set. When SAF rules are used to control FM/DB2 auditing, the decision to write audit records is based on the DB2 object name being processed, and the current userid (TSO logonid). Therefore auditing may be required when processing one DB2 object, but not another. In addition, it is possible that the auditing requirement cannot be changed for a particular DB2 object; this might be different for another DB2 object, and the requirement may be different for other TSO userids.

If you do not have the authority to change the audit setting when processing a particular DB2 object, FM/DB2 prompts you if you enter the wrong setting. If you do have the authority to set the audit setting for a particular DB2 object, FM/DB2 honors whatever value is specified in this field.

Parent panels
- “Primary Option Menu panel” on page 617
- “Tables, Views and Aliases panel” on page 704

Child panels

<table>
<thead>
<tr>
<th>To display this panel...</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>“Table View panel” on page 700</td>
<td>No options selected and press Enter</td>
</tr>
<tr>
<td>“Editor Options (1 of 8) panel” on page 528</td>
<td>Select Edit options</td>
</tr>
<tr>
<td>“Column Selection/Edit panel” on page 407</td>
<td>Select Edit template or Re-edit template</td>
</tr>
</tbody>
</table>

Related tasks
- Chapter 4, “Viewing and changing DB2 data,” on page 91
- “Selecting options on FM/DB2 panels” on page 24
- “Specifying a data set and a member name” on page 30
- “Where you can use templates” on page 4
- Chapter 3, “Working with templates,” on page 49
- “View and Edit options (options 1 and 2)” on page 43

DB2I Primary Option Menu panel

You use the DB2I Primary Option Menu panel to interface to DB2 Interactive.
DB2I Primary Option Menu panel

Panel and field definitions

Select one of the following DB2 functions and press ENTER.

1 SPUFI (Process SQL statements)
2 DCLGEN (Generate SQL and source language declarations)
3 PROGRAM PREPARATION (Prepare a DB2 application program to run)
4 PRECOMPILE (Invoke DB2 precompiler)
5 BIND/REBIND/FREE (BIND, REBIND, or FREE plans or packages)
6 RUN (RUN an SQL program)
7 DB2 COMMANDS (issue DB2 commands)
8 UTILITIES (Invoke DB2 utilities)
D DB2I DEFAULTS (Set global parameters)
X EXIT (Leave DB2I)

F1=HELP F2=SPLIT F3=END F4=RETURN F5=RFIND F6=RCHANGE
F7=UP F8=DOWM F9=SWAP F10=LEFT F11=RIGHT F12=RETRIEVE

Parent panels
- "Primary Option Menu panel" on page 617

Child panels
None.

Related tasks
- Chapter 13, "Interfacing with DB2 Interactive," on page 357

Details panels

These panels are:
- Database Details panel
- Table Space Details panel
- Table Details panel
- View Details panel
- Index Details panel
- Column Details panel
- Synonym Details panel
- Application Plan Details panel
- Package Details panel
- DBRM Details panel
- Distinct Type Details panel
- Function Details panel
- Procedure Details panel
- Trigger Details panel

FM/DB2 displays one of these panels when you specify the I line command against an object displayed in a list of DB2 objects.
Details panels

Child panels

None.

Display Row panel

FM/DB2 displays the Display Row panel if you use the ROW line command on any panel where you can use the ROW line command.

Related tasks

Distinct Types panel

You use the Distinct Types panel to list the distinct types in the DB2 catalog.

Panel and field definitions

The columns that are displayed include the SEL field and columns of SYSIBM.SYSDATATYPES.

Note:

1. The system option, Show all catalog table columns, affects which columns FM/DB2 displays when you use the Object List utility.
2. You can also customize which columns FM/DB2 displays when you use the Object List utility. For details, see the File Manager Customization Guide.

To view other columns on this panel, press the Right function key (F11).

For a complete list of columns, refer to “DB2 catalog tables” in the DB2 for z/OS SQL Reference relevant to your version of DB2.

SEL Line command area.
Distinct Types panel

For a list of the line commands you can use on this panel, see Table 22 on page 250.

Parent panels

- "Object List Utility panel" on page 610
- "Schemas panel" on page 657
- "Functions panel" on page 584

Child panels

<table>
<thead>
<tr>
<th>To display this panel...</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>See &quot;Comment panels&quot; on page 413</td>
<td>Line command COM</td>
</tr>
<tr>
<td>&quot;Create Distinct Type panel&quot; on page 433</td>
<td>Line command CR</td>
</tr>
<tr>
<td>See &quot;Drop panels&quot;</td>
<td>Line command DR</td>
</tr>
<tr>
<td>&quot;Grant privileges panels&quot; on page 587</td>
<td>Line command G</td>
</tr>
<tr>
<td>&quot;Generate SQL From DB2 Catalog panel&quot; on page 585</td>
<td>Line command GEN</td>
</tr>
<tr>
<td>&quot;Details panels&quot; on page 522</td>
<td>Line command I</td>
</tr>
<tr>
<td>&quot;Privileges panels&quot; on page 624</td>
<td>Line command P</td>
</tr>
<tr>
<td>Line command PS</td>
<td></td>
</tr>
<tr>
<td>&quot;Revoke privileges panels&quot; on page 647</td>
<td>Line command R</td>
</tr>
<tr>
<td>&quot;Display Row panel&quot; on page 525</td>
<td>Line command ROW</td>
</tr>
<tr>
<td>&quot;Schemas panel&quot; on page 657</td>
<td>Line command SCH</td>
</tr>
<tr>
<td>&quot;Tables, Views and Aliases panel&quot; on page 704</td>
<td>Line command T</td>
</tr>
<tr>
<td>&quot;Sort Fields panel&quot; on page 679</td>
<td>Primary command SORT</td>
</tr>
</tbody>
</table>

Related tasks

- "Working with object list panels" on page 244
- "Using the line command area (Cmd)" on page 250

Related references

- "SORT primary command" on page 777

Drop panels

These panels are:
- Drop Alias panel
- Drop Database panel
- Drop Distinct Type panel
- Drop Function panel
- Drop Index panel
- Drop Stored Procedure panel
- Drop Synonym Procedure panel
- Drop Table panel
- Drop Table Space panel
- Drop Trigger panel
- Drop View panel

FM/DB2 displays one of these panels when you specify the DR line command against an object displayed in a list of DB2 objects.
You use the appropriate Drop panel when you want to drop a DB2 object.

If you have checked the **Confirm object drop** option on the DB2 Object Functions panel, the Object List Utility panel, or the Explain Utilities panel, before dropping the DB2 object FM/DB2 displays the Drop Confirmation panel.

**Child panels**

- [“Drop Confirmation panel”](#)

**Related tasks**

- [“Dropping DB2 objects”](#) on page 189

---

### Drop Confirmation panel

FM/DB2 displays the Drop Confirmation panel when you have specified that you want to drop a DB2 object and you have checked the **Confirm object drop** option on the DB2 Object Functions panel or the Explain Utilities panel.

To proceed with the drop, press Enter; to cancel the drop, press the Cancel function key (F12).

**Panel and field definitions**

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>Drop Confirmation</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table : "ADDRESS BOOK"

Instruction:

- Press Enter to confirm drop.
- (The table will be dropped)
- (Warning: All dependent objects will also be dropped.)

- Press CANCEL or EXIT to cancel drop.
- Type SQL to generate the SQL for the drop statement only.

---

**Parent panels**

- [“DB2 Object Functions panel”](#) on page 510
- [“DB2 Object Functions panel”](#) on page 510

**Child panels**

None.

**Related tasks**

- [“Dropping DB2 objects”](#) on page 189
Related references
- “DB2 Object Functions panel” on page 510

Edit/Execute SQL (Data Set) panel

You use the Edit/Execute SQL (Data Set) panel to edit and execute SQL statements contained in a data set.

Panel and field definitions

Data set name
The name of the data set or PDS member containing the SQL statements you want to edit. It can be a fully-qualified data set name or a pattern. The name may include a member name or name pattern in parenthesis. If the member is specified here, the associated Member field must be empty.

Member
If you specified the name of a partitioned data set (PDS) without including a member name or name pattern in parenthesis in the Data set name field, then you can use this field to specify the member name or a member name pattern.

Volume serial
The serial number of the volume that contains the data set (if applicable).

Row count
Limits the rows loaded into the FM/DB2 editor when the SQL statement being processed is a SELECT statement. To load all rows for the SELECT statement, specify "*" or ALL in this field.

First column
First data column. When generating SQL statements, FM/DB2 ignores any data to the left of this column number. If you do not specify a value, the statements are assumed to start in column 1.

Last column
Last data column. When generating SQL statements, FM/DB2 ignores any...
Edit/Execute SQL (Data Set) panel

data to the right of this column number. If you do not specify a value, the statements are assumed to continue to the end of the data set record.

Comment chars
Use this option to specify one or two characters (for example “--”) that denote records in the data set that are comment lines. If the specified characters appear as the first nonblank characters in a data set record (and that part of the statement is not in a quoted string), FM/DB2 ignores the record.

Edit data set
FM/DB2 starts an ISPF Edit session for the specified member or data set.

Use this option to use a library of SQL statements that you can maintain ready for execution.

To leave the Edit session, use the Exit function key (F3). Any changes you have made are saved.

Execute SQL from data set
FM/DB2 reads the data set or member, executes the SQL statements that it can extract, returning the results if applicable.

To execute the next SQL statement in the data set or member after FM/DB2 displays a result table or error panel, use the Exit function key (F3).

To return to the Edit and Execute SQL Statements from a Data Set panel without executing the statements, or at any time when an error or result panel has been displayed, use the Cancel function key (F12).

Re-edit data set after execution
After FM/DB2 has completed executing the statements in the data set, it restarts the Edit session (instead of returning to the Edit and Execute SQL Statements from a Data Set panel).

Create audit trail

Note: “(*)” is appended to the option name for SAF-controlled auditing when required by SAF rules.
Controls whether access to DB2 by FM/DB2 is written to the FM/DB2 audit log. This option only appears when SAF-rule controlled auditing is in effect.

FM/DB2 may write audit records to either SMF, or a data set.

When SAF rules are used to control FM/DB2 auditing, the decision to write audit records is based on the DB2 object name being processed, and the current userid (TSO logonid). Therefore auditing may be required when processing one DB2 object, but not another. In addition, it is possible that the auditing requirement cannot be changed for a particular DB2 object; this might be different for another DB2 object, and the requirement may be different for other TSO userids.

If you do not have the authority to change the audit setting when processing a particular DB2 object, FM/DB2 prompts you if you enter the wrong setting. If you do have the authority to set the audit setting for a particular DB2 object, FM/DB2 honors whatever value is specified in this field.

Parent panels
• “SQL Prototyping, Execution and Analysis panel” on page 680
Edit/Execute SQL (Data Set) panel

Child panels

None.

Related tasks

- “Editing and executing SQL statements from a data set” on page 354
- “Specifying a data set and a member name” on page 30
- “SAF-rule controlled auditing” on page 47

Editor Options (1 of 8) panel

You use the seven Editor Options panels (1 to 7) to set your default editor options.

If you get to this panel from the Set Processing Options panel or from the action bar, these options are saved in your ISPF profile and persist from one FM/DB2 session to the next.

Otherwise (if you get to this panel by selecting Edit options on the DB2 View panel or the DB2 Edit panel), then any changes you make to these options last only for the life of the current FM/DB2 editor session.

Panel and field definitions

<table>
<thead>
<tr>
<th>Process Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>Editor Options (1 of 8)</td>
<td>Global Settings</td>
</tr>
</tbody>
</table>

Display Format:

1. Previous
   - Table
   - Single

More Options:

Enter "/" to select option

/ Optimization, large table and data sampling options
7 Table/Single format, prefix area and shadow options
7 Key, index and export options
7 Varying length and nullable column options
7 Enter key and commit processing options
7 Concurrency and locking options

Initial display format

Use this section of the Editor Options panel to specify how editor panels appear at the start of an FM/DB2 editor session. Set this option to one of the following:

1. Previous
   The display format field (Format) on the View or Edit panel is prefilled with the value specified in the previous FM/DB2 editor session. This is the default setting.

2. Table
   The View or Edit panel is displayed in TABLE display format.
3. Single
   The View or Edit panel is displayed in SNGL display format.

Editor CAPS Setting
   Indicates the initial setting of CAPS in an FM/DB2 editor session and whether you can change the setting.

Initially ON (fixed at installation)
   CAPS set to ON. You cannot change the setting by means of the edit options.

Initially OFF (fixed at installation)
   CAPS set to OFF. You cannot change the setting by means of the edit options.

Enter "/" to select option ... Initially ON
   You can change the CAPS setting:
   / Set CAPS to ON
   (blank) Set CAPS to OFF

Create audit trail
   Determines if FM/DB2 generates an audit trail report of all successful modifications to DB2 data made during an Edit session:
   / Audit trail report generated.
   (blank) Audit trail report not generated.

Note:
1. The display of this option depends on whether SAF-rule controlled auditing is in effect and if auditing is to occur for the DB2 system. See “SAF-rule controlled auditing” on page 47.
2. The ability to change this option depends on installation options within the DB2 subsystem to which FM/DB2 is connected. The Create an audit trail option can be:
   • Permanently selected ("/") shown in the option). Audit trail reporting occurs for Edit processing. You cannot change the setting of the option.
   • Permanently deselected (blank shown in the option). Audit trail reporting does not occur for Edit processing. You cannot change the setting of the option.
   • Optional. Type a "/" in the option to generate audit reporting for all Edit processing; otherwise, type a space in the option to suppress audit trail reporting of any Edit processing.

At the end of your FM/DB2 session, the Create an audit trail option that you specify can be saved to your ISPF profile, or it can be set to blank (audit trail report not selected), depending on the installation options within the DB2 subsystem to which FM/DB2 is connected. The audit installation options for the currently connected DB2 system can be determined by selecting option 7 from the HELP pull down menu. When the auditing option is "OPTIONAL,OFF", the Create an audit trail option is always reset to blank (not selected) at the start of an FM/DB2 session. This means that, for an audit log to be written, auditing must be explicitly selected each time a new FM/DB2 session is started. Further, when this option is selected, there is no distinction
More options
These options control whether FM/DB2 displays the further editor options panels.

The FM/DB2 editor options are too numerous to fit on a single panel. To access additional options, select the fields in this section of the panel.

To navigate to the other options panels, press the PF key assigned to NxtPage (PF11 by default). To return to the previous panel, press the PF key assigned to PrvPage (PF10 by default).

You can omit the display of one or more of the additional options panels by de-selecting the appropriate field in the "More options" section of the panel.

Optimization, large table and data sampling options
Indicates whether you want the second Editor Options panel displayed when you navigate between Editor Options panels using the NxtPage function key (F11) or the PrvPage function key (F10).
/  Display the second Editor Options panel
   (blank)  Do not display the second Editor Options panel

Table/Single format, prefix area and shadow options
Indicates whether you want the third Editor Options panel displayed when you navigate between Editor Options panels using the NxtPage function key (F11) or the PrvPage function key (F10).
/  Display the third Editor Options panel
   (blank)  Do not display the third Editor Options panel

Key, index and export options
Indicates whether you want the fourth Editor Options panel displayed when you navigate between Editor Options panels using the NxtPage function key (F11) or the PrvPage function key (F10).
/  Display the fourth Editor Options panel
   (blank)  Do not display the fourth Editor Options panel

Varying length and nullable column options
Indicates whether you want the fifth Editor Options panel displayed when you navigate between Editor Options panels using the NxtPage function key (F11) or the PrvPage function key (F10).
/  Display the fifth Editor Options panel
   (blank)  Do not display the fifth Editor Options panel

Enter key and commit processing options
Indicates whether you want the sixth Editor Options panel displayed when you navigate between Editor Options panels using the NxtPage function key (F11) or the PrvPage function key (F10).
/  Display the sixth Editor Options panel
   (blank)  Do not display the sixth Editor Options panel
Editor Options (1 of 8) panel

Do not display the sixth Editor Options panel

Concurrency and locking options
Indicates whether you want the seventh Editor Options panel displayed when you navigate between Editor Options panels using the NxtPage function key (F11) or the PrvPage function key (F10).

Display the seventh Editor Options panel

Do not display the seventh Editor Options panel

Parent panels
- “Set Processing Options panel” on page 671
- “DB2 View panel” on page 518
- “DB2 Edit panel” on page 505
- “DB2 Browse panel” on page 501
- “Action bar pull-down menu” on page 365

Child panels

<table>
<thead>
<tr>
<th>To display this panel...</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>“Editor Options (2 of 8) panel”</td>
<td>Press the NxtPage function key (F11)</td>
</tr>
</tbody>
</table>

Related tasks
- “View and Edit options (options 1 and 2)” on page 43
- “Setting default processing options” on page 39
- “Setting options for the current FM/DB2 editor session” on page 95
- “Selecting a display format” on page 118
- “Installation settings that affect CAPS” on page 138

Editor Options (2 of 8) panel

You use the Editor Options panel (2 of 7) to set your default editor options.

Note: See the description for the Editor Options (1 of 8) panel (“Editor Options (1 of 8) panel” on page 528) about how long any changes you make to these options last.
Panel and field definitions

Use row count value

This option influences DB2 optimization.

When FM/DB2 prepares an SQL statement to access DB2 data, the number of rows in the result table may affect the method used by DB2 to produce the result table. In some cases, DB2 uses a different, more efficient, access method if it is known that only a few rows of an otherwise very large result table are required. This option enables an additional clause to be added to the SQL statement used by FM/DB2 when accessing DB2. The OPTIMIZE FOR \textit{nnn} ROWS clause is documented in the SQL Reference manual for the appropriate version of DB2.

Enter a number in the range 0-999999 inclusive. This option and the Use row count value option combine in these ways:

Use row count value option selected

The value specified for Row count on the editor function entry panel is used, unless ALL (also 0, \*) is specified. In the latter case, the Optimize for value is considered. A zero value results in no a OPTIMIZE FOR clause. Any other value results in an OPTIMIZE FOR \textit{nnn} ROWS clause being added. \textit{nnn} is the value entered in the Optimize for option.

Use row count value option not selected

The Optimize for value is considered. A zero value results in a no OPTIMIZE FOR clause. Any other value results in an OPTIMIZE FOR \textit{nnn} ROWS clause being added. \textit{nnn} is the value entered in the Optimize for option.

Scrollable cursor type

Controls the access method used by FM/DB2 when processing large DB2 objects.

The FM/DB2 editor can operate in two modes. When there is sufficient memory available, FM/DB2 loads all of the rows for the result table into the editor. For DB2 objects with large result tables, FM/DB2 loads only a
small number of rows initially and keeps only a small number of rows in memory during the editor session. This is achieved using DB2 scrollable cursors. The type of cursor specified can influence the behavior of the editor, particularly in edit mode.

1. **Insensitive**
   An insensitive cursor operates like a point-in-time snapshot of the result table. The number of rows in the result table is fixed, and is not changed by insert, update, or delete operations made to the table by other processes. FM/DB2 always uses an insensitive cursor in browse or view modes.

2. **Sensitive static**
   A sensitive static cursor also has a fixed number of rows in the result table, however delete and update operations made using the cursor are reflected the next time the row is processed by the cursor. In addition, update and delete operations made to the underlying table may cause the row to be no longer visible when processed by the cursor.

   A sensitive static cursor is only used in edit mode.

**Sampling type**
Controls whether data sampling is used.

Data sampling enables a subset of the rows in a DB2 object to be loaded into the editor. The possible values for this option are:

1. **None**
   All rows of the object are eligible for inclusion, this is the default.

2. **Clustered sampling**
   A specified number of rows are loaded into the editor.

3. **Random sampling**
   Each row of the object is considered for inclusion in the editor. A random number generator and a frequency are used to determine whether the row is actually loaded.

When data sampling is selected, you also need to consider these options:

**Sampling limit**
For both clustered and random sampling.

**Initial skip count**
For clustered sampling.

**Include count**
For clustered sampling.

**Skip count**
For clustered sampling.

**Sampling frequency**
For random sampling.

**Sampling seed**
For random sampling.

**Clustered Sampling**
Clustered data sampling divides the data into consecutive groups of rows. The number of rows in each group is the sum of these values:
- Initial skip count
- Include count
- (Final) skip count
Within each group, the number of rows specified in Initial skip count are ignored. The number of rows in Include count are sampled (added to the editor session). Finally, the number of rows specified in the Skip count are ignored. Sampling then recommences with the next group of rows.

These options apply only to clustered data sampling. Any values specified are ignored when clustered data sampling is not in use.

**Initial skip count**
Controls how many rows are initially skipped in each clustered sample group. It can be zero, or a positive integer.

**Include count**
Controls how many rows are sampled in each clustered sample group. It must be a positive integer.

**Skip count**
Controls how many rows are skipped at the end of each clustered sample group. It can be zero, or a positive integer.

### Random Sampling
Random data sampling uses a random number generator, together with the sampling frequency to determine whether a row is included in the sample. These options apply only to random data sampling. Any values specified are ignored when random data sampling is not in use.

**Sampling frequency**
Controls the sampling frequency for random data sampling.

Enter a value in the range 0.000000001 to 0.999999999 inclusive. To sample approximately 10% of rows enter a frequency of 0.1. To sample approximately 1% of rows enter a frequency of 0.01.

**Sampling seed**
A value used to initiate the random number generator used for random data sampling.

Enter an integer in the range 0 to 2147483647 inclusive. A zero value indicates that FM/DB2 should use the system clock to initiate the random number generator. Specify a zero value, when repeatability of data samples is not required. Any non-zero value used to initiate the random number generator ensures that data samples are repeatable, given the same table and other input conditions.

### Parent panels
You can go to the Editor Options (1 of 8) panel from the “parent” panels shown below. To go to the Editor Options (2 of 8) panel, you then press the NxtPage function key (F11).

- “Set Processing Options panel” on page 671
- “DB2 View panel” on page 518
- “DB2 Browse panel” on page 501
- “DB2 Edit panel” on page 505
- “Action bar pull-down menu” on page 365

You can only display the Editor Options (2 of 8) panel if the Optimization, large table and data sampling options option on the Editor Options (1 of 8) panel is selected.
Editor Options (2 of 8) panel

Child panels

<table>
<thead>
<tr>
<th>To display this panel...</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>“Editor Options (1 of 8) panel” on page 528</td>
<td>Press the PrvPage function key (F10)</td>
</tr>
<tr>
<td>“Editor Options (3 of 8) panel”</td>
<td>Press the NxtPage function key (F11)</td>
</tr>
</tbody>
</table>

Related tasks

- “View and Edit options (options 1 and 2)” on page 43
- “Setting default processing options” on page 39
- “Setting options for the current FM/DB2 editor session” on page 95
- “Data sampling” on page 133
- “Large table options” on page 132

Editor Options (3 of 8) panel

You use the Editor Options (3 of 8) panel to set your default editor options.

**Note:** See the description for the Editor Options (1 of 8) panel “Editor Options (1 of 8) panel” on page 528 about how long any changes you make to these options last.

Panel and field definitions

<table>
<thead>
<tr>
<th>Process Options Utilities Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DGF2)</td>
</tr>
</tbody>
</table>

**Table Display Format Options**

Use this section of the Editor Options panel to specify the information you want displayed in table display format (TABL):

**Show data type**

Indicates if you want FM/DB2 to display data type information:

```
/ Show the data type for each column displayed in table display format.
```

<table>
<thead>
<tr>
<th>Command ==</th>
<th>Process Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>F1=Help</td>
<td>F2=Split</td>
<td>F3=Exit</td>
<td>F7=Backward</td>
</tr>
<tr>
<td>F8=Forward</td>
<td>F9=Swap</td>
<td>F10=PrvPage</td>
<td>F11=NxtPage</td>
</tr>
<tr>
<td>F12=Cancel</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Editor Options (3 of 8) panel

FM/DB2 displays an additional line (immediately below the line containing the column reference numbers, if displayed) showing the data type for each column.

(blank)
Do not show data type information. This is the default action.

Show scale line
Indicates if you want FM/DB2 to display scale line information:

/   Show a scale line for each column displayed in table display format.

The scale shows the left and right boundary of the data for the column and, for non-numeric data types, positioning information.

FM/DB2 displays the scale line immediately above the first line of data. The scale line also displays information about primary and foreign keys, and unique indexes.

(blank)
Do not show scale line information. This is the default setting.

Show column number
Indicates if you want FM/DB2 to display column number information.

/   Show the column number (as defined in the DB2 catalog) for each column displayed in table display format.

FM/DB2 displays the column numbers immediately below the line containing the column names.

(blank)
Do not show column number information. This is the default setting.

Prefix Area
These options control the prefix area in an FM/DB2 editor session.

Prefix length
Controls the width of the prefix area in an FM/DB2 editor session.

The FM/DB2 editor can display a prefix area on either the left, or the right of the screen. The width of the prefix area can be 6, 7, 8 or 9 characters. The default is 6, specified by typing A in the field. You can type a different value to set the width of the prefix area to your preference. Note that the prefix areas does not appear in an FM/DB2 browse session, and only appears in an FM/DB2 view or edit session when the Display prefix area option is selected. You can alter the prefix length from within an editor session using the PREFIX editor primary command.

Display prefix area
Controls whether the prefix area is displayed in an FM/DB2 edit or view session.

The FM/DB2 editor can optionally display a prefix area, where prefix commands may be entered.

/   Display the prefix area.
(blank)

Do not display the prefix area.

You can turn the prefix area on or off from within an FM/DB2 editor session (edit and view only) by issuing the PREFIX editor primary command.

**Show prefix area on the right**

Controls the location of the FM/DB2 editor's prefix area.

The FM/DB2 editor can display the prefix area on either the left, or the right of the screen.

/ Display the prefix area on the right of the screen.

((blank)

Display the prefix area on the left of the screen.

**Note:**

1. The prefix area does not appear in an FM/DB2 browse session.
2. This option has no effect unless the Display prefix area option is selected.
3. You can move the prefix area, from within the FM/DB2 editor, by issuing the PREFIX LEFT or PREFIX RIGHT editor primary commands.

**Show SQLCODE**

Controls whether the FM/DB2 editor shows the SQLCODE in the prefix area, when an insert, update, or delete operation for a row fails.

FM/DB2 processes changes to data in the edit session one row at a time. If DB2 returns an error when the change is presented to DB2, FM/DB2 shows either =ERR or an SQLCODE in the prefix area for the row.

/ FM/DB2 editor shows the SQLCODE in the prefix area.

(Blank)

FM/DB2 editor shows the =ERR in the prefix area.

**Single display Format Options**

Use this section of the Editor Options panel to specify the information you want displayed in single display format (SNGL).

**Show data type**

Indicates if you want FM/DB2 to display data type information:

/ Show the data type for each column displayed in single display format.

FM/DB2 displays an additional column (immediately to the right of the column containing the name of the column) showing the data type for each column.

**Note:** When you are displaying data in single display mode, you can toggle the display of data type information by using the TYPE primary command.

(Blank)

Do not show data type information. This is the default setting.
Editor Options (3 of 8) panel

Show keys, indexes
Indicates if you want FM/DB2 to display information showing if column is a part of a key or index:

/ Show an additional column, 4 characters wide, indicating if the column is part of a key or index.

Note: When you are displaying data in single display mode, you can toggle the display of index information by using the INDEXINF primary command.

(blank)
Do not show key or index information. This is the default setting.

Show column number
Indicates if you want FM/DB2 to display column number information:

/ Show the column number (as defined in the DB2 catalog) for each column displayed in single display format.

FM/DB2 displays an additional column (the leftmost on the panel) containing the column numbers.

Note:
1. When you are displaying data in single display mode, you can toggle the display of column numbers by using the REFS primary command.
2. In single display mode, when FM/DB2 displays the column number, it also displays the column heading “Rdf” (as the second column heading from the left). The “Rdf” column is unused by FM/DB2 and can be ignored.

(blank)
Do not show column number information. This is the default setting.

Left justify numerics
Indicates if you want FM/DB2 to display numeric data left-justified, in single display format:

/ Data for numeric columns is displayed left-justified. That is, the first characters of any numbers are shown at the extreme left of the column.

(blank)
Data for numeric columns is displayed right-justified. That is, the data is displayed so that the last character of each number is shown at the extreme right of the column.

Show char position range
Indicates that you want FM/DB2 to show the start and end character positions for long character columns, in SNGL display mode.

/ When the data for a character column requires two or more rows on the SNGL display, the second and subsequent rows include a character position range, for example, ssss-nnnn. ssss is the character position of the first
Editor Options (3 of 8) panel

character shown in the row. \textit{nnnn} is the character position of the last character shown in the row.

(blank)

The character position range is not shown.

You can toggle the character position on and off from within the editor using the CHARPOS editor primary command.

Show excluded shadow lines

Controls whether a shadow line, which indicates one or more excluded rows, is displayed.

The FM/DB2 editor allows rows to be excluded from display, using the EXCLUDE editor primary command, or the X (and its variations) editor prefix commands. Like the ISPF editor, excluded rows are shown using a shadow line. Each shadow line marks the location of one or more contiguous excluded rows. However, unlike the ISPF editor, the excluded shadow lines can be hidden from display by de-selecting this option.

\begin{verbatim}
/ Display excluded shadow lines.
\end{verbatim}

(blank)

Do not display excluded shadow lines.

You can change the display status of excluded rows, from within the FM/DB2 editor, by issuing the SHADOW X OFF or SHADOW X ON commands.

Parent panels

You can go to the Editor Options (1 of 8) panel from the “parent” panels shown below. To go to the Editor Options (3 of 8) panel, you then press the NxtPage function key (F11) twice.

- "Set Processing Options panel" on page 671
- "DB2 View panel" on page 518
- "DB2 Browse panel" on page 501
- "DB2 Edit panel" on page 505
- "Action bar pull-down menu" on page 365

You can only display the Editor Options (3 of 8) panel if the Table/Single format, prefix area and shadow options option on the Editor Options (1 of 8) panel is selected.

Child panels

<table>
<thead>
<tr>
<th>To display this panel...</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>“Editor Options (2 of 8 panel” on page 531</td>
<td>Press the PrvPage function key (F10)</td>
</tr>
<tr>
<td>“Editor Options (4 of 8 panel” on page 540</td>
<td>Press the NxtPage function key (F11)</td>
</tr>
</tbody>
</table>

Related concepts

- "TABL display format” on page 118
- "SNGL display format” on page 121

Chapter 15. FM/DB2 panels and fields 539
Editor Options (3 of 8) panel

Related tasks
- “View and Edit options (options 1 and 2)” on page 43
- “Setting default processing options” on page 39
- “Setting options for the current FM/DB2 editor session” on page 95
- “Selecting a display format” on page 118
- “Managing the prefix area” on page 129
- “Showing or hiding excluded rows” on page 127

Editor Options (4 of 8) panel

You use the Editor Options (4 of 8) panel to set your default editor options.

Note: See the description for the Editor Options (1 of 8) panel “Editor Options (1 of 8) panel” on page 528 about how long any changes you make to these options last.

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>Editor Options (4 of 8)</td>
<td>Global Settings</td>
<td></td>
</tr>
</tbody>
</table>

Primary and Foreign Keys, Indexes:
Enter "/" to select option
/ Allow updates to primary key
7 Show primary key
7 Show foreign key
7 Show index indicators

Export Command:
Enter "/" to select option
- Show export options

Command ===>
F1=Help F2=Split F3=Exit F7=Backward F8=Forward F9=Swap
F10=PrvPage F12=Cancel

Allow updates to primary key
Indicates if you can edit the primary key information:
/ Changes to primary key information allowed.
(blank) Changes to primary key information not allowed.

Show primary key
Indicates if you want FM/DB2 to display primary key information:
/ Include information showing the columns that comprise the primary key for the table.

In table display format, FM/DB2 shows the primary key information only if you have also specified that the scale line is to be displayed (see Show scale line in “Editor Options (3 of 8) panel” on page 533). The scale line shows a “P” in the first position of each column that forms part of the table’s primary key.
Editor Options (4 of 8) panel

In single display format, if the **Show keys, indexes** option is selected (see "Editor Options (3 of 8) panel" on page 535), FM/DB2 displays an additional column, immediately to the left of the **Column name** column (and immediately to the right of the **Column number** column, if displayed). This column is 4 characters wide. The presence of a “P” in the first position of the column indicates that the column is part of the table’s primary key.

*Blank*

Do not include information showing the columns that comprise the primary key for the table. This is the default setting.

**Show foreign key**

Indicates if you want FM/DB2 to display foreign key information:

/ Include information showing the columns that are part of a foreign key.

In table display format, FM/DB2 shows the foreign key information only if you have also specified that the scale line is to be displayed (see "Editor Options (3 of 8) panel" on page 535). The scale line shows an “F” in the fourth position of each column that forms part of a foreign key.

In single display format, if the **Show keys, indexes** option is selected (see "Editor Options (3 of 8) panel" on page 535), FM/DB2 displays an additional column, immediately to the left of the **Column name** column (and immediately to the right of the **Column number** column, if displayed). This column is 4 characters wide. The presence of an “F” in the fourth position of the column indicates that the column is part of a foreign key.

*Blank*

Do not include information showing the columns that are part of a foreign key. This is the default setting.

**Note:**

1. For tables with multiple foreign keys defined, the indicator character cannot be used to distinguish between columns that are part of the same foreign key or part of different foreign keys.

2. The foreign key information is only displayed if it is stored in the template. If the **Retrieve foreign key information when building templates** system option was not selected when the template for the object was built, no foreign key indicators are displayed.

**Show index indicators**

Indicates if you want FM/DB2 to display index information:

/ Include information showing the columns that are part of an index.

In table display format, FM/DB2 shows the index information only if you have also specified that the scale line is to be displayed (see "Editor Options (3 of 8) panel" on page 535). In the second position of the scale line, a “U” is displayed for each column that forms part of a unique index. In the third position of the scale line, an “N” is displayed for each column that forms part of a non-unique index. See "Table Edit panel" on page 691.
Editor Options (4 of 8) panel

In single display format, if the Show keys, indexes option is selected (see "Editor Options (3 of 8) panel" on page 535), FM/DB2 displays an additional column, immediately to the left of the Column name column (and immediately to the right of the Column number column, if displayed). This column is 4 characters wide. In the second position of this column, a “U” is displayed for each column that forms part of a unique index. In the third position of this column, an “N” is displayed for each column that forms part of a non-unique index.

Do not include information showing the columns that are part of an index. This is the default setting.

Note: For tables with multiple foreign keys defined, the indicator character cannot be used to distinguish between columns that are part of the same index or part of different indexes.

Show export options

The EXPORT primary command can be issued in an FM/DB2 editor session to export data to a external file. Issuing the EXPORT command can optionally display the first of the export options panels, where you can specify the data format for the export operation.

/ The EXPORT primary command dialog displays the first of the export options panels.

The EXPORT primary command dialog does not display the first of the export options panels.

Parent panels

You can go to the Editor Options (1 of 8) panel from the “parent” panels shown below. To go to the Editor Options (4 of 8) panel, you then press the NxtPage function key (F11) three times.

- “Set Processing Options panel” on page 671
- “DB2 View panel” on page 518
- “DB2 Browse panel” on page 501
- “DB2 Edit panel” on page 505
- “Action bar pull-down menu” on page 365

You can only display the Editor Options (4 of 8) panel if the Key, index and export options option on the Editor Options (1 of 8) panel is selected.

Child panels

<table>
<thead>
<tr>
<th>To display this panel...</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>“Editor Options (3 of 8) panel” on page 535</td>
<td>Press the PrvPage function key (F10)</td>
</tr>
<tr>
<td>“Editor Options (5 of 8) panel” on page 543</td>
<td>Press the NxtPage function key (F11)</td>
</tr>
</tbody>
</table>

Related tasks

- “View and Edit options (options 1 and 2)” on page 43
- “Setting default processing options” on page 39
You use the Editor Options (5 of 8) panel to set your default editor options.

**Note:** See the description for the Editor Options (1 of 8) panel ("Editor Options (1 of 8) panel" on page 528) about how long any changes you make to these options last.

### Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>FM/DB2 (DFQ2)</td>
<td>Editor Options (5 of 8)</td>
<td>Global Settings</td>
</tr>
</tbody>
</table>

**Varying Length Columns:**
- Enter "/" to select option
- Disable input delimiter
- Show end of string
- Remove trailing spaces

**String Delimiters:**
- Input . . . . . #
- Display . . . . .$

**Null Column Indicators:**
- Input . . . . . 0
- Display . . . . .$

### Disable input delimiter

This option is used to disable processing for the Input string delimiter. By default this option is not selected. When a change is made to a varying field, FM/DB2 scans the entire varying field from left to right, looking for the first occurrence of the Input string delimiter character. If an occurrence is found, the field is truncated at the character before the location of the Input string delimiter.

Consider disabling the Input string delimiter when:
- The DB2 object being processed contains VARCHAR fields that cannot be displayed in full, without scrolling. In this situation a change to the field may trigger silent truncation when the input string delimiter is present in a part of the field that is not currently visible on the display.
- A VARCHAR field is used to store data other than readable text, for example binary data, or data that has an internal format. In this situation the data may contain a character that matches the input string delimiter character, leading to unwanted truncation if the data in that field is changed.
Show end of string
This option only applies to the processing of varying-length (VARCHAR) input fields. Specifies if the end of a varying-length string is to be indicated using the delimiter specified in the String delimiters: Display option:

/  Show the end of a varying-length string using the delimiter specified in the String delimiters: Display option. If the string is of maximum length, the delimiter character is not shown.

(blank)  Do not show the end of a varying-length string.

Remove trailing spaces
This option only applies to the processing of varying-length (VARCHAR) input fields. When you enter a value that is less than the maximum length for the column, determines if FM/DB2 removes trailing spaces:

/  Remove trailing spaces and store the specified value using the minimum length string.

(blank)  Do not remove trailing spaces.

String Delimiters
These options control the end of string delimiter when entering and displaying data in a varying column.

Input  This option only applies to the processing of varying-length (VARCHAR) input fields. For input, FM/DB2 sets the end-of-string delimiter character to this value. If you enter data that requires one or more trailing spaces in a varying-length field, you can use this option to provide a character of your choice to indicate the end of the string. The default value is the number sign (#).

Valid values: any character.

Display  This option only applies to the processing of varying-length (VARCHAR) input fields. FM/DB2 sets the end-of-string delimiter character, for display, to a this value. If you have also selected Show end of string, FM/DB2 displays the value in the field, terminated with the delimiter character you have selected. If the column is of maximum length, the delimiter character is not shown. The default value is the “less than” sign (<).

Valid values: any character.

Convert spaces to DB2 null
When any column that allows null values is changed so that it contains only spaces, the value of the column is set to null.

If the column is defined as NOT NULL, the column remains unchanged.

Null Column Indicators
These options allow the specification of characters to represent the DB2 null value when entering and displaying data.

Input  A single character which, when entered in the first byte of an input field, causes FM/DB2 to update the value for the column to NULL. The default value is the “at.” symbol (@).
Editor Options (5 of 8) panel

Valid values: any character

If the column allows null values, when you enter this character in the first byte of the input field, FM/DB2 updates the value for the column to NULL. Any non-blank character following the indicator invalidates the action.

If the column is defined as NOT NULL, FM/DB2 treats the character in the same way as any other input character.

Display
A single character indicating that the column contains a null value (as opposed to, for example, blanks). The default value is the “¢” symbol.

Valid values: any character

Parent panels

You can go to the Editor Options (1 of 8) panel from the “parent” panels shown below. To go to the Editor Options (5 of 8) panel, you then press the NxtPage function key (F11) four times.

- “Set Processing Options panel” on page 671
- “DB2 View panel” on page 518
- “DB2 Browse panel” on page 501
- “DB2 Edit panel” on page 505
- “Action bar pull-down menu” on page 365

You can only display the Editor Options (5 of 8) panel if the Varying length and nullable column options option on the Editor Options (1 of 8) panel is selected.

Child panels

To display this panel... Use/do this

“Editor Options (4 of 8) panel” on page 540 Press the PrvPage function key (F10)

“Editor Options (6 of 8) panel” Press the NxtPage function key (F11)

Related tasks

- “View and Edit options (options 1 and 2)” on page 43
- “Setting default processing options” on page 39
- “Setting options for the current FM/DB2 editor session” on page 95
- “Varying-length columns” on page 137
- “Null values and VARCHAR processing” on page 136

Editor Options (6 of 8) panel

You use the Editor Options (6 of 8) panel to set your default editor options.

Note: See the description for the Editor Options (1 of 8) panel “Editor Options (1 of 8) panel” on page 528 about how long any changes you make to these options last.
## Enter key usage

Set this option to one of the following:

1. **No action**
   
   This is the default setting. No special action is taken when the Enter key is pressed and no other operations are pending. Any changes made during an Edit session are only presented to DB2 when an explicit SAVE command is issued, or at the end of the Edit session.

2. **Save data**
   
   All pending changes are submitted to DB2 for processing whenever the Enter key is pressed (and no other operations are pending). Successful changes are not committed until the end of the Edit session, unless the **Commit changes when Save issued** option is selected. Rows for which there was an unsuccessful change are marked with “=ERR”, “ERR R”, or the SQLCODE in the prefix area (see **Show SQLCODE** on page 537).

3. **Save data and commit changes**
   
   All pending changes are submitted to DB2 for processing whenever the Enter key is pressed (and no other operations are pending). Rows for which there was an unsuccessful change are marked with “=ERR”, “ERR R”, or the SQLCODE in the prefix area (see **Show SQLCODE in prefix area** on page 537). FM/DB2 issues an explicit DB2 commit at the end of the save process; however, this only occurs if:
   - No save errors were encountered, or
   - **Commit when no save errors** is turned off.

Committing the changes to DB2 is an irrevocable process.

### Commit when save issued

This is the default action. FM/DB2 issues an explicit DB2 commit at the end of the save process; however, this only occurs if:

- No save errors were encountered, or
Editor Options (6 of 8) panel

- **Commit when no save errors** is turned off.
  
  Note that the Enter key processing options **Save data** and **Save data and commit changes** behave as though an explicit save command had been issued when the Enter key was pressed.

**Commit when no save errors**
The default action for this option is to always commit at the end of the save process. When this option is selected, FM/DB2 only issues an explicit DB2 commit if there were no errors encountered during the save process.

Committing the changes to DB2 is an irrevocable process.

**Auto-commit count**
The default action, indicated by a zero value, is that no automatic DB2 commit points are during the editor SAVE process. When a positive value is specified, FM/DB2 issues an automatic DB2 commit point whenever there have been value successful changes during the SAVE process (where value is the value specified for **Auto-commit count**). Specifying a non-zero value can cause multiple DB2 commit points to be issued during the SAVE process.

Example:
With an auto-commit count of 2 and 11 successful changes processed during a SAVE process, five DB2 commit points are issued. The changes for the last row processed remain uncommitted.

**Parent panels**
You can go to the Editor Options (1 of 8) panel from the “parent” panels shown below. To go to the Editor Options (6 of 8) panel, you then press the NxtPage function key (F11) five times.

- “Set Processing Options panel” on page 671
- “DB2 View panel” on page 518
- “DB2 Browse panel” on page 501
- “DB2 Edit panel” on page 505
- “Action bar pull-down menu” on page 365

You can only display the Editor Options (6 of 8) panel if the **Enter key and commit processing options** option on the Editor Options (1 of 8) panel is selected.

**Child panels**

<table>
<thead>
<tr>
<th>To display this panel…</th>
<th>Use/do this</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>“Editor Options (5 of 8) panel” on page 543</td>
<td>Press the PrvPage function key (F10)</td>
<td></td>
</tr>
<tr>
<td>“Editor Options (7 of 8) panel” on page 548</td>
<td>Press the NxtPage function key (F11)</td>
<td></td>
</tr>
</tbody>
</table>

**Related tasks**

- “View and Edit options (options 1 and 2)” on page 43
- “Setting default processing options” on page 39
- “Setting options for the current FM/DB2 editor session” on page 95
- “Ending an FM/DB2 editor session” on page 101
You use the Editor Options (7 of 8) panel to set your default editor options.

**Note:** See the description for the Editor Options (1 of 8) panel (“Editor Options (1 of 8) panel” on page 528) about how long any changes you make to these options last.

### Panel and field definitions

**Concurrency options**
For a detailed explanation about this subject, refer to the DB2 UDB for z/OS Administration Guide.

Concurrency options affect the way DB2 takes row and page locks for data in a DB2 table. Locking options affect the way DB2 takes table locks. When a table locking option is specified, it overrides any concurrency option.

In general terms, specifying a higher value for the concurrency option reduces the likelihood of deadlocks when editing an object, at the expense of other users' ability to access and change data.

FM/DB2 provides options that can minimize DB2 locking, and the duration of the DB2 locks taken, when FM/DB2 accesses DB2 and retrieves data for an FM/DB2 editor session.

FM/DB2 also provides options to lock the table to prevent other users from either updating or reading data. The locking options should be used with caution.

Note that both the concurrency and table locking options only apply to the initial fetch of data from DB2. If either of the commit options **Commit changes when save issued** or **Commit changes when no save errors** is specified, each commit point releases any locks held by DB2 against the table. For more information, see the description about **Enter Key Processing, Commit Options** in “Editor Options (6 of 8) panel” on page 545.
After each commit point, FM/DB2 re-locks the table if a table-locking option has been specified; however, other DB2 users may access the table between the time of the commit point and FM/DB2 re-locking the table.

**Note:**
1. Use table locking options with caution.
2. For the table locking option to be effective, the DB2 object must be a table. It is not possible to lock a view.
3. Any concurrency option applies to the initial fetch of data from DB2 only.
4. Any concurrency or locking option persists only until the next commit point. See below for more information.
5. Set the **Concurrency** option to 0.

FM/DB2 provides options for accessing data in “read-only” mode, and for reducing the life of the DB2 locks taken when data is retrieved from DB2. It also provides control over the locking mechanism used when data is retrieved from DB2.

**Read-only access**

If you select this option, FM/DB2 accesses data from DB2 in “read-only” mode, by adding the FOR FETCH ONLY clause to the SQL statement used to retrieve data from DB2. By default, this option is not selected.

Selecting this option does not prevent edit operations from occurring.

Note that, for an FM/DB2 Browse session, the FOR FETCH ONLY clause is always added to the SQL statement used to retrieve data from DB2.

**Commit after data fetch**

If you select this option, FM/DB2 issues an explicit COMMIT once the data from the FM/DB2 Edit session has been retrieved from DB2. This has the effect of releasing all DB2 locks taken during the retrieval of data for the FM/DB2 Edit session. It also releases any explicit table locks taken when a locking option of shared or exclusive is specified. By default, this option is selected.

This option is recommended in high concurrency environments, where it is important to minimize the duration of any DB2 locks held.

Selecting this option does not prevent edit operations from occurring; however, it increases the likelihood of errors when changed data is presented to DB2 for validation. This is because other processes are free to change the DB2 data for which a copy is stored in the FM/DB2 Edit session. Therefore, when a change made in the FM/DB2 Edit session is presented to DB2, the original data might have changed, resulting in various SQL errors.

Note that, for an FM/DB2 Browse session, FM/DB2 always issues an EXPLICIT commit once the data for the Browse session has been retrieved from DB2.

**SQL Cursor Concurrency Options (Concurrency)**

FM/DB2 can add a WITH clause to the SQL statement used to fetch data from DB2. When a WITH clause is specified, DB2 takes locks (of varying
levels) against either rows or pages within the table, as the data is
accessed. The following options are possible.

0. Uncommitted read
  DB2 locking is disabled. The data retrieved from DB2 may be
  inconsistent if another process is rolled back after the data access.
  Use this option in a high concurrency environment.

1. No with clause
  This is the default value and recommended for most application
  programs. FM/DB2 does not add a WITH clause to the SELECT
  statement for the table. Any rows retrieved by FM/DB2 initially
  have a S(hare) lock. With this option it is possible that other
  processes can update the table you are editing.

2. Cursor stability
  A row or page lock is held only long enough for the cursor to
  move to another row or page.

3. Read stability
  A row or page lock is held for rows or pages that are read by the
  editor at least until the next commit point.

4. Read stability keep locks
  A row or page lock is held for rows or pages that are read by the
  editor at least until the next commit point. The Read stability
  option takes an “S” or “U” lock, while this option takes an “X”
  lock on affected rows or pages.

5. Repeatable read
  A row or page lock is held for all accessed rows, qualifying or not,
  at least until the next commit point. FM/DB2 can return to a
  previously read page and read the same row again, and the data is
  not changed and no new rows are inserted.

6. Repeatable read keep locks
  This option is similar to the Repeatable read option, except that
  FM/DB2 takes an “X” lock, rather than a “S” or “U” lock.

SQL Cursor Concurrency Options (Keep locks option)
FM/DB2 can add a WITH clause to the SQL statement used to fetch data
from DB2. When a WITH clause is specified, DB2 takes locks (of varying
levels) against either rows or pages within the table, as the data is
accessed. The following options are possible.

1. Use/Keep share locks
  FM/DB2 does not issue an explicit table lock for the object being
  edited. Locking of rows and pages within the object is managed by
  DB2 according to FM/DB2 concurrency options and DB2
  installation and object definition options. This is the default setting.

2. Use/Keep update locks
  FM/DB2 issues a LOCK TABLE IN SHARE MODE statement
  before reading the data from DB2. This restricts all other users to
  read-only operations on the object. If either the Enter key option
  Save data and commit changes, or the Commit changes when
  SAVE issued option is selected, any commit point take during the
  Edit session releases the table lock. FM/DB2 re-issues the lock
  immediately after the commit point. However, other processes can
  access the table between the time of the commit point and
  FM/DB2 re-locking the table.
3. Use/Keep exclusive locks
   FM/DB2 issues a LOCK TABLE IN EXCLUSIVE MODE statement before reading the data from DB2. This prevents all other users from accessing the object, unless the other process is running with an isolation level of uncommitted read. If either the Enter key option Save data and commit changes, or the Commit changes when SAVE issued option is selected, any commit point taken during the Edit session releases the table lock. FM/DB2 re-issues the lock immediately after the commit point. However, other processes can access the table between the time of the commit point and FM/DB2 re-locking the table.

Parent panels

You can go to the Editor Options (1 of 8) panel from the “parent” panels shown below. To go to the Editor Options (7 of 8) panel, you then press the NxtPage function key (F11) six times.

- “Set Processing Options panel” on page 671
- “DB2 View panel” on page 518
- “DB2 Browse panel” on page 501
- “DB2 Edit panel” on page 505
- “Action bar pull-down menu” on page 365

You can only display the Editor Options (7 of 8) panel if the Concurrency and locking options option on the Editor Options (1 of 8) panel is selected.

Child panels

To display this panel... Use/do this
- “Editor Options (6 of 8) panel” on page 545 Press the PrvPage function key (F10)

Related tasks

- “View and Edit options (options 1 and 2)” on page 43
- “Setting default processing options” on page 39
- “Setting options for the current FM/DB2 editor session” on page 95

Editor Options (8 of 8) panel

You use the Editor Options (8 of 8) panel to set your default editor options.

Note: See the description for the Editor Options (1 of 8) panel “Editor Options (1 of 8) panel” on page 528 about how long any changes you make to these options last.
Show part/all data

When the DB2 object contains LOB (BLOB, CLOB or DBCLOB), or XML columns, you can display some, or possibly all of the data within those columns. The default action is not selected, in which case no data is displayed. The LOB and XML column appears in the editor as a column heading only, with no data displayed.

To display some, or possibly all, of the data in a LOB or XML column, select the option. You must also specify a maximum number of characters to displayed using the Max characters option.

Note that you can display all of the data in a LOB or XML column using the LOBBRWS, LOBVIEW, LOBEDIT, XMLBRWS, XMLVIEW and XMLEDIT editor primary commands.

Max characters

This option works in conjunction with the Show part/all data option. You can specify a value in the range 0-1000. When the Show part/all data option is selected, and a non-zero value is specified for the Max characters option, FM/DB2 retrieves and displays that many characters of the LOB or XML data in the editor session.

Initial Display Format

Use this section of the Editor Options panel to specify how editor panels appear at the start of an FM/DB2 LOB editor session. This option works in conjunction with the Split Lobs into pieces using option. Set this option to one of the following:

1. Previous
   The display format field (Format) on the LOB editor panel is pre-filled with the value specified in the previous FM/DB2 LOB editor session.

2. Table
   The LOB Editor panel is displayed in TABLE display format. When this option is selected it is suggested that the Split LOBs into pieces using option be set to “Table display size”.

3. Single
   The LOB Editor panel is displayed in SNGL display format. This is
Editor Options (8 of 8) panel

the default setting. When this option is selected it is suggested that
the Split LOBs into pieces using option be set to “Single display
size”.

Split LOBs into pieces using
This option determines the size of the pieces, when FM/DB2 displays a
LOB column in the LOB editor. LOB data can be very large, and a 3270
display is not well suited to displaying large columns directly, so the LOB
editor breaks the LOB column into pieces, and displays those pieces as
though the LOB column was actually a table with a single varying column.

The length of the varying column is the maximum length of any piece.
This option works in conjunction with the Initial Display Format option
(for LOBs) as follows:
1. Table Display size
   The size of each piece is approximately the same as the number of
   characters displayed on a single line in TABL display format. The
   actual number of characters displayed depends on the terminal
   characteristics. This may not be the best choice when processing
   large objects because of the large number of pieces required.
2. Single Display size
   The size of each piece is approximately the same as the number of
   characters displayed on the terminal in SNGL display format. The
   actual number of characters displayed depends on the terminal
   characteristics. On a 24x80 terminal the value is approximately 850;
on a 48x132 terminal the value is approximately 4250.
   This allows large amounts of data to be displayed at one time –
   when compared with the Table Display size option.
   Usage tip 1: Turn on the character position range indicator by
   issuing the CHARPOS command. This shows the starting and
   ending character position for each row of data on the display.
   Usage tip 2: You can use the NEXT (PF11 by default) and
   PREVIOUS (PF10 by default) PFKeys to scroll through the LOB
   data. The character position range indicator values show the
   absolute position from the start of the LOB, rather than the relative
   position within each piece of the LOB.
   You can specify your preference for the piece size by specifying ‘3’.
   Enter your preference for piece size in the Manual input field. A
   value in the range 60-32760 is required.

Parent panels
You can go to the Editor Options (1 of 8) panel from the “parent” panels shown
below. To go to the Editor Options (8 of 8) panel, you then press the NxtPage
function key (F11) seven times.
- “Set Processing Options panel” on page 671
- “DB2 View panel” on page 518
- “DB2 Browse panel” on page 501
- “DB2 Edit panel” on page 505
- “Action bar pull-down menu” on page 365

You can only display the Editor Options (8 of 8) panel if the Concurrency and
locking options option on the Editor Options (1 of 8) panel is selected.
Enter and Execute DB2 Commands panel

You use the Enter and Execute DB2 Commands panel to enter DB2 commands and view the output returned by DB2.

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFE2)</td>
<td>Enter and Execute DB2 Commands</td>
<td>DB2 command options:</td>
<td></td>
</tr>
<tr>
<td>Line count</td>
<td>Number of lines of output</td>
<td>DB2 Command:</td>
<td></td>
</tr>
</tbody>
</table>

Command ===> Scroll PAGE
F1=Help  F2=Split  F3=Exit  F7=Backward  F8=Forward  F9=Swap
F12=Cancel

Line count
Only applies to DISPLAY THREAD and DISPLAY DATABASE commands. This figure controls the number of lines to be returned and is included in the LIMIT parameter for the command which has been entered.

DB2 Command
The DB2 command to be issued. This command may or may not be prefixed by a hyphen (-). If it is not prefixed by a "-", a "-" is added.

Parent panels
- “Primary Option Menu panel” on page 617
- Any panel where the Process pull-down menu is available.

Child panels
None.
Related tasks
- “Issuing DB2 commands from within an FM/DB2 session” on page 156

Enter, Execute and Explain SQL Statements panel

You use the Enter, Execute and Explain SQL Statements panel to enter freeform SQL statements, run freeform SQL statements, or obtain information about an SQL statement.

Panel and field definitions

<table>
<thead>
<tr>
<th>Process Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>Enter, Execute and Explain SQL</td>
<td></td>
</tr>
</tbody>
</table>

**SELECT Options:**
- Row count: ALL

**EXPLAIN Options:**
- Query number

- Enter "/*" to select option
- / Show results
- _ Enter "/" to select option
- _ Create audit trail (*)

SQL statement:

**Command ==> Scroll PAGE**
F1=Help F2=Split F3=Exit F7=Backward F8=Forward F9=Swap F12=Cancel

**Row count**
Limits the rows loaded into the FM/DB2 editor when the SQL statement being processed is a SELECT statement. To load all rows for the SELECT statement, specify "/*" or ALL in this field.

**Query number**
A number to help identify the results of this explanation. If you leave this field blank, FM/DB2 generates a query number for you in the format YYYYMMDDss, where ss is a sequence number.

**Show results**
If selected, FM/DB2 attempts to display the relevant rows that DB2 has inserted in the plan table as a result of explaining the statement.

**Create audit trail**

**Note:** "(*)" is appended to the option name for SAF-controlled auditing when required by SAF rules.
Controls whether access to DB2 by FM/DB2 is written to the FM/DB2 audit log. This option only appears when SAF-rule controlled auditing is in effect.

FM/DB2 may write audit records to either SMF, or a data set.
When SAF rules are used to control FM/DB2 auditing, the decision to write audit records is based on the DB2 object name being processed, and
Enter, Execute and Explain SQL Statements panel

the current userid (TSO logonid). Therefore auditing may be required when processing one DB2 object, but not another. In addition, it is possible that the auditing requirement cannot be changed for a particular DB2 object; this might be different for another DB2 object, and the requirement may be different for other TSO userids.

If you do not have the authority to change the audit setting when processing a particular DB2 object, FM/DB2 prompts you if you enter the wrong setting. If you do have the authority to set the audit setting for a particular DB2 object, FM/DB2 honors whatever value is specified in this field.

SQL statement
Freeform SQL statement entry area. If you want to enter more than one SQL statement, you must separate the statements with a semicolon (;).

Parent panels
- “SQL Prototyping, Execution and Analysis panel” on page 680

Child panels
None.

Related tasks
- “Entering SQL statements” on page 351
- “Executing SQL statements” on page 352
- “Explaining SQL” on page 353
- “SAF-rule controlled auditing” on page 47

Related references
- “DOWN primary command” on page 738
- “EXECUTE primary command” on page 744
- “EXPLAIN primary command” on page 745
- “SQL primary command” on page 778
- “UP primary command” on page 780

EXPLAIN Interpretation panel

FM/DB2 displays the EXPLAIN Interpretation panel if you use the I line command against an item listed on:
- The “Plan Table Rows panel” on page 615, or
- The “Statement Table Rows panel” on page 681

Explain Utilities panel

You use the Explain Utilities panel to select functions to support the use of the DB2 EXPLAIN SQL statement.
Panel and field definitions

List a plan table. FM/DB2 displays the rows of the specified plan table. If you leave the Plan table owner field blank, FM/DB2 uses your current SQL ID. You can qualify the rows by Plan name, DBRM/package name, and Collection ID. You can list any related statement table and function table rows, if they exist, using line commands from the plan table. The table name for a plan table is always PLAN_TABLE.

Create a plan table. A plan table is needed for the SQL EXPLAIN statement. If you leave the Plan table owner field blank, FM/DB2 uses your current SQL ID. You can optionally use the Database and Table space fields to define the database and table space to be used when the plan table is created. The table name for a plan table is always PLAN_TABLE.

Create a function table, where the SQL EXPLAIN statement can store information about how function references are resolved. FM/DB2 uses the Plan table owner field (or, if it is left blank, your current SQL ID) as the owner of the function table. The table name for a function table is always DSN_FUNCTION_TABLE.

Create a statement table, where the SQL EXPLAIN statement can store the estimated costs for an SQL statement. FM/DB2 uses the Plan table owner field (or, if it is left blank, your current SQL ID) as the owner of the statement table. The table name for a statement table is always DSN_STATEMNT_TABLE.

Create an index on the plan table for the DB2 optimizer. An index on the plan table is recommended if you use DB2 optimizer hints.

Upgrade a plan table to the current DB2 version. Use this option to add columns so that the plan table is at the level that DB2 supports.

Drop plan table.

Drop function table.

Drop statement table.
Plan table owner
See function descriptions below.

Database
See function descriptions below.

Table space
See function descriptions below.

Plan name
See function descriptions below.

DBRM/package name
See function descriptions below.

Collection ID
See function descriptions below.

Confirm object drop
Indicates if you want FM/DB2 to display a confirmation panel each time you enter a DR(op) command against an object.

Parent panels
- “SQL Prototyping, Execution and Analysis panel” on page 680

Child panels

<table>
<thead>
<tr>
<th>To display this panel...</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>“Plan Table Rows panel” on page 615</td>
<td>Function L</td>
</tr>
<tr>
<td>“Create Index panel” on page 442</td>
<td>Function CX</td>
</tr>
</tbody>
</table>

Related tasks
- “Managing tables used by SQL explain” on page 355

Export Options (1 of 3) panel

You use the Export Options (1 of 3) panel to set various options that affect the way in which FM/DB2 operates. These options are saved in your ISPF profile and persist from one FM/DB2 session to the next.

Panel and field definitions
Export Options (1 of 3) panel

Data format

The format in which Export Utility unloads data:

**FM/DB2 (SQLDA) format**

DB2 internal format.

**DB2 UNLOAD format**

DB2 external (or unload) format.

**DSNTIAUL format**

DB2 DSNTIAUL unload format. This is the format used by the DB2 DSNTIAUL sample program.

**User defined**

User-defined format.

**Delimited Variables (CSV)**

FM/DB2 writes the data to the output file as a comma-delimited (CSV) file.

The data for each column appears in the output record in column number order, and each data value is separated from the next by a separator character.

DB2 data exported in this way cannot be imported using Import (option 3.6). The data is intended for use by a spreadsheet application program.

**Native unicode processing**

Determines how FM/DB2 processes Unicode encoded data when exporting data from a table to a dataset. Normally data stored in DB2 in Unicode is converted automatically to the CCSID of the DB2 plan, typically CCSID 37. This automatic translation results in conversion errors when a Unicode character does not have an equivalent code point in CCSID 37. When data is exported in "native Unicode" format, the automatic translation of data from unicode to the CCSID of the DB2 plan is suppressed. This ensures that there are no data conversion errors, but means the data is not directly visible using z/OS utilities such as the editor. Use this option if the data is
to be transferred to other systems that support Unicode data. When the option is not enabled, the option has no effect; that is, the default processing is used. When the data being exported is not Unicode encoded, the option has no effect, even if enabled.

/ Process Unicode data natively.

(blank) Do not Process Unicode data natively. This is the default setting.

Execution Options

Online
The export is run in foreground.

Batch Builds the necessary JCL to perform the Export utility in a batch job, and displays it in an edit session so that you can review the JCL prior to submitting the job.

Batch, using DB2 utility
Selecting this option runs the export using the DB2 UNLOAD utility.

Batch data set creation
This option only applies when batch execution is selected, and is ignored for online execution. You use this option to defer the creation of the export data set to batch job execution time. When this option is not selected, FM/DB2 determines whether the export data set exists before generation of the batch job JCL. If the data set does not exist, you are prompted to create it.

Edit DB2 UNLOAD options
Used in conjunction with option 3 Batch, using DB2 UNLOAD. The UNLOAD Options panel is displayed after you press the Exit function key (F3).

More Options

Null indicators / CSV options
If selected, you can navigate to the Export Options (2 of 3) panel by pressing the NxtPage function key (F11).

This option is only selectable if you have also selected either the data format option 4. User defined or 5. Delimited variables (CSV).

Data type format
If selected, you can navigate to the Export Options (3 of 3) panel by pressing the NxtPage function key (F11).

This option is only selectable if you have also selected the data format option 4. User defined.

Parent panels
- “Set Utility Options panel” on page 674
To display this panel... Use/do this

“Export Options (2 of 3) panel”

Select the Null indicators / CSV options option (data format option 4 or 5 must also be selected), and press the NxtPage function key (F11).

Related tasks
- “Setting default processing options” on page 39
- “FM/DB2 system options (option 0.2)” on page 41
- “Copying data to a VSAM or QSAM file” on page 221

Export Options (2 of 3) panel

You use the Export Options (2 of 3) panel to set various options that affect the way in which FM/DB2 operates. These options are saved in your ISPF profile and persist from one FM/DB2 session to the next.

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>Export Options - (2 of 3)</td>
<td>Global Settings</td>
<td></td>
</tr>
</tbody>
</table>

Null Indicators:
- Usage Placement Type Indicator
  1. Separate 1. Before 2. One byte Char
  2. None 2. After 2. Two byte Integer
  3. CSV, use NI char 3. User defined

Delimited Variables (CSV):
- Separator character
  Enter “/” to select option
  Include column headers
  "ASIS" encapsulation

Command ===> F1=Help  F2=Split  F3=Exit  F7=Backward  F8=Forward  F9=Swap
F10=PrvPage  F11=NxtPage  F12=Cancel

Figure 106. Export Options (2 of 3) panel

Usage  For nullable columns, determines whether fields in the exported data correspond to the null indicator:

- Separate
  A null indicator is output, taking into account the settings for Placement, Type and Indicator.

- None  No fields in the exported data correspond to the null indicator.
Export Options (2 of 3) panel

CSV, use NI char
If you opt for CSV records (data format 5), this option outputs a null indicator as the separator.

Placement
Determines where the null indicator is placed:

Before The null indicator is placed before the data for the column to which it applies.

After The null indicator is placed after the data for the column to which it applies.

User defined
A 1-byte (character format) or 2-byte (signed integer format) null indicator is placed somewhere in the output record. Placement depends on mapping the null indicator to a corresponding field in the output record.

Type
Determines the type of null indicator to be used:

One byte
1-byte (character format) null indicator.

Two byte
2-byte (signed integer format) null indicator.

Indicator
Determines the value of the null indicator:

Char The character used as the 1-byte (character format) null indicator.

Integer The integer used as the 2-byte (signed integer format) null indicator.

Separator character
The character used to separate data when the Delimited Variables (CSV) option is specified.

Include column delimiters
Select this option to produce comma-delimited column headers as the first export record.

"ASIS" encapsulation
Determines whether automatic encapsulation of column data is disabled. Select this option when the data has already been encapsulated in a user-defined character, prior to storage in DB2.

Parent panels
- “Set Utility Options panel” on page 674

Child panels

<table>
<thead>
<tr>
<th>To display this panel...</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>“Export Options (1 of 3) panel” on page 558</td>
<td>Press the PrvPage function key (F10)</td>
</tr>
</tbody>
</table>
To display this panel...

Use/do this

[“Export Options (3 of 3) panel”]

Press the NxtPage function key (F11) (the data format option 4. User defined, and the Data type format option, on the Export Options (1 of 3) panel must be selected).

Export Options (3 of 3) panel

You use the Export Options (3 of 3) panel to set various options that affect the way in which FM/DB2 operates. These options are saved in your ISPF profile and persist from one FM/DB2 session to the next.

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>Export Options - (3 of 3)</td>
<td>Global Settings</td>
<td></td>
</tr>
</tbody>
</table>

Decimal Data:
Select option
1. Internal (packed) format
2. Zoned format
3. External format

Integer Data:
Enter “/” to select option
- External format

Floating Point Data:
Enter “/” to select option
- External format

Graphic Data (fixed length):
Enter “/” to select option
- Use shift-out/shift-in (SOSI) characters

Figure 107. Export Options (3 of 3) panel

Decimal Data

Internal (packed) format
Any decimal field occupies the minimum number of bytes needed to represent the data as a packed decimal field.

Zoned format
In Zoned format the field occupies the number of bytes equal to the precision.
Export Options (3 of 3) panel

**External format**
In External format the field occupies the number of bytes equal to the precision + 2.

**Integer Data**

**External format**
In External format the field occupies 6 bytes for smallint(2), 11 bytes for integer(4), 20 bytes for bigint.

**Floating Point Data**

**External format**
In External format the field occupies 14 bytes for float(4), 23 bytes for float(8).

**Graphic Data (fixed length)**

**Use shift-out/shift-in (SOSI) characters**
Use shift-out/shift-in (SOSI) characters to enclose the exported graphic fixed length data.

**Note:** External format is the same format that is used to display data in FM/DB2 Editor.

**Parent panels**
- “Set Utility Options panel” on page 674

**Child panels**

<table>
<thead>
<tr>
<th>To display this panel...</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>“Export Options (2 of 3) panel” on page 561</td>
<td>Press the PrvPage function key (F10)</td>
</tr>
</tbody>
</table>

**Related tasks**
- “Setting default processing options” on page 39
- “FM/DB2 system options (option 0.2)” on page 41
- “Copying data to a VSAM or QSAM file” on page 221

**Export “To” panel**

You use the Export “To” panel to specify the name of the data set that contains the data to be exported, the (optional) name of a template data set, and processing options.
Panel and field definitions

**Data set name (To Partitioned, Sequential or VSAM Data Set)**
The name of the output data set.

**Member (To Partitioned, Sequential or VSAM Data Set)**
The output member name (if applicable).

**Volume**
The volume serial number if the data set is not catalogued.

**Data set name (To Copybook or Template)**
The name of the sequential or partitioned data set that contains the
copybook or template that describes the data set. It can be a fully-qualified
data set name or a pattern. The name may include a member name or
name pattern in parenthesis. If the member is specified here, the associated
**Member** field must be empty.

**Member (To Copybook or Template)**
If you specified the name of a partitioned data set (PDS) without including
a member name or name pattern in parenthesis in the **Data set name** field,
then you can use this field to specify the member name or a member name
pattern.

**Template usage**
The Export Utility “To” panel provides five options for template
processing:

1. **Above**
   Requires that you enter the name of a copybook or template in the
   **To Copybook or Template** section of the panel. If you do not
   specify a member name, or if you specify a pattern, FM/DB2
displays a member selection list. After you have specified the
   member containing a copybook or template, processing continues
   using this template in place of the automatically-generated
template for the table (see Template usage option 3).

2. **Previous**
   Uses the last (previously used) template for this table.
3. Generate from input
Generates a template based on the input template, but which includes only those columns selected in the input template. If you have specified a data set name in the To Copybook or Template section of the panel, it is ignored. The generated template is not saved.

4. Generate and save
Generates a template based on the input template, but which includes only those columns selected in the input template. This option requires you to enter the name of a template data set (and optionally a member name) in the To Copybook or Template section of the panel. If you do not specify a member name, or if you specify a pattern, FM/DB2 displays a member selection list. The generated template is saved.

5. None. (CSV output)
FM/DB2 does not utilize an output template. FM/DB2 writes the data to the output file as a comma-delimited (CSV) file. DB2 data exported in this way cannot be imported using Import (option 3.6). The data is intended for use by a spreadsheet application program.

You can select the following options by entering a “/” or an “A”:

Edit options
Select this option to display the second of the Export Options panels.

Edit template mapping
Use to edit the template for the partitioned, sequential, or VSAM data set to which you are exporting the data, before exporting the data.

Note: Template mapping does not apply for CSV output. See the template usage option 5. None. (CSV output).

Parent panels
- “Export Utility panel” on page 567
- “Column Selection/Edit panel” on page 407

Child panels

<table>
<thead>
<tr>
<th>To display this panel...</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>“Template Mapping panel” on page 710</td>
<td>Select Edit template mapping</td>
</tr>
</tbody>
</table>

Related tasks
- “Copying data to a VSAM or QSAM file” on page 221
- “Specifying a data set and a member name” on page 30
- “Exported data formats” on page 225
- “Selecting options on FM/DB2 panels” on page 24
- Chapter 3, “Working with templates,” on page 49
Export Utility panel

You use the Export Utility panel to specify the name of the DB2 object that contains the data to be exported, the (optional) name of a template data set, and processing options.

Panel and field definitions

| Process Options Utilities Help |
|------------------------------|------------------|------------|
| FM/DB2 (DFG2) Export Utility |

From DB2 Object:
- **Location** . . . .
- **Owner** . . . . FMNUSER +
- **Name** . . . . EMP +
- **Export Count** ... ALL Number of rows to export

From Template:
- **Data set name** . .
- **Member** . . . .

Processing Options:
- **Template usage** Enter "/", "A"lways to select option
  - 1. Above Edit options
  - 2. Previous Edit template
  - 3. Generate from table Create audit trail (+)
  - 4. Generate/Replace

Command """"
<table>
<thead>
<tr>
<th>F1=Help</th>
<th>F2=Split</th>
<th>F3=Exit</th>
<th>F4=Expand</th>
<th>F7=Backward</th>
<th>F8=Forward</th>
</tr>
</thead>
<tbody>
<tr>
<td>F9=Swap</td>
<td>F10=Left</td>
<td>F11=Right</td>
<td>F12=Cancel</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Location
The location name for the DB2 object. You can specify a generic name to select from a list. If you enter a value in this field, synonym object names are not recognized or included in the selection list.

Database
The database name for the DB2 object. You can use this field to optionally specify a database to further define the object to be processed. If you enter a value in this field:
- The values displayed in the **Owner** and **Name** selection lists are limited to those values defined within the specified database.
- Synonym names are not recognized or included in the selection list.

Owner
The name of the owner of the object. You can specify a generic name to select from a list. To specify a synonym in the object **Name** field, either leave this field blank, or specify a generic name that includes the current SQUAL or specify the current QUAL. Synonyms are not recognized or included on the table selection list if any of the **Table space**, **Database**, or **Location** fields are non-blank.

If the **Translate DB2 object names** option is selected, lowercase characters are converted to uppercase. If this option is not selected, lowercase characters are not converted to uppercase; in this situation, a lowercase owner name such as “sysibm” does not match “SYSIBM” in the DB2 catalog.
Export Utility panel

Table space
The table space name for the DB2 object. To further define the object to be processed, you can use this field to optionally specify a table space. If you enter a value in this field:

- The values displayed in the Owner and Name selection lists are limited to those values defined within the specified table space.
- Synonym names are not recognized, nor included in the selection list.

Name
The name of the object name to be processed. You can specify a generic name to select from a list. To specify a synonym in this field, type the name of the synonym and ensure that the Owner field is blank, contains a generic name that includes the current SQLID, or contains the current SQLID. Synonyms are not recognized or included in the table selection list if any of the Table space, Database, or Location fields are non-blank.

If the Translate DB2 object names option is selected, lowercase characters are converted to uppercase. If this option is not selected, lowercase characters are not converted to uppercase; in this situation, a lowercase owner name such as “sysibm” does not match “SYSIBM” in the DB2 catalog.

Export Count
The number of rows to be exported. This indicates the number of rows to be read from the DB2 table. It can be either a number in the range 1–99999999, or *, or ALL. The default is ALL, where all rows selected from the DB2 object are exported.

Data set name
The name of the data set or PDS member containing the template for the “From” table. It can be a fully-qualified data set name or a pattern. The name may include a member name or name pattern in parenthesis. If the member is specified here, the associated Member field must be empty.

Member
If you specified the name of a partitioned data set (PDS) without including a member name or name pattern in parenthesis in the Data set name field, then you can use this field to specify the member name or a member name pattern.

Template usage
The Export Utility “From” panel provides four options for template processing:

1. Above
   Requires that you enter the name of a template data set (and optionally a member name) in the From Template section of the panel. If you do not specify a member name, or if you specify a pattern, FM/DB2 displays a member selection list. After you have specified the member containing a template, processing continues using this template in place of the automatically-generated template for the table (see Template usage option 3).

2. Previous
   Uses the last (previously used) template for this table.

3. Generate from table
   Generates a template based on the information in the DB2 catalog for the specified table. This is the default setting. If you have
specified a member name in the From Template section of the panel, it is ignored. FM/DB2 generates the required template in memory.

4. Generate/Replace
Generates a template based on the information in the DB2 catalog for the specified table, and saves this information in the member specified in the From Template section of the panel. If the member already exists, it is replaced. Any prior customization is lost.

You can select the following options by entering a “/” or an “A”:

Edit options
Select this option to display the first of the Export Options panels.

Edit template
Select this option to edit the template before exporting the data.

Create audit trail

Note: “(*)” is appended to the option name for SAF-controlled auditing when required by SAF rules.
Controls whether access to DB2 by FM/DB2 is written to the FM/DB2 audit log. This option only appears when SAF-rule controlled auditing is in effect.

FM/DB2 may write audit records to either SMF, or a data set.
When SAF rules are used to control FM/DB2 auditing, the decision to write audit records is based on the DB2 object name being processed, and the current userid (TSO logonid). Therefore auditing may be required when processing one DB2 object, but not another. In addition, it is possible that the auditing requirement cannot be changed for a particular DB2 object; this might be different for another DB2 object, and the requirement may be different for other TSO userids.
If you do not have the authority to change the audit setting when processing a particular DB2 object, FM/DB2 prompts you if you enter the wrong setting. If you do have the authority to set the audit setting for a particular DB2 object, FM/DB2 honors whatever value is specified in this field.

Parent panels
- “Utility Functions panel” on page 720

Child panels

<table>
<thead>
<tr>
<th>To display this panel...</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>“Export “To” panel” on page 564</td>
<td>Select Edit template</td>
</tr>
<tr>
<td>“Column Selection/Edit panel” on page 407</td>
<td>Do not select Edit template</td>
</tr>
<tr>
<td>“Export Options (1 of 3) panel” on page 558</td>
<td>Select Edit options</td>
</tr>
</tbody>
</table>

Related tasks
- “Copying data to a VSAM or QSAM file” on page 221
- “Specifying a data set and a member name” on page 30
- “Selecting options on FM/DB2 panels” on page 24
- Chapter 3, “Working with templates,” on page 49
Field Selection/Edit panel

You use the Field Selection/Edit panel to specify the records you want to select when you use the Import utility function (option 3.6).

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>File Manager</td>
<td>Field Selection/Edit</td>
<td>Line 1 of 27</td>
</tr>
</tbody>
</table>

---------- Criteria - Enter 1 or 2 to specify expression by field ----------

1 Id : 
2 Sel : F9 = 'BOO'

Offset 0

<table>
<thead>
<tr>
<th>Cmd</th>
<th>SHE</th>
<th>Ref</th>
<th>Field Name</th>
<th>Picture</th>
<th>Type</th>
<th>Start</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>**** Top of data ****</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td></td>
<td></td>
<td>1 $501</td>
<td>AN</td>
<td>1</td>
<td>144</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td>2 EMPNO</td>
<td>AN</td>
<td>1</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td>3 len</td>
<td>BI</td>
<td>7</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td>4 FIRSTNAME</td>
<td>VC</td>
<td>9</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td>5 MIDINIT</td>
<td>AN</td>
<td>21</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td>6 len</td>
<td>BI</td>
<td>22</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
<td></td>
<td>7 LASTNAME</td>
<td>VC</td>
<td>24</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
<td></td>
<td>8 null</td>
<td>BI</td>
<td>39</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td></td>
<td></td>
<td>9 WORKDEPT</td>
<td>AN</td>
<td>41</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td></td>
<td></td>
<td>10 null</td>
<td>BI</td>
<td>44</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>11</td>
<td>PHONENO</td>
<td>AN</td>
<td>46</td>
<td>4</td>
<td></td>
</tr>
</tbody>
</table>

Command ==> Scroll

F1=Help F2=Split F3=Exit F4=Expand F5=RFind F6=RunTemp
F7=Up F8=Down F9=Swap F10=Left F11=Right F12=Cancel

1 Id, 2 Sel

Use option 1 and 2 to specify record identification and selection criteria by field.

If an expression has not previously been updated by field, you can enter the expression on the adjacent line which is left and right scrollable with the Left function key (F10) and the Right function key (F11), and expandable with the Expand function key (F4). The “+” and “-” indicators at the end of the field denote the directions (right and left respectively) in which you can scroll.

If an expression has previously been entered by field, this line is protected. To change back to a freeform REXX expression, select the “by field” option (by entering 1 or 2 on the command line) and delete the expression. When you return to this panel the line is available for input (unless it is a dynamic template in which case you must specify the expression).

Note: For more information about specifying record selection criteria, see the File Manager User’s Guide and Reference.

Cmd  Prefix command area. Used to enter a template editor prefix command.
Seq  Sequence field. Used to order fields for display in an FM/DB2 editor session and printing data using the Print utility.
SHE  Select/Hold/Edit field. Shows the field status.
Field Selection/Edit panel

S The field has been selected for processing.
H The field has been selected to be held on the edit display.
E The field has been previously edited to add information such as data create, field headings, and scrambling options.

Ref Field reference. Shows the field reference number assigned to the field name. Field reference numbers are assigned to all fields in a record, and are used to identify fields in selection criteria expressions.

Field Name Shows the level number and field name of the field. If the field is an array, the dimensions are shown in brackets after the field name. If the field has been selected for processing, the field name is highlighted.

Picture For COBOL, shows the picture clause. For HLASM shows the DC or DS operand. For PL/I, shows the length and scale (if non zero) for binary and packed fields. Also shows the bit length for bit fields.

Type Shows the data type of the field:
AN Alphanumeric (includes alphabetic, and group items)
BI Binary
BT Bit
DB DBCS
FE Floating point (external)
FP Floating point (internal)
G Graphic
PD Packed decimal (internal decimal)
VC Variable Character
VD Variable DBCS
VG Variable Graphic
ZC Character Null terminated
ZD Zoned decimal (external decimal)
ZE Zoned decimal edited
ZG Graphic Null terminated
Z2 DBCS Null terminated

Start Shows the start column of the field. For variable located fields the start location is based on the maximum length of the record.

Length Shows the length of the field. For a level-01 field, if the record is variable-length, the length shown is the maximum record length.

Record identification criteria Used to specify the criteria that FM/DB2 uses to identify this record type. FM/DB2 first uses the record length to identify record type.
Field Selection/Edit panel

For fixed-length records (as defined by the record definition in the template, not the data set attributes), the length of the record read must exactly match the length of the corresponding record definition in the template.

For variable-length records, the length of the record read must be equal to or greater than the minimum length and equal to or less than maximum length of the corresponding record definition in the template.

If FM/DB2 cannot match the length of a record to one of the record definitions in the template, the record is not selected for processing. If the length of a record matches more than one record definition in the template, and no further record identification criteria is specified, FM/DB2 uses the first matching record type in the template. Therefore, if the record length of each different record type in the file is not unique, you should use this field to specify some unique identification criteria.

Record selection criteria

Used to specify the criteria that FM/DB2 uses to select records for processing. You can use this field to limit the records of a given record type that you want to process. If you do not provide any record selection criteria, all records of this record type are selected for processing (if the record type itself has been selected for processing).

Parent panels

- “Import Utility (“From”) panel” on page 591

Child panels

None.

Related tasks

- “Copying data from a VSAM or QSAM file” on page 198

FM/DB2 Release News panel

FM/DB2 displays the FM/DB2 Release News panel if you:
1. Select Help from the Action bar menu, and
2. Select 8. News about FM/DB2... from the drop-down menu.

FM/DB2 Object List Options (1 of 2) panel

You use the FM/DB2 Object List Options (1 of 2) panel to set various options that affect the way in which FM/DB2 operates. These options are saved in your ISPF profile and persist from one FM/DB2 session to the next.

Panel and field definitions
Show all catalog table columns
Display of catalog table information includes every available column, including any columns marked as "unused" or "internal use only" in the DB2 for z/OS SQL Reference. This option overrides any settings for the Show "unused" catalog table columns and Show "internal use only" catalog table columns options.

The default for this option is: unselected.

Show "unused" catalog table columns
Display of catalog table information includes columns marked as "unused" in the DB2 for z/OS SQL Reference. This option can be overridden by selecting the Show all catalog table columns option.

The default for this option is: unselected.

Show "internal use only" catalog table columns
Display of catalog table information includes columns marked as "internal use only" in the DB2 for z/OS SQL Reference. This option can be overridden by selecting the Show all catalog table columns option.

The default for this option is: unselected.

Use automatic column sizing
Varying-length (VARCHAR) columns of the catalog table being displayed may be automatically sized, based on the data stored within the column. Automatic column sizing determines the longest data value in each varying column as the data is read from DB2. This value is then used to set the display width for the column, prior to the data being displayed.

The default for this option is: selected.

Use extended column name headings
Column names for the catalog table being processed are replaced by longer, more meaningful, column headings.

The default for this option is: selected.
Interpret columns containing codes

Selected columns in the DB2 catalog table being processed are expanded, based on the value stored within the column. This feature typically operates on one-character columns that contain a code. As an example, the TYPE column in SYSIBM.SYSTABLES is a one-character code containing A,G,M,T,V, or X. The expanded text for these codes is "Alias", "Created global temporary table", "Materialized query table", "Table", "View", and "Auxiliary Table".

This option is useful if you are not familiar with the meanings of the codes in the various DB2 catalog tables. Selecting this option overrides any setting for the Use extended column name headings option, and sets that option on.

The default for this option is: selected.

Modify order of columns displayed

Columns in the DB2 catalog table may be re-ordered from the order as defined in the DB2 catalog. In addition certain columns may be marked as held. This feature is intended to place the most important columns from the catalog table on the left of the display, in the most logical order. These columns are held, and do not move when scrolling left and right.

The default for this option is: selected.

Row count

Limits the number of rows fetched from the DB2 catalog. The value entered applies to:

- Object List selection panels displayed by FM/DB2 functions such as the editor, print, copy, import, export, and data create. For more information on object list selection, see "Specifying a DB2 object name” on page 815.

Note that FM/DB2 functions such as edit, copy, print, and so on also have a Row count field on the function entry panel. Once an object has been selected, the function entry Row count field acts to control the number of rows processed in the specified DB2 object.

- Object lists displayed by these FM/DB2 functions:
  - Object List Utility (3.4).
  - Object Privileges Utility (3.5).
  - Explain Utilities (4.5).

A value of 0 or ALL specifies no limit; all rows from the relevant catalog tables are fetched and displayed. When you specify 0, FM/DB2 accesses the DB2 catalog using large mode, thereby minimizing storage usage for the list of DB2 objects.

If you experience “insufficient storage” conditions, you can specify a value to limit the number of rows fetched from DB2. Another way to limit the number of rows fetched in the Object List utility (option 3.4) is to specify selection criteria on the function entry panel.

The default for this option is 1000.

Display width for varying columns

The Object List utility displays information in DB2 catalog tables. Some of the columns within those tables are defined as varying columns with potentially very long maximum lengths.

You can use the Use automatic column sizing option to reduce the maximum display width for varying columns to the maximum that is needed to display the data for each column.
Additional control can be achieved by specifying minimum and maximum display widths for the various column types shown on the panel. All values are optional. (See “FM/DB2 Object List Options (2 of 2) panel.” A minimum value ensures that the display width is at least the value specified, even if there is no displayable data for the column in all rows processed. Specifying a minimum value does not prevent the display width expanding, if needed, to accommodate longer data.

Parent panels
- “Set Processing Options panel” on page 671

Child panels

<table>
<thead>
<tr>
<th>To display this panel…</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>“FM/DB2 Object List Options (2 of 2) panel”</td>
<td>Press the NxtPage function key (F11)</td>
</tr>
</tbody>
</table>

Related tasks
- “Setting default processing options” on page 39
- “FM/DB2 system options (option 0.2)” on page 41
- “TIME columns” on page 131
- “INDEXINF primary command” on page 756
- “No Primary Key” errors on page 158
- “Row Not Found” errors on page 159

Related references
- “SQL primary command” on page 778

FM/DB2 Object List Options (2 of 2) panel

You use the FM/DB2 Object List Options (2 of 2) panel to set various options that affect the way in which FM/DB2 operates. These options are saved in your ISPF profile and persist from one FM/DB2 session to the next.

Panel and field definitions
FM/DB2 Object List Options (2 of 2) panel

![Panel Image]

**Process Options Utilities Help**

FM/DB2 (DFG2) FM/DB2 Object List Options (2 of 2) Global Settings

Display width for varying columns:
Enter minimum and maximum values for each column type

<table>
<thead>
<tr>
<th></th>
<th>Min</th>
<th>Max</th>
</tr>
</thead>
<tbody>
<tr>
<td>Databases</td>
<td>8</td>
<td>30</td>
</tr>
<tr>
<td>Column names</td>
<td>8</td>
<td>18</td>
</tr>
<tr>
<td>Authorization identifiers</td>
<td>8</td>
<td>18</td>
</tr>
<tr>
<td>Object names</td>
<td>8</td>
<td>18</td>
</tr>
<tr>
<td>Other</td>
<td>8</td>
<td>30</td>
</tr>
</tbody>
</table>

eg REMARKS, LABEL

**Command ===>
F1=Help F2=Split F3=Exit F7=Backward F8=Forward F9=Swap F10=PrevPage F12=Cancel

Figure 109. FM/DB2 Object List Options (2 of 2) panel

Allows you to specify the minimum and maximum display widths for the various column types shown on the object list panel. All values are optional.

- **Min** Ensures that the display width is at least the value specified, even if there is no displayable data for the column in all rows processed. Specifying a minimum value does not prevent the display width expanding, if needed, to accommodate longer data.

- **Max** Limits the maximum display width to the value specified, even if data in some rows has a longer width. Specifying a maximum value does not prevent the display width being set to a smaller value, if no rows have data exceeding the value specified.

**Parent panels**
- “Set Processing Options panel” on page 671

**Child panels**

To display this panel... Use/do this

| “FM/DB2 Object List Options (1 of 2) panel” on page 572 | Press the PrevPage function key (F10) |

**Related tasks**
- “Setting default processing options” on page 39
- “FM/DB2 system options (option 0.2)” on page 41
- “TIME columns” on page 131
- “INDEXINF primary command” on page 756
- “"No Primary Key" errors” on page 158
- “"Row Not Found" errors” on page 159

**Related references**
- “SQL primary command” on page 778
You use the FM/DB2 Systems Options (1 of 4) panel to set various options that affect the way in which FM/DB2 operates. These options are saved in your ISPF profile and persist from one FM/DB2 session to the next.

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>FM/DB2 System Options (1 of 4)</td>
<td>Global Settings</td>
<td></td>
</tr>
</tbody>
</table>

Uppercase Translation:
/ Translate DB2 object names
/ Translate input SQL statements

Encapsulation of SQL Identifiers in Double Quotes:
Use double quotes
1. For DB2 SQL reserved words
2. For IBM and DB2 SQL reserved words
3. Always

More Options:
Enter "/" to select option
/ Arbitrary select statements, time display format, template creation
/ CCSID warning option

Figure 110. FM/DB2 System Options (1 of 4) panel

Translate DB2 object names
Translates DB2 object owner and DB2 object name to uppercase. (The DB2 object location is always translated to uppercase.)

If you do not select this option, the DB2 object owner and DB2 object name remain as entered by the user.

Translate input SQL statements
Translates any free-form SQL statements entered to uppercase.

If you do not select this option, any free-form SQL statements entered remain as entered by the user.

Encapsulation of SQL Identifiers in Double Quotes
The SQL identifiers encapsulated in double quotes in SQL statements generated by FM/DB2. You can select one of the following:

- 1. For DB2 SQL reserved words
  If your DB2 objects include column names that are DB2 SQL reserved words, you must enclose these column names in double quotes. Use this option to only enclose identifiers that are DB2 SQL reserved words in double quotes. Any other identifiers appear in SQL statements without encapsulating double quotes. This option is recommended if you are using the SQL primary command (see “SQL primary command” on page 778) to generate and save SQL statements for later use.

- 2. For IBM and DB2 SQL reserved words
FM/DB2 Systems Options (1 of 4) panel

Use this option to recognize any identifiers that are either a DB2 SQL reserved word, or an IBM SQL reserved word, and enclose these words in double quotes. Any other identifiers appear in SQL statements without encapsulating double quotes.

- 3. Always. This is the setting. This option applies to view, edit, browse, print, copy, import and export functions only and results in all identifiers being enclosed in double quotes in SQL statements generated by FM/DB2.

Note:
1. For information about the rules covering the construction of SQL identifiers, see the *DB2 for z/OS SQL Reference*.
2. In general, you can use any uppercase alphabetic character and three special characters ($, _, and @) in an SQL identifier without special consideration. If your identifier includes lowercase characters, spaces, or other characters, then you must enclose the identifier in double quotes. FM/DB2 detects identifiers with lowercase and other unusual characters and encloses these in double quotes. This occurs regardless of the encapsulation option selected.
3. The lists of DB2 and IBM SQL reserved words used by FM/DB2 are the latest available. When connected to an older release of DB2, if you have selected either options 1 or 2, FM/DB2 might recognize an identifier as a reserved word, even though the identifier is not defined as a reserved word in the earlier release of DB2.

Parent panels
- "Set System Processing Options panel" on page 673

Child panels

<table>
<thead>
<tr>
<th>To display this panel...</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;FM/DB2 Systems Options (2 of 4) panel&quot;</td>
<td>Press the NxtPage function key (F11)</td>
</tr>
</tbody>
</table>

Related tasks
- "Setting default processing options" on page 39
- "FM/DB2 system options (option 0.2)" on page 41
- "TIME columns" on page 131
- "INDEXINF primary command" on page 756
- ""No Primary Key" errors" on page 158
- ""Row Not Found" errors" on page 159

Related references
- "SQL primary command" on page 778

FM/DB2 Systems Options (2 of 4) panel

You use the FM/DB2 Systems Options (2 of 4) panel to set various options that affect the way in which FM/DB2 operates. These options are saved in your ISPF profile and persist from one FM/DB2 session to the next.

Panel and field definitions
Use edit (instead of browse) to display results

FM/DB2 uses Edit to display the result table generated by a freeform SQL statement.

External Format for TIME Data Type Columns

The external format for data retrieved from DB2 TIME data type columns. You can select one of the following:

- 1. ISO (HH.MM.SS). This is the default format.
- 2. JIS (HH:MM:SS)
- 3. DB2 DSNHDECP setting

Depending on the time format used, information can be lost when data is retrieved from DB2 (for example, the USA format, which discards the seconds portion of any time). If LOCAL is specified (a time format exit is in use), ensure that there is no loss of information when the time value is converted to an external format. When you use this option, you can encounter:

- SQLCODE 100 (row not found) conditions when attempting to delete or update tables that contain a TIME column.
- Removal of the seconds portion of any TIME column (the seconds value is set to 00) in print, copy, or export operations.

Use uncommitted read when accessing the DB2 catalog

This option ensures that all DB2 catalog accesses for the view, edit, print, copy, export and import functions use the uncommitted read option (that is, no locks are taken against the DB2 catalog).

Retrieve foreign key information when building templates

FM/DB2 stores information about primary keys, indexes and foreign keys in the template for an object. This information is retrieved from the DB2 catalog and stored at template creation time, and can be displayed in an FM/DB2 editor session. Retrieval of the foreign key information can result in slow response if an additional index has not been built on the SYSIBM.SYSFOREIGNKEYS catalog table (see the File Manager).
Customization Guide for details). When this option is not selected, any templates generated do not contain any foreign key information.

Parent panels
- “FM/DB2 Systems Options (1 of 4) panel” on page 577

Child panels

<table>
<thead>
<tr>
<th>To display this panel...</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>“FM/DB2 Systems Options (1 of 4) panel” on page 577</td>
<td>Press the PrvPage function key (F10)</td>
</tr>
<tr>
<td>“FM/DB2 Systems Options (3 of 4) panel”</td>
<td>Press the NxtPage function key (F11)</td>
</tr>
</tbody>
</table>

Related tasks
- “Setting default processing options” on page 39
- “FM/DB2 system options (option 0.2)” on page 41
- “TIME columns” on page 131
- “INDEXINF primary command” on page 756
- “’No Primary Key’ errors” on page 158
- “’Row Not Found’ errors” on page 159

Related references
- “SQL primary command” on page 778

FM/DB2 Systems Options (3 of 4) panel

You use the FM/DB2 Systems Options (3 of 4) panel to set various options that affect the way in which FM/DB2 operates. These options are saved in your ISPF profile and persist from one FM/DB2 session to the next.

Panel and field definitions
CCSID Warning Message

You can set your preferred initial system CCSID Warning message option by changing this option. To request a warning message to be displayed when the CCSID of the terminal differs from the CCSID of the plan, select the option. To suppress the displaying of a warning message when the CCSID of the terminal differs from the CCSID of the plan, do not select the option.

CURRENT DECFLOAT ROUNDING MODE

Sets the current decimal floating point rounding mode for the currently connected DB2 system.

For a detailed explanation of the various decimal floating point rounding modes you can specify, see the DB2 SQL Reference.

Setting the option causes FM/DB2 to issue an SQL statement that changes the CURRENT DECFLOAT ROUNDING MODE special register.

Parent panels

- “FM/DB2 Systems Options (2 of 4) panel” on page 578

Child panels

<table>
<thead>
<tr>
<th>To display this panel...</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>“FM/DB2 Systems Options (2 of 4) panel” on page 578</td>
<td>Press the PrvPage function key (F10)</td>
</tr>
<tr>
<td>“FM/DB2 Systems Options (4 of 4) panel” on page 582</td>
<td>Press the NxtPage function key (F11)</td>
</tr>
</tbody>
</table>

Related tasks

- “Setting default processing options” on page 39
- “FM/DB2 system options (option 0.2)” on page 41
You use the FM/DB2 Systems Options (4 of 4) panel to set various options that affect the way in which FM/DB2 operates. These options are saved in your ISPF profile and persist from one FM/DB2 session to the next.

**Panel and field definitions**

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFE2)</td>
<td>FM/DB2 System Options (4 of 4)</td>
<td>Global Settings</td>
<td></td>
</tr>
</tbody>
</table>

Editor choice for viewing output data, members:
- Enter "/" to select option
- Use File Manager editor

**Use File Manager editor**

This option controls which editor is invoked when viewing or browsing a member from a member selection panel or when viewing generated output. Select this option to invoke the File Manager editor. If the option is not selected, the ISPF Browse, View or Edit facility is invoked when available.

**Parent panels**

- 

**Child panels**

<table>
<thead>
<tr>
<th>To display this panel...</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>“FM/DB2 Systems Options (3 of 4) panel” on page 580</td>
<td>Press the PrvPage function key (F10)</td>
</tr>
</tbody>
</table>

**Related tasks**

- “Setting default processing options” on page 39
- “FM/DB2 system options (option 0.2)” on page 41
Foreign Keys for Table panel

FM/DB2 displays the Foreign Keys for Table panel if you use the FK line command against an item listed on the “Tables, Views and Aliases panel” on page 704.

Free Application Plan panel

FM/DB2 displays the Free Application Plan panel if you use the FRE line command against an item listed on the “Application Plans panel” on page 389.

Free Package panel

FM/DB2 displays the Free Package panel if you use the FRE line command against an item listed on the “Application Packages panel” on page 387.

From Column Mapping panel

Panel and field definitions

Parent panels

• “Template Mapping panel” on page 710

Child panels

To display this panel... Use/do this

“Template Mapping panel” on page 710 Press F3 or F12

Related tasks
Functions panel

You use the Functions panel to display information about the functions you have selected.

Panel and field definitions

The columns that are displayed include the SEL field and columns of SYSIBM.SYSROUTINES.

<table>
<thead>
<tr>
<th>SEL</th>
<th>SCHEMA</th>
<th>NAME</th>
<th>ORIGINEX</th>
<th>OWNER</th>
<th>ROUTINEX</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>KEISTE2</td>
<td>DT1</td>
<td>System generated KEISTE2 Function/Cast Function</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>KEISTE2</td>
<td>CHAR</td>
<td>System generated KEISTE2 Function/Cast Function</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>KEISTEW</td>
<td>DT1</td>
<td>System generated KEISTE2 Function/Cast Function</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>KEISTE2</td>
<td>CHAR80</td>
<td>System generated KEISTE2 Function/Cast Function</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>KEISTE2</td>
<td>DOCMATCH</td>
<td>External KEISTE2 Function/Cast Function</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

To view other columns on this panel, press the Right function key (F11).

For a complete list of columns, refer to “DB2 catalog tables” in the DB2 for z/OS SQL Reference relevant to your version of DB2.

SEL Line command area.

For a list of the line commands you can use on this panel, see Table 22 on page 250.

Note:
1. The system option, Show all catalog table columns, affects which columns FM/DB2 displays when you use the Object List utility.
2. You can also customize which columns FM/DB2 displays when you use the Object List utility. For details, see the File Manager Customization Guide.

To display this panel...

<table>
<thead>
<tr>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>“Alter External Scalar Function panel” on page 372</td>
</tr>
</tbody>
</table>
Generate SQL From DB2 Catalog panel

You use the Generate SQL From DB2 Catalog panel to specify how the generation of SQL from the DB2 catalog should work. You can specify which SQL statement types FM/DB2 should generate, and optionally which names and values should be changed.
Generate SQL From DB2 Catalog panel

Panel and field definitions

<table>
<thead>
<tr>
<th>Process Options Utilities Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2) Generate SQL From DB2 Catalog</td>
</tr>
</tbody>
</table>

Generate SQL statements for database: DSN8D10A +

SQL statement types to be generated from the DB2 catalog:
- CREATE DATABASE . . . Y
- CREATE TABLESPACE . . Y
- CREATE TABLE . . . Y
- CREATE VIEW . . . Y
- CREATE INDEX . . . Y
- CREATE SYNONYM . . . Y
- CREATE ALIAS . . . Y
- CREATE TRIGGER . . . Y

New names/values for generated SQL (leave blank to use current values):
- Object owner . . . +
- Alloc TS size as . . . DEFINED
- Database name . . .
- Storage group for TS . . +
- Storage group for IX . . +
- Target DB2 version . . (Current DB2 version: 1015)

Output file and execution mode:
- Data set name . . .
- Data set disposition . OLD
- Execution mode . . . BATCH
- Commit statements per . (Db, ts, Tb, All, None)
- DB2 defaults handling - (Keep, or Remove)

Generate SQL statements for database
Specify Y for each SQL statement type you want FM/DB2 to generate, and N for each SQL statement type you do not want generated.

Object owner
New owner (authorization ID) of the objects.

Alloc TS size as
Specifies how to generate the primary quantity:
- DEFINED
  - Use the size defined in the catalog
- USED
  - Use the size that actually is used
- ALLOCATED
  - Use the allocated size

USED and ALLOCATED require that the STOSPACE utility has been run for the storage groups for the objects being generated.

Database name
New database name for the objects.

Storage group for TS
New storage group for the table spaces.

Storage group for IX
New storage group for the indexes.

Target DB2 version
DB2 level of generated SQL statements.
Generate SQL From DB2 Catalog panel

Data set name
An existing valid SPUFI input data set name, or as SYSOUT=x. SYSOUT=* is the default if the field is blank.

Data set disposition
The disposition of the output data set.

Execution mode
Specify:
   BATCH Generates a batch job
   TSO Runs the generation online

Commit statements per
When to generate COMMIT statements:
   D Ffor each database
   S Ffor each table space
   T Ffor each table
   A Ffor all objects
   N Never

DB2 defaults handling
Specifies how DB2 default parameters should be handled:
   K Keep DB2 default parameters
   R Remove DB2 default parameters

Parent panels
- “Databases panel” on page 500
- “Table Spaces panel” on page 698
- “Tables, Views and Aliases panel” on page 704
- “Indexes panel” on page 595
- “Schemas panel” on page 657
- “Distinct Types panel” on page 523
- “Functions panel” on page 584
- “Stored Procedures panel” on page 685

Child panels
None.

Related tasks
- “Reverse engineering” on page 257

Grant privileges panels
These panels are:
- Grant Application Plan Privileges panel
- Grant Buffer Pool Privileges panel
- Grant Collection Privileges panel
- Grant Column Privileges panel
- Grant Database Privileges panel
- Grant Distinct Type Privileges panel
- Grant Function Privileges panel
- Grant Package Privileges panel
- Grant Schema Privileges panel
Grant privileges panels

- Grant Sequence Privileges panel
- Grant Storage Group Privileges panel
- Grant Stored Procedure Privileges panel
- Grant System Privileges panel
- Grant Table Privileges panel
- Grant Table Space Use Privileges panel

FM/DB2 displays one of these panels if you use the Privileges utility function (3.5) to grant privileges for an object type, or if you specify the G line command against an object displayed in a list of DB2 objects.

Format of grant privilege panels

Each grant privilege panel shows:

- The available privileges for the object type, with a corresponding single-character entry field in which you can specify one of the following values:
  - **Y** Grants the specified privilege to the grantee, but does not permit the grantee to grant this privilege to other users.
  - **G** Grants the specified privilege to the grantee, and permits the grantee to grant this privilege to other users.

A blank in the field indicates that the grantee does not have the specified privilege.

**Note:** To change an existing privilege, you must first revoke the existing privilege before granting the new privilege (if applicable) as described in this section.

- A field (for some types of privilege there are two fields), that you use to specify the name of the object for which the privilege is to be granted.
- A **To** field. Use this to specify the SQLID of the user to whom the privilege is to be granted.

Related tasks

- "Displaying privileges" on page 273
- Chapter 8, "Working with lists of DB2 objects," on page 241
- "Granting privileges" on page 273
- "Revoking privileges" on page 275

Import Options panel

You use the Import Options panel to set either global or local import options, depending on how the panel is displayed.
Panel and field definitions

Data format

Allows you to specify the data format that corresponds to the template or copybook used:

- 1 FM/DB2 (SQLDA) format
- 2 DB2 UNLOAD format
- 3 DSNTIAUL format
- 4 User defined

Duplicate key processing

1. Ignore
   - Ignore duplicate rows.

2. Update
   - Updates duplicate rows when DB2 returns a duplicate row error (SQLCODE -803) in response to an INSERT SQL statement. When you select this option, FM/DB2 attempts to update the existing row. All columns of the table that are part of at least one unique index are used to identify the row. No comparison is made between the imported row and the existing row prior to the update operation (that is, the update is done regardless of whether the imported row and existing row are the same). When you use this option it is recommended that you specify a DB2 table name, rather than a view name. FM/DB2 does not attempt to update existing rows in a view defined on a view.

Max duplicates

The maximum number of duplicates permitted.

Valid values: A number in the range 0–9999999999, or “ALL”.
Import Options panel

Auto-commit count
Use this option to take progressive DB2 commit points as records are imported. Any positive value entered will cause FM/DB2 to issue a DB2 commit automatically after the specified number of successful DB2 changes. A successful DB2 change is either the successful insert of a record; or the successful update of a row when a duplicate key error is encountered on the insert. The change counter is reset after each DB2 commit is issued, so multiple DB2 commit points may be issued. Specify a zero value if no progressive DB2 commit points are required; in this case the DB2 commit is deferred until the end of the import utility and any error encountered during the import will cause all changes to be backed out.

Delete existing rows
Deletes all the rows in the target table before importing.

Parent panels
- “Set Utility Options panel” on page 674
- “Import Utility (“From”) panel” on page 591
- “Import Utility (“To”) panel” on page 592

Child panels
None.

Related tasks
Import Utility ("From") panel

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>Import Utility</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

From Partitioned, Sequential or VSAM Data Set:
- Data set name ... 'FMNUSER.EXPORT'
- Member .......
- Volume ........
- Start position ...
- Import Count ... ALL Number of rows to import

From Copybook or Template:
- Data set name ... 'FMNUSER.TEMPLATE'
- Member ...... EXPORT1

Processing Options:
- Template usage Enter "/", "A"lways to select option
  1 1. Above Edit options
  - 2. Previous Edit template

Command ===>
F1=Help F2=Split F3=Exit F7=Backward F8=Forward F9=Swap
F12=Cancel

You can select the following options by entering a "/" or an "A" (see "Selecting options on FM/DB2 panels" on page 24).

Edit template
- Use to edit the template for the table from which you are importing the data, before importing the data. For more details, see Chapter 3, “Working with templates,” on page 49.

Edit options
- Select this option to display the Import Options panel.

Parent panels
- "Utility Functions panel" on page 720

Child panels

<table>
<thead>
<tr>
<th>To display this panel...</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;Import Options panel&quot; on page 588</td>
<td>Select Edit options</td>
</tr>
<tr>
<td>&quot;Field Selection/Edit panel&quot; on page 570</td>
<td>Select Edit template</td>
</tr>
</tbody>
</table>

Related tasks
- "Selecting options on FM/DB2 panels" on page 24
- Chapter 3, “Working with templates,” on page 49
Import Utility (“To”) panel

Panel and field definitions

**Location**

The location name for the DB2 object. You can specify a generic name to select from a list. If you enter a value in this field, synonym object names are not recognized or included in the selection list.

**Database**

The database name for the DB2 object. You can use this field to optionally specify a database to further define the object to be processed. If you enter a value in this field:

- The values displayed in the **Owner** and **Name** selection lists are limited to those values defined within the specified database.
- Synonym names are not recognized or included in the selection list.

**Owner**

The name of the owner of the object. You can specify a generic name to select from a list. To specify a synonym in the object **Name** field, either leave this field blank, or specify a generic name that includes the current SQLID or specify the current SQLID. Synonyms are not recognized or included on the table selection list if any of the **Table space**, **Database**, or **Location** fields are non-blank.

If the **Translate DB2 object names** option is selected, lowercase characters are converted to uppercase. If this option is not selected, lowercase characters are not converted to uppercase; in this situation, a lowercase owner name such as “sysibm” does not match “SYSIBM” in the DB2 catalog.
**Import Utility ("To") panel**

**Table space**
The table space name for the DB2 object. To further define the object to be processed, you can use this field to optionally specify a table space. If you enter a value in this field:

- The values displayed in the *Owner* and *Name* selection lists are limited to those values defined within the specified table space.
- Synonym names are not recognized, nor included in the selection list.

**Name**
The name of the object name to be processed. You can specify a generic name to select from a list. To specify a synonym in this field, type the name of the synonym and ensure that the *Owner* field is blank, contains a generic name that includes the current SQLID, or contains the current SQLID. Synonyms are not recognized or included in the table selection list if any of the *Table space*, *Database*, or *Location* fields are non-blank.

If the **Translate DB2 object names** option is selected, lowercase characters are converted to uppercase. If this option is not selected, lowercase characters are not converted to uppercase; in this situation, a lowercase owner name such as “sysibm” does not match “SYSIBM” in the DB2 catalog.

**Data set name**
The name of the data set or PDS member containing the template for the “To” table. It can be a fully-qualified data set name or a pattern. The name may include a member name or name pattern in parenthesis. If the member is specified here, the associated *Member* field must be empty.

**Member**
If you specified the name of a partitioned data set (PDS) without including a member name or name pattern in parenthesis in the *Data set name* field, then you can use this field to specify the member name or a member name pattern.

The Import Utility “To” panel provides four options for template processing:

1. **Above**
   Requires that you enter the name of a template in the *To Template* section of the panel.

2. **Previous**
   Uses the last (previously used) template for this table.

3. **Generate from table**
   Generates a template based on the information in the DB2 catalog for the specified table. This is the default setting. If you have specified a data set name in the *To Template* section of the panel, it is ignored. FM/DB2 generates the required template in memory.

4. **Generate/Replace**
   Generates a template based on the information in the DB2 catalog for the specified table, and saves this information in the data set specified in the *To Template* section of the panel. If the template already exists, it is replaced and any prior customization is lost.

You can select the following options by entering a “/” or an “A”:

**Edit options**
Select this option to display the Import Options panel.
Import Utility ("To") panel

**Edit template mapping**
Use to edit the template for the table to which you are importing the data, before importing the data.

**Batch execution**
Builds the necessary JCL to perform the Import utility in a batch job.

**Create audit trail**

*Note:* "(*)" is appended to the option name for SAF-controlled auditing when required by SAF rules.
Controls whether access to DB2 by FM/DB2 is written to the FM/DB2 audit log. This option only appears when SAF-rule controlled auditing is in effect.

FM/DB2 may write audit records to either SMF, or a data set.

When SAF rules are used to control FM/DB2 auditing, the decision to write audit records is based on the DB2 object name being processed, and the current userid (TSO logonid). Therefore auditing may be required when processing one DB2 object, but not another. In addition, it is possible that the auditing requirement cannot be changed for a particular DB2 object; this might be different for another DB2 object, and the requirement may be different for other TSO userids.

If you do not have the authority to change the audit setting when processing a particular DB2 object, FM/DB2 prompts you if you enter the wrong setting. If you do have the authority to set the audit setting for a particular DB2 object, FM/DB2 honors whatever value is specified in this field.

**Use REXX proc**
Use this option to perform either of these actions:
- Enter a temporary REXX procedure for one-time use by entering an asterisk (*). FM/DB2 displays a REXX edit panel that you can use to create a new REXX procedure.
- Specify the name of the member containing the REXX procedure you want to use. The member must belong to the PDS allocated to ddname FMNEXEC. You can enter any of the following:
  - The name of the member.
  - A member name pattern (other than *) to list all matching members. You can then select the required member by entering an S in the Sel field. A member name pattern can consist of any characters that are valid in a member name and the following two special pattern characters:
    - **asterisk (*)**
      Represents any number of characters. As many asterisks as required can appear anywhere in a member name. For example, if you enter *d*, a list of all members in the data set whose name contains “d” is displayed.
    - **percent sign (%)**
      A place-holding character representing a single character. As many percent symbols as necessary can appear anywhere in a member name. For example, if you enter %%%, a list of all members in the data set whose name is four characters in length is displayed.
Import Utility ("To") panel

Note: If you select this option but leave the Use REXX proc field blank, FM/DB2 displays a member name list. You can then select the required member by entering S in the Sel field.

Parent panels
- "Import Utility ("From") panel" on page 591
- "Field Selection/Edit panel" on page 570

Child panels
To display this panel... Use/do this
- "Import Options panel" on page 588 Select Edit options

Related tasks
- Chapter 3, "Working with templates," on page 49
- "Selecting options on FM/DB2 panels" on page 24
- "Using a REXX procedure with the Import utility" on page 203
- "Enhancing the Import utility using a REXX procedure" on page 211
- "SAF-rule controlled auditing" on page 47

Index Parts panel

FM/DB2 displays the Index Parts panel if you use the PT line command against an item listed on the "Indexes panel."

Indexes panel

You use the Indexes panel to list index object types in the DB2 catalog.

Panel and field definitions

The columns that are displayed include the SEL field and columns of SYSIBM.SYSINDEXES.
Indexes panel

Note:
1. The system option, **Show all catalog table columns**, affects which columns FM/DB2 displays when you use the **Object List** utility.
2. You can also customize which columns FM/DB2 displays when you use the **Object List** utility. For details, see the [File Manager Customization Guide](#).

To view other columns on this panel, press the Right function key (F11).

For a complete list of columns, refer to “DB2 catalog tables” in the **DB2 for z/OS SQL Reference** relevant to your version of DB2.

SEL Line command area.

For a list of the line commands you can use on this panel, see Table 22 on page 250.

Parent panels
- “Object List Utility panel” on page 610
- “Databases panel” on page 500
- “Tables, Views and Aliases panel” on page 704
- “Columns panel” on page 411
- “Application Packages panel” on page 387

Child panels

<table>
<thead>
<tr>
<th>To display this panel...</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>“Alter Index panel” on page 373</td>
<td>Line command A</td>
</tr>
<tr>
<td>“Columns in Index panel” on page 413</td>
<td>Line command COL</td>
</tr>
<tr>
<td>“Create Index panel” on page 442</td>
<td>Line command CR</td>
</tr>
<tr>
<td>“Databases panel” on page 500</td>
<td>Line command D</td>
</tr>
<tr>
<td>See “Drop panels” on page 524</td>
<td>Line command DR</td>
</tr>
<tr>
<td>“Generate SQL From DB2 Catalog panel” on page 585</td>
<td>Line command GEN</td>
</tr>
</tbody>
</table>
Indexes panel

To display this panel... Use/do this

“Details panels” on page 522
Line command I

“Recovery Information panel” on page 638
Line command ICS

“Application Packages panel” on page 387
Line command PKG

“Application Plans panel” on page 389
Line command PL

“Index Parts panel” on page 595
Line command PT

“Display Row panel” on page 523
Line command ROW

“Tables, Views and Aliases panel” on page 704
Line command T

“DB2 Utilities panel” on page 515
Line command UTL

“Sort Fields panel” on page 679
Primary command SORT

Related tasks
• “Working with object list panels” on page 244
• “Using the line command area (Cmd)” on page 250

Related references
• “SORT primary command” on page 777

Indexes and Columns for panel

FM/DB2 displays the Indexes and Columns for panel if you use the XC line command against an item listed on the “Tables, Views and Aliases panel” on page 704.

ISPF Primary Option panel

Panel and field definitions

<table>
<thead>
<tr>
<th>Option</th>
<th>Field/Command</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Settings</td>
<td>Terminal and user parameters User ID . : FMUSER</td>
</tr>
<tr>
<td>1</td>
<td>View</td>
<td>Display source data or listings Time . . : 15:11</td>
</tr>
<tr>
<td>2</td>
<td>Edit</td>
<td>Create or change source data Terminal . : 3278</td>
</tr>
<tr>
<td>3</td>
<td>Utilities</td>
<td>Perform utility functions Screen . . : 1 Language . : ENGLISH</td>
</tr>
<tr>
<td>4</td>
<td>Foreground</td>
<td>Interactive language processing Appl ID . : ISR</td>
</tr>
<tr>
<td>5</td>
<td>Batch</td>
<td>Submit job for language processing TSO logon . : ISPFPROC</td>
</tr>
<tr>
<td>6</td>
<td>Command</td>
<td>Enter TSO or Workstation commands TSO prefix: FMUSER</td>
</tr>
<tr>
<td>7</td>
<td>Dialog</td>
<td>Perform dialog testing System ID . : FM02</td>
</tr>
<tr>
<td>8</td>
<td>LM Facility</td>
<td>Library administrator functions System ID . : USER</td>
</tr>
<tr>
<td>9</td>
<td>IBM Products</td>
<td>IBM program development products Release . : ISPF 5.0</td>
</tr>
<tr>
<td>10</td>
<td>SCLM</td>
<td>SW Configuration Library Manager</td>
</tr>
<tr>
<td>11</td>
<td>Workplace</td>
<td>ISPF Object/Action Workplace</td>
</tr>
<tr>
<td>12</td>
<td>OS/390 System</td>
<td>System OS/390 system programmer applications</td>
</tr>
<tr>
<td>13</td>
<td>OS/390 User</td>
<td>User OS/390 user applications</td>
</tr>
<tr>
<td>F</td>
<td>File Manager</td>
<td>File Manager z/OS and OS/390</td>
</tr>
<tr>
<td>FD</td>
<td>FM/DB2</td>
<td>File Manager/DB2</td>
</tr>
</tbody>
</table>

Parent panels
• None.
ISPF Primary Option panel

Child panels

To display this panel... Use/do this

"Primary Option Menu panel" on page 617 Select File Manager/DB2

Related tasks

ISPF Settings panel

You use the ISPF Settings panel display and modify selected ISPF parameters.

Parent panels

Related tasks

"ISPF settings (option 0.11)" on page 43

Label Column panel

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>Label Column</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

LABEL ON COLUMN

Table owner ....... BUDGER2 +
Table name ....... DEPT +
Column name ....... DEPTNAME

IS
Label ........... +

Command ===>
F1=Help F2=Split F3=Exit F4=Expand F7=Backward F8=Forward
F9=Swap F10=Left F11=Right F12=Cancel

Parent panels

- "Columns panel" on page 411
- "Constraint Columns for panel" on page 415
- "Columns in Table panel" on page 413

Child panels

None.

Related tasks
Label Table panel

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>Label Table</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**LABEL ON TABLE**
- **Owner**: BUDGER2 +
- **Name**: ERP +

**IS**
- **Label**: +

**Command ===>
F1=Help F2=Split F3=Exit F4=Expand F7=Backward F8=Forward
F9=Swap F10=Left F11=Right F12=Cancel

**Parent panels**
- “Tables, Views and Aliases panel” on page 704

**Child panels**
None.

**Related tasks**

**LISTCAT Output Display panel**

FM/DB2 displays the LISTCAT Output Display panel when you use the LC line command on the Recovery Information panel or the Table Space Parts panel.

**LISTDEF Options panel**

You use the LISTDEF Options panel to collect information used in generating LISTDEF statements and the proper JCL statement to access the LISTDEF library in a utility batch job.
**LISTDEF Options panel**

**Panel and field definitions**

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>LISTDEF Options</td>
<td>Global Settings</td>
<td></td>
</tr>
</tbody>
</table>

Enter RESET to restore installation defaults.

**LISTDEF Library Option:**

- **JCL DD statement:**
  - //SYSLISTD DD
  - // DD
  - // DD
  - // DD

**Initial Utility Statement Option:**

- **Utility statement:**

**Additional statements:**

- Enter "/* to select options
- Use additional statements
- Edit additional statements

**Command ====>**

F1=Help  F2=Split  F3=Exit  F4=Retrieve  F6=Reset  F7=Backward

F8=Forward  F9=Swap  F10=Actions  F12=Cancel

**JCL DD statement**

The information used to generate the JCL DD statement. This information is:

- **DD name**
  - A read-only field. This holds the value specified in the
  - **LISTDEFDD DD name** field of the OPTIONS Options panel
  - "OPTIONS Options panel" on page 612. If the library field is
  - blank then no DD statement is generated in the utility batch job.

- **Text values**
  - The values entered on these four lines are used directly to generate
  - the DD statement, provided that the **DD name** field is not blank.

**Utility statement**

This value is used to generate the initial utility statements. If the field is

- blank, and the Use additional statements option is not selected, then no
- LISTDEF statement is generated into the utility batch job. If the field is
- non-blank, then the specified LISTDEF is generated with any additional
- LISTDEF statements supplied when the Use additional statements option is
- selected.

**Use additional statements**

If this option is set then more than one LISTDEF statement may be

- generated into the utility batch job. Select the Edit additional statements
- option to edit additional statements. Only LISTDEF specifications that have
- non-blank input fields are used in the generation.

If this option is not selected, then only a non-blank value in the Utility

- statement field is used to generate the LISTDEF statement.

**Edit additional statements**

When you select this option and press Enter, the Additional LISTDEF

- Statements panel "Additional LISTDEF Statements panel" on page 367 is
- displayed. After you have entered values on this panel and exited the
LISTDEF Options panel

panel, this option is deselected. The additional statements are used when you select the Use additional statements option.

The LISTDEF Library Option panel has an installation-defined default. You can restore the default for the library option by entering the RESET command or pressing the RESET PF key. Each subsystem has its own installation-defined defaults. RESET also clears the Use additional statements option and any values in the Utility statement fields.

For more information, see LISTDEF and OPTIONS statements in DB2 Utility Guide and Reference.

Parent panels

Child panels

To display this panel... Use/do this

“Additional LISTDEF Statements panel” on page 367 Select the Edit additional statements option.

Related tasks

- “RESET primary command” on page 771
- “Setting options for DB2 utilities” on page 278
- “DB2 Utility LISTDEF options (option 3.UL)” on page 45

LOAD from panel

You use the LOAD Utility: “To” template panel to specify the template for the target DB2 table when you use the LOAD utility.

Panel and field definitions
LOAD from panel

Table owner
The name of the owner for the target table. You can specify a generic object
to select from a list. This field is mandatory.

Table name
The name of the target table. You can specify at generic object name to
select from a list. This field is mandatory.

Data set name
The name of the data set or PDS member containing the template for the
“To” object. It can be a fully-qualified data set name or a pattern. The
name may include a member name or name pattern in parenthesis. If the
member is specified here, the associated Member field must be empty.

Member
If you specified the name of a partitioned data set (PDS) without including
a member name or name pattern in parenthesis in the Data set name field,
then you can use this field to specify the member name or a member name
pattern.

Template usage
The LOAD Utility: “To” template panel provides four options for template
processing:

1. Above
   Requires that you enter the name of a template in the To Template
   section of the panel.

2. Previous
   Uses the last (previously used) template for this table.

3. Generate from table
   Generates a template based on the information in the DB2 catalog
   for the specified table. This is the default setting. If you have
   specified a data set name in the To Template section of the panel,
   it is ignored. FM/DB2 generates the required template in memory.

4. Generate/Replace
   Generates a template based on the information in the DB2 catalog
   for the specified table, and saves this information in the data set
   specified in the To Template section of the panel. If the template
   already exists, it is replaced and any prior customization is lost.

You can select the following options by entering a “/” or an “A”:

Edit load utility options
Displays the Load Utility Options panel

Edit template mapping
Use to edit the template for the table to which you are importing the data,
before importing the data.

Parent panels
• “LOAD Utility - Using Templates panel” on page 605

Child panels

<table>
<thead>
<tr>
<th>To display this panel...</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>“LOAD Utility Options panel” on page 607</td>
<td>Select Edit LOAD utility options</td>
</tr>
</tbody>
</table>
LOAD Utility panel

You use the LOAD Utility panel to specify details about the data you want to load.

Panel and field definitions

Data set name (Input Data)
A sequential or partitioned data set that contains the data to be loaded into the DB2 table. Specify a generic data set name to select from a list. It can be a fully-qualified data set name or a pattern. The name may include a member name or name pattern in parenthesis. If the member is specified here, the associated Member field must be empty.

Member (Input Data)
If you specified the name of a partitioned data set (PDS) without including a member name or name pattern in parenthesis in the Data set name field, then you can use this field to specify the member name or a member name pattern.

Volume
The volume serial number if the data set is not cataloged.

Data set name (Utility Control Statements)
A sequential or partitioned data set that contains the DB2 load utility control statements. Specify a generic data set name to select from a list. It can be a fully-qualified data set name or a pattern. The name may include
LOAD Utility panel

a member name or name pattern in parenthesis. If the member is specified here, the associated Member field must be empty.

Member (Utility Control Statements)
If you specified the name of a partitioned data set (PDS) without including a member name or name pattern in parenthesis in the Data set name field, then you can use this field to specify the member name or a member name pattern.

Table owner
The name of the owner for the target table. You can specify a generic object name to select from a list. This field is mandatory.

Database
A fully or partially-qualified database name to limit the list of object names to the specified databases.

Table name
The name of the target table. You can specify a generic object name to select from a list. This field is mandatory.

Table space
A fully or partially-qualified table space name to limit the list of object names to the specified table spaces.

1. The data is described by utility control statements
Use the supplied DB2 load utility control statements when generating a DB2 load utility batch job.

2. The data is described by a template or copybook
Use a template or copybook when generating a DB2 load utility batch job.

Parent panels
• “DB2 Utilities panel” on page 515

Child panels

To display this panel... Use/do this
“LOAD Utility - Using Templates panel” on page 605 Select option 2

Related tasks
• “LOAD utility” on page 286

LOAD Utility panel using DB2 templates

You use the LOAD Utility using DB2 templates panel to load new data into tables and build or extend any indexes defined on those tables.
LOAD Utility panel using DB2 templates

Panel and field definitions

```
Process Options Utilities Help

FM/DB2 (DFG2) LOAD Utility

Input Data:
  Template name... LODIVDDN

Target DB2 Table:
  + Database . . (optional)
  Owner . . . . . + Table space ______ (optional)
  Name ...... 

Command ===> F1=Help F2=Split F3=Exit F4=Expand F7=Backward F8=Forward

F9=Swap F10=Left F11=Right F12=Cancel
```

Parent panels

• “DB2 Utilities panel” on page 515

Child panels

To display this panel... Use/do this
```
“LOAD Utility - Using Templates panel” Press Enter
```

Related tasks

• “LOAD utility” on page 286

LOAD Utility - Using Templates panel

You use the LOAD Utility - Using Templates panel to specify the “From” copybook or template that describes the input data.
LOAD Utility - Using Templates panel

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2</td>
<td>LOAD</td>
<td>Utility</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>-</td>
<td></td>
</tr>
<tr>
<td>Input</td>
<td>Template</td>
<td>or Copybook:</td>
<td></td>
</tr>
<tr>
<td>Data</td>
<td>set</td>
<td>name</td>
<td>.</td>
</tr>
<tr>
<td>Member</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Processing Options:</td>
<td></td>
<td>Enter &quot;/&quot;, &quot;A&quot;ways to select option</td>
<td></td>
</tr>
<tr>
<td>1. Above</td>
<td>Member</td>
<td>View template</td>
<td></td>
</tr>
<tr>
<td>2. Previous</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Data set name
A sequential or partitioned data set that contains the data to be loaded into the DB2 table. Specify a generic data set name to select from a list. It can be a fully-qualified data set name or a pattern. The name may include a member name or name pattern in parenthesis. If the member is specified here, the associated Member field must be empty.

Member
If you specified the name of a partitioned data set (PDS) without including a member name or name pattern in parenthesis in the Data set name field, then you can use this field to specify the member name or a member name pattern.

Template usage
The LOAD Utility - Using Templates panel provides two options for template processing:

1. Above
   Requires that you enter the name of a template in the Input Data Template or Copybook section of the panel.

2. Previous
   Uses the last (previously used) template for this table.

You can select the following option by entering a ”/” or an “A”:

View template
Select this option to edit the “To” template.

Parent panels
- “LOAD Utility panel” on page 603
LOAD Utility Options panel

You use the LOAD Utility Options panel to specify the options that apply when you use the LOAD utility to transfer data.

These options are saved in your ISPF profile and persist from one FM/DB2 session to the next.

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>Load Utility Options</td>
<td>Global Settings</td>
<td></td>
</tr>
</tbody>
</table>

Utility Options:
Enter "/" to select option
Preformat unused pages
Log changes --NO--> Do not set COPY pending flag
ASCII data
Use IEEE floating point
Allow substitution chars
Enforce constraints
Resume (append new data) --NO--> Replace existing data
Build indexes in parallel --NO--> Reuse existing dataset
Number of index keys 0

CCSID Values:
SBCS data........0
Mixed DBCS data.....0
DBCS data........0

Discarded Records:
Maximum discards ....0 (0 means no limit)

Command ===>
F1=Help F2=Split F3=Exit F7=Backward F8=Forward F9=Swap
F12=Cancel

Preformat unused pages
Indicates whether the allocated file space for a table is preformatted to enable quicker INSERT processing.

Log changes
Indicates whether DB2 logs details in the DB2 log of the rows loaded.

Do not set COPY pending flag
If Log changes is not selected, indicates whether DB2 is not to set the table space in COPY pending status.
LOAD Utility Options panel

ASCII data
Indicates whether the input data is ASCII.

Use IEEE floating point
Indicates whether floating-point numbers are in IEEE Binary Floating Point (BFP) format.

Allow substitution chars
Indicates whether the LOAD utility accepts substitution characters in a string.

Enforce constraints
Indicates whether the LOAD utility enforces check constraints and referential constraints.

Resume (append new data)
Indicates whether records are loaded into an empty or non-empty table space, without existing data being deleted.

Replace existing data
If Resume (append new data) is not selected, indicates whether the table space and all its indexes are reset to empty before records are loaded.

Reuse existing dataset
If Resume (append new data) is selected, indicates whether the LOAD utility logically resets and reuses DB2-managed data sets without deleting and redefining them.

Build indexes in parallel
Indicates whether index keys are sorted in parallel during LOAD.

Number of index keys
Estimated number of index keys to sort.

SBCS data
The coded character set for SBCS data in the input file.

Mixed DBCS data
The coded character set for mixed DBCS data in the input file.

DBCS data
The coded character set for DBCS data in the input file.

Maximum discards
The maximum number of discarded records allowed before LOAD abends.

Note: FM/DB2 does not attempt to validate every possible combination of load utility options. In particular, some combinations of options fail given the current state of the DB2 object being loaded. This can mean that the DB2 load utility job will fail if an invalid combination of options has been specified. If this occurs examine the failing messages for the DB2 LOAD utility job and respond accordingly.

Parent panels

Child panels
None.
Manage DB2 Privileges panel

You use the Manage DB2 Privileges panel to display, grant, and revoke DB2 privileges.

**Panel and field definitions**

<table>
<thead>
<tr>
<th>Location</th>
<th>The location of the objects you want listed.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Owner</td>
<td>The name of the person who created the DB2 object.</td>
</tr>
<tr>
<td>Name</td>
<td>The name of the DB2 object.</td>
</tr>
<tr>
<td>In</td>
<td>An optional field you can use to qualify certain object types.</td>
</tr>
</tbody>
</table>

**Object Type**

- 1. Database
- 2. Table space
- 3. Table
- 4. Column
- 5. Plan
- 6. Package
- 7. Collection
- 8. Schema
- 9. Type
- 10. Function
- 11. Stored procedure
- 12. Storage group
- 13. Buffer pool
- 14. System
- 15. Sequence

**Related tasks**

- “DB2 LOAD utility options (option 3.L)” on page 45
- “LOAD utility” on page 286

**Parent panels**

- “Utility Functions panel” on page 720

**Child panels**

<table>
<thead>
<tr>
<th>To display this panel...</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>“Privileges panels” on page 624</td>
<td>Blank command line</td>
</tr>
<tr>
<td>“Grant privileges panels” on page 587</td>
<td>“G” on command line</td>
</tr>
<tr>
<td>“Revoke privileges panels” on page 647</td>
<td>“R” on command line</td>
</tr>
</tbody>
</table>
Manage DB2 Privileges panel

Related tasks
- Chapter 10, “Managing DB2 privileges,” on page 271

Object List Line Commands panel

FM/DB2 displays the Object List Line Commands panel if you use the ? line command against an item listed on one of these panels:
- “Application Plans panel” on page 389
- “Collections panel” on page 396
- “Columns panel” on page 411
- “Database Request Modules panel” on page 498
- “Databases panel” on page 500
- “Distinct Types panel” on page 523
- “Functions panel” on page 584
- “Indexes panel” on page 595
- “Application Packages panel” on page 387
- “Schemas panel” on page 657
- “Stored Procedures panel” on page 685
- “Synonyms panel” on page 687
- “Table Spaces panel” on page 698
- “Tables, Views and Aliases panel” on page 704
- “Triggers panel” on page 712

Related tasks
- “Line command query (?)” on page 253

Object List Utility panel

You use the Object List Utility panel to specify details about the objects you want to list or print.
Object List Utility panel

Panel and field definitions

**Location**
The location of the objects you want listed

**Owner**
The name of the person who created the DB2 object

**Name**
The name of the DB2 object

**Database/collect/schema**
The name of a database, a collection, or a schema

**Column**
The name of column in the relevant DB2 system catalog table

**Operator**
A valid SQL operator

**Value**
The value to be compared with the contents of the column

**Object Type**
The type of DB2 object that you want listed

**Confirm object drop**
Indicates if you want FM/DB2 to display a confirmation panel each time you enter a DR(op) command against an object

**Parent panels**
- "Utility Functions panel" on page 720
Object List Utility panel

Child panels

To display this panel... Use/do this

“Databases panel” on page 500 Object Type 1
“Table Spaces panel” on page 698 Object Type 2
“Tables, Views and Aliases panel” on page 704 Object Type 3
“Indexes panel” on page 595 Object Type 4
“Columns panel” on page 411 Object Type 5
“Synonyms panel” on page 687 Object Type 6
“Application Plans panel” on page 389 Object Type 7
“Application Packages panel” on page 387 Object Type 8
“Collections panel” on page 396 Object Type 9
“Database Request Modules panel” on page 498 Object Type 10
“Schemas panel” on page 657 Object Type 11
“Distinct Types panel” on page 523 Object Type 12
“Functions panel” on page 584 Object Type 13
“Stored Procedures panel” on page 685 Object Type 14
“Triggers panel” on page 712 Object Type 15

Related tasks
• Chapter 8, “Working with lists of DB2 objects,” on page 241

OPTIONS Options panel

You use the OPTIONS Options panel to collect information used in generating an OPTIONS statement in a utility batch job.

Panel and field definitions

<table>
<thead>
<tr>
<th>Process Options Utilities Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2) OPTIONS Options Global Settings</td>
</tr>
</tbody>
</table>

Processing Options:

RESET restores installation options

1. Do not create
2. OFF
3. KEY
4. Use the below options

Enter "/" to select options

PREVIEW
LISTDEFDD DD name SYSLISTD
TEMPLATEDDD DD name SYSTEMPL
EVENT( Event spec )
Event spec:

Command ===>  F1=Help  F2=Split  F3=Exit  F4=CRetrieve  F6=Reset  F7=Forward  F8=Forward  F9=Swap  F10=Actions  F12=Cancel
OPTIONS Options panel

Processing Options
The options in capitals are keywords that are used in the OPTIONS statement if the option is selected. For more information, see OPTIONS statement in DB2 Utility Guide and Reference.

The processing options are:

1. **Do not create**
   When this option is selected, FM/DB2 does not create the OPTIONS statement in a utility batch job.

2. **OFF**
   When this option is selected, FM/DB2 restores all default options in a utility batch job.

3. **KEY**
   When this option is selected, FM/DB2 uses the KEY clause in the OPTIONS statement.
   
   You should only use this option when instructed to do so by IBM Service. Enter the value provided by IBM Service in the **KEY** field.

4. **Use the below options:**
   When this option is selected, you can select other keywords for addition to the OPTIONS statement. The relevant keyword is only added to the OPTIONS statement if the option is selected. The options are:

**PREVIEW**
When this option is selected, the generated utility batch job, when executed, runs in PREVIEW mode. This option corresponds to the PREVIEW keyword in an OPTIONS statement.

**LISTDEFDD**
When this option is selected, it specifies the DD name of the LISTDEF definition library. The default DD name for the input field is “SYSLISTD”. You can enter a different DD name. The specified DD name appears as a read-only field in the [LISTDEF Options panel](#) on page 599.

This option corresponds to the LISTDEFDD clause in an OPTIONS statement.

**TEMPLATEDD**
When this option is selected, it specifies the DD name of the TEMPLATE definition library. The default DD name for the input field is “SYSTEMPL”. You can enter a different DD name. The specified DD name appears as a read-only field in the **TEMPLATEDD Options (1 of 2) panel** on page 707.

This option corresponds to the TEMPLATEDD clause in an OPTIONS statement.

**EVENT**
When this option is selected, the utility batch job includes the specified events or events. Enter the events on the following lines. You must enter at least one. The values you enter are placed immediately after the EVENT keyword in the OPTIONS statement, without further validation by FM/DB2.
OPTIONS Options panel

This option corresponds to the EVENT clause in an OPTIONS statement.

All fields have installation-defined defaults. You can reset all values back to the installation defaults (site policy) at will by using the RESET command or RESET PF Key. Each subsystem has its own installation-defined defaults.

As an example, the following clauses are added to any generated OPTION statement when the options in the figure are selected:

- LISTDEFDD SYSLISTD
- TEMPLATEDD SYSTEMPL

For the DB2 defaults settings of SYSLISTD for LISTDEFDD and SYSTEMPL for TEMPLATEDD, it is also acceptable to "optimize", which means to not generate these clauses, since they are the DB2 defaults.

Parent panels

- 

Child panels

None.

Related tasks

- "RESET primary command" on page 771
- "Setting options for DB2 utilities" on page 278
- "DB2 Utility OPTIONS options (option 3.UO)" on page 45

Package Dependencies panel

FM/DB2 displays the Package Dependencies panel if you use the DEP line command against an item listed on the "Application Packages panel" on page 387.

Child panels

None.

Related tasks

Package List panel

FM/DB2 displays the Package List panel if you use the PKL line command against an item listed on:

- The "Application Plans panel" on page 389, or
- The "Application Packages panel" on page 387, or
- The "Collections panel" on page 396

Related tasks

Parents of Table panel

FM/DB2 displays the Parents of Table panel if you use the PAR line command against an item listed on the "Tables, Views and Aliases panel" on page 704.
Partitioned Table Statistics panel

FM/DB2 displays the Partitioned Table Statistics panel if you use the PST line command against an item listed on the "Tables, Views and Aliases panel" on page 704.

Plan Dependencies panel

FM/DB2 displays the Plan Dependencies panel if you use the DEP line command against an item listed on the "Application Plans panel" on page 389.

Plan Enabled/Disabled Connections panel

FM/DB2 displays the Plan Enabled/Disabled Connections panel if you use the EDC line command against an item listed on the "Application Plans panel" on page 389.

Plan Table Rows panel

FM/DB2 displays the Plan Table Rows panel if you:
- Select option L (List plan table rows) on the Explain Utilities panel, or
- Enter the EXPLAIN primary command on the Enter, Execute and Explain SQL Statements with the Show results option selected.

Use this panel to display rows from a plan table.

Use this panel to display rows from a plan table.

Command ===> Scroll PAGE
F1=Help    F2=Split    F3=Exit    F4=Expand    F7=Backward    F8=Forward
F9=Swap    F10=Left    F11=Right    F12=Cancel

Figure 114. Plan Table Rows panel (collection ID format)

The Plan Table Rows panel shows you rows from a plan table:

C Input field where you can enter one of the following line commands:
D Delete row
DK Delete rows for package
DP Delete rows for plan
DQ Delete rows for query number
Plan Table Rows panel

FR  Show function table rows. FM/DB2 displays the Function Table Rows panel (not shown).
I  Interpret EXPLAIN data
IH  Insert optimizer hint
K  Show package
M  Show DBRM
P  Show plan
SR  Show statement table rows. FM/DB2 displays the Statement Table Rows panel (see “Statement Table Rows panel” on page 681).
T  Show table
UH  Update optimizer hint
X  Show index

Query Number
A number that identifies the SQL statement.

Q Bl Query block number: the number reflects the query’s order of appearance in the EXPLAIN statement.

Collect.
Collection ID or blank for dynamic EXPLAIN. This information only is displayed for the COL format of the panel.

Applname
PLAN name or blank for dynamic EXPLAIN. This information only is displayed for the PLAN format of the panel.

Hint ID
Optimization hint ID or blank for dynamic EXPLAIN. This information only is displayed for the HINT format of the panel.

Progname
DBRM/package name. This information only is displayed for the COL and PLAN formats of the panel.

Hint Used
The optimization hint used or blank for dynamic EXPLAIN. This information only is displayed for the HINT format of the panel.

Pl No Plan number. Indicates order in which the statement is run.

M T Method: Join method that is used.
Ac Ty Access type. Method of accessing the table.
M Co Matching columns. Number of index keys used in an index scan.
I O Index only. Whether only the index is accessed in this step.
T No Table number. Position of the table in the statement.

Table Owner
Owner of the table being accessed.

Table Name
Name of the table being accessed.

Parent panels
• “Explain Utilities panel” on page 556
Plan Table Rows panel

Child panels

To display this panel... Use/do this

“EXPLAIN Interpretation panel” on page 556 Line command I
“Set Optimizer Hint panel” on page 665 Line commands IH, UH
“Application Packages panel” on page 387 Line command K
“Statement Table Rows panel” on page 681 Line command SR
“Tables, Views and Aliases panel” on page 704 Line command T

Related tasks

- “Explaining SQL” on page 353
- “Managing tables used by SQL explain” on page 355

Related references

- “COL primary command” on page 736
- “PLAN primary command” on page 766
- “HINT primary command” on page 756

Primary Key for Table panel

FM/DB2 displays the Primary Key for Table panel if you use the PK line command against an item listed on the “Tables, Views and Aliases panel” on page 704.

Primary Option Menu panel

You use the Primary Option Menu panel to select the FM/DB2 function you want to perform.

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DF72)</td>
<td>Primary Option Menu</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0 Settings</td>
<td>Set processing options</td>
<td>User ID : KEISTE2</td>
<td></td>
</tr>
<tr>
<td>1 View</td>
<td>View DB2 object</td>
<td>System ID : FMD2</td>
<td></td>
</tr>
<tr>
<td>2 Edit</td>
<td>Edit DB2 table</td>
<td>Appl ID : FMN2</td>
<td></td>
</tr>
<tr>
<td>3 Utilities</td>
<td>Perform utility functions</td>
<td>Version : 13.1.0</td>
<td></td>
</tr>
<tr>
<td>4 SQL</td>
<td>Prototype, execute and analyze SQL</td>
<td>Terminal : 3278</td>
<td></td>
</tr>
<tr>
<td>5 DB2I</td>
<td>Start DB2 Interactive</td>
<td>Screen : 1</td>
<td></td>
</tr>
<tr>
<td>6 Command</td>
<td>Enter and execute a DB2 Command</td>
<td>Date : 2013/10/04</td>
<td></td>
</tr>
<tr>
<td>7 Template</td>
<td>Template utilities</td>
<td>Time : 11:22</td>
<td></td>
</tr>
<tr>
<td>X Exit</td>
<td>Terminate FM/DB2</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

DB2 SSID : DF72
SQL ID : KEISTE2

Command =>
F1=Help F2=Split F3=Exit F4=Expand F7=Backward F8=Forward
F9=Swap F10=Left F11=Right F12=Cancel

Chapter 15. FM/DB2 panels and fields  617
Print Audit Trail panel

You use the Print Audit Trail panel to specify the name of the Audit Trail data set to be used to generate an Audit Trail report.

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>Print Audit Trail</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Audit Trail:
- Data set name
- Description

Processing Options:
- Enter "/" to select option
- / Print only changed columns
- Keep data set after printing
- 7 Browse report
- 8 Batch execution

Command ===> F1=Help F2=Split F3=Exit F7=Backward F8=Forward F9=Swap

Data set name
The name of the Audit Trail data set to be used.
Print Audit Trail panel

Description
A description for the report. If the descriptions contains blanks, you must
enclose it in quotes. This field is optional.

You can select the following options by entering a “/” in the selection field:

Print only changed columns
Limits the amount of printed lines.

Keep data set after printing
Whether to keep or delete the data set after it is printed.

Browse report
The audit trail report is displayed using Print Browse (option 3.11).

Batch execution
Generates JCL for printing the audit trail report using a batch job. For
command syntax, see the AUD function in File Manager User's Guide and
Reference.

Hex Format
To produce an UPDOWN hexadecimal display below the standard field
display.

Non-display Hex
To produce an UPDOWN hexadecimal display below the standard field
display only for fields that contain non-displayable \ characters.

Highlight changes
Highlight the changed fields. An asterisk is placed to left of the before data
to indicate the field has been changed.

Parent panels
• “Utility Functions panel” on page 720

Child panels
None.

Related tasks
• “Printing an audit trail report” on page 267

Print Utility Options panel

You use the Print Utility Options panel to set various options for the Print utility.
Panel and field definitions

Print Mode
The format of the print output:

1. Table
In Table print mode (the default mode), the rows of the table are formatted into columns and printed horizontally across the page. Each column has a heading of either the column name or a heading specified by you. Multiple rows are printed on each page. Each column of the table is formatted according to the data type for that column. For example, numeric data types are printed as whole or decimal numbers, and date and time columns are printed as dates or times. Print data might be truncated, depending on the LRECL of the print data set.

2. Single
In Single mode, only one column of the table is printed on each line of the page. The columns of the table are arranged vertically on the page, with the column header on the left of the page, and the data for that column on the right. Each column of the table is formatted according to the data type for that column. For example, numeric data types are printed as whole or decimal numbers, and date and time columns are printed as dates or times.

Display
Null Column Display Indicator. This option, applying to columns that allow DB2 nulls only, allows you to specify a character to represent the DB2 null value for the FM/DB2 print utility.

Batch execution
Builds the necessary JCL to perform the Print function in a batch job. The generated JCL is displayed using the ISPF editor, where you can review the JCL and submit the job for batch processing.

Use uncommitted read
Enables data to be read that has not yet been committed. The default setting is to not read uncommitted data.
Print Utility Options panel

Note: If FM/DB2 attempts to retrieve data that has a write lock, an SQL error might result.

Print HEX representation
Prints the hexadecimal representation of the data.

Parent panels
- "Set Processing Options panel" on page 671
- "Print Utility panel"

Child panels
None.

Related tasks
- "Printing the contents of a DB2 table" on page 261
- "Selecting options on FM/DB2 panels" on page 24

Print Utility panel

You use the Print Utility panel to print the contents of a table or view.

Panel and field definitions

<table>
<thead>
<tr>
<th>Location</th>
<th>Database . . (optional)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Owner . . .</td>
<td>Table space . . (optional)</td>
</tr>
<tr>
<td>Name . . .</td>
<td>SYSIBM + Row space +</td>
</tr>
<tr>
<td>Row count . . .</td>
<td>20 Number of rows to print</td>
</tr>
</tbody>
</table>

DB2 Template:
Data set name ...
Member . . . .

Processing Options:

Print Mode Template usage Enter "/", "A"lways to select
3 1. Above Edit options
2. Previous Edit template
3. Generate from table Create audit trail (+)
4. Generate/Replace

Command ===>
F1=Help F2=Split F3=Exit F4=Expand F7=Backward F8=Forward
F9=Swap F10=Left F11=Right F12=Cancel

Location
The location of the DB2 object. Leave this field blank if the table or view that you want to work with is at your current location. This field is optional.

database
The name of the database to which you want to restrict your selection. This field is optional.

Owner
The authorization ID of the owner of the DB2 object. This field is optional.
Print Utility panel

Table space
The name of the table space to which you want to restrict your selection.
This field is optional.

Name
The name of the DB2 object.

Row count
The number of rows of the DB2 object you want to print.
Valid values:
0       All rows printed
1–99999999       Number of rows printed
ALL       All rows printed

Data set name
The name of the template data set. It can be a fully-qualified data set name or a pattern. The name may include a member name or name pattern in parenthesis. If the member is specified here, the associated Member field must be empty.

Member
If you specified the name of a partitioned data set (PDS) without including a member name or name pattern in parenthesis in the Data set name field, then you can use this field to specify the member name or a member name pattern.

Processing Options
You can use these options to specify, amongst other things, the format in which you want the data printed, which template you want to use for the print process, and whether you want to edit the template before the print process.

Template usage
The Print Utility panel provides four options for template processing:

1. Above
   Requires that you enter the name of a template data set (and optionally a member name) in the Template section of the panel. If you do not specify a member name, or if you specify a pattern, FM/DB2 displays a member selection list. After you have specified the member containing a template, processing continues using this template in place of the automatically-generated template for the table (for a detailed explanation, see Template Usage option 3).

2. Previous
   Uses the last (previously used) template for this table.

3. Generate from table
   Generates a template based on the information in the DB2 catalog for the specified table. This is the default setting. If you have specified a member name in the Template section of the panel, it is ignored. FM/DB2 generates the required template in memory.

4. Generate/Replace
   Generates a template based on the information in the DB2 catalog for the specified table, and saves this information
Print Utility panel

in the member specified in the Template section of the panel. If the member already exists, it is replaced. Any prior customization is lost.

You can select the following options by entering a “/” or an “A”:

Edit options
Use to display the FM/DB2 print utility options, prior to printing the data for the DB2 object.

Edit template
Use to edit the template before printing the data.

Create audit trail
Controls whether access to DB2 by FM/DB2 is written to the FM/DB2 audit log. This option only appears when SAF-rule controlled auditing is in effect.

FM/DB2 may write audit records to either SMF, or a data set.

When SAF rules are used to control FM/DB2 auditing, the decision to write audit records is based on the DB2 object name being processed, and the current userid (TSO logonid). Therefore auditing may be required when processing one DB2 object, but not another. In addition, it is possible that the auditing requirement cannot be changed for a particular DB2 object; this might be different for another DB2 object, and the requirement may be different for other TSO userids.

If you do not have the authority to change the audit setting when processing a particular DB2 object, FM/DB2 prompts you if you enter the wrong setting. If you do have the authority to set the audit setting for a particular DB2 object, FM/DB2 honors whatever value is specified in this field.

Parent panels
- “Utility Functions panel” on page 720
- “Tables, Views and Aliases panel” on page 704

Child panels

<table>
<thead>
<tr>
<th>To display this panel...</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>“Print Utility Options panel” on page 619</td>
<td>Select Edit options</td>
</tr>
<tr>
<td>“Column Selection/Edit panel” on page 407</td>
<td>Select Edit template</td>
</tr>
</tbody>
</table>

Related tasks
- “Printing the contents of a DB2 table” on page 261
- “Selecting options on FM/DB2 panels” on page 24
- Chapter 3, “Working with templates,” on page 49
- “SAF-rule controlled auditing” on page 47
Privilege details panels

These panels are:
- Application Plan Privilege Details panel
- Column Privilege Details panel
- Database Privilege Details panel
- Distinct Type Usage Privilege Details panel
- Function Privilege Details panel
- Package Privilege Details panel
- Schema Privilege Details panel
- Stored Procedure Privilege Details panel
- Table Space Use Privilege Details panel
- Table/View Privilege Details panel

FM/DB2 displays one of these panels when you specify the I line command against an object displayed in a privilege list of DB2 objects.

Privileges panels

These panels are:
- Application Plan Privileges panel
- Collection Privileges panel
- Column Privileges panel
- Database Privileges panel
- Distinct Type Privileges panel
- Function Privileges panel
- Package Privileges panel
- Schema Privileges panel
- Stored Procedure Privileges panel
- Table Privileges panel
- Table Space Use Privileges panel

FM/DB2 displays one of these panels if you use the Privileges utility function (3.5) to display privilege details for an object type, or if you specify the P line command against an object displayed in a list of DB2 objects.

Format of privilege display panels

All of the privilege display panels use an ISPF table display of information retrieved from the DB2 catalog. You can scroll through the information in a typical ISPF manner. The information is presented in read-only format.

For each privilege display panel:
- You can use the **Cmd** field to enter a line command. The line commands you can enter vary according to the object type. To display a list of valid line commands for the type of objects you are viewing, enter a “?” in the **Cmd** field.
- Single-character fields show the current status for each privilege that can apply to the object type listed:
  - **Blank**  The grantee does not have the specified privilege.
  - **Y**  The grantee has the specified privilege, but cannot grant this privilege to other users.
  - **G**  The grantee has the specified privilege, and can grant this privilege to other users.
Privileges panels

- At the top of each column of data, immediately below the column name, there are input fields. When the panel is initially displayed, all of these input fields contain an asterisk (*) indicating “all values”. You can use one or more of these input fields to restrict the list of objects listed. For example, by specifying BRA in the entry field below the Grantor column name on the Database Privileges panel shown in “Privileges panels” on page 624, the list of databases is restricted to those where the name of the grantor starts with “BRA”.

Related tasks
- “Displaying privileges” on page 275
- Chapter 8, “Working with lists of DB2 objects,” on page 241

Rebind Application Plan panel

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

FM/DB2 (DFG2) Rebind Application Plan
Verify rebind parameters:

REBIND PLAN( Plan name .. FMN2PLN3 )
  OWNER( Plan owner . KEISTEW +
  QUALIFIER( Qualifier .. KEISTEW + (qualifier for unqualified SQL)
  PKLIST( Package lists .
          (use ? to get current values from the catalog)
  NOPKLIST (Yes or No to remove current package list)
  DEFER(PREPARE) (Yes or No, used for distributed dynamic SQL)
  VALIDATE( Validation time . B (Run or Bind, Bind preferred)
  ISOLATION(                |

Figure 115. Rebind Application Plan panel (1 of 2)
Rebind Application Plan panel

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Isolation level</td>
<td>CS</td>
<td>(CS, RR, RS, or UR)</td>
</tr>
<tr>
<td>Cache</td>
<td>3072</td>
<td>(Cache size in bytes for authorization IDs)</td>
</tr>
<tr>
<td>Acquire</td>
<td></td>
<td>(Use or Allocate, Use preferred)</td>
</tr>
<tr>
<td>Release</td>
<td></td>
<td>(Commit or Deallocate, Commit preferred)</td>
</tr>
<tr>
<td>Explain</td>
<td></td>
<td>(Yes or No to explain access path)</td>
</tr>
<tr>
<td>Inhibit blocking</td>
<td></td>
<td>(Yes or No)</td>
</tr>
<tr>
<td>Server name</td>
<td></td>
<td>(blank=local, else first location)</td>
</tr>
<tr>
<td>Enable connects</td>
<td></td>
<td>(use to get current values from the catalog)</td>
</tr>
<tr>
<td>Disable connects</td>
<td></td>
<td>(use to get current values from the catalog)</td>
</tr>
<tr>
<td>En/disable names</td>
<td></td>
<td>(use to get current values from the catalog)</td>
</tr>
<tr>
<td>Parallelism</td>
<td>1</td>
<td>(1 or ANY)</td>
</tr>
<tr>
<td>Connect2 rules</td>
<td>D</td>
<td>(DB2 or STD)</td>
</tr>
<tr>
<td>Disconnect</td>
<td></td>
<td>(Explicit, Automatic or Conditional)</td>
</tr>
<tr>
<td>Dynamic rules</td>
<td></td>
<td>(Run or Bind)</td>
</tr>
<tr>
<td>Keep dynamic SQL</td>
<td>N</td>
<td>(Yes or No)</td>
</tr>
<tr>
<td>Defer/prepare</td>
<td></td>
<td>(Yes or No)</td>
</tr>
<tr>
<td>Re-opt at exec</td>
<td>N</td>
<td>(Yes or No)</td>
</tr>
<tr>
<td>3 part protocol</td>
<td>D</td>
<td>(DRDA or Private)</td>
</tr>
<tr>
<td>Access path hint</td>
<td></td>
<td>(Hint id)</td>
</tr>
<tr>
<td>UDF/UDF/STP path</td>
<td></td>
<td>(Hint id)</td>
</tr>
</tbody>
</table>

Command: F1=Help F2=Split F3=Exit F4=Expand F7=Backward F8=Forward F9=Swap F10=Left F11=Right F12=Cancel

Figure 116. Rebind Application Plan panel (2 of 2)

Parent panels
- "Application Plans panel" on page 389

Child panels
None.

Related tasks

---

File Manager/DB2 Data V13R1 User's Guide
Rebind Package panel

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>Rebind Package</td>
<td>More: +</td>
<td></td>
</tr>
</tbody>
</table>

Verify rebind parameters:

REBIND PACKAGE(
Location . . . .
Collection . . . . DSNUTIL +
Package . . . . DSNUGSQL +
(
Version . . . .
) OWNER(
Package Owner . . . ARROWSB +
) QUALIFIER(
Qualifier . . . ARROWSB +
) VALIDATE(
Validation time . B (Run or Bind, Bind preferred)
) ISOLATION(
Isolation level . CS (CS, RR, RS, or UR)
) RELEASE(
Release at . . . .
) EXPLAIN(
Explain plan . . N (Yes or No)
) CURRENTDATA(
Inhibit blocking . N (Yes or No)
) ENABLE(
Enable connects .
) DISABLE(
Disable connects .
) En/disable names .
) DEGREE(
Parallellism . . 1 (1 or ANY)
) DYNAMICRULES(
Dynamic rules . .
) KEEP DYNAMIC(
Keep dynamic SQL . N (Yes or No)
) DEFER(PREPARE)/NODEFER(PREPARE)
) Defer prepare . . (Yes or No)
) REOPT(VARS)/NOREOPT(VARS)
) Re-opt at exec . . N (Yes or No)
) 3 part protocol . D (DRDA or Private)
) OPT HINT(
Access path hint .
) PATH(
UDT/UDF/STP path .
) Command ===> F1=Help F2=Split F3=Exit F4=Expand F7=Backward F8=Forward F9=Swap F10=Left F11=Right F12=Cancel

Parent panels

- “Application Packages panel” on page 387

Child panels

None.

Related tasks
REBUILD (Indexes for Table Spaces) panel

You use the REBUILD (Indexes for Table Spaces) panel to generate utility control statements using the REBUILD INDEX (ALL) TABLESPACE form of REBUILD utility.

Panel and field definitions

| Sel | Selection column. To select a table space, type S in this column. Selected table spaces show an asterisk (*) in the PRC column. |
| PRC | Processing indicator. An asterisk (*) indicates the item is selected. |
| Database | The name of the database in which the table space is defined. |
| Table Space | The name of the table space. |
| Part'n Number | The partition number for the table space. This is 0 for a non-partitioned table space. For a partitioned table space, the number represents the partition of the table space. There is one row in the table for each partition of a partitioned table space. This is a display-only field. |
| Reuse | Controls whether DB2 deletes and redefines the index space data set as part of the rebuild operation: YES Keeps the existing data set. NO Deletes and redefines the index space data set. |
| WORKDDN | The ddname for the optional temporary work file used by the utility job. The value displayed is either SYSUT1, or another value if you have changed the default ddname. You can type a new value. |
| WORKDDN Template | Indicates whether the value of the associated WORKDDN field is a template name. |
Sort Device Type
The device type used by the utility to allocate sort data sets. The value displayed is either SYSALLDA, or another value if you have changed the default device type. You can type a new value.

Sort Num
The number of sort data sets used by the utility. The value displayed is either 4, or another value if you have changed the default setting. You can type a new value.

Sort Keys
Controls whether the REBUILD utility rebuilds indexes in parallel:
YES Builds the indexes in parallel.
NO Does not build the indexes in parallel.

You can type a new value.

Parent panels
- “DB2 Utilities panel” on page 515

Child panels
None.

Related tasks
- “REBUILD utility” on page 292
- “REBUILD utility (indexes for table spaces)” on page 293
- “Loading data described by a copybook or template” on page 289

REBUILD (Indexes for Table Spaces) with LISTDEF panel

You use the REBUILD (Indexes for Table Spaces) with LISTDEF panel to generate utility control statements using the REBUILD INDEX TABLESPACE form of the REBUILD utility.
Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>REBUILD (Indexes for Table Spaces)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>P</th>
<th>R</th>
<th>LISTDEF</th>
<th>Sort</th>
</tr>
</thead>
<tbody>
<tr>
<td>C</td>
<td>Name</td>
<td>Reuse</td>
<td>WORKDDN</td>
</tr>
<tr>
<td></td>
<td>LDEF1</td>
<td>N</td>
<td>SYSUT1</td>
</tr>
<tr>
<td>Y</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

An asterisk (*) in the PRC column shows that the LISTDEF in the row is selected. The S command toggles the selection of rows; unselected rows become selected and selected rows become unselected.

LISTDEF Name
On entry to the panel, this field is set to the value in the LISTDEF name field in the DB2 Utilities panel.

Parent panels
- “DB2 Utilities panel” on page 515

Child panels
None.

Related tasks
- “REBUILD utility” on page 292
- “REBUILD utility (indexes for table spaces)” on page 293

REBUILD Utility (Indexes) panel

You use the REBUILD Utility (Indexes) panel to generate utility control statements using the REBUILD INDEX (index_name) form of the REBUILD utility.
Panel and field definitions

Cmd  Selection column. To select an index, type S in this column. Selected indexes show an asterisk (*) in the PRC column.

PRC  Processing indicator. An asterisk (*) indicates the item is selected.

Index Owner  
The name of the owner of the index.

Index Name  
The name of the index.

Part'n Number  
The partition number for the index. This is 0 for a non-partitioned index. For a partitioned index, the number represents the partition of the index. There is one row in the table for each partition of a partitioned index. This is a display-only field.

Reuse  
Controls whether DB2 deletes and redefines the index data set as part of the rebuild operation:

   YES  Keeps the existing data set.
   NO  Deletes and redefines the index data set.

WORKDDN  
The ddname for the optional temporary work file used by the utility job. The value displayed is either SYSUT1, or another value if you have changed the default ddname. You can type a new value.

WORK DDN Template  
Indicates whether the value of the associated WORKDDN field is a template name.

Sort Device Type  
The device type used by the utility to allocate sort data sets. The value displayed is either SYSALLDA, or another value if you have changed the default device type. You can type a new value.
REBUILD Utility (Indexes) panel

Sort Num
The number of sort data sets used by the utility. The value displayed is either 4, or another value if you have changed the default setting. You can type a new value.

Sort Keys
Controls whether the utility rebuilds indexes in parallel:
YES  Builds the indexes in parallel.
NO   Does not build the indexes in parallel.

Parent panels
- “DB2 Utilities panel” on page 515

Child panels
None.

Related tasks
- “REBUILD utility” on page 292
- “REBUILD utility (indexes)” on page 292
- “Loading data described by a copybook or template” on page 289

REBUILD (Indexes) with LISTDEF panel

You use the REBUILD (Indexes) with LISTDEF panel to generates utility control statements using the REBUILD INDEX TABLESPACE form of the REBUILD utility.

Panel and field definitions

<table>
<thead>
<tr>
<th>Cmd</th>
<th>P</th>
<th>R</th>
<th>LISTDEF</th>
<th>WORKDDN</th>
<th>Device</th>
<th>Sort</th>
<th>Type</th>
<th>Sort</th>
<th>Num</th>
<th>Keys</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Y</td>
<td>FRED</td>
<td>N</td>
<td>SYST1</td>
<td>N</td>
<td>SYSLDA</td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Y</td>
<td></td>
<td></td>
<td>GEORGE</td>
<td>N</td>
<td>SYST1</td>
<td>N</td>
<td>SYSLDA</td>
<td>4</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

******************************* Bottom of data ********************************

Command ===> Scroll PAGE
F1=Help  F2=Split  F3=Exit  F7=Backward  F8=Forward  F9=Swap
F12=Cancel

Cmd  You can enter commands in this field to manipulate the panel rows.
I(n)  Insert (n) new row(s)
R(n)  Repeat the current row (n times)
REBUILD (Indexes) with LISTDEF panel

D(n)  Delete (n) row(s)
S(n)  Select (n) rows(s)

An asterisk (*) in the PRC column shows that the LISTDEF in the row is selected. The S command toggles the selection of rows; unselected rows become selected and selected rows become unselected.

LISTDEF Name
On entry to the panel, this field is set to the value in the LISTDEF name field in the DB2 Utilities panel.

Parent panels
• “DB2 Utilities panel” on page 515

Child panels
None.

Related tasks
• “REBUILD utility” on page 292
• “REBUILD utility (indexes)” on page 292

Record Type Selection panel
Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>File Manager</td>
<td>Record Type Selection</td>
<td>Line 1 of 2</td>
<td></td>
</tr>
<tr>
<td>Cmd</td>
<td>SIE Field Name</td>
<td>Prompt</td>
<td>Offset</td>
</tr>
<tr>
<td>**** Top of data ****</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IE</td>
<td>REC-TYPE01</td>
<td>0</td>
<td>80</td>
</tr>
<tr>
<td>SI</td>
<td>REC-TYPE02</td>
<td>0</td>
<td>80</td>
</tr>
<tr>
<td>**** End of data ****</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Parent panels
• “LOAD Utility - Using Templates panel” on page 605

Child panels
None.

Related tasks
RECOVER Utility (Index Spaces) panel

You use the RECOVER Utility (Index Spaces) panel to generate utility control statements using the RECOVER INDEXSPACE form of the RECOVER utility. You can only copy and recover indexes that have been created with the COPY YES clause.

Panel and field definitions

The fields in the ISPF table part of the panel are:

**Sel**  Selection column. Type S to select an index.

**PRC**  Processing column. An asterisk (*) indicates the selected index spaces.

**Database**  The name of the database in which the index space is defined.

**Index Space**  The name of the index space.

**DS Num**  The partition number for this table space. Non-partitioned table spaces have a partition number of 0. For a partitioned table space, the number indicates the partition of the table space.

**ICDATE**  The date that the image copy was taken. The format is **yyymmdd**.

**ICTIME**  The time that the image copy was taken. The format is **hHmmss**.

**Reuse**  Reuse option:

- **Y**  The data set for the index space is not deleted and is re-defined as part of the RECOVER utility processing.

- **N**  The data set for the index space is deleted and redefined as part of the RECOVER utility processing.
RECOVER Utility (Index Spaces) panel

Image copy data set name
The name (which must be cataloged) of the image copy data set. Note that the existence of an entry in the catalog table does not mean that the data set still exists.

Parent panels
• “DB2 Utilities panel” on page 515

Child panels
None.

Related tasks
• “RECOVER (index spaces)” on page 297

RECOVER Utility (Index Spaces) with LISTDEF panel

You use the RECOVER Utility (Index Spaces) with LISTDEF panel to generate utility control statements using the RECOVER INDEXSPACE form of the RECOVER utility.

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>RECOVER Utility (Index Spaces)</td>
<td>Row 1 to 1 of 1</td>
<td></td>
</tr>
</tbody>
</table>

Cmd C Name Reuse
LDEF1 N

******************************* Bottom of data ********************************

Command ===> Scroll PAGE
F1=Help F12=Cancel
F2=Split F3=Exit F7=Backward F8=Forward F9=Swap

Cmd
You can enter commands in this field to manipulate the panel rows.

I(n) Insert (n) new row(s)
R(n) Repeat the current row (n times)
D(n) Delete (n) row(s)
S(n) Select (n) rows(s)
RECOVER Utility (Index Spaces) with LISTDEF panel

An asterisk (*) in the PRC column shows that the LISTDEF in the row is selected. The S command toggles the selection of rows; unselected rows become selected and selected rows become unselected.

**LISTDEF Name**

On entry to the panel, this field is set to the value in the LISTDEF name field in the DB2 Utilities panel.

**Parent panels**

- “DB2 Utilities panel” on page 515

**Child panels**

None.

**Related tasks**

- “RECOVER utility” on page 294
- “RECOVER (index spaces)” on page 297

---

RECOVER Utility (Table Spaces) panel

You use the RECOVER Utility (Table Spaces) panel to generate utility control statements using the RECOVER TABLESPACE form of the RECOVER utility.

**Panel and field definitions**

<table>
<thead>
<tr>
<th>Process Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>RECOVER Utility (Table Spaces)</td>
<td>Row 1 to 12 of 131</td>
<td></td>
</tr>
</tbody>
</table>

The fields in the ISPF table part of the panel are:

**Sel** Selection column. Type S to select a table space.

**PRC** Processing column. An asterisk (*) indicates the selected table spaces.

**Database**

The name of the database in which the table space is defined.

**Table Space**

The name of the table space.
RECOVER Utility (Table Spaces) panel

**DS Num**
The partition number for this table space. Non-partitioned table spaces have a part number of 0. For a partitioned table space, the number indicates the partition of the table space.

**ICDATE**
The date that the image copy was taken. The format is *yymmdd*.

**ICTIME**
The time that the image copy was taken. The format is *hhmmss*.

**Typ**
The type of entry found in the SYSIBM.SYSCOPY table:
- F Full image copy.
- I Incremental image copy.

**Reuse**
Reuse option:
- Y The data set for the table space is not deleted and re-defined as part of the RECOVER utility processing.
- N The data set for the table space is deleted and redefined as part of the RECOVER utility processing.

**Re-bld Idx**
Rebuild indexes after recovery:
- Y A REBUILD INDEX (ALL) TABLE SPACE command is included in the JCL to perform a full rebuild of all of the indexes on all of the tables in the table space. This is the standard method for recovering a table space, and prevents the indexes being left in an unusable rebuild pending (RBDP) state.
- N Indexes not rebuilt.

**Image copy data set name**
The name (which must be cataloged) of the image copy data set. Note that the existence of an entry in the catalog table does not mean that the data set (still) exists.

**Parent panels**
- “DB2 Utilities panel” on page 515

**Child panels**
None.

**Related tasks**
- “RECOVER (table spaces)” on page 295

---

**RECOVER Utility (Table Spaces) with LISTDEF panel**

You use the RECOVER Utility (Table Spaces) with LISTDEF panel to generate utility control statements using the RECOVER INDEXSPACE form of the RECOVER utility.
Panel and field definitions

You can enter commands in this field to manipulate the panel rows.

- **I(n)**: Insert (n) new row(s)
- **R(n)**: Repeat the current row (n times)
- **D(n)**: Delete (n) row(s)
- **S(n)**: Select (n) rows(s)

An asterisk (*) in the PRC column shows that the LISTDEF in the row is selected. The S command toggles the selection of rows; unselected rows become selected and selected rows become unselected.

**LISTDEF Name**

On entry to the panel, this field is set to the value in the LISTDEF name field in the DB2 Utilities panel.

**Parent panels**

- ["DB2 Utilities panel" on page 515](#)

**Child panels**

None.

**Related tasks**

- ["RECOVER utility" on page 294](#)
- ["RECOVER (table spaces)" on page 295](#)

**Recovery Information panel**

FM/DB2 displays the Recovery Information panel if you use the ICS line command against an item listed on:

- The ["Databases panel" on page 500](#), or
- The ["Table Spaces panel" on page 698](#), or
Related tables panel

You use the Related tables panel to display information about the tables related to the table being edited when you issued the REDIT command.

The panel displays a selection list showing all the DB2 objects that are directly related to the object being edited by an RI constraint. This includes tables that are parent tables in a relationship with the table being edited, and tables that are dependent tables in a relationship with the table being edited.

Panel and field definitions

The entries shown on the Related tables panel depend on the cursor position when you issued the REDIT command:

- If the cursor was not on the data for a column that is part of a foreign, parent or primary key, all related tables are shown.
- If the cursor was on the data for a column that is part of a foreign, parent or primary key, only the related tables that include that column in a foreign, parent or primary key are shown.

The columns that are displayed include the SEL field, and columns of SYSIBM.SYSRELS.

**SEL** Selection column. You can issue any of these line commands against the DB2 objects in the list:

- **A** Alter table.
- **ALS** Show aliases.
- **B** Browse table. (See Note 1)
- **C** Copy table.
- **CDI** Show column distribution statistics.
- **CH** Show child (dependent) tables.
- **CHR** Show child (dependent) relations.
Related tables panel

CK  Show table check constraints.
COL  Show columns.
CON  Show table (unique and primary) constraints.
D  Show database.
E  Edit table. (See Note 1)
EE  Edit table - new session. (See Note 3)
FK  Show foreign keys.
I  Details about table or view.
KEY  Show relationship keys. (See Note 2)
P  Show privileges.
PA  Show parent tables.
PAR  Show parent relations.
PK  Show primary key.
PKG  Show packages.
PL  Show plans.
PR  Print table.
PST  Show partition statistics.
ROW  Show all columns for this row.
S  Show table spaces.
SYN  Show synonyms.
V  View table. (See Note 1)
VE  View table - new session. (See Note 3)
VW  Show views.
X  Show indexes.
XC  Show indexes and index columns.

Note:
1. The E, B, and V commands edit, browse, or view the object by stacking a new FM/DB2 editor session on top of the existing FM/DB2 editor session. You cannot return to the original FM/DB2 editor session without first ending the new FM/DB2 editor session.
2. For a parent relationship type, the KEY command shows the primary keys for the two tables in the relationship.
   For a dependent relationship type, the KEY command shows the foreign key columns for the two tables in the relationship.
3. The EE and VE commands edit (or view) the object by starting a new ISPF logical session (split screen) and then starting an FM/DB2 editor session in the new logical session.
   • When the REDIT command was issued and the cursor was not located on the data for a column that is part of a parent, primary or foreign key, the new FM/DB2 editor session shows all rows of the table.
   • When the REDIT command was issued with the cursor located on the data for a column that is part of a parent, primary or foreign key, the new FM/DB2 editor session only shows those rows related to the row where the cursor was located.
   The FM/DB2 editor session of the related table may have no rows when:
   • For a dependent table, there are no rows in the table with a matching foreign key value.
   • For a parent table, the foreign key value contains one or more columns with the DB2 null value.
Related tables panel

**Parent panels**
- "Table Browse panel" on page 689
- "Table Edit panel" on page 691
- "Table View panel" on page 700

**Child panels**
Various. Depends on which line command entered.

**Related tasks**
- "Specifying a DB2 object name" on page 25
- "Listing related tables" on page 150
- "Displaying or printing a list of objects" on page 242
- "DB2 Location Selection panel" on page 509

---

**Remote DB2 Location Selection panel**

You use the Remote DB2 Location Selection panel to display all the available remote locations. You can only access remote locations that have been defined to your local DB2 subsystem.

**Panel and field definitions**

```
<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>Remote DB2 Location Selection</td>
<td>Row 1 to 4 of 4</td>
<td></td>
</tr>
</tbody>
</table>

Sel Location
* QXPMVS8DB2D
  QXPMVS8DB26
  QXPMVS8DB2D
  QXPMVS8DB62

**************************************** END OF DB2 DATA ****************************************
```

- **Sel** Selection column. Type $ to select a location.
- **Location** The name of the location.

**Parent panels**
An asterisk (*) or pattern entered in the **Location** field on any Object List utility panel.
Remote DB2 Location Selection panel

Child panels

None.

Related tasks

- “Specifying a DB2 object name” on page 25
- “Displaying or printing a list of objects” on page 242
- “DB2 Location Selection panel” on page 509

REORG Utility (Indexes) panel

You use the REORG Utility (Indexes) panel to generate utility control statements using the REORG INDEX form of the REORG utility.

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

| Sel  | Selection column. Type S to select an index. |
| PRC  | Processing column. An asterisk (*) indicates the selected index. |

Index Creator

The name of the creator of the index.

Index Name

The name of the index.

Part Num

The partition number for the index. For a non-partitioned index, this is 0. Partitioned indexes appear as multiple rows, one row for each partition of the index. This is a display-only field.

Uni

The UNIQUERULE flag for the index, from the SYSINDEXES catalog table.

Clg

The CLUSTERING flag for the index, from the SYSINDEXES catalog table.

Cld

The CLUSTERED flag for the index, from the SYSINDEXES catalog table.
**REORG Utility (Indexes) panel**

**Cls Rat**
The CLUSTERRATIO flag for the index, from the SYSINDEXES catalog table.

**Leaf Dist**
The LEAFDIST flag for the index, from the SYSINDEXPART catalog table.

**Off Pos**
This is a display-only field. This value indicates the number of rows relocated out of their optimal position for the index. If the CARD value for the index is greater than zero, the value is calculated using the expression:

\[(AVG(NEAROFFPOS + FAROFFPOS) * 100)/CARD\]

This value helps you identify indexes that might require organization. For more information, see the *DB2 Utility Guide and Reference*.

**Reuse Data set**
Whether DB2 physically deletes and redefines the data sets for the index:

- **Y** Data set for the index reused.
- **N** Data set for the index not reused. This is the default setting.

**Parent panels**
- [“DB2 Utilities panel” on page 515](#)

**Child panels**
None.

**Related tasks**
- [“REORG (indexes)” on page 300](#)

---

**REORG Utility (Indexes) with LISTDEF panel**

You use the REORG Utility (Indexes) with LISTDEF panel to generate utility control statements using the REORG INDEX form of the REORG utility.
REORG Utility (Indexes) with LISTDEF panel

Panel and field definitions

<table>
<thead>
<tr>
<th>Process Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>REORG Utility (Indexes)</td>
<td>Row 1 to 2 of 2</td>
</tr>
</tbody>
</table>

- **P** Reuse
- **R** LISTDEF Data
- **Cmd** C Name Set
  - FRED N
  - JAMES N

An asterisk (*) in the PRC column shows that the LISTDEF in the row is selected. The S command toggles the selection of rows; unselected rows become selected and selected rows become unselected.

**LISTDEF Name**

On entry to the panel, this field is set to the value in the LISTDEF name field in the DB2 Utilities panel.

**Parent panels**

- "DB2 Utilities panel" on page 515

**Child panels**

None.

**Related tasks**

- "REORG utility" on page 298
- "REORG (indexes)" on page 300

**REORG Utility (Table Spaces) panel**

You use the REORG Utility (Table Spaces) panel to generate utility control statements using the REORG TABLESPACE form of the REORG utility.
Panel and field definitions

Sel  Selection column. Type S to select a table space.

PRC  Processing column. An asterisk (*) indicates the selected table space.

Database Name
   The name of the database in which the table space is defined.

Table Space Name
   The name of the table space.

Prt Num
   The partition number for the table space. This is 0 for a non-partitioned
   table space. Partitioned table spaces appear as multiple rows, one row for
   each partition of the table space. This is a display-only field.

Avg Ind Ref
   This is a display-only field. This is an average value for the number of
   rows relocated out of their original position. If the table space/partition's
   cardinality is greater than zero, the calculated value is:

   \[
   \frac{(\text{AVG(NEARINDREF} + \text{FARINDREF}) \times 100)}{\text{CARD}}
   \]

   This value helps you identify table spaces that might require organization.
   For more information about these values, see the DB2 Utility Guide and
   Reference.

Avg Off Pos
   This is a display-only field. This is an average value for the number of
   rows relocated out of their optimal position for all indexes within the table
   space or partition. The average is for all indexes which have a CARD value
   greater than zero, using the expression:

   \[
   \frac{(\text{AVG(NEAROFFPOS} + \text{FAROFFPOS}) \times 100)}{\text{CARD}}
   \]

   This value helps you identify table spaces that might require organization.
   For more information, see the DB2 Utility Guide and Reference.
REORG Utility (Table Spaces) panel

Unload External (Y/N)
Whether the REORG process leaves the data in external format and therefore acts as an unload utility:

Y  REORG processing halts at the unload phase and leaves the data in external format in the SYSREC00 data set. Load control cards are produced in the SYSPUNCH data set. The data produced is in a format suitable for loading into any similar table structure, in any DB2 subsystem. The DD names in the job and resultant SYSPUNCH are compatible with the DSNTIAUL sysrec naming (for example, SYSREC00), therefore the FM/DB2 LOAD Utility can accept both formats, without requiring any JCL editing before submission.

N  Full REORG performed.

Log (Y/N)
LOG option:

Y  Changes written to the DB2 log.

N  Changes not written to the DB2 log. This is the default setting.

For more information, see the DB2 Utility Guide and Reference.

Reuse Data Set
Reuse option:

Y  The data set for the table space is not deleted and re-defined as part of the RECOVER utility processing.

N  The data set for the table space is deleted and redefined as part of the RECOVER utility processing.

Parent panels
• “DB2 Utilities panel” on page 515

Child panels
None.

Related tasks
• “REORG (table spaces)” on page 298

REORG Utility (Table Spaces) with LISTDEF panel

You use the REORG Utility (Table Spaces) with LISTDEF panel to generate utility control statements using the REORG TABLESPACE form of the REORG utility.
Panel and field definitions

Cmd  You can enter commands in this field to manipulate the panel rows.
  I(n)  Insert (n) new row(s)
  R(n)  Repeat the current row (n times)
  D(n)  Delete (n) row(s)
  S(n)  Select (n) rows(s)

An asterisk (*) in the PRC column shows that the LISTDEF in the row is
selected. The S command toggles the selection of rows; unselected rows
become selected and selected rows become unselected.

LISTDEF Name
On entry to the panel, this field is set to the value in the LISTDEF name
field in the DB2 Utilities panel.

Parent panels

• “DB2 Utilities panel” on page 515

Child panels

None.

Related tasks

• “REORG utility” on page 298
• “REORG (table spaces)” on page 298

Revoke privileges panels

These panels are:
• Revoke Application Plan Privileges panel
• Revoke Buffer Pool Privileges panel
• Revoke Collection Privileges panel
Revoke privileges panels

- Revoke Column Privileges panel
- Revoke Database Privileges panel
- Revoke Distinct Type Privileges panel
- Revoke Function Privileges panel
- Revoke Package Privileges panel
- Revoke Schema Privileges panel
- Revoke Sequence Privileges panel
- Revoke Stored Procedure Privileges panel
- Revoke System Privileges panel
- Revoke Table Privileges panel
- Revoke Table Space Use Privileges panel

FM/DB2 displays one of these panels if you use the Privileges utility function (3.5) to revoke privileges for an object type, or if you specify the R line command against an object displayed in a list of DB2 objects.

Format of revoke privilege panels

The revoke privileges panels are similar to the corresponding grant privilege panels. However, in place of the To field, the revoke privileges panels contain the following fields:

- A From field. Use this field to specify the SQLID of the user for whom the privileges are to be revoked (the revokee).
- A By field. Use this field to specify the SQLID of the user who granted the privileges (the grantor), or “ALL” to indicate all grantors.

Related tasks

Routine Parameters panel

FM/DB2 displays the Routine Parameters panel if you use the PARM line command against an item listed on:

- The “Functions panel” on page 584, or
- The “Stored Procedures panel” on page 685

Row Selection Criteria panel

You use the Row Selection Criteria panel to specify row selection criteria by column.
Panel and field definitions

**Cmd**  Prefix command column.

You can use the following prefix commands when editing row selection criteria by column:

- **A** Identifies the line after which data is to be moved or copied.
- **B** Identifies the line before which data is to be moved or copied.
- **C** Copy one line.
- **Cn** Copy n rows.
- **CC** Copy block of lines. Mark start and end of block.
- **D** Delete one line.
- **Dn** Delete n lines.
- **DD** Delete block of lines. Mark start and end of block.
- **M** Move one line.
- **Mn** Move n lines.
- **MM** Move block of lines. Mark start and end of block.
- **R** Repeat one line.
- **Rn** Repeat n lines.
- **RR** Repeat block of lines. Mark start and end of the block.
- **RRn** Repeat block of lines n times. Mark start and end of block.

**Con**  Connector column.

You specify the AND and OR connectors in this column. The default value is AND. You can use abbreviations. For example, if you overtype the existing connector, AND, with 0, FM/DB2 generates an OR connector.
### Row Selection Criteria panel

Connectors are only relevant to lines containing expressions (that is, where you have specified an operator in the **Op** column).

You can overtype the existing connector with one of the following values:

- **AND or &**
  - FM/DB2 generates an AND connector in the resultant WHERE clause.

- **OR or |**
  - FM/DB2 generates an OR connector in the resultant WHERE clause.

- **(**
  - Left parenthesis column.

If you need to define the sequence of evaluation for the resultant WHERE clause by enclosing part of the clause in parentheses, use this column to type a left parenthesis ("(") at the appropriate point.

You can only use a parenthesis on a line containing expressions (that is, where you have specified an operator in the **Op** column). The number of left parentheses must match the number of right parentheses.

**Column name**

The name of the column.

On 80-byte screens, FM/DB2 displays either the column name details (the default) or the data type details for the column. To toggle between FM/DB2 displaying the Column name and Data type(length) details, press the PF11 function key.

**Data type(length)**

The DB2 data type and, where applicable, length of the column.

On 80-byte screens, FM/DB2 displays either the column name details (the default) or the data type details for the column. To toggle between FM/DB2 displaying the Column name and Data type(length) details, press the PF11 function key.

**Op**

Operator column. Operators are used to create expressions. When you type an operator in the Op column, it indicates that you want FM/DB2 to form an expression using the column name on that line of the panel, the operator you have specified and, with the exception of the operators NL, ¬NL, and NNL, the contents of the Value column.

FM/DB2 supports the following operators:

- **= or EQ**
  - True if the terms are equal (numerically, or when padded).

- **¬= or NE**
  - True if the terms are not equal (inverse of =)

- **> or GT**
  - Greater than

- **< or LT**
  - Less than

- **>= or GE**
  - Greater than or equal to

- **\= or \<**
  - Not less than
Row Selection Criteria panel

<= or LE
    Less than or equal
\> or =>
    Not greater than
LK    LIKE predicate. Column must not be numeric.
~LK or NLK
    NOT LIKE. Column must not be numeric.
BTW    BETWEEN. Value must be in the form value1,value2.
~BT or NBT
    NOT BETWEEN. Value must be in the form value1,value2.
IN    IN. Value must be in the form value1,value2,....
~IN or NIN
    NOT IN. Value must be in the form value1,value2,....
NL    IS NULL. Value must be blank.
~NL or NNL
    IS NOT NULL. Value must be blank.
RG    Range. Value must be in the form value1,value2.
NR    Not in range. Value must be in the form value1,value2.

Value
    The character or numeric constant to be used with the specified operator and column name to create an expression in the resultant WHERE clause.

When the operator is NL, ~NL, or NNL, the Value column must be blank.
For all other operators, the Value column must not be blank.

For operators that support more than one value (BTW, ~BT, NBT, IN, ~IN, NIN, RG, or NR), the values must be separated by commas. For operators IN, ~IN, or NIN, and the column is alphanumeric, you must enclose each value in quotes.

If the data type of the column is alphanumeric, the value can be:
- A character string not starting or ending with a single quotation mark.
  and not containing any embedded quotes or commas. The case of the string is ignored.
- A character string enclosed in single quotation marks. The string can contain blanks and commas, but not any embedded quotes. The case of the string is ignored.

If the data type of the column is numeric, the value must be numeric.

For DATE, TIME, and TIMESTAMP columns, if you specify a “=” in the Value field, FM/DB2 displays the current date, time, or timestamp values respectively in the required format.

)    Right parenthesis column.

If you need to define the sequence of evaluation for the resultant WHERE clause by enclosing part of the clause in parentheses, use this column to type a right parenthesis (”) at the appropriate point.

You can only use parentheses on lines containing expressions (that is, where you have specified an operator in the Op column). The number of left parentheses must match the number of right parentheses.
## RUNSTATS Utility (Indexes) panel

You use the RUNSTATS Utility (Indexes) panel to generate utility control statements using the RUNSTATS INDEX form of the RUNSTATS utility.

### Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>RUNSTATS Utility (Indexes)</td>
<td>Row 1 to 12 of 341</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Sel</th>
<th>PRC</th>
<th>Index Creator</th>
<th>Index Name</th>
<th>Part'n Number</th>
<th>Report Stats (Y/N)</th>
<th>Option</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Y</td>
</tr>
</tbody>
</table>

### Sel
Selection column. Type S to select an index.

### PRC
Processing column. An asterisk (*) indicates the selected index.

### Index Creator
The name of the creator of the index.

### Index Name
The name of the index.

### Part'n Number
The partition number for the index. A zero indicates a non-partitioned index. A non-zero value indicates the partition number of a partitioned index.

### Report Stats (Y/N)
Whether a report outlining the results of the utility is written to the SYSPRINT DDNAME:

| Y | Report written to the SYSPRINT DDNAME. |
RUNSTATS Utility (Indexes) panel

N Report not written to the SYSPRINT DDNAME. This is the default setting.

For more information, see the DB2 Utility Guide and Reference.

Update Option (A/P/S/N)
Indicates which catalog tables (if any) are updated by the RUNSTATS utility. The available options are:
A All. All catalog tables are updated. This is the default setting.
P Access path only. Statistics related to access path selection are updated.
S Space only. Only statistics related to space information are updated.
N None. No catalog tables are updated.

Parent panels
- “DB2 Utilities panel” on page 515

Child panels
None.

Related tasks
- “RUNSTATS (indexes)” on page 304

RUNSTATS Utility (Indexes) with LISTDEF panel
You use the RUNSTATS Utility (Indexes) with LISTDEF panel to generate utility control statements using the RUNSTAT INDEX form of the RUNSTAT utility.

Panel and field definitions

<table>
<thead>
<tr>
<th>Process Options Utilities Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2) RUNSTATS Utility (Indexes) Row 1 to 2 of 2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>P Report Update</th>
</tr>
</thead>
<tbody>
<tr>
<td>R LISTDEF Stats Option</td>
</tr>
<tr>
<td>C Name (Y/N) (A/P/S/N)</td>
</tr>
</tbody>
</table>

* FRED N A
** JAMES N A

******************************* Bottom of data ******************************************

Cmd You can enter commands in this field to manipulate the panel rows.
RUNSTATS Utility (Indexes) with LISTDEF panel

I(n) Insert (n) new row(s)
R(n) Repeat the current row (n times)
D(n) Delete (n) row(s)
S(n) Select (n) rows(s)

An asterisk (*) in the PRC column shows that the LISTDEF in the row is selected. The S command toggles the selection of rows; unselected rows become selected and selected rows become unselected.

LISTDEF Name
On entry to the panel, this field is set to the value in the LISTDEF name field in the DB2 Utilities panel.

Parent panels
- "DB2 Utilities panel" on page 515

Child panels
None.

Related tasks
- "RUNSTATS utility" on page 302
- "RUNSTATS (indexes)" on page 304

RUNSTATS Utility (Table Spaces) panel

You use the RUNSTATS Utility (Table Spaces) panel to generate either RUNSTATS TABLESPACE or RUNSTATS INDEX control statements, depending on the value selected for the Indexes Only column.

Panel and field definitions

<table>
<thead>
<tr>
<th>Process Options Utilities Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2) RUNSTATS Utility (Table Spaces) Row 1 to 12 of 237</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>S P e R Database Space Part'n Number Tables (Y/N) Indexes Only Stats Option</th>
</tr>
</thead>
<tbody>
<tr>
<td>L C Name Name Number (A/P/S/N) (Y/N) (Y/N) (A/P/S/N)</td>
</tr>
<tr>
<td>BILL SDB1 TS1D 0 1 Y N A</td>
</tr>
<tr>
<td>BILL SDB1 TS1E 1 1 Y N A</td>
</tr>
<tr>
<td>BILL SDB1 TS1E 2 1 Y N A</td>
</tr>
<tr>
<td>BILL SDB1 TS1E 3 1 Y N A</td>
</tr>
<tr>
<td>BILL SDB1 TS1E 4 1 Y N A</td>
</tr>
<tr>
<td>BILL SDB2 UTILTS1 0 2 Y N A</td>
</tr>
<tr>
<td>CJMA ODA CJMAOSD 0 1 Y N A</td>
</tr>
<tr>
<td>CJMA ODA CJMAOSE 1 1 Y N A</td>
</tr>
<tr>
<td>CJMA ODA CJMAOSE 2 1 Y N A</td>
</tr>
<tr>
<td>CJMA ODA CJMAOSE 3 1 Y N A</td>
</tr>
<tr>
<td>CJMA ODA CJMAOSE 4 1 Y N A</td>
</tr>
<tr>
<td>CJMA ODA CJMAOSP 0 4 Y N A</td>
</tr>
</tbody>
</table>

Command ===> Scroll PAGE
F1=Help F2=Split F3=Exit F7=Backward F8=Forward F9+Swap
F12=Cancel

Sel Selection column. Type S to select a table space.
RUNSTATS Utility (Table Spaces) panel

**PRC**  Processing column. An asterisk (*) indicates the selected table space.

**Database Name**
The name of the database in which the table space is defined.

**Table Space Name**
The name of the table space.

**Part'n Number**
The partition number for the table space. A zero value indicates a non-partitioned table space; a non-zero value indicates the partition number.

**Number Tables**
The number of tables in the table space.

**Indexes Only (Y/N)**
Whether RUNSTATS is performed for only the indexes in the table spaces (without gathering statistics for the tables in the table space itself):
- **Y** RUNSTATS performed for indexes only.
- **N** RUNSTATS performed for tables and indexes. This is the default setting.

**Report Stats (Y/N)**
Whether a report outlining the results of the utility is written to the SYSPRINT DDNAME:
- **Y** Report written to the SYSPRINT DDNAME.
- **N** Report not written to the SYSPRINT DDNAME. This is the default setting.

For more information, see the DB2 Utility Guide and Reference.

**Update Option (A/P/S/N)**
Indicates which catalog tables (if any) are updated by the RUNSTATS utility:
- **A** All. All catalog tables are updated. This is the default setting.
- **P** Access path only. Statistics related to access path selection are updated.
- **S** Space only. Only statistics related to space information are updated.
- **N** None. No catalog tables are updated.

**Parent panels**
- "DB2 Utilities panel" on page 515

**Child panels**
None.

**Related tasks**
- "RUNSTATS (table spaces)" on page 303
You use the RUNSTATS Utility (Table Spaces) with LISTDEF panel to generate utility control statements using the RUNSTAT TABLESPACE form of the RUNSTAT utility.

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>RUNSTATS Utility (Table Spaces)</td>
<td>Row 1 to 2 of 2</td>
<td></td>
</tr>
</tbody>
</table>

P Report Update
R LISTDEF Stats Option
Cmd C Name (Y/N) (A/P/S/N)
  FRED N A
  * GEORGE N A

An asterisk (*) in the PRC column shows that the LISTDEF in the row is selected. The S command toggles the selection of rows; unselected rows become selected and selected rows become unselected.

LISTDEF Name
On entry to the panel, this field is set to the value in the LISTDEF name field in the DB2 Utilities panel.

Parent panels
- “DB2 Utilities panel” on page 515

Child panels
None.

Related tasks
- “RUNSTATS utility” on page 302
- “RUNSTATS (table spaces)” on page 303
Schemas panel

You use the Schemas panel to list information about the schemas you have selected.

Panel and field definitions

The columns that are displayed include the SEL field and columns of various catalog tables.

Note:
1. The system option, Show all catalog table columns, affects which columns FM/DB2 displays when you use the Object List utility.
2. You can also customize which columns FM/DB2 displays when you use the Object List utility. For details, see the File Manager Customization Guide.

To view other columns on this panel, press the Right function key (F11).

For a complete list of columns, refer to “DB2 catalog tables” in the DB2 for z/OS SQL Reference relevant to your version of DB2.

SEL Line command area.

For a list of the line commands you can use on this panel, see Table 22 on page 250.

Parent panels

- “Object List Utility panel” on page 610
- “Distinct Types panel” on page 523
- “Functions panel” on page 584
- “Stored Procedures panel” on page 685
- “Triggers panel” on page 712
Scramble Exit Specification panel

The Scramble Exit Specification panel allows you to provide the name of a scramble exit, a constant to be passed to the exit, and to specify whether numeric fields are to be formatted before the value is passed to the exit.

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>Scramble Exit Specification</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Fixed Attributes:
- Column name: COLCARD
- Type: INTEGER

Scramble Exit:
- Program name
- Optional constant +

Scramble Exit Processing Options:
- Format Option
  1. None
  2. Format
  3. Leading Zeros

Figure 117. Scramble Exit Specification panel

Program name
A 1 to 8 character valid load module member name that identifies the
Scramble Exit Specification panel

scramble exit invoked during a copy operation for this output column. The exit must be in the form of a load module, in any load library available to FM/DB2 at the time of the copy process, either by a STEPLIB DD statement, or in LINKLIST, or LPALIST.

Optional constant
A constant value that is passed to the exit for each call type. To enter a hexadecimal value, press F4 or enter the EXPAND command with the cursor positioned in this field and use the command HEX ON from the expand window.

Format Option
This option is only displayed for numeric columns and allows you to request FM/DB2 to pass the value specified in Optional constant to the exit in one of the following forms:

1. None
   Field value passed to the exit in the internal format.

2. Format
   Field value passed to the exit as displayed numeric with leading zeros suppressed.

3. Leading Zeros
   Field value passed to the exit as displayed numeric with leading zeros.

Parent panels

- “Column Attributes panel (alphanumeric)” on page 397
- “Column Attributes panel (numeric)” on page 402

Related tasks

- “Scrambling data” on page 232

Select Statement Browse panel

FM/DB2 displays the Select Statement Browse panel if the Use edit (instead of browse) to display results option is not selected, and you press Enter when one of these panels is displayed:

- “Basic SELECT Prototyping panel” on page 391
- “Advanced SELECT Prototyping panel” on page 368

Related tasks

- “FM/DB2 Systems Options (1 of 4) panel” on page 577

Select Statement Edit panel

FM/DB2 displays the Select Statement Edit panel if the Use edit (instead of browse) to display results option is selected, and you press Enter when one of these panels is displayed:

- “Basic SELECT Prototyping panel” on page 391
- “Advanced SELECT Prototyping panel” on page 368

Related tasks

- “FM/DB2 Systems Options (1 of 4) panel” on page 577
Sequences panel

You use the Sequences panel to display information about the sequences you have selected.

Panel and field definitions

The columns that are displayed include the SEL field and columns of SYSIBM.SYSSEQUENCES.

To view other columns on this panel, press the Right function key (F11).

For a complete list of columns, refer to “DB2 catalog tables” in the DB2 for z/OS SQL Reference relevant to your version of DB2.

<table>
<thead>
<tr>
<th>SEL</th>
<th>SCHEMA</th>
<th>OWNER</th>
<th>NAME</th>
<th>SEQTYPEX</th>
<th>SEQUENCEID</th>
<th>CREATEDBY</th>
</tr>
</thead>
<tbody>
<tr>
<td>###</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
</tr>
</tbody>
</table>

**** Top of data ****

KEISTEW KEISTEW SEQUENCE1 Sequence object 256 KEISTEW

**** End of data ****

Figure 118. Sequences panel

To view other columns on this panel, press the Right function key (F11).

For a complete list of columns, refer to “DB2 catalog tables” in the DB2 for z/OS SQL Reference relevant to your version of DB2.

**SEL**  Line command area.

For a list of the line commands you can use on this panel, see Table 22 on page 250.

Parent panels

• “Object List Utility panel” on page 610

Child panels

<table>
<thead>
<tr>
<th>To display this panel...</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create Sequence panel</td>
<td>Line command CR</td>
</tr>
<tr>
<td>Sequence Dependencies panel</td>
<td>Line command DEP</td>
</tr>
<tr>
<td>Drop Sequence panel</td>
<td>Line command DR</td>
</tr>
<tr>
<td>Grant Sequence Privileges panel</td>
<td>Line command G</td>
</tr>
</tbody>
</table>

“Display Row panel” on page 523  Line command ROW
Set Batch Job Card Information panel

You use the Set Batch Job Card Information panel to specify a JCL jobcard that FM/DB2 is to use whenever it generates a batch job. You can also specify a JOBLIB or other statements.

These options are saved in your ISPF profile and persist from one FM/DB2 session to the next.

Panel and field definitions

Batch Submission Job Statement Information:

The JCL job card information that is used for any batch job that is created.

Parent panels

Child panels

None.

Related tasks

- “Job card specifications (option 0.3)” on page 42
Set COBOL Processing Options panel

You use the Set COBOL Processing Options panel to customize how FM/DB2 compiles COBOL copybooks to generate FM/DB2 templates.

These options are saved in your ISPF profile and persist from one FM/DB2 session to the next.

Panel and field definitions

Additional SYSLIB data sets

When you use one of the preceding FM/DB2 functions, if the COBOL copybook that you specify uses COPY compiler-directing statements to include other members that do not belong to the same PDSs as the copybooks, then you need to specify those other PDSs here.

Here's why: members included by COPY compiler-directing statements must be found in the SYSLIB concatenation. When you specify a copybook on an FM/DB2 panel, the SYSLIB concatenation consists of the PDS of the copybook, plus up to ten additional PDSs that you can specify on this panel. These PDSs are searched in order (PDS of the copybook, followed by these additional PDSs, 1–10.)

COBOL REPLACE options

“From” and “To” pseudo-text character strings for the REPLACE compiler-directing statement.
Set COBOL Processing Options panel

If the COBOL copybook contains characters that you want to replace with other characters before compiling the copybook into a template, then use these replacing options.

For example, if the copybook contains colon characters (:) that you want to remove before compiling, then specify `**:**` as a “From” string and `******` as the matching “To” string.

For details on specifying “From” and “To” strings for REPLACE, see the [IBM COBOL Language Reference](#).

**COBOL Compiler Options**

Compiler options to be used when compiling a COBOL copybook into a template:

- **DBCS** Instructs File Manager to use the DBCS compiler option.

- **Decimal-point is comma**
  Uses the "Decimal-point is comma" SPECIAL-NAMES paragraph when compiling COBOL copybooks.

- **Arith(extend)**
  Uses the Arith(extend) COBOL compile option when compiling COBOL copybooks.

- **Mixed case field names**
  Retains the original case of the field name as coded in the COBOL copybook.

- **Maximum Return Code**
  Sets the maximum warning/error code accepted from the compiler, when compiling a copybook. Any code higher than specified causes FM/DB2 to display a pop-up panel, in which you can review the compilation and decide about future processing.

**Additional options**

Additional COBOL compiler options which are added via the CBL statement when a COBOL compile is run to build or update a template. These options are validated during the compile process. To avoid compile errors please ensure the syntax is correct and that any additional data sets required by these options are allocated prior to invoking File Manager.

**Parent panels**

- 

**Child panels**

None.

**Related tasks**

- "COBOL compiler specifications (option 0.5)" on page 42

Set HLASM Processing Options panel

The HLASM Processing Options panel controls various options used by the HLASM compiler to generate templates from your HLASM copybooks.
**Panel and field definitions**

```
<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>File Manager</td>
<td>Set HLASM Processing Options</td>
<td></td>
</tr>
</tbody>
</table>

**Additional SYSLIB data sets:**

1. _____________________________________________
2. _____________________________________________
3. _____________________________________________
4. _____________________________________________
5. _____________________________________________
6. _____________________________________________
7. _____________________________________________
8. _____________________________________________
9. _____________________________________________
10. _____________________________________________

**HLASM Compiler Options:**

Enter "/" to select option
- DBCS
- NOALIGN

**Additional options**

Additional HLASM compiler options which are added via the *PROCESS statement when the assembler is run to build or update a template. These options are validated during the assembly process. To avoid assembly errors please ensure the syntax is correct.

**Parent panels**
- “Set System Processing Options panel” on page 673
- “Set Processing Options panel” on page 671

**Child panels**
- None.

**Comments**
- “HLASM compiler specifications (option 0.6)” on page 42
Set Optimizer Hint panel

Panel and field definitions

Insert a hint for the optimizer into FMUSER.PLAN_TABLE.

Columns Considered by the Optimizer:

- Query number
- Query block number
- Application plan
- DBRM/package
- Collection
- Version
- Access method
- Access type
- Table owner
- Table name
- Correlation name
- Table number
- Index owner
- Index name
- Sort new table
- Sort composite
- Prefetch
- Parallelism
- Access degree
- Hint ID
- Comment

Command ===>
F1=Help  F2=Split  F3=Exit  F4=Retrieve  F7=Backward  F8=Forward
F9=Swap  F10=Actions  F12=Cancel

Parent panels

- “Plan Table Rows panel” on page 615

Child panels

None.

Related tasks
Set Output Data Set Allocation Options panel

Panel and field definitions

Space Units
Defines the unit of primary and secondary space to be allocated.
- **BLK**: Block of average size.
- **KB**: Kilobyte, a kilobyte is 1024 bytes.
- **MB**: Megabyte, a megabyte is 1048576 bytes.
- **TRK**: Track of a direct access storage device (DASD).
- **CYL**: Cylinder of a DASD.

Primary Quantity
Amount of DASD space to be used for primary space allocation. The range depends on the space unit specified and the DASD device type.

Secondary Quantity
Amount of DASD space to be used for secondary space allocation. The range depends on the space unit specified and the DASD device type.

Management Class
Name of a management class defined in your Storage Management System (SMS) installation. Do not enter a value on a system without SMS.

Storage Class
Name of a storage class defined in your Storage Management System (SMS) installation. Do not enter a value on a system without SMS.

Data Class
Name of a data class defined in your Storage Management System (SMS) installation. Do not enter a value on a system without SMS.

Device Type
Specifies the device type of the volume to contain the data set. Enter an IBM device type or a generic device name that is supported by your system.
Set PL/I Processing Options panel

You use the Set PL/I Processing Options panel to customize how FM/DB2 compiles PL/I copybooks to generate FM/DB2 templates.

These options are saved in your ISPF profile and persist from one FM/DB2 session to the next.

Panel and field definitions

<table>
<thead>
<tr>
<th>Process Options Utilities Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2) Set PL/I Processing Options</td>
</tr>
</tbody>
</table>

Additional SYSLIB data sets:
1.  
2.  
3.  
4.  
5.  
6.  
7.  
8.  
9.  
10.  

PL/I Compiler Options for Copybooks:
Enter /* to select option

GRAPHIC UNALIGNED

63 bit binary 31 digit decimal

Additional options

Maximum Return Code to be accepted from compiler 04 (0-99)

Command =>
F1=Help F2=Split F3=Exit F7=Backward F8=Forward F9=Swap
F12=Cancel

Additional SYSLIB data sets

Any additional SYSLIB data sets to be used in the compilation of PL/I copybooks.

PL/I Compiler Options

These options determine how FM/DB2 compiles PL/I copybooks to generate FM/DB2 templates.

Additional SYSLIB data sets

The names of data sets to be searched if your copybooks contain references to other PL/I copybooks.

GRAPHIC

Select this option if the copybook contains DBCS or mixed data.
Set PL/I Processing Options panel

**UNALIGNED**
Select this option if fields are not aligned on full-word boundaries.

**63 bit binary**
Select this option if you require support for 63 bit binary numbers. The PL/I compiler option LIMITS(FIXEDBIN(63)) is used when you select this option.

**31 digit decimal**
Select this option if you require support for 31 digit decimal numbers. The PL/I compiler option LIMITS(FIXEDDEC(31)) is used when you select this option.

**Additional options**
Additional PL/I compiler options which are added via the *PROCESS statement when a PL/I compile is run to build or update template. These options are validated during the compile process. To avoid compile errors please ensure the syntax is correct and that any additional data sets required by these options are allocated prior to invoking File Manager.

For details on the effects of using these options, see the *IBM VisualAge PL/I for OS/390 Programming Guide*

**Maximum Return Code**
Sets the maximum warning/error code accepted from the compiler, when compiling a copybook. Any code higher than specified causes FM/DB2 to display a pop-up panel, in which you can review the compilation and decide about future processing.

**Parent panels**

**Child panels**
None.

**Related tasks**
- *“Setting default processing options” on page 39*

Set Print Processing Options panel

You use the Set Print Processing Options panel to control processing of the current FM/DB2 session.

These options are saved in your ISPF profile and persist from one FM/DB2 session to the next.
**Set Print Processing Options panel**

**Panel and field definitions**

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFE2)</td>
<td>Set Print Processing Options</td>
<td></td>
</tr>
</tbody>
</table>

**Print Options:**

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Output destination</td>
<td>Record length</td>
</tr>
<tr>
<td>1</td>
<td>SYSPRINT</td>
</tr>
<tr>
<td>2</td>
<td>Terminal</td>
</tr>
<tr>
<td>3</td>
<td>Data set</td>
</tr>
<tr>
<td>4</td>
<td>REXX</td>
</tr>
</tbody>
</table>

**Data set DISP**

- | 1 | OLD |
- | 2 | MOD |

**Data set name**: FMUSER.FMN.LIST

**Output class**: A

**Lines per page**: 50

**Record limits**: (1,7) (n,m) n=begin column, m=end column

**Command**: __________________________________________________________________

- F1=Help
- F2=Split
- F3=Exit
- F4=Retrieve
- F6=Reset
- F7=Backward
- F8=Forward
- F9=Swap
- F10=Actions
- F12=Cancel

**Output destination**

Specifies where print output is sent, except for batch execution.

- **SYSPRINT**: Send print output to the current SYSPRINT allocation.
- **Terminal**: Send print output to the TSO terminal.
- **Data set**: Output is accumulated in the print data set specified in the **Data set name** field. This data set can be browsed using the PB command, or sent for printing to the JES spool queue with class specified in the **Output class** field using the PRINT command while browsing.
- **REXX**: Send output to a REXX stem variable. Each line corresponds to a variable named FILEM.mn. FILEM.0 contains the line count.

When you use the PRINT primary command in Print Browse (option 3.9), or if you specify batch execution in an FM/DB2 panel, then print output is always sent to a SYSOUT class.

**Record length**

Specifies the line length of print output.

- **80**: The line length of print output is 80 characters, suitable for a terminal.
- **132**: The line length of print output is 132 characters, suitable for a printer.

**Note**: Record length is not applicable when:

- The data is printed in TABL format, or
- You specify the wide format for the compare output listing

In either of the above cases, when data is printed the line length is determined from the number and size of fields printed. If this length is greater than the record length specified for the data set, the print line is truncated. The maximum print line length is 32760.
Set Print Processing Options panel

Data set DISP
Specifies the disposition of a print data set.

OLD    The print data set is cleared before each print operation, and print output is written from the beginning of the data set.

MOD    Default setting. Print output is appended to the existing print data set.

Page skip
If selected, print output from each function begins on a new page.

Wide print
If selected, for to-be-allocated (new) print output datasets (online) or for SYSPRINT allocation (in batch), FM/DB2 uses the maximum record length/blocksize of 32756/32760 and record format of VBA.

If not selected, for to-be-allocated (new) print output datasets (online) or for SYSPRINT allocation (in batch), FM/DB2 uses the record length/blocksize of 133/137 and record format of VBA.

Note: This setting only applies when the output dataset is newly-allocated during execution. It is ignored for pre-allocated print output datasets.

Translate non-printable chars
Specifies how FM/DB2 should translate non-printable characters.

If selected, non-printable characters are translated to blanks using a translate table.

If not selected, no translation is performed. Use PRTTRANS=OFF to support special print chains. This may make printing faster.

For instructions on altering the print translate table, see the File Manager Customization Guide.

Uppercase message text
All message text is translated to uppercase.

Data set name
If you specified the Output destination as 3 (Data set), then after you enter the PRINT command while executing the Print Browse (PB) function, FM/DB2 sends print output to this temporary data set.

The default is userid.FMN.LIST, unless changed in your FM/DB2 installation.

Output class
The class of the JES spool queue to be used when the PRINT command is issued while browsing the temporary print data set.

Lines per page
Specifies the number of lines printed on each page of print output. You can specify a value from 1 to 999. The default is 60.

Record limits
When data is printed in CHAR or LHEX print format, limits print output for each record (or OAM object).

(1,*) The entire record (or block) is printed.

(n,m) Print output is limited to the data from columns (bytes) n through m. An asterisk (*) specified for m indicates the end of the record.
Set Print Processing Options panel

Parent panels

- “Set System Processing Options panel” on page 673

Child panels

None.

Related tasks

- “Print settings (option 0.1)” on page 41
- “PRINT primary command” on page 768
- “Looking at the print output from your FM/DB2 session” on page 266

Set Processing Options panel

You use the Set Processing Options panel to navigate to other panels from which you can set various types of options that affect the behavior of FM/DB2 panels. These options are saved in your ISPF profile and persist from one FM/DB2 session to the next.

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>Set Processing Options</td>
</tr>
<tr>
<td>0 System</td>
<td>System Options</td>
</tr>
<tr>
<td>1 View</td>
<td>View options</td>
</tr>
<tr>
<td>2 Edit</td>
<td>Edit options</td>
</tr>
<tr>
<td>3 Utilities</td>
<td></td>
</tr>
<tr>
<td>3.1 Print</td>
<td>Print utility options</td>
</tr>
<tr>
<td>3.3 Copy</td>
<td>Copy utility options</td>
</tr>
<tr>
<td>3.4 Object List</td>
<td>Object list utility options</td>
</tr>
<tr>
<td>3.6 Import</td>
<td>Import utility options</td>
</tr>
<tr>
<td>3.7 Export</td>
<td>Export utility options</td>
</tr>
<tr>
<td>3.L Load</td>
<td>DB2 LOAD utility options</td>
</tr>
<tr>
<td>3.U ULISTDEF</td>
<td>DB2 Utility LISTDEF options</td>
</tr>
<tr>
<td>3.UO OPTIONS</td>
<td>DB2 Utility OPTIONS options</td>
</tr>
<tr>
<td>3.UT TEMPLATE</td>
<td>DB2 Utility TEMPLATE options</td>
</tr>
<tr>
<td>3.UU Unload</td>
<td>DB2 Unload utility options</td>
</tr>
</tbody>
</table>

Command ===>
F1=Help  F2=Split  F3=Exit  F7=Backward  F8=Forward  F9=Swap
F12=Cancel

0 System Options
Displays the Set System Processing Options panel

1 View
Displays the first Editor Options panel

2 Edit
Displays the first Editor Options panel

3 Utilities
Displays the Set Utilities Processing Options panel

3.1 Print
Displays the Print Utility Options panel
### Set Processing Options panel

3.3 Copy
Displays the Copy Options panel

3.4 Object List
Displays the FM/DB2 Object List Options (1 of 2) panel

3.6 Import
Displays the Import Options panel

3.7 Export
Displays the first Export Options panel

3.1 LOAD
Displays the Load Utility Options panel

3.UL LISTDEF
Displays the LISTDEF Options panel

3.UO OPTIONS
Displays the OPTIONS Options panel

3.UT TEMPLATE
Displays the TEMPLATE Options (1 of 2) panel

3.UU Unload
Displays the UNLOAD Options panel

#### Parent panels
- “Primary Option Menu panel” on page 617

#### Child panels

<table>
<thead>
<tr>
<th>To display this panel...</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>“Set System Processing Options panel” on page 673</td>
<td>Option 0</td>
</tr>
<tr>
<td>“Set Print Processing Options panel” on page 668</td>
<td>Option 0.1</td>
</tr>
<tr>
<td>“FM/DB2 Systems Options (1 of 4) panel” on page 577</td>
<td>Option 0.2</td>
</tr>
<tr>
<td>“Set Batch Job Card Information panel” on page 661</td>
<td>Option 0.3</td>
</tr>
<tr>
<td>“Compiler Language Selection panel” on page 413</td>
<td>Option 0.4</td>
</tr>
<tr>
<td>“Set COBOL Processing Options panel” on page 662</td>
<td>Option 0.5</td>
</tr>
<tr>
<td>“Set HLASM Processing Options panel” on page 663</td>
<td>Option 0.6</td>
</tr>
<tr>
<td>“Set PL/I Processing Options panel” on page 667</td>
<td>Option 0.7</td>
</tr>
<tr>
<td>“Set Temporary Data Set Allocation Options panel” on page 676</td>
<td>Option 0.8</td>
</tr>
<tr>
<td>“Set Output Data Set Allocation Options panel” on page 666</td>
<td>Option 0.9</td>
</tr>
<tr>
<td>“ISPF Settings panel” on page 598</td>
<td>Option 0.10</td>
</tr>
<tr>
<td>“Editor Options (1 of 8) panel” on page 528</td>
<td>Option 1 or 2</td>
</tr>
<tr>
<td>“Set Utility Options panel” on page 674</td>
<td>Option 3</td>
</tr>
<tr>
<td>“Print Utility Options panel” on page 619</td>
<td>Option 3.1</td>
</tr>
<tr>
<td>“Copy Options panel” on page 415</td>
<td>Option 3.3</td>
</tr>
<tr>
<td>“FM/DB2 Object List Options (1 of 2) panel” on page 572</td>
<td>Option 3.4</td>
</tr>
<tr>
<td>“Import Options panel” on page 588</td>
<td>Option 3.6</td>
</tr>
<tr>
<td>“Export Options (1 of 3) panel” on page 558</td>
<td>Option 3.7</td>
</tr>
<tr>
<td>“LOAD Utility Options panel” on page 607</td>
<td>Option 3.1</td>
</tr>
<tr>
<td>“LISTDEF Options panel” on page 599</td>
<td>Option 3.UL</td>
</tr>
</tbody>
</table>
Set Processing Options panel

To display this panel... Use/do this

“OPTIONS Options panel” on page 612 Option 3.UO
“TEMPLATE Options (1 of 2) panel” on page 707 Option 3.UT
“UNLOAD Options panel” on page 713 Option 3.UU

Related tasks
• “Setting default processing options” on page 39

Set System Processing Options panel

You use the Set System Processing Options panel to navigate to other options panels.

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Help</th>
<th>Set System Processing Options</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Print</td>
<td>Print settings</td>
<td></td>
</tr>
<tr>
<td>2 System</td>
<td>FM/DB2 system options</td>
<td></td>
</tr>
<tr>
<td>3 Batch</td>
<td>Job card specifications</td>
<td></td>
</tr>
<tr>
<td>4 LANG</td>
<td>Compiler language selection</td>
<td></td>
</tr>
<tr>
<td>5 COBOL</td>
<td>COBOL compiler specifications</td>
<td></td>
</tr>
<tr>
<td>6 HLASM</td>
<td>HLASM compiler specifications</td>
<td></td>
</tr>
<tr>
<td>7 PL/I</td>
<td>PL/I compiler specifications</td>
<td></td>
</tr>
<tr>
<td>8 Temporary</td>
<td>Temporary Data Set Allocations</td>
<td></td>
</tr>
<tr>
<td>9 Output</td>
<td>Output Data Set Allocations</td>
<td></td>
</tr>
<tr>
<td>10 Trace</td>
<td>Trace options</td>
<td></td>
</tr>
<tr>
<td>11 ISPF</td>
<td>ISPF settings</td>
<td></td>
</tr>
</tbody>
</table>

Command ===> F1=Help F2=Split F3=Exit F7=Backward F8=Forward F9=Swap F12=Cancel

1 Print
Displays the Set Print Processing Options panel

2 System
Displays the first FM/DB2 System Options panel

3 Batch
Displays the Set Batch Job Card Information panel

4 LANG
Displays the Compiler Language Selection panel

5 COBOL
Displays the Set COBOL Processing Options panel

6 HLASM
Displays the Set HLASM Processing Options panel

7 PL/I
Displays the Set PL/I Processing Options panel
Set System Processing Options panel

8 Temporary
Displays the Set Temporary Data Set Allocation Options panel

9 Output
Displays the Set Output Data Set Allocation Options panel

10 Trace
Displays the Set Trace options panel

11 ISPF
Displays the ISPF Settings panel

For a description of the fields on the ISPF Settings panel, see the relevant z/OS ISPF User's Guide Vol II.

Parent panels
- “Set Processing Options panel” on page 671

Child panels

<table>
<thead>
<tr>
<th>To display this panel...</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>“Set Print Processing Options panel” on page 668</td>
<td>Option 1</td>
</tr>
<tr>
<td>“FM/DB2 Systems Options (1 of 4) panel” on page 577</td>
<td>Option 2</td>
</tr>
<tr>
<td>“Set Batch Job Card Information panel” on page 661</td>
<td>Option 3</td>
</tr>
<tr>
<td>“Compiler Language Selection panel” on page 413</td>
<td>Option 4</td>
</tr>
<tr>
<td>“Set COBOL Processing Options panel” on page 662</td>
<td>Option 5</td>
</tr>
<tr>
<td>“Set HLASM Processing Options panel” on page 663</td>
<td>Option 6</td>
</tr>
<tr>
<td>“Set PL/I Processing Options panel” on page 667</td>
<td>Option 7</td>
</tr>
<tr>
<td>“Set Temporary Data Set Allocation Options panel” on page 676</td>
<td>Option 8</td>
</tr>
<tr>
<td>“Set Output Data Set Allocation Options panel” on page 666</td>
<td>Option 9</td>
</tr>
<tr>
<td>“Set Trace options panel” on page 677</td>
<td>Option 10</td>
</tr>
<tr>
<td>“ISPF Settings panel” on page 598</td>
<td>Option 11</td>
</tr>
</tbody>
</table>

Related tasks
- “Setting default processing options” on page 39

Set Utility Options panel

You use the Set Utility Processing Options panel to navigate to other options panels.
Set Utility Processing Options panel

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>Set Utility Processing Options</td>
</tr>
<tr>
<td>1 Print</td>
<td>Print utility options</td>
</tr>
<tr>
<td>3 Copy</td>
<td>Copy utility options</td>
</tr>
<tr>
<td>4 Object list</td>
<td>Object list utility options</td>
</tr>
<tr>
<td>6 Import</td>
<td>Import utility options</td>
</tr>
<tr>
<td>7 Export</td>
<td>Export utility options</td>
</tr>
<tr>
<td>L LOAD</td>
<td>DB2 LOAD utility options</td>
</tr>
<tr>
<td>UL LISTDEF</td>
<td>DB2 Utility LISTDEF options</td>
</tr>
<tr>
<td>UO OPTIONS</td>
<td>DB2 Utility OPTIONS options</td>
</tr>
<tr>
<td>UT TEMPLATE</td>
<td>DB2 Utility TEMPLATE options</td>
</tr>
<tr>
<td>UU Unload</td>
<td>DB2 Unload utility options</td>
</tr>
</tbody>
</table>

Command ===>
F1=Help  F2=Split  F3=Exit  F7=Backward  F8=Forward  F9=Swap  F12=Cancel

1 Print
Displays the Copy Options panel

3 Copy
Displays the Copy Options panel

4 Object list
Displays the FM/DB2 Object List Options (1 of 2) panel

6 Import
Displays the Import Options panel

7 Export
Displays the first Export Options panel

L LOAD
Displays the Load Utility Options panel

UL LISTDEF
Displays the LISTDEF Options panel

UO OPTIONS
Displays the OPTIONS Options panel

UT TEMPLATE
Displays the TEMPLATE Options (1 of 2) panel

UU Unload
Displays the UNLOAD Options panel

Parent panels
- “Set Processing Options panel” on page 671
Set Utility Processing Options panel

Child panels

To display this panel... Use/do this

"Copy Options panel” on page 415 Option 3
"FM/DB2 Object List Options (1 of 2) panel” on page 572 Option 4
“Import Options panel” on page 588 Option 6
“Export Options (1 of 3) panel” on page 588 Option 7
“LOAD Utility Options panel” on page 607 Option L
“LISTDEF Options panel” on page 599 Option UL
“OPTIONS Options panel” on page 612 Option UO
“TEMPLATE Options (1 of 2) panel” on page 707 Option UT
“UNLOAD Options panel” on page 713 Option UU

Related tasks

- “Setting default processing options” on page 39

Set Temporary Data Set Allocation Options panel

Panel and field definitions

<table>
<thead>
<tr>
<th>Process Options Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2) Set Temporary Data Set Allocation Options</td>
</tr>
</tbody>
</table>

Set processing options as desired and enter EXIT (F3) to save your changes. Enter RESET to restore installation defaults.

Temporary data set allocation defaults:

- Unit . . . . . . . SYSALLDA leave blank for default
- Data class . . . . leave blank for default
- Storage class . . . . leave blank for default
- Management class . . . . leave blank for default

Default High Level Qualifiers:

Temporary Data Sets ________________ HLQ (opt. &USER/&PREFIX)

Command ===> F1=Help F2=Split F3=Exit F4=Retrieve F6=Reset F7=Backward F9=Swap F10=Actions F12=Cancel

Figure 120. Set Temporary Data Set Allocation Options panel

Temporary data set allocation defaults

- **Unit** Can be one of these:
  - A 3-digit hexadecimal device number.
  - A 4-digit hexadecimal device number prefixed with a slash (/).
  - Device type (generic name).
  - Group name (installation-defined symbolic name).

- **Data class** Name of a data class defined in your Storage Management System (SMS) installation. Do not enter a value on a system without SMS.
Set Temporary Data Set Allocation Options panel

Storage class
Name of a storage class defined in your Storage Management System (SMS) installation. Do not enter a value on a system without SMS.

Management class
Name of a management class defined in your Storage Management System (SMS) installation. Do not enter a value on a system without SMS.

Default High Level Qualifiers

Temporary Data Sets
Can be any multi-level qualifier, which is used by File Manager as the high-level qualifier (HLQ) in names of created datasets. You can use these symbols as part of the HLQ:

&USER
Represents the user ID.

&PREFIX
Represents the TSO prefix.

The default is none (unless changed in your File Manager installation).

Parent panels
- “Set Processing Options panel” on page 671

Child panels
- None.

Comments
- None.

Set Trace options panel

The Set Trace options panel allows you to view and modify parameters that control the Trace output produced by FM/DB2 when it is run in debug mode.
Set Trace options panel

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFE2)</td>
<td>Set Trace options</td>
<td></td>
</tr>
</tbody>
</table>

Trace Options:
- Trace destination
  1. FMNTRC
  2. Terminal
  3. Data set
  
  Data set name: JOHNLEV.FMN.TRACE
  Output class: A
  Trace limit: 500

Command ===> F1=Help  F2=Split  F3=Exit  F4=CRetrieve  F6=Reset  F7=Backward  F8=Forward  F9=Swap  F10=Actions  F12=Cancel

Figure 121. Set Trace options panel

Trace destination
- Specifies where debug trace FM/DB2 output is to be sent:
  - FMNTRC
    - Send trace output to the current FMNTRC allocation.
  - Terminal
    - Send trace output to the terminal.
  - Data set
    - Send trace output to the temporary data set specified by the Data set name field.

Data set name
- The temporary trace data set where FM/DB2 trace output is directed when the Data set trace destination is specified.

Output class
- The class of the JES spool queue to be used when the PRINT command is issued when browsing the temporary trace data set.

Trace limit
- When run in debug mode, FM/DB2 writes the Trace output to a buffer (as for normal debug mode), but the buffer is only written to the Trace destination, if FM/DB2 abnormally terminates.

This field specifies the number of Trace statements that are retained in the buffer. If you specify a value of zero, background debug mode is turned off.

Parent panels

“Set System Processing Options panel” on page 673
Child panels
None.

Related tasks

Sort Fields panel
You use the Sort Fields panel to sort multiple columns within the current object list.

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>Sort Fields</td>
<td>Row 1 to 11 of 43</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Sort Column Number</th>
<th>Column Asc/Desc</th>
<th>Column Name</th>
<th>Column Type</th>
<th>DB2 Column Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>ASC</td>
<td>NAME</td>
<td>VARCHAR</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>ASC</td>
<td>CREATOR</td>
<td>CHAR</td>
<td>2</td>
</tr>
<tr>
<td>0</td>
<td>ASC</td>
<td>TYPE</td>
<td>CHAR</td>
<td>3</td>
</tr>
<tr>
<td>0</td>
<td>ASC</td>
<td>DBNAME</td>
<td>CHAR</td>
<td>4</td>
</tr>
<tr>
<td>0</td>
<td>ASC</td>
<td>TSNAME</td>
<td>CHAR</td>
<td>5</td>
</tr>
<tr>
<td>0</td>
<td>ASC</td>
<td>DBID</td>
<td>SMALLINT</td>
<td>6</td>
</tr>
<tr>
<td>0</td>
<td>ASC</td>
<td>OBID</td>
<td>SMALLINT</td>
<td>7</td>
</tr>
<tr>
<td>0</td>
<td>ASC</td>
<td>COCOUNT</td>
<td>SMALLINT</td>
<td>8</td>
</tr>
<tr>
<td>0</td>
<td>ASC</td>
<td>EDPROC</td>
<td>CHAR</td>
<td>9</td>
</tr>
<tr>
<td>0</td>
<td>ASC</td>
<td>VALPROC</td>
<td>CHAR</td>
<td>10</td>
</tr>
<tr>
<td>0</td>
<td>ASC</td>
<td>CLUSTERTYPE</td>
<td>CHAR</td>
<td>11</td>
</tr>
</tbody>
</table>

Command ===> Scroll PAGE
F1=Help F2=Split F3=Exit F4=Retrieve F7=Backward F8=Forward
F9=Swap F10=Actions F12=Cancel

Sort Column Number
The relative position of this column in the sort sequence. Specify 1 for the most significant column, 2 for the next significant, and so on.

Asc/Desc
Indicates if you want FM/DB2 to sort the column in ascending or descending sequence:

ASC  Ascending. To change from DESC, overtype with ASC or A.
DESC Descending. To change from ASC, overtype with DESC or D.

Column Name
The name of the column as held in the DB2 catalog table.

Column Type
The data type of the column.

DB2 Column Number
The number of the column as held in the DB2 catalog table.

Parent panels
The Sort Fields panel is displayed when you enter SORT on the command line from any object list and the cursor is not positioned within any column.
Sort Fields panel

Child panels

To display this panel... Use/do this

- "Databases panel" on page 500 Object Type 1
- "Table Spaces panel" on page 698 Object Type 2
- "Tables, Views and Aliases panel" on page 704 Object Type 3
- "Indexes panel" on page 595 Object Type 4
- "Columns panel" on page 411 Object Type 5
- "Synonyms panel" on page 687 Object Type 6
- "Application Plans panel" on page 389 Object Type 7
- "Application Packages panel" on page 387 Object Type 8
- "Collections panel" on page 396 Object Type 9
- "Database Request Modules panel" on page 498 Object Type 10
- "Schemas panel" on page 657 Object Type 11
- "Distinct Types panel" on page 523 Object Type 12
- "Functions panel" on page 584 Object Type 13
- "Stored Procedures panel" on page 685 Object Type 14
- "Triggers panel" on page 712 Object Type 15

Related tasks
- "Sorting the displayed data" on page 257

SQL Prototyping, Execution and Analysis panel

You use the SQL Prototyping, Execution and Analysis panel to select the FM/DB2 SQL function you want to use.

Panel and field definitions

<table>
<thead>
<tr>
<th>Command</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>F1=Help</td>
<td>F2=Split</td>
<td>F3=Exit</td>
<td>F7=Backward</td>
<td>F8=Forward</td>
</tr>
<tr>
<td>F12=Cancel</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>SQL Prototyping, Execution and Analysis</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 Basic</td>
<td>Prototype SELECT statements (basic)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2 Advanced</td>
<td>Prototype SELECT statements (advanced)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3 Enter</td>
<td>Enter, execute and explain SQL statements</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4 Edit</td>
<td>Edit and execute SQL statements from a data set</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5 Utility</td>
<td>Manage tables used by SQL explain</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
SQL Prototyping, Execution and Analysis panel

1 (Basic)
   Use this option to build and execute a basic SELECT statement.

2 (Advanced)
   Use this option to build and execute a complex SELECT statement.

3 (Enter)
   Use this option to enter, execute, and explain SQL statements.

4 (Edit)
   Use this option to edit and execute SQL statements from a data set.

5 (Utility)
   Use this option to select utilities to support the use of the DB2 EXPLAIN statement.

Parent panels
   • “Primary Option Menu panel” on page 617

Child panels

<table>
<thead>
<tr>
<th>To display this panel...</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>“Basic SELECT Prototyping panel” on page 391</td>
<td>Option 1</td>
</tr>
<tr>
<td>“Advanced SELECT Prototyping panel” on page 368</td>
<td>Option 2</td>
</tr>
<tr>
<td>“Enter, Execute and Explain SQL Statements panel” on page 555</td>
<td>Option 3</td>
</tr>
<tr>
<td>“Edit/Execute SQL (Data Set) panel” on page 526</td>
<td>Option 4</td>
</tr>
<tr>
<td>“Explain Utilities panel” on page 556</td>
<td>Option 5</td>
</tr>
</tbody>
</table>

Related tasks
   • “Using basic SQL prototyping” on page 317
   • “Using advanced SQL prototyping” on page 325
   • “Entering, executing, and explaining SQL statements” on page 351
   • “Editing and executing SQL statements from a data set” on page 351
   • “Managing tables used by SQL explain” on page 355

Statement Table Rows panel

FM/DB2 displays the Statement Table Rows panel if you use the SR line command against an item listed on the “Plan Table Rows panel” on page 615.

Use this panel to display the rows from a table.
The Statement Table Rows panel shows you rows from a statement table:

C Input field where you can enter one of the following line commands:

- DK Delete rows for package
- DP Delete rows for plan
- DQ Delete rows for query number
- I Interpret EXPLAIN data
- K Show package
- M Show DBRM
- P Show plan

Query Numb
A number that identifies the SQL statement.

Applname
PLAN name or blank for dynamic EXPLAIN. This information only is displayed for the PLAN format of the panel.

Collect.
Collection name. This information only is displayed for the COL format of the panel.

Progname
DBRM/package name.

Stmt Type
Statement type.

C C Cost category:
A Not using default values.
B Using default values when calculating costs.

Est. SU
Estimated processor cost in service units.
Statement Table Rows panel

Est. P ms.
Estimated processor cost in milliseconds.

Reason
A string indicating the reason for putting an estimate into cost category B (see C C).

Parent panels
• “Plan Table Rows panel” on page 615

Child panels

To display this panel... Use/do this
“EXPLAIN Interpretation panel” on page 556 Line command I
“Application Packages panel” on page 387 Line command K

Related tasks
• “Managing tables used by SQL explain” on page 355

Storage Group(s) panel

You use the Stored Group(s) panel to display information about the storage groups you have selected.

Panel and field definitions

The columns that are displayed include the SEL field and columns of SYSIBM.SYSSTOGROUP.

<table>
<thead>
<tr>
<th>SEL</th>
<th>NAME</th>
<th>CREATOR</th>
<th>VCATNAME</th>
<th>SPACE</th>
<th>SPCDATE</th>
<th>CREATEDBY</th>
<th>STATSTIME</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>DSNB810</td>
<td>JCULEN</td>
<td>DFA2</td>
<td>0</td>
<td>JCULEN</td>
<td>0001-01-01-00.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>JLV3G810</td>
<td>KEISTEW</td>
<td>DFA2</td>
<td>0</td>
<td>KEISTEW</td>
<td>0001-01-01-00.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>JOHNB810</td>
<td>KEISTEW</td>
<td>DFA2</td>
<td>0</td>
<td>KEISTEW</td>
<td>0001-01-01-00.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>KPSB810</td>
<td>KEISTEW</td>
<td>DFA2</td>
<td>0</td>
<td>KEISTEW</td>
<td>0001-01-01-00.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>KPSAGB810</td>
<td>KEISTEW</td>
<td>DFA2</td>
<td>0</td>
<td>KEISTEW</td>
<td>0001-01-01-00.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SYDFLT</td>
<td>JCULEN</td>
<td>DFA2</td>
<td>0</td>
<td>JCULEN</td>
<td>0001-01-01-00.</td>
<td></td>
</tr>
</tbody>
</table>

Figure 123. Storage Group(s) panel

Note:
1. The system option, Show all catalog table columns, affects which columns FM/DB2 displays when you use the Object List utility.
Storage Group(s) panel

2. You can also customize which columns FM/DB2 displays when you use the Object List utility. For details, see the File Manager Customization Guide.

To view other columns on this panel, press the Right function key (F11).

For a complete list of columns, refer to “DB2 catalog tables” in the DB2 for z/OS SQL Reference relevant to your version of DB2.

SEL Line command area.

For a list of the line commands you can use on this panel, see Table 22 on page 250.

Parent panels

- “Object List Utility panel” on page 610

Child panels

<table>
<thead>
<tr>
<th>To display this panel...</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>“Databases panel” on page 500</td>
<td>Line command D</td>
</tr>
<tr>
<td>“Storage Group Usage Privileges panel”</td>
<td>Line command P</td>
</tr>
<tr>
<td>“Table Space Parts panel” on page 697</td>
<td>Line command PT</td>
</tr>
<tr>
<td>“Display Row panel” on page 523</td>
<td>Line command ROW</td>
</tr>
<tr>
<td>“Table Spaces panel” on page 698</td>
<td>Line command S</td>
</tr>
<tr>
<td>“Volumes panel” on page 724</td>
<td>Line command VOL</td>
</tr>
<tr>
<td>“Indexes panel” on page 595</td>
<td>Line command X</td>
</tr>
<tr>
<td>“Index Parts panel” on page 595</td>
<td>Primary command XPT</td>
</tr>
</tbody>
</table>

Related tasks

- “Working with object list panels” on page 244
- “Using the line command area (Cmd)” on page 250

Related references

- “SORT primary command” on page 777

Related tasks

- “Managing tables used by SQL explain” on page 355

Storage Group Usage Privileges panel
You use the Stored Procedures panel to display information about the stored procedures you have selected.
Panel and field definitions

The columns that are displayed include the SEL field and columns of SYSIBM.SYSROUTINES.

Note:
1. The system option, Show all catalog table columns, affects which columns FM/DB2 displays when you use the Object List utility.
2. You can also customize which columns FM/DB2 displays when you use the Object List utility. For details, see the File Manager Customization Guide.

To view other columns on this panel, press the Right function key (F11).

For a complete list of columns, refer to “DB2 catalog tables” in the DB2 for z/OS SQL Reference relevant to your version of DB2.

SEL Line command area.

For a list of the line commands you can use on this panel, see Table 22 on page 250.

Parent panels
- “Object List Utility panel” on page 610
- “Schemas panel” on page 657

Child panels

<table>
<thead>
<tr>
<th>To display this panel...</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>“Alter Stored Procedure panel” on page 374</td>
<td>Line command A</td>
</tr>
<tr>
<td>See “Comment panels” on page 413</td>
<td>Line command COM</td>
</tr>
<tr>
<td>“Create Procedure panel” on page 452</td>
<td>Line command CR</td>
</tr>
<tr>
<td>See “Drop panels” on page 524</td>
<td>Line command DR</td>
</tr>
<tr>
<td>“Grant privileges panels” on page 587</td>
<td>Line command G</td>
</tr>
</tbody>
</table>
### Stored Procedures panel

<table>
<thead>
<tr>
<th>To display this panel...</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>“Generate SQL From DB2 Catalog panel” on page 585</td>
<td>Line command GEN</td>
</tr>
<tr>
<td>“Details panels” on page 522</td>
<td>Line command I</td>
</tr>
<tr>
<td>“Privileges panels” on page 624</td>
<td>Line command P</td>
</tr>
<tr>
<td>“Routine Parameters panel” on page 648</td>
<td>Line command PARM</td>
</tr>
<tr>
<td>“Application Packages panel” on page 387</td>
<td>Line command PKG</td>
</tr>
<tr>
<td>Line command PS</td>
<td></td>
</tr>
<tr>
<td>“Revoke privileges panels” on page 647</td>
<td>Line command R</td>
</tr>
<tr>
<td>“Display Row panel” on page 523</td>
<td>Line command ROW</td>
</tr>
<tr>
<td>“Schemas panel” on page 657</td>
<td>Line command SCH</td>
</tr>
<tr>
<td>“Sort Fields panel” on page 679</td>
<td>Primary command SORT</td>
</tr>
</tbody>
</table>

### Related tasks
- “Working with object list panels” on page 244
- “Using the line command area (Cmd)” on page 250

### Synonym Dependencies panel

FM/DB2 displays the Synonym Dependencies panel if you use the SYN line command against an item listed on:
- The “Application Plans panel” on page 389
- The “Application Packages panel” on page 387

### Synonyms panel

You use the Synonyms panel to list synonyms in the DB2 catalog.

### Panel and field definitions

The columns that are displayed include the SEL field and columns of SYSIBM.SYSSYNONYMS.
Synonyms panel

Note:
1. The system option, **Show all catalog table columns**, affects which columns FM/DB2 displays when you use the Object List utility.
2. You can also customize which columns FM/DB2 displays when you use the Object List utility. For details, see the [File Manager Customization Guide](#).

To view other columns on this panel, press the Right function key (F11).

For a complete list of columns, refer to “DB2 catalog tables” in the *DB2 for z/OS SQL Reference* relevant to your version of DB2.

SEL  Line command area.

For a list of the line commands you can use on this panel, see [Table 22 on page 250](#).

Parent panels
- “Object List Utility panel” on page 610
- “Tables, Views and Aliases panel” on page 704

Child panels

<table>
<thead>
<tr>
<th>To display this panel...</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>“Create Synonym panel” on page 458</td>
<td>Line command CR</td>
</tr>
<tr>
<td>“Create Alias panel” on page 430</td>
<td>Line command CRA</td>
</tr>
<tr>
<td>See “Drop panels” on page 524</td>
<td>Line command DR</td>
</tr>
<tr>
<td>“Details panels” on page 522</td>
<td>Line command I</td>
</tr>
<tr>
<td>“Display Row panel” on page 523</td>
<td>Line command ROW</td>
</tr>
<tr>
<td>“Tables, Views and Aliases panel” on page 704</td>
<td>Line command T</td>
</tr>
<tr>
<td>“Sort Fields panel” on page 679</td>
<td>Primary command SORT</td>
</tr>
</tbody>
</table>
Table Browse panel

Panel and field definitions

<table>
<thead>
<tr>
<th>Process Options Utilities Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
</tr>
<tr>
<td>TABLE PNNUSER.EMP</td>
</tr>
<tr>
<td>EMPNO FIRSTNAME MIDINIT LASTNAME WORKDEPT PHONENO HIREDATE JOB</td>
</tr>
<tr>
<td>#1 #2 #3 #4 #5 #6 #7 #8 +</td>
</tr>
<tr>
<td>CH(6) VARCHAR(12) CH(1) VARCHAR(15) CH(3) CH(4) DATE CH(8)</td>
</tr>
<tr>
<td>PU+&gt; &lt;-----+----1-&gt; -</td>
</tr>
<tr>
<td>**** Top of data ****</td>
</tr>
<tr>
<td>000010 CHRISTINE&lt; I HAAS&lt; A00 3978 01/01/1965 PRES</td>
</tr>
<tr>
<td>000110 VINCENZO&lt; G LUCCHESI&lt; A00 3490 05/16/1958 SALESR</td>
</tr>
<tr>
<td>200010 DIAN&lt; J HEMMINGER&lt; A00 3978 01/01/1965 SALESR</td>
</tr>
<tr>
<td>**** End of data ****</td>
</tr>
</tbody>
</table>

Command ===> Scroll PAGE
F1=Help  F2=Zoom  F3=Exit  F4=Retrieve  F5=RFind  F6=RChange
F7=Up  F8=Down  F9=Swap  F10=Left  F11=Right  F12=Cancel
Table Browse panel

**Parent panels**

- "DB2 Browse panel" on page 501
- "Column Selection/Edit panel" on page 407

**Child panels**

None.

**Related tasks**

- Process Options
- Utilities
- Help

---

**FM/DB2 (DFG2) Table Browse**

44 rows fetched

**TABLE FMUSER.EMP**

<table>
<thead>
<tr>
<th>EMPNO</th>
<th>FIRSTNME</th>
<th>MIDINIT</th>
<th>LASTNAME</th>
<th>WORKDEPT</th>
<th>PHONENO</th>
<th>HIREDATE</th>
<th>JOB</th>
</tr>
</thead>
<tbody>
<tr>
<td>000010</td>
<td>CHRISTINE</td>
<td>I</td>
<td>HAAS</td>
<td>A00</td>
<td>3978</td>
<td>01/01/1965</td>
<td>PRES</td>
</tr>
<tr>
<td>000020</td>
<td>MICHAEL</td>
<td>L</td>
<td>THOMPSON</td>
<td>B01</td>
<td>3476</td>
<td>10/10/1973</td>
<td>MANAGE</td>
</tr>
<tr>
<td>000030</td>
<td>SALLY</td>
<td>A</td>
<td>KWAY</td>
<td>C01</td>
<td>4738</td>
<td>04/05/1975</td>
<td>MANAGE</td>
</tr>
<tr>
<td>000050</td>
<td>JOHN</td>
<td>B</td>
<td>GEYER</td>
<td>E01</td>
<td>6789</td>
<td>08/17/1949</td>
<td>MANAGE</td>
</tr>
<tr>
<td>000060</td>
<td>IRVING</td>
<td>F</td>
<td>STERN</td>
<td>D11</td>
<td>6423</td>
<td>09/14/1973</td>
<td>MANAGE</td>
</tr>
<tr>
<td>000070</td>
<td>EVAN</td>
<td>D</td>
<td>PULASKI</td>
<td>D21</td>
<td>7831</td>
<td>09/30/1980</td>
<td>MANAGE</td>
</tr>
<tr>
<td>000090</td>
<td>EILEEN</td>
<td>W</td>
<td>HENDERSON</td>
<td>E11</td>
<td>5498</td>
<td>08/15/1970</td>
<td>MANAGE</td>
</tr>
<tr>
<td>000100</td>
<td>THEODORE</td>
<td>Q</td>
<td>SPENESS</td>
<td>E21</td>
<td>0972</td>
<td>06/19/1980</td>
<td>MANAGE</td>
</tr>
<tr>
<td>000110</td>
<td>VINCENZO</td>
<td>G</td>
<td>LUCCHESI</td>
<td>A00</td>
<td>3490</td>
<td>05/16/1958</td>
<td>SALESR</td>
</tr>
<tr>
<td>000120</td>
<td>SEAN</td>
<td>O'CONNELL</td>
<td>A00</td>
<td>2167</td>
<td>12/05/1963</td>
<td>CLERK</td>
<td></td>
</tr>
<tr>
<td>000130</td>
<td>DOLORES</td>
<td>M</td>
<td>QUINTANA</td>
<td>C01</td>
<td>4578</td>
<td>07/28/1971</td>
<td>ANALYS</td>
</tr>
<tr>
<td>000140</td>
<td>HEATHER</td>
<td>A</td>
<td>NICHOLS</td>
<td>C01</td>
<td>1793</td>
<td>12/15/1976</td>
<td>ANALYS</td>
</tr>
<tr>
<td>000150</td>
<td>BRUCE</td>
<td>D</td>
<td>ADAMSON</td>
<td>D11</td>
<td>4510</td>
<td>02/12/1972</td>
<td>DESIGN</td>
</tr>
<tr>
<td>000160</td>
<td>ELIZABETH</td>
<td>R</td>
<td>PIANKA</td>
<td>D11</td>
<td>3782</td>
<td>10/11/1977</td>
<td>DESIGN</td>
</tr>
<tr>
<td>000170</td>
<td>MASATOSHI</td>
<td>J</td>
<td>YOSHIMURA</td>
<td>D11</td>
<td>2890</td>
<td>09/15/1978</td>
<td>DESIGN</td>
</tr>
<tr>
<td>000180</td>
<td>MARILYN</td>
<td>S</td>
<td>SCOUTTEN</td>
<td>D11</td>
<td>1682</td>
<td>07/07/1973</td>
<td>DESIGN</td>
</tr>
<tr>
<td>000190</td>
<td>JAMES</td>
<td>H</td>
<td>WALKER</td>
<td>D11</td>
<td>2986</td>
<td>07/26/1974</td>
<td>DESIGN</td>
</tr>
<tr>
<td>000200</td>
<td>DAVID</td>
<td>D</td>
<td>BROWN</td>
<td>D11</td>
<td>4501</td>
<td>03/03/1966</td>
<td>DESIGN</td>
</tr>
<tr>
<td>000210</td>
<td>WILLIAM</td>
<td>T</td>
<td>JONES</td>
<td>D11</td>
<td>0942</td>
<td>04/11/1979</td>
<td>DESIGN</td>
</tr>
<tr>
<td>000220</td>
<td>JENNIFER</td>
<td>K</td>
<td>LUTZ</td>
<td>D11</td>
<td>0672</td>
<td>08/29/1968</td>
<td>DESIGN</td>
</tr>
<tr>
<td>000230</td>
<td>JAMES</td>
<td>J</td>
<td>JEFFERSON</td>
<td>D21</td>
<td>2094</td>
<td>11/21/1966</td>
<td>CLERK</td>
</tr>
<tr>
<td>000240</td>
<td>SALVATORE</td>
<td>M</td>
<td>MARINO</td>
<td>D21</td>
<td>3780</td>
<td>12/05/1979</td>
<td>CLERK</td>
</tr>
<tr>
<td>000250</td>
<td>DANIEL</td>
<td>S</td>
<td>SMITH</td>
<td>D21</td>
<td>0961</td>
<td>10/30/1969</td>
<td>CLERK</td>
</tr>
<tr>
<td>000260</td>
<td>SYBILL</td>
<td>V</td>
<td>JOHNSON</td>
<td>D21</td>
<td>8953</td>
<td>09/11/1975</td>
<td>CLERK</td>
</tr>
<tr>
<td>000270</td>
<td>MARIJA</td>
<td>L</td>
<td>PEREZ</td>
<td>D21</td>
<td>9061</td>
<td>09/30/1980</td>
<td>CLERK</td>
</tr>
<tr>
<td>000280</td>
<td>ETHEL</td>
<td>R</td>
<td>SCHNEIDER</td>
<td>E11</td>
<td>8997</td>
<td>03/24/1967</td>
<td>OPERAT</td>
</tr>
<tr>
<td>000290</td>
<td>JOHN</td>
<td>R</td>
<td>PARKER</td>
<td>E11</td>
<td>4502</td>
<td>05/30/1980</td>
<td>OPERAT</td>
</tr>
<tr>
<td>000300</td>
<td>PHILLIP</td>
<td>X</td>
<td>SMITH</td>
<td>E11</td>
<td>2095</td>
<td>06/19/1972</td>
<td>OPERAT</td>
</tr>
<tr>
<td>000310</td>
<td>MAUDE</td>
<td>F</td>
<td>SETRIGHT</td>
<td>E11</td>
<td>3332</td>
<td>09/12/1964</td>
<td>OPERAT</td>
</tr>
<tr>
<td>000320</td>
<td>RAMMAL</td>
<td>V</td>
<td>MEHTA</td>
<td>E21</td>
<td>9990</td>
<td>07/07/1965</td>
<td>FIELDR</td>
</tr>
<tr>
<td>000330</td>
<td>WING</td>
<td>L</td>
<td>LEE</td>
<td>E21</td>
<td>2103</td>
<td>02/23/1976</td>
<td>FIELDR</td>
</tr>
<tr>
<td>000340</td>
<td>JASON</td>
<td>R</td>
<td>GOUNOT</td>
<td>E21</td>
<td>5698</td>
<td>05/05/1947</td>
<td>FIELDR</td>
</tr>
<tr>
<td>200010</td>
<td>DIAN</td>
<td>J</td>
<td>HEMMINGER</td>
<td>A00</td>
<td>3978</td>
<td>01/01/1965</td>
<td>SALESR</td>
</tr>
<tr>
<td>200120</td>
<td>GREDD</td>
<td>A</td>
<td>ORLANDO</td>
<td>A00</td>
<td>2167</td>
<td>05/05/1972</td>
<td>CLERK</td>
</tr>
<tr>
<td>200140</td>
<td>KIMM</td>
<td>N</td>
<td>NATZ</td>
<td>C01</td>
<td>1793</td>
<td>12/15/1976</td>
<td>ANALYS</td>
</tr>
<tr>
<td>200170</td>
<td>KIYOSHI</td>
<td>K</td>
<td>YAMAMOTO</td>
<td>D11</td>
<td>2890</td>
<td>09/15/1978</td>
<td>DESIGN</td>
</tr>
<tr>
<td>200220</td>
<td>REBA</td>
<td>K</td>
<td>JOHN</td>
<td>D11</td>
<td>0672</td>
<td>09/28/1968</td>
<td>DESIGN</td>
</tr>
<tr>
<td>200240</td>
<td>ROBERT</td>
<td>M</td>
<td>MONTERDE</td>
<td>D21</td>
<td>3780</td>
<td>12/05/1979</td>
<td>CLERK</td>
</tr>
<tr>
<td>200280</td>
<td>EILEN</td>
<td>R</td>
<td>SCHWARTZ</td>
<td>E11</td>
<td>8997</td>
<td>03/24/1967</td>
<td>OPERAT</td>
</tr>
<tr>
<td>200310</td>
<td>MICHELLE</td>
<td>F</td>
<td>SPRINGER</td>
<td>E11</td>
<td>3332</td>
<td>09/12/1964</td>
<td>OPERAT</td>
</tr>
<tr>
<td>200330</td>
<td>HELENA</td>
<td>W</td>
<td>WONG</td>
<td>E21</td>
<td>2103</td>
<td>02/23/1976</td>
<td>FIELDR</td>
</tr>
<tr>
<td>200340</td>
<td>ROY</td>
<td>R</td>
<td>ALONZO</td>
<td>E21</td>
<td>5698</td>
<td>05/05/1947</td>
<td>FIELDR</td>
</tr>
</tbody>
</table>

**** End of data ****

---

Figure 125. Sample data
Table Check Constraint Dependencies panel

FM/DB2 displays the Table Check Constraint Dependencies panel if you use the DEP line command against an item listed on the "Table Check Constraints panel."

Table Check Constraint Details panel

FM/DB2 displays the Table Check Constraint panel if you use the I line command against an item listed on the "Table Check Constraints panel."

Table Check Constraints panel

FM/DB2 displays the Table Check Constraints panel if you use the CK line command against an item listed on the "Tables, Views and Aliases panel" on page 704.

Table Edit panel

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TABLE</td>
<td>FMNUSER.EMP</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EMPNO</td>
<td>FIRSTNAME</td>
<td>MIDINIT</td>
<td>LASTNAME</td>
</tr>
<tr>
<td>#1</td>
<td>#2</td>
<td>#3</td>
<td>#4</td>
</tr>
<tr>
<td>CH(6)</td>
<td>VARCHAR(12)</td>
<td>CH(1)</td>
<td>VARCHAR(15)</td>
</tr>
<tr>
<td>Pu-----&lt;-----+----1-&gt; - &lt;---+----1----&gt; &lt;-NF &lt;--&gt; &lt;---+----&gt;</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>000000 **** Top of data ****</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>000001 000010 CHRISTINE&lt; I HAAS&lt; A00 3978 01.01.1965</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>000002 000020 MICHAEL&lt; L THOMPSON&lt; B01 3476 10.10.1973</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>000003 000030 SALLY&lt; A KWA&lt; C01 4738 05.04.1975</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>000004 000050 JOHN&lt; B GEYER&lt; E01 6789 17.08.1949</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>000005 000060 IRVING&lt; F STERN&lt; D11 6423 14.09.1973</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>000006 000070 EVA&lt; D PULASKI&lt; D21 7831 30.09.1980</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>000007 000090 EILEEN&lt; W HENDERSON&lt; E11 5948 15.08.1970</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>000008 000100 THEODORE&lt; Q SPENSER&lt; E21 0972 19.06.1980</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>000009 000110 VINCENZO&lt; G LUCHE&lt; A00 3490 16.05.1958</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>000010 000120 sean&lt; O'CONNELL&lt; A00 2167 05.12.1963</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>000011 000130 DOLORES&lt; M QINTANA&lt; C01 4578 28.07.1971</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>000012 000140 HEATHER&lt; A NICHOLS&lt; C01 1793 15.12.1976</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>000013 000150 BRUCE&lt; A ADAMSON&lt; D11 4510 12.12.1972</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Command ===&gt; Scroll PAGE</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>F1=Help</td>
<td>F2=Zoom</td>
<td>F3=Exit</td>
<td>F4=Retrieve</td>
</tr>
<tr>
<td>F7=Up</td>
<td>F8=Down</td>
<td>F9=Swap</td>
<td>F10=Left</td>
</tr>
</tbody>
</table>

Prefix area

In TABL format mode, you can enter the following commands in the prefix area of the data display:

A  Identifies the row after which data is to be moved or copied.
B  Identifies the row before which data is to be moved or copied.
C  Copy one row.
Cn Copy n rows.
CC Copy block of rows. Mark start and end of block.
**Table Edit panel**

<table>
<thead>
<tr>
<th>Command</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>D</td>
<td>Delete one row.</td>
</tr>
<tr>
<td>Dn</td>
<td>Delete n rows.</td>
</tr>
<tr>
<td>DD</td>
<td>Delete block of rows. Mark start and end of block.</td>
</tr>
<tr>
<td>E</td>
<td>Display a panel containing information about why a change to this row failed. The E command is only valid when a row is marked in error. Rows marked in error have “=ERR”, “ERR R”, or the SQLCODE in the prefix field.</td>
</tr>
<tr>
<td>F</td>
<td>Display the first row of a block of excluded rows.</td>
</tr>
<tr>
<td>Fn</td>
<td>Display the first n rows of a block of excluded rows.</td>
</tr>
<tr>
<td>FS</td>
<td>Change the display format to SNGL and display the row.</td>
</tr>
<tr>
<td>G</td>
<td>Get the data stored on the clipboard. Rows saved on the clipboard with the S, Sn or SS prefix commands can only be retrieved from the clipboard with the G prefix command during the same Edit session. The clipboard is a temporary memory buffer that is cleared when you exit from the DB2 Edit function panel (and FM/DB2 returns you to the Primary Option Menu).</td>
</tr>
<tr>
<td>I</td>
<td>Insert one row. The row is initialized using the row description of the currently displayed row. For each column, if it has a default value (DEFAULT=“value”, and not just the DEFAULT keyword), then it is initialized to that value. Otherwise, the column is initialized according to its data type:</td>
</tr>
<tr>
<td></td>
<td><strong>Character strings</strong></td>
</tr>
<tr>
<td></td>
<td>Set to blanks</td>
</tr>
<tr>
<td></td>
<td><strong>Numeric</strong></td>
</tr>
<tr>
<td></td>
<td>Set to zeros</td>
</tr>
<tr>
<td></td>
<td><strong>Row identifier</strong></td>
</tr>
<tr>
<td></td>
<td>Set to blanks</td>
</tr>
<tr>
<td>DATE</td>
<td>Set to the current date</td>
</tr>
<tr>
<td>TIME</td>
<td>Set to the current time</td>
</tr>
<tr>
<td>TIMESTAMP</td>
<td>Set to the current date and time</td>
</tr>
</tbody>
</table>

**Note:** If a column of the inserted row can contain a null value, it is set to the null value and the null column display indicator is displayed in the displayed column.

For more information about setting the value of the null column display indicator, see “View and Edit options (options 1 and 2)” on page 43.

**Related tasks**
- “View and Edit options (options 1 and 2)” on page 43

**Related references**
- Chapter 14, “DB2 data types,” on page 361
only, insert a row with the key columns set to the key value that
originally caused the error in the dependent table of the
relationship.

**IP**\(n\) Insert \(n\) primary keys. In an Edit session of a related parent table
only, insert \(n\) rows with the key columns set to the key value that
originally caused the error in the dependent table of the
relationship. If you attempt to save the data, it results in duplicate
key errors for the second and all subsequent inserted rows, as
primary key values must be unique.

**J** Join together two or more records - as shown in TABL display
mode - into a single, longer record. The record where the prefix
command is issued is the record that becomes longer. Any
repetition count (for example, “J4”) is ignored. The J command can
only be issued against a visible record. The join process continues
downwards from the current location until:
- The new length for the record would exceed the maximum
  allowed.
- The last record in the editor session is encountered.

This prefix command is only available in an FM/DB2 LOB editor
session.

**L** Display the last row of a block of excluded rows.

**Ln** Display the last \(n\) rows of a block of excluded rows.

**LC** Convert all uppercase characters in a row to lowercase.

**Note:** The LC (LC\(n\), LCC) command affects all characters in a row,
not just characters in columns with a CHAR or VARCHAR data
type. Therefore, numeric data, such as binary data or packed
decimal data, can be corrupted when you use these commands.

**LC\(n\)** Convert all uppercase characters in \(n\) rows to lowercase. (See note
for **LC**.)

**LCC** Convert all uppercase characters in a block of rows to lowercase.
Mark start and end of block. (See note for **LC**.)

**M** Move one row.

**M\(n\)** Move \(n\) rows.

**MM** Move block of rows. Mark start and end of the block.

**O** Overlay one row.

**On** Overlay \(n\) rows.

**OO** Overlay block of rows. Mark start and end of the block.

**P** Identifies the row preceding which data is to be moved or copied.

**R or "** Repeat one row.

**Rn or "\(n\)** Repeat \(n\) rows.

**RE** Edit a related table. You can use the RE prefix command to edit the
related table when the row is marked with a error caused by a
referential constraint.
You can only enter the RE prefix command against a row marked in error, where the rightmost position of the prefix area contains an “R”. RE starts another FM/DB2 dependent edit session, in a new ISPF logical session. The new (dependent) FM/DB2 Edit session is for the related table in the relationship that caused the referential constraint error. For more details, see “Editing related tables” on page 149.

**Note:** When you use the RE prefix command for a row whose prefix area contains “=ERR R”, take care to avoid the editor interpreting the RE command as an R prefix command instead by ensuring the “RE” is followed by one or more blanks).

The RE command lets you quickly identify and resolve referential constraint errors. After you use the RE command to edit a related table, you cannot save data until you end the dependent Edit session of the related table.

This restriction also applies to tables that are part of a larger referential structure. For example, say A, B and C are related tables and A and B are part of a relationship in which A is the parent table, and B and C are part of a relationship in which B is the parent table. If you edit table A, then use the RE command to edit related table B, then use the RE command in the Edit session of table B to edit table C, it would result in 3 FM/DB2 Edit sessions. The Edit sessions for tables B and C are both dependent edit sessions of the Edit session for table A. You cannot save data in the Edit session for table A until you end the Edit sessions for tables B and C.

**RR or ""**
Repeat block of rows. Mark start and end of the block.

**RRn or ""n**
Repeat block of rows $n$ times. Mark start and end of block.

**S**
Save row on clipboard. The clipboard is a temporary memory buffer that is cleared when you exit from the DB2 Edit function panel (and FM/DB2 returns you to the Primary Option Menu).

**Sn**
Save $n$ rows on the clipboard.

**SP**
Split a single record into multiple pieces, the length of each piece depends on the terminal characteristics. Each piece appears in TABL display mode with the end of the record marker displayed - in other words all of the data for each piece is visible in TABL display mode without the need to scroll.

This prefix command is only available in an FM/DB2 LOB editor session.

**SS**
Save blocks of rows on the clipboard. Mark start and end of block.

**UC**
Convert all lowercase characters in a row to uppercase.

**Note:** The UC (UCn, UCC) command affects all characters in a row, not just characters in columns with a CHAR or VARCHAR data type. Therefore, numeric data, such as binary data or packed decimal data, can be corrupted when you use these commands.
<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>UCn</td>
<td>Convert all lowercase characters in n rows to uppercase. (See note for UC.)</td>
</tr>
<tr>
<td>UCC</td>
<td>Convert all lowercase characters in a block of rows to uppercase. Mark start and end of block. (See note for UC.)</td>
</tr>
<tr>
<td>X</td>
<td>Exclude row from display. If the display of excluded row shadow lines is enabled (see “SHADOW primary command” on page 775), a shadow line is shown indicating how many rows are being excluded at this position. To display an excluded row, use the F or LA prefix command.</td>
</tr>
<tr>
<td>Xn</td>
<td>Exclude n rows from display. If the display of excluded row shadow lines is enabled (see “SHADOW primary command” on page 775), a shadow line is shown indicating how many rows are being excluded at this position.</td>
</tr>
<tr>
<td>XX</td>
<td>Exclude block of rows from display. Mark start and end of block. If the display of excluded row shadow lines is enabled (see “SHADOW primary command” on page 775), a shadow line is shown indicating how many rows are being excluded at this position.</td>
</tr>
<tr>
<td>xxxx</td>
<td>Label a row. The label, xxxx, is a string of 1 to 4 alphabetic characters, that does not start with the character “Z” (labels starting with “Z” are system labels). Labels can be assigned to any row. Labels cannot be assigned to shadow lines.</td>
</tr>
</tbody>
</table>

**Parent panels**
- “DB2 Edit panel” on page 505
- “Column Selection/Edit panel” on page 407

**Child panels**
None.

**Related tasks**
Table Edit (related) panel

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>Table Edit (related)</td>
<td>Parent table</td>
<td>Format TABL</td>
</tr>
<tr>
<td>DPDNT FMUSER.PROJACT NAME FMUSER.ACT</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ACTNO ACTKWD ACTDESC</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>#1 #2 #3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SMINT CH(6) VARCHAR(20)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PJ=+K ===&gt; &lt;--------</td>
<td>--------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>000000 **** Top of data ****</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>000001 10 MANAGE MANAGE/ADVISE&lt;</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>000002 20 ECOST ESTIMATE COST&lt;</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>000003 30 DEFINE DEFINE SPECS&lt;</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>000004 40 LEADPR LEAD PROGRAM/DESIGN&lt;</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>000005 50 SPECS WRITE SPECS&lt;</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>000006 60 LOGIC DESCRIBE LOGIC&lt;</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>000007 70 CODE CODE PROGRAMS&lt;</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>000008 80 TEST TEST PROGRAMS&lt;</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>000009 90 ADMOS ADM QUERY SYSTEM&lt;</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>000010 100 TEACH TEACH CLASSES&lt;</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>000011 110 COURSE DEVELOP COURSES&lt;</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>000012 120 STAFF PERS AND STAFFING&lt;</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>000013 130 OPERAT OPER COMPUTER SYS&lt;</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>000014 140 MAINT MAINT SOFTWARE SYS&lt;</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>000015 150 ADMSYS ADM OPERATING SYS&lt;</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>000016 160 ADMDB ADM DATA BASES&lt;</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>000017 170 ADMDC ADM DATA COMM&lt;</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>000018 180 DOC DOCUMENT&lt;</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>000019 **** End of data ****</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Parent panels

This panel appears when you use the REDIT primary command from within an error information panel (providing the error information panel offers REDIT as an option) when a referential integrity error has occurred.

Child panels

None.

Related tasks

Table Selection panel

You use the Table Selection panel to select a table from a list of tables produced when you have used an asterisk (*) or a pattern when specifying a DB2 object name.
Panel and field definitions

Process Opti ┌────────────────────── Table Selection ──────────────────────┐
FM/DB2 (DB26) │ Row 1 to 9 of 9 │
Specify the DB2 Location .
Owner ... FMDB2 AUX_EMP_RESUME FMDB2DL FMDB2SN AUX TABLE
Name ... - FMDB2 EEMP FMDB2DA FMDB2SR TABLE
Table space - FMDB2 EMP PHOTO_RESUME FMDB2DL FMDB2SB TABLE
Processing Opti Template: - FMDB2 EMPLOYEE FMDB2DA FMDB2SP TABLE
Template use - FMDB2 VEMP TABLE
3 1. Above
2. Previo
3. Genera
4. Genera
Command ===> Command ===> F1=Help F2=Split F3=Exit F7=Backward
F1=Help F8=Forward F9=Swap F12=Cancel

S Selection column. To select a table, type S.
Owner The name of the owner of the table.
Name The name of the table.
Database The name of the database containing the table.
Table Space The name of the table space containing the table.
Type The DB2 object type.

Parent panels

This panel appears when you enter an asterisk (*) or pattern in one or more of the following fields when specifying a DB2 object name: Owner, Name, Database, Table space.

Child panels

None.

Related tasks

- [“Specifying a DB2 object name” on page 25](#)

Table Space Parts panel

FM/DB2 displays the Table Space Parts panel if you use the PT line command against an item listed on the [“Table Spaces panel” on page 698](#).
Table Spaces panel

You use the Table Spaces panel to list table space object types in the DB2 catalog.

Panel and field definitions

The columns that are displayed include the SEL field and columns of SYSIBM.SYSTABLESPACE.

Note:
1. The system option, Show all catalog table columns, affects which columns FM/DB2 displays when you use the Object List utility.
2. You can also customize which columns FM/DB2 displays when you use the Object List utility. For details, see the File Manager Customization Guide.

To view other columns on this panel, press the Right function key (F11).

For a complete list of columns, refer to “DB2 catalog tables” in the DB2 for z/OS SQL Reference relevant to your version of DB2.

SEL Line command area.

For a list of the line commands you can use on this panel, see Table 22 on page 250.

Parent panels
- “Object List Utility panel” on page 610
- “Databases panel” on page 500
- “Tables, Views and Aliases panel” on page 704
- “Application Packages panel” on page 387
## Child panels

<table>
<thead>
<tr>
<th>Panel</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>“Alter Table Space panel” on page 387</td>
<td>Line command A</td>
</tr>
<tr>
<td>“Create Table Space panel” on page 462</td>
<td>Line command CR</td>
</tr>
<tr>
<td>“Create Table panel” on page 459</td>
<td>Line command CRT</td>
</tr>
<tr>
<td>“Databases panel” on page 500</td>
<td>Line command D</td>
</tr>
<tr>
<td>See “Drop panels” on page 524</td>
<td>Line command DR</td>
</tr>
<tr>
<td>“Database Structure panel” on page 500</td>
<td>Line command DS</td>
</tr>
<tr>
<td>“Grant privileges panels” on page 587</td>
<td>Line command G</td>
</tr>
<tr>
<td>“Generate SQL From DB2 Catalog panel” on page 585</td>
<td>Line command GEN</td>
</tr>
<tr>
<td>“Details panels” on page 522</td>
<td>Line command I</td>
</tr>
<tr>
<td>“Recovery Information panel” on page 638</td>
<td>Line command ICS</td>
</tr>
<tr>
<td>“Privileges panels” on page 624</td>
<td>Line command P</td>
</tr>
<tr>
<td>“Table Space Parts panel” on page 697</td>
<td>Line command PT</td>
</tr>
<tr>
<td>“Revoke privileges panels” on page 647</td>
<td>Line command R</td>
</tr>
<tr>
<td>“Display Row panel” on page 523</td>
<td>Line command ROW</td>
</tr>
<tr>
<td>“Tables, Views and Aliases panel” on page 704</td>
<td>Line command T</td>
</tr>
<tr>
<td>“DB2 Utilities panel” on page 515</td>
<td>Line command UTL</td>
</tr>
<tr>
<td>“Sort Fields panel” on page 679</td>
<td>Primary command SORT</td>
</tr>
</tbody>
</table>

### Related tasks
- “Working with object list panels” on page 244
- “Using the line command area (Cmd)” on page 250

### Related references
- “SORT primary command” on page 777
Prefix area

In TABL format mode, you can enter the following commands in the prefix area of the data display:

A  Identifies the row after which data is to be moved or copied.
B  Identifies the row before which data is to be moved or copied.
C  Copy one row.
Cn  Copy n rows.
CC  Copy block of rows. Mark start and end of block.
D  Delete one row.
Dn  Delete n rows.
DD  Delete block of rows. Mark start and end of block.
E  Display a panel containing information about why a change to this row failed. The E command is only valid when a row is marked in error. Rows marked in error have "=ERR", "ERR R", or the SQLCODE in the prefix field.
F  Display the first row of a block of excluded rows.
Fn  Display the first n rows of a block of excluded rows.
FS  Change the display format to SNGL and display the row.
G  Get the data stored on the clipboard. Rows saved on the clipboard with the S, Sn or SS prefix commands can only be retrieved from the clipboard with the G prefix command during the same Edit session. The clipboard is a temporary memory buffer that is cleared when you exit from the DB2 Edit function panel (and FM/DB2 returns you to the Primary Option Menu).
Insert one row. The row is initialized using the row description of the currently displayed row. For each column, if it has a default value (DEFAULT="value", and not just the DEFAULT keyword), then it is initialized to that value. Otherwise, the column is initialized according to its data type:

**Character strings**
Set to blanks

**Numeric**
Set to zeros

**Row identifier**
Set to blanks

**DATE**
Set to the current date

**TIME**
Set to the current time

**TIMESTAMP**
Set to the current date and time

**Note:** If a column of the inserted row can contain a null value, it is set to the null value and the null column display indicator is displayed in the displayed column.

For more information about setting the value of the null column display indicator, see "View and Edit options (options 1 and 2)" on page 43.

**Related tasks**
- "View and Edit options (options 1 and 2)" on page 43

**Related references**
- Chapter 14, “DB2 data types,” on page 361

**Insert**
Insert n empty rows.

**IP**
Insert a primary key. In an Edit session of a related parent table only, insert a row with the key columns set to the key value that originally caused the error in the dependent table of the relationship.

**IPn**
Insert n primary keys. In an Edit session of a related parent table only, insert n rows with the key columns set to the key value that originally caused the error in the dependent table of the relationship. If you attempt to save the data, it results in duplicate key errors for the second and all subsequent inserted rows, as primary key values must be unique.

**L**
Display the last row of a block of excluded rows.

**Ln**
Display the last n rows of a block of excluded rows.

**LC**
Convert all uppercase characters in a row to lowercase.

**Note:** The LC (LCn, LCC) command affects all characters in a row, not just characters in columns with a CHAR or VARCHAR data type. Therefore, numeric data, such as binary data or packed decimal data, can be corrupted when you use these commands.

**LCn**
Convert all uppercase characters in n rows to lowercase. (See note for LC.)

Chapter 15. FM/DB2 panels and fields  701
### Table View panel

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>LCC</strong></td>
<td>Convert all uppercase characters in a block of rows to lowercase. Mark start and end of block. (See note for LC.)</td>
</tr>
<tr>
<td><strong>M</strong></td>
<td>Move one row.</td>
</tr>
<tr>
<td><strong>Mn</strong></td>
<td>Move n rows.</td>
</tr>
<tr>
<td><strong>MM</strong></td>
<td>Move block of rows. Mark start and end of the block.</td>
</tr>
<tr>
<td><strong>P</strong></td>
<td>Identifies the row preceding which data is to be moved or copied.</td>
</tr>
<tr>
<td><strong>R or &quot;</strong></td>
<td>Repeat one row.</td>
</tr>
<tr>
<td><strong>Rn or &quot;&quot;n</strong></td>
<td>Repeat n rows.</td>
</tr>
<tr>
<td><strong>RE</strong></td>
<td>Edit a related table. You can use the RE prefix command to edit the related table when the row is marked with an error caused by a referential constraint. You can only enter the RE prefix command against a row marked in error, where the rightmost position of the prefix area contains an “R”. RE starts another FM/DB2 dependent edit session, in a new ISPF logical session. The new (dependent) FM/DB2 Edit session is for the related table in the relationship that caused the referential constraint error. For more details, see “Editing related tables” on page 149. <strong>Note:</strong> When you use the RE prefix command for a row whose prefix area contains “=ERR R”, take care to avoid the editor interpreting the RE command as an R prefix command instead by ensuring the “RE” is followed by one or more blanks. The RE command lets you quickly identify and resolve referential constraint errors. After you use the RE command to edit a related table, you cannot save data until you end the dependent Edit session of the related table. This restriction also applies to tables that are part of a larger referential structure. For example, say A, B and C are related tables and A and B are part of a relationship in which A is the parent table, and B and C are part of a relationship in which B is the parent table. If you edit table A, then use the RE command to edit related table B, then use the RE command in the Edit session of table B to edit table C, it would result in 3 FM/DB2 Edit sessions. The Edit sessions for tables B and C are both dependent edit sessions of the Edit session for table A. You cannot save data in the Edit session for table A until you end the Edit sessions for tables B and C.</td>
</tr>
<tr>
<td><strong>RR or &quot;&quot;</strong></td>
<td>Repeat block of rows. Mark start and end of the block.</td>
</tr>
<tr>
<td><strong>RRn or &quot;&quot;n</strong></td>
<td>Repeat block of rows n times. Mark start and end of block.</td>
</tr>
<tr>
<td><strong>S</strong></td>
<td>Save row on clipboard. The clipboard is a temporary memory buffer that is cleared when you exit from the DB2 Edit function panel (and FM/DB2 returns you to the Primary Option Menu).</td>
</tr>
<tr>
<td><strong>Sn</strong></td>
<td>Save n rows on the clipboard.</td>
</tr>
<tr>
<td><strong>SS</strong></td>
<td>Save blocks of rows on the clipboard. Mark start and end of block.</td>
</tr>
</tbody>
</table>
Table View panel

UC Convert all lowercase characters in a row to uppercase.

Note: The UC (UCn, UCC) command affects all characters in a row, not just characters in columns with a CHAR or VARCHAR data type. Therefore, numeric data, such as binary data or packed decimal data, can be corrupted when you use these commands.

UCn Convert all lowercase characters in n rows to uppercase. (See note for UC.)

UCC Convert all lowercase characters in a block of rows to uppercase. Mark start and end of block. (See note for UC.)

X Exclude row from display. If the display of excluded row shadow lines is enabled (see “SHADOW primary command” on page 775), a shadow line is shown indicating how many rows are being excluded at this position. To display an excluded row, use the F or LA prefix command.

Xn Exclude n rows from display. If the display of excluded row shadow lines is enabled (see “SHADOW primary command” on page 775), a shadow line is shown indicating how many rows are being excluded at this position.

XX Exclude block of rows from display. Mark start and end of block. If the display of excluded row shadow lines is enabled (see “SHADOW primary command” on page 775), a shadow line is shown indicating how many rows are being excluded at this position.

.XXX Label a row. The label, xxx, is a string of 1 to 4 alphabetic characters, that does not start with the character “Z” (labels starting with “Z” are system labels). Labels can be assigned to any row. Labels cannot be assigned to shadow lines.

Parent panels
- “DB2 View panel” on page 518
- “Column Selection/Edit panel” on page 407

Child panels
None.

Related tasks

Table/View/Alias Selection panel

You use the Table/View/Alias Selection panel to select a table, view, or alias from a list produced when you have used an asterisk (*) or a pattern when specifying a DB2 object name.
Table/View/Alias Selection panel

Panel and field definitions

S  Selection column. To select a table, type S.

Owner  
The name of the owner of the table.

Table/View/Alias Name  
The name of the table, view, or alias.

Database  
The name of the database containing the table.

Table Space  
The name of the table space containing the table.

Type  
The DB2 object type.

Parent panels

This panel appears when you enter an asterisk (*) or pattern in one or more of the following fields when specifying a DB2 object name: Owner, Name, Database, Table space.

Child panels

None.

Related tasks

- "Specifying a DB2 object name" on page 25

Tables, Views and Aliases panel

You use the Tables, Views and Aliases panel to list table, view, and alias object types in the DB2 catalog.
Panel and field definitions

The columns that are displayed include the SEL field and columns of SYSIBM.SYSTABLES.

---

SEL | TABLE | TABLE DATABASE OBJECT +
---- | ------ | ------ | ------ | ------ | ------ |
* | * | * | * | * |
--- | ----- | ----- | ----- | ----- | ----- |
DMYAPP | TABLES | DSN0B06 | View |
DMYAPP | TABLESPACE | DSN0B06 | View |
DMYAPP | VIEWS | DSN0B06 | View |
DSN81010 | EMP_COPY | DSN00245 | Table |
DSN81010 | ACT | DSN8D10A | Table |
DSN81010 | CATALOG | DSN8D10X | Table |
DSN81010 | CUSTOMER | DSN8D10X | Table |
DSN81010 | DEPT | DSN8D10A | Table |
DSN81010 | EACT | DSN8D10A | Table |
DSN81010 | EDEPT | DSN8D10A | Table |
DSN81010 | EEMP | DSN8D10A | Table |

Command => Scroll PAGE
F1=Help F2=Zoom F3=Exit F4=Retrieve F5=Find F6=Change
F7=Up F8=Down F9=Swap F10=Left F11=Right F12=Cancel

Note:
1. The system option, Show all catalog table columns, affects which columns FM/DB2 displays when you use the Object List utility.
2. You can also customize which columns FM/DB2 displays when you use the Object List utility. For details, see the File Manager Customization Guide.

To view other columns on this panel, press the Right function key (F11).

For a complete list of columns, refer to “DB2 catalog tables” in the DB2 for z/OS SQL Reference relevant to your version of DB2.

SEL | Line command area.

For a list of the line commands you can use on this panel, see Table 22 on page 250.

Parent panels

This panel appears when you specify the T line command against an object displayed in a list of DB2 objects.

Child panels

To display this panel... Use/do this
“Alter Table panel” on page 375 Line command A
“Alter Table - ADD CHECK Constraint panel” on page 376 Line command ADD
“Tables, Views and Aliases panel” on page 704 Line command ALS
“Tables, Views and Aliases panel” on page 704 Line command AT
“DB2 Browse panel” on page 501 Line command B
### Tables, Views and Aliases panel

<table>
<thead>
<tr>
<th>To display this panel...</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>“Tables, Views and Aliases panel” on page 704</td>
<td>Line command BT</td>
</tr>
<tr>
<td>“Copy Utility (“From”) panel” on page 418</td>
<td>Line command C</td>
</tr>
<tr>
<td>“Column Distribution panel” on page 407</td>
<td>Line command CDI</td>
</tr>
<tr>
<td>“Tables, Views and Aliases panel” on page 704</td>
<td>Line command CH</td>
</tr>
<tr>
<td>“Children of Table panel” on page 396</td>
<td>Line command CHR</td>
</tr>
<tr>
<td>“Table Check Constraints panel” on page 697</td>
<td>Line command CK</td>
</tr>
<tr>
<td>“Columns in Table panel” on page 413</td>
<td>Line command COL</td>
</tr>
<tr>
<td>See “Comment panels” on page 413</td>
<td>Line command COM</td>
</tr>
<tr>
<td>“Create Table panel” on page 459</td>
<td>Line command CR</td>
</tr>
<tr>
<td>“Create Alias panel” on page 430</td>
<td>Line command CRA</td>
</tr>
<tr>
<td>“Create Index panel” on page 442</td>
<td>Line command CRX</td>
</tr>
<tr>
<td>“Columns in Table panel” on page 413</td>
<td>Line command CS</td>
</tr>
<tr>
<td>“Databases panel” on page 500</td>
<td>Line command D</td>
</tr>
<tr>
<td>See “Drop panels” on page 524</td>
<td>Line command DR</td>
</tr>
<tr>
<td>“DB2 Edit panel” on page 505</td>
<td>Line command E</td>
</tr>
<tr>
<td>“Foreign Keys for Table panel” on page 583</td>
<td>Line command FK</td>
</tr>
<tr>
<td>“Grant privileges panels” on page 587</td>
<td>Line command G</td>
</tr>
<tr>
<td>“Generate SQL From DB2 Catalog panel” on page 585</td>
<td>Line command GEN</td>
</tr>
<tr>
<td>“Details panels” on page 522</td>
<td>Line command I</td>
</tr>
<tr>
<td>“Label Table panel” on page 599</td>
<td>Line command LAB</td>
</tr>
<tr>
<td>“Privileges panels” on page 624</td>
<td>Line command P</td>
</tr>
<tr>
<td>“Tables, Views and Aliases panel” on page 704</td>
<td>Line command PA</td>
</tr>
<tr>
<td>“Parents of Table panel” on page 614</td>
<td>Line command PAR</td>
</tr>
<tr>
<td>“Primary Key for Table panel” on page 617</td>
<td>Line command PK</td>
</tr>
<tr>
<td>“Application Packages panel” on page 387</td>
<td>Line command PKG</td>
</tr>
<tr>
<td>“Application Plans panel” on page 389</td>
<td>Line command PL</td>
</tr>
<tr>
<td>“Print Utility panel” on page 621</td>
<td>Line command PR</td>
</tr>
<tr>
<td>“Revoke privileges panels” on page 647</td>
<td>Line command R</td>
</tr>
<tr>
<td>“Display Row panel” on page 523</td>
<td>Line command ROW</td>
</tr>
<tr>
<td>“Table Spaces panel” on page 698</td>
<td>Line command S</td>
</tr>
<tr>
<td>“Basic SELECT Prototyping panel” on page 391</td>
<td>Line command SEL</td>
</tr>
<tr>
<td>“Synonyms panel” on page 687</td>
<td>Line command SYN</td>
</tr>
<tr>
<td>“Tables, Views and Aliases panel” on page 704</td>
<td>Line command T</td>
</tr>
<tr>
<td>“DB2 Utilities panel” on page 515</td>
<td>Line command UTL</td>
</tr>
<tr>
<td>“DB2 View panel” on page 515</td>
<td>Line command V</td>
</tr>
<tr>
<td>“Create View Source Statements panel” on page 495</td>
<td>Line command VS</td>
</tr>
<tr>
<td>“Indexes panel” on page 595</td>
<td>Line command X</td>
</tr>
<tr>
<td>“Indexes and Columns for panel” on page 597</td>
<td>Line command XC</td>
</tr>
<tr>
<td>“Sort Fields panel” on page 679</td>
<td>Primary command SORT</td>
</tr>
</tbody>
</table>
Related tasks

- “Working with object list panels” on page 244
- “Using the line command area (Cmd)” on page 250

Related references

- “SORT primary command” on page 777

TEMPLATE Options (1 of 2) panel

You use the TEMPLATE Options (1 of 2) panel to enter information used by FM/DB2 to generate TEMPLATE statements and the proper JCL statement to access the TEMPLATE library in a utility batch job.

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>TEMPLATE Options (1 of 2)</td>
<td>Global Settings</td>
<td></td>
</tr>
</tbody>
</table>

Enter RESET to restore installation defaults.

**TEMPLATE Library Option:**

- **JCL DD statement:** No DD stmt is generated if field is blank
  ```
  //SYSTEMPL DD
  // DD
  // DD
  // DD
  ```

**Initial Utility Statement Option:**

- **Utility statement:** No statement is generated if field is blank
  ```
  TEMPLATE
  ```

**Additional statements:**

- Enter "/" to select options
  - Use additional statements
  - Edit additional statements

Command ===>

<table>
<thead>
<tr>
<th>F1=Help</th>
<th>F2=Split</th>
<th>F3=Exit</th>
<th>F4=CMetrieve</th>
<th>F5=Reset</th>
<th>F7=Backward</th>
</tr>
</thead>
<tbody>
<tr>
<td>F8=Forward</td>
<td>F9=Swap</td>
<td>F11=NxtPage</td>
<td>F12=Cancel</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

JCL DD statement

The information used to generate the JCL DD statement. This information is:

**DD name**

A read-only field. This holds the value that was specified in the `TEMPLATEDD DD name` field on the OPTIONS Options panel. If the library field is blank then no DD statement is generated in the utility batch job.

**Text values**

The values entered on these four lines are used directly to generate the DD statement, provided that the `DD name` field is not blank.

**Utility statement**

The field of the Utility statement is used to generate the initial utility statements. If the field is blank, and the `Use additional statements` option is not selected, then no TEMPLATE statement is generated into the utility batch job. If the input field is non-blank, then the specified TEMPLATE is
generated with any additional TEMPLATE statements supplied when the
Use additional statements option is selected.

Use additional statements
If this option is set then more than one TEMPLATE statement may be
generated into the utility batch job. Select the Edit additional statements
option to edit additional statements. Only TEMPLATE specifications that
have non-blank input fields are used in the generation.

If this option is not selected, then only a non-blank value in the Utility
statement field is used to generate the TEMPLATE statement.

Edit additional statements
When you select this option and press Enter, the Additional TEMPLATE
Statements panel ("Additional TEMPLATE Statements panel" on page 368)
is displayed. After you have entered values on this panel and exited the
panel, this option is deselected. The additional statements are used when
you select the Use additional statements option.

The TEMPLATE Library Option panel has an installation-defined default. You can
restore the default for the library option by entering the RESET command or
pressing the RESET PF key. Each subsystem has its own installation-defined
defaults. RESET also clears the Use additional statements option and any values
in the Utility statement fields.

For more information, see TEMPLATE and OPTIONS statements in DB2 Utility
Guide and Reference.

Parent panels

Child panels
To display this panel... Use/do this
"Additional LISTDEF Statements panel" on page 367 Select the Edit additional
statements option.
"TEMPLATE Options (2 of 2) panel" Press F11

Related tasks
- "RESET primary command" on page 771
- "Setting options for DB2 utilities" on page 278
- "DB2 Utility TEMPLATE options (option 3.UT)" on page 45

TEMPLATE Options (2 of 2) panel

You use the TEMPLATE Options (2 of 2) panel to enter more information used by
FM/DB2 to generate TEMPLATE statements and the proper JCL statement to
access the TEMPLATE library in a utility batch job.
Panel and field definitions

COPYDDN template name
A new FM/DB2 default setting for a COPYDDN clause. The original default is the DD-name SYSCOPY.

INDDN template name
A new FM/DB2 default setting for an INDDN clause. The original default is the DD-name SYSREC00.

WORKDDN template name
A new FM/DB2 default setting for a WORKDDN clause. The original default is SYSUT1.

UNLDDN template name
A new FM/DB2 default setting for a UNLDDN clause.

The UNLDDN clause is only generated if the Unload External option of the REORG Utility (Table Spaces) panel \footnotesize{REORG Utility (Table Spaces) panel\footnotesize{}} on page 644} is set to “Y”.

The original default is either the DD-name SYSREC00 or SYSREC. It depends on a number of factors including what options are chosen.

PUNCHDDN template name
A new FM/DB2 default setting for the PUNCHDDN clause. The original default is the DD-name SYSPUNCH.

UNLDDN template name
A new FM/DB2 default setting for the UNLDDN clause. The original default is the DD-name SYREC00.

Parent panels
• "TEMPLATE Options (1 of 2) panel" on page 707

Child panels
None.
Related tasks

- "RESET primary command" on page 771
- "Setting options for DB2 utilities" on page 278
- "COPY utility" on page 283
- "LOAD utility" on page 286
- "Loading data described by a copybook or template" on page 289
- "REBUILD utility (indexes)" on page 292
- "REORG (table spaces)" on page 298
- "UNLOAD utility" on page 306
- "UNLOAD (tables)" on page 306
- "UNLOAD (table spaces)" on page 312
- "DB2 Utility TEMPLATE options (option 3.UT)" on page 45

Template Mapping panel

You use the Template Mapping panel to specify the columns in the source (the “From” columns) you want transferred to the columns in the target (the “To” columns) when you are moving (copying) data.

Panel and field definitions

```
<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>Template Mapping</td>
<td>Line 1 of 16</td>
<td></td>
</tr>
<tr>
<td>To</td>
<td>FMNUSER.EEMP</td>
<td>From</td>
<td>FMNUSER.EEMP</td>
</tr>
<tr>
<td>Cmd</td>
<td>CL#</td>
<td>To Column name</td>
<td>Datatype</td>
</tr>
<tr>
<td>***** Top of data *****</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>EMPNO</td>
<td>CH(6)</td>
<td>NN</td>
</tr>
<tr>
<td>2</td>
<td>FIRSTNME</td>
<td>VC(12)</td>
<td>NN</td>
</tr>
<tr>
<td>3</td>
<td>MIDINIT</td>
<td>CH(1)</td>
<td>NN</td>
</tr>
<tr>
<td>4</td>
<td>LASTNAME</td>
<td>VC(15)</td>
<td>NN</td>
</tr>
<tr>
<td>5</td>
<td>WORKDEPT</td>
<td>CH(3)</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>PHONENO</td>
<td>CH(4)</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>HIREDATE</td>
<td>DATE</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>JOB</td>
<td>CH(8)</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>EDLEVEL</td>
<td>SMINT</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>SEX</td>
<td>CH(1)</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>BIRTHDATE</td>
<td>DATE</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>SALARY</td>
<td>DEC(9,2)</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>BONUS</td>
<td>DEC(9,2)</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>COMM</td>
<td>DEC(9,2)</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>RID</td>
<td>CH(4)</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>TSTAMP</td>
<td>TIMESTAMP</td>
<td></td>
</tr>
<tr>
<td>***** End of data *****</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

TO
The “To” DB2 object. The DB2 object to which data will be imported.

FROM
The “From” data set. The name of the import data set.

Cmd
Prefix command area, used to enter a template editor prefix command. S or E to edit the column mapping.

CL# (TO)
The “To” column number. The number of the column in the “To” template.
### Template Mapping panel

**Column name (TO)**
The “To” column name. The name of the column in the “To” template.

**Datatype (TO)**
The “To” data type. The data type and, optionally, the length of the column in the “To” template.

**NI (TO)**
The “To” Null Indicator. The null indicator for the column in the “To” template:
- **(blank)**: The column allows null values
- **NN**: The column does not allow null values.

**Cl# (FROM)**
The “From” column number. The number of the column in the “From” template.

**Column name (FROM)**
The “From” column name. The name of the column in the “From” template.

**Datatype (FROM)**
The “From” data type. The data type and, optionally, the length of the column in the “From” template.

**NI (FROM)**
The “From” Null Indicator. The null indicator for the column in the “From” template:
- **(blank)**: The column allows null values
- **NN**: The column does not allow null values.

### Parent panels
- "Copy Utility (“To”) panel" on page 424
- "Import Utility (“To”) panel" on page 592
- "Export “To” panel" on page 564
- "LOAD from panel" on page 601

### Child panels

<table>
<thead>
<tr>
<th>To display this panel...</th>
<th>Use/do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;From Column Mapping panel“ on page 583</td>
<td>Enter “S” in Cmd field</td>
</tr>
</tbody>
</table>

### Related tasks
- "Specifying your own mapping“ on page 81
- "Copying data from one DB2 object to another“ on page 193
- "Copying data from a VSAM or QSAM file“ on page 198
- "Copying data to a VSAM or QSAM file“ on page 221
- "LOAD utility“ on page 286
Template Save panel

FM/DB2 displays the Template Save panel if you press F3 when the Column Selection/Edit panel is displayed.

Triggers panel

You use the Triggers panel to display information about the triggers you have selected.

Panel and field definitions

The columns that are displayed include the SEL field and columns of SYSIBM.SYSTRIGGERS.

Note:
1. The system option, Show all catalog table columns, affects which columns FM/DB2 displays when you use the Object List utility.
2. You can also customize which columns FM/DB2 displays when you use the Object List utility. For details, see the File Manager Customization Guide.

To view other columns on this panel, press the Right function key (F11).

For a complete list of columns, refer to “DB2 catalog tables” in the DB2 for z/OS SQL Reference relevant to your version of DB2.

SEL Line command area.

For a list of the line commands you can use on this panel, see Table 22 on page 250.

Parent panels
- “Object List Utility panel” on page 610
- “Schemas panel” on page 657
UNLOAD Options panel

You use the UNLOAD Options panel to enter information that is used in generating an UNLOAD statement in a utility batch job.

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>UNLOAD Options</td>
<td>Global Settings</td>
<td></td>
</tr>
</tbody>
</table>

**Utility Options:**
- Enter "/" to select option
- Allow data changes
- Use IEEE floating point
- Allow substitution chars
- Allow padding

**Isolation Level:**
- Isolation Level
- 1. Cursor stability
- 2. Uncommitted rows

**Number of Records Value:**
- Maximum in Error
- 2

**Character Representation Options:**
- Encoding Scheme
- 1. Preserve encoding scheme
- 2. EBCDIC
- 3. ASCII
- 4. UNICODE

**Encoding Scheme Values:**
- SBCS data...
- Mixed DBCS data...
- DBCS data...

**Command:**
- F1=Help
- F2=Split
- F3=Exit
- F4=Retrieve
- F5=Backward
- F6=Forward
- F7=Swap
- F8=Actions
- F9=Cancel
Allow data changes
When this option is selected, you must supply a value for the Isolation Level.

If not selected, FM/DB2 adds the SHRLEVEL REFERENCE clause to the UNLOAD utility control statements, meaning that during the unload operation, rows of the tables can be read, but cannot be inserted, updated, nor deleted by other DB2 threads.

Isolation Level
This parameter comes into effect when the Allow data changes parameter is selected. Valid values are:

1 (Cursor stability)
FM/DB2 omits a SHRLEVEL clause to the UNLOAD utility statement. This is equivalent to adding a SHRLEVEL CHANGE ISOLATION CS clause to the UNLOAD utility control statement, meaning that rows can be read, inserted, updated, and deleted from the table space or partition while the data is being unloaded and that the UNLOAD utility reads rows in cursor stability mode. This is the default value.

2 (Uncommitted rows)
FM/DB2 adds a SHRLEVEL CHANGE ISOLATION UR clause to the UNLOAD utility control statement. This means that rows can be read, inserted, updated, and deleted from the table space or partition while the data is being unloaded and that uncommitted rows, if they exist, are unloaded.

Use IEEE floating point
When this option is selected, FM/DB2 adds the FLOAT IEEE clause to the UNLOAD utility control statements.

If not selected, no clause is added; this means that the DB2 UNLOAD utility output format of the numeric floating point numbers are produced in S/390® hexadecimal Floating Point (HFP) format.

Allow substitution chars
When this option is not selected, FM/DB2 adds the NOSUBS clause to the UNLOAD utility control statement. The default is not selected. This means that substitution characters are not allowed in strings. If character substitution is attempted while unloading data, it is treated as a conversion error.

A substitution character is sometimes placed in a string when that string is being converted from ASCII to EBCDIC, or converted from one CCSID to another. For example, substitution occurs when a character (sometimes referred to as a code point) that exists in the source CCSID (code page) does not exist in the target CCSID (code page).

Allow padding
When this option is selected, the NOPAD clause is not added to the UNLOAD utility control statements.

If not selected, FM/DB2 adds the NOPAD clause to the UNLOAD utility control statements.

The DB2 default for UNLOAD processing is to pad variable length columns in the unloaded records to their maximum length, and the unloaded records have the same length for each table. Select the item if you wish to use the DB2 defined default behavior.
UNLOAD Options panel

When padding is performed:
- The padded data fields are preceded by the length fields that indicate the size of the actual data without the padding.
- When the output records are reloaded using the LOAD utility, padded data fields are treated as varying length data.

Maximum in Error
This option specifies the value for the maximum number of records in error that are allowed. The unloading process terminates when this value is reached. FM/DB2 adds a MAXERR clause with specified value to the UNLOAD utility statements. The default value is 1.

Encoding Scheme
This option specifies the encoding scheme used by the generated UNLOAD utility control statement for all output data of the character type. Valid values are:

1 (Preserve encoding scheme)
The encoding scheme of the source data is preserved (default).
2 EBCDIC is the encoding scheme for all output data of the character type.
3 ASCII is the encoding scheme for all output data of the character type.
4 UNICODE is the encoding scheme for all output data of the character type.

CCSIDs
This option specifies the encoding scheme used by the generated UNLOAD utility control statement for all output data of the character type. Valid values are:

SBCS data
Specifies the CCSID value for single byte character set (SBCS) data in output records, including data unloaded in the external character formats.

Mixed DBCS data
Specifies the CCSID value for mixed double byte character set (DBCS) data in output records, including data unloaded in the external character formats.

DBCS data
Specifies the CCSID value for DBCS data in output records, including data unloaded in the external character formats.

Zero is the default value for each of the above fields.

With the example settings shown in the figure, the following clauses are added to any generated UNLOAD:

• SHRLEVEL CHANGE ISOLATION UR
• FLOAT IEEE
• NOSUB
• MAXERR 2
• EBCDIC
UNLOAD Utility (Table Spaces) panel

You use the UNLOAD Utility (Table Spaces) panel to generate utility control statements using the UNLOAD TABLESPACE form of the UNLOAD utility.

Panel and field definitions

Parent panels

Child panels

Related tasks

- “UNLOAD utility” on page 306
- “UNLOAD (table spaces)” on page 312
- “Where you can use templates” on page 4
You use the UNLOAD Utility (Table Spaces) with LISTDEF panel to generate utility control statements using the UNLOAD TABLESPACE form of the UNLOAD utility.

Panel and field definitions

**Cmd**
You can enter commands in this field to manipulate the panel rows.
- **I(n)** Insert (n) new row(s)
- **R(n)** Repeat the current row (n times)
- **D(n)** Delete (n) row(s)
- **S(n)** Select (n) row(s)

An asterisk (*) in the PRC column shows that the LISTDEF in the row is selected. The S command toggles the selection of rows; unselected rows become selected and selected rows become unselected.

**LISTDEF Name**
On entry to the panel, this field is set to the value in the LISTDEF name field in the DB2 Utilities panel.

**Parent panels**
- “DB2 Utilities panel” on page 515

**Child panels**
None.

**Related tasks**
- “UNLOAD utility” on page 306
- “UNLOAD (table spaces) with LISTDEF specified” on page 314
- “Where you can use templates” on page 4
UNLOAD Utility (Tables) "From" panel

You use the UNLOAD Utility (Tables) "From" panel to specify the name of the DB2 object that contains the data to be unloaded, the (optional) name of a template data set, and processing options.

Panel and field definitions

<table>
<thead>
<tr>
<th>Process Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>UNLOAD Utility (Tables)</td>
<td></td>
</tr>
</tbody>
</table>

From DB2 Object:
- Location . . . Database . . (optional)
- Owner . . . FMNUSER + Table space (optional)
- Name . . . EMP

Unload Count . . . ALL Number of rows to unload

From Template:
- Data set name . . .
- Member . . .

Processing Options:
- Template usage
  - Enter "/", "A"lways to select option
  - 3 Above
  - 2 Previous
  - 1. Generate from table
  - 4. Generate/Replace

Command ==> F1=Help F2=Split F3=Exit F4=Expand F7=Backward F8=Forward F9=Swap F10=Left F11=Right F12=Cancel

Parent panels
- “DB2 Utilities panel” on page 515

Child panels

To display this panel... Use/do this

“UNLOAD Utility (Tables) "To" panel” Press Enter

Related tasks
- “UNLOAD utility” on page 306
- “UNLOAD (tables)” on page 306
- “Where you can use templates” on page 4

UNLOAD Utility (Tables) "To" panel

You use the UNLOAD Utility (Tables) "To" panel to specify the name of the data set that will contain the data to be unloaded, the (optional) name of a template data set, and processing options.
UNLOAD Utility (Tables) "To" panel

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Unload from FMNUSER.EMP

Output Data:
- Data set name: FRED
- Member: .......
- Volume: .......

To Copybook or Template:
- Data set name: 'FMNUSER.DATA'
- Member: FMNCDATA

Processing Options:
- Template usage
- Disposition
- Enter "/", "A"lways
- View UNLOAD options
- Edit template mapping
- 1. Above
- 2. Old or Reuse
- 2. Mod
- 2. Previous
- 3. Generate from input
- 4. Generate and save
- 5. None

Command ===> F1=Help F2=Split F3=Exit F7=Backward F8=Forward F9=Swap F12=Cancel

Parent panels
- "UNLOAD Utility (Tables) "From" panel" on page 718

Child panels
None.

Related tasks
- "UNLOAD utility" on page 306
- "UNLOAD (tables)" on page 306
- "Specifying a data set and a member name" on page 30
- "Letting FM/DB2 do the mapping" on page 81
- "Specifying your own mapping" on page 81

UNLOAD Utility (Tables) with LISTDEF panel

You use the UNLOAD Utility (Tables) with LISTDEF panel to generate utility control statements using the UNLOAD TABLESPACE form of the UNLOAD utility.
UNLOAD Utility (Tables) with LISTDEF panel

Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>UNLOAD Utility (Tables)</td>
<td>Row 1 to 1 of 1</td>
<td></td>
</tr>
</tbody>
</table>

Cmd
- I(n) Insert (n) new row(s)
- R(n) Repeat the current row (n times)
- D(n) Delete (n) row(s)
- S(n) Select (n) row(s)

An asterisk (*) in the PRC column shows that the LISTDEF in the row is selected. The S command toggles the selection of rows; unselected rows become selected and selected rows become unselected.

LISTDEF Name
On entry to the panel, this field is set to the value in the LISTDEF name field in the DB2 Utilities panel.

Parent panels
- “DB2 Utilities panel” on page 515

Child panels
None.

Related tasks
- “UNLOAD utility” on page 306
- “UNLOAD (tables) with LISTDEF specified” on page 311
- “Where you can use templates” on page 4

Utility Functions panel

You use the Utility Functions panel to select the FM/DB2 utility you want to use.
Panel and field definitions

<table>
<thead>
<tr>
<th>Process</th>
<th>Options</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (DFG2)</td>
<td>Utility Functions</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1 Print  
Print DB2 table or view

2 Objects  
Create and drop DB2 objects

3 Copy  
Copy data within DB2

4 Object List  
Display and process DB2 object lists

5 Privileges  
Manage DB2 privileges

6 Import  
Import sequential or VSAM data into DB2

7 Export  
Export DB2 data to sequential or VSAM data set

8 Create  
Create DB2 test data

9 Utilities  
DB2 utility job generation

10 Audit trail  
Print an audit trail report

11 Print browse  
Browse FM/DB2 print data set

Parent panels

- "Primary Option Menu panel" on page 617
Utility Functions panel

Child panels

To display this panel... Use/do this

“Print Utility panel” on page 621 Option 1
“DB2 Object Functions panel” on page 510 Option 2
“Copy Utility (“From”) panel” on page 418 Option 3
“Object List Utility panel” on page 610 Option 4
“Manage DB2 Privileges panel” on page 609 Option 5
“Import Utility (“From”) panel” on page 591 Option 6
“Export Utility panel” on page 567 Option 7
“Data Create Utility panel” on page 495 Option 8
“DB2 Utilities panel” on page 515 Option 9
“Print Audit Trail panel” on page 618 Option 10

Related tasks

• “Printing the contents of a DB2 table” on page 261
• Chapter 5, “Creating and dropping DB2 objects,” on page 163
• “Copying data from one DB2 object to another” on page 193
• Chapter 8, “Working with lists of DB2 objects,” on page 241
• Chapter 10, “Managing DB2 privileges,” on page 271
• “Copying data from a VSAM or QSAM file” on page 198
• “Copying data to a VSAM or QSAM file” on page 221
• Chapter 6, “Populating a DB2 table with data,” on page 191
• Chapter 11, “Generating batch JCL for DB2 utility jobs,” on page 277
• “Printing an audit trail report” on page 267
• “Looking at the print output from your FM/DB2 session” on page 266
• “Working with object list panels” on page 244
• “Using the line command area (Cmd)” on page 250

Value List Edit panel

This panel allows you to create and edit value lists to be used in field scrambling.
Panel and field definitions

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Identifies the line after which data is to be moved or copied.</td>
</tr>
<tr>
<td>B</td>
<td>Identifies the line before which data is to be moved or copied.</td>
</tr>
<tr>
<td>C</td>
<td>Copy one line.</td>
</tr>
<tr>
<td>Cn</td>
<td>Copy n lines.</td>
</tr>
<tr>
<td>CC</td>
<td>Copy block of lines. Marks the start and end of the block.</td>
</tr>
<tr>
<td>D</td>
<td>Delete one line.</td>
</tr>
<tr>
<td>Dn</td>
<td>Delete n lines.</td>
</tr>
<tr>
<td>DD</td>
<td>Delete block of lines. Marks the start and end of the block.</td>
</tr>
<tr>
<td>I</td>
<td>Insert one empty line.</td>
</tr>
<tr>
<td>In</td>
<td>Insert n empty lines.</td>
</tr>
<tr>
<td>M</td>
<td>Move one line.</td>
</tr>
<tr>
<td>Mn</td>
<td>Move n lines.</td>
</tr>
<tr>
<td>MM</td>
<td>Move block of lines. Marks the start and end of the block.</td>
</tr>
<tr>
<td>R</td>
<td>Repeat line once.</td>
</tr>
<tr>
<td>Rn</td>
<td>Repeat line n times.</td>
</tr>
<tr>
<td>RR</td>
<td>Repeat block of lines. Marks the start and end of the block.</td>
</tr>
<tr>
<td>RRn</td>
<td>Repeat block of lines n times. Marks the start and end of the block.</td>
</tr>
</tbody>
</table>

Value  A value that can be used when field scrambling is in effect.
Value List Edit panel

Parent panels

- “Column Attributes panel (alphanumeric)” on page 397
- “Column Attributes panel (numeric)” on page 402

Related tasks

- “Specifying and editing a value list” on page 80
- “Scrambling data” on page 232

Volumes panel

Note: To view other columns on this panel, press the Right function key (F11).

For a complete list of columns, refer to “DB2 catalog tables” in the DB2 for z/OS SQL Reference relevant to your version of DB2.

SEL Line command area.

For a list of the line commands you can use on this panel, see Table 22 on page 250.

Parent panels

- “Storage Group(s) panel” on page 683

Child panels

To display this panel... Use/do this

“Display Row panel” on page 523 Line command ROW

Related tasks

- “Working with object list panels” on page 244
- “Using the line command area (Cmd)” on page 250
Volumes panel

Related tasks

- "Managing tables used by SQL explain" on page 355
Volumes panel
Chapter 16. FM/DB2 commands

This chapter describes the syntax and parameters for the primary commands that you can use in various FM/DB2 panels.

You enter primary commands on the command line of a panel.

Where the syntax allows you to specify more than one operand, you can use either a blank or a comma as a separator between each operand.

For example, both of the following commands are correct:

```
SORT col_num1 col_num2
SORT col_num1, col_num2
```

### ABOUT primary command

The ABOUT command displays the current FM/DB2 version number and the PTF number of the File Manager DB2 component in a window. The popup also shows the copyright information and any notes from IBM that are shipped with the product.

**Syntax**

```
ABOUT
```

**Availability**

Available on all panels

**Related tasks**

“Checking your FM/DB2 version” on page 9

### ALL primary command

Use the ALL primary command to select all columns from all, or one, of the tables you have specified on the primary Basic SELECT Prototyping panel.

```
ALL
```

**Parameters**

- **Asterisk (*)**
  - Selects all columns from all the specified tables.
- **#n**
  - Selects all columns from the table associated with correlation name #n.
ALL primary command

Availability

“Basic SELECT Prototyping panel” on page 391

Related tasks

“Using basic SQL prototyping” on page 317

BACKWARD primary command

The BACKWARD primary command scrolls backwards (up) through your data.

The amount (number of rows or columns) scrolled is determined by either an optional parameter or, if no parameter is entered, by the amount indicated in the Scroll field.

Syntax

```
>> BACKWARD
    |.scroll_field |
```

```
    
    | CSR    |
```

```
    | DATA   |
```

```
    | HALF   |
```

```
    | Max    |
```

```
    | num_rows |
```

```
    | num_cols |
```

```
    | PAGE    |
```

Notes:

1  num_rows available in TABL display format; num_cols available in SNGL display format.

**scroll_field**

Scroll backwards by the amount indicated in the Scroll field. This is the default value if no parameter is used.

**CSR**

Scroll backwards to the cursor position.

**DATA**

Scroll backwards one row (TABL) or column (SNGL) less than a page of data.

**HALF**

Scroll backwards half a page of data.

**Max**

Scroll to top of the table. This has the same effect as TOP command.

**num_rows**


**num_cols**


**PAGE**

Scroll backwards one page of data.

Availability

- “Table Browse panel” on page 689
- “Table Edit panel” on page 691
BOTTOM primary command

The BOTTOM primary command scrolls to the last page of data.

Syntax

```
Bottom
```

Availability

- "Table Browse panel" on page 689
- "Table Edit panel" on page 691
- "Table View panel" on page 700

Related tasks

- Chapter 4, “Viewing and changing DB2 data,” on page 91
- “Scrolling” on page 125

BROWSE primary command

The BROWSE primary command starts a new FM/DB2 Browse editor session.

Syntax

For editor panels

```
Browse current_object_owner.current_object_name
```

For Primary Option Menu panel

```
B current_object_owner.
new_object_owner.
new_object_name
```

- `current_object_owner`
The name of the owner of the object in the current FM/DB2 editor session.
- `current_object_name`
The name of the object in the current FM/DB2 editor session.
- `new_object_owner`
The name of the owner of the object to be pre-filled in the entry panel for the new FM/DB2 editor session.
- `new_object_name`
The name of the object to be pre-filled in the entry panel for the new FM/DB2 editor session.

Availability

- "Table Browse panel” on page 689
- "Table Edit panel” on page 691
CANCEL primary command

The CANCEL primary command ends the current FM/DB2 editor session without saving any changes.

In an editor session, the CANCEL primary command ends the current FM/DB2 editor session and any dependent related sessions. In Edit only, if you have made changes to your data, you are asked to confirm that you want to cancel your Edit session. Cancellation of an Edit session causes any uncommitted changes to the data (in the current or in a dependent related session) to be lost.

Note that FM/DB2 may issue DB2 commit points during an Edit session, depending on the current setting of the Enter Key Processing and Commit Options options (see “Editor Options (6 of 8) panel” on page 545). If commit points have been issued, CANCEL will only discard changes made since the last commit point.

If you have selected the Edit Commit Option Whenever Enter is pressed, the DB2 table is updated whenever you press Enter after having made a change. CANCEL in this situation does not reverse any changes made before the last time you pressed Enter.

QUIT is a synonym for CANCEL.

Syntax

```
  CANCEL
```

Availability

Available on most FM/DB2 panels

Related tasks

- Chapter 4, “Viewing and changing DB2 data,” on page 91
- Chapter 3, “Working with templates,” on page 49
- Chapter 12, “Working with SQL statements,” on page 317

CAPS primary command

The CAPS command determines whether data in the data area is to be translated to uppercase on input (CAPS ON), or whether it is to be left alone (CAPS OFF). The CAPS command overrides any previously issued CASE command. By default, lowercase data is not translated to uppercase.

Syntax

```
  CAPS
```

**CAPS primary command**

CAPS ON
Translates all data in rows that are changed by overtyping from lowercase to uppercase, and translates replacement data specified in the CHANGE command from lowercase to uppercase.

CAPS ON only affects rows changed after the CAPS ON command is entered.

CAPS OFF
Does not translate changed records to uppercase.

**Availability**
- [“Table Browse panel” on page 689](#)
- [“Table Edit panel” on page 691](#)
- [“Table View panel” on page 700](#)

**Related tasks**
- [“Installation settings that affect CAPS” on page 138](#)
- [Chapter 4, “Viewing and changing DB2 data,” on page 91](#)

---

**CASE primary command**

The CASE command determines whether data in the data area is translated to uppercase on input (CASE UPPER), or whether it is left unchanged (CASE MIXED). The CASE command overrides any previously issued CAPS command. By default, lowercase data is not translated to uppercase.

**Syntax**

CASE MIXED
Does not translate changed or new rows to uppercase.

CASE UPPER
Translates all new rows from lowercase to uppercase, and translates all data in changed rows from lowercase to uppercase, regardless of how the rows are changed.

**Availability**
- [“Table Browse panel” on page 689](#)
- [“Table Edit panel” on page 691](#)
- [“Table View panel” on page 700](#)

**Related tasks**
- [“Installation settings that affect CAPS” on page 138](#)
- [Chapter 4, “Viewing and changing DB2 data,” on page 91](#)
In an FM/DB2 editor session, use the CHANGE command to locate and replace one or more occurrences of a character string or numeric value in the data being edited.

**Note:** If you limit the number of rows loaded into the FM/DB2 editor (by either specifying a row limit or a WHERE clause), the CHANGE command applies only to the data in the FM/DB2 editor session.

To find a string without changing it, use the FIND primary command.

If you have zoomed in on a row, the CHANGE command affects only that row.

If the CHANGE command changes data, FM/DB2 places the cursor at the end (when scanning forward), or start (when scanning backwards), of the changed string or numeric data type column. If necessary, FM/DB2 scrolls to bring the string into view. If the changed data is in an excluded row, the excluded row is displayed.

**Syntax**

```
(1) Change
   CX CHG
   string details column list or range
   .ZFIRST .ZLAST
   label1 label2

string details:
   from_string to_string
   NEXT

   column list or range:
   #ALL
   ( col_num )
   ( col_num_1 -- col_num_2 )
```
Notes:

1. You can specify parameters in any order.

* (asterisk)

When used in place of the search string, uses the search string specified on the previous CHANGE command as the search string.

When used in place of the replacement string, uses the replacement string specified on the previous CHANGE command as the replacement string.

The position of the * is important for CHANGE. If it is positioned before a string, it indicates the previous search argument; that is, it is treated as the first string. If a string is found prior to the *, then it is treated as the second string (the change argument), taking the change argument from the previous CHANGE command. To use both the previous search string and the previous change string, specify CHANGE * *.

from_string

The string you want to search for. The string, which can be up to 100 characters in length, can be:

- A character string not starting or ending with a single quotation mark and not containing any embedded blanks or commas. The case of the string is ignored. Uppercase and lowercase representations of the same character match. For example, the following command changes the strings black, Black, and BLACK:
  CHANGE black white

- A character string enclosed in single quotation marks. The string can contain blanks and commas. The case of the string is ignored. For example, 'Exact string' matches 'exact string'.

- C followed by a character string enclosed in quotation marks (C'Frog'), or a character string enclosed in quotation marks followed by C ('Frog'C). The string can contain blanks and commas. The string must match exactly (including case). For example, C'Exact string' does not match C'exact string'.

- P preceded or followed by a picture string enclosed in single or double quotation marks to describe a type of string to be found rather than the exact characters. It can contain blanks, alphabetic and numeric characters which represent themselves, or any of the special characters listed here, each of which represents a class of characters:
  = Any character.
  @ Alphabetic characters.
  # Numeric characters.
  $ Special characters.

& ntsym;

  Non-blank characters.
  . Invalid characters.
  - Non-numeric characters.
  < Lowercase alphabetics.
  > Uppercase alphabetics.

- X followed by a hexadecimal string enclosed in single quotation marks (X'C1C2').

- A numeric value (only when you limit the search by specifying column numbers, and only when the column being searched is a numeric column).
CHANGE primary command

\textit{to\_string}

The string you want to replace \textit{from\_string}. The string, which can be up to 100 characters in length, can be:

- A character string not starting or ending with a single quotation mark and not containing any embedded blanks or commas. If CAPS ON or CASE UPPER is in effect, then \textit{to\_string} is converted to uppercase.
- A character string enclosed in single quotation marks. The string can contain blanks and commas. If CAPS ON or CASE UPPER is in effect, then \textit{to\_string} is converted to uppercase. The string can be a null string ('').
- C followed by a character string enclosed in quotation marks (C'Frog'), or a character string enclosed in quotation marks followed by C ('Frog'C). The string can contain blanks and commas. Case is respected and retained. The string can be a null string (C'').
- P preceded or followed by a picture string enclosed in single or double quotation marks to describe the change to be made. You can change characters from uppercase to lowercase or from lowercase to uppercase, or leave the field the same using these string special characters.

- \textbf{=} Any character.
- \textbf{<} Lowercase alphabetics.
- \textbf{>} Uppercase alphabetics.

**Examples of picture strings used with the CHANGE command:**

\begin{itemize}
  \item \texttt{c p'<' p>}' #3
    \begin{itemize}
      \item Uppercase characters in column 3.
    \end{itemize}
  \item \texttt{c p'/>' '¬' #2}
    \begin{itemize}
      \item Change any uppercase to "not sign" column 2 (note: "not sign" has no special meaning in a "to" string).
    \end{itemize}
  \item \texttt{c p'<# p'>' #1}
    \begin{itemize}
      \item Change any lowercase, any number to uppercase, the number found in column 1.
    \end{itemize}
\end{itemize}

When this notation is used, numeric, bit and unicode fields (for SNGL and TABL display formats) are excluded from the search process.

- X followed by a hexadecimal string enclosed in single quotation marks (X'C1C2'). Case is respected and retained (the hexadecimal values are used exactly as specified).
- A numeric value.

**NEXT** This is the default setting. Causes the search to begin at the cursor location (if the cursor is within the data portion of the display) or the beginning of the first row displayed, and searches ahead to find the next occurrence of the string. If the next occurrence of the string is in an excluded row and you do not limit the search to non-excluded rows, only the first excluded row containing the string is shown.

**ALL** Causes the search to begin at the top of the data and find and replace all occurrences of the string. If you do not limit the search to non-excluded rows, the string is replaced in all rows (excluded and non-excluded). Any excluded rows affected by the change are redisplayed.

**FIRST** Causes the search to begin at the beginning of the table and search ahead.

**LAST** Causes the search to begin at the end of the table and search backwards.

**PREV** Causes the search to begin at the cursor location (if the cursor is within the
CHANGE primary command

data portion of the display) or the beginning of the first row displayed,
and searches backwards to find the string.

CHARS
Matches the search string anywhere in the data.

PREFIX
Matches the search string wherever it appears as a prefix in the data. To be
a prefix, the matched text must be preceded by a non-alphanumeric
character or be the start of a line or field, and must be followed by an
alphanumeric character.

SUFFIX
Matches the search string wherever it appears as a suffix in the data. To be
a suffix, the matched text must be preceded by an alphanumeric character,
and must be followed by a non-alphanumeric character or be the end of a
line or field.

WORD
Matches the search string wherever it appears as a word in the data. To be
a word, the matched text must be preceded by a non-alphanumeric
character or be the start of a line or field, and must be followed by a
non-alphanumeric character or be the end of a line or field.

#ALL
Each column is searched according to its template attributes.

col_num
The DB2 column number (specified as #n) of a column to be included in
the data search. Multiple column numbers must be separated by a comma
or, if enclosed in brackets, separated by a blank or comma.

col_num_1
The first field reference of a range of fields. It cannot be subscripted. If the
col_num_1 field reference value is less than the lowest displayed field
reference value, the lowest displayed field reference value is used.

col_num_2
The last field reference of a range of fields. It cannot be subscripted. If the
col_num_2 field reference value is greater than the highest displayed field
reference value, the highest displayed field reference value is used.

The col_num_1 and col_num_2 field reference values must be separated by a
hyphen (-). Spaces are permitted between the hyphen and the field
reference values.

If col_num_1 is a higher value than col_num_2, the search process reverses
the operands.

label1
Label identifying the start of a range of rows. The label must start with a
period (.) followed by one to four alphabetic characters (no numeric or
special characters). Labels starting with the letter “Z” indicate an
editor-assigned label.

label2
Label identifying the end of a range of rows. The label must start with a
period (.) followed by one to four alphabetic characters (no numeric or
special characters). Labels starting with the letter “Z” indicate an
editor-assigned label.

EX
Excluded rows only.

NX
Non-excluded rows only.

X
Same as EX.
CHANGE primary command

Availability

- “Table Edit panel” on page 691

Related tasks

- “Changing data with the CHANGE command” on page 140

COL primary command

When you use the SQL prototyping, execution and analysis option (4), you can use the COL primary command:

- On the Plan Table Rows panel to switch panel format to show the collection ID.
- On the Statement Table Rows panel to switch panel format to show the collection ID.

Syntax

```

```

Availability

Available on all panels “Plan Table Rows panel” on page 615

Related tasks

Chapter 12, “Working with SQL statements,” on page 317

CHARPOS primary command

In SNGL display mode only, the CHARPOS command displays, or hides, the “character position range” for long character or graphic columns. The command is used to dynamically turn on or off the “column position range” information.

Syntax

```

```

The CHARPOS command is a toggle command. Each time you enter the command the display status of the “column position range” is reversed.

When displayed, the “character position range” information is displayed below the column name for the second and subsequent rows used to display data for long columns. Here is an example:

<table>
<thead>
<tr>
<th>Column</th>
<th>Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>HIGH2KEY</td>
<td>(this row has data for chars 1-103 of column HIGH2KEY)</td>
</tr>
<tr>
<td>104-206</td>
<td>(this row has data for chars 104-206 of column HIGH2KEY)</td>
</tr>
<tr>
<td>207-309</td>
<td>(this row has data for chars 207-309 of column HIGH2KEY)</td>
</tr>
</tbody>
</table>

Availability

- “Table Browse panel” on page 689
- “Table Edit panel” on page 691
- “Table View panel” on page 700
Related tasks

- "Locating a column” on page 109
- "Locating a row or column in a list of DB2 objects” on page 250

DELETE primary command

The DELETE primary command deletes rows from the data being edited.

Syntax

**TABL display format**

```
DELETE ALL

EX label1 label2
NX (1)
X

EX label1 label2
```

Notes:

1. If none of these parameters (EX, NX, or X) are specified, then the DELETE command applies to both excluded and non-excluded rows.

**SNGL display format**

```
DELETE ALL

Causes the DELETE command to delete all rows (or all excluded or non-excluded rows) within the specified range of rows (or within the entire table, if you do not specify a range).

If you do not specify the ALL parameter, then the DELETE command deletes only the first row (or only the first excluded or non-excluded row) within the specified range.

EX Excluded rows only.
NX Non-excluded rows only.
X Same as EX.

**label1** Label identifying the start of a range of rows. The label must start with a period (.) followed by one to four alphabetic characters (no numeric or special characters). Labels starting with the letter “Z” indicate an editor-assigned label.

**label2** Label identifying the end of a range of rows. The label must start with a period (.) followed by one to four alphabetic characters (no numeric or special characters). Labels starting with the letter “Z” indicate an editor-assigned label.

Availability

- “Table Edit panel” on page 691

Chapter 16. FM/DB2 commands
DELETE primary command

Related tasks
- “Deleting rows” on page 148

DESCRIBE primary command

The DESCRIBE primary command displays the DB2 Template Description panel where you can enter the required description for the template.

Syntax

```
DESCRIBE
```

Availability
- “Column Selection/Edit panel” on page 407

Related tasks
- “Editing a template” on page 57

DESELECT primary command

In advanced SELECT prototyping, if you enter the DESELECT primary command and the cursor is not currently placed on an element in the currently displayed fragment, the currently displayed fragment and its dependent fragments are removed from the syntax, and control returns to the parent of the currently displayed fragment. If the currently displayed fragment is the top fragment, this has the same effect as CANCEL or END.

If you enter the DESELECT primary command with the cursor placed on a fragment field in the currently displayed fragment, the selected fragment and its dependent fragments are removed from the syntax, and control remains in the currently displayed fragment.

If the deselected fragment is the last selected (or only) item in a repeat instance, the entire instance is deleted.

Syntax

```
DESELECT
```

Availability
- “Advanced SELECT Prototyping panel” on page 368

Related tasks
- “Using advanced SQL prototyping” on page 325

DOWN primary command

The DOWN primary command scrolls down (forward) through your data.
DOWN primary command

The amount (number of rows or columns) scrolled is determined by either an optional parameter or, if no parameter is entered, by the amount indicated in the Scroll field.

Syntax

```
DOWN scroll_field
  CSR
  DATA
  HALF
  Max
    num_rows (1)
    num_cols
  PAGE
```

Notes:
1. `num_rows` available in TABL display format; `num_cols` available in SNGL display format.

**scroll_field**
 Scroll down by the amount indicated in the Scroll field. This is the default value if no parameter is used.

**CSR**
 Scroll down to the cursor position.

**DATA**
 Scroll down one row (TABL) or column (SNGL) less than a page of data.

**HALF**
 Scroll down half a page of data.

**Max**
 Scroll to bottom of table. This has the same effect as the BOTTOM command.

**num_rows**

**num_cols**
 Scroll down `num_cols` columns. Valid range: 1–9999. Available in SNGL display format.

**PAGE**
 Scroll down one page of data.

Examples

**DOWN 8** Scrolls down 8 lines.

**DOWN** Scrolls down the number of lines indicated by the Scroll field.

**DOWN M** Scrolls down to the last page of data.

Availability

- "Table Browse panel” on page 689
- "Table Edit panel” on page 691
- "Table View panel” on page 700
- "Enter, Execute and Explain SQL Statements panel” on page 555

Related tasks

- Chapter 4, “Viewing and changing DB2 data,” on page 91
- “Scrolling” on page 125
EDIT primary command

The EDIT primary command starts a new FM/DB2 Edit editor session.

**Syntax**

```
EDIT current_object_owner current_object_name current_object_owner new_object_owner new_object_name
```

- `current_object_owner`: The name of the owner of the object in the current FM/DB2 editor session.
- `current_object_name`: The name of the object in the current FM/DB2 editor session.
- `new_object_owner`: The name of the owner of the object to be pre-filled in the entry panel for the new FM/DB2 editor session.
- `new_object_name`: The name of the object to be pre-filled in the entry panel for the new FM/DB2 editor session.

**Availability**

- “Table Browse panel” on page 689
- “Table Edit panel” on page 691
- “Table View panel” on page 700

**Related tasks**

- Chapter 4, “Viewing and changing DB2 data,” on page 91

END primary command

END is a pseudonym for FILE.

- In View or Browse, the END primary command:
  - Ends the View or Browse editor session.
- In Edit, the END primary command:
  - Saves any uncommitted changes and ends the Edit session.
  - Fails if DB2 reports errors when FM/DB2 attempts to save changes made in the Edit session, or there are dependent edit sessions of related tables.
- When you edit a template, the END primary command:
  - Displays the save panel where you can save the template as another name.

**Syntax**

```
END
```

**DOWN primary command**

- “Entering, executing, and explaining SQL statements” on page 351
EXCLUDE primary command

The EXCLUDE primary command excludes from display certain rows in the data being viewed, browsed, or edited.

You can only use the EXCLUDE primary command in TABL display format.

To redisplay excluded rows, enter the command:

```
RESET
```

### Syntax

```
EXclude \\
string \\
NEXT \\
CHARs \\
column list or range,
```

```
FIRST, \\
PREV, \\
LAST, \\
label range,
```

```
column list or range: \\
#ALL, \\
(col_num), \\
(col_num_1-col_num_2),
```

```
label range: \\
.ZFIRST-.ZLAST, \\
label1-label2,
```

### Notes:

1. You can specify parameters in any order.
EXCLUDE primary command

* (asterisk)
Uses the search string specified on the previous EXCLUDE command as the search string.

string
The search string you want to search for. Rows containing this string, within the limits imposed by the other EXCLUDE command parameters, are excluded. The string can be:

- A character string not starting or ending with a single quotation mark and not containing any embedded blanks or commas. The case of the string is ignored. Uppercase and lowercase representations of the same character match. For example, “Mixed string” matches “MIXED string”.
- A character string enclosed in single quotation marks. The string can contain blanks and commas. The case of the string is ignored. For example, “Exact string” matches “exact string”.
- C followed by a character string enclosed in quotation marks (C’Frog’), or a character string enclosed in quotation marks followed by C (’Frog’C). The string can contain blanks and commas. The string must match exactly (including case). For example, “Exact string” does not match “exact string”.
- P preceded or followed by a picture string enclosed in single or double quotation marks to describe a type of string to be found rather than the exact characters. It can contain blanks, alphabetic and numeric characters which represent themselves, or any of the special characters listed here, each of which represents a class of characters:
  - = Any character.
  - @ Alphabetic characters.
  - # Numeric characters.
  - $ Special characters.
  - & notsym; Non-blank characters.
  - . Invalid characters.
  - - Non-numeric characters.
  - < Lowercase alphabetics.
  - > Uppercase alphabetics.

Examples of picture strings used with the EXCLUDE command:

ex p'.' #4
Exclude invalid character in column 4.

ex p'###'
Exclude 3-digit number.

ex '01'p #1
Exclude label a1,b1,c1, (and so on) in column 1.

ex p'<'
Exclude the next lowercase alphabetic character.

ex p'-' #2
Exclude the next non-blank in column 2.

When this notation is used, numeric, bit and unicode fields (for SNGL and TABL display formats) are excluded from the search process.

- X followed by a hexadecimal string enclosed in single quotation marks For example, X'C1C2'.
- A numeric value (only when you limit the search by specifying column numbers, and only when the column being searched is a numeric column).
EXCLUDE primary command

**NEXT**  This is the default setting. Causes the search to begin at the cursor location (if the cursor is within the data portion of the display) or the beginning of the first row displayed, and searches ahead to find the next occurrence of `string` in a row that is not already excluded.

**ALL**  Causes the search to begin at the top of the data and find all occurrences of `string`. If not specified, the search begins at the cursor location (if the cursor is within the data portion of the display) or the beginning of the first row displayed, and searches ahead to find the next occurrence of `string` in a row that is not already excluded.

The command EXCLUDE ALL excludes all displayed rows.

**FIRST**  Causes the search to begin at the beginning of the table and search ahead.

**LAST**  Causes the search to begin at the end of the table and search backwards.

**PREV**  Causes the search to begin at the cursor location (if the cursor is within the data portion of the display) or the beginning of the first row displayed, and searches backwards to find the string.

**CHARS**  Matches the search string anywhere in the data.

**PREFIX**  Matches the search string wherever it appears as a prefix in the data. To be a prefix, the matched text must be preceded by a non-alphanumeric character or be the start of a line or field, and must be followed by an alphanumeric character.

**SUFFIX**  Matches the search string wherever it appears as a suffix in the data. To be a suffix, the matched text must be preceded by an alphanumeric character, and must be followed by a non-alphanumeric character or be the end of a line or field.

**WORD**  Matches the search string wherever it appears as a word in the data. To be a word, the matched text must be preceded by a non-alphanumeric character or be the start of a line or field, and must be followed by a non-alphanumeric character or be the end of a line or field.

**#ALL**  Each column is searched according to its template attributes.

`col_num`  The DB2 column number (specified as `#n`) of a column to be included in the data search. Multiple column numbers must be separated by a comma or, if enclosed in brackets, separated by a blank or comma.

`col_num_1`  The first field reference of a range of fields. It cannot be subscripted. If the `col_num_1` field reference value is less than the lowest displayed field reference value, the lowest displayed field reference value is used.

`col_num_2`  The last field reference of a range of fields. It cannot be subscripted. If the `col_num_2` field reference value is greater than the highest displayed field reference value, the highest displayed field reference value is used.

The `col_num_1` and `col_num_2` field reference values must be separated by a hyphen (`-`). Spaces are permitted between the hyphen and the field reference values.
EXCLUDE primary command

If `col_num_1` is a higher value than `col_num_2`, the search process reverses the operands.

`label1` Label identifying the start of a range of rows. The label must start with a period (.) followed by one to four alphabetic characters (no numeric or special characters). Labels starting with the letter “Z” indicate an editor-assigned label.

`label2` Label identifying the end of a range of rows. The label must start with a period (.) followed by one to four alphabetic characters (no numeric or special characters). Labels starting with the letter “Z” indicate an editor-assigned label.

Availability

- “Table Browse panel” on page 689
- “Table View panel” on page 700
- “Table Edit panel” on page 691

Related tasks

- “Excluding rows” on page 126

EXECUTE primary command

You use the EXECUTE primary when you are working with SQL statements.

For basic SELECT prototyping, the EXECUTE command executes the SELECT statement.

Note: Before you use the EXECUTE primary command, you must have selected at least one column.

For advanced SELECT prototyping, the EXECUTE primary command executes the prototyped command if it is syntactically complete. FM/DB2 displays a result table similar to the Select Statement Browse panel.

When you use the Enter, Execute and Explain SQL Statements panel, the EXECUTE primary command executes the SQL statements in the statement entry area. The EXECUTE primary command is executed by default when you press Enter and the command line is blank.

Syntax

```sql
>>> EXECUTE
```

Availability

- “Basic SELECT Prototyping panel” on page 391
- “Advanced SELECT Prototyping panel” on page 368
- “Enter, Execute and Explain SQL Statements panel” on page 555

Related tasks

- “Using basic SQL prototyping” on page 317
- “Using advanced SQL prototyping” on page 325
- “Entering, executing, and explaining SQL statements” on page 351
EXPAND primary command

The EXPAND primary command displays a panel for inputting or editing long row selection criteria. You use this command from the Column Selection/Edit panel (you must position the cursor on the Row Selection Criteria input field).

Syntax

```
EXPAND
```

Availability

- “Column Selection/Edit panel” on page 407

Related tasks

- “Selecting rows” on page 58

EXPLAIN primary command

The EXPLAIN primary command explains the SQL statements in the statement entry area. FM/DB2 builds an SQL EXPLAIN statement using the contents of the statement entry area to build the FOR clause. If applicable, FM/DB2 also builds a SET QUERYNO clause using the contents of the Query number option or a generated value.

When you use the EXPLAIN primary command, FM/DB2 shows the information obtained by the generated SQL EXPLAIN statement in the Plan Table Rows panel.

Syntax

```
EXPLAIN
```

Availability

- “Enter, Execute and Explain SQL Statements panel” on page 555

Related tasks

- “Entering, executing, and explaining SQL statements” on page 351

EXPORT primary command

The EXPORT primary command starts a dialog that allows either the result table currently shown in the FM/DB2 editor session, or rows in the current FM/DB2 editor session, to be exported to a data set.

Syntax

```
EXPORT
```

```
NOopt Opt Quiet SQLda DB2 DSNTiaul USER CSV
```

ONline BATch

Chapter 16. FM/DB2 commands 745
**EXPORT primary command**

<table>
<thead>
<tr>
<th>(1)</th>
<th>(2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALL</td>
<td>label1 label2</td>
</tr>
<tr>
<td>NX</td>
<td></td>
</tr>
<tr>
<td>X</td>
<td></td>
</tr>
<tr>
<td>EX</td>
<td></td>
</tr>
</tbody>
</table>

**Notes:**

1. NX, X, EX valid for TABL format only.
2. Label range valid for TABL format only.

**NOOPT**

Overrides the current setting for the **Show export options** option on the Editor Options (4 of 8) panel. The Export Options panel is not displayed.

The data format for the export is determined as follows: The current value for the export data format is used, unless a data format parameter is specified in the EXPORT command. In the latter case, the parameter specified in the EXPORT command overrides the current export data format value.

**OPT**

Overrides the current setting for the **Show export options** option on the Editor Options (4 of 8) panel. The Export Options panel is displayed.

**QUIET**

Overrides the current setting for the **Show export options** option on the Editor Options (4 of 8) panel. The Export Options panel is not displayed. This option also suppresses the display of the Export "To" panel. The current values specified on the Export "To" panel are used to process the EXPORT command.

**SQLDA**

Data is exported in FM/DB2 internal (SQLDA) format. This is equivalent to specifying 1 for the **Data Format** option on the Export Options (1 of 3) panel.

**DB2**

Data is exported in DB2 unload format. This is equivalent to specifying 2 for the **Data Format** option on the Export Options (1 of 3) panel.

**DSNTIAUL**

Data is exported in same format as that used by the DSNTIAUL sample program. This is equivalent to specifying 3 for the **Data Format** option on the Export Options (1 of 3) panel.

**USER**

Data is exported in a user-defined data format. This is equivalent to specifying 4 for the **Data Format** option on the Export Options (1 of 3) panel.

**CSV**

Data is exported in a comma-separated variables (delimited) format. This is equivalent to specifying 5 for the **Data Format** option on the Export Options (1 of 3) panel.

**ONLINE**

The export is performed on-line.

**BATCH**

Generate a JCL deck to execute the export for the DB2 object or SQL SELECT statement in batch mode. You cannot specify BATCH with any of ALL | NX | X | EX.
**EXPORT primary command**

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALL</td>
<td>Copies all rows in the current FM/DB2 editor session to the target data set.</td>
</tr>
<tr>
<td>NX</td>
<td>Copies all non-excluded rows in the current FM/DB2 editor session to the target data set.</td>
</tr>
<tr>
<td>X</td>
<td>Copies all excluded rows in the current FM/DB2 editor session to the target data set.</td>
</tr>
<tr>
<td>EX</td>
<td>Same as X.</td>
</tr>
</tbody>
</table>

**label1** Label identifying the start of a range of rows to be processed. The label must start with a period (.) followed by one to four alphabetic characters (no numeric or special characters). Labels starting with the letter “Z” indicate an editor-assigned label.

**label2** Label identifying the end of a range of rows to be processed. The label must start with a period (.) followed by one to four alphabetic characters (no numeric or special characters). Labels starting with the letter “Z” indicate an editor-assigned label.

**Availability**
- “Table Browse panel” on page 689
- “Table Edit panel” on page 691
- “Table View panel” on page 700

**Related tasks**
- “Excluding rows” on page 126

---

**FILE primary command**

FILE is a pseudonym for END.

- In View or Browse, the FILE primary command:
  - Ends the View or Browse session.
- In Edit, the FILE primary command:
  - Saves any uncommitted changes and ends the Edit session.
  - Fails if DB2 reports errors when FM/DB2 attempts to save changes made in the Edit session, or there are dependent edit sessions of related tables.
- When you edit a template, the FILE primary command:
  - Displays the save panel where you can save the template as another name.

**Syntax**

```
>>>FILE
```

**Availability**
- “Table Browse panel” on page 689
- “Table Edit panel” on page 691
- “Table View panel” on page 700
- “Column Selection/Edit panel” on page 407

**Related tasks**
- Chapter 4, “Viewing and changing DB2 data,” on page 91
- Chapter 3, “Working with templates,” on page 49
FIND primary command

The FIND primary command searches the data for a character string or a numeric field with a particular numeric value.

In an FM/DB2 editor session, or the Object List utility:
- If the string or numeric value is found, then the FIND command positions the cursor at the beginning of the string or numeric field and, if necessary, automatically scrolls the found data into view.
- The FIND command highlights all occurrences of the search string or numeric value (even when you specify FIRST, LAST, PREV, NEXT, EX, NX, or X). To turn off the highlighting, enter the RESET FIND command.

In Print Browse:
- If the character string is found, the record containing the string is displayed at the top position.

To find the next occurrence of the same string, press the RFind function key (F5), enter FIND *, or enter the FIND command with no parameters. A message is displayed if the string cannot be found.

Note: FIND * does not repeat the previous FIND command with the same parameters. FIND * repeats the previous FIND command with the same string argument, but all other parameters revert to their default values unless specified.

Syntax

Editor session

```
Find
Fx
string
NEXT
CHARs
ALL
PREV
SUFFIX
LAST
FIRST

column list or range

label range
```

column list or range (SNGL or TABL display format):

```
#ALL

\( \text{col\_num} \)

\( \text{col\_num\_1} - \text{col\_num\_2} \)
```
FIND primary command

label range:

Notes:
1. You can specify parameters in any order.

Object List utility

Print Browse

* (asterisk)
    Uses the search string specified on the previous FIND command as the search string.

string
    The string you want to search for. The string, that can be up to 100 characters in length, can be:
    • A character string not starting or ending with a single quotation mark and not containing any embedded blanks or commas. The case of the string is ignored. Uppercase and lowercase representations of the same character match. For example, the following command finds the strings black, Black, and BLACK:
      FIND black
    • A character string enclosed in single quotation marks. The string can contain blanks and commas. The case of the string is ignored. For example, 'Exact string' matches 'exact string'.
    • C followed by a character string enclosed in quotation marks (C'Frog'), or a character string enclosed in quotation marks followed by C ('Frog'C). The string can contain blanks and commas. The string must match exactly (including case). For example, C'Exact string' does not match C'exact string'.
    • P preceded or followed by a picture string enclosed in single or double quotation marks to describe a type of string to be found rather than the exact characters. It can contain blanks, alphabetic and numeric characters which represent themselves, or any of the special characters listed here, each of which represents a class of characters:
FIND primary command

= Any character.
@ Alphabetic characters.
# Numeric characters.
$ Special characters.
& notsym;
   Non-blank characters.
. Invalid characters.
- Non-numeric characters.
< Lowercase alphabetic.
> Uppercase alphabetic.

Examples of picture strings used with the FIND command:

find p'.' #3
Find invalid character in columns 3.

find p'###'
Find 3 digit number (for example, 101 but not 99).

find '@1'p #1
Find label a1,b1,c1, (and so on) in column 1.

find p'<'
Find lowercase alphabetic character.

find p'-' #2
Find non-blank character in column 2.

When this notation is used, numeric, bit and unicode fields (for SNGL and TABL display formats) are excluded from the search process.

• X followed by a hexadecimal string enclosed in single quotation marks. For example, X'C1C2'.

• (FM/DB2 editor session only.) A numeric value (only when you limit the search by specifying column numbers, and only when the column being searched is a numeric column).

NEXT This is the default setting. Causes the search to begin at the cursor location (if the cursor is within the data portion of the display) or the beginning of the first row displayed, and searches ahead to find the next occurrence of string.

ALL Causes the search to begin at the top of the data and find all occurrences of the string.

Note: Not-selected rows that are hidden from display or represented by shadow lines are not processed by the FIND command, even when the ALL parameter is specified.

FIRST Causes the search to begin at the beginning of the table and search ahead.

LAST Causes the search to begin at the end of the table and search backwards.

PREV Causes the search to begin at the cursor location (if the cursor is within the data portion of the display) or the beginning of the first row displayed, and searches backwards to find the string.

CHARS Matches the search string anywhere in the data.

PREFIX
Matches the search string wherever it appears as a prefix in the data. To be
FIND primary command

a prefix, the matched text must be preceded by a non-alphanumeric character or be the start of a line or field, and must be followed by an alphanumeric character.

**SUFFIX**
Matches the search string wherever it appears as a suffix in the data. To be a suffix, the matched text must be preceded by an alphanumeric character, and must be followed by a non-alphanumeric character or be the end of a line or field.

**WORD**
Matches the search string wherever it appears as a word in the data. To be a word, the matched text must be preceded by a non-alphanumeric character or be the start of a line or field, and must be followed by a non-alphanumeric character or be the end of a line or field.

**#ALL**
Each column is searched according to its template attributes.

**col_num**
The DB2 column number (specified as #n) of a column to be included in the data search. Multiple column numbers must be separated by a comma or, if enclosed in brackets, separated by a blank or comma.

**col_num_1**
The first field reference of a range of fields. It cannot be subscripted. If the col_num_1 field reference value is less than the lowest displayed field reference value, the lowest displayed field reference value is used.

**col_num_2**
The last field reference of a range of fields. It cannot be subscripted. If the col_num_2 field reference value is greater than the highest displayed field reference value, the highest displayed field reference value is used.

The col_num_1 and col_num_2 field reference values must be separated by a hyphen (-). Spaces are permitted between the hyphen and the field reference values.

If col_num_1 is a higher value than col_num_2, the search process reverses the operands.

**label1**
(FM/DB2 editor session only.) Label identifying the start of a range of rows. The label must start with a period (.) followed by one to four alphabetic characters (no numeric or special characters). Labels starting with the letter “Z” indicate an editor-assigned label.

**label2**
(FM/DB2 editor session only.) Label identifying the end of a range of rows. The label must start with a period (.) followed by one to four alphabetic characters (no numeric or special characters). Labels starting with the letter “Z” indicate an editor-assigned label.

**EX**
Excluded rows only.

**NX**
Non-excluded rows only.

**X**
Same as EX.

**Availability**
- “Table Browse panel” on page 689
- “Table Edit panel” on page 691
- “Table View panel” on page 700
- Any Print Browse display panel
- “Collections panel” on page 396
FMAP primary command

The FMAP command can be used to edit or view a column in a table using a template. If the column being mapped is shorter than the template mapping a warning message is issued when data is changed and any data which is beyond the end of the mapped field is ignored.

The row containing the column to be mapped must be at the top of the screen or the cursor may be positioned on the row when the FMAP command is issued.

If the column identification parameter is not specified and the cursor is positioned on a column that column is used. Otherwise a Field List Selection panel will be displayed.

If the template name is not specified a panel is displayed where this information can be supplied.

Syntax

```
>> FMAP

  column# column_name template_name
```

- `column#` The reference number of the column to be mapped.
- `column_name` The name of the column to be mapped.
- `template_name` The name of the template to be used to map the column. A partially or fully qualified data set name together with the member name can be specified.
FORMAT primary command

The FORMAT primary command toggles between tabular display format (TABL) and single-row display format (SNGL). It has the same effect as changing the Format field (in the upper right corner of the panel).

Syntax

```
FORMAT
  Tabl
  Sngl
```

Tabl  In SNGL display format, changes to TABL display format and positions the current row at the top of the panel. This parameter is optional.

Sngl  In TABL display format, changes to SNGL display format and displays the row previously shown at the top of the panel. This parameter is optional.

Note:
1. You can also use the FS and FT primary commands to toggle between TABL and SNGL display formats.
2. In View and Edit, when in TABL display format, you can use the FS prefix command against a row to change to SNGL display format and display that row.

Availability

- “Table Browse panel” on page 689
- “Table Edit panel” on page 691
- “Table View panel” on page 700

Related tasks

- Chapter 4, “Viewing and changing DB2 data,” on page 91

FORWARD primary command

The FORWARD primary command scrolls forward (down) through your data.

The amount (number of rows or columns) scrolled is determined by either an optional parameter or, if no parameter is entered, by the amount indicated in the Scroll field.

Syntax
FORWARD primary command

Notes:
1  num_rows available in TABL display format; num_cols available in SNGL display format.

scroll_field  Scroll forward by the amount indicated in the Scroll field. This is the default value if no parameter is used.

CSR   Scroll forward to the cursor position.

DATA  Scroll forward one row (TABL) or column (SNGL) less than a page of data.

HALF  Scroll forward half a page of data.

Max   Scroll to bottom of table. This has the same effect as the BOTTOM command.


PAGE   Scroll forward one page of data.

Availability
- “Table Browse panel” on page 689
- “Table Edit panel” on page 691
- “Table View panel” on page 700

Related tasks
- Chapter 4, “Viewing and changing DB2 data,” on page 91

FS primary command

In TABL display format, you can use the FS primary command to change to SNGL display format. The row currently displayed at the top of the panel in TABL display format is shown in SNGL display format.

The FS primary command has the same effect as changing the Format field (in the upper right corner of the panel) from “TABL” to “SNGL”.

Syntax
FT primary command

In SNGL display format, you can use the FT primary command to change to TABL display format. The row currently displayed in SNGL display format is shown at the top of the panel in TABL display format.

The FT primary command has the same effect as changing the Format field (in the upper right corner of the panel) from “SNGL” to “TABL”.

Syntax

```
/SM590000/SM590000
```

HEX primary command

The HEX primary command sets or resets the hexadecimal display format.

Syntax

```
/SM590000/SM590000
```

(no parameter)

Toggles the correct hexadecimal-display status. That is, it turns hexadecimal display ON if currently OFF, and turns hexadecimal display OFF if currently ON.

`OFF` Display the data in character format.

`ON` Display the hexadecimal representation of the data vertically (three lines for each byte).

Availability

- “Table Browse panel” on page 689
- “Table Edit panel” on page 691
- “Table View panel” on page 700

Related tasks

- Chapter 4, “Viewing and changing DB2 data,” on page 91
HEX primary command

Related tasks

- “Displaying data in hexadecimal format” on page 128

HINT primary command

When you use the SQL prototyping, execution and analysis option (4), you can use the HINT primary command:

- On the Plan Table Rows panel to switch panel format to show hint IDs and hints used
- On the Statement Table Rows panel to switch panel format to show hint IDs and hints used

Syntax

```
 HINT
```

Availability

- “Plan Table Rows panel” on page 615
- “Statement Table Rows panel” on page 681

Related tasks

Chapter 12, “Working with SQL statements,” on page 317

INDEXINF primary command

The INDEXINF primary command shows or hides index information about each column when displayed in SNGL display format.

In SNGL display format, FM/DB2 displays an additional column (to the left of the column containing the column name) showing information about primary keys, foreign keys, unique indexes. (If you enter the INDEXINF primary command in TABL display format, the display is unaltered. However, if you change to SNGL display format, the display status of the additional column is reversed.)

If the index information is already displayed, the INDEXINF primary command removes it from display.

Note:

1. Information about primary keys, foreign keys, and unique indexes is only shown for tables (but not for views).
2. Foreign key information can only be displayed if the Retrieve foreign key information when building templates option is selected when the template is created.
3. The INDEXINF primary command has the same effect as selecting (or deselecting) the Show keys, indexes option.

Syntax

```
 INDEXINF
```

INSRPT primary command

Insert repeat. In advanced SELECT prototyping, if the cursor is positioned on a repeatable item, FM/DB2 creates a new repeat instance and selects the element on which the cursor was placed.

Syntax

```
/SM590000/SM590000
```

Availability

“Advanced SELECT Prototyping panel” on page 368

Related tasks

“Using advanced SQL prototyping” on page 325

LEFT primary command

- In View or Edit:
  - In TABL display format, the LEFT primary command scrolls to the left of your data. (You cannot use LEFT in SNGL display format.)
  - The amount (number of panel columns) scrolled is determined by either an optional parameter or, if no parameter is entered, by the amount indicated in the Scroll field.
- When editing a template:
  - Scrolls the Row Selection Criteria field on the Column Selection/Edit panel (the cursor must be positioned on the field at the time you issue the LEFT primary command)
  - Toggles the display (on an 80-character screen) on the Row Selection Criteria panel between the Column name and Data type(length) details appearing (the cursor must not be positioned on a Value field at the time you issue the LEFT primary command)

Syntax

FM/DB2 editor session
LEFT primary command

```
>>>LEFT
  scroll_field
  CSR
  DATA
  HALF
  Max
  num_cols
  PAGE
```

Template editing

```
>>>LEFT
```

scroll_field
Scroll left by the amount indicated in the Scroll field. This is the default value if no parameter is used.

CSR Scroll left to the cursor position.

DATA Scroll left one panel column less than a page of data.

HALF Scroll left half a page of data.

Max Scroll to the beginning of the row.

num_cols Scroll left num_cols panel columns. Valid range: 1–9999.

PAGE Scroll left one page of data.

Availability

- “Table Browse panel” on page 689
- “Table Edit panel” on page 691
- “Table View panel” on page 700
- “Column Selection/Edit panel” on page 407
- “Row Selection Criteria panel” on page 648

Related tasks

- “Scrolling” on page 125
- “Scrolling the Row Selection Criteria field” on page 64
- “Toggling the display (80-character screens only)” on page 61

LOBBRWS primary command

The LOBBRWS primary command starts a new FM/DB2 LOB editor session, in browse mode, for a LOB (BLOB, CLOB or DBCLOB) column. The DB2 object being processed must contain a LOB column.

The command is cursor sensitive. Type the command, move the cursor to the LOB column of interest and press Enter. An FM/DB2 LOB editor session, in browse mode, for the LOB column at the cursor position is displayed.

The command can also be issued without cursor positioning, in which case parameters may be required:
LOBBRWS primary command

Syntax

In SNGL display format

```
---LOBBRWS---
  #n
```

In TABL display format

```
---LOBBRWS---
  #n row
  row #n
```

#n  A valid reference. The data type for #n must be BLOB, CLOB or DBCLOB. #n can be omitted when the DB2 object contains only one LOB column. If #n is omitted when the DB2 object contains multiple LOB columns, FM/DB2 searches for the first selected LOB column, starting at #1.

row  A number that identifies a row in the editor session. A row number should not be specified when the command is issued in SNGL mode, the command applies to the currently displayed row.

Availability

- “Table Browse panel” on page 689
- “Table Edit panel” on page 691
- “Table View panel” on page 700

LOBEDIT primary command

The LOBEDIT primary command starts a new FM/DB2 LOB editor session, in edit mode, for a LOB (CLOB or DBCLOB) column. Edit of a BLOB is not supported. The DB2 object being processed must contain a LOB column.

The command is cursor sensitive. Type the command, move the cursor to the LOB column of interest and press Enter. An FM/DB2 LOB editor session, in edit mode, for the LOB column at the cursor position is displayed.

The command can also be issued without cursor positioning, in which case parameters may be required:

Syntax

In SNGL display format

```
---LOBEDIT---
  #n
```

In TABL display format

```
---LOBEDIT---
  #n row
  row #n
```

#n  A valid reference. The data type for #n must be a CLOB or DBCLOB. #n
LOBEDIT primary command

can be omitted when the DB2 object contains only one LOB column. If $n$
is omitted when the DB2 object contains multiple LOB columns, FM/DB2
searches for the first selected LOB column, starting at #1.

$\text{row}$  A number that identifies a row in the editor session. A row number should
not be specified when the command is issued in SNGL mode, the
command applies to the currently displayed row.

Availability
• “Table Edit panel” on page 691

LOBLEN primary command

The LOBLEN primary command displays the length for a LOB (BLOB, CLOB or
DBCLOB) column. The DB2 object being processed must contain a LOB column.

The command is cursor sensitive. Type the command, move the cursor to the LOB
column of interest and press Enter. The length for the LOB column at the cursor
position is displayed.

The command can also be issued without cursor positioning, in which case
parameters may be required:

Syntax

In SNGL display format

\[ \text{LOBLEN} \]

In TABL display format

\[ \text{LOBLEN} \]

#n  A valid reference. The data type for #n must be BLOB, CLOB or DBCLOB.
#n can be omitted when the DB2 object contains only one LOB column. If
#n is omitted when the DB2 object contains multiple LOB columns,
FM/DB2 searches for the first selected LOB column, starting at #1.

$\text{row}$  A number that identifies a row in the editor session. A row number should
not be specified when the command is issued in SNGL mode, the
command applies to the currently displayed row.

Availability
• “Table Browse panel” on page 689
• “Table Edit panel” on page 691
• “Table View panel” on page 700

LOBVIEW primary command

The LOBVIEW primary command starts a new FM/DB2 LOB editor session, in
view mode, for a LOB (BLOB, CLOB or DBCLOB) column. The DB2 object being
processed must contain a LOB column
**LOBVIEW primary command**

The command is cursor sensitive. Type the command, move the cursor to the LOB column of interest and press Enter. An FM/DB2 LOB editor session, in view mode, for the LOB column at the cursor position is displayed.

The command can also be issued without cursor positioning, in which case parameters may be required:

**Syntax**

In **SNGL display format**

```
>>LOBVIEW #n
```

In **TABL display format**

```
>>LOBVIEW #n row
```

```
row #n
```

#n  A valid reference. The data type for #n must be BLOB, CLOB or DBCLOB. #n can be omitted when the DB2 object contains only one LOB column. If #n is omitted when the DB2 object contains multiple LOB columns, FM/DB2 searches for the first selected LOB column, starting at #1.

row  A number that identifies a row in the editor session. A row number should not be specified when the command is issued in SNGL mode, the command applies to the currently displayed row.

**Availability**

- "Table Edit panel" on page 691
- "Table View panel" on page 700

---

**LOCATE primary command**

- When you use the LOCATE primary command in an editor session or the Object List utility:
  - FM/DB2 moves the display to the row, or first column, that matches the search criteria.
  - To move to a particular row, specify the row number. You can specify a label to move to a row to which that label has previously been assigned.
  - To search for a particular column, specify either the column number (for example, #4), or part or all of a column name (for example, SAL). When you search for part or all of a column name, you can use one of the optional parameters to nominate the direction and starting point for the search. Where there may be ambiguity, use the FLD parameter to differentiate between a column name and a DB2 column number.
- When you use the LOCATE primary command in Print Browse (3.11):
  - Positions a specified line number at the top of the display.
LOCATE primary command

Syntax

FM/DB2 editor session

LOCATE

Notes:
1. These forms of the locate command locate a column.
2. These forms of the locate command locate a row.

Object List utility

Print Browse

string A column name, or part of a column name
#n A valid reference to a column.
pos An integer representing a character position within a long (>50 characters) character or graphic column.
n The sequence number of a row.
label The label of a labeled row. label may be a user or system label.
row_num The number of the row you want to locate. The row number is shown in
LOCATE primary command

the prefix area (TABL display mode), or in the top right-hand part of the
screen (SNGL display mode). For example:

LOCATE 23

locates row number 23.

Note: The row number is not shown when data is displayed with the
Object List utility.

col_num
The DB2 column number (specified as #n) of the column you want to
locate. For example:

L #12

locates column #12.

label
An existing user-assigned or editor-assigned label, identifying the row that
you want to locate. The label must start with a period (.) followed by one
to four alphabetic characters (no numeric or special characters). Labels
starting with the letter “Z” indicate an editor-assigned label. For example:

LOCATE .HERE

locates a row marked with an existing label (.HERE), and

L .ZLST

locates the last row.

FLD
Indicates to FM/DB2 that the string (string) following the FLD keyword is
the name or part of the name of the column you want to locate. For
example:

L FLD #ITEMS

locates the column named #ITEMS. (In this case, if you omit the FLD
parameter, the command L #ITEMS results in an error as FM/DB2 expects
the # symbol to be followed by a numeric DB2 column number.)

string
The name (or part of a name) of the column you want to locate. The string
can occur anywhere in the column name. For example:

L SAL

or

L ARY

locates a column named “SALARY”.

NEXT
This is the default setting. In TABL display format, the search for string
starts at (but excludes) the column currently displayed on the left of the
screen and scans right. In SNGL display format, the search for string starts
at (but excludes) the column currently displayed at the top of the screen
and scans down. For example:

L NEXT SAL

or

L SAL NEXT

locates the next column whose name contains the string “SAL”.

Chapter 16. FM/DB2 commands 763
LOCATE primary command

**FIRST**  In TABL display format, the search for *string* starts at (but excludes) the first column for the DB2 object and scans right. In SNGL display format, the search for *string* starts at (but excludes) the first column for the DB2 object and scans down. For example:

L FIRST SAL

or

L SAL FIRST

locates the first column whose name contains the string “SAL”.

**LAST**  In TABL display format, the search for *string* starts at the last column for the DB2 object and scans left. In SNGL display format, the search for *string* starts at the last column for the DB2 object and scans up. For example:

L LAST SAL

or

L SAL LAST

locates the last column whose name contains the string “SAL”.

**PREV**  In TABL display format, the search for *string* starts at the column currently displayed on the left of the screen and scans left. In SNGL display format, the search for *string* starts at the column currently displayed at the top of the screen and scans up. For example:

L PREV SAL

or

L SAL PREV

locates the previous column whose name contains the string “SAL”.

**line_num**  The line number you want positioned at the top of the display.

**Availability**

- “Table Browse panel” on page 689
- “Table Edit panel” on page 691
- “Table View panel” on page 700
- “Collections panel” on page 396
- “Columns panel” on page 411
- “Databases panel” on page 500
- “Database Request Modules panel” on page 498
- “Distinct Types panel” on page 523
- “Functions panel” on page 584
- “Indexes panel” on page 595
- “Application Packages panel” on page 387
- “Application Plans panel” on page 389
- “Schemas panel” on page 657
- “Stored Procedures panel” on page 685
- “Synonyms panel” on page 687
- “Tables, Views and Aliases panel” on page 704
- “Table Spaces panel” on page 698
- “Triggers panel” on page 712
- “Storage Group(s) panel” on page 683
NEW, NEWS primary command

The NEW (or NEWS) primary command displays a panel providing general information about the current FM/DB2 version and release.

Syntax

```
NEW
NEWS
```

Availability

- All panels

NEXT primary command

Scrolls forward the number of rows specified.

Syntax

```
Next
```

```
num_rows
Max
```

`num_rows`

Scroll forward `num_rows` rows. Valid range: 1–999999999.

`Max`

Scroll forward to make the last row visible.

Availability

- “Table Browse panel” on page 689
- “Table Edit panel” on page 691
- “Table View panel” on page 700

Related tasks

- Chapter 4, “Viewing and changing DB2 data,” on page 91
- “Scrolling” on page 125

NEXTRPT primary command

In advanced SELECT prototyping, if the cursor is positioned on a repeatable item within the syntax, the NEXTRPT primary command makes the next repeatable item current.
NEXTRPT primary command

Syntax

\[\text{NEXTRPT}\]

Availability

“Advanced SELECT Prototyping panel” on page 368

Related tasks

“Using advanced SQL prototyping” on page 325

PB primary command

The PB primary command lets you browse, and optionally purge, the print output from your current FM/DB2 editor session held in the print data set.

If you enter the PB primary command when you have not created any print output, FM/DB2 displays the message *Empty print data set*.

Syntax

\[\text{PB}\]

Availability

Available on all panels

Related tasks

“Looking at the print output from your FM/DB2 session” on page 266

PLAN primary command

When you use the SQL prototyping, execution and analysis option (4), you can use the PLAN primary command:

- On the Plan Table Rows panel to switch panel format to show the plan name.
- On the Statement Table Rows panel to switch panel format to show the plan name.

Syntax

\[\text{PLAN}\]

Availability

“Plan Table Rows panel” on page 615

Related tasks

Chapter 12, “Working with SQL statements,” on page 317
PREFIX primary command

For TABL display format only, the PREFIX primary command sets the position and display state of the prefix area.

Syntax

\[
\text{PREFIX} \begin{align*}
\text{LEFT} \\
\text{RIGHT} \\
\text{OFF} \\
\text{ON} \\
n \\
A
\end{align*}
\]

LEFT  Displays the prefix area on the left side.
RIGHT Displays the prefix area on the right side.
OFF   Does not display the prefix area.
ON    Displays the prefix area at the position last set, with the display width last set.
\(n\)  Displays the prefix area at the position last set, with a display width of \(n\) digits. \(n\) must be in the range: 6–9.
A     Displays the prefix area at the position last set, with a display width of 6 digits or, if necessary, larger (up to 9 digits) in order to display the whole record number.

Availability
- “Table Edit panel” on page 691

Related tasks
- “Managing the prefix area” on page 129

PREVIOUS primary command

Scrolls backward the number of rows specified.

Syntax

\[
\text{PREVIOUS} \begin{align*}
\text{num\_rows} \\
\text{Max}
\end{align*}
\]

\(\text{num\_rows}\)  Scroll backward \(\text{num\_rows}\) rows. Valid range: 1–999999999.
Max   Scroll backward to make the first row visible.

Availability
- “Table Browse panel” on page 689
- “Table Edit panel” on page 691
PREVIOUS primary command

- "Table View panel" on page 700

Related tasks
- Chapter 4, "Viewing and changing DB2 data," on page 91
- "Scrolling" on page 125

PREVRPT primary command

In advanced SELECT prototyping, if the cursor is positioned on a repeatable item within the syntax, the PREVRPT primary command makes the previous repeatable item current.

Syntax

```plaintext
PREVRPT
```

Availability

"Advanced SELECT Prototyping panel" on page 368

Related tasks

"Using advanced SQL prototyping" on page 325

PRINT primary command

In Print Browse, the PRINT primary command transfers the contents of the print data set to a SYSOUT class.

You specify the print data set using the PRINTDSN entry field on the Set Print Processing Options panel.

The SYSOUT class is determined by one of the following (whichever is found to exist first):
- The class specified by SYSOUT=c in the PRINTOUT entry field on the Print Utility panel (unless this field contains a value other than SYSOUT=c)
- Your default TSO SYSOUT class (if defined)
- SYSOUT class A (if neither of the above apply)

Note: When you enter PRINT on the command line for any FM/DB2 panel other than the Print Browse panel, it is interpreted as the ISPF system command, PRINT, and records a snapshot of the physical screen image in the print data set.

Syntax

```plaintext
PRINT
```

Availability

Any Print Browse display panel
PRINT primary command

Related tasks
- “Looking at the print output from your FM/DB2 session” on page 266

PURGE primary command

In Print Browse, the PURGE command purges the print output by clearing the print data set.

Syntax

/PURGE/

Availability

A Print Browse display panel

Related tasks
- “Looking at the print output from your FM/DB2 session” on page 266

QUIT primary command

See “CANCEL primary command” on page 730.

RCHANGE primary command

The RCHANGE primary command changes the next occurrence of the string or numeric value by repeating the previous CHANGE primary command.

Syntax

/RC CHANGE

Availability

- “Table Edit panel” on page 691

Related tasks
- “Changing data with the CHANGE command” on page 140

RD primary command

The RD (“Record Dump”) primary command prints, for the number of rows specified, in dump format using the current display format (TABL or SNGL).

Syntax

/RD num_rows

- Max
- All
RD primary command

num_rows
Print num_rows rows, starting from the current row. Valid range: 1–999999999.

Max Print all remaining rows, from the current row to the end of the DB2 object.

All Print all rows in the DB2 object.

Availability
- “Table Browse panel” on page 689
- “Table Edit panel” on page 691
- “Table View panel” on page 700

Related tasks
- “Printing rows from a DB2 table” on page 264

REDIT primary command

When you issue the REDIT primary command from within an information panel for a referential integrity error, FM/DB2 displays an Edit session.

When you issue the REDIT primary command from within an FM/DB2 View or Edit editor session, FM/DB2 displays the Related tables panel listing information about the tables related to the table you are currently editing.

Note:
1. The REDIT primary command is not available in FM/DB2 Browse.
2. The REDIT command works with DB2 tables only. You cannot use the command with DB2 views.

Syntax

Syntax (when issued for a referential integrity error)

REDIT

Syntax (when issued from within an FM/DB2 editor session)

REDIT col_num

col_num
A DB2 column number (specified as #n).

Availability
- Within an information panel for a referential integrity error.
- From within an FM/DB2 editor session.

Related tasks
- “Starting an Edit session of a related table” on page 152
- “Listing related tables” on page 150
REFRESH primary command

The REFRESH primary command refreshes the list of DB2 subsystems on the DB2 Subsystem Selection menu.

Syntax

```
REFRESH
```

Availability

“DB2 Subsystem Selection panel” on page 512

Related tasks

“Selecting the DB2 subsystem when more than one is available” on page 11

REFS primary command

The REFS primary command shows or hides the column number (as defined in the DB2 catalog) for each displayed column when in SNGL display format.

Syntax

```
REFS
```

Availability

• “Table Browse panel” on page 689
• “Table Edit panel” on page 691
• “Table View panel” on page 700

Related tasks

• “Displaying the column number” on page 130

RESET primary command

In an FM/DB2 editor session, you use the RESET command to redisplay excluded rows, or to reset strings highlighted by the FIND command. You can also use the RESET command to reset pending prefix commands, or to remove labels from rows.

For basic SELECT prototyping, the RESET primary command clears previous information from the SELECT, WHERE, and ORDER BY clauses in the SELECT statement at the top of the Basic SELECT prototyping panel.

When you are viewing the DB2 utility options panels, the RESET primary command sets the values for each option back to the installation default.
RESET primary command

Syntax
Standard

COMMAND
All pending prefix commands are reset.

EXCLuded
All excluded rows are redisplayed.

FIND
All strings highlighted by the FIND command are reset.

LABEL
All row labels are deleted.

Availability
- “Table View panel” on page 700
- “Table Browse panel” on page 689
- “Table Edit panel” on page 691
- “Basic SELECT Prototyping panel” on page 391

Related tasks
- Chapter 4, “Viewing and changing DB2 data,” on page 91
- “Using basic SQL prototyping” on page 317
- “Setting options for DB2 utilities” on page 278

RFIND primary command

The RFIND primary command repeats the search performed by the previous FIND primary command.

Syntax

Availability
- “Table Browse panel” on page 689
- “Table Edit panel” on page 691
- “Table View panel” on page 700

Related tasks
- “Repeating the search” on page 112
RIGHT primary command

- In View or Edit:
  In TABL display format, the RIGHT primary command scrolls to the right of your data. (You cannot use RIGHT in SNGL display format.)
  The amount (number of panel columns) scrolled is determined by either an optional parameter or, if no parameter is entered, by the amount indicated in the Scroll field.
- When editing a template:
  When you are editing a template, the RIGHT primary command:
  - Scrolls the Row Selection Criteria field on the Column Selection/Edit panel (the cursor must be positioned on the field at the time you issue the RIGHT primary command)
  - Toggles the display (on an 80-character screen) on the Row Selection Criteria panel between the Column name and Data type(length) details appearing (the cursor must not be positioned on a Value field at the time you issue the RIGHT primary command)

Syntax

View or Edit

```
RIGHT scroll_field
```

Template editing

```
RIGHT scroll_field
```

**scroll_field**
- Scroll right by the amount indicated in the Scroll field. This is the default value if no parameter is used.
- CSR Scroll right to the cursor position.
- DATA Scroll right one panel column less than a page of data.
- HALF Scroll right half a page of data.
- Max Scroll to the end of the row.
- num_cols Scroll right num_cols panel columns. Valid range: 1–9999.
- PAGE Scroll right one page of data.

Availability
- “Table Browse panel” on page 689
- “Table Edit panel” on page 691
- “Table View panel” on page 700
- “Column Selection/Edit panel” on page 407
RIGHT primary command

- “Row Selection Criteria panel” on page 648

Related tasks
- “Scrolling” on page 125
- “Scrolling the Row Selection Criteria field” on page 64
- “Toggling the display (80-character screens only)” on page 61

RP primary command

The RP (“Record Print”) primary command prints, for the number of rows specified, in character format using the current display format (TABL or SNGL).

Syntax

```
RP
    num_rows
    Max
    All
```

`num_rows`
Print `num_rows` rows, starting from the current row. Valid range: 1–999999999.

`Max`
Print all remaining rows, from the current row to the end of the DB2 object.

`All`
Print all rows in the DB2 object.

Availability
- “Table Browse panel” on page 689
- “Table Edit panel” on page 691
- “Table View panel” on page 700

Related tasks
- “Printing rows from a DB2 table” on page 264

RUNTEMP primary command

The RUNTEMP primary command runs the current function with the template changes you have made but does not save the changes. RUNTEMP is only available when you run a function that uses a template.

Syntax

```
RUNTEMP
```

Availability
- “Template Save panel” on page 712
- “Column Selection/Edit panel” on page 407

Related tasks
- Chapter 4, “Viewing and changing DB2 data,” on page 91
- “Printing the contents of a DB2 table” on page 261
SAVE primary command

In Edit, the SAVE primary command saves any uncommitted changes without ending the Edit session.

Syntax

```
SAVE
```

Availability

- "Table Edit panel" on page 691

Related tasks

- "Ending an FM/DB2 editor session" on page 101
- "Updating a primary key" on page 154

SAVEAS primary command

When you edit a template, the SAVEAS primary command displays the save panel where you can save the template as another name.

Syntax

```
SAVEAS
```

Availability

- "Column Selection/Edit panel" on page 407

Related tasks

- Chapter 3, "Working with templates," on page 49

SHADOW primary command

The SHADOW command hides or shows shadow lines.

Note: Shadow lines represent rows that have been excluded from display by the EXCLUDE primary command.

Syntax

```
SHADOW EX OFF TEMP
   ALL X ON PERM
```
**SHADOW primary command**

<table>
<thead>
<tr>
<th>EX</th>
<th>Show or hide “excluded” rows shadow lines. This is the default setting.</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALL</td>
<td>Show or hide “excluded” rows shadow lines.</td>
</tr>
<tr>
<td>X</td>
<td>Same as EX.</td>
</tr>
<tr>
<td>OFF</td>
<td>Hide “excluded” rows shadow lines.</td>
</tr>
<tr>
<td>ON</td>
<td>Show “excluded” rows shadow lines.</td>
</tr>
<tr>
<td>TEMP</td>
<td>Shadow setting applies to the current FM/DB2 editor session only. This is the default setting.</td>
</tr>
<tr>
<td>PERM</td>
<td>Shadow setting is saved in your FM/DB2 profile, and applies to all FM/DB2 editor sessions until changed.</td>
</tr>
</tbody>
</table>

**Availability**
- “Table Browse panel” on page 689
- “Table Edit panel” on page 691
- “Table View panel” on page 700

**Related tasks**
- “Excluding rows” on page 126

---

**SHOW primary command**

The SHOW command restricts the list of DB2 subsystems shown on the DB2 Subsystem Selection menu.

**Syntax**

```
SHOW partial_ID
```

* Show all DB2 subsystems.

**partial_ID**

The partial ID of a DB2 subsystem, starting or ending with an asterisk (*), or both. Shows all DB2 subsystems whose ID matches the partial ID specified.

**Availability**
- “DB2 Subsystem Selection panel” on page 512

**Related tasks**
- “Restricting the list of DB2 subsystems” on page 14

---

**SHOWCOB primary command**

The SHOWCOB command displays details showing which COBOL compiler is currently being used.

**Syntax**

```
SHOWCOB
```

---

776 File Manager/DB2 Data V13R1 User's Guide
SHOWCOB primary command

Availability

Available on all panels

Related tasks

• “Checking which COBOL compiler you’re using” on page 10

SORT primary command

• In View or Edit, the SORT primary command changes the order of the data you are browsing or editing in TABL display format.

  FM/DB2 sorts the data according to its hexadecimal representation using a hierarchy of keys, starting with the first specified column as the primary key, and subsequent specified columns as less significant keys. You specify the hierarchy left to right.

• When you use the Object List utility (3.4) to display a list of objects, you can sort a column in the current object list by issuing the SORT primary command with the cursor positioned on the column you want to sort.

Syntax

Syntax (View or Edit)

```
SORT col_num
```

Syntax (Object List utility)

```
SORT
```

col_num

  The DB2 column number (specified as #n) of a column to be used as the sort key. You can specify between 1 and 5 column numbers. Multiple column numbers must be separated by a blank.

A  Sort data in ascending sequence. This is the default sequence.

D  Sort data in descending sequence.

NX Non-excluded rows only. Sorts rows as if any excluded rows did not exist.

X  Excluded rows only. Sorts excluded rows as if any non-excluded rows did not exist.

Availability

• “DB2 View panel” on page 518
• “DB2 Browse panel” on page 501
• “DB2 Edit panel” on page 505
• Any panels displayed with the Object List utility (3.4)

Related tasks

• Chapter 4, “Viewing and changing DB2 data,” on page 91
• Chapter 8, “Working with lists of DB2 objects,” on page 241
SQL primary command

The SQL primary command displays the SQL SELECT statement for the object you are processing. You can view the SELECT statement and save it in an ISPF data set for future reference.

When you enter the SQL command, the SELECT statement is presented in a standard ISPF EDIT panel. You can save the statement using the ISPF CREATE command. To do this:
1. On the command line, type CREATE.
2. In the prefix area of the top line of the SELECT statement, type c9999.
3. Press Enter.

You are asked to supply the data set and member name in which to save the SELECT statement. If the data set does not exist, you are asked to specify the allocation attributes to create it. You cannot use the attributes of the temporary data set displayed; you must provide the required attributes yourself.

Note: When you start an editor session from the function entry panel, the SELECT clause in the SQL statement includes every column of the table, even if you have edited the template and deselected one or more columns.

When you are working with SQL statements, any changes you make to the command in the ISPF Edit session are not preserved when you return to the prototyper.

Syntax

Availability

- “Table Browse panel” on page 689
- “Table Edit panel” on page 691
- “Table View panel” on page 700
- “Basic SELECT Prototyping panel” on page 391
- “Advanced SELECT Prototyping panel” on page 368
- “Enter, Execute and Explain SQL Statements panel” on page 555

Related tasks

- Chapter 4, “Viewing and changing DB2 data,” on page 91
- Chapter 12, “Working with SQL statements,” on page 317

SQLID primary command

The SQLID primary command allows you to change the SQLID to be used as part of the connection to DB2.

Syntax

DB2_sqlid

The required SQLID to be used as part of the connection to DB2.
SQLID primary command

**Availability**
- The SQLID command can be issued on most FM/DB2 processing panels.
  You cannot issue the command from:
  - Pop-up panels
  - Tutorial panels
  - Panels displayed by ISPF. For example, the Audit log listing.

**Related tasks**
- "Specifying the SQLID" on page 29

TEDIT primary command

The TEDIT command displays the Column Selection/Edit panel from within an FM/DB2 editor session.

**Syntax**

```
<<<TEdit
```

**Availability**
- "Table Edit panel" on page 691
- "Table Browse panel" on page 689
- "Table View panel" on page 700

**Related tasks**
- "Editing a template" on page 57

TOP primary command

The TOP primary command scrolls to the first page of data.

**Syntax**

```
<<<Top
```

**Availability**
- "Table Browse panel" on page 689
- "Table Edit panel" on page 691
- "Table View panel" on page 700

**Related tasks**
- Chapter 4, "Viewing and changing DB2 data," on page 91
- "Scrolling" on page 125

TP primary command

The TP primary command prints the current template.

**Syntax**
TP primary command

Availability
- “Column Selection/Edit panel” on page 407

Related tasks
- “Printing a template” on page 52

TYPE primary command

The TYPE primary command shows or hides the data type (and, for non-numeric columns, the length of the column) for each displayed column when in SNGL display format.

Syntax

Availability
- “Table Browse panel” on page 689
- “Table Edit panel” on page 691
- “Table View panel” on page 700

Related tasks
- “Displaying the data type” on page 130

UNDO primary command

Use the UNDO primary command to undo the last set of changes you made to the SELECT statement at the top of the Basic SELECT prototyping panel.

Syntax

Availability
- “Basic SELECT Prototyping panel” on page 391

Related tasks
- “Using basic SQL prototyping” on page 317

UP primary command

The UP primary command scrolls up (backward) through your data.

The amount (number of rows or columns) scrolled is determined by either an optional parameter or, if no parameter is entered, by the amount indicated in the Scroll field.
UP primary command

Syntax

```
UP [scroll_field]
CSR
DATA
HALF
Max
(num_rows)
(num_cols)
PAGE
```

Notes:

1 num_rows available in TABL display format; num_cols available in SNGL display format.

scroll_field
Scroll up by the amount indicated in the Scroll field. This is the default value if no parameter is used.

CSR
Scroll up to the cursor position.

DATA
Scroll up one row (TABL) or column (SNGL) less than a page of data.

HALF
Scroll up half a page of data.

Max
Scroll to top of the table. This has the same effect as TOP command.

num_rows

num_cols
Scroll up num_cols columns. Valid range: 1–9999. Available in SNGL display format.

PAGE
Scroll up one page of data.

Availability

- “Table Browse panel” on page 689
- “Table Edit panel” on page 691
- “Table View panel” on page 700
- “Enter, Execute and Explain SQL Statements panel” on page 555

Related tasks

- Chapter 4, “Viewing and changing DB2 data,” on page 91
- “Scrolling” on page 125
- “Entering, executing, and explaining SQL statements” on page 351

VER primary command

The VER primary command displays the current FM/DB2 version number and the PTF number for each FM/DB2 component in a window. The window also indicates whether or not FM/DB2 is APF-authorized.
VER primary command

Syntax

```plaintext
VER current_object_owner...current_object_name
```

Availability

Available on all panels

Related tasks

"Checking your FM/DB2 version" on page 9

VIEW primary command

The VIEW primary command starts a new FM/DB2 View editor session.

Syntax

```plaintext
VIEW current_object_owner...current_object_name
```

```
current_object_owner
The name of the owner of the object in the current FM/DB2 editor session.

current_object_name
The name of the object in the current FM/DB2 editor session.

new_object_owner
The name of the owner of the object to be pre-filled in the entry panel for the new FM/DB2 editor session.

new_object_name
The name of the object to be pre-filled in the entry panel for the new FM/DB2 editor session.

Availability

- "Table Browse panel" on page 689
- "Table Edit panel" on page 691
- "Table View panel" on page 700

Related tasks

- Chapter 4, “Viewing and changing DB2 data,” on page 91

XMLBRWS primary command

The XMLBRWS primary command starts a new ISPF editor session, in browse mode, for an XML document. The DB2 object being processed must contain an XML column.

The command is cursor sensitive. Type the command, move the cursor to the XML column of interest and press Enter. An ISPF editor session, in browse mode, for the XML column at the cursor position is displayed.
XMLBRWS primary command

The command can also be issued without cursor positioning, in which case parameters may be required:

Syntax

**In SNGL display format**

```
--------XMLBRWS-#n--------
```

**In TABL display format**

```
--------XMLBRWS-#n-row--row--#n--------
```

#n   A valid reference. The data type for #n must be XML. #n can be omitted when the DB2 object contains only one XML column. If #n is omitted when the DB2 object contains multiple XML columns, FM/DB2 searches for the first selected XML column, starting at #1.

row  A number that identifies a row in the editor session. A row number should not be specified when the command is issued in SNGL mode, the command applies to the currently displayed row.

Availability

- “Table Browse panel” on page 689
- “Table Edit panel” on page 691
- “Table View panel” on page 700

XMLEDIT primary command

The XMLEDIT primary command starts a new ISPF editor session, in edit mode, for an XML document. The DB2 object being processed must contain an XML column.

The command is cursor sensitive. Type the command, move the cursor to the XML column of interest and press Enter. An ISPF editor session, in edit mode, for the XML column at the cursor position is displayed.

The command can also be issued without cursor positioning, in which case parameters may be required:

Syntax

**In SNGL display format**

```
--------XMLEDIT-#n--------
```
XMLEDIT primary command

In TABL display format

```
**XMLEDIT**
  \#n---row---
  row---\#n
```

\#n  A valid reference. The data type for \#n must be XML. \#n can be omitted when the DB2 object contains only one XML column. If \#n is omitted when the DB2 object contains multiple XML columns, FM/DB2 searches for the first selected XML column, starting at \#1.

row  A number that identifies a row in the editor session. A row number should not be specified when the command is issued in SNGL mode, the command applies to the currently displayed row.

Availability

- "Table Edit panel" on page 691

XMLLEN primary command

The XMLLEN primary command displays the length for an XML document. The DB2 object being processed must contain an XML column.

The command is cursor sensitive. Type the command, move the cursor to the XML column of interest and press Enter. The length for the XML document at the cursor position is displayed.

The command can also be issued without cursor positioning, in which case parameters may be required:

Syntax

In SNGL display format

```
**XMLLEN**
  \#n
```

In TABL display format

```
**XMLLEN**
  \#n---row---
  row---\#n
```

\#n  A valid reference. The data type for \#n must be XML. \#n can be omitted when the DB2 object contains only one XML column. If \#n is omitted when the DB2 object contains multiple XML columns, FM/DB2 searches for the first selected XML column, starting at \#1.

row  A number that identifies a row in the editor session. A row number should not be specified when the command is issued in SNGL mode, the command applies to the currently displayed row.

Availability

- "Table Browse panel" on page 689
- "Table Edit panel" on page 691
XMLVIEW primary command

The XMLVIEW primary command starts a new ISPF editor session, in view mode, for an XML document. The DB2 object being processed must contain an XML column.

The command is cursor sensitive. Type the command, move the cursor to the XML column of interest and press Enter. An ISPF editor session, in view mode, for the XML column at the cursor position is displayed.

The command can also be issued without cursor positioning, in which case parameters may be required:

Syntax

In SNGL display format

```
XMLVIEW #n
```

In TABL display format

```
XMLVIEW #n row
row #n
```

#n A valid reference. The data type for #n must be XML. #n can be omitted when the DB2 object contains only one XML column. If #n is omitted when the DB2 object contains multiple XML columns, FM/DB2 searches for the first selected XML column, starting at #1.

row A number that identifies a row in the editor session. A row number should not be specified when the command is issued in SNGL mode, the command applies to the currently displayed row.

Availability

• "Table Edit panel" on page 691
• "Table View panel" on page 700

ZOOM primary command

In TABL display format, the ZOOM command zooms in on the row selected by the cursor, or the top row displayed if the cursor is not positioned on a row.

In SNGL display format, the ZOOM command shows the current row. If some columns have been deselected in the template, these columns are now shown.

Syntax

```
ZOOM
```
ZOOM primary command

**Availability**
- "Table Browse panel" on page 689
- "Table Edit panel" on page 691
- "Table View panel" on page 700

**Related tasks**
- "Zooming in to see all of a row" on page 123
Chapter 17. FM/DB2 functions

This chapter contains information about the FM/DB2 functions that can be executed in batch jobs.

The preferred method for generating the FM/DB2 batch utility control statements is to allow FM/DB2 to generate the required statements.

The syntax diagrams and keyword descriptions in this section are provided for reference only.

How to use this reference chapter

This chapter lists the FM/DB2 batch functions in alphabetical order.

The following subsections are supplied, where relevant, for each function:

Purpose
   A short summary of what you can do with the function.

Usage notes
   A more detailed explanation of how to use the function.

Syntax
   A syntax diagram, followed by a parameter list. The parameter list describes the parameter, and shows (where appropriate) its default and other possible values. If you specify parameters that are not relevant to the function, File Manager ignores them. If you specify the same parameter more than once, File Manager uses the first value that you specify.

Batch example
   A listing of a sample batch job.

Return codes
   Lists the return codes that are specific to the function.

Related functions
   Other functions that are related in some way (for example, the export and import functions can be used to transfer data between DB2 systems)

General tips about performance when you use File Manager functions

The following tips are provided as a guide to improving performance when using File Manager functions:

Use the FASTREXX subset of REXX
   If a user PROC is required, it is strongly recommended that, rather than the full REXX language, the FASTREXX subset of REXX is used. File Manager provides a set of external REXX functions designed to allow many common tasks to be performed within the FASTREXX subset. See Chapter 14, “Enhancing File Manager processing” on page 345 for a discussion of FASTREXX.

Template and copybook processing
   The utilities DBX and DBI allow a copybook or template to be specified for the target/input data set, respectively. The utilities DBX, DBI, DBC, D2G
General tips about performance when you use File Manager functions

and DBP optionally require a DB2 template that describes the DB2 object to be processed. When using these functions, consider the following performance-related points:

- File Manager can be most efficient if a template is used instead of a copybook. If a copybook must be used, specify the language of the copybook. (LANG=COBOL, LANG=HLASM, or LANG=FLI, instead of the default LANG=AUTO) for best performance.
- Template expressions should be written, if at all possible, using expressions suitable for File Manager internal processing. For more details, see File Manager User's Guide and Reference "Improving performance using internal expression processing" on page 144.

File Manager DB2 batch functions

This section describes the File Manager DB2 functions that you can use in batch jobs. For more information about using these functions, see Chapter 6, "Populating a DB2 table with data," on page 191, Chapter 7, "Copying data," on page 193, and Chapter 9, "Printing," on page 261.

DBC (Copy) batch command

Purpose

To copy data from one DB2 object to another DB2 object. Both objects need to be accessible from the currently connected DB2 system.

Usage

The FM/DB2 copy utility can:

- Optionally delete all rows from the target table prior to copying any data.
- Select the rows and columns to be copied.
- Limit the total number of rows to be copied.
- Reformat data into one or more new columns.
- Initialize columns with a new value or pattern.

The FM/DB2 copy utility takes no account of any referential integrity relationships defined on the source DB2 object.
DBC (Copy) batch command

**OBJOUT=**

The optional name of the DB2 remote server (location) where the source object is located; the optional name of the owner of the source object (owner) and the source object name (name) to be copied.

When location is not specified the current (local) DB2 server is used. When the owner is not specified the object name is qualified using the current SQLID. When FM/DB2 generates the utility control statements, the owner value is non-blank.

OBJIN=location,owner,name

The optional name of the DB2 remote server (location) where the source object is located; the optional name of the owner of the source object (owner) and the source object name (name) to be copied.

When location is not specified the current (local) DB2 server is used. When the owner is not specified the object name is qualified using the current SQLID. When FM/DB2 generates the utility control statements, the owner value is non-blank.

OBJILOCN=location

The optional name of the DB2 remote server (location) where the source object is located. See “Specifying a DB2 object name” on page 815.

OBJIOWNR=owner

The optional name of the owner of the source object (owner) to be copied. See “Specifying a DB2 object name” on page 815.

OBJINAME=name

The object name (name) for the source object to be copied. See “Specifying a DB2 object name” on page 815.

NATIVE

Specifies whether FM/DB2 should process Unicode encoded data natively when copying data between two Unicode encoded tables. This option can be ignored if the default processing of Unicode data is desired; that is, Unicode data is potentially substituted with error characters when copying data between Unicode tables.

YES Process Unicode data natively.
NO Do not Process Unicode data natively. This is the default.
DBC (Copy) batch command

**TINPUT=ddname**
Defines a reference to a DD statement for the data sets which contain the DB2 template that describes the source DB2 object to be copied. If you specify a concatenated DD, then you must provide the member name, *member*, via the TINMEM keyword. See “Specifying the template for a DB2 object” on page 815.

**TINMEM=member**
The name of the template member in the dataset(s) identified by the TINPUT parameter, if it has not been specified on the DD statement. This parameter must not be specified if the TMIN parameter is specified. See “Specifying the template for a DB2 object” on page 815.

**TMIN=template_data_set_name(member)**
The PDS (*template_data_set_name*) and member name (*member*) of the File Manager DB2 template that describes the source DB2 object to be copied. See “Specifying the template for a DB2 object” on page 815.

**OBJOUT=location.owner.name**
The optional name of the DB2 remote server (*location*) where the target object is located; the optional name of the owner of the target object (*owner*) and the target object name (*name*) to be copied.

When location is not specified the current (local) DB2 server is used. When the owner is not specified the object name is qualified using the current SQLID. When FM/DB2 generates the utility control statements, the owner value is non-blank.

OBJOUT should be used when the fully qualified name fits on a single line in the JCL deck. The last usable column is column 71. When the fully qualified name does not fit on a single line in the JCL deck, use one or more of the OBJOLOCN, OBJOOWNR, OBJONAME keywords to specify the object.

**OBJOLOCN=location**
The optional name of the DB2 remote server (*location*) where the target object is located. See “Specifying a DB2 object name” on page 815.

**OBJOOWNR=owner**
The optional name of the owner of the target object (*owner*) to be copied. See “Specifying a DB2 object name” on page 815.

**OBJONAME=name**
The object name (*name*) for the target object to be copied. See “Specifying a DB2 object name” on page 815.

**TOUTPUT=ddname**
Defines a reference to a DD statement for the data sets which contain the DB2 template that describes the target DB2 object. If you specify a concatenated DD, then you must provide the member name, *member*, via the TOUTMEM keyword. See “Specifying the template for a DB2 object” on page 815.

**TOUTMEM=member**
The name of the template member in the dataset(s) identified by the TOUTPUT parameter, if it has not been specified on the DD statement. This parameter must not be specified if the TMOUT parameter is specified. See “Specifying the template for a DB2 object” on page 815.

**TMOUT=template_data_set_name(member)**
The PDS (*template_data_set_name*) and member name (*member*) of the File
Manager DB2 template that describes the target DB2 object. See “Specifying the template for a DB2 object” on page 815.

UCREAD
Specifies whether FM/DB2 should use uncommitted read, when accessing the source object:
YES Uncommitted read is used. In this case, DB2 takes no locks when accessing the data in the source table, and it is possible that the data for a row changes after being read by FM/DB2, resulting in data inconsistencies.
NO Uncommitted read is not used. This is the default.

LOCK
Specifies whether the source table should be locked, prior to copying data. The possible options are:
NONE The default and recommended value. Any locks taken by DB2 depend on DB2 installation options, and options specified when the source object was created, unless negated by UCREAD=YES.
SHARE FM/DB2 locks the source object in SHARE mode before copying data. This option is only effective for tables, it has no effect if the source object is a view. When an object is locked in SHARE mode, other users may read the object, but no updates by other users are possible.
EXCL FM/DB2 locks the source object in EXCLUSIVE mode before copying the data. This option is only effective for tables, it has no effect if the source object is a view. When an object is locked in EXCLUSIVE mode, no other users may access the object at all, unless using uncommitted read.

DUPKEY
Specifies how FM/DB2 should respond to any duplicate key errors:
IGNORE FM/DB2 ignores the error and continues processing. The row that caused the duplicate key error is not copied to the target DB2 object.
UPDATE FM/DB2 attempts to update the existing row. The columns of the table that are part of the unique index that caused the SQLCODE-803 error are used to identify the row. No comparison is made between the copied row and the existing row prior to the update operation (that is, the update is done regardless of whether the copied row and existing row are the same). If there is more than one unique index defined on the target table, it is possible for an SQLCODE-803 to occur when the update is attempted. This is considered to be an update error.

DUPMAX
The number of duplicate key errors that are allowed before FM/DB2 terminates import processing:
ALL or 0 There is no limit.
num The number of duplicate key errors allowed, up to a maximum of 2147483647.

REFI Specifies whether to ignore errors arising from the violation of a check
constraint (SQLCODE-545). The default is REFI=FAIL, meaning any check constraint error terminates copy processing. Otherwise (REFI=IGNORE) check constraint errors are ignored.

**AUDIT**

Specifies whether FM/DB2 is to write records to the FM/DB2 audit log:

- **YES** Write records to the FM/DB2 audit log. This is the default.
- **NO** Do not write records to the FM/DB2 audit log.

This option may be ignored if an installation audit option has been set that prevents a user modifying the installation setting.

**DELROWS**

Specifies whether FM/DB2 should attempt to delete all rows from the target table, prior to starting the copy operation:

- **YES** Attempt to delete all rows.
- **NO** Do not delete any rows. This is the default.

The deletion of rows from the target table is done using DELETE * FROM <object name> and fails if there are any referential integrity related errors arising from the deletion of rows.

**ROWS**

The maximum number of rows to be copied:

- **ALL** All rows from the source object are written to the target object.
- **num** The maximum number of rows to be copied. Valid range: 1-99999999.

**Examples**

Example 1: Copy the DSN8810.EMP table to a remote location. Delete all rows in the target table before copying the data. Use uncommitted read when accessing DSN8810.EMP. Audit the access to DSN8810.EMP.

```
//DBC JOB (acct),'name'
//* Copy DSN8810.EMP table to the same table at a remote location
//*
//FMNDDB2 EXEC PGM=FMNDDB2,PARM=('SSID=DSN1,SQID=ID1')
//STEPLIB DD DSN=FMN.SFMNMOD1,DISP=SHR
// DD DSN=DB2V810.DSN1.SDSNEXIT,DISP=SHR
// DD DSN=DB2.V810.SDSNLOAD,DISP=SHR
//SYSPRINT DD SYSOUT=*...
//FMNTSPRT DD SYSOUT=*...
//SYSTEM DD SYSOUT=*...
//SYSPRINT DD SYSOUT=*...
//SYSIN DD *
$FILEM DBC OBJIN="DSN8810"."EMP",
$FILEM OBJOUT=REMLOC."DSN8810"."EMP",
$FILEM UREAD=YES,
$FILEM LOCK=NONE,
$FILEM DUPKEY=IGNORE,
$FILEM DUPMAX=ALL,
$FILEM REFI=FAIL,
$FILEM AUDIT=YES,
$FILEM DELROWS=YES,
$FILEM ROWS=ALL
/*
```

Example 2: Copy the DSN8810.EMP table to a DSN8COPY.EMP. Update any rows in the target table that generate duplicate key errors, to a maximum of 100 duplicate key errors.
DBC (Copy) batch command

//DBC JOB (acct), 'name'
//* Copy DSN8810.EMP table to a copy of the same table.
//*FMNDB2 EXEC PGM=FMNDB2,PARM=('SSID=DSN1,SQID=ID1')
//STEPLIB DD DSN=FMN.SFMNMOD1,DISP=SHR
// DD DSN=DB2V810.DSN1.SDSNEXIT,DISP=SHR
// DD DSN=DB2.V810.SDSNLOAD,DISP=SHR
//SYSPRINT DD SYSOUT=* 
//FMNTSPRT DD SYSOUT=* 
//SYSTERM DD SYSOUT=* 
//SYSIN DD *
$$FILEM DBC OBJIN="DSN8810"."EMP",
$$FILEM OBJOUT="DSN8COPY"."EMP",
$$FILEM LOCK=NULL,
$$FILEM DUPKEY=UPDATE,
$$FILEM DUPMAX=100,
$$FILEM REFI=FAIL,
$$FILEM ROWS=ALL
/*

Return codes

The return codes from the DBC function:

0 The function completed successfully
16 A serious error was encountered

Related functions

DBI Import data from a sequential or VSAM file into a DB2 table
DBX Export data from a DB2 object to a sequential or VSAM file
D2G Create (generate) DB2 data

DBI (Import) batch command

Purpose

To import data, in a variety of formats, from a QSAM or VSAM file, into a DB2 object. The DB2 object must be accessible from the currently connected DB2 system.

Usage

The FM/DB2 import utility can:

• Optionally delete all rows from the target table prior to importing any data.
• Limit the number of records to be imported.
• Select the records to be imported.
• Map fields (using template mapping) in the import data set to columns in the target table, with appropriate data type and length conversions.
• Optionally update existing rows, when the values in an imported row’s unique index columns match an existing table row.
DBI (Import) batch command

A REXX procedure can be used during the import to modify mapped and converted data, prior to inserting it into the target DB2 table. Any REXX functions can be used at this point, and records can be dropped, or the import stopped via the REXX proc.

At the end of the import process the following statistics are shown:

- The count of rows successfully imported.
- The number of duplicate row errors.
- Optionally, the number of duplicate rows updated.
- Optionally, the number of duplicate row updates that failed.
- The number of rows in error.
- The number of rows skipped via template selection.
- The number of rows dropped by the REXX proc.

**DBI**

```
DSNIN=data_set_name
VOLSER=volume_serial_number

TCIN=template/copybook_data_set_name(member)
TINPUT=TDDIN
TINPUT=ddname
TINMEM=member

OBJOUT=name
OBJLOCN=location
OBJOWNR=owner
OBJONAME=name

TOUTPUT=TDDOUT
TOUTPUT=ddname
TOUTMEM=member
TMOUT=template_data_set_name(member)

UPDATE=NO
UPDATE=YES

AUTOCOMMIT=0
AUTOCOMMIT=value
STARTPOS=1
STARTPOS=position
DUPMAX=ALL

DELROWS=NO
DELROWS=YES
ROWS=ALL
num

PROC=procname
```

**DSNIN**=data_set_name

The data set name containing the data to be imported. If the data set is partitioned, a member name is required.
DBI (Import) batch command

VOLSER=volume_serial_number
The volume serial number for the input data set. This is only required for uncataloged data sets.

TINPUT=ddname
Defines a reference to a DD statement for the data sets which contains the template member that describes the record structure of your input data. The default is TDDIN.

TINMEM=member
The name of the template member in the dataset(s) identified by the TINPUT parameter, if it has not been specified on the DD statement. This parameter must not be specified if the TCIN parameter is specified.

TCIN=template/copybook_data_set_name(member)
The PDS (template/copybook_data_set_name) and member name (member) of the template or copybook that describes the record structure of the data to be imported. The JCL generated by FM/DB2 specifies TCIN. See “Specifying the template/copybook for a data set” on page 816.

OBJOUT=location.owner.name
The optional name of the DB2 remote server (location) where the target object is located; the optional name of the owner of the target object (owner) and the target object name (name) to be copied.

When location is not specified the current (local) DB2 server is used. When the owner is not specified the object name is qualified using the current SQLID. When FM/DB2 generates the utility control statements, the owner value is non-blank.

OBJOUT should be used when the fully qualified name fits on a single line in the JCL deck. The last usable column is column 71. When the fully qualified name does not fit on a single line in the JCL deck, use one or more of the OBJOLOCN, OBJOOWNR, OBJONAME keywords to specify the object.

OBJOLOCN=location
The optional name of the DB2 remote server (location) where the target object is located. See “Specifying a DB2 object name” on page 815.

OBJOOWNR=owner
The optional name of the owner of the target object (owner) for the import. See “Specifying a DB2 object name” on page 815.

OBJONAME=name
The object name (name) for the target object for the import. See “Specifying a DB2 object name” on page 815.

TOUTPUT=ddname
Defines a reference to a DD statement for the data sets which contain the DB2 template that describes the target DB2 object. If you specify a concatenated DD, then you must provide the member name, member, via the TOUTMEM keyword. See “Specifying the template for a DB2 object” on page 815.

TOUTMEM=member
The name of the template member in the dataset(s) identified by the TOUTPUT parameter, if it has not been specified on the DD statement. This parameter must not be specified if the TMOUT parameter is specified. See “Specifying the template for a DB2 object” on page 815.
**DBI (Import) batch command**

**TMOUT=**

The PDS (template_data_set_name) and member name (member) of the File Manager DB2 template that describes the target DB2 object. See [“Specifying the template for a DB2 object” on page 815](#).

**UPDATE**

Specifies whether FM/DB2 should attempt to update any rows that generate SQLCODE-803 (duplicate key) errors:

**YES** FM/DB2 attempts to update the existing row. All columns of the table that are part of at least 1 unique index are used to identify the row. No comparison is made between the imported row and the existing row prior to the update operation (that is, the update is done regardless of whether the imported row and existing row are the same). When using this option it is recommended that a DB2 table name, rather than a view name, be specified. FM/DB2 does not attempt to update existing rows in a view defined on a view.

**NO** No rows are updated. This is the default.

**AUTOCOMMIT=**

Specifies a numeric value indicating how often FM/DB2 should issue a DB2 commit during the import. The default is zero, meaning that no DB2 commits are issued during the import operation. Any positive value results in FM/DB2 issuing a DB2 commit after value successful changes are made during the import. A successful change occurs when either a record is successfully inserted, or when a row is successfully updated after the attempted insert of the record resulted in a duplicate key error. The counter is reset after each DB2 commit is issued, therefore multiple DB2 commits may be issued. If an error occurs during the import and one or more DB2 commits have been issued, only the changes made subsequent to the last DB2 commit point are backed out.

**STARTPOS=**

Specifies a numeric value indicating the starting record for the import. The default is 1, meaning FM/DB2 starts the import with the first record of the import file. Any value greater than 1 causes FM/DB2 to skip position-1 records before commencing the import.

**DUPMAX**

The number of duplicate key errors that are allowed before FM/DB2 terminates import processing:

**ALL or 0**

There is no limit.

**num** The number of duplicate key errors allowed, up to a maximum of 2147483647.

**DELROWS**

Specifies whether FM/DB2 should attempt to delete all rows from the target table, prior to starting the copy operation:

**YES** Attempt to delete all rows.

**NO** Do not delete any rows. This is the default.

The deletion of rows from the target table is done using DELETE * FROM <object name> and fails if there are any referential integrity related errors arising from the deletion of rows.

**ROWS**

The maximum number of records to be imported:
DBI (Import) batch command

**ALL**  All records from the import data set are inserted into the target object.

**num**  The maximum number of records to be imported. Valid range: 1-99999999.

**PROC**  REXX procedure:

```
procname
```

The name of the REXX procedure to be used when processing import records.

```
*  An inline procedure. The * is followed by one or more REXX procedure statements, and the terminating "/+.
```

**Examples**

Example 1: Import data from a sequential file into DSN8810.EMP. Delete all rows in DSN8810.EMP before importing the data.

```
//DBI JOB (acct),'name'
//* Import data from a sequential file into DSN8810.EMP.
//* Delete all the rows from the table before importing the data.
//*
//FMNDB2 EXEC PGM=FMNDB2,PARM=('SSID=DSN1,SQID=ID1')
//STEPLIB DD DSN=FMN.SFMNMOD1,DISP=SHR
// DD DSN=DB2V810.DSN1.SDSNEXIT,DISP=SHR
// DD DSN=DB2.V810.SDSNLOAD,DISP=SHR
//SYSPRINT DD SYSOUT=*  
//FMNTSPRT DD SYSOUT=*  
//SYSTEAM DD SYSOUT=*
//SYSIN DD *  
$$FILEM DBI DSNIN=ID1.EXPORT.EMP.EXAMPLE1,  
$$FILEM TCIN=ID1.FM.TEMPLATE(EMP),  
$$FILEM OBJOUT="DSN8810"."EMP",  
$$FILEM UPDATE=NO,  
$$FILEM DUPMAX=ALL,  
$$FILEM DELROWS=YES,  
$$FILEM ROWS=ALL  
/*
```

Example 2: Import data from a sequential file into DSN8810.EMP. Use REXX proc PROC1 to process each record in the sequential file before inserting the record. Update any rows that cause duplicate key errors, to a maximum of 50 duplicate key errors.

```
//DBI JOB (acct),'name'
//* Import data from a sequential file into DSN8810.EMP.
//*
//FMNDB2 EXEC PGM=FMNDB2,PARM=('SSID=DSN1,SQID=ID1')
//STEPLIB DD DSN=FMN.SFMNMOD1,DISP=SHR
// DD DSN=DB2V810.DSN1.SDSNEXIT,DISP=SHR
// DD DSN=DB2.V810.SDSNLOAD,DISP=SHR
//SYSPRINT DD SYSOUT=*  
//FMNTSPRT DD SYSOUT=*  
//SYSTEAM DD SYSOUT=*
//FMNEXEC DD DSN=ID1.EXEC,DISP=SHR
//SYSIN DD *  
$$FILEM DBI DSNIN=ID1.EXPORT.EMP.EXAMPLE1,  
$$FILEM TCIN=ID1.FM.TEMPLATE(EMP),  
$$FILEM OBJOUT="DSN8810"."EMP",  
$$FILEM UPDATE=YES,  
$$FILEM DUPMAX=50,  
$$FILEM DELROWS=NO,  
$$FILEM ROWS=ALL,  
$$FILEM PROC=PROC1  
/*
```
**DBI (Import) batch command**

**Return codes**

The return codes from the DBI function:

- 0 The function completed successfully
- 16 A serious error was encountered

**Related functions**

- **DBC** Copy data from one DB2 object to another ("DBC (Copy) batch command" on page 788)
- **DBX** Export data from a DB2 object to a sequential or VSAM file ("DBX (Export) batch command" on page 800)
- **D2G** Create (generate) DB2 data ("D2G (Data Generate) batch command" on page 810)

**DBP (Print) batch command**

**Purpose**

To print the contents of a DB2 object in a variety of formats.

**Usage**

The FM/DB2 Print utility function can:

- Print data in table or single formats, in either character or HEX.
- Select the rows and columns to be printed.
- Limit the number of rows to be printed.

```plaintext
DBP FORMAT= TABLE
      SINGLE

OBJIN= name
    owner.
    location.

OBJNAME= name

OBJIOWNR= owner

OBJILOCN= location

TINPUT=DDIN

TINPUT= ddname

TINMEM= member

TIN= template_data_set_name(member)

NULLIND= character

ROWS= ALL
```

**FORMAT**

The formatting mode for the printed data:

- **TABLE**
  In table mode each row of the table occupies a single line, and columns appear left to right across the page.
SINGLE
In single mode each row of the table occupies multiple print lines.
The first column of the object is printed on the first, and possibly
more lines, followed by the second column and so on.

OBJIN=location.owner.name
The optional name of the DB2 remote server (location) where the object
is located; the optional name of the owner of the object (owner) and the
object name (name) to be printed.

When location is not specified the current (local) DB2 server is used. When
the owner is not specified the object name is qualified using the current
SQLID. When FM/DB2 generates the utility control statements, the owner
value is non-blank.

OBJIN should be used when the fully qualified name fits on a single line
in the JCL deck. The last usable column is column 71. When the fully
qualified name does not fit on a single line in the JCL deck, use one or
more of the OBJILOCN, OBJIOWNR, OBJINAME keywords to specify the
object.

OBJILOCN=location
The optional name of the DB2 remote server (location) where the object
to be printed is located. See “Specifying a DB2 object name” on page 815.

OBJIOWNR=owner
The optional name of the owner of the object (owner) to be printed. See
“Specifying a DB2 object name” on page 815.

OBJINAME=name
The object name (name) for the object to be printed. See “Specifying a DB2
object name” on page 815.

TINPUT=ddname
Defines a reference to a DD statement for the data sets which contain the
DB2 template that describes the source DB2 object to be printed. If you
specify a concatenated DD, then you must provide the member name,
member, via the TINMEM keyword. See “Specifying the template for a DB2
object” on page 815.

TINMEM=member
The name of the template member in the dataset(s) identified by the
TINPUT parameter, if it has not been specified on the DD statement. This
parameter must not be specified if the TMIN parameter is specified. See
“Specifying the template for a DB2 object” on page 815.

TMIN=template_data_set_name(member)
The PDS (template_data_set_name) and member name (member) of the File
Manager DB2 template that describes the source DB2 object to be copied.
See “Specifying the template for a DB2 object” on page 815.

UCREAD
Specifies whether FM/DB2 should use uncommitted read, when accessing
the source object:

YES Uncommitted read is used. In this case, DB2 takes no locks when
accessing the data in the source table, and it is possible that the
data for a row changes after being read by FM/DB2, resulting in
data inconsistencies.

NO Uncommitted read is not used. This is the default.
DBP (Print) batch command

HEX=YES
Specifies that FM/DB2 should print the data in hexadecimal format. Omit this option if normal printing is required.

NULLIND=character
Defines the character used to indicate that the value for a column is the DB2 null value, in the printed data. The default is the underscore character. The value for this character is set using the editor Null Column Indicators: Display option. When specified in the File Manager utility control statements, the character must be a printable character other than a comma. The value cannot be enclosed in quotes.

ROWS
The maximum number of rows to be printed:
- ALL All rows from the source object are printed.
- num The maximum number of rows to be printed. Valid range: 1-99999999.

Examples

Example 1: Print data from DSN8810.EMP, in single mode. Use uncommitted read when accessing the DB2 data. Print the data in hexadecimal format.

```
//DBP JOB (acct),'name'
//* Print table DSN8810.EMP in SINGLE format.
//*
//FMNDB2 EXEC PGM=FMNDB2,PARM=('SSID=DSN1,SQID=ID1')
//STEPLIB DD DSN=FMN.SFMNMOD1,DISP=SHR
// DD DSN=DB2V810.DSN1.SDSNEXIT,DISP=SHR
// DD DSN=DB2.V810.SDSNLOAD,DISP=SHR
//SYSPRINT DD SYSOUT=*  
//FMNTSPRT DD SYSOUT=*  
//SYSTERM DD SYSOUT=*  
//SYSIN DD *
$$FILEM DBP FORMAT=SINGLE, 
$$FILEM OBJIN="DSN8810"."DEPT", 
$$FILEM UCREAD=YES, 
$$FILEM HEX=YES, 
$$FILEM NULLIND=_, 
$$FILEM ROWS=ALL 
/*
```

Return codes

The return codes from the DBP function:
- 0 The function completed successfully
- 16 A serious error was encountered

Related functions

None.

DBX (Export) batch command

Purpose

To export data from a DB2 object (table, view, synonym or alias) or the result table from a SELECT statement, in any local or distributed DB2 system, to a sequential or VSAM file.
**DBX (Export) batch command**

**Usage**

The FM/DB2 Export Utility can export data from a DB2 table or view using one of:

**File Manager**
- The output data set can be a sequential data set, a partitioned data set or a VSAM file.

**The DB2 UNLOAD utility**
- The output data set is constrained by the DB2 UNLOAD utility to be a sequential data set. Any reformatting of the output data is performed by the DB2 UNLOAD utility. Therefore some template mapping and re-formatting options that are available when using File Manager are not available.

**The DB2 DSNTIAUL sample program**
- The output data set is constrained by the DB2 DSNTIAUL utility to be a sequential data set. Any reformatting of the output data is performed by the DB2 DSNTIAUL sample program utility. Therefore some template mapping and re-formatting options that are available when using File Manager are not available.

When exporting large volumes of data, the DB2 UNLOAD utility offers superior performance when compared with File Manager, or the DSNTIAUL sample program.

You can choose a pre-defined (standard) export data format, or specify the data format using a copybook or template. You can specify that File Manager is to export the data, or use a DB2 utility or sample program to export the data. For a user-defined data format you can specify how the DB2 null indicator is represented (either one byte or two byte) and where the null indicator is located in the exported data. You can also specify that the data is to be exported in delimited format (comma separated variables).
Null (null indicator):

- NIUSAGE=None
- NIUSAGE=SEPARATE

Placement:

- PLACE=BEFORE
- PLACE=AFTER
- PLACE=USER

Null type:

- TYPE=CHAR
  - NICHAR=character
- TYPE=INTEGER
  - NIINTEGER=signed_integer

Data type:

- DECIMAL=INTERNAL
- DECIMAL=ZONED
- DECIMAL=EXTERNAL
- INTEGER=NO
- INTEGER=EXTERNAL
- FLOAT=NO
- FLOAT=EXTERNAL

SQLDD=ddname

- ddname is the DDNAME, specified in the JCL, that identifies either a data set containing an SQL SELECT statement to be processed, or a DDNAME used to specify an in-line SQL SELECT statement.

SQLRECSZ=nn

- nn is the usable record size for the SQL statement. For in-line SQL statements, specified in the JCL, use a value of 72, to avoid problems with sequence numbers in columns 73-80 inclusive.

- For a data set, set an appropriate value.

OBJIN=location.owner.name

- The optional name of the DB2 remote server (location) where the object is located; the optional name of the owner of the object (owner) and the object name (name) to be exported.
DBX (Export) batch command

When location is not specified the current (local) DB2 server is used. When the owner is not specified the object name is qualified using the current SQLID. When FM/DB2 generates the utility control statements, the owner value is non-blank.

OBJIN should be used when the fully qualified name fits on a single line in the JCL deck. The last usable column is column 71. When the fully qualified name does not fit on a single line in the JCL deck, use one or more of the OBJILOCN, OBJIOWNR, OBJINAME keywords to specify the object.

OBJILOCN=location
The optional name of the DB2 remote server (location) where the object to be exported is located. See “Specifying a DB2 object name” on page 815.

OBJIOWNR=owner
The optional name of the owner of the object (owner) to be exported. See “Specifying a DB2 object name” on page 815.

OBJINAME=name
The object name (name) for the object to be exported. See “Specifying a DB2 object name” on page 815.

TINPUT=ddname
Defines a reference to a DD statement for the data sets which contain the DB2 template that describes the DB2 object to be exported. If you specify a concatenated DD, then you must provide the member name, member, via the TINMEM keyword. See “Specifying the template for a DB2 object” on page 815.

TINMEM=member
The name of the template member in the dataset(s) identified by the TINPUT parameter, if it has not been specified on the DD statement. This parameter must not be specified if the TMIN parameter is specified. See “Specifying the template for a DB2 object” on page 815.

TMIN=template_data_set_name(member)
The PDS (template_data_set_name) and member name (member) of the File Manager DB2 template that describes the DB2 object to be exported. See “Specifying the template for a DB2 object” on page 815.

OUTPUT=ddname
Defines a reference to a DD statement for the export data set. This keyword and the DSNOUT keyword are mutually exclusive. The JCL generated by FM/DB2 specifies OUTPUT=FMNOUT in the File Manager utility control statements, and an appropriate DD statement in the JCL.

DSNOUT=dsname
Defines the name of the export data set or an absolute path to the export HFS file (directory). This keyword and the OUTPUT keyword are mutually exclusive. The name may include a member name in parenthesis. An absolute path to an HFS file (directory) must be enclosed in apostrophes. If the data set name does not fit on one line, you can split it over more than one line.

TOUTPUT=ddname
Defines a reference to a DD statement for the data sets which contain the template member that describes the record structure of your output data. The default is TDDOUT.
TOUTMEM=\textit{member}

The name of the template member in the dataset(s) identified by the TOUTPUT parameter, if it has not been specified on the DD statement. This parameter must not be specified if the TCOUT parameter is specified.

TCOUT=\textit{template/copybook\_data\_set\_name}(\textit{member})

The PDS (\textit{template/copybook\_data\_set\_name}) and member name (\textit{member}) of the template or copybook that describes the record structure of the exported data. The JCL generated by FM/DB2 specifies TCOUT, when an output copybook or template is required. See "Specifying the template/copybook for a data set" on page 816.

NULLIND=\textit{character}

For CSV format only, defines the character used to indicate that the value for a column is the DB2 null value, in the exported data. The default is the underscore character. The value for this character is set using the editor Null Column Indicators: Display option. When specified in the File Manager utility control statements, the character must be a printable character other than a comma. The value cannot be enclosed in quotes.

NULLIND is always specified in the File Manager utility control statements. The value specified is ignored for all DATAFORMAT specifications other than CSV.

ROWS

The number of records to be exported:

\textbf{ALL} All records selected from the object are written to the export data set.

\textit{num} The maximum number of records to be written to the export data set. Valid range: 1-99999999.

NATIVE

Specifies whether FM/DB2 should process Unicode encoded data natively when exporting data from a Unicode encoded table. This option can be ignored if the default processing of Unicode data is desired; that is, Unicode data is potentially substituted with error characters when copying data between Unicode tables.

\textbf{YES} Process Unicode data natively.

\textbf{NO} Do not Process Unicode data natively. This is the default.

CSV

 Specifies whether the exported data is to be in delimited variables format or not:

\textbf{NO} Export data is not to be in delimited format. This is the default.

\textbf{YES} Export data is to be in delimited format.

DATAFORMAT

The data format for the exported data:

\textbf{SQLDA}

The default. Any nullable column is preceded by a 2-byte binary field containing the null indicator. This field contains -1 if the column has the null value and 0 otherwise. Any decimal field occupies 16 bytes, with the data left-justified in the field and trailing binary zeros. Any graphic or vargraphic field is followed by two bytes containing binary zeros.

\textbf{UNLOAD}

In this format any nullable column is preceded by a 1-byte field containing the null indicator. This field contains hexadecimal "FF" if the column had a null value and 0 otherwise. Any decimal field occupies the minimum number of bytes needed to represent the
data as a packed decimal field. Variable length columns in the unloaded records are padded to their maximum length and the padded data fields are preceded by two-byte length fields that indicate the size of the actual data without the padding. CCSID conversion is not supported. The exported data is in the CCSID used to bind the FM/DB2 plan.

**DSNTIAUL**
Any nullable column is followed by a 1-byte field containing the null indicator. This field contains "?" if the column had a null value. Any decimal field occupies the minimum number of bytes needed to represent the data as a packed decimal field.

**USER**
The data format is defined by a user-specified copybook or template and template mapping. Alternatively the data format can be defined using the NIASUSAGE, PLACE, TYPE (NICHAR, NIINTEGER), DECIMAL, INTEGER, FLOAT, and SOSI keywords.

**CSV**
The data is exported as delimited variables. The default delimiter is the comma, the SEPARATOR keyword can be used to specify a different delimiter. A copybook/template cannot be used to describe the delimited variables data format.

**CSVHDR**
Specifies whether column headers are to be produced on the first record of the export file when CSV=YES has been specified. Possible values are:
- **NO** Column headers not produced.
- **YES** Column headers produced.

**SEPARATOR='character'**
The character to be used as the delimiter, for delimited variables (CSV) format. The default is the comma. You can specify any printable characters in the control statements. To specify non-printable characters use NIASUSAGE=CSV and then define the delimiter using the TYPE, NICHAR or NIINTEGER keywords.

**PLACE**
The location of any DB2 null indicator fields in the exported data:
- **BEFORE** The field for the DB2 null indicator appears immediately before the data for the column.
- **AFTER** The field for the DB2 null indicator appears immediately after the data for the column.
- **USER** The field for the DB2 null indicator is defined using a copybook or template, with template mapping.

**NIUSAGE**
Specifies, for CSV=NO, whether or not null indicators appear in the exported data. NIASUSAGE=NONE or NIASUSAGE=SEPARATE should only be specified with CSV=NO and DATAFORMAT= USER.
- **NONE** The exported data is not to contain any DB2 null indicators. Using this option with a DB2 object that has nullable columns results in a loss of information.
- **SEPARATE** Specifies, for CSV=YES, that the TYPE, NICHAR and NIINTEGRER keywords define the delimiter characters(s). NIASUSAGE=CSV should only be specified with CSV=YES and DATAFORMAT=CSV.
- **CSV** Indicates that the TYPE and NICHAR/NIINTEGRER keywords define a one or two character delimiter, to be used with
DBX (Export) batch command

DATAFORMAT=CSV only. This option should be used when the delimited variable's separator character (default is comma) cannot be used, and a suitable alternative cannot be specified using a single displayable character. Either a one byte or two byte delimiter can be specified by setting the TYPE keyword (see below) to the appropriate value. A one byte indicator is interpreted as a character, and therefore acts in a similar way to a one character value specified directly using the SEPARATOR keyword. A two byte indicator value is interpreted as a two byte signed integer, allowing a two byte, non-displayable value to be entered.

TYPE=CHAR/INTEGER
The data format for the null indicator fields, or for the CSV delimiter. The null indicator/delimiter can be either a one byte (character) or two byte (signed integer) field. The character or integer used to indicate the DB2 null value/CSV delimiter can be specified using the NICCHAR (TYPE=CHAR), or NIINTEGER (TYPE=INTEGER) keywords.

NICCHAR=character
The character used to indicate that a column has the DB2 null value, or the CSV delimiter character. NICCHAR should only be specified with TYPE=CHAR. You can specify any printable character other than comma in the control statements.

NIINTEGER=signed_integer
The integer used to indicate that a column has the DB2 null value, or a two byte CSV delimiter. NIINTEGER should only be specified with TYPE=INTEGER. You can specify any signed integer in the range -32767 - 32768 inclusive.

DECIMAL
Specifies the data format for any decimal columns in the exported data. Possible values are:
- INTERNAL
  Decimal data appears in internal packed decimal format.
- ZONED
  Decimal data appears as a string of zoned decimal digits.
- EXTERNAL
  Decimal data appears as a string of characters, the string represents the decimal number, for example '3.1415'.

INTEGER
Specifies the data format for any integer columns in the exported data. An integer column can be a SMALLINT, INTEGER or BIGINT. Possible values are:
- NO
  Integer data appears in internal format, occupying 2 bytes (SMALLINT), 4 bytes (INTEGER) or 8 bytes (BIGINT).
- EXTERNAL
  Integer data appears as a string of characters, the string represents the integer, for example, '42', '1000000'.

FLOAT
Specifies the data format for any floating point columns in the exported data. A floating point column is a REAL, FLOAT, DOUBLE or DECFLOAT column. Possible values are:
- NO
  The floating point data appears in internal format.
- EXTERNAL
  Floating point data appears as a string of characters, the string represents the number, for example '-1.0000000000000000E+00'.
DBX (Export) batch command

**SOSI** Specifies whether the data for graphic data type columns (GRAPHIC, VARGRAPHIC) should be enclosed in shift-out/shift-in (sosi) characters.

Possible values are:

**NO** The graphics data is not enclosed in sosi characters.

**YES** The graphics data is enclosed in sosi characters.

**Examples**

Example 1: Export data from DB2 sample table DSN8810.EMP to a sequential file, using SQLDA format. The output data set is allocated in the batch job.

```sql
//DBX JOB (acct), 'name'
// * Export DSN8810.EMP table to a data set
// *
/FANDARD EXEC PGM=FMNDB2,PARM=('SSID=DSN1,SQID=ID1')
//STELIB DD DSN=FMN.SFMNMOD1,DISP=SHR
//    DD DSN=DB2V810.DSN1.SDSNEXIT,DISP=SHR
//    DD DSN=DB2.V810.SDSNLOAD,DISP=SHR
//SYSPRINT DD SYSOUT=*
//FMNOUT DD SYSOUT=*
//SYMTSR DD SYSOUT=*
//FILEM DBX OBJIN="DSN8810"."EMP",
//FILEM OUTPUT=FMNOUT,
//FILEM NULLIND=_,
//FILEM CSV=NO, _
//FILEM DATAFORMAT=FMSQLDA,
//FILEM ROWS=ALL
/*
```

Example 2: Export data from DB2 sample table DSN8810.EMP to a sequential file, using DB2 UNLOAD format. The output data set is allocated in the batch job.

```sql
//DBX JOB (acct), 'name'
// * Export DSN8810.EMP table to a data set
// *
/FANDARD EXEC PGM=FMNDB2,PARM=('SSID=DSN1,SQID=ID1')
//STELIB DD DSN=FMN.SFMNMOD1,DISP=SHR
//    DD DSN=DB2V810.DSN1.SDSNEXIT,DISP=SHR
//    DD DSN=DB2.V810.SDSNLOAD,DISP=SHR
//SYSPRINT DD SYSOUT=*
//FMNOUT DD SYSOUT=*
//SYMTSR DD SYSOUT=*
//FILEM DBX OBJIN="DSN8810"."EMP",
//FILEM OUTPUT=FMNOUT,
//FILEM NULLIND=_,
//FILEM CSV=NO, _
//FILEM DATAFORMAT=UNLOAD,
//FILEM ROWS=ALL
/*
```

Example 3: Export data from DB2 sample table DSN8810.EMP to a sequential file, using DB2 DSNTIAUL format. The output data set is allocated in the batch job.

```sql
Example 4: Export data from a table with a long name containing spaces to a sequential file, using SQLDA format. The output data set is pre-allocated.

Example 5: Export data from DB2 sample table DSN8810.EMP to a sequential file, using a user-defined data format. The output data set is pre-allocated. In the user-defined data format:

- Null indicators are used (NIUSAGE=SEPARATE)
- Null indicators appear immediately after the data for the column (PLACE=AFTER)
- The null indicator is a two byte (signed integer) field (TYPE= INTEGER)
- The value used to indicate a column has the DB2 null value is -55.
- Any DECIMAL data type columns appear in external format (for example '3.1414')
- Any INTEGER data type columns appear in external format (for example '1234')
- Any floating point data type columns appear in external format.
Example 6: Export data from DB2 sample table DSN8810.EMP to a sequential file, using a CSV (delimited variables) format. The delimiter character is the ‘!’ character. The output data set is pre-allocated.

Example 7: Export data from DB2 sample table DSN8810.EMP to a sequential file, using a CSV (delimited variables format). The delimiter character is x'FFFF', defined using the NIUSAGE, TYPE and NIINTEGRER keywords. The output data set is pre-allocated.
DBX (Export) batch command

```plaintext
//FMNOUT DD DISP=MOD, DSN=ID1.EXPORT.EMP.CSV2
//SYSIN DD *
$$FILEM DBX OBJIN="DSN8910"."EMP",
$$FILEM OUTPUT=FMNOUT,
$$FILEM NULLIND=_,
$$FILEM CSV=YES,
$$FILEM DATAFORMAT=CSV,
$$FILEM NIUSAGE=CSV,
$$FILEM TYPE=INTEGER,
$$FILEM NIINTEGER=-1,
$$FILEM ROWS=ALL */
```

Return codes

The return codes from the DBX function:

0    The function completed successfully
16   A serious error was encountered

Related functions

- DBI Import data from a sequential or VSAM file into a DB2 table
- DBC Copy data from one DB2 object to another
- D2G Create (generate) DB2 data

D2G (Data Generate) batch command

**Purpose**

To load a DB2 object with test data.

**Usage**

The FM/DB2 Data Create utility function can:

- Limit the number of rows to be created.
- Specify how, on a column-by-column basis, each row is be initialized.
D2G (Data Generate) batch command

OBJOUT location.owner.object
The optional name of the DB2 remote server (location) where the object is
located; the optional name of the owner of the object (owner) and the
object name (name).

When location is not specified the current (local) DB2 server is used. When
the owner is not specified the object name is qualified using the current
SQLID. When FM/DB2 generates the utility control statements, the owner
value is non-blank.

OBJOUT should be used when the fully qualified name fits on a single line
in the JCL deck. The last usable column is column 71. When the fully
qualified name does not fit on a single line in the JCL deck, use one or
more of the OBJOLOCN, OBJOOWNR, OBJONAME keywords to specify
the object.

OBJOLOCN=location
The optional name of the DB2 remote server (location) where the object is
located. See “Specifying a DB2 object name” on page 815.

OBJOOWNR=owner
The optional name of the owner of the object (owner). See “Specifying a
DB2 object name” on page 815.

OBJONAME=name
The object name (name) for the object. See “Specifying a DB2 object name”
on page 815.

TMOUT=template_data_set_name(member)
The PDS (template_data_set_name) and member name (member) of the File
Manager DB2 template that describes the DB2 object where the data is to
be created. See “Specifying the template for a DB2 object” on page 815. The
JCL generated by FM/DB2 always specifies TMOUT.

TOUTPUT=ddname
Defines a reference to a DD statement for the data sets which contain the
template member that describes the record structure of your output data.
The default is TDDOUT.

TOUTMEM=member
The name of the template member in the dataset(s) identified by the
TOUTPUT parameter, if it has not been specified on the DD statement.
This parameter must not be specified if the TCOUT parameter is specified.

ROWS=num
The number of rows to be created. Valid range 1-99999999.

Examples

Example 1: Create 1000 rows of test data in table "ID1"."TABLE1". The initialization
information for each column of the table are defined in template EMPDG.

//D2G JOB (acct),'name'
//* Create rows in table ID2.TABLE1.
//*
//FMNDB2 EXEC PGM=FMNDB2,PARM=('SSID=DSN1,SQID=ID1')
//STEP1B DD DSN=FMN.SFMNMOD1,DISP=SHR
// DD DSN=DB2V810.DSN1.SDSNEXIT,DISP=SHR
// DD DSN=DSN=DB2V810.DSN1.SDSNLOAD,DISP=SHR
//SYSPRINT DD SYSOUT=*  
//FMNTPRT DD SYSOUT=*  
//SYSTEMP DD SYSOUT=*  
//SYSIN DD *

Chapter 17. FM/DB2 functions  811
D2G (Data Generate) batch command

Return codes

The return codes from the D2G function:

0    The function completed successfully
16   A serious error was encountered

Related functions

DBC   Copy data from one DB2 object to another "DBC (Copy) batch command" on page 788
DBI   Import data from a sequential or VSAM file into a DB2 table "DBI (Import) batch command" on page 793
DBX   Export data from a DB2 object to a sequential or VSAM file "DBX (Export) batch command" on page 800
D2G   Create (generate) DB2 data "D2G (Data Generate) batch command" on page 810

D2TP (Template create/update) batch command

Purpose

To create or update a template based on one DB2 object. The object must be accessible from the currently connected DB2 system.

Usage

The FM/DB2 template create/update utility creates a template if it does not exist, or updates the template if it does exist.

OBJIN location.owner.object

The optional name of the DB2 remote server (location) where the source object is located; the optional name of the owner of the source object (owner) and the source object name (name).
D2TP (Template create/update) batch command

When location is not specified the current (local) DB2 server is used. When the owner is not specified the object name is qualified using the current SQLID. When FM/DB2 generates the utility control statements, the owner value is non-blank.

OBJIN should be used when the fully qualified name fits on a single line in the JCL deck. The last usable column is column 71. When the fully qualified name does not fit on a single line in the JCL deck, use one or more of the OBJILOCN, OBJIOWNR, OBJINAME keywords to specify the object.

OBJILOCN=location
The optional name of the DB2 remote server (location) where the source object is located. See “Specifying a DB2 object name” on page 815.

OBJIOWNR=owner
The optional name of the owner of the source object (owner). See “Specifying a DB2 object name” on page 815.

OBJINAME=name
The object name (name) for the source object. See “Specifying a DB2 object name” on page 815.

TOUTPUT=ddname
Defines a reference to a DD statement for the data sets which contain the DB2 template that describes the target DB2 object. Concatenated DD statements are not supported and the referenced data set must be catalogued. If you have not specified a member name in the referenced DD statement, then you must provide a TOUTMEM keyword. If no TOUTPUT or TMOUT parameter has been provided, then TOUTPUT=TDDOUT is used. See “Specifying the template for a DB2 object” on page 815.

TOUTMEM=member
The name of the template member in the dataset identified by the TOUTPUT or TMOUT parameter. This parameter is ignored if the member name is provided with the DD statement or the TMOUT parameter. See “Specifying the template for a DB2 object” on page 815.

TMOUT=template_data_set_name(member)
The PDS (template_data_set_name) and member name (member) of the File Manager DB2 template that describes the target DB2 object. See “Specifying the template for a DB2 object” on page 815.

Examples

Example 1: Create a DB2 template for the DSN8810.EMP using TMOUT keyword.
//D2TPJOB (acct),'name'
//* Create template FMN.TEMPLATE(EMP) for DSN8810.EMP table
//*
  //FMNDB2 EXEC PGM=FMNDB2,PARM=('SSID=DSN1,SQID=ID1')
  //STEPLIB DD DSN=FMN.SFMNMOD1,DISP=SHR
  // DD DSN=DB2V810.DSN1.SDSNEXIT,DISP=SHR
  // DD DSN=DB2.V810.SDSNLOAD,DISP=SHR
  //SYSPRINT DD SYSOUT=* 
  //FMNTSPT DD SYSOUT=* 
  //SYSTERM DD SYSOUT=* 
  //SYsin DD *
  $$FILEM D2TP OBJIN='DSN8810'."EMP",
  $$FILEM TMOUT=FMN.TEMPLATE(EMP)
/*
D2TP (Template create/update) batch command

Example 2: Use TOUTPUT, TOUTMEM keywords to create output template for DSN8810.EMP

//D2TPJOB (acct), 'name'
/** Create template FMN.TEMPLATE(EMP) for DSN8810.EMP table
/**
//FMNDB2 EXEC PGM=FMNDB2,PARM=('SSID=DSN1,SQID=ID1')
//STEPLIB DD DSN=FMN.SFMNMOD1,DISP=SHR
// DD DSN=DB2V810.DSN1.SDNSEXIT,DISP=SHR
// DD DSN=DB2.V810.SDSNLOAD,DISP=SHR
//TOUT DD DSN=FMN.TEMPLATE,DISP=SHR
//SYSPRINT DD SYSOUT=* 
//FMNTRPRT DD SYSOUT=* 
//SYSTERM DD SYSOUT=* 
//SYSIN DD * 
$$FILEM D2TP OBJIN="DSN8810"."EMP", 
$$FILEM TOUTPUT=TOUT,TOUTMEM=EMP 
/**

Example 3: Use Default output DD TDDOUT with TOUTMEM to create output template for DSN8810.EMP

//D2TPJOB (acct), 'name'
/** Create template FMN.TEMPLATE(EMP) for DSN8810.EMP table
/**
//FMNDB2 EXEC PGM=FMNDB2,PARM=('SSID=DSN1,SQID=ID1')
//STEPLIB DD DSN=FMN.SFMNMOD1,DISP=SHR
// DD DSN=DB2V810.DSN1.SDNSEXIT,DISP=SHR
// DD DSN=DB2.V810.SDSNLOAD,DISP=SHR
//TDDOUT DD DSN=FMN.TEMPLATE,DISP=SHR
//SYSPRINT DD SYSOUT=* 
//FMNTRPRT DD SYSOUT=* 
//SYSTERM DD SYSOUT=* 
//SYSIN DD * 
$$FILEM D2TP OBJIN="DSN8810"."EMP", 
$$FILEM TOUTMEM=EMP 
/**

Example 4: Use Default output DD TDDOUT with member to create output template for DSN8810.EMP

//D2TPJOB (acct), 'name'
/** Create template FMN.TEMPLATE(EMP) for DSN8810.EMP table
/**
//FMNDB2 EXEC PGM=FMNDB2,PARM=('SSID=DSN1,SQID=ID1')
//STEPLIB DD DSN=FMN.SFMNMOD1,DISP=SHR
// DD DSN=DB2V810.DSN1.SDNSEXIT,DISP=SHR
// DD DSN=DB2.V810.SDSNLOAD,DISP=SHR
//TDDOUT DD DSN=FMN.TEMPLATE(EMP),DISP=SHR
//SYSPRINT DD SYSOUT=* 
//FMNTRPRT DD SYSOUT=* 
//SYSTERM DD SYSOUT=* 
//SYSIN DD * 
$$FILEM D2TP OBJIN="DSN8810"."EMP" 
$$FILEM TOUTMEM=EMP 
/**

Example 5: Create 3 templates

//D2TPJOB (acct), 'name'
/** Create template TOUT1-3 for DSN8810.EMP table
/**
//FMNDB2 EXEC PGM=FMNDB2,PARM=('SSID=DSN1,SQID=ID1')
//STEPLIB DD DSN=FMN.SFMNMOD1,DISP=SHR
// DD DSN=DB2V810.DSN1.SDNSEXIT,DISP=SHR
// DD DSN=DB2.V810.SDSNLOAD,DISP=SHR
//TOUT1 DD DSN=FMN.TEMPLATE(T1),DISP=SHR
//TOUT2 DD DSN=FMN.TEMPLATE(T2),DISP=SHR
//TOUT3 DD DSN=FMN.TEMPLATE(T3),DISP=SHR
Specifying a DB2 object name

For those batch functions that require a DB2 object name, FM/DB2 automatically determines the appropriate keywords to specify the DB2 object(s). For an input DB2 object this is OBJIN, or OBJILOCN (for location), OBJIOWNR (for owner) and OBJINAME (for name). For an output DB2 object this is OBJOUT, or OBJOLOCN (for location), OBJOOWNR (for owner) and OBJONAME (for name). The latter values are used when the OBJIN/OBJOUT value does not fit on a single line, or when the owner or name values contain special characters, for example, lower case letters. In what follows, a special character is any character other than the upper case letters A-Z, the digits 0-9, the 3 characters reserved as alphabetic extenders for national languages (@#$), and the underscore character (_).

When the owner or name value is split over two or more lines, the rules for continuing File Manager batch parameters must be observed. See the File Manager (Base) User’s Guide for a full explanation. Briefly, each part of the value must be enclosed in apostrophes, and a comma must appear at the end of each line. The second and subsequent lines should include $FILEM followed by a space and the next part of the parameter value, enclosed in apostrophes. When the owner/name value contains special characters FM/DB2 automatically determines a unique two-character delimiter for the parameter value. The default is \$, another delimiter is used if \$ appears in the owner/name. The two-character delimiter appears at the start and finish of the owner/name value. When the owner/name value spans two or more lines, the two-character delimiter appears at the start and end of every line, and apostrophes are used at the start and end of each partial parameter value - to satisfy the continuation rules for File Manager batch parameters. When you code your own batch control statements and the owner/name values extend over 2 or more lines, you should specify each part of the name separately and not use OBJIN/OBJOUT. If the owner/name values contain special characters, you must specify a two- character delimiter at the start and end of the parameter value (or partial parameter value). The delimiter must not appear in the owner/name value, and the first character of the delimiter must be a special character except for apostrophe and quote characters. When 2 character delimiters are used for an owner/name value, specify the owner/name exactly as is, do not double apostrophes or quotation characters, or omit any spaces.

Specifying the template for a DB2 object

The File Manager template for a DB2 object must be an FM/DB2 template that was generated by FM/DB2. Unlike File Manager base, you cannot use a copybook to describe the structure of a DB2 object.

The JCL generated by FM/DB2 for the various batch functions uses the following batch keywords:

TMIN  For functions where there is an input DB2 object (export, print, copy, data create)

TMOU T  For functions where there is an output DB2 object (import, copy).
Specifying the template for a DB2 object

When it is necessary to specify the template data set name in the JCL statements, rather than via a File Manager batch keyword, you can use the alternative keywords TINPUT and TINMEM.

Specifying the template/copybook for a data set

To describe the structure of a data set you can use:
- A COBOL copybook
- A HLASM copybook
- A PL/I copybook
- A File Manager template

The JCL generated by FM/DB2 for the various batch functions always uses the TCIN or TCOUT batch keywords, as described below:

TCIN   For functions where there is an input data set (import)

TCOUT  For functions where there is an output data set (export)

When it is necessary to specify the template/copybook data set name in the JCL statements, rather than via a File Manager batch keyword, you can use the alternative keywords TINPUT and TINMEM, or TOUTPUT and TOUTMEM.
Part 3. Appendixes
Appendix. FM/DB2 messages

Messages

In this section, selected batch messages are listed in alphanumeric order. For each message, the information provided comprises:

- The message identifier.
- The text of the message.
- An explanation of the message.
- The required user response.

Messages issued by File Manager DB2 have a unique alphanumeric identifier with the format:

FMNAaannn

where:

- **a** is an alphabetic character.
- **annn** is a 3-digit number.

When the message is issued by a batch function, the identifier is printed in front of the text of the message.

When the message is issued by an online function, the identifier is not displayed with the message. However, you can obtain the identifier of a message by typing MSGID on the command line and pressing Enter. This command causes the message identifier of the last message issued to be displayed on your screen.

Many of the online messages are comprised of a short message and long message. The short message is displayed in the upper right corner of the panel. The long message is only displayed if you press the F1 key when the short message is displayed.

<table>
<thead>
<tr>
<th>FMNB130</th>
<th>Incompatible field (short). The LOCATE command with a character position can only be used on character fields with lengths &gt; 50 bytes. (long).</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Explanation:</strong></td>
<td>The user entered a LOCATE command with a field number and character position. The command is invalid because either the specified field is not a character field, or the length of the specified field is 50 bytes or less.</td>
</tr>
<tr>
<td><strong>User response:</strong></td>
<td>Issue the LOCATE command without the character position. When the display has scrolled to show the field of interest, use the scale line to locate the character position.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>FMNB131</th>
<th>Invalid char posn (short). The character position is invalid for the specified field. (long).</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Explanation:</strong></td>
<td>The user entered a LOCATE command with a field number and character position. The command is invalid because the specified character position is greater than the maximum number of characters in the field.</td>
</tr>
<tr>
<td><strong>User response:</strong></td>
<td>Reduce the value for the character position.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>FMNB132</th>
<th>Only in SNGL (short). The CHARPOS command is only available in SNGL display format. (long).</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Explanation:</strong></td>
<td>The user entered the CHARPOS editor primary command when the display mode was not SINGLE (SNGL). The CHARPOS command is only valid in SNGL display format. The command cannot be processed.</td>
</tr>
<tr>
<td><strong>User response:</strong></td>
<td>Change the editor display mode to SINGLE and re-issue the command.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>FMNB133</th>
<th>CHARPOS off (short). Character position information is off. (long).</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Explanation:</strong></td>
<td>The user entered the CHARPOS editor</td>
</tr>
<tr>
<td><strong>User response:</strong></td>
<td></td>
</tr>
</tbody>
</table>

© Copyright IBM Corp. 2001, 2013  819
primary command. The character position information
has been turned off.

**User response:** Informational message - none required.

**FMNBF134** CHARPOS on (short). Character
position information is on. (long).

**Explanation:** The user entered the CHARPOS editor
primary command. The character position information
has been turned on.

**User response:** Informational message - none required.

**FMNBF139** Command not supported (short). Insert,
repeat and copy commands are not
supported when the DB2 object contains
large binary object (BLOB) columns.

**Explanation:** The user entered either an insert, copy or
repeat prefix command in an FM/DB2 editor session of
an object that contains a binary large object column.
Insert, repeat and copy are not supported when the
object being processed contains a binary large object
column. The operation cannot be performed.

**User response:** None - informational message.

**FMNDA001** IBM File Manager for z/OS Version 13
Release 1

**Explanation:** Shows the version and release
information for File Manager.

**User response:** No action required. Informational
message.

**FMNDA002** IBM File Manager for z/OS DB2
Component

**Explanation:** Shows the full component name for File
Manager.

**User response:** No action required. Informational
message.

**FMNDA003** IBM File Manager/DB2

**Explanation:** Shows the abbreviated component name
for File Manager.

**User response:** No action required. Informational
message.

**FMNDA004** DB2 Component

**Explanation:** Shows the short component name for
File Manager.

**User response:** No action required. Informational
message.

**FMNDA009** IBM File Manager for z/OS Version 13
Release 1 Licensed Materials Property of
IBM
5655-Q12
Copyright IBM Corporation 2013
All Rights Reserved.
Trademark of International Business
Machines

**Explanation:** Shows the product name and copyright
information for File Manager, DB2 component.

**User response:** No action required. Informational
message.

**FMNDA011** Panel panel name error. (short). PQUERY
error processing ISPF panel panel name.

**Explanation:** An ISPF error was encountered issuing a
PQUERY service for panel name panel name.

**User response:** This is a File Manager internal error.
Contact IBM service for assistance.

**FMNDA012** Installation error. DB2 catalog table DB2
catalog table name is not available. Some
FM/DB2 functions will be inoperative.
Contact your Systems Programmer.

**Explanation:** File Manager DB2 attempted to access
the DB2 catalog table DB2 catalog table name. The access
failed because DB2 catalog table name does not exist.

**User response:** File Manager DB2 component requires
access to various DB2 catalog tables for correct and
optimal functionality. This message indicates an
installation related problem. It is possible to define
views of the DB2 catalog tables with a user-specified
owner (example SYSIBM), in place of the actual
owner, (SYSIBM). One cause of this error is the use of a
user-specified owner, but the views of the DB2 catalog
tables with the user-specified owner do not exist (have
not been created). Check the CATOWNER parameter,
FMN2POPI macro in the FMN2POPT. Check that views
of the DB2 catalog tables, (for example
SYSIBM.SYSTABLES) exist. See "Preparing to
customize FM/DB2" in the File Manager Customization
Guide, for more information.

**FMNDA013** Installation error. SELECT access to DB2
catalog table DB2 catalog table name is
required, but is not available. Some
FM/DB2 functions will be inoperative.
Contact your Systems Programmer.

**Explanation:** File Manager DB2 attempted to access
the DB2 catalog table DB2 catalog table name. The access
failed. Access to DB2 catalog table name is essential to
complete the current FM/DB2 function.

**User response:** File Manager DB2 component requires
SELECT access to various DB2 catalog tables for correct
and optimal functionality. This message indicates an installation problem; specifically that the DB2 authid used by the FM/DB2 user does not have SELECT access to the DB2 catalog table. See "Preparing to customize FM/DB2" in the File Manager Customization Guide, and to point 8 in 'Grant access to the DB2 catalogs', for more information. SELECT access to DB2 catalog table name needs to be GRANTed to the FM/DB2 user who encountered the error.

FMNDA014 Installation error. Access to column name in DB2 catalog table DB2 catalog table name is required, but is not available. Some FM/DB2 functions will be inoperative. Contact your Systems Programmer.

Explanation: File Manager DB2 attempted to access column name of DB2 catalog table DB2 catalog table name. The access failed. SELECT access to column name of DB2 catalog table name is essential to complete the current FM/DB2 function.

User response: File Manager DB2 component requires access to various DB2 catalog tables for correct and optimal functionality. This message indicates an installation related problem, specifically that the DB2 authid used by the FM/DB2 user does not have SELECT access on column name in DB2 catalog. See "Preparing to customize FM/DB2" in the File Manager Customization Guide, and to point 8 in 'Grant access to the DB2 catalogs - required'. SELECT access to column name of DB2 catalog table name needs to be GRANTed to the FM/DB2 user who encountered the error.

FMNDA020 REQUIRED, BROWSE AUDITED

Explanation: Auditing is required, and auditing records are written to SMF. Auditing options are specified in either the FMN2POPT module, or externally, using a security product such as RACF. See File Manager Customization Guide for more information. This message is displayed on the "DB2 connection Information" pop-up panel.

User response: No action required.

FMNDA021 REQUIRED, BROWSE NOT AUDITED

Explanation: Auditing is required, and auditing records are not subject to audit. Auditing options are specified in either the FMN2POPT module, or externally, using a security product such as RACF. See File Manager Customization Guide for more information. This message is displayed on the "DB2 connection Information" pop-up panel.

User response: No action required.

FMNDA022 SAF-RULE CONTROLLED

Explanation: Auditing within FM/DB2 is controlled using SAF rules, defined using a security product such as RACF. Auditing options are specified in either the FMN2POPT module, or externally, using a security product such as RACF. See File Manager Customization Guide for more information. This message is displayed on the "DB2 connection Information" pop-up panel.

User response: No action required.

FMNDA023 SAF-RULE CONTROLLED, AUDITING NOT ALLOWED

Explanation: Auditing within FM/DB2 is controlled using SAF rules, defined using a security product such as RACF. These rules specifically prevent the writing of audit records by FM/DB2. Auditing options are specified in either the FMN2POPT module or externally, using a security product such as RACF. See File Manager Customization Guide for more information. This message is displayed on the "DB2 connection Information" pop-up panel.

User response: No action required.
FMNDA027 DEMAND, BROWSE AUDITED

Explanation: Demand auditing is in effect, browse functions are subject to audit. Demand auditing means that an audit report is automatically printed at the end of the current editor session. Auditing options are specified in either the FMN2POPT module or externally, using a security product such as RACF. See [File Manager Customization Guide](#) for more information. This message is displayed on the “DB2 connection Information” pop-up panel.

User response: No action required.

FMNDA028 DEMAND, BROWSE NOT AUDITED

Explanation: Demand auditing is in effect, browse functions are not subject to audit. Demand auditing means that an audit report is automatically printed at the end of the current editor session. Auditing options are specified in either the FMN2POPT module, or externally using a security product such as RACF. See [File Manager Customization Guide](#) for more information. This message is displayed on the “DB2 connection Information” pop-up panel.

User response: No action required.

FMNDA030 OPTIONAL, OFF, BROWSE AUDITED

Explanation: Auditing is optional and initially off, browse functions are subject to audit. Auditing options are specified in either the FMN2POPT module, or externally using a security product such as RACF. See [File Manager Customization Guide](#) for more information. This message is displayed on the “DB2 connection Information” pop-up panel.

User response: No action required.

FMNDA031 OPTIONAL, OFF, BROWSE NOT AUDITED

Explanation: Auditing is optional and initially off, browse functions are not subject to audit. This message is displayed on the “DB2 connection Information” pop-up panel.

User response: No action required.

FMNDA032 OPTIONAL, PROFILE, BROWSE AUDITED

Explanation: Auditing is optional and the current value is set based on the user ISPF profile. Browse functions are subject to audit. Auditing options are specified in either the FMN2POPT module, or externally using a security product such as RACF. See [File Manager Customization Guide](#) for more information. This message is displayed on the “DB2 connection Information” pop-up panel.

User response: No action required.

FMNDA033 OPTIONAL, PROFILE, BROWSE NOT AUDITED

Explanation: Auditing is optional and the current value is set based on the user ISPF profile. Browse functions are not subject to audit. Auditing options are specified in either the FMN2POPT module, or externally using a security product such as RACF. See [File Manager Customization Guide](#) for more information. This message is displayed on the “DB2 connection Information” pop-up panel.

User response: No action required.

FMNDA034 OPTIONAL, SMF, OFF, BROWSE AUDITED

Explanation: Auditing is optional and audit records are written to SMF. The audit option is initially off, and browse functions are subject to audit. Auditing options are specified in either the FMN2POPT module, or externally using a security product such as RACF. See [File Manager Customization Guide](#) for more information. This message is displayed on the “DB2 connection Information” pop-up panel.

User response: No action required.

FMNDA035 OPTIONAL, SMF, OFF, BROWSE NOT AUDITED

Explanation: Auditing is optional and audit records are written to SMF. The audit option is initially off, browse functions are not subject to audit. Auditing options are specified in either the FMN2POPT module, or externally using a security product such as RACF. See [File Manager Customization Guide](#) for more information. This message is displayed on the “DB2 connection Information” pop-up panel.

User response: No action required.

FMNDA036 OPTIONAL, SMF, PROFILE, BROWSE AUDITED

Explanation: Auditing is optional and audit records are written to SMF. The current value for the audit option is set based on the user ISPF profile, browse functions are subject to audit. Auditing options are specified in either the FMN2POPT module, or externally using a security product such as RACF. See [File Manager Customization Guide](#) for more information. This message is displayed on the “DB2 connection Information” pop-up panel.

User response: No action required.

FMNDA037 OPTIONAL, SMF, PROFILE, BROWSE NOT AUDITED

Explanation: Auditing is optional and audit records are written to SMF. The current value for the audit option is set based on the user ISPF profile, browse
functions are not subject to audit. Auditing options are specified in either the FMN2POPT module, or externally using a security product such as RACF. See File Manager Customization Guide for more information. This message is displayed on the “DB2 connection Information” pop-up panel.

User response: No action required.

FMNDA038 DEMAND, SMF, BROWSE AUDITED

Explanation: Demand auditing is in effect, audit records are written to SMF, browse functions are subject to audit. Demand auditing means that an audit report is automatically printed at the end of the current editor session. Auditing options are specified in either the FMN2POPT module, or externally using a security product such as RACF. See File Manager Customization Guide for more information. This message is displayed on the “DB2 connection Information” pop-up panel.

User response: No action required.

FMNDA039 DEMAND, SMF, BROWSE NOT AUDITED

Explanation: Demand auditing is in effect, audit records are written to SMF, browse functions are not subject to audit. Demand auditing means that an audit report is automatically printed at the end of the current editor session. Auditing options are specified in either the FMN2POPT module, or externally using a security product such as RACF. See File Manager Customization Guide for more information. This message is displayed on the “DB2 connection Information” pop-up panel.

User response: No action required.

FMNDA040 Recovered from DB2 abend (short).

FM/DB2 has recovered from an abend in DB2. The FM/DB2 function that was executing when the DB2 abend occurred has been terminated. (long)

Explanation: A DB2 abend occurred during processing for the current function. The File Manager subtask responsible for connection to DB2 has been terminated, and a new subtask established. Any uncommitted changes made by the FM/DB2 function prior to the abend will have been backed out.

User response: Document information about the function being used when the abend occurs. If the problem occurs again notify your System Programmer. If there is no DB2-related problem that might explain the abend, contact IBM support.

FMNDA100 Alloc. error -unit (short). The attempt to allocate a temporary data set for the SQL statement failed. UNIT unit is invalid. Check the temporary data set allocation option values. (long).

Explanation: File Manager attempted to allocate a temporary data set, in which to store SQL or some other information. The allocation attempt failed, the most likely explanation is that the UNIT parameter specified on the “Set Temporary Data Set Allocation Options” panel is invalid. This system option panel is accessed using 0.0.8 from the FM/DB2 main menu panel.

User response: Check the value specified for the UNIT parameter on the “Set Temporary Data Set Allocation Options” panel. Remove any value and retry the operation. If this resolves the problem, contact your system’s programmer for advice on valid/invalid UNIT values.

FMNDA997 **** Limit of Rows Retrieved ****

Explanation: This message appears at the bottom of an FM/DB2 editor session, when a row count value (example 100) was specified on the editor function entry panel, and FM/DB2 read that many rows into the editor session without reaching the end of the DB2 object.

User response: No action required. Informational message.

FMNDA998 Internal Logic Error (module name) - message

Explanation: This is an internal error message, indicating a problem with the processing of the DB2 attachment.

User response: Contact IBM support.

FMNDB000 JESCT not available (short). JESCT control block currently not available for DB2 subsystem selection. (long).

Explanation: This is an internal error. FM/DB2 attempted to access the JESCT however the pointer was zero.

User response: Contact IBM support.

FMNDB001 No DB2 subsystems (short). No DB2 Subsystems have been defined on this system. File Manager/DB2 can not run on this system. (long).

Explanation: FM/DB2 could not find any DB2 systems defined on the current z/OS system.

User response: If there are DB2 systems defined on the current z/OS system, contact IBM support.
FMNDB002  SSID DB2 SSID not matched (short). No DB2 Subsystems matching DB2 SSID are defined on this system. (long).

Explanation:  This message may be issued during DB2 subsystem selection when a name with a wildcard (example DSN*) is entered in the DB2 SSID field on the FM/DB2 main menu. No matching DB2 systems were found.

User response:  No action required. Informational message. Clear the DB2 SSID entirely to show all DB2 systems.

FMNDB003  DB2 SSID not defined to z/OS (short). DB2 Subsystem/Group DB2 SSID is not defined to the local z/OS system. (long).

Explanation:  DB2 system DB2 SSID is not defined on the local z/OS system. This message may be issued in response to an attempt to change the currently connected DB2 system using a command like SSID XXXX.

User response:  Check that DB2 SSID is correctly specified. Correct the name and try again.

FMNDB004  SSID DB2 SSID not active (short). DB2 subsystem/group DB2 SSID is not currently active. (long).

Explanation:  The DB2 system or group DB2 SSID is defined on the local z/OS system but is inactive (stopped).

User response:  FM/DB2 cannot connect to inactive DB2 systems. Determine the reason why the DB2 system is inactive. When the DB2 system is started, attempt the connection again.

FMNDB005  Already connected (short). FM/DB2 is currently connected to DB2 system DB2 SSID. (long).

Explanation:  FM/DB2 is already connected to DB2 system DB2 SSID. This message may be issued in response to an attempt to change the currently connected DB2 system using a command like SSID XXXX.

User response:  None if DB2 SSID was correctly specified. Otherwise, check the DB2 SSID, make appropriate corrections, and try again.

FMNDB006  Cannot connect to DB2 DB2 SSID, RRSAF function RC &RC. Reason reason code.

Explanation:  A call to the DB2 RRSAF attachment failed. DB2 SSID is the currently connected DB2 system; function is the failing RRSAF function and reason is the failing reason code.

User response:  This error indicates an installation error. Confirm that the RRSAF attachment is required. You will typically find that the CAF attachment meets all your requirements. The DB2 attachment used by FM/DB2 is specified in the FMN2POPT module. See File Manager Customization Guide for a detailed explanation. Ensure that the appropriate DB2 interface module (DSNRLI) is accessible to FM/DB2. If these actions fail to resolve the problem, contact IBM support.

FMNDB007  A DB2 Subsystem has not been specified.

Explanation:  A DB2 system has not been specified. FM/DB2 must connect to a DB2 system before it can perform useful work.

User response:  Use the "DB2 SSID" field on the FM/DB2 main menu to specify the name of a DB2 system.

FMNDB008  Subsystem DB2 SSID does not identify a uniquely active DB2 subsystem

Explanation:  A DB2 SSID value was specified, but it does not uniquely identify a DB2 system. An example is DSN*.

User response:  Specify an unambiguous value.

FMNDB009  DB2 subsystem DB2 SSID is not currently active on this system.

Explanation:  The specified DB2 SSID value DB2 SSID is inactive.

User response:  Take action to start the DB2 system. When the DB2 system is active, retry the operation.

FMNDB010  Row Row value 1 of Row value 2

Explanation:  This message is displayed on various selection lists, including the DB2 subsystem selection panel. Row value 1 is the first row shown on the display and there are a total of Row value 2 rows available for display.

User response:  No action required. Informational message.

FMNDB011  Invalid selection (short). Enter "S" or "/" to select the required DB2 subsystem. (long).

Explanation:  A character other than "S" or "/" was entered against an entry on the DB2 subsystem selection panel. The only valid characters for selecting an entry are "S" or "/".

User response:  Use either 'S' or "/" to select the required entry.
FMNDB012 SSID  DB2 SSID not active (short).
Selected DB2 subsystem  DB2 SSID  is not currently active on this system.
(long).
Explanation: An attempt was made to select DB2 system  DB2 SSID  from the DB2 subsystem selection panel. This DB2 system is not currently active.
User response: Determine why the DB2 system is inactive and take action to start it. When the DB2 system is active, retry the operation.

FMNDB013 Selection terminated (short) DB2 subsystem selection was terminated with the EXIT PF key command. File Manager DB2 is not currently connected to any DB2 subsystem. (long)
Explanation: The user pressed the EXIT PF key on the DB2 subsystem selection menu without making any selection. The FM/DB2 main menu is re-displayed. FM/DB2 is not connected to any DB2 system.
User response: No action required. Informational message. It is necessary to connect to a DB2 system prior to performing any FM/DB2 functions.

FMNDB014 Display refreshed (short). List of DB2 Subsystems has been refreshed using current information. (long)
Explanation: The user issued the REFRESH primary command on the DB2 subsystem selection list panel. The list of DB2 systems displayed is refreshed.
User response: No action required. Informational message.

FMNDB015 SSID  DB2 SSID  displayed (short). DB2 Subsystems matching  DB2 SSID  are displayed. (long)
Explanation: The user issued a SHOW DB2 SSID command on the DB2 subsystem selection list panel. The list of DB2 systems that match the SSID specification are displayed. The SSID value specified may include wild cards, example DSN*.
User response: No action required. Informational message.

FMNDB016 Command not active (short). The command to change the DB2 subsystem is not available from this panel. (long).
Explanation: A command to change the currently connected DB2 SSID was issued on a panel where the command is not available.
User response: The currently connected DB2 SSID can only be changed when no FM/DB2 function is active.

FMNDB017 SSID name required (short). Enter the DB2 subsystem name that FM/DB2 should connect to. (long)
Explanation: A command to change the currently connected DB2 system was issued without an operand.
User response: The command to change the currently connected DB2 system requires an operand, the DB2 SSID of the DB2 system to which connection is desired. Reissue the command with the required DB2 system SSID as the operand.

FMNDB018 Invalid remote server (short). Remote server  DB2 server name  is not a DB2 UDB for z/OS system. Connection to remote servers that are not DB2 UDB for z/OS servers is not supported. (long)
Explanation: A DB2 server name was entered in the location field on a FM/DB2 function entry panel. While the DB2 server name is defined as a remote DB2 server at the local DB2 system, this server is not a DB2 UDB for z/OS system. FM/DB2 only supports remote connections to DB2 UDB for z/OS servers.
User response: Connection to this remote DB2 server is not supported. Specify another name.

FMNDB019 Version  DB2 version  invalid (short). The remote DB2 version/release is  DB2 version. Connection is supported for DB2 Version 9 and later systems only. (long)
Explanation: A DB2 server name was entered in the location field on a FM/DB2 function entry panel. The version of the remote DB2 server is  DB2 version. Connection to this version of DB2 is not supported.
User response: Connection to this remote DB2 server is not supported. Specify another name.

FMNDB020 Operating System is unsupported. Can not run File Manager/DB2.
Explanation: This is an internal error. During MVS token initialization an unsupported version of z/OS was detected. It is not possible to run FM/DB2 against this version of z/OS, or its predecessors.
User response: Ensure that the host operating system is at a level specified in the Program Directory. If you believe that the host operating system is at a supported level, contact IBM support.
**FMNDB021** Error in function Callable Service Routine, Return code return code.

*Explanation:* This is an internal error. A call to function failed with return code return code.

*User response:* Contact IBM support.

**FMNDB022** DB2 Subsystem not set (short). Error invoking File Manager/DB2. DB2 Subsystem has not been set. (long).

*Explanation:* An attempt was made to run FM/DB2, without correctly specifying a DB2 SSID name, or without first connecting to a DB2 system.

*User response:* If the message occurs in batch, check that the DB2 SSID value is correctly specified in the JCL. If this fails to resolve the problem, start FM/DB2 online and regenerate the batch JCL. If the batch job still fails contact IBM support. If the message occurs online, return to the FM/DB2 main menu and enter the name of a valid DB2 SSID in the DB2 SSID field, press Enter to connect.

**FMNDB023** Cannot connect to DB2 DB2 SSID, CAF function RC return code, Reason reason code.

*Explanation:* FM/DB2 attempted to call the DB2 CAF interface with function function. The call failed with return code return code and reason code reason code.

*User response:* This message usually indicates an installation or set up problem. The most likely causes are:

1. Not binding the FM/DB2 plan (FMN2PLAN) by default.
2. Specifying an incorrect DB2 plan name in the FMN2POPT module.

See File Manager Customization Guide for more information.

**FMNDB024** Unable to load DB2 Interface module module name. Abend Code abend code-reason code. Check STEPLIB/LINKLIST for DB2 load libraries.

*Explanation:* FM/DB2 attempted to load the DB2 interface module module name. The load failed with abend code abend code and reason reason code.

*User response:* This is probably a customization or setup problem. If the Abend is 806 (module not found):

1. Check that the DB2 load library has been correctly specified in the FMN2SSDM macro for the DB2 system in question.
2. If the FMN2POPT does not include the DB2 load library names for the DB2 system, check any applicable STEPLIB/JOB LIB and LINKLIST for the DB2 load library. The DB2 load library must reside in a library that is accessible using the standard search order for the LOAD macro.

For other abends refer to point (1) as a starting point. If this fails to resolve the problem, contact IBM support.

**FMNDB025** Not currently connected to a DB2 subsystem.

*Explanation:* FM/DB2 is not currently connected to any DB2 subsystem.

*User response:* No action required. Informational message. It will be necessary to connect to a DB2 system prior to performing any FM/DB2 functions.

**FMNDB026** Subsystem: DB2 SSID, DB2 release: DB2 VRM indicator.

*Explanation:* FM/DB2 is connected to DB2 system DB2 SSID. The version of this DB2 system is DB2 VRM indicator.

*User response:* No action required. Informational message. issued in an FM/DB2 batch job.

**FMNDB027** FM/DB2 plans: Plan name 1, Plan name 2

*Explanation:* The FM/DB2 plan names are Plan name 1 and Plan name 2.

*User response:* No action required. Informational message. issued in an FM/DB2 batch job.

**FMNDB028** Description: user specified description.

*Explanation:* This message user specified description appears in the “About DB2” popup panel. The user specified description is the value specified in the FMN2SSDM macro, DESC parameter for the appropriate DB2 system.

*User response:* No action required. Informational message.

**FMNDB029** DB2 SSID executing DB2 level DB2 level 1. Minimum supported level is DB2 level 2.

*Explanation:* This message is issued when an attempt is made to connect to a DB2 system at an unsupported version/maintenance level. The level of DB2 system DB2 SSID is DB2 level 1. The minimum supported level is DB2 level 2.

*User response:* Connection to the DB2 system is not supported. See File Manager Customization Guide and File Manager User’s Guide and Reference for DB2 Data for DB2 data for information on supported DB2 versions, for the version of FM/DB2 in use.
FMNDB030  DB2I unavailable (short). Unable to start DB2I. Check STEPLIB/LINKLIST for DB2 libraries. (long).

Explanation: A ISPF PQUERY service for panel DSNEPRI was issued, the request failed.

User response: This error indicates an installation or setup problem. Ensure that the DB2 libraries are available to the FM/DB2 user. The recommended way to do this is to code the DB2 libraries in the FMN2SSDM macro entry for each DB2 system that will be accessed by FM/DB2. See File Manager Customization Guide for more information.

FMNDB031  SQL ID not authorized (short). You are not authorized to use the SQL user ID DB2 authid on DB2 Subsystem DB2 SSID. (long).

Explanation: An attempt was made to change the current SQLID on DB2 system DB2 SSID to DB2 authid. The attempt failed. The usual reason for this is that the FM/DB2 user is not authorized to use DB2 authid.

User response: Consult with either the DB2 or Security Administrator as to the reason why access to DB2 authid is not available.

FMNDB032  You are not authorized to use the SQL user ID DB2 authid.

Explanation: An attempt was made to change the current SQLID to DB2 authid. The attempt failed. The usual reason for this is that the FM/DB2 user is not authorized to use DB2 authid.

User response: Consult with either the DB2 or Security Administrator as to the reason why access to DB2 authid is not available.

FMNDB033  Auditing: Audit status:

Explanation: This message is used to display the audit status for the currently connected DB2 system, and appears on the "About DB2" pop-up panel.

User response: No action required. Informational message.

FMNDB034  No DB2 subsystem with name DB2 SSID. CAF function RC return code, Reason reason code.

Explanation: A call to the CAF interface, function function for DB2 system DB2 SSID failed with RC return code and reason reason code. The reason is that the nominated DB2 system does not exist.

User response: Check the DB2 system name and try again.

FMNDB035  DB2 group/subsystem DB2 SSID is not active, CAF function RC return code, Reason reason code.

Explanation: A call to the CAF interface, function function for DB2 system DB2 SSID failed with RC return code and reason reason code. The reason is that the nominated DB2 system is not active.

User response: Start the DB2 system. Retry the operation.

FMNDB036  Not authorized to connect to DB2 SSID, CAF function function RC return code, Reason reason code.

Explanation: A call to the CAF interface, function function for DB2 system DB2 SSID failed with RC return code and reason reason code. The likely reason is that the FM/DB2 user is not authorized to connect to the nominated DB2 system.

User response: Consult with either the DB2 or Security Administrator as to the reason why the FM/DB2 user is not authorized to connect to the DB2 system DB2 SSID.

FMNDB037  FM/DB2 not installed on DB2 SSID, CAF function function RC return code, Reason reason code.

Explanation: A call to the CAF interface, function function for DB2 system DB2 SSID failed with RC return code and reason reason code. The reason is that FM/DB2 has not been installed on DB2 system DB2 SSID.

User response: Verify that FM/DB2 is installed on DB2 system DB2 SSID. Retry the operation.

FMNDB038  Unable to load options module module name. Probable installation error.

Explanation: FM/DB2 attempted to load module module name. The attempt failed. The cause is an installation error; the File Manager load library is not available to the FM/DB2 user.

User response: Check the allocation of the FM/DB2 load library using, for example, 'TSO ISRDDN&' (under ISPF). Ensure that module name is located in a load library accessible to the FM/DB2 user.

FMNDB039  FMN2POPI module invalid (short). The internal structure of FMN2POPT is invalid. Probable installation error. (long).

Explanation: FM/DB2 loaded the FMN2POPT module, the contents are corrupt and unusable.

User response: The FMN2POPT module was assembled using an older version of FM/DB2. Reassemble the FMN2POPT module, paying attention to the documentation provided with the module.
FMNDB040 • FMNDB047

to the SYSLIB statements in the JCL, to ensure the
correct versions of the FM/DB2 macros are used. Retry
the operation.

**FMNDB040** SSID DB2 SSID not available (short). Installation settings prevent the selection of DB2 subsystem/group DB2 SSID. (long).

**Explanation:** An attempt was made to connect to DB2 system DB2 SSID. This DB2 system is marked “Unavailable” in the FMN2POPT installation module. The DB2 system cannot be used.

**User response:** Select another DB2 system.

**FMNDB041** Incompatible connection already active, CAF function, RC return code, Reason reason code.

**Explanation:** FM/DB2 attempted to call the CAF interface, function function. An error occurred, RC return code and Reason reason code. The cause is an incompatibility (release level mismatch) between the DB2 and call attachment code.

**User response:** This is an installation error. Check the specification of the DB2 libraries in the FMN2POPT, FMN2SSDM macro for the appropriate DB2 system, with attention to the Version/maintenance level of these libraries compared with the Version/maintenance level of the DB2 system.

**FMNDB042** DB2 SSID Release Minimum Release (short). DB2 SSID executing DB2 level Release. The minimum supported level of DB2 for this release of File Manager is Minimum Release. FM/DB2 may operate successfully with DB2 Version 8, however no support will be provided for any Version 8 specific problems that may be encountered.

**Explanation:** FM/DB2 connected to DB2 SSID which is at version and release Release. The minimum supported DB2 version/release for this version of FM/DB2 is Minimum Release, which is greater than Release.

**User response:** No action required. Informational message. FM/DB2 should operate without issues. However should a problem be encountered that is specific to DB2 version 8, no support will be available.

**FMNDB043** DB2 SSID Release Maximum Release (short). DB2 SSID executing DB2 level Release. This version and release of File Manager does not currently support connection to DB2 Maximum Release (or later) subsystems.

**Explanation:** An attempt was made to connect to DB2 SSID which is at version and release Release. The maximum supported DB2 version/release for this version of FM/DB2 is Maximum Release, which is less than Release. The connection is not established and the DB2 system cannot be accessed. All versions of software are designed to work with a limited set (of version and release levels) of other software.

**User response:** Contact your IBM representative for information about more recent versions of FM/DB2 which may support the DB2 version/release of DB2 SSID.

**FMNDB044** Communications error (short). FM/DB2 encountered an SQL error SQL error code attempting to access the DB2 system at DB2 location name. This indicates that the remote DB2 system is not available, or that there was a security related problem. (long).

**Explanation:** FM/DB2 attempted to connect to remote DB2 server DB2 location name. The connection failed with SQLCODE SQL error code.

**User response:** Look up SQL error code in the DB2 messages and codes manual and take the action indicated. This error indicates a set up problem. One possibility is incorrect DRDA® definitions for DB2 location name at the local DB2 server.

**FMNDB045** Not defined.

**Explanation:** A location nickname was not defined in the FMN2POPT module.

**User response:** Check the FMN2SSDM definitions in the FMN2POPT module and add the location nickname to the appropriate entry.

**FMNDB046** Defined but no location specified.

**Explanation:** A location nickname was specified, but no location, to which the nickname refers, was specified.

**User response:** Check the FMN2SSDM macro for the currently connected DB2 system. Ensure that both a location and location nickname are specified.

**FMNDB047** Authorization error (short). FM/DB2 encountered an SQL error (-725) attempting to access the DB2 system at Remote location name. This can occur when the user has changed the SSQLID at the local DB2 system, but the user does not have authority to use this SSQLID at the remote DB2 system. (long).

**Explanation:** Refer to the long message.

**User response:** Ensure that the user has access to the
remote DB2 system Remote location name using the currently set DB2 SQLID.

**FMNDB048** FMN2DATA load failed (short). Unable to load FMN2DATA module. Probable installation error. (long).

*Explanation:* FM/DB2 attempted to load data module FMN2DATA. The attempt failed. FM/DB2 cannot operate without this module.

*User response:* This is a customization or setup problem. Check that FMN2DATA is present in the load libraries accessible to the FM/DB2 user. This module is normally located in the SFMNMOD1 library.

**FMNDB049** FMN2DENU load failed (short). Unable to load FMN2DENU module. Probable installation error. (long).

*Explanation:* FM/DB2 attempted to load data module FMN2DENU. The attempt failed. FM/DB2 cannot operate without this module.

*User response:* This is a customization or setup problem. Check that FMN2DENU is present in the load libraries accessible to the FM/DB2 user. This module is normally located in the SFMNMOD1 library.

**FMNDB050** ATTACH of FM/DB2 subtask program failed with return code return code.

*Explanation:* FM/DB2 attempted to attach the subtask responsible for DB2 access. The attempt failed with RC return code. FM/DB2 cannot operate without the subtask for consistency.

*User response:* This is an internal error. Retry the operation. If the problem persists check for installation related problems and contact IBM support.

**FMNDB051** Communications error (SQLCODE=SQL error code) accessing DB2 SSID.

*Explanation:* FM/DB2 attempted to access DB2 SSID and received a communications error SQL error code.

*User response:* Look up SQL error code in the DB2 messages and codes manual and take the action indicated.

**FMNDB052** DB2 server server name is not a DB2 UDB for z/OS server, code=return code.

*Explanation:* FM/DB2 (in batch) connected to DB2 server server name. The connect information indicates that the DB2 server is not a DB2 UDB for z/OS server. Connection is only supported to DB2 UDB for z/OS servers.

*User response:* Specify the name of a DB2 UDB for z/OS server only.

**FMNDB053** SET DECFLOAT ROUNDING MODE failed with return code return code.

*Explanation:* FM/DB2 attempted to set the DECFLOAT ROUNDING MODE, the attempt failed with return code return code.

*User response:* Check that the DB2 system is operational. If the problem persists contact IBM support.

**FMNDB054** SSID DB2 SSID not defined (short). There is no definition (FMN2SSDM macro entry) for DB2 subsystem or group DB2 SSID in the FM/DB2 installation module (FMN2POPT). Contact the person who installed FM/DB2. (long).

*Explanation:* The FMN2POPT installation module is used to specify the DB2 systems that FM/DB2 can connect to. The recommends defining an FMN2SSDM macro entry for every DB2 system that FM will connect to. The FMN2POPI macro includes a CONNECT option, the default setting is DEFINED. When CONNECT=DEFINED is in effect, FM/DB2 will only attempt connection to DB2 systems that have an FMN2SSDM entry in the FMN2POPT. An attempt to connect to a DB2 system with no FMN2SSDM entry results in this error.

*User response:* There are two usual approaches for resolving this error:

1. Add an FMN2SSDM macro entry for every DB2 system that FM/DB2 will access to the FMN2POPT, as recommended in the...

2. Specify CONNECT=ANY in the FMN2POPI macro in the FMN2POPT. While this will allow FM/DB2 to attempt connection to ANY DB2 system, it does not necessarily remove the need to define an FMN2SSDM macro entry for every DB2 system that FM/DB2 will connect to. See the...

**FMNDB055** Selection canceled (short). DB2 subsystem selection was canceled. Connection to the previously connected DB2 system has been re-established.

*Explanation:* The user exited the DB2 subsystem selection panel without selecting a DB2 subsystem or group.

*User response:* No action required. This is an informational message.
FMNDB056  Selection canceled (short). DB2 subsystem selection was canceled. File Manager DB2 is not currently connected to any DB2 subsystem.

Explanation: The user exited the DB2 subsystem selection panel without selecting a DB2 subsystem or group. FM/DB2 is not currently connected to any DB2 subsystem or group.

User response: No action required. This is an informational message. It will be necessary to connect to a DB2 system prior to performing any FM/DB2 functions.

FMNDB057  DB2 limits the output returned from a DB2 command. This limit has been reached and the output has been truncated.

Explanation: A DB2 command that produced large amounts of output was entered. DB2 has limited the amount of output produced, resulting in truncation.

User response: No action required. This is an informational message. Refine the DB2 command to be more specific.

FMNDB058  You cannot change the SQLID until FM/DB2 is connected to a DB2 subsystem or group.

Explanation: FM/DB2 was not connected to any DB2 system, and the user typed a new value in the SQLID field on the FM/DB2 main menu. It is not possible to change the current SQLID without first being connected to some DB2 system.

User response: Change the DB2 SQLID back to the default value, typically the user TSO logon ID. Now enter a valid DB2 SSID in the DB2 SSID field and press Enter to establish connection to the DB2 system. When the connection is successful, attempt to change the DB2 SQLID.

FMNDB059  ** not defined in installation module.

Explanation: This message appears in the DB2 subsystem selection display against any DB2 system that has no FMN2SSDM macro entry in the FMN2POPT.

User response: No action required. This is an informational message. Refer to File Manager Customization Guide for detailed information about how to code entries in the FMN2POPT module.

FMNDB060  Unable to load DB2 Interface module name. Abend Code abend code-reason code. Check STEPLIB/LINKLIST for DB2 load libraries.

Explanation: An FM/DB2 batch job attempted to load the DB2 interface module name. The LOAD abended with Abend code abend code, reason reason code.

User response: Check the STEPLIB or JOBLIB statement in the batch job. Ensure that the appropriate DB2 load library is included. If the batch job was generated using FM/DB2 online, the DB2 libraries are automatically included, as specified in the FMN2SSDM macro entry in the FMN2POPT for the appropriate DB2 system. This will not occur if there is no FMN2SSDM macro entry for the DB2 system, or if the DB2 libraries are not included in the FMN2SSDM macro.

FMNDB061  Catalog table table name is missing.

Explanation: An FM/DB2 batch job attempted to access DB2 catalog table table name. The access failed with SQLCODE-204 (object not found).

User response: If views of the DB2 catalog are in use (CATOWNER=xxxx in the FMN2POPI macro, FMN2POPT module, xxxx is not SYSIBM) then ensure that the appropriate view xxxx.table name has been created.

FMNDB062  No select access to catalog table table name.

Explanation: An FM/DB2 batch job attempted to access DB2 catalog table table name. The access failed with SQLCODE-551 (insufficient authority).

User response: Ensure that the DB2 authid associated with the batch job has SELECT access to table table name.

FMNDB063  Cannot access column in catalog table table name.

Explanation: An FM/DB2 batch job attempted to access column column in DB2 catalog table table name. The access failed with SQLCODE-206 error. The column column is not defined in table table name.

User response: If views of the DB2 catalog are in use (CATOWNER=xxxx in the FMN2POPI macro, FMN2POPT module, xxxx is not SYSIBM) then ensure that the appropriate view xxxx.table name has been created. Further, ensure that column column is included in view xxxx.table name.

FMNDB064  DB2 Abend Code abend code-reason code accessing name. Notify your systems programmer.

Explanation: A DB2 Abend occurred while running an FM/DB2 batch job. The DB2 Abend code is abend code, the reason code is reason code and the DB2 module that was accessed is name. The batch job ends and any uncommitted operations are backed out.

User response: Look up the DB2 Abend code and reason code in the DB2 UDB for z/OS Messages and
In most cases this type of error indicates an error with the batch job (for example, the wrong DB2 libraries being specified). Check the specification of the DB2 libraries in the FMN2SSDM macro entry for the appropriate DB2 system (FMN2POPT module). See the File Manager Customization Guide for more information.

If the batch job was not generated using the appropriate FM/DB2 online function, use FM/DB2 to regenerate the batch JCL and retry the operation.

FMNDB065  GEN command failed (short). The FM/DB2 GEN command failed - RC=return code. Please see your Systems Support staff for assistance. (long).

Explanation: The SQL Reverse Engineering function (GEN command) failed with reason return code.

User response: Possible failures include space related abends (x37) for the data sets used by the GEN code. If this is the case there will be an associated abend issued. Respond to the Abend by deleting and recreating the data set with more space, and retry the operation.

FMNDB066  No SSID parameter was passed to FMN2INEX. The last connected DB2 subsystem, if any, is used. Check that the connected DB2 subsystem is, in fact, what was intended.

Explanation: The REXX EXEC FMN2INEX can be used to invoke FM/DB2 by an external call. The REXX EXEC requires various parameters, including the DB2 SSID of the DB2 system that FM/DB2 should connect to. This was not provided.

User response: Check the method used to invoke the FMN2INEX exec and ensure that a DB2 SSID is provided.

FMNDB067  Invalid option code (short). The option code passed to FMN2INEX is either invalid or not supported. The option code has been discarded. (long).

Explanation: The REXX exec FMN2INEX can be used to invoke FM/DB2 by an external call. The exec requires various parameters, including a function code that indicates which FM/DB2 function to run. The function code provided was invalid.

User response: The function codes that may be passed to FMN2INEX are documented in the File Manager Customization Guide. Ensure that a supported function code is specified.

FMNDB068  Command not active (short). The command to change the DB2 subsystem is not available when FM/DB2 is invoked remotely. (long).

Explanation: FM/DB2 was invoked from an external application. In this situation it is not possible to change the currently connected DB2 SSID.

User response: No action required. Informational message.

FMNDB069  Command not available (short). DB2 commands cannot be issued when FM/DB2 has been started remotely. (long).

Explanation: FM/DB2 was invoked from an external application. In this situation it is not possible to issue DB2 commands.

User response: No action required. Informational message.

FMNDB070  Possible causes - Missing/invalid FMN2POPT module, eg invalid plan name.

Explanation: An FM/DB2 batch job encountered an error attempting to use the CAF attachment.

User response: The usual causes of this error are:
1. FM/DB2 is not installed on the target DB2 system.
2. The FMN2POPT module in use specifies an invalid plan name, perhaps the specified plan does not exist, or the plan exists but is invalid.

FMNDB071  Current SQLID (short). The current SQLID is the value specified. (long).

Explanation: The SQLID command was issued to change the current SQLID. However the specified SQLID is already the current SQLID.

User response: No action required. Informational message.

FMNDB072  SQLID value required (short). The SQLID command requires a value. Enter the new SQLID after the SQLID keyword, eg "SQLID FRANK". (long).

Explanation: The SQLID command was issued without an operand.

User response: Specify, for example, the desired SQLID after the SQLID command:

SQLID FRED

FMNDB073  SQLID cmd not available (short). The SQLID command is not available from this panel. (long).

Explanation: The SQLID command was issued from a panel where the command is not supported.

User response: Return to a higher level panel, for example a function entry panel, or menu panel, and reissue the command.
FMNDB074  UNICODE conversion error codepage CCSID 1 to codepage CCSID 2.

Explanation: In an FM/DB2 editor session of a Unicode encoded object, the conversion between CCSID CCSID 1 and CCSID CCSID 2 failed.

User response: One reason for this error is that the z/OS Unicode translation service for the specified CCSID pair has not been set up. Contact your System Programmer. If this does not solve the problem, contact IBM support.

FMNDB075  SQLID = SQLID (short). The current SQLID has been changed to SQLID. (long).

Explanation: A command to change the current SQLID was issued. The command succeeded, the current SQLID is now SQLID.

User response: No action required. Informational message.

FMNDB076  *** Rollback occurred. Transactions since start of session or last commit have been backed out. ***

Explanation: A DB2 rollback was issued. This message appears in audit log reports.

User response: No action required. Informational message.

FMNDB077  A DB2 commit has been issued.

Explanation: A DB2 commit was issued. This message appears in audit log reports.

User response: No action required. Informational message.

FMNDB078  UNICODE conversion error. UNICODE conversion error from codepage CCSID 1 to codepage CCSID 2. Return code return code, reason code reason code from CUNLCNV routine.

Explanation: In an FM/DB2 editor session of a Unicode-encoded object, the conversion between CCSID CCSID 1 and CCSID CCSID 2 failed. The CUNLCNV system service, used to perform the conversion, issued return code return code and reason code reason code.

User response: One reason for this error is that the z/OS Unicode translation service for the specified CCSID pair has not been set up. Contact your System Programmer. If this does not resolve the problem, contact IBM support.

FMNDB079  Change SSID first (short). An attempt to navigate to an FM/DB2 function was made whilst simultaneously changing the DB2 SSID. This is not supported. First change the DB2 SSID, and then navigate to the required function. (long).

Explanation: Processing on the FM/DB2 main menu panel includes the capability to change the current SQLID by typing a new value in the SQLID field, followed by the user pressing enter to validate the change. The ISPF short cut prefix (‘=’) can be used to navigate to some other FM/DB2 function. However, the use of this short cut bypasses the normal panel processing, and is therefore not supported.

User response: The use of ‘=’ is not supported within FM/DB2 online. See the File Manager Customization Guide for DB2 data. In situations where a change in SQLID is required to execute some FM/DB2 function, first change the SQLID, then navigate to the required function.

FMNDB088  Return code return code, reason code reason code from CUNLCNV routine.

Explanation: A call to the z/OS translation service routine CUNLCNV failed with return code return code, reason code reason code.

User response: One reason for this error is that the z/OS Unicode translation service for the specified CCSID pair has not been set up. Contact your System Programmer. If this fails to resolve the problem contact IBM support.

FMNDB091  Invalid selection (short). Enter ‘S’ or ‘/’ to select the required object type. (long).

Explanation: An invalid character was entered against an entry in an object selection list. The only characters that may be used to select the entry are ‘S’ and ‘/’.

User response: Use one of the characters indicated to select the object.

FMNDB092  Invalid selection (short). Enter ‘S’ or ‘/’ to select the required line command. (long).

Explanation: An invalid character was entered against an entry in an object selection list. The only characters that may be used to select the entry are ‘S’ and ‘/’.

User response: Use one of the characters indicated to select the object.


Explanation: This command is issued by the Basic SQL Prototyping function. An invalid command was
entered in the 'S' (Selection) field. The only valid characters that may be used are as shown in the long message.

User response: Use one of the character or characters shown in the long message to select the column.

FMNDB094 OBJIN|OBJOUT DB2 object not found.

Explanation: This message is issued by FM/DB2 functions that are executed in batch, when a DB2 object name does not exist. OBJIN appears when an input DB2 object name does not exist, for example print, copy and export. OBJOUT appears when an output DB2 object name does not exist, for example copy, import and data create.

User response: Correct the DB2 object name, re-submit the batch job.

FMNDB096 OBJIN|OBJOUT DB2 location not found.

Explanation: This message is issued by FM/DB2 functions that are executed in batch, when a DB2 location, specified as part of a DB2 object name, does not exist. OBJIN appears when the location appears in an input DB2 object name, for example print, copy and export. OBJOUT appears when the location appears in an output DB2 object name, for example copy, import and data create.

User response: Determine the failing DB2 location value. Ensure that this DB2 location is accessible from the DB2 system specified in the PARM statement, re-submit the batch job.

FMNDB098 DB2 SSID; executing DB2 level release. Maximum supported level is Maximum release.

Explanation: FM/DB2 connected to DB2 SSID, which is at DB2 version or release release. The maximum supported DB2 version or release for the current version of FM/DB2 is Maximum release, which is less than release. The connection to this DB2 system is not supported.

User response: Contact your IBM representative for information about which versions of FM/DB2 support DB2 systems at version or release release.

FMNDB099 Invalid DB2 SSID (short). The DB2 SSID is too long. Specify a 1-4 character value. (long).

Explanation: A command to change the currently connected DB2 system was issued, however the value entered for the desired DB2 system is too long. Valid DB2 SSIDs are 1-4 characters in length.

User response: Check the DB2 SSID that was specified; make corrections. Retry the operation.

FMNDB100 SQLCODE SQLCODE. Statement type (short). SQLCODE SQLCODE was encountered while issuing a Statement type SQL statement.

Explanation: An error was encountered during execution of a Statement type SQL statement. The failing SQLCODE is SQLCODE.

User response: Look up the failing SQLCODE in the DB2 UDB for z/OS Messages and Codes manual. If the message was issued when processing a user-specified SQL statement, make adjustments to the statement and retry the operation.

FMNDB101 SQL error message

Explanation: An FM/DB2 batch function encountered an error during execution of an SQL statement. The formatted SQL error messages is shown in SQL error message.

User response: Look up the failing SQLCODE in the DB2 Messages and Codes manual. If the message was issued when processing a user-specified SQL statement, make adjustments to the statement and retry the operation.

FMNDB107 Remote server not found (short). FM/DB2 encountered an SQL error SQLCODE attempting to access the DB2 system at DB2 remote server. This might indicate that the remote DB2 system is not available or does not exist. Check the definitions in SYSIBM.Locations. (long).

Explanation: FM/DB2 attempted to connect to a remote DB2 server name, This may have been entered in the location field of various FM/DB2 function entry panels. The connection attempt failed with SQLCODE SQLCODE. The connection is not established and no operations at the remote DB2 server are possible until the reason for the connection failure is resolved.

User response: Look up SQLCODE in the DB2 UDB for z/OS Messages and Codes manual and follow the recommended course of action. This error can occur in the following situations (non-exhaustive list):
1. DRDA definition problems.
2. The remote DB2 system does not exist, is not active, is not available.
3. Communication problems.
FMNDB108  Different DB2 release (short). The release level of the local DB2 system is different to the release level of the DB2 system which you have just accessed. The choice of DB2 object types has been refreshed to show those object types supported on the remote or local system. Press ENTER again to get the requested list. (long).

Explanation: This message is issued by the FM/DB2 Object list utility when the user changes the value in the location field on the function entry panel and presses ENTER. FM/DB2 attempts to connect to the new location value (or to reconnect to the local server, if the new value is blank). When the DB2 version or release for the previously connected DB2 system is different to the newly connected DB2 system, it is possible that the list of DB2 object types requires changes. The object list request is processed in two steps:
1. Updating the list of objects and showing this message.
2. Displaying the list of objects for the selected object type.

User response: Ensure that the requested object type is still valid at the new DB2 system. Press Enter again to see the information for the requested object types.

FMNDB109  No matching locations (short). The location name specification does not match any locations defined to the local DB2 server. (long).

Explanation: A DB2 server name, including wild cards, was entered in the Location field on the FM/DB2 Object List utility function entry panel. There are no DB2 server names that match the wild card specification.

User response: Check the location specification and retry the operation. Use * in the location field to list all possible locations.

FMNDB110  Internal error (ABEND) Abend code.

Function terminated.

Explanation: File Manager DB2 detected a serious internal error and issued an internal ABEND to prevent further processing. The internal abend code is Abend Code

User response: This is an internal error. The list of File Manager internal abend codes is not documented for external use. Record details of the circumstances that lead to the Abend, including the Abend code, and contact support.

FMNDB118  Function active (short). The command to change the DB2 subsystem is not available while an FM/DB2 function is in progress.

Explanation: A command to change the currently connected DB2 SSID (for example SSID XXXX) was issued on a panel where the command is not available. The reason for this is that an FM/DB2 function is active.

User response: Exit the current FM/DB2 function by returning to a function entry panel, or menu panel. Reissue the command to change the DB2 system.

FMNDB119  DB2 authid not authorized (short). You are not authorized to use the DB2 authid on DB2 Subsystem DB2 SSID. (long).

Explanation: An attempt was made to change the current SQLID, using the SQLID command. The user is not authorized to use DB2 authid DB2 authid. The attempt to change the current SQLID fails, the existing value for SQLID is retained.

User response: The rules governing the use of DB2 authids can be complex, depending on whether an external security server is used to control DB2 authorizations and whether a DB2 authorization exit is in use. In most cases a user needs SYSADM authority or similar to use other SQLIDs. Contact either your DB2 System Administrator, or the site Security Administrator for assistance.

FMNDB145  Recursive alias (short). The object name entered is an alias that refers to an alias defined at the current server, which is not permitted. Specify an alias name that refers to an object, that is, not an alias. (long).

Explanation: A DB2 object name was entered in the Owner or Name fields on an FM/DB2 function entry panel. The values entered refer to an ALIAS defined at the local DB2 server, however this ALIAS refers to another ALIAS. Recursive ALIASes are not supported.

User response: An ALIAS that refers to a DB2 object (Table/View/Synonym) is supported. Resolve the ALIAS to the point where the name entered refers to an actual object and specify that value. Alternatively, specify the target object name directly.

FMNDB146  Large table mode - initial fetch of count rows.

Explanation: This message is issued by the FM/DB2 editor, when the data is first displayed. It shows the number of rows that have been fetched from DB2 and loaded into the current editor session, and also indicates that the editor is operating in "large" mode. Large editor mode is explained at length in the
Chapter 4, “Viewing and changing DB2 data,” on page 91.

Large editor mode is selected when the user enters 0 (also *, ALL) in the "Row count" field of an FM/DB2 function that uses the editor to display the result table.

User response: No action required. Informational message. In Large editor mode the number of rows in the result table is not available until the user scrolls to the end of the DB2 object. The count value in the message is approximately 3 times the number of rows that are displayed on a single screen, in TABL mode.

FMNDB148  1 row fetched.

Explanation: This message is issued by the FM/DB2 editor, when the data is first displayed. It is reserved for the situation where only 1 row was fetched from DB2.

User response: No action required. Informational message.

FMNDB149  Count of unknown

Explanation: This message is issued by the FM/DB2 editor, when operating in Large mode. Count shows the row number of the first row currently displayed (in TABL display mode), or the currently displayed row (in SNGL display mode). Large editor mode is explained at length in the File Manager User’s Guide and Reference for DB2 data. Large editor mode is selected when the user enters 0 (also *, ALL) in the "Row count" field of an FM/DB2 function that uses the editor to display the result table.

User response: No action required. Informational message. In Large editor mode the number of rows in the result table is not available until the user scrolls to the end of the DB2 object. After the user scrolls to the end of the DB2 object, 'unknown' is replaced with the number of rows in the DB2 object/result table.

FMNDB150  Table locking disabled (short). Table locking was requested but failed, as can occur with eg catalog tables. (long).

Explanation: This message is issued by the FM/DB2 editor when the data is first displayed. The user specified a table locking option (not recommended) on the 7th Editor Options panel. When FM/DB2 attempted to lock the table, the operation failed.

User response: No action required. Informational message. This message can be issued when a DB2 catalog table is selected for edit.

FMNDB153  Commit issued (short). All pending changes to the database have been successfully committed. (long).

Explanation: This message is issued by the FM/DB2 editor (only for edit) when an explicit DB2 COMMIT has been issued. The DB2 COMMIT finalizes any pending changes made in the current editor session.

User response: No action required. Informational message.

FMNDB154  Commit issued (Errors) (short).

Successful changes committed, unsuccessful changes remain pending.

(long).

Explanation: This message is issued by the FM/DB2 editor (only for edit) when an explicit DB2 COMMIT has been issued. The DB2 COMMIT finalizes any pending changes made in the current editor session. The FM/DB2 editor processes changes data in an editor session on a row-by-row basis. It is therefore possible for some, but not all, changes to have been committed.

User response: No action required. Informational message. Use the 'e' prefix command to identify rows that are in error. If in doubt as to which changes have been made, exit the current editor session and restart.

FMNDB155  Key information on (short). Key and index information will be displayed in single format (long).

Explanation: The user entered the INDEXINF primary command in an FM/DB2 editor session, and the prior state of the toggle was off. When the data is displayed in SNGL mode, an additional column appears on the left of the display, showing the columns that are part of keys (indexes) defined against the table.

User response: No action required. Informational message.

FMNDB156  Key information off (short). Key and index information will not be displayed in single format (long).

Explanation: The user entered the INDEXINF primary command in an FM/DB2 editor session, and the prior state of the toggle was on. See also message "FMNDB155."

User response: No action required. Informational message.

FMNDB157  Rollback issued (short). A DB2 rollback has been issued. Any uncommitted changes are lost. (long).

Explanation: This message is issued by the FM/DB2 editor when the user presses the PF key assigned to CANCEL, or types CANCEL, within the editor session. The current FM/DB2 editor session ends and an explicit DB2 ROLLBACK is issued. Any changes made after the last DB2 COMMIT are lost.

User response: No action required. Informational message.
FMNDB158  Auxiliary Table (short). Edit or browse of an auxiliary table is not possible. (long).

Explanation: An attempt was made to edit an auxiliary table. This action is not supported.

User response: No action required. Informational message. Only base DB2 objects can be edited.

FMNDB159  count rows fetched.

Explanation: This message is issued by the FM/DB2 editor when the data is first displayed. It shows the number of rows that have been fetched from DB2 and loaded into the current editor session. It indicates that the user specified a non zero 'row count' value, that the specified number of rows has been loaded into the editor, and that the end of the DB2 object or result table was not reached.

User response: No action required. Informational message. Not all the rows in the DB2 table have been loaded into the FM/DB2 editor. Consequently, any edit operations apply only to those rows loaded.

FMNDB160  Insufficient storage (short). There was insufficient virtual storage to edit/browse this object. Use the Rows to Retrieve option or specify a row selection criteria to limit the amount of data retrieved from DB2. (long).

Explanation: This message is issued by the FM/DB2 editor when there is insufficient memory available to load all the rows of a DB2 object or result table into memory. The editor operation cannot proceed.

User response: Review "Restricting the data that FM/DB2 loads from DB2" on page 105, covering memory usage by the FM/DB2 editor. There are various options for resolving this issue, including (non-exhaustive list):
1. Increase the TSO Region Size.
2. Use the FM/DB2 editor in "Large" mode, to minimize memory usage.
3. Specify a lower 'row count' value.
4. Use record selection criteria, WHERE clause, to reduce the number of rows that might be loaded.

FMNDB161  Warning: USA time format (short). The DB2 object being edited contains at least 1 column with the TIME data type and the system options are set to display these columns in USA format (for example, HH:MM AM|PM). You may not be able to delete or update rows in this DB2 object because of the loss of information associated with the use of this format. See the Systems Options tutorial for more information. (long).

Explanation: This message is issued by the FM/DB2 editor when processing a DB2 object, or result, table that has at least one column with the TIME data type. In addition, the FM/DB2 system options are set to display TIME columns in USA format. This format truncates the seconds part of any TIME value, making it impossible to distinguish between, for example, 09:13:52 and 09:13:00. For this reason it may be impossible to update or change data for this table.

User response: If edit operations are required for this table, consider changing the FM/DB2 system options to display TIME columns in either ISO or JIS formats, both of which show the full seconds value. For more information about the TIME display formats supported by DB2, see the DB2 UDB for z/OS SQL Reference.

FMNDB163  Copy Pending (short). Updates are not allowed to tablespace table space name in database name because the tablespace is marked as COPY PENDING. In order to edit the table you should cancel the current edit session, then take action to remove the copy pending status. One way to do this is to run the DB2 COPY utility. See the DB2 Administration Guide for a full explanation. (long).

Explanation: This message is issued by the FM/DB2 editor when an attempt is made to edit a DB2 object that is marked as COPY PENDING. The DB2 object being edited is stored in table space table space name in database name. It is not possible to change DB2 tables while they are in COPY PENDING status. COPY PENDING status can be set for a DB2 object after, for example, an initial data load, using the DB2 utilities, when certain utility options are specified.

User response: The recommended way to remove the COPY PENDING status is to run the DB2 COPY utility against the DB2 object. Consult your DB2 System Administrator for the best way to proceed.

FMNDB164  Syntax error (short). The WHERE clause contains a syntax error. (long).

Explanation: This message is issued by the template editor, or Column Selection, or Edit panel, when a
syntax error is detected in the user-specified WHERE clause.

**User response:** Examine the WHERE clause for any possible errors. You can copy the complete SQL to either the clipboard, or a temporary data set, and then use an SQL processor such as FM/DB2 options 4.3, 4.4, or SPUFI to execute the statement. Any SQL errors for the statement will also be reported.

---

**FMNDB165** Invalid column name (short). The WHERE clause contains an invalid column name, *column name*. (long).

**Explanation:** This message is issued by the template editor, or Column Selection/Edit panel, when the WHERE clause specifies an invalid column name. The reason is that the column name does not exist in the DB2 object or result table being processed.

**User response:** Check the WHERE clause and locate any occurrences of *column name*. Ensure that this column name is one of the columns in the DB2 object, or result table. Retry the operation.

---

**FMNDB166** Open quoted string (short). The WHERE clause contains an open quoted string. (long).

**Explanation:** This message is issued by the template editor, or Column Selection/Edit panel, when the WHERE clause contains an open quoted string. Examples of open quoted strings are:

1. WHERE EMPNO = '
2. WHERE EMPNO =" 

**User response:** Check the WHERE clause, and count the number of apostrophes and double quotes. An odd number for either indicates a problem. Make corrections and retry the operation. The rules for quoted strings can be found in the [DB2 UDB for z/OS SQL Reference](#).

---

**FMNDB167** SQLCODE SQLCODE (short). Execution of the SQL statement will result in SQLCODE SQLCODE. (long).

**Explanation:** This message is issued by the template editor, or Column Selection and Edit panel, during validation of a user specified WHERE clause. The validation process used by FM/DB2 is to prepare a SELECT statement against the DB2 object being processed after appending any user specified WHERE clause. The prepare can fail for many reasons, for example syntax errors and incorrect column names. The prepare step may also fail for other reasons, in which case the failing SQLCODE is returned to the user for action.

**User response:** Look up the SQLCODE in the [DB2 UDB for z/OS Messages and Codes](#) manual and respond as directed.

---

**FMNDB170** Dependent rows (short). This related edit session shows only those rows that would be affected by the change to parent table *Parent table name*. The foreign key columns are marked with a K in the scale line, and are shown on the left of the display in table display format. In general, the foreign key value must be changed, or the row deleted, for the operation on the parent table to proceed.

**Explanation:** This message is displayed when the data for a new related edit session is first displayed, and the new table is a dependent table.

**User response:** The reason for editing a dependent table is to resolve foreign key values that are preventing the deletion of the corresponding primary key in the parent table. In some situations the dependent edit session will show all rows, not just those with the foreign key values that match the primary key value in the parent table.

---

**FMNDB171** Parent table (short). In this related edit session the primary key columns are marked with a P in the scale line and are shown on the left of the display (table display format). You can use the IP prefix command to insert a new row with the primary key set to the foreign key value entered in the edit session of dependent table *dependent table name*. (long).

**Explanation:** This message is displayed when the data for a new related edit session is first displayed, and the new table is a parent table.

**User response:** The reason for editing the parent table is to add a new primary key, so that a row in a dependent table with a new foreign key, that matches the primary key, can be added.

---

**FMNDB173** All rows (short). This related edit session shows all rows of the table. Parent table *parent table name* is subject to a self referencing constraint such that deleting or changing a primary or parent key value may result in the recursive deletion of other rows in the parent table. FM/DB2 cannot determine which rows in table *dependent table name* will be affected by such an operation. (long).

**Explanation:** This message is displayed when the data for a new related edit session is first displayed, and the new table is a dependent table.

**User response:** The reason for editing a dependent table is to resolve foreign key values that are preventing the deletion of the corresponding primary key in the parent table.
FMNDB174 Commit issued (short). A DB2 commit has been issued. (long).
Explanation: This message is issued at the end of an FM/DB2 editor session. It indicates that a DB2 COMMIT has been issued to finalize any pending changes to the DB2 object being processed.
User response: No action required. Informational message.

FMNDB175 Invalid value (short). Valid values are TABL and SNGL. There are no unformatted edit modes in FM/DB2. (long).
Explanation: The user typed an invalid value in the Format field in an FM/DB2 editor session.
User response: Check the value that was entered. There are two display modes when using the FM/DB2 editor:
1. TABL (Can be abbreviated to T, TA, TAB).
   In this mode each horizontal line on the screen shows one row of data from the DB2 object or result table being processed.
2. SNGL (Can be abbreviated to S, SN, SNG).
   In this mode the data for a single row of the DB2 object or result table being processed is displayed. The data for each column is shown, with the column name on the left and the data for that column on the right.

FMNDB176 Object/Alias not found (short). The specified object or alias was not found in the catalog, or the target of an alias does not exist.
Explanation: The user entered the name of a DB2 object in the "Owner" and "Name" fields on an FM/DB2 function entry panel. FM/DB2 attempted to resolve the name and the attempt failed. The reason this message is issued is described in the following example:Alias ALIAS is defined, this refers to OBJECT. However, OBJECT does not exist. The user enters ALIAS in the "Owner" and "Name" fields on an FM/DB2 function entry panel. FM/DB2 resolves the ALIAS name to OBJECT. However, the latter does not exist, resulting in this error message.
User response: Determine why the DB2 Object Name does not exist.

FMNDB177 Warning: USA time format (short). The DB2 object being exported contains at least 1 column with the TIME data type and the system options are set to externalize these columns in USA format (HH:MM AM|PM). The exported data will not contain seconds information because of the loss of information associated with the use of this format. See the Systems Options tutorial for more information. (long).
Explanation: This message is issued by either the FM/DB2 export, or FM/DB2 UNLOAD utilities when:
1. The DB2 object being processing has a TIME data type column.
2. The "External Format for TIME Data Type Columns" system option is set to "DB2 DSNHDECP setting".
3. The DSNHDECP setting for TIME columns is 'USA'.
User response: This is an informational message. As noted in the long message text, in this situation the exported data for the TIME columns will lose any seconds value. If this is intended or desired, then no action is required. When no loss of information is required, consider changing the "External Format for TIME Data Type Columns" value to either 'ISO' or 'JIS', both of which include the seconds information.

FMNDB178 Warning: USA time format (short). The DB2 object being copied contains at least 1 column with the TIME datatypen and the system options are set to externalize these columns in USA format (HH:MM AMPM). The data copied to the target table will not contain second's information because of the loss of information associated with the use of this format. See the Systems Options tutorial for more information. (long).
Explanation: This message is issued by the FM/DB2 copy utility when:
1. The DB2 object being processing has a TIME data type column.
2. The "External Format for TIME Data Type Columns" system option is set to "DB2 DSNHDECP setting".
3. The DSNHDECP setting for TIME columns is 'USA'.
User response: This is an informational message. As noted in the long message text, in this situation the copied data for the TIME columns will lose any seconds value. If this is intended or desired then no action is required. When no loss of information is required, consider changing the "External Format for TIME Data Type Columns" value to either 'ISO' or 'JIS', both of which include the seconds information.
**FMNDB179** Cancel recommended (short). DB2 has rolled back the current unit of work (SQLCODE -911) because of a deadlock with another task. ALL UNCOMMITTED CHANGES made during the current edit session have been lost. Consequently, the data displayed in the current edit session may not reflect the current state of the DB2 table. It is recommended that you cancel the current edit session and restart with a fresh copy of the data.

**Explanation:** This message is issued by the FM/DB2 editor when it receives an SQLCODE-911 error attempting to change data for a row that was modified by the user. DB2 detected a deadlock with another task and resolved the situation by rolling back the changes for the FM/DB2 user. This means that any changes made since the last DB2 COMMIT point have been lost. In this situation there is no guarantee that the data displayed in the FM/DB2 editor still reflects the data stored in DB2. It is best to cancel the current editor session and restart.

**User response:** Cancel the current edit session and restart. Make a change to a single row and attempt to save the change. If the SQLCODE-911 reoccurs, contact your DB2 System Administrator for assistance in determining why a deadlock is occurring.

**FMNDB180** Related edit failed (short). Initialization of the related edit session failed. The reason is that the product has not been installed correctly. If LIBDEFS are used to allocate the File Manager/DB2 libraries, the related edit session exec, FMN2RESS, should be installed in the same library as the exec used to allocate the libraries for, and start, the first File Manager/DB2 session. Refer to the File Manager Customization Guide for further information.

**Explanation:** The initialization process for the related edit session failed. Each related edit session is effectively a new invocation of FM/DB2. Once started the user can use ISPF commands to switch between the different logical sessions. During the invocation process a REXX EXEC, FMN2RESS, is called. However, this REXX EXEC may not be accessible when LIBDEFS are used to define the FM/DB2 libraries.

**User response:** Contact the person who installed FM/DB2.

**FMNDB181** Browse substituted (short). The object being edited is a read-only view. In DB2 parlance a read-only view is any view that has at least one column marked as not updateable in the catalog. For this view every column is marked as not updateable. Consequently no edit operations of any kind are possible, and browse has been substituted for edit.

**Explanation:** This message is issued by the FM/DB2 editor when processing a view that cannot be updated. While the user selected the edit (as opposed to view/browse) to process the object, no edit operations are possible. To avoid any possible confusion the edit session has been converted to browse.

**User response:** No action required. Informational message.

**FMNDB182** Table locking failed (short). Table locking was requested but failed. The object being edited is a view, and views cannot be explicitly locked.

**Explanation:** This message is issued by the FM/DB2 editor when the data for a view is first displayed. The user has set a table locking option in the editor options. This is not recommended. Only tables can be explicitly locked, therefore the attempt to lock the view failed.

**User response:** No action required. Informational message. Table locking should be avoided because of the potential for deadlocks.

**FMNDB183** Read-only view (short). The object being edited is a read-only view. In DB2 parlance a read-only view is any view that has at least one column marked as not updateable in the catalog. For this view at least one column is marked as updateable. Those columns that are not updateable are protected. Some edit operations may fail.

**Explanation:** The object being processed by the FM/DB2 editor is a read-only view, although it does have one or more columns marked as updateable.

**User response:** This is an informational message. It is possible to successfully edit some read-only views, even though some columns of the view cannot be updated. However, attempting to edit a read-only view is not advisable. Edit the underlying tables directly.

**FMNDB184** Non-updateable columns (short). The table being edited contains at least one column marked as not updateable in the DB2 catalog. This can occur when editing DB2 catalog tables. The not updateable columns are protected. Some
edit operations may fail. (long).

Explanation: The object being edited contains one or
more columns marked as not updateable in the DB2
catalog SYSIBM.SYSCOLUMNS.UPDATES = 'Y'. These
columns will appear in the editor session as protected
fields. One reason for this message is when edit is used
to display the data from a DB2 catalog table.

User response: This is an informational message. You
will not be able to change the data for those columns
marked as not updateable. It may also not be possible
to delete or insert rows.

FMNDB185 Recursive alias (short). The object name
entered is an alias that refers to an alias
at a remote location, which is not
permitted. Specify the alias name at the
remote server directly. (long).

Explanation: The user entered an alias name in the
"Owner" and "Name" fields on an FM/DB2 function
entry panel. FM/DB2 resolved the alias name to
another alias name defined at a remote location. This is
not supported. After resolution, any alias name entered
must refer to an actual DB2 object defined either
locally, or remotely.

User response: Enter the target DB2 object name
directly, or ensure that any alias used refers directly to
a DB2 object.

FMNDB186 Not available (View) (short). The RE
command is not available when editing
a DB2 view. Edit the DB2 table directly
to access related table edit processing.

(long).

Explanation: The user entered the RE (Related edit)
command in an FM/DB2 editor session. The object
being edited is a view; related edit functionality is only
available for tables. The operation cannot be performed.

User response: If related edit processing is required,
ensure that all objects referred to are tables.

FMNDB187 Command not available (short).
Commands such as INSERT, REPEAT
and COPY are not available in this edit
session. The data for the DB2 object is
stored in UNICODE format, but there is
no conversion routine for graphic data
type columns. (long).

Explanation: This message is issued by the FM/DB2
editor when processing a DB2 object with graphic data
type columns that is encoded in Unicode. In this
situation it is possible for data corruption to occur
when processing data for these columns and initializing
the field contents. To avoid this situation, the INSERT,
REPEAT, and COPY commands are disabled.

User response: Collect information including the DDL.
used to create the DB2 object being processed. Contact IBM Support.

FMNDB191 Changes discarded (short). The invalid change to the Initial Display option has been discarded. (long).

Explanation: The user made an invalid change to the "Initial display format" editor option, and then pressed the PF key assigned to END, or typed END. The invalid value is discarded and the editor options panel is redisplayed.

User response: No action required. Informational message. After making changes to the options, press ENTER to validate the changes before attempting to exit the panel.

FMNDB192 Changes discarded (short). The invalid change to the Concurrency option has been discarded. Enter one of the listed values for the Concurrency option. (long).

Explanation: The user made an invalid change to the "Concurrency" editor option, and then pressed the PF key assigned to END, or typed END. The invalid value is discarded and the editor options panel is redisplayed.

User response: No action required. Informational message. After making changes to the options, press ENTER to validate the changes before attempting to exit the panel.

FMNDB193 Changes discarded (short). The invalid change to the Enter key usage option has been discarded. Enter one of the listed values for the Enter key usage option. (long).

Explanation: The user made an invalid change to the "Enter key usage" editor option, and then pressed the PF key assigned to END, or typed END. The invalid value is discarded and the editor options panel is redisplayed.

User response: No action required. Informational message. After making changes to the options, press ENTER to validate the changes before attempting to exit the panel.

FMNDB194 Use unique characters (short). The character used to indicate the "String delimiter (display)" must be different to the character used to indicate the "String delimiter (input)". Select a different character. (long).

Explanation: The user specified the same character for the "Input" and "Display" string delimiters on the 5th Editor options panel. While most characters can be used, the characters that are used must be distinct.

User response: Ensure that the characters specified for the "Input" and "Display" indicators and delimiters are unique.

FMNDB195 Use unique characters (short). The character used to indicate the "Null column display indicator" must be different to the character used to indicate the "Null column input indicator". Select a different character. (long).

Explanation: The user specified the same character for the "Input" and "Display" Null column indicators on the 5th Editor options panel. While most characters can be used, the characters that are used must be distinct.

User response: Ensure that the characters specified for the "Input" and "Display" indicators and delimiters are unique.

FMNDB196 Use unique characters (short). The character used to indicate the "Null column input indicator" must be different to the character used to indicate the "String delimiter (input)". Select a different character. (long).

Explanation: The user specified the same character for the "Input" Null column indicator and "Input" String delimiter. While most characters can be used, the characters that are used must be distinct.

User response: Ensure that the characters specified for the "Input" and "Display" indicators and delimiters are unique.

FMNDB197 Use unique characters (short). The character used to indicate the "Null column display indicator" must be different to the character used to indicate the "String delimiter (input)". Select a different character (long).

Explanation: The user specified the same character for the "Display" Null column indicator and "Input" String delimiter. While most characters can be used, the characters that are used must be distinct.

User response: Ensure that the characters specified for the "Input" and "Display" indicators and delimiters are unique.

FMNDB198 Use unique characters (short). The character used to indicate the "Null column input indicator" must be different to the character used to indicate the "String delimiter (display)". Select a different character (long).

Explanation: The user specified the same character for
the "Input" Null column indicator and "Display" String delimiter. While most characters can be used, the characters that are used must be distinct.

**User response:** Ensure that the characters specified for the "Input" and "Display" indicators and delimiters are unique.

FMNDB199 Use unique characters (short). The character used to indicate the "Null column display indicator" must be different to the character used to indicate the "String delimiter (display)". Select a different character. (long).

**Explanation:** The user specified the same character for the "Display" Null column indicator and "Display" String delimiter. While most characters can be used, the characters that are used must be distinct.

**User response:** Ensure that the characters specified for the "Input" and "Display" indicators and delimiters are unique.

FMNDB206 Not on remote object (short). The requested function can not be performed on an object at a remote location. (long).

**Explanation:** This message is issued by the FM/DB2 Object List utility in various circumstances when a requested command cannot be performed. The reason the command cannot be performed is that the object is at a remote location. Certain SQL commands, for example GRANT and REVOKE, may only be issued when connected to the local DB2 server.

**User response:** Connect to the remote DB2 server directly and reissue the command.

FMNDB208 Not on catalog object (short). The requested function can not be performed on a DB2 catalog object. (long).

**Explanation:** The LISTCAT line command was issued against a DB2 catalog object. For example, a table or index space. This command cannot be issued against DB2 catalog objects.

**User response:** No action required. Informational message.

FMNDB210 Invalid value (short). The valid values are 16, for an 8 byte number with 16 digit coefficient, or 34 for a 16 byte number with 34 digit coefficient. (long).

**Explanation:** This message is issued by the FM/DB2 Object functions. For a column defined with the DECFLOAT data type, a value other than 16 or 34 was entered in the 'PR' column.

**User response:** Enter either 16 for an 8-byte number, or 34 for a 16-byte number.

FMNDB215 Unsupported data type (short). This File Manager DB2 function does not support columns of the specified data type. **Supported data types are** List of data types. (long).

**Explanation:** This message is issued by the FM/DB2 Object functions. When defining the data type for a column, an invalid, or unsupported, data type value was entered.

**User response:** Check that the data type entered is spelled correctly. The list of supported data types is shown in the long version of the message. Press the PF key assigned to HELP, PF1 by default, to access. Check that the data type is supported by the version of DB2 in use. For example, the DECFLOAT data type is not supported at DB2 version 8.

FMNDB216 Length error (short). The specified length, precision, or scale is not valid for the data type of the column. (long).

**Explanation:** This message is issued by the FM/DB2 Object functions. The value entered for a length, precision or scale is invalid.

**User response:** Check the value entered. Consult the DB2 UDB for z/OS SQL Reference manual for detailed information on the permissible values for data type lengths, and the scale and precision values that may be specified.

FMNDB217 Scale > precision (short). The scale of a decimal must be less than or equal to the precision of the number. (long).

**Explanation:** This message is issued by the FM/DB2 Object functions. The combination of scale and precision values entered is invalid.

**User response:** For Decimal numbers the precision can be 1 to 31, inclusive. The scale for a number can be 0 to p, where p is the precision of the number.

FMNDB220 Non updateable field (short). A field reference in a change command refers to a column that is not updateable. (long).

**Explanation:** In the FM/DB2 editor, a change command was issued that included a field reference, or references. The target column for the change is not updateable, meaning that the proposed change is not possible.

**User response:** Ensure that the field reference identifies a column that can be changed. Non updateable columns are shown as protected fields in the editor display.
FMNDB223  Start too large (short). The start position (start position) is larger than the number of rows (number of rows) in the result table.

Explanation: This message is issued by the FM/DB2 editor. The user entered a value start position in the "Start position" field on the DB2 Edit panel. When FM/DB2 attempted to skip to the nominated start position, it reached the end of the result table. The number of rows in the result table is number of rows.

User response: Informational message. Reduce the start position value to at least number of rows and retry the operation.

FMNDB224  Start out of range (short). The first sample row is at row row number, which is larger than the number of rows, number of rows, in the result table. Reduce the start position and/or initial skip count values.

Explanation: This message is issued by the FM/DB2 editor when clustered sampling is selected in the editor options. When FM/DB2 attempted to skip to the nominated start position, determined by the value in the "start position field", and the value in the "Initial skip count fields", it reached the end of table. The number of rows in the result table is number of rows (n).

User response: Informational message. Reduce either or both of the "start positions" and "Initial skip count" (i) values, so that s+i+1 <=n.

FMNDB240  Invalid collection (short). The collection must be different from the COPY collection. (long).

Explanation: When binding a copy of a package, the collection and COPY collection values must be different.

User response: Specify different values.

FMNDB242  DASD data set required (short). This command can only be used on a row where DSNAME refers to a DASD data set. (long).

Explanation: The 'B' (Browse) command was issued against an image copy entry displayed by the FM/DB2 Object List Utility. The browse command can only be entered against entries associated with an image copy data set stored on DASD (disk). The command cannot be processed.

User response: Issue the command against an image copy data set with a non blank device type. The device type must not be a tape type.

FMNDB245  No plan (short). The command is only valid when application plan is non-blank. (long).

Explanation: The DBRM command was entered against an entry in a PLAN table. The command is only valid when the entry is associated with an application plan; that is the APPLNAME column for the row is non blank. The command cannot be processed.

User response: Ensure that the APPLNAME column value is non blank.

FMNDB246  No collection (short). The command is only valid when collection is non-blank. (long).

Explanation: The PACKage command was entered against an entry in a PLAN table. The command is only valid when the entry is associated with a package; that is the COLLID column for the row is non blank. The command cannot be processed.

User response: Ensure that the COLLID column value is non blank.

FMNDB247  No DBRM (short). The command is only valid when DBRM is non-blank. (long).

Explanation: The PACKage command was entered against an entry in a PLAN table. The command is only valid when the entry is associated with a DBRM; that is the PROGNAME column for the row is non blank. The command cannot be processed.

User response: Ensure that the PROGNAME column value is non blank.

FMNDB250  Invalid SQL identifier (short). The value entered is not a valid SQL ordinary identifier. (long).

Explanation: This message is issued in various places in the FM/DB2 Object List Utility. A value was entered for a field where an ordinary SQL Identifier is required. An example is a DB2 authid used as an "Owner" or S QLID.

User response: An SQL ordinary identifier cannot include characters other than A-Z, 0-9, _, and the national characters, and must not begin with a digit. See the DB2 SQL Reference manual for a precise definition. Refer to the Appendix "Limits in DB2 for z/OS" for a complete list of valid values for various types of DB2 items.

FMNDB270  Invalid buffer pool (short). Valid buffer pool names are: List. (long).

Explanation: A value entered for a buffer pool is invalid. List shows the valid values that can be entered.

User response: Compare the value entered against the
values in List. Change the value entered to match one of the values in List. Retry the operation.

**FMNDB271**  
**DB2 version invalid (short).** Table constraints are only available when connected to a DB2 Version 8 or later system. Current DB2 version is DB2 VRM indicator. (long).

**Explanation:** The “CON” command (Show table constraints) was issued against a list of tables in the FM/DB2 Object List utility. This command is not valid when connected to a DB2 version 7, or earlier system. The currently connected DB2 system is at DB2 VRM indicator, which is less than ‘0800’. The command cannot be executed.

**User response:** No action required. Informational message.

**FMNDB272**  
**Not valid for object (short).** "Line command" is not a valid command for this type of object. (long).

**Explanation:** This message is issued in various contexts by the FM/DB2 Object List utility when processing a list of columns. The row against which the line command was entered is of the wrong type. For example, the command applies only to columns defined with a user-defined (DISTINCT) data type. The command cannot be executed.

**User response:** No action required. Informational message. List of types contains information about the types of columns that the command can be issued against.

**FMNDB273**  
**Not valid for function (short).** "Line command" is only valid for external user-defined functions. (long).

**Explanation:** This message is issued in various contexts by the FM/DB2 Object List utility when processing a list of functions. The row against which the line command was entered is of the wrong function type. The command is not applicable to that type of function.

**User response:** The type of function is shown in the "Type of Routine" column. Enter the command against a row showing a function of the appropriate type.

**FMNDB274**  
**No upgrade done (short).** The PLAN_TABLE was already at the correct level. (long).

**Explanation:** A command was issued to upgrade a PLAN_TABLE, however the table was already at the correct level. No action was taken.

**User response:** No action required. Informational message.

**FMNDB275**  
**Not valid for column (short).** "Line command" is not a valid command for this type of column. List of types. (long).

**Explanation:** This message is issued in various contexts by the FM/DB2 Object List utility when processing a list of columns. The row against which the line command was entered is of the wrong type. For example, the command applies only to columns defined with a user-defined (DISTINCT) data type. The command cannot be executed.

**User response:** No action required. Informational message.

**FMNDB276**  
**Not valid for column (short).** Line command is not a valid command for ‘identity column’ type sequence objects. (long).

**Explanation:** The 'COM' (COMMENT) line command was issued against a row in a list of sequences displayed by the FM/DB2 Object List Utility. A comment cannot be specified for sequences that represent internal sequence objects used by DB2. The command cannot be executed.

**User response:** No action required. Informational message.

**FMNDB277**  
**Data set required (short).** This command can only be used on a row where DSNNAME refers to a data set. (long).

**Explanation:** The user issued the 'LC' (LISTCAT) command against an entry in a list on the Recovery Information panel in the FM/DB2 object list utility that did not have an associated data set. The Recovery Information panel shows all the rows in the SYSIBM.SYSCOPY catalog table. Only rows that represent full and incremental image copies are suitable targets for the LISTCAT command.

**User response:** No action required. Informational message. Ensure that the selected row represents either a full or incremental copy. The command may still fail if, for example, the data set has been deleted.

**FMNDB278**  
**Invalid criteria (short).** The criteria specified in the Additional Selection Criteria fields produced an invalid SQL SELECT statement WHERE clause. DB2 returned an SQL code of -199 to the SQL statement "SQL STATEMENT". (long).

**Explanation:** This message is issued by the FM/DB2 object list utility when the user has entered values in the "Additional selection criteria" section of the panel. This section of the panel can be used to add a simple
FMNDB295 • FMNDB302

Predicate to the SQL being generated. For example, when listing tables, show only tables that have an EDITPROC. This can be achieved by setting Column=EDPROC; Operator= ≠ Value = ". The expression entered resulted in SQLCODE-199, and cannot be processed.

User response: Look up SQLCODE-199 in the DB2 Administration Tool for z/OS User’s Guide and Reference, modify the predicate, and retry the operation.

FMNDB295 Invalid location (short). The location name is invalid or is not defined to the DB2 subsystem. (long).

Explanation: The user entered a value in a Location field on an FM/DB2 function entry panel. The attempt to connect to this remote server failed with SQLCODE-950.

User response: Look up SQLCODE-950 in the DB2 SQL Messages and Codes manual. The most likely explanation is that the value entered does not appear in column SYSIBM.LOCATIONS.LOCATIONS at the local DB2 server. If the value is incorrect then amend it and retry the operation. If the value is correct contact the person responsible for maintaining DRDA definitions for your DB2 systems.

FMNDB296 No plan table (short). Plan table owner.PLAN_TABLE does not exist. (long).

Explanation: This message is issued by the Explain Utilities functions when the user issues the ‘L’ (List plan table rows) command, but the plan table owner.PLAN_TABLE does not exist. The operation cannot be performed.

User response: The PLAN_TABLE is a user-defined table. The explain utilities allow an "owner" value to be specified. Check that this value is not preventing access to the user’s PLAN_TABLE. All PLAN_TABLE entries can be listed using the FM/DB2 Object List utility. If the PLAN_TABLE does not exist, use the ‘C’ command to create it.

FMNDB298 Incomplete quoted string (short). You entered an incomplete quoted string. DB2 returned an SQL code of -010 to the SQL statement SQL Statement. (long).

Explanation: This message is issued when a generated SQL statement contains a syntax error, specifically an open quoted string. The SQL statement cannot be executed.

User response: Examine the SQL statement to determine the source of the open quoted string. Modify the values entered to avoid the problem, retry the operation.

FMNDB299 Not COPY FULL (short). The RT command is only valid against an image copy entry. (long).

Explanation: The user issued the ‘RT’ (Recover TOCOPY) command against an entry in a list on the Recovery Information panel in the FM/DB2 object list utility that did not have an associated data set. The Recovery Information panel shows all the rows in the SYSIBM.SYSCOPY catalog table. Only rows that represent full and incremental image copies are suitable targets for the RT command.

User response: Ensure that the row selected represents either a full or incremental image copy.

FMNDB300 No previous template has been used with this DB2 object. (long).

Explanation: This message is issued by an FM/DB2 function entry panel when the user specifies "Previous" for the Template usage option. However, no template has been previously saved for the current DB2 object.

User response: You can generate and save a template for the current DB2 object using the "Generate/Replace" template usage option. Once a template has been successfully saved, use the "Previous" option to access the template, without needing to enter a template data set name or member.

FMNDB301 Invalid FM/DB2 template (short). The specified template does not represent a FM/DB2 template. Specify a valid FM/DB2 template. (long).

Explanation: This message is issued by an FM/DB2 function entry panel when the user specifies a template data set name for a DB2 object, and that the specified template should be used. However, the specified template is not a valid FM/DB2 template. There are different types of File Manager templates, depending on the source of the template. A base template cannot be used to describe a DB2 object.

User response: You can generate and save a template for the current DB2 object using the "Generate/Replace" template usage option. Once a template has been successfully saved, you can access it and possibly modify it using the "Edit template" option on the appropriate FM/DB2 function entry panel.

FMNDB302 FM/DB2 template loaded (short). File Manager/DB2 template has been successfully loaded. (long).

Explanation: This message is issued by an FM/DB2 function entry panel when the user specifies a template data set name for a DB2 object, and that the specified template should be used when processing the DB2 object. The specified template was successfully loaded, compared against the DB2 object and found to contain
FMNDB303  FMNDB312

a valid description of the DB2 object.

User response: No action required. Information message.

FMNDB303  Template not loaded (short). Loading of FM/DB2 template canceled by user. Row selection criteria for the template must be revised before proceeding. (long).

Explanation: This message is issued by an FM/DB2 function entry panel when the user specifies a template data set name for a DB2 object, and that the specified template should be used when processing the DB2 object. When FM/DB2 compared the template specified by the user with the template generated by FM/DB2, using the information in the DB2 catalog, differences were found. The "Template not matched" panel is typically shown before this message; the user pressed PF3 on the "Template not matched" panel. In most cases this error means that the template in the user-specified data set refers to a different DB2 object to that specified on the function entry panel. See also message FMNDB304.

User response: Resolve the reason for the difference between the data set template and the DB2 object, and retry the operation. Note that FM/DB2 always generates a template for any DB2 object and compares this with any user-specified template; with the former template considered to be correct. There are many reasons why the two templates may be inconsistent (for example, Different DB2 objects; Object dropped changed and recreated).

FMNDB304  Template mismatch (short). The selected template does not match the DB2 object. Press ENTER to replace selected template, or PF3/END to select another template or processing option. (long).

Explanation: This message is issued by an FM/DB2 function entry panel when the user specifies a template data set name for a DB2 object, and when the user specifies the template should be used when processing the DB2 object. When FM/DB2 compared the template specified by the user with the template generated by FM/DB2, using the information in the DB2 catalog, differences were found. The "Template not matched" panel is typically shown at the same time as this message.

User response: There are two choices: 1. Press PF3 to cancel the current function. You may then attempt to resolve why the specified template data set does not match the DB2 object and retry the operation. 2. Press ENTER to replace the template in the data set with a correct version for the DB2 object. Any customization stored in the data set template will be lost. Note that FM/DB2 always generates a template for any DB2 object and compares this with any user-specified template; with the former template considered to be correct. There are many reasons why the two templates may be inconsistent (for example, Different DB2 objects; Object dropped changed and recreated).

FMNDB305  Template not updated (short). Loading of File Manager/DB2 template canceled by user as the specified template does not match the DB2 object. Specify another template or processing option. (long).

Explanation: This message is issued by an FM/DB2 function entry panel when the user specifies a template data set name for a DB2 object, and that the specified template should be used when processing the DB2 object. When FM/DB2 compared the template specified by the user with the template generated by FM/DB2, using the information in the DB2 catalog, differences were found. The user canceled the template merge process.

User response: No action required. Informational message.

FMNDB310  No statement table (short). Plan table
Owner.DSN_STATEMNT_TABLE does not exist. (long).

Explanation: This message is issued when an attempt is made to access Owner.DSN_STATEMNT_TABLE. The user-defined table does not exist, the operation cannot be performed.

User response: Create the table using the FM/DB2 Explain Utilities function. After populating the table using the EXPLAIN command, retry the operation.

FMNDB311  No function table (short). Plan table
Owner.DSN_FUNCTION_TABLE does not exist. (long).

Explanation: This message is issued when an attempt is made to access Owner.DSN_FUNCTION_TABLE. The user-defined table does not exist, the operation cannot be performed.

User response: Create the table using the FM/DB2 Explain Utilities function. After populating the table using the EXPLAIN command, retry the operation.

FMNDB312  executed

Explanation: This message is issued by either: the FM/DB2 Enter, Execute and Explain SQL, or FM/DB2 Edit/Execute SQL (Data Set) functions, as part of a longer message.

User response: No action required. Message insert used for an informational message.
FMNDB313 • FMNDB337

FMNDB313  explained
Explanation:  This message is issued by the FM/DB2 Enter, Execute and Explain SQL function, as part of a longer message.
User response:  No action required. Message insert used for an informational message.

FMNDB330  Invalid row number (short). Either a row number has not been specified, or a specified value is invalid (long).
Explanation:  The user entered a command to process a LOB or XML column. The row number specified as part of the command is invalid. You can use cursor positioning to specify the row in which the column of interest is located - in which case a row number is not required. If cursor positioning is not used a row number is required. This message can be issued when an invalid value (not a number) is specified for the row number.
User response:  Check the syntax for the command and re-issue it.

FMNDB331  Row Row not found (short). The specified row number Row does not exist in the DB2 object, or result table. (long).
Explanation:  The user entered a command to process a LOB or XML column. The row number specified as part of the command is invalid because it does not exist in the current editor session. The command cannot be processed.
User response:  Check the row number value and ensure that it is between 1 and the maximum number of rows in the current editor session.

FMNDB332  Not on XML/LOB column (short). The cursor is not positioned on an XML/LOB column. (long).
Explanation:  The user entered a command to process a LOB or XML column. The cursor was positioned within the data area of the display and FM/DB2 attempted to use cursor positioning to identify the column and row of interest. The column where the cursor is located is not a LOB or XML column. The command cannot be processed.
User response:  Check the cursor position and ensure that it is located on a column of the appropriate type for the command.

FMNDB333  J command invalid (short). The J command is only valid in an FM/DB2 editor session of a LOB. (long).
Explanation:  The user entered the J prefix command in a File Manager edit or view session. This command is only available in an FM/DB2 component editor session when processing a LOB, that is in a view or edit session initiated using the LOBVIEW or LOBEDIT command. The command cannot be processed.
User response:  None - informational message.

FMNDB334  SP command invalid (short). The SP command is only valid in an FM/DB2 editor session of a LOB. (long).
Explanation:  The user entered the SP prefix command in a File Manager edit or view session. This command is only available in an FM/DB2 component editor session when processing a LOB, that is in a view or edit session initiated using the LOBVIEW or LOBEDIT command. The command cannot be processed.
User response:  None - informational message.

FMNDB335  Large table mode is not available when the object being processed contains LOB or XML columns. Specify a row count greater than zero to process this DB2 object. (long).
Explanation:  This message is issued on the FM/DB2 editor function entry panel when:
1. The user enters the name of a DB2 object that contains a LOB or XML column.
2. The row count value is set to 0 (also * or ALL).
This combination is not supported. Large editor mode is not available when the DB2 object contains LOB or XML columns.
User response:  Change the row count to specify a positive integer value.

FMNDB337  LOB/XML Columns (short). The result table for the SELECT statement contains either LOB or XML columns. Display of the result table is not supported. Use Browse, View or Edit of the DB2 object containing the LOB/XML columns to display, and possibly change, LOB and XML data. (long)
Explanation:  The user entered a SELECT statement using FM/DB2 options 4.1-4.4. The result table for the SELECT statement includes either LOB or XML columns. Display of the result table is not supported, the operation cannot be performed.
User response:  To display and possibly change LOB or XML data, use the FM/DB2 editor functions (Browse, View or Edit) and specify the DB2 object name that contains the LOB or XML columns.
FMNDB338 • FMNDB345

FMNDB338  XML Columns (short). Edit of objects with XML columns is only supported when the target DB2 system is at DB2 V10 NFM. The function cannot be performed. (long)

Explanation: The user attempted to process a DB2 object containing XML columns using the FM/DB2 editor. The target DB2 system is not at DB2 V10 new function mode (or later). Display of the DB2 object is not supported, the operation cannot be performed.

User response: To display and possibly change XML data, the target DB2 system must be at DB2 V10 new function mode, or later.

FMNDB339  Length error Len (short). The maximum length of the LOB Len has been exceeded. Either reduce the amount of data or cancel out of the LOB editor session. (long)

Explanation: This message is issued when attempting to save changes made in an FM/DB2 LOB editor session. The total length of the LOB data exceeds the maximum size for the LOB column. The changes made in the LOB editor session cannot be saved.

User response: Either reduce the size of the LOB data, or issue the CANCEL command to abandon any changes made. You can use the TE command in the originating editor session to see the maximum length for the LOB. You can use the LOBLEN command in the originating editor session to show the current length for a LOB column.

FMNDB340  Not a LOB column (short). The field reference does not identify a LOB column. (long)

Explanation: The user issued a command to process a LOB column (LOBLEN, LOBBRWS, LOBVIEW or LOBEDIT). The field reference provided does not identify a LOB column. The command cannot be processed.

User response: Check the field reference entered as part of the command and ensure that it identifies a LOB column. Re-issue the command.

FMNDB341  Not a LOB column (short). The column at the cursor position is not a LOB column. (long)

Explanation: The user issued a command to process a LOB column (LOBLEN, LOBBRWS, LOBVIEW or LOBEDIT), using cursor positioning to identify the row and LOB column of interest. The column identified is not a LOB column. The command cannot be processed.

User response: Ensure that the cursor is located within the bounds for a LOB column. Re-issue the command.

FMNDB342  LOB Length = Length (short). The length of the LOB data (Row number = Row number, REF# = Field Ref) is Length. (long)

Explanation: The user issued the LOBLEN command. The length of the column identified is Length. Row number is the row number for the row that includes the identified column. Field Ref is the field reference value entered in the LOBLEN command.

User response: None - informational message.

FMNDB343  No LOB columns (short). The DB2 object does not contain any LOB columns. (long)

Explanation: The user attempted to process a DB2 object containing XML columns using the FM/DB2 editor. The target DB2 object being processed does not include any LOB columns. The command cannot be processed.

User response: None - informational message.

FMNDB344  LOBVIEW not available (short). LOBVIEW is not available. You may only view a LOB column from a view or edit session (not browse) of the parent table. (long)

Explanation: The user issued the LOBVIEW command from an FM/DB2 browse session. The only LOB-related commands that are available in an FM/DB2 browse session are LOBLEN and LOBBRWS. The command cannot be processed.

User response: Use the LOBBRWS command instead. If LOBVIEW is required exit the editor session and re-invoke the editor using either the view or edit function.

FMNDB345  LOBEDIT not available (short). LOBEDIT is not available. You may only edit a LOB column from an edit session (not browse or view) of the parent table. (long)

Explanation: The user issued the LOBEDIT command from either an FM/DB2 browse, or FM/DB2 view session. The only LOB-related commands that are available in an FM/DB2 view session are LOBLEN, LOBBRWS and LOBVIEW. The command cannot be processed.

User response: Use one of the available commands as documented in the explanation. If LOBEDIT is required exit the editor session and re-invoke the editor using either the edit function.

None - informational message.
FMNDB346 Empty LOB (short). The LOB has no data. (long).

Explanation: This message is issued when the data for a LOB column is initially displayed by the FM/DB2 LOB editor. There is no data in the LOB column.

User response: None - informational message.

FMNDB348 No matching columns (short). The DB2 object contains LOB columns, however there are no selected LOB columns. You can use the TEdit command to change the selection status for a column. (long).

Explanation: The user issued a command to process a LOB column (LOBLEN, LOBBRWS, LOBVIEW or LOBEDIT), however the template for the object has been modified so that all LOB columns are not displayed. The command cannot be processed.

User response: Issue the Tedit command to display the Column Selection/Edit panel (Template editor). Change the Selection status of the LOB column of interest and save the resulting changes. Re-issue the command.

FMNDB349 BLOB edit not supported (short). Edit of a binary large object column is not supported. (long).

Explanation: The LOBEDIT command was issued against a BLOB column. Edit of BLOB data is not supported. The command cannot be processed.

User response: None - informational message.

FMNDB350 Not an XML column (short). The field reference does not identify an XML column. (long).

Explanation: The user issued a command to process an XML column (XMLLEN, XMLBRWS, XMLVIEW or XMLEDIT). The field reference provided does not identify an XML column. The command cannot be processed.

User response: Check the field reference entered as part of the command and ensure that it identifies an XML column. Re-issue the command.

FMNDB351 Not an XML column (short). The column at the cursor position is not an XML column. (long).

Explanation: The user issued a command to process an XML column (XMLLEN, XMLBRWS, XMLVIEW or XMLEDIT), using cursor positioning to identify the row and XML column of interest. The column identified is not an XML column. The command cannot be processed.

User response: Ensure that the cursor is located within the bounds for an XML column. Re-issue the command.

FMNDB352 XML Length = Length (short). The length of the XML data (Row number = Row number, REF# = Field Ref) is Length. (long).

Explanation: The user issued the XMLLEN command. The length of the column identified is Length. Row number is the row number for the row that includes the identified column. Field Ref is the field reference value entered in the XMLLEN command.

User response: None - informational message.

FMNDB353 No XML columns (short). The DB2 object does not contain any XML columns. (long).

Explanation: The user issued a command to process an XML column (XMLLEN, XMLBRWS, XMLVIEW or XMLEDIT), however the DB2 object being processed does not include any XML columns. The command cannot be processed.

User response: None - informational message.

FMNDB354 XMLVIEW not available (short). XMLVIEW is not available. You may only view an XML column from a view or edit session (not browse) of the parent table. (long).

Explanation: The user issued the XMLVIEW command from an FM/DB2 browse session. The only XML-related commands that are available in an FM/DB2 browse session are XMLLEN and XMLBRWS. The command cannot be processed.

User response: Use the XMLBRWS command instead. If XMLVIEW is required exit the editor session and re-invoked the editor using either the view or edit function.

FMNDB355 XMLEDIT not available (short). XMLEDIT is not available. You may only edit an XML column from an edit session (not browse or view) of the parent table. (long).

Explanation: The user issued the XMLEDIT command from either an FM/DB2 browse, or FM/DB2 view session. The only XML-related commands that are available in an FM/DB2 browse session are XMLLEN and XMLBRWS. The only LOB-related commands that are available in an FM/DB2 view session are XMLLEN, XMLBRWS and XMLVIEW. The command cannot be processed.

User response: Use one of the available commands as documented in the explanation. If XMLEDIT is required...
exit the editor session and re-invoke the editor using either the edit function.

**FMNDB358** No matching columns (short). The DB2 object contains XML columns, however there are no selected XML columns. You can use the TEdit command to change the selection status for a column. (long).

**Explanation:** The user issued a command to process an XML column (XMLLEN, XMLBRWS, XMLVIEW or XMLEDIT), however the template for the object has been modified so that all XML columns are not displayed. The command cannot be processed.

**User response:** Issue the Tedit command to display the Column Selection/Edit panel (Template editor). Change the Selection status of the XML column of interest and save the resulting changes. Re-issue the command.

**FMNDB359** Error line Line (short). The DB2 XML parser reported an error on or near line Line. (long).

**Explanation:** This message is issued in response to an attempt to save changes made in an XML editor session. There is a syntax error in the XML document that will prevent the updated document being saved in the DB2 object. Line indicates the line number where the syntax error is located. The XML document cannot be saved until the syntax error is corrected.

**User response:** Locate the specified line number, correct the syntax error and attempt to exit the XML edit session again. If it is not possible to resolve the error issue the CANCEL command to abandon any changes made.

**FMNDB400** Incomplete clause (short). The syntax clause just exited is not yet complete. (long).

**Explanation:** This message is issued by the FM/DB2 Advanced SELECT Prototyping function when the user exits the syntax diagram for a code fragment without making any choice.

**User response:** No action required. Informational message.

**FMNDB401** Command command ignored (short). You entered the Command command but the cursor was not on a syntax element. (long).

**Explanation:** This message is issued by the FM/DB2 Advanced SELECT Prototyping function when the user presses a PFKey assigned to a command (for example INSRPT, assigned to PF5 by default), however the cursor was not located on an applicable syntax element. The command cannot be performed.

**User response:** No action required. Informational message.

**FMNDB402** Command command ignored (short). You entered the Command command with the cursor on a syntax element but the element is not in a repeat group. (long).

**Explanation:** This message is issued by the FM/DB2 Advanced SELECT Prototyping function when the user presses a PFKey assigned to a command (for example INSRPT, assigned to PF5 by default), however the cursor was not located on an applicable syntax element. Some commands are only applicable to repeatable syntax elements. The command cannot be performed.

**User response:** No action required. Informational message.

**FMNDB403** No data entered (short). No data was entered for the data entry syntax element. (long).

**Explanation:** This message is issued by the FM/DB2 Advanced SELECT Prototyping function when the user exits a data entry syntax element without entering a value. An example is the "table" option of the "table reference clause". A DB2 table name, such as DSN8x10.EMP is required.

**User response:** No action required. Informational message. If this element is a required part of the SQL syntax you will need to enter a valid value to be able to complete the SQL SELECT statement.

**FMNDB404** Statement empty (short). You entered a scroll command but there is no generated statement data. (long).

**Explanation:** This message is issued by the FM/DB2 Advanced SELECT Prototyping function when the user enters a scroll command (or presses a PFKey assigned to a scroll command), however there is not data to scroll.

**User response:** No action required. Informational message.

**FMNDB405** End of data (short). The last line of the generated statement is already displayed. (long).

**Explanation:** This message is issued by the FM/DB2 Advanced SELECT Prototyping function when the user enters a scroll down command (or presses a PFKey assigned to scroll down), however the last line of the generated statement is already visible. No further scrolling is possible.

**User response:** No action required. Informational message.
FMNDB406  Top of data (short). The first line of the generated statement is already displayed. (long).

Explanation: This message is issued by the FM/DB2 Advanced SELECT Prototyping function when the user enters a scroll up command (or presses a PFKey assigned to scroll up), however there the first line of the generated statement is already visible. No further scrolling is possible.

User response: No action required. Informational message.

FMNDB410  Nothing to execute (short). The EXECUTE command cannot be executed because no statement has been generated for execution. (long).

Explanation: This message is issued by the FM/DB2 Advanced SELECT Prototyping function. The user entered the EXECUTE command; or pressed the PFKey assigned to EXECUTE, however no SELECT statement has been generated.

User response: Use the prototyping facilities to generate a SELECT statement prior to attempting execution.

FMNDB407  Autoscrollled (short). The generated statement has changed, and the statement generation area has been scrolled to bring the end of the data into view. (long).

Explanation: This message is issued by the FM/DB2 Advanced SELECT Prototyping function. The SELECT statement being generated changed as a result of user input. The display of the generated statement was automatically scrolled to show the last part of the statement.

User response: No action required. Informational message.

FMNDB411  Repeat separator invalid (short). There is no repeat separator associated with the last repeated element. (long).

Explanation: This message is issued by the FM/DB2 Advanced SELECT Prototyping function. An attempt was made to insert a repeat separator, however there is no repeat separator for the last repeated element.

User response: No action required. Informational message. If in doubt consult the syntax diagram for SELECT in the DB2 SQL Reference manual.

FMNDB408  Statement too long (short). The length of the generated statement exceeds the maximum length allowed for the target system: execution failure expected. (long).

Explanation: This message is issued by the FM/DB2 Advanced SELECT Prototyping function. The length of the generated SQL statement exceeds 32765 bytes, which is the limit for the function. Execution of the statement is expected to fail and should not be attempted. Note that while DB2 supports SQL statements up to 2M-1 the limit of 32K-2 is imposed by File Manager/ISPF.

User response: Cancel the current operation and restart.

FMNDB412  Nothing to execute (short). The EXECUTE command cannot be executed because no statement has been entered for execution. (long).

Explanation: This message is issued by the FM/DB2 Enter, Execute and Explain SQL function. The user entered the EXECUTE command; however the SQL Statement input area is blank.

User response: No action required. Informational message.

FMNDB409  Repeat separator invalid (short). A repeat separator cannot be entered because no elements of the repeat item have been selected. (long).

Explanation: This message is issued by the FM/DB2 Advanced SELECT Prototyping function, when the user attempted to add a repeat separator (for example AND, OR), however no repeatable items have been selected.

User response: Select at least one repeatable item before attempting to insert a repeat separator.

FMNDB413  Invalid statement type (short). An invalid statement or a statement that is not dynamically executable was found in the statement entry area. (long).

Explanation: This message is issued by the FM/DB2 Enter, Execute and Explain SQL function, when the user has entered an SQL statement that cannot be dynamically prepared/executed. An example is ‘CONNECT TO server’, which can only be coded in an application program. This statement cannot be dynamically prepared and executed. The operation cannot be performed.

User response: Consult the DB2 SQL Reference manual for information about those statements that can/cannot be dynamically prepared/executed.
### FMNDB414
**Statement type statement executed (SQL cost)**

**Explanation:** This message is issued by FM/DB2 functions after the successful execution of some SQL statement. **Statement type** is SELECT, INSERT, and so on. **Cost** is the timer on value returned by DB2 when the SQL statement was prepared, giving a relative estimate of cost of executing the statement.

**User response:** No action required. Informational message.

### FMNDB415
**Count statements executed (SQL cost)**

**Explanation:** This message is issued by the FM/DB2 Enter, Execute, and Explain SQL function after successful execution of multiple SQL statements. **Count** is the number of SQL statements that were executed. The statements may be of different types. **Cost** is the total of the timer on values returned by DB2 when each SQL statement was prepared.

**User response:** No action required. Informational message.

### FMNDB416
**TABLE**

**Explanation:** This is a message insert, used by the FM/DB2 editor.

**User response:** No action required. Used to create a larger message.

### FMNDB417
**VIEW**

**Explanation:** This is a message insert, used by the FM/DB2 editor.

**User response:** No action required. Used to create a larger message.

### FMNDB418
**RESULTS FOR**

**Explanation:** This is a message insert, used by the FM/DB2 editor.

**User response:** No action required. Used to create a larger message.

### FMNDB419
**Execution error (short). SQL execution failed. The cursor is placed on the start of the failing statement, or for a syntax error, on the token in error.**

**Explanation:** This message is issued by the FM/DB2 Enter, Execute and Explain SQL function when execution of some SQL statement fails. Depending on the type of error, the cursor is placed at the start of the failing SQL statement, or on the token within the SQL statement considered to be in error.

**User response:** For syntax errors, the cursor location is at the failing token as determined by the DB2 syntax parser. The actual error may be earlier in the statement. Example: SELECT FROM DSN8810.EMP gives SQLCODE-104 and the cursor is located on the period in “DSN8810.EMP”. The actual error is the omission of ‘*’ after the SELECT keyword. Consult the DB2 SQL Reference manual for the statement type for the syntax diagram for that statement.

### FMNDB420
**Data truncated (short). Data found beyond column 72 in the SQL edit session has been discarded. (long).**

**Explanation:** This message is issued by the FM/DB2 Enter, Execute, and Explain SQL function after the user exits from an ISPF editor session displayed in response to the SQL command. Within the ISPF editor, session changes may be made to the SQL statement. However, the statement must be entered in columns 1-72, inclusive. Any data entered in columns 73-80 is truncated (discarded).

**User response:** Examine the SQL statement to determine where the truncation occurred and make corrections.

### FMNDB421
**SQL too long (short). The SQL data returned from the SQL edit session was too large for the execution buffer. Some lines have been discarded. (long).**

**Explanation:** This message is issued by the FM/DB2 Enter, Execute and Explain SQL function after the user exits from an ISPF editor session displayed in response to the SQL command. Not all of the SQL statement entered in the ISPF editor session could be copied to the display area for the FM/DB2 Enter, Execute, and Explain SQL function. The statement is too long.

**User response:** You may be able to circumvent this problem by using less lines to code the SQL statement in the ISPF editor session. For example, by removing some of the white space in the statement.

### FMNDB422
**Nothing to edit (short). There is no SQL statement for the SQL command to edit. (long).**

**Explanation:** This message is issued by the FM/DB2 Advanced SELECT Prototyping function. The user entered the SQL command; however there was no SQL to display in an ISPF editor session. The SQL command cannot be processed.

**User response:** Enter some part of the SQL SELECT statement before entering the SQL command.
FMNDB423  Select an option (short). Either edit or execute must be selected. (long).

Explanation: This message is issued by the FM/DB2 Edit/Execute SQL (Data Set) function. Neither the "Edit data set", nor the "Execute SQL from data set" options have been selected.

User response: You must select one, or both of the "Edit data set" and "Execute SQL from data set" options to use this function.

FMNDB424  Statement type statement Action.

Explanation: This message is issued by the FM/DB2 Enter, Execute, and Explain SQL function when an SQL statement other than a SELECT, DELETE, or UPDATE is processed. Statement type is the type of statement (Example SET) and Action indicates the type of operation. For example, "executed".

User response: Enter a value appropriate for the column being included in the WHERE clause.

FMNDB425  count statements Action

Explanation: This message is issued by the FM/DB2 Enter, Execute, and Explain SQL function when multiple SQL statements are processed. count is the type of statement and Action indicates the type of operation. For example, "executed".

User response: No action required. Informational message.

FMNDB426  SQL execution failed (count statement(s) executed successfully).

Explanation: This message is issued by the FM/DB2 Enter, Execute, and Explain SQL function, after multiple SQL statements were processed. The execution of at least one SQL statement failed, however count statements were processed successfully.

User response: Examine the list of SQL statements processed to determine whether the failure to execute a statement (or statements) is appropriate.

FMNDB427  No columns selected (short). The EXECUTE command cannot be executed because no columns have been selected from which to retrieve data. (long).

Explanation: This message is issued by the FM/DB2 Basic SELECT Prototyping function. The user attempted to execute the SQL SELECT statement. However, the SELECT statement is incomplete and cannot be executed. There are no columns specified in the SELECT clause.

User response: You need to specify at least one column in the SELECT statement, for the statement to be executable. Select a column by typing 'S' next to the entry for the column, or use 'ALL' to select all columns.

FMNDB429  Value required (short). A value must be entered to build a where clause predicate. (long).

Explanation: This message is issued by the FM/DB2 Basic SELECT Prototyping function. The user entered a value in the 'Op' (Operator) column for a column, but no value was entered in the corresponding 'Value' column. A value is required to construct a valid WHERE clause, which has the form: COLUMN operator value.

User response: Enter a value appropriate for the column being included in the WHERE clause.

FMNDB431  Invalid operator (short). Enter a valid logical operator: AND, OR, NOT, NOR. NOT represents AND NOT and NOR represents OR NOT. (long).

Explanation: This message is issued by the FM/DB2 Basic SELECT Prototyping function. The user entered an invalid value in the 'LOp' (Logical Operator) column for a column. The LOp column is used to specify a logical operator. This is used between any WHERE clause predicates specified for the SQL statement being prototyped.

User response: Enter one of the values shown in the long version of the message; the default value is 'AND', selected automatically when no value is specified.

FMNDB432  Invalid operator (short). Enter a valid expression operator: =, ≠, <, >, <=, >=, IN, NI, LI, NL. LI represents the SQL LIKE operator, NL represents NOT LIKE, and NI represents NOT IN. (long).

Explanation: This message is issued by the FM/DB2 Basic SELECT Prototyping function. The user entered an invalid value in the 'Op' (Operator) column for a column. The Op column is used to specify a comparison operator. This is used in the WHERE clause predicate being added to the SQL statement being prototyped.

User response: Enter one of the values shown in the long version of the message; the default value is '=' selected automatically when no value is specified.

FMNDB433  Invalid margins (short). The left margin must be less than or equal to the right margin, and both values must be greater than zero. (long).

Explanation: This message is issued by the FM/DB2 Edit/Execute SQL (Data Set) function. The user entered values in the "First column" and "Last column" fields that are invalid. Suggested values are:
User response: No action required. Informational message.

FMNDB437 Statement executed, no rows updated.

Explanation: This message is issued by the FM/DB2 Enter, Execute, and Explain SQL function. The user entered an UPDATE SQL statement, and the statement was executed. However, DB2 reported that no rows were updated.

User response: No action required. Informational message.

FMNDB438 SQL execution halted (short). Execution of the SQL statement was halted because the user pressed CANCEL on the predictive governor. (long).

Explanation: This message is issued by the FM/DB2 Enter, Execute, and Explain SQL function. The execution of an SQL statement was halted because it exceeds values set by the SQL predictive governor. In this situation a warning pop-up panel is displayed, and the user elected to CANCEL execution of the statement.

User response: No action required. Informational message.

FMNDB440 VIEW (clustered sampling)

Explanation: This message is displayed by the FM/DB2 Edit/Execute SQL (Data Set) function. The input data set contains a CONNECT SQL statement. This statement cannot be dynamically prepared and executed. It can only be imbedded in an application program. The statement cannot be executed.

User response: Remove the CONNECT statement and retry the operation.

FMNDB441 TABLE (clustered sampling)

Explanation: This message is displayed by the FM/DB2 Editor, when using clustered sampling.

User response: No action required. Informational message.
FMNDB442  VIEW (random sampling)
Explanation: This message is displayed by the FM/DB2 Editor, when using random sampling.
User response: No action required. Informational message.

FMNDB443  TABLE (random sampling)
Explanation: This message is displayed by the FM/DB2 Editor, when using random sampling.
User response: No action required. Informational message.

FMNDB444  LOB
Explanation: This message is displayed by the FM/DB2 Editor, when processing LOB data.
User response: No action required. Informational message.

FMNDB445  XML
Explanation: This message is displayed by the FM/DB2 Editor, when processing XML data.
User response: No action required. Informational message.

FMNDB450  Copy not allowed (short). The index has not been defined with COPY YES, so it cannot be selected for copying. (long).
Explanation: This message is issued by the FM/DB2 "DB2 Utilities" function. The user selected an index created with the 'COPY NO' option for copy. This is not allowed.
User response: You can only copy a DB2 index when it was originally created with the 'COPY YES' option. See the DB2 Utilities Guide and Reference for more information.

FMNDB451  Chng limits not allowed (short). Change limits cannot be specified because the table space has not been defined with TRACKMOD YES. (long).
Explanation: This message is issued by the FM/DB2 "DB2 Utilities" function. The user entered change limit values for a table space defined with 'TRACKMOD NO'. This is not allowed.
User response: You can only specify change limit values for the DB2 COPY utility when the table space was defined with 'TRACKMOD YES'. See the DB2 Utilities Guide and Reference for more information.

FMNDB453  Cannot recover object (short). Either the DB2 object specified does not exist in the DB2 catalog, or no SYSCOPY entries exist for the DB2 object. (long).
Explanation: This message is issued by the FM/DB2 "DB2 Utilities" function. The user is attempting to recover either a table space, or index. When FM/DB2 retrieved a list of entries from SYSIBM.SYSCOPY, no rows were returned. This might occur when wildcard values for database name and table and index space name were specified on the function entry panel. There are no matching table and index spaces. Another possibility is that there are matching table and index spaces, but no suitable image copies are available for these objects, so the RECOVER utility cannot execute.
User response: No action required. Informational message.

FMNDB454  Invalid change limit (short). The change limit must be an integer value between 0 and 100. (long).
Explanation: This message is issued by the FM "DB2 Utilities" function. Change limit values for the COPY utility are expressed as percentage values (0-100 inclusive). The value entered is out of range.
User response: Specify a value in the range 0-100 inclusive. See the DB2 Utilities Guide and Reference for more information.

FMNDB455  Function canceled (short). Load utility processing was canceled by user. (long).
Explanation: This message is issued by the FM/DB2 "DB2 Utilities" function. The user pressed the PFKey assigned to CANCEL (or typed the CANCEL command), when processing for a LOAD utility job, using an FM template, was incomplete.
User response: No action required. Informational message.

FMNDB456  Function completed (short). Load utility processing has completed. (long).
Explanation: This message is issued by the FM/DB2 "DB2 Utilities" function. Processing for a DB2 LOAD utility completed normally.
User response: No action required. Informational message.

FMNDB457  Choose at least one item (short). For the given processing option, you must choose at least one item. (long).
Explanation: This message is issued by the FM/DB2 "OPTIONS Options" panel. The user selected a processing option of 4 "Options to be included", but
The source DB2 column \textit{Column name} is not nullable and yet the corresponding target data set field is associated with a nullable field. (long).

**Explanation:** This message is issued by the FM/DB2 Unload utility. During mapping between the columns of the source object and fields of the output template, a mismatch for column \textit{Column name}, and the corresponding field, has been detected. \textit{Column name} is defined as NOT NULL. However, the target field has an associated null field indicator.

**User response:** This usually indicates a mapping problem. The target field for a column defined as "NOT NULL" should not have a null field indicator. Check the target template copybook or template definition and make changes. Retry the operation.

---

**FMNDB458** Null indicator mismatch (short). The source DB2 column \textit{Column name} is not nullable and yet the corresponding target data set field is associated with a nullable field. (long).

**Explanation:** This message is issued by the FM/DB2 Unload utility. During mapping between the columns of the source object and fields of the output template, a mismatch for column \textit{Column name}, and the corresponding field, has been detected. \textit{Column name} is defined as NOT NULL. However, the target field has an associated null field indicator.

**User response:** This usually indicates a mapping problem. The target field for a column defined as "NOT NULL" should not have a null field indicator. Check the target template copybook or template definition and make changes. Retry the operation.

---

**FMNDB459** Enter required field (short). The field where the cursor is positioned is mandatory. (long).

**Explanation:** This message is issued by the FM/DB2 "OPTIONS Options" panel. The user selected a processing option of 3 (KEY), but did specify a key value.

**User response:** Specify a key value in the field indicated by the cursor position.

---

**FMNDB460** Function canceled (short). Unload table utility processing was canceled by user. (long).

**Explanation:** This message is issued by the FM/DB2 Unload utility, when the user presses the PFKey assigned to CANCEL (or types the CANCEL command), prior to completion of the function.

**User response:** No action required. Informational message.

---

**FMNDB461** Function completed (short). Unload utility processing has completed. (long).

**Explanation:** This message is issued by the FM/DB2 Unload utility, when the function completes normal processing.

**User response:** No action required. Informational message.

---

**FMNDB462** Settings failure (short). The TEMPLATE, LISTDEF, or OPTIONS settings for the current subsystem may be incorrect. (long).

**Explanation:** This message is issued during FM/DB2 options initialization. An error was encountered when storing values in an ISPF table in the user ISPF profile.

**User response:** Logoff completely, logon again, and retry the operation. If the problem persists, try deleting all FMN2* members from the user ISPF profile data set. If this fails to resolve the problem contact IBM support.

---

**FMNDB463** Setting failure (short). When you return from another subsystem, the changes that you made to LISTDEF, TEMPLATE, or OPTIONS settings may not be restored. (long).

**Explanation:** This message is issued during FM/DB2 options initialization. An error was encountered when storing values in an ISPF table in the user's ISPF profile.

**User response:** Logoff completely, logon again, and retry the operation. If the problem persists, try deleting all FMN2* members from the user ISPF profile data set. If this fails to resolve the problem contact IBM support.

---

**FMNDB464** Null indicator mismatch (short). A null indicator byte maps to field name \textit{field name (column number)}, but the name does not contain 'NULL'. (long).

**Explanation:** This message is issued by the FM/DB2 Unload utility. A problem has been detected when mapping columns of the DB2 object to corresponding fields in the File Manager (base) template for the unloaded data.

**User response:** Refer to the online tutorial topic "Null indicator byte processing", accessed as follows: 3.9.7.1.7 (Utilities > Utilities > Unload Utility > Unload (table) > Null indicator byte processing. Briefly, when using a File Manager (base) template to describe the format for the UNLOAD data set, it is possible to use a field with length > 1 byte for a DB2 null indicator. However, the field name must include the string 'NULL' to be automatically mapped.

---

**FMNDB465** Null indicator mismatch (short). A null indicator byte maps to the byte before the first field. (long).

**Explanation:** This message is issued by the FM/DB2 Unload utility. This is a "should not occur" error.

**User response:** Check the template for the target (unload) data set is appropriate for the DB2 object being unloaded. Consider using an FM/DB2 template generated from the source object to map the unloaded data. If both of these techniques fail to resolve the
problem, contact IBM support.

**FMNDB467** Null indicator mismatch (short). The null indicator byte `<column name>` maps to field `<field name>` (column `<number>`) which is mapped already to another table column. (long).

**Explanation:** This message is issued by the FM/DB2 Unload utility. A problem has been detected when processing the mapping for a DB2 null indicator for column `<column name>`.

**User response:** Refer to the online tutorial topic "Null indicator byte processing accessed as follows: 3.9.7.1.7 (Utilities > Utilities > Unload Utility > Unload (table) > Null indicator byte processing. Briefly, a field in the template for the unload data set can only be mapped to a single column, or DB2 null indicator.

**FMNDB468** Utility service failed (short). FM DB2 initiated an unload table utility operation from export. However, the unload table utility failed to initialize properly. (long).

**Explanation:** This message is issued when there is a problem generating a JCL deck using ISPF file tailoring. This is a "should not occur" error under normal circumstances.

**User response:** Contact the person responsible for installing and maintaining FM/DB2. Possible causes of this error include set up related issues such as the SFMNSLIB library not being available to the FM/DB2 user.

**FMNDB469** Settings failed (short). The TEMPLATE, LISTDEF, or OPTIONS settings (for subsystem `DB2 SSID`) cannot be loaded or the associated settings could not be found. (long).

**Explanation:** This message is issued during FM/DB2 options processing. Once cause of this error is a missing options module (FMN2POPT), or no definitions for the current DB2 subsystem.

**User response:** Contact the person who is responsible for installing/maintaining FM/DB2. Defining FMN2SSDM macro entries for every DB2 system accessed by FM/DB2 is a required installation and customization step. Contact IBM support if analysis shows no obvious problem in the FMN2POPT module.

**FMNDB470** Init install settings (short). The installation-defined settings for this subsystem are used to set options TEMPLATE, LISTDEF, and OPTIONS. (long).

**Explanation:** This message is issued during FM/DB2 options processing. When processing TEMPLATE, LISTDEF or OPTIONS, no options had previously been saved in the user's ISPF profile. The option values have been initialized based on any values set in the FMN2POPT (installation) module.

**User response:** No action required. Informational message.

**FMNDB471** Init FM DB2 settings (short). The installation defined setting for this subsystem is: CONNECT = ANY. The FM/DB2 defaults are used for options TEMPLATE, LISTDEF, and OPTIONS. (long).

**Explanation:** This message is issued during FM/DB2 options processing. When processing the connection to a DB2 subsystem, the combination of CONNECT=ANY (FMN2POPI macro parameter) and no FMN2SSDM macro entry for the DB2 subsystem was detected. In this situation there can be no installation settings for TEMPLATE, LISTDEF and OPTIONS options (since there is no FMN2SSDM macro entry where these settings are defined). The FM/DB2 defaults are used to initialize the option values.

**User response:** No action required. Informational message. The FM/DB2 Installation and Customization Guide lists defining an FMN2SSDM macro entry for every DB2 system that FM/DB2 may connect to as a required step.

**FMNDB472** Init FM DB2 settings (short). FM DB2 default settings for options TEMPLATE, LISTDEF, and OPTIONS are used. Your installation defined settings uses an old version of the FMN2SSDM macro. (long).

**Explanation:** This is a "should not occur" error. The FMN2POPT module in use is out-of-date and should be reassembled with the latest macro versions.

**User response:** Contact the person responsible for installing or maintaining FM/DB2. Reassemble the FMN2POPT module using the directions in the File Manager Installation and Customization Guide. Carefully check the SYSLIB DD statement to ensure that the current version of SFMNSAM1 libraries is in use, and check the SYSLMOD DD statement to ensure that the output module is located in the appropriate library.

**FMNDB473** Must provide name (short). A LISTDEF name must be provided when specification LISTDEF is selected. (long).

**Explanation:** The user selected the 'LISTDEF' specification on the FM/DB2 "DB2 Utilities" function entry panel. However, no value was entered in the
“LISTDEF name” field. The use of a LISTDEF specification requires a non-blank LISTDEF name.

User response: Enter a valid LISTDEF name, and retry the operation; or change the specification to "Object name".

FMNDB474 //** NULLABLE COLUMN MAPPING

Explanation: This is a message insert used by the FM/DB2 Unload utility when preparing JCL.

User response: No action required. This message may be issued when documenting the mapping of nullable DB2 columns in the DB2 object to corresponding fields in the template for the unloaded data.

FMNDB477 //** F (FOOTNOTE) -

Explanation: This is a message insert used by the FM/DB2 Unload utility when preparing JCL.

User response: No action required. This message may be issued when documenting the mapping of nullable DB2 columns in the DB2 object to corresponding fields in the template for the unloaded data.

FMNDB478 //** *= FIELD CORRESPONDING TO NULL INDICATOR BYTE IS LARGER

Explanation: This is a message insert used by the FM/DB2 Unload utility when preparing JCL.

User response: No action required. This message may be issued when documenting the mapping of nullable DB2 columns in the DB2 object to corresponding fields in the template for the unloaded data.

FMNDB479 //** THAN ONE BYTE. INDICATOR MAPS TO LAST BYTE.

Explanation: This is a message insert used by the FM/DB2 Unload utility when preparing JCL.

User response: No action required. This message may be issued when documenting the mapping of nullable DB2 columns in the DB2 object to corresponding fields in the template for the unloaded data.

FMNDB480 WHERE clause too long (short). The WHERE clause is longer than 32767 bytes, and cannot be processed. Specify a shorter WHERE clause. (long).

Explanation: This message may be issued when exiting the ISPF edit session used to specify a WHERE clause in the FM/DB2 template editor (Column Selection/Edit panel). The WHERE clause exceeds the maximum allowable length of 32767 bytes, and cannot be processed.

User response: Specify a shorter WHERE clause.

FMNDB481 Invalid reference (short). The correlation reference Reference is invalid. Valid values are #1 - Maximum reference inclusive. (long).

Explanation: This message is issued by the FM/DB2 Basic SELECT Prototyping function. When using an explicit table correlation reference (example #n, n=1, 2 ..) the n must not be greater than the number of DB2 objects entered when the function was first started. “n” must also be a numeric value, example #F is invalid.

User response: Correct the table correlation reference, and retry the operation.

FMNDB482 Invalid column name (short). Column name is not a column of the referenced table. (long).

Explanation: This message is issued by the FM/DB2 Basic SELECT Prototyping function. A qualified column name was entered in the "Value" column, presumably to indicate a join between two columns. The table correlation reference is valid but the column name is not.

User response: Check the value entered for spelling or other errors. Check that the column name is a column of the table indicated by the table correlation reference.

FMNDB483 Ambiguous reference (short). This column name appears in more than one of the selected tables. Use a correlation reference, eg #1.COLNAME to identify the required. (long).

Explanation: This message is issued by the FM/DB2 Basic SELECT Prototyping function. A value entered in the "Value" column matches more than one column name in the DB2 objects selected for prototyping. A table correlation reference needs to be used to remove any ambiguity.

User response: Precede the column name with a table correlation reference, example #n., where “n” is a positive integer.

FMNDB484 Ambiguous input (short). The value specified is a column name, but can also be interpreted as a constant. To specify a constant enclose the value in quotes. To specify a column use a correlation reference, eg #1.COLNAME. (long).

Explanation: This message is issued by the FM/DB2 Basic SELECT Prototyping function. A value entered in the "Value" column matches a column name in a DB2 object selected for prototyping. There are two interpretations for the value entered:
1. A comparison between the column and a string constant.
2. A join between the column and the column name entered.

User response: Depending on context, either enclose the value in quotes, (to indicate the first interpretation shown above) or precede to the column name with a table correlation reference (example #n.) to indicate the second interpretation above.

FMNDB490 Parameters have been reset to defaults.

Explanation: This message is issued in response to the RESET command entered on an FM/DB2 options panel. The values on the panel have been reset to the default values.

User response: No action required. Informational message.

FMNDB491 RESET command failed.

Explanation: This message is issued in response to the RESET command, entered on an FM/DB2 options panel. The RESET operation failed, this is a "should not occur" error.

User response: Exit FM/DB2 completely. Start FM/DB2 again, and retry the operation. If the problem recurs, check for ISPF-related problems such as a full user ISPF profile data set. If there are no obvious external problems, contact IBM support.

FMNDB500 Copying...

Explanation: This message is issued by the FM/DB2 copy utility to indicate that copy operations are in progress.

User response: No action required. Informational message.

FMNDB501 number of rows rows

Explanation: This message is issued by the FM/DB2 copy utility to give an indication of progress for the current copy operation. number of rows is the number of rows that have been processed.

User response: No action required. Informational message.

FMNDB502 number of rows(s) copied (short). Number of rows row(s) copied from DB2 object source DB2 object to target DB2 object

Explanation: This message is issued by the FM/DB2 copy utility, at the completion of the copy operation, to report the number of rows copied from source DB2 object to target DB2 object. This message indicates that there were no duplicate key or referential integrity errors encountered.

User response: No action required. Informational message.

FMNDB503 number of rows row(s) copied. insert string

Explanation: This message is issued by the FM/DB2 copy utility, at the completion of the copy operation, to report the number of rows copied. insert string contains additional information, typically messages FMNDB504, FMNDB505, or FMNDB507.

User response: No action required. Informational message.

FMNDB504 count duplicate row(s) ignored.

Explanation: This message is issued by the FM/DB2 copy utility, at the completion of the copy operation, to report the number of duplicate row errors that were ignored. This message may be issued as part of message FMNDB503.

User response: No action required. Informational message.

FMNDB505 count RI/Constraint errors.

Explanation: This message is issued by the FM/DB2 copy utility, at the completion of the copy operation, to report the number of referential integrity, or constraint errors that were encountered. This message may be issued as part of message FMNDB503.

User response: No action required. Informational message.

FMNDB506 Copying

Explanation: This is a message insert used to construct a larger message.

User response: No action required.

FMNDB507 count duplicate row(s) updated.

Explanation: This message is issued by the FM/DB2 copy utility, at the completion of the copy operation, to report the number of duplicate rows that were updated. This message may be issued as part of message FMNDB503.

User response: No action required. Informational message.

FMNDB508 ->

Explanation: This is a message insert used to construct a larger message.

User response: No action required.
**FMNDB509 • FMNDB533**

**FMNDB509**  
*message*  
Explanation: This is a message insert used by the FM/DB2 copy utility. It contains a larger message constructed for other messages.  
User response: No action required.

**FMNDB510**  
Function canceled (short). Copy function was canceled by user. (long).  
Explanation: The user terminated the FM/DB2 copy function by issuing the CANCEL command, typically by pressing the PF12 PFKey (assigned to CANCEL by default).  
User response: No action required.

**FMNDB511**  
Rollback completed (short) Copy of source DB2 object to target DB2 object was rolled back due to errors.  
Explanation: This message is issued by the FM/DB2 copy utility when the current copy operation was terminated because of errors. A DB2 ROLLBACK is issued automatically in this situation, and all copy operations are backed out.  
User response: Determine why the copy operation was terminated, resolve the error condition, and retry the operation.

**FMNDB512**  
Copy canceled (short) Output object target DB2 object cannot be same as input object source DB2 object.  
Explanation: The same DB2 object name was specified on the “from” and “to” panels of the FM/DB2 copy utility. Copy from an object to itself is not supported. The Copy operation was canceled.  
User response: Specify different source or target DB2 objects, and retry the operation.

**FMNDB519**  
Audit failure (short) Auditing to SMF failed - function terminated. BPX1SMF RC=return code (Dec) REASON CODE=reason code (Dec)  
Explanation: FM/DB2 is set-up to write audit records to SMF for the current function. A failure occurred while attempting to write an audit record to SMF. The BPX1SMF call resulted in return code *return code* and reason code *reason code*, both expressed in decimal. The current FM/DB2 function is terminated.  
User response: This error usually indicates a set-up problem. Possible reason for an auditing failure to SMF are:  
1. The userid does have READ access to the SAF FACILITY class profile BPX.SMF.

**FMNDB520**  
SQL statement too long (short) The SQL statement is > 32K-2 in length. The maximum length for an SQL statement for DB2 versions 6 or 7 is 32765 characters. The statement cannot be executed. (long)  
Explanation: FM/DB2 detected an SQL statement greater than the 32K-2 limit, which is applicable to DB2 versions prior to DB2 version 8. The SQL statement cannot be executed.  
User response: Simplify the SQL statement, and retry the operation.

**FMNDB521**  
SQL statement too long (short) The SQL statement is > 2M in length. The maximum length for an SQL statement for DB2 versions 8 or later is 2097152 characters. The statement cannot be executed.  
Explanation: FM/DB2 detected an SQL statement greater than the 2M (2097152) limit, which is applicable to DB2 versions 8, and later. The SQL statement cannot be executed.  
User response: Simplify the SQL statement, and retry the operation.

**FMNDB530**  
ALIAS  
Explanation: This is a message insert used by the FM/DB2 audit report.  
User response: No action required.

**FMNDB531**  
AUX. TABLE  
Explanation: This is a message insert used by the FM/DB2 audit report.  
User response: No action required.

**FMNDB532**  
DATABASE  
Explanation: This is a message insert used by the FM/DB2 audit report.  
User response: No action required.
<table>
<thead>
<tr>
<th>Message ID</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FMNDB534</td>
<td>GBL. TMP. TABLE</td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a message insert used by the FM/DB2 audit report.</td>
</tr>
<tr>
<td>User response:</td>
<td>No action required.</td>
</tr>
<tr>
<td>FMNDB535</td>
<td>INDEX</td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a message insert used by the FM/DB2 audit report.</td>
</tr>
<tr>
<td>User response:</td>
<td>No action required.</td>
</tr>
<tr>
<td>FMNDB536</td>
<td>PROCEDURE</td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a message insert used by the FM/DB2 audit report.</td>
</tr>
<tr>
<td>User response:</td>
<td>No action required.</td>
</tr>
<tr>
<td>FMNDB537</td>
<td>ROLE</td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a message insert used by the FM/DB2 audit report.</td>
</tr>
<tr>
<td>User response:</td>
<td>No action required.</td>
</tr>
<tr>
<td>FMNDB538</td>
<td>SEQUENCE</td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a message insert used by the FM/DB2 audit report.</td>
</tr>
<tr>
<td>User response:</td>
<td>No action required.</td>
</tr>
<tr>
<td>FMNDB539</td>
<td>STOGROUP</td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a message insert used by the FM/DB2 audit report.</td>
</tr>
<tr>
<td>User response:</td>
<td>No action required.</td>
</tr>
<tr>
<td>FMNDB540</td>
<td>SYNONYM</td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a message insert used by the FM/DB2 audit report.</td>
</tr>
<tr>
<td>User response:</td>
<td>No action required.</td>
</tr>
<tr>
<td>FMNDB541</td>
<td>TABLE</td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a message insert used by the FM/DB2 audit report.</td>
</tr>
<tr>
<td>User response:</td>
<td>No action required.</td>
</tr>
<tr>
<td>FMNDB542</td>
<td>TABLESPACE</td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a message insert used by the FM/DB2 audit report.</td>
</tr>
<tr>
<td>User response:</td>
<td>No action required.</td>
</tr>
<tr>
<td>FMNDB543</td>
<td>TRIGGER</td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a message insert used by the FM/DB2 audit report.</td>
</tr>
<tr>
<td>User response:</td>
<td>No action required.</td>
</tr>
<tr>
<td>FMNDB544</td>
<td>TRUSTED CONTEXT</td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a message insert used by the FM/DB2 audit report.</td>
</tr>
<tr>
<td>User response:</td>
<td>No action required.</td>
</tr>
<tr>
<td>FMNDB545</td>
<td>TYPE</td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a message insert used by the FM/DB2 audit report.</td>
</tr>
<tr>
<td>User response:</td>
<td>No action required.</td>
</tr>
<tr>
<td>FMNDB546</td>
<td>VIEW</td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a message insert used by the FM/DB2 audit report.</td>
</tr>
<tr>
<td>User response:</td>
<td>No action required.</td>
</tr>
<tr>
<td>FMNDB550</td>
<td>COLLECTION</td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a message insert used by the FM/DB2 audit report.</td>
</tr>
<tr>
<td>User response:</td>
<td>No action required.</td>
</tr>
<tr>
<td>FMNDB551</td>
<td>JAR</td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a message insert used by the FM/DB2 audit report.</td>
</tr>
<tr>
<td>User response:</td>
<td>No action required.</td>
</tr>
<tr>
<td>FMNDB552</td>
<td>PACKAGE</td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a message insert used by the FM/DB2 audit report.</td>
</tr>
<tr>
<td>User response:</td>
<td>No action required.</td>
</tr>
<tr>
<td>FMNDB553</td>
<td>PLAN</td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a message insert used by the FM/DB2 audit report.</td>
</tr>
<tr>
<td>User response:</td>
<td>No action required.</td>
</tr>
<tr>
<td>FMNDB554</td>
<td>SCHEMA</td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a message insert used by the FM/DB2 audit report.</td>
</tr>
<tr>
<td>User response:</td>
<td>No action required.</td>
</tr>
</tbody>
</table>
**FMNDB560 • FMNDB658**

**FMNDB560  SYSTEM**

Explanation: This is a message insert used by the FM/DB2 audit report.

User response: No action required.

---

**FMNDB600  Global Settings**

Explanation: This message appears on some FM/DB2 options panels to indicate that the global options are being processed. FM/DB2 offers two levels of option settings for most functions. These are:

1. Global options, selected using option 0 from the FM/DB2 main menu. These options are saved in the ISPF profile data set for the user in the ISPF PROFILE pool, and persist between each FM/DB2 session.
2. Local options, selected from the function entry panel ("Edit options"). These options persist until the user exits the current FM/DB2 function, and overrides any Global options that may be in effect. See the File Manager User’s Guide for DB2 data for a full explanation.

User response: No action required. Informational message.

---

**FMNDB601  Changes discarded (short)**

Invalid changes to the default high level qualifier have been discarded. (long)

Explanation: An invalid value for the "default high level qualifier" field was specified, followed by an attempt to leave the panel. The invalid value has been discarded.

User response: Examine the value for "default high level qualifier" and re-enter the desired value. Press enter to validate the value, prior to leaving the panel.

---

**FMNDB602  Changes discarded (short)**

Invalid changes to the display width values have been discarded. (long).

Explanation: An invalid value for one of the "Display width for varying columns" fields was entered on the "Object List Options (2 of 2)" panel, followed by an attempt to leave the panel. The invalid value (or values) has been discarded.

User response: Use the field level help for each field (place the cursor on the field, press the HELP key, PF1 by default) to determine the valid values for each field. After entering a new value, press ENTER to validate the values, prior to leaving the panel.

---

**FMNDB650  Table Edit.**

Explanation: This is a message insert used by the FM/DB2 editor as a panel title.

User response: No action required.

---

**FMNDB651  Table Browse.**

Explanation: This is a message insert used by the FM/DB2 editor as a panel title.

User response: No action required.

---

**FMNDB652  Select Statement Edit.**

Explanation: This is a message insert used by the FM/DB2 editor as a panel title.

User response: No action required.

---

**FMNDB653  Select Statement Browse.**

Explanation: This is a message insert used by the FM/DB2 editor as a panel title.

User response: No action required.

---

**FMNDB654  Table Edit (Related).**

Explanation: This is a message insert used by the FM/DB2 editor as a panel title.

User response: No action required.

---

**FMNDB655  Table View.**

Explanation: This is a message insert used by the FM/DB2 editor as a panel title.

User response: No action required.

---

**FMNDB656  Select Statement View.**

Explanation: This is a message insert used by the FM/DB2 editor as a panel title.

User response: No action required.

---

**FMNDB657  Table Browse (LOB).**

Explanation: This is a message insert used by the FM/DB2 editor as a panel title.

User response: No action required.

---

**FMNDB658  Table Edit (LOB).**

Explanation: This is a message insert used by the FM/DB2 editor as a panel title.

User response: No action required.
FMNDB659 Table View (LOB).
Explanation: This is a message insert used by the FM/DB2 editor as a panel title.
User response: No action required.

FMNDB660 DB2 Browse.
Explanation: This is a message insert used by the FM/DB2 editor as a panel title.
User response: No action required.

FMNDB661 DB2 Edit.
Explanation: This is a message insert used by the FM/DB2 editor as a panel title.
User response: No action required.

FMNDB662 Advanced SELECT Prototyping.
Explanation: This is a message insert used as a panel title.
User response: No action required.

FMNDB663 Create Table: Nulls and Default Values.
Explanation: This is a message insert used as a panel title.
User response: No action required.

FMNDB664 Create Table: Nulls
Explanation: This is a message insert used as a panel title.
User response: No action required.

FMNDB665 DB2 View.
Explanation: This is a message insert used by the FM/DB2 editor as a panel title.
User response: No action required.

FMNDB670 SQL Statement.
Explanation: This is a message insert used in FM/DB2 panel text.
User response: No action required.

FMNDB671 Record Number.
Explanation: This is a message insert used in FM/DB2 panel text.
User response: No action required.

FMNDB672 Press.
Explanation: This is a message insert used to construct a larger message.
User response: No action required.

FMNDB673 PF3 (End) or Enter.
Explanation: This is a message insert used to construct a larger message.
User response: No action required.

FMNDB674 PF12 (Cancel).
Explanation: This is a message insert used to construct a larger message.
User response: No action required.

FMNDB675 to terminate the import.
Explanation: This is a message insert used in the FM/DB2 import utility to create a larger message.
User response: No action required.

FMNDB700 Audit Trail Report.
Explanation: This is a message insert used in the FM/DB2 audit report utility.
User response: No action required.

FMNDB701 Formatted Audit Event Records.
Explanation: This is a message insert used in the FM/DB2 audit report utility.
User response: No action required.

FMNDB702 Date: date.
Explanation: This is a message insert used in the FM/DB2 audit report utility.
User response: No action required.

FMNDB703 Time: time.
Explanation: This is a message insert used in the FM/DB2 audit report utility.
User response: No action required.

FMNDB707 Summary Statistics.
Explanation: This is a message insert used in the FM/DB2 audit report utility.
User response: No action required.
FMNDB708 • FMNDB735

FMNDB708  DB2 access commenced on date at time by user ID TSO userid.
Explanation: This is a message insert used in the FM/DB2 audit report utility. It shows the start date date, and time time, when the FM/DB2 function being audited first accessed DB2, and the TSO user ID TSO userid associated with the transaction.
User response: No action required.

FMNDB709  DB2 access completed on date at time.
Explanation: This is a message insert used in the FM/DB2 audit report utility. It shows the completion date date, and time time, for the FM/DB2 function being audited.
User response: No action required.

FMNDB710  Audit trail for function function.
Explanation: This is a message insert used in the FM/DB2 audit report utility. It shows the FM/DB2 function function for which the audit report applies.
User response: No action required.

FMNDB711  Audit description: description.
Explanation: This is a message insert used in the FM/DB2 audit report utility. description is a user-specified value entered on the "Print Audit Trail" panel.
User response: No action required.

FMNDB712  Data set name: data set name.
Explanation: This is a message insert used in the FM/DB2 audit report utility. data set name is the data set name used to produce the audit report, and is the value entered on the "Print Audit Trail" panel.
User response: No action required.

FMNDB720  System Information:
Explanation: This is a message insert used in the FM/DB2 audit report utility.
User response: No action required.

FMNDB721  MVS System Id: MVS ID.
Explanation: This is a message insert used in the FM/DB2 audit report utility. MVS ID shows the z/OS system ID on which the FM/DB2 function was executed.
User response: No action required.

FMNDB722  DB2 Subsystem: DB2 SSID.
Explanation: This is a message insert used in the FM/DB2 audit report utility. DB2 SSID is the DB2 subsystem or group identifier to which FM/DB2 was connected when the function being audited was executed.
User response: No action required.

FMNDB723  Current SQLID: SQLID.
Explanation: This is a message insert used in the FM/DB2 audit report utility. SQLID shows the DB2 authid associated with the connection to DB2 when the FM/DB2 function was executed.
User response: No action required.

FMNDB730  DB2 Object Information:
Explanation: This is a message insert used in the FM/DB2 audit report utility.
User response: No action required.

FMNDB731  Source DB2 Object Information:
Explanation: This is a message insert used in the FM/DB2 audit report utility.
User response: No action required.

FMNDB732  Target DB2 Object Information:
Explanation: This is a message insert used in the FM/DB2 audit report utility.
User response: No action required.

FMNDB733  Location: location.
Explanation: This is a message insert used in the FM/DB2 audit report utility. location is the remote server ID (which may be blank) that identifies the DB2 object processed by the FM/DB2 function.
User response: No action required.

FMNDB734  DB2Owner: owner.
Explanation: This is a message insert used in the FM/DB2 audit report utility. owner is the owner of the DB2 object processed by the FM/DB2 function.
User response: No action required.

FMNDB735  Name: Name.
Explanation: This is a message insert used in the FM/DB2 audit report utility. name is the name of the DB2 object processed by the FM/DB2 function.
User response: No action required.
FMNDB736  Database: database.
Explanation: This is a message insert used in the FM/DB2 audit report utility. database is the database name, which may be blank, associated with the DB2 object processed by the FM/DB2 function.
User response: No action required.

FMNDB737  Table Space: table space.
Explanation: This is a message insert used in the FM/DB2 audit report utility. table space is the table space name, which may be blank, associated with the DB2 object processed by the FM/DB2 function.
User response: No action required.

FMNDB738  Result Table Information:
Explanation: This is a message insert used in the FM/DB2 audit report utility.
User response: No action required.

FMNDB739  SELECT statement:
Explanation: This is a message insert used in the FM/DB2 audit report utility.
User response: No action required.

FMNDB740  Template Information:
Explanation: This is a message insert used in the FM/DB2 audit report utility.
User response: No action required.

FMNDB741  Source Template Information:
Explanation: This is a message insert used in the FM/DB2 audit report utility.
User response: No action required.

FMNDB742  Target Template Information:
Explanation: This is a message insert used in the FM/DB2 audit report utility.
User response: No action required.

FMNDB750  Audit event number: number
Explanation: This is a message insert used in the FM/DB2 audit report utility. number is a positive integer. Each audit event is assigned a consecutive number starting at 1, the numbers increment.
User response: No action required.

FMNDB751  Data action on date at time.
Explanation: This is a message insert used in the FM/DB2 audit report utility. action is the type of operation. For example INSERT, DELETE, and UPDATE. date and time show the date and time when the SQL statement was executed.
User response: No action required.

FMNDB752  All data action on date at time.
Explanation: This is a message insert used in the FM/DB2 audit report utility. action is the type of operation. For example FETCH, or CHANGE. date, and time show the date and time when the SQL statement was executed.
User response: No action required.

FMNDB753  COMMIT issued on date at time, SQLCODE=SQLCODE.
Explanation: This is a message insert used in the FM/DB2 audit report utility. A DB2 COMMIT was issued on date date at time time. The SQLCODE associated with the COMMIT was SQLCODE.
User response: No action required.

FMNDB754  ROLLBACK issued on date at time, SQLCODE=SQLCODE.
Explanation: This is a message insert used in the FM/DB2 audit report utility. A DB2 ROLLBACK was issued on date date at time time. The SQLCODE associated with the ROLLBACK was SQLCODE.
User response: No action required.

FMNDB755  All pending changes committed.
Explanation: This is a message insert used in the FM/DB2 audit report utility.
User response: No action required.

FMNDB756  All pending changes backed out. ***
Explanation: This is a message insert used in the FM/DB2 audit report utility.
User response: No action required.

FMNDB757  SQLCODE=SQLCODE, row count row(s) affected.
Explanation: This is a message insert used in the FM/DB2 audit report utility. SQLCODE is the SQLCODE associated with an SQL statement, which might have affected 1 or more rows. row count shows the number of rows, as reported by DB2, that were affected by the SQL statement. For SQL statements that
affect DB2 objects subject to referential integrity constraints, the row count value does not show rows affected in objects other than the primary object affected by the SQL statement.

User response: No action required.

FMNDB758  DB2 command issued on date at time.
Explanation: This is a message insert used in the FM/DB2 audit report utility. It shows the date date and time time that a DB2 command was issued.
User response: No action required.

FMNDB759  SQLCODE=SQLCODE, mass delete.
Explanation: This is a message insert used in the FM/DB2 audit report utility. SQLCODE shows the SQLCODE for a "mass delete" SQL statement, such as "DELETE * FROM some DB2 object."
User response: No action required.

FMNDB760  SQL statement:
Explanation: This is a message insert used in the FM/DB2 audit report utility.
User response: No action required.

FMNDB761  Column Name Key Type Original Data
Explanation: This is a message insert used in the FM/DB2 audit report utility.
User response: No action required.

FMNDB762  DB2 command:
Explanation: This is a message insert used in the FM/DB2 audit report utility.
User response: No action required.

FMNDB763  *** excludes rows deleted via mass delete ***
Explanation: This is a message insert used in the FM/DB2 audit report utility.
User response: No action required.

FMNDB764  DDL statements: count
Explanation: This is a message insert used in the FM/DB2 audit report utility. count shows the number of DDL SQL statements issued by the FM/DB2 function.
User response: No action required.
FMNDB774  GRANT issued on date at time.
Explanation:  This is a message insert used in the FM/DB2 audit report utility. A GRANT statement was issued on date at time.
User response:  No action required.

FMNDB775  REVOKE issued on date at time
Explanation:  This is a message insert used in the FM/DB2 audit report utility. A REVOKE statement was issued on date at time.
User response:  No action required.

FMNDB776  SQLCODE=+100, no matching rows. **
Explanation:  This is a message insert used in the FM/DB2 audit report utility.
User response:  No action required.

FMNDB777  SQLCODE=+100, no rows fetched. **
Explanation:  This is a message insert used in the FM/DB2 audit report utility.
User response:  No action required.

FMNDB778  SQLCODE=SQLCODE, successful execution.
Explanation:  This is a message insert used in the FM/DB2 audit report utility. An SQL statement was processed, the SQLCODE was SQLCODE, the statement was successfully executed.
User response:  No action required.

FMNDB779  SQLCODE=SQLCODE, statement failed. ***
Explanation:  This is a message insert used in the FM/DB2 audit report utility. An SQL statement was processed, the SQLCODE was SQLCODE. The statement was not successfully executed.
User response:  No action required.

FMNDB780  Total SQL/DB2 statements: count
Explanation:  This is a message insert used in the FM/DB2 audit report utility. count shows the total of DB2 and SQL statements executed by the FM/DB2 function.
User response:  No action required.

FMNDB781  Rows inserted: count
Explanation:  This is a message insert used in the FM/DB2 audit report utility. count shows how many rows were inserted by the FM/DB2 function.
User response:  No action required.

FMNDB782  Rows deleted: count
Explanation:  This is a message insert used in the FM/DB2 audit report utility. count shows how many rows were deleted by the FM/DB2 function.
User response:  No action required.

FMNDB783  Rows updated: count
Explanation:  This is a message insert used in the FM/DB2 audit report utility. count shows how many rows were updated by the FM/DB2 function.
User response:  No action required.

FMNDB784  TOTAL Data base changes: count
Explanation:  This is a message insert used in the FM/DB2 audit report utility. count shows the total number of changes made by the FM/DB2 function.
User response:  No action required.

FMNDB785  Records bypassed: count
Explanation:  This is a message insert used in the FM/DB2 audit report utility. count shows the number of records bypassed, as a result of record selection.
User response:  No action required.

FMNDB786  Row count (in): count
Explanation:  This is a message insert used in the FM/DB2 audit report utility. count shows the number of rows read by the FM/DB2 function.
User response:  No action required.

FMNDB787  Row count editor
Explanation:  This is a message insert used in the FM/DB2 audit report utility. count shows the number of rows read by the FM/DB2 editor.
User response:  No action required.

FMNDB788  Row count (out): count
Explanation:  This is a message insert used in the FM/DB2 audit report utility. count shows the number of rows written by the FM/DB2 function.
User response:  No action required.
FMNDB789 • FMNDB805

FMNDB789  Rollbacks reported: count
Explanation:  This is a message insert used in the FM/DB2 audit report utility. count shows the number of ROLLBACK statements issued by the FM/DB2 function.
User response:  No action required.

FMNDB790  Audit Trail Report Options
Explanation:  This is a message insert used in the FM/DB2 audit report utility.
User response:  No action required.

FMNDB791  Maximum Print Lines per Page: line count
Explanation:  This is a message insert used in the FM/DB2 audit report utility. line count is the maximum number of print lines per page of the audit report.
User response:  No action required.

FMNDB792  Print Only Changed Fields on an Update: YES|NO
Explanation:  This is a message insert used in the FM/DB2 audit report utility. YES indicates that only those fields that were changed by an UPDATE operation are shown in the audit report.
User response:  No action required.

FMNDB800  count rows exported (short) count rows exported, from file input file name to DB2 object target DB2 object. duplicate count duplicate rows not imported. error count rows in error not imported. selection count rows dropped via record selection. REXX drop count rows dropped by REXX proc. (long).
Explanation:  This message is issued at the end of an FM/DB2 export function, and shows summary statistics for the import operation.
- count is the number of rows imported.
- duplicate count is the number of duplicate row errors that were encountered. The record that caused the duplicate row error has not been imported.
- error count is the number of records that generated errors other than a duplicate row error. The record that caused the error has not been imported.
- selection count is the number of records (rows) that were dropped as a result of record selection.
- REXX drop count is the number of records (rows) that were dropped because of user-specified processing in the REXX procedure. The same information is presented on the "Import Report" panel. This message is issued when the Update option for duplicate key errors is not selected on the import options panel.
User response:  No action required.

FMNDB801  Export canceled (short) Export function was canceled by the user. (long).
Explanation:  This message is issued by the FM/DB2 export utility. The current FM/DB2 export function is canceled and the FM/DB2 export function entry panel is redisplayed.
User response:  No action required. Informational message.

FMNDB802  Exporting...
Explanation:  This message is issued by the FM/DB2 export utility to indicate that export operations are in progress.
User response:  No action required. Informational message.

FMNDB803  Import canceled (short) Import function was canceled by the user. (long).
Explanation:  The user canceled the import operation, by pressing the END or CANCEL PFkey.
User response:  No action required. Informational message.

FMNDB804  Importing...
Explanation:  This message is issued by the FM/DB2 import utility to indicate that import operations are in progress.
User response:  No action required. Informational message.
**FMNDB806** Rollback completed (short) Import of import file name to target DB2 object was rolled back due to errors. count rows rolled back, duplicate count duplicate rows not imported, updated count duplicate rows updated, not updated count duplicate rows not updated, error count rows in error not imported, drop count rows dropped by record selection. REXX drop count rows dropped by REXX procedure. Note that PF12/CANCEL on SQL error panel will skip each row in error.

**Explanation:** This is a message insert, used in the FM/DB2 audit report utility.

**User response:** No action required.

---

**FMNDB807** row count rows created (short) row count rows created, in DB2 object output DB2 object name, duplicate count duplicate rows not created, error count in error not created.

**Explanation:** This message is issued by the FM/DB2 data create utility. row count is the number of rows that were inserted (created) in the output DB2 object output DB2 object name.

- duplicate count is the number of rows that could not be inserted because of duplicate key (SQLCODE-803) errors.
- error count is the number of rows that could not be inserted because of errors other than duplicate key errors.

**User response:** No action required. Informational message. If there are multiple duplicate key errors, you can edit the template for those columns that form part of the unique index keys to specify a pattern to be used when creating values for that column.

---

**FMNDB808** Create canceled (short) Create function was canceled by the user.

**Explanation:** The user canceled the FM/DB2 data create function, typically be pressing the PF key assigned to CANCEL.

**User response:** No action required. Informational message.

---

**FMNDB809** Creating...

**Explanation:** This message is issued by the FM/DB2 data create utility to indicate that data create operations are in progress.

**User response:** No action required. Informational message.
The FM/DB2 import utility, executing in batch mode, did not import row count rows because the corresponding input record was dropped as part of the associated REXX procedure processing.

**User response:** This is one of several messages that may be issued by the FM/DB2 import utility when run in batch mode. The user response depends on context. It may be reasonable for records to be dropped. If, however, records should not be dropped as part of the REXX procedure processing, examine the input data to determine why the records were dropped.

The FM/DB2 data create utility, executing in batch mode, did not create row count rows because DB2 raised a duplicate key error (SQLCODE-803) when FM/DB2 attempted to insert the newly created row in the target DB2 object.

**User response:** This is one of several messages that may be issued by the FM/DB2 data create utility when run in batch mode. The user response depends on context. DB2 will always reject rows that violate uniqueness constraints. When many duplicate row errors are received consider adjusting the input parameters to avoid duplicate values for any index columns.

The FM/DB2 data create utility, executing in batch mode, did not create row count rows because DB2 raised an error (other than SQLCODE-803) when FM/DB2 attempted to insert the newly created row in the target DB2 object.

**User response:** This is one of several messages that may be issued by the FM/DB2 data create utility when run in batch mode. This message usually indicates an error. DB2 may reject the insert of a row for many reasons. Examples of error SQLCODEs that might result in this message include -530, -531, and -532 (referential integrity errors).

The FM/DB2 import utility, executing in batch mode, did not import row count rows. Only columns that are not part of any unique index key are updated.

**User response:** This is one of several messages that may be issued by the FM/DB2 import utility, when run in batch mode. The user response depends on context. In most cases, the update of existing rows is best. If, however, the import operation is not expected to cause duplicate key errors (SQLCODE-803), compare the input records and DB2 object, to discover why there are duplicates.
FMNDB826  All columns selected (short)  No columns were selected in the template for DB2 object name. Processing cannot proceed without at least one selected column. FM/DB2 has automatically selected all columns for processing.

Explanation:  This message is issued by either the FM/DB2 export, or FM/DB2 UNLOAD utilities. The template editor was used to deselect all columns, followed by an attempt to execute the function with the modified template. Since an FM/DB2 function requires at least one selected column to proceed, all columns are automatically selected and this message is issued.

User response:  No action required. Informational message. By default, all columns of a DB2 object are selected for processing. While it is valid to deselect some of the columns in the DB2 object, it is invalid to deselect all columns, since this results in a "no-operation" condition.

FMNDB827  Import to view failed (short)  The import operation has been canceled. Import to a view is only supported if the view is defined directly on a table. Repeat the operation using the actual table that the view refers to.

Explanation:  The FM/DB2 import utility was used to import data into a DB2 view. The DB2 view refers to more than one DB2 table, or refers to one or more views. Import to this type of view is not supported. The import operation is canceled.

User response:  In general, import to a DB2 view defined as "SELECT * FROM DB2 table name" is supported. Import to views that refer to multiple DB2 tables, or views, is not supported. Where possible, use the FM/DB2 import utility with a table name, rather than a dependent view.

FMNDB828  Remapping required (short)  No columns in the target table are mapped to any fields in the input template. This can occur if the column names in the target object are different to the field names in the input template. Edit the template for the target object to set the mapping between columns/fields in the target/input templates.

Explanation:  The FM/DB2 import utility was used with an input template that has field names that do not match any of the columns in the target DB2 object. The automatic field mapping process used by FM/DB2, cannot reliably determine which fields in the input template correspond with columns in the target DB2 object. Manual mapping of each field in the input template to a column in the target DB2 object is required.

User response:  This message may be issued in the situation where the wrong input file template is specified for an FM/DB2 import function. Advanced usage of the FM/DB2 import utility allows individual mapping of each column in the target DB2 object to a corresponding field of the input template. The column and field names need not match, nor the relative positions of the columns in the DB2 object and fields in the input record. If this is the intended usage, use the template editor ("edit template mapping") to set the relationship between columns and fields, and retry the operation.

FMNDB829  Invalid value (short)  The valid values are ALL or a number in the range 0-2147483647. (long).

Explanation:  An invalid value was entered in the "Max duplicates" field of the "Copy Options" panel. The valid values are "ALL" or an integer in the range 0-2147483647, inclusive.

User response:  Correct the invalid value.

FMNDB831  Browse substituted (short)  The DB2 result table is not updateable, specifically because not all columns of the first table named in the FROM clause are present, nor are all columns of any unique index(es) defined on this table. Browse has been substituted for edit. (long).

Explanation:  A SELECT statement was entered using one of the FM/DB2 functions that allow free-form SQL statements to be processed. When FM/DB2 uses the editor to display the result table from SELECT statement, edit operations may not be possible for free-form SQL statements. This message occurs when the conditions listed in the message are not met. For tables with unique indexes defined, FM/DB2 can use the values in the columns that comprise the unique index to uniquely identify any row of the table. However, this requires that every column in the unique index key is represented in the result table. If reliable edit is not possible, FM/DB2 converts the edit session to browse to avoid any possibility of data corruption.

User response:  No action required. Informational message. If edit operations of the result table are required, the preferred method is to edit the DB2 object directly.
**FMNDB832**  
Update not possible (short) FM/DB2 cannot update the data in the row(s) marked with =INERR. This error may occur attempting to edit the result table from an arbitrary select statement entered using option 4.2, 4.3 or 4.4. You should cancel the current edit session and then edit the underlying DB2 object directly.

**Explanation:** An attempt was made to update data in the row of a result table for an SQL statement entered using FM/DB2 options 4.2, 4.3, or 4.4. The update failed because DB2 rejected the update. There are various reasons why this error may occur, including editing the result table for a join. The update operation cannot be performed.

**User response:** If edit operations of the result table are required, the preferred method is to edit the DB2 object directly.

**FMNDB833**  
Delete not possible (short) FM/DB2 cannot delete the data in the row(s) marked with =INERR. This error may occur attempting to edit the result table from an arbitrary select statement entered using option 4.2, 4.3 or 4.4. You should cancel the current edit session and then edit the underlying DB2 object directly.

**Explanation:** An attempt was made to delete a row of a result table for an SQL statement entered using FM/DB2 options 4.2, 4.3, or 4.4. The update failed because DB2 rejected the delete. There are various reasons why this error may occur, including editing the result table for a join. The update operation cannot be performed.

**User response:** If edit operations of the result table are required, the preferred method is to edit the DB2 object directly.

**FMNDB835**  
Parameter marker invalid (short) The SELECT statement contains one or more "?". DB2 interprets these as parameter markers, resulting in an SQL error. Replace the "?" with constants and rerun the statement.

**Explanation:** A SELECT statement was entered using FM/DB2 options 4.3, or 4.4, and possibly other options. The SELECT statement includes DB2 parameter markers ('?'). While the use of DB2 parameter markers is valid when using, for example, an application programming language, the use of DB2 parameter markers in SELECT statements processed by FM/DB2 is invalid.

**User response:** Locate the DB2 parameter markers in the SELECT statement. Replace each occurrence with a constant of the appropriate type. Rerun the statement.

**FMNDB836**  
Browse substituted (short) This version of FM/DB2 does not support new DB2 data types introduced with DB2 version 9. Browse has been substituted for edit. (long).

**Explanation:** FM/DB2 detected a data type in the DB2 object being edited that is not currently supported. To avoid the possibility of data corruption, FM/DB2 has substituted browse for edit.

**User response:** Determine which column has the unsupported data type. Contact your IBM sales representative for information about later versions of FM/DB2 which support the new data type.

**FMNDB837**  
XML column(s) (short) This version of FM/DB2 does not support the XML data type introduced with DB2 version 9. The function cannot be performed.

**Explanation:** FM/DB2 detected one or more XML columns in the object being processed by the current FM/DB2 function (editor, copy, import, export, and print). XML columns are not supported. The function cannot be performed, and processing is terminated.

**User response:** Specify a DB2 object that does not include XML columns.

**FMNDB838**  
Duplicate count reached (short) The specified duplicate count was reached. The import has been canceled. (long).

**Explanation:** An FM/DB2 import was specified with a non-zero "Max duplicate" count. During the processing of the input data, the number of duplicate row errors reached the specified "Max duplicate" value. The import operation if canceled.

**User response:** The "Max duplicate" count value is specified on the Import Options panel, selected using the "edit options" field on the Import function entry panel. The response depends on the context of when duplicate row errors are expected. Rerun the import after increasing the "Max duplicate" value. When duplicate row errors are not expected, examine the input data to determine why duplicate row errors are occurring.

**FMNDB839**  
Unable to process (short) The DB2 object contains one or more LOB columns. There is insufficient memory available for FM/DB2 to process data from this object, because of the large size of the LOB column(s). Refer to the Edit Tutorial, General information, Restrictions for additional information on how to avoid this problem.
FMNDB840 • FMNDB850

**Explanation:** FM/DB2 does not support the processing of DB2 objects with LOB columns. The function cannot be performed and processing is terminated.

**User response:** Specify a DB2 object that does not include LOB columns.

---

**FMNDB840**  Format not for template (short) The Data Format is not compatible with that used to create the template. Template is template type.

**Explanation:** This message is issued by the FM/DB2 import utility when there is a mismatch between the template describing the input data, and the data format option for the import data set. The import cannot be processed, and the import is canceled. The type of template specified is template type, which will be one of the following: DB2 Unload; DB2 DSNTIAUL; User defined; Base; FM/DB2 SQLDA; Other. (long).

**User response:** The data format for the input data set is specified on the Import options panel, selected using the "edit options" field on the Import function entry panel. The value specified must match the actual data format of the input data. It must also match the type of template specified in the template, or copybook, specified on the import function entry panel. Verify that the specified template correctly describes the input data. Determine the data format of the input data, use that information to set the appropriate data format value on the Import options panel, and retry the import operation.

---

**FMNDB841**  DB2 unload.

**Explanation:** This is a message insert used in message FMNDB840.

**User response:** No action required.

---

**FMNDB842**  DB2 DSNTIAUL.

**Explanation:** This is a message insert used in message FMNDB840.

**User response:** No action required.

---

**FMNDB843**  User defined.

**Explanation:** This is a message insert used in message FMNDB840.

**User response:** No action required.

---

**FMNDB844**  BASE.

**Explanation:** This is a message insert used in message FMNDB840.

**User response:** No action required.

---

**FMNDB845**  FM/DB2 (SQLDA).

**Explanation:** This is a message insert used in message FMNDB840.

**User response:** No action required.

---

**FMNDB846**  OTHER.

**Explanation:** This is a message insert used in message FMNDB840.

**User response:** None

---

**FMNDB847**  Batch, using DB2 UNLOAD

**Explanation:** This is a message insert, displayed on the first export options panel.

**User response:** No action required.

---

**FMNDB848**  Edit DB2 UNLOAD options

**Explanation:** This is a message insert, displayed on the first export options panel.

**User response:** No action required.

---

**FMNDB849**  Incompatible options (short) The data format must be 2 (DB2 unload format) when execution option 3 (Batch, using DB2 UNLOAD) is selected. (long).

**Explanation:** The FM/DB2 export function can be used to generate a DB2 utility job that will run the DB2 UNLOAD utility for a DB2 table. When this option is selected, the format of the output data is determined by DB2. Therefore, the data format must be set to '2' (DB2 unload format). The data format for the output data set is specified on the first export options panel, selected using the "edit options" field on the Export function entry panel.

**User response:** Adjust the data format option as indicated, and retry the operation. Note, that it is also possible to use the FM/DB2 export function to export data in "DB2 unload format", without running a DB2 batch job. If this is what is intended, set the execution option on the first export options panel to either '1' (online) or '2' (batch), and retry the operation.

---

**FMNDB850**  Must be sequential file (short) The DB2 UNLOAD utility does not support unload of data to a partitioned data set. (long).

**Explanation:** The FM/DB2 export utility was executed with an execution option of '3' (Batch, using DB2 unload). The output data set specified on the export 'To' panel, is a partitioned data set includes a member name. While the FM/DB2 export utility can export data in DB2 unload format to a member of a partitioned
FMNDB851 • FMNDB859

User response: Specify a sequential file (Data set organization of PS), rather than a partitioned data set (Data set organization of PO), for the output data set. Retry the operation.

Explanation: The FM/DB2 export utility was executed with an execution option of '3' (Batch, using DB2 unload). The output data set specified on the export 'To' panel, is not a sequential data set. While the FM/DB2 export utility can export data in DB2 unload format to non-sequential data sets, the only data set type supported by the DB2 UNLOAD utility is sequential.

User response: Check the type for the output data set specified for the DB2 unload operation on the FM/DB2 export 'To' panel. Ensure that a sequential file (Data set organization of PS) is specified. Retry the operation.

Explanation: An SQL statement and query number value were entered on the "Enter, Execute and Explain SQL" panel (FM/DB2 option 4.3). A query number is only required when the EXPLAIN primary command is issued.

User response: Remove the query number and retry the operation. If the intention was to EXPLAIN the SQL statement, type 'EXPLAIN' on the command line, and press Enter.

Explanation: The user entered a value in the "Separator character" field on the second FM/DB2 Export options panel. This field is used to enter a one or two character value for the separator character in "delimited variables (CSV)" export format. The value entered contains either a single or double quote, which is not supported.

User response: The delimited variables (CSV, comma separated variables) format is loosely defined. In situations where the default separator character (comma) is inappropriate, FM/DB2 allows the specification of a one, or two, character alternative value. The separator value cannot include either a single, or double quote, because this causes syntactical ambiguity when the exported data also includes single or double quotes, either within the data, or inserted to encapsulate the data. Select another value.

Explanation: A value entered in the "start position" field on the editor function entry panel, or the "create count" field on the data create function entry panel, is invalid. A numeric value is required, which must be a positive integer in the range 1-2147483647, inclusive.

User response: Correct the value, retry the operation.

Explanation: A value entered in a field on an FM/DB2 function entry panel is invalid. A numeric value is required, which must be a non-negative integer in the range 0-2147483647, inclusive. This message is
issued for various "row count" input fields. A value of * and ALL can also be entered as synonyms for 0. These values indicate that there is no limit to the number of rows or records to be processed.

**User response:** Correct the value, retry the operation.

**FMNDB860** Audit trail for FM/DB2 browse function.

**Explanation:** This is a message issued by the FM/DB2 audit report, indicating the audit report relates to the specified function.

**User response:** No action required.

**FMNDB861** Audit trail for FM/DB2 view function.

**Explanation:** This is a message issued by the FM/DB2 audit report, indicating the audit report relates to the specified function.

**User response:** No action required.

**FMNDB862** Audit trail for FM/DB2 edit function.

**Explanation:** This is a message issued by the FM/DB2 audit report, indicating the audit report relates to the specified function.

**User response:** No action required.

**FMNDB863** Audit trail for FM/DB2 copy utility.

**Explanation:** This is a message issued by the FM/DB2 audit report, indicating the audit report relates to the specified function.

**User response:** No action required.

**FMNDB864** Audit trail for FM/DB2 import utility

**Explanation:** This is a message issued by the FM/DB2 audit report, indicating the audit report relates to the specified function.

**User response:** No action required.

**FMNDB865** Audit trail for FM/DB2 export utility.

**Explanation:** This is a message issued by the FM/DB2 audit report, indicating the audit report relates to the specified function.

**User response:** No action required.

**FMNDB866** Audit trail for FM/DB2 data create utility.

**Explanation:** This is a message issued by the FM/DB2 audit report, indicating the audit report relates to the specified function.

**User response:** No action required.

**FMNDB867** Audit trail for FM/DB2 DB2 commands utility.

**Explanation:** This is a message issued by the FM/DB2 audit report, indicating the audit report relates to the specified function.

**User response:** No action required.

**FMNDB868** Audit trail for FM/DB2 DB2 objects utility.

**Explanation:** This is a message issued by the FM/DB2 audit report, indicating the audit report relates to the specified function.

**User response:** No action required.

**FMNDB869** Audit trail for FM/DB2 DB2 privileges utility.

**Explanation:** This is a message issued by the FM/DB2 audit report, indicating the audit report relates to the specified function.

**User response:** No action required.

**FMNDB870** Audit trail for FM/DB2 "Enter, Execute and Explain SQL" utility.

**Explanation:** This is a message issued by the FM/DB2 audit report, indicating the audit report relates to the specified function.

**User response:** No action required.

**FMNDB871** Audit trail for FM/DB2 "Edit/Execute SQL (Data Set)" utility.

**Explanation:** This is a message issued by the FM/DB2 audit report, indicating the audit report relates to the specified function.

**User response:** No action required.

**FMNDB872** Audit trail for FM/DB2 object list utility.

**Explanation:** This is a message issued by the FM/DB2 audit report, indicating the audit report relates to the specified function.

**User response:** No action required.

**FMNDB873** Audit trail for FM/DB2 print utility.

**Explanation:** This is a message issued by the FM/DB2 audit report, indicating the audit report relates to the specified function.

**User response:** No action required.
FMNDB874  •  FMNDB907

**FMNDB874** Audit trail for FM/DB2 basic SQL prototyper.

**Explanation:** This is a message issued by the FM/DB2 audit report, indicating the audit report relates to the specified function.

**User response:** No action required.

**FMNDB875** Audit trail for FM/DB2 advanced SQL prototyper.

**Explanation:** This is a message issued by the FM/DB2 audit report, indicating the audit report relates to the specified function.

**User response:** No action required.

**FMNDB879** Audit trail for File Manager DB2 component.

**Explanation:** This is a message issued by the FM/DB2 audit report, indicating the audit report pertains to FM/DB2 component.

**User response:** No action required.

**FMNDB890** DB2 subsystem DB2 SSID not defined to z/OS?.

**Explanation:** This message is issued by an FM/DB2 batch job, when connection to DB2 system DB2 SSID failed. One reason for this message being issued is that D2 SSID does not exist (has not been defined to the local z/OS system). This might occur in sites running multiple LPARs with shared DASD, where DB2 systems are available on some, but not other, z/OS systems.

**User response:** Check that the DB2 SSID specified as a parameter is defined to the z/OS system where the batch job was executed. If this is not the problem, look up DB2 error code '00F30006' in the DB2 Codes manual and proceed as directed for that error.

**FMNDB903** Identifier too long (short) The identifier is too long. See the SQL Reference for information about the maximum lengths for identifiers. DB2 returned an SQL code of -107 for SQL statement SQL statement. (long).

**Explanation:** This message is issued when an SQL statement contains an identifier that is too long. When connected to DB2 V8 systems or later, this error is unlikely, given the large maximum lengths for most DB2 identifiers, compared to the corresponding DB2 V7 values.

**User response:** Respond as for SQLCODE-107.

**FMNDB904** Not authorized (short) Your authorization ID does not have the authority to change the privileges for this object. DB2 returned an SQL code of -551 for SQL statement SL statement. (long).

**Explanation:** This message is issued by the FM/DB2 Object Privileges utility. The DB2 authorization IDs for the current user do not have the authority to change (GRANT or REVOKE) for the object.

**User response:** Respond as for SQLCODE-551. See your DB2 System Administrator.

**FMNDB905** Object not found (short) The specified DB2 object does not exist. DB2 returned an SQL code of -204 for the SQL statement SL statement. (long).

**Explanation:** This message is issued by the FM/DB2 Object Privileges utility. The DB2 object for which the privileges are to be changed could not be found (SQLCODE-204).

**User response:** Respond as for SQLCODE-204. Check the DB2 object name.

**FMNDB907** Cannot be revoked (short) You cannot revoke the specified privilege because the revokee does not possess the privilege or because the specified DB2 authid did not grant the privilege. DB2 returned an SQL code of -556 for SQL statement SL Statement. (long).

**Explanation:** This message is issued by the FM/DB2 Object Privileges utility. The privilege cannot be revoked.

**User response:** Respond as for SQLCODE-556. Refer to the SQL Codes manual, for the appropriate version of DB2, for a detailed list of possible causes of this error.
FMNDB909  Not authorized (short) Your authorization ID does not have the privilege to issue a REVOKE statement with the BY clause. DB2 returned an SQL code of -552 for SQL statement SL statement.long

Explanation:  This message is issued by the FM/DB2 Object Privileges utility. The privilege cannot be revoked.

User response:  Respond as for SQLCODE-552.

FMNDB910  Privilege conflict (short) One or more privileges conflict with one or more other privileges.  (long).

Explanation:  This message is issued by the FM/DB2 Object Privileges utility. An example that causes this error is selecting the 'ALL' option for a privilege, as well as one or more of the individual privileges that are included when 'ALL' is specified.

User response:  Correct the specification of the privileges to be processed. In general selecting the 'ALL' option means that you should deselect all of the privileges that are included when 'ALL' is specified.

FMNDB911  Not required (short) This field should be blank for the specified option.  (long).

Explanation:  On the "Alter Table - VALIDPROC" panel, the "type of VALIDPROC" is set to 1, and a value is also specified for "program name". A value for "program name" is only required when the "type of VALID-PROC" is set to 2.

User response:  Check the "type of VALIDPROC" specification. If NULL is the intended specification, remove the value from the "program name" field, and retry the operation. If the intention is to add a program name, change the "type of VALIDPROC" specification to 2, and retry the operation.

FMNDB913  Invalid option (short) The selected option is not one of the listed values or the object type is not supported on the remote DB2 system you specified. This error can occur when you try to list certain DB2 objects on a remote DB2 system that is at an earlier release of DB2 than the system to which you are connected.

Explanation:  This message is issued by the FM/DB2 Object Privileges utility. The value entered in the "object type" field is invalid. The request cannot be processed.

User response:  The valid options for the "Object type" field are listed on the panel. Specify one of the values listed (1-15 inclusive).

FMNDB914  Different DB2 release (short) The release level of the remote DB2 system is different than the release level of the DB2 system to which you are connected. The choice of DB2 object types has been refreshed to show those object types supported on the remote system. Press ENTER again to get the requested list. (long).

Explanation:  This message is issued by the FM/DB2 Object List and Object Privileges utilities when the user enters a remote location value. FM/DB2 connects to the specified location and compares the DB2 version of the remote location against the version of the currently connected DB2 system. When the values are different this message is issued.

User response:  No action required. Informational message. Press enter to refresh the list of available object types.

FMNDB915  Empty table (short) The DB2 object contains no rows, or no rows match the specified row selection criteria.  (long).

Explanation:  This message is issued by the FM/DB2 editor when the result table for an object or select statement contains no rows.

User response:  No action required. Informational message.

FMNDB916  Excluded rows shadow is off. There are no unexcluded rows to display.  (long).

Explanation:  In an FM/DB2 editor session, the shadow-line for excluded-rows has been turned off, and there are no rows to display. Note, that when the excluded-row shadow-line has been turned off, the string 'SHAD' appears immediately to the left of "Format", at the top right of the display, in TABL mode. For those familiar with ISPF shadow lines, the FM/DB2 editor allows the display of the shadow-line indicator to be suppressed, using a command like "SHADOW X OFF". This is not supported in the ISPF editor. Excluded lines are always indicated with a shadow-line. When excluded-row shadow-lines are turned off and when all of the rows in the current editor session are shadow-lines, the editor display appears as though there are no rows in the table (no shadow-line indicator). This message is displayed to alert the user to the fact that there are rows in the result table, even though there are no lines displayed.

User response:  No action required. Informational message. You can turn on the shadow-line indicator with "SHADOW X ON".
FMNDB917 • FMNDB924

FMNDB917  No rows to display.

Explanation: In an FM/DB2 editor session, the shadow-line for excluded-rows is turned on, and there are no rows to display. See also the message description for FMNDB916. This message is typically issued when edit operations have deleted all rows from the current editor session.

User response: No action required. Informational message.

FMNDB918  Excluded rows shadow is off. Either no unexcluded rows to display, or no rows match the filter specification. (long).

Explanation: In an FM/DB2 object list/privileges display, the shadow-line for excluded-rows has been turned off, and there are no rows to display. Note, that when the excluded-row shadow line has been turned off the string 'SHAD' appears immediately to the left of 'Format', at the top right of the display, in TABL mode. The FM/DB2 object list/privileges utilities use the FM/DB2 editor to display lists of objects and privileges. When a filter value is specified for any column, the editor display is adjusted to show only those rows that match the filter value for that column. Any nonmatching rows are converted to excluded-rows, and these rows do NOT produce a shadow-line. This occurs regardless of the current setting for the excluded-row shadow-line. For those familiar with ISPF shadow-lines, the FM/DB2 editor allows the display of the shadow-line indicator to be suppressed, using a command like 'SHADOW X OFF'. This is not supported in the ISPF editor. Excluded-lines are always indicated with a shadow-line. When all of the rows in the current display are shadow-lines, the editor display appears as though there are no rows object list/privilege display (no shadow-line indicator). This message is displayed to alert the user to the fact that there are rows in the object list/privilege display, even though there are no lines displayed. This message is displayed when either:
• A filter value entered for a column results in no matches, or;
• When all rows are excluded.

User response: No action required. Informational message.

FMNDB919  No rows to display, or no rows match the filter specification. (long).

Explanation: In an FM/DB2 object/privileges display, the shadow-line for excluded-rows is turned on, and there are no rows to display. The FM/DB2 object list/privileges utilities use the FM/DB2 editor to display lists of objects and privileges. See also the message description for FMNDB918. This message is displayed when a filter value entered for a column results in no matches.

User response: No action required. Informational message.

FMNDB921  No rows match key (short) There are no rows in the parent table that match the foreign key value. This can occur when a foreign key column(s) has the NULL value. (long).

Explanation: A related edit session of a parent table has no rows that match a foreign key in a dependent table.

User response: No action required. As noted in the long message text, this condition can occur when a foreign key value in the dependent table is NULL.

FMNDB922  No rows match key (short) There are no rows in the dependent table that match the primary/parent key value. (long).

Explanation: A related edit session of a dependent table has no rows with a foreign key that match the primary or parent key value in the parent table.

User response: No action required. A parent/primary key value may exist for which there are no corresponding rows (in terms of foreign key value) in the dependent table.

FMNDB923  Already granted warning (short) At least one of the authid(s) or role(s) already possesses the privilege(s) from the grantor.

Explanation: This message is issued by the FM/DB2 Object privileges utility when a GRANT statement returns an SQLCODE+562. This is a warning condition indicating that the GRANT has already been executed for one or more privilege/authid/role combination. The execution of the SQL statement is successful. Any privileges previously held by the target DB2 authids/roles from the grantor are retained (not changed). Any new privileges specified have been successfully GRANNTed.

User response: No action required. When there are multiple DB2 authids or roles as target for the GRANT, or multiple privileges being GRANNTed, it is not possible to determine which privileges were previously available.

FMNDB924  Grant option warning (short) The WITH GRANT OPTION was ignored for one or more of the privilege(s). (long).

Explanation: A GRANT statement generated by the FM/DB2 privileges utility was executed. The statement included the "WITH GRANT OPTION" clause. DB2 returned SQLCODE+558. This is a warning condition indicating that the "WITH GRANT OPTION" clause is redundant. The GRANT statement was successfully
executed. An example of an operation that results in this warning condition is attempting to grant the SYSADM privilege to a DB2 authid, with "WITH GRANT OPTION". The FM/DB2 privileges utility attempted the operation. Consult with your DB2 security administrator to obtain the necessary DB2 authority to execute the statement.

**User response:** No action required.

---

**FMNDB925** Not authorized (short) Your authorization ID does not have the authority to GRANT\|REVOKE the privilege type system privilege. DB2 returned an SQL code of -552 for SQL statement SL statement.

**Explanation:** The FM/DB2 privileges utility attempted to execute SQL statement SL statement. Execution failed with DB2 returning SQLCODE -552, which indicates that the user does not have the authority to execute the statement.

**User response:** This is an error condition. Consult with your DB2 security administrator to obtain the necessary DB2 authority to execute the statement.

---

**FMNDB926** Own authorization ID (short) An authorization ID cannot revoke a privilege from itself. (long).

**Explanation:** The FM/DB2 privileges utility was used to issue a REVOKE statement. Execution of the statement failed with DB2 returning SQLCODE -555. This error is issued when the REVOKE statement would have the effect of removing a privilege from a DB2 authid or role that is possessed by the same DB2 authid or role. This is not supported.

**User response:** Examine the grantees and remove the DB2 authid or role that corresponds to the current DB2 authid or role of the user.

---

**FMNDB927** Invalid name (short) The name is invalid. A database/table space/plan name must be an SQL ordinary identifier - no lower case characters. See the SQL Reference manual for a definition. DB2 returned an SQL code of -113 for SQL statement SQL statement.

**Explanation:** A database/table space/plan name entered using the FM/DB2 object privileges utility is invalid. An SQL ordinary identifier is required. See the SQL Reference for a definition. An ordinary identifier is an uppercase letter, followed by zero or more characters, each of which is an uppercase letter, a digit, or the underscore character.

**User response:** Correct the name and retry the operation.

---

**FMNDB928** Insufficient authority (short). You cannot revoke this privilege. The current SQLID (DB2 authid) does not have SYSADM or SYSCTRL authority. You can only revoke a privilege when the current SQLID (DB2 authid) originally granted the privilege. (long).

**Explanation:** The FM/DB2 privileges utility was used to attempt to issue a REVOKE statement. The user current DB2 authid D2 authid, is different to the DB2 authid originally used to grant the privilege. While it is possible to revoke a privilege originally granted by another DB2 authid, this requires SYSADM or SYSCTRL authority, which the current D2 authid does not have. The statement cannot be executed.

**User response:** In general, without either SYSADM or SYSCTRL authority, you will be unable to revoke DB2 privileges granted by another DB2 authid. Consult with your DB2 Security Administrator to obtain the necessary DB2 authority to execute the statement.

---

**FMNDB940** Invalid PIECESIZE (short). PIECESIZE in Kilobytes|Megabytes|Gigabytes must be a multiple of 2 between mn value and mx value. (long).

**Explanation:** An invalid value was entered for "piecesize" on the 'Create Index: Type' panel (DB2 Object Functions). mn value and mx value show the minimum and maximum values that can be used for the specified space-multiplier, which will be either kilobytes, megabytes, or gigabytes.

**User response:** Select an integer value between the two values shown that is a power of 2. For kilobytes, valid values are: 256, 512, 1024, .. 67,108,864. For Megabytes valid values are: 1, 2, 4 .. 65,536. For Gigabytes, valid values are: 1, 2, 4 .. 64.

---

**FMNDB941** Invalid combination (short) A VCAT name cannot be specified with a storage group name, primary, or secondary space requirements, or the ERASE option.

**Explanation:** An invalid combination of values was entered when attempting to create a table space, or index, using the DB2 Object Functions. When creating either a table space, or index, a VCAT name can be specified, indicating that space allocation is under user control. Otherwise, the VCAT name must be blank, and a storage group name can be specified, indicating that space allocation is controlled by DB2. In the latter case, primary and secondary space values may also be entered.

**User response:** Decide whether a VCAT name or storage group name is appropriate. In the former case do not specify any of: storage group name, primary or secondary space allocation values. In the latter case do
FMNDB942  •  FMNDB948

not specify a VCAT name value.

FMNDB942  Storage group required (short) Primary and secondary space requirements, and the ERASE option can only be specified if a storage group name is also specified. (long)

Explanation: An invalid combination of values was entered when attempting to create an index using the DB2 Object Functions. Either:

1. One or more non-blank values have been specified for "Primary space" or "Secondary space" on the "Create Index:Allocation" panel. However the "Storage group name" value is blank. You must enter a "storage group name" if you intend to specify space allocation values.

2. One or more non-blank values have been specified for "Primary space", "Secondary space", or "Erase data" on the "Create Index: Partitions" panel. However, the "Storage group name" value is blank. You must enter a "storage group name" if you intend to specify space allocation or an "Erase data" value.

User response: Enter a valid storage group name. Alternatively do not specify "Primary space" "Secondary space" or "Erase data" values.

FMNDB943  Invalid selection (short) Select or deselect columns using the character "S". Order columns explicitly in the key/index using numerics.

Explanation: An invalid character was entered in either the Create Table, or Create Index functions, of the DB2 Objects utility.

User response: You can select a column by typing an 'S' in the appropriate field. All other letters are invalid. If a column has previously been selected, you deselect it by typing 'S' in the appropriate field. Once a column has been selected, you can enter any numeric (1, 2, 3, and so on) to specify the order of the columns in the index, or key. On the "Create Table: Unique Constraints" panel you can also select a column by typing any numeric in the appropriate field, and deselect a column by removing the number in the appropriate field.

FMNDB944  Already used (short). Order number number was already set for column column name on row row number of the display.

Explanation: An invalid number was entered in either the "Create Table" or "Create Index" functions of the DB2 Objects utility. When defining an index, or key with multiple columns, the order of the columns in the index, or key, is specified using numbers. Thus, for an index comprising three columns (COL4, COL7 and COL10), you would enter the numbers 1, 2 and 3 (in some combination) against those three columns.

User response: The number for the column has already been used for another column. Determine which columns should be in the index or key, then assign a unique number to each column. The column with the lowest used number will be the first column in the index or key, followed by the column with next-lowest-used number, and so on.

FMNDB945  Maximum partitions (short). The insert (I) or repeat (R) command cannot be executed because the maximum number of partitions allowed by DB2 has already been defined. (long)

Explanation: The user issued the 'I' or 'R' command against the 254th row on the "Create Index: Partitions" panel of the DB2 Objects utility. The maximum number of partitions that can be defined is 254 (DB2 version 7), so extra partitions cannot be added.

User response: No action required.

FMNDB946  Invalid value (short). Enter one of the listed values: list of valid values. (long)

Explanation: An invalid value was entered for a field in the DB2 Objects function. list of valid values contains the acceptable values for the field in error.

User response: Specify a value shown in list of valid values.

FMNDB947  Set columns/partitions (short). You must use option 2 to select the columns comprising the index and option 5 to define the partitions before assigning partition values. (long)

Explanation: The user selected option 6 on the "Create Index" panel of the DB2 Objects function. Either the columns that should comprise the index, or the partitions for the index, have not been defined. Both tasks must be completed before partition values may be assigned.

User response: Select option 2 to confirm that the columns that should comprise the index have been identified; then select option 5 to define the partitions for the new index.

FMNDB948  IT:Define &$ITEM:TE &$ITEM---17

T:You must use option &$OPTN to define the &$ITEM for the &$OBJ :TE
T:before using this option.:TE

Explanation: T

User response: I
FMNDB949 Table not found (short). The table to be indexed was not found in the DB2 catalog. (long).

Explanation: The user entered a table name on the “Create Index” panel of the DB2 Objects function. That table name does not exist on the current server.

User response: Check the table name for spelling mistakes. You can use the FM/DB2 Object List utility to show all table names defined at the current DB2 server if required. Correct the table name and retry the operation.

FMNDB950 Set partition values (short) Partitions have been defined for this index using option 5. You must set partition values using option 6 before you can define the index.

Explanation: This message is issued by the “Create Index” dialog (DB2 Object Functions). For a partitioned index, the number of partitions is set using option 5 (Partitions). Once the partitions are defined, you also need to set the maximum values for each partition using option 6 (Partition values). The index cannot be created until the partition values have been specified.

User response: If a partitioned index is required, select option 6 (Partition values), and specified appropriate values. If a non-partitioned index is required, select option 5 (Partitions), and delete all but one partition entry.

FMNDB951 Update partition values (short) The column selections or partition definitions have changed since the partition values were set. You must reset the partition values using option 6 before you can define the index.

Explanation: This message is issued by the “Create Index” dialog (DB2 Object Functions). Partition values have been set for all partitions and columns previously specified. However, a subsequent change has invalidated the partition values. There are two possibilities:
1. The columns specified for the new index have changed (columns added or deleted).
2. The partitions specified using option 5 (Partitions) have changed. The index cannot be created until the partition values have been specified.

User response: If a partitioned index is required, select option 6 (Partition values) and specified appropriate values. If a non-partitioned index is required, select option 5 (Partitions) and delete all but one partition entry.

FMNDB952 Conflicting selection (short) value 1 is not valid because value 2.

Explanation: This message is issued in many places within the DB2 Object Function dialog. The SQL syntax used to create the object has restrictions on the use of both value 1 and value 2. In most cases you can resolve the error by specifying either value 1 or value 2, but not both. The Object cannot be created until the conflicting selections are resolved.

User response: Determine which of the two selections is appropriate and remove the unwanted selection. Retry the operation. If the appropriate choice cannot be determined, refer to the SQL Reference manual for the appropriate version of DB2.

FMNDB953 Errors retained (short) Invalid data has been saved because the CANCEL command was entered in response to an input verification message. The statement type statement may be invalid if executed: return to the previous panel and correct the error, or return to the panel name panel and start from the beginning.

Explanation: This message is issued in many places within the DB2 Object Function dialog. An invalid value was entered on an option panel for the current CREATE or DROP function, followed by the user pressing CANCEL (PF12 by default) to return to the option selection panel for the current CREATE or DROP function. The invalid value is unresolved and may cause problems if an attempt is made to executed the current CREATE or DROP function.

User response: In most cases the problem can be resolved by reselecting the previous option panel (where CANCEL was entered), and correcting the invalid values entered there. If this does not resolve the problem, return to the DB2 Object Functions panel and restart the operation.

FMNDB954 No suitable rows (short). The current table definition does not contain columns specified as not null. Use option 2 to define not null columns before defining :primary/unique keys. (long).

Explanation: This message is issued by the “Create Table” dialog (DB2 Object Functions). The user selected the “unique constraints” option after previously defining columns for the new table using option 1 (Columns). A column can only be included in a unique constraint key if it is defined as NOT NULL, and none of the columns currently defined for the new table meet this requirement.

User response: Select option 2 (Nulls/Default values) and set the Null status for each column as appropriate.
You will need to specify either NN (NOT NULL) or ND (NOT NULL WITH DEFAULT) for at least one column to be able to specify a unique constraint for the new table.

**FMNDB955**  
No suitable rows (short). The current table definition does not contain any non-ROWID, non-LOB columns specified as not null, nor any identity columns. Use option 2 to define not null columns, or option 8 to define identity columns before defining primary/unique keys. (long).

**Explanation:** This message is issued by the "Create Table" dialog (DB2 Object Functions). The user selected the "unique constraints" option after previously defining columns for the new table using option 1 (Columns). A column can only be included in a unique constraint key if it is defined as NOT NULL, and none of the columns currently defined for the new table meet this requirement. This message is issued when there are columns defined as ROWID, or CLOB/DBCLOB, BLOB.

**User response:** Select option 2 (Nulls/Default values), and set the Null status for each column as appropriate. You will need to specify either NN (NOT NULL) or ND (NOT NULL WITH DEFAULT) for at least one column to be able to specify a unique constraint for the new table. You can also define numeric columns as INDEIDITY columns using option 8 (Generate values). An IDENTITY column can be included in a unique constraint.

**FMNDB956**  
Too many selections (short). You have already selected Max column count columns for this key, which is the maximum allowed by DB2. (long).

**Explanation:** This message is issued by the "Create Table" dialog (DB2 Object Functions), usually when defining columns to be included in the primary key, or other unique constraint defined for the table. Max column count is 64 for DB2 for z/OS versions 8 - 10 inclusive.

**User response:** Reduce the number of columns in the primary key or unique constraint.

**FMNDB957**  
Table/view not found (short). The model table/view was not found in the DB2 catalog. (long).

**Explanation:** This message is issued by the "Create Table" dialog (DB2 Object Functions), when a model table name is entered, but the DB2 object does not exist.

**User response:** Correct the DB2 object name, retry the operation. You can use the Object List utility (Option 3.4) to show all the DB2 objects (tables/views/aliases ) defined at the local server.

**FMNDB958**  
Too many constraints (short) Too many unique constraints have been found in model tables. Only the first three unique constraints found in the DB2 catalog have been loaded.

**Explanation:** This message is issued by the "Create Table" dialog (DB2 Object Functions), when a model table name is entered and a load of the information for the model table is requested. The "Create table" dialog supports a maximum of three unique constraints for any table being defined. This message might be issued for the load of a single DB2 object, with 4 or more unique constraints defined, or when multiple loads are requested.

**User response:** This is an informational message. In the unlikely event that 4 or more unique constraints need to be defined for the table being created, consider defining the table with 3 initial constraints, and using ALTER TABLE to add any additional constraints.

**FMNDB959**  
Invalid option (short). Only column information can be loaded when the add option is selected and columns have already been defined or loaded. (long).

**Explanation:** This message is issued by the "Create Table" dialog (DB2 Object Functions), when a model table name is entered. The user loaded information for one model table, then attempted to add load additional information for another. For the second and subsequent loads, only the "Basic table information" and "Column information" options can be selected.

**User response:** You can only load "Primary key information", "Unique key information", "Referential constraints" and "Check constraint" information for the first model table specified. Deselect the option for all of the above and retry the operation. Alternatively set the Refresh or Add option to Refresh and repeat the operation. This will replace any existing information with that for the current model table.

**FMNDB960**  
Too many constraints (short) Too many table foreign keys have been defined for this table. The number of table foreign keys is limited to the number of ISPF tables named FMN2CTnn that can be created.

**Explanation:** This message is issued by the "Create Table" dialog (DB2 Object Functions). An attempt was made to create more than 100 foreign keys for the current table. While there is no DB2 limit on the number of foreign keys, the "Create Table" dialog is limited to a maximum of 100 foreign keys.

**User response:** Review the number of foreign keys
that have been specified, reduce the number to 100 or less.

**FMNDB961** Order required (short) A referenced column name has been entered for this column, but no order. An order is required if the column is to be included in the foreign key.

**Explanation:** This message is issued on the "Create Table: Table Referential Constraint" panel. When creating a table referential constraint that includes more than one column (the usual usage is for a multiple-column foreign key definition), an order specification is required for every column that is part of the foreign key being defined.

**User response:** Specify an order for every column that is to be part of the foreign key.

**FMNDB962** Ref. column required (short). If any of the foreign key columns reference a specific column in the parent table, then all of the foreign key columns must do so. (long).

**Explanation:** This message is issued on the "Create Table: Table Referential Constraint" panel. When creating a table referential constraint that includes more than one column (the usual usage is for a multiple-column foreign key definition), an order specification for a column was entered without a corresponding entry in the Parent Key column field.

**User response:** Specify a column name in the Parent Key column field, for every column that is to be part of the foreign key.

**FMNDB963** No columns selected (short). No columns have been selected to define the foreign key for this constraint. (long).

**Explanation:** This message is issued when creating referential constraints for a table (Create Table dialog of DB2 Object functions). The user created a new constraint name and specified the parent table for the relationship. However no foreign key columns have been defined.

**User response:** Use the 's' command to select the new constraint entry. When the "Create Table: Table Referential Constraint" panel is displayed, select the columns that comprise the foreign key, and enter the names of the corresponding columns in the parent table.

**FMNDB964** Some data not loaded (short). The DB2 catalog did not contain any information for the model table or view for at least one of the model load options specified. For example, check constraints were requested but the referenced model table has no check constraints. (long).

**Explanation:** This message is issued by the "Create Table" dialog (DB2 Object Functions), when a model table name is entered, and a load of existing DB2 object information is requested.

**User response:** No action required. Informational message.

**FMNDB965** No suitable rows (short) The current table definition does not contain any short string columns specified as not null. Use option 2 to define not null columns before defining Procedure Exits.

**Explanation:** This message is issued by the "Create Table" dialog (DB2 Object Functions). The user selected option 7 (Procedure exits). However, there are no suitable columns defined on which a procedure exit can be defined. A procedure exit cannot be defined unless a suitable column is defined.

**User response:** Option 7 (Procedure exits) allows a FIELDPROC specification for suitable columns of the table. In general, only character (CHARACTER/GRAFIC) columns of length <=255, defined as NOT NULL, or NOT NULL WITH DEFAULT, are suitable. Define a suitable column using options 1 (Columns) and 2 (Nulls/Default values) and then reselect option 7.

**FMNDB966** External name required (short). Specify an external name for the object type using option number before using this option. This is required because the specified name exceeds eight characters. (long).

**Explanation:** This message is issued by the "Create Function" and "Create Stored Procedure" dialogs of the DB2 Object Functions. The length of the name for the function, or stored procedure, exceeds 8 characters. In this situation an external name for the function must also be specified. object type is either "function" or "stored procedure", depending on the Object function. number is 4 when creating a function, and 3 when creating a stored procedure.

**User response:** Select the specified option, and enter an external name for the function or stored procedure. Consult the DB2 for z/OS SQL Reference for the appropriate version of DB2 for details of acceptable formats for the external name value.
FMNDB967  String too long (short). The string constant specified as a default value is too long for the associated string column. (long).

Explanation: This message is issued by the "Create Table" dialog (DB2 Object Functions) when specifying a default value for a character data type column. The default value is too long. For example, specifying a default value of 'ABCDE' (length 5) for a CHAR(3) (length 3) column.

User response: Compare the length of the default value string, and the maximum length for the column, as defined in the length field for the data type. Adjust either value as appropriate.

FMNDB968  Invalid data (short). The decimal number specified as a default value is invalid or out of range for the associated decimal number column. (long).

Explanation: This message is issued by the "Create Table" dialog (DB2 Object Functions) when specifying a default value for a DECIMAL data type column. The specified default value is invalid in the context of the precision and scale specified for the column.

User response: Compare the definition for the DECIMAL column with the implied precision, and scale from the default value. Adjust either the data type (precision and scale) in the column, or the default value.

FMNDB969  Invalid DATE | TIME | TIMESTAMP (short) The format of the DATE | TIME | TIMESTAMP field or the data it contains is invalid.

Explanation: This message is issued by the "Create Table" dialog (DB2 Object Functions) when specifying a default value for a DATE, TIME, or TIMESTAMP data type column. The specified default value is invalid.

User response: Examine the default value entered and the data type for the column. The two possible causes of this error are:
1. An invalid format. Example: '12/10/2001' is not a valid format for a TIME column.
2. An invalid value. Example: '32/11/2001' is an invalid date, since 32 is not a valid day (or month).

Consult the DB2 for z/OS SQL Reference for the appropriate version of DB2 for a detailed description of valid DATE/TIME and TIMESTAMP constants, including supported formats.

FMNDB970  Date/time not validated (short). The default values for one or more date or time fields were not validated because the DB2 subsystem date/time format is LOCAL or unknown.

Explanation: This message is issued by the "Create Table" dialog (DB2 Object Functions) when specifying a default value for a DATE or TIME column. The DB2 default for either the DATE or TIME format is either LOCAL, or unknown. In either case, the panel logic cannot validate the specified default value.

User response: No action required. Informational message. If any default value entered is invalid, an SQL error will occur when the attempt is made to create the DB2 object.

FMNDB971  Date/time not validated (short) The specified default date or time value was not validated because the DB2 subsystem date/time format is LOCAL or unknown. (long).

Explanation: This message is issued by the "Create Table" dialog (DB2 Object Functions) when specifying a default value for a DATE, TIME, or TIMESTAMP column. This message is issued on pop-up panel "Create Table: Default Value". The DB2 default for either the DATE or TIME format is either LOCAL, or unknown. In either case, the panel logic cannot validate the specified default value.

User response: No action required. Informational message. If any default value entered is invalid an SQL error will occur when the attempt is made to create the DB2 object.

FMNDB972  Invalid data (short). The floating point number specified as a default value is invalid or outside the range supported by DB2. (long).

Explanation: This message is issued by the "Create Table" dialog (DB2 Object Functions) when specifying a default value for a binary floating point column (REAL, FLOAT, DOUBLE). The specified default value is outside the range supported by DB2.

User response: The limits for floating point constants are documented in "Limits in DB2 for z/OS", in the DB2 for z/OS SQL Reference manual, for the appropriate version of DB2. Compare the specified default value against the documented values, and adjust as appropriate.

FMNDB973  GENERATED ALWAYS added (short). The GENERATED ALWAYS clause has been added to the definition of the column with a data type of ROWID. To change the GENERATED clause select option 8. (long).
Explanation: This message is issued by the "Create Table" dialog (DB2 Object Functions), when adding a column definition for a ROWID column.

User response: No action required. Informational message. In most cases it should not be necessary to modify the GENERATED clause for a ROWID column. Option 8 (Generated Values) allows the GENERATED clause to be modified.

FMNDB974 Invalid database name (short). You cannot specify the name of an implicitly created database when creating a table. Implicitly created database names have the form DSNxxxxx, where xxxx is any 5 numeric digits. (long)

Explanation: This message is issued by the "Create Table" dialog (DB2 Object Functions) when attempting to create a table. The database name specified is invalid because it has the form 'DSNxxxxx', where xxxx is 00000-99999 inclusive. Names of this form are reserved for use by DB2, as implicitly created database names. The table cannot be created.

User response: Specify a different DB2 database name, or remove the database name completely if an implicitly required database is acceptable.

FMNDB980 Invalid value (A/D) (short). The valid values for the A/D field are A, for ascending; D, for descending, or blank. (long)

Explanation: This message is issued on the "Column Selection/Edit" panel, when a value other than 'A' or 'D' is entered in the "A/D" column.

User response: The "A/D" column in the "Column Selection/Edit" panel is used to specify the sort order for columns included in the ORDER BY clause. The valid values are 'A' for ascending (the default) or 'D' for descending. Specify one of these values, or leave the field blank for the default behaviour.

FMNDB981 Enter sort order (short). A sort order is required when a sort column is specified (A or D in the A/D field). Enter a number in the range 1..999. (long)

Explanation: The 'Order' column in the "Column Selection/Edit" panel is used to specify the sort sequence for columns included in the ORDER BY clause. The user specified either 'A' or 'D' in the "A/D" column, but no value in the 'Order' column.

User response: Enter a number in the 'Order' column for every column that also has a non-blank value in the "A/D" column.

FMNDB990 Mutually exclusive value (short). Specify a JOBNAME value, or an ADDRESS value, but not both. (long)

Explanation: This message is issued by the "Create Trusted Context" dialog (DB2 Object Functions). A value was entered for both the JOBNAME and ADDRESS fields, however only one value is acceptable.

User response: Specify either a JOBNAME value; or an ADDRESS value, but not both. The syntax for the CREATE TRUSTED CONTEXT SQL statement can be found in the DB2 for z/OS SQL Reference manual for the appropriate version of DB2.

FMNDB991 Value required (short). Either a JOBNAME value or an ADDRESS value is required. (long)

Explanation: This message is issued by the "Create Trusted Context" dialog (DB2 Object Functions). No value was entered in either the JOBNAME or ADDRESS fields; a value is required for one.

User response: Specify either a JOBNAME value; or an ADDRESS value, but not both. The syntax for the CREATE TRUSTED CONTEXT SQL statement can be found in the DB2 for z/OS SQL Reference manual for the appropriate version of DB2.

FMNDB992 Value not required (short). Do not specify a "Default Role Name" value when the "No Default Role" option is selected. (long)

Explanation: This message is issued by the "Create Trusted Context" dialog (DB2 Object Functions). The user selected the "No default role" option, and also entered a value in the "Default role name" field. The two selections are mutually exclusive.

User response: If a default role name is to be specified, deselect the "No default role" option. If there is to be no default role name, remove the "Default role name".

FMNDB993 Value not required (short). Do not specify a "Security Label Name" value when the "No Default Security Label" option is selected. (long)

Explanation: This message is issued by the "Create Trusted Context" dialog (DB2 Object Functions). The user selected the "No default security label" option, and also entered a value in the "Security label name" field. The two selections are mutually exclusive.

User response: If a default security label name is to be specified, deselect the "No default security label" option. If there is to be no default security label, remove the "Security label name".
FMNDB994  FMNDC015

FMNDB994  Value not required (short). Do not select the "Without authentication" option when the "PUBLIC" option is selected. (long).

Explanation: This message is issued by the "Create Trusted Context" dialog (DB2 Object Functions). The user did not select the "PUBLIC" use option, and selected the "Without authentication" option. The second selection requires that the first option also be selected.

User response: The valid choices for the two options ("PUBLIC" and "Without authentication") are: 1) Both options not selected. 2) Both options selected. 3) Only the first option selected.

FMNDC000  Count rows

Explanation: This message is issued by the FM/DB2 Print function; it shows the progressive total when rows are being printed.

User response: No action required. Informational message.

FMNDC001  Count row(s) printed (short). Count row(s) printed from DB2 object DB2 Object name. (long).

Explanation: This message is issued by the FM/DB2 Print function; it shows the total number of rows printed and is issued when printing is complete.

User response: No action required. Informational message.

FMNDC002  Object list printed.

Explanation: This message is issued when a list of DB2 objects is printed by the FM/DB2 Object List utility (P command). The requested object list was successfully printed.

User response: No action required. Informational message.

FMNDC010  DB2 SSID: DB2 SSID SQL ID: SQLID

Explanation: This is a message insert used by the FM/DB2 Print utility. It shows the DB2 SSID of the currently connected DB2 system, and the user current DB2 SQLID.

User response: No action required. Informational message.

FMNDC011  Location: Location Table/View: Object Name

Explanation: This is a message insert used by the FM/DB2 Print utility. It shows the DB2 server name (Location) of the DB2 object being printed (blank indicates the locally connected DB2 system) and the DB2 Object Name Object Name being printed.

User response: No action required. Informational message.

FMNDC012  Row Number -

Explanation: This is a message insert used by the FM/DB2 Print utility. It shows the row number for the row being printed.

User response: No action required. Informational message.

FMNDC013  Location: Location Object Name: Object Name

Explanation: This is a message insert used by the FM/DB2 Print utility. It shows the DB2 server name (Location) of the DB2 object being printed (blank indicates the locally connected DB2 system) and the DB2 Object Name Object Name being printed.

User response: No action required. Informational message.

FMNDC014  Rows rows exported (short). Rows row(s) exported from SELECT statement (user-specified) to file data set name. (long).

Explanation: This is a message insert used by the FM/DB2 Export function. It shows the number of rows rows exported by the export function. The target data set is data set name. This message is specific to the export of data from a user-specified SELECT statement initially entered using FM/DB2 options 4.1-4.4.

User response: No action required. Informational message.

FMNDC015  Rows rows+hdr expd (short). Rows row(s) and column header row exported from SELECT statement (user-specified) to file data set name. (long).

Explanation: This is a message insert used by the FM/DB2 Export function. It shows the number of rows rows exported by the export function. The target data set is data set name. This message is specific to the export of data in CSV format, with column header data, from a user-specified SELECT statement initially entered using FM/DB2 options 4.1-4.4.

User response: No action required. Informational message.
FMNDC019  Not in large mode (short). The export command with the ALL|X|EX|NX parameter is not available in large editor mode (row count = 0). (long).

Explanation: The editor EXPORT command with the ALL|X|EX|NX parameter specified was issued from an FM/DB2 editor session, in large mode (row count = 0, also ALL, *.). This is not supported and the command cannot be processed.

User response: Use normal editor mode and re-try the operation.

FMNDC020  count row(s) exported from DB2 object
source DB2 object to file file name. The open returned SQLCODE sqlcode.
trunc_count rows had fetch SQLCODE +445 (truncation) warnings, warn_count other rows had fetch warnings (SQLCODE>0, but not +445).

Explanation: This message is issued at the end of an FM/DB2 export function, when at least one positive SQLCODE was received retrieving data from DB2. The number of rows exported, count; the DB2 object from which the rows were exported, source DB2 object; the output file name that contains the exported data, file name; the SQLCODE for the OPEN cursor, sqlcode; the number of rows fetched that had SQLCODE+445 warnings, trunc_count and the number of rows fetched that gave positive SQLCODEs (but not +445), warn_count are shown. SQLCODE+445 is a warning condition issued when data truncation occurs.

User response: For SQLCODE+445 warnings, examine the exported data to see if the data truncation has resulted in any unexpected consequences. Take action as appropriate. For other warning SQLCODEs: If the export operation was run on-line, repeat the operation using the batch execution option. Run the resulting job and examine the messages produced. Look up the SQLCODEs reported in the DB2 for z/OS SQL Codes manual, for the appropriate version of DB2. Take action as appropriate.

FMNDC021  Row count row(s) and column header row exported from DB2 object object name to file data set name. The open returned SQLCODE sqlcode. trunc_count rows had fetch SQLCODE +445 (truncation) warnings, warn_count other rows had fetch warnings (SQLCODE>0, but not +445).

Explanation: This message is issued by the FM/DB2 export function, when at least one positive SQLCODE was received retrieving data from DB2 and is specific for an export with CSV format and the "Include column headers" option selected. The number of rows exported, count; the DB2 object from which the rows were exported, source DB2 object; the output file name that contains the exported data, file name; the SQLCODE for the OPEN cursor, sqlcode; the number of rows fetched that had SQLCODE+445 warnings, trunc_count and the number of rows fetched that gave positive SQLCODEs (but not +445), warn_count are shown. SQLCODE+445 is a warning condition issued when data truncation occurs.

User response: For SQLCODE+445 warnings, examine the exported data to see if the data truncation has resulted in any unexpected consequences. Take action as appropriate. For other warning SQLCODEs: If the export operation was run on-line, repeat the operation using the batch execution option. Run the resulting job and examine the messages produced. Look up the SQLCODEs reported in the DB2 for z/OS SQL Codes manual, for the appropriate version of DB2. Take action as appropriate.

FMNDC022  Rows row(s) exported from SELECT statement (user-specified) to file data set name. The open returned SQLCODE sqlcode, trunc_count rows had fetch SQLCODE +445 (truncation) warnings, warn_count other rows had fetch warnings (SQLCODE>0, but not +445).

Explanation: This message is issued by the FM/DB2 export function when at least one positive SQLCODE was received retrieving data from DB2. It shows the number of rows rows exported by the export function; the target data set is data set name; the SQLCODE for the OPEN cursor is sqlcode; the number of rows fetched that had SQLCODE+445 warnings is trunc_count and the number of rows fetched that gave positive SQLCODEs (but not +445) is warn_count.

SQLCODE+445 is a warning condition issued when data truncation occurs.

User response: For SQLCODE+445 warnings, examine the exported data to see if the data truncation has resulted in any unexpected consequences. Take action as appropriate. For other warning SQLCODEs: If the export operation was run on-line, repeat the operation using the batch execution option. Run the resulting job and examine the messages produced. Look up the SQLCODEs reported in the DB2 for z/OS SQL Codes manual, for the appropriate version of DB2. Take action as appropriate.

FMNDC023  Rows row(s) and column header row exported from SELECT statement (user-specified) to file data set name. The open returned SQLCODE sqlcode, trunc_count rows had fetch SQLCODE +445 (truncation) warnings, warn_count other rows had fetch warnings (SQLCODE>0, but not +445).

Explanation: This message is issued by the FM/DB2 export function, when at least one positive SQLCODE was received retrieving data from DB2 and is specific for an export with CSV format and the "Include column headers" option selected. The number of rows exported, count; the DB2 object from which the rows were exported, source DB2 object; the output file name that contains the exported data, file name; the SQLCODE for the OPEN cursor, sqlcode; the number of rows fetched that had SQLCODE+445 warnings, trunc_count and the number of rows fetched that gave positive SQLCODEs (but not +445), warn_count are shown. SQLCODE+445 is a warning condition issued when data truncation occurs.

User response: For SQLCODE+445 warnings, examine the exported data to see if the data truncation has resulted in any unexpected consequences. Take action as appropriate. For other warning SQLCODEs: If the export operation was run on-line, repeat the operation using the batch execution option. Run the resulting job and examine the messages produced. Look up the SQLCODEs reported in the DB2 for z/OS SQL Codes manual, for the appropriate version of DB2. Take action as appropriate.
Explanation: This message is issued by the FM/DB2 export function when at least one positive SQLCODE was received retrieving data from DB2. It shows the number of rows rows exported by the export function, the target data set is data set name; the SQLCODE for the OPEN cursor is sqlcode; the number of rows fetched that had SQLCODE+445 warnings is trunc_count and the number of rows fetched that gave positive SQLCODEs (but not +445) is warn_count.

SQLCODE+445 is a warning condition issued when data truncation occurs. This message is specific to the export of data in CSV format, with column header data, from a user-specified SELECT statement initially entered using FM/DB2 options 4.1-4.4.

User response: For SQLCODE+445 warnings, examine the exported data to see if the data truncation has resulted in any unexpected consequences. Take action as appropriate. For other warning SQLCODEs: If the export operation was run on-line, repeat the operation using the batch execution option. Run the resulting job and examine the messages produced. Look up the SQLCODEs reported in the DB2 for z/OS SQL Codes manual, for the appropriate version of DB2. Take action as appropriate.

FMNDC026 Open cursor SQLCODE=sqlcode.
Explanation: This message is issued by the FM/DB2 export function, when executed in batch mode, when a positive SQLCODE is received from DB2 when the cursor is opened.
User response: Look up the SQLCODE reported in the DB2 for z/OS SQL Codes manual, for the appropriate version of DB2. Take action as appropriate.

FMNDC027 Fetch SQLCODE=sqlcode for row row (data truncation).
Explanation: This message is issued by the FM/DB2 export function, when executed in batch mode, whenever an SQLCODE+445 is received when fetching a row of data from DB2. SQLCODE+445 indicates that data truncation occurred.
User response: Examine the exported data to see if the data truncation has resulted in any unexpected consequences. Take action as appropriate.

FMNDC028 Fetch SQLCODE=sqlcode for row row (warning).
Explanation: This message is issued by the FM/DB2 export function, when executed in batch mode, whenever a positive SQLCODE other than +445 is received when fetching a row of data from DB2.
User response: Look up the SQLCODE reported in the DB2 for z/OS SQL Codes manual, for the appropriate version of DB2. Take action as appropriate.
**FMNDC110**  Skip locked rows

**Explanation:**  This is a translatable message displayed on FM/DB2 editor options panels.

**User response:**  No action required. Panel text.

**FMNDC111**  Keep locks option

**Explanation:**  This is a translatable message displayed on FM/DB2 editor options panels.

**User response:**  No action required. Panel text.

**FMNDC112**  1. Use/Keep share locks

**Explanation:**  This is a translatable message displayed on FM/DB2 editor options panels.

**User response:**  No action required. Panel text.

**FMNDC113**  2. Use/Keep update locks

**Explanation:**  This is a translatable message displayed on FM/DB2 editor options panels.

**User response:**  No action required. Panel text.

**FMNDC114**  3. Use/Keep exclusive locks

**Explanation:**  This is a translatable message displayed on FM/DB2 editor options panels.

**User response:**  No action required. Panel text.

**FMNDC170**  Changes discarded (short). The invalid change to the data format option has been discarded. Enter one of the following values: 1, 2, 3, 4. (long).

**Explanation:**  The user made an invalid change to the "Data format" value on the Import options panel, and then pressed the PFKey assigned to END (or typed END). The invalid value is discarded and the import options panel is redisplayed.

**User response:**  No action required. Informational message. After making changes to the options, press ENTER to validate the changes before attempting to exit the panel.

**FMNDC171**  Changes discarded (short). The invalid change to the duplicate key processing option has been discarded. Enter one of the following values: 1, 2. (long).

**Explanation:**  The user made an invalid change to the "Duplicate key processing" value on the Import options panel, and then pressed the PFKey assigned to END (or typed END). The invalid value is discarded and the import options panel is redisplayed.

**User response:**  No action required. Informational message. After making changes to the options, press ENTER to validate the changes before attempting to exit the panel.

**FMNDC172**  Changes discarded (short). The invalid change to the maximum duplicates option has been discarded. Enter a number in the range 0-2147483647 inclusive. (long).

**Explanation:**  The user made an invalid change to the "Max duplicates" value on the Import options panel, and then pressed the PFKey assigned to END (or typed END). The invalid value is discarded and the import options panel is redisplayed.

**User response:**  No action required. Informational message. After making changes to the options, press ENTER to validate the changes before attempting to exit the panel.

**FMNDC174**  Changes discarded (short). The invalid change to the auto-commit count option has been discarded. Enter a number in the range 0-2147483647 inclusive. (long).

**Explanation:**  The user made an invalid change to the "Auto commit count" value on the Import options panel, and then pressed the PFKey assigned to END (or typed END). The invalid value is discarded and the import options panel is re-displayed.

**User response:**  No action required. Informational message. After making changes to the options, press ENTER to validate the changes before attempting to exit the panel.

**FMNDC180**  Enter a value (short). A value is required. Enter a frequency between 0 and 1 exclusive. As an example 0.1 will sample approximately 10% of rows. (long).

**Explanation:**  The user selected "random sampling" on the FM/DB2 Editor options panel, but did not enter a value for "sampling frequency". A sampling frequency is required when using random sampling.

**User response:**  Enter a decimal value between 0.0 and 1.0. As an example, enter 0.1 will result in approximately 10% of the rows in the object being sampled. These rows will appear in the FM/DB2 editor session. Higher values will result in more rows being sampled; lower values will result in less rows being sampled.

**FMNDC181**  Invalid value (short). Valid frequencies are 0.000000001 - 0.999999999 inclusive. Enter a value within these limits. (long).

**Explanation:**  The user selected "random sampling" on the FM/DB2 Editor options panel, but did not enter a value for "sampling frequency". A sampling frequency is required when using random sampling.

**User response:**  Enter a decimal value between 0.0 and 1.0. As an example, enter 0.1 will result in approximately 10% of the rows in the object being sampled. These rows will appear in the FM/DB2 editor session. Higher values will result in more rows being sampled; lower values will result in less rows being sampled.
valid value for "sampling frequency". A valid sampling frequency is a decimal number between 0.0 and 1.0; with 9 digits of precision after the decimal point.

**User response:** Enter a decimal value between 0.0 and 1.0. As an example, enter 0.1 will result in approximately 10% of the rows in the object being sampled. These rows will appear in the FM/DB2 editor session. Higher values will result in more rows being sampled; lower values will result in less rows being sampled.

**Explanation:** The user selected "random sampling" on the FM/DB2 Editor options panel, but did not enter a valid value for "sampling seed". The seed value is used to initiate the pseudo-random number generator.

**User response:** Enter an integer between 0-2147483647 inclusive. A value of 0 sets the seed to a value based on the current value of the system clock on the computer, accurate to the microsecond, and is therefore a random value for most purposes. Any other value will result in repeatable behaviour, in terms of the rows sampled, for a given DB2 object and the same sampling frequency.

**Explanation:** The user made an invalid change to the "Sampling seed" value on an Editor options panel, and then pressed the PFKey assigned to END (or typed END). The invalid value is discarded and the editor options panel is redisplayed.

**User response:** No action required. Informational message. After making changes to the options, press ENTER to validate the changes before attempting to exit the panel.
options panel is redisplayed.

User response: No action required. Informational message. After making changes to the options, press ENTER to validate the changes before attempting to exit the panel.

FMNDC189 Changes discarded (short). The invalid change to the Change count before auto commit has been discarded. The Change count before auto commit value must be in the range 0-2147483647. (long).

Explanation: The user made an invalid change to the "Change count before auto commit" value on an Editor options panel, and then pressed the PFKey assigned to END (or typed END). The invalid value is discarded and the editor options panel is redisplayed.

User response: No action required. Informational message. After making changes to the options, press ENTER to validate the changes before attempting to exit the panel.

FMNDC190 Changes discarded (short). The invalid change to the Prefix Length option has been discarded. (long).

Explanation: The user made an invalid change to the "Prefix length" value on an Editor options panel, and then pressed the PFKey assigned to END (or typed END). The invalid value is discarded and the editor options panel is redisplayed.

User response: No action required. Informational message. After making changes to the options, press ENTER to validate the changes before attempting to exit the panel.

FMNDC191 Changes discarded (short). The invalid change to the input/display characters has been discarded. The input/display delimiter characters must be non-blank and unique. (long).

Explanation: The user made an invalid change to the String delimiter and/or Null column indicator input/display characters on an Editor options panel, and then pressed the PFKey assigned to END (or typed END). The invalid value is discarded and the editor options panel is redisplayed.

User response: No action required. Informational message. After making changes to the options, press ENTER to validate the changes before attempting to exit the panel.

FMNDC192 Changes discarded (short). The invalid change to the Edit locking option has been discarded. Enter one of the listed values for the Edit locking option. (long).

Explanation: The user made an invalid change to the "Edit locking" value on an Editor options panel, and then pressed the PFKey assigned to END (or typed END). The invalid value is discarded and the editor options panel is redisplayed.

User response: No action required. Informational message. After making changes to the options, press ENTER to validate the changes before attempting to exit the panel.

FMNDC193 Changes discarded (short). The invalid change to the "Optimization for" value has been discarded. Enter a number in the range 0-999999 inclusive. Use 0 if no optimization clause is required. (long).

Explanation: The user made an invalid change to the "Optimization for" value on an Editor options panel, and then pressed the PFKey assigned to END (or typed END). The invalid value is discarded and the editor options panel is redisplayed.

User response: No action required. Informational message. After making changes to the options, press ENTER to validate the changes before attempting to exit the panel.

FMNDC194 Changes discarded (short). The invalid change to the Keep locks option has been discarded. Enter one of the listed values for the Keep locks option. (long).

Explanation: The user made an invalid change to the "Keep locks" value on an Editor options panel, and then pressed the PFKey assigned to END (or typed END). The invalid value is discarded and the editor options panel is redisplayed.

User response: No action required. Informational message. After making changes to the options, press ENTER to validate the changes before attempting to exit the panel.

FMNDC195 Changes discarded (short). The invalid change to the Scrollable cursor type option has been discarded. Enter one of the listed values for the Scrollable cursor type option. (long).

Explanation: The user made an invalid change to the "Scrollable cursor type" value on an Editor options panel, and then pressed the PFKey assigned to END (or typed END). The invalid value is discarded and the editor option's panel is redisplayed.

User response: No action required. Informational message. After making changes to the options, press ENTER to validate the changes before attempting to exit the panel.
FMNDC196 Changes discarded (short). The invalid change to the Data sampling type option has been discarded. Enter one of the listed values for the Data sampling type option. (long).

Explanation: The user made an invalid change to the 'Data sampling type' value on an Editor options panel, and then pressed the PFKey assigned to END (or typed END). The invalid value is discarded and the editor options panel is redisplayed.

User response: No action required. Informational message. After making changes to the options, press ENTER to validate the changes before attempting to exit the panel.

FMNDC197 Enter a positive number (short). A positive number is required for this option when clustered sampling is selected. Enter a number in the range 1-999999999 inclusive. (long).

Explanation: The user selected “clustered sampling” on an FM/DB2 Editor options panel, but entered an invalid value for one of the “Include count” field. A positive integer value in the indicated range is required.

User response: Enter a positive integer value in the range 1-999999999 inclusive.

FMNDC198 Invalid value or number (short). A number is required for this option. Valid values are range inclusive. (long).

Explanation: The user selected “cluster sampling”, or “random sampling” on an FM/DB2 Editor options panel, but entered an invalid value in one of the other sampling fields. Valid values for the indicated field are shown in the message range.

User response: Enter a value in the range shown in the long message. To access the long message, press the PFKey assigned to HELP (typically PF1), or type HELP on the command line, followed by ENTER.

FMNDC199 Enter a number (short). A number is required for the optimization clause. Valid values are 0-999999 inclusive. Use 0 if no optimization clause is required. (long).

Explanation: The user cleared the “Optimize for” value on an FM/DB2 Editor options panel (and pressed ENTER). A value is required for this field.

User response: Enter a value in the range 0-999999 inclusive. Specifying 0 results in no “OPTIMIZE FOR nn ROWS” clause being added to the SELECT statement used to fetch the data for the current editor session. In some situations specifying a small value (eg 1) may change the access path used by DB2 when retrieving the data, thereby improving the response time. See the DB2 SQL Reference manual for a more complete explanation.

FMNDC200 Invalid label range (short). The label range entered is invalid. Press help for more information. (long).

Explanation: The user entered the EXPORT editor primary command, with labels. The labels specified are invalid. The export command cannot be performed.

User response: Check the labels specified carefully. One common cause of this error is specifying a label that does not exist. You can press the PFKey assigned to HELP (PF1 by default) to display additional help for this error.

FMNDC201 NX|X|EX invalid (short). The NX|X|EX parameters cannot be specified when in SNGL mode. Press help for more information. (long).

Explanation: The user entered the EXPORT editor primary command, in SNGL mode, with the NX, X or EX option. The excluded or not-excluded row options are not applicable in SNGL mode. The export command cannot be performed.

User response: Change the display mode to TABL and re-issue the command. Alternatively, remove the excluded or not-excluded option and re-issue the command. You can press the PFKey assigned to HELP (PF1 by default) to display additional help for this error.

FMNDC202 Not in zoom mode (short). The export primary command is invalid in Zoom mode. (long).

Explanation: The user entered the EXPORT editor primary command, in ZOOM mode, which is not supported. The export command cannot be performed.

User response: Return to either TABL or SNGL display modes and re-issue the command. You can press the PFKey assigned to HELP (PF1 by default) to display additional help for this error.

FMNDC203 No labels (short). Labels cannot be specified when in SNGL mode. Press help for more information. (long).

Explanation: The user entered the EXPORT editor primary command, in SNGL mode, with a label range. Labels may only be specified for the EXPORT command in TABL display mode. The export command cannot be performed.

User response: Return to TABL display mode and re-issue the command. You can press the PFKey assigned to HELP (PF1 by default) to display additional help for this error.
| FMNDC204 | ALL | NX | X | EX missing (short). Labels cannot be specified unless ALL | NX | X | EX is also specified. Press help for more information. (long). |
|-----------|-----|-----|-----|-----------------------------------------------|

**Explanation:** The user entered the EXPORT editor primary command, with a label range, but without either the ALL or NX | X | EX parameters. When you specify a label range one of the ALL, NX | X | EX options is required. The export command cannot be performed.

**User response:** Press the PFKey assigned to HELP (PF1 by default) and review the following topics: "Syntax diagram". ALL | NX | X | EX parameter Modify the command to meet the requirements of the syntax diagram, and the intended usage of the EXPORT command, re-issue the command.

<table>
<thead>
<tr>
<th>FMNDC206</th>
<th>Export failed (short). The export operation failed. (long).</th>
</tr>
</thead>
</table>

**Explanation:** The user issued the EXPORT editor primary command in an FM/DB2 editor session. The command failed.

**User response:** Check the output data set for allocation/space and related errors.

<table>
<thead>
<tr>
<th>FMNDC207</th>
<th>LRECL mismatch (short). The (RECFM=F) export data set has LRECL=LRECL 1. The required LRECL is LRECL 2. Specify a different export data set. (long).</th>
</tr>
</thead>
</table>

**Explanation:** The user attempted an export operation and specified an LRECL=F data set that is incompatible with the exported data. The export operation cannot be performed.

**User response:** Use an export data set with LRECL = LRECL 2.

<table>
<thead>
<tr>
<th>FMNDC208</th>
<th>Inconsistent selection (short). The template usage option should only be set to None when the export data format is set to CSV. Select another template usage option. (long).</th>
</tr>
</thead>
</table>

**Explanation:** This message is issued by the FM/DB2 export function, on the "To" panel. The user entered '5' (CSV) for the "Template usage" option. However, the export data format option set on the first export options panel is not CSV. The export cannot proceed.

**User response:** The choice for export data format, on the first export options panel, places restrictions on the template usage option that can be specified on the export 'To' panel. The following list shows which 'template usage' options can be specified for each export data format:

<table>
<thead>
<tr>
<th>Export data format</th>
<th>Template usage option</th>
</tr>
</thead>
<tbody>
<tr>
<td>FM/DB2 (SQLDA) format</td>
<td>All except 5 (CSV)</td>
</tr>
<tr>
<td>DB2 UNLOAD format</td>
<td>All except 5 (CSV)</td>
</tr>
<tr>
<td>DSN/TIAU format</td>
<td>All except 5 (CSV)</td>
</tr>
<tr>
<td>User defined</td>
<td>All except 5 (CSV)</td>
</tr>
<tr>
<td>Delimited variables (CSV)</td>
<td>5 (CSV) only.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>FMNDC209</th>
<th>LRECL mismatch (short). The (RECFM=V) export data set has LRECL=LRECL 1. The required LRECL is at least LRECL 2. Specify a different export data set.</th>
</tr>
</thead>
</table>

**Explanation:** The user attempted an export operation and specified an LRECL=F data set that is incompatible with the exported data. The export operation cannot be performed.

**User response:** Use an export data set with LRECL = LRECL 2.

<table>
<thead>
<tr>
<th>FMNDC211</th>
<th>Result table for:</th>
</tr>
</thead>
</table>

**Explanation:** This is a message inserted used by the FM/DB2 export function.

**User response:** None. Message insert used for panel text.

| FMNDC220 | NX | X | EX invalid (short). The NX | X | EX parameters cannot be specified when in browse mode. Press help for more information. (long). |
|-----------|-----|-----|-----|-----------------------------------------------------------------|

**Explanation:** The user entered the EXPORT editor primary command in a browse session, the EXPORT command included a reference to excluded, or not-excluded rows. This is not supported in an FM/DB2 browse session.

**User response:** Modify the EXPORT command to remove any reference to excluded, or not-excluded rows, retry the operation. If the intended usage requires the use of excluded or not-excluded rows as part of the EXPORT command, repeat the process using an FM/DB2 view session.

<table>
<thead>
<tr>
<th>FMNDC221</th>
<th>No labels (short). Labels cannot be specified when in browse mode. Press help for more information. (long).</th>
</tr>
</thead>
</table>

**Explanation:** The user entered the EXPORT editor primary command in a browse session, the EXPORT command included label references. There is no prefix area in an FM/DB2 browse session, so labels cannot be entered, nor can they be referred to.

---

**Appendix. FM/DB2 messages**
FMNDC230  •  FMNDC536

User response: Modify the EXPORT command to remove any reference to labels, retry the operation.

FMNDC230 Warning - fixed RECFM (short). The output data set has a fixed record format, which is not recommended when using CSV format. Specify a data set with RECFM=VB or RECFM=V.

Explanation: This message is issued by the FM/DB2 Export utility, when CSV format is selected. The attributes of the output data set are inconsistent with CSV format, which produces variable length output records.

User response: In general RECFM=F and RECFM=FB should not be used with CSV format. You can specify the name of a new data set on the export 'To' panel and FM/DB2 will calculate the correct LRECL for you.

FMNDC500 Executed OK (short). The ALTER|COMMENT|CREATE|DROP|GRANT|LABEL|REVOKE statement executed successfully. DB2 returned an SQLCODE of SQLCODE. (long).

Explanation: This message is issued by the FM/DB2 Object List and Object Privileges utilities. An SQL statement was executed successfully.

User response: No action required. Informational message.

FMNDC510 No rows returned (short). No rows were fetched, SQL code = +100. (long).

Explanation: This message is issued by the FM/DB2 Object List, Object Privileges and Explain utilities functions. A SELECT statement was executed, but returned no rows (SQLCODE=+100).

User response: No action required. Informational message.

FMNDC520 No allocation data (short). The P QTY/SQTY values in SYSIBM.SYSTABLEPART indicate that RUNSTATS has not been run. The primary and/or secondary space allocation data for the table space is unavailable. (long).

Explanation: This message is issued by the Alter table space and Alter index panels, accessed from the FM/DB2 Object list utility. FM/DB2 accessed the DB2 catalog to determine the current space allocation values for the table space or index, and then repeating the ALTER operation.

User response: No action required. Informational message.

FMNDC521 No allocation data (short). The P QTY/SQTY values in SYSIBM.SYNSINDEXPART indicate that RUNSTATS has not been run. The primary and secondary space allocation data for the index space is unavailable. (long).

Explanation: This message is issued by the Alter index panel, accessed from the FM/DB2 Object list utility. FM/DB2 accessed the DB2 catalog to determine the current space allocation values for the index. However, this information is not available because the RUNSTATS utility has not been executed. You can resolve this problem by executing the RUNSTATS utility against the appropriate index, and then repeating the ALTER operation.

User response: No action required. Informational message.

FMNDC531 INTEGER datatype, DEC(nn,0)

Explanation: This is a translatable message displayed by the FM/DB2 Create sequence function.

User response: No action required. Panel text.

FMNDC533 3. UNICODE

Explanation: This is a translatable message displayed by the FM/DB2 Object list function.

User response: No action required. Panel text.

FMNDC534 3. Temporary database

Explanation: This is a translatable message displayed by the FM/DB2 Create database function.

User response: No action required. Panel text.

FMNDC535 Log changes to LOB columns

Explanation: This is a translatable message displayed by the FM/DB2 Create table space function.

User response: No action required. Panel text.

FMNDC536 Log changes to data

Explanation: This is a translatable message displayed by the FM/DB2 Create table space function.

User response: No action required. Panel text.
FMNDC539 Instead of
Explanation: This is a translatable message displayed by the FM/DB2 Create trigger function.
User response: No action required. Panel text.

FMNDC540 1. Database
Explanation: This is a translatable message displayed by the FM/DB2 Object Privileges function.
User response: No action required. Panel text.

FMNDC541 2. Table space
Explanation: This is a translatable message displayed by the FM/DB2 Object Privileges function.
User response: No action required. Panel text.

FMNDC542 3. Table
Explanation: This is a translatable message displayed by the FM/DB2 Object Privileges function.
User response: No action required. Panel text.

FMNDC543 4. Column
Explanation: This is a translatable message displayed by the FM/DB2 Object Privileges function.
User response: No action required. Panel text.

FMNDC544 5. Plan
Explanation: This is a translatable message displayed by the FM/DB2 Object Privileges function.
User response: No action required. Panel text.

FMNDC545 6. Package
Explanation: This is a translatable message displayed by the FM/DB2 Object Privileges function.
User response: No action required. Panel text.

FMNDC546 7. Collection
Explanation: This is a translatable message displayed by the FM/DB2 Object Privileges function.
User response: No action required. Panel text.

FMNDC547 8. Schema
Explanation: This is a translatable message displayed by the FM/DB2 Object Privileges function.
User response: No action required. Panel text.

FMNDC548 9. Distinct type
Explanation: This is a translatable message displayed by the FM/DB2 Object Privileges function.
User response: No action required. Panel text.

FMNDC549 9. Type
Explanation: This is a translatable message displayed by the FM/DB2 Object Privileges function.
User response: No action required. Panel text.

Explanation: This is a translatable message displayed by the FM/DB2 Object list function.
User response: No action required. Panel text.

FMNDC551 2. Table space 9. Collection 16. Storage group
Explanation: This is a translatable message displayed by the FM/DB2 Object list function.
User response: No action required. Panel text.

FMNDC552 3. Table/view/alias 10. DBRM
Explanation: This is a translatable message displayed by the FM/DB2 Object list function.
User response: No action required. Panel text.

FMNDC553 4. Index 11. Schema
Explanation: This is a translatable message displayed by the FM/DB2 Object list function.
User response: No action required. Panel text.

FMNDC554 5. Column 12. Distinct type
Explanation: This is a translatable message displayed by the FM/DB2 Object list function.
User response: No action required. Panel text.

Explanation: This is a translatable message displayed by the FM/DB2 Object list function.
User response: No action required. Panel text.

Explanation: This is a translatable message displayed by the FM/DB2 Object list function.
User response: No action required. Panel text.
FMNDC557 3. Table/view/alias 10. DBRM 17. Sequence
Explanation: This is a translatable message displayed by the FM/DB2 Object list function.
User response: No action required. Panel text.

Explanation: This is a translatable message displayed by the FM/DB2 Object list function.
User response: No action required. Panel text.

Explanation: This is a translatable message displayed by the FM/DB2 Object list function.
User response: No action required. Panel text.

FMNDC570 Database...
Explanation: This is a translatable message displayed by the FM/DB2 Object list function.
User response: No action required. Panel text.

FMNDC571 Table Space...
Explanation: This is a translatable message displayed by the FM/DB2 Create table function.
User response: No action required. Panel text.

FMNDC572 Usage:
Explanation: This is a translatable message displayed by the FM/DB2 Create table function.
User response: No action required. Panel text.

FMNDC573 1. Generate LIKE clause
Explanation: This is a translatable message displayed by the FM/DB2 Create table function.
User response: No action required. Panel text.

FMNDC574 2. Load table information
Explanation: This is a translatable message displayed by the FM/DB2 Create table function.
User response: No action required. Panel text.

FMNDC578 Create Global Temporary Table
Explanation: This is a translatable message displayed by the FM/DB2 object functions.
User response: No action required. Panel text.

FMNDC579 Create Table
Explanation: This is a translatable message displayed by the FM/DB2 object functions.
User response: No action required. Panel text.

FMNDC580 Must be blank (short). Specify one value or the other, but not both. (long).
Explanation: This message is issued by the FM/DB2 Privileges function. The user selected both options when the two options are mutually exclusive.
User response: Select one option, or the other, but not both. Consult the SQL Reference manual for the relevant syntax diagram for the command.

FMNDC581 Cannot drop directly (short). An auxiliary table cannot be dropped directly. To drop an auxiliary table drop the table that uses the auxiliary table. (long).
Explanation: This message is issued by the DB2 Object functions utility. The user specified the combination of 'D' (Drop) and an object type of 12 (Auxiliary table). This combination is not supported.
User response: You cannot directly drop an auxiliary table. Auxiliary tables should be dropped by a DROP of the table of which the auxiliary table is a part. See the DB2 SQL Reference for a full explanation of the use of auxiliary tables.

FMNDC583 Reserved (short). Sequence number 1 is reserved for the SEL field. Specify a larger value. (long).
Explanation: This message is issued on various Revoke Privilege panels displayed by the FM/DB2 Privileges function. When REVOKEing a privilege with SYSADM authority you may REVOKE the privilege from ALL users, or list 1-3 individual DB2 authids.

User response: Specify the required ordering starting at '2', instead of '1'.

FMNDC584 Must be blank (short). Specify the ALL parameter, or one or more values, but not both. (long).
Explanation: This message is issued by the "Column Selection/Edit" panel, when displayed from the FM/DB2 Object List, Object Privileges and Explain functions (Collectively ISQ functions). The object displays for ISQ functions use a modified version of the FM/DB2 editor to display the data, and allow the same template editor functions as a normal FM/DB2 editor session; with some restrictions. While you can specify a sequence number in the 'Seq' field, a value of 1 is reserved for the 'Sel' field, which appears on most ISQ panels.
User response: Specify the required ordering starting at '2', instead of '1'.
These options are mutually exclusive, you cannot select ALL and also enter an individual DB2 authid.

**User response:** Select ALL if the privilege is to be REVOKed from all DB2 authids. Alternatively enter one or more individual DB2 authids.

**FMNDC585** Must be selected (short). The PUBLIC selection field must be selected when the AT ALL LOCATIONS selection field is selected. (long).

**Explanation:** This message is issued by the Grant table and Grant column privileges' panels. The user selected the 'AT ALL LOCATIONS' field, but not the 'PUBLIC' field. Selecting the former requires that the latter also be selected.

**User response:** Select both the 'PUBLIC' and 'AT ALL LOCATIONS' fields.

**FMNDC586** Not with ROLE (short). The ROLE and PUBLIC options cannot both be selected. PUBLIC is a special DB2 authid, not a valid role name. (long).

**Explanation:** This message is issued by the DB2 Privileges utility when the user selects both the 'ROLE' and 'PUBLIC' fields. The utility does not support the simultaneous use of both the PUBLIC and ROLE options.

**User response:** To GRANT/REVOKE the privileges to PUBLIC, clear the ROLE field and process the request. Similarly, to GRANT/REVOKE the privileges to a ROLE, clear the PUBLIC field and process the request.

**FMNDC587** Not with ROLE (short). The ROLE and ALL options cannot both be selected. ALL can only be specified with DB2 authids. (long).

**Explanation:** This message is issued by the DB2 Privileges utility when the user selects both the 'ROLE' and 'ALL' fields. The utility does not support the simultaneous use of both the PUBLIC and ALL options.

**User response:** To REVOKE the privileges from ALL DB2 authids that possess the privilege, clear the ROLE field and process the request. Similarly, to REVOKE the privileges from a ROLE, clear the ALL field and process the request.

**FMNDC588** Not AT ALL LOCATIONS (short). The privilege type privilege cannot be granted to PUBLIC AT ALL LOCATIONS. (long).

**Explanation:** This message is issued by the DB2 Privileges utility, when granting table or column privileges. The 'AT ALL LOCATIONS' may not be used.

**User response:** Deselect the "AT ALL LOCATIONS" option and process the request.

---

**FMNDC590** 1. Database

**Explanation:** This is a translatable message displayed by the FM/DB2 object functions.

**User response:** No action required. Panel text.

**FMNDC591** 2. Table space

**Explanation:** This is a translatable message displayed by the FM/DB2 object functions.

**User response:** No action required. Panel text.

**FMNDC592** 3. Table

**Explanation:** This is a translatable message displayed by the FM/DB2 object functions.

**User response:** No action required. Panel text.

**FMNDC593** 4. View

**Explanation:** This is a translatable message displayed by the FM/DB2 object functions.

**User response:** No action required. Panel text.

**FMNDC594** 5. Alias

**Explanation:** This is a translatable message displayed by the FM/DB2 object functions.

**User response:** No action required. Panel text.

**FMNDC595** 6. Index

**Explanation:** This is a translatable message displayed by the FM/DB2 object functions.

**User response:** No action required. Panel text.

**FMNDC596** 7. Synonym

**Explanation:** This is a translatable message displayed by the FM/DB2 object functions.

**User response:** No action required. Panel text.

**FMNDC597** 8. Distinct type

**Explanation:** This is a translatable message displayed by the FM/DB2 object functions.

**User response:** No action required. Panel text.
<table>
<thead>
<tr>
<th>Type</th>
<th>Explanation</th>
<th>User response</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>This is a translatable message displayed by the FM/DB2 object functions.</td>
<td>No action required. Panel text.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Function</th>
<th>Explanation</th>
<th>User response</th>
</tr>
</thead>
<tbody>
<tr>
<td>Function</td>
<td>This is a translatable message displayed by the FM/DB2 object functions.</td>
<td>No action required. Panel text.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Database</th>
<th>Explanation</th>
<th>User response</th>
</tr>
</thead>
<tbody>
<tr>
<td>Database</td>
<td>This is translatable message text used in various places.</td>
<td>No action required. Translatable text.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table space</th>
<th>Explanation</th>
<th>User response</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table space</td>
<td>This is translatable message text used in various places.</td>
<td>No action required. Translatable text.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table</th>
<th>Explanation</th>
<th>User response</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table</td>
<td>This is translatable message text used in various places.</td>
<td>No action required. Translatable text.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>View</th>
<th>Explanation</th>
<th>User response</th>
</tr>
</thead>
<tbody>
<tr>
<td>View</td>
<td>This is translatable message text used in various places.</td>
<td>No action required. Translatable text.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Alias</th>
<th>Explanation</th>
<th>User response</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alias</td>
<td>This is translatable message text used in various places.</td>
<td>No action required. Translatable text.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Index</th>
<th>Explanation</th>
<th>User response</th>
</tr>
</thead>
<tbody>
<tr>
<td>Index</td>
<td>This is translatable message text used in various places.</td>
<td>No action required. Translatable text.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Synonym</th>
<th>Explanation</th>
<th>User response</th>
</tr>
</thead>
<tbody>
<tr>
<td>Synonym</td>
<td>This is translatable message text used in various places.</td>
<td>No action required. Translatable text.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Distinct type</th>
<th>Explanation</th>
<th>User response</th>
</tr>
</thead>
<tbody>
<tr>
<td>Distinct type</td>
<td>This is translatable message text used in various places.</td>
<td>No action required. Translatable text.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Function</th>
<th>Explanation</th>
<th>User response</th>
</tr>
</thead>
<tbody>
<tr>
<td>Function</td>
<td>This is translatable message text used in various places.</td>
<td>No action required. Translatable text.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Stored procedure</th>
<th>Explanation</th>
<th>User response</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stored procedure</td>
<td>This is translatable message text used in various places.</td>
<td>No action required. Translatable text.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Trigger</th>
<th>Explanation</th>
<th>User response</th>
</tr>
</thead>
<tbody>
<tr>
<td>Trigger</td>
<td>This is translatable message text used in various places.</td>
<td>No action required. Translatable text.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Storage Group</th>
<th>Explanation</th>
<th>User response</th>
</tr>
</thead>
<tbody>
<tr>
<td>Storage Group</td>
<td>This is translatable message text used in various places.</td>
<td>No action required. Translatable text.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Global Temporary Table</th>
<th>Explanation</th>
<th>User response</th>
</tr>
</thead>
<tbody>
<tr>
<td>Global Temporary Table</td>
<td>This is translatable message text used in various places.</td>
<td>No action required. Translatable text.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Sequence</th>
<th>Explanation</th>
<th>User response</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sequence</td>
<td>This is translatable message text used in various places.</td>
<td>No action required. Translatable text.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Role</th>
<th>Explanation</th>
<th>User response</th>
</tr>
</thead>
<tbody>
<tr>
<td>Role</td>
<td>This is translatable message text used in various places.</td>
<td>No action required. Translatable text.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Trusted Context</th>
<th>Explanation</th>
<th>User response</th>
</tr>
</thead>
<tbody>
<tr>
<td>Trusted Context</td>
<td>This is translatable message text used in various places.</td>
<td>No action required. Translatable text.</td>
</tr>
</tbody>
</table>
FMNDC616  Type
Explanation:   This is translatable message text used in various places.
User response: No action required. Translatable text.

FMNDC620  Columns
Explanation:   This is translatable message text used in various places.
User response: No action required. Translatable text.

FMNDC621  Nulls/default values
Explanation:   This is translatable message text used in various places.
User response: No action required. Translatable text.

FMNDC622  CCSID/for data
Explanation:   This is translatable message text used in various places.
User response: No action required. Translatable text.

FMNDC623  Table options
Explanation:   This is translatable message text used in various places.
User response: No action required. Translatable text.

FMNDC624  Unique constraints
Explanation:   This is translatable message text used in various places.
User response: No action required. Translatable text.

FMNDC625  Referential constraints
Explanation:   This is translatable message text used in various places.
User response: No action required. Translatable text.

FMNDC626  Check constraints
Explanation:   This is translatable message text used in various places.
User response: No action required. Translatable text.

FMNDC627  Procedure exits
Explanation:   This is translatable message text used in various places.
User response: No action required. Translatable text.

FMNDC628  Generate values
Explanation:   This is translatable message text used in various places.
User response: No action required. Translatable text.

FMNDC629  Partitioning columns
Explanation:   This is translatable message text used in various places.
User response: No action required. Translatable text.

FMNDC630  Partitioning values
Explanation:   This is translatable message text used in various places.
User response: No action required. Translatable text.

FMNDC631  Create table
Explanation:   This is translatable message text used in various places.
User response: No action required. Translatable text.

FMNDC632  Char
Explanation:   This is translatable message text used in various places.
User response: No action required. Translatable text.

FMNDC633  Integer
Explanation:   This is translatable message text used in various places.
User response: No action required. Translatable text.

FMNDC634  Edit options
Explanation:   This is translatable message text used in various places.
User response: No action required. Translatable text.

FMNDC635  View options
Explanation:   This is translatable message text used in various places.
User response: No action required. Translatable text.

FMNDC636  To Sequential Data Set:
Explanation:   This is translatable message text, displayed by the FM/DB2 Export utility.
User response: No action required. Panel text.
FMNDC637  To Partitioned, Sequential or VSAM Data Set:
Explanation:  This is translatable message text, displayed by the FM/DB2 Export utility.
User response:  No action required. Panel text.

FMNDC638  Member . . . . .
Explanation:  This is translatable message text used in various places.
User response:  No action required. Translatable text.

FMNDC640  (Use option 1 before options 2, 4-9)
Explanation:  This is translatable message text used by the FM/DB2 Object functions utility.
User response:  No action required. Panel text.

FMNDC641  (Use option 1 before options 2,4-9,PC,PV)
Explanation:  This is translatable message text used by the FM/DB2 Object functions utility.
User response:  No action required. Panel text.

FMNDC642  (normally required, invalid with model LIKE)
Explanation:  This is translatable message text used by the FM/DB2 Object functions utility.
User response:  No action required. Panel text.

FMNDC643  (editproc,validproc,more; optional)
Explanation:  This is translatable message text used by the FM/DB2 Object functions utility.
User response:  No action required. Panel text.

FMNDC644  (primary/unique keys; optional)
Explanation:  This is translatable message text used by the FM/DB2 Object functions utility.
User response:  No action required. Panel text.

FMNDC645  (foreign keys; optional)
Explanation:  This is translatable message text used by the FM/DB2 Object functions utility.
User response:  No action required. Panel text.

FMNDC646  (optional)
Explanation:  This is translatable message text used by the FM/DB2 Object functions utility.
User response:  No action required. Panel text.

FMNDC647  (Use option 1 before options 2, 4)
Explanation:  This is translatable message text used by the FM/DB2 Object functions utility.
User response:  No action required. Panel text.

FMNDC648  (normally required, invalid with model)
Explanation:  This is translatable message text used by the FM/DB2 Object functions utility.
User response:  No action required. Panel text.

FMNDC649  (CCSID; optional)
Explanation:  This is translatable message text used by the FM/DB2 Object functions utility.
User response:  No action required. Panel text.

FMNDC651  Invalid data type (short) ROWID, CLOB, BLOB and DBCLOB columns cannot be specified in a global temporary table definition. (long).
Explanation:  This message is issued by the FM/DB2 Object functions utility, when attempting to create a global temporary table (GTT). One or more of the columns defined for the GTT has a prohibited data type (ROWID, CLOB, BLOB or DBCLOB). The GTT cannot be created.
User response:  Correct the definition for any columns defined as ROWID, CLOB, BLOB or DBCLOB, retry the operation.

FMNDC660  Column information
Explanation:  This is translatable message text used by the FM/DB2 Object functions utility.
User response:  No action required. Panel text.

FMNDC661  Primary key information
Explanation:  This is translatable message text used by the FM/DB2 Object functions utility.
User response:  No action required. Panel text.

FMNDC662  Unique key information
Explanation:  This is translatable message text used by the FM/DB2 Object functions utility.
User response:  No action required. Panel text.
FMNDC663  Referential constraints
Explanation: This is translatable message text used by the FM/DB2 Object functions utility.
User response: No action required. Panel text.

FMNDC664  Check constraints
Explanation: This is translatable message text used by the FM/DB2 Object functions utility.
User response: No action required. Panel text.

FMNDC665  Column referential constraints
Explanation: This is translatable message text used by the FM/DB2 Object functions utility.
User response: No action required. Panel text.

FMNDC666  Table referential constraints
Explanation: This is translatable message text used by the FM/DB2 Object functions utility.
User response: No action required. Panel text.

FMNDC667  Load single-column referential constraints as
Explanation: This is translatable message text used by the FM/DB2 Object functions utility.
User response: No action required. Panel text.

FMNDC668  Default Value
Explanation: This is translatable message text used by the FM/DB2 Object functions utility.
User response: No action required. Panel text.

FMNDC670  DB2 Utility LISTDEF options
Explanation: This is translatable message text displayed on the FM/DB2 Set Processing Options panel.
User response: No action required. Panel text.

FMNDC671  DB2 Utility OPTIONS options
Explanation: This is translatable message text displayed on the FM/DB2 Set Processing Options panel.
User response: No action required. Panel text.

FMNDC672  DB2 Utility TEMPLATE options
Explanation: This is translatable message text displayed on the FM/DB2 Set Processing Options panel.
User response: No action required. Panel text.

FMNDC673  DB2 Unload utility options
Explanation: This is translatable message text displayed on the FM/DB2 Set Processing Options panel.
User response: No action required. Panel text.

FMNDC680  3. CSV, use NI char
Explanation: This is translatable message text displayed on the FM/DB2 Export Options panel.
User response: No action required. Panel text.

FMNDC690  Compress . . . . .
Explanation: This is translatable message text used by the FM/DB2 Object functions utility.
User response: No action required. Panel text.

FMNDC691  2. Table space 10. Stored procedure
Explanation: This is translatable message text used by the FM/DB2 Object functions utility.
User response: No action required. Panel text.

FMNDC700  1. AUDIT 7. DROP FOREIGN KEY constraint
Explanation: This is translatable message text used by the FM/DB2 Object functions utility.
User response: No action required. Panel text.

FMNDC701  2. ADD column 8. DATA CAPTURE
Explanation: This is translatable message text used by the FM/DB2 Object functions utility.
User response: No action required. Panel text.

FMNDC702  3. VALIDPROC 9. ADD CHECK constraint
Explanation: This is translatable message text used by the FM/DB2 Object functions utility.
User response: No action required. Panel text.

FMNDC703  4. ADD PRIMARY KEY 10. DROP CHECK constraint
Explanation: This is translatable message text used by the FM/DB2 Object functions utility.
User response: No action required. Panel text.
FMNDC704 5. ADD FOREIGN KEY constraint 11.
ADD RESTRICT ON DROP
Explanation: This is translatable message text used by
the FM/DB2 Object functions utility.
User response: No action required. Panel text.

FMNDC705 6. DROP PRIMARY KEY 12. DROP
RESTRICT ON DROP
Explanation: This is translatable message text used by
the FM/DB2 Object functions utility.
User response: No action required. Panel text.

FMNDC710 1. AUDIT 8. DATA CAPTURE
Explanation: This is translatable message text used by
the FM/DB2 Object functions utility.
User response: No action required. Panel text.

FMNDC711 2. ADD column 9. ADD CHECK
constraint
Explanation: This is translatable message text used by
the FM/DB2 Object functions utility.
User response: No action required. Panel text.

FMNDC712 3. VALIDPROC 10. DROP CHECK
constraint
Explanation: This is translatable message text used by
the FM/DB2 Object functions utility.
User response: No action required. Panel text.

FMNDC713 4. ADD PRIMARY KEY 11. ADD
RESTRICT ON DROP
Explanation: This is translatable message text used by
the FM/DB2 Object functions utility.
User response: No action required. Panel text.

FMNDC714 5. ADD FOREIGN KEY constraint 12.
DROP RESTRICT ON DROP
Explanation: This is translatable message text used by
the FM/DB2 Object functions utility.
User response: No action required. Panel text.

FMNDC715 6. DROP PRIMARY KEY 13. RENAME
COLUMN
Explanation: This is translatable message text used by
the FM/DB2 Object functions utility.
User response: No action required. Panel text.

FMNDC716 7. DROP FOREIGN KEY constraint:
Explanation: This is translatable message text used by
the FM/DB2 Object functions utility.
User response: No action required. Panel text.

FMNDC720 C. ROUND_CEILING
Explanation: This is translatable message text
displayed by the FM/DB2 System options function.
User response: No action required. Panel text.

FMNDC721 3 D. ROUND_DOWN
Explanation: This is translatable message text
displayed by the FM/DB2 System options function.
User response: No action required. Panel text.

FMNDC722 E. ROUND_HALF_EVEN
Explanation: This is translatable message text
displayed by the FM/DB2 System options function.
User response: No action required. Panel text.

FMNDC723 G. ROUND_HALF_DOWN
Explanation: This is translatable message text
displayed by the FM/DB2 System options function.
User response: No action required. Panel text.

FMNDC724 H. ROUND_HALF_UP
Explanation: This is translatable message text
displayed by the FM/DB2 System options function.
User response: No action required. Panel text.

FMNDC725 U. ROUND_UP
Explanation: This is translatable message text
displayed by the FM/DB2 System options function.
User response: No action required. Panel text.

FMNDC730 (optional, valid for FLOAT and
DECIMAL only)
Explanation: This is translatable message text used by
the FM/DB2 Object functions utility.
User response: No action required. Panel text.
FMNDC731  (optional, valid for FLOAT, DEC and DECFLOAT only)
Explanation:  This is translatable message text used by the FM/DB2 Object functions utility.
User response:  No action required. Panel text.

FMNDC732  Selective Partition Locking
Explanation:  This is translatable message text used by the FM/DB2 Object functions utility.
User response:  No action required. Panel text.

FMNDC733  Use selective partition locking
Explanation:  This is translatable message text used by the FM/DB2 Object functions utility.
User response:  No action required. Panel text.

FMNDC740  Maximum number of partitions
Explanation:  This is translatable message text used by the FM/DB2 Object functions utility.
User response:  No action required. Panel text.

FMNDD000  Command list error (short). An error was encountered processing the list of valid commands for panel panel_name.
Explanation:  This message is issued by the FM/DB2 Interactive SQL (ISQ) processor (used by the following options: 3.4; 3.5; 4.5). The line commands that may be entered against a list of objects are defined using a customization string. An error occurred during the processing of this string for panel panel_name.
User response:  This is an internal error. Contact IBM support.

FMNDD001  Customization error (short). An error was encountered processing the template customization string for panel panel_name. (long).
Explanation:  This message is issued by the FM/DB2 Interactive SQL (ISQ) processor (used by the following options: 3.4; 3.5; 4.5). Object lists are displayed using the FM/DB2 editor, the template for the object list is modified prior to display. The template customization is defined using a customization string. An error occurred during the processing of this string for panel panel_name.
User response:  This is an internal error. Contact IBM support.

FMNDD002  No line commands (short). There are no line commands for this panel. (long).
Explanation:  This message is issued by the FM/DB2 Interactive SQL (ISQ) processor (used by the following options: 3.4; 3.5; 4.5). In most cases you may enter one or more line commands against each row shown in the object list. In some few cases no line commands are available, in which case this message is issued.
User response:  No action required. Informational message.

FMNDD003  VDEFINE error, RC=Return code short).
A serious ISPF error has been encountered. VDEFINE RC=Return code. (long)
Explanation:  FM/DB2 attempted to use the ISPF VDEFINE service, the call failed with RC = Return code.
In most cases the current function will terminate.
User response:  The causes of this error are:
• Problems with the ISPF environment that have caused the ISPF service call to fail. Contact your Systems Programmer for assistance.
• An internal error. Contact IBM support.
Points for the Systems Programmer to consider include:
• ISPF libraries allocated incorrectly or incompletely.
• Full ISPF profile data sets.
• Out of memory, or short on memory, conditions. Are other ISPF applications running? Are multiple ISPF logical sessions active?

FMNDD004  ISPF error, RC=Return code (short). A serious ISPF error has been encountered in FMN2PSU. ISPF SELECT SERVICE RC=Return code (long).
Explanation:  This message is issued by the FM/DB2 Interactive SQL (ISQ) processor (used by the following options: 3.4; 3.5; 4.5). A serious error was encountered in response to an ISPF SELECT service call (return code = Return code) In most cases the function will terminate.
User response:  The usual causes of this error are:
• Problems with the ISPF environment that have caused the ISPF service call to fail. Contact your Systems Programmer for assistance.
• An internal error. Contact IBM support.
Points for the System Programmer to consider include:
• ISPF libraries allocated incorrectly or incompletely.
• Full ISPF profile data sets.
• Out of memory, or short on memory, conditions. Are other ISPF applications running? Are multiple ISPF logical sessions active?
FMNDD005  ISPF error, RC=Return code (short). A serious ISPF error has been encountered in FMN2IPSU. ISPF CONTROL SERVICE RC=Return code (long).

Explanation: This message is issued by the FM/DB2 Interactive SQL (ISQ) processor (used by the following options: 3.4; 3.5; 4.5). A serious error was encountered in response to an ISPF CONTROL service call (return code = Return code). In most cases the function will terminate.

User response: The usual causes of this error are:
- Problems with the ISPF environment that have caused the ISPF service call to fail. Contact your Systems Programmer for assistance.
- An internal error. Contact IBM support.

Points for the Systems Programmer to consider include:
- ISPF libraries allocated incorrectly or incompletely.
- Full ISPF profile data sets.
- Out of memory, or short on memory, conditions. Are other ISPF applications running? Are multiple ISPF logical sessions active?

FMNDD006 VDELETE error, RC=Return code (short). A serious ISPF error has been encountered. VDELETE RC=Return code (long).

Explanation: This message is issued by the FM/DB2 Interactive SQL (ISQ) processor (used by the following options: 3.4; 3.5; 4.5). A serious error was encountered in response to an ISPF DELETE service call (return code = Return code). In most cases the function will terminate.

User response: The usual causes of this error are:
- Problems with the ISPF environment that have caused the ISPF service call to fail. Contact your Systems Programmer for assistance.
- An internal error. Contact IBM support.

Points for the Systems Programmer to consider include:
- ISPF libraries allocated incorrectly or incompletely.
- Full ISPF profile data sets.
- Out of memory, or short on memory, conditions. Are other ISPF applications running? Are multiple ISPF logical sessions active?

FMNDD007 No dynamic area (short). A serious error has been encountered. Panel Panel name does not contain a dynamic area and therefore cannot be used to display ISQ editor output. (long).

Explanation: This message is issued by the FM/DB2 Interactive SQL (ISQ) processor (used by the following options: 3.4; 3.5; 4.5). A serious error was encountered attempting to display an ISPF panel.

User response: This is an internal error. Contact IBM support.

FMNDD008 Command not available (short). The SQL command is not supported under CICS. (long).

Explanation: This message is issued when FM/DB2 is running under CICS, and the user issues the SQL command. The SQL command is not available when running under CICS.

User response: No action required. Informational message.

FMNDD009 VREP error, RC=RC (short). A serious ISPF error has been encountered.

VREPLACE RC=RC. (long).

Explanation: This message is issued by the Basic SQL prototyper. In indicates that a serious ISPF error was encountered.

User response: The usual causes of this error are:
- Problems with the ISPF environment that have caused the ISPF service call to fail. Contact your Systems Programmer for assistance.
- An internal error. Contact IBM support.

Points for the Systems Programmer to consider include:
- ISPF libraries allocated incorrectly or incompletely.
- Full ISPF profile data sets.
- Out of memory, or short on memory, conditions. Are other ISPF applications running? Are multiple ISPF logical sessions active?

FMNDD010 Invalid command Command (short). "Command" is not a valid File Manager DB2 command. List of valid commands. (long).

Explanation: This message is issued by the FM/DB2 Interactive SQL (ISQ) processor (used by the following options: 3.4; 3.5; 4.5). A command (for example XXXX) was issued on the TSO command line. However, this command is not recognized by the ISQ processor and cannot be run.

User response: No action required. Informational message.

FMNDD011 Too many parms (short). "Command"> is followed by one or more parameters. It is only valid without parameters. PARMS=Parameters. (long).

Explanation: This message is issued by the FM/DB2 Interactive SQL (ISQ) processor (used by the following options: 3.4; 3.5; 4.5). A command Command, with parameters, was entered on the TSO command line. Command does not require parameters.
User response: Remove the extraneous parameters, re-issue the command.

FMNDD012 No SQL to display (short). There is no SQL statement to display. (long).

Explanation: The user entered the SQL command; however there is no current SQL statement to display.

User response: No action required. Informational message.

FMNDD018 Object not found (short). No DB2 object at location Location matches the combination of object details entered at DB2 system SSID. Press Help for more information. (long).

Explanation: The user entered a DB2 object name, including location, owner, and name. This object does not exist. You can press PF1 to see additional help for this error.

User response: Press PF1 to see a more detailed explanation of the error. The usual reasons for this error include:
• The nominated object does not exist. Is it spelled correctly?
• When the object name includes lower case letters you must ensure that automatic translation of object names is turned off. See the FM/DB2 systems options.
• When the automatic translation of object names is turned off, you must ensure that the object name entered matches the case of the actual object name. For example, DSN81010.EMP matches DSN81010.EMP, but dsn81010.EMP does not.

FMNDD020 Invalid selection (short). The value entered is invalid. Type '?' in the field to display a list of valid values. (long).

Explanation: This message is issued by the FM/DB2 Interactive SQL (ISQ) processor (used by the following options: 3.4; 3.5; 4.5). The user entered a line command against a row. However, the line command is not valid for that type of object.

User response: Check for obvious mistakes such as typos. Type '?' in the field and press Enter to show the list of valid commands for the object.

FMNDD022 No SQL to display (short). There is no SQL statement to display. (long).

Explanation: The user entered the SQL command; however there is no current SQL statement to display.

User response: No action required. Informational message.

FMNDD028 Requires DB2 V9 (short). This command can only be issued when connected to a DB2 version 9 (or later) system. (long).

Explanation: This message is issued by the FM/DB2 Interactive SQL (ISQ) processor (used by the following options: 3.4; 3.5; 4.5). When connected to a DB2 version 8 system, the user entered a line command against a row; however the line command is only valid when connected to a DB2 version 9 (or later) system. The command cannot be processed.

User response: No action required. Informational message.

FMNDD029 Must be <= max value (short). The min value entered must be less than or equal to, the max value entered. (long).

Explanation: The user entered a value in a "minimum" field on the FM/DB2 Object List Options (2 of 2) panel. The value entered is greater than the corresponding maximum value, which is invalid.

User response: Check the minimum and maximum values for the appropriate column type. The minimum value sets the minimum display width for that column type; the maximum value set the maximum display width for that column type. The minimum display width cannot exceed the maximum.

FMNDD100 Location:

Explanation: This is translatable message text used in various places.

User response: No action required. Translatable text.

FMNDD501 SQLCOST = Cst>

Explanation: This message is issued when a SQL statement is about to be executed, and indicates the comparative cost Cost of executing the current statement. The cost values are those returned by DB2 when the SQL statement to be executed is prepared. See the SQL Reference manual for the appropriate version of DB2 for a more detailed explanation.

User response: No action required. Informational message.

FMNDD502 Invalid at remote server (short). "Line command" is only a valid command when entered against an object at the local server. For example, when the location field is blank. (long).

Explanation: This message is issued by the FM/DB2 Interactive SQL (ISQ) Processor (Options 3.4, 3.5 and 4.5 use the ISQ processor). Some line commands are only valid when FM/DB2 is connected to the local DB2 server, an example is privilege related commands such as GRANT
and REVOKE. The command cannot be executed.
User response: Connect directly to the remote server and reissue the command.

FMNDD503 Data refreshed
Explanation: This message is issued by the FM/DB2 Interactive SQL (ISQ) processor (used by the following options: 3.4, 3.5, 4.5). The data for the currently displayed object list has been refreshed. This can occur when, for example, a line command issued against an entry on the current panel was used to change the entry. One example of this is deleting the object.
User response: No action required. Informational message.

FMNDD504 Invalid command (short). "Line command" is not a valid command. Valid commands are D(n), I(n), R(n), and S. (long).
Explanation: This message is issued by the FM/DB2 Object functions utility. When a list of, for example, columns is displayed, the user can issue various commands to modify the entries in the list. The command line command entered is invalid.
User response: No action required. Informational message.

FMNDD505 Invalid command (short). "Line command" is not a valid command. Valid commands are D(n), I(n), R(n). (long).
Explanation: This message is issued by the FM/DB2 Object functions utility. When a list of, for example, columns is displayed, the user can issue various commands to modify the entries in the list. The command line command entered is invalid.
User response: No action required. Informational message.

FMNDD506 Invalid selection (short). Use S to select the column. (long).
Explanation: This message is issued by the FM/DB2 Object functions utility. An invalid character was entered against an entry in a list of columns.
User response: Use 'S' to select the column.

FMNDD507 Waiting for DB2 prepare ...
Explanation: This message is issued by those FM/DB2 functions that use the FM/DB2 editor to display the result table for a SELECT statement. The FM/DB2 editor is operating in "Large" mode, meaning that a DB2 scrollable cursor is in use. The two operating modes for the FM/DB2 editor are described in detail in the FM User's Guide and Reference for DB2 data. Briefly, when a scrollable cursor is used, there can be a delay while DB2 prepares and opens the SQL statement used to access the data. See also message FMNDD508.
User response: No action required. Informational message.

FMNDD508 Waiting for DB2 open ...
Explanation: This message is issued by those FM/DB2 functions that use the FM/DB2 editor to display the result table for a SELECT statement. The FM/DB2 editor is operating in "Large" mode, meaning that a DB2 scrollable cursor is in use. The two operating modes for the FM/DB2 editor are described in detail in the FM User's Guide and Reference for DB2 data. Briefly, when a scrollable cursor is used, there can be a delay while DB2 prepares and opens the SQL statement used to access the data. See also message FMNDD507.
User response: No action required. Informational message.

FMNDD510 Press enter to display the SQL for the view. You can modify/execute the SQL by selecting the "Execute SQL from data set" option.
Explanation: This message is issued when the user enters the VS (show SQL used to create a view) command against a view in an object list display. The Edit/Execute SQL (Data Set) function is used to display the resulting SQL. When the user presses END or CANCEL from the SQL display, the object list is redisplayed.
User response: Press enter to see the SQL used to create the view.

FMNDD511 Enter the name of a database and table space defined with at least 8K pages. The DB2 Version 8 DSN_FUNCTION_TABLE requires a larger page size than previous DB2 versions.
Explanation: This message is issued when the user attempts to create a function table (DSN_FUNCTION_TABLE) on the Explain Utilities panel. This table requires a table space with a page size of 8K or greater, therefore both a database and table space name are required (The default database/table space will have a page size of 4K, which is inadequate).
User response: Enter the name of a database and table space that have a page size of at least 8K. You may need to contact your DB2 Database/System's Administrator for authority/information on how to do this.
FMNDD540 Related tables for:

**Explanation:** This is translatable message text used in various places.

**User response:** No action required. Translatable text.

FMNDD541 No RI relationships (short). The related edit (REEDIT) command was issued.

**Explanation:** The user entered the REEDIT editor primary command. When the object being edited is part of a referential integrity relationship this command displays the other DB2 objects in the relationship. No RI relationships exist that include the object being edited.

**User response:** No action required. Informational message.

FMNDD542 Related tables

**Explanation:** This is translatable message text used in various places.

**User response:** No action required. Translatable text.

FMNDD590 10. Stored procedure

**Explanation:** This is translatable message text used by the FM/DB2 Object functions utility.

**User response:** No action required. Panel text.

FMNDD591 11. Trigger

**Explanation:** This is translatable message text used by the FM/DB2 Object functions utility.

**User response:** No action required. Panel text.

FMNDD592 12. Auxiliary table

**Explanation:** This is translatable message text used by the FM/DB2 Object functions utility.

**User response:** No action required. Panel text.

FMNDD593 13. Storage group

**Explanation:** This is translatable message text used by the FM/DB2 Object functions utility.

**User response:** No action required. Panel text.

FMNDD594 14. Global temp. table

**Explanation:** This is translatable message text used by the FM/DB2 Object functions utility.

**User response:** No action required. Panel text.

FMNDD595 15. Sequence

**Explanation:** This is translatable message text used by the FM/DB2 Object functions utility.

**User response:** No action required. Panel text.

FMNDD596 16. Role

**Explanation:** This is translatable message text used by the FM/DB2 Object functions utility.

**User response:** No action required. Panel text.

FMNDD597 17. Trusted context

**Explanation:** This is translatable message text used by the FM/DB2 Object functions utility.

**User response:** No action required. Panel text.

FMNDE495 No DB2 TABLE matches the value entered. It is possible a DB2 object (eg view) that matches the name entered exists - however the selected DB2 utility supports DB2 tables ONLY.

**Explanation:** This message is issued when a DB2 object name - other than a table - is entered when preparing a DB2 LOAD or UNLOAD utility job. The DB2 LOAD and UNLOAD utilities support only table objects - not views, aliases, synonyms and so on.

**User response:** Ensure that the DB2 object name specified is a DB2 table.

FMNDE540 10. Function

**Explanation:** This is translatable message text used by the FM/DB2 Object privileges utility.

**User response:** No action required. Panel text.

FMNDE541 11. Stored procedure

**Explanation:** This is translatable message text used by the FM/DB2 Object privileges utility.

**User response:** No action required. Panel text.

FMNDE542 12. Storage group

**Explanation:** This is translatable message text used by the FM/DB2 Object privileges utility.

**User response:** No action required. Panel text.

FMNDE543 13. Buffer pool

**Explanation:** This is translatable message text used by the FM/DB2 Object privileges utility.

**User response:** No action required. Panel text.
FMNDE544  14. System
Explanation: This is translatable message text used by the FM/DB2 Object privileges utility.
User response: No action required. Panel text.

FMNDE545  15. Sequence
Explanation: This is translatable message text used by the FM/DB2 Object privileges utility.
User response: No action required. Panel text.

FMNDE590  1. Database 9. Function
Explanation: This is translatable message text used by the FM/DB2 Object functions utility.
User response: No action required. Panel text.

FMNDE591  2. Table space 10. Stored procedure
Explanation: This is translatable message text used by the FM/DB2 Object functions utility.
User response: No action required. Panel text.

FMNDE592  3. Table 11. Trigger
Explanation: This is translatable message text used by the FM/DB2 Object functions utility.
User response: No action required. Panel text.

FMNDE593  4. View 12. Auxiliary table
Explanation: This is translatable message text used by the FM/DB2 Object functions utility.
User response: No action required. Panel text.

FMNDE594  5. Alias 13. Storage group
Explanation: This is translatable message text used by the FM/DB2 Object functions utility.
User response: No action required. Panel text.

Explanation: This is translatable message text used by the FM/DB2 Object functions utility.
User response: No action required. Panel text.

FMNDE596  7. Synonym 15. Sequence
Explanation: This is translatable message text used by the FM/DB2 Object functions utility.
User response: No action required. Panel text.

FMNDE597  8. Distinct type
Explanation: This is translatable message text used by the FM/DB2 Object functions utility.
User response: No action required. Panel text.

FMNDE690  Object not found (DB2 SSID) (short).
No DB2 object matches the combination of object details entered in DB2 system DB2 SSID. Press Help for more information. (long).
Explanation: This message is issued when a DB2 object name has been entered on a function entry panel, but DB2 reports that the object name does not exist.
DB2 SSID is the name of the currently connected DB2 sub-system. The function cannot be processed until the reason for the error is determined and corrected.
User response: Press the the Help Pfkey (PF1 by default) twice to show, in order, the long message (see above) and the extended help, which is a panel. The information on the help panel is shown below.
- The specified DB2 object does not exist:
  - Is the spelling correct?
  - Are you connected to the correct DB2 system?
- There was an uppercase/lowercase mismatch.
  The Translate DB2 object name option is OFF; if the name that was entered contains lower case characters, it may not be recognized. To change the Translate DB2 object name option select option 0.0.2.
- If no OWNER was specified, FM/DB2 used authid to qualify the name.
- A synonym name was specified, but the owner of the synonym is not authid.
  A synonym name is only recognized when the current SQLID and the synonym owner match.
- The object name specified is too long to be displayed on the function entry panel. If you used EXPAND to enter a long name in a field, you must use EXPAND again to clear all parts of the name. Clearing the field on the function entry panel only sets the visible part of the field to spaces.

Read the information on the panel and consider the various causes of the error in turn until the cause is determined.

FMNDE691  Invalid location (short). Location Location is not defined in SYSIBM.LOCATIONS at the local server. (long).
Explanation: This message is issued when a DB2 location value has been entered on a function entry panel, but DB2 reports that the location cannot be found. Location is the name of the location value that could not be located. The function cannot be processed until the reason for the error is determined and corrected.
User response: You can enter * in the location field.
and press Enter to show a list of locations defined at
the local server. This feature may not be available if the
product installer has chosen to disable it. If you cannot
determine the reason why the entered location value is
invalid contact your DB2 administrator for assistance.

FMNDE692  Object not found (short). No DB2 object
at location Location matches the
combination of object details entered at
DB2 system DB2 SSID. Press Help for
more information. (long).

Explanation: This message is issued when a DB2
object name has been entered on a function entry panel,
but DB2 reports that the object name does not exist.
DB2 SSID is the name of the currently connected DB2
sub-system. Location is the value entered in the
LOCATION entry field. The function cannot be
processed until the reason for the error is determined
and corrected.

User response: Press the the Help PFkey (PF1 by
default) twice to show, in order, the long message (see
above) and the extended help, which is a panel. The
information on the help panel is shown below:

Possible reasons why DB2 could not find this object
include:
• The specified DB2 object does not exist:
  – Is the spelling correct?
  – Have you specified the correct location?
• There was an uppercase/lowercase mismatch. The
  Translate DB2 object name option is OFF, if the name
  that was entered contains lower case characters, it
  may not be recognized. To change the Translate DB2
  object name option select option 0.0.2.
• If no OWNER was specified, FM/DB2 used authid to
  qualify the name.
• A synonym name was specified, but the owner of the
  synonym is not [KEITH: authid?].
  A synonym name is only recognized when the
  current SQLID and the synonym owner match.
• The object name specified is too long to be displayed
  on the function entry panel. If you used EXPAND to
  enter a long name in a field, you must use EXPAND
  again to clear all parts of the name. Clearing the field
  on the function entry panel only sets the visible part
  of the field to spaces.

Read the information on the panel and consider the
various causes of the error in turn until the cause is
determined.

FMNDE693  Check name (spaces) (short). The DB2
object name cannot be found. There are
two or more consecutive spaces in the
object name, this might indicate
unwanted text in the non-visible parts
of a scrollable field. Place the cursor on
the object name field and press the
expand PF key to verify the contents.
(long).

Explanation: This message is issued when a DB2
object name has been entered on a function entry panel,
but DB2 reports that the object name does not exist.
The DB2 object name contains two or more consecutive
spaces, which may be valid, but more likely indicates
the cause of the error.

User response: Place the cursor on the object name
field and press the PFKey assigned to EXPAND (PF4 by
default) to show the complete field contents. If the
EXPAND key is not available you can type the
EXPAND command on the command line, position the
cursor on the object name field and press ENTER to
show the complete field contents. Remove any spurious
data and retry the operation.

FMNDE694  Check owner (spaces) (short). The DB2
object name cannot be found. There are
two or more consecutive spaces in the
object owner, this might indicate
unwanted text in the non-visible parts
of a scrollable field. Place the cursor on
the object owner field and press the
expand PF key to verify the contents.
(long).

Explanation: This message is issued when a DB2
object name has been entered on a function entry panel,
but DB2 reports that the object name does not exist.
The DB2 object owner contains two or more
consecutive spaces, which may be valid, but more
likely indicates the cause of the error.

User response: Place the cursor on the object owner
field and press the PFKey assigned to EXPAND (PF4 by
default) to show the complete field contents. If the
EXPAND key is not available you can type the
EXPAND command on the command line, position the
cursor on the object owner field and press ENTER to
show the complete field contents. Remove any spurious
data and retry the operation.

FMNDE695  No DB2 TABLE matches the value
entered. It is possible a DB2 object (eg
view) that matches the name entered
exists - however the selected DB2 utility
supports DB2 tables ONLY. (long).

Explanation: This message is issued by the DB2
utilities function when an object not found condition is
encountered. Some DB2 utilities require a table as
input, and will not process a view, synonym or other objects.

**User response:** Use the FM/DB2 object list utility to display information about the DB2 object. Determine whether the object exists, and determine the type of the object (table, view etc). Consult the DB2 utilities manual for the appropriate version of DB2 to determine which object types may be used.


**Explanation:** This is translatable message text used by the FM/DB2 Object functions utility.

**User response:** No action required. Panel text.

**FMNDF591** 2. Table space 10. Stored procedure

**Explanation:** This is translatable message text used by the FM/DB2 Object functions utility.

**User response:** No action required. Panel text.

**FMNDF592** 3. Table 11. Trigger

**Explanation:** This is translatable message text used by the FM/DB2 Object functions utility.

**User response:** No action required. Panel text.

**FMNDF593** 4. View 12. Auxiliary table

**Explanation:** This is translatable message text used by the FM/DB2 Object functions utility.

**User response:** No action required. Panel text.

**FMNDF594** 5. Alias 13. Storage group

**Explanation:** This is translatable message text used by the FM/DB2 Object functions utility.

**User response:** No action required. Panel text.


**Explanation:** This is translatable message text used by the FM/DB2 Object functions utility.

**User response:** No action required. Panel text.

**FMNDF596** 7. Synonym 15. Sequence

**Explanation:** This is translatable message text used by the FM/DB2 Object functions utility.

**User response:** No action required. Panel text.
FMNDH015  HLASM compiler specifications
Explanation: This is translatable message text that appears in pull-down menu lists.
User response: No action required. Panel text.

FMNDH016  PL/I compiler specifications
Explanation: This is translatable message text that appears in pull-down menu lists.
User response: No action required. Panel text.

FMNDH017  Temporary Data Set Allocations
Explanation: This is translatable message text that appears in pull-down menu lists.
User response: No action required. Panel text.

FMNDH018  Output Data Set Allocations
Explanation: This is translatable message text that appears in pull-down menu lists.
User response: No action required. Panel text.

FMNDH019  ISPF settings
Explanation: This is translatable message text that appears in pull-down menu lists.
User response: No action required. Panel text.

FMNDH020  Editor options
Explanation: This is translatable message text that appears in pull-down menu lists.
User response: No action required. Panel text.

FMNDH021  Copy utility options
Explanation: This is translatable message text that appears in pull-down menu lists.
User response: No action required. Panel text.

FMNDH022  Object list utility options
Explanation: This is translatable message text that appears in pull-down menu lists.
User response: No action required. Panel text.

FMNDH023  Export utility options
Explanation: This is translatable message text that appears in pull-down menu lists.
User response: No action required. Panel text.

FMNDH024  DB2 LOAD utility options
Explanation: This is translatable message text that appears in pull-down menu lists.
User response: No action required. Panel text.

FMNDH025  DB2 Utility LISTDEF options
Explanation: This is translatable message text that appears in pull-down menu lists.
User response: No action required. Panel text.

FMNDH026  DB2 Utility OPTIONS options
Explanation: This is translatable message text that appears in pull-down menu lists.
User response: No action required. Panel text.

FMNDH027  DB2 Utility TEMPLATE options
Explanation: This is translatable message text that appears in pull-down menu lists.
User response: No action required. Panel text.

FMNDH028  DB2 Unload utility options
Explanation: This is translatable message text that appears in pull-down menu lists.
User response: No action required. Panel text.

FMNDH029  Trace options
Explanation: This is translatable message text that appears in pull-down menu lists.
User response: No action required. Panel text.

FMNDH030  Print utility options
Explanation: This is translatable message text that appears in pull-down menu lists.
User response: No action required. Panel text.

FMNDH031  Import utility options
Explanation: This is translatable message text that appears in pull-down menu lists.
User response: No action required. Panel text.

FMNDH040  Print
Explanation: This is translatable message text that appears in pull-down menu lists.
User response: No action required. Panel text.
**FMNDH041 • FMNDH067**

- **FMNDH041 Objects**
  - **Explanation:** This is translatable message text that appears in pull-down menu lists.
  - **User response:** No action required. Panel text.

- **FMNDH042 Copy**
  - **Explanation:** This is translatable message text that appears in pull-down menu lists.
  - **User response:** No action required. Panel text.

- **FMNDH043 Object List**
  - **Explanation:** This is translatable message text that appears in pull-down menu lists.
  - **User response:** No action required. Panel text.

- **FMNDH044 DB2 Privileges**
  - **Explanation:** This is translatable message text that appears in pull-down menu lists.
  - **User response:** No action required. Panel text.

- **FMNDH045 Import**
  - **Explanation:** This is translatable message text that appears in pull-down menu lists.
  - **User response:** No action required. Panel text.

- **FMNDH046 Export**
  - **Explanation:** This is translatable message text that appears in pull-down menu lists.
  - **User response:** No action required. Panel text.

- **FMNDH047 Create**
  - **Explanation:** This is translatable message text that appears in pull-down menu lists.
  - **User response:** No action required. Panel text.

- **FMNDH048 DB2 Utilities**
  - **Explanation:** This is translatable message text that appears in pull-down menu lists.
  - **User response:** No action required. Panel text.

- **FMNDH049 Audit Report**
  - **Explanation:** This is translatable message text that appears in pull-down menu lists.
  - **User response:** No action required. Panel text.

- **FMNDH050 Print Browse**
  - **Explanation:** This is translatable message text that appears in pull-down menu lists.
  - **User response:** No action required. Panel text.

- **FMNDH060 Help for help...**
  - **Explanation:** This is translatable message text that appears in pull-down menu lists.
  - **User response:** No action required. Panel text.

- **FMNDH061 Extended help...**
  - **Explanation:** This is translatable message text that appears in pull-down menu lists.
  - **User response:** No action required. Panel text.

- **FMNDH062 Keys help...**
  - **Explanation:** This is translatable message text that appears in pull-down menu lists.
  - **User response:** No action required. Panel text.

- **FMNDH063 Help index A-M**
  - **Explanation:** This is translatable message text that appears in pull-down menu lists.
  - **User response:** No action required. Panel text.

- **FMNDH064 Help index N-Z**
  - **Explanation:** This is translatable message text that appears in pull-down menu lists.
  - **User response:** No action required. Panel text.

- **FMNDH065 Tutorial...**
  - **Explanation:** This is translatable message text that appears in pull-down menu lists.
  - **User response:** No action required. Panel text.

- **FMNDH066 About...**
  - **Explanation:** This is translatable message text that appears in pull-down menu lists.
  - **User response:** No action required. Panel text.

- **FMNDH067 About DB2...**
  - **Explanation:** This is translatable message text that appears in pull-down menu lists.
  - **User response:** No action required. Panel text.
FMNDH068  News about FM/DB2...
Explanation: This is translatable message text that appears in pull-down menu lists.
User response: No action required. Panel text.

FMNDI000  Data Format:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDI001  Execution Options:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDI002  Null Indicators:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDI003  Delimited Variables (CSV):
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDI004  More Options:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDI010  Decimal Data:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDI011  Integer Data:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDI012  Floating Point Data:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDI013  Graphic Data (fixed length):
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDI015  Selected Options:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDI016  Export data format:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDI017  Execution:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDI018  Data set allocation:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDI020  1. Separate
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDI021  2. None
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDI022  3. CSV, use NI char
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDI023  1. Before
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.
<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
<th>Explanation</th>
<th>User response</th>
</tr>
</thead>
<tbody>
<tr>
<td>FMNDI024</td>
<td>2. After</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDI025</td>
<td>3. User defined</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDI026</td>
<td>1. One byte</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDI027</td>
<td>2. Two byte</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDI028</td>
<td>3. Zoned format</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDI029</td>
<td>1. Internal (packed) format</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDI030</td>
<td>2. Zoned format</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDI031</td>
<td>3. External format</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDI032</td>
<td>1. Online</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDI033</td>
<td>2. Batch</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDI034</td>
<td>3. Use shift-out/shift-in (SOSI) characters</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDI035</td>
<td>Edit DB2 UNLOAD options</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDI036</td>
<td>1. Batch, using DB2 UNLOAD</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDI037</td>
<td>3. Batch, using DB2 UNLOAD</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDI038</td>
<td>Null indicators / CSV options</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDI039</td>
<td>Data type format</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDI040</td>
<td>&quot;ASIS&quot; encapsulation</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDI041</td>
<td>Include column headers</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDI042</td>
<td>Separator character</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDI043</td>
<td>Data type format</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDI044</td>
<td>Data type format</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDI045</td>
<td>External format</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDI046</td>
<td>Edit DB2 UNLOAD options</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
</tbody>
</table>

914  File Manager/DB2 Data V13R1 User's Guide
FMNDI050 (For user-defined, CSV data formats)
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDI051 (For user-defined data format)
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDI052 Usage
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDI053 Placement
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDI054 Indicator
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDI055 Select option
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDI056 Type
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ000 ROLE
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ001 Privilege to UPDATE column :
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ002 Privilege to REFERENCE column :
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ003 Held by application plan . :
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ004 Held by application package . :
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ005 Privilege ............. :
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ006 Can create REFERENCES on this column
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ007 Can UPDATE this column
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ008 Collection ID ............. :
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ009 Consistency token ........
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ010 Held by authorization ID . :
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.
FMNDJ101  Grant Column Privileges
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ102  Grant Database Privileges
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ103  Grant Distinct Type Privileges
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ104  Grant Type Privileges
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ105  Grant Function Privileges
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ106  Grant Storage Group Privileges
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ107  Grant Schema Privileges
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ108  Grant Package Privileges
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ109  Grant Collection Privileges
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.
<table>
<thead>
<tr>
<th>Message ID</th>
<th>Description</th>
<th>Explanation</th>
<th>User response</th>
</tr>
</thead>
<tbody>
<tr>
<td>FMNDJ110</td>
<td>Grant Stored Procedure Privileges</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ111</td>
<td>Grant Application Plan Privileges</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ112</td>
<td>Grant Sequence Privileges</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ113</td>
<td>Grant Table Space Privileges</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ114</td>
<td>Grant Table Privileges</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ115</td>
<td>Grant System Privileges</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ120</td>
<td>Revoke Buffer Pool Privileges</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ121</td>
<td>Revoke Database Privileges</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ122</td>
<td>Revoke Distinct Type Privileges</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ123</td>
<td>Revoke Type Privileges</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ124</td>
<td>Revoke Function Privileges</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ125</td>
<td>Revoke Storage Group Privileges</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ126</td>
<td>Revoke Schema Privileges</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ127</td>
<td>Revoke Package Privileges</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ128</td>
<td>Revoke Collection Privileges</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ129</td>
<td>Revoke Stored Procedure Privileges</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ130</td>
<td>Revoke Application Plan Privileges</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ131</td>
<td>Revoke Sequence Privileges</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
</tbody>
</table>
FMNDJ132 • FMNDJ155

FMNDJ132  Revoke Table Space Privileges
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.

FMNDJ133  Revoke Table Privileges
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.

FMNDJ134  Revoke System Privileges
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.

FMNDJ140  ROLE
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.

FMNDJ141  or
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.

FMNDJ142  Select with Y, G (with Grant option)
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.

FMNDJ143  Select with any character
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.

FMNDJ144  PUBLIC
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.

FMNDJ145  ALL
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.

FMNDJ146  To
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.

FMNDJ147  By
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.

FMNDJ148  (Defaults to DSND04 if left blank)
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.

FMNDJ150  Column
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.

FMNDJ151  Plan
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.

FMNDJ152  Collection
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.

FMNDJ153  Package
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.

FMNDJ154  Schema
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.

FMNDJ155  ON DISTINCT TYPE
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.
FMNDJ156  ON TYPE
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.

FMNDJ157  Storage group
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.

FMNDJ158  Bufferpool
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.

FMNDJ159  From
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.

FMNDJ161  USAGE ON DISTINCT TYPE
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.

FMNDJ162  USAGE ON TYPE
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.

FMNDJ170  From
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.

FMNDJ173  FM/DB2 System Options (3 of 4)
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.

FMNDJ174  Editor Options (1 of 8)
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.

FMNDJ175  Editor Options (2 of 8)
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.

FMNDJ176  Editor Options (3 of 8)
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.

FMNDJ177  Editor Options (4 of 8)
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.

FMNDJ178  Editor Options (5 of 8)
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.

FMNDJ179  Editor Options (6 of 8)
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.

FMNDJ180  Editor Options (7 of 8)
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.

FMNDJ181  Editor Options (8 of 8)
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.
FMNDJ184 • FMNDJ209

FMNDJ184  Copy Options
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ185  Object List Options (1 of 2)
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ186  Object List Options (2 of 2)
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ187  Import Options
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ189  Print Utility Options
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ190  COPY From
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ191  IMPORT From
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ192  EXPORT From
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ199  Location
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ200  OWNER
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ201  Name
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ202  Database
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ203  Table Space
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ204  Row count
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ184  Data set name
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ205  Member
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ206  Copy count
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ184  Volume
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.
FMNDJ210  DB2 Object:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ211  Template:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ212  Processing Options:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ213  From Template:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ214  To Template:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ215  From DB2 Object:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ216  To DB2 Object:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ217  From Partitioned, Sequential or VSAM Data Set:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ218  From Copybook or Template:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ219  To Copybook or Template:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ220  (optional)
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ221  Edit options
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ222  Edit template
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ223  Re-edit template
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ224  Batch execution
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ225  Use uncommitted read
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ226  Print HEX representation
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ227  Copy panel values
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.
<table>
<thead>
<tr>
<th>Code</th>
<th>Message Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FMNDJ228</td>
<td>Edit copy options</td>
</tr>
<tr>
<td><strong>Explanation</strong>: This is a translatable message displayed on various FM/DB2 panels.</td>
<td></td>
</tr>
<tr>
<td><strong>User response</strong>: No action required. Panel text.</td>
<td></td>
</tr>
<tr>
<td>FMNDJ229</td>
<td>Edit template mapping</td>
</tr>
<tr>
<td><strong>Explanation</strong>: This is a translatable message displayed on various FM/DB2 panels.</td>
<td></td>
</tr>
<tr>
<td><strong>User response</strong>: No action required. Panel text.</td>
<td></td>
</tr>
<tr>
<td>FMNDJ230</td>
<td>Template usage</td>
</tr>
<tr>
<td><strong>Explanation</strong>: This is a translatable message displayed on various FM/DB2 panels.</td>
<td></td>
</tr>
<tr>
<td><strong>User response</strong>: No action required. Panel text.</td>
<td></td>
</tr>
<tr>
<td>FMNDJ231</td>
<td>Enter &quot;/&quot;, &quot;A&quot;lways to select option</td>
</tr>
<tr>
<td><strong>Explanation</strong>: This is a translatable message displayed on various FM/DB2 panels.</td>
<td></td>
</tr>
<tr>
<td><strong>User response</strong>: No action required. Panel text.</td>
<td></td>
</tr>
<tr>
<td>FMNDJ232</td>
<td>Print mode</td>
</tr>
<tr>
<td><strong>Explanation</strong>: This is a translatable message displayed on various FM/DB2 panels.</td>
<td></td>
</tr>
<tr>
<td><strong>User response</strong>: No action required. Panel text.</td>
<td></td>
</tr>
<tr>
<td>FMNDJ233</td>
<td>Enter &quot;/&quot;, &quot;A&quot;lways</td>
</tr>
<tr>
<td><strong>Explanation</strong>: This is a translatable message displayed on various FM/DB2 panels.</td>
<td></td>
</tr>
<tr>
<td><strong>User response</strong>: No action required. Panel text.</td>
<td></td>
</tr>
<tr>
<td>FMNDJ234</td>
<td>Data format</td>
</tr>
<tr>
<td><strong>Explanation</strong>: This is a translatable message displayed on various FM/DB2 panels.</td>
<td></td>
</tr>
<tr>
<td><strong>User response</strong>: No action required. Panel text.</td>
<td></td>
</tr>
<tr>
<td>FMNDJ235</td>
<td>Duplicate row options</td>
</tr>
<tr>
<td><strong>Explanation</strong>: This is a translatable message displayed on various FM/DB2 panels.</td>
<td></td>
</tr>
<tr>
<td><strong>User response</strong>: No action required. Panel text.</td>
<td></td>
</tr>
<tr>
<td>FMNDJ236</td>
<td>Disposition</td>
</tr>
<tr>
<td><strong>Explanation</strong>: This is a translatable message displayed on various FM/DB2 panels.</td>
<td></td>
</tr>
<tr>
<td><strong>User response</strong>: No action required. Panel text.</td>
<td></td>
</tr>
<tr>
<td>FMNDJ237</td>
<td>Object type</td>
</tr>
<tr>
<td><strong>Explanation</strong>: This is a translatable message displayed on various FM/DB2 panels.</td>
<td></td>
</tr>
<tr>
<td><strong>User response</strong>: No action required. Panel text.</td>
<td></td>
</tr>
<tr>
<td>FMNDJ240</td>
<td>1. Above</td>
</tr>
<tr>
<td><strong>Explanation</strong>: This is a translatable message displayed on various FM/DB2 panels.</td>
<td></td>
</tr>
<tr>
<td><strong>User response</strong>: No action required. Panel text.</td>
<td></td>
</tr>
<tr>
<td>FMNDJ241</td>
<td>2. Previous</td>
</tr>
<tr>
<td><strong>Explanation</strong>: This is a translatable message displayed on various FM/DB2 panels.</td>
<td></td>
</tr>
<tr>
<td><strong>User response</strong>: No action required. Panel text.</td>
<td></td>
</tr>
<tr>
<td>FMNDJ242</td>
<td>3. Generate from table</td>
</tr>
<tr>
<td><strong>Explanation</strong>: This is a translatable message displayed on various FM/DB2 panels.</td>
<td></td>
</tr>
<tr>
<td><strong>User response</strong>: No action required. Panel text.</td>
<td></td>
</tr>
<tr>
<td>FMNDJ243</td>
<td>4. Generate/Replace</td>
</tr>
<tr>
<td><strong>Explanation</strong>: This is a translatable message displayed on various FM/DB2 panels.</td>
<td></td>
</tr>
<tr>
<td><strong>User response</strong>: No action required. Panel text.</td>
<td></td>
</tr>
<tr>
<td>FMNDJ244</td>
<td>5. No action required. (CSV output)</td>
</tr>
<tr>
<td><strong>Explanation</strong>: This is a translatable message displayed on various FM/DB2 panels.</td>
<td></td>
</tr>
<tr>
<td><strong>User response</strong>: No action required. Panel text.</td>
<td></td>
</tr>
<tr>
<td>FMNDJ245</td>
<td>1. Table</td>
</tr>
<tr>
<td><strong>Explanation</strong>: This is a translatable message displayed on various FM/DB2 panels.</td>
<td></td>
</tr>
<tr>
<td><strong>User response</strong>: No action required. Panel text.</td>
<td></td>
</tr>
<tr>
<td>FMNDJ246</td>
<td>2. Single</td>
</tr>
<tr>
<td><strong>Explanation</strong>: This is a translatable message displayed on various FM/DB2 panels.</td>
<td></td>
</tr>
<tr>
<td><strong>User response</strong>: No action required. Panel text.</td>
<td></td>
</tr>
<tr>
<td>FMNDJ247</td>
<td>1. Old or Reuse</td>
</tr>
<tr>
<td><strong>Explanation</strong>: This is a translatable message displayed on various FM/DB2 panels.</td>
<td></td>
</tr>
<tr>
<td><strong>User response</strong>: No action required. Panel text.</td>
<td></td>
</tr>
<tr>
<td>Message Code</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>--------------------------------------------</td>
</tr>
<tr>
<td>FMNDJ248</td>
<td>2. Mod</td>
</tr>
<tr>
<td>FMNDJ250</td>
<td>Print Utility</td>
</tr>
<tr>
<td>FMNDJ251</td>
<td>Copy Utility</td>
</tr>
<tr>
<td>FMNDJ252</td>
<td>Import Utility</td>
</tr>
<tr>
<td>FMNDJ253</td>
<td>:Export Utility</td>
</tr>
<tr>
<td>FMNDJ254</td>
<td>Data Create Utility</td>
</tr>
<tr>
<td>FMNDJ255</td>
<td>UNLOAD Utility (Tables)</td>
</tr>
<tr>
<td>FMNDJ256</td>
<td>DB2 Object Functions</td>
</tr>
<tr>
<td>FMNDJ257</td>
<td>Manage DB2 Privileges</td>
</tr>
<tr>
<td>FMNDJ258</td>
<td>Basic SELECT Prototyping</td>
</tr>
<tr>
<td>FMNDJ260</td>
<td>Number of rows to print</td>
</tr>
<tr>
<td>FMNDJ261</td>
<td>Number of rows to copy</td>
</tr>
<tr>
<td>FMNDJ262</td>
<td>Number of rows to import</td>
</tr>
<tr>
<td>FMNDJ263</td>
<td>Number of rows to export</td>
</tr>
<tr>
<td>FMNDJ264</td>
<td>Number of rows to create</td>
</tr>
<tr>
<td>FMNDJ265</td>
<td>:Number of rows to unload</td>
</tr>
<tr>
<td>FMNDJ266</td>
<td>Number of rows to display</td>
</tr>
<tr>
<td>FMNDJ267</td>
<td>(required)</td>
</tr>
</tbody>
</table>
FMNDJ268  •  FMNDJ311

FMNDJ268  (default)
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.

FMNDJ269  Schema
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.

FMNDJ270  Object Identification Criteria:
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.

FMNDJ271  Select columns (S/A/D) or enter predicates to build the SELECT statement:
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.

FMNDJ272  Type SQL to generate the SQL for the drop statement only.
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.

FMNDJ279  Import dataset:
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.

FMNDJ280  In
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.

FMNDJ281  P
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.

FMNDJ290  Enter * for list
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.

FMNDJ291  Database/table/collection/schema
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.

FMNDJ300  Import count
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.

FMNDJ301  Export count
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.

FMNDJ302  Create count
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.

FMNDJ303  Unload count
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.

FMNDJ309  Start position
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.

FMNDJ310  Create object
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.

FMNDJ311  Drop object
Explanation:  This is a translatable message displayed on various FM/DB2 panels.
User response:  No action required. Panel text.
FMNDJ312  Display privileges
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ313  Grant privileges
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ314  Revoke privileges
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ320  Use REXX proc
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ321  REXX proc name
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ322  Delete existing rows
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ323  1. Ignore
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ324  2. Update
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ325  Duplicate count
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ326  NO LIMIT
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ327  Edit UNLOAD options
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ330  Confirm object drop
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ340  1. FM/DB2 (SQLDA) format
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ341  2. DB2 UNLOAD format
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ342  3. DSNTIAUL format
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ343  4. User defined
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ344  5. Delimited variables (CSV)
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ350  Object List Utility Options:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.
<table>
<thead>
<tr>
<th>ID</th>
<th>Description</th>
<th>Explanation</th>
<th>User response</th>
</tr>
</thead>
<tbody>
<tr>
<td>FMNDJ351</td>
<td>Display width for varying columns</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required.</td>
</tr>
<tr>
<td>FMNDJ352</td>
<td>Filtering</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required.</td>
</tr>
<tr>
<td>FMNDJ356</td>
<td>Export Options (1 of 3)</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required.</td>
</tr>
<tr>
<td>FMNDJ357</td>
<td>Export Options (2 of 3)</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required.</td>
</tr>
<tr>
<td>FMNDJ358</td>
<td>Export Options (3 of 3)</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required.</td>
</tr>
<tr>
<td>FMNDJ360</td>
<td>Show all catalog table columns</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required.</td>
</tr>
<tr>
<td>FMNDJ361</td>
<td>Show &quot;unused&quot; catalog table columns</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required.</td>
</tr>
<tr>
<td>FMNDJ362</td>
<td>Show &quot;internal use only&quot; catalog table columns</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required.</td>
</tr>
<tr>
<td>FMNDJ363</td>
<td>Use automatic column sizing</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required.</td>
</tr>
<tr>
<td>FMNDJ364</td>
<td>Use extended column name headings</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required.</td>
</tr>
<tr>
<td>FMNDJ365</td>
<td>Interpret columns containing codes</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required.</td>
</tr>
<tr>
<td>FMNDJ366</td>
<td>Modify order of columns</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required.</td>
</tr>
<tr>
<td>FMNDJ367</td>
<td>Display widths, filtering</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required.</td>
</tr>
<tr>
<td>FMNDJ368</td>
<td>Reposition to first matching row</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required.</td>
</tr>
<tr>
<td>FMNDJ370</td>
<td>Databases</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required.</td>
</tr>
<tr>
<td>FMNDJ371</td>
<td>Column names</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required.</td>
</tr>
<tr>
<td>FMNDJ372</td>
<td>Authorization identifiers</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required.</td>
</tr>
<tr>
<td>FMNDJ373</td>
<td>Object names</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required.</td>
</tr>
<tr>
<td>Message Code</td>
<td>Remarks</td>
<td></td>
<td></td>
</tr>
<tr>
<td>--------------</td>
<td>---------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FMNDJ374</td>
<td>Remarks</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>User response:</td>
<td>No action required. Panel text.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FMNDJ380</td>
<td>Enter minimum and maximum values for each column type</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>User response:</td>
<td>No action required. Panel text.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FMNDJ381</td>
<td>Min</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>User response:</td>
<td>No action required. Panel text.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FMNDJ382</td>
<td>Max</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>User response:</td>
<td>No action required. Panel text.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FMNDJ390</td>
<td>Maximum number of rows to fetch</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>User response:</td>
<td>No action required. Panel text.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FMNDJ400</td>
<td>Uppercase Translation:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>User response:</td>
<td>No action required. Panel text.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FMNDJ401</td>
<td>Encapsulation of SQL Identifiers in Double Quotes:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>User response:</td>
<td>No action required. Panel text.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FMNDJ402</td>
<td>More Options:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>User response:</td>
<td>No action required. Panel text.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FMNDJ403</td>
<td>Arbitrary SQL Select Statements:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>User response:</td>
<td>No action required. Panel text.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FMNDJ404</td>
<td>External Format for TIME Data Type Columns:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>User response:</td>
<td>No action required. Panel text.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FMNDJ405</td>
<td>Catalog Queries and Template Creation:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>User response:</td>
<td>No action required. Panel text.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FMNDJ406</td>
<td>CCSID Warning Message:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>User response:</td>
<td>No action required. Panel text.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FMNDJ407</td>
<td>CURRENT DECFLOAT Rounding Mode:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>User response:</td>
<td>No action required. Panel text.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FMNDJ411</td>
<td>Use double quotes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>User response:</td>
<td>No action required. Panel text.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FMNDJ412</td>
<td>Editor option</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>User response:</td>
<td>No action required. Panel text.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FMNDJ413</td>
<td>Display format</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>User response:</td>
<td>No action required. Panel text.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FMNDJ420</td>
<td>Translate DB2 object names</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>User response:</td>
<td>No action required. Panel text.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FMNDJ421</td>
<td>Translate input SQL statements</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>User response:</td>
<td>No action required. Panel text.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FMNDJ422</td>
<td>Arbitrary select statements, time display format, template creation</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-----------</td>
<td>---------------------------------------------------------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Explanation:</strong></td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>User response:</strong></td>
<td>No action required. Panel text.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>FMNDJ423</th>
<th>CCSID warning option</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Explanation:</strong></td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
</tr>
<tr>
<td><strong>User response:</strong></td>
<td>No action required. Panel text.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>FMNDJ424</th>
<th>Use uncommitted read when accessing the DB2 catalog</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Explanation:</strong></td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
</tr>
<tr>
<td><strong>User response:</strong></td>
<td>No action required. Panel text.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>FMNDJ425</th>
<th>Retrieve foreign key information when building templates</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Explanation:</strong></td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
</tr>
<tr>
<td><strong>User response:</strong></td>
<td>No action required. Panel text.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>FMNDJ426</th>
<th>1. For DB2 SQL reserved words</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Explanation:</strong></td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
</tr>
<tr>
<td><strong>User response:</strong></td>
<td>No action required. Panel text.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>FMNDJ427</th>
<th>2. For IBM and DB2 SQL reserved words</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Explanation:</strong></td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
</tr>
<tr>
<td><strong>User response:</strong></td>
<td>No action required. Panel text.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>FMNDJ428</th>
<th>3. Always</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Explanation:</strong></td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
</tr>
<tr>
<td><strong>User response:</strong></td>
<td>No action required. Panel text.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>FMNDJ429</th>
<th>1. Browse</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Explanation:</strong></td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
</tr>
<tr>
<td><strong>User response:</strong></td>
<td>No action required. Panel text.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>FMNDJ432</th>
<th>Use File Manager editor</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Explanation:</strong></td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
</tr>
<tr>
<td><strong>User response:</strong></td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>Message Code</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-------------</td>
</tr>
<tr>
<td>FMNDJ450</td>
<td>Use uncommitted read</td>
</tr>
<tr>
<td>FMNDJ451</td>
<td>Delete existing rows</td>
</tr>
<tr>
<td>FMNDJ452</td>
<td>Ignore RI/Constraint errors</td>
</tr>
<tr>
<td>FMNDJ459</td>
<td>Duplicate key processing</td>
</tr>
<tr>
<td>FMNDJ460</td>
<td>1. Ignore</td>
</tr>
<tr>
<td>FMNDJ461</td>
<td>2. Update</td>
</tr>
<tr>
<td>FMNDJ462</td>
<td>Max duplicates</td>
</tr>
<tr>
<td>FMNDJ500</td>
<td>Display Format:</td>
</tr>
<tr>
<td>FMNDJ501</td>
<td>Prefix Area:</td>
</tr>
<tr>
<td>FMNDJ502</td>
<td>Enter Key Processing:</td>
</tr>
<tr>
<td>FMNDJ503</td>
<td>Commit Options:</td>
</tr>
<tr>
<td>FMNDJ504</td>
<td>Varying Length Columns:</td>
</tr>
<tr>
<td>FMNDJ505</td>
<td>Columns that allow Null Values:</td>
</tr>
<tr>
<td>FMNDJ506</td>
<td>Table Display Format:</td>
</tr>
<tr>
<td>FMNDJ507</td>
<td>Single Display Format:</td>
</tr>
<tr>
<td>FMNDJ508</td>
<td>Primary and Foreign Keys, Indexes:</td>
</tr>
<tr>
<td>FMNDJ509</td>
<td>Editor CAPS Setting:</td>
</tr>
<tr>
<td>FMNDJ510</td>
<td>Shadow Lines:</td>
</tr>
</tbody>
</table>
FMNDJ511 Export Command:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ512 Concurrency Options:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ513 Optimization Options:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ514 Edit Locking Option:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ515 Audit Option:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ516 SQL Cursor Concurrency Options:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ517 String Delimiters:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ518 Null Column Indicators:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ519 Data Sampling:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ520 Initial display format
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ521 Enter key usage
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ522 Concurrency
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ523 Locking
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ524 Scrollable cursor type
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ525 Sampling type
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ526 Null Column Indicator:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ527 Auto Commit (Changes):
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ528 LOB and XML Columns:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.
FMNDJ529  LOB piece size:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ530  Display prefix area
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ531  Show prefix area on the right
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ532  Show excluded shadow lines
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ533  Table/Single format, prefix area and shadow options
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ534  Key, index and export options
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ535  Commit when save issued
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ536  Commit when no save errors
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ537  Remove trailing spaces
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ538  Show end of string
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ539  Convert spaces to DB2 null
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ540  Show data type
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ541  Show scale line
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ542  Show column number
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ543  Show keys, indexes
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ544  Left justify numerics
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ545  Allow updates to primary key
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ546  Show primary key
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.
FMNDJ547  •  FMNDJ564

FMNDJ547  •  Show foreign key
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ548  •  Show index indicators
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ549  •  Show SQLCODE
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ550  •  Show export options
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ551  •  Read-only access
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ552  •  Commit after data fetch
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ553  •  Varying length and nullable column options
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ554  •  Enter key and commit processing options
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ555  •  Optimization, large table and data sampling options
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ556  •  Use uncommitted read
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ557  •  Concurrency and locking options
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ558  •  Use row count value
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ559  •  Auto-commit count
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ560  •  1. Previous
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ561  •  2. Table
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ562  •  3. Single
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ563  •  1. No action
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ564  •  2. Save data
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.
FMNDJ565 3. Save data and commit changes
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ566 0. Uncommitted read
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ567 1. No with clause
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ568 2. Cursor stability
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ569 3. Read stability
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ570 4. Read stability keep locks
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ571 5. Repeatable read
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ572 6. Repeatable read keep locks
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ573 1. None
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ574 2. Share mode
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ575 3. Exclusive mode
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ576 1. Insensitive
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ577 2. Sensitive static
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ578 3. Sensitive dynamic
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ580 Prefix length
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ581 Input
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ582 Display
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ583 Optimize for
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.
<table>
<thead>
<tr>
<th>Message Code</th>
<th>Message Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FMNDJ584</td>
<td>Sampling frequency</td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
</tr>
<tr>
<td>User response:</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ585</td>
<td>Sampling limit</td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
</tr>
<tr>
<td>User response:</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ586</td>
<td>Initial skip count</td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
</tr>
<tr>
<td>User response:</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ587</td>
<td>Include count</td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
</tr>
<tr>
<td>User response:</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ588</td>
<td>Skip count</td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
</tr>
<tr>
<td>User response:</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ589</td>
<td>Sampling seed</td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
</tr>
<tr>
<td>User response:</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ590</td>
<td>(A,6,7,8,9)</td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
</tr>
<tr>
<td>User response:</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ591</td>
<td>Rows</td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
</tr>
<tr>
<td>User response:</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ595</td>
<td>Auto-size display width</td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
</tr>
<tr>
<td>User response:</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ596</td>
<td>Show part/all data</td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
</tr>
<tr>
<td>User response:</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ597</td>
<td>Max characters</td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
</tr>
<tr>
<td>User response:</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ599</td>
<td>RESET</td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
</tr>
<tr>
<td>User response:</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ600</td>
<td>Large Table Options:</td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
</tr>
<tr>
<td>User response:</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ601</td>
<td>Clustered Sampling:</td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
</tr>
<tr>
<td>User response:</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ602</td>
<td>Random Sampling:</td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
</tr>
<tr>
<td>User response:</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ603</td>
<td>Zoned Timestamp Columns:</td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
</tr>
<tr>
<td>User response:</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ606</td>
<td>Import Options:</td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
</tr>
<tr>
<td>User response:</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ607</td>
<td>From Table Concurrency Option:</td>
</tr>
<tr>
<td>Explanation:</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
</tr>
<tr>
<td>User response:</td>
<td>No action required. Panel text.</td>
</tr>
</tbody>
</table>
FMNDJ608  To Table Locking Option:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ609  Copy Options:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ610  1. None
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ611  2. Clustered sampling
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ612  3. Random sampling
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ614  1. Table display size
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ615  2. Single display size
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ616  3. Manual ...
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ620  Drop Database
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ621  Drop Table Space
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ622  Drop Table
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ623  Drop View
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ624  Drop Alias
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ625  Drop Index
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ626  Drop Synonym
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ627  Drop Distinct Type
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ628  Drop Type
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ629  Drop Function
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.
<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
<th>Explanation</th>
<th>User response</th>
</tr>
</thead>
<tbody>
<tr>
<td>FMNDJ630</td>
<td>Drop Stored Procedure</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ631</td>
<td>Drop Trigger</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ632</td>
<td>Drop Storage Group</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ633</td>
<td>Drop Sequence</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ634</td>
<td>Drop Role</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ635</td>
<td>Drop Trusted Context</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ640</td>
<td>Database</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ641</td>
<td>Table Space</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ642</td>
<td>Table</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ643</td>
<td>View</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ644</td>
<td>Alias</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ645</td>
<td>Index</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ646</td>
<td>Synonym</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ647</td>
<td>Distinct Type</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ648</td>
<td>Type</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ649</td>
<td>Function</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ650</td>
<td>Stored Procedure</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
<tr>
<td>FMNDJ651</td>
<td>Trigger</td>
<td>This is a translatable message displayed on various FM/DB2 panels.</td>
<td>No action required. Panel text.</td>
</tr>
</tbody>
</table>
FMNDJ652  Storage Group:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ653  Sequence:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ654  Role:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ655  Trusted Context:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ660  Import Report
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ670  Source and target:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ671  Rows imported:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ672  Rows updated:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ673  Errors:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ674  Row update not selected
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ675  Records read (Start, First error)
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ680  Source dataset
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ681  Committed (successful) changes
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ682  Uncommitted (unsuccessful) changes
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ683  Total successful changes
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ684  Duplicate keys/Max duplicates
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ685  Updates
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ686  Record selection
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.
FMNDJ687 • FMNDJ712

FMNDJ687  Dropped (REXX proc)
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ688  Other
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ689  DB2 object
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ690  Native Unicode processing
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ691  Batch data set creation
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ692  Edit DB2 UNLOAD options
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ693  Show char position range
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ700  (The database will be dropped)
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ701  (The table will be dropped)
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ702  (The table space will be dropped)
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ703  (The view will be dropped)
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ704  (The alias will be dropped)
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ705  (The synonym will be dropped)
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: None/ Panel text.

FMNDJ706  (The DISTINCT TYPE will be dropped)
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ707  (The TYPE will be dropped)
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ710  (Warning: Plans/packages may be invalidated by this operation)
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ711  (Warning: Cast functions and privileges may also be dropped)
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ712  (Warning: All dependent objects will also be dropped)
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ713 Press ENTER to confirm drop.
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ714 Press CANCEL or EXIT to cancel drop.
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ720 Enter, Execute and Explain SQL
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ721 Edit/Execute SQL (Data Set)
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ730 EXPLAIN command options:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ731 SELECT statement options:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ732 SQL statement:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ733 Input Data Set:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ734 SELECT Options:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ735 EXPLAIN Options:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ738 LOB Display Format:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ739 LOB Piece Size:
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ740 Query number
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ741 Execution options
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ742 Show results
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ750 First column
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ751 Last column
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ752 Comment chars
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.
FMNDJ753 • FMNDJ999

FMNDJ753  Edit data set
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ754  Execute SQL from data set
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ755  Re-edit data set after execution
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ760  XML and LOB column processing options
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ765  Split LOBs into pieces using
Explanation: This is a translatable message displayed on various FM/DB2 panels.
User response: No action required. Panel text.

FMNDJ999  File Manager/DB2 problem - message
Message number not found in table.
Explanation: This message is issued when an attempt is made to issue message Message number, but the message number could not be found. It is a "catch all" message.
User response: If this message occurs during normal processing the likely cause is a logic error. Contact IBM support for assistance.
Support resources and problem solving information

This section shows you how to quickly locate information to help answer your questions and solve your problems. If you have to call IBM support, this section provides information that you need to provide to the IBM service representative to help diagnose and resolve the problem.


- “Searching IBM support Web sites for a solution”
- “Obtaining fixes” on page 942
- “Receiving support updates through e-mail notification” on page 943
- “Receiving support updates through RSS feeds” on page 944
- “If you need to contact IBM Software Support” on page 944

Searching IBM support Web sites for a solution

You can search the available knowledge bases to determine whether your problem was already encountered and is already documented.

- “Searching the information center”
- “Searching product support documents”
- “IBM Support Assistant” on page 942

Searching the information center

You can find this publication and documentation for many other products in the IBM System z Enterprise Development Tools & Compilers information center at http://publib.boulder.ibm.com/infocenter/pdthelp/v1r1/index.jsp. Using the information center, you can search product documentation in a variety of ways. You can search across the documentation for multiple products, search across a subset of the product documentation that you specify, or search a specific set of topics that you specify within a document. Search terms can include exact words or phrases, wild cards, and Boolean operators.

To learn more about how to use the search facility provided in the IBM System z Enterprise Development Tools & Compilers information center, you can view the multimedia presentation at http://publib.boulder.ibm.com/infocenter/pdthelp/v1r1/index.jsp?topic=/com.ibm.help.doc/InfoCenterTour800600.htm.

Searching product support documents

Use the System z Enterprise Development Tools & Compilers information center or the IBM support site at www.ibm.com/software/support to search for the latest, most complete information that might help you resolve your problem.

When you access the IBM support site, you can specify any of the following products for which you want information to be displayed:
• Application Performance Analyzer for z/OS
• Debug Tool for z/OS
• Enterprise COBOL for z/OS
• Enterprise PL/I for z/OS
• Fault Analyzer for z/OS
• File Manager for z/OS
• Optim Move for DB2
• WebSphere Developer Debugger for System z
• Workload Simulator for z/OS and OS/390 Support

When you access the IBM support site, you can also use the IBM Support Portal to customize the support information to be displayed and save product names that you specify. There is also a search facility provided with the IBM Support Portal that allows you to narrow the search scope and search only product support documents for the products that you specify. The IBM Support Portal can be accessed through the IBM support site at [www.ibm.com/software/support](http://www.ibm.com/software/support) or directly at [www.ibm.com/support/entry/portal](http://www.ibm.com/support/entry/portal). For information about customizing your IBM support site experience using the IBM Support Portal, refer to [https://www.ibm.com/blogs/SPNA/entry/the_ibm_support_portal_videos?lang=en_us](https://www.ibm.com/blogs/SPNA/entry/the_ibm_support_portal_videos?lang=en_us).

**IBM Support Assistant**

The IBM Support Assistant (also referred to as ISA) is a free local software serviceability workbench that helps you resolve questions and problems with IBM software products. It provides quick access to support-related information. You can use the IBM Support Assistant to help you in the following ways:

• Search through IBM and non-IBM knowledge and information sources across multiple IBM products to answer a question or solve a problem.
• Find additional information through product and support pages, customer news groups and forums, skills and training resources and information about troubleshooting and commonly asked questions.

In addition, you can use the built-in Updater facility in IBM Support Assistant to obtain IBM Support Assistant upgrades and new features to add support for additional software products and capabilities as they become available.


**Obtaining fixes**

A product fix might be available to resolve your problem. To determine what fixes and other updates are available, the following information is available from the IBM support site. You can also view the following information from the IBM Support Portal when you specify the applicable products.

• Latest PTFs for Application Performance Analyzer for z/OS
• Latest PTFs for Debug Tool for z/OS
Latest PTFs for Fault Analyzer for z/OS
Latest PTFs for File Export for z/OS
Latest PTFs for File Manager for z/OS
Latest fixes for Optim Move for DB2
Latest PTFs for WebSphere Studio Asset Analyzer for Multiplatforms
Latest PTFs for Workload Simulator for z/OS and OS/390

When you find a fix that you are interested in, click the name of the fix to read its description and to optionally download the fix.

The IBM Support Portal is a way for you to specify specific products for which you want to display support information. The Support Portal can be accessed through the IBM support site at www.ibm.com/software/support or directly at www.ibm.com/support/entry/portal. For information about how to customize your IBM support site experience using the IBM Support Portal, refer to https://www.ibm.com/blogs/SPNA/entry/the.ibm_support_portal_videos?lang=en_us.

For more information about the types of fixes that are available, see the IBM Software Support Handbook at http://techsupport.services.ibm.com/guides/handbook.html.

Receiving support updates through e-mail notification

To receive e-mail notifications about fixes and other software support news, follow the steps below. Additional information is provided at http://www.ibm.com/support/docview.wss?rs=615&uid=swg21172598.

2. Click Request e-mail updates in the Additional support links section of the page.
3. Click any My Notifications link on the page that is displayed.
4. If you have already registered for My notifications, sign in and skip to the next step. If you have not registered, click register now. Complete the registration form using your e-mail address as your IBM ID and click Submit.
5. In the My notifications tool, click the Subscribe tab to specify products for which you want to receive e-mail updates.
6. To specify Problem Determination Tools products, click Other software and then select the products for which you want to receive e-mail updates, for example, Debug Tool for z/OS and File Manager for z/OS.
7. To specify a COBOL or PL/I compiler, click Rational® and then select the products for which you want to receive e-mail updates, for example, Enterprise COBOL for z/OS.
8. After selecting all products that are of interest to you, scroll to the bottom of the list and click Continue.
9. Determine how you want to save your subscription. You can use the default subscription name or create your own by entering a new name in the Name field. It is recommended that you create your own unique subscription name using a something easily recognized by you. You can create a new folder by entering a folder name in the New field or select an existing folder from the pulldown list. A folder is a container for multiple subscriptions.
10. Specify the types of documents you want and the e-mail notification frequency.
11. Scroll to the bottom of the page and click **Submit**.

To view your current subscriptions and subscription folders, click **My subscriptions**.

If you experience problems with the **My notifications** feature, click the **Feedback** link in the left navigation panel and follow the instructions provided.

---

**Receiving support updates through RSS feeds**

To receive RSS feeds about fixes and other software support news, go to the following web site and select the products in which you are interested:


---

**If you need to contact IBM Software Support**

IBM Software Support provides assistance with product defects.

Before contacting IBM Software Support, your company must have an active IBM software maintenance contract, and you must be authorized to submit problems to IBM. The type of software maintenance contract that you need depends on the type of product you have:

- For IBM distributed software products (including, but not limited to, Tivoli®, Lotus®, and Rational products, as well as DB2 and WebSphere® products that run on Windows, or UNIX operating systems), enroll in Passport Advantage® in one of the following ways:
  
  **Online**
  
  Go to the Passport Advantage Web site at [http://www.lotus.com/services/passport.nsf/_WebDocs/Passport_Advantage_Home](http://www.lotus.com/services/passport.nsf/_WebDocs/Passport_Advantage_Home) and click **How to Enroll**.
  
  **By phone**
  
  For the phone number to call in your country, go to the IBM Software Support Web site at [http://techsupport.services.ibm.com/guides/contacts.html](http://techsupport.services.ibm.com/guides/contacts.html) and click the name of your geographic region.

- For customers with Subscription and Support (S & S) contracts, go to the Software Service Request Web site at [https://techsupport.services.ibm.com/ssr/login](https://techsupport.services.ibm.com/ssr/login)


- For IBM eServer™ software products (including, but not limited to, DB2 and WebSphere products that run in zSeries, pSeries, and iSeries environments), you can purchase a software maintenance agreement by working directly with an IBM sales representative or an IBM Business Partner. For more information about support for eServer software products, go to the IBM Technical Support Advantage Web site at [http://www.ibm.com/servers/eserver/techsupport.html](http://www.ibm.com/servers/eserver/techsupport.html)

If you are not sure what type of software maintenance contract you need, call 1-800-IBMSERV (1-800-426-7378) in the United States. From other countries, go to the contacts page of the **IBM Software Support Handbook** on the Web at [http://techsupport.services.ibm.com/guides/contacts.html](http://techsupport.services.ibm.com/guides/contacts.html) and click the name of your geographic region for phone numbers of people who provide support for your location.
To contact IBM Software support, follow these steps:
1. “Determining the business impact”
2. “Describing problems and gathering information”
3. “Submitting problems” on page 946

**Determining the business impact**

When you report a problem to IBM, you are asked to supply a severity level. Therefore, you need to understand and assess the business impact of the problem that you are reporting. Use the following criteria:

**Severity 1**

The problem has a **critical** business impact. You are unable to use the program, resulting in a critical impact on operations. This condition requires an immediate solution.

**Severity 2**

The problem has a **significant** business impact. The program is usable, but it is severely limited.

**Severity 3**

The problem has **some** business impact. The program is usable, but less significant features (not critical to operations) are unavailable.

**Severity 4**

The problem has **minimal** business impact. The problem causes little impact on operations, or a reasonable circumvention to the problem was implemented.

**Describing problems and gathering information**

When describing a problem to IBM, be as specific as possible. Include all relevant background information so that IBM Software Support specialists can help you solve the problem efficiently.

To save time, if there is a Mustgather document available for the product, refer to the Mustgather document and gather the information specified. Mustgather documents contain specific instructions for submitting your problem to IBM and gathering information needed by the IBM support team to resolve your problem. To determine if there is a Mustgather document for this product, go to the product support page and search on the term Mustgather. At the time of this publication, the following Mustgather documents are available:


If the product does not have a Mustgather document, please provide answers to the following questions:

- What software versions were you running when the problem occurred?
- Do you have logs, traces, and messages that are related to the problem symptoms? IBM Software Support is likely to ask for this information.
- Can you re-create the problem? If so, what steps were performed to re-create the problem?
- Did you make any changes to the system? For example, did you make changes to the hardware, operating system, networking software, and so on.
- Are you currently using a workaround for the problem? If so, be prepared to explain the workaround when you report the problem.

## Submitting problems

You can submit your problem to IBM Software Support in one of two ways:

### Online


### By phone

Call 1-800-IBMSERV (1-800-426-7378) in the United States or, from other countries, go to the contacts page of the IBM Software Support Handbook at [http://techsupport.services.ibm.com/guides/contacts.html](http://techsupport.services.ibm.com/guides/contacts.html) and click the name of your geographic region.

If the problem you submit is for a software defect or for missing or inaccurate documentation, IBM Software Support creates an Authorized Program Analysis Report (APAR). The APAR describes the problem in detail. Whenever possible, IBM Software Support provides a workaround that you can implement until the APAR is resolved and a fix is delivered. IBM publishes resolved APARs on the Software Support Web site daily, so that other users who experience the same problem can benefit from the same resolution.

After a Problem Management Record (PMR) is open, you can submit diagnostic MustGather data to IBM using one of the following methods:

- FTP diagnostic data to IBM
- If FTP is not possible, email diagnostic data to techsupport@mainz.ibm.com. You must add PMR xxxxx bbb ccc in the subject line of your email. xxxxx is your PMR number, bbb is your branch office, and ccc is your IBM country code. Click [here](http://itcenter.mainz.de.ibm.com/ecurep/mail/subject.html) for more details.

Always update your PMR to indicate that data has been sent. You can update your PMR online or by phone as described above.
Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
1623-14, Shimotsuruma, Yamato-shi
Kanagawa 242-8502 Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.
Notices

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
J46A/G4
555 Bailey Avenue
San Jose, CA
95141-1003
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this information and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrates programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not
been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows: ©(your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs. ©Copyright IBM Corp. _enter the year or years_.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

---

Programming interface information

The User's Guide and Reference documents intended Programming Interfaces that allow the customer to write programs to obtain the services of File Manager.

---

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at “Copyright and trademark information”, http://www.ibm.com/legal/copytrade.shtml.
Bibliography

File Manager publications

File Manager Customization Guide SC19-4118
File Manager User’s Guide and Reference SC19-4119
File Manager User’s Guide and Reference for DB2 Data SC19-4120
File Manager User’s Guide and Reference for IMS Data SC19-4121
File Manager User’s Guide and Reference for CICS SC19-4122
File Manager Fact Sheet GC25-2429
File Manager License Information GC19-4117
File Manager Program Directory GI10-8968

Related publications for Problem Determination Tools


Related publications for COBOL

IBM COBOL Language Reference SC26-9046
IBM COBOL Programming Guide for OS/390 & VM SC26-9049

Related publications for PL/I

IBM VisualAge PL/I Language Reference SC26-9476
IBM VisualAge PL/I for OS/390 Programming Guide SC26-9473

Related publications for DB2

See the following publications for your version of DB2:
DB2 for z/OS Administration Guide
DB2 Application Programming and SQL Guide
DB2 Utility Guide and Reference
DB2 for z/OS Messages and Codes
DB2 for z/OS Codes
DB2 for z/OS SQL Reference
DB2 Administration Tool for z/OS User’s Guide and Reference

Related publications for z/OS

z/OS DFSMS Access Method Services for Catalogs SC26-7394
z/OS DFSMS Object Access Method Application Programmer’s Reference SC35-0425
z/OS DFSMS: Using Data Sets SC26-7410
z/OS DFSMS: Using Magnetic Tapes SC26-7412
z/OS ISPF User’s Guide Vol I SC34-4822
z/OS ISPF User’s Guide Vol II SC34-4823
z/OS MVS JCL Reference SA22-7597
z/OS MVS System Messages, Vol 5 SA22-7635
z/OS Support for Unicode Using Conversion Services SA22-7649
z/OS TSO/E Command Reference SA22-7782
z/OS TSO/E Programming Services SA22-7789
z/OS TSO/E REXX Reference SA22-7790
z/OS TSO/E REXX User’s Guide SA22-7791
Index

Special characters
- See underscore (_) command to list valid prefix commands 129
\ (synonym for FIND command) 748
(1,*), print option 670
(n,m), print option 670
$ See national characters
@
% See national characters
> row selection criteria operator 650
>= row selection criteria operator 650
< row selection criteria operator 650
<= row selection criteria operator 650
| row selection criteria connector 650
& row selection criteria connector 650
# See national characters
= row selection criteria operator 650
~= row selection criteria operator 650
!= row selection criteria operator 650
BT row selection criteria operator 651
IN row selection criteria operator 651
LK row selection criteria operator 651
NL row selection criteria operator 651

Numerics
132, print option 669
80, print option 669

A
A line command
databases 501
functions 584
indexes 596
row selection criteria 649
stored procedures 686
table edit 691, 700
table spaces 699
tables, views and aliases 705
ABOUT primary command 727
action bar
on FM/DB2 panels 17
pull-down menu, description 365
ADD line command
tables, views and aliases 705
Additional LISTDEF Statements panel, description 367
Additional TEMPLATE Statements panel, description 368
Advanced SELECT Prototyping panel, description 368
ALL primary command 727
allow
substitution chars, DB2 LOAD utility option 608
updates to primary key option 540
alphanumeric columns (Object List Utility) 246
ALS line command
tables, views and aliases 705
AND row selection criteria connector 650
APF-authorized 9
Application Packages panel, description 387
Application Plan Details panel, description as a details panel 522
Application Plan Privilege Details panel, description as a privilege details panel 624
Application Plan Privileges panel, description as a privileges panel 624
Application Plans panel, description 389
arithmetic operators, REXX 208
ASCII data, DB2 LOAD utility option 608
asterisk (*) filtering object lists 246
using in FM/DB2 entry fields 28
AT line command
columns 412
tables, views and aliases 705
attention interrupt key 34
attributes changing for a column 85
for a numeric column, changing 75
for an alphanumeric column, changing 74
Audit trail (option 3.10), description 267
audit trail report
description 269
summary statistics 269
Auto detect, compiler language selection option 414
auto-commit count, editor option 547

B
B line command
DBRMs 499
row selection criteria 649
table edit 691, 700
tables, views and aliases 705
BACKWARD primary command 16, 728
Basic SELECT Prototyping panel, description 391
batch functions
D2G (Data Generate) 810

batch functions (continued)
D2TP (Template create/update) 812
DBC (Copy) 788
DBI (Import) 793
DBP (Print) 798
DBX (Export) 800
batch processing
Audit trail (option 3.10) 197, 267
Create (option 3.8) 498
Export (option 3.7) 227
Import (option 3.6) 202
Object List (option 3.4) 258, 259
prerequisite knowledge xii
Print (option 3.1) 263
Utilities (option 3.9) 282, 286
Batch Submission Job Statement
Information processing option 661
BI line command
application plans 390
BIGINT data type, description 361
BINARY data type, description 361
Bind Application Plan panel, description 396
Bind Package panel, description 396
BLOB (large object) 363
boolean operators, REXX 209
BOTTOM primary command 16, 729
Browse
non-updateable columns 147
read-only views 147
BROWSE primary command 729
BT line command
tables, views and aliases 706
BTW 651
build indexes in parallel, DB2 LOAD utility option 608
built-in functions, REXX 209

C
C line command
row selection criteria 649
table edit 691, 700
tables, views and aliases 706
CA-Panvalet
specifying a data set 30
using copybooks with non-DB2 data 53
CANCEL primary command 730
canceling long-running DB2 queries 34
CAPS primary command 146, 730
CASE primary command 146, 731
CC line command 649
CDI line command
tables, views and aliases 706
CH line command
tables, views and aliases 706
CHANGE
external function 213
primary command 732
Change Extended (CX) command 141
Data description support
HLASM
   DS/DC  55
   DSECT  55
   PL/I   not supported  55
   REFER  55
   VARYING  55
   VARYINGZ  55
data format, export options  559
data set
   name, specifying on panel  30
   Data set DISP, print option  670
   Data set name, print option  670
   Data set, print option  669
data sharing environment  513
data type
   displaying  130
   rules, conversion  86
data types, description
   BIGINT  361
   BINARY  361
   CHAR  361
   DATE  361
   DECFLOAT  361
   DECIMAL  361
   DOUBLE  361
   FLOAT  361
   GRAPHIC  361
   INTEGER  361
   NUMERIC  361
   REAL  361
   ROWID  361
   SMALLINT  361
   TIME  362
   VARBINARY  361
   VARCHAR  361
   VARGRAPHIC  361
data, displayed
   changing view  117
   limiting
      excluding rows  126
      row labels  116
database
   for a DB2 object, specifying  28
   Database Details panel, description as a details panel  522
   Database Privilege Details panel, description as a privilege details panel  624
   Database Privileges panel, description as a privileges panel  624
   Database Request Modules panel, description  498
   Database Structure panel, description  500
   Databases panel, description  500
   DATE data type, description  361
   DB2
   commands, entering from FM/DB2 session  156
   data
      sharing environment  513
      types supported  361
      Interactive, interface to  357
   object name, specifying on panel  25
   DB2 (continued)
   objects
      creating and dropping  163
      working with lists of  241
      privileges, managing  271
      releases supported by FM/DB2  22
      subsystem
         changing using DB2 SSID field  11
         changing using SSID command  13
         selecting  11
      Subsystem Selection menu  14
      table, printing contents  261
      utilities
         COPY  283
         generating  277
         LOAD  286
         REBUILD  292
         RECOVER  294
         REORG  298
         RUNSTATS  302
         UNLOAD  234, 306
   DB2 Browse panel, description  501
   DB2 Edit panel, description  505
   DB2 Location Selection panel, description  509
   DB2 Object Functions panel, description  510
   DB2 Save Error Action panel, description  511
   DB2 SSID field, changing DB2 subsystem  11
   DB2 Subsystem Selection panel, description  512
   DB2 tables
      populating  191
   DB2 Template Description panel, description  514
   DB2 Utilities panel, description  515
   DB2 View panel, description  518
   DB2 Primary Option Menu panel, description  521
   DB2SYS command
      See SSID command, changing DB2 subsystem
   DBC (Copy), FM/DB2 batch function  788
   DBCLOB (large object)  363
   DBCS
      data, DB2 LOAD utility option  608
      displaying or editing  46
      Settings: Load utility options (option 0.5)  608
      DBCS processing option  664
   DBI (Import), FM/DB2 batch function  793
   DBP (Print), FM/DB2 batch function  798
   DBRM Details panel, description as a details panel  522
   DBX (Export), FM/DB2 batch function  800
   DD line command
      row selection criteria  649
      table edit  692, 700
   DECIMAL data type, description  361
   DECIMAL data type, description  361
   decimal data, export options  563
   delete
      existing rows, option  417
      line (row selection criteria)  649
      row  692, 700
      row (partition information)  468
   DELETE primary command  737
   DEP line command
      packages  388
      sequences  660
   DESCRIBE primary command  738
   description, adding to a template  58
   DESELECT primary command  738
   details panels
      description  522
      list of  522
   DI line command
      columns  412
   display
      excluded row  692, 700
      format
         examples  118
         single (SNGL)  121
      table (TABL)  118
      using Zoom function key  125
   display prefix area option  536
   Display Row panel, description  523
   display, print output, when browsing or editing  118
   displaying privileges  273
   Distinct Type Details panel, description as a details panel  522
   Distinct Type Privileges panel, description as a privileges panel  624
   Distinct Type Usage Privilege Details panel, description as a privilege details panel  624
   Distinct Types panel, description  523
   DK line command
      for EXPLAIN plan table  615
      for EXPLAIN statement table  682
   do not set COPY pending flag, DB2
   LOAD utility option  607
   DOUBLE data type, description  361
   DOUBLE PRECISION data type, description  361
   DOWN primary command  16, 738
   DP line command
      for EXPLAIN plan table  615
      for EXPLAIN statement table  682
   DQ line command
      for EXPLAIN plan table  615
      for EXPLAIN statement table  682
   DR line command
      databases  501
      distinct types  524
      functions  585
      indexes  596
      sequences  660
      stored procedures  686
      synonyms  688
      table spaces  699
      tables, views and aliases  706
      triggers  713
   drop
      function  189
drop (continued)
panels
description 524
list of 524
Drop Alias panel, description as a drop panel 524
Drop Confirmation panel, description 525
Drop Database panel, description as a drop panel 524
Drop Distinct Type panel, description as a drop panel 524
Drop Function panel, description as a drop panel 524
Drop Index panel, description as a drop panel 524
Drop Stored Procedure panel, description as a drop panel 524
Drop Synonym panel, description as a drop panel 524
Drop Table panel, description as a drop panel 524
Drop Table Space panel, description as a drop panel 524
Drop Trigger panel, description as a drop panel 524
Drop View panel, description as a drop panel 524
DROP, RETURN return value 220
DS line command
databases 501
table spaces 699
DS/DC 55
DSECT 55
DT line command
columns 412
schemas 658
duplicate key processing, option 417
E line command
table edit 692, 700
tables, views and aliases 706
Edit (option 2)
converting data to uppercase 146
displaying data in hexadecimal format 128
errors, handling 157
limiting data displayed excluding rows 126
scrolling 125
sorting data 128
VARCHAR, options for processing 136
varying-length columns, options for processing 136
zooming 123
Edit options 43
edit prefix commands, list of 691, 700
EDIT primary command 740
edit related table 693, 702
Edit/Execute SQL (Data Set) panel, description 526
editing templates 57
editor
finding string 112
locating row 108
locating a labeled row 108
element CAPS Setting 529
Editor Options (1 of 8) panel, description 528
Editor Options (2 of 8) panel, description 531
Editor Options (3 of 8) panel, description 535
Editor Options (4 of 8) panel, description 540
Editor Options (5 of 8) panel, description 543
Editor Options (6 of 8) panel, description 545
Editor Options (7 of 8) panel, description 548
Editor Options (8 of 8) panel, description 551
editor session
data, displayed, changing view 117
finding numeric column 117
large numeric fields 131
large table options 132
limiting data displayed row labels 116
locating column 109
specifying type 93
SQL optimization options 132
start position, specifying 95
Unicode data 137
Encapsulation of SQL Identifiers in Double Quotes processing option 577
end of string delimiter display option 544
input option 544
END primary command 740
ending a session FM/DB2 editor 101
enforce constraints, DB2 LOAD utility option 608
engineering, reverse 257
Enter and Execute DB2 Commands panel, description 554
Enter key and commit processing options 530
Enter, Execute and Explain SQL Statements panel, description 555
EQ row selection criteria operator 650
errors in an Edit session, handling 157
SQL statements 353
EXCLUDE primary command description 741
exclude row from display 695, 703
excluded rows
sorting in editor session 128
using RESET to redisplay excluding rows 126
EXECUTE primary command 744
execution, export options 560
Exit 10
EXIT statement 211
EXPAND primary command 745
EXPLAIN
interpretation panel, description 556
plan table primary commands
D 615
DK 615
DP 615
DQ 615
FR 616
I 616
IH 616
K 616
M 616
P 616
SR 616
T 616
UH 616
X 616
primary command 745
statement table primary commands
DK 682
DP 682
DQ 682
I 682
K 682
M 682
P 682
Export Utilities panel, description 556
export options
batch data set creation 560
data format 559
decimal data 563
execution 560
floating point data 564
graphic data 564
indicator 562
integer data 564
placement 562
type 562
usage 561
Options (1 of 3) panel, description 558
Options (2 of 3) panel, description 561
Options (3 of 3) panel, description 563
Export (option 3.7)
data formats 225
description 221
notes on using 228
running in batch 234
scenarios 233
setting options for the current session 223
Export “To” panel, description 564
export data set, export options 560
EXPORT primary command description 745
Export utility options 45
Export Utility panel, description 567
exporting
scrambling data 232
expressions, REXX 204

Index 957
large
numeric fields 131
objects (LOBs) 363
table options 132
large mode editor session, specifying 93
last page, scrolling to 16
LAST parameter, using in LOCATE
command 112
launching FM/DB2 8
LC line command
table edit 693, 701
LCC line command, table edit 693, 702
LE row selection criteria operator 651
leading zeros, showing 76
left justify numerics, editor option 538
LEFT primary command 757
limiting data 113
limiting data in editor sessions
excluding rows 126
row labels 116
line (row selection criteria)
copy 649
delete 649
move 649
repeat 649
line command
area (Object List Utility) 250
query (Object List Utility) 253
Lines per page, print option 670
LISTCAT
Output Display panel, description 599
LISTDEF Options panel, description 599
LISTDEF statement, generating in DB2 utility job 280
listdef utility
options 45
LK row selection criteria operator 651
LOAD
from panel, description 601
generating DB2 utility 286
options 45
setting options for the current session 290
LOAD Utility - Using Templates panel, description 605
LOAD Utility Options panel, description 607
LOAD Utility panel using DB2 templates panel, description 604
LOAD Utility panel, description 603
LOB
description 363
LOBBWVS primary command 758
LOBEDIT primary command 759
LOBLEN primary command 760
LOBVIEW primary command 760
LOCATE primary command
description 761
in editor sessions 108
Object List utility 250
locating
column 109, 250
label (assigned to a row) 109
labeled row 108
row 108, 250
location of a DB2 object, specifying 26
locking
copy option 416
copying, log changes, DB2 LOAD utility option 607
database, REXX 209
long strings
changing 141
finding 112
long-running DB2 queries, canceling 34
LPT line command
application plans 390
packages 389
LT 650
M
M edit prefix command 693, 702
line command
application plans 390
EXPLAIN plan table 616
EXPLAIN statement table 682
row selection criteria 649
Manage DB2 Privileges panel
description 271
using 271
mapping
automatically by FM/DB2 81
c contaminated columns 86
data using templates 80
specifying 81
templates, where you can use 81
maximum discards, DB2 LOAD utility option 608
maximum return code, compiling a COBOL copybook 663
maximum return code, compiling a PL/I copybook 668
member name
PDS, specifying on panel 30
specifying on panel 30
mixed DBCS data, DB2 LOAD utility option 608
MM line command
row selection criteria 649
table edit 693, 702
MOD, print option 670
mode of editor session, specifying 93
model table, using for Create Table utility 176
move
line (row selection criteria) 649
row 693, 702
NEWS primary command 765
NEXT
parameter, using in LOCATE
command 111
primary command 765
NEXTRPT primary command 765
NIN row selection criteria operator 651
NL row selection criteria operator 651
NLK row selection criteria operator 651
NNL row selection criteria operator 651
no with clause option 550
NOALIGN processing option 664
non-DB2 data, using templates with 52
non-excluded rows, sorting in editor session 128
non-updateable columns, edit 147
normal mode editor session, specifying 93
notices 947
NR row selection criteria operator 651
null column input indicator
exporting in data 240
option 544
nullable columns, exporting data containing 240
nulls and default values, create table option 180
number of index keys, DB2 LOAD utility option 608
numeric
column, finding 117
columns (Object List Utility) 246
databases, large 131
NUMERIC data type, description 361
numerics, left justify
table display format 538
O
O line command
schemas 658
O prefix command 693
object
description 693
name, specifying on panel 25
Object List (option 3.4) 241
description 241
filter line 245
finding string 250
GEN line command 257
limiting data displayed 245
line command
area 250
query 253
locating a row or column 250
operator line 245
primary commands
FIND 250
LOCATE 250
SORT 257
reverse engineering 257
ROW line command 255
sort
multiple columns 257
one column 257
valid commands 254
working with object list panels 244
Object List Line Commands panel,
description 610
960   File Manager/DB2 Data V13R1 User's Guide
primary commands (continued)
CAPS 730
CAPS (editing) 146
CASE 731
CASE (editing) 146
CHANGE 732
CHARPOS 736
COL 736
DELETE 737
DESCRIBE 738
DESELECT 738
DOWN 16, 738
EDIT 740
END 740
EXECUTE 744
EXPAND 745
EXPLAIN 745
EXPLAIN plan table
D 615
DK 615
DP 615
DQ 615
FR 616
I 616
IH 616
K 616
M 616
P 616
SR 616
T 616
UH 616
X 616
EXPLAIN statement table
DK 682
DP 682
DQ 682
I 682
K 682
M 682
P 682
EXPORT 745
FILE 747
FIND 748
FMAP 752
FORMAT 753
FORWARD 16, 753
FS 754
FT 755
HEX 755
HINT 756
INDEXINF 756
INSRPT 757
LEFT 757
LOBBRWS 758
LOBEDIT 759
LOBLEN 760
LOBVIEW 760
LOCATE 108, 761
NEW 765
NEWS 765
NEXT 765
NEXTRPT 765
object list
sort multiple column 257
sort one column 257
primary commands (continued)
Object List
SORT 257
PB 766
PLAN 766
PREFIX 767
PREVIOUS 767
PREVRPT 768
PRINT 768
PURGE 769
QUIT
See CANCEL primary command
RCHANGE 769
RD 769
REDIT 770
REFRESH 771
REFS 771
RESET 771
RESET (in Edit sessions) 145
RFIND 772
RIGHT 773
RP 774
RUNTEMP 774
SAVE 775
SAVEAS 775
SHADOW 775
SHOW 14, 776
SHOWCOB 776
SORT 777
SORT (Object List) 257
SQL 778
SQLID 778
SSID 13
TEDIT 118, 779
TP 779
TYPE 780
UNDO 780
UP 16
VER 781
VIEW 782
XMLBRWS 782
XMLEDIT 783
XMLEN 784
XMLVIEW 785
ZOOM 123, 785
primary key
insert 692, 701
Primary Key for Table panel,
description 617
Primary Option Menu
Create (option 3.8) 191
panel description 617
Set Processing Options (option 0) 39
Settings (option 0) 671, 674
SQL prototyping, execution and
analysis (option 4) 317
Start DB2 Interactive (option 5) 357
Utilities (option 3.9) 277
PRINT
external function 216
primary command 768
Print (option 3.1) 261, 263
Print Audit Trail panel, description 618
Print Browse (option 3.11) description 266
Print Browse (option 3.11) (continued)
primary commands
FIND 748
LOCATE 761
PRINT 768
PURGE 769
Print settings 41
Print utility options 44
Print Utility Options panel,
description 619
Print Utility panel, description 621
printing
audit trail report 267
contents of a DB2 table 261
looking at the output 266
rows from a DB2 table 264
template 52
privilege, for object, revoking
list of panels 624
privileges
description 624
display panels, format 624
displaying 273
functions 271
grant panels, format 588
granting 273
panels, list of 624
revoke panels, format 648
revoking 275
using the Manage DB2 Privileges
panel 271
Privileges (option 3.5) 271
description 271
problem determination
describing problems 945
determining business impact 945
submitting problems 946
Procedure Details panel, description as a
details panel 522
procedure exits, create table option 183
Process menu 365
processing options
batch
Batch Submission Job Statement
Information 661
COBOL
additional SYSLIB data sets 662
compiler specifications 42
replacing options 662
compiler
auto detect 414
COBOL 414, 663
HLASM 414
language selection 42, 414
PL/I 414, 667
copy
Copy Options panel 197
Delete existing rows 417
duplicate key processing 417
ignore RI/constraint errors 417
locking 416
Native unicode processing 417
use uncommitted read 416
data set allocation
output 43
temporary 43
Edit options 43
processing options (continued)

edit, create audit trail 508, 529
editor
allow updates to primary key 540
Auto-commit count 547
clustered sampling 533
commit when saved 546
concurrency and locking 548
Concurrency and locking options 531
convert spaces to nulls 544
display prefix area 536
display prefix area (display) 544
display prefix area (input) 544
End of string delimiter (display) 544
End of string delimiter (input) 544
Enter key and commit processing options 530
include count 534
initial display 528
initial skip count 534
Insensitive cursor type 533
Key, index and export options 530
null with clause 550
null column input indicator 544
Only commit when no save errors 547
Optimization, large table and data sampling options 530
prefix area 536
prefix length 536
random sampling 533, 534
read stability 550
read stability keep locks 550
repeatable read 550
repeatable read keep locks 550
row count 503, 506, 519
sampling frequency 534
sampling seed 534
scrollable cursor type 532
sensitive cursor type 533
Show end of string 544
Show excluded shadow lines 539
Show export options 542
show foreign key 541
show index indicators 541
show prefix area on the right 537
show primary key 540
show SQLCODE 537
single display format options 537
skip count 534
table display format 535
table display format (1,*) 567
table display format (n,m) 567
Table/Single format, prefix area and shadow options 530
Use row count value 532
Use/Keep exclusive locks 551
Use/Keep share locks 550
Use/Keep update locks 550
Varying length and nullable column options 530
export
Native unicode processing 559
processing options (continued)
Export utility options 45
FM/DB2 system
Encapsulation of SQL Identifiers in Double Quotes 577
External Format for TIME Data
Type Columns 579
FM/DB2 Object List Options (1 of 2) panel 572
FM/DB2 Object List Options (2 of 2) panel 575
FM/DB2 System Options (1 of 4) panel 577
FM/DB2 System Options (2 of 4) panel 578
FM/DB2 System Options (3 of 4) panel 580
FM/DB2 System Options (4 of 4) panel 582
Retrieve foreign key information when building templates 579
Translate DB2 object names 577
Translate input SQL statements 577
Use edit (instead of browse) to display results 579
Use uncommitted read when accessing the DB2 catalog 579
FM/DB2 system options 41
global, setting for FM/DB2 671
HLASM
DBCS 664
NOALIGN 664
ISPF settings 43, 674
Job card specifications 42
listdef utility options 45
load
allow substitution chars 608
ASCII data 608
build indexes in parallel 608
DBCS data 608
do not set COPY pending flag 607
Enforce constraints 608
log changes 607
maximum discards 608
mixed DBCS data 608
number of index keys 608
preformat unused pages 607
replace existing data 608
resume (append new data) 608
reuse existing data 608
SBCS data 608
use IEEE floating point 608
Object List Options 44
option utility options 45
PL/1
additional SYSLIB data sets 667
compiler specifications 42, 43
print
(1,*) 670
(n,m) 670
132 669
80 669
Data set 669
Data set DISP 670
Data set name 670
processing options (continued)
print (continued)
Lines per page 670
MOD 670
OLD 670
output class 670
output destination 669
Output destination 669
Page skip 670
Record length 669
Record limits 670
REXX 669
SYSPRINT 669
Terminal 669
Translate non-printable char 670
Uppercase message text 670
Wide print 670
Print Processing Options 44, 45
Print settings 41
setting default 39
system options 41
template utility options 45	race options 43
unload utility options 45
utilities 44
View options 43
prototyping, primary commands
ALL 727
CANCEL 730
deselect 738
EXECUTE 744
INSRPT 757
NEXTRPT 765
PREVRPT 768
RESET 771
SQL 778
UNDO 780
PRTCOUNT external function 217
PS line command
distinct types 524
functions 585
stored procedures 687
triggers 713
PT line command
indexes 597
storage groups 684
table spaces 699
PTF level, displaying FM/DB2 9
pull-down menus action bar 17
PURGE primary command 769
Q
QUIT primary command
See CANCEL primary command
R
R
edit prefix command 693, 702
line command
application plans 391
collections 397
databases 501
distinct types 524
functions 585
Index 963
row (continued)
move 693, 702
overwriting 693
repeat 693, 702
repeat (partition information) 468
retrieve from clipboard 692, 700
save to clipboard 694, 702

ROW line command
application plans 391
collections 397
columns 413
databases 501
DBRMs 499
distinct types 524
functions 585
indexes 597
Object List (option 3.4) 255
packages 389
schemas 658
sequences 660
storage group storage group usage
privileges 685, 724
storage groups 684
stored procedures 687
synonyms 688
table spaces 699
tables, views and aliases 706
triggers 713
row selection criteria
connectors, list of 650
information stored in templates 50
operators, list of 650
prefix commands, list of 649
Row Selection Criteria panel,
description 648
ROWID data type, description 361
rows
printing 264
scrolling to specified number 108
selecting
for display 58
to view in editor session 106
using to limit data displayed 116, 126
RP primary command 774
RR
edit prefix command 694, 702
line command, row selection
criteria 649
rules, column mapping 86
RUNSTATS Utility (Indexes) panel,
description 652
RUNSTATS Utility (Indexes) with
LISTDEF panel, description 653
RUNSTATS Utility (Table Spaces) panel,
description 654
RUNSTATS Utility (Table Spaces) with
LISTDEF panel, description 656
RUNSTATS, generating DB2 utility 302
RUNTEMP primary command 774

S
S (continued)
line command (continued)
databases 501
packages 389
storage groups 684
tables, views and aliases 706
SAF-rule controlled auditing
Create audit trail option
Copy Utility (From) panel 420
Data Create Utility panel 498
DB2 Edit panel 508
DB2 View panel 520
Edit/Execute SQL (Data Set) panel 527
Editor Options (1 of 8) panel 529
Enter, Execute and Explain SQL panel 555
Export Utility panel 569
Import (To) panel 594
Print Utility panel 623
specifying DB2 object 104
description 47
sampling data
clustered 133
random 134
sampling frequency option 534
sampling seed option 534
sampling type option 533
SAVE primary command 775
save row to clipboard 694, 702
SAVEAS primary command 775
saving changes, editor session 96
SBCS data, DB2 LOAD utility option 608
SCH line command
distinct types 524
functions 585
stored procedures 687
triggers 713
Schema Privilege Details panel,
description as a privilege details panel 624
Schema Privileges panel, description as a
privileges panel 624
Schemas panel, description 657
scramble
options
range values, specifying 79
scramble type, specifying 77
value columns, specifying 79
value data set name,
specifying 79
value list 80
value option, specifying 78
scrambling data
copy, import, export 232
scrollable cursor type option 532
scrolling 69, 125
in panels 16
to first page 16
to labeled row 108
to last page 16
to specified column 108
to specified row 108
search
direction, LOCATE primary command 110
SEL line command
tables, views and aliases 706
Select Statement Browse panel,
description 659
Select Statement Edit panel,
description 659
selecting
column for attribute change 73
options on FM/DB2 panels 24
rows 106
selecting a column for specifying
scrambling options 77
self-referencing constraint 151, 154
sensitive cursor option 533
sequence of data, changing 70
Sequences panel, description 660
session, starting another 103
Set Batch Job Card Information panel,
description 661
Set COBOL Processing Options panel,
description 662
Set PL/I Processing Options panel,
description 666
Set Processing Options
HLASM Processing Options 663
Set Processing Options panel,
description 671
Set System Processing Options panel,
description 673
Set Utility Processing Options panel,
description 674
Settings (option 0), description 671
shadow lines
assigning labels to 695, 703
hiding or showing 775
SHADOW primary command 775
show
column number option
single display format 537, 538
table display format 536
data type option 535
end of string option 544
foreign key option 541
index indicators option 541
keys, indexes 538
primary key option 540
scale line option 530
table display format 535
show excluded shadow lines option 539
Show export options panel option 542
show prefix area on the right option 537
SHOW primary command 14, 776
show SQLCODE option 537
SHOWCOB primary command
description 776
using 10
showing
leading zeros 76
single display format
description 121
options 537
skip count option 534
SMALLINT data type, description 361
Sn edit prefix command 694, 702
templates (continued)

<table>
<thead>
<tr>
<th>Templates</th>
</tr>
</thead>
<tbody>
<tr>
<td>selecting column for attribute change 73</td>
</tr>
<tr>
<td>columns for display 68</td>
</tr>
<tr>
<td>rows for display 58</td>
</tr>
<tr>
<td>working with non-DB2 data 52</td>
</tr>
<tr>
<td>where you can use 4</td>
</tr>
<tr>
<td>working with 49</td>
</tr>
<tr>
<td>temporary data set allocation, option 43</td>
</tr>
<tr>
<td>Terminal, print option 669</td>
</tr>
<tr>
<td>TIME data type, description 362</td>
</tr>
<tr>
<td>TM external function 218</td>
</tr>
<tr>
<td>TOP primary command 17, 779</td>
</tr>
<tr>
<td>TR line command schemas 658</td>
</tr>
<tr>
<td>trace options settings 43</td>
</tr>
<tr>
<td>Translate DB2 object names processing option 577</td>
</tr>
<tr>
<td>Translate input SQL statements processing option 577</td>
</tr>
<tr>
<td>Translate non-printable chars, print option 670</td>
</tr>
<tr>
<td>Trigger Details panel, description as a details panel 522</td>
</tr>
<tr>
<td>Triggers panel, description 712</td>
</tr>
<tr>
<td>TSO region size minimum supported 7 storage considerations 105</td>
</tr>
<tr>
<td>viewing large tables or views 106</td>
</tr>
<tr>
<td>TYPE primary command 780</td>
</tr>
<tr>
<td>type, export options 562</td>
</tr>
<tr>
<td>types of DB2 data 361</td>
</tr>
</tbody>
</table>

**U**

<table>
<thead>
<tr>
<th>U</th>
</tr>
</thead>
<tbody>
<tr>
<td>UC edit prefix command 694, 703</td>
</tr>
<tr>
<td>UCC line command 695, 703</td>
</tr>
<tr>
<td>UH line command (for EXPLAIN plan table) 616</td>
</tr>
<tr>
<td>underscore (_)</td>
</tr>
<tr>
<td>filtering object lists 246</td>
</tr>
<tr>
<td>using in FM/DB2 entry fields 28</td>
</tr>
<tr>
<td>UNDO primary command 780</td>
</tr>
<tr>
<td>Unicode data 137</td>
</tr>
<tr>
<td>unique constraints, create table option 180</td>
</tr>
<tr>
<td>UNLOAD Options panel, description 713</td>
</tr>
<tr>
<td>unload utility options 45</td>
</tr>
<tr>
<td>UNLOAD Utility (Table Spaces) panel, description 716</td>
</tr>
<tr>
<td>UNLOAD Utility (Table Spaces) with LISTDEF panel, description 717</td>
</tr>
<tr>
<td>UNLOAD Utility (Tables) &quot;From&quot; panel, description 718</td>
</tr>
<tr>
<td>UNLOAD Utility (Tables) &quot;To&quot; panel, description 718</td>
</tr>
<tr>
<td>UNLOAD Utility (Tables) with LISTDEF panel, description 719</td>
</tr>
<tr>
<td>UNLOAD, generating DB2 utility 306</td>
</tr>
<tr>
<td>UP primary command 16</td>
</tr>
<tr>
<td>Uppercase message text, print option 670</td>
</tr>
</tbody>
</table>

**V**

<table>
<thead>
<tr>
<th>V</th>
</tr>
</thead>
<tbody>
<tr>
<td>V line command application plans 391</td>
</tr>
<tr>
<td>packages 389</td>
</tr>
<tr>
<td>tables, views and aliases 706</td>
</tr>
<tr>
<td>valid commands (Object List Utility) 254</td>
</tr>
<tr>
<td>validating changes, editor session 96</td>
</tr>
<tr>
<td>value list, scrambling data 80</td>
</tr>
<tr>
<td>VARGRAPHIC data type, description 361</td>
</tr>
<tr>
<td>VARCHAR data type</td>
</tr>
<tr>
<td>description 361</td>
</tr>
<tr>
<td>processing options 136</td>
</tr>
<tr>
<td>options for processing 136</td>
</tr>
<tr>
<td>VARGRAPHIC data type, description 361</td>
</tr>
<tr>
<td>variables, syntax diagrams xiii</td>
</tr>
<tr>
<td>VARYING 55</td>
</tr>
<tr>
<td>Varying length and nullable column options 530</td>
</tr>
<tr>
<td>varying-length columns, options for processing 136</td>
</tr>
<tr>
<td>VARYINGZ 55</td>
</tr>
<tr>
<td>VER line command packages 389</td>
</tr>
<tr>
<td>primary command 9, 781</td>
</tr>
</tbody>
</table>

**W**

<table>
<thead>
<tr>
<th>W</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wide print, print option 670</td>
</tr>
<tr>
<td>wildcards, using in FM/DB2 entry fields 28</td>
</tr>
<tr>
<td>working with object list panels 244</td>
</tr>
<tr>
<td>WRITE external function 219</td>
</tr>
</tbody>
</table>

**X**

<table>
<thead>
<tr>
<th>X</th>
</tr>
</thead>
<tbody>
<tr>
<td>X edit prefix command 695, 703</td>
</tr>
<tr>
<td>line command application plans 391</td>
</tr>
<tr>
<td>columns 413</td>
</tr>
<tr>
<td>databases 501</td>
</tr>
<tr>
<td>EXPLAIN plan table 616</td>
</tr>
<tr>
<td>packages 389</td>
</tr>
<tr>
<td>storage groups 684</td>
</tr>
<tr>
<td>tables, views and aliases 706</td>
</tr>
<tr>
<td>XC line command tables, views and aliases 706</td>
</tr>
<tr>
<td>XPT primary command storage groups 684</td>
</tr>
<tr>
<td>XX line command storage groups 684</td>
</tr>
</tbody>
</table>

**Z**

<table>
<thead>
<tr>
<th>Z</th>
</tr>
</thead>
<tbody>
<tr>
<td>ZCSR editor-assigned label 116</td>
</tr>
<tr>
<td>ZFST editor-assigned label 116</td>
</tr>
<tr>
<td>ZLST editor-assigned label 116</td>
</tr>
<tr>
<td>Zoom function key 125</td>
</tr>
<tr>
<td>ZOOM primary command 125, 785</td>
</tr>
<tr>
<td>zooming 123</td>
</tr>
</tbody>
</table>