



DB2 照会報告書作成プログラム

## DB2 QMF 解説書

バージョン 9 リリース 1





DB2 照会報告書作成プログラム

## DB2 QMF 解説書

バージョン 9 リリース 1

お願い

本書および本書で紹介する製品をご使用になる前に、313 ページの『付録 D. 特記事項』に記載されている情報をお読みください。

本書は、IBM DB2 (z/OS 版) 5635-DB2 バージョン 9 リリース 1 のフィーチャーである IBM DB2 照会報告書作成プログラム、および改訂版で特に断らない限り、それ以降のリリースおよび修正に適用されます。本書は、SC88-9834-00 (英文原典: SC18-7446-00) の改訂版です。

IBM 発行のマニュアルに関する情報のページ

<http://www.ibm.com/jp/manuals/>

こちらから、日本語版および英語版のオンライン・ライブラリーをご利用いただけます。また、マニュアルに関するご意見やご感想を、上記ページよりお送りください。今後の参考にさせていただきます。

(URL は、変更になる場合があります)

お客様の環境によっては、資料中の円記号がバックスラッシュと表示されたり、バックスラッシュが円記号と表示されたりする場合があります。

原 典： SC18-9685-00

DB2 Query Management Facility

DB2 QMF Reference

Version 9 Release 1

発 行： 日本アイ・ビー・エム株式会社

担 当： ナショナル・ランゲージ・サポート

第1刷 2007.3

この文書では、平成明朝体™W3、平成明朝体™W7、平成明朝体™W9、平成角ゴシック体™W3、平成角ゴシック体™W5、および平成角ゴシック体™W7を使用しています。この(書体\*)は、(財)日本規格協会と使用契約を締結し使用しているものです。フォントとして無断複製することは禁止されています。

注\* 平成明朝体™W3、平成明朝体™W7、平成明朝体™W9、平成角ゴシック体™W3、  
平成角ゴシック体™W5、平成角ゴシック体™W7

© Copyright International Business Machines Corporation 1982, 2007. All rights reserved.

© Copyright IBM Japan 2006

---

## 目次

<b>本書について</b>	. . . . .	xi
<b>本書の使い方</b>	. . . . .	xi
<b>前提条件および関連情報</b>	. . . . .	xi
<b>サポート情報</b>	. . . . .	xii
<b>アクセシビリティ機能</b>	. . . . .	xii
<b>ご意見の送付方法</b>	. . . . .	xii
 <b>第 1 章 QMF のコマンド</b>	. . . . .	1
<b>QMF コマンド環境</b>	. . . . .	2
<b>コマンドの入力</b>	. . . . .	3
<b>コマンド行からの入力</b>	. . . . .	3
<b>ファンクション・キーによる入力</b>	. . . . .	3
<b>プロンプト・パネルからの入力</b>	. . . . .	3
<b>プロシージャーからの入力</b>	. . . . .	5
<b>アプリケーションからの入力</b>	. . . . .	5
<b>リモート・データ・アクセスの使用</b>	. . . . .	6
<b>確認パネル</b>	. . . . .	6
<b>コマンドの取り消し</b>	. . . . .	7
<b>構文図の読み方</b>	. . . . .	8
<b>コマンドのパラメーター</b>	. . . . .	9
<b>ADD</b>	. . . . .	10
<b>BACKWARD</b>	. . . . .	10
<b>説明</b>	. . . . .	10
<b>注</b>	. . . . .	11
<b>BATCH</b>	. . . . .	12
<b>BOTTOM</b>	. . . . .	13
<b>CANCEL</b>	. . . . .	14
<b>CHANGE</b>	. . . . .	15
<b>CHECK</b>	. . . . .	15
<b>注</b>	. . . . .	15
<b>エラー状態</b>	. . . . .	16
<b>警告状態</b>	. . . . .	16
<b>CICS</b>	. . . . .	16
<b>説明</b>	. . . . .	17
<b>注</b>	. . . . .	17
<b>例</b>	. . . . .	17
<b>CLEAR</b>	. . . . .	18
<b>TSO における CONNECT</b>	. . . . .	18
<b>説明</b>	. . . . .	18
<b>注</b>	. . . . .	19
<b>例</b>	. . . . .	19
<b>QMF プロシージャーでの CONNECT コマンドの使用</b>	. . . . .	20
<b>分散ネットワーク内の DB2 データベースへの接続</b>	. . . . .	20
<b>新規のロケーションとの接続が、長い名前のサポートに与える影響</b>	. . . . .	20
<b>CONNECT (CICS の場合)</b>	. . . . .	21
<b>説明</b>	. . . . .	21
<b>注</b>	. . . . .	22
<b>例</b>	. . . . .	23
<b>CONVERT</b>	. . . . .	23

説明 . . . . .	24
注 . . . . .	25
例 . . . . .	26
DELETE . . . . .	27
DESCRIBE . . . . .	27
DISPLAY . . . . .	28
説明 . . . . .	29
注 . . . . .	30
例 . . . . .	31
DPRE . . . . .	31
DRAW . . . . .	32
説明 . . . . .	32
注 . . . . .	33
例 . . . . .	33
EDIT オブジェクト . . . . .	34
説明 . . . . .	34
注 . . . . .	35
例 . . . . .	35
EDIT TABLE . . . . .	35
説明 . . . . .	36
注 . . . . .	37
例 . . . . .	37
END . . . . .	37
ENLARGE . . . . .	39
ERASE . . . . .	39
説明 . . . . .	39
注 . . . . .	40
例 . . . . .	40
EXIT . . . . .	41
EXPORT (CICS の場合) . . . . .	42
説明 . . . . .	46
注 . . . . .	48
例 . . . . .	49
EXPORT (TSO の場合) . . . . .	50
説明 . . . . .	54
注 . . . . .	57
例 . . . . .	58
FORWARD . . . . .	59
説明 . . . . .	60
注 . . . . .	60
GET GLOBAL . . . . .	60
説明 . . . . .	61
注 . . . . .	61
GETQMF マクロ . . . . .	61
HELP . . . . .	62
説明 . . . . .	62
注 . . . . .	63
IMPORT (CICS の場合) . . . . .	63
説明 . . . . .	65
注 . . . . .	67
例 . . . . .	68
IMPORT (TSO の場合) . . . . .	68

説明 . . . . .	71
注 . . . . .	72
例 . . . . .	73
INSERT . . . . .	73
INTERACT . . . . .	74
ISPF . . . . .	75
LAYOUT . . . . .	75
説明 . . . . .	75
注 . . . . .	75
例 . . . . .	76
LEFT . . . . .	77
説明 . . . . .	77
注 . . . . .	77
LIST . . . . .	78
説明 . . . . .	78
注 . . . . .	79
例 . . . . .	81
MESSAGE . . . . .	81
説明 . . . . .	82
注 . . . . .	83
例 . . . . .	83
NEXT . . . . .	84
説明 . . . . .	84
注 . . . . .	84
PREVIOUS . . . . .	85
説明 . . . . .	85
注 . . . . .	85
PRINT (TSO の場合) . . . . .	85
説明 . . . . .	89
注 . . . . .	92
例 . . . . .	93
PRINT (CICS の場合) . . . . .	93
説明 . . . . .	98
注 . . . . .	101
例 . . . . .	103
QMF . . . . .	103
説明 . . . . .	103
注 . . . . .	103
例 . . . . .	103
REDUCE . . . . .	104
REFRESH . . . . .	104
RESET GLOBAL . . . . .	104
説明 . . . . .	104
注 . . . . .	105
例 . . . . .	105
RESET オブジェクト . . . . .	105
説明 . . . . .	107
例 . . . . .	108
RETRIEVE . . . . .	109
説明 . . . . .	109
注 . . . . .	109
例 . . . . .	110

RIGHT . . . . .	110
説明 . . . . .	111
注 . . . . .	111
RUN . . . . .	111
説明 . . . . .	113
注 . . . . .	115
Run コマンドの変数値 . . . . .	116
システム考慮事項 . . . . .	117
例 . . . . .	117
SAVE . . . . .	118
説明 . . . . .	120
注 . . . . .	121
例 . . . . .	122
SEARCH . . . . .	123
SET GLOBAL . . . . .	123
説明 . . . . .	124
注 . . . . .	124
例 . . . . .	126
SET PROFILE . . . . .	126
説明 . . . . .	127
注 . . . . .	130
例 . . . . .	131
SHOW . . . . .	131
説明 . . . . .	133
Detail Variation . . . . .	134
注 . . . . .	134
SORT . . . . .	135
SPECIFY . . . . .	135
説明 . . . . .	135
注 . . . . .	136
START . . . . .	136
QMF プログラム・パラメーター . . . . .	137
説明 . . . . .	139
STATE . . . . .	139
SWITCH . . . . .	140
TOP . . . . .	140
TSO . . . . .	140
説明 . . . . .	141
注 . . . . .	141
例 . . . . .	141
<b>第 2 章 QMF 照会で使用される基本的な SQL ステートメントと関数 . . . . .</b>	<b>143</b>
ADD . . . . .	144
ALL . . . . .	144
ALTER TABLE . . . . .	145
AND . . . . .	145
ANY . . . . .	146
AS . . . . .	146
AVG . . . . .	147
BETWEEN x AND y . . . . .	147
CALL . . . . .	148
パラメーターの使用方法 . . . . .	149

CALL ステートメントの使用のガイドライン . . . . .	149
長い ID を伴う CALL ステートメントの作成方法 . . . . .	150
使用する結果セットを示す大域変数 . . . . .	150
QMF 書式の使用 . . . . .	150
COUNT . . . . .	151
CREATE SYNONYM . . . . .	152
CREATE TABLE . . . . .	152
CREATE VIEW . . . . .	154
DELETE . . . . .	155
DISTINCT . . . . .	156
DROP . . . . .	157
EXISTS . . . . .	158
GRANT . . . . .	158
GROUP BY . . . . .	159
HAVING . . . . .	161
IN . . . . .	163
INSERT . . . . .	163
行に列値を挿入する . . . . .	164
表から表へ行をコピーする . . . . .	164
IS . . . . .	165
LIKE . . . . .	165
文字ストリングの選択 . . . . .	165
特定の文字の無視 . . . . .	166
MAX および MIN . . . . .	167
NOT . . . . .	167
NULL . . . . .	168
OR . . . . .	169
ORDER BY . . . . .	170
ソート・シーケンス . . . . .	170
複数列による配列 . . . . .	171
列番号による配列 . . . . .	172
REVOKE . . . . .	172
SELECT . . . . .	173
表から全列を選択する . . . . .	173
表から列を選択する . . . . .	174
記述列を追加する . . . . .	174
副照会の使用 . . . . .	175
例: . . . . .	175
SOME . . . . .	175
SUM . . . . .	176
UNION . . . . .	177
UNION で重複行を残す . . . . .	178
UNION の使用規則 . . . . .	179
いつ UNION を使用するか、いつ表を結合するか . . . . .	180
UPDATE . . . . .	180
WHERE . . . . .	181
計算結果 . . . . .	184
@IF 関数 . . . . .	185
SQL スカラー関数 . . . . .	187
日付 / 時刻関数 . . . . .	187
他の変換関数 . . . . .	188
ストリング関数 . . . . .	189

連結	190
<b>第 3 章 書式、報告書、および図表</b>	
QMF 書式の使用	191
QMF における報告書の作成	192
データのない報告書の表示	192
報告書で用いられるエラー記号	192
報告書のフォーマット設定の早見表	193
QMF における図表の作成	194
FORM.MAIN	195
FORM.BREAKn	198
FORM.CALC	206
FORM.COLUMNS	211
列属性を指定する	216
印刷の考慮事項	221
FORM.CONDITIONS	221
FORM.DETAIL	223
FORM.FINAL	229
FORM.OPTIONS	234
FORM.PAGE	240
QMF の書式エラーの評価	246
エラー状態	246
警告状態	246
誤りの検査とその訂正	247
書式とデータの不整合	247
QMF 書式での REXX の使用	248
報告書での計算値の使用	249
QMF と REXX の対話	249
REXX はいつ式を評価するか	250
REXX 演算子	251
報告書計算式の例	253
取扱コード	254
ACROSS 取扱コード	254
集約取扱コード	254
BREAK 取扱コード	259
CALCid 取扱コード	260
GROUP 取扱コード	260
ブランクおよび OMIT 取扱コード	261
日付 / 時刻取扱コード	261
編集コード	261
文字データの編集コード	262
文字または 2 進データの編集コード	264
グラフィック・データの編集コード	264
数値データの編集コード	264
日付データの編集コード	265
時刻データの編集コード	267
タイム・スタンプ・データの編集コード	267
QMF が列メタデータを表示するためのデータ・タイプ	268
ユーザー定義編集コード	268
総計機能および編集コードの考慮事項	268
書式で使用される変数	269

<b>第 4 章 一般事項</b>	271
命名規則	271
2 バイト文字を含む名前	272
小数点に代えてコンマでの 10 進数のフォーマット設定	272
QMF 一時記憶域	273
報告書の完成と不完全データ指示	274
長時間を要する照会に対する QMF の応答の変更	275
QMF オブジェクトの編集時におけるデータとしてのヌルの使用の回避	275
照会の書き方	276
SQL	276
指示照会 (Prompted Query)	276
例示照会 (QBE)	276
プロシージャー	276
ロジックを持つプロシージャー	277
線形手順	278
オブジェクトの印刷	279
表エディター	279
オンライン・ヘルプ	281
トピック・ヘルプ	281
メッセージ・ヘルプ	281
フィールド別ヘルプ	281
リモート・データ・アクセス	281
分散作業単位 (DB2 (z/OS 版) のみ)	282
リモート作業単位	282
管理プログラムによる割り込み	283
<b>付録 A. QMF のサンプル表</b>	285
Q.APPLICANT	285
Q.INTERVIEW	286
Q.ORG	286
Q.PARTS	287
Q.PRODUCTS	287
Q.PROJECT	288
Q.SALES	289
Q.STAFF	290
Q.SUPPLIER	291
<b>付録 B. QMF 大域変数表</b>	293
大域変数の命名規則	293
プロファイル関連状態情報の大域変数	294
プロファイル関連でない状態情報の大域変数	295
CICS 関連の大域変数	299
最後に発行されたコマンドで生成されたメッセージに関連する大域変数	299
表エディター関連の大域変数	300
さまざまな表示を制御する大域変数	302
コマンドとプロシージャーの実行を制御する大域変数	305
CONVERT QUERY の結果を保管する大域変数	309
RUN QUERY エラー・メッセージ情報を示す大域変数	309
<b>付録 C. 特定のサポートを必要とする QMF 機能</b>	311
<b>付録 D. 特記事項</b>	313
商標	314

用語集	315
索引	325

---

## 本書について

DB2® 照会報告書作成プログラム (QMF™) は、データベースのデータの作成または変更、あるいはデータベースからのデータの取得を簡単に行うためのデータベース・アプリケーション・プログラムです。データを取り出したあと、それを報告書または図表にフォーマット設定することができます。

次のトピックを使用すると、ここに提供された参照情報の使用方法がよく分かります。

- ・『本書の使い方』
- ・『前提条件および関連情報』
- ・xii ページの『サポート情報』
- ・xii ページの『アクセシビリティ機能』
- ・xii ページの『ご意見の送付方法』

---

## 本書の使い方

本書は、照会報告書作成プログラムを使用したことのある人を対象としています。本書で説明している主要なトピックは次のとおりです。

- ・QMF コマンド
- ・QMF 照会で使用される SQL キーワード
- ・書式、報告書および図表 (取扱コードと編集コードを含む)

本書に掲載した例の多くは、DB2 (VM 版) データベースに QMF を使用して作成しました。VM での QMF サポート・レベルの最高は、バージョン 7.2 です。したがって、TSO または CICS® 環境の QMF バージョン 9.1 で表示される結果は、いくらか異なることがあります。

---

## 前提条件および関連情報

「*DB2 QMF 使用の手引き*」の中のトピックには、基本的な QMF 情報が含まれており、本書の参照情報では、読者はそこに記載されたタスクおよび概念についての知識が十分にあることを前提にしています。QMF を使い始めるため、および SQL 照会の書き方を知るために必要なステップに加えて、「*DB2 QMF 使用の手引き*」には、照会とフォームの作成方法をステップを追って示した詳しいシナリオがいくつか記載されています。また、例示照会 (Query-By-Example) に関する情報も記載されています。QMF の資料は、IBM Publications Center ([www.ibm.com/shop/publications/order](http://www.ibm.com/shop/publications/order)) または、次の QMF Web サイトのページから入手できます。

<http://www.ibm.com/software/data/qmf/library.html>

---

## サポート情報

サポート情報 (PTF、よくある質問 (FAQ)、技術情報、トラブルシューティング情報、およびダウンロードを含む) は、次の Web ページを参照してください。

[www.ibm.com/software/data/qmf/support.html](http://www.ibm.com/software/data/qmf/support.html)

---

## アクセシビリティ機能

アクセシビリティ機能は、運動障害または視覚障害など身体に障害を持つユーザーがソフトウェア・プロダクトを快適に使用できるようにサポートします。 QMF のアクセシビリティの主要機能により、ユーザーは次のことができます。

- 画面読み上げ機能および画面拡大機能などの支援機能の使用この支援機能を使用して z/OS® インターフェースにアクセスする場合、その特定情報については支援機能の資料を参照してください。
- 色、コントラスト、フォント・サイズなど表示属性をカスタマイズする。
- 特定の機能または同等の機能をキーボードのみを使用して操作する。ISPF インターフェースのアクセスに関する情報は、次の資料を参照してください。
  - 対話式システム生産性向上機能 (ISPF) ユーザーズ・ガイド 第 1 卷 z/OS VIR6.0、SC88-8965
  - z/OS TSO/E 入門、SA88-8632
  - z/OS TSO/E ユーザーズ・ガイド、SA88-8638

上記の資料では、キーボード・ショートカットまたはファンクション・キー (PF キー) の使用をはじめとする ISPF の使用方法に加えて、PF キーのデフォルトの設定値、およびそれらの機能の変更方法についても説明しています。

---

## ご意見の送付方法

本マニュアルに関するご意見やご感想は、次の URL からお送りください。

<http://www.ibm.com/jp/manuals/>

---

# 第 1 章 QMF のコマンド

この章には、以下の QMF コマンドの情報が含まれています。

- 2 ページの『QMF コマンド環境』
- 3 ページの『コマンドの入力』
- 6 ページの『確認パネル』
- 7 ページの『コマンドの取り消し』
- 8 ページの『構文図の読み方』
- 9 ページの『コマンドのパラメーター』
- 10 ページの『ADD』
- 10 ページの『BACKWARD』
- 12 ページの『BATCH』
- 13 ページの『BOTTOM』
- 14 ページの『CANCEL』
- 15 ページの『CHANGE』
- 15 ページの『CHECK』
- 16 ページの『CICS』
- 18 ページの『CLEAR』
- 18 ページの『TSO における CONNECT』
- 21 ページの『CONNECT (CICS の場合)』
- 23 ページの『CONVERT』
- 27 ページの『DELETE』
- 27 ページの『DESCRIBE』
- 28 ページの『DISPLAY』
- 31 ページの『DPRE』
- 32 ページの『DRAW』
- 34 ページの『EDIT オブジェクト』
- 35 ページの『EDIT TABLE』
- 37 ページの『END』
- 39 ページの『ENLARGE』
- 39 ページの『ERASE』
- 41 ページの『EXIT』
- 42 ページの『EXPORT (CICS の場合)』
- 50 ページの『EXPORT (TSO の場合)』
- 59 ページの『FORWARD』
- 60 ページの『GET GLOBAL』
- 61 ページの『GETQMF マクロ』
- 62 ページの『HELP』
- 63 ページの『IMPORT (CICS の場合)』

## QMF コマンド

- 68 ページの『IMPORT (TSO の場合)』
- 73 ページの『INSERT』
- 74 ページの『INTERACT』
- 75 ページの『ISPF』
- 75 ページの『LAYOUT』
- 77 ページの『LEFT』
- 78 ページの『LIST』
- 81 ページの『MESSAGE』
- 84 ページの『NEXT』
- 85 ページの『PREVIOUS』
- 85 ページの『PRINT (TSO の場合)』
- 93 ページの『PRINT (CICS の場合)』
- 103 ページの『QMF』
- 104 ページの『REDUCE』
- 104 ページの『REFRESH』
- 104 ページの『RESET GLOBAL』
- 105 ページの『RESET オブジェクト』
- 109 ページの『RETRIEVE』
- 110 ページの『RIGHT』
- 111 ページの『RUN』
- 118 ページの『SAVE』
- 123 ページの『SEARCH』
- 123 ページの『SET GLOBAL』
- 126 ページの『SET PROFILE』
- 131 ページの『SHOW』
- 135 ページの『SORT』
- 135 ページの『SPECIFY』
- 136 ページの『START』
- 139 ページの『STATE』
- 140 ページの『SWITCH』
- 140 ページの『TOP』
- 140 ページの『TSO』

---

## QMF コマンド環境

QMF コマンドは、TSO、または CICS 環境から入力することができます。TSO では、ISPF の使用も可能です。各コマンド説明の最初に記載してある表には、そのコマンドを使用できる環境が X で示されています。アスタリスク (\*) は、そのコマンドの特定の機能だけが使用できることを表します。たとえば、

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X	X	*

コマンドを入力できる環境が 1 つしかない場合は、その環境名がトピック・タイトルに含まれ、表は表示されません。

## コマンドの入力

QMF コマンドは、次に挙げる方法で入力できます。

- コマンド行からの入力
- ファンクション・キーによる入力
- プロンプト・パネルからの入力
- プロシージャーからの入力
- アプリケーションからの入力

使用しているサイトで QMF コマンドと同じ名前のコマンド同義語が定義されている場合は、その同義語より優先させるために、コマンドの前に QMF を付ける必要があります。

### コマンド行からの入力

コマンド行の表示がある画面の場合には、矢印の後に QMF コマンドをフル・スペルでタイプします。たとえば、

```
COMMAND ==> RUN MYQUERY (FORM=FORM2
```

このコマンドを実行するには、Enter を押します。

### ファンクション・キーによる入力

コマンドによっては、ファンクション・キーで入力できるコマンドがあります。

QMF では、パネルごとにデフォルトのファンクション・キー・セットを用意しています。 QMF を使用する時点で、ファンクション・キーがデフォルト設定と異なる場合がありますが、本書ではファンクション・キーのデフォルト設定を用いて説明しています。

ファンクション・キーに割り当てられたコマンドにパラメーターが必要な場合には、まずパラメーターをコマンド行にタイプしてから、ファンクション・キーを押してください。たとえば、照会パネルが表示されているときに (FORM=FORM2 と入力してから Run ファンクション・キーを押します。次のコマンドが実行されます。

```
RUN QUERY (FORM=FORM2
```

### プロンプト・パネルからの入力

構文エラーのあるコマンドを入力するか（またはつづりの誤りがあるコマンドを連続して 2 回入力する）、あるいはコマンド行にコマンド名に続けて疑問符を入力すると、QMF はコマンド・プロンプト・パネルを表示します。長いコマンドを入力する場合には、このプロンプト・パネルを使用すると便利です。

## QMF コマンド

たとえば、RUN ? と入力すると、図 1 に示したコマンド・プロンプト・パネルが表示されます。ここに必要な情報を入力できます。

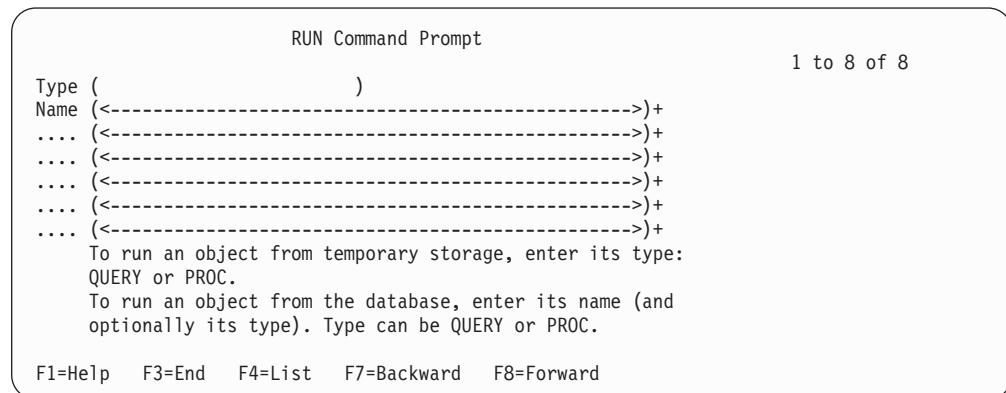


図 1. 「RUN コマンド・プロンプト」パネル

コマンドを完成させるのにさらに情報が必要である場合には、QMF は 2 番目のパネルでプロンプトを表示して、コマンド・パラメーターの入力を求めます。

この 2 ステップのプロンプトで最初のパネルを省略したいときは、コマンド行にコマンド名、オブジェクト・タイプ、オブジェクト名を入力し、その後に疑問符を入力します。指定されたオブジェクトに適用できるパラメーターだけを含んだパネルが表示されます。

コマンドのパラメーター部分 (左括弧の後) に疑問符を使用してはなりません。疑問符の後に入力されたパラメーターは無視されます。たとえば、次のコマンドの「(FORM=FORM2」は無視されます。

RUN QUERY MYQUERY? (FORM=FORM2

次のファンクション・キーがほとんどのプロンプト・パネルに現れます。

### Help (ヘルプ)

このキーは、表示されているメッセージについてのヘルプ情報を、画面の一番下に表示します。

### List (リスト)

オブジェクト・リストを表示します。そのリストから所要のオブジェクトを選択できます

### End (終了)

プロンプトが出されたパネルに戻ります

多くの QMF コマンドは長いオブジェクト名をサポートします。このようなコマンドの「name」オプションでは、「Location.Owner.Name」タイプの長い名前を指定できます。ここで、Location (データベース・サーバー名) は最長 16 バイトまでで、Owner (ユーザーのデータベース許可 ID) と Name (オブジェクト名) は、それぞれ最長 128 バイトまでです。これらのコマンドで指定するフルのオブジェクト名は、最長 280 バイトにまでできます。

## プロシージャーからの入力

他のプロシージャーを実行する RUN コマンドも含め、ほとんどの QMF コマンドは、プロシージャーの中に 1 行として含めることができます。コマンド行から入力するにはコマンドが長すぎる場合には、この方法が便利です。

コマンドをプロシージャーの中に組み入れる場合には、コマンドは省略形ではなく、コマンド名、パラメーター、および値をいずれもフル・スペルで指定してください。ある語について現在認められている省略形は、将来のリリースで変更される可能性があります。その場合、現在の省略形を用いているプロシージャーは、実行できなくなることがあります。

## ロジックを持つプロシージャー

ロジックを持つプロシージャーで QMF コマンドを使用する場合、そのコマンドは、

- プロファイルの設定に関係なく、必ず大文字でなければなりません
- コンマで現在の行を終了することにより、次の行へ継続できます
- 置換変数を含むことができます

## 線形手順のコマンド

線形手順のコマンドは、継続する各行の 1 桁目に継続文字として正符号 (+) を入れることにより、複数行に継続できます。継続行自体は 2 桁目から始めます。

図 2 に示すように、複数行にわたるオブジェクト名、許可 ID、またはロケーションは、二重引用符 (区切り ID) で囲む必要があります。

```
PROC MODIFIED LINE 1
  ERASE QUERY
  +"LOCATION12345678"."LONGOWNERID12345678912123456789312345678941234567
  +123456789112345678921234567893123456789412345678951234567896123456789712345"
  +LONGNAME1234567891123456789212345678931234567894123456789512345678961234567897
  +1234567891123456789212345678931234567894123456789712345"
```

図 2. 線形手順での複数行にわたる修飾オブジェクト名の継続

LIST コマンドを使用する場合は、单一引用符を使用してください。詳細については、78 ページの『LIST』を参照してください。

詳細については、276 ページの『プロシージャー』を参照してください。

## アプリケーションからの入力

アプリケーション内で使用する QMF コマンドは、プロファイルでの設定に関係なく、大文字でなければなりません。

### コマンド・インターフェース

このインターフェースは、ISPF から QMF を受け取ります。アプリケーション、または CLIST の実行よりも前に、QMF を開始する必要があります。コマンド・インターフェースの機能は ISPF に依存するので、CICS では使用できません。

### 呼び出し可能インターフェース

QMF の共通プログラミング・インターフェース (CPI) から QMF コマンドを直接受け取ります。QMF は、アプリケーションから、開始および終了を行うことができます。ISPF は必要ありません。

アプリケーションでのコマンドの使用に関する詳細については、「*DB2 QMF アプリケーション開発の手引き*」を参照してください。

## リモート・データ・アクセスの使用

分散作業単位またはリモート作業単位を用いてコマンドを出す場合には、

- 表と視点への参照は、3 部分名または別名によって特定のロケーションを参照していないかぎり、現行ロケーションへの参照となります。しかし、コマンドで 3 部構成の名前を使用し、データベース管理者が、複数行フェッチ機能を使用するように QMF をセットアップしている場合は、操作しているデータベースは両方とも DB2 (z/OS 版) 8.1.5 以降でなければなりません。そうでなければ、データの表示または挿入のコマンドは失敗します。データベース管理者は、必要であればこの機能をオフにできます。詳細は、「*TSO/CICS 用 DB2 QMF インストールおよび管理の手引き*」を参照してください。
- データベース内の QMF プロシージャー、照会、書式への参照は、現行ロケーションへの参照となります。プロシージャー、照会、書式を 3 部分名で参照することはできません。
- QMF コマンドで指定するデータ・セットやファイルは、QMF が稼働しているシステムになければなりません。
- QMF コマンドで指定する CICS データ・キューは、QMF が稼働しているシステムで定義されたものでなければなりません。
- 保管されているプロファイル値への参照は、現行ロケーションに対して適用されます。ただし、TRACE パラメーターを除きます。
- QMF が CICS で稼働している場合、リモート DB2 のロケーションにあるすべてのデータベース・オブジェクト (表、視点、プロシージャー、照会、書式) は、読み取り専用となります。

---

## 確認パネル

コマンドに CONFIRM パラメーターがあるときは、そこに YES または NO を指定することができます (プロファイル中のデフォルト値を使用することもできます)。データベースに変更を加えるコマンドの場合、CONFIRM パラメーターを YES に指定すると、7 ページの図 3 に示すような確認パネルが表示されます。

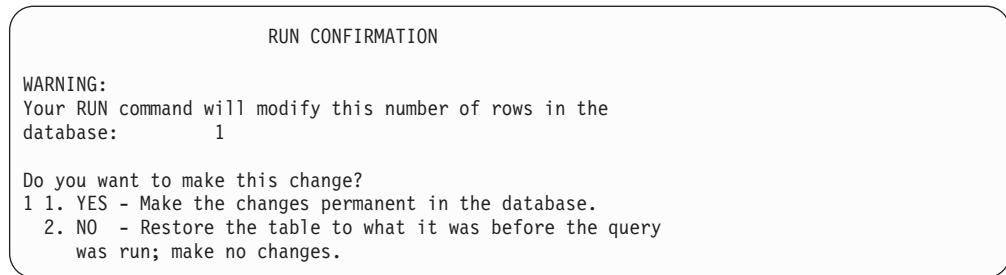


図 3. 確認パネルの例

データベース変更の際に表示される QMF 確認パネルの多くは、実際には、その変更をコミットする (YES) のか、またはロールバックする (NO) のかの確認を求めるプロンプトを出します。

データベースに対してすでに変更が加えられているため、データベース・マネージャーは、確認パネルに YES または NO の応答があるまでそのデータをロックします。

DB2 Server (VM または VSE 版) に接続している場合には、扱う表が回復不能な DB スペースに置かれていることがあります。その場合には、加えた変更が直ちにデータベースにコミットされるため、ロールバックを実行できません。したがって、表が回復不能な DB スペースにあるときは、確認パネルで NO を指定しても、データベース変更は回避できません。

DB スペースの詳細は、QMF 管理者に問い合わせるか、IBM Publications Center ([www.ibm.com/shop/publications/order](http://www.ibm.com/shop/publications/order)) にある、DB2 Server (VM または VSE 版) の管理情報を参照してください。

## コマンドの取り消し

現在処理中の QMF コマンドまたは照会を取り消す方法は、端末接続と環境によって異なります。

TSO の場合:

- 端末がシステムに直接接続されている場合には、まず RESET キーを押し、次に PA1 キーを押します。
- 端末がネットワークに接続されている場合は、ATTN キーを押します。
- 端末エミュレーターを使用して稼働環境をシミュレートしている場合は、取り消すセッションのポップアップ・メニューを表示します。このメニューに、PA1 および ATTN キーがあります。

CICS の場合、他の CICS トランザクション同様、CICS オペレーターが QMF トランザクションを取り消さなければなりません。CICS で PA1 キーや ATTN キーを使用することはできません。QMF トランザクションが取り消されると、すべての作業は失われます。

## 構文図の読み方

本書で使用されている構文図では、以下の規則が適用されます。

- 構文図は線の経路に沿って、左から右、上から下へ読んでいきます。

記号 **►——** は、ステートメントの開始を示します。

記号 **——→** は、ステートメント構文が次の行に継続することを示します。

記号 **——▶** は、ステートメントが直前の行から継続していることを示します。

記号 **→◀** はステートメントの終了を示します。

完全なステートメントではない構文単位の場合、図は記号 **——▶** で開始し、記号 **→** で終了します。

- コマンドは、必ず構文図の主経路上に示されます。コマンドとパラメーターを省略形で最小にした場合は、大文字で示されます。変数は小文字のイタリック体(斜体)で表示されます(例、*column-name*)。変数はユーザー定義のパラメータ一、または下位のオプションを表します。

コマンドを入力する場合、句読点が間に入らないときは、パラメーターやキーワードを少なくとも 1 つのスペースで分離します。

- 句読記号(スラッシュ、コンマ、ピリオド、括弧、引用符、等号など)や数値は示されているとおりに正確に入力します。
- 脚注は括弧付きの番号で、たとえば (1) のように、表示されます。
- 必須項目は水平線、すなわち主経路上に表示されます。

**►——required\_item——▶◀**

- オプション項目は主経路より下に表示されます。

**►——required\_item——[optional\_item]——▶◀**

- 複数の項目から選択できる場合、項目は縦方向に並べられて表示されます。

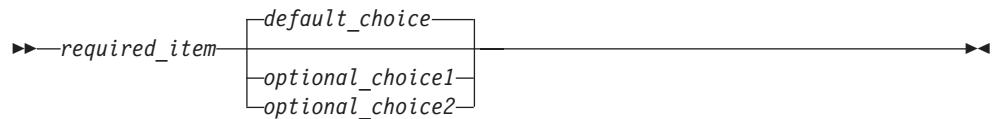
項目のうちの 1 つを選択しなければならない場合、縦に並べられた項目のうちの 1 つは主経路上に表示されます。

**►——required\_item——[required\_choice1——required\_choice2]——▶◀**

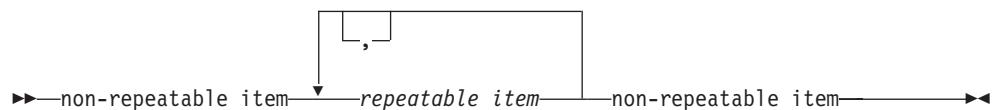
項目のうちの 1 つを選択することがオプションである場合、縦に並べられた項目はすべて主経路より下に表示されます。

**►——required\_item——[optional\_choice1——optional\_choice2]——▶◀**

項目のうちの 1 つを選択することがオプションであり、デフォルト値が 1 つある場合には、そのデフォルト値が主経路より上に表示されます。



- 項目が反復できる項目である場合には、左矢印が図でのループを示します。オプションとして、項目の間をコンマで区切ることができます。



## コマンドのパラメーター

コマンドで使用されるパラメーターには 2 つのタイプがあります。定位置パラメーターは、常にコマンド内の特定の位置に指定されるパラメーターです。キーワード・パラメーターは、値を与えられるパラメーターであり、コマンド内での指定順序は自由です。コマンドに指定する最初のキーワード・パラメーターの前には、左括弧が必要です。

キーワード・パラメーターの指定が可能なコマンドでは、必要なだけいくつでも指定できます。あるキーワード・パラメーターを同一コマンドで何回も使用し、そのたびに異なる値を指定した場合は、最後の値が有効になります。どのようなパラメーター値も、80 文字を超えることはできません。

パラメーターとパラメーターとの間の区切りには、1 つのブランク、1 つのコンマ、あるいは、1 つのコンマとそれに続けてオプションの 1 つのブランクのいずれかを使用します。たとえば、次の指定はいずれも有効です。

```
(MEMBER=member CONFIRM=YES
(MEMBER=member, CONFIRM=YES
(MEMBER=member,CONFIRM=YES
(MEMBER member CONFIRM=YES
(MEMBER member CONFIRM YES
```

右括弧は不要ですが、コマンドを終了させるために使用することもできます。右括弧の後に指定した内容はすべてコメントと見なされ、処理されません。

## ADD

### ADD

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X	X	X

ADD コマンドは、

- 表エディターで表に行を追加します
- 大域変数リストに大域変数を追加します

►—Add—►

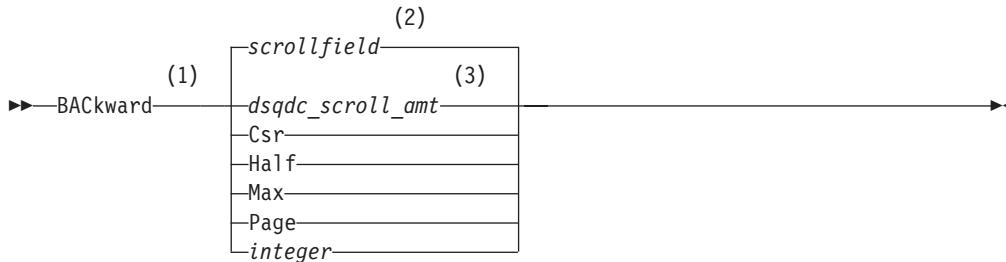
#### 注:

- 表エディターでは、EDIT コマンドの SAVE オプションの指定方法に応じて、トランザクションをただちに保管させることも、表編集セッションの終了時に保管させることもできます。
- 大域変数リストでは、ADD コマンドが「変数の追加」パネルを表示し、新しい変数を追加することができます。

## BACKWARD

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X	X	X

BACKWARD コマンドは、アクティブ・パネルの上方へスクロールします（表エディターを使用している場合は、現在行の最初のフィールドの方へスクロールします）。パネルで、カーソル位置またはページの先頭までスクロールして戻るか、半ページ、1 ページ、または指定の行数分、上方にスクロールできます。



#### 注:

- 活動中のパネルに画面移動フィールドがある場合にのみ画面移動量を指定します。その他のすべての場合は、PAGE が想定されます。
- 画面移動フィールドに表示される値が使用されます。この値は、大域変数 DSQDC\_SCROLL\_AMT にも保持されます。
- この大域変数で設定された値が使用されます。

## 説明

**CSR** カーソルが位置する行をスクロール可能域のいちばん下まで移動させます

**HALF** スクロール可能域の半分だけ上方へスクロールするか、(それ以前に先頭に到達するときは) 先頭までスクロールします。

**MAX** スクロール可能域の先頭までスクロールします

**PAGE** スクロール可能域分だけ上方へスクロールするか、(それ以前に先頭に到達するときは) 先頭までスクロールします。

**integer**

この数字の行数分だけパネル上で上方にスクロールします。数字は 9999 までの正の整数です

## 注

- MAX は現行のコマンドの間だけ有効です。この値はコマンドが完了すると SCROLL フィールドには残りません。大域変数 DSQDC\_SCROLL\_AMT をこの値に設定することはできません。
- 書式パネルの後書きテキストで上方スクロールを行なうには、後書きテキストが存在するパネル部分にカーソルを合わせてから、BACKWARD コマンドを入力してください。
- QMF が使用するスクロール量は、大域変数 DSQDC\_SCROLL\_AMT を Csr、Half、Page、または 9999 までの正の整数に設定することによっても変更することができます。

## BATCH

### BATCH

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X		

BATCH は、QMF バッチ・アプリケーションにアクセスする、QMF が提供するコマンド同義語です。この適用業務で、照会や手順を対話式ではなく QMF バッチ・ジョブとして実行できます。

►►BATch►►

QMF BATCH コマンドは、表 1 に示した長さのオブジェクト名をサポートします。

表 1. QMF BATCH コマンドでのオブジェクト名のフィールド長

フィールド名	最大長
オブジェクト名 (照会またはプロシージャーの名前)	77
フォーム名	77
バッチ名 (QMF バッチ・プロシージャーの名前)	31
保管データ (保管されるデータの名前)	77

BATCH コマンドでは長い変数の入力もできます。このような変数の入力に、スクロール標識を利用することもできます。スクロール標識は、次のようなものです。

< > 31 60

左右の不等号は方向を示し、数字は、開始および終了位置を示します。

---

**BOTTOM**

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X	X	X

BOTTOM は、照会、プロシージャー、報告書、大域変数リスト、スクロール可能書式パネルのそれぞれで、最終行までスクロールします。

►►—Botton—►►

**注:**

- BOTTOM は FORWARD MAX と同じです。
- 書式パネルで後書きテキストの末尾までスクロールするには、後書きテキストが配置されているパネル部分にカーソルを合わせてから、BOTTOM コマンドを入力してください。

## CANCEL

### CANCEL

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X	X	X

CANCEL コマンドは、

- 表編集セッションでまだコミットされていない変更を廃棄します
- ヘルプ・パネルから基本の QMF パネルに戻ります
- コマンドの確認パネルを取り消します。確認パネルから Cancel (取り消し) ファンクション・キーを押すと、確認を求められているアクションのコマンドが取り消され、そのコマンドを入力した「QMF」パネルに戻ります。

►►—CAncel—►►

#### 注:

- CANCEL コマンドはファンクション・キーとしてのみ使用可能です。CANCEL ファンクション・キーは、表エディター、QMF ヘルプ・パネル、確認パネルで使用できます。
- 以下のような EDIT TABLE コマンドの SAVE オプションの値に応じて、表エディター・セッションで CANCEL を使用できる場合とできない場合があります。
  - SAVE=END の場合、取り消し機能キーが押されると変更は廃棄されます。
  - SAVE=IMMEDIATE の場合、CANCEL は受け入れられません。

---

**CHANGE**

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X	X	X

CHANGE コマンドは、指示照会では変更用のパネルを表示します。

表エディターでは、CHANGE コマンドは、表または視点に含まれる行を変更します。

►►—CHAnge—►►

**注:**

- 指示照会では、次のいずれかの方法を用いて変更を行うことができます。
  - 変更したい項目にカーソルを合わせ、Change (変更) ファンクション・キーを押す。
  - コマンド行で CHANGE と入力し、変更する項目にカーソルを移動して、ENTER を押します。
- 表エディターでは、Change (変更) ファンクション・キーを押すと、
  - SAVE=IMMEDIATE であれば、トランザクションが処理されると同時に変更が保管されます
  - SAVE=END であれば、END コマンドが処理される時に変更が保管されます

---

**CHECK**

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X	X	X

CHECK コマンドは、書式パネルに誤りがないか検査します。

►►—CHEck—►►

**注**

- 書式パネルが表示されているときに、コマンド行に CHECK を入力するか、Check ファンクション・キーを押すことができます。QMF は表示されたパネルの検出可能エラーを検査し、残っている書式パネルを検査します。
- メッセージ行は、他のエラーを表示する前に訂正する必要のあるエラーを説明します。
- 1 つのエラーが表示されている場合、表示中のエラーを訂正してから検査ファンクション・キーを押すことにより、その他のエラーを表示することができます。
- CHECK はエラーをすべて検出できるわけではありません。報告書を表示して初めて明らかになるエラーもあり、その時点で QMF がエラー・メッセージを表示します。

## CHECK

### エラー状態

書式パネルにエラーがあると、QMF は最初にエラーが発生したパネルを表示し、そのパネルのいちばん上に ERROR と表示します。 QMF は、エラーを含む書式パネルが 1 つだけでも、すべての書式パネルに ERROR が表示されます。エラーを含む入力域が強調表示され、その隣にカーソルが位置づけられます。どのようなエラーかを説明するメッセージがメッセージ行に現われます。

指摘されたエラーを訂正しないと、次のエラーを見ることも、報告書を作成することもできません。エラーの詳細と、それを訂正するのに行う必要があることを知るには、Help (ヘルプ) ファンクション・キーを押してください。次のエラーを識別するには、再び CHECK コマンドを入力し、そのエラーを訂正してください。これを繰り返して、すべてのエラーを訂正します。

FORM.CALC または FORM.CONDITIONS にエラーを含む式があるとき、あるいは FORM.COLUMNS の列定義パネルにエラーを含む式があるときは、QMF が REXX に値を渡して評価させるまでエラーが発見されないことがあります。

### 警告状態

書式パネルにまったくエラーがないか、すべてのエラーの訂正が終わると、QMF は次に警告状態の有無を検査します。警告状態が見つかると、QMF は最初の警告状態を含む書式パネルを表示し、そのパネルの最上部に WARNING と表示します。さらに、矛盾する値のある入力域のわきにカーソルを位置づけ、どのような状態かを説明するメッセージを表示します。

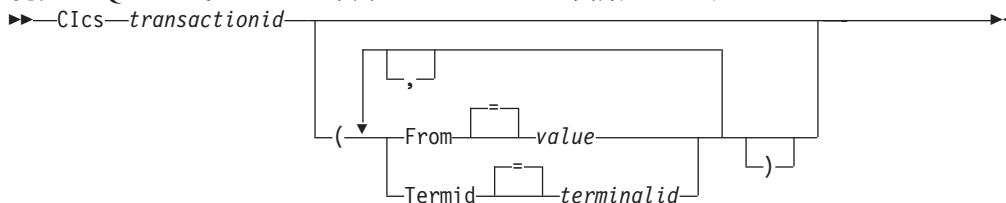
エラーと異なり、警告は強調表示されません。また、値を訂正しないまま次々に CHECK コマンドを出して、すべての警告状態をみることができます。警告状態の原因となった値を変更する必要はありません。QMF は値を適切に解釈して、報告書のフォーマットを設定します。しかし、報告書は期待通りの結果を示さないかもしれません。

---

## CICS

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
		X

CICS コマンドは CICS トランザクションを開始させます。トランザクションは、現在の QMF セッションを終了させることなく開始できます。



## 説明

### **transactionid**

開始する CICS トランザクションの名前。これは 1 から 4 文字の値です。

**FROM** トランザクションに渡されるデータを指定します。最大 78 文字のデータを渡すことができます。

### **value** データの内容を構成する文字ストリング

ブランク文字を含む値は、区切り文字で囲む必要があります。データ値用の有効な区切り文字は、単一引用符、括弧、および二重引用符です。

### **TERMID**

トランザクションに関連した CICS 端末の指定

このオプションは、端末と通信する必要のあるトランザクションの場合には必須となります。それ以外の場合は、このオプションを省略して、関連した端末を使わずにトランザクションを開始してください。

### **terminalid**

CICS 端末 ID。これは 1 から 4 文字の英数字です。

QMF セッションの現行の CICS 端末 ID は、QMF CICS コマンド・プロンプト・パネルにリストされています。

## 注

- QMF CICS CICS コマンドのパラメーター (transactionid、FROM、および TERMID) は、CICS START コマンド・オプション (TRANSID、FROM、および TERMID) と同じ意味をもっています。これらのコマンド・オプションの説明は、IBM Publications Center ([www.ibm.com/shop/publications/order](http://www.ibm.com/shop/publications/order)) にある CICS 情報を参照してください。
- CICS トランザクションは即時的に開始するようにスケジュールされます。
- CICS トランザクションは、CICS 基本マッピング・サービス、GDDM® アプリケーション、および CICS START コマンドを支配する規則に準拠する必要があります。

## 例

- QMF CICS コマンドのプロンプト・パネルを表示するには、次を入力します。  

```
CICS ?
```
- FROM パラメーターに大域変数を使用する場合には、その大域変数を括弧で囲みます。たとえば、  

```
CICS transid (FROM=(&DSQAP_CICS_PQNAME)
```

大域変数は、単一引用符で囲まないでください。正しく解決されなくなります。

## CLEAR

### CLEAR

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X	X	X

表エディターのすべてのフィールドの入力を消去するには、CLEAR コマンドを使用します。

►►—CLear—►►

注: QMF プロファイルの中で CONFIRM が YES に設定されている場合、変更モードにあれば確認パネルが表示されます。

## TSO における CONNECT

TSO (ISPF 付き)	TSO (ISPF なし)
X	X

QMF セッション内から、CONNECT コマンドを使って、分散ネットワークの一部となっている任意のデータベース・サーバーに接続できます。

### データベース・サーバーへの接続

►►—CONNect—To—servername—►►

### データベース・ユーザーの変更

(1)

►►—CONNect—authorizationid—(-Password——password—►►

#### 注:

- 1 現行のサーバーは、DB2 (OS/390® 版および z/OS 版) バージョン 7 以降でなければなりません。

### データベース・サーバーへの接続およびユーザーの設定

(1)

►►—CONNect—authorizationid—To—servername—(-Password——password—►►

#### 注:

- 1 servername は、DB2 (OS/390 版および z/OS 版) バージョン 7 以降のサーバーを指定している必要があります。

## 説明

### servername

ロケーション・パラメーターで、分散ネットワーク内のデータベース・アプリケーション・サーバーの名前を指定します。

サーバー名は二重引用符で区切ることができます。

CONNECT コマンド・プロンプト・パネルを使用しているときは、このパラメーター用のサーバー名のリストが使用可能です。『例』 の例 1 を参照してください。

#### **password**

データベース・ユーザー用のパスワード。パスワードをブランクにすることはできません。単一引用符または二重引用符で囲むことができます。

### **注**

- データベース・サーバーに接続するには、データベース許可 ID をリセットしてください。
- QMF 線形手順内で許可 ID を複数行にわたって継続する場合は、二重引用符を使用する必要があります。
- Q という DB2 (z/OS 版) 許可 ID は、QMF におけるすべての制御表、サンプル表、およびカタログ視点を所有しています。
- セキュリティーを保証し、カタログおよび制御表スペースへの無許可アクセスから保護するためには、パスワードが必要です。
- DB2 (z/OS 版) は、RACF® を使用してユーザー ID とパスワードを定義します。使用しているサイトで RACF の大/小文字混合パスワードのサポートを利用している場合、QMF プロファイルの CASE オプションが MIXED に設定されていることを確認してください。そうでなければ、QMF はすべての入力を大文字に変換してしまい、CONNECT コマンドは失敗します。
- 各サーバーの省略データベース許可 ID はシステムで定義されます。
- DB2 (z/OS 版) サーバーにおけるデータベース許可 ID は、SET CURRENT SQLID ステートメントを使った QMF SQL 照会を実行することで、変更できます。たとえば、

```
SET CURRENT SQLID = 'QMFADM'
```

大域変数 DSQAO\_DB\_MANAGER が 2 という値をもつ場合、QMF セッションは DB2 (z/OS 版) サーバーに接続されます。大域変数 DSQAO\_DB\_MANAGER が 2 以外の値をもつ場合、データベース許可 ID を変更することはできません。

- CONNECT コマンドと DSQSDBNM プログラム・パラメーターとの違いは次のとおりです。
  - 初期データベース・サーバーを確立する DSQSDBNM パラメーターは QMF セッション用に使用されます。
  - QMF セッションが確立されたあと、CONNECT コマンドがデータベース・サーバーを変更します。
- CONNECT コマンドは QMF 照会では使用できません。
- CONNECT コマンドは長い所有者名をサポートしています。プロンプト・パネルでの長い名前の入力方法の詳細は、3 ページの『プロンプト・パネルからの入力』を参照してください。

### **例**

1. CONNECT コマンド・プロンプト・パネルを表示するには、次のようにします。

```
CONNECT ?
```

## TSO における CONNECT

2. MIAMI のロケーション名を遠隔データベース・サーバーに接続するには、次のようにします。

```
CONNECT TO MIAMI
```

## QMF プロシージャーでの CONNECT コマンドの使用

QMF 線形手順内で許可 ID を複数行にわたって継続する場合は、二重引用符を使用する必要があります。図 4 に示すように、すべての継続行の 1 桁目は、正符号 (+) でなければなりません。

```
PROC          Test_Connect      MODIFIED LINE 1
CONNECT "A234567890123456789012345678901234567890123456789012345678901234567890
+12345678901234567890123456789012345678901234567890123456789012345678901234567890" (PASSWORD=XYZ)
```

図 4. QMF 線形手順内での複数行にわたる許可 ID の継続

## 分散ネットワーク内の DB2 データベースへの接続

リモートに接続すると、そこがそのユーザーの現行ロケーションになります。これらの接続は、同種のロケーション間 (DB2 - DB2) 接続でも、異種のロケーション間 (DB2 Server (VSE または VM 版) バージョン 7.3 以降、DB2 (Linux®、UNIX®、および Windows® 版) バージョン 8.1 以降、および DB2 (iSeries™) バージョン 5.2 以降) 接続でもかまいません。この接続は、QMF 初期化時に、START コマンドの DSQSDBNM プログラム・パラメーターを使用して確立できるほか、QMF CONNECT コマンドを使用する QMF セッション内からも確立できます。

リモートへの接続後、そのユーザーが出すすべての (CONNECT 以外の) SQL ステートメントはそのリモートのデータベースに送られ、そこで処理されます。したがって、リモートにあるデータと QMF オブジェクトを、ユーザーのロケーションにあるデータやオブジェクトの場合と同じようにアクセスできます。たとえば、まずリモート作業単位でリモートに接続しておけば、そこで表を作成したり、表のコメントを置き換えたりできます。

## 新規のロケーションとの接続が、長い名前のサポートに与える影響

長い名前のサポートは、接続するデータベースのリリースによって異なります。QMF CONNECT コマンドでサポートされるデータベースのリリースの詳細は、「TSO/CICS 用 DB2 QMF インストールおよび管理の手引き」を参照してください。また、接続するデータベースでどのように長い名前をサポートするかについては、該当する参照情報を参照してください。IBM Publications Center ([www.ibm.com/shop/publications/order](http://www.ibm.com/shop/publications/order)) には、各データベースの解説書が含まれています。

新しいロケーションに接続する場合、次のようになります。

- 接続に使用される許可 ID の長さは、データベースでサポートされている許可 ID、または QMF 制御表でサポートされている許可 ID のいずれかよりも長くなつてはいけません。

- 表名の最大長は、接続先のデータベースでサポートされている最大長によって異なります。
- 列名の最大長は、接続先のデータベースでサポートされている最大長によって異なります。
- QMF オブジェクト名の最大長は、QMF の制御表でサポートされている最大長 (QMF バージョン 7.2 の場合は 18 バイト、長い名前のサポートを提供する QMF 新規機能モードにマイグレーション後の QMF バージョン 8.1 以降の場合には 128 バイト) によって異なります。

## CONNECT (CICS の場合)

QMF セッション内から、CONNECT コマンドを使って、分散ネットワークの一部となっている任意のデータベース・サーバーに接続できます。DB2 Server (VSE 版) データベースに接続している場合、CONNECT コマンドを使用してデータベース・ユーザーを変更することもできます。

### データベース・サーバーへの接続

►►CONNect—To—*servername*—►►

**データベース・ユーザーの変更 (DB2 Server (VSE 版))** データベースに接続している場合のみ)

►►CONNect—*authorizationid*—(-Password——password—►►

**サーバーへの接続およびユーザーの設定 (DB2 Server (VSE 版))** データベースに接続している場合のみ)

►►CONNect—*authorizationid*—To—*servername*—(-Password——password—►►

## 説明

### **authorizationid**

遠隔データベース管理システムでのユーザー ID です。ユーザー ID はデータベースへの CONNECT 権限を保有している必要があります。

ユーザー ID は二重引用符で区切ることができます。ユーザー ID が "TO"、または "TO" の省略形であれば、二重引用符で囲む必要があります。たとえば、

CONNECT "T" TO MIAMI (PASSWORD=password)

### **Password**

データベース・ユーザー用のパスワード。パスワードをブランクにすることはできません。

パスワードを区切り文字で囲むことができます。有効な区切り文字は单一引用符または二重引用符です。

### **servername**

ロケーション・パラメーターで、分散ネットワーク内のデータベース・アプリケーション・サーバーの名前です。

## CONNECT (CICS の場合)

サーバー名は二重引用符で区切ることができます。

CONNECT コマンド・プロンプト・パネルを使用しているときは、このパラメーター用のサーバー名のリストが使用可能です。23 ページの『例』の例 1 を参照してください。

## 注

- CICS とリモート・データベース・サーバーを使用している場合、サーバーにあるデータはすべて読み取り専用に制限されます。
- データベース・サーバーに接続するには、データベース許可 ID をリセットしてください。
- 各サーバーの省略データベース許可 ID はシステムで定義されます。
- DB2 (z/OS 版) サーバーにおけるデータベース許可 ID は、SET CURRENT SQLID ステートメントを使った QMF SQL 照会を実行することで、変更できます。たとえば、

```
SET CURRENT SQLID = 'QMFADM'
```

大域変数 DSQAO\_DB\_MANAGER が 2 という値をもつ場合、QMF セッションは DB2 (z/OS 版) サーバーに接続されます。

- CONNECT コマンドと DSQSDBNM プログラム・パラメーターとの違いは次のとおりです。
  - 初期データベース・サーバーを確立する DSQSDBNM パラメーターは QMF セッション用に使用されます。
  - CONNECT コマンドは、QMF セッションが確立された後、Database Server を変更します。
- CONNECT コマンドは QMF 照会では使用できません。
- CONNECT コマンドは長い所有者名をサポートしています。プロンプト・パネルでの長い名前の入力方法の詳細は、3 ページの『プロンプト・パネルからの入力』を参照してください。
- DB2 Server (VSE 版) のみ:
  - データベース・ユーザーを変更すると、USER 特殊レジスターが変更されます。QMF セッションは新しく確立された実行時の許可 ID により特権保留を操作します。

これは、接続先を DBADM または SYSADM 権限を持つユーザー ID に変更することにより、特権データベース管理タスクを実行する場合に利用できます。

- DB2 Server (VSE 版) のユーザー・パスワードは GRANT SQL ステートメントを使用する QMF SQL 照会で実行することにより、設定または変更できます。たとえば、

```
GRANT CONNECT TO &DSQAO_CONNECT_ID  
IDENTIFIED BY password
```

- データベース・サーバーに接続するには、現行のデータベース・ユーザー ID がリモート・データベース・システムに定義されている必要があります。データベース・ユーザーが CONNECT コマンドで指定されている場合でも、この定義は必要です。

**例**

- CONNECT コマンド・プロンプト・パネルを表示するには、次のようにします。

```
CONNECT ?
```

- MIAMI のロケーション名を遠隔データベース・サーバーに接続するには、次のようにします。

```
CONNECT TO MIAMI
```

- DB2 Server (VSE 版) のみ:

- パスワード A12ZDT をもつデータベース・ユーザーを QMFADM に変更するには次のようにします。

```
CONNECT "QMFADM" (PASSWORD="A12ZDT"
```

- 他のロケーションに接続して、データベース・ユーザーを変更するには次のようにします。

```
CONNECT QMFADM TO MIAMI (PASSWORD=A12ZDT
```

**QMF プロシージャーでの CONNECT コマンドの使用**

QMF 線形手順内で許可 ID を複数行にわたって継続する場合は、二重引用符を使用する必要があります。図 5 に示すように、すべての継続行の 1 行目は、正符号 (+) でなければなりません。

PROC	Test_Connect	MODIFIED LINE 1
CONNECT "A23456789012345678901234567890123456789012345678901234567890 +1234567890123456789012345678901234567890123456789012345678901234567890		

図 5. QMF 線形手順内の複数行にわたる許可 ID の継続

**長いオブジェクト名のサポート**

新しいロケーションに接続するときのオブジェクト名の最大長と修飾子については、20 ページの『新規のロケーションとの接続が、長い名前のサポートに与える影響』を参照してください。

**CONVERT**

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X	X	X

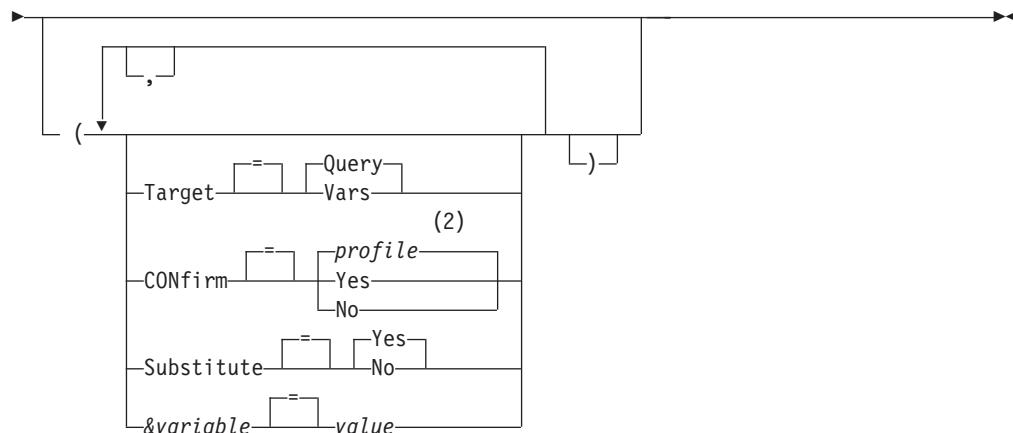
CONVERT コマンドは、指示照会、SQL 照会、または QBE 照会を標準の SQL 構文の照会に変換します。置換変数は、指定した値または大域変数で定義された値で置き換えられます。 CONVERT コマンドは、変数に値を割り当て、もとの照会のすべての注釈を除去します。

**一時記憶域にある照会の変換**

(1)



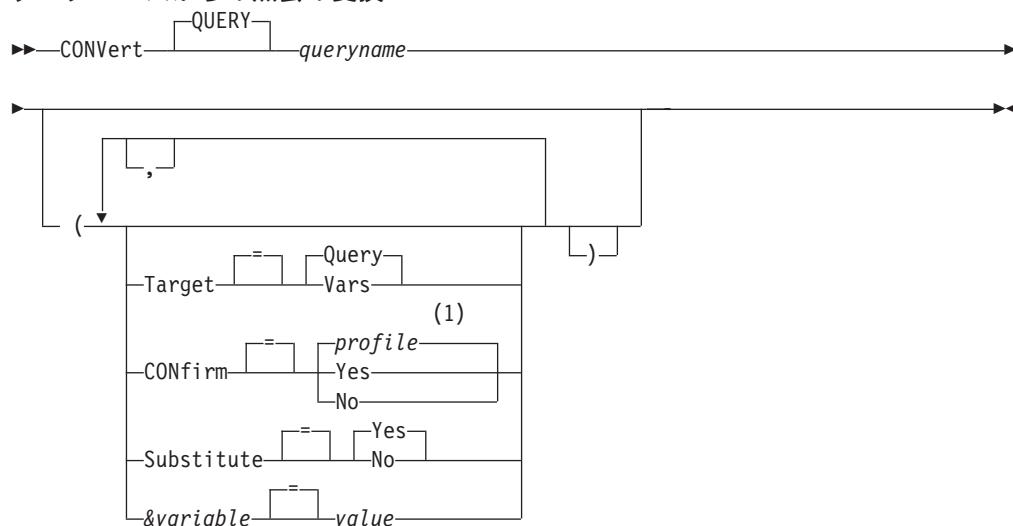
## CONVERT



注:

- 該当する場合は、現在表示されている QMF 対象パネルの名前が使用されます。
- プロファイルで設定された値が使用されます。

### データベースからの照会の変換



注:

- プロファイルで設定された値が使用されます。

## 説明

### queryname

データベースに保管されている照会の名前です。データベースに保管されている照会は変更されず、QMF 一時記憶域にある照会が、保管されている照会のコピーで置き換えられます。

### TARGET

変換された照会の配置の制御。

**QUERY**

変換された照会を SQL 照会パネル上に置きます。一時記憶域にある照会は、変換された照会で置き換えられます。

**VARS** 変換された照会および DSQQC\_ で始まる QMF 大域変数の照会に関する情報を置きます。(詳細については、293 ページの『付録 B. QMF 大域変数表』を参照してください。) ISPF が使用できる場合は、変換された照会もまた、ISPF 対話管理機能の変数プールに置かれます。(ISPF は CICS では利用不能です。) 一時記憶域にある照会は変更されません。大域変数と ISPF 変数プールだけが変更されます。

**CONFIRM**

このコマンドがデータベース内のオブジェクトを置き換えようとするときに、確認パネルを表示するかどうかを指示します。

**SUBSTITUTE**

照会中の置換変数を値で置き換えるかどうかを指定します。

**YES** 照会に変数を持っている場合、QMF は値を変数に置き換えようとします。すべての変数が定義された場合、指示パネルは表示されません。QMF がすべての変数を解決できなかった場合は、値を入力するように指示されます。QMF はまず変数定義をコマンドで探し、次に既存の大域変数を探します。

**NO** 照会中のどの変数名も解決されていません。

**&variable**

CONVERT コマンド用の置換変数を識別します。このオプションでは、変数に対して 1 バイト文字で 55 文字までの値を割り当てることができます。1 つのコマンドで、10 個までの置換変数を指定することができます。

変数名は アンパーサンド記号で始める必要があります。線形手順内で CONVERT コマンドを実行する場合は、& 記号を 2 個続けて使用してください。

**value** 置換変数の内容を構成する文字ストリング。

ブランク文字を含む値は、区切り文字で囲む必要があります。置換変数の値で有効な区切り文字は、单一引用符、二重引用符および括弧です。区切り文字が引用符の場合は、引用符は値の一部として組み込まれます。区切り文字が括弧の場合は、括弧は値の一部として組み込まれません。

**注**

- CONVERT コマンドを使用して、一時記憶域にある既存の SQL 照会の編成を改良することもできます。
- CONVERT コマンドに 10 個を超える変数を指定すると、コマンドは拒否され、エラー・メッセージが表示されます。
- 照会の中のパラメーターと対応しない変数名は無視されます。

## CONVERT

SET GLOBAL コマンドで変数を定義した場合には、CONVERT コマンドで変数を指定する必要はありません。 CONVERT コマンドで指定された値は、SET GLOBAL で設定された値を上書きします。

照会の中に変数があるのに、それを置換する値が CONVERT コマンドで指定されていないと、プロンプト・パネルが表示されます。そのプロンプト・パネルには、与えられたすべてのパラメーター値が表示されています。 照会の中で値を割り当てられていない変数名があると、その変数名がリストされ、メッセージが表示されます。

- 照会は 3 部分名を使用できません。
- 置換変数用の変数を指定している一方で、SUBSTITUTE=NO も指定していると、エラー・メッセージが発行されます。
- 変数の値として照会コメントを入力しないでください。 照会コメントは、先頭に 2 個のダッシュ (-) が付いていますが、データベースはそれを負符号 (-) と解釈します。
- TARGET を下記のように指定して照会を変換する場合。
  - QUERY を指定した場合、変換済みの照会は QMF 一時記憶域に置かれます。 変換したい照会が QMF 一時記憶域にある場合は、変換された照会がそれを置き換えます。 変換したい照会がデータベースに保管されている場合、変換された照会は QMF 一時記憶域に置かれ、表示されます。
  - VARS を指定した場合、変換済みの照会は ISPF ダイアログ・マネージャー・プール、および大域変数プールに置かれます。 変換済み照会は、QMF 一時記憶域にある照会を置き換えません。
- 1 つの QBE 挿入照会または削除照会が、複数の SQL 照会になることもあります。 これらの照会は単一の SQL 照会オブジェクトに入れられます。 ただし、2 番目以降の照会はすべてコメントに変わります。(各行の先頭に 2 個のハイフンが付けられます。)
- CONVERT コマンドは長いオブジェクト名をサポートしています。 プロンプト・パネルでの長い名前の入力方法の詳細は、3 ページの『プロンプト・パネルからの入力』を参照してください。

## 例

1. QMF 一時記憶域にある照会を SQL 照会に変換し、変換済み照会中の変数 DEPT に 38 という値を置き換えるには、次のようにします。

```
CONVERT QUERY (&DEPT=38
```
2. 次の例は、CONVERT コマンドを使用して既存の SQL 照会の編成を改善する方法を示しています。 たとえば、一時記憶域に下記のような SQL 照会があるとします。

```
SELECT 'JOB', JOB, 'SERIAL', ID FROM Q.STAFF  
WHERE ID<99 ORDER BY 2
```

CONVERT コマンドを実行したとの照会は下記のようになります。

```
SELECT 'JOB', JOB, 'SERIAL', ID  
FROM Q.STAFF  
WHERE ID < 99  
ORDER BY 2
```

3. 照会を QBEQUERY というデータベースから QMF 一時記憶域の SQL 照会に変換するには、次のようにします。

```
CONVERT QUERY QBEQUERY
```

4. 照会を MYQUERY というデータベースから SQL 照会に変換し、 ISPF 対話管理機能プールおよび大域変数プールに入れるには、次のようにします。

```
CONVERT QUERY MYQUERY (TARGET=VARS)
```

## DELETE

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X	X	X

DELETE コマンドは次に挙げる項目を除去します。

- SQL 照会またはプロシージャーから 1 行
- 指示照会のパネルから 1 行
- FORM.MAIN または FORM.COLUMNS から列情報を 1 行
- FORM.CALC パネルから計算行を 1 行
- FORM.CONDITIONS から 1 条件
- FORM.BREAK、FORM.DETAIL、FORM.FINAL、または FORM.PAGE から 1 テキスト行
- 照会の下に表示されるエラー・メッセージ
- 表エディターを使用しているとき、データベース中の表から 1 行  
►—DELetे—►►

### 注:

- 行を削除するには、削除したい行にカーソルを合わせて、削除キーを押してください。
- 表エディターで DELETE を使用するときは、そのトランザクションをただちに保管させることも、表編集セッションの終了時に保管させることもできます。どちらにするかは、EDIT TABLE コマンドの SAVE オプションで指定できます。
- 表または表結合を指示照会から削除すると、QMF は他の結合を再評価して、残っている表がまだ接続（または結合）状態にあるかどうかを調べます。
  - そうである場合、残りのすべての結合は照会に残ります。
  - そうでない場合、照会に選択された最初の表に接続されている表についてのみ、結合が残ります。「表の結合」パネルが表示され、他の表の残りの結合を作成することを指示されます。

## DESCRIBE

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X	X	X

## DESCRIBE

DESCRIBE コマンドを使用して、QMF オブジェクトまたは表の列に関する情報を表示します。「記述」機能キーは、データベース対象リスト・パネルまたは「指示照会」パネルから使用することができます。

►►DEScrite

**注:** データベース・オブジェクト・リスト・パネルで DESCRIBE を使用すると、ある 1 つのオブジェクトについての明細情報が表示されます。表示される情報の量は、オブジェクト・タイプによって異なります。たとえば、「指示照会」パネルでは、DESCRIBE は、リストされた列の情報を示す「列記述」パネルを表示します。

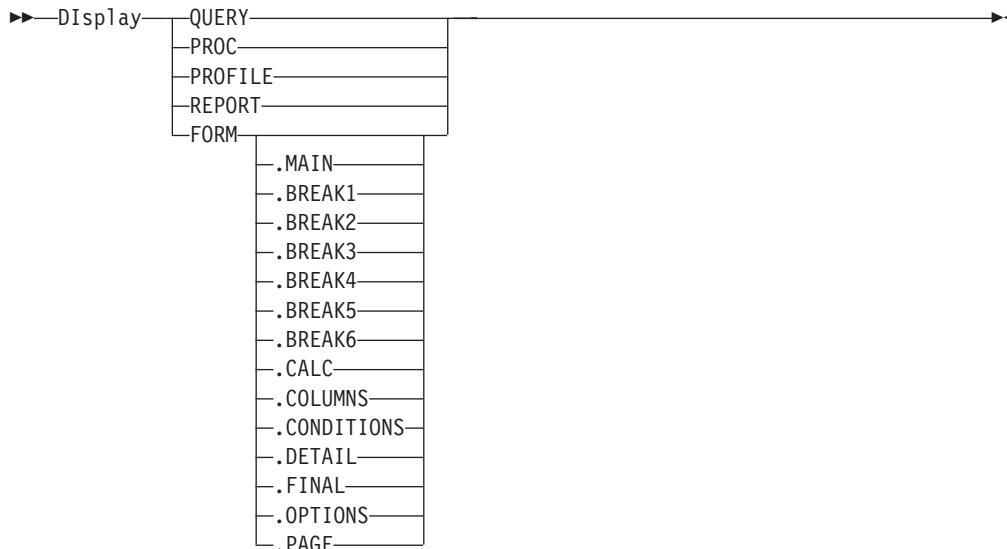
---

## DISPLAY

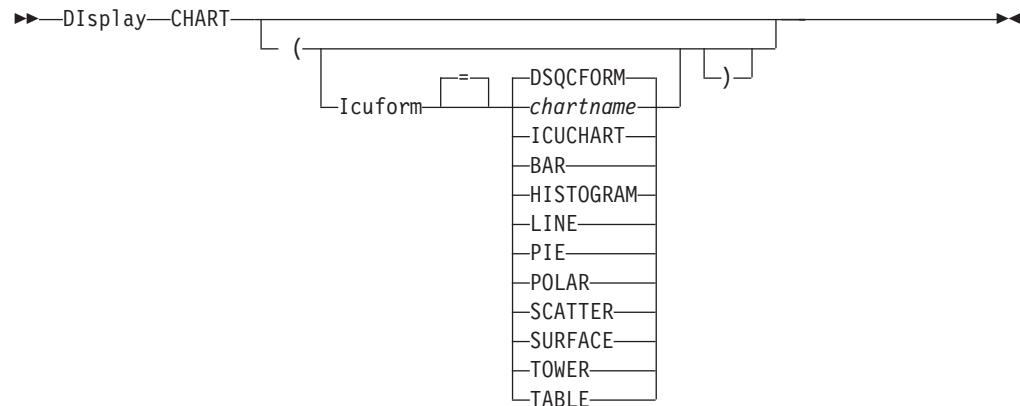
TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X	X	X

DISPLAY コマンドは、QMF 一時記憶域にあるオブジェクト、またはデータベースからのオブジェクトを表示します。

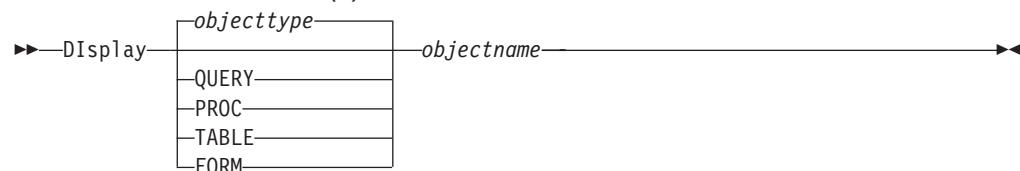
### 一時記憶域にある QMF オブジェクトの表示



### DisplayaCHART



### データベースからのオブジェクトの表示 (1)



#### 注:

- 該当する場合は、名付けられた対象のタイプが使用されます。 QMF オブジェクトは他のタイプのオブジェクト (たとえば、データベース・オブジェクト) より優先されます。

## 説明

### **objectname**

データベース内の対象の名前です。有効なオブジェクトは下記のとおりです。

- QMF オブジェクト (PROC、QUERY、FORM)
- 表オブジェクト (TABLE、VIEW、SYNONYM、ALIAS)

### **ICUFORM**

GDDM 対話式图表ユーティリティー (ICU) で使用する图表フォーマットを指示します。 QMF にはそのまま使用できる图表スタイルがいくつか備わっています。

### **DSQCFORM**

QMF が提供するデフォルトの图表フォーマットの名前。 QMF 管理者がカスタマイズしていなければ、このオプションは棒グラフの图表を提供します。

### **ICUCHART**

ICU 提供の省略時图表形式の名前です。

### **chartname**

前に ICU に保管された图表フォーマットの名前を示します。

## 注

- QMF 管理者は、データベースに保管されている任意の QMF オブジェクトを表示できます。
- 指定されたオブジェクトが表でないときは、QMF 一時記憶域にある同じオブジェクトの内容が、そのオブジェクトで置き換えられます。

指定された対象が表の場合は、一時記憶域の QMF データ対象および QMF 書式対象の内容を置き換えます。表のデータに合わせて新たな書式が作成されます。この書式は、表示される報告書のデフォルトのフォーマット設定を行います。

- 権限を与えられていれば、他のユーザーが所有している表を表示することができます。他のユーザーが所有している表を表示するには、所有者修飾子を使用してください。
- 現行のデータベースのロケーションが DB2 (z/OS 版) サーバーである場合には、リモート・ロケーションにある表を表示することができます。表オブジェクトを 3 部分名で指定してください。(31 ページの『例』に例を示しています。) データベース管理者が複数行フェッチ機能を使用するように QMF をセットアップしている場合、操作しているデータベースは(ローカルとリモートの)両方とも DB2 (z/OS 版) 8.1.5 以降でなければなりません。そうでなければ、コマンドは失敗します。データベース管理者は、複数行フェッチをオフにできます。この機能の設定については、「*TSO/CICS 用 DB2 QMF インストールおよび管理の手引き*」に説明されています。
- SHOW コマンドは DISPLAY コマンドと似ています。違いは以下のとおりです。

**SHOW** QMF 一時記憶域にあるオブジェクト・パネル、大域変数、および特定のパネルの部分を表示します。

**DISPLAY** QMF オブジェクトとデータベース・オブジェクトの両方を表示します。

- 表示された SQL 照会、書式、または手順を、Insert および Delete ファンクション・キーを使用して変更できます。書式のテキストまたはデータに上書きすることもできます。変更されたオブジェクトを SAVE コマンドで保管します。
- 前に書式パネルを見ていれば、DISPLAY FORM は、最後に見た書式パネルを表示します。現行の書式のどの部分も見ていないときは、DISPLAY FORM は FORM.MAIN を表示します。
- DISPLAY CHART を使用すると、DATA の内容が FORM でのフォーマット設定どおりに表示されます。対話式図表ユーティリティー (ICU) でさらに処理すれば、報告書データをグラフィックに表示することもできます。図表を表示するには、グラフィック・ディスプレイ装置が必要です。
- ICU で図表を処理したのち、ICU を終了すると、DISPLAY CHART コマンドを入力した QMF パネルが再度表示されます。いずれかの書式パネルに戻りたいときは、その書式パネルから DISPLAY CHART コマンドを入力してください。
- DISPLAY コマンド・プロンプトで CHART と入力した場合は、DISPLAY CHART コマンド・プロンプトが表示され、図表の表示に必要なパラメーターを指定することができます。
- 報告書または図表を表示しているとき、書式がデータと整合しなかったり、書式にエラーがあつたりすると、エラーが含まれている最初の書式パネルが表示さ

れ、そのエラーが強調表示されます。表示された最初のエラーを訂正してから、CHECK コマンドを出すか、書式あるいは図表を再度表示して、次のエラーを見る必要があります。

- QMF は、特定のデータ・タイプの列に、編集コード M を自動的に割り当てます。この編集コードは、実際のデータではなく、列メタデータ（データ・タイプと長さ）を表示します。この編集コードは、データ・タイプによって、変更できたりできなかったりします。詳細については、268 ページの『QMF が列メタデータを表示するためのデータ・タイプ』を参照してください。
- XML データを含むデータベース・オブジェクトの表示は、32K に制限されます。
- DISPLAY CHART コマンドを使用して、BINARY、VARBINARY、XML、または DECFLOAT と定義された列を含むデータまたは表の図表は作成できません。
- DISPLAY コマンドは長いオブジェクト名をサポートしています。プロンプト・パネルでの長い名前の入力方法の詳細は、3 ページの『プロンプト・パネルからの入力』を参照してください。

## 例

1. QMF DISPLAY コマンド用のプロンプト・パネルを表示するには次のようにします。

```
DISPLAY ?
```

2. 現行の QMF プロシージャー・オブジェクトを表示します。

```
DISPLAY PROC
```

3. ユーザー名が JANET のユーザーが所有する MONTHLY という名前の共有 QMF 照会を表示するには、次のようにします。

```
DISPLAY QUERY JANET.MONTHLY
```

4. 現行ロケーションが DB2 (z/OS 版) サーバーで、BOISE という名前のリモート・データベースにある、ユーザー名 JOHNSON が所有している VISION という表を表示するには、次のコマンドを入力します。

```
DISPLAY TABLE BOISE.JOHNSON.VISION
```

5. 図 6 の例では、QMF プロシージャーへの入力で、DISPLAY コマンドに複数行にわたる表名を指定する方法を示しています。

```
PROC MODIFIED LINE 1DISPLAY TABLE
+"LOCATION12345678"."LONGOWNERID123456789112345678921345678931234567894123
+4567123456789112345678921234567893123456789412346789512345678961234567897
+12345"."LONGNAME123456789112345678921234567893123456789412345678951234567
+8961234567897123456789112345678921234567893123456789412345678951234567
```

図 6. 線形手順への、複数行にわたるオブジェクト名の入力

---

## DPRE

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X		

## DPRE

DPRE は、QMF が提供するコマンドの同義語（印刷報告書表示のアプリケーションを実行する）です。この適用業務を使用して、書式化された報告書を端末に表示することができます。ここには、現在 QMF 一時記憶域にある報告書が表示されます。

►►DPre—►►

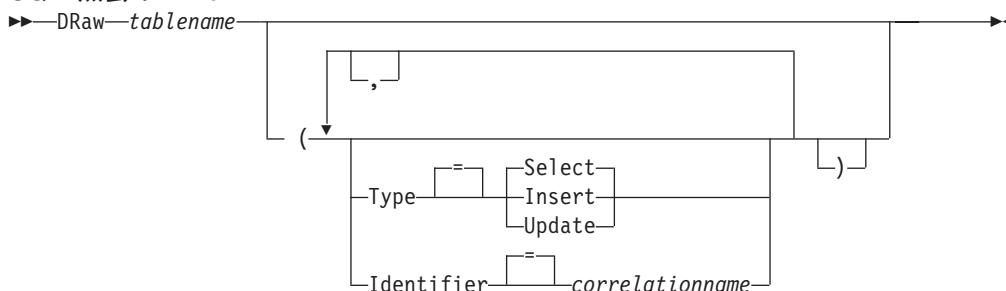
さらに詳しい DPRE の使い方については、「*TSO/CICS 用 DB2 QMF インストールおよび管理の手引き*」を参照してください。

## DRAW

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X	X	X

DRAW コマンドは、基本 SQL 照会や QBE 照会を組み立てるときに役に立ちます。

### SQL 照会の DRaw



### QBE 照会の DRaw

►►DRaw—tablename—►►

## 説明

### tablename

データベースにある表の名前。

これを TABLE、VIEW、SYNONYM、または ALIAS の名前にできます。

**TYPE** 組み立てようとしている照会のタイプ。

### SELECT

表または視点の各列からデータを選択するための基本照会を組み立てます。照会が表示されたならば、必要に応じて他の文節を入力してください。複数の表から選択するには、表ごとに DRAW コマンドを使用してください。これがデフォルトの照会タイプです。

### INSERT

表または視点にデータを挿入するための基本照会を組み立てます。照会が表示されたならば、列名の左に新規のデータを入力してください。

**UPDATE**

表または視点の指定の行の値を変更するために、基本照会を組み立てます。照会が表示されたら、変更を列名の右に入れ、不要な行を削除してください。

**IDENTIFIER**

組み立てられた照会で、識別子を一意的に指定する表に指定してください。  
TYPE=INSERT の場合、このオプションは無視されます。

**correlationname**

ユーザー定義の名前は、組み立てられた照会で表の相関名となります。この名前は、あいまいさを避けるため、または副照会の相関参照を確立するために照会の修飾列に使用されます。単に、照会で表の名前を読みやすい名前に改善するために使用することもできます。

このオプションを指定しないと、組み立て後の照会に相関名は追加されません。

**注**

- DRAW コマンドは SQL QUERY パネルまたは QBE QUERY パネルでのみ有効です。
- 既存の SQL SELECT 照会に他の表を追加するときは必ず、IDENTIFIER オプションを使用してください。
- 実行の前に追加情報を必要とする照会もあります。
- 表名にロケーション修飾子を含めることによって、他のロケーションにある表や視点をドローすることができます。
- QBE での DRAW コマンドの働き方に関する情報を得るために、詳細ヘルプ・キーを押してください。
- DRAW コマンドは長いオブジェクト名をサポートしています。プロンプト・パネルでの長い名前の入力方法の詳細は、3 ページの『プロンプト・パネルからの入力』を参照してください。

**例**

- S で一意的に識別される表 Q.STAFF のために SELECT 照会をドローするには次のようにします。

```
DRAW Q.STAFF (TYPE=SELECT IDENTIFIER=S)
```

結果は次のようにになります。

```
SELECT S.ID, S."NAME", S.DEPT, S.JOB, S."YEARS"
      , S.SALARY, S.COMM
   FROM Q.STAFF S
```

- 表名または列名に次のものが含まれると、DRAW コマンドは、名前を二重引用符で囲みます。
  - 特殊文字
  - QMF 予約語
  - IBM® SQL 予約語
  - DB2 予約語

## DRAW

たとえば、MYTABLE という表に特殊文字または予約語が含まれていて、コマンド DRAW MYTABLE を発行したとします。

結果は次のようになります。

```
SELECT NORMALNAME, KEYWORDFOLLOWS, "UNION"
      , "HAS BLANKS IN IT", "SPECIAL+CHARS_IN!"
      , "Mixed_Case_%S" FROM USER.MYTABLE
```

3. QMF プロシージャーで DRAW コマンドを使用する場合、QMF 線形手順内で複数行にわたる照会オブジェクト名を継続するには、区切り ID (二重引用符) を使用する必要があります。図 7 に示すように、すべての継続行の 1 桁目は、正符号 (+) でなければなりません。

```
PROC                               MODIFIED LINE    1
                                   
RESET QUERY
DRAW
+"LOCXXXXXXXXXXXXXX"."AUTHXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
+XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX"."OBJXX
+XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
+XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX"
```

図 7. 線形手順での、複数行にわたる DRAW コマンドの継続

## EDIT オブジェクト

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X	*	

現在一時記憶域に存在する QMF プロシージャーまたは SQL 照会を変更するには、EDIT オブジェクト・コマンドを使用してください。



注:

- 該当する場合は、現在表示されている QMF 対象パネルの名前が使用されます。

## 説明

### EDITOR

QMF プロシージャーまたは SQL 照会の編集に使用するエディターの名前を指定します。

### PDF

ISPF/PDF 編集機能を手順または照会の編集に使用すること

を指定します。 PDF 編集機能を使用して照会または手順を編集するには、QMF を ISPF 対話として開始します。

*editorname*

使用できるその他の編集機能の名前です。これには、編集機能を開始する CLIST の名前を指定することもできます。使用できるエディターの詳細は、QMF 管理者に問い合わせてください。

**注**

- EDIT コマンドを使用して新しい照会または手順を組み立てたい場合は、最初に照会または手順をリセットして QMF 一時記憶域をクリアしてください。それを行うには、QUERY パラメーターまたは PROC パラメーターを指定した RESET コマンドを使用します。
- 既存の照会またはプロシージャーを変更したいときは、まず、それを表示して、その照会またはプロシージャーを QMF 一時記憶域に入れます。次に、EDIT コマンドを用いて、照会またはプロシージャーを変更します。
- 照会またはプロシージャーを編集したあと、それを保管できます。これによって、QMF 一時記憶域に入っていた内容が新しい内容で置き換えられます。照会またはプロシージャーが大きすぎて QMF の一時記憶域に入りきらないときは、照会またはプロシージャーはデータ・セットに保管されます。これが起こった場合には、そのプロシージャーまたは照会の保管先となったデータ・セットの名前を伝えるメッセージが表示されます。
- エディターの SAVE コマンドは、QMF の SAVE コマンドと同じではありません。エディターは、QMF 一時記憶域にしか保管しません。照会または手順をデータベースに保管したい場合は、QMF SAVE コマンドを使用してください。
- CICS 環境では、EDIT コマンドによる QMF 照会の編集や QMF プロシージャーの編集はできません。しかし、QMF DISPLAY コマンドまたは SHOW コマンドでそのオブジェクトを表示してから、QMF で変更することができます。

**例**

1. EDIT コマンド・プロンプトパネルを表示するには、次のようにします。

```
EDIT ?
```

2. 現行の照会をエクスポートして、ISPF/PDF エディターに入れるには、次のようにします。

```
EDIT QUERY
```

編集セッションが終了すると、編集済みデータ・セットは現行の照会オブジェクトにインポートされます。

PDF エディターを使用するには、QMF を ISPF ダイアログとして開始しておかなければなりません。

**EDIT TABLE**

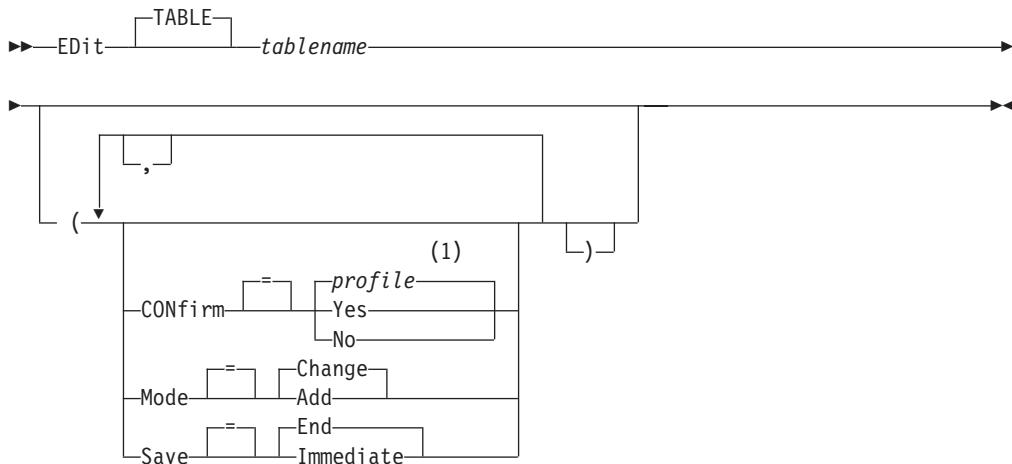
TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X	X	*

## EDIT TABLE

EDIT TABLE コマンドは QMF 表エディターを起動します。表エディターのセッションの間に、表示されているパネルのフィールドを使用して、表に対するレコードの追加、変更、または削除を行なうことができます。

表編集機能セッションを終了するには、END コマンドを発行してください。

### 表の編集



### 注:

- 1 プロファイルで設定された値が使用されます。

## 説明

### tablename

データベースにある表の名前。

### MODE 実行させる表エディター・セッションのタイプ。

#### CHANGE

表の中の行を変更できるようにするモードで、表エディターを動かします。変更モードには下記の機能が含まれています。

- 行を検索する。
- 行のデータを表示する。
- 行の中の欄を更新する。
- 行を削除する。
- 複数の行を進める。

#### ADD 新しいレコードを表へ追加できるようにするモードで、表エディターを作動させます。

#### SAVE 表エディター・セッションで発生した変更と削除をコミットするときを指定します。

#### IMMEDIATE

編集セッションの間に行なわれた変更は、行ごとに個別に処理されます。これを選択すると、編集セッションが活動状態のときの、別のユーザーによる表の可用性が大きくなりま

#### END

編集セッションで発生した変更が、セッションの終了時まで

保持されます。変更はすべて、いつでもを取り消すことができます。これを選択すると、編集セッションの進行中の、別のユーザーによる表の可用性が小さくなります。

#### **CONFIRM**

表エディター・セッションの間、確認パネルを表示するかどうかを指示します。

下記のようなセッション・イベント用に確認パネルがあります。

- 行の追加
- 行の変更
- 行の削除
- 入力した項目を消そうとしている
- セッションの終了

#### **注**

- 10 進浮動小数点 (DECFLOAT)、2 進 (BINARY または VARBINARY)、または XML データ・タイプを含む表を、表エディターを使用して編集することはできません。
- 表編集機能は、CHANGE モードでは、VARCHAR 列の末尾ブランクを取り除きます。更新後の VARCHAR 列にはブランクしか含まれていない場合は、この列の長さはゼロになります。
- QMF は、さまざまな編集セッション確認パネルの活動化を個別に制御するため大域変数のセットを提供しています。詳細については、293 ページの『付録 B. QMF 大域変数表』を参照してください。
- 表エディターは特別な予約文字を使用して NULL 値、デフォルト値をサポートしています。編集セッションに先立ち、大域変数の値を変更することによって、これらの予約文字の定義を変更することができます。詳細については、293 ページの『付録 B. QMF 大域変数表』を参照してください。
- 表エディターは長い表名の入力をサポートします。プロンプト・パネルでの長い名前の入力方法の詳細は、3 ページの『プロンプト・パネルからの入力』を参照してください。

#### **例**

1. QMF EDIT TABLE コマンドのためのプロンプト・パネルを表示するには次のようにします。

```
EDIT TABLE ?
```

2. ユーザー Bill が所有している TABTWO という表に新しい数行を追加します。

```
EDIT TABLE BILL.TABTWO (MODE=ADD)
```

---

#### **END**

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X	X	X

END コマンドは現在の操作を終了させ、操作開始前の状態に戻ります。

►► END ━━━━ ►►

END コマンドの結果は、どのパネルを使用しているか、また初期手順を実行しているかによって変わります。

- QMF ホーム・パネルから END を入力すると（または End (終了) ファンクション・キーを押すと）、QMF セッションは終了します。
- 次のいずれかの QMF パネルから END を入力すると（または End (終了) ファンクション・キーを押すと）、QMF ホーム・パネルが表示されます。

QUERY	FORM.MAIN	FORM.COLUMNS
PROC	FORM.CALC	FORM.OPTIONS
PROFILE	FORM.DETAIL	FORM.BREAK.n
REPORT	FORM.FINAL	FORM.CONDITIONS
	FORM.PAGE	Global variable list

- プロンプト・パネルであれば、そのプロンプト・パネル表示の原因となったコマンドが出されたパネルが表示されます。（そのパネルとしては、QMF ホーム・パネル、書式、プロファイル、照会、報告書の関連パネルがあります。）

プロンプト・パネルに入力を行ったあと、まだ Enter を押さないうちに End (終了) ファンクション・キーを押すと、その入力は処理されません。

- 「表編集機能」パネルから変更がコミットされ、表編集機能を呼び出したパネルが表示されます。

「表編集機能」パネルから終了機能キーを押すと、確認パネルが表示され、終了する（変更をデータベースにコミットする）か、終了しない（「表編集機能」パネルに戻る）かを決定することができます。

次の状況では、END コマンドの働きが上記の場合と異なります。

- QMF の開始が初期処理手順で行なわれていたときは、END で初期処理手順が再実行され、QMF ホーム・パネルは表示されません。
- 現在のパネルが QMF ホーム・パネルで、END が QMF コマンドまたは呼び出し可能インターフェースを介して発行された場合、QMF セッションはすぐには終了しません。その代わりに、END コマンドが含まれている CLIST またはプログラムが制御を取り戻します。この場合、その CLIST またはプログラムが終了するまで、QMF セッションは終了しません。
- END が INTERACT コマンドにより開始された新しい対話式セッションから発行された場合、制御は、INTERACT コマンドが発行された適用業務または手順に戻ります。この場合、END はセッションを終了したり、QMF ホーム・パネルを表示したりしません。
- データベース・オブジェクト・リスト・パネルでなんらかのコマンドを出した結果として新しい対話式セッションが開始されて、そこで END コマンドが出されたときは、データベース・オブジェクト・リストが表示されます。この場合、END はセッションを終了したり、QMF ホーム・パネルを表示したりしません。

呼び出し可能インターフェースおよびコマンド・インターフェース・プログラムとの END コマンドの使用についての詳細は、「*DB2 QMF アプリケーション開発の手引き*」を参照してください。

## ENLARGE

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X	X	X

QMF で ENLARGE コマンドを発行すると、QBE 例示表のサイズが大きくなります。

►►—ENLarge————►►

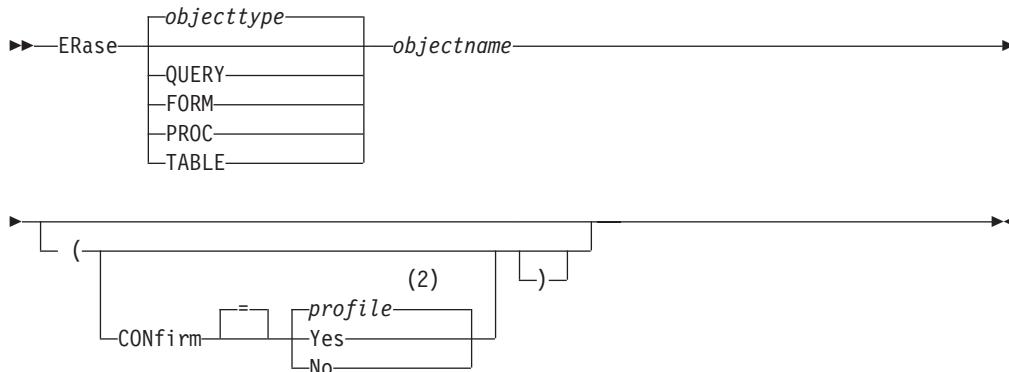
## ERASE

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X	X	X

ERASE コマンドはデータベースからオブジェクトを消去します。

データベースからのオブジェクトの消去

(1)



注:

- 1 該当する場合は、名付けられた対象のタイプが使用されます。 QMF オブジェクトは他のタイプのオブジェクト (たとえば、データベース・オブジェクト) より優先されます。
- 2 プロファイルで設定された値が使用されます。

## 説明

### objectname

データベースにある QMF オブジェクトの名前。

FORM オブジェクトの名前を指定すると、フォームのすべての部分が消去されます。

### CONFIRM

確認パネルを表示するかどうかの指定。

## **ERASE**

**YES** このコマンドでデータベース内のオブジェクトが消去されるときは、確認パネルを表示します。

**NO** 確認パネルを表示しません。

注

- ・ 現行データベース・ロケーションからのみ、対象を消去できます。3部分名を使ってリモート表を消去することはできません。代わりに、まずその表が置かれてるロケーションに接続し、それから ERASE コマンドを発行します。
  - ・ 存在しないオブジェクト名を指定しても、線形手順からは警告メッセージは発行されません。
  - ・ ERASE コマンドは長いオブジェクト名をサポートしています。プロンプト・パネルでの長い名前の入力方法の詳細は、3ページの『プロンプト・パネルからの入力』を参照してください。

例

1. コマンド・プロンプト・パネルを表示する場合:  
ERASE ?
  2. PATTI.TABLEONE: という表を消去するには次のようにします。  
ERASE TABLE PATTI.TABLEONE
  3. JBQUERY という名前の照会を消去し、確認パネルを表示するには次のようにします。  
ERASE JBQUERY (CONFIRM=YES)
  4. ローカル・ロケーションが BOISE であり、DALLAS にある  
PATTI.TABLETWO という表を消去するには、まず、次のようにして DALLAS  
に接続しなければなりません。

[CONNECT TO DALLAS](#)

次に、ERASE コマンドを

- ERASE TABLE PATTI.TABLETWO

5. QMF プロシージャーで ERASE コマンドを使用する場合、QMF 線形手順内で複数行にわたる許可 ID を継続するには、二重引用符を使用する必要があります。図 8 に示すように、すべての継続行の 1 枠目には、正符号 (+) がなければなりません。

図8. 複数行にわたる、ERASE コマンド行を使用する許可 ID の継続

---

EXIT

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X	X	X

EXIT コマンドは、QMF セッションを終了させます。

►►—EXIT—►►

---

このコマンドは、QMF ホーム・パネルから出せるほか、  
QUERY、REPORT、FORM、PROFILE の各パネル、または大域変数リスト・パネル  
から出せます。プロシージャーに入れておくこともできます。

QMF データベース・オブジェクト・リスト・パネルの任意のオブジェクトの QMF  
コマンド域からも EXIT コマンドを入力することができます (78 ページの『LIST』  
を参照)。コマンド・プロンプト・パネル、確認パネル、ヘルプ・パネルには、EXIT  
コマンドを入力することはできません。

**QMF アプリケーション開発ユーザーへ:** QMF コマンド・インターフェースを介して、またはコマンド・インターフェースを介して実行されるプロシージャーにおいて、EXIT を実行しても、セッションは即時には終了しません。その代わり、コマンド・インターフェースから実行されている CLIST、またはアプリケーション・プログラムが制御を取り戻します。CLIST またはアプリケーションが完了するまで、セッションは終了しません。

## EXPORT (CICS の場合)

### EXPORT (CICS の場合)

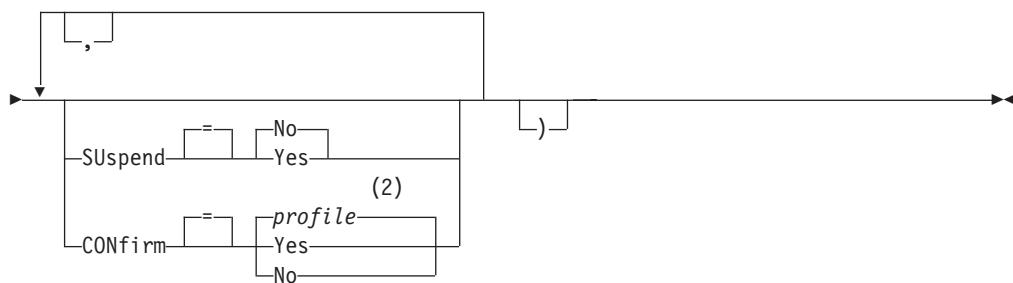
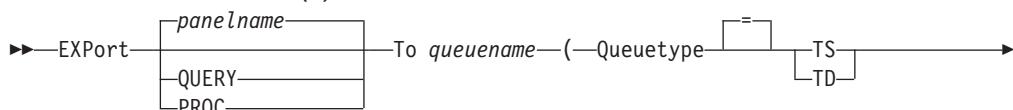
CICS の場合、EXPORT コマンドは次のものを送ります。

- QMF 一時記憶域にある照会、書式、プロシージャー、報告書、およびデータを CICS データ・キューへ。
- データベースにある照会、書式、プロシージャー、および表を CICS データ・キューへ。
- QMF の図表を、GDF ファイルが入っている GDDM ライブラリへ。

オブジェクトを QMF 一時記憶域からエクスポートするための構文と、データベースからエクスポートするための構文は異なります。

#### 一時記憶域からの QMF 照会またはプロシージャーのエクスポート

(1)

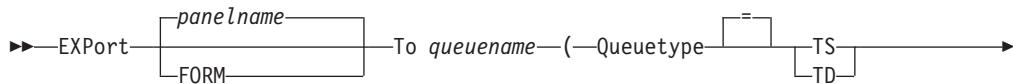


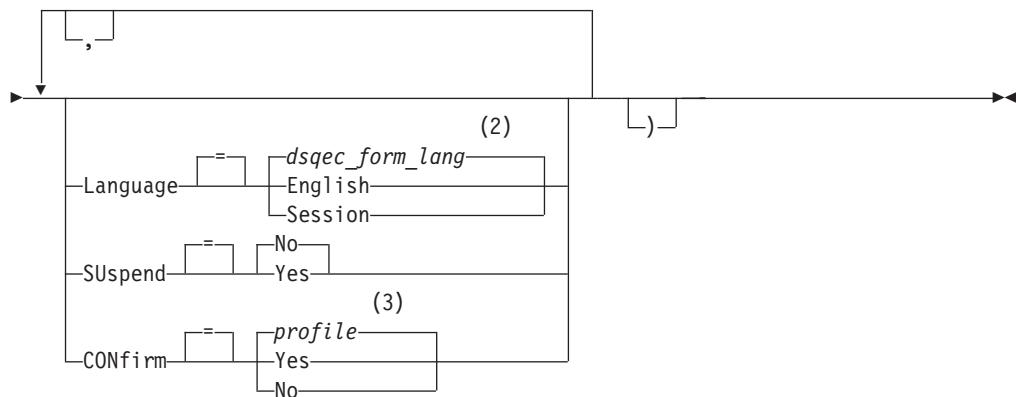
#### 注:

- 1 該当する場合は、現在表示されている QMF 対象パネルの名前が使用されます。
- 2 プロファイルで設定された値が使用されます。

#### 一時記憶域からの QMF 書式のエクスポート

(1)

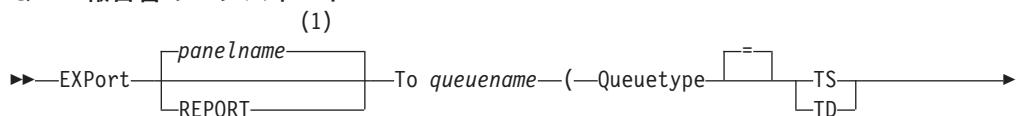




注:

- 1 該当する場合は、現在表示されている QMF 対象パネルの名前が使用されます。
- 2 この大域変数で設定された値が使用されます。
- 3 プロファイルで設定された値が使用されます。

#### QMF 報告書のエクスポート

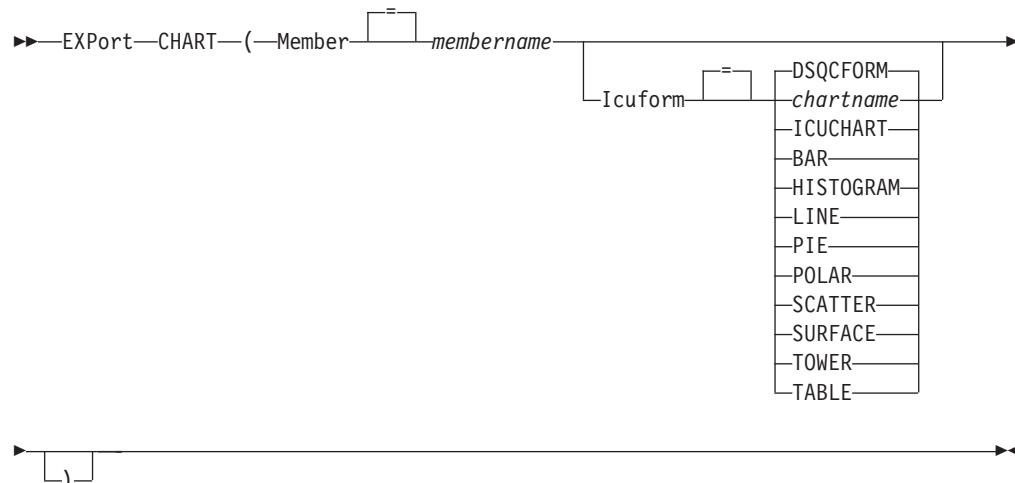


注:

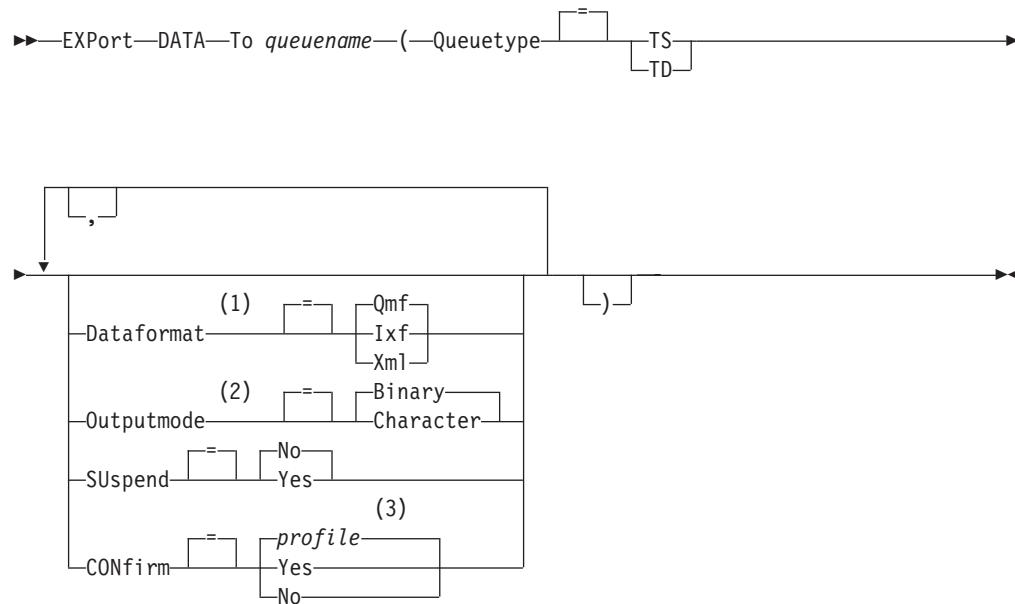
- 1 該当する場合は、現在表示されている QMF 対象パネルの名前が使用されます。
- 2 プロファイルで設定された値が使用されます。

#### QMF 図表のエクスポート

## EXPORT (CICS の場合)



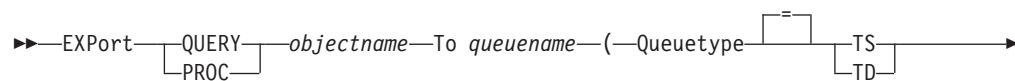
## QMF データのエクスポート

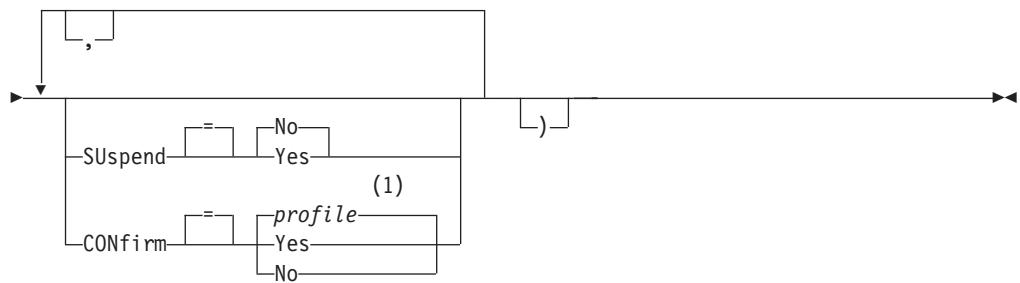


### 注:

- 1 XML フォーマットの QMF データをエクスポートするには、その前に、z/OS 変換環境を Unicode サポートに設定する必要があります。z/OS 変換サービスの使用に関する詳細は、IBM Publications Center ([www.ibm.com/shop/publications/order](http://www.ibm.com/shop/publications/order)) にある z/OS 情報を参照してください。
- 2 DATAFORMAT=IXF の場合のみ指定できます。
- 3 プロファイルで設定された値が使用されます。

## データベースからの QMF 照会またはプロシージャーのエクスポート



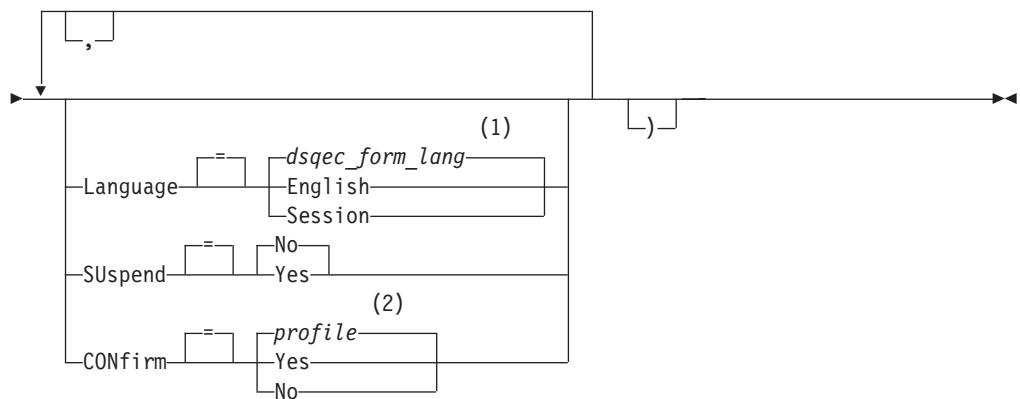


注:

- 1 プロファイルで設定された値が使用されます。

#### データベースからの QMF 書式のエクスポート

►►—EXPort—FORM—*objectname*—To *queuename*—(—Queuetype [ ] = [ ] TS [ ] TD [ ] )



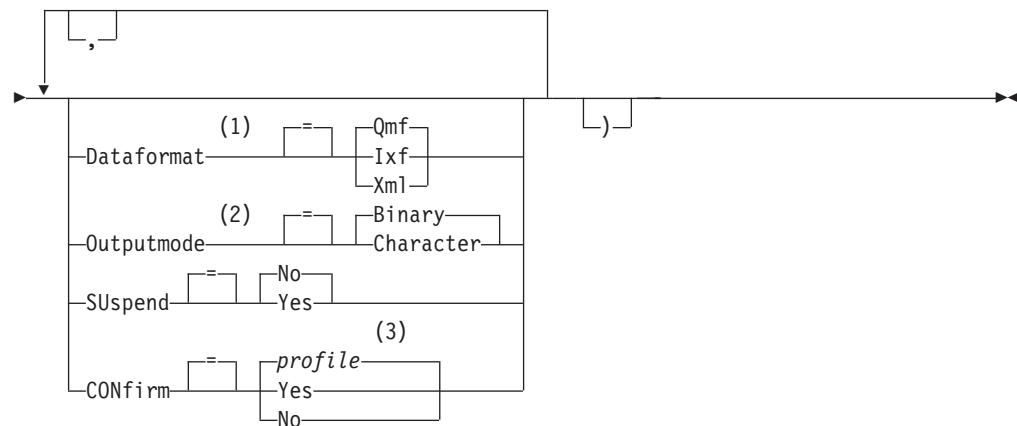
注:

- 1 この大域変数で設定された値が使用されます。
- 2 プロファイルで設定された値が使用されます。

#### データベースからの 表のエクスポート

►►—EXPort—TABLE—*objectname*—To *queuename*—(—Queuetype [ ] = [ ] TS [ ] TD [ ] )

## EXPORT (CICS の場合)



注:

- 1 XML フォーマットの QMF データをエクスポートするには、その前に、z/OS 変換環境を Unicode サポートに設定する必要があります。z/OS 変換サービスの使用に関する詳細は、IBM Publications Center ([www.ibm.com/shop/publications/order](http://www.ibm.com/shop/publications/order)) にある z/OS 情報を参照してください。
- 2 DATAFORMAT=IXF の場合のみ指定できます。
- 3 プロファイルで設定された値が使用されます。

## 説明

### objectname

データベースにある QMF オブジェクトの名前。

### tablename

データベースにある表の名前。

これは、表、視点、同義語、または別名の名前にできます。

### queuename

エクスポートされるオブジェクトを受け取るための CICS データ・キューに名前を付けます。名前の最大長は次のとおりです。

- QUEUETYPE が TD の場合、4 文字。
- QUEUETYPE が TS の場合、8 文字。

TS キューの場合、ピリオドのような特殊文字が名前に含まれているとき、名前を单一引用符で囲んでください。

キュー用の記憶域のタイプは、QUEUETYPE パラメーターで指定されたタイプと一致している必要があります。

### QUEUETYPE

オブジェクト受け取り用のデータ・キューとして使用する CICS 記憶域のタイプを指示します。QUEUETYPE にはデフォルトがないので、指定する必要があります。

**TS** CICS 一時記憶域データ・キュー。

**TD** CICS 一時データ・キュー。

**SUSPEND**

データ・キューが使用中で使用不能な場合のアクションを指定します。

**NO** エクスポート要求を取り消します。

**YES** データ・キューが使用可能になるまで待ちます。

**MEMBER**

エクスポートされるオブジェクトが、QMF 環境によって GDDM GDF (グラフィックス・データ・フォーマット) 用に定義された VSAM ファイル内のメンバーになることを指示します。メンバーがすでに存在する場合には、そのメンバーは置き換えられます。

**membername**

エクスポートされるオブジェクトを受け取るメンバー用に名前を付けます。メンバー名は 8 文字以内です。

**CONFIRM**

このコマンドがデータ・キューを変更しようとするときに、確認パネルを表示するかどうかを指示します。このオプションは CICS 一時記憶域キューの場合、すなわち QUEUETYPE=TS の場合のみ有効です。

**LANGUAGE**

移出された書式に含まれる QMF キーワードが、英語で記録されるか、または現行の NLF セッション言語で記録されるかを示します。

英語の QMF キーワードを含む QMF 書式は、すべての QMF セッションで使用できます。QMF がサポートする別の各国語の QMF キーワードを含む QMF 書式は、それと同じ各国語のセッション内でのみ使用できます。

**DATAFORMAT**

対象を移出するために使用するファイル形式を指定します。

**QMF** QMF フォーマットを使用します。これは、報告書、DATA オブジェクト、または表をエクスポートするときのデフォルトのフォーマットです。

**HTML** HTML フォーマットを使用します。報告書をエクスポートする場合だけ使用できます。

**IXF** 統合交換形式の使用。これは、DATA オブジェクトまたは表をエクスポートするときにのみ使用できます。

**XML** Extensible Markup Language (XML) フォーマットを使用します。データは、CCSID 1208 の Unicode UTF-8 フォーマットの XML 文書としてエクスポートされます。このフォーマットは、DATA オブジェクトまたは表をエクスポートするときにのみ使用でき、データまたは表を UNIX パス名にエクスポートするときの唯一のオプションです。QMF が XML データに使用するエクスポート・フォーマットの詳細は、「DB2 QMF アプリケーション開発の手引き」を参照してください。

**OUTPUTMODE**

エクスポートされるオブジェクト内の数値データの表現方法を指定します。

このオプションは、エクスポート・ファイル・フォーマットが IXF の場合だけ指定できます。

## **EXPORT (CICS の場合)**

### **BINARY**

数値列データは固有の内部フォーマットでエンコードされます。

これは移出された対象の見出しレコードに任意の数値データを適用できません。すでに文字形式で表現されています。

### **CHARACTER**

数値列データは EBCDIC の文字表現に変換されます。

### **ICUFORM**

図表形式の名前を指定します。図表形式は、データを図表に変換するために必要な仕様を含んでいます。様々な形式が、様々なタイプの図表を作成するために使用されます。

### **DSQCFORM**

QMF が提供するデフォルトの図表フォーマットの名前。

このフォーマットは、QMF 管理者がカスタマイズすることができます。カスタマイズされない場合は、棒グラフが使用されます。

### **chartname**

保管されている図表形式の名前

### **ICUCHART**

GDDM 対話式図表ユーティリティーのデフォルトの図表フォーマットを指定します。

### **BAR**

### **HISTOGRAM**

### **LINE**

### **PIE**

### **POLAR**

### **SCATTER**

### **SURFACE**

### **TOWER**

### **TABLE**

QMF が提供する図表フォーマットの名前。

## **注**

- 一時データ・キューにエクスポートする場合、キューをオープンして使用可能にし、さらに中身を空にしてから、EXPORT コマンドを実行する必要があります。CICS 一時データ・キューの詳細は、IBM Publications Center ([www.ibm.com/shop/publications/order](http://www.ibm.com/shop/publications/order)) にある CICS 情報を参照してください。
- 指定された CICS データ・キューがすでに存在する場合は、その内容がエクスポートされたオブジェクトで置き換えられます。エクスポートされるオブジェクトのフォーマットの詳細な説明については、「DB2 QMF アプリケーション開発の手引き」を参照してください。
- EXPORT コマンドの実行でエラーが起こると、空の CICS データ・キューや、一部分だけの CICS データ・キューが生じことがあります。

- 現在のデータのインポート元と同じデータ・キューにオブジェクトをエクスポートすると、場合によっては、不完全データ・プロンプトが表示されることがあります。このプロンプトでは、NO を選択して、別のデータ・キューにオブジェクトをエクスポートしてください。
- 書式をエクスポートする場合、その書式のすべての部分がエクスポートされます。

ただし、QMF は、デフォルト値を変更していない FORM.DETAIL パネル・バリエーションをすべて除去します。この方法で、同じ書式をエクスポートしてからまたインポートすることによって、不要な FORM.DETAIL バリエーションを除去することができます。

- 報告書や図表の移出時に、書式がデータと互換性がないかエラーを含む場合、エラーを含んでいる最初の書式パネルが表示され、エラーが強調表示されます。他のエラーを表示するには、現在表示されているエラーを訂正してから Check (検査) ファンクション・キーを押します。
- QMF にはサンプルの XML ファイルがいくつか付いています。これらのファイルのデータは、ブラウザーで表示できます。これらのサンプルの詳細は、「DB2 QMF アプリケーション開発の手引き」を参照してください。
- 10 進浮動小数点 (DECFLOAT) データを含むデータ・オブジェクトまたは表は、エクスポートできません。
- データベース管理者が複数行フェッチ機能を使用するように QMF をセットアップしている場合に、3 部構成の名前を使用して表またはデータをデータベースにエクスポートするときは、操作しているデータベースは (ローカルとリモートの) 両方とも DB2 (z/OS 版) 8.1.5 以降でなければなりません。そうでなければ、コマンドは失敗します。データベース管理者は、複数行フェッチをオフにできます。この機能の設定については、「TSO/CICS 用 DB2 QMF インストールおよび管理の手引き」に説明されています。
- EXPORT コマンドは長いオブジェクト名をサポートしています。プロンプト・パネルでの長い名前の入力方法の詳細は、3 ページの『プロンプト・パネルからの入力』を参照してください。

## 例

- 表をエクスポートするためのコマンド・プロンプト・パネルを表示するためには次のようにします。

```
EXPORT TABLE ?
```

- 照会を QMF 一時記憶域から一時データ・キューへエクスポートします。

```
EXPORT QUERY TO queueName (QUEUETYPE = TD)
```

- データを IXF データ・フォーマットで一時データ・キューにエクスポートするには、次のようにします。

```
EXPORT DATA TO queueName (QUEUETYPE=TD  
CONFIRM=NO DATAFORMAT=IXF
```

コマンド・キーワードを下記のように省略できます。たとえば、

```
EXP DATA TO queueName (QUEUE=TD CONF=N DATA=IXF
```

## EXPORT (CICS の場合)

4. 現行ロケーションがリモート・データ・アクセスをサポートしている DB2 であれば、オブジェクト名にロケーション修飾子を加えることにより、リモート DB2 のロケーションから表をエクスポートすることができます。

```
EXPORT TABLE VENICE.LARA.STATSTAB  
      TO queuename (QUEUETYPE = TS)
```

5. 表を IXF 文字フォーマットで TS キューにエクスポートします。

```
EXPORT TABLE KMMTABLE TO MYQUEUE  
(QUEUETYPE=TS DATAFORMAT=IXF OUTPUTMODE=CHARACTER)
```

---

## EXPORT (TSO の場合)

TSO (ISPF 付き)	TSO (ISPF なし)
X	X

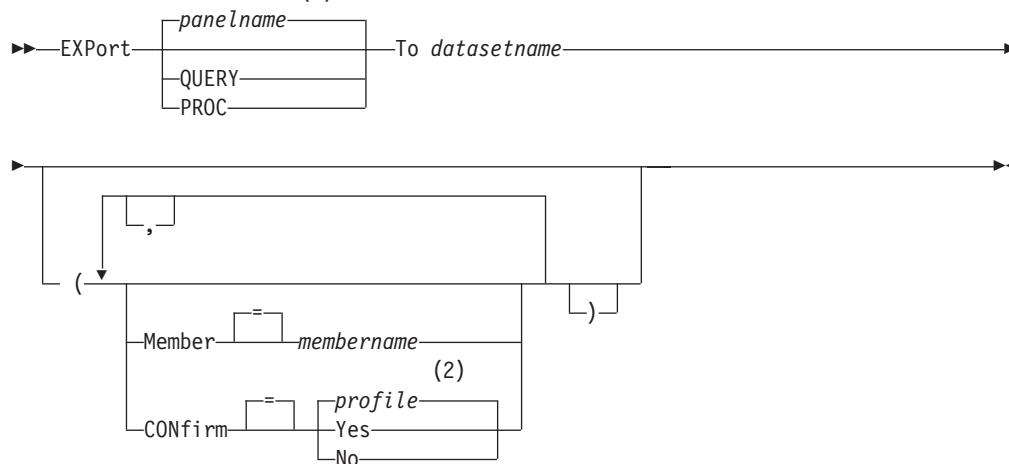
EXPORT コマンドは下記のものを送ります。

- QMF 一時記憶域にある照会、書式、プロシージャー、報告書、およびデータを TSO データ・セットへ。
- データベースにある照会、書式、プロシージャー、および表を TSO データ・セットへ。
- QMF の図表を、GDF ファイルを含む GDDM 区分データ・セットへ。
- HTML フォーマットの報告書と XML フォーマットのデータおよび表を UNIX パス名へ。

オブジェクトを QMF 一時記憶域からエクスポートするための構文と、データベースからエクスポートするための構文は異なります。

### 一時記憶域からの QMF 照会またはプロシージャーのエクスポート

(1)

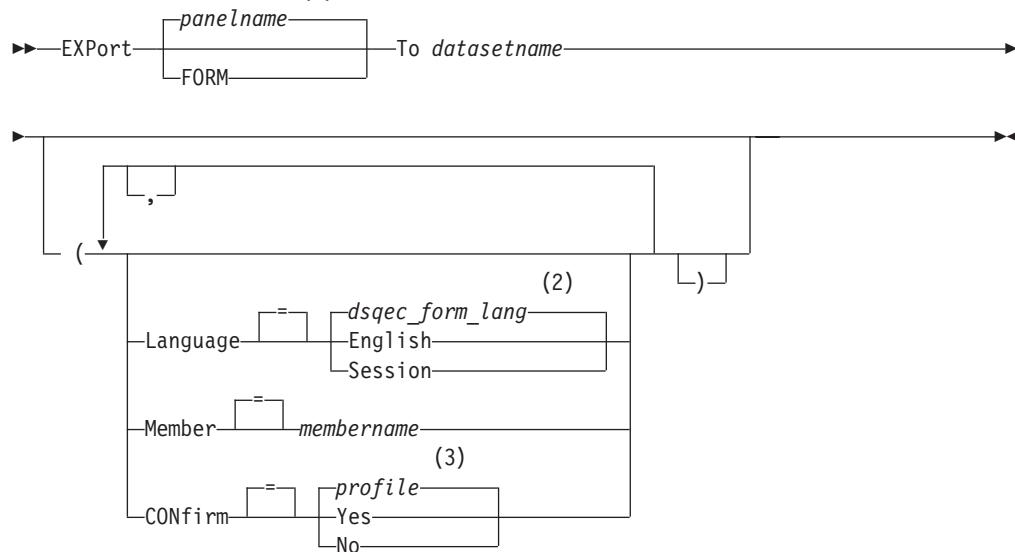


### 注:

- 1 該当する場合は、現在表示されている QMF 対象パネルの名前が使用されます。
- 2 プロファイルで設定された値が使用されます。

## 一時記憶域からの QMF 書式のエクスポート

(1)

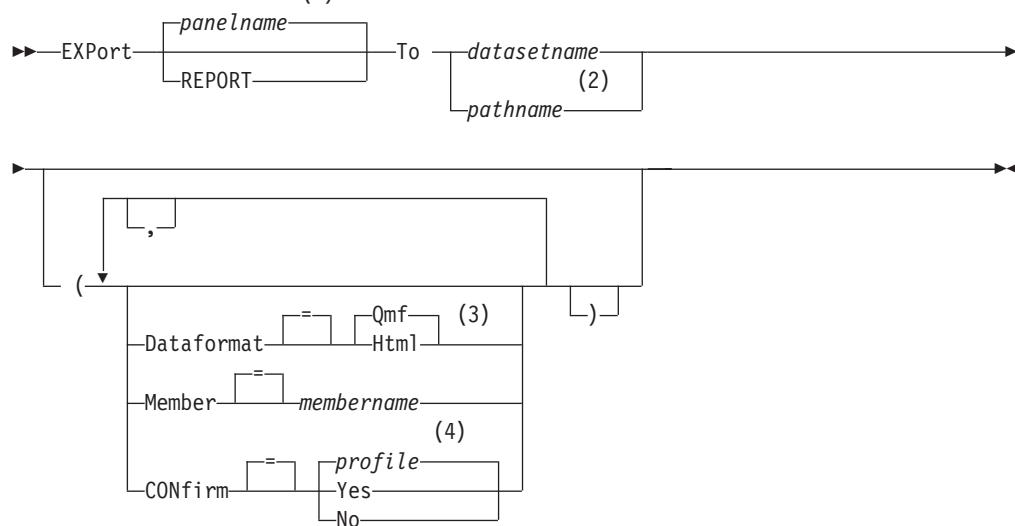


## 注:

- 1 該当する場合は、現在表示されている QMF 対象パネルの名前が使用されます。
- 2 この大域変数で設定された値が使用されます。
- 3 プロファイルで設定された値が使用されます。

## QMF 報告書の一時記憶域から TSO データ・セットまたは UNIX パス名へのエクスポート

(1)

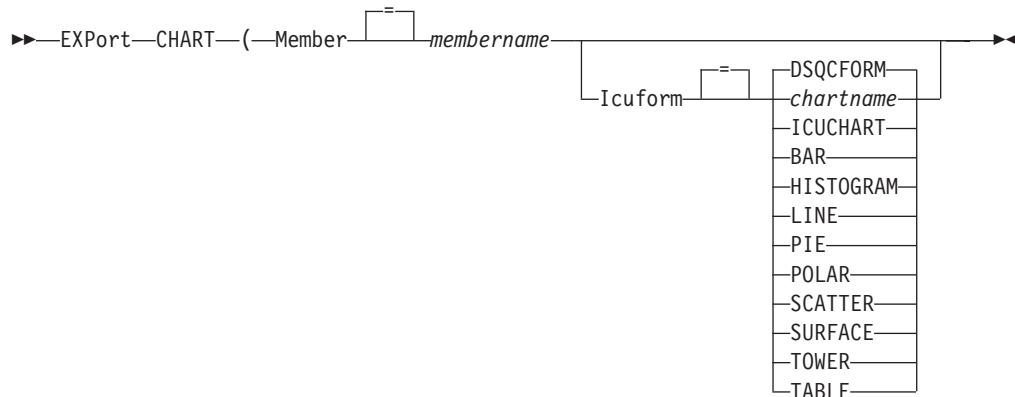


## EXPORT (TSO の場合)

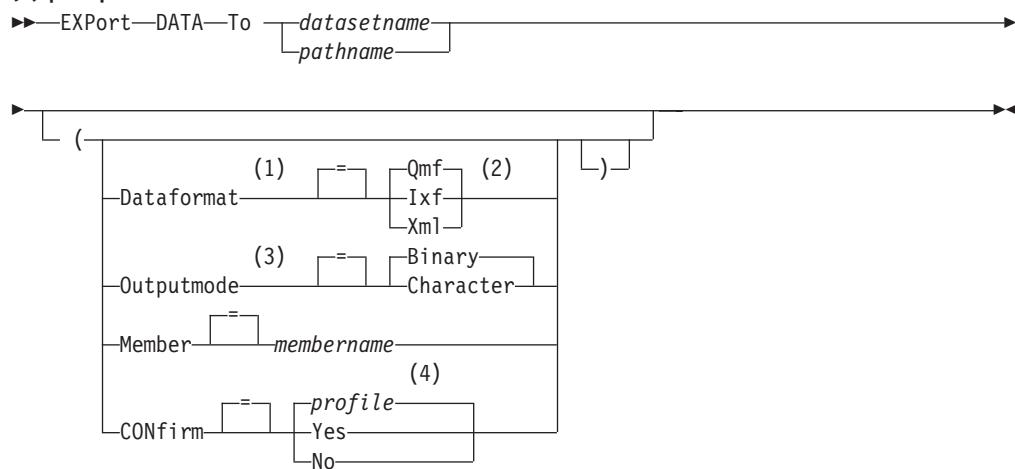
注:

- 1 該当する場合は、現在表示されている QMF 対象パネルの名前が使用されます。
- 2 DATAFORMAT=HTML の場合にのみ有効です。
- 3 報告書を UNIX パス名にエクスポートするときは、DATAFORMAT キーワードに HTML を指定します。
- 4 プロファイルで設定された値が使用されます。

### 一時記憶域からの QMF 図表のエクスポート



### QMF データの一時記憶域から TSO データ・セットまたは UNIX パス名へのエクスポート

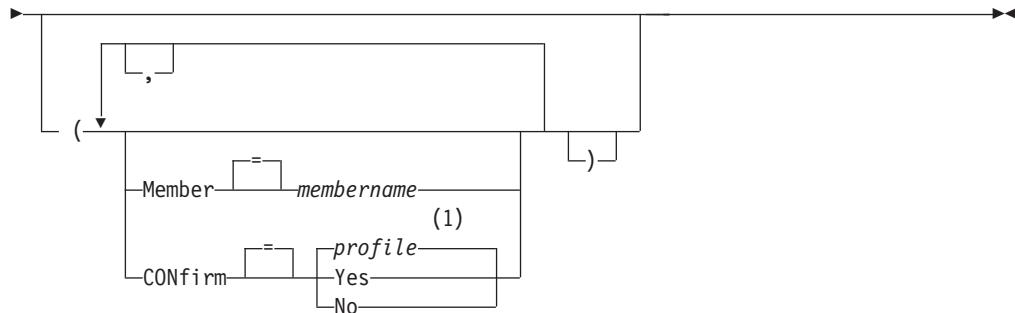
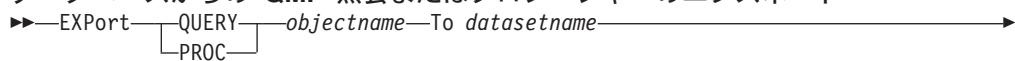


注:

- 1 XML フォーマットの QMF データをエクスポートするには、その前に、z/OS 変換環境を Unicode サポートに設定する必要があります。z/OS 変換サービスの使用に関する詳細は、IBM Publications Center ([www.ibm.com/shop/publications/order](http://www.ibm.com/shop/publications/order)) にある z/OS 情報を参照してください。
- 2 UNIX パス名にエクスポートするときは、DATAFORMAT キーワードに XML を指定します。
- 3 DATAFORMAT=IXF の場合のみ指定できます。

4 プロファイルで設定された値が使用されます。

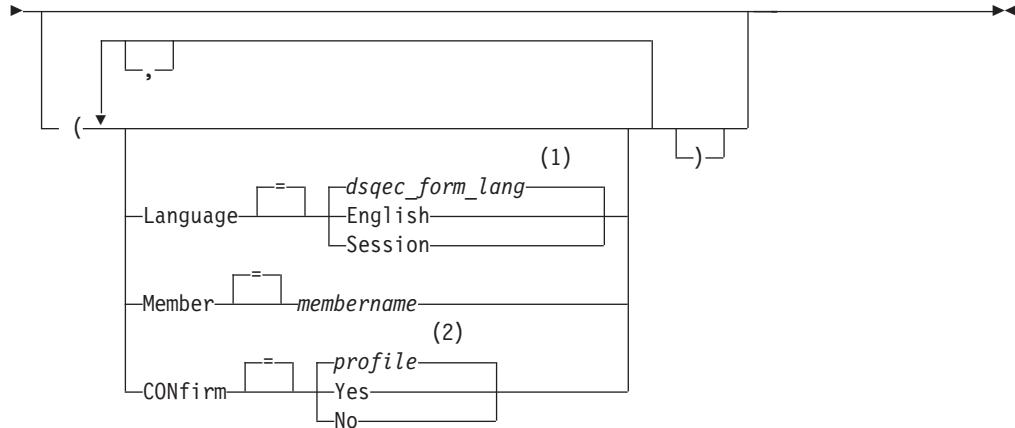
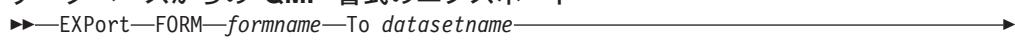
データベースからの QMF 照会またはプロシージャーのエクスポート



注：

1 プロファイルで設定された値が使用されます。

## データベースからの QMF 書式のエクスポート

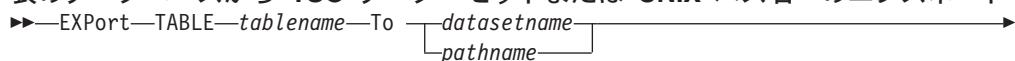


注：

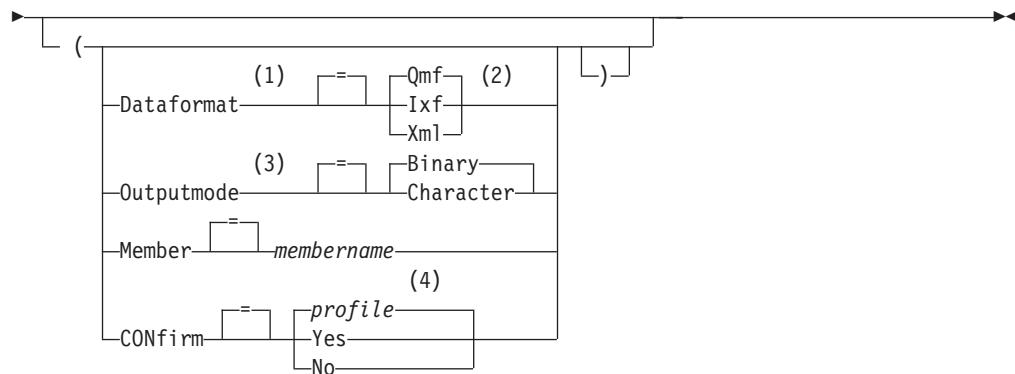
1 この大域変数で設定された値が使用されます。

2 プロファイルで設定された値が使用されます。

表のデータベースから TSO データ・セットまたは UNIX パス名へのエクスポート



## EXPORT (TSO の場合)



注:

- 1 XML フォーマットの QMF データをエクスポートするには、その前に、z/OS 変換環境を Unicode サポートに設定する必要があります。z/OS 変換サービスの使用に関する詳細は、IBM Publications Center ([www.ibm.com/shop/publications/order](http://www.ibm.com/shop/publications/order)) にある z/OS 情報を参照してください。
- 2 UNIX パス名にエクスポートするときは、DATAFORMAT キーワードに XML を指定します。
- 3 DATAFORMAT=IXF の場合のみ指定できます。
- 4 プロファイルで設定された値が使用されます。

## 説明

### objectname

データベースにある QMF オブジェクトの名前。

### tablename

データベースにある表の名前。

これは、表、視点、同義語、または別名の名前にできます。

### datasetname

エクスポートされるオブジェクトの TSO データ・セットの名前。データ・セット名は下記のいずれかの方法で指定されます。

- 単一引用符のない、部分 TSO 名。

完全に修飾された TSO データ・セット名は、TSO 接頭部を先頭の修飾子として使用し、オブジェクト・タイプを最後の修飾子として付加することにより生成されます。

- 名前全体を单一引用符で囲んだ、完全修飾の TSO データ・セット名。

データ・セット名の接頭部がユーザー自身のものと異なる場合、この形を使用する必要があります。

大域変数を使用して、エクスポートされるオブジェクトを含むことになる新規データ・セットのタイプとサイズを指定できます。

- 大域変数 DSQEC\_PO を使用して、オブジェクトを新規データ・セットのメンバーにエクスポートするときに作成する、区分データ・セットのタイプを指定します。値ゼロを指定すると、使用しているサイトのデフォルト

のタイプを使用します。値 1 は PDS データ・セットを使用し、値 2 は PDSE データ・セットを使用します。

- 大域変数 DSQEC\_DSALLOC\_DIR を使用して、新規 PDS データ・セットのメンバーをエクスポートするときのディレクトリー・ブロック数を指定します。デフォルトは 20 です。
- 大域変数 DSQEC\_DSALLOC\_PRI を使用して、1 次スペース割り振りのトラック数を指定します。デフォルトは、15 トラックです。
- 大域変数 DSQEC\_DSALLOC\_SEC を使用して、2 次スペース割り振りのトラック数を指定します。デフォルトは、105 トラックです。

これらの大域変数の詳細は、293 ページの『付録 B. QMF 大域変数表』を参照してください。

#### **pathname**

エクスポートされるオブジェクトを受け取るための UNIX ファイルに名前を付けます。UNIX パス名は引用符で囲み、250 文字以下であることを確認してください。パス名を引用符で囲まなければ、QMF は、QMF オブジェクト・タイプをパス名の終わりに追加し、パス名全体を引用符で囲みます。

#### **MEMBER**

エクスポートされるオブジェクトが TSO 区分データ・セットのメンバーになることを指示します。

図表の場合、エクスポートされるオブジェクトは、QMF 環境によって GDDM GDF (グラフィックス・データ・フォーマット) 用に定義された区分データ・セットのメンバーになることを指示します。メンバーがすでに存在する場合には、そのメンバーは置き換えられます。

#### **membername**

エクスポートされるオブジェクトを受け取るメンバー用に名前を付けます。メンバー名は 8 文字以内に制限されています。メンバー名が、データ・セット名のサフィックスとして (括弧つきで) 追加されます。

MEMBER コマンド・オプションは、UNIX パスとファイル名を指定したときは無視されます。

#### **CONFIRM**

このコマンドが既存の TSO データ・セットまたは区分データ・セット内のメンバーを置き換えるようとするときに、確認パネルを表示するかどうかを指示します。

#### **LANGUAGE**

移出された書式に含まれる QMF キーワードが、英語で記録されるか、または現行の NLF セッション言語で記録されるかを示します。

英語の QMF キーワードを含む QMF 書式は、すべての QMF セッションで使用できます。QMF がサポートする別の各国語の QMF キーワードを含む QMF 書式は、それと同じ各国語のセッション内でのみ使用できます。

#### **DATAFORMAT**

対象を移出するために使用するファイル形式を指定します。

## EXPORT (TSO の場合)

- QMF** これは、報告書、DATA オブジェクト、または表をエクスポートするときのデフォルトのフォーマットです。
- HTML** HTML は、報告書をエクスポートするときのみ指定できます。これは、UNIX パス名にエクスポートするときのデフォルトです。TSO データ・セットまたは UNIX ファイルは、Web ブラウザーで表示するために、Web サーバーに転送できます。
- IXF** 統合交換形式。これは、DATA オブジェクトまたは表をエクスポートするときにのみ使用できます。
- XML** Extensible Markup Language (XML) フォーマット。データは、CCSID 1208 の Unicode UTF-8 フォーマットの XML 文書としてエクスポートされます。このオプションは、DATA オブジェクトまたは表をエクスポートするときにのみ使用でき、データまたは表を UNIX パス名にエクスポートするときの唯一のオプションです。QMF が XML データに使用するエクスポート・フォーマットの詳細は、「DB2 QMF アプリケーション開発の手引き」を参照してください。

### OUTPUTMODE

エクスポートされるオブジェクト内の数値データの表現方法を指定します。

このオプションは、エクスポート・ファイル・フォーマットが IXF の場合だけ指定できます。

### BINARY

数値列データは固有の内部フォーマットでエンコードされます。

これは移出された対象の見出しレコードに任意の数値データを適用できません。すでに文字形式で表現されています。

### CHARACTER

数値列データは EBCDIC の文字表現に変換されます。

### ICUFORM

図表形式の名前を指定します。図表形式は、データを図表に変換するために必要な仕様を含んでいます。様々な形式が、様々なタイプの図表を作成するために使用されます。

### DSQCFORM

QMF が提供するデフォルトの図表フォーマットの名前。

このフォーマットは、QMF 管理者がカスタマイズすることができます。カスタマイズされない場合は、棒グラフが使用されます。

### chartname

保管されている図表形式の名前

### ICUCHART

GDDM 対話式図表ユーティリティーのデフォルトの図表フォーマットを指定します。

### BAR

### HISTOGRAM

### LINE

### PIE

**POLAR  
SCATTER  
SURFACE  
TOWER  
TABLE**

QMF が提供する図表フォーマットの名前。

## 注

- QMF は、既存のものでない限り、指定された名前でデータ・セットを動的に割り振ります。しかし、標準 DASD 装置を使用していないユーザーは、EXPORT コマンドを使用する前にデータ・セットを割り振る必要があります。
- 指定されたデータ・セット名またはファイル名がすでに存在している場合、そのファイルまたはデータ・セット属性が適切であれば（たとえば、レコード・フォーマットと論理レコード長に、エクスポートされるデータを収められるだけの十分な大きさがあれば）、内容がエクスポートされるオブジェクトで置き換えられながら、オブジェクトは既存のデータ・セットのメンバーにエクスポートされます。ファイルまたはデータ・セットの必要な属性、およびエクスポートされるオブジェクトのフォーマットの詳細な説明は、「DB2 QMF アプリケーション開発の手引き」を参照してください。
- EXPORT コマンドの実行でエラーが起こると、空だったり一部分しかなかったりするデータ・セットまたは区分データ・セット・メンバーが生じることがあります。
- 現在のデータのインポート元と同じデータ・セットにオブジェクトをエクスポートすると、場合によっては、不完全データ・プロンプトが表示されることがあります。このプロンプトでは、NO を選択して、別のデータ・セットにオブジェクトをエクスポートしてください。
- 書式をエクスポートする場合、その書式のすべての部分がエクスポートされます。

ただし、QMF は、デフォルト値を変更していない FORM.DETAIL パネル・バリエーションをすべて除去します。この方法で、同じ書式をエクスポートしてからまたインポートすることによって、不要な FORM.DETAIL バリエーションを除去することができます。

- 報告書や図表の移出時に、書式がデータと互換性がないかエラーを含む場合、エラーを含んでいる最初の書式パネルが表示され、エラーが強調表示されます。他のエラーを表示するには、現在表示されているエラーを訂正してから Check (検査) ファンクション・キーを押します。
- DATA オブジェクトのレコード長は 44 です。オブジェクトが、SQL 照会とプロセッサー・オブジェクトをメンバーとして含む、既存の PDS または PDSE データ・セットにエクスポートされると、次のエラー・メッセージを受け取ることができます。Record format of F should be V (F のレコード形式は V でなければなりません) このメッセージは、レコード・フォーマットは、固定ではなく可変に定義される必要があることを示しています。
- UNIX ファイルが存在しなければ、QMF は新規ファイルを作成します。ファイルは読み取りおよび書き込み用に作成されます (PATHOPTS(ORDWR,OCREAT))。ファイル所有者のファイル・アクセス権は、読み取り、書き込み、および実行に設

## EXPORT (TSO の場合)

定されます (PATHMOD(SIRWXU))。ファイルに別の属性が必要であれば、TSO ALLOCATE コマンドを使用してファイルを割り振ってから、オブジェクトをエクスポートします。パス名に指定した UNIX ファイルが存在する場合、QMF は、オブジェクトをファイルの終わりに追加します。

- パス名の大/小文字を保持するには、QMF プロファイルの中の CASE オプションに、MIXED または STRING を使用します。
- QMF にはサンプルの XML ファイルがいくつか付いており、これらのファイルのデータは、ブラウザーで表示できます。これらのサンプルの詳細は、「DB2 QMF アプリケーション開発の手引き」を参照してください。
- データベース管理者が複数行フェッチ機能を使用するように QMF をセットアップしている場合に、3 部構成の名前を使用して表またはデータをデータベースにエクスポートするときは、操作しているデータベースは (ローカルとリモートの) 両方とも DB2 (z/OS 版) 8.1.5 以降でなければなりません。そうでなければ、コマンドは失敗します。データベース管理者は、複数行フェッチをオフにできます。この機能の設定については、「TSO/CICS 用 DB2 QMF インストールおよび管理の手引き」に説明されています。
- 10 進浮動小数点 (DECFLOAT) 列を含む DATA オブジェクトまたは表は、エクスポートできません。
- EXPORT コマンドは長いオブジェクト名をサポートしています。プロンプト・パネルでの長い名前の入力方法の詳細は、3 ページの『プロンプト・パネルからの入力』を参照してください。

## 例

1. 表をエクスポートするためのコマンド・プロンプト・パネルを表示するためには次のようにします。

```
EXPORT TABLE ?
```

2. リモート作業単位を使用している場合、オブジェクト (表、書式、プロシージャ、照会、または報告書) を現行ロケーションから、QMF が稼動中のシステムにあるデータ・セットにエクスポートできます。たとえば、

```
EXPORT PROC KATIE.PANELID TO dataset
```

3. 現行ロケーションがリモート・データ・アクセスをサポートしている DB2 であれば、オブジェクト名にロケーション修飾子を加えることにより、リモート DB2 のロケーションから表をエクスポートすることができます。

```
EXPORT TABLE VENICE.LARA.STATSTAB TO dataset
```

4. TSO 接頭部が TOM の場合に、TSO データ・セット 'TOM.LOREN.QUERY (GAMMA)' を使用します。

```
EXPORT QUERY FIRSTQ TO LOREN (MEMBER=GAMMA)
```

TSO 接頭部がないときは、TSO ユーザー ID が使用されます。

ユーザーの接頭部がブランクに設定されているときは、TSO 名に接頭部は付けられません。

5. データを IXF 文字フォーマットでエクスポートします。

```
EXPORT DATA TO JBLP  
(CONFIRM=NO DATAFORMAT=IXF OUTPUTMODE=CHARACTER)
```

6. 書式を現行セッション言語でエクスポートします。

EXPORT FORM TO MYFORM (LANGUAGE=SESSION)

7. 現行ロケーションにある FORMA という書式を、QMF が稼働しているシステムにある FORMS というデータ・セットにコピーするには、次のようにします。

EXPORT FORM FORMA TO FORMS

8. 3 部名をサポートしないリモート・データベースから表をエクスポートするには、まずそのデータベースに接続しなければなりません。

CONNECT TO VENICE

次に、表をエクスポートします。

EXPORT TABLE JULIA.STATUS TO NONSTD

9. TOKYO の DB2 データベースにある OKAMOTO.STATUS という表を、QMF が稼働しているシステムの YOURDATA というデータ・セットにコピーします。

EXPORT TABLE TOKYO.OKAMOTO.STATUS TO YOURDATA

10. Q.STAFF 表を UNIX ファイル '/u/DEPTJ49/pernal/mystaff.personnel' にエクスポートするには、プロファイルの CASE オプションを MIXED または STRING に設定して、次のコマンドを発行します。

EXPORT TABLE Q.STAFF TO '/u/DEPTJ49/pernal/mystaff.personnel' (DATAFORMAT=XML)

11. 報告書を、引用符で囲まれていない UNIX パス名 /u/QMFDEV/Robin/reports/test にエクスポートするには、QMF プロファイルの CASE オプションを MIXED または STRING に設定して、次のコマンドを発行します。

EXPORT REPORT TO /u/QMFDEV/Robin/reports/test (DATAFORMAT=HTML)

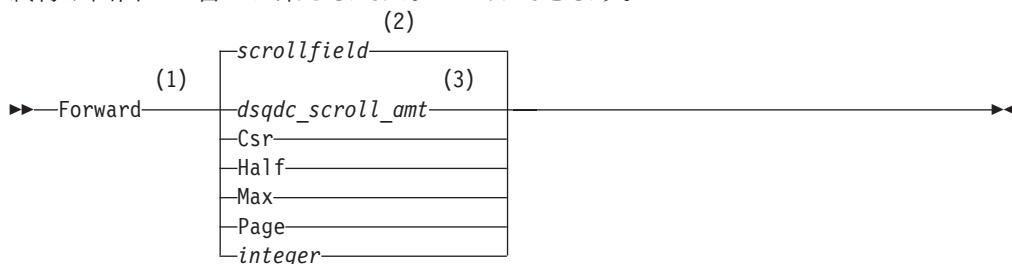
このコマンドは、報告書を次の絶対パス名にエクスポートします。

/u/QMFDEV/Robin/reports/test.REPORT

## FORWARD

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X	X	X

FORWARD コマンドは、スクロール可能域の下方に向かってスクロールします。最終行が画面の一番上に来るまでスクロールできます。



注:

- 1 活動中のパネルに画面移動フィールドがある場合にのみ画面移動量を指定します。その他のすべての場合は、PAGE が想定されます。

## FORWARD

- 2 画面移動フィールドに表示される値が使用されます。この値は、大域変数 DSQDC\_SCROLL\_AMT にも保持されます。
- 3 この大域変数で設定された値が使用されます。

## 説明

<b>CSR</b>	カーソルがスクロール可能域の先頭に置かれます。
<b>HALF</b>	スクロール可能域の半分だけ下方へスクロールするか、(それ以前に末尾に到達するときは) 末尾までスクロールします。
<b>MAX</b>	スクロール可能域の末尾までスクロールします。FORWARD MAX は BOTTOM と同じです。
<b>PAGE</b>	スクロール可能域分だけ下方へスクロールするか、(それ以前に末尾に到達するときは) 末尾までスクロールします。
<b>integer</b>	パネル上でこの行数分下方に画面移動を行います (1 から 9999 までの範囲のすべての数)。

## 注

- MAX は現行のコマンドの間だけ有効です。この値はコマンドが完了すると SCROLL フィールドには残りません。大域変数 DSQDC\_SCROLL\_AMT をこの値を設定することはできません。
- 書式パネルの後書きテキストで下方スクロールを行なうには、後書きテキストが指定されているパネル部分にカーソルを合わせてから、FORWARD コマンドを入れてください。

---

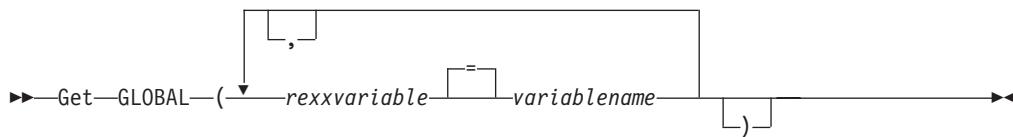
## GET GLOBAL

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X	X	X

線形構文の GET GLOBAL コマンドは、QMF 大域変数の値を、REXX で書かれたアプリケーションおよびプロシージャーの中の REXX 変数に割り当てます。

拡張構文の GET GLOBAL コマンドによって、アプリケーション・プログラム (REXX 以外の言語で作成されたもの) が、呼び出し可能インターフェースを使用して QMF 大域変数プールからのデータにアクセスできます。拡張構文の GET GLOBAL コマンドの詳細は、「DB2 QMF アプリケーション開発の手引き」を参照してください。

### 線形構文 (REXX でのみ使用)



## 説明

### **rexxyvariable**

ロジックを持つプロセッサーまたは REXX アプリケーションの中の REXX 変数の名前。

### **variablename**

QMF 大域変数の名前。

## 注

このコマンドは、QMF コマンド行では有効ではありません。

GET GLOBAL コマンドで複数の変数をアクセスする場合は、下記の規則が適用されます。

- *rexxyvariable* と *variablename* の間の等号はオプションです。
- 大域変数と値のペアごとの間のコンマはオプションです。
- *rexxyvariable* と *variablename* の間の区切り文字は、1 つ以上のブランクか、またはブランクを伴うか伴わない 1 つの等号でなければなりません。
- 大域変数と値のペア (*rexxyvariable* と *variablename* の両方) ごとの間の区切り文字は、1 つ以上のブランクであるか、1 つのコンマ (そのあとにブランクが付く場合も付かない場合もある) でなければなりません。
- 各 REXX 変数は、関連付けられた変数名を 1 つだけ持つことができます。

GET GLOBAL コマンドには、関連したコマンド・プロンプト・パネルがありません。このコマンドの場合、コマンド・プロンプトは使用不能です。

QMF では要求されませんが、すべての変数名に大文字を使用することをお勧めします。

同義語が指定されていなければ、QMF は get global (小文字) をエラーと見なします。システム間の整合性を保つため、この QMF コマンドとその他のすべての QMF コマンドを、(QMF または REXX 手順、あるいは呼び出しインターフェースのどれにおいても) 大文字で指定してください。

- REXX で書かれた QMF アプリケーションで、次の例は QMF 大域変数 DSQAITEM の値を REXX 変数 ITEM に割り当てます。

ADDRESS QRW "GET GLOBAL (ITEM = DSQAITEM"

- REXX で書かれた QMF プロセッサーで、次の例は QMF 大域変数 DSQCIQMG の値を次の REXX 変数 MSG に割り当てます。

"GET GLOBAL (MSG = DSQCIQMG"

## GETQMF マクロ

TSO (ISPF あり)	TSO (ISPF なし)	CICS
X	X	

## GETQMF マクロ

GETQMF は編集マクロです。 QMF コマンドではありません。 QMF 報告書を文書に挿入します。

編集セッションで GETQMF マクロを使用すれば、編集セッションにいながら、編集中の文書に QMF 報告書を挿入できます。ただし、その QMF 報告書は前もって QMF セッションで印刷してからでないと、文書には挿入できません。

*GETQMF type option name*

**type** SCRIPT/VIS 制御ワードを挿入するかどうかを指定します。

**DCF** SCRIPT/VIS 文書の場合。文書構成プログラム (DCF) が、その QMF 報告書の前後に SCRIPT/VIS 制御ワードを挿入します。また、プリンターの改ページ・コードを SCRIPT/VIS の改ページ・コードで置き換え、各ページの見出しと脚注の位置に SCRIPT/VIS 制御ワードを挿入します。

**ASIS** QMF 報告書の場合そのまま。TYPE の指定を省略すると、ASIS と見なされます。

**option name**

新しい報告書を作成するのか、既存の報告書を挿入するのかを指定します。

**USEQMF**

報告書を印刷するプロシージャーを用いて、動的に QMF 報告書を作成します。name は、保管されているプロシージャーの名前です。

**DSN** 既存の報告書を TSO データ・セットから挿入します。name は、その報告書が入っている TSO データ・セットの名前です。

---

## HELP

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X	X	X

HELP コマンドは、QMF に関する情報を表示します。2 種類の形式のヘルプ情報が使用可能です。

トピック・ヘルプ

►►Help————►►

メッセージ・ヘルプ

►►Help—messageid————►►

## 説明

**messageid**

QMF メッセージ ID。QMF は、指定されたメッセージ ID に関連付けられたメッセージ・ヘルプを見つけようとなります。見つかった場合は、ヘルプが表示されます。見つからなかった場合は、エラー・メッセージが表示されま

す。 QMF バッチ・ジョブでは、メッセージは L レベルのトレース・データ・セットまたはデータ・キューの中にメッセージ番号を含んでいます。

メッセージ ID は、「DSQ」の 3 文字で始め、そのあとに 5 桁の番号を続ける必要があります (たとえば、DSQ20114)。「DB2 QMF メッセージおよびコード」に、メッセージ番号とテキストがリストされています。

## 注

*messageid* パラメーターなしで HELP コマンドを実行した場合に、どのような情報が表示されるかは、そのときの画面内容によって異なります。

### QMF ホーム・パネル:

HELP を発行すると、コマンド、図表、プロシージャー、報告書、書式、といった QMF フィーチャーおよび機能に関するトピックのリストが表示されます。

### エラー・メッセージを伴うパネル:

HELP を発行すると、エラー・メッセージの情報が表示されます。

### 他のヘルプ・パネル :

HELP を発行すると、表示されたパネルの詳細情報が表示されます。次のパネルについては、それぞれ別個の HELP のシーケンスがあります。

- QUERY
- PROC
- PROFILE
- REPORT
- すべての書式パネル
- データベース・オブジェクト・リスト
- 大域変数リスト
- 指示照会
- 表エディター

HELP とともにメッセージ ID を指定すると、メッセージに関する情報が表示されます。たとえば、エラー・メッセージ DSQ20047 の情報を表示する場合は、次のコマンドを発行します。

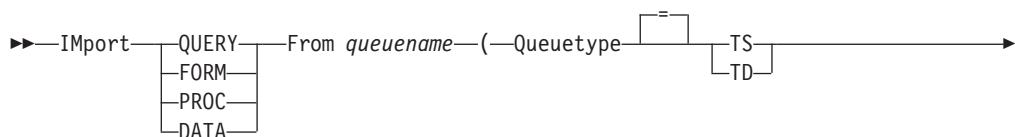
HELP DSQ20047

---

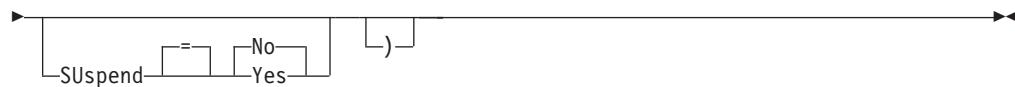
## IMPORT (CICS の場合)

IMPORT コマンドは、CICS データ・キューの内容を QMF 一時記憶域またはデータベースにコピーします。

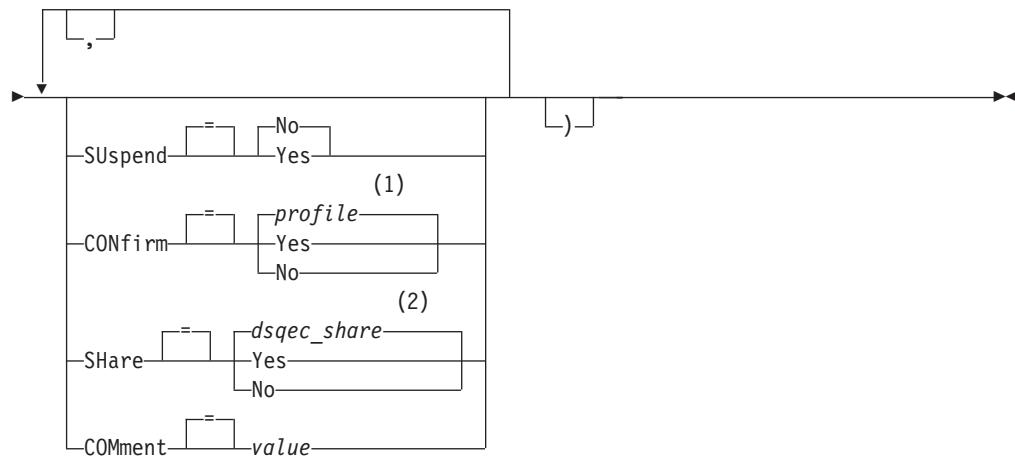
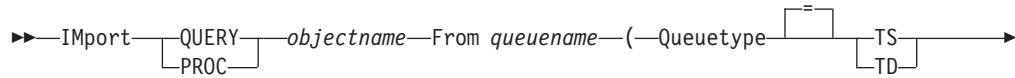
### 一時記憶域への QMF オブジェクトのインポート



## IMPORT (CICS の場合)



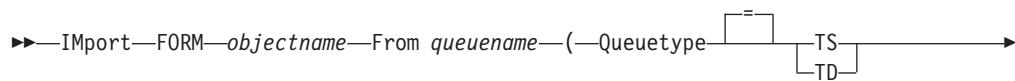
データベースへの QMF 照会またはプロシージャーのインポート

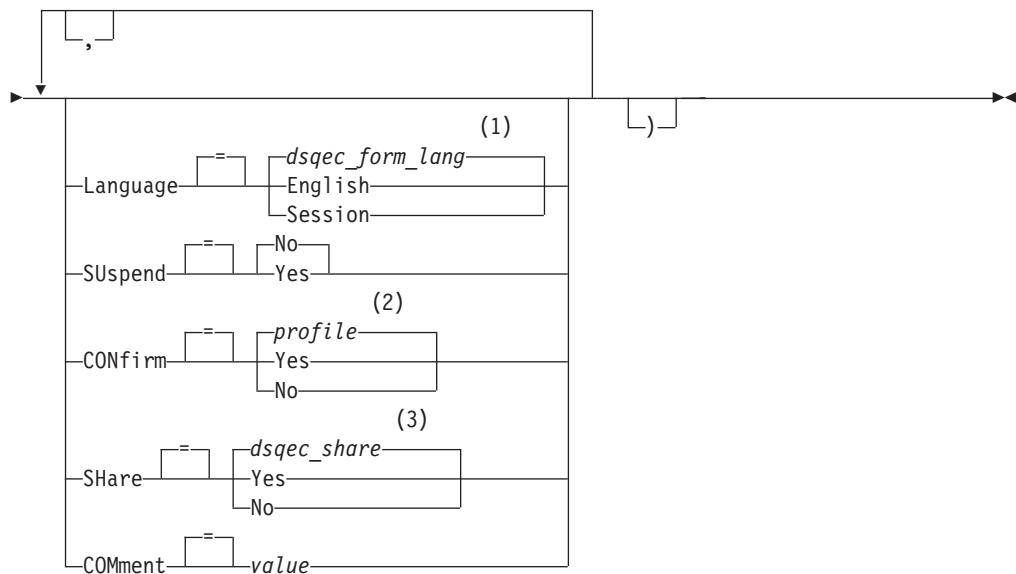


注:

- 1 プロファイルで設定された値が使用されます。
- 2 置換される対象については、現在の値は変更されないままになります。それ以外の場合は、この大域変数で設定された値が使用されます。

データベースへの QMF 書式のインポート



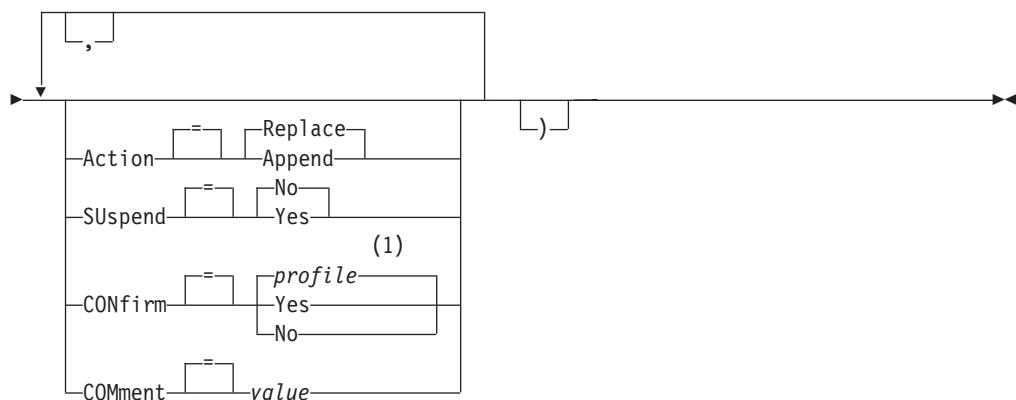


**注:**

- 1 この大域変数で設定された値が使用されます。
- 2 プロファイルで設定された値が使用されます。
- 3 置換される対象については、現在の値は変更されないままになります。それ以外の場合は、この大域変数で設定された値が使用されます。

#### データベースへの表のインポート

►—IMport—TABLE—tablename—From queuename—(—Queuetype—) TS TD



**注:**

- 1 プロファイルで設定された値が使用されます。

## 説明

### objectname

データベース内の QMF 対象の名前です。

## IMPORT (CICS の場合)

### tablename

データベース内の表の名前です。

既存のデータベース・オブジェクトの場合、これは、表、視点、同義語、または別名の名前にすることができます。

### queuename

QMF オブジェクトが入っている CICS データ・キューの名前。名前の最大長は次のとおりです。

- QUEUETYPE が TD の場合、4 文字。
- QUEUETYPE が TS の場合、8 文字。

TS キューの場合、ピリオドのような特殊文字が名前に含まれているとき、名前を单一引用符で囲んでください。

### QUEUETYPE

QMF オブジェクトが入っているデータ・キューのタイプを指示します。

QUEUETYPE にはデフォルトがないので、指定する必要があります。

**TS** CICS 一時記憶域データ・キュー。

**TD** CICS 一時データ・キュー。

### ACTION

データベース表全体を、移入されたデータに置き換えるのか、または移入されたデータを既存の表に追加するのかを示します。

### LANGUAGE

移入する書式内の QMF キーワードを英語で記録するか、現在の NLF セッション言語で記録するかを示します。

英語の QMF キーワードを含む QMF 書式は、すべての QMF セッションで使用できます。 QMF がサポートする別の各国語の QMF キーワードを含む QMF 書式は、それと同じ各国語のセッション内でのみ使用できます。

### SUSPEND

データ・キューが使用中または使用不可の場合のアクションを指定します。

**NO** インポート要求を取り消します。

**YES** データ・キューが使用可能になるまで待ちます。

### CONFIRM

このコマンドがデータベース内のオブジェクトを置き換えようとするときに、確認パネルを表示するかどうかを指示します。

### SHARE

他の QMF ユーザーが移入された対象にアクセスできるかどうかを決定します。

### COMMENT

移入される対象に関する注釈を保管します。

**value** 注釈の内容を構成する文字ストリングです。

プランク文字を含む値は、区切り文字で囲む必要があります。注釈値のための有効な区切り文字は単一引用符、括弧、および二重引用符です。 QMF コマンド行またはプロシージャーの中で IMPORT

コマンドを使用してオブジェクトと一緒にコメントを保管しようとする場合、コメント・テキストは 78 文字まで許されます。IMPORT コマンド・プロンプト・パネルを使用してコメントを入力する場合、コメントの長さは 57 文字までです。

## 注

- CICS では、TSO データ・セットを使用しないでください。
- QMF 管理者は他のユーザー用に QMF オブジェクトをインポートすることができます。
- IMPORT コマンドを実行する前に、キューに単一の完全な QMF オブジェクトを入れておく必要があります。
- データが移入されると、新しい書式が作成されます。一時記憶域内の既存の書式は置き換えられます。
- リモート・ロケーションに接続されている場合、そのサーバーにある表は読み取り専用です。オブジェクトをそのデータベースにインポートすることはできません。
- データベースに対象を移入するときに、指定した名前と同じ名前の対象がすでに存在していた場合、QMF は以下の条件に基づいて対象を置き換えます。
  - 書式は書式のみを置き換えることができる。
  - 手順は手順のみを置き換えることができる。
  - 照会は照会のみを置き換えることができる。
  - 表は類似の表対象のみを置き換えることができる。

同類の表とは、列の数が同じで、対応する列が同じデータ・タイプと長さをもつ表のことです。列名とラベルは一致する必要はありません。

- XML 列を含む表をインポートする場合、列には整形式 XML 文書が含まれていることを確認してください。インポートするデータは、「DB2 QMF アプリケーション開発の手引き」に説明された QMF XML フォーマットに従っている必要があります。
- 既存の表に移入を行うとき、列名とラベルは変更されません。表が存在しない場合は、移入された対象の列名およびラベルを使用して新しい表が作成されます。
- 3 部分名を使用しても、他のユーザーの所有している表、あるいはリモートにある表のコメントを置き換えることはできません。
- CICS の場合、他のユーザーに対して QMF パフォーマンスに悪影響が出るので、IMPORT コマンドの使用は控えめにしてください。
- インポート中にエラーが発生すると、CICS TD キューの内容は廃棄されます。現在キューにあるオブジェクトに合った、正しいオブジェクト・タイプを使用するようしてください。ミスマッチがあると、空のキューができ、オブジェクトはインポートされません。
- QMF は CICS TD キューと CICS TS キューを区別して取り扱います。

### 一時データ・キュー

QMF は一時データ・キュー全体をインポートするため、オブジェクトを表示するまで長い遅延が生ずることがあります。オブジェクト全体が記憶域や予備域に適合する必要があります。

## IMPORT (CICS の場合)

- 区画内 TD キューには 32K までの行データを保持できます。
- 区画外 TD キューは、オブジェクトを保持するのに必要なだけの大きさにすることができます。

### 一時記憶域キュー

一時記憶域キューには 32K までの行データを保持できます。CICS TS キューから DATA をインポートするとき、QMF は、DSQSIROW パラメーターに示された行数を取り出したあと、報告書を表示するため、一時停止します。インポートを完了させるには、BOTTOM コマンドを出してください。報告書を完了させるのに十分な記憶域がなければ、QMF RESET コマンドを使用してデータをリセットしてください。このコマンドの詳細は、105 ページの『RESET オブジェクト』を参照してください。

- IMPORT コマンドは長いオブジェクト名をサポートしています。プロンプト・パネルでの長い名前の入力方法の詳細は、3 ページの『プロンプト・パネルからの入力』を参照してください。
- 10 進浮動小数点 (DECFLOAT) 列を含むデータは、インポートできません。

## 例

1. QMF IMPORT コマンドのための指示パネルを表示するには次のようにします。

```
IMPORT ?
```

2. データ・キュー VTAB を表 REYNOLDS.VISIONS にコピーします。

```
IMPORT TABLE REYNOLDS.VISIONS FROM VTAB (QUEUETYPE=TD)
```

3. データ・キュー QUERY.A を照会 REYNOLDS.QUERYA にコピーします。

```
IMPORT QUERY REYNOLDS.QUERYA  
      FROM 'QUERY.A' (QUEUETYPE=TS)
```

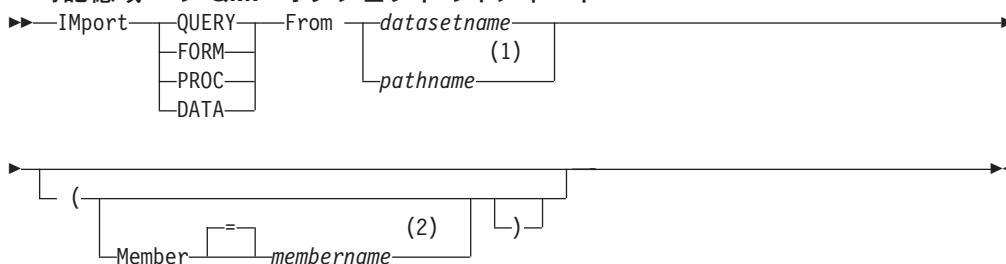
---

## IMPORT (TSO の場合)

TSO (ISPF 付き)	TSO (ISPF なし)
X	X

IMPORT コマンドは、TSO データ・セットまたは UNIX ファイルの内容を QMF 一時記憶域またはデータベースにコピーします。

### 一時記憶域への QMF オブジェクトのインポート



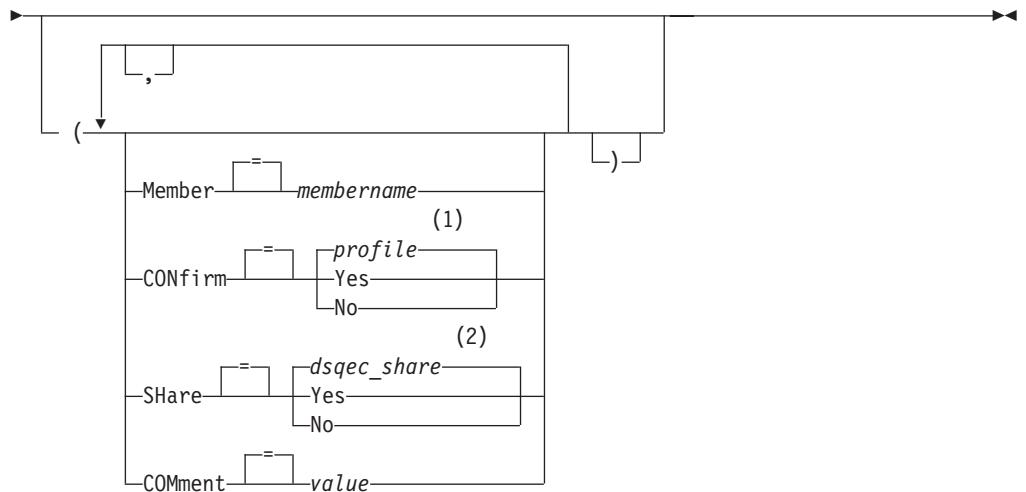
## IMPORT (TSO の場合)

注:

- 1 QMF は、オブジェクトが DATA で、データが XML フォーマットである場合にのみパス名を受け入れます。
- 2 TSO データ・セットからインポートするときにのみ指定できます。

データベースへの QMF 照会またはプロシージャーのインポート

►►IMport—QUERY—*objectname*—From *datasetname*—►►



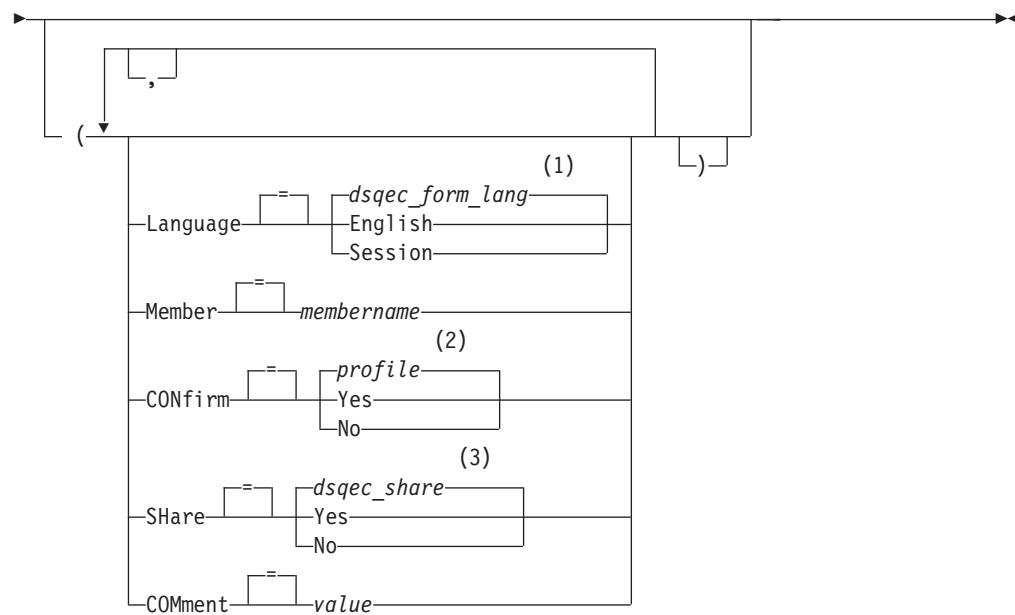
注:

- 1 プロファイルで設定された値が使用されます。
- 2 置換される対象については、現在の値は変更されないままになります。それ以外の場合は、この大域変数で設定された値が使用されます。

データベースへの QMF 書式のインポート

►►IMport—FORM—*objectname*—From *datasetname*—►►

## IMPORT (TSO の場合)

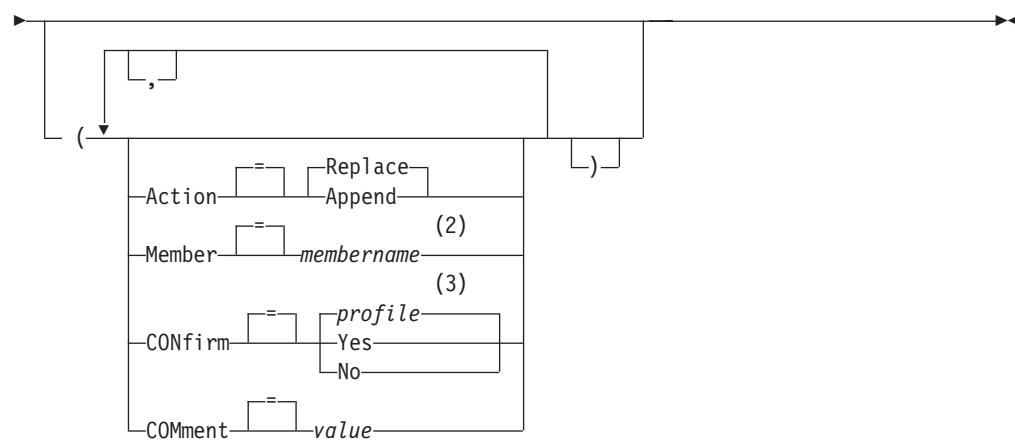


注:

- 1 この大域変数で設定された値が使用されます。
- 2 プロファイルで設定された値が使用されます。
- 3 置換される対象については、現在の値は変更されないままになります。それ以外の場合は、この大域変数で設定された値が使用されます。

## データベースへの表のインポート

➤—IMport—TABLE—tablename—From datasetname  
 (1)  
 pathname



注:

- 1 QMF は、表が XML フォーマットである場合にのみパス名を受け入れます。
- 2 TSO データ・セットからインポートするときにのみ指定できます。
- 3 プロファイルで設定された値が使用されます。

## 説明

### **datasetname**

コピーする TSO データ・セット。データ・セット名は下記のいずれかの方  
法で指定されます。

- 単一引用符のない、部分 TSO 名。

完全に修飾された TSO データ・セット名は、TSO 接頭部を先頭の修飾  
子として使用し、オブジェクト・タイプを最後の修飾子として付加するこ  
とにより生成されます。

- 名前全体を単一引用符で囲んだ、完全修飾の TSO データ・セット名。

引用符は、データ・セット名の接頭部がユーザーのものと異なる場合に使  
用する必要があります。

### **pathname**

そこからオブジェクトを取り出す UNIX ファイルを指定します。UNIX パ  
ス名は引用符で囲み、250 文字以下であることを確認してください。パス名  
を引用符で囲まなければ、QMF は、QMF オブジェクト・タイプをパス名  
の終わりに追加し、パス名を引用符で囲みます。

### **objectname**

データベース内の QMF 対象の名前です。

### **tablename**

データベース内の表の名前です。

既存のデータベース・オブジェクトの場合、これは、表、視点、同義語、ま  
たは別名の名前にすることができます。

### **ACTION**

データベース表全体を、移入されたデータに置き換えるのか、または移入さ  
れたデータを既存の表に追加するのかを示します。

### **LANGUAGE**

移入する書式内の QMF キーワードを英語で記録するか、現在の NLF セッ  
ション言語で記録するかを示します。

英語の QMF キーワードを含む QMF 書式は、すべての QMF セッション  
で使用できます。 QMF がサポートする別の各国語の QMF キーワードを  
含む QMF 書式は、それと同じ各国語のセッション内でのみ使用できます。

### **MEMBER**

インポートされるオブジェクトが TSO 区分データ・セットのメンバーであ  
ることを指示します。

### **membername**

インポートするメンバーの名前。メンバー名は 8 文字以内に制限さ  
れています。メンバー名が、データ・セット名のサフィックスとし  
て (括弧つきで) 追加されます。

### **CONFIRM**

このコマンドがデータベース内のオブジェクトを置き換えようとするとき  
に、確認パネルを表示するかどうかを指示します。

## IMPORT (TSO の場合)

### SHARE

他の QMF ユーザーが移入された対象にアクセスできるかどうかを決定します。

### COMMENT

移入される対象に関する注釈を保管します。

**value** 注釈の内容を構成する文字ストリングです。

プランク文字を含む値は、区切り文字で囲む必要があります。注釈値のための有効な区切り文字は単一引用符、括弧、および二重引用符です。 QMF コマンド行またはプロシージャーの中で IMPORT コマンドを使用してオブジェクトと一緒にコメントを保管しようとする場合、コメント・テキストは 78 文字まで許されます。

IMPORT コマンド・プロンプト・パネルを使用してコメントを入力する場合、コメントの長さは 57 文字までです。

## 注

- QMF 管理者は他のユーザー用に QMF オブジェクトをインポートすることができます。
- データが移入されると、新しい書式が作成されます。一時記憶域内の既存の書式は置き換えられます。
- データベースに対象を移入するときに、指定した名前と同じ名前の対象がすでに存在していた場合、QMF は以下の条件に基づいて対象を置き換えます。
  - 書式は書式のみを置き換えることができる。
  - 手順は手順のみを置き換えることができる。
  - 照会は照会のみを置き換えることができる。
  - 表は類似の表対象のみを置き換えることができる。

同類の表とは、列の数が同じで、対応する列が同じデータ・タイプと長さをもつ表のことです。列名とラベルは一致する必要はありません。

- 既存の表に移入を行うとき、列名とラベルは変更されません。表が存在しない場合は、移入された対象の列名およびラベルを使用して新しい表が作成されます。
- 対象は遠隔ロケーションに移入できます。最初に QMF CONNECT コマンドを使用して遠隔ロケーションを現行ロケーションにしてから、IMPORT コマンドを使用してください。
- 現行ロケーションが DB2 (z/OS 版) サーバーであれば、表名として 3 パート名を指定することにより、リモート・ロケーションにある既存の表にインポートすることができます。(ただし、この方法では、新しい表、または QMF オブジェクトはインポートできません。) データベース管理者が複数行フェッチ機能を使用するように QMF をセットアップしている場合、3 部構成の名前を使用するときは、操作しているデータベースは(ローカルとリモートの)両方とも DB2 (z/OS 版) 8.1.5 以降でなければなりません。そうでなければ、コマンドは失敗します。データベース管理者は、複数行フェッチをオフにできます。この機能の設定については、「*TSO/CICS 用 DB2 QMF インストールおよび管理の手引き*」に説明されています。
- 所有していない表または遠隔表の注釈を、3 部分名を使用して置き換えることはできません。

- XML 列を含む表をインポートする場合、列には整形式 XML 文書が含まれていることを確認してください。インポートするデータは、「DB2 QMF アプリケーション開発の手引き」に説明された QMF XML フォーマットに従っている必要があります。
- IMPORT コマンドは長いオブジェクト名をサポートしています。プロンプト・パネルでの長い名前の入力方法の詳細は、3 ページの『プロンプト・パネルからの入力』を参照してください。
- 10 進浮動小数点 (DECFLOAT) 列を含むデータまたは表は、インポートできません。

## 例

- QMF IMPORT コマンドのための指示パネルを表示するには次のようにします。

```
IMPORT ?
```

- TSO 接頭部が JULIA で、区分データ・セット 'JULIA.LOREN.QUERY (GAMMA)' のメンバーをデータベースにコピーし、その名前を FIRSTQ にし対場合、次のようにします。

```
IMPORT QUERY FIRSTQ FROM LOREN (MEMBER=GAMMA)
```

- データ NEW.ROWS を表 MYTABLE に追加するには、次のようにします。

```
IMPORT TABLE MYTABLE FROM NEW.ROWS (ACTION=APPEND)
```

- 遠隔データベース・サーバー (VENICE) に表を移入するためには、まず次を指定してそのロケーションに接続します。

```
CONNECT TO VENICE
```

次に、表をインポートします。

```
IMPORT TABLE LARA.STATSTAB FROM YOURDATA
```

- 現行ロケーションが DB2 (z/OS 版) サーバーであり、QMF が稼働しているシステムからデータ・セット ('G7 STATS TABLE') を、リモート・データベースのロケーション (TOKYO) にある既存の表 (OKAMOTO.STATUS) にコピーします。

```
IMPORT TABLE TOKYO.OKAMOTO.STATUS FROM 'G7 STATS TABLE'
```

- ユーザーが QMF 管理者 (QMADM) である場合に、別のユーザー (JEAN) の書式をインポートするには、次のコマンドを発行します。

```
IMPORT FORM JEAN.REPORT12 FROM FORMTEST (COMMENT='12 MONTH FORMAT')
```

- UNIX ファイル '/u/DEPTJ49/pernal/mystaff.personnel' からデータをインポートするには、次のコマンドを発行します。

```
IMPORT DATA FROM '/u/DEPTJ49/pernal/mystaff.personnel'
```

小文字を維持する場合は、QMF プロファイルの CASE オプションが STRING または MIXED に設定されていることを確認してください。

---

## INSERT

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X	X	X

INSERT コマンドの働きは下記のとおりです。

## INSERT

- テキスト行を FORM.PAGE、FORM.FINAL、FORM.BREAKN、または FORM.DETAIL パネルに挿入する
- FORM.MAIN パネルまたは FORM.COLUMNS パネルに 1 行の列記述を挿入します。
- FORM.CALC パネルまたは FORMS.CONDITIONS パネルに 1 行の報告書計算式を挿入します。
- SQL 照会、指示照会、または PROC の各パネルに 1 行挿入します。

►—INSert————►

### 注:

- 画面移動可能域の上部に行を挿入するには、カーソルを最初の行の上に直接置き、挿入キーを押します。
- 計算行を FORM.CALC パネルに挿入するには、行を追加したい位置の上の行にカーソルを置き、挿入キーを押します。別の方法として、コマンド行に INSERT と入力してから、カーソルを挿入位置に合わせて Enter キー を押してください。

---

## INTERACT

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X	X	X

INTERACT コマンドは、プロセッサーまたはアプリケーションの実行中に、ユーザーとの対話を可能にします。対話は 2 つの形式で使用できます。

**セッション** 現行 QMF セッション内で対話式対話を開始します。

**コマンド** 対話式対話内で單一コマンドを実行します。

### セッション形式の INTERACT

(1)

►—INTeract————►

### 注:

- この形式は、QMF プロセッサーまたは呼び出し可能インターフェース・アプリケーションで有効です。

### コマンド形式の INTERACT

(1)

►—INTeract—qmfcmd————►

### 注:

- コマンド・インターフェース (DSQCCI) で使用します。呼び出し可能インターフェースから発行されたときには効果がありません。

### qmfcmd

実行する QMF コマンド。

## ISPF

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X		

ISPF は、対話式システム生産向上機能 (ISPF) を呼び出す、QMF 提供のコマンド同義語の 1 つです。

### QMF からの ISPF の呼び出し



#### option

ISPF/PDF に渡す初期オプション。たとえば、3 を入力すると、3 番目の ISPF パネル・オプションが直接選択されます。

オプションを指定しない場合は、ISPF/PDF 基本オプション・メニューが表示されます。

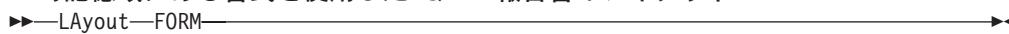
## LAYOUT

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X		

LAYOUT コマンドは、QMF 書式だけを入力に使用して QMF 報告書のサンプルを生成します。このコマンドは担当者報告書のビジュアル・レンダリングを提供することによって、QMF 書式の開発を助けます。

LAYOUT は、QMF が提供する ISPF アプリケーションのためのコマンド同義語の 1 つです。書式を分析し、QMF 書式に指定された基本報告書機能を実行するのに十分な汎用データを作成します。照会は不要です。

### 一時記憶域にある書式を使用した QMF 報告書のレイアウト



### データベースからの書式を使用した QMF 報告書のレイアウト



## 説明

### formname

データベースにある QMF 書式の名前。

## 注

- サンプル書式を使用して、データを表すいろいろな文字の入った報告書を表示することができます。報告書に切れ目がない場合、以下の文字が表示されます。

X 文字データ

## LAYOUT

### 0 数値データ

報告書に切れ目がある場合、次の文字で切れ目レベルが示されます。

- A** 切れ目 1 の文字データ
- 1** 切れ目 1 の数値データ
- B** 切れ目 2 の文字データ
- 2** 切れ目 2 の数値データ

書式の外観を見て変更が必要と思えば、照会を実行せずに、必要な変更を加えることができます。

- LAYOUT コマンドは QMF (2 進数) データ・フォーマットでデータを作成します。このフォーマットについては、「DB2 QMF アプリケーション開発の手引き」で説明しています。
- LAYOUT コマンドは長いオブジェクト名をサポートしています。ISPF は、スクロール可能なフィールドをサポートしており、QMF はこれを使用して、従来の短い名前のフィールド内に、長い名前を指定することができます。ISPF コマンドの、RIGHT、LEFT、および EXPAND を使用すると、スクロール可能なフィールド内にデータを入力したり、表示したりすることができます。
- LAYOUT コマンドは、QMF コマンド・インターフェースを使用して、ISPF アプリケーションとしてインプリメントされています。この「コマンド・プロンプト」パネルは、ISPF サービスを使用して定義されており、ISPF パネルとして ISPF に割り振られています。

## 例

1. プロンプト・パネルを表示するには次のようにします。

```
LAYOUT ?
```

2. データベースにある既存の書式 (MYFORM) を使用してサンプル報告書を作成する場合:

```
LAYOUT MYFORM
```

または

```
LAYOUT FORM MYFORM
```

3. 一時記憶域にある書式を使って LAYOUT コマンドを実行するには次のようにします。

```
LAYOUT FORM
```

4. QMF プロシージャーから LAYOUT コマンドを入力するには、区切り ID (二重引用符) を使用して、QMF 線形手順で複数の行にわたるフォーム・オブジェクト名を継続する必要があります。77 ページの図 9 に示すように、すべての継続行の 1 桁目は、正符号 (+) でなければなりません。

```

PROC                               MODIFIED LINE
LAYOUT TABLE
+"AUTHXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX"
+"XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX".
+"OBJXXXXXXXXXXXXXXXXXXXXXX"
+"XXXXXXXXXXXXXXXXXXXXXX"

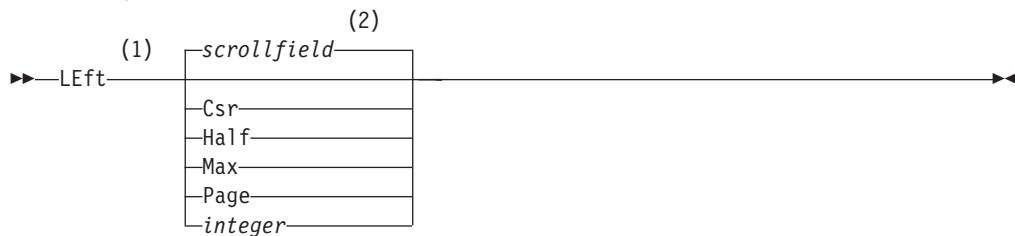
```

図9. QMF プロシージャーからの LAYOUT コマンドの入力

**LEFT**

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X	X	X

LEFT コマンドは、報告書パネルまたは QBE 照会の左方の境界に向かってスクロールします。

**注:**

- 活動中のパネルに画面移動フィールドがある場合にのみ画面移動量を指定します。その他のすべての場合は、PAGE が想定されます。
- 画面移動フィールドに表示される値が使用されます。この値は、大域変数 DSQDC\_SCROLL\_AMT にも保持されます。

**説明**

<b>CSR</b>	左方へスクロールして、カーソルがある桁をパネルの右端に位置変更します。カーソルがパネルの左端にあるとき、LEFT CSR と LEFT PAGE は同じ効果があります。
<b>HALF</b>	パネルの半分の幅だけ左方へスクロールするか、(その前に左方の境界に到達する場合は) 左方の境界までスクロールします。
<b>MAX</b>	左方の境界までスクロールします。
<b>PAGE</b>	パネルの幅だけ左方へスクロールするか、(その前に左方の境界に到達する場合は) 左方の境界までスクロールします。
<b>integer</b>	この桁数分左側に画面移動します (総数は 1 から 9999 の範囲)。

**注**

- MAX は現行のコマンドの間だけ有効です。この値はコマンドが完了すると SCROLL フィールドには残りません。大域変数 DSQDC\_SCROLL\_AMT をこの値に設定することはできません。

## LEFT

- 報告書で左方にスクロールするときは、LEFT ファンクション・キーを使用してください。スクロール量を指定するには、スクロールしたい桁数をコマンド行にタイプしてから、LEFT ファンクション・キーを押します。

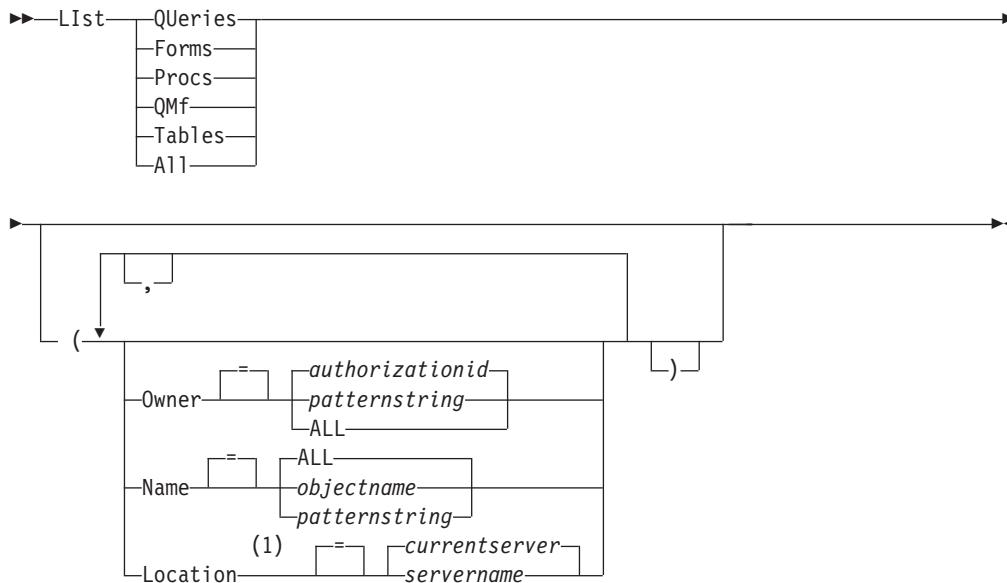
## LIST

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X	X	X

データベースに保管されている QMF オブジェクトおよびデータベースのリストを表示する場合、LIST コマンドを使用してください。QMF セッションで LIST コマンドを最初に発行する場合には、必ず、Queries、Forms、Procs、QMF、Tables、または All のパラメーターのうちいずれか 1 つを使用するようにしてください。

LIST コマンドをパラメーターを付けずに出すと、QMF は要求された最新リストを表示します。

### データベースからのオブジェクトのリストの作成



### 注:

- 使用は表に限定されます。

### オブジェクトの現行リストの表示

```
►► LIst
```

## 説明

- |                |                                   |
|----------------|-----------------------------------|
| <b>ALL</b>     | すべての対象 (QMF 対象およびデータベース表) を表示します。 |
| <b>TABLES</b>  | データベース表の対象 (表、視点、別名) のみをリストします。   |
| <b>QMF</b>     | QMF 対象 (照会、書式、手順) のみをリストします。      |
| <b>QUERIES</b> | QMF 照会のみをリストします。                  |

<b>FORMS</b>	QMF 書式のみをリストします。
<b>PROCS</b>	QMF 手順のみをリストします。
<b>OWNER</b>	リストする対象の所有権修飾子を指定します。ユーザーのデータベース許可 ID がデフォルトになります。
<b>authorizationid</b>	ユーザー、スキーマ、またはデータベース・コレクションの名前です。
<b>patternstring</b>	特定のパターンの所有者名を検索します。パターンは、特別な意味を持つ下線および % 記号文字の文字で指定されます。
<b>ALL</b>	所有者に関係なく、すべての対象をリストします。
<b>NAME</b>	リストする対象の名前を指定します。
<b>ALL</b>	名前に関係なく、すべての対象をリストします。
<b>objectname</b>	QMF 対象またはデータベース表の名前です。
<b>patternstring</b>	特定のパターンの対象名を検索します。パターンは、特別な意味を持つ下線および % 記号文字の文字で指定されます。
<b>LOCATION</b>	リストする対象のロケーションを指定します。現行データベース・サーバーが省略時値です。
<b>servername</b>	分散ネットワーク内のデータベース・アプリケーション・サーバーの名前です。
	このオプションは、現行のロケーションが DB2 for z/OS サーバーである場合に限り使用することができます。大域変数 DSQAO_DB_MANAGER が 2 という値をもつ場合、QMF セッションは DB2 (z/OS 版) サーバーに接続されます。

## 注

- 所有していない QMF 対象は、SHARE=YES オプションで保管されている場合のみリストされます。
- OWNER および NAME パラメーターと併用されるパターン・ストリングは、次に示すようにして指定することができます。
  - % 記号は、ゼロあるいは 1 つ以上の文字のストリングを表します。
  - \_ 記号は任意の 1 文字を表します。

たとえば、第 2 文字位置に文字 D がある所有者をもっているすべての QMF オブジェクトをリストするには、次のように入力します。

```
LIST QMF (OWNER=_D%
```
- オブジェクトのリストを要求すると、QMF は、オブジェクトを、デフォルトの順序、すなわち所有者が先で、次に名前という順序で表示します。省略時値のリストの順序を変更するには、DSQDC\_LIST\_ORDER 大域変数を変更します。

## LIST

DSQDC\_LIST\_ORDER 大域変数は 2 文字の値です。1 文字目は分類文字を指定し、2 文字目は分類が昇順か降順かを指定します。DSQDC\_LIST\_ORDER の値の変更は、現行セッションにのみ適用されます。省略時値は 1A です。

表 2 に、1 文字目の値を示します。

表 2. LIST コマンドのソート・シーケンス

値	特性(主キー)	ソート・シーケンス
1	デフォルト	所有者、名前
2	所有者	所有者、名前
3	名前	名前、所有者
4	タイプ	タイプ、名前、所有者
5	変更されたもの	変更されたもの、最後に使用されたもの、所有者、名前、タイプ
6	最後に使用されたもの	最後に使用されたもの、変更されたもの、所有者、名前、タイプ

2 文字目は、次の指定をします。

- A 昇順
- D 降順

たとえば、リストの先頭に 1 番最近変更した対象の新しいリストを作成するには、以下の SET GLOBAL コマンドを入力します。

```
SET GLOBAL (DSQDC_LIST_ORDER=5D)
```

リストの先頭に現行の所有者の対象の新しいリストを作成するには、以下の SET GLOBAL コマンドを入力します。

```
SET GLOBAL (DSQDC_LIST_ORDER=1A)
```

これらの例では、既存のリストの順序は変更されません。

- いま表示されているオブジェクト・リストの作成後に新しいロケーションへの接続を行なっていれば、そのリストはすでに最新状態を反映していません。リストをリフレッシュするか、または取り消して新しいリストを作成してください。旧リストのアクション欄にコマンドを入れても、それは実行されません。
- LOCATION パラメーターを使用して遠隔ロケーションで照会、手順または書式をリストすることはできません。リモート・ロケーションでこれらのオブジェクトをリストするためには、まずそのロケーションに接続してから、LIST コマンドを使用してください。
- 表のリストを要求すると、QMF は視点を使用して情報を検索します。
  - 現行ロケーションが DB2 (z/OS 版) で、そこからリストを要求した場合、(LOCATION が指定されていなかったり、現行ロケーションが指定されていたりした場合)、QMF は大域変数 DSQEC\_ALIASES および DSQEC\_TABS\_LDB2 で名前を付けられた視点を使用します。
  - 現行ロケーションが DB2 (z/OS 版) で、別の DB2 (z/OS 版) ロケーションからリストを要求した場合、QMF は大域変 DSQEC\_ALIASES および DSQEC\_TABS\_RDB2 で名前を付けられた視点を使用します。

- 現行ロケーションが DB2 Server (VM 版) または DB2 Server (VSE 版) の場合、QMF は、大域変数 DSQEC\_TABS\_SQL で指定された視点を使用します。
  - QMF がバッチ・モードで稼働しているときに LIST コマンドを出すと、エラーになります。また、QMF コマンド・インターフェースを通じて LIST を出すときは、その LIST の前に INTERACT を出しておかなければなりません（ただし、指示照会ダイアログ・パネルから出す場合を除きます）。
  - LIST コマンドは長い所有者名と表名をサポートしています。プロンプト・パネルでの長い名前の入力方法の詳細は、3 ページの『プロンプト・パネルからの入力』を参照してください。
  - LIST コマンドが有効なところであれば、LIST コマンドで单一引用符を使用できます。LIST コマンドで、二重引用符は使用できません。

例

1. 所有するすべての書式のリストを表示するには、以下のように入力します

## LIST FORMS

2. 所有者に関係なく、名前が APP1 で始まるすべての照会を含むリストを表示するには、以下のように入力します。

LIST QUERIES (OWNER=ALL NAME=APP1%

3. 所有者に関係なく、名前に文字 CUST が付く、ダラス・データベース・サーバー内の表のすべてを含むリストを表示するには、以下のように入力します。

```
LIST TABLES (LOCATION=DAILASDB OWNER=ALL NAME=%CUST%
```

4. QMF プロシージャーで LIST コマンドを使用する場合、QMF 線形手順内で複数行にわたる許可 ID を継続するには、単一引用符を使用する必要があります。図 10 に示すように、すべての継続行の 1 桁目は、正符号 (+) でなければなりません。

図 10. OMF 線形手順での複数行にわたる許可 ID の継続

## MESSAGE

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X	*	*

QMF メッセージ域にメッセージを渡すには、ユーザー・アプリケーション (プロシージャー、プログラム、EXEC、および CLIST) から MESSAGE コマンドを使用します。

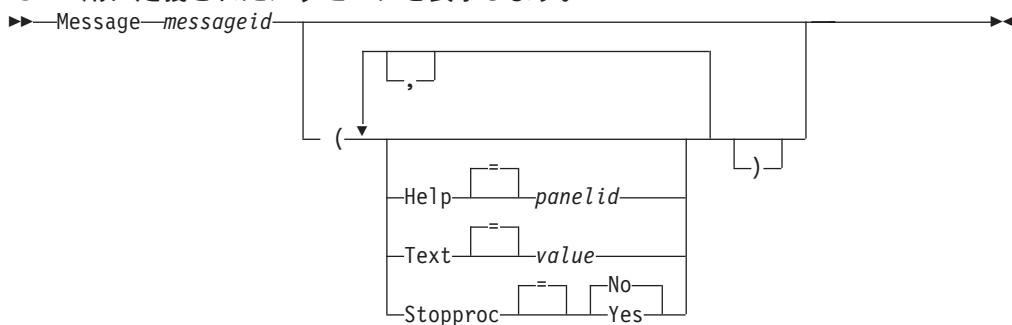
MESSAGE コマンドでは、次に挙げることを行うことができます。

- JSPE ライブラリーからメッセージを表示

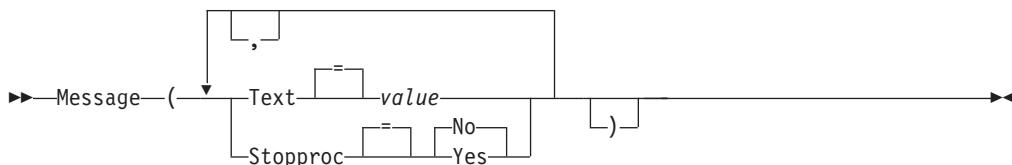
## MESSAGE

- ISPF メッセージのヘルプ・パネルを割り当てる
- QMF 類似のメッセージを生成
- QMF 線形手順の実行を抑止

ISPF 用に定義されたメッセージを表示します。



QMF 類似のメッセージの作成



## 説明

### messageid

ISPF メッセージ・ライブラリーのメッセージ定義の識別番号です。指定されたライブラリーは ISPMILIB データ・セットに連結されている必要があります。

**HELP** メッセージに付随するヘルプ・パネルを指定します。このオプションは、ISPF メッセージ定義で指定されているチュートリアル用ヘルプ・パネルを上書きします。

### panelid

ISPF パネル・ライブラリーのパネルの名前です。指定されたライブラリーは ISPPLIB データ・セットに連結されている必要があります。

**TEXT** メッセージ・テキストを定義します。このオプションを使用して、360 までの 1 バイト文字メッセージ・テキストを発行することができます。

ISPF メッセージ定義と共に使用した場合、このオプションは ISPF メッセージ定義で指定されたメッセージをオーバーライドします。

**value** メッセージの内容を構成する文字ストリングです。

プランク文字を含む値は、区切り文字で囲む必要があります。メッセージの値で有効な区切り文字は、単一引用符、括弧、および二重引用符です。

**STOPPROC**

QMF 線形手順の終了切り替えを設定します。現行適用業務が終了するまで、あるいは適用業務によって設定が再度変更されるまで、設定は活動状態のままです。

**YES** 手順終了切り替えをオンに設定します。制御を受け取る任意の QMF 線形手順は、即座に実行を終了します。

**NO** 手順終了切り替えをオフに設定します。 QMF 線形手順の実行を抑止します。

**注**

- MESSAGE コマンドは QMF コマンド行からは入力できません。これは、QMF コマンドまたは呼び出し可能インターフェースを使用して、QMF プロシージャーまたはアプリケーションからのみ発行できます。
- STOPPROC オプションは、線形手順内でのみ使用が制限されています。プロシージャー終了スイッチがオンに設定されると、プロシージャーは即座に終了します。

MESSAGE コマンドの完全な説明については、「DB2 QMF アプリケーション開発の手引き」を参照してください。

**例**

- ユーザーの (CMDHELP という名前の) ヘルプ・パネルに、ISPF メッセージ ISPG053 を表示するには、次のようにします。  

```
MESSAGE ISPG053 (HELP=CMDHELP)
```
- QMF 類似のメッセージを発行するには、次のようにします。  

```
MESSAGE (TEXT=(Sales report for YE '05 is complete.))
```
- 図 11 に、ロジックを持つ QMF プロシージャーで、複数行にわたる MESSAGE コマンドの指定方法を示しています。

```
/* QMF REXX PROCEDURE */
MSGTEXT="MESSAGE _TEXT_AAAAAAAAAAAAAAAAAAAAAAAA"
"AAAAAAAAA"
"AAAAAAA"
"AAAAAA"
"AAAAAA"
"AAAAAA"
"AAAAAA"
"AAAAAA"
"AAAAAA"
"AAAAAA"
"AAAAAA"
"MESSAGE (TEXT=(\"MSGTEXT\"))"
EXIT
```

図 11. ロジックを持つ QMF プロシージャーからの MESSAGE コマンドの発行

- 84 ページの図 12 に、線形手順で、複数行にわたる MESSAGE コマンドの指定方法を示しています。

## NEXT

```
MESSAGE (TEXT='MESSAGE_TEXTXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX  
+XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX  
+XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX  
+XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX  
+XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX')
```

図 12. 線形手順からの QMF MESSAGE コマンドの実行

## NEXT

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X	X	X

NEXT コマンドは、

- FORM.DETAIL パネルと関連づけられている 1 組のバリエーションを前方へナビゲートさせます。
- 列定義または列位置合せパネルから次の列または次の定義を表示します。
- 表エディターで、アクセスされる行のセットの次の行を表示します。



## 説明

### COLUMN

列定義パネルまたは列調整パネルで、次の列を表示します。

### DEFINITION

列定義パネルから、次の非ブランクの定義式を持つ列を表示します。

## 注

- 定義は REXX 機能を必要としますが、CICS ではサポートされていません。
- COLUMN と DEFINITION パラメーターは、
  - FORM.COLUMNS パネルまたは FORM.DEFINITION パネルがアクティブのとき、パネル間ナビゲーションを制御します。
  - コマンド行または適用業務から正常に入力されなくても使用できます。
- FORM.DETAIL パネルでの NEXT コマンドは、
  - 次のパネル・バリエーションを表示します (結果がエラーではなかった場合)。
  - コマンド行からファンクション・キーを押して入力するか、またはアプリケーションから入力できます。
- 表エディターでは、NEXT コマンドはファンクション・キーでしか入力できません。

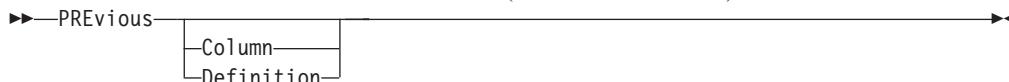
---

## PREVIOUS

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X	X	X

PREVIOUS コマンドは、

- FORM.DETAIL パネルと関連づけられている 1 組のバリエーションを後方にナビゲートします。
- 書式定義が表示されているとき、直前の列または定義を表示します。
- 表エディター・セッションでは、追加されたばかりの行 (追加モードの場合)、または正しく実行された最新の検索基準 (検索モードの場合) を表示します。



## 説明

### COLUMN

列定義パネルまたは列調整パネルで、直前の列が表示されます。

### DEFINITION

定義パネルの場合、非ブランク定義式とともに最新の列が表示されます。

## 注

- 定義は REXX 機能を必要としますが、CICS ではサポートされていません。
- Column と Definition パラメーターは、FORM.COLUMNS パネルまたは FORM.DEFINITION パネルがアクティブのとき、パネル間ナビゲーションを制御します。
- FORM.DETAIL パネルでの PREVIOUS コマンドは、
  - 前のパネル・バリエーションを表示します (結果がエラーではなかった場合)。
  - コマンド行からファンクション・キーを押して入力するか、またはアプリケーションから入力できます。
- 表エディターでは、PREVIOUS コマンドはファンクション・キーでしか入力できません。

---

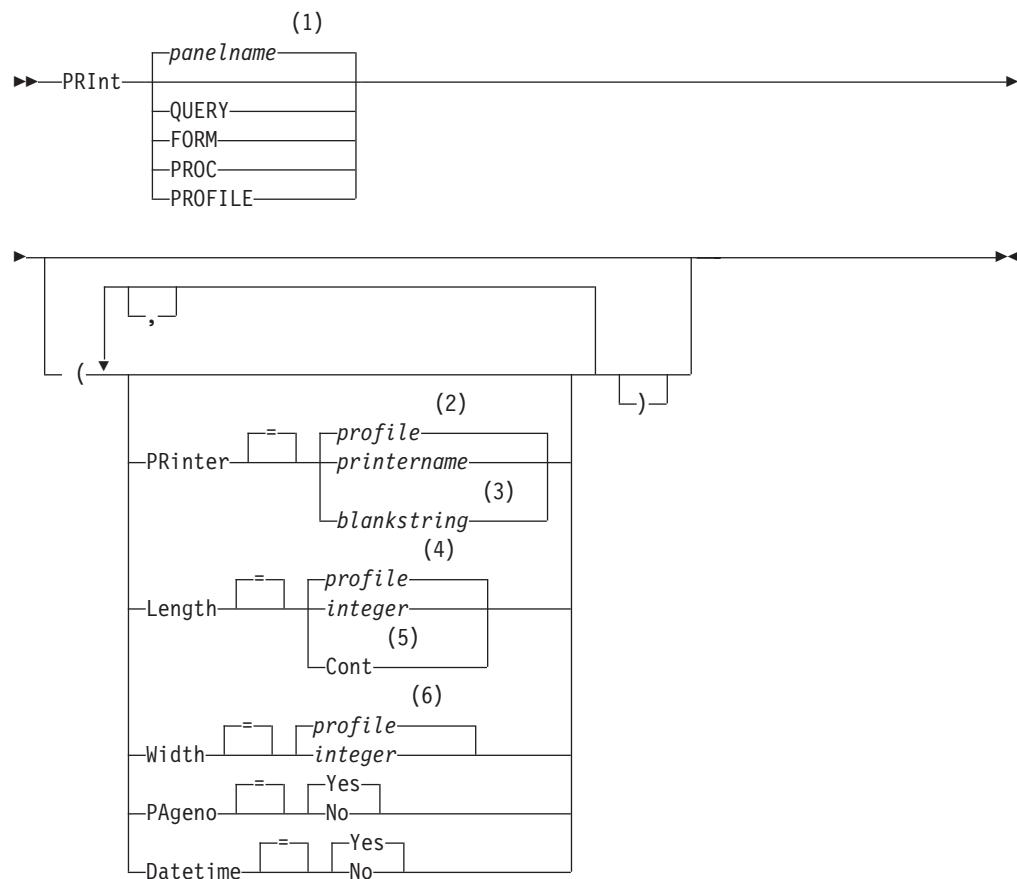
## PRINT (TSO の場合)

TSO (ISPF 付き)	TSO (ISPF なし)
X	X

PRINT コマンドは、QMF 一時記憶域にあるオブジェクトや、データベースに保管されているオブジェクトを印刷します。

一時記憶域からの QMF オブジェクトの印刷

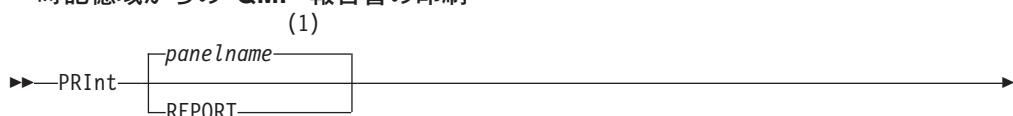
## PRINT (TSO の場合)

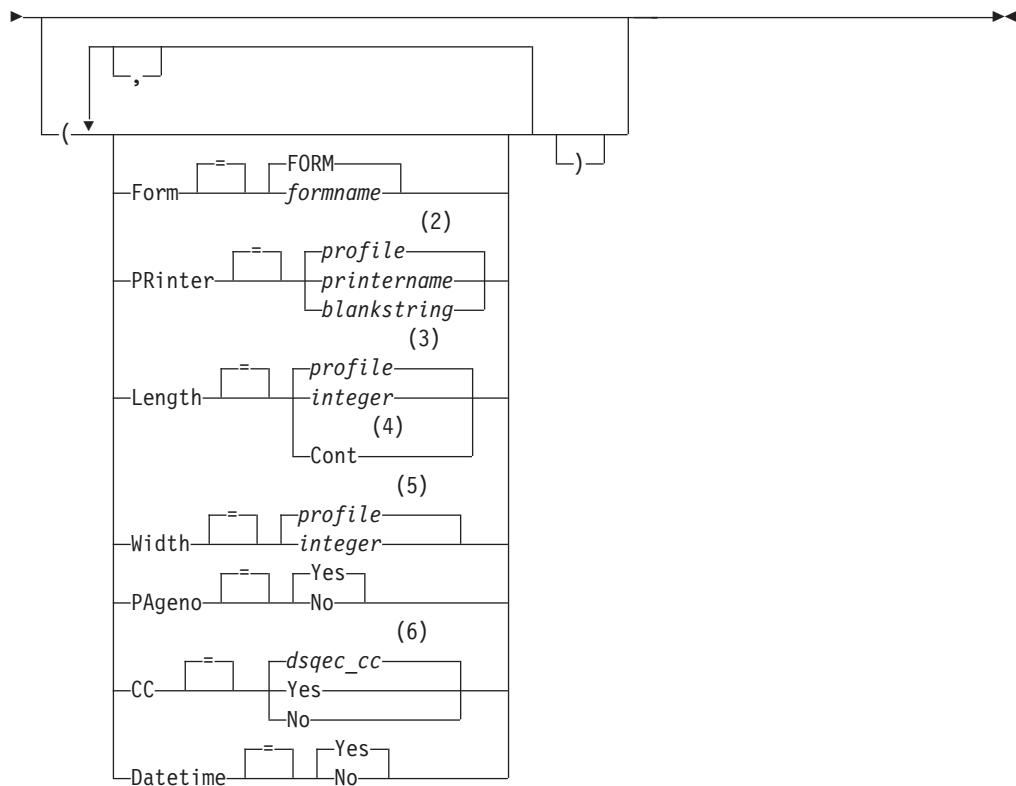


### 注:

- 1 該当する場合は、現在表示されている QMF 対象パネルの名前が使用されます。
- 2 プロファイルで設定された値が使用されます。
- 3 このオプションの使用は限定されます。以下の説明を参照してください。
- 4 プロファイルで設定された値が使用されます。
- 5 このオプションの使用は限定されます。以下の説明を参照してください。
- 6 プロファイルで設定された値が使用されます。

## 一時記憶域からの QMF 報告書の印刷

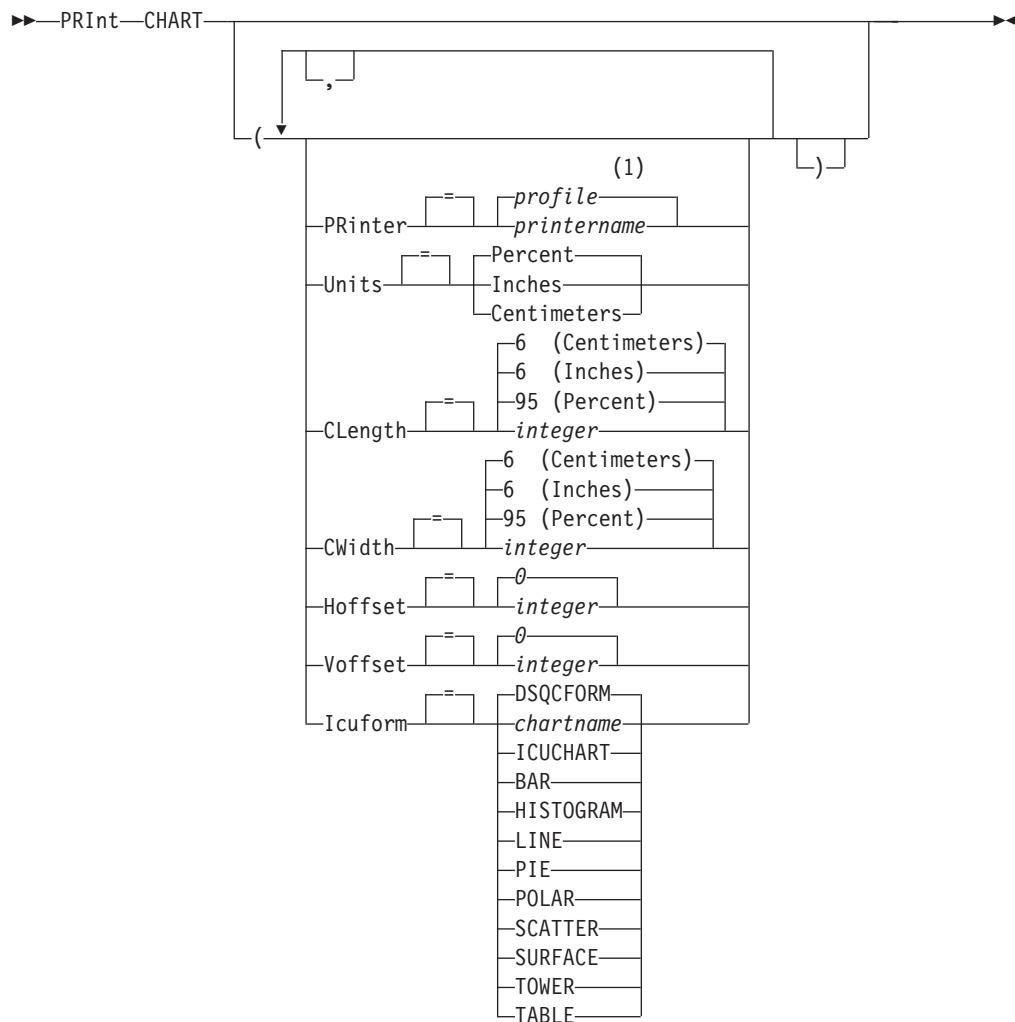


**注:**

- 1 該当する場合は、現在表示されている QMF 対象パネルの名前が使用されます。
- 2 プロファイルで設定された値が使用されます。
- 3 プロファイルで設定された値が使用されます。
- 4 このオプションの使用は限定されます。以下の説明を参照してください。
- 5 プロファイルで設定された値が使用されます。
- 6 *dsqec\_cc* を 1 に設定して (*cc* が有効な場合に)、報告書の 1 桁目に紙送り制御文字を出力します。この大域変数を 0 に設定すると、紙送り制御文字は出力されません。

**図表の印刷**

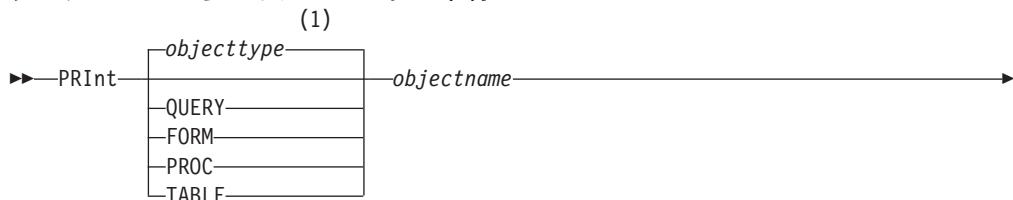
## PRINT (TSO の場合)

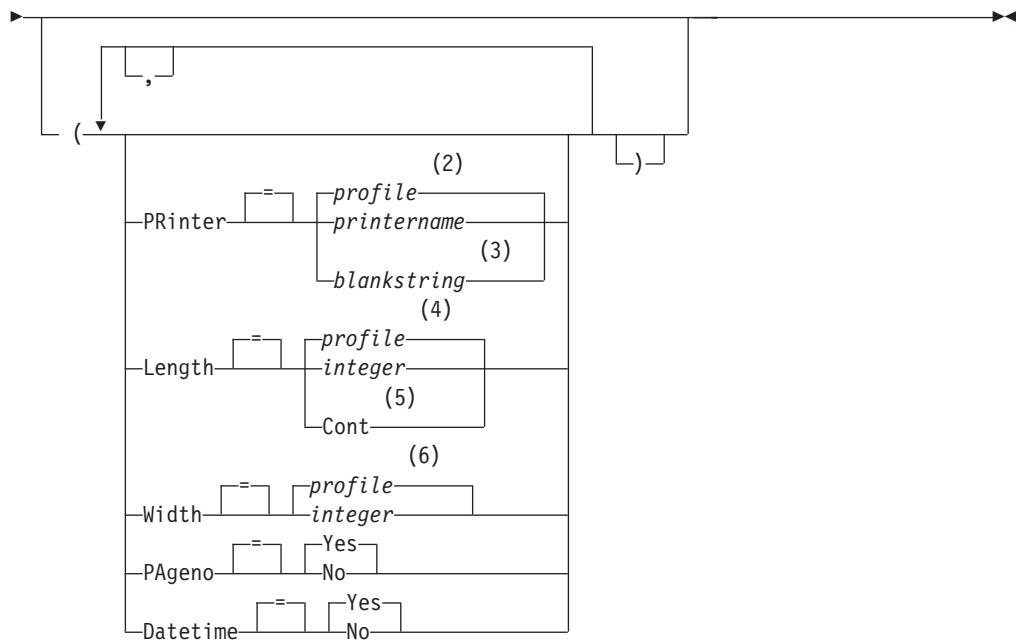


注:

- 1 プロファイルで設定された値が使用されます。

### データベースからのオブジェクトの印刷



**注:**

- 1 該当する場合は、名付けられた対象のタイプが使用されます。 QMF オブジェクトは他のタイプのオブジェクト (たとえば、データベース・オブジェクト) より優先されます。
- 2 プロファイルで設定された値が使用されます。
- 3 このオプションの使用は限定されます。以下の説明を参照してください。
- 4 プロファイルで設定された値が使用されます。
- 5 このオプションの使用は限定されます。以下の説明を参照してください。
- 6 プロファイルで設定された値が使用されます。

**説明****objectname**

データベース内の対象の名前です。有効なオブジェクトは下記のとおりです。

- QMF オブジェクト (PROC、QUERY、FORM)
- 表オブジェクト (TABLE、VIEW、SYNONYM、ALIAS)

**PRINTER**

PRINT コマンドの出力宛先を指定します。

**printername**

プリント出力先を指定します。GDDM プリンターのニックネームである必要があります。

**blankstring**

ファイルの宛先を指定します。この値は、单一引用符 (' ') で囲まれた 0 から 8 文字のブランクで指定されなければなりません。

## PRINT (TSO の場合)

印刷出力の物理的な宛先は、QMF DSQPRINT に割り振られたデータ・セットまたは装置です。 QMF 環境に関する詳細については、QMF 管理者にお問い合わせください。

このオプションは、図表、書式、または指示照会オブジェクトの場合無効です。

### LENGTH

印刷するページの長さを指定します。長さの単位は行です。

#### integer

ページの切れ目までの最大行数を指定します。数値は 1 から 999 までの整数です。

表 3 に、オブジェクトに適用される最小の長さを示します。

表 3. オブジェクトと、印刷時の最小の長さ

オブジェクト	最小の長さ
書式	25
SQL 照会	25
プロシージャー	25
指示照会	25
表	8
QBE 照会	7 (データ・セットに印刷する場合は 5)
プロファイル	7 (データ・セットに印刷する場合は 5)

報告書の場合の最小値は、使用される書式、あるいはコマンド・オプション DATETIME と PAGENO の値によって変わります。

印刷される長さの最大値は 66 行です。

### CONT

ページの切れ目がない連続印刷を指定します。

このオプションは、図表、書式、または指示照会オブジェクトの場合、あるいはプリンターナンバーが指定された場合には無効です。

### WIDTH

印刷するページの幅を指定します。幅の単位は 1 バイト文字です。

#### integer

1 行に印刷する最大文字数を指定します。数値は 22 から 999 までの整数です。

印刷する対象が報告書でない限り、指定した値より長い行は右端で切り捨てられます。この場合、FORM.OPTIONS パネルで行の折り返しを指定した場合を除き、指定した値より長い行は後続のページで書式化されます。

### PAGENO

印刷するオブジェクトにページ番号を入れるかどうかを指定します。

報告書を印刷する場合で、書式に変数 &PAGE が含まれていると、このオプションは無視されます。

#### YES

ページ番号がページの下部に組み込まれます。

**NO** ページ番号は組み込まれません。

#### DATETIME

印刷するオブジェクトの各ページにシステム日付と時刻を入れるかどうかを指定します。

報告書を印刷する場合で、書式に変数 &DATE または &TIME が含まれていると、このオプションは無視されます。

**YES** 日時がページの下部に組み込まれます。

**NO** 日時は組み込まれません。

#### FORM

報告書を印刷する場合に使用する書式を指定します。

**FORM** 一時記憶域にある現在の FORM オブジェクト。これはデフォルトです。

#### formname

データベースにある QMF 書式の名前。一時記憶域にある現行の書式は、この書式で置き換えられます。

#### UNITS

図表寸法パラメーターの CLENGTH、CWIDTH、HOFFSET、および VOFFSET の計測単位を指定します。

#### PERCENT

図表寸法は画面サイズに対する相対的な値 (パーセント) です。

#### CENTIMETERS

図表寸法をセンチメートルで表現します。

#### INCHES

図表寸法をインチで表現します。

#### CLENGTH

図表域の長さを数値で表現します。計測単位は UNITS パラメーターで決定されます。デフォルトは計測単位によって異なります。

#### CWIDTH

図表域の幅を数値で表現します。 計測単位は UNITS パラメーターで決定されます。デフォルトは計測単位によって異なります。

#### HOFFSET

ページの左端から図表域までの水平オフセットを数値で指定します。計測単位は UNITS パラメーターで決定されます。

#### VOFFSET

ページの上端から図表域までの垂直オフセットを数値で指定します。計測単位は UNITS パラメーターで決定されます。

#### ICUFORM

図表形式の名前を指定します。図表形式は、データを図表に変換するために必要な仕様を含んでいます。様々な形式が、様々なタイプの図表を作成するために使用されます。

#### DSQCFORM

QMF が提供するデフォルトの図表フォーマットの名前。

このフォーマットは、QMF 管理者がカスタマイズすることができます。カスタマイズされない場合は、棒グラフが使用されます。

## PRINT (TSO の場合)

### **chartname**

図表フォーマットの名前。

### **ICUCHART**

GDDM 対話式図表ユーティリティーのデフォルトの図表フォーマットを指定します。

### **BAR**

### **HISTOGRAM**

### **LINE**

### **PIE**

### **POLAR**

### **SCATTER**

### **SURFACE**

### **TOWER**

### **TABLE**

QMF が提供する図表フォーマットの名前。

## 注

- 書式の印刷では、書式を構成するすべての部分が印刷されます。
- 報告書を印刷する際に、報告書は書式指定にしたがって印刷されます。
- 表を印刷する場合、表はデフォルト書式でフォーマット設定されます。

デフォルト書式以外の書式でフォーマット設定された表を印刷するには、表を表示し、次に目的の書式を表示してから、PRINT REPORT コマンドを発行します。(例 2 (93 ページ) を参照してください。)

しかし、その書式で、データの各行がソートされた順序になっている必要がある場合には(たとえば、制御の切れ目を使用する書式では)、表を表示するのではなくて、その表からソート済みの順序で表中のデータを選択するような照会を最初に実行する必要があります。

- 図表を印刷する際に、書式指定がデータに適用され、図表は GDDM 対話式図表ユーティリティーによって書式化されます。
- 報告書または図表を印刷するときに書式にエラーがあると、最初のエラーが見つかった書式パネルが表示され、エラーが強調表示されます。他のエラーを表示するには、表示された最初のエラーを訂正する必要があります。

報告書を作成するまで検出されないエラーもあります。

- DBCS 印刷装置を使用すると、DBCS データを表示する端末を持っていなくても、DBCS データを含む報告書を印刷することができます。プログラム・パラメーター DSQSDBCS を YES に設定して、QMF を開始します。QMF の開始プロセッジャーのカスタマイズの詳細については、QMF 管理者にお問い合わせください。
- DBCS データを使用中に QMF がページを分割する場合、ページの左側から 4 バイト目のところで、報告書の 2 番目およびそれに続くページの印刷が再開されます。

- FORM.PAGE パネルで &PAGE、&DATE、および &TIME を指定することによって、ページ番号、日付、時刻をそれぞれ図表の表題に組み込むことができます。
- 印刷される報告書と画面に表示される報告書とは、表 4 に示すように、異なります。

表 4. 表示される報告書と印刷される報告書との違い

報告書部分	表示された報告書	印刷された報告書
ページ数	スクロール可能な 1 ページ	1 ページ以上
ページ見出しと脚注	1 回だけ表示	各ページの先頭と末尾
明細ヘッダー	報告書の始めの最初の明細行の前と続く全画面	報告書の始めの最初の明細行の前と続く全ページ
固定列	報告書を横にスクロールしても動かない	各ページの左端に印刷

- データベース管理者が複数行フェッチ機能を使用するように QMF をセットアップしている場合に、3 部構成の名前を使用して表を印刷するときは、操作しているデータベースは（ローカルとリモートの）両方とも DB2 (z/OS 版) 8.1.5 以降でなければなりません。そうでなければ、コマンドは失敗します。データベース管理者は、複数行フェッチをオフにできます。この機能の設定については、「TSO/CICS 用 DB2 QMF インストールおよび管理の手引き」に説明されています。
- PRINT コマンドは、長いオブジェクト名の入力をサポートします。プロンプト・パネルでの長い名前の入力方法の詳細は、3 ページの『プロンプト・パネルからの入力』を参照してください。

## 例

- QMF PRINT コマンドのためのプロンプト・パネルを表示するには次のようにします。  
PRINT ?
- デフォルト書式以外の書式でフォーマット設定された表を印刷するには、次のようにします。

```
DISPLAY tablename
DISPLAY formname
PRINT REPORT
```

---

## PRINT (CICS の場合)

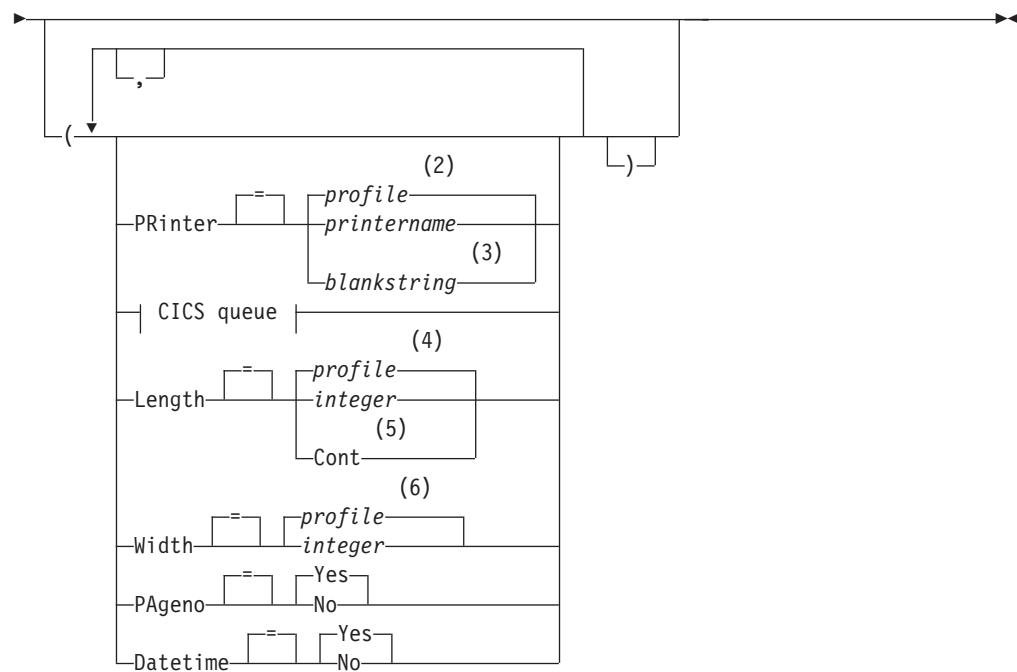
CICS の場合、PRINT コマンドは、QMF 一時記憶域にあるオブジェクト、またはデータベースに保管されているオブジェクトのコピーを印刷します。

### 一時記憶域からの QMF オブジェクトの印刷

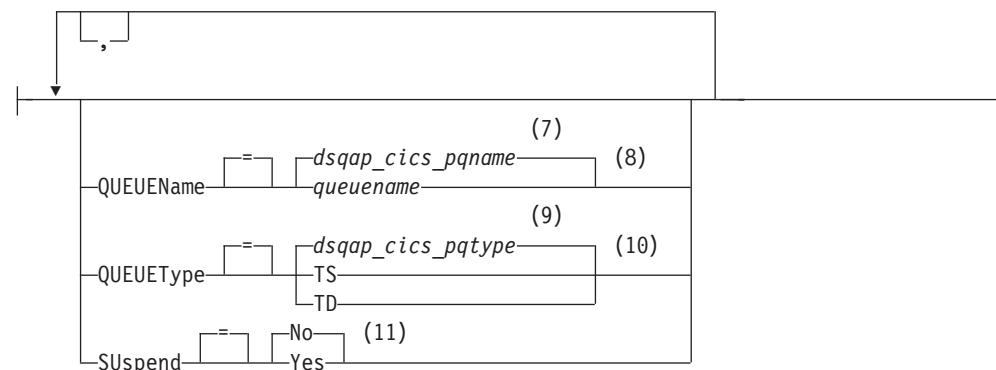
(1)



## PRINT (CICS の場合)



### CICS queue:

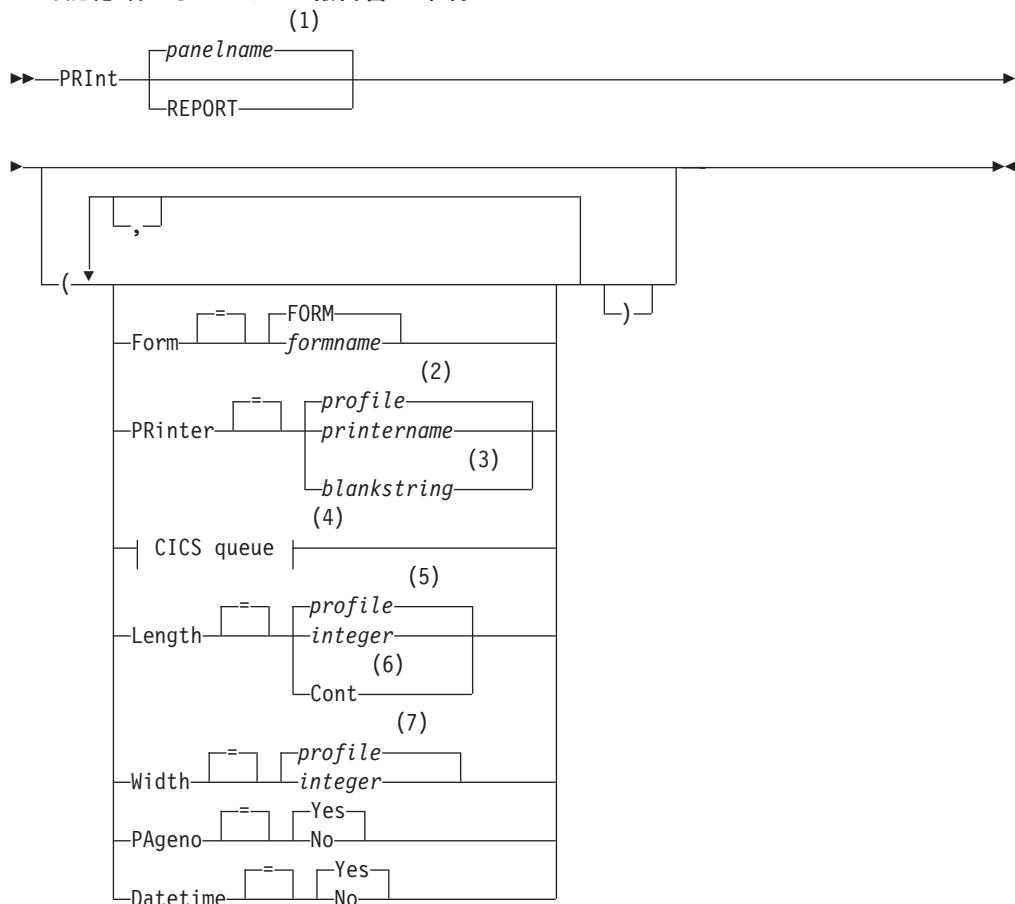


### 注:

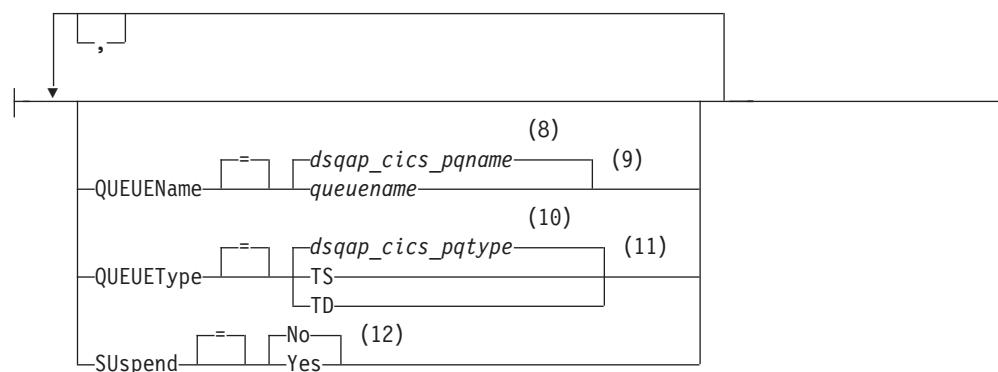
- 1 該当する場合は、現在表示されている QMF 対象パネルの名前が使用されます。
- 2 プロファイルで設定された値が使用されます。
- 3 このオプションの使用は限定されます。以下の説明を参照してください。
- 4 プロファイルで設定された値が使用されます。
- 5 このオプションの使用は限定されます。以下の説明を参照してください。
- 6 プロファイルで設定された値が使用されます。
- 7 この大域変数で設定された値が使用されます。
- 8 この大域変数で設定された値が使用されます。
- 9 この大域変数で設定された値が使用されます。

- 10 この大域変数で設定された値が使用されます。  
 11 このオプションの使用は限定されます。以下の説明を参照してください。

### 一時記憶域からの QMF 報告書の印刷



### CICS queue:



### 注:

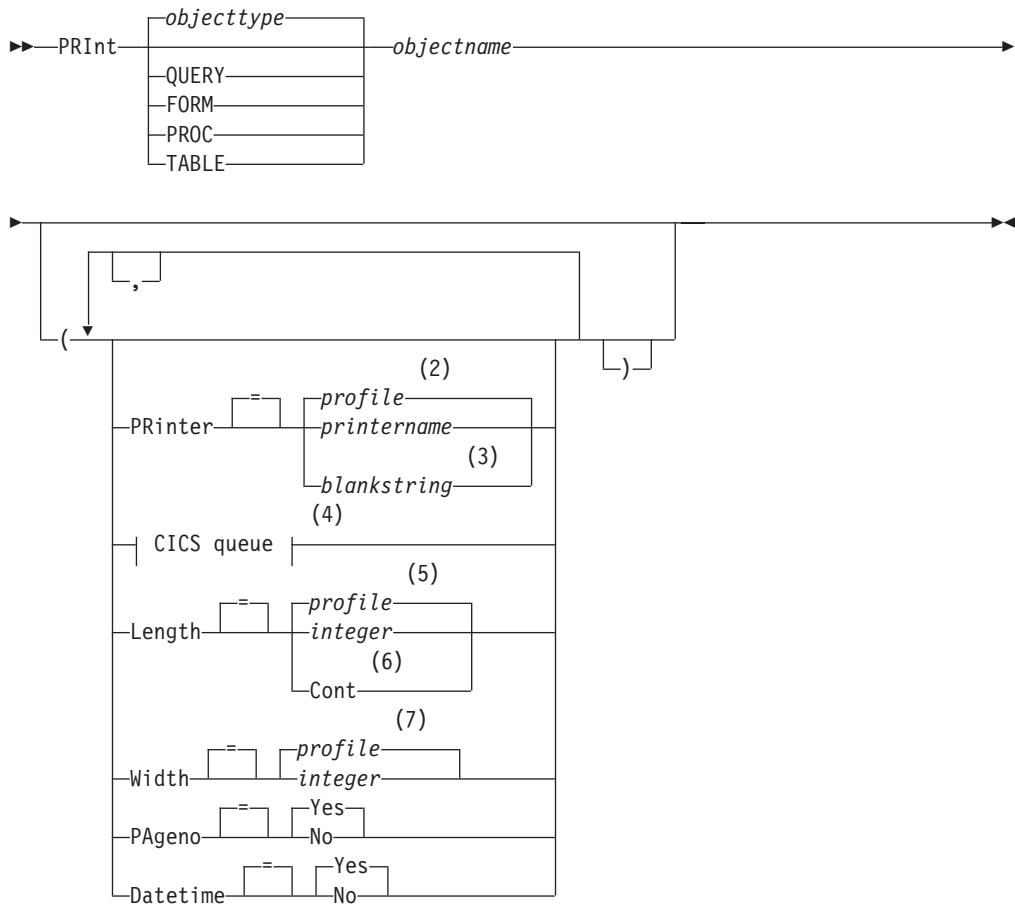
- 該当する場合は、現在表示されている QMF 対象パネルの名前が使用されます。
- プロファイルで設定された値が使用されます。

## PRINT (CICS の場合)

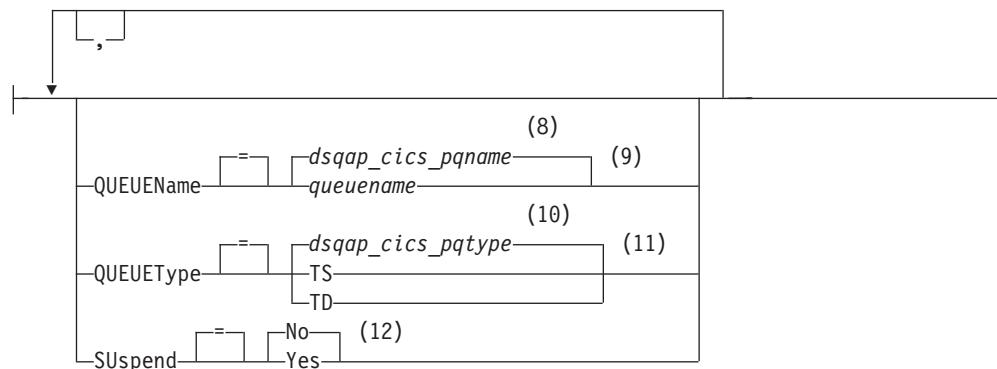
- 3 このオプションの使用は限定されます。以下の説明を参照してください。
- 4 このオプションの使用は限定されます。以下の説明を参照してください。
- 5 プロファイルで設定された値が使用されます。
- 6 このオプションの使用は限定されます。以下の説明を参照してください。
- 7 プロファイルで設定された値が使用されます。
- 8 この大域変数で設定された値が使用されます。
- 9 このオプションの使用は限定されます。以下の説明を参照してください。
- 10 この大域変数で設定された値が使用されます。
- 11 このオプションの使用は限定されます。以下の説明を参照してください。
- 12 このオプションの使用は限定されます。以下の説明を参照してください。

### データベースからのオブジェクトの印刷

(1)



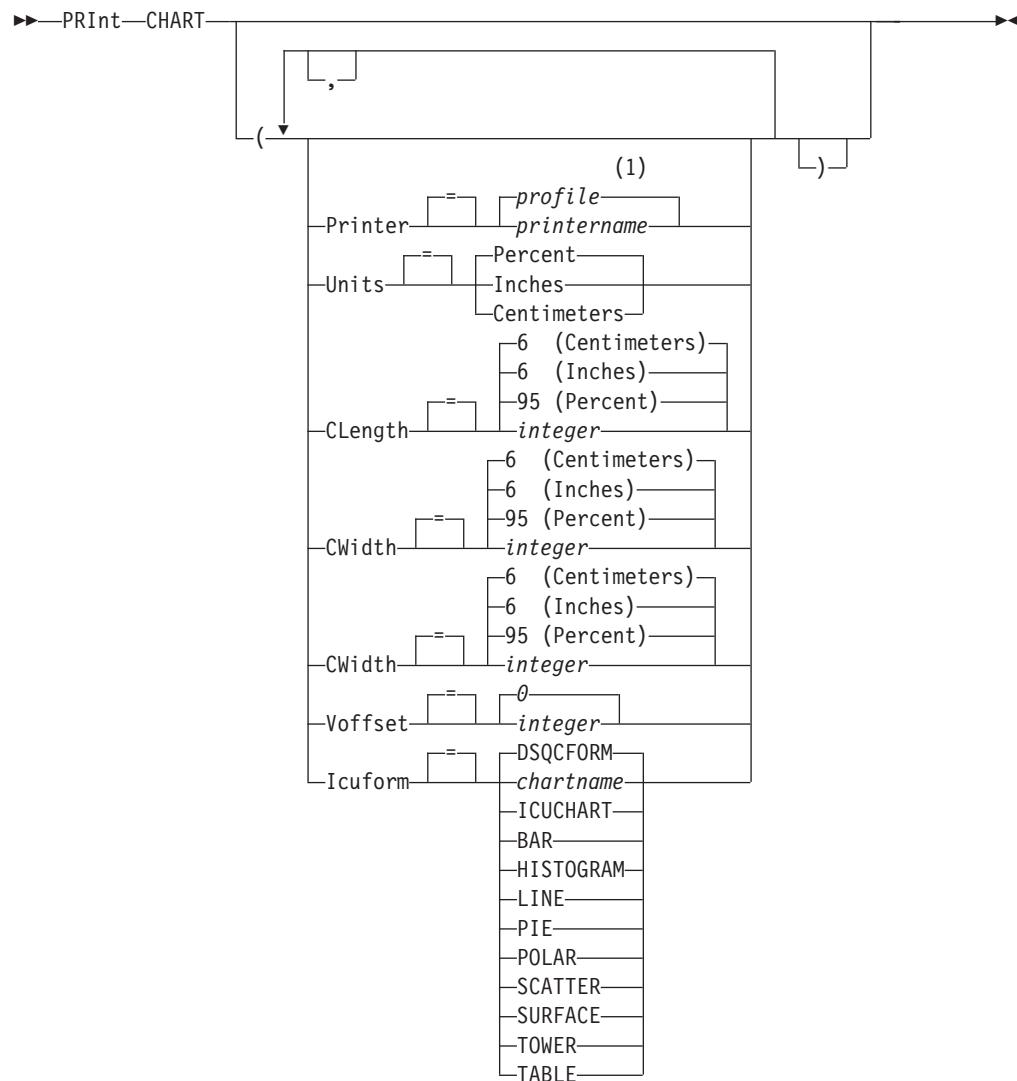
**CICS queue:**

**注:**

- 1 該当する場合は、名付けられた対象のタイプが使用されます。 QMF オブジェクトは他のタイプのオブジェクト (たとえば、データベース・オブジェクト) より優先されます。
- 2 プロファイルで設定された値が使用されます。
- 3 このオプションの使用は限定されます。以下の説明を参照してください。
- 4 このオプションの使用は限定されます。以下の説明を参照してください。
- 5 プロファイルで設定された値が使用されます。
- 6 このオプションの使用は限定されます。以下の説明を参照してください。
- 7 プロファイルで設定された値が使用されます。
- 8 この大域変数で設定された値が使用されます。
- 9 このオプションの使用は限定されます。以下の説明を参照してください。
- 10 この大域変数で設定された値が使用されます。
- 11 このオプションの使用は限定されます。以下の説明を参照してください。
- 12 このオプションの使用は限定されます。以下の説明を参照してください。

**図表の印刷**

## PRINT (CICS の場合)



注:

- 1 プロファイルで設定された値が使用されます。

## 説明

### objectname

データベース内の対象の名前です。有効なオブジェクトは下記のとおりです。

- QMF オブジェクト (PROC、QUERY、FORM)
- 表オブジェクト (TABLE、VIEW、SYNONYM、ALIAS)

### PRINTER

PRINT コマンドの出力宛先を指定します。

### printername

プリント出力先を指定します。 GDDM プリンターのニックネームである必要があります。

**blankstring**

キュー宛先を指定します。この値は、单一引用符 ( ' ) で囲まれた 0 から 8 文字のブランクで指定されなければなりません。

このオプションは、図表、書式、または指示照会オブジェクトの場合無効です。

このオプションは、キュー宛先に印刷する場合だけ、すなわちオプション PRINTER=blankstring が指定されている場合だけ有効です。

**QUEUEENAME**

印刷されたオブジェクトを受け取るための CICS データ・キューの名前。デフォルトは、QMF 大域変数 DSQAP\_CICS\_PQNAME の現行値です。

**queueuname**

CICS データ・キューの名前。キュー用の記憶域のタイプは、QUEUEUTYPE パラメーターで指定されたタイプと一致している必要があります。

**QUEUEUTYPE**

QUEUEENAME パラメーターで指定された CICS データ・キュー用に使用される CICS ストレージのタイプを識別します。デフォルトは、QMF 大域変数 DSQAP\_CICS\_PQTYPE の現行値です。

**TS** 助記憶装置上の CICS 一時記憶域を指定します。

**TD** CICS 一時データ・キューを指定します。

**SUSPEND**

データ・キューが使用中で使用不能な場合のアクションを指定します。

**NO** 印刷要求を取り消します。

**YES** データ・キューが使用可能になるまで待ちます。

**LENGTH**

印刷するページの長さを指定します。長さの単位は行です。

**integer**

ページの切れ目までの最大行数を指定します。数値は 1 から 999 までの整数です。

表 5 に、オブジェクトに適用される最小の長さを示します。

表 5. オブジェクトと、印刷時の最小の長さ

オブジェクト	最小の長さ
書式	25
SQL 照会	25
プロシージャー	25
指示照会	25
表	8
QBE 照会	7 (データ・セットに印刷する場合は 5)
プロファイル	7 (データ・セットに印刷する場合は 5)

## PRINT (CICS の場合)

報告書の場合の最小値は、使用される書式、あるいはコマンド・オプション DATETIME と PAGENO の値によって変わります。

印刷される長さの最大値は 66 行です。

**CONT** ページの切れ目がない連続印刷を指定します。

このオプションは、図表、書式、または指示照会オブジェクトの場合、あるいはプリンターナー名が指定された場合には無効です。

### WIDTH

印刷するページの幅を指定します。 幅の単位は 1 バイト文字です。

#### integer

1 行に印刷する最大文字数を指定します。 数値は 22 から 999 までの整数です。

印刷する対象が報告書でない限り、指定した値より長い行は右端で切り捨てられます。この場合、FORM.OPTIONS パネルで行の折り返しを指定した場合を除き、指定した値より長い行は後続のページで書式化されます。

### PAGENO

印刷するオブジェクトにページ番号を入れるかどうかを指定します。

報告書を印刷する場合で、書式に変数 &PAGE が含まれていると、このオプションは無視されます。

**YES** ページ番号がページの下部に組み込まれます。

**NO** ページ番号は組み込まれません。

### DATETIME

印刷するオブジェクトの各ページにシステム日付と時刻を入れるかどうかを指定します。

報告書を印刷する場合で、書式に変数 &DATE または &TIME が含まれていると、このオプションは無視されます。

**YES** 日時がページの下部に組み込まれます。

**NO** 日時は組み込まれません。

**FORM** 報告書を印刷する場合に使用する書式を指定します。

**FORM** 一時記憶域にある現在の書式オブジェクト。これはデフォルトです。

#### formname

データベースにある QMF 書式の名前。一時記憶域にある現行の書式は、この書式で置き換えられます。

**UNITS** 図表寸法パラメーターの CLENGTH、CWIDTH、HOFFSET、および VOFFSET の計測単位を指定します。

### PERCENT

図表寸法は画面サイズに対する相対的な値 (パーセント) です。

### CENTIMETERS

図表寸法をセンチメートルで表現します。

**INCHES**

図表寸法をインチで表現します。

**CLENGTH**

図表域の長さを数値で表現します。計測単位は UNITS パラメーターで決定されます。デフォルトは計測単位によって異なります。

**CWIDTH**

図表域の幅を数値で表現します。 計測単位は UNITS パラメーターで決定されます。デフォルトは計測単位によって異なります。

**HOFFSET**

ページの左端から図表域までの水平オフセットを数値で指定します。計測単位は UNITS パラメーターで決定されます。

**VOFFSET**

ページの上端から図表域までの垂直オフセットを数値で指定します。計測単位は UNITS パラメーターで決定されます。

**ICUFORM**

図表形式の名前を指定します。図表形式は、データを図表に変換するためには必要な仕様を含んでいます。様々な形式が、様々なタイプの図表を作成するために使用されます。

**DSQCFORM**

QMF が提供するデフォルトの図表フォーマットの名前。

このフォーマットは、QMF 管理者がカスタマイズすることができます。カスタマイズされない場合は、棒グラフが使用されます。

**chartname**

図表フォーマットの名前。

**ICUCHART**

GDDM 対話式図表ユーティリティーのデフォルトの図表フォーマットを指定します。

**BAR****HISTOGRAM****LINE****PIE****POLAR****SCATTER****SURFACE****TOWER****TABLE**

QMF が提供する図表フォーマットの名前。

**注**

- 書式の印刷では、書式を構成するすべての部分が印刷されます。
- 報告書を印刷する際に、報告書は書式指定にしたがって印刷されます。
- 表を印刷する場合、表はデフォルト書式でフォーマット設定されます。

## PRINT (CICS の場合)

デフォルト書式以外の書式でフォーマット設定された表を印刷するには、表を表示し、次に目的の書式を表示してから、PRINT REPORT コマンドを発行します。(例 2 (103 ページ) を参照してください。)

しかし、その書式で、データの各行がソートされた順序になっている必要がある場合には(たとえば、制御の切れ目を使用する書式では)、表を表示するのではなく、その表からソート済みの順序で表中のデータを選択するような照会を最初に実行する必要があります。

- 図表を印刷する際に、書式指定がデータに適用され、図表は GDDM 対話式図表ユーティリティーによって書式化されます。
- データ・キューに印刷出力するには、QUEUENAME パラメーターを使用して、CICS 区画外一時データ・キュー (QUEUETYPE=TD) を指定します。最初に CICS DCT (宛先管理テーブル) に、出力をデータ・キューに経路指定するようデータ・キューを定義しておく必要があります。
- 報告書または図表を印刷するときに書式にエラーがあると、最初のエラーが見つかった書式パネルが表示され、エラーが強調表示されます。他のエラーを表示するには、表示された最初のエラーを訂正する必要があります。

報告書を作成するまで検出されないエラーもあります。

- DBCS 印刷装置を使用すると、DBCS データを表示する端末を持っていなくても、DBCS データを含む報告書を印刷することができます。プログラム・パラメーター DSQSDBCS を YES に設定して、QMF を開始します。QMF の開始プロシージャのカスタマイズの詳細については、QMF 管理者にお問い合わせください。
- DBCS データを使用中に QMF がページを分割する場合、ページの左側から 4 バイト目のところで、報告書の 2 番目およびそれに続くページの印刷が再開されます。
- FORM.PAGE パネルで &PAGE、&DATE、および &TIME を指定することによって、ページ番号、日付、時刻をそれぞれ図表の表題に組み込むことができます。
- 印刷される報告書と画面に表示される報告書とは、表 6 に示すように、異なります。

表 6. 表示される報告書と印刷される報告書との違い

報告書部分	表示された報告書	印刷された報告書
ページ数	スクロール可能な 1 ページ	1 ページ以上
ページ見出しと脚注	1 回だけ表示	各ページの先頭と末尾
明細ヘッダー	報告書の始めの最初の明細行の前と続く全画面	報告書の始めの最初の明細行の前と続く全ページ
固定列	報告書を横にスクロールしても動かない	各ページの左端に印刷

- データベース管理者が複数行フェッチ機能を使用するように QMF をセットアップしている場合に、3 部構成の名前を使用して表を印刷するときは、操作しているデータベースは(ローカルとリモートの) 両方とも DB2 (z/OS 版) 8.1.5 以降でなければなりません。そうでなければ、コマンドは失敗します。データベース

| 管理者は、複数行フェッチをオフにできます。この機能の設定については、  
| 「*TSO/CICS 用 DB2 QMF インストールおよび管理の手引き*」に説明されています。

- CICS の場合、PRINT コマンドは長いオブジェクト名をサポートします。プロンプト・パネルでの長い名前の入力方法の詳細は、3 ページの『プロンプト・パネルからの入力』を参照してください。

## 例

1. QMF PRINT コマンドのためのプロンプト・パネルを表示するには次のようにします。

```
PRINT ?
```

2. デフォルト書式以外の書式でフォーマット設定された表を印刷するには、次のようにします。

```
DISPLAY tablename
DISPLAY formname
PRINT REPORT
```

## QMF

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X	X	X

コマンド同義語の認識をバイパスして基本 QMF コマンドを実行するには、QMF というコマンドを使用してください。これにより、基本 QMF コマンドと同じ名前を持つ、サイトの定義コマンドとまぎらわしくなるのを防ぎます。

### 基本コマンドの発行

►►—*Qmf—qmfcmd*—►►

## 説明

### **qmfcmd**

実行する QMF コマンド。

## 注

QMF コマンドは、コマンド行、プロシージャー、データベース・オブジェクト・リスト・パネル、またはアプリケーションのいずれからでも出すことができます。

## 例

使用しているサイトが別の機能を持つ LIST コマンドを定義している場合に QMF データベース・オブジェクト・リストを表示するには、次のように入力します。

```
QMF LIST
```

## REDUCE

### REDUCE

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X	X	X

REDUCE コマンドは、報告書および QBE で使用されます。このコマンドの詳細については、DB2 QMF 使用の手引きを参照してください。

►►—REDuce—►►

### REFRESH

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X	X	X

REFRESH は、以下の場合に使用されます。

- データベース対象リストで、リストを再作成します。
- 変更キーを押す前に、表編集変更パネルで入力されたキーを廃棄します。パネルは、データベースにあるまだ未変更の行の値で更新されます。

►►—REFresh—►►

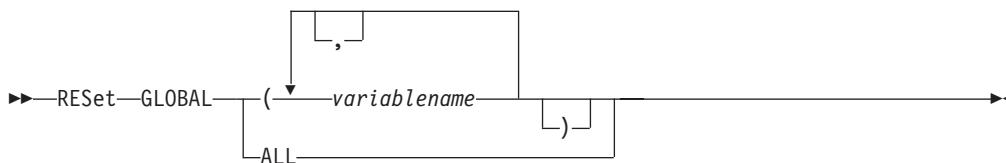
表エディターでは、入力済みの項目が REFRESH コマンドによってすべて失われる前に、確認パネルを表示することができます。結合編集表コマンドでオプション CONFIRM=YES を使用することによって使用可能になり、大域変数 DSQCP\_TEMOD に設定されます。

### RESET GLOBAL

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X	X	X

RESET GLOBAL コマンドは、SET GLOBAL コマンドで設定された大域変数の名前と値を削除します。

#### 大域変数のリセット



## 説明

### variablename

削除する特定の変数の名前です。 SET GLOBAL コマンドで設定された変数を 10 個まで指定できます。

**ALL** これまでに SET GLOBAL コマンドで設定されているすべての変数の名前と値を削除します。いくつもの大域変数を定義していないとき、あるいは大域変数の名前をよくおぼえていないときは、このパラメーターで一度に全部の大域変数をリセットできます。

## 注

- 大域変数は、照会、プロシージャー、書式で使用できますが、表エディターでは使用できません。
- RESET GLOBAL ? を発行すると、リセットする変数の名前を入力できるプロンプト・パネルが表示されます。
- 大域変数リスト・パネルで、削除したい変数の行にカーソルを置いてから Delete キーを押すと、変数をリセットできます。

## 例

- これまでに設定された大域変数の値をすべて削除するには、次のようにします。

RESET GLOBAL ALL

- DEPT と LOCATION という名前の変数の値だけを削除するには次のようにします。

RESET GLOBAL (DEPT LOCATION)

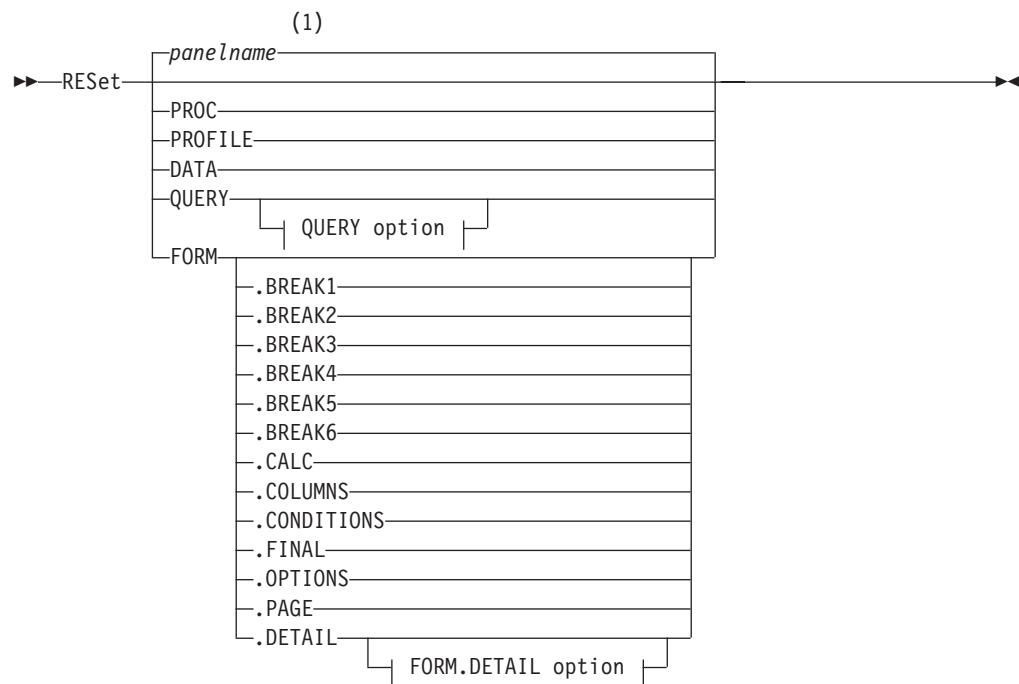
## RESET オブジェクト

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X	X	X

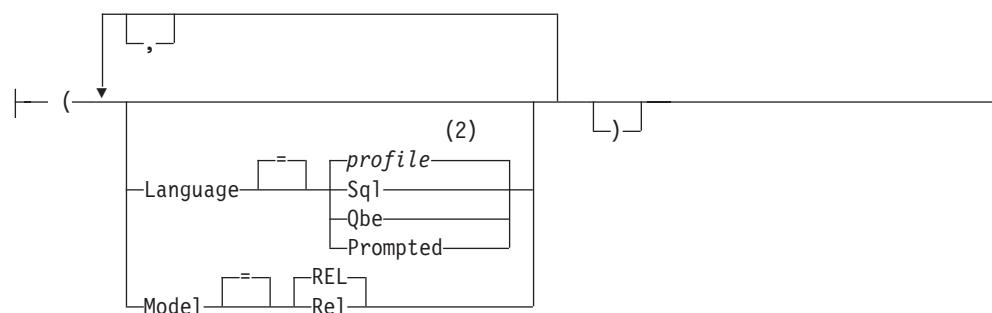
RESET コマンドは、一時記憶域にあるオブジェクトを初期状態に復元します。

### 一時記憶域にある QMF オブジェクトのリセット

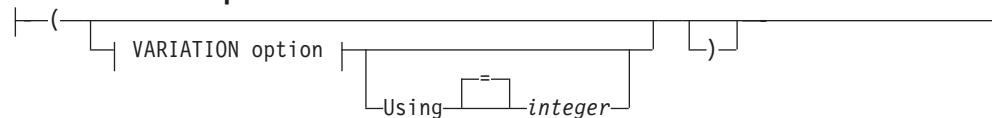
## RESET オブジェクト



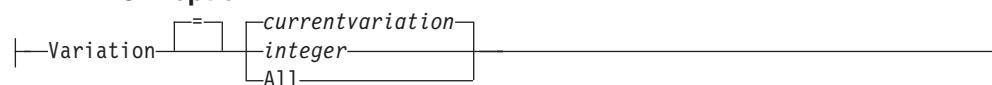
### QUERY option:



### FORM.DETAIL option:



### VARIATION option:



### 注:

- 1 該当する場合は、現在表示されている QMF 対象パネルの名前が使用されます。
- 2 プロファイルで設定された値が使用されます。

## 説明

**PROC** 空のプロシージャー・パネルを表示します。

### PROFILE

値をリセットしたプロファイルを現行ロケーションのデータベースに保管するため表示します。

**DATA** DATA 一時記憶域にある全データを除去し、データベースのカーソルをクローズします。一時記憶域にある REPORT オブジェクトは廃棄されます。RESET コマンドが REPORT パネルから発行された場合は、QMF ホーム・パネルが表示されます。

### QUERY

空の照会パネルを表示します。

#### QUERY オプション

##### LANGUAGE

照会パネルで、どの照会言語を初期化するのかを指定します。

##### SQL

プランク SQL 照会パネルを表示します。

##### QBE

プランク QBE 照会パネルを表示します。

##### PROMPTED

プランクの「指示照会」パネルを表示し、新規の「指示照会」ダイアログを開始します。

##### MODEL

照会に使用するデータ・モデルを指定します。リレーションナル・データだけが、サポートされている値 (REL) です。

**FORM** 書式のすべての部分がデフォルト値にリセットされた FORM.MAIN パネルを表示します。デフォルトは DATA オブジェクトにある列情報と一致するように設定されています。DATA 対象が空の場合、その形式のカラム情報はなくなります。

現行のパネルが FORM.MAIN の場合、RESET コマンドのデフォルトのオブジェクトは FORM です。

#### FORM.COLUMNS

書式の該当する部分だけが、DATA オブジェクトにある列情報と一致するようにリセットされて、FORM.COLUMNS パネルを表示します。

DATA オブジェクトが空の場合は、列情報はありません。

#### FORM.BREAK1

#### FORM.BREAK2

#### FORM.BREAK3

#### FORM.BREAK4

#### FORM.BREAK5

#### FORM.BREAK6

#### FORM.CALC

#### FORM.CONDITIONS

## RESET オブジェクト

### **FORM.FINAL** **FORM.OPTIONS** **FORM.PAGE** **FORM.DETAIL**

書式の該当する部分だけをデフォルト値にリセットして、指定された書式パネルを表示します。

### **FORM.DETAIL オプション**

#### **VARIATION**

フィールド・リセットで表示するために詳細バリエーションを指定します。

このオプションを省略すると、現在の詳細バリエーションがリセットされます。例外は、複数の詳細バリエーションが存在していて、現行のパネルが FORM.DETAIL でない場合です。この状態では、このオプションを指定する必要があります。

#### **integer**

詳細バリエーションの数です。1から99の整数でなければなりません。

指定された詳細バリエーションがまだ作成されていなければ、番号は、既存の全詳細バリエーションに続く次の順序の番号になります。

**ALL** すべての詳細バリエーションを省略時値へリセットします。

#### **USING**

別のバリエーションへリセットあるいは作成するためのテンプレートとして使用する詳細バリエーションを指定します。

これは、詳細パネルにいくつかの変更を加えたあと、同様の変更を盛り込んだ別のバリエーションを作成する場合に有用です。

#### **integer**

既存の詳細バリエーションの数です。1から99の整数でなければなりません。

## 例

1. QMF RESET コマンドのためのプロンプト・パネルを表示するには次のようにします。

```
RESET ?
```

2. 空の SQL 照会パネルを表示する場合:

```
RESET QUERY (LANGUAGE=SQL)
```

3. QMF 一時記憶域内のデータを消去する場合:

```
RESET DATA
```

4. ユーザー・データ用のデフォルト値に設定された FORM.BREAK6 パネルを表示するには次のようにします。

```
RESET FORM.BREAK6
```

5. FORM.DETAIL バリエーション 1 だけをリセットするには次のようにします。

```
RESET FORM.DETAIL (VARIATION=1)
```

6. テンプレートとして詳細バリエーション 1 を使用し、詳細バリエーション 2 をリセットする場合:

```
RESET FORM.DETAIL (VARIATION=2 USING=1)
```

7. すべての詳細バリエーションをリセットする場合:

```
RESET FORM.DETAIL (VARIATION=ALL)
```

## RETRIEVE

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X	X	X

RETRIEVE コマンドは、コマンド行に入力されたコマンド（または、コマンドの一部）を再表示します。 RETRIEVE コマンドを使用すると、コマンドを再入力せずにコマンドを変更することができます。

### 直前のコマンド行項目の検索



### これまでのコマンド行入力の取り出し (オプションの構文)



## 説明

RETRIEVE を使用して、QMF コマンド行に最近入力されたコマンドを表示します。

RETRIEVE コマンドの代わりに、1 つ以上の ? 文字をコマンド行に入力できます。指定する ? 文字の数に従って、どのくらい前に入力したコマンドかが決まり、? 文字の数を 1 つ多くすると、それよりもまた 1 つ前に入力したコマンドが表示されます。たとえば、3 つ前のコマンド行入力を取り出すには、??? と入力します。

RETRIEVE を入力した後で受け取る確認メッセージは、取り出されるコマンドが、最も新しいコマンドからいくつ前に入力されたものかを示します。最も古いコマンドが取り出されたあと、再度 RETRIEVE コマンドを入力すると、コマンドの取り出しが最初に折り返して、最も新しく入力されたコマンドが再度表示されます。

## 注

- 複数の同一のコマンドが連続して実行される場合、RETRIEVE コマンドで 1 つだけが再表示されます。

## RETRIEVE

- コマンドを実行するのにファンクション・キーが使用された場合、そのときにコマンド行に入力されたテキストだけが再表示されます。そのコマンドを実行するためには、同じファンクション・キーを再度押す必要があります。
- コマンドが再表示されている状態で Enter キーを押すと、そのコマンドを再度発行できます。コマンドが完全でなければ、コマンドを変更してから Enter キーを押すか、コマンド行に表示されたテキストと互換のファンクション・キーを押します。再表示されるテキストが大文字に変換されるかどうかは、プロファイル中の CASE パラメーターの値に従います。
- コマンド行にすでにテキストがあるところへ RETRIEVE コマンドを入れると、
  - ? とテキストのそれ以外の部分との間の空白があるかどうかにかかわらず、1 つ以上の? を入力することができます。たとえば、すでにコマンド DISPLAY QUERY がコマンド行に表示されているところに、? を入力して ??DISPLAY QUERY とした場合、これは受け入れられます。
  - RET を入力することができますが、RET とテキストのそれ以外の部分との間に少なくとも 1 つのプランクがなければなりません。たとえば、次は受け入れられます。

RET LAY QUERY

次は受け入れられません。

RETPLAY QUERY

## 例

いずれかのオブジェクト・パネル (LIST パネル以外) または QMF ホーム・パネルから、最も新しいコマンド行入力を取り出すには、コマンド行に次を入力します。

RETRIEVE

代わりに、コマンド行に次を入力することもできます。

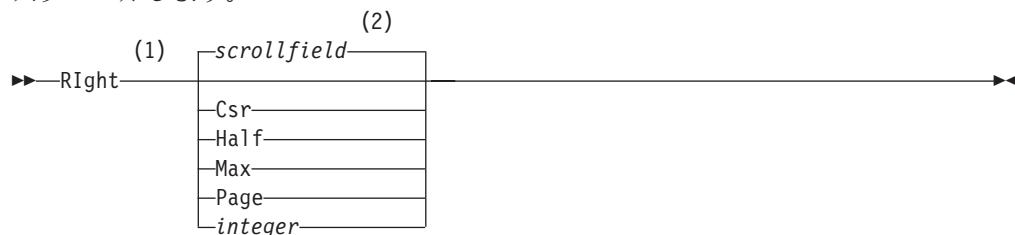
?

---

## RIGHT

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X	X	X

RIGHT コマンドは、QBE 照会パネルまたは報告書パネルの右方の境界に向かってスクロールします。



**注:**

- 1 活動中のパネルに画面移動フィールドがある場合にのみ画面移動量を指定します。その他のすべての場合は、PAGE が想定されます。
- 2 画面移動フィールドに表示される値が使用されます。この値は、大域変数 DSQDC\_SCROLL\_AMT にも保持されます。

**説明**

<b>CSR</b>	右方へスクロールして、カーソルがあるカラムをパネルの左端に位置変更します。カーソルがパネルの右端にある場合には、RIGHT CSR と RIGHT PAGE は同じ効果があります。
<b>HALF</b>	パネルの半分の幅だけ右方へスクロールするか、(その前に右方の境界に到達する場合は) 右方の境界までスクロールします。
<b>MAX</b>	右方の境界までスクロールします。
<b>PAGE</b>	パネルの幅だけ右方へスクロールするか、(その前に右方の境界に到達する場合は) 右方の境界までスクロールします。
<b>integer</b>	この桁数分右側に画面移動します (総数は 1 から 9999 の範囲)。

**注**

- MAX は現行のコマンドの間だけ有効です。この値はコマンドが完了すると SCROLL フィールドには残りません。大域変数 DSQDC\_SCROLL\_AMT をこの値に設定することはできません。
- 報告書で右方にスクロールするときは、RIGHT ファンクション・キーを使用してください。スクロール量を指定するには、スクロールしたい桁数をコマンド行にタイプしてから、RIGHT ファンクション・キーを押します。

**RUN**

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X	X	*

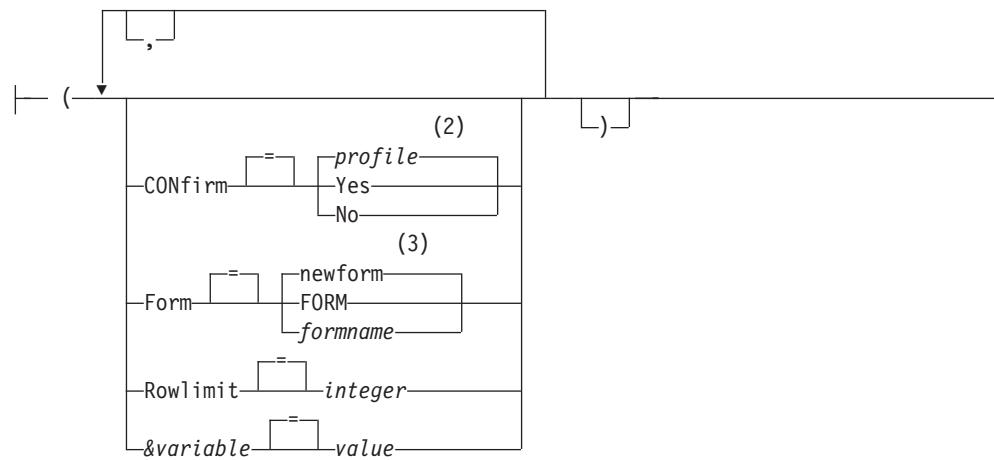
RUN コマンドは、照会またはプロシージャーを QMF 一時記憶域から実行します。または、現行ロケーションのデータベースから実行します。

**一時記憶域からの QMF 照会またはプロシージャーの実行  
(1)**

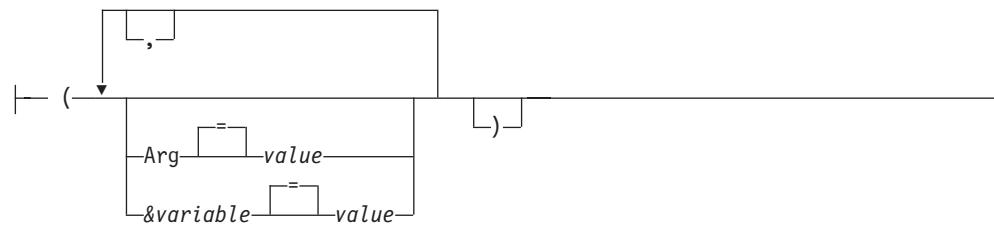


**QUERY option:**

## RUN



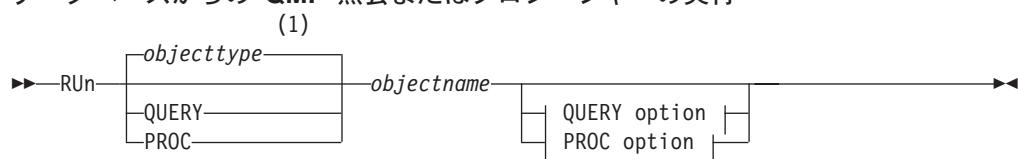
## PROC option:



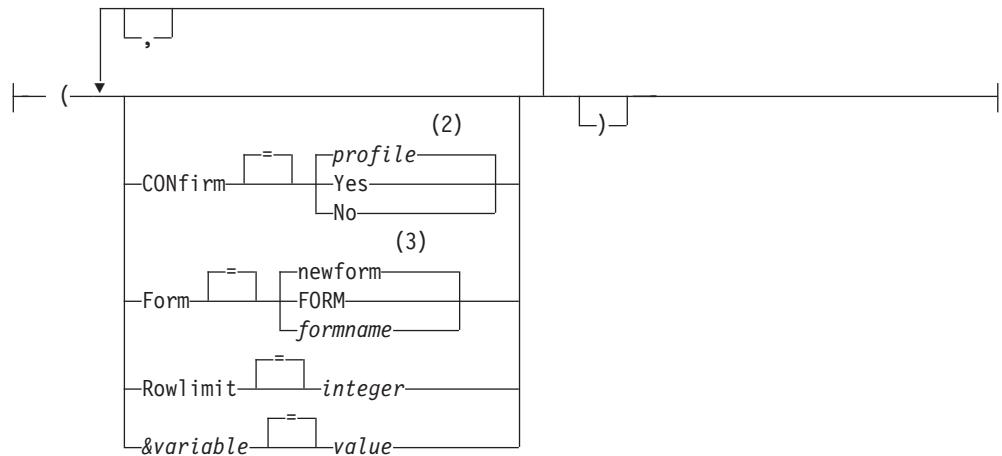
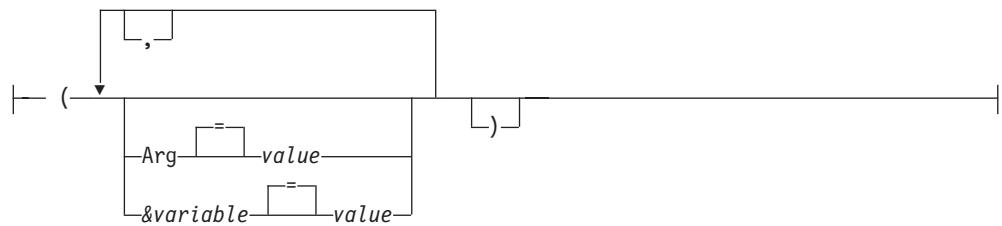
## 注:

- 1 該当する場合は、現在表示されている QMF 対象パネルの名前が使用されます。
- 2 プロファイルで設定された値が使用されます。
- 3 新しい書式対象は、選択されたデータに基づいて作成されます。

## データベースからの QMF 照会またはプロシージャーの実行



## QUERY option:

**PROC option:****注:**

- 1 該当する場合は、名付けられた対象のタイプが使用されます。 QMF オブジェクトは他のタイプのオブジェクト（たとえば、データベース・オブジェクト）より優先されます。
- 2 プロファイルで設定された値が使用されます。
- 3 新しい書式対象は、選択されたデータに基づいて作成されます。

## 説明

**objectname**

データベースにある QMF オブジェクトの名前。別のユーザーが所有する対象は、その所有者の名前で修飾する必要があります。

**&variable**

RUN コマンド用の置換変数を識別します。このオプションでは、変数に対して 1 バイト文字で 55 文字までの値を割り当てることができます。1 つのコマンドで、10 個までの置換変数を指定することができます。

変数名は アンパーサンド記号で始める必要があります。線形手順内で RUN コマンドを実行する場合は、アンパーサンド記号を 2 個続けて使用してください。

**value** 置換変数の内容を構成する文字ストリング。

ブランク文字を含む値は、区切り文字で囲む必要があります。置換変数の値で有効な区切り文字は、単一引用符、二重引用符および括

弧です。区切り文字が引用符の場合は、引用符は値の一部として組み込まれます。区切り文字が括弧の場合は、括弧は値の一部として組み込まれません。

#### **Query option:**

##### **CONFIRM**

照会が次のいずれかを行う場合、確認パネルを表示するかどうかを指示します。

- データベースにあるオブジェクトを変更する。
- リソース限界機能 (DB2 Governor) で指定された見積制限を超える。

**FORM** 選択されたデータを初期化するときにどの QMF FORM を使用するかを指示します。

##### **newform**

このオプションを指定すると、QMF は新規の書式オブジェクトを作成し、一時記憶域にある現行の書式を置き換えます。新しい書式は、照会によって選択されたデータと一致します。これは、表示された報告書に省略時の書式化を提供します。

**FORM** 現在一時記憶域にある QMF FORM が使用されます。この選択項目を使用するには、FORM が一時記憶域になければなりません。

現行 FORM が選択されたデータに対して適切な場合は、報告書が表示されます。

##### **formname**

データベース内の QMF FORM の名前です。別のユーザーが所有する書式は、その所有者の名前で修飾する必要があります。その他の要件は次のとおりです。

- FORM は現行ロケーションのデータベースに存在する必要があります。
- 別のユーザーが所有する書式の使用を許可されている必要があります。

指定された FORM が一時記憶域内の現行 FORM になります。 FORM が選択されたデータに対して適切な場合、報告書が表示されます。

##### **ROWLIMIT**

照会によって返されるデータの行数の限界を設定します。報告書に使用可能なデータ行の数を 1 から 99999999 までに制約したい場合だけ、このオプションを使用してください。

##### **integer**

1 から 99999999 の間の整数です。

#### **PROC option:**

**ARG** 引数ストリングをロジック (REXX 手順) を持つ QMF 手順に渡します。このオプションでは、80 文字までの 1 つの引数を渡すことができます。

引数ストリングは、REXX コマンド PARSE ARG または REXX 関数 ARG(1) を使用して REXX 手順によって受け取られます。

**value** 引数の内容を構成する文字ストリングです。

ブランク文字を含む値は、区切り文字で囲む必要があります。引数の値で有効な区切り文字は、単一引用符、括弧、および二重引用符です。区切り文字として二重引用符を使用する場合、二重引用符も値の一部に組み込まれます。

## 注

- QMF SAVE コマンドの SHARE=YES オプションを使用して QMF 対象をデータベースに保管することにより、QMF 対象を他のユーザーと共に用することができます。
- QMF 管理権限は RUN コマンドを拡張しません。SHARE=NO オプションでデータベースに保管されている QMF 対象は直接 QMF 管理者によって実行できません。しかし、QMF 管理者は、RUN コマンドを発行する前に DISPLAY コマンドを使用してこれらのオブジェクトを一時記憶域内に入れることができます。その後 RUN コマンドを発行できます。
- QMF 照会またはプロシージャーのオブジェクトで使用される変数は、RUN コマンドの実行までに値を設定しておく必要があります。指示パネルが表示され、以下のいずれでも指定されていない変数の値を収集します。
  - このコマンドの一部である *&variable* オプション。
  - 前に設定された大域変数
- 英語で書かれた QMF コマンドを含む QMF プロシージャーは、大域変数 DSQEC\_NLFCMD\_LANG が 1 に設定されている場合、どの QMF セッションでも実行できます。ただし、QMF プロシージャーがそれ以外の QMF 言語で保管されていると、同じ言語によるセッションでしか実行できません。
- RUN コマンドを使用すると、QMF は対象の最終使用日付を更新します。この日付はデータベース対象リスト内に表示され、特定の QMF 対象の最終アクセス日または最終実行日を示します。QMF はこのフィールドを、オブジェクトごとに 1 日に 1 回、最初に使用されたときに更新します。
- QMF 手順または照会対象の注釈は変数として処理できません。変数の値の中で、連続する 2 個のダッシュ (--) を使用しないでください。使用すると、コメントとしてではなく、実行すべきコマンドまたは照会の一部として扱われるからです。
- ロジック (REXX 手順) を持つ QMF 手順は CICS 環境ではサポートされていません。

- QMF は、QMF 書式に指定されたオプションに基づいて、結果の報告書の中のデータをフォーマット設定します。編集コードは、いろいろなタイプのデータの表示方法を制御します。詳細については、261 ページの『編集コード』を参照してください。
- データベース管理者が複数行フェッチ機能を使用するように QMF をセットアップしている場合に、3 部構成の名前を使用してリモート・データベースで照会を実行するときは、操作しているデータベースは（ローカルとリモートの）両方とも DB2 (z/OS 版) 8.1.5 以降でなければなりません。そうでなければ、コマンドは失敗します。データベース管理者は、複数行フェッチをオフにできます。この機能の設定については、「*TSO/CICS 用 DB2 QMF インストールおよび管理の手引き*」に説明されています。
- RUN コマンドは長いオブジェクト名をサポートします。プロンプト・パネルでの長い名前の入力方法の詳細は、3 ページの『プロンプト・パネルからの入力』を参照してください。

## Run コマンドの変数値

QMF は、ブランク、コンマ、左括弧、右括弧、单一引用符、二重引用符、等号のいずれかを見つけると、そこが RUN コマンドに指定された変数の値の終わりであると見なします。値が引用符で囲まれていれば、その引用符も値に含まれます。値が括弧で囲まれていれば、その括弧は値に含まれません。最終値に括弧を含めたいときは、括弧を二重にしてください。たとえば、コマンド行からの処理で、单一引用符または二重引用符が見つかると、QMF はそれと対になる文字を検索します。引用符で始まるストリングは、同種の引用符で終わる必要があります。最初の引用符と対になる引用符が見つからないと、QMF は、コマンド指定の残りの部分を、最初の引用符とともに値の一部として受け入れます。

ブランク、コンマ、右括弧、左括弧、单一引用符、二重引用符、等号といった文字を変数に含めるには、それらを含んだ値 の指定を括弧で囲んでください。たとえば、次の RUN コマンドは、変数 &X の値指定が最初のコマンドで終了し、NAME を RUN キーワードとしては受け入れません。

```
RUN QUERY (&X=DEPT,NAME,SALARY)
```

この照会をコマンド行から入れて正しく処理させるためには、次のように括弧を追加してください。

```
RUN QUERY (&X=(DEPT,NAME,SALARY))
```

プロシージャー内の RUN コマンドで照会を実行するときは、変数パラメーターを用いて、その照会内の変数に値を渡すことができます。たとえば、照会で &DEPARTMENT という名前の変数を使用していると想定します。&&DEPARTMENT = 66 の指定は、&DEPARTMENT をプロシージャーの変数とせずに、値 66 を照会内の変数 &DEPARTMENT に割り当てます。&&DEPARTMENT = &DEPT の指定は、&DEPT をプロシージャーの変数とし、その値を照会内の &DEPARTMENT に割り当てます。変数の値は、RUN コマンドを実行する前に、SET GLOBAL コマンドで設定しておくこともできます。しかし、RUN コマンドで値を指定すると、それが SET GLOBAL で設定された同パラメーターの値をオーバーライドします。

照会またはプロシージャを実行するとき、変数の値がまだ設定されていないと、QMF は値を指定できるようにプロンプト・パネルを表示します。変数に割り当てる値が 1 バイト文字で 55 文字（または同等数の 2 バイト文字）を超えないように注意してください。

1 つの照会またはプロシージャで最大 100 個の変数に値を指定できます。そのうち RUN コマンドで指定できるのは 10 個の変数までで、残りは SET GLOBAL で指定しなければなりません。QMF は、まず値の有無についてコマンドを調べ、次にグローバル値を探します。制限を超えていると、コマンドは拒否され、エラー・メッセージができます。照会のパラメーターと一致しない変数名は無視されます。

線形手順の中で SET GLOBAL によって変数に値を割り当てても、同じプロシージャ中のコマンドはその変数値を使用できません。しかし、そのプロシージャによって呼ばれる照会やプロシージャの中では、その変数値を使用できます。

実行するオブジェクトが変数を用いる照会で、その変数に大域変数が設定されていない場合、&変数パラメーターの指定を省略すると、プロンプト・パネルが表示されます。そこで変数値を指定できます。RUN コマンドでは、変数を他の変数で置き換えることはできません。

## システム考慮事項

RUN PROC コマンドで指定されたプロシージャに含まれているコマンドは、QMF が稼働しているシステムで実行されます。したがって、QMF が稼働しているシステムで無効なコマンドがプロシージャに含まれていれば（たとえば、TSO プロシージャに含まれた CICS コマンドや、その反対に、CICS プロシージャに含まれた TSO コマンドなど）、そのようなコマンドは、プロシージャを実行したときに失敗になります。

## 例

1. QMF RUN コマンドのためのプロンプト・パネルを表示します。

```
RUN ?
```

2. 現在一時記憶域にある照会を実行し、データベースからの書式を持ち、別のユーザー (MARIA) が所有する報告書 (REPORT3) を書式化する場合は、次のようにします。

```
RUN QUERY (FORM=MARIA.REPORT3
```

3. 置換変数 YR に値を設定して、データベース (SALESQ) から照会を実行するには次のようにします。

```
RUN QUERY SALESQ (&YR=1999
```

同じコマンドが QMF 線形手順にある場合には、コマンドは次のようになります。

```
RUN QUERY SALESQ (&&YR=1999
```

4. RUN QUERY コマンドを実行すると、現行ロケーションに保管されている照会が実行されます（あるいは、現行ロケーションに見つかった書式が使用されます）。たとえば、照会 STATSCHK に次が含まれているとします。

```
SELECT * FROM JOHNSON.STATUS
```

## RUN

次のコマンドは、照会、書式、およびデータを、現行ロケーションから取り出します。

```
RUN QUERY STATSCHK (FORM=FORMCHK)
```

しかし、次の照会の場合は異なります。

```
SELECT * FROM BILLINGS.JOHNSON.STATUS
```

この場合、次のコマンドは、データを BILLINGS ロケーションから、照会と書式を現行ロケーションから取り出します。

```
RUN QUERY STATSCHK (FORM=FORMCHK)
```

5. 複数の行にわたる長いオブジェクト名を伴う RUN QUERY コマンドを発行するには、次のようにします。

```
RUN QUERY  
+"LONGOWNERID2345678901234567890123456789012345678901234567890123  
+4567890123456789012345678901234567890123456789012345678901234567890123  
+6789012345678901234567890123456789012345678901234567890123456789012345678  
+90123456789012345678901234567890123456789012345678901234567890123456789012345678
```

---

## SAVE

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X	X	X

SAVE コマンドは、現在 QMF 一時記憶域にあるオブジェクトを、現行ロケーションのデータベースに保管します。

### QMF プロファイルのデータベースへの保管

(1)



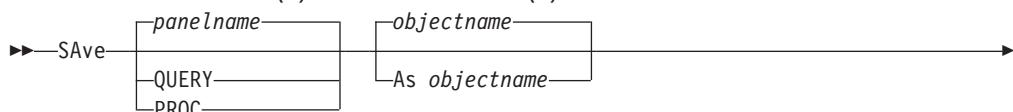
### 注:

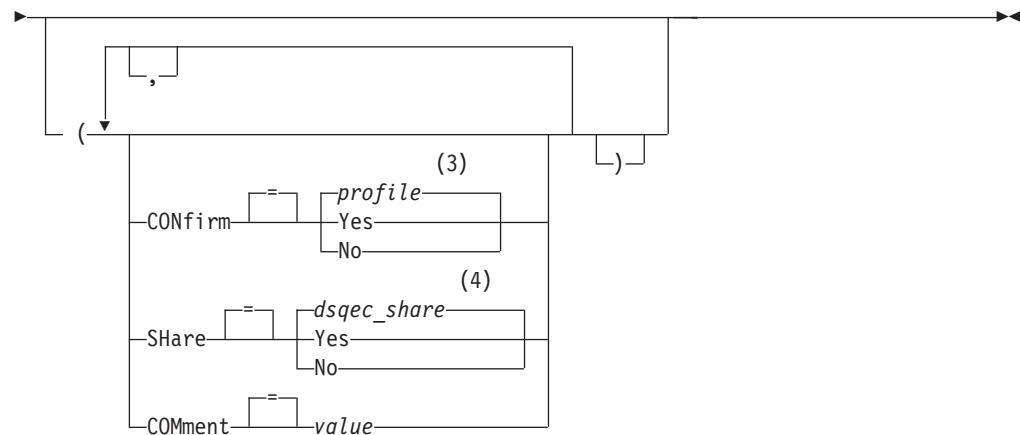
- 1 該当する場合は、現在表示されている QMF 対象パネルの名前が使用されます。

### QMF 照会またはプロシージャーのデータベースへの保管

(1)

(2)

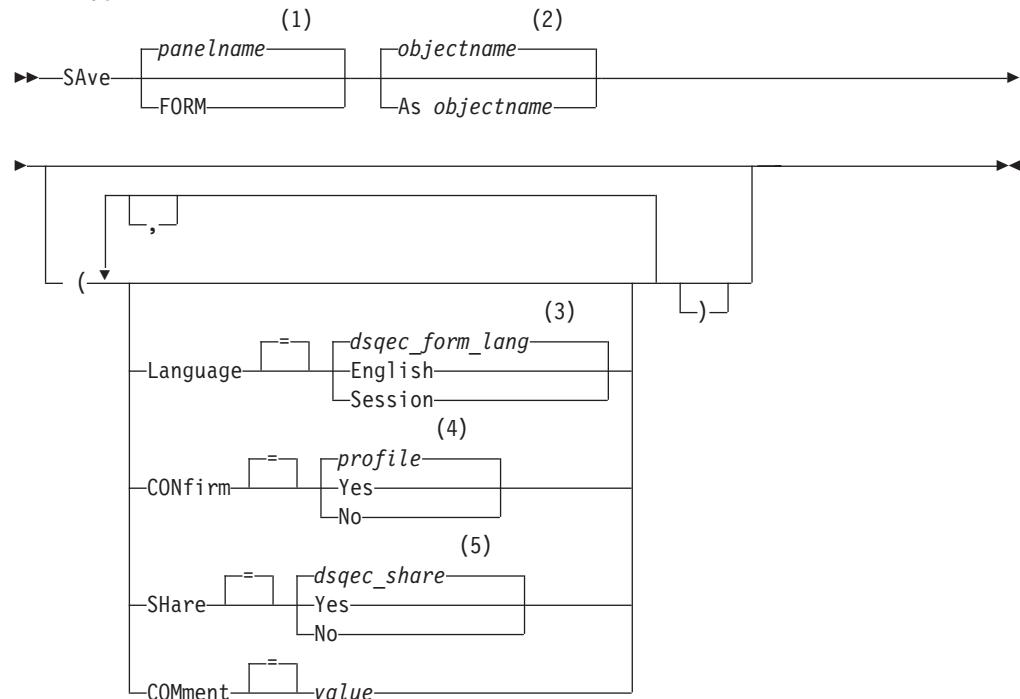




注：

- 該当する場合は、現在表示されている QMF 対象パネルの名前が使用されます。
  - 存在する場合は、現在 QMF 一時記憶域にある対象の名前が使用されます。
  - プロファイルで設定された値が使用されます。
  - 置換される対象については、現在の値は変更されないままになります。それ以外の場合は、この大域変数で設定された値が使用されます。

## QMF 書式のデータベースへの保管



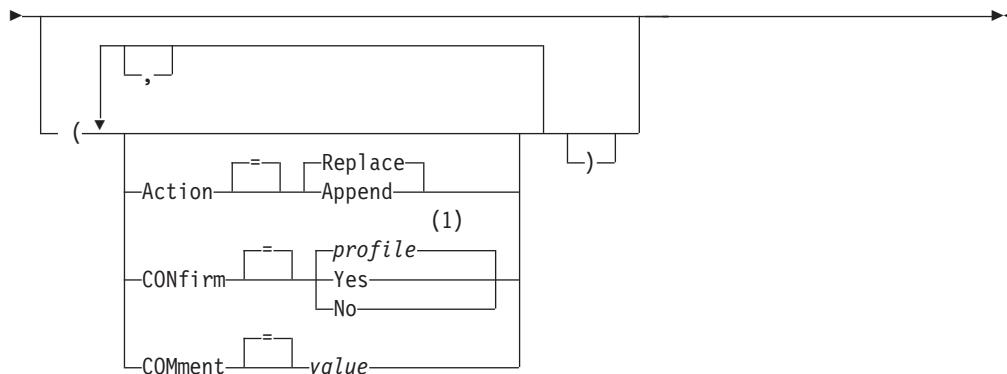
## SAVE

注:

- 1 該当する場合は、現在表示されている QMF 対象パネルの名前が使用されます。
- 2 存在する場合は、現在 QMF 一時記憶域にある対象の名前が使用されます。
- 3 この大域変数で設定された値が使用されます。
- 4 プロファイルで設定された値が使用されます。
- 5 置換される対象については、現在の値は変更されないままになります。それ以外の場合は、この大域変数で設定された値が使用されます。

### QMF データのデータベースへの保管

►►—SAve—DATA—As *tablename*—►►



注:

- 1 プロファイルで設定された値が使用されます。

## 説明

### objectname

データベース内の QMF 対象の名前です。オブジェクト名の最大長は、現在接続しているデータベースによって決まります。

### tablename

データベース内の表の名前です。

既存のデータベース・オブジェクトの場合、これは、表、視点、同義語、または別名の名前にすることができます。

### ACTION

保管されるデータで既存のデータベース表全体を置き換えるのか、それを既存の表に付加するのかを指定します

### LANGUAGE

保管される書式の中の QMF キーワードを、英語または現行の NLF セッションの言語のどちらで記録するかを指示します

英語の QMF キーワードを含む QMF 書式は、すべての QMF セッションで使用できます。QMF がサポートする別の各国語の QMF キーワードを含む QMF 書式は、それと同じ QMF 各国語のセッション内でのみ使用できます。

**CONFIRM**

このコマンドがデータベース内のオブジェクトを置き換えようとするときに、確認パネルを表示するかどうかを指示します。

**SHARE**

他の QMF ユーザーが、保管されたオブジェクトをアクセスできるかどうかを決定します

**COMMENT**

保管オブジェクトと一緒にコメントが保管されます。QMF コマンド行またはプロシージャーの中で SAVE コマンドを使用してオブジェクトと一緒にコメントを保管しようとする場合、コメント・テキストは 78 文字まで許されます。SAVE コマンド・プロンプト・パネルを使用してコメントを入力する場合、コメントの長さは 57 文字までです。

**value** 注釈の内容を構成する文字ストリングです。

ブランク文字を含む値は、区切り文字で囲む必要があります。注釈値のための有効な区切り文字は単一引用符、括弧、および二重引用符です。

**注**

- QMF 管理者は別のユーザーの QMF 対象を保管できます。(例 6(122 ページ) を参照してください。)
- データベースに保管しようとしたときに、指定した名前のオブジェクトがすでに存在している場合、QMF は下記の条件に従ってそのオブジェクトを置き換えます。
  - 照会は照会のみを置き換えることができる。
  - 手順は手順のみを置き換えることができる。
  - 書式は書式のみを置き換えることができる。
  - データは類似の表対象のみを置き換えることができる。

同類の表とは、列の数が同じで、対応する列が同じデータ・タイプと長さをもつ表のことです。列名とラベルは一致する必要はありません。

- 既存の表に保管するときには、列名とラベルは変更されません。表が存在しない場合は、QMF データ対象に記録された列名およびラベルを使用して新しい表が作成されます。
- オブジェクトをリモート・ロケーションに保管することができます。まず、QMF CONNECT コマンドを使用して、リモート・ロケーションをユーザーの現行ロケーションとして設定し、それから SAVE コマンドを発行します。

現行ロケーションが DB2 (z/OS 版) サーバーであれば、表名として 3 パート名を指定することにより、リモート・ロケーションにある既存の表に保管することができます。この方法では、新しい表、または QMF オブジェクトは保管できません。データベース管理者が複数行フェッチ機能を使用するように QMF をセットアップしている場合、3 部構成の名前を使用するときは、操作しているデータベースは (ローカルとリモートの) 両方とも DB2 (z/OS 版) 8.1.5 以降でなければなりません。そうでなければ、コマンドは失敗します。データベース管理者は、複数行フェッチをオフにできます。この機能の設定については、「*TSO/CICS 用 DB2 QMF インストールおよび管理の手引き*」に説明されています。

## SAVE

- 所有していない表または遠隔表の注釈を、3部分名を使用して置き換えることはできません。
- 10進浮動小数点 (DECFLOAT) 列を含むデータに対して、SAVE DATA コマンドは発行できません。
- XML 列を含むデータは、32K に制限されます。
- SAVE コマンドは長いオブジェクト名をサポートします。プロンプト・パネルでの長い名前の入力方法の詳細は、3ページの『プロンプト・パネルからの入力』を参照してください。

## 例

- 書式を保管するために指示パネルを表示する場合:

```
SAVE FORM ?
```

- 注釈を保管された照会に組み込む場合:

```
SAVE QUERY AS STAFFQ2 (COMMENT=(Staff report for departments))
```

- QMF 一時記憶域内の照会を現行ロケーションのデータベースに保管する場合:

```
SAVE QUERY AS HAZEL.QUERY3
```

- QMF 対象を遠隔データベース・サーバー (MADRID) に保管する場合、まずそのロケーションに接続します:

```
CONNECT TO MADRID
```

次に対象を保管する場合:

```
SAVE FORM AS FORMAT2
```

- 現行ロケーションが DB2 (z/OS 版) であり、リモート・データベース・ロケーション (BILLINGS) の表 (HAZEL.STATUS) にデータを保管したい場合、次のようにします。

```
SAVE DATA AS BILLINGS.HAZEL.STATUS
```

- QMF 管理者 (QMFADM) が他のユーザー (HAZEL) 用にプロシージャーを保管します。

```
SAVE PROC AS HAZEL.MONTHLY (COMMENT=(MONTHLY PROCESS))
```

- QMF 線形手順で、複数の行にわたる長いオブジェクト名を持つ SAVE コマンドを使用するには、図 13 に示すように、各継続行の 1 桁目に正符号 (+) を置きます。

```
PROC MODIFIED LINE 1
```

```
SAVE DATA AS  
+ "LOCATION12345678"."LONGOWNERID1234567891123456789212345678931234567894123  
+4567123456789112345678921234567893123456789412345678951234567896123456789  
+712345"."LONGNAME12345678911234567892123456789312345678941234567895123456  
+78961234567897123456789112345678921234567893123456789412345678951234567894123456
```

図 13. 線形手順での、長いオブジェクト名を持つ SAVE コマンドの行の継続

---

## SEARCH

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X	X	X

表エディターでは、SEARCH コマンドは、指定情報をデータベース表の中で探し出します。

### 表エディターを使用した情報の検索

►►—SEArch————►►

#### 注:

- 特定の末尾をもつデータを搜すときは、捜している列のデータ・タイプに注意してください。列幅が固定されていて、列内のデータの長さが可変であれば、検索基準の末尾にパーセント記号を置いて、後続ブランクが何個かかるかもしれないことを表さなければなりません。検索オブジェクトの列のデータ・タイプが VARCHAR であれば、後続ブランクはありません。
- SEARCH モードでは、検索基準を指定してSEARCH ファンクション・キーを押すと、その検索基準に一致する列をもつ行が取り出されます。
- 値の一部しかわからないデータを検索するには、検索基準の中で、検索パターンのワイルドカードとして次の記号を単独または組み合わせて使用します。

#### パーセント記号 (%)

まったく文字を含まない場合も含めて、任意の数と組み合わせの、文字のプレースホルダーとして使用します。

#### 下線 (\_)

1 文字のプレースホルダーとして使用します。

% と \_ は、1 つの値の中で併用できます。それぞれ、繰り返して使用できます。たとえば、検索基準として \_OS% というパターンを使用すると、ROSS や DOS や BOSLEY などの値が見つかります。

---

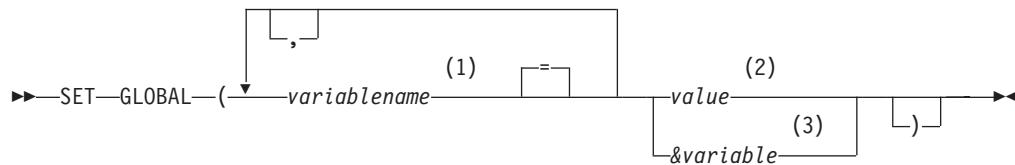
## SET GLOBAL

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X	X	X

SET GLOBAL コマンドは、QMF コマンド行、プロシージャー、または呼び出し可能インターフェースから大域変数に値を割り当てます。 QMF コマンド行またはプロシージャーから、10 個までの置換変数を定義できます。呼び出し可能インターフェースでは、環境で決められた数だけの変数が割り当てられ、またコマンドの正確な構文は使用言語で決まります。このトピックでは、REXX アプリケーションおよびロジックを持つプロシージャーで使用する、コマンドの線形構文を示します。コマンドの拡張構文については、「DB2 QMF アプリケーション開発の手引き」を参照してください。

## SET GLOBAL

### 線形構文 (REXX でのみ使用)



注:

- 1 値が割り当てられる大域変数を識別するします。
- 2 大域変数の内容を構成する文字ストリング。
- 3 大域変数の内容を含む大域変数名。

## 説明

### variablename

値が割り当てられる大域変数を識別するします。

**value** 大域変数の内容を構成する文字ストリング。

ブランク文字を含む値は、区切り文字で囲む必要があります。大域変数の値で有効な区切り文字は、単一引用符、括弧、および二重引用符です。区切り文字として二重引用符を使用する場合、二重引用符も大域変数の一部に組み込まれます。

SET GLOBAL コマンドが線形手順から入力され、変数値が複数の線にスパンする場合は、値は引用符で囲み、各行の最初に継続文字 (+) を使用する必要があります。複数の線にスパンする場合は、括弧は区切り文字として使用できません。

## 注

- 大域変数は照会、プロシージャー、書式で使用できます。 QMF オブジェクトで使用するときは、変数の頭に 1 つ以上のアンパーサンド (&) を付けてください。
- 大域変数名には数字を使用できますが、大域変数名の先頭の文字を数字にすることはできません。
- 大域変数名の先頭の文字は英字 (A から Z) か、または以下のいずれかの特殊文字でなければなりません。

¢ ! \$ ~ { } ? @ # % ¥

- 大域変数名には、ブランクまたは次の文字が含まれていてはなりません。  
., ; : < > ( ) | + - \* / = & \_ ^ "
- 変数名は 1 バイト文字で 17 文字分 (または 2 バイト文字で同等数分) に制限されています。文字定数は单一引用符で囲む必要はありません。
- SET GLOBAL コマンドでは、RUN コマンドおよび CONVERT コマンドと異なり、変数名の頭にアンパーサンドがつきません。
- 疑問符を含む大域変数名は、QMF 書式では認識されません。
- 大域変数を書式変数名や集合変数名にセットしておいても、QMF 書式では認識されません。

- ・大域変数名を DSQ で始めることはできません。DSQ は、QMF で定義ずみの大域変数に使われる予約文字であるためです。
- ・大域変数名に後続ブランクがあっても、それは認識されません。
- ・変数が文字ストリングで、列名、表名、演算子名などの名前であるときは、
  - 値の内部に含まれる引用符をすべて二重にしてください。
  - 単一引用符でストリング全体を囲む。（これらの引用符は値の一部とは見なされません。）

たとえば、SELECT ステートメントが次の場合、

```
SELECT DEPT, &COL FROM &TABLE
```

この SELECT ステートメントの変数を設定する SET GLOBAL コマンドは、次のようなものです。

```
SET GLOBAL (COL='NAME', TABLE='Q.STAFF'
```

- ・変数が、ある列内の値として（SQL ステートメントの WHERE 文節でのみ）使われる文字ストリングであれば、2 種の方法のいずれかを使用して、ストリングを指定することができます。

#### 方法 1 (引用符)

1. 元のストリングで開始する。
2. すべての引用符を二重にしてください（ある場合）。
3. ストリングを 2 対の単一引用符で囲んでください。
4. ストリングの中に入っている 1 個の引用符は、それぞれ 2 個ずつにします（最外部の引用符を除くすべて）。

#### 方法 2 (括弧)

1. 元のストリングで開始する。
2. ストリングを 1 対の括弧で囲んでください。

たとえば、SELECT ステートメントが次の場合、

```
SELECT DEPT FROM &TABLE WHERE NAME=&ABC
```

方法 1 によって値を指定する場合、SET GLOBAL コマンドは、次のようになります（JAMES が変数 ABC に置き換わります）。

```
SET GLOBAL (ABC='''JAMES''' , TABLE='Q.STAFF'
```

方法 2 によって値を指定する場合、SET GLOBAL コマンドは、次のようになります（O'BRIEN が変数 ABC に置き換わります）。

```
SET GLOBAL (ABC=(O'BRIEN) , TABLE='Q.STAFF'
```

- ・変数にブランク、コンマ、単一引用符、二重引用符、等号のいずれかが含まれているときは、値の全体を 1 対の括弧で囲まなければなりません。しかし、対になっていない左括弧や右括弧を含む値、または左括弧で始まるか右括弧で終わる値の場合、括弧の代わりに引用符を使用しなければなりません。

たとえば、次の SELECT ステートメントがあるとします。

```
SELECT &COLS FROM Q.STAFF
```

## SET GLOBAL

この SELECT ステートメントに複数の列名を指定するには、コンマで値を区切る必要があります。コンマのあとに、オプションでブランクを付けることもできます。したがって、SET GLOBAL コマンドは、次の例に示すように、括弧で囲む必要があります。

```
SET GLOBAL (COLS=(NAME, JOB, SALARY)
```

- 少なくとも 1 つは変数を指定しなければなりません。
- 変数の値に引用符が必要な場合は、単一引用符を 2 個続けて使用してください。
- 変数の値として照会コメントを使用しないでください。照会コメントは 2 個のダッシュ (-) で始めますが、データベースはそれを負符号 (-) と解釈します。
- いくつもの変数に値をセットするときは、プロシージャーを使用したほうが簡単に管理できます。
- 変数が数値の場合、引用符を使用する必要はありません。
- 変数名が QMF 製品大域変数プール内に見つからなければ、新規の変数が作成されます。変数名が見つかれば、新しい値が古い値を置き換えます。
- SET GLOBAL コマンドが線形手順から入力され、変数値が複数の線にスパンする場合は、値は引用符で囲み、各行の最初に継続文字 (+) を使用する必要があります。区切り文字が二重引用符のとき、二重引用符は大域変数の一部として大域変数に組み込まれます。複数の線にスパンする場合は、括弧は区切り文字として使用できません。

### 例

- プロンプト・パネルを表示させて、設定したい変数と新しい値をそこで指定したいときは、次のように入力します。

```
SET GLOBAL ?
```

- 変数 DEPT に 38 を、変数 JOB に 'SALES' を割り当てます。

```
SET GLOBAL (DEPT = 38, JOB = '''SALES'''
```

- 値 'O'BRIEN' を変数 NAME に割り当てる方法の 1 つです。

```
SET GLOBAL (NAME ='''O''''BRIEN'''
```

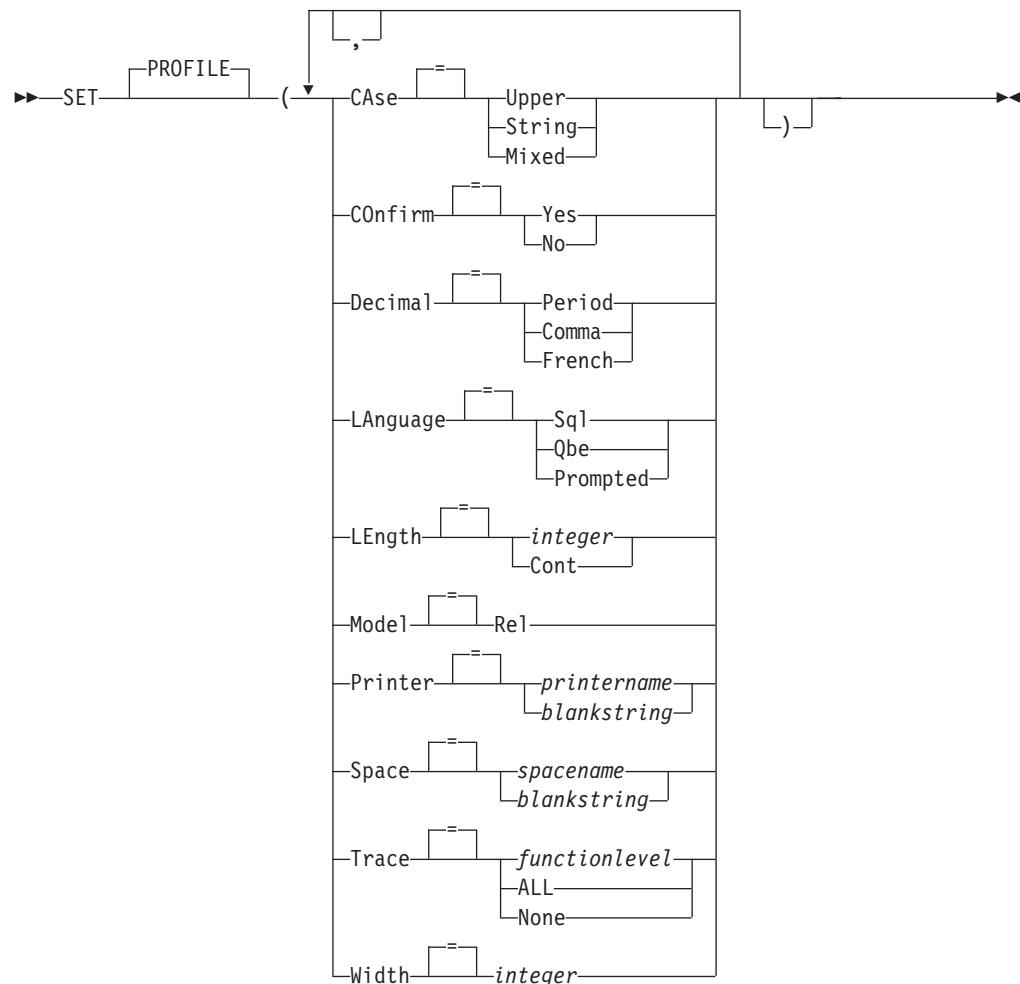
---

## SET PROFILE

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X	X	X

SET PROFILE コマンドは、QMF プロファイルの値を変更します。これらの値は QMF セッションの動作に影響を与えます。

### 一時記憶域の QMF プロファイルの変更



## 説明

**CASE** コマンドおよび他の入力を大文字に変換するかどうかを指定します。

### UPPER

すべての入力を大文字に変換します。

### STRING

次を除いて、入力を大文字に変換します。

- 単一引用符または二重引用符で囲まれた文字
- SQL または QBE の照会および手順の注釈
- 列見出し、ページ見出しと脚注、切れ目の見出し、または明細見出し
- 表編集機能に入力されたデータ
- ロジックを持つプロシージャー (REXX を使用する) 内のすべてのテキスト

### MIXED

入力を大文字に変換しません。入力はタイプされたとおりに使用されます。この値が使用されると、QBE 照会でのすべての演算子、すべての予約語、すべての QMF コマンドを大文字で入力しなければ

## SET PROFILE

ならなくなります。 QBE 照会の列名は、データベース内で小文字を使用して書かれていない限り、大文字で入力しなければなりません。

このオプションは、TSO で QMF CONNECT コマンドを使用し、使用しているサイトが RACF の大/小文字混合パスワードを使用している場合に、使用します。そうでなければ、QMF はパスワードを大文字に変換してしまい、CONNECT コマンドは失敗します。

### CONFIRM

CONFIRM オプションをサポートしている QMF コマンドに関して、確認プロンプトのときのデフォルト・アクションを指定します。このデフォルトは、当該コマンドが CONFIRM オプションを指定しない場合に適用されます。

確認プロンプトでは、取り消しができないコマンド処理を実行する前にその処理を取り消す機会を提供しています。取り消しができないコマンド処理には、データ・セット、データベースに関するものなどの変更、置き換え、除去などがあります。

- YES** 確認プロンプトは QMF セッションの省略時として使用できます。  
**NO** 確認プロンプトは QMF セッションの省略時として使用できません。

### DECIMAL

報告書でどのように 10 進数を区切るか指定してください。このオプションは 10 進数の編集コードで書式設定された数値の小数点と 3 桁ごとの区切りの特性の書式設定を制御します。

### PERIOD

小数点にはピリオド (.) を使用し、3 桁ごとの区切り記号にはコンマ (,) を使用します。

### COMMA

小数点にはコンマ (,) を使用し、3 桁ごとの区切り記号にはピリオド (.) を使用します。

### FRENCH

小数点にはコンマ (,) を使用し、3 桁ごとの区切り記号にはスペースを使用します。

### LANGUAGE

照会パネル用の省略時照会言語を指定します。

**SQL** 構造化照会言語

**QBE** 例示照会

**PROMPTED** 指示照会

### LENGTH

印刷ページの省略時長を指定します。長さの単位は行です。

### integer

ページの切れ目までの最大行数を指定します。数値は 1 から 999 までの整数です。

**CONT** ページの切れ目がない連続印刷を指定します。

#### MODEL

照会に使用するデータ・モデルを指定します。

**REL** 関係データ・モデル

#### PRINTER

QMF PRINT コマンドのデフォルトの出力宛先を指定します。

**printername**

プリント出力先を指定します。 GDDM プリンターのニックネームである必要があります。

**blankstring**

ファイルの宛先を指定します。この値は、单一引用符 (‘ ’) で囲まれた 0 から 8 文字のブランクで指定されなければなりません。

印刷出力の物理的な宛先は、ユーザーの QMF 環境と QMF 管理者による調整により決定されます。

- TSO の場合、出力先は、QMF ファイル DSQPRINT に割り振られたデータ・セットまたは装置です。
- CICS の場合、出力先は、PRINT コマンドの QUEUENAME オプションで指定された CICS キュー、またはそのデフォルトです。

#### SPACE

SAVE DATA コマンドによって作成された表を保管するための、データベース上のデフォルトのストレージ・スペースを指定します。

**spacename**

現行データベース・ロケーションの有効な記憶域構造名です。 DB スペース名、データベース名、表スペース名、またはデータベース名と表スペース名の組み合わせが可能です。

**blankstring**

ストレージ構造のデフォルトを指定します。これは、現在接続しているデータベースによって異なります。この値は、单一引用符 (‘ ’) で囲まれた 0 から 50 文字のブランクで指定されなければなりません。

#### TRACE

QMF 追跡機能をオンまたはオフにします。

**functionlevel**

個々の機能の追跡活動を使用可能に設定し、指定した各機能について、トレース内容のレベルを指定できます。

*functionlevel* を代替文字 (トレースする機能を示すコード) と番号 (指定した各機能のトレース内容のレベル) の並びで指定します。表 7 に、コードとレベルを示します。

表 7. 特定の QMF 機能のトレースのコード

コード	トレースする機能
A	アプリケーション
C	共通サービス

## SET PROFILE

表7. 特定の QMF 機能のトレースのコード (続き)

コード	トレースする機能
D	ドライバー・モジュール
E	フロントエンド・プロセッサー
F	フォーマッター
G	グラフィック変換プログラム
I	データベース・インターフェース
L	メッセージとコマンド
P	グラフィックス・プロッター
R	Radix partition tree
U	ユーザー出口

129 ページの表 7 に示したコードのあとに次の番号の 1 つを入力して、各機能に対するトレース内容のレベルを指定します。

- 0 = トレースなし
- 1 = 入力点、出口点、およびパラメーターの入出力に対してトレースする
- 2 = レベル 1 のデータに加えて、内部データもトレースする

たとえば、トレース・コード A2L2 では、アプリケーション、およびメッセージとコマンドに対し、共に最高レベルの内容でトレースします。

**ALL** すべての機能とすべてのレベルの追跡活動を使用可能にします。

**NONE** 追跡活動を使用できないようにします。

### WIDTH

印刷ページの省略時の幅を指定します。幅の単位は 1 バイト文字です。

### integer

1 行に印刷する最大文字数を指定します。数値は 22 から 999 までの整数です。

印刷する対象が報告書でない限り、指定した値より長い行は右端で切り捨てられます。この場合、FORM.OPTIONS パネルで行の折り返しを指定した場合を除き、指定した値より長い行は後続のページで書式化されます。

## 注

- SET PROFILE コマンドの結果、有効になった変更は、現行の QMF セッションの間だけ有効です。別の QMF セッションでも存続するように、これらの変更をプロファイルに保管するには、SET PROFILE を入力したあとで SAVE PROFILE コマンドを使用します。
- QMF プロファイルの内容を SET PROFILE を使用しないで変更するには、プロファイル・パネル上で SHOW PROFILE を入力してから、任意のオプションを変更してください。

- L についてのトレース機能レベルの指定により、次のいずれかがトレースされます。
  - メッセージ (L1)
  - メッセージと QMF コマンド (L2)

L トレース・コードにより、バッチ・モード・プロシージャー内のエラーを検出しやすくなります。

## 例

DECIMAL オプションにより作成されたさまざまな表記の例として、値 7654321 を小数部 2 桁でフォーマット設定する場合を次に示します。

<b>PERIOD</b>	7,654,321.00
<b>COMMA</b>	7.654.321,00
<b>FRENCH</b>	7 654 321,00

## SHOW

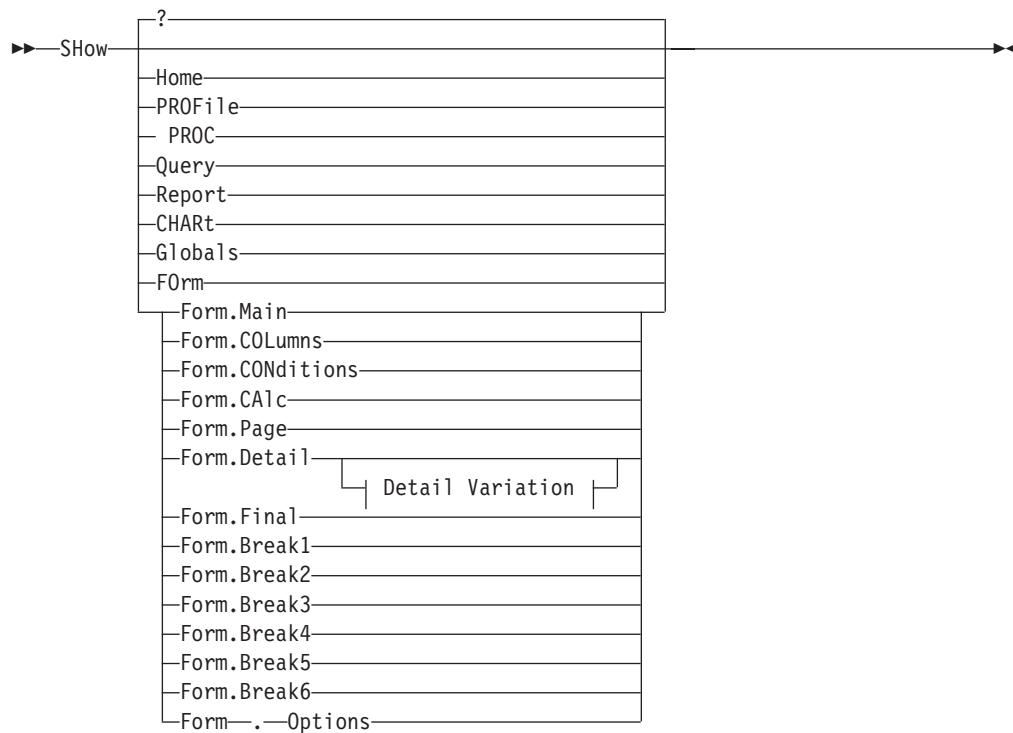
TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X	X	X

SHOW コマンドは下記の目的で使用します。

- オブジェクト・パネル相互の間をナビゲートします
- 大域変数リストを表示します
- 長すぎてパネルに入りきらないフィールドを表示します
- 指示照会の SQL 同等の表示
- データベース・オブジェクト・リストからコマンド・パネルを表示します。このパネルで、QMF コマンドまたは同義語を指定できます
- FORM.DETAIL パネルのバリエーションを表示します
- 現行オブジェクトのオブジェクト名と許可 ID を表示します

### オブジェクト・パネルの表示

## SHOW



### Detail Variation オプション:

```
( Variation [ currentvariation ] )
```

特定のパネルのフィールドの詳細情報の表示

```
►►—Show—Field————►►
```

指示照会と同等の SQL の表示

```
►►—Show—SQL————►►
```

表エディターの変更パネルの表示

```
►►—Show—CHANge————►►
```

表エディターの検索パネルの表示

```
►►—Show—SEArch————►►
```

コマンド入力パネルの表示

(1)

```
►►—Show—COnCommand————►►
```

注:

- 1 処理列のあるデータベース対象リスト・パネルからのみ有効です。

現行オブジェクトの名前または許可 ID の表示

►►—SHow—Name—►►

## 説明

**HOME**  
**PROFILE**  
**PROC**  
**QUERY**  
**REPORT**  
**CHART**  
**Globals**  
**FORM.MAIN**  
**FORM.COLUMNS**  
**FORM.CONDITIONS**  
**FORM.CALC**  
**FORM.PAGE**  
**FORM.DETAIL**  
**FORM.FINAL**  
**FORM.BREAK1**  
**FORM.BREAK2**  
**FORM.BREAK3**  
**FORM.BREAK4**  
**FORM.BREAK5**  
**FORM.BREAK6**  
**FORM.OPTIONS**  
**NAME**

指定した対象パネルが現行パネルとして表示されます。

### FORM

現行の書式対象パネルが現行パネルとして表示されます。これは前もって表示される様々な形式のいずれか 1 つになる場合があります。

### FIELD

基本パネル上のフィールドの追加情報を参照してください。このコマンド・オプションでは、下記の状況のパネルからファンクション・キーだけを使用します。

- 表エディターを使用中に、列の特性を表示する、あるいは長い文字フィールド用の入力域を拡張する
- 指示照会で比較値を入力するときに、入力域を拡張する
- 大域変数リスト・パネルで、大域変数値を変更または表示するときに、入力域を拡張する

### SQL

指示照会の SQL ステートメント同等の表示 SQL ステートメントを参照することができますが、変更はできません。

### CHANGE

### SEARCH

変更モード編集セッション中に指定した表編集機能パネルを参照してください。交互に使用して、2 つのパネル間で切り替えを行ないます。

## SHOW

このコマンド・オプションは表編集機能で提供される機能キー以外は使用できません。

### COMMAND

データベース対象リスト・パネルを使用するときは、 QMF コマンド入力パネルを参照してください。 QMF コマンド、またはコマンド同義語は、はじめに対象リストをそのまま、おのおの実行することができます。

このコマンド・オプションは、データベース・オブジェクト・リストで提供されるファンクション・キーからのみ使用できます。

### NAME

現在表示されているオブジェクトの完全名を表示します。オブジェクト名が切り捨てられた場合は、SHOW NAME コマンドを実行すると、ポップアップ・パネルにオブジェクトの完全名が表示されます。場合によっては、報告書オブジェクトは、報告書に関連付けられたオブジェクト名ではないこともあります。そのような場合、SHOW NAME コマンドは、ブランクの許可 ID とオブジェクト名を表示します。

## Detail Variation

### VARIATION

表示する詳細バリエーションを指定します。

このオプションを省略すると現行の詳細バリエーションが表示されます。

番号は FORM.DETAIL パネルに直接入れられるため、このオプションは SHOW コマンドのプロンプト・パネルには表示されません。

### integer

詳細バリエーションの数です。 1 から 99 の整数でなければなりません。

指定された詳細バリエーションがまだ作成されていなければ、番号は、既存の全詳細バリエーションに続く次の順序の番号になります。

## 注

- SHOW コマンドは DISPLAY コマンドと似ています。違いは次のとおりです。
  - SHOW コマンドは、QMF 一時記憶域にあるオブジェクト・パネル、大域変数、特定パネル部分を表示します。
  - DISPLAY コマンドは、データベース中のオブジェクトや現在 QMF 一時記憶域にあるオブジェクトを表示します。
- SHOW REPORT と SHOW CHART が失敗するのは、書式がデータと非互換のとき、または書式にエラーがあるときです。 QMF は、最初のエラーが起こった書式パネルを表示し、エラーを含む入力域を強調表示します。残っているエラーを見るには、まず、表示されている最初のエラーを訂正して、ENTER キーを押してください。
- 1. QMF SHOW コマンドのプロンプト・パネルを表示するには、次のコマンドの 1 つを入力します。

```
SHOW  
SHOW ?
```

2. 現在の QMF オブジェクトの名前を表示するには、次のようにします。  
SHOW NAME
3. QMF ホーム・パネルへ直接に移動するには次のようにします。  
SHOW HOME
4. FORM.DETAIL のバリエーション 2 を表示するには、次のようにします。  
SHOW FORM.DETAIL (VARIATION=2)
5. FORM.DETAIL の新しいバリエーションを作成するには、次のようにします。  
SHOW FORM.DETAIL (VARIATION=99)

## SORT

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X	X	X

SORT コマンドは、データベース・オブジェクト・リストに含まれている諸項目をソートします。このコマンドは、Sort ファンクション・キーを押すことしか出せません。ソートを要求すると、オブジェクト名の順序を選択できるパネルが表示されます。

デフォルトのソート順序を指定するために、グローバル・ソート変数を設定することができます。

►►—SOrt—►►

## SPECIFY

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X	X	X

SPECIFY コマンドは、指示照会と FORM.COLUMNS で使用できます。

### FORM.COLUMNS での指定



### 指示照会での指定



## 説明

FORM.COLUMNS パネルでは、SPECIFY は、書式中の列についての追加情報を与えたり、書式に新しい列を定義したりするパネルを表示します。

## SPECIFY

### ALIGNMENT

列番号、列見出し、見出し調整値、およびデータ調整値を表示します。変更できるのは、ヘッダー調整とデータ調整の値だけです。

### DEFINITION

列番号、列見出し、および列の定義（値が存在する場合）を表示します。変更できるのは定義値だけです。

指示照会での Specify (指定) キーは、「指定」パネルを表示します。そのパネルで、指示照会の、次の部分を指定できます。

**COLUMNS** 照会に入る列を指定します。

**DUPLICATES** 重複項目を表示するかどうかを指定します。

**ROWS** 戻すデータの行を指定できます。

**SORT** 行のソート方法を指定できます。

**TABLES** 照会で使用する表を指定できます。

## 注

- 列を定義するためには、列情報行にカーソルを合わせて SPECIFY を発行します。
  - 列調整では、SPECIFY コマンドを出したときのカーソル位置によって、調整パネルにどの列が表示されるかが決まります。
  - 列定義では、SPECIFY コマンドを出したときのカーソル位置によって、定義パネルにどの列が表示されるかが決まります。
- カーソルが列情報行にない場合は、パネルには最初の列が表示されます。
- 列定義のある「FORM.COLUMNS」パネルでは、次のことができます。
  - 他の列に基づいて列を定義する
  - 値の範囲に基づいて結果をグループ分けする
  - 個々のデータ値に対してユーザー関数を定義する
  - 部分列を表示する
  - 部分列に制御の切れ目を設定する
  - 1 つの列に複数の取り扱いを設ける
- パラメーターなしの SPECIFY は、選択用の項目リストを表示します。オブジェクトを指定した SPECIFY は、指定したオブジェクトのパネルを表示します。

---

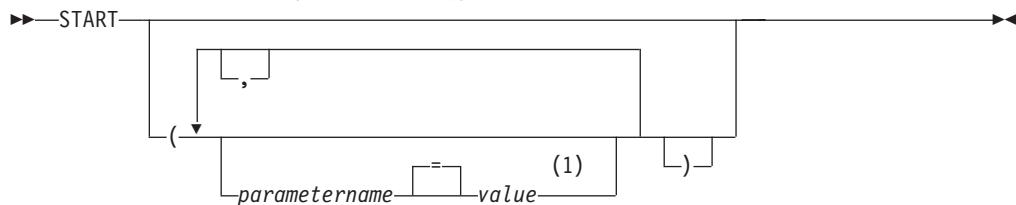
## START

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X	X	X

START コマンドは、新しい QMF セッションを開始します。コマンド構文は、使う言語ごとに異なります。ここでは、REXX で用いられる線形構文を示します。

REXX 以外の言語 (C、COBOL、FORTRAN、PL/I、またはアセンブラー) は、**START** コマンドの拡張構文を使用します。この構文の詳細は、「DB2 QMF アプリケーション開発の手引き」を参照してください。

#### REXX からの QMF セッションの開始



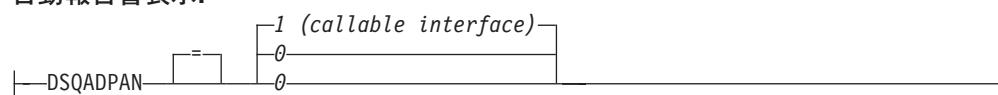
注:

- どのパラメーターでも、値 NULL を指定することにより、明示的にデフォルトを指示することができます。

### QMF プログラム・パラメーター

QMF プログラム・パラメーターの詳細は、「DB2 QMF アプリケーション開発の手引き」を参照してください。

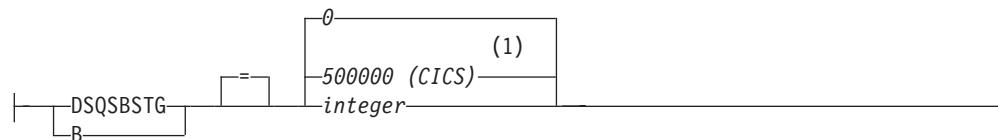
#### 自動報告書表示:



#### 主言語:



#### 報告書用記憶域の制限値:



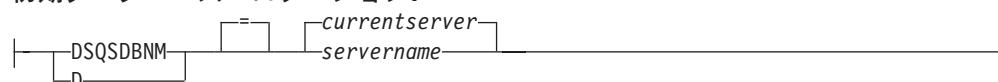
#### 開始パラメーターを渡すプログラム (TSO のみ):



#### DBCS サポート:



#### 初期データベース・ロケーション:

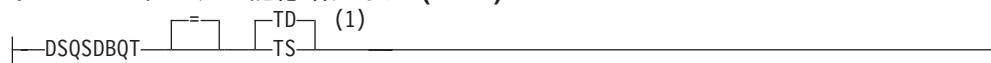


## START

トレース・データの記憶域名 (CICS):



トレース・データの記憶域タイプ (CICS):



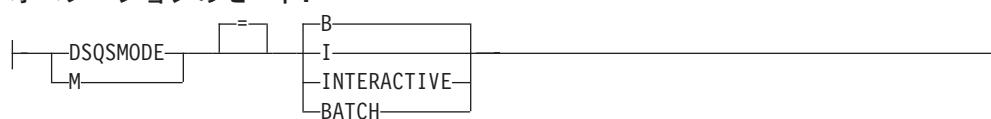
初期トレース:



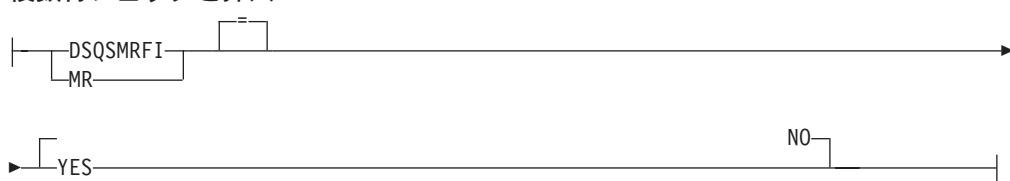
表示前にフェッチされる行:



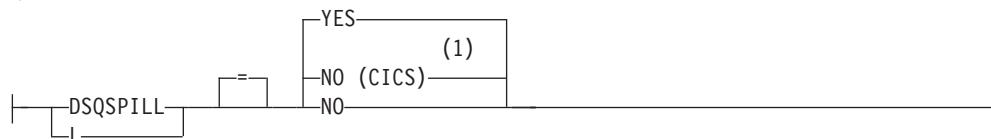
オペレーションのモード:



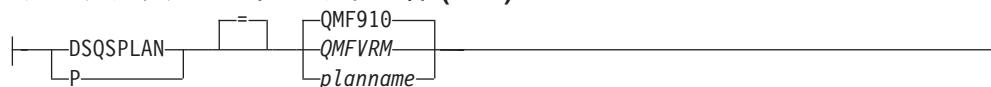
複数行フェッチと挿入:



予備ファイルを使用するかどうか:



QMF アプリケーション・プラン名 (TSO):



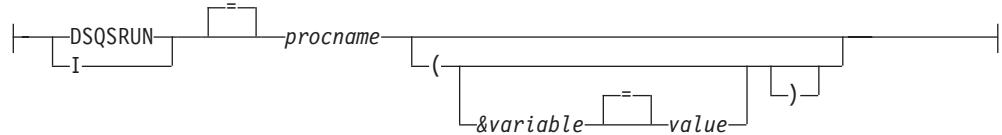
QMF プロファイル・キー (TSO):



### 予約ストレージ量 (TSO):



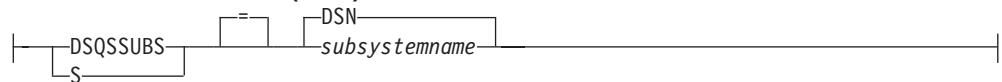
初期 QMF プロシージャ:



### 予備データ記憶域名 (CICS):



## DB2 サブシステム ID (TSO):



注：

- 1 徹底のために表示。QMF は CICS 環境で REXX をサポートしません。

## 説明

QMFvrm

QMF のレベルを区別するためのフォーマット。"vrm" は、バージョン、リリース、および修正レベルの ID の組み合わせを表します。

DSQS*termid*

CICS 環境の予備データ・キューのデフォルト名で、*termid* は 4 文字の CICS 端末 ID を表します。

---

## STATE

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X		

STATE コマンドは、選択された QMF 『状態』 変数の値を QMF 大域変数プールに保管します。STATE は、アプリケーション・サポート・コマンドであり、QMF コマンド・インターフェースからしか実行できません。STATE は、アプリケーションまたは CLIST から使用してください。

►—STATE—

STATE コマンドを発行すると、現行オブジェクトに関連付けられたデータベース・ロケーションに新しい変数が送信されます。

## STATE

状態変数の詳細は、293 ページの『付録 B. QMF 大域変数表』を参照してください。

---

## SWITCH

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X	X	X

SWITCH コマンドは、データベース・オブジェクト・リストにあるコメントを表示したり消したりします。また、指示照会で、表にあるコメントを表示したり消したりします。

►►—SWitch—コメント—►►

注:

SWITCH コマンドを出すと、

- パネルにコメントが表示されていれば、それが消えます。
- パネルにコメントが表示されていなければ、現行リスト・パネルがコメント行付きで再表示されます。各オブジェクトのコメントがあれば、パネルに表示されます。長すぎるコメントは、画面に合わせて末尾が切り捨てられます。

---

## TOP

TSO (ISPF 付き)	TSO (ISPF なし)	CICS
X	X	X

TOP コマンドは、照会、プロシージャー、大域変数リスト、およびスクロール可能な書式パネルで用いられ、その先頭までスクロールします。

►►—T0p—►►

注:

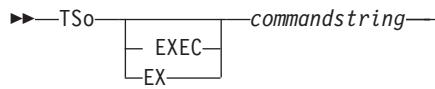
- TOP は BACKWARD MAX と同じです。
- 書式パネルで後書きテキストの先頭までスクロールするには、後書きテキストが配置されているパネル部分にカーソルを合わせてから、TOP コマンドを入力してください。

---

## TSO

TSO (ISPF 付き)	TSO (ISPF なし)
X	X

TSO コマンドを使用すると、QMF の使用を終了させずに、TSO 環境でコマンドを実行することができます。

**TSO コマンドの発行****説明****EXEC または EX**

*commandstring* の値が、TSO コマンドではなく、CLIST または REXX のプログラムのデータ・セット名であることを示します。

**commandstring**

TSO 環境での有効なコマンドまたは EXEC を構成する文字ストリング。

**注**

- TSO コマンドの後に入力されたものはすべて TSO に送られ、そこで解釈されます。
  - 実行が正常に終了すれば、TSO コマンドを入力した QMF の同じパネルに戻ります。
  - 実行が正常に終了しなければ、TSO からエラー・メッセージを受け取りますが、そのメッセージは QMF を実行できない場合と同じ内容になります。

**例**

1. ユーザー ID PEGGY5 に、TSO SEND コマンドでメッセージを送るには次のようにします。

TSO SEND 'I RECEIVED YOUR PROC2. THANK YOU.' USER(PEGGY5)

2. データ・セット KELLY1.EXEC にある REXX プログラム SAMPLE を実行するには、次のようにします。

TSO EXEC 'KELLY1.EXEC(SAMPLE)'



---

## 第 2 章 QMF 照会で使用される基本的な SQL ステートメントと関数

以下では、QMF 照会で使用される SQL キーワードのいくつかについて説明しています。これは、使用できる SQL キーワードをすべてリストしたものではありません。詳細は、IBM Publications Center ([www.ibm.com/shop/publications/order](http://www.ibm.com/shop/publications/order)) にある、使用しているデータベース・サーバーの SQL 参照情報を参照してください。

ワードの中には、データベース管理システムでキーワードになっているものがあります。多くの場合、これらのワードは二重引用符で囲まなければ、照会の中で表、視点、列、または索引の名前として使用することができません。

このトピックには、次の SQL キーワードおよび概念に関する情報が含まれています。

- 144 ページの『ADD』
- 144 ページの『ALL』
- 145 ページの『ALTER TABLE』
- 145 ページの『AND』
- 146 ページの『ANY』
- 146 ページの『AS』
- 147 ページの『AVG』
- 147 ページの『BETWEEN x AND y』
- 148 ページの『CALL』
- 151 ページの『COUNT』
- 152 ページの『CREATE SYNONYM』
- 152 ページの『CREATE TABLE』
- 154 ページの『CREATE VIEW』
- 155 ページの『DELETE』
- 156 ページの『DISTINCT』
- 157 ページの『DROP』
- 158 ページの『EXISTS』
- 158 ページの『GRANT』
- 159 ページの『GROUP BY』
- 161 ページの『HAVING』
- 163 ページの『IN』
- 163 ページの『INSERT』
- 165 ページの『IS』
- 165 ページの『LIKE』
- 167 ページの『MAX および MIN』
- 167 ページの『NOT』
- 168 ページの『NULL』

## SQL キーワードと関数

- 169 ページの『OR』
- 170 ページの『ORDER BY』
- 172 ページの『REVOKE』
- 173 ページの『SELECT』
- 175 ページの『SOME』
- 176 ページの『SUM』
- 177 ページの『UNION』
- 180 ページの『UPDATE』
- 181 ページの『WHERE』
- 184 ページの『計算結果』
- 185 ページの『@IF 関数』
- 187 ページの『SQL スカラー関数』
- 190 ページの『連結』

---

## ADD

表に列を追加できるのは、それがユーザー自身で作成した表か、特に列の追加を許可されている表に限られます。次の例では、表 PERS の記述に 1 つの列を追加します。

```
ALTER TABLE PERS  
ADD PHONENO SMALLINT
```

新しい列には、最初は NULL 値が入っています。 UPDATE ステートメントを使用すれば、新しい列に実際の値が与えられます。

DB2 (Linux、UNIX、および Windows 版) では、列を NOT NULL WITH DEFAULT と定義できますが、追加される列を NOT NULL とは定義できません。

現行ロケーションが DB2 Server (VSE または VM 版) であれば、NOT NULL WITH DEFAULT は無効です。

---

## ALL

副照会は、通常、値を 1 つしか返しません。しかし、照会には、1 組の値を返さることができます。 ALL のもとでは、返されてくる 1 組の値のそれが条件を満たしていなければなりません。

照会から単一の値ではなく 1 組の値が返されるようにするには、次の比較演算子の 1 つと一緒に ALL キーワードを使用します。

=    !=    >    >=    <    <=

記号 != は、<> (不等号) の代替記号です。これは、米国規格協会 (ANSI) の SQL 演算子です。(リモート・データ・アクセスでは、なるべく <> 記号を使用してください。)

次の照会は、平均給与が最高である部門をリストする報告書を作成します。 ALL キーワードが使用されていますから、主 SELECT ステートメントで選択される部門の平均給与は、他のあらゆる部門の平均給与以上でなければなりません。

```
SELECT DEPT, AVG(SALARY) FROM Q.STAFF
GROUP BY DEPT
HAVING AVG(SALARY) >= ALL
       (SELECT AVG(SALARY) FROM Q.STAFF
        GROUP BY DEPT)
```

等号 (=) 以外の演算子は、ALL キーワードと併用できます。副照会の結果に NULL が含まれているとき、ALL による比較結果がどうなるかは予測できません。

## ALTER TABLE

表を変更できるのは、それがユーザー自身で作成した表であるか、その表の変更を特に許可されている場合に限られます。 ALTER TABLE ステートメントは、既存の表のどれを変更するかを指定します。 ALTER TABLE に続いて、たとえば ADD ステートメントを用いて表の右端に新しい列を追加することができます。(144 ページの『ADD』に例があります。)

## AND

条件を AND や OR で結合すれば、複数の条件に基づく行選択ができます。2つの条件を AND で結ぶと、その両方の条件を満たす行だけが選択されます。たとえば、

次の照会で、

```
SELECT ID, NAME, YEARS, SALARY
FROM Q.STAFF
WHERE YEARS = 10 AND SALARY > 20000
```

以下の報告書が生成されます。

ID	NAME	YEARS	SALARY
50	HANES	10	20659.80
210	LU	10	20010.00

AND の使用結果を、169 ページの『OR』と比較してください。

AND と OR の両方を使用するときは、AND 条件と OR 条件の評価順序を括弧で指定してください。次の例を比べてください。

### 括弧付き

次の照会は、次の条件のうち少なくとも 1 つを満たす従業員を選択します。

- 職種が販売員で、歩合が \$1,200 を超える
- 勤続年数が 10 年を超える

```
WHERE (JOB='SALES' AND COMM > 1200) OR YEARS > 10
```

照会は、従業員 ID が 90、260、310、340 の情報を返します。

## AND

複数レベルの括弧を使用できます。条件は、代数式と 同様に、ネストされた条件の一番内側の条件から外側に向かって 評価されます。

### 括弧がない場合:

括弧を使用しないときは、AND で結合されたすべての条件が最初に評価されて、次に OR で結合された条件が評価されます。したがって、A、B、C を条件とすれば、次の 2 つの句は同じ結果になります。

```
A AND B OR C  
(A AND B) OR C
```

---

## ANY

副照会は、通常、値を 1 つしか返しません。しかし、照会には、1 組の値を返させることができます。照会から単一の値ではなく 1 組の値が返されるようにするために、次の比較演算子と一緒に ANY キーワードを使用できます。

```
=    ~=    >    >=    <    <=
```

ANY のもとでは、返される 1 組の値のなかに、条件を満たす値が少なくとも 1 つなければなりません。

副照会では、= ANY の代りに IN を使用できます。また、SOME は ANY の同義語です。

記号 ~= は、< > (不等号) の代替記号です。これは ANSI SQL 演算子です。 (リモート・データ・アクセスでは、なるべく < > 記号を使用してください。)

次の照会では、Eastern 地区で働く従業員のリストが得られます。まず、副照会で Eastern 地区にある部門の番号を求めます。次に、主照会でこれらの部門のいずれかで働く従業員を求めます。

次の照会では、Eastern 地区で働く従業員の名前と ID のリストが作成されます。

```
SELECT NAME, ID  
FROM Q.STAFF  
WHERE DEPT = ANY  
(SELECT DEPTNUMB FROM Q.ORG WHERE DIVISION='EASTERN')
```

この照会で ANY キーワードを使用したのは、Eastern 地区には複数の部門があるためです。 ANY の代わりに ALL を使用したとすれば、照会結果として空集合が返されてしまいます。 (Eastern 地区のすべての部門で働いている従業員はいません。)

---

## AS

SELECT ステートメントで AS 文節を使用して、照会内の結果列に名前を付けることや、結果列の名前を変更することができます。名前は修飾されていてはなりませんが、固有である必要はありません。

次の例は、DB2 (z/OS 版) で実行依頼される照会での AS 文節の使用を示しています。

```
SELECT NAME, SALARY*0.05 AS "RAISE"  
FROM Q.STAFF
```

AS 文節の指定がなく、しかも結果列が列名を受け継いでいる場合には、結果列名はその列の非修飾名になります。

## AVG

AVG は、数値データが含まれる列にのみ有効な列関数です。次の例では、SELECT ステートメントに複数の列関数が組み込まれています。部門 10 の従業員給与の合計額、最低額、平均額、最高額、従業員数 (COUNT) を計算して、表示します。

次の照会で、

```
SELECT SUM(SALARY), MIN(SALARY), AVG(SALARY),
       MAX(SALARY), COUNT(*)
  FROM Q.STAFF
 WHERE DEPT = 10
```

以下の報告書が生成されます。

SUM(SALARY)	MIN(SALARY)	AVG(SALARY)	MAX(SALARY)	COUNT(EXPRESSION)
83463.45	19260.25	20865.8625000000	22959.20	4

AVG 列関数は次のように記述します。

*AVG(expression)*

括弧は必ず付けてください。上記の構文で、*expression* は、ほとんどの場合は列名ですが、以下のものも指定できます。

- 少なくとも 1 つの列名を含む算術式。
- DISTINCT キーワードと、それに続く列名。

関数中の列名で、長ストリング列を参照してはなりません。また、列関数から導かれる列を参照してはなりません（視点の列は、関数から導かれることができます）。列関数を他の列関数内にネストさせることはできません。列関数による計算に、NULL 値は使用されません。

## BETWEEN x AND y

WHERE 文節で指定する列が上限と下限のあいだの値をもつとき、その列の各行からデータを取り出します。以上 ( $>=$ ) および以下 ( $<=$ ) 演算子を指定するときは、AND 条件の代わりに BETWEEN を使用してください。

指定する上限値と下限値は、範囲に含まれます。BETWEEN 条件の下限値（小さい方の値）を先に、上限値（大きい方の値）を後に指定してください。次の例は、給与が \$20,000 と \$21,000 の間の従業員を選択します。GRAHAM の給与は、丁度 \$21,000 です。

次の照会で、

```
SELECT ID, NAME, SALARY
  FROM Q.STAFF
 WHERE SALARY BETWEEN 20000 AND 21000
```

以下の報告書が生成されます。

## BETWEEN

ID	NAME	SALARY
50	HANES	20659.80
210	LU	20010.00
310	GRAHAM	21000.00

### 例

- 名前の並び順で HANES と MOLINARE の間に来る全員を選択します。

```
SELECT ID, NAME  
FROM Q.STAFF  
WHERE NAME BETWEEN 'HANES' AND 'MOLINARE'
```

- 勤続年数が 10 年から 12 年まで (両端を含む) の全員を選択します。

```
SELECT ID, NAME, YEARS  
FROM Q.STAFF  
WHERE YEARS BETWEEN 10 AND 12
```

- 給与が \$19,000 から \$21,000 の範囲にない従業員を選択します。

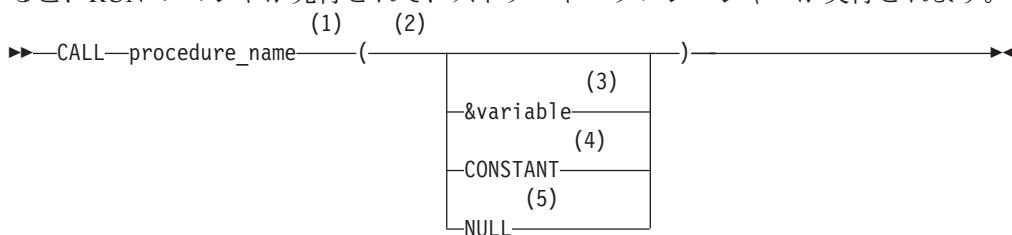
```
SELECT ID, NAME, YEARS  
FROM Q.STAFF  
WHERE SALARY NOT BETWEEN 19000 AND 21000
```

給与が \$19,000 未満か \$21,000 を超える従業員が報告書に記載されます。給与が \$19,000 から \$21,000 までの従業員は除外されます。

---

## CALL

QMF からストアード・プロシージャを実行するには、「SQL 照会」パネルから CALL ステートメントを発行する必要があります。CALL ステートメントを入力すると、RUN コマンドが発行されて、ストアード・プロシージャが実行されます。



### 注:

- 1 これは、呼び出すストアード・プロシージャを示します。
- 2 パラメーター値は、入力、出力、または入出力パラメーターです。
- 3 これは、ストアード・プロシージャに対する入力または出力として使用される QMF 置換変数を示します。
- 4 これは、ストアード・プロシージャに対する入力または出力として使用される定数を示します。
- 5 このパラメーターは、空白値です。ストアード・プロシージャの対応するパラメーターは、IN として定義されている必要があります、ストアード・プロシージャの記述に、NULL パラメーターを指定できるようにする必要があります。

## パラメーターの使用方法

CALL ステートメントのパラメーターは、次の方法で使用されます。

- 入力パラメーター (IN)

ストアード・プロシージャーに渡される入力パラメーターのリストを示します。

- 出力パラメーター (OUT)

ユーザー定義済みの QMF 置換変数の名前は、DB2 によってストアード・プロシージャーから戻される出力変数の値を受け取ります。ユーザーは、CALL ステートメントを使用する前に、QMF SET GLOBAL コマンドを使用して、これらの名前を設定しておく必要があります。

- 入出力パラメーター (INOUT)

入力または出力として使用でき、入力パラメーターまたは出力パラメーターとして振る舞うことができます。

## CALL ステートメントの使用のガイドライン

- QMF は、CALL ステートメント上にある 3 部構成の名前は処理しません。現行ロケーション (QMF が現在稼働中のロケーション) にあるストアード・プロシージャーのみ実行されます。3 パート名が入力されると、QMF はそれを受け入れますが、入力されたロケーションが現在のロケーションに一致しない場合は、エラー・メッセージが発行されます。
- QMF は、プロシージャー・ライブラリーまたは絶対パスを、DB2 (Linux、UNIX、および Windows 版) サーバーで有効なストアード・プロシージャー名の一部としてサポートしません。
- 許可検査は、データベースによって行われます。CALL ステートメントで指定されたストアード・プロシージャーを実行するには、現在の SQLID を許可する必要があります。
- 入力されたパラメーター数が、ストアード・プロシージャーの DB2 (Linux、UNIX、および Windows 版) カタログ定義で指定された数と同じである場合のみ、DB2 (Linux、UNIX、および Windows 版) はそのストアード・プロシージャーを実行します。一致していないければ、エラー・メッセージが表示されます。
- 出力を表示する場合は、QMF 大域変数を使用して、ストアード・プロシージャーの出力パラメーターを指定する必要があります。これにより、SHOW GLOBALS コマンドを使用して、出力パラメーターを表示できます。
- QMF 置換変数の最大サイズは 32K です。
- 「SQL 照会」パネルからは最大で 10 個の QMF 大域変数を入力することができます。
- 値がストアード・プロシージャーの出力パラメーターにコピーされる QMF 大域変数には、特別な初期化が必要です。数値データ型を持つ出力パラメーターは、0 に初期化する必要があります。データ型が CHAR の出力パラメーターは、ブランクまたは空白に初期化する必要があります。
- データ型が、DATE、TIME、または TIMESTAMP で定義されているパラメーターは、その値を单一引用符で囲む必要があります。QMF はこれらのデータ型を文字ストリングとして処理します。

## CALL

- BINARY、VARBINARY、VARGRAPHIC、GRAPHIC、LONG VARGRAPHIC、CLOB、BLOB、DBCLOB、ROWID、および DECFLOAT のタイプのデータは、CALL ステートメント上のパラメーターに渡すことはできません。
- 結果セットを戻すストアード・プロシージャーが実行されている場合は、QMF は戻された最初の 32 個の結果セットをサポートします。大域変数 DSQEC\_SP\_RS\_NUM を設定して、1 を選択してください。1 がデフォルトの設定値です。結果セットを無視するには、この大域変数をゼロに設定します。

## 長い ID を伴う CALL ステートメントの作成方法

「QMF SQL 照会」パネルでは、SQL 照会の 1 行は、79 バイトに制限されています。「SQL 照会」パネルで入力される CALL ステートメントで、複数行にわたる ID は、区切り文字で区切られた ID でなければなりません。以下に、長い CALL ステートメントを作成する方法を示した例をいくつか示します。

- 複数行にわたる区切り ID としての長いパラメーター:

```
CALL USERID.PROC ('THIS IS THE FIRST PARM', 4, 1954, "THIS IS ANOTHER  
PARM THAT WILL SPAN TWO LINES ON THIS PANEL", 14, 99)
```

- 複数行にわたる区切り ID としての長いストアード・プロシージャー名:

```
CALL USERID. 'THIS_IS_A REALLY_LONG_STORED_PROCEDURE_NAME_THAT_EXCEEDS_  
MORE_THAN_ONE_LINE_ON_THE_QUERY_PANEL' ('PARM1', ' ', 0, 'PARM4')
```

- ID 間で行を区切る:

```
CALL USERID.PROC ('THIS IS THE FIRST PARM', 4, 1964,  
'THIS IS ANOTHER PARM THAT WILL NOW FIT ON THIS LINE',  
14, 99)
```

```
CALL USERID.PROC ('THIS IS THE FIRST PARM', 333333,  
123456789012345678901234567890, 200305,  
'THIS IS THE LAST PARM')
```

- テキストが 3 行以上にわたる場合は、次のように区切り文字 (この場合は、二重引用符) を使用します。

```
CALL USERID.PROC ("THIS IS THE FIRST PARM AND IT WILL NOT ONLY EXTEND  
PAST THE FIRST LINE, IT WILL ALSO EXTEND BEYOND THE SECOND LINE BECAUSE  
THERE ARE TOO MANY WORDS TO FIT IN TWO LINES ALONE").
```

## 使用する結果セットを示す大域変数

次の QMF 大域変数は、ストアード・プロシージャーから戻された結果セットのどれを使用して、報告書を作成する必要があるのかを示します。

- 名前: DSQEC\_SP\_RS\_NUM
- 長さ: 31
- 値:
  - 0 - 結果セットを無視します
  - 1 - 最初の結果セットを戻します
  - 2 - 2 番目の結果セットを戻します
  - n - n 番目の結果セットを戻します (n の最大値は 32 です)

## QMF 書式の使用

RUN コマンドでフォームを指定しなければ、戻された結果セットに基づいて、デフォルトのフォームが作成されます。データを選択する照会を実行する場合は、RUN

コマンドに FORM を指定できます。結果セットを戻すストアード・プロシージャには、これと同じサポートが提供されています。ストアード・プロシージャにより複数の結果セットが戻される場合は、大域変数 DSQEC\_SP\_RS\_NUM で結果セットのいずれかの番号を指定すれば、その結果セットを表示することができます。結果セットの残りは無視されます。この場合、RUN コマンドで指定されたフォームには、結果セットで戻されるデータに一致するデータ定義がなければなりません。FORM 定義が、結果セットで戻されるデータに一致しなければ、エラー・メッセージが発行されます。その場合は、新規のフォームをロードするか、既存のフォームを変更するか、フォームをデフォルトのフォームにリセットして、ストアード・プロシージャを再実行することができます。

## COUNT

COUNT 関数は、非ヌル値の個数しかカウントしません。したがって、COUNT 関数の結果のデータ・タイプは、必ず NOT NULL 属性になります。COUNT には 2 つの使用方法があります。

- DISTINCT キーワードを伴う COUNT。次の 2 つの形式があります。

- COUNT (DISTINCT *colname* )

指定された列にヌル以外の値が入っている行を、カウントします。重複したものはカウントしません。

この形式では、列名を使用しなければなりません。式は使用できません。詳細については、156 ページの『DISTINCT』を参照してください。

```
SELECT COUNT(DISTINCT DIVISION)
      FROM Q.ORG
```

結果は 4 になります。

- COUNT(DISTINCT *expression* )

グループ内の列の他と異なる値を戻します。たとえば、次の照会では、Q.APPLICANT 表の中の求職者の他と異なる教育レベルと、求職者の平均教育年数を戻します。

```
SELECT COUNT(DISTINCT EDLEVEL), AVG(EDLEVEL)
      FROM Q.APPLICANT
```

DISTINCT キーワードの詳細は、156 ページの『DISTINCT』を参照してください。

- COUNT(\*) — 列の値に関係なく、戻ってきた行をカウントします。この形式では、列名を使用しません。

```
SELECT SUM(SALARY), MIN(SALARY), AVG(SALARY),
      MAX(SALARY), COUNT(*)
      FROM Q.STAFF WHERE DEPT = 10
```

次の例では、SELECT ステートメントに複数の列関数が含まれています。部門 10 の従業員給与の合計額、最低額、平均額、最高額、従業員数 (COUNT) を計算して、表示します。次の報告書を作成します。

SUM(SALARY)	MIN(SALARY)	AVG(SALARY)	MAX(SALARY)
-----	-----	-----	-----
83463.45	19260.25	20865.8625000000	22959.20

## CREATE SYNONYM

CREATE SYNONYM ステートメントは、表または視点の代替名を定義します。こうすることによって、他ユーザーが所有する表を、完全修飾名を使わずに参照できるようになります。自分自身の表および視点のために同義語を作成することもできます。同義語は、ドロップされるまで定義されたまま残ります。

次の例は、表 Q.APPLICANT の新しい名前を作成します。

```
CREATE SYNONYM APPLS FOR Q.APPLICANT
```

このステートメントを実行したあとは、前に Q.APPLICANT を使用していたすべてのコマンドおよびステートメントの中で、APPLS を使用できます。

同義語は、完全修飾された表名（区切りのピリオドを除いて 26 文字まで）より短くなければ定義する意味がありません。また、他ユーザーの作成した表を使用する場合、同義語は照会を保護する貴重な手段となります。たとえば、ユーザー BDJ1385L が Q.APPLICANT 表を除去して、再作成したとします。使用している照会は、すべて APPLS という同義語を使って書いてあります。最初のステップで次のコマンドを使用して、同義語を除去します。

```
DROP SYNONYM APPLS
```

そして、次の変更を行います。

```
CREATE SYNONYM APPLS FOR BDJ1385L.APPLICANT
```

同義語を使った照会を共用する場合、共用する他ユーザーが同じ同義語を作成するまで、照会の共用はできません。許可 ID のもとで定義した同義語は共用できません。しかし、同じ意味をもつ同じ同義語を他ユーザーにも定義してもらえば、問題は解決します。

使用しているサイトが DBCS データを使用する場合、データベースが表名での 2 バイト文字使用を特にサポートしていない限り、内部的に二重引用符として表現される 2 バイト文字を含む同義語は作成しないでください。2 バイト文字を含む名前の書き方の詳細については、272 ページの『2 バイト文字を含む名前』を参照してください。

## CREATE TABLE

CREATE TABLE ステートメントは、表を定義します。ステートメントには、表の名前およびその列の名前と属性を指定します。作成した表の使用許可を他人に与えたり、取り消したりすることができます。158 ページの『GRANT』および 172 ページの『REVOKE』を参照してください。

CREATE TABLE ステートメントの構文は、次のとおりです。

```
CREATE TABLE tablename (column1 type1 NOT NULL,  
column2 type2 . . .)  
IN space-name
```

*tablename*

表に割り当てる名前。

使用しているサイトが DBCS データを使用する場合、データベースが表名での 2 バイト文字使用を特にサポートしていない限り、内部的に二重引用符として表現される 2 バイト文字を表名に含めてはなりません。(詳細については、272 ページの『2 バイト文字を含む名前』を参照してください。)

**列 1 タイプ 1**

第 1 列に割り当てる名前とそのデータ・タイプ。データ・タイプが CHAR、VARCHAR、GRAPHIC、VARGRAPHIC、DECIMAL のいずれかであれば、データ・エレメントの最大長を括弧つきで指定してください。DECIMAL の場合は、小数部の桁数も指定しなければなりません。

**列 2 タイプ 2**

第 2 列に割り当てる名前とそのデータ・タイプ。

**NOT NULL**

定義する列に対して任意に指定できます。表定義では NOT NULL を指定しておくと、表中のその列に何かの値を入れないかぎり、エラー・メッセージが出ます。NOT NULL を省略すれば、列値がヌル値であっても認められます。

**IN *space-name***

表を作成する表スペースまたは DB スペース。使用しているサイトがデフォルトの解釈ではスペースを割り振るようになっている場合、この文節は不要です。

SAVE DATA または IMPORT TABLE コマンドで QMF に表を作成させたとき、どの *space-name* が使われたかを知るには、QMF コマンドの DISPLAY PROFILE を発行して、SPACE オプションの値を見てください。

次の CREATE ステートメントは、PERS という表を定義しています。PERS を構成する各列は Q.STAFF と同じ特性を持っていますが、データを含んでいません。

```
CREATE TABLE PERS
  (ID SMALLINT NOT NULL,
   NAME VARCHAR(9),
   DEPT SMALLINT,
   JOB CHAR(5),
   YEARS SMALLINT,
   SALARY DECIMAL(7,2),
   COMM DECIMAL(7,2))
  IN space-name
```

**ID** 従業員番号は短精度整数です。ヌルは指定できません。

**NAME** 名前の最大長は 9 文字です。

**DEPT** 部門番号列のデータ・タイプは短精度整数です。

**JOB** 職種は 5 文字です。

**YEARS** 勤続年数は短精度整数です。

**SALARY** 7 桁の数値で、小数部は 2 桁です。

**COMM** 7 桁の数値で、小数部は 2 桁です。(最後の括弧を忘れないでください。)

## CREATE TABLE

NOT NULL は、CREATE TABLE ステートメントのどの列セットにも使用できます。上の例では、ID 列に使用されています。したがって、PERS に入力されるすべての行は、少なくとも従業員番号を含んでいなければなりません。

次のステートメントは、Q.APPLICANT という表を定義します。

```
CREATE TABLE APPLICANT
(TEMPID  SMALLINT NOT NULL,
NAME      VARCHAR(9),
ADDRESS    VARCHAR(17),
EDLEVEL   SMALLINT,
COMMENTS  VARCHAR(29))
IN space-name
```

次のステートメントは、Q.INTERVIEW という表を定義します。

```
CREATE TABLE INTERVIEW
(TEMPID   SMALLINT,
INTDATE   DATE,
STARTTIME TIME,
ENDTIME   TIME,
MANAGER   SMALLINT,
DISP      VARCHAR(6),
LASTNAME  VARCHAR(9),
FIRSTNAME VARCHAR(9))
IN space-name
```

表を定義するだけでは、データは入りません。表にデータを入力する方法については、163 ページの『INSERT』を参照してください。

---

## CREATE VIEW

視点は、既存の表から選択されたデータが含まれている論理表です。視点では、列の改名と再配置、不要な列または行の除去、式による列定義、結果のグループ分け、複数の表の結合ができます。視点により、1 つ以上の表の一部に存在するデータを表示することが可能になります。視点には、データは実際には存在しません。

ORDER BY 文節を含まない SELECT ステートメントは、どれも視点の基礎として使用できます。選択された列と行が、視点の列と行になります。次の例では、Q.STAFF 表からの NAME、ID、および JOB の各列が、D42 視点の列になります。D42 での列名は、LAST NAME、EMP.ID、JOB です。

```
CREATE VIEW D42
("LAST NAME", "EMP. ID", JOB)
AS SELECT NAME, ID, JOB
FROM Q.STAFF
WHERE DEPT = 42
```

コマンド DISPLAY TABLE D42 を発行すると、次の視点が表示されます。

LAST NAME	EMP. ID	JOB
KOONITZ	90	SALES
PLOTZ	100	MGR
YAMAGUCHI	130	CLERK
SCOUTTEN	200	CLERK

視点を使用する主な理由は、次の 2 つです。

- 照会が簡単に書けます。

- データへのアクセスを防止できます。上の例の視点 D42 を使用するユーザーは、給与データを見ることができません。

視点は、表と同様、名前で扱います。表であるかのように見なし、表の場合と同様の SELECT ステートメントでデータを選択できます。たとえば、D42 視点について次の照会を実行できます。

```
SELECT * FROM D42
WHERE JOB='CLERK'
```

若干の制約はありますが、視点でも行の挿入、更新、削除ができます。視点に変更を加えると、その視点を支えている表にも変更が加えられます。

視点ではできないこともあります。

- 視点に次のものが含まれていると、視点を使用して、データの挿入、更新、または削除はできません。
  - 複数の表のデータ。
  - 列関数で定義されている列。たとえば、SUM(SALARY)。
  - DISTINCT キーワードや GROUP BY キーワードで選択されたデータ。
- 式 (SALARY/12 など) で定義された列を含んでいる視点では、データの更新と挿入ができません。ただし、この場合、データは削除できます。
- 視点を作成するときは、UNION キーワードは使用できません。
- GROUP BY 文節を使用して作成された視点を別の表または視点に結合することはできません。

## **DELETE**

行を削除できるのは、それがユーザー自身で作成した表であるか、行の削除をとくに許可されている場合に限られます。行単位で表から情報を削除できます。行内の個々のフィールド、または情報のすべての列を削除することはできません。

DELETE ステートメントは、次の 2 つの部分で構成されます。

### **DELETE FROM**

行を削除する表の名前。

### **WHERE**

削除される行を決める基準。

WHERE 文節を指定せずに DELETE を入力すると、表中のすべての行が削除されます。表は残りますが、行はなくなります。

次のステートメントは、PERS 表から従業員番号 140 を削除します。

```
DELETE FROM PERS
WHERE ID = 140
```

同名の従業員が複数いる可能性があるため、この例では従業員名でなく ID を使用して、思ぬ行まで削除される危険を防いでいます。

削除する行を示す条件を含めることで、1 つの DELETE ステートメントで複数の行を削除できます。次の例では、部門 10 の全員を削除しています。

## **DELETE**

```
DELETE FROM PERS  
WHERE DEPT = 10
```

データの削除に必要な許可の詳細は、158ページの『GRANT』を参照してください。

---

## **DISTINCT**

SQL ステートメントで列名の前に DISTINCT キーワードを指定しておくと、重複行を選択しないですみます。次の例は、Q.ORG 表に存在する固有の地区のみリストします。

次の照会で、

```
SELECT DISTINCT DIVISION  
FROM Q.ORG
```

以下の報告書が生成されます。

```
DIVISION  
-----  
CORPORATE  
EASTERN  
MIDWEST  
WESTERN
```

この結果を次の例と比べてください。

次の照会で、

```
SELECT DIVISION  
FROM Q.ORG
```

以下の報告書が生成されます。

```
DIVISION  
-----  
WESTERN  
WESTERN  
CORPORATE  
EASTERN  
EASTERN  
EASTERN  
MIDWEST  
MIDWEST
```

DISTINCT は、データの他と異なる組み合わせも選択できます。たとえば、

```
SELECT DISTINCT DEPT, JOB  
FROM Q.STAFF  
ORDER BY DEPT
```

この例から生成される報告書は、各部門で提示されるジョブを示しています。

DISTINCT を使用しているときは、これらのプロパティーを思い出してください。

- DISTINCT は SELECT の後に書かれます。
- DISTINCT は、最初の列名の前に書かれます。列名との間にはコンマは不要です。
- DISTINCT は、選択されるすべての列に適用されます。

DISTINCT は、COUNT と一緒に使用できます（詳細は、151 ページの『COUNT』を参照してください）。

使用するグループ内の各列の他と異なる値だけが必要であれば、他の列関数と一緒に DISTINCT を使用してください。たとえば、AVG(DISTINCT PRICE) とすれば、列中の重複価格は無視され、別値の価格を一度ずつ含むリストの平均値が求められます。AVG(PRICE) とすれば、重複価格の有無は無視され、列中のすべての価格の平均値が求められます。

### 例

- YEARS に現れる他と異なる値をリストするには、次のような照会を使用します。

```
SELECT DISTINCT YEARS
FROM Q.STAFF
ORDER BY YEARS
```

- 勤続 10 年以上の従業員が最低 1 人いる部門の部門番号をリストするには、次のような照会を使用します。

```
SELECT DISTINCT DEPT
FROM Q.STAFF
WHERE YEARS >= 10
```

## DROP

DROP ステートメントは、表、視点、同義語、別名、その他のオブジェクト（索引など）をデータベースから消去します。データベースから表や視点を除去するには、その許可が必要です。同義語を除去するには、その同義語の所有者でなければなりません。別名を除去するには、その所有者であるか、または SYSADM あるいは SYSCTRL 権限を持っていなければなりません。

DROP ステートメントの構文は、次のとおりです。

`DROP object object-name`

オブジェクト (*object*)

TABLE、VIEW、SYNONYM、または ALIAS のいずれかです。

*object-name* データベースで通用しているそのオブジェクトの名前です。

表 8 に、DROP ステートメントの使用例と結果をいくつか示します。

表 8. DROP ステートメントの使用例

次のステートメントは、	オブジェクトの消去
<code>DROP TABLE PERS</code>	PERS という表
<code>DROP VIEW D42</code>	D42 という視点
<code>DROP SYNONYM APPLS</code>	同義語 APPLS
<code>DROP ALIAS PROJECTIONS1</code>	PROJECTIONS1 という別名

**重要:** DROP TABLE の使用には、十分な注意が必要です。表を除去すると、その中のデータが壊れ、その表に支えられている視点もすべて壊れます。表を廃棄したあと再作成した場合、その表または表の視点に以前に付与された特権は、すべて再度付与される必要があります。

## DROP

コマンド ERASE TABLE *name* を発行することは、次のいずれかのコマンドを実行するのと同じです。

```
DROP TABLE name
DROP VIEW name
DROP SYNONYM name
DROP ALIAS name
```

DROP VIEW は、視点を支えている表には影響せず、データベース中の表も壊しません。除去された視点は、簡単に再作成できます。ただし、DROP VIEW は、視点に付与されていたすべての特権を取り消します。

DROP SYNONYM は、指定された同義語を同義語辞書から除去します。除去された同義語は、データベース中のなにものも参照しません。その同義語でアクセスされていた表や視点には、なんの影響もありません。たとえば、Q.APPLICANT の同義語表に APPLS があったとして、ステートメント DROP SYNONYM APPLS を実行しても、Q.APPLICANT には影響ありません。

---

## EXISTS

EXISTS ステートメントは、下記の照会の副照会のように、与えられた条件を満たす行が存在するかどうかを判断します。

```
SELECT ID, NAME, DEPT
  FROM Q.STAFF CORRVAR
 WHERE EXISTS
   (SELECT * FROM Q.ORG WHERE MANAGER = CORRVAR.ID)
```

条件によって値を選択する方法はほかにもあります。 163 ページの『IN』を参照してください。

---

## GRANT

GRANT ステートメントは、ある表に 1 つ以上の操作を行なう権限をユーザーに与えます。所有していない表への行の挿入、更新、削除、変更、または選択には、許可が必要です。この許可を与えることのできるのは、表の作成者か、作成者から許可されているユーザーに限られます。（172 ページの『REVOKE』も参照してください。）

GRANT ステートメントの構文は、次のとおりです。

```
GRANT operation-list ON tablename
  TO user-list WITH GRANT OPTION
```

*operation-list*

ALTER、DELETE、INSERT、SELECT、UPDATE (*column-list*) の内、1 つ以上の特権。複数指定する場合は、コンマで区切れます。

ALL は、認可者に許可されているすべての操作を行うための許可を付与します。

*tablename*

許可が付与される表または視点を指定します。

*user-list*

各ユーザー ID をコンマで区切ってリストします。*user-list* の代わりに PUBLIC を指定すれば、すべてのユーザーに許可を付与できます。

**WITH GRANT OPTION 文節**

別のユーザーに、他のユーザーに同じ特権を付与できる GRANT キーワードの使用を許可します。この文節はオプションです。

次のステートメントは、表 PERS に対する SELECT 照会をすべてのユーザーに許可します。

```
GRANT SELECT ON PERS TO PUBLIC
```

次のステートメントは、ユーザー HSAM4419 に PERS に対する行の挿入と削除を許可します。

```
GRANT INSERT, DELETE ON PERS TO HSAM4419
```

次のステートメントは、PERS 更新の許可を SMITH に与え、さらに SMITH が他のユーザーに同じ許可を与えることを認めます。

```
GRANT UPDATE ON PERS TO SMITH WITH GRANT OPTION
```

許可の付与の詳細は、IBM Publications Center ([www.ibm.com/shop/publications/order](http://www.ibm.com/shop/publications/order)) にある、使用しているデータベースの SQL 参照情報を参照してください。

**GROUP BY**

GROUP BY ステートメントには、グループ分けの基準となる列を指定します。データは、その列の値に基づいてグループ分けされ、グループごとに 1 行ずつの結果行が返されます。GROUP BY で複数の列名を指定できます (指定する列名をコンマで区切ります)。照会で GROUP BY を使用するときは、必ず FROM と WHERE の後、HAVING と ORDER BY の前に指定してください。

選択された列で、集約のオブジェクトになっていないものは、すべて GROUP BY 文節に含まれていなければなりません。

GROUP BY は結果をグループごとにまとめますが、必ずしもグループ同士の並び順を規定しません。その目的には ORDER BY ステートメントを使用してください。表から複数の行を検索するとき、GROUP BY、HAVING、および ORDER BY の各文節は、それぞれ次のことを指定します。

- 行グループの分け方 (GROUP BY)
- 行グループの条件 (HAVING)
- 行が返されてくる順序 (ORDER BY)

たとえば、次の照会は、各部門の平均給与を選択します。

```
SELECT DEPT, AVG(SALARY)
  FROM Q.STAFF
 GROUP BY DEPT
```

この照会は、次の報告書を作成します。

DEPT	AVG(SALARY)
10	20865.8625000000
15	15482.3325000000
20	16071.5250000000
38	15457.1100000000

## GROUP BY

42	14592.2625000000
51	17218.1600000000
66	17215.2400000000
84	16536.7500000000

上の例では、表が GROUP BY で部門番号に基づく行グループに分けられ、グループごとに 1 行ずつの結果が返されます。組み込み関数なしでも DEPT 列を選択できるのは、それが GROUP BY で指定されており、各グループ内ではすべてのメンバーの DEPT 列が同じ値をもっているからです。SELECT 文節に含まれるすべての列名は、組み込み関数で使われるか、GROUP BY 文節で指定されていなければなりません。たとえば、DEPT が GROUP BY 文節に含まれていなかったとしたら、平均給与のリストにはほとんど意味がありません。

### 正しい例:

```
SELECT DEPT, AVG(SALARY), JOB  
FROM Q.STAFF  
GROUP BY DEPT, JOB
```

### 誤った例:

```
SELECT DEPT, AVG(SALARY), JOB  
FROM Q.STAFF  
GROUP BY DEPT
```

GROUP BY は一般に、GROUP BY 文節に指定された列の異なった値ごとに、1 行ずつ結果を報告書に返します。GROUP BY 文節に複数の列名が指定されているときは、いずれかの列の値が変わるとたびに報告書に新しい行が作られます。しかし、列にヌル値が含まれているときは、すべてのヌル値が、それぞれ単一のメンバーからなる別個のグループとして扱われます。

SQL で GROUP BY を使用するのは、書式で GROUP 取扱コードを使用する(260 ページの『GROUP 取扱コード』を参照)のに代わる方法です。GROUP BY は、書式によるグループ分け機能を拡張するもので、書式だけではできないデータの条件付き選択も可能にします。たとえば、いま、部門ごとの給与合計を求め、その最低額、最高額、平均額を見たいとします。

1. 次の照会を作成し、実行します。

```
SELECT DEPT, SUM(SALARY), SUM(SALARY), SUM(SALARY)  
FROM Q.STAFF  
GROUP BY DEPT
```

2. 書式で次の取扱コードを使用します。

NUM	COLUMN HEADING	USAGE
1	DEPT	
2	SUM(SALARY)	MINIMUM
3	SUM(SALARY)1	AVERAGE
4	SUM(SALARY)2	MAXIMUM

報告書は 4 つの欄からなり、最後の 3 つの欄はほとんど同内容です。どの欄にも各部門の給与合計が示されます。ただし、最終行には、部門給与合計のうちの最小値、平均値、最大値が示されます。

### 例

- 管理者を除いて、部門ごとに、職種別の最高給与と最低給与をリストするには、次のような照会を使用します。

```
SELECT DEPT, JOB, MIN(SALARY), MAX(SALARY)
FROM Q.STAFF
WHERE JOB < > 'MGR'
GROUP BY DEPT, JOB
```

- 勤続年数ごとに、従業員の人数と平均給与をリストするには、次のような照会を使用します。

```
SELECT YEARS, COUNT(*), AVG(SALARY)
FROM Q.STAFF
GROUP BY YEARS
```

HAVING キーワードには、必ずグループ分けされたデータを指定しなければなりません。HAVING ステートメントと GROUP BY ステートメントをともに使用するときは、GROUP BY ステートメントを前、HAVING ステートメントを後に置かなければなりません。

- 平均給与が \$12,000 を超える各部門について、管理者を除き、その部門の給与の最低額、最高額、平均額をリストするには、次のような照会を使用します。

```
SELECT DEPT, MIN(SALARY), MAX(SALARY), AVG(SALARY)
FROM Q.STAFF
WHERE JOB < > 'MGR'
GROUP BY DEPT
HAVING AVG(SALARY) > 12000
```

- 勤続年数ごとに、従業員が 2 名以下のグループを除いて、従業員の人数と平均給与をリストするには、次のような照会を使用します。

```
SELECT YEARS, COUNT(*), AVG(SALARY)
FROM Q.STAFF
GROUP BY YEARS
HAVING COUNT(*) > 2
```

## HAVING

HAVING 文節は、GROUP BY 文節で得た結果のフィルターとして働きます。次の例では、HAVING COUNT(\*) > 4 の文節によって、4 人以下のメンバーの部門が最終結果から除かれます。

次の照会で、

```
SELECT DEPT, AVG(SALARY)
FROM Q.STAFF
GROUP BY DEPT
HAVING COUNT(*) > 4
```

以下の報告書が生成されます。

DEPT	AVG(SALARY)
38	15457.110000000
51	17218.160000000
66	17215.240000000

WHERE と HAVING は、どちらも報告書から不要なデータを除去します。

WHERE 条件は列選択に用いられ、個々の行を含めるかどうかを決定します。

HAVING 条件は組み込み関数で使用され、グループそのものを含めるかどうかを決定します。グループそのものを含めるかどうかを決定します。

## HAVING

HAVING の後には、必ず列関数 (SUM、AVG、MAX、MIN、COUNT など) が続きます。また、グループ値を見つけて HAVING 条件を完成させる副照会が続くことがあります。不要な行データを除くには WHERE、不要なグループ・データを除くには HAVING を使用してください。

たとえば、

- 正しい例: HAVING MIN(YEARS) > 6
- 誤った例: HAVING YEARS > 6

### 例 1

平均給与が \$12,000 を超える各部門について、管理者を除き、その部門の給与の最低額、最高額、平均額をリストするには、次のような照会を使用します。

```
SELECT DEPT, MIN(SALARY), MAX(SALARY), AVG(SALARY)
FROM Q.STAFF
WHERE JOB < > 'MGR'
GROUP BY DEPT
HAVING AVG(SALARY) > 12000
```

この照会は、次の報告書を作成します。

DEPT	MIN(SALARY)	MAX(SALARY)	AVG(SALARY)
15	12258.50	16502.83	13756.5100000000
20	13504.60	18171.25	15309.5333333333
38	12009.75	18006.00	14944.7000000000
42	10505.90	18001.75	13338.7500000000
51	13369.80	19456.50	16235.2000000000
66	10988.00	21000.00	16880.1750000000
84	13030.50	17844.00	15443.0000000000

HAVING キーワードには、必ずグループ分けされたデータを指定しなければなりません。HAVING 文節と GROUP BY 文節をともに使用するときは、GROUP BY ステートメントを前、HAVING ステートメントを後に置かなければなりません。

### 例 2

勤続年数ごとに、従業員が 2 名以下のグループを除いて、従業員の人数と平均給与をリストするには、次のような照会を使用します。

```
SELECT YEARS, COUNT(*), AVG(SALARY)
FROM Q.STAFF
GROUP BY YEARS
HAVING COUNT(*) > 2
```

この照会は、次の報告書を作成します。

YEARS	COUNT(EXPRESSION 1)	AVG(SALARY)
5	5	15552.0400000000
6	6	16930.0250000000
7	6	18611.8050000000
10	3	20162.6000000000
-	4	13694.0625000000

## IN

IN ステートメントは、指定した値のグループの少なくとも 1 つの値に一致する行を取り出すために使用します。IN ステートメントは、複数の OR ステートメントを使って条件を結合することと同じです。検索条件を列に適用するときは、複数の OR ステートメントを使うよりは IN ステートメントを使った方が簡単なことがあります。IN では、括弧内に少なくとも 2 つの値を指定する必要があります。値のリスト（ただし、IN では NULL を使えません）を括弧つきで指定してください。値と値はコンマで区切れます。ブランクを置いてもかまいません。

リスト内のオブジェクトの順序は任意です。どのような順序で指定しても同じ行を受け取ります。リスト中のオブジェクトの並び順で、結果の並び順が決まるということはありません。結果を特定の順序に配列するには、ORDER BY 文節を使用します。

次の照会で、

```
SELECT DEPTNUMB, DEPTNAME
FROM Q.ORG
WHERE DEPTNUMB IN (20, 38, 42)
```

以下の報告書が生成されます。

DEPTNUMB	DEPTNAME
20	MID ATLANTIC
38	SOUTH ATLANTIC
42	GREAT LAKES

上の照会で、IN(20, 38, 42) は、(DEPTNUMB = 20 OR DEPTNUMB = 38 OR DEPTNUMB = 42) と同じことです。

### 例

- Eastern 地区と Midwestern 地区のすべての部門を選択します。

```
SELECT DEPTNAME, DIVISION, LOCATION
FROM Q.ORG
WHERE DIVISION IN ('EASTERN', 'MIDWEST')
```

- 部門 15、20、38 の販売員と事務員を全員選択します。

```
SELECT ID, NAME, JOB, DEPT
FROM Q.STAFF
WHERE JOB IN ('CLERK', 'SALES')
AND DEPT IN (15, 20, 38)
```

- 勤続年数 (YEARS) が 1 年、2 年、3 年、または null の従業員を全員選択します。

```
SELECT ID, NAME, YEARS
FROM Q.STAFF
WHERE YEARS IN (1, 2, 3) OR YEARS IS NULL
```

## INSERT

INSERT は表にデータを追加する SQL ステートメントです。INSERT ステートメントの形式は、次のとおりです。

```
INSERT INTO tablename
VALUES (value1, value2, ...)
```

## INSERT

この構文で、*tablename* はデータの挿入先である表または視点の名前、*value1*, *value2* などは挿入する値です。

VALUES に続くデータ値のリストは、挿入先の表の列リストに対応していなければなりません。つまり、値と列が同数で、値と対応列のデータ・タイプが一致していなければなりません。次の例に示すように、NULL を指定してヌル値を挿入できます。

次のステートメントは、

```
INSERT INTO PERS  
VALUES (400, 'HARRISON', 20, 'SALES', NULL, 18000.66, 0)
```

次の行が **PERS** 表に挿入されます。

ID	NAME	DEPT	JOB	YEARS	SALARY	COMM
400	HARRISON	20	SALES	-	18000.66	0.00

PERS 表は Q.STAFF サンプル表のコピーです。どのように作成されたかについては、152 ページの『CREATE TABLE』に例が示されています。CREATE TABLE ステートメントを使いたくなければ、次の 2 つのコマンドでも PERS を作成できます。

```
DISPLAY Q.STAFF  
SAVE DATA AS PERS
```

## 行に列値を挿入する

行を挿入するとき、INSERT ステートメントに列リストを指定することで、行中の特定の列にだけ値を入れることができます。

次の例のようにして、列と挿入値を指定します。

```
INSERT INTO PERS (ID, NAME, JOB, SALARY)  
VALUES (510, 'BUCHANAN', 'CLERK', 11500.75)
```

DRAW コマンドを使って INSERT 照会を簡単に作成する方法は次のとおりです。

```
DRAW tablename (TYPE=INSERT
```

値指定のない列には、値が入りません (NULL)。NOT NULL と定義されている列には、必ず値を指定しなければなりません。

## 表から表へ行をコピーする

行を表に挿入するとき、INSERT の VALUES 文節の代わりに副照会を用いて、挿入すべき列を指定して別の表からデータをコピーしてくることもできます。副照会から検索された情報は、あたかも複数の INSERT コマンドが入力されたかのように、表の中に置かれます。

次のステートメントは、部門 38 の従業員の ID、NAME、JOB、YEARS 行を Q.STAFF から PERS へコピーします。

```
INSERT INTO PERS (ID, NAME, JOB, YEARS)  
SELECT ID, NAME, JOB, YEARS  
FROM Q.STAFF  
WHERE DEPT = 38
```

NOT NULL と定義されている列には、必ず値を指定しなければなりません。

選択する列と挿入する列のあいだには、必ずしも 1 対 1 の対応関係がなくてもかまいません。ただ、挿入する列より選択する列が多くてはなりません。挿入される列よりも選択される列が少ない場合は、残りの列に空白が挿入されます。ある表から選択した行を、同じ表へ挿入することはできません。

データを表に挿入するときに必要な許可の詳細は、158 ページの『GRANT』を参照してください。

## IS

IS キーワードは、必ず NULL または NOT NULL をともないます。例については、168 ページの『NULL』を参照してください。

## LIKE

値の一部だけがわかっている文字データ、漢字データ、または 2 進データを選択するには、WHERE 文節に LIKE を用い、不明データを記号で表します。

- パーセント記号 (%) は、不特定数 (0 個も含みます) の文字を表します。
- 下線 (\_) 記号は任意の単一文字を表す記号である。文字が何であるかは不明ながら正確な文字数がわかっているときは、その数だけ下線を連続指定します。

% と \_ を組み合わせることもできます。たとえば、第 2 文字と第 3 文字が AN または ON であるようなすべての名前を選択するには、次のようにします。

```
SELECT ID, NAME
FROM Q.STAFF
WHERE NAME LIKE '_AN%' OR NAME LIKE '_ON%'
```

LIKE は、SQL 照会では文字データ、漢字データ、および 2 進データでのみ使用でき、QBE 照会では、文字データと漢字データでのみ使用できます。文字データの場合、LIKE の後に指定する値を单一引用符で囲んでください。漢字データの場合には、LIKE の後に指定する値の前に单一引用符で囲まれた 1 バイト文字の 'G' を入れます。パーセント記号と下線には 2 バイト文字を使用します。

## 文字ストリングの選択

ある単語または数値がデータ中にあることがわかっているとき、その一部を文字ストリングとして指定することによって、その単語または数値を含んでいる行を選択できます。次の例にある WHERE ADDRESS LIKE '%NY' は、末尾に NY が含まれる住所を選択します。パーセント記号 (%) は、不特定数 (0 個も含みます) の文字を表します。

次の照会で、

```
SELECT NAME, ADDRESS
FROM Q.APPLICANT
WHERE ADDRESS LIKE '%NY'
```

以下の報告書が生成されます。

## LIKE

NAME	ADDRESS
JACOBS	POUGHKEEPSIE, NY
REID	ENDICOTT, NY
LEEDS	EAST FISHKILL, NY

LIKE を使って特定の終わりを持つデータを検索するときは、目的の列のデータ・タイプに注意してください。列が固定幅で、列内のデータが可変幅であれば、列データ中のブランクに合わせて文字ストリングのブランクを追加してください。

たとえば、上の例で ADDRESS 列のデータ・タイプが CHAR(17) だとすれば、列幅は固定されていて、データが列幅ほどないときは残りがスペースで埋められることになります。文字ストリングの末尾で検索する場合、データ中の後続ブランクの数が幾通りもありうるときは、そのすべてを予期して検索を行わなければなりません。

ADDRESS 列のデータ・タイプが VARCHAR であれば、データに後続ブランクは付加されず、列幅はデータ幅に合わせて変化します。したがって、データベースでは、列中のどの行にも後続ブランクがありません。

たとえば、名前が W で始まる人をすべて選択するには、次のような照会を使用します。

```
SELECT ID, NAME  
FROM Q.STAFF  
WHERE NAME LIKE 'W%'
```

## 特定の文字の無視

文字数が不定のストリングを搜すときに、下線 (\_) を使用できます。検索で無視したい文字数に合わせ、それと同数の下線を使用してください。たとえば、次の文節は、8 桁の部品番号の列で、2 から 6 桁目が「G2044」の文字ストリングを検索するため使用します。先頭の 1 文字と末尾の 2 文字は無視されます。DB2 (z/OS 版) では、すべて数字の値は、单一引用符で囲む必要があります。(値のあとに 2 個の下線付けられ、全体が引用符で囲まれていることに注意してください。)

```
WHERE PARTNO LIKE '_G2044_'
```

### 例

- 2 番目以降のどこかの文字位置に S を含むすべての名前を選択するには、次のような照会を使用します。

```
SELECT ID, NAME  
FROM Q.STAFF  
WHERE NAME LIKE '_%S%'
```

- SON で終わるすべての名前を選択するには、次のような照会を使用します。

```
SELECT ID, NAME  
FROM Q.STAFF  
WHERE NAME LIKE '%SON'
```

この例のとおりでよいのは、NAME 列のデータ・タイプが VARCHAR で、データベースではデータに後続ブランクが含まれていないからです。列が固定の幅をもつデータ・タイプ CHAR の場合、照会は、末尾に SON を持つあらゆる長さの名前を想定し、そのすべての組み合わせを検索値に含めなければなりません。

## MAX および MIN

MAX と MIN は、文字値、漢字値、数値、日時、および 2 進データ（バイナリーラージ・オブジェクト、つまり BLOB データを除く）をもつ列に使えます。

MIN および MAX 関数は、次のように書きます。

```
MAX(expression)
MIN(expression)
```

括弧は必ず付けてください。この構文で、*expression* は、ほとんどの場合は列名ですが、以下のものも指定できます。

- 少なくとも 1 つの列名を含む算術式。
- DISTINCT キーワードと、それに続く列名。

関数中の列名は、長ストリング列や、列関数から導かれた列を参照してはなりません。（視点中の列は、関数から導かれることができます。）列関数を他の列関数内にネストさせることはできません。

MAX 関数または MIN 関数では、オペランドが NOT NULL と定義されていても、結果のデータ・タイプはヌルであってかまいません。組み込み関数による計算には、ヌル値は用いられません。

次の例では、SELECT ステートメントに複数の列関数が組み込まれています。部門 10 の従業員給与の合計額、最低額、平均額、最高額、従業員数 (COUNT) を計算して、表示します。

```
SELECT SUM(SALARY), MIN(SALARY), AVG(SALARY),
       MAX(SALARY), COUNT(*)
  FROM Q.STAFF
 WHERE DEPT = 10
```

文字データに MAX または MIN を使用するときは、データの比較に 2 進照合シンケンスが使用されます。

---

## NOT

照会の WHERE 文節で NOT キーワードを使用することで、データを除外できます。次の例では、EASTERN と WESTERN を除くすべての部門が選択されます。

次の照会で、

```
SELECT DEPTNUMB, LOCATION,
       DIVISION FROM Q.ORG
 WHERE NOT
       (DIVISION = 'EASTERN' OR DIVISION = 'WESTERN')
```

以下の報告書が生成されます。

DEPTNUMB	LOCATION	DIVISION
10	NEW YORK	CORPORATE
42	CHICAGO	MIDWEST
51	DALLAS	MIDWEST

## NOT

NOT 条件の適用範囲を明確にするには、括弧を使用してください。括弧なしで NOT を AND や OR と併用すると、まず NOT で条件が否定され、続いて AND 結合または OR 結合が行われます。たとえば、A、B、C を条件とすれば、次の 2 つの句は同等です。

```
NOT A AND B OR C  
((NOT A) AND B) OR C
```

「より大」、「より小」、「等しい」を否定するときは、条件全体の前に NOT を置かなければなりません。たとえば、WHERE NOT YEARS = 10 のようにします。等号の否定には、否定記号 ( $\neg$ ) も使用できます。

### 正しい例:

- WHERE YEARS  $\neg > 10$
- WHERE NOT YEARS = 10

### 誤った例:

```
WHERE YEARS NOT = 10
```

記号  $\neg=$  は、 $<>$  (不等号) の代替演算子です。これは ANSI SQL 演算子です。  
(リモート・データ・アクセスでは、なるべく  $<>$  記号を使用してください。)

NOT NULL、NOT LIKE、NOT IN、NOT BETWEEN を使用できます。これらの場合に限り、NOT を条件全体の後に使用できます。たとえば、

```
WHERE YEARS IS NOT NULL
```

### 例

- 給与が \$17,000 と \$21,000 の間にはない従業員をすべて選択するには、次のような照会を使用します。

```
SELECT ID, NAME, SALARY  
FROM Q.STAFF  
WHERE SALARY NOT BETWEEN 17000 AND 21000
```

- 給与が \$18,000 未満で歩合が \$500 未満の者を除き、すべての従業員を選択するには、次のような照会を使用します。

```
SELECT ID, NAME, SALARY, COMM  
FROM Q.STAFF  
WHERE NOT (SALARY < 18000 AND COMM < 500)
```

- Q.ORG 表で部門管理者になっていない Q.STAFF の管理者のみを選択するには、次のような照会を使用します。

```
SELECT ID, NAME, DEPT  
FROM Q.STAFF  
WHERE JOB = 'MGR'  
AND ID NOT IN (SELECT MANAGER FROM Q.ORG)
```

---

## NULL

表が作成され、まだ一部しかデータで埋められていないれば、データの入っていないフィールドは NULL と見なされ、値が不明ということを意味します。NULL は、次のどの値とも同じではありません。

- 数値のゼロ
- 全文字ブランクの文字ストリング

- ・長さがゼロの文字ストリング
- ・NULL という 4 桁の長さをもつ文字ストリング

これらの値はいずれも、表の行や列に入力できます。NULL は、値が入力されていない個所や値が NULL に特に設定される個所の値です。印刷や表示では、デフォルトで、1 個のハイフン (-) で表されます。

- ・正しい例: WHERE *columnname* IS NULL
- ・誤った例: WHERE *columnname* = ''

VALUE スカラー関数を使用すれば、印刷や表示での NULL 値の表し方を変更できます。詳細については、189 ページの『ストリング関数』を参照してください。

列に NULL 値をもつ行を選択するには、次を入力します。

```
WHERE columnname IS NULL
```

#### 例

- ・歩合制でない人をすべて選択するには、次のような照会を使用します。

```
SELECT ID, NAME
FROM Q.STAFF
WHERE COMM IS NULL
```

- ・歩合がゼロの人をすべて選択するには、次のような照会を使用します。

```
SELECT ID, NAME
FROM Q.STAFF
WHERE COMM = 0
```

ゼロは、NULL 値と同じではありません。Q.STAFF サンプル表に、この条件を満たす行はありません。

- ・歩合制で働いている人をすべて選択するには、次のような照会を使用します。

```
SELECT ID, NAME
FROM Q.STAFF
WHERE COMM IS NOT NULL
```

## OR

条件を OR で結合すれば、複数の条件に基づく行選択ができます。OR で結合された条件のもとでは、条件のどれかを 1 つでも満たす行が選択されます。

次の照会で、

```
SELECT ID, NAME, YEARS, SALARY
FROM Q.STAFF
WHERE YEARS = 10 OR SALARY > 20000
```

以下の報告書が生成されます。

ID	NAME	YEARS	SALARY
50	HANES	10	20659.80
140	FRAYE	6	21150.00
160	MOLINARE	7	22959.20
210	LU	10	20010.00
260	JONES	12	21234.00
290	QUILL	10	19818.00
310	GRAHAM	13	21000.00

## ORDER BY

OR の結果を、 145 ページの『AND』と比較してみてください。

---

## ORDER BY

SQL SELECT ステートメントでは、選択された行の表示順序も指定できます。また、選択された行の中から、重複行を除去することもできます。

ORDER BY では、選択した行を報告書にどのように配列するかを指定します。 ORDER BY を使用するときは、それをステートメント全体の最後の文節にしてください。 ORDER BY の後に指定する列名は、SELECT の後にも指定していなければなりません。

ORDER BY 文節の形式は、次のとおりです。

ORDER BY *columnname* DESC

DESC キーワードは、降順にデータを表示することを指定します。配列順序をとくに指定しなければ、昇順になります。

次の報告書では、行が昇順 に配列されます。

次の照会で、

```
SELECT NAME, JOB, YEARS
FROM Q.STAFF
WHERE DEPT = 84
ORDER BY JOB
```

以下の報告書が生成されます。

NAME	JOB	YEARS
GAFNEY	CLERK	5
QUILL	MGR	10
DAVIS	SALES	5
EDWARDS	SALES	7

## ソート・シーケンス

文字データを数値順にソートすると、次のように配列されます。

1. 特殊文字 (ブランクを含む)
2. 小文字 (英字順)
3. 大文字 (英字順)
4. 数字
5. NULL 値

数値のソートのデフォルト・シーケンスは、昇順です。

DATE、TIME、TIMESTAMP の値のソートのデフォルト・シーケンスは、日時順です。DBCS データのソート・シーケンスはデータの内部値で決まり、通常、あまり意味がありません。

### 例

- 従業員を給与額で降順にリストするには、次のような照会を使用します。

```
SELECT ID, NAME, SALARY
FROM Q.STAFF
ORDER BY SALARY DESC
```

- 従業員をラストネームで昇順にリストするには、次のような照会を使用します。

```
SELECT ID, NAME, SALARY
FROM Q.STAFF
ORDER BY NAME
```

## 複数列による配列

配列基準に複数の列を使うときは、ORDER BY の後に列名または列番号のリストを指定します。1つのリストに列名と列番号が混在してもかまいません。

定義された列を配列の基準に使いたいときは、必ずその列番号を使用しなければなりません。詳細については、172ページの『列番号による配列』を参照してください。

ORDER BY 文節の列名（と、それに続く ASC または DESC）は、ソート指定です。ソート指定をリストするときは、1つ1つをコンマで区切ってください。まず ORDER BY 文節の直後に指定されている列を基準にした配列が行なわれ、続いてその配列の枠内で、2番目に指定されている列に基づく配列が行なわれます。以下、同様です。

### 例

- 職種別に勤続年数で配列するには、次のような照会を使用します。

```
SELECT NAME, JOB, YEARS
FROM Q.STAFF
WHERE DEPT=84
ORDER BY JOB, YEARS DESC
```

この照会は、次の報告書を作成します。

NAME	JOB	YEARS
GAFNEY	CLERK	5
QUILL	MGR	10
EDWARDS	SALES	7
DAVIS	SALES	5

- 勤続年数別に職種で配列するには、次のような照会を使用します。

```
SELECT NAME, JOB, YEARS
FROM Q.STAFF
WHERE DEPT=84
ORDER BY YEARS DESC, JOB
```

この照会は、次の報告書を作成します。

NAME	JOB	YEARS
QUILL	MGR	10
EDWARDS	SALES	7
GAFNEY	CLERK	5
DAVIS	SALES	5

- 従業員を勤続年数で降順に配列し、年数が同じときは給与額で降順にリストするには、次のような照会を使用します。

```
SELECT YEARS, ID, NAME, SALARY
FROM Q.STAFF
ORDER BY YEARS DESC, SALARY DESC
```

## ORDER BY

- 従業員を部門別に配列し、部門内では、給与額で昇順にリストするには、次のような照会を使用します。

```
SELECT DEPT, ID, NAME, SALARY  
FROM Q.STAFF  
ORDER BY DEPT, SALARY
```

## 列番号による配列

ORDER BY ステートメントの後に、SALARY+COMM のような式を使用することはできません。式で定義された列を並べるには、照会の SELECT ステートメントの中で、列の位置を指定する番号を使用します。たとえば、次の照会があるとします。

```
SELECT ID, NAME, SALARY+COMM  
FROM Q.STAFF  
WHERE COMM IS NOT NULL  
ORDER BY 3
```

上の照会で、SALARY+COMM は、SELECT ステートメントの中で列 3 です。したがって、ORDER BY 3 は、その列で配列することを指定しています。照会の最後の行は、次のように書けます。

```
ORDER BY 3 DESC, NAME
```

ORDER BY の後のリストには複数の列番号を含めることができます。また、列名と列番号を同じリストに混在させることができます。たとえば、従業員を部門別に配列し、部門内では、給与額で降順にリストするには、次のような照会を使用します。

```
SELECT DEPT, ID, NAME, SALARY  
FROM Q.STAFF  
ORDER BY 1, 4 DESC
```

---

## REVOKE

REVOKE ステートメントは、GRANT ステートメントで与えられた許可を取り消します。 REVOKE ステートメントの構文は、次のとおりです。

```
REVOKE operation-list ON tablename FROM user-list
```

*operation-list*

ALTER、DELETE、INSERT、SELECT、UPDATE を、コンマで区切りながら 1 つ以上並べたリスト。すべての特権を一度に取り消すには、ALL を使用します。

*tablename*

許可取り消しのオブジェクトになっている表または視点の名前。

*user-list*

各ユーザー ID をコンマで区切ってリストします。ユーザー・リストの代わりに PUBLIC も指定できます。 PUBLIC を指定しても、個々に認可を与えたユーザー ID からは特権がはく奪されません。そのような特権の取り消しは、やはり個々に行わなければなりません。

REVOKE と GRANT は似ていますが、次の相違があります。

- REVOKE では、UPDATE の後に列リストを指定できません。UPDATE はすべての列の更新許可を取り消します。特定の列についてのみ更新許可を取り消し、他の列については更新を許可し続けるには、次のようにします。
  - すべての列の更新許可を取り消す。
  - 特定の列のリストを指定して更新許可を与える。
- JONES に特権を与え、JONES が JACOBS に特権を与えた場合、JONES の特権を取り消すと、JACOBS の特権も取り消されます。

次のステートメントは、PERS 表を使って、SELECT 照会を書く権限を、JACOBS から取り消します。

```
REVOKE SELECT ON PERS FROM JACOBS
```

次のステートメントは、PERS 表の列の更新権限を ユーザー HSAM4419 から取り消します。

```
REVOKE UPDATE ON PERS FROM HSAM4419
```

## SELECT

SELECT ステートメントでは、表から取り出したい各列の名前を指定できます。表または視点に含まれる 1 つまたは複数の列を個々に指定することもできます。1 つの SELECT ステートメントで複数の表から情報を選択できます。複数の表からデータを選択するときに、重複する情報を除去する方法は、156 ページの『DISTINCT』を参照してください。

QMF は、選択されたデータを、そのデータ・タイプのデフォルトの編集コードに基づいて表示します。さまざまなデータ・タイプでどのように表示されるかについては、261 ページの『編集コード』を参照してください。

使用しているデータベース・マネージャーの SQL 参照情報に、SELECT ステートメントで使用する表、視点、および列の制限についての詳細が記載されています。該当する参照情報については、IBM Publications Center ([www.ibm.com/shop/publications/order](http://www.ibm.com/shop/publications/order)) を参照してください。

## 表から全列を選択する

表からすべての列を取り出すには、列名を指定する代わりにアスタリスク (\*) を使用します。全列選択に指定する SELECT ステートメントのフォーマットは、次のとおりです。

```
SELECT * FROM tablename
```

このステートメントで、*tablename* は、検索対象の表または視点の名前です。たとえば、次のステートメントは Q.ORG の中のすべての列を戻します。

```
SELECT * FROM Q.ORG
```

次の照会はすべての列を戻しますが、部門番号が 10 の行のみ表示します。

```
SELECT *
FROM Q.STAFF
WHERE DEPT = 10
```

## SELECT

### 表から列を選択する

表から列を選択するには、SELECT に続いて目的の列の正確な名前を、報告書での望みの並び順に従って左から右へ指定します。列名はコンマで区切ってください。

次のステートメントは、部門名が左、部門番号が右に置かれた報告書を作成します。

```
SELECT DEPTNAME, DEPTNUMB  
FROM Q.ORG
```

報告書での列の並び順は、書式を変えることで変更できます。書式での列のデフォルトの順序は、照会で指定した列の順序と同じです。

1 つの列を複数回選択することも可能です。これにより、複数の集約関数を書式で使用できます。

DB2 (z/OS 版) では、最高 750 個の列名 (または式) を選択できます。DB2 (VM および VSE 版) データベースに接続しているときは、255 個までです。

SELECT 文節で使用されていない列名でも、WHERE 文節で使用してかまいません。

#### 例

- Q.STAFF 表から ID 列と NAME 列だけを選択するには、次のような照会を使用します。

```
SELECT ID, NAME  
FROM Q.STAFF
```

- Q.STAFF 表から NAME 列と ID 列を選択し、NAME 列を先に並べるには、次のような照会を使用します。

```
SELECT NAME, ID  
FROM Q.STAFF
```

### 記述列を追加する

SELECT ステートメントの列リストに引用符で囲んだ定数を入れることで、報告書に説明的な情報を追加できます。引用符で囲まれる定数情報は、英字でも、数字でも、英数字を組み合わせたものでもかまわず、長さは 256 文字まで許されます。次の例では、Q.APPLICANT 表から 14 年間の学歴 (EDLEVEL) をもつ人を選び出し、その名前 (NAME) と住所 (ADDRESS) を示すとともに、その人が応募者 (APPLICANT) であることを補足しています。

次の照会を実行すると、

```
SELECT NAME, ADDRESS, 'APPLICANT'  
FROM Q.APPLICANT  
WHERE EDLEVEL = 14  
ORDER BY NAME
```

次の報告書が作成されます。

NAME	ADDRESS	EXPRESSION 1
CASALS	PALO ALTO,CA	APPLICANT
REID	ENDICOTT,NY	APPLICANT
RICHOWSKI	TUCSON,AZ	APPLICANT

この報告書には、3列が並びます。NAME(名前)列とADDRESS(住所)列、そして選択された行ごとにAPPLICANT(応募者)という言葉が入る新しい列です。新しく作られた列には、データベース・マネージャーが列名を付けます。この名前は、使用しているサイトのデータベース・マネージャーによって異なります。この列名は、書式パネルで変更できます。

## 副照会の使用

副照会は、表からデータを選択します。選択されたデータは、次に、主照会のWHERE文節にある条件のテストに使用されます。たとえば、次の照会には、Eastern地区で働く従業員のリストを作成する副照会(SELECT DEPTNUMBステートメントで始まる)があります。

```
SELECT NAME, ID
FROM Q.STAFF
WHERE DEPT = SOME
  (SELECT DEPTNUMB
   FROM Q.ORG
   WHERE DIVISION='EASTERN')
```

まず、副照会でEastern地区にある部門の番号を求めます。次に、主照会でこれらの部門のいずれかで働く従業員を求めます。

複数の副照会があるときは、最後の副照会が最初、最初の副照会が最後に実行されます。

### 例:

次の各例には、それぞれ1つの副照会が含まれており、副照会は強調表示されています。

```
SELECT DEPT, NAME, SALARY
FROM Q.STAFF CORRVAR
WHERE SALARY =
  (SELECT MAX(SALARY)
   FROM Q.STAFF
   WHERE DEPT = CORRVAR.DEPT)

SELECT ID, NAME
FROM Q.STAFF
WHERE DEPT IN
  (SELECT DISTINCT DEPTNUMB
   FROM Q.ORG
   WHERE DIVISION = 'MIDWEST')
ORDER BY ID

SELECT DEPT, AVG(SALARY)
FROM Q.STAFF
GROUP BY DEPT
HAVING AVG(SALARY) >
  (SELECT AVG(SALARY) FROM Q.STAFF)
```

## SOME

単一の値でなく、1組の値を返させたいときは、照会でSOMEキーワードを比較演算子を併用します。SOMEは、次の比較演算子と併用できます。

=    !=    >    >=    <    <=    < >

## SOME

記号  $\neq$  は、 $< >$  (不等号) の代替記号です。これは ANSI SQL 演算子です。 (リモート・データ・アクセスでは、なるべく  $< >$  記号を使用してください。)

1 組の値を返させるのに、ALL、ANY、IN も使用できます。

- ALL では、返されてくるすべての値が条件を満たします。
- ANY または SOME では、返されてくる値の中に条件を満たすものが少なくとも 1 つあります。
- 副照会で、次の代わりに、IN を使用できます。

= SOME  
= ANY

次の照会では、Eastern 地区で働く従業員のリストが得られます。まず、副照会で Eastern 地区にある部門の番号を求めます。次に、主照会でこれらの部門で働く従業員を求めます。

```
SELECT NAME, ID
FROM Q.STAFF
WHERE DEPT = SOME
  (SELECT DEPTNUMB FROM Q.ORG WHERE DIVISION='EASTERN')
```

この照会で SOME キーワードを用いたのは、Eastern 地区に複数の部門があり、そのいずれかで働いている従業員を選択したいからです。SOME (または ANY) でなく ALL を使用したとすれば、結果は空集合となります。Eastern 地区のすべての部門で働いている従業員はいません。

---

## SUM

SUM は、数値を含んでいる列にしか使用できません。

SUM の結果データ・タイプは、たとえば、オペランドが NOT NULL でも、ヌルであってかまいません。組み込み関数による計算には、ヌル値は用いられません。

次の例では、SELECT ステートメントに複数の列関数が組み込まれています。部門 10 の従業員給与の合計額、最低額、平均額、最高額、従業員数 (COUNT) を計算して、表示します。

次の照会を実行すると、

```
SELECT SUM(SALARY), MIN(SALARY), AVG(SALARY),
       MAX(SALARY), COUNT(*)
  FROM Q.STAFF
 WHERE DEPT = 10
```

次の報告書が作成されます。

SUM(SALARY)	MIN(SALARY)	AVG(SALARY)	MAX(SALARY)	COUNT(EXPRESSION)
-----	-----	-----	-----	-----
83463.45	19260.25	20865.8625000000	22959.20	4

SUM 列関数の書き方は次のように書けます。

SUM(*expression*)

括弧は必ず付けてください。上記の構文で、*expression* は、ほとんどの場合は列名ですが、以下のものも指定できます。

- 少なくとも 1 つの列名を含む算術式。

- DISTINCT と、それに続く列名。

関数中の列名で、長ストリング列を参照してはなりません。また、列関数から導かれる列を参照してはなりません（視点の列は、関数から導かれることができます）。列関数を他の列関数内にネストさせることはできません。

## UNION

UNION は、複数の表の行をマージして、1 つの報告書を作成します。組み合わせられる行は、相互に関連性をもち、同じデータ幅と同じデータ・タイプを持っているものでないと、組み合わせる意味がありません。UNION では、複数の表の値をマージして、同じ報告書の同列異行に入れることができます。1 つの照会で複数の UNION を使用できます。

このトピックの例には UNION ALL の使用例も含まれていますが、UNION ALL には拡張 UNION サポートが必要です。詳細については、311 ページの『付録 C. 特定のサポートを必要とする QMF 機能』を参照してください。

次の例は、Q.STAFF から名前 (NAME) 列と 従業員 (EMPLOYEE) 列、Q.APPLICANT から NAME (名前) 列と APPLICANT (応募者) 列を選択します。

次の照会を実行すると、

```
SELECT NAME, 'EMPLOYEE'
FROM Q.STAFF
WHERE YEARS < 3
UNION
SELECT NAME, 'APPLICANT'
FROM Q.APPLICANT
WHERE EDLEVEL > 14
```

次の報告書が作成されます。

NAME	EXPRESSION 1
BURKE	EMPLOYEE
GASPARD	APPLICANT
JACOBS	APPLICANT

Q.STAFF から選択する照会部分では、報告書に EMPLOYEE という定数の入った列も作成します。Q.APPLICANT から選択する照会部分でも、APPLICANT という定数の入った列を作成します。新しく作られる列にはデフォルトの列名が与えられます、この名前は書式パネルで簡単に変更できます。

どの照会でも、対応する列の間で長さの突き合わせが行なわれます。上の照会では、APPLICANT の長さに合わせて、EMPLOYEE には 1 個のブランクが付加されています。

次の例は、すべての管理者と、その管理者が面接した人々を、Q.STAFF と Q.INTERVIEW から選択します。

次の照会を実行すると、

```
SELECT NAME, ''
FROM Q.STAFF, Q.INTERVIEW
WHERE MANAGER = ID
UNION
```

## UNION

```
SELECT NAME, 'NO INTERVIEWS'  
FROM Q.STAFF  
WHERE JOB = 'MGR'  
AND ID NOT IN (SELECT MANAGER FROM Q.INTERVIEW)
```

次の報告書が作成されます。

NAME	EXPRESSION 1
DANIELS	NO INTERVIEWS
FRAYE	
HANES	
JONES	NO INTERVIEWS
LEA	
LU	NO INTERVIEWS
MARENghi	NO INTERVIEWS
MOLINARE	
PLOTZ	
QUILL	
SANDERS	

## UNION で重複行を残す

UNION の前提是、2 つの SELECT ステートメントで指定された列から、DISTINCT な（他と異なる）行だけを選択するということです。

UNION 操作の結果に重複行を残したければ、UNION に続けて ALL キーワードを指定してください。 UNION ALL が指定されると、結果から重複行が除去されません。

次の例では、勤続年数が 5 年を超えるか、歩合が \$850 を超えるすべての販売員を、Q.STAFF から選択します。条件を 2 つとも満たす販売員は、報告書に 2 回記載されます。

次の照会を実行すると、

```
SELECT * FROM Q.STAFF  
WHERE JOB = 'SALES' AND YEARS > 5  
UNION ALL  
SELECT * FROM Q.STAFF  
WHERE JOB = 'SALES' AND COMM > 850  
ORDER BY 2
```

次の報告書が作成されます。

ID	NAME	DEPT	JOB	YEARS	SALARY	COMM
340	EDWARDS	84	SALES	7	17844.00	1285.00
340	EDWARDS	84	SALES	7	17844.00	1285.00
310	GRAHAM	66	SALES	13	21000.00	200.30
90	KOONITZ	42	SALES	6	18001.75	1386.70
90	KOONITZ	42	SALES	6	18001.75	1386.70
40	O'BRIEN	38	SALES	6	18006.00	846.55
20	PERNAL	20	SALES	8	18171.25	612.45
70	ROTHMAN	15	SALES	7	16502.83	1152.00
70	ROTHMAN	15	SALES	7	16502.83	1152.00
220	SMITH	51	SALES	7	17654.50	992.80
220	SMITH	51	SALES	7	17654.50	992.80
150	WILLIAMS	51	SALES	6	19456.50	637.65
280	WILSON	66	SALES	9	18674.50	811.50

UNION ALL でなく UNION を指定すると、図 14 に示すように、条件を 2 つとも満たしている販売員でも報告書には 1 度しか現われないので、誰がそうかを見極めるのに多少の労力を要します。

ID	NAME	DEPT	JOB	YEARS	SALARY	COMM
340	EDWARDS	84	SALES	7	17844.00	1285.00
310	GRAHAM	66	SALES	13	21000.00	200.30
90	KOONITZ	42	SALES	6	18001.75	1386.70
40	O'BRIEN	38	SALES	6	18006.00	846.55
20	PERNAL	20	SALES	8	18171.25	612.45
70	ROTHMAN	15	SALES	7	16502.83	1152.00
220	SMITH	51	SALES	7	17654.50	992.80
150	WILLIAMS	51	SALES	6	19456.50	637.65
280	WILSON	66	SALES	9	18674.50	811.50

図 14. UNION ステートメントの結果の例

各副照会の評価順序によって照会全体の結果が変わることはありません。しかし、2 つの SELECT 照会を UNION ALL または UNION で結合するときは、評価順序によって結果が変わります。括弧で囲まれた部分が最初に評価されます。内側の括弧ほど先に評価されます。次に、各文節が左から右へ評価されます。

たとえば、次の 2 つの照会では異なる結果が生じます。

- この例では、TABLE1 のすべての行が TABLE2 のすべての行とマージされて、中間表を作ります。次に、この中間表が TABLE3 とマージされて、重複行が除かれます。

(TABLE1 UNION ALL TABLE2) UNION TABLE3

- 次の例では、TABLE2 のすべての行が TABLE3 とマージされ、そこから重複行を除いたものが中間表となります。次に、その中間表が TABLE1 のすべての行とマージされます。

TABLE1 UNION ALL (TABLE2 UNION TABLE3)

## UNION の使用規則

- 2 つの SELECT ステートメントの間に UNION を入れられるのは、その 2 つのステートメントが同じ数の列を選択しており、対応する列が互換データ・タイプである（たとえば、数値対数値）の場合のみです。
- UNION でマージされる 2 つの選択ステートメントの対応列は、必ずしも同名でなくてかまいません。割り込み列の名前が異なるようになるため、ORDER BY の後には列名を使用しないでください。代わりに、いつも ORDER BY 1 のように列番号を使用します。
- SELECT ステートメントに指定される列の長さとデータ・タイプについては、比較可能が条件となります。ともに数値、ともに文字、ともに図形、ともに日付、ともに時刻、ともにタイム・スタンプのいずれかでなければなりません。これらのデータ・タイプの組み合わせは認められません。たとえば、

```
SELECT ID  
:  
UNION
```

## UNION

```
SELECT DEPT  
:
```

ID が CHAR(6) で、DEPT が CHAR(3) なら、結果表の対応列は CHAR(6) となります。DEPT から結果表に取り込まれた値は、右側にブランクが付加されます。

## いつ UNION を使用するか、いつ表を結合するか

いつ UNION を使い、いつ表を結合するかは、報告書にどのような結果がほしいかで決まります。

- UNION は、2 つの照会で選択された行を割り込ませながら、1 つの報告書を作成します。
- 表の結合では、一方の表の行を割り込ませるのではなく、1 つの表の各行をもう1 つの表の対応行と水平的に結合させます。結合時には、一方の表のすべての行が他方のすべての行と結合しないよう、組み合わせの数を制限するための条件(WHERE 文節)を定めることが必要不可欠です。

次の照会では、177 ページの『UNION』の UNION 照会のような読みやすい、また意味のある報告書は得られません。この照会の WHERE 条件には、2 つの表を結合するための共通の列が使用されておらず、したがって報告書では行の重複が起こります。

次の照会を実行すると、

```
SELECT S.NAME, 'EMPLOYEE', A.NAME, 'APPLICANT'  
FROM Q.STAFF S, Q.APPLICANT A  
WHERE YEARS < 3 AND EDLEVEL > 14
```

次の報告書が作成されます。

NAME	EXPRESSION 1	NAME1	EXPRESSION 2
BURKE	EMPLOYEE	JACOBS	APPLICANT
BURKE	EMPLOYEE	GASPARD	APPLICANT

2 つの SELECT ステートメントが同一の表を参照しているとき、その 2 つを UNION で結ぶこともできます。たとえば、部門ごとにすべての従業員を番号順にリストし、勤続年数が 10 年の者を識別するには、次のような照会を使用します。

```
SELECT DEPT, ID, NAME, YEARS, 'TEN YEARS'  
FROM Q.STAFF  
WHERE YEARS = 10  
UNION  
SELECT DEPT, ID, NAME, YEARS, ''  
FROM Q.STAFF  
WHERE NOT YEARS = 10  
ORDER BY 1, 2
```

---

## UPDATE

UPDATE ステートメントは、表の各行にある既存の列のうち、指定された列の値を更新します。表を更新できるのは、それがユーザー自身で作成した表であるか、その表の更新を特に許可されている場合に限られます。データ更新に必要な特権の詳細は、158 ページの『GRANT』を参照してください。

UPDATE ステートメントは、次の部分で構成されます。

- UPDATE は、更新すべき表を指定します。
- SET は、更新すべき列と、表に入れる新しい値を指定します。
- WHERE は、更新すべき行を指定します。

次の例は、PERS 表の従業員 250 を更新します。職種を SALES に変更し、給与を 15% 増額します。

```
UPDATE PERS
SET JOB='SALES', SALARY=SALARY * 1.15
WHERE ID = 250
```

UPDATE 照会を簡単に作成する方法は、DRAW コマンドを発行するときに、UPDATE 照会タイプを指定することです。詳細については、32 ページの『DRAW』を参照してください。

1 つの UPDATE ステートメントで表中の複数の行を更新できますし(次に示す最初の例を参照)、WHERE 文節を省略することで、ある列の全行を更新することもできます。

#### 例

- PERS にあるすべての事務員の給与を \$300 増額するには、次のように UPDATE ステートメントを使用します。

```
UPDATE PERS
SET SALARY = SALARY+300
WHERE JOB = 'CLERK'
```

- PERS 表にある全員の勤続年数を 1 年増やすには、次のように UPDATE ステートメントを使用します。

```
UPDATE PERS
SET YEARS = YEARS + 1
```

## WHERE

SELECT ステートメントの中で WHERE 文節を使用して、検索、更新、または削除する行を特定するための条件(1 つ以上の選択基準)を指定します。検索条件を満たす行のみが影響を受けます。次の例では、DEPT 列の値が 20 であることを検索条件にしています。

次の照会で、

```
SELECT DEPT, NAME, JOB
FROM Q.STAFF
WHERE DEPT = 20
```

以下の報告書が生成されます。

DEPT	NAME	JOB
20	SANDERS	MGR
20	PERNAL	SALES
20	JAMES	CLERK
20	SNEIDER	CLERK

WHERE も HAVING も、報告書にいらないデータを排除する働きをもっています。

## WHERE

- WHERE 条件は列選択に用いられ、個々の行を含めるかどうかを決定します。  
WHERE は、不要な行を除くために使用します。
- HAVING 条件は組み込み関数で使用され、グループそのものを含めるかどうかを決定します。グループそのものを含めるかどうかを決定します。

HAVING には、必ず列関数 (SUM、AVG、MAX、MIN、COUNT など) が続きます。また、グループ値を見つけて HAVING 条件を完成させる副照会が続くことがあります。

HAVING は、不要なグループ・データを除く目的に使用してください。

たとえば、平均給与が \$12,000 を超える各部門について、管理者を除き、その部門の給与の最低額、最高額、平均額をリストするには、次に示すような照会を使用します。

次の照会で、

```
SELECT DEPT, MIN(SALARY),  
       MAX(SALARY), AVG(SALARY)  
  FROM Q.STAFF  
 WHERE JOB < > 'MGR'  
 GROUP BY DEPT  
 HAVING AVG(SALARY) > 12000
```

以下の報告書が生成されます。

DEPT	MIN(SALARY)	MAX(SALARY)	AVG(SALARY)
15	12258.50	16502.83	13756.5100000000
20	13504.60	18171.25	15309.5333333333
38	12009.75	18006.00	14944.7000000000
42	10505.90	18001.75	13338.7500000000
51	13369.80	19456.50	16235.2000000000
66	10988.00	21000.00	16880.1750000000
84	13030.50	17844.00	15443.0000000000

表9に示す演算子を使用して、列値を比較できます。最初の欄に条件、2番目の欄にその比較を行なうための単語または記号を示します。

表9. 比較演算子

比較	使用する比較演算子
等しい	=
等しくない	<> または !=
より大きい	>
より大きいまたは等しい	>=
より大きくなき (DB2 (z/OS 版) のみ)	¬>
より小さい	<
より小さいまたは等しい	<=
より小さくなき (DB2 (z/OS 版) のみ)	¬<
複数条件	AND OR
範囲内の値	BETWEEN x AND y
リスト中の値と一致する値	IN (x, y, z)

表9. 比較演算子(続き)

比較	使用する比較演算子
文字ストリングの選択	% (例: LIKE '%abc%')
特定文字の無視	下線 (例: LIKE '_a_')
否定条件	NOT

否定の記号 ( $\neg$ ) は、データベース管理システム間で受け渡されるステートメントで解析エラーを起こす可能性があります。リモートで実行するステートメントで問題が起こることを避けるため、否定記号が出てくる演算には、同等の別記号を使ってください。たとえば、 $\neg=$  には  $<>$ 、 $\neg>$  には  $<=$ 、および  $\neg<$  には  $>=$  を代わりに使ってください。

文字データの列と比較する値は、単一引用符で囲んでください(たとえば、WHERE NAME = 'JONES' など)。数値データは引用符で囲みません。

漢字データを使用するときは、WHERE に指定する値の頭に 1 バイト文字 'G' を置き、全体を単一引用符で囲みます。パーセント記号と下線には 2 バイト文字を使用します。

WHERE 検索条件には、182 ページの表9 に示す比較演算子をどれでも使用できます。たとえば、\$1,000 以上の歩合を得ている従業員だけを選択するには、次のような照会を使用します。

次の照会で、

```
SELECT ID, COMM
      FROM Q.STAFF
 WHERE COMM >= 1000
```

以下の報告書が生成されます。

ID	COMM
70	1152.00
90	1386.70
340	1285.00

次に、照会の中での WHERE 文節の使い方の例をいくつか示します。

- 勤続年数が 10 年以上の従業員をすべて選択します。

```
SELECT ID, NAME, YEARS
      FROM Q.STAFF
 WHERE YEARS >= 10
```

- 勤続年数が 10 年を超える従業員をすべて選択します。

```
SELECT ID, NAME, YEARS
      FROM Q.STAFF
 WHERE YEARS > 10
```

- 管理者をすべて選択します。

```
SELECT ID, NAME, YEARS
      FROM Q.STAFF
 WHERE JOB = 'MGR'
```

- 英字順で SMITH より後の従業員をすべて選択します。

## WHERE

```
SELECT NAME, ID  
FROM Q.STAFF  
WHERE NAME > 'SMITH'
```

- 部門 10 にいないすべての従業員の名前を、Q.STAFF から選択します。

```
SELECT NAME, ID  
FROM Q.STAFF  
WHERE DEPT < > 10
```

---

## 計算結果

検索条件では、一部に計算値も使用できます。また、選択した各行については、列値を表示する要領で計算値も表示できます。

照会の SELECT 文節または WHERE 文節では、算術式が使えます。

- SELECT 文節の中に式がある場合、報告書に式の計算結果の列が新たに設けられます。
- WHERE 文節の中に式がある場合、その式は検索条件の一部で、報告書の中に列が新しく設けられることはできません。

次の 2 つの照会は、SELECT 文節での算術式の使い方を示しています。

次の照会で、

```
SELECT DEPT, NAME, SALARY  
FROM Q.STAFF  
WHERE DEPT = 38
```

以下の報告書が生成されます。

DEPT	NAME	SALARY
38	MARENGHI	17506.75
38	O'BRIEN	18006.00
38	QUIGLEY	16808.30
38	NAUGHTON	12954.75
38	ABRAHAMS	12009.75

次の照会を実行すると、

```
SELECT DEPT, NAME, SALARY/12  
FROM Q.STAFF  
WHERE DEPT = 38
```

次の報告書が作成されます。

DEPT	NAME	EXPRESSION 1
38	MARENGHI	1458.8958333333
38	O'BRIEN	1500.5000000000
38	QUIGLEY	1400.6916666666
38	NAUGHTON	1079.5625000000
38	ABRAHAMS	1000.8125000000

計算の中で次の算術演算子を使用できます。

+	加算
-	減算
*	乗算
/	除算

式には、列名 (RATE\*HOURS など)、列名と定数 (RATE\*1.07 など)、組み込み関数 (AVG(SALARY)/2 など) を使用できます。数値定数 (3\*7 など) や文字定数 (SALARY + COMM など) からなる式もあります。

表が作成されるとき、表中の各列は、特定タイプのデータを入れる列として定義されます。算術演算は、数値データ・タイプに対してしか実行できません。演算結果は、オペランドのデータ・タイプによって異なることがあります。

例

- 年収(給与と歩合)が\$20,000を超えるすべての従業員の、名前と年収を選択するには、次のような照会を使用します。

```
SELECT NAME, SALARY + COMM  
FROM Q.STAFF  
WHERE SALARY + COMM > 20000
```

上の照会では、給与が \$20,000 を超えていても、歩合がヌルであるような従業員は選択されません。これは、不明に対する演算結果は不明であるためです。

- 歩合が年収の 5% 以上にのぼる従業員をリストするには、次のような照会を使用します。

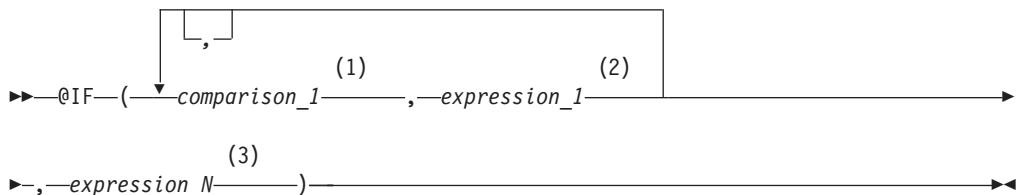
```
SELECT NAME, SALARY, COMM  
FROM Q.STAFF  
WHERE COMM >= 0.05 * (SALARY + COMM)
```

@IF 関数

REXX @IF 関数は、REXX 式内の特定の値を検査するために使用され、関連する REXX 式を解釈して、結果を返します。

@IF 関数は、通常は REXX 式を使用する場所であればどこでも使用できます。

REXX 式は、FORM.CALC、FORM.CONDITIONS、および FORM.COLUMNS（列定義）内で使用できます。



注:

- 0 または 1 を返す有効な REXX 式です。一般に、REXX 比較演算子を指定します。`@IF` 関数は比較を行い、結果が 1 の場合は、関数の次の式が評価され、結果が返されます。`@IF` 関数は、真である比較が見つかるまで、比較を左から右へ評価していきます。真の比較結果が得られない場合は、最後の式が解釈された後に結果が返されます。
  - 有効な REXX 式で、演算子と括弧を含む項目（文字列、記号、および関数）で構成されます。式の前の比較が真の場合、この式が解釈され、結果が返されます。

## WHERE

- 3 有効な REXX 式。比較結果が真でない場合は、*expression\_N* が解釈され、結果が返されます。

### @IF 関数の使用のガイドライン

- 引数は奇数個でなければなりません。
- 引数の最小値は 3、最大値は 19 です。
- 先頭のトークンは @IF でなければなりません。また、直後に左括弧が必要です。
- 複数の引数はコンマで区切る必要があります。
- 引数リストの最後には右括弧が必要です。
- 最後の引数は、「その他の場合」の式、またはデフォルトの式として使用されます。
- 最後の引数が奇数値でない場合は、比較です。
- PASS NULLS が YES に設定され、式に含まれる置換変数がヌル、未定義、オーバーフローになっているか、あるいはインスタンスまたは関係を持たない場合は、式全体がその状態を表す値に設定されます。この縮約は比較ではなく式で行われます。
- PASS NULLS が YES に設定され、式に含まれる複数の置換変数がヌル、未定義、オーバーフローになっているか、あるいはインスタンスまたは関係を持たない場合は、次の優先順位で式が縮約されます。
  - 未定義
  - オーバーフロー
  - 空白
  - インスタンスがない
  - 関係が存在しない

@IF 関数に複数の引数（比較および式）を渡す場合は、@IF 関数のネストが不要になります（ネストされた @IF 関数は式の縮約ではサポートされません）。

SELECT ID, NAME, DEPT, SALARY, COMM FROM Q.STAFF を指定すると、新しい列が次の式で定義され、PASS NULLS が YES に設定されます。

```
@If(&3=10,'MGMT',&5=DSNULL,'N/A',&5/&4*100)
```

この式は、論理的には次のように書き換えることができます。

```
Select
When &3 = 10      Return MGMT      /* All Department 10 are managers */
When &5 is NULL   Return N/A       /* Comission is NULL, mark N/A */
Otherwise         Return &7/&6*100 /* All others, calculate Comm % */
```

結果は次のように表示されます。

ID	NAME	DEPT	SALARY	COMM	%
10	SANDERS	20	18357.50	-	N/A
20	PERNAL	20	18171.25	612.45	3.37
30	MARENghi	38	17506.75	-	N/A
110	NGAN	15	12508.20	206.60	1.65
120	NAUGHTON	38	12954.75	180.00	1.38
160	MOLINARE	10	22959.20	-	MGMT

## SQL スカラー関数

以下では、3 種類のスカラー関数について説明します。

- 日付 / 時刻関数
- 変換関数
- スtring関数

### 日付 / 時刻関数

日時関数は、以下を行います。

- DATE、TIME、TIMESTAMP は、引数のデータ・タイプを関数で指定されたデータ・タイプに変換します。
- CHAR は、引数 (DATE 値または TIME 値) のデータ・タイプを CHAR データ・タイプに変換します。
- DAYS は、ある日付から別の日付までの日数を計算します。
- YEAR、MONTH、DAY、HOUR、MINUTE、SECOND、MICROSECOND は、DATE 値、TIME 値、TIMESTAMP 値からそれぞれ該当する一部を選択します。

各日付 / 時刻関数の後には、引数を括弧つきで指定してください。次の例は、1998 年中の開始が予定されているプロジェクトを選び出し、そのプロジェクト番号をリストします。Q.PROJECT 表の STARTD 列に対して YEAR 関数を実行し、プロジェクトを選択しています。

次の照会で、

```
SELECT PROJNO, STARTD, ENDD, TIMESTAMP
FROM Q.PROJECT
WHERE YEAR(STARTD) = 1998
```

以下の報告書が生成されます。

PROJNO	STARTD	ENDD	TIMESTAMP
1409	1998-06-15	1999-12-31	1996-03-13-09.12.57.149572
1410	1998-09-29	2000-03-31	1996-03-13-12.18.23.402917

日付 / 時刻関数 (表 10 を参照) は、式が使える個所であればどこでも使用できます。どの関数でも、操作すべき値をもたらす式が最初の (または唯一の) 引数となります。

表 10. 日付 / 時刻関数

機能	引数	結果
DATE	日付か、タイム・スタンプか、日付のストリング表現	日付
TIME	時刻か、タイム・スタンプか、時刻のストリング表現	時刻
TIMESTAMP	タイム・スタンプ、タイム・スタンプのストリング表現、または日付 (あるいは日付のストリング表現) と時刻 (または時刻のストリング表現)	Timestamp
DAY、MONTH、YEAR	日付かタイム・スタンプ、または日付期間	日、月、年

## SQL スカラー関数

表 10. 日付 / 時刻関数 (続き)

機能	引数	結果
HOUR、MINUTE、SECOND	時刻かタイム・スタンプ、または時刻期間	時、分、秒
MICROSECOND	Timestamp	マイクロ秒
DAYS	日付か、タイム・スタンプか、日付のストリング表現	0000 年 12 月 31 日以降の日数
CHAR	日付か時刻と、指定の日付 / 時刻出力フォーマット	指定された日付 / 時刻フォーマットでのストリング表現。フォーマットの指定がないときは、ISO フォーマットで返されます。

## 他の変換関数

スカラー関数 (表 11 を参照) では、値のあるデータ・タイプから別のデータ・タイプへ変換できます。

表 11. 変換関数

関数と構文	引数	結果
BIGINT(V)	V = 数値式またはストリング式	V (またはストリング表現の V) の 64 ビット整数表現
BINARY(V, length)	V = ストリング式  length = 結果のストリングの長さを指定する整数	固定長バイナリー・ストリング
DECFLOAT(V,P)	V = 数値式またはストリング式  P = 結果の精度桁数 (16 または 34。デフォルトは 34)	数値 (またはストリング表現の数値) の DECFLOAT 表現
DECIMAL(V,P,S)	V = 数値  P = 結果の精度。1 から 31 の範囲  S = 結果の位取り	V の 10 進表現
DIGITS (引数)	2 進整数または 10 進数	引数の桁数を表す文字ストリング
FLOAT (引数)	数値	引数の単精度浮動小数点表現
HEX (引数)	長文字ストリングと長漢字ストリングを除くデータ・タイプ	引数の実 16 進桁数を表す文字ストリング
INTEGER (引数)	2 進整数範囲内の数値	引数のフルワード表現
VARBINARY(V, length)	V = ストリング式  length = 結果のストリングの長さを指定する整数	可変長バイナリー・ストリング
VARGRAPHIC (引数)	短文字ストリング	漢字ストリング。引数の DBCS 表現です。

次の照会は、表 11 に説明したいいくつかの関数の結果を返します。

次の照会で、

```

SELECT SALARY,          --SALARY
       DECIMAL(SALARY,9,3),   --COL1
       DIGITS(SALARY),      --COL2
       FLOAT(SALARY),        --COL3
       HEX(NAME),           --COL4
       VARGRAPHIC(JOB)      --COL5
  FROM Q.STAFF
 WHERE DEPT = 10

```

以下の報告書が生成されます。

SALARY	COL1	COL2	COL3	COL4	COL5
22959.20	22959.200	2295920	2.295920E+04	D4D6D3C9D5C1D9C5	-M-G-R
20010.00	20010.000	2001000	2.001000E+04	D3E4	-M-G-R
19260.25	19260.250	1926025	1.926025E+04	C4C1D5C9C5D3E2	-M-G-R
21234.00	21234.000	2123400	2.123400E+04	D1D6D5C5E2	-M-G-R

他の使用可能な変換関数の詳細は、IBM Publications Center ([www.ibm.com/shop/publications/order](http://www.ibm.com/shop/publications/order)) の中の、SQL 参照情報を参照してください。

## ストリング関数

表 12 に示したスカラー関数は、ストリング・セグメントの操作と取り出しを行います。

表 12. ストリング関数

関数と構文	引数	結果
LENGTH(引数)	任意のデータ・タイプ	V の桁数を表す整数
SUBSTR(S,N,L)	S: 評価される文字または漢字ストリング。 N: 2 進整数。S 内のサブストリングの開始位置を表します。 L: 2 進整数。サブストリングの長さを表します。	S のサブストリング
VALUE(引数 1,引数 2 )	2 つの引数のデータ・タイプは整合性があること。	引数 1 が非ヌルなら、引数 1 を表す非ヌル値。引数 1 がヌルなら、引数 2 を表す非ヌル値。

桁数関数は、データ・タイプが VARCHAR なら、データの実際長を返します。CHAR なら、固定データ長を返します。

次のステートメントは、Q.INTERVIEW 表にある応募者のうち、管理者 270 の面接を受けた人々の応募状況をリストに書き出します。応募者の DISP 列にデータがないと（したがって、ヌル値が入っていると）、その行の結果はヌル記号 (-) でなく「unknown」となります。

```

SELECT VALUE(DISP, 'unknown')
FROM Q.INTERVIEW
WHERE MANAGER = 270

```

どの関数でも、操作すべき値をもたらす式が最初の（または唯一の）引数となります。 LENGTH では、この式の値はどのデータ・タイプであってもかまいません。 SUBSTR では、文字ストリングか漢字ストリングでなければなりません。 VALUE では、2 つ以上の値を指定しなければならず、そのデータ・タイプは互換性がなければなりません。

たとえば、次の照会では、暫定的な ID 番号として 400 をもつ応募者の、ファーストネームのイニシャルとラストネームを取り出します。

```
SELECT SUBSTR(FIRSTNAME,1,1)||LASTNAME  
FROM Q.INTERVIEW  
WHERE TEMPID = 400
```

## 連結

連結演算子 (CONCAT) は、式の 2 つの値を結合して、1 つのストリングにします。 CONCAT の代替演算子として || も用いられます。縦線は、データベース管理システム間で受け渡されるステートメントで解析エラーを起こす可能性があります。リモートで実行するステートメントには、なるべく CONCAT 演算子を使用してください。

CONCAT 演算子を使用するときは、次の規則に従います。

- 連結演算子の 2 つのオペランドは、ともに文字ストリングか、ともに漢字ストリングでなければなりません。
- 結果の長さは、オペランドの長さの合計になります。
- 結果のデータ・タイプは、次のとおりです。
  - 1 つ以上のオペランドが VARCHAR なら、VARCHAR。
  - 両オペランドがともに CHAR なら、CHAR。
  - 1 つ以上のオペランドが VARGRAPHIC なら、VARGRAPHIC。
  - 両オペランドがともに GRAPHIC なら、GRAPHIC。
- ヌル値のオペランドがあれば、結果もヌル値です。たとえば、  
`VALUE(FNAME, 'unknown') CONCAT VALUE(LNAME, 'unknown')`

FNAME が CHAR(6) で BEN という値を持ち、LNAME が CHAR(8) で JOHNSON という値を持つとき、FNAME CONCAT LNAME の結果は BEN JOHNSON で、長さは 14 となります。(ファーストネームとラストネームの間に、3 個のブランク・スペースが入ります。)

ヌル値を避けるには、VALUE 関数を使用してください。 VALUE の詳細については、189 ページの『ストリング関数』を参照してください。

- LIKE 文節、または UPDATE ステートメントの SET 文節には、連結を指定できません。
- 次の照会は、Q.INTERVIEW から M より大きい文字で始まるすべてのラストネームを検索し出し、対応するファーストネームと組み合わせて書き出します。

```
SELECT LASTNAME CONCAT ', ' CONCAT FIRSTNAME  
FROM Q.INTERVIEW  
WHERE LASTNAME > 'M'
```

## 第 3 章 書式、報告書、および図表

QMF は、データベースに保管されているデータから報告書を作成します。QMF 書式は、報告書のフォーマット設定用に用いられる多くのパネルから構成されています。データを選択するとき（照会を実行するとき、データをインポートするとき、表または視点を表示するとき）には、その QMF 書式パネルを用いて、データを報告書や図表の形に整えることができます。また、報告書データに対して、列の加算や比率の計算など、特定の計算をするときも、書式パネルを使用できます。

このトピックでは、QMF 書式パネルを具体的に示し、各パネル上の入力域について説明します。次のサブトピックが含まれます。

- 『QMF 書式の使用』
- 192 ページの『QMF における報告書の作成』
- 194 ページの『QMF における図表の作成』
- 195 ページの『FORM.MAIN』
- 198 ページの『FORM.BREAKn』
- 206 ページの『FORM.CALC』
- 211 ページの『FORM.COLUMNS』
- 221 ページの『FORM.CONDITIONS』
- 223 ページの『FORM.DETAIL』
- 229 ページの『FORM.FINAL』
- 234 ページの『FORM.OPTIONS』
- 240 ページの『FORM.PAGE』
- 246 ページの『QMF の書式エラーの評価』
- 248 ページの『QMF 書式での REXX の使用』
- 254 ページの『取扱コード』
- 261 ページの『編集コード』
- 269 ページの『書式で使用される変数』

### QMF 書式の使用

表を表示するときや SELECT 照会を実行するとき、とくに書式を指定しないと、QMF が自動的に書式パネルを生成します。結果の報告書は、QMF が提供するデフォルトのフォーマット設定に基づきます。RUN コマンドで書式名を指定せずに照会を実行後、DISPLAY FORM.MAIN (または DISPLAY FORM) と入れれば、デフォルトの書式を見ることができます。

どの書式パネルにも入力域があります。ここで情報の追加や変更を行ないます。この説明では、各書式パネル上の各入力域に識別用の文字 (**C** など) を割り当て、パネルに続く入力域説明との関連づけをはかっています。デフォルト値があるときは、パネル上の入力域にそれを示します。それぞれの入力域を、報告書への影響の点から説明します。図表に作用するときは、続いてそれを説明します。

## QMF における報告書の作成

報告書は、まず、照会結果として得られたデータをデフォルトの書式にはめこむことで作成されます。報告書のデフォルトのフォーマットを変更する（たとえば、列幅を変える、ページ・ヘッダーを追加する、報告書中の行間隔を変えるなど）には、書式パネルに表示されているデフォルト選択を変更してください。プロファイルの CASE オプションを適切に設定しておけば、入力域に入れるデータを大文字に変換できます。

## データのない報告書の表示

LAYOUT コマンドを使用すれば、汎用データの入った報告書を表示できるため、作成中の書式をテストできます。変数データは、英字の A、B、C、D、E、F、と X、および数字 0、1、2、3、4、5、と 6 で表示されます。その他のテキストは、ヘッダーを含め、入力どおりに表示されます。各種書式パネルに手を加えて、データに依存しない報告書の見本を作成しておくとよいでしょう。LAYOUT コマンドと組み合わせれば、複雑な変数を含む書式でも繰り返し使用できます。このコマンドの詳細については、75 ページの『LAYOUT』を参照してください。

## 報告書で用いられるエラー記号

報告書に表示できない値があると、QMF は値の代わりに特殊記号を表示します。表示される記号は、その原因が何であるかによって異なります。記号とその意味のリストについては、表 13 を参照してください。

表 13. QMF エラー記号

表示される記号	原因
*****	列幅が足りないため、フォーマット設定された値を表示できません。数値列でのみ用いられる記号です（文字の列では切り捨てが起こります。）
>>>>>	値が、その列のデータ・タイプで許される最大値を越えています。これはオーバーフローと呼ばれ、普通 QMF によって検出されます。
???????	定義されていない値です。次の状態のときは、報告書に未定義値が現われます。 <ul style="list-style-type: none"> <li>• 数値のアンダーフロー</li> <li>• データベースが検出した数値のオーバーフロー</li> <li>• ゼロでの除算（照会、計算、列定義で）</li> <li>• REXX が評価できない式</li> <li>• 評価すると非数値になる REXX 式</li> <li>• 未定義値を用いて計算された集約値（FIRST と LAST を除きます）</li> </ul>
'' (ブランク)	データにインスタンスがありません (DSQNOINS)。または、関連がありません (DSQNOREL)。

## 報告書のフォーマット設定の早見表

表 14 は、報告書のフォーマットを変えるためによく行われる追加あるいは変更、およびそのとき用いられる書式パネルを示しています。

表 14. 報告書フォーマット設定の早見表

追加または変更が必要な 報告書項目	追加または変更が必要な特定の 要素	使用する書式パネル
報告書の切れ目	省略時切れ目テキスト	MAIN、OPTIONS
	切れ目テキストの幅	OPTIONS
	切れ目見出しテキスト	BREAKn
	切れ目脚注テキスト	MAIN、BREAKn
	切れ目合計	BREAKn
	ページ上での配置	BREAKn
	一括表示	MAIN、OPTIONS
計算	(特定の要素なし)	CALC
報告書の列の仕様	配置	COLUMNS (パネル指定)
	定義	COLUMNS (パネル指定)
	ヘッダー	MAIN、COLUMNS
	使用法	MAIN、COLUMNS
	間隔	MAIN、COLUMNS
	幅	MAIN、COLUMNS
	編集	MAIN、COLUMNS
	並び順	MAIN、COLUMNS
	自動配列	OPTIONS
	切れ目でのヘッダー反復	BREAKn
条件付きフォーマット設定	明細ブロックでのヘッダー反復	DETAIL
	(特定の要素なし)	CONDITIONS
明細ブロック・テキスト	表情報の除去	DETAIL
	表情報の配置指定	DETAIL
	列値へのテキスト組み込み	DETAIL
明細見出しテキスト	(特定の要素なし)	DETAIL
報告書の最終テキスト	ページ上での配置	FINAL
	幅	OPTIONS
	最終合計	FINAL
固定列	(特定の要素なし)	OPTIONS
改ページするかどうか	切れ目	MAIN、BREAKn
	明細ブロック・テキスト	DETAIL
	最終テキスト	FINAL
ページのヘッダーと後書き	(特定の要素なし)	MAIN、PAGE
パネル・バリエーションと 条件との関連づけ	(特定の要素なし)	DETAIL
区切り線	(特定の要素なし)	OPTIONS

表 14. 報告書フォーマット設定の早見表 (続き)

追加または変更が必要な 報告書項目	追加または変更が必要な特定の 要素	使用する書式パネル
明細ブロック間の間隔	(特定の要素なし)	OPTIONS、DETAIL

## QMF における図表の作成

書式パネル上の入力域のなかには、図表ヘッダー、凡例、軸ラベル、X 軸と Y 軸にプロットされるデータなど、図表の内容に関係するものもあります。もちろん、すべてのパネルのすべての入力域が図表に関係するわけではありません。書式パネルの説明 (195 ページの『FORM.MAIN』以降) では、図表に関係するパネルとパネル入力域を指摘し、これらのパネルの変更方法も示します。

表 15 は、QMF 内で図表を変えるためによく行われる追加あるいは変更、およびそのとき用いられる書式パネルを示しています。

表 15. 図表変更の早見表

追加または変更項目	使用する書式パネル
凡例ラベル (Y データ列ヘッダー)	MAIN、COLUMNS
X 軸データ・ラベル	MAIN、COLUMNS
(BREAK 列または GROUP 列)	
Y 軸データ (数値データ列)	MAIN、COLUMNS
図表ヘッダー (ページ・ヘッダー)	MAIN、PAGE
図表ヘッダーの垂直位置	PAGE
凡例ラベル内の関数名	OPTIONS

BINARY、VARBINARY、XML、または DECFLOAT と定義された列を含むデータまたは表の図表は作成できません。

## FORM.MAIN

報告書または図表に簡単な変更を施すときは、FORM.MAIN を使用してください。他のパネル（表 16 を参照）は FORM.MAIN と一緒に、報告書または図表の体裁を変更します。

表 16. 報告書 / 図表の外見変更の手引き

フォーム名	機能	追加情報
FORM.MAIN	報告書または図表の基本書式	『FORM.MAIN』
FORM.BREAK <i>n</i> ( <i>n</i> = 1 から 6)	報告書の切れ目前後のテキスト	198 ページの『FORM.BREAK <i>n</i> 』
FORM.CALC	報告書内の計算式	206 ページの『FORM.CALC』
FORM.COLUMNS	報告書または図表での列の取り扱い	211 ページの『FORM.COLUMNS』
FORM.CONDITIONS	条件つきフォーマット設定のための式	221 ページの『FORM.CONDITIONS』
FORM.DETAIL	列値やヘッダーとともに報告書に組み込まれるテキスト	223 ページの『FORM.DETAIL』
FORM.FINAL	報告書の最終テキストの内容と配置	229 ページの『FORM.FINAL』
FORM.OPTIONS	報告書に加えるその他の調整	234 ページの『FORM.OPTIONS』
FORM.PAGE	報告書または図表におけるページ・ヘッダーとページ後書きの内容と配置	240 ページの『FORM.PAGE』

FORM.MAIN に入力された内容は、自動的に、他のいずれかの書式パネルの関連入力域に反映されます。しかし、他のパネルの入力域に入れられた内容が、必ず FORM.MAIN に反映されるとは限りません。

196 ページの図 15 は、FORM.MAIN パネルの入力域を示しています。FORM.MAIN パネルと FORM.COLUMNS パネルには、入力域でない区域が 2 つあります。それは、報告書列の幅の合計と NO. 域です。

## FORM.MAIN

FORM.MAIN						
COLUMNS:		Total Width of Report Columns: 42				
A	B	C	D	E	F	
NUM	COLUMN	HEADING	USAGE	INDENT	WIDTH	EDIT SEQ
1	ID			2	6	L 1
2	NAME			2	9	C 2
3	DEPT			2	6	L 3
4	JOB			2	5	C 4
5	YEARS			2	6	L 5
PAGE: HEADING ==> <b>G</b>						
FOOTING ==>						
FINAL: TEXT ==> <b>H</b>						
BREAK1: NEW PAGE FOR BREAK? ==> NO						
FOOTING ==> <b>I</b>						
BREAK2: NEW PAGE FOR BREAK? ==> NO						
FOOTING ==> <b>J</b>						
OPTIONS: OUTLINE? ==> YES				DEFAULT BREAK TEXT? ==> YES		
1=Help	2=Check	3=End	4>Show	5=Chart	6=Query	
7=Backward	8=Forward	9=	10=Insert	11=Delete	12=Report	
OK, cursor positioned.						
COMMAND ==>						
SCROLL ==> PAGE						

図 15. FORM.MAIN の入力域

入力域 **A** から **F** は、FORM.COLUMNS パネル上の同名の各入力域に対応します。（詳細については、211 ページの『FORM.COLUMNS』を参照してください。）書式に含まれているすべての列が FORM.MAIN パネル上に見えなくても、画面を前後にスクロールすれば見られます。

これらの入力域で、次のことができます。

- A** 列ヘッダーを割り当てる。
- B** 列の処理方法を選択する。
- C** 列の字下げを調整する。
- D** 列の幅を調整する。
- E** 列のフォーマット設定を指定する。
- F** 列の順序を変更する。

**報告書:** 書式における列の並び順は、照会の SELECT ステートメントで指定された順序で決まります。報告書での列の並び順を変えるには、自動再配列オプションを使用するか、「FORM.MAIN」パネルの順序 (SEQ) 列 (**F**) を変更します。自動再配列オプションについては、234 ページの『FORM.OPTIONS』を参照してください。

**図表:** これらの入力域のうち、COLUMN HEADING (列ヘッダー)、USAGE (取扱)、WIDTH (列幅)、および EDIT (編集) の 4 つは、図表にも適用されます。取り扱いの入力域に表示されるコードは、処理に影響します。詳細については、211 ページの『FORM.COLUMNS』、254 ページの『取扱コード』、261 ページの『編集コード』を参照してください。

入力域の **G** から **J** までには、対応する書式パネルがあります。

**G PAGE**

**報告書:** 報告書のページ・ヘッダーおよび後書きテキストを 1 行で入力します。QMF が、ヘッダー行と後書き行の垂直位置、水平位置を決めます。「PAGE (ページ)」入力域は、FORM.PAGE パネル上の 2 つの入力域に対応しています。

**図表:** PAGE (ページ) 入力域に報告書ヘッダーとして現れる内容は、図表にもヘッダーとして現れます。図表には後書きテキストを指定できません。

PAGE (ページ) 域で制御されるフォーマット設定の詳細は、240 ページの『FORM.PAGE』を参照してください。

**H FINAL**

**報告書:** 1 行の、報告書の最終テキストを入力します。行のデフォルト配置は、FORM.FINAL パネルで変更できます。「最終テキスト」項目は、FORM.FINAL パネルの 1 つの項目に対応しています。

FINAL (最終テキスト) 域で制御されるフォーマット設定の詳細は、229 ページの『FORM.FINAL』を参照してください。

**I BREAK1 および BREAK2**

**報告書:** 2 レベルまでの制御の切れ目用の後書きテキストを入力し、指定された切れ目列の値が変わるたびに改ページするかどうかを指定します。

QMF が、制御の切れ目の後書きの垂直位置、水平位置を決めます。

「BREAK 1」入力域と「BREAK 2」入力域は、それぞれ FORM.BREAK1 パネルと FORM.BREAK2 パネルの入力域に対応しています。

これらの区域で制御されるフォーマット設定の詳細は、198 ページの『FORM.BREAKn』を参照してください。

**J OPTIONS**

**報告書:** 制御の切れ目のある報告書では、QMF は報告書の表データ行に切れ目列の値を表示させるかどうかを、「OUTLINE」オプションで指定します。YES なら、切れ目列の値自体が変わったとき (のみ)、その値を表示します。

制御の切れ目のある報告書では、BREAK 総計行の目印にデフォルトの切れ目後書きテキストを入れるかどうかを、「DEFAULT BREAK TEXT」オプションで指定します。切れ目後書きテキストを入れずに、ここを YES としておくと、デフォルトの切れ目後書きとして一連のアスタリスクが表示されます。

この入力域は、FORM.OPTIONS パネル上の 2 つの入力域に対応します。(詳細については、234 ページの『FORM.OPTIONS』を参照してください。)

**報告書列の幅の合計**

**報告書:** この区域には、報告書の各列の幅 (文字数) の合計が示されます。

この区域を直接変更することはできませんが、ある列の間隔、列幅、編集コードのいずれかを変えると (あるいは、取扱コードとして OMIT または ACROSS を使用すると)、報告書の各列の幅の新しい合計 (文字数) が、コロンの後に示されます。

DBCS データに編集コードの G を使用すると、1 個の 2 バイト文字が 2 文字と数えられます。DBCS データを含む列の幅の計算については、「DB2 QMF 使用の手引き」を参照してください。

## FORM.MAIN

取扱コードの ACROSS を使用すると、幅は  $a + (N \times b)$  という形の代数式で表示されます。

*a* 定数値

*N* 1 組の列をページの横方向に何回繰り返すか (ACROSS 列の異なる値ごとに 1 回) を表す未知数

*b* 各列グループの幅

**NUM 報告書:** この区域には、実行された照会での選択順序で各列の番号が示されます。この区域は変更できませんが、「順」入力域を用いて列の並び順を変えることができます。

列番号を使用して、置換変数として使用する列を指示できます。たとえば、&6 は、報告書で 6 番目に並ばなくとも照会で選択された 6 番目の列を指します。

報告書では、普通、各列が左から右へ順序番号どおりに並びます。しかし、BREAK、GROUP を使用するか、FORM.MAIN または FORM.COLUMNS で集約関数を使用し、FORM.OPTIONS の報告書列の自動並べ換えに YES を指定すると、QMF は報告書の列を自動的に並べ替えます。

列の自動並び換えを行なうとき、取扱コードとして 1 つ以上の BREAK コードを使用すると、制御列が報告書の左端へ移動します。BREAK コード番号順に並べられます。

さらに、取り扱いがいざれかの集約取扱コード (AVERAGE、COUNT、FIRST、LAST、CALC*id*、MAXIMUM、MINIMUM、STDEV、SUM、CPCT、CSUM、PCT、TPCT、TCPCT) であるような列は、報告書の右端に移され、そこに列番号順に並べられます。

列の幅と並び順の詳細は、234 ページの『FORM.OPTIONS』の『**C** 報告書のテキスト行幅』と『**J** 報告書列の自動再配列』を参照してください。

---

## FORM.BREAKn

報告書には 6 つまでの制御の切れ目を設けることができます。テキストとその配置についての選択には、FORM.BREAK*n* (*n* は 1 から 6 の数字) パネルを使用してください。QMF は、報告書の対応する制御の切れ目の後に、各切れ目パネルで指定されたテキストを配置します。

FORM.BREAK*n* パネルは、図表には影響しません。

FORM.MAIN または FORM.COLUMNS の列名と対になっている「取扱」入力域 (**B**) に切れ目取扱コードを指定してください (詳細は、195 ページの『FORM.MAIN』と 211 ページの『FORM.COLUMNS』を参照)。指定すると、その列が制御列 になります。この制御列の値が変わったときに必ず、報告書の制御の切れ目が起ります。

VARCHAR 列の値を評価する場合に、QMF は、末尾をブランクや 16 進数のゼロで埋められた値と、それらの後続文字をもたない値とを (値自体は同じでも) 区別します。そのような場合に FORM.BREAK*n* パネルを使用すると、制御の切れ目が起ります。

複数の列に同一レベルの制御の切れ目を使用してしまいません。その場合には、これらの列のどれかに値の変更があると、制御の切れ目が起こります。

FORM.MAIN の区域 **I** には、報告書における切れ目 1 と切れ目 2 の後書きテキストを指定し、制御列の変化するたびに改ページするかどうかを指定します。

FORM.MAIN の区域 **I** への指定は、すべて FORM.BREAK1 と FORM.BREAK2 に反映されます。 FORM.BREAK1 と FORM.BREAK2 の区域 **H** と **N** への指定は、FORM.MAIN に反映されます。

制御の切れ目の各レベルに 1 つずつ、最大 6 つの FORM.BREAK パネルがあります。いずれも、パネルの表題以外は同内容です。 図 16 は、FORM.BREAK パネルの入力フィールドを示しています。

<b>FORM.BREAK1</b>												
<b>A</b>	New Page for Break?	====> NO	<b>B</b>	Repeat Detail Heading?	====> NO							
<b>C</b>	Blank Lines Before Heading	====> 0	<b>D</b>	Blank Lines After Heading	====> 0							
<b>E</b>	LINE	<b>F</b>	ALIGN	<b>G</b>	BREAK1 HEADING TEXT							
						-----	-----+-----1-----+-----2-----+-----3-----+-----4-----+-----5-----+					
1	LEFT											
2	LEFT											
3	LEFT											
*** END ***												
<b>H</b>	New Page for Footing?	====> NO	<b>I</b>	Put Break Summary at Line	====> 1							
<b>J</b>	Blank Lines Before Footing	====> 0	<b>K</b>	Blank Lines After Footing	====> 1							
<b>L</b>	LINE	<b>M</b>	ALIGN	<b>N</b>	BREAK1 FOOTING TEXT							
						-----	-----+-----1-----+-----2-----+-----3-----+-----4-----+-----5-----+					
1	RIGHT											
2	RIGHT											
3	RIGHT											
*** END ***												
1=Help	2=Check	3=End	4>Show	5=Chart	6=Query							
7=Backward	8=Forward	9=	10=Insert	11=Delete	12=Report							
OK, FORM.BREAK1 is displayed.												
COMMAND ===>												
SCROLL ===> PAGE												

図 16. FORM.BREAK パネルの入力フィールド

#### **A 切れ目で改ページ**

切れ目の制御列の値が変化するたびに改ページを開始するかどうかを指定します。この値は印刷報告書およびエクスポート報告書に影響します。これは表示報告書には影響しません。改ページは、報告書がまだページの上端にない場合に開始します。

1 つの切れ目レベル以上で YES と指定すると、印刷報告書またはエクスポート報告書で期待以上のページを生成します。これは、複数の切れ目が同時に起こる場合に生じます。

2 つ以上の切れ目を指定し、さらに各切れ目で切れ目で改ページに YES を指定すると、最高位の切れ目が生じるたびにそれぞれ指定された切れ目にについてページが生成されます。複数の切れ目がしばしば一緒に生じます。最高位の切れ目がすべての低位切れ目レベルを生じさせるからです。すべての切れ目は報告書のデータの第 1 行で生じます。

**B 明細ヘッダーの反復**

新しい切れ目レベルが始まるたびに、切れ目ヘッダー・テキストの後、かつ明細ブロック・テキストの前に、明細ヘッダーを繰り返すかどうか指定します。

印刷された報告書では、制御の切れ目がページの上端で始まり、かつ、ここに YES が指定されていると、1 組の明細ヘッダーしか現われません。

明細ヘッダーは、FORM.DETAIL パネルで指定した明細ヘッダー・テキストと、列ヘッダー (FORM.DETAIL パネルで列ヘッダーを抑止しない場合) から構成されます。詳細については、223 ページの『FORM.DETAIL』を参照してください。

FORM.DETAIL で明細ヘッダーの反復に YES を指定すると、それがこの指定に優先します。

**C ヘッダー前のブランク行数**

切れ目ヘッダー・テキストの最初の行より前にあるブランク行数 (指定がある場合)、または最初の切れ目メンバー行より前にあるブランク行数 (切れ目ヘッダー・テキストがない場合) を入力します。この値には、0 から 999 までの任意の数値を指定できます。

**D ヘッダー後のブランク行数**

切れ目ヘッダー・テキストの最後の行より後にあるブランク行数を入力します (指定されている場合)。この項目は、0 から 999 までの任意の数値が使用できます。

**E 行**

切れ目ヘッダー・テキストの各行を識別するとともに、それらの行自体から

と切れ目ヘッダー開始行 (ヘッダー前のブランク行数入力域で指示) からの相対的な位置を指定します。1 から 999 までの任意の数値またはブランクを指定できます。ブランクの場合は、QMF は対応するテキストがあっても無視します。

指定する数値は、それぞれ必ずしも 1 から始まる必要はなく、連続している必要もありません。

たとえば、FORM.BREAK1 に次の値があるとします。

```
LINE ALIGN BREAK1 HEADING TEXT
----- -----
3   LEFT   DEPARTMENT &4
2   LEFT   BEGINNING OF LISTING
```

これらの値は次のように表示されます。

```
BEGINNING OF LISTING
DEPARTMENT 35
```

テキスト第 1 行の前に 1 行のブランク行が置かれることに注意してください。

**F 調整**

報告書上での切れ目ヘッダー・テキスト各行の水平位置を指定します。報告書の幅の範囲内であれば、どこでも行を位置づけることができます。ここでの幅は、オンライン報告書の場合は、表示される報告書の幅であり、印刷報告書の場合は、ページの幅になります。

**Left**

切れ目ヘッダー・テキストを左寄せにします。

<b>Right</b>	切れ目ヘッダー・テキストを右寄せにします。
<b>Center</b>	切れ目ヘッダー・テキストを中央に合わせます。
<i>n</i>	切れ目ヘッダー・テキストを、その行の <i>n</i> 桁目 から開始します。ここで、 <i>n</i> は、1 から 999999 の任意の数値を指定できます。

**Append**

その行を、前の切れ目ヘッダー・テキスト行の終わりに付加します。 APPEND を切れ目ヘッダー・テキストの最初の行に使用すると、テキスト行は左寄せになります。

付加されるテキスト行は、付加先のテキスト行と同じ「行」値をもっていなければなりません。 報告書の幅が不十分で、付加されたテキスト行の全体が入りきらないと、その一部が切り捨てられますことがあります。

たとえば、FORM.BREAK1 に次のように入力されたとします。

```
Blank Lines Before Heading ===> 0
LINE  ALIGN   BREAK1 HEADING TEXT
----- -----
1    LEFT     DEPARTMENT
1    APPEND   &4
3    LEFT
```

これらの値により、作成される報告書では、列の配置が次のようになります。

DEPT	COMM	JOB	SALARY
DEPARTMENT 66			
66	55.50	CLERK	10988.00
	-	MGR	18555.50
	844.00	SALES	16858.20
	200.30	SALES	21000.00
	811.50	SALES	18674.50
		*	86076.20
DEPARTMENT 84			
84	188.00	CLERK	13030.50
	-	MGR	19818.00

**G 切れ目 1 のヘッダー・テキスト**

制御の切れ目に対応するヘッダー・テキストを入力します。列の値が変わったびに、この項目に指定するテキストが報告書に表示されます。 INSERT コマンドで、最大 999 行の切れ目ヘッダー・テキストを追加できます。 テキストの各行の長さは、最大 55 文字です。「調整」(ALIGN) 値に APPEND を指定するか、具体的な水平位置を指定すれば、その行にテキストを追加できます。

使用しているサイトが DBCS データをサポートする場合は、272 ページの『2 バイト文字を含む名前』を参照してください。

デフォルトでは、切れ目ヘッダー・テキストは、報告書の左マージンから右マージンまで広がります。ただし、切れ目ヘッダー・テキストの幅は、 FORM.OPTIONS の報告書のテキスト行幅項目で選択できます（詳細は、234 ページの『FORM.OPTIONS』を参照してください）。

## FORM.BREAKn

報告書の切れ目ヘッダー・テキストに大文字と小文字の両方を使いたいときは、プロファイルの CASE 値に STRING か MIXED を指定してください。

### STRING

切れ目ヘッダー・テキストは入力したとおりに表示しますが、その他の入力は大文字に変換します。

### MIXED

入力したとおりにすべての入力を表示します。

切れ目ヘッダー・テキストには、次の変数を入れることができます。

### 大域変数

SET GLOBAL を使用して、切れ目ヘッダー・テキストに使う変数をセットします。このコマンドの詳細については、123 ページの『SET GLOBAL』を参照してください。

**&n** *n* は、この報告書に使用される書式の列 *n* の現在行を表す番号です。列 *n* は、必ずしも報告書上の *n* 番目の列であるとは限りません。FORM.MAIN と FORM.COLUMNS に示されている *n* 番目の列を表します。たとえば、切れ目ヘッダー・テキスト BEGINNING OF DEPARTMENT &3 は、報告書に次のような行を表示します。

BEGINNING OF DEPARTMENT 38

切れ目ヘッダー・テキストでは、DATE 値、TIME 値、TIMESTAMP 値と共に次の変数も使用できます。

### &DATE

現在日付は、使用しているサイトのデフォルトに従ってフォーマット設定されます。デフォルトのフォーマットは、次の日付形式の 1 つに準じます。

- USA (米国)
- EUR (ヨーロッパ)
- ISO (国際標準化機構)
- JIS (日本工業規格)
- 使用しているサイトで定めた代替日付形式

### &TIME

現在時刻は使用しているサイトのデフォルトに従ってフォーマット設定されます。デフォルトのフォーマットは、&DATE のところでリストした形式の 1 つに準じます。

### &PAGE

ページ番号は、報告書をフォーマット設定する時点でページごとに印刷されます。

報告書のページが、プリンターの幅、またはユーザー・プロファイルで指定したデフォルトの印刷幅のどちらかより広い場合には、QMF がそのページを分割します。分割されたどのページ部分も同じページ番号になりますが、添え字がつきます。(DBCS 使用中に

QMF がページを分割すると、報告書の 2 番目以降のページから印刷が続けられますが、開始位置はそのページの左端から 4 バイト目になります。)

#### &ROW

現行の制御の切れ目レベル内にある最初のデータ行の番号が報告書に印刷または表示されます。

#### H 改ページ後の後書き

報告書を印刷するとき、指定された切れ目後書きテキストを表示する前に改ページするかどうかを指定します。改ページは、報告書がまだページの上端にない場合に開始します。

#### I 切れ目合計行の位置

切れ目合計のフォーマットを設定するかどうか、設定する場合には、切れ目後書きテキスト行に対してどこに位置づけるか、を指定します。指定できる値は 1 から 999 までか NONE (切れ目合計なし) です。

#### J 後書き前のブランク行数

切れ目後書きテキストの最初の行の前のブランク行数を指定します。指定できる値は 0 から 999 までか BOTTOM です。

#### K 後書き後のブランク行数

切れ目後書きテキストの最終行のブランク行数を指定します。指定できる値は 0 から 999 までです。

制御の切れ目を指定し、しかも 列内折り返しを行なう列の取扱コードが FIRST、LAST、MIN、MAX のいずれかの場合、このフィールドの値を大きめにしておかないと、切れ目合計中のすべての折り返し行を見られないことがあります。列内折り返しの詳細については、262 ページの『文字データの編集コード』の CW 項目を参照してください。

#### L 行 切れ目ヘッダー・テキストの各行を識別すると共に、それらの行自体からと切れ目後書きテキスト開始行 (後書き前のブランク行数 入力域で指示) からの相対的な位置を指定します。1 から 999 までの任意の数値またはブランクを指定できます。ブランクの場合は、対応するテキストは無視されます。

指定する数値は、それぞれ必ずしも 1 から始まる必要はなく、連続している必要もありません。

たとえば、FORM.BREAK1 に次の値があるとします。

```
LINE ALIGN BREAK1 FOOTING TEXT
----- -----
3   LEFT   DEPARTMENT &4
2   LEFT   END OF LISTING
```

これらの値は報告書の中に次のように表示されます。

```
END OF LISTING
DEPARTMENT 35
```

#### M 調整

報告書上で切れ目後書きテキスト各行の水平位置を指定します。切れ目合計のない制御の切れ目では、報告書の幅の範囲内であれば、どこにでも切れ目後書きテキスト行を位置づけることができます。報告書全体の幅は、FORM.MAIN の一番上に示されています。

## FORM.BREAKn

取扱コード (OMIT、BREAKn、GROUP、ACROSS を除く) で切れ目合計を作る制御の切れ目では、左マージンより右、合計データの左端の桁と関連づけられている列間隔域の始まりより左のどこかに、QMF が切れ目後書きテキスト行を配置します。

<b>Left</b>	切れ目後書きテキストを左寄せにします。
<b>Right</b>	切れ目後書きテキストを右寄せにします。
<b>Center</b>	切れ目後書きテキストを中心寄せに配置します。
<i>n</i>	切れ目後書きテキストを、その行の <i>n</i> 桁目 から開始します。ここで、 <i>n</i> は、1 から 999999 の任意の数値を指定できます。
<b>Append</b>	その行を、前の切れ目後書きテキスト行の終わりに付加します。 APPEND が他の行に付加されないテキスト行に使用される場合、テキストの行は左寄せされます。

付加されるテキスト行は、付加先のテキスト行と同じ「行」値をもっていかなければなりません。 報告書の幅が不十分で、付加されたテキスト行の全体が入りきらないと、その一部が切り捨てられることがあります。

たとえば、FORM.BREAK1 に次のように入力されたとします。

```
LINE ALIGN BREAK1 FOOTING TEXT
---- -----
1   RIGHT  TOTAL
1   APPEND SALARIES--DEPT. &4;
3   RIGHT
4   RIGHT
5   RIGHT
```

これらの値によって、報告書の中で列は次のように配置されます。

DEPT	COMM	JOB	SALARY
66	55.50	CLERK	10988.00
	-	MGR	18555.50
	844.00	SALES	16858.20
	200.30	SALES	21000.00
	811.50	SALES	18674.50
TOTAL SALARIES--DEPT.	66		86076.20
84	188.00	CLERK	13030.50
	-	MGR	19818.00
	806.10	SALES	15454.50
	1285.00	SALES	17844.00
TOTAL SALARIES--DEPT.	84		66147.00

### N 切れ目 1 の後書きテキスト

制御の切れ目に対応する後書きテキストを入力します。列の値が変わるたびに、この項目に指定するテキストが報告書に表示されます。 INSERT コマンドで、最大 999 行の切れ目後書きテキストを追加できます。テキストの各行の長さは、最大 55 文字です。「調整」(ALIGN) 値に APPEND を指定するか、具体的な水平位置を指定すれば、その行にテキストを追加できます。

使用しているサイトが DBCS データをサポートする場合は、272 ページの『2 バイト文字を含む名前』を参照してください。

デフォルトでは、切れ目後書きテキストは、報告書の左マージンから、切れ目合計の始まり（ある場合）または報告書の右マージンのどちらかまで広がります。ただし、切れ目後書きテキストの幅は、報告書のテキスト行幅項目で選択できます（234 ページの『FORM.OPTIONS』に説明されています）。

報告書の切れ目後書きテキストに大文字と小文字の両方を使いたいときは、プロファイルの CASE 値に STRING か MIXED を指定してください。

#### **STRING**

切れ目後書きテキストは入力したとおりに表示しますが、その他の入力は大文字に変換します。

#### **MIXED**

入力したとおりにすべての入力を表示します。

切れ目後書きテキストには、次の変数を入れることができます。

#### **大域変数**

SET GLOBAL を使用して、切れ目後書きテキストに使う変数をセットします。詳細については、123 ページの『SET GLOBAL』を参照してください。

**&n** *n* は、この報告書に使用される書式の列 *n* の最新値を表す番号です。列 *n* は、必ずしも報告書上の *n* 番目の列であるとは限りません。データベースから選択された *n* 番目の列、または FORM.MAIN と FORM.COLUMNS に示されている *n* 番目の列を表します。

たとえば、切れ目後書きテキスト END OF DEPARTMENT &3 は、報告書に次のような行を表示します。

END OF DEPARTMENT 38

#### **&COUNT**

同レベルの最後の制御の切れ目から後に表示または印刷された行数。この値は、データ行ごとに加算されます。

#### **&ROW**

最終データ行の番号が報告書に印刷または表示されます。

#### **&CALCid**

計算値

#### **&DATE**

現在の日付

#### **&TIME**

現在の時刻

#### **&PAGE**

現行ページ番号

&CALCid については、206 ページの『FORM.CALC』を参照してください。

## FORM.BREAKn

このトピックの前の方で説明した &DATE、&TIME、および &PAGE の説明を参照してください。

**&an** *n* は有効な列番号です。*a* は QMF 集約関数で、AVG、COUNT、CPCT、CSUM、FIRST、LAST、MAX、MIN、PCT、STDEV、SUM、TCPCT、TPCT のいずれかです。総計値は、現切れ目レベル内の通しの値に基づきます。

たとえば、報告書の 4 番目の列が給与であり、切れ目後書きテキストで各グループの給与合計を出したいとします。次の切れ目 1 の後書きテキストを入力します。

```
TOTAL SALARY FOR DEPARTMENT &3 IS &SUM4
```

報告書の切れ目後書きテキストは、次のようにになります。

```
TOTAL SALARY FOR DEPARTMENT 38 IS $77,285.55
```

切れ目後書きテキストの中に総計変数を指定すると、その列の取り扱いとして同じ集約関数を指定する必要はありません。しかし、指定される集約関数は、編集コードや列のデータ・タイプと整合性がなければなりません。たとえば、列 3 のデータが文字編集コードをもつときに、最終テキストで &SUM3 を指定することはできません。

パーセンテージ集約変数 (PCT、TPCT、または TCPCT) を切れ目後書きテキストの中で使用し、それに、D 編集コードをもつ列を関連付けると、QMF は、そのパーセンテージの値を L 編集コードをもつ場合と同じようにフォーマット設定します。同様に、STDEV (標準偏差) 集約変数を使用して、それを P または D 編集コードをもつ列と関連づけると、QMF は、L 編集コードをもつ場合と同じように標準偏差をフォーマット設定します。

詳細については、264 ページの『数値データの編集コード』および 269 ページの『書式で使用される変数』の L コードを参照してください。

---

## FORM.CALC

### CICS ユーザーへの注

FORM.CALC は、REXX で書かれた式を使用します。CICS では REXX を使用できません。

FORM.CALC パネルには、報告書の計算の式を入力することができます。このパネルには最初は 1 行 (つまり 1 つの式を入力できる分) しか含まれていません。しかし、最大 998 行を追加 (挿入) することができます。

それぞれの入力域を、報告書への影響の点から説明します。FORM.CALC は、図表に影響を与えません。

図 17 は、FORM.CALC パネルの入力フィールドを示しています。

FORM.CALC		<b>C</b> PASS NULLS?	<b>D</b> For &CALCid WIDTH	<b>E</b> EDIT
<b>A</b> ID	<b>B</b> CALCULATION EXPRESSION			
		NO	10	C
*** END ***				
1=Help 7=Backward OK, Cursor positioned. COMMAND ==>	2=Check 8=Forward 9= 10=Insert 11=Delete 12=Report	3=End 4>Show 5=Chart 6=Query	SCROLL ==>	PAGE

図 17. FORM.CALC パネルの入力フィールド

**A ID** 各計算式を指し示すための 1 から 3 文字の ID を入れてください。ID には、1 から 999 までの任意の番号を指定できます。 CALC 取扱コード (254 ページの『取扱コード』を参照) または &CALC 変数に追加すると、 FORM.CALC のどの式を計算に使用するかを指定できます。

&CALCid 変数は、明細ブロック・テキスト、最終テキスト、および切れ目後書きテキストでのみ使用することができます。 CALCid 取扱コードと &CALCid 変数は、id に等しい ID をもつ FORM.CALC の計算式の評価を活動化します。

&CALC 変数では、評価結果が、FORM.CALC パネルの中の式に指定されている幅と編集コードに従って編集されます。 CALCid 取扱コードでは、評価結果が、列幅と CALC の編集コードに従って編集されます。

#### **B 計算式**

式を入力します。最大 50 文字を入れることができます。 FORM.CALC で使用する REXX プログラム内から、QMF コマンドを実行する (呼び出し可能インターフェースまたはコマンド・インターフェースを使用して) ことはできません。

&CALCid 以外は、式で任意の有効な書式変数を使用することができます。次の変数が有効です。

#### 大域変数

SET GLOBAL を使用して、計算式で使用する変数をセットします。このコマンドの詳細については、123 ページの『SET GLOBAL』を参照してください。

#### 列変数: &n

n は有効な列番号です。

#### 総計変数: &a

n は有効な列番号です。a は QMF 集約関数で、AVG、COUNT、CPCT、CSUM、FIRST、LAST、MAX、MIN、PCT、STDEV、SUM、TCPCT、TPCT のいずれかです。

**&ROW**

計算実行時のデータ行の番号を印刷します。&ROW 変数は、&CALCid 変数または CALC 取扱コードの評価の直前に置き換えられます。

**&COUNT**

行数

**&DATE**

現在の日付

**&TIME**

現在の時刻

**&PAGE**

現行ページ (表示される報告書では常に 1)

&COUNT、&DATE、&TIME、および &PAGE の詳細な説明が 198 ページの『FORM.BREAKn』にあります。

式が入力されると、その変数の妥当性が検査されます。列変数については、列番号が正しいかどうか、取扱コードまたは編集コード (またはその両方) が整合性があるかどうか、が検査されます。算術式内に非数値の置換変数があっても、QMF はそれを検出しないため、置換変数には、必ず式と互換性のあるものを使用してください。たとえば、6 番目の列の編集コードが C の場合に式で &SUM6 を使用すると、エラーになり、メッセージが出されます。

式に構文エラーがある場合、REXX プログラム自体の中で、または REXX 式の中でそのエラーを訂正する必要があります。以下の REXX コーディング規則に従ってください。 IBM Publications Center ([www.ibm.com/shop/publications/order](http://www.ibm.com/shop/publications/order)) を参照して、REXX 情報を表示してください。

たとえば、存在しないプログラム名を式に含めたとします。プログラム名を訂正するか、プログラムを作成したあと、SHOW F.CALC を入力し、必要な変更を行います。他に変更する必要がなければ、その式で文字の 1 つを再タイプします。これにより、QMF は変数を再度有効化し、ご自身の書式を正しく構成するようにします。ご自身の書式を再有効化しない場合、予想外の結果となる場合があります。

**C NULL 置換**

YES または NO を入力します。

**YES** 表 17 に示した QMF に用意されている値を用いて、それぞれの状況における値の処理を変更できます。

表 17. データベースのヌル値のデフォルトの取り扱い

状況	値と置換される文字ストリング
データは空白	DSNULL
データが未定義である	DSUNDEF
データに数値オーバーフローがある	DSQOFLOW
データにインスタンスがない	DSQNOINS
データに関連がない	DSQNOREL

**NO**

上記の値にヌルを返します。REXX には、評価のための値を何も渡しません。

たとえば、ヌルであるデータベース変数(データベース・ヌル)はいずれも、式を REXX に渡して評価する前に、文字ストリング DSQNULL と置き換えられます。REXX の式またはプログラムを用意すれば、ストリングを検査して、データベース・ヌルを 0(またはユーザーの目的に合った値)に置き換えることができます。

REXX 式からヌル値が返されたとき、それを報告書に渡すことができます。

式に含まれる置換値がヌル、未定義、オーバーフローになっているか、あるいはインスタンスまたは関係を持たない場合は、式全体がその状態を表す値に設定されます。この縮約は比較ではなく式で行われます。

式に含まれる複数の置換値がヌル、未定義、オーバーフローになっているか、あるいはインスタンスまたは関係を持たない場合は、次の優先順位で式が縮約されます。

1. 未定義
2. オーバーフロー
3. 空白
4. インスタンスがない
5. 関係が存在しない

詳細については、185 ページの『@IF 関数』を参照してください。

**D 列幅**

対応する式の評価結果を報告書テキスト内で編集する幅(1 バイト文字で)を入力します。これは、&CALCid 変数について得られた結果にだけ適用されます。CALCid 取扱コードを、その列の編集コードに基づいて編集できないときは、CALCid の編集コードが用いられます。

幅は 5 文字の入力フィールドです。1 から 32,767 の数値を入力してください。デフォルト値は 10 です。

**E 編集**

対応する式の評価結果を報告書テキスト内で編集する場合に使用する編集コードを入力します。これは、&CALCid 変数について得られた結果にだけ適用されます。CALCid 取扱コードの結果の編集には、FORM.MAIN または FORM.COLUMNS で当該列に対して指定された編集コードが用いられます。

編集は 5 文字のフィールドです。FORM.COLUMNS に行が挿入された場合、デフォルトは、C(文字データ用)です。210 ページの表 18 に示す編集コードのみ使用できます。

## FORM.CALC

表 18. FORM.CALC で使用できる編集コード

フォーマット対象のデータ・タイプ	FORM.CALC で使用できる編集コード	結果または使用法
数値	D, E, I, J, K, L, P	これらの数値編集コードにはオプションのサフィックスを使用できます。Z は、すべての数値編集コードに対するオプションのサフィックスで、ゼロ値の抑止に使用できます。C は、D 編集コードに対するオプションのサフィックスであり、デフォルトの通貨記号ではなく DSQDC_CURRENCY 大域変数で指定した通貨記号を QMF で使用することを示します。E を除き、任意の数値編集コードには 0 ~ 99 の 10 進数位取り値を不可することができます。
文字	C	文字編集 (デフォルト値)
ユーザー定義	Uxxxx, Vxxxx	数値編集または文字編集のためのユーザー編集コード。Uxxxx および Vxxxx 編集コードの定義と使用の詳細は、「 <i>TSO/CICS 用 DB2 QMF インストールおよび管理の手引き</i> 」を参照してください。

表 19 は、編集コードを式に適用した場合に戻される結果を要約したものです。計算の編集コードに関する詳細については、261 ページの『編集コード』を参照してください。

表 19. 編集コードの要約

ユーザーの式の結果	適用可能な編集コード	編集結果	
数値	数値	編集コードによる編集	
	非数値	編集コードによる編集結果の文字表現	
	Uxxxx, Vxxxx	ユーザー編集ルーチンによる編集 (Uxxxx の式の結果が拡張浮動小数点データとしてルーチンに渡されます)	
非数値	数値	C (文字) として編集	
	非数値	Cxx	文字
		Uxxxx, Vxxxx	ユーザー編集ルーチンによる編集

注: COBOL では、数値データの最初の 8 バイトの長精度浮動小数点フォーマットで十分な精度が得られるはずです。そうでない場合は、Vxxxx 編集コードで最大限の精度が得られます。XML、DECFLOAT、BLOB、CLOB、またはDBCLOB 列を含むデータの編集に、Uxxxx および Vxxxx コードは使用できません。

## FORM.COLUMNS

FORM.COLUMNS では、列の使い方について各種の選択を行ないます。

FORM.COLUMNS での指定は、FORM.MAIN に反映されます。逆に、FORM.MAIN (A から F) に指定した内容も、FORM.COLUMNS に反映されます。

図 18 は、FORM.COLUMNS パネルの入力フィールドを示しています。

FORM.COLUMNS								
COLUMNS:		Total Width of Report Columns: 66						
NUM	A COLUMN HEADING	B USAGE	C INDENT	D WIDTH	E EDIT	F SEQ		
1	ID	2	6	L	1			
2	NAME	2	9	C	2			
3	DEPT	2	6	L	3			
4	JOB	2	5	C	4			
5	YEARS	2	6	L	5			
6	SALARY	2	10	L2	6			
7	COMM	2	10	L2	7			
8	Total Earnings	2	12	L2	8			
*** END ***								
1=Help      2=Check      3=End      4>Show      5=Chart      6=Query		7=Backward      8=Forward      9=Specify      10=Insert      11=Delete      12=Report						
OK, FORM.COLUMNS is displayed.								
COMMAND ==>				SCROLL ==> PAGE				

図 18. FORM.COLUMNS パネルの入力フィールド

### A 列ヘッダー

**報告書:** 列ヘッダーを割り当てます。デフォルトの書式では、列ヘッダーを次のいずれかにすることができます。

- 列に割り当てられているラベル (使用しているサイトでラベルを使用している場合)
- その列を選択した表の中での列の名前
- 定数または計算値の列用に QMF で組み立てたヘッダー

使用しているサイトが DBCS データをサポートする場合は、272 ページの『2 バイト文字を含む名前』を参照してください。

列ヘッダー域に示されたヘッダーの上に、最大 40 文字の新しいヘッダーを入力することができます。ヘッダーには、元の列名と同様、ブランクまたは特殊文字を入れることができます。複数行のヘッダーを作成するときは、列ヘッダーの中に下線を使用して、行の切れ目を指定します。たとえば、EMPLOYEE\_NAME は、報告書の中で次のように表示されます。

EMPLOYEE  
NAME

列ヘッダーの全体の前または後の単独の下線は、何の効果もありません。たとえば、\_EMPLOYEE NAME としても、ブランク行は追加されません。しか

## FORM.COLUMNS

し、列ヘッダーのテキストに連続した下線があると、列タイトルに 1 行以上のブランクが追加されます。1 つの列ヘッダーを最大 9 行にすることができます。

たとえば、次の 2 つの列名があるとします。

```
1 ONE_TWO_THREE_FOUR_FIVE_SIX_SEVEN  
2 SIX_LINE_TITLE
```

入力された 1 個の下線につき 1 行のブランク行が追加されます。したがって、値は、報告書の中で次のように表示されます。

ONE	SIX
TWO	
THREE	LINE
FOUR	
FIVE	
SIX	TITLE
SEVEN	

列ヘッダーに 2 バイト文字を使用しているときも、1 バイト文字の下線で行の切れ目を指定できます。

列ヘッダーに大文字と小文字の両方を使いたいときは、プロファイルの CASE 値に STRING か MIXED を指定してください。これらの値の詳細は、126 ページの『SET PROFILE』を参照してください。

ヘッダーは、文字データの列では左寄せされ、数値データの列では右寄せされます。ヘッダーが複数行ある場合は、最も長い行が位置合わせされ、それより短い行は最も長い行の中央に置かれます。新しい調整値を入れると、それがこれらのデフォルト値に優先します。詳細については、216 ページの『列調整』 を参照してください。

ヘッダーの行が列の幅より長い場合は、列の幅一杯に埋められた後、右側が切り捨てられます。

列ヘッダーには、大域変数置換は行なわれません。

**図表:** Y 軸上にプロットされるデータの列ヘッダーは、図表の凡例に現われます。したがって、これらの列ヘッダーはできるだけ簡単にして、図表上で凡例がスペースを取り過ぎないようにしてください。

### B 取り扱い

**報告書:** 報告書の列をどのように処理するかを指定します。列の取扱コードがブランクの場合は、その列の値がリストされるだけで、他に処理は行なわれません(ただし、報告書の 1 つまたは複数の列の取扱コードが GROUP であり、少なくとも 1 つの列の取扱コードが総計の場合を除く)。その場合、ブランクの取扱の列は省略されます。この区域には、表 20 に示した多数の集約関数を入力できます。

表 20. 集約関数

集約	取扱コード	最小省略語	追加情報
横計	ACROSS	AC	254 ページの『ACROSS 取扱コード』

表20. 集約関数 (続き)

集約	取扱コード	最小省略語	追加情報
平均	AVERAGE (または AVG)	AV	255 ページの『列データの総計』
切れ目 1	BREAK、BREAK1	B、B1	198 ページの『FORM.BREAKn』
切れ目 1x	BREAKX、 BREAK1X	BX、B1X	198 ページの『FORM.BREAKn』
切れ目 2	BREAK2	B2	198 ページの『FORM.BREAKn』
切れ目 2x	BREAK2X	B2X	198 ページの『FORM.BREAKn』
切れ目 3	BREAK3	B3	198 ページの『FORM.BREAKn』
切れ目 3x	BREAK3X	B3X	198 ページの『FORM.BREAKn』
切れ目 4	BREAK4	B4	198 ページの『FORM.BREAKn』
切れ目 4x	BREAK4X	B4X	198 ページの『FORM.BREAKn』
切れ目 5	BREAK5	B5	198 ページの『FORM.BREAKn』
切れ目 5x	BREAK5X	B5X	198 ページの『FORM.BREAKn』
切れ目 6	BREAK6	B6	198 ページの『FORM.BREAKn』
切れ目 6x	BREAK6X	B6X	198 ページの『FORM.BREAKn』
計算	CALCid	CA	206 ページの『FORM.CALC』
カウント	COUNT	CO	255 ページの『列データの総計』
累積パーセンテージ	CPCT	CP	256 ページの『計算によるデータ値の置き換え』
累計	CSUM	CS	256 ページの『計算によるデータ値の置き換え』
最初の値	FIRST	F	255 ページの『列データの総計』
グループ	GROUP	G	260 ページの『GROUP 取扱コード』
最終値	LAST	L	255 ページの『列データの総計』
最大	MAXIMUM	MA	255 ページの『列データの総計』
最小	MINIMUM	MI	255 ページの『列データの総計』
除外	OMIT	O	261 ページの『ブランクおよび OMIT 取扱コード』
パーセンテージ	PCT	P	256 ページの『計算によるデータ値の置き換え』
標準偏差	STDEV	ST	255 ページの『列データの総計』
合計	SUM	SU	255 ページの『列データの総計』
合計累積パーセンテージ	TCPCT	TC	256 ページの『計算によるデータ値の置き換え』
合計パーセンテージ	TPCT	TP	256 ページの『計算によるデータ値の置き換え』

**C 間隔**

**報告書:** 列の左側のブランク・スペースの数を指定します。 その列は、1 つ前の列または左マージンからこのブランク・スペース分だけ離れます。 間隔には、0 から 999 までの任意の数値を指定できます。 漢字編集コードを使用する列では、最小の列間隔が 1 となります。 デフォルトの列間隔は、どの列についても、2 です。

間隔は、常に、1 バイト文字の数で指定されます。

## D 列幅

**報告書:** 列のデータまたは列ヘッダーを表示するために確保する文字数を示します。列幅には、1 から 32,767 までの任意の数値を指定できます。

表示している列で漢字編集コードが使用されているときは、列幅に 1 から 16,383 の値を指定できます。漢字編集コードを使用する列を表示または印刷すると、その幅は、列幅の 2 倍に 1 文字分のスペースを加えた値になります。DBCS データを含む列の幅の計算方法については、「DB2 QMF 使用の手引き」を参照してください。

数値データの列幅を指定するときは、指定する値に、数字のスペースだけでなく以下の文字のスペースも含まれていることを確認してください。

- 負符号 (編集コードが J のときを除きます)
- 小数点 (編集コードで指定された場合)
- 3 桁単位の区切り記号 (編集コードが D、K、P のとき)
- 通貨記号 (編集コードが D のとき)
- パーセント記号 (編集コードが P のとき)

表示する値の長さが列幅を超えると、

- 数値データなら、1 行分のアスタリスク (\*\*\*\*\*\*) で置き換えられます。

場合によっては、別のデータ・タイプを使用して数値のオーバーフローを避けることができます。たとえば、算術演算で、オペランドがすべて 10 進数で、オーバーフローが起こった場合、少なくとも 1 つのオペランドを浮動小数点数に変更することができます。この例では、オペランドは浮動小数点定数または浮動小数点の表列にすることができます。

- 文字、日付、時刻、タイム・スタンプなら、右端または左端 (データの調整指定によって決まります) が切り捨てられます。

列幅の問題を解決するには、列幅を変更して、報告書を表示し直してください。あるいは、列幅をそのままに保ちながら、1 行に収まらないデータを折り返して同列中の次の行に表示するように指定できます。列の折り返しは、非数値データだけに適用されます。列の折り返しの詳細については、261 ページの『編集コード』を参照してください。

デフォルト書式の列幅は、列ヘッダー中の最長行と少なくとも同じになるように定められます。それ以外の書式では、割り当てられる幅が、列のデータ・タイプによって表 21 に示すように変わります。

表 21. データ・タイプごとのデフォルト幅

データ・ タイプ	デフォルト・フォームの幅
SMALLINT	4
INTEGER	6
BIGINT	8
DECIMAL	データベースでの列幅に +3 文字分のスペース
FLOAT	10
DECFLOAT(16)	8

表 21. データ・タイプごとのデフォルト幅 (続き)

データ・タイプ	デフォルト・フォームの幅
DECFLOAT(34)	16
CHAR	データベースでの列幅
VARCHAR	データベースでの最大列幅
LONG VARCHAR	以下の小さい方 <ul style="list-style-type: none"> <li>• 列幅</li> <li>• 報告書上の他の列の数量とタイプに基づいて QMF が決める列幅</li> </ul>
GRAPHIC	データベースでの列幅
VARGRAPHIC	データベースでの列幅
LONG VARGRAPHIC	以下の小さい方 <ul style="list-style-type: none"> <li>• 列幅</li> <li>• 報告書上の他の列の数量とタイプに基づいて QMF が決める列幅</li> </ul>
DATE	10。または、使用しているサイトでローカルに日付形式が定義されているときは、次のうち大きい方: <ul style="list-style-type: none"> <li>• 列ヘッダーの幅</li> <li>• ローカルに定義されている日付形式の幅</li> </ul>
TIME	8。または、使用しているサイトでローカルに時刻形式が定義されているときは、次のうち大きい方: <ul style="list-style-type: none"> <li>• 列ヘッダーの幅</li> <li>• ローカルに定義されている時刻形式の幅</li> </ul>
TIMESTAMP	26
BINARY	データベースでの列幅
VARBINARY	データベースでの最大列幅

FORM.COLUMNS で行を挿入するとき、そのデフォルト幅は 10 です。

単精度浮動小数点データの場合は、データ・タイプ FLOAT の値は、単精度または倍精度と同じに扱われます。

**図表:** 図表の X 軸のラベルの文字数を指定します。

割り当てられているスペースより大きな幅を指定すると、ラベルが省略されることがあります。 ラベル省略の問題に対処するための 1 つの方法は、列ヘッダーの幅を切り縮めることです。 ラベルが切り捨てられると、割り当てられたスペース内により多くのラベルが収まります。

図表フォーマットの設定では、単精度浮動小数点データと倍精度浮動小数点データは同じ扱いになります。

データ・タイプが DATE、TIME、または TIMESTAMP である列の値は、文字ストリングとして扱われるため、Y 軸には表示できません。

### E 編集

**報告書:** 表示するデータのフォーマットを QMF にどう設定させるかを指定します。 FORM.COLUMNS に行を挿入するときのデフォルト値は、C です。

**図表:** X 軸のラベルには、GROUP または BREAK を用いる列からのもの (GROUP も BREAK もないときは、報告書の左端の列) が使われます。したがって、編集コードがこれらの列のデータに与える影響は、X 軸のラベルにも現われます。たとえば、X 軸用に選択されたデータが列内で折り畳まれていれば、最初の 1 行だけがラベルに用いられます。

Uxxxx または Vxxxx で編集される数値列は、Y 軸データとして使用できません。

列置換値 (&n ) をページ・ヘッダーとして (したがって、図表ヘッダーとして) 使用する場合、その値は書式内のその列の編集コードに従って編集されます。

219 ページの表 23 に、各データ・タイプに指定できる編集コードを示します。

DATE 列、TIME 列、TIMESTAMP 列で文字編集コードを使うと、これらの列内で折り返しができます。

### F 順序

**報告書:** この列に番号を入力して、報告書内の列の順序を変更することができます。初期の設定は、「NO.」列から取られます。1 ~ 999 の任意の番号を指定できます。番号が同じ 2 つの列は、書式でリストされたとおりの順序で表示されます。順序が、列の再配列に効力をもつためには、FORM.OPTIONS パネルの報告書列の自動再配列オプションを NO (デフォルト値) にセットしなければなりません。

変数が決定されると、列番号は「順序」列でなく「NO.」列から取られます。

「順」番号は、ACROSS 報告書では無視されます。

## 列属性を指定する

SPECIFY コマンドを使用すると、列ヘッダーの調整の変更、列内のデータの変更、または、列の定義を行うことができます。調整パネルや定義パネルにアクセスする方法には、次の 2 つがあります。

- 指定ファンクション・キーを押して指定パネルを表示させ、調整または定義を選択する。
- 調整指定 または 定義指定 (または有効な省略形) をコマンド行に入れ、カーソルを目的の列に移動して、ENTER キーを押す。この場合は指定パネルがバイパスされ、直接、調整ウィンドウまたは定義ウィンドウが表示されます。

## 列調整

調整を指定すると、217 ページの図 19 に示したパネルが FORM.COLUMNS パネルに上から重ねて表示され、選択する列の調整指定が示されます。

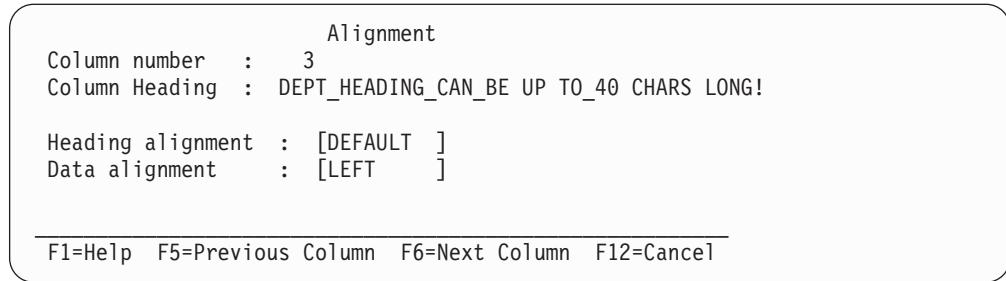


図 19. FORM.COLUMNS 調整パネル

ヘッダー調整とデータ調整に指定できる値は、LEFT、RIGHT、CENTER、DEFAULT の 4 つです。デフォルト値は、文字データを含んでいる列のヘッダーとデータは右寄せ、数値データを含んでいる列のヘッダーとデータは左寄せです。

調整値を変えるには、現在の値に重ねて新しい値を入れてください。ヘッダー調整とデータ調整の入力フィールド間の移動と、列調整指定間の移動には、タブ・キーを使用します。

列調整は主として、表データに適用されます。しかし、置換変数で \_B を使用すると、次のようにしてデータの位置調整が行なわれます。

- ・編集コードと列幅によるデータ編集が行なわれます。
- ・調整値が DEFAULT でないなら、先行ブランクと後続ブランクが除かれます。
- ・指定された調整値に従って、値の位置が調整されます。
  - データが文字なら、後続ブランクが除かれます。
  - データが数値なら、先行ブランクが除かれます。
  - &\_B が使用されていれば、ブランクは除かれません。

報告書では、データ調整の値が LEFT、RIGHT、CENTER のいずれかであれば、先行ブランクと後続ブランクが除かれます。データ調整の値が DEFAULT なら、ブランクは除かれません。

編集済みの文字データに先行ブランクが含まれている場合、また編集済みの数値データに後続ブランクが含まれている場合には、調整値のいかんにかかわらずブランクは除かれません。

## 列定義

### CICS ユーザーへの注

列定義は REXX に依存する機能であるため、CICS では使用できません。

列定義では、式を使って新しいデータ列を定義できます。照会で取り出される列と新しく定義される列との間には、いくつかの違いがあります。主な違いは、ユーザ一定義列に割り当てるデータ・タイプと長さです。

## FORM.COLUMNS

列の定義ではプロンプト・パネルが表示されて、列を定義するための式を入れるよう指示され、REXX がその式を評価するときにヌル値を含めるかどうかを尋ねられます。 QMF は、FORM.COLUMNS でその列用に指定された編集コードと列幅に従って、列のデータ・タイプと長さを決定します。しかし、定義されたその列に対して取扱コードが指定されていて、その取扱コードが同列の編集コードと整合しないときは、取扱コードに基づいてデータ・タイプが決定されます。

データベースから取り出された列とユーザー定義列のもう 1 つの違いは、データを保管またはエクスポートするとき、ユーザー定義列の値は保持されないことです。

列内の折り返しも、定義列では異なって見えることがあります。

- 定義列のデータが 254 バイトより短い場合は、列を折り返す方法に明らかな違いはありません。
- 定義列のデータが 254 バイトより大きく、列幅が 254 以下であれば、データは 254 番目まで折り返されますが、そのデータの残りは切り捨てられます。
- 定義列のデータが 254 バイトより大きく、列幅が 255 以上であれば、そのデータは列の幅のところで折り返されます。

FORM.COLUMNS から Definition を指定すると、図 20 に示したパネルが表示され、そこで新しい列を定義する式 (最大 50 文字) を入力できます。

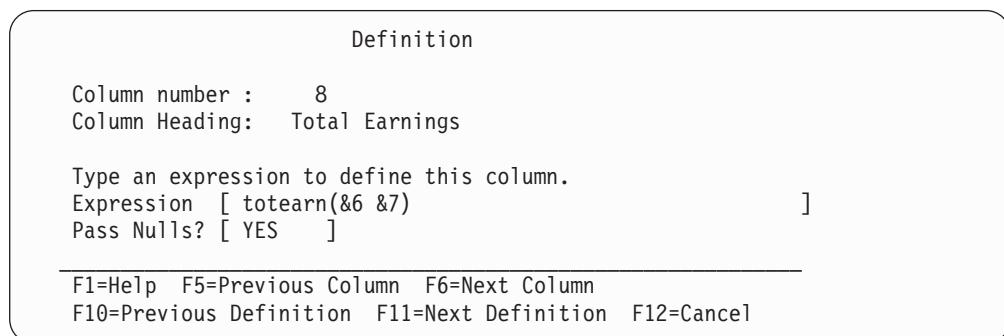


図 20. FORM.COLUMNS 定義パネル

新しい列の定義には、次のものが使えます。

- 文字定数または数値定数
- 次の書式変数 (QMF 書式変数の全般的な説明は、198 ページの『FORM.BREAKn』を参照)
  - &n (n は、照会の SELECT ステートメントの中の列位置を示す数値)
  - &DATE
  - &TIME
  - &ROW
  - 任意の大域変数。ただし、123 ページの『SET GLOBAL』に従っていること。
- 有効な REXX 式または REXX 関数
- 上記のいずれかを含む式

列定義に REXX 式を含めるとき、REXX から返される値が 32,767 文字を超えると、予期しない結果になることがあります。

1 つの列定義パネルから他の列定義パネルへ移動するには、前ファンクション・キーと次機能機能を使用してください。

### NULL 置換

NULL 置換を YES にした場合、表 22 に示した QMF に用意されている値を用いて、それぞれの状況における値の処理を変更できます。

表 22. データベースのヌル値のデフォルトの取り扱い

状況	値と置換される文字ストリング
データは空白	DSQNULL
データが未定義である	DSQUNDEF
データに数値オーバーフローがある	DSQOFLOW
データにインスタンスがない	DSQNOINS
データに関連がない	DSQNOREL

たとえば、ヌルであるデータベース変数 (データベース・ヌル値) はいずれも、式を REXX に渡して評価する前に、文字ストリング DSQNULL と置き換えられます。REXX の式またはプログラムを用意すれば、ストリングを検査して、データベース・ヌルを 0 (またはユーザーの目的に合った値) に置き換えることができます。

REXX 式からヌル値が返されたとき、それを報告書に渡すことができます。

NULL 置換が YES に設定され、式に、ヌル、未定義、オーバーフローの置換変数、またはインスタンスまたは関係が存在しない置換変数が含まれる場合、式全体がその状態を表す値に設定されます。この縮約は比較ではなく式で行われます。 詳細については、185 ページの『@IF 関数』を参照してください。

NULL 置換が NO であれば、上記にリストされている値にヌルが戻されます。REXX には、評価のための値を何も渡しません。

### 編集コード、データ・タイプ、および長さ

QMF は、定義された列のデータ・タイプと長さを、FORM.COLUMNS パネルでその列用に指定された編集コードと列幅に従って決定します。表 23 に結果を要約します。表にリストされた編集コードの詳細は、261 ページの『編集コード』を参照してください。

表 23. 編集コード、データ・タイプ、および長さ

編集コード	データ・タイプと長さ
文字 (C、CW、CT、CDx、B、BW、X、 XW)	CHAR - データベースでの列幅 VARCHAR - データベースでの最大列幅 LONG VARCHAR - 次のうちの小さい方 • 列幅 • 報告書の他の列の数とタイプに基づいて、QMF によって決定される幅
数値 (D、E、I、J、K、L、P)	数値 - 拡張浮動小数点

## FORM.COLUMNS

表 23. 編集コード、データ・タイプ、および長さ (続き)

編集コード	データ・タイプと長さ
2 進 (B、BW、X、XW)	BINARY - データベースでの列幅
	VARBINARY - データベースでの最大列幅
メタデータ (M)	CHAR - データベースでの列幅
	VARCHAR - データベースでの最大列幅
	LONG VARCHAR - 次のうちの小さい方
	<ul style="list-style-type: none"> <li>• 列幅</li> <li>• 報告書の他の列の数とタイプに基づいて、QMF によって決定される幅</li> </ul>
	SMALLINT - 4
	DECIMAL (DEC)、NUMERIC (NUM) - データベース内の列の幅と 3 文字分のスペースの合計
	FLOAT - 10
	GRAPHIC - データベースでの列幅
	VARGRAPHIC - データベースでの最大列幅
	LONG VARGRAPHIC - 次のうちの小さい方
U および V ユーザー編集コード (数値取扱コードなし)	<ul style="list-style-type: none"> <li>• 列幅</li> <li>• 報告書の他の列の数とタイプに基づいて、QMF によって決定される幅</li> </ul>
	DATE - 10。または、使用しているサイトでローカルに日付形式が定義されているときは、次のうち大きい方: <ul style="list-style-type: none"> <li>• 列ヘッダーの幅</li> <li>• ローカルに定義されている日付形式の幅</li> </ul>
U および V ユーザー編集コード (少なくとも 1 つの数値取扱コード)	TIME - 8。または、使用しているサイトでローカルに時刻形式が定義されているときは、次のうち大きい方: <ul style="list-style-type: none"> <li>• 列ヘッダーの幅</li> <li>• ローカルに定義されている時刻形式の幅</li> </ul>
	VARCHAR - データベースでの最大列幅
U および V ユーザー編集コード (少なくとも 1 つの数値取扱コード)	LONG VARCHAR - 次のうちの小さい方
	<ul style="list-style-type: none"> <li>• 列幅</li> <li>• 報告書の他の列の数とタイプに基づいて、QMF によって決定される幅</li> </ul>
U および V ユーザー編集コード (少なくとも 1 つの数値取扱コード)	数値 - 拡張浮動小数点

DB2 Server (VSE または VM 版) データベースと DB2 (z/OS 版) データベースは、拡張浮動小数点データ・タイプをサポートしていません。これらのデータベースに接続してデータを操作する場合、数値列を拡張浮動小数点列として定義しておくといかもしません (たとえば、データベース・データ・タイプとして使うとオーバーフロー状態を起こすような DECIMAL や INTEGER のデータを扱う場合など)。

## 印刷の考慮事項

FORM を印刷すると、FORM.COLUMNS に続くページには、画面上に見る調整指定ウィンドウと定義指定ウィンドウではなく、列定義情報と列調整情報が印刷されます。NUM フィールドは、列定義と列調整ごとに繰り返されます。図 21 に例を示します。

1 FORM.COLUMNS				FORM:
NUM	HEADING ALIGN	DATA ALIGN	DEFINITION	PASS NULLS?
1	DEFAULT	DEFAULT		NO
2	CENTER	CENTER		NO
3	DEFAULT	DEFAULT		NO
4	LEFT	DEFAULT		NO
5	DEFAULT	DEFAULT		NO
6	DEFAULT	DEFAULT		NO
7	DEFAULT	DEFAULT		NO
8	RIGHT	RIGHT	&6 + &7	NO
9	DEFAULT	DEFAULT	(&6 + &7) * &5	NO
*** END ***				
05/05/91 11:10 AM				PAGE 3

図 21. 書式を印刷すると表示される列定義と調整の情報

## FORM.CONDITIONS

### CICS ユーザーへの注

FORM.CONDITIONS は、REXX で書かれた式を使用します。CICS では REXX を使用できません。

FORM.CONDITIONS では、条件付きフォーマット設定のための式を入力します。条件付きフォーマット設定で入力した式により、FORM.DETAIL で指定されたフォーマット設定バリエーションがいつ現れるかが決まります。

条件付きフォーマット設定を使用して、グループ・データの詳細なテキストを指定することができます。この条件は、グループの第 1 行からのデータを使用して評価されます。条件が 真 であれば、このバリエーションの詳細なテキストが印刷されます。条件が 偽 であれば、このバリエーションの詳細なテキストが印刷されません。

222 ページの図 22 は、FORM.CONDITIONS パネルの入力フィールドを示しています。

## FORM.CONDITIONS

FORM.CONDITIONS		
<b>A</b>	<b>B</b>	<b>C</b>
ID	CONDITIONAL EXPRESSION	PASS NULLS?
-----		
NO		
*** END ***		
1=Help	2=Check	3=End
7=Backward	8=Forward	4>Show
OK, FORM.CONDITIONS is displayed.	9=	5=Chart
COMMAND ===>	10=Insert	11=Delete
	12=Report	6=Query
		SCROLL ===> PAGE

**A ID** 条件式の 1 から 3 文字の ID を入力します。ID には、1 から 999 までの任意の番号を指定できます。FORM.DETAIL パネルのパネル・バリエーションの選択フィールドに C 選択コードを付加すると、詳細バリエーションをフォーマット設定する FORM.CONDITIONS 内の式を識別します。FORM.DETAIL パネルの詳細は、223 ページの『FORM.DETAIL』を参照してください。

### **B 条件式**

有効な REXX 式を入力します。FORM.CONDITIONS の式は、結果が真と偽のどちらかの値になるという点で、FORM.CALC の式とは異なります。結果が 1 になる式が真、その他の結果になる式が偽です。非数値データは、ブランクやヌルを含め、偽と見なされます。条件式には、有効な大域変数が使用できます。ただし、条件式で使用できる QMF 書式変数は、&ROW、&DATE、&TIME、および &n のみです。ここで、n は、照会の SELECT ステートメントの中の列位置を示します。

条件付きフォーマット設定の使用の詳細は、248 ページの『QMF 書式での REXX の使用』を参照してください。

### **C NULL 置換**

YES または NO を入力します。

**YES** 表 24 に示した QMF に用意されている値を用いて、それぞれの状況における値の処理を変更できます。

表 24. データベースのヌル値のデフォルトの取り扱い

状況	値と置換される文字ストリング
データは空白	DSNULL
データが未定義である	DSUNDEF
データに数値オーバーフローがある	DSQOFLOW
データにインスタンスがない	DSQNOINS
データに関連がない	DSQNOREL

**NO** 上記の値にヌルを返します。REXX には、評価のための値を何も渡しません。

図 22. FORM.CONDITIONS パネルの入力フィールド

## FORM.DETAIL

次のとき、FORM.DETAIL を使用します。

- 列ヘッダーの前にテキストを指定するとき
- 表データとテキストを結合するとき
- 表データを省略し、各データ値全体をテキストとして表示するとき

FORM.DETAIL は、ユーザーが定義する詳細バリエーションから構成されます。最大 99 通りのバリエーションを作成でき、各バリエーションを FORM.CONDITIONS に入力された条件と対応させることができます。各条件が相いれない場合を除き、1 つのデータ行に対していく通りもの詳細バリエーションを表示できます。

FORM.DETAIL は、図表に影響を与えません。

図 23 は、FORM.DETAIL パネルの入力フィールドを示しています。

FORM.DETAIL		A VAR 1 of 1
<b>B</b>	Include Column Headings with Detail Heading? ===> YES	
<b>C</b>	LINE <b>D</b> ALIGN <b>E</b> DETAIL HEADING TEXT	
1	LEFT	-----+-----1-----+-----2-----+-----3-----+-----4-----+-----5-----+
2	LEFT	
*** END ***		
<b>F</b>	New Page for Detail Block? ===> NO	<b>G</b> Repeat Detail Heading? ===> NO
<b>H</b>	Keep Block on Page? ===> NO	<b>I</b> Blank Lines After Block ===> 0
<b>J</b>	Put Tabular Data at Line (Enter 1-999 or NONE) ===> 1	
<b>K</b>	LINE <b>L</b> ALIGN <b>M</b> DETAIL BLOCK TEXT	
1	LEFT	-----+-----1-----+-----2-----+-----3-----+-----4-----+-----5-----+
2	LEFT	
*** END ***		
<b>N</b>	Select Panel Variation? ===> YES	
1=Help      2=Check      3=End      4>Show      5=Chart      6=Query		
7=Backward      8=Forward      9=      10=Insert      11=Delete      12=Report		
OK, FORM.DETAIL is displayed.		SCROLL ===> PAGE
COMMAND ===>		

図 23. FORM.DETAIL パネルの入力フィールド

### A VAR 1 / 1

最初の数値は現行パネルのバリエーションを表し、2 番目は作成したバリエーションの総数 (最大 99) を表します。デフォルトの書式は、VAR 1 / 1 を表示します。

新しい詳細バリエーションを作成するには、バリエーション・パネルの総数より 1 だけ大きい値を現パネル・バリエーション値に上書きします。新しいパネルは順次に追加していかなければなりません。

現パネル・バリエーション値を他の識別値で上書きすれば、その識別値で表される既存のパネル・バリエーションに移動できます。コマンド行に NEXT

コマンドと PREVIOUS コマンドを入れても、他のパネル・バリエーションを表示できます。(詳細については、84 ページの『NEXT』および 85 ページの『PREVIOUS』を参照。)

**B** から **E** までの部分には、報告書内で、FORM.COLUMNS で指定される列ヘッダーの前にくるテキストを指定します。

#### **B 明細ヘッダーに列ヘッダーを含める？**

**YES** 列ヘッダーが明細ヘッダーの一部になります。その結果得られる明細ヘッダーは、BREAK パネルまたは **G 明細ヘッダーの反復**

**NO** 列ヘッダーは抑止されます。

**C 行** 明細ヘッダー・テキスト行とその相対位置を特定します。何行でも指定できます。行番号は、1 から 999 の任意の番号かブランクです。

複数の行に同じ「行」値を使用すると、それらの行は追加行の「調整」値に従って結合されます。同じ「行」をもつ行が報告書の幅より長かったり、その「調整」値が矛盾していたりすると、行の重なり合いが起こります。

#### **D 調整 (位置合わせ)**

報告書上での明細ヘッダー・テキスト各行の水平位置を指定します。報告書の幅の範囲内であれば、どこにでも行を位置づけることができます。

**Left** 明細ヘッダー・テキストを左寄せにします。

**Right** 明細ヘッダー・テキストを右寄せにします。

**Center** 明細ヘッダー・テキストを中央に合わせます。

**n** 明細ヘッダー・テキストを、その行の *n* 桁目 から開始します。ここで、*n* は、1 から 999999 の任意の数値を指定できます。

**Append** APPEND が他の行に付加されないテキスト行に使用される場合、テキストの行は左寄せされます。

前のテキスト行と付加行を同一行に置きたいときは、両者の「行」値が同じでなければなりません。報告書の幅が不十分で、付加されたテキスト行の全体が入りきらないと、その一部が切り捨てられることがあります。

#### **E 明細ヘッダー・テキスト**

明細ヘッダー・テキストを指定します。INSERT コマンドで、最大 999 行のテキストを追加できます。テキストの各行の長さは、最大 55 文字です。

「調整」(ALIGN) 値に APPEND を指定するか、具体的な水平位置を指定すれば、その行にテキストを追加できます。

使用しているサイトが DBCS データをサポートする場合は、272 ページの『2 バイト文字を含む名前』を参照してください。

報告書では、常に明細ヘッダー・テキストが列ヘッダーに先行します。明細ヘッダーは、明細ヘッダー・テキストか列ヘッダー、またはその両方から構成されます。省略しない限り、明細ヘッダー・テキストと列ヘッダーで、明細ヘッダーを構成します。

デフォルトでは、明細ヘッダーは報告書の左マージンから右マージンまで広げることができます。右マージンを超えるどのテキストも、表示または印刷

されません。 FORM.OPTIONS パネルで報告書テキストの幅を変更することによって、幅を変更することができます。幅を明示的に指定しない場合は、表データの幅で右マージンが決まります。

報告書を印刷すると、ページ・ヘッダーのフォーマット設定時に現データ行用に選択されたすべての明細ヘッダーが印刷されます。明細ヘッダーの行数が、そのページで使用できる行数を超えると、それらの超過したヘッダー行は失われます。

明細ヘッダーには、次の変数の値を入れることができます。

#### **大域変数**

SET GLOBAL を使用して、明細ヘッダー・テキストに使う変数をセットします。このコマンドの詳細については、123 ページの『SET GLOBAL』を参照してください。

**&n** この報告書に使用する書式上の *n* 番目の列の値。たとえば、次の明細ヘッダーがあるとします。

```
ID NUMBER: &1 EMPLOYEE NAME: &2
```

この明細ヘッダーは、報告書の中に次のヘッダーを作成します。

```
ID NUMBER: 50 EMPLOYEE NAME: HANES
```

&*n* 値は、改ページ直後に列 *n* の現在行にある値です。無条件に選択されるバリエーションの明細ヘッダーは、表示報告書では各画面の一番上に表示されます。しかし、&*n* の値は表示報告書の第 1 画面にしか現われません。報告書をページの切れ目つきでオンライン表示したいときは、DPRE コマンドを出してください。このコマンドの詳細については、31 ページの『DPRE』を参照してください。

この特殊な構文を使用すると、置換値の幅が、FORM.COLUMNS または FORM.MAIN パネル上の対応する列で指定された幅によって決まります。

#### **&ROW**

明細ヘッダーがフォーマット設定されたときの現在データ行の番号。

#### **&DATE**

PRINT コマンドが実行されたときの日付（印刷報告書）、または現在日付（表示報告書）。

#### **&TIME**

PRINT コマンドが実行されたときの時刻（印刷報告書）、または現在時刻（表示報告書）。

#### **&PAGE**

現行ページ番号

&DATE、&TIME、および &PAGE の説明は、198 ページの『FORM.BREAKn』を参照してください。

**F** から **M** までは、報告書でデータ行ごとに繰り返される報告データを規定します。このデータを明細ブロックと呼びます。表データ（選択した場合）と、单一データ行または単一明細行（たとえば、表の 1 行）と関連づけられたテキストをいいます。

**F 明細ブロックで改ページ**

印刷報告書で、明細ブロックが現れるたびに、それを新しいページに印刷するかどうか指定します。改ページは、報告書がまだページの上端にない場合に開始します。

**G 明細ヘッダーの反復**

明細ブロック・テキストがあるたびに、その前に明細ヘッダーを（繰り返して）置くかどうかを指定します。明細ヘッダーには、FORM.DETAIL パネルで指定された明細ヘッダー・テキストと、FORM.COLUMNS パネルでリストされたそれに続く列ヘッダー（抑止しない場合）が含まれます。

**NO** 明細ヘッダーは、オンライン報告書では各画面の始め、印刷報告書では各ページの始めに設定されます。

**YES** 明細ブロック・テキストが現れるたびに、その前に明細ヘッダーを設定します。

**H ブロックをページ内に維持**

1 つの明細ブロックを構成する全テキストを、印刷報告書の 1 ページ内に収めるかどうかを指定します。

**NO** 明細ブロックが印刷報告書の複数ページに分割されることがあります。

**YES** 明細ブロックが複数ページに分割されることを禁止します。明細ブロックがページの残りの部分に印刷するには大き過ぎると、改ページが起こります。

**I ブロック後のブランク行数**

明細ブロック・テキストの後のブランク行数を指定します。

さらに、FORM.OPTIONS パネル上の明細行の行送りオプションも、明細ブロック・テキストの後のブランク行数に影響を与えます。

**J 表データ表示行 (1-999 または NONE で指定)**

表データを（FORM.COLUMNS または FORM.MAIN で指定された表フォーマットで）生成するかどうか、またその表データをどこに配置するかを指定します。番号は、表データを入れる明細ブロック・テキスト行の番号に対応します。NONE（または N）は、表データをフォーマット設定しないことを意味します。NONE は、切れ目テキストまたは総計値に影響を与えません。

このオプションは、テキストと表示データを混在させる時に使用できます。数値を指定すると、表データが、同一行の明細ブロック・テキストの上に重ね書きされるか、またはそれと結合されます。

NONE を指定すると、表データはフォーマット設定されませんが、列置換値を使用すれば、列の値を明細ブロック・テキストに組み込むことができます。

**K 行** 明細ブロック・テキスト行を特定し、その相対位置を指定します。表データ

行を何行でも指定できます。1から999までの任意の数値またはブランクを指定できます。追加情報は、このトピックの前の方で説明した『**C** 行』を参照してください。

#### **L 調整**

報告書上の明細ブロック・テキスト各行の水平位置を指定します。報告書の幅の範囲内であれば、どこにでも行を位置づけることができます。有効な値は、LEFT、RIGHT、CENTER、APPEND、または1から999,999の任意の値です。

調整値は、表データの水平位置に影響を与えません。表データの配置を変更するには、FORM.COLUMNS または FORM.MAIN で列幅または列間隔を変更します。追加情報は、このトピックの前の方で説明した『**D 調整**』を参照してください。

#### **M 明細ブロック・テキスト**

明細ブロック・テキストを指定します。INSERT コマンドで、最大999行の明細ブロック・テキストを追加できます。テキストの各行の長さは、最大55 文字です。「調整」(ALIGN) 値に APPEND を指定するか、具体的な水平位置を指定すれば、その行にテキストを追加できます。

使用しているサイトが DBCS データをサポートする場合は、272 ページの『2 バイト文字を含む名前』を参照してください。

デフォルトでは、明細ブロック・テキストを、報告書の左マージンから右マージンまで広げることができます。右マージンを超えるどのテキストも、表示または印刷されません。FORM.OPTIONS パネルで報告書テキストの幅を変更することによって、幅を変更することができます。幅を指定しない場合は、表データの幅によって右マージンが決まります。

明細ブロック・テキストには、リテラル・テキストと次の変数の値を入れることができます。

##### **大域変数**

SET GLOBAL を使用して、明細ブロック・テキストに使う変数をセットします。このコマンドの詳細については、123 ページの『SET GLOBAL』を参照してください。

**&n** この報告書に使用する書式上の *n* 番目の列の値。たとえば、次の明細ブロック・テキストがあるとします。

DEPARTMENT: &3 EMPLOYEE NAME: &2

この明細ブロック・テキストは、報告書の中に次のような行を作成します。

DEPARTMENT: 20 EMPLOYEE NAME: SANDERS

##### **&COUNT**

最後の制御の切れ目以降に表示または印刷された行の数。この値は通しでカウントされ、データ行ごとに加算されていきます。

##### **&ROW**

明細ブロックのデータ行の番号が報告書に印刷または表示されます。

## FORM.DETAIL

グループ合計を出す報告書の明細ブロック・テキストでは、グループの最終行のデータ行番号が印刷されます。

### &CALCid

計算値

### &DATE

現在の日付

### &TIME

現在の時刻

### &PAGE

現行ページ番号

&CALCid 変数の説明は、206 ページの『FORM.CALC』を参照してください。

&DATE、&TIME、および &PAGE の説明は、198 ページの『FORM.BREAKn』を参照してください。

**&an** *n* は有効な列番号です。*a* は QMF 集約関数で、AVG、COUNT、CPCT、CSUM、FIRST、LAST、MAX、MIN、PCT、STDEV、SUM、TCPCT、TPCT のいずれかです。総計値は、現切れ目レベル内の通しの値に基づきます。

明細ブロック・テキストの総計の値は、最後の制御の切れ目から現在行までのデータ値に基づきます。さらに AVG や STDEV などの計算値も、最後の制御の切れ目以降のデータ値に基づきます。たとえば、&AVG6 は列 6 の(現在行までの)合計を行数で割った値です。

明細レベルでは、&SUM と &CSUM は同じ結果になります。明細ブロック・テキスト中に置かれた &SUM6 および &CSUM6 は、どちらも列 6 の現在行までの合計値を与えます。

明細ブロック・テキストの中でパーセンテージ集約変数(PCT、TPCT、または TCPCT)を使用し、それと D 編集コードをもつ列を関連づけると、QMF は、明細ブロック・テキストの中のパーセンテージの値を L 編集コードをもつ場合と同じようにフォーマット設定します。同様に、明細ブロック・テキストの中で STDEV 集約変数を使用し、それを P または D 編集コードをもつ列と関連づけると、QMF は、明細ブロック・テキストの中の標準偏差を、L 編集コードをもつ場合と同じようにフォーマット設定します。

詳細については、264 ページの『数値データの編集コード』および 269 ページの『書式で使用される変数』の L コードを参照してください。

### N パネル・バリエーションの選択

パネル・バリエーションを一つ選択するか指定します。指定できる値は次のどちらかです。ブランクは認められません。

**YES** 報告書のフォーマット設定に必ず選択されます。バリエーション番号 1 では、これがデフォルト値です。

**NO** フォーマット設定には選択されません。バリエーション番号 2 から 99 では、これがデフォルト値です。この値は、報告書であるバリエーションのフォーマット設定を一時的に禁止する場合に使用してください。

次の 2 つの選択項目で、報告書を選択的にフォーマット設定できます。1 パネル分全部の明細テキストとフォーマット設定オプションを、FORM.CONDITIONS パネルの特定条件と関連づけることもできますし (条件フォーマット設定)、ツリー・データの枝に対応する特定データ列と関連づけることもできます。

#### C1-C999

FORM.CONDITIONS での条件を指定するために選択できます。条件が真であれば、関連の FORM.DETAIL バリエーションがフォーマット設定されます。

#### E1-E999

これで指定された列のデータが存在するとき、フォーマット設定のために選択されます。列は E に続く番号で指定します。この番号は、FORM.MAIN または FORM.COLUMNS 上の列の NUM 値に対応します。

## FORM.FINAL

FORM.FINAL では、報告書の最終テキストの内容と配置に関する詳しい選択を行います。QMF はこのテキストを報告書の末尾に配置しますから、たとえば、報告書全体の最終合計などに使用できます。

FORM.MAIN の区域 **H** (195 ページの『FORM.MAIN』を参照) は、報告書の最終テキストを指定します。FORM.MAIN のここに指定した内容は、すべて FORM.FINAL に反映されます。同様に、最終テキストの最初の行は FORM.MAIN に反映されます。

230 ページの図 24 は、FORM.FINAL パネルの入力フィールドを示しています。

## FORM.FINAL

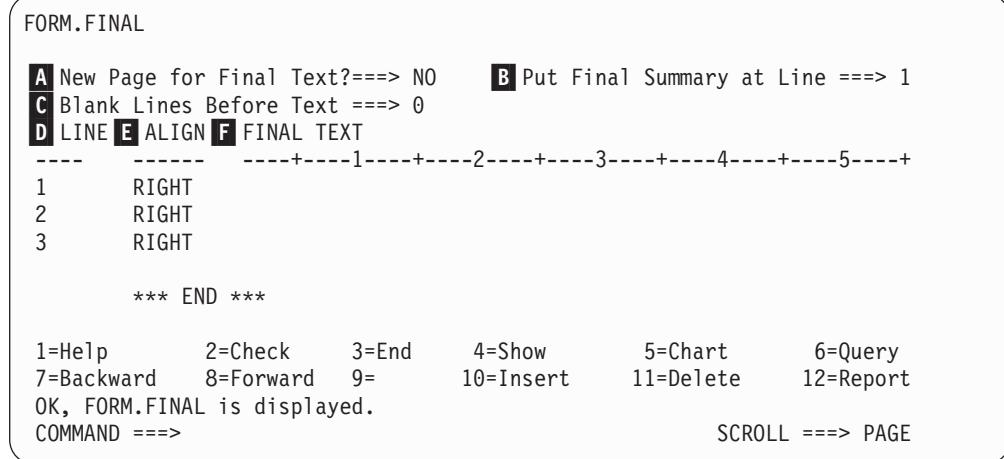


図 24. FORM.FINAL パネルの入力フィールド

### A 切れ目で改ページ

**報告書:** 印刷報告書本文とは別のページに最終テキストを印刷するかどうかを指定します。改ページは、報告書がまだページの上端にない場合に開始します。

### B 最終合計行の位置

**報告書:** 報告書の最終合計を生成するかどうか、生成するときはそれを最終テキストに対してどこに配置するか、を指定します。指定できる値は 1 から 999 までか NONE です。番号は最終テキスト行の番号で、その行の横に最終合計が配置されます。NONE (または N) とすれば、最終合計が省略されます。

列内折り返しがある列で、最終合計値の長さが 2 行以上にわたると予想されるときは、折り返された最終合計値の予想最終行に最終テキストを入れてください。これが必要になるのは、折り返し列の取扱コードが MAX、MIN、FIRST、または LAST の場合です。

たとえば、列 NAME (Q.STAFF の) が列幅 2 にセットされていて、編集コードが CW、取扱コードが MAX なら、FORM.FINAL の 5 番目の行に最終テキストを (たとえば、ピリオドだけでも) 置けば、その列の最終合計値全体 (YAMAGUCHI) を見ることができます。

ACROSS 報告書で 1 つの合計に 2 データ行を表示できるのは、横方向合計欄と最終合計の両方がある場合だけです。つまり、書式の列の取り扱いが CSUM、CPCT、PCT、TPCT、または TCPCT の場合です。

横方向合計欄が FORM.OPTIONS で省略されると、ACROSS 横方向の値も省略され、1 グループにつき 1 行だけ (ACROSS 縦方向の値を含む) がフォーマット設定されます。

最終合計が FORM.FINAL で省略されると、ACROSS 縦方向の値が省略され、1 グループにつき 1 行 (ACROSS 横方向の値を含む) だけがフォーマット設定されます。

**図表:** 合計行が 2 行あるが、そのうち 1 行しか対話式图表ユーティリティ (ICU) で図表化されない場合、2 番目合計データ行には、PCT、CPCT、または CSUM の指定がある列にのみ値が入ります。このようないい例では、次のようにになります。

- 最初の行の値は、ACROSS 横方向 (グループ) 合計に関する小計値です。
- 2 行目の値は、ACROSS 縦方向 (カテゴリー) 合計に関する小計値です。

ICU を使用した图表作成の詳細は、「DB2 QMF 使用の手引き」を参照してください。

### C テキスト前のブランク行数

**報告書:** 報告書の本文と最終テキストの最初の行の間のブランク行数を指定します。指定できる値は 1 から 999 までか BOTTOM です。デフォルトは 0 です。

たとえば、報告書の本文と最終テキストの最初の行とのあいだにブランク行を 1 行だけ入れたい場合は、このフィールドに 1 をタイプします。最終テキストと本文との間を 2 行のブランク行で区切りたい場合は、このフィールドに 2 をタイプします。

報告書本文の終了位置に関係なく、最終テキストを現行ページの末尾に表示する場合は、このフィールドに BOTTOM (または B) を入力します。

### D 行 報告書:

最終テキストの各行に番号を与えると共に、それらの行自体および最終テキスト開始行 (テキスト前のブランク行数 で指示) との相対的な位置を指定します。

指定する数値は、それぞれ必ずしも 1 から始まる必要はなく、連続している必要もありません。最終テキストの行間および報告書の本文と最終テキストの最初の行の間を任意に空けることができます。たとえば、最終テキストが 3 行にわたるとして、テキストの「行」値に 1、3、5 を選択したとすれば、QMF は、テキスト前のブランク行数フィールドで指定された行から最終テキストを開始し、テキスト行間に 1 行ずつのブランク行をはさみます。「行」値のどれにも 1 を指定しないと、最終テキストは、テキスト前のブランク行数フィールドで指定された行からは始まりません。指定された最初の行番号になるまでブランク行が続きます。「行」値をブランクにしておくと、QMF は関連テキストを無視します。

たとえば、FORM.FINAL に次の値があるとします。

LINE	ALIGN	FINAL TEXT
2	LEFT	GRAND TOTALS FOR
3	LEFT	ALL DEPARTMENTS

作成される報告書の中で、これらの値は、図 25 のように表示されます。

GRAND TOTALS FOR  
ALL DEPARTMENTS

図 25. 報告書の最終テキストの前のブランク行の追加

テキスト第 1 行の前に 1 行のブランク行が置かれることに注意してください。

上の例で、テキスト前のブランク行数フィールドに 0 を指定してあったとすれば、GRAND TOTALS FOR というテキストが、報告書本文の直後の行に表示されるはずだと考えられます。しかし、テキスト第 1 行の「行」値が 2 ですから、QMF は 1 行のブランク行（最終テキストの存在しない第 1 行が入るはずの行）を飛ばして、FORM.FINAL に指定された最初の行を、報告書の最終テキストの 2 行目に印刷します。

複数の行に同じ「行」値を使用すると、それらの行は追加行の「調整」値に従って結合されます。同じ「行」値をもつ行は、「調整」値が同じかなんらかの意味で矛盾していると、互いに重なり合います。たとえば、2 つの最終テキスト行に同じ「行」値を指定し、最初の行の「調整」値を LEFT、2 番目の行の「調整」値を CENTER にしたとします。この場合、最初の行のテキストが報告書の中央を超えると、2 番目の行が最初の行と一部重なり合うことになります。

## E 調整

**報告書:** 報告書上での最終テキスト各行の水平位置を指定します。報告書に最終合計データが入る場合、最終テキスト行の長さは左マージンから合計データの始めまでです。

しかし、報告書に最終合計データが入らない場合は、1 行分全体の長さ（左マージンから右マージンまで）が最終テキスト行の長さになります。その行長は、オンライン報告書では表示報告書の幅、印刷報告書では印刷報告書の幅です。

<b>Left</b>	最終テキストの行を左寄せします。
<b>Right</b>	最終テキストの行を右寄せします。これはデフォルトです。
<b>Center</b>	最終テキストの行を中央に合わせます。
<i>n</i>	最終テキストを、その行の <i>n</i> 柄目から開始します。ここで、 <i>n</i> は、1 から 999999 の任意の数値を指定できます。
<b>Append</b>	前の最終テキスト行の終わりに、行を付加します。最終テキストの最初の行（最も小さな「行」値をもつテキスト行）に APPEND を使用すると、そのテキスト行は左寄せされます。
	付加されるテキスト行は、付加先のテキスト行と同じ「行」値をもっていかなければなりません。報告書の幅が不十分で、付加されたテキスト行の全体が入りきらないと、その一部が切り捨てられることがあります。

たとえば、FORM.FINAL に次のように入力されたとします。

```
Blank Lines Before Text ===> 0
LINE  ALIGN   FINAL TEXT
-----
1    RIGHT    TOTAL
1    APPEND   SALARIES
3    RIGHT
```

これらの値によって、次のような報告書が作成されます。

DEPT	COMM	JOB	SALARY
66	55.50	CLERK	10988.00
			.
			.
	1285.00	SALES	17844.00
			.
		*	66147.00
			=====
		TOTAL SALARIES	152223.20

## F 最終テキスト

**報告書:** INSERT コマンドで、最大 999 行の最終テキストを追加できます。テキストの各行の長さは、最大 55 文字です。「調整」(ALIGN) 値に APPEND を指定するか、具体的な水平位置を指定すれば、その行にテキストを追加できます。

使用しているサイトが DBCS データをサポートする場合は、272 ページの『2 バイト文字を含む名前』を参照してください。

デフォルトの解釈では、最終テキストは報告書の左マージンから始まり、報告書に合計データがあればその始めまで、なければ報告書の右マージンまで続きます。しかし、最終テキストの幅は、FORM.OPTIONS の報告書のテキスト行幅項目を変更することで選択できます (234 ページの『FORM.OPTIONS』を参照してください)。

報告書の最終テキストに大文字と小文字の両方を使いたいときは、プロファイルの CASE 値に STRING か MIXED を指定してください。これらのオプションの詳細は、126 ページの『SET PROFILE』を参照してください。

最終テキストには、次の変数値を入れることができます。

### 大域変数

SET GLOBAL を使用して、最終テキストで使う変数をセットします。このコマンドの詳細については、123 ページの『SET GLOBAL』を参照してください。

**&n** この報告書に使用する書式上の *n* 番目の列の最終値

### &COUNT

最後の制御の切れ目以降に表示または印刷された行の数。この値は通しでカウントされ、データ行ごとに加算されていきます。

### &ROW

報告書全体の最終データ行の行番号が報告書に印刷または表示されます。

### &CALCid

計算値

### &DATE

現在の日付

### &TIME

現在の時刻

### &PAGE

現行ページ番号

&CALC*id* については、206 ページの『FORM.CALC』を参照してください。

&DATE、&TIME、および &PAGE の説明は、198 ページの『FORM.BREAKn』を参照してください。

**&an** *n* は有効な列番号です。*a* は QMF 集約関数で、AVG、COUNT、CPCT、CSUM、FIRST、LAST、MAX、MIN、PCT、STDEV、SUM、TCPCT、TPCT のいずれかです。総計値は、現切れ目レベル内の通しの値に基づきます。

明細ブロック・テキストの中でパーセンテージ集約変数 (PCT、TPCT、または TCPCT) を使用し、それと D 編集コードをもつ列を関連づけると、QMF は、明細ブロック・テキストの中のパーセンテージの値を L 編集コードをもつ場合と同じようにフォーマット設定します。同様に、明細ブロック・テキストの中で STDEV 集約変数を使用し、それを P または D 編集コードをもつ列と関連づけると、QMF は、明細ブロック・テキストの中の標準偏差を、L 編集コードをもつ場合と同じようにフォーマット設定します。

詳細については、264 ページの『数値データの編集コード』および 269 ページの『書式で使用される変数』の L コードを参照してください。

---

## FORM.OPTIONS

FORM.OPTIONS は、報告書の外観を調整する目的に使用します。

FORM.MAIN の区域 **J** (一括表示 と省略時切れ目テキスト - 195 ページの『FORM.MAIN』を参照) では、報告書全体の体裁に関する 2 つのオプションを指定します。FORM.MAIN のこの区域に指定した内容は、すべて FORM.OPTIONS に反映されます。同様に、FORM.OPTIONS に指定した内容の一部は、FORM.MAIN に反映されます。

235 ページの図 26 は、FORM.OPTIONS パネルの入力フィールドを示しています。

## FORM.OPTIONS

What do you want for					
<b>A</b>	Detail spacing?      ==> 1				
<b>B</b>	Line wrapping width?      ==> NONE				
<b>C</b>	Report text line width?      ==> DEFAULT				
<b>D</b>	Number of fixed columns in report?      ==> NONE				
Do you want					
<b>E</b>	Outlining for break columns?      ==> YES				
<b>F</b>	Default break text (*)?      ==> YES				
<b>G</b>	Function name in column heading when grouping?      ==> YES				
<b>H</b>	Column wrapped lines kept on a page?      ==> YES				
<b>I</b>	Across summary column?      ==> YES				
<b>J</b>	Automatic reordering of report columns?      ==> NO				
<b>K</b>	Page renumbering at the highest break level?      ==> NO				
Do you want separators for					
<b>L</b>	Column heading? ==> YES	<b>M</b>	Break summary? ==> YES		
<b>N</b>	Across heading? ==> YES	<b>O</b>	Final summary? ==> YES		
1=Help	2=Check	3=End	4>Show	5=Chart	6=Query
7=	8=	9=	10=	11=	12=Report
OK, FORM.OPTIONS is displayed.					
COMMAND ==>				SCROLL ==> PAGE	

図 26. FORM.OPTIONS パネルの入力フィールド

**A 明細行の行送り**

**報告書**：表データ行同士または明細ブロック同士の行間隔を選択します。明細ブロック・テキスト内の行送りには影響しません。この値には、1 ~ 999 を指定できます。デフォルト値は 1 行の行送りで、ブロック・テキスト間にプランク行を入れません。

FORM.DETAIL パネルの明細ブロック間の間隔オプション（223 ページの『FORM.DETAIL』を参照）も、明細ブロック間の間隔に影響します。

**B 表の折り返し幅**

**報告書**：報告書で列内折り返しを行なうかどうか指定します。行なうときは、その折り返し幅を指定します。指定できる値は 1 から 999 までか NONE です。デフォルト値は NONE で、報告書の列内折り返しを行なわないことを意味します。

**ACROSS** 報告書または列の折り返しがある報告書では、行の折り返しはできません。明細ヘッダー・テキストおよび明細ブロック・テキストは折り返されず、報告書のテキスト行幅で切り捨てられます。ただし、報告書のテキスト行幅が DEFAULT で、行の折り返し幅が NONE でない場合は、明細ヘッダー・テキストおよび明細ブロック・テキストは行の折り返し幅で切り捨てられます。

この入力域の値が印刷幅より大きい場合は、報告書の列内のデータは右端で切り捨てられます。

行を折り返したい場合（すなわち、報告書の明細行を 1 つの行から始め、それを次の行以降に連続させる場合）は、この入力域に数値をタイプして、報告書のデータ行の最大幅を指示します。それによって、できるだけ多く

## FORM.OPTIONS

の列が報告書の横方向に位置づけられ、残りの列は、報告書のそれ以降の行(1行または複数行)に置かれます。折り返し行はすべて列間隔がとられて始まり、次に表データを組み込みます。

列と列間隔が広すぎて、指定された行の折り返し幅に収まらない場合でも、その列については改行されず、列の右端が切り捨てられます。

列ヘッダー、表データ、および列合計だけが、指定された幅で折り返されます。報告書内のその他のデータは、すべて通常どおりフォーマット設定されます。

図 27 は、行折り返し(幅 35)と、表データ行送りが 2 の報告書の一部を示しています。

ID	NAME	DEPT	JOB
YEARS	SALARY	COMM	
160	MOLINARE	10	MGR
7	22959.20	-	
210	LU	10	MGR
10	20010.00	-	
240	DANIELS	10	MGR
5	19260.25	-	

図 27. 報告書における行折り返し

### C 報告書のテキスト行幅

**報告書**：報告書の最終テキスト、明細ヘッダー・テキスト、明細ブロック・テキスト、切れ目テキストの幅を指定します。この入力域には、DEFAULT、COLUMNS、または 1 から 999999 までの値のいずれかを指定できます。

#### DEFAULT

切れ目後書きテキストおよび最終後書きテキストは、FORM.COLUMN および FORM.MAIN で示された最初の合計列までの全列の合計幅を使用します。

#### COLUMNS

すべてのテキスト域が、FORM.COLUMNS および FORM.MAIN で示された前列の合計幅を使用します。(明細ヘッダー・テキストと明細ブロック・テキストでは、このオプションは DEFAULT と同じです。)

#### 0 から 999999

すべてのテキスト・タイプの幅を表す文字数。0 は、テキストを設定しないことを意味します。

### D 報告書の固定列の数

**報告書**：報告書を横にスクロールしても位置が変わらない列の数を指定します。固定列が指定されると、報告書は固定域とスクロール可能域に分けら

れます。複数ページにわたる印刷報告書では、固定列は各ページの左端に繰り返し印刷されます。印刷報告書のスクロール可能域は、ページ分割時に変更する区域に関係します。

指定できる値は、1 から 999 の任意の数値、または NONE (デフォルト値) です。

指定した値が報告書内の列の総数より大きいと、すべての列が固定されます。取り扱いが OMIT の列は、固定列としてカウントされません。

固定列は列の再配列 (順) と併用できます。列を並び替えた後、 $n$  個の列を固定列として選択すると、新しい並び順で最初の  $n$  個の列が固定列になります。これは、自動再配列でもユーザーによる再配列でも同じです。

報告書の固定列域は、報告書のテキストに影響を与えることがあります。固定列内にある切れ目、明細、最終テキストの各部分は、報告書の印刷ページの左側で繰り返されます。スクロール可能域内にある切れ目、明細、最終テキストの部分は、印刷報告書の最初のページに現われますが、ページ分割が起きる場合後続のページには現われません。

ページ・ヘッダーおよび後書きテキストは、表示報告書または印刷報告書のいずれでも、固定列設定値により影響を受けません。

固定列は、他の報告書オプションと矛盾することがあります。固定列では行の折り返しを使用できません (**B** 表の折り返し幅 を参照)。また、報告書内のすべての固定列の幅の合計が表示可能な画面幅より大きいと、表示報告書にも印刷報告書にも影響が出ます。表示報告書では、報告書を上下にスクロールできますが、左右にはスクロールできません。印刷報告書では、次のメッセージが表示されます。

*The report cannot be printed; the fixed area is too wide.*

#### **E 切れ目列の一括表示**

**報告書**：いずれかの列に取扱コードとして BREAK を割り当ててあるとき、その BREAK 列の値をいつ表示するのかを指定します。値が変わったときだけ表示するのか、報告書のすべての行に表示するのかを指定してください。

**YES** BREAK 列の値を、それが変化したときだけ表示します。

**NO** BREAK 列の値を、報告書のすべての表データ行に表示します。

一括表示は、ページの上端から始めます。その値は、前のページの最下行から変わっていない場合でも、ページの上端に印刷されます。

#### **F デフォルト切れ目テキスト (\*)**

**報告書**：報告書に含まれている制御の切れ目に対してとくに切れ目後書きテキストを指定していないとき、BREAK 総計行の目印として切れ目後書きテキストを生成するかどうかを、ここで指定します。

デフォルトの切れ目テキストは、最高の切れ目レベルのテキストについては、アスタリスク 1 つ、2 番目の切れ目レベルのテキストについてはアスタリスク 2 つ (以下同様) から構成されます。

## FORM.OPTIONS

### G グループ化の場合の列ヘッダーに関数名を使用する？

**報告書**：報告書に総計データ（たとえば、列の合計結果）があり、取扱コード GROUP で表データ行を抑止しているときは、この入力域で総計列のヘッダーを決定します。

**YES** 列ヘッダーの一部として集約のタイプを示す言葉を表示します。

**NO** 列ヘッダーに集約名を表示しません。

**図表：**図表に YES を使用すると、図表の凡例内に関数名が表示されます。 NO を指定することをお勧めします。

### H 列折り返し行のページ内表示

**報告書**：報告書の 1 つ以上の列について列内での折り返しを指定しているとき、その折り返された列が 2 ページにまたがってもよいかどうかを、ここで指定します。

**YES** 折り返しの列がページ行数を超えない限り、同じページに列の折り返し行を入れます。

**NO** 必要に応じて、折り返し列がページにまたがることを許可します。

### I 横方向合計列の表示

**報告書**：自動生成される「横方向要約」列を表示するかどうかを指定します。横方向合計列フィールドは、指定された列を横方向に 合計した列を新たに追加します。

図 28 に示した ACROSS 報告書で、部門 10 から 部門 84 までの各行を横方向に見ると、部門の職種別の平均給与とその部門の平均給与（最後の列）がわかります。その職種全体の平均給与は、各列の一番下（最終合計区切り線の下）に示されています。

DEPT	<----- JOB ----->			
	<- CLERK -->	<- MGR -->	<- SALES -->	<- TOTAL -->
	AVERAGE SALARY	AVERAGE SALARY	AVERAGE SALARY	AVERAGE SALARY
10				20865.86
15	12383.35	20659.80	16502.83	15482.33
20	13878.68	18357.50	18171.25	16071.53
38	12482.25	17506.75	17407.15	15457.11
42	11007.25	18352.80	18001.75	14592.26
51	13914.90	21150.00	18555.50	17218.16
66	10988.00	18555.50	18844.23	17215.24
84	13030.50	19818.00	16649.25	16536.75
	=====	=====	=====	=====
	12612.61	19805.80	17869.36	16675.64

図 28. 列全体にわたる平均を示した報告書

横方向合計列は、報告書の右端に表示されます。

取扱コードが PCT、CPCT、CSUM のいずれかである列が少なくとも 1 つある ACROSS 報告書では、合計ごとに 2 行のデータ行を使うことができます。ただし、横方向合計列と最終合計がともに存在するか、ともに存在しない場合に限られます。

1 つの合計につき 2 行が返されてくるとき、2 番目の合計データ行には、PCT、CPCT、CSUM 指定列の値だけが含まれています。そのような列の場合、最初の行にある値は、ACROSS 横方向（グループ）合計を基準にしたサブカテゴリーの合計値です。2 番目の行にある値は、ACROSS 縦方向（サブカテゴリー）合計を基準にしたサブカテゴリーの合計値です。

横方向合計列が省略される (FORM.OPTIONS で) と、ACROSS 横方向合計値も省略され、グループごとに 1 行 (ACROSS 縦方向値を含む 1 行) だけがフォーマット設定されます。

最終合計が省略される (FORM.FINAL で) と、ACROSS 縦方向値も省略され、グループごとに 1 行 (ACROSS 横方向値を含む 1 行) だけがフォーマット設定されます。

**図表:** 使用できる 2 行の「横方向」要約の行データのうち、ICU に転送できるのは 1 行だけです。図表には、2 つのデータ行をともに表示することはできません。各グループ内の 1 つの列に 2 つの値が存在する場合は、2 番目の行 (ACROSS 縦方向) の値が ICU に渡され、図表に示されます。

最終合計を省略すると、ACROSS 横方向の値を強制的に図表に表示することができます。この場合は、ACROSS 縦方向の値は省略されます。

## J 報告書列の自動再配列

**報告書 :** 取扱コードの BREAK $n$  か GROUP、またはいずれかの集約関数 (AVERAGE、COUNT、FIRST、LAST、MAXIMUM、MINIMUM、STDEV、SUM、CPCT、CSUM、PCT、TPCT、TCPCT など) を指定するとき、報告書の列の再配列を自動的に行なわせるかどうかを、ここで指定します。

デフォルトは NO です。(列は自動的には再配列されません。たとえ、取扱コードの BREAK $n$  や GROUP、または集約関数の 1 つを使用していても、各列は、FORM.MAIN または FORM.COLUMNS に示されるとおりに報告書に並びます。)

YES を指定すると、列は次の規則に従って並べ替えられます。

- BREAK $n$  列が左端
- GROUP 列が BREAK $n$  列の右
- すべての非集約列が BREAK $n$  列と GROUP 列の右
- すべての集約列が右端

取扱コードに ACROSS を使用している場合、列の再配列ができないと ACROSS 報告書の意味がありませんから、この入力域の値は無視されます。

**図表:** 報告書の列の自動再配列を YES を指定すると、図表の X 軸に対してどの Y データ列が選択されるかに影響することがあります。列の自動再配列が効果を発揮するためには、次の条件が満たされていなければなりません。

- 書式で、図表の X 軸に対する Y データ列の選択に GROUP 取扱コードも BREAK $n$  取扱コードも使用されていないこと。
- 書式で、列の 1 つに集約関数 (AVERAGE、SUM、COUNT など) が使用されていること。

## FORM.OPTIONS

上記の条件が満たされると、総計列が報告書の左端から右端へ移動されます。たとえば、YEARS はもとは報告書の左端に並んでいたとします。したがって、図表を表示すると、YEARS 列が X 軸にプロットされました。(X 軸に対するデータ列の選択に GROUP も BREAK も指定されていません。) いま、YEARS に集約関数 AVERAGE を使用することにしたとします。この場合、YEARS 列は報告書の右端に移動します。 もはや左端の列ではありませんから、図表の X 軸にはプロットされません。 X 軸にプロットされるのは、いま報告書の左端にある列です。

### K 最高位の切れ目ごとに新しいページ番号

**報告書**：最高位の切れ目をもつ制御列の値が変わるたびに、印刷報告書のページを変え、また番号 1 から始めるかどうかを指定します。最高位の切れ目は、最も小さな番号をもちます。このオプションは印刷報告書にしか影響しません。オンライン報告書は、QMF によって長い 1 ページとして扱われます。

このオプションにデフォルト値 (NO) を使用すると、最高位の切れ目列の値が変わっても、報告書のページ番号を 1 から再開しません。この入力域に YES を入力すれば、ページ番号は 1 から再開されます。ここで YES を指定しても、書式で少なくとも 1 つの BREAK 取扱コードを使用し、対応する FORM.BREAKn パネルの切れ目での改ページ入力域に YES を入れておかないと、ここに YES は無視されます。

### L 列ヘッダー

**報告書**：列ヘッダーと表データ行を分ける破線を報告書に表示するかどうか指定します。

### M 切れ目合計

**報告書**：切れ目合計と切れ目メンバー行を分ける等号を表示するかどうか指定します。

### N 横方向ヘッダー

**報告書**：ACROSS 報告書で列を示す破線と矢印を表示するかどうか指定します。

### O 総最終合計

**報告書**：最終合計と報告書本文とを分ける等号を表示するかどうか指定します。

---

## FORM.PAGE

FORM.PAGE は、報告書のページ・ヘッダーとページ後書きの内容と配置について細かな選択を行なうために用いられます。オンライン報告書と印刷報告書では扱いが異なります。QMF は、オンライン報告書の一番上にヘッダー、一番下に後書きを配置します。印刷報告書では、各ページの一番上にヘッダー、一番下に後書きを配置します。

FORM.MAIN パネルの区域 **G** (195 ページの『FORM.MAIN』を参照) には、報告書のページ・ヘッダーとページ後書きを指定します。FORM.MAIN の **G** 域に指定した内容が、FORM.PAGE にも表示されます。同様に、FORM.PAGE でページ・ヘッダーとページ後書きを指定すると、その 1 行目が FORM.MAIN に表示されます。

図 29 は、FORM.PAGE パネルの入力フィールドを示しています。

FORM.PAGE															
<b>A</b>	Blank Lines Before Heading ==> 0			<b>B</b>	Blank Lines After Heading ==> 2										
<b>C</b>	LINE	<b>D</b>	ALIGN	<b>E</b>	PAGE HEADING TEXT										
-----+-----1-----+-----2-----+-----3-----+-----4-----+-----5-----+															
1	CENTER			2				3				4			
2	CENTER			3				4							
3	CENTER			4											
4	CENTER														
<b>F</b> Blank Lines Before Footing ==> 2 <b>G</b> Blank Lines After Footing ==> 0															
<b>H</b>	LINE	<b>I</b>	ALIGN	<b>J</b>	PAGE FOOTING TEXT										
-----+-----1-----+-----2-----+-----3-----+-----4-----+-----5-----+															
1	CENTER			2				3				4			
2	CENTER			3				4							
3	CENTER			4											
4	CENTER														
*** END ***															
1=Help	2=Check	3=End	4>Show	5=Chart	6=Query										
7=Backward	8=Forward	9=	10=Insert	11=Delete	12=Report										
OK, FORM.PAGE is displayed.															
COMMAND ==>															
SCROLL ==> PAGE															

図 29. FORM.PAGE パネルの入力フィールド

#### **A** ヘッダー前のブランク行数

**報告書:** ページの最上行からページ・ヘッダーの最初の行の間のブランク行を指定します。この値には、1 ~ 999 を指定できます。

**図表:** この区域への入力により、図表のヘッダーの垂直方向の配置が決まります。しかし、ブランク行数が多過ぎると、Y 軸のラベルが変化することがあります。

#### **B** ヘッダー後のブランク行数

**報告書:** ページ・ヘッダーの最後の行と報告書の本文との間のブランク行数を指定します。この値には、1 ~ 999 を指定できます。デフォルト値は 2 です。

#### **C** 行 報告書:

ページ・ヘッダー・テキストの各行に番号を付けると共に、それらの行自体およびページ・ヘッダー開始行（ヘッダー前のブランク行数入力域で指定）との相対的な位置を指定します。

指定する数値は、それぞれ必ずしも 1 から始まる必要はなく、連続している必要もありません。ページ・ヘッダーの行間、およびページの最上行とページ・ヘッダー・テキストの最初の行のあいだを任意に空けることができます。ブランクの場合は、対応するテキストは無視されます。

たとえば、FORM.PAGE に次の値があるとします。

```
LINE ALIGN PAGE HEADING TEXT
-----+-----1-----+-----2-----
4 LEFT MONTHLY INVENTORY
4 RIGHT PAGE &PAGE
2 CENTER ABC COMPANY
```

これらの値は作成される報告書の中に次のように表示されます。

ABC COMPANY  
MONTHLY INVENTORY PAGE 1

**図表:** 行を使って、ヘッダー・テキスト行の垂直方向の位置を、テキスト自体および図表(ページ)ヘッダー開始行と相対的に指定します。

#### D 調整(位置合わせ)

**報告書:** 報告書上でのページ・ヘッダー・テキスト各行の水平位置を指定します。報告書の幅の範囲内であれば、どこでも行を位置づけることができます。ここでの幅は、オンライン報告書の場合は、表示される報告書の幅であり、印刷報告書の場合は、ページの幅になります。

<b>Left</b>	ページ・ヘッダー・テキストの行を左寄せします。
<b>Right</b>	ページ・ヘッダー・テキストの行を右寄せします。
<b>Center</b>	ページ・ヘッダー・テキストの行を中央に合わせます。
<i>n</i>	ページ・ヘッダー・テキストを、その行の <i>n</i> 桁目から開始します。ここで、 <i>n</i> は、1 から 999999 の任意の数値を指定できます。
<b>Append</b>	前のページ・ヘッダー・テキスト行の終わりに、行を付加します。APPEND をページ・ヘッダー・テキストの最初の行に使用すると、テキスト行は左寄せになります。  付加されるテキスト行は、付加先のテキスト行と同じ「行」値をもっていかなければなりません。報告書の幅が不十分で、付加されたテキスト行の全体が入りきらないと、その一部が切り捨てられることがあります。たとえば、FORM.PAGE に次のように入力されたとします。

LINE	ALIGN	PAGE	HEADING	TEXT
---	-----	---	-----+-----1-----+-----2-----	---
1	CENTER	ABC COMPANY	MANAGERS	--
1	APPEND		&DATE,	&TIME
3	CENTER			
4	CENTER			
5	CENTER			

これらの入力により、列は図 30 に示すように配置されます。

ABC COMPANY MANAGERS -- 98/08/04, 14:20						
ID	NAME	DEPT	JOB	YEARS	SALARY	COMM
10	SANDERS	20	MGR	7	18357.50	-
30	MARENghi	38	MGR	5	17506.75	-

図 30. 報告書の行への行の追加

**図表:** LINE(行)値を使ってヘッダーと同じ行に複数のテキスト行を置く場合を除き ALIGN(調整)値が図表ヘッダーに影響することはありません。

## E ページ・ヘッダー・テキスト

**報告書:** 印刷報告書の各ページの最上行、または表示報告書の最初の行の前に現れるようにするテキストを入力します。INSERT コマンドで、最大 999 行のページ・ヘッダー・テキストを追加できます。テキストの各行の長さは、最大 55 文字です。「調整」(ALIGN) 値に APPEND を指定するか、具体的な水平位置を指定すれば、その行にテキストを追加できます。

使用しているサイトが DBCS データをサポートする場合は、272 ページの『2 バイト文字を含む名前』を参照してください。

報告書のページ・ヘッダー・テキストに大文字と小文字の両方を使いたいときは、プロファイルの CASE 値に STRING か MIXED を指定してください。これらのオプションの詳細は、126 ページの『SET PROFILE』を参照してください。

ページ・ヘッダーには次の変数値を含めることができます。

**&n** *n* は、報告書の現行ページ上にある列 *n* の、最初の値を表します。列 *n* は、データベースから選択された *n* 番目の列、または FORM.MAIN と FORM.COLUMNS に示されている *n* 番目の列です。

**&ROW** 現在のページの最初のデータ行の番号が報告書に印刷または表示されます。

**&DATE** 現在の日付

**&TIME** 現在の時刻

**&PAGE** 現行ページ番号

&DATE、&TIME、または &PAGE がページ・ヘッダー・テキストに入力されると、システム日付、時刻、またはページ番号は印刷報告書の最下行に印刷されません。これは、FORM.PAGE でこれら 3 つの変数を入力した場合にだけ適用されます。

&DATE、&TIME、および &PAGE の説明は、198 ページの『FORM.BREAKn』を参照してください。

**図表:** このトピックでのページ・ヘッダー・テキストに関する説明は、「調整」の説明部分を除き、図表にもあてはまります。「調整」に指定した値で図表ヘッダーが左右されるのは、FORM.PAGE に入力された 1 行以上のテキスト行を、「行」を使ってフォーマット設定後の報告書の同一行に配置した場合に限られます。「行」機能を使用していなければ、図表ヘッダーは自動的に中央に合わせられます。

## F 後書き前のブランク行数

**報告書:** 報告書の本文からページ後書きテキストの最初の行までのブランク行数を指定します。指定できる値は 1 から 999 までです。デフォルト値は 2 です。

## G 後書き後のブランク行数

**報告書:** ページ後書きテキストの最後の行からページの最下行までのブランク行数を指定します。指定できる値は 1 から 999 までです。

## FORM.PAGE

切れ目合計データと 1 つ以上の折り返し列を含んでいるような報告書では、ここに指定する値がある程度大きくしないと、合計データ行を全部見られないことがあります。詳細は、262 ページの『文字データの編集コード』の CW コードの説明を参照してください。

- H 行 報告書:** ページ後書きテキストの各行に番号を付けると共に、それらの行自身およびページ後書き開始行(後書き前のブランク行数入力域で指定)との相対的な位置を指定します。1 から 999 までの任意の数値またはブランクを指定できます。

たとえば、FORM.PAGE に次の値があるとします。

```
LINE ALIGN PAGE FOOTING TEXT
---- ----- -----+---1---+---2---
3 LEFT   MONTHLY INVENTORY
3 RIGHT  PAGE &PAGE
2 LEFT   ABC COMPANY
```

これらの値は作成される報告書の中に次のように表示されます。

```
ABC COMPANY
MONTHLY INVENTORY      PAGE 1
```

テキスト第 1 行の前に 1 行のブランク行が置かれることに注意してください。

### I 調整

**報告書:** 報告書上でのページ後書きテキスト各行の水平位置を指定します。テキスト行は、左マージンと右マージンの間のどこにでも配置できます。ここで幅は、オンライン報告書の場合は、表示される報告書の幅であり、印刷報告書の場合は、ページの幅になります。

**Left** ページ後書きテキストの行を左寄せします。

**Right** ページ後書きテキストの行を右寄せします。

**Center**

ページ後書きテキストの行を中央に合わせます。

*n* ページ後書きテキストを、その行の *n* 桁目 から開始します。ここで、*n* は、1 から 999999 の任意の数値を指定できます。

**Append**

前のページ後書きテキスト行の終わりに、行を付加します。ページ後書きテキストの最初の行(最も小さな「行」値をもつテキスト行)に APPEND を使用すると、そのテキスト行は左寄せされます。

付加されるテキスト行は、付加先のテキスト行と同じ「行」値をもっていなければなりません。報告書の幅が不十分で、付加されたテキスト行の全体が入りきらないと、その一部が切り捨てられることがあります。

たとえば、FORM.PAGE に次のように入力されたとします。

```
LINE ALIGN PAGE FOOTING TEXT
---- ----- -----+---1---+---2---
1 CENTER ABC COMPANY MANAGERS --
1 APPEND &DATE, &TIME
```

これによって、列の調整は次のように変わります。

ID	NAME	DEPT	JOB	YEARS	SALARY	COMM
.	.	.	.	.	.	.
10	SANDERS	20	MGR	7	18357.50	-
30	MARENGHI	38	MGR	5	17506.75	-

ABC COMPANY MANAGERS -- 98/08/04, 16:20

**J ページ後書きテキスト**

**報告書:** 印刷報告書の各ページの最下行、または表示報告書の最後の行の前に現れるようにするテキストを入力します。INSERT コマンドで、最大 999 行のページ後書きテキストを追加できます。テキストの各行の長さは、最大 55 文字です。「調整」(ALIGN) 値に APPEND を指定するか、具体的な水平位置を指定すれば、その行にテキストを追加できます。

使用しているサイトが DBCS データをサポートする場合は、272 ページの『2 バイト文字を含む名前』を参照してください。

報告書のページ後書きテキストに大文字と小文字の両方を使いたいときは、プロファイルの CASE 値に STRING か MIXED を指定してください。これらのオプションの詳細は、126 ページの『SET PROFILE』を参照してください。

ページ後書きには、次の変数値を含めることができます。

**大域変数**

SET GLOBAL を使用して、ページ後書きテキストで使う変数をセットします。このコマンドの詳細については、123 ページの『SET GLOBAL』を参照してください。

**&n** *n* は、この報告書の現ページ上で処理された列 *n* の最後の行を表します。列 *n* は、データベースから選択された *n* 番目の列、または FORM.MAIN と FORM.COLUMNS に示されている *n* 番目の列です。

**&ROW**

現在のページの最終データ行の番号が報告書に印刷または表示されます。

**&DATE**

現在の日付

**&TIME**

現在の時刻

**&PAGE**

現行ページ番号

ページ後書きテキストに &DATE、&TIME、&PAGE を含めておくと、印刷報告書の末尾にはその値が（システムの日付、時刻、ページ番号ではなく）印刷されます。これは、FORM.PAGE でこれら 3 つの変数を入力した場合にだけ適用されます。

&DATE、&TIME、および &PAGE の説明は、198 ページの『FORM.BREAKn』を参照してください。

## QMF の書式エラーの評価

QMF は 2 種類の誤りを区別します。

- エラー状態 - 訂正しなければその書式を使用できないような誤り
- 警告状態 - 訂正しなくとも書式を使用できる程度の誤り

### エラー状態

エラー状態は、入力域に誤った値を入れることから生じます。たとえば、FORM.OPTIONS の「切れ目列の一括表示」(OUTLINE) フィールドに Y0 と入れたとします。Y0 は、この入力域に許された値ではありませんから、エラーになります。

報告書の表示を妨げるような矛盾があるときも、エラーが起こります。たとえば、SUM は数値列の取扱コードとして有効な項目です。しかし、SUM を文字データの列に入力すると、エラーになります。

エラーを訂正しないと、その書式を使用して報告書を表示することはできません。ただし、エラーのある書式でも、保管、インポート、エクスポート、表示、印刷はできます。

エラーが訂正されると、QMF は次に警告状態の有無を調べます。

### 警告状態

ある入力域の値が他の入力域の値と矛盾していると、警告状態が生じます。エラーと異なり、警告状態では、訂正しないまま書式を使うことも可能です。QMF は矛盾があることをユーザーに伝え、その状態を適宜解釈して報告書または図表の書式を設定します。

QMF から示された報告書または図表は、そのまま受け入れることもできるし、矛盾している 1 つ以上の値を変更して、書式を訂正することもできます。

表 25 に、警告状態と、それぞれの場合に QMF がどのように報告書をフォーマットするかをリストしています。これらの警告状態が、その報告書の図表表現に影響することもあります。

表 25. フォーマット設定の警告状態

Condition	QMF のアクション
複数の ACROSS 取り扱い	最初の ACROSS を受け入れ、報告書から残りの ACROSS 列を除きます。
ACROSS 取り扱いがあって、GROUP 取り扱いがない	報告書から ACROSS 列を除きます。
GROUP 取り扱いがあって、集約取り扱いがない	報告書から GROUP 列を除きます。

表 25. フォーマット設定の警告状態 (続き)

Condition	QMF のアクション
ACROSS 取り扱いと GROUP 取り扱いがある、1つ以上のブランク取り扱いがない	集約するなら、報告書からブランク取り扱いの列を除きます。集約しないなら、報告書から ACROSS 列と GROUP 列を除きます。
GROUP 取り扱いがある、少なくとも 1つ以上の集約取り扱いと 1つ以上のブランク取り扱いがある	報告書からブランク取り扱いの列を除きます。
行の折り返しがある、ACROSS 取り扱いまたは列内折り返し編集コードがある	行の折り返しを無視します。
ACROSS 取り扱いがある、列の自動再配列がない	列の自動再配列オプションの値を無視し、標準の ACROSS 報告書を作成します。

## 誤りの検査とその訂正

通常、書式パネルが表示されている時に Enter キーを押すと、カーソルがコマンド行に位置づけられます。しかし、書式に 1つ以上の誤った値を入力して、その後に Enter キーを押すと、QMF はエラーを強調表示して、最初のエラーを記述するメッセージをユーザーに送ります。Enter キーを押しても、以前の対話で犯したエラーは指摘されません。

再度 Enter キーを押すと (最初のエラーが訂正されていてもいなくても) QMF はコマンド行にカーソルを位置づけます。書式に含まれる次のエラーに関するメッセージを受け取るには、CHECK コマンドを使用してください (15 ページの『CHECK』を参照)。

書式を使用するコマンドが出されると、QMF はそのたびに書式を検査し、エラーの有無を調べます (たとえば、DISPLAY REPORT、PRINT CHART、PRINT REPORT、EXPORT REPORT、EXPORT CHART、または FORM オプションを指定した RUN QUERY など)。コマンドはコマンド行に入れても、ファンクション・キーを押しても出せます。書式が表示されるときにも、QMF はエラーの有無を検査します。

書式に含まれる式にエラーがある場合、QMF が REXX に値を渡して評価させまるまで、そのエラーは検出されません。書式を表示している間に、QMF コマンドを入れると (CHECK、DISPLAY REPORT、DISPLAY CHART、PRINT REPORT、PRINT CHART、FORM オプションつきの RUN QUERY を除きます)、書式にエラーがあるてもなくとも、QMF はそのコマンドを処理します。このとき表示されるメッセージは、入力されたコマンドに関するメッセージです。したがって、書式にエラー状態または警告状態があっても、その書式の表示、保管、インポート、エクスポートは可能です。

## 書式とデータの不整合

書式に変更を加えた結果、書式とデータに不整合が生じことがあります。この状態は、エラー状態や警告状態とは別個に扱われます。カーソルを位置づけても画面のいちばん上にエラー・メッセージは表示されず、CHECK コマンドを出しても問題は指摘されません。その代わり、ユーザーが報告書を表示しようとするメッセージが表示され、不整合を含む書式パネルが表示されます。

## QMF の書式エラーの評価

不整合を回避するために、次のガイドラインに従ってください。

- 書式内の列数 (定義列を除く) とデータ内の列数は等しくなければなりません。
- 書式内の編集コードが、データ内の各列のデータ・タイプに一致していなければなりません。
- データ内のすべての LONG VARCHAR、LONG VARGRAPHIC、XML、または DECFLOAT 列は、書式内でブランクか OMIT 取扱コードをもっていなければなりません。

## QMF 書式での REXX の使用

### CICS ユーザーへの注

FORM.CALC、FORM.CONDITIONS、および FORM.COLUMNS (列定義) では、REXX で書かれた式を使用します。QMF は CICS ではこれをサポートしていません。

FORM.CALC、FORM.CONDITIONS、FORM.COLUMNS (列定義) で使用する式は、項 (ストリング、記号、関数) とその間に挿入する演算子および括弧で構成されます。REXX 式またはプログラムの内部から (呼び出し可能インターフェースまたはコマンド・インターフェースを用いて) QMF コマンドを実行してはなりません。

ストリング は、单一引用符か二重引用符で囲まれたリテラル定数です。たとえば、'High' と "Low" は、どちらもストリングです。

記号 は、数値リテラル (数値)、変数、引用符で囲まない非数値リテラルのいずれかです。

- 数値リテラル は、整数、10 進数、指数表記で表現できます。たとえば、

```
123  
25.45  
.432  
1.7E4 (equivalent to 17000)  
7.6e-3 (equivalent to .0076)
```

小数点として使う場合を除き、コンマは使用できません。(QMF でコンマを小数点に使用するには、データベース・マネージャーに対してそう定義しておかなければなりません。)

- 変数 は、式の用法によって制約されます。使用可能な変数の要約については、269 ページの『書式で使用される変数』 の表を参照してください。
- 非数値リテラル は、数値でも変数でもない記号です。式の評価では、ストリング同様に扱われます。

関数 の構文は、次のとおりです。

```
function-name([[expression][,][expression][,] ...])
```

この構文では、0 から *n* 個の expression 引数を使用できます (ここで、*n* は、REXX で許されているコンマで区切られた式の最大数です)。

上記の構文で、*function-name* には、組み込み関数か外部関数（たとえば、REXX プログラム）を指定しなければなりません。式の評価は左から右へ行なわれますが、一般的な代数計算同様、括弧と演算子の優先順序に従います（負符号は例外）。詳細については、252 ページの『演算子の優先順位』を参照してください。

## 報告書での計算値の使用

QMF 報告書に計算値を盛り込む方法には、次の 3 通りがあります。

- SQL ステートメントを使って照会に計算を含める。
- 式に基づいた新しい列を定義する。
- FORM.CALC パネルで定義された式を指定して使用する。

報告書に計算を含める第 1 の方法はデータベースによって扱われ、他の 2 つの方法は、書式の仕様に基づいて QMF によって扱われます。書式に評価が含まれていると、それは REXX で処理されます。

照会によって書式がロードされ、インポートされ、表示され、書式で実行されると、QMF はそのたびに条件や列定義とともに式が正しいことを確認します。条件、列定義、または式に変更があると、QMF はそれが正しいかどうか再度確認します。検査中に QMF から予期しないデータが渡されると、REXX エラーが発生することがあります。計算を、その妥当性を検査するステートメントとともに REXX プログラムに含めておくと、この種の REXX エラーを避けることができます。

FORM.CONDITIONS または FORM.COLUMNS（列定義）を使用する場合は、同じパラメーターを使って式またはプログラムを何回呼び出しても、必ず同じ値が返されるようにしてください。プログラムがいつも同じ値を返してこないと、期待どおりに切れ目が起こらず、合計値が印刷された結果と一致しないこともあります。

データベースが行なう計算と REXX による評価では、パフォーマンス、能力、柔軟性に相当な違いが現れることがあります。REXX プログラムは、複雑な論理で処理された値や、REXX 関数で処理された値を返すことができます。しかし、REXX では、使用する関数やプログラミング・オプションに幅広い選択がありますが、報告書に含める計算のすべてを REXX に依存することには欠点もあります。

REXX で式を評価するには、ある程度のリソースが必要です。1 つの報告書を完成させるのに REXX を頻繁に呼ぶようだと、パフォーマンスに悪影響が出るかもしれません。このため、計算の一部を照会の中で行なうという選択もあります。たとえば、報告書に、次の計算に基づいた新しい列を作成する必要があるとします。

```
((Column A - Column B) * 100) / Column B
```

列を作成するには、式を SQL で入力して照会を実行しなおすことも、新しい列の定義として書式に式を入れ、その報告書を表示することもできます。書式で式を定義すると、報告書作成のために明細行を 1 行処理するごとに REXX が呼ばれますから、この新しい列は照会で定義したほうがよいかもしれません。

## QMF と REXX の対話

QMF は、DSQCXPR プログラムを REXX 機能として呼び出して REXX の式を解釈します。式を解釈する際、次の順序のイベントが起ります。

1. 該当する場合は、PASS NULLS リテラルが置換されます。

## QMF 書式での REXX の使用

2. 式の中で大域変数と置換変数はすべて置き換えられ、二重引用符で囲まれます。
3. 式は連結されて、"DSQ\$#VAL=" になります。
4. REXX が呼び出され、プログラム名 (DSQCXPR) と引数リスト (式) が渡されます。
5. DSQCXPR が式のために REXX インタープリター命令を呼び出します。
6. 構文エラーが取り込まれます。
7. DSQ\$#VAL 記号を介しての式からの結果またはエラー結果が戻されます。

@IF ルーチンは次のことを行います。

- 少なくとも 3 つの引数が渡されることを検証します。
- 奇数の引数が渡されることを検証します。
- 奇数の引数 (比較) が解釈されます。最初の式が真である場合は、次の式が解釈されて結果が戻され、そして、またその次の式が同様に処理されていきます。
- 奇数の引数がどれも真でない場合、最後の引数が解釈され、戻されます。

詳細については、185 ページの『@IF 関数』を参照してください。

REXX 式に数値があっても、QMF はそれを二重引用符で囲みません。このため、式に負の値が含まれていると、それが負の値として扱われないことがあります。負符号が減算記号として扱われるなどを防ぐためには、REXX に渡される変数同士を(スペースでなく)コンマで区切るか、すべての負の値を(結果的に負の値になる)置換変数も含めて二重引用符で囲んでください。たとえば、myexec(A -1) は評価エラーを起こしますが、myexec(A,-1) や myexec("A" "-1") は起こしません。ただし、コンマを使用するときは、次の点に注意してください。

- 式で使用できるコンマの数には、制限があります。
- 解析ステートメントを変更して、コンマを解析オブジェクトに含める必要があるかもしれません。

REXX は、1 ストリングの最大長を制限しています。QMF は文字をストリングに追加するため、QMF による処理のあとにストリングの長さが制限を超えることがあります。REXX から QMF へ 32,767 バイトを超えるストリングが渡されると、そのストリングは 32,767 バイトに切り縮められます。

式におけるコンマとストリング長の制限については、IBM Publications Center ([www.ibm.com/shop/publications/order](http://www.ibm.com/shop/publications/order)) の中の TSO のプロシージャー型言語情報を参照してください。

パフォーマンス向上のためには、REXX 呼び出し可能インターフェースを通じて QMF を起動してください。

## REXX はいつ式を評価するか

FORM.CALC パネルで指定され、書式のテキスト域で置換変数 (&CALCn) として使われる式は、書式のどこに配置されるかによって、次の時点で REXX に渡され評価されます。

- 計算は、フォーマット設定時に処理されます。

- ある計算が、選択された FORM.DETAIL パネルで参照されていても、 SELECT=NO または SELECT=Cn (n 条件が偽) であると、その計算は実行されません。
- ある計算が 1 つのバリエーションの複数行にあるときは、複数回計算されることがあります。
- ある計算が、選択された複数の FORM.DETAIL バリエーションで参照されていて、パネル・バリエーションの選択が YES か Cn (n が『真』) であると、その計算は複数回実行されることがあります。
- FORM.CALC パネルに指定されていて、FORM.COLUMNS パネルで取扱コードとして使われている式は、その値がフォーマット設定のために必要とされると、REXX によって評価されます。
- FORM.COLUMNS 定義パネルに指定されている新列定義のための式は、データベースから行が 1 つ取り出されるたびに REXX によって評価されます。行は何度も取り出されることがあります（たとえば、ページ分割が必要な報告書の印刷のサポート、あるいは、TCPCT など、最初にすべてのデータを取り出しておくことが必要な取扱コードのサポートなど）。
- FORM.CONDITIONS パネルに指定され、FORM.DETAIL パネル・バリエーションで参照される式は、報告書に設定される各明細行につき少なくとも 1 回ずつ、REXX によって評価されます。

## REXX 演算子

### CICS ユーザー

FORM.CALC、FORM.CONDITIONS、列定義では、REXX で書かれた式を使用します。QMF は CICS ではこれをサポートしていません。

各演算子（接頭演算子を除く）は 2 つの項に作用します。これらの項とは、記号、関数、または括弧の中の副次式のことです。各接頭演算子は、後に続く項または副次式に作用します。このトピックでは、QMF 式で使用できる、算術、比較、連結、および論理（ブール）タイプの演算子について説明します。

### 算術演算子

<b>+</b>	加算
<b>-</b>	減算
<b>*</b>	乗算
<b>/</b>	除算
<b>%</b>	除算後、商の整数部分だけを返す
<b>//</b>	除算後、余りだけを返す（結果が負になることがあるので、モジュロではありません）
<b>**</b>	数を整数乗する（累乗）
<b>接頭部 -</b>	次の項を否定する
<b>接頭部 +</b>	次の項を現状のまま引き受ける

## 比較演算子

<code>==</code>	完全に等しい (合同)
<code>=</code>	等しい (数値として、または埋め込みをすれば)
<code>!==, /==</code>	完全には等しくない (== の逆)
<code>!=, /=</code>	等しくない (= の逆)
<code>&gt;</code>	より大きい
<code>&lt;</code>	より小さい
<code>&lt; &gt;</code>	等しくない
<code>&gt;=</code>	以上 (より大きいか等しい)
<code>&lt;</code>	より小さくない
<code>&lt;=</code>	以下 (より小さいか等しい)
<code>!&gt;</code>	より大きくはない

## 連結演算子

`||` 項を連結する (ブランクなし、または 1 個のブランクあり)

REXX には、他の連結演算子も用意されています。詳細は、IBM Publications Center ([www.ibm.com/shop/publications/order](http://www.ibm.com/shop/publications/order)) にある TSO のプロシージャー型言語の情報を参照してください。

## 論理 (ブール) 演算子

<code>&amp;</code>	AND (2 つの項がともに真なら、1 を返します)
<code> </code>	包含 OR (2 つの項の一方でも真なら、1 を返します)
<code>&amp;&amp;</code>	排他 OR (どちらか一方だけが真なら、1 を返します)
接頭部 <code>~</code>	論理 NOT (否定。1 は 0 になり、0 は 1 になります)

## 演算子の優先順位

式の評価は左から右へ行なわれます。この順序は、括弧と演算子の優先順位を使用して変更できます。

演算子の優先順位が明白でない場合には、括弧を使って意味を明確にしてください。括弧内の式が最初に評価されます。

次の式で、operator1 より operator2 のほうが優先順位が高いと、式 (term2 operator2 term3 ...) が先に評価されます。この規則が必要なだけ、適用されます。

term1 operator1 term2 operator2 term3 ...

たとえば、\* (乗算) は + (加算) より高い優先順位をもっています。したがって、 $3 + 2 * 5$  を評価すると 13 になり、厳密に左から右へ評価していった場合の 25 にはなりません。

演算子の優先順位は、高い方から低い方へ次のようになっています。

<b>+ - ^</b>	接頭演算子
<b>**</b>	累乗
<b>* / % //</b>	乗算と除算
<b>+ -</b>	加算と減算
<b>//</b>	連結 (ブランクつきまたはブランクなし)
<b>=, &gt;, ...</b>	すべての比較演算子
<b>&amp;</b>	AND
<b> , &amp;&amp;</b>	OR、排他 OR

& 演算子と && 演算子は、計算式では後に 1 個のブランクを置いて、置換変数と区別しなければなりません。

優先順位が等しい演算子同士 (たとえば、乗算演算子と除算演算子) は、左から右への計算規則に従います。

これらの優先順位が一般の代数と異なる点は、負の接頭演算子が指数演算子より高い優先順位をもつことです。したがって、 $-3^{**}2$  は 9 で、-9 ではありません。

## 報告書計算式の例

表 26 に示した例では、次のように仮定しています。

- &SUM1 has the value 1600
- &SUM2 has the value 400
- &DATE has the value "87/12/15"

表 26. 計算式の使用例

式	結果
&SUM2/25	16
&SUM2-&SUM1*.25	0
&SUM1+&SUM2 < 4000	1 (真)
' ' = "	1 (真)
' ' == "	0 (偽)
&SUM1+(&DATE<'88')&*&SUM2	2000
date(u) (組み込み関数)	"12/15/87"

次の式は、*date(u)* と同じ結果になります。

```
substr(&DATE,4,5) || "/" ||
substr(&DATE,7,8) || "/" ||
substr(&DATE,1,2)
```

この他の FORM.CALC で使用できる計算の例は、「DB2 QMF 使用の手引き」を参照してください。

## 取扱コード

QMF 取扱コードは、報告書および図表の作成にどのように列データを使用するのかを定義するものです。

このトピックでは、QMF 取扱コードについて、それぞれ簡単に説明します。これには、取扱コードと、取扱コードによる報告書と図表の変更方法の演習と例が記載されています。これらの取扱コードの追加情報と例は、「DB2 QMF 使用の手引き」を参照してください。

## ACROSS 取扱コード

**報告書:** 列に取扱コード、ACROSS が指定できるのは、1 つまたは複数の列が GROUP の取扱コードをもつ場合のみです。 GROUP 列が 1 つ以上あるときは、各グループ値の合計行に総計行からの結果を何組か含めることができます。つまり、ACROSS 列の値グループごとに 1 組ずつの結果があります。 ACROSS 列のヘッダーには、3 つのレベルがあります。

1. 書式に入力された列ヘッダー
2. 列内の 1 組の値
3. その組に含まれる値ごとに、集約列の列ヘッダー

複数の ACROSS 列があると、QMF は最初の ACROSS 列だけを受け入れ、残りの ACROSS 列を報告書から除きます。取扱コードに ACROSS をもつ 1 列が存在するときは、他に取扱コードにブランクを持つ列があってはなりません。 ACROSS 報告書に取扱コードがブランクになっている列があると、QMF はその報告書を作成するものの、すべてのブランク取り扱いの例を報告書から除きます。

AVG 取扱コードを用いた横方向合計報告書の例については、234 ページの『FORM.OPTIONS』の中の『I 横方向合計列の表示』を参照してください。

**図表:** 報告書について述べたことは、図表にもあてはまります。図表上の ACROSS は、ある大力テゴリー (たとえば部門) に含まれるなんらかのカテゴリー (たとえば、職種) のデータを、いくつかのサブカテゴリー (たとえば、販売、事務など) に細分化して表示します。これらのサブカテゴリーのデータは、棒グラフで表示されます。カラー・ディスプレイ装置では、サブカテゴリーごとに棒の色が異なります。

## 集約取扱コード

ここでは、2 種類の総計について説明します。

- 列データのまとめとしての要約:

AVERAGE	COUNT	FIRST	LAST
MAXIMUM	MINIMUM	STDEV	SUM

- 計算してデータ値を置き換え、中間結果と最終結果を得るための総計:

CSUM	PCT	CPCT	TPCT	TCPCT
------	-----	------	------	-------

表 27 に、どのデータ・タイプにどの集約取扱コードを使用できるかを示します。

表 27. データ・タイプ別の有効な取扱コード

データ・タイプ	有効な取扱コード
数値	AVG、COUNT、CPCT、CSUM、FIRST、LAST、MAX、MIN、PCT、STDEV、SUM、TCPCT、TPCT
文字、日付、時刻、 タイム・スタンプ	COUNT、FIRST、LAST、MAX、MIN
注: LONG VARCHAR 列と LONG VARGRAPHIC 列は、集約できません。これらのデータ・タイプで有効な取扱コードは、BLANK と OMIT だけです。これらのコードを使用する必要のある他のデータ・タイプについては、261 ページの『BLANK および OMIT 取扱コード』を参照してください。	

## 列データの総計

### 報告書:

集約取扱コードは、列内のデータを総計します。 総計の結果は、報告書の中間に小計として、末尾に合計として示されます。

<b>AVERAGE</b>	列の値の平均
<b>COUNT</b>	列の値の個数
<b>FIRST</b>	列の最初の値
<b>LAST</b>	列の最後の値
<b>MAXIMUM</b>	列の最大値
<b>MINIMUM</b>	列の最小値
<b>STDEV</b>	列の値の標準偏差
<b>SUM</b>	列の値の合計

文字、日付、時刻、タイム・スタンプ、漢字の各種データに MAXIMUM と MINIMUM を使用すると、QMF は EBCDIC 照合順序に基づくデータ比較を行ないます。数値データの最大と最小の決定には、QMF は代数的な比較を使用します。 MAX、MIN、FIRST、LAST では、結果にヌルが含まれていてかまいません。

DATE 値、TIME 値、TIMESTAMP 値に日付 / 時刻関数を適用すると、値のデータ・タイプが数値に変わります。したがって、結果の値を総計できます。

結果のフォーマットはその列の編集コードで決まりますが、COUNT、STDEV、およびパーセンテージ総計は例外です。 COUNT はどのタイプのデータにも適用できまですが、結果は常に整数になります。したがって、その結果は編集コード K でフォーマットされます。 STDEV、PCT、CPCT、TPCT、および TCPCT は編集コード L でフォーマットされます (詳細は 264 ページの『数値データの編集コード』を参照)。

### 図表:

これらの取扱コードに関する報告書での情報は、図表にも当てはまります。

## 集約取扱コード

AVERAGE、MAXIMUM、MINIMUM、STDEV、SUM は、いずれも QMF データの図表化に有用です。 FIRST や LAST は、図表フォーマットでは使い道がないかもしれません。

報告書の図表を表示するとき、次の値はヌル値として ICU に送られます。

- 報告書内のヌル値
- 長すぎて列幅に入らないデータ値
- 未定義値
- 算術オーバーフロー値

### 計算によるデータ値の置き換え

**報告書:** 次のコードで参照される集約関数は、列中の各明細行値を計算値で置き換え、報告書の末尾に集約の最終結果を示します。また、報告書の途中に小計として示すこともできます。

<b>CSUM</b>	列中の各値の累計
<b>PCT</b>	各値が合計に占める割合のパーセンテージ <ul style="list-style-type: none"><li>• 取扱コード BREAK または ACROSS を含む報告書では、切れ目グループまたは横方向グループ中の各値について、それが切れ目合計または横方向合計に占めている割合をパーセンテージで示します。</li><li>• その他のすべての報告書で、PCT は列合計に対する各列の値のパーセンテージを示します。</li></ul>
<b>CPCT</b>	列中の各値の累積パーセンテージ <ul style="list-style-type: none"><li>• 取扱コード BREAK または ACROSS を含む報告書では、切れ目グループまたは横方向グループ中の各値について、それが切れ目合計または横方向合計に占める割合を累積パーセンテージで示します。</li><li>• それ以外の報告書では、列中の各値が列合計に締める割合を累積パーセンテージで示します。</li></ul>
<b>TPCT</b>	各値が列合計に占めるパーセンテージの合計 <ul style="list-style-type: none"><li>• 取扱コード BREAK または ACROSS を含む報告書では、列中の各値が列合計に占める割合をパーセンテージで示します。</li><li>• それ以外の報告書では、列合計を表示します。</li></ul>
<b>TCPCT</b>	列中の各値の合計累積パーセンテージ <ul style="list-style-type: none"><li>• 取扱コード BREAK または ACROSS を含む報告書では、列中の各値が列合計に占める割合を累積パーセンテージで示します。</li><li>• それ以外の報告書では、列合計を表示します。</li></ul>

これらの集約は数値データにのみ作用します。列中のヌルは結果に含まれませんが、未定義値と数値オーバーフローは評価されます。結果の書式は、列の編集コードで決まります。

次に報告書を 4 種類提示します。これらの報告書で異なるのは、給与列の総計結果だけです。いずれも書式に指定された総計を行なっています。

## 報告書 1: SUM SALARY (合計)

NAME	JOB	SUM SALARY
MOLINARE	MGR	22959.20
LU	MGR	20010.00
DANIELS	MGR	19260.25
JONES	MGR	21234.00
		=====
		83463.45

## 報告書 2: CSUM SALARY (累計)

NAME	JOB	CSUM SALARY
MOLINARE	MGR	22959.20
LU	MGR	42969.20
DANIELS	MGR	62229.45
JONES	MGR	83463.45
		=====
		83463.45

## 報告書 3: PCT SALARY (パーセンテージ)

NAME	JOB	PCT SALARY
MOLINARE	MGR	27.51
LU	MGR	23.97
DANIELS	MGR	23.08
JONES	MGR	25.44
		=====
		100.00

## 報告書 4: CPCT SALARY (累計)

NAME	JOB	CPCT SALARY
MOLINARE	MGR	27.51
LU	MGR	51.48
DANIELS	MGR	74.56
JONES	MGR	100.00
		=====
		100.00

次に示す 2 つの報告書は、同じ報告書ですが、制御の切れ目を使用しています。最初の報告書は、PCT によって次の 2 点を示しています。

- 各給与が切れ目グループ合計に占める割合のパーセンテージ
- 各切れ目グループが列合計に占める割合のパーセンテージ

JOB	NAME	PCT SALARY
CLERK	JAMES	25.71
	KERMISCH	23.34
	NGAN	23.81
	SNEIDER	27.14
	*	-----
	*	41.61
MGR	HANES	52.95
	SANDERS	47.05
		-----

## 集約取扱コード

*	30.91
SALES PERNAL	52.41
ROTHMAN	47.59
-----	
*	27.47
=====	
100.00	

2 番目の報告書は、TPCT によって次の 2 点を示しています。

- 各給与が列合計に占める割合のパーセンテージ
- 切れ目ごとの小計

JOB	NAME	TPCT SALARY
-----	-----	-----
CLERK	JAMES	10.70
	KERMISCH	9.71
	NGAN	9.91
	SNEIDER	11.29
-----	-----	-----
*		41.61
-----	-----	-----
MGR	HANES	16.37
	SANDERS	14.54
-----	-----	-----
*		30.91
-----	-----	-----
SALES	PERNAL	14.40
	ROTHMAN	13.08
-----	-----	-----
*		27.47
=====		
100.00		

パーセンテージ取扱コード (PCT、CPCT、TPCT、および TCPCT) を使用すると、QMF はパーセンテージの合計を 100 として示します。しかし、ときには、個々のパーセンテージを加えていった結果が 100 より少し大きくなったり、少し小さくなったりすることもあります。これが起こるのは、個々のパーセンテージの計算で四捨五入が行なわれるためです。

### 図表:

これらの取扱コードに関する報告書での情報は、図表にも当てはまります。次の理由により、図表で使うのは、報告書で使うほど意味のないコードもあります。

- 累積パーセンテージや合計に、意味のある図表的表現を与えることは困難です。
- エラーで未定義のデータ値が生じたときは、ヌル値と見なされ、報告書では疑問符として示されます。
- 図表化される報告書に次の記号が入っていると、どれもヌル値と見なされます。
  - ハイフンは、報告書ではヌル値を表します。
  - アスタリスクは、長すぎて列幅に入らないデータ値を表します。
  - より大きい (>) 記号は、算術オーバーフローを表します。
  - 疑問符 (?) は、未定義値を表します。

## BREAK 取扱コード

BREAK 取扱コードは、報告書に 6 レベルにわたる制御の切れ目（またはグループ）を設けます。

**報告書:** BREAK1 を取扱コードにもつ列は、レベル 1 の切れ目の制御列です。この列の値が変わると、制御の切れ目が起こります。集約取扱コードをもつ列の小計が表示されて、レベル 1 の切れ目テキストが表示されます。

BREAK 取扱コードを使用するときは、次の点に注意してください。

- 列の値が変わるたびに報告書の制御の切れ目になるようにする場合は、照会で ORDER BY 文節を使用しなければなりません。そうすれば、報告書には、列の値が変わるたびに制御の切れ目が示されることになります。ORDER BY を指定しない場合は、報告書の行数と同じだけの制御の切れ目が示されることがあります。
- 照会への回答セットが大きいと、QMF が数度にわたってデータベースからデータを取り出すことがあります。データが毎回同じ順序で返されるようにするには、照会に必ず ORDER BY ステートメントを含めてください。同様に、定義列で BREAK を使用するときは、列が何度も評価されても、そのたびに同じ結果が得られるように注意してください。
- BREAK 取扱コードをもつ列が複数あってもかまいません。その場合、それらの列全部が制御の切れ目に関与します。たとえば、ある表に YEAR、MONTH、DAY という列があるとき、それぞれに BREAK1 取扱コードを指定すると、日付が変化するたびにレベル 1 の切れ目が起こります。
- 取扱コード BREAK2 は、レベル 2 の切れ目の列を制御します。この列は、レベル 1 の切れ目の制御列の右隣に表示されます (FORM.OPTIONS で列の自動再配列オプションが YES にセットされている場合)。切れ目番号は連続していません。たとえば、書式で BREAK2、BREAK3、BREAK5 を使用し、BREAK1 または BREAK4 は使用しない、といったことができます。

BREAK、GROUP、および集約取扱コードを使用すると、報告書上で列の並び順が変わることがあります。報告書の列の自動的な再配列を選択すると、制御列が報告書の左端に移動し、集約を使用する列が右端に移動します。詳細は、234 ページの『FORM.OPTIONS』の『報告書列の自動再配列』を参照してください。デフォルトの解釈では、列の並べ換えは行なわれません。

BREAKn X (*n* は 1 から 6) を使用して、報告書から制御列を除くことができます。

**図表:** BREAK1 取扱コードは、図表の変更に使用できます。取扱コードに BREAK をもつ列の値は、X 軸用に選択されます。他の列のうち数値列は Y 軸データとしてプロットされ、非数値列は無視されます。

BREAKn X (*n* は 1 から 6) を使用して、図表から制御列を除くことができます。また、これによって X 軸上に数値データを等間隔に置くことができます。

QMF に用意されている図表フォーマットは、連続データに対して離散データを扱えるように調整されています。

## CALCid 取扱コード

### CALCid 取扱コード

CALCid 取扱コードは、FORM.CALC にある計算式のうち、その ID が報告書におけるグループ合計、切れ目合計、または最終列合計の *id* に等しいものを実行します。結果は、FORM.CALC で指定された編集コードと、FORM.COLUMNS で指定された幅に従って編集されます。

取扱コードとして CALCid を使用すると、計算が最後のデータ行に適用されます。列値が計算のなかで使用される場合は、その最後のデータ行だけが評価されます。この点が他の取扱コードと異なります。

### GROUP 取扱コード

**報告書:** GROUP 取扱コードは、その列の 1 組の値ごとに合計データを 1 行だけ表示します。合計行に表示できるのは、グループの各メンバーにとって同じ値（たとえば、制御列の値）、または取り扱いが集約関数の 1 つである列の結果としての値だけです。

列中の値グループごとの合計行を報告書に入れたいときは、GROUP BY と ORDER BY という文節を含んでいる照会を使用してください。GROUP BY は照会の結果をグループ単位で累計し、ORDER BY はそのグループを配列します。これにより、列中に異なる値がいくつあるかにより、それと同数の合計行が報告書にもうけられることになります。照会に ORDER BY がないと、報告書の行数と同じだけの合計行がつくられかねません。

GROUP BY と ORDER BY を使用すると、照会のパフォーマンスも向上します。

GROUP 取扱コードを使用するときは、次の点に注意してください。

- データを選択する照会は、ORDER BY 文節を含んでいかなければなりません。ORDER BY 文節がないと、報告書で予想外の結果が作成されることがあります。
- GROUP 取扱コードをもつ列が複数あってもかまいません。これが当てはまる場合は、そのうちのどの列の値が変化しても、新しいグループが始まります。GROUP 列が 2 つあると、報告書に書き込まれるグループ値の行数がきわめて多くなることがあります。
- 取扱コードにブランクをもつ列があり、次のすべてが該当すると、報告書は作成されるものの、そのブランク取り扱いの列がすべて除かれます。
  - 報告書中に GROUP 取扱コードをもつ列が 1 つ以上ある。
  - ほかに集約取扱コードをもっている列がある。
  - 残りの列に、取扱コードがブランクになっている列がある。
- ある列の取扱コードが GROUP で、他のすべての列の取扱コードがブランクなら、GROUP 取扱コードを含む列は報告書から除かれます。

**図表:** 図表に対する GROUP の働きは、報告書のフォーマット設定の場合と似ています。

## プランクおよび OMIT 取扱コード

取扱コードが OMIT なら、その列と列値が表報告書や図表から除かれます。ただし、書式変数 ( $&n$  など、照会の SELECT ステートメントの中の列位置を示す) を使えば、その列の値を報告書に含めることができます。

取扱コードがプランクのままであれば、列データは、その列の編集コードに従って表示されます。詳細については、『編集コード』を参照してください。

| プランクおよび OMIT 取扱コードのみが、LONG、VARCHAR、LONG  
| VARGRAPHIC、DECFLOAT、および XML データ・タイプに有効です。

## 日付 / 時刻取扱コード

DATE 値、TIME 値、および TIMESTAMP 値には、算術関数を指定できません。

DATE 値、TIME 値、TIMESTAMP 値に使用できる取扱コードは、次のとおりです。

ACROSS  
GROUP  
BREAK $n$  ( $n=1,2,\dots,6$ )  
BREAK $nX$  ( $n=1,2,\dots,6$ )  
FIRST  
LAST  
COUNT  
MINIMUM  
MAXIMUM  
OMIT

DATE 値、TIME 値、TIMESTAMP 値に使用できない取扱コードは、次のとおりです。

AVERAGE  
STDEV  
PCT  
CPCT  
TPCT  
TCPCT  
SUM  
CSUM

## 編集コード

編集コードは、報告書の特定列のデータをどのようにフォーマット設定するかを QMF に指示する文字セットです。編集コードはデータベース内のデータを変更しません。単にデータの表示方法を制御するだけです。編集コードは、FORM.MAIN、FORM.COLUMNS、または FORM.CALC パネルで操作しているデータに指定します。

## 編集コード

このトピックでは、データ・タイプごとに有効な編集コードを説明しています。定義列に対する編集コードの働きについては、219ページの『編集コード、データ・タイプ、および長さ』を参照してください。

### 文字データの編集コード

次の編集コードを文字データに使用できます。

**C** 値を表示するときに変更しません。

**CW** 変更を加えずに値を表示しますが、値が列の1行に収まらない場合、このコードは、列幅に応じてテキストを折り返します。QMFは、列の末尾でデータを切り捨てず、列内の1行にできるだけのデータを置き、そこで折り返して、残りのデータを列内の次行に置きます。

列内折り返し列 (CW、CT、CD、XW、BW 編集コード) のデータは、常にデフォルトの調整値で位置合わせされます。(列内折り返し列のヘッダーについては、位置合わせを変更できます。) LEFT、CENTER、RIGHT の位置合わせを指定しても、これらの編集コードでは無視されます。(詳細については、216ページの『列調整』を参照してください。)

使用しているサイトが DBCS データを使用する場合、2 バイト文字と 1 バイト文字のデータが混在する列にも CW 編集コードを使用できます。そのような列の最小幅は 4 です。

次の例は、LOCATION 列の幅を短くして、編集コードを CW に変えた、変更前と変更後の報告書を示しています。

列の折り返しの前:

DEPTNAME	LOCATION
HEAD OFFICE	NEW YORK
PACIFIC	SAN FRANCISCO

列の折り返しの後:

DEPTNAME	LOCAT
HEAD OFFICE	NEW Y ORK
PACIFIC	SAN F RANCI SCO

**CT** 変更を加えずに値を表示しますが、値が列の1行に収まらない場合は、テキストに応じて列内折り返しをするよう QMF に指示します。QMF は、列の末尾でデータを切り捨てず、列内の1行にできるだけのデータを置き、ブランクがあるとそこで行を打ち切り、残りのデータを列内の次行に置きます。列に収まらないほど長いデータのストリングがあり、その中にブランクが含まれていない場合は、QMF はブランクが出てくるまで列幅単位でデータを折り返し、その後はテキスト単位で折り返しを続行します。

使用しているサイトが DBCS データを使用する場合、2 バイト文字と 1 バイト文字のデータが混在する列にも CT 編集コードを使用できます。この場合、QMF は SBCS ブランクで行を打ち切ります。そのような列の最小幅は 4 です。

次の例は、LOCATION 列の幅を短くして、編集コードを CT に変えた、変更前と変更後の報告書を示しています。

#### 列の折り返しの前:

DEPTNAME	LOCATION
HEAD OFFICE	NEW YORK
PACIFIC	SAN FRANCISCO

#### 列の折り返しの後:

DEPTNAME	LOCAT
HEAD OFFICE	NEW YORK
PACIFIC	SAN FRANC ISCO

#### CDx

テキスト内の区切り文字に従って列を折り返すよう QMF に指示します。QMF は、テキストに特殊区切り文字を検出するたびに、その列で改行する。この編集コードでは、x を自分で選択した特殊区切り文字に置き換えてください。区切り文字には、ブランクを含め任意の文字を使用できます。区切り文字自体は、出力には現われません。

使用しているサイトが DBCS データを使用する場合、2 バイト文字と 1 バイト文字のデータが混在する列にも CDx 編集コードを使用できます。そのような列の最小幅は 4 です。区切り文字は DBCS スtring の外になければなりません。

データのストリングが列に対して長過ぎ、かつ区切り文字が含まれていない場合、QMF は区切り文字を検出するまで列幅でデータを折り返し、その区切り文字によって折り返しを継続できます。データのストリングが複数の連続した区切り文字が含まれていると、QMF は 2 番目以降の各区切り文字について 1 ブランク行を示します。たとえば、データに 2 つの区切り文字がある場合は、最初の区切り文字が出てくると改行し、2 番目の区切り文字が出てくると 1 行だけスキップし、その後出力を折り返していきます。

次の例は、LOCATION 列の幅を短くして、編集コードを CD& に変えた、変更前と変更後の報告書を示しています。ここで、& は区切り文字です。

#### 列の折り返しの前:

DEPTNAME	LOCATION
HEAD OFFICE	NEW YORK
PACIFIC	SAN FRANCISCO

#### 列の折り返しの後:

DEPTNAME	LOCATION
HEAD OFFICE	NEW YORK
PACIFIC	SAN FRANCISCO

DATE、TIME、および TIMESTAMP 値で列折り返しができるようにするには、CW、CT、および CDx 編集コードを使用します。

## 編集コード

これらの編集コードを(任意のデータ・タイプに対して)使用する場合、列内折り返しは、表データを表示または印刷するときにしか起こりません。テキスト行で `&n` を参照しても、折り返されるデータの1行目が表示されるだけです。

## 文字または2進データの編集コード

次の編集コードを2進または文字データに使用できます。

- X** データを一連の16進文字として設定します。
- XW** データを一連の16進文字としてフォーマット設定します。列の境界で改行してデータを折り返します。
- B** データを2進形式(0と1の連続)にフォーマット設定します。
- BW** データを2進形式にフォーマット設定します。列の境界で改行してデータを折り返します。

XWまたはBW編集コードを使用する場合、列内折り返しは、表データを表示または印刷するときにしか起こりません。テキスト行で `&n` を参照しても、折り返されるデータの1行目が表示されるだけです。

2進データに対してCまたはCW編集コードを使用できるにもかかわらず、QMFはデータをフィルタリングしません。これらの編集コードを使用して2進データを表示しようとすると、QMFセッションが異常終了したり、現在表示されているデータを失ったりする可能性があります。

## グラフィック・データの編集コード

次の編集コードを漢字データに使用できます。

- G** 値を表示するときに変更しません。
- GW** 変更を加えずに値を表示しますが、値が列の1行に収まらない場合は、テキストを列幅に応じて折り返すようQMFに指示します。QMFは、列の末尾でデータを切り捨てず、列内の1行にできるだけのデータを置き、そこで折り返して、残りのデータを列内の次行に置きます。

## 数値データの編集コード

次の編集コードを数値データに使用できます。

- E<Z>** 数値を浮動小数で表示します。たとえば、このコードを使用すると、数値 -1234.56789 は -1.234E+03 と表示されます。E編集コードは、データ・タイプがFLOATと定義された列のデフォルトです。

QMFは、浮動小数点データの編集時には最大17桁の有効数字を示し、拡張浮動小数点データの編集時には最大34桁の有効数字を示します。これは、列幅がさらに収容可能であっても同様です。有効数字の桁数は、他のデータ・タイプではさらに少なくなります。

2番目の位置にあるZ編集コードは、ゼロ値を抑止します。

- D<Z><C>, I<Z>, J<Z>, K<Z>, L<Z>, および P<Z>**

これらの編集コードは、数値を10進表現で表示します。それぞれの編集記号における先行ゼロの有無、負数につける負符号、3桁単位の区切り記号、通貨記号、パーセント記号は、265ページの表28に示すとおりです。

各コードの後には、小数部分の桁数を示す値 (0 から 99) を付けることがあります。小数部分の桁数がそれより多い値は丸められ、少ない値にはゼロが付加されます。

デフォルト書式では、FLOAT 以外の数値データ・タイプをもつすべての列に L が使用されます。使用される小数部の桁数は、列定義のときと同じです。

適用される編集コードによって、値にわずかな違いが出ることがあります。たとえば、編集コード L6 を使用すると、値 0.068124999 は、0.068125 と表示されます。しかし、編集コード L5 のもとでは 0.06812 となります。この場合、もとの数値では小数部分の 6 桁目が 4 であるため、5 桁目の 2 は四捨五入で 3 とはなりません。

2 番目の位置にある Z 編集コードは、ゼロ値を抑止します。オプションで、2 番目または 3 番目の位置に C 編集コードを含めて、標準の通貨記号の代わりにユーザー定義の通貨記号を表示できます。通貨記号は、大域変数 DSQDC\_CURRENCY を使用して定義できます。Z と C を両方とも使用する場合には、Z の後に C を続ける必要があります。

表 28 では、編集コード D、I、J、K、L、および P の働きをまとめています。-1234567.885 という数値を例にとって、それがどのようにフォーマット設定されるかを示します。次のことを前提としています。

- 列幅は 15 です。
- QMF プロファイルの中の DECIMAL の値は PERIOD です。(3 桁単位の区切り記号と小数点に使用する文字が、この値で決まります。)

表 28. 10 進編集コードの属性と例

編集コード	先行ゼロ	負符号	3 桁ごとの 区切り記号		通貨記号 記号	パーセント	例
			Y	N			
D2	N	Y	Y	Y	N		-\$1,234,567.89
DC2	N	Y	Y	Y	N		-DM1,234,567.89
I2	Y	Y	N	N	N		-00001234567.89
J2	Y	N	N	N	N		000001234567.89
K2	N	Y	Y	N	N		-1,234,567.89
L2	N	Y	N	N	N		-1234567.89
P2	N	Y	Y	N	Y		-1,234,567.89%

## 日付データの編集コード

このトピックで説明する編集コードでは、x は、日付値の区切り記号を使う文字を表しています。この区切り文字には、ブランクを含め任意の特殊文字を使用できますが、英字や数字は使えません。

### デフォルト日付フォーマット

TD 編集コードは、日付データのデフォルトの編集コードです。TD 編集コードは、データベース・リクエスターに指定されたフォーマットで日付を表示します。

## 4 桁の年

表 29 に、日付を 4 桁の年でフォーマット設定するときに使用できる編集コードを示します。

表 29. 4 桁の年にフォーマット設定される編集コード

編集コード	結果	フォーマット
TDYx	最初が年	YYYYxMMxDD
TDMx	最初が月	MMxDDxYYYY
TDDx	最初が日	DDxMMxYYYY

## 略式の 2 桁の年

表 30 に、日付を 2 桁の年でフォーマット設定するときに使用できる編集コードを示します。

表 30. 2 桁の年にフォーマット設定される編集コード

編集コード	結果	フォーマット
TDYAx	最初が年	YYxMMxDD
TDMAx	最初が月	MMxDDxYY
TDDAx	最初が日	DDxMMXYY

## 代替日付フォーマット

**TDL** インストール先で定義フォーマットの情報は、QMF 管理者に問い合わせてください。

## 日付編集コードの例

表 31 の例は、1989 年 7 月 17 日という日付を各種の日付編集コードによってフォーマット設定したものです。

表 31. 日付編集コードの例

編集コード	フォーマット	コメント
TDD.	17.07.1989	ヨーロッパ・フォーマット
TDY-	1989-07-17	国際標準規格 (ISO) と日本工業規格 (JIS)
TDM/	07/17/1989	USA フォーマット
TDD-	17-07-1989	4 桁の年、最初が日、区切り文字はダッシュ (-)
TDDA/	17/07/89	2 桁の年、最初が日、区切り文字はスラッシュ (/)
TDDA.	17.07.89	2 桁の年、最初が日、区切り文字はピリオド (.)
TDDA-	17-07-89	2 桁の年、最初が日、区切り文字はダッシュ (-)
TDDA	17 07 89	2 桁の年、最初が日、区切り文字はブランク ( )
TDMA/	07/17/89	2 桁の年、最初が月、区切り文字はスラッシュ (/)
TDMA-	07-17-89	2 桁の年、最初が月、区切り文字はダッシュ (-)
TDYA/	89/07 /17	2 桁の年、最初が年、区切り文字はスラッシュ (/)

## 時刻データの編集コード

表 32 にある『x』は、時刻値の区切り記号に使う文字を表しています。この区切り文字には、ブランクを含め任意の特殊文字を使用できますが、英字や数字は使えません。

表 32. 時刻フォーマットの編集コード

編集コード	フォーマット	コメント
TTSx	HHxMMxSS	24 時間制、秒を含む
TTCx	HHxMMxSS	12 時間制、秒を含む
TTAx	HHxMM	略式(秒なし)
TTAN	HHMM	略式(秒なし、区切り文字なし)
TTUx	HHxMM AM HHxMM PM	USA フォーマット
TTL	ローカルで定義	フォーマットの情報は、QMF 管理者に問い合わせてください

### デフォルト時刻フォーマット

デフォルトの時刻編集コードの TT は、データベース・リクエスターに指定されたフォーマットで時刻を表示します。

### 時刻編集コードの例

表 33 の例は、時刻 1:25:10 PM が各種の時刻編集コードによってどのようにフォーマット設定されるかを示しています。

表 33. 時刻データの編集コードの例

編集コード	フォーマット	コメント
TTS.	13.25.10	ISO、ヨーロッパ・フォーマット
TTS:	13:25:10	JIS フォーマット
TTU:	01:25 PM	USA フォーマット
TTS,	13,25,10	時、分、秒(24 時間制)、区切り文字はコンマ(,)
TTC:	01:25:10	時、分、秒(12 時間制)、区切り文字はコロン(:)
TTA.	13.25	時、分(24 時間制)、区切り文字はピリオド(.)
TTA,	13,25	時、分(24 時間制)、区切り文字はコンマ(,)
TTAN	1325	時、分(24 時間制)、区切り文字なし

## タイム・スタンプ・データの編集コード

タイム・スタンプは日付 / 時刻を表す 7 部からなる値で、マイクロ秒を含みます。TIMESTAMP データ・タイプには、1 つの編集コード(TSI)しかありません。この編集コードは、タイム・スタンプ・データ・タイプをもつ列でしか使用できません。TSI 編集コードは、データを次のようにフォーマット設定します。

yyyy-mm-dd hh.mm.ss.nnnnnn

このフォーマットで、文字は次を意味します。

yyyy 年を表す 4 桁の値

## 編集コード

<i>mm</i>	月を表す 2 桁の値
<i>dd</i>	日を表す 2 桁の値
<i>hh</i>	時を表す 2 桁の値
<i>mm</i>	分を表す 2 桁の値
<i>ss</i>	秒を表す 2 桁の値
<i>NNNNNN</i>	マイクロ秒を表す 6 桁の値

TSI 編集コードでフォーマット設定された値は、たとえば次のように表示されます。

1991-12-29-23.25.15.123000

## QMF が列メタデータを表示するためのデータ・タイプ

列がヌルでなければ、M 編集コードを使用して、列の実際のデータではなく、メタデータ（列のデータ・タイプと長さ）を表示できます。QMF は、表示できないデータ・タイプ（BINARY、VARBINARY、XML、DECFLOAT、BLOB,CLOB、およびDBCLOB）に対して、自動的にこの編集コードを割り当てます。メタデータは、列幅が短くて全部を表示できないときは、表示できない部分は切り捨てられます。

FORM.MAIN または FORM.COLUMNS を使用して、データ・タイプに応じて、次のように M 編集コードを別のコードに変更できます。

表 34. デフォルトで編集コードが M であるデータ・タイプ

データ・タイプ	M 以外の有効な編集コード
XML	文字データに有効な編集コード
BINARY、 VARBINARY	2 進データに有効な編集コード
DECFLOAT	なし（10 進浮動小数点データは表示できません）。
LOB データ・タイプ (CLOB、BLOB、 DBCLOB)	BLOB データの 2 進編集コード。他の LOB データ・タイプの編集コードに変更すると、意味のない結果になります。

## ユーザー定義編集コード

BLOB、CLOB、DBCLOB、XML、および 10 進浮動小数点（DECFLOAT）を除くすべてのタイプのデータを特殊な目的にフォーマット設定するために、編集コード Uxxxx と Vxxxx があります。xxxx は文字を任意に組み合わせた 4 文字にできますが、内部にブランクやヌルを埋め込むことはできません。使用できるユーザー編集コードと、それがサポートするデータ・タイプについては、QMF 管理者に問い合わせてください。

## 総計機能および編集コードの考慮事項

QMF は、列の編集コードから得られる値ではなく、データベース表に保管されている実際の値に基づいて集約関数の結果を計算します。列の編集コードから得られる値を使用して総計結果を入手するには、新しい列を定義してから REXX 関数を使用する、などの方式を使用する必要があります。

たとえば、

1. 次の Q1 という名前の照会を作成して保管します。

```
SELECT 10.5 from Q.ORG
```

2. コマンド RUN Q1 (ROW 2 を発行します。報告書は次のように表示されます。

```
COL1
-----
10.5
10.5
```

3. コマンド SH F. COL を出します。

4. カーソルを COL1 の位置に移動し、「Insert」ファンクション・キーを押します。

5. 図 31 に示すように、COLUMN HEADING (列ヘッダー) の下に COLNEW と入力し、COL1 と COLNEW の両方の USAGE (取扱) の下に SUM と入力し、COLNEW の編集コードを L に変更します。

FORM.COLUMNS		MODIFIED			
NUM	COLUMN HEADING	Total Width of Report Columns: 20			
		USAGE	INDENT	WIDTH	EDIT
---	---	---	---	---	---
1	COL1	SUM	2	6	L1 1
2	COLNEW	SUM	2	10	L 1
*** END ***					

図 31. 列の編集コードからの値を使用した集約結果の取得

6. カーソルを COLNEW の位置に移動し、「指定」ファンクション・キーを押します。

7. 「定義」を選択してから、Enter キーを押します。

8. 次の REXX 式を入力して、Enter キーを押します。

```
format(&1,5,0)
```

9. F12 を押して、「指定」ウィンドウを閉じます。

10. 以下の報告書を表示するには、「Report」ファンクション・キーを押します。

```
COL1      COLNEW
-----
10.5      11
10.5      11
=====  =====
21.0      22
```

COLNEW には、行ごとに丸めた値が入っており、合計はこの丸めた値の合計になることに注意してください。

## 書式で使用される変数

QMF 書式では、大域変数（ユーザ一定義と QMF 提供の両方）と書式変数を使用できます。変数は、テキスト・ストリングや数値の代わりに使用できます。変数にさまざまな値を割り当てることで、書式そのものをえずくに、さまざまな形の報告書を作成することができます。

## 書式変数

書式内の大域変数は、複数の照会で同じ書式を共用できるようにします。たとえば、*Annual Report for 2005* というテキスト・ストリングを SET GLOBAL コマンドで *&ann* 変数に割り当てておき、書式の中でそれを使用するというようなことができます。（このコマンドの詳細は、123 ページの『SET GLOBAL』を参照してください。）SHOW GLOBAL コマンドを使用すれば、現在使用できる大域変数の一部または全部を見ることができます。

普通、QMF は置換変数の文字値から後続ブランクを取り除きます。数値からは先行ブランクを除去します。報告書の中で置換変数の先行または後続ブランクを残すときは、書式パネルの変数に \_B を付けます（たとえば、&3\_B）。この特殊構文は、書式パネルの置換変数でのみ有効です。照会やプロシージャーで使用される置換変数、あるいは &ROW、&DATE、&TIME、および &PAGE の各変数では使用できません。

QMF は、書式変数 と呼ばれる変数をもち、この変数によってシステム情報または報告書に関する情報を戻します。書式変数には次のものがあります。

- &ROW
- &COUNT
- &DATE
- &TIME
- &PAGE
- &CALC*id*
- &*n*
- &*an*

これらの変数は書式パネルの文脈の中で定義され、報告書に現れる位置に入力されます。書式変数については、該当する個々の書式パネルの項で説明します（適用可能な場合）。

表 35 は、さまざまな書式パネルで使用できる変数について示します。

表 35. 書式パネルで使用できる変数

	F.PAGE		F.BREAK		F.CALC	F.COLUMNS (列定義)	F.CONDITIONS	F.DETAIL		F.FINAL
	ヘッダー	後書き	ヘッダー	後書き				ヘッダー	ロック	
&ROW	x	x	x	x	x	x	x	x	x	x
&DATE	x	x	x	x	x	x	x	x	x	x
&TIME	x	x	x	x	x	x	x	x	x	x
&PAGE	x	x	x	x	x			x	x	x
&COUNT				x	x				x	x
&CALC <i>id</i>				x					x	x
& <i>n</i>	x	x	x	x	x	x	x	x	x	x
& <i>an</i>				x	x				x	x
大域変数	x	x	x	x	x	x	x	x	x	x

单一引用符または二重引用符は、書式で使用する変数に影響を与えません。

## 第 4 章 一般事項

このトピックには次の情報が含まれています。

- 『命名規則』
- 272 ページの『2 バイト文字を含む名前』
- 272 ページの『小数点に代えてコンマでの 10 進数のフォーマット設定』
- 273 ページの『QMF 一時記憶域』
- 274 ページの『報告書の完成と不完全データ指示』
- 275 ページの『長時間を要する照会に対する QMF の応答の変更』
- 275 ページの『QMF オブジェクトの編集時におけるデータとしてのヌルの使用の回避』
- 276 ページの『照会の書き方』
- 276 ページの『プロシージャー』
- 279 ページの『オブジェクトの印刷』
- 279 ページの『表エディター』
- 281 ページの『オンライン・ヘルプ』
- 281 ページの『リモート・データ・アクセス』
- 283 ページの『管理プログラムによる割り込み』

### 命名規則

データベースにオブジェクトを保管するときは、次の命名規則が適用されます。

- 照会、書式、プロシージャー、表、視点の名前は、他と同じであってはなりません。 (たとえば、照会と書式の名前が同じであってはなりません。)
- 名前を数字で始めることはできません。
- 名前を二重引用符で囲むときは、その名前を、二重引用符とブランクを除く任意の文字で始めるすることができます。
- QMF オブジェクト名には、次の特殊文字を 除く 任意の文字で始めるすることができます。

., ; : < > ( ) | + - \* / = & - ' "

英語以外の 1 バイト文字セットのなかには、否定記号 (¬) の代わりに曲折アクセント記号 (^) を表示したり、縦線 (l) の代わりに感嘆符 (!) を表示したりするものがあります。.

- 上記の特殊文字を名前に使うことは避けてください。これらの文字を SQL 名に使うときは、必ず名前全体を二重引用符で囲まなければなりません。SQL 名で特殊文字を使用する場合の規則については、IBM Publications Center ([www.ibm.com/shop/publications/order](http://www.ibm.com/shop/publications/order)) にある SQL 参照情報を参照してください。
- 完全修飾名 (*location.owner.name* の形式) は、QMF バージョン 8.1 以降 (新規機能モードにマイグレーション後) では、280 文字より長くはできません。*location* 修飾子は 16 バイトまで、*owner* 修飾子は 128 バイトまで、オブジェクトの *name* は 128 バイトまでです。たとえば、次の完全修飾名があります。

NEW\_YORK.Q.STAFF

これは、NEW\_YORK というロケーションに所属し、ユーザー Q が作成した、STAFF という名前の表を意味します。

- QMF 予約語を名前に使用してはなりません。そのような名前を QMF コマンドで指定した場合、データベース中のオブジェクトを参照しているとは見なされません。 QMF 予約語は次のとおりです。

CHART FORM QUERY DATA TABLE PROC REPORT FORM PROFILE

- SQL 予約語を名前に使用してはなりません。 予約語のリストは、IBM Publications Center ([www.ibm.com/shop/publications/order](http://www.ibm.com/shop/publications/order)) にある SQL 参照情報を参照してください。

---

## 2 バイト文字を含む名前

使用しているサイトが 2 バイト文字セット (DBCS) データをサポートしている場合、名前に 2 バイト文字だけを使用することも、1 バイト文字セット (SBCS) データと 2 バイト文字を混在させることもできます。2 バイト文字の使用には、次の規則が適用されます。

- 2 バイト文字と 1 バイト文字が混在する名前には、271 ページの『命名規則』に示す 1 バイト文字も使用できます。
- 書式内の列ヘッダーには、2 バイト文字と 1 バイト文字を混在させることができます。2 バイト文字だけからなるヘッダーには、最大 19 個の 2 バイト文字を使用できます。
- 2 バイト文字だけの名前には、QMF 互換モード・データベースでは 2 バイト文字を最大 8 個まで、QMF 新規機能モード・データベースでは 2 バイト文字を最大 63 個まで使用できます。名前はユーザー ID で修飾できます。ユーザー ID は、すべて 1 バイト文字で構成するか、あるいはすべて 2 バイト文字で構成するかのどちらかです。QMF 互換モード・データベースでは、ユーザー ID は、8 個までの 1 バイト文字か、3 個までの 2 バイト文字に制限され、QMF 新規機能モード・データベースでは、ユーザー ID は、128 個までの 1 バイト文字か、63 個までの 2 バイト文字に制限されます。
- データベースが表名について明示的に 2 バイト文字をサポートしている場合は、すべての名前に任意の 2 バイト文字を入れることができます。
- データベースが表名について明示的に DBCS データをサポートしていない場合でも、すべての名前に任意の 2 バイト文字を入れることができますが、内部的に二重引用符 (X'7F) として表される 2 バイト文字を入れることはできません。

DBCS データの使用と取り扱いについては、「DB2 QMF 使用の手引き」および「DB2 QMF アプリケーション開発の手引き」を参照してください。

---

## 小数点に代えてコンマでの 10 進数のフォーマット設定

小数点にコンマを使っている場合、コンマで終る数値は整数と解釈されます。たとえば、

RUN PROC (&1=3, は、RUN PROC (&1=3 と解釈されます。

分離記号としてコンマを使うには、コンマの後に 1 個のブランクを置いて小数点と区別しなければなりません。

## QMF 一時記憶域

QMF の中でオブジェクトを作成したり操作しているとき、それらのオブジェクトは、次の 1 時記憶域に入っています。

<b>QUERY</b>	すべてのタイプの照会を保持します。すべての照会タイプ（指示照会、SQL 照会、および QBE 照会）用に、1 つの 1 時記憶域があります。QUERY 1 時記憶域の内容を表示するには、SHOW QUERY を入力します。
<b>PROC</b>	QMF プロシージャーを保持します。2 つのタイプのプロシージャー（線形手順とロジックを持つプロシージャー）用に、1 つの 1 時記憶域があります。PROC 1 時記憶域の内容を表示するには、SHOW PROC を入力します。
<b>FORM</b>	報告書のフォーマット仕様を保持します。報告書のいろいろな部分のフォーマット仕様は、SHOW FORM <i>formname</i> を入力して表示できます。書式名のリストは、131 ページの『SHOW』を参照してください。
<b>DATA</b>	IMPORT、RUN、または DISPLAY コマンドからの結果のデータを保持します。DATA を FORM でフォーマット設定すると、報告書が作成されます。 DATA の内容を表示するには、SHOW REPORT を入力してください。DATA が直接表示されるのではなく（直接表示の手段はありません）、FORM 1 時記憶域の中の書式によってフォーマット設定して DATA 1 時記憶域の内容が表示されます。 対話式図表ユーティリティー（ICU）を使って DATA を図表形式で表示するには、SHOW CHART と入れてください。
<b>REPORT</b>	現在 FORM 1 時記憶域の中にある書式によってフォーマット設定して DATA オブジェクトの内容を保持します。 報告書の内容を表示するには、SHOW REPORT を入力します。
<b>CHART</b>	CHART オブジェクトを保持します。これは、GDDM 対話式図表ユーティリティーにより図形フォーマットで表示される報告書仕様から構成されます。
<b>PROFILE</b>	ユーザーの QMF プロファイルが入っています。PROFILE 1 時記憶域の内容を表示するには、SHOW PROFILE を入力します。

このどれかの一時記憶域の内容を保管するには、SAVE コマンドを使用します。これについては、118 ページの『SAVE』に説明しています。

操作しているオブジェクトを保管しないければ、そのオブジェクトは、QMF を終了すると削除されるか、あるいは、そのオブジェクトと同じタイプの新しいオブジェクトを同じ一時記憶域に入れてしまうコマンドによって上書きされてしまします。そのようなコマンドは、次のとおりです。

- IMPORT

## 一般事項

- RUN QUERY または RUN PROCEDURE
- DISPLAY *objectname*。ここで、*objectname* は、現在、一時記憶域にあるオブジェクトと同じタイプで、それとは別のデータベースに保管されているオブジェクトです。

たとえば、SQL 照会を操作していてまだ保管していないとします。そこで、コマンド DISPLAY QUERY MYQUERY を発行すると、MYQUERY は、現在 SQL 照会パネルにある、保管されていない SQL 照会に上書きします。

データベース中にある一部の表、たとえば、Q.STAFF などは、永続的です。所有していない表を消去するには、許可を持っている必要があります。

IMPORT、RUN、および DISPLAY コマンドの詳細は、1 ページの『第 1 章 QMF のコマンド』を参照してください。

---

## 報告書の完成と不完全データ指示

照会を実行したり、表または視点を表示したりすると、QMF は、とりあえず報告書を表示するのに必要な行だけをデータベースから取り出します。QMF で報告書を完成させるにはさらに行の取り出しが必要ですが、こうすることで報告書の迅速な表示ができます。

ユーザーが報告書を完成させないと、(たとえば、途中で DATA をリセットしたり、報告書の末尾までスクロールしたりすると)、QMF はユーザーがデータベースに関係する次の操作を要求した時に、報告書を完成させます。次のコマンドを発行すると、QMF はまず報告書を完成させてから、そのコマンドを実行します。

- CONNECT
- DISPLAY *tablename*
- DPRE
- DRAW *tablename*
- EDIT TABLE
- ERASE
- EXPORT (データベースから)
- IMPORT (データベースへ)
- LIST
- PRINT (データベースから)
- REFRESH (データベース・オブジェクト・リストの)
- RUN (データベース内のオブジェクト)
- SAVE (DATA、FORM、PROC、または PROFILE)

QMF が報告書を完成させようとしているときに QMF 一時記憶域がいっぱいになると、QMF は、275 ページの図 32 に示す「不完全データ・オブジェクト」プロンプト・パネルを表示します。

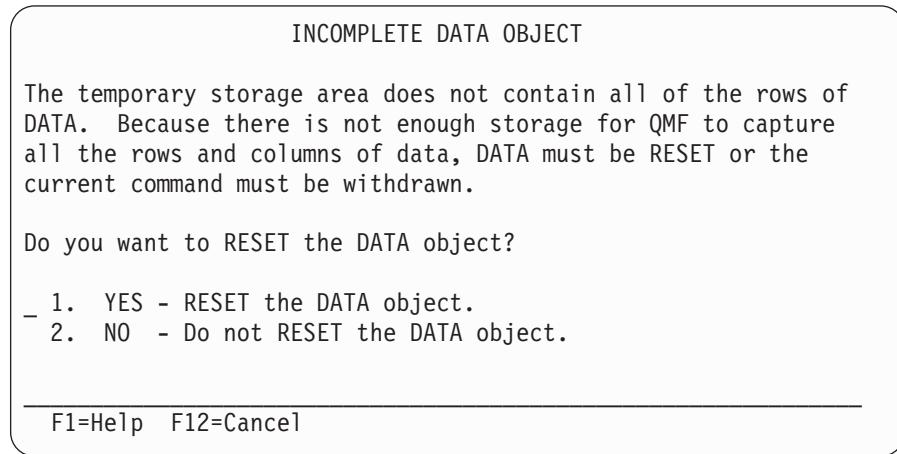


図 32. 「不完全データ・オブジェクト」プロンプト・パネル

このプロンプトには、次のどちらかの応答をします。

**YES** QMF 一時記憶域にあるすべてのデータを消去します。データはまったく使用できなくなります。DATA オブジェクトの内容にもう用がないときは、YES を選択してください。

**NO** コマンドを取り消し、DATA オブジェクトをいまのまま残します。

## 長時間を要する照会に対する QMF の応答の変更

QMF コマンドによっては、照会からの結果の行がすべて一時記憶域に入るまで実行できないものもあります。照会を実行中に新しいコマンドを出すと、デフォルトの QMF の応答はその照会を完了させ、その後で新しいコマンドを実行します。

DSQEC\_RESET\_RPT 大域変数を次のように設定して、このような場合の QMF の応答を変更することができます。

SET GLOBAL DSQEC\_RESET\_RPT=*n*

このコマンドで、*n* の値は次のとおりです。

- 0** リセット報告書プロンプト・パネルは表示されず、QMF は照会を実行します。
- 1** リセット報告書プロンプト・パネルが表示されます。このパネルは、照会を停止するか、あるいは継続するかを尋ねるプロンプトを出します。
- 2** リセット報告書プロンプト・パネルは表示されず、照会は停止します。

## QMF オブジェクトの編集時におけるデータとしてのヌルの使用の回避

QMF は、パネルの表示に GDDM を使用しており、NULL 値 (内部で 16 進表現の X'00' の値をもつ) は GDDM 画面表示に影響を与えます。したがって、「照会編集」パネルなどの QMF パネルではヌルの使用を避けてください。代わりに、16 進表示の定数または SQL 照会のデータベース HEX 関数などを使用してください。

## 一般事項

たとえば、16進値 03C1549F の入った FLD1 という列を持つ TEST という表の中の 1 バイトを NULL 値 (2進ゼロ) に変更するには、次の更新ステートメントを実行します。

```
UPDATE TEST SET FLD1=X'0300549F' WHERE FLD=X'03C1549F'
```

これで、このフィールドはデータベース HEX 関数を使用して表示できます。

```
SELECT HEX(FLD1) FROM TEST
```

---

## 照会の書き方

照会は、構造化照会言語 (SQL) で書くことができます。あるいは、指示照会および例示照会 (QBE) と呼ばれる照会を書くための支援メソッドを使用することもできます。

### SQL

SQL をよく知っている場合は、「SQL 照会」パネルを使ってデータベースに直接 SQL ステートメントと SQL 照会を発行できます。SQL の使用の詳細は、143 ページの『第 2 章 QMF 照会で使用される基本的な SQL ステートメントと関数』を参照してください。また、SQL ステートメントの使用方法と例が含まれた「DB2 QMF 使用の手引き」も参照してください。

### 指示照会 (Prompted Query)

指示照会では、指示に従って段階的に照会を作成できます。指示照会を開始するには、次のコマンドを発行します。

```
RESET QUERY (LANGUAGE=PROMPTED)
```

プロファイルの中で照会言語がすでに PROMPTED に設定されていれば、コマンドに LANGUAGE パラメーターを指定する必要はありません。(プロファイルの中のこのオプションの設定方法の詳細は、126 ページの『SET PROFILE』を参照してください。)

新しい指示照会作成にかかると、QMF が画面の右側に対話パネルを表示し、照会作成が完了するまで対話で導きます。対話パネルで作業するにしたがって、指示照会が画面の左側の確認域に作成されていきます。

### 例示照会 (QBE)

QBE は、SQL で照会を作成する 1 つの方法であり、図形を使用します。例示照会の使用方法の詳細は、「DB2 QMF 使用の手引き」を参照してください。

---

## プロシージャー

QMF を開始すると、システム初期化プロシージャーが、QMF セッションを構成するするために実行されます。一連の QMF コマンドをプロシージャーとして作成しておけば、1 つの RUN コマンドでそのプロシージャー全体を実行できます。長すぎてコマンド行から入れられないコマンドを使用するときは、この方法が便利です。プロシージャーを書いたシステム以外のシステムでそのプロシージャーを実行する必要が出てくる場合もあるので、プロシージャーの中に、可能であれば、システム固有のコマンドは使用しないでください。

プロシージャを実行すると、コマンド行からコマンドを入れた場合と同じように、QMF 一時記憶域の DATA、FORM、QUERY の内容は変わります。

プロシージャで使うコマンド、オプション、値には、省略形でなくフルネームを使用してください。他との混同なく使用できる最小の省略形が、将来の QMF リリースでは変わるかもしれません。

作成できるプロシージャには、ロジックを持つプロシージャと線形手順の 2 種類があります。プロシージャの最初のステートメントが REXX コメントであれば、QMF はそれをロジックを持つプロシージャ と見なし、それ以外の場合を、QMF は線形手順 と見なします。

ロジックを持つプロシージャで線形手順を実行することも、線形手順でロジックを持つプロシージャを実行することもできます。プロシージャに長さの制限はありません。

## ロジックを持つプロシージャ

### CICS ユーザーへの注

ロジックを持つプロシージャの機能は REXX に依存しますから、CICS では使用できません。

ロジックを持つプロシージャには、条件つきの論理や計算を実行し、ストリングを作成し、コマンドを QMF が稼働している環境に送り返す REXX 命令が含まれます。

ロジックを持つプロシージャには、独自の REXX 変数プールがあります。ロジックを持つプロシージャを使用すれば、QMF 大域変数を入手し、設定することができます。ロジックを持つプロシージャの中の QMF コマンドには、置換変数を含めることができます。

ロジックを持つプロシージャの中の QMF コマンドは、ユーザー・プロファイルでの設定値に関係なく必ず 大文字でなければなりません。

### 置換変数

置換変数の値は、各コマンドの実行時に決定されます。

置換変数は、プロシージャを実行している期間中存在する私用プロシージャ変数であることができます。あるいは、置換変数は大域変数を参照することもできます。

### 大域変数

大域変数の値は、プロシージャで即時に使用することができます。

GET GLOBAL コマンドを使用して大域変数の値をプロシージャ変数にコピーするか、SET GLOBAL コマンドを使用して、新しい大域変数値を設定します。

### リターン・コードとプロシージャの終了

コードの実行の成否は、リターン・コードで伝えられます。プロシージャは、リターン・コードを調べて、エラー状態に対処する適切なアクションを取らなければなりません。

## 一般事項

プロシージャーは、ゼロ以外のリターン・コードが発生したときはいつでも `signal on error` ステートメントを使用して `error` ラベルに移動できます。

**継続行** 前の行の末尾にコンマを追加して、複数行にわたることを示します。コマンド・キーワードおよび置換変数は、複数行にまたがることはできません。

### コメント

次のように、スラッシュとアスタリスクの組み合わせで囲みます。

*/\*comment\*/*

## 線形手順

線形手順には以下のものを含めることができます。

- 任意の QMF コマンド
- コメント行
- ブランク行
- 他のプロシージャーまたは照会を実行させる RUN コマンド
- 置換変数

線形手順の中で `SET GLOBAL` コマンドを使用して変数がセットされている場合は、その変数の値を同じプロシージャーの中のコマンドで使用できません。なぜなら、線形手順の中にある置換変数はすべて、プロシージャーが実行される前に解決されていなければならぬからです。プロシージャーの中で未解決の変数についてはプロンプトが出されます。ただし、変数を設定したプロシージャーから呼び出された照会またはプロシージャーはいずれも、その変数を使用することができます。

### 置換変数

QMF はプロシージャー全体を走査して置換変数を検索し、その値を解決してからプロシージャーを実行します。

### 大域変数

線形手順では、置換変数を使って大域変数値にアクセスします。

大域変数を設定した後、その大域変数をリセットする必要がある場合は、プロシージャーの終わりに `RESET GLOBAL` ステートメント・コードを入れておく必要があります。そうでなければ、置換値の前の設定が使用されます。

### リターン・コードとプロシージャーの終了

コードの実行の成否は、リターン・コードで伝えられます。なんらかのコマンドが失敗に終ると、プロシージャーは終了し、プロシージャー域の一番上に誤コマンドが表示されます。

**継続行** 続きの行の 1 桁目に正符号 (+) を置きます。コマンド・キーワード、置換変数、およびコメントは、複数行にまたがることはできません。

### コメント

次のように、前に 2 個のダッシュを置きます。

*--comment*

## オブジェクトの印刷

QMF オブジェクトの印刷規則は、印刷するオブジェクトのタイプと、使用するオペレーティング・システムによって異なります。

報告書、表、プロファイル、プロセッサー、SQL 照会、および QBE 照会を印刷する場合、次のガイドラインに従います。

- 非 GDDM 印刷には、プリンターのニックネームは不要です。

- GDDM を使用せずに印刷するには、

```
PRINTER=' '
```

- PRINT コマンドかユーザー・プロファイルでニックネームが指定されると、GDDM が制御を獲得します。

- ニックネームが示されないと（つまり、PRINTER=' ' のように、プリンターナンバーにブランクを指定すると）、出力は DSQPRINT に書き出されます。ニックネームが示されると、出力は GDDM に書き出されます。

図表を印刷する場合、次のガイドラインに従います。

- GDDM プリンターの正しいニックネームが必要です。

- プリンターナンバーが示されないと、プロファイル中のデフォルトのプリンターナンバーが使用されます。

- 装置トークンは有効な印刷装置でなければなりません。

- PRINT コマンドが出されると、必ず GDDM 対話式図表ユーティリティーが制御を獲得します。

指示照会と書式を印刷する場合、次のガイドラインに従います。

- GDDM プリンターの正しいニックネームが必要です。

- PRINT コマンドが出されると、必ず GDDM が制御を獲得します。

- 出力は、プリンターのニックネームに関連付けられている DD 名に書き出されます。

## 表エディター

表エディターは、表に行を追加したり表中の行を変更したりするための簡便な方法です。照会を書くまでもなく、更新権限をもつ列を変更できます。

できることは、表への行の追加、表からの行の削除、表内にある行の検索と変更です。

表エディターにアクセスする場合、既存の行を変更するのか、表に行を追加するのかによって、次のどちらかのコマンドを入力します。

```
EDIT tablename (MODE=CHANGE)
EDIT tablename (MODE=ADD)
```

ファンクション・キーを使用して表エディターのコマンドを発行します。ADD モードか CHANGE モードかで、表示されるファンクション・キー・セットが異なります。さらに、これらのモードでは、VARCHAR、VARGRAPHIC、または LONG VARGRAPHIC のタイプをもつ縦欄データを編集する場合、表エディターは自動的に後続ブランクを取り除きます。

## 一般事項

検索を行う場合、検索文字列の長さは列の長さに等しくなければなりません。等しくない場合には、そのデータベースで一致する項目は見つかりません。データの長さが列の長さよりも短い場合、一致が見つかるようにその検索文字列にワイルドカードを埋め込む必要があります。1 文字を表すワイルドカードとして下線 (\_)、または複数の文字を表すワイルドカードとしてパーセント記号 (%) を使用することができます。たとえば、FLD1 が 5 文字のフィールドとして定義されているとします。その値が AB\_D で、長さは 4 文字で、下線のワイルドカード文字 (\_) を含んでいます。検索を行うときは、列幅の 5 桁の文字位置すべてを表す値、たとえば、AB\_D\_、AB\_D%、AB\_%、または AB% を入力します。4 文字の実際の値 AB\_D を入力すると、QMF は次の SELECT ステートメントを発行します。

```
SELECT FLD1 FROM tablename WHERE FLD1 LIKE 'AB_D'
```

この場合、FLD1 は 5 文字のフィールドなので、該当のデータベースで一致する項目は見つからないはずです。たとえば、AB\_D\_ を使用すると、QMF は次のステートメントを生成します。

```
SELECT FLD1 FROM tablename WHERE FLD1 LIKE 'AB_D_'
```

AB% を使用した場合には、QMF は次のものを生成します。

```
SELECT FLD1 FROM tablename WHERE FLD1 LIKE 'AB%'
```

最後の 2 つの場合には、データベースはどちらも正しい行を検出します。これは、ワイルドカードがデータベースの FLD1 に対応する 5 桁の文字位置すべてを操作の対象にするためです。

表エディターには、どちらのモードであるかによって、異なるセットのファンクション・キーが現れます。たとえば、CHANGE モードで、変更したい行をさがすには、「SEARCH」というラベルのついたファンクション・キーを押します。SEARCH モードでは別のセットのファンクション・キーを表示します。

表 36 に、示されたモードの種々のパネルで表示されるファンクション・キーをリストしています。

表 36. 表エディターのモード別のファンクション・キー

CHANGE モード	ADD モード	SEARCH モード
BACKWARD	ADD	BACKWARD
CANCEL	BACKWARD	CANCEL
CHANGE	CANCEL	CLEAR
DELETE	CLEAR	END
END	END	FORWARD
FORWARD	FORWARD	HELP
HELP	HELP	PREVIOUS
NEXT	PREVIOUS	SEARCH
REFRESH	SHOW FIELD	SHOW CHANGE
SHOW FIELD		SHOW FIELD
SHOW SEARCH		

「入力域」パネルで、Enter キーを押すと情報が保管されてパネルが閉じられます。Cancel キーを押すと、情報は保管されずにパネルが閉じられます。

表エディターの使用方法の詳細は、「DB2 QMF 使用の手引き」を参照してください。表エディターを使用して、BINARY、VARBINARY、XML、または DECFLOAT と定義された列を含む表を変更することはできません。

## オンライン・ヘルプ

QMF には、次のタイプのオンライン・ヘルプが用意されています。

### トピック・ヘルプ

QMF パネルと概念の説明

### メッセージ・ヘルプ

ユーザー・エラーのために出されたメッセージの説明

### フィールド別ヘルプ

QMF 書式パネル上の入力フィールドについての情報

## トピック・ヘルプ

QMF パネルが表示されていて、そこになんのエラー・メッセージも出でていないときは、いつでもヘルプ・ファンクション・キーを押して情報を得ることができます。たとえば、QMF ホーム・パネルが表示されているときにヘルプ・ファンクション・キーを押すと、一般事項に属する題目のほか、コマンドや書式など、QMF のあらゆる部分に関する具体的情報を選ぶことができます。

## メッセージ・ヘルプ

QMF がエラーを検出すると、コマンド行のすぐ上にメッセージが表示されます。たとえば、RUN コマンドでタイプ・ミスをすると、次のようなメッセージが現れます。

```
RNU is not a command.  
COMMAND ==> RNU ROUTINE123
```

コマンド行でコマンドを訂正し、ENTER キーを押します。

メッセージだけではエラーの内容が明らかにならないときは、Help ファンクション・キーを押すか、HELP コマンドを入れると、もっと情報が得られます。それよりさらに詳しい情報が必要な場合は More Help ファンクション・キーを押してください。元のパネルに戻るには、Cancel ファンクション・キーを押します。

## フィールド別ヘルプ

フィールド別ヘルプでは、あらゆる書式パネル上の入力フィールドについてのオンライン・ヘルプ情報に、直接、アクセスできます。フィールド別ヘルプでは、あらゆる書式パネル上のフィールドについてのオンライン・ヘルプを得るには、目的の入力域にカーソルを合わせて Help ファンクション・キーを押してください。

ヘルプ機能の詳細については、62 ページの『HELP』を参照してください。

## リモート・データ・アクセス

リモートのデータにアクセスする方法には、分散作業単位 を使用する方法とリモート作業単位 を使用する方法があります。

## 分散作業単位 (DB2 (z/OS 版) のみ)

分散作業単位では、リモート・ロケーションにあるデータにアクセスし (オブジェクトのロケーション、所有者、名前を指定する 3 部構成の名前を使用して)、そのデータを自分の現行ロケーションで使用できます。

現行ロケーションが DB2 (z/OS 版) データベースである場合、ローカルの DB2 (z/OS 版) データベースに定義されている通信ネットワークの一部であるリモートの DB2 (z/OS 版) データベースで管理される表と視点を読み取り、それらを更新することができます。リモートの照会、プロシージャー、または書式にはアクセスできません。

照会では、リモート表またはリモート視点の指定に 3 部分名 または別名 を使用できます。3 部分名には、表の所在地の名前、表の所有者の名前、そして表そのものの名前が含まれます。次のように各部分をピリオドで区切れます。

NEW\_YORK.JBP.STAMPS

別名は、ローカルに定義された名前で、z/OS データベースの同一またはリモートの DB2 (z/OS 版) にある表や視点の参照に用いられます。ユーザー自身が 1 次許可 ID と 現 DB2 許可 ID のもとにどのような別名を所有しているかは、LIST コマンドで見られます。照会または QMF コマンドで別名を使用すると、その別名で参照される表または視点の使用許可の有無が検査されます。

リモート表またはリモート視点には、表 37 に示すコマンドでアクセスできます。

表 37. 3 部構成の名前で分散作業単位に使用できるコマンド

コマンド	制約事項
DISPLAY	TABLE オブジェクト・タイプを使用しなければなりません。
DRAW	TABLE オブジェクト・タイプを使用しなければなりません。
EDIT	なし
EXPORT	TABLE オブジェクト・タイプを使用しなければなりません。
IMPORT	TABLE オブジェクト・タイプを使用しなければなりません。
PRINT	なし
SAVE	DATA オブジェクト・タイプを使用しなければなりません。

リモート表は、SAVE コマンドまたは IMPORT コマンドで置き換えられます。

## リモート作業単位

リモート作業単位では、リモートに接続し、そのリモートでデータにアクセスして使用します。リモート作業単位での接続では、さらにもう 1 つ別のロケーションのデータにアクセスして、それを現在の接続先で使用することもできます。

QMF によって、分散ネットワーク内の DB2 (Linux、UNIX、および Windows 版) または DB2 Server (VSE または VM 版) データベースに接続できます。リモー

ト・データ・アクセスは、DB2 Server (VM 版) および DB2 (z/OS 版) によって完全にサポートされています。VSE 環境では、DRDA® リモート作業単位サーバー機能が VSE から提供されます。

リモートに接続すると、そこがそのユーザーの 現行ロケーション になります。  
 「同種間」(DB2 (z/OS 版) と DB2 (z/OS 版))、および「異種間」(DB2 (z/OS 版) と DB2 Server (VSE または VM 版)) ロケーションで、接続できます。QMF 初期化時に (START コマンドの DSQSDBNM プログラム・パラメーターを使用して) 接続を確立できるほか、QMF セッション内からも (QMF CONNECT コマンドを使用しても) 確立できます。

リモートへの接続後、そのユーザーが出すすべての (CONNECT 以外の) SQL ステートメントはそのリモートのデータベースに送られ、そこで処理されます。したがって、リモートにあるデータと QMF オブジェクトを、ユーザー所在地のデータやオブジェクトとほぼ変わりなくアクセスできます。

「*TSO/CICS 用 DB2 QMF インストールおよび管理の手引き*」には、リモート作業単位の準備の詳細が説明されています。リモートの作業単位の詳細については、「*DB2 QMF 使用の手引き*」を参照してください。

## 管理プログラムによる割り込み

サイトによっては、照会やプロセッサーが使用できるデータベース・リソースに制限を設けていることがあります。照会やプロセッサーが、サイトで設定された、時間制限を超過するか、データベースからの行の取り出し制限を超えると、処理に割り込みが行われます。そしてパネルが表示され、その照会またはプロセッサーを続行するか、取り消すかを尋ねてきます。TSO では、経過した CPU 時間が秒単位で表示されます。

照会やプロセッサーは、指示の有無にかかわらず取り消すことも、続行することもできます。しかし、続行を決めて、QMF 管理プログラムがその照会やプロセッサーを取り消すかもしれません。

TSO/CICS 用の QMF からは、管理プログラムによる割り込みの画面が表示されます。使用しているサイトが別の管理プログラム機能を使用しているときは、選択肢が異なるかもしれません。使用しているサイトで設定されている制限については、QMF 管理者に問い合わせてください。QMF High Performance Option の中で使用できる管理プログラム機能の詳細は、「*DB2 QMF High Performance Option User's Guide for TSO and CICS*」を参照してください。



## 付録 A. QMF のサンプル表

QMF は、プロダクト機能の理解を助け、テストしてみるために使用できるサンプルの表を提供しています。これらの表には、架空の電子部品製造メーカー J & H Supply Company の、応募者、面接、部品、製品、従業員、および供給業者に関するデータが収められています。次のサンプル表があります。

- ・『Q.APPLICANT』
- ・286 ページの『Q.INTERVIEW』
- ・286 ページの『Q.ORG』
- ・287 ページの『Q.PARTS』
- ・287 ページの『Q.PRODUCTS』
- ・288 ページの『Q.PROJECT』
- ・289 ページの『Q.SALES』
- ・290 ページの『Q.STAFF』
- ・291 ページの『Q.SUPPLIER』

### Q.APPLICANT

この表は、会社への就職を応募した人々に関する情報を提供します。各行は、応募者を示します。各列は次のとおりです。

**TEMPID** 応募者の一時識別子

**NAME** 応募者の姓

**ADDRESS** 応募者が居住する州および都市

**EDLEVEL** 応募者の学歴

**COMMENTS** 面接者によるコメント

Q.APPLICANT 表を図 33に示します。

TEMPID	NAME	ADDRESS	EDLEVEL	COMMENTS
400	FROMMHERZ	SAN JOSE, CA	12	NO SALES EXPERIENCE
410	JACOBS	POUGHKEEPSIE, NY	16	GOOD CANDIDATE FOR WASHINGTON
420	MONTEZ	DALLAS, TX	13	OFFER SALES POSITION
430	RICHOWSKI	TUCSON, AZ	14	CAN'T START WORK UNTIL 12/96
440	REID	ENDICOTT, NY	14	1 YEAR SALES EXPERIENCE
450	JEFFREYS	PHILADELPHIA, PA	12	GOOD CLERICAL BACKGROUND
460	STANLEY	CHICAGO, IL	11	WANTS PARTIME JOB
470	CASALS	PALO ALTO, CA	14	EXPERIENCED SALESMAN
480	LEEDS	EAST FISHKILL, NY	12	NEEDS INTERVIEW WITH BROWN
490	GASPARD	PARIS, TX	16	WORKED HERE FROM 1/94 TO 6/94

図 33. Q.APPLICANT 表

**Q.INTERVIEW**

この表は、日付 / 時刻データをサポートするサイト用です。ISO 形式で日付と時刻を示します。報告書の DATE、TIME、TIMESTAMP の各データの形式は、使用しているサイトでデフォルトとして選択された形式によって異なります。この形式は、DATE、TIME、TIMESTAMP の各編集コードで変更可能です。Q.INTERVIEW 表の列は次のとおりです。

<b>TEMPID</b>	応募者の一時識別子
<b>INTDATE</b>	面接の日付
<b>STARTTIME</b>	面接開始時刻
<b>ENDTIME</b>	面接終了時刻
<b>MANAGER</b>	応募者に面接した管理者の従業員番号
<b>DISP</b>	応募者採用の可否
<b>LASTNAME</b>	応募者の姓
<b>FIRSTNAME</b>	応募者の名

Q.INTERVIEW 表を図 34 に示します。

TEMPID	INTDATE	STARTTIME	ENDTIME	MANAGER	DISP	LASTNAME	FIRSTNAME
400	1996-02-05	13.00.00	15.12.00	270	NOHIRE	FROMMHERZ	RICHARD
410	1996-02-11	15.00.00	16.18.00	10	HIRE	JACOBS	SUSAN
420	1996-04-07	09.00.00	09.58.00	140	HIRE	MONTEZ	rita
430	1996-04-24	10.30.00	11.30.00	290	NOHIRE	RICHOWSKI	JOHN
440	1996-03-13	10.15.00	11.23.00	160	HIRE	REID	CATHY
450	1996-09-19	09.45.00	11.00.00	50	HIRE	JEFFREYS	PAUL
460	1996-10-06	14.45.00	16.22.00	100	HIRE	STANLEY	JOHN
470	1996-02-05	16.30.00	18.00.00	270	HIRE	CASALS	DAVID
480	1996-03-13	13.30.00	14.45.00	160	NOHIRE	LEEDS	DIANE
490	1996-09-30	15.00.00	15.44.00	140	NOHIRE	GASPARD	PIERRE

図 34. Q.INTERVIEW 表

**Q.ORG**

この表は、会社組織に関する情報を提供します。各行は部門を示します。各列は次のとおりです。

<b>DEPTNUMB</b>	部門番号 (固有でなければならない)
<b>DEPTNAME</b>	部門の名称
<b>MANAGER</b>	部門の管理者の従業員番号
<b>DIVISION</b>	部門が属する地域
<b>LOCATION</b>	部門が所在する都市名

Q.ORG 表を 287 ページの図 35 に示します。

DEPTNUMB	DEPTNAME	MANAGER	DIVISION	LOCATION
10	HEAD OFFICE	160	CORPORATE	NEW YORK
15	NEW ENGLAND	50	EASTERN	BOSTON
20	MID ATLANTIC	10	EASTERN	WASHINGTON
38	SOUTH ATLANTIC	30	EASTERN	ATLANTA
42	GREAT LAKES	100	MIDWEST	CHICAGO
51	PLAINS	140	MIDWEST	DALLAS
66	PACIFIC	270	WESTERN	SAN FRANCISCO
84	MOUNTAIN	290	WESTERN	DENVER

図 35. *Q.ORG* 表

## Q.PARTS

この表は部品に関する情報を提供します。各列は次のとおりです。

<b>SUPPNO</b>	供給業者の番号
<b>PARTNAME</b>	部品名
<b>PRODUCT</b>	部品が必要な製品
<b>PRODNO</b>	製品番号
<b>PROJNO</b>	プロジェクト番号

Q.PARTS 表を図 36に示します。

SUPPNO	PARTNAME	PRODUCT	PRODNO	PROJNO
1100P	PLASTIC	RELAY	30	1501
1100P	STEEL	WRENCHSET	509	1520
1200S	WIRE	GENERATOR	10	1401
1200S	BEARINGS	MOTOR	50	1402
1300S	COPPER	RELAY	30	1501
1300S	BLADES	SAW	205	1510
1400P	MAGNETS	GENERATOR	10	1409
1400P	VALVES	MOTOR	50	1407
1400P	OIL	GEAR	160	1405

図 36. *Q.PARTS* 表

## Q.PRODUCTS

この表は、いくつかの製品とその価格に関する情報を提供します。各列は次のとおりです。

<b>PRODNUM</b>	製品番号
<b>PRODNAME</b>	製品の名称
<b>PRODGRP</b>	製品の一般タイプ
<b>PRODPRI</b>	製品の価格

## サンプル表

Q.PRODUCTS 表を図 37に示します。

PRODNUM	PRODNAME	PRODGRP	PRODPRIICE
10	GENERATOR	ELECTRICAL	45.75
505	SCREWDRIVER	TOOL	3.70
101	SHAFT	MECHANICAL	8.65
20	SWITCH	ELECTRICAL	2.60
30	RELAY	ELECTRICAL	7.55
40	SOCKET	ELECTRICAL	1.40
50	MOTOR	ELECTRICAL	35.80
150	CAM	MECHANICAL	1.15
160	GEAR	MECHANICAL	9.65
190	BUSHING	MECHANICAL	5.90
205	SAW	TOOL	18.90
330	HAMMER	TOOL	9.35
450	CHISEL	TOOL	7.75
509	WRENCHSET	TOOL	25.90

図 37. Q.PRODUCTS 表

## Q.PROJECT

この表は、プロジェクトのスケジュールに関する情報を提供します。各列は次のとおりです。

<b>PROJNO</b>	プロジェクト番号 (固有でなければならない)
<b>PRODNUM</b>	製品番号
<b>DEPT</b>	プロジェクトを担当する部門の番号
<b>STARTD</b>	プロジェクトの開始日付
<b>ENDD</b>	プロジェクトの終了日付
<b>TIMESTAMP</b>	報告書の年、月、日、時刻

Q.PROJECT 表には、日付/時刻データが含まれており、日時が ISO 形式で表示されます。この形式は任意選択です。表の形態は、QMF 管理者による選択によって異なります。Q.PROJECT 表を 289 ページの図 38に示します。

PROJNO	PRODNUM	DEPT	STARTD	ENDD	TIMESTAMP
1401	10	20	1996-01-01	1998-03-31	1994-12-18-10.14.44.000001
1402	50	66	1996-01-30	1997-06-30	1994-12-18-10.15.01.999998
1403	150	51	1996-02-02	1999-05-29	1994-12-18-10.22.23.000001
1404	190	38	1997-01-04	1999-06-30	1994-12-18-10.25.43.999999
1405	160	15	1997-04-29	1999-10-30	1995-12-31-14.23.00.999999
1406	20	20	1997-07-11	1998-12-31	1996-01-05-13.31.18.009999
1407	50	42	1997-12-12	2000-06-15	1996-01-05-13.42.27.000000
1408	30	42	1999-03-13	2000-09-30	1996-01-05-13.44.16.999999
1409	10	66	1998-06-15	1999-12-31	1996-03-13-09.12.57.149572
1410	190	10	1998-09-29	2000-03-31	1996-03-13-12.18.23.402917
1501	30	51	1999-01-04	1999-12-31	1996-03-13-12.22.14.201966
1502	150	38	1999-03-01	2000-07-17	1996-03-13-13.17.48.948276

図 38. *Q.PROJECT* 表**Q.SALES**

この表は、営業担当員が獲得した注文データを提供します。この表の列は次のとおりです。

<b>ORDERNO</b>	営業担当員が獲得した部品注文の固有の番号
<b>SALESREPNO</b>	注文を獲得した営業担当員の固有の従業員シリアル番号
<b>PRODNO</b>	販売製品の固有のプロダクト番号
<b>QUANTITY</b>	CUSTNO 列の顧客により注文された製品の数
<b>CUSTNO</b>	各顧客を示す固有の数値 ID

Q.SALES 表を 290 ページの図 39 に示します。

## サンプル表

ORDERNO	SALESREPNO	PRODNO	QUANTITY	CUSTNO
3456	20	10	50	1200
6667	20	160	120	4400
1991	40	150	600	4500
7777	60	30	150	8500
1020	60	30	150	8500
3333	70	50	240	9600
1115	70	101	120	8300
3580	20	190	360	4900
2345	90	450	360	2500
5770	70	205	100	8300
6432	40	150	120	8900
4432	90	505	150	2550
3455	150	190	360	8800
4477	220	330	480	5600
6540	150	150	200	8850
6688	280	150	300	6600
4080	300	101	500	5900
5456	300	20	60	6300
3360	310	101	120	3600
4596	310	160	100	2000
4321	340	330	200	3000
4567	40	450	100	4100
7010	20	505	150	3500
1550	90	160	200	4000
2888	90	50	240	5000
5432	220	20	100	6000
6677	40	10	150	9111
5521	60	50	150	9666
4010	150	205	225	4297
3968	220	509	200	7329
5832	280	509	300	7299
4491	300	50	100	5581
3962	340	10	240	3681

図 39. Q.SALES 表

## Q.STAFF

この表は、J&H Supply Company の従業員に関するデータを提供します。各列は次のとおりです。

<b>ID</b>	従業員シリアル番号 (固有でなければならない)
<b>NAME</b>	従業員名
<b>DEPT</b>	従業員の部門番号
<b>JOB</b>	従業員の職務
<b>YEARS</b>	従業員の会社での勤務年数
<b>SALARY</b>	従業員の年収 (単位 : ドル、セント)
<b>COMM</b>	従業員の歩合 (単位 : ドル、セント)

Q.STAFF 表を 291 ページの図 40 に示します。

ID	NAME	DEPT	JOB	YEARS	SALARY	COMM
10	SANDERS	20	MGR	7	18357.50	-
20	PERNAL	20	SALES	8	18171.25	612.45
30	MARENghi	38	MGR	5	17506.75	-
40	O'BRIEN	38	SALES	6	18006.00	846.55
50	HANES	15	MGR	10	20659.80	-
60	QUIGLEY	38	SALES	-	16808.30	650.25
70	ROTHMAN	15	SALES	7	16502.83	1152.00
80	JAMES	20	CLERK	-	13504.60	128.20
90	KOONITZ	42	SALES	6	18001.75	1386.70
100	PLOTZ	42	MGR	7	18352.80	-
110	NGAN	15	CLERK	5	12508.20	206.60
120	NAUGHTON	38	CLERK	-	12954.75	180.00
130	YAMAGUCHI	42	CLERK	6	10505.90	75.60
140	FRAYE	51	MGR	6	21150.00	-
150	WILLIAMS	51	SALES	6	19456.50	637.65
160	MOLINARE	10	MGR	7	22959.20	-
170	KERMISCH	15	CLERK	4	12258.50	110.10
180	ABRAHAMS	38	CLERK	3	12009.75	236.50
190	SNEIDER	20	CLERK	8	14252.75	126.50
200	SCOUTTEN	42	CLERK	-	11508.60	84.20
210	LU	10	MGR	10	20010.00	-
220	SMITH	51	SALES	7	17654.50	992.80
230	LUNDQUIST	51	CLERK	3	13369.80	189.65
240	DANIELS	10	MGR	5	19260.25	-
250	WHEELER	51	CLERK	6	14460.00	513.30
260	JONES	10	MGR	12	21234.00	-
270	LEA	66	MGR	9	18555.50	-
280	WILSON	66	SALES	9	18674.50	811.50
290	QUILL	84	MGR	10	19818.00	-
300	DAVIS	84	SALES	5	15454.50	806.10
310	GRAHAM	66	SALES	13	21000.00	200.30
320	GONZALES	66	SALES	4	16858.20	844.00
330	BURKE	66	CLERK	1	10988.00	55.50
340	EDWARDS	84	SALES	7	17844.00	1285.00
350	GAFNEY	84	CLERK	5	13030.50	188.00

図 40. Q.STAFF 表

## Q.SUPPLIER

この表は、J&H Supply Company の供給業者に関するデータを提供します。各列は次のとおりです。

<b>ACCTNO</b>	会社の顧客番号
<b>COMPANY</b>	会社名
<b>STREET</b>	会社の番地
<b>CITY</b>	会社が所在する都市
<b>STATE</b>	会社が所在する州
<b>ZIP</b>	会社の郵便番号
<b>NOTES</b>	供給業者に関するその他の情報

Q.SUPPLIER 表を 292 ページの図 41 に示します。

## サンプル表

ACCTNO	COMPANY	STREET	CITY	STATE	ZIP	NOTES
1100P	WESTCO, INC.	1900 115TH ST.	EMERYVILLE	CA	16600	THIS COMPANY HAS A STRONG HISTORY OF ON-TIME DELIVERY. WESTCO IS GROWING QUICKLY.
1200S	MAJOR ELECTRICS	4250 BENSON ST.	DALLAS	TX	87050	MAJOR ELECTRICS DECLARED BANKRUPTCY IN 1987, BUT HAS RECOVERED. FORESEE NO FURTHER PROBLEMS.
1300S	FRANKLIN, INC.	40025 EASTLAND	DOVER	DE	99000	DUE TO ITS LOCATION ON EASTERN SEABOARD, FRANKLIN HAS EXCELLENT TRANSPORTATION FACILITIES.
1400P	MOTORWORKS , INC.	19503 BESWICK	JOLIET	IL	12000	PROXIMITY TO CHICAGO ENSURES GOOD TRANSPORTATION, BOTH BY RAIL AND TRUCK. A RELIABLE SUPPLIER.

図41. Q.SUPPLIER 表

NOTES 列の情報をすべて表示するために、FORM.MAIN または FORM.COLUMNS の列幅を調整しなければならないかもしれません。

## 付録 B. QMF 大域変数表

QMF は、ユーザーのアプリケーションで使用するための変数を多数用意しています。このトピックでは次の項目について説明します。

- 『大域変数の命名規則』
- 294 ページの『プロファイル関連状態情報の大域変数』
- 295 ページの『プロファイル関連でない状態情報の大域変数』
- 299 ページの『CICS 関連の大域変数』
- 299 ページの『最後に発行されたコマンドで生成されたメッセージに関連する大域変数』
- 300 ページの『表エディター関連の大域変数』
- 302 ページの『さまざまな表示を制御する大域変数』
- 305 ページの『コマンドとプロシージャーの実行を制御する大域変数』
- 309 ページの『CONVERT QUERY の結果を保管する大域変数』
- 309 ページの『RUN QUERY エラー・メッセージ情報を示す大域変数』

### 大域変数の命名規則

Q.SYSTEM\_INI という特別なプロシージャーが QMF で提供され、それにより、初期化時に大域変数をカスタマイズすることができます。

呼び出し可能インターフェースのユーザーは、18 文字以下の名前の大域変数を使用できます。コマンド・インターフェースのユーザーは、8 文字以下の名前を使用する必要があります。

大域変数の命名規則は、 **DSQcc\_xxxxxxxxxxxxxx** で、ストリングの意味は次のとおりです。

**cc** 次のカテゴリー ID のいずれか。

**AP** プロファイル関連状態情報の変数

**AO** その他の（プロファイル関連でない）状態情報の変数

**CM** 前のコマンドで生じたメッセージに関する情報の変数

**CP** 表エディターに関する情報の変数

**DC** QMF が画面に情報を表示する方法の制御の変数

**EC** QMF がコマンドとプロシージャーを実行する方法を制御する変数

**QC** CONVERT QUERY オプションにより値が生成される変数

**QM** RUN QUERY エラー・メッセージ情報を含む変数

**QW** QMF のワークステーションごとに固有の変数（これらの変数の詳細は、「DB2 QMF for Workstation and DB2 QMF for WebSphere 入門」を参照してください）

— 下線文字

**XXXXXXXXXXXX**

最大 12 文字の記述的な名前

## プロファイル関連状態情報の大域変数

表 38 は、QMF プロファイル設定に関連する情報を保管する大域変数のセットを示しています。これらの大域変数のどれも、SET GLOBAL コマンドでは変更できません。

表 38. プロファイル関連状態情報の大域変数

呼び出し可能インターフェースの 変数名	コマンド・インターフェースの 変数名	長さ	説明
DSQAP_CASE	DSQAPCAS	01	CASE パラメーター。以下の値を取ることができます。  1 UPPER の場合 2 MIXED の場合 3 STRING の場合
			使用しているサイトが、TSO のもとで大/小文字混合パスワードをサポートする RACF を使用している場合は、この値は 2 に設定します。そうしないと、すべての入力（パスワードも含めて）は大文字に変換され、CONNECT コマンドが失敗します。CASE を MIXED に設定した場合は、すべての入力は大文字で入力してください。理由は、QMF が大文字のコマンドのみ認識するためです。
DSQAP_CONFIRM	DSQAPRMP	01	CONFIRM パラメーター。以下の値を取ることができます。  0 NO の場合 1 YES の場合
DSQAP_DECIMAL	DSQAPDEC	01	DECIMAL パラメーター。以下の値を取ることができます。  1 PERIOD の場合 2 COMMA の場合 3 FRENCH の場合
DSQAP_LENGTH	DSQAPLEN	18	LENGTH パラメーター。値はパラメーターの値です ('1' から '999' までか、'CONT')。
DSQAP_PFK_TABLE	DSQAPPFK	31	ファンクション・キー表の名前
DSQAP_PRINTER	DSQAPPRT	08	PRINTER パラメーター。以下の値を取ることができます。  • GDDM プリンターのニックネーム • ブランク (DSQPRINT と関連づけられたプリンター)
DSQAP_QUERY_LANG	DSQAPLNG	01	LANGUAGE パラメーター。以下の値を取ることができます。  1 - SQL 2 - QBE 3 PROMPTED の場合

表 38. プロファイル関連状態情報の大域変数 (続き)

呼び出し可能インターフェースの 変数名	コマンド・インターフェースの 変数名	長さ	説明
DSQAP_QUERY_MODEL	DSQAMODP	01	MODEL パラメーター (リレーショナルの場合、値は '1')。
DSQAP_RESOURC_GRP	DSQAPGRP	16	RESOURCE GROUP パラメーター。
DSQAP_SPACE	DSQAPSPC	50	SPACE パラメーター。値はパラメーターの値です。
DSQAP_SYNONYM_TBL	DSQAPSYN	31	現行の QMF セッションに使用される同義語表の名前。
DSQAP_TRACE	DSQAPTRC	18	TRACE パラメーター。以下の値を取ることができます。 <b>ALL</b> (最大トレース) <b>NONE</b> (最小トレース)  トレース対象のコンポーネントとトレース内容のレベルを示す一連の文字と数字を指定することができます (たとえば、A2L2C1)。コンポーネントを示す文字とトレース内容のレベルの詳細は、「 <i>TSO/CICS 用 DB2 QMF インストールおよび管理の手引き</i> 」を参照してください。
DSQAP_WIDTH	DSQAPWID	18	WIDTH パラメーター。値はパラメーターの値です ('22' から '999')。

## プロファイル関連でない状態情報の大域変数

表 39 は、状況情報を含むか、パラメーターまたはフラグの設定値を含む大域変数のセットを示しています。これらの大域変数のどれも、SET GLOBAL コマンドでは変更できません。

表 39. プロファイル関連でない状態情報の大域変数

呼び出し可能インターフェース の変数名	コマンド・インターフェースの 変数名	長さ	説明
DSQAO_APPL_TRACE	DSQATRAC	01	アプリケーション・トレース・レベル。以下の値を取ることができます。 <b>0</b> レベル A0 の場合 <b>1</b> レベル A1 の場合 <b>2</b> レベル A2 の場合
DSQAO_ATTENTION	DSQCATTN	01	ユーザー・アテンション・フラグ
DSQAO_BATCH	DSQABATC	01	バッチまたは対話モード。以下の値を取ることができます。 <b>1</b> 対話式セッションの場合 <b>2</b> バッチ・モード・セッションの場合
DSQAO_CONNECT_ID	DSQAAUTH	08	データベースへの接続に使用されるユーザー ID。このユーザー ID のもとで作業が行われます。
DSQAO_CONNECT_LOC	なし	18	ユーザーが現在接続しているデータベースのロケーション名。名前は 18 文字 (必要な場合は、右端をブランクで埋めます)。

## QMF 大域変数

表 39. プロファイル関連でない状態情報の大域変数（続き）

呼び出し可能インターフェースの変数名	コマンド・インターフェースの変数名	長さ	説明
DSQAO_CURSOR_OPEN	DSQACRSR	01	データベース・カーソル状況。以下の値を取ることができます。 <b>1</b> カーソルがオープン <b>2</b> カーソルがクローズ
DSQAO_DB_MANAGER	DSQADBMG	01	データベース・マネージャー。次のいずれかの値を選択します。 <b>1</b> DB2 (VM および VSE 版) <b>2</b> DB2 (z/OS 版) <b>3</b> DB2 (Linux、UNIX、および Windows 版) <b>4</b> DB2 (iSeries 版)
DSQAO_DBCS	DSQADBCS	01	DBCS サポート状況。以下の値を取ることができます。 <b>1</b> DBCS サポートあり <b>2</b> DBCS サポートなし
DSQAO_FORM_PANEL	DSQASUBP	02	現行の書式パネル。以下の値を取ることができます。 <b>1</b> - FORM.MAIN <b>2</b> - FORM.COLUMNS <b>3</b> - FORM.PAGE <b>4</b> - FORM.FINAL <b>5</b> - FORM.BREAK1 <b>6</b> - FORM.BREAK2 <b>7</b> - FORM.BREAK3 <b>8</b> - FORM.BREAK4 <b>9</b> - FORM.BREAK5 <b>10</b> - FORM.BREAK6 <b>11</b> - FORM.OPTIONS <b>12</b> - FORM.CALC <b>13</b> - FORM.DETAIL <b>14</b> - FORM.CONDITIONS ブランク値は、QMF 一時記憶域に書式がないことを意味します。
DSQAO_INTERACT	DSQAIACT	01	対話フラグの設定。以下の値を取ることができます。 <b>0</b> 対話式実行でない場合 <b>1</b> 対話式実行ができる場合

表 39. プロファイル関連でない状態情報の大域変数 (続き)

呼び出し可能インターフェースの変数名	コマンド・インターフェースの変数名	長さ	説明
DSQAO_LOCAL_DB2	なし	18	ローカル DB2 (z/OS 版) データベースのロケーション名。これは、変数 DSQAO_SUBSYS_ID で名指しされたサブシステムのロケーション名です。  リモート作業単位環境では、DSQ_LOCAL_DB2 はアプリケーション・リクエスターの名前。名前は 16 文字 (必要な場合は、右端をブランクで埋めます)。
DSQAO_LOCATION	DSQAITLO	16	現行オブジェクトが存在する場合、現行オブジェクトのロケーション名。この値は、3 部構成の名前が使用された場合に限り適用できます。
DSQAO_NLF_LANG	DSQALANG	01	ユーザーの国語。英語環境では、'E'。
DSQAO_NUM_FETCHED	DSQAROWS	16	取り出されたデータ行。DATA オブジェクトが空の場合は '0'。
DSQAO_OBJ_NAME	DSQAITMN	128	現在表示されているパネル上にある表 (報告書に含まれているもの)、照会、プロシージャー、書式の名前。  現パネル上にどのオブジェクトも表示されていないとき、または表示されているオブジェクトに名前がないときは、この変数にはブランクが入っています。
DSQAO_OBJ_OWNER	DSQAITMO	128	現在表示されているパネル上にある表 (報告書に含まれているもの)、照会、プロシージャー、書式の所有者。  現パネル上にどのオブジェクトも表示されていないとき、または表示されているオブジェクトに所有者がいるときは、この変数にはブランクが入っています。
DSQAO_PANEL_TYPE	DSQAITEM	01	現行パネルのタイプ。以下の値を取ることができます。  1 - HOME 2 - QUERY 3 - REPORT 4 - FORM 5 - PROC 6 - PROFILE 7 - CHART 8 - LIST 9 - 表エディター A - GLOBALS
DSQAO_QMF_RELEASE	DSQAREVN	02	QMF のリリース番号を表す数値で、エクスポートされた書式、報告書、および指示照会のヘッダー・レコードに現れます。QMF バージョン 9.1 の場合、これは '15' です。

## QMF 大域変数

表 39. プロファイル関連でない状態情報の大域変数 (続き)

呼び出し可能インターフェース の変数名	コマンド・インターフェースの 変数名	長さ	説明
DSQAO_QMF_VER_RLS	DSQAQMFI	10	QMF のバージョンとリリース。 QMF バージョン 9.1 の場合、これは 'QMF V9R1.0' です。
DSQAO_QMFADM	なし	01	<p>QMF 管理者権限。</p> <p><b>0</b> 現行の許可 ID は QMF 管理者権限を持っていません。</p> <p><b>1</b> 現行の許可 ID は管理者権限を持っています。</p>
DSQAO_QRY_SUBTYPE	DSQASUBI	01	<p>照会サブタイプ。以下の値を取ることができます。</p> <p><b>1</b> - SQL サブタイプ</p> <p><b>2</b> - QBE サブタイプ</p> <p><b>3</b> - PROMPTED サブタイプ</p> <p>ブランクは、現パネルが QUERY でないことを意味します。</p>
DSQAO_QUERY_MODEL	DSQAMODL	01	現行の照会のモデル。値は '1' (リレーションナル・データ・モデル) のみ指定できます。
DSQAO_SAME_CMD	DSQACMDM	01	<p>以下の値を取ることができます。</p> <p><b>0</b> 2 つのコマンドが同じでない場合</p> <p><b>1</b> 2 つのコマンドが同じである場合</p>
DSQAO_SUBSYS_ID	なし	04	<p>QMF が TSO で稼働しているとき、QMF が接続しているローカル DB2 サブシステムの ID。</p> <p>CICS の中で DSQSSUBS プログラム・パラメーターの値を指定すると、この大域変数はその値を含みます。これは、そのパラメーターが默認され、値が処理されないからです。その値は大域変数フィールドに置かれ、何も処理されません。この論理によって、同一のプログラムを複数の環境で使用することが可能になります。</p>
DSQAO_SYSTEM_ID	DSQASYST	01	現行のオペレーティング・システム。以下の値を取ることができます。
			<p><b>2</b> z/OS のもとの TSO</p> <p><b>3</b> TSO またはネイティブの z/OS</p> <p><b>5</b> CICS</p>
DSQAO_TERMINATE	DSQCSESC	01	<p>QMF 終了フラグ。以下の値を取ることができます。</p> <p><b>0</b> セッションが終了とマークされなかった場合</p> <p><b>1</b> セッションが終了とマークされた場合</p>
DSQAO_VARIATION	DSQAVARN	02	書式パネル・バリエーション番号。ブランクは、FORM.DETAIL が現行のパネルではないことを意味します。

## CICS 関連の大域変数

表 40 に示した変数のうち、SET GLOBAL コマンドで変更できるのは DSQAP\_CICS\_PQNAME と DSQAP\_CICS\_PQTYPE のみです。

キュー・タイプが一時データ (TD) であれば、対応するキューナーの最大長は 4 です。たとえば、DSQAO\_CICS\_SQTYPE が TD であれば、DSQAO\_CICS\_SQNAME の最大長は 4 です。

表 40. CICS 環境関連の大域変数

呼び出し可能インターフェースの変数名	コマンド・インターフェースの変数名	長さ	説明
DSQAP_CICS_PQNAME	なし	08	QMF 印刷出力を入れる CICS データ・キューの名前。
DSQAP_CICS_PQTYPE	なし	02	<p>QMF 印刷出力を入れる CICS 記憶域のタイプ。</p> <p><b>TS</b> QMF 印刷を、補助記憶装置上の CICS 一時記憶域キューに書き出します。これはデフォルトです。</p> <p><b>TD</b> QMF 印刷を CICS 一時データ・キューに書き出します。</p>
DSQAO_CICS_SQNAME	なし	08	予備ファイルに使用する CICS データ・キューの名前。
DSQAO_CICS_SQTYPE	なし	02	<p>QMF 予備ファイルを入れる CICS 記憶域のタイプ。</p> <p><b>TS</b> QMF 予備データを、補助記憶装置上の CICS 一時記憶域キューに書き出します。これはデフォルトです。</p> <p><b>TD</b> QMF 予備データを CICS 一時データ・キューに書き出します。</p>
DSQAO_CICS_TQNAME	なし	08	QMF トレース・データを入れる CICS データ・キューの名前。
DSQAO_CICS_TQTYPE	なし	02	<p>QMF トレース・データを入れる CICS 記憶域のタイプ。</p> <p><b>TS</b> QMF トレースを、補助記憶装置上の CICS 一時記憶域キューに書き出します。</p> <p><b>TD</b> QMF トレースを CICS 一時データ・キューに書き出します。これはデフォルトです。</p>

## 最後に発行されたコマンドで生成されたメッセージに関する大域変数

300 ページの表 41 は、最後に発行された QMF コマンドに関する情報を取り込む大域変数を示しています。これらの大域変数のどれも、SET GLOBAL コマンドでは変更できません。

## QMF 大域変数

表 41. 最後に発行されたコマンドに関する情報を取り込む大域変数

呼び出し可能インターフェースの 変数名	コマンド・インターフェースの 変数名	長さ	説明
DSQCM_MESSAGE	DSQCIMSG	80	メッセージ・テキスト
DSQCM_MESSAGE_ALL	DSQCIMSA	360	完全なメッセージ・テキスト
DSQCM_MSG_HELP	DSQCIMID	08	メッセージ・ヘルプ・パネルの ID
DSQCM_MSG_NUMBER	DSQCIMNO	08	メッセージ番号
DSQCM_SUB_TXT_nn	DSQCIMnn	20	置換値 nn
DSQCM_SUBST_VARS	DSQCIM00	04	メッセージ中にある置換変数の個数

## 表エディター関連の大域変数

表 42 は、表エディターの操作に関する大域変数を示しています。いずれも SET GLOBAL コマンドで変更できます。

EDIT TABLE の CONFIRM オプションが NO なら、表エディターはすべての確認パネルの表示を抑止します。CONFIRM オプションが YES なら、表エディターはこの表に示す大域変数の値を調べ、どのカテゴリーの確認が有効になっているか判断します。

表エディターのデフォルト値は、EDIT TABLE コマンドの SAVE キーワードに応じて次のように異なります。

- SAVE=IMMEDIATE なら、各カテゴリーのデフォルト値が有効です。
- SAVE=END なら、DELETE、MODIFY、END/CANCEL カテゴリーのデフォルト値が有効です。ADD カテゴリーと CHANGE カテゴリーのデフォルト値は無効です。

表 42. 表エディター関連の大域変数

呼び出し可能インターフェースの 変数名	コマンド・インターフェースの 変数名	長さ	説明
DSQCP_TEADD	なし	01	ADD サブコマンドのあと、確認パネルを表示するかどうか。以下の値を取ることができます。  0 - パネルを表示しません。 1 - パネルを表示します。 2 パネルを表示するかどうかは、表エディターのデフォルト値によります。これはデフォルトです。
DSQCP_TECHG	なし	01	CHANGE サブコマンドのあと、確認パネルを表示するかどうか。以下の値を取ることができます。  0 - パネルを表示しません。 1 - パネルを表示します。 2 パネルを表示するかどうかは、表エディターのデフォルト値によります。これはデフォルトです。

表42. 表エディター関連の大域変数 (続き)

呼び出し可能インターフェースの 変数名	コマンド・インターフェースの 変数名	長さ	説明
DSQCP_TEEND	なし	01	<p>END サブコマンドまたは CANCEL サブコマンドを発行して表エディター・サブセッションを終了させるとき、確認パネルを表示するかどうか。</p> <p>出されたのが END か CANCEL か、データベースへの変更が行なわれたのか、END または CANCEL が出されたとき画面に変更後のデータが含まれていたかどうかによって、いくとおりかのパネルが表示されます。以下の値を取ることができます。</p> <ul style="list-style-type: none"> <li><b>0</b> - パネルを表示しません。</li> <li><b>1</b> - パネルを表示します。</li> <li><b>2</b> パネルを表示するかどうかは、表エディターのデフォルト値によります。これはデフォルトです。</li> </ul>
DSQCP_TEDEL	なし	01	<p>DELETE サブコマンドのあと、確認パネルを表示するかどうか。以下の値を取ることができます。</p> <ul style="list-style-type: none"> <li><b>0</b> - パネルを表示しません。</li> <li><b>1</b> - パネルを表示します。</li> <li><b>2</b> パネルを表示するかどうかは、表エディターのデフォルト値によります。これはデフォルトです。</li> </ul>
DSQCP_TEDFLT	なし	01	表エディターで列のデフォルト値を示すのに使用される予約文字。最初は、正符号 (+) 文字に設定されています。
DSQCP_TEDFLT_DBCS	なし	04	<p>表エディターで漢字ストリング列のデフォルト値を示すのに使用される DBCS の予約文字。</p> <p>この値は、4 バイトの混合ストリングで、その前にシフトアウト文字を伴い、その後にシフトイン文字を伴う 1 文字の DBCS 文字で構成されます。最初は、DBCS の正符号 (+) 文字に設定されています。この大域変数は DBCS 環境でのみ使用されます。</p>
DSQCP_TEMOD	なし	01	<p>表示されたデータが変更され、PREVIOUS、CLEAR、SHOW CHANGE、SHOW SEARCH、REFRESH、または NEXT の各サブコマンドが実行されるときに、確認パネルを表示するかどうか。結果のパネルには、パネル・テキストの一部として、サブコマンドの名前が組み込まれます。以下の値を取ることができます。</p> <ul style="list-style-type: none"> <li><b>0</b> - パネルを表示しません。</li> <li><b>1</b> - パネルを表示します。</li> <li><b>2</b> パネルを表示するかどうかは、表エディターのデフォルト値によります。</li> </ul>

## QMF 大域変数

表 42. 表エディター関連の大域変数 (続き)

呼び出し可能インターフェースの 変数名	コマンド・インターフェースの 変数名	長さ	説明
DSQCP_TENULL	なし	01	表エディターで列の NULL 値を示すのに使用される予約文字。最初は、ハイフン (-) 文字に設定されています。
DSQCP_TENULL_DBCS	なし	04	表エディターで漢字ストリング列の NULL 値(または検索基準の文脈では「無視」)を示すに使用される DBCS の予約文字。  この値は、4 バイトの混合ストリングで、その前にシフトアウト文字を伴い、その後にシフトイン文字を伴う 1 文字の DBCS 文字で構成されます。最初は、DBCS のハイフン (-) 文字に設定されています。この大域変数は DBCS 環境でのみ使用されます。

## さまざまな表示を制御する大域変数

表 43 は、特定の種類の情報の表示を制御する大域変数を示しています。いずれも SET GLOBAL コマンドで変更できます。

表 43. 特定の種類の情報の表示を制御する大域変数

呼び出し可能インターフェースの 変数名	コマンド・インターフェースの 変数名	長さ	説明
DSQDC_COST_EST	なし	01	データベースのコスト見積の表示を制御します。以下の値を取ることができます。  0 コスト見積もりを表示しません。 1 コスト見積もりを表示します。これはデフォルトです。
DSQDC_CURRENCY	なし	18	DC 編集コードの指定がある場合に使用される通貨記号。  この値は、1 から 18 バイトまれの長さのストリングです。英語の場合、デフォルト値はユーロ通貨記号です。デフォルト値は言語によって異なります。DBCS 環境では、この値は SBCS と DBCS 文字の混合ストリングでできます。シフトアウト文字とシフトイン文字を含む混合ストリングの合計の長さは、18 バイトを超えることはできません。

表 43. 特定の種類の情報の表示を制御する大域変数 (続き)

呼び出し可能インターフェースの 変数名	コマンド・インターフェースの 変数名	長さ	説明
DSQDC_DISPLAY_RPT	DSQADPAN	01	<p>RUN QUERY のあとで、報告書を表示するかどうか。以下の値を取ることができます。</p> <p><b>0</b> QMF は、RUN QUERY コマンドからの結果の報告書を表示しません。</p> <p>QMF を DSQQMFn (n は使用している各国語機能を表す文字) で対話式に起動したとき、または BATCH モードで起動した場合、これがデフォルト値です。QMF が BATCH モードで起動している場合にこの変数を変更しても、QMF 画面は表示されません。各国語機能と 1 文字の ID のリストは、「<i>TSO/CICS 用 DB2 QMF インストールおよび管理の手引き</i>」を参照してください。</p> <p><b>1</b> QMF は自動的に報告書を表示します。</p> <p>QMF が呼び出し可能インターフェースで起動された場合は、これがデフォルト値です。START コマンドの DSQADPAN プログラム・パラメーターがこの大域変数に優先します。</p> <p>この大域変数は、アプリケーション専用です。コマンド行から入力された RUN QUERY コマンドには影響がありません。</p>

## QMF 大域変数

表 43. 特定の種類の情報の表示を制御する大域変数（続き）

呼び出し可能インターフェースの 変数名	コマンド・インターフェースの 変数名	長さ	説明
DSQDC_LIST_ORDER	なし	02	<p>データベース・オブジェクトのリストでのオブジェクトのデフォルトのソート順を設定します。先頭文字の値:</p> <ul style="list-style-type: none"> <li><b>1</b> リストはデフォルトの順を使用します。</li> <li><b>2</b> リストはオブジェクト所有者によりソートされます。</li> <li><b>3</b> リストはオブジェクト名によりソートされます。</li> <li><b>4</b> リストはオブジェクト・タイプによりソートされます。</li> <li><b>5</b> リストは変更日付によりソートされます。</li> <li><b>6</b> リストは最後に使用された日付によりソートされます。</li> </ul> <p>2 番目の文字の値:</p> <ul style="list-style-type: none"> <li><b>A</b> リストは昇順にソートされます。</li> <li><b>D</b> リストは降順にソートされます。</li> </ul> <p>この変数は、LIST コマンドの結果としてリストされるオブジェクトについてのみ適用されます。表示プロンプト・パネルからなどの他の文脈でつくられたリストには適用されません。また、表のリストにも適用されません。</p>
DSQDC_SCROLL_AMT	なし	04	<p>QMF パネルのスクロール量を設定します。以下の値を取ることができます。</p> <ul style="list-style-type: none"> <li><b>Csr</b> スクロール量を「カーソル」に設定します。 後方、前方、左、または右にスクロールするかどうかによって、QMF は、カーソルがある位置の行または列をスクロール可能域の最下部、最上部、左端、または右端にスクロールします。</li> <li><b>Half</b> スクロール量をスクロール可能域の半分に設定します。</li> <li><b>Page</b> スクロール量を全ページに設定します。これはデフォルトです。</li> <li><b>n</b> スクロール量を <i>n</i> 行数または列数に設定します。<i>n</i> には 1 から 9999 までの任意の値を指定できます。</li> </ul>

表 43. 特定の種類の情報の表示を制御する大域変数 (続き)

呼び出し可能インターフェースの 変数名	コマンド・インターフェースの 変数名	長さ	説明
DSQDC_SHOW_PANID	DSQCPDSP	01	<p>QMF パネルのパネル ID を表示するかどうか。以下の値を取ることができます。</p> <p><b>0</b> パネル ID を表示しません。これはデフォルトです。</p> <p><b>1</b> パネル ID を表示します。</p>

## コマンドとプロシージャーの実行を制御する大域変数

表 44 は、コマンドとプロシージャーの実行を制御する大域変数を示しています。いずれも SET GLOBAL コマンドで変更できます。

表 44. コマンドとプロシージャーの実行を制御する大域変数

呼び出し可能インターフェースの 変数名	コマンド・インターフェースの 変数名	長さ	説明
DSQEC_ALIASES	なし	31	ユーザーが DB2 (z/OS 版) ロケーションから表リストを要求したとき、または現行サーバーが DB2 (z/OS 版) または DB2 (Linux, UNIX, および Windows 版) である場合に、表と視点の別名のリストを取り出すための視点。
DSQEC_CC	なし	01	<p>紙送り制御文字が報告書出力フォーマットに出力されないようにします。以下の値を取ることができます。</p> <p><b>0</b> 紙送り制御文字を 1 桁目に出来ません。</p> <p><b>1</b> 紙送り制御が有効です。報告書の 1 桁目に紙送り制御文字が出力されます。</p>
DSQEC_COLS_LDB2	なし	31	現行ロケーションが DB2 (z/OS 版) のとき、そこにある表の列情報を取り出すための視点。
DSQEC_COLS_RDB2	なし	31	リモート DB2 (z/OS 版) が現行ロケーションでないとき、そこにある表の列情報を取り出すための視点。
DSQEC_COLS_SQL	なし	31	DB2 (VM または VSE 版) データベース内の表の列情報を検索するための視点。

## QMF 大域変数

表 44. コマンドとプロシージャーの実行を制御する大域変数 (続き)

呼び出し可能インターフェースの 変数名	コマンド・インターフェースの 変数名	長さ	説明
DSQEC_DISABLEADM	なし	01	<p>QMF 管理者権限の抑止。この大域変数の値が変更されると、それが即時に反映されます。指定できる値は次のとおりです。</p> <p><b>0</b> QMF 管理者権限が使用可能です (許可 ID が QMF 管理者権限を持つ場合)。</p> <p><b>1</b> QMF 管理者権限が抑止されます (許可 ID の権限にかかわらず)。</p> <p>注: この大域変数の初期のデフォルト値は、QMF インストール・システム出口ルーチンによってオーバーライドされます。この出口ルーチンの詳細は、「<i>TSO/CICS 用 DB2 QMF インストールおよび管理の手引き</i>」を参照してください。</p>
DSQEC_DSALLOC_DIR	なし	3	<p>TSO の中で、新規 PDS データ・セットのメンバーをエクスポートするときに使用するディレクトリー・ブロック数を指定します。PDS データ・セットの場合、値はゼロより大きくなればなりません。</p> <p>使用しているサイトのデフォルトのタイプのデータ・セット (DSQEC_PO を 0 に設定して指定) または PDSE データ・セット (DSQEC_PO を 2 に設定して指定) を使用している場合、QMF は、この大域変数の値を無視します。</p> <p>使用しているサイトが順次データ・セットを使用する場合、この大域変数はゼロに設定します。</p>
DSQEC_DSALLOC_PRI	なし	8	<p>QMF はデータ・セットをトラック数で割り振ります。この大域変数は、QMF EXPORT コマンドの結果の保管に使用される TSO データ・セットの 1 次スペース割り振りのトラック数を指定します。</p> <p>指定できる値は、1 から、ストレージ・デバイスおよびオペレーティング・システムで許される最大サイズまでです。デフォルト値は 15 です。値ゼロは指定できません。</p> <p>PS および PDS データ・セットは、最大 65535 トラックをもつことができます。PDSE データ・セットは最大 16777215 トラックをもつことができます。</p>

表44. コマンドとプロシージャの実行を制御する大域変数 (続き)

呼び出し可能インターフェースの 変数名	コマンド・インターフェースの 変数名	長さ	説明
DSQEC_DSALLOC_SEC	なし	8	<p>QMF はデータ・セットをトラック数で割り振ります。この大域変数は、QMF EXPORT コマンドの結果の保管に使用される TSO データ・セットの 2 次スペース割り振りのトラック数を指定します。</p> <p>指定できる値は、ゼロから、ストレージ・デバイスおよびオペレーティング・システムで許される最大サイズまでです。デフォルト値は、105 トラックです。</p> <p>PS および PDS データ・セットは、最大 65535 トラックをもつことができます。PDSE データ・セットは最大 16777215 トラックをもつことができます。</p>
DSQEC_FORM_LANG	なし	01	<p>保管またはエクスポートされる書式のデフォルトの NLF 言語を定めます。以下の値を取ることができます。</p> <p><b>0</b> 書式には現行の NLF 言語を使用します。</p> <p><b>1</b> 書式には英語を使用します。これはデフォルトです。</p>
DSQEC_ISOLATION	なし	01	<p>デフォルトの照会分離レベル。以下の値を取ることができます。</p> <p><b>0</b> 分離レベル UR (非コミット読み取り)。</p> <p><b>1</b> 分離レベル CS (カーソル固定)。これはデフォルトです。</p> <p><b>重要:</b> 値を '0' に設定すると、存在しないデータが QMF 報告書に入ることがあります。QMF 報告書に存在しないデータが入らないようにするには、値を '0' に設定しないでください。</p>
DSQEC_NLFCMD_LANG	なし	01	<p>コマンドに使う NLF 言語を定めます。以下の値を取ることができます。</p> <p><b>0</b> コマンドには現行の NLF 言語を使用します。これはデフォルトです。</p> <p><b>1</b> コマンドには英語を使用します。</p>

## QMF 大域変数

表 44. コマンドとプロシージャーの実行を制御する大域変数 (続き)

呼び出し可能インターフェースの 変数名	コマンド・インターフェースの 変数名	長さ	説明
DSQEC_PO	なし	1	<p>QMF オブジェクトを新規 TSO データ・セットにエクスポートするときに作成する、区分(PO) データ・セットのタイプを指定します。以下の値を取ることができます。</p> <p><b>0</b> データ・セットのタイプを、使用しているサイトのデフォルトとしてリストされたタイプに割り振ります。このタイプは、SYS1.PARMLIB の IGDSMSXX メンバーの中に指定されています。これはデフォルト値です。</p> <p><b>1</b> エクスポートされるデータに、PDS データ・セットを割り振ります。</p> <p><b>2</b> エクスポートされるデータに、PDSE データ・セットを割り振ります。</p>
DSQEC_RERUN_IPROC	なし	01	<p>END コマンドの実行後、呼び出しプロシージャーを再実行します。以下の値を取ることができます。</p> <p><b>0</b> END コマンドの実行後、呼び出しプロシージャーを再実行しません。</p> <p><b>1</b> END コマンドの実行後、呼び出しプロシージャーを再実行します。これはデフォルトです。</p> <p>QMF を呼び出しプロシージャーで起動しておいてから、この変数を' 0' に設定すると、プロシージャー再実行の代りに QMF が終了します。</p>
DSQEC_RESET_RPT	なし	31	<p>一時記憶域にある不完全な DATA オブジェクトがパフォーマンスに影響するように見える場合、QMF がユーザーにプロンプトを出すかどうかを決定します。</p> <p><b>0</b> リセット報告書プロンプト・パネルは表示されず、QMF は報告書の実行を完了します。これはデフォルト値です。</p> <p><b>1</b> リセット報告書プロンプト・パネルが表示されます。このパネルは、新しいコマンドを開始する前に現在実行中の報告書を完了させるかリセットするかを尋ねるプロンプトを出します。</p> <p><b>2</b> リセット報告書プロンプト・パネルは表示されず、QMF は現在、実行中の報告書を停止させます。</p>

表44. コマンドとプロシージャーの実行を制御する大域変数 (続き)

呼び出し可能インターフェースの 変数名	コマンド・インターフェースの 変数名	長さ	説明
DSQEC_SHARE	なし	31	SHARE パラメーターのデフォルト値を指定します。指定できる値は次のとおりです。 <b>0</b> データを他のユーザーと共に用しない。 <b>1</b> データを他のユーザーと共に用する。
DSQEC_TABS_LDB2	なし	31	現行サーバーが DB2 (z/OS 版) または DB2 (Linux、UNIX、および Windows 版) である場合に、現行サーバーにある表と視点のリストを取り出すための視点。
DSQEC_TABS_RDB2	なし	31	リモート DB2 サブシステムにある表と視点のリストを取り出すための視点。
DSQEC_TABS_SQL	なし	31	DB2 (VM または VSE 版) データベースの表および視点のリストを取り出すための視点。

## CONVERT QUERY の結果を保管する大域変数

表 45 は、CONVERT QUERY コマンドの結果を反映する大域変数を示しています。これらの大域変数のどれも、SET GLOBAL コマンドでは変更できません。

表45. CONVERT QUERY コマンドの結果を反映する大域変数

呼び出し可能インターフェース の変数名	コマンド・インターフェースの 変数名	長さ	説明
DSQQC_LENGTH_nnn	DSQCLnnn	05	変換結果の長さ nnn
DSQQC_QRY_COUNT	DSQQCQNT	03	変換結果に含まれる照会の数。もとの照会が QBE I. または QBE U. 照会でない限り、値は常に '1' でなければなりません。
DSQQC_QRY_LANG	DSQCQLNG	01	変換された照会の言語。以下の値を取ることができます。 <b>1</b> - SQL <b>2</b> - QBE <b>3</b> - プロンプト
DSQQC_QRY_TYPE	DSQCQTYP	指定なし	変換結果中の最初の値
DSQQC_RESULT_nnn	DSQCQnnn	指定なし	変換結果 nnn

## RUN QUERY エラー・メッセージ情報を示す大域変数

表 46 は、RUN QUERY コマンドの結果を保管する大域変数を示しています。これらの大域変数のどれも、SET GLOBAL コマンドでは変更できません。

表46. RUN QUERY コマンドの結果を保管する大域変数

呼び出し可能インターフェースの 変数名	コマンド・インターフェースの 変数名	長さ	説明
DSQQM_MESSAGE	DSQCIQMG	80	照会メッセージのテキスト
DSQQM_MESSAGE_ALL	DSQCIQMA	360	完全照会メッセージ・テキスト
DSQQM_MSG_HELP	DSQCIQID	08	メッセージ・ヘルプ・パネルの ID

## QMF 大域変数

表 46. RUN QUERY コマンドの結果を保管する大域変数 (続き)

呼び出し可能インターフェースの 変数名	コマンド・インターフェースの 変数名	長さ	説明
DSQQM_MSG_NUMBER	DSQCIQNO	08	メッセージ番号
DSQQM_SQL_RC	DSQCISQL	16	最後のコマンドまたは照会からの SQLCODE
DSQQM_SQL_STATE	なし	05	データベース・マネージャーから返されたとき、DSQQM_SQL_RC の SQLCODE と関連づけられている SQLSTATE。
DSQQM_SUB_TXT_nn	DSQCIQnn	20	置換値 nn
DSQQM_SUBST_VARS	DSQCIQ00	04	置換変数の数

## 付録 C. 特定のサポートを必要とする QMF 機能

表 47. これらの機能では、特定のデータベース管理システムのサポートを必要とします。

サポートされる機能	DB2 (z/OS 版)	DB2 (Linux、UNIX、 および Windows 版)	DB2 Server (VSE または VM 版)
照会ステートメントの長さ	32,765	32,765	8,192
SELECT ステートメントの列数	750	255	255
単精度浮動小数点数のインポート	X		X
LIKE ステートメントによる 長フィールド	X		X
データベース同義語	X		X
表または視点のデータベース別名	X		X
表エディターで	X		X
SAVE=IMMEDIATE オプションが 使用可能 (CURSOR HOLD をサポ ート)			
分散作業単位 (3 部分名)	X		
リモート作業単位	X	X	VSE では、バ ージョン 3 リリー ス 4 が必要

以下の QMF および QMF 関連機能は CICS では使用できません。

- コマンド・インターフェース
- EDIT PROC
- EDIT QUERY
- 文書インターフェース
- BATCH アプリケーション
- トランザクションの取り消し
- ISPF
- DPRE
- 報告書の計算
- 外部変数
- LAYOUT アプリケーション
- 条件付きフォーマット設定
- 列定義
- ロジックを持つプロシージャー

## 特定のサポートを必要とする QMF 機能

---

## 付録 D. 特記事項

本書は米国 IBM が提供する製品およびサービスについて作成したものです。本書に記載の製品、サービス、または機能が日本においては提供されていない場合があります。日本で利用可能な製品、サービス、および機能については、日本 IBM の営業担当員にお尋ねください。本書で IBM 製品、プログラム、またはサービスに言及していても、その IBM 製品、プログラム、またはサービスのみが使用可能であることを意味するものではありません。これらに代えて、IBM の知的所有権を侵害することのない、機能的に同等の製品、プログラム、またはサービスを使用することができます。ただし、IBM 以外の製品とプログラムの操作またはサービスの評価および検証は、お客様の責任で行っていただきます。

IBM は、本書に記載されている内容に関して特許権（特許出願中のものを含む）を保有している場合があります。本書の提供は、お客様にこれらの特許権について実施権を許諾することを意味するものではありません。実施権についてのお問い合わせは、書面にて下記宛先にお送りください。

〒106-8711  
東京都港区六本木 3-2-12  
IBM World Trade Asia Corporation  
Intellectual Property Law & Licensing

**以下の保証は、国または地域の法律に沿わない場合は、適用されません。** IBM およびその直接または間接の子会社は、本書を特定物として現存するままの状態で提供し、商品性の保証、特定目的適合性の保証および法律上の瑕疵担保責任を含むすべての明示もしくは黙示の保証責任を負わないものとします。国または地域によっては、法律の強行規定により、保証責任の制限が禁じられる場合、強行規定の制限を受けるものとします。

この情報には、技術的に不適切な記述や誤植を含む場合があります。本書は定期的に見直され、必要な変更は本書の次版に組み込まれます。IBM は予告なしに、隨時、この文書に記載されている製品またはプログラムに対して、改良または変更を行うことがあります。

IBM は、お客様が提供するいかなる情報も、お客様に対してなんら義務も負うことのない、自ら適切と信ずる方法で、使用もしくは配布することができるものとします。

本プログラムのライセンス保持者で、(i) 独自に作成したプログラムと他のプログラム（本プログラムを含む）との間での情報交換、および (ii) 交換された情報の相互利用を可能にすることを目的として、本プログラムに関する情報を必要とする方は、下記に連絡してください。

IBM Corporation  
J46A/G4  
555 Bailey Avenue  
San Jose, CA 95141-1003  
U.S.A.

本プログラムに関する上記の情報は、適切な使用条件の下で使用することができますが、有償の場合もあります。

本書で説明されているライセンス・プログラムまたはその他のライセンス資料は、IBM 所定のプログラム契約の契約条項、IBM プログラムのご使用条件、またはそれと同等の条項に基づいて、 IBM より提供されます。

本書には、日常の業務処理で用いられるデータや報告書の例が含まれています。より具体性を与えるために、それらの例には、個人、企業、ブランド、あるいは製品などの名前が含まれている場合があります。これらの名称はすべて架空のものであり、名称や住所が類似する企業が実在しているとしても、それは偶然にすぎません。

---

## 商標

以下は、IBM Corporation の商標です。

CICS	IBM
DB2	iSeries
Distributed Relational Database Architecture	OS/390
DRDA	QMF
eServer	RACF
GDDM	z/OS

Intel は、Intel Corporation の米国およびその他の国における商標です。

Linux は、Linus Torvalds の米国およびその他の国における商標です。

Windows は、Microsoft Corporation の米国およびその他の国における商標です。

UNIX は、The Open Group の米国およびその他の国における登録商標です。

他の会社名、製品名およびサービス名等はそれぞれ各社の商標です。

## 用語集

### [ア行]

**宛先管理テーブル (destination control table - DCT).** CICS で、それぞれの一時データ・キューの宛先を収めているテーブル。

**アプリケーション (application).** QMF プログラム自体を変更せずに QMF の機能を拡張する、QMF ユーザー作成のプログラム。QMF プロシージャーの RUN コマンド、地域別定義コマンド、あるいは CLIST を呼び出す TSO コマンドを発行することによって、QMF セッションから開始される。

**アプリケーション・サーバー (application server).** アプリケーション・リクエスターからの要求のターゲット。(1) アプリケーション・プロセスが接続されるローカルまたはリモートのデータベース・マネージャー。アプリケーション・サーバーは、所要のデータをもっているシステムで実行される。(2) DRDA では、アプリケーション・リクエスターからの要求のターゲット。DB2 (z/OS 版) では、アプリケーション・サーバーは、完全な DB2 (z/OS 版) サブシステムの一部である。

DB2 (VM および VSE 版) では、アプリケーション・サーバーは、DB2 (VM および VSE 版) データベース・マシンの一部である。

**アプリケーション・サポート・コマンド (application-support command).** アプリケーション・プログラム内で使用され、アプリケーション・プログラムと QMF 間で情報を交換できるようにする、QMF コマンドの 1 つ。この種のコマンドには、INTERACT、MESSAGE、STATE、および QMF がある。

**アプリケーション・リクエスター (application requester).** (1) アプリケーション・プロセスからデータベース要求を受け入れ、それをアプリケーション・サーバーに渡す機能。(2) DRDA において、リモート・リレーショナル・データベース管理システムへの送信元。

アプリケーション・リクエスターは、分散接続の QMF 終端を扱うデータベース管理システム・コードである。QMF が接続するローカル DB2 (z/OS 版) サブシステムは、QMF に対するアプリケーション・リクエスターとして知られる。DB2 (z/OS 版) のアプリケーション・リクエスターはローカル・データベース・マネージャー内でインストールされるからである。したがって、DB2 (z/OS 版) サブシステム全体 (データを含む) がアプリケーション・リクエスターに関連するが、SQL ステートメントは現行ロケーションで処理される。

**異常終了 (abend).** タスクの異常終了。

**一時記憶域 (temporary storage).** 現在使用している QUERY、FORM、PROC、PROFILE、REPORT、DATA、および CHART オブジェクトが保管されている区域。

**一時記憶域キュー (temporary storage queue).** CICS において、QMF とアプリケーションまたはシステム・サービスの間のオブジェクト転送に用いられる一時記憶域。

**一時データ・キュー (transient data queue).** CICS において、宛先管理テーブル (DCT) で定義されている名前をもつ記憶域。ここには、後続の内部処理または外部処理に備えてオブジェクトが保管される。

**永続記憶域 (permanent storage).** すべての表および QMF オブジェクトが保管されているデータベース。一時記憶域 (temporary storage) と対比。

### [カ行]

**拡張構文 (extended syntax).** QMF 呼び出し可能インターフェースで使用する QMF コマンド構文。この構文は、呼び出し可能インターフェース・アプリケーションで獲得され、QMF と共に共用される記憶域に保管される変数を定義する。

## 用語集

**確認域 (echo area).** 指示照会を作成する指示照会基本パネルの一部。

**キーワード・パラメーター (keyword parameter).** キーワードと割り当て値から構成される QMF コマンドの 1 エレメント。

**基本 QMF 環境 (base QMF environment).** QMF のインストール時に設定される QMF の英語環境。他の言語環境は、インストール後に設定される。

**基本パネル (primary panel).** ユーザーの照会を入れる指示照会のメイン・パネル。

**行演算子域 (row operator area).** QBE ターゲット表または例示表の最左端の列。

**区域分離記号 (area separator).** 表示される報告書の固定域とその他の区域を区分するバリア。

**組み込み関数 (built-in function).** スカラー関数または列関数の総称。「関数」と呼ばれることもある。

**グループ化行 (grouped row).** G キーワードまたは組み込み関数のいずれかで合計される QBE ターゲット表または例示表のデータ行。

**計算変数 (calculation variable).** CALCid は、ユーザ一定義の計算値が入る書式用の特殊変数である。各 CALCid は FORM.CALC パネルで定義される。

**現行オブジェクト (current object).** 一時記憶域にあって、現在表示されているオブジェクト。

**現行ロケーション (current location).** QMF セッションが現在接続されているアプリケーション・サーバー。CONNECT などの接続タイプのステートメント（これらはアプリケーション・リクエスターによって処理される）を除いて、このサーバーはすべての SQL ステートメントを処理する。QMF を初期化する場合、現行ロケーションは DSQSDBNM 始動プログラム・パラメーターにより指示される。

**構造化照会言語 (SQL).** リレーションナル・データベースのデータを定義して操作するための標準化された言語。

**顧客情報管理システム (Customer Information Control System (CICS)).** リモート端末で入力されるトランザクションをユーザー作成アプリケーション・プログラムによって並行して処理できるようにする IBM ライセンス・プログラム。これには、データベースの構築、使用、維持管理の機能が含まれる。

**コマンド同義語 (command synonym).** サイトの定義コマンドの verb または verb / オブジェクトの部分。ユーザーは、コマンドの代わりにコマンド同義語を入力し、その後に他の必要情報を続ける。

**コマンド同義語表 (command synonym table).** それぞれの行にサイトの定義コマンドを記述する表。各ユーザーに、これらの表の 1 つを割り当てることができる。

**コマンド・インターフェース (command interface).** QMF コマンドを実行するためのインターフェース。QMF コマンドは、アクティブ QMF セッションからしか出せない。呼び出し可能インターフェース (*callable interface*) と対比。

**コミット (commit).** データに永続的な変更を加える処理。コミットされると、データ・ロックは解除され、コミットされたばかりのデータを、他のアプリケーションが使用できるようになる。ロールバック (*rollback*) も参照。

## [サ行]

**作業単位 (unit of work).** (1) アプリケーション・プロセス内のリカバリー可能な操作シーケンス。どの時点でも、アプリケーション・プロセスは、単一の操作単位であるが、アプリケーション・プロセスの寿命には、コミット操作またはロールバック操作の結果として多くの作業単位が関係する場合がある。(2) DRDA において、データベース・マネージャーが单一のエンティティーとして扱う一連の SQL コマンド群。データベース・マネージャーは、ある作業単位の間に行われたすべてのデータ変更が実行されたか、あるいはデータ変更が 1 つも行われなかつたかのどちらかを確認することによって、データの整合性を確保する。

**サブストリング (substring).** SUBSTR 関数で開始位置と長さを指定する、ストリングの一部。

**指示照会 (Prompted Query).** 1 組のダイアログ・パネルに対するユーザーの応答に従って作成される照会。

**システム・ログ (System Log (SYSLOG)).** オペレーターとの間での、ジョブ関連情報、操作上のデータ、通常でないオカレンスの記述、コマンド、およびオペレーターとの間のメッセージを保管できるデータ・セットまたはファイル。

**修飾子 (qualifier).** (1) QMF オブジェクトについて用いられるときは、オブジェクトの所有者またはロケーションを識別する名前の部分。 (2) TSO データ・セットについて用いられるときは、それ以外の名前の部分とピリオドで区切られている名前の部分。たとえば、‘TCK’、‘XYZ’、および ‘QUERY’ は、すべてデータ・セット名 ‘TCK.XYZ.QUERY’ の修飾子である。

**集約関数 (aggregation function).** 列中のデータを総計する関数の任意のグループ。書式パネル上で取扱コード AVERAGE、CALC、COUNT、FIRST、LAST、MAXIMUM、MINIMUM、STDEV、SUM、CSUM、PCT、CPCT、TPCT、TCPCT を使って要求される。

**集約変数 (aggregation variable).** FORM.BREAK、FORM.CALC、FORM.DETAIL、または FORM.FINAL パネルを使って報告書の中に入れられる集約関数。その値は、作成された報告書に、切れ目脚注、明細ブロック・テキスト、または最終テキストの一部として現れる。

**初期化プログラム (initialization program).** QMF プログラム・パラメーターを設定するプログラム。このプログラムは、呼び出し可能インターフェースの START コマンドの DSQSCMDn パラメーター (n は使用している各国語機能を識別する 1 文字) で指定される。デフォルト・プログラムは DSQSCMDE (QMF プログラム・パラメーター値を渡す英語のプログラム) である。各国語機能と 1 文字の ID のリストは、「*TSO/CICS 用 DB2 QMF インストールおよび管理の手引き*」を参照のこと。

**スカラー (scalar).** 列、リテラルの値、または他のスカラーを含む式の値。

**スカラー関数 (scalar function).** 1 つの値を別の値から作成し、関数名の後ろに括弧で囲んだ引数リストを付けた形で表す操作。

**図形データ表示管理プログラム (GDDM).** ピクチャーが、グラフィック・プリミティブに対応する機能ルーチンを通してプロセッサーにのっとり定義および表示できるようにするルーチンのグループ。

**スレッド (thread).** アプリケーションの接続の記述、その進行のトレース、リソース機能処理能力の提供、および DB2 リソースとサービスへのアクセス可能性の限度の決定を行なう DB2 (z/OS 版) の構造。ほとんどの DB2 (z/OS 版) 機能は、スレッド構造のもとで実行される。

**線形構文 (linear syntax).** あるプログラムまたはプロセッサーの 1 ステートメントに入力するか、または QMF コマンド行に入力できる QMF のコマンド構文。

**線形手順 (linear procedure).** REXX コメントで始まっていない プロセッサー。線形手順には、QMF コマンド、コメント、ブランク行、RUN コマンド、置換変数を入れることができる。ロジックを持つプロセッサー (procedure with logic) も参照。

**相関名 (correlation name).** SELECT 照会の FROM 文節で指定された表名の別名。列名と連結させると、列が属する表を識別できる。

## [夕行]

**ターゲット表 (target table).** 例示エレメントを使用して、列を組み合わせたり、行を組み合わせたり、報告書に定数値を組み込んだりする空の表。

**ダイアログ・パネル (dialog panel).** 指示照会基本パネルの一部を重ね書きし、照会の作成を援助するダイアログを拡張するパネル。

## 用語集

**大域変数 (global variable).** 1 回セットすると、1 つの QMF セッションの間使用できる変数。大域変数はプロセッサー、照会、または書式で使用できる。ランタイム変数 (*run-time variable*) と対比。

**タイム・スタンプ (timestamp).** 日付および時刻、必要に応じてマイクロ秒 (6 または 7 部分値)。

**置換変数 (substitution variable).** (1) 値が、大域変数またはランタイム変数のいずれかで指定される、プロセッサー内または照会内の変数。 (2) 値が、大域変数で指定される書式の変数。

**データベース管理システム (database management system (DBMS)).** データベースを定義、作成、操作、制御、管理、使用するためのシステム。データベース管理システムはまた、データの保全性を保護するためのトランザクション管理およびデータ管理機能をもつ。

**データベース管理者 (database administrator).** データベースの内容とデータベースへのアクセスを管理する担当者。

**データベース・サーバー (database server).** (1) DRDA において、アプリケーション・サーバーから受け取った要求のターゲット。

**データベース・マネージャー (database manager).** データベースを作成し維持管理するため、またデータベースをアクセスする必要のあるプログラムと通信するために使用されるプログラム。

**定位置パラメーター (positional parameter).** コマンド内の決まった位置に指定しなければならない QMF コマンドの 1 エレメント。

**デフォルト書式 (default form).** 照会が実行されるときに、QMF が作成する書式。保管されている書式が照会で実行されるときは、デフォルト書式は作成されない。

## [八行]

**バインド (bind).** DRDA で、アプリケーション・プログラム内の SQL ステートメントを、アプリケーション・サポート・プロトコル (およびデータベース・サポート・プロトコル) フロー上でデータベース管理システムに認識させるプロセス。バインドの際に、プリコンパイラまたはプリプロセッサーからの出力は、パッケージと呼ばれる制御構造に変換される。さらに、参照されたデータへのアクセス・パスが選択され、一部の許可検査が実行される。(DB2 (z/OS 版) では、出力がアプリケーション・プランとなることがある。)

**パッケージ (package).** アプリケーション・プログラムの SQL ステートメントがリレーションナル・データベース管理システムにバインドされる場合につくられる制御構造。データベース管理システムは、この制御構造を使用して、ステートメントの実行時に見出される SQL ステートメントを処理する。

**バッチ QMF セッション (batch QMF session).** バックグラウンドで実行する QMF セッション。指定された QMF プロセッサーが呼び出されたときに開始し、そのプロセッサーが終了したときに終了する。バックグラウンド QMF セッション時には、ユーザー対話もパネル表示対話も行えない。

**パネル (panel).** ウィンドウに表示するためにグループ分けした情報の特定の配列。パネルには、通知用のテキスト、入力フィールド、ユーザーが選択できるオプション、またはそれらの組み合わせを入れることができる。

**パラメーター (parameter).** QMF コマンドの 1 エレメント。この用語は、QMF 資料では、キーワード・パラメーター または 定位置パラメーター の総称として用いられる。

**バリエーション (variation).** FORM.DETAIL パネルに指定されて、条件に応じて、1 つの報告書または報告書の部分のフォーマットの設定に使用できる、データ・フォーマット設定定義。

**引数 (argument).** 独立変数。

**日付 / 時刻データ (date/time data).** データ・タイプが DATE、TIME、または TIMESTAMP である、表の列内のデータ。

**日付 / 時刻デフォルト・フォーマット (date/time default formats).** データベース・マネージャーのインストール・オプションで指定される日付 / 時刻フォーマット。そのフォーマットには、EUR、ISO、JIS、USA、または LOC (LOCAL) がある。

**表データ (tabular data).** 縦欄形式のデータ。データの内容および書式は、FORM.MAIN および FORM.COLUMNS で指定される。

**副照会 (subquery).** 別の照会 (主照会またはより高位の副照会) の WHERE または HAVING 文節の中に現れる完結した SQL 照会。

**プラン (plan).** 複数のプログラムの SQL ステートメントがバインド時に一緒に集められてプランが作られる、パッケージの 1 つの形式。

**プロファイル (profile).** ユーザーのセッションの特性情報が入っているオブジェクト。保管プロファイルとは、永続記憶域に保管されているプロファイルのことである。一時記憶域にあるプロファイルは、PROFILE という名前をもつ。ユーザーは、それぞれプロファイルを 1 つしかもつことができない。

**分散作業単位 (distributed unit of work).** 分散リレーショナル・データにアクセスする方式。单一の作業単位内で、ユーザーまたはアプリケーションが SQL ステートメントを複数のロケーションに実行依頼できる (ただし、SQL ステートメントと RDBMS は 1 対 1 に対応する)。

DB2 (z/OS 版) では、QMF がサポートする、システム指示アクセスと呼ばれる制限された形の分散作業単位サポートが採用されている。

**分散データ (distributed data).** ネットワーク内の複数のシステムに保管され、リモート・ユーザーおよびアプリケーション・プログラムで使用できるデータ。

**分散データベース (distributed database).** ユーザーからは論理的な全体として見ることができ、ローカルにアクセスできるデータベースであるが、実際は複数のロケーションにあるデータベースから構成されている。

**分散リレーショナル・データベース (distributed relational database).** すべてのデータがリレーショナル・モデルに従って保管されている分散データベースの 1 つ。

**分散リレーショナル・データベース体系™ (DRDA).** IBM とベンダーのリレーショナル・データベース・プロダクトで使用される分散リレーショナル・データベース処理の接続プロトコル。

**別名 (alias).** DB2 (z/OS 版) で、同じ、またはリモートの DB2 (z/OS 版) サブシステム中の表あるいは視点を参照する SQL ステートメント中で使用できる代替名。QMF で、ローカルまたはリモート DB2 (z/OS 版) サブシステムで保管されている QMF 表または視点にアクセスするために使用されるローカルに定義された名前。

## [マ行]

**無名列 (unnamed column).** 例示表に追加される空の列。無名列は、ターゲット表と同様に、列を組み合わせたり、行を組み合わせたり、あるいは定数値を報告書に組み込んだりするのに使用される。

**明細ブロック・テキスト (detail block text).** 個々のデータ行に対応している、報告書の本文中のテキスト。

**明細見出しテキスト (detail heading text).** 報告書の見出しのテキスト。見出しを印刷するかどうかは、FORM.DETAIL で指定する。

## [ヤ行]

**ヨーロッパ・フォーマット (EUR (European) format).** 以下のように日付 / 時刻値を表すフォーマット。

- 日付: dd.mm.yyyy
- 時刻: hh.mm.ss

## 用語集

**呼び出し CLIST または EXEC (invocation CLIST or EXEC).** QMF を開始するプログラム。

**呼び出し可能インターフェース (callable interface).** QMF サービスへのアクセスを可能にするプログラミング・インターフェース。アプリケーションは、QMF セッション外で実行している場合でも、これらのサービスにアクセスできる。コマンド・インターフェース (*command interface*) と対比。

## [ラ行]

**ランタイム変数 (run-time variable).** プロシージャーまたは照会が実行される時にユーザーがその値を指定するプロシージャーまたは照会の変数。ランタイム変数の値は、現行のプロシージャーまたは照会でしか使用できない。大域変数 (*global variable*) と対比。

**リテラル (literal).** プログラミング言語において、1つの値を直接表す字句単位；文字自体によって値が与えられている文字ストリング。

**リモート作業単位 (remote unit of work).** (1) アプリケーションがリレーショナル・データベースとは異なるシステム上にある、SQL 分散処理の形。単独アプリケーション・サーバーが单一の論理作業単位内ですべてのリモート作業単位要求にこたえる。(2) SQL ステートメントのリモートでの準備と実行を可能にする作業単位。

**リレーショナル・データベース管理システム (relational database management system (RDBMS)).** リレーショナル・データベースを定義、作成、操作、制御、管理、使用するためのシステム。

**例示照会 QBE (Query-By-Example).** 照会を図形的に作成するときに使用する言語。詳細については、DB2 QMF 使用の手引きを参照してください。

**列折り返し (column wrapping).** 1つの列が数行にまたがることを可能にする報告書内の書式設定の値。長さが列幅を超える値を列に入れると、しばしば使用される。

**連結 (concatenation).** 2番目のストリングを1番目のストリングに付加して、2つのストリングを1つに結合すること。

**ローカル DB2 (z/OS 版) (local DB2 for z/OS).** DB2 (z/OS 版) と併用する場合は、アプリケーション・リクエスターは QMF と同じ z/OS システムで実行される DB2 (z/OS 版) サブシステムの一部である。したがって、DB2 (z/OS 版) サブシステム全体 (データを含む) がアプリケーション・リクエスターに関連するが、SQL ステートメントは現行ロケーションで処理される。このサブシステムは、QMF プランがバインドされる個所である。

**ロールバック (rollback).** アプリケーションまたはユーザーが行った、コミットされていないデータベース変更を除去するプロセス。ロールバックが起こると、ロックは解除され、変更が加えられたリソースの状態は、前回コミット、ロールバック、または開始されたときの状態に戻される。コミット (*commit*) も参照。

**ロケーション (location).** 分散リレーショナル・データベース・システムの特定のリレーショナル・データベース管理システム。各 DB2 (z/OS 版) サブシステムは1つのロケーションと見なされる。

**ロジックを持つプロシージャー (procedure with logic).** REXX のコメントで始まる QMF プロシージャー。ロジックを持つプロシージャーでは、条件つきロジックの実行、計算、ストリングの作成、および TSO または CICS 環境へのコマンドの戻しが可能である。線形手順 (*linear procedure*) も参照。

## [数字]

**1 次 QMF セッション (primary QMF session).** QMF の外部で開始される対話式セッション。このセッションの中で、他のセッションを開始するときは、INTERACT コマンドを使用する。

**2 バイト文字 (double-byte character).** 2 個の文字バイトを必要とするエンティティー。

**2 バイト文字セット (double-byte character set (DBCS))。** 各文字が 2 バイトで表現される文字セット。日本語、中国語、韓国語など、256 個のコード・ポイントでは表現できないほど多くの記号をもつ言語には、2 バイト文字セットが必要である。各文字が 2 バイトを必要とするため、DBCS 文字のタイプ、表示、印刷には、DBCS をサポートするハードウェアとプログラムが必要となる。

**2 フェーズ・コミット (two-phase commit)。** 参加しているリレーショナル・データベース管理システムが 1 作業単位を矛盾なくコミットまたはロールバックできるようにするために分散作業単位で使用されるプロトコル。

**3 部分名 (three-part name)。** ロケーション名、所有者 ID、およびオブジェクト名から構成される表または視点の完全修飾名。アプリケーション・サーバーでサポートされる場合は、3 部分名を SQL ステートメントで使って、指定されたロケーションの指定された表または視点を検索または更新することができる。

## A

**ABENDx.** 異常終了問題のキーワード。

**APAR.** プログラム診断依頼書 (Authorized Program Analysis Report (APAR))。

## C

**CICS.** 顧客情報管理システム (Customer Information Control System)。

**CSECT.** 制御セクション (Control section)。

## D

**DB2 (Linux、UNIX、および Windows 版) (DB2 for Linux, UNIX, and Windows)。** IBM ファミリーの DRDA データベース製品で、UNIX および Intel® プラットフォーム上で稼働する。

**DBCS.** 2 バイト文字 セット (double-byte character set)。

**DBMS.** データベース管理システム (Database management System (DBMS))。

## E

**EBCDIC.** 拡張 2 進化 10 進コード (Extended Binary-Coded Decimal Interchange Code)。

## H

**HTML.** ハイパーテキスト・マークアップ言語 (Hypertext Markup Language)。インターネットで表示される文書用の標準化されたマークアップ言語。

## I

**ICU.** 対話式図表ユーティリティー (Interactive Chart Utility)。

**INCORROUT.** 正しくない出力のキーワード。

**ISO (国際標準化機構) フォーマット (International Standards organization format)。** 以下のように日付 / 時刻値を表すフォーマット。

- 日付: yyyy-mm-dd
- 時刻: hh.mm.ss

## 用語集

**ISPF.** 対話式システム生産性機能 (Interactive System Productivity Facility)。

**IXF.** 統合交換フォーマット (Integration Exchange Format)。各種のソフトウェア・プロダクト間で表データを転送するためのプロトコル。

## J

**JCL.** OS/390 および z/OS 用のジョブ制御言語。

**JIS (日本工業規格) フォーマット (Japanese Industrial Standard format).** 以下のように日付 / 時刻値を表すフォーマット。

- 日付: yyyy-mm-dd
- 時刻: hh:mm:ss

## M

**MSGx.** メッセージ問題のキーワード。

## N

**NLF.** 各国語機能 (National Language Feature)。英語以外の言語を 1 つ選択できるようにする、 QMF とともに使用できるいくつかのオプション・フィーチャーの 1 つ。

**NLS.** 各国語サポート (National Language Support)。

## P

**PERFM.** パフォーマンス問題のキーワード。

**PSW.** プログラム状況ワード (Program status word)。

**PTF.** プログラム一時修正 (Program temporary fix)。

## Q

**QMF 管理権限 (QMF administrative authority).** 最低でも、Q.PROFILES コントロール表に対する挿入または削除の権限。

**QMF 管理者 (QMF administrator).** QMF 管理権限を持つ QMF ユーザー。

**QMF コマンド (QMF command).** QMF 言語の一部である任意のコマンドを指す。サイトの定義コマンドを含まない。

**QMF セッション (QMF session).** ユーザーが QMF を呼び出した時点から、 EXIT コマンドを出すまでの、ユーザーと QMF 間で行なうすべての対話。

## R

**RDBMS.** リレーションナル・データベース管理システム (Relational database management system)。

**REXX.** 再構造化拡張実行プログラム (Restructured extended executor)。QMF によって使用され、また、QMF と一緒に使用するために作成できるアプリケーションの中で使用されるプロシージャー型言語。

## S

**SBCS.** 1 バイト文字セット (single-byte character set)。

**SNAP ダンプ (SNAP dump).** 異常終了時に QMF が生成する 1 つまたは複数の記憶域の内容の動的ダンプ。

**SQL.** 構造化照会言語 (*Structured Query Language*) を参照。

**SQLCA.** 構造化照会言語連絡域 (Structured Query Language Communication Area)。

## T

**TSO.** タイム・シェアリング・オプション (Time Sharing Option)。

## W

**WAIT.** 無限待ち状態問題のキーワード。

## Z

**z/OS.** IBM eServer<sup>TM</sup> 製品ラインのオペレーティング・システムで、64 ビット実ストレージを使用する。

## 用語集

# 索引

日本語, 数字, 英字, 特殊文字の順に配列されています。なお, 潤音と半潤音は清音と同等に扱われています。

## [ア行]

アクセシビリティ xii  
アスタリスク (\*)  
  式で 184  
値、計算された 161, 162, 184  
  GROUP BY 161, 162  
  WHERE 文節 184  
引用符  
  LIKE SQL キーワード 165

## [カ行]

改ページ  
  最終テキストの 230  
  明細ブロック・テキストの 226  
下線 (\_)  
  B(\_B) 269  
  LIKE SQL キーワード 165  
漢字データ  
  編集コード 264  
  LIKE SQL キーワード 165  
キーボード・ショートカット xii  
キーワード、SQL  
  移入元 173  
  ALL 144  
  ALTER TABLE 145, 158, 172  
  AND 145  
  ANY 146  
  AS 146  
  AVG 147  
  BETWEEN 168  
  COUNT(DISTINCT) 156  
  CREATE 154  
  CREATE TABLE 152  
  CREATE VIEW 154  
  DELETE 158, 172  
  DELETE FROM 155  
  DISTINCT 156  
  DROP 157  
  GRANT 158  
  GROUP BY 159  
  HAVING 161  
  IN 153, 163, 168  
  INSERT 158, 172

キーワード、SQL (続き)  
  INSERT INTO 163, 164  
  IS 165, 168  
  LIKE 165, 168  
  MAX 167  
  MIN 167  
  NOT 167  
  NOT NULL 154  
  NULL 168  
  OR 169  
  ORDER BY 170, 172, 173  
  REVOKE 172  
  SELECT 158, 172, 173  
  SET 180  
  SOME 175  
  SUM 176  
  TABLE 152, 157  
  UNION 177  
  UPDATE 158, 172, 180  
  VALUES 163, 164  
  VIEW 154, 157  
  WHERE 180, 181  
  WITH GRANT OPTION SQL キーワード 158  
  WITH REVOKE OPTION SQL キーワード 172  
記述子域 (DA) 268  
行 145  
  折り返し  
    制御 235  
    FORM.OPTIONS パネルの幅 235  
  更新 180  
  更新許可  
    取り消し 172  
    付与 158  
  削除 155  
  条件による選択  
    AND 169  
    NULL 168  
    OR 169  
    SELECT 173  
    WHERE 181  
  挿入 163, 164  
  重複を除去 156  
  入力域  
    FORM.DETAIL パネル 224  
    FORM.PAGE パネル 241  
  ヌル 168  
  配列 170  
  許可  
    更新 180

許可 (続き)  
削除 155  
視点の作成 154  
選択 173  
挿入 163  
取り消し 172  
表の行を更新するため 158, 172  
表を作成 152  
表を使用するため 158  
付与 158  
変更 145  
区切り線 240  
区分データ・セット  
参照： データ・セット、エクスポートのための定義  
組み込み SQL 関数  
AVG 147  
COUNT(DISTINCT) 156  
MAX 167  
MIN 167  
SUM 176  
計算 250  
計算値 161  
グループ 159  
AVG 147  
COUNT(DISTINCT) 156  
GROUP BY 161, 162  
MAX 167  
MIN 167  
SUM 176  
WHERE 文節 184  
結合、表の 177, 180  
UNION を使用 177, 180  
更新 180  
行 180  
構造化照会言語  
参照： SQL  
後続ブランク、保持 269

## [サ行]

サーバー  
参照： データベース  
最終テキスト  
要約  
FORM.FINAL パネル 230  
算術  
演算子 184  
式 184  
サンプル表 285  
式  
記号と演算 184  
算術 184

式 (続き)  
条件で 168  
書式での使用 250  
評価 184  
REXX プログラムでいつ評価されるか 250  
視点  
作成 154  
除去 158  
制約 155  
視点でデータを保護 154  
視点によるデータ保護 154  
修飾名  
表の 158  
集約  
取扱コード 254  
照会  
許可を与える 158  
許可を取り消し 172  
計算値 159, 184  
条件 168, 181  
選択 173  
一部の文字ストリングで 165  
条件 181  
等号と不等号で 183  
特定の行 181  
特定列 174  
否定条件で 167  
複数条件で 145, 169  
リストの値で 163  
全列 173  
重複行を除去 156  
データ定義 152  
データ入力  
行の更新 180  
行の挿入 163  
での式 184  
副照会  
ALL SQL キーワード付き 144  
ANY SQL キーワード 146  
SOME SQL キーワード 175  
報告書における行の配列 170, 172  
DELETE FROM 155  
SQL 143  
照会の定数 174  
照会報告書作成プログラム  
参照： QMF  
消去  
別名 157  
条件  
書き込み 181  
式での 168  
等号での 183

条件 (続き)  
 否定 167  
 複数の 145, 169  
   AND 145  
   OR 169  
   不等号での 183  
   リストの値 163  
 乗算演算子 (\*) 185  
 乗算記号 (\*)  
   式で 184  
   乗算演算子 185  
 書式 (form)  
   パネル  
     入力域 191  
     変更 191  
   GROUP 取扱コード 160  
 身体障害を持つユーザーのための機能 xii  
 数値  
   データ  
     式で 185  
     編集コード 264  
   定数 174  
 スカラー関数 187, 188, 189  
   ストリング 189  
   日付 / 時刻 187  
   変換 188  
 スクリーン・リーダー (読み上げソフトウェア) xii  
 スtring  
   機能 189  
 ストレージ  
   エクスポートでの指定 55, 306, 308  
   XML データのサイズ制限 31  
 図表  
   印刷 279  
     GDDM 279  
     データ・タイプの制約 194  
   入力域 195  
 スラッシュ (/)  
   式で 184  
   除算演算子 185  
 正符号 (+)  
   演算子 185  
   式で 184  
 線形手順 278  
 先行 / 後続ブランクの保持 (\_B)  
   書式での 269  
   変数での 269  
 先行ブランク、保持 269  
 選択  
   条件  
     一部の文字ストリングでの 165  
     等号と不等号により 183

選択 (続き)  
 条件 (続き)  
   否定 167  
   複数の 145, 169  
   リストの値 163  
   全列 173  
   特定の行 181  
   特定列 174  
   複数の表から最大数 175  
 選択記号  
   LIKE SQL キーワード 165  
 ソート・シーケンス、ORDER BY 170  
 挿入  
   行 163  
 揿入、複数行、設定 139  

## [夕行]

 大域変数  
   書式での 269  
   のリスト 293  
   RUW を通じて使用される QMF 294  
 大小文字の混合  
   切れ目後書きの 205  
 タイム・スタンプ  
   編集コード 267  
 大/小文字混合パスワード 19, 294  
 調整  
   図表 242  
   ページ・ヘッダー 242  
   報告書 242  
   重複行の除去 156  
   重複行を除去 156  
 データ  
   削除 155  
   定義 152  
   入力  
     行の更新 180  
     行の削除 155  
     行の挿入 163, 164  
   保護 154  
 データベース  
   サポートされるバージョン 20  
   名前 158  
   複数行フェッチ/挿入、設定 139  
   リモート作業単位の使用 282  
 データ・セット、エクスポートのための定義 55, 308  
 データ・タイプ  
   式で 185  
   デフォルトの幅 214  
   表示できない 268  
   CREATE TABLE 152

定義  
表 152  
ディレクトリー・ブロック、エクスポートでの指定 55, 306  
等号 183  
トグル・スイッチ、管理プログラム出口 144  
取扱コード 254

## [ナ行]

名前  
修飾された 158  
ヌル  
値  
印刷と表示 169  
出力での表し方 169  
ALL による副照会から 144, 146  
ALTER TABLE で追加された列で 145  
GROUP BY SQL キーワード付き 160  
INSERT SQL キーワードで 163  
INSERT で暗黙 163  
NOT NULL で保護 154  
SOME による副照会から 175  
条件で 168  
定義 168

## [ハ行]

パーセント記号 (%)  
LIKE SQL キーワード 165, 166  
配列  
報告書における行 170, 172  
判断、行が存在するかどうかの 158  
否定条件 NOT SQL キーワード 167  
表  
行の削除 155  
行の挿入 163, 164  
作成 152  
サンプル 285  
Q.APPLICANT 285  
Q.INTERVIEW 286  
Q.ORG 286  
Q.PARTS 287  
Q.PRODUCTS 287  
Q.PROJECT 288  
Q.SALES 289  
Q.STAFF 290  
Q.SUPPLIER 291  
使用許可 158, 172  
除去 157  
ヌル 168  
複数の 180

表 (続き)  
別名 157  
リソース制御、管理プログラム出口 144  
表エディターのデータ・タイプの制約 37, 281  
表の定義 152  
フェッチ/挿入、複数行 139  
不完全データプロンプト・パネル 275  
副照会

ALL SQL キーワード付き 144  
ANY SQL キーワード 146  
SOME SQL キーワード 175  
複数行フェッチ/挿入、設定 139  
複数の  
条件 145, 169  
表 180  
不整合、書式とデータの 247  
不等号 183  
WHERE 文節で 183  
不等号 (<>) 146, 183  
不等号の代替記号 (=)  
演算子 146  
検索条件 183

負符号 (-)  
演算子 185  
式で 184

ブランク行  
後書きでの 243  
ヘッダー 241  
FORM.PAGE パネル 241, 243

ブランク取扱コード 261

プロシージャー  
線形 277  
ロジック 277  
REXX 277

分散作業単位  
参照： DUW (分散作業単位)

ページ  
後書き 245  
ヘッダー 243  
変数 243

別名  
除去 157  
変更  
報告書式 191  
編集コード 261  
記述された 268  
ユーザ一定義 210, 268

変数  
グローバル 293  
書式 (form) 269  
書式での 269

## 報告書

- 表示できないデータ・タイプ 268
- HTML
  - エクスポート 52

## [マ行]

マージ、表の 177

明細

- ヘッダー・テキスト
- FORM.DETAIL パネル 224
- メタデータ編集コード 268
- 文字
  - データ
    - 編集コード 264
    - LIKE SQL キーワード 165
  - 定数 174

## [ヤ行]

ユーザー定義編集コード 210, 268

予約語 143

## [ラ行]

リモート

- 表
  - リモートへのアクセス 282
  - 3 部分名 282
- リモート作業単位
  - 現行ロケーション 282
  - 使用 282
  - データベースへの接続 282
  - SQL ステートメント 282
- リモート・データ
  - アクセス
    - 分散作業単位 281
    - リモート作業単位 281

列

- 機能
  - AVG 147
  - COUNT(DISTINCT) 156
  - MAX 167
  - MIN 167
  - SUM 176
- 選択
  - 最大行数 174
  - すべて 173
  - 複数の表からの 180
- 置換変数 207
- ヘッダー
  - 切り捨て 215

## 列 (続き)

- ヘッダー (続き)
- グループ化したときの関数名 238
- 図表の 215
- 入力域 196
- FORM.MAIN パネル 196
- 2 つの表から 180
- CREATE TABLE での定義 152
- 論理否定 ( $\neg$ )
  - 演算子 146
  - 検索条件 183

## [数字]

- 1 次スペース割り振り、エクスポートでの 55, 306
- 10 進浮動小数点データ・タイプ
  - 参照: DECFLOAT データ・タイプ
- 2 次スペース割り振り、エクスポートでの 55, 307
- 3 部構成の名前、失敗 6
- 3 部構成の名前を指定したコマンドの失敗 6

## A

- ADD コマンド 144
- ALIGN 入力域
  - FORM.PAGE パネル 242
- ALL SQL キーワード 144
- ALL キーワード
  - SQL 144
- ALTER TABLE SQL キーワード
  - 許可を与える 158
  - 許可を取り消し 172
- ALTER ステートメント
  - TABLE キーワード
    - 許可を与える 158
    - 許可を取り消し 172
  - AND SQL キーワード 145
  - ANY SQL キーワード 146
  - AS キーワード 146
  - AVG キーワード 147

## B

- B 編集コード 220, 264
- BETWEEN SQL キーワード 168
- BETWEEN キーワード
  - 例 168
- BIGINT データ、書式での幅 214
- BINARY データ
  - 制約
    - 図表の表示 31
    - 表エディター 37

BINARY データ (続き)  
  デフォルトの幅 215  
  編集コード 220, 264  
BLOB データ・タイプの制約 210, 268  
BW 編集コード 220, 264  
B、を頭にもつ (\_B) 269

## C

CHAR  
  スカラー関数 187  
CLOB データ・タイプの制約 210, 268  
CONNECT コマンドと大/小文字混合パスワード 294  
CREATE SQL キーワード 152, 154  
CREATE ステートメント、SQL  
  TABLE 152  
  VIEW 154

## D

D 編集コード 264  
DATE  
  スカラー関数 187  
  データ・タイプ、編集コード 265  
  変数 243  
DAY スカラー関数 187  
DAYS スカラー関数 187  
DBCLOB データ・タイプの制約 210, 268  
DBCS (2 バイト文字セット)  
  同義語 152  
DBCS (2 バイト文字セット) 同義語 152  
DECFLOAT データ・タイプ  
  制約  
    インポート 68, 73  
    エクスポート 49, 58  
    図表作成 31, 194  
    表エディター 37, 281  
    ユーザー編集コード 210  
    SAVE DATA コマンド 122  
  取扱コード 261  
  編集コード 268  
DECIMAL  
  SQL スカラー関数 188  
DELETE  
  SQL キーワード 155  
DIGITS スカラー関数 188  
DISTINCT SQL キーワード 156  
DROP SQL キーワード 157  
DSQCXPR プログラム 249  
DSQEC\_NLFCMD\_LANG 変数 307  
DUW (分散作業単位) 6

## E

EDIT  
  入力域  
    FORM.COLUMNS パネル 216  
EXISTS SQL キーワード 158  
Extensible Markup Language データ・タイプ  
  参照： XML データ・タイプ

## F

FLOAT  
  SQL スカラー関数 188  
FROM SQL キーワード 173

## G

G 編集コード 264  
GDDM (図形データ表示管理プログラム)  
  QMF オブジェクトの印刷 279  
GRANT SQL キーワード 158  
GROUP BY SQL キーワード 159  
GW 編集コード 264

## H

HAVING SQL キーワード 161  
HEX スカラー関数 188  
HOUR スカラー関数 187  
HTML 報告書のエクスポート 52, 56

## I

I 編集コード 264  
IN SQL キーワード  
  リストの値について 163  
  CREATE TABLE 153  
  NOT で使用される 168  
IN キーワード  
  リストの値について 163  
  CREATE TABLE 153  
  NOT で使用される 168  
INSERT INTO SQL キーワード 163  
INSERT SQL キーワード 163  
INTEGER  
  SQL スカラー関数 188  
IS SQL キーワード 168

## J

J 編集コード 264

## K

K 編集コード 264

## L

L 編集コード 264

LENGTH

  スカラー関数 189

LIKE SQL キーワード 165, 166, 168

LOB データ・タイプ 268

LONG VARCHAR データ

  取扱コード 261

LONG VARGRAPHIC データ

  取扱コード 261

## M

M 編集コード 268

MAX SQL キーワード 167

MICROSECOND スカラー関数 187

MIN SQL キーワード 167

MINUTE スカラー関数 187

MONTH スカラー関数 187

## N

NOT NULL SQL キーワード

  表定義で 154

  ALTER TABLE で許可されない 145

NOT SQL キーワード 168

NULL SQL キーワード 168

NULL 置換 186

  入力域

  FORM.CALC パネル 208

## O

OMIT 取扱コード 261

OR

  SQL キーワード 169

ORDER BY SQL キーワード 170, 172, 173

## P

P 編集コード 264

PDS および PDSE データ・セット

  エクスポート・ストレージの定義 306, 307

  へのエクスポート 55, 57

  QMF へのタイプの定義 308

PS データ・セット、エクスポートのための定義 306,  
307

## Q

QMF

  一時記憶域

  内容の置き換え 274

Q.APPLICANT サンプル表 285

Q.INTERVIEW サンプル表 286

Q.ORG サンプル表 286

Q.PARTS サンプル表 287

Q.PRODUCTS サンプル表 287

Q.PROJECT サンプル表 288

Q.SALES サンプル表 289

Q.STAFF サンプル表 290

Q.SUPPLIER サンプル表 291

## R

RACF と大/小文字混合パスワード 294

REVOKE SQL キーワード 172

REXX

  ロジックを持つプロシージャー 277

REXX @IF 関数 185

## S

SCOPE リソース・オプション 144

SECOND スカラー関数 187

SET SQL キーワード 180

SOME SQL キーワード 175

SQL

  照会

    保管 143

    ステートメント 143

    予約語のリスト 143

    SQL キーワード 143

  SQL キーワード

    移入元 173

    ALL 144

    ALTER TABLE 145, 158, 172

    AND 145

    ANY 146

    AS 146

    AVG 147

    BETWEEN 168

    COUNT(DISTINCT) 156

    CREATE 154

    CREATE TABLE 152

    CREATE VIEW 154

    DELETE 158, 172

    DELETE FROM 155

    DISTINCT 156

    DROP 157

## SQL キーワード (続き)

GRANT 158  
GROUP BY 159  
HAVING 161  
IN 153, 163, 168  
INSERT 158, 172  
INSERT INTO 163, 164  
IS 165, 168  
LIKE 165, 168  
MAX 167  
MIN 167  
NOT 167  
NOT NULL 154  
NULL 168  
OR 169  
ORDER BY 170, 172, 173  
REVOKE 172  
SELECT 158, 172, 173  
SET 180  
SOME 175  
SUM 176  
TABLE 152, 157  
UNION 177  
UPDATE 158, 172, 180  
VALUES 163, 164  
VIEW 154, 157  
WHERE 180, 181  
WITH REVOKE OPTION SQL キーワード 172  
SUBSTR スカラー関数 189  
SUM  
  SQL キーワード 176

## T

TABLE  
  SQL キーワード 157  
TD 編集コード 265  
TIME  
  スカラー関数 187  
  データ・タイプ、編集コード 267  
  変数 243  
TIMESTAMP  
  スカラー関数 187  
TSI 編集コード 267  
TT- タイプの編集コード 267

## U

UNION SQL キーワード 177  
  複数列のマージ 177  
UNIX パス名  
  インポート元 69, 71, 73

## UNIX パス名 (続き)

  へのエクスポート 52, 53, 58  
UPDATE SQL キーワード  
  行の変更 180  
  許可を与える 158  
  許可を取り消し 172  
Uxxxx 編集コード 268

## V

VALUE スカラー関数 189  
VALUES SQL キーワード 163, 164  
VARBINARY データ・タイプ  
  制約  
    図表作成 31, 194  
    表エディター 37, 281  
    デフォルトの幅 215  
    編集コード 220, 268  
VARGRAPHIC  
  SQL スカラー関数 188  
VIEW SQL キーワード 154, 157  
Vxxxx 編集コード 268

## W

WHERE SQL キーワード 180  
WITH GRANT OPTION SQL キーワード 158  
WITH REVOKE OPTION SQL キーワード 172

## X

X 編集コード 220, 264  
XML データ・タイプ  
  インポート  
    CICS 67  
    TSO 70, 73  
  エクスポート  
    CICS 47, 49  
    TSO 56  
  制約  
    図表作成 31, 194  
    表エディター 37, 281  
    ユーザー編集コード 210  
  取扱コード 261  
  編集コード 268  
XW 編集コード 220, 264

## Y

YEAR スカラー関数 187

## [特殊文字]

&COUNT 変数  
最終テキスト 233  
&ROW 変数  
最終テキスト 233  
@IF 250







プログラム番号: 5635-DB2

Printed in Japan

SD88-7608-00



日本アイ・ビー・エム株式会社  
〒106-8711 東京都港区六本木3-2-12