



System i

プログラミング

WebSphere Development Studio for System i コマンド

バージョン 6 リリース 1







System i

プログラミング

WebSphere Development Studio for System i コマンド

バージョン 6 リリース 1

#### ご注意

本書および本書で紹介する製品をご使用になる前に、411 ページの『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 WebSphere Development Studio for System i (プロダクト番号 5761-WDS) のバージョン 6、リリース 1、モディフィケーション 0 に適用されます。また、改訂版で断りがない限り、それ以降のすべてのリリースおよびモディフィケーションに適用されます。このバージョンは、すべての RISC モデルで稼働するとは限りません。また CISC モデルでは稼働しません。

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： System i  
Programming  
WebSphere Development Studio for System i commands  
Version 6 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 2008.2

© Copyright International Business Machines Corporation 1998, 2008. All rights reserved.

---

## 目次

PDM省略時値の変更 (CHGPDMDFT) . . . . .	1	COBOLデバッグ終了 (ENDCBLDBG) . . . . .	313
物理ファイル・メンバーの比較 (CMPPFM) . . . . .	9	ISDBの終了 (ENDISDB) . . . . .	315
DBCSマスター分類テーブルのコピー (CPYIGCSRT) . . . . .	17	PDMを使用したストリングの検索 (FNDSTRPDM) . . . . .	317
結合Cプログラム作成 (CRTBNDC) . . . . .	19	C/C++ソースの生成 (GENCSRC) . . . . .	325
バインドCOBOL PGMの作成 (CRTBNDCBL) . . . . .	45	用紙記述組み合わせ (MRGFORMD) . . . . .	331
結合C++プログラム作成 (CRTBNDCPP) . . . . .	73	ソースの組み合わせ (MRGSRC) . . . . .	339
バインドRPG PGMの作成 (CRTBNDRPG) . . . . .	99	CODEバッチ・ジョブの投入 (SBMCODEJOB) . . . . .	343
COBOLモジュールの作成 (CRTCBLMOD) . . . . .	119	拡張印刷機能 (STRAPF) . . . . .	347
COBOLプログラム作成 (CRTCBLPGM) . . . . .	145	COBOLデバッグ開始 (STRCBLDBG) . . . . .	349
Cモジュール作成 (CRTCMOD) . . . . .	163	CGU開始 (STRCGU) . . . . .	351
C++モジュール作成 (CRTCPPMOD) . . . . .	191	CODEの開始 (STRCODE) . . . . .	353
DFU表示装置ファイルの作成 (CRTDFUDSPF) . . . . .	219	CODEコマンドの開始 (STRCODECMD) . . . . .	357
RPGモジュールの作成 (CRTRPGMOD) . . . . .	223	ISDBの開始 (STRISDB) . . . . .	359
RPG/400プログラムの作成 (CRTRPGPGM) . . . . .	243	PDM開始 (STRPDM) . . . . .	365
AUTO REPORT RPG プログラム作成 (CRTRPTPGM) . . . . .	255	報告書設計ユーティリティ開始 (STRRLU) . . . . .	367
S/36 COBOLプログラム作成 (CRTS36CBL) . . . . .	267	RSEサーバーの開始 (STRRSESVR) . . . . .	371
RPG IIプログラムの作成 (CRTS36RPG) . . . . .	279	SDAの開始 (STRSDA) . . . . .	375
コンソール表示装置ファイル作成 (CRTS36RPGR) . . . . .	289	SEU開始 (STRSEU) . . . . .	381
S/36 RPG II報告書簡易作成 (CRTS36RPT) . . . . .	295	PDM使用のライブラリーの処理 (WRKLIBPDM) . . . . .	387
RPGソースの変換 (CVTRPGSRC) . . . . .	307	PDM使用のメンバーの処理 (WRKMBRPDM) . . . . .	391
		PDM使用のオブジェクトの処理 (WRKOBJPDM) . . . . .	399
		付録. 特記事項 . . . . .	411



## PDM省略時値の変更 (CHGPDMDFT)

実行可能場所: すべての環境 (\*ALL)

スレッド・セーフ: いいえ

パラメーター

例

エラー・メッセージ

PDM省略時値の変更 (CHGPDMDFT)コマンドによって特定ユーザーのためのプログラム開発管理機能 (PDM)の省略時の値を割り当てることができます。PDMの使用時にF18を押すと表示される省略時の値変更パネルに類似します。ただし、このコマンドは他のユーザーのPDM省略時の値の変更に使用でき、バッチで実行できます。

### 制約事項

- PDMの省略時の値が変更されたユーザーのユーザー・プロファイルにはオブジェクト管理(\*OBJMGT)および使用 (\*USE)権限が必要です。

上

## パラメーター

キーワード	記述	選択項目	ノート
USER	ユーザー	単純名	必須, 定位置 1
OBJLIB	オブジェクト・ライブラリー	名前, *SAME, *SRCLIB, *CURLIB	オプション
RPLOBJ	オブジェクトの置き換え	*SAME, *NO, *YES	オプション
CRTBCH	バッチで作成／コンパイル	*SAME, *YES, *NO	オプション
RUNBCH	バッチで実行	*SAME, *NO, *YES	オプション
SAVRSTOPT	保管／復元オプション	*SAME, *SINGLE, *ALL	オプション
JOB	ジョブ記述	単一値: *SAME, *USRPRF その他の値: 修飾オブジェクト名	オプション
	修飾子 1: ジョブ記述	名前	
	修飾子 2: ライブラリー	名前, *LIBL, *CURLIB	
CHGTYPTXT	タイプおよびテキストの変更	*SAME, *YES, *NO	オプション
FILE	オプション・ファイル	単一値: *SAME その他の値: 修飾オブジェクト名	オプション
	修飾子 1: オプション・ファイル	名前	
	修飾子 2: ライブラリー	名前, *LIBL, *CURLIB	
MBR	オプション・ファイル・メンバー	名前, *SAME	オプション
FULLSCN	全画面モード	*SAME, *NO, *YES	オプション
LOGCOM	オプション・コマンドのログ	*SAME, *NO, *YES	オプション
EXITENT	実行キーでリストを終了	*SAME, *NO, *YES	オプション
DSPINMSG	通知メッセージの表示	*SAME, *YES, *NO	オプション

上

---

## ユーザー (USER)

PDM省略時の値を変更するユーザーを指定します。

これは必須パラメーターです。

**単純名** 省略時の値を変更するユーザーの名前を指定します。指定した名前のユーザー・プロファイルがシステム上に存在する必要があります。

上

---

## オブジェクト・ライブラリー (OBJLIB)

ソース・ファイル・メンバーのコンパイル(PDMを使用したメンバー処理画面から)で作成されたオブジェクトを保管するライブラリーを指定します。

### \*SAME

このパラメーターが以前に設定されていた場合、値は変わりません。そうでない場合には\*SRCLIBが使用されます。

### \*SRCLIB

ソース・メンバーを含むライブラリーをオブジェクト・ライブラリーとして使用する。

### \*CURLIB

現行ライブラリーをオブジェクト・ライブラリーとして使用する。現行ライブラリーが定義されていない場合には、QGPLが使用されます。

**名前** ソース・ファイル・メンバーのコンパイル後のオブジェクトを保管するライブラリー名を指定します。

上

---

## オブジェクトの置き換え (RPLOBJ)

既存オブジェクトが、メンバーのコンパイルまたはモジュール作成時に新規オブジェクトに置き換えられるかどうかを指定します。

### \*SAME

このパラメーターが以前に設定されていた場合、値は変わりません。そうでない場合には\*NOが使用されます。

**\*NO** 既存のオブジェクトはメンバーのコンパイル開始またはモジュール作成の前には削除されません。オブジェクトが存在している場合には、メンバー・コンパイル確認画面が表示されます。

**\*YES** 既存のオブジェクトはメンバーのコンパイル開始またはモジュール作成の前には削除されます。コンパイルが正常に実行されなかった場合には、オブジェクトは復元されません。

上

---

## バッチで作成／コンパイル (CRTBCH)

メンバーのコンパイルまたはモジュールの作成時にジョブをバッチに投入するかどうかを選択します。



#### **\*SAME**

このパラメーターが以前に設定されていた場合、値は変わりません。そうでない場合には\*YESが使用されます。

**\*YES** バッチでメンバーのコンパイルまたはモジュールの作成を行います。

**\*NO** 対話式でメンバーのコンパイルまたはモジュールの作成を行います。

上

---

## **バッチで実行 (RUNBCH)**

オブジェクトの実行時にジョブをバッチに投入するかどうかを指定します。

#### **\*SAME**

このパラメーターが以前に設定されていた場合、値は変わりません。そうでない場合には\*NOが使用されます。

**\*NO** オブジェクトはバッチで実行できません。

**\*YES** オブジェクトはバッチで実行します。

上

---

## **保管／復元オプション (SAVRSTOPT)**

オブジェクトおよびメンバーを保管または復元するのに個別に行なうかそれとも1つのコマンドで行なうかを選択します。

#### **\*SAME**

このパラメーターが以前に設定されていた場合、値は変わりません。そうでない場合には\*SINGLEが使用されます。

#### **\*SINGLE**

選択したオブジェクトまたはメンバーを個別に（各オブジェクトまたはメンバーごとに別々のコマンドを使用する）保管または復元します。

**\*ALL** 選択したオブジェクトまたはメンバーを1つのコマンドで同時にすべて保管または復元します。

上

---

## **ジョブ記述 (JOBID)**

ジョブをバッチ・モードで投入するためのジョブ記述の名前を指定します。

#### **単一値**

#### **\*SAME**

このパラメーターが以前に設定されていた場合、値は変わりません。そうでない場合には\*LIBL/QBATCHが使用されます。

#### **\*USRPRF**

**ユーザー (USER)**パラメーターに指定されたユーザーのユーザー・プロファイルに定義されたジョブ記述が使用されます。

### 修飾子1: ジョブ記述

**名前** 使用するジョブ記述の名前を指定します。

### 修飾子2: ライブラリー

**\*LIBL** ジョブ・ライブラリー・リストにあるすべてのライブラリーから指定ジョブ記述を検索します。

#### **\*CURLIB**

指定ジョブ記述のためにジョブの現行ライブラリーを検索します。現行ライブラリーが定義されていない場合には、現行ライブラリーにQGPLが使用されます。

**名前** 指定ジョブ記述を検索するために使用するライブラリー名を指定します。

上

---

## タイプおよびテキストの変更 (CHGTYPTXT)

PDMを使用したメンバー処理画面で**タイプおよびテキスト**プロンプトの上に重ねて入力することによってこれらのプロンプトを変更することができるかどうかを指定します。

#### **\*SAME**

このパラメーターが以前に設定されていた場合、値は変わりません。そうでない場合には\*YESが使用されます。

**\*YES** PDMを使用したメンバー処理画面で**タイプおよびテキスト**プロンプトの上に重ねて入力することによってこれらのプロンプトを変更することができます。

**\*NO** PDMを使用したメンバー処理画面で**タイプおよびテキスト**プロンプトの上に重ねて入力することによってこれらのプロンプトを変更することはできません。

上

---

## オプション・ファイル (FILE)

ユーザー定義のオプションつきメンバーを持つファイルを指定します。このファイルのユーザー定義オプションは活動ユーザー定義オプションです。ユーザー定義オプション・ファイルは特別の形式をもっています。この形式およびユーザー定義オプション・ファイルのコピー方法の詳細については、「プログラム開発管理機能 使用者の手引きと参照」を参照してください。

### 単一値

#### **\*SAME**

このパラメーターが以前に設定されていた場合、値は変わりません。そうでない場合にはFILE(\*LIBL/QAUOOPT)が使用されます。

### 修飾子1: オプション・ファイル

**名前** アクティブなユーザー定義のオプション・ファイルの名前を指定します。

### 修飾子2: ライブラリー

**\*LIBL** ジョブ・ライブラリー・リストにあるすべてのライブラリーからアクティブなユーザー定義オプション・ファイルを検索します。

### **\*CURLIB**

ジョブの現行ライブラリーからアクティブなユーザー定義オプション・ファイルを検索します。現行ライブラリーが定義されていない場合には、現行ライブラリーにQGPLが使用されます。

**名前** アクティブなユーザー定義オプション・ファイルを検索するために使用するライブラリー名を指定します。

上

---

## **オプション・ファイル・メンバー (MBR)**

ユーザー定義オプションを含むメンバーを指定します。このメンバーに入っているユーザー定義オプションは活動ユーザー定義オプションです。

### **\*SAME**

このパラメーターが以前に設定されていた場合、値は変わりません。そうでない場合にはQAUOOPTが使用されます。

**名前** ユーザー定義オプションを含むメンバーの名前を指定します。

上

---

## **全画面モード (FULLSCN)**

処理画面にオプションおよび機能キーを表示するかどうかを指定します。このオプションはユーザー定義オプションの処理画面には適用されません。

### **\*SAME**

このパラメーターが以前に設定されていた場合、値は変わりません。そうでない場合には\*NOが使用されます。

**\*NO** 処理画面にオプションおよび機能キーを表示します。

**\*YES** 処理画面にオプションおよび機能キーを表示しません。

上

---

## **オプション・コマンドのログ (LOGCOM)**

PDMオプションまたはユーザー定義オプションから発生したコマンドがジョブ・ログにログされるかどうかを指定します。

### **\*SAME**

このパラメーターが以前に設定されていた場合、値は変わりません。そうでない場合には\*NOが使用されます。

**\*NO** PDMオプションまたはユーザー定義オプションから発生したコマンドをログしません。

**\*YES** PDMオプションまたはユーザー定義オプションから発生したコマンドをログします。

上

---

## 実行キーでリストを終了 (EXITENT)

ユーザーが実行キーでリスト・パネルを終了できるかどうかを指定します。リスト・パネルには、ライブラリー、オブジェクト、メンバー、ユーザー定義オプションが含まれます。

### \*SAME

このパラメーターが以前に設定されていた場合、値は変わりません。そうでない場合には\*NOが使用されます。

**\*NO** ユーザーは実行キーでリスト・パネルを終了できません。

**\*YES** ユーザーは実行キーでリスト・パネルを終了できます。

上

---

## 通知メッセージの表示 (DSPINFMSG)

新規ツール(RSEおよびCODE)に関する通知メッセージを表示するかどうかを指定します。

### \*SAME

このパラメーターが以前に設定されていた場合、値は変わりません。そうでない場合には\*YESが使用されます。

**\*YES** 通知メッセージが表示されます。

**\*NO** 通知メッセージが表示されません。

上

---

## 例

### Example 1:新規PDMユーザーに省略時の値を設定

```
CHGPDMDFT USER(USER2)
```

このコマンドはUSER2が新規PDMユーザーの場合にPDM省略時の値を設定します。省略時の値がすでに設定されている場合にはそれらは変更されません。

### 例 2:既存PDMユーザーの省略時の値を変更

```
CHGPDMDFT USER(USER2) CRTBCH(*NO)  
RUNBCH(*YES)
```

このコマンドでは既存PDMユーザー、USER2のPDM省略時の値を変更します。これにより、プログラム・コンパイルおよびモジュール作成が対話式で行われオブジェクトの実行はバッチで行われます。

上

---

## エラー・メッセージ

### \*ESCAPE メッセージ

#### PDM0004

ライブラリー&1が見つからない。

**PDM0010**

ライブラリー名&1が間違っている。

**CPF0001**

&1コマンドでエラーが見つかった。

**CPF2204**

ユーザー・プロファイル&1が見つからない。

**CPF2209**

ライブラリー&1が見つかりません。

**CPF2228**

ユーザー・プロファイル変更は認可されていない。

上



## 物理ファイル・メンバーの比較 (CMPPFM)

実行可能場所: すべての環境 (\*ALL)  
スレッド・セーフ: いいえ

パラメーター  
例  
エラー・メッセージ

このコマンドによって、ソース物理ファイル・メンバーを比較することができます。

### エラー・メッセージ: CMPPFM

なし

上

### パラメーター

キーワード	記述	選択項目	ノート
NEWFILE	新ファイル	修飾オブジェクト名	必須, 定位置 1
	修飾子 1: 新ファイル	名前	
	修飾子 2: ライブラリー	名前, *LIBL, *CURLIB	
NEWMBR	新メンバー	単一値: *FIRST, *ALL その他の値 (最大 25 回の繰り返し): 文字値	オプション, 定位置 2
OLDFILE	旧ファイル	修飾オブジェクト名	オプション, 定位置 3
	修飾子 1: 旧ファイル	名前, *NEWFILE	
	修飾子 2: ライブラリー	名前, *LIBL, *CURLIB	
OLDMBR	旧メンバー	単一値: *NEWMBR, *FIRST その他の値 (最大 25 回の繰り返し): 名前	オプション, 定位置 4
CMPTYPE	比較タイプ	*LINE, *FILE, *WORD	オプション, 定位置 5
RPTTYPE	報告書タイプ	*DIFF, *SUMMARY, *CHANGE, *DETAIL	オプション, 定位置 6
OUTPUT	出力	*, *PRINT, *OUTFILE	オプション, 定位置 7
OUTFILE	出力を受け取るファイル	修飾オブジェクト名	オプション, 定位置 8
	修飾子 1: 出力を受け取るファイル	名前	
	修飾子 2: ライブラリー	名前, *LIBL, *CURLIB	
OUTMBR	出力メンバー・オプション	要素リスト	オプション, 定位置 9
	要素 1: 出力を受け取るメンバー	名前, *FIRST	
	要素 2: レコードの置き換えまたは追加	*REPLACE, *ADD	
SRCTYPE	ソース・タイプの選択	名前, *ALL	オプション, 定位置 10

キーワード	記述	選択項目	ノーツ
<b>OPTION</b>	処理オプション	値 (最大 10 回の繰り返し): *IGNORECASE, *COUNT, *CBLSRCCOL, *RPGSRCCOL, *RPGLESRCCOL, *OMTDUP, *OMTREFMT, *OMTBASCMT, *OMTBLANK, *OMTCCMT, *OMTCBLCMT, *OMTCLCMT, *OMTCMDCMT, *OMTDDSCMT, *OMTPASCMT, *OMTPLICMT, *OMTRPGCMT, *OMTRPGLECMT, *FLGMOVLIN, *CHGFLGS, *CHANGES, *LONGLINES, *NARROW, *CMPSEQDAT, *COUNTREFMT, *WIDE	オプションル, 定位置 11
<b>STMTFILE</b>	ステートメント・ファイル	修飾オブジェクト名	オプションル, 定位置 12
	修飾子 1: ステートメント・ファイル	名前	
	修飾子 2: ライブラリー	名前, *LIBL, *CURLIB	
<b>STMTMBR</b>	ステートメント・メンバー	名前, *FIRST	オプションル, 定位置 13

上

## 新ファイル (NEWFILE)

比較されるメンバーが入っている新物理ファイルを識別します。

**\*LIBL** ジョブのライブラリー・リストを使用します。

**\*CURLIB**

ジョブの現行ライブラリーを使用します。

**ライブラリー名**

指定されたライブラリーを使用します。

**新ファイル名**

指定された新ファイルを使用します。

上

## 新メンバー (NEWMBR)

新ファイル中の比較されるメンバーを識別します。

**\*FIRST**

ファイルの最初のメンバーを使用します。

**新ファイル・メンバー名**

指定されたメンバーまたはメンバーのリストを使用します。

このパラメーターに複数の値を入力するためには、値の続きは+ プロンプトにプラス符号(+)を入力して、実行キーを押してください。



**総称\*** 指定されたパターンと一致するメンバーを使用します。

**\*ALL** ファイル中のすべてのメンバーを使用します。

上

---

## 旧ファイル (OLDFILE)

比較されるメンバーが入っている旧物理ファイルを識別します。

**\*LIBL** ジョブのライブラリー・リストを使用します。

**\*CURLIB**

ジョブの現行ライブラリーを使用します。

**ライブラリー名**

指定されたライブラリーを使用します。

**\*NEWFILE**

NEWFILEキーワードに指定されたものと同じ名前の旧ファイルを使用します。

**旧ファイル名**

指定された旧ファイルを使用します。

上

---

## 旧メンバー (OLDMBR)

旧ファイル中の比較されるメンバーを識別します。

**\*NEWMBR**

NEWMBRキーワードに指定されたものと同じメンバーまたはメンバー・リストを使用します。

**旧ファイル・メンバー名**

指定されたメンバーまたはメンバーのリストを使用します。

このパラメーターに複数の値を入力するためには、値の続きは+ プロンプトにプラス符号(+)を入力して、実行キーを押してください。

**\*FIRST**

ファイルの最初のメンバーを使用します。

上

---

## 比較タイプ (CMPTYPE)

実行する比較のタイプを指定します。

**\*LINE** 行レベルの相違点を比較し、挿入された行および削除された行を識別します。

**\*FILE** ファイル・レベルの相違点を比較し、相違点がどこにあるかは報告しません。このタイプの比較の結果には、比較されたメンバーが異なるかまたは同じかが指示され、非対のすべてのメンバーの名前が示されます。この方式は要約情報を作成するだけですが、最高速タイプの比較です。

**\*WORD**

語句レベルの相違点を比較します。この比較は、隣接した行の語句を突き合わせることができる点

を除き、\*LINE比較と類似しています。語句は、ブランクまたは行の終わりで区切られます。メンバーは、行の境界またはレコード長のない長い一連の語句として処理されます。この比較の出力では、必ずしも語句の元のスペーシングが維持されるわけではありません。相違点をはっきり見分けることができるように、出力リストにブランクが追加されることがあります。

上

---

## 報告書タイプ (RPTTYPE)

結果報告書のリスト・タイプを指定します。

**\*DIFF** 比較中のメンバーの相違点のみをリストし、続いて要約をリストします。リスト中で、相違点にはフラグが付けられます。

**\*SUMMARY**

比較の結果の要約をリストし、詳細な相違点は表示しません。グループ比較では、処理オプションのリストとともに、グループ中の各メンバーごとに個別の要約行が生成されます。

**\*CHANGE**

相違点の前後10行ずつ、\*DIFF報告書タイプと同じ情報を提供します。この余分な行によって、周囲のデータの文脈内の相違点を参照することができます。

**\*DETAIL**

新ファイル・メンバー全体（および旧ファイル・メンバーからの削除分）をリストし、相違点を指示して、結果の要約を提供します。

上

---

## 出力 (OUTPUT)

比較の結果を表示、印刷、あるいは物理ファイルに記憶するかどうかを指定します。

**\*** 結果を表示します。

**\*PRINT**

結果をスプール・ファイルに印刷します。

**\*OUTFILE**

結果を物理ファイルに記憶します。

上

---

## 出力を受け取るファイル (OUTFILE)

出力結果の宛先となるファイルを指定します。

**\*LIBL** ジョブのライブラリー・リストを使用します。

**\*CURLIB**

ジョブの現行ライブラリーを使用します。

**ライブラリー名**

指定されたライブラリーを使用します。

### 出力物理ファイル名

指定された物理ファイルを使用します。

上

---

## 出力メンバー・オプション (OUTMBR)

出力の宛先となるメンバーを指定します。OUTFILEキーワードに印刷装置ファイルを指定した場合には、このパラメーターの値は指定しないでください。

### \*FIRST

ファイルの最初のメンバーを使用します。

### 出力ファイル・メンバー名

指定されたメンバーを使用します。

### \*REPLACE

メンバーが存在していた場合には、それを置き換えます。

**\*ADD** このメンバーをファイルに追加します。

上

---

## ソース・タイプの選択 (SRCTYPE)

比較するソース・メンバー・タイプを指定します。1つのタイプまたはすべてのタイプを選択することができます。

**\*ALL** すべてのソース・メンバーを比較します。

### ソース・メンバー・タイプ

指定されたソース仕様タイプのメンバーだけを比較します。

上

---

## 処理オプション (OPTION)

比較をカスタマイズするための処理オプションのリストを指定します。リストに10 以内の処理オプションを指定することができます。

このパラメーターに複数の値を入力するためには、値の続きは+ プロンプトにプラス符号(+)を入力して、実行キーを押してください。

注: 注記を省略するためのOMTXXXオプションを使用した場合には、ユーティリティは注記でない一部のストリングを注記として認識することがあります。

たとえば、ストリング/\* \*/がプログラムの実行可能コードに組み込まれていた場合で、\*OMTCCMT処理オプションを使用した場合には、そのストリングは処理時に無視されることもあれば、無視されないこともあります。

#### **\*CBLSRCCOL**

COBOLのソース桁(7-72桁)のみを比較します。このオプションでは、比較の有効範囲がソース・コード桁の区域に限定されます。このオプションは、行および語句比較の場合に使用することができます。

#### **\*CHANGES**

変更された項目のみを要約にリストします。グループ比較では、通常、対になったすべてのメンバーがリストされます。グループ比較で\*CHANGESを指定した場合には、変更されたメンバー対のみが要約セクションにリストされます。このオプションは、行、ファイル、および語句比較の場合に使用することができます。

#### **\*CHGFLGS**

新ファイルの該当する行の1桁目に変更フラグ(>)を入れることによって、変更を指示するリストを生成します。削除された行は、削除された行の次の行にフラグを付けることによって指示されます。このオプションは、行および語句比較の場合に使用することができます。

#### **\*CMPSEQDAT**

ソース順序の起点を1に設定します。ソース物理ファイル・メンバーの順序および日付フィールドを比較します。このオプションは、行および語句比較の場合に使用することができます。

#### **\*COUNT**

グループ（行）比較で、非対のメンバーからの行をカウントし、結果を要約に組み込みます。このオプションを使用しない場合には、対になったメンバーからの行に関する統計しか得られません。

#### **\*COUNTREFMT**

再形式設定された行にフラグは付けられませんが、全体的な要約統計用にはカウントされます。このオプションは、行および語句比較の場合に使用することができます。

#### **\*FLGMOVLIN**

移動した行にフラグを付けます。新ファイルに挿入された行で、旧ファイルの削除された行と一致するものを識別します。このオプションを使用できるのは、行比較の場合だけです。

#### **\*IGNORECASE**

文字（大文字、小文字、混合）による相違点は無視します。このオプションは、行および語句比較の場合に使用することができます。

#### **\*LONGLINES**

ファイルからの最高176桁を反映する198桁のリストを作成します。このオプションは、行比較の場合に使用することができます。

#### **\*NARROW**

横方向に55桁ずつ132桁のリスト・ファイルを作成します。挿入および削除された行にはフラグが付けられ、リスト出力に横並びで表示されます。このオプションは、行比較の場合に使用することができます。

#### **\*OMTBASCMT**

BASICの注記を省略します。BASICの注記はブランクとされ、除外されます。

#### **\*OMTBLANK**

比較されている桁がブランクである行を省略します。このオプションは、行および語句比較の場合に使用することができます。

#### **\*OMTCBLCMT**

COBOLの注記を省略します。COBOLの注記およびブランク行は、すべての注記が除去またはブランクとされたリストを生成するために、比較セットから除外されます。このオプションは、行および語句比較の場合に使用することができます。

**\*OMTCCMT**

Cの注記を省略します。Cの注記およびブランク行は、すべての注記が除去またはブランクとされたリストを生成するために、比較セットから除外されます。このオプションは、行および語句比較の場合に使用することができます。

**\*OMTCLCMT**

CLの注記を省略します。CLプログラムの注記およびブランク行は、すべての注記が除去またはブランクとされたリストを生成するために、比較セットから除外されます。このオプションは、行および語句比較の場合に使用することができます。

**\*OMTCMDCMT**

CMDの注記を省略します。CLコマンドの注記およびブランク行は、すべての注記が除去またはブランクとされたリストを生成するために、比較セットから除外されます。このオプションは、行および語句比較の場合に使用することができます。

**\*OMTDDSCMT**

DDSの注記を省略します。DDSの注記およびブランク行は、すべての注記が除去またはブランクとされたリストを生成するために、比較セットから除外されます。このオプションは、行および語句比較の場合に使用することができます。

**\*OMTDUP**

重複行を省略します。新ファイルのソース行と一致する旧ファイルのソース行が横並びのリストから省略されます。このオプションは、行比較の場合に使用することができます。

**\*OMTPASCMT**

PASCALの注記を省略します。PASCALの注記およびブランク行は、すべての注記が除去またはブランクとされたリストを生成するために、比較セットから除外されます。このオプションは、行および語句比較の場合に使用することができます。

**\*OMTPLICMT**

PLIの注記を省略します。PLIの注記およびブランク行は、すべての注記が除去またはブランクとされたリストを生成するために、比較セットから除外されます。このオプションは、行および語句比較の場合に使用することができます。

**\*OMTREFMT**

再形式設定された行を省略します。旧ファイル・メンバーの再形式設定された行がリストから省略されます。新ファイル・メンバーの再形式設定された行はリストに含まれます。通常は、この両方がリストされます。このオプションは、行比較の場合に使用することができます。

**\*OMTRPGCMT**

RPGの注記を省略します。RPGの注記およびブランク行は、すべての注記が除去またはブランクとされたリストを生成するために、比較セットから除外されます。このオプションは、行および語句比較の場合に使用することができます。

**\*OMTRPGLECMT**

ILE RPGの注記を省略します。ILE RPGの注記およびブランク行は、すべての注記が除去またはブランクとされたリストを生成するために、比較セットから除外されます。このオプションは、行および語句比較の場合に使用することができます。

**\*RPGLESRCOL**

ILE RPGのソース桁(6-100桁)のみを比較します。比較の有効範囲がソース・コード桁の区域に限定されます。このオプションは、行および語句比較の場合に使用することができます。

#### **\*RPGSRCCOL**

RPGのソース桁(6-74桁)のみを比較します。比較の有効範囲がソース・コード桁の区域に限定されます。このオプションは、行および語句比較の場合に使用することができます。

#### **\*WIDE**

(横方向に80桁ずつ) 198桁の幅の横並びのリストを作成します。このオプションは、行および語句比較の場合に使用することができます。

上

---

## **ステートメント・ファイル (STMTFILE)**

処理ステートメントを保留するユーザー定義のソース物理ファイルを指定します。このファイルのレコードは、任意の長さとしてできますが、読み取られるのは最初の80バイトだけです。

**\*LIBL** ジョブのライブラリー・リストを使用します。

#### **\*CURLIB**

ジョブの現行ライブラリーを使用します。

#### **ライブラリー名**

指定されたライブラリーを使用します。

#### **ステートメント・ファイル名**

指定されたファイルをステートメント・ファイルとして使用します。

上

---

## **ステートメント・メンバー (STMTMBR)**

比較で使用する処理ステートメントが入っているファイル・メンバーを指定します。

#### **\*FIRST**

ファイルの最初のメンバーを使用します。

#### **ステートメント・ファイル・メンバー名**

指定されたメンバーを使用します。

上

---

## **例**

なし

上

---

## **エラー・メッセージ**

なし

上

## DBCSマスター分類テーブルのコピー (CPYIGCSRT)

実行可能場所:

- バッチ・ジョブ (\*BATCH)
- 対話式ジョブ (\*INTERACT)
- 対話式プログラム (\*IPGM)
- バッチ REXX プロシージャ (\*BREXX)
- 対話式 REXX プロシージャ (\*IREXX)
- QCMDEXEC, QCAEXEC, または QCAPCMD API (\*EXEC) の使用

スレッド・セーフ: いいえ

パラメーター

例

エラー・メッセージ

CPYIGCSRT (DBCSマスター分類テーブルのコピー) コマンドは、データ・ファイルからDBCSマスター分類テーブルのオブジェクトに、あるいはDBCSマスター分類テーブルのオブジェクトからデータ・ファイルに、日本語DBCS文字のDBCSマスター分類テーブルをコピーします。システム/36システムに移動中、あるいはシステム/36システムから移動したマスター・ファイルとの間で文字をコピーするためには、CPYIGCSRTコマンドを使用してください。

## エラー・メッセージ: CPYIGCSRT

なし

上

## パラメーター

キーワード	記述	選択項目	ノート
OPTION	コピー方向	*OUT, *IN	必須, 定位置 1
FILE	ファイル	修飾オブジェクト名	オプション, 定位置 2
	修飾子 1: ファイル	名前, # <u>KAMAST</u>	
	修飾子 2: ライブラリー	名前, * <u>CURLIB</u> , *LIBL	
MBR	メンバー	名前, * <u>FIRST</u> , *FILE	オプション, 定位置 3

上

## コピー方向 (OPTION)

DBCSマスター分類テーブルの複写の方法を指定します。

指定できる値は次のとおりです。

**\*IN** DBCSマスター分類テーブルはファイルからコピーされます。

**\*OUT** DBCSマスター分類テーブルはファイルへコピーされます。

---

## ファイル (FILE)

コピーに使用するファイルの名前を指定します。

指定できる値は次のとおりです。

### #KAMAST

別のファイル名を指定しない場合には、このファイルが使用されます。#KAMAST はシステム/36 のマスター・ファイルの名前です。

### ファイル名

レコード長が20桁の物理ファイルの名前。このファイルから分類テーブルをコピーしている場合には、このファイル名は前もって存在していなければなりません。

---

## メンバー (MEMBER)

コピーに使用する物理ファイル・メンバーを指定します。

指定できる値は次のとおりです。

### \*FIRST

ファイルの最初のメンバーが使用されます。

**\*FILE** ファイル名は使用されるファイル・メンバーを識別します。

### メンバー名

コピーに使用する物理ファイル・メンバーの名前。

---

## 例

なし

---

## エラー・メッセージ

なし



## 結合Cプログラム作成 (CRTBNDC)

実行可能場所: すべての環境 (\*ALL)

スレッド・セーフ: いいえ

パラメーター

例

エラー・メッセージ

結合Cプログラム作成(CRTBNDC)コマンドはILE Cコンパイラーを開始します。このコンパイラーは、ソース・コード内のILE Cステートメントに基づいてプログラム・オブジェクトを作成しようとします。完全なコンパイル手順が常に実行されます。

注: CRTBNDCコマンドが呼び出された場合には、一時\*MODULEオブジェクトがQTEMPライブラリーの中に作成されます。一時モジュール・オブジェクトの名前は、**プログラム (PGM)**パラメーターに指定された名前と同じになります。その名前を持つ\*MODULEオブジェクトがすでにQTEMPに存在していた場合には、エラー・メッセージが生成され、コンパイルは停止します。CRTBNDCコマンドによって使用される一時モジュール・オブジェクトは、コンパイルが正常に行われようといわれまいと、コンパイルが停止した後に削除されます。

上

### パラメーター

キーワード	記述	選択項目	ノート
PGM	プログラム	修飾オブジェクト名	必須, 定位置 1
	修飾子 1: プログラム	名前	
	修飾子 2: ライブラリー	名前, <u>*CURLIB</u>	
SRCFILE	ソース・ファイル	修飾オブジェクト名	オプション, 定位置 2
	修飾子 1: ソース・ファイル	名前, <u>QCSRC</u>	
	修飾子 2: ライブラリー	名前, <u>*LIBL</u> , <u>*CURLIB</u>	
SRCMBR	ソース・メンバー	名前, <u>*PGM</u>	オプション, 定位置 3
SRCSTMF	ソース・ストリーム・ファイル	パス名	オプション
TEXT	テキスト・記述	文字値, <u>*SRCMBRTXT</u> , *BLANK	オプション
OUTPUT	出力オプション	単一値: <u>*NONE</u> , '*トナトオ' その他の値: 要素リスト	オプション
	要素 1: 出力ファイル名	パス名, *PRINT, '*ニネクトホ'	
	要素 2: タイトル	文字値, <u>*BLANK</u>	
	要素 3: サブタイトル	文字値, <u>*BLANK</u>	

キーワード	記述	選択項目	ノーツ
<b>OPTION</b>	コンパイラー・オプション	値 (最大 35 回の繰り返し): *NOAGR, *AGR, *DIGRAPH, *NODIGRAPH, *NOEVENTF, *EVENTF, *NOEXPMAC, *EXPMAC, *NOFULL, *FULL, *GEN, *NOGEN, *NOINCDIRFIRST, *INCDIRFIRST, *LOGMSG, *NOLOGMSG, *NOSECLVL, *SECLVL, *NOSHOWINC, *SHOWINC, *NOSHOWSKP, *SHOWSKP, *SHOWSRC, *NOSHOWSRC, *NOSHOWSYS, *SHOWSYS, *NOSHOWUSR, *SHOWUSR, *STDINC, *NOSTDINC, *NOSTDLOGMSG, *STDLOGMSG, *NOSTRUCREF, *STRUCREF, *NOSYSINCPATH, *SYSINCPATH, *NOXREF, *XREF, *NOXREFREF, *XREFREF	オプション
<b>CHECKOUT</b>	チェックアウト・オプション	値 (最大 39 回の繰り返し): <u>*NONE</u> , *USAGE, *ALL, *NOCOND, *COND, *NOCONST, *CONST, *NOEFFECT, *EFFECT, *NOENUM, *ENUM, *NOEXTERN, *EXTERN, *NOGENERAL, *GENERAL, *NOGOTO, *GOTO, *NOINIT, *INIT, *NOPARM, *PARM, *NOPORT, *PORT, *NOPPCHECK, *PPCHECK, *NOPPTRACE, *PPTRACE, *NOREACH, *REACH, *NOTRUNC, *TRUNC, *NOUNUSED, *UNUSED	オプション
<b>OPTIMIZE</b>	最適化	<u>10</u> , 20, 30, 40	オプション
<b>INLINE</b>	インライン・オプション	要素リスト	オプション
	要素 1: インライナー	<u>*OFF</u> , *ON	
	要素 2: モード	<u>*AUTO</u> , *NOAUTO	
	要素 3: しきい値	1-65535, <u>250</u> , *NOLIMIT	
	要素 4: 限界	1-65535, <u>2000</u> , *NOLIMIT	
	要素 5: 報告書	<u>*NO</u> , *YES	
<b>DBGVIEW</b>	デバッグ・ビュー	<u>*NONE</u> , *ALL, *STMT, *SOURCE, *LIST	オプション
<b>DEFINE</b>	名前の定義	単一値: <u>*NONE</u> その他の値 (最大 32 回の繰り返し): 文字値	オプション
<b>LANGLVL</b>	言語レベル	<u>*EXTENDED</u> , *ANSI	オプション

キーワード	記述	選択項目	ノーツ
ALIAS	別名	値 (最大 3 回の繰り返し): <u>*ANSI</u> , *NOANSI, *ADDRTAKEN, *NOADDRTAKEN, *ALLPTRS, *NOALLPTRS, *TYPEPTR, *NOTYPEPTR	オプション
SYSIFCOPT	SYSTEMインターフェースOPT	値 (最大 2 回の繰り返し): *NOIFSIO, *IFSIO, *IFS64IO, *NOASYNCSIGNAL, *ASYNCSIGNAL	オプション
LOCALETYPE	LOCALEオブジェクト・タイプ	<u>*LOCALE</u> , *LOCALEUCS2, *LOCALEUTF, *CLD	オプション
FLAG	メッセージのフラグ・レベル	<u>0</u> , 10, 20, 30	オプション
MSGMT	コンパイラ・メッセージ	要素リスト	オプション
	要素 1: メッセージ限界	0-32767, <u>*NOMAX</u>	
	要素 2: メッセージ限界の重大度	0, 10, 20, <u>30</u>	
REPLACE	PGMオブジェクトの置き換え	<u>*YES</u> , *NO	オプション
USRPRF	ユーザー・プロファイル	<u>*USER</u> , *OWNER	オプション
AUT	権限	名前, <u>*LIBCRTAUT</u> , *ALL, *CHANGE, *USE, *EXCLUDE	オプション
TGTRLS	ターゲット・リリース	単純名, <u>*CURRENT</u> , *PRV	オプション
ENBPFRCOL	パフォーマンス収集の使用可能	要素リスト	オプション
	要素 1: 収集レベル	<u>*PEP</u> , *ENTRYEXIT, *FULL	
	要素 2: プロシージャ	*NONLEAF, *ALLPRC	
PFOPT	パフォーマンス・オプション	値 (最大 2 回の繰り返し): *SETFPCA, *NOSETFPCA, *NOSTRDONLY, *STRDONLY	オプション
PRFDTA	プロファイル作成データ	<u>*NOCOL</u> , *COL	オプション
TERASPACE	テラスペース・オプション	単一値: <u>*NO</u> その他の値: 要素リスト	オプション
	要素 1: テラスペース使用可能	<u>*YES</u>	
	要素 2: TERASPACE INTERFACESの使用	<u>*NOTSIFC</u> , *TSIFC	
STGMDL	ストレージ・モデル	<u>*SINGLVL</u> , *TERASPACE	オプション
DTAMD	データ・モデル	<u>*P128</u> , *LLP64	オプション
PACKSTRUCT	パック構造	<u>*NATURAL</u> , 1, 2, 4, 8, 16	オプション
ENUM	Enumサイズ	<u>*SMALL</u> , 1, 2, 4, *INT	オプション
MAKEDEP	依存関係情報	パス名, <u>*NONE</u>	オプション
INCDIR	組み込みディレクトリー	単一値: <u>*NONE</u> その他の値 (最大 32 回の繰り返し): パス名	オプション
CSOPT	コンパイラ・サービスOPT	文字値, <u>*NONE</u>	オプション
LICOPT	ライセンス内部コードOPT	文字値, <u>*NONE</u>	オプション
DFTCHAR	省略時の文字タイプ	<u>*UNSIGNED</u> , *SIGNED	オプション
TGTCCSID	ターゲットCCSID	1-65535, <u>*SOURCE</u> , *JOB, *HEX	オプション

キーワード	記述	選択項目	ノーツ
DECFLTRND	10進浮動丸めモード	*HALFEVEN, *DOWN, *UP, *HALFUP, *HALFDOWN, *FLOOR, *CEILING	オプション

上

---

## プログラム (PGM)

作成中のILE Cプログラム・オブジェクトを指定します。

これは必須パラメーターです。

### 修飾子1: プログラム

**name** プログラム・オブジェクトの名前を指定してください。

### 修飾子2: ライブラリー

#### \*CURLIB

プログラム・オブジェクトは現行ライブラリーに保管されます。ジョブに現行ライブラリーがない場合には、プログラム・オブジェクトはQGPLライブラリーの中に作成されます。

**name** 作成中のプログラム・オブジェクトの保管先となるライブラリーの名前を指定してください。

上

---

## ソース・ファイル (SRCFILE)

コンパイルが必要なILE Cソース・コードが入っているソース・ファイルを指定します。

### 修飾子1: ソース・ファイル

#### QCSRC

QCSRCという名前のソース・ファイルには、コンパイルしたいILE Cソース・コードを含むメンバーが入っています。

**name** ILE Cソース・コードが含まれたメンバーが入っているファイルの名前を指定してください。

### 修飾子2: ライブラリー

**\*LIBL** ソース・ファイルが入っているライブラリーを見つけるためにライブラリー・リストが検索されます。

#### \*CURLIB

ソース・ファイルが現行ライブラリーから検索されます。ジョブに現行ライブラリーがない場合には、ソース・ファイルはQGPLから検索されます。

**name** ソース・ファイルが入っているライブラリーの名前を指定してください。

上

---

## ソース・メンバー (SRCMBR)

コンパイルされるソース・コードが入っているメンバーの名前を指定します。

**\*PGM プログラム (PGM)** パラメーターに指定された名前は、ソース・メンバー名として使用されます。

**name** ソース・コードが入っているメンバーの名前を指定してください。

上

---

## ソース・ストリーム・ファイル (SRCSTMF)

コンパイルが必要なソース・コードが入っているストリーム・ファイルを指定します。

パス名は絶対修飾パス名か相対修飾パス名のいずれかとすることができます。絶対パス名は'/'で始まり、相対パス名は'/'以外の文字で始まります。絶対修飾した場合には、そのパス名で完全です。相対修飾した場合には、パス名は、ジョブの現行作業ディレクトリーをパス名に対して事前保留することによって完全なものとなります。

SRCMBRおよびSRCFILEパラメーターをSRCSTMFパラメーターと一緒に指定することはできません。

**path-name**

C++ソース・コードが入っているストリーム・ファイルのパス名を指定してください。

上

---

## テキスト'記述' (TEXT)

プログラム・オブジェクトを簡潔に記述するテキストを指定します。

**\*SRCMBRTXT**

ソース・ファイル・メンバーに関連付けられたテキスト記述が、プログラム・オブジェクトに使用されます。ソース・ファイルがインライン・ファイル、ストリーム・ファイル、または装置ファイルである場合には、テキストはブランクとなります。

**\*BLANK**

テキストは表示されないことを指定します。

**文字値** 50文字を超えないテキストをアポストロフィで囲んで指定します。

上

---

## 出力オプション (OUTPUT)

コンパイラー・リストが生成されるかどうかを指定します。

**単一値**

**\*NONE**

コンパイラー・リストは生成しません。リストが必要でない場合には、コンパイル時のパフォーマンスを改善するためにこのパラメーター値を使用する必要があります。\*NONEを指定した場合には、OPTIONパラメーターに指定されたリストに関連したすべてのパラメーター値が無視されます。

## 要素1: 出力ファイル名

### **\*PRINT**

リストを含むスプール・ファイルが生成されます。

### ***path-name***

リストを保留するストリーム・ファイルのパス名を指定します。

## 要素2: タイトル

### **\*BLANK**

テキストは表示されないことを指定します。

**文字値** リスト・ファイルのタイトル文字列（最大80文字）を指定します。

## 要素3: サブタイトル

### **\*BLANK**

テキストは表示されないことを指定します。

**文字値** リスト・ファイルのサブタイトル文字列（最大80文字）を指定します。

上

---

## コンパイラー・オプション (OPTION)

ILE Cソース・コードのコンパイル時に使用するオプションを指定します。それらは、1つまたは複数のブランクで区切り、任意の順序で指定できます。1つのオプションが複数回指定されたり、2つのオプションが競合する場合には、最後に指定されたものが使用されます。

このパラメーターには35個の値を指定できます。

このパラメーターには単一のデフォルト値はありません。以下に示す一連のパラメーター値の説明では、下線付きの値は、コンパイラーで使用するデフォルト値を表します。

### **\*NOAGR**

リストの中に集約マップは生成しません。

**\*AGR** リストの中にすべての集約のマップを生成します。マップには構造と共用体が組み込まれます。構造マップには、メンバーの埋め込みが示されます。このオプションによって\*STRUCREFオプションが上書きされます。

### **\*DIGRAPH**

ソース・コードの二重音字の使用を許可します。

### **\*NODIGRAPH**

ソース・コードの二重音字の使用は許可しません。

### **\*NOEVENTF**

CoOperative Development Environment/400 (CODE/400)で使用するためのイベント・ファイルは作成しません。

### **\*EVENTF**

CoOperative Development Environment/400 (CODE/400)で使用するためのイベント・ファイルを作成します。イベント・ファイルは、作成中のモジュールまたはプログラム・オブジェクトの保管先のライブラリーにあるファイルEVFEVENTにメンバーとして作成されます。ファイルEVFEVENTが存在しない場合には、それが自動的に作成されます。イベント・ファイル・メンバー名は作成中のオ

プロジェクトの名前と同じです。CODE/400はこのファイルを使用して、CODE/400エディターに統合されたエラー・フィードバックを提供します。通常、イベント・ファイルは、CODE/400内からモジュールまたはプログラム・オブジェクトを作成するときに生成されます。

#### **\*NOEXPMAC**

マクロの中で構文エラーが検出されないかぎり、マクロはリスト中で拡張されません。

#### **\*EXPMAC**

リスト中ですべてのマクロを拡張します。このパラメーターは、DBGVIEW(\*ALL)およびDBGVIEW(\*LIST)と競合します。DBGVIEW(\*ALL)やDBGVIEW(\*LIST)と一緒にOPTION(\*EXPMAC)を使用すると、エラー・メッセージを出力してコンパイルは停止します。

#### **\*NOFULL**

すべてのリスト・オプションをオンにしません。

#### **\*FULL**

すべてのリスト・オプションをオンにします。

#### **\*NOINCDIRFIRST**

INCDIRパラメーターとして指定された組み込みディレクトリーは、標準ヘッダー・ファイル組み込みパスの前には組み込まれません。

#### **\*INCDIRFIRST**

INCDIRパラメーターとして指定された組み込みディレクトリーが、標準ヘッダー・ファイル組み込みパスの前に組み込まれます。

#### **\*LOGMSG**

コンパイル・メッセージをジョブ・ログに書き込みます。

このオプションとFLAGパラメーターを指定した場合には、FLAGパラメーターに指定された（およびそれ以上の）重大度のメッセージがジョブ・ログに入れます。

このオプションと，MSGLMTパラメーターにメッセージの最大数を指定した場合には，指定された重大度のその数のメッセージがジョブ・ログに入れられた時にコンパイルが停止します。

#### **\*NOLOGMSG**

コンパイル・メッセージをジョブ・ログに書き込みません。

#### **\*NOSECLVL**

リストの中に第2レベル・メッセージ・テキストは生成しません。

#### **\*SECLVL**

リストの中に第2レベル・メッセージ・テキストを生成します。このオプションを有効にするには、OUTPUTオプションを指定する必要があります。

#### **\*NOSHOWINC**

リストのソース・セクションで、またはデバッグ・ビューでユーザー組み込みファイルまたはシステム組み込みファイルを拡張しません。

#### **\*SHOWINC**

リストのソース・セクションで、またはデバッグ・ビューでユーザー組み込みファイルとシステム組み込みファイルの両方を拡張します。OUTPUTオプションか、\*ALL、\*SOURCE、または\*LISTのDBGVIEWパラメーター値を指定する必要があります。

#### **\*NOSHOWSKP**

リストのソース・セクションまたはデバッグ・リスト・ビューでプリプロセッサが無視したステートメントを組み込みません。プリプロセッサは、プリプロセッサのディレクティブが偽（ゼロ）と評価した結果としてステートメントを無視します。



#### **\*SHOWSKP**

プリプロセッサがスキップしたかどうかには関係なく、リストのソース・セクションまたはデバッグ・リスト・ビュー内のすべてのステートメントを組み込みます。OUTPUTオプションか、\*ALLまたは\*LISTのDBGVIEWパラメーター値を指定する必要があります。

#### **\*SHOWSRC**

リストにソース・コードを表示します。このオプションは、\*SHOWINC、\*SHOWSYS,または\*SHOWUSRオプションによって変更することができます。

#### **\*NOSHOWSRC**

リストにソース・コードは表示しません。このオプションは、\*EXPMAC、\*SHOWINC、\*SHOWSYS,または\*SHOWUSRオプションで変更できます。

#### **\*NOSHOWSYS**

リストのソース・セクションで、またはデバッグ・ビューで#includeディレクティブ上のシステム組み込みファイルを拡張しません。システム組み込みファイルは、includeディレクティブの後に続けて不等号括弧(< >)で囲まれます。

#### **\*SHOWSYS**

リストのソース・セクションで、またはデバッグ・ビューで#includeディレクティブ上のシステム組み込みファイルを拡張します。OUTPUTオプションか、\*ALL、\*SOURCE,または\*LISTのDBGVIEWパラメーター値を指定する必要があります。システム組み込みファイルは、includeディレクティブの後に続けて不等号括弧(< >)で囲まれます。

#### **\*NOSHOWUSR**

リストまたはデバッグ・ビューでユーザー組み込みファイルは拡張しません。ユーザー組み込みファイルは、includeディレクティブの後に続けて二重引用符(" ")で囲まれます。

#### **\*SHOWUSR**

リストのソース・セクションで、またはデバッグ・ビューで#includeディレクティブ上のユーザー組み込みファイルを拡張します。OUTPUTオプションか、\*ALL、\*SOURCE,または\*LISTのDBGVIEWパラメーター値を指定する必要があります。ユーザー組み込みファイルは、includeディレクティブの後に続けて二重引用符(" ")で囲まれます。

#### **\*STDINC**

システム提供のヘッダー・ファイルがコンパイルのための検索パスに組み込まれます。

#### **\*NOSTDINC**

システム提供のヘッダー・ファイルはコンパイルのための検索パスに組み込まれません。

#### **\*NOSTDLOGMSG**

コンパイル・メッセージはSTDOUTストリームへ送られません。

#### **\*STDLOGMSG**

コンパイル・メッセージがSTDOUTストリームへ送られます。

#### **\*NOSTRUCREF**

参照されたすべてのSTRUCTまたはUNION変数のマップはリスト・ファイルの中に生成しません。

#### **\*STRUCREF**

参照されたすべてのSTRUCTまたはUNION変数のマップをリスト・ファイルの中に生成します。

#### **\*NOSYSINCPATH**

ユーザー組み込みファイルの検索パスは影響を受けません。

#### **\*SYSINCPATH**

ユーザー組み込みファイルの検索パスをシステム組み込みファイルの検索パスに変更します。機能



的に、このオプションを使用することは、ユーザー#includeディレクティブ(#include "file\_name")の二重引用符を不等号括弧(#include <file\_name>)に変更することに相当します。

#### **\*NOXREF**

リストの中に相互参照テーブルは生成しません。

#### **\*XREF**

ソース・コード中の識別コードをそれらが表示される行の番号とともに示すリストを含む相互参照テーブルを生成します。OUTPUTオプションの指定が必要です。

#### **\*NOXREFREF**

リストの中で参照される識別コードの相互参照テーブルは生成しません。

#### **\*XREFREF**

リスト・ファイルで参照される変数、構造および関数名の相互参照テーブルを生成します。このテーブルには、識別コードが宣言された場所の行番号が示されます。OUTPUTオプションの指定が必要です。

上

---

## **チェックアウト・オプション (CHECKOUT)**

考えられるプログラミング・エラーを指示する通知メッセージの生成を選択できるオプションを指定します。1つのオプションを複数回指定したり、2つのオプションが競合する場合には、最後に指定されたものが使用されます。

このパラメーターには39個の値を指定できます。

注: CHECKOUTでは多くのメッセージが生成されることがあります。これらのメッセージがジョブ・ログへ送られるのを防止するには、OUTPUTパラメーターと一緒にOPTION(\*NOLOGMSG \*NOSTDLOGMSG)を指定して、そのチェックアウト・メッセージをリスト・ファイルに入れます。

#### **\*NONE**

CHECKOUTのためのすべてのオプションを使用不可とします。

#### **\*USAGE**

これは、\*ENUM、\*EXTERN、\*INIT、\*PARM、\*PORT、\*GENERAL,および\*TRUNCを指定するのと同様です。

**\*ALL** CHECKOUTのためのすべてのオプションを使用可能とします。

#### **\*NOCOND**

条件式で考えられる冗長度または問題についての警告はしません。

#### **\*COND**

条件式で考えられる冗長度または問題について警告します。

#### **\*NOCONST**

定数と関係した演算についての警告はしません。

#### **\*CONST**

定数と関係した演算について警告します。

#### **\*NOEFFECT**

有効でないステートメントについての警告はしません。

**\*EFFECT**

有効でないステートメントについて警告します。

**\*NOENUM**

列挙型の使用はリストしません。

**\*ENUM**

列挙型の使用をリストします。

**\*NOEXTERN**

外部宣言を持つ未使用変数はリストしません。

**\*EXTERN**

外部宣言を持つ未使用変数をリストします。

**\*NOGENERAL**

一般チェックアウト・メッセージはリストしません。

**\*GENERAL**

一般チェックアウト・メッセージをリストします。

**\*NOGOTO**

GOTOステートメントの出現および使用はリストしません。

**\*GOTO**

GOTOステートメントの出現および使用をリストします。

**\*NOINIT**

明示的に初期化されない自動変数はリストしません。

**\*INIT** 明示的に初期化されない自動変数をリストします。

**\*NOPARM**

使用されない関数パラメーターはリストしません。

**\*PARM**

使用されない関数パラメーターをリストします。

**\*NOPORT**

C言語の移植不能使用はリストしません。

**\*PORT**

C言語の移植不能使用をリストします。

**\*NOPPCHECK**

プリプロセッサのディレクティブはリストしません。

**\*PPCHECK**

プリプロセッサのすべてのディレクティブをリストします。

**\*NOPPTRACE**

プリプロセッサによる組み込みファイルのトレースはリストしません。

**\*PPTRACE**

プリプロセッサによる組み込みファイルのトレースをリストします。

**\*NOREACH**

到達不能ステートメントについての警告はしません。

**\*REACH**

到達不能ステートメントについて警告します。

### **\*NOTRUNC**

データの考えられる切り捨てまたは喪失についての警告はしません。

### **\*TRUNC**

データの考えられる切り捨てまたは喪失について警告します。

### **\*NOUNUSED**

未使用の自動または静的変数は検査しません。

### **\*UNUSED**

未使用の自動または静的変数を検査します。

上

---

## **最適化 (OPTIMIZE)**

生成されるオブジェクトの最適化のレベルを指定します。

- 10** 生成されるコードは最適化されません。このレベルでのコンパイル時間は最短です。このレベルでは、デバッグ中に変数を表示して変更することができます。
- 20** コードについて一部の最適化が実行されます。このレベルでは、デバッグ中にユーザーが変数を表示することができますが、変更することはできません。
- 30** 生成されるコードについて完全な最適化が実行されます。デバッグ・セッション中は、ユーザー変数は変更することができませんが、表示することはできます。表示される値は、変数の現行値でない場合があります。
- 40** レベル30で実行されたすべての最適化が生成されるコードについて実行されます。さらに、コードは、命令トレースおよび呼び出しトレース・システム機能を使用可能にするプロシージャーのプロローグおよびエピローグ・ルーチンから除去されます。このコードを除去することによって、リーフ・プロシージャーの作成が可能になります。リーフ・プロシージャーとは、他のプロシージャーに対する呼び出しを含まないプロシージャーです。リーフ・プロシージャーに対するプロシージャー呼び出しのパフォーマンスは、通常のプロシージャーに対する呼び出しより高速となります。

上

---

## **インライン・オプション (INLINE)**

コンパイラーが呼び出し先関数の命令による関数呼び出しの置き換えを考慮する必要があるかどうかを指定します。関数をインライン化することによって、呼び出しのオーバーヘッドが取り除かれるので、結果としてより良好に最適化することができます。何度も呼び出される小さな関数は、インライン化の候補として適しています。

### **要素1: インライナー**

インライン化を使用するかどうかを指定します。

**\*OFF** コンパイル単位にインライン化は実行されないことを指定します。

**\*ON** コンパイル単位にインライン化が実行されることを指定します。デバッグ・ビューが指定された場合には、インライン化はオフにされます。

### **要素2: モード**

そのしきい値および限界値に基づいて、インライナーは関数のインライン化を自動的に試みる必要があるかどうかを指定します。

#### **\*AUTO**

インライナーは指定されたしきい値および限界値に基づいて関数をインライン化できるかどうかを判断する必要があることを指定します。**\*AUTO**は**#PRAGMA NOINLINE**ディレクティブによって上書きされます。

#### **\*NOAUTO**

**#PRAGMA INLINE**ディレクティブによって指定された関数のみをインライン化の候補として考慮する必要があることを指定します。

### **要素3: しきい値**

自動インライン化の候補にできる関数の最大サイズを指定します。このサイズは、抽象コード単位(ACU)で測定されます。ACUのサイズは関数中の実行可能コードと比例します。ソース・コードは、コンパイラによってACUに変換されます。

**250** 250のしきい値を指定します。

#### ***I-65535***

ACUのしきい値を指定します。

#### **\*NOLIMIT**

しきい値をプログラム・オブジェクトの最大サイズとして定義します。

### **要素4: 限界**

自動インライン化を停止する前に関数を拡張できる最大相対サイズを指定します。

**2000** 2000 ACUの限界を指定します。

#### **\*NOLIMIT**

限界はプログラム・オブジェクトの最大サイズとして定義されます。システム限界が検出されることがあります。

#### ***I-65535***

ACUの限界を指定してください。

### **要素5: 報告書**

コンパイラ・リストともにインライナー報告書を生成するかどうかを指定します。

**\*NO** インライナー報告書は生成されません。

**\*YES** インライナー報告書がコンパイラ・リストの一部として生成されます。インライナー報告書を生成するには、**OUTPUT**オプションの指定が必要です。

上

---

## **デバッグ・ビュー (DBGVIEW)**

作成されたプログラム・オブジェクトでモジュールに対して使用できるデバッグのレベルを指定します。ソース・レベルのデバッグ用にどのソース・ビューが使用可能かも指定されます。デバッグ・ビューを要求すると、インライン化はオフにされます。

### **\*NONE**

デバッグ機能はプログラム・オブジェクトに挿入されません。

**\*ALL** すべてのデバッグ・オプション(\*STMT, \*SOURCE, および\*LIST)が使用可能になります。

### **\*STMT**

プログラム・ステートメント番号およびシンボリックIDを使用してプログラム・オブジェクトをデバッグできるようにします。

注: \*STMTオプションを使用してモジュール・オブジェクトをデバッグするには、リストが必要です。

### **\*SOURCE**

プログラム・オブジェクトのデバッグのためのソース・ビューを生成します。OPTIONパラメータの値\*NOSHOWINC, \*SHOWINC, \*SHOWSYS, および\*SHOWUSRによって、作成されるソース・ビューの内容が決まります。

注: このビューをデバッグで使用するためには、プログラム・オブジェクトの作成後に、ルート・ソース・ファイルの変更、名前変更、または移動を行わないようにしてください。

**\*LIST** プログラム・オブジェクトのデバッグのためのリスト・ビューを生成します。OPTIONパラメータの値\*SHOWINC, \*SHOWUSR, \*SHOWSYS, および\*NOSHOWINCによって、作成されるリスト・ビューの内容が決まります。

上

---

## **名前の定義 (DEFINE)**

ファイルがコンパイラによって処理される前に有効となるプリプロセッサ・マクロを指定します。形式DEFINE(MACRO)の使用はDEFINE('MACRO=1')と同等です。

### **単一値**

### **\*NONE**

マクロは定義されません。

### **その他の値 (反復は最大32回まで)**

#### **'名前'または'名前=値'**

最大32個のマクロを定義できます。各マクロ名はアポストロフィで囲まれます。マクロ名の最大長は80文字です。アポストロフィは80の文字列の一部ではありません。アポストロフィは、大文字小文字を区別するマクロに必要です。

注: コマンドで定義されたマクロによって、ソースにある同じ名前のすべてのマクロ定義が上書きされますが、コンパイラにより警告メッセージが生成されます。 #define max(a,b) ((a)>(b)?(a):(b))など関数と類似したマクロをコマンド行で定義することはできません。

上

---

## **言語レベル (LANGVL)**

コンパイラの機能と、ソースの作成時に宣言されるプロトタイプを指定します。

### **\*EXTENDED**

プリプロセッサ変数 `__EXTENDED__` を定義し、他の言語レベル変数は未定義とします。このパラメーターは、ILE Cのすべての機能が必要な場合に使用する必要があります。

**\*ANSI** プリプロセッサ変数 `__ANSI__` および `__STDC__` を定義し、他の言語レベル変数は未定義とします。ANSI標準Cだけが使用可能とされます。

注: ILE Cコンパイラーは、常に `__ILEC400__` マクロを事前に定義します。

上

---

## **別名 (ALIAS)**

作成中のプログラム・オブジェクトで適用される別名割り当てアサーションを指定します。

このパラメーターには3個の値を指定できます。

**\*ANSI** プログラム・オブジェクトで、ポインターが、同一タイプのオブジェクトのみを指し示すようになります。

### **\*NOANSI**

プログラム・オブジェクトは、\*ANSI別名割り当て規則を使用しません。

### **\*ADDRTAKEN**

プログラム・オブジェクトで、変数のアドレスが使用されない場合に、その変数のクラスがポインターから切り離されます。

### **\*NOADDRTAKEN**

プログラム・オブジェクトは、\*ADDRTAKEN別名割り当て規則を使用しません。

### **\*ALLPTRS**

プログラム・オブジェクトで、2つのポインターに別名を割り当てることができなくなります。

### **\*NOALLPTRS**

プログラム・オブジェクトは、\*ALLPTRS別名割り当て規則を使用しません。

### **\*TYPEPTR**

プログラム・オブジェクトで、タイプの異なる2つのポインターに別名を割り当てることができなくなります。

### **\*NOTYPEPTR**

プログラム・オブジェクトは、\*TYPEPTR別名割り当て規則を使用しません。

上

---

## **SYSTEMインターフェースOPT (SYSIFCOPT)**

作成中のモジュール・オブジェクトに使用されるシステム・インターフェース・オプションを指定します。それらは、1つまたは複数のブランクで区切り、任意の順序で指定できます。1つのオプションが複数回指定されたり、2つのオプションが競合する場合には、最後に指定されたものが使用されます。

このパラメーターには2個の値を指定できます。

### **\*NOIFSIO**

モジュール・オブジェクトは、Cストリーム入出力操作にデータ管理機能ファイル・システムを使用します。

### **\*IFSIO**

モジュール・オブジェクトは、Cストリーム入出力操作に統合ファイル・システムを使用します。

### **\*IFS64IO**

モジュール・オブジェクトは、64ビットCストリーム入出力操作に統合ファイル・システムを使用します。

### **\*NOASYNC SIGNAL**

同期信号関数の非同期信号関数への実行時マッピングを使用可能にしません。

### **\*ASYNC SIGNAL**

同期信号関数の非同期信号関数への実行時マッピングを使用可能にします。このオプションを指定すると、Cの実行時に同期SIGNAL()およびRAISE()関数がそれぞれ非同期SIGACTION()およびKILL()関数にマップされます。

上

---

## **LOCALEオブジェクト・タイプ (LOCALETYPE)**

作成中のプログラム・オブジェクトで使用するロケール・サポートのタイプを指定します。

### **\*LOCALE**

このオプションで作成されたプログラム・オブジェクトは、\*LOCALEオブジェクトによって指定されるロケール・サポートを使用します。

### **\*LOCALEUCS2**

このオプションで作成されたプログラム・オブジェクトは、\*LOCALEオブジェクトによって指定されるロケール・サポートを使用します。広幅(WIDE)文字タイプには2つの2バイト汎用文字セットの値が含まれることになります。

**\*CLD** このオプションで作成されたプログラム・オブジェクトは、\*CLDオブジェクトによって指定されるロケール・サポートを使用します。

### **\*LOCALEUTF**

このオプションで作成されたプログラム・オブジェクトは、\*LOCALEオブジェクトによって指定されるロケール・サポートを使用します。広幅(WIDE)文字タイプには、4バイトのUTF-32の値が含まれます。狭幅(NARROW)文字タイプには、UTF-8の値が含まれます。

上

---

## **メッセージのフラグ・レベル (FLAG)**

リストに表示するメッセージのレベルを指定します。

- 0** 通知レベルで始まるすべてのメッセージが表示されます。
- 10** 警告レベルで始まるすべてのメッセージが表示されます。
- 20** エラー・レベルで始まるすべてのメッセージが表示されます。
- 30** 重大エラー・レベルで始まるすべてのメッセージが表示されます。



---

## コンパイラー・メッセージ (MSGLMT)

指定されたメッセージ重大度(またはそれ以上)のメッセージの最大数を指定します。コンパイルが停止するまでに、このメッセージ数までメッセージを生成できます。

### 要素1: メッセージ限界

指定したメッセージ重大度レベル以上で生じさせることができるメッセージの最大数を指定します。

#### **\*NOMAX**

指定されたメッセージ重大度レベルで生じたメッセージの数とは無関係に、コンパイルが続行されます。

#### **0-32767**

生じさせることができるメッセージの数を指定します。

### 要素2: メッセージ限界の重大度

メッセージ重大度を指定します。ここで指定した重大度以上のメッセージが特定の数（要素1に指定された数）だけ生成されるとき、コンパイルを停止できます。

- 30**      コンパイルが停止するまでに、重大度30のメッセージを指定の数だけ生成できます。
- 0**        コンパイルが停止するまでに、重大度0以上のメッセージを指定の数だけ生成できます。
- 10**      コンパイルが停止するまでに、重大度10以上のメッセージを指定の数だけ生成できます。
- 20**      コンパイルが停止するまでに、重大度20以上のメッセージを指定の数だけ生成できます。

---

## PGMオブジェクトの置き換え (REPLACE)

オブジェクトの既存のバージョンを現行バージョンで置き換えるかどうかを指定します。

**\*YES**    既存のオブジェクトが新規バージョンで置き換えられます。旧バージョンはQRPLIBに移動し、システム日付および時刻に基づいて名前変更されます。置き換えられたオブジェクトのテキスト記述は、オリジナルのオブジェクトの名前に変更されます。旧オブジェクトは、それが明示的に削除されていない限り、次のIPL時に削除されます。

**\*NO**     既存のオブジェクトは置き換えられません。指定のライブラリーで同じ名前のオブジェクトが見つかった場合、メッセージが表示されて、コンパイルが停止します。

---

## ユーザー・プロファイル (USRPRF)

プログラム・オブジェクトがオブジェクトごとに持つ権限も含めて、作成されたプログラム・オブジェクトの実行時に使用されるユーザー・プロファイルを指定します。プログラム・オブジェクトが使用できるオブジェクトの制御には、プログラム所有者かプログラム・ユーザーのどちらかのプロファイルが使用されます。



### **\*USER**

プログラム・オブジェクトを実行中のユーザーのプロファイルが使用されます。

### **\*OWNER**

プログラム・オブジェクトの処理時には、プログラム所有者とプログラム・ユーザーの両方のユーザー・プロファイルが使用されます。両方のユーザー・プロファイルの中のオブジェクト権限の集合セットが、プログラムの処理中にオブジェクトを検出してアクセスするために使用されます。所有側のユーザー・プロファイルのグループ・プロファイルの権限は、実行されるプログラム・オブジェクトの権限には含まれません。

上

---

## **権限 (AUT)**

オブジェクトに対する特定権限がないユーザー、権限リスト上にないユーザー、またはそのグループにオブジェクトに対する特定権限がないユーザーに許可される権限を指定します。

### **\*LIBCRTAUT**

オブジェクトに対する共通権限はターゲット・ライブラリー（作成されたオブジェクトが入れられるライブラリー）のCRTAUTキーワードからとられます。この値は、オブジェクトの作成時に決定されます。オブジェクトの作成後にライブラリーのCRTAUT値が変更された場合には、この新規の値はライブラリー中の既存のどのオブジェクトにも影響しません。

**\*ALL** 所有者に限定されるか、あるいは権限リスト管理権限によって制御される操作を除いて、オブジェクトでのすべての操作に対する権限が提供されます。どのユーザーも、オブジェクトの存在を制御し、その機密保護を指定し、変更し、また、その所有権の転送も含めてオブジェクトに対する基本操作を実行することができます。

### **\*CHANGE**

すべてのデータ権限、および所有者に限定されるか、あるいは権限リスト管理権限によって制御される操作を除いて、オブジェクトでのすべての操作を実行する権限が提供されます。オブジェクトを変更し、それに基本操作を実行することができます。

**\*USE** オブジェクト操作権、読み取り権限、およびモジュール・オブジェクトのバインドなど、オブジェクトへの基本読み取り専用操作に対する権限が提供されます。特定権限のないユーザーは、オブジェクトを変更することができません。

### **\*EXCLUDE**

特殊権限のないユーザーはオブジェクトにアクセスすることができません。

**name** オブジェクトの追加先となるユーザーと権限で構成された権限リストの名前を指定してください。オブジェクトはこの権限リストによって保護され、そのオブジェクトの共通権限は\*AUTLに設定されます。権限リストは、コマンドが出された時にシステム上に存在していなければなりません。

上

---

## **ターゲット・リリース (TGTRLS)**

ユーザーが作成中のオブジェクトを使用したいオペレーティング・システムのリリースを指定します。

\*CURRENTおよび\*PRVの値について示される例の中、および リリース・レベル の値を指定する場合には、形式VXRXXMXを使用してリリースが指定されます。ここで、VXはバージョン、RXはリリース、そしてMXはモディフィケーション・レベルです。たとえば、V4R5M0は、バージョン4、リリース5、モディフィケーション・レベル0です。

#### **\*CURRENT**

オブジェクトは、ユーザーのシステムで現在実行されているオペレーティング・システムのリリースで使用されます。例えば、V5R5M0がシステムで実行されている場合に、\*CURRENTは、ユーザーはオブジェクトをV5R5M0の導入されたシステムで使いたいことを意味します。ユーザーは、また、このリリース以降のオペレーティング・システムが導入されたシステムでもオブジェクトを使用することができます。

**\*PRV** オブジェクトは、オペレーティング・システムの前のリリース（モディフィケーション・レベル0）で使用されます。例えば、V5R5M0がシステムで実行されている場合に、\*PRVは、ユーザーはオブジェクトをV5R5M0の導入されたシステムで使いたいことを意味します。ユーザーは、また、このリリース以降のオペレーティング・システムが導入されたシステムでもオブジェクトを使用することができます。

#### **リリース・レベル**

リリースを形式VXRXXMXで指定します。オブジェクトは、指定したリリースのシステムでか、あるいはこのリリース以降のオペレーティング・システムが導入されたシステムで使用することができます。

有効な値は、現行バージョン、リリース、およびモディフィケーション・レベルによって異なります。これらは各新規リリースごとに変更されます。このコマンドによってサポートされる最も初期のリリース・レベルより前のリリース・レベルを指定した場合には、エラー・メッセージが送られます。

上

---

## **パフォーマンス収集の使用可能化 (ENBPFRCOL)**

オブジェクト内でパフォーマンス測定コードを生成すべきかどうかを指定します。収集されたデータは、システム・パフォーマンス・ツールによって使用し、アプリケーションのパフォーマンスのプロファイルを作成することができます。作成されたオブジェクト内でパフォーマンス測定コードを生成すると、オブジェクトのサイズがわずかに増えるため、パフォーマンスに影響が及ぶ可能性があります。

注: V5R5M0 以降では、このパラメーターは、作成されたオブジェクトに対して作用しなくなります。このパラメーターは、V5R5M0 より前のリリースとの互換性を保つために用意されています。

**\*PEP** パフォーマンス統計は、プログラム・エントリー・プロシージャーの入り口と出口でのみ収集されます。この値は、アプリケーションの全般的なパフォーマンス情報を収集したい場合に選択します。

#### **\*ENTRYEXIT \*NONLEAF**

パフォーマンス統計は、プログラム・オブジェクトの全プロシージャー（リーフ・プロシージャーを除く）の入り口と出口で収集されます。これにはプログラムPEPルーチンが含まれます。

この選択項目は、アプリケーション内の他のルーチンを呼び出すルーチンに関する情報だけを取り込みたい場合に有用です。

#### **\*ENTRYEXIT \*ALLPRC**

パフォーマンス統計は、プログラム・オブジェクトの全プロシージャ（リーフ・プロシージャを含む）の入り口と出口で収集されます。これには、プログラムPEPルーチンが含まれます。

この選択項目は、すべてのルーチンに関する情報を取り込みたい場合に有用です。このオプションは、ユーザーのアプリケーションから呼び出されるすべてのプログラム・オブジェクトが、\*PEP、\*ENTRYEXIT、\*FULLのいずれかのオプションを使用して作成されたことが判明している場合に使用します。それ以外の場合、パフォーマンス測定に対して使用可能にならない他のプログラム・オブジェクトがユーザーのアプリケーションから呼び出されると、パフォーマンス・ツールは、そのプログラム・オブジェクトによる資源の使用についてユーザーのアプリケーションに任せます。このため、資源が実際に使用されている場所を判別するのが困難になることがあります。

#### **\*FULL \*NONLEAF**

パフォーマンス統計は、リーフ・プロシージャではないすべてのプロシージャの入り口と出口で収集されます。また、外部プロシージャに対する個々の呼び出しの前および後にも統計が収集されます。

#### **\*FULL \*ALLPRC**

パフォーマンス統計は、リーフ・プロシージャを含むすべてのプロシージャの入り口と出口で収集されます。また、外部プロシージャに対する個々の呼び出しの前および後にも統計が収集されます。

このオプションは、作成時に\*PEP、\*ENTRYEXIT、\*FULLのいずれも使用されなかった他のプログラム・オブジェクトをユーザーのアプリケーションから呼び出そうと考えている場合に使用します。このオプションを使用すると、パフォーマンス・ツールは、ユーザーのアプリケーションによって使用されている資源と、そのアプリケーションから呼び出されたプログラム・オブジェクトによって使用されている資源を、（そのプログラム・オブジェクトがパフォーマンス測定に対して使用可能にならない場合でも）区別できるようになります。このオプションを使用すると、効率は最も悪くなりますが、アプリケーション内の各種プログラム・オブジェクトを選択的に分析できるようになります。

上

---

## **パフォーマンス・オプション (PFROPT)**

パフォーマンスを高めるために使用可能な各種のオプションを指定します。それらは、1つまたは複数のブランクで区切り、任意の順序で指定できます。1つのオプションが複数回指定されたり、2つのオプションが競合する場合には、最後に指定されたものが使用されます。

このパラメーターには2個の値を指定できます。

#### **\*SETFPCA**

浮動小数点計算に対するANSIのセマンティクスを達成するために、コンパイラーに浮動小数点計算属性を設定させます。

#### **\*NOSETFPCA**

計算属性は設定されません。このオプションは、作成中のオブジェクトに浮動小数点計算が含まれない場合にのみ使用してください。

#### **\*NOSTRONLY**

コンパイラーは文字列を書き込み可能メモリーに入れる必要があることを指定します。

### **\*STRDONLY**

コンパイラーは文字列を読み取り専用メモリーに入れることができることを指定します。

上

---

## **プロファイル作成データ (PRFDTA)**

プログラム・オブジェクトのプログラム・プロファイル作成データ属性を指定します。プログラム・プロファイル作成は、プロシージャおよびプロシージャ内のコードを統計データ（プロファイル作成データ）に基づいて配列し直すために使用される拡張最適化手法です。

### **\*NOCOL**

プログラム・オブジェクトは、プロファイル作成データの収集に使用できません。

**\*COL** プログラム・オブジェクトは、プロファイル作成データの収集に使用できます。**\*COL**は、最適化レベルが30以上である場合にのみ指定することができます。

上

---

## **テラスペース・オプション (TERASPACE)**

テラスペース・ストレージの処理にプログラム・オブジェクトを使用可能にするかどうかを指定します。これには、プログラム・オブジェクトによって割り振られるテラスペース・ストレージや、他のテラスペース対応プログラムおよびサービス・プログラムのオブジェクトから渡されたパラメーターによって割り振られるテラスペース・ストレージが含まれます。

### **単一値**

**\*NO** プログラム・オブジェクトは、テラスペースから割り振られたストレージのアドレス指定の処理に使用できません。

注: V5R5M0以降では、テラスペースから割り振られたストレージのアドレス指定の処理に、すべてのプログラム・オブジェクトが使用できます。ただし、**\*NO**が指定された場合には、以下にリストされているコンパイラー機能が使用できなくなります。

### **要素1: テラスペース使用可能**

**\*YES** プログラム・オブジェクトは、他のテラスペース対応プログラムやサービス・プログラムのオブジェクトから渡されたパラメーターも含めて、テラスペースから割り振られたストレージのアドレス指定の処理に使用できます。さらに、以下のコンパイラー機能も使用できます。

- テラスペース・ストレージへのアクセスに使用される8バイト・ポインターを作成できるように、ポインターは\_\_ptr64で修飾できます。
- テラスペース・ストレージ・モデルは、STGMDL(\*TERASPACE)コンパイラー・オプションを使用して指定できます。
- LLP64データ・モデルは、DTAMDLL(\*LLP64)コンパイラー・オプションまたは#pragma datamodel(llp64)ディレクティブを使用して指定できます。
- ポインター差分演算は、ptrdiff\_tの結果ではなく、signed long longの結果を戻します。

### **要素2: TERASPACE INTERFACESの使用**

### **\*NOTSIFC**

プログラム・オブジェクトは、デフォルトで、ストレージ機能の非テラスペース・バージョンを使用するようになります。

### **\*TSIFC**

プログラム・オブジェクトは、デフォルトで、ストレージ機能のテラスペース・バージョンを使用するようになります。コンパイラーは、マクロ変数\_\_TERASPACE\_\_を定義することになります。

上

---

## **ストレージ・モデル (STGMDL)**

作成されたオブジェクトで使用する記憶域のタイプを指定します。

### **\*SNGLVL**

作成されたオブジェクトは、単一レベルの記憶域を使用します。

### **\*TERASPACE**

作成されたオブジェクトは、テラスペース記憶域を使用します。

上

---

## **データ・モデル (DTAMD)**

INT, LONG, POINTERとして宣言される変数のサイズ（バイト数）を指定します。

**\*P128** INT, LONG, POINTERのサイズがそれぞれ4, 4, 16となるようにします。

### **\*LLP64**

INT, LONG, POINTERのサイズがそれぞれ4, 4, 8となるようにします。コンパイラーは、マクロ\_\_LLP64\_IFC\_\_を定義することになります。

上

---

## **パック構造 (PACKSTRUCT)**

構造のメンバーに使用する位置合わせ境界を指定します。

### **\*NATURAL**

構造メンバーはその自然境界で位置合わせされます。たとえば、短形式整数は2バイトで位置合わせされます。16バイト・ポインターは、常に16バイト境界で位置合わせされます。

- 1** 構造メンバーを1バイトの位置合わせでパックします。
- 2** 構造メンバーを2バイトの位置合わせでパックします。
- 4** 構造メンバーを4バイトの位置合わせでパックします。
- 8** 構造メンバーを8バイトの位置合わせでパックします。
- 16** 構造メンバーを16バイトの位置合わせでパックします。

上

---

## Enumサイズ (ENUM)

コンパイラーが列挙型を表すために使用するバイト数を指定します。

### \*SMALL

すべてのENUM変数を、値の範囲を表すことができる最も小さいサイズにします。

- 1 全てのENUM変数を 1 バイトにします。
- 2 全てのENUM変数を 2 バイトにします。
- 4 全てのENUM変数を 4 バイトにします。

**\*INT** ANSI標準のENUMサイズである 4 バイトを使用します。

上

---

## 依存関係情報 (MAKEDEP)

依存関係情報をファイルの中に生成するかどうかを指定します。 この情報は、MAKE TOOLによって使用されます。

### \*NONE

依存関係情報は生成しません。

### *path-name*

依存関係情報を保管するストリーム・ファイルのパス名を指定します。

上

---

## 組み込みディレクトリー (INCDIR)

組み込みファイルを検出するためにコンパイラーが使用する検索パスに追加する 1 つまたは複数のディレクトリーを指定します。 INCDIRを使用すると、INCLUDE環境変数が上書きされます。

検索パスは、OPTIONキーワードで次のパラメーターを使用することにより、さらに変更することができます。

- \*INCDIRFIRSTまたは\*NOINCDIRFIRST
- \*SYSINCPATHまたは\*NOSYSINCPATH
- \*STDINCまたは\*NOSTDINC

### \*NONE

変更されなければ、省略時のシステム組み込みディレクトリーおよびソース・ディレクトリーからユーザー組み込みファイルが検索されます。

### *path-name*

組み込みファイルの検索場所となるディレクトリーのパス名を指定してください。最大32個のディレクトリーを指定できます。指定したディレクトリーに加えて、ソース・ディレクトリーからもユーザー組み込みファイルが検索されます。

上



---

## コンパイラー・サービスOPT (CSOPT)

1つまたは複数のコンパイラー・サービス・オプションを指定します。このパラメーターにより、IBMはリリース相互間で切り替え可能なコンパイラーを提供することができます。

### \*NONE

コンパイラー・サービス・オプションは選択されません。

**文字値** モジュール・オブジェクトの作成時に使用するコンパイラー・サービス・オプションを含む文字ストリングを指定してください。有効な文字列は、PTFカバー・レターまたはリリース情報に記述されます。

上

---

## ライセンス内部コードOPT (LICOPT)

1つまたは複数のライセンス内部コード・コンパイル時オプションを指定します。このパラメーターは、個別のコンパイル時オプションを選択できるようにするもので、選択された各タイプのコンパイラー・オプションの潜在的な長所と短所を理解している上級プログラマーを対象としています。

### \*NONE

コンパイル時オプションは選択されません。

**文字値** 選択されたライセンス内部コード・コンパイル時オプションが、モジュール・オブジェクトの作成時に使用されます。特定のオプションでは、作成されたモジュール・オブジェクトをデバッグする能力が低下することがあります。

上

---

## 省略時の文字タイプ (DFTCHAR)

CHARデータ・タイプの省略時の符号を指定します。

### \*UNSIGNED

省略時のCHARタイプを符号なしとします。

### \*SIGNED

省略時のCHARタイプを符号付きとします。

上

---

## ターゲットCCSID (TGTCSSID)

結果のプログラム・オブジェクトに保管されるデータを記述するために使用されるターゲット・コード化文字セットID (CCSID)を指定します。

### \*SOURCE

ルート・ソース・ファイルのCCSIDが使用されます。

**\*JOB** 現行ジョブのCCSIDが使用されます。

**\*HEX** CCSID 65535が使用されます。これは、文字データはバイナリー・データとして取り扱われ、変換されないことを指示します。

使用するCCSIDを指定します。

上

---

## 10進浮動丸めモード (DECFLTRND)

定数10進浮動小数点式の評価のコンパイル時丸めモードを指定します。このオプションは、実行時10進浮動小数点丸めモード(setca組み込み関数を使用して設定される)には影響しません。

### **\*HALFEVEN**

最も近い値に丸めます。最も近い値が2つある場合、偶数を選択します。たとえば、5.22の場合は5.2に丸め、5.67の場合は5.7に丸め、5.55の場合は5.6に丸め、5.65の場合は5.6に丸め、-5.23の場合は-5.2に丸め、-5.55の場合は-5.6に丸めます。

### **\*DOWN**

ゼロに近い方の数字に丸めるか、または結果の数字を切り捨てます。たとえば、5.22の場合は5.2に丸め、5.67の場合は5.6に丸め、5.55の場合は5.5に丸め、5.65の場合は5.6に丸め、-5.23の場合は-5.2に丸め、-5.55の場合は-5.5に丸めます。

**\*UP** ゼロから遠い方の数字に丸めます。たとえば、5.22の場合は5.3に丸め、5.67の場合は5.7に丸め、5.55の場合は5.6に丸め、5.65の場合は5.7に丸め、-5.23の場合は-5.3に丸め、-5.55の場合は-5.6に丸めます。

### **\*HALFUP**

最も近い値に丸めます。最も近い値が2つある場合、ゼロから遠い方の数字に丸めます。たとえば、5.22の場合は5.2に丸め、5.67の場合は5.7に丸め、5.55の場合は5.6に丸め、5.65の場合は5.7に丸め、-5.23の場合は-5.2に丸め、-5.55の場合は-5.6に丸めます。

### **\*HALFDOWN**

最も近い値に丸めます。最も近い値が2つある場合、ゼロから近い方の数字に丸めます。たとえば、5.22の場合は5.2に丸め、5.67の場合は5.7に丸め、5.55の場合は5.5に丸め、5.65の場合は5.6に丸め、-5.23の場合は-5.2に丸め、-5.55の場合は-5.5に丸めます。

### **\*FLOOR**

小さい数字になるように丸めます。たとえば、5.22の場合は5.2に丸め、5.67の場合は5.6に丸め、5.55の場合は5.5に丸め、5.65の場合は5.6に丸め、-5.23の場合は-5.3に丸め、-5.55の場合は-5.6に丸めます。

### **\*CEILING**

大きい数字になるように丸めます。たとえば、5.22の場合は5.3に丸め、5.67の場合は5.7に丸め、5.55の場合は5.6に丸め、5.65の場合は5.7に丸め、-5.23の場合は-5.2に丸め、-5.55の場合は-5.5に丸めます。

上

---

## 例

### 例1:ソース・ファイル・メンバーからのプログラムの作成

```
CRTBNDC  PGM(QTEMP/TEST) SRCFILE(QTEMP/QCSRC)
```



このコマンドは、プログラム・オブジェクトTESTをライブラリーQTEMP内に作成します。Cソースは、ライブラリーQTEMPにあるソース・ファイルQCSRCのメンバーTEST内に指定されます。

#### 例2:ストリーム・ファイルからのプログラムの作成

```
CRTBNDC  PGM(QTEMP/TEST) SRCSTMF('/tmp/mySrc.c')
```

このコマンドは、プログラム・オブジェクトTESTをライブラリーQTEMP内に作成します。Cソースは、ストリーム・ファイル/tmp/mySrc.c内に指定されます。

#### 例3:コンパイル・リストのコンパイルおよび生成

```
CRTBNDC  PGM(TEST) OUTPUT(*PRINT) OPTION(*XREF)
```

このコマンドは、プログラム・オブジェクトTESTをジョブの現行ライブラリー内に作成しますが、現行ライブラリーが定義されていない場合は、ライブラリーQGPL内に作成します。Cソースは、ソース・ファイルQCSRCのメンバーTEST内に指定されると想定されます（これは、ライブラリー・リストを使用して指定されます）。コンパイラー・リストが入ったスプール・ファイルが生成されます。コンパイラー・リスト・ファイルには、相互参照リストが格納されます。

#### 例4:すべてのデバッグ・ビューを備えたプログラムの作成

```
CRTBNDC  PGM(QTEMP/TEST) SRCFILE(QTEMP/QCSRC)
          SRCMBR(TESTER) DBGVIEW(*ALL)
```

このコマンドは、デバッグ時に使用されるステートメント、ソース、およびリストの各ビューを備えたプログラム・オブジェクトTESTをライブラリーQTEMP内に作成します。Cソースは、ライブラリーQTEMPにあるソース・ファイルQCSRCのメンバーTESTER内に指定されます。

#### 例5:テラスペース対応プログラムの作成

```
CRTBNDC  PGM(QTEMP/TEST) SRCFILE(QTEMP/QCSRC)
          TERASPACE(*YES *TSIFC) STGMDL(*TERASPACE)
          DTAMDLL(*LLP64)
```

このコマンドは、プログラム・オブジェクトTESTをライブラリーQTEMP内に作成します。Cソースは、ライブラリーQTEMPにあるソース・ファイルQCSRCのメンバーTEST内に指定されます。このモジュールは、テラスペース・ストレージを使用する場合に使用可能にされます。ヒープ・スペース、自動ストレージ、および静的ストレージは、テラスペース・ストレージ内に入れられます。デフォルト・ポインター・サイズは8バイトになります。

上

---

## エラー・メッセージ

### \*ESCAPE メッセージ

#### CZM1613

コンパイルに失敗しました。

上



## バインドCOBOL PGMの作成 (CRTBNDCBL)

実行可能場所: すべての環境 (\*ALL)

スレッド・セーフ: いいえ

パラメーター

例

エラー・メッセージ

CRTBNDCBLコマンドはILE COBOLソース・プログラムを1ステップで実行可能プログラムにコンパイルします。このコマンドは、対話式、バッチ・モード、またはCLプログラムで使用することができます。

CRTBNDCBLコマンドに指定するすべてのオブジェクト名は、サーバーの命名規則に従っていなければなりません。

CRTBNDCBLコマンドのパラメーターの説明は、次の通りです。

上

### パラメーター

キーワード	記述	選択項目	ノーツ
PGM	プログラム	修飾オブジェクト名	オプションル, 定位置 1
	修飾子 1: プログラム	名前, <u>*PGMID</u>	
	修飾子 2: ライブラリー	名前, <u>*CURLIB</u>	
SRCFILE	ソース・ファイル	修飾オブジェクト名	オプションル, 定位置 2
	修飾子 1: ソース・ファイル	名前, <u>QCBLLSRC</u>	
	修飾子 2: ライブラリー	名前, <u>*LIBL</u> , <u>*CURLIB</u>	
SRCMBR	ソース・メンバー	名前, <u>*PGM</u>	オプションル, 定位置 3
SRCSTMF	ソース・ストリーム・ファイル	パス名	オプションル
GENLVL	生成重大度レベル	0-30, <u>30</u>	オプションル
TEXT	テキスト記述	文字値, <u>*SRCMBRTXT</u> , <u>*BLANK</u>	オプションル
OUTPUT	出力	<u>*PRINT</u> , <u>*NONE</u>	オプションル, 定位置 4

キーワード	記述	選択項目	ノート
<b>OPTION</b>	コンパイラ・オプション	値 (最大 50 回の繰り返し): *SOURCE, *NOSOURCE, *SRC, *NOSRC, *NOXREF, *XREF, *GEN, *NOGEN, *NOSEQUENCE, *SEQUENCE, *NOVBSUM, *VBSUM, *NONUMBER, *NUMBER, *LINENUMBER, *NOMAP, *MAP, *NOOPTIONS, *OPTIONS, *QUOTE, *APOST, *NOSECLVL, *SECLVL, *PRTCORR, *NOPRTCORR, *MONOPRC, *NOMONOPRC, *RANGE, *NORANGE, *NOUNREF, *UNREF, *NOSYNC, *SYNC, *NOCRTF, *CRTF, *NODUPKEYCHK, *DUPKEYCHK, *NOINZDLT, *INZDLT, *NOBLK, *BLK, *STDINZ, *NOSTDINZ, *STDINZHEX00, *NODDSFILLER, *DDSFILLER, *NOIMBEDERR, *IMBEDERR, *STDTRUNC, *NOSTDTRUNC, *NOCHGPOSSGN, *CHGPOSSGN, *NOEVENTF, *EVENTF, *MONOPIC, *NOMONOPIC, *NOCRTARKIDX, *CRTARKIDX	オプション, 定位置 5
<b>CVTOPT</b>	変換オプション	値 (最大 8 回の繰り返し): *NOVARCHAR, *VARCHAR, *NODATETIME, *DATETIME, *NOPICXGRAPHIC, *PICXGRAPHIC, *NOPICGGRAPHIC, *PICGGRAPHIC, *NOFLOAT, *FLOAT, *NODATE, *DATE, *NOTIME, *TIME, *NOTIMESTAMP, *TIMESTAMP, *NOCVTTODATE, *CVTTODATE, *NOPICNGRAPHIC, *PICNGRAPHIC	オプション
<b>MSGMT</b>	メッセージ限界	要素リスト	オプション
	要素 1: メッセージの数	0-9999, <u>*NOMAX</u>	
	要素 2: メッセージ限界重大度	0-30, <u>30</u>	
<b>DBGVIEW</b>	デバッグ・ビューのオプション	要素リスト	オプション
	要素 1: デバッグ・ビュー	<u>*STMT</u> , *SOURCE, *LIST, *ALL, *NONE	
	要素 2: リスト・ビューの圧縮	<u>*NOCOMPRESSDBG</u> , *COMPRESSDBG	
<b>OPTIMIZE</b>	最適化レベル	<u>*NONE</u> , *BASIC, *FULL	オプション

キーワード	記述	選択項目	ノート
FLAGSTD	FIPSフラグ付け	値 (最大 2 回の繰り返し): *NOFIPS, *MINIMUM, *INTERMEDIATE, *HIGH, *NOBSOLETE, *OBSOLETE	オプション
EXTDSPOPT	拡張表示オプション	値 (最大 3 回の繰り返し): *DFRWRT, *NODFRWRT, *UNDSPCHR, *NOUNDSPCHR, *ACCUPDALL, *ACCUPDNE	オプション
FLAG	フラグ重大度	0-99, <u>0</u>	オプション
REPLACE	プログラムの置き換え	*YES, *NO	オプション
USRPRF	ユーザー・プロファイル	*USER, *OWNER	オプション
AUT	権限	名前, *LIBCRTAUT, *ALL, *CHANGE, *USE, *EXCLUDE	オプション
LINKLIT	リンク・リテラル	*PGM, *PRC	オプション
SIMPLEPGM	単純プログラム	*YES, *NO	オプション
TGTRLS	ターゲット・リリース	単純名, *CURRENT, *PRV	オプション
SRTSEQ	ソート順序	単一値: *HEX, *JOB, *JOBRUN, *LANGIDUNQ, *LANGIDSHR その他の値: 修飾オブジェクト名	オプション
	修飾子 1: ソート順序	名前	
	修飾子 2: ライブラリー	名前, *LIBL, *CURLIB	
LANGID	言語ID	文字値, *JOBRUN, *JOB	オプション
ENBPFRCOL	パフォーマンス収集の活動化	単一値: *PEP その他の値: 要素リスト	オプション
	要素 1: 収集レベル	*FULL, *ENTRYEXIT	
BNDDIR	バインディング・ディレクトリー	単一値: *NONE その他の値 (最大 50 回の繰り返し): 修飾オブジェクト名	オプション
	修飾子 1: バインディング・ディレクトリー	名前	
	修飾子 2: ライブラリー	名前, *LIBL, *CURLIB, *USRLIBL	
ACTGRP	活動化グループ	名前, QILE, *NEW, *CALLER	オプション
PRFDTA	プロファイル・データ	*NOCOL, *COL	オプション
CCSID	コード化文字セットID	整数, *JOBRUN, *HEX, *JOB	オプション
NTLCCSID	国別CCSID	整数, 13488	オプション
ARITHMETIC	演算モード	*NOEXTEND, *EXTEND31, *EXTEND63	オプション
NTLPADCHAR	PADDING CHARACTER	要素リスト	オプション
	要素 1: SINGLE BYTE TO NATIONAL	文字値, *DEFAULT	
	要素 2: DOUBLE BYTE TO NATIONAL	文字値, *DEFAULT	
	要素 3: NATIONAL TO NATIONAL	文字値, *DEFAULT	
LICOPT	LICENSED INTERNAL CODE OPTIONS	文字値	オプション

キーワード	記述	選択項目	ノート
INCDIR	ディレクトリー組み込み	値 (最大 32 回の繰り返し): パス名, <b>*NONE</b>	オプション
PGMINFO	PGMインターフェース情報	要素リスト	オプション
	要素 1: 生成	<b>*NO</b> , *PCML	
	要素 2: 位置	<b>*STMF</b> , *MODULE, *ALL	
INFOSTMF	プログラム・インターフェース・ストリーム・ファイル	パス名	オプション

上

## プログラム (PGM)

作成中のプログラム・オブジェクトのプログラム名およびライブラリー名を指定します。プログラム名およびライブラリー名は、サーバーの命名規則に従っていなければなりません。指定できる値は次の通りです。

### **\*PGMID**

コンパイルされたプログラムの名前は、ILE COBOLソース・プログラムのPROGRAM-ID段落から取り出されます。SIMPLEPGM(\*NO)が指定されている場合には、コンパイルされたプログラム・オブジェクトの名前は 一連のソース・プログラム（単一のソース・ファイル・メンバーの複数のコンパイル単位）の最初のILE COBOLソース・プログラムのPROGRAM-ID段落から取り出されます。

### プログラム名

コンパイルされたILE COBOLプログラムを識別する名前を入力してください。このパラメーターにプログラム名を指定して 一連のソース・プログラム をコンパイルし、SIMPLEPGM(\*YES)が指定されている場合には、一連の最初のプログラムがこの名前を使用します。他のプログラムは対応するILE COBOLソース・プログラムのPROGRAM-ID段落で指定された名前を使用します。

指定可能なライブラリー値は、次の通りです。

### **\*CURLIB**

作成されたプログラム・オブジェクトは現行ライブラリーの中に記憶されます。現行ライブラリーとしてライブラリーが割り当てられていない場合には、QGPLが使用されます。

### ライブラリー名

作成されたプログラム・オブジェクトを記憶するライブラリーの名前を入力してください。

上

## ソース・ファイル (SRCFILE)

コンパイルするILE COBOLソース・コードが入っているソース・ファイルおよびライブラリーの名前を指定します。このソース・ファイルのレコード長は92でなければなりません。指定できる値は次の通りです。

### **QCBLLSRC**

ソース・ファイルQCBLLSRCにコンパイルするILE COBOLのソースが入っていることを指定します。

### ソース・ファイル名

コンパイルするILE COBOLのソースが入っているソース・ファイルの名前を入力してください。

指定可能なライブラリー値は、次の通りです。

**\*LIBL** ソース・ファイルが入っているライブラリーを見つけるためにライブラリー・リストが探索されます。

**\*CURLIB**

現行ライブラリーが使用されます。現行ライブラリーとしてライブラリーが割り当てられていない場合には、QGPLが使用されます。

**ライブラリー名**

ソース・ファイルが入っているライブラリーの名前を入力してください。

上

---

## ソース・メンバー (SRCMBR)

コンパイルするILE COBOLソース・コードが入っているメンバーの名前を指定します。SRCFILEパラメーターで参照しているソース・ファイルがデータベース・ファイルである場合にだけ、このパラメーターを指定することができます。指定できる値は次の通りです。

**\*PGM** PGMパラメーターで指定されたプログラム名と同じ名前をもつソース・ファイル・メンバーが使用されます。

PGMパラメーターにプログラム名を指定しない場合には、コンパイラーはデータベース・ソース・ファイルの最初のソース・メンバーを使用します。

**ソース・ファイル・メンバー名**

ILE COBOLソース・コードが入っているメンバーの名前を入力してください。

上

---

## ソース・ストリーム・ファイル (SRCSTMF)

コンパイルするILE COBOLソース・コードが入っているストリーム・ファイルのパス名を指定します。

このパス名は絶対的または相対的に修飾することができます。絶対パス名は'/'で始まり、相対パス名は'/'以外の文字で始まります。

絶対修飾の場合には、パス名は完全です。相対修飾の場合には、パス名は、パス名にジョブの現行作業ディレクトリーを付加することによって完了します。

SRCMBRおよびSRCFILEパラメーターをSRCSTMFパラメーターと一緒に指定することはできません。

上

---

## 生成重大度レベル (GENLVL)

モジュール・オブジェクトを作成するかどうかを決定する重大度レベルを指定します。重大度レベルはコンパイル時に生成されるメッセージの重大度レベルと対応しています。このパラメーターは、ソース・ファイル・メンバーの各コンパイル単位に個別に適用されます。前のコンパイル単位が正常に実行されない場合でも、ソース・ファイル・メンバー中のその他のコンパイル単位はコンパイルされることになります。

指定できる値は次の通りです。

**30** 30以上の重大度レベルのエラーが起こった場合には、プログラム・オブジェクトは作成されません。

#### 重大度レベル

プログラム・オブジェクトを作成するかどうかを決定するために使用したい重大度レベルを、0-30の範囲の1桁または2桁の数字で指定します。この重大度レベル以上の重大度レベルのエラーが起こった場合には、プログラム・オブジェクトは作成されません。

上

---

## テキスト記述 (TEXT)

プログラムおよびその機能を簡単に説明するテキストを入力することができます。

#### **\*SRCMBRTXT**

プログラム・オブジェクトの記述には、ILE COBOLソース・コードが入っているデータベース・ファイル・メンバーを記述するのと同じテキストが使用されます。情報源が装置ファイルまたはインライン・ファイルからの場合には、\*SRCMBRTXTの指定は\*BLANKの指定と同じ効果となります。

#### **\*BLANK**

テキストは指定されません。

#### テキスト記述

プログラムおよびその機能を簡単に説明するテキストを入力してください。テキストは最大50桁の長さのSBCS文字にすることができ、単一引用符で囲む必要があります。単一引用符は50桁の文字ストリングの一部にはなりません。

上

---

## 出力 (OUTPUT)

コンパイル・リストを生成するかどうかを指定します。指定できる値は次の通りです。

#### **\*PRINT**

コンパイル・リストが生成されます。メンバーがコンパイルされる場合には、出力ファイルはメンバーと同じ名前になります。ストリーム・ファイルがコンパイルされて、PGMパラメーターに\*PGMIDが指定されている場合には、出力ファイルはCOBOLPGM00という名前になります。そうでない場合には、プログラムと同じ名前になります。

#### **\*NONE**

コンパイル・リストは生成されません。

上

---

## コンパイラー・オプション (OPTION)

ILE COBOLソース・コードのコンパイル時に使用するオプションを指定します。



ILE COBOLソース・プログラムのPROCESSステートメントで指定されたオプションによって、OPTIONパラメーターの対応するオプションが指定変更されます。

OPTIONパラメーターに指定できる値は、次の通りです。

**\*SOURCEまたは\*SRC**

コンパイラーは、ILE COBOLソース・プログラムとすべてのコンパイル時エラー・メッセージから構成されるソース・リストを作成します。

**\*NOSOURCEまたは\*NOSRC**

コンパイラーはリストのソース・パートを作成しません。ソース・リストが不要である場合には、コンパイル時間を短くすることができるので、このオプションを使用してください。

**\*NOXREF**

コンパイラーはILE COBOLソース・プログラムの相互参照表を作成しません。

**\*XREF**

コンパイラーはILE COBOLソース・プログラムの相互参照表を作成します。

**\*GEN** コンパイラーは、ILE COBOLソースのコンパイル後に、プログラム・オブジェクトを作成します。

**\*NOGEN**

コンパイラーは、ILE COBOLソース・プログラムのコンパイル後に、プログラム・オブジェクトを作成しません。エラー・メッセージまたはリストだけが必要な場合には、このオプションを指定することができます。

**\*NOSEQUENCE**

参照番号のシーケンス・エラーは検査されません。

**\*SEQUENCE**

参照番号のシーケンス・エラーが検査されます。\*LINENUMBERオプションを指定した場合には、シーケンス・エラーは起こりません。

**\*NOVBSUM**

動詞使用カウントは印刷されません。

**\*VBSUM**

動詞使用カウントが印刷されます。

**\*NONUMBER**

ソース・ファイルの順序番号が参照番号に使用されます。

**\*NUMBER**

ユーザー提供の順序番号（1-6桁目）が参照番号に使用されます。

**\*LINENUMBER**

コンパイラーによって作成された順序番号が参照番号に使用されます。このオプションはILE COBOLプログラムのソース・コードとCOPYステートメントで導入されたソース・コードを結合して、1つの連続番号順にします。FIPS（米国情報処理規格）フラグ付けまたはSAAフラグ付けを指定する場合には、このオプションを使用してください。

**\*NOMAP**

コンパイラーはデータ部マップをリストしません。

**\*MAP** コンパイラーはデータ部マップをリストします。

**\*NOOPTIONS**

このコンパイルに有効となっているオプションがリストされません。

## **\*OPTIONS**

このコンパイルに有効となっているオプションがリストされます。

## **\*QUOTE**

区切り文字引用符(")が非数値リテラル、16進リテラル、およびブール・リテラルに使用されることを指定します。また、このオプションは、表意定数QUOTEの値が引用符のEBCDIC値をもつことも指定します。

## **\*APOST**

区切り文字アポストロフィ(')が非数値リテラル、16進リテラル、およびブール・リテラルに使用されることを指定します。また、このオプションは、表意定数QUOTEの値がアポストロフィのEBCDIC値をもつことも指定します。

## **\*NOSECLVL**

このコンパイルで第2レベル・メッセージ・テキストはリストされません。

## **\*SECLVL**

このコンパイルで、コンパイル・リストのメッセージ・セクションに、第1レベル・エラー・テキストと一緒に第2レベル・メッセージ・テキストがリストされます。

## **\*PRTCORR**

CORRESPONDING句の使用の結果として基本項目が含まれることを示すコメント行がコンパイル・リストの中に挿入されます。

## **\*NOPRTCORR**

CORRESPONDING句が使用された時に、コメント行はコンパイル・リストの中に挿入されません。

## **\*MONOPRC**

PROGRAM-ID段落、CALL、CANCEL、またはSET ENTRYステートメント、およびEND PROGRAMヘッダーで見つかったプログラム名（リテラルまたは語句）は、すべて大文字（単一シフト）に変換され、プログラム名作成規則が強制的に適用されます。

## **\*NOMONOPRC**

PROGRAM-ID段落、CALL、CANCEL、またはSET ENTRYステートメント、およびEND PROGRAMヘッダーで見つかったプログラム名（リテラルまたは語句）は、すべて大文字（非単一シフト）に変換されず、プログラム名作成規則は強制されません。このオプションによって、標準COBOLでは使用できない特殊文字を、CALL行き先に使用することができます。

## **\*RANGE**

実行時に、添え字は正しい範囲内にあることを確認するために検査されますが、指標の範囲は検査されません。参照変更およびコンパイラ生成のサブストリング命令も検査されます。

形式が正しいこと、さらに正しい日付、時刻、またはタイム・スタンプが表示されていることを確認するために、日付－時刻項目の内容を検査します。

## **\*NORANGE**

実行時に範囲は検査されません。

注: \*RANGEオプションによって添え字範囲を検査するコードが生成されます。たとえば、20 要素の配列の21番目の要素をアクセスしようとしていないことが確認されます。

\*NORANGEオプションは添え字の範囲を検査するコードを生成しません。結果として、

\*NORANGEオプションによって高速の実行コードが作成されます。

## **\*NOUNREF**

参照されていないデータ項目はコンパイル済みモジュールの中に含まれません。これによって、使用記憶域の量が減少し、より大きなプログラムをコンパイルすることができます。\*NOUNREFオプ

ションを選択している時には、デバッグ中に、参照されていないデータ項目を表示したり割り当てたりすることはできません。参照されていないデータ項目は、OPTION (\*XREF)を指定して生成された相互参照表にはまだ表示されます。

#### **\*UNREF**

参照されていないデータ項目は、コンパイル済みプログラムに含まれます。

#### **\*NOSYNC**

SYNCHRONIZED文節は構文検査のみが行なわれます。

#### **\*SYNC**

SYNCHRONIZED文節がコンパイラによってコンパイルされます。SYNCHRONIZED文節によって、データ項目の位置は、右端（最小有効文字）が自然の記憶域境界になるように位置合わせされます。自然の記憶域境界は、記憶するデータの長さおよびタイプによって、記憶域の中で次に最も近い4バイト境界、8バイト境界、または16バイト境界となります。この位置合わせを達成するために同期された項目に隣接する特別の記憶域が予約されます。SYNCHRONIZEDと記述された各基本データ項目は、そのデータ記憶域割り当てに対応する自然の記憶域境界に位置合わせされます。

#### **\*NOCRTF**

OPEN命令の実行時に使用できないディスク・ファイルは、動的には作成されません。

#### **\*CRTF**

OPEN命令の実行時に使用できないディスク・ファイルが、動的に作成されます。

注: 動的に作成するファイルの最大レコード長は32766です。\*CRTFオプションが指定された場合でも、索引付きファイルは動的に作成されません。

#### **\*NODUPKEYCHK**

INDEXEDファイルの重複した基本レコード・キーと代替レコード・キーを検査しません。

#### **\*DUPKEYCHK**

INDEXEDファイルの重複した基本レコード・キーと代替レコード・キーを検査します。

#### **\*NOINZDLT**

順次アクセスによる相対ファイルは、ファイルがOUTPUT用にオープンされた場合には、CLOSE命令時に削除済みレコードを初期設定しません。レコード境界はOPEN OUTPUT時に書き出されたレコード数によって決まります。次のOPEN命令によって、レコード境界までに限ってアクセスすることができます。

#### **\*INZDLT**

順次アクセスによる相対ファイルは、ファイルがOUTPUT用にオープンされた場合には、CLOSE命令時に削除済みレコードを初期設定します。ファイルの活動レコードは影響を受けません。レコード境界は、次のOPEN命令のファイル・サイズとして定義されます。

#### **\*NOBLK**

コンパイラは、STARTステートメントによらないSEQUENTIALアクセス・ファイルのブロック化だけを許します。BLOCK CONTAINS文節は、指定された場合には、テープ・ファイルの場合を除いて無視されます。

**\*BLK** \*BLKが使用されて、BLOCK CONTAINS文節が指定された時には、コンパイラは、STARTステートメントによるDYNAMICアクセス・ファイルおよびSEQUENTIALアクセス・ファイルのブロック化を許します。出力操作でオープンされたRELATIVEファイルのブロック化は許されません。BLOCK CONTAINS文節はブロック化するレコードの数を制御します。

\*BLKが使用され、BLOCK CONTAINS文節が指定されていない時には、コンパイラーはSTARTステートメントによらないSEQUENTIALアクセス・ファイルのブロック化だけを許します。オペレーティング・システムがブロック化するレコード数を決定します。

#### **\*STDINZ**

VALUE文節をもたないこれらの項目の場合に、コンパイラーはデータ項目をシステムの省略時の値に初期設定します。

#### **\*NOSTDINZ**

VALUE文節をもたないこれらの項目の場合に、コンパイラーはデータ項目をシステムの省略時の値に初期設定しません。

#### **\*STDINZHEX00**

VALUE文節をもたないこれらの項目の場合に、コンパイラーはデータ項目を16進数のゼロに初期設定します。

#### **\*NODDSFILLER**

COPY DDSステートメントによって突き合わせフィールドが見つからない場合には、フィールド記述は生成されません。

#### **\*DDSFILLER**

COPY DDSステートメントによって突き合わせフィールドが見つからない場合には、単一文字のFILLERフィールド記述"07 FILLER PIC X"が常に作成されます。

#### **\*NOIMBEDERR**

エラー・メッセージはコンパイル・リストのソース・リスト・セクションに含まれません。エラー・メッセージは、コンパイル・リストのエラー・メッセージ・セクションにだけ表示されます。

#### **\*IMBEDERR**

第1レベル・エラー・メッセージは、コンパイル・リストのソース・リスト・セクションに含まれ、エラーが起こった行の直後に表示されます。エラー・メッセージは、コンパイル・リストのエラー・メッセージ・セクションにも表示されます。

#### **\*STDTRUNC**

このオプションはUSAGE BINARYデータにだけ適用されます。\*STDTRUNCを選択した時には、USAGE BINARYデータはBINARY受け取りフィールドのPICTURE文節の桁数まで切り捨てられます。

#### **\*NOSTDTRUNC**

このオプションはUSAGE BINARYデータにだけ適用されます。\*NOSTDTRUNCを選択した時には、BINARY受け取りフィールドはハーフ・ワード、フル・ワード、またはダブル・ワード境界までのみ切り捨てられます。また、BINARY送り出しフィールドもハーフ・ワード、フル・ワード、またはダブル・ワードとして処理されます。したがって、フィールドの全2進数の内容が有効です。また、DISPLAYステートメントはBINARY フィールドの内容全体が切り捨てなしで変換されます。

#### **\*NOCHGPOSSGN**

ゾーンおよびパック数字データの省略時の正符号として16進数のFが使用されます。16進数のFは、オペレーティング・システムのシステム省略時値です。

#### **\*CHGPOSSGN**

ゾーンおよびパック数字データの省略時の正符号として16進数のCが使用されます。これは、VALUE文節の結果の他にMOVE, ADD, SUBTRACT, MULTIPLY, DIVIDE, COMPUTE,およびINITIALIZEステートメントのすべての結果に適用されます。

#### **\*NOEVENTF**

クライアント・ツールで使用するイベント・ファイルを作成しません。クライアント・ツールは、統合エラー・フィードバックを提供するときにこのファイルを使用します。イベント・ファイルは通常、クライアント・ツール内からモジュールまたはプログラムを作成する時に作成されます。

#### **\*EVENTF**

クライアント・ツールで使用するイベント・ファイルを作成します。イベント・ファイルは、作成されるモジュールまたはプログラム・オブジェクトが記憶されるライブラリー中のファイル EVFEVENT のメンバーとして作成されます。ファイル EVFEVENT が存在していない場合には、自動的にこれが作成されます。イベント・ファイルのメンバー名は、作成されるオブジェクトの名前と同じです。

クライアント・ツールは、統合エラー・フィードバックを提供するときにこのファイルを使用します。イベント・ファイルは通常、クライアント・ツール内からモジュールまたはプログラムを作成する時に作成されます。

#### **\*MONOPIC**

PICTURE 文字ストリングは、すべて大文字（単一シフト）に変換されます。

#### **\*NOMONOPIC**

PICTURE 文字ストリングに使用される通貨記号は大文字・小文字が区別されます。すなわち、PICTURE 記号 A, B, E, G, N, P, S, V, X, Z, CR, および DB の大文字に対応する小文字は、PICTURE 文字ストリングの中のそれらの大文字表記と同じです。他の小文字はすべて対応するそれらの大文字表記と同じではありません。

#### **\*NOCRTARKIDX**

永続索引が見つからない場合には、一時代替レコード・キー (ARK) 索引は作成されません。

#### **\*CRTARKIDX**

永続索引が見つからない場合に、一時代替レコード・キー (ARK) 索引が作成されます。

上

---

## **変換オプション (CVTOPT)**

コンパイラーが、COPY DDS を介して外部記述ファイルからプログラムに渡された日付、時刻、およびタイム・スタンプ・フィールド・タイプ、DBCS フィールド・タイプ、可変長文字フィールド・タイプ、および浮動小数点フィールド・タイプを処理する方法を指定します。指定できる値は次の通りです。

#### **\*NOVARCHAR**

可変長フィールドは FILLER フィールドとして宣言されます。

#### **\*VARCHAR**

可変長フィールドは、グループ項目として宣言され、ILE COBOL ソース・プログラムにアクセスすることができます。

#### **\*NODATETIME**

日付、時刻、およびタイム・スタンプ・データ・タイプが FILLER フィールドとして宣言されます。

#### **\*DATETIME**

日付、時刻、およびタイム・スタンプの DDS データ・タイプは、DDS 名に基づいて指定された COBOL データ項目名です。CVTOPT パラメーター値 \*DATE, \*TIME, または \*TIMESTAMP の 1 つが指定されない限り、COBOL データ項目のカテゴリーは英数字です。この場合には、COBOL データ項目のカテゴリーは、それぞれ日付、時刻、またはタイム・スタンプです。



#### **\*NOPIXGRAPHIC**

DBCSグラフィック・データ・タイプがFILLERフィールドとして宣言されます。

#### **\*PICXGRAPHIC**

固定長DBCSグラフィックデータ・タイプは固定長英数字フィールドとして宣言され、ILE COBOLソース・プログラムにアクセスすることができます。

\*VARCHARオプションも使用されている場合には、可変長DBCSグラフィック・データ・タイプは固定長グループ項目として宣言されて、ILE COBOLソース・プログラムにアクセスすることができます。

#### **\*PICGGRAPHIC**

固定長DBCSグラフィック・データ・タイプは固定長Gタイプ・フィールドとして宣言され、ILE COBOLソース・プログラムにアクセスすることができます。

\*VARCHARオプションも使用されている場合には、可変長DBCSグラフィック・データ・タイプは固定長グループ項目（後にGタイプ・フィールドが続く数値フィールドから構成される）として宣言され、ILE COBOLソース・プログラムにアクセスすることができます。

#### **\*NOPIXGRAPHIC**

DBCSグラフィック・データ・タイプがFILLERフィールドとして宣言されます。

#### **\*NOFLOAT**

浮動小数点データ・タイプは2進数のUSAGEが指定されたFILLERフィールドとして宣言されます。

#### **\*FLOAT**

浮動小数点データ・タイプが、そのDDS名およびCOMP-1（単精度）またはCOMP-2（倍精度）のUSAGEが指定されてプログラムに組み込まれます。これらのフィールドはILE COBOLソース・プログラムにアクセスできるようになります。

#### **\*NODATE**

DDS日付データ・タイプが、カテゴリ英数字COBOL日付項目として宣言されます。たとえば、次のようになります。

06 FILLER PIC X(10).

COBOLデータ項目名が、\*NODATETIME/\*DATETIME CVTOPTパラメーターによって判別されます。

#### **\*DATE**

DDS日付データ・タイプは、カテゴリ日付COBOLデータ項目として宣言されます。たとえば、次のようになります。

06 FILLER FORMAT DATE '@Y-%M-%D'.

COBOLデータ項目名が、\*NODATETIME/\*DATETIME CVTOPTパラメーターによって判別されます。

#### **\*NOTIME**

DDS時刻データ・タイプは、カテゴリ英数字COBOLデータ項目として宣言されます。たとえば次のようになります。

06 FILLER PIC X(10).

COBOLデータ項目名が、\*NODATETIME/\*DATETIME CVTOPTパラメーターによって判別されます。

#### **\*TIME**

DDS時刻データ・タイプが、カテゴリ時刻COBOLデータ項目として宣言されます。たとえば、次のようになります。

06 FILLER FORMAT TIME '%H:%M:%S'.

COBOLデータ項目名が、\*NODATETIME/\*DATETIME CVTOPTパラメーターによって判別されます。

#### **\*NOTIMESTAMP**

DDSタイム・スタンプ・データ・タイプが、カテゴリ英数字COBOLデータ項目として宣言されます。たとえば、次のようになります。

06 FILLER PIC X(10).

COBOLデータ項目名が、\*NODATETIME/\*DATETIME CVTOPTパラメーターによって判別されます。

#### **\*TIMESTAMP**

DDSタイム・スタンプ・データ・タイプが、カテゴリのタイム・スタンプCOBOLデータ項目として宣言されます。たとえば、次のようになります。

06 FILLER FORMAT TIMESTAMP '%H:%M:%S'.

COBOLデータ項目名が、\*NODATETIME/\*DATETIME CVTOPTパラメーターによって判別されます。

#### **\*NOCVTODATE**

DATFMTキーワードを指定したDDSデータ・タイプ(DDS日付データ・タイプを除く)は、元のDDSタイプに基づいてILE COBOLで宣言されます。

#### **\*CVTTODATE**

DATFMTキーワードを指定したDDSデータ・タイプ(DDS日付データ・タイプを除く)は、日付データ・タイプとしてILE COBOLで宣言されます。

#### **\*NOPICNGRAPHIC**

DBCSグラフィック・データ・タイプがFILLERフィールドとして宣言されます。

#### **\*PICNGRAPHIC**

固定長DBCSグラフィック・データ・タイプは固定長各国語データ・フィールドとして宣言され、ILE COBOLソース・プログラムにアクセスすることができます。

上

---

## **メッセージ限界 (MSGLMT)**

各コンパイル単位について、その数だけのエラーが発生するとコンパイルが停止する特定のエラー重大度レベルのメッセージの最大数を指定します。1つのコンパイル単位がこの最大数に達すると、ソース・メンバー全体のコンパイルが停止されます。

たとえば、メッセージの最大数に3を指定し、エラー重大度レベルに20を指定した場合には、20以上の重大度レベルのエラーが3つまたはそれ以上起こった場合に、コンパイルが停止します。指定のエラー重大度レベルと等しいかまたは超えているメッセージがない場合には、見つかったエラーの数に関係なくコンパイルは続行されます。

### **メッセージの数**

メッセージの最大数を指定してください。指定できる値は次の通りです。

#### **\*NOMAX**

見つかったエラーの数に関係なく正常完了までコンパイルは続行されます。

**最大数** コンパイルが停止する前に指定のエラー重大度レベルでまたはそれ以上で発生可能なメッセージの最大数を指定します。有効な範囲は0-9999です。

### メッセージ限界重大度

コンパイルを停止するかどうかを判別するのに使用されるエラー重大度レベルを指定します。指定できる値は次の通りです。

**30** 重大度レベル30以上のエラーの数が指定されたメッセージの最大数を超過している場合に、コンパイルが停止します。

### エラー重大度レベル

コンパイルを停止するかどうかを判別するのに使用したいエラー重大度レベルを、0-30の1桁または2桁の数字で入力してください。この重大度レベル以上のエラーの数が指定されたメッセージの最大数を超過している場合に、コンパイルが停止します。

上

---

## デバッグ・ビューのオプション (DBGVIEW)

コンパイルされたプログラムのデバッグにどのソース・プログラムまたは生成されたリストのビューが使用可能かを制御するオプション、およびデバッグ・リスト・ビューが圧縮されるかどうかを制御するオプションを指定します。

### デバッグ・ビュー

デバッグに使用可能になるビューを指定してください。指定できる値は次の通りです。

#### **\*STMT**

記号名およびステートメント番号を使用してコンパイル済みプログラムをデバッグすることができます。

#### **\*SOURCE**

COPYステートメントを介して含まれたコピーされたメンバーの他に、1次ソース・メンバーには、コンパイル済みプログラムのデバッグ用に使用可能なソース・ビューがあります。これらのビューは、1次ソース・メンバーおよびコピーされたソース・メンバーがローカル・データベース・ソース・ファイルから参照される場合だけ使用可能です。コンパイルおよびデバッグ中にメンバーを変更または削除しないでください。

**\*LIST** COPYおよびREPLACEステートメントの処理後にソース・コードを表示するリストは、コンパイル済みモジュールのデバッグに使用することができます。このオプションはコンパイル済みモジュールのサイズが増えますが、コンパイル済みプログラムの実行時パフォーマンスには影響しません。

リスト・ビューは、対応するコンパイラー・オプションが要求されたときに、相互参照表、データ部のマップ、および動詞の使用カウントを含めます。たとえば、OPTION(\*XREF)が指定されると、相互参照表が含まれます。

1次ソース・メンバーおよびコピーされたソース・メンバーがどこにあっても、リスト・ビューを生成することができます。リスト・ビューはコンパイル後のソース・メンバーの変更または削除の影響を受けません。

**\*ALL** \*STMT, \*SOURCE,および\*LISTを組み合わせることで指定することと等価です。



#### **\*NONE**

コンパイル済みプログラムをデバッグすることはできません。これはコンパイル済みプログラムのサイズを減少しますが、その実行時パフォーマンスに影響しません。このオプションを指定した時には、定様式ダンプをとることはできません。

#### **リスト・ビューの圧縮**

デバッグ・ビューで\*LISTまたは\*ALLを指定した場合に、リスト・ビューを圧縮するかどうかを指定します。指定できる値は次の通りです。

#### **\*NOCOMPRESSDBG**

リスト・ビューは圧縮されません。

#### **\*COMPRESSDBG**

デバッグ・ビューで\*LISTまたは\*ALLを指定した場合に、リスト・ビューは圧縮されます。

上

---

## **最適化レベル (OPTIMIZE)**

プログラムの最適化のレベルを指定します。指定できる値は次の通りです。

#### **\*NONE**

コンパイル済みプログラムで最適化は実行されません。このオプションを使用した時には、コンパイル時間は最小化されます。このオプションによって、デバッグ中に変数を表示および変更することができます。

#### **\*BASIC**

コンパイル済みプログラムで一部の最適化（ローカル・ブロック・レベルでのみ）が実行されます。このオプションによって、デバッグ中にユーザー変数を表示できますが、変更することはできません。

#### **\*FULL**

コンパイル済みモジュールで完全な最適化（グローバル・レベルで）が実行されます。この最適化によって、コンパイル時間は増えますが、最も効率的なコードが生成されます。このオプションによって、デバッグ中にユーザー変数を表示できますが、変更することはできません。表示された変数の値は現行値でない場合があります。一部の変数は、表示されない場合があります。

注: 選択した最適化レベルに関係なく、全面的な最適化を可能にするすべての情報が生成されます。ユーザーは、ソース・プログラムを再コンパイルすることなく、CHGMOD コマンドを使用して、プログラム・オブジェクトの\*NONEから\*FULLまで最適化レベルを変更することができます。

上

---

## **FIPSフラグ付け (FLAGSTD)**

FIPSフラグ付けのオプションを指定します。（FIPSメッセージで使用された参照番号が固有であることを確認するためには、\*LINENUMBERオプションを選択してください。）指定できる値は次の通りです。

#### **\*NOFIPS**

ILE COBOLソース・プログラムにはFIPSフラグ付きではありません。

#### **\*MINIMUM**

最低サブセット以上のFIPSフラグ。

#### **\*INTERMEDIATE**

中間サブセット以上のFIPSフラグ。

#### **\*HIGH**

高サブセットのFIPSフラグ。

#### **\*NOOBSOLETE**

使用しない言語要素にフラグが付けられません。

#### **\*OBSOLETE**

使用されなくなった言語要素にフラグが付けられます。

上

---

## **拡張表示オプション (EXTDSPOPT)**

ワークステーション入出力用に拡張ACCEPTおよび拡張DISPLAYステートメントを使用するためのオプションを指定します。指定できる値は次の通りです。

#### **\*DFRWRT**

拡張DISPLAYステートメントは、拡張ACCEPTステートメントが見つかるか、あるいはバッファーが満たされるまで、バッファーの中に保留されます。

バッファーの内容は、拡張ACCEPTステートメントが見つかるか、あるいはバッファーが満たされた時に表示装置に書き出されます。

#### **\*NODFRWRT**

各拡張DISPLAYステートメントはそれが出てきた時に実行されます。

#### **\*UNDSPCHR**

表示可能および表示不能文字は、拡張ACCEPTおよび拡張DISPLAYステートメントによって処理されます。

#### **\*NOUNDSPCHR**

表示可能文字だけが、拡張ACCEPTおよび拡張DISPLAYステートメントによって処理されます。

リモート3174および3274制御装置に接続された表示装置にこのオプションを使用しなければなりません。ローカル・ワークステーションにもこのオプションを使用することができます。このオプションを使用する場合には、データに表示可能文字だけが入っていなければなりません。データに16進数20より小さい値が入っている場合には、予期しない画面様式から重大エラーに至る結果を予測することはできません。

#### **\*ACCUPDALL**

UPDATE句の存在に関係なく、拡張ACCEPTステートメントですべてのタイプのデータが事前表示されます。

#### **\*ACCUPDNE**

UPDATE句が含まれていない拡張ACCEPTステートメントで数字編集されたデータだけが事前表示されます。

---

## フラグ重大度 (FLAG)

コンパイル・リストに表示するメッセージの最小重大度レベルを指定します。指定できる値は次の通りです。

**0** 全てのメッセージがコンパイル・リストに表示されます。

### 重大度レベル

コンパイル・リストに表示したいメッセージの最小重大度レベルを指定する1桁または2桁の数字を入力してください。指定されたこの値以上の重大度レベルをもつメッセージがコンパイル・リストに表示されます。

---

## プログラムの置き換え (REPLACE)

指定されたライブラリーまたは暗黙のライブラリーに同じ名前のプログラムがすでに存在している時に、新しいプログラムを作成するかどうかを指定します。CRTBNDCBL コマンドの処理中に作成される中間モジュール・オブジェクトはREPLACE指定の対象ではなく、QTEMPライブラリーに対してはREPLACE(\*NO)が暗黙に指定されたものと見なされます。中間モジュール・オブジェクトは、CRTBNDCBLコマンドが処理を完了すると削除されます。REPLACEパラメーターに指定できる値は、次の通りです。

**\*YES** 新しいプログラムが作成され、指定されたライブラリーまたは暗黙のライブラリーの同じ名前のすべてのプログラムを置き換えます。指定されたライブラリーまたは暗黙のライブラリーの同じ名前の既存のプログラムはライブラリーQRPLOBJに移動されます。

**\*NO** 指定されたライブラリーまたは暗黙のライブラリーに同じ名前のプログラムがすでに存在している場合には、新しいプログラムは作成されません。既存のプログラムは置き換えられず、メッセージが表示され、コンパイルは停止します。

---

## ユーザー・プロファイル (USRPRF)

作成されたプログラム・オブジェクトを実行するユーザー・プロファイルを指定します。プログラムを実行し、プログラム（各プログラムに対してオブジェクトがもっている権限を含む）によってオブジェクトを使用可能にすることを制御するために、プログラム所有者またはプログラム・ユーザーのプロファイルが使用されます。プログラムがすでに存在している場合には、このパラメーターは更新されません。USRPRFの値を変更するためには、プログラムを削除し、正しい値を使用してコンパイルし直してください（あるいは、構成要素\*MODULEオブジェクト（1つまたは複数）が存在する場合には、CRTPGMコマンドを呼び出すことを選択することができます）。

指定できる値は次の通りです。

### **\*USER**

プログラムが実行される時に、プログラムの使用者のユーザー・プロファイルが使用されます。

### **\*OWNER**

プログラムが実行される時に、プログラムの所有者と使用者の両方のユーザー・プロファイルが使

用されます。プログラム・オブジェクトの実行時に、オブジェクトを見つけてアクセスするために、所有者とユーザーの両方のプロファイルのオブジェクト権限の集合セットが使用されます。プログラムの実行時に作成されるオブジェクトは、すべてプログラムのユーザーによって所有されます。

上

---

## 権限 (AUT)

プログラム・オブジェクトに対する特定権限を持っていないユーザー、権限リスト上にないユーザー、またはグループがプログラム・オブジェクトに対する特定権限を持っていないユーザーに与える権限を指定します。プログラム・オブジェクトを作成した後で、GRTOBJAUT（オブジェクト権限認可）またはRVKOBJAUT（オブジェクト権限取り消し）コマンドを使用してすべてのユーザーまたは特定のユーザーの権限を変更することができます。

指定できる値は次の通りです。

### **\*LIBCRTAUT**

オブジェクトの共通権限は、ターゲット・ライブラリー（作成されたプログラム・オブジェクトを入れるライブラリー）のCRTAUTキーワードから引用されます。この値はプログラム・オブジェクトが作成される時に決定されます。プログラム・オブジェクトが作成された後でライブラリーのCRTAUT値が変更された場合には、新しい値は既存のすべてのオブジェクトに影響しません。

**\*ALL** 所有者に限定されているか、または権限リスト管理権限によって管理されている以外のプログラム・オブジェクトにすべての操作を実行する権限を提供します。ユーザーはプログラム・オブジェクトの存在を制御し、それを機密保護し、それを変更し、それに対して基本機能を実行することができますが、その所有権を転送することはできません。

### **\*CHANGE**

所有者に限定されているか、またはオブジェクト権限およびオブジェクト管理権限によって制御されているものを除き、プログラム・オブジェクトのすべて操作を行うための権限、およびすべてのデータ権限を提供します。ユーザーはオブジェクトを変更し、それに対して基本機能を実行することができます。

**\*USE** プログラム・オブジェクトに対する基本操作の権限である、オブジェクト操作権および読み取り権限を提供します。ユーザーはオブジェクトに対して基本的な操作を実行することができますが、オブジェクトを変更することはできません。

注：プログラムの変数の定様式ダンプを取るには、ユーザーにプログラムの**\*USE**権限が必要です。変数をダンプするには、プログラムに検討可能な情報も必要です。

一部のユーザーに変数のダンプをさせたくない場合には、そのユーザーにプログラムに対する**\*OBJOPR**権限と**\*EXECUTE**権限だけを割り当ててください。これにより、そのユーザーはプログラムを呼び出すことはできますが、その変数をダンプすることはできません。作成済みのプログラムの権限を変更するには、EDTOBJAUT、GRTOBJAUT、またはRVKOBJAUTを使用してください。

いかなるユーザーにも変数のダンプをさせたくない場合には、プログラム変更(CHGPGM)を使用してプログラムの検討可能な情報を除去してください。

### **\*EXCLUDE**

ユーザーはプログラム・オブジェクトをアクセスすることができません。

### 権限リスト名

ユーザーの権限リストの名前およびプログラムを追加する権限。この権限リストによってプログラ

ム・オブジェクトが保護され、プログラム・オブジェクトに対する共通権限が\*AUTLにセットされます。CRTBNDCBLコマンドが出された時に、この権限リストはシステム上に存在していなければなりません。権限リスト作成(CRTAUTL)コマンドを使用してユーザー専用の権限リストを作成してください。

上

---

## リンク・リテラル (LINKLIT)

外部CALL/CANCEL 'リテラル' 行き先およびSET ENTRY行き先の関係タイプを指定します。SPECIAL-NAMES段落に次の文を指定することによって、特定の外部CALL/CANCEL 'リテラル' 行き先およびSET ENTRY行き先リストに対するこのオプションを一時変更することができます。

LINKAGE TYPE IS プログラム内ファイル名 FOR 行き先リスト。

LINKLITに指定できる値は、次の通りです。

**\*PGM** CALL/CANCELまたはSET ENTRYの行き先はプログラム・オブジェクトです。

**\*PRC** CALL/CANCELまたはSET ENTRYの行き先はILEプロシージャです。

上

---

## 単純プログラム (SIMPLEPGM)

ソース・プログラムの順序にしたがってコンパイル単位ごとにプログラム・オブジェクトを作成するかどうかを指定します。このオプションは、このコマンドに対する入力ソース・メンバーに複数のモジュールを生成する一連のソース・プログラムが入っている場合にだけ意味があります。このオプションが指定され、入力ソース・メンバーに一連のソース・プログラムが入っていない場合には、このオプションは無視されます。指定できる値は次の通りです。

**\*YES** 一連のソース・プログラム中の各コンパイル単位ごとにプログラム・オブジェクトが作成されます。REPLACE(\*NO)を指定し、一連のソース・プログラムのコンパイル単位に同じ名前のプログラム・オブジェクトが存在している場合には、そのプログラム・オブジェクトは置き換えられず、次のコンパイル単位からコンパイルが続行されます。

**\*NO** 一連のコンパイル単位すべてから、単一のプログラム・オブジェクトが作成され、最初のコンパイル単位がプログラムの入り口を表します。SIMPLEPGM(\*NO)を指定し、一連のソース・プログラムの中の1つのソース・プログラムがモジュールを正常に生成できなかった場合には、後続の一連のソース・プログラムはすべて、モジュールを正常に生成しないことになります。

上

---

## ターゲット・リリース (TGTRLS)

作成されるプログラム・オブジェクトを使用する予定であるオペレーティング・システムのリリースを指定します。\*CURRENTおよび\*PRV値の説明で示した例および ターゲット・リリース 値を指定するときのリリースの指定方法はVXRXXMXの形式です。ここで、VXはバージョンで、RXはリリースで、MXはモディフィケーション・レベルです。たとえば、V2R3M0はバージョン2,リリース3,モディフィケーション・レベル0です。

このパラメーターに有効な値はリリースごとに変わります。指定できる値は次の通りです。



## **\*CURRENT**

オブジェクトは、現在システムで実行されているオペレーティング・システムのリリースで 사용됩니다。たとえば、システムでV2R3M5が実行されている場合には、\*CURRENTはV2R3M5が導入されているシステムでオブジェクトを使用する予定であることを意味します。また、このオブジェクトは、オペレーティング・システムの後続のリリースが導入されたシステムで使用することもできます。

注: システム上でV2R3M5が実行されている場合で、オブジェクトをV2R3M0が導入されているシステムで使いたい場合には、TGTRLS(\*CURRENT)ではなくTGTRLS(V2R3M0)を指定してください。

**\*PRV** オブジェクトは、モディフィケーション・レベルが0のオペレーティング・システムの前のリリースで使われます。たとえば、ユーザーのシステムでV2R3M5が実行されている場合には、\*PRVはV2R2M0が導入されているシステムでオブジェクトを使用する予定であることを意味します。また、このオブジェクトは、オペレーティング・システムの後続のリリースが導入されたシステムで使用することもできます。

## **ターゲット・リリース**

リリースをVXRXXMXの形式で指定してください。オブジェクトは、指定されたリリースのシステムまたはオペレーティング・システムの後続のリリースが導入されたシステムで使うことができます。

有効な値は、現行バージョン、リリース、およびモディフィケーション・レベルによって異なり、新しいリリースごとに変わります。このコマンドでサポートされている最も古いリリース・レベルよりも前の ターゲット・リリース を指定した場合には、エラー・メッセージが出され、サポートされる最も古いリリースが表示されます。

注: コマンドの現行バージョンは、前のコマンドのリリースでは使えないオプションをサポートすることもあります。コマンドが前のリリースで使うオブジェクトを作成するために使われた場合には、そのリリースに適したコンパイラで処理されて、サポートされないオプションは認識されません。コンパイラは、必ずしも処理できないオプションについての警告を出すとは限りません。

上

---

## **ソート順序 (SRTSEQ)**

ALPHABET文節の中でNLSSORTが英字名と関連している時に使われるソート順序を指定します。SRTSEQパラメーターは、プログラム・オブジェクトが使うシステム定義またはユーザー定義のソート順序テーブルを決定するために、LANGIDパラメーターと一緒に使われます。指定できる値は次の通りです。

**\*HEX** ソート順序テーブルは使われず、ソート順序を決定するために文字の16進数値が使われます。

**\*JOB** ソート順序は、コンパイル時にコンパイル・ジョブのソート順序を使って分析解決され、プログラム・オブジェクトと関連づけられます。ソート順序テーブルはコンパイル時にシステムに存在していなければなりません。実行時に、実行時ジョブのCCSIDのソート順序がコンパイル時ジョブのCCSIDと異なっている場合には、コンパイル時にロードされたソート順序テーブルが、実行時ジョブのCCSIDと一致するように変換されます。

## **\*JOBRUN**

プログラムのソート順序は、実行時に分析解決され、プログラムと関連づけられます。コンパイル時に、コンパイラはコンパイル・ジョブのソート順序をプログラムと関連づけます。実行時に、

このソート順序は実行時にジョブと関連づけるソート順序に置き換えられます。この値によって、プログラムはいったんコンパイルされて、実行時に別のソート順序と一緒に使用されます。

#### **\*LANGIDUNQ**

使用中のソート順序テーブルにはコード・ページ中の各文字に対する固有の重みが入っていないかならないことを指定します。使用されるソート順序テーブルは、LANGID パラメーターで指定された言語と対応した、固有の重みづけされたテーブルでなければなりません。

#### **\*LANGIDSHR**

使用中のソート順序テーブルにはコード・ページ中の複数の文字に同じ重みを入れることができることを指定します。使用されるソート順序テーブルは、LANGIDパラメーターで指定された言語と対応した、共用の重みづけされたテーブルです。

#### **テーブル名**

使用するソート順序テーブルの名前を指定してください。テーブルには指定されたコード・ページ中のすべての文字に対する重みが入っています。重みは、コード・ポイントで定義された文字と関連づけられています。ソート順序テーブル名を使用する時に、オブジェクトが存在するライブラリーを指定することができます。有効なライブラリーの値は次の通りです。

**\*LIBL** ソート順序テーブルが入っているライブラリーを見つけるためにライブラリー・リストが探索されます。

#### **\*CURLIB**

現行ライブラリーが使用されます。現行ライブラリーとしてライブラリーが割り当てられていない場合には、QGPLが使用されます。

#### **ライブラリー名**

ソート順序テーブルが入っているライブラリーの名前を入力してください。

上

---

## **言語ID (LANGID)**

ソート順序との組み合わせで使用される言語IDを指定します。LANGIDパラメーターは、有効なSRTSEQ値が\*LANGIDUNQまたは\*LANGIDSHRの時にだけ使用されます。指定できる値は次の通りです。

#### **\*JOBRUN**

プログラムの言語IDは実行時に分析解決されます。コンパイル済みプログラムを実行する時に、ジョブの言語IDが使用されます。この値によって、プログラムはいったんコンパイルされて、実行時に別の言語IDと一緒に使用されます。

**\*JOB** プログラムの言語IDはコンパイル時に分析解決されます。

#### **言語ID名**

有効な3文字の言語IDを入力してください。

上

---

## パフォーマンス収集の活動化 (ENBPFRCOL)

モジュールまたはプログラムの中でパフォーマンス測定コードを生成するかどうかを指定します。収集されたデータを使用して、システム・パフォーマンス・ツールでアプリケーションのパフォーマンスのプロファイルを作成することができます。コンパイル済みのモジュールまたはプログラムにパフォーマンス測定コードの追加を生成することにより、オブジェクトがわずかに大きくなり、パフォーマンスに影響することがあります。

**\*PEP** パフォーマンス統計は、プログラム・エントリー・プロシーチャーの入り口および出口でのみ収集されます。アプリケーションの全体的なパフォーマンス情報を収集したい場合には、この値を選択してください。このサポートは、前にTPSTツールが提供されたサポートと同等です。これは省略時の値です。

### **\*ENTRYEXIT**

パフォーマンス統計は、プログラムのすべてのプロシーチャーの入り口および出口に収集されます。これには、プログラムPEPルーチンが含まれます。

この選択項目は、すべてのルーチンについての情報を収集したい場合に便利です。ユーザー・アプリケーションによって呼び出されるすべてのプログラムが\*PEP、\*ENTRYEXITまたは\*FULLオプションを使用してコンパイルされたことが分かっている場合には、このオプションを使用してください。そうでない場合には、ユーザー・アプリケーションがパフォーマンス測定が不能な他のプログラムを呼び出した場合には、パフォーマンス測定ツールは、それらのプログラムが資源を使用することはユーザー・アプリケーションに責任があるものと見なします。これにより、実際に資源がどこで使用されるのかを判別することが困難となります。

### **\*FULL**

パフォーマンス統計はすべてのプロシーチャーの入り口および出口に収集されます。また、統計は外部プロシーチャーに対する各呼び出しの前後に収集されます。

ユーザー・アプリケーションが、\*PEP、\*ENTRYEXITまたは\*FULLのいずれかを使用してコンパイルされたものではない他のプログラムを呼び出すと思われる場合には、このオプションを使用してください。このオプションを使用することにより、パフォーマンス・ツールは、ユーザー・アプリケーションによって使用される資源と、ユーザー・アプリケーションが呼び出すプログラムによって使用される資源を区別することができます（呼び出されるプログラムがパフォーマンス測定不能であっても）。このオプションは最も不経済ですが、アプリケーション中のいろいろなプログラムを選択的に分析することができます。

上

---

## バインディング・ディレクトリー (BNDDIR)

記号解決で使用されるバインディング・ディレクトリーのリストを指定します。

### **\*NONE**

バインディング・ディレクトリーは指定されません。

### **バインディング・ディレクトリー名**

記号解決で使用されるバインディング・ディレクトリーの名前を指定してください。ディレクトリー名は次のライブラリー値の1つで修飾することができます：

**\*LIBL** システムは、バインディング・ディレクトリーが記憶されているライブラリーを見つけるために、ライブラリー・リストを検索します。これは省略時の値です。



#### **\*CURLIB**

ジョブの現行ライブラリーが検索されます。ジョブの現行ライブラリーとしてライブラリーが指定されていない場合には、ライブラリーQGPLが使用されます。

#### **\*USRLIBL**

ジョブのライブラリー・リストのユーザー部分の中のライブラリーだけが検索されます。

#### **ライブラリー名**

検索するライブラリーの名前を指定してください。

上

---

## **活動化グループ (ACTGRP)**

このプログラムが呼び出された時にこのプログラムに関連する活動化グループを指定します。

**QILE** このプログラムが呼び出されると、プログラムは指定の活動化グループQILEに活動化されます。これは省略時の値です。

**\*NEW** このプログラムが呼び出されると、プログラムは新しい活動化グループに活動化されます。

#### **\*CALLER**

このプログラムが呼び出されると、プログラムは呼び出し側の活動化グループに活動化されます。

#### **活動化グループ名**

このプログラムが呼び出された時に使用する活動化グループの名前を指定してください。

上

---

## **プロファイル・データ (PRFDTA)**

プログラムにプログラム・プロファイル・データ属性を指定します。プログラム・プロファイルは、統計データ（プロファイル・データ）に基づいてプロシージャおよびプロシージャ内のコードを再順序づけるために使用される拡張最適化手法です。

#### **\*NOCOL**

このプログラムはプロファイル・データを収集できません。これは省略時の値です。

**\*COL** プログラムはプロファイル・データを収集できます。**\*COL**を指定できるのは、モジュールの最適化レベルが**\*FULL**の時だけです。

注: BNDDIRパラメーターを使用して追加のモジュールおよびサービス・プログラムをバインドする場合に、**\*COL**または**\*NOCOL**がプログラムに指定されている時にはこれら追加のオブジェクトは影響を受けません。モジュールのプログラム・プロファイル・データ属性は、モジュール作成時に設定されます。

上

---

## **コード化文字セットID (CCSID)**

実行時にファイル中のレコードとLOCALEに関連したデータが変換されるコード化文字セットID (CCSID)を指定します。

### **\*JOBRUN**

プログラムのCCSIDが実行時に解決されます。コンパイル済みプログラムを実行すると、現行ジョブの省略時値CCSIDが使用されます。

**\*JOB** コンパイル時の現行ジョブの省略時値CCSIDが使用されます。

**\*HEX** CCSID 65535が使用されます。これは、フィールドのデータがビット・データとして扱われ、変換されないことを示します。

### **コード化文字セットID**

使用するCCSIDを指定します。

上

---

## **国別CCSID (NTLCCSID)**

すべての国別データ項目に使用される、国別コード化文字セットID (NTLCCSID)を指定します。指定できる値は次の通りです。

**13488** UCS-2 CCSID 13488が、すべての国別データ項目に使用されます。

### **コード化文字セットID**

指定されたCCSIDが、すべての国別データ項目に使用されます。指定するCCSIDはUCS-2互換（たとえばUTF-16 CCSID 1200）でなければなりません。

上

---

## **演算モード (ARITHMETIC)**

数字データに演算モードを指定します。指定できる値は次の通りです。

### **\*NOEXTEND**

このオプションは、数字データの省略時の演算モードを指定します。固定小数点演算式の間中間結果は最大30桁までで、数値リテラルの最大長は18桁だけです。

### **\*EXTEND31**

固定小数点演算の間中間結果の精度を増すには、このオプションを使用してください。固定小数点演算式の間中間結果は最大31桁までで、数値リテラルの最大長は31桁の場合があります。

### **\*EXTEND63**

固定小数点演算の間中間結果の精度を増すには、このオプションを使用してください。固定小数点演算式の間中間結果は最大63桁までで、数値リテラルの最大長は63桁とすることができます。

上

---

## **PADDING CHARACTER (NTLPADCHAR)**

以下のような変換状態で埋め込みが行われる時に使用される各国語埋め込み文字(NTLPADCHAR)を指定します。

1. 単一バイト文字を国別文字へ。
2. 2バイト文字を国別文字へ。

3.国別文字を国別文字へ。

#### **\*DEFAULT**

このオプションは、以下のような省略時の埋め込み文字を指定します。

- 1.単一バイト文字を国別文字へ(NX"0020")
2. 2バイト文字を国別文字へ(NX"3000")
- 3.国別文字を国別文字へ(NX"3000")

#### **各国語16進リテラル**

長さが1の有効な任意の各国語16進リテラルをNX" "またはNX' 'の形式で指定します。

上

---

## **LICENSED INTERNAL CODE OPTIONS (LICOPT)**

1つまたは複数のライセンス内部コード・コンパイル時オプションを指定します。このパラメーターを使用すれば個別のコンパイル時オプションを選択できますが、これは、選択した個々のコンパイラー・オプションのタイプの潜在的な利点と欠点を理解している上級プログラマーを対象としています。

上

---

## **ディレクトリー組み込み (INCDIR)**

コピー・ファイルを探すためにコンパイラーが使用する検索パスに追加する1つまたは複数のディレクトリーを指定します。コンパイラーは、ソース・プログラムのコピー・ファイルを解決できない場合には、ここで指定したディレクトリーを検索します。

#### **\*NONE**

ユーザー・ディレクトリーでコピー・ファイルは検索されません。省略時値では、現行ディレクトリーが検索されます。

#### **'ディレクトリー'**

コピー・ファイルを検索する最大32のディレクトリーを指定してください。指定されたディレクトリーに加えて、現行ディレクトリーでもコピー・ファイルが検索されます。

上

---

## **PGMインターフェース情報 (PGMINFO)**

プログラム・インターフェース情報を生成するかどうか、および生成する位置を制御するオプションを指定します。

**生成** プログラム・インターフェース情報を生成するかどうかを指定します。指定できる値は次の通りです。

**\*NO** このオプションは、プログラム・インターフェース情報を生成しない省略時値を指定します。

#### **\*PCML**

PCML（プログラム呼び出しマークアップ言語）を生成するように指定します。生成された

PCMLは、JAVAメソッドによるこのCOBOLプログラム内のプロシージャーの呼び出しを容易にするので、JAVAコードは少なくて済みます。

**位置** 「生成」パラメーターが\*PCMLである場合、プログラム情報が生成される位置を指定します。指定できる値は次の通りです。

**\*STMF**

プログラム情報をストリーム・ファイルに生成するように指定します。生成される情報が入るストリーム・ファイルの名前は、INFOSTMF オプションに指定されていなければなりません。

**\*MODULE**

プログラム情報をCOBOLモジュールに保管するように指定します。

**\*ALL** プログラム情報をストリーム・ファイルに生成し、さらにモジュールに保管するように指定します。生成される情報が入るストリーム・ファイルの名前は、INFOSTMF オプションに指定されていなければなりません。

上

---

## プログラム・インターフェース・ストリーム・ファイル (INFOSTMF)

PGMINFOオプションで指定された生成されたプログラム・インターフェース情報を入れるストリーム・ファイルのパス名を指定します。

このパス名は絶対的または相対的に修飾することができます。絶対パス名は'/'で始まり、相対パス名は'/'以外の文字で始まります。

絶対修飾の場合には、パス名は完全です。相対修飾の場合には、パス名は、パス名にジョブの現行作業ディレクトリーを付加することによって完了します。

このパラメーターを指定できるのは、PGMINFOパラメーターに\*NO以外の値がある場合だけです。

上

---

## 例

### 例1:ソース・プログラムをプログラム・オブジェクトにコンパイル

```
CRTBNDCBL  PGM(MYLIB/XMPLE1) SRCFILE(MYLIB/QCBLLESRC)
           SRCMBR(XMPLE1)  OUTPUT(*PRINT)
           TEXT('MY ILE COBOLプログラム')
```

このコマンドは、ILE COBOLコンパイラーを呼び出し、XMPLE1という名前のプログラムを作成します。ソース・プログラムは、ライブラリーMYLIBのソース・ファイルQCBLLESRCのメンバーXMPLE1に入っています。コンパイル・リストが作成されます。

上

---

## エラー・メッセージ

### \*ESCAPE メッセージ

#### LNC9001

コンパイルは正常に行なわれません。&1は作成されていません。

#### LNC9006

TGTRLS(&1)が指定されましたが、コンパイラーが導入されていません。

#### LNC9007

プロダクト・ライブラリーに損傷があるか、あるいはユーザーにはその使用が認可されていない。

#### LNC9015

TGTRLS(&1)が正しくない。

上



## 結合C++プログラム作成 (CRTBNDCPP)

実行可能場所: すべての環境 (\*ALL)

スレッド・セーフ: いいえ

パラメーター

例

エラー・メッセージ

結合C++プログラム作成(CRTBNDCPP)コマンドはILE C++コンパイラーを開始します。このコンパイラーは、ソース・コード内のILE C++ステートメントに基づいてプログラム・オブジェクトを作成しようとしています。

注: CRTBNDCPPコマンドが呼び出された場合には、一時モジュール・オブジェクトがQTEMPライブラリーの中に作成されます。一時モジュール・オブジェクトの名前は、**プログラム (PGM)**パラメーターに指定した名前と同じになります。その名前を持つモジュール・オブジェクトがすでにQTEMPに存在していた場合には、エラー・メッセージが生成され、コンパイルは停止します。CRTBNDCPPコマンドによって使用される一時モジュール・オブジェクトは、コンパイルが正常に行われようといわれまいと、コンパイルが停止した後には削除されます。

上

### パラメーター

キーワード	記述	選択項目	ノート
PGM	プログラム	修飾オブジェクト名	必須, 定位置 1
	修飾子 1: プログラム	名前	
	修飾子 2: ライブラリー	名前, <u>*CURLIB</u>	
SRCFILE	ソース・ファイル	修飾オブジェクト名	オプション, 定位置 2
	修飾子 1: ソース・ファイル	名前, <u>QCPSRC</u>	
	修飾子 2: ライブラリー	名前, <u>*LIBL</u> , <u>*CURLIB</u>	
SRCMBR	ソース・メンバー	名前, <u>*PGM</u>	オプション, 定位置 3
SRCSTMF	ソース・ストリーム・ファイル	パス名	オプション
TEXT	テキスト・記述	文字値, <u>*SRCMBRTXT</u> , *BLANK	オプション
OUTPUT	出力オプション	単一値: <u>*NONE</u> , '*トナトオ' その他の値: 要素リスト	オプション
	要素 1: 出力ファイル名	パス名, *PRINT, '*ニネクトホ'	
	要素 2: タイトル	文字値, <u>*BLANK</u>	
	要素 3: サブタイトル	文字値, <u>*BLANK</u>	

キーワード	記述	選択項目	ノーツ
<b>OPTION</b>	コンパイラー・オプション	値 (最大 35 回の繰り返し): *NOBITSIGN, *BITSIGN, *NOEVENTF, *EVENTF, *NOEXPMAC, *EXPMAC, *NOLOGMSG, *LOGMSG, *NOLONGLONG, *LONGLONG, *NORTTI, *RTTIAL, *RTTITYPE, *RTTICAST, *NOSHOWINC, *SHOWINC, *NOSHOWSRC, *SHOWSRC, *NOSHOWSYS, *SHOWSYS, *NOSHOWUSR, *SHOWUSR, *NOSYSINCPATH, *SYSINCPATH, *NOSTDINC, *STDINC, *NOINCDIRFIRST, *INCDIRFIRST, *NOXREF, *XREF, *NOXREFREF, *XREFREF, *NOFULL, *FULL, *NOSTDLOGMSG, *STDLOGMSG	オプション
<b>CHECKOUT</b>	チェックアウト・オプション	値 (最大 45 回の繰り返し): <u>*NONE</u> , *USAGE, *ALL, *NOCLASS, *CLASS, *NOCOND, *COND, *NOEFFECT, *EFFECT, *NOGENERAL, *GENERAL, *NOLANG, *LANG, *NOPARM, *PARM, *NOPORT, *PORT, *NOREACH, *REACH, *NOTEMP, *TEMP, *NOTRUNC, *TRUNC, *NOUNUSED, *UNUSED	オプション
<b>OPTIMIZE</b>	最適化	<u>10</u> , 20, 30, 40	オプション
<b>INLINE</b>	インライン・オプション	要素リスト	オプション
	要素 1: インライナー	<u>*OFF</u> , *ON	
	要素 2: モード	<u>*AUTO</u> , *NOAUTO	
	要素 3: しきい値	1-65535, <u>250</u> , *NOLIMIT	
	要素 4: 限界	1-65535, <u>2000</u> , *NOLIMIT	
	要素 5: 報告書	<u>*NO</u> , *YES	
<b>DBGVIEW</b>	デバッグ・ビュー	<u>*NONE</u> , *ALL, *STMT, *SOURCE, *LIST	オプション
<b>DEFINE</b>	名前の定義	単一値: <u>*NONE</u> その他の値 (最大 32 回の繰り返し): 文字値	オプション
<b>LANGLVL</b>	言語レベル	<u>*EXTENDED</u> , *ANSI, *LEGACY	オプション



キーワード	記述	選択項目	ノーツ
ALIAS	別名	値 (最大 3 回の繰り返し): <u>*ANSI</u> , *NOANSI, *ADDRTAKEN, *NOADDRTAKEN, *ALLPTRS, *NOALLPTRS, *TYPEPTR, *NOTYPEPTR	オプション
SYSIFCOPT	SYSTEMインターフェースOPT	<u>*IFS64IO</u> , *IFSIO, *NOIFSIO	オプション
LOCALETYPE	LOCALEオブジェクト・タイプ	<u>*LOCALE</u> , *LOCALEUCS2, *LOCALEUTF	オプション
FLAG	メッセージのフラグ・レベル	<u>0</u> , 10, 20, 30	オプション
MSGMT	コンパイラ・メッセージ	要素リスト	オプション
	要素 1: メッセージ限界	0-32767, <u>*NOMAX</u>	
	要素 2: メッセージ限界の重大度	0, 10, 20, <u>30</u>	
REPLACE	PGMオブジェクトの置き換え	<u>*YES</u> , *NO	オプション
USRPRF	ユーザー・プロファイル	<u>*USER</u> , *OWNER	オプション
AUT	権限	名前, <u>*LIBCRTAUT</u> , *ALL, *CHANGE, *USE, *EXCLUDE	オプション
TGTRLS	ターゲット・リリース	単純名, <u>*CURRENT</u> , *PRV	オプション
ENBPFRCOL	パフォーマンス収集の使用可能	要素リスト	オプション
	要素 1: 収集レベル	<u>*PEP</u> , *ENTRYEXIT, *FULL	
	要素 2: プロシージャ	*NONLEAF, *ALLPRC	
PFROPT	パフォーマンス・オプション	値 (最大 2 回の繰り返し): *SETFPCA, *NOSETFPCA, *NOSTRDONLY, *STRDONLY	オプション
PRFDTA	プロファイル作成データ	<u>*NOCOL</u> , *COL	オプション
TERASPACE	テラスペース・オプション	単一値: <u>*NO</u> その他の値: 要素リスト	オプション
	要素 1: テラスペース使用可能	<u>*YES</u>	
	要素 2: TERASPACE INTERFACESの使用	<u>*NOTSIFC</u> , *TSIFC	
STGMDL	ストレージ・モデル	<u>*SNGLVL</u> , *TERASPACE	オプション
DTAMD	データ・モデル	<u>*P128</u> , *LLP64	オプション
RTBND	ランタイム・バインディング	<u>*DEFAULT</u> , *LLP64	オプション
PACKSTRUCT	パック構造	<u>*NATURAL</u> , 1, 2, 4, 8, 16	オプション
ENUM	Enumサイズ	<u>*SMALL</u> , 1, 2, 4, *INT	オプション
MAKEDEP	依存関係情報	パス名, <u>*NONE</u>	オプション
INCDIR	組み込みディレクトリー	単一値: <u>*NONE</u> その他の値 (最大 32 回の繰り返し): パス名	オプション
CSOPT	コンパイラ・サービスOPT	文字値, <u>*NONE</u>	オプション
LICOPT	ライセンス内部コードOPT	文字値, <u>*NONE</u>	オプション
DFTCHAR	省略時の文字タイプ	<u>*UNSIGNED</u> , *SIGNED	オプション
TGTCCSID	ターゲットCCSID	1-65535, <u>*SOURCE</u> , *JOB, *HEX	オプション

キーワード	記述	選択項目	ノーツ
TEMPLATE	テンプレート・オプション	要素リスト	オプション
	要素 1: 一時組み込みディレクトリー	パス名, <u>*NONE</u> , *TEMPINC	
	要素 2: 生成済み最大見出し数	1-99999, <u>1</u>	
	要素 3: テンプレートの妥当性検査	<u>*NO</u> , *WARN, *ERROR	

上

---

## プログラム (PGM)

作成されるプログラム・オブジェクトを指定します。

### 修飾子1: プログラム

**name** プログラム・オブジェクトの名前を指定してください。

### 修飾子2: ライブラリー

#### \*CURLIB

プログラム・オブジェクトは現行ライブラリーに保管されます。ジョブに現行ライブラリーがない場合には、プログラム・オブジェクトはQGPLライブラリーの中に作成されます。

**name** 作成中のプログラム・オブジェクトの保管先となるライブラリーの名前を指定してください。

上

---

## ソース・ファイル (SRCFILE)

コンパイルが必要なILE C++ソース・コードが入っているソース・ファイルを指定します。

### 修飾子1: ソース・ファイル

#### QCPPSRC

QCPPSRCという名前のソース・ファイルには、コンパイルしたいILE C++ソース・コードを含むメンバーが入っています。

**name** ILE C++ソース・コードが含まれたメンバーが入っているソース・ファイルの名前を指定してください。

### 修飾子2: ライブラリー

\*LIBL ソース・ファイルが入っているライブラリーを見つけるためにライブラリー・リストが検索されます。

#### \*CURLIB

ソース・ファイルが現行ライブラリーから検索されます。ジョブに現行ライブラリーがない場合には、ソース・ファイルはQGPLから検索されます。

**name** ソース・ファイルが入っているライブラリーの名前を指定してください。

上

---

## ソース・メンバー (SRCMBR)

コンパイルされるソース・コードが入っているメンバーの名前を指定します。

**\*PGM プログラム (PGM)** パラメーターに指定された名前は、ソース・メンバー名として使用されます。

**name** ソース・コードが入っているメンバーの名前を指定してください。

上

---

## ソース・ストリーム・ファイル (SRCSTMF)

コンパイルが必要なソース・コードが入っているストリーム・ファイルを指定します。

パス名は絶対修飾パス名か相対修飾パス名のいずれかとすることができます。絶対パス名は'/'で始まり、相対パス名は'/'以外の文字で始まります。絶対修飾した場合には、そのパス名で完全です。相対修飾した場合には、パス名は、ジョブの現行作業ディレクトリーをパス名に対して事前保留することによって完全なものとなります。

SRCMBRおよびSRCFILEパラメーターをSRCSTMFパラメーターと一緒に指定することはできません。

**path-name**

C++ソース・コードが入っているストリーム・ファイルのパス名を指定してください。

上

---

## テキスト'記述' (TEXT)

プログラム・オブジェクトを簡潔に記述するテキストを指定します。

**\*SRCMBRTXT**

ソース・ファイル・メンバーに関連付けられたテキスト記述が、プログラム・オブジェクトに使用されます。ソース・ファイルがインライン・ファイル、ストリーム・ファイル、または装置ファイルである場合には、テキストはブランクとなります。

**\*BLANK**

テキストは表示されないことを指定します。

**文字値** 50文字を超えないテキストをアポストロフィで囲んで指定します。

上

---

## 出力オプション (OUTPUT)

コンパイラー・リストが生成されるかどうかを指定します。

**単一値**

**\*NONE**

コンパイラー・リストは生成しません。リストが必要でない場合には、コンパイル時のパフォーマンスを改善するためにこのパラメーター値を使用する必要があります。\*NONEを指定した場合には、OPTIONパラメーターに指定されたリストに関連したすべてのパラメーター値が無視されます。

## 要素1: 出力ファイル名

### **\*PRINT**

リストを含むスプール・ファイルが生成されます。

### ***path-name***

リストを保留するストリーム・ファイルのパス名を指定します。

## 要素2: タイトル

### **\*BLANK**

テキストは表示されないことを指定します。

**文字値** リスト・ファイルのタイトル文字列（最大80文字）を指定します。

## 要素3: サブタイトル

### **\*BLANK**

テキストは表示されないことを指定します。

**文字値** リスト・ファイルのサブタイトル文字列（最大80文字）を指定します。

上

---

## コンパイラー・オプション (OPTION)

ILE C++ソース・コードのコンパイル時に使用するオプションを指定します。それらは、1つまたは複数のブランクで区切り、任意の順序で指定できます。1つのオプションが複数回指定されたり、2つのオプションが競合する場合には、最後に指定されたものが使用されます。

このパラメーターには35個の値を指定できます。

このパラメーターには単一のデフォルト値はありません。以下に示す一連のパラメーター値の説明では、下線付きの値は、コンパイラーで使用するデフォルト値を表します。

### **\*NOBITSIGN**

ビット・フィールドを符号なしとして指定します。

### **\*BITSIGN**

ビット・フィールドを符号付きとして指定します。

### **\*NOEVENTF**

CoOperative Development Environment/400 (CODE/400)で使用するためのイベント・ファイルは作成しません。

### **\*EVENTF**

CoOperative Development Environment/400 (CODE/400)で使用するためのイベント・ファイルを作成します。イベント・ファイルは、作成中のモジュールまたはプログラム・オブジェクトの保管先のライブラリーにあるファイルEVFEVENTにメンバーとして作成されます。ファイルEVFEVENTが存在しない場合には、それが自動的に作成されます。イベント・ファイル・メンバー名は作成中のオブジェクトの名前と同じです。CODE/400はこのファイルを使用して、CODE/400エディターに統合されたエラー・フィードバックを提供します。通常、イベント・ファイルは、CODE/400内からモジュールまたはプログラム・オブジェクトを作成するときに生成されます。

### **\*NOEXPMAC**

マクロの中で構文エラーが検出されないかぎり、マクロはリスト中で拡張されません。

**\*EXPMAC**

リスト中ですべてのマクロを拡張します。

**\*NOFULL**

すべてのリスト・オプションをオンにしません。

**\*FULL**

すべてのリスト・オプションをオンにします。

**\*NOINCDIRFIRST**

INCDIRパラメーターとして指定された組み込みディレクトリーは、標準ヘッダー・ファイル組み込みパスの前には組み込まれません。

**\*INCDIRFIRST**

INCDIRパラメーターとして指定された組み込みディレクトリーが、標準ヘッダー・ファイル組み込みパスの前に組み込まれます。

**\*LOGMSG**

コンパイル・メッセージをジョブ・ログに書き込みます。

このオプションとFLAGパラメーターを指定した場合には、FLAGパラメーターに指定された（およびそれ以上の）重大度のメッセージがジョブ・ログに入れます。

このオプションと、MSGLMTパラメーターにメッセージの最大数を指定した場合には、指定された重大度のその数のメッセージがジョブ・ログに入れられた時にコンパイルが停止します。

**\*NOLOGMSG**

コンパイル・メッセージをジョブ・ログに書き込みません。

**\*LONGLONG**

LONG LONGデータ・タイプの使用を許可します。

**\*NOLONGLONG**

LONG LONGデータ・タイプの使用は許可しません。

**\*NORTTI**

実行時タイプID (RTTI)情報を生成しません。

**\*RTTIALL**

RTTI typeidおよびdynamic\_cast演算子に必要な情報を生成します。

**\*RTTITYPE**

RTTI typeid演算子に必要な情報のみを生成します。

**\*RTTICAST**

dynamic\_cast演算子に必要な情報のみを生成します。

**\*NOSHOWINC**

リストのソース・セクションで、またはデバッグ・ビューでユーザー組み込みファイルまたはシステム組み込みファイルを拡張しません。

**\*SHOWINC**

リストのソース・セクションで、またはデバッグ・ビューでユーザー組み込みファイルとシステム組み込みファイルの両方を拡張します。OUTPUTオプションか、\*ALL, \*SOURCE,または\*LISTの値のDBGVIEWパラメーターを指定する必要があります。

**\*SHOWSRC**

リストにソース・コードを表示します。このオプションは、\*SHOWINC, \*SHOWSYS,または\*SHOWUSRオプションによって変更することができます。

#### **\*NOSHOWSRC**

リストにソース・コードは表示しません。このオプションは、\*EXPMAC、\*SHOWINC、\*SHOWSYS、または\*SHOWUSRオプションで変更できます。

#### **\*NOSHOWSYS**

リストのソース・セクションで、またはデバッグ・ビューで#includeディレクティブ上のシステム組み込みファイルを拡張しません。システム組み込みファイルは、includeディレクティブの後に続けて不等号括弧(< >)で囲まれます。

#### **\*SHOWSYS**

リストのソース・セクションで、またはデバッグ・ビューで#includeディレクティブ上のシステム組み込みファイルを拡張します。OUTPUTオプションを指定するか、あるいは\*ALL、\*SOURCE、または\*LISTのDBGVIEWパラメーターを指定する必要があります。システム組み込みファイルは、includeディレクティブの後に続けて不等号括弧(< >)で囲まれます。

#### **\*NOSHOWUSR**

リストまたはデバッグ・ビューでユーザー組み込みファイルは拡張しません。ユーザー組み込みファイルは、includeディレクティブの後に続けて二重引用符(" ")で囲まれます。

#### **\*SHOWUSR**

リストのソース・セクションで、またはデバッグ・ビューでincludeディレクティブ上のユーザー組み込みファイルを拡張します。OUTPUTオプションか、\*ALL、\*SOURCE、または\*LISTの値のDBGVIEWパラメーターを指定する必要があります。ユーザー組み込みファイルは、includeディレクティブの後に続けて二重引用符(" ")で囲まれます。

#### **\*STDINC**

システム提供のヘッダー・ファイルがコンパイルのための検索パスに組み込まれます。

#### **\*NOSTDINC**

システム提供のヘッダー・ファイルはコンパイルのための検索パスに組み込まれません。

#### **\*NOSTDLOGMSG**

コンパイル・メッセージはSTDOUTストリームへ送られません。

#### **\*STDLOGMSG**

コンパイル・メッセージがSTDOUTストリームへ送られます。

#### **\*NOSYSINCPATH**

ユーザー組み込みファイルの検索パスは影響を受けません。

#### **\*SYSINCPATH**

ユーザー組み込みファイルの検索パスをシステム組み込みファイルの検索パスに変更します。機能的に、このオプションを使用することは、ユーザーincludeディレクティブ(include "file\_name")の二重引用符を不等号括弧(include <file\_name>)に変更することに相当します。

#### **\*NOXREF**

リストの中に相互参照テーブルは生成しません。

#### **\*XREF**

ソース・コード中の識別コードをそれらが表示される行の番号とともに示すリストを含む相互参照テーブルを生成します。OUTPUTオプションの指定が必要です。

#### **\*NOXREFREF**

リストの中で参照される識別コードの相互参照テーブルは生成しません。

### **\*XREFREF**

リスト・ファイルで参照される変数、構造および関数名の相互参照テーブルを生成します。このテーブルには、識別コードが宣言された場所の行番号が示されます。OUTPUTオプションの指定が必要です。

上

---

## **チェックアウト・オプション (CHECKOUT)**

考えられるプログラミング・エラーを指示する通知メッセージの生成を選択できるオプションを指定します。1つのオプションを複数回指定したり、2つのオプションが競合する場合には、最後に指定されたものが使用されます。

このパラメーターには45個の値を指定できます。

### **\*NONE**

CHECKOUTのためのすべてのオプションを使用不可とします。

### **\*USAGE**

これは\*CONDを指定するのと同等です。

**\*ALL** CHECKOUTのためのすべてのオプションを使用可能とします。

以下に示す一連のパラメーター値の説明では、下線付きの値は、コンパイラーで使用するデフォルト値を表します。

### **\*NOCLASS**

クラスの使用に関する情報は表示しません。

### **\*CLASS**

クラスの使用に関する情報を表示します。

### **\*NOCOND**

条件式で考えられる冗長度または問題についての警告はしません。

### **\*COND**

条件式で考えられる冗長度または問題について警告します。

### **\*NOEFFECT**

有効でないステートメントについての警告はしません。

### **\*EFFECT**

有効でないステートメントについて警告します。

### **\*NOGENERAL**

一般チェックアウト・メッセージは生成しません。

### **\*GENERAL**

一般チェックアウト・メッセージを生成します。

### **\*NOLANG**

言語レベルの影響に関する情報は表示しません。

### **\*LANG**

言語レベルの影響に関する情報を表示します。



#### **\*NOPARM**

未使用パラメーターについての警告はしません。

#### **\*PARM**

未使用パラメーターについて警告します。

#### **\*NOPORT**

移植不能言語構造についての警告はしません。

#### **\*PORT**

移植不能言語構造について警告します。

#### **\*NOREACH**

到達不能ステートメントについての警告はしません。

#### **\*REACH**

到達不能ステートメントについて警告します。

#### **\*NOTEMP**

コンパイラーが一時変数を作成した場合にメッセージは生成しません。

#### **\*TEMP**

コンパイラーが一時変数を作成した場合にメッセージを生成します。

#### **\*NOTRUNC**

データの考えられる切り捨てまたは喪失についての警告はしません。

#### **\*TRUNC**

データの考えられる切り捨てまたは喪失について警告します。

#### **\*NOUNUSED**

未使用の自動または静的変数は検査しません。

#### **\*UNUSED**

未使用の自動または静的変数を検査します。

上

---

## **最適化 (OPTIMIZE)**

生成されるオブジェクトの最適化のレベルを指定します。

- 10** 生成されるコードは最適化されません。このレベルでのコンパイル時間は最短です。このレベルでは、デバッグ中に変数を表示して変更することができます。
- 20** コードについて一部の最適化が実行されます。このレベルでは、デバッグ中にユーザーが変数を表示することができますが、変更することはできません。
- 30** 生成されるコードについて完全な最適化が実行されます。デバッグ・セッション中は、ユーザー変数は変更することができませんが、表示することはできます。表示される値は、変数の現行値でない場合があります。
- 40** レベル30で実行されたすべての最適化が生成されるコードについて実行されます。さらに、コードは、命令トレースおよび呼び出しトレース・システム機能を使用可能にするプロシージャーのプロローグおよびエピローグ・ルーチンから除去されます。このコードを除去することによって、リーフ・プロシージャーの作成が可能になります。リーフ・プロシージャーとは、他のプロシージャーに対する呼び出しを含まないプロシージャーです。リーフ・プロシージャーに対するプロシージャー呼び出しのパフォーマンスは、通常のプロシージャーに対する呼び出しより高速となります。



---

## インライン・オプション (INLINE)

コンパイラーが呼び出し先関数の命令による関数呼び出しの置き換えを考慮する必要があるかどうかを指定します。関数をインライン化することによって、呼び出しのオーバーヘッドが取り除かれるので、結果としてより良好に最適化することができます。何度も呼び出される小さな関数は、インライン化の候補として適しています。

### 要素1: インライナー

インライン化を使用するかどうかを指定します。

**\*OFF** コンパイル単位にインライン化は実行されないことを指定します。

**\*ON** コンパイル単位にインライン化が実行されることを指定します。デバッグ・ビューが指定された場合には、インライン化はオフにされます。

### 要素2: モード

そのしきい値および限界値に基づいて、インライナーは関数のインライン化を自動的に試みる必要があるかどうかを指定します。

#### **\*AUTO**

インライナーは指定されたしきい値および限界値に基づいて関数をインライン化できるかどうかを判断する必要があることを指定します。**\*AUTO**は**#PRAGMA NOINLINE**ディレクティブによって上書きされます。

#### **\*NOAUTO**

**#PRAGMA INLINE**ディレクティブによって指定された関数のみをインライン化の候補として考慮する必要があることを指定します。

### 要素3: しきい値

自動インライン化の候補にできる関数の最大サイズを指定します。このサイズは、抽象コード単位(ACU)で測定されます。ACUのサイズは関数中の実行可能コードと比例します。ソース・コードは、コンパイラーによってACUに変換されます。

**250** 250のしきい値を指定します。

**1-65535**

ACUのしきい値を指定します。

#### **\*NOLIMIT**

しきい値をプログラム・オブジェクトの最大サイズとして定義します。

### 要素4: 限界

自動インライン化を停止する前に関数を拡張できる最大相対サイズを指定します。

**2000** 2000 ACUの限界を指定します。

#### **\*NOLIMIT**

限界はプログラム・オブジェクトの最大サイズとして定義されます。システム限界が検出されることがあります。

ACUの限界を指定してください。

## 要素5: 報告書

コンパイラー・リストともにインライナー報告書を生成するかどうかを指定します。

**\*NO** インライナー報告書は生成されません。

**\*YES** インライナー報告書がコンパイラー・リストの一部として生成されます。インライナー報告書を生成するには、OUTPUTオプションの指定が必要です。

上

---

## デバッグ・ビュー (DBGVIEW)

作成されたプログラム・オブジェクトでモジュールに対して使用できるデバッグのレベルを指定します。ソース・レベルのデバッグ用にどのソース・ビューが使用可能かも指定されます。デバッグ・ビューを要求すると、インライン化はオフにされます。

### **\*NONE**

デバッグ機能はプログラム・オブジェクトに挿入されません。

**\*ALL** すべてのデバッグ・オプション(\*STMT, \*SOURCE,および\*LIST)が使用可能になります。

### **\*STMT**

プログラム・ステートメント番号およびシンボリックIDを使用してプログラム・オブジェクトをデバッグできるようにします。

注: \*STMTオプションを使用してモジュール・オブジェクトをデバッグするには、リストが必要です。

### **\*SOURCE**

プログラム・オブジェクトのデバッグのためのソース・ビューを生成します。OPTIONパラメーターの値\*NOSHOWINC, \*SHOWINC, \*SHOWSYS,および\*SHOWUSRによって、作成されるソース・ビューの内容が決まります。

注: このビューをデバッグで使用するためには、プログラム・オブジェクトの作成後に、ルート・ソース・ファイルの変更、名前変更、または移動を行わないようにしてください。

**\*LIST** プログラム・オブジェクトのデバッグのためのリスト・ビューを生成します。OPTIONパラメーターの値\*SHOWINC, \*SHOWUSR, \*SHOWSYS, および\*NOSHOWINCによって、作成されるリスト・ビューの内容が決まります。

上

---

## 名前の定義 (DEFINE)

ファイルがコンパイラーによって処理される前に有効となるプリプロセッサ・マクロを指定します。形式DEFINE(MACRO)の使用はDEFINE('MACRO=1')と同等です。

### 単一値

### **\*NONE**

マクロは定義されません。

その他の値（反復は最大32回まで）

‘名前’または‘名前=値’

最大32個のマクロを定義できます。各マクロ名はアポストロフィで囲まれます。マクロ名の最大長は80文字です。アポストロフィは80の文字列の一部ではありません。アポストロフィは、大文字小文字を区別するマクロに必要です。

注: コマンドで定義されたマクロによって、ソースにある同じ名前のすべてのマクロ定義が上書きされますが、コンパイラにより警告メッセージが生成されます。 `#define max(a,b) ((a)>(b)?(a):(b))` など関数と類似したマクロをコマンド行で定義することはできません。

上

---

## 言語レベル (LANGVL)

コンパイラの機能と、ソースの作成時に宣言されるプロトタイプを指定します。

### **\*EXTENDED**

プリプロセッサ変数 `__EXTENDED__` を定義し、他の言語レベル変数は未定義とします。このパラメーターは、ILE C++の機能が必要な場合に使用する必要があります。

**\*ANSI** プリプロセッサ変数 `__ANSI__`, `__STDC__`, および `__cplusplus98_interface__` を定義し、他の言語レベル変数を未定義にします。ANSI標準C++だけが使用可能とされます。

### **\*LEGACY**

このオプションによって、一部のソース構成が初期のコンパイラで受け入れ可能になります。

上

---

## 別名 (ALIAS)

作成中のプログラム・オブジェクトで適用される別名割り当てアサーションを指定します。

このパラメーターには3個の値を指定できます。

**\*ANSI** プログラム・オブジェクトで、ポインターが、同一タイプのオブジェクトのみを指し示すようになります。

### **\*NOANSI**

プログラム・オブジェクトは、\*ANSI別名割り当て規則を使用しません。

### **\*ADDRTAKEN**

プログラム・オブジェクトで、変数のアドレスが使用されない場合に、その変数のクラスがポインターから切り離されます。

### **\*NOADDRTAKEN**

プログラム・オブジェクトは、\*ADDRTAKEN別名割り当て規則を使用しません。

### **\*ALLPTRS**

プログラム・オブジェクトで、2つのポインターに別名を割り当てることができなくなります。

### **\*NOALLPTRS**

プログラム・オブジェクトは、\*ALLPTRS別名割り当て規則を使用しません。

#### **\*TYPEPTR**

プログラム・オブジェクトで、タイプの異なる2つのポインターに別名を割り当てることができなくなります。

#### **\*NOTYPEPTR**

プログラム・オブジェクトは、\*TYPEPTR別名割り当て規則を使用しません。

上

---

## **SYSTEMインターフェースOPT (SYSIFCOPT)**

作成中のプログラム・オブジェクトに使用されるシステム・インターフェース・オプションを指定します。

#### **\*IFS64IO**

プログラム・オブジェクトは、64ビットCストリーム入出力操作に統合ファイル・システムを使用します。

#### **\*IFSIO**

プログラム・オブジェクトは、Cストリーム入出力操作に統合ファイル・システムを使用します。

#### **\*NOIFSIO**

プログラム・オブジェクトは、Cストリーム入出力操作にデータ管理機能ファイル・システムを使用します。

上

---

## **LOCALEオブジェクト・タイプ (LOCALETYPE)**

作成中のプログラム・オブジェクトで使用されるロケール・サポートのタイプを指定します。

#### **\*LOCALE**

このオプションで作成されたプログラム・オブジェクトは、\*LOCALEオブジェクトによって指定されるロケール・サポートを使用します。

#### **\*LOCALEUCS2**

このオプションで作成されたプログラム・オブジェクトは、\*LOCALEオブジェクトによって指定されるロケール・サポートを使用します。広幅(WIDE)文字タイプには2つの2バイト汎用文字セットの値が含まれることになります。

#### **\*LOCALEUTF**

このオプションで作成されたプログラム・オブジェクトは、\*LOCALEオブジェクトによって指定されるロケール・サポートを使用します。広幅(WIDE)文字タイプには、4バイトのUTF-32の値が含まれます。狭幅(NARROW)文字タイプには、UTF-8の値が含まれます。

上

---

## **メッセージのフラグ・レベル (FLAG)**

リストに表示するメッセージのレベルを指定します。

**0** 通知レベルで始まるすべてのメッセージが表示されます。

**10** 警告レベルで始まるすべてのメッセージが表示されます。

- 20 エラー・レベルで始まるすべてのメッセージが表示されます。
- 30 重大エラー・レベルで始まるすべてのメッセージが表示されます。

上

---

## メッセージ限界 (MSG LMT)

指定されたメッセージ重大度(またはそれ以上)のメッセージの最大数を指定します。コンパイルが停止するまでに、このメッセージ数までメッセージを生成できます。

### 要素1: メッセージ限界

指定したメッセージ重大度レベル以上で生じさせることができるメッセージの最大数を指定します。

#### \*NOMAX

指定されたメッセージ重大度レベルで生じたメッセージの数とは無関係に、コンパイルが続行されます。

#### 0-32767

生じさせることができるメッセージの数を指定します。

### 要素2: メッセージ限界の重大度

メッセージ重大度を指定します。ここで指定した重大度以上のメッセージが特定の数（要素1に指定された数）だけ生成されるとき、コンパイルを停止できます。

- 30 コンパイルが停止するまでに、重大度30のメッセージを指定の数だけ生成できます。
- 0 コンパイルが停止するまでに、重大度0以上のメッセージを指定の数だけ生成できます。
- 10 コンパイルが停止するまでに、重大度10以上のメッセージを指定の数だけ生成できます。
- 20 コンパイルが停止するまでに、重大度20以上のメッセージを指定の数だけ生成できます。

上

---

## PGMオブジェクトの置き換え (REPLACE)

オブジェクトの既存のバージョンを現行バージョンで置き換えるかどうかを指定します。

\*YES 既存のオブジェクトが新規バージョンで置き換えられます。旧バージョンはQRPLOBJライブラリーに移動し、システム日付および時刻に基づいて名前変更されます。置き換えられたオブジェクトのテキスト記述は、オリジナルのオブジェクトの名前に変更されます。旧オブジェクトは、それが明示的に削除されていなければ、次のIPL時に削除されます。

\*NO 既存のオブジェクトは置き換えられません。指定のライブラリーで同じ名前のオブジェクトが見つかった場合、メッセージが表示されて、コンパイルが停止します。

上

---

## ユーザー・プロファイル (USRPRF)

プログラム・オブジェクトがオブジェクトごとに持つ権限も含めて、作成されたプログラム・オブジェクトの実行時に使用されるユーザー・プロファイルを指定します。プログラム・オブジェクトが使用できるオブジェクトの制御には、プログラム所有者かプログラム・ユーザーのどちらかのプロファイルが使用されます。

### **\*USER**

プログラム・オブジェクトを実行中のユーザーのプロファイルが使用されます。

### **\*OWNER**

プログラム・オブジェクトの処理時には、プログラム所有者とプログラム・ユーザーの両方のユーザー・プロファイルが使用されます。両方のユーザー・プロファイルの中のオブジェクト権限の集合セットが、プログラムの処理中にオブジェクトを検出してアクセスするために使用されます。所有者側のユーザー・プロファイルのグループ・プロファイルの権限は、実行されるプログラム・オブジェクトの権限には含まれません。

上

---

## 権限 (AUT)

オブジェクトに対する特定権限がないユーザー、権限リスト上にないユーザー、またはそのグループにオブジェクトに対する特定権限がないユーザーに許可される権限を指定します。

### **\*LIBCRTAUT**

オブジェクトに対する共通権限はターゲット・ライブラリー（作成されたオブジェクトが入れられるライブラリー）のCRTAUTキーワードからとられます。この値は、オブジェクトの作成時に決定されます。オブジェクトの作成後にライブラリーのCRTAUT値が変更された場合には、この新規の値はライブラリー中の既存のどのオブジェクトにも影響しません。

**\*ALL** 所有者に限定されるか、あるいは権限リスト管理権限によって制御される操作を除いて、オブジェクトでのすべての操作に対する権限が提供されます。どのユーザーも、オブジェクトの存在を制御し、その機密保護を指定し、変更し、また、その所有権の転送も含めてオブジェクトに対する基本操作を実行することができます。

### **\*CHANGE**

すべてのデータ権限、および所有者に限定されるか、あるいは権限リスト管理権限によって制御される操作を除いて、オブジェクトでのすべての操作を実行する権限が提供されます。オブジェクトを変更し、それに基本操作を実行することができます。

**\*USE** オブジェクト操作権、読み取り権限、およびモジュール・オブジェクトのバインドなど、オブジェクトへの基本読み取り専用操作に対する権限が提供されます。特定権限のないユーザーは、オブジェクトを変更することができません。

### **\*EXCLUDE**

特殊権限のないユーザーはオブジェクトにアクセスすることができません。

**name** オブジェクトの追加先となるユーザーと権限で構成された権限リストの名前を指定してください。オブジェクトはこの権限リストによって保護され、そのオブジェクトの共通権限は\*AUTLに設定されます。権限リストは、コマンドが出された時にシステム上に存在していなければなりません。

上



---

## ターゲット・リリース (TGTRLS)

ユーザーが作成中のオブジェクトを使用したいオペレーティング・システムのリリースを指定します。

\*CURRENTおよび\*PRVの値について示される例の中、および リリース・レベル の値を指定する場合には、形式VXRXMXを使用してリリースが指定されます。ここで、VXはバージョン、RXはリリース、そしてMXはモディフィケーション・レベルです。たとえば、V4R5M0は、バージョン4、リリース5、モディフィケーション・レベル0です。

### **\*CURRENT**

オブジェクトは、ユーザーのシステムで現在実行されているオペレーティング・システムのリリースで使用されます。例えば、V5R5M0がシステムで実行されている場合に、\*CURRENTは、ユーザーはオブジェクトをV5R5M0の導入されたシステムで使用したいことを意味します。ユーザーは、また、このリリース以降のオペレーティング・システムが導入されたシステムでもオブジェクトを使用することができます。

**\*PRV** オブジェクトは、オペレーティング・システムの前のリリース（モディフィケーション・レベル0）で使用されます。例えば、V5R5M0がシステムで実行されている場合に、\*PRVは、ユーザーはオブジェクトをV5R5M0の導入されたシステムで使用したいことを意味します。ユーザーは、また、このリリース以降のオペレーティング・システムが導入されたシステムでもオブジェクトを使用することができます。

### **リリース・レベル**

リリースを形式VXRXMXで指定します。オブジェクトは、指定したリリースのシステムでか、あるいはこのリリース以降のオペレーティング・システムが導入されたシステムで使用することができます。

有効な値は、現行バージョン、リリース、およびモディフィケーション・レベルによって異なります。これらは各新規リリースごとに変更されます。このコマンドによってサポートされる最も初期のリリース・レベルより前のリリース・レベルを指定した場合には、エラー・メッセージが送られます。

上

---

## パフォーマンス収集の使用可能化 (ENBPFRCOL)

オブジェクト内でパフォーマンス測定コードを生成すべきかどうかを指定します。収集されたデータは、システム・パフォーマンス・ツールによって使用し、アプリケーションのパフォーマンスのプロファイルを作成することができます。作成されたオブジェクト内でパフォーマンス測定コードを生成すると、オブジェクトのサイズがわずかに増えるため、パフォーマンスに影響が及ぶ可能性があります。

注: V5R5M0 以降では、このパラメーターは、作成されたオブジェクトに対して作用しなくなります。このパラメーターは、V5R5M0 より前のリリースとの互換性を保つために用意されています。

**\*PEP** パフォーマンス統計は、プログラム・エントリー・プロシージャーの入り口と出口でのみ収集されます。この値は、アプリケーションの全般的なパフォーマンス情報を収集したい場合に選択します。

### **\*ENTRYEXIT \*NONLEAF**

パフォーマンス統計は、プログラム・オブジェクトの全プロシージャー（リーフ・プロシージャーを除く）の入り口と出口で収集されます。これにはプログラムPEPルーチンが含まれます。

この選択項目は、アプリケーション内の他のルーチンを呼び出すルーチンに関する情報だけを取り込みたい場合に有用です。

#### **\*ENTRYEXIT \*ALLPRC**

パフォーマンス統計は、プログラム・オブジェクトの全プロシージャー（リーフ・プロシージャーを含む）の入り口と出口で収集されます。これには、プログラムPEPルーチンが含まれます。

この選択項目は、すべてのルーチンに関する情報を取り込みたい場合に有用です。このオプションは、ユーザーのアプリケーションから呼び出されるすべてのプログラム・オブジェクトが、\*PEP、\*ENTRYEXIT、\*FULLのいずれかのオプションを使用して作成されたことが判明している場合に使用します。それ以外の場合、パフォーマンス測定に対して使用可能にならない他のプログラム・オブジェクトがユーザーのアプリケーションから呼び出されると、パフォーマンス・ツールは、そのプログラム・オブジェクトによる資源の使用についてユーザーのアプリケーションに任せます。このため、資源が実際に使用されている場所を判別するのが困難になることがあります。

#### **\*FULL \*NONLEAF**

パフォーマンス統計は、リーフ・プロシージャーではないすべてのプロシージャーの入り口と出口で収集されます。また、外部プロシージャーに対する個々の呼び出しの前および後にも統計が収集されます。

#### **\*FULL \*ALLPRC**

パフォーマンス統計は、リーフ・プロシージャーを含むすべてのプロシージャーの入り口と出口で収集されます。また、外部プロシージャーに対する個々の呼び出しの前および後にも統計が収集されます。

このオプションは、作成時に\*PEP、\*ENTRYEXIT、\*FULLのいずれも使用されなかった他のプログラム・オブジェクトをユーザーのアプリケーションから呼び出そうと考えている場合に使用します。このオプションを使用すると、パフォーマンス・ツールは、ユーザーのアプリケーションによって使用されている資源と、そのアプリケーションから呼び出されたプログラム・オブジェクトによって使用されている資源を、（そのプログラム・オブジェクトがパフォーマンス測定に対して使用可能にならない場合でも）区別できるようになります。このオプションを使用すると、効率は最も悪くなりますが、アプリケーション内の各種プログラム・オブジェクトを選択的に分析できるようになります。

上

---

## **パフォーマンス・オプション (PFROPT)**

パフォーマンスを高めるために使用可能な各種のオプションを指定します。それらは、1つまたは複数のブランクで区切り、任意の順序で指定できます。1つのオプションが複数回指定されたり、2つのオプションが競合する場合には、最後に指定されたものが使用されます。

このパラメーターには2個の値を指定できます。

#### **\*SETFPCA**

浮動小数点計算に対するANSIのセマンティクスを達成するために、コンパイラーに浮動小数点計算属性を設定させます。

#### **\*NOSETFPCA**

計算属性は設定されません。このオプションは、作成中のオブジェクトに浮動小数点計算が含まれない場合にのみ使用してください。



### **\*NOSTRDONLY**

コンパイラーは文字列を書き込み可能メモリーに入れる必要があることを指定します。

### **\*STRDONLY**

コンパイラーは文字列を読み取り専用メモリーに入れることができることを指定します。

上

---

## **プロファイル作成データ (PRFDTA)**

プログラム・オブジェクトのプログラム・プロファイル作成データ属性を指定します。プログラム・プロファイル作成は、プロシージャおよびプロシージャ内のコードを統計データ（プロファイル作成データ）に基づいて配列し直すために使用される拡張最適化手法です。

### **\*NOCOL**

プログラム・オブジェクトは、プロファイル作成データの収集に使用できません。

**\*COL** プログラム・オブジェクトは、プロファイル作成データの収集に使用できます。**\*COL**は、最適化レベルが30以上である場合にのみ指定することができます。

上

---

## **テラスペース・オプション (TERASPACE)**

テラスペース・ストレージの処理にプログラム・オブジェクトを使用可能にするかどうかを指定します。これには、プログラム・オブジェクトによって割り振られるテラスペース・ストレージや、他のテラスペース対応プログラムおよびサービス・プログラムのオブジェクトから渡されたパラメーターによって割り振られるテラスペース・ストレージが含まれます。

### **単一値**

**\*NO** プログラム・オブジェクトは、テラスペースから割り振られたストレージのアドレス指定の処理に使用できません。

注: V5R5M0以降では、テラスペースから割り振られたストレージのアドレス指定の処理に、すべてのプログラム・オブジェクトが使用できます。ただし、**\*NO**が指定された場合には、以下にリストされているコンパイラー機能が使用できなくなります。

### **要素1: テラスペース使用可能**

**\*YES** プログラム・オブジェクトは、他のテラスペース対応プログラムやサービス・プログラムのオブジェクトから渡されたパラメーターも含めて、テラスペースから割り振られたストレージのアドレス指定の処理に使用できます。さらに、以下のコンパイラー機能も使用できます。

- テラスペース・ストレージへのアクセスに使用される8バイト・ポインターを作成できるように、ポインターは\_\_ptr64で修飾できます。
- テラスペース・ストレージ・モデルは、STGMDL(\*TERASPACE)コンパイラー・オプションを使用して指定できます。
- LLP64データ・モデルは、DTAMD(\*LLP64)コンパイラー・オプションまたは#pragma datamodel(llp64)ディレクティブを使用して指定できます。
- ポインター差分演算は、ptrdiff\_tの結果ではなく、signed long longの結果を戻します。

### **要素2: TERASPACE INTERFACESの使用**

#### **\*NOTSIFC**

プログラム・オブジェクトは、デフォルトで、ストレージ機能の非テラスペース・バージョンを使用するようになります。

#### **\*TSIFC**

プログラム・オブジェクトは、デフォルトで、ストレージ機能のテラスペース・バージョンを使用するようになります。コンパイラーは、マクロ変数\_\_TERASPACE\_\_を定義することになります。

上

---

## **ストレージ・モデル (STGMDL)**

作成されたオブジェクトで使用する記憶域のタイプを指定します。

#### **\*SNGLVL**

作成されたオブジェクトは、単一レベルの記憶域を使用します。

#### **\*TERASPACE**

作成されたオブジェクトは、テラスペース記憶域を使用します。

上

---

## **データ・モデル (DTAMD)**

INT, LONG, POINTERとして宣言される変数のサイズ（バイト数）を指定します。

**\*P128** INT, LONG, POINTERのサイズがそれぞれ4, 4, 16となるようにします。

#### **\*LLP64**

INT, LONG, POINTERのサイズがそれぞれ4, 4, 8となるようにします。コンパイラーは、マクロ\_\_LLP64\_IFC\_\_を定義することになります。

上

---

## **ランタイム・バインディング (RTBND)**

作成されたオブジェクトの実行時バインディング・ディレクトリーを指定します。

#### **\*DEFAULT**

作成されたオブジェクトは、デフォルトのバインディング・ディレクトリーを使用します。

#### **\*LLP64**

作成されたオブジェクトは、64ビット実行時バインディング・ディレクトリーを使用します。この値を使用できるのは、テラスペース・ストレージ・モデル、64ビット・データ・モデル、およびテラスペース・ストレージ機能インターフェースの各オプションと一緒に使用する場合があります。コンパイラーは、マクロ\_\_LLP64\_RTBN\_\_を定義します。

上

---

## **パック構造 (PACKSTRUCT)**

構造のメンバーに使用する位置合わせ境界を指定します。

### **\*NATURAL**

構造メンバーはその自然境界で位置合わせされます。たとえば、短形式整数は2バイトで位置合わせされます。16バイト・ポインターは、常に16バイト境界で位置合わせされます。

- 1 構造メンバーを1バイトの位置合わせでパックします。
- 2 構造メンバーを2バイトの位置合わせでパックします。
- 4 構造メンバーを4バイトの位置合わせでパックします。
- 8 構造メンバーを8バイトの位置合わせでパックします。
- 16 構造メンバーを16バイトの位置合わせでパックします。

上

---

## **Enumサイズ (ENUM)**

コンパイラーが列挙型を表すために使用するバイト数を指定します。

### **\*SMALL**

すべてのENUM変数を、値の範囲を表すことができる最も小さいサイズにします。

- 1 すべてのENUM変数を1バイトにします。
- 2 すべてのENUM変数を2バイトにします。
- 4 すべてのENUM変数を4バイトにします。

**\*INT** ANSI標準のENUMサイズである4バイトを使用します。

上

---

## **依存関係情報 (MAKEDEP)**

依存関係情報をファイルの中に生成するかどうかを指定します。この情報は、MAKE TOOLによって使用されます。

### **\*NONE**

依存関係情報は生成しません。

### ***path-name***

依存関係情報を保管するストリーム・ファイルのパス名を指定します。

上

---

## **組み込みディレクトリー (INCDIR)**

組み込みファイルを検出するためにコンパイラーが使用する検索パスに追加する1つまたは複数のディレクトリーを指定します。

検索パスは、OPTIONキーワードで次のパラメーターを使用することにより、さらに変更することができます。

- \*INCDIRFIRSTまたは\*NOINCDIRFIRST
- \*SYSINCPATHまたは\*NOSYSINCPATH

- \*STDINCまたは\*NOSTDINC

#### **\*NONE**

変更されなければ、省略時のシステム組み込みディレクトリーおよびソース・ディレクトリーからユーザー組み込みファイルが検索されます。

#### ***path-name***

組み込みファイルの検索場所となるディレクトリーのパス名を指定してください。最大32個のディレクトリーを指定できます。指定したディレクトリーに加えて、ソース・ディレクトリーからもユーザー組み込みファイルが検索されます。

上

---

## **コンパイラー・サービスOPT (CSOPT)**

1つまたは複数のコンパイラー・サービス・オプションを指定します。このパラメーターにより、IBMはリリース相互間で切り替え可能なコンパイラーを提供することができます。

#### **\*NONE**

コンパイラー・サービス・オプションは選択されません。

**文字値** モジュール・オブジェクトの作成時に使用するコンパイラー・サービス・オプションを含む文字ストリングを指定してください。有効な文字列は、PTFカバー・レターまたはリリース情報に記述されます。

上

---

## **ライセンス内部コードOPT (LICOPT)**

1つまたは複数のライセンス内部コード・コンパイル時オプションを指定します。このパラメーターは、個別のコンパイル時オプションを選択できるようにするもので、選択された各タイプのコンパイラー・オプションの潜在的な長所と短所を理解している上級プログラマーを対象としています。

#### **\*NONE**

コンパイル時オプションは選択されません。

**文字値** 選択されたライセンス内部コード・コンパイル時オプションが、モジュール・オブジェクトの作成時に使用されます。特定のオプションでは、作成されたモジュール・オブジェクトをデバッグする能力が低下することがあります。

上

---

## **省略時の文字タイプ (DFTCHAR)**

CHARデータ・タイプの省略時の符号を指定します。

#### **\*UNSIGNED**

省略時のCHARタイプを符号なしとします。

#### **\*SIGNED**

省略時のCHARタイプを符号付きとします。

上

---

## ターゲットCCSID (TGTCSSID)

結果のプログラム・オブジェクトに保管されるデータを記述するために使用されるターゲット・コード化文字セットID (CCSID)を指定します。

### \*SOURCE

ルート・ソース・ファイルのCCSIDが使用されます。

**\*JOB** 現行ジョブのCCSIDが使用されます。

**\*HEX** CCSID 65535が使用されます。これは、文字データはバイナリー・データとして取り扱われ、変換されないことを指示します。

### ***1-65535***

使用するCCSIDを指定します。

上

---

## テンプレート・オプション (TEMPLATE)

コンパイラーに対するテンプレート・オプションを指定します。

**要素1: 一時組み込みディレクトリー**

### \*NONE

テンプレートは生成されません。

### **\*TEMPINC**

テンプレートは、ルート・ソース・ファイルが見つかったディレクトリーの中に作成されるTEMPINCという名前のディレクトリーの中に生成されます。ソース・ファイルがストリーム・ファイルでない場合、ソース・ファイルが置かれているライブラリー内に、ファイルTEMPINCが作成されます。TEMPLATE(\*TEMPINC)パラメーターとTMPLREGパラメーターは相互に排他的です。

### ***path-name***

コンパイラーがテンプレートを生成するディレクトリーのパス名を指定してください。

**要素2: 生成済み最大見出し数**

**1** テンプレートが入れられる、生成されるヘッダーの最大数。

### ***1-99999***

生成されるヘッダー・ファイルの最大数を指定してください。

**要素3: テンプレートの妥当性検査**

構文解析とセマンティック検査が、テンプレート定義の実装に適用されるのか、テンプレートのインスタンス化にのみ適用されるのかを制御します。コンパイラーには、警告やエラー・メッセージを生成するオプションを指定できます。使用可能なパラメーターは次のとおりです。

**\*NO** 構文解析時に、前のバージョンのコンパイラーに合わせて作成されたコードで発行されるエラーの数を減らしません。

### **\*WARN**

セマンティック・エラーに関して警告メッセージを発行します。構文解析時に検出されたエラーに関しては、エラー・メッセージを発行します。

## \*ERROR

テンプレートがインスタンス化されていない場合でも、テンプレート実装での問題をエラーとして扱います。

上

---

## 例

### 例1:ソース・ファイル・メンバーからのプログラムの作成

```
CRTBNDCPP  PGM(QTEMP/TEST)  SRCFILE(QTEMP/QCPPSRC)
```

このコマンドは、プログラム・オブジェクトTESTをライブラリーQTEMP内に作成します。C++ソースは、ライブラリーQTEMPにあるソース・ファイルQCPPSRCのメンバーTEST内に指定されます。

### 例2:ストリーム・ファイルからのプログラムの作成

```
CRTBNDCPP  PGM(QTEMP/TEST)  SRCSTMF('/tmp/mySrc.C')
```

このコマンドは、プログラム・オブジェクトTESTをライブラリーQTEMP内に作成します。C++ソースは、ストリーム・ファイル/tmp/mySrc.C内に指定されます。

### 例3:コンパイル・リストのコンパイルおよび生成

```
CRTBNDCPP  PGM(TEST)  OUTPUT(*PRINT)  OPTION(*XREF)
```

このコマンドは、プログラム・オブジェクトTESTをジョブの現行ライブラリー内に作成しますが、現行ライブラリーが定義されていない場合は、ライブラリーQGPL内に作成します。C++ソースは、ソース・ファイルQCPPSRCのメンバーTEST内に指定されると想定されます（これは、ライブラリー・リストを使用して指定されます）。コンパイラ・リストが入ったスプール・ファイルが生成されます。コンパイラ・リスト・ファイルには、相互参照リストが格納されます。

### 例4:すべてのデバッグ・ビューを備えたプログラムの作成

```
CRTBNDCPP  PGM(QTEMP/TEST)  SRCFILE(QTEMP/QCPPSRC)
           SRCMBR(TESTER)  DBGVIEW(*ALL)
```

このコマンドは、デバッグ時に使用されるステートメント、ソース、およびリストの各ビューを備えたプログラム・オブジェクトTESTをライブラリーQTEMP内に作成します。C++ソースは、ライブラリーQTEMPにあるソース・ファイルQCPPSRCのメンバーTESTER内に指定されます。

### 例5:テラスペース対応プログラムの作成

```
CRTBNDCPP  PGM(QTEMP/TEST)  SRCFILE(QTEMP/QCPPSRC)
           TERASPACE(*YES *TSIFC)  STGMDL(*TERASPACE)
           DTAMD(*LLP64)  RTBND(*LLP64)
```

このコマンドは、プログラム・オブジェクトTESTをライブラリーQTEMP内に作成します。C++ソースは、ライブラリーQTEMPにあるソース・ファイルQCPPSRCのメンバーTEST内に指定されます。このモジュールは、テラスペース・ストレージを使用する場合に使用可能にされます。ヒープ・スペース、自動ストレージ、および静的ストレージは、テラスペース・ストレージ内に入れます。デフォルト・ポインター・サイズは8バイトになり、8バイト・ランタイムが使用されます。

上

---

## エラー・メッセージ

\*ESCAPE メッセージ

**CZS1613**

コンパイルに失敗しました。

上





## バインドRPG PGMの作成 (CRTBNDRPG)

実行可能場所: すべての環境 (\*ALL)

スレッド・セーフ: いいえ

パラメーター

例

エラー・メッセージ

バインドRPGプログラム作成(CRTBNDRPG)コマンドは、RPGソース・コードをコンパイルしてバインドし、ILEプログラム・オブジェクト(\*PGM)を作成します。このコマンドは、対話モードでもバッチ・モードでも使用することができます。

上

### パラメーター

キーワード	記述	選択項目	ノート
PGM	プログラム	修飾オブジェクト名	オプション, 定位置 1
	修飾子 1: プログラム	名前, <u>*CTLSPEC</u>	
	修飾子 2: ライブラリー	名前, <u>*CURLIB</u>	
SRCFILE	ソース・ファイル	修飾オブジェクト名	オプション, 定位置 2
	修飾子 1: ソース・ファイル	名前, <u>QRPGLESRC</u>	
	修飾子 2: ライブラリー	名前, <u>*LIBL</u> , <u>*CURLIB</u>	
SRCMBR	ソース・メンバー	名前, <u>*PGM</u>	オプション, 定位置 3
SRCSTMF	ソース・ストリーム・ファイル	パス名	オプション
GENLVL	生成重大度レベル	0-20, <u>10</u>	オプション
TEXT	テキスト'記述'	文字値, <u>*SRCMBRTXT</u> , *BLANK	オプション
DFTACTGRP	省略時の活動化グループ	<u>*YES</u> , *NO	オプション
ACTGRP	活動化グループ	名前, <u>QILE</u> , *NEW, *CALLER	オプション
BNDDIR	BINDINGディレクトリー	単一値: <u>*NONE</u> その他の値 (最大 50 回の繰り返し): 修飾オブジェクト名	オプション
	修飾子 1: BINDINGディレクトリー	名前	
	修飾子 2: ライブラリー	名前, <u>*LIBL</u> , <u>*CURLIB</u> , *USRLIBL	
OPTION	コンパイラー・オプション	値 (最大 20 回の繰り返し): *XREF, *NOXREF, *GEN, *NOGEN, *SECLVL, *NOSECLVL, *SHOWCPY, *NOSHOWCPY, *EXPDDS, *NOEXPDDS, *EXT, *NOEXT, *NOSHOWSKP, *SHOWSKP, *NOSRCSTMT, *SRCSTMT, *DEBUGIO, *NODEBUGIO, *UNREF, *NOUNREF, *NOEVENTF, *EVENTF	オプション

キーワード	記述	選択項目	ノーツ
DBGVIEW	デバッグ用ビュー	<u>*STMT</u> , *SOURCE, *LIST, *COPY, *ALL, *NONE	オプション
OUTPUT	出力	<u>*PRINT</u> , *NONE	オプション
OPTIMIZE	最適化レベル	<u>*NONE</u> , *BASIC, *FULL	オプション
INDENT	ソース・リストの字下げ	文字値, <u>*NONE</u>	オプション
CVTOPT	タイプ変換オプション	単一値: <u>*NONE</u> その他の値 (最大 4 回の繰り返し): *DATETIME, *GRAPHIC, *VARCHAR, *VARGRAPHIC	オプション
SRTSEQ	ソート順序	単一値: <u>*HEX</u> , *JOB, *JOBRUN, *LANGIDUNQ, *LANGIDSHR その他の値: 修飾オブジェクト名	オプション
	修飾子 1: ソート順序	名前	
	修飾子 2: ライブラリー	名前, <u>*LIBL</u> , *CURLIB	
LANGID	言語識別コード	名前, <u>*JOBRUN</u> , *JOB	オプション
REPLACE	プログラムの置き換え	<u>*YES</u> , *NO	オプション
USRPRF	ユーザー・プロファイル	<u>*USER</u> , *OWNER	オプション
AUT	権限	名前, <u>*LIBCRTAUT</u> , *ALL, *CHANGE, *USE, *EXCLUDE	オプション
TRUNCNBR	数字の切り捨て	<u>*YES</u> , *NO	オプション
FIXNBR	数値の修正	単一値: <u>*NONE</u> その他の値 (最大 2 回の繰り返し): *ZONED, *INPUTPACKED	オプション
TGTRLS	ターゲット・リリース	単純名, <u>*CURRENT</u> , *PRV	オプション
ALWNULL	ヌル値可能	<u>*NO</u> , *INPUTONLY, *USRCTL, *YES	オプション
DEFINE	条件名の定義	値 (最大 32 回の繰り返し): 単純名, <u>*NONE</u>	オプション
ENBPFRCOL	パフォーマンス収集使用可能化	<u>*PEP</u> , *ENTRYEXIT, *FULL	オプション
PRFDTA	プロファイル・データ	<u>*NOCOL</u> , *COL	オプション
LICOPT	ライセンス内部コード・オプション	文字値, <u>X</u>	オプション
INCDIR	組み込みディレクトリー	値 (最大 32 回の繰り返し): パス名, <u>*NONE</u>	オプション
PGMINFO	Prgmインターフェース情報	要素リスト	オプション
	要素 1: 生成	<u>*NO</u> , *PCML	
	要素 2: 場所	<u>*STMF</u> , *MODULE, *ALL	
INFOSTMF	プログラム・インターフェース・ストリーム・ファイル	パス名	オプション
PPGENOPT	プリプロセッサ・オプション	単一値: <u>*NONE</u> , *DFT その他の値 (最大 3 回の繰り返し): *RMVCOMMENT, *NORMVCOMMENT, *EXPINCLUDE, *NOEXPINCLUDE, *SEQSRC, *NOSEQSRC	オプション

キーワード	記述	選択項目	ノート
PPSRCFILE	出力ソース・ファイル	修飾オブジェクト名	オプション
	修飾子 1: 出力ソース・ファイル	名前	
	修飾子 2: ライブラリー	名前, <b>*CURLIB</b>	
PPSRCMBR	出力ソース・メンバー	名前, <b>*PGM</b>	オプション
PPSRCSTMF	出力ストリーム・ファイル	パス名, <b>*SRCSTMF</b>	オプション

上

## プログラム (PGM)

作成しているプログラム・オブジェクト(\*PGM)のプログラム名およびライブラリー名を指定します。プログラム名およびライブラリー名は、サーバーの命名規則に従っていなければなりません。ライブラリーを指定しない場合には、作成されたプログラムは現行ライブラリーに保管されます。

### \*CTLSPEC

コンパイル済みプログラムの名前は、制御仕様書のDFTNAMEキーワードに指定された名前から取られます。制御仕様書にプログラム名が指定されていないで、ソース・メンバーがデータベース・ファイルから取られる場合には、SRCMBRパラメーターに指定されたメンバー名がプログラム名として使用されます。ソースがデータベース・ファイルから取られない場合には、省略時の値としてプログラム名にRPGPGMが使用されます。

### プログラム名

プログラム・オブジェクトの名前を入力してください。

### \*CURLIB

作成されたプログラム・オブジェクトは現行ライブラリーに保管されます。現行ライブラリーが指定されていない場合には、QGPLが使用されます。

### ライブラリー名

作成されたプログラム・オブジェクトが保管されるライブラリーの名前を入力してください。

上

## ソース・ファイル (SRCFILE)

コンパイルされるILE RPGソース・メンバーが入っているソース・ファイルおよびソース・ファイルが保管されているライブラリーの名前を指定します。望ましいソース物理ファイルの長さは112文字で、内12文字は順序番号および日付用で、80文字はコード用で、20文字は注記用です。これは、コンパイル・リストに示されるソースの最大容量です。

### QRPGLESRC

省略時のソース・ファイルQRPGLESRCにコンパイルされるILE RPGソース・メンバーが入っています。

### ソース・ファイル名

コンパイルされるILE RPGソース・メンバーが入っているソース・ファイルの名前を入力してください。

**\*LIBL** ソース・ファイルが保管されているライブラリーを見つけるために、システムはライブラリー・リストを検索します。これが省略時の値です。

## \*CURLIB

ソース・ファイルを見つけるために、現行ライブラリーが使用されます。現行ライブラリーが指定されていない場合には、QGPLが使用されます。

### ライブラリー名

ソース・ファイルが入っているライブラリーの名前を入力してください。

上

---

## ソース・メンバー (SRCMBR)

コンパイルされるILE RPGソース・プログラムが入っているソース・ファイルのメンバーの名前を指定します。

**\*PGM** ソース・ファイル・メンバー名としてPGMパラメーターに指定された名前を使用します。コンパイル済みプログラム・オブジェクトは、ソース・ファイル・メンバーと同じ名前をもちます。PGMパラメーターでプログラム名が指定されていない場合には、コマンドは、ソース・ファイルに最初に作成または追加されたメンバーをソース・メンバー名として使用します。

### ソース・ファイル・メンバー名

ILE RPGソース・プログラムが入っているメンバーの名前を入力してください。

上

---

## ソース・ストリーム・ファイル (SRCSTMF)

コンパイルするILE RPGソース・コードが入っているストリーム・ファイルのパス名を指定します。

このパス名は絶対修飾または相対修飾のいずれかとすることができます。絶対パス名は/で始まり、相対パス名は/以外の文字で始まります。

絶対修飾の場合は、パス名は完全です。相対修飾の場合は、パス名はジョブの現行作業ディレクトリーをそのパス名に付加することにより完全になります。

SRCMBRおよびSRCFILEパラメーターをSRCSTMFパラメーターと一緒に指定することはできません。

上

---

## 生成重大度レベル (GENLVL)

プログラム・オブジェクトの作成を制御します。プログラム・オブジェクトは、コンパイル中に見つかったすべてのエラーの重大度レベルが指定された生成重大度レベル以下である場合に作成されます。

値は0-20でなければなりません。重大度が20を超えるエラーの場合には、プログラム・オブジェクトは生成されません。

**10** コンパイル時エラーの重大度レベルが10以下であれば、プログラム・オブジェクトが生成されます。これが省略時の値です。

### 重大度レベル値

0 - 20の数値を入力してください。

---

## テキスト'記述' (TEXT)

プログラムおよびその機能を簡単に説明するテキストを入力することができます。プログラム情報が表示される時には、常にこのテキストが現れます。

### \*SRCMBRTXT

ソース・メンバーのテキストが使用されます。

### \*BLANK

テキストはありません。

**'記述'** ソース仕様の機能を簡単に記述するテキストを入力してください。テキストは最大50文字にすることができますが、アポストロフィで囲まなければなりません。アポストロフィは50文字のストリングの一部ではありません。プロンプト画面にテキストを入力する場合には、アポストロフィは必要ありません。

---

## 省略時の活動化グループ (DFTACTGRP)

作成されるプログラムを常に省略時の活動化グループで実行するかどうかを指定します。

**\*YES** このプログラムは、呼び出されると、常に省略時の活動化グループで実行されます。省略時の活動化グループとは、すべてのオリジナル・プログラム・モデル(OPM)プログラムが実行される活動化グループのことです。

DFTACTGRP(\*YES)を指定すると、ILE RPGプログラムはファイル共用、ファイル有効範囲設定、およびRCLRSCの面でOPMプログラムと同様に機能します。

プログラムがDFTACTGRP(\*YES)で作成された時には、ILE静的バインドは使用可能でありません。これは、このプログラムの作成時にBNDDIRまたはACTGRPパラメーターを使用できないことを意味します。さらに、ソース中の呼び出し命令はプロシージャでなくプログラムを呼び出すものでなければなりません。

DFTACTGRP(\*YES)は、プログラム単位でアプリケーションをILE RPGに移動する時に便利です。

**\*NO** プログラムは、ACTGRPパラメーターで指定された活動化グループに関連付けられます。\*NOを指定した時には、静的バインドが可能です。

ACTGRP(\*CALLER)を指定し、このプログラムが省略時の活動化グループで実行するプログラムによって呼び出される場合には、このプログラムは、ファイル共用、ファイル有効範囲設定、およびRCLRSCの面でILEセマンティクスに従って機能します。

DFTACTGRP(\*NO)は、名前を指定された活動化グループでの実行やサービス・プログラムへのバインドなどのILE概念の利点を利用する時に便利です。

---

## 活動化グループ (ACTGRP)

呼び出された時にこのプログラムが関連付けられる活動化グループを指定します。

**QILE** このプログラムは、呼び出されると、名前を指定された活動化グループQILE中に活動化されます。これが省略時の値です。

**\*NEW** このプログラムは、呼び出されると、新しい活動化グループ中に活動化されます。

**\*CALLER**

このプログラムは、呼び出されると、呼び出し元の活動化グループ中に活動化されます。

**活動化グループ名**

このプログラムが呼び出された時に使用される活動化グループの名前を指定してください。

上

---

## BINDINGディレクトリー (BNDDIR)

記号分析解決で使用するバインディング・ディレクトリーのリストを指定します。

**\*NONE**

バインディング・ディレクトリーは指定されません。

**バインディング・ディレクトリー名**

記号分析解決で使用するバインディング・ディレクトリーの名前を指定してください。ディレクトリー名は、次の1つのライブラリー値で修飾することができます。

**\*LIBL** バインディング・ディレクトリーが保管されているライブラリーを見つけるために、システムはライブラリー・リストを検索します。これが省略時の値です。

**\*CURLIB**

ジョブの現行ライブラリーが検索されます。ジョブの現行ライブラリーとしてライブラリーが指定されていない場合には、ライブラリーQGPLが使用されます。

**\*USRLIBL**

ジョブのライブラリー・リストのユーザー部分にあるライブラリーだけが検索されます。

**ライブラリー名**

検索するライブラリーの名前を指定してください。

上

---

## コンパイラー・オプション (OPTION)

ソース・メンバーのコンパイル時に使用するオプションを指定します。一部またはすべてのオプションをどのような順序でも指定することができます。各オプションは1つまたは複数のブランク・スペースで区切ってください。オプションが複数回指定された場合には、最後のものが使用されます。

**\*XREF**

(適切な場合には) ソース・メンバーの相互参照表を作成します。

**\*NOXREF**

相互参照表は作成されません。

**\*GEN** コンパイラーによって戻された最高の重大度レベルがGENLVLオプションに指定された重大度を超えない場合にプログラム・オブジェクトを作成します。

**\*NOGEN**

プログラム・オブジェクトを作成しません。

#### **\*NOSECLVL**

第1レベル・メッセージ・テキストの次の行に第2レベル・メッセージ・テキストを印刷しません。

#### **\*SECLVL**

メッセージ要約セクションの第1レベル・メッセージ・テキストの次の行に第2レベル・メッセージ・テキストを印刷します。

#### **\*SHOWCPY**

/COPYコンパイラー・ディレクティブによって含まれるメンバーのソース・レコードを表示します。

#### **\*NOSHOWCPY**

/COPYコンパイラー・ディレクティブによって含まれるメンバーのソース・レコードを表示しません。

#### **\*EXPDDS**

外部記述ファイルの拡張をリストに表示し、キー・フィールド情報を表示します。

#### **\*NOEXPDDS**

外部記述ファイルの拡張をリストに表示せず、キー・フィールド情報を表示しません。

**\*EXT** コンパイル時に参照された外部プロシージャおよびフィールドのリストをリストに表示します。

#### **\*NOEXT**

コンパイル時に参照された外部プロシージャおよびフィールドのリストをリストに表示しません。

#### **\*NOSHOWSKP**

リストのソース部分の中の無視されたステートメントは表示されません。コンパイラーは、/IF、/ELSEIFまたは/ELSEディレクティブの結果としてのステートメントを無視します。

#### **\*SHOWSKP**

リストのソース部分の中のすべてのステートメントを、コンパイラーがそれらをスキップしたかどうかにかかわらず表示します。

#### **\*NOSRCSTMT**

リスト中の行番号は、順次に割り当てられます。これらの番号は、ステートメント番号を使用してデバッグするときに使用されます。行番号は、リストの左端の桁に示されます。リストの右端の2桁にはソースIDとSEU順序番号が表示されます。

#### **\*SRCSTMT**

デバッグのためのステートメント番号は、次のようにSEU順序番号およびソースIDを使用して生成されます。

$$\text{ステートメント番号} = \text{ソースID} * 1000000 + \text{ソースSEU順序番号}$$

SEU順序番号は、リストの左端の桁に示されます。ステートメント番号は、リストの右端の桁に示されます。これらの番号は、ステートメント番号を使用してデバッグするときに使用されます。

注: OPTION(\*SRCSTMT)が指定されているときには、ソース・ファイル中のすべての順序番号が有効な数値でなければなりません。同じソース・ファイルに重複した順序番号がある場合には、デバッガーの振る舞いは予測できず、診断メッセージまたは相互参照項目のステートメント番号が無意味となることがあります。

#### **\*DEBUGIO**

すべての入出力仕様に停止点を生成します。



#### **\*NODEBUGIO**

入出力仕様に停止点を生成しません。

#### **\*UNREF**

参照されないデータ項目が、コンパイル済みのモジュールに組み込まれます。

#### **\*NOUNREF**

参照されないデータ項目は、コンパイル済みのモジュールに組み込まれません。これにより、使用されるストレージ容量が減り、より大きなプログラムをコンパイルできるようになります。

\*NOUNREFオプションが選択されている場合、デバッグ時に、参照されないデータ項目を確認したり、参照されないデータ項目に割り当てを行ったりすることはできません。なお、参照されないデータ項目は、OPTION(\*XREF)を指定することで作成される相互参照リストには表示されます。

#### **\*NOEVENTF**

クライアント・ツールで使用するためのイベント・ファイルを作成しません。クライアント・ツールでは、統合エラー・フィードバックを行うために、このファイルが使用されます。通常、イベント・ファイルは、クライアント・ツール内からモジュールまたはプログラムを生成するときに作成されます。

#### **\*EVENTF**

クライアント・ツールで使用するためのイベント・ファイルを作成します。イベント・ファイルは、作成されたモジュールまたはプログラム・オブジェクトが保管されるライブラリー中のファイルEVFEVENTのメンバーとして作成されます。ファイルEVFEVENT が存在しない場合には、自動的に作成されます。イベント・ファイル・メンバー名は、作成されるオブジェクトの名前と同じです。

クライアント・ツールでは、統合エラー・フィードバックを行うために、このファイルが使用されます。通常、イベント・ファイルは、クライアント・ツール内からモジュールまたはプログラムを生成するときに作成されます。

上

---

## **デバッグ用ビュー (DBGVIEW)**

コンパイル済みプログラム・オブジェクトに使用可能なデバッグのレベルおよびソース・レベルのデバッグに使用可能なソース・ビューを指定します。

#### **\*STMT**

コンパイラー・リストの行番号またはステートメント番号を使用して、プログラム・オブジェクトをデバッグできるようにします。OPTION(\*NOSRCSTMT)が指定されているときには、行番号がコンパイラー・リストのソース・セクションの左端の桁に示されます。OPTION(\*SRCSTMT)が指定されているときには、ステートメント番号がコンパイラー・リストのソース・セクションの右端の桁に示されます。

#### **\*SOURCE**

コンパイル済みプログラム・オブジェクトのデバッグのためのソース・ビューを生成します。ルートソース・メンバーがDDMファイルである場合には、このビューは使用可能ではありません。また、コンパイル後、プログラムのデバッグを試みる前にソース・メンバーに変更が加えられた場合にも、これらのソース・メンバーのビューが使用できないことがあります。

**\*LIST** コンパイル済みプログラム・オブジェクトのデバッグのためのリスト・ビューを生成します。リスト・ビューに含まれる情報は、OPTIONパラメーターに\*SHOWCPY, \*EXPDDS,および\*SRCSTMTを指定していたかどうかによります。



注: リスト・ビューには、字下げオプションを使用して要求した字下げは示されません。

#### **\*COPY**

コンパイル済みプログラム・オブジェクトのデバッグのためのソース・ビューおよびコピー・ビューを生成します。このオプションのソース・ビューは、\*SOURCEオプションの場合に生成されたソース・ビューと同じです。コピー・ビューは、すべての/COPYソース・メンバーが含まれるデバッグ・ビューです。ルートのソース・メンバーがDDMファイルである場合には、これらのビューは使用可能ではありません。また、コンパイル後、プログラムのデバッグを試みる前にソース・メンバーに変更が加えられた場合にも、これらのソース・メンバーのビューが使用できないことがあります。

**\*ALL** コンパイル済みプログラム・オブジェクトのデバッグのためのリスト・ビュー、ソース・ビュー、およびコピー・ビューを生成します。リスト・ビューに含まれる情報は、OPTIONパラメーターに\*SHOWCPY, \*EXPDDS,および\*SRCSTMTを指定していたかどうかによります。

#### **\*NONE**

コンパイル済みプログラム・オブジェクトのデバッグのためのすべてのデバッグ・オプションを使用できないようにします。

上

---

## **出力 (OUTPUT)**

コンパイル・リストが生成されるかどうかを指定します。

#### **\*PRINT**

ILE RPGプログラム・ソースおよびすべてのコンパイル時メッセージからなるコンパイル・リストを作成します。リストに含まれる情報は、OPTIONパラメーターに\*XREF, \*SECLVL, \*SHOWCPY, \*EXPDDS, \*EXT, \*SHOWSKP,および\*SRCSTMTを指定しているかどうかによって異なります。

#### **\*NONE**

コンパイル・リストを生成しません。

上

---

## **最適化レベル (OPTIMIZE)**

モジュールの最適化のレベルがあればそれを指定します。

#### **\*NONE**

生成されたコードは最適化されません。これは、変換時間の面で一番早いものです。デバッグ・モードになっている時には、変数を表示して修正することができます。

#### **\*BASIC**

生成されたコードに対してある種の最適化が実行されます。これにより、デバッグ・モードになっている時に、ユーザー変数は表示できますが、修正することはできません。

#### **\*FULL**

最も効果的なコードを生成する最適化。変換時間は最も遅くなります。提示されている値が現在値でない場合でも、ユーザー変数は変更できませんが、表示することはできます。

上

---

## ソース・リストの字下げ (INDENT)

読み易さを増すために、構造化された命令をソース・リストで字下げするかどうかを指定します。構造化された命令の文節をマークするために使用される文字も指定します。

注: ここで要求した字下げは、DBGVIEW(\*LIST)を指定した時に作成されるリスト・デバッグ・ビューで反映されません。

### \*NONE

ソース・リストで構造化された命令は字下げされません。

**文字値** 構造化された命令の文節に対してソース・リストが字下げされます。ステートメントおよび文節の位置合わせは、選択した文字を使用してマークされます。最大2文字の長さの任意の文字ストリングを選択することができます。文字ストリング中にブランクを使用したい場合には、そのストリングを単一引用符で囲まなければなりません。

注: プログラムにエラーがある場合には、字下げは期待通りに現れないことがあります。

上

---

## タイプ変換オプション (CVTOPT)

ILE RPGコンパイラによる外部記述データベース・ファイルから検索された日付、時刻、タイム・スタンプ、図形データ・タイプ、および可変長データ・タイプの操作方法を指定します。

### \*NONE

可変長データベースのデータ・タイプを無視し、RPG固有の日付、時刻、タイム・スタンプ、および図形データ・タイプを使用します。

### **\*DATETIME**

日付、時刻、およびタイム・スタンプのデータベース・データ・タイプが固定長文字フィールドとして宣言されることを指定します。

### **\*GRAPHIC**

2バイト文字セット(DBCS)図形データ・タイプが固定長文字フィールドとして宣言されることを指定します。

### **\*VARCHAR**

可変長文字データ・タイプが固定長文字フィールドとして宣言されることを指定します。

### **\*VARGRAPHIC**

可変長2バイト文字セット(DBCS)図形データ・タイプが固定長文字フィールドとして宣言されることを指定します。

上

---

## ソート順序 (SRTSEQ)

ILE RPGソース・プログラムで使用されるソート順序テーブルを指定します。

**\*HEX** ソート順序テーブルは使用されません。

**\*JOB** \*PGMが作成される時にジョブのSRTSEQ値を使用します。

### **\*JOBRUN**

\*PGMが実行される時にジョブのSRTSEQ値を使用します。

### **\*LANGIDUNQ**

固有の重みテーブルを使用します。この特殊値は、正しいソート順序テーブルを決定するためにLANGIDパラメーターと一緒に使用されます。

### **\*LANGIDSHR**

共用重みテーブルを使用します。この特殊値は、正しいソート順序テーブルを決定するためにLANGIDパラメーターと一緒に使用されます。

### **ソート・テーブル名**

プログラムで使用されるソート順序テーブルの修飾名を入力してください。

**\*LIBL** ソート順序テーブルが保管されているライブラリーを見つけるために、システムはライブラリー・リストを検索します。

### **\*CURLIB**

ソート順序テーブルを見つけるために、現行ライブラリーが使用されます。現行ライブラリーが指定されていない場合には、QGPLが使用されます。

### **ライブラリー名**

ソート順序テーブルが保管されているライブラリーの名前を入力してください。

上

---

## **言語識別コード (LANGID)**

ソート順序が\*LANGIDUNQおよび\*LANGIDSHRである時に使用される言語識別コードを指定します。LANGIDパラメーターは、ソート順序テーブルを選択するためにSRTSEQ パラメーターと一緒に使用されます。

### **\*JOBRUN**

RPGプログラムが実行される時にジョブと関連したLANGID値を使用します。

**\*JOB** RPGプログラムが作成される時にジョブと関連したLANGID値を使用します。

### **言語識別コード**

指定された言語識別コードを使用します。(たとえば、フランス語の場合にはFRA で、ドイツ語の場合にはDEU。)

上

---

## **プログラムの置き換え (REPLACE)**

指定された（または暗黙に指定された）ライブラリーに同じ名前のプログラムがすでに存在している時に、新しいプログラムが作成されるかどうかを指定します。CRTBNDRPGコマンドの処理の時に作成された中間モジュールは、REPLACE指定に従ったものではなく、QTEMPライブラリーに対してはREPLACE(\*NO)が暗示されます。CRTBNDRPGコマンドが処理を完了すると、中間モジュールは削除されます。

**\*YES** 指定されたライブラリーに新しいプログラムが作成されます。指定されたライブラリーの同じ名前の既存のプログラムはライブラリーQRPLOBJに移動されます。

**\*NO** 指定されたライブラリーに同じ名前のプログラムがすでに存在している場合には、新しいプログラムは作成されません。既存のプログラムは置き換えられず、メッセージが表示され、コンパイルは停止します。

上

---

## ユーザー・プロファイル (USRPRF)

作成されたプログラム・オブジェクトを実行するユーザー・プロファイルを指定します。プログラムを実行し、プログラムで使用できるオブジェクト（各プログラムに対してプログラムがもつ権限を含む）を制御するために、プログラム所有者またはプログラム・ユーザーのプロファイルが使用されます。プログラムがすでに存在している場合には、このパラメーターは更新されません。USRPRFの値を変更するためには、プログラムを削除し、正しい値を使用してコンパイルし直してください（あるいは構成する\*MODULEオブジェクトが存在している場合には、CRTPGMコマンドを呼び出すように選択することができます）。

### **\*USER**

プログラムはプログラムの使用者のユーザー・プロファイルのもとで実行されます。

### **\*OWNER**

プログラムは、プログラムの使用者と所有者の両方のユーザー・プロファイルのもとで実行されます。プログラムの実行中にオブジェクトを見つけ、それにアクセスするために、両方のユーザー・プロファイルに収集可能なオブジェクト権限のセットが使用されます。プログラムの実行中に作成されたオブジェクトは、すべてプログラムのユーザーによって所有されます。

上

---

## 権限 (AUT)

このオブジェクトに対する特定権限をもっていないユーザー、権限リスト上にないユーザー、および所属するユーザー・グループがこのオブジェクトに対する特定権限をもっていないユーザーに与えられる権限を指定します。プログラムの作成後に、CLコマンドのオブジェクト権限認可(GRTOBJAUT)またはオブジェクト権限取り消し(RVKOBJAUT)を使用してすべてまたは指定したユーザーの権限を変更することができます。これらのコマンドの詳細については、i5/OS Information Center (<http://www.ibm.com/systems/i/infocenter/>)の「プログラミング」カテゴリーに含まれる制御言語(CL) トピック・コレクションを参照してください。

### **\*LIBCRTAUT**

オブジェクトの共通権限は、ターゲット・ライブラリー（オブジェクトが入っているライブラリー）のCRTAUTキーワードから取られます。値は、オブジェクトの作成時に決定されます。作成後にライブラリーのCRTAUT値が変わった場合には、新しい値は既存のオブジェクトに反映されません。

**\*ALL** 所有者に限定されているか、あるいは権限リスト管理権限によって制御されるものを除いてプログラム・オブジェクトに対するすべての命令の権限。ユーザーはプログラム・オブジェクトの存在を制御し、オブジェクトに対するこの機密保護を指定し、オブジェクトを変更し、オブジェクトに対する基本機能を実行することができますが、所有権を移すことはできません。

### **\*CHANGE**

所有者に限定されているか、あるいはオブジェクト権とオブジェクト管理権によって制御されているものを除いて、プログラム・オブジェクトに対するすべてのデータ権限およびすべての命令を実行する権限を提供します。ユーザーはオブジェクトを変更し、オブジェクトに対して基本機能を実行することができます。

**\*USE** オブジェクト操作権および読み取り権限、すなわちプログラム・オブジェクトに対する基本操作の権限を提供します。ユーザーはオブジェクトを変更することができません。

注: プログラムの変数の定様式ダンプを取るには、ユーザーにプログラムの\*USE権限が必要です。変数をダンプするには、プログラムに、識別可能な情報も必要です。

一部のユーザーに変数をダンプさせたくない場合には、そのユーザーにプログラムに対する\*OBJOPR権限と、\*EXECUTE権限だけを割り当ててください。これにより、そのユーザーはプログラムを呼び出すことはできますが、その変数をダンプすることはできません。作成済みプログラムの権限を変更するには、EDTOBJAUT, GRTOBJAUT,またはRVKOBJAUTを使用してください。

どのユーザーにも変数をダンプさせたくない場合には、プログラム変更(CHGPGM)を使用して、プログラムの識別可能な情報を除去してください。

#### **\*EXCLUDE**

ユーザーは、オブジェクトをアクセスできません。

#### **権限リスト名**

ユーザーおよびプログラムが追加される権限の権限リストの名前を入力してください。プログラム・オブジェクトはこの権限リストによって保護され、プログラム・オブジェクトの共通権限は\*AUTLにセットされます。CRTBNDRPGコマンドを出す時には、この権限リストがシステム上に存在していなければなりません。

注: システムの機密保護要件を反映させるためには、AUTパラメーターを使用してください。使用可能な機密保護機能は、System i機密保護解説書(SD88-5027)で詳しく説明されています。

上

---

## **数字の切り捨て (TRUNCNBR)**

プログラムの実行中に数値オーバーフローが起こった場合に、結果のフィールドに切り捨て値を入れるか、それともエラーを生成するかを指定します。

注: TRUNCNBRオプションは式の中で行なわれる演算には適用されません。(式は拡張演算項目2フィールドにあります。) これらの演算でオーバーフローが起こった場合には、常にエラーが発生します。

**\*YES** 数値オーバーフローを無視して、結果のフィールドに切り捨て値を入れます。

**\*NO** 数値オーバーフローが検出された時に、実行時エラーが生成されます。

上

---

## **数値の修正 (FIXNBR)**

正しくない10進数データをコンパイラーによって訂正するかどうかを指定します。

#### **\*NONE**

正しくない10進数データが使用された場合に、実行時に10進数エラーとなることを指示します。

#### **\*ZONED**

正しくないゾーン10進数データは、バック・データへの変換時にコンパイラーによって訂正されます。数値フィールドのブランクはゼロとして扱われます。各桁は妥当性検査されます。桁が有効でない場合には、その桁はゼロで置き換えられます。符号が有効でない場合には、その符号は16進数



'F'の正符号に強制的に変更されます。符号が有効である場合には、必要に応じ16進数'F'の正符号または16進数'D'の負符号に変更されます。結果のパック・データが正しくない場合には、そのデータは訂正されません。

#### **\*INPUTPACKED**

正しくないパック10進数データが入力仕様の処理中に出てきた場合に、内部変数がゼロに設定されることを指示します。

上

---

## **ターゲット・リリース (TGTRLS)**

作成されるオブジェクトを使用するオペレーティング・システムのリリースを指定します。**\*CURRENT**および**\*PRV**値の場合の例では、ターゲット・リリース 値を指定する時には、形式VXRXXMXを使用してリリースを指定します。VX はバージョン、RXはリリース、MXはモディフィケーション・レベルです。たとえば、V2R3M0は、バージョン2、リリース3、モディフィケーション・レベル0です。

このパラメーターに対する有効な値はリリースごとに変わります。指定できる値は次の通りです。

#### **\*CURRENT**

オブジェクトは、現在ユーザー・システムで実行中のオペレーティング・システムのリリースで使用されます。たとえば、システムでV2R3M5を実行中の場合には、**\*CURRENT**はV2R3M5が導入されているシステムでオブジェクトを使用しようとしていることを意味します。また、オペレーティング・システムの後続のリリースが導入されたシステム上で、このオブジェクトを使用することもできます。

注: システムでV2R3M5が実行中で、オブジェクトをV2R3M0が導入されたシステムで使用しようとする場合には、TGTRLS(**\*CURRENT**)ではなくTGTRLS(V2R3M0)を指定してください。

**\*PRV** オブジェクトは、オペレーティング・システムの前のリリース、モディフィケーション・レベル0で使用されます。たとえばシステムでV2R3M5を実行中の場合には、**\*PRV**はV2R2M0が導入されているシステムでオブジェクトを使用しようとしていることを意味します。また、オペレーティング・システムの後続のリリースが導入されたシステム上で、このオブジェクトを使用することもできます。

#### **ターゲット・リリース**

リリースをVXRXXMXの形式で指定してください。オブジェクトは、指定したリリースのシステムまたはオペレーティング・システムの後続のリリースが導入されたシステムで 사용할 ことが できます。

有効な値は、現在のバージョン、リリース、およびモディフィケーション・レベルによって異なり、新しいリリースごとに変わります。このコマンドでサポートされている最も古いリリース・レベルよりも前の ターゲット・リリース を指定した場合には、エラー・メッセージが出され、サポートされる最も古いリリースが表示されます。

注: コマンドの現在のバージョンは、コマンドの前のリリースで使用可能でないオプションをサポートすることがあります。前のリリースで 使用されるオブジェクトを作成するためにコマンドが 使用される場合には、コマンドはそのリリースに該当するコンパイラーで処理され、サポートされないオプションはどれも認識されません。コンパイラーは、処理に使用可能でないオプションについて警告を出すとは限りません。

上

---

## ヌル値可能 (ALWNULL)

ILE RPGプログラムが、外部記述データベース・ファイルから、ヌル値可能フィールドをもつレコードをどのように使用できるかを指定します。

**\*NO** ILE RPGプログラムが外部記述ファイルからのヌル値フィールドをもつレコードを処理しないことを指定します。ヌル値が入っているレコードを検索しようとした場合には、レコード中のデータはILE RPGプログラムに対してアクセス不能となり、データ・マッピング・エラーが起こります。

### **\*INPUTONLY**

ILE RPGプログラムが、外部記述入力専用データベース・ファイルから、ヌル値の入ったヌル値可能フィールドをもつレコードを正常に読み取ることができることを指定します。ヌル値の入ったレコードを検索する時には、データ・マッピング・エラーは起こらず、データベースの省略時の値がヌル値の入った任意のフィールドに入れられます。プログラムは以下を実行することはできません。

- ヌル値可能キー・フィールドの使用
- ヌル値可能フィールドの入ったレコードの作成または更新
- プログラムの実行時に、ヌル値可能フィールドが実際にヌルであるかどうかを判別すること
- ヌル値可能フィールドをヌルに設定すること

### **\*USRCTL**

ILE RPGプログラムが、外部記述データベース・ファイルから、ヌル値の入ったレコードを読み取り、書き出し、および更新できることを指定します。ヌル・キーのあるレコードはキー順操作を使用して検索することができます。プログラムは、ヌル値可能フィールドが実際にヌルであるかどうかを判別することができ、出力または更新用に、ヌル値可能フィールドをヌルに設定することができます。プログラマーは、ヌル値の入ったフィールドがプログラム内で正しく使用されていることを確認する責任があります。

**\*YES** \*INPUTONLYと同じ。

上

---

## 条件名の定義 (DEFINE)

コンパイルの開始前に定義される条件名を指定します。パラメーターDEFINE(条件名)を使用することは、ソース・ファイルの最初の行に/DEFINE条件名ディレクティブをエンコードすることと同じです。

### **\*NONE**

条件名は定義されません。これが省略時の値です。

**条件名** 最大32個までの条件名を指定することができます。各名前の長さは50桁までとすることができます。条件名はコンパイルの開始時に定義されるものと見なされます。

上

---

## パフォーマンス収集使用可能化 (ENBPFRCOL)

パフォーマンス収集を使用可能にするかどうかを指定します。

**\*PEP** パフォーマンス統計は、プログラム入りロプロシーチャーの入り口および出口でのみ収集されます。これは、プログラムの実際のプログラム入りロプロシーチャーに対して適用され、プログラム内のモジュールのメイン・プロシーチャーには適用されません。これが省略時の値です。

#### **\*ENTRYEXIT**

パフォーマンス統計はプログラムのすべてのプロシーチャーの入り口および出口に収集されます。

#### **\*FULL**

パフォーマンス統計はすべてのプロシーチャーの入り口および出口に収集されます。また、統計は、外部プロシーチャーに対する各呼び出しの前後に収集されます。

上

---

## **プロファイル・データ (PRFDTA)**

プログラム・プロファイル・データ属性をプログラムに指定します。プログラム・プロファイルは、統計データ（プロファイル・データ）に基づいてプロシーチャーおよびプロシーチャー内のコードを再順序づけるために使用される拡張最適化手法です。

#### **\*NOCOL**

このプログラムはプロファイル・データを収集できません。これが省略時の値です。

**\*COL** プログラムはプロファイル・データを収集できます。**\*COL**は、モジュールの最適化レベルが**\*FULL**の時と**\*CURRENT**のターゲット・リリースでコンパイルしている時にだけ指定することができます。

注: BNDDIRパラメーターを使用して追加のモジュールおよびサービス・プログラムをバインドする場合に、これらの追加のオブジェクトは**\*COL**または**\*NOCOL**がプログラムに指定されている時には影響を受けません。モジュールのプログラム・プロファイル・データ属性は、モジュール作成時に設定されます。

上

---

## **ライセンス内部コード・オプション (LICOPT)**

1つまたは複数のライセンス内部コード・コンパイル時オプションを指定します。このパラメーターは、個別のコンパイル時オプションを選択できるようにするもので、選択したそれぞれのタイプのコンパイラー・オプションの潜在的な利点と欠点を理解した高度のプログラマーを対象にしています。

上

---

## **組み込みディレクトリー (INCDIR)**

コピー・ファイルを検索するためにコンパイラーにより使用される検索パスに追加する1つ以上のディレクトリーを指定します。ソース・プログラム中のコピー・ファイルを解決できない場合に、コンパイラーはここに指定されたディレクトリーを検索します。

#### **\*NONE**

ユーザー・ディレクトリーでコピー・ファイルは検索されません。省略時の値により、ソース・ディレクトリーは検索されます。

#### **'ディレクトリー'**

コピー・ファイルを検索するディレクトリーを最大32個まで指定してください。指定されたディレクトリーの他に、ソース・ディレクトリーからもコピー・ファイルが検索されます。

上



---

## Prgmインターフェース情報 (PGMINFO)

プログラム・インターフェース情報を生成するかどうか、およびどこに生成するのを制御するオプションを指定します。

**生成** プログラム・インターフェース情報を生成するかどうかを指定します。指定できる値は次の通りです。

**\*NO** このオプションは、プログラム・インターフェース情報を生成しない省略時の値を指定します。

**\*PCML**

PCML（プログラム呼び出しマークアップ言語）を生成することを指定します。生成されたPCMLは、JAVAメソッドが少ないJAVAコードでこのRPGプログラムを呼び出しやすくします。

**場所** 生成パラメーターが\*PCMLの場合、生成されたプログラム情報の場所を指定します。指定できる値は次の通りです。

**\*STMF**

プログラム情報をストリーム・ファイルに生成するように指定します。生成される情報が入るストリーム・ファイルの名前はINFOSTMFオプションに指定してください。

**\*MODULE**

CRTBNDRPGコマンドの最初のステップとして作成されたRPGモジュールにプログラム情報を保管するように指定します。

**\*ALL** プログラム情報をストリーム・ファイルに生成し、モジュールに保管するように指定します。生成される情報が入るストリーム・ファイルの名前はINFOSTMFオプションに指定してください。

上

---

## プログラム・インターフェース・ストリーム・ファイル (INFOSTMF)

PGMINFOオプションに指定された生成されたプログラム・インターフェース情報が入るストリーム・ファイルのパス名を指定します。

このパス名は絶対修飾または相対修飾のいずれかとすることができます。絶対パス名は/で始まり、相対パス名は/以外の文字で始まります。

絶対修飾の場合は、パス名は完全です。相対修飾の場合は、パス名はジョブの現行作業ディレクトリーをそのパス名に付加することにより完全になります。

このパラメーターを指定できるのは、PGMINFOパラメーターに\*NO以外の値が指定されている場合だけです。

上

---

## プリプロセッサ・オプション (PPGENOPT)

ソース・コードのコンパイル時に使用するプリプロセッサ生成オプションを指定します。

考えられるオプションは次の通りです。

### **\*NONE**

ソース・ファイルに対してコンパイラ全体を実行します。プリプロセッサ出力をファイルにコピーしません。

**\*DFT** 入力ソースに対してプリプロセッサを実行します。プリプロセッサ出力の生成のオプションとして、**\*RMVCOMMENT**、**\*EXPINCLUDE**および**\*NOSEQSRC**が使用されます。出力ソース・ファイルおよびメンバーを指定するには、**PPSRCFILE**および**PPSRCMBR**を使用し、プリプロセッサ出力を含めるストリーム・ファイルを指定するには、**PPSRCSTMF**を使用してください。

### **\*RMVCOMMENT**

プリプロセス中にコメント、ブランク行、およびほとんどのディレクティブを除去します。RPG仕様のみ、および仕様の正しい変換処理に必要なすべてのディレクティブは保存します。

### **\*NORMVCOMMENT**

プリプロセス中にコメント、ブランク行、およびリスト制御ディレクティブ（たとえば、**/EJECT**、**/TITLE**）を保存します。プリプロセス中にソース制御ディレクティブ（たとえば、**/COPY**、**/IF**）をコメントに変換します。

### **\*EXPINCLUDE**

生成された出力ファイルの**/INCLUDE**ディレクティブを展開します。

### **\*NOEXPINCLUDE**

**/INCLUDE**ディレクティブは生成された出力ファイルに変更されずに配置されます。

注: **/COPY**ディレクティブは常に展開されます。

### **\*SEQSRC**

**PPSRCFILE**を指定した場合は、生成された出力メンバーには順次に順序番号が付けられ、000001で始まり、000001ずつ増分されます。

### **\*NOSEQSRC**

**PPSRCFILE**を指定した場合は、生成された出力メンバーは、プリプロセッサが読み取った元のソースと同じ順序番号を持ちます。

上

---

## 出力ソース・ファイル (PPSRCFILE)

プリプロセッサ出力のソース・ファイル名およびライブラリーを指定します。

### ソース・ファイル名

プリプロセッサ出力のソース・ファイルの名前を指定します。

考えられるライブラリー値は次の通りです。

### **\*CURLIB**

プリプロセッサ出力は現行ライブラリー中に作成されます。ジョブに現行ライブラリーがない場合は、プリプロセッサ出力ファイルは**QGPL**ライブラリー中に作成されます。

### ライブラリー名

プリプロセッサ出力のライブラリーの名前を指定します。

---

## 出力ソース・メンバー (PPSRCMBR)

プリプロセッサ出力のソース・ファイル・メンバーの名前を指定します。

**\*PGM** PGMパラメーターに提供した名前は、プリプロセッサ出力メンバー名として使用されます。

### メンバー名

プリプロセッサ出力のメンバーの名前を指定します。

---

## 出力ストリーム・ファイル (PPSRCSTMF)

プリプロセッサ出力のストリーム・ファイルのパス名を指定します。

### \*SRCSTMF

SRCSTMFパラメーターに提供したパス名は、プリプロセッサ出力パス名として使用されます。  
ファイルは拡張子'.I'を持ちます。

### 'パス名'

プリプロセッサ出力ストリーム・ファイルのパス名を指定します。

このパス名は絶対修飾または相対修飾のいずれかとすることができます。絶対パス名は'/'で始まり、相対パス名は'/'以外の文字で始まります。

絶対修飾の場合は、パス名は完全です。相対修飾の場合は、パス名はジョブの現行作業ディレクトリーをそのパス名に付加することにより完全になります。

---

## 例

### 例1: ソース・プログラムのプログラム・オブジェクトへのコンパイル

```
CRTBNDRPG  PGM(MYLIB/XMPLE1)
           SRCFILE(MYLIB/QRPGSRC) SRCMBR(XMPLE1)
           OUTPUT(*PRINT) TEXT('MY RPG IV PROGRAM')
```

このコマンドはILE RPGのコンパイラーを呼び出して、XMPLE1の名前のプログラムを作成します。ソース・プログラムは、ライブラリーMYLIB中のソース・ファイルQRPGLESRCのメンバーXMPLE1中にありません。コンパイラー・リストが作成されます。

---

## エラー・メッセージ

### \*ESCAPE メッセージ

**RNS9310**

コンパイルは正常に実行されなかった。プログラム&1がライブラリー&2に作成されませんでした。

上

## COBOLモジュールの作成 (CRTCBLMOD)

実行可能場所: すべての環境 (\*ALL)

スレッド・セーフ: いいえ

パラメーター

例

エラー・メッセージ

CRTCBLMODコマンドはILE COBOLソース・プログラムをモジュール内にコンパイルします。このコマンドは、対話式、バッチ・モード、またはCLプログラムで使用することができます。

CRTCBLMODコマンドに指定するすべてのオブジェクト名は、サーバーの命名規則に従っていなければなりません。

CRTCBLMODコマンドのパラメーターについての説明は、次の通りです。

上

### パラメーター

キーワード	記述	選択項目	ノーツ
MODULE	モジュール	修飾オブジェクト名	オプションル, 定位置 1
	修飾子 1: モジュール	名前, *PGMID	
	修飾子 2: ライブラリー	名前, *CURLIB	
SRCFILE	ソース・ファイル	修飾オブジェクト名	オプションル, 定位置 2
	修飾子 1: ソース・ファイル	名前, QCBLLSRC	
	修飾子 2: ライブラリー	名前, *LIBL, *CURLIB	
SRCMBR	ソース・メンバー	名前, *MODULE	オプションル, 定位置 3
SRCSTMF	ソース・ストリーム・ファイル	パス名	オプションル
GENLVL	生成重大度レベル	0-30, 30	オプションル
TEXT	テキスト記述	文字値, *SRCMBRTXT, *BLANK	オプションル
OUTPUT	出力	*PRINT, *NONE	オプションル, 定位置 4

キーワード	記述	選択項目	ノート
<b>OPTION</b>	コンパイラ・オプション	値 (最大 50 回の繰り返し): *SOURCE, *NOSOURCE, *SRC, *NOSRC, *NOXREF, *XREF, *GEN, *NOGEN, *NOSEQUENCE, *SEQUENCE, *NOVBSUM, *VBSUM, *NONUMBER, *NUMBER, *LINENUMBER, *NOMAP, *MAP, *NOOPTIONS, *OPTIONS, *QUOTE, *APOST, *NOSECLVL, *SECLVL, *PRTCORR, *NOPRTCORR, *MONOPRC, *NOMONOPRC, *RANGE, *NORANGE, *NOUNREF, *UNREF, *NOSYNC, *SYNC, *NOCRTF, *CRTF, *NODUPKEYCHK, *DUPKEYCHK, *NOINZDLT, *INZDLT, *NOBLK, *BLK, *STDINZ, *NOSTDINZ, *STDINZHEX00, *NODDSFILLER, *DDSFILLER, *NOIMBEDERR, *IMBEDERR, *STDTRUNC, *NOSTDTRUNC, *NOCHGPOSSGN, *CHGPOSSGN, *NOEVENTF, *EVENTF, *MONOPIC, *NOMONOPIC, *NOCRTARKIDX, *CRTARKIDX	オプション, 定位置 5
<b>CVTOPT</b>	変換オプション	値 (最大 8 回の繰り返し): *NOVARCHAR, *VARCHAR, *NODATETIME, *DATETIME, *NOPICXGRAPHIC, *PICXGRAPHIC, *NOPICGGRAPHIC, *PICGGRAPHIC, *NOFLOAT, *FLOAT, *NODATE, *DATE, *NOTIME, *TIME, *NOTIMESTAMP, *TIMESTAMP, *NOCVTTODATE, *CVTTODATE, *NOPICNGRAPHIC, *PICNGRAPHIC	オプション
<b>MSGMT</b>	メッセージ限界	要素リスト	オプション
	要素 1: メッセージの数	0-9999, <u>*NOMAX</u>	
	要素 2: メッセージ限界重大度	0-30, <u>30</u>	
<b>DBGVIEW</b>	デバッグ・ビューのオプション	要素リスト	オプション
	要素 1: デバッグ・ビュー	<u>*STMT</u> , *SOURCE, *LIST, *ALL, *NONE	
	要素 2: リスト・ビューの圧縮	<u>*NOCOMPRESSDBG</u> , *COMPRESSDBG	
<b>OPTIMIZE</b>	最適化レベル	<u>*NONE</u> , *BASIC, *FULL	オプション

キーワード	記述	選択項目	ノート
FLAGSTD	FIPSフラグ付け	値 (最大 2 回の繰り返し): *NOFIPS, *MINIMUM, *INTERMEDIATE, *HIGH, *NOOBSOLETE, *OBSOLETE	オプション
EXTDSPOPT	拡張表示オプション	値 (最大 3 回の繰り返し): *DFRWRT, *NODFRWRT, *UNDSPCHR, *NOUNDSPCHR, *ACCUPDALL, *ACCUPDNE	オプション
FLAG	フラグ重大度	0-99, <u>0</u>	オプション
REPLACE	モジュールの置き換え	<u>*YES</u> , *NO	オプション
AUT	権限	名前, <u>*LIBCRTAUT</u> , *ALL, *CHANGE, *USE, *EXCLUDE	オプション
LINKLIT	リンク・リテラル	<u>*PGM</u> , *PRC	オプション
TGTRLS	ターゲット・リリース	単純名, <u>*CURRENT</u> , *PRV	オプション
SRTSEQ	ソート順序	単一値: <u>*HEX</u> , *JOB, *JOBRUN, *LANGIDUNQ, *LANGIDSHR その他の値: 修飾オブジェクト名	オプション
	修飾子 1: ソート順序	名前	
	修飾子 2: ライブラリー	名前, <u>*LIBL</u> , *CURLIB	
LANGID	言語ID	文字値, <u>*JOBRUN</u> , *JOB	オプション
ENBPFRCOL	パフォーマンス収集の活動化	単一値: <u>*PEP</u> その他の値: 要素リスト	オプション
	要素 1: 収集レベル	<u>*FULL</u> , *ENTRYEXIT	
PRFDTA	プロファイル・データ	<u>*NOCOL</u> , *COL	オプション
CCSID	コード化文字セットID	整数, <u>*JOBRUN</u> , *HEX, *JOB	オプション
NTLCCSID	国別CCSID	整数, <u>13488</u>	オプション
ARITHMETIC	演算モード	<u>*NOEXTEND</u> , *EXTEND31, *EXTEND63	オプション
NTLPADCHAR	PADDING CHARACTER	要素リスト	オプション
	要素 1: SINGLE BYTE TO NATIONAL	文字値, <u>*DEFAULT</u>	
	要素 2: DOUBLE BYTE TO NATIONAL	文字値, <u>*DEFAULT</u>	
	要素 3: NATIONAL TO NATIONAL	文字値, <u>*DEFAULT</u>	
LICOPT	LICENSED INTERNAL CODE OPTIONS	文字値	オプション
INCDIR	ディレクトリー組み込み	値 (最大 32 回の繰り返し): パス名, <u>*NONE</u>	オプション
PGMINFO	PGMインターフェース情報	要素リスト	オプション
	要素 1: 生成	<u>*NO</u> , *PCML	
	要素 2: 位置	<u>*STMF</u> , *MODULE, *ALL	
INFOTMF	プログラム・インターフェース・ストリーム・ファイル	パス名	オプション

上



---

## モジュール (MODULE)

作成中のモジュール・オブジェクトのモジュール名およびライブラリー名を指定します。モジュール名およびライブラリー名はサーバーの命名規則に従っていなければなりません。指定できる値は次の通りです。

### \*PGMID

モジュールの名前は、コンパイル単位の一番外側のILE COBOLソース・プログラムのPROGRAM-ID段落から引用されます。

### モジュール名

コンパイル済みILE COBOLモジュールを識別する名前を入力してください。このパラメーターにモジュール名を指定して、この名前を使用する順序で ソース・プログラムの順序（単一ソース・ファイル・メンバーの複数のコンパイル単位）をコンパイルすると、この順序の最初のモジュールがこの名前を使用し、他のモジュールは対応する一番外側のコンパイル単位のILE COBOLソースのPROGRAM-ID段落に指定された名前を使用します。

指定可能なライブラリー値は、次の通りです。

### \*CURLIB

作成されたモジュール・オブジェクトは現行ライブラリーの中に記憶されます。現行ライブラリーとしてライブラリーが割り当てられていない場合には、QGPLが使用されます。

### ライブラリー名

作成されたモジュール・オブジェクトを記憶するライブラリーの名前を入力してください。

上

---

## ソース・ファイル (SRCFILE)

コンパイルするILE COBOLソース・コードが入っているソース・ファイルおよびライブラリーの名前を指定します。このソース・ファイルのレコード長は92でなければなりません。指定できる値は次の通りです。

### QCBLLSRC

ソース・ファイルQCBLLSRCにコンパイルするCOBOLのソース・コードが入っていることを指定します。

### ソース・ファイル名

コンパイルするILE COBOLのソース・コードが入っているソース・ファイルの名前を入力します。

指定可能なライブラリー値は、次の通りです。

\*LIBL ソース・ファイルが入っているライブラリーを見つけるためにライブラリー・リストが探索されます。

### \*CURLIB

現行ライブラリーが使用されます。現行ライブラリーとしてライブラリーが割り当てられていない場合には、QGPLが使用されます。

### ライブラリー名

ソース・ファイルが入っているライブラリーの名前を入力してください。

上

---

## ソース・メンバー (SRCMBR)

コンパイルするILE COBOLソース・コードが入っているメンバーの名前を指定します。SRCFILEパラメーターで参照しているソース・ファイルがデータベース・ファイルである場合にだけ、このパラメーターを指定することができます。指定できる値は次の通りです。

### \*MODULE

MODULEパラメーターで指定されたモジュール名と同じ名前をもつソース・ファイル・メンバーが使用されます。

MODULEパラメーターにモジュール名を指定しない場合には、データベース・ソース・ファイルの最初のメンバーが使用されます。

### ソース・ファイル・メンバー名

ILE COBOLソース・コードが入っているメンバーの名前を入力してください。

上

---

## ソース・ストリーム・ファイル (SRCSTMF)

コンパイルするILE COBOLソース・コードが入っているストリーム・ファイルのパス名を指定します。

このパス名は絶対的または相対的に修飾することができます。絶対パス名は'/'で始まり、相対パス名は'/'以外の文字で始まります。

絶対修飾の場合には、パス名は完全です。相対修飾の場合には、パス名は、パス名にジョブの現行作業ディレクトリーを付加することによって完了します。

SRCMBRおよびSRCFILEパラメーターをSRCSTMFパラメーターと一緒に指定することはできません。

上

---

## 生成重大度レベル (GENLVL)

モジュール・オブジェクトを作成するかどうかを決定する重大度レベルを指定します。重大度レベルはコンパイル時に生成されるメッセージの重大度レベルと対応しています。このパラメーターは、ソース・ファイル・メンバーの各コンパイル単位に個別に適用されます。前のコンパイル単位が正常に実行されない場合でも、ソース・ファイル・メンバー中のその他のコンパイル単位はコンパイルされることになります。

指定できる値は次の通りです。

- 30**      30以上の重大度レベルのエラーが起こった場合には、モジュール・オブジェクトは作成されません。

### 重大度レベル

モジュール・オブジェクトを作成するかどうかを決定するために使用したい重大度レベルを、0-30の範囲の1桁または2桁の数字で指定してください。この重大度レベル以上の重大度レベルのエラーが起こった場合には、モジュール・オブジェクトは作成されません。

上

---

## テキスト記述 (TEXT)

モジュールおよびその機能を簡単に説明するテキストを入力することができます。

### \*SRCMBRTXT

モジュール・オブジェクトの記述には、ILE COBOLソース・コードが入っているデータベース・ファイル・メンバーを記述するのと同じテキストが使用されます。情報源が装置ファイルまたはインライン・ファイルからの場合には、\*SRCMBRTXTの指定は\*BLANKの指定と同じ効果となります。

### \*BLANK

テキストは指定されません。

### テキスト記述

モジュールおよびその機能を簡単に説明するテキストを入力してください。テキストは最大50桁の長さのSBCS文字にすることができ、単一引用符で囲む必要があります。単一引用符は50桁の文字ストリングの一部にはなりません。

上

---

## 出力 (OUTPUT)

コンパイル・リストを生成するかどうかを指定します。指定できる値は次の通りです。

### \*PRINT

コンパイル・リストが生成されます。メンバーがコンパイルされる場合には、出力ファイルはメンバーと同じ名前になります。ストリーム・ファイルがコンパイルされて、PGMパラメーターに\*PGMIDが指定されている場合には、出力ファイルはCOBOLPGM00という名前になります。そうでない場合には、プログラムと同じ名前になります。

### \*NONE

コンパイル・リストは生成されません。

上

---

## コンパイラ・オプション (OPTION)

ILE COBOLソース・コードのコンパイル時に使用するオプションを指定します。

ILE COBOLソース・プログラムのPROCESSステートメントで指定されたオプションによって、OPTIONパラメーターの対応するオプションが指定変更されます。

OPTIONパラメーターに指定できる値は、次の通りです。

### \*SOURCEまたは\*SRC

コンパイラは、ILE COBOLソース・プログラムとすべてのコンパイル時エラー・メッセージから構成されるソース・リストを作成します。

### \*NOSOURCEまたは\*NOSRC

コンパイラはリストのソース・パートを作成しません。ソース・リストが不要である場合には、コンパイル時間を短くすることができるので、このオプションを使用してください。

### \*NOXREF

コンパイラはILE COBOLソース・プログラムの相互参照表を作成しません。

**\*XREF**

コンパイラーはILE COBOLソース・プログラムの相互参照表を作成します。

**\*GEN** コンパイラーは、ILE COBOLソースのコンパイル後に、モジュール・オブジェクトを作成します。

**\*NOGEN**

コンパイラーは、ILE COBOLソース・プログラムのコンパイル後に、モジュール・オブジェクトを作成しません。エラー・メッセージまたはリストだけが必要な場合には、このオプションを指定することができます。

**\*NOSEQUENCE**

参照番号のシーケンス・エラーは検査されません。

**\*SEQUENCE**

参照番号のシーケンス・エラーが検査されます。\*LINENUMBERオプションを指定した場合には、シーケンス・エラーは起こりません。

**\*NOVBSUM**

動詞使用カウントは印刷されません。

**\*VBSUM**

動詞使用カウントが印刷されます。

**\*NONUMBER**

ソース・ファイルの順序番号が参照番号に使用されます。

**\*NUMBER**

ユーザー提供の順序番号（1-6桁目）が参照番号に使用されます。

**\*LINENUMBER**

コンパイラーによって作成された順序番号が参照番号に使用されます。このオプションはILE COBOLプログラムのソース・コードとCOPYステートメントで導入されたソース・コードを結合して、1つの連続番号順にします。FIPS（米国情報処理規格）フラグ付けまたはSAAフラグ付けを指定する場合には、このオプションを使用してください。

**\*NOMAP**

コンパイラーはデータ部マップをリストしません。

**\*MAP** コンパイラーはデータ部マップをリストします。

**\*NOOPTIONS**

このコンパイルに有効となっているオプションがリストされません。

**\*OPTIONS**

このコンパイルに有効となっているオプションがリストされます。

**\*QUOTE**

区切り文字引用符(")が非数値リテラル、16進リテラル、およびブール・リテラルに使用されることを指定します。また、このオプションは、表意定数QUOTEの値が引用符のEBCDIC値をもつことも指定します。

**\*APOST**

区切り文字アポストロフィ(')が非数値リテラル、16進リテラル、およびブール・リテラルに使用されることを指定します。また、このオプションは、表意定数QUOTEの値がアポストロフィのEBCDIC値をもつことも指定します。

#### **\*NOSECLVL**

このコンパイルで第2レベル・メッセージ・テキストはリストされません。

#### **\*SECLVL**

このコンパイルで、コンパイル・リストのメッセージ・セクションに、第1レベル・エラー・テキストと一緒に第2レベル・メッセージ・テキストがリストされます。

#### **\*PRTCORR**

CORRESPONDING句の使用の結果として基本項目が含まれることを示すコメント行がコンパイル・リストの中に挿入されます。

#### **\*NOPRTCORR**

CORRESPONDING句が使用された時に、コメント行はコンパイル・リストの中に挿入されません。

#### **\*MONOPRC**

PROGRAM-ID段落、CALL、CANCEL、またはSET ENTRYステートメント、およびEND PROGRAMヘッダーで見つかったプログラム名（リテラルまたは語句）は、すべて大文字（単一シフト）に変換され、プログラム名作成規則が強制的に適用されます。

#### **\*NOMONOPRC**

PROGRAM-ID段落、CALL、CANCEL、またはSET ENTRYステートメント、およびEND PROGRAMヘッダーで見つかったプログラム名（リテラルまたは語句）は、すべて大文字（非単一シフト）に変換されず、プログラム名作成規則は強制されません。このオプションによって、標準COBOLでは使用できない特殊文字を、CALL行き先に使用することができます。

#### **\*RANGE**

実行時に、添え字は正しい範囲内にあることを確認するために検査されますが、指標の範囲は検査されません。参照変更およびコンパイラ生成のサブストリング命令も検査されます。

形式が正しいこと、さらに正しい日付、時刻、またはタイム・スタンプが表示されていることを確認するために、日付-時刻項目の内容を検査します。

#### **\*NORANGE**

実行時に範囲は検査されません。

注: \*RANGEオプションによって添え字範囲を検査するコードが生成されます。たとえば、20 要素の配列の21番目の要素をアクセスしようとしていないことが確認されます。

\*NORANGEオプションは添え字の範囲を検査するコードを生成しません。結果として、

\*NORANGEオプションによって高速の実行コードが作成されます。

#### **\*NOUNREF**

参照されていないデータ項目はコンパイル済みモジュールの中に含まれません。これによって、使用記憶域の量が減少し、より大きなプログラムをコンパイルすることができます。\*NOUNREFオプションを選択している時には、デバッグ中に、参照されていないデータ項目を表示したり割り当てたりすることはできません。参照されていないデータ項目は、OPTION (\*XREF)を指定して生成された相互参照表にはまだ表示されます。

#### **\*UNREF**

参照されていないデータ項目は、コンパイル済みモジュールに含まれます。

#### **\*NOSYNC**

SYNCHRONIZED文節は構文検査のみが行なわれます。

#### **\*SYNC**

SYNCHRONIZED文節がコンパイラによってコンパイルされます。SYNCHRONIZED文節によって、データ項目の位置は、右端（最小有効文字）が自然の記憶域境界になるように位置合わせされ

ます。自然の記憶域境界は、記憶するデータの長さおよびタイプによって、記憶域の中で次に最も近い4バイト境界、8バイト境界、または16バイト境界となります。この位置合わせを達成するために同期された項目に隣接する特別の記憶域が予約されます。SYNCHRONIZEDと記述された各基本データ項目は、そのデータ記憶域割り当てに対応する自然の記憶域境界に位置合わせされます。

#### **\*NOCRTF**

OPEN命令の実行時に使用できないディスク・ファイルは、動的には作成されません。

#### **\*CRTF**

OPEN命令の実行時に使用できないディスク・ファイルが、動的に作成されます。

注: 動的に作成されるファイルの最大レコード長は32766です。\*CRTFオプションが指定された場合でも、索引付きファイルは動的に作成されません。

#### **\*NODUPKEYCHK**

INDEXEDファイルの重複した基本レコード・キーと代替レコード・キーを検査しません。

#### **\*DUPKEYCHK**

INDEXEDファイルの重複した基本レコード・キーと代替レコード・キーを検査します。

#### **\*NOINZDLT**

順次アクセスによる相対ファイルは、ファイルがOUTPUT用にオープンされた場合には、CLOSE命令時に削除済みレコードを初期設定しません。レコード境界はOPEN OUTPUT時に書き出されたレコード数によって決まります。次のOPEN命令によって、レコード境界までに限ってアクセスすることができます。

#### **\*INZDLT**

順次アクセスによる相対ファイルは、ファイルがOUTPUT用にオープンされた場合には、CLOSE命令時に削除済みレコードを初期設定します。ファイルの活動レコードは影響を受けません。レコード境界は、次のOPEN命令のファイル・サイズとして定義されます。

#### **\*NOBLK**

コンパイラーは、STARTステートメントによらないSEQUENTIALアクセス・ファイルのブロック化だけを許します。BLOCK CONTAINS文節は、指定された場合には、テープ・ファイルの場合を除いて無視されます。

**\*BLK** \*BLKが使用されて、BLOCK CONTAINS文節が指定された時には、コンパイラーは、STARTステートメントによるDYNAMICアクセス・ファイルおよびSEQUENTIALアクセス・ファイルのブロック化を許します。出力操作でオープンされたRELATIVEファイルのブロック化は許されません。BLOCK CONTAINS文節はブロック化するレコードの数を制御します。

\*BLKが使用され、BLOCK CONTAINS文節が指定されていない時には、コンパイラーはSTARTステートメントによらないSEQUENTIALアクセス・ファイルのブロック化だけを許します。オペレーティング・システムがブロック化するレコード数を決定します。

#### **\*STDINZ**

VALUE文節をもたないこれらの項目の場合に、コンパイラーはデータ項目をシステムの省略時の値に初期設定します。

#### **\*NOSTDINZ**

VALUE文節をもたないこれらの項目の場合に、コンパイラーはデータ項目をシステムの省略時の値に初期設定しません。

#### **\*STDINZHEX00**

VALUE文節をもたないこれらの項目の場合に、コンパイラーはデータ項目を16進数のゼロに初期設定します。



#### **\*NODDSFILLER**

COPY DDSステートメントによって突き合わせフィールドが見つからない場合には、フィールド記述は生成されません。

#### **\*DDSFILLER**

COPY DDSステートメントによって突き合わせフィールドが見つからない場合には、単一文字の FILLER フィールド記述 "07 FILLER PIC X" が常に作成されます。

#### **\*NOIMBEDERR**

エラー・メッセージはコンパイル・リストのソース・リスト・セクションに含まれません。エラー・メッセージは、コンパイル・リストのエラー・メッセージ・セクションにだけ表示されます。

#### **\*IMBEDERR**

第1レベル・エラー・メッセージは、コンパイル・リストのソース・リスト・セクションに含まれ、エラーが起こった行の直後に表示されます。エラー・メッセージは、コンパイル・リストのエラー・メッセージ・セクションにも表示されます。

#### **\*STDTRUNC**

このオプションはUSAGE BINARYデータにだけ適用されます。\*STDTRUNCを選択した時には、USAGE BINARYデータはBINARY受け取りフィールドのPICTURE文節の桁数まで切り捨てられます。

#### **\*NOSTDTRUNC**

このオプションはUSAGE BINARYデータにだけ適用されます。\*NOSTDTRUNCを選択した時には、BINARY受け取りフィールドはハーフ・ワード、フル・ワード、またはダブル・ワード境界までのみ切り捨てられます。また、BINARY送り出しフィールドもハーフ・ワード、フル・ワード、またはダブル・ワードとして処理されます。したがって、フィールドの全2進数の内容が有効です。また、DISPLAYステートメントはBINARY フィールドの内容全体が切り捨てなしで変換されます。

#### **\*NOCHGPOSSGN**

ゾーンおよびパック数字データの省略時の正符号として16進数のFが使用されます。16進数のFは、オペレーティング・システムのシステム省略時値です。

#### **\*CHGPOSSGN**

ゾーンおよびパック数字データの省略時の正符号として16進数のCが使用されます。これは、VALUE文節の結果の他にMOVE, ADD, SUBTRACT, MULTIPLY, DIVIDE, COMPUTE,およびINITIALIZEステートメントのすべての結果に適用されます。

#### **\*NOEVENTF**

クライアント・ツールで使用するイベント・ファイルを作成しません。クライアント・ツールは、統合エラー・フィードバックを提供するときにこのファイルを使用します。イベント・ファイルは通常、クライアント・ツール内からモジュールまたはプログラムを作成する時に作成されます。

#### **\*EVENTF**

クライアント・ツールで使用するイベント・ファイルを作成します。イベント・ファイルは、作成されるモジュールまたはプログラム・オブジェクトが記憶されるライブラリー中のファイルEVFEVENTのメンバーとして作成されます。ファイルEVFEVENTが存在していない場合には、自動的にこれが作成されます。イベント・ファイルのメンバー名は、作成されるオブジェクトの名前と同じです。

クライアント・ツールは、統合エラー・フィードバックを提供するときにこのファイルを使用します。イベント・ファイルは通常、クライアント・ツール内からモジュールまたはプログラムを作成する時に作成されます。



### **\*MONOPIC**

PICTURE文字ストリングは、すべて大文字（単一シフト）に変換されます。

### **\*NOMONOPIC**

PICTURE文字ストリングに使用される通貨記号は大文字・小文字が区別されます。すなわち、PICTURE記号A, B, E, G, N, P, S, V, X, Z, CR,およびDBの大文字に対応する小文字は、PICTURE文字ストリングの中のそれらの大文字表記と同じです。他の小文字はすべて対応するそれらの大文字表記と同じではありません。

### **\*NOCRTARKIDX**

永続索引が見つからない場合には、一時代替レコード・キー(ARK)索引は作成されません。

### **\*CRTARKIDX**

永続索引が見つからない場合に、一時代替レコード・キー(ARK)索引が作成されます。

上

---

## **変換オプション (CVTOPT)**

コンパイラーが、COPY DDSを介して外部記述ファイルからプログラムに渡された日付、時刻、およびタイム・スタンプ・フィールド・タイプ、DBCSフィールド・タイプ、可変長文字フィールド・タイプ、および浮動小数点フィールド・タイプを処理する方法を指定します。指定できる値は次の通りです。

### **\*NOVARCHAR**

可変長フィールドはFILLERフィールドとして宣言されます。

### **\*VARCHAR**

可変長フィールドは、グループ項目として宣言され、ILE COBOLソース・プログラムにアクセスすることができます。

### **\*NODATETIME**

日付、時刻、およびタイム・スタンプ・データ・タイプがFILLERフィールドとして宣言されます。

### **\*DATETIME**

日付、時刻、およびタイム・スタンプのDDSデータ・タイプは、DDS名に基づいて指定されたCOBOLデータ項目名です。CVTOPTパラメーター値\*DATE, \*TIME,または\*TIMESTAMPの1つが指定されない限り、COBOLデータ項目のカテゴリーは英数字です。この場合には、COBOLデータ項目のカテゴリーは、それぞれ日付、時刻、またはタイム・スタンプです。

### **\*NOPICXGRAPHIC**

DBCSグラフィック・データ・タイプがFILLERフィールドとして宣言されます。

### **\*PICXGRAPHIC**

固定長DBCSグラフィックデータ・タイプは固定長英数字フィールドとして宣言され、ILE COBOLソース・プログラムにアクセスすることができます。

\*VARCHARオプションも使用されている場合には、可変長DBCSグラフィック・データ・タイプは固定長グループ項目として宣言されて、ILE COBOLソース・プログラムにアクセスすることができます。

### **\*PICGGRAPHIC**

固定長DBCSグラフィック・データ・タイプは固定長Gタイプ・フィールドとして宣言され、ILE COBOLソース・プログラムにアクセスすることができます。

\*VARCHARオプションも使用されている場合には、可変長DBCSグラフィック・データ・タイプは固定長グループ項目（後にGタイプ・フィールドが続く数値フィールドから構成される）として宣言され、ILE COBOLソース・プログラムにアクセスすることができます。

#### **\*NOPICGRAPHIC**

DBCSグラフィック・データ・タイプがFILLERフィールドとして宣言されます。

#### **\*NOFLOAT**

浮動小数点データ・タイプは2進数のUSAGEが指定されたFILLERフィールドとして宣言されます。

#### **\*FLOAT**

浮動小数点データ・タイプが、そのDDS名およびCOMP-1（単精度）またはCOMP-2（倍精度）のUSAGEが指定されてプログラムに組み込まれます。これらのフィールドはILE COBOLソース・プログラムにアクセスできるようになります。

#### **\*NODATE**

DDS日付データ・タイプは、カテゴリー英数字COBOLデータ項目として宣言されます。たとえば、次のようになります。

```
06 FILLER PIC X(10).
```

COBOLデータ項目名が、\*NODATETIME/\*DATETIME CVTOPTパラメーターによって判別されます。

#### **\*DATE**

DDS日付データ・タイプは、カテゴリー日付COBOLデータ項目として宣言されます。たとえば、次のようになります。

```
06 FILLER FORMAT DATE '@Y-%M-%D'.
```

COBOLデータ項目名が、\*NODATETIME/\*DATETIME CVTOPTパラメーターによって判別されます。

#### **\*NOTIME**

DDS時刻データ・タイプは、カテゴリー英数字COBOLデータ項目として宣言されます。たとえば次のようになります。

```
06 FILLER PIC X(8).
```

COBOLデータ項目名が、\*NODATETIME/\*DATETIME CVTOPTパラメーターによって判別されます。

#### **\*TIME**

DDS時刻データ・タイプが、カテゴリー時刻COBOLデータ項目として宣言されます。たとえば、次のようになります。

```
06 FILLER FORMAT TIME '%H:%M:%S'.
```

COBOLデータ項目名が、\*NODATETIME/\*DATETIME CVTOPTパラメーターによって判別されます。

#### **\*NOTIMESTAMP**

DDSタイム・スタンプ・データ・タイプが、カテゴリー英数字COBOLデータ項目として宣言されます。たとえば、次のようになります。

```
06 FILLER PIC X(26).
```

COBOLデータ項目名が、\*NODATETIME/\*DATETIME CVTOPTパラメーターによって判別されます。

### **\*TIMESTAMP**

DDSタイム・スタンプ・データ・タイプが、カテゴリーのタイム・スタンプCOBOLデータ項目として宣言されます。たとえば、次のようになります。

06 FILLER FORMAT TIMESTAMP.

COBOLデータ項目名が、\*NODATETIME/\*DATETIME CVTOPTパラメーターによって判別されます。

### **\*NOCVTTODATE**

DATFMTキーワードを指定したDDSデータ・タイプ(DDS日付データ・タイプを除く)は、元のDDSタイプに基づいてILE COBOLで宣言されます。

### **\*CVTTODATE**

DATFMTキーワードを指定したDDSデータ・タイプ(DDS日付データ・タイプを除く)は、日付データ・タイプとしてILE COBOLで宣言されます。

### **\*NOPICNGRAPHIC**

DBCSグラフィック・データ・タイプがFILLERフィールドとして宣言されます。

### **\*PICNGRAPHIC**

固定長DBCSグラフィック・データ・タイプは固定長各国語データ・フィールドとして宣言され、ILE COBOLソース・プログラムにアクセスすることができます。

上

---

## **メッセージ限界 (MSGLMT)**

各コンパイル単位について、その数だけのエラーが発生するとコンパイルが停止する特定のエラー重大度レベルのメッセージの最大数を指定します。1つのコンパイル単位がこの最大数に達すると、ソース・メンバー全体のコンパイルが停止されます。

たとえば、メッセージの最大数に3を指定し、エラー重大度レベルに20を指定した場合には、20以上の重大度レベルのエラーが3つまたはそれ以上起こった場合に、コンパイルが停止します。指定のエラー重大度レベルと等しいかまたは超えているメッセージがない場合には、見つかったエラーの数に関係なくコンパイルは続行されます。

### **メッセージの数**

メッセージの最大数を指定してください。指定できる値は次の通りです。

#### **\*NOMAX**

見つかったエラーの数に関係なく正常完了までコンパイルは続行されます。

**最大数** コンパイルが停止する前に指定のエラー重大度レベルでまたはそれ以上で発生可能なメッセージの最大数を指定します。有効な範囲は0-9999です。

### **メッセージ限界重大度**

コンパイルを停止するかどうかを判別するのに使用されるエラー重大度レベルを指定します。指定できる値は次の通りです。

**30** 重大度レベル30以上のエラーの数が指定されたメッセージの最大数を超えている場合に、コンパイルが停止します。

#### **エラー重大度レベル**

コンパイルを停止するかどうかを判別するのに使用したいエラー重大度レベルを、0-30の1

桁または2桁の数字で入力してください。この重大度レベル以上のエラーの数が指定されたメッセージの最大数を超えている場合に、コンパイルが停止します。

上

---

## デバッグ・ビューのオプション (DBGVIEW)

コンパイルされたモジュールのデバッグにどのソース・プログラムまたは生成されたリストのビューが使用可能かを制御するオプション、およびデバッグ・リスト・ビューが圧縮されるかどうかを制御するオプションを指定します。

### デバッグ・ビュー

デバッグに使用可能になるビューを指定してください。指定できる値は次の通りです。

#### **\*STMT**

記号名およびステートメント番号を使用して、コンパイルされたモジュールをデバッグすることができます。

#### **\*SOURCE**

COPYステートメントを介して含まれたコピーされたメンバーの他に、1次ソース・メンバーには、コンパイル済みモジュールのデバッグ用に使用可能なソース・ビューがあります。これらのビューは、1次ソース・メンバーおよびコピーされたソース・メンバーがローカル・データベース・ソース・ファイルから参照される場合だけ使用可能です。コンパイルおよびデバッグ中にメンバーを変更または削除しないでください。

**\*LIST** COPYおよびREPLACEステートメントの処理後にソース・コードを表示するリストは、コンパイル済みモジュールのデバッグに使用することができます。このオプションはコンパイル済みモジュールのサイズが増えますが、コンパイル済みプログラムの実行時パフォーマンスには影響しません。

リスト・ビューは、対応するコンパイラ・オプションが要求されたときに、相互参照表、データ部のマップ、および動詞の使用カウントを含めます。たとえば、OPTION(\*XREF)が指定されると、相互参照表が含まれます。

1次ソース・メンバーおよびコピーされたソース・メンバーがどこにあっても、リスト・ビューを生成することができます。リスト・ビューはコンパイル後のソース・メンバーの変更または削除の影響を受けません。

**\*ALL** \*STMT, \*SOURCE,および\*LISTを組み合わせると指定することと等価です。

#### **\*NONE**

コンパイルされたモジュールをデバッグすることはできません。これはコンパイル済みプログラムのサイズを減少しますが、その実行時パフォーマンスに影響しません。このオプションを指定した時には、定様式ダンプをとることはできません。

### リスト・ビューの圧縮

デバッグ・ビューで\*LISTまたは\*ALLを指定した場合に、リスト・ビューを圧縮するかどうかを指定します。指定できる値は次の通りです。

#### **\*NOCOMPRESSDBG**

リスト・ビューは圧縮されません。

### **\*COMPRESSDBG**

デバッグ・ビューで\*LISTまたは\*ALLを指定した場合に、リスト・ビューは圧縮されます。

上

---

## **最適化レベル (OPTIMIZE)**

モジュールの最適化のレベルを指定します。指定できる値は次の通りです。

### **\*NONE**

コンパイル済みモジュールで最適化は実行されません。このオプションを使用した時には、コンパイル時間は最小化されます。このオプションによって、デバッグ中に変数を表示および変更することができます。

### **\*BASIC**

コンパイル済みモジュールで一部の最適化（ローカル・ブロック・レベルでのみ）が実行されます。このオプションによって、デバッグ中にユーザー変数を表示できますが、変更することはできません。

### **\*FULL**

コンパイル済みモジュールで完全な最適化（グローバル・レベルで）が実行されます。この最適化によって、コンパイル時間は増えますが、最も効率的なコードが生成されます。このオプションによって、デバッグ中にユーザー変数を表示できますが、変更することはできません。表示された変数の値は現行値でない場合があります。一部の変数は、表示されない場合があります。

注: 選択した最適化レベルに関係なく、全面的な最適化を可能にするすべての情報が生成されます。ユーザーは、ソース・プログラムを再コンパイルすることなく、CHGMOD コマンドを使用して、モジュール・オブジェクトの\*NONEから\*FULLまで最適化レベルを変更することができます。

上

---

## **FIPSフラグ付け (FLAGSTD)**

FIPSフラグ付けのオプションを指定します。（FIPSメッセージで使用された参照番号が固有であることを確認するためには、\*LINENUMBERオプションを選択してください。）指定できる値は次の通りです。

### **\*NOFIPS**

ILE COBOLソース・プログラムにはFIPSフラグ付きではありません。

### **\*MINIMUM**

最低サブセット以上のFIPSフラグ。

### **\*INTERMEDIATE**

中間サブセット以上のFIPSフラグ。

### **\*HIGH**

高サブセットのFIPSフラグ。

### **\*NOOBSOLETE**

使用しない言語要素にフラグが付けられません。

#### **\*OBSOLETE**

使用されなくなった言語要素にフラグが付けられます。

上

---

## **拡張表示オプション (EXTDSPOPT)**

ワークステーション入出力用に拡張ACCEPTおよび拡張DISPLAYステートメントを使用するためのオプションを指定します。指定できる値は次の通りです。

#### **\*DFRWRT**

拡張DISPLAYステートメントは、拡張ACCEPTステートメントが見つかるか、あるいはバッファーが満たされるまで、バッファーの中に保留されます。

バッファーの内容は、拡張ACCEPTステートメントが見つかるか、あるいはバッファーが満たされた時に表示装置に書き出されます。

#### **\*NODFRWRT**

各拡張DISPLAYステートメントはそれが出てきた時に実行されます。

#### **\*UNDSPCHR**

表示可能および表示不能文字は、拡張ACCEPTおよび拡張DISPLAYステートメントによって処理されます。

#### **\*NOUNDSPCHR**

表示可能文字だけが、拡張ACCEPTおよび拡張DISPLAYステートメントによって処理されます。

リモート3174および3274制御装置に接続された表示装置にこのオプションを使用しなければなりません。ローカル・ワークステーションにもこのオプションを使用することができます。このオプションを使用する場合には、データに表示可能文字だけが入っていないければなりません。データに16進数20より小さい値が入っている場合には、予期しない画面様式から重大エラーに至る結果を予測することはできません。

#### **\*ACCUPDALL**

UPDATE句の存在に関係なく、拡張ACCEPTステートメントですべてのタイプのデータが事前表示されます。

#### **\*ACCUPDNE**

UPDATE句が含まれていない拡張ACCEPTステートメントで数字編集されたデータだけが事前表示されます。

上

---

## **フラグ重大度 (FLAG)**

コンパイル・リストに表示するメッセージの最小重大度レベルを指定します。指定できる値は次の通りです。

**0** 全てのメッセージがコンパイル・リストに表示されます。

#### **重大度レベル**

コンパイル・リストに表示したいメッセージの最小重大度レベルを指定する1桁または2桁の数字を入力してください。指定されたこの値以上の重大度レベルをもつメッセージがコンパイル・リストに表示されます。



---

## モジュールの置き換え (REPLACE)

指定されたライブラリーまたは暗黙のライブラリーに同じ名前のモジュールがすでに存在している時に、新しいモジュールを作成するかどうかを指定します。指定できる値は次の通りです。

- \*YES** 新しいモジュールが作成され、指定されたライブラリーまたは暗黙のライブラリーで同じ名前のすべての既存のモジュールを置き換えます。指定されたライブラリーまたは暗黙のライブラリーで同じ名前の既存のモジュールはライブラリーQRPLIBに移動されます。
- \*NO** 指定されたライブラリーまたは暗黙のライブラリーに同じ名前のモジュールがすでに存在する場合には、新しいモジュールは作成されません。既存のモジュールは置き換えられず、メッセージが表示され、コンパイルは停止します。

上

---

## 権限 (AUT)

モジュール・オブジェクトに対する特定権限を持っていないユーザー、権限リスト上にないユーザー、またはグループがモジュール・オブジェクトに対する特定権限を持っていないユーザーに与える権限を指定します。モジュール・オブジェクトを作成した後で、GRTOBJAUT（オブジェクト権限認可）またはRVKOBJAUT（オブジェクト権限取り消し）コマンドを使用してすべてのユーザーまたは特定のユーザーの権限を変更することができます。

指定できる値は次の通りです。

### **\*LIBCRTAUT**

オブジェクトの共通権限は、ターゲット・ライブラリー（作成されたモジュール・オブジェクトが入っているライブラリー）のCRTAUTキーワードから引用されます。この値はモジュール・オブジェクトが作成される時に決定されます。モジュール・オブジェクトが作成された後でライブラリーのCRTAUT値が変更された場合には、新しい値は既存のすべてのオブジェクトに影響しません。

- \*ALL** 所有者に限定されているか、または権限リスト管理権限によって管理されている以外のモジュール・オブジェクトにすべての操作の権限を与えます。ユーザーはモジュール・オブジェクトの存在を制御し、それに対する機密保護を指定して、それに対して基本機能を実行することができますが、その所有権を転送することはできません。

### **\*CHANGE**

所有者に限定されているか、またはオブジェクト権限およびオブジェクト管理権限によって制御されているものを除き、モジュール・オブジェクトのすべて操作を行うための権限、およびすべてのデータ権限を提供します。ユーザーはオブジェクトを変更し、それに対して基本機能を実行することができます。

- \*USE** モジュール・オブジェクトに基本的な操作を実行するための権限であるオブジェクト操作権および読み取り権限を提供します。ユーザーはオブジェクトに対して基本的な操作を実行することができますが、オブジェクトを変更することはできません。

注: プログラムの変数の定様式ダンプを取るには、ユーザーにプログラムの\*USE権限が必要です。変数をダンプするには、プログラムに検討可能な情報も必要です。

一部のユーザーに変数のダンプをさせたくない場合には、そのユーザーにモジュールを含むプログラムまたはサービス・プログラムに対する\*OBJOPR権限と\*EXECUTE権限だけを割り当ててください



い。これにより、そのユーザーはプログラムを呼び出すことはできますが、その変数をダンプすることはできません。作成済みのプログラムまたはサービス・プログラムの権限を変更するには、EDTOBJAUT, GRTOBJAUT,またはRVKOBJAUTを使用してください。

いかなるユーザーにも変数のダンプをさせたくない場合には、モジュールでモジュール変更(CHGMOD)を使用するか、またはモジュールを含むプログラムでプログラム変更(CHGPGM)あるいはサービス・プログラム変更(CHGSRVPGM)を使用して、検討可能な情報を除去してください。

#### **\*EXCLUDE**

ユーザーはモジュール・オブジェクトをアクセスすることができません。

#### **権限リスト名**

ユーザーおよびモジュールを追加する権限の権限リストの名前を入力してください。モジュール・オブジェクトは、この権限リストによって保護され、モジュール・オブジェクトの共通権限は\*AUTLに設定されます。CRTCLBLMODコマンドが出された時に、この権限リストはシステム上に存在していなければなりません。権限リスト作成(CRTAUTL)コマンドを使用してユーザー専用の権限リストを作成してください。

上

---

## **リンク・リテラル (LINKLIT)**

外部CALL/CANCEL 'リテラル' 行き先およびSET ENTRY行き先の関係タイプを指定します。SPECIAL-NAMES段落に次の文を指定することによって、特定の外部CALL/CANCEL 'リテラル' 行き先およびSET ENTRY行き先リストに対するこのオプションを一時変更することができます。

LINKAGE TYPE IS プログラム内ファイル名 FOR 行き先リスト。

LINKLITに指定できる値は、次の通りです。

**\*PGM** CALL/CANCELまたはSET ENTRYの行き先はプログラム・オブジェクトです。

**\*PRC** CALL/CANCELまたはSET ENTRYの行き先はILEプロシージャです。

上

---

## **ターゲット・リリース (TGTRLS)**

作成されるモジュール・オブジェクトを使用する予定であるオペレーティング・システムのリリースを指定します。**\*CURRENT**および**\*PRV**値の説明で示した例および ターゲット・リリース 値を指定するときのリリースの指定方法はVXRXXMXの形式です。ここで、VXはバージョンで、RXはリリースで、MXはモディフィケーション・レベルです。たとえば、V2R3M0はバージョン2,リリース3,モディフィケーション・レベル0です。

このパラメーターに有効な値はリリースごとに変わります。指定できる値は次の通りです。

#### **\*CURRENT**

オブジェクトは、現在システムで実行されているオペレーティング・システムのリリースで使用されます。たとえば、システムでV2R3M5が実行されている場合には、\*CURRENTはV2R3M5が導入されているシステムでオブジェクトを使用する予定であることを意味します。また、このオブジェクトは、オペレーティング・システムの後続のリリースが導入されたシステムで使用することもできます。

注: システム上でV2R3M5が実行されている場合で、オブジェクトをV2R3M0が導入されているシステムで使いたい場合には、TGTRLS(\*CURRENT)ではなくTGTRLS(V2R3M0)を指定してください。

- \*PRV** オブジェクトは、モディフィケーション・レベルが0のオペレーティング・システムの前のリリースで使用されます。たとえば、ユーザーのシステムでV2R3M5が実行されている場合には、\*PRVはV2R2M0が導入されているシステムでオブジェクトを使用する予定であることを意味します。また、このオブジェクトは、オペレーティング・システムの後続のリリースが導入されたシステムで使用することもできます。

### ターゲット・リリース

リリースをVXRXXMXの形式で指定してください。オブジェクトは、指定されたリリースのシステムまたはオペレーティング・システムの後続のリリースが導入されたシステムで使用することができます。

有効な値は、現行バージョン、リリース、およびモディフィケーション・レベルによって異なり、新しいリリースごとに変わります。このコマンドでサポートされている最も古いリリース・レベルよりも前の ターゲット・リリース を指定した場合には、エラー・メッセージが出され、サポートされる最も古いリリースが表示されます。

注: コマンドの現行バージョンは、前のコマンドのリリースでは使用できないオプションをサポートすることもあります。コマンドが前のリリースで使用するオブジェクトを作成するために使用された場合には、そのリリースに適したコンパイラーで処理されて、サポートされないオプションは認識されません。コンパイラーは、必ずしも処理できないオプションに関しての警告を出すとは限りません。

上

---

## ソート順序 (SRTSEQ)

ALPHABET文節の中でNLSSORTが英字名と関連している時に使用されるソート順序を指定します。SRTSEQパラメーターは、モジュールが使用するシステム定義またはユーザー定義のソート順序テーブルを決定するために、LANGIDパラメーターと一緒に使用されます。指定できる値は次の通りです。

- \*HEX** ソート順序テーブルは使用されず、ソート順序を決定するために文字の16進数値が使用されます。
- \*JOB** ソート順序は、コンパイル時にコンパイル・ジョブのソート順序を使用して分析解決され、モジュールと関連づけられます。ソート順序テーブルはコンパイル時にシステムに存在していなければなりません。実行時に、実行時ジョブのCCSIDのソート順序がコンパイル時ジョブのCCSIDと異なっている場合には、コンパイル時にロードされたソート順序テーブルが、実行時ジョブのCCSIDと一致するように変換されます。

### \*JOBRUN

プログラムのソート順序は、実行時に分析解決され、モジュールと関連づけられます。コンパイル時に、コンパイラーはコンパイル・ジョブのソート順序をモジュールと関連づけます。実行時に、このソート順序は実行時にジョブと関連づけるソート順序に置き換えられます。この値によって、モジュールはいったんコンパイルされて、実行時に別のソート順序と一緒に使用されます。

### \*LANGIDUNQ

使用中のソート順序テーブルにはコード・ページ中の各文字に対する固有の重みが入っていないことを指定します。使用されるソート順序テーブルは、LANGID パラメーターで指定された言語と対応した、固有の重みづけされたテーブルでなければなりません。

### **\*LANGIDSHR**

使用中のソート順序テーブルにはコード・ページ中の複数の文字に同じ重みを入れることができることを指定します。使用されるソート順序テーブルは、LANGIDパラメーターで指定された言語と対応した、共用の重みづけされたテーブルです。

### **テーブル名**

使用するソート順序テーブルの名前を指定してください。テーブルには指定されたコード・ページ中のすべての文字に対する重みが入っています。重みは、コード・ポイントで定義された文字と関連づけられています。ソート順序テーブル名を使用する時に、オブジェクトが存在するライブラリーを指定することができます。有効なライブラリーの値は次の通りです。

**\*LIBL** ソート順序テーブルが入っているライブラリーを見つけるためにライブラリー・リストが探索されます。

### **\*CURLIB**

現行ライブラリーが使用されます。現行ライブラリーとしてライブラリーが割り当てられていない場合には、QGPLが使用されます。

### **ライブラリー名**

ソート順序テーブルが入っているライブラリーの名前を入力してください。

上

---

## **言語ID (LANGID)**

ソート順序との組み合わせで使用される言語IDを指定します。LANGIDパラメーターは、有効なSRTSEQ値が\*LANGIDUNQまたは\*LANGIDSHRの時にだけ使用されます。指定できる値は次の通りです。

### **\*JOB RUN**

プログラムの言語IDは実行時に分析解決されます。コンパイル済みプログラムを実行する時に、ジョブの言語IDが使用されます。この値によって、モジュールはいったんコンパイルされて、実行時に別の言語IDと一緒に使用されます。

**\*JOB** モジュールの言語IDはコンパイル時に分析解決されます。

### **言語ID名**

有効な3文字の言語IDを入力してください。

上

---

## **パフォーマンス収集の活動化 (ENBPFRCOL)**

モジュールまたはプログラムの中でパフォーマンス測定コードを生成するかどうかを指定します。収集されたデータを使用して、システム・パフォーマンス・ツールでアプリケーションのパフォーマンスのプロファイルを作成することができます。コンパイル済みのモジュールまたはプログラムにパフォーマンス測定コードの追加を生成することにより、オブジェクトがわずかに大きくなり、パフォーマンスに影響することがあります。

**\*PEP** パフォーマンス統計は、プログラム・エントリー・プロシーチャーの入り口および出口でのみ収集されます。アプリケーションの全体的なパフォーマンス情報を収集したい場合には、この値を選択してください。このサポートは、前にTPSTツールが提供されたサポートと同等です。これは省略時の値です。

### **\*ENTRYEXIT**

パフォーマンス統計は、プログラムのすべてのプロシージャーの入り口および出口に収集されます。これには、プログラムPEPルーチンが含まれます。

この選択項目は、すべてのルーチンについての情報を収集したい場合に便利です。ユーザー・アプリケーションによって呼び出されるすべてのプログラムが\*PEP、\*ENTRYEXITまたは\*FULLオプションを使用してコンパイルされたことが分かっている場合には、このオプションを使用してください。そうでない場合には、ユーザー・アプリケーションがパフォーマンス測定が不能な他のプログラムを呼び出した場合には、パフォーマンス測定ツールは、それらのプログラムが資源を使用することはユーザー・アプリケーションに責任があるものと見なします。これにより、実際に資源がどこで使用されるのかを判別することが困難となります。

### **\*FULL**

パフォーマンス統計はすべてのプロシージャーの入り口および出口に収集されます。また、統計は外部プロシージャーに対する各呼び出しの前後に収集されます。

ユーザー・アプリケーションが、\*PEP、\*ENTRYEXITまたは\*FULLのいずれかを使用してコンパイルされたものではない他のプログラムを呼び出すと思われる場合には、このオプションを使用してください。このオプションを使用することにより、パフォーマンス・ツールは、ユーザー・アプリケーションによって使用される資源と、ユーザー・アプリケーションが呼び出すプログラムによって使用される資源を区別することができます（呼び出されるプログラムがパフォーマンス測定不能であっても）。このオプションは最も不経済ですが、アプリケーション中のいろいろなプログラムを選択的に分析することができます。

上

---

## **プロファイル・データ (PRFDTA)**

モジュールにプログラム・プロファイル・データ属性を指定します。プログラム・プロファイルは、統計データ（プロファイル・データ）に基づいてプロシージャーおよびプロシージャー内のコードを再順序づけるために使用される拡張最適化手法です。

### **\*NOCOL**

このモジュールはプロファイル・データを収集できません。これは省略時の値です。

**\*COL** このモジュールはプロファイル・データを収集できます。

注: \*COLを指定できるのは、モジュールの最適化レベルが\*FULLの時だけです。

上

---

## **コード化文字セットID (CCSID)**

実行時にファイル中のレコードとLOCALEに関連したデータが変換されるコード化文字セットID (CCSID)を指定します。

### **\*JOBRUN**

プログラムのCCSIDが実行時に解決されます。コンパイル済みプログラムを実行すると、現行ジョブの省略時値CCSIDが使用されます。

**\*JOB** コンパイル時の現行ジョブの省略時値CCSIDが使用されます。

**\*HEX** CCSID 65535が使用されます。これは、フィールドのデータがビット・データとして扱われ、変換されないことを示します。

## コード化文字セットID

使用するCCSIDを指定します。

上

---

## 国別CCSID (NTLCCSID)

すべての国別データ項目に使用される、国別コード化文字セットID (NTLCCSID)を指定します。指定できる値は次の通りです。

**13488** UCS-2 CCSID 13488が、すべての国別データ項目に使用されます。

### コード化文字セットID

指定されたCCSIDが、すべての国別データ項目に使用されます。指定するCCSIDはUCS-2互換（たとえばUTF-16 CCSID 1200）でなければなりません。

上

---

## 演算モード (ARITHMETIC)

数字データに演算モードを指定します。指定できる値は次の通りです。

### \*NOEXTEND

このオプションは、数字データの省略時の演算モードを指定します。固定小数点演算式の間接結果は最大30桁までで、数値リテラルの最大長は18桁だけです。

### \*EXTEND31

固定小数点演算の間接結果の精度を増すには、このオプションを使用してください。固定小数点演算式の間接結果は最大31桁までで、数値リテラルの最大長は31桁の場合があります。

### \*EXTEND63

固定小数点演算の間接結果の精度を増すには、このオプションを使用してください。固定小数点演算式の間接結果は最大63桁までで、数値リテラルの最大長は63桁とすることができます。

上

---

## PADDING CHARACTER (NTLPADCHAR)

以下のような変換状態で埋め込みが行われる時に使用される各国語埋め込み文字(NTLPADCHAR)を指定します。

1. 単一バイト文字を国別文字へ。
2. 2バイト文字を国別文字へ。
3. 国別文字を国別文字へ。

### \*DEFAULT

このオプションは、以下のような省略時の埋め込み文字を指定します。

1. 単一バイト文字を国別文字へ(NX"0020")
2. 2バイト文字を国別文字へ(NX"3000")

3.国別文字を国別文字へ(NX"3000")

#### 各国語16進リテラル

長さが1の有効な任意の各国語16進リテラルをNX" "またはNX' 'の形式で指定します。

上

---

## LICENSED INTERNAL CODE OPTIONS (LICOPT)

1つまたは複数のライセンス内部コード・コンパイル時オプションを指定します。このパラメーターを使用すれば個別のコンパイル時オプションを選択できますが、これは、選択した個々のコンパイラー・オプションのタイプの潜在的な利点と欠点を理解している上級プログラマーを対象としています。

上

---

## ディレクトリー組み込み (INCDIR)

コピー・ファイルを探すためにコンパイラーが使用する検索パスに追加する1つまたは複数のディレクトリーを指定します。コンパイラーは、ソース・プログラムのコピー・ファイルを解決できない場合には、ここで指定したディレクトリーを検索します。

### \*NONE

ユーザー・ディレクトリーでコピー・ファイルは検索されません。省略時値では、現行ディレクトリーが検索されます。

### 'ディレクトリー'

コピー・ファイルを検索する最大32のディレクトリーを指定してください。指定されたディレクトリーに加えて、現行ディレクトリーでもコピー・ファイルが検索されます。

上

---

## PGMインターフェース情報 (PGMINFO)

プログラム・インターフェース情報を生成するかどうか、および生成する位置を制御するオプションを指定します。

**生成** プログラム・インターフェース情報を生成するかどうかを指定します。指定できる値は次の通りです。

**\*NO** このオプションは、プログラム・インターフェース情報を生成しない省略時値を指定します。

### **\*PCML**

PCML（プログラム呼び出しマークアップ言語）を生成するように指定します。生成されたPCMLは、JAVAメソッドによるこのCOBOLモジュール内のプロシーチャーの呼び出しを容易にするのでJAVAコードは少なく済みます。

**位置** 「生成」パラメーターが\*PCMLである場合、プログラム情報が生成される位置を指定します。指定できる値は次の通りです。



#### **\*STMF**

プログラム情報をストリーム・ファイルに生成するように指定します。生成される情報が入るストリーム・ファイルの名前は、INFOSTMF オプションに指定されていなければなりません。

#### **\*MODULE**

プログラム情報をCOBOLモジュールに保管するように指定します。

**\*ALL** プログラム情報をストリーム・ファイルに生成し、さらにモジュールに保管するように指定します。生成される情報が入るストリーム・ファイルの名前は、INFOSTMF オプションに指定されていなければなりません。

上

---

## プログラム・インターフェース・ストリーム・ファイル (INFOSTMF)

PGMINFOオプションで指定された生成されたプログラム・インターフェース情報を入れるストリーム・ファイルのパス名を指定します。

このパス名は絶対的または相対的に修飾することができます。絶対パス名は'/'で始まり、相対パス名は'/'以外の文字で始まります。

絶対修飾の場合には、パス名は完全です。相対修飾の場合には、パス名は、パス名にジョブの現行作業ディレクトリーを付加することによって完了します。

このパラメーターを指定できるのは、PGMINFOパラメーターに\*NO以外の値がある場合だけです。

上

---

## 例

### 例1: ソース・プログラムをモジュール・オブジェクトにコンパイル

```
CRTCLMOD  MODULE(MYLIB/XMPLE1) SRCFILE(MYLIB/QCBLLESRC)
          SRCMBR(XMPLE1) OUTPUT(*PRINT)
          TEXT('MY ILE COBOL MODULE')
```

このコマンドは、ILE COBOLコンパイラーを呼び出し、XMPLE1という名前のモジュールを作成します。ソース・プログラムは、ライブラリーMYLIBのソース・ファイルQCBLLESRCのメンバーXMPLE1に入っています。コンパイル・リストが作成されます。

上

---

## エラー・メッセージ

### **\*ESCAPE** メッセージ

#### **LNC9001**

コンパイルは正常に行なわれません。&1は作成されていません。



**LNC9006**

TGTRLS(&1)が指定されましたが、コンパイラーが導入されていません。

**LNC9007**

プロダクト・ライブラリーに損傷があるか、あるいはユーザーにはその使用が認可されていない。

**LNC9015**

TGTRLS(&1)が正しくない。

上



## COBOLプログラム作成 (CRTCLPGM)

実行可能場所: すべての環境 (\*ALL)  
スレッド・セーフ: いいえ

パラメーター  
例  
エラー・メッセージ

CRTCLPGMコマンドは、COBOLソース・プログラムをコンパイルしてプログラム・オブジェクトを作成します。このコマンドは対話式でも、バッチ・モードでも、またCLプログラムに組み込んでも使用することができます。

CRTCLPGMコマンドに指定するすべてのオブジェクト名は英数字で構成しなければならず、また最初の文字は英字でなければなりません。名前の長さは10桁を超えることはできません。

上

### パラメーター

キーワード	記述	選択項目	ノート
PGM	プログラム	修飾オブジェクト名	オプション, 定位置 1
	修飾子 1: プログラム	名前, *PGMID	
	修飾子 2: ライブラリー	名前, *CURLIB	
SRCFILE	ソース・ファイル	修飾オブジェクト名	オプション, 定位置 2
	修飾子 1: ソース・ファイル	名前, <u>QLBLSRC</u>	
	修飾子 2: ライブラリー	名前, *LIBL, *CURLIB	
SRCMBR	ソース・メンバー	名前, *PGM	オプション, 定位置 3
GENLVL	生成重大度レベル	0-29, <u>29</u>	オプション
TEXT	テキスト記述	文字値, *SRCMBRTXT, *BLANK	オプション
OPTION	ソース・リスト・オプション	値 (最大 50 回の繰り返し): *SOURCE, *NOSOURCE, *SRC, *NOSRC, *NOXREF, *XREF, *GEN, *NOGEN, *NOSEQUENCE, *SEQUENCE, *NOVBSUM, *VBSUM, *NONUMBER, *NUMBER, *LINENUMBER, *NOMAP, *MAP, *NOOPTIONS, *OPTIONS, *QUOTE, *APOST, *SECLVL, *NOSECLVL, *PRTCORR, *NOPRTCORR, *NOSRCDBG, *SRCDBG, *NOLSTDBG, *LSTDBG, *PRINT, *NOPRINT	オプション, 定位置 4

キーワード	記述	選択項目	ノーツ
<b>GENOPT</b>	生成オプション	値 (最大 50 回の繰り返し): *NOLIST, *LIST, *NOXREF, *XREF, *NOPATCH, *PATCH, *NODUMP, *DUMP, *NOATR, *ATR, *RANGE, *NORANGE, *UNREF, *NOUNREF, *NOOPTIMIZE, *OPTIMIZE, *NODDSFILLER, *DDSFILLER, *NOSYNC, *SYNC, *NOCRTF, *CRTF, *NODUPKEYCHK, *DUPKEYCHK, *STDERR, *NOSTDERR, *NOEXTACCDSP, *EXTACCDSP, *NOINZDLT, *INZDLT, *NOBLK, *BLK, *STDINZ, *NOSTDINZ, *FS21DUPKY, *NOFS21DUPKY	オプション, 定位置 5
<b>CVTOPT</b>	変換オプション	値 (最大 50 回の繰り返し): *NOVARCHAR, *VARCHAR, *NODATETIME, *DATETIME, *NOGRAPHIC, *GRAPHIC	オプション
<b>SRTSEQ</b>	ソート順序	単一値: * <u>HEX</u> , *JOB, *JOBRUN, *LANGIDUNQ, *LANGIDSHR その他の値: 修飾オブジェクト名	オプション
	修飾子 1: ソート順序	名前	
	修飾子 2: ライブラリー	名前, * <u>LIBL</u> , *CURLIB	
<b>LANGID</b>	言語識別コード	文字値, * <u>JOBRUN</u> , *JOB	オプション
<b>MSGLMT</b>	メッセージ限界	要素リスト	オプション
	要素 1: メッセージの数	1-9999, * <u>NOMAX</u>	
	要素 2: メッセージ限界重大度	0-29, <u>29</u>	
<b>PRTFILE</b>	印刷ファイル	修飾オブジェクト名	オプション
	修飾子 1: 印刷ファイル	名前, <u>QSYSPRT</u>	
	修飾子 2: ライブラリー	名前, * <u>LIBL</u> , *CURLIB	
<b>FLAGSTD</b>	FIPSフラグ付け	値 (最大 24 回の繰り返し): *NOFIPS, *MINIMUM, *INTERMEDIATE, *HIGH, *NOSEG, *SEG1, *SEG2, *NODEB, *DEB1, *DEB2, *NOOBSOLETE, *OBSOLETE	オプション
<b>SAAFLAG</b>	SAAフラグ付け	* <u>NOFLAG</u> , *FLAG	オプション
<b>EXTDSPOPT</b>	拡張表示オプション	値 (最大 3 回の繰り返し): *DFRWRT, *NODFRWRT, *UNDSPCHR, *NOUNDSPCHR, *ACCUPDALL, *ACCUPDNE	オプション
<b>FLAG</b>	フラグ重大度	0-99, <u>0</u>	オプション
<b>REPLACE</b>	プログラムの置き換え	*NO, * <u>YES</u>	オプション
<b>TGTRLS</b>	ターゲット・リリース	単純名, * <u>CURRENT</u> , *PRV	オプション
<b>USRPRF</b>	ユーザー・プロファイル	* <u>USER</u> , *OWNER	オプション

キーワード	記述	選択項目	ノーツ
<b>AUT</b>	権限	名前, *LIBCRTAUT, *ALL, *CHANGE, *USE, *EXCLUDE	オプション
<b>DUMP</b>	コンパイラ・デバッグ・ダンプ	要素リスト	オプション
	要素 1:	1-65535, *	
	要素 2:	1-65535	
<b>ITDUMP</b>	中間テキスト・ダンプ	0-31, <u>0</u>	オプション

上

## プログラム (PGM)

作成中のCOBOLプログラム・オブジェクトのプログラム名およびライブラリー名を指定します。考えられる値は次の通りです。

### \*PGMID

プログラム・オブジェクトの名前はCOBOLソース・プログラムのPROGRAM-ID段落から取られます。

### プログラム名

コンパイル済みCOBOLプログラムを識別する名前を入力してください。このパラメーターにプログラム名を指定して、コンパイルをバッチ・モードで実行した場合には、バッチ・ジョブの最初のプログラムがこの名前を使用し、他のすべてのプログラムはソース・プログラムのPROGRAM-ID段落に指定された名前を使用します。

使用できるライブラリーの値は次の通りです。

### \*CURLIB

ライブラリー名を指定しない場合には、現行ライブラリーが使用されます。現行ライブラリーとしてライブラリーが割り当てられていない場合には、QGPLが使用されます。

### ライブラリー名

作成されたプログラム・オブジェクトを入れるライブラリーの名前を入力してください。

上

## ソース・ファイル (SRCFILE)

コンパイルしたいCOBOLソースが入っているソース・ファイルの名前を指定します。考えられる値は次の通りです。

### QLBLSRC

弊社提供のソース・ファイルQLBLSRCに、コンパイルされるCOBOLソース・プログラムが入っています。

### ソース・ファイル名

コンパイルされるCOBOLソース・プログラムが入っているソース・ファイルの名前を入力してください。このソース・ファイルはレコード長が92でなければなりません。

使用できるライブラリーの値は次の通りです。

**\*LIBL** ライブラリー名を指定しない場合には、ソース・ファイルが入っているライブラリーを見つけるために、システムはライブラリー・リストを検索します。

**\*CURLIB**

現行ライブラリーが使用されます。現行ライブラリーとしてライブラリーが割り当てられていない場合には、QGPLが使用されます。

**ライブラリー名**

ソース・ファイルが入っているライブラリーの名前を入力してください。

上

---

## ソース・メンバー (SRCMBR)

コンパイルしたいCOBOLソースが入っているメンバーの名前を指定します。このパラメーターを指定できるのは、SRCFILEパラメーターで参照したソース・ファイルがデータベース・ファイルの場合だけです。考えられる値は次の通りです。

**\*PGM** PGMパラメーターにプログラム名を指定した場合には、コンパイラーはそのプログラムと同じ名前のメンバーのソース・プログラムを探し、そのプログラムおよびメンバーと同じ名前のオブジェクト・プログラムを作成します。

PGMパラメーターにプログラム名を指定しなかった場合には、コンパイラーはデータベース・ソース・ファイルの最初のメンバーでプログラム・ソースを探し、PROGRAM-ID段落に指定された名前を使用してオブジェクト・プログラムを作成します。

**ソース・ファイル・メンバー名**

COBOLソース・プログラムが入っているメンバー名を入力してください。

上

---

## 生成重大度レベル (GENLVL)

プログラム・オブジェクトを作成するかどうかを決定する重大度レベルを指定します。この重大度レベルはプログラムのコンパイル時に作成されるメッセージの重大度レベルに対応します。エラー・メッセージの重大度レベルが指定した値より大きい場合には、プログラム・オブジェクトは作成されません。たとえば、このパラメーターに19を指定した場合には、メッセージのどれかの重大度レベルが20以上であれば、プログラム・オブジェクトは作成されません。

考えられる値は次の通りです。

**29** 重大度レベルが29より大きいエラーが起こった場合には、プログラム・オブジェクトは作成されません。

**重大度レベル**

1桁または2桁の数字(0-29)を指定してください。このレベルより大きい重大度レベルのエラーが起こった場合には、プログラム・オブジェクトは作成されません。

上

---

## テキスト'記述' (TEXT)

プログラムとその機能についての簡単な説明。考えられる値は次の通りです。

### \*SRCMBRTXT

プログラム・オブジェクトのテキストには、COBOLソースが入っているデータベース・ファイル・メンバーを記述するテキストと同じものを使用します。ソースが装置またはインライン・ファイルから入力された場合には、\*SRCMBRTXTの指定は、\*BLANKの指定と同じ効果となります。

### \*BLANK

テキストは指定されません。

### テキスト記述

プログラムおよびその機能を簡単に説明するテキストを入力してください。テキストは最長で50桁とすることができますが、アポストロフィで囲まなければなりません。アポストロフィは50桁のストリングの一部にはなりません。

上

---

## ソース・リスト・オプション (OPTION)

COBOLソースのコンパイル時に使用するオプションを指定します。考えられる値は次の通りです。

### \*SOURCEまたは\*SRC

コンパイラーは、COBOLソース・コードおよびすべてのコンパイル時エラー・メッセージから構成されるソース・リストを作成します。

### \*NOSOURCEまたは\*NOSRC

コンパイラーは、リストのソース部分を作成しません。ソース・リストが不要な場合には、このオプションを使用してください。コンパイル時間が短くなることがあります。

### \*NOXREF

コンパイラーは、ソース・プログラムの相互参照表を作成しません。

### \*XREF

コンパイラーは、ソース・プログラムの相互参照表を作成します。

\*GEN コンパイラーは、ソース・プログラムのコンパイル後にプログラム・オブジェクトを作成します。

### \*NOGEN

コンパイラーは、ソース・プログラムのコンパイル後にプログラム・オブジェクトを作成しません。この時点でエラー・リストだけが必要な場合には、このオプションを指定することができます。

### \*NOSEQUENCE

参照番号のシーケンス・エラーは検査されません。

### \*SEQUENCE

参照番号のシーケンス・エラーが検査されます。\*LINENUMBERオプションを指定した場合には、シーケンス・エラーは起こりません。

### \*NOVBSUM

動詞使用カウントは印刷されません。

### \*VBSUM

動詞使用カウントが印刷されます。



#### **\*NONUMBER**

ソース・ファイルの順序番号が参照番号として使用されます。

#### **\*NUMBER**

参照番号としてユーザー提供の順序番号（1-6桁目）が使用されます。

#### **\*LINENUMBER**

コンパイラーによって作成される順序番号が参照番号に使用されます。このオプションはプログラム・ソース・コードとCOPYステートメントによって取り入れられるソース・コードを結合して、1つの連続番号順序を生成します。FIPS（連邦情報処理標準）フラグ付けまたは、SAA\*フラグ付けを指定する場合には、このオプションを使用してください。

#### **\*NOMAP**

コンパイラーは、データ部マップをリストしません。

**\*MAP** コンパイラーは、データ部マップをリストします。

#### **\*NOOPTIONS**

このコンパイルでは、有効なオプションはリストされません。

#### **\*OPTIONS**

このコンパイルでは、有効なオプションがリストされます。

#### **\*QUOTE**

区切り引用符(")が非数値リテラルおよびブール・リテラルに使用されることを指定します。これはまた、表意定数QUOTEの値が引用符のEBCDIC値をもつことをも指定します。

#### **\*APOST**

区切りアポストロフィ(')が非数値リテラルおよびブール・リテラルに使用されることを指定します。これはまた、表意定数QUOTEの値がアポストロフィのEBCDIC 値をもつことをも指定します。

#### **\*NOSECLVL**

このコンパイルでは第2レベル・メッセージ・テキストはリストされません。

#### **\*SECLVL**

このコンパイルでは第2レベル・メッセージ・テキストがリストされます。

#### **\*PRTCORR**

コンパイラーは、CORRESPONDING句を使用した結果としてどの基本項目が組み込まれたかを示す注記行を、コンパイル・リストに挿入します。

#### **\*NOPRTCORR**

コンパイラーは、CORRESPONDING句が使用されても、注記行をコンパイル・リストに挿入しません。

#### **\*NOSRCDBG**

このオプションは、クライアント・ツールを使用してCOBOLプログラムをコンパイルした場合にプログラム式ワークステーションに表示される情報の種類を決めます。

コンパイラーは、ソース・レベル・デバッグ情報を作成しません。\*NOLSTDBGも有効となっている場合には、コンパイラーはソース・レベル・エラー情報も作成しません。

#### **\*SRCDBG**

このオプションは、クライアント・ツールを使用してCOBOLプログラムをコンパイルした場合にプログラム式ワークステーションに表示される情報の種類を決めます。

コンパイラーは、ソース・レベル・エラー情報およびソース・レベル・デバッグ情報を作成します。

\*SRCDBGと\*LSTDBGを同時に指定することはできません。いずれか一方のみを指定してください。

#### **\*NOLSTDBG**

このオプションは、クライアント・ツールを使用してCOBOLプログラムをコンパイルした場合にプログラム式ワークステーションに表示される情報の種類を決めます。

コンパイラーは、リスト・ビューまたはリスト・レベル・デバッグ情報を作成しません。

\*NOSRCDBGも有効であれば、コンパイラーはソース・レベル・エラー情報も作成しません。

#### **\*LSTDBG**

このオプションは、クライアント・ツールを使用してCOBOLプログラムをコンパイルした場合にプログラム式ワークステーションに表示される情報の種類を決めます。

コンパイラーは、リスト・ビュー、ソース・レベル・エラー情報、およびリスト・レベル・デバッグ情報を作成します。

\*SRCDBGと\*LSTDBGを同時に指定することはできません。いずれか一方のみを指定してください。

#### **\*PRINT**

コンパイラーはスプール・リストを作成します。

#### **\*NOPRINT**

コンパイラーはスプール・リストを作成しません。

上

---

## **生成オプション (GENOPT)**

プログラム・オブジェクトの作成時に使用するオプションを指定します。COBOLに問題がある場合には、リストが必要になることがあります。考えられる値は次の通りです。

#### **\*NOLIST**

IRP（プログラムの中間表現）、関連した16進コード、またはエラー・メッセージはリストされません。

\*LIST IRP,関連した16進コード、およびエラー・メッセージがリストされます。

#### **\*NOXREF**

IRPに定義されているオブジェクトの相互参照表は作成されません。

#### **\*XREF**

IRPに定義されているすべてのオブジェクトの相互参照表が作成されます。

#### **\*NOPATCH**

コンパイル済みプログラムにプログラム・パッチ域のスペースは予約されません。

#### **\*PATCH**

コンパイル済みプログラムにプログラム・パッチ域のスペースが予約されます。プログラム・パッチ域はデバッグ用に使用することができます。

#### **\*NODUMP**

プログラム・テンプレートはリストされません。

**\*DUMP**

プログラム・テンプレートがリストされます。

**\*NOATR**

IRPソースの属性はリストされません。

**\*ATR** IRPソースの属性がリストされます。

**\*RANGE**

実行時に、システムは添え字が正しい範囲内にあるかどうかを検査しますが、指標の範囲は検査しません。また、参照変更およびコンパイラ生成のサブストリング操作も検査します。

**\*NORANGE**

実行時に範囲は検査されません。

**\*UNREF**

参照されていないデータ項目がコンパイル済みプログラムに含まれます。

**\*NOUNREF**

参照されていないデータ項目はコンパイル済みプログラムに含まれません。これにより、使用されるODT（オブジェクト定義テーブル）の数が減るので、より大きなプログラムのコンパイルが可能になります。ただし、\*XREFオプションによって作成される相互参照表には、参照されていないデータ項目が現われます。

**\*NOOPTIMIZE**

コンパイラはプログラム用の標準最適化だけを実行します。

**\*OPTIMIZE**

作成されたプログラム・オブジェクトをより効率的に実行することができ、また必要な記憶域も少なく済みます。ただし、\*OPTIMIZEを指定すると、プログラムのコンパイルに必要な時間が大幅に増えることがあります。

**\*NODDSFILLER**

COPY DDSステートメントによって突き合わせフィールドが見つからない場合には、フィールド記述は作成されません。

**\*DDSFILLER**

COPY DDSステートメントによって突き合わせフィールドが見つからない場合には、単一文字のFILLERフィールド記述"07 FILLER PIC X"が常に作成されます。

**\*NOSYNC**

SYNCHRONIZED文節が構文検査されます。

**\*SYNC**

SYNCHRONIZED文節によって、記憶域の自然境界で基本項目の位置合わせが行われます。

**\*NOCRTF**

OPEN操作の時点で使用可能でないファイルは、動的に作成されません。

**\*CRTF**

OPEN操作の時点で使用可能でないファイルが、動的に作成されます。

**\*NODUPKEYCHK**

索引付きファイルの重複キー検査は行われません。

**\*DUPKEYCHK**

索引付きファイルの重複キー検査が行われます。

#### **\*STDERR**

標準エラー処理が使用されます。エラー処理の詳細については、COBOL/400ユーザーの手引きのエラー処理に関する章を参照してください。

#### **\*NOSTDERR**

バージョン1，リリース1および2のエラー処理方式が使用されます。エラー処理の詳細については、COBOL/400ユーザーの手引きのエラー処理に関する章を参照してください。

#### **\*NOEXTACCDSP**

コンパイラーで、拡張ACCEPTまたは拡張DISPLAYステートメントを使用することができません。

#### **\*EXTACCDSP**

コンパイラーで、拡張ACCEPTおよび拡張DISPLAYステートメントを使用することができます。

#### **\*NOINZDLT**

順次アクセスの相対ファイルが、OUTPUT用にオープンされた場合に、CLOSE操作時に削除済みレコードについて初期設定されません。すなわち、レコード境界はOPEN OUTPUT時に書き出されたレコード数によって決まります。後続のOPEN操作では、このレコード境界までしかアクセスすることができません。

#### **\*INZDLT**

順次アクセスの相対ファイルが、OUTPUT用にオープンされた場合に、CLOSE操作時に削除済みレコードについて初期設定されます。ファイル中の活動レコードは影響を受けません。すなわち、レコード境界は後続のOPEN操作のファイル・サイズとして定義されます。

#### **\*NOBLK**

コンパイラーは、STARTステートメントのない順次アクセス・ファイルのみのブロック化を許します。

BLOCK CONTAINS文節が指定されている場合、テープ・ファイルを除いてBLOCK CONTAINS文節は無視されます。

**\*BLK** BLOCK CONTAINSとともに使用すると、コンパイラーは動的アクセス・ファイルおよびSTARTステートメントのある順次アクセス・ファイルからのブロック化を許します。出力操作用にオープンされた相対ファイルのブロック化は許されません。

BLOCK CONTAINS文節は、ブロック化するレコードの数を制御します。

BLOCK CONTAINS文節を指定しない場合には、コンパイラーはSTARTステートメントのない順次アクセス・ファイルのみのブロック化を許します。オペレーティング・システムがブロック化するレコードの数を決定します。

#### **\*STDINZ**

コンパイラーは、データ項目がVALUE文節に依存していなければ、そのデータ項目をシステムの省略時の値に初期設定します。

#### **\*NOSTDINZ**

VALUE文節のない項目については、コンパイラーはデータ項目をシステム省略時値に初期設定しません。

#### **\*FS21DUPKY**

必須のREADステートメントと後続のREWRITEまたはDELETEステートメントの間でキーの値が変更された場合には、コンパイラーは、重複キーを含む索引付きファイルをランダムまたは動的アクセス・モードで処理している時に、ファイル状況21を報告します。

#### **\*NOFS21DUPKY**

コンパイラーは、重複キーを含む索引付きファイルをランダムまたは動的アクセス・モードで処理している時に、ファイル状況21を報告しません。REWRITEステートメントはレコードのキーを変更することができます。

上

---

## **変換オプション (CVTOPT)**

外部記述ファイルからCOPY DDSによってプログラムに渡されたシステム・アプリケーション体系(SAA)の日付、時刻、ならびにタイム・スタンプのデータ・タイプ、DBCSグラフィック・データ・タイプ、および可変長文字フィールドを、コンパイラーがどのように処理するかを指定します。考えられる値は次の通りです。

#### **\*NOVARCHAR**

可変長フィールドは無視され、FILLERフィールドとして宣言されます。

#### **\*VARCHAR**

可変長フィールドは固定長グループ項目として宣言され、プログラムでアクセスすることができます。

#### **\*NODATETIME**

日付、時刻、およびタイム・スタンプのデータ・タイプは無視され、FILLERフィールドとして宣言されます。

#### **\*DATETIME**

日付、時刻、および、タイム・スタンプのデータ・タイプは固定長文字フィールドとして宣言され、プログラムでアクセスすることができます。

#### **\*NOGRAPHIC**

グラフィック・データ・タイプは無視され、FILLERフィールドとして宣言されます。

#### **\*GRAPHIC**

固定長グラフィック・データ・タイプは固定長英数字フィールドとして宣言され、プログラムでアクセスすることができます。

\*VARCHARオプションも使用中の場合には、可変長DBCSグラフィック・データ・タイプは固定長グループ項目として宣言され、プログラムでアクセスすることができます。

上

---

## **ソート順序 (SRTSEQ)**

ALPHABET文節のアルファベット名がNLSSORTと関連付けられている場合に使用されるソート順序を指定します。SRTSEQパラメーターは、LANGIDパラメーターと一緒に使用して、プログラムがシステム定義のソート順序テーブルを使用するかまたはユーザー定義のソート順序テーブルを使用するかを判別します。考えられる値は次の通りです。

**\*HEX** ソート順序テーブルを使用せずに、ソート順序を判別するために、文字の16進数値を使用します。

**\*JOB** プログラムのソート順序は、コンパイル時に分析解決されてプログラムと関連付けられます。ソート順序テーブルは、コンパイル時にシステム内に存在しなければなりません。

### **\*JOBRUN**

プログラムのソート順序は、実行時に分析解決されてプログラムと関連付けられます。コンパイル時には、コンパイラーがコンパイル・ジョブのソート順序をプログラムと関連付けます。実行時には、このソート順序がジョブの実行時に関連付けられたソート順序と置き換えられます。

### **\*LANGIDUNQ**

使用されるソート順序テーブルにコード・ページの各文字に対する固有の重み付けが入っていないことを示します。使用されるソート順序テーブルは、LANGID パラメーターで指定された言語に対応する固有の重み付けテーブルとなります。

### **\*LANGIDSHR**

使用されるソート順序テーブルにコード・ページの複数の文字に対して同じ重み付けを入れることができることを示します。使用されるソート順序テーブルは、LANGIDパラメーターで指定された言語に対応する共用重み付けテーブルとなります。

### **テーブル名**

使用するソート順序テーブルの名前を入力してください。テーブルには、所定のコード・ページのすべての文字に対する重み付けが入っています。重み付けは、コード・ポイントで定義されている文字と対応します。ソート順序テーブル名を使用する場合には、テーブルが入っているライブラリーを指定することができます。有効な値は次の通りです。

**\*LIBL** ソート順序テーブルが入っているライブラリーを見つけるために、ライブラリー・リストを検索します。

### **\*CURLIB**

現行ライブラリーを検索します。現行ライブラリーとしてライブラリーが割り当てられていない場合には、QGPLが使用されます。

### **ライブラリー名**

ソート順序テーブルが入っているライブラリーの名前を入力してください。

注: SRTSEQの有効なPROCESSステートメントは、SRTSEQ(HEX), SRTSEQ(JOB), SRTSEQ(JOBRUN), SRTSEQ(LANGIDUNQ), SRTSEQ(LANGIDSHR), SRTSEQ("テーブル名"), SRTSEQ("ライブラリー名/テーブル名"), SRTSEQ("LIBL/テーブル名"),およびSRTSEQ("CURLIB/テーブル名")です。

上

---

## **言語識別コード (LANGID)**

ソート順序と一緒に使用する言語識別コードを指定します。LANGIDパラメーターが使用されるのは、有効なSRTSEQ値が\*LANGIDUNQまたは\*LANGIDSHRの場合だけです。考えられる値は次の通りです。

### **\*JOBRUN**

プログラムの言語識別コードは、実行時に分析解決されます。コンパイル済みプログラムが実行される場合には、ジョブの言語識別コードが使用されます。この値によってプログラムが1回コンパイルされ、実行時に別の言語識別コードで使用されます。

**\*JOB** プログラムの言語識別コードは、コンパイル時に分析解決されます。

### **言語識別コード**

有効な3文字の言語識別コードを入力してください。



注: LANGIDの有効なPROCESSステートメントは、LANGID(JOBRUN)、LANGID(JOB)、およびLANGID("言語識別コード")です。

上

---

## メッセージ限界 (MSG LMT)

一定のエラー重大度レベルのエラー・メッセージがいくつ出されるまでコンパイルを停止しないかを示す、発生してよいエラー・メッセージの最大数を指定して、コンパイルを制御します。

たとえば、重大度レベル20以上のエラーが4つ以上起こった場合にコンパイルを停止するよう指定することができます。この例の場合には、エラー・メッセージの最大数に3を、最大エラー重大度レベルに20を指定します。重大度レベルが20以上のエラーが3つ起こった場合コンパイルは続行されますが、4つ目のエラーが検出された時点でコンパイルは停止します。最大重大度レベルに達しないメッセージの場合には、検出されたエラーの数に関係なくコンパイルは続行されます。

### メッセージ限界

エラー・メッセージの最大数に使用できる値は次の通りです。

#### \*NOMAX

見つけたエラーの数にかかわらず、正常完了までコンパイルは続行されます。

**1-9999** 指定した重大度レベル以上のエラーが指定した数を超えた場合に、コンパイルは停止します。最大重大度レベルに達しないメッセージの場合には、検出されたエラーの数に関係なくコンパイルは続行されます。

### メッセージ重大度

最大エラー重大度レベルに使用できる値は次の通りです。

**29** 重大度レベルが29以上のエラーが指定したエラー・メッセージの最大数を超えた場合に、コンパイルは停止します。

#### 最大重大度レベル

1桁または2桁の数字(0-29)を指定してください。指定した重大度レベル以上のエラーの数が、指定したエラー・メッセージの最大数を超えると、コンパイルは停止します。

上

---

## 印刷ファイル (PRTFILE)

コンパイル・リストの送り先のファイルの名前、およびそのファイルが入っているライブラリーを指定します。このファイルのレコード長は最小でも132が必要です。132より小さいレコード長のファイルを指定した場合には、情報が失われます。

考えられる値は次の通りです。

#### QSYSPRT

ファイル名を指定しない場合には、コンパイル・リストは弊社提供ファイルQSYSPRTに送られます。



## ファイル名

コンパイラー・リストが送られるファイル名を入力してください。

使用できるライブラリーの値は次の通りです。

**\*LIBL** ファイルが入っているライブラリーを見つけるために、システムはライブラリー・リストを検索します。

## **\*CURLIB**

現行ライブラリーが使用されます。現行ライブラリーとしてライブラリーが割り当てられていない場合には、QGPLが使用されます。

## ライブラリー名

ファイルを入れるライブラリーの名前を入力してください。

上

---

## FIPSフラグ付け (FLAGSTD)

FIPSフラグ付けのオプションを指定します。考えられる値は次の通りです。

## **\*NOFIPS**

ソース・プログラムにFIPSフラグがありません。

## **\*MINIMUM**

最低サブセット以上のFIPSフラグです。

## **\*INTERMEDIATE**

中間サブセット以上のFIPSフラグです。

## **\*HIGH**

高サブセットのFIPSフラグです。

## **\*NOSEG**

任意選択モジュールSEGMENTATIONはFIPSフラグが付けられません。

**\*SEG1** 任意選択モジュールSEGMENTATIONレベル 1 以上のFIPSフラグです。

**\*SEG2** 任意選択モジュールSEGMENTATIONレベル 2 のFIPSフラグです。

## **\*NODEB**

任意選択モジュールDEBUGは、FIPSフラグが付けられません。

## **\*DEB1**

任意選択モジュールDEBUGレベル 1 以上のFIPSフラグです。

## **\*DEB2**

任意選択モジュールDEBUGレベル 2 のFIPSフラグです。

## **\*NOOBSOLETE**

使用されなくなった言語要素には、フラグが付けられません。

## **\*OBSOLETE**

使用されなくなった言語要素にフラグが付けられます。

上

---

## SAAフラグ付け (SAAFLAG)

SAA COBOLでサポートされていないCOBOL/400\*機能のフラグ付けが必要であるかどうかを指定します。考えられる値は次の通りです。

### \*NOFLAG

SAA COBOLフラグ付けは実行されません。

### \*FLAG

SAA COBOLフラグ付けが実行されます。

上

---

## 拡張表示オプション (EXTDSPOPT)

ワークステーション入出力用の拡張ACCEPTおよびDISPLAYステートメントのために使用するオプションを指定します。考えられる値は次の通りです。

### \*DFRWRT

拡張DISPLAYステートメントは、拡張ACCEPTステートメントが検出されるまで、またはバッファがいっぱいになるまで、バッファに保持されます。

拡張ACCEPTステートメントが検出されないうちにバッファがいっぱいになった場合には、バッファの内容が画面に書き出されます。拡張ACCEPTステートメントが検出されると、その時点のバッファの内容が画面に書き出されます。

### \*NODFRWRT

拡張DISPLAYステートメントは、それぞれが見つかった時に実行されます。

### \*UNDSPCHR

表示可能文字と表示不能文字が拡張ACCEPTおよび拡張DISPLAYステートメントによって処理されます。

### \*NOUNDSPCHR

表示可能文字だけが拡張ACCEPTおよび拡張DISPLAYステートメントによって処理されます。

このオプションはリモート3174および3274制御装置に接続された表示装置について使用するものですが、ローカル・ワークステーションについても使用することができます。このオプションを使用する場合には、データに表示可能文字が入っていないければなりません。データに16進20より小さい値が入っているとその結果は予期できず、予期しない画面様式になる場合から重大エラーになる場合まであります。

### \*ACCUPDALL

UPDATE句の有無にかかわらず、すべてのタイプのデータが拡張ACCEPTステートメントで事前表示されます。

### \*ACCUPDNE

数字編集データのみが、UPDATE句を含まない拡張ACCEPTステートメントで事前表示されます。

上

---

## フラグ重大度 (FLAG)

印刷されるメッセージの最低重大度レベルを指定します。考えられる値は次の通りです。

**0**       すべてのメッセージが印刷されます。

### 重大度レベル

印刷されるメッセージの最低重大度レベルを指定する 1 桁または 2 桁の数字を入力してください。  
指定した値以上の重大度レベルをもつメッセージがリストされます。

上

---

## プログラムの置き換え (REPLACE)

同じライブラリーに同じ名前のプログラム・オブジェクトがすでに存在している場合に、新しいプログラム・オブジェクトを作成するかどうかを指定します。考えられる値は次の通りです。

**\*YES**   新しいプログラム・オブジェクトが作成され、指定したライブラリー中の同じ名前の既存のプログラム・オブジェクトはライブラリーQRPLOBJに移動されます。

**\*NO**    同じ名前のプログラム・オブジェクトが指定したライブラリーにすでに存在している場合には、新しいプログラム・オブジェクトは作成されません。

上

---

## ターゲット・リリース (TGTRLS)

作成中のオブジェクトを使用したいオペレーティング・システムのリリース・レベルを指定します。

正確なリリース・レベルはVXRXXMXの形式で指定することができます。ここで、VX はバージョン、RXはリリース、およびMXはモディフィケーション・レベルです。たとえば、V3R1M0はバージョン3、リリース1、モディフィケーション・レベル0です。

注: ターゲット・システムでオブジェクトを使用するためには、そのオブジェクトを作成コマンドに指定したターゲット・リリース・レベルに保管してから、それをターゲット・システムに復元しなければなりません。

たとえば、システムがV3R1M0を実行中に、V2R3M0システムに配布するプログラム・オブジェクトを作成したい場合には、そのプログラムをTGTRLS(V2R3M0)またはTGTRLS(\*PRV)で作成し、プログラムをTGTRLS(V2R3M0)またはTGTRLS(\*PRV)で保管して、V2R3M0システムで復元しなければなりません。また、このプログラム・オブジェクトは、V3R1M0システムでも復元することもできます。

注: プログラムは、作成コマンドに指定したものより前のリリース・レベルに復元することができます。プログラムを実行することができる最も古いリリースを判別するためには、DSPPGMを使用してください。

### **\*CURRENT**

このオブジェクトは、現在ユーザーのシステムで実行中のオペレーティング・システムのリリースで使用するものです。たとえば、システムでV3R1M0が実行中の場合には、\*CURRENTは、V3R1M0が導入されているシステムでこのオブジェクトを使用することを意味します。また、オペレーティング・システムの後続のリリースが導入されたシステムでもこのオブジェクトを使用することができます。

**\*PRV**   このオブジェクトは、オペレーティング・システムのモディフィケーション・レベルが0である前のリリースで使用するものです。たとえば、システムでV3R1M0 が実行中の場合には、

\*PRVは、V2R3M0が導入されているシステムでこのオブジェクトを使用することを意味します。また、オペレーティング・システムの後続のリリースが導入されたシステムでもこのオブジェクトを使用することができます。

### リリース・レベル

リリースをVXRXXMXの形式で指定してください。オブジェクトは、指定されたリリースのシステムまたはオペレーティング・システムの後続のリリースが導入されたシステムで使用することができます。

有効な値は現行のバージョン、リリース、およびモディフィケーション・レベルによって決まり、新しいリリースごとに変わります。有効な値のリストを参照するためには、そのコマンドのTGTRLSパラメーターでF4=プロンプトを押してください。

上

---

## ユーザー・プロファイル (USRPRF)

コンパイル済みCOBOLプログラムを実行するユーザー・プロファイルを指定します。プログラムを実行し、プログラム（プログラムが各オブジェクトに対してもっている権限を含む）によって使用できるオブジェクトを制御するために、プログラム所有者またはプログラム・ユーザーのプロファイルが使用されます。

考えられる値は次の通りです。

### \*USER

プログラムの実行時に、プログラムの使用者のユーザー・プロファイルが使用されます。

### \*OWNER

プログラムの実行時に、プログラムの所有者と使用者の両方のユーザー・プロファイルが使用されます。プログラムの実行中にオブジェクトを見つけてアクセスするためには、両方のユーザー・プロファイルのオブジェクト権限が一括して使用されます。プログラムの実行中に作成されたオブジェクトはそのプログラムのユーザーが所有します。

上

---

## 権限 (AUT)

プログラム・オブジェクトに対して特定権限をもっていないか、権限リスト上にないか、あるいはグループがプログラム・オブジェクトに対して特定権限をもっていないユーザーに認可される権限を指定します。この権限は、プログラム・オブジェクトの作成後に、GRTOBJAUT（オブジェクト権限認可）またはRVKOBJAUT（オブジェクト権限取り消し）コマンドを使用して、すべてのユーザーまたは個々のユーザーについて変更することができます。

考えられる値は次の通りです。

### \*LIBCRTAUT

オブジェクトの共通権限は、ターゲット・ライブラリー（作成されたプログラム・オブジェクトが入れられるライブラリー）のCRTAUTキーワードから取られます。この値はプログラム・オブジェクトの作成時に決まります。プログラム・オブジェクトの作成後にライブラリーのCRTAUT値が変わっても、新しい値が既存のオブジェクトに影響を及ぼすことはありません。

**\*ALL** 所有者だけに限定されるかまたは権限リスト管理権限によって制御される操作を除くすべての操作をプログラム・オブジェクトに対して実行できる権限を提供します。ユーザーは、プログラム・オ

プロジェクトの存在の制御、プログラム・オブジェクトの機密保護の指定、プログラム・オブジェクトの変更、およびプログラム・オブジェクトに対する基本的な機能の実行を行うことができますが、プログラム・オブジェクトの所有権を移すことはできません。

#### **\*CHANGE**

すべてのデータ権限と、所有者にだけ限定されるかまたはオブジェクト権限およびオブジェクト管理権限によって制御される操作を除くすべての操作をプログラム・オブジェクトに対して実行できる権限を提供します。ユーザーは、プログラム・オブジェクトを変更したり、プログラム・オブジェクトの実行やデバッグなどの基本的な機能をプログラム・オブジェクトに対して実行したりすることができます。

**\*USE** オブジェクト操作権限および読み取り権限（プログラムの実行などの基本的な操作をプログラム・オブジェクトに対して実行できる権限）を提供します。ユーザーはオブジェクトを変更することができません。

注：プログラムの変数の定様式ダンプを取得するには、プログラムに対して**\*USE**権限を持っていないければなりません。変数をダンプするには、さらにプログラムが検討可能な情報も持っていないければなりません。

一部のユーザーが変数のダンプができることを望まない場合は、ユーザーに**\*OBJOPR**だけに加えてプログラムに**\*EXECUTE**権限を与えてください。これはユーザーにプログラムの呼び出しは許可しますが、その変数をダンプすることは許可しません。

すべてのユーザーに変数のダンプができることを望まない場合は、プログラムの変更(**CHGPGM**)を使用して、プログラムの検討可能な情報を除去してください。

#### **\*EXCLUDE**

ユーザーはプログラム・オブジェクトにアクセスすることができません。

#### **権限リスト名**

プログラムが追加されるユーザーおよび権限の権限リストの名前を入力してください。プログラム・オブジェクトはこの権限リストによって保護され、プログラム・オブジェクトの共通権限は**\*AUTL**に設定されます。この権限リストは、**CRTCLPGM**コマンドが出される時にシステム上に存在していなければなりません。

上

---

## **コンパイラー・デバッグ・ダンプ (DUMP)**

IBM COBOLデバッグ援助機能。(弊社技術員用)

上

---

## **中間テキスト・ダンプ (ITDUMP)**

コンパイル中の一定の時点にコンパイラーに内部テキストをダンプさせるIBMデバッグ援助機能。(弊社技術員用)

上

---

## 例

### 例1: ソース・プログラムをプログラム・オブジェクトにコンパイル

```
CRTCBLPGM  PGM(MYLIB/XMPLE1) SRCFILE(MYLIB/QLBLSRC)
            SRCMBR(XMPLE1)  OPTION(*SOURCE)
            TEXT('MY COBOL PROGRAM')
```

このコマンドはCOBOLコンパイラを呼び出して、XMPLE1という名前のプログラムを作成します。ソース・プログラムはライブラリーMYLIB中のソース・ファイルQLBLSRCのメンバーXMPLE1にあります。コンパイラ・リストが作成されます。

上

---

## エラー・メッセージ

### \*ESCAPE メッセージ

#### **LBL9001**

コンパイルは正常に行われませんでした。プログラムは作成されません。

#### **LBL9004**

プログラムはすでに存在している。コンパイルは正常に行われませんでした。

#### **LBL9006**

TGTRLS(&1)が指定されたが、前のコンパイラが導入されていない。

#### **LBL9007**

プロダクト・ライブラリーに損傷があるか、あるいはユーザーにはその使用が認可されていない。

#### **SQL9002**

SQLプリコンパイルと&7コンパイルのTGTRLSパラメーターが矛盾している。

#### **SQL9003**

このSQLソースには、&7コンパイルのレベルが正しくない。

上

## Cモジュール作成 (CRTCMOD)

実行可能場所: すべての環境 (\*ALL)

スレッド・セーフ: いいえ

パラメーター

例

エラー・メッセージ

Cモジュール作成(CRTCMOD)コマンドはILE Cコンパイラーを開始します。このコンパイラーは、ソース・コード内のILE Cステートメントに基づいてモジュール・オブジェクトの作成を試みます。

上

### パラメーター

キーワード	記述	選択項目	ノーツ
MODULE	モジュール	修飾オブジェクト名	必須, 定位置 1
	修飾子 1: モジュール	名前	
	修飾子 2: ライブラリー	名前, <u>*CURLIB</u>	
SRCFILE	ソース・ファイル	修飾オブジェクト名	オプション, 定位置 2
	修飾子 1: ソース・ファイル	名前, <u>QCSRC</u>	
	修飾子 2: ライブラリー	名前, <u>*LIBL</u> , *CURLIB	
SRCMBR	ソース・メンバー	名前, <u>*MODULE</u>	オプション, 定位置 3
SRCSTMF	ソース・ストリーム・ファイル	パス名	オプション
TEXT	テキスト・記述	文字値, <u>*SRCMBRTXT</u> , *BLANK	オプション
OUTPUT	出力オプション	単一値: <u>*NONE</u> , '*トナトオ' その他の値: 要素リスト	オプション
	要素 1: 出力ファイル名	パス名, *PRINT, '*ニネクトホ'	
	要素 2: タイトル	文字値, <u>*BLANK</u>	
	要素 3: サブタイトル	文字値, <u>*BLANK</u>	



キーワード	記述	選択項目	ノート
<b>OPTION</b>	コンパイラー・オプション	値 (最大 35 回の繰り返し): *NOAGR, *AGR, *DIGRAPH, *NODIGRAPH, *NOEVENTF, *EVENTF, *NOEXPMAC, *EXPMAC, *NOFULL, *FULL, *GEN, *NOGEN, *NOINCDIRFIRST, *INCDIRFIRST, *LOGMSG, *NOLOGMSG, *NOPONLY, *PPONLY, *NOSECLVL, *SECLVL, *NOSHOWINC, *SHOWINC, *NOSHOWSKP, *SHOWSKP, *SHOWSRC, *NOSHOWSRC, *NOSHOWSYS, *SHOWSYS, *NOSHOWUSR, *SHOWUSR, *STDINC, *NOSTDINC, *NOSTDLOGMSG, *STDLOGMSG, *NOSTRUCREF, *STRUCREF, *NOSYSINCPATH, *SYSINCPATH, *NOXREF, *XREF, *NOXREFREF, *XREFREF	オプション
<b>CHECKOUT</b>	チェックアウト・オプション	値 (最大 39 回の繰り返し): <u>*NONE</u> , *USAGE, *ALL, *NOCOND, *COND, *NOCONST, *CONST, *NOEFFECT, *EFFECT, *NOENUM, *ENUM, *NOEXTERN, *EXTERN, *NOGENERAL, *GENERAL, *NOGOTO, *GOTO, *NOINIT, *INIT, *NOPARM, *PARM, *NOPORT, *PORT, *NOPPCHECK, *PPCHECK, *NOPPTRACE, *PPTRACE, *NOREACH, *REACH, *NOTRUNC, *TRUNC, *NOUNUSED, *UNUSED	オプション
<b>OPTIMIZE</b>	最適化	<b>10</b> , 20, 30, 40	オプション
<b>INLINE</b>	インライン・オプション	要素リスト	オプション
	要素 1: インライナー	<u>*OFF</u> , *ON	
	要素 2: モード	<u>*AUTO</u> , *NOAUTO	
	要素 3: しきい値	1-65535, <b>250</b> , *NOLIMIT	
	要素 4: 限界	1-65535, <b>2000</b> , *NOLIMIT	
	要素 5: 報告書	<u>*NO</u> , *YES	
<b>MODCRTOPT</b>	モジュール作成オプション	<u>*NOKEEPILDTA</u> , *KEEPILDTA	オプション
<b>DBGVIEW</b>	デバッグ・ビュー	<u>*NONE</u> , *ALL, *STMT, *SOURCE, *LIST	オプション
<b>DEFINE</b>	名前の定義	単一値: <b>*NONE</b> その他の値 (最大 32 回の繰り返し): 文字値	オプション

キーワード	記述	選択項目	ノート
LANGLVL	言語レベル	<u>*EXTENDED</u> , *ANSI	オプション
ALIAS	別名	値 (最大 3 回の繰り返し): <u>*ANSI</u> , *NOANSI, *ADDRTAKEN, *NOADDRTAKEN, *ALLPTRS, *NOALLPTRS, *TYPEPTR, *NOTYPEPTR	オプション
SYSIFCOPT	SYSTEMインターフェースOPT	値 (最大 2 回の繰り返し): *NOIFSIO, *IFSIO, *IFS64IO, *NOASYNCSIGNAL, *ASYNCSIGNAL	オプション
LOCALETYPE	LOCALEオブジェクト・タイプ	<u>*LOCALE</u> , *LOCALEUCS2, *LOCALEUTF, *CLD	オプション
FLAG	メッセージのフラグ・レベル	<u>0</u> , 10, 20, 30	オプション
MSGMT	コンパイラ・メッセージ	要素リスト	オプション
	要素 1: メッセージ限界	0-32767, <u>*NOMAX</u>	
	要素 2: メッセージ限界の重大度	0, 10, 20, <u>30</u>	
REPLACE	MODULE OBJECTの置き換え	<u>*YES</u> , *NO	オプション
AUT	権限	名前, <u>*LIBCRTAUT</u> , *ALL, *CHANGE, *USE, *EXCLUDE	オプション
TGTRLS	ターゲット・リリース	単純名, <u>*CURRENT</u> , *PRV	オプション
ENBPFCOL	パフォーマンス収集の使用可能	要素リスト	オプション
	要素 1: 収集レベル	<u>*PEP</u> , *ENTRYEXIT, *FULL	
	要素 2: プロシージャ	*NONLEAF, *ALLPRC	
PFOPT	パフォーマンス・オプション	値 (最大 2 回の繰り返し): *SETFPCA, *NOSETFPCA, *NOSTRDONLY, *STRDONLY	オプション
PRFDTA	プロファイル作成データ	<u>*NOCOL</u> , *COL	オプション
TERASPACE	テラスペース・オプション	単一値: <u>*NO</u> その他の値: 要素リスト	オプション
	要素 1: テラスペース使用可能	<u>*YES</u>	
	要素 2: TERASPACE INTERFACESの使用	<u>*NOTSIFC</u> , *TSIFC	
STGMDL	ストレージ・モデル	<u>*SINGLVL</u> , *TERASPACE, *INHERIT	オプション
DTAMD	データ・モデル	<u>*P128</u> , *LLP64	オプション
PACKSTRUCT	パック構造	<u>*NATURAL</u> , 1, 2, 4, 8, 16	オプション
ENUM	Enumサイズ	<u>*SMALL</u> , 1, 2, 4, *INT	オプション
MAKEDEP	依存関係情報	パス名, <u>*NONE</u>	オプション
PPGENOPT	プリプロセッサ・オプション	単一値: <u>*NONE</u> , *DFT その他の値 (最大 2 回の繰り返し): *RMVCOMMENT, *NORMVCOMMENT, *GENLINE, *NOGENLINE	オプション
PPSRCFILE	出力ソース・ファイル	修飾オブジェクト名	オプション
	修飾子 1: 出力ソース・ファイル	名前	
	修飾子 2: ライブラリー	名前, <u>*CURLIB</u>	
PPSRCMBR	出力ソース・メンバー	名前, <u>*MODULE</u>	オプション

キーワード	記述	選択項目	ノーツ
PPSRCSTMF	出力ストリーム・ファイル	パス名, *SRCSTMF	オプション
INCDIR	組み込みディレクトリー	単一値: <u>*NONE</u> その他の値 (最大 32 回の繰り返し): パス名	オプション
CSOPT	コンパイラ・サービスOPT	文字値, <u>*NONE</u>	オプション
LICOPT	ライセンス内部コードOPT	文字値, <u>*NONE</u>	オプション
DFTCHAR	省略時の文字タイプ	<u>*UNSIGNED</u> , *SIGNED	オプション
TGTCCSID	ターゲットCCSID	1-65535, <u>*SOURCE</u> , *JOB, *HEX	オプション
DECFLTRND	10進浮動丸めモード	<u>*HALFEVEN</u> , *DOWN, *UP, *HALFUP, *HALFDOWN, *FLOOR, *CEILING	オプション

上

## モジュール (MODULE)

作成するモジュール・オブジェクトを指定します。

これは必須パラメーターです。

### 修飾子1: モジュール

**name** モジュール・オブジェクトの名前を指定してください。

### 修飾子2: ライブラリー

#### \*CURLIB

モジュール・オブジェクトは現行ライブラリーに保管されます。ジョブに現行ライブラリーがない場合には、モジュール・オブジェクトはQGPLライブラリーの中に作成されます。

**name** モジュール・オブジェクトの保管先となるライブラリーの名前を指定してください。

上

## ソース・ファイル (SRCFILE)

コンパイルが必要なILE Cソース・コードが入っているソース・ファイルを指定します。

### 修飾子1: ソース・ファイル

#### QCSRC

QCSRCという名前のソース・ファイルには、コンパイルしたいILE Cソース・コードを含むメンバーが入っています。

**name** ILE Cソース・コードが含まれたメンバーが入っているファイルの名前を指定してください。

### 修飾子2: ライブラリー

**\*LIBL** ソース・ファイルが入っているライブラリーを見つけるためにライブラリー・リストが検索されます。

### **\*CURLIB**

ソース・ファイルが現行ライブラリーから検索されます。ジョブに現行ライブラリーがない場合には、ソース・ファイルはQGPLから検索されます。

**name** ソース・ファイルが入っているライブラリーの名前を指定してください。

上

---

## **ソース・メンバー (SRCMBR)**

コンパイルされるソース・コードが入っているソース・ファイル・メンバーを指定します。

### **\*MODULE**

モジュール (MODULE)パラメーターに指定された名前は、ソース・メンバー名として使用されます。

**name** ソース・コードが入っているメンバーの名前を指定してください。

上

---

## **ソース・ストリーム・ファイル (SRCSTMF)**

コンパイルが必要なソース・コードが入っているストリーム・ファイルを指定します。

パス名は絶対修飾パス名か相対修飾パス名のいずれかとすることができます。絶対パス名は/で始まり、相対パス名は/以外の文字で始まります。絶対修飾した場合には、そのパス名で完全です。相対修飾した場合には、パス名は、ジョブの現行作業ディレクトリーをパス名に対して事前保留することによって完全なものとなります。

SRCMBRおよびSRCFILEパラメーターをSRCSTMFパラメーターと一緒に指定することはできません。

### **path-name**

C++ソース・コードが入っているストリーム・ファイルのパス名を指定してください。

上

---

## **テキスト'記述' (TEXT)**

モジュール・オブジェクトを簡潔に記述するテキストを指定します。

### **\*SRCMBRTXT**

モジュール・オブジェクトに使用されるソース・ファイル・メンバーと関連したテキスト記述。ソース・ファイルがインライン・ファイル、ストリーム・ファイル、または装置ファイルである場合には、テキストはブランクとなります。

### **\*BLANK**

テキストは表示されないことを指定します。

**文字値** 50文字を超えないテキストをアポストロフィで囲んで指定します。

上

---

## 出力オプション (OUTPUT)

コンパイラー・リストが生成されるかどうかを指定します。

単一値

### \*NONE

コンパイラー・リストは生成しません。 リストが必要でない場合には、コンパイル時のパフォーマンスを改善するためにこのパラメーター値を使用する必要があります。 \*NONEを指定した場合には、OPTIONパラメーターに指定されたリストと関連したすべてのパラメーター値が無視されます。

要素1: 出力ファイル名

### \*PRINT

リストを含むスプール・ファイルが生成されます。

### path-name

リストを保留するストリーム・ファイルのパス名を指定します。

要素2: タイトル

### \*BLANK

テキストは表示されないことを指定します。

文字値 リスト・ファイルのタイトル文字列（最大80文字）を指定します。

要素3: サブタイトル

### \*BLANK

テキストは表示されないことを指定します。

文字値 リスト・ファイルのサブタイトル文字列（最大80文字）を指定します。

上

---

## コンパイラー・オプション (OPTION)

ILE Cソース・コードのコンパイル時に使用するオプションを指定します。 それらは、1つまたは複数のブランクで区切り、任意の順序で指定できます。1つのオプションが複数回指定されたり、2つのオプションが競合する場合には、最後に指定されたものが使用されます。

このパラメーターには35個の値を指定できます。

このパラメーターには単一のデフォルト値はありません。以下に示す一連のパラメーター値の説明では、下線付きの値は、コンパイラーで使用されるデフォルト値を表します。

### \*NOAGR

リストの中に集約マップは生成しません。

**\*AGR** リストの中にすべての集約のマップを生成します。マップには構造と共用体が組み込まれます。構造マップには、メンバーの埋め込みが示されます。このオプションによって\*STRUCREFオプションが上書きされます。

### \*DIGRAPH

ソース・コードの二重音字の使用を許可します。

#### **\*NODIGRAPH**

ソース・コードの二重音字の使用は許可しません。

#### **\*NOEVENTF**

CoOperative Development Environment/400 (CODE/400)で使用するためのイベント・ファイルは作成しません。

#### **\*EVENTF**

CoOperative Development Environment/400 (CODE/400)で使用するためのイベント・ファイルを作成します。イベント・ファイルは、作成中のモジュールまたはプログラム・オブジェクトの保管先のライブラリーにあるファイルEVFEVENTにメンバーとして作成されます。ファイルEVFEVENTが存在しない場合には、それが自動的に作成されます。イベント・ファイル・メンバー名は作成中のオブジェクトの名前と同じです。CODE/400はこのファイルを使用して、CODE/400エディターに統合されたエラー・フィードバックを提供します。通常、イベント・ファイルは、CODE/400内からモジュールまたはプログラム・オブジェクトを作成するときに生成されます。

#### **\*NOEXPMAC**

マクロの中で構文エラーが検出されないかぎり、マクロはリスト中で拡張されません。

#### **\*EXPMAC**

リスト中ですべてのマクロを拡張します。このパラメーターは、DBGVIEW(\*ALL)およびDBGVIEW(\*LIST)と競合します。DBGVIEW(\*ALL)やDBGVIEW(\*LIST)と一緒にOPTION(\*EXPMAC)を使用すると、エラー・メッセージを出力してコンパイルは停止します。

#### **\*NOFULL**

すべてのリスト・オプションをオンにしません。

#### **\*FULL**

すべてのリスト・オプションをオンにします。

**\*GEN** コンパイル・プロセスのすべてのフェーズが実行されます。

#### **\*NOGEN**

コンパイルは構文検査の後に停止します。モジュール・オブジェクトは作成されません。

#### **\*NOINCDIRFIRST**

INCDIRパラメーターとして指定された組み込みディレクトリーは、標準ヘッダー・ファイル組み込みパスの前には組み込まれません。

#### **\*INCDIRFIRST**

INCDIRパラメーターとして指定された組み込みディレクトリーが、標準ヘッダー・ファイル組み込みパスの前に組み込まれます。

#### **\*LOGMSG**

コンパイル・メッセージをジョブ・ログに書き込みます。

このオプションとFLAGパラメーターを指定した場合には、FLAGパラメーターに指定された（およびそれ以上の）重大度のメッセージがジョブ・ログに入れます。

このオプションと,MSGLMTパラメーターにメッセージの最大数を指定した場合には、指定された重大度のその数のメッセージがジョブ・ログに入れられた時にコンパイルが停止します。

#### **\*NOLOGMSG**

コンパイル・メッセージをジョブ・ログに書き込みません。

#### **\*NOPPONLY**

\*GENがOPTIONの場合の省略時値のまま残されている場合には、コンパイラーはコンパイル手順全体を実行します。

#### **\*PPONLY**

プリプロセッサだけが実行されて、出力は保管されます。コンパイル手順の残りの部分は実行されません。

SRCFILEまたはSRCMBRが指定された場合には、出力はライブラリーQTEMPのソース・ファイルQACZEXPANDに保管されます。メンバー名は、MODULEパラメーターに指定された名前と同じです。ジョブがバッチ・モードで投入された場合には、出力はQTEMPに保管されているので、ジョブが完了すると出力は削除されます。

SRCSTMFが指定された場合には、出力はユーザーの現行ディレクトリー中のストリーム・ファイルに保管されます。ファイル名は、SRCSTMFのファイルに拡張子'.I'を付けたものと同じです。

OPTION(\*PPONLY)は、パラメーターPPGENOPT, PPSRCFILE, PPSRCMBR,およびPPSRCSTMFによって置き換えられています。

#### **\*NOSECLVL**

リストの中に第2レベル・メッセージ・テキストは生成しません。

#### **\*SECLVL**

リストの中に第2レベル・メッセージ・テキストを生成します。このオプションを有効にするには、OUTPUTオプションを指定する必要があります。

#### **\*NOSHOWINC**

リストのソース・セクションで、またはデバッグ・ビューでユーザー組み込みファイルまたはシステム組み込みファイルを拡張しません。

#### **\*SHOWINC**

リストのソース・セクションで、またはデバッグ・ビューでユーザー組み込みファイルとシステム組み込みファイルの両方を拡張します。OUTPUTオプションか、\*ALL, \*SOURCE,または\*LISTのDBGVIEWパラメーター値を指定する必要があります。

#### **\*NOSHOWSKP**

リストのソース・セクションまたはデバッグ・リスト・ビューでプリプロセッサが無視したステートメントを組み込みません。プリプロセッサは、プリプロセッサのディレクティブが偽（ゼロ）と評価した結果としてステートメントを無視します。

#### **\*SHOWSKP**

プリプロセッサがスキップしたかどうかには関係なく、リストのソース・セクションまたはデバッグ・リスト・ビュー内のすべてのステートメントを組み込みます。OUTPUTオプションか、\*ALLまたは\*LISTのDBGVIEWパラメーター値を指定する必要があります。

#### **\*SHOWSRC**

リストにソース・コードを表示します。このオプションは、\*SHOWINC, \*SHOWSYS,または\*SHOWUSRオプションによって変更することができます。

#### **\*NOSHOWSRC**

リストにソース・コードは表示しません。このオプションは、\*EXPMAC, \*SHOWINC, \*SHOWSYS,または\*SHOWUSRオプションで変更できます。

#### **\*NOSHOWSYS**

リストのソース・セクションで、またはデバッグ・ビューで#includeディレクティブ上のシステム組み込みファイルを拡張しません。システム組み込みファイルは、includeディレクティブの後に続けて不等号括弧(< >)で囲まれます。

#### **\*SHOWSYS**

リストのソース・セクションで、またはデバッグ・ビューで#includeディレクティブ上のシステム



組み込みファイルを拡張します。OUTPUTオプションか、\*ALL、\*SOURCE,または\*LISTのDBGVIEWパラメーター値を指定する必要があります。システム組み込みファイルは、#includeディレクティブの後に続けて不等号括弧(< >)で囲まれます。

#### **\*NOSHOWUSR**

リストまたはデバッグ・ビューでユーザー組み込みファイルは拡張しません。ユーザー組み込みファイルは、#includeディレクティブの後に続けて二重引用符(" ")で囲まれます。

#### **\*SHOWUSR**

リストのソース・セクションで、またはデバッグ・ビューで#includeディレクティブ上のユーザー組み込みファイルを拡張します。OUTPUTオプションか、\*ALL、\*SOURCE,または\*LISTのDBGVIEWパラメーター値を指定する必要があります。ユーザー組み込みファイルは、#includeディレクティブの後に続けて二重引用符(" ")で囲まれます。

#### **\*STDINC**

システム提供のヘッダー・ファイルがコンパイルのための検索パスに組み込まれます。

#### **\*NOSTDINC**

システム提供のヘッダー・ファイルはコンパイルのための検索パスに組み込まれません。

#### **\*NOSTDLOGMSG**

コンパイル・メッセージはSTDOUTストリームへ送られません。

#### **\*STDLOGMSG**

コンパイル・メッセージがSTDOUTストリームへ送られます。

#### **\*NOSTRUCREF**

参照されたすべてのSTRUCTまたはUNION変数のマップはリスト・ファイルの中に生成しません。

#### **\*STRUCREF**

参照されたすべてのSTRUCTまたはUNION変数のマップをリスト・ファイルの中に生成します。

#### **\*NOSYSINCPATH**

ユーザー組み込みファイルの検索パスは影響を受けません。

#### **\*SYSINCPATH**

ユーザー組み込みファイルの検索パスをシステム組み込みファイルの検索パスに変更します。機能的に、このオプションを使用することは、ユーザー#includeディレクティブ(#include "file\_name")の二重引用符を不等号括弧(#include <file\_name>)に変更することに相当します。

#### **\*NOXREF**

リストの中に相互参照テーブルは生成しません。

#### **\*XREF**

ソース・コード中の識別コードをそれらが表示される行の番号とともに示すリストを含む相互参照テーブルを生成します。OUTPUTオプションの指定が必要です。

#### **\*NOXREFREF**

リストの中で参照される識別コードの相互参照テーブルは生成しません。

#### **\*XREFREF**

リスト・ファイルで参照される変数、構造および関数名の相互参照テーブルを生成します。このテーブルには、識別コードが宣言された場所の行番号が示されます。OUTPUTオプションの指定が必要です。

上

---

## チェックアウト・オプション (CHECKOUT)

考えられるプログラミング・エラーを指示する通知メッセージの生成を選択できるオプションを指定します。1つのオプションを複数回指定したり、2つのオプションが競合する場合には、最後に指定されたものが使用されます。

このパラメーターには39個の値を指定できます。

注: CHECKOUTでは多くのメッセージが生成されることがあります。これらのメッセージがジョブ・ログへ送られるのを防止するには、OUTPUTパラメーターと一緒にOPTION(\*NOLOGMSG \*NOSTDLOGMSG)を指定して、そのチェックアウト・メッセージをリスト・ファイルに入れます。

### **\*NONE**

CHECKOUTのためのすべてのオプションを使用不可とします。

### **\*USAGE**

これは、\*ENUM, \*EXTERN, \*INIT, \*PARM, \*PORT, \*GENERAL,および\*TRUNCを指定するのと同様です。

**\*ALL** CHECKOUTのためのすべてのオプションを使用可能とします。

### **\*NOCOND**

条件式で考えられる冗長度または問題についての警告はしません。

### **\*COND**

条件式で考えられる冗長度または問題について警告します。

### **\*NOCONST**

定数と関係した演算についての警告はしません。

### **\*CONST**

定数と関係した演算について警告します。

### **\*NOEFFECT**

有効でないステートメントについての警告はしません。

### **\*EFFECT**

有効でないステートメントについて警告します。

### **\*NOENUM**

列挙型の使用はリストしません。

### **\*ENUM**

列挙型の使用をリストします。

### **\*NOEXTERN**

外部宣言を持つ未使用変数はリストしません。

### **\*EXTERN**

外部宣言を持つ未使用変数をリストします。

### **\*NOGENERAL**

一般チェックアウト・メッセージはリストしません。

### **\*GENERAL**

一般チェックアウト・メッセージをリストします。

### **\*NOGOTO**

GOTOステートメントの出現および使用はリストしません。

**\*GOTO**

GOTOステートメントの出現および使用をリストします。

**\*NOINIT**

明示的に初期化されない自動変数はリストしません。

**\*INIT** 明示的に初期化されない自動変数をリストします。

**\*NOPARM**

使用されない関数パラメーターはリストしません。

**\*PARM**

使用されない関数パラメーターをリストします。

**\*NOPORT**

C言語の移植不能使用はリストしません。

**\*PORT**

C言語の移植不能使用をリストします。

**\*NOPPCHECK**

プリプロセッサのディレクティブはリストしません。

**\*PPCHECK**

プリプロセッサのすべてのディレクティブをリストします。

**\*NOPPTRACE**

プリプロセッサによる組み込みファイルのトレースはリストしません。

**\*PPTRACE**

プリプロセッサによる組み込みファイルのトレースをリストします。

**\*NOREACH**

到達不能ステートメントについての警告はしません。

**\*REACH**

到達不能ステートメントについて警告します。

**\*NOTRUNC**

データの考えられる切り捨てまたは喪失についての警告はしません。

**\*TRUNC**

データの考えられる切り捨てまたは喪失について警告します。

**\*NOUNUSED**

未使用の自動または静的変数は検査しません。

**\*UNUSED**

未使用の自動または静的変数を検査します。

上

---

## 最適化 (OPTIMIZE)

生成されるオブジェクトの最適化のレベルを指定します。

**10** 生成されるコードは最適化されません。このレベルでのコンパイル時間は最短です。このレベルでは、デバッグ中に変数を表示して変更することができます。

- 20** コードについて一部の最適化が実行されます。このレベルでは、デバッグ中にユーザーが変数を表示することができますが、変更することはできません。
- 30** 生成されるコードについて完全な最適化が実行されます。デバッグ・セッション中は、ユーザー変数は変更することができませんが、表示することはできます。表示される値は、変数の現行値でない場合があります。
- 40** レベル30で実行されたすべての最適化が生成されるコードについて実行されます。さらに、コードは、命令トレースおよび呼び出しトレース・システム機能を使用可能にするプロシージャーのプロローグおよびエピローグ・ルーチンから除去されます。このコードを除去することによって、リーフ・プロシージャーの作成が可能になります。リーフ・プロシージャーとは、他のプロシージャーに対する呼び出しを含まないプロシージャーです。リーフ・プロシージャーに対するプロシージャー呼び出しのパフォーマンスは、通常のプロシージャーに対する呼び出しより高速となります。

上

---

## インライン・オプション (INLINE)

コンパイラーが呼び出し先関数の命令による関数呼び出しの置き換えを考慮する必要があるかどうかを指定します。関数をインライン化することによって、呼び出しのオーバーヘッドが取り除かれるので、結果としてより良好に最適化することができます。何度も呼び出される小さな関数は、インライン化の候補として適しています。

### 要素1: インライナー

インライン化を使用するかどうかを指定します。

**\*OFF** コンパイル単位にインライン化は実行されないことを指定します。

**\*ON** コンパイル単位にインライン化が実行されることを指定します。デバッグ・ビューが指定された場合には、インライン化はオフにされます。

### 要素2: モード

そのしきい値および限界値に基づいて、インライナーは関数のインライン化を自動的に試みる必要があるかどうかを指定します。

#### **\*AUTO**

インライナーは指定されたしきい値および限界値に基づいて関数をインライン化できるかどうかを判断する必要があることを指定します。**\*AUTO**は**#PRAGMA NOINLINE**ディレクティブによって上書きされます。

#### **\*NOAUTO**

**#PRAGMA INLINE**ディレクティブによって指定された関数のみをインライン化の候補として考慮する必要があることを指定します。

### 要素3: しきい値

自動インライン化の候補にできる関数の最大サイズを指定します。このサイズは、抽象コード単位(ACU)で測定されます。ACUのサイズは関数中の実行可能コードと比例します。ソース・コードは、コンパイラーによってACUに変換されます。

**250** 250のしきい値を指定します。

#### ***I-65535***

ACUのしきい値を指定します。

#### **\*NOLIMIT**

しきい値をプログラム・オブジェクトの最大サイズとして定義します。

#### **要素4: 限界**

自動インライン化を停止する前に関数を拡張できる最大相対サイズを指定します。

**2000** 2000 ACUの限界を指定します。

#### **\*NOLIMIT**

限界はプログラム・オブジェクトの最大サイズとして定義されます。システム限界が検出されることがあります。

#### ***I-65535***

ACUの限界を指定してください。

#### **要素5: 報告書**

コンパイラー・リストともにインライナー報告書を生成するかどうかを指定します。

**\*NO** インライナー報告書は生成されません。

**\*YES** インライナー報告書がコンパイラー・リストの一部として生成されます。インライナー報告書を生成するには、OUTPUTオプションの指定が必要です。

上

---

## **モジュール作成オプション (MODCRTOPT)**

モジュール・オブジェクトの作成時に使用するオプションを指定します。

#### **\*NOKEEPILDTA**

モジュール・オブジェクトと一緒に中間言語データは保管されません。

#### **\*KEEPILDTA**

モジュール・オブジェクトと一緒に中間言語データが保管されます。

上

---

## **デバッグ・ビュー (DBGVIEW)**

作成されたプログラムまたはサービス・プログラム・オブジェクトでモジュールに対して使用できるデバッグのレベルを指定します。ソース・レベルのデバッグ用にどのソース・ビューが使用可能かも指定されます。デバッグ・ビューを要求すると、インライン化はオフにされます。

#### **\*NONE**

デバッグ機能はモジュール・オブジェクトに挿入されません。

**\*ALL** すべてのデバッグ・オプション(\*STMT, \*SOURCE,および\*LIST)が使用可能になります。

#### **\*STMT**

プログラム・ステートメント番号および記号識別コードを使用したモジュール・オブジェクトのデバッグが可能になります。

注: \*STMTオプションを使用してモジュール・オブジェクトをデバッグするには、リストが必要です。

#### **\*SOURCE**

モジュール・オブジェクトのデバッグのためのソース・ビューを生成します。OPTIONパラメーターの値\*NOSHOWINC, \*SHOWINC, \*SHOWSYS, および\*SHOWUSRによって、作成されるソース・ビューの内容が決まります。

注: このビューをデバッグ用に使用するには、モジュールが作成された後に、ルート・ソース・ファイルを変更、名前変更、または移動しないようにしてください。

**\*LIST** モジュール・オブジェクトのデバッグ用のリスト・ビューを生成します。OPTIONパラメーターの値\*SHOWINC, \*SHOWUSR, \*SHOWSYS, および\*NOSHOWINCによって、作成されるリスト・ビューの内容が決まります。

上

---

## **名前の定義 (DEFINE)**

ファイルがコンパイラーによって処理される前に有効となるプリプロセッサ・マクロを指定します。形式DEFINE(MACRO)の使用はDEFINE('MACRO=1')と同等です。

### **単一値**

#### **\*NONE**

マクロは定義されません。

### **その他の値 (反復は最大32回まで)**

#### **'名前'または'名前=値'**

最大32個のマクロを定義できます。各マクロ名はアポストロフィで囲まれます。マクロ名の最大長は80文字です。アポストロフィは80の文字列の一部ではありません。アポストロフィは、大文字小文字を区別するマクロに必要です。

注: コマンドで定義されたマクロによって、ソースにある同じ名前のすべてのマクロ定義が上書きされますが、コンパイラーにより警告メッセージが生成されます。 #define max(a,b) ((a)>(b)?(a):(b))など関数と類似したマクロをコマンド行で定義することはできません。

上

---

## **言語レベル (LANGLVL)**

コンパイラーの機能と、ソースの作成時に宣言されるプロトタイプを指定します。

#### **\*EXTENDED**

プリプロセッサ変数\_\_EXTENDED\_\_を定義し、他の言語レベル変数は未定義とします。このパラメーターは、ILE Cのすべての機能が必要な場合に使用する必要があります。

**\*ANSI** プリプロセッサ変数\_\_ANSI\_\_および\_\_STDC\_\_を定義し、他の言語レベル変数は未定義とします。ANSI標準Cだけが使用可能とされます。

注: ILE Cコンパイラーは、常に\_\_ILEC400\_\_マクロを事前に定義します。

上

---

## 別名 (ALIAS)

作成中のモジュール・オブジェクトに適用される別名割り当てアサーションを指定します。

このパラメーターには3個の値を指定できます。

**\*ANSI** モジュール・オブジェクトで、ポインターが、同一タイプのオブジェクトのみを指し示すようになります。

**\*NOANSI**

モジュール・オブジェクトは、\*ANSI別名割り当て規則を使用しません。

**\*ADDRTAKEN**

モジュール・オブジェクトで、変数のアドレスが使用されない場合に、その変数のクラスがポインターから切り離されます。

**\*NOADDRTAKEN**

モジュール・オブジェクトは、\*ADDRTAKEN別名割り当て規則を使用しません。

**\*ALLPTRS**

モジュール・オブジェクトで、2つのポインターに別名を割り当てることができなくなります。

**\*NOALLPTRS**

モジュール・オブジェクトは、\*ALLPTRS別名割り当て規則を使用しません。

**\*TYPEPTR**

モジュール・オブジェクトで、タイプの異なる2つのポインターに別名を割り当てることができなくなります。

**\*NOTYPEPTR**

モジュール・オブジェクトは、\*TYPEPTR別名割り当て規則を使用しません。

上

---

## SYSTEMインターフェースOPT (SYSIFCOPT)

作成中のモジュール・オブジェクトに使用されるシステム・インターフェース・オプションを指定します。それらは、1つまたは複数のブランクで区切り、任意の順序で指定できます。1つのオプションが複数回指定されたり、2つのオプションが競合する場合には、最後に指定されたものが使用されます。

このパラメーターには2個の値を指定できます。

**\*NOIFSIO**

モジュール・オブジェクトは、Cストリーム入出力操作にデータ管理機能ファイル・システムを使用します。

**\*IFSIO**

モジュール・オブジェクトは、Cストリーム入出力操作に統合ファイル・システムを使用します。

**\*IFS64IO**

モジュール・オブジェクトは、64ビットCストリーム入出力操作に統合ファイル・システムを使用します。



### **\*NOASYNCSIGNAL**

同期信号関数の非同期信号関数への実行時マッピングを使用可能にしません。

### **\*ASYNCSIGNAL**

同期信号関数の非同期信号関数への実行時マッピングを使用可能にします。このオプションを指定すると、Cの実行時に同期SIGNAL()およびRAISE()関数がそれぞれ非同期SIGACTION()およびKILL()関数にマップされます。

上

---

## **LOCALEオブジェクト・タイプ (LOCALETYPE)**

作成中のモジュール・オブジェクトで使用するロケール・サポートのタイプを指定します。

### **\*LOCALE**

このオプションで作成されたモジュール・オブジェクトは、\*LOCALEオブジェクトによって指定されるロケール・サポートを使用します。

### **\*LOCALEUCS2**

このオプションで作成されたモジュール・オブジェクトは、\*LOCALEオブジェクトによって指定されるロケール・サポートを使用します。広幅(WIDE)文字タイプには2つの2バイト汎用文字セットの値が含まれることになります。

**\*CLD** このオプションで作成されたモジュール・オブジェクトは、\*CLDオブジェクトによって指定されるロケール・サポートを使用します。

### **\*LOCALEUTF**

このオプションで作成されたモジュール・オブジェクトは、\*LOCALEオブジェクトによって指定されるロケール・サポートを使用します。広幅(WIDE)文字タイプには、4バイトのUTF-32の値が含まれます。狭幅(NARROW)文字タイプには、UTF-8の値が含まれます。

上

---

## **メッセージのフラグ・レベル (FLAG)**

リストに表示するメッセージのレベルを指定します。

- 0** 通知レベルで始まるすべてのメッセージが表示されます。
- 10** 警告レベルで始まるすべてのメッセージが表示されます。
- 20** エラー・レベルで始まるすべてのメッセージが表示されます。
- 30** 重大エラー・レベルで始まるすべてのメッセージが表示されます。

上

---

## コンパイラー・メッセージ (MSGLMT)

指定されたメッセージ重大度(またはそれ以上)のメッセージの最大数を指定します。コンパイルが停止するまでに、このメッセージ数までメッセージを生成できます。

### 要素1: メッセージ限界

指定したメッセージ重大度レベル以上で生じさせることができるメッセージの最大数を指定します。

#### **\*NOMAX**

指定されたメッセージ重大度レベルで生じたメッセージの数とは無関係に、コンパイルが続行されます。

#### **0-32767**

生じさせることができるメッセージの数を指定します。

### 要素2: メッセージ限界の重大度

メッセージ重大度を指定します。ここで指定した重大度以上のメッセージが特定の数（要素1に指定された数）だけ生成されるとき、コンパイルを停止できます。

- 30**      コンパイルが停止するまでに、重大度30のメッセージを指定の数だけ生成できます。
- 0**        コンパイルが停止するまでに、重大度0以上のメッセージを指定の数だけ生成できます。
- 10**      コンパイルが停止するまでに、重大度10以上のメッセージを指定の数だけ生成できます。
- 20**      コンパイルが停止するまでに、重大度20以上のメッセージを指定の数だけ生成できます。

上

---

## MODULE OBJECTの置き換え (REPLACE)

オブジェクトの既存のバージョンを現行バージョンで置き換えるかどうかを指定します。

**\*YES**    既存のオブジェクトが新規バージョンで置き換えられます。旧バージョンはQRPLOBJライブラリーに移動し、システム日付および時刻に基づいて名前変更されます。置き換えられたオブジェクトのテキスト記述は、オリジナルのオブジェクトの名前に変更されます。旧オブジェクトは、それが明示的に削除されていない場合は、次のIPL時に削除されます。

**\*NO**     既存のオブジェクトは置き換えられません。指定のライブラリーで同じ名前のオブジェクトが見つかった場合、メッセージが表示されて、コンパイルが停止します。

上

---

## 権限 (AUT)

オブジェクトに対する特定権限がないユーザー、権限リスト上にないユーザー、またはそのグループにオブジェクトに対する特定権限がないユーザーに許可される権限を指定します。

#### **\*LIBCRTAUT**

オブジェクトに対する共通権限はターゲット・ライブラリー（作成されたオブジェクトが入れられるライブラリー）のCRTAUTキーワードからとられます。この値は、オブジェクトの作成時に決定されます。オブジェクトの作成後にライブラリーのCRTAUT値が変更された場合には、この新規の値はライブラリー中の既存のどのオブジェクトにも影響しません。

**\*ALL** 所有者に限定されるか、あるいは権限リスト管理権限によって制御される操作を除いて、オブジェクトでのすべての操作に対する権限が提供されます。どのユーザーも、オブジェクトの存在を制御し、その機密保護を指定し、変更し、また、その所有権の転送も含めてオブジェクトに対する基本操作を実行することができます。

**\*CHANGE**

すべてのデータ権限、および所有者に限定されるか、あるいは権限リスト管理権限によって制御される操作を除いて、オブジェクトでのすべての操作を実行する権限が提供されます。オブジェクトを変更し、それに基本操作を実行することができます。

**\*USE** オブジェクト操作権、読み取り権限、およびモジュール・オブジェクトのバインドなど、オブジェクトへの基本読み取り専用操作に対する権限が提供されます。特定権限のないユーザーは、オブジェクトを変更することができません。

**\*EXCLUDE**

特殊権限のないユーザーはオブジェクトにアクセスすることができません。

**name** オブジェクトの追加先となるユーザーと権限で構成された権限リストの名前を指定してください。オブジェクトはこの権限リストによって保護され、そのオブジェクトの共通権限は\*AUTLに設定されます。権限リストは、コマンドが出された時にシステム上に存在していなければなりません。

上

---

## ターゲット・リリース (TGTRLS)

ユーザーが作成中のオブジェクトを使用したいオペレーティング・システムのリリースを指定します。

\*CURRENTおよび\*PRVの値について示される例の中、および リリース・レベル の値を指定する場合には、形式VXRXXMXを使用してリリースが指定されます。ここで、VXはバージョン、RXはリリース、そしてMXはモディフィケーション・レベルです。たとえば、V4R5M0は、バージョン4、リリース5、モディフィケーション・レベル0です。

**\*CURRENT**

オブジェクトは、ユーザーのシステムで現在実行されているオペレーティング・システムのリリースで使用されます。例えば、V5R5M0がシステムで実行されている場合に、\*CURRENTは、ユーザーはオブジェクトをV5R5M0の導入されたシステムで使用したいことを意味します。ユーザーは、また、このリリース以降のオペレーティング・システムが導入されたシステムでもオブジェクトを使用することができます。

**\*PRV** オブジェクトは、オペレーティング・システムの前のリリース（モディフィケーション・レベル0）で使用されます。例えば、V5R5M0がシステムで実行されている場合に、\*PRVは、ユーザーはオブジェクトをV5R5M0の導入されたシステムで使用したいことを意味します。ユーザーは、また、このリリース以降のオペレーティング・システムが導入されたシステムでもオブジェクトを使用することができます。

**リリース・レベル**

リリースを形式VXRXXMXで指定します。オブジェクトは、指定したリリースのシステムでか、あるいはこのリリース以降のオペレーティング・システムが導入されたシステムで使用することができます。

有効な値は、現行バージョン、リリース、およびモディフィケーション・レベルによって異なります。これらは各新規リリースごとに変更されます。このコマンドによってサポートされる最も初期のリリース・レベルより前のリリース・レベルを指定した場合には、エラー・メッセージが送られます。

上

---

## パフォーマンス収集の使用可能化 (ENBPFRCOL)

オブジェクト内でパフォーマンス測定コードを生成すべきかどうかを指定します。収集されたデータは、システム・パフォーマンス・ツールによって使用し、アプリケーションのパフォーマンスのプロファイルを作成することができます。作成されたオブジェクト内でパフォーマンス測定コードを生成すると、オブジェクトのサイズがわずかに増えるため、パフォーマンスに影響が及ぶ可能性があります。

注: V5R5M0 以降では、このパラメーターは、作成されたオブジェクトに対して作用しなくなります。このパラメーターは、V5R5M0 より前のリリースとの互換性を保つために用意されています。

**\*PEP** パフォーマンス統計は、プログラム・エントリー・プロシージャーの入り口と出口でのみ収集されます。この値は、アプリケーションの全般的なパフォーマンス情報を収集したい場合に選択します。

### **\*ENTRYEXIT \*NONLEAF**

パフォーマンス統計は、プログラム・オブジェクトの全プロシージャー（リーフ・プロシージャーを除く）の入り口と出口で収集されます。これにはプログラムPEPルーチンが含まれます。

この選択項目は、アプリケーション内の他のルーチン呼び出すルーチンに関する情報だけを取り込みたい場合に有用です。

### **\*ENTRYEXIT \*ALLPRC**

パフォーマンス統計は、プログラム・オブジェクトの全プロシージャー（リーフ・プロシージャーを含む）の入り口と出口で収集されます。これには、プログラムPEPルーチンが含まれます。

この選択項目は、すべてのルーチンに関する情報を取り込みたい場合に有用です。このオプションは、ユーザーのアプリケーションから呼び出されるすべてのプログラム・オブジェクトが、\*PEP、\*ENTRYEXIT、\*FULLのいずれかのオプションを使用して作成されたことが判明している場合に使用します。それ以外の場合、パフォーマンス測定に対して使用可能にならない他のプログラム・オブジェクトがユーザーのアプリケーションから呼び出されると、パフォーマンス・ツールは、そのプログラム・オブジェクトによる資源の使用についてユーザーのアプリケーションに任せます。このため、資源が実際に使用されている場所を判別するのが困難になることがあります。

### **\*FULL \*NONLEAF**

パフォーマンス統計は、リーフ・プロシージャーではないすべてのプロシージャーの入り口と出口で収集されます。また、外部プロシージャーに対する個々の呼び出しの前および後にも統計が収集されます。

### **\*FULL \*ALLPRC**

パフォーマンス統計は、リーフ・プロシージャーを含むすべてのプロシージャーの入り口と出口で収集されます。また、外部プロシージャーに対する個々の呼び出しの前および後にも統計が収集されます。

このオプションは、作成時に\*PEP、\*ENTRYEXIT、\*FULLのいずれも使用されなかった他のプログラム・オブジェクトをユーザーのアプリケーションから呼び出そうと考えている場合に使用します。このオプションを使用すると、パフォーマンス・ツールは、ユーザーのアプリケーションによ

って使用されている資源と、そのアプリケーションから呼び出されたプログラム・オブジェクトによって使用されている資源を、（そのプログラム・オブジェクトがパフォーマンス測定に対して使用可能にならない場合でも）区別できるようになります。このオプションを使用すると、効率は最も悪くなりますが、アプリケーション内の各種プログラム・オブジェクトを選択的に分析できるようになります。

上

---

## パフォーマンス・オプション (PFROPT)

パフォーマンスを高めるために使用可能な各種のオプションを指定します。それらは、1つまたは複数のブランクで区切り、任意の順序で指定できます。1つのオプションが複数回指定されたり、2つのオプションが競合する場合には、最後に指定されたものが使用されます。

このパラメーターには2個の値を指定できます。

### **\*SETFPCA**

浮動小数点計算に対するANSIのセマンティクスを達成するために、コンパイラーに浮動小数点計算属性を設定させます。

### **\*NOSETFPCA**

計算属性は設定されません。このオプションは、作成中のオブジェクトに浮動小数点計算が含まれない場合にのみ使用してください。

### **\*NOSTRONLY**

コンパイラーは文字列を書き込み可能メモリーに入れる必要があることを指定します。

### **\*STRONLY**

コンパイラーは文字列を読み取り専用メモリーに入れることができることを指定します。

上

---

## プロファイル作成データ (PRFDTA)

モジュール・オブジェクトのプログラム・プロファイル作成データ属性を指定します。プログラム・プロファイル作成は、プロシージャーおよびプロシージャー内のコードを統計データ（プロファイル作成データ）に基づいて配列し直すために使用される拡張最適化手法です。

### **\*NOCOL**

モジュール・オブジェクトは、プロファイル作成データの収集に使用できません。

**\*COL** モジュール・オブジェクトは、プロファイル作成データの収集に使用できます。**\*COL**は、最適化レベルが30以上である場合にのみ指定することができます。

上

---

## テラスペース・オプション (TERASPACE)

テラスペース・ストレージの処理にモジュール・オブジェクトを使用可能にするかどうかを指定します。これには、モジュール・オブジェクトによって割り振られるテラスペース・ストレージや、他のテラスペース対応プログラムおよびサービス・プログラムのオブジェクトから渡されたパラメーターによって割り振られるテラスペース・ストレージが含まれます。

## 単一値

**\*NO** モジュール・オブジェクトは、テラスペースから割り振られたストレージのアドレス指定の処理に使用できません。

注: V5R5M0以降では、テラスペースから割り振られたストレージのアドレス指定の処理に、すべてのモジュールが使用できます。ただし、\*NOが指定された場合には、以下にリストされているコンパイラー機能が使用できなくなります。

### 要素1: テラスペース使用可能

**\*YES** モジュール・オブジェクトは、他のテラスペース対応プログラムやサービス・プログラムのオブジェクトから渡されたパラメーターも含めて、テラスペースから割り振られたストレージのアドレス指定の処理に使用できます。さらに、以下のコンパイラー機能も使用できます。

- テラスペース・ストレージへのアクセスに使用される8バイト・ポインターを作成できるように、ポインターは\_\_ptr64で修飾できます。
- テラスペース・ストレージ・モデルは、STGMDL(\*TERASPACE)コンパイラー・オプションを使用して指定できます。
- LLP64データ・モデルは、DTAMDLL(\*LLP64)コンパイラー・オプションまたは#pragma datamodel(llp64)ディレクティブを使用して指定できます。
- ポインター差分演算は、ptrdiff\_tの結果ではなく、signed long longの結果を返します。

### 要素2: TERASPACE INTERFACESの使用

#### **\*NOTSIFC**

モジュール・オブジェクトは、デフォルトで、ストレージ機能の非テラスペース・バージョンを使用するようになります。

#### **\*TSIFC**

モジュール・オブジェクトは、デフォルトで、ストレージ機能のテラスペース・バージョンを使用するようになります。コンパイラーは、マクロ変数\_\_TERASPACE\_\_を定義することになります。

上

---

## ストレージ・モデル (STGMDL)

作成されたオブジェクトで使用する記憶域のタイプを指定します。

#### **\*SNGLV**

作成されたオブジェクトは、単一レベルの記憶域を使用します。

#### **\*TERASPACE**

作成されたオブジェクトは、テラスペース記憶域を使用します。

#### **\*INHERIT**

作成されたオブジェクトは、単一レベルとテラスペースのどちらの記憶域も使用することができます。使用される記憶域のタイプは、呼び出し元に必要な記憶域のタイプによって異なります。

上

---

## データ・モデル (DTAMDLL)

INT, LONG, POINTERとして宣言される変数のサイズ（バイト数）を指定します。



**\*P128** INT, LONG, POINTERのサイズがそれぞれ4, 4, 16となるようにします。

**\*LLP64**

INT, LONG, POINTERのサイズがそれぞれ4, 4, 8となるようにします。コンパイラーは、マクロ `__LLP64_IFC__` を定義することになります。

上

---

## パック構造 (PACKSTRUCT)

構造のメンバーに使用する位置合わせ境界を指定します。

**\*NATURAL**

構造メンバーはその自然境界で位置合わせされます。たとえば、短形式整数は2バイトで位置合わせされます。16バイト・ポインターは、常に16バイト境界で位置合わせされます。

- 1 構造メンバーを1バイトの位置合わせでパックします。
- 2 構造メンバーを2バイトの位置合わせでパックします。
- 4 構造メンバーを4バイトの位置合わせでパックします。
- 8 構造メンバーを8バイトの位置合わせでパックします。
- 16 構造メンバーを16バイトの位置合わせでパックします。

上

---

## Enumサイズ (ENUM)

コンパイラーが列挙型を表すために使用するバイト数を指定します。

**\*SMALL**

すべてのENUM変数を、値の範囲を表すことができる最も小さいサイズにします。

- 1 すべてのENUM変数を1バイトにします。
- 2 すべてのENUM変数を2バイトにします。
- 4 すべてのENUM変数を4バイトにします。

**\*INT** ANSI標準のENUMサイズである4バイトを使用します。

上

---

## 依存関係情報 (MAKEDEP)

依存関係情報をファイルの中に生成するかどうかを指定します。この情報は、MAKE TOOLによって使用されます。

**\*NONE**

依存関係情報は生成しません。

***path-name***

依存関係情報を保管するストリーム・ファイルのパス名を指定します。

上



---

## プリプロセッサ・オプション (PPGENOPT)

ソース・コードのコンパイル時に使用するプリプロセッサ生成オプションを指定します。

### 単一値

#### \*NONE

ソース・ファイルに対してコンパイラ全体を実行します。プリプロセッサの出力をファイルにはコピーしません。

**\*DFT** 入力ソースに対してプリプロセッサを実行します。\*RMVCOMMENTおよび\*GENLINEは、プリプロセッサの出力を生成するためのオプションとして使用されます。出力ソース・ファイルおよびメンバーを指定するにはPPSRCFILEおよびPPSRCMBRを使用し、あるいはプリプロセッサの出力を入れるストリーム・ファイルを指定するにはPPSRCSTMFを使用します。

その他の値（反復は最大2回まで）

#### **\*RMVCOMMENT**

前処理中に、注記を除去します。

#### **\*NORMVCOMMENT**

前処理中に、注記を保存します。

#### **\*GENLINE**

プリプロセッサ出力の中に#LINEディレクティブを生成します。

#### **\*NOGENLINE**

プリプロセッサの出力から#LINEディレクティブを抑制します。

上

---

## 出力ソース・ファイル (PPSRCFILE)

プリプロセッサ出力用の物理ファイルを指定します。

### 修飾子1: 出力ソース・ファイル

**name** プリプロセッサの出力のための物理ファイルの名前を指定します。

### 修飾子2: ライブラリー

#### \*CURLIB

プリプロセッサの出力は現行ライブラリーに作成されます。ジョブに現行ライブラリーがない場合には、プリプロセッサの出力ファイルはQGPLライブラリーの中に作成されます。

**name** プリプロセッサの出力のためのライブラリーの名前を指定します。

上

---

## 出力ソース・メンバー (PPSRCMBR)

プリプロセッサ出力用の物理ファイル・メンバーを指定します。

### **\*MODULE**

このモジュール (**MODULE**)パラメーターに指定された名前が、プリプロセッサ出力メンバー名として使用されます。

**name** プリプロセッサの出力のためのメンバーの名前を指定します。

上

---

## **出力ストリーム・ファイル (PPSRCSTMF)**

プリプロセッサの出力のためのストリーム・ファイルのパス名を指定します。

### **\*SRCSTMF**

ソース・ストリーム・ファイル (**SRCSTMF**)パラメーターに指定されたパス名が、プリプロセッサ出力パス名として使用されます。このファイルには拡張子'.I'があります。

### ***path-name***

プリプロセッサの出力ストリーム・ファイルのためのパス名を指定します。

上

---

## **組み込みディレクトリー (INCDIR)**

組み込みファイルを検出するためにコンパイラーが使用する検索パスに追加する1つまたは複数のディレクトリーを指定します。INCDIRを使用すると、INCLUDE環境変数が上書きされます。

検索パスは、OPTIONキーワードで次のパラメーターを使用することにより、さらに変更することができます。

- \*INCDIRFIRSTまたは\*NOINCDIRFIRST
- \*SYSINCPATHまたは\*NOSYSINCPATH
- \*STDINCまたは\*NOSTDINC

### **\*NONE**

変更されなければ、省略時のシステム組み込みディレクトリーおよびソース・ディレクトリーからユーザー組み込みファイルが検索されます。

### ***path-name***

組み込みファイルの検索場所となるディレクトリーのパス名を指定してください。最大32個のディレクトリーを指定できます。指定したディレクトリーに加えて、ソース・ディレクトリーからもユーザー組み込みファイルが検索されます。

上

---

## **コンパイラー・サービスOPT (CSOPT)**

1つまたは複数のコンパイラー・サービス・オプションを指定します。このパラメーターにより、IBMはリリース相互間で切り替え可能なコンパイラーを提供することができます。

### **\*NONE**

コンパイラー・サービス・オプションは選択されません。

**文字値** モジュール・オブジェクトの作成時に使用するコンパイラー・サービス・オプションを含む文字ストリングを指定してください。有効な文字列は、PTFカバー・レターまたはリリース情報に記述されます。

上

---

## ライセンス内部コードOPT (LICOPT)

1つまたは複数のライセンス内部コード・コンパイル時オプションを指定します。このパラメーターは、個別のコンパイル時オプションを選択できるようにするもので、選択された各タイプのコンパイラー・オプションの潜在的な長所と短所を理解している上級プログラマーを対象としています。

### **\*NONE**

コンパイル時オプションは選択されません。

**文字値** 選択されたライセンス内部コード・コンパイル時オプションが、モジュール・オブジェクトの作成時に使用されます。特定のオプションでは、作成されたモジュール・オブジェクトをデバッグする能力が低下することがあります。

上

---

## 省略時の文字タイプ (DFTCHAR)

CHARデータ・タイプの省略時の符号を指定します。

### **\*UNSIGNED**

省略時のCHARタイプを符号なしとします。

### **\*SIGNED**

省略時のCHARタイプを符号付きとします。

上

---

## ターゲットCCSID (TGTCSSID)

結果のモジュール・オブジェクトに保管されるデータを記述するために使用されるターゲット・コード化文字セットIDを指定します。

### **\*SOURCE**

ルート・ソース・ファイルのCCSIDが使用されます。

**\*JOB** 現行ジョブのCCSIDが使用されます。

**\*HEX** CCSID 65535が使用されます。これは、文字データはバイナリー・データとして取り扱われ、変換されないことを指示します。

### **I-65535**

使用するCCSIDを指定します。

上

---

## 10進浮動丸めモード (DECFLTRND)

定数10進浮動小数点式の評価のコンパイル時丸めモードを指定します。このオプションは、実行時10進浮動小数点丸めモード(setca組み込み関数を使用して設定される)には影響しません。

### \*HALFEVEN

最も近い値に丸めます。最も近い値が2つある場合、偶数を選択します。たとえば、5.22の場合は5.2に丸め、5.67の場合は5.7に丸め、5.55の場合は5.6に丸め、5.65の場合は5.6に丸め、-5.23の場合は-5.2に丸め、-5.55の場合は-5.6に丸めます。

### \*DOWN

ゼロに近い方の数字に丸めるか、または結果の数字を切り捨てます。たとえば、5.22の場合は5.2に丸め、5.67の場合は5.6に丸め、5.55の場合は5.5に丸め、5.65の場合は5.6に丸め、-5.23の場合は-5.2に丸め、-5.55の場合は-5.5に丸めます。

**\*UP** ゼロから遠い方の数字に丸めます。たとえば、5.22の場合は5.3に丸め、5.67の場合は5.7に丸め、5.55の場合は5.6に丸め、5.65の場合は5.7に丸め、-5.23の場合は-5.3に丸め、-5.55の場合は-5.6に丸めます。

### \*HALFUP

最も近い値に丸めます。最も近い値が2つある場合、ゼロから遠い方の数字に丸めます。たとえば、5.22の場合は5.2に丸め、5.67の場合は5.7に丸め、5.55の場合は5.6に丸め、5.65の場合は5.7に丸め、-5.23の場合は-5.2に丸め、-5.55の場合は-5.6に丸めます。

### \*HALFDOWN

最も近い値に丸めます。最も近い値が2つある場合、ゼロから近い方の数字に丸めます。たとえば、5.22の場合は5.2に丸め、5.67の場合は5.7に丸め、5.55の場合は5.5に丸め、5.65の場合は5.6に丸め、-5.23の場合は-5.2に丸め、-5.55の場合は-5.5に丸めます。

### \*FLOOR

小さい数字になるように丸めます。たとえば、5.22の場合は5.2に丸め、5.67の場合は5.6に丸め、5.55の場合は5.5に丸め、5.65の場合は5.6に丸め、-5.23の場合は-5.3に丸め、-5.55の場合は-5.6に丸めます。

### \*CEILING

大きい数字になるように丸めます。たとえば、5.22の場合は5.3に丸め、5.67の場合は5.7に丸め、5.55の場合は5.6に丸め、5.65の場合は5.7に丸め、-5.23の場合は-5.2に丸め、-5.55の場合は-5.5に丸めます。

上

---

## 例

### 例1:ソース・ファイル・メンバーからのモジュールの作成

```
CRTCMOD  MODULE(QTEMP/TEST) SRCFILE(QTEMP/QCSRC)
```

このコマンドは、モジュール・オブジェクトTESTをライブラリーQTEMP内に作成します。Cソースは、ライブラリーQTEMPにあるソース・ファイルQCSRCのメンバーTEST内に指定されます。

### 例2:ストリーム・ファイルからのモジュールの作成

```
CRTCMOD  MODULE(QTEMP/TEST) SRCSTMF('/tmp/mySrc.c')
```

このコマンドは、モジュール・オブジェクトTESTをライブラリーQTEMP内に作成します。Cソースは、ストリーム・ファイル/tmp/mySrc.c内に指定されます。

### 例3:コンパイル・リストのコンパイルおよび生成

```
CRTCMOD  MODULE(TEST)  OUTPUT(*PRINT)  OPTION(*XREF)
```

このコマンドは、モジュール・オブジェクトTESTをジョブの現行ライブラリー内に作成しますが、現行ライブラリーが定義されていない場合は、ライブラリーQGPL内に作成します。Cソースは、ソース・ファイルQCSRCのメンバーTEST内に指定されると想定されます（これは、ライブラリー・リストを使用して指定されます）。コンパイラ・リストが入ったスプール・ファイルが生成されます。コンパイラ・リスト・ファイルには、相互参照リストが格納されます。

### 例4:すべてのデバッグ・ビューを備えたモジュールの作成

```
CRTCMOD  MODULE(QTEMP/TEST)  SRCFILE(QTEMP/QCSRC)
          SRCMBR(TESTER)  DBGVIEW(*ALL)
```

このコマンドは、デバッグ時に使用されるステートメント、ソース、およびリストの各ビューを備えたモジュール・オブジェクトTESTをライブラリーQTEMP内に作成します。Cソースは、ライブラリーQTEMPにあるソース・ファイルQCSRCのメンバーTESTER内に指定されます。

### 例5:テラスペース対応モジュールの作成

```
CRTCMOD  MODULE(QTEMP/TEST)  SRCFILE(QTEMP/QCSRC)
          TERASPACE(*YES *TSIFC)  STGMDL(*TERASPACE)
          DTAMD(*LLP64)
```

このコマンドは、モジュール・オブジェクトTESTをライブラリーQTEMP内に作成します。Cソースは、ライブラリーQTEMPにあるソース・ファイルQCSRCのメンバーTEST内に指定されます。このモジュールは、テラスペース・ストレージを使用する場合に使用可能にされます。ヒープ・スペース、自動ストレージ、および静的ストレージは、テラスペース・ストレージ内に入れられます。デフォルト・ポインター・サイズは8バイトになります。

### 例6:ファイルへの前処理済みソースの保管

```
CRTCMOD  MODULE(TEST)  SRCFILE(QTEMP/QCSRC)
          PPGENOPT(*DFT)  PPSRCFILE(QTEMP/QCSRC)
          PPSRCMBR(TESTI)
```

このコマンドは、ライブラリーQTEMPにあるソース・ファイルQCSRCのメンバーTEST内に指定されているCソースに対してコンパイラ・プリプロセッサを実行し、同じソース・ファイルのメンバーTESTI内に出力ソースを保管します。モジュール・オブジェクトは作成されません。

上

---

## エラー・メッセージ

### \*ESCAPE メッセージ

#### CZM0613

コンパイルに失敗しました。

上



## C++モジュール作成 (CRTCPPMOD)

実行可能場所: すべての環境 (\*ALL)

スレッド・セーフ: いいえ

パラメーター

例

エラー・メッセージ

C++モジュール作成(CRTCPPMOD)コマンドはILE C++コンパイラーを開始します。 このコンパイラーは、ソース・コード内のILE C++ステートメントに基づいてモジュール・オブジェクトを作成しようとします。

上

### パラメーター

キーワード	記述	選択項目	ノーツ
MODULE	モジュール	修飾オブジェクト名	必須, 定位置 1
	修飾子 1: モジュール	名前	
	修飾子 2: ライブラリー	名前, *CURLIB	
SRCFILE	ソース・ファイル	修飾オブジェクト名	オプション, 定位置 2
	修飾子 1: ソース・ファイル	名前, QCPPSRC	
	修飾子 2: ライブラリー	名前, *LIBL, *CURLIB	
SRCMBR	ソース・メンバー	名前, *MODULE	オプション, 定位置 3
SRCSTMF	ソース・ストリーム・ファイル	パス名	オプション
TEXT	テキスト・記述	文字値, *SRCMBRTXT, *BLANK	オプション
OUTPUT	出力オプション	単一値: *NONE, 'トナトオ' その他の値: 要素リスト	オプション
	要素 1: 出力ファイル名	パス名, *PRINT, 'ニネケトホ'	
	要素 2: タイトル	文字値, *BLANK	
	要素 3: サブタイトル	文字値, *BLANK	



キーワード	記述	選択項目	ノーツ
<b>OPTION</b>	コンパイラー・オプション	値 (最大 35 回の繰り返し): *NOBITSIGN, *BITSIGN, *NOEVENTF, *EVENTF, *NOEXPMAC, *EXPMAC, *NOFULL, *FULL, *GEN, *NOGEN, *NOINCDIRFIRST, *INCDIRFIRST, *LOGMSG, *NOLOGMSG, *LONGLONG, *NOLONGLONG, *NORTTI, *RTTIAL, *RTTITYPE, *RTTICAST, *NOSHOWINC, *SHOWINC, *SHOWSRC, *NOSHOWSRC, *NOSHOWSYS, *SHOWSYS, *NOSHOWUSR, *SHOWUSR, *STDINC, *NOSTDINC, *NOSTDLOGMSG, *STDLOGMSG, *NOSYSINCPATH, *SYSINCPATH, *NOXREF, *XREF, *NOXREFREF, *XREFREF	オプション
<b>CHECKOUT</b>	チェックアウト・オプション	値 (最大 45 回の繰り返し): <u>*NONE</u> , *USAGE, *ALL, *NOCLASS, *CLASS, *NOCOND, *COND, *NOEFFECT, *EFFECT, *NOGENERAL, *GENERAL, *NOLANG, *LANG, *NOPARM, *PARM, *NOPORT, *PORT, *NOREACH, *REACH, *NOTEMP, *TEMP, *NOTRUNC, *TRUNC, *NOUNUSED, *UNUSED	オプション
<b>OPTIMIZE</b>	最適化	<u>10</u> , 20, 30, 40	オプション
<b>INLINE</b>	インライン・オプション	要素リスト	オプション
	要素 1: インライナー	<u>*OFF</u> , *ON	
	要素 2: モード	<u>*AUTO</u> , *NOAUTO	
	要素 3: しきい値	1-65535, <u>250</u> , *NOLIMIT	
	要素 4: 限界	1-65535, <u>2000</u> , *NOLIMIT	
	要素 5: 報告書	<u>*NO</u> , *YES	
<b>MODCRTOPT</b>	モジュール作成オプション	<u>*NOKEEPILDTA</u> , *KEEPILDTA	オプション
<b>DBGVIEW</b>	デバッグ・ビュー	<u>*NONE</u> , *ALL, *STMT, *SOURCE, *LIST	オプション
<b>DEFINE</b>	名前の定義	単一値: <u>*NONE</u> その他の値 (最大 32 回の繰り返し): 文字値	オプション
<b>LANGLVL</b>	言語レベル	<u>*EXTENDED</u> , *ANSI, *LEGACY	オプション

キーワード	記述	選択項目	ノート
ALIAS	別名	値 (最大 3 回の繰り返し): <u>*ANSI</u> , *NOANSI, *ADDRTAKEN, *NOADDRTAKEN, *ALLPTRS, *NOALLPTRS, *TYPEPTR, *NOTYPEPTR	オプション
SYSIFCOPT	SYSTEMインターフェースOPT	<u>*IFS64IO</u> , *IFSIO, *NOIFSIO	オプション
LOCALETYPE	LOCALEオブジェクト・タイプ	<u>*LOCALE</u> , *LOCALEUCS2, *LOCALEUTF	オプション
FLAG	メッセージのフラグ・レベル	<u>0</u> , 10, 20, 30	オプション
MSGMT	コンパイラ・メッセージ	要素リスト	オプション
	要素 1: メッセージ限界	0-32767, <u>*NOMAX</u>	
	要素 2: メッセージ限界の重大度	0, 10, 20, <u>30</u>	
REPLACE	MODULE OBJECTの置き換え	<u>*YES</u> , *NO	オプション
AUT	権限	名前, <u>*LIBCRTAUT</u> , *ALL, *CHANGE, *USE, *EXCLUDE	オプション
TGTRLS	ターゲット・リリース	単純名, <u>*CURRENT</u> , *PRV	オプション
ENBPFCOL	パフォーマンス収集の使用可能	要素リスト	オプション
	要素 1: 収集レベル	<u>*PEP</u> , *ENTRYEXIT, *FULL	
	要素 2: プロシージャ	*NONLEAF, *ALLPRC	
PFOPT	パフォーマンス・オプション	値 (最大 2 回の繰り返し): *SETFPCA, *NOSETFPCA, *NOSTRDONLY, *STRDONLY	オプション
PRFDA	プロファイル作成データ	<u>*NOCOL</u> , *COL	オプション
TERASPACE	テラスペース・オプション	単一値: <u>*NO</u> その他の値: 要素リスト	オプション
	要素 1: テラスペース使用可能	<u>*YES</u>	
	要素 2: TERASPACE INTERFACESの使用	<u>*NOTSIFC</u> , *TSIFC	
STGMDL	ストレージ・モデル	<u>*SNGLVL</u> , *TERASPACE, *INHERIT	オプション
DTAMD	データ・モデル	<u>*P128</u> , *LLP64	オプション
RTBND	ランタイム・バインディング	<u>*DEFAULT</u> , *LLP64	オプション
PACKSTRUCT	パック構造	<u>*NATURAL</u> , 1, 2, 4, 8, 16	オプション
ENUM	Enumサイズ	<u>*SMALL</u> , 1, 2, 4, *INT	オプション
MAKEDEP	依存関係情報	パス名, <u>*NONE</u>	オプション
PPGENOPT	プリプロセッサ・オプション	単一値: <u>*NONE</u> , *DFT その他の値 (最大 2 回の繰り返し): *RMVCOMMENT, *NORMVCOMMENT, *GENLINE, *NOGENLINE	オプション
PPSRCFILE	出力ソース・ファイル	修飾オブジェクト名	オプション
	修飾子 1: 出力ソース・ファイル	名前	
	修飾子 2: ライブラリー	名前, <u>*CURLIB</u>	
PPSRCMBR	出力ソース・メンバー	名前, <u>*MODULE</u>	オプション
PPSRCSTMF	出力ストリーム・ファイル	パス名, *SRCSTMF	オプション

キーワード	記述	選択項目	ノート
INCDIR	組み込みディレクトリー	単一値: <u>*NONE</u> その他の値 (最大 32 回の繰り返し): パス名	オプション
CSOPT	コンパイラ・サービスOPT	文字値, <u>*NONE</u>	オプション
LICOPT	ライセンス内部コードOPT	文字値, <u>*NONE</u>	オプション
DFTCHAR	省略時の文字タイプ	<u>*UNSIGNED</u> , *SIGNED	オプション
TGTCCSID	ターゲットCCSID	1-65535, <u>*SOURCE</u> , *JOB, *HEX	オプション
TEMPLATE	テンプレート・オプション	要素リスト	オプション
	要素 1: 一時組み込みディレクトリー	パス名, <u>*NONE</u> , *TEMPINC	
	要素 2: 生成済み最大見出し数	1-99999, <u>1</u>	
	要素 3: テンプレートの妥当性検査	<u>*NO</u> , *WARN, *ERROR	
TMPLREG	テンプレート・レジストリー	パス名, *DFT, <u>*NONE</u>	オプション
WEAKTMPL	Weakテンプレート定義	<u>*YES</u> , *NO	オプション

上

## モジュール (MODULE)

作成するモジュール・オブジェクトを指定します。

これは必須パラメーターです。

### 修飾子1: モジュール

**name** モジュール・オブジェクトの名前を指定してください。

### 修飾子2: ライブラリー

#### \*CURLIB

モジュール・オブジェクトは現行ライブラリーに保管されます。ジョブに現行ライブラリーがない場合には、モジュール・オブジェクトはQGPLライブラリーの中に作成されます。

**name** モジュール・オブジェクトの保管先となるライブラリーの名前を指定してください。

上

## ソース・ファイル (SRCFILE)

コンパイルが必要なILE C++ソース・コードが入っているソース・ファイルを指定します。

### 修飾子1: ソース・ファイル

#### QCPPSRC

QCPPSRCという名前のソース・ファイルには、コンパイルしたいILE C++ソース・コードを含むメンバーが入っています。

**name** ILE C++ソース・コードが含まれたメンバーが入っているソース・ファイルの名前を指定してください。

## 修飾子2: ライブラリー

**\*LIBL** ソース・ファイルが入っているライブラリーを見つけるためにライブラリー・リストが検索されます。

### **\*CURLIB**

ソース・ファイルが現行ライブラリーから検索されます。ジョブに現行ライブラリーがない場合には、ソース・ファイルはQGPLから検索されます。

**name** ソース・ファイルが入っているライブラリーの名前を指定してください。

上

---

## ソース・メンバー (SRCMBR)

コンパイルされるソース・コードが入っているソース・ファイル・メンバーを指定します。

### **\*MODULE**

モジュール (MODULE)パラメーターに指定された名前は、ソース・メンバー名として使用されます。

**name** ソース・コードが入っているメンバーの名前を指定してください。

上

---

## ソース・ストリーム・ファイル (SRCSTMF)

コンパイルが必要なソース・コードが入っているストリーム・ファイルを指定します。

パス名は絶対修飾パス名か相対修飾パス名のいずれかとすることができます。絶対パス名は'/'で始まり、相対パス名は'/'以外の文字で始まります。絶対修飾した場合には、そのパス名で完全です。相対修飾した場合には、パス名は、ジョブの現行作業ディレクトリーをパス名に対して事前保留することによって完全なものとなります。

SRCMBRおよびSRCFILEパラメーターをSRCSTMFパラメーターと一緒に指定することはできません。

### **path-name**

C++ソース・コードが入っているストリーム・ファイルのパス名を指定してください。

上

---

## テキスト'記述' (TEXT)

モジュール・オブジェクトを簡潔に記述するテキストを指定します。

### **\*SRCMBRTXT**

モジュール・オブジェクトに使用されるソース・ファイル・メンバーと関連したテキスト記述。ソース・ファイルがインライン・ファイル、ストリーム・ファイル、または装置ファイルである場合には、テキストはブランクとなります。

### **\*BLANK**

テキストは表示されないことを指定します。

**文字値** 50文字を超えないテキストをアポストロフィで囲んで指定します。

---

## 出力オプション (OUTPUT)

コンパイラー・リストが生成されるかどうかを指定します。

### 単一値

#### \*NONE

コンパイラー・リストは生成しません。 リストが必要でない場合には、コンパイル時のパフォーマンスを改善するためにこのパラメーター値を使用する必要があります。 \*NONEを指定した場合には、OPTIONパラメーターに指定されたリストと関連したすべてのパラメーター値が無視されます。

### 要素1: 出力ファイル名

#### \*PRINT

リストを含むスプール・ファイルが生成されます。

#### *path-name*

リストを保留するストリーム・ファイルのパス名を指定します。

### 要素2: タイトル

#### \*BLANK

テキストは表示されないことを指定します。

**文字値** リスト・ファイルのタイトル文字列（最大80文字）を指定します。

### 要素3: サブタイトル

#### \*BLANK

テキストは表示されないことを指定します。

**文字値** リスト・ファイルのサブタイトル文字列（最大80文字）を指定します。

---

## コンパイラー・オプション (OPTION)

ILE C++ソース・コードのコンパイル時に使用するオプションを指定します。それらは、1つまたは複数のブランクで区切り、任意の順序で指定できます。1つのオプションが複数回指定されたり、2つのオプションが競合する場合には、最後に指定されたものが使用されます。

このパラメーターには35個の値を指定できます。

このパラメーターには単一のデフォルト値はありません。以下に示す一連のパラメーター値の説明では、下線付きの値は、コンパイラーで使用するデフォルト値を表します。

#### \*NOBITSIGN

ビット・フィールドを符号なしとして指定します。

#### \*BITSIGN

ビット・フィールドを符号付きとして指定します。

#### **\*NOEVENTF**

CoOperative Development Environment/400 (CODE/400)で使用するためのイベント・ファイルは作成しません。

#### **\*EVENTF**

CoOperative Development Environment/400 (CODE/400)で使用するためのイベント・ファイルを作成します。イベント・ファイルは、作成中のモジュールまたはプログラム・オブジェクトの保管先のライブラリーにあるファイルEVFEVENTにメンバーとして作成されます。ファイルEVFEVENTが存在しない場合には、それが自動的に作成されます。イベント・ファイル・メンバー名は作成中のオブジェクトの名前と同じです。通常、イベント・ファイルは、CODE/400内からモジュールまたはプログラム・オブジェクトを作成するときに生成されます。CODE/400はこのファイルを使用して、CODE/400エディターに統合されたエラー・フィードバックを提供します。

#### **\*NOEXPMAC**

マクロの中で構文エラーが検出されないかぎり、マクロはリスト中で拡張されません。

#### **\*EXPMAC**

リスト中ですべてのマクロを拡張します。

#### **\*NOFULL**

すべてのリスト・オプションをオンにしません。

#### **\*FULL**

すべてのリスト・オプションをオンにします。

**\*GEN** コンパイル・プロセスのすべてのフェーズが実行されます。

#### **\*NOGEN**

コンパイルは構文検査の後に停止します。モジュール・オブジェクトは作成されません。

#### **\*NOINCDIRFIRST**

INCDIRパラメーターとして指定された組み込みディレクトリーは、標準ヘッダー・ファイル組み込みパスの前には組み込まれません。

#### **\*INCDIRFIRST**

INCDIRパラメーターとして指定された組み込みディレクトリーが、標準ヘッダー・ファイル組み込みパスの前に組み込まれます。

#### **\*LOGMSG**

コンパイル・メッセージをジョブ・ログに書き込みます。

このオプションとFLAGパラメーターを指定した場合には、FLAGパラメーターに指定された（およびそれ以上の）重大度のメッセージがジョブ・ログに入れます。

このオプションとメッセージ限界 (MSGMT)パラメーターにメッセージの最大数を指定した場合には、指定された重大度のその数のメッセージがジョブ・ログに入れられた時に、コンパイルが停止します。

#### **\*NOLOGMSG**

コンパイル・メッセージをジョブ・ログに書き込みません。

#### **\*LONGLONG**

LONG LONGデータ・タイプの使用を許可します。

#### **\*NOLONGLONG**

LONG LONGデータ・タイプの使用は許可しません。

#### **\*NORTTI**

実行時タイプID (RTTI)情報を生成しません。

#### **\*RTTIALl**

RTTI typeidおよびdynamic\_cast演算子に必要な情報を生成します。

#### **\*RTTITYPE**

RTTI typeid演算子に必要な情報のみを生成します。

#### **\*RTTICAST**

dynamic\_cast演算子に必要な情報のみを生成します。

#### **\*NOSHOWINC**

リストのソース・セクションで、またはデバッグ・ビューでユーザー組み込みファイルまたはシステム組み込みファイルを拡張しません。

#### **\*SHOWINC**

リストのソース・セクションで、またはデバッグ・ビューでユーザー組み込みファイルとシステム組み込みファイルの両方を拡張します。OUTPUTオプションか、\*ALL, \*SOURCE,または\*LISTの値のDBGVIEWパラメーターを指定する必要があります。

#### **\*SHOWSRC**

リストにソース・コードを表示します。このオプションは、\*SHOWINC, \*SHOWSYS,または\*SHOWUSRオプションによって変更することができます。

#### **\*NOSHOWSRC**

リストにソース・コードは表示しません。このオプションは、\*EXPMAC, \*SHOWINC, \*SHOWSYS,または\*SHOWUSRオプションで変更できます。

#### **\*NOSHOWSYS**

リストのソース・セクションで、またはデバッグ・ビューで#includeディレクティブ上のシステム組み込みファイルを拡張しません。システム組み込みファイルは、includeディレクティブの後に続けて不等号括弧(< >)で囲まれます。

#### **\*SHOWSYS**

リストのソース・セクションで、またはデバッグ・ビューで#includeディレクティブ上のシステム組み込みファイルを拡張します。OUTPUTオプションを指定するか、あるいは\*ALL, \*SOURCE,または\*LISTのDBGVIEWパラメーターを指定する必要があります。システム組み込みファイルは、includeディレクティブの後に続けて不等号括弧(< >)で囲まれます。

#### **\*NOSHOWUSR**

リストまたはデバッグ・ビューでユーザー組み込みファイルは拡張しません。ユーザー組み込みファイルは、includeディレクティブの後に続けて二重引用符(" ")で囲まれます。

#### **\*SHOWUSR**

リストのソース・セクションで、またはデバッグ・ビューで#includeディレクティブ上のユーザー組み込みファイルを拡張します。OUTPUTオプションか、\*ALL, \*SOURCE,または\*LISTの値のDBGVIEWパラメーターを指定する必要があります。ユーザー組み込みファイルは、includeディレクティブの後に続けて二重引用符(" ")で囲まれます。

#### **\*STDINC**

システム提供のヘッダー・ファイルがコンパイルのための検索パスに組み込まれます。

#### **\*NOSTDINC**

システム提供のヘッダー・ファイルはコンパイルのための検索パスに組み込まれません。



#### **\*NOSTDLOGMSG**

コンパイル・メッセージはSTDOUTストリームへ送られません。

#### **\*STDLOGMSG**

コンパイル・メッセージがSTDOUTストリームへ送られます。

#### **\*NOSYSINCPATH**

ユーザー組み込みファイルの検索パスは影響を受けません。

#### **\*SYSINCPATH**

ユーザー組み込みファイルの検索パスをシステム組み込みファイルの検索パスに変更します。機能的に、このオプションを使用することは、ユーザー#includeディレクティブ(#include "file\_name")の二重引用符を不等号括弧(#include <file\_name>)に変更することに相当します。

#### **\*NOXREF**

リストの中に相互参照テーブルは生成しません。

#### **\*XREF**

ソース・コード中の識別コードをそれらが表示される行の番号とともに示すリストを含む相互参照テーブルを生成します。OUTPUTオプションの指定が必要です。

#### **\*NOXREFREF**

リストの中で参照される識別コードの相互参照テーブルは生成しません。

#### **\*XREFREF**

リスト・ファイルで参照される変数、構造および関数名の相互参照テーブルを生成します。このテーブルには、識別コードが宣言された場所の行番号が示されます。OUTPUTオプションの指定が必要です。

上

---

## **チェックアウト・オプション (CHECKOUT)**

考えられるプログラミング・エラーを指示する通知メッセージの生成を選択できるオプションを指定します。1つのオプションを複数回指定したり、2つのオプションが競合する場合には、最後に指定されたものが使用されます。

このパラメーターには45個の値を指定できます。

#### **\*NONE**

CHECKOUTのためのすべてのオプションを使用不可とします。

#### **\*USAGE**

これは\*CONDを指定するのと同様です。

**\*ALL** CHECKOUTのためのすべてのオプションを使用可能とします。

以下に示す一連のパラメーター値の説明では、下線付きの値は、コンパイラーで使用されるデフォルト値を表します。

#### **\*NOCLASS**

クラスの使用に関する情報は表示しません。

#### **\*CLASS**

クラスの使用に関する情報を表示します。

**\*NOCOND**

条件式で考えられる冗長度または問題についての警告はしません。

**\*COND**

条件式で考えられる冗長度または問題について警告します。

**\*NOEFFECT**

有効でないステートメントについての警告はしません。

**\*EFFECT**

有効でないステートメントについて警告します。

**\*NOGENERAL**

一般チェックアウト・メッセージは生成しません。

**\*GENERAL**

一般チェックアウト・メッセージを生成します。

**\*NOLANG**

言語レベルの影響に関する情報は表示しません。

**\*LANG**

言語レベルの影響に関する情報を表示します。

**\*NOPARM**

未使用パラメーターについての警告はしません。

**\*PARM**

未使用パラメーターについて警告します。

**\*NOPORT**

移植不能言語構造についての警告はしません。

**\*PORT**

移植不能言語構造について警告します。

**\*NOREACH**

到達不能ステートメントについての警告はしません。

**\*REACH**

到達不能ステートメントについて警告します。

**\*NOTEMP**

コンパイラーが一時変数を作成した場合にメッセージは生成しません。

**\*TEMP**

コンパイラーが一時変数を作成した場合にメッセージを生成します。

**\*NOTRUNC**

データの考えられる切り捨てまたは喪失についての警告はしません。

**\*TRUNC**

データの考えられる切り捨てまたは喪失について警告します。

**\*NOUNUSED**

未使用の自動または静的変数は検査しません。

**\*UNUSED**

未使用の自動または静的変数を検査します。

---

## 最適化 (OPTIMIZE)

生成されるオブジェクトの最適化のレベルを指定します。

- 10**      生成されるコードは最適化されません。このレベルでのコンパイル時間は最短です。このレベルでは、デバッグ中に変数を表示して変更することができます。
- 20**      コードについて一部の最適化が実行されます。このレベルでは、デバッグ中にユーザーが変数を表示することができますが、変更することはできません。
- 30**      生成されるコードについて完全な最適化が実行されます。デバッグ・セッション中は、ユーザー変数は変更することができませんが、表示することはできます。表示される値は、変数の現行値でない場合があります。
- 40**      レベル30で実行されたすべての最適化が生成されるコードについて実行されます。さらに、コードは、命令トレースおよび呼び出しトレース・システム機能を使用可能にするプロシージャーのプロログおよびエピログ・ルーチンから除去されます。このコードを除去することによって、リーフ・プロシージャーの作成が可能になります。リーフ・プロシージャーとは、他のプロシージャーに対する呼び出しを含まないプロシージャーです。リーフ・プロシージャーに対するプロシージャー呼び出しのパフォーマンスは、通常のプロシージャーに対する呼び出しより高速となります。

上

---

## インライン・オプション (INLINE)

コンパイラーが呼び出し先関数の命令による関数呼び出しの置き換えを考慮する必要があるかどうかを指定します。関数をインライン化することによって、呼び出しのオーバーヘッドが取り除かれるので、結果としてより良好に最適化することができます。何度も呼び出される小さな関数は、インライン化の候補として適しています。

### 要素1: インライナー

インライン化を使用するかどうかを指定します。

**\*OFF**      コンパイル単位にインライン化は実行されないことを指定します。

**\*ON**      コンパイル単位にインライン化が実行されることを指定します。デバッグ・ビューが指定された場合には、インライン化はオフにされます。

### 要素2: モード

そのしきい値および限界値に基づいて、インライナーは関数のインライン化を自動的に試みる必要があるかどうかを指定します。

**\*AUTO**

インライナーは指定されたしきい値および限界値に基づいて関数をインライン化できるかどうかを判断する必要があることを指定します。\*AUTOは#PRAGMA NOINLINEディレクティブによって上書きされます。

### **\*NOAUTO**

#PRAGMA INLINEディレクティブによって指定された関数のみをインライン化の候補として考慮する必要があることを指定します。

### **要素3: しきい値**

自動インライン化の候補にできる関数の最大サイズを指定します。このサイズは、抽象コード単位(ACU)で測定されます。ACUのサイズは関数中の実行可能コードと比例します。ソース・コードは、コンパイラによってACUに変換されます。

**250** 250のしきい値を指定します。

**I-65535**

ACUのしきい値を指定します。

### **\*NOLIMIT**

しきい値をプログラム・オブジェクトの最大サイズとして定義します。

### **要素4: 限界**

自動インライン化を停止する前に関数を拡張できる最大相対サイズを指定します。

**2000** 2000 ACUの限界を指定します。

### **\*NOLIMIT**

限界はプログラム・オブジェクトの最大サイズとして定義されます。システム限界が検出されることがあります。

**I-65535**

ACUの限界を指定してください。

### **要素5: 報告書**

コンパイラ・リストとともにインライナー報告書を生成するかどうかを指定します。

**\*NO** インライナー報告書は生成されません。

**\*YES** インライナー報告書がコンパイラ・リストの一部として生成されます。インライナー報告書を生成するには、OUTPUTオプションの指定が必要です。

上

---

## **モジュール作成オプション (MODCRTOPT)**

モジュール・オブジェクトの作成時に使用するオプションを指定します。

### **\*NOKEEPILDTA**

モジュール・オブジェクトと一緒に中間言語データは保管されません。

### **\*KEEPILDTA**

モジュール・オブジェクトと一緒に中間言語データが保管されます。

上

---

## デバッグ・ビュー (DBGVIEW)

作成されたプログラムまたはサービス・プログラム・オブジェクトでモジュールに対して使用できるデバッグのレベルを指定します。ソース・レベルのデバッグ用にどのソース・ビューが使用可能かも指定されます。デバッグ・ビューを要求すると、インライン化はオフにされます。

### \*NONE

デバッグ機能はモジュール・オブジェクトに挿入されません。

**\*ALL** すべてのデバッグ・オプション(\*STMT, \*SOURCE,および\*LIST)が使用可能になります。

### **\*STMT**

プログラム・ステートメント番号および記号識別コードを使用したモジュール・オブジェクトのデバッグが可能になります。

注: \*STMTオプションを使用してモジュール・オブジェクトをデバッグするには、リストが必要です。

### **\*SOURCE**

モジュール・オブジェクトのデバッグのためのソース・ビューを生成します。OPTIONパラメータの値\*NOSHOWINC, \*SHOWINC, \*SHOWSYS,および\*SHOWUSRによって、作成されるソース・ビューの内容が決まります。

注: このビューをデバッグ用に使用するには、モジュールが作成された後に、ルート・ソース・ファイルを変更、名前変更、または移動しないようにしてください。

**\*LIST** モジュール・オブジェクトのデバッグ用のリスト・ビューを生成します。OPTIONパラメータの値\*SHOWINC, \*SHOWUSR, \*SHOWSYS, および\*NOSHOWINCによって、作成されるリスト・ビューの内容が決まります。

上

---

## 名前の定義 (DEFINE)

ファイルがコンパイラによって処理される前に有効となるプリプロセッサ・マクロを指定します。形式DEFINE(MACRO)の使用はDEFINE('MACRO=1')と同等です。

### 単一値

### \*NONE

マクロは定義されません。

その他の値 (反復は最大32回まで)

**'名前'または'名前=値'**

最大32個のマクロを定義できます。各マクロ名はアポストロフィで囲まれます。マクロ名の最大長は80文字です。アポストロフィは80の文字列の一部ではありません。アポストロフィは、大文字小文字を区別するマクロに必要です。

注: コマンドで定義されたマクロによって、ソースにある同じ名前のすべてのマクロ定義が上書きされますが、コンパイラにより警告メッセージが生成されます。 #define max(a,b) ((a)>(b)?(a):(b))など関数と類似したマクロをコマンド行で定義することはできません。

上

---

## 言語レベル (LANGLVL)

コンパイラーの機能と、ソースの作成時に宣言されるプロトタイプを指定します。

### **\*EXTENDED**

プリプロセッサ変数 `__EXTENDED__` を定義し、他の言語レベル変数は未定義とします。このパラメーターは、ILE C++の機能が必要な場合に使用する必要があります。

**\*ANSI** プリプロセッサ変数 `__ANSI__`, `__STDC__`, および `__cplusplus98_interface__` を定義し、他の言語レベル変数を未定義にします。ANSI標準C++だけが使用可能とされます。

### **\*LEGACY**

このオプションによって、一部のソース構成が初期のコンパイラーで受け入れ可能になります。

上

---

## 別名 (ALIAS)

作成中のモジュール・オブジェクトに適用される別名割り当てアサーションを指定します。

このパラメーターには3個の値を指定できます。

**\*ANSI** モジュール・オブジェクトで、ポインターが、同一タイプのオブジェクトのみを指し示すようになります。

### **\*NOANSI**

モジュール・オブジェクトは、\*ANSI別名割り当て規則を使用しません。

### **\*ADDRTAKEN**

モジュール・オブジェクトで、変数のアドレスが使用されない場合に、その変数のクラスがポインターから切り離されます。

### **\*NOADDRTAKEN**

モジュール・オブジェクトは、\*ADDRTAKEN別名割り当て規則を使用しません。

### **\*ALLPTRS**

モジュール・オブジェクトで、2つのポインターに別名を割り当てることができなくなります。

### **\*NOALLPTRS**

モジュール・オブジェクトは、\*ALLPTRS別名割り当て規則を使用しません。

### **\*TYPEPTR**

モジュール・オブジェクトで、タイプの異なる2つのポインターに別名を割り当てることができなくなります。

### **\*NOTYPEPTR**

モジュール・オブジェクトは、\*TYPEPTR別名割り当て規則を使用しません。

上

---

## SYSTEMインターフェースOPT (SYSIFCOPT)

作成中のモジュール・オブジェクトに使用されるシステム・インターフェース・オプションを指定します。

### **\*IFS64IO**

モジュール・オブジェクトは、64ビットCストリーム入出力操作に統合ファイル・システムを使用します。

### **\*IFSIO**

モジュール・オブジェクトは、Cストリーム入出力操作に統合ファイル・システムを使用します。

### **\*NOIFSIO**

モジュール・オブジェクトは、Cストリーム入出力操作にデータ管理機能ファイル・システムを使用します。

上

---

## **LOCALEオブジェクト・タイプ (LOCALETYPE)**

作成中のモジュール・オブジェクトで使用されるロケール・サポートのタイプを指定します。

### **\*LOCALE**

このオプションで作成されたモジュール・オブジェクトは、\*LOCALEオブジェクトによって指定されるロケール・サポートを使用します。

### **\*LOCALEUCS2**

このオプションで作成されたモジュール・オブジェクトは、\*LOCALEオブジェクトによって指定されるロケール・サポートを使用します。広幅(WIDE)文字タイプには2つの2バイト汎用文字セットの値が含まれることになります。

### **\*LOCALEUTF**

このオプションで作成されたモジュール・オブジェクトは、\*LOCALEオブジェクトによって指定されるロケール・サポートを使用します。広幅(WIDE)文字タイプには、4バイトのUTF-32の値が含まれます。狭幅(NARROW)文字タイプには、UTF-8の値が含まれます。

上

---

## **メッセージのフラグ・レベル (FLAG)**

リストに表示するメッセージのレベルを指定します。

- 0** 通知レベルで始まるすべてのメッセージが表示されます。
- 10** 警告レベルで始まるすべてのメッセージが表示されます。
- 20** エラー・レベルで始まるすべてのメッセージが表示されます。
- 30** 重大エラー・レベルで始まるすべてのメッセージが表示されます。

上



---

## メッセージ限界 (MSGLMT)

指定されたメッセージ重大度(またはそれ以上)のメッセージの最大数を指定します。コンパイルが停止するまでに、このメッセージ数までメッセージを生成できます。

### 要素1: メッセージ限界

指定したメッセージ重大度レベル以上で生じさせることができるメッセージの最大数を指定します。

#### \*NOMAX

指定されたメッセージ重大度レベルで生じたメッセージの数とは無関係に、コンパイルが続行されます。

#### 0-32767

生じさせることができるメッセージの数を指定します。

### 要素2: メッセージ限界の重大度

メッセージ重大度を指定します。ここで指定した重大度以上のメッセージが特定の数（要素1に指定された数）だけ生成されるとき、コンパイルを停止できます。

- 30**      コンパイルが停止するまでに、重大度30のメッセージを指定の数だけ生成できます。
- 0**        コンパイルが停止するまでに、重大度0以上のメッセージを指定の数だけ生成できます。
- 10**      コンパイルが停止するまでに、重大度10以上のメッセージを指定の数だけ生成できます。
- 20**      コンパイルが停止するまでに、重大度20以上のメッセージを指定の数だけ生成できます。

上

---

## MODULE OBJECTの置き換え (REPLACE)

オブジェクトの既存のバージョンを現行バージョンで置き換えるかどうかを指定します。

**\*YES**    既存のオブジェクトが新規バージョンで置き換えられます。旧バージョンはQRPLOBJライブラリーに移動し、システム日付および時刻に基づいて名前変更されます。置き換えられたオブジェクトのテキスト記述は、オリジナルのオブジェクトの名前に変更されます。旧オブジェクトは、それが明示的に削除されていないければ、次のIPL時に削除されます。

**\*NO**     既存のオブジェクトは置き換えられません。指定のライブラリーで同じ名前のオブジェクトが見つかった場合、メッセージが表示されて、コンパイルが停止します。

上

---

## 権限 (AUT)

オブジェクトに対する特定権限がないユーザー、権限リスト上にないユーザー、またはそのグループにオブジェクトに対する特定権限がないユーザーに許可される権限を指定します。

#### \*LIBCRTAUT

オブジェクトに対する共通権限はターゲット・ライブラリー（作成されたオブジェクトが入れられるライブラリー）のCRTAUTキーワードからとられます。この値は、オブジェクトの作成時に決定されます。オブジェクトの作成後にライブラリーのCRTAUT値が変更された場合には、この新規の値はライブラリー中の既存のどのオブジェクトにも影響しません。

**\*ALL** 所有者に限定されるか、あるいは権限リスト管理権限によって制御される操作を除いて、オブジェクトでのすべての操作に対する権限が提供されます。どのユーザーも、オブジェクトの存在を制御し、その機密保護を指定し、変更し、また、その所有権の転送も含めてオブジェクトに対する基本操作を実行することができます。

**\*CHANGE**

すべてのデータ権限、および所有者に限定されるか、あるいは権限リスト管理権限によって制御される操作を除いて、オブジェクトでのすべての操作を実行する権限が提供されます。オブジェクトを変更し、それに基本操作を実行することができます。

**\*USE** オブジェクト操作権、読み取り権限、およびモジュール・オブジェクトのバインドなど、オブジェクトへの基本読み取り専用操作に対する権限が提供されます。特定権限のないユーザーは、オブジェクトを変更することができません。

**\*EXCLUDE**

特殊権限のないユーザーはオブジェクトにアクセスすることができません。

**name** オブジェクトの追加先となるユーザーと権限で構成された権限リストの名前を指定してください。オブジェクトはこの権限リストによって保護され、そのオブジェクトの共通権限は\*AUTLに設定されます。権限リストは、コマンドが出された時にシステム上に存在していなければなりません。

上

---

## ターゲット・リリース (TGTRLS)

ユーザーが作成中のオブジェクトを使用したいオペレーティング・システムのリリースを指定します。

**\*CURRENT**および**\*PRV**の値について示される例の中、および リリース・レベル の値を指定する場合には、形式VXRXXMXを使用してリリースが指定されます。ここで、VXはバージョン、RXはリリース、そしてMXはモディフィケーション・レベルです。たとえば、V4R5M0は、バージョン4、リリース5、モディフィケーション・レベル0です。

**\*CURRENT**

オブジェクトは、ユーザーのシステムで現在実行されているオペレーティング・システムのリリースで使用されます。例えば、V5R5M0がシステムで実行されている場合に、\*CURRENTは、ユーザーはオブジェクトをV5R5M0の導入されたシステムで使用したいことを意味します。ユーザーは、また、このリリース以降のオペレーティング・システムが導入されたシステムでもオブジェクトを使用することができます。

**\*PRV** オブジェクトは、オペレーティング・システムの前のリリース（モディフィケーション・レベル0）で使用されます。例えば、V5R5M0がシステムで実行されている場合に、\*PRVは、ユーザーはオブジェクトをV5R5M0の導入されたシステムで使用したいことを意味します。ユーザーは、また、このリリース以降のオペレーティング・システムが導入されたシステムでもオブジェクトを使用することができます。

**リリース・レベル**

リリースを形式VXRXXMXで指定します。オブジェクトは、指定したリリースのシステムでか、あるいはこのリリース以降のオペレーティング・システムが導入されたシステムで使用することができます。

有効な値は、現行バージョン、リリース、およびモディフィケーション・レベルによって異なります。これらは各新規リリースごとに変更されます。このコマンドによってサポートされる最も初期のリリース・レベルより前のリリース・レベルを指定した場合には、エラー・メッセージが送られます。

上

---

## パフォーマンス収集の使用可能化 (ENBPFRCOL)

オブジェクト内でパフォーマンス測定コードを生成すべきかどうかを指定します。収集されたデータは、システム・パフォーマンス・ツールによって使用し、アプリケーションのパフォーマンスのプロファイルを作成することができます。作成されたオブジェクト内でパフォーマンス測定コードを生成すると、オブジェクトのサイズがわずかに増えるため、パフォーマンスに影響が及ぶ可能性があります。

注: V5R5M0 以降では、このパラメーターは、作成されたオブジェクトに対して作用しなくなります。このパラメーターは、V5R5M0 より前のリリースとの互換性を保つために用意されています。

**\*PEP** パフォーマンス統計は、プログラム・エントリー・プロシージャーの入り口と出口でのみ収集されます。この値は、アプリケーションの全般的なパフォーマンス情報を収集したい場合に選択します。

### **\*ENTRYEXIT \*NONLEAF**

パフォーマンス統計は、プログラム・オブジェクトの全プロシージャー（リーフ・プロシージャーを除く）の入り口と出口で収集されます。これにはプログラムPEPルーチンが含まれます。

この選択項目は、アプリケーション内の他のルーチン呼び出すルーチンに関する情報だけを取り込みたい場合に有用です。

### **\*ENTRYEXIT \*ALLPRC**

パフォーマンス統計は、プログラム・オブジェクトの全プロシージャー（リーフ・プロシージャーを含む）の入り口と出口で収集されます。これには、プログラムPEPルーチンが含まれます。

この選択項目は、すべてのルーチンに関する情報を取り込みたい場合に有用です。このオプションは、ユーザーのアプリケーションから呼び出されるすべてのプログラム・オブジェクトが、\*PEP、\*ENTRYEXIT、\*FULLのいずれかのオプションを使用して作成されたことが判明している場合に使用します。それ以外の場合、パフォーマンス測定に対して使用可能にならない他のプログラム・オブジェクトがユーザーのアプリケーションから呼び出されると、パフォーマンス・ツールは、そのプログラム・オブジェクトによる資源の使用についてユーザーのアプリケーションに任せます。このため、資源が実際に使用されている場所を判別するのが困難になることがあります。

### **\*FULL \*NONLEAF**

パフォーマンス統計は、リーフ・プロシージャーではないすべてのプロシージャーの入り口と出口で収集されます。また、外部プロシージャーに対する個々の呼び出しの前および後にも統計が収集されます。

### **\*FULL \*ALLPRC**

パフォーマンス統計は、リーフ・プロシージャーを含むすべてのプロシージャーの入り口と出口で収集されます。また、外部プロシージャーに対する個々の呼び出しの前および後にも統計が収集されます。

このオプションは、作成時に\*PEP、\*ENTRYEXIT、\*FULLのいずれも使用されなかった他のプログラム・オブジェクトをユーザーのアプリケーションから呼び出そうと考えている場合に使用します。このオプションを使用すると、パフォーマンス・ツールは、ユーザーのアプリケーションによ

って使用されている資源と、そのアプリケーションから呼び出されたプログラム・オブジェクトによって使用されている資源を、（そのプログラム・オブジェクトがパフォーマンス測定に対して使用可能にならない場合でも）区別できるようになります。このオプションを使用すると、効率は最も悪くなりますが、アプリケーション内の各種プログラム・オブジェクトを選択的に分析できるようになります。

上

---

## パフォーマンス・オプション (PFROPT)

パフォーマンスを高めるために使用可能な各種のオプションを指定します。それらは、1つまたは複数のブロックで区切り、任意の順序で指定できます。1つのオプションが複数回指定されたり、2つのオプションが競合する場合には、最後に指定されたものが使用されます。

このパラメーターには2個の値を指定できます。

### **\*SETFPCA**

浮動小数点計算に対するANSIのセマンティクスを達成するために、コンパイラーに浮動小数点計算属性を設定させます。

### **\*NOSETFPCA**

計算属性は設定されません。このオプションは、作成中のオブジェクトに浮動小数点計算が含まれない場合にのみ使用してください。

### **\*NOSTRONLY**

コンパイラーは文字列を書き込み可能メモリーに入れる必要があることを指定します。

### **\*STRONLY**

コンパイラーは文字列を読み取り専用メモリーに入れることができることを指定します。

上

---

## プロファイル作成データ (PRFDTA)

モジュール・オブジェクトのプログラム・プロファイル作成データ属性を指定します。プログラム・プロファイル作成は、プロシージャーおよびプロシージャー内のコードを統計データ（プロファイル作成データ）に基づいて配列し直すために使用される拡張最適化手法です。

### **\*NOCOL**

モジュール・オブジェクトは、プロファイル作成データの収集に使用できません。

**\*COL** モジュール・オブジェクトは、プロファイル作成データの収集に使用できます。**\*COL**は、最適化レベルが30以上である場合にのみ指定することができます。

上

---

## テラスペース・オプション (TERASPACE)

テラスペース・ストレージの処理にモジュール・オブジェクトを使用可能にするかどうかを指定します。これには、モジュール・オブジェクトによって割り振られるテラスペース・ストレージや、他のテラスペース対応プログラムおよびサービス・プログラムのオブジェクトから渡されたパラメーターによって割り振られるテラスペース・ストレージが含まれます。

## 単一値

**\*NO** モジュール・オブジェクトは、テラスペースから割り振られたストレージのアドレス指定の処理に使用できません。

注: V5R5M0以降では、テラスペースから割り振られたストレージのアドレス指定の処理に、すべてのモジュールが使用できます。ただし、\*NOが指定された場合には、以下にリストされているコンパイラー機能が使用できなくなります。

### 要素1: テラスペース使用可能

**\*YES** モジュール・オブジェクトは、他のテラスペース対応プログラムやサービス・プログラムのオブジェクトから渡されたパラメーターも含めて、テラスペースから割り振られたストレージのアドレス指定の処理に使用できます。さらに、以下のコンパイラー機能も使用できます。

- テラスペース・ストレージへのアクセスに使用される8バイト・ポインターを作成できるように、ポインターは\_\_ptr64で修飾できます。
- テラスペース・ストレージ・モデルは、STGMDL(\*TERASPACE)コンパイラー・オプションを使用して指定できます。
- LLP64データ・モデルは、DTAMDL(\*LLP64)コンパイラー・オプションまたは#pragma datamodel(llp64)ディレクティブを使用して指定できます。
- ポインター差分演算は、ptrdiff\_tの結果ではなく、signed long longの結果を戻します。

### 要素2: TERASPACE INTERFACESの使用

#### **\*NOTSIFC**

モジュール・オブジェクトは、デフォルトで、ストレージ機能の非テラスペース・バージョンを使用するようになります。

#### **\*TSIFC**

モジュール・オブジェクトは、デフォルトで、ストレージ機能のテラスペース・バージョンを使用するようになります。コンパイラーは、マクロ変数\_\_TERASPACE\_\_を定義することになります。

上

---

## ストレージ・モデル (STGMDL)

作成されたオブジェクトで使用する記憶域のタイプを指定します。

#### **\*SNGLVL**

作成されたオブジェクトは、単一レベルの記憶域を使用します。

#### **\*TERASPACE**

作成されたオブジェクトは、テラスペース記憶域を使用します。

#### **\*INHERIT**

作成されたオブジェクトは、単一レベルとテラスペースのどちらの記憶域も使用することができます。使用される記憶域のタイプは、呼び出し元に必要な記憶域のタイプによって異なります。

上

---

## データ・モデル (DTAMDL)

INT, LONG, POINTERとして宣言される変数のサイズ（バイト数）を指定します。

**\*P128** INT, LONG, POINTERのサイズがそれぞれ4, 4, 16となるようにします。

**\*LLP64**

INT, LONG, POINTERのサイズがそれぞれ4, 4, 8となるようにします。コンパイラーは、マクロ `__LLP64_IFC__` を定義することになります。

上

---

## ランタイム・バインディング (RTBND)

作成されたオブジェクトの実行時バインディング・ディレクトリーを指定します。

**\*DEFAULT**

作成されたオブジェクトは、デフォルトのバインディング・ディレクトリーを使用します。

**\*LLP64**

作成されたオブジェクトは、64ビット実行時バインディング・ディレクトリーを使用します。この値を使用できるのは、テラスペース・ストレージ・モデル、64ビット・データ・モデル、およびテラスペース・ストレージ機能インターフェースの各オプションと一緒に使用する場合のみです。コンパイラーは、マクロ `__LLP64_RTBNB__` を定義します。

上

---

## パック構造 (PACKSTRUCT)

構造のメンバーに使用する位置合わせ境界を指定します。

**\*NATURAL**

構造メンバーはその自然境界で位置合わせされます。たとえば、短形式整数は2バイトで位置合わせされます。16バイト・ポインターは、常に16バイト境界で位置合わせされます。

- 1 構造メンバーを1バイトの位置合わせでパックします。
- 2 構造メンバーを2バイトの位置合わせでパックします。
- 4 構造メンバーを4バイトの位置合わせでパックします。
- 8 構造メンバーを8バイトの位置合わせでパックします。
- 16 構造メンバーを16バイトの位置合わせでパックします。

上

---

## Enumサイズ (ENUM)

コンパイラーが列挙型を表すために使用するバイト数を指定します。

**\*SMALL**

すべてのENUM変数を、値の範囲を表すことができる最も小さいサイズにします。

- 1 すべてのENUM変数を1バイトにします。
- 2 すべてのENUM変数を2バイトにします。
- 4 すべてのENUM変数を4バイトにします。

**\*INT** ANSI標準のENUMサイズである4バイトを使用します。



---

## 依存関係情報 (MAKEDEP)

依存関係情報をファイルの中に生成するかどうかを指定します。 この情報は、MAKE TOOLによって使用されます。

### \*NONE

依存関係情報は生成しません。

### *path-name*

依存関係情報を保管するストリーム・ファイルのパス名を指定します。

---

## プリプロセッサ・オプション (PPGENOPT)

ソース・コードのコンパイル時に使用するプリプロセッサ生成オプションを指定します。

### 単一値

### \*NONE

ソース・ファイルに対してコンパイラ全体を実行します。 プリプロセッサの出力をファイルにはコピーしません。

**\*DFT** 入力ソースに対してプリプロセッサを実行します。 \*RMVCOMMENTおよび\*GENLINEは、プリプロセッサの出力を生成するためのオプションとして使用されます。 出力ソース・ファイルおよびメンバーを指定するにはPPSRCFILEおよびPPSRCMBRを使用し、あるいはプリプロセッサの出力を入れるストリーム・ファイルを指定するにはPPSRCSTMFを使用します。

### その他の値（反復は最大2回まで）

### **\*RMVCOMMENT**

前処理中に、注記を除去します。

### **\*NORMVCOMMENT**

前処理中に、注記を保存します。

### **\*GENLINE**

プリプロセッサ出力の中に#LINEディレクティブを生成します。

### **\*NOGENLINE**

プリプロセッサの出力から#LINEディレクティブを抑制します。

---

## 出力ソース・ファイル (PPSRCFILE)

プリプロセッサ出力用の物理ファイルを指定します。

### 修飾子1: 出力ソース・ファイル

**name** プリプロセッサの出力のための物理ファイルの名前を指定します。



## 修飾子2: ライブラリー

### \*CURLIB

プリプロセッサの出力は現行ライブラリーに作成されます。 ジョブに現行ライブラリーがない場合には、プリプロセッサの出力ファイルはQGPLライブラリーの中に作成されます。

**name** プリプロセッサの出力のためのライブラリーの名前を指定します。

上

---

## 出力ソース・メンバー (PPSRCMBR)

プリプロセッサ出力用の物理ファイル・メンバーを指定します。

### \*MODULE

このモジュール (**MODULE**)パラメーターに指定された名前が、プリプロセッサ出力メンバー名として使用されます。

**name** プリプロセッサの出力のためのメンバーの名前を指定します。

上

---

## 出力ストリーム・ファイル (PPSRCSTMF)

プリプロセッサの出力のためのストリーム・ファイルのパス名を指定します。

### \*SRCSTMF

ソース・ストリーム・ファイル (**SRCSTMF**)パラメーターに指定されたパス名が、プリプロセッサ出力パス名として使用されます。このファイルには拡張子'.I'があります。

### path-name

プリプロセッサの出力ストリーム・ファイルのためのパス名を指定します。

上

---

## 組み込みディレクトリー (INCDIR)

組み込みファイルを検出するためにコンパイラーが使用する検索パスに追加する1つまたは複数のディレクトリーを指定します。

検索パスは、OPTIONキーワードで次のパラメーターを使用することにより、さらに変更することができます。

- \*INCDIRFIRSTまたは\*NOINCDIRFIRST
- \*SYSINCPATHまたは\*NOSYSINCPATH
- \*STDINCまたは\*NOSTDINC

### \*NONE

変更されなければ、省略時のシステム組み込みディレクトリーおよびソース・ディレクトリーからユーザー組み込みファイルが検索されます。

### *path-name*

組み込みファイルの検索場所となるディレクトリーのパス名を指定してください。最大32個のディレクトリーを指定できます。指定したディレクトリーに加えて、ソース・ディレクトリーからもユーザー組み込みファイルが検索されます。

上

---

## コンパイラー・サービスOPT (CSOPT)

1つまたは複数のコンパイラー・サービス・オプションを指定します。このパラメーターにより、IBMはリリース相互間で切り替え可能なコンパイラーを提供することができます。

### \*NONE

コンパイラー・サービス・オプションは選択されません。

**文字値** モジュール・オブジェクトの作成時に使用するコンパイラー・サービス・オプションを含む文字ストリングを指定してください。有効な文字列は、PTFカバー・レターまたはリリース情報に記述されます。

上

---

## ライセンス内部コードOPT (LICOPT)

1つまたは複数のライセンス内部コード・コンパイル時オプションを指定します。このパラメーターは、個別のコンパイル時オプションを選択できるようにするもので、選択された各タイプのコンパイラー・オプションの潜在的な長所と短所を理解している上級プログラマーを対象としています。

### \*NONE

コンパイル時オプションは選択されません。

**文字値** 選択されたライセンス内部コード・コンパイル時オプションが、モジュール・オブジェクトの作成時に使用されます。特定のオプションでは、作成されたモジュール・オブジェクトをデバッグする能力が低下することがあります。

上

---

## 省略時の文字タイプ (DFTCHAR)

CHARデータ・タイプの省略時の符号を指定します。

### \*UNSIGNED

省略時のCHARタイプを符号なしとします。

### \*SIGNED

省略時のCHARタイプを符号付きとします。

上

---

## ターゲットCCSID (TGTCSSID)

結果のモジュール・オブジェクトに保管されるデータを記述するために使用されるターゲット・コード化文字セットIDを指定します。

### \*SOURCE

ルート・ソース・ファイルのCCSIDが使用されます。

**\*JOB** 現行ジョブのCCSIDが使用されます。

**\*HEX** CCSID 65535が使用されます。これは、文字データはバイナリー・データとして取り扱われ、変換されないことを指示します。

### ***1-65535***

使用するCCSIDを指定します。

上

---

## テンプレート・オプション (TEMPLATE)

コンパイラーに対するテンプレート・オプションを指定します。

**要素1: 一時組み込みディレクトリー**

### \*NONE

テンプレートは生成されません。

### **\*TEMPINC**

テンプレートは、ルート・ソース・ファイルが見つかったディレクトリーの中に作成されるTEMPINCという名前のディレクトリーの中に生成されます。ソース・ファイルがストリーム・ファイルでない場合、ソース・ファイルが置かれているライブラリー内に、ファイルTEMPINCが作成されます。TEMPLATE(\*TEMPINC)パラメーターとTMPLREGパラメーターは相互に排他的です。

### ***path-name***

コンパイラーがテンプレートを生成するディレクトリーのパス名を指定してください。

**要素2: 生成済み最大見出し数**

**1** テンプレートが入れられる、生成されるヘッダーの最大数。

### ***1-99999***

生成されるヘッダー・ファイルの最大数を指定してください。

**要素3: テンプレートの妥当性検査**

構文解析とセマンティック検査が、テンプレート定義の実装に適用されるのか、テンプレートのインスタンス化にのみ適用されるのかを制御します。コンパイラーには、警告やエラー・メッセージを生成するオプションを指定できます。使用可能なパラメーターは次のとおりです。

**\*NO** 構文解析時に、前のバージョンのコンパイラーに合わせて作成されたコードで発行されるエラーの数を減らしません。

### **\*WARN**

セマンティック・エラーに関して警告メッセージを発行します。構文解析時に検出されたエラーに関しては、エラー・メッセージを発行します。

## **\*ERROR**

テンプレートがインスタンス化されていない場合でも、テンプレート実装での問題をエラーとして扱います。

上

---

## **テンプレート・レジストリー (TMPLREG)**

ソース内で検出されたすべてのテンプレートを記録し、テンプレートごとにインスタンス化が1回だけ行われるようにします。TMPLREGパラメーターとTEMPLATE(\*TEMPINC)パラメーターは相互に排他的です。

### **\*NONE**

テンプレート情報の追跡にテンプレート・レジストリー・ファイルを使用しません。

**\*DFT** ソース・ファイルがストリーム・ファイルの場合、デフォルト名が'templateregistry'のテンプレート・レジストリー・ファイルが、ソース・ディレクトリー内に作成されます。ソース・ファイルがストリーム・ファイルでない場合、ソースが置かれているライブラリー内に、メンバーQTmplRegを含むファイルQTmplRegが作成されます。

### **path-name**

テンプレート・レジストリー情報の保管先となるストリーム・ファイルのパス名を指定します。

上

---

## **Weakテンプレート定義 (WEAKTMPL)**

弱い定義をテンプレート・クラスの静的メンバーに使用するかどうかを指定します。テンプレート・クラスの静的メンバーを弱く定義すると、プログラムやサービス・プログラムのオブジェクトで複数の定義が衝突することを防止できます。

**\*YES** テンプレート・クラスの静的メンバーに、弱い定義が使用されます。

**\*NO** テンプレート・クラスの静的メンバーに、弱い定義は使用されません。

上

---

## **例**

### **例1:ソース・ファイル・メンバーからのモジュールの作成**

```
CRTCPPMOD MODULE(QTEMP/TEST) SRCFILE(QTEMP/QCPPSRC)
```

このコマンドは、モジュール・オブジェクトTESTをライブラリーQTEMP内に作成します。C++ソースは、ライブラリーQTEMPにあるソース・ファイルQCPPSRCのメンバーTEST内に指定されます。

### **例2:ストリーム・ファイルからのモジュールの作成**

```
CRTCPPMOD MODULE(QTEMP/TEST) SRCSTMF('/tmp/mySrc.C')
```

このコマンドは、モジュール・オブジェクトTESTをライブラリーQTEMP内に作成します。C++ソースは、ストリーム・ファイル/tmp/mySrc.C内に指定されます。

### **例3:コンパイル・リストのコンパイルおよび生成**

```
CRTCPPMOD MODULE(TEST) OUTPUT(*PRINT) OPTION(*XREF)
```

このコマンドは、モジュール・オブジェクトTESTをジョブの現行ライブラリー内に作成しますが、現行ライブラリーが定義されていない場合は、ライブラリーQGPL内に作成します。C++ソースは、ソース・ファイルQCPPSRCのメンバーTEST内に指定されると想定されます（これは、ライブラリー・リストを使用して指定されます）。コンパイラー・リストが入ったスプール・ファイルが生成されます。コンパイラー・リスト・ファイルには、相互参照リストが格納されます。

#### 例4:すべてのデバッグ・ビューを備えたモジュールの作成

```
CRTCPMOD  MODULE(QTEMP/TEST) SRCFILE(QTEMP/QCPPSRC)
          SRCMBR(TESTER)  DBGVIEW(*ALL)
```

このコマンドは、デバッグ時に使用されるステートメント、ソース、およびリストの各ビューを備えたモジュール・オブジェクトTESTをライブラリーQTEMP内に作成します。C++ソースは、ライブラリーQTEMPにあるソース・ファイルQCPPSRCのメンバーTESTER内に指定されます。

#### 例5:テラスペース対応モジュールの作成

```
CRTCPMOD  MODULE(QTEMP/TEST) SRCFILE(QTEMP/QCPPSRC)
          TERASPACE(*YES *TSIFC) STGMDL(*TERASPACE)
          DTAMDLL(*LLP64)  RTBND(*LLP64)
```

このコマンドは、モジュール・オブジェクトTESTをライブラリーQTEMP内に作成します。C++ソースは、ライブラリーQTEMPにあるソース・ファイルQCPPSRCのメンバーTEST内に指定されます。このモジュールは、テラスペース・ストレージを使用する場合に使用可能にされます。ヒープ・スペース、自動ストレージ、および静的ストレージは、テラスペース・ストレージ内に入れられます。デフォルト・ポインター・サイズは8バイトになり、8バイト・ランタイムが使用されます。

#### 例6:ファイルへの前処理済みソースの保管

```
CRTCPMOD  MODULE(TEST) SRCFILE(QTEMP/QCPPSRC)
          PPGENOPT(*DFT) PPSRCFILE(QTEMP/QCPPSRC)
          PPSRCMBR(TESTI)
```

このコマンドは、ライブラリーQTEMPにあるソース・ファイルQCPPSRCのメンバーTEST内に指定されているC++ソースに対してコンパイラー・プリプロセッサを実行し、同じソース・ファイルのメンバーTESTI内に出力ソースを保管します。モジュール・オブジェクトは作成されません。

上

---

## エラー・メッセージ

### \*ESCAPE メッセージ

#### CZS0613

コンパイルに失敗しました。

上



## DFU表示装置ファイルの作成 (CRTDFUDSPF)

実行可能場所: すべての環境 (\*ALL)  
スレッド・セーフ: いいえ

パラメーター  
例  
エラー・メッセージ

制御言語(CL)コマンド(CRTDFUDSPF)はDFU表示装置ファイルを作成します。

### エラー・メッセージ: CRTDFUDSPF

なし

上

### パラメーター

キーワード	記述	選択項目	ノート
FILE	表示装置ファイル	修飾オブジェクト名	必須, 定位置 1
	修飾子 1: 表示装置ファイル	名前	
	修飾子 2: ライブラリー	名前, *CURLIB	
SRCFILE	ソース・ファイル	修飾オブジェクト名	必須, 定位置 2
	修飾子 1: ソース・ファイル	名前	
	修飾子 2: ライブラリー	名前, *LIBL, *CURLIB	
SRCMBR	ソース・メンバー	名前, *FILE	オプション, 定位置 3
AUTH	権限	名前, *CHANGE, *ALL, *USE, *EXCLUDE	オプション
REPLACE	置き換え	*YES, *NO	オプション
PRINT	ソース・リストの印刷	*NO, *YES	オプション
TEXT	テキスト'記述'	文字値, *SRCMBRTXT, *BLANK	オプション

上

### 表示装置ファイル (FILE)

作成するDFU表示装置ファイルの名前を指定します。この名前は、ソースDDSを保管した時にDFUが作成したプログラムの名前と同じでなければなりません。

#### \*CURLIB

作成時に表示装置ファイルを記憶するために現行ライブラリーを使用するためには、\*CURLIBを入力してください。ライブラリー・リストに現行ライブラリーが存在しない場合には、QGPLが使用されます。

#### ライブラリー名

DFU表示装置ファイルを作成したいライブラリーの名前を入力してください。ライブラリーは、元のDFUプログラムを作成したライブラリーと同じライブラリーでなければなりません。



## 表示装置ファイル名

作成する表示装置ファイルの名前を入力してください。表示装置ファイル名は、元の装置ファイルを作成したプログラムの名前と一致していなければなりません。

上

---

## ソース・ファイル (SRCFILE)

ソースDDSが入っているソース・ファイルの名前を指定します。

考えられる値は次の通りです。

### ソース・ファイル名

この表示装置ファイルのDDSが入っているソース・ファイルの名前を入力してください。

**\*LIBL** DFUは指定されたソース・ファイルの検索にユーザーのライブラリー・リストを使用します。

### \*CURLIB

ソース・ファイルを見つけるために現行ライブラリーを使用するためには、\*CURLIBを入力してください。ライブラリー・リストに現行ライブラリー項目が存在しない場合には、QGPLが使用されます。

### ライブラリー名

ソース・ファイルが入っているライブラリーの修飾名を入力してください。

上

---

## ソース・メンバー (SRCMBR)

このDFU表示装置ファイルのDDSが入っているソース・ファイルのメンバーの名前を指定します。

**\*FILE** DFUは、ソース・ファイル・メンバー名としてFILEパラメーターに指定された名前を使用します。

### ソース・ファイル・メンバー名

ソースDDSが入っているメンバーの名前を入力してください。

上

---

## 権限 (AUTH)

表示装置ファイルに対して与える権限を指定します。

考えられる値は次の通りです。

### \*LIBCRTAUT

表示装置ファイルが作成される時に、表示装置ファイルの共通権限を、ライブラリーのCRTAUTパラメーターに指定された値に設定します。表示装置ファイルが作成された後でライブラリーのCRTAUT値が変わっても、新しい値がライブラリーの既存のオブジェクトに影響しません。

### \*CHANGE

所有者に限定されている操作とオブジェクト存在権またはオブジェクト管理権限によって制御される操作を除くすべての操作を他のユーザーが表示装置ファイルに対して行うことができるようにし

ます。他のユーザーはこのファイルに対して基本的な操作を行うことができます。変更権限は、オブジェクト操作権限およびすべてのデータ権限を提供します。

**\*ALL** 所有者に限定されている操作と権限リスト管理権限によって制御される操作を除くすべての操作を他のユーザーが表示装置ファイルに対して行うことができるようにします。他のユーザーはこのファイルの存在の制御、保護の指定、変更などを行うことができます。他のユーザーがこのファイルの所有権を移動することはできません。

**\*USE** 他のユーザーがファイルの読み取りなど、基本的な操作を行うことができるようにします。他のユーザーがファイルを変更することはできません。

**\*EXCLUDE**

他のユーザーが表示装置ファイルにアクセスできないようにします。他のユーザーには何の権限もありません。

**権限リスト名**

表示装置ファイルに使用される権限リストの名前を指定することができます。

上

---

## 置き換え (REPLACE)

同じライブラリーに同じ名前の表示装置ファイルがすでに存在していても新しい表示装置ファイルが作成されるように指定します。

**\*YES** 既存のファイルを置き換え、新しいファイルを再作成するには、\*YESのままにします。

REPLACEに\*NOを指定した場合には、新しいDFU表示装置ファイルの作成を要求する前に元の（調整前の）DFU表示装置ファイルを削除しなければなりません。

**\*NO** 既存のファイルを置き換えたくない場合には、\*NOを入力してください。

上

---

## ソース・リストの印刷 (PRINT)

ファイルの作成時に生成されるリストの印刷を指定します。

**\*NO** このリストを印刷したくない場合には、\*NOのままにしておいてください。

**\*YES** 表示装置ファイルの作成に使用されるソース・ステートメント(DDS)のリストを印刷するためには、\*YESを入力してください。エラーも一緒に印刷されます。

上

---

## テキスト'記述' (TEXT)

表示装置ファイルおよびその機能を簡単に記述した文字ストリングを指定します。

**\*SRCMBRTXT**

DFUは表示装置ファイルの作成に使用されたソース・ファイル・メンバーからテキストを取り出します。

**\*BLANK**

テキストは指定されません。

**‘記述’** 表示装置ファイルを簡単に記述するために最大50文字をアポストロフィで囲んで入力することができます。

上

---

## 例

なし

上

---

## エラー・メッセージ

なし

上

## RPGモジュールの作成 (CRTRPGMOD)

実行可能場所: すべての環境 (\*ALL)

スレッド・セーフ: いいえ

パラメーター

例

エラー・メッセージ

## RPGモジュールの作成

RPGモジュール作成(CRTRPGMOD)コマンドは、RPGソース・コードをコンパイルしてモジュール・オブジェクト(\*MODULE)を作成します。このコマンドはバッチ・モードでも、対話モードでも使用することができます。

上

## パラメーター

キーワード	記述	選択項目	ノート
MODULE	モジュール	修飾オブジェクト名	オプション, 定位置 1
	修飾子 1: モジュール	名前, <u>*CTLSPEC</u>	
	修飾子 2: ライブラリー	名前, <u>*CURLIB</u>	
SRCFILE	ソース・ファイル	修飾オブジェクト名	オプション, 定位置 2
	修飾子 1: ソース・ファイル	名前, <u>QRPGLESRC</u>	
	修飾子 2: ライブラリー	名前, <u>*LIBL</u> , <u>*CURLIB</u>	
SRCMBR	ソース・メンバー	名前, <u>*MODULE</u>	オプション, 定位置 3
SRCSTMF	ソース・ストリーム・ファイル	パス名	オプション
GENLVL	生成重大度レベル	0-20, <u>10</u>	オプション
TEXT	テキスト'記述'	文字値, <u>*SRCMBRTXT</u> , <u>*BLANK</u>	オプション
OPTION	コンパイラー・オプション	値 (最大 20 回の繰り返し): <u>*XREF</u> , <u>*NOXREF</u> , <u>*GEN</u> , <u>*NOGEN</u> , <u>*SECLVL</u> , <u>*NOSECLVL</u> , <u>*SHOWCPY</u> , <u>*NOSHOWCPY</u> , <u>*EXPDDS</u> , <u>*NOEXPDDS</u> , <u>*EXT</u> , <u>*NOEXT</u> , <u>*NOSHOWSKP</u> , <u>*SHOWSKP</u> , <u>*NOSRCSTMT</u> , <u>*SRCSTMT</u> , <u>*DEBUGIO</u> , <u>*NODEBUGIO</u> , <u>*UNREF</u> , <u>*NOUNREF</u> , <u>*NOEVENTF</u> , <u>*EVENTF</u>	オプション
DBGVIEW	デバッグ用ビュー	<u>*STMT</u> , <u>*SOURCE</u> , <u>*LIST</u> , <u>*COPY</u> , <u>*ALL</u> , <u>*NONE</u>	オプション
OUTPUT	出力	<u>*PRINT</u> , <u>*NONE</u>	オプション
OPTIMIZE	最適化レベル	<u>*NONE</u> , <u>*BASIC</u> , <u>*FULL</u>	オプション
INDENT	ソース・リストの字下げ	文字値, <u>*NONE</u>	オプション

キーワード	記述	選択項目	ノート
CVTOPT	タイプ変換オプション	単一値: <u>*NONE</u> その他の値 (最大 4 回の繰り返し): *DATETIME, *GRAPHIC, *VARCHAR, *VARGRAPHIC	オプション
SRTSEQ	ソート順序	単一値: <u>*HEX</u> , *JOB, *JOBRUN, *LANGIDUNQ, *LANGIDSHR その他の値: 修飾オブジェクト名	オプション
	修飾子 1: ソート順序	名前	
	修飾子 2: ライブラリー	名前, <u>*LIBL</u> , *CURLIB	
LANGID	言語識別コード	名前, <u>*JOBRUN</u> , *JOB	オプション
REPLACE	モジュールの置き換え	<u>*YES</u> , *NO	オプション
AUT	権限	名前, <u>*LIBCRTAUT</u> , *ALL, *CHANGE, *USE, *EXCLUDE	オプション
TRUNCNBR	数字の切り捨て	<u>*YES</u> , *NO	オプション
FIXNBR	数値の修正	単一値: <u>*NONE</u> その他の値 (最大 2 回の繰り返し): *ZONED, *INPUTPACKED	オプション
TGTRLS	ターゲット・リリース	単純名, <u>*CURRENT</u> , *PRV	オプション
ALWNULL	ヌル値可能	<u>*NO</u> , *INPUTONLY, *USRCTL, *YES	オプション
DEFINE	条件名の定義	値 (最大 32 回の繰り返し): 単純名, <u>*NONE</u>	オプション
ENBPFCOL	パフォーマンス収集使用可能化	<u>*PEP</u> , *ENTRYEXIT, *FULL	オプション
PRFDTA	プロファイル・データ	<u>*NOCOL</u> , *COL	オプション
BNDDIR	BINDINGディレクトリー	単一値: <u>*NONE</u> その他の値 (最大 50 回の繰り返し): 修飾オブジェクト名	オプション
	修飾子 1: BINDINGディレクトリー	名前	
	修飾子 2: ライブラリー	名前, <u>*LIBL</u> , *CURLIB	
LICOPT	ライセンス内部コード・オプション	文字値, <u>X</u>	オプション
INCDIR	組み込みディレクトリー	値 (最大 32 回の繰り返し): パス名, <u>*NONE</u>	オプション
PGMINFO	Prgmインターフェース情報	要素リスト	オプション
	要素 1: 生成	<u>*NO</u> , *PCML	
	要素 2: 場所	<u>*STMF</u> , *MODULE, *ALL	
INFOTMF	プログラム・インターフェース・ストリーム・ファイル	パス名	オプション
PPGENOPT	プリプロセッサ・オプション	単一値: <u>*NONE</u> , *DFT その他の値 (最大 3 回の繰り返し): *RMVCOMMENT, *NORMVCOMMENT, *EXPINCLUDE, *NOEXPINCLUDE, *SEQSRC, *NOSEQSRC	オプション

キーワード	記述	選択項目	ノート
PPSRCFILE	出力ソース・ファイル	修飾オブジェクト名	オプション
	修飾子 1: 出力ソース・ファイル	名前	
	修飾子 2: ライブラリー	名前, *CURLIB	
PPSRCMBR	出力ソース・メンバー	名前, *MODULE	オプション
PPSRCSTMF	出力ストリーム・ファイル	パス名, *SRCSTMF	オプション

上

## モジュール (MODULE)

作成しているモジュール・オブジェクトのライブラリー名およびモジュール名を指定します。モジュール名およびライブラリー名はサーバーの命名規則に従っていなければなりません。ライブラリーを指定しない場合には、作成されたモジュールは現行ライブラリーに保管されます。

### \*CTLSPEC

コンパイル済みモジュールの名前は、制御仕様書のDFTNAMEキーワードに指定された名前から取られます。制御仕様書にモジュール名を指定せず、ソース・メンバーがデータベース・ファイルからのものである場合には、SRCMBRパラメーターで指定されたメンバー名がモジュール名として使用されます。ソース・メンバーがデータベース・ファイルからのものでない場合には、省略時の値としてモジュール名にRPGMODが使用されます。

### モジュール名

モジュール・オブジェクトの名前を入力してください。

### \*CURLIB

コンパイル済みモジュール・オブジェクトは現行ライブラリーに保管されます。現行ライブラリーが指定されていない場合には、QGPLが使用されます。

### ライブラリー名

コンパイル済みモジュール・オブジェクトが保管されるライブラリーの名前を入力してください。

上

## ソース・ファイル (SRCFILE)

コンパイルされるILE RPGソース・メンバーが入っているソース・ファイルおよびソース・ファイルが保管されているライブラリーの名前を指定します。望ましいソース物理ファイルの長さは112文字で、内12文字は順序番号および日付用で、80文字はコード用で、20文字は注記用です。これは、コンパイル・リストに示されるソースの最大容量です。

### QRPGLESRC

省略時のソース・ファイルQRPGLESRCにコンパイルされるILE RPGソース・メンバーが入っています。

### ソース・ファイル名

コンパイルされるILE RPGソース・メンバーが入っているソース・ファイルの名前を入力してください。これが省略時の値です。

**\*LIBL** ソース・ファイルが保管されているライブラリーを見つけるために、システムはライブラリー・リストを検索します。これが省略時の値です。

### **\*CURLIB**

ソース・ファイルを見つけるために、現行ライブラリーが使用されます。現行ライブラリーが指定されていない場合には、QGPLが使用されます。

#### **ライブラリー名**

ソース・ファイルが入っているライブラリーの名前を入力してください。

上

---

## **ソース・メンバー (SRCMBR)**

コンパイルされるILE RPGソース仕様が入っているソース・ファイルのメンバーの名前を指定します。

### **\*MODULE**

ソース・ファイル・メンバー名としてMODULEパラメーターに指定された名前を使用します。コンパイル済みモジュール・オブジェクトは、ソース・ファイル・メンバーと同じ名前をもちます。MODULEパラメーターにモジュール名が指定されていない場合には、このコマンドは、ソース・ファイルに最初に作成または追加されたメンバーの名前をソース・メンバー名として使用します。

#### **ソース・ファイル・メンバー名**

ILE RPGソース仕様が入っているメンバーの名前を入力してください。

上

---

## **ソース・ストリーム・ファイル (SRCSTMF)**

コンパイルするILE RPGソース・コードが入っているストリーム・ファイルのパス名を指定します。

このパス名は絶対修飾または相対修飾のいずれかとすることができます。絶対パス名は/で始まり、相対パス名は/以外の文字で始まります。

絶対修飾の場合は、パス名は完全です。相対修飾の場合は、パス名はジョブの現行作業ディレクトリーをそのパス名に付加することにより完全になります。

SRCMBRおよびSRCFILEパラメーターをSRCSTMFパラメーターと一緒に指定することはできません。

上

---

## **生成重大度レベル (GENLVL)**

モジュール・オブジェクトの作成を制御します。モジュール・オブジェクトは、コンパイル中に見つかったすべてのエラーの重大度レベルが指定された生成重大度レベル以下である場合に作成されます。

値は0-20でなければなりません。重大度が20を超えるエラーの場合には、モジュール・オブジェクトは生成されません。

**10**      コンパイル時エラーの重大度レベルが10以下であれば、モジュール・オブジェクトが生成されます。これが省略時の値です。

#### **重大度レベル値**

0 - 20の数値を入力してください。



---

## テキスト'記述' (TEXT)

モジュールおよびその機能を簡単に説明するテキストを入力することができます。モジュール情報が表示される時には、常にこのテキストが現れます。

### \*SRCMBRTXT

ソース・メンバーのテキストが使用されます。これが省略時の値です。

### \*BLANK

テキストはありません。

**'記述'** ソース仕様の機能を簡単に記述するテキストを入力してください。テキストは最大50文字にすることができますが、アポストロフィで囲まなければなりません。アポストロフィは50文字のストリングの一部ではありません。プロンプト画面にテキストを入力する場合には、アポストロフィは必要ありません。

---

## コンパイラ・オプション (OPTION)

ソース・メンバーのコンパイル時に使用するオプションを指定します。一部またはすべてのオプションをどのような順序でも指定することができます。各オプションは1つまたは複数のブランク・スペースで区切ってください。オプションが複数回指定された場合には、最後のものが使用されます。

### \*XREF

(適切な場合には) ソース・メンバーの相互参照表を作成します。

### \*NOXREF

相互参照表は作成されません。

**\*GEN** コンパイラによって戻された最高の重大度レベルがGENLVLオプションに指定された重大度を超えない場合にCRTPGMコマンドを使用してバインドできるモジュール・オブジェクトを作成します。

### \*NOGEN

モジュール・オブジェクトを作成しません。

### \*NOSECLVL

第1レベル・メッセージ・テキストの次の行に第2レベル・メッセージ・テキストを印刷しません。

### \*SECLVL

メッセージ要約セクションの第1レベル・メッセージ・テキストの次の行に第2レベル・メッセージ・テキストを印刷します。

### \*SHOWCPY

/COPYコンパイラ・ディレクティブによって含まれるメンバーのソース・レコードを表示します。

### \*NOSHOWCPY

/COPYコンパイラ・ディレクティブによって含まれるメンバーのソース・レコードを表示しません。



ルでは、統合エラー・フィードバックを行うために、このファイルが使用されます。通常、イベント・ファイルは、クライアント・ツール内からモジュールまたはプログラムを生成するときに作成されます。

#### **\*EVENTF**

クライアント・ツールで使用するためのイベント・ファイルを作成します。イベント・ファイルは、作成されたモジュールまたはプログラム・オブジェクトが保管されるライブラリー中のファイルEVFEVENTのメンバーとして作成されます。ファイルEVFEVENT が存在しない場合には、自動的に作成されます。イベント・ファイル・メンバー名は、作成されるオブジェクトの名前と同じです。

クライアント・ツールでは、統合エラー・フィードバックを行うために、このファイルが使用されます。通常、イベント・ファイルは、クライアント・ツール内からモジュールまたはプログラムを生成するときに作成されます。

上

---

## **デバッグ用ビュー (DBGVIEW)**

コンパイル済みモジュール・オブジェクトに使用可能なデバッグのレベルおよびソース・レベルのデバッグに使用可能なソース・ビューを指定します。

#### **\*STMT**

コンパイラ・リストの行番号またはステートメントを使用して、モジュール・オブジェクトをデバッグできるようにします。OPTION(\*NOSRCSTMT)が指定されているときには、行番号がコンパイラ・リストのソース・セクションの左端の桁に示されます。OPTION(\*SRCSTMT)が指定されているときには、ステートメント番号がコンパイラ・リストのソース・セクションの右端の桁に示されます。

#### **\*SOURCE**

コンパイル済みモジュール・オブジェクトのデバッグのためのソース・ビューを生成します。ルートソース・メンバーがDDMファイルである場合には、このビューは使用可能ではありません。また、コンパイル後、プログラムのデバッグを試みる前にソース・メンバーに変更が加えられた場合にも、これらのソース・メンバーのビューが使用できないことがあります。

**\*LIST** コンパイル済みモジュール・オブジェクトのデバッグのためのリスト・ビューを生成します。リスト・ビューに含まれる情報は、OPTIONパラメーターに\*SHOWCPY, \*EXPDDS,および\*SRCSTMTを指定していたかどうかによります。

注: リスト・ビューには、字下げオプションを使用して要求した字下げは示されません。

#### **\*COPY**

コンパイル済みモジュール・オブジェクトのデバッグのためのソース・ビューおよびコピー・ビューを生成します。このオプションのソース・ビューは、\*SOURCEオプションの場合に生成されたソース・ビューと同じです。コピー・ビューは、すべての/COPYソース・メンバーが含まれるデバッグ・ビューです。ルートソース・メンバーがDDMファイルである場合には、これらのビューは使用可能ではありません。また、コンパイル後、プログラムのデバッグを試みる前にソース・メンバーに変更が加えられた場合にも、これらのソース・メンバーのビューが使用できないことがあります。

**\*ALL** コンパイル済みモジュール・オブジェクトのデバッグのためのリスト・ビュー、ソース・ビュー、およびコピー・ビューを生成します。リスト・ビューに含まれる情報は、OPTIONパラメーターに\*SHOWCPY, \*EXPDDS,および\*SRCSTMTを指定していたかどうかによります。

#### **\*NONE**

コンパイル済みモジュール・オブジェクトのデバッグのためのすべてのデバッグ・オプションを使用できないようにします。

上

---

## **出力 (OUTPUT)**

コンパイル・リストが生成されるかどうかを指定します。

#### **\*PRINT**

ILE RPGモジュール・ソースおよびすべてのコンパイル時メッセージからなるコンパイル・リストを作成します。リストに含まれる情報は、OPTIONパラメーターに\*XREF, \*SECLVL, \*SHOWCPY, \*EXPDDS, \*EXT, \*SHOWSKP,および\*SRCSTMTを指定しているかどうかによって異なります。

#### **\*NONE**

コンパイル・リストを生成しません。

上

---

## **最適化レベル (OPTIMIZE)**

モジュールの最適化のレベルがあればそれを指定します。

#### **\*NONE**

生成されたコードは最適化されません。これは、変換時間の面で一番早いものです。デバッグ・モードになっている時には、変数を表示して修正することができます。

#### **\*BASIC**

生成されたコードに対してある種の最適化が実行されます。これにより、デバッグ・モードになっている時に、ユーザー変数は表示できますが、修正することはできません。

#### **\*FULL**

最も効果的なコードを生成する最適化。変換時間は最も遅くなります。提示されている値が現在値でない場合でも、ユーザー変数は変更できませんが、表示することはできます。

上

---

## **ソース・リストの字下げ (INDENT)**

読み易さを増すために、構造化された命令をソース・リストで字下げするかどうかを指定します。構造化された命令の文節をマークするために使用される文字も指定します。

注: ここで要求した字下げは、DBGVIEW(\*LIST)を指定した時に作成されるリスト・デバッグ・ビューで反映されません。

#### **\*NONE**

ソース・リストで構造化された命令は字下げされません。これが省略時の値です。

**文字値** 構造化された命令の文節に対してソース・リストが字下げされます。ステートメントおよび文節の

位置合わせは、選択した文字を使用してマークされます。最大2文字の長さの任意の文字ストリングを選択することができます。文字ストリング中にブランクを使用したい場合には、そのストリングを単一引用符で囲まなければなりません。

注: モジュールにエラーがある場合には、字下げは期待通りに現れないことがあります。

上

---

## タイプ変換オプション (CVTOPT)

ILE RPGコンパイラによる外部記述データベース・ファイルから検索された日付、時刻、タイム・スタンプ、図形データ・タイプ、および可変長データ・タイプの操作方法を指定します。

### **\*NONE**

可変長データベースのデータ・タイプを無視し、RPG固有の日付、時刻、タイム・スタンプ、および図形データ・タイプを使用します。

### **\*DATETIME**

日付、時刻、およびタイム・スタンプのデータベース・データ・タイプが固定長文字フィールドとして宣言されることを指定します。

### **\*GRAPHIC**

2バイト文字セット(DBCS)図形データ・タイプが固定長文字フィールドとして宣言されることを指定します。

### **\*VARCHAR**

可変長文字データ・タイプが固定長文字フィールドとして宣言されることを指定します。

### **\*VARGRAPHIC**

可変長2バイト文字セット(DBCS)図形データ・タイプが固定長文字フィールドとして宣言されることを指定します。

上

---

## ソート順序 (SRTSEQ)

ILE RPGソース・プログラムで使用されるソート順序テーブルを指定します。

**\*HEX** ソート順序テーブルは使用されません。

**\*JOB** モジュールが作成される時にジョブのSRTSEQ値を使用します。

### **\*JOBRUN**

モジュールが実行される時（バインド後）にジョブのSRTSEQ値を使用します。

### **\*LANGIDUNQ**

固有の重みテーブルを使用します。この特殊値は、正しいソート順序テーブルを選択するためにLANGIDパラメーターと一緒に使用されます。

### **\*LANGIDSHR**

共用重みテーブルを使用します。この特殊値は、正しいソート順序テーブルを選択するためにLANGIDパラメーターと一緒に使用されます。

### **ソート・テーブル名**

ソート順序テーブルの名前を入力してください。

**\*LIBL** ソート順序テーブルが保管されているライブラリーを見つけるために、システムはライブラリー・リストを検索します。

**\*CURLIB**

ソート順序テーブルを見つけるために、現行ライブラリーが使用されます。現行ライブラリーが指定されていない場合には、QGPLが使用されます。

**ライブラリー名**

ソート順序テーブルが保管されているライブラリーの名前を入力してください。

上

---

## 言語識別コード (LANGID)

ソート順序が\*LANGIDUNQおよび\*LANGIDSHRである時に使用される言語識別コードを指定します。LANGIDパラメーターは、ソート順序テーブルを選択するためにSRTSEQ パラメーターと一緒に使用されます。

**\*JOBRUN**

RPGモジュールが実行される時（バインド後）にジョブと関連したLANGID値を使用します。

**\*JOB** RPGモジュールが作成される時にジョブと関連したLANGID値を使用します。

**言語識別コード**

指定された言語識別コードを使用します。（たとえば、フランス語の場合にはFRA で、ドイツ語の場合にはDEU。）

上

---

## モジュールの置き換え (REPLACE)

指定されたライブラリーに同じ名前のモジュールがすでに存在している時に、新しいモジュール・オブジェクトが作成されるかどうかを指定します。

**\*YES** 指定されたライブラリーに新しいモジュール・オブジェクト・プログラムが作成されます。指定されたライブラリーの同じ名前の既存のモジュール・オブジェクトはライブラリーQRPLIBに移動されます。

**\*NO** 指定されたライブラリーに同じ名前のモジュール・オブジェクトがすでに存在している場合には、新しいモジュール・オブジェクトは作成されません。

上

---

## 権限 (AUT)

このオブジェクトに対する特定権限をもっていないユーザー、権限リスト上にないユーザー、および所属するユーザー・グループがこのオブジェクトに対する特定権限をもっていないユーザーに与えられる権限を指定します。モジュールの作成後に、CLコマンドのオブジェクト権限認可(GRTOBJAUT)またはオブジェクト権限取り消し(RVKOBJAUT)を使用してすべてまたは指定したユーザーの権限を変更することができます。これらのコマンドの詳細については、i5/OS Information Center (<http://www.ibm.com/systems/i/infocenter/>)の「プログラミング」カテゴリーに含まれる制御言語(CL) トピック・コレクションを参照してください。



## **\*LIBCRTAUT**

オブジェクトの共通権限は、ターゲット・ライブラリー（オブジェクトが入っているライブラリー）のCRTAUTキーワードから取られます。値は、オブジェクトの作成時に決定されます。作成後にライブラリーのCRTAUT値が変わった場合には、新しい値は既存のオブジェクトに反映されません。

**\*ALL** 所有者に限定されているか、あるいは権限リスト管理権限によって制御されるものを除いてモジュール・オブジェクトに対するすべての命令の権限。ユーザーはモジュール・オブジェクトの存在を制御し、オブジェクトに対するこの機密保護を指定し、オブジェクトを変更し、オブジェクトに対する基本機能を実行することができますが、所有権を移すことはできません。

## **\*CHANGE**

所有者に限定されているか、あるいはオブジェクト権とオブジェクト管理権によって制御されているものを除いて、モジュール・オブジェクトに対するすべてのデータ権限およびすべての命令を実行する権限を提供します。ユーザーはオブジェクトを変更し、オブジェクトに対して基本機能を実行することができます。

**\*USE** オブジェクト操作権と読み取り権限、およびプログラムへのバインドなどのモジュール・オブジェクトに対する基本操作の権限を提供します。ユーザーはオブジェクトを変更することができます。

注: プログラムの変数の定様式ダンプを取るには、ユーザーにプログラムの\*USE権限が必要です。変数をダンプするには、プログラムに、識別可能な情報も必要です。

一部のユーザーに変数をダンプさせたくない場合には、そのユーザーにモジュールを含むプログラムまたはサービス・プログラムに対する\*OBJOPR権限と\*EXECUTE権限だけを割り当ててください。これにより、そのユーザーはプログラムを呼び出すことはできますが、その変数をダンプすることはできません。作成済みプログラムまたはサービス・プログラムの権限を変更するには、EDTOBJAUT, GRTOBJAUT,またはRVKOBJAUTを使用してください。

どのユーザーにも変数をダンプさせたくない場合には、モジュールではモジュール変更(CHGMOD)を使用し、モジュールを含むプログラムではプログラム変更(CHGPGM)またはサービス・プログラム変更(CHGSRVPGM)を使用して、識別可能な情報を除去してください。

## **\*EXCLUDE**

ユーザーは、オブジェクトをアクセスできません。

## **権限リスト名**

ユーザーおよびモジュールが追加される権限の権限リストの名前を入力してください。モジュール・オブジェクトはこの権限リストによって保護され、モジュール・オブジェクトの共通権限は\*AUTLにセットされます。CRTRPGMODコマンドを出す時には、この権限リストがシステム上に存在していなければなりません。

注: システムの機密保護要件を反映させるためには、AUTパラメーターを使用してください。使用可能な機密保護機能は、System i機密保護解説書(SD88-5027)で詳しく説明されています。

上

---

## **数字の切り捨て (TRUNCNBR)**

プログラムの実行中に数値オーバーフローが起こった場合に、結果のフィールドに切り捨て値を入れるか、それともエラーを生成するかを指定します。



注: TRUNCNBRオプションは式の中で行なわれる演算には適用されません。(式は拡張演算項目 2 フィールドにあります。) これらの演算でオーバーフローが起こった場合には、常にエラーが発生します。

**\*YES** 数値オーバーフローを無視して、結果のフィールドに切り捨て値を入れます。

**\*NO** 数値オーバーフローが検出された時に、実行時エラーが生成されます。

上

---

## 数値の修正 (FIXNBR)

正しくない10進数データをコンパイラーによって訂正するかどうかを指定します。

### **\*NONE**

正しくない10進数データが使用された場合に、実行時に10進数エラーとなることを指示します。

### **\*ZONED**

正しくないゾーン10進数データは、パック・データへの変換時にコンパイラーによって訂正されます。数値フィールドのブランクはゼロとして扱われます。各桁は妥当性検査されます。桁が有効でない場合には、その桁はゼロで置き換えられます。符号が有効でない場合には、その符号は16進数'F'の正符号に強制的に変更されます。符号が有効である場合には、必要に応じ16進数'F'の正符号または16進数'D'の負符号に変更されます。結果のパック・データが正しくない場合には、そのデータは訂正されません。

### **\*INPUTPACKED**

正しくないパック10進数データが入力仕様の処理中に出てきた場合に、内部変数がゼロに設定されることを指示します。

上

---

## ターゲット・リリース (TGTRLS)

作成されるオブジェクトを使用するオペレーティング・システムのリリースを指定します。\*CURRENTおよび\*PRV値の場合の例では、ターゲット・リリース 値を指定する時には、形式VXRXXMXを使用してリリースを指定します。VX はバージョン、RXはリリース、MXはモディフィケーション・レベルです。たとえば、V2R3M0は、バージョン 2、リリース 3、モディフィケーション・レベル 0 です。

このパラメーターに対する有効な値はリリースごとに変わります。指定できる値は次の通りです。

### **\*CURRENT**

オブジェクトは、現在ユーザー・システムで実行中のオペレーティング・システムのリリースで使用されます。たとえば、システムでV2R3M5を実行中の場合には、\*CURRENTはV2R3M5が導入されているシステムでオブジェクトを使用しようとしていることを意味します。また、オペレーティング・システムの後続のリリースが導入されたシステム上で、このオブジェクトを使用することもできます。

注: システムでV2R3M5が実行中で、オブジェクトをV2R3M0が導入されたシステムで使用しようとする場合には、TGTRLS(\*CURRENT)ではなくTGTRLS(V2R3M0)を指定してください。

**\*PRV** オブジェクトは、オペレーティング・システムの前のリリース、モディフィケーション・レベル 0 で使用されます。たとえばシステムでV2R3M5を実行中の場合には、\*PRVはV2R2M0が導入されて

いるシステムでオブジェクトを使用しようとしていることを意味します。また、オペレーティング・システムの後続のリリースが導入されたシステム上で、このオブジェクトを使用することもできます。

### ターゲット・リリース

リリースをVXRXXMXの形式で指定してください。オブジェクトは、指定したリリースのシステムまたはオペレーティング・システムの後続のリリースが導入されたシステムで使用することができます。

有効な値は、現在のバージョン、リリース、およびモディフィケーション・レベルによって異なり、新しいリリースごとに変わります。このコマンドでサポートされている最も古いリリース・レベルよりも前の ターゲット・リリース を指定した場合には、エラー・メッセージが出され、サポートされる最も古いリリースが表示されます。

注: コマンドの現在のバージョンは、コマンドの前のリリースで使用可能でないオプションをサポートすることがあります。前のリリースで使用されるオブジェクトを作成するためにコマンドが使用される場合には、コマンドはそのリリースに該当するコンパイラーで処理され、サポートされないオプションはどれも認識されません。コンパイラーは、処理に使用可能でないオプションについて警告を出すとは限りません。

上

---

## ヌル値可能 (ALWNULL)

ILE RPGモジュールが、外部記述データベース・ファイルから、ヌル値可能フィールドの入ったレコードをどのように使用できるようにするかを指定します。

**\*NO** ILE RPGモジュールが外部記述ファイルからのヌル値フィールドをもつレコードを処理しないことを指定します。ヌル値が入っているレコードを検索しようとした場合には、レコード中のデータはILE RPGモジュールに対してアクセス不能となり、データ・マッピング・エラーが起こります。

### \*INPUTONLY

ILE RPGモジュールが、外部記述入力専用データベース・ファイルから、ヌル値の入ったヌル値可能フィールドをもつレコードを正常に読み取ることができることを指定します。ヌル値の入ったレコードを検索する時には、データ・マッピング・エラーは起こらず、データベースの省略時の値がヌル値の入った任意のフィールドに入れられます。モジュールは以下を実行することはできません。

- ヌル値可能キー・フィールドの使用
- ヌル値可能フィールドの入ったレコードの作成または更新
- モジュールの実行中に、ヌル値可能フィールドが実際にヌルであるかどうかを判別すること
- ヌル値可能フィールドをヌルに設定すること

### \*USRCTL

ILE RPGモジュールが、外部記述データベース・ファイルから、ヌル値の入ったレコードを読み取り、書き出し、更新できることを指定します。ヌル・キーのあるレコードはキー順操作を使用して検索することができます。モジュールは、ヌル値可能フィールドが実際にヌルであるかどうかを判別することができ、出力または更新用に、ヌル値可能フィールドをヌルに設定することができます。プログラマーは、ヌル値の入ったフィールドがモジュール内で正しく使用されていることを確認する責任があります。

**\*YES** \*INPUTONLYと同じ。

---

## 条件名の定義 (DEFINE)

コンパイルの開始前に定義される条件名を指定します。パラメーターDEFINE(条件名)を使用することは、ソース・ファイルの最初の行に/DEFINE条件名ディレクティブをエンコードすることと同じです。

### \*NONE

条件名は定義されません。これが省略時の値です。

**名前** 最大32個までの条件名を指定することができます。各名前の長さは50桁までとすることができます。条件名はコンパイルの開始時に定義されるものと見なされます。

---

## パフォーマンス収集使用可能化 (ENBPFRCOL)

パフォーマンス収集を使用可能にするかどうかを指定します。

**\*PEP** パフォーマンス統計は、プログラム入り口プロシーチャーの入り口および出口でのみ収集されます。これは、プログラムの実際のプログラム入り口プロシーチャーに対して適用され、プログラム内のモジュールのメイン・プロシーチャーには適用されません。これが省略時の値です。

### **\*ENTRYEXIT**

パフォーマンス統計はモジュールのすべてのプロシーチャーの入り口および出口に収集されます。

### **\*FULL**

パフォーマンス統計はすべてのプロシーチャーの入り口および出口に収集されます。また、統計は、外部プロシーチャーに対する各呼び出しの前後に収集されます。

---

## プロファイル・データ (PRFDTA)

プログラム・プロファイル・データ属性をモジュールに指定します。プログラム・プロファイルは、統計データ（プロファイル・データ）に基づいてプロシーチャーおよびプロシーチャー内のコードを再順序づけるために使用される拡張最適化手法です。

### \*NOCOL

このモジュールはプロファイル・データを収集できません。これが省略時の値です。

**\*COL** このモジュールはプロファイル・データを収集できます。**\*COL**は、モジュールの最適化レベルが**\*FULL**の時と**\*CURRENT**のターゲット・リリースでコンパイルしている時にだけ指定することができます。

---

## BINDINGディレクトリー (BNDDIR)

モジュールの記号分析解決で使用するバインディング・ディレクトリーのリストを指定します。バインディング・ディレクトリーの検索は、モジュールがCRTPGMまたはCRTSRVPGM時刻にバインドされる時に行われます。

### \*NONE

バインディング・ディレクトリーは指定されません。

### バインディング・ディレクトリー名

記号分析解決で使用するバインディング・ディレクトリーの名前を指定してください。ディレクトリー名は、次の1つのライブラリー値で修飾することができます。

**\*LIBL** バインディング・ディレクトリーが保管されているライブラリーを見つけるために、システムはライブラリー・リストを検索します。これが省略時の値です。

### **\*CURLIB**

ジョブの現行ライブラリーは、コンパイル時に分析解決されます。次に、モジュールをバインドする時に、バインディング・ディレクトリーの検索に使用されます。ジョブの現行ライブラリーとしてライブラリーが指定されていない場合には、ライブラリーQGPLが使用されます。

### ライブラリー名

検索するライブラリーの名前を指定してください。

上

---

## ライセンス内部コード・オプション (LICOPT)

1つまたは複数のライセンス内部コード・コンパイル時オプションを指定します。このパラメーターは、個別のコンパイル時オプションを選択できるようにするもので、選択したそれぞれのタイプのコンパイラー・オプションの潜在的な利点と欠点を理解した高度のプログラマーを対象にしています。

上

---

## 組み込みディレクトリー (INCDIR)

コピー・ファイルを検索するためにコンパイラーにより使用される検索パスに追加する1つ以上のディレクトリーを指定します。ソース・プログラム中のコピー・ファイルを解決できない場合に、コンパイラーはここに指定されたディレクトリーを検索します。

### \*NONE

ユーザー・ディレクトリーでコピー・ファイルは検索されません。省略時の値により、ソース・ディレクトリーは検索されます。

### ’ディレクトリー’

コピー・ファイルを検索するディレクトリーを最大32個まで指定してください。指定されたディレクトリーの他に、ソース・ディレクトリーからもコピー・ファイルが検索されます。

上

---

## Prgmインターフェース情報 (PGMINFO)

プログラム・インターフェース情報を生成するかどうか、およびどこに生成するのかを制御するオプションを指定します。

**生成** プログラム・インターフェース情報を生成するかどうかを指定します。指定できる値は次の通りです。

**\*NO** このオプションは、プログラム・インターフェース情報を生成しない省略時の値を指定します。

**\*PCML**

PCML（プログラム呼び出しマークアップ言語）を生成することを指定します。生成されたPCMLにより、JAVAメソッドは、より少ないJAVAコードを使用してこのRPGモジュールでプロシーチャーをより簡単に呼び出すことができますようになります。

**場所** 生成パラメーターが\*PCMLの場合、生成されたプログラム情報の場所を指定します。指定できる値は次の通りです。

**\*STMF**

プログラム情報をストリーム・ファイルに生成するように指定します。生成される情報が入るストリーム・ファイルの名前はINFOSTMFオプションに指定してください。

**\*MODULE**

プログラム情報をRPGモジュールに保管するように指定します。

**\*ALL** プログラム情報をストリーム・ファイルに生成し、モジュールに保管するように指定します。生成される情報が入るストリーム・ファイルの名前はINFOSTMFオプションに指定してください。

上

---

## プログラム・インターフェース・ストリーム・ファイル (INFOSTMF)

PGMINFOオプションに指定された生成されたプログラム・インターフェース情報が入るストリーム・ファイルのパス名を指定します。

このパス名は絶対修飾または相対修飾のいずれかとすることができます。絶対パス名は'/'で始まり、相対パス名は'/'以外の文字で始まります。

絶対修飾の場合は、パス名は完全です。相対修飾の場合は、パス名はジョブの現行作業ディレクトリーをそのパス名に付加することにより完全になります。

このパラメーターを指定できるのは、PGMINFOパラメーターに\*NO以外の値が指定されている場合だけです。

上

---

## プリプロセッサ・オプション (PPGENOPT)

ソース・コードのコンパイル時に使用するプリプロセッサ生成オプションを指定します。

考えられるオプションは次の通りです。

### **\*NONE**

ソース・ファイルに対してコンパイラ全体を実行します。プリプロセッサ出力をファイルにコピーしません。

**\*DFT** 入力ソースに対してプリプロセッサを実行します。プリプロセッサ出力の生成のオプションとして、**\*RMVCOMMENT**、**\*EXPINCLUDE**および**\*NOEQSRC**が使用されます。出力ソース・ファイルおよびメンバーを指定するには、**PPSRCFILE**および**PPSRCMBR**を使用し、プリプロセッサ出力を含めるストリーム・ファイルを指定するには、**PPSRCSTMF**を使用してください。

### **\*RMVCOMMENT**

プリプロセス中にコメント、ブランク行、およびほとんどのディレクティブを除去します。RPG仕様のみ、および仕様の正しい変換処理に必要なすべてのディレクティブは保存します。

### **\*NORMVCOMMENT**

プリプロセス中にコメント、ブランク行、およびリスト制御ディレクティブ（たとえば、**/EJECT**、**/TITLE**）を保存します。プリプロセス中にソース制御ディレクティブ（たとえば、**/COPY**、**/IF**）をコメントに変換します。

### **\*EXPINCLUDE**

生成された出力ファイルの**/INCLUDE**ディレクティブを展開します。

### **\*NOEXPINCLUDE**

**/INCLUDE**ディレクティブは生成された出力ファイルに変更されずに配置されます。

注: **/COPY**ディレクティブは常に展開されます。

### **\*SEQSRC**

**PPSRCFILE**を指定した場合は、生成された出力メンバーには順次に順序番号が付けられ、000001で始まり、000001ずつ増分されます。

### **\*NOSEQSRC**

**PPSRCFILE**を指定した場合は、生成された出力メンバーは、プリプロセッサが読み取った元のソースと同じ順序番号を持ちます。

上

---

## 出力ソース・ファイル (PPSRCFILE)

プリプロセッサ出力のソース・ファイル名およびライブラリーを指定します。

### ソース・ファイル名

プリプロセッサ出力のソース・ファイルの名前を指定します。

考えられるライブラリー値は次の通りです。

### **\*CURLIB**

プリプロセッサ出力は現行ライブラリー中に作成されます。ジョブに現行ライブラリーがない場合は、プリプロセッサ出力ファイルは**QGPL**ライブラリー中に作成されます。

### ライブラリー名

プリプロセッサ出力のライブラリーの名前を指定します。



---

## 出力ソース・メンバー (PPSRCMBR)

プリプロセッサ出力のソース・ファイル・メンバーの名前を指定します。

### **\*MODULE**

MODULEパラメーターに提供した名前は、プリプロセッサ出力メンバー名として使用されます。

### メンバー名

プリプロセッサ出力のメンバーの名前を指定します。

---

## 出力ストリーム・ファイル (PPSRCSTMF)

プリプロセッサ出力のストリーム・ファイルのパス名を指定します。

### **\*SRCSTMF**

SRCSTMFパラメーターに提供したパス名は、プリプロセッサ出力パス名として使用されます。  
ファイルは拡張子'.I'を持ちます。

### 'パス名'

プリプロセッサ出力ストリーム・ファイルのパス名を指定します。

このパス名は絶対修飾または相対修飾のいずれかとすることができます。絶対パス名は'/'で始まり、相対パス名は'/'以外の文字で始まります。

絶対修飾の場合は、パス名は完全です。相対修飾の場合は、パス名はジョブの現行作業ディレクトリーをそのパス名に付加することにより完全になります。

---

## 例

### 例1: ソース・モジュールのモジュール・オブジェクトへのコンパイル

```
CRTRPGMOD  MODULE(MYLIB/XMPLE1)
           SRCFILE(MYLIB/QRPGLESRC) SRCMBR(XMPLE1)
           OUTPUT(*PRINT) TEXT('MY RPG IV MODULE')
```

このコマンドはILE RPGのコンパイラーを呼び出して、XMPLE1の名前のモジュールを作成します。ソース・モジュールは、ライブラリーMYLIB中のソース・ファイルQRPGLESRCのメンバーXMPLE1中にあります。コンパイラー・リストが作成されます。

---

## エラー・メッセージ

### \*ESCAPE メッセージ



**RNS9309**

コンパイルは正常に実行されなかった。モジュール&1がライブラリー&2に作成されませんでした。

上



## RPG/400プログラムの作成 (CRTRPGPGM)

実行可能場所: すべての環境 (\*ALL)

スレッド・セーフ: いいえ

パラメーター

例

エラー・メッセージ

RPG/400コンパイラーを開始するためには、RPGプログラム作成(CRTRPGPGM)コマンドが使用されます。

上

### パラメーター

キーワード	記述	選択項目	ノーツ
PGM	プログラム	修飾オブジェクト名	オプション, 定位置 1
	修飾子 1: プログラム	名前, <u>*CTLSPEC</u>	
	修飾子 2: ライブラリー	名前, <u>*CURLIB</u>	
SRCFILE	ソース・ファイル	修飾オブジェクト名	オプション, 定位置 2
	修飾子 1: ソース・ファイル	名前, <u>QRPGSRC</u>	
	修飾子 2: ライブラリー	名前, <u>*LIBL</u> , <u>*CURLIB</u>	
SRCMBR	ソース・メンバー	名前, <u>*PGM</u>	オプション, 定位置 3
GENLVL	生成重大度レベル	0-99, <u>9</u>	オプション
TEXT	テキスト'記述'	文字値, <u>*SRCMBRTXT</u> , <u>*BLANK</u>	オプション
OPTION	ソース・リスト・オプション	値 (最大 14 回の繰り返し): <u>*SOURCE</u> , <u>*NOSOURCE</u> , <u>*SRC</u> , <u>*NOSRC</u> , <u>*XREF</u> , <u>*NOXREF</u> , <u>*GEN</u> , <u>*NOGEN</u> , <u>*DUMP</u> , <u>*NODUMP</u> , <u>*SECLVL</u> , <u>*NOSECLVL</u> , <u>*SRCDDBG</u> , <u>*NOSRCDBG</u> , <u>*LSTDBG</u> , <u>*NOLSTDBG</u>	オプション
GENOPT	生成オプション	値 (最大 10 回の繰り返し): <u>*LIST</u> , <u>*NOLIST</u> , <u>*XREF</u> , <u>*NOXREF</u> , <u>*ATR</u> , <u>*NOATR</u> , <u>*DUMP</u> , <u>*NODUMP</u> , <u>*PATCH</u> , <u>*NOPATCH</u> , <u>*OPTIMIZE</u> , <u>*NOOPTIMIZE</u>	オプション
INDENT	ソース・リストの字下げ	文字値, <u>*NONE</u>	オプション
CVTOPT	タイプ変換オプション	単一値: <u>*NONE</u> その他の値 (最大 3 回の繰り返し): <u>*DATETIME</u> , <u>*VARCHAR</u> , <u>*GRAPHIC</u>	オプション

キーワード	記述	選択項目	ノート
SRTSEQ	ソート順序	単一値: <u>*HEX</u> , *JOB, *JOBRUN, *LANGIDUNQ, *LANGIDSHR その他の値: 修飾オブジェクト 名	オプション
	修飾子 1: ソート順序	名前	
	修飾子 2: ライブラリー	名前, <u>*LIBL</u> , *CURLIB	
LANGID	言語ID	名前, <u>*JOBRUN</u> , *JOB	オプション
SAAFLAG	SAAフラグ付け	<u>*NOFLAG</u> , *FLAG	オプション
PRTFILE	印刷ファイル	修飾オブジェクト名	オプション
	修飾子 1: 印刷ファイル	名前, <u>*QSYSPRT</u>	
	修飾子 2: ライブラリー	名前, <u>*LIBL</u> , *CURLIB	
REPLACE	プログラムの置き換え	<u>*YES</u> , *NO	オプション
TGTRLS	ターゲット・リリース	単純名, <u>*CURRENT</u> , *PRV	オプション
USRPRF	ユーザー・プロファイル	<u>*USER</u> , *OWNER	オプション
AUT	権限	名前, <u>*LIBCRTAUT</u> , *ALL, *CHANGE, *USE, *EXCLUDE	オプション
PHSTRC	フェーズ・トレース	<u>*NO</u> , *YES	オプション
ITDUMP	中間テキスト・ダンプ	文字値, <u>*NONE</u>	オプション
SNPDUMP	スナップ・ダンプ	文字値, <u>*NONE</u>	オプション
CODELIST	コード・リスト	文字値, <u>*NONE</u> , *ALL	オプション
IGNDECERR	10進データ・エラーの無視	<u>*NO</u> , *YES	オプション
ALWNULL	ヌル値使用可能	<u>*NO</u> , *YES	オプション

上

## プログラム (PGM)

コンパイル済みRPGプログラムのプログラム名およびライブラリーを指定します。

### \*CTLSPEC

制御仕様の75-80桁目で指示されたプログラム名。

### プログラム名

プログラムを識別する名前を入力してください。

### \*CURLIB

現行ライブラリーが使用されます。 現行ライブラリーを指定していない場合には、QGPLが使用されます。

### ライブラリー名

コンパイル済みプログラムを保管するライブラリーの名前を入力してください。

上

## ソース・ファイル (SRCFILE)

ソース・プログラムが入っているソース・ファイルの名前を指定します。

## **QRPGSRC**

省略時のソース・ファイルQRPGSRCには、コンパイルされるRPGプログラムが入っています。

### **ソース・ファイル名**

コンパイルされるRPGソース・プログラムが入っているソース・ファイル名を入力します。

**\*LIBL** システムはライブラリー・リストを検索して、ソース・ファイルが入っているライブラリーを見つけてます。

### **\*CURLIB**

現行ライブラリーが使用されます。 現行ライブラリーを指定していない場合には、QGPLが使用されます。

### **ライブラリー名**

ソース・ファイルが入っているライブラリーの名前を入力してください。

上

---

## **ソース・メンバー (SRCMBR)**

ソース・ファイルのメンバーの名前を指定します。

**\*PGM** ソース・ファイル・メンバー名としてPGMパラメーターで指定した名前。

### **ソース・ファイル・メンバー名**

ソース・プログラムが入っているメンバーの名前を入力してください。

上

---

## **生成重大度レベル (GENLVL)**

プログラム・オブジェクトの作成を停止する診断メッセージの重大度レベルを指定します。

**9** 省略時の重大度レベルは9です。

### **重大度レベル値**

01-50の2桁の数字を入力してください。

上

---

## **テキスト'記述' (TEXT)**

プログラム機能の説明に使用するテキストを指定します。

### **\*SRCMBRTXT**

ソース・ファイル・メンバーのテキスト記述を使用します。

### **\*BLANK**

テキストはありません。

上

---

## ソース・リスト・オプション (OPTION)

ソース・プログラムをコンパイルする時に使用するオプションを指定します。

### \*SOURCEまたは\*SRC

コンパイラーがソース・リストを提供します。

### \*NOSOURCEまたは\*NOSRC

コンパイラーはソース・リストを提供しません。

### \*XREF

コンパイラーが相互参照表を提供します。

### \*NOXREF

コンパイラーは相互参照表を提供しません。

\*GEN プログラムのコンパイル後に実行可能なプログラム・オブジェクトが作成されます。

### \*NOGEN

プログラム・オブジェクトは作成されません。

### \*NODUMP

エラーが起こった時にプログラム・テンプレートを印刷しません。

### \*DUMP

エラーが起こった時にプログラム・テンプレートを印刷します。

### \*NOSECLVL

メッセージ・テキストは印刷されません。

### \*SECLVL

メッセージ・テキストが印刷されます。

### \*NOSRCDBG

ソース・レベルのデバッグ情報は作成されません。ソース・レベルのエラー情報は、\*LSTDBGを指定しない限り作成されません。

### \*SRCDBG

コンパイラーは、クライアント・ツールで使用するためのソース・レベルのエラー情報およびソース・レベルのデバッグ情報を作成します。また、システム・デバッガー(STRDBG OPMSRC(\*YES))を使用してOPMプログラムとILEプログラムを同時にデバッグしたい場合には、ソース・レベルまたはリスト・レベルのデバッグ情報が必要です。

\*SRCDBGと\*LSTDBGと一緒に指定することはできません。どちらか一方を指定してください。

### \*NOLSTDBG

リスト・ビューまたはリスト・レベルのデバッグ情報は作成されません。ソース・レベルのエラー情報は、\*SRCDBGを指定しない限り作成されません。

### \*LSTDBG

コンパイラーは、クライアント・ツールで使用するためのリスト・ビュー、ソース・レベルのエラー情報、およびリスト・レベルのデバッグ情報を作成します。また、システム・デバッガー(STRDBG OPMSRC(\*YES))を使用してOPMプログラムとILEプログラムを同時にデバッグしたい場合には、ソース・レベルまたはリスト・レベルのデバッグ情報が必要です。

\*SRCDBGと\*LSTDBGと一緒に指定することはできません。どちらか一方を指定してください。

上

---

## 生成オプション (GENOPT)

オブジェクト・コードの作成に使用するオプションを指定してください。

### \*NOLIST

プログラムの中間表現(IRP)リストを作成しません。

**\*LIST** プログラムの中間表現(IRP)をリストします。

### \*NOXREF

プログラムの中間表現(IRP)の相互参照リストを作成しません。

### **\*XREF**

プログラムの中間表現(IRP)に定義されたすべてのオブジェクトの相互参照リストを作成します。

### \*NOATR

属性リストを作成しません。

**\*ATR** プログラムの中間表現(IRP)ソース・プログラムの属性をリストします。

### \*NODUMP

エラーが起こった時にプログラム・テンプレートを印刷しません。

### **\*DUMP**

プログラム・テンプレートを印刷します。

### \*NOPATCH

コンパイル済みプログラムにプログラム・パッチ域を予約しません。

### **\*PATCH**

コンパイル済みプログラムにプログラム・パッチ域のスペースを予約します。

### \*NOOPTIMIZE

コンパイラはプログラムの最適化を実行しません。

### **\*OPTIMIZE**

コンパイラは処理効率の高いプログラムを作成します。

上

---

## ソース・リストの字下げ (INDENT)

DOステートメントおよびIF-ELSE文節は、読みやすいように字下げを指定します。また、対応するDO-ENDDOの対とIF-ELSEの対の結合に使用される文字も指定します。

### \*NONE

DOステートメントまたはIF-ELSE文節の中では、リストは字下げされません。

### 文字ストリング

対応するネスト・レベルを結合するためには所定の文字ストリングを使用します。

上

---

## タイプ変換オプション (CVTOPT)

日付、時刻、およびタイム・スタンプ・データベースのデータ・タイプ、および外部記述ファイルから検索される可変長データ・タイプをRPG/400コンパイラで処理する方法を指定します。



#### **\*NONE**

日付、時刻、タイム・スタンプ、および可変長データベースのデータ・タイプは無視されて、RPG/400プログラムでアクセスすることはできません。

#### **\*DATETIME**

日付、時刻、タイム・スタンプのデータベースのデータ・タイプは固定長文字フィールドとして宣言されて、RPG/400プログラムでアクセスすることができます。

#### **\*VARCHAR**

可変長データベースのデータ・タイプは固定長文字フィールドとして宣言されて、RPG/400プログラムでアクセスすることができます。

#### **\*GRAPHIC**

DBCSグラフィック・データ・タイプは固定長の文字フィールドとして宣言され、RPG/400プログラムでアクセス可能です。

注: プログラム中で可変長DBCSグラフィック・データ・タイプを宣言する必要がある場合には、

\*VARCHARおよび\*GRAPHICの両方のパラメーターを選択してください。

上

---

## **ソート順序 (SRTSEQ)**

使用するソート順序テーブルを指定します。

注: CRTRPGPGMまたはCRTRPTPGMコマンドのSRTSEQおよびLANGIDパラメーターにコーディングされた値を使用するためには、制御仕様の代替照合順序フィールドにDを指定しなければなりません。代替照合順序はコンパイル時または実行時のいずれかにシステムから検索されます。Dオプションを指定すると、代替照合順序は以下に影響を与えます。すなわち、すべての文字比較操作、文字テーブルおよび配列のLOKUPおよびSORTA、および文字のコンパイル時データおよび実行時前配列とテーブルの順序検査に影響を与えます。実行時に代替照合順序が検索される場合には、コンパイル時データの順序検査は実行時まで延期されます。

**\*HEX** ソート順序を決定するために、文字の16進数値を使用します。これが省略時の値です。

**\*JOB** RPGプログラムの作成時に、ジョブに対応したSRTSEQ値を使用します。

#### **\*JOBRUN**

RPGプログラムの実行時に、ジョブに対応したSRTSEQ値を使用します。

#### **\*LANGIDUNQ**

固有の重みづけテーブルを使用します。この特殊値は、ソート順序テーブルの選択で、LANGIDパラメーターと一緒に使用されます。

#### **\*LANGIDSHR**

共用の重みづけテーブルを使用します。この特殊値は、ソート順序テーブルの選択で、LANGIDパラメーターと一緒に使用されます。

#### **ソート順序テーブル名**

ソート順序テーブルの名前を入力してください。

**\*LIBL** コンパイラーは、ライブラリー・リストを検索してソート順序テーブルが入っているライブラリーを見つけます。これが省略時の値です。

### **\*CURLIB**

ソート順序テーブルを見つけるために、現行ライブラリーが検索されます。現行ライブラリーが指定されていない場合には、QGPLが使用されます。

### **ライブラリー名**

ソート順序テーブルが入っているライブラリーの名前を入力してください。

上

---

## **言語ID (LANGID)**

ソート順序が\*LANGIDUNQまたは\*LANGIDSHRの時に使用する言語IDを指定します。LANGIDパラメーターは、ソート順序テーブルの選択で、SRTSEQパラメーターと一緒に使用されます。

### **\*JOB RUN**

RPGプログラムの実行時に、ジョブに対応したLANGID値を使用します。これが省略時の値です。

**\*JOB** RPGプログラムの作成時に、ジョブに対応したLANGID値を使用します。

### **言語ID**

使用する言語ID（たとえば、フランス語の場合にはFRA、ドイツ語の場合にはDEU）を入力してください。

上

---

## **SAAフラグ付け (SAAFLAG)**

SAA RPGによってサポートされない仕様のフラグ付けを行うかどうかを指定します。

### **\*NOFLAG**

フラグ付けは実行されません。

### **\*FLAG**

フラグ付けが実行されます。

上

---

## **印刷ファイル (PRTFILE)**

コンパイル・リストを入れるファイルの名前およびファイルが入っているライブラリーを指定します。

### **QSYSPRT**

コンパイル・リストがQSYSPRTファイルに入れられます。

### **ファイル名**

コンパイル・リストを入れるファイルの名前を入力してください。

**\*LIBL** システムはライブラリー・リストを検索して、ライブラリーを見つけます。

### **\*CURLIB**

現行ライブラリーが使用されます。現行ライブラリーを指定していない場合には、QGPLが使用されます。

### **ライブラリー名**

ファイルが入っているライブラリーの名前を入力してください。

---

## プログラムの置き換え (REPLACE)

同じライブラリーに同じ名前の既存のプログラム・オブジェクトがある時に、新しいプログラム・オブジェクトを作成するかどうかを指定します。

**\*YES** 新しいプログラム・オブジェクトが作成され、指定したライブラリーにある同じ名前の既存のプログラム・オブジェクトはライブラリーQRPLOBJに移動されます。

**\*NO** 同じ名前のプログラム・オブジェクトが指定したライブラリーにすでに存在している場合には、新しいプログラム・オブジェクトは作成されません。

---

## ターゲット・リリース (TGTRLS)

作成されるオブジェクトが使用されるオペレーティング・システムのリリースを指定します。**\*CURRENT** および**\*PRV**値の例の場合で、ターゲット・リリース を指定する時には、**VXRXXMX**の形式を使用してリリースを指定します。ここで、**VX**はバージョン、**RX**はリリース、**MX**はモディフィケーション・レベルです。たとえば、**V2R3M0**は、バージョン2、リリース3、モディフィケーション・レベル0です。

このパラメーターに対する有効な値はリリースごとに変わります。

考えられる値は次の通りです。

### **\*CURRENT**

オブジェクトは、現在ユーザーのシステムで実行中のオペレーティング・システムのリリースで使用されます。たとえば、**V2R3M5**がシステムで実行されている場合には、**\*CURRENT**は、ユーザーは**V2R3M5**が導入されているシステムでオブジェクトを使用することを意味します。また、オペレーティング・システムの後続のリリースが導入されたシステム上でこのオブジェクトを使用することもできます。

**\*PRV** オブジェクトは、前のリリースのモディフィケーション・レベル0のオペレーティング・システムで使用されます。たとえば、**V2R3M5**がシステムで実行されている場合には、**\*PRV**は**V2R2M0**が導入されているシステムでオブジェクトを使用することを意味します。また、オペレーティング・システムの後続のリリースが導入されたシステムで、このオブジェクトを使用することもできます。

### ターゲット・リリース

リリースを**VXRXXMX**の形式で指定してください。オブジェクトは、指定されたリリースのシステムまたはオペレーティング・システムの後続のリリースが導入されているシステムで使用することができます。

有効な値は、現行のバージョン、リリース、およびモディフィケーション・レベルによって異なり、新しいリリースごとに変わります。このコマンドでサポートされている最も古いリリースよりも前のリリース・レベルを指定した場合には、エラー・メッセージが出されて、サポートされる最も古いリリースを表示します。

注: プログラムは、作成コマンドに指定したリリースより前のリリース・レベルで復元することができません。プログラムを実行することができる最も古いリリースを判別するためには、**DSPPGM**を使用してください。

---

## ユーザー・プロファイル (USRPRF)

コンパイル済みRPGプログラムを実行するユーザー・プロファイルを指定します。

### \*USER

プログラムはプログラム・ユーザーのユーザー・プロファイルのもとで実行されます。

### \*OWNER

プログラムは、プログラムの所有者とユーザーの両方のユーザー・プロファイルで実行されます。  
プログラムがすでに存在している場合には、USRPRFパラメーターは更新されません。

---

## 権限 (AUT)

このプログラムのために認可する権限を指定します。

### \*LIBCRTAUT

作成されるオブジェクトの省略時の共通権限は、ターゲット・ライブラリーと関連のあるCRTAUTキーワードからとられます。この値は作成時に決定されます。ライブラリーのCRTAUTキーワードの値が作成後に変更された場合には、その新しい値は既存のオブジェクトに影響しません。

**\*ALL** オブジェクト所有権の転送の場合を除き、プログラムの権限を完了します。

### \*CHANGE

オブジェクトの存在および管理を取り扱う操作以外のすべての操作が認められます。

**\*USE** デバッグまたは変更権限を除く、コンパイル済みプログラムの読み取りまたは実行権限。

注: プログラムの変数の定様式ダンプを取得するには、プログラムに対して\*USE権限を持っていないければなりません。変数をダンプするには、さらにプログラムが検討可能な情報も持っていないければなりません。

一部のユーザーが変数のダンプができることを望まない場合は、ユーザーに\*OBJOPRだけに加えてプログラムに\*EXECUTE権限を与えてください。これはユーザーにプログラムの呼び出しは許可しますが、その変数をダンプすることは許可しません。

すべてのユーザーに変数のダンプができることを望まない場合は、プログラムの変更(CHGPGM)を使用して、プログラムの検討可能な情報を除去してください。

### \*EXCLUDE

権限なし。

### 権限リスト名

オブジェクトを保護する権限リストの名前。共通権限は\*AUTLになります。

---

## フェーズ・トレース (PHSTRC)

コンパイラーについてのフェーズのトレース情報をリストに含めるかどうかを指定します。

**\*NO** コンパイラー・フェーズの情報を作成しません。

**\*YES** コンパイラー・フェーズの情報を作成します。

上

---

## 中間テキスト・ダンプ (ITDUMP)

中間テキストの動的リストの作成を指定します。

### **\*NONE**

中間テキスト・ダンプを作成しません。

### フェーズ名

各フェーズ名の最後の 2 桁を入力してください。

上

---

## スナップ・ダンプ (SNPDUMP)

主要データ域および中間テキストのリストを作成するかどうかを指定します。

### **\*NONE**

スナップ・ダンプを作成しません。

### フェーズ名

各フェーズ名の最後の 2 桁を入力してください。

上

---

## コード・リスト (CODELIST)

特定のフェーズに対してIRPの動的リストを作成するかどうかを指定します。

### **\*NONE**

中間IRPダンプを作成しません。

**\*ALL** 中間IRPダンプを作成します。

### フェーズ名

各フェーズ名の最後の 2 桁を入力してください。

上

---

## 10進データ・エラーの無視 (IGNDECERR)

1 0 進数データ・エラーを無視するかどうかを指定します。

**\*NO** 1 0 進数データ・エラーは無視されません。

**\*YES** 1 0 進数データ・エラーは無視されます。

上

---

## ヌル値使用可能 (ALWNULL)

RPG/400プログラムが外部記述入力ファイルのヌル可能フィールドからヌル値を受け入れるかどうかを指定します。

**\*NO** RPG/400プログラムがヌル値フィールドを受け入れないことを指定します。

**\*YES** RPG/400プログラムが外部記述入力ファイルのヌル値フィールドを受け入れることを指定します。

上

---

## 例

### 例1: ソース・プログラムをプログラム・オブジェクトにコンパイル

```
CRTRPGPGM  PGM(MYLIB/XMPLE1)
            SRCFILE(MYLIB/QRPGSRC) SRCMBR(XMPLE1)
            OPTION(*SOURCE) TEXT('MY RPG III PROGRAM')
```

このコマンドはRPG/400コンパイラーを呼び出して、XMPLE1という名前のプログラムを作成します。ソース・プログラムはライブラリーMYLIB中のソース・ファイルQRPGSRCのメンバーXMPLE1にあります。コンパイラー・リストが作成されます。

上

---

## エラー・メッセージ

### \*ESCAPE メッセージ

#### QRG9001

コンパイルは正常に行われませんでした。プログラムは作成されません。

#### QRG9004

TGTRLSオプションに指定されたリリース&1はサポートされていない。

#### QRG9005

\*SRCDBGまたは\*LSTDBGコンパイラー・オプションを処理することはできない。

上





# AUTO REPORT RPG プログラム作成 (CRTRPTPGM)

実行可能場所: すべての環境 (\*ALL)

スレッド・セーフ: いいえ

パラメーター

例

エラー・メッセージ

報告書簡易作成機能のRPG/400コンパイラーを開始するためには、報告書簡易作成プログラム作成 (CRTRPTPGM) コマンドが使用されます。

上

## パラメーター

キーワード	記述	選択項目	ノート
PGM	プログラム	修飾オブジェクト名	オプション, 定位置 1
	修飾子 1: プログラム	名前, <u>*CTLSPEC</u>	
	修飾子 2: ライブラリー	名前, <u>*CURLIB</u>	
SRCFILE	ソース・ファイル	修飾オブジェクト名	オプション, 定位置 2
	修飾子 1: ソース・ファイル	名前, <u>QRPGSRC</u>	
	修飾子 2: ライブラリー	名前, <u>*LIBL</u> , *CURLIB	
SRCMBR	ソース・メンバー	名前, <u>*PGM</u>	オプション, 定位置 3
GENLVL	生成重大度レベル	0-99, <u>9</u>	オプション
TEXT	テキスト・記述	文字値, <u>*SRCMBRTXT</u> , *BLANK	オプション
OPTION	ソース・リスト・オプション	値 (最大 14 回の繰り返し): *SOURCE, *NOSOURCE, *SRC, *NOSRC, *XREF, *NOXREF, *GEN, *NOGEN, *DUMP, *NODUMP, *SECLVL, *NOSECLVL, *LSTDBG, *NOLSTDBG	オプション
GENOPT	生成オプション	値 (最大 10 回の繰り返し): *LIST, *NOLIST, *XREF, *NOXREF, *ATR, *NOATR, *DUMP, *NODUMP, *PATCH, *NOPATCH, *OPTIMIZE, *NOOPTIMIZE	オプション
INDENT	ソース・リストの字下げ	文字値, <u>*NONE</u>	オプション
CVTOPT	タイプ変換オプション	単一値: <u>*NONE</u> その他の値 (最大 3 回の繰り返し): *DATETIME, *VARCHAR, *GRAPHIC	オプション

キーワード	記述	選択項目	ノーツ
SRTSEQ	ソート順序	単一値: <u>*HEX</u> , *JOB, *JOB RUN, *LANGIDUNQ, *LANGIDSHR その他の値: 修飾オブジェクト名	オプション
	修飾子 1: ソート順序	名前	
	修飾子 2: ライブラリー	名前, <u>*LIBL</u> , *CURLIB	
LANGID	言語ID	名前, <u>*JOB RUN</u> , *JOB	オプション
PRTFILE	印刷ファイル	修飾オブジェクト名	オプション
	修飾子 1: 印刷ファイル	名前, <u>QSYSPRT</u>	
	修飾子 2: ライブラリー	名前, <u>*LIBL</u> , *CURLIB	
RPTOPT	報告書簡易作成機能オプション	値 (最大 10 回の繰り返し): *SOURCE, *NOSOURCE, *SRC, *NOSRC, *FLOW, *NOFLOW, *AST, *NOAST, *DATE, *NODATE, *COMPILE, *NOCOMPILE, *SECLVL, *NOSECLVL	オプション
OUTFILE	報告書簡易作成機能出力ファイル	修飾オブジェクト名	オプション
	修飾子 1: 報告書簡易作成機能出力ファイル	名前, <u>*NONE</u>	
	修飾子 2: ライブラリー	名前, <u>*LIBL</u> , *CURLIB	
OUTMBR	報告書簡易作成機能出力メンバー	名前, <u>*NONE</u>	オプション
REPLACE	プログラムの置き換え	<u>*YES</u> , *NO	オプション
TGTRLS	ターゲット・リリース	単純名, <u>*CURRENT</u> , *PRV	オプション
USRPRF	ユーザー・プロファイル	<u>*USER</u> , *OWNER	オプション
AUT	権限	名前, <u>*LIBCRTAUT</u> , *ALL, *CHANGE, *USE, *EXCLUDE	オプション
PHSTRC	フェーズ・トレース	<u>*NO</u> , *YES	オプション
ITDUMP	中間テキスト・ダンプ	文字値, <u>*NONE</u>	オプション
SNPDUMP	スナップ・ダンプ	文字値, <u>*NONE</u>	オプション
CODELIST	コード・リスト	文字値, <u>*NONE</u> , *ALL	オプション
IGNDECERR	10進データ・エラーの無視	<u>*NO</u> , *YES	オプション
ALWNULL	ヌル値使用可能	<u>*NO</u> , *YES	オプション

上

## プログラム (PGM)

コンパイル済みRPGプログラムのプログラム名およびライブラリーを指定します。

### \*CTLSPEC

制御仕様の75-80桁目で指示されたプログラム名。

### プログラム名

プログラムを識別する名前を入力してください。

#### **\*CURLIB**

現行ライブラリーが使用されます。 現行ライブラリーを指定していない場合には、QGPLが使用されます。

#### **ライブラリー名**

コンパイル済みプログラムを保管するライブラリーの名前を入力してください。

上

---

## **ソース・ファイル (SRCFILE)**

ソース・プログラムが入っているソース・ファイルの名前を指定します。

#### **QRPGSRC**

省略時のソース・ファイルQRPGSRCには、コンパイルされるRPGプログラムが入っています。

#### **ソース・ファイル名**

コンパイルされるRPGソース・プログラムが入っているソース・ファイル名を入力します。

**\*LIBL** システムはライブラリー・リストを検索して、ソース・ファイルが入っているライブラリーを見つけます。

#### **\*CURLIB**

現行ライブラリーが使用されます。 現行ライブラリーを指定していない場合には、QGPLが使用されます。

#### **ライブラリー名**

ソース・ファイルが入っているライブラリーの名前を入力してください。

上

---

## **ソース・メンバー (SRCMBR)**

ソース・ファイルのメンバーの名前を指定します。

**\*PGM** ソース・ファイル・メンバー名としてPGMパラメーターで指定した名前。

#### **ソース・ファイル・メンバー名**

ソース・プログラムが入っているメンバーの名前を入力してください。

上

---

## **生成重大度レベル (GENLVL)**

プログラム・オブジェクトの作成を停止する診断メッセージの重大度レベルを指定します。

**9** 省略時の重大度レベルは9です。

#### **重大度レベル値**

01-50の2桁の数字を入力してください。

上

---

## テキスト'記述' (TEXT)

プログラム機能の説明に使用するテキストを指定します。

### \*SRCMBRTXT

ソース・ファイル・メンバーのテキスト記述を使用します。

### **\*BLANK**

テキストはありません。

上

---

## ソース・リスト・オプション (OPTION)

ソース・プログラムをコンパイルする時に使用するオプションを指定します。

### \*SOURCEまたは**\*SRC**

コンパイラーがソース・リストを提供します。

### **\*NOSOURCE**または**\*NOSRC**

コンパイラーはソース・リストを提供しません。

### \*XREF

コンパイラーが相互参照表を提供します。

### **\*NOXREF**

コンパイラーは相互参照表を提供しません。

**\*GEN** プログラムのコンパイル後に実行可能なプログラム・オブジェクトが作成されます。

### **\*NOGEN**

プログラム・オブジェクトは作成されません。

### \*NODUMP

エラーが起こった時にプログラム・テンプレートを印刷しません。

### **\*DUMP**

エラーが起こった時にプログラム・テンプレートを印刷します。

### \*NOSECLVL

メッセージ・テキストは印刷されません。

### **\*SECLVL**

メッセージ・テキストが印刷されます。

### \*NOLSTDBG

リスト・ビューまたはリスト・レベルのデバッグ情報、あるいはソース・レベルのエラー情報は作成されません。

### **\*LSTDBG**

コンパイラーは、クライアント・ツールで使用するためのリスト・ビュー、ソース・レベルのエラー情報、およびリスト・レベルのデバッグ情報を作成します。また、システム・デバッガー (STRDBG OPMSRC(\*YES))を使用してOPMプログラムとILEプログラムを同時にデバッグしたい場合には、リスト・レベルのデバッグ情報が必要です。

上

---

## 生成オプション (GENOPT)

オブジェクト・コードの作成に使用するオプションを指定してください。

### \*NOLIST

プログラムの中間表現(IRP)リストを作成しません。

**\*LIST** プログラムの中間表現(IRP)をリストします。

### \*NOXREF

プログラムの中間表現(IRP)の相互参照リストを作成しません。

### **\*XREF**

プログラムの中間表現(IRP)に定義されたすべてのオブジェクトの相互参照リストを作成します。

### \*NOATR

属性リストを作成しません。

**\*ATR** プログラムの中間表現(IRP)ソース・プログラムの属性をリストします。

### \*NODUMP

エラーが起こった時にプログラム・テンプレートを印刷しません。

### **\*DUMP**

プログラム・テンプレートを印刷します。

### \*NOPATCH

コンパイル済みプログラムにプログラム・パッチ域を予約しません。

### **\*PATCH**

コンパイル済みプログラムにプログラム・パッチ域のスペースを予約します。

### \*NOOPTIMIZE

コンパイラはプログラムの最適化を実行しません。

### **\*OPTIMIZE**

コンパイラは処理効率の高いプログラムを作成します。

上

---

## ソース・リストの字下げ (INDENT)

DOステートメントおよびIF-ELSE文節は、読みやすいように字下げを指定します。また、対応するDO-ENDDOの対とIF-ELSEの対の結合に使用される文字も指定します。

### \*NONE

DOステートメントまたはIF-ELSE文節の中では、リストは字下げされません。

### 文字ストリング

対応するネスト・レベルを結合するためには所定の文字ストリングを使用します。

上

---

## タイプ変換オプション (CVTOPT)

日付、時刻、およびタイム・スタンプ・データベースのデータ・タイプ、および外部記述ファイルから検索される可変長データ・タイプをRPG/400コンパイラで処理する方法を指定します。

#### **\*NONE**

日付、時刻、タイム・スタンプ、および可変長データベースのデータ・タイプは無視されて、RPG/400プログラムでアクセスすることはできません。

#### **\*DATETIME**

日付、時刻、タイム・スタンプのデータベースのデータ・タイプは固定長文字フィールドとして宣言されて、RPG/400プログラムでアクセスすることができます。

#### **\*VARCHAR**

可変長データベースのデータ・タイプは固定長文字フィールドとして宣言されて、RPG/400プログラムでアクセスすることができます。

#### **\*GRAPHIC**

DBCSグラフィック・データ・タイプは固定長の文字フィールドとして宣言され、RPG/400プログラムでアクセス可能です。

注: プログラム中で可変長DBCSグラフィック・データ・タイプを宣言する必要がある場合には、

\*VARCHARおよび\*GRAPHICの両方のパラメーターを選択してください。

上

---

## **ソート順序 (SRTSEQ)**

使用するソート順序テーブルを指定します。

注: CRTRPGPGMまたはCRTRPTPGMコマンドのSRTSEQおよびLANGIDパラメーターにコーディングされた値を使用するためには、制御仕様の代替照合順序フィールドにDを指定しなければなりません。代替照合順序はコンパイル時または実行時のいずれかにシステムから検索されます。Dオプションを指定すると、代替照合順序は以下に影響を与えます。すなわち、すべての文字比較操作、文字テーブルおよび配列のLOKUPおよびSORTA、および文字のコンパイル時データおよび実行時前配列とテーブルの順序検査に影響を与えます。実行時に代替照合順序が検索される場合には、コンパイル時データの順序検査は実行時まで延期されます。

**\*HEX** ソート順序を決定するために、文字の16進数値を使用します。これが省略時の値です。

**\*JOB** RPGプログラムの作成時に、ジョブに対応したSRTSEQ値を使用します。

#### **\*JOBRUN**

RPGプログラムの実行時に、ジョブに対応したSRTSEQ値を使用します。

#### **\*LANGIDUNQ**

固有の重みづけテーブルを使用します。この特殊値は、ソート順序テーブルの選択で、LANGIDパラメーターと一緒に使用されます。

#### **\*LANGIDSHR**

共用の重みづけテーブルを使用します。この特殊値は、ソート順序テーブルの選択で、LANGIDパラメーターと一緒に使用されます。

#### **ソート順序テーブル名**

ソート順序テーブルの名前を入力してください。

**\*LIBL** コンパイラーは、ライブラリー・リストを検索してソート順序テーブルが入っているライブラリーを見つけます。これが省略時の値です。

#### **\*CURLIB**

ソート順序テーブルを見つけるために、現行ライブラリーが検索されます。現行ライブラリーが指定されていない場合には、QGPLが使用されます。

#### **ライブラリー名**

ソート順序テーブルが入っているライブラリーの名前を入力してください。

上

---

## **言語ID (LANGID)**

ソート順序が\*LANGIDUNQまたは\*LANGIDSHRの時に使用する言語IDを指定します。LANGIDパラメーターは、ソート順序テーブルの選択で、SRTSEQパラメーターと一緒に使用されます。

#### **\*JOB RUN**

RPGプログラムの実行時に、ジョブに対応したLANGID値を使用します。これが省略時の値です。

**\*JOB** RPGプログラムの作成時に、ジョブに対応したLANGID値を使用します。

#### **言語ID**

使用する言語ID（たとえば、フランス語の場合にはFRA、ドイツ語の場合にはDEU）を入力してください。

上

---

## **印刷ファイル (PRTFILE)**

コンパイル・リストを入れるファイルの名前およびファイルが入っているライブラリーを指定します。

#### **QSYSPRT**

コンパイル・リストがQSYSPRTファイルに入れられます。

#### **ファイル名**

コンパイル・リストを入れるファイルの名前を入力してください。

**\*LIBL** システムはライブラリー・リストを検索して、ライブラリーを見つけます。

#### **\*CURLIB**

現行ライブラリーが使用されます。現行ライブラリーを指定していない場合には、QGPLが使用されます。

#### **ライブラリー名**

ファイルが入っているライブラリーの名前を入力してください。

上

---

## **報告書簡易作成機能オプション (RPTOPT)**

報告書簡易作成機能ソース・プログラムのコンパイル時に使用するオプションを指定します。

#### **\*NOSOURCEまたは\*NOSRC**

コンパイラーはソース・リストを提供しません。

#### **\*SOURCEまたは\*SRC**

コンパイラーがソース・リストを提供します。



#### **\*NOFLOW**

メジャー・ルーチンの実行の流れは書き出されません。

#### **\*FLOW**

メジャー・ルーチンの実行の流れが書き出されます。

#### **\*NOAST**

合計書き出し行のアスタリスクは生成されません。

**\*AST** 合計書き出し行のアスタリスクは生成されます。

#### **\*DATE**

最初の\*AUTOページ見出し行に日付およびページ番号が印刷されます。

#### **\*NODATE**

最初の\*AUTOページ見出し行に日付およびページ番号は印刷されません。

#### **\*COMPILE**

報告書簡易作成機能のソース・コンパイルの後に、RPG/400コンパイラーが呼び出されます。

#### **\*NOCOMPILE**

RPG/400コンパイラーは呼び出されません。

#### **\*NOSECLVL**

メッセージ・テキストは印刷されません。

#### **\*SECLVL**

メッセージ・テキストが印刷されます。

上

---

## **報告書簡易作成機能出力ファイル (OUTFILE)**

報告書簡易作成機能で作成されるRPGソース・プログラムの出力ファイルの名前を指定します。

#### **\*NONE**

報告書簡易作成機能は、RPGソース・プログラムをコンパイラーに渡すための一時ファイルを作成します。

#### **ファイル名**

生成されたRPGソース・プログラムが入っているファイルの名前を入力します。

**\*LIBL** システムはライブラリー・リストを検索して、ソース・ファイルが入っているライブラリーを見つけます。

#### **\*CURLIB**

現行ライブラリーが使用されます。 現行ライブラリーを指定していない場合には、QGPLが使用されます。

#### **ライブラリー名**

ソース・ファイルが入っているライブラリーの名前を入力してください。

上

---

## **報告書簡易作成機能出力メンバー (OUTMBR)**

出力ファイルのメンバーの名前を指定します。

#### **\*NONE**

メンバー名としてソース・メンバー名を使用します。

#### **ファイル・メンバー名**

報告書簡易作成機能プログラムからの出力を受け取るメンバーの名前を入力します。

上

---

## **プログラムの置き換え (REPLACE)**

同じライブラリーに同じ名前の既存のプログラム・オブジェクトがある時に、新しいプログラム・オブジェクトを作成するかどうかを指定します。

**\*YES** 新しいプログラム・オブジェクトが作成され、指定したライブラリーにある同じ名前の既存のプログラム・オブジェクトはライブラリーQRPLOBJに移動されます。

**\*NO** 同じ名前のプログラム・オブジェクトが指定したライブラリーにすでに存在している場合には、新しいプログラム・オブジェクトは作成されません。

上

---

## **ターゲット・リリース (TGTRLS)**

作成されるオブジェクトが使用されるオペレーティング・システムのリリースを指定します。**\*CURRENT** および**\*PRV**値の例の場合で、ターゲット・リリース を指定する時には、**VXRXMX**の形式を使用してリリースを指定します。ここで、**VX**はバージョン、**RX**はリリース、**MX**はモディフィケーション・レベルです。たとえば、**V2R3M0**は、バージョン2、リリース3、モディフィケーション・レベル0です。

このパラメーターに対する有効な値はリリースごとに変わります。

考えられる値は次の通りです。

#### **\*CURRENT**

オブジェクトは、現在ユーザーのシステムで実行中のオペレーティング・システムのリリースで使用されます。たとえば、**V2R3M5**がシステムで実行されている場合には、**\*CURRENT**は、ユーザーは**V2R3M5**が導入されているシステムでオブジェクトを使用することを意味します。また、オペレーティング・システムの後続のリリースが導入されたシステム上でこのオブジェクトを使用することもできます。

**\*PRV** オブジェクトは、前のリリースのモディフィケーション・レベル0のオペレーティング・システムで使用されます。たとえば、**V2R3M5**がシステムで実行されている場合には、**\*PRV**は**V2R2M0**が導入されているシステムでオブジェクトを使用することを意味します。また、オペレーティング・システムの後続のリリースが導入されたシステムで、このオブジェクトを使用することもできます。

#### **ターゲット・リリース**

リリースを**VXRXMX**の形式で指定してください。オブジェクトは、指定されたリリースのシステムまたはオペレーティング・システムの後続のリリースが導入されているシステムで使用することができます。

有効な値は、現行のバージョン、リリース、およびモディフィケーション・レベルによって異なり、新しいリリースごとに変わります。このコマンドでサポートされている最も古いリリースよりも前のリリース・レベルを指定した場合には、エラー・メッセージが出されて、サポートされる最も古いリリースを表示します。

注: プログラムは、作成コマンドに指定したリリースより前のリリース・レベルで復元することができません。プログラムを実行することができる最も古いリリースを判別するためには、DSPPGMを使用してください。

上

---

## ユーザー・プロファイル (USRPRF)

コンパイル済みRPGプログラムを実行するユーザー・プロファイルを指定します。

### **\*USER**

プログラムはプログラム・ユーザーのユーザー・プロファイルのもとで実行されます。

### **\*OWNER**

プログラムは、プログラムの所有者とユーザーの両方のユーザー・プロファイルで実行されます。プログラムがすでに存在している場合には、USRPRFパラメーターは更新されません。

上

---

## 権限 (AUT)

このプログラムのために認可する権限を指定します。

### **\*LIBCRTAUT**

作成されるオブジェクトの省略時の共通権限は、ターゲット・ライブラリーと関連のあるCRTAUTキーワードからとられます。この値は作成時に決定されます。ライブラリーのCRTAUTキーワードの値が作成後に変更された場合には、その新しい値は既存のオブジェクトに影響しません。

**\*ALL** オブジェクト所有権の転送の場合を除き、プログラムの権限を完了します。

### **\*CHANGE**

オブジェクトの存在および管理を取り扱う操作以外のすべての操作が認められます。

**\*USE** デバッグまたは変更権限を除く、コンパイル済みプログラムの読み取りまたは実行権限。

注: プログラムの変数の定様式ダンプを取得するには、プログラムに対して\*USE権限を持っていないけません。変数をダンプするには、さらにプログラムが検討可能な情報も持っていないけません。

一部のユーザーが変数のダンプができることを望まない場合は、ユーザーに\*OBJOPRだけに加えてプログラムに\*EXECUTE権限を与えてください。これはユーザーにプログラムの呼び出しは許可しますが、その変数をダンプすることは許可しません。

すべてのユーザーに変数のダンプができることを望まない場合は、プログラムの変更(CHGPGM)を使用して、プログラムの検討可能な情報を除去してください。

### **\*EXCLUDE**

権限なし。

## 権限リスト名

オブジェクトを保護する権限リストの名前。共通権限は\*AUTLになります。

上

---

## フェーズ・トレース (PHSTRC)

コンパイラについてのフェーズのトレース情報をリストに含めるかどうかを指定します。

**\*NO** コンパイラ・フェーズの情報を作成しません。

**\*YES** コンパイラ・フェーズの情報を作成します。

上

---

## 中間テキスト・ダンプ (ITDUMP)

中間テキストの動的リストの作成を指定します。

### **\*NONE**

中間テキスト・ダンプを作成しません。

### フェーズ名

各フェーズ名の最後の2桁を入力してください。

上

---

## スナップ・ダンプ (SNPDUMP)

主要データ域および中間テキストのリストを作成するかどうかを指定します。

### **\*NONE**

スナップ・ダンプを作成しません。

### フェーズ名

各フェーズ名の最後の2桁を入力してください。

上

---

## コード・リスト (CODELIST)

特定のフェーズに対してIRPの動的リストを作成するかどうかを指定します。

### **\*NONE**

中間IRPダンプを作成しません。

**\*ALL** 中間IRPダンプを作成します。

### フェーズ名

各フェーズ名の最後の2桁を入力してください。

上

---

## 10進データ・エラーの無視 (IGNDECERR)

1 0 進数データ・エラーを無視するかどうかを指定します。

**\*NO** 1 0 進数データ・エラーは無視されません。

**\*YES** 1 0 進数データ・エラーは無視されます。

上

---

## ヌル値使用可能 (ALWNULL)

RPG/400プログラムが外部記述入力ファイルのヌル可能フィールドからヌル値を受け入れるかどうかを指定します。

**\*NO** RPG/400プログラムがヌル値フィールドを受け入れないことを指定します。

**\*YES** RPG/400プログラムが外部記述入力ファイルのヌル値フィールドを受け入れることを指定します。

上

---

## 例

### 例1: ソース・プログラムをプログラム・オブジェクトにコンパイル

```
CRTRPTPGM  PGM(MYLIB/XMPLE1)
            SRCFILE(MYLIB/QRPGSRC) SRCMBR(XMPLE1)
            OPTION(*SOURCE)
            TEXT('MY RPG III AUTO REPORT PROGRAM')
```

このコマンドはRPG/400コンパイラを呼び出して、XMPLE1という名前の自動レポート・プログラムを作成します。ソース・プログラムはライブラリーMYLIB中のソース・ファイルQRPGSRCのメンバーXMPLE1にあります。コンパイラ・リストが作成されます。

上

---

## エラー・メッセージ

### **\*ESCAPE** メッセージ

#### **RPT9001**

報告書簡易作成機能が正常に実行されなかった。

#### **RPT0082**

重大なエラーが起こったため、報告書簡易作成機能の生成は打ち切られた。

#### **QRG9004**

TGTRLSオプションに指定されたリリース&1はサポートされていない。

上

## S/36 COBOLプログラム作成 (CRTS36CBL)

実行可能場所: すべての環境 (\*ALL)

スレッド・セーフ: いいえ

パラメーター

例

エラー・メッセージ

CRTS36CBLコマンドは、COBOLソース・プログラムをシステム/36環境で使用するプログラム・オブジェクトにコンパイルします。このコマンドは、対話式、バッチ・モード、あるいはCLプログラムから使用することができます。

CRTS36CBLコマンドで指定するオブジェクト名は、すべて英数字で構成されなければならず、その最初の文字は英字でなければなりません。名前の長さは、8桁以下でなければなりません。

上

### パラメーター

キーワード	記述	選択項目	ノーツ
PGM	プログラム	修飾オブジェクト名	オプション, 定位置 1
	修飾子 1: プログラム	名前, *PGMID	
	修飾子 2: ライブラリー	名前, *CURLIB	
SRCFILE	ソース・ファイル	修飾オブジェクト名	オプション, 定位置 2
	修飾子 1: ソース・ファイル	名前, QS36SRC	
	修飾子 2: ライブラリー	名前, *LIBL, *CURLIB	
SRCMBR	ソース・メンバー	名前, *PGM	オプション, 定位置 3
GENLVL	生成重大度レベル	0-29, 29	オプション, 定位置 4
NEP	非終了プログラム	*NO, *YES	オプション, 定位置 5
MRTMAX	MRT装置の最大数	0-99, 0	オプション, 定位置 6
TEXT	テキスト'記述'	文字値, *SRCMBRTXT, *BLANK	オプション
OPTION	ソース・リスト・オプション	値 (最大 50 回の繰り返し): *SOURCE, *NOSOURCE, *SRC, *NOSRC, *NOXREF, *XREF, *GEN, *NOGEN, *NOSEQUENCE, *SEQUENCE, *NOVBSUM, *VBSUM, *NONUMBER, *NUMBER, *LINENUMBER, *NOMAP, *MAP, *NOOPTIONS, *OPTIONS, *QUOTE, *APOST, *DEBUG, *NODEBUG, *SECLVL, *NOSECLVL, *PRINT, *NOPRINT	オプション

キーワード	記述	選択項目	ノーツ
<b>GENOPT</b>	生成オプション	値 (最大 50 回の繰り返し): *NOLIST, *LIST, *NOXREF, *XREF, *NOPATCH, *PATCH, *NODUMP, *DUMP, *NOATR, *ATR, *RANGE, *NORANGE, *UNREF, *NOUNREF, *NOOPTIMIZE, *OPTIMIZE	オプション
<b>PRTFILE</b>	印刷ファイル	修飾オブジェクト名	オプション
	修飾子 1: 印刷ファイル	名前, <u>QSYSPRT</u>	
	修飾子 2: 印刷ファイル・ライブラリー	名前, *LIBL, *CURLIB	
<b>ICFLIB</b>	ICFファイルのライブラリー	名前	オプション
<b>REPLACE</b>	プログラムの置き換え	*NO, *YES	オプション
<b>TGTRLS</b>	ターゲット・リリース	単純名, *CURRENT, *PRV	オプション
<b>USRPRF</b>	ユーザー・プロファイル	*USER, *OWNER	オプション
<b>AUT</b>	権限	名前, *LIBCRTAUT, *ALL, *CHANGE, *USE, *EXCLUDE	オプション
<b>DUMP</b>	コンパイラ・デバッグ・ダンプ	要素リスト	オプション
	要素 1:	1-32767, <u>1</u> , *	
	要素 2:	1-32767, <u>32767</u>	
<b>ITDUMP</b>	中間テキスト・ダンプ	0-31, <u>0</u>	オプション
<b>FIXDECDTA</b>	1 0 進数データの修正	*NO, *YES	オプション
<b>CPYLIB</b>	コピー・ファイル・ライブラリー	名前, *LIBL, *CURLIB	オプション

上

## プログラム (PGM)

作成中のCOBOLプログラム・オブジェクトのプログラム名およびライブラリー名を指定します。考えられる値は次の通りです。

### \*PGMID

プログラム・オブジェクトの名前は、COBOLソース・プログラムのPROGRAM-ID 段落から取られます。

### プログラム名

コンパイル済みCOBOLプログラムを識別する名前を入力してください。このパラメーターにプログラム名を指定してバッチ・モードでコンパイルを実行すると、バッチ・ジョブの最初のプログラムがこの名前を使用し、他のプログラムはソース・プログラムのPROGRAM-ID段落に指定された名前を使用します。

使用できるライブラリーの値は次の通りです。

### \*CURLIB

現行ライブラリーが使用されます。現行ライブラリーとしてライブラリーが割り当てられていない場合には、QGPLが使用されます。



### ライブラリー名

作成したプログラム・オブジェクトを入れるライブラリーの名前を入力してください。

上

---

## ソース・ファイル (SRCFILE)

コンパイルしたいCOBOLソースが入っているソース・ファイルの名前を指定します。考えられる値は次の通りです。

### QS36SRC

弊社提供ソース・ファイルQS36SRCに、コンパイルしたいCOBOLソースが入っています。

### ソース・ファイル名

コンパイルしたいCOBOLソース・プログラムが入っているソース・ファイルの名前を入力してください。このソース・ファイルのレコード長は92でなければなりません。

使用できるライブラリーの値は次の通りです。

**\*LIBL** ファイルが入っているライブラリーを見つけるために、システムはライブラリー・リストを検索します。

### **\*CURLIB**

現行ライブラリーが使用されます。現行ライブラリーとしてライブラリーが割り当てられていない場合には、QGPLが使用されます。

### ライブラリー名

ソース・ファイルが入っているライブラリーの名前を入力してください。

上

---

## ソース・メンバー (SRCMBR)

コンパイルしたいCOBOLソースが入っているソース・ファイルのメンバーの名前を指定します。このパラメーターを指定できるのは、SRCFILEパラメーターに参照されているソース・ファイルがデータベース・ファイルの場合だけです。考えられる値は次の通りです。

**\*PGM** PGMパラメーターにプログラム名を指定した場合には、コンパイラーはこのプログラムと同じ名前のメンバー中のソース・プログラムを検索し、このプログラムおよびメンバーと同じ名前でプログラム・オブジェクトを作成します。

PGMパラメーターにプログラム名を指定しない場合には、コンパイラーはデータベース・ソース・ファイルの最初のメンバー中のソース・プログラムを検索し、PROGRAM-ID 段落に指定された名前を使用してプログラム・オブジェクトを作成します。

### ソース・ファイル・メンバー名

COBOLソース・プログラムが入っているメンバー名を入力してください。

上

---

## 生成重大度レベル (GENLVL)

目的プログラムを作成するかどうかを判別する重大度レベルを指定します。この重大度レベルは、プログラムのコンパイル時に作成されるメッセージの重大度レベルに対応します。エラー・メッセージの重大度レベルが指定した値よりも大きい場合には、プログラム・オブジェクトは作成されません。たとえば、このパラメーターに19を指定して、メッセージのいずれかの重大度レベルが20またはそれ以上であれば、プログラム・オブジェクトは作成されません。

**29** プログラム中で29より大きい重大度レベルのエラーが起こった場合には、プログラム・オブジェクトは作成されません。

### 重大度レベル

2桁の数字(00-29)を入力してください。

上

---

## 非終止プログラム (NEP)

このプログラムが非終止（または長時間実行）プログラムであるかどうかを指定します。非終止プログラムは休むことなく実行され、他のプログラムとは共用されないシステム資源（ディスク記憶域、表示装置、または印刷装置など）を使用します。非終止プログラムは、その要求元がすべて解放されても終了しません。

考えられる値は次の通りです。

**\*NO** プログラムは非終止プログラムではありません。

**\*YES** プログラムは非終止プログラムです。

上

---

## MRT装置の最大数 (MRTMAX)

コンパイル済みプログラムの実行中に使用できる要求元表示装置の最大数を指定します。最大は99です。考えられる値は次の通りです。

**0** 要求表示装置はありません。

**1-99** 2桁の数字(00-99)を指定することができます。

上

---

## テキスト'記述' (TEXT)

プログラムおよびその機能の簡単な記述。考えられる値は次の通りです。

### **\*SRCMBRTXT**

COBOLソースが入っているデータベース・ファイル・メンバーを記述するものと同じテキストをプログラム・オブジェクトに使用します。ソースが装置またはインライン・ファイルから入力された場合には、\*SRCMBRTXTの指定は、\*BLANKの指定と同じ効果となります。

### **\*BLANK**

テキストは指定されません。

## テキスト記述

プログラムおよびその機能を簡単に記述するテキストを入力してください。テキストの長さは最大50桁とすることができますが、アポストロフィで囲まなければなりません。アポストロフィは、50桁のストリングの一部ではありません。

上

---

## ソース・リスト・オプション (OPTION)

COBOLソースのコンパイル時に使用するオプションを指定します。考えられる値は次の通りです。

### **\*SOURCE**または**\*SRC**

コンパイラーは、COBOLソース入力とすべてのコンパイル時エラー・メッセージから構成されるソース・リストを作成します。

### **\*NOSOURCE**または**\*NOSRC**

コンパイラーは、ソース・リストを作成しません。

### **\*NOXREF**

コンパイラーは、ソース・プログラムの相互参照表を作成しません。

### **\*XREF**

コンパイラーは、ソース・プログラムの相互参照表を作成します。

**\*GEN** コンパイラーは、ソース・プログラムのコンパイル後にプログラム・オブジェクトを作成します。

### **\*NOGEN**

コンパイラーは、構文検査を実行して適切なエラー・メッセージをリストしますが、コンパイル済みプログラム・オブジェクトを作成しません。

### **\*SEQUENCE**

コンパイラーは、ソース・プログラム・ステートメントの順序を検査します。ステートメントが順序どおりではない場合には、メッセージが印刷されます。

### **\*NOSEQUENCE**

コンパイラーは、ソース・プログラム・ステートメントの順序を検査しません。

### **\*NOVBSUM**

動詞使用カウントは印刷されません。

### **\*VBSUM**

動詞使用カウントが印刷されます。

### **\*NUMBER**

参照番号として、ユーザー提供の順序番号（1-6桁目）が使用されます。

### **\*NONUMBER**

参照番号として、ソース・ファイルの順序番号（1-6桁目）が使用されます。

### **\*LINENUMBER**

コンパイラーによって作成された順序番号が参照番号として使用されます。このオプションは、プログラム・ソース・コードとCOPYステートメントによって提示されたソース・コードを1つの連続番号順に結合します。FIPS（連邦情報処理標準）フラグ付けを指定する場合には、このオプションを使用してください。

### **\*NOMAP**

コンパイラーは、データ部マップをリストしません。

**\*MAP** コンパイラーはデータ部マップをリストします。

**\*OPTIONS**

このコンパイルに有効なオプションがリストされます。

**\*NOOPTIONS**

このコンパイルに有効なオプションはリストされません。

**\*QUOTE**

リテラルを表すために引用符(")を使用することを指定します。

**\*APOST**

リテラルを表すためにアポストロフィ(')を使用することを指定します。

**\*DEBUG**

WITH DEBUGGING MODE文節を含んでいないプログラムにDEBUGを指定しても、このオプションは効果がありません。WITH DEBUGGING ON文節を通常の指定として取り扱います。

**\*NODEBUG**

WITH DEBUGGING ON文節を注記として取り扱います。

**\*NOSECLVL**

このコンパイルでは、第2レベル・メッセージ・テキストはリストされません。

**\*SECLVL**

このコンパイルでは、第2レベル・メッセージ・テキストがリストされます。

**\*PRINT**

COBOLCプロシージャーによって作成されたコンパイル・リストが印刷されます。

**\*NOPRINT**

COBOLCプロシージャーによって作成されたコンパイル・リストは印刷または表示されません。

上

---

## 生成オプション (GENOPT)

プログラム・オブジェクトの作成時に使用するオプションを指定します。このリストは、COBOLで問題が起こった場合に必要になることがあります。考えられる値は次の通りです。

**\*NOLIST**

IRP（プログラムの中間表示）、関連した16進コード、またはエラー・メッセージはリストされません。

**\*LIST** IRP,関連した16進コード、およびエラー・メッセージがリストされます。

**\*NOXREF**

IRPで定義されているすべてのオブジェクトの相互参照表は作成されません。

**\*XREF**

IRPで定義されているすべてのオブジェクトの相互参照表が作成されます。

**\*NOPATCH**

コンパイル済みプログラムに、プログラム・パッチ域用のスペースは予約されません。

**\*PATCH**

コンパイル済みプログラムに、プログラム・パッチ域用のスペースが予約されます。プログラム・パッチ域は、デバッグ用に使用することができます。

#### **\*NODUMP**

プログラム・テンプレートはリストされません。

#### **\*DUMP**

プログラム・テンプレートがリストされます。

#### **\*NOATR**

IRPソースの属性はリストされません。

**\*ATR** IRPソースの属性がリストされます。

#### **\*NORANGE**

実行時に範囲は検査されません。

#### **\*RANGE**

実行時にシステムは添え字が正しい範囲内にあるかどうかを検査しますが、指標の範囲は検査しません。また、コンパイラーによって作成されたコードのサブストリング操作も検査します。

#### **\*UNREF**

コンパイル済みプログラムに、参照されていないデータ項目が組み込まれます。

#### **\*NOUNREF**

コンパイル済みプログラムには、参照されていないデータ項目は組み込まれません。これにより、使用されるODT（オブジェクト定義テーブル）項目の数が減らされ、より大きいプログラムをコンパイルすることができます。ただし、参照されていないデータ項目は、\*XREFオプションによって作成された相互参照表には表示されます。

#### **\*NOOPTIMIZE**

コンパイラーはプログラムの標準最適化だけを実行します。

#### **\*OPTIMIZE**

作成されたプログラム・オブジェクトはより効率的に実行でき、記憶域も小さくて済みますが、\*OPTIMIZEを指定すると、プログラムのコンパイルに必要な時間がかなり長くなる場合があります。

上

---

## 印刷ファイル (PRTFILE)

コンパイル・リストが送られるファイルの名前およびそのファイルが入れられるライブラリーを指定します。このファイルのレコード長は最低132でなければなりません。132より小さいレコード長のファイルを指定すると、情報が失われます。

考えられる値は次の通りです。

#### **QSYSPRT**

ファイル名を指定しない場合には、コンパイル・リストは弊社提供のQSYSPRTファイルに送られます。

#### **ファイル名**

コンパイラー・リストが送られるファイル名を入力してください。

使用できるライブラリーの値は次の通りです。

**\*LIBL** ファイルが入っているライブラリーを見つけるために、システムはライブラリー・リスト\*LIBLを検索します。

### \*CURLIB

現行ライブラリーとして指定したライブラリーが検索されます。現行ライブラリーとしてライブラリーを割り当てていない場合には、QGPLが使用されます。

#### ライブラリー名

ファイルが入れられるライブラリーの名前を入力してください。

上

---

## ICFファイルのライブラリー (ICFLIB)

プログラムで使用するICFレコード様式定義が入っているライブラリーを指定します。

#### ICFライブラリー名

ICFファイルのライブラリーの名前を入力してください。

上

---

## プログラムの置き換え (REPLACE)

同じライブラリーに同じ名前のプログラム・オブジェクトがすでに存在する場合に、新しいプログラム・オブジェクトが作成されるかどうかを指定します。考えられる値は次の通りです。

**\*YES** 新しいプログラム・オブジェクトが作成され、指定したライブラリー中の同じ名前の既存のプログラム・オブジェクトはライブラリーQRPLOBJに移動されます。

**\*NO** 指定したライブラリーに同じ名前のプログラム・オブジェクトがすでに存在する場合には、新しいプログラム・オブジェクトは作成されません。

上

---

## ターゲット・リリース (TGTRLS)

作成中のオブジェクトを使用したいオペレーティング・システムのリリース・レベルを指定します。

正確なリリース・レベルをVXRXXMXの形式で指定することができます。ここで、VX はバージョン、RXはリリース、およびMXはモディフィケーション・レベルです。たとえば、V3R1M0はバージョン3、リリース1、モディフィケーション・レベル0です。

**注:** ターゲット・システムでオブジェクトを使用するためには、作成コマンドで指定されたターゲット・リリース・レベルに合わせてオブジェクトを保管し、次にターゲット・システムでこれを復元しなければなりません。

たとえば、ユーザー・システムがV3R1M0で実行している時に、V2R3M0システムに配布するプログラム・オブジェクトを作成したい場合には、TGTRLS(V2R3M0)またはTGTRLS(\*PRV)でプログラムを作成し、これをTGTRLS(V2R3M0)またはTGTRLS(\*PRV)で保管して、このプログラムをV2R3M0システムで復元しなければなりません。このプログラム・オブジェクトは、V3R1M0システムに復元することもできます。

**注:** プログラムは、作成コマンドに指定したリリースより前のリリース・レベルで復元することができません。プログラムを実行することができる最も古いリリースを判別するためには、DSPPGMを使用してください。

### **\*CURRENT**

このオブジェクトは、現在ユーザーのシステムで実行中のオペレーティング・システムのリリースで使用するためのものです。たとえば、システムでV3R1M0が実行中の場合には、

\*CURRENTはV3R1M0が導入されたシステムでオブジェクトを使用することを意味します。また、オペレーティング・システムの後続のリリースが導入されたシステム上でこのオブジェクトを使用することもできます。

**\*PRV** このオブジェクトは、オペレーティング・システムの前のリリースのモディフィケーション・レベル0で使用するためのものです。たとえば、ユーザー・システムでV3R1M0が実行中の場合には、\*PRVはV2R3M0を導入したシステムでオブジェクトを使用することを意味します。また、オペレーティング・システムの後続のリリースが導入されたシステム上でこのオブジェクトを使用することもできます。

### **リリース・レベル**

リリースはVXRXXMXの形式で指定してください。オブジェクトは、指定したリリースのシステムまたはオペレーティング・システムの後続のリリースが導入されたシステムで使用することができます。

有効な値は、現行のバージョン、リリース、およびモディフィケーション・レベルによって異なり、新しいリリースごとに変更されます。有効な値のリストを参照するためには、コマンドのTGTRLSパラメーターでF4=プロンプトを押してください。

上

---

## **ユーザー・プロファイル (USRPRF)**

コンパイル済みCOBOLプログラムを実行するユーザー・プロファイルを指定します。プログラムを実行し、プログラムで使用できるオブジェクト（そのプログラムが各オブジェクトに対して持つ権限を含む）を制御するためには、プログラム所有者またはプログラム・ユーザーのプロファイルが使用されます。

考えられる値は次の通りです。

### **\*USER**

プログラムの実行時には、プログラムの使用者のユーザー・プロファイルが使用されます。

### **\*OWNER**

プログラムの実行時には、プログラムの所有者と使用者の両方のユーザー・プロファイルが使用されます。プログラムの実行時にオブジェクトを見つけてアクセスするためには、両方のユーザー・プロファイルを合わせたオブジェクト権限が使用されます。プログラムの実行中に作成されたオブジェクトは、そのプログラムのユーザーが所有します。

上

---

## **権限 (AUT)**

プログラム・オブジェクトに対して特定権限を持たないか、権限リスト上にないか、あるいはユーザー・グループがプログラム・オブジェクトに対して特定権限を持たないユーザーに与えられる権限を指定します。権限は、プログラムの作成後にGRTOBJAUT（オブジェクト権限認可）またはRVKOBJAUT（オブジェクト権限取り消し）コマンドを使用して、すべてのユーザーまたは特定のユーザーに対して変更することができます。

考えられる値は次の通りです。



### **\*LIBCRTAUT**

オブジェクトの共通権限は、ターゲット・ライブラリー（作成されたプログラム・オブジェクトを入れるライブラリー）のCRTAUTキーワードから取り出されます。この値はオブジェクトの作成時に決定されます。オブジェクトの作成後にライブラリーのCRTAUT値が変更された場合には、その新しい値は既存のオブジェクトに影響を与えません。

**\*ALL** 所有者に限定され、あるいは権限リスト管理権限によって制御されるものを除き、プログラム・オブジェクトに対するすべての操作を認可します。ユーザーはプログラム・オブジェクトの存在を制御し、その機密保護を指定し、それを変更し、その基本機能（プログラム・オブジェクトの実行およびデバッグなど）を実行することができます。

### **\*CHANGE**

所有者に限定され、あるいはオブジェクト権限およびオブジェクト管理権限によって制御されるものを除き、すべてのデータ権限およびプログラム・オブジェクトに対してすべての操作を実行する権限を提供します。ユーザーは、オブジェクトを変更し、オブジェクトに対する基本機能（プログラム・オブジェクトの実行およびデバッグなど）を実行することができます。

**\*USE** オブジェクト操作権および読み取り権限（プログラムの実行などのプログラム・オブジェクトに対する基本操作権限）を提供します。ユーザーはオブジェクトを変更することができません。

注：プログラムの変数の定様式ダンプを取得するには、プログラムに対して**\*USE**権限を持っていないければなりません。変数をダンプするには、さらにプログラムが検討可能な情報も持っていないければなりません。

一部のユーザーが変数のダンプができることを望まない場合は、ユーザーに**\*OBJOPR**だけに加えてプログラムに**\*EXECUTE**権限を与えてください。これはユーザーにプログラムの呼び出しは許可しますが、その変数をダンプすることは許可しません。

すべてのユーザーに変数のダンプができることを望まない場合は、プログラムの変更(CHGPGM)を使用して、プログラムの検討可能な情報を除去してください。

### **\*EXCLUDE**

一般ユーザーはプログラム・オブジェクトにアクセスすることができません。

### **権限リスト名**

ユーザーの権限リストの名前およびプログラムが追加する権限を入力してください。プログラム・オブジェクトはこの権限リストによって保護され、プログラム・オブジェクトの共通権限は**\*AUTL**に設定されます。CRTS36CBLを出す時には、システム上に権限リストが存在していなければなりません。

上

---

## **コンパイラー・デバッグ・ダンプ (DUMP)**

IBM COBOLデバッグ援助機能。(弊社技術員用)

上

---

## **中間テキスト・ダンプ (ITDUMP)**

コンパイル中の一定の時間にコンパイラーに内部テキストをダンプさせるIBMデバッグ援助機能。(弊社技術員用)

---

## 1 0 進数データの修正 (FIXDECDTA)

10進数データ・エラーを調べます。10進数データ・エラーは、プログラムが正しくないパックまたはゾーン10進数を使用しようとした時に起こります。考えられる値は次の通りです。

**\*YES** コンパイラーは監視するためのコードを生成して、10進数データ・エラーを訂正します。

**\*NO** コンパイラーは、10進数データ・エラーを監視するコードを生成しません。そのようなエラーが起こった場合には、プログラムは停止します。

---

## コピー・ファイル・ライブラリー (CPYLIB)

COPYステートメントに出合った時に検索されるライブラリーの名前を指定します。ライブラリーを指定しない場合には、#LIBRARYが使用されます。

使用できるライブラリーの値は次の通りです。

### \*CURLIB

現行ライブラリーとして指定したライブラリーが検索されます。現行ライブラリーとしてライブラリーを割り当てていない場合には、QGPLが使用されます。

**\*LIBL** COPYファイルが入っているライブラリーを見つけるために、ライブラリー・リストが検索されます。

### ライブラリー名

COPYファイルが入っているライブラリーの名前を入力してください。

---

## 例

### 例1: ソース・プログラムをシステム/36環境で使用するプログラム・オブジェクトにコンパイル

```
CRTS36CBL  PGM(MYLIB/XMPLE1) SRCFILE(MYLIB/QS36SRC)
           SRCMBR(XMPLE1)  OPTION(*SOURCE)
           TEXT('MY COBOL PROGRAM')
```

このコマンドはCOBOLコンパイラーを呼び出して、XMPLE1という名前のプログラムを作成します。ソース・プログラムはライブラリーMYLIB中のソース・ファイルQS36SRCのメンバーXMPLE1にあります。コンパイラー・リストが作成されます。

---

## エラー・メッセージ

### \*ESCAPE メッセージ

#### CBL1019

コンパイルが正常に行なわれなかった。プログラムは作成されません。

**SBL9001**

コンパイルが正常に行われませんでした。プログラムは作成されません。

**SBL9006**

TGTRLS(&1)が指定されたが、前のコンパイラーが導入されていない。

**SBL9007**

プロダクト・ライブラリーに損傷があるか、あるいはユーザーにはその使用が認可されていない。

上

# RPG IIプログラムの作成 (CRTS36RPG)

実行可能場所: すべての環境 (\*ALL)  
スレッド・セーフ: いいえ

パラメーター  
例  
エラー・メッセージ

システム/36互換RPG IIコンパイラーを呼び出すためには、システム/36 RPGプログラム作成(CRTS36RPG) コマンドを使用します。

上

## パラメーター

キーワード	記述	選択項目	ノート
PGM	プログラム	修飾オブジェクト名	オプション, 定位置 1
	修飾子 1: プログラム	名前, <u>*CTLSPEC</u>	
	修飾子 2: ライブラリー	名前, <u>*CURLIB</u>	
SRCFILE	ソース・ファイル	修飾オブジェクト名	オプション, 定位置 2
	修飾子 1: ソース・ファイル	名前, <u>QS36SRC</u>	
	修飾子 2: ライブラリー	名前, <u>*LIBL</u> , <u>*CURLIB</u>	
SRCMBR	ソース・メンバー	名前, <u>*PGM</u>	オプション, 定位置 3
GENLVL	生成重大度レベル	0-99, <u>21</u>	オプション
NEP	非終了プログラム	<u>*NO</u> , <u>*YES</u>	オプション
MRTMAX	最大MRT装置数	0-99, <u>0</u>	オプション
TEXT	テキスト'記述'	文字値, <u>*SRCMBRTXT</u> , <u>*BLANK</u>	オプション
OPTION	ソース・リスト・オプション	値 (最大 10 回の繰り返し): <u>*SOURCE</u> , <u>*NOSOURCE</u> , <u>*SRC</u> , <u>*NOSRC</u> , <u>*XREF</u> , <u>*NOXREF</u> , <u>*GEN</u> , <u>*NOGEN</u> , <u>*DUMP</u> , <u>*NODUMP</u> , <u>*SECLVL</u> , <u>*NOSECLVL</u> , <u>*CONSOLE</u> , <u>*NOCONSOLE</u>	オプション
GENOPT	生成オプション	値 (最大 10 回の繰り返し): <u>*LIST</u> , <u>*NOLIST</u> , <u>*XREF</u> , <u>*NOXREF</u> , <u>*ATR</u> , <u>*NOATR</u> , <u>*DUMP</u> , <u>*NODUMP</u> , <u>*PATCH</u> , <u>*NOPATCH</u> , <u>*OPTIMIZE</u> , <u>*NOOPTIMIZE</u>	オプション
PRTFILE	印刷ファイル	修飾オブジェクト名	オプション
	修飾子 1: 印刷ファイル	名前, <u>QSYSPRT</u>	
	修飾子 2: ライブラリー	名前, <u>*LIBL</u> , <u>*CURLIB</u>	
ICFLIB	ICFファイルのライブラリー	名前	オプション
REPLACE	プログラムの置き換え	<u>*YES</u> , <u>*NO</u>	オプション
TGTRLS	ターゲット・リリース	単純名, <u>*CURRENT</u> , <u>*PRV</u>	オプション
USRPRF	ユーザー・プロファイル	<u>*USER</u> , <u>*OWNER</u>	オプション

キーワード	記述	選択項目	ノーツ
AUT	権限	名前, *LIBCRTAUT, *ALL, *CHANGE, *USE, *EXCLUDE	オプション
PHSTRC	フェーズのトレース	*NO, *YES	オプション
ITDUMP	中間テキスト・ダンプ	文字値, *NONE	オプション
SNPDUMP	スナップ・ダンプ	文字値, *NONE	オプション
CODELIST	コード・リスト	文字値, *NONE, *ALL	オプション
FIXDECDTA	固定小数点データ	*YES, *NO	オプション

上

## プログラム (PGM)

コンパイル済みRPGプログラムの認識に使用されるライブラリー名およびプログラム名を指定します。

### \*CTLSPEC

システムによって使用される制御仕様の75-80桁目のプログラム名を指定します。

### プログラム名

プログラム・オブジェクトの認識に使用される名前を入力してください。

### \*CURLIB

現行ライブラリーが使用されます。現行ライブラリーが指定されていない場合には、#LIBRARYが使用されることになります。

### ライブラリー名

コンパイル済みプログラムが入れられるライブラリーの名前を入力してください。

上

## ソース・ファイル (SRCFILE)

ソース・プログラムが入っているファイルの名前およびそのファイルが入っているライブラリーの名前を指定します。

### QS36SRC

省略時のファイル名を指定します。

### ソース・ファイル名

コンパイルされるプログラムが入っているソース・ファイルの名前を入力してください。

**\*LIBL** ソース・ファイルが入っているライブラリーを見つけるために、システムがライブラリー・リストを検索します。

### \*CURLIB

現行ライブラリーが使用されます。現行ライブラリーが指定されていない場合には、#LIBRARYが使用されることになります。

### ライブラリー名

ソース・ファイルが入っているライブラリーの名前を入力してください。

上

---

## ソース・メンバー (SRCMBR)

ソース・ファイルのメンバーの名前を指定します。

**\*PGM** コンパイルされるソース・プログラムのメンバー名を指定します。

**ソース・ファイル・メンバー名**

ソース・プログラムが入っているメンバーの名前を入力してください。

上

---

## 生成重大度レベル (GENLVL)

見つかったエラーの重大度によって、プログラム・オブジェクトが生成されるかどうかを指定します。

**21** エラー・メッセージの重大度レベルが21またはこれ以上の場合には、プログラム・オブジェクトを生成しません。

**重大度レベル値**

0から99の数値を入力してください。RPGメッセージの重大度レベルの値は50を超えてはいけません。

上

---

## 非終止プログラム (NEP)

現行プログラムを非終止プログラムとするかどうかを指定します。

**\*NO** このプログラムが非終止プログラムでないことを指定します。

**\*YES** このプログラムが非終止プログラムであることを指定します。

上

---

## 最大MRT装置数 (MRTMAX)

要求元表示装置の最大数を指定します。

**0** このプログラムはSRTプログラムです。

**最大端末数**

要求端末の最大数を示す0から99の数値を入力してください。

上

---

## テキスト'記述' (TEXT)

ユーザーはプログラムおよびその機能を記述するテキストを入力することができます。

**\*SRCMBRTXT**

ソース・プログラム中のテキストを使用します。

**\*BLANK**

テキストを省略します。

## テキスト

50文字以下の記述をアポストロフィで囲んで入力してください。

上

---

## ソース・リスト・オプション (OPTION)

ソース・プログラムのコンパイル時に使用するオプションを指定します。

### **\*SOURCE**または**\*SRC**

コンパイル時のエラーのついたソース・リストを作成します。

### **\*NOSOURCE**または**\*NOSRC**

ソース・リストを作成しません。

### **\*XREF**

相互参照表およびキー・フィールド情報テーブルを作成します。

### **\*NOXREF**

相互参照表およびキー・フィールド情報テーブルを作成しません。\*NOSOURCEを指定した時には、これが省略時の値です。

### **\*GEN** 実行可能なプログラムを作成します。

### **\*NOGEN**

実行可能なプログラムを作成しません。

### **\*NODUMP**

エラーが起こった時にプログラム・テンプレートを印刷しません。

### **\*DUMP**

エラーが起こった時にプログラム・テンプレートを印刷します。

### **\*NOSECLVL**

第2レベル・メッセージを印刷しません。

### **\*SECLVL**

第1レベル・メッセージに続いて第2レベル・メッセージを印刷します。

### **\*CONSOLE**

コンパイル中のプログラムにCONSOLEファイルが含まれている場合に、CONSOLE ファイル用の画面様式を作成します。

### **\*NOCONSOLE**

CONSOLEファイル用の画面様式を作成しません。

上

---

## 生成オプション (GENOPT)

プログラム・オブジェクトの作成時に使用するオプションを指定します。

### **\*NOLIST**

プログラムの中間表現(IRP)のリストを作成しません。

### **\*LIST** プログラムの中間表現(IRP)のリストを作成します。



#### **\*NOXREF**

IRPの相互参照表を印刷しません。

#### **\*XREF**

IRPの相互参照表を印刷します。

#### **\*NOATR**

IRPの属性リストを作成しません。

**\*ATR** IRPの属性リストを作成します。

#### **\*NODUMP**

エラーが起こった時にプログラム・テンプレートを印刷しません。

#### **\*DUMP**

エラーが起こった時にプログラム・テンプレートを印刷します。

#### **\*NOPATCH**

コンパイル済みプログラム中にプログラム・パッチ域を予約しません。

#### **\*PATCH**

コンパイル済みプログラム中にプログラム・パッチ域を予約します。

#### **\*NOOPTIMIZE**

プログラムの処理効率を改善しません。

#### **\*OPTIMIZE**

プログラムの処理効率を改善します。

上

---

## **印刷ファイル (PRTFILE)**

コンパイル・リストが入れられるファイルの名前およびそのファイルが入っているライブラリーの名前を指定します。

#### **QSYSPRT**

ファイル名を指定しない場合には、コンパイル・リストがQSYSPRTファイルに入れられます。

#### **印刷ファイル名**

コンパイル・リストが入れられるファイルの名前を入力してください。

**\*LIBL** ライブラリーを見つけるために、システムはライブラリー・リストを検索します。

#### **\*CURLIB**

現行ライブラリーが使用されます。現行ライブラリーが指定されていない場合には、#LIBRARYが使用されることになります。

#### **ライブラリー名**

ファイルが入っているライブラリーの名前を入力してください。

上

---

## **ICFファイルのライブラリー (ICFLIB)**

プログラムで使用されるICFレコード様式定義が入っているライブラリーを指定します。

## ICFライブラリー名

ICFファイルのライブラリーの名前を入力してください。

上

---

## プログラムの置き換え (REPLACE)

指定されたライブラリーの中にすでに同じ名前のプログラムが存在している場合に、新しいプログラム・オブジェクトを作成するかどうかを指定します。

**\*YES** 新しいプログラム・オブジェクトが作成され、指定したライブラリーの中の同じ名前の既存のプログラム・オブジェクトはライブラリーQRPLOBJに移動されます。

**\*NO** 指定したライブラリーの中に同じ名前のプログラム・オブジェクトがあった場合には、新しいプログラム・オブジェクトは作成されません。

上

---

## ターゲット・リリース (TGTRLS)

作成されているオブジェクトを使用する予定のオペレーティング・システムのリリースを指定します。

\*CURRENTおよび\*PRV値の説明で示した例、および ターゲット・リリース 値を指定するときのリリースの指定方法は、VXRXXMXの形式です。ここで、VXはバージョン、RXはリリース、MXはモディフィケーション・レベルです。たとえば、V2R3M0は、バージョン2,リリース3,モディフィケーション・レベル0となります。

このパラメーターに対する有効な値はリリースごとに変わります。

考えられる値は次の通りです。

### **\*CURRENT**

オブジェクトは、ユーザーのシステムで現在実行されているオペレーティング・システムのリリースで使用されます。たとえば、V2R3M5がシステム上で実行されている場合には、\*CURRENTは、ユーザーがV2R3M5の導入されているシステムでオブジェクトを使用する予定であることを意味します。また、オペレーティング・システムの後続のリリースが導入されたシステム上でこのオブジェクトを使用することもできます。

**\*PRV** オブジェクトは、モディフィケーション・レベルが0であるオペレーティング・システムの前のリリースで使用されます。たとえば、V2R3M5がユーザーのシステムで実行されている場合には、\*PRVはオブジェクトをV2R2M0が導入されているシステムで使用する予定であることを意味します。また、オペレーティング・システムの後続のリリースが導入されたシステム上で、このオブジェクトを使用することもできます。

### ターゲット・リリース

リリースをVXRXXMXの形式で指定してください。オブジェクトは、指定されたリリースのシステム、またはオペレーティング・システムの後続のリリースが導入されているシステムで使用することができます。

有効な値は、現行のバージョン、リリース、およびモディフィケーション・レベルによって異なり、新しいリリースごとに変わります。このコマンドでサポートされている最も古いリリースよりもっと前のリリース・レベルを指定した場合には、エラー・メッセージが出され、サポートされている最も古いリリースを表示します。

注: プログラムは、作成コマンドに指定したリリースより前のリリース・レベルで復元することができます。プログラムを実行することができる最も古いリリースを判別するためには、DSPPGMを使用してください。

上

---

## ユーザー・プロファイル (USRPRF)

コンパイル済みRPGプログラムを実行するユーザー・プロファイルを指定します。

### **\*USER**

プログラムの実行時にプログラムの使用者のプロファイルを使用します。

### **\*OWNER**

プログラムの実行時にプログラムの所有者と使用者の両方のユーザー・プロファイルを使用します。プログラムがすでに存在している場合には、USRPRFパラメーターは更新されません。

上

---

## 権限 (AUT)

このプログラムのために認可する権限を指定します。

### **\*LIBCRTAUT**

作成されたオブジェクトに対する省略時の共通権限は、ターゲット・ライブラリーと対応したCRTAUTキーワードから取り出されます。この値は作成時に決定されます。作成後にライブラリーのCRTAUTキーワードの値が変わっても、新しい値は既存のオブジェクトに影響しません。

**\*ALL** プログラムに対する完全な権限（オブジェクト所有権の移転を除く）が認められます。

### **\*CHANGE**

オブジェクトの存在および管理を取り扱う操作以外のすべての操作が認められます。

**\*USE** コンパイル済みプログラムの読み取りまたは実行は認められますが、デバッグまたは変更はできません。

注: プログラムの変数の定様式ダンプを取得するには、プログラムに対して**\*USE**権限を持っていないければなりません。変数をダンプするには、さらにプログラムが検討可能な情報も持っていません。

一部のユーザーが変数のダンプができることを望まない場合は、ユーザーに**\*OBJOPR**だけに加えてプログラムに**\*EXECUTE**権限を与えてください。これはユーザーにプログラムの呼び出しは許可しますが、その変数をダンプすることは許可しません。

すべてのユーザーに変数のダンプができることを望まない場合は、プログラムの変更(CHGPGM)を使用して、プログラムの検討可能な情報を除去してください。

### **\*EXCLUDE**

権限は認められません。

### 権限リスト

オブジェクトを保護する権限リストの名前。共通権限はAUTLになります。

上

---

## フェーズのトレース (PHSTRC)

コンパイラー・フェーズのトレース情報をリストに含めるかどうかを指定します。

**\*NO** コンパイラー・フェーズの情報を含めません。

**\*YES** コンパイラー・フェーズの情報を含めます。

上

---

## 中間テキスト・ダンプ (ITDUMP)

指定されたコンパイラー・フェーズの中間テキストの動的リストを作成します。

**\*NONE**

中間テキスト・ダンプを作成しません。

**フェーズ名**

1-25個のフェーズ名の4-5桁目を入力してください。

上

---

## スナップ・ダンプ (SNPDUMP)

1つまたは複数の指定されたフェーズのコンパイルの後に主要データ域および中間テキストのリストを作成します。

**\*NONE**

スナップ・ダンプを作成しません。

**フェーズ名**

1-25個のフェーズ名の4-5桁目を入力してください。

上

---

## コード・リスト (CODELIST)

指定されたコンパイラー・フェーズのプログラムの中間表現(IRP)の動的リストを作成します。

**\*NONE**

IRPリストを作成しません。

**\*ALL** 各コンパイラー・フェーズのIRPリストを作成します。

**フェーズ名**

1-25個のフェーズ名の4-5桁目を入力してください。

上

---

## 固定小数点データ (FIXDECDTA)

正しくない10進数データを訂正するか、あるいはエラーを通知するかを指定します。

**\*YES** A-Fの範囲の数値をゼロに、正しくない符号をプラスにセットすることによって、正しくないデータが訂正されます。

**\*NO** 正しくないデータを訂正せずに、プログラムにエラーを通知します。

上

---

## 例

### 例1: ソース・プログラムをプログラム・オブジェクトにコンパイル

```
CRTS36RPG  PGM(MYLIB/XMPLE1)
            SRCFILE(MYLIB/QS36SRC) SRCMBR(XMPLE1)
            OPTION(*SOURCE) TEXT('MY RPG II PROGRAM')
```

このコマンドはSYSTEM/36互換RPG IIコンパイラを呼び出して、XMPLE1という名前のプログラムを作成します。ソース・プログラムはライブラリーMYLIB中のソース・ファイルQS36SRCのメンバーXMPLE1にあります。コンパイラ・リストが作成されます。

上

---

## エラー・メッセージ

### \*ESCAPE メッセージ

#### QRG9001

コンパイルが正常に実行されなかった。プログラムは作成されません。

#### QRG9004

TGTRLSオプションに指定されたリリース&1はサポートされていない。

上



## コンソール表示装置ファイル作成 (CRTS36RPGR)

実行可能場所: すべての環境 (\*ALL)

スレッド・セーフ: いいえ

パラメーター

例

エラー・メッセージ

CONSOLE表示装置ファイルを作成するためには、システム/36 RPGR作成(CRTS36RPGR)コマンドを使用します。

上

### パラメーター

キーワード	記述	選択項目	ノート
<b>SRCMBR</b>	ソース・メンバー	名前	必須, 定位置 1
<b>SRCFILE</b>	ソース・ファイル	修飾オブジェクト名	オプション, 定位置 2
	修飾子 1: ソース・ファイル	名前, <u>QS36SRC</u>	
	修飾子 2: ライブラリー	名前, <u>*CURLIB</u>	
<b>OUTLIB</b>	出力ライブラリー	名前, <u>*CURLIB</u>	オプション, 定位置 3
<b>FMTSRCF</b>	S/36形式を受け取るファイル	単一値: *NONE その他の値: 修飾オブジェクト名	オプション, 定位置 4
	修飾子 1: S/36形式を受け取るファイル	名前, <u>*SRCFILE</u>	
	修飾子 2: ライブラリー	名前, <u>*SRCLIB</u>	
<b>FMTMBR</b>	形式を受け取るメンバー	名前, <u>*CRTDFT</u>	オプション, 定位置 5
<b>DDSSRCF</b>	DDSを受け取るファイル	単一値: *NONE その他の値: 修飾オブジェクト名	オプション
	修飾子 1: DDSを受け取るファイル	名前, <u>*SRCFILE</u>	
	修飾子 2: ライブラリー	名前, <u>*SRCLIB</u>	
<b>DDSMBR</b>	DDSを受け取るメンバー	名前, <u>*CRTDFT</u>	オプション
<b>GEN</b>	コンソール形式の生成	<u>*YES</u> , *NO	オプション
<b>REPLACE</b>	出力メンバーの置き換え	<u>*YES</u> , *NO	オプション
<b>AUT</b>	権限	名前, <u>*LIBCRTAUT</u> , *ALL, *CHANGE, *USE, *EXCLUDE	オプション

上

### ソース・メンバー (SRCMBR)

RPG IIプログラム仕様が入っているソース・メンバーを指定します。

ソース・ファイル・メンバー名

ソース・プログラムが入っているメンバーの名前を入力してください。

上



---

## ソース・ファイル (SRCFILE)

RPG IIソース・メンバーが入っているファイルおよびそのファイルが入っているライブラリーを指定します。

### QS36SRC

省略時のファイル名を指定します。

#### ソース・ファイル名

ソース・メンバーが入っているソース・ファイルの名前を入力してください。

### \*CURLIB

現行ライブラリーが使用されます。現行ライブラリーが指定されていない場合には、#LIBRARYが使用されることになります。

#### ライブラリー名

ソース・ファイルが入っているライブラリーの名前を入力してください。

上

---

## 出力ライブラリー (OUTLIB)

画面様式のオブジェクト・メンバーが入っているライブラリーを指定します。

### \*CURLIB

現行ライブラリーが使用されます。現行ライブラリーが指定されていない場合には、#LIBRARYが使用されることになります。

#### 出力ライブラリー名

オブジェクト・メンバーが入れられるライブラリーの名前を入力してください。

上

---

## S/36形式を受け取るファイル (FMTSRCF)

S仕様およびD仕様を含むメンバーが入れられるファイルおよびそのファイルが入れられるライブラリーを指定します。

### \*SRCFILE

ソース・ファイル・パラメーター(SRCFILE)に指定されたファイル名を使用します。

### \*NONE

S仕様およびD仕様が作成されないことを指定します。

#### S仕様およびD仕様のソース・ファイル名

S仕様およびD仕様が入れられるファイルの名前を入力してください。

### \*SRCLIB

ソース・ファイル・ライブラリー名(SRCFILEパラメーター)に指定された同じライブラリーを使用します。

#### S仕様およびD仕様のライブラリー名

S仕様およびD仕様のソース・ファイルが入れられるライブラリーの名前を入力してください。

上

---

## 形式を受け取るメンバー (FMTMBR)

S仕様およびD仕様が入れられるメンバーを指定します。

### \*CRTDFT

指定されたソース・メンバー名(SRCMBR)の終わりに'FM'を追加して使用します。

### S仕様およびD仕様のメンバー名

S仕様およびD仕様が入れられるメンバーの名前を入力してください。

上

---

## DDSを受け取るファイル (DDSSRCF)

DDS仕様を含むメンバーが入れられるファイルおよびそのファイルが入れられるライブラリーを指定します。

### \*SRCFILE

ソース・ファイル・パラメーター(SRCFILE)に指定されたファイル名を使用します。

### \*NONE

DDS仕様が作成されないことを指定します。

### DDSソース・ファイル名

DDSメンバーが入れられるソース・ファイルの名前を入力してください。

### \*SRCLIB

ソース・ファイル・ライブラリー名(SRCFILEパラメーター)に指定された同じライブラリーを使用します。

### DDSライブラリー名

ソースDDSファイルが入れられるライブラリーの名前を入力してください。

上

---

## DDSを受け取るメンバー (DDSMBR)

DDS仕様が入れられるメンバーを指定します。

### \*CRTDFT

指定されたソース・メンバー名(SRCMBR)の終わりに'A'を追加して使用します。

### DDSメンバー名

DDS仕様が入れられるメンバーの名前を入力してください。

上

---

## コンソール形式の生成 (GEN)

CONSOLEファイル用の画面様式が作成されることを指定します。

\*YES 画面様式が生成されることを指定します。

\*NO 画面様式が生成されないことを指定します。

---

## 出力メンバーの置き換え (REPLACE)

指定されたライブラリーの中に同じ名前のファイルがすでに存在している場合に、新しいCONSOLE表示装置ファイルを作成するかどうかを指定します。

**\*YES** 新しいCONSOLE表示装置ファイルが作成され、指定したライブラリーの中の同じ名前の既存のCONSOLE表示装置ファイルはすべて削除されることを指定します。

**\*NO** 指定したライブラリーに同じ名前のメンバーが存在している場合には、新しいCONSOLE表示装置ファイルが作成されないことを指定します。

---

## 権限 (AUT)

このプログラムのために認可する権限を指定します。

### **\*LIBCRTAUT**

作成されたオブジェクトに対する省略時の共通権限は、ターゲット・ライブラリーと対応したCRTAUTキーワードから取り出されます。この値は作成時に決定されます。作成後にライブラリーのCRTAUTキーワードの値が変わっても、新しい値は既存のオブジェクトに影響しません。

**\*ALL** プログラムに対する完全な権限（オブジェクト所有権の移転を除く）が認められます。

### **\*CHANGE**

オブジェクトの存在および管理を取り扱う操作以外のすべての操作が認められます。

**\*USE** コンパイル済みプログラムの読み取りまたは実行は認められますが、デバッグまたは変更はできません。

### **\*EXCLUDE**

権限は認められません。

### 権限リスト

オブジェクトを保護する権限リストの名前。共通権限は\*AUTLになります。

---

## 例

### 例1: ソース・プログラムをプログラム・オブジェクトにコンパイル

```
CRTS36RPGR  SRCMBR(XMPLE1) SRCFILE(MYLIB/QS36SRC)
             OUTLIB(MYLIB)
             FMTSRCF(*SRCLIB/QS36DDSSRC) FMTMBR(*CRTDFT)
             DDSSRCF(*SRCLIB/QDDSSRC) DDSMBR(*CRTDFT)
             REPLACE(*YES)
```

このコマンドはRPG IIソース・メンバーXMPLE1からCONSOLEファイル用の表示装置ファイルを作成します。ファイルの名前はRPGソースの制御(H)仕様の75桁目の名前の後にFMが続きます。例えば、制御仕様の75-80桁目にXMPLE1が含まれる場合は、表示装置ファイルはXMPLE1FMと名前付けられます。ソース・メンバーXMPLE1FMは、ファイルMYLIB/QS36DDSSRCに作成されて、SYSTEM/36様式SおよびD仕様）を入

れます。ソース・メンバーXMPLE1Aは、ファイルMYLIB/QDDSSRCに作成されて、ファイルMYLIB/XMPLE1FMの作成に使用したDSPF DDSを入れます。

上

---

## エラー・メッセージ

なし

上



## S/36 RPG II報告書簡易作成 (CRTS36RPT)

実行可能場所: すべての環境 (\*ALL)

スレッド・セーフ: いいえ

パラメーター

例

エラー・メッセージ

システム/36互換RPG IIコンパイラを呼び出すためには、システム/36報告書簡易作成(CRTS36RPT)コマンドを使用します。

上

### パラメーター

キーワード	記述	選択項目	ノーツ
PGM	プログラム	修飾オブジェクト名	オプション, 定位置 1
	修飾子 1: プログラム	名前, <u>*CTLSPEC</u>	
	修飾子 2: ライブラリー	名前, <u>*CURLIB</u>	
SRCFILE	ソース・ファイル	修飾オブジェクト名	オプション, 定位置 2
	修飾子 1: ソース・ファイル	名前, <u>QS36SRC</u>	
	修飾子 2: ライブラリー	名前, <u>*LIBL</u> , <u>*CURLIB</u>	
SRCMBR	ソース・メンバー	名前, <u>*PGM</u>	オプション, 定位置 3
GENLVL	生成重大度レベル	0-99, <u>21</u>	オプション
NEP	非終了プログラム	<u>*NO</u> , <u>*YES</u>	オプション
MRTMAX	最大MRT装置数	0-99, <u>0</u>	オプション
TEXT	テキスト'記述'	文字値, <u>*SRCMBRTXT</u> , <u>*BLANK</u>	オプション
OPTION	ソース・リスト・オプション	値 (最大 10 回の繰り返し): <u>*SOURCE</u> , <u>*NOSOURCE</u> , <u>*SRC</u> , <u>*NOSRC</u> , <u>*XREF</u> , <u>*NOXREF</u> , <u>*GEN</u> , <u>*NOGEN</u> , <u>*DUMP</u> , <u>*NODUMP</u> , <u>*SECLVL</u> , <u>*NOSECLVL</u> , <u>*CONSOLE</u> , <u>*NOCONSOLE</u>	オプション
GENOPT	生成オプション	値 (最大 10 回の繰り返し): <u>*LIST</u> , <u>*NOLIST</u> , <u>*XREF</u> , <u>*NOXREF</u> , <u>*ATR</u> , <u>*NOATR</u> , <u>*DUMP</u> , <u>*NODUMP</u> , <u>*PATCH</u> , <u>*NOPATCH</u> , <u>*OPTIMIZE</u> , <u>*NOOPTIMIZE</u>	オプション
PRTFILE	印刷ファイル	修飾オブジェクト名	オプション
	修飾子 1: 印刷ファイル	名前, <u>QSYSPRT</u>	
	修飾子 2: ライブラリー	名前, <u>*LIBL</u> , <u>*CURLIB</u>	
ICFLIB	ICFファイルのライブラリー	名前	オプション

キーワード	記述	選択項目	ノーツ
<b>RPTOPT</b>	報告書簡易作成機能OPT	値 (最大 10 回の繰り返し): *SOURCE, *NOSOURCE, *SRC, *NOSRC, *FLOW, *NOFLOW, *AST, *NOAST, *DATE, *NODATE, *COMPILE, *NOCOMPILE, *SECLVL, *NOSECLVL	オプション
<b>OUTFILE</b>	報告書簡易作成機能用SAVF	修飾オブジェクト名	オプション
	修飾子 1: 報告書簡易作成機能用SAVF	名前, <u>*NONE</u>	
	修飾子 2: ライブラリー	名前, <u>*LIBL</u> , *CURLIB	
<b>OUTMBR</b>	報告書簡易作成機能用SAVF MBR	名前, <u>*NONE</u>	オプション
<b>REPLACE</b>	プログラムの置き換え	<u>*YES</u> , *NO	オプション
<b>TGTRLS</b>	ターゲット・リリース	単純名, <u>*CURRENT</u> , *PRV	オプション
<b>USRPRF</b>	ユーザー・プロファイル	<u>*USER</u> , *OWNER	オプション
<b>AUT</b>	権限	名前, <u>*LIBCRTAUT</u> , *ALL, *CHANGE, *USE, *EXCLUDE	オプション
<b>PHSTRC</b>	フェーズのトレース	<u>*NO</u> , *YES	オプション
<b>ITDUMP</b>	中間テキスト・ダンプ	文字値, <u>*NONE</u>	オプション
<b>SNPDUMP</b>	スナップ・ダンプ	文字値, <u>*NONE</u>	オプション
<b>CODELIST</b>	コード・リスト	文字値, <u>*NONE</u> , *ALL	オプション
<b>FIXDECDTA</b>	固定小数点データ	<u>*YES</u> , *NO	オプション

上

## プログラム (PGM)

コンパイル済みRPGプログラムの認識に使用されるライブラリー名およびプログラム名を指定します。

### \*CTLSPEC

システムによって使用される制御仕様の75-80桁目のプログラム名を指定します。

### プログラム名

プログラム・オブジェクトの認識に使用される名前を入力してください。

### \*CURLIB

現行ライブラリーが使用されます。現行ライブラリーが指定されていない場合には、#LIBRARYが使用されることになります。

### ライブラリー名

コンパイル済みプログラムが入れられるライブラリーの名前を入力してください。

上

## ソース・ファイル (SRCFILE)

ソース・プログラムが入っているファイルの名前およびそのファイルが入っているライブラリーの名前を指定します。



## **QS36SRC**

省略時のファイル名を指定します。

### **ソース・ファイル名**

コンパイルされるプログラムが入っているソース・ファイルの名前を入力してください。

**\*LIBL** ソース・ファイルが入っているライブラリーを見つけるために、システムがライブラリー・リストを検索します。

### **\*CURLIB**

現行ライブラリーが使用されます。現行ライブラリーが指定されていない場合には、**#LIBRARY**が使用されることになります。

### **ライブラリー名**

ソース・ファイルが入っているライブラリーの名前を入力してください。

上

---

## **ソース・メンバー (SRCMBR)**

ソース・ファイルのメンバーの名前を指定します。

**\*PGM** プログラム名(PGMパラメーター) に指定された名前を使用します。

### **ソース・ファイル・メンバー名**

ソース・プログラムが入っているメンバーの名前を入力してください。

上

---

## **生成重大度レベル (GENLVL)**

見つかったエラーの重大度によって、プログラム・オブジェクトが生成されるかどうかを指定します。

**21** エラー・メッセージの重大度レベルが21またはこれ以上の場合には、プログラム・オブジェクトを生成しません。

### **重大度レベル値**

0から99の数値を入力してください。RPGメッセージの重大度レベルの値は50を超えてはいけません。

上

---

## **非終了プログラム (NEP)**

現行プログラムを非終了プログラムとするかどうかを指定します。

**\*NO** このプログラムが非終了プログラムでないことを指定します。

**\*YES** このプログラムが非終了プログラムであることを指定します。

上

---

## 最大MRT装置数 (MRTMAX)

要求元表示装置の最大数を指定します。

0 このプログラムはSRTプログラムです。

### 最大端末数

要求端末の最大数を示す0から99の数値を入力してください。

上

---

## テキスト'記述' (TEXT)

ユーザーはプログラムおよびその機能を記述するテキストを入力することができます。

### \*SRCMBRTXT

ソース・プログラム中のテキストを使用します。

### \*BLANK

テキストを省略します。

### テキスト

50文字以下の記述をアポストロフィで囲んで入力してください。

上

---

## ソース・リスト・オプション (OPTION)

ソース・プログラムのコンパイル時に使用するオプションを指定します。

### \*SOURCEまたは\*SRC

コンパイル時のエラーのついたソース・リストを作成します。

### \*NOSOURCEまたは\*NOSRC

ソース・リストを作成しません。

### \*XREF

相互参照表およびキー・フィールド情報テーブルを作成します。

### \*NOXREF

ソース・リストおよびキー・フィールド情報テーブルは作成されません。\*NOSOURCEを指定した時には、これが省略時の値です。

\*GEN 実行可能なプログラムを作成します。

### \*NOGEN

実行可能なプログラムを作成しません。

### \*NODUMP

エラーが起こった時にプログラム・テンプレートを印刷しません。

### \*DUMP

エラーが起こった時にプログラム・テンプレートを印刷します。

### \*NOSECLVL

第2レベル・メッセージを印刷しません。

**\*SECLVL**

第1レベル・メッセージに続いて第2レベル・メッセージを印刷します。

**\*CONSOLE**

コンパイル中のプログラムにCONSOLEファイルが含まれている場合に、CONSOLE ファイル用の画面様式を作成します。

**\*NOCONSOLE**

CONSOLEファイル用の画面様式を作成しません。

上

---

## 生成オプション (GENOPT)

プログラム・オブジェクトの作成時に使用するオプションを指定します。

**\*NOLIST**

プログラムの中間表現(IRP)のリストを作成しません。

**\*LIST** プログラムの中間表現(IRP)のリストを作成します。

**\*NOXREF**

IRPの相互参照表を印刷しません。

**\*XREF**

IRPの相互参照表を印刷します。

**\*NOATR**

IRPの属性リストを作成しません。

**\*ATR** IRPの属性リストを作成します。

**\*NODUMP**

エラーが起こった時にプログラム・テンプレートを印刷しません。

**\*DUMP**

エラーが起こった時にプログラム・テンプレートを印刷します。

**\*NOPATCH**

コンパイル済みプログラム中にプログラム・パッチ域を予約しません。

**\*PATCH**

コンパイル済みプログラム中にプログラム・パッチ域を予約します。

**\*NOOPTIMIZE**

プログラムの処理効率を改善しません。

**\*OPTIMIZE**

プログラムの処理効率を改善します。

上

---

## 印刷ファイル (PRTFILE)

コンパイル・リストが入れられるファイルの名前およびそのファイルが入っているライブラリーの名前を指定します。

## **QSYSPRT**

ファイル名を指定しない場合には、コンパイル・リストがQSYSPRTファイルに入れられます。

### **印刷ファイル名**

コンパイル・リストが入れられるファイルの名前を入力してください。

**\*LIBL** ライブラリーを見つけるために、システムはライブラリー・リストを検索します。

### **\*CURLIB**

現行ライブラリーが使用されます。現行ライブラリーが指定されていない場合には、#LIBRARYが使用されることになります。

### **ライブラリー名**

ファイルが入っているライブラリーの名前を入力してください。

上

---

## **ICFファイルのライブラリー (ICFLIB)**

プログラムで使用されるICFレコード様式定義が入っているライブラリーを指定します。

### **ICFライブラリー名**

ICFファイルのライブラリーの名前を入力してください。

上

---

## **報告書簡易作成機能OPT (RPTOPT)**

報告書簡易作成機能のソース・プログラムのコンパイル時に使用するオプションを指定します。

### **\*NOSOURCEまたは**\*NOSRC****

ソース・リストを作成しません。

### **\*SOURCE**または**SRC**

ソース・リストを作成します。

### **\*NOFLOW**

主要ルーチンのフローを書き出しません。

### **\*FLOW**

主要ルーチンのフローを書き出します。

### **\*NOAST**

合計出力行にアスタリスクを作成しません。

**\*AST** 合計出力行にアスタリスクを作成します。

### **\*DATE**

最初の\*AUTOページ見出し行に日付およびページ番号を印刷します。

### **\*NODATE**

最初の\*AUTOページ見出し行に日付およびページ番号を印刷しません。

### **\*COMPILE**

報告書簡易作成機能のソースのコンパイル後にRPGコンパイラーを呼び出します。

#### **\*NOCOMPIL**

報告書簡易作成機能のソースのコンパイル後にRPGコンパイラーを呼び出しません。

#### **\*NOSECLVL**

第2レベル・メッセージを印刷しません。

#### **\*SECLVL**

第1レベル・メッセージに続いて第2レベル・メッセージを印刷します。

上

---

## **報告書簡易作成機能用SAVF (OUTFILE)**

コンパイルされた報告書簡易作成機能プログラムが入れられる出力ファイルの名前およびそのファイルが入れられるライブラリーの名前を指定します。

#### **\*NONE**

作成されるソース・プログラムをコンパイラーに渡すためにQTEMP中にファイルを作成します。

#### **出力ファイル名**

正しいRPG IIソース・プログラムが入っているファイルの名前を入力してください。

**\*LIBL** ソース・ファイルが入っているライブラリーを見つけるために、システムがライブラリー・リストを検索します。

#### **\*CURLIB**

現行ライブラリーが使用されます。現行ライブラリーが指定されていない場合には、#LIBRARYが使用されることになります。

#### **ライブラリー名**

ソース・ファイルが入っているライブラリーの名前を入力してください。

上

---

## **報告書簡易作成機能用SAVF MBR (OUTMBR)**

報告書簡易作成機能プログラムからの出力が入れられるファイルのメンバーの名前を指定します。

#### **\*NONE**

メンバー名としてファイル中の最初のメンバーを使用します。

#### **ファイル・メンバー名**

報告書簡易作成機能の出力を入れるメンバーの名前を入力してください。

上

---

## **プログラムの置き換え (REPLACE)**

指定されたライブラリーの中にすでに同じ名前のプログラムが存在している場合に、新しいプログラム・オブジェクトを作成するかどうかを指定します。

**\*YES** 新しいプログラム・オブジェクトが作成され、指定したライブラリーの中の同じ名前の既存のプログラム・オブジェクトはライブラリーQRPLOBJに移動されます。

**\*NO** 指定したライブラリーの中に同じ名前のプログラム・オブジェクトがあった場合には、新しいプログラム・オブジェクトは作成されません。

上

---

## ターゲット・リリース (TGTRLS)

作成されているオブジェクトを使用する予定のオペレーティング・システムのリリースを指定します。

**\*CURRENT**および**\*PRV**値の説明で示した例、および ターゲット・リリース 値を指定するときのリリースの指定方法は、VXRXXMXの形式です。ここで、VXはバージョン、RXはリリース、MXはモディフィケーション・レベルです。たとえば、V2R3M0は、バージョン2,リリース3,モディフィケーション・レベル0となります。

このパラメーターに対する有効な値はリリースごとに変わります。

考えられる値は次の通りです。

### **\*CURRENT**

オブジェクトは、ユーザーのシステムで現在実行されているオペレーティング・システムのリリースで使用されます。たとえば、V2R3M5がシステム上で実行されている場合には、**\*CURRENT**は、ユーザーがV2R3M5の導入されているシステムでオブジェクトを使用する予定であることを意味します。また、オペレーティング・システムの後続のリリースが導入されたシステム上でこのオブジェクトを使用することもできます。

**\*PRV** オブジェクトは、モディフィケーション・レベルが0であるオペレーティング・システムの前のリリースで使用されます。たとえば、V2R3M5がユーザーのシステムで実行されている場合には、**\*PRV**はオブジェクトをV2R2M0が導入されているシステムで使用する予定であることを意味します。また、オペレーティング・システムの後続のリリースが導入されたシステム上で、このオブジェクトを使用することもできます。

### ターゲット・リリース

リリースをVXRXXMXの形式で指定してください。オブジェクトは、指定されたリリースのシステム、またはオペレーティング・システムの後続のリリースが導入されているシステムで使用することができます。

有効な値は、現行のバージョン、リリース、およびモディフィケーション・レベルによって異なり、新しいリリースごとに変わります。このコマンドでサポートされている最も古いリリースよりもっと前のリリース・レベルを指定した場合には、エラー・メッセージが出され、サポートされている最も古いリリースを表示します。

注: プログラムは、作成コマンドに指定したリリースより前のリリース・レベルで復元することができます。プログラムを実行することができる最も古いリリースを判別するためには、DSPPGMを使用してください。

上

---

## ユーザー・プロファイル (USRPRF)

コンパイル済みRPGプログラムを実行するユーザー・プロファイルを指定します。

### **\*USER**

プログラムの実行時にプログラムの使用者のプロファイルを使用します。

## \*OWNER

プログラムの実行時にプログラムの所有者と使用者の両方のユーザー・プロファイルを使用します。プログラムがすでに存在している場合には、USRPRFパラメーターは更新されません。

上

---

## 権限 (AUT)

このプログラムのために認可する権限を指定します。

### \*LIBCRTAUT

作成されたオブジェクトに対する省略時の共通権限は、ターゲット・ライブラリーと対応したCRTAUTキーワードから取り出されます。この値は作成時に決定されます。作成後にライブラリーのCRTAUTキーワードの値が変わっても、新しい値は既存のオブジェクトに影響しません。

\*ALL プログラムに対する完全な権限（オブジェクト所有権の移転を除く）が認められます。

### \*CHANGE

オブジェクトの存在および管理を取り扱う操作以外のすべての操作が認められます。

\*USE コンパイル済みプログラムの読み取りまたは実行は認められますが、デバッグまたは変更はできません。

注: プログラムの変数の定様式ダンプを取得するには、プログラムに対して\*USE権限を持っていないければなりません。変数をダンプするには、さらにプログラムが検討可能な情報も持っていないければなりません。

一部のユーザーが変数のダンプができることを望まない場合は、ユーザーに\*OBJOPRだけに加えてプログラムに\*EXECUTE権限を与えてください。これはユーザーにプログラムの呼び出しは許可しますが、その変数をダンプすることは許可しません。

すべてのユーザーに変数のダンプができることを望まない場合は、プログラムの変更(CHGPGM)を使用して、プログラムの検討可能な情報を除去してください。

### \*EXCLUDE

権限は認められません。

### 権限リスト

オブジェクトを保護する権限リストの名前。共通権限は\*AUTLになります。

上

---

## フェーズのトレース (PHSTRC)

コンパイラー・フェーズのトレース情報をリストに含めるかどうかを指定します。

\*NO コンパイラー・フェーズの情報を含めません。

\*YES コンパイラー・フェーズの情報を含めます。

上

---

## 中間テキスト・ダンプ (ITDUMP)

指定されたコンパイラー・フェーズの中間テキストの動的リストを作成します。



#### **\*NONE**

中間テキスト・ダンプを作成しません。

#### **フェーズ名**

1-25個のフェーズ名の4-5桁目を入力してください。

上

---

## **スナップ・ダンプ (SNPDUMP)**

1つまたは複数の指定されたフェーズのコンパイルの後に主要データ域および中間テキストのリストを作成します。

#### **\*NONE**

スナップ・ダンプを作成しません。

#### **フェーズ名**

1-25個のフェーズ名の4-5桁目を入力してください。

上

---

## **コード・リスト (CODELIST)**

指定されたコンパイラ・フェーズのプログラムの中間表現(IRP)の動的リストを作成します。

#### **\*NONE**

IRPリストを作成しません。

**\*ALL** 各コンパイラ・フェーズのIRPリストを作成します。

#### **フェーズ名**

1-25個のフェーズ名の4-5桁目を入力してください。

上

---

## **固定小数点データ (FIXDECDTA)**

正しくない10進数データを訂正するか、あるいはエラーを通知するかを指定します。

**\*YES** A-Fの範囲の数値をゼロに、正しくない符号をプラスにセットすることによって、正しくないデータが訂正されます。

**\*NO** 正しくないデータを訂正せずに、プログラムにエラーを通知します。

上

---

## **例**

**例1:** ソース・プログラムをプログラム・オブジェクトにコンパイル

```
CRTS36RPT  PGM(MYLIB/XMPLE1)
            SRCFILE(MYLIB/QS36SRC) SRCMBR(XMPLE1)
            OPTION(*SOURCE)
            TEXT('MY RPG II AUTO REPORT PROGRAM')
```

このコマンドはSYSTEM/36互換RPG IIコンパイラを呼び出して、XMPLE1という名前の報告書簡易作成プログラムを作成します。ソース・プログラムはライブラリーMYLIB中のソース・ファイルQS36SRCのメンバーXMPLE1にあります。コンパイラ・リストが作成されます。

上

---

## エラー・メッセージ

### \*ESCAPE メッセージ

#### **RPT9001**

報告書簡易作成機能が正常に実行されなかった。

#### **QRG9004**

TGTRLSオプションに指定されたリリース&1はサポートされていない。

#### **RPT0082**

重大エラーが起こったために、報告書簡易作成機能による生成は打ち切られた。

上



## RPGソースの変換 (CVTRPGSRC)

実行可能場所: すべての環境 (\*ALL)

スレッド・セーフ: いいえ

パラメーター

例

エラー・メッセージ

## RPGソースの変換

RPGソース変換(CVTRPGSRC)コマンドは、RPG IIIまたはRPG/400ソース・コードをILE RPGソース・コードに変換します。

上

## パラメーター

キーワード	記述	選択項目	ノート
FROMFILE	FROMファイル	修飾オブジェクト名	必須, 定位置 1
	修飾子 1: FROMファイル	名前	
	修飾子 2: ライブラリー	名前, *LIBL, *CURLIB	
FROMMBR	FROMメンバー	総称名, 名前, *ALL	必須, 定位置 2
TOFILE	TOファイル	単一値: *NONE その他の値: 修飾オブジェクト名	オプション, 定位置 3
	修飾子 1: TOファイル	名前, QRPGLSRC	
	修飾子 2: ライブラリー	名前, *LIBL, *CURLIB	
TOMBR	TOメンバー	名前, *FROMMBR	オプション, 定位置 4
EXPCPY	コピー・メンバーの拡張	*NO, *YES	オプション
CVTRPT	変更報告書の印刷	*YES, *NO	オプション
SECLVL	第2レベルのテキストの組込	*NO, *YES	オプション
INSRTPL	仕様テンプレートの挿入	*NO, *YES	オプション
LOGFILE	ログ・ファイル	単一値: *NONE その他の値: 修飾オブジェクト名	オプション
	修飾子 1: ログ・ファイル	名前, QRNCVTLG	
	修飾子 2: ライブラリー	名前, *LIBL, *CURLIB	
LOGMBR	ログ・ファイル・メンバー	名前, *FIRST, *LAST	オプション

上

## FROMファイル (FROMFILE)

変換されるRPG IIIまたはRPG/400ソース・コードが入っているソース・ファイルおよびこのソース・ファイルが入っているライブラリーの名前を指定します。これは必須パラメーターです。省略時のファイル名はありません。

### ソース・ファイル名

変換されるソース・メンバーが入っているソース・ファイルの名前を入力してください。

**\*LIBL** ソース・ファイルが保管されているライブラリーを見つけるために、システムはライブラリー・リストを検索します。

**\*CURLIB**

ソース・ファイルを見つけるために、現行ライブラリーが使用されます。現行ライブラリーが指定されていない場合には、ライブラリーQGPLが使用されます。

**ライブラリー名**

ソース・ファイルが入っているライブラリーの名前を入力してください。

上

---

## FROMメンバー (FROMMBR)

変換されるソース・メンバーの名前を指定します。これは必須パラメーターです。省略時のメンバー名はありません。

変換されるソース・メンバーの有効なソース・メンバー・タイプはRPG, RPT, RPG38, RPT38, SQLRPG, およびブランクです。RPGソース変換コマンドはソース・メンバー・タイプがRPG36, RPT36, およびRPG以外のソース・メンバー・タイプ（たとえば, CLPやTXT）をサポートしません。

**ソース・ファイル・メンバー名**

変換されるソース・メンバーの名前を入力してください。

**\*ALL** このコマンドは、指定されたソース・ファイル中のすべてのメンバーを変換します。

**総称\*ソース・ファイル・メンバー名**

同じ接頭部で後に'\*'（アスタリスク）が続くメンバー名の総称名を入力してください。コマンドは、指定されたソース・ファイル中でこの総称名をもつすべてのメンバーを変換します。たとえば、FROMMBR(PR\*)を指定すると、名前が'PR'で始まるすべてのメンバーが変換されます。

上

---

## TOファイル (TOFILE)

変換されたソース・メンバーが入るソース・ファイルおよび変換済みファイルが保管されているライブラリーの名前を指定します。変換済みソース・ファイルは存在していなければならず、112文字のレコード長でなければなりません。12文字は順序番号および日付用で、80文字はコード用、20文字は注記用です。

**QRPGLESRC**

変換されたソース・メンバーが省略時のソース・ファイルQRPGLESRCに入ります。

**\*NONE**

変換済みソース・メンバーは生成されません。TOMBRパラメーターの値は無視されます。

CVTRPT(\*YES)も指定しなければなりません。さもないと変換がただちに終了します。

この機能によって、変換済みソース・メンバーを作成しなくても潜在的な問題を見つけることができます。

**ソース・ファイル名**

変換されたソース・メンバーが入る変換済みソース・ファイルの名前を入力してください。

TOFILEライブラリー名がFROMFILEライブラリー名と同じである場合には、TOFILEソース・ファイル名がFROMFILEソース・ファイル名と異ならないければなりません。

**\*LIBL** 変換済みソース・ファイルが保管されているライブラリーを見つけるために、システムはライブラリー・リストを検索します。

**\*CURLIB**

変換済みソース・ファイルを見つけるために、現行ライブラリーが使用されます。現行ライブラリーが指定されていない場合には、ライブラリーQGPLが使用されます。

**ライブラリー名**

変換済みソース・ファイルが保管されているライブラリーの名前を入力してください。

上

---

## TOメンバー (TOMBR)

変換済みソース・ファイル中の変換済みソース・メンバーの名前を指定します。FROMMBRパラメーターに指定された値が(\*ALL)または総称\*である場合には、TOMBR はFROMMBRと等しくなければなりません。

**\*FROMMBR**

FROMMBRパラメーターに指定されたメンバー名が変換済みソース・メンバー名として使用されます。FROMMBR(\*ALL)が指定された場合には、FROMFILE中のすべてのソース・メンバーが変換されます。変換済みソース・メンバーは元のソース・メンバーと同じ名前をもちます。FROMMBRパラメーターに総称名が指定された場合には、その名前に指定された接頭部をもつすべてのソース・メンバーが変換されます。変換済みソース・メンバーは元の総称ソース・メンバーと同じ名前をもちます。

**ソース・ファイル・メンバー名**

変換済みソース・メンバーの名前を入力してください。このメンバーが存在していない場合には、作成されます。

上

---

## コピー・メンバーの拡張 (EXPCPY)

/COPYメンバーが変換済みソース・メンバーに拡張されるかどうかを指定します。/COPYメンバーに関連する変換上の問題がある場合にのみEXPCPY(\*YES)を指定してください。

**\*NO** /COPYファイル・メンバーは変換済みソースに拡張されません。これが省略時の値です。

**\*YES** /COPYファイル・メンバーは変換済みソースに拡張されます。

注: メンバーのタイプがRPTまたはRPT38の場合には、報告書簡易作成プログラムは常に/COPYメンバーを展開するので、EXPCPY(\*YES)またはEXPCPY(\*NO)は影響をもちません。

上

---

## 変更報告書の印刷 (CVTRPT)

変換報告書を印刷するかしないかを指定します。

**\*YES** 変換報告書が印刷されます。これが省略時の値です。

**\*NO** 変換報告書が印刷されません。

---

## 第 2 レベルのテキストの組込 (SECLVL)

変換報告書に第 2 レベルのテキストを印刷するかしないかを指定します。

**\*NO** 変換報告書に第 2 レベル・メッセージ・テキストは印刷されません。これが省略時の値です。

**\*YES** 変換報告書に第 2 レベル・メッセージ・テキストが印刷されます。

---

## 仕様テンプレートの挿入 (INSRTPL)

ILE RPG仕様テンプレート(H-, F-, D-, I-, C-, O-仕様テンプレート)を変換済みソース・メンバーに挿入するかどうかを指定します。

**\*NO** 仕様テンプレートが変換済みソース・メンバーに挿入されません。これが省略時の値です。

**\*YES** 仕様テンプレートが変換済みソース・メンバーに挿入されます。各仕様テンプレートは、適切な仕様セクションの先頭に挿入されます。

---

## ログ・ファイル (LOGFILE)

変換情報を追跡するために使用されるログ・ファイルの名前を指定します。**\*NONE** を指定しない限り、ログ・ファイルが必要です。ファイルがすでに存在していて、物理データ・ファイルでなければなりません。ライブラリーQRPGLE中の「元オブジェクト」ファイルQRNCVTLGおよびユーザー・ライブラリー中の「新しいオブジェクト」ファイルQRNCVTLGを指定したCPYFコマンドを使用してログ・ファイルを作成してください。

### QRNCVTLG

変換情報を入れるために省略時のログ・ファイルQRNCVTLGが使用されます。

### **\*NONE**

変換情報はログ・ファイルに書き込まれません。

### ログ・ファイル名

変換情報を追跡するために使用されるログ・ファイルの名前を入力してください。

**\*LIBL** ログ・ファイルが保管されているライブラリーを見つけるために、システムはライブラリー・リストを検索します。

### ライブラリー名

ログ・ファイルが保管されているライブラリーの名前を入力してください。

---

## ログ・ファイル・メンバー (LOGMBR)

変換情報を追跡するために使用されるログ・ファイル・メンバーの名前を指定します。新しい情報は、指定されたログ・ファイル・メンバーの既存のデータに追加されます。



ログ・ファイルにメンバーが入っていない場合には、ログ・ファイルと同じ名前をもつメンバーが作成されます。

#### **\*FIRST**

コマンドは、指定されたログ・ファイル中の最初のメンバーを使用します。これが省略時の値です。

#### **\*LAST**

コマンドは、指定されたログ・ファイル中の最後のメンバーを使用します。

#### **ログ・ファイル・メンバー名**

変換情報を追跡するために使用されるファイル・メンバーの名前を入力してください。

上

---

## **例**

### **例1: RPG IIIソースのRPG IVソースへの変換**

```
CVTRPGSRC  FROMFILE(MYLIB/QRPGSRC) FROMMBR(XMPLE1)
            TOFILE(MYLIB/QRPGLESRC) TOMBR(*FROMMBR)
            EXPCPY(*NO)
            CVTRPT(*YES) LOGFILE(MYLIB/QRNCVTLG)
```

このコマンドはファイルMYLIB/QRPGSRC中のRPG IIIソース・メンバーXMPLE1を、ファイルMYLIB/QRPGLESRC中の同じ名前のRPG IVソース・メンバーに変換します。RPG IIIプログラムの/COPYステートメントは展開されません。これらは、RPG IVプログラムで/COPYステートメントとして残ります。報告書が印刷されます。それぞれの変換の状況は、ファイルMYLIB/QRNCVTLGに配置されます。

上

---

## **エラー・メッセージ**

### **\*ESCAPE メッセージ**

#### **RNS9350**

変換が終了した。

上



---

## COBOLデバッグ終了 (ENDCBLDBG)

実行可能場所: すべての環境 (\*ALL)  
スレッド・セーフ: いいえ

パラメーター  
例  
エラー・メッセージ

このコマンドは、COBOLプログラムでWITH DEBUGGING MODE文節が使用されている場合に作成されるデバッグ・コードを非活動化します。このコマンドは、デバッグを停止する各COBOLプログラムに対して入力する必要があります。

上

---

### パラメーター

キーワード	記述	選択項目	ノーツ
PGM	プログラム	修飾オブジェクト名	必須, 定位置 1
	修飾子 1: プログラム	名前	
	修飾子 2: ライブラリー	名前, *LIBL, *CURLIB	

上

---

### プログラム (PGM)

コンパイル済みCOBOLプログラムの名前およびそれが入っているライブラリーを指定します。これは必須パラメーターです。考えられる値は次の通りです。

#### プログラム名

コンパイル済みCOBOLプログラムが認識される名前を入力してください。

使用できるライブラリーの値は次の通りです。

**\*LIBL** システムは、プログラムが入っているライブラリーを見つけるために、ライブラリー・リストを検索します。

#### \*CURLIB

現行ライブラリーが使用されます。現行ライブラリーとしてライブラリーが割り当てられていない場合には、QGPLが使用されます。

#### ライブラリー名

作成されたプログラムが入っているライブラリーの名前を入力してください。

上

---

### 例

例1: COBOLプログラムのデバッグ・コードを非活動化します

```
ENDCBLDBG PGM(MYLIB/XMPLE1)
```

このコマンドは、ライブラリーMYLIBに作成されたCOBOLプログラムXMPLE1のデバッグ・コードを非活性化します。

上

---

## エラー・メッセージ

### \*ESCAPE メッセージ

#### **CBE7018**

プログラム&1が見つからない。

#### **CBE7019**

ライブラリー&1が見つからない。

#### **LBE7018**

プログラム&1が見つからない。

#### **LBE7019**

ライブラリー&1が見つからない。

上

---

## ISDBの終了 (ENDISDB)

実行可能場所: 対話環境 (\*INTERACT \*IPGM \*IMOD \*IREXX  
\*EXEC)

スレッド・セーフ: いいえ

パラメーター

例

エラー・メッセージ

---

### 対話型ソース・デバッガーの終了

サービスするジョブの対話型ソース・デバッガー・セッションを終了するためには、このコマンドを使用してください。このコマンドにはパラメーターはありません。

このコマンドを出すと、デバッグしていたすべてのプログラムがデバッグ・モードから除去されます。

デバッグ・セッションで使用したコマンドの記録は、QTEMP/QIXALOGという名前のファイルに保管されます。

ユーザーが新しいセッションを開始するたびに、ISDBはこのログ・ファイルに上書きします。コマンドを再度使用するためにログ・ファイルを保持する場合は、ファイルの名前を変更してください。

---

### エラー・メッセージ: ENDISDB

なし

上

---

### パラメーター

なし

上

---

### 例

なし

上

---

### エラー・メッセージ

なし

上



## PDMを使用したストリングの検索 (FNDSTRPDM)

実行可能場所: すべての環境 (\*ALL)

スレッド・セーフ: いいえ

パラメーター

例

エラー・メッセージ

PDMを使用したストリングの検索 (FNDSTRPDM)コマンドによって、ソース物理ファイル・メンバーまたはデータ物理ファイル・メンバー中の文字ストリングまたは16進数ストリングを検索することができます。ストリングに一致がある各メンバーについて、いずれかのプログラム開発管理機能(PDM)オプションまたは独自のユーザー定義オプションの1つを使用することができます。

上

### パラメーター

キーワード	記述	選択項目	ノーツ
STRING	'ストリング'の検索	文字値	必須, 定位置 1
FILE	ファイル	修飾オブジェクト名	必須, 定位置 2
	修飾子 1: ファイル	名前	
	修飾子 2: ライブラリー	名前, *LIBL, *CURLIB	
MBR	メンバー	値 (最大 300 回の繰り返し): 文字値, *ALL	必須, 定位置 3
OPTION	実行する操作	要素リスト	必須, 定位置 4
	要素 1: オプション	文字値, *EDIT, *COPY, *DLT, *DSP, *PRT, *RNM, *DSPD, *SAVE, *CHGT, *CMPR, *CMPL, *MOD, *MRG, *RUNP, *SDA, *DFU, *RLU, *NONE	
	要素 2: プロンプト	*NOPROMPT, *PROMPT	
COL	検索する欄	要素リスト	オプション
	要素 1: 開始桁	文字値, 1, *RCDLEN	
	要素 2: 終了桁	文字値, *RCDLEN	
CASE	一致の種類	*IGNORE, *MATCH	オプション
PRTMBRLIST	リストの印刷	*NO, *YES	オプション
PRTRCDS	レコードの印刷	単一値: *NONE その他の値: 要素リスト	オプション
	要素 1: 検索する数	文字値, '____', *ALL	
	要素 2: 印刷形式	*CHAR, *HEX, *ALTHEX	
	要素 3: レコードのマーク付け	*MARK, *NOMARK	
	要素 4: レコード・オーバーフロー	*FOLD, *TRUNCATE	
PARM	パラメーター	文字値	オプション

上



---

## ‘STRING’の検索 (STRING)

検索したいSTRINGを入力します。

これは必須パラメーターです。

**文字値** 検索するSTRINGを単一引用符で囲んで指定してください。このSTRINGは文字または16進数とすることができます。

上

---

## ファイル (FILE)

検索したいメンバーが入っているファイルを指定します。検索するファイルは、ソース物理ファイルまたはデータ物理ファイルとすることができます。

これは必須パラメーターです。

### 修飾子1: ファイル

**名前** 検索したいメンバーが入っている物理ファイルの名前を指定してください。

### 修飾子2: ライブラリー

**\*LIBL** ジョブ・ライブラリー・リストにあるすべてのライブラリーから指定ファイルを検索します。

### **\*CURLIB**

ジョブの現行ライブラリーから指定ファイルを検索します。現行ライブラリーが定義されていない場合には、現行ライブラリーにQGPLが使用されます。

**名前** 指定ファイルを検索するために使用するライブラリー名を指定します。

上

---

## メンバー (MBR)

検索するメンバーを指定します。このパラメーターを使用して、指定したファイル中のすべてのメンバーまたはメンバーのサブセットを検索することができます。

これは必須パラメーターです。

このパラメーターには値を300個指定することができます。

**\*ALL** すべてのファイル・メンバーから指定STRINGを検索します。

**総称名** 検索するファイル・メンバーの総称名を指定します。

総称名は次の形式の1つとすることができます。

**ABC\*** 名前がABCで始まるすべてのメンバーを検索します。たとえば、ABC, ABCD, または ABCTESTなど。

**\*ABC** 名前がABCで終わるすべてのメンバーを検索します。たとえば、ABC, DABC, または TESTABCなど。

**\*B\*** メンバー名のどこかに文字Bがあるすべてのメンバーを検索します。たとえば、B, BALL, ABCDなど。

**A\*C** 名前がAで始まり、Cで終るすべてのメンバーを検索します。たとえば、AC, ABC, AZZZC など。

**"A"** 名前が英字Aで始まり、引用符付きであるすべてのメンバーを検索します。たとえば、"A", "AB", "AD"など。

**\*\*ALL**

名前がALLで終わるすべてのメンバーを検索します。たとえば、ALL, BALL,または TESTALLなど。\*ALLはオブジェクト属性に関係なくファイルのすべてのメンバーを検索する特殊値として定義されているので、この場合には、2個のアスタリスクが必要です。

**名前** 検索するメンバーの名前を指定します。

上

---

## 実行する操作 (OPTION)

ストリングの一致が見つかった各メンバーで実行したいプログラム開発管理機能(PDM)オプションを指定してください。パラメーターは2つの要素から構成され、1つはオプション選択用で、1つはプロンプト用です。オプションは、このタイプのファイルに有効なPDMオプション、または活動オプション・ファイルのユーザー定義オプションとすることができます。有効なオプションは、ソース物理ファイルの場合とデータ物理ファイルの場合では異なります。

これは必須パラメーターです。

### 要素1: オプション

**\*NONE**

処置がないストリングを含むメンバーに実行されます。メンバーのリストまたはストリングを含むレコードを印刷するときこの値を使用します。

### ソース物理ファイル・メンバー・リスト

**\*EDIT** SEU (原始ステートメント入力ユーティリティー)の編集プログラムを使用して1つまたは複数のメンバーを編集します。

**\*CHGT**

1つまたは複数のメンバーの属性を変更します。

**\*CMPL**

1つまたは複数のメンバーをコンパイルします。システムは、コンパイルするメンバーに基づいてオブジェクトを作成します。メンバーは、省略時の値の変更画面またはPDM省略時の値変更コマンド(CHGPDMDFT)での指定にしたがって、対話モードまたはバッチ・モードでコンパイルされます。

コンパイルすることができるメンバー・タイプ: BAS, BAS36, BAS38, C, CBL, CBLLE, CBL36, CBL38, CICSC, CICSCBL, CICSCBLLE, CICSSQLCBL,CLD, CLLE, CLP, CLP38, CMD, CMD38, CPP, DSPF, DSPF36, DSPF38, FTN, ICF, LF, LF38, MENU, PAS, PF, PF38, PLI, PLI38, PNLGRP, PRTE, PRINT38, QRY38, RMC, RPG, RPGLE, RPG36, RPG38, RPT, RPT36, RPT38, SPADCT, SQLC, SQLCPP, SQLCBL, SQLCBLLE, SQLFTN, SQLPLI, SQLRPG, SQLRPGLE,および TBL

プログラム開発管理機能が必要な作成コマンドを使用してプログラムをコンパイルする時には、作成するオブジェクト名は常にソース・メンバー名として指定されます。オプションのプロンプトに指示するか、またはコマンド入力行に正しいパラメーターを入力することによって、オブジェクト名パラメーターを別のオブジェクト名に変更することができます。プログラム開発管理機能は、オブジェクト名がすでに存在しているかどうかをチェックし、存在している場合には、メンバーのコンパイルの確認画面を表示します。この画面には、既存のオブジェクトの削除オプションがあります。

注: 省略時の値の変更画面の**オブジェクトの置き換え**プロンプト(またはCHGPDMDFTコマンド)がY(または\*YES)に設定されている場合には、この画面は表示されません。

オブジェクト名パラメーターを特殊値に変更した場合には、PDMは、オブジェクトが存在しているかどうかをチェックしません。たとえば、RPGプログラムをコンパイルし、**プログラム**プロンプトを\*CTLSPECに変更した場合には、プログラム開発管理機能はオブジェクトが存在するかどうかをチェックしません。

**\*CMPR**

1つまたは複数のメンバーを比較します。

**\*COPY**

1つまたは複数のメンバーを1つまたは複数の新しいメンバーにコピーします。また、メンバーを別のファイル、別のライブラリー、あるいはその両方にコピーすることもできます。

**\*DLT** ファイルから1つまたは複数のメンバーを削除します。

**\*DSP** SEU (原始ステートメント入力ユーティリティー)を使用して1つまたは複数のメンバーを表示します。

**\*DSPD**

1つまたは複数のメンバーに関する情報を表示します。

**\*MOD** ILEソース・タイプのモジュール・オブジェクトを作成します。

**\*MRG** ターゲット・メンバーを他のメンバーと組み合わせます。

**\*PRT** SEU (原始ステートメント入力ユーティリティー)を使用して1つまたは複数のメンバーを印刷します。

**\*RNM** 1つまたは複数のメンバーを名前変更します。

**\*RUNP**

REXX, OCL36, BASP, またはBASP38のメンバー・タイプのソース・メンバーを実行します。実行できないタイプのメンバーを実行しようとした場合には、エラー・メッセージが表示されます。OCL36プロシージャーを実行するためには、ファイル名はQS36PRCでなければなりません。省略時の値の変更画面の**バッチで実行**プロンプトまたはCHGPDMDFTコマンドの指定にしたがって、メンバーをバッチ・モードまたは対話モードで実行することができます。

**\*SAVE**

メンバーをディスクまたはテープに保管します。

**\*SDA** SDA (画面設計機能)を使用して選択したメンバーを処理します。

- メンバーのタイプがDSPFまたはDSPF38の場合には、画面を処理するためにSDA が呼び出されます。
- メンバーのタイプがMNU, MNUDDS,またはMNUCMDの場合には、メニューを処理するためにSDAが呼び出されます。

- メンバーのタイプがDSPF36またはMNU36の場合には、システム/36のSDAメインメニューが表示されます。
- メンバーの以前のタイプMNUを入力すると、SDAはこれをMNUDDSに変換します。
- PDMのメニュー・メンバーのタイプはイメージ・メンバーの場合にはタイプMNUDDSで、コマンド・ソース・メンバーの場合にはタイプMNUCMDであることに注意してください。この2つは一緒に連係されてグループを構成しているので、一方のタイプを指定することは同時に連係されたメンバーでも操作することを意味しています。

**\*RLU** RLU (報告書設計ユーティリティー)を使用して選択したメンバーを処理します。

#### データ物理ファイル・メンバー・リスト・オプション

##### **\*CHGT**

物理ファイルの1つまたは複数のメンバーの属性を変更します。

##### **\*CMPR**

1つまたは複数のメンバーを比較します。

##### **\*COPY**

1つまたは複数のメンバーを1つまたは複数の新しいメンバーにコピーします。また、メンバーを別のファイル、別のライブラリー、あるいはその両方にコピーすることもできます。

**\*DFU** DFU (データ・ファイル・ユーティリティー)を呼び出して選択したメンバーを変更します。

**\*DLT** ファイルから1つまたは複数のメンバーを削除します。

**\*DSP** 1つまたは複数のメンバーを表示します。

##### **\*DSPD**

1つまたは複数のメンバーに関する情報を表示します。

**\*RNM** 1つまたは複数のメンバーを名前変更します。

##### **\*SAVE**

メンバーをディスクまたはテープに保管します。

#### ユーザー定義メンバー・オプション

**文字値** 活動オプション・ファイルに定義しているオプションの名前を指定してください。

#### 要素2: プロンプト

このパラメーターのプロンプト部分は、オプションのコマンドが実行されるごとにプロンプトを表示するかどうかを指定します。

##### **\*NOPROMPT**

ストリングと一致した各メンバーごとに実行するコマンドのプロンプトを表示しません。

##### **\*PROMPT**

ストリングと一致した各メンバーごとに実行するコマンドのプロンプトを表示します。

上

---

## 検索する欄 (COL)

検索する各ファイル・レコードの部分の開始および終了列番号を指定します。これにより各レコードの開始、終了または中間を検索することができます。

開始列番号はレコード長より長くてもなりません。

### 要素1: 開始桁

1 各レコードの1列目から検索を開始します。

#### \*RCDLEN

各レコードの最終列のみを検索します。

番号 検索する各レコードの部分の最初の列を指定します。

### 要素2: 終了桁

#### \*RCDLEN

各レコードの検索対象はレコードの指定された開始列から最終列までです。

番号 検索する各レコードの部分の最終の列を指定します。

上

---

## 一致の種類 (CASE)

検索で大文字小文字を区別するかどうかを指定します。

#### \*IGNORE

大文字小文字の区別なしに指定ストリングでメンバーを検索します。

#### \*MATCH

指定したストリングとの正確な一致を検索します。

上

---

## リストの印刷 (PRTMBRLIST)

一致したメンバーのリストを印刷するかどうかを指定します。

\*NO ストリングとの一致があったメンバーのリストを印刷しません。

\*YES ストリングとの一致があったメンバーのリストを印刷します。

上

---

## レコードの印刷 (PRTRCDS)

ストリングを含むレコードのうちどのレコードを印刷するかを指定します。

### 単一値

#### \*NONE

指定したストリングを含むどのレコードも印刷しません。

### 要素1: 検索する数

**\*ALL** 指定したストリングを含むすべてのレコードを印刷します。

#### **1から99999**

指定したストリングを含むレコードのうち印刷するレコードの数を指定します。

### 要素2: 印刷形式

#### **\*CHAR**

文字形式でレコードを印刷します。

**\*HEX** 16進数の上／下形式でレコードを印刷します。これは、文字値の下に16進数が印刷されることを意味します。

#### **\*ALTHEX**

16進数の横並び形式でレコードを印刷します。

### 要素3: レコードのマーク付け

印刷レコードでストリングにマークを付けることができます。ストリング自身は、高速認識のための文字検索用マーカーとして使用されます。16進数検索の場合には、ストリングにアスタリスク(\*)でマークが付けられます。

#### **\*MARK**

レコード中のストリングをマークします。

#### **\*NOMARK**

レコード中のストリングをマークしません。

### 要素4: レコード・オーバーフロー

レコードが、印刷行の長さより大きい場合に、折り返すかまたは切り捨てるかを指定することができます。

#### **\*FOLD**

複数の印刷行にわたってレコード全体を印刷します。

#### **\*TRUNCATE**

単一の印刷行に収まるレコードの部分だけを印刷します。**\*ALTHEX**を使用した場合には1から32桁だけが印刷され、**\*CHAR**または**\*HEX**を使用した場合には1から100桁が印刷されます。

上

---

## パラメーター (PARM)

**実行する操作 (OPTION)**パラメーターの要素1として指定された値の結果として実行されるコマンドに付加したいパラメーターを指定してください。

**文字値** OPTIONパラメーターに指定された値に関連したコマンドに受け渡すパラメーターを指定します。  
このパラメーターの省略時の値はブランクで、パラメーターを受け渡しません。

上

---

## 例

**例1: 1つのメンバー中のストリングを検索**

```
FNDSTRPDM  STRING('Ms') FILE(*LIBL/CUST)
           MBR('NEW') CASE(*MATCH)
           OPTION(*NONE) PRTRCDS(*ALL *CHAR)
```

このコマンドはファイルCUSTのメンバーNEWからストリング**Ms**を含むすべてのレコードを検索します。このファイルはジョブ・リストを使用して探し出します。検索は大文字小文字を区別します。検索ストリングを含む各レコードをリストしたスプール・ファイルが作成されます。

## 例2:メンバーのセットからストリングを検索

```
FNDSTRPDM  STRING('*TEST*') FILE(MYLIB/MYFILE)
           MBR('*APP') CASE(*IGNORE)
           OPTION(FIX)
```

このコマンドを使用して、MYLIBライブラリーのMYFILEファイルのうちメンバー名が英文字**APP**で終わるメンバーからストリング、**\*TEST\***を検索します。大文字小文字は区別されないため、**\*test\***（すべて小文字のTEST）または**\*Test\***（一部小文字のTEST）を含むレコードは検索ストリングの一致と見なされます。検索ストリングを含む各メンバーに対してユーザー定義オプション、FIXをアクティブ・オプション・ファイルから実行します。

上

---

## エラー・メッセージ

### \*ESCAPE メッセージ

#### PDM0055

&1コマンドの処理中にエラーが起こった。

上

## C/C++ソースの生成 (GENCSRC)

実行可能場所: すべての環境 (\*ALL)

スレッド・セーフ: いいえ

パラメーター

例

エラー・メッセージ

C/C++ソース生成(GENCSRC)コマンドは、外部記述ファイル情報を、ILE CまたはILE C++プログラムへの組み込みが可能な同等の構造にマップします。指定されたファイル・オブジェクトから生成された構造は、ソース・ファイル・メンバーかストリーム・ファイルのどちらかに書き込まれます。

上

### パラメーター

キーワード	記述	選択項目	ノート
OBJ	FROMオブジェクト	パス名	オプション, 定位置 1
SRCFILE	TOソース・ファイル	修飾オブジェクト名	オプション
	修飾子 1: TOソース・ファイル	名前	
	修飾子 2: ライブラリー	名前, *CURLIB	
SRCMBR	TOソース・メンバー	名前, *OBJ	オプション
SRCSTMF	TOソース・ストリームFILE	パス名	オプション
RCDFMT	レコード様式	単一値: *ALL その他の値 (最大 20 回の繰り返し): 名前	オプション
SLTFLD	フィールドの選択	値 (最大 6 回の繰り返し): *BOTH, *OUTPUT, *INPUT, *KEY, *INDICATOR, *LVLCHK, *NULLFLDS	オプション
PKDDECFLD	パック10進数フィールド	*DECIMAL, *CHAR	オプション
STRUCTURE	構造	*PACKED, *NONPACKED	オプション
ONEBYTE	1 バイト文字フィールド	*ARRAY, *CHAR	オプション
UNIONDFN	共用体定義名	文字値, *OBJ, *NONE	オプション
TYPEDEFPFX	Typedef接頭部	文字値, *OBJ, *NONE	オプション

上

### FROMオブジェクト (OBJ)

マップするオブジェクトを指定します。このオブジェクトは、QSYSファイル・システムにある\*FILEオブジェクトでなければなりません。

上



---

## TOソース・ファイル (SRCFILE)

生成された構造が組み込まれる物理ファイルを指定します。この物理ファイルが存在しなければなりません。

注: データの切り捨てを防ぐために、物理ファイルのレコード長が十分に大きいことを確認してください。

### 修飾子1: TOソース・ファイル

**name** 既存のファイルの名前を指定します。

### 修飾子2: ライブラリー

#### **\*CURLIB**

物理ファイルを見つけるためにジョブの現行ライブラリーが使用されます。ライブラリー・リストに現行ライブラリーの項目が存在しない場合には、QGPLが使用されます。

**name** ファイルが入っているライブラリーの名前を指定します。

上

---

## TOソース・メンバー (SRCMBR)

マップされた構造を入れるファイル・メンバーの名前を指定します。この名前のメンバーが存在しない場合には、それが自動的に追加されます。この名前の既存のメンバー中のレコードは、置き換えられます。

**\*OBJ FROMオブジェクト (OBJ)** パラメーターから引き出されたオブジェクト名がメンバー名として使用されます。

**name** 生成されたC/C++構造が組み込まれるファイル・メンバーの名前を指定します。

上

---

## TOソース・ストリームFILE (SRCSTMF)

生成された構造が組み込まれるストリーム・ファイルを指定します。ストリーム・ファイルのパスのすべてのディレクトリーが存在していなければなりません。この名前のファイルが存在しない場合には、それが自動的に作成されます。この名前の既存のファイル中のデータは、置き換えられます。

### **path-name**

生成されたソース・コードが組み込まれるストリーム・ファイルの完全パス名を指定します。

上

---

## レコード様式 (RCDFMT)

C/C++構造が生成されるファイルのレコード様式を指定します。

### 単一値

**\*ALL** ファイルのすべてのレコード様式で構造が生成されます。

### その他の値

**name** 構造が生成されるレコード様式を指定します。最大20個のレコード様式を指定することができます。

上

---

## フィールドの選択 (SLTFLD)

マップされるフィールドの使用タイプを指定します。

このパラメーターには6個の値を指定できます。

### **\*BOTH**

DDSでINPUT, OUTPUT,またはBOTHとして宣言されたフィールドがTYPEDEF構造に組み込まれます。装置ファイルのための外部ファイル記述にキーワードINDARAが指定されていないかぎり、オプションおよび応答標識が両方の構造に組み込まれます。

### **\*INPUT**

INPUTかBOTHのいずれかとして宣言されたフィールドがマップされた構造に組み込まれます。装置ファイルのための外部ファイル記述にキーワードINDARAが指定されていないかぎり、応答標識が入力構造に組み込まれます。

### **\*OUTPUT**

OUTPUTかBOTHのいずれかとして宣言されたフィールドがレコード構造に組み込まれます。装置ファイルのための外部ファイル記述にキーワードINDARAが指定されていないかぎり、オプション標識が出力構造に組み込まれます。

**\*KEY** 外部ファイル記述でキーとして宣言されたフィールドが組み込まれます。このオプションは、データベース・ファイルおよびDDMファイルにのみ有効です。

### **\*INDICATOR**

標識オプションが指定された場合に、標識のために別個の99バイトの構造が作成されます。このオプションは、装置ファイルにのみ有効です。

### **\*LVLCHK**

\_LVLCHK\_Tという名前の、STRUCTの配列のTYPEDEFが生成されます。タイプ\_LVLCHK\_Tのオブジェクトに対するポインターも生成され、レベル検査情報（形式名およびレベル識別コード）によって初期化されます。

### **\*NULLFLDS**

DDSのレコード様式の中に少なくとも1つのヌル値可能フィールドがある場合には、その様式内のすべてのフィールドについて文字フィールドを含むヌル・マップTYPEDEFが生成されます。このTYPEDEFによって、ヌルにするフィールドを指定（各ヌル・フィールドの値は'1'に設定し、それ以外は'0'に設定する）ことができます。また、\*KEYオプションが\*NULLFLDSオプションと一緒に使用され、様式の中に少なくとも1つのヌル値可能フィールドがある場合には、その様式内のすべてのキー・フィールドに付いて文字フィールドを含む追加のTYPEDEFが生成されます。

物理および論理ファイルの場合には、\*INPUT, \*BOTH, \*KEY, \*LVLCHK,および\*NULLFLDSを指定することができます。装置ファイルの場合には、\*INPUT, \*OUTPUT, \*BOTH, \*INDICATORS,および\*LVLCHKを指定することができます。

上

---

## パック10進数フィールド (PKDDECFLD)

パック10進数フィールドのマッピングを指定します。

### **\*DECIMAL**

パック10進数フィールドは\_DECIMALデータ・タイプとして宣言されます。

### **\*CHAR**

パック10進数フィールドは文字配列として宣言されます。

上

---

## 構造 (STRUCTURE)

パック構造が生成されるかどうかを指定します。

### **\*PACKED**

パック構造が生成されます。

### **\*NONPACKED**

パック構造は生成されません。

上

---

## 1 バイト文字フィールド (ONEBYTE)

1 バイト・フィールド用に配列と単一文字のどちらが生成されるかを指定します。

### **\*ARRAY**

文字の1要素配列が1バイト文字用に生成されます。

### **\*CHAR**

1バイト文字フィールドが1バイト文字用に生成されます。

上

---

## 共用体定義名 (UNIONDFN)

生成される共用体名を指定します。

**\*OBJ** OBJパラメーターから引き出されたファイル名を使用します。

### **\*NONE**

共用体は生成されません。

### **共用体名**

共用体名\_Tという名前を持つ共用体定義を生成します。最大長は50文字です。

上

---

## Typedef接頭部 (TYPEDEFPRFX)

生成された構造の接頭部を指定します。

**\*OBJ** OBJパラメーターから引き出されたファイル名を使用します。

**\*NONE**

生成された構造に接頭部は使用しません。

### 接頭部名

構造名の接頭部を指定します。最大長は50文字です。

上

---

## 例

### 例1:すべてのレコード様式のマッピングの作成

```
GENCSRC  OBJ('/QSYS.LIB/QTEMP.LIB/MYTBL.FILE')
          SRCFILE(QTEMP/H)
```

このコマンドは、ライブラリーQTEMPにあるファイル・オブジェクトMYTBLのすべてのレコード様式のマッピングを作成します。このマッピングは、ライブラリーQTEMPにあるソース・ファイルHのメンバーMYTBL内に保管されます。

生成されたマッピングは、以下のincludeステートメントを使用してC/C++プログラムに組み込むことができます。

```
#include "QTEMP/H(MYTBL)"
```

### 例2: 1つのレコード様式のマッピングの作成

```
GENCSRC  OBJ('/QSYS.LIB/QTEMP.LIB/MYTBL.FILE')
          SRCSTMF('/tmp/myHdr.h')  RCDfmt(PRSNL)
          SLTFLD(*INPUT)  PKDDECFLD(*CHAR)
          STRUCTURE(*NONPACKED)  TYPEDEFPIX(TAB1)
```

このコマンドは、ライブラリーQTEMPにあるファイル・オブジェクトMYTBLのレコード様式PRSNLのマッピングを作成します。このレコード様式の入力フィールドのみがマップされます。生成された構造はパックされません。この構造では、バック10進データの代わりにchar配列がレコード内に含められ、構造名には接頭部としてTAB1が付けられます。マッピングは、ストリーム・ファイル/tmp/myTbl.h内に保管されます。

生成されたマッピングは、以下のincludeステートメントを使用してC/C++プログラムに組み込むことができます。

```
#include "/tmp/myTbl.h"
```

上

---

## エラー・メッセージ

**\*ESCAPE** メッセージ

**CZM2613**

GENCSRCが組み込みファイルの生成に失敗しました。

上



---

## 用紙記述組み合わせ (MRGFORMD)

実行可能場所: すべての環境 (\*ALL)  
スレッド・セーフ: いいえ

パラメーター  
例  
エラー・メッセージ

用紙記述組み合わせ(MRGFORMD)コマンドは、スプール出力ファイルを拡張印刷機能(STRAPF)コマンドで設計された用紙記述の入っているデータベース・ファイルと組み合わせます。出力は後から印刷するためにスプールするか、あるいはただちに印刷装置に向けることができます。

用紙記述組み合わせ(MRGFORMD)コマンドは、IBMアプリケーション開発ツール・プログラム(5761-WDS)の一部です。拡張印刷機能(APF)ツールの詳細については、ADVANCED PRINTER FUNCTION GUIDE(SC09-1361)を参照してください。

注: この項目が、(画面自身またはヘルプ情報に)アスタリスク付きで示される「特殊値」でない限り、アスタリスクを項目の前に置かないでください。

---

## エラー・メッセージ: MRGFORMD

### \*ESCAPE メッセージ

#### APF5101

印刷装置ファイル&1をオープンすることができない。

#### APF5102

スプール・ファイルにアクセスしようとした時にエラーが起こった。

#### APF5104

用紙記述が使用可能な状態にない。

#### APF5105

用紙記述が正しくない。

#### APF5106

要求された機能の実行は許可されていない。

#### APF5107

装置&3にエラーが起こった。

#### APF5121

APFユーティリティーの場合、&2のファイル&1は正しくない。

#### APF9901

APFユーティリティーにエラーが起こった。

#### APF9910

表示装置ファイル入出力操作を完了することができない。

#### APF9911

データベース・ファイル入出力操作を完了することができない。

#### APF9912

拡張印刷機能ファイルをオープンすることができない。

## パラメーター

キーワード	記述	選択項目	ノーツ
<b>FORMD</b>	用紙記述	名前	必須, 定位置 1
<b>FILE</b>	ファイル	修飾オブジェクト名	必須, 定位置 2
	修飾子 1: ファイル	名前	
	修飾子 2: ライブラリー	名前, <b>*LIBL</b> , <b>*CURLIB</b>	
<b>SPLF</b>	スプール・ファイル	名前, <b>*NONE</b>	オプション
<b>JOB</b>	ジョブ名	単一値: * その他の値: 修飾ジョブ名	オプション
	修飾子 1: ジョブ名	名前	
	修飾子 2: ユーザー	名前	
	修飾子 3: 番号	000000-999999	
<b>SPLNBR</b>	スプール・ファイル番号	1-999999, <b>*ONLY</b> , <b>*LAST</b>	オプション
<b>COPIES</b>	コピー枚数	1-255, <b>*FILE</b>	オプション
<b>DEV</b>	装置	名前, <b>*FILE</b> , <b>*JOB</b> , <b>*SYSVAL</b>	オプション
<b>SPOOL</b>	データのスプール	<b>*YES</b> , <b>*NO</b> , <b>*FILE</b>	オプション
<b>OUTQ</b>	出力待ち行列	単一値: <b>*FILE</b> , <b>*JOB</b> , <b>*DEV</b> その他の値: 修飾オブジェクト名	オプション
	修飾子 1: 出力待ち行列	名前	
	修飾子 2: ライブラリー	名前, <b>*LIBL</b> , <b>*CURLIB</b>	
<b>FORMTYPE</b>	用紙タイプ	文字値, <b>*FILE</b> , <b>*STD</b>	オプション
<b>OUTSPLF</b>	出力スプール・ファイル	名前, <b>*FRMD</b> , ' , '	オプション
<b>SCHEDULE</b>	スプール出力のスケジュール	<b>*FILE</b> , <b>*IMMED</b> , <b>*JOBEND</b> , <b>*FILEEND</b>	オプション
<b>JOB</b>	ジョブ記述	単一値: <b>*NONE</b> その他の値: 修飾オブジェクト名	オプション
	修飾子 1: ジョブ記述	名前, <b>*QBATCH</b>	
	修飾子 2: ライブラリー	名前, <b>*LIBL</b> , <b>*CURLIB</b>	

## 用紙記述 (FORMD)

用紙の印刷または組み合わせ操作で使用する用紙記述の名前を指定します。

これは必須パラメーターです。

---

## ファイル (FILE)

用紙記述の入っているファイルの名前およびライブラリーを指定します。

これは必須パラメーターです。

使用できるライブラリーの値は次の通りです。

**\*LIBL** ライブラリー・リストが、ファイルを検索する場合に使用されます。

**\*CURLIB**

ジョブの現行ライブラリーが、ファイルを検索する場合に使用されます。ライブラリー・リストに現行ライブラリー項目が存在しない場合には、QGPLが使用されます。

**ライブラリー名**

ファイルが入っているライブラリーを指定してください。

上

---

## スプール・ファイル (SPLF)

用紙記述と組み合わせるデータの入っているスプール出力ファイルの名前を指定します。

指定できる値は次の通りです。

**\*NONE**

スプールされた出力ファイルは指定されません。

**スプール・ファイル名**

スプール出力ファイルの名前を指定してください。

上

---

## ジョブ名 (JOB)

用紙記述と組み合わせるスプール出力ファイルの入っているジョブの名前を指定します。

指定できる値は次の通りです。

**\* \_** スプール・ファイルが入っている現行ジョブを指定します。

**ジョブ名**

組み合わせるスプール・ファイルを作成したジョブの名前を指定してください。ジョブ名を指定しない場合には、単純ジョブ名システム内に現在あるすべてのジョブの中から検索されます。

**ユーザー**

ユーザー名は、ジョブがどのユーザー・プロファイルのもとで実行されるかを示します。

**番号** システム割り当てジョブ番号。

上



---

## スプール・ファイル番号 (SPLNBR)

用紙記述と組み合わせるスプール出力ファイルの番号を指定します。

指定できる値は次の通りです。

### **\*LAST**

指定した名前の最後のスプール・ファイルが、用紙記述と組み合わせられるということを指定します。

### **\*ONLY**

1つだけのスプール・ファイルが、スプール・ファイルプロンプト (SPLF パラメーター)で指定した名前をもつということを指定します。

### **スプール・ファイル名**

用紙記述と組み合わせるスプール出力ファイルの番号を入力してください。

上

---

## コピー枚数 (COPIES)

印刷される組み合わせスプール出力ファイルのコピー数を指定します。

指定できる値は次の通りです。

**\*FILE** 印刷するコピーの数は、印刷装置ファイル(QPAPFPRT)に指定したCOPIESの値から選択されます。

### **コピー数**

印刷される組み合わせスプール出力ファイルのコピー数を入力してください。

上

---

## 装置 (DEV)

用紙または組み合わせ出力の印刷に使用される印刷装置の名前を指定します。

指定できる値は次の通りです。

**\*FILE** 印刷装置として用いられる装置は、印刷装置ファイル(QPAPFPRT)のDEVパラメーターで指定されたものと同じものになります。

### **\*SYSVAL**

印刷装置として用いられる装置は、システム値QPRTDEVにより指定されます。

**\*JOB** 印刷装置として用いられる装置は、ジョブの装置ファイルにより指定されます。

**装置名** 用紙または組み合わせ出力の印刷に使用される印刷装置の名前を指定してください。

上

---

## データのスプール (SPOOL)

データをスプールするか否かを指定します。

指定できる値は次の通りです。

**\*FILE** スプール・ファイル属性は、印刷装置ファイル(QPAPFPRT)で使用されたものと同じものになります。

**\*YES** データがスプールされます。

**\*NO** データはスプールされません。

上

---

## 出力待ち行列 (OUTQ)

組み合わせスプール出力ファイルが置かれる出力待ち行列を指定します。

指定できる値は次の通りです。

**\*FILE** 出力待ち行列名は、印刷装置ファイル(QPAPFPRT)で指定されたものと同じものになります。

**\*DEV** 装置プロンプト (DEVパラメーター) で指定された印刷装置に対応した出力待ち行列の省略時の値を使用してください。

**\*JOB** スプール出力のジョブと結び付けられたジョブ記述に指定されている出力待ち行列を使用してください。

### 出力待ち行列名

スプールされたデータベース出力ファイルを入れる出力待ち行列の名前およびライブラリーを入力してください。指定できるライブラリー名の値は次の通りです。

**\*LIBL** ライブラリー・リストが、出力待ち行列の検索に使用されます。

### \*CURLIB

ジョブ用の現行ライブラリーが、出力待ち行列の検索に使用されます。ライブラリー・リストに現行ライブラリー項目が存在しない場合には、QGPLが使用されます。

### ライブラリー名

出力待ち行列が置かれるライブラリーを指定してください。

上

---

## 用紙タイプ (FORMTYPE)

組み合わせスプール出力ファイルが印刷される用紙のタイプを指定します。

指定できる値は次の通りです。

**\*FILE** 組み合わせスプール出力ファイルは、印刷装置ファイル(QPAPFPRT)で指定された用紙タイプの用紙に印刷されます。

**\*STD** 組み合わせスプール出力ファイルは、導入システムで使用する標準用紙タイプの用紙に印刷されます。

### 用紙タイプ

スプール出力ファイルが印刷される用紙の用紙タイプの名前を入力してください。

上

---

## 出力スプール・ファイル (OUTSPLF)

出力待ち行列上の組み合わせスプール出力ファイルの名前を指定します。

指定できる値は次の通りです。

### **\*FRMD**

用紙記述名が、出力待ち行列上の組み合わせスプール出力ファイルの名前として使用されます。

### **出力スプール・ファイル名**

出力待ち行列上の組み合わせスプール出力ファイルの名前（最大 10 文字）を入力してください。

上

---

## スプール出力のスケジュール (SCHEDULE)

組み合わせスプール出力ファイルが、スプール書き出しプログラムにとって使用可能となる時点を指定します。

指定できる値は次の通りです。

**\*FILE** 組み合わせスプール出力ファイルは、印刷装置ファイル(QPAPFPRT)のスプール出力のスケジュールプロンプト(SCHEDULEパラメーター)で指定された通りに、スプール書き出しプログラムで使用可能となります。

### **\*IMMED**

組み合わせスプール出力ファイルは、スプール書き出しプログラムでただちに使用可能となります。

### **\*JOBEND**

組み合わせスプール出力ファイルは、現行ジョブが終了した時点で、スプール書き出しプログラムで使用可能となります。

### **\*FILEEND**

組み合わせスプール出力ファイルは、現行ファイルの終わりになった時点で、スプール書き出しプログラムで使用可能となります。

上

---

## ジョブ記述 (JOBID)

ジョブの投入に使用するジョブ記述の名前を指定します。

指定できる値は次の通りです。

### **\*NONE**

印刷は、現行ジョブ記述のもとで行なわれます。

### **ジョブ記述名**

ジョブの投入に使用するジョブ記述の名前およびライブラリーを指定してください。指定できるライブラリー名の値は次の通りです。

**\*LIBL** ライブラリー・リストが、ジョブ記述を見つけるために使用されます。

## \*CURLIB

ジョブ用の現行ライブラリーが、ジョブ記述を見つけるために使用します。ライブラリー・リストに現行ライブラリー項目が存在しない場合には、QGPLが使用されます。

## ライブラリー名

ジョブ記述が置かれるライブラリーを指定してください。

上

---

## 例

なし

上

---

## エラー・メッセージ

### \*ESCAPE メッセージ

#### APF5101

印刷装置ファイル&1をオープンすることができない。

#### APF5102

スプール・ファイルにアクセスしようとした時にエラーが起こった。

#### APF5104

用紙記述が使用可能な状態にない。

#### APF5105

用紙記述が正しくない。

#### APF5106

要求された機能の実行は許可されていない。

#### APF5107

装置&3にエラーが起こった。

#### APF5121

APFユーティリティーの場合、&2のファイル&1は正しくない。

#### APF9901

APFユーティリティーにエラーが起こった。

#### APF9910

表示装置ファイル入出力操作を完了することができない。

#### APF9911

データベース・ファイル入出力操作を完了することができない。

#### APF9912

拡張印刷機能ファイルをオープンすることができない。

上



## ソースの組み合わせ (MRGSRC)

実行可能場所: すべての環境 (\*ALL)  
スレッド・セーフ: いいえ

パラメーター  
例  
エラー・メッセージ

ソース物理ファイルのマージヘルプ

MRGSRCコマンドに次のソース・ファイルの名前を指定します。

ルート 更新の基礎となったソース・ファイルの元のバージョン

保守 宛先ファイルにマージする更新が入っているソース・ファイル

宛先 保守ファイルからの更新がマージされるソース・ファイル

MRGSRCコマンドは、それぞれの宛先メンバーおよび保守メンバーを対応するルートのメンバーと比較します。この比較の結果は、行なわれた更新の判別に使用されます。

### エラー・メッセージ: MRGSRC

なし

上

### パラメーター

キーワード	記述	選択項目	ノート
TGTFILE	ターゲット・ファイル	修飾オブジェクト名	必須, 定位置 1
	修飾子 1: ターゲット・ファイル	名前	
	修飾子 2: ライブラリー	名前, *LIBL, *CURLIB	
TGTMBR	ターゲット・メンバー	名前, *ALL	必須, 定位置 2
MAINTFILE	ファイルの保守	修飾オブジェクト名	オプション, 定位置 3
	修飾子 1: ファイルの保守	名前, *TARGET	
	修飾子 2: ライブラリー	名前, *LIBL, *CURLIB	
MAINTMBR	メンバーの保守	名前, *TARGET	オプション, 定位置 4
ROOTFILE	ルート・ファイル	修飾オブジェクト名	オプション, 定位置 5
	修飾子 1: ルート・ファイル	名前, *MAINT	
	修飾子 2: ライブラリー	名前, *LIBL, *CURLIB	
ROOTMBR	ルート・メンバー	名前, *MAINT	オプション, 定位置 6
SELECT	更新の選択	*YES, *NO	オプション
RPTONLY	報告専用	*NO, *YES	オプション

上

---

## ターゲット・ファイル (TGTFILE)

保守更新がマージされるソース物理ファイルを指定します。

**\*LIBL** ライブラリー・リストを使用します。

**\*CURLIB**

現行ライブラリーを使用します。

**ライブラリー名**

指定したライブラリーを使用します。

**ファイル名**

指定したファイルを使用します。

上

---

## ターゲット・メンバー (TGTMBR)

保守更新がマージされるメンバーを指定します。

**\*ALL** すべてのメンバーを選択します。

**メンバー名**

指定したメンバーを選択します。

上

---

## ファイルの保守 (MAINTFILE)

マージする更新が入っているソース物理ファイルを指定します。

**\*LIBL** ライブラリー・リストを使用します。

**\*CURLIB**

現行ライブラリーを使用します。

**ライブラリー名**

指定したライブラリーを使用します。

**\*TARGET**

宛先ファイルに指定したファイルを使用します。

**ファイル名**

指定したファイルを使用します。

上

---

## メンバーの保守 (MAINTMBR)

マージする更新が入っているメンバーを指定します。

**\*TARGET**

TGTMBRキーワードに指定したものと同一メンバーを選択します。

#### メンバー名

指定したメンバーを選択します。

上

---

## ルート・ファイル (ROOTFILE)

マージ処理の基礎となるソース・ファイルを指定します。

**\*LIBL** ライブラリー・リストを使用します。

#### **\*CURLIB**

現行ライブラリーを使用します。

#### ライブラリー名

指定したライブラリーを使用します。

#### **\*MAINT**

保守ファイルに指定したファイルを使用します。

#### ファイル名

指定したファイルを使用します。

上

---

## ルート・メンバー (ROOTMBR)

マージ処理の基礎となるソース・ファイル・メンバーを指定します。

#### **\*MAINT**

MAINTMBRパラメーターに指定したものと同一メンバーを選択します。TGTMBR(\*ALL)が指定された場合には、このパラメーターは必須パラメーターです。

#### メンバー名

指定したメンバーを選択します。

上

---

## 更新の選択 (SELECT)

保守更新の選択に分割マージ画面を表示するかどうかを指定します。

**\*YES** 保守更新を選択して宛先メンバーにマージすることができるように分割マージ画面を表示します。  
報告書は生成されません。

**\*NO** 分割マージ画面を表示しないで、マージ要約報告書を印刷します。

上

---

## 報告専用 (RPTONLY)

保守更新を宛先メンバーにマージするか、マージ要約報告書を印刷して更新の範囲を表示するかを指示します。



**\*NO**    マージを実行してマージ要約報告書を印刷します。

**\*YES**   マージを実行しないでマージ要約報告書を印刷します。

上

---

## 例

なし

上

---

## エラー・メッセージ

なし

上

## CODEバッチ・ジョブの投入 (SBMCODEJOB)

実行可能場所:

- バッチ・ジョブ (\*BATCH)
- 対話式ジョブ (\*INTERACT)
- バッチ・プログラム (\*BPGM)
- 対話式プログラム (\*IPGM)
- バッチ REXX プロシージャ (\*BREXX)
- 対話式 REXX プロシージャ (\*IREXX)
- QCMDEXEC, QCAEXEC, または QCAPCMD API (\*EXEC) の使用

スレッド・セーフ: いいえ

パラメーター

例

エラー・メッセージ

CLコマンドSBMCODEJOBは、バッチで実行されるコマンドを投入し、任意選択で、ジョブが完了した時点でワークステーションに通知し、特定のコンパイル・コマンドからのエラー・フィードバック情報をワークステーションに戻します。

このコマンドは、"IBM適用業務開発ツールセット・クライアント/サーバー/400" (ADTS CS/400)プログラム・プロダクトの"連携開発環境プログラム/400"および"VRPGクライアント/2"機能によって使用されることが前提となっています。

上

## パラメーター

キーワード	記述	選択項目	ノート
<b>CMD</b>	実行するコマンド	コマンド・ストリング	必須, 定位置 1
<b>JOB</b>	ジョブ記述	修飾オブジェクト名	オプション, 定位置 2
	修飾子 1: ジョブ記述	名前, * <u>USRPRF</u>	
	修飾子 2: ライブラリー	名前, * <u>LIBL</u> , * <u>CURLIB</u>	
<b>JOBQ</b>	ジョブ待ち行列	修飾オブジェクト名	オプション, 定位置 3
	修飾子 1: ジョブ待ち行列	名前, * <u>JOB</u>	
	修飾子 2: ライブラリー	名前, * <u>LIBL</u> , * <u>CURLIB</u>	
<b>NOTIFY</b>	通知メッセージ	*YES, * <u>NO</u>	オプション, 定位置 4
<b>SERVER</b>	ホスト・サーバー名	文字値, <u>OS400</u>	オプション, 定位置 5
<b>ADDPARMS</b>	追加のSBMJOBパラメーター	文字値, * <u>NONE</u>	オプション, 定位置 6

上

## 実行するコマンド (CMD)

バッチで実行されるCLコマンドを指定します。

上

---

## ジョブ記述 (JOB D)

ジョブがシステムによって処理される時に、そのジョブと関連づけられる名前を指定します。

使用できる値は、次の通りです。

### **\*USRPRF**

投入済みジョブが実行されるユーザー・プロファイル中のジョブ記述が、その投入済みジョブのジョブ記述として使用されます。

### **ジョブ記述名**

ジョブに使用されるジョブ記述の名前（ライブラリー名／ジョブ記述名）を指定してください。

使用できる値は、次の通りです。

**\*LIBL** 最初的一致が見つかるまで、ジョブのライブラリー・リスト中のすべてのライブラリーが検索されます。

### **\*CURLIB**

ジョブ記述名を見つけるために、ジョブの現行ライブラリーが使用されます。ジョブの現行ライブラリーとしてライブラリーを指定しない場合には、QGPLが使用されます。

### **ライブラリー名**

ジョブ記述名が入っているライブラリーの名前を指定してください。

上

---

## ジョブ待ち行列 (JOB Q)

このジョブが入られるジョブ待ち行列の名前を指定します。

使用できる値は、次の通りです。

### **\*JOB D**

投入済みジョブは、指定されたジョブ記述に名前が指定されているジョブ待ち行列に入れられます。

### **ジョブ待ち行列名**

投入済みジョブが入られるジョブ待ち行列の名前（ライブラリー名／ジョブ待ち行列名）を指定してください。

使用できる値は、次の通りです。

**\*LIBL** 最初的一致が見つかるまで、ジョブのライブラリー・リスト中のすべてのライブラリーが検索されます。

### **\*CURLIB**

ジョブ待ち行列名を見つけるために、ジョブの現行ライブラリーが使用されます。ジョブの現行ライブラリーとしてライブラリーを指定しない場合には、QGPLが使用されます。

### **ライブラリー名**

ジョブ待ち行列名が入っているライブラリーの名前を指定してください。

上

---

## 通知メッセージ (NOTIFY)

バッチ・ジョブに、投入されたコマンドの完了をワークステーションに通知させるためには、YESを指定してください。

このオプションを使用するためには、EVFCICFFという名前のICFファイルがバッチ・ジョブのライブラリー・リスト中で見つからなければなりません。これは、ワークステーションの位置を識別します。ICFファイルのセットアップの詳細については、CODE/400の導入の手引きを参照してください。

上

---

## ホスト・サーバー名 (SERVER)

CODE開始(STRCODE)コマンドによってすでに開始されているCODE/400またはVRPG/2 サーバーの名前を指定します。

このパラメーターは、投入されているコマンドがCODE/400ワークステーション編集プログラムで使用されるエラー・フィードバック情報を生成する場合にのみ使用されます。パラメーターOPTION(\*SRCDBG), OPTION(\*LSTDBG),およびOPTION(\*EVENTF)の指定されたコマンドだけがこのようなエラー・フィードバックを生成します。

上

---

## 追加のSBMJOBパラメーター (ADDPARMS)

ジョブ投入(SBMJOB)コマンドで使用されるパラメーターを指定します。

SBMCODEJOBコマンドはSBMJOBを呼び出します。これには、CMD, JOB,およびJOBQを除き、すべてのSBMJOBパラメーターにシステムの省略時の値が使用されます。このパラメーターによって、SBMJOBコマンドに追加のパラメーターを指定して、システムの省略時の値を一時変更することができます。たとえば、バッチ・ジョブのジョブ記述のライブラリー・リストを使用するためには、このパラメーターに'INLLIBL(\*JOB)'を指定してください。

複数のSBMJOBパラメーターを指定することができますが、それぞれを1つ以上のブランクで区切る必要があります。

'INLLIBL(\*JOB)'を指定する場合には、ジョブ記述のライブラリー・リストにEVFCICFF ICFファイルをもつライブラリーが組み込まれていなければなりません。

上

---

## 例

なし

上

---

## エラー・メッセージ

不明

上

---

## 拡張印刷機能 (STRAPF)

実行可能場所: 対話環境 (\*INTERACT \*IPGM \*IREXX \*EXEC)

スレッド・セーフ: いいえ

パラメーター

例

エラー・メッセージ

拡張印刷機能(STRAPF)コマンドにより、カスタマイズ用紙の設計用のオプションを表示するメニューが呼び出されます。このメニューにより、特殊な記号、用紙のレイアウト（用紙記述と呼ばれる）の設計、用紙記述のコピー印刷、またはスプール・ファイルと用紙記述の組み合わせ、および結果の印刷を行なうことができます。

拡張印刷機能(APF)ツールには、IBM 5224および5225ワークステーション印刷装置の特殊な印刷機能が使用されます。

このコマンドにはパラメーターはありません。

---

### エラー・メッセージ: STRAPF

なし

上

---

### パラメーター

なし

上

---

### 例

なし

上

---

### エラー・メッセージ

なし

上



---

## COBOLデバッグ開始 (STRCBLDBG)

実行可能場所: すべての環境 (\*ALL)

スレッド・セーフ: いいえ

パラメーター

例

エラー・メッセージ

このコマンドは、COBOLプログラムでWITH DEBUGGING MODE文節が使用されている場合に作成するデバッグ・コードを活動化します。このコマンドは、各COBOL RUN UNITでデバッグされる各COBOLプログラムに対して入力する必要があります。

上

---

### パラメーター

キーワード	記述	選択項目	ノート
PGM	プログラム	修飾オブジェクト名	必須, 定位置 1
	修飾子 1: プログラム	名前	
	修飾子 2: ライブラリー	名前, *LIBL, *CURLIB	

上

---

### プログラム (PGM)

コンパイル済みCOBOLプログラムの名前およびそれが入っているライブラリーを指定します。これは必須パラメーターです。考えられる値は次の通りです。

#### プログラム名

コンパイル済みCOBOLプログラムが認識される名前を指定します。

使用できるライブラリーの値は次の通りです。

**\*LIBL** システムは、プログラムが入っているライブラリーを見つけるために、ライブラリー・リストを検索します。

#### \*CURLIB

現行ライブラリーが使用されます。現行ライブラリーとしてライブラリーが割り当てられていない場合には、QGPLが使用されます。

#### ライブラリー名

作成されたプログラムが記憶されているライブラリーの名前を入力してください。

上

---

### 例

#### 例1: COBOLプログラムのデバッグ・コードを活動化します

STRCBLDBG PGM(MYLIB/XMPLE1)



このコマンドは、ライブラリーMYLIBに作成されたCOBOLプログラムXMPLE1のデバッグ・コードを活動化します。

上

---

## エラー・メッセージ

### \*ESCAPE メッセージ

#### **LBE7018**

プログラム&1が見つからない。

#### **LBE7019**

ライブラリー&1が見つからない。

上

# CGU開始 (STRCGU)

実行可能場所:

- バッチ・ジョブ (\*BATCH)
- 対話式ジョブ (\*INTERACT)
- 対話式プログラム (\*IPGM)
- バッチ REXX プロシージャ (\*BREXX)
- 対話式 REXX プロシージャ (\*IREXX)
- QCMDEXEC, QCAEXEC, または QCAPCMD API (\*EXEC) の使用

スレッド・セーフ: いいえ

パラメーター  
例  
エラー・メッセージ

STRCGU (CGU開始) コマンドは、文字作成ユーティリティ(CGU)を開始します。

## エラー・メッセージ: STRCGU

なし

上

## パラメーター

キーワード	記述	選択項目	ノート
OPTION	CGUオプション	<u>*SELECT</u> , 1, 2, 3, 4, 5, 6	オプション, 定位置 1

上

## CGUオプション (OPTION)

文字作成ユーティリティ(CGU)メニューから直接利用するオプションを指定します。

指定できる値は次のとおりです。

### \*SELECT

文字作成ユーティリティ(CGU)メニューを表示します。

### メインメニュー・オプション番号

CGUメニューでオプションと対応する1-6の数を入力します。このパラメーター値を選択した場合には、CGUメニューは現れません。

上

## 例

なし

上

---

## エラー・メッセージ

なし

上

# CODEの開始 (STRCODE)

実行可能場所:

- バッチ・ジョブ (\*BATCH)
- 対話式ジョブ (\*INTERACT)
- バッチ・プログラム (\*BPGM)
- 対話式プログラム (\*IPGM)
- バッチ REXX プロシージャ (\*BREXX)
- 対話式 REXX プロシージャ (\*IREXX)
- QCMDEXEC, QCAEXEC, または QCAPCMD API (\*EXEC) の使用

スレッド・セーフ: いいえ

パラメーター  
例  
エラー・メッセージ

CLコマンドSTRCODEは、連携開発環境プロダクトのホスト・サーバーを実行するジョブを開始します。このコマンドは、OS/400 のホスト環境で呼び出さなければなりません。

上

## パラメーター

キーワード	記述	選択項目	ノート
SERVER	ホスト・サーバー名	文字値, <u>OS400</u>	オプション, 定位置 1
RMTLOCNAME	リモート・ロケーション名	文字値, *RESOLVE, *PRV	オプション, 定位置 2
CMNTYPE	通信タイプ	*PRV, *APPC, *TCPIP	オプション, 定位置 3
PARMS	追加のパラメーター	文字値, *NONE	オプション, 定位置 7
PORT	TCP/IPポート番号	1-65534, *PRV, *DFT	オプション, 定位置 4
USERID	ユーザー識別コード	文字値, *NONE	オプション, 定位置 5
PASSWORD	パスワード	文字値	オプション, 定位置 6

上

## ホスト・サーバー名 (SERVER)

システムによって処理されるジョブと関連した名前を指定します。

使用できる値は、次の通りです。

**OS400** これが省略時の値です。

**サーバー名**

システムによって処理されるジョブの識別に使用する名前を指定します。

注: 次のサーバー名は、予約されているので使用することはできません。

- LOCAL
- REMOTE

---

## リモート・ロケーション名 (RMTLOCNAME)

ホスト・サーバーが通信するリモート・ワークステーションの名前を指定します。TCP/IP 接続の場合には、これは、ユーザーのPCのIP名またはIPアドレスです。APPC接続の場合には、この名前は、リモート・ワークステーションの論理装置(LU)またはユーザーの通信プログラムのSNA基本プロファイルに指定されたPCローカル名です。

使用できる値は、次の通りです。

**\*PRV** このコマンドの最後の呼び出しで使用された値が省略時の値となります。

### **\*RESOLVE**

CODE通信にリモート・ロケーション名を解決させます。TCP/IP DHCPユーザーには、この値をお勧めします。バッチ・ジョブでは、この値は機能しません。

### **PCローカル名**

ホスト・サーバーが通信するリモート・ワークステーションを識別する名前を入力してください。

上

---

## 通信タイプ (CMNTYPE)

ホスト・サーバーが通信するプロトコルのタイプの名前を指定します。

使用できる値は、次の通りです。

**\*PRV** このコマンドの最後の呼び出しで使用された値が省略時の値となります。

### **\*APPC**

ホスト・サーバーはAPPCプロトコルを使用してワークステーションと通信します。

### **\*TCPIP**

ホスト・サーバーはTCP/IPプロトコルを使用してワークステーションと通信します。

上

上

---

## TCP/IPポート番号 (PORT)

ホスト・サーバーによって処理されるTCP/IPプロトコルと関連したポートの名前を指定します。

使用できる値は、次の通りです。

**\*PRV** このコマンドの最後の呼び出しで使用された値が省略時の値となります。

**\*DFT** システムとの通信に使用された値が省略時のCODE TCP/IPポートとなります。ポートを変更したい場合には、新しいTCP/IPポートの値を入力してください。

上

---

## ユーザー識別コード (USERID)

ワークステーションが「通信プロパティ」ウィンドウを介して会話機密保護を使用可能にした時に、通信の確立に使用するユーザーIDを指定します。

「通信プロパティ」ウィンドウで会話機密保護がセットアップされている場合には、このフィールドと「パスワード」フィールドの両方が必要です。使用できる値は、次の通りです。

### \*NONE

省略時の値は\*NONEです。

### ユーザーID

ワークステーションの「通信プロパティ」ウィンドウのSTRCODEユーザーIDフィールドに指定されたユーザーIDを入力してください。

上

---

## パスワード (PASSWORD)

「ユーザー識別コード」フィールドに指定されたユーザーIDのパスワードを指定します。このパスワードは、「通信プロパティ」ウィンドウのSTRCODEパスワード・フィールドに指定されたものと同じでなければなりません。

「通信プロパティ」ウィンドウで会話機密保護プロファイルがセットアップされている場合には、このフィールドと「ユーザー識別コード」フィールドの両方が必要です。

上

---

## 例

なし

上

---

## エラー・メッセージ

不明

上



# CODEコマンドの開始 (STRCODECMD)

実行可能場所:

- バッチ・ジョブ (\*BATCH)
- 対話式ジョブ (\*INTERACT)
- バッチ・プログラム (\*BPGM)
- 対話式プログラム (\*IPGM)
- バッチ REXX プロシージャ (\*BREXX)
- 対話式 REXX プロシージャ (\*IREXX)
- QCMDEXEC, QCAEXEC, または QCAPCMD API (\*EXEC) の使用

スレッド・セーフ: いいえ

パラメーター

例

エラー・メッセージ

CLコマンドSTRCODECMDは、ワークステーションで実行されるコマンドを投入します。

このコマンドを使用するためには、ワークステーションに連携開発環境/400 (CODE/400)プロダクトが導入されていなければなりません。

上

## パラメーター

キーワード	記述	選択項目	ノート
CMD	ワークステーション・コマンド	文字値	必須, 定位置 1

上

## ワークステーション・コマンド (CMD)

実行されるワークステーション・コマンドを指定します。ユーザーのワークステーションを見つけるために、STRCODE \*USRSPCオブジェクト中の値が使用されます。この値は一般的にはCODE開始(STRCODE)コマンドに最後に指定するものです。このオブジェクトがライブラリー・リストに見つからない場合には、CODE通信は自動的にリモート・ロケーションを解決しようとします。このコマンドがバッチ・ジョブで実行された時には、CODE通信はリモート・ロケーションを自動的に解決できません。さらに、APPC通信では、システム間通信機能(ICF)ファイルが必要になります。このコマンドは、ライブラリー・リストでICFファイルEVFCICFFを検索し、これを使用してユーザーのワークステーションに情報を送信します。

上

## 例

なし

上



---

## エラー・メッセージ

不明

上

## ISDBの開始 (STRISDB)

実行可能場所: 対話環境 (\*INTERACT \*IPGM \*IMOD \*IREXX  
\*EXEC)

スレッド・セーフ: いいえ

パラメーター

例

エラー・メッセージ

### 対話型ソース・デバッガーの開始

対話型ソース・デバッガー(ISDB)ツールは、CL, COBOL,またはRPGプログラムに対する対話式デバッグ環境を提供します。

注: ISDBを実行するためには、ライブラリーQTEMPが必要です。ISDBは、ライブラリーQTEMPを使用して、ログ・ファイルなどのファイルにアクセスします。

これを開始するためには:

1. ユーザー・プログラムに必要なSTRISDBパラメーターを指定してください。
2. 実行キーを押してください。ISDBが開始します。

プログラムのソースはISDBソース画面に表示され、その画面に提供されている機能を使用してプログラムをデバッグすることができます。

### エラー・メッセージ: STRISDB

なし

上

### パラメーター

キーワード	記述	選択項目	ノート
PGM	プログラム	修飾オブジェクト名	必須, 定位置 1
	修飾子 1: プログラム	名前	
	修飾子 2: ライブラリー	名前, *CURLIB, *LIBL	
UPDPD	実動ファイルの更新	*YES, *NO	オプション, 定位置 2
INVPGM	プログラムの呼び出し	*YES, *NO, *CMD	オプション, 定位置 3
PARM	呼び出しのパラメーター	値 (最大 40 回の繰り返し): 文字値	オプション, 定位置 4
CMD	コマンドの呼び出し	コマンド・ストリング	オプション, 定位置 5
SRCMBR	ソース・メンバー	名前, *PGM	オプション, 定位置 6
SRCF	ソース・ファイル	修飾オブジェクト名	オプション, 定位置 7
	修飾子 1: ソース・ファイル	名前	
	修飾子 2: ライブラリー	名前, *CURLIB, *LIBL	
SRVJOB	サービスするジョブ	*, *SELECT	オプション, 定位置 8

上

---

## プログラム

デバッグするプログラムの名前およびライブラリーを指定するためには、このパラメーターを使用してください。

指定できるライブラリーの値は次の通りです。

**\*LIBL** プログラムを見つけるためにライブラリー・リストが使用されます。ライブラリー名が指定されていない場合には、これが省略時の値となります。

**\*CURLIB**

プログラムを見つけるためにライブラリー・リスト中の現行ライブラリーが使用されます。(現行ライブラリーが指定されていない場合には、QGPLと見なされます。)

### ライブラリー名

デバッグするプログラムが入っているライブラリーの名前を指定してください。

注: ISDBでCOBOLプログラムをデバッグするためには、\*SRCDBGオプション付きでコンパイルされていることを確認してください。

上

---

## 実動ファイルの更新(UPDPROD)

実行用ライブラリーの中のファイルがデバッグ・モードになっている時に、それらを変更することができるかどうかを指定するためには、このパラメーターを使用してください。

使用できる値は次の通りです。

**\*NO** ファイルがデバッグ・モードになっている時は、それを更新することはできません。

**\*YES** ファイルがデバッグ・モードになっている時にも、それを更新することができます。

ソース画面から実動ファイルの更新値を変更することができることに注意してください。そうするためには、次のようにしてください。

1. ソース画面からデバッグ・メニュー・バー選択項目を選択してください。
2. 「デバッグ」プルダウン・ウィンドウでオプション1（デバッグ変更）を選択してください。「デバッグ・オプションの設定」ウィンドウが現れます。
3. オプション1（実動ファイルの更新）またはオプション2（実動ファイルを更新しない）のいずれかを選択して、実行キーを押してください。

上

---

## プログラムの呼び出し

プログラムを呼び出すためにISDBが必要であるかどうかを指定するこのパラメーターを使用するか、プログラムを呼び出す前に他の準備を実行するか、あるいはプログラムを呼び出すためのコマンドを使用します。

使用できる値は次の通りです。

**\*YES** ISDBは、PARMフィールドに指定したパラメーターでプログラムを呼び出します。

注: サービス・ジョブの場合には、このパラメーターを使用することができません。

**\*NO** プログラムを開始するためのコマンドは出されません。プログラムを開始するためのコマンドを出す前に必要な準備を実行できるように、コマンド入力画面(QCMD)が提供されます。

コマンド入力画面(QCMD)からプログラムを開始した場合には、プログラムが実行を完了した時に、制御はこの画面に戻ります。ISDBを終了するかまたは再始動するためには、最初にこのコマンド入力画面を終了しなければなりません(F3またはF12)。次にISDBの終了または再始動を選択できるプログラムの終了画面が表示されます。

注: 他のジョブのサービス中の場合には、これは使用できるただ1つのオプションですが、QCMDパネルは呼び出されません。

**\*CMD** ISDBは、CMDパラメーターに指定されたコマンドを実行します。(このコマンドはユーザー・プログラムを呼び出す必要があります。)

上

---

## パラメーター

プログラムを呼び出すために必要なパラメーターを指定するためには、このパラメーターを使用します。

このパラメーターは、INVPGM(\*NO)およびINVPGM(\*CMD)呼び出しコマンド用のものではありません。

注: このフィールドには空値(X'00')文字の入っている数値リテラル(500など)およびストリングは使用できません。代わりにINVPGM(\*CMD)パラメーターを使用してCMDパラメーターに適切なCALLコマンドを指定してください。たとえば、次のコマンドを入力するかわりに

```
STRISDB PGM(MYPGM) INVPGM(*YES) PARM(123 X'00')
```

次のコマンドを使用してください。

```
STRISDB PGM(MYPGM) INVPGM(*CMD) CMD(CALL PGM(MYPGM)
PARM(123 X'00'))
```

上

---

## 呼び出しコマンド

プログラムを呼び出すために使用したいコマンドを指定するためには、このパラメーターを使用します。

このパラメーターは、INVPGM(\*NO)およびINVPGM(\*YES)呼び出しコマンド用のものではありません。

上

---

## ソース・メンバー

このパラメーターは任意指定です。ソース・メンバーを指定する必要があるのは、それがプログラムのオブジェクト記述に指定されているものと異なる場合だけです。指定したこのメンバーのソースは、プログラムを呼び出した時に、ソース画面に表示されます。

次の場合には、このパラメーターを使用する必要があります。

- プログラムが最後にコンパイルされた以降に、ソースのライブラリー、ファイル、またはメンバー名が変更された場合。(メンバーに正しいプログラム・ソースが入っていることを確認してください。そうでないと、予測できない結果となります。)
- プログラムがRPG報告書簡易作成プログラムである場合。CRTTRPTPGMコマンドには、展開ソースをソース物理ファイルに入れるためのパラメーターがあります。
- ソースが別のサーバー・マシン上にあって、そのプログラムがDDMファイルを使用して作成されたものでない場合。

使用できる値は次の通りです。

**\*PGM** DSPOBJDコマンドを使用して、オブジェクト記述からソース情報が検索されます。

#### メンバー名

表示したいソース・メンバーの名前を指定して、そのファイルおよびライブラリー名を、示されたパラメーターに入力してください。

上

---

## ソース・ファイルおよびライブラリー

ソース・メンバー・プロンプトで指定したソース・メンバーのファイルおよびライブラリー名を指定するためには、これらのパラメーターを使用してください。ファイルおよびライブラリーのいずれかの名前がプログラムのオブジェクト記述に指定されている名前と異なる場合には、両方の名前を指定しなければなりません。

ソース・ファイル名として指定できる値は次の通りです。

#### ファイル名

表示したいソース・メンバーが入っているファイルの名前を指定してください。

ソース・ライブラリー名として指定できる値は次の通りです。

**\*LIBL** ライブラリー・リストでソース・ファイルを探します。

#### **\*CURLIB**

ソース・ファイルを見つけるために現行ライブラリーが使用されます。(現行ライブラリーが指定されていない場合には、QGPLと見なされます。)

#### ライブラリー名

表示したいソース・ファイルが入っているライブラリーの名前を指定してください。

上

---

## サービスするジョブ(SRVJOB)

サイン・オンしているジョブとは異なるジョブで実行されているプログラムをデバッグするためには、このパラメーターを使用してください。これは、バッチ・ジョブまたは他の対話式ジョブをデバッグする場合に有効です。

指定できる値は次の通りです。

**\*** 現行ジョブでデバッグします。

### **\*SELECT**

活動ジョブのリストからジョブを選択することのできる「サービスするジョブの選択」画面が表示されます。これらのジョブの1つを選択すると、STRSRVJOBコマンドが出されて、そのジョブがデバッグ・モードになります。

注: SRVJOB(\*SELECT)はINVPGM(\*YES)またはINVPGM(\*CMD)と一緒にでは正しくありません。

上

---

### **例**

なし

上

---

### **エラー・メッセージ**

なし

上



---

## PDM開始 (STRPDM)

実行可能場所: 対話環境 (\*INTERACT \*IPGM \*IREXX \*EXEC)

スレッド・セーフ: いいえ

パラメーター

例

エラー・メッセージ

PDM開始(STRPDM)コマンドは、プログラム開発管理機能(PDM)ユーティリティを呼び出します。メニューが表示され、このメニューで、ライブラリー、オブジェクト、メンバー、およびユーザー定義オプションを処理するためのオプションを選択します。

このコマンドにはパラメーターはありません。

上

---

### パラメーター

なし

上

---

### 例

STRPDM

このコマンドは、プログラム開発管理機能(PDM)ユーティリティを開始し、PDMの初期メニューを表示します。

上

---

### エラー・メッセージ

\*ESCAPE メッセージ

**PDM0055**

&1コマンドの処理中にエラーが起こった。

上





## 報告書設計ユーティリティー開始 (STRRLU)

実行可能場所: 対話環境 (\*INTERACT \*IPGM \*IREXX \*EXEC)  
スレッド・セーフ: いいえ

パラメーター  
例  
エラー・メッセージ

制御言語(CL)コマンドSTRRLUは、報告書設計ユーティリティー(RLU)を開始するために使用されるコマンドです。

報告書設計ユーティリティーを開始するためのパラメーター値を指定するためには、この画面を使用してください。

### エラー・メッセージ: STRRLU

なし

上

### パラメーター

キーワード	記述	選択項目	ノート
SRCFILE	ソース・ファイル	単一値: <u>*PRV</u> その他の値: 修飾オブジェクト名	オプションル, 定位置 1
	修飾子 1: ソース・ファイル	名前	
	修飾子 2: ライブラリー	名前, <u>*LIBL</u> , *CURLIB, *PRV	
SRCMBR	ソース・メンバー	名前, <u>*PRV</u>	オプションル, 定位置 2
OPTION	オプション	<u>2</u> , 6	オプションル, 定位置 3
PAGWIDTH	ページ幅	1-378, <u>*SAME</u>	オプションル, 定位置 4
TEXT	テキスト'記述'	文字値, <u>*BLANK</u>	オプションル

上

### ソース・ファイル (SRCFILE)

前のセッションで変更または作成したメンバーが入っているソース物理ファイル、あるいは新しいメンバーを記憶したいソース物理ファイルの修飾名を指定します。考えられる値は、次の通りです。

\*PRV 最後に使用したソース物理ファイルの修飾名を指定します。

名前 既存のソース・ファイル名を指定します。

### Warning: Temporary Level 3 Header

#### ライブラリー(LIB)

前のセッションで変更または作成したメンバーが入っているライブラリー、あるいはソース・ファイルを記憶したいライブラリーの修飾名を指定します。考えられるライブラリーの値は、次の通りです。

**\*LIBL** ライブラリー・リストが表示され、ここでソース・ファイルを含むライブラリー、またはソース・ファイルを記憶したいライブラリーを選択することができます。

**\*CURLIB**

ソース・ファイルを記憶するためにジョブの現行ライブラリーが使用されます。ライブラリー・リストに現行ライブラリー項目が存在しない場合には、ユーザーのソース・ファイルはQGPLに入れます。

**名前** ソース・ファイルが入っているライブラリーを指定します。

上

---

## ソース・メンバー (SRCMBR)

変更または作成されたメンバーの名前を指定します。SRCFILEパラメーターを使用する場合には、このパラメーターの省略時の値が変更されます。考えられる値は、次の通りです。

**\*PRV** 前に使用したソース・メンバー名を選択します。

**名前** 前のセッションで変更または作成したメンバーの名前、あるいは新しいメンバーを作成している場合には、必要な名前を指定します。

上

---

## オプション (OPTION)

報告書进行处理するためには、オプションを指定します。考えられる値は、次の通りです。

**2** 報告書の変更

**6** プロトタイプ報告書の印刷

上

---

## ページ幅 (PAGWIDTH)

ページ幅を行当たりの桁数で指定します。省略時の値は\*SAMEです。RLUを最初に使用する時に、他の値を指定しない場合は、ページ幅は132にセットされます。メンバーがRLUによって前に編集されている場合には、ページ幅はそのメンバーの前の編集セッションから取り出されます。

**\*SAME**

報告書を最後に作成または変更した時に使用した幅と同じ報告書の幅を指定します。

**1 - 378**

ページの幅を決定するためには、1から378の範囲の値を入力してください。

上

---

## テキスト'記述' (TEXT)

記述を指定して、メンバーのテキスト・プロンプトに記憶します。考えられる値は、次の通りです。

### **\*BLANK**

新しいメンバーの場合には、RLUはそのメンバーの テキスト プロンプトにブランクを指定します。既存のメンバーの場合には、この省略時の値によってメンバーの テキスト プロンプトは変更されません。

’記述’ 50桁以内のテキストを指定します。

上

---

### **例**

なし

上

---

### **エラー・メッセージ**

なし

上



# RSEサーバーの開始 (STRRSESVR)

実行可能場所: 対話環境 (\*INTERACT \*IPGM \*IREXX \*EXEC)  
スレッド・セーフ: いいえ

パラメーター  
例  
エラー・メッセージ

RSEサーバーの開始 (STRRSESVR)コマンドは、REMOTE SYSTEM EXPLORER (RSE)通信サーバーがまだ開始されていない場合にはこれを開始し、現行の対話式ジョブを指定されたRSE接続に関連付けます。

STRRSESVRコマンドの目的は、対話式アプリケーションを実行してデバッグするために、エミュレーターをRSE接続に関連付けることです。ジョブは、関連したRSE接続での排他使用のためにロックされます。ジョブは、関連したRSE接続を切断するか、あるいはそのRSE接続に対して「対話式ジョブ解放」処置を行うことによって解放することができます。

上

## パラメーター

キーワード	記述	選択項目	ノーツ
NAME	CONNECTION NAME	文字値, <u>*PRV</u>	オプションル, 定位置 1
WRKLIB	作業ライブラリー	名前, <u>*PRV</u> , *DFT	オプションル, 定位置 2
RMTLOCNAME	リモート・ロケーション名	文字値, *RESOLVE, <u>*PRV</u>	オプションル, 定位置 3
PORT	TCP/IPポート	1-65534, <u>*PRV</u> , *DFT	オプションル

上

## CONNECTION NAME (NAME)

この対話式ジョブと関連付ける必要があるRSE接続を指定します。この接続名は、それだけで指定することができますが、任意でRSEプロファイル名により修飾することができます。

\*PRV STRRSESVRコマンドの直前の呼び出しで指定された接続名が使用されます。直前の呼び出しが存在しない場合には、接続名を指定しなければなりません。

文字値 このジョブと関連付ける必要がある接続名を指定します。

上

## 作業ライブラリー (WRKLIB)

RSEサーバーと通信するために、STRRSESVRコマンドがデータ待ち行列を作成するライブラリーを指定します。

\*PRV STRRSESVRコマンドの直前の呼び出しで指定されたライブラリー名が使用されます。ライブラリーを指定しなかった場合は、QGPLが使用されます。

\*DFT ライブラリーQGPLが使用されます。

**名前** 使用されるライブラリーの名前を指定します。

上

---

## リモート・ロケーション名 (RMTLOCNAME)

RSE通信デーモンを実行しているクライアント・マシンのTCP/IPホスト名またはIPアドレスを指定します。

### **\*RESOLVE**

STRRSESVRコマンドにリモート・ロケーション名の解決を試みさせます。この値はTCP/IP DHCPユーザーにお勧めします。

**\*PRV** STRRSESVRコマンドの直前の呼び出しで指定されたホスト名が使用されます。ホスト名を指定しなかった場合は、\*RESOLVEが使用されます。

**文字値** クライアント・マシンのTCP/IPホスト名またはIPアドレスを指定します。

上

---

## TCP/IPポート (PORT)

RSE通信デーモンがLISTENするポート番号を指定します。

**\*PRV** STRRSESVRコマンドの直前の呼び出しで指定されたポート番号が使用されます。ポート番号を指定しなかった場合は、ポート4300が使用されます。

**\*DFT** 省略時のポート(4300)が使用されます。

### **1-65534**

使用されるTCP/IPポート番号を指定します。

上

---

## 例

### **例1: 接続名のみを指定してRSEサーバーを開始**

```
STRRSESVR NAME(DEVELOPMENT)
```

このコマンドは、現行の対話式ジョブを、名前がDEVELOPMENTのRSE接続に関連付けます。

### **例2: 接続名と追加のパラメーターを指定してRSEサーバーを開始**

```
STRRSESVR NAME(TEST.DEVELOPMENT) RMTLOCNAME('192.168.1.45')  
WRKLIB(DEVLIB)
```

このコマンドは、現行の対話式ジョブを、指定のIPアドレスをもつクライアント・マシン上のTESTプロファイルでDEVELOPMENTという名前のRSE接続に関連付けます。このジョブとRSEとの間の通信に使用されるデータ待ち行列は、ライブラリーDEVLIB内に作成されます。

上

---

## エラー・メッセージ

### \*ESCAPE メッセージ

#### **RSE2001**

REMOTE SYSTEMS EXPLORERへの接続エラーです。

#### **RSE2002**

接続名が定義されていません。

#### **RSE2003**

接続は異なるホスト名に対して定義されています。

#### **RSE2004**

データ待ち行列の作成エラーです。

#### **RSE2005**

作業ライブラリーが無効です。

#### **RSE2006**

RSEサーバーの始動または接続が失敗しました。

#### **RSE2007**

データ待ち行列のアクセス・エラーです。

#### **RSE2008**

データ待ち行列を削除することができませんでした。

#### **RSE2009**

リモート・ロケーションIPアドレスの解決エラーです。

#### **RSE2010**

接続はすでに使用中です。

#### **RSE2011**

ソケット・エラーです。

#### **RSE2012**

ジョブ情報の検索エラーです。

#### **RSE2013**

前の接続名が存在していません。

上





# SDAの開始 (STRSDA)

実行可能場所: 対話環境 (\*INTERACT \*IPGM \*IREXX \*EXEC)

スレッド・セーフ: いいえ

パラメーター  
例  
エラー・メッセージ

CLコマンドSTRSDAは、IBM画面設計機能(SDA)ユーティリティの基本コマンドです。このコマンドは、システムの3つの環境のいずれでも呼び出すことができます。

## エラー・メッセージ： STRSDA

### \*ESCAPE メッセージ

- SDA0001  
SDAは&1ワークステーションをサポートしていない。
- SDA0002  
SDAはその表示装置ファイルをアクセスすることができない。
- SDA0003  
重大なエラーのためにSDAが終了した。
- SDA0004  
STRSDAコマンドの処理中にエラーが起こった。
- SDA0005  
SDAはそのパネル・グループをアクセスすることができない。
- SDA0601  
SDAはすでに活動中である。

上

## パラメーター

キーワード	記述	選択項目	ノーツ
OPTION	SDAオプション	*SELECT, 1, 2, 3	オプション, 定位置 1
SRCFILE	ソース・ファイル	修飾オブジェクト名	オプション, 定位置 2
	修飾子 1: ソース・ファイル	名前, *PRV	
	修飾子 2: ライブラリー	名前, *PRV, *LIBL, *CURLIB	
SRCMBR	ソース・メンバー	名前, *PRV, *SELECT	オプション, 定位置 3
OBJLIB	オブジェクト・ライブラリー	名前, *PRV, *CURLIB	オプション
JOBID	ジョブ記述	修飾オブジェクト名	オプション
	修飾子 1: ジョブ記述	名前, *PRV, *USRPRF	
	修飾子 2: ライブラリー	名前, *PRV, *LIBL, *CURLIB	
TSTFILE	テスト・ファイル	修飾オブジェクト名	オプション
	修飾子 1: テスト・ファイル	名前, *PRV	
	修飾子 2: ライブラリー	名前, *PRV, *LIBL, *CURLIB	

キーワード	記述	選択項目	ノーツ
MODE	モード	<u>*STD</u> , *S38, *S36	オプション

上

## SDAオプション (OPTION)

SDAメインメニューの値として使用するオプションを指定します。MODE (\*S36) が指定されている場合は、このパラメーターは無視されます。

考えられる値は次の通りです。

### \*SELECT

SDAメインメニューを表示します。

### メインメニュー・オプション番号

SDAメインメニュー上のオプションに対応する1から3の範囲の数字を入力してください。このパラメーター値を選択した場合は、SDAメインメニューは現れません。

上

## ソース・ファイル (SRCFILE)

更新されるソース・メンバーが入っているか、または新しいソース・メンバーの追加が行なわれるソース・ファイルの名前を指定します。MODE (\*S36)が指定されている場合は、ライブラリー修飾子だけが使用されます。

考えられる値は次の通りです。

\*PRV SDAで、最後のSDAセッションで使用されたソース・ファイルおよびライブラリーの名前を使用する(MODEが(\*STD)の場合のみ) ことを指定します。

### 修飾ソース・ファイル名

SDAによって使用される既存のソース・ファイルの修飾名を入力してください。\*CURLIBがライブラリー修飾子として指定された場合は、ライブラリー・リストの現行ライブラリー項目が検索されます。現行ライブラリーがライブラリー・リストに存在しない場合は、ライブラリーQGPLが検索されます。\*LIBLがライブラリー修飾子として指定された場合は、ファイルを見つけるためにライブラリー・リストが使用されます。

上

## ソース・メンバー (SRCMBR)

SDAによって更新または作成される画面またはメニューのソースが現在入っているか、または今後入ることになる既存または新しいソース・ファイル・メンバーの名前を指定します。

次の値を指定することができます

\*PRV SDAで、最後のSDAセッションで使用されたソース・メンバーの名前を使用する(MODEが(\*STD)の場合のみ) ことを指定します。

### **\*SELECT**

SDAは画面設計画面を表示します。この画面から、メンバー・フィールドでF4キーを押して、そこから選択するソース・メンバーのリストを表示することができます。

### **ソース・メンバー名**

作成または更新されるソース・メンバーの名前を入力してください。

上

---

## **オブジェクト・ライブラリー (OBJLIB)**

SDAが作成するプログラムまたは表示装置ファイルが記憶されるライブラリーの名前を指定します。

考えられる値は次の通りです。

**\*PRV** SDAで、最後のSDAセッションで使用されたオブジェクト・ライブラリーの名前を使用する (MODEが(\*STD)の場合のみ) ことを指定します。

### **\*CURLIB**

SDAで、ライブラリー・リストの現行ライブラリー項目を使用することを指定します。現行ライブラリー項目がライブラリー・リストに存在しない場合は、ライブラリーQGPLが使用されます。

### **オブジェクト・ライブラリー名**

SDAによって作成されたオブジェクトが記憶されるライブラリーの名前を入力してください。

上

---

## **ジョブ記述 (JOBBD)**

SDAによって投入されたバッチ・ジョブで使用されるジョブ記述の修飾名を指定します。MODE (\*S36)が使用されている場合は、このパラメーターは使用されません。

考えられる値は次の通りです。

**\*PRV** SDAで、最後のSDAセッションで使用されたジョブ記述およびライブラリーの名前を使用する (MODEが(\*STD)の場合のみ) ことを指定します。

### **ジョブ記述名**

投入されたジョブで使用されるジョブ記述の修飾名を入力してください。\*CURLIBがライブラリー修飾子として指定された場合は、ライブラリー・リストの現行ライブラリー項目が検索されます。現行ライブラリーがライブラリー・リストに存在しない場合は、ライブラリーQGPLが検索されます。\*LIBLがライブラリー修飾子として指定された場合は、ファイルを見つけるためにライブラリー・リストが使用されます。

### **\*USRPRF**

SDAで、ユーザー・プロファイルの中で定義されているジョブ記述の名前を使用することを指定します。

上

---

## テスト・ファイル (TSTFILE)

テストのために使用される表示装置ファイルの修飾名を指定します。MODE (\*S36) が指定されている場合は、このパラメーターは無視されます。

考えられる値は次の通りです。

**\*PRV** SDAで、最後のSDAセッションで使用された表示装置ファイルおよびライブラリーの名前を使用する(MODEが(\*STD)の場合のみ) ことを指定します。

### テスト・ファイル名

テストのために使用される表示装置ファイルの修飾名を入力してください。\*CURLIBがライブラリー修飾子として指定された場合は、ライブラリー・リストの現行ライブラリー項目が検索されます。現行ライブラリーがライブラリー・リストに存在しない場合は、ライブラリーQGPLが検索されます。\*LIBLがライブラリー修飾子として指定された場合は、ファイルを見つけるためにライブラリー・リストが使用されます。

上

---

## モード (MODE)

SDAのどのバージョンを使用するかを指定します。

考えられる値は次の通りです。

**\*STD** SDAを指定し、SDAメインメニューを表示します。OPTIONパラメーターが指定された場合は、メインメニューは表示されません。

**\*S36** SDAのシステム/36環境を指定し、システム/36のSDAメインメニューを表示します。

**\*S38** SDAのシステム/38ビューを指定し、システム/38のSDAメインメニューを表示します。OPTIONパラメーターが指定された場合は、メインメニューは表示されません。

上

---

## 例

なし

上

---

## エラー・メッセージ

### \*ESCAPE メッセージ

#### SDA0001

SDAは&1ワークステーションをサポートしていない。

#### SDA0002

SDAはその表示装置ファイルをアクセスすることができない。

#### SDA0003

重大なエラーのためにSDAが終了した。

**SDA0004**

STRSDAコマンドの処理中にエラーが起こった。

**SDA0005**

SDAはそのパネル・グループをアクセスすることができない。

**SDA0601**

SDAはすでに活動中である。

上



# SEU開始 (STRSEU)

実行可能場所:

- バッチ・ジョブ (\*BATCH)
- 対話式ジョブ (\*INTERACT)
- 対話式プログラム (\*IPGM)
- 対話式 REXX プロシージャー (\*IREXX)
- QCMDEXEC, QCAEXEC, または QCAPCMD API (\*EXEC) の使用

スレッド・セーフ: いいえ

パラメーター  
例  
エラー・メッセージ

STRSEU（原始ステートメント入力ユーティリティ開始）コマンドによって、ソース・メンバーを作成、変更、表示、または印刷することができます。

## エラー・メッセージ: STRSEU

\*ESCAPE メッセージ

EDT9007

&1コマンドでエラーが見つかった。

上

## パラメーター

キーワード	記述	選択項目	ノーツ
SRCFILE	ソース・ファイル	単一値: <u>*PRV</u> その他の値: 修飾オブジェクト名	オプションル, 定位置 1
	修飾子 1: ソース・ファイル	名前	
	修飾子 2: ライブラリー	名前, <u>*LIBL</u> , *CURLIB, *PRV	
SRCMBR	ソース・メンバー	名前, <u>*PRV</u> , *SELECT	オプションル, 定位置 2



キーワード	記述	選択項目	ノート
<b>TYPE</b>	ソース仕様タイプ	単純名, <b>*SAME</b> , BAS, BASP, BND, C, CBL, CBLLE, CICSC, CICSCBL, CICSCBLLE, CICSMA, CICSSQLCBL, CL, CLD, CLP, CLLE, CMD, CPP, DFU, DSPF, FTN, ICF, LF, MENU, MNU, MNUCMD, MNUDDS, PAS, PF, PLI, PNLGRP, PRTF, QRY, REXX, RMC, RPG, RPGLE, RPT, SPADCT, SQLC, SQLCBL, SQLCBLLE, SQLFTN, SQLPLI, SQLRPG, SQLRPGLE, SRT, TBL, TXT, BAS38, BASP38, BSCF38, CBL38, CL38, CLP38, CMD38, CMNF38, DFU38, DSPF38, LF38, MXDF38, PF38, PLI38, PRTF38, QRY38, RPG38, RPT38, SRT38, TXT38, ARS36, ASM36, BAS36, BASP36, BGC36, BGD36, BGF36, CBL36, DFU36, DTA36, DSPF36, FOR36, MNU36, MSGF36, OCL36, PHL36, RPG36, RPT36, SRT36, TXT36, UNS36, WSU36	オプション, 定位置 3
<b>OPTION</b>	オプション	<b>*BLANK</b> , ' ', 2, 5, 6	オプション, 定位置 4
<b>TEXT</b>	テキスト'記述'	文字値, <b>*BLANK</b>	オプション

上

## ソース・ファイル (SRCFILE)

編集または作成するメンバーが入っているソース物理ファイルおよびライブラリーの名前を入力してください。

考えられる値は、次の通りです。

**\*PRV** SEUが最後のSEUセッションで使用されたソース・ファイルおよびライブラリーの名前を使用することを指定します。ソース・ファイル・パラメーターに**\*PRV**を指定する場合には、ライブラリーを指定する必要はありません。

### ソース・ファイル名

使用する既存のソース・ファイルの名前を入力してください。ソース・ファイル名およびライブラリー名を指定した場合には、SEUは指定したライブラリーでソース・ファイルを検索します。ライブラリー名とともにソース・ファイル名を指定しない場合には、**\*LIBL**が使用されます。

### ライブラリー名

使用される既存のライブラリーの名前を入力します。ライブラリーとして**\*CURLIB**を使用した場合には、SEUはライブラリー・リストの**現行ライブラリー**を検索します。ライブラリーとして**\*LIBL**を使用した場合には、SEUはファイルをライブラリー・リスト内のライブラリーで検索します。

---

## ソース・メンバー (SRCMBR)

編集または作成するソース物理ファイル・メンバー名を指定します。このパラメーターの省略時の値は、SRCFILEパラメーターを指定したかどうかによって決まります。

考えられる値は、次の通りです。

### \*SELECT

これはSRCFILEパラメーターを指定した場合の省略時の値です。\*SELECTを選択した場合には、指定したファイルおよびライブラリーのすべてのメンバーのリストが表示されます。ここから編集、走査検索、印刷、または削除するメンバーを選択してください。

\*PRV これは、SRCFILEパラメーターを指定しない場合の省略時の値です。\*PRVは前のソース物理ファイル・メンバー名かまたはメンバー処理画面です。

### ソース・ファイル・メンバー名

作成または編集するソース物理ファイル・メンバー名を入力してください。

---

## ソース仕様タイプ (TYPE)

編集または作成するソース・メンバーのタイプを指定します。使用できる値は、次の通りです。

### \*SAME

省略時の値は、このメンバーを最後に編集した時に使用されたものと同じタイプです。新しいメンバーの場合には、省略時の値はTXTです。

**タイプ** この値によって、使用するソース仕様のタイプを指定できます。最大10桁の任意の文字を指定するか、あるいはSEUでサポートしているタイプを指定することができます。

メンバーは、意味のある任意のタイプにすることができます。SEUは、次のメンバー・タイプをサポートしています。

### オペレーティング・システム・タイプ

BAS, BASP, BND, C, CBLLE, CBL, CICSC, CICSCBLLE, CICSCBL, CICS MAP, CICSSQLCBL, CL, CLD, CLLE, CLP, CMD, CPP, DFU, DSPF, FTN, ICF, LF, MENU, MNU, MNUCMD, MNUDDS, PAS, PF, PLI, PNLGRP, PRTF, QRY, REXX, RMC, RPG, RPGLE, RPT, SPADCT, SQLC, SQLCLE, SQLCBL, SQLCBLLE, SQLFTN, SQLPLI, SQLRPG, SQLRPGLE, SRT, TBL, およびTXT。

### システム/38タイプ

BAS38, BASP38, BSCF38, CBL38, CL38, CLP38, CMD38, CMNF38, DFU38, DSPF38, LF38, MXDF38, PF38, PLI38, PRTF38, QRY38, RPG38, RPT38, SRT38, およびTXT38

### システム/36タイプ

ARS36, ASM36, BAS36, BASP36, BGC36, BGD36, BGF36, CBL36, DFU36, DSPF36, DTA36, FOR36, MNU36, MSGF36, OCL36, PHL36, RPG36, RPT36, SRT36, TXT36, UNS36, およびWSU36

上記のタイプの他に、ユーザーが独自のメンバー・タイプ名を選択することもできます。

---

## オプション (OPTION)

選択したメンバーで実行する機能を指定します。省略時の値はメンバー名を指定するかどうかで決まります。メンバー名を指定しない場合の省略時の値は\*BLANKで、処置を行なわないことを示します。メンバー名を指定した場合の省略時の値は2（編集）です。これはメンバーの編集セッションを行なうことを示します。

使用できる値は次の通りです。

### \*BLANKまたは' '

これはメンバー名を指定しない場合の省略時の値です。\*BLANKは、処置が行なわれないことを示します。

### 2=メンバーの編集

編集画面に進むためには、オプション・パラメーターを選択した後に2を入力します。

### 5=メンバーの走査検索

走査検索画面に進むためには、オプション・パラメーターを選択した後に5を入力します。

### 6=メンバーの印刷

指定したメンバーを印刷するためには、6を入力してください。

---

## テキスト'記述' (TEXT)

メンバー用のテキスト・フィールドでメンバーを記述する文字ストリングを指定してください。

考えられる値は、次の通りです。

### \*BLANK

SEUが新しいメンバーのテキスト・フィールドにブランクを入力することを指定します。この省略時の値は既存のメンバーのテキスト・フィールドを変更しません。

**記述**    メンバーを記述する最大50桁までの文字ストリングを指定してください。先行または後書きのブランクを使用するためには、ストリングをアポストロフィで囲んでください。

---

## 例

なし

---

## エラー・メッセージ

### \*ESCAPE   メッセージ

**EDT9007**

&1コマンドでエラーが見つかった。

上



# PDM使用のライブラリーの処理 (WRKLIBPDM)

実行可能場所: 対話環境 (\*INTERACT \*IPGM \*IREXX \*EXEC)  
スレッド・セーフ: いいえ

パラメーター  
例  
エラー・メッセージ

PDM使用のライブラリーの処理 (WRKLIBPDM)コマンドを使用すると単一または複数ライブラリーを処理することができます。このコマンドを使用すると、プログラム開発管理機能(PDM)メニューおよび処理ライブラリー指定画面をバイパスすることができます。

上

## パラメーター

キーワード	記述	選択項目	ノーツ
LIB	ライブラリー	文字値, *PRV, *LIBL, *USRLIBL, *ALL, *ALLUSR, *CURLIB	オプションル, 定位置 1
ASP	ASP番号	1-32, *ALL	オプションル
ASPDEV	ASP装置	名前, *, *SYSBAS, *CURASPGRP	オプションル

上

## ライブラリー (LIB)

処理するライブラリーを指定します。

**\*PRV** 同じライブラリーまたは前のWRKLIBPDMセッションで処理したライブラリーを処理します。

**\*LIBL** ジョブのライブラリー・リストにあるすべてのライブラリーを処理します。

**\*USRLIBL**  
ジョブのライブラリー・リストのユーザー部分にあるすべてのライブラリーを処理します。

**\*ALL** QSYSおよびQTEMPを含めシステムにあるすべてのライブラリーを処理します。

**\*ALLUSR**  
すべてのユーザー・ライブラリーを含めすべての非システム・ライブラリーを処理します。ライブラリーはライブラリー名のアルファベット順でリストされます。

**\*CURLIB**  
ジョブの現行ライブラリーを処理します。ジョブの現行ライブラリーが指定されていない場合には、QGPLと見なされます。

**総称名** 処理するライブラリーの総称名を指定します。

総称名は次の形式の1つとすることができます。

**ABC\*** 文字ABCで始まるすべてのライブラリーのリストを表示します。たとえば, ABC, ABCD, またはABCTESTなど。

**\*ABC** 文字ABCで終るすべてのライブラリーのリストを表示します。たとえば, ABC, DABC, またはTESTABCなど。

- \*B\*** 名前の中に文字Bを含むすべてのライブラリーのリストを表示します。たとえば、B, BALL, ABCDなど。
- A\*C** 文字Aで始まり、文字Cで終わるすべてのライブラリーのリストを表示します。たとえば、AC, ABC, AZZZCなど。
- "A"** 名前が英字Aで始まり、引用符付きであるすべてのライブラリーのリストを表示します。たとえば、"A", "AB", "AD"など。
- \*\*ALL**  
ALLで終わるすべてのライブラリーのリストを表示します。たとえば、ALL, BALL,またはTESTALLなど。\*ALLはすべてのライブラリーのリストを表示する特殊値として定義されているので、この場合には、2個のアスタリスクが必要です。

**名前** 処理する単一ライブラリーの名前を指定します。

上

---

## ASP番号 (LIB)

- 「ライブラリーの処理」で表示するライブラリーの補助記憶域プール(ASP)を指定します。
- \*LIBL, \*CURLIBまたは\*USRLIBLの値がライブラリー・パラメーターに入力された場合、このパラメーターは無視されます。このパラメーターに数値が指定された場合、ASP装置(ASPDEV)のパラメーター値は\*でなければなりません。
- \*ALL** ASP装置(ASPDEV)のパラメーター値で定義されたすべてのASPが検索されます。

### 1から32

検索するシステムまたは基本ユーザーASPの番号を指定します。ASP 1はシステムASPで、常時構成されています。基本ユーザーASPは2から32であり、システム上で構成されているASPを指定する必要があります。ASPの構成については、BACKUP AND RECOVERY BOOK, SC41-5304を参照してください。

上

---

## ASP装置 (LIB)

表示されたライブラリーの記憶域が割り振られた補助記憶域プール(ASP)装置名を指定します。スレッドのライブラリー・ネーム・スペースにないASPにライブラリーがある場合、正しいライブラリーが表示されるようにこのパラメーターを指定する必要があります。ASP番号(ASP)パラメーターに数値が指定された場合、ASPDEVパラメーター値は\*でなければなりません。

- \*** 現在スレッドのライブラリー・ネーム・スペースの一部であるASPがライブラリー検出のために検索されます。これはシステムASP (ASP 1),定義済みのすべての基本ユーザーASP (ASP 2から32),およびスレッドにASPグループがある場合には、スレッドのASPグループの1次および2次ASPを含みます。

### \*SYSBAS

システムASP (ASP 1)および定義済みのすべての基本ユーザーASP (ASP 2から32)が、ライブラリーを検出するために検索されます。スレッドがASPグループを持つ場合でも1次および2次ASPは検索されません。

### \*CURASPGRP

スレッドがASPグループを持つ場合、スレッドのASPグループにある1次および2次ASPがライブラリー検出のために検索されます。システムASP (ASP 1)および定義済みの基本ユーザーASP (ASP 2から32)は検索されません。スレッドに関連するASPグループがない場合エラーが発生します。

**名前** 検索する1次または2次ASPの装置名を指定します。1次または2次ASPは(ASP装置をオンに変更して)アクティブにし、使用可能の状態である必要があります。システムASP (ASP 1)および構成済みの基本ユーザーASP (ASP 2から32) は検索されません。

上

---

## 例

### 例1: ジョブ・ライブラリー・リストにあるライブラリーの処理

```
WRKLIBPDM LIB('*LIBL')
```

このコマンドを使用すると、現行ジョブのライブラリー・リストにあるすべてのライブラリーを処理することができます。ライブラリーは、ライブラリー・リストと同じ順序でリストされます。

### 例2: 一般ライブラリーの処理

```
WRKLIBPDM LIB('*PAY*')
```

このコマンドを使用すると、ライブラリー名に文字**PAY**を含むライブラリーのサブセットを処理することができます。ライブラリーはアルファベット順でリストされます。

上

---

## エラー・メッセージ

### \*ESCAPE メッセージ

#### PDM0055

&1コマンドの処理中にエラーが起こった。

上





## PDM使用のメンバーの処理 (WRKMBRPDM)

実行可能場所: 対話環境 (\*INTERACT \*IPGM \*IREXX \*EXEC)

スレッド・セーフ: いいえ

パラメーター

例

エラー・メッセージ

PDM使用のメンバーの処理 (WRKMBRPDM)コマンドを使用すると1つのデータベース・ファイルにあるメンバーを処理することができます。このコマンドを使用すると、プログラム開発管理機能(PDM)メニューおよび処理メンバー指定画面をバイパスすることができます。

上

### パラメーター

キーワード	記述	選択項目	ノーツ
FILE	ファイル	単一値: <b>*PRV</b> その他の値: 修飾オブジェクト名	オプションル, 定位置 1
	修飾子 1: ファイル	名前	
	修飾子 2: ライブラリー	名前, *PRV, <b>*LIBL</b> , *CURLIB	
MBR	メンバー	文字値, <b>*ALL</b> , *PRV	オプションル, 定位置 2
MBRTYPE	メンバー・タイプ	文字値, <b>*ALL</b> , *PRV, *BLANK, BAS, BAS36, BAS38, BASP, BASP38, C, CBL, CBLLE, CBL36, CBL38, CICSC, CICSCBL, CICSCBLLE, CICSMAP, CICSSQLCBL, CLD, CLLE, CLP, CLP38, CMD, CMD38, CPP, DSPF, DSPF36, DSPF38, FTN, ICFE, LF, LF38, MENU, MNU, MNUCMD, MNUDDS, MNU36, MSGF36, OCL36, PAS, PF, PF38, PLI, PLI38, PNLGRP, PRTF, PRTF38, QRY38, REXX, RMC, RPG, RPGLE, RPG36, RPG38, RPT, RPT36, RPT38, SPADCT, SQLC, SQLCPP, SQLCBL, SQLCBLLE, SQLFTN, SQLPLI, SQLRPG, SQLRPGLE, TBL, TXT	オプションル, 定位置 3

上

---

## ファイル (FILE)

処理したいメンバーが入っているデータベースを指定します。ファイルは、ソース物理ファイルでも、データ物理ファイルでもかまいません。

### 単一値

**\*PRV** 前のWRKMBRPDMセッションで使用したものと同一ファイルのメンバーを処理します。

### 修飾子1: ファイル

**名前** 処理したいメンバーが入っている物理ファイルの名前を指定してください。

### 修飾子2: ライブラリー

**\*LIBL** 処理したいファイルを探すために現行ライブラリー・リストを検索するためには、\*LIBLを入力して、その後にファイル名を続けてください。WRKMBRPDM FILE(ファイル名)というコマンドを使用した場合には、PDMは指定されたファイルを見つけるためにライブラリー・リストを検索します。

**\*PRV** 前のWRKMBRPDMセッションで使用したものと同一ライブラリーにある物理ファイルを処理します。次の形式を使用することによって、前のライブラリーの別のファイルを指定することができます: FILE(\*PRV/QRPGSRC)

### \*CURLIB

ジョブの現行ライブラリーにある物理ファイルを処理します。現行ライブラリーが定義されていない場合には、QGPLが省略時の値です。

**名前** 処理したいファイルおよびメンバーを含むライブラリーの名前を入力してください。

上

---

## メンバー (MBR)

処理したいメンバーを指定します。このパラメーターを使用して、指定したファイル中のすべてのメンバーまたはメンバーのサブセットを処理することができます。

**\*ALL** 指定したファイルのすべてのメンバーを処理します。

**\*PRV** 前のWRKMBRPDMセッションで使用したものと同一メンバーを処理します。

**総称名** 処理したいファイル・メンバーの総称名を指定してください。

総称名は次の形式の1つとすることができます。

**ABC\*** 文字ABCで始まるすべてのメンバーのリストを表示します。たとえば、ABC, ABCD, またはABCTESTなど。

**\*ABC** 文字ABCで終わるすべてのメンバーのリストを表示します。たとえば、ABC, DABC, またはTESTABCなど。

**\*B\*** 名前の中に文字Bが入っているすべてのメンバーのリストを表示します。たとえば、B, BALL, ABCDなど。

**A\*C** 文字Aで始まり、文字Cで終わるすべてのメンバーのリストを表示します。たとえば、AC, ABC, AZZZCなど。

**"A"** 名前が英字Aで始まり、引用符付きであるすべてのメンバーのリストを表示します。たとえば、"A", "AB", "AD"など。

## **\*\*ALL**

ALLで終わるすべてのメンバーのリストを表示します。たとえば、ALL, BALL,またはTESTALLなど。\*ALLはすべてのメンバーのリストを表示する特殊値として定義されているので、この場合には、2個のアスタリスクが必要です。

**名前** 処理したい1つのファイル・メンバーの名前を指定します。

上

---

## **メンバー・タイプ (MBRTYPE)**

処理したいメンバーのメンバー・タイプを指定します。このパラメーターを使用して、指定したファイルにあるすべてのメンバー・タイプ、または、特定または総称メンバー・タイプに一致するメンバーのサブセットを処理することができます。

**\*ALL** すべてのメンバー・タイプのファイル・メンバー(タイプなしのものを含む)を処理します。

**\*PRV** 前のWRKMBRPDMセッションで使用したものと同じメンバー・タイプのメンバーを処理します。

### **\*BLANK**

メンバー・タイプ値のないメンバーを処理します。

### **メンバー・タイプ**

特定のタイプのすべてのメンバーのリストを表示するためにメンバー・タイプを指定してください。

作成したメンバー・タイプを使用するか、あるいはPDMコマンドによって使用される次の標準メンバー・タイプの1つを使用することができます。

**BAS** BASIC

**BAS36** システム/36 BASIC

**BAS38** システム/38 BASIC

**BASP** 固有モードのBASICプロシージャー

**BASP38**

システム/38固有モードのBASICプロシージャー

**C** C言語

**CBL** COBOL

**CBLLE**

統合化言語環境\* COBOL

**CBL36**

システム/36 COBOL

**CBL38**

システム/38 COBOL

**CICSC**

CICS C

**CICSCBL**

CICS\* COBOL

**CICSCBLLE**  
CICS統合化言語環境COBOL

**CICSMAP**  
CICSマップ

**CICSSQLCBL**  
CICS DB2 Query管理機能COBOL

**CLD** Cロケール記述

**CLLE** 制御言語の統合化言語環境\*

**CLP** 制御言語

**CLP38** システム/38制御言語

**CMD** コマンド

**CMD38**  
システム/38コマンド

**CPP** C++

**DSPF** 表示装置ファイル

**DSPF36**  
システム/36表示装置ファイル

**DSPF38**  
システム/38表示装置ファイル

**FTN** FORTRAN/400

**ICFF** システム間通信機能ファイル

**LF** 論理ファイル

**LF38** システム/38論理ファイル

**MENU**  
UIMメニュー

**MNU** メニュー

**MNUCMD**  
メニュー・コマンド

**MNUDDS**  
メニュー・データ記述仕様

**MNU36**  
システム/36メニュー

**MSGF36**  
システム/36メッセージ・ファイル

**OCL36**  
システム/36操作員制御言語

**PAS** PASCAL

**PF** 物理ファイル

**PF38** システム/38物理ファイル

**PLI** PL/I

**PLI38** システム/38 PL/I

**PNLGRP**  
パネル・グループ

**PRTF** 印刷装置ファイル

**PRTF38**  
システム/38印刷装置ファイル

**QRY38**  
S/38 QUERY

**REXX** 再構造化拡張実行プログラム言語

**RMC** RM/COBOL-85\*\*

**RPG** RPG/400

**RPGLE**  
統合化言語環境RPG

**RPG36**  
システム/36 RPG

**RPG38**  
システム/38 RPG

**RPT** RPG報告書簡易作成機能

**RPT36** システム/36 RPG報告書簡易作成機能

**RPT38** システム/38 RPG報告書簡易作成機能

**SPADCT**  
スペル援助辞書

**SQLC** DB2 Query管理機能C

**SQLCPP**  
DB2 Query管理機能C++

**SQLCBL**  
DB2 Query管理機能COBOL

**SQLCBLLE**  
DB2 Query管理機能統合化言語環境COBOL

**SQLCLE**  
DB2 Query管理機能C統合化言語環境

**SQLFTN**  
DB2 Query管理機能FORTRAN

**SQLPLI**  
DB2 Query管理機能PL/I

**SQLRPG**  
DB2 Query管理機能RPG

## SQLRPGLE

DB2 Query管理機能統合化言語環境RPG

**TBL** テーブル

**TXT** テキスト

### 総称メンバー・タイプ

処理したいファイル・メンバーの総称メンバー・タイプを指定してください。

総称メンバー・タイプは次の代表的な形式の1つとすることができます。

**RPG\*** メンバー・タイプが文字RPGで始まるすべてのメンバーのリストを表示します。たとえば、RPG、RPG36、およびRPG38など。

**\*C** メンバー・タイプが文字Cで終わるすべてのメンバーのリストを表示します。たとえば、CおよびSQLCなど。

**\*I\*** メンバー・タイプの中に文字Iが入っているすべてのメンバーのリストを表示します。たとえば、ICFF、PLI、PLI38、およびSQLPLIなど。

**R\*36** メンバー・タイプが文字Rで始まり、文字36で終わるすべてのメンバーのリストを表示します。たとえば、RPG36およびRPT36などです。

**"A"** 名前が英字Aで始まり、引用符付きであるメンバー・タイプのすべてのメンバーを表示します。たとえば、"A"、"AB"、"AD"など。

**\*\*ALL**

メンバー・タイプが文字ALLで終わるすべてのメンバーのリストを表示します。たとえば、ALL、BALL、またはTESTALLなど。\*ALLはすべてのメンバー・タイプのメンバーのリストを表示する特殊値として定義されているので、この場合には、2個のアスタリスクが必要です。

上

---

## 例

### 例1: ファイルにあるすべてのメンバーの処理

```
WRKMGRPDM FILE(*PRV) MBR('*ALL')
```

このコマンドを使用すると、前のWRKMGRPDMセッションで処理したものと同一ファイルにあるすべてのメンバーを処理することができます。

### 例2: 1つのタイプのメンバーの処理

```
WRKMGRPDM FILE(*LIBL/MYSRCFILE) MBRTYPE('CLP')
```

このコマンドを使用すると、CLP(CONTROL LANGUAGE PROGRAM)のメンバー・タイプを持つソース・ファイルMYSRCFILEのすべてのメンバーを処理することができます。ソース・ファイルはジョブ・ライブラリー・リストを使用して配置されます。

### 例3: 総称名でメンバーを処理

```
WRKMGRPDM FILE(MYLIB/MYSRCFILE) MBR('PAY*')
```

このコマンドを使用するとMYLIBライブラリーのソース・ファイル、MYSRCFILEにあるメンバー名が**PAY**で始まるすべてのメンバーを処理することができます。

上

---

## エラー・メッセージ

### \*ESCAPE メッセージ

#### **PDM0055**

&1コマンドの処理中にエラーが起こった。

上





## PDM使用のオブジェクトの処理 (WRKOBJPDM)

実行可能場所: 対話環境 (\*INTERACT \*IPGM \*IREXX \*EXEC)  
スレッド・セーフ: いいえ

パラメーター  
例  
エラー・メッセージ

PDM使用のオブジェクトの処理 (WRKOBJPDM)コマンドを使用すると1つのライブラリー内にあるオブジェクトを処理することができます。このコマンドを使用すると、プログラム開発管理機能(PDM)メニューおよび処理オブジェクト指定画面をバイパスすることができます。

上

### パラメーター

キーワード	記述	選択項目	ノーツ
LIB	ライブラリー	名前, <u>*PRV</u> , *CURLIB	オプション, 定位置 1
OBJ	オブジェクト	文字値, <u>*ALL</u> , *PRV	オプション, 定位置 2
OBJTYPE	オブジェクト・タイプ	<u>*ALL</u> , *PRV, *ALRTBL, *AUTL, *BNDDIR, *CFGL, *CHTFMT, *CLD, *CLS, *CMD, *CNNL, *COSD, *CRG, *CRQD, *CSI, *CSPMAP, *CSPTBL, *CTLD, *DEVD, *DOC, *DTAARA, *DTADCT, *DTAQ, *EDTD, *EXITRG, *FCT, *FILE, *FNTRSC, *FNTTBL, *FORMDF, *FTR, *GSS, *IGCDCT, *IGCSRT, *IGCTBL, *IMGCLG, *IPXD, *JOBQ, *JOBQ, *JOBSCD, *JRN, *JRNRCV, *LIB, *LIND, *LOCALE, *M36, *M36CFG, *MEDDFN, *MENU, *MGTCOL, *MODD, *MODULE, *MSGF, *MSGQ, *NODGRP, *NODL, *NTBD, *NWID, *NWSD, *NWSCFG, *OUTQ, *OVL, *PAGDFN, *PAGSEG, *PDFMAP, *PDG, *PGM, *PNLGRP, *PRDAVL, *PRDDFN, *PRDLOD, *PSFCFG, *QMFORM, *QMORY, *QRYDFN, *RCT, *SBSD, *SCHIDX, *SPADCT, *SQLPKG, *SQLUDT, *SRVPGM, *SSND, *SVRSTG, *S36, *TBL, *TIMZON, *USRIDX, *USRPRF, *USRQ, *USRSPC, *VLDL, *WSCST	オプション, 定位置 3

キーワード	記述	選択項目	ノート
<b>OBJATR</b>	オブジェクト属性	文字値, <b>*ALL</b> , <b>*PRV</b> , <b>*BLANK</b> , <b>BAS</b> , <b>BAS36</b> , <b>BAS38</b> , <b>BSCF38</b> , <b>C</b> , <b>CBL</b> , <b>CBL36</b> , <b>CBL38</b> , <b>CLE</b> , <b>CLP</b> , <b>CLP38</b> , <b>CMD</b> , <b>CMD38</b> , <b>CMNF38</b> , <b>CSPA</b> , <b>DDMF</b> , <b>DFU</b> , <b>DFUEXEC</b> , <b>DFUNOTEXC</b> , <b>DKTF</b> , <b>DSPF</b> , <b>DSPF36</b> , <b>DSPF38</b> , <b>FTN</b> , <b>ICFF</b> , <b>LF</b> , <b>LF38</b> , <b>MXDF38</b> , <b>PAS</b> , <b>'PF-DTA'</b> , <b>'PF-SRC'</b> , <b>PF38</b> , <b>PLI</b> , <b>PLI38</b> , <b>PRTF</b> , <b>PRTF38</b> , <b>QRY38</b> , <b>RMC</b> , <b>RPG</b> , <b>RPG36</b> , <b>RPG38</b> , <b>RPT</b> , <b>RPT36</b> , <b>RPT38</b> , <b>SAVF</b> , <b>SPADCT</b> , <b>SQLC</b> , <b>SQLCBL</b> , <b>SQLCLE</b> , <b>SQLFTN</b> , <b>SQLPLI</b> , <b>SQLRPG</b> , <b>TAPF</b> , <b>TBL</b>	オプション

上

## ライブラリー (LIB)

処理したいオブジェクトが入っているライブラリーを指定します。

**\*PRV** 前のWRKOBJPDMセッションで使用したライブラリーが使用されます。

**\*CURLIB**

ジョブの現行ライブラリーが使用されます。現行ライブラリーが定義されていない場合には、QGPLライブラリーが使用されます。

**名前** 処理したいオブジェクトが入っているライブラリーの名前を指定してください。

上

## オブジェクト (OBJ)

処理したいオブジェクトを指定します。このパラメーターを使用して、指定したライブラリー中のすべてのオブジェクトまたはオブジェクトのサブセットを処理することができます。

**\*ALL** **ライブラリー (LIB)**パラメーターに指定したライブラリーにあるすべてのオブジェクトを処理します。**オブジェクト・タイプ (OBJTYPE)**および**オブジェクト属性 (OBJATR)**パラメーターに**\*ALL**以外の値を指定することによってオブジェクトのリストをサブセットすることができます。

**\*PRV** 同じオブジェクトまたは前のWRKOBJPDMセッションで処理したオブジェクトを処理します。

**総称名** 処理するオブジェクトの総称名を指定します。

総称名は次の形式の1つとすることができます。

**ABC\*** 文字ABCで始まるすべてのオブジェクトのリストを表示します。たとえば、ABC, ABCD, またはABCTESTなど。

**\*ABC** 文字ABCで終るすべてのオブジェクトのリストを表示します。たとえば、ABC, DABC, またはTESTABCなど。

- \*B\*** 名前の中に文字Bが入っているすべてのオブジェクトのリストを表示します。たとえば、B, BALL, ABCDなど。
- A\*C** 文字Aで始まり、文字Cで終わるすべてのオブジェクトのリストを表示します。たとえば、AC, ABC, AZZZCなど。
- "A"** 名前が英字Aで始まり、引用符付きであるすべてのオブジェクトのリストを表示します。たとえば、"A", "AB", "AD"など。
- \*\*ALL**  
ALLで終わるすべてのオブジェクトのリストを表示します。たとえば、ALL, BALL,またはTESTALLなど。**\*ALL**はすべてのオブジェクトのリストを表示する特殊値として定義されているので、この場合には、2個のアスタリスクが必要です。

**名前** 処理するオブジェクト名(単数複数可)を指定します。同じ名前で異なるオブジェクト・タイプのオブジェクトがある場合には複数のオブジェクトがリストされ、OBJTYPEパラメーターには**\*ALL**が指定されます。

上

---

## オブジェクト・タイプ (OBJTYPE)

処理したいオブジェクトのオブジェクト・タイプを指定します。このパラメーターを使用して、すべてのオブジェクト・タイプまたはオブジェクトのサブセットを処理することができます。

**\*ALL** **オブジェクト (OBJ)**パラメーターに指定したオブジェクト名に一致するオブジェクトを処理します。

**\*PRV** 前のWRKOBJPDMセッションで処理したものと同一オブジェクト・タイプのオブジェクトを処理します。

### オブジェクト・タイプ

処理したいオブジェクトのシステム・オブジェクト・タイプを指定します。

次のオブジェクト・タイプから選択してください。

**\*ALRTBL**

警報テーブル

**\*AUTL**

権限リスト

**\*BNDDIR**

バインド・ディレクトリー

**\*CFGL**

構成リスト

**\*CHTFMT**

図表様式

**\*CLD** Cロケール記述

**\*CLS** クラス

**\*CMD** コマンド

**\*CNL**  
接続リスト

**\*COSD**  
サービス・クラスの定義

**\*CRG** クラスター資源グループ

**\*CRQD**  
変更要求記述

**\*CSI** 通信側情報

**\*CSPMAP**  
システム共通プロダクトのマップ

**\*CSPTBL**  
システム共通プロダクトのテーブル

**\*CTLD**  
制御記述

**\*DEV D**  
装置記述

**\*DOC** 文書

**\*DTAARA**  
データ域

**\*DTADCT**  
データ・ディクショナリー

**\*DTAQ**  
データ待ち行列

**\*EDTD**  
編集記述

**\*EXITRG**  
出口登録

**\*FCT** 用紙制御テーブル

**\*FILE** ファイル

**\*FNTRSC**  
フォント資源

**\*FNTTBL**  
フォント・マッピング・テーブル

**\*FORMDF**  
用紙定義

**\*FTR** フィルター

**\*GSS** グラフィックス記号セット

**\*IGCDCT**  
漢字辞書

**\*IGCSRT**  
漢字分類

**\*IGCTBL**  
漢字テーブル

**\*IMGCLG**  
光ディスク・イメージ・カタログ

**\*IPXD** インターネット・パケット交換記述

**\*JOB**  
ジョブ記述

**\*JOBQ**  
ジョブ待ち行列

**\*JOBSCD**  
ジョブ・スケジュール

**\*JRN** ジャーナル

**\*JRNRCV**  
ジャーナル・レシーバー

**\*LIB** ライブラリー

**\*LIND** 回線記述

**\*LOCALE**  
ロケール空間

**\*M36** アドバンスト36マシン

**\*M36CFG**  
アドバンスト36マシン構成

**\*MEDDFN**  
媒体定義

**\*MENU**  
メニュー

**\*MGTCOL**  
管理収集

**\*MODD**  
モード記述

**\*MODULE**  
モジュール

**\*MSGF**  
メッセージ・ファイル

**\*MSGQ**  
メッセージ待ち行列

**\*NODGRP**  
ノード・グループ

**\*NODL**  
ノード・リスト

**\*NTBD**  
NETBIOS構成データ

**\*NWID**  
ネットワーク・インターフェース記述

**\*NWSCFG**  
ネットワーク・サーバー構成

**\*NWSD**  
ネットワーク・サーバー記述

**\*OUTQ**  
出力

**\*OVL** オーバーレイ

**\*PAGDFN**  
ページ定義

**\*PAGSEG**  
ページ・セグメント

**\*PDFMAP**  
PDFマップ

**\*PDG** 印刷管理機能CPI: 論理印刷記述子

**\*PGM** プログラム

**\*PNLGRP**  
パネル・グループ

**\*PRDAVL**  
プロダクト使用可能性

**\*PRDDFN**  
プロダクト定義

**\*PRDLOD**  
プロダクト・ロード

**\*PSFCFG**  
印刷サービス機能構成

**\*QMFORM**  
QUERY管理機能書式

**\*QMQRy**  
QUERY管理機能プログラム

**\*QRYDFN**  
QUERY定義

**\*RCT** リモート制御テーブル

**\*SBSD** サブシステム記述

<b>*SCHIDX</b>	検索索引
<b>*SPADCT</b>	スペル援助辞書
<b>*SQLPKG</b>	DB2 Query管理機能パッケージ
<b>*SQLUDT</b>	SQLユーザー定義タイプ
<b>*SRVPGM</b>	サービス・プログラム
<b>*SSND</b>	セッション記述
<b>*SVRSTG</b>	サーバー記憶域スペース
<b>*S36</b>	システム/36マシン記述
<b>*TBL</b>	テーブル
<b>*TIMZON</b>	タイムゾーン記述
<b>*USRIDX</b>	ユーザー見出し
<b>*USRPRF</b>	ユーザー・プロファイル
<b>*USRQ</b>	ユーザー待ち行列
<b>*USRSPC</b>	ユーザー・スペース
<b>*VLDL</b>	妥当性検査リスト
<b>*WSCST</b>	ワークステーション・ユーザー・カスタマイズ

上

---

## オブジェクト属性 (OBJATR)

処理したいオブジェクトのオブジェクト属性を指定します。このパラメーターを使用して、**オブジェクト (OBJ)**および**オブジェクト・タイプ (OBJTYPE)**パラメーターに指定したオブジェクト名およびオブジェクト・タイプに一致するオブジェクトのサブセットを処理することができます。

**\*ALL** OBJおよびOBJTYPEパラメーターに指定した値と一致するすべてのオブジェクトを処理します。



**\*PRV** 前のWRKOBJPDMセッションで処理したものと同じオブジェクト属性を持つオブジェクトを処理します。

**\*BLANK**

属性値なしのすべてのオブジェクトを処理します。

**値** 処理したいオブジェクトのオブジェクト属性を指定します。オブジェクト属性を指定する場合には、オブジェクト・タイプを指定する必要はありません。

次のオブジェクト属性値から選択してください。

**BAS** BASIC

**BAS36** システム/36 BASIC

**BAS38** システム/38 BASIC

**BSCF38**

システム/38 2進データ同期通信ファイル

**C** C言語

**CBL** COBOL

**CBLLE**

統合化言語環境COBOL

**CBL36**

システム/36 COBOL

**CBL38**

システム/38 COBOL

**CLLE** 制御言語の統合化言語環境

**CLP** 制御言語

**CLP38** システム/38制御言語

**CMD** コマンド

**CMD38**

システム/38コマンド

**CMNF38**

通信ファイル

**CSPAE**

システム共通プロダクト適用業務の実行

**DDMF** 分散データ管理機能(DDM)

**DFU** データ・ファイル・ユーティリティ

**DFUEXEC**

データ・ファイル・ユーティリティ実行可能ファイル

**DFUNOTEXC**

データ・ファイル・ユーティリティ実行不能ファイル

**DKTF** ディスケット・ファイル

**DSPF** 表示装置ファイル

**DSPF36**  
システム/36表示装置ファイル

**DSPF38**  
システム/38表示装置ファイル

**FTN** FORTTRAN/400

**ICFF** システム間通信機能ファイル

**LF** 論理ファイル

**LF38** システム/38論理ファイル

**MXDF38**  
システム/38混合ファイル

**PAS** PASCAL

**PF-DTA**  
物理ファイルーデータ

**PF-SRC**  
物理ファイルーソース

**PF38** システム/38物理ファイル

**PLI** PL/I

**PLI38** システム/38 PL/I

**PRTF** 印刷装置ファイル

**PRTF38**  
システム/38印刷装置ファイル

**QRY38**  
システム/38 QUERY

**RMC** RM/COBOL-85\*\*

**RPG** RPG/400

**RPGLE**  
統合化言語環境RPG

**RPG36**  
システム/36 RPG

**RPG38**  
システム/38 RPG

**RPT** RPG報告書簡易作成機能

**RPT36** システム/36 RPG報告書簡易作成機能

**RPT38** システム/38 RPG報告書簡易作成機能

**SAVF** 保管ファイル

**SPADCT**  
スペル援助辞書

**SQLC** DB2 Query管理機能C

**SQLCBL**

DB2 Query管理機能COBOL

**SQLCBLLE**

DB2 Query管理機能統合化言語環境COBOL

**SQLFTN**

DB2 Query管理機能FORTRAN

**SQLPLI**

DB2 Query管理機能PL/I

**SQLRPG**

DB2 Query管理機能RPG

**SQLRPGLE**

DB2 Query管理機能統合化言語環境RPG

**TAPF** テープ・ファイル

**TBL** テーブル

**総称値** 処理するオブジェクトの総称属性を指定します。

総称値は、次の形式の1つとすることができます。

**RPG\*** 属性タイプが文字RPGで始まっているすべてのオブジェクトのリストを表示します。たとえば、RPG、RPG36、およびRPG38など。

**\*C** 属性タイプが文字Cで終わっているすべてのオブジェクトのリストを表示します。たとえば、CおよびSQLCなど。

**\*I\*** 属性タイプの中に文字Iが入っているすべてのオブジェクトのリストを表示します。たとえば、ICFF、PLI、PLI38、およびSQLPLIなど。

**P\*38** 属性タイプが文字Pで始まり、文字38で終わるすべてのオブジェクトのリストを表示します。たとえば、PLI38およびPRTF38など。

**"A"** 属性タイプが引用符内に文字aを含むすべてのオブジェクトのリストを表示します。たとえば、"A"、"AB"、"AD"など。

**\*\*ALL**

属性タイプがALLで終わるすべてのオブジェクトのリストを表示します。たとえば、ALL、BALL、またはTESTALLなど。**\*ALL**はオブジェクト属性に関係なくオブジェクトを表示する特殊値として定義されているので、この場合には、2個のアスタリスクが必要です。

上

---

## 例

### 例1: ライブラリーにあるすべてのオブジェクトの処理

```
WRKOBJPDM  LIB(MYLIB)  OBJ('*ALL')
```

このコマンドを使用するとMYLIBライブラリーにあるすべてのオブジェクトを処理することができます。

### 例 2: 1つのタイプのオブジェクトを処理する

```
WRKOBJPDM  LIB(*PRV)  OBJ('*ALL')  OBJTYPE(*CMD)
```

このコマンドを使用すると、前のWRKOBJPDMセッションで処理したものと同一ライブラリーにあるすべてのコマンド(\*CMD)オブジェクトを処理することができます。

上

---

## エラー・メッセージ

### \*ESCAPE メッセージ

#### **PDM0055**

&1コマンドの処理中にエラーが起こった。

上



---

## 付録. 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  
日本アイ・ビー・エム株式会社  
法務・知的財産  
知的財産権ライセンス渉外

**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を提供し、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なります。

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、随時、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。

本書において IBM 以外の Web サイトに言及している場合がありますが、便宜のため記載しただけであり、決してそれらの Web サイトを推奨するものではありません。それらの Web サイトにある資料は、この IBM 製品の資料の一部ではありません。それらの Web サイトは、お客様の責任でご使用ください。

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その他のプログラム (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  
Software Interoperability Coordinator, Department YBWA  
3605 Highway 52 N  
Rochester, MN 55901  
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、IBM 機械コードのご使用条件、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。

この文書に含まれるいかなるパフォーマンス・データも、管理環境下で決定されたものです。そのため、他の操作環境で得られた結果は、異なる可能性があります。一部の測定が、開発レベルのシステムで行われた可能性があります。その測定値が、一般に利用可能なシステムのものと同じである保証はありません。さらに、一部の測定値が、推定値である可能性があります。実際の結果は、異なる可能性があります。お客様は、お客様の特定の環境に適したデータを確かめる必要があります。

IBM 以外の製品に関する情報は、その製品の供給者、出版物、もしくはその他の公に利用可能なソースから入手したものです。IBM は、それらの製品のテストは行っておりません。したがって、他社製品に関する実行性、互換性、またはその他の要求については確証できません。IBM 以外の製品の性能に関する質問は、それらの製品の供給者にお願いします。

IBM の将来の方向または意向に関する記述については、予告なしに変更または撤回される場合があります、単に目標を示しているものです。

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。

著作権使用許諾:

本書には、様々なオペレーティング・プラットフォームでのプログラミング手法を例示するサンプル・アプリケーション・プログラムがソース言語で掲載されています。お客様は、サンプル・プログラムが書かれているオペレーティング・プラットフォームのアプリケーション・プログラミング・インターフェースに準拠したアプリケーション・プログラムの開発、使用、販売、配布を目的として、いかなる形式においても、IBM に対価を支払うことなくこれを複製し、改変し、配布することができます。このサンプル・プログラムは、あらゆる条件下における完全なテストを経ていません。従って IBM は、これらのサンプル・プログラムについて信頼性、利便性もしくは機能性があることをほのめかしたり、保証することはできません。

それぞれの複製物、サンプル・プログラムのいかなる部分、またはすべての派生的創作物にも、次のように、著作権表示を入れていただく必要があります。

© (お客様の会社名) (西暦年). このコードの一部は、IBM Corp. のサンプル・プログラムから取られています。 © Copyright IBM Corp. \_年を入れる\_. All rights reserved.

この情報をソフトコピーでご覧になっている場合は、写真やカラーの図表は表示されない場合があります。

---

## プログラミング・インターフェース情報

本書 (WebSphere Development Studio for System i コマンド) には、プログラムを作成するユーザーが IBM WebSphere Development Studio for System i のサービスを使用するためのプログラミング・インターフェースが記述されています。

---

## 商標

以下は、International Business Machines Corporation の米国およびその他の国における商標です。

Advanced Function Printing  
AFP  
AS/400  
CICS  
COBOL/400  
C/400  
DataPropagator  
DB2  
Distributed Relational Database Architecture  
Lotus Domino  
DRDA  
IBM  
Infoprint  
InfoWindow  
i5/OS  
iSeries  
Integrated Language Environment  
Lotus  
LPDA  
OfficeVision  
Print Services Facility  
RPG/400  
System i  
System x  
SystemView  
System/36  
TCS  
Tivoli  
WebSphere  
z/OS

Adobe、Adobe ロゴ、PostScript、および PostScript ロゴは、米国およびその他の国における Adobe Systems Incorporated の商標または登録商標です。

Intel、Intel (ロゴ)、Intel Inside、Intel Inside (ロゴ)、Intel Centrino、Intel Centrino (ロゴ)、Celeron、Intel Xeon、Intel SpeedStep、Itanium、Pentium は、Intel Corporation または子会社の米国およびその他の国における商標または登録商標です。

Linux は、Linus Torvalds の米国およびその他の国における商標です。

Microsoft、Windows、Windows NT および Windows ロゴは、Microsoft Corporation の米国およびその他の国における商標です。

Java およびすべての Java 関連の商標およびロゴは、Sun Microsystems, Inc. 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。

---

## 使用条件

これらの資料は、以下の条件に同意していただける場合に限りご使用いただけます。

**個人使用:** これらの資料は、すべての著作権表示その他の所有権表示をしていただくことを条件に、非商業的な個人による使用目的に限り複製することができます。ただし、IBM の明示的な承諾をえずに、これらの資料またはその一部について、二次的著作物を作成したり、配布（頒布、送信を含む）または表示（上映を含む）することはできません。

**商業的使用:** これらの資料は、すべての著作権表示その他の所有権表示をしていただくことを条件に、お客様の企業内に限り、複製、配布、および表示することができます。ただし、IBM の明示的な承諾をえずにこれらの資料の二次的著作物を作成したり、お客様の企業外で資料またはその一部を複製、配布、または表示することはできません。

ここで明示的に許可されているもの以外に、資料や資料内に含まれる情報、データ、ソフトウェア、またはその他の知的所有権に対するいかなる許可、ライセンス、または権利を明示的にも黙示的にも付与するものではありません。

資料の使用が IBM の利益を損なうと判断された場合や、上記の条件が適切に守られていないと判断された場合、IBM はいつでも自らの判断により、ここで与えた許可を撤回できるものとさせていただきます。

お客様がこの情報をダウンロード、輸出、または再輸出する際には、米国のすべての輸出入関連法規を含む、すべての関連法規を遵守するものとします。

IBM は、これらの資料の内容についていかなる保証もしません。これらの資料は、特定物として現存するままの状態を提供され、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なしで提供されます。

---

## コードに関するライセンス情報および特記事項

IBM は、お客様に、すべてのプログラム・コードのサンプルを使用することができる非独占的な著作使用权を許諾します。お客様は、このサンプル・コードから、お客様独自の特別のニーズに合わせた類似のプログラムを作成することができます。

強行法規で除外を禁止されている場合を除き、IBM、そのプログラム開発者、および供給者は「プログラム」および「プログラム」に対する技術的サポートがある場合にはその技術的サポートについて、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。

いかなる場合においても、IBM および IBM のサプライヤーならびに IBM ビジネス・パートナーは、その予見の有無を問わず発生した以下のものについて賠償責任を負いません。

1. データの喪失、または損傷。
2. 直接損害、特別損害、付随的損害、間接損害、または経済上の結果的損害
3. 逸失した利益、ビジネス上の収益、あるいは節約すべきし費用

国または地域によっては、法律の強行規定により、上記の責任の制限が適用されない場合があります。





Printed in Japan