CA-MetaCOBOL™ +

Panel Definition Facility
Command Reference Manual

Release 1.1

R203M+11DRP

-- PROPRIETARY AND CONFIDENTIAL INFORMATION --

Release 1.1, January, 1992

Contents

Figures

.

1. About this Manual

This manual provides information about the CA-MetaCOBOL+ Panel Definition Facility commands.

1.1 Purpose

This document provides general information on command defaults, abbreviations, and delimiter strings. Also included is a detailed description of all PDF/CICS commands. Familiarity with BMS maps is assumed. For general information on PDF/CICS usage, refer to the *Panel Definition Facility User Guide.*

1.2 Organization

This manual is organized as follows:

Chapter	Description
1	Introduces the contents and organization of this manual. Additional reference materials used with this manual are also listed.
2	Describes command defaults, abbreviations, and string delimiters.
3	Describes in detail the CA-MetaCOBOL+ PDF/CICS commands.
4	Describes in detail the CA-MetaCOBOL+ PDF/CICS editing commands.

1.3 Publications

In addition to this manual, the following publications are supplied with CA-MetaCOBOL+:

Title	Contents
Introduction to CA-MetaCOBOL+	Introduces the CA-MetaCOBOL+ Work Bench, Structured Programming Facility, Quality Assurance Facility, CA DATACOM/DB Facility, Macro Facility, Panel Definition Facility, the Online Programming Language and the Chart Writer Language.
Installation Guide - MVS	Explains how to install CA-MetaCOBOL+ in the MVS environment.
CA ACTIVATOR Installation Supplement - MVS	Explains how to install CA-MetaCOBOL+ in the MVS environment using CA ACTIVATOR
Installation Guide - VSE	Explains how to install CA-MetaCOBOL+ in the VSE environment.
Installation Guide - CMS	Explains how to install CA-MetaCOBOL+ in the VM environment.
User Guide	Explains how to customize, get started, and use CA-MetaCOBOL+. Includes information on keyword expansion, the CA-MetaCOBOL+ translator, and CA macro sets and programs.
Structured Programming Guide	Introduces the Structured Programming Facility. Includes information on creating, testing, and maintaining structured programs.
Macro Facility Tutorial	Introduces the Macro Facility. Includes information on writing basic macros, model programming, macro writing techniques, and debugging.
Macro Facility Reference	Includes detailed information on the program flow of the CA-MetaCOBOL+ macro translator, macro format, definition of comments, macro nesting, macro prototypes, symbolic words, and model programming.

Title	Contents
Quality Assurance Guide	Introduces the Quality Assurance Facility. Includes all the necessary information to perform quality assurance testing on COBOL source programs.
Program Development Guide CA DATACOM/DB	Includes all the information necessary to develop programs that make full use of the functions and features of the CA DATACOM/DB environment.
Program Development Reference CA DATACOM/DB	Contains all CA DATACOM/DB Facility constructs and statements.
Panel Definition Facility User Guide	Includes all the information necessary to create, edit, duplicate, rename, delete, index, and print panel definitions and members. Also describes how to generate BMS source.
Online Programming Language Reference	Contains all Online Programming Language statements.
Online Programming Guide	Describes the basic OPL functions and describes their use through examples.

All manuals are updated as required. Instructions accompany each update package.

1.4 Related Publications

The following publication relates to CA-MetaCOBOL+ PDF/CICS and is supplied by Computer Associates:

Title
CAIPC Messages and Codes

The following publications relate to CA-MetaCOBOL+ PDF/CICS and are supplied by IBM.

Title
CICS Application Programmer's Guide
VS COBOL II Application Programming Guide
VS COBOL II Application Programming Reference
OS/VS COBOL Compiler and Library Programmer's Guide

1.5 Notation Conventions

This manual uses the following rules and special characters in syntax illustrations.

Enter the following exactly as they appear in command descriptions:

UPPERCASE	Identifies commands, keywords, and keyword values which must be entered exactly as shown or replaced by an authorized abbreviation.
symbols	All special characters such as parentheses and quotation marks (but not ellipses, brackets, and braces) must be entered as shown.

The following notations clarify command syntax; do not enter them as shown.

lower case *italics*	Represent a value you must supply.
Brackets, [ ]	Identify optional keywords or clauses, or a group of options from which, if included at all, a choice of one must be made.
Braces, {}	Indicate that one of the keywords or clauses must be entered.
Underlining	Indicates a CA IDEAL/PC default that cannot be changed with a SET command, and therefore need not be specified.
Ellipses, ...	Indicate that the preceding word or clause can be repeated.

2. Preliminary Concepts

The CA-MetaCOBOL+ Panel Definition Facility (PDF/CICS) is a CICS "panel painter" and BMS map generator. "Panel painting" is a term that describes the function of a user at a terminal, interactively laying out, or "painting", a visual description of a panel. This panel can be used by a programming system to send and receive images to and from a terminal. The result of painting the panel and providing information about the panel is called a panel definition. With a panel definition as input, PDF/CICS can generate CICS BMS map source to be used as input to the assembler. The resulting BMS map source is stored in a PDF/CICS member. Panel definitions and members can be displayed, edited, printed, duplicated, renamed, and deleted with PDF/CICS.

The terms panel and map, and panelset and mapset, are used interchangeably throughout this manual. A mapset consists of one map.

2.1 PDF/CICS Defaults

When this document makes reference to defaults, in most cases no actual default value is mentioned. This is because defaults used by PDF/CICS are established in a number of ways, and are often specific to the site, or even to the individual user.

PDF/CICS is delivered with default values for all options. Some of these defaults can be changed and some cannot.

Defaults That Cannot Be Changed

Some defaults are for certain choices within PDF/CICS commands and fill-ins, and cannot be changed. This type of default is underlined in the command syntax examples in this manual. An example of this type of default is the command,

```
                                  [ IDENTIFICATION ]
                                  | SUMMARY      |
EDIT [SITE] PANEL name            | PARAMETER    |
                                  | PICTURE      |
                                  | LAYOUT       |
                                  [              ]
```

which defaults to the layout component of the panel definition.

Defaults That Can Be Changed for the Entire Site

Defaults can be established for the entire site. Some site defaults can be changed only during installation, and can be reset later only by rerunning installation jobs. An example of a default in this category is a default library name.

Other site defaults can be reset by using either the SET SITE commands or the fill-ins provided for setting certain session control and print options. Any default that is reset with either a SET SITE command or a site options fill-in becomes a new site default, and remains in effect unless it is reset with another SET SITE command or fill-in.

The SET SITE commands are described in Chapter 3 of this manual.

Defaults That Can Be Set for an Individual Session

Finally, most of the defaults that can be set for the site can also be set by each PDF/CICS user for an individual session using SET commands or session options fill-ins.

A default set with a SET command or session options fill-in is changed only for the user who issued the command, and remains in effect only for the current session. The user has the option of storing SET commands in a member called BMS#ON. This causes the commands to be executed automatically each time the user signs on, and works as if that user's default settings had been changed permanently.

References to defaults in this document refer to whatever default is currently in effect for any given option. The actual default for any given option, for any given user, in any given session, at any given site, depends on what choices for setting defaults have been made.

The default option settings for the current session can be displayed or printed using the following commands:

```
DISPLAY SESSION OPTIONS
PRINT SESSION OPTIONS
```

The SET commands are described in chapter 3 of this manual.

2.2 Use of Abbreviations in PDF/CICS Commands

Standard Abbreviations

The standard abbreviation for a command or command option is the first three characters of the word. For example, the standard abbreviation for the GENERATE command is GEN.

Alternate Abbreviations

For some commands and options, PDF/CICS accepts alternate abbreviations in addition to the standard three-character abbreviations. For example, in addition to BAC, the abbreviations BACK and BWD are accepted for BACKWARD:

Command/Option	Exception	Alternate
BACKWARD		BACK, BWD

Abbreviation Exceptions

There are a number of exceptions to the standard first-three-character abbreviation. These exceptions are abbreviations for command words whose first three characters are not unique and, therefore, would conflict with an abbreviation for another command. Other abbreviations (or no abbreviation at all) are used.

For example, the COMBINE command does not have a three-character abbreviation, and the COMMAND option accepts the abbreviation CMD only:

Command/Option	Exception	Alternate
COMBINE	COMBINE	
COMMAND	CMD	

Summary of Command Abbreviations

Because most abbreviations are standard, they are not shown in the syntax illustrations in this manual. The following list shows command abbreviations that are either exceptions to the standard abbreviation or are alternatives available in addition to the standard abbreviation. Any commands or options not in this list accept only the standard three-character abbreviation.

PDF/CICS Command Abbreviations

Command/Option	Exception	Alternate
BACKWARD		BACK, BWD
BLANKFILL		BLANK, BLANKS
CHANGE		C
COLUMNS		COLUMN
COMBINE	COMBINE	
COMMAND	CMD	
COMMENT	COMMENT	
DELIMITER	DLM	DELIM
DESCRIPTION	DESC	
DESTINATION	DEST	
DISPLAY		D
DISPOSITION	DISP	
EDIT		E
EXCLUDE		EXCL, EXCLUDING
EXECUTE		EXEC
FIND		F
FIELD		FLD
FORWARD		FWD
HELP	HELP	?
IDENTIFICATION		IDENT
INCLUDE		INCL, INCLUDING
LEFT (in SET EDIT)		L

PDF/CICS Command Abbreviations (continued)

Command/Option	Exception	Alternate
LINES		LINE
NAMES		NAME
NO		N
NULLFILL		NULLS, NULLFIL
OFF	OFF	
ON	ON	
OPTIONS		OPTION
PAGE		PAGES
PANEL		PNL, MAP
PARAMETERS		PARAMETER, PARM, PARMS
PREVIOUS		PREV
REPEAT	REPEAT	
RESHOW	RESHOW	
RIGHT (in SET EDIT)		R
ROLLBACK		ROLL
SCREEN	SCREEN	
USER		USR
YES		Y

2.3 Use of String Delimiters in PDF/CICS Commands

The character used to delimit strings in PDF/CICS commands is shown as either an apostrophe (') or slash (/). Any special character can be used as a string delimiter in a command except:

- The currently defined command delimiter; semicolon (;) is the installed default
- The currently defined comment character; colon (:) is the installed default
- Asterisk (*)
- At sign (@)
- Pound sign (#)
- Dollar sign ($)

To display the currently defined command delimiter character and comment character use the DISPLAY SESSION OPTIONS command.

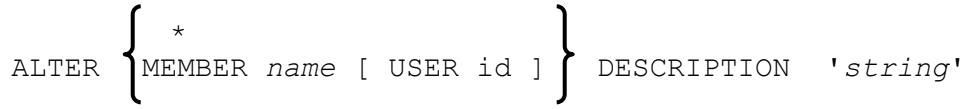
3. PDF/CICS Commands

This chapter describes the PDF/CICS commands and command syntax. Refer to the Chapter 1 section entitled 'Notation Conventions' for a description of the syntax notation conventions.

ALTER MEMBER Command

The ALTER MEMBER command changes the description of a member. Use the DIS INDEX MEMBER command to display a member's description.

Format:



*****
An asterisk may be substituted for MEMBER *name* to alter the current member's description.

name
The 1 to 8 character member name.

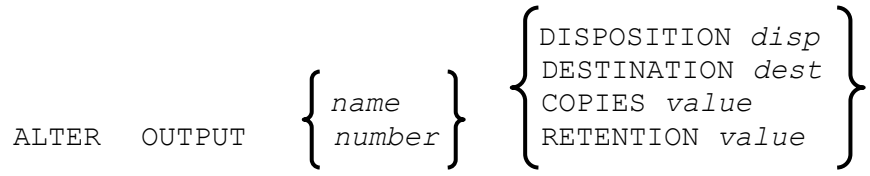
USER id
The 1 to 3 character user ID. This is required when altering a member belonging to another user. The USER id corresponds to the one used at signon to PDF/CICS. If you use the express signon, the USER id will be taken from CICS TCTTE01.

DESCRIPTION '*string*'
A new 1 to 24 character description of the member. For description of valid string delimiters, see Section 2.3.

ALTER OUTPUT Command

This command changes the output disposition, destination, number of copies, or for outputs residing in the output library, the retention period.

Format:



name
The 1 to 8 character name of the output to be altered.

number
The 1 to 4 digit output number.

DISPOSITION *disp*
Changes the disposition of the output to HOLD, RELEASE, or KEEP, as specified.

DESTINATION *dest*
Changes the destination of the output.

COPIES *value*
Changes the number of copies of output being requested. The value can be any number from one up to the site maximum.

Note: Under VSE, this clause is ignored for a system printer.

RETENTION *value*
Changes the number of days the output is retained. The value can be any number of days from one up to the site maximum.

ALTER OUTPUT DESTINATION Command

This command changes the disposition of all output with a specific printer destination or all output regardless of the printer destination.

Format:

ALL
The disposition is defined for all output regardless of the destination.

destname
The disposition is defined for the named printer destination.

disp
The disposition for all output printed at the named destination. The valid values are READY or HOLD.

ALTER OUTPUT DESTINATION HEADER Command

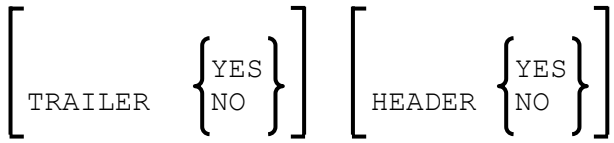
This command alters the default settings for the specified network printer. The settings include:

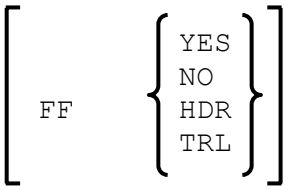
- Printing a header page
- Printing a trailer page
- Width of the print line
- Number of lines per page

These values override the actual printer settings.

Format:

```
ALTER OUTPUT DESTINATION destname[ WIDTH width ] [ LINES lines ]
                                 [ PERTASK nnn ]
```





destname
The name of the destination printer.

WIDTH *width*
The width of the print line. For network printers the value may be 80-250. Specify 0 to reset the width to the default value.

LINE *lines*
The number of lines to be printed per page. The default is 60 lines per page.

PERTASK *nnn*
The number of outputs allowed to print per print transaction. The value can be from 0 to 255.

HEADER
Defines whether a header page is printed. Specify YES to print a header page and NO to suppress the header page.

TRAILER
Defines whether a trailer page is printed. Specify YES to print a trailer page and NO to suppress the trailer page.

FF
Defines whether form feeds are generated for header and trailer pages at network printers.

YES
A form feed occurs before header page and after trailer page.

NO
No form feed occurs for either the header page or trailer page.

HDR
A form feed occurs for the header page, but not the trailer page.

TRL
A form feed occurs for the trailer page, but not the header page.

Notes:

- The LINES operand is only used when CA-MetaCOBOL+ is run under CA ROSCOE, otherwise, it is ignored. This value allows network printers to be shared by CA ROSCOE and other systems such as CICS.
- To reset the named network printer width to the default value, specify zero for the WIDTH.

ALTER SIGNON PANEL Command

The ALTER SIGNON PANEL command enables you to change the PDF/CICS signon panel greeting message. An image of the panel is displayed, enabling you to type over the old message.

Note: The signon panel greeting is changed for the entire site.

Format:

```
ALTER SIGNON PANEL
```

BLANKFILL Command

This command is used specify blankfill mode during panel layout. In blankfill mode, layout lines are filled with spaces when each line is inserted. The spaces can be overtyped; however, characters may not be inserted into a line without deleting or erasing spaces. The advantage of blankfill mode is that fields can be positioned in the panel without typing preceding spaces.

Format:

```
BLANKFILL
```

Notes:

- See the NULLFILL command to edit the panel layout without padding.
- PF18 can be used to toggle between blankfill and nullfill mode.

COMBINE Command

The COMBINE command eliminates the last region when in split screen. The COMBINE command causes regions to combine from the bottom up. If there are three regions, the second and third regions combine, restoring region 2. Region 2 now occupies the space occupied by regions 2 and 3 before the COMBINE. (The size of region 1 remains unchanged.) If another COMBINE command is issued, the first and second regions combine, leaving the first region restored.

Note that each step requires a COMBINE command; i.e., to combine three regions, specify two separate COMBINE commands. Note also that the OFF command performs the same function.

See also the SPLIT and REFORMAT commands.

Format:

```
COMBINE
```

Comment

A comment is a character string that serves as documentation for a PDF/CICS command in a member. Comments are not executable. A line that begins with a colon is treated as a comment. Additionally, the portion of a line to the right of a colon is treated as a comment. A comment character may be placed before any command to prevent its execution.

A comment is indicated by a character set for the site or session by the SET COMMAND COMMENT command, or the installed default, a colon. No special character is required to terminate the comment.

Example:

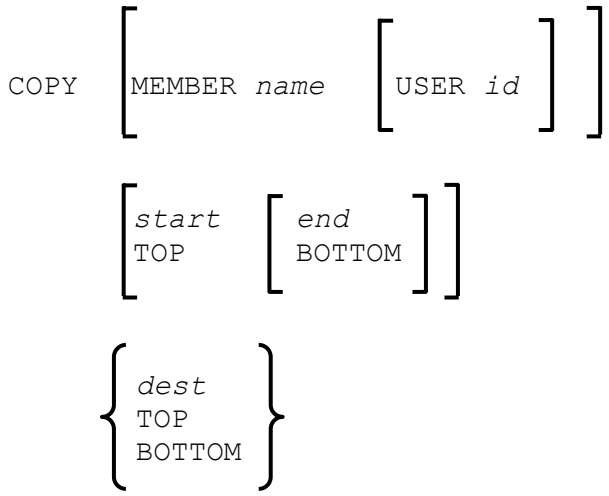
Submit BMS assembly

```
SET GEN LANGUAGE COBOL    : set up cobol map
GEN PNL A MEM A           : generate BMS source
SUB JCL1 A JCL2           : submit BMS assembly
```

COPY Command

This command copies all or part of a member. If the member name is not specified, data is copied from one specified destination in the current member to another specified destination in the current member.

Format:



name
The 1 to 8 character name of the member from which data is to be copied.

USER *id*
The 1 to 3 character user ID. This is required when copying a member belonging to another user.

start
TOP

Specification for the beginning of a range.

start
The sequence number of the line to be copied or, if specified in conjuction with *end* or BOTTOM, the first line of a range of lines to be copied.

TOP
Indicates the first line in the member.

end
BOTTOM

Specification for the ending of a range.

end
The sequence number of the line marking the end of the range to be copied.

BOTTOM
Indicates the last line in the member.

dest
TOP
BOTTOM

Specification for the COPY destination.

dest
The sequence number of the line of the current member after which the line or lines being copied are inserted.

TOP
Indicates the top of the current member.

BOTTOM
Indicates the bottom of the current member.

CREATE MEMBER Command

The CREATE MEMBER command initiates creation of a member.

Format:

```
CREATE MEMBER name [USER id] [DESCRIPTION 'string']
```

name

The 1 to 8 character member name.

USER *id*

The 1 to 3 character user ID required only if the member is to belong to another user.

'string'

A 1 to 24 character description of the member being created. For a description of valid string delimiters, see Section 2.3. Use the DIS INDEX MEMBER command to display a member's description.

Examples:

```
CREATE MEMBER JCLDEMO
CREATE MEMBER JCLDEMO DESCRIPTION 'JCL TO DEMONSTRATE'
```

CREATE PANEL Command

The CREATE command or equivalent CREATE prompter is used to initiate the creation of a new panel definition. The command presents a panel identification fill-in. Additional information is entered using the subcommands PARAMETER, LAYOUT, SUMMARY, AND FIELD. Once created, the panel exists in the current library and is prefixed with the 3 character system identifier.

Format:

```
CREATE PANEL [name]
```

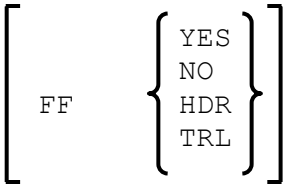
name
The 1 to 7 character name to be assigned to the new panel.

DEFINE OUTPUT DESTINATION NETWORK Command

The DEFINE OUTPUT DESTINATION command establishes the output destination of a network printer.

Format:

```
DEFINE OUTPUT DESTINATION  NETWORK  name  [ PERTASK nnn ]
```



name

Specifies a network printer name. For CICS, the 4 character value entered on the TRMIDENT parameter for the DFHTCT TYPE=TERMINAL macro in the CICS TCT. For TSO and CA ROSCOE, the 1 to 8 character VTAM network node name.

PERTASK

The current specification for how many outputs are allowed to print per print transaction. The value can be from 0 to 255.

HEADER

Defines whether a header page is printed. Specify YES to print a header page and NO to suppress the header page.

TRAILER

Defines whether a trailer page is printed. Specify YES to print a trailer page and NO to suppress the trailer page.

FF

The current specification for whether form feeds are generated for header and trailer pages at network printers. The values can be:

YES

A form feed occurs before header page and after trailer page.

NO

No form feed occurs for either the header page or trailer page.

HDR

A form feed occurs for the header page, but not the trailer page.

TRL

A form feed occurs for the trailer page, but not the header page.

Notes:

- MetaCOBOL+ supports network printers that are 328x compatible and that recognize the following printer control orders:

  Forms Feed (FF): X'0C'
  Carriage Retrun (CR): X'0D'
  New Line (NL): X'15'
  End of Message (EM): X'19'

- LIBRARY is already defined as an ouput destination when CA-MetaCOBOL+ is installed.

- For TSO: A printer can be defined as either a sytem or network destination. Prints to network destinations run asynchronously. Prints to system destinations submit batch print jobs.

DEFINE OUTPUT DESTINATION SYSTEM Command

The DEFINE OUTPUT DESTINATION command establishes the output destination of a system printer.

Format:

```
DEFINE OUTPUT DESTINATION SYSTEM name
```

name
Specifies the 1 to 8 character name of a system printer. For TSO and CA ROSCOE, the VTAM network node name.

Notes:

- LIBRARY is already defined as an ouput destination when CA-MetaCOBOL+ is installed.
- For TSO: A printer can be defined as either a sytem or network destination. Prints to network destinations run asynchronously. Prints to system destinations submit batch print jobs.

DEFINE OUTPUT DESTINATION HEADER Command

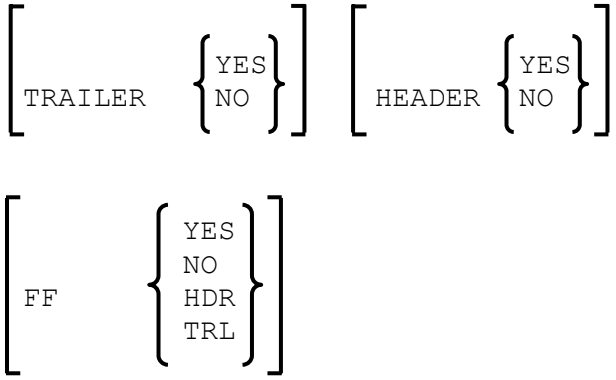
This command defines the default settings for the specified network printer. The settings include:

- Printing a header page
- Printing a trailer page
- Width of the print line
- Number of lines per page

These values override the actual printer settings.

Format:

```
DEFINE OUTPUT DESTINATION NETWORK destname [WIDTH width] [LINES lines]
                                           [PERTASK nnn ]
```



destname
The name of the destination printer.

WIDTH *width*
The width of the print line. For network printers the value may be 80-250. Specify 0 to reset the width to the default value.

LINE *lines*
The number of lines to be printed per page. The default is 60 lines per page.

PERTASK
The current specification for how many outputs are allowed to print per print transaction. The value can be from 0 to 255.

HEADER
Defines whether a header page is printed. Specify YES to print a header page and NO to suppress the header page.

TRAILER
Defines whether a trailer page is printed. Specify YES to print a trailer page and NO to suppress the trailer page.

FF
Defines whether form feeds are generated for header and trailer pages at network printers.

YES A form feed occurs before header page and after trailer page.

NO No form feed occurs for either the header page or trailer page.

HDR A form feed occurs for the header page only.

TRL A form feed occurs for the trailer page only.

Notes:

- The LINES operand is only used when CA-MetaCOBOL+ is run under CA ROSCOE, otherwise, it is ignored. This value allows network printers to be shared by CA ROSCOE and other systems such as CICS.

- To reset the named network printer width to the default value, specify zero for the WIDTH.

DELETE PANEL Command

The DELETE PANEL command or equivalent DELETE PANEL prompter is used to delete a panel. Panels can also be deleted using the DISPLAY INDEX PANEL delete subcommand.

Format:

```
DELETE PANEL [ name ]
```

name
The 1 to 7 character panel name to be deleted.

DELETE MEMBER Command

The DELETE MEMBER command or equivalent DELETE MEMBER prompter is used to delete a member. Members can also be deleted using the DISPLAY INDEX MEMBER delete subcommand.

Format:

```
DELETE MEMBER [ name ] [USER id]
```

name
The 1 to 8 character member name.

id
The 1 to 3 character user ID. This clause is required only if the specified member belongs to another user.

Example:

```
DELETE MEMBER JCLDEMO
```

DELETE OUTPUT Command

The DELETE OUTPUT command is used to remove an output from the output library. Output can also be deleted using the DISPLAY INDEX OUTPUT or DISPLAY OUTPUT STATUS delete subcommand.

Format:

```
DELETE OUTPUT { name   }
              { number }
```

name
The 1 to 8 character name of the output. The name must be unique; otherwise a message and a list of all outputs with that name appear.

number
The 1 to 4 digit output number.

Examples:

```
DELETE OUT RPT23
DEL OUTPUT COMPLIST
DEL OUT 1234
```

DELETE OUTPUT DESTINATION Command

The DELETE OUTPUT DESTINATION command removes the output destination of either a system printer or a network printer.

Format:

```
DELETE OUTPUT DESTINATION {SYSTEM name  }
                          {NETWORK name }
```

SYSTEM *name*
Specifies the 1 to 8 character name of a system printer. For TSO and CA ROSCOE, the VTAM network node name.

NETWORK *name*

Specifies a network printer name. For CICS, the 4 character value entered on the TRMIDENT parameter for the DFHTCT TYPE=TERMINAL macro in the CICS TCT. For TSO and CA ROSCOE, the 1 to 8 character VTAM network node name.

DEQUEUE Command

The DEQUEUE command releases an entity from shared or exclusive use. This command is useful after certain system failures when CA-MetaCOBOL+ maintains an entity's status as currently being edited and, therefore, unavailable for other editing. Dequeueing may also be necessary when a system failure occurs while displaying an index, output, or jobcard.

This command should be used with care and only when the administrator is certain that there is an erroneous enqueue. DEQUEUE can not be used to dequeue a resource enqueued from a different address space (MVS) or partition (VSE).

Caution: If an entity is being editied when it is dequeued, library integrity can be lost.

Format:

```
DEQUEUE PANEL name

DEQUEUE MEMBER name   { USER  id }

DEQUEUE LIBRARY libname

                      { number }
DEQUEUE OUTPUT        { name   }

                      { user id }
DEQUEUE JOBCARD       { MASTER  }
```

Notes:

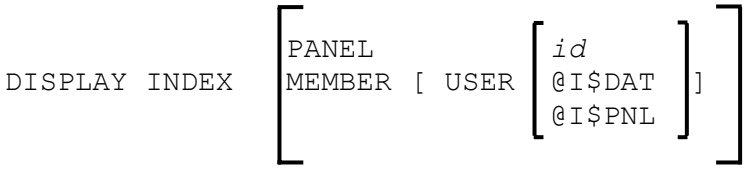
- *libname* is the name of the library specified in an error message.
- Under TSO, the DEQUEUE command is supported, but can not be used to dequeue a resource that was enqueued by a different user (from a different address space).

DISPLAY INDEX Command

The DISPLAY INDEX command lists the name, user, description, number of records, and dates created/updated for either panels or members. Panels and members can be processed by entering subcommands in the command area of the resulting index list (see below for a list of subcommands).

See also the DISPLAY INDEX OUTPUT command.

Format:



DISPLAY INDEX without an entity type of PANEL or MEMBER calls up a prompter.

PANEL
Specifies that panels are to be indexed.

MEMBER
Specifies that members are to be indexed.

id
The 1 to 3 character user ID. This clause is required only when obtaining an index of another user's members.

@I$PNL
Creates a panel index for all systems with panels in the current library. The panel system ID is the first 3 characters of the internal VLS member name.

@I$DAT
Creates a member index for all user IDs with members in the user/member library. The user ID prefix is the first 3 characters of the member name.

DISPLAY INDEX Line Commands

A number of line commands can be entered directly in the margin of the index display.

The asterisk (*) line command can be used to scroll that line to the top of the display. (Refer to Chapter 4, Editing Commands, for more information.) Other line commands can be used with panels and members, as appropriate. They perform the same functions available through primary commands. Due to the six character width of the display margin area, some commands can only be entered in abbreviated form.

The following shows a sample index of members with line commands to delete three members.

```
=>


********************************************************************************
PDF/CICS: DISPLAY INDEX       MEM                              SYS: BMS  DISPLAY

Command Name    User Description                       Records Created  Updated
=================================== T O P ================================
000001 BMS#ON   FRC  signon member                           4 12/10/90 01/08/91
DEL002 BMSMACS  FRC  BMS MACROS                             22 12/07/90 01/08/91
000003 FIRSTGEN FRC  original gen parameters                 38 01/04/91 01/08/91
000004 JCL1     FRC  1st half of BMS JCL                    26 10/25/90 01/08/91
000005 JCL2     FRC  2nd half of BMS JCL                    73 10/25/90 01/08/91
DEL006 PANVALID FRC  Order validation panel                  0 12/04/90 12/11/90
DEL007 ROUTPRT  FRC  JCL route print card                   33 12/04/90 01/08/91
000008 TESTMEM1 FRC  test data                              10 12/21/90 12/21/90
================================= B O T T O M ============================
```

The following commands can be entered in the index display for both panels and members, except where noted:

DEL, DELETE
DIS, DISPLAy
EDI, EDIT
GEN (panels only)
PRI, PRINT
SUB, SUBMIT (members only)
*

Processing Hierarchy:

You can enter any number of line commands in one display. Multiple commands are handled in three passes, as follows.

First Pass: Commands that do not need to interact with the user. This includes all commands displayed in the previous table except EDIT, DISPLAY, or *.

Successful execution of these commands clears the line command. The Updated column of the index display indicates which command was processed. This indicator remains in the data area until another command is issued against the same entity.

Second Pass: DISPLAY and EDIT.

The DISPLAY and EDIT commands replace the index display. The processing of commands is interrupted after each EDIT or DISPLAY, and the resulting panel replaces the index display. You can scroll the panels, and the index and all pending commands are kept intact, as long as you do not change entities (for example from a panel to member) or change components (for example, from panel identification to layout) during the display or edit. You can either go on to the next pending display or edit, or you can return to the index display:

- To change from one selected panel or member to the next, use the "NEXT PANEL" or "NEXT MEMBER" command, respectively.

  NEXT does not redisplay the index. Processing will proceed directly to the next entity display. The word NEXT must be followed by the current entity type (either panel or member). The common three character abbreviation of either PNL or MEM is allowed. If no more selected entities remain in the index, NEXT operates just like RETURN, showing the index display again.

- To refresh the index display panel and reshow all pending commands, issue the RETURN command.

Third Pass: *

You can execute only one asterisk (*) command. It is executed following all other pending commands.

Error Processing:

If an error is encountered during command processing, the next command will be suspended. The margin area will be highlighted and the cursor placed there. An error message will remain at the top of the screen unless cleared. If the command prepared a display (such as DUPLICATE), that screen will be displayed. To get back to the index display, issue the RETURN command or use the PF2 key. Do not use the NEXT command, as this will only try to re-execute the same command and encounter the same error again.

DISPLAY INDEX OUTPUT Command

The DISPLAY INDEX OUTPUT command lists the name and status of each output member for the current user. It is equivalent to DISPLAY OUTPUT STATUS.

Command

```
DISPLAY INDEX OUTPUT
```

DISPLAY JOBCARD Command

Each PDF/CICS user is provided with a special fill-in called JOBCARD that can be edited to contain all JOB card information. Each site has a master JOBCARD fill-in. The JOBCARD contains a maximum of five lines. The user's JOBCARD is automatically used by PDF/CICS when a PRINT command is issued online with a destination of a system printer (resulting in a batch job submission). See also the EDIT JOBCARD command.

Format:

```
                  [ USER id ]
DISPLAY JOBCARD   [ MASTER  ]
```

If neither USER *id* or MASTER is entered, the current user's JOBCARD is displayed.

MASTER
Causes the master JOBCARD to be displayed.

id
The 1 to 3 character ID of the user who owns the JOBCARD. This clause is required if the JOBCARD does not belong to the current user.

Note: Under TSO and CA ROSCOE the JOBCARD facility is not used to submit system prints. However, the DISPLAY JOBCARD command is supported for TSO or CA ROSCOE users who share ADRLIB with CICS users, and who wish to create the JOBCARD under TSO or CA ROSCOE for use in CICS.

DISPLAY MEMBER Command

Use the DISPLAY MEMBER command to display a member. DISPLAY MEMBER alone produces a prompter.

Format:

```
DISPLAY MEMBER name [USER id]
```

name
The 1 to 8 character member name.

id
The 1 to 3 character user ID required. This clause is required only if the member belongs to another user.

Example:

```
DISPLAY MEMBER BMS#ON
```

DISPLAY OUTPUT Command

The DISPLAY OUTPUT command displays an output that resides in the output library. Once the top of an output is displayed with this command, it can be scrolled and viewed using the various "browsing" commands described in Chapter 4. After the browsing activity is ended (by initiating another activity or entering the command END), the output is retained in the output library until its retention period is reached or a DELETE OUTPUT command is issued to delete the output. If the disposition is RELEASE, the output will be automatically deleted when the browsing activity is ended. To prevent the output from being deleted, issue the KEEP or KEEP OUTPUT command while displaying the output.

Output can also be displayed using the DISPLAY INDEX OUTPUT display subcommand.

Format:

name
The 1 to 8 character name of the output.

number
The 1 to 4 digit output number.

Examples:

```
DISPLAY OUTPUT COMPLIST
DISPLAY OUTPUT 1234
```

DISPLAY OUTPUT DESTINATION Command

This command produces a display of all valid output destinations.

Format:

```
DISPLAY OUTPUT DESTINATION
```

The following is the display produced by the DISPLAY OUTPUT DESTINATION command.

```
=>

******************************************************************************* >>>
BMS    DISPLAY OUTPUT         OUT DSPDESTS (00260)                      DISPLAY

+-----------------------------------------------------------------------------+
°                            D E S T I N A T I O N                            °
+----------+------+----------+-------+--------+---------+------+----+---------+
°  NAME    ° TYPE °   DISP   ° WIDTH ° HEADER ° TRAILER ° LINES° FF ° PERTASK °
+----------+------+----------+-------+--------+---------+------+----+---------+
  LOCAL      SYS     READY
  R105       SYS     READY
  LK86       NET     READY              YES      YES           YES
  RMT198     SYS     READY
  R198       SYS     READY
  LU86       NET     P-DISA             YES      YES           YES
  PU86       NET     READY              YES      YES           YES
```

The following information is displayed:

NAME
: The name of the printer. Note that CICS does not access network printers whose name is greater than 4 characters.

TYPE
: The printer type as either SYS for system or NET for network.

DISP

The current printer disposition. The user can define the disposition for a printer as READY, HOLD or STOP. The system will assign P-DISA when CICS can not access a network printer whose name is greater than 4 characters in length.

WIDTH

The current width specification. This value is displayed for network printers only. A value is displayed only when a user has changed the printer width using the ALTER OUTPUT DESTINATION command.

HEADER

The current specification for whether a header page is printed with each output at the network printer.

TRAILER

The current specification for whether a trailer page is printed with each output at the network printer.

LINES

The network printer setting for the number of lines to be printed per page. This value is displayed when CA-MetaCOBOL+ PDF/CICS is run under CA ROSCOE.

FF

The current specification for whether form feeds are generated for header and trailer pages at network printers. The values can be:

YES

A form feed occurs before header page and after trailer page.

NO

No form feed occurs for either the header page or trailer page.

HDR

A form feed occurs for the header page, but not the trailer page.

TRL

A form feed occurs for the trailer page, but not the header page.

PERTASK

The current specification for how many outputs are allowed to print per print transaction. The value can be from 0 to 255.

DISPLAY OUTPUT STATUS Command

The DISPLAY OUTPUT STATUS command is used to display or edit the status of outputs in the output library.

Format:

```
                          [ OWN    ]
                          [ ALL    ]
                          [ name   ]
DISPLAY OUTPUT            [ number ]   STATUS
```

OWN

Displays the status of outputs under the user's identification. This is optional. DISPLAY OUTPUT STATUS with no operands obtains the same display.

ALL

Displays the status of all outputs.

name

Displays the status of the output with the specified 1 to 8 character name. The name must be unique; otherwise a message and a list of all outputs with that name appear.

number

Displays the status of the output with the specified 1 to 4 digit number.

On the output status display, you can position the cursor on the line showing an output and type a command to perform one of the functions shown in the following table. (Output dispositions are explained later in this section.)

Command	Meaning
Z	Delete (more than one allowed on the screen)
D or S	Display (only one allowed on the screen at a time)
H	Alter disposition to HOLD
K	Alter disposition to KEEP
L	Alter disposition to LEAVE
R	Alter disposition to READY
P	Reissue print for output already assigned to system or network print

The following is an example of the display produced by the DISPLAY OUTPUT ALL STATUS.

```
 =>

 ********************************************************************************
>>>
 Actions: D=display, Z=delete, P=print, R=ready, L/K=Leave/Keep
 A NUM   UID NAME     CP RT DISP  DATE_CRE TIME DESCRIPTION  NRECS DST
DST_NAME
 = ===== === ======== == == ===== ======== ==== ============ ===== ===
=========
 _   188 FRC INDEX    01 02 KEEP  12/31/90 1017 PDF PRINT SE     9 LIB
 _   238 FRC PRSCREEN 01 02 READY 01/08/91 1030 DUPLICATE       13 LIB
 _   239 FRC PRSCREEN 01 02 READY 01/08/91 1031 DUPLICATE       13 LIB
 _   240 FRC PRSCREEN 01 02 READY 01/08/91 1056 HELP INFORMA    44 LIB
 _   241 FRC INDEX    01 02 READY 01/08/91 1103 PDF PRINT SE    17 SYS R105
 _   244 FRC INDEX    01 02 READY 01/08/91 1114 PDF PRINT SE    10 LIB
 _   245 FRC PRTSTATS 01 02 READY 01/08/91 1130 PRINT   STAT    12 LIB
 _   253 FRC PRSCREEN 01 02 READY 01/08/91 1143 PRINT MENU      22 LIB
 _   254 FRC PRTSTATS 01 02 READY 01/08/91 1147 PRINT   STAT    16 LIB
 _   255 FRC PRTSTATS 01 02 READY 01/08/91 1149 PRINT   STAT    16 LIB
```

A

The column where commands can be placed.

NUM

The number of the output.

UID

The user identification.

NAME

The name of the output. The name must be unique; otherwise a message and a list of all outputs with that name appear.

Defaults:

For PRINT

The default output name is the name of the entity, when the destination is the output library; or the default constant "PRTLIST", when the destination is the system printer.

For PRINT SCREEN

PRSCREEN is always displayed in the NAME column.

CP (COPIES)

The number of copies to be printed on a system or network printer.

RT (RETENTION TIME)

The retention time in days before the output is eligible to be automatically deleted.

DISP (DISPOSITION)
The disposition of the output:

CRTIN	Output in the output library is in the process of being created.
PRINT	Output in the output library is in the process of being printed.
READY	Output is available for print or display.
HOLD	Output is being held.
QHELD	Output destination's queue is on hold.
KEEP	Output is ready to be printed with a copy retained in output library after printing.
LEAVE	Output has been printed but was left in the output library at the request of the user using a KEEP disposition.
PRINTD	Output already printed but is kept for further browsing or printing.
DELET	Output deleted.

DATE_CRE
The date of the output request in a format specified at installation.

TIME (TIME REQUEST)
The time of the output request in 24-hour format.

DESCRIPTION
A 1 to 32 character description of the output. The following defaults apply if the user does not provide a description in the commands given below.

Command	Default description
PRINT	PDF Services PRINT
PRINT SCREEN	Name of the screen being printed

NRECS
The number of records.

DST (DESTINATION TYPE)
The possible destinations are:

SYS for a system printer.
LIB for the output library.
NET for a network printer.

DST_NAME (DESTINATION NAME)
The destination name.

DISPLAY PANEL Command

The following DISPLAY command or equivalent prompter is used to display an existing panel definition and to make it current.

The EDIT/DISPLAY prompter is accessed either by selecting option 1 on the Panel Maintenance Menu or by issuing the DISPLAY PANEL command.

You can display another component of the current panel definition by issuing NEXT, PREVIOUS, or the name of one of the panel definition components.

Format:

```
[DISPLAY]  [ *           ]  [ component ]
           [ PANEL name  ]
```

*****
An asterisk displays the current panel.

name
The 1 to 7 character name of the panel definition. If omitted, the default is the current panel.

component
One of the following:

IDENTIFICATION
Displays the panel identification definition fill-in.

PARAMETER
Displays the parameter definition fill-in.

LAYOUT
(Default) Displays the panel layout fill-in.

PICTURE
Displays a form of the panel layout in the top region of a split screen with each field numbered sequentially, and the field picture table in the bottom region.

SUMMARY
Displays a form of the panel layout in the top region of a split screen with each field numbered sequentially, and the field summary table in the bottom region.

FACSIMILE
Displays the specified panel layout as it would appear in its finished form.

FIELD { **nnn** / **name** }

Displays the extended field definition of the specified field.

NEXT
Displays the extended field definition for the field following the current extended field definition position.

PREVIOUS
Displays the extended field definition for the field that precedes the current extended field definition position.

Note: After a panel definition is current, the keyword DISPLAY is not needed to access other components in the same panel. However, to change modes, the word EDIT must be entered, followed by the name of the desired component or by an asterisk (*) to indicate the layout. To demonstrate, consider the following command sequence:

```
DISPLAY PANEL XYZ     : defaults to LAYOUT
DISPLAY PANEL XYZ PAR : displays PARAMETER fill-in
FACSIMILE             : displays FACSIMILE of current panel
EDIT IDE              : switches to EDIT mode, displays
                        IDENTIFICATION fill-in
```

DISPLAY SESSION OPTIONS Command

This command displays the values of parameters set for the current session. You can display every option, or a specified option. It includes those options set in the signon procedure, set during the current session, and the installation defaults.

Format:

```
                          { OPTIONS }
DISPLAY SESSION           { option  }
```

OPTIONS
Display all options.

option
Specific session option to display. Options are:

COMMAND
EDIT
ENVIRONMENT
GENERATE
OUTPUT
PANEL
SCROLL

See this manual for the SET commands used to change parameters.

DUPLICATE PANEL Command

The DUPLICATE PANEL command or equivalent DUPLICATE PANEL prompter is used to copy an existing panel definition to a new panel definition.

After DUPLICATE has been performed successfully, the new panel definition becomes the current definition, and the user is presented with an identification fill-in. The content of the newly created panel is identical to the original panel, with a new name, and is presented for modification. This new definition can also be modified during subsequent sessions.

See also the DUPLICATE MEMBER command.

Format:

```
DUPLICATE { *           }  NEWNAME name [ SYSTEM sid ]
          { PANEL name  }      [ LIBRARY libname ]
```

*****
An asterisk can be substituted for the current panel.

name
The 1 to 7 character panel name to be duplicated.

sid
The 3 character system identifier of the panel to be duplicated. If an ID is not specified, the current system is the default.

libname
The name of the library that contains the panel to be duplicated. Required only when duplicating a panel from another library.

newname
The name the panel definition is to be copied to.

Examples:

To duplicate panel XYZ to a new panel called ABC within the same library, enter:

```
DUPLICATE PANEL XYZ NEWNAME  ABC
```

To duplicate panel UPDATE from library MCTPANEL to a new panel called UPTD, in system MCT, enter:

```
DUPLICATE PANEL UPDATE SYSTEM MCT LIBNAME MCTPANEL NEWNAME UPDT
```

DUPLICATE MEMBER Command

The DUPLICATE MEMBER command or equivalent prompter copies an entire existing member to a new member. After successful completion of DUPLICATE, the new member is now the current member. The new member can also be modified at subsequent sessions.

Format:

```
             ┌ *                          ┐
DUPLICATE    │ MEMBER name  [ USER id ]   │ [TO] NEWNAME newname
             └                            ┘
               [DESCRIPTION 'string']
```

Without an asterisk or member name, the DUPLICATE prompter is displayed.

*****
: An asterisk duplicates the current member.

name
: The 1 to 8 character name of the member to be duplicated.

id
: The 1 to 3 character ID of the user who owns the member. This clause is required if the member does not belong to the current user.

newname

The name to be given to the new member. This name must not be the name of an existing member belonging to the current or specified user.

DESCRIPTION '*string*'

The 1 to 24 character descriptive text to be stored with the new member. If this clause is omitted, the description of the old member is copied. For a description of valid string delimiters, see Section 2.3.

Examples:

```
DUPLICATE MEMBER BATCH NEWNAME JCL
DUP MEM BATCH NEW JCL
```

EDIT JOBCARD Command

Each PDF/CICS user is provided with a special fill-in called JOBCARD that can be edited to contain all JOB record information. Each site has a master JOBCARD fill-in. The JOBCARD contains a maximum of five lines. The user's JOBCARD is automatically used by PDF/CICS when a PRINT command is issued online with a destination of a system printer (resulting in a batch job submission).

Format:

```
              ┌         ┐
              │ USER id │
EDIT JOBCARD  │ MASTER  │
              └         ┘
```

If neither USER *id* or MASTER is entered, the current user's JOBCARD is displayed.

MASTER

Causes the master JOBCARD to be displayed for editing.

id

The 1 to 3 character ID of the user who owns the JOBCARD. This clause is required if the JOBCARD does not belong to the current user.

Note: Under TSO and CA ROSCOE, the JOBCARD facility is not used to submit system prints. However, the EDIT JOBCARD command is supported for TSO or CA ROSCOE users who share ADRLIB with CICS users, and who wish to create the JOBCARD under TSO or CA ROSCOE for use in CICS.

EDIT MEMBER Command

The EDIT MEMBER command or equivalent EDIT MEMBER prompter edits a member.

Format:

```
EDIT MEMBER [ name ][ USER id ]
```

name
The 1 to 8 character member name.

id
The 1 to 3 character ID of the user who owns the member. This clause is required if the member does not belong to the current user.

Example:

```
EDIT MEMBER JCLDEMO
```

EDIT PANEL Command

The EDIT PANEL or equivalent EDIT PANEL prompter displays an existing panel definition and makes that panel definition current.

You can edit another component of the current panel definition by issuing NEXT, PREVIOUS, or the name of one of the panel definition component without EDIT.

Format:

```
EDIT [ * ]          [component]
     [ PANEL name ]
```

EDIT PANEL calls up a prompter. EDIT * displays the current panel.

name
The 1 to 7 character name of the panel definition.

component
One of the following:

IDENTIFICATION
Displays the panel identification definition fill-in.

PARAMETER
Displays the parameter definition fill-in.

LAYOUT [**NULLFILL** | **BLANKFILL**]

Displays the panel layout fill-in (default).

PICTURE
Displays the field picture table for all the fields in the panel.

SUMMARY
Displays the field summary table for all the fields in the panel.

FIELD { **nnn** | **name** }

Displays the extended field definition of the specified field.

FACSIMILE
Displays the specified panel layout as it would appear in its finished form.

NEXT
Displays the extended field definition for the field following the current extended field definition position.

PREVIOUS
Displays the extended field definition for the field that precedes the current extended field definition position.

Note: After a panel definition is current, the keyword EDIT is not needed to access other components in the same panel. To demonstrate, consider the following command sequence:

```
EDIT PANEL XYZ        : defaults to LAYOUT
EDIT PANEL XYZ PAR    : edits PARAMETER fill-in
FACSIMILE             : displays FACSIMILE of current panel
(from existing editing session)
IDENTIFICATION        : edits IDENTIFICATION fill-in
```

EXECUTE Command

Use the EXECUTE command or equivalent prompter to initiate the processing of PDF/CICS commands in a member. Storing commands in a member is useful when the same sequence of commands is used repetitively.

Each command stored in the member is successively executed until one of the following occurs:

- The last line is processed.
- A command causes an error. Subsequent commands in the member are not processed.

You can continue execution after an error by using the SET ENVIRONMENT EXECERROR command in the member, or before the member is executed.

Format:

```
                  *
EXECUTE         [ [ MEMBER ] name  [ USER id ] ]
```

EXECUTE * executes the current member.

name
: The 1 to 8 character member name.

id
: The 1 to 3 character user ID required. This clause is required only if the member belongs to another user.

Example:

```
EXECUTE BMS#ON
```

Re-executes the user's signon member.

GENERATE Command

The GENERATE command generates CICS-VS BMS map source statements for PDF/CICS panel definitions. The resulting BMS map source is ready for macro assembly. Refer to the *Panel Definition Facility User Guide* for more information on the GENERATE command. See also the SET PANEL and SET GENERATE commands in this manual.

Format:

```
GENERATE [ * ] [ MEMBER memname [ USER id ] ]
         [ [ PANEL ] name ]
```

*****
The asterisk generates BMS map source for the current panel definition.

name
The 1 to 7 character panel name for which BMS source macros are to be generated.

memname
The 1 to 8 character member name into which the BMS source will be placed. If omitted, the member name will be the same as the panel name.

id
The 1 to 3 character user ID required. This clause is required only if the member is to belong to another user.

HELP Command

The HELP command (PF1/13) displays a panel or series of panels that explain how to complete the current panel or that describe the specified command.

Format:

```
        [ SCROLL  ]   [ command ]
HELP    [ LINE    ]   [ SYNTAX  ]
        [ COMMAND ]   [ HELP    ]
```

HELP without any command specification describes the current panel or fill-in. PF2 or RETURN is used to return to the original activity.

When an operand is used on the HELP command, a HELP member is displayed. The HELP session allows use of editor commands such as POSITION, FIND, and SCROLL. The SPLIT command may be used in order to display the help in another region.

SCROLL
Provides an explanation of scrolling commands used in HELP members and throughout PDF/CICS.

LINE
Displays a list of available line commands.

COMMAND
Displays a list of available commands.

command
Any PDF/CICS command that can be entered in the command area.

SYNTAX
Provides an explanation of notation used in all HELP members.

HELP
Displays a list of available commands and statements.

IDENTIFICATION Command

The IDENTIFICATION command can only be given after an EDIT or DISPLAY PANEL session. This command invokes the IDENTIFICATION display or edit.

Format:

```
IDENTIFICATION
```

This command may be abbreviated as **IDE**.

KEEP OUTPUT Command

The KEEP OUTPUT command is valid only during the DISPLAY OUTPUT activity. This command changes the output's disposition to KEEP and leaves the output in the output library after browsing. (When the output disposition is RELEASE, it is deleted upon termination of the browsing activity.)

Command

```
KEEP [ OUTPUT ]
```

Note: To change the output's disposition to KEEP outside of the DISPLAY OUTPUT activity, use the ALTER OUTPUT command.

LAYOUT Command

Without an operand, the LAYOUT command is used to access the panel layout fill-in during edit or display of a panel. With an operand of BLANKFILL or NULLFILL, the LAYOUT command determines the fill mode during an edit of panel layout.

Format:

```
          [BLANKFILL]
LAYOUT    [NULLFILL ]
```

BLANKFILL
Specifies that the lines in the panel layout are padded with blanks to the full width of the panel.

NULLFILL
Specifies that the lines in the panel layout are not padded with blanks.

Notes:

- The commands NULLFILL and BLANKFILL are available during layout editing to switch the fill mode. PF18 has been assigned as a toggle between the modes.

- The fill mode, either blankfill or nullfill, affects how the panel layout is edited:

  1. Creating a new field on an empty line or after the last entry on the line:

     A null-filled screen requires leading characters to position the fields, otherwise the fields will shift left after the ENTER key is pressed. To maintain space (a gap) between fields, spaces must be entered in place of the null characters. To leave a blank line, at least one space must be entered on the line.

     A blank-filled screen will retain the fields in the exact position they are placed after the ENTER key is pressed.

  2. Inserting fields before an existing field:
     A null-filled screen allows data to be inserted by pressing the INSERT key and then typing the data.

     A blank-filled screen requires that as many characters as are being inserted must be deleted prior to inserting the new data.

- Trailing blanks and trailing blank lines are not retained with the panel definition. They are only present during the layout activity.

NULLFILL Command

The NULLFILL command specifies that the panel layout is not padded with blanks. See the BLANKFILL command to edit the panel layout padded with blanks. PF18 can be used to toggle between blankfill and nullfill mode.

Format:

```
NULLFILL
```

OFF Command

To sign off from a CA-MetaCOBOL+ session, enter the following command.

Format:

```
OFF
```

The sign-off panel appears. All information on this panel is supplied by the system. User input is not required.

OFFON Command

The OFFON command signs off from a CA-MetaCOBOL+ session and requests a new CA-MetaCOBOL+ signon panel.

Format:

```
OFFON
```

PARAMETER Command

This command or its equivalent PF key locates the parameter definition for the current panel definition.

Format:

```
PARAMETER
```

PRINT Command

The PRINT command or equivalent PRINT prompter is used to print a panel or member. The PRINT prompter is accessed either by selecting the PRINT option on the Main Menu or by issuing a PRINT command.

Format:

```
        ┌                          ┐ ┌                     ┐
        │*                         │ │                     │
PRINT   │PANEL name                │ │destination clause   │
        │MEMBER name [ USER id ]   │ │                     │
        └                          ┘ └                     ┘
```

*****

An asterisk can be substituted for the current entity type (panel or member).

name

The 1 to 7 character panel name or 1 to 8 character member name.

id

The 1 to 3 character user ID required if printing a member that belongs to another user.

destination clause

The destination clause determines the following for print output:

- Print destination and number of copies
- Disposition
- Output name
- Output description

Any or all of the following options can be specified.

```
             ┌ NETWORK name [ copies ]  DISPOSITION ┐
DESTINATION  │ LIBRARY                  DESCRIPTION │
             └ SYSTEM  name [ copies ]  NAME        ┘
```

NETWORK *name*

Specifies a network printer name. For CICS, the 4 character value entered on the TRMIDENT parameter for the DFHTCT TYPE=TERMINAL macro in the CICS TCT. For TSO and CA ROSCOE, the 1 to 8 character VTAM network node name.

LIBRARY

Specifies the output library.

SYSTEM *name*

Specifies the 1 to 8 character name of a system printer. For TSO and CA ROSCOE, the VTAM network node name.

copies

Establishes the default number of copies to be printed on a system or network printer. Any number from 1 to 99.

Note: Under VSE, *copies* is ignored for a system printer.

```
DISPOSITION  { KEEP    }
             { RELEASE }
             { HOLD    }
```

Specifies a disposition.

When a print request is issued (online or batch) and the destination is the output library, output is placed in the output library for browsing.

When a print request is printed and the destination is a system or network printer, the output is placed in the output library with the disposition having the following effects:

KEEP

The job is printed and a copy of the output is retained in the output library.

RELEASE

The job is printed and a copy of the output is retained in the output library.

HOLD

The output is held until released.

When a print request is issued in batch and the destination is a system printer, the output is printed on the system printer and no copy is retained in the output library.

NAME *print-output*
The name to be assigned to the output if the destination is the output library.

DESCRIPTION *'string'*
A 1 to 32 character delimited description of the output if the destination is the output library. For a description of valid string delimiters, see Section 2.3.

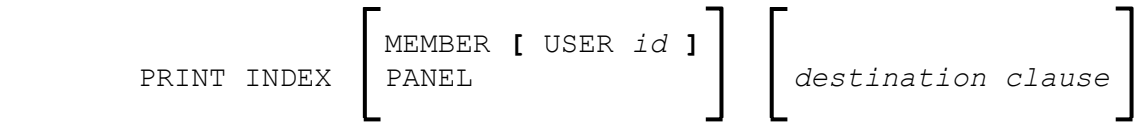
Note: Under TSO, if a PRINT command to a network printer is still executing when you enter the CA-MetaCOBOL+ OFF command, the OFF command will be redisplayed with a message explaining that the subtask is still executing. To continue executing the subtask, erase the OFF command. You can then select another CA-MetaCOBOL+ activity. To cancel the subtask and signoff, press the ENTER key to re-enter the OFF command. Canceling the subtask causes an abend.

PRINT INDEX Command

The PRINT INDEX command or equivalent PRINT INDEX prompter lists the name and status of each occurrence of the specified entity type. The PRINT INDEX prompter can be accessed by entering the PRINT INDEX command without an entity type specification.

See also the PRINT INDEX OUTPUT command.

Format:



PANEL
Specifies that panels are to be indexed.

MEMBER
Specifies that members are to be indexed.

id
The 1 to 3 character user ID required for an index of members belonging to another user.

destination clause
See page 65 for a description of the destination clause.

The following output was generated by the PRINT INDEX MEMBER command.

```
=>

*****************************************************************************
PDF/CICS: DISPLAY INDEX       MEM                              SYS: BMS  DISPLAY

Command Name    User Description                       Records Created  Updated
====== ================================ T O P ===================================
000001 BMS#ON   FRC  signon member                           4 12/10/90 01/08/91
000002 BMSMACS  FRC  BMS MACROS                              22 12/07/90 01/08/91
000003 FIRSTGEN FRC  original gen parameters                 38 01/04/91 01/08/91
000004 JCL1     FRC  1st half of BMS JCL                    26 10/25/90 01/08/91
000005 JCL2     FRC  2nd half of BMS JCL                    73 10/25/90 01/08/91
000006 PANEL123 FRC  Order validity panel                    0 12/04/90 12/11/90
000007 ROUTPRT  FRC  jcl route print card                   33 12/04/90 01/08/91
000008 TESTMEM1 FRC                                         10 12/21/90 12/21/90
====== ============================ B O T T O M ===============================
```

Descriptions of each column, as they appear from left to right, follow.

Name
The 1 to 8 character entity name.

User
The user who created this entity.

Description
A 1 to 32 character description of the entity.

Blocks
For a panel index only. The number of blocks the panel occupies.

Records
For a member index only. The number of records in the member.

Created
The date on which the entity was created.

Updated
The date on which the entity was last accessed in edit mode.

PRINT INDEX OUTPUT Command

The PRINT INDEX OUTPUT command lists the name and status of each output member for the current user. For a description of each field in the generated output, see the DISPLAY OUTPUT STATUS command in this manual.

Command

```
PRINT INDEX OUTPUT
```

PRINT OUTPUT Command

The PRINT OUTPUT command prints output residing in the output library.

Format:

```
PRINT OUTPUT { num  } [ destination clause ]
             { name }
```

num
The output number to be printed.

name
The 1 to 8 character output name. If the output name is not unique, specify the output *num* instead.

destination clause
See page 65 for a description of the destination clause.

PRINT OUTPUT DESTINATION Command

The PRINT OUTPUT DESTINATION command prints a list of all valid printer destinations and their status. The output is the same as the display produced by the DISPLAY OUTPUT DESTINATION command.

Format:

```
PRINT OUTPUT DESTINATION [ destination clause ]
```

destination clause
See page 65 for a description of the destination clause.

PRINT OUTPUT STATUS Command

The PRINT OUTPUT STATUS command prints the status of a print request. For a description of each field in the generated output, see the DISPLAY OUTPUT STATUS command in this manual.

Format:

```
                    [ OWN    ]
                    [ ALL    ]
PRINT OUTPUT        [ name   ]  STATUS [ destination clause ]
                    [ number ]
```

OWN
The default. Prints the status of output under the user's identification.

ALL
Prints the status of all output.

name
Prints the status of the output with the specified 1 to 8 character name. The name must be unique; otherwise a message and a list of all outputs with that name appear.

number
Prints the status of the output with the specified 1 to 4 digit number.

destination clause
See page 65 for a description of the destination clause.

Examples:

The following illustrates the use of PRINT OUTPUT.

```
PRINT OUTPUT X STATUS DESTINATION SYSTEM LPT1
PRINT OUTPUT ALL STATUS DEST SYS LPT1
PRINT OUTPUT STATUS
```

PRINT SCREEN Command

This command generates an output of the current screen image.

Format:

```
PRINT SCREEN [ destination clause ]
```

destination clause
See page 65 for a description of the destination clause.

PRINT SESSION OPTIONS Command

This command prints the current value of every option. The options shown include those set in the signon procedure and during the current session, and the installation defaults.

Format:

```
PRINT SESSION OPTIONS [ destination clause ]
```

destination clause
See page 65 for a description of the destination clause.

REFORMAT Command

The REFORMAT command changes the distribution of the number of lines in each region affected by the reformat (regions 1, 2, and possibly 3).

Format:

```
REFORMAT n [m]
```

n
The number of lines for the first region.

m
The number of lines for the second region.

The remaining lines are occupied by the third region.

For two regions, one amount is specified because the remainder of the screen appears for the second region. For example,

```
REFORMAT 15
```

formats 15 lines in the first region and the remainder of the screen for the second region.

For three regions, the first two amounts are specified, and the third region occupies the remainder of the screen. For example:

```
REFORMAT 10 10
```

reformats 10 lines in each of the first and second regions, and the third screen occupies the remainder of the screen.

Note: If the number of lines specified is greater than the number of lines available or less than the minimum lines per region, reformatting is ignored and a message is issued.

RENAME MEMBER Command

The RENAME MEMBER command can be used to rename a member and optionally change its description.

Format:

```
        { *           }
 RENAME { MEMBER name } [TO]  NEWNAME newname
                             [ DESCRIPTION 'string' ]
```

*****

The current member is renamed.

name

The 1 to 8 character member to be renamed.

newname

The 1 to 8 character new member name.

'string'

The 1 to 24 character descriptive text to be stored with the member. If this clause is omitted, the old description is retained. For a description of valid string delimiters, see Section 2.3.

RENAME PANEL Command

The RENAME PANEL command renames a panel.

Format:

```
        ⎧*                                                    ⎫
RENAME  ⎨PANEL name [SYSTEM sss] [LIBRARY libname]            ⎬[TO]
        ⎩                                                     ⎭NEWNAME newname
```

*

The current panel is renamed.

name

The 1 to 7 character panel to be renamed.

sss

The 1 to 3 character system ID required if the panel is in a different system. When renaming a panel from another system, the panel is placed in the current system.

libname

The 1 to 8 character library name required if the panel is in a different library.

newname

The 1 to 7 character new panel name.

RESET Command

The RESET command is used during an edit session to cancel any pending line command operations during MEMBER edit.

During panel layout editing, the pending copy and/or move operations, as well as any designated destinations, are canceled.

Format:

```
RESET
```

SET Command

The SET command allows you to specify options in a variety of areas within PDF/CICS. The command has the following format:

Format:

```
SET area keyword  [option keyword  [option keyword]  ]
```

Area keywords are the following:

COMMAND
EDIT
ENVIRONMENT
GENERATE
OUTPUT
PANEL
SCROLL

Valid options for each area keyword are discussed in the following sections.

This command is used to specify options for the command area of the PDF/CICS screen display.

Format:

```
                  { COMMENT   }
                  { DATEFOR   }
                  { DELIMITER }
                  { LINE      }
                  { REPEAT    }
SET COMMAND       { RESHOW    }  [ value ]
                  { SITE      }
                  { SEPARATOR }
                  { SESSION   }
                  { UPPERCASE }
```

COMMENT

defines the character that sets off a comment from a command. A comment is a character string that serves as documentation for a PDF/CICS command in a member. Comments are not executable. A line that begins with a colon (the default SET COMMAND COMMENT character) is treated as a comment. A comment character may be placed before any command to prevent its execution. Because the comment ends at the end of the line, no special character is required to terminate the comment.

value
is any character except a period (.) or the letter S.

DATEFOR
establishes a default date format used in the PSS header/trailer for NETWORK destination prints, and the assembler TITLE literal in members created by the GENERATE command.

value
is one of the following:

A	American (mm/dd/yy)
E	European (dd/mm/yy)
I	International (yy/mm/dd)

DELIMITER
defines the character that separates commands entered on the same command line.

value
is any character except a period (.) or the letter S.

LINE
defines the number of lines in the command area.

value
is any number from zero to five, inclusive.

REPEAT
defines the character that is entered to cause the previous command to be re-executed.

value
is any character except a period (.) or the letter S.

RESHOW
defines the character that is entered to cause the previous command to be re-displayed.

value { x | OFF | ON }

x is any character except a period (.) or the letter S. OFF disables the display of the last executed command (except when in error). ON causes all subsequent commands to be redisplayed after execution.

SITE [OPTIONS]
displays a fill-in in which command area options can be set for the site.

SEPARATOR
defines the character to be used to separate the command area from the display area.

value
is any character. Specify N (for null) to eliminate the separator line. Specify G (for grid) to produce a numbered line that shows columns.

SESSION [OPTIONS]
displays a fill-in in which command area options can be set for the session.

UPPERCASE
specifies whether PDF/CICS system panels and messages will be displayed in uppercase or in lowercase. Can be abbreviated to UPC.

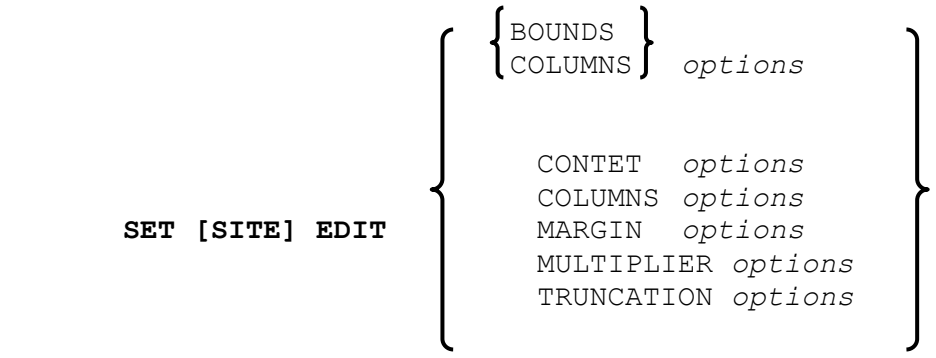
value { YES / NO }

YES indicates that uppercase will be used. NO indicates that lowercase will be used.

SET EDIT Command

This command establishes the default values used during a PDF/CICS edit session.

Format:



SITE

sets the option for the entire site. If omitted, the option is set only for the current user's session.

BOUNDS
COLUMNS

establishes the default column boundaries used when searching for a string, changing a string, or shifting data.

Warning: Use of the SITE option with this keyword will set the column boundaries for the entire site.

num-1 [MAX / *num-2*]

num-1
The left-hand column that begins the range.

num-2
The right-hand column that ends the range.

MAX
(Default) The farthest right-hand column.

Case Options

establishes whether text entered during an edit session is to be translated into uppercase (UPPER) or as entered (MIXED).

CASE { UPPER | MIXED }

UPPER
translates text to uppercase.

MIXED
translates text to uppercase.

CONTEXT ***n***
establishes the default number of context lines (lines retained on the screen for reference) appearing above and below the area on the screen opened by an INPUT command.

n
is a value from 1 to 15.

MARGIN { RIGHT | LEFT }

establishes the position of the sequence number and command area when an EDIT or DISPLAY panel is presented: Either on the right side or left side of the data area.

MULTIPLIER { RIGHT | LEFT }

establishes whether to place the number that acts as a multiplier or replication factor for a line command to the right or to the left of the command.

TRUNCATION { Y/ON | N/OFF }

establishes whether or not truncation can occur as a result of a SHIFT or CHANGE command.

If SET EDIT TRUNC is OFF, any CHANGE command that would cause data to overflow the current column boundaries is prohibited, and that line is scrolled to the top of the edit window. Any shift that would move data on a line past the current column boundaries stops when the data on that line reaches the column boundary.

SET ENVIRONMENT Command

This command controls print and display formats.

Format:

```
                                  ⎧ DATEFOR  ⎫
SET [SITE] ENVIRONMENT            ⎪ EXECERROR ⎪ value
                                  ⎪ LINES    ⎪
                                  ⎩ PRINT-ID ⎭
```

SITE
sets the option for the entire site. If omitted, the option is set only for the current user's session.

DATEFOR *pattern*
establishes the date format to be used by the PRINT commands in page headings. The *pattern* is a delimited sequence of characters (maximum of 30) that represents the format of the date.

Specify any logical combination of the date patterns shown the chart below. For a description of valid string delimiters, see Section 2.3.

Notation	Meaning	Example assuming January 5, 1991
YEAR	Year in full	1991
YY	Year without century	91
Y	Year without decade	0
MONTH	Month spelled out (uppercase)	JANUARY
LCMONTH	Month spelled out (initial letter uppercase)	January
MON	Month abbreviation (uppercase)	JAN
LCMON	Month abbreviation (initial letter uppercase)	Jan
MM	Month number with leading zero if necessary	01
M	Month number with no leading zero	1
DD	Day with leading zero if necessary	05
D	Day with no leading zero	5
DDD	Julian day (numeric day of the year (1-366))	005
WEEKDAY	Day spelled out (uppercase)	SATURDAY
LCWEEKDAY	Day spelled out (initial letter uppercase)	Saturday
DAY	Day abbreviation (all letters uppercase)	SAT
LCDAY	Day abbreviation (initial letter uppercase)	Sat

Any characters except uppercase alphabets in the date pattern remain unchanged.

The actual text indicated by the keywords MONTH, LCMONTH, MON, LCMON, WEEKDAY, LCWEEKDAY, DAY, AND LCDAY is defined for each site by the site administrator in the PMS table PMSTBLS.

Example:

Assuming a date of January 5, 1991, the command

```
SET ENVIRONMENT DATEFOR 'M/D/YY'
```

results in a date of

```
1/5/91
```

EXECERROR { QUIT | CONTINUE }

specifies whether to quit or continue after an error resulting from the execution of commands in a member. The SITE option is not valid for this keyword. Including this command in a member affects all commands that follow it in the member until the next SET ENVIRONMENT EXECERROR command or until the end of the member. QUIT specifies to stop executing the member when any of the statements following causes an error. CONTINUE specifies to continue executing the member when any of the statements following causes an error.

Example:

Executing the following PDF/CICS member renames a series of panels. If an error is encountered, a message is displayed and processing continues.

```
SET ENVIRONMENT EXECERROR CONTINUE
RENAME PANEL PRODDEM NEWNAME OLDDEM
RENAME PANEL TSTDEM NEWNAME PRODDEM
RENAME PANEL PRODJCL NEWNAME OLDJCL
RENAME PANEL TSTJCL NEWNAME PRODJCL
```

LINES *nn*

specifies the maximum number of lines (20 through 99) per print page to be used by the PRINT command. It applies to print requests to network and system printers when printing entities, session options, and indexes.

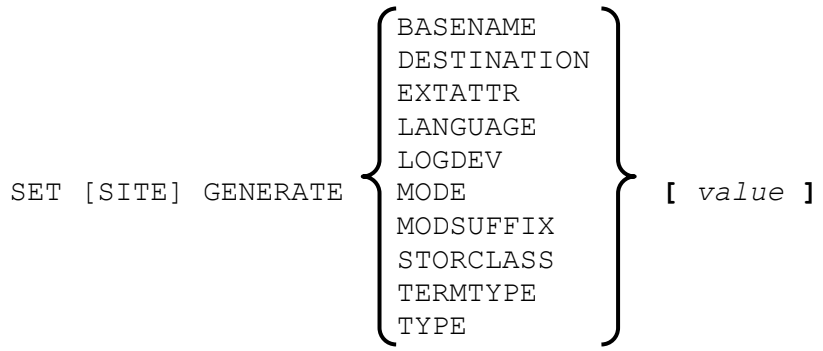
PRINT-ID xxxx

specifies a 1 to 4 character CICS transaction ID. This keyword provides the ability to assign print tasks to a separate user-specified CICS transaction. This allows CICS performance analysis packages or transaction accounting packages to isolate transactions for chargeback and resource analysis. This command is ignored under TSO and CA ROSCOE.

SET GENERATE Command

This command establishes default values for the DFHMSD macro operands created by the GENERATE command. The DFHMSD macro operands are described briefly. For more detailed information, refer to the IBM CICS documentation.

Format:



SITE

sets the option for the entire site. If omitted, the option is set only for the current user's session.

BASENAME [*name***]**

specifies that the same storage base will be used for the symbolic description maps from more than one map set. Determines whether the DHFMSD macro operand BASE= is generated. If no name is supplied, the BASE= operand is omitted. This command is ignored when SET GENERATE STORCLASS AUTO is in effect.

name

the 1 to 8 character name used on the BASE= operand. BASE=name is generated.

specifies the destination library for members created by the GENERATE PANEL command.

LIBRARY

members are stored in the VLS library.

ROSCOE

members are stored in the user's CA ROSCOE library, provided that the PDF/CICS user ID OR the 'USER *id*' specified on the GENERATE command matches the user's CA ROSCOE prefix. If the PDF/CICS user ID (supplied during signon or explicitly by the GENERATE command) is not the CA ROSCOE prefix, an error message is issued when attempting to generate a member into the CA ROSCOE library.

EXTATT [NO / YES]

specifies the attribute types to be included in the symbolic description map and physical map. Determines whether the DFHMSD macro operands DSATTS= and MAPATTS= are generated. The extended attribute operands in DSATTS= and MAPATTS= will be generated only when TERMTYPE is 3270, 3270-1, 3270-2, or ALL.

If neither YES or NO is specified, the DFHMSD macro operands DSATTS= and MAPATTS= are conditionally generated. The panel parameter settings and field specifications are examined to determine which extended attributes are required. The appropriate options are generated.

YES
: The DFHMSD macro operands DSATTS= and MAPATTS= are generated. The following options are generated for each operand:

```
DSATTS=(COLOR,HILIGHT,OUTLINE,PS,SOSI,TRANSP,VALIDN)
MAPATTS=(COLOR,HILIGHT,OUTLINE,PS,SOSI,TRANSP,VALIDN)
```

NO
: The DFHMSD macro operands DSATTS= and MAPATTS= are omitted.

LANGUAGE [COBOL / ASM / PLI / RPG]

specifies the source language of the application programs into which the symbolic description maps in the map set will be copied. Determines whether the DFHMSD macro operand LANG= is generated. If no value is supplied, the LANG= operand is omitted.

COBOL
: LANG=COBOL is generated.

ASM
: LANG=ASM is generated.

PLI
: LANG=PLI is generated.

RPG
: LANG=RPG is generated.

LOGDEV [*xx***]**
: specifies the code to be used by CICS to determine the logical device mnemonic used for BMS output operation. Determines whether the DFHMSD macro operand LDC= is generated. If no value is supplied, the LDC= operand is omitted.

xx
: LDC=*xx* is generated.

MODE [IN | OUT | BOTH]
: specifies whether the map is to be used for input, output, or both. Determines whether the DFHMSD macro operand MODE= is generated. If no value is supplied, the MODE= operand is omitted.

IN
: MODE=IN is generated.

OUT
: MODE=OUT is generated.

BOTH
: MODE=INOUT is generated.

MODSUFFIX [*x***]**
: specifies a one character user-defined device dependent suffix for this map set. Determines whether the DFHMSD macro operand SUFFIX= is generated. If no value is supplied, the SUFFIX= operand is omitted. When this option is in effect, the SET GENERATE TERMTYPE command is ignored.

STORCLASS [AUTO | BASED]
: specifies which areas of storage maps will occupy (based on the language of the application programs). Determines whether the DFHMSD macro operand STORAGE=AUTO is generated. If no value is supplied, the STORAGE=AUTO operand is omitted.

AUTO
: STORAGE=AUTO is generated. When this option is in effect, the option SET GENERATE BASENAME is ignored.

BASED
: STORAGE=AUTO is omitted. Refer to BASENAME for more information.

TERMTYPE [*xxxxxxxx***]**

specifies the type of terminal or logical unit associated with the map set. Determines whether the DFHMSD macro operand TERM= is generated. If no value is supplied, the TERM= operand is omitted. This command is ignored when SET GENERATE MODSUFFIX is in effect.

xxxxxxxxx

1 to 9 character terminal or logical unit. TERM=xxxxxxxxx is generated.

TYPE { OBJECT | DSECT }

specifies the type of map to be generated using the definition. Determines the setting for the DFHMSD macro operand TYPE=.

OBJECT

TYPE=MAP is generated.

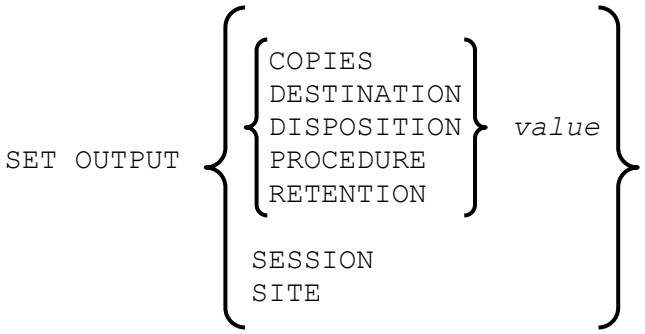
DSECT

TYPE=DSECT is generated.

SET OUTPUT Command

This command establishes default values for output processing.

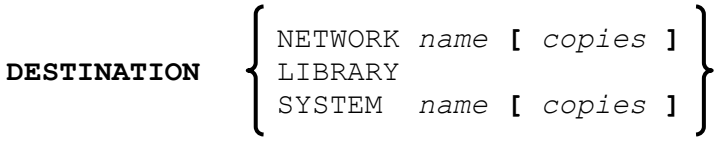
Format:



COPIES *nn*
establishes the default number of copies to be printed on a system or network printer if the COPIES= parameter is omitted in a DESTINATION clause.

Note: Under VSE, *copies* is ignored for a system printer.

nn
Any number from 1 to 99.



establishes the default printout destination.

NETWORK *name*
Specifies a network printer name. For CICS, the 4 character value entered on the TRMIDENT parameter for the DFHTCT TYPE=TERMINAL macro in the CICS TCT. For TSO and CA ROSCOE, the 1 to 8 character VTAM network node name.

LIBRARY
Specifies the output library.

SYSTEM *name*
Specifies the 1 to 8 character name of a system printer. For TSO and CA ROSCOE, the VTAM network node name.

copies

establishes the default number of copies to be printed on a system or network printer. Any number from 1 to 99.

Note: Under VSE, *copies* is ignored for a system printer.

DISPOSITION { HOLD | RELEASE | KEEP }

establishes the default output disposition. This default can be overridden on a PRINT command.

The following table shows the effects of specifying each of the disposition options on print requests.

Destination		
Output Disposition	Output Library	System or Network Printer
Release	Output is placed in the output library for browsing on the screen.	Output is placed in the output library and printed on a system or network printer. No copy is retained in the output library.
Hold	Same as above.	Output is placed in the output library and held until released (by using the ALTER OUTPUT or PRINT OUTPUT commands.)
.Keep	Same as above.	Output is placed in the output library and a batch job is printed on a system or network printer. A copy is retained in output.

Figure 11. Table of Output Disposition Options

PROCEDURE *procname*
sets the name of the cataloged procedure executed to print an output at a SYSTEM printer. If you change this parameter, be sure to rename the installed cataloged procedure PSSUTIL to the same name, or create a new copy of PSSUTIL with the new name.

procname
Any valid cataloged procedure name.

RETENTION *nn*
sets the default number of days an output can reside in the output library before being deleted.

nn
Any number from 1 to 99.

{ **SESSION** | **SITE** }
calls up a fill-in in which all output options can be set for this session (SESSION) or for all sessions (SITE).

SET PANEL Command

This command establishes default values for panel creation and editing.

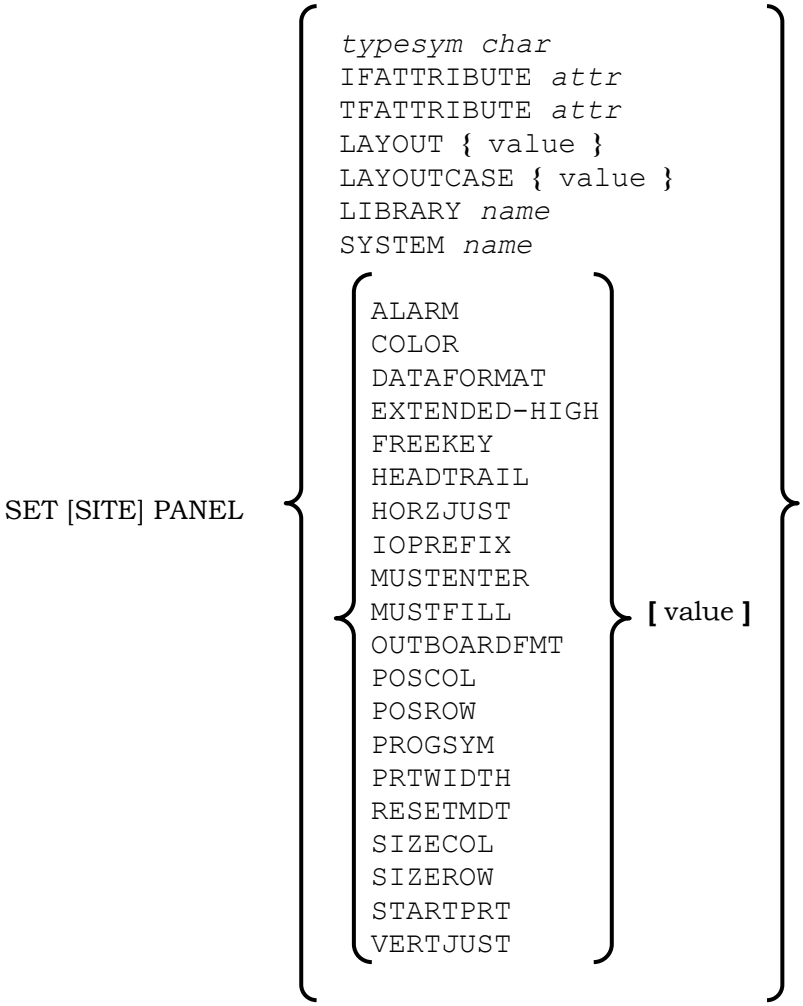
When creating a panel, the current SET PANEL values are copied to the panel's parameter section, accessible from the Panel Parameter Fill-in. If a SET PANEL value has not been set, the corresponding value in the Panel Parameter Fill-in is left blank. The values in the Panel Parameter Fill-in are used to generate keywords on the BMS DFHMDI macro created by the GENERATE command. In most cases, if a value is not set on the Panel Parameter Fill-in, the corresponding keyword is omitted from the generated macro. Exceptions are noted in the SET PANEL keyword descriptions that follow. The DFHMDI macro keywords are described here briefly. For more detailed information, refer to the IBM CICS documentation.

When editing a panel, defaults such as panel layout symbols, letter case, fill mode, and screen attributes are obtained from the current corresponding SET PANEL values.

To view the current value for all of the SET PANEL keywords, issue the DISPLAY SESSION OPTIONS command.

For more information on creating and editing panels, and Panel Parameter Fill-in, refer to the *Panel Definition Facility User Guide*.

Format:



SITE

sets the option for the entire site. If omitted, the option is set only for the current user's session.

typesym *char*

specifies the name of the symbol used in the panel layout whose function is assigned to the designated character. Note that the symbols can be defined for an individual panel on the Panel Layout fill-in.

STARTSYM	start of field
ENDSYM	end of field
NEWSYM	add field
DELSYM	delete field
MOVESYM	move field to new location
COPYSYM	copy field to new location
DESTSYM	mark destination of copy or move operation

char
Any character.

IFATTRIBUTE *attr*
specifies the default screen attributes for panel fields defined for input. The installed default is UAL. Panel attributes are described in the *Panel Definition Facility User Guide*. Can be abbreviated to IFATT.

TFATTRIBUTE *attr*
specifies the default screen attributes for panel fields defined with text. The installed default is PSL. Panel attributes are described in the *Panel Definition Facility User Guide*. Can be abbreviated to TFATT.

LAYOUT { BLANKFILL | NULLFILL }

specifies the fill mode during panel layout editing. Can be abbreviated to LAY.

BLANKFILL
indicates that panel layout is padded with blanks. Can be abbreviated to BLK.

NULLFILL
indicates that panel layout is padded with nulls. Can be abbreviated to NUL.

LAYOUTCASE { MIXED | UPPER }

specifies whether panel text is translated to uppercase. Can be abbreviated to LAYCASE.

MIXED
Panel text remains in mixed case (as entered). Can be abbreviated to M.

UPPER
Panel text is translated to uppercase. Can be abbreviated to U.

LIBRARY *name*
specifies the 1 to 8 character panel library name in which the panel is stored. Can be abbreviated to LIB. For more information on panel libraries, see the *Panel Definition Facility User Guide*.

SYSTEM *xxx*
specifies a 3 character system name to which the panel will belong. Can be abbreviated to SYS. For more information on systems, see the *Panel Definition Facility User Guide*. If the specified system does not exist, it will be created. Use the DISPLAY INDEX PANEL USER @I$PNL command to display a list of all systems containing panels in the current library. See the DISPLAY INDEX command for more information.

ALARM [NO | YES]

Specifies whether the 3270 audible alarm is activated. ALARM YES specifies that the DFHMDI macro operand CTRL=ALARM is generated. If no value is supplied, or ALARM NO is specified, the CTRL=ALARM operand is omitted.
Corresponding Panel Parameter Fill-in field: Alarm?.
Corresponding DFHMDI macro operand: CTRL=.

COLOR [NEUTRALWHITE | BLUE | RED | PINK | GREEN | TURQUOISE | YELLOW | DEFAULT]

Specifies the default color for the panel. If no value is supplied, the COLOR=DEFAULT operand is generated. Can be abbreviated to the first letter of each value (e.g., B for blue, N for neutral).
Corresponding Panel Parameter Fill-in field: Color:
Corresponding DFHMDI macro operand: COLOR=

DATAFORMAT [FIELD | BLOCK]

Specifies the format of the data from a screen as seen by the application program. If no value is supplied, the DATA= operand is omitted.
Corresponding Panel Parameter Fill-in field: Data Format:
Corresponding DFHMDI macro operand: DATA=

FIELD
: data is to be passed as a sequence of fields between the device and the application program, by BMS run-time services. DATA=FIELD is generated.

BLOCK
: data is to be passed as a contiguous stream between the device and the application program, by BMS run-time services. DATA=BLOCK is generated.

EXTENDED-HIGH [NONEOFF | BLINK | REVERSE | UNDERSCOREUNDERLINE]

Specifies the type of extended attribute support to be included in the panel. If no value is supplied, the HILIGHT= operand is omitted. Can be abbreviated to EXHI.
Corresponding Panel Parameter Fill-in field: Extended Highlight?
Corresponding DFHMDI macro operand: HILIGHT=

NONEOFF
: indicates that no extended attributes are supported. HILIGHT=OFF is generated. Can be abbreviated to N or O.

BLINK
: indicates that the "blinking" extended attribute condition is set. HILIGHT=BLINK is generated. Can be abbreviated to B.

REVERSE
: indicates that the "reverse video" extended attribute condition is set. HILIGHT=REVERSE is generated. Can be abbreviated to R.

UNDERSCOREUNDERLINE
: indicates that the "underscore" extended attribute condition is set. HILIGHT=UNDERLINE is generated. Can be abbreviated to U.

FREEKEY [NO | YES]

Specifies whether the keyboard is to be locked after a screen is sent. If no value is supplied, the CTRL=FREEKB operand is omitted.
Corresponding Panel Parameter Fill-in field: Free keyboard?
Corresponding DFHMDI macro operand: CTRL=

HEADTRAIL [HEADER | TRAILER]

Specifies whether the panel is positioned by BMS as a header or trailer panel. If no value is supplied, neither operand is generated.
Corresponding Panel Parameter Fill-in field: Header or Trailer?
Corresponding DFHMDI macro operands: HEADER= and TRAILER=

HORZJUST [RIGHT | LEFT]

Specifies whether the panel is justified to the right or left margin. JUSTIFY=LEFT or JUSTIFY=RIGHT is generated. See the VERTJUST operand for other JUSTIFY= settings.
Corresponding Panel Parameter Fill-in field: Justification?
Corresponding DFHMDI macro operand: JUSTIFY=

IOPREFIX [NO | YES]

Specifies whether BMS should include a filler in the symbolic description maps to allow for the unused TIOA prefix. If no value is supplied, the TIOPREFIX= operand is omitted.
Corresponding Panel Parameter Fill-in field: I/O Prefix Area?
Corresponding DFHMDI macro operand: TIOAPREFIX=

MUSTENTER [NO | YES]

Specifies whether the CICS validation rule "must enter" is in effect. If no value is supplied, or MUSTENTER=NO is specified, the VALIDN=MUSTENTER operand is omitted.
Corresponding Panel Parameter Fill-in field: Must enter?
Corresponding DFHMDI macro operand: VALIDN=

NO
: indicates that the "must enter" rule is not in effect. VALIDN=MUSTENTER is omitted.

YES
: indicates that data must be entered into a field. An attempt to move the cursor from an empty field causes the inhibit input condition. VALIDN=MUSTENTER is generated.

MUSTFILL [NO | YES]

Specifies whether the CICS validation rule "must fill" is in effect. If no value is supplied, or MUSTFILL NO is specified, the VALIDN=MUSTFILL operand is omitted.

Corresponding Panel Parameter Fill-in field: Must fill?
Corresponding DFHMDI macro operand: VALIDN=

NO
: indicates that the "must fill" rule is not in effect. VALIDN=MUSTFILL is omitted.

YES
: indicates that a field must be filled completely with data. An attempt to move the cursor from the field before it has been filled, or to transmit data from the incomplete field, will cause the inhibit input condition. VALIDN=MUSTFILL is generated.

OUTBOARDFMT [NO | YES]

Specifies whether 3650-device outboard formatting is to be used. If no value is supplied, the OBFMT= operand is omitted.
Corresponding Panel Parameter Fill-in field: Outboard format?
Corresponding DFHMDI macro operand: OBFMT=

POSCOL [NEXT | SAME | *nnn*]

Specifies the left-to-right position of the panel on the screen. If no value is supplied, the COLUMN= operand is omitted.
Corresponding Panel Parameter Fill-in field: Position:
Corresponding DFHMDI macro operand: COLUMN=

NEXT
: the left or right margin is established in the next available column from the left or right on the current line. COLUMN=NEXT is generated.

SAME
: the left or right margin is established in the same column as the last nonheader or nontrailer panel used. COLUMN=RIGHT is generated.

nnn
: any number from 1 to 240. COLUMN=nnn is generated.

POSROW [NEXT | SAME | *nnn*]

Specifies the starting line on a page in which data for the map is to be formatted. If no value is supplied, the LINE= operand is omitted.
Corresponding Panel Parameter Fill-in field: Position:
Corresponding DFHMDI macro operand: LINE=

NEXT
: formatting of data begins on the next available completely empty line. LINE=NEXT is generated.

SAME
: formatting of data begins on the same line as that used for the preceding BMS command. LINE=SAME is generated.

nnn
: any number from 1 to 240. LINE=nnn is generated.

PROGSYM [*x* | *hh* | BASE]

Specifies the program symbols to be used.
Corresponding Panel Parameter Fill-in field: Programmed symbols?
Corresponding DFHMDI macro operand: PS=

x
: specifies a single EBCDIC character that identifies a set of programmed symbols.

hh
: specifies a 2 character hex-value that identifies a set of programmed symbols.

BASE
: specifies that the base symbol set is to be used.

PRTWIDTH [L4040 | L6464 | L8080 | HONEOMEOM]

Specifies the line length of the printer. L40, L64, and L80 force a new line after 40, 64, or 80 characters, respectively. HONEOM causes the default printer length to be used. If no value is supplied, the CTRL=*length* operand is omitted.
Corresponding Panel Parameter Fill-in field: Print width?
Corresponding DFHMDI macro operand: CTRL=

RESETMDT [NO | YES]

Specifies whether the modify data tags are to be reset to an unmodified condition before any panel data is written to the buffer. If no value is supplied, or RESETMDT=NO is specified, the CTRL=FRSET operand is omitted.
Corresponding Panel Parameter Fill-in field: Reset MDTs?
Corresponding DFHMDI macro operand: CTRL=

SIZECOL [*nnn***]**

Specifies a number of columns in a panel. The number can be from 1 to 240. If no value is supplied, the SIZE=*column* operand is omitted.
Corresponding Panel Parameter Fill-in field: Size:
Corresponding DFHMDI macro operand: SIZE=

Note: CICS limits the size of a map to match the dimensions of the device for which it is defined. Therefore, regardless of the size specified here, the ultimate size specification will be made on the DFHMDI macro when the BMS generation specification is filled in.

SIZEROW [*nnn***]**

Specifies a number of rows in a panel. The number can be from 1 to 240. If no value is supplied, the SIZE=*line* operand is omitted.
Corresponding Panel Parameter Fill-in field: Size:
Corresponding DFHMDI macro operand: SIZE=

Note: CICS limits the size of a map to match the dimensions of the device for which it is defined. Therefore, regardless of the size specified here, the ultimate size specification will be made on the DFHMDI macro when the BMS generation specification is filled in.

```
          [ NO  ]
STARTPRT  [ YES ]
```

Specifies whether to start the printer. If no value is supplied, or STARTPRT NO is specified, the CTRL=PRINT operand is omitted.
Corresponding Panel Parameter Fill-in field: Start printer?
Corresponding DFHMDI macro operand: CTRL=

```
          [ TOP    ]
VERTJUST  [ BOTTOM ]
```

Specifies the top-to-bottom positioning of the panel. See the HORZJUST operand for other JUSTIFY= settings.
Corresponding Panel Parameter Fill-in field: Justification?
Corresponding DFHMDI macro operand: JUSTIFY=

TOP
: justify the panel to force the panel to be at the top of a new page. JUSTIFY=FIRST is generated.

BOTTOM
: justify the panel to force the panel to be at the bottom of the current page. JUSTIFY=BOTTOM is generated.

SET SCROLL Command

This command establishes the default means of scrolling forward and scrolling backward.

Format:

```
              { CURSOR }
SET SCROLL    { FRAME  }
```

CURSOR
: The line containing the cursor is positioned at the top, for scroll forward, or at the bottom, for scroll backward.

FRAME
: An entire region's contents is scrolled, except for overlapping "context" lines.

SPLIT Command

The SPLIT command divides the screen display into regions. The SPLIT command is entered in one of three ways: with the cursor position in the command area, with the cursor position in the display area, or with line specification. The SPLIT command without arguments divides the last or only region of the display area. See also the COMBINE command in this manual.

SPLIT with the cursor positioned in the command area divides the regions evenly. The minimum region size is nine lines. If there is not enough room on the screen for another region, the SPLIT is not applied and an error message is issued. The current contents of the display area appear in the first region. If the content for the second region is not specified with a command, the Main Menu appears in the second region.

When there is more than one region, commands can be prefixed with a region number and a space. The commands apply by default to region 1 if no prefix is specified. Commands also can be entered on multiple lines or delimited on a single line. For example:

```
SPLIT
2 EDIT MEMBER JCLDEMO; 1 DISPLAY PANEL CHKPNT
```

To display panel CHKPNT in one region and to edit member JCLDEMO in a second region when only one region exists, enter:

```
SPLIT
1 DISPLAY PANEL CHKPNT
2 EDIT MEMBER JCLDEMO
```

Cursor Position in Display Area

The cursor position can also be used to control where splitting occurs. With the cursor positioned at some point in the display area, the SPLIT command splits the screen into regions at that point. For example, if the cursor was positioned on the twentieth line of the screen, the screen is divided into regions at line 20, assuming enough lines remain to accommodate a second region.

Line Specification

SPLIT can also be entered with explicit control of region sizes.

Format:

```
SPLIT [n [m]]
```

n

Specifies the new size of the existing, last, or only region (the number of lines).

m

Specifies the size of the new region (the number of lines). If *m* is specified, it stipulates a minimum number of lines that must be available to a second region in order for the split to take effect. When *m* is not specified, the second region occupies the remaining available lines.

For example, the command

```
SPLIT 2O
```

formats 20 lines in the first region and the remainder in a second region. The command,

```
SPLIT 20 10
```

formats 20 lines in the first region and a minimum of 10 lines in the second region.

Note that if the number of lines specified is greater than the number of lines available, the SPLIT command is ignored and a message is issued.

Splitting the Second Region

The SPLIT command always applies to the last region displayed. If there are already two regions, the SPLIT applies to the second region. The cursor in the display area marks the point at which splitting occurs; specifying SPLIT with the cursor positioned in the command area divides the regions evenly; SPLIT with line numbers may divide the second region unevenly.

SUBMIT Command

The SUBMIT command or equivalent SUBMIT prompter submits a member containing a batch jobstream or a series of members that contain portions of a jobstream.

Format:

```
        ⎧*                        ⎫
SUBMIT  ⎨name-1 [USER user-id]    ⎬
        ⎩                         ⎭
         [ name-2 [USER user-id]]
         [ name-8 [USER user-id]]
```

SUBMIT * submits the current member.

name-1 name2...name-8
From one to eight member names. If more than one member is specified, the contents of the members are first concatenated in the order the members are specified and then submitted.

user-id
The 1 to 3 character ID of the user who owns the member. This clause is required if the member does not belong to the current user.

TIME Command

The TIME command is entered in the command area and displays the current date and time.

Format:

```
TIME
```

Sample Display:

```
MCADTIME01I - Current date and time 02/26/91 08:39:14
```

The format of the date is determined by the date format option on the SET COMMAND SESSION OPTIONS fill-in. The date format can also be changed by the SET COMMAND DATEFOR command.

4. Editing Commands

Editing in PDF/CICS

This section describes the editing features available in CA-MetaCOBOL+ PDF/CICS.

Introduction to a PDF/CICS Editing Session

Issuing any PDF/CICS EDIT command begins an editing session. The EDIT MEMBER command, for example, results in a data display in the region, with a scale line across the top of the region and a 6-digit sequence number and command field on the right-hand side (or left-hand side, see SET EDIT MARGIN), as shown below.

```
Command....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
 ********************************** T O P ********************************
  TITLE 'BMS DSECT FOR PANEL ORDSYS  GENERATED ON 01/24/91 AT 10:36'
 ORDSYS   DFHMSD TYPE=DSECT,                                          X 000100
               TERM=3270,                                             X 000200
               LANG=COBOL,                                            X 000300
               MODE=INOUT,                                            X 000400
               STORAGE=AUTO                                             000500
 ORDSYS   DFHMDI COLOR=BLUE,                                          X 000600
               DSATTS=(COLOR,HILIGHT,PS,VALIDN,OUTLINE,SOSI,TRANSP),  X 000700
               MAPATTS=(COLOR,HILIGHT,PS,VALIDN,OUTLINE,SOSI,TRANSP), X 000800
               SIZE=(15,80),                                          X 000900
               TIOAPFX=YES                                              001000
 ORDER#   DFHMDF POS=(4,11),                                          X 001100
               LENGTH=7,                                              X 001200
               INITIAL='ORDER #',                                     X 001300
               ATTRB=(PROT,ASKIP,NORM)                                  001400
 ORDINP   DFHMDF POS=(4,19),                                          X 001500
               LENGTH=8,                                              X 001600
               Aa                                                    15001700
```

Figure 23. Sample PDF/CICS Edit Session

During an edit session, a component of a PDF/CICS entity can be edited by:

- Using the 3270 keyboard
- Using line commands
- Using primary editing commands
- Using PF keys

The components of PDF/CICS that can be edited or displayed and a summary of the type of editing possible within a component are described in the table below.

Component	Editing Using 3270 Keyboard	Primary Commands	Line Commands
Panel Definition			
Identification	yes	SCROLL FORWARD/BACKWARD, TOP/BOTTOM, plus CHECKPOINT and ROLLBACK commands in EDIT mode.	none
Parameters	yes	Same as above.	none
Layout	yes	SCROLL FORWARD/BACKWARD, TOP/BOTTOM, LEFT/RIGHT, POSITION; plus CHECKPOINT, ROLLBACK, and INPUT commands in EDIT mode.	none
Picture	yes	SCROLL FORWARD/BACKWARD, TOP/BOTTOM, POSITION; plus CHECKPOINT and ROLLBACK in EDIT mode.	none
Summary	yes	SCROLL FORWARD/BACKWARD, TOP/BOTTOM, POSITION; plus CHECKPOINT and ROLLBACK in EDIT mode.	none
Field	yes	SCROLL FORWARD/BACKWARD, TOP/BOTTOM, plus CHECKPOINT and ROLLBACK commands in EDIT mode.	none
Member	yes	All commands in EDIT mode; SCROLL FORWARD/BACKWARD, TOP/BOTTOM, FIND, NEXT, INCLUDE, EXCLUDE, FIRST, LAST, PREVIOUS, POSITION commands in DISPLAY mode.	All line commands in EDIT mode. Only the * line command in DISPLAY mode.

Figure 3. Table of EDIT/DISPLAY Summary

Editing Using 3270 Hardware Facilities

Editing keys affect the field in which the cursor is found. Note that each line in the region contains two fields, the "text field" and the "sequence number and command field".

ERASE EOF key

When the cursor is within a display area field, erases data following the cursor to the end of the field.

Erasing a sequence number with the EOF key has no effect. If the entire text field on a line is erased with the EOF key, any line commands specified in the sequence number and command field on that line are ignored.

DELETE key

Deletes characters and spaces one at a time at the cursor, while shifting data from the right and leaving null characters in the right-most columns of the field.

INSERT key

Allows characters to be entered in a field in the column containing the cursor. Each character entered shifts existing data one column to the right and replaces a null character to the right of the cursor until no null characters remain in that field.

Alphanumeric and special character keys

Enters the character represented on the key into the field in the column containing the cursor.

Tab keys

Forward Tab	Positions the cursor to the beginning of the next field.
Back Tab	Positions the cursor to the beginning of the current field, or if the cursor is at the beginning of the current field, the back tab positions the cursor to the beginning of the previous field.

Editing Using Primary Editing Commands

This section describes the CA-MetaCOBOL+ PDF/CICS primary editing commands in alphabetical order. One or more primary editing commands can be entered in the command area of the screen at one time, and can be used in conjunction with line commands and PF keys.

Editing Using Line Commands

This section describes the CA-MetaCOBOL+ PDF/CICS line commands used to perform editing functions on individual lines or on a range of lines. Line commands are entered in the six-character sequence number field on the right or left of the screen and can overlay the sequence number anywhere in the sequence number field. Edit multipliers are numbers attached to line commands that specify how many times to perform the editing function.

The following rules apply to using line commands.

- Symbols entered in the sequence number field that are not valid line commands result in an error. For example,

  0**QQ**100

  results in an error, because QQ is not a valid line command.

- A line can only contain one line command; however, up to 15 lines within the data may contain line commands.

- Line commands can be used to edit a single line or a range of lines (two or more lines). For single lines, the line command is entered in the sequence number field on the line to be affected. For a range of lines, line commands mark the beginning and end of the range (included in the range are the lines containing the line commands).

- After the start of a range of lines has been specified with a line command, scrolling, as well as other line commands can be performed before the end of the range is specified. A PENDING message is displayed on the message line for the line command that is pending. This message remains until a range is fully specified or, in the case of COPY and MOVE, the destination is specified. A line command that is pending can be canceled by the RESET line command, deleting the command, or typing over the command. RESET also cancels all other pending line commands from the data.

- The following rules and examples describe how CA-MetaCOBOL+ distinguishes between multipliers attached to line commands and digits remaining in the sequence number field.

  CA-MetaCOBOL+'s first rule is: Ignore leading zeros and leading blanks.

  The second rule is: If the cursor is not in the sequence number field, ignore digits that have not changed (that is, digits that are in the same location and have the same value as before the line command was entered).

  For example, the following line commands delete line number 003100:

  D03100
  00**D**100
  003**D**00
  00310**D**

  The SET EDIT MULTIPLIER command determines where to place the multiplier for a line command: To the right or to the left. When SET EDIT MULTIPLIER equals left, the same rules hold. For example, the following line commands repeat line 003100 3 times:

  3R3100
  0**3R**100

  However, 00**3R**00 only repeats the line once; the 3 is ignored because it has not changed the digit 3 from the sequence number.

  When SET EDIT MULTIPLIER equals right, if the cursor is not in the sequence number field, the same rules hold. For example, the following line command entries are correct for repeating line 003100 4 times:

  R43100
  0**R4**100

However, line commands are cursor sensitive when SET EDIT MULTIPLIER equals right. That is, CA-MetaCOBOL+ interprets a line command to include all numbers between the line command and the cursor. In the following example, line 400 will be replicated 4 times. (The cursor position is underlined.)

```
000100
000200
000300
00R400
```

In the next example, line 400 will be replicated 40 times.

```
000100
000200
000300
00R400
```

Remember, line commands with the multiplier on the left are not cursor sensitive.

CA-MetaCOBOL+ works this way because of how the 3270 terminal works. When CA-MetaCOBOL+ receives the panel with your editing commands, the 3270 returns the contents of the sequence number fields and whether the fields changed. This introduces an ambiguity -- if the digits in the sequence number area following your line command are not changed, is that because the digits were not typed or were the same digits typed over what was already there? If the sequence number field contains 000400, was the 40 typed, or not?

CA-MetaCOBOL+ assumes that:

- If you typed a multiplier with a line command, your cursor is immediately to the right of the last digit you typed.

- If you did not type a multiplier, then your cursor is immediately to the right of the line command or not in the sequence number area.
  This usually works fine. However, you might, for example, type a 4 and inadvertently move the cursor one space to the right, giving you 40 repetitions of your line instead of 4. To help protect against such mistakes, CA-MetaCOBOL+ imposes a limit of 100 as a multiplier, and does not allow multipliers on the DELETE, COPY, and MOVE commands.

Order of Command Processing

Commands are processed in the following order:

1. PF1/13 (Help) or PF3/15 (Print Screen)
2. RESHOW
3. IGNORE
4. Updated text
5. RESET
6. Other line commands, in top-down order. Each is explained fully later in this section.

   D deletes one line of data. (DELETE Command)
   DD used in pairs to mark the start and end of a range of lines to be deleted. (DELETE Command)
   DT indicates the end of a range of lines to be deleted starting with the first line of the member. (DELETE Command)
   DB indicates the start of a range of lines to be deleted ending with the last line of the member. (DELETE Command)
   M specifies a single line to be moved. (MOVE Command)
   MM used in pairs to mark the start and end of a range of lines to be moved. (MOVE Command)
   R specifies a single line to be repeated. (REPEAT Command)
   RR used in pairs to mark the start and end of a range of lines to be repeated. (REPEAT Command)
   C specifies a single line to be copied. (COPY Command)
   CC used in pairs to mark the start and end of a range of lines to be copied. (COPY Command)
   A marks the start of a series of lines to be copied or moved.
   B marks the end of a series of lines to be copied or moved.
   ***** used to position a line at or near the top of a region. (INPUT LINE Command)
   I specifies the line before which null lines are to be inserted. (INPUT LINE Command)
   IB specifies the line after which null lines are to be inserted. (INPUT LINE Command)

7. Primary commands
8. Other PF keys

CHANGE Command

The CHANGE command is used to change occurrences of a character or a string of characters to another character or string of characters in EDIT MEMBER.

CHANGE can be restricted to an inclusive range of lines and/or columns with the LINES and COLUMNS operands, or by using the SET EDIT BOUNDS command. You can also change a specific number of occurrences of a character or a string.

Format:

```
CHANGE   [ ALL ]   [ FORWARD  ]   /string-a/string-b/
         [ n   ]   [ BACKWARD ]

        [LINE[S] start-line [end-line]]

        [ COLUMNS start-column [ MAX        ] ]
        [                      [ end-column ] ]
```

ALL
n
The number of occurrences of the string to be changed. ALL is the default.

FORWARD
Default. The search for string-a begins with the first line of the component or the specified start line and processes through subsequent lines.

BACKWARD
The search for string-a begins with the last line of the component or the specified end line and processes backward.

/
The character used to delimit the strings in the command. The same character must be used consistently within a command. Any special character can be used except: the currently defined command delimiter (installed default is a semicolon [;]) or comment character (installed default is a colon [:]), an asterisk (*), an at sign (@), a pound sign (#), or dollar sign ($).

string-a
A character or string of characters that is to be changed to string-b.

The string may not include the string delimiter character or the current command delimiter.

string-b
A character or string of characters that replaces the character or string of characters in string-a. An empty string-b deletes string-a. For example:

CHANGE /*string-a*//

The string may not include the string delimiter character or the current command delimiter.

start-line

The line at which the search for string-a begins. It can be specified as:

num	The sequence number of the start line.
TOP	Indicates the first line of the member. If a start line is not specified, TOP is the default.
CURSOR	Indicates the position at which the cursor is located.
offset	A position relative to the top line displayed: ***** The top line displayed. *** +n** The line 'n' number of lines below the top line displayed. *** -n** The line 'n' number of lines before the top line displayed.

Note: The keyword LINE can be omitted when you specify TOP, BOTTOM, CURSOR, or an offset.

end-line

The line at which the search for string-a ends. It can be specified as:

BOTTOM	Default. The last line of the member.
num	A sequence number.
CURSOR	Indicates the position at which the cursor is located.
offset	A position relative to the top line displayed: ***** The top line displayed. *** +n** The line 'n' number of lines below the top line displayed. *** -n** The line 'n' number of lines before the top line displayed.

start-column

The column at which the search for string-a begins.

end-column

The column at which the search for string-a ends.

MAX
Default. The farthest right-hand column.

Notes:

1) Because TOP and BOTTOM are the default limits on a range of lines, specifying,

   CHANGE /*string-a*/*string-b*/

   changes the top of the member through the bottom. Specifying,

   CHANGE /*string-a*/*string-b*/ LINE 200

   changes lines 200 through the bottom of the member.

   To change a single line by sequence number, specify the sequence number as the start line and end line. For example:

   CHANGE /*string-a*/*string-b*/ LINE 200 200

2) Specifying,

   CHANGE /*string-a*/*string-b*/ * *

   only changes the top line displayed.

3) Line and column clauses may be specified in either order. For example:

   CHANGE /DISPANEL/DISPNL/ LINES 165 175 COLUMNS 10 40

   or,

   CHANGE /DISPANEL/DISPNL/ COLUMNS 10 40 LINES 165 175

4) The column range can be the entire line, a range specified with the command SET EDIT BOUNDS, or an explicit range specified with CHANGE.

5) String-a must be entirely within the column range.

6) If string-b is longer than string-a, then string-b replaces string-a including trailing blanks up to the length of string-b. If there are not enough trailing blanks within the column range, the current setting of SET EDIT TRUNCATION controls the result as follows:

   If SET EDIT TRUNCATION is ON, data is truncated from the left starting at the end of the column range.

   If SET EDIT TRUNCATION is OFF, the change is not made.

For instance, in the following example "ABCbDEF" and "BCbbDEF" are in columns 1-6 (b represents a blank.) The command,

```
CHANGE /C/123/ COL 1 4
```

results in the following:

Before	After (Truncation Off)	After (Truncation On)
ABCbDEF	ABCbDEF	AB123EF
BCbbDEF	B123DEF	B123DEF

If the change cannot be made, a message is displayed and the affected line is brought to the top of the region. All changes prior to that line are kept, but no changes are made to the affected line.

The following example illustrates the use of the CHANGE command to change all occurrences of W#EMPLOYEE to W#EMPDATA without affecting the dataview name EMPLOYEE. The column range implicitly specifies columns 40 to Max.

```
 => CHANGE /EMPLOYEE/EMPDATA/ LINES 1200 2100 COLUMNS 40
 =>
 =>
Command....+....1....+....2....+....3....+....4....+....5....+....6....+....7
..
 000900    FOR EACH EMPLOYEE
 001000         WHERE where-condition
 001200       MOVE  EMPLOYEE.FLD1           TO   W#EMPLOYEE.FLD1
 001300       MOVE  EMPLOYEE.FLD2           TO   W#EMPLOYEE.FLD2
 001400       MOVE  EMPLOYEE.FLD3           TO   W#EMPLOYEE.FLD3
 001500       MOVE  EMPLOYEE.FLD4           TO   W#EMPLOYEE.FLD4
 001600       MOVE  EMPLOYEE.FLD5           TO   W#EMPLOYEE.FLD5
 001700       MOVE  EMPLOYEE.FLD6           TO   W#EMPLOYEE.FLD6
 001800       MOVE  EMPLOYEE.FLD7           TO   W#EMPLOYEE.FLD7
 001900       MOVE  EMPLOYEE.FLD8           TO   W#EMPLOYEE.FLD8
 002000       MOVE  EMPLOYEE.FLD9           TO   W#EMPLOYEE.FLD9
 002100       MOVE  EMPLOYEE.FLD10          TO   W#EMPLOYEE.FLD10
 002200    WHEN NONE
 002300       statements
 002400    ENDFOR
```

The results of the CHANGE command are shown next.

```
 =>
 =>
 =>
 1-ICEDITOR20I - Changed 10 field(s) in 10 record(s)

Command....+....1....+....2....+....3....+....4....+....5....+....6....+....7
..
 000900    FOR EACH EMPLOYEE
 001000         WHERE where-condition
 001200       MOVE  EMPLOYEE.FLD1           TO   W#EMPDATA.FLD1
 001300       MOVE  EMPLOYEE.FLD2           TO   W#EMPDATA.FLD2
 001400       MOVE  EMPLOYEE.FLD3           TO   W#EMPDATA.FLD3
 001500       MOVE  EMPLOYEE.FLD4           TO   W#EMPDATA.FLD4
 001600       MOVE  EMPLOYEE.FLD5           TO   W#EMPDATA.FLD5
 001700       MOVE  EMPLOYEE.FLD6           TO   W#EMPDATA.FLD6
 001800       MOVE  EMPLOYEE.FLD7           TO   W#EMPDATA.FLD7
 001900       MOVE  EMPLOYEE.FLD8           TO   W#EMPDATA.FLD8
 002000       MOVE  EMPLOYEE.FLD9           TO   W#EMPDATA.FLD9
 002100       MOVE  EMPLOYEE.FLD10          TO   W#EMPDATA.FLD10
 002200    WHEN NONE
 002300       statements
 002400    ENDFOR
```

The following example illustrates the use of the FIND and CHANGE commands to find all occurrences of ASK and change all occurrences of RLSASK to RLSDIS.

```
 => FIND ALL /ASK/
 =>

Command....+....1....+....2....+....3....+....4....+....5....+....6....+....7
..
 000100 <<ASK>> PROCEDURE
 000200    IF EXIST
 000300       MOVE $SPACES TO RLSASK.MSG
 000301       LOOP
 000400       TRANSMIT RLSASK CLEAR
 000401    WHILE $PANEL-ERROR
 000404       ENDLOOP
 000500    ELSE
 000600       TRANSMIT RLSASK
 000701       IF RLSASK.NUMBER = 99999
 000800          QUIT RUN
 000900       ENDIF
 001000       SET EXIST EQ TRUE
 001100    ENDIF
 001200    DO DIS-ASK
 001300 ENDPROC
```

Using the display of lines containing ASK, the CHANGE is applied to the applicable line range.

```
 => CHANGE /ASK/DIS/ LINES 300 701
 =>
 INCL /ASK/

Command....+....1....+....2....+....3....+....4....+....5....+....6....+....7
..
 000100 <<ASK>> PROCEDURE
 000300       MOVE $SPACES TO RLSASK.MSG
 000400       TRANSMIT RLSASK CLEAR
 000600       TRANSMIT RLSASK
 000701       IF RLSASK.NUMBER = 99999
 001200    DO DIS-ASK
```

The results of the CHANGE are underlined next.

```
 =>
 =>
 1-ICEDITOR20I - Changed 4 field(s) in 4 record(s)

Command....+....1....+....2....+....3....+....4....+....5....+....6...
.+....7..
 000100 <<ASK>> PROCEDURE
 000200    IF EXIST
 000300       MOVE $SPACES TO RLSDIS.MSG
 000301       LOOP
 000400       TRANSMIT RLSDIS CLEAR
 000401    WHILE $PANEL-ERROR
 000404       ENDLOOP
 000500    ELSE
 000600       TRANSMIT RLSDIS
 000701       IF RLSDIS.NUMBER = 99999
 000800          QUIT RUN
 000900       ENDIF
 001000       SET EXIST EQ TRUE
 001100    ENDIF
 001200    DO DIS-ASK
 001300 ENDPROC
```

CHECKPOINT Command

The CHECKPOINT command is used to establish a new stable point in an editing session to which the user is returned if a ROLLBACK command is specified.

Format:

```
CHECKPOINT
```

At the start of every editing session, CA-MetaCOBOL+ makes a temporary copy of the component being edited. This copy is used as the initial automatic checkpoint. (This temporary copy is used by the CHECKPOINT and ROLLBACK commands only. Editing activities affect the actual entity after the ENTER key is pressed.) A CHECKPOINT issued during an editing session replaces the existing temporary copy of the entity with a copy that reflects any changes made to the entity during the editing session up to the CHECKPOINT. This new copy is then the checkpoint.

Note that CHECKPOINT merely establishes a backup copy; to access that copy after editing activity, the ROLLBACK command must be used.

COPY Command

The COPY command is used to copy all or part of a member.

Format:

```
COPY  [ MEMBER name [ USER id ] ]

      [ start   [ end    ] ]
      [ TOP     [ BOTTOM ] ]

      { dest   }
      { TOP    }
      { BOTTOM }
```

If a member name is not specified, data is copied from one specified destination in the current member to another specified destination in the current member.

name
The 1- to 8-character name of the member from which data is to be copied.

id
The 1- to 3-character user ID. This is required only when copying from a member belonging to another user.

start
TOP
The line or the start of the range of lines to be copied.

start
The sequence number of the line to be copied or, if specified in conjunction with *end* or BOTTOM, the first line of a range of lines to be copied.

TOP
Indicates that the first line in the member is to be copied. If specified in conjunction with *end* or BOTTOM, it indicates that the first line of a range of lines to be copied is the top line.

Note: Only one value is specified, it is the destination of the lines to be copied. If two values are specified, they are the line to be copied and the destination line.

end
BOTTOM
The end of the range of lines to be copied.

end

The sequence number of the line marking the end of the range to be copied.

BOTTOM
Indicates that the last line in the member is the end of the range of lines to be copied.

Note: If only one value is specified, it is the destination of the lines to be copied. If two values are specified, they are the line to be copied and the destination line.

dest
TOP
BOTTOM
The destination of the lines to be copied. The destination is always within the component being edited in the current member.

dest
The sequence number of the line in the component of the current member after which the line or lines being copied are inserted.

TOP
Indicates that the destination of the line or lines to be copied is the top of the component of the current member.

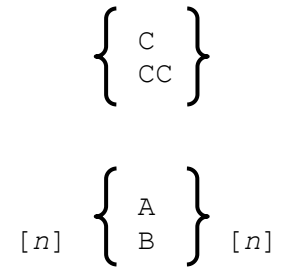
BOTTOM
Indicates that the destination of the line or lines to be copied is the bottom of the current member.

Note: If only one value is specified, it is the destination of the lines to be copied. If two values are specified, they are the line to be copied and the destination line.

Copy Line Command

The copy line commands insert the designated line or range of lines at a location marked as the destination of the copy function and also retain the line or lines at the original location. The format of the copy line commands is as follows.

Format:



C

Specifies a single line to be copied. A destination indicated by A or B, must be specified with each C line command. The destination can occur before or after the C line command.

CC

Used in pairs to indicate both the start and end of the range of lines to be copied. A destination indicated by the line command A or B must be specified with each delimited range of lines. The destination can occur before or after the range.

After specifying the start of a range with the CC line command, the region can be scrolled forward or backward and other line commands can be performed.

A

(After) Indicates that the destination for a line or a range of lines being copied is immediately after the line containing the A line command.

B

(Before) Indicates that the destination for a line or a range of lines being copied is immediately before the line containing the B line command.

n

(Optional) Specifies the number of times (from 1 through 100) the line or range of lines is inserted at the specified location. The SET EDIT MULTIPLIER command determines whether n is placed to the left or right. For example, if you specify,

```
SET EDIT MULTIPLIER RIGHT
```

then you specify:

```
A7 or B7
```

If you specify,

```
SET EDIT MULTIPLIER LEFT
```

then you specify:

```
7A or 7B
```

Note that only one destination (for a move or a copy) can be marked at any one time.

The inserted lines are assigned sequence numbers in increments of one starting from the previous existing line. When the number of inserted lines exceeds the number of integers between the sequence numbers of the existing lines, the sequence numbers on the lines following the insertion are changed. The following example illustrates copying a range of lines, 016800 to 017500, to a new location before line 018100.

```
....+....2....+....3....+....4....+....5....+....6....+....7... COMMAND
 SELECT                                                  016700
 WHEN $ENTER-KEY                                         CC6800
  FOR EMPLOYEE                                           016900
  WHERE EMPLOYEE.NUMBER = DISPANEL.NUMBER                017000
  MOVE DISPANEL TO EMPLOYEE BY NAME                      017100
  WHEN NONE                                              017200
  SET DISPANEL.MSG = 'EMPLOYEE DELETED'                  017300
  SET ATTR HIGHLIGHT TEMP ON DISPANEL.NUMBER             017400
  ENDFOR                                                 CC7500
  SET NEXT-PANEL = 'DISPANEL'                            017600
 WHEN $PF11                                              017700
  SET NEXT-PANEL = 'EMPMENU'                             017800
 WHEN $PF12                                              017900
  DO TERMINATE-SESSION                                   018000
 WHEN OTHER                                              B18100
  SET DISPANEL.MSG = 'UNABLE TO INTERPRET KEY ENTERED'   018200
  SET NEXT-PANEL = 'DISPANEL'                            018300
 ENDSEL                                                  018400
```

Note that the original lines retain their position while a copy of the lines is inserted before line 018100.

```
....+....2....+....3....+....4....+....5....+....6....+....7... COMMAND
 SELECT                                                          016700
 WHEN $ENTER-KEY                                                 016800
  FOR EMPLOYEE                                                   016900
  WHERE EMPLOYEE.NUMBER = DISPANEL.NUMBER                        017000
  MOVE DISPANEL TO EMPLOYEE BY NAME                              017100
  WHEN NONE                                                      017200
  SET DISPANEL.MSG = 'EMPLOYEE DELETED'                          017300
  SET ATTR HIGHLIGHT TEMP ON DISPANEL.NUMBER                     017400
  ENDFOR                                                         017500
  SET NEXT-PANEL = 'DISPANEL'                                    017600
 WHEN $PF11                                                      017700
  SET NEXT-PANEL = 'EMPMENU'                                     017800
 WHEN $PF12                                                      017900
  DO TERMINATE-SESSION                                           018000
 WHEN $ENTER-KEY                                                 018001
  FOR EMPLOYEE                                                   018002
  WHERE EMPLOYEE.NUMBER = DISPANEL.NUMBER                        018003
  MOVE DISPANEL TO EMPLOYEE BY NAME                              018004
  WHEN NONE                                                      018005
  SET DISPANEL.MSG = 'EMPLOYEE DELETED'                          018006
  SET ATTR HIGHLIGHT TEMP ON DISPANEL.NUMBER                     018007
  ENDFOR                                                         018008
 WHEN OTHER                                                      018100
  SET DISPANEL.MSG = 'UNABLE TO INTERPRET KEY ENTERED'           018200
  SET NEXT-PANEL = 'DISPANEL'                                    018300
 ENDSEL                                                          018400
```

DELETE Command

The DELETE command deletes a single line, a range of lines, or all of a member.

Format:

```
DELETE [ LINES ]  { nnn | TOP }  [ nnn | BOT ]
```

You must specify a beginning line, an ending line, or both. If only the beginning or only the ending line of a range is specified, only one line is deleted.

nnn
The sequence number of the line at which the delete begins or ends.

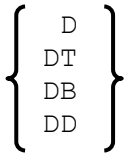
TOP
The first line of the member. If only the ending line of a range is specified, TOP is the default beginning line.

BOT
The last line of the member. If only the start line of a range is specified, BOT is the default ending line.

Delete Line Command

The delete line command deletes a single line or a range of lines.

Format:



D

Entered on a line of data to be deleted. Several D line commands can be specified on the same screen.

DT

Indicates the end of a range of lines to be deleted starting with the first line of the member.

DB

Indicates the start of a range of lines to be deleted ending with the last line of the member.

DD

Used in pairs to indicate the start and end of a range of lines to be deleted. The second occurrence of this command on the same or subsequent screen is interpreted as the end of the range.

After specifying the start of a range with the DD line command, the region can be scrolled forward or backward and other line commands can be performed.

Until the end of the range started by a DD line command is found, a PENDING message appears in the message line.

When interpreting line commands that mark ranges of lines to be deleted, the commands are matched into pairs from the top down, regardless of the order in which the line commands were entered.

The sequence numbers of deleted lines are also deleted (see the following example). To renumber the sequence numbers and eliminate gaps in numbering caused by deletions, refer to the RENUMBER primary editing command described later in this section.

The following example illustrates deleting a single line, 017400, and a range of lines, 017700 to 018000.

```
....+....2....+....3....+....4....+....5....+....6....+....7... COMMAND
<<PROCESS-PANEL>> PROC                                        016500
    TRANSMIT DISPANEL                                         016600
    SELECT                                                    016700
    WHEN $ENTER-KEY                                           016800
        FOR EMPLOYEE                                          016900
        WHERE EMPLOYEE.NUMBER = DISPANEL.NUMBER               017000
        MOVE DISPANEL TO EMPLOYEE BY NAME                     017100
        WHEN NONE                                             017200
        SET DISPANEL.MSG = 'EMPLOYEE DELETED'                 017300
        SET ATTR HIGHLIGHT TEMP ON DISPANEL.NUMBER            D17400
        ENDFOR                                                017500
        SET NEXT-PANEL = 'DISPANEL'                           017600
    WHEN $PF11                                                DD7700
        SET NEXT-PANEL = 'EMPMENU'                            017800
    WHEN $PF12                                                017900
        DO TERMINATE-SESSION                                  DD8000
    WHEN OTHER                                                018100
        SET DISPANEL.MSG = 'UNABLE TO INTERPRET KEY ENTERED'   018200
        SET NEXT-PANEL = 'DISPANEL'                           018300
    ENDSEL                                                    018400
```

Once the data is applied, the designated lines are deleted as shown below. Note that the sequence numbers on deleted lines are also deleted.

```
....+....2....+....3....+....4....+....5....+....6....+....7... COMMAND
<<PROCESS-PANEL>> PROC                                        016500
    TRANSMIT DISPANEL                                         016600
    SELECT                                                    016700
    WHEN $ENTER-KEY                                           016800
        FOR EMPLOYEE                                          016900
        WHERE EMPLOYEE.NUMBER = DISPANEL.NUMBER               017000
        MOVE DISPANEL TO EMPLOYEE BY NAME                     017100
        WHEN NONE                                             017200
        SET DISPANEL.MSG = 'EMPLOYEE DELETED'                 017300
        ENDFOR                                                017500
        SET NEXT-PANEL = 'DISPANEL'                           017600
    WHEN OTHER                                                018100
        SET DISPANEL.MSG = 'UNABLE TO INTERPRET KEY ENTERED'   018200
        SET NEXT-PANEL = 'DISPANEL'                           018300
    ENDSEL                                                    018400
```

The following illustrates deleting lines 016700 to 018400, using the DB command.

```
....+....2....+....3....+....4....+....5....+....6....+....7... COMMAND
<<PROCESS-PANEL>> PROC                                      016500
    TRANSMIT DISPANEL                                       016600
    SELECT                                                  DB6700
    WHEN $ENTER-KEY                                         016800
        FOR EMPLOYEE                                        016900
        WHERE EMPLOYEE.NUMBER = DISPANEL.NUMBER             017000
        MOVE DISPANEL TO EMPLOYEE BY NAME                   017100
        WHEN NONE                                           017200
        SET DISPANEL.MSG = 'EMPLOYEE DELETED'               017300
        SET ATTR HIGHLIGHT TEMP ON DISPANEL.NUMBER          017400
        ENDFOR                                              017500
        SET NEXT-PANEL = 'DISPANEL'                         017600
    WHEN $PF11                                              017700
        SET NEXT-PANEL = 'EMPMENU'                          017800
    WHEN $PF12                                              017900
        DO TERMINATE-SESSION                                018000
    WHEN OTHER                                              018100
        SET DISPANEL.MSG = 'UNABLE TO INTERPRET KEY ENTERED'    018200
        SET NEXT-PANEL = 'DISPANEL'                         018300
    ENDSEL                                                  018400
```

Once the DB command is applied, the designated lines are deleted as shown below.

```
....+....2....+....3....+....4....+....5....+....6....+....7... COMMAND
<<PROCESS-PANEL>> PROC                                      016500
 TRANSMIT DISPANEL                                          016600
```

DISPLAY ROLLBACK Command

The DISPLAY ROLLBACK command can be issued at any time during an edit session to display the backup copy maintained by CA-MetaCOBOL+ for the current entity only. You cannot edit or switch to another component while in DISPLAY ROLLBACK.

Format:

```
DISPLAY ROLLBACK
```

To return to your editing session, you can enter RETURN on the command line or press the RETURN PF key (PF2 or PF14). The RETURN command will return you back to an editing session after a DISPLAY ROLLBACK.

When the user signs on to CA-MetaCOBOL+ after a system failure, the user is returned to the last edit session at the time of the abend. At this point, the user receives an error message. The user can either view or edit the primary member or the backup member by issuing DISPLAY ROLLBACK.

EXCLUDE Command

The EXCLUDE command locates all lines which do not contain a string of characters and places those lines into the display region.

Use the COLUMNS operand or the SET EDIT BOUNDS command to restrict the search to an inclusive range of columns. You can also limit the range of lines to be searched.

Format:

```
EXCLUDE[ /string/ ] [LINE[S]  start-line  [end-line]]
        [ COLUMNS start-column  [ MAX ] ]
                               [ end-column ]
```

/
: The character used to delimit the search string. The same character must be used consistently within a command. Any special character can be used except: the currently defined command delimiter (installed default is a semicolon [;]) or comment character (installed default is a colon [:]), an asterisk (*), an at sign (@), a pound sign (#), or a dollar sign ($).

string
: A character or string of characters used in the search. If you do not specify a string, the string specified on the prior FIND, INCLUDE, EXCLUDE, FIRST, NEXT, PREVIOUS, or LAST command is used as a default value.

  The string may not include the string delimiter character or the current command delimiter.

start-line
: The line at which the search for string-a begins. It can be specified as:

num	The sequence number of the start line. When a single line is specified by a sequence number, the keyword LINE is required.
TOP	Indicates the first line of the member. If a start line is not specified, TOP is the default.
CURSOR	Indicates the position at which the cursor is located.
offset	A position relative to the top line displayed:
	***** The top line displayed.
	*** +n** The line 'n' number of lines below the top line displayed.
	*** -n** The line 'n' number of lines before the top line displayed.

Note: The keyword LINE can be omitted when you specify TOP, BOTTOM, CURSOR, or an offset.

end-line

The line at which the search for string-a ends. It can be specified as:

BOTTOM	Default. The last line of the member.
num	A sequence number.
CURSOR	Indicates the position at which the cursor is located.
offset	A position relative to the top line displayed:
	***** The top line displayed.
	*** +n** The line 'n' number of lines below the top line displayed.
	*** -n** The line 'n' number of lines before the top line displayed.

start-column

The column at which the search begins.

end-column

The column at which the search ends.

MAX

Default. The farthest right-hand column.

Note: Specifying lines or columns is optional. Line and column clauses may be specified in either order. For example,

```
EXCL /DISPANEL/ LINES 16500 17500 COLUMNS 10 40
```

or,

```
EXCL /DISPANEL/ COLUMNS 10 40 LINES 16500 17500
```

Examples:

The EXCLUDE command in the following example searches for lines which do not contain the string FOR in columns 20 through 50 (notice the blank following the R).

```
 => EXCL /FOR / COL 20 50

Command....+....1....+....2....+....3....+....4....+....5....+....6....+....7
..
 ====== ============================== T O P
================================
 000100         FOR EACH EMPLOYEE
 000200               WHERE STATE-ADDRESS = 'CA' OR 'IL'
 000300               AND NUMBER > 125 AND < 150
 000400               FOR PAYROLL
 000500                   WHERE NUMBER = EMPLOYEE.NUMBER
 000600                   AND NUMBER > 100 AND < 150
 000700                  SET SALARY = YTD-WAGES + YTD-COMMISSION - YTD-TAX
 000800                  SET STATE1 = 'CA'
 000900                  SET STATE2 = 'IL'
 001000               PRODUCE ADRM1
 001100          ENDFOR
 001200          FOR EACH EMPLOYEE
 001300              ORDERED BY STATE-ADDRESS CITY-ADDRESS EMPLOYEE.NAME
 001400               WHERE STATE-ADDRESS = 'CA' OR 'IL'
```

All lines which exclude the string are found and displayed in the region.

```
 =>

Command....+....1....+....2....+....3....+....4....+....5....+....6....+....7
..
 000200               WHERE STATE-ADDRESS = 'CA' OR 'IL'
 000300               AND NUMBER > 125 AND < 150
 000500                   WHERE NUMBER = EMPLOYEE.NUMBER
 000600                   AND NUMBER > 100 AND < 150
 000700                  SET SALARY = YTD-WAGES + YTD-COMMISSION - YTD-TAX
 000800                  SET STATE1 = 'CA'
 000900                  SET STATE2 = 'IL'
 001000               PRODUCE ADRM1
 001100               ENDFOR
 001200          ENDFOR
 001600               AND NUMBER > 125 AND < 150
 001800                   WHERE NUMBER = EMPLOYEE.NUMBER
 001900                   AND NUMBER > 100 AND < 150
```

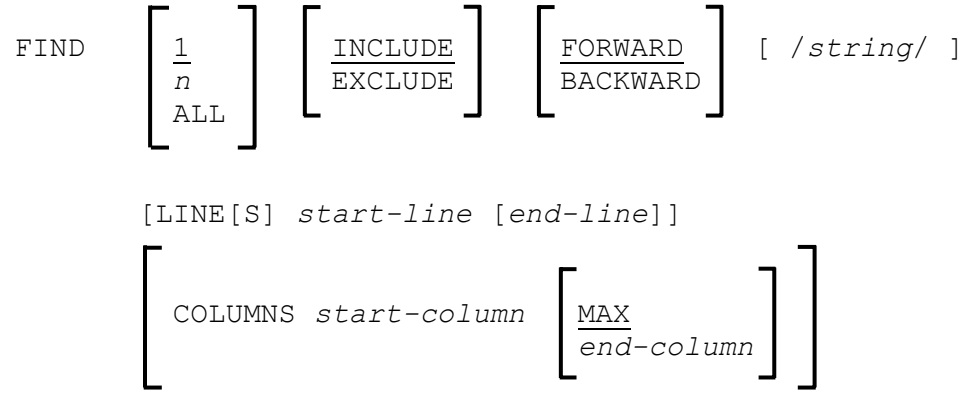
FIND Command

The FIND command searches for lines that either contain a string or for lines that do not contain a string.

FIND can be directed forward or backward and it can locate and display a specific number of lines. Use the COLUMNS operand or the SET EDIT BOUNDS command to restrict the FIND to an inclusive range of columns.

Once a line is found, FIND can be specified without the search string to locate the next line. You can specify any of the optional operands with subsequent FIND commands, without specifying the search string. If you do not specify any optional operands with subsequent FIND commands, the operands specified on the prior FIND will be used as default values.

Format:



1
n
ALL

Specifies the maximum number of lines to be found. 1 is the default. Specifying 1 causes the first line to be positioned at the top of the region with subsequent lines filling the region. Specify ALL to locate all lines. If you specify ALL or any value greater than 1, only the lines containing the string will be displayed. Use scrolling commands or PF keys to view all occurrences if the display exceeds the size of the region.

INCLUDE

Default. Searches for lines that contain the specified string.

EXCLUDE

Searches for lines that do not contain the specified string.

FORWARD
Default. Searches beginning with the first line and processing through subsequent lines.

BACKWARD
Searches beginning with the last line and processing backward.

/
The character used to delimit the search string. The same character must be used consistently within a command. Any special character can be used except: the currently defined command delimiter (installed default is a semicolon [;]) or comment character (installed default is a colon [:]), an asterisk (*), an at sign (@), a pound sign (#), or a dollar sign ($).

string
A character or string of characters used in the search. If you use the INCLUDE option, FIND searches for lines which contain this string. If you use the EXCLUDE operand, the command locates lines which do not contain this string. If you do not specify a string, the string specified on the prior FIND, INCLUDE, EXCLUDE, FIRST, NEXT, LAST, or PREVIOUS command is used as a default value.

The string may not include the string delimiter character or the current command delimiter.

start-line
The line at which the search for string-a begins. It can be specified as:

num	The sequence number of the start line. **Note:** When a single line is specified by a sequence number, the keyword LINE is required.
TOP	Indicates the first line of the member. If a start line is not specified, TOP is the default.
CURSOR	Indicates the position at which the cursor is located.
offset	A position relative to the top line displayed: ***** The top line displayed. *** +n** The line 'n' number of lines below the top line displayed. *** -n** The line 'n' number of lines before the top line displayed.

Note: The keyword LINE can be omitted when you specify TOP, BOTTOM, CURSOR, or an offset.

end-line
The line at which the search for string-a ends. It can be specified as:

BOTTOM	Default. The last line of the member.
num	A sequence number.
CURSOR	Indicates the position at which the cursor is located.
offset	A position relative to the top line displayed: ***** The top line displayed.

*** +n**	The line 'n' number of lines below the top line displayed.
*** -n**	The line 'n' number of lines before the top line displayed.

start-column
The column at which the search begins.

send-column
The column at which the search ends.

MAX
Default. The farthest right-hand column.

Note: Specifying lines or columns is optional. Line and column clauses may be specified in either order. For example,

FIND /DISPANEL/ LINES 16500 17500 COLUMNS 10 40

or,

FIND /DISPANEL/ COLUMNS 10 40 LINES 16500 17500

If you use the n operand to search for all occurrences or any number of occurrences greater than 1, FIND displays only those lines containing the string. You can use the following commands with this display:

FIND, FIRST, NEXT, LAST, or PREVIOUS (where n > 1)
IGNORE
SCROLL

The use of any command other than those listed above will cause this display to be replaced by a display showing the lines of the data in sequence.

Examples:

The FIND command in this example is looking for the string DISPANEL.NUMBER within a specific range.

```
=> FIND /DISPANEL.NUMBER/ LINES 16800 17500
=>
=>
....+....2....+....3....+....4....+....5....+....6....+....7... COMMAND
<<PROCESS-PANEL>> PROC                                      016500
 TRANSMIT DISPANEL                                          016600
    SELECT                                                  016700
    WHEN $ENTER-KEY                                         016800
        FOR EMPLOYEE                                        016900
        WHERE EMPLOYEE.NUMBER = DISPANEL.NUMBER             017000
        MOVE DISPANEL TO EMPLOYEE BY NAME                   017100
        WHEN NONE                                           017200
        SET DISPANEL.MSG = 'EMPLOYEE DELETED'               017300
        SET ATTR HIGHLIGHT TEMP ON DISPANEL.NUMBER          017400
        ENDFOR                                              017500
```

The line with the string is found and positioned at the top of the region, as shown below, with subsequent lines filling the region. The find operation is then continued by specifying the command FIND with no string or by pressing PF9/21 (FIND while in editing mode). Note that the same LINE and COLUMN range is also assumed.

```
=> FIND
=>
=>
....+....2....+....3....+....4....+....5....+....6....+....7... COMMAND
        WHERE EMPLOYEE.NUMBER = DISPANEL.NUMBER        017000
        MOVE DISPANEL TO EMPLOYEE BY NAME              017100
        WHEN NONE                                      017200
        SET DISPANEL.MSG = 'EMPLOYEE DELETED'          017300
        SET ATTR HIGHLIGHT TEMP ON DISPANEL.NUMBER     017400
        ENDFOR                                         017500
        SET NEXT-PANEL = 'DISPANEL'                    017600
    WHEN $PF11                                         017700
        SET NEXT-PANEL = 'EMPMENU'                     017800
    WHEN $PF12                                         017900
        DO TERMINATE-SESSION                           018000
```

The find operation continues, and the line containing the next occurrence of the string originally specified in the FIND command is positioned to the top of the region.

```
=>
=>
=>
....+....2....+....3....+....4....+....5....+....6....+....7... COMMAND
        SET ATTR HIGHLIGHT TEMP ON DISPANEL.NUMBER     017400
        ENDFOR                                         017500
        SET NEXT-PANEL = 'DISPANEL'                    017600
    WHEN $PF11                                         017700
        SET NEXT-PANEL = 'EMPMENU'                     017800
    WHEN $PF12                                         017900
        DO TERMINATE-SESSION                           018000
    WHEN $ENTER-KEY                                    018100
        FOR EMPLOYEE                                   018200
        WHERE EMPLOYEE.NUMBER = DISPANEL.NUMBER        018300
        MOVE DISPANEL TO EMPLOYEE BY NAME              018400
        WHEN NONE                                      018500
        SET DISPANEL.MSG = 'EMPLOYEE DELETED'          018600
        SET ATTR HIGHLIGHT TEMP ON DISPANEL.NUMBER     018700
        ENDFOR                                         018800
```

The FIND command in the following example uses the operands ALL and EXCLUDE to search for all lines which do not contain the string EMPLOYEE.

```
 => FIND ALL EXCLUDE /EMPLOYEE/
 =>
 ------------------------------------------------------------------------------
---
PDF/CICS: DISPLAY MEMBER     MEM FRC.EMPLRPT                 SYS: DOC  FILL-
IN
....+....1....+....2....+....3....+....4....+....5....+....6....+....7
COMMAND
 ================================ T O P ================================
======
         FOR EACH EMPLOYEE
000100
             ORDERED BY STATE-ADDRESS CITY-ADDRESS EMPLOYEE.NAME
000200
               WHERE STATE-ADDRESS = 'CA' OR 'IL'
000301
               AND NUMBER > 125 AND < 150
000302
               FOR PAYROLL
000400
                   WHERE NUMBER = EMPLOYEE.NUMBER
000401
                   AND NUMBER > 100 AND < 150
000402
                  SET SALARY = YTD-WAGES + YTD-COMMISSION - YTD-TAX
000500
                  SET STATE1 = 'CA'
000501
                  SET STATE2 = 'IL'
000502
               PRODUCE ADRM1
000601
               ENDFOR
000602
         ENDFOR
000700
         FOR EACH EMPLOYEE
000800
             ORDERED BY STATE-ADDRESS CITY-ADDRESS EMPLOYEE.NAME
000900
```

FIND displays all lines which do not contain the string. The status line shows the compare rule (EXCL) and the search string (EMPLOYEE).

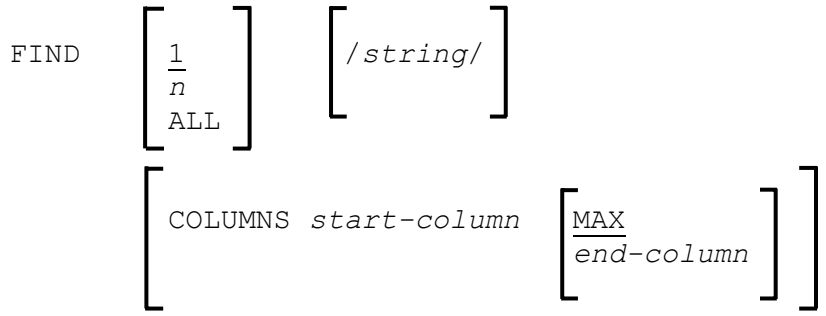
```
 =>
 =>
 ------------------------------------------------------------------------
---
PDF/CICS: DISPLAY MEMBER     MEM FRC.EMPLRPT                 SYS: DOC  FILL-
IN
EXCL /EMPLOYEE/
....+....1....+....2....+....3....+....4....+....5....+....6....+....7
COMMAND
                   SET SALARY = YTD-WAGES + YTD-COMMISSION - YTD-TAX
000500
                   SET STATE1 = 'CA'
000501
                   SET STATE2 = 'IL'
000502
                PRODUCE ADRM1
000601
                ENDFOR
000602
          ENDFOR
000700
                WHERE STATE-ADDRESS = 'CA' OR 'IL'
001000
                AND NUMBER > 125 AND < 150
001100
                FOR PAYROLL
001200
                    AND NUMBER > 100 AND < 150
001400
                   SET SALARY = YTD-WAGES + YTD-COMMISSION - YTD-TAX
001500
                   SET STATE1 = 'CA'
001600
                   SET STATE2 = 'IL'
001700
                PRODUCE ADRM1
001800
                ENDFOR
001900
          ENDFOR
002000
```

FIRST Command

The FIRST command locates the first occurrence of a string of characters. The search begins with the first line and processes forward through the last line.

Use the COLUMNS operand or the SET EDIT BOUNDS command to restrict the search to an inclusive range of columns. FIRST can be used to locate a specific number of occurrences.

Format:



1
n
ALL

Specifies the maximum number of occurrences of the string to be found. 1 is the default. Specifying 1 causes the first line containing the string to be positioned to the top of the region with subsequent lines filling the region. Use ALL to locate all occurrences of the string. If you specify ALL or any value greater than 1, only the lines containing the string will be displayed. Use scrolling commands or PF keys to view all occurrences if the display exceeds the size of the region.

/

The character used to delimit the search string. The same character must be used consistently within a command. Any special character can be used except: the currently defined command delimiter (installed default is a semicolon [;]) or comment character (installed default is a colon [:]), an asterisk (*), an at sign (@), a pound sign (#), or a dollar sign ($).

string

A character or string of characters that the command searches for. If you do not specify a string, the string specified on the prior FIND, INCLUDE, EXCLUDE, FIRST, NEXT, PREVIOUS, or LAST command is used as a default value.

The string may not include the string delimiter character or the current command delimiter.

start-column
The column at which the search for the string begins.

end-column
The column at which the search for the string ends.

MAX
Default. The farthest right-hand column.

Note: If you use the n operand to search for all occurrences or any number of occurrences greater than 1, FIRST displays only those lines containing the string. You can use the following commands without causing this display to be changed:

FIND, FIRST, NEXT, LAST, PREVIOUS (where n > 1)
IGNORE
SCROLL

The use of any command other than those listed above will cause this display to be replaced by a display showing the lines of data in sequence.

Example:

The FIRST command searches for the first three occurrences of the string NAME.

```
=> FIRST 3 /NAME/
=>
=>
------------------------------------------------------------------------------
--
PDF/CICS: DISPLAY MEMBER     MEM FRC.EMPLRPX                 SYS: DOC  FILL-
IN
Command....+....1....+....2....+....3....+....4....+....5....+....6....+....7
..
====== ================================ T O P
=================================
000100         FOR EACH EMPLOYEE
000200                WHERE STATE-ADDRESS = 'CA' OR 'IL'
000300                AND NUMBER > 125 AND < 150
000400                FOR PAYROLL
000500                    WHERE NUMBER = EMPLOYEE.NUMBER
000600                    AND NUMBER > 100 AND < 150
000700                   SET SALARY = YTD-WAGES + YTD-COMMISSION - YTD-TAX
000800                   SET STATE1 = 'CA'
000900                   SET STATE2 = 'IL'
001000                PRODUCE ADRM1
001100                ENDFOR
001200          ENDFOR
001300          FOR EACH EMPLOYEE
001400              ORDERED BY STATE-ADDRESS CITY-ADDRESS EMPLOYEE.NAME
001500                WHERE STATE-ADDRESS = 'CA' OR 'IL'
```

The only line that contains the string is displayed.

```
=>
=>
=>
------------------------------------------------------------------------------
--
PDF/CICS: DISPLAY MEMBER     MEM FRC.EMPLRPX                 SYS: DOC  FILL-
IN
INCL /NAME/
Command....+....1....+....2....+....3....+....4....+....5....+....6....+....7
..
001400              ORDERED BY STATE-ADDRESS CITY-ADDRESS EMPLOYEE.NAME
```

IGNORE Line Command

The IGNORE line command refreshes only the line containing it. The contents of the line are restored as they were at the start of the current transaction. Any change made to the data on that line is not applied.

Format:

```
IGNORE
```

In the following example, changes are made to the data in the region in the form of comments being added. The IGNORE line command is then entered on a line where a comment has been typed.

```
....+....2....+....3....+....4....+....5....+....6....+....7... COMMAND
<<PROCESS-PANEL>> PROC                                    017400
   TRANSMIT DISPANEL                                      017500
   SELECT  :Include additional PFkey                      017600
   WHEN $PF1  :selections                                 017700
       SET NEXT-PANEL = 'EMPMENU'  :and the CLEAR key  IGNORE
   WHEN $PF12                                             017900
       DO TERMINATE-SESSION                               018000
   WHEN OTHER                                             018100
       SET DISPANEL.MSG = 'UNABLE TO INTERPRET KEY ENTERED'    018200
       SET NEXT-PANEL = 'DISPANEL'                        018300
   ENDSEL                                                 018400
```

The following example shows that the comment entered on the line containing the IGNORE line command was not applied.

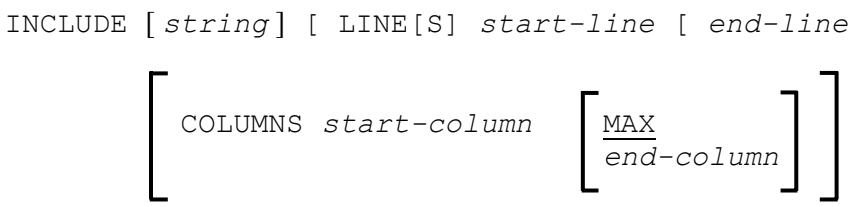
```
....+....2....+....3....+....4....+....5....+....6....+....7... COMMAND
<<PROCESS-PANEL>> PROC                                    017400
   TRANSMIT DISPANEL                                      017500
   SELECT    :Include additional PFkey                    017600
   WHEN $PF11    :selections                              017700
       SET NEXT-PANEL = 'EMPMENU'                         017800
   WHEN $PF12                                             017900
       DO TERMINATE-SESSION                               018000
   WHEN OTHER                                             018100
       SET DISPANEL.MSG = 'UNABLE TO INTERPRET KEY ENTERED'    018200
       SET NEXT-PANEL = 'DISPANEL'                        018300
   ENDSEL                                                 018400
```

INCLUDE Command

The INCLUDE command locates all lines that contain a string of characters and places those lines in the display region.

Use the COLUMNS operand or the SET EDIT BOUNDS command to restrict the search to an inclusive range of columns. You can also limit the range of lines to be searched.

Format:



/

The character used to delimit the search string. The same character must be used consistently within a command. Any special character can be used except: the currently defined command delimiter (installed default is a semicolon [;]) or comment character (installed default is a colon [:]), an asterisk (*), an at sign (@), a pound sign (#), or a dollar sign ($).

string

A character or string of characters that the command searches for. If you do not specify a string, the string specified on the prior FIND, INCLUDE, EXCLUDE, FIRST, NEXT, PREVIOUS, or LAST command is used as a default value.

The string may not include the string delimiter character or the current command delimiter.

start-line

The line at which the search for string-a begins. It can be specified as:

num	The sequence number of the start line. When a single line is specified by a sequence number, the keyword LINE is required.
TOP	Indicates the first line of the member. If a start line is not specified, TOP is the default.
CURSOR	Indicates the position at which the cursor is located.
offset	A position relative to the top line displayed:
	***** The top line displayed.
	*** +n** The line 'n' number of lines below the top line displayed.
	*** -n** The line 'n' number of lines before the top line displayed.

Note: The keyword LINE can be omitted when you specify TOP, BOTTOM, CURSOR, or an offset.

end-line

The line at which the search for string-a ends. It can be specified as:

BOTTOM	Default. The last line of the member.
num	A sequence number.
CURSOR	Indicates the position at which the cursor is located.
offset	A position relative to the top line displayed:
	***** The top line displayed.
	*** +n** The line 'n' number of lines below the top line displayed.
	*** -n** The line 'n' number of lines before the top line displayed.

start-column

The column at which the search for the string begins.

end-column

The column at which the search for the string ends.

MAX

Default. The farthest right-hand column.

Note: Specifying lines or columns is optional. Line and column clauses may be specified in either order. For example,

```
INCL /DISPANEL/ LINES 16500 17500 COLUMNS 10 40
```

or,

```
INCL /DISPANEL/ COLUMNS 10 40 LINES 16500 17500
```

Examples:

In the following example, INCLUDE searches for the string NUMBER, from the cursor position through the last line of the member.

```
=> INCL /NUMBER/ CURSOR BOTTOM
=>
=>
Command....+....1....+....2....+....3....+....4....+....5....+....6....+....7
..
000100         FOR EACH EMPLOYEE
000200                WHERE STATE-ADDRESS = 'CA' OR 'IL'
000300                AND NUMBER > 125 AND < 150
000400 _              FOR PAYROLL
000500                    WHERE NUMBER = EMPLOYEE.NUMBER
000600                    AND NUMBER > 100 AND < 150
000700                  SET SALARY = YTD-WAGES + YTD-COMMISSION - YTD-TAX
000800                  SET STATE1 = 'CA'
000900                  SET STATE2 = 'IL'
001000                PRODUCE ADRM1
001100                ENDFOR
001200          ENDFOR
001300          FOR EACH EMPLOYEE
001400              ORDERED BY STATE-ADDRESS CITY-ADDRESS EMPLOYEE.NAME
001500                WHERE STATE-ADDRESS = 'CA' OR 'IL'
```

INCLUDE displays all lines in this range containing the string.

```
=>
=>
=>
--------------------------------------------------------------------------------
--
PDF/CICS: DISPLAY MEMBER     MEM FRC.EMPLRPY                 SYS: DOC  FILL-
IN
INCL /NUMBER/
Command....+....1....+....2....+....3....+....4....+....5....+....6....+....7
..
000500                    WHERE NUMBER = EMPLOYEE.NUMBER
000600                    AND NUMBER > 100 AND < 150
001600                AND NUMBER > 125 AND < 150
001800                    WHERE NUMBER = EMPLOYEE.NUMBER
001900                    AND NUMBER > 100 AND < 150
```

INPUT Command

The INPUT command is used to insert null lines (open a window) into the data based on the cursor position.

Format:

```
        ┌                 ┐
INPUT   │ [CONTEXT] n     │
        │  WINDOW m       │
        └                 ┘
```

n

Indicates the number of required context lines (lines retained on the screen for reference). n lines are displayed at the bottom of the screen if the cursor remains in the command area; or n lines are placed at the top and bottom of the screen if the cursor is placed in the region. The rest of the screen is opened up for input.

The default number of context lines can be set by the SET EDIT CONTEXT command, described in the previous section.

m

Indicates the number of null lines to be inserted. The window is opened at the top of the screen if the cursor remains in the command area; or the window is opened following the line with the cursor if the cursor is placed in the region.

When possible, CA-MetaCOBOL+ scrolls the window forward or backward, to center the window in the region. When the value of n exceeds the number of lines left in the region, minus context lines, the maximum number of lines that can fit between the context lines is inserted.

In the following example, a window 5 lines long is requested in the region starting below the line with the cursor (017600).

```
=>INPUT WINDOW 5
=>
....+....2....+....3....+....4....+....5....+....6....+....7... COMMAND
<<PROCESS-PANEL>> PROC                                        016500
   TRANSMIT DISPANEL                                          016600
   SELECT                                                     016700
   WHEN $ENTER-KEY                                            016800
       FOR EMPLOYEE                                           016900
       WHERE EMPLOYEE.NUMBER = DISPANEL.NUMBER                017000
       MOVE DISPANEL TO EMPLOYEE BY NAME                      017100
       WHEN NONE                                              017200
       SET DISPANEL.MSG = 'EMPLOYEE DELETED'                  017300
       SET ATTR HIGHLIGHT TEMP ON DISPANEL.NUMBER             017400
       ENDFOR                                                 017500
       SET NEXT-PANEL = 'DISPANEL'                            017600
   WHEN $PF12                                                 017700
       DO TERMINATE-SESSION                                   017800
   WHEN OTHER                                                 017900
       SET DISPANEL.MSG = 'UNABLE TO INTERPRET KEY ENTERED'    018000
       SET NEXT-PANEL = 'DISPANEL'                            018100
   ENDSEL                                                     018200
```

After applying the data, five null lines are inserted as shown in the next example. Note that the window is centered in the region.

```
=>
=>
=>
....+....2....+....3....+....4....+....5....+....6....+....7... COMMAND
       WHERE EMPLOYEE.NUMBER = DISPANEL.NUMBER                017000
       MOVE DISPANEL TO EMPLOYEE BY NAME                      017100
       WHEN NONE                                              017200
       SET DISPANEL.MSG = 'EMPLOYEE DELETED'                  017300
       SET ATTR HIGHLIGHT TEMP ON DISPANEL.NUMBER             017400
       ENDFOR                                                 017500
       SET NEXT-PANEL = 'DISPANEL'                            017600
                                                              ......
                                                              ......
                                                              ......
                                                              ......
                                                              ......
   WHEN $PF12                                                 017700
       DO TERMINATE-SESSION                                   017800
   WHEN OTHER                                                 017900
       SET DISPANEL.MSG = 'UNABLE TO INTERPRET KEY ENTERED'    018000
       SET NEXT-PANEL = 'DISPANEL'                            018100
   ENDSEL                                                     018200
```

Input Line Command

The input line command opens a window of null lines for insertion of data. You can specify how many lines are to be inserted.

Format:

```
[n]I     or  I[n]

[n]IB    or IB[n]
```

The SET EDIT MULTIPLIER command determines whether n is placed on the left or right.

I

Specified on the line after which null lines are to be inserted.

IB

Specified on the line before which null lines are to be inserted.

n

Optional. The number of null lines (1 through 100) to be inserted. If n is not specified, one null line is inserted.

If the value of n exceeds the number of lines remaining in the region, the maximum number of lines that can fit are inserted.

Any unused null lines are removed when the data is applied (when the ENTER key or any PF key is pressed).

Sequence numbers do not appear on blank lines inserted in a window until data is applied. The inserted lines are assigned sequence numbers in increments of one starting from the previous existing line. When the number of inserted lines exceeds the number of integers between the sequence numbers of the existing lines, the sequence numbers on the lines following the insertion are changed.

To renumber sequence numbers refer to the RENUMBER primary editing command.

The following examples illustrate the use of the input line command and the numbering of inserted lines.

In the following example, the input line command, 5I, is entered in line 017600.

```
....+....2....+....3....+....4....+....5....+....6....+....7... COMMAND
<<PROCESS-PANEL>> PROC                                        016500
   TRANSMIT DISPANEL                                          016600
   SELECT                                                     016700
   WHEN $ENTER-KEY                                            016800
       FOR EMPLOYEE                                           016900
       WHERE EMPLOYEE.NUMBER = DISPANEL.NUMBER                017000
       MOVE DISPANEL TO EMPLOYEE BY NAME                      017100
       WHEN NONE                                              017200
       SET DISPANEL.MSG = 'EMPLOYEE DELETED'                  017300
       SET ATTR HIGHLIGHT TEMP ON DISPANEL.NUMBER             017400
       ENDFOR                                                 017500
       SET NEXT-PANEL = 'DISPANEL'                            5I7600
   WHEN $PF12                                                 017700
       DO TERMINATE-SESSION                                   017800
   WHEN OTHER                                                 017900
       SET DISPANEL.MSG = 'UNABLE TO INTERPRET KEY ENTERED'    018000
       SET NEXT-PANEL = 'DISPANEL'                            018100
   ENDSEL                                                     018200
```

After the ENTER key is pressed, five null lines are inserted as shown in the next example. The two lines in bold print represent newly entered data. Note that neither the null lines nor the newly added lines have sequence numbers.

```
....+....2....+....3....+....4....+....5....+....6....+....7... COMMAND
<<PROCESS-PANEL>> PROC                                        016500
   TRANSMIT DISPANEL                                          016600
   SELECT                                                     016700
   WHEN $ENTER-KEY                                            016800
       FOR EMPLOYEE                                           016900
       WHERE EMPLOYEE.NUMBER = DISPANEL.NUMBER                017000
       MOVE DISPANEL TO EMPLOYEE BY NAME                      017100
       WHEN NONE                                              017200
       SET DISPANEL.MSG = 'EMPLOYEE DELETED'                  017300
       SET ATTR HIGHLIGHT TEMP ON DISPANEL.NUMBER             017400
       ENDFOR                                                 017500
       SET NEXT-PANEL = 'DISPANEL'                            017600
   WHEN  $PF11                                                ......
      SET NEXT-PANEL = 'EMPMENU                               ......
                                                              ......
                                                              ......
                                                              ......
   WHEN $PF12                                                 017700
```

After pressing the ENTER key, the unused null lines are removed and the new lines are inserted with sequence numbers.

```
....+....2....+....3....+....4....+....5....+....6....+....7... COMMAND
<<PROCESS-PANEL>> PROC                                        016500
   TRANSMIT DISPANEL                                          016600
   SELECT                                                     016700
   WHEN $ENTER-KEY                                            016800
       FOR EMPLOYEE                                           016900
       WHERE EMPLOYEE.NUMBER = DISPANEL.NUMBER                017000
       MOVE DISPANEL TO EMPLOYEE BY NAME                      017100
       WHEN NONE                                              017200
       SET DISPANEL.MSG = 'EMPLOYEE DELETED'                  017300
       SET ATTR HIGHLIGHT TEMP ON DISPANEL.NUMBER             017400
       ENDFOR                                                 017500
       SET NEXT-PANEL = 'DISPANEL'                            017601
   WHEN $PF11                                                 017602
      SET NEXT-PANEL = 'EMPMENU'                              017600
  WHEN $PF12                                                  017700
       DO TERMINATE-SESSION                                   017800
   WHEN OTHER                                                 017900
       SET DISPANEL.MSG = 'UNABLE TO INTERPRET KEY ENTERED'    018000
       SET NEXT-PANEL = 'DISPANEL'                            018100
   ENDSEL                                                     018200
```

In the following example, the input-before line command, IB2, is entered in line 017700.

```
....+....2....+....3....+....4....+....5....+....6....+....7... COMMAND
<<PROCESS-PANEL>> PROC                                        016500
   TRANSMIT DISPANEL                                          016600
   SELECT                                                     016700
   WHEN $ENTER-KEY                                            016800
       FOR EMPLOYEE                                           016900
       WHERE EMPLOYEE.NUMBER = DISPANEL.NUMBER                017000
       MOVE DISPANEL TO EMPLOYEE BY NAME                      017100
       WHEN NONE                                              017200
       SET DISPANEL.MSG = 'EMPLOYEE DELETED'                  017300
       SET ATTR HIGHLIGHT TEMP ON DISPANEL.NUMBER             017400
       ENDFOR                                                 017500
       SET NEXT-PANEL = 'DISPANEL'                            017600
   WHEN $PF12                                                 IB2700
       DO TERMINATE-SESSION                                   017800
   WHEN OTHER                                                 017900
       SET DISPANEL.MSG = 'UNABLE TO INTERPRET KEY ENTERED'    018000
       SET NEXT-PANEL = 'DISPANEL'                            018100
   ENDSEL                                                     018200
```

After the ENTER key is pressed, two null lines are inserted as shown in the next example. The two underlined lines represent newly entered data. Note that the newly added lines do not have sequence numbers.

```
....+....2....+....3....+....4....+....5....+....6....+....7... COMMAND
<<PROCESS-PANEL>> PROC                                          016500
   TRANSMIT DISPANEL                                            016600
   SELECT                                                       016700
   WHEN $ENTER-KEY                                              016800
       FOR EMPLOYEE                                             016900
       WHERE EMPLOYEE.NUMBER = DISPANEL.NUMBER                  017000
       MOVE DISPANEL TO EMPLOYEE BY NAME                        017100
       WHEN NONE                                                017200
       SET DISPANEL.MSG = 'EMPLOYEE DELETED'                    017300
       SET ATTR HIGHLIGHT TEMP ON DISPANEL.NUMBER               017400
       ENDFOR                                                   017500
       SET NEXT-PANEL = 'DISPANEL'                              017600
        WHEN  $PF11                                             ......
     SET NEXT-PANEL = 'EMPMENU'                                 ......
  WHEN $PF12                                                    017700
```

After pressing the ENTER key, the new lines are inserted with sequence numbers.

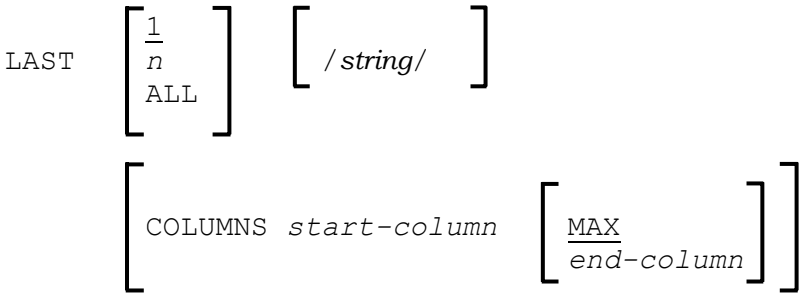
```
....+....2....+....3....+....4....+....5....+....6....+....7... COMMAND
   <<PROCESS-PANEL>> PROC                                       016500
   TRANSMIT DISPANEL                                            016600
   SELECT                                                       016700
   WHEN $ENTER-KEY                                              016800
       FOR EMPLOYEE                                             016900
       WHERE EMPLOYEE.NUMBER = DISPANEL.NUMBER                  017000
       MOVE DISPANEL TO EMPLOYEE BY NAME                        017100
       WHEN NONE                                                017200
       SET DISPANEL.MSG = 'EMPLOYEE DELETED'                    017300
       SET ATTR HIGHLIGHT TEMP ON DISPANEL.NUMBER               017400
       ENDFOR                                                   017500
       SET NEXT-PANEL = 'DISPANEL'                              017600
      WHEN $PF11                                                017601
          SET NEXT-PANEL = 'EMPMENU'                            017602
  WHEN $PF12                                                    017700
       DO TERMINATE-SESSION                                     017800
   WHEN OTHER                                                   017900
       SET DISPANEL.MSG = 'UNABLE TO INTERPRET KEY ENTERED'     018000
       SET NEXT-PANEL = 'DISPANEL'                              018100
   ENDSEL                                                       018200
```

LAST Command

The LAST command locates the last occurrence of a string of characters. The search begins with the last line and processes backward.

Use the COLUMNS operand or the SET EDIT BOUNDS command to restrict the search to an inclusive range of columns. LAST can also locate a specific number of occurrences.

Format:



1
n
ALL

The maximum number of occurrences of the string to be found. 1 is the default. Specifying 1 causes the last line containing the string to be positioned at the top of the region with subsequent lines filling the region. Specify ALL to locate all occurrences of the string. If you specify ALL or any value greater than 1, only the lines containing the string will be displayed. Use scrolling commands or PF keys to view all occurrences if the display exceeds the size of the region.

/

The character used to delimit the search string. The same character must be used consistently within a command. Any special character can be used except: the currently defined command delimiter (installed default is a semicolon [;]) or comment character (installed default is a colon [:]), an asterisk (*), an at sign (@), a pound sign (#), or a dollar sign ($).

string

A character or string of characters that the command searches for. If you do not specify a string, the string specified on the prior FIND, INCLUDE, EXCLUDE, FIRST, NEXT, PREVIOUS, or LAST command is used as a default value.

The string may not include the string delimiter character or the current command delimiter.

start-column
The column at which the search for the string begins.

end-column
The column at which the search for the string ends.

MAX
Default. The farthest right-hand column.

Note: If you use the n operand to search for all occurrences or any number of occurrences greater than 1, LAST displays only those lines containing the string. You can use the following commands without causing this display to be changed:

FIND, FIRST, LAST, NEXT, or PREVIOUS (where n > 1)
IGNORE
SCROLL

The use of any command other than those listed above will cause this display to be replaced by a display showing the lines of data in sequence.

MOVE Command

The MOVE command inserts the designated line or range of lines at a location marked as the destination of the move and deletes the line or lines at the original location.

Format:

```
MOVE  [LINES]  { nnn }  [ nnn ]  { nnn }
               { TOP }   [ BOT ]  { TOP }
```

nnn
The sequence number of the line at which the move begins or ends.

TOP
(Default). The first line of the member.

BOT
(Default). The last line of the member.

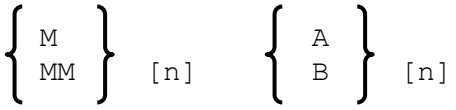
Notes: After the MOVE, new sequence numbers are assigned to the new locations.

The command MOVE LINES TOP BOTTOM is prohibited.

Move Line Command

The MOVE LINE command inserts the designated line or range of lines at a location marked as the destination of the move and deletes the line or lines at the original location. This line command also allows the line or range of lines moved to be repeated a specified number of times. The format of the move line command is as follows.

Format:



M

Specifies a single line to be moved. A destination, indicated by A or B, must be specified with each M line command. The destination can occur before or after the M line command.

MM

Used in pairs to indicate both the start and end of the range of lines to be moved. A destination, indicated by the line command A or B, must be specified with each delimited range of lines. The destination can occur before or after the range. After specifying the start of a range with the MM line command, the region can be scrolled forward or backward and other line commands can be performed with the following exception: while a move is pending, no other moves or copies can be specified.

A

Indicates that the destination for a line or a range of lines being moved is immediately after the line containing the A line command.

B

Indicates that the destination for a line or a range of lines being moved is immediately before the line containing the B line command.

n

Optionally, specifies the number of times (between 1 and 100) the line or range of lines is inserted at the specified destination. The SET EDIT MULTIPLIER command determines whether n is placed on the left or right. For example, if you specify,

```
SET EDIT MULTIPLIER RIGHT
```

then you specify,

```
A7 or B7
```

If you specify,

```
SET EDIT MULTIPLIER LEFT
```

then you specify,

```
7A or 7B
```

Note, that only one destination (for a move or a copy) can be marked at any one time.

The inserted lines are assigned sequence numbers in increments of one starting from the previous existing line. When the number of inserted lines exceeds the number of integers between the sequence numbers of the existing lines, the sequence numbers on the lines following the insertion are changed. The following example illustrates moving a range of lines, 016800 through 017600, before line 018100. Notice the PENDING message caused by scrolling the region forward after the start of the range is marked.

```
=> SCROLL FORWARD
=>
=>
....+....2....+....3....+....4....+....5....+....6....+....7... COMMAND
<<PROCESS-PANEL>> PROC                                          016500
TRANSMIT DISPANEL                                               016600
   SELECT                                                       016700
   WHEN $ENTER-KEY                                              MM6800
        FOR EMPLOYEE                                            016900
        WHERE EMPLOYEE.NUMBER = DISPANEL.NUMBER                 017000
        MOVE DISPANEL TO EMPLOYEE BY NAME                       017100
        WHEN NONE                                               017200
        SET DISPANEL.MSG = 'EMPLOYEE DELETED'                   017300
        SET ATTR HIGHLIGHT TEMP ON DISPANEL.NUMBER              017400
```

```
PENDING: MM (16800)
....+....2....+....3....+....4....+....5....+....6....+....7... COMMAND
       ENDFOR                                                   017500
       SET NEXT-PANEL = 'DISPANEL'                              MM7600
   WHEN $PF11                                                   017700
       SET NEXT-PANEL = 'EMPMENU'                               017800
   WHEN $PF12                                                   017900
       DO TERMINATE-SESSION                                     018000
   WHEN OTHER                                                   B18100
       SET DISPANEL.MSG = 'UNABLE TO INTERPRET KEY ENTERED'     018200
       SET NEXT-PANEL = 'DISPANEL'                              018300
   ENDSEL                                                       018400
```

After the data is applied, the original sequence numbers of the lines being moved are deleted and the lines are inserted in the new location with NEW sequence numbers.

```
....+....2....+....3....+....4....+....5....+....6....+....7... COMMAND
<<PROCESS-PANEL>> PROC                                          016500
    TRANSMIT DISPANEL                                           016600
    SELECT                                                      016700
    WHEN $PF11                                                  017700
        SET NEXT-PANEL = 'EMPMENU'                              017800
    WHEN $PF12                                                  017900
        DO TERMINATE-SESSION                                    018000
    WHEN $ENTER-KEY                                             018001
        FOR EMPLOYEE                                            018002
        WHERE EMPLOYEE.NUMBER = DISPANEL.NUMBER                 018003
        MOVE DISPANEL TO EMPLOYEE BY NAME                       018004
        WHEN NONE                                               018005
        SET DISPANEL.MSG = 'EMPLOYEE DELETED'                   018006
        SET ATTR HIGHLIGHT TEMP ON DISPANEL.NUMBER              018007
        ENDFOR                                                  018008
        SET NEXT-PANEL = 'DISPANEL'                             018009
  WHEN OTHER                                                    018100
        SET DISPANEL.MSG = 'UNABLE TO INTERPRET KEY ENTERED'    018200
        SET NEXT-PANEL = 'DISPANEL'                             018300
    ENDSEL                                                      018400
```

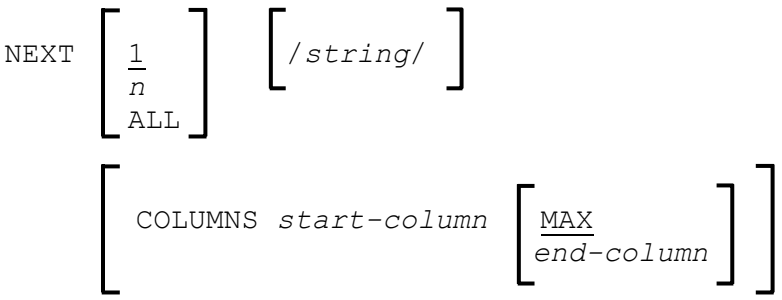
NEXT Command

The NEXT command locates the next occurrence of a string of characters. The search begins with the second line of the display and processes forward.

Use the COLUMNS operand or the SET EDIT BOUNDS command to restrict the search to an inclusive range of columns. NEXT can also locate a specific number of occurrences.

This command has the following syntax:

Format:



1
n
ALL

The maximum number of occurrences of the string to be found. 1 is the default. Specifying 1 causes the next line containing the string to be positioned at the top of the region with subsequent lines filling the region. Specify ALL to locate all occurrences of the string. If you specify ALL or any value greater than 1, only the lines containing the string will be displayed. Use scrolling commands or PF keys to view all occurrences if the display exceeds the size of the region.

/

The character used to delimit the search string. The same character must be used consistently within a command. Any special character can be used except: the currently defined command delimiter (installed default is a semicolon [;]) or comment character (installed default is a colon [:]), an asterisk (*), an at sign (@), a pound sign (#), or a dollar sign ($).

string

A character or string of characters that the command searches for. If you do not specify a string, the string specified on the prior FIND, INCLUDE, EXCLUDE, FIRST, NEXT, PREVIOUS, or LAST command is used as a default value.

The string may not include the string delimiter character or the current command delimiter.

start-column
The column at which the search for the string begins.

end-column
The column at which the search for the string ends.

MAX
Default. The farthest right-hand column.

Note: If the n operand is specified as ALL or any value greater than 1, only the lines which contain the string will be displayed. You can use the following commands on this display:

FIND, FIRST, NEXT, PREVIOUS or LAST (where n > 1)
IGNORE
SCROLL

The use of any command other than those listed above will cause this display to be replaced by a display showing the lines of data in sequence.

POSITION Command

The POSITION command is used to position a specific line at the top of the region.

Format:

```
POSITION n
```

n

For members, n is the sequence number of the line to be positioned at the top of the region.

For the panel field summary table or the panel field picture table, n is the field number; e.g., "POSITION 7" brings the row for field 7 to the top of the region in the layout portion and the entry for field 7 to the top of the table portion of the region.

The following example positions line 16900 at the top of the region.

```
=> POSITION 16900
....+....2....+....3....+....4....+....5....+....6....+....7... COMMAND
<<PROCESS-PANEL>> PROC                                        016500
    TRANSMIT DISPANEL                                         016600
    SELECT                                                    016700
    WHEN $ENTER-KEY                                           016800
        FOR EMPLOYEE                                          016900
        WHERE EMPLOYEE.NUMBER = DISPANEL.NUMBER               017000
        MOVE DISPANEL TO EMPLOYEE BY NAME                     017100
        WHEN NONE                                             017200
        SET DISPANEL.MSG = 'EMPLOYEE DELETED'                 017300
        SET ATTR HIGHLIGHT TEMP ON DISPANEL.NUMBER            017400
        ENDFOR                                                017500
```

Line 16900 is now at the top of the region with subsequent lines filling the region.

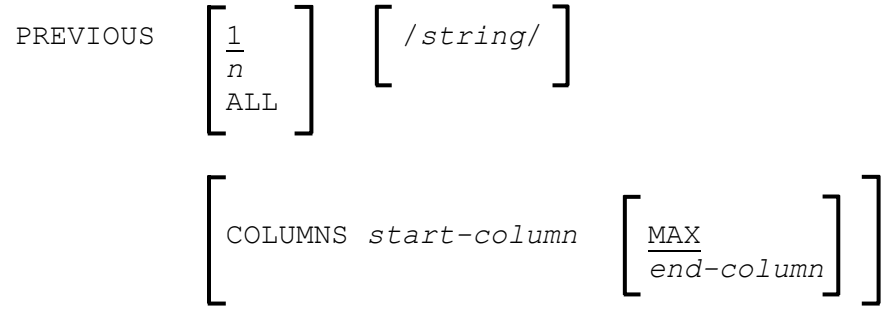
```
=>
....+....2....+....3....+....4....+....5....+....6....+....7....       COMMAND
        FOR EMPLOYEE                                          016900
        WHERE EMPLOYEE.NUMBER = DISPANEL.NUMBER               017000
        MOVE DISPANEL TO EMPLOYEE BY NAME                     017100
        WHEN NONE                                             017200
        SET DISPANEL.MSG = 'EMPLOYEE DELETED'                 017300
        SET ATTR HIGHLIGHT TEMP ON DISPANEL.NUMBER            017400
        ENDFOR                                                017500
```

PREVIOUS Command

The PREVIOUS command locates the previous occurrence of a string of characters. The search begins with the line prior to the current top line of the display region and processes backward.

Use the COLUMNS operand or the SET EDIT BOUNDS command to restrict the search to an inclusive range of columns. You can also locate a specific number of occurrences.

Format:



1
n
ALL

The maximum number of occurrences of the string to be found. 1 is the default. Specifying 1 causes the previous occurrence of the string to be positioned at the top of the region with subsequent lines filling the region. Specify ALL to locate all occurrences of the string. If you specify ALL or any value greater than 1, only the lines containing the string will be displayed. Use scrolling commands or PF keys to view all occurrences if the display exceeds the size of the region.

/

The character used to delimit the search string. The same character must be used consistently within a command. Any special character can be used except: the currently defined command delimiter (installed default is a semicolon [;]) or comment character (installed default is a colon [:]), an asterisk (*), an at sign (@), a pound sign (#), or a dollar sign ($).

string

A character or string of characters that the command searches for. If you do not specify a string, the string specified on the prior FIND, INCLUDE, EXCLUDE, FIRST, NEXT, PREVIOUS, or LAST command is used as a default value.

The string may not include the string delimiter character or the current command delimiter.

start-column
The column at which the search for the string begins.

end-column
The column at a which the search for the string ends.

MAX
Default. The farthest right-hand column.

Note: If you use the n operand and specify ALL or any value greater than 1, only the lines which contain the string are displayed. You can use the following commands without causing this display to be changed:

FIND, FIRST, NEXT, PREVIOUS, or LAST (where n > 1)
IGNORE
SCROLL

The use of any command other than those listed above will cause this display to be replaced by a display showing the lines of data in sequence.

RENUMBER Command

The RENUMBER command is used to renumber the sequence numbers of the current member and to set the increment used by the sequence numbers.

Format:

```
RENUMBER [ [BY] n ]
```

[BY] *n*

Specifies the increment to be used in the sequence numbers. n can be any integer (that allows the entity to be completely numbered). If no increment is specified, a default of 100 is used.

The following example illustrates the RENUMBER command. First, a move is specified.

```
=>
=>
=>
....+....2....+....3....+....4....+....5....+....6....+....7... COMMAND
<<PROCESS-PANEL>> PROC                                          016500
    TRANSMIT DISPANEL                                           016600
    SELECT                                                      016700
    WHEN $ENTER-KEY                                             MM6800
        FOR EMPLOYEE                                            016900
        WHERE EMPLOYEE.NUMBER = DISPANEL.NUMBER                 017000
        MOVE DISPANEL TO EMPLOYEE BY NAME                       017100
        WHEN NONE                                               017200
        SET DISPANEL.MSG = 'EMPLOYEE DELETED'                   017300
        SET ATTR HIGHLIGHT TEMP ON DISPANEL.NUMBER              017400
        ENDFOR                                                  017500
        SET NEXT-PANEL = 'DISPANEL'                             MM7600
    WHEN $PF11                                                  017700
        SET NEXT-PANEL = 'EMPMENU'                              017800
    WHEN $PF12                                                  017900
        DO TERMINATE-SESSION                                    018000
    WHEN OTHER                                                  B18100
    ...
```

After the move, new sequence numbers are assigned to the new locations. The RENUMBER command is entered.

```
=> RENUMBER BY 100
=>
=>
....+....2....+....3....+....4....+....5....+....6....+....7... COMMAND
<<PROCESS-PANEL>> PROC                                          016500
    TRANSMIT DISPANEL                                           016600
    SELECT                                                      016700
    WHEN $PF11                                                  017700
        SET NEXT-PANEL = 'EMPMENU'                              017800
    WHEN $PF12                                                  017900
        DO TERMINATE-SESSION                                    018000
    WHEN $ENTER-KEY                                             018001
        FOR EMPLOYEE                                            018002
        WHERE EMPLOYEE.NUMBER = DISPANEL.NUMBER                 018003
        MOVE DISPANEL TO EMPLOYEE BY NAME                       018004
        WHEN NONE                                               018005
        SET DISPANEL.MSG = 'EMPLOYEE DELETED'                   018006
        SET ATTR HIGHLIGHT TEMP ON DISPANEL.NUMBER              018007
        ENDFOR                                                  018008
        SET NEXT-PANEL = 'DISPANEL'                             018009
    WHEN OTHER                                                  018100
    ...
```

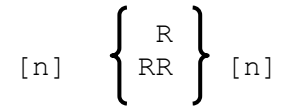
The following example illustrates the renumbered sequence numbers.

```
=>
=>
=>
....+....2....+....3....+....4....+....5....+....6....+....7... COMMAND
<<PROCESS-PANEL>> PROC                                          016500
    TRANSMIT DISPANEL                                           016600
    SELECT                                                      016700
    WHEN $PF11                                                  016800
        SET NEXT-PANEL = 'EMPMENU'                              016900
    WHEN $PF12                                                  017000
        DO TERMINATE-SESSION                                    017100
    WHEN $ENTER-KEY                                             017200
        FOR EMPLOYEE                                            017300
        WHERE EMPLOYEE.NUMBER = DISPANEL.NUMBER                 017400
        MOVE DISPANEL TO EMPLOYEE BY NAME                       017500
        WHEN NONE                                               017600
        SET DISPANEL.MSG = 'EMPLOYEE DELETED'                   017700
        SET ATTR HIGHLIGHT TEMP ON DISPANEL.NUMBER              017800
        ENDFOR                                                  017900
        SET NEXT-PANEL = 'DISPANEL'                             018000
    WHEN OTHER                                                  018100
    ...
```

Repeat Line Command

The repeat line command duplicates a line or range of lines. Lines duplicated with the repeat line commands are inserted immediately after the line containing the repeat line command (R) or the ending repeat range line command (RR).

Format:



R

Specified on the line to be repeated.

RR

Used in pairs to indicate the start and end of a range of lines to be repeated. The second occurrence of this command on the same or subsequent screen is interpreted as the end of the range.

After specifying the start of a range with the RR line command, the region can be scrolled forward or backward and other line commands can be performed. Until the end of the range started by an RR line command is found, the "PENDING" message appears in the message line.

n

Optionally specifies the number of times (between 1 and 100) the line or range of lines is to be repeated. If n is not specified, the line is repeated once. The SET EDIT MULTIPLIER command determines whether n is placed on the left or right. For example, if you specify,

```
SET EDIT MULTIPLIER RIGHT
```

then you specify,

```
R7
```

If you specify,

```
SET EDIT MULTIPLIER LEFT
```

then you specify,

```
7R
```

When interpreting line commands that mark ranges of lines to be repeated, the commands are matched into pairs from the top down, regardless of the order the line commands were entered.

The inserted lines are assigned sequence numbers in increments of one starting from the previous existing line. When the number of inserted lines exceeds the number of integers between the sequence numbers of the existing lines, the sequence numbers on the lines following the insertion are changed. (See the following example.) To renumber the sequence numbers, refer to the RENUMBER primary editing command.

The following example shows the use of the repeat line command. Line 017600 contains a line command that causes a line to repeat twice.

```
....+....2....+....3....+....4....+....5....+....6....+....7... COMMAND
<<PROCESS-PANEL>> PROC                                        016500
    TRANSMIT DISPANEL                                         016600
    SELECT                                                    016700
    WHEN $ENTER-KEY                                           016800
        FOR EMPLOYEE                                          016900
        WHERE EMPLOYEE.NUMBER = DISPANEL.NUMBER               017000
        MOVE DISPANEL TO EMPLOYEE BY NAME                     017100
        WHEN NONE                                             017200
        SET DISPANEL.MSG = 'EMPLOYEE DELETED'                 017300
        SET ATTR HIGHLIGHT TEMP ON DISPANEL.NUMBER            017400
        ENDFOR                                                017500
        SET NEXT-PANEL = 'DISPANEL'                           2R7600
    WHEN $PF11                                                017700
        SET NEXT-PANEL = 'EMPMENU'                            017800
    WHEN $PF12                                                017900
        DO TERMINATE-SESSION                                  018000
    WHEN OTHER                                                018100
        SET DISPANEL.MSG = 'UNABLE TO INTERPRET KEY ENTERED'   018200
        SET NEXT-PANEL = 'DISPANEL'                           018300
    ENDSEL                                                    018400
```

The following example shows the repeated lines with sequence numbers assigned (017601 and 017602).

```
....+....2....+....3....+....4....+....5....+....6....+....7... COMMAND
<<PROCESS-PANEL>> PROC                                        016500
    TRANSMIT DISPANEL                                         016600
    SELECT                                                    016700
    WHEN $ENTER-KEY                                           016800
        FOR EMPLOYEE                                          016900
        WHERE EMPLOYEE.NUMBER = DISPANEL.NUMBER               017000
        MOVE DISPANEL TO EMPLOYEE BY NAME                     017100
        WHEN NONE                                             017200
        SET DISPANEL.MSG = 'EMPLOYEE DELETED'                 017300
        SET ATTR HIGHLIGHT TEMP ON DISPANEL.NUMBER            017400
        ENDFOR                                                017500
        SET NEXT-PANEL = 'DISPANEL'                           017600
        SET NEXT-PANEL = 'DISPANEL'                           017601
     SET NEXT-PANEL = 'DISPANEL'                              017602
  WHEN $PF11                                                  017700
        SET NEXT-PANEL = 'EMPMENU'                            017800
    WHEN $PF12                                                017900
        DO TERMINATE-SESSION                                  018000
    WHEN OTHER                                                018100
        SET DISPANEL.MSG = 'UNABLE TO INTERPRET KEY ENTERED'   018200
        SET NEXT-PANEL = 'DISPANEL'                           018300
    ENDSEL                                                    018400
```

In the following example, lines 017700 and 018000 contain line commands that cause a range of lines to repeat.

```
....+....2....+....3....+....4....+....5....+....6....+....7... COMMAND
<<PROCESS-PANEL>> PROC                                      016500
    TRANSMIT DISPANEL                                       016600
    SELECT                                                  016700
    WHEN $ENTER-KEY                                         016800
        FOR EMPLOYEE                                        016900
        WHERE EMPLOYEE.NUMBER = DISPANEL.NUMBER             017000
        MOVE DISPANEL TO EMPLOYEE BY NAME                   017100
        WHEN NONE                                           017200
        SET DISPANEL.MSG = 'EMPLOYEE DELETED'               017300
        SET ATTR HIGHLIGHT TEMP ON DISPANEL.NUMBER          017400
        ENDFOR                                              017500
        SET NEXT-PANEL = 'DISPANEL'                         017600
    WHEN $PF11                                              RR7700
        SET NEXT-PANEL = 'EMPMENU'                          017800
    WHEN $PF12                                              017900
        DO TERMINATE-SESSION                                RR8000
    WHEN OTHER                                              018100
        SET DISPANEL.MSG = 'UNABLE TO INTERPRET KEY ENTERED'    018200
        SET NEXT-PANEL = 'DISPANEL'                         018300
    ENDSEL                                                  018400
```

The following example shows the repeated lines inserted after line 018000 with sequence numbers assigned (018001 through 018004).

```
....+....2....+....3....+....4....+....5....+....6....+....7... COMMAND
<<PROCESS-PANEL>> PROC                                      016500
    TRANSMIT DISPANEL                                       016600
    SELECT                                                  016700
    WHEN $ENTER-KEY                                         016800
        FOR EMPLOYEE                                        016900
        WHERE EMPLOYEE.NUMBER = DISPANEL.NUMBER             017000
        MOVE DISPANEL TO EMPLOYEE BY NAME                   017100
        WHEN NONE                                           017200
        SET DISPANEL.MSG = 'EMPLOYEE DELETED'               017300
        SET ATTR HIGHLIGHT TEMP ON DISPANEL.NUMBER          017400
        ENDFOR                                              017500
        SET NEXT-PANEL = 'DISPANEL'                         017600
    WHEN $PF11                                              017700
        SET NEXT-PANEL = 'EMPMENU'                          017800
    WHEN $PF12                                              017900
        DO TERMINATE-SESSION                                018000
    WHEN $PF11                                              018001
      SET NEXT-PANEL = 'EMPMENU'                            018002
    WHEN $PF12                                              018003
      DO TERMINATE-SESSION                                  018004
  WHEN OTHER                                                018100
        SET DISPANEL.MSG = 'UNABLE TO INTERPRET KEY ENTERED'    018200
        SET NEXT-PANEL = 'DISPANEL'                         018300
    ENDSEL                                                  018400
```

RESET Command

The RESET command resets any pending M, MM, C, CC, DD, RR, A, or B line commands, panel LAYOUT move and copy commands, and errors. This primary command, issued in the command area, performs the same function as the Reset line command.

Format:

```
RESET
```

RESET Line Command

The RESET line command resets any pending M, MM, DD, <<,)), >>, ((, C, CC, RR, A, or B line commands and errors.

Format:

```
RESET
```

The reset line command can be entered on any line and applies to all pending line commands.

The following example illustrates how the RESET line command is entered in the sequence number and command field. Note that a move is pending on line 1500.

```
PENDING: MM(1500)
....+....2....+....3....+....4....+....5....+....6....+....7... COMMAND
<<PROCESS-PANEL>> PROC                                      RESET0
    TRANSMIT DISPANEL                                       016600
    SELECT                                                  016700
    WHEN $PF11                                              016800
        SET NEXT-PANEL = 'EMPMENU'                          016900
    WHEN $PF12                                              017000
        DO TERMINATE-SESSION                                017100
    WHEN OTHER                                              017200
        SET DISPANEL.MSG = 'UNABLE TO INTERPRET KEY ENTERED'    017300
        SET NEXT-PANEL = 'DISPANEL'                         017400
    ENDSEL                                                  017500
```

After the data is applied, the pending move command is deleted and the PENDING message ends.

```
....+....2....+....3....+....4....+....5....+....6....+....7... COMMAND
<<PROCESS-PANEL>> PROC                                      016500
    TRANSMIT DISPANEL                                       016600
    SELECT                                                  016700
    WHEN $PF11                                              016800
        SET NEXT-PANEL = 'EMPMENU'                          016900
    WHEN $PF12                                              017000
        DO TERMINATE-SESSION                                017100
    WHEN OTHER                                              017200
        SET DISPANEL.MSG = 'UNABLE TO INTERPRET KEY ENTERED'    017300
        SET NEXT-PANEL = 'DISPANEL'                         017400
    ENDSEL                                                  017500
```

RESHOW Line Command

The RESHOW line command is used to refresh the region (e.g. <<PROCESS-PANEL>> PROC) with the data as it originally appeared before data was entered for the current transaction. Changes made to the data on the screen during the current transaction are not applied. The RESHOW line command can be specified on any line in the region and applies to the entire region.

Format:

```
RESHOW
```

The following example illustrates changes made to data in the region, and the restoration of the data in the region without the changes being applied. The current transaction begins with the following data in the region.

```
....+....2....+....3....+....4....+....5....+....6....+....7... COMMAND
<<PROCESS-PANEL>> PROC                                    017400
    TRANSMIT DISPANEL                                     017500
    SELECT                                                017600
    WHEN $PF11                                            017700
        SET NEXT-PANEL = 'EMPMENU'                        017800
    WHEN $PF12                                            017900
        DO TERMINATE-SESSION                              018000
    WHEN OTHER                                            018100
        SET DISPANEL.MSG = 'UNABLE TO INTERPRET KEY ENTERED'   018200
        SET NEXT-PANEL = 'DISPANEL'                       018300
    ENDSEL                                                018400
```

The data is then changed on the screen (018000), and delete line commands are specified. The RESHOW line command is then entered on line 017400 (although the RESHOW command can be specified anywhere in the region).

```
....+....2....+....3....+....4....+....5....+....6....+....7....       COMMAND
<<PROCESS-PANEL>> PROC                                    RESHOW
    TRANSMIT DISPANEL                                     017500
    SELECT                                                017600
    WHEN $PF11                                            D17700
        SET NEXT-PANEL = 'EMPMENU'                        D17800
    WHEN $PF12                                            017900
        DO CANCEL-SESSION                                 018000
    WHEN OTHER                                            018100
        SET DISPANEL.MSG = 'UNABLE TO INTERPRET KEY ENTERED'   018200
        SET NEXT-PANEL = 'DISPANEL'                       018300
    ENDSEL                                                018400
```

The result, as shown below, is the data as it appeared at the start of the current transaction with no changes applied to the data and no lines deleted.

```
....+....2....+....3....+....4....+....5....+....6....+....7... COMMAND
<<PROCESS-PANEL>> PROC                                      017400
    TRANSMIT DISPANEL                                       017500
    SELECT                                                  017600
    WHEN $PF11                                              017700
        SET NEXT-PANEL = 'EMPMENU'                          017800
    WHEN $PF12                                              017900
        DO TERMINATE-SESSION                                018000
    WHEN OTHER                                              018100
        SET DISPANEL.MSG = 'UNABLE TO INTERPRET KEY ENTERED'   018200
        SET NEXT-PANEL = 'DISPANEL'                         018300
    ENDSEL                                                  018400
```

ROLLBACK Command

The ROLLBACK command restores the contents of the component as of the last CHECKPOINT or as of the beginning of the edit session (the automatic checkpoint).

Format:

```
ROLLBACK
```

SCROLL Command

The SCROLL command provides the ability to view and edit data outside the screen presentation area. The command has three basic formats; the first two are for scrolling vertically, and the third is for horizontal scrolling.

Vertical Scrolling

The following form of the SCROLL command moves the window overlooking data forward or backward, according to the cursor or frame position, or by number of pages or lines.

Format:

```
           { FORWARD  }  [   { CURSOR }  ]
           { BACKWARD }  [   { FRAME  }  ]
[SCROLL]   { +        }  [   { PAGES  }  ]
           { -        }  [ n { LINES  }  ]
```

The keyword SCROLL is optional.

FORWARD
: Scrolls forward either one frame (when used with the FRAME option) or until the line containing the cursor is positioned at the top of the region (when used with the CURSOR option).

BACKWARD
Scrolls backward either one frame (when used with the FRAME option) or until the line containing the cursor is positioned at the top of the region (when used with the CURSOR option).

+
+ is equivalent to FORWARD.

-
- is equivalent to BACKWARD.

CURSOR
Scrolls forward or backward until the line containing the cursor is at the bottom or top of the region, respectively. If SET SCROLL CURSOR is in effect, CURSOR is the default.

FRAME
Scrolls forward or backward by an entire frame but repeats one context line at the top of the next frame from the current frame. If SET SCROLL FRAME is in effect, FRAME is the default.

***n* PAGES**
(for outputs in the output library only) Scrolls forward or backward according to the number of specified report pages.

***n* LINES**
Scrolls forward or backward according to the number of specified lines.

Note: To use a + as SCROLL FORWARD or - as SCROLL BACKWARD you must define other characters as the current defaults for the REPEAT and RESHOW commands.

Format:

```
          { TOP    }
[SCROLL]  { BOTTOM }
```

This form of the SCROLL command moves the window to the top or bottom of the data.

The keyword SCROLL is optional.

TOP
Positions the window so that the first line of data is at the top of the region.

BOTTOM
Positions the window so that the last line of the data is at the bottom of the region.

Examples:

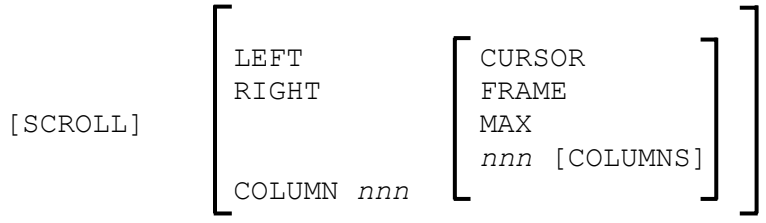
```
SCROLL +
```

```
+ 5
SCROLL -
- 10
SCROLL FOR CUR
SCR FOR 12 LINES
SCR BAC 7 PAGES
SCROLL TOP
TOP
SCROLL BOTTOM
BOTTOM
BOT
```

Horizontal Scrolling

The following SCROLL command scrolls the window overlooking an output in the output library or a panel layout in a horizontal direction. Options with column numbers cannot be used with output members.

Format:



The keyword SCROLL is optional if at least one other argument is specified.

LEFT
(default) Scrolls the window to the left (by cursor, frame, or column, or until the left margin).

RIGHT
Scrolls the window to the right (by cursor, frame, or column, or until the right margin).

CURSOR
Scrolls left or right until the column containing the cursor is at the left or right of the region, respectively. If SET SCROLL CURSOR is in effect, CURSOR is the default.

FRAME
Scrolls left or right by an entire frame but repeats one context column at the left or right of the next frame. If SET SCROLL FRAME is in effect, FRAME is the default.

MAX
Specifies the left or right scroll to be to the left or right edge of the defined panel.

nnn **COLUMNS**
Specifies the number of columns to be scrolled to the left or right from the current position.

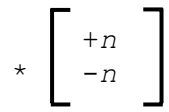
COLUMN *nnn*
Specifies the absolute column nnn as the left margin. LEFT and RIGHT are optional with SCROLL COLUMN.

Scroll Line Command

The scroll line command is used to scroll a specific line to the top of the region with subsequent lines filling the region.

The format of the scroll line command is as follows.

Format:



*****

Position line to the top of the region.

+n

The line 'n' number of lines after the line containing the command.

-n

The line 'n' number of lines before the line containing the command. n can be any number from 1 to 100.

The following example shows the use of the scroll line command, an asterisk (*) in line 017700, to position line 017700 to the top of the region.

```
....+....2....+....3....+....4....+....5....+....6....+....7..  COMMAND
<<PROCESS-PANEL>> PROC                                        016500
    TRANSMIT DISPANEL                                         016600
    SELECT                                                    016700
    WHEN $ENTER-KEY                                           016800
        FOR EMPLOYEE                                          016900
        WHERE EMPLOYEE.NUMBER = DISPANEL.NUMBER               017000
        MOVE DISPANEL TO EMPLOYEE BY NAME                     017100
        WHEN NONE                                             017200
        SET DISPANEL.MSG = 'EMPLOYEE DELETED'                 017300
        SET ATTR HIGHLIGHT TEMP ON DISPANEL.NUMBER            017400
        ENDFOR                                                017500
        SET NEXT-PANEL = 'DISPANEL'                           017600
    WHEN $PF11                                                *17700
        SET NEXT-PANEL = 'EMPMENU'                            017800
    WHEN $PF12                                                017900
        DO TERMINATE-SESSION                                  018000
    WHEN OTHER                                                018100
        SET DISPANEL.MSG = 'UNABLE TO INTERPRET KEY ENTERED'   018200
        SET NEXT-PANEL = 'DISPANEL'                           018300
    ENDSEL                                                    018400
```

Line 17700 is now at the top of the region with subsequent lines filling the region. To position line 016800 to the top of the region, the command *-9 can be used.

```
....+....2....+....3....+....4....+....5....+....6....+....7... COMMAND
    WHEN $PF11                                          *-9700
        SET NEXT-PANEL = 'EMPMENU'                      017800
    WHEN $PF12                                          017900
        DO TERMINATE-SESSION                            018000
    WHEN OTHER                                          018100
        SET DISPANEL.MSG = 'UNABLE TO INTERPRET KEY ENTERED'   018200
        SET NEXT-PANEL = 'DISPANEL'                     018300
    ENDSEL                                              018400
         <<DELETE-DESIRED-EMPLOYEE>> PROC               018500
    FOR EMPLOYEE                                        018600
        WHERE EMPLOYEE.NUMBER = EMPMENU.NUMBER          018700
        DELETE EMPLOYEE                                 018800
    WHEN NONE                                           018900
```

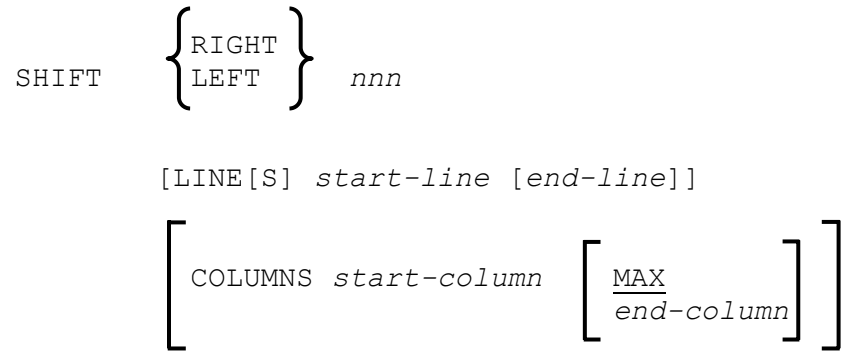
Line 016800 is now at the top of the region with subsequent lines filling the region.

```
....+....2....+....3....+....4....+....5....+....6....+....7... COMMAND
    WHEN $ENTER-KEY                                     016800
        FOR EMPLOYEE                                    016900
        WHERE EMPLOYEE.NUMBER = DISPANEL.NUMBER         017000
        MOVE DISPANEL TO EMPLOYEE BY NAME               017100
        WHEN NONE                                       017200
        SET DISPANEL.MSG = 'EMPLOYEE DELETED'           017300
        SET ATTR HIGHLIGHT TEMP ON DISPANEL.NUMBER      017400
        ENDFOR                                          017500
        SET NEXT-PANEL = 'DISPANEL'                     017600
    WHEN $PF11                                          017700
        SET NEXT-PANEL = 'EMPMENU'                      017800
    WHEN $PF12                                          017900
        DO TERMINATE-SESSION                            018000
    WHEN OTHER                                          018100
        SET DISPANEL.MSG = 'UNABLE TO INTERPRET KEY ENTERED'   018200
        SET NEXT-PANEL = 'DISPANEL'                     018300
    ENDSEL                                              018400
```

SHIFT Command

The SHIFT command is used to shift a line or range of lines by one or more columns to the right or to the left. SHIFT can be restricted to an inclusive range of lines and/or columns using the format below, or using the SET EDIT BOUNDS command.

Format:



Line and column ranges specified in a SHIFT command only apply to the current command. Specifying lines or columns is optional.

nnn
The number of columns to shift the data.

start-line
The line at which the shift begins. It can be specified as:

num	The sequence number of the start line.
TOP	Indicates the first line of the member. If a start line is not specified, TOP is the default.
CURSOR	Indicates the position at which the cursor is located.
offset	A position relative to the top line displayed:
	***** The top line displayed.
	*** +n** The line 'n' number of lines below the top line displayed.
	*** -n** The line 'n' number of lines before the top line displayed.

Note: The keyword LINE can be omitted when you specify TOP, BOTTOM, CURSOR, or an offset.

end-line

The line at which the shift ends. It can be specified as:

BOTTOM	Default. The last line of the member.
num	A sequence number.
CURSOR	Indicates the position at which the cursor is located.
offset	A position relative to the top line displayed:
	***** The top line displayed.
	*** +n** The line 'n' number of lines below the top line displayed.
	*** -n** The line 'n' number of lines before the top line displayed.

start-column

The column at which the shift begins.

end-column

The column at which the shift ends.

MAX

Default. The farthest right-hand column.

Note: Using SET EDIT TRUNCATION Y, data may be shifted past the end of the column range (with truncation). Using SET EDIT TRUNCATION N, data which would be truncated is instead positioned at the margin, and the SHIFT proceeds to other lines in the range. This allows data to be left or right-aligned. Note that this differs from the CHANGE command.

Shift Line Command

The shift line command is used to shift a line or range of lines by one or more columns to the right or to the left. Shift can be restricted to an inclusive range of lines using the formats below, or using the SET EDIT BOUNDS command). This command has two formats:

```
(   )    ((   ))
<   >    <<   >>
```

The first format is used to unconditionally shift data within the column range of 1 to 72. If the shift causes the data to move out of the edit window, the data will be truncated. The format is as follows.

)
Single line to be shifted to the right.

))
Block of lines to be shifted to the right.)) must be entered on the first and last lines of the block to be shifted.

)B
Lines to be shifted right include from this line through the last line.

)T
Lines to be shifted right include from the first line through this line.

(
Single line to be shifted to the left.

((
Block of lines to be shifted left. ((must be entered on the first and last lines of the block to be shifted.

(B
Lines to be shifted left include from this line through the last line.

(T
Lines to be shifted left include from the first line through this line

The second format of the shift line command only affects leading or trailing blanks within the column range of 1 to 72. The format is as follows:

>
Single line to be shifted to the right.

>>

Block of lines to be shifted right. >> must be entered on the first and last lines of the block to be shifted.

>B

Lines to be shifted right include from this line through the last line.

>T

Lines to be shifted right include from the first line through this line.

<

Single line to be shifted left.

<<

Block of lines to be shifted left. << must be entered on the first and last lines of the block to be shifted.

<B

Lines to be shifted left include from this line through the last line.

<T

Lines to be shifted left include from the first line through this line

Notes: You can limit the shift to part of a line or range of lines by placing the cursor on the line being shifted or within the range of lines at the first column you want shifted. Columns to the left of the cursor are not affected.

You can also specify how many columns to shift the data by including a multiplier before or after the command (depending on whether the SET EDIT MULTIPLIER is left or right). With a block shift -- ((,)), <<, or >> -- the multiplier is specified at the start of the block.

The range of lines to be shifted does not need to be on the display screen.

Scrolling and SET EDIT TRUNCATION do not affect the margin command.

Index

A

B

C

D

E

F

G

H

I

K

L

M

N

O

P

R

S

T

U