

CA 2E

Toolkit Reference Guide

r8.5



This documentation and any related computer software help programs (hereinafter referred to as the "Documentation") is for the end user's informational purposes only and is subject to change or withdrawal by CA at any time.

This Documentation may not be copied, transferred, reproduced, disclosed, modified or duplicated, in whole or in part, without the prior written consent of CA. This Documentation is confidential and proprietary information of CA and protected by the copyright laws of the United States and international treaties.

Notwithstanding the foregoing, licensed users may print a reasonable number of copies of the documentation for their own internal use, and may make one copy of the related software as reasonably required for back-up and disaster recovery purposes, provided that all CA copyright notices and legends are affixed to each reproduced copy. Only authorized employees, consultants, or agents of the user who are bound by the provisions of the license for the product are permitted to have access to such copies.

The right to print copies of the documentation and to make a copy of the related software is limited to the period during which the applicable license for the Product remains in full force and effect. Should the license terminate for any reason, it shall be the user's responsibility to certify in writing to CA that all copies and partial copies of the Documentation have been returned to CA or destroyed.

EXCEPT AS OTHERWISE STATED IN THE APPLICABLE LICENSE AGREEMENT, TO THE EXTENT PERMITTED BY APPLICABLE LAW, CA PROVIDES THIS DOCUMENTATION "AS IS" WITHOUT WARRANTY OF ANY KIND, INCLUDING WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NONINFRINGEMENT. IN NO EVENT WILL CA BE LIABLE TO THE END USER OR ANY THIRD PARTY FOR ANY LOSS OR DAMAGE, DIRECT OR INDIRECT, FROM THE USE OF THIS DOCUMENTATION, INCLUDING WITHOUT LIMITATION, LOST PROFITS, BUSINESS INTERRUPTION, GOODWILL, OR LOST DATA, EVEN IF CA IS EXPRESSLY ADVISED OF SUCH LOSS OR DAMAGE.

The use of any product referenced in the Documentation is governed by the end user's applicable license agreement.

The manufacturer of this Documentation is CA.

Provided with "Restricted Rights." Use, duplication or disclosure by the United States Government is subject to the restrictions set forth in FAR Sections 12.212, 52.227-14, and 52.227-19(c)(1) - (2) and DFARS Section 252.227-7014(b)(3), as applicable, or their successors.

All trademarks, trade names, service marks, and logos referenced herein belong to their respective companies.

Copyright © 2009 CA. All rights reserved.

Contact CA

Contact Technical Support

For your convenience, CA provides one site where you can access the information you need for your Home Office, Small Business, and Enterprise CA products. At <http://ca.com/support>, you can access the following:

- Online and telephone contact information for technical assistance and customer services
- Information about user communities and forums
- Product and documentation downloads
- CA Support policies and guidelines
- Other helpful resources appropriate for your product

Provide Feedback

If you have comments or questions about CA product documentation, you can send a message to techpubs@ca.com.

If you would like to provide feedback about CA product documentation, complete our short [customer survey](#), which is also available on the CA support website, found at <http://ca.com/support>.

Contents

Chapter 1: Introduction

Documentation	1-1
Arrangement of this Manual	1-1
Related Publications	1-1
Documentation Presentation Standards	1-2
Calling CA 2E Commands	1-2
Data Object Commands	1-3
CA 2E Modules	1-3
CA 2E Object Names	1-3
Abbreviated Command Names	1-4

Chapter 2: 2E Commands, YA to YDOCMSGREF

YADDDSNFM (Add Design File Member)	2-1
Syntax Diagram	2-1
Parameters	2-2
YADDHLPTBL (Add Help Vector Table)	2-2
Syntax Diagram	2-3
Parameters	2-3
YADDLLE (Add Library List Entry)	2-6
Syntax Diagram	2-7
Parameters	2-8
YADDMLE (Add Member List)	2-9
Syntax Diagram	2-10
Parameters	2-10
YADDOLE (Add Object List Entry)	2-12
Syntax Diagram	2-12
Parameters	2-13
YADDSRCM (Add Source Member)	2-14
Syntax Diagram	2-14
Parameters	2-14
YAPYY1DCHG (Apply Data Changes)	2-15
Syntax Diagram	2-15
Parameters	2-16
YBLDDBFLST (Build Database File List)	2-17
Syntax Diagram	2-17

Parameters	2-18
YBLDDOC (Build Master Document)	2-19
Syntax Diagram	2-19
Parameters	2-20
YBLDFMTLST (Build Format List)	2-21
Syntax Diagram	2-22
Parameters	2-22
YBLDLIBLST (Build Library List)	2-23
Syntax Diagram	2-24
Parameters	2-25
YBLDMBRLST (Build Member List)	2-27
Syntax Diagram	2-28
Parameters	2-28
YBLDOBJLST (Build Object List)	2-29
Syntax Diagram	2-30
Parameters	2-30
YCHGCMD (Change Command)	2-31
Syntax Diagram	2-32
Parameters	2-32
YCHGJOB DLL (Change Job Description Library List)	2-37
Syntax Diagram	2-37
Parameters	2-38
YCHGLIBL (Change Library List)	2-38
Syntax Diagram	2-39
Parameters	2-39
YCHGLIBLST (Change Library List Description)	2-40
Syntax Diagram	2-41
Parameters	2-41
YCHGLST (Change List)	2-42
Syntax Diagram	2-43
Parameters	2-43
YCHGPWD (Change Password)	2-46
Syntax Diagram	2-46
Parameters	2-46
YCHGOBJOWN (Change Object Ownership)	2-46
Syntax Diagram	2-47
Parameters	2-47
YCHGUSRPRF (Change User Profile)	2-49
Syntax Diagram	2-49
Parameters	2-53
YCHKIFS OBJ (Check IFS Object)	2-63
Parameters	2-63
YCHKLIBLST (Check Library List)	2-66

Syntax Diagram	2-66
Parameters	2-66
YCHKLSTE (Check List Entries)	2-68
Syntax Diagram	2-69
Parameters	2-70
YCHKMNU (Check Menu)	2-75
Syntax Diagram	2-76
Parameters	2-76
YCHKPNL (Check Panel Design)	2-78
Syntax Diagram	2-78
Parameters	2-78
YCHKPWDVAL (Check New Password)	2-80
Syntax Diagram	2-80
Parameters	2-80
YCHKRPT (Check Report Design)	2-81
Syntax Diagram	2-81
Parameters	2-81
YCHKVN (Check System Name Is Valid)	2-83
Syntax Diagram	2-83
Parameters	2-83
YCMPSRC (Compare Source)	2-84
Syntax Diagram	2-85
Parameters	2-86
YCPYF (Copy Files)	2-89
Syntax Diagram	2-90
Parameters	2-90
YCPYLIBLST (Copy Library List)	2-93
Syntax Diagram	2-94
Parameters	2-94
YCPYLST (Copy List)	2-95
Syntax Diagram	2-95
Parameters	2-95
YCPYMNU (Copy Menu)	2-96
Syntax Diagram	2-97
Parameters	2-97
YCPYMSGD (Copy Message Description)	2-98
Syntax Diagram	2-99
Parameters	2-99
YCPYPNL (Copy Panel Design)	2-100
Syntax Diagram	2-101
Parameters	2-101
YCPYRPT (Copy Report Design)	2-103
Syntax Diagram	2-103

Parameters	2-104
YCPYUSRPRF (Copy User Profile)	2-105
Syntax Diagram	2-106
Parameters	2-106
YCRTDSNF (Create Design File)	2-107
Syntax Diagram	2-108
Parameters	2-108
YCRTDUPOBJ (Create Duplicate Objects)	2-110
Syntax Diagram	2-110
Parameters	2-111
YCRTOBJ (Create Objects)	2-113
Syntax Diagram	2-114
Parameters	2-116
YC RTPNLDDS (Create DDS From Panel Design)	2-122
Syntax Diagram	2-123
Parameters	2-123
YC RTRPTDDS Create DDS From Report Design)	2-125
Syntax Diagram	2-126
Parameters	2-126
YC RTSRCPF (Create Source Files)	2-128
Syntax Diagram	2-128
Parameters	2-129
YC RTUSRPRF (Create User Profile)	2-131
Syntax Diagram	2-132
Parameters	2-135
YCVTAUTL (Convert Authorization List)	2-142
Syntax Diagram	2-143
Parameters	2-143
YCVTBIN (Convert Binary To Decimal)	2-144
Syntax Diagram	2-145
Parameters	2-145
YCVTDBFLST (Convert DBF List To Member List)	2-146
Syntax Diagram	2-146
Parameters	2-146
YCVTDDBR (Convert Database Relations)	2-148
Syntax Diagram	2-148
Parameters	2-149
YCVTDDSCKY (Convert DDS Command Keys)	2-150
Syntax Diagram	2-151
Parameters	2-151
YCVTDDSIGC (Convert DDS To IGC)	2-155
Syntax Diagram	2-155
Parameters	2-155

YCVTDEC (Convert Decimal To Binary)	2-161
Syntax Diagram	2-162
Parameters	2-162
YCVTOBJLST (Convert Object List to Member List)	2-163
Syntax Diagram	2-163
Parameters	2-163
YCVTPGMREF (Convert Program References)	2-165
Syntax Diagram	2-166
Parameters	2-166
YCVTPRT (Convert Print Output To Source Member)	2-168
Syntax Diagram	2-168
Parameters	2-168
YCVTSPLF (Convert Spooled File)	2-170
Parameters	2-171
Notes	2-180
Examples	2-180
YCVTUSRPRF (Convert User Profile)	2-181
Syntax Diagram	2-181
Parameters	2-182
YDFNPNLDSN (Define Panel Design)	2-183
Syntax Diagram	2-183
Parameters	2-183
YDLTLIBLST (Delete Library List)	2-185
Syntax Diagram	2-185
Parameters	2-185
YDLTLST (Delete Library List)	2-186
Syntax Diagram	2-186
Parameters	2-186
YDLTMNU (Delete Menu)	2-187
Syntax Diagram	2-187
Parameters	2-187
YDLTOBJ (Delete Object)	2-188
Syntax Diagram	2-188
Parameters	2-188
YDLTPNL (Delete Panel Design)	2-190
Syntax Diagram	2-190
Parameters	2-190
YDLTRPT (Delete Report Design)	2-191
Syntax Parameters	2-191
Parameters	2-191
YDLTUSRPRF (Delete User Profile)	2-192
Syntax Diagram	2-192
Parameters	2-193

YDOCAUT (Document Object Authorities)	2-194
Syntax Diagram	2-194
Parameters	2-194
YDOCDBFLST (Document DBF List)	2-196
Syntax Diagram	2-196
Parameters	2-197
YDOCEXCREF (Document Execution References)	2-197
Syntax Diagram	2-198
Parameters	2-199
YDOCF (Document File)	2-201
Syntax Diagram	2-202
Parameters	2-203
YDOCFLDREF (Document Field References)	2-206
Syntax Diagram	2-206
Parameters	2-207
YDOCFMTLST (Document Format List)	2-208
Syntax Diagram	2-208
Parameters	2-208
YDOCLIBLST (Document Library)	2-209
Syntax Diagram	2-209
Parameters	2-209
YDOCMBRLST (Document Member List)	2-210
Syntax Diagram	2-211
Parameters	2-211
YDOCMNU (Document Menu)	2-211
Syntax Diagram	2-212
Parameters	2-212
YDOCMNUREF (Document Menu References)	2-213
Syntax Diagram	2-214
Parameters	2-214
Syntax Diagram	2-215
Parameters	2-216

Chapter 3: CA 2E Commands, YD to YW

YDOCOBJLST (Document Object List)	3-1
Syntax Diagram	3-1
Parameters	3-2
YDOCPGM (Document Program command)	3-3
Syntax Diagram	3-3
Parameters	3-4
YDOCPGMREF (Document Program References)	3-6

Syntax Diagram	3-6
Parameters	3-7
YDOCPNL (Document Panel Design)	3-10
Syntax Diagram	3-10
Parameters	3-10
YDOCRPT (Document Report Design)	3-13
Syntax Diagram	3-13
Parameters	3-13
YDOCSRC (Document Source File Members)	3-16
Syntax Diagram	3-16
Parameters	3-16
YDOCUSRPRF (Document User Profile)	3-19
Syntax Diagram	3-19
Parameters	3-19
YDSPABR (Display Abbreviated)	3-20
Syntax Diagram	3-20
Parameters	3-21
YDSPEXPDAT (Display Expiry Date)	3-24
Syntax Diagram	3-24
Parameters	3-24
YDSPHLP (Display Help Text)	3-25
Syntax Diagram	3-26
Parameters	3-26
YDSPLIBLST (Display Library List)	3-34
Syntax Diagram	3-34
Parameters	3-34
YDSPMBRLST (Display Member)	3-35
Syntax Diagram	3-35
Parameters	3-36
YDSPPGMQ (Display a Program's Message Queue)	3-38
Syntax Diagram	3-38
Parameters	3-38
YDSPPNL (Display Panel Design)	3-39
Syntax Diagram	3-39
Parameters	3-39
YDSPUSRPRF (Display User Profile)	3-41
Syntax Diagram	3-41
Parameters	3-41
YEDTCKYTBL (Edit Command Key Table)	3-42
Syntax Diagram	3-42
Parameters	3-42
YEDTDBFLST (Edit DBF List)	3-43
Syntax Diagram	3-43

Parameters	3-44
YEDTDSSNDFT (Design Default Data Areas)	3-44
Syntax Diagram	3-44
Parameters	3-45
YEDTDTAARA (Edit Data Area Contents)	3-46
Syntax Diagram	3-47
Parameters	3-47
YEDTFMTLST (Edit Format List)	3-48
Syntax Diagram	3-48
Parameters	3-49
YEDTGDA (Edit Group Data Area)	3-49
Syntax Diagram	3-49
Parameters	3-49
YEDTLDA (Edit Local Data Area)	3-50
Syntax Diagram	3-50
Parameters	3-50
YEDTLIBLST (Edit Library List)	3-51
Syntax Diagram	3-51
Parameters	3-51
YEDTMBRLST (Edit Member List)	3-52
Syntax Diagram	3-52
Parameters	3-52
YEDTMSGD (Edit Message Description)	3-53
Syntax Diagrams	3-53
Parameters	3-54
YEDTOBJLST (Edit Object List)	3-54
Syntax Diagram	3-54
Parameters	3-55
YEDTPWDVAL (Edit Password Values)	3-55
Syntax Diagram	3-56
Parameters	3-56
YENDSPLRTR (End Spooled File Router)	3-56
Parameters	3-56
YEXCCL (Execute CL Member)	3-57
Syntax Diagram	3-57
Parameters	3-58
YEXCDBFLST (Execute Database File List)	3-58
Syntax Diagram	3-59
Parameters	3-59
YEXCFTP (File Transfer Protocol)	3-61
Syntax Diagram	3-62
Parameters	3-62
YEXCMBRLST (Execute Member List)	3-66

Syntax Diagram	3-67
Parameters	3-68
YEXCOBJLST (Execute Object List)	3-71
Syntax Diagram	3-72
Parameters	3-73
YFLTDBFLST (Filter Database List)	3-76
Syntax Diagram	3-77
Parameters	3-78
YFLTFMTLST (Filter Format List)	3-82
Syntax Diagram	3-83
Parameters	3-84
YFLTMBRLLST (Filter Member List)	3-87
Syntax Diagram	3-87
Parameters	3-89
YFLTOBJLST (Filter Object List)	3-95
Syntax Diagram	3-95
Parameters	3-98
YGO (Go to Menu)	3-106
Syntax Diagram	3-107
Parameters	3-108
YGRTPRDAUT (Grant Product Authority)	3-110
Syntax Diagram	3-111
Parameters	3-111
YINXLST (Index List)	3-111
Syntax Diagram	3-112
Parameters	3-112
YMOVLST (Move List)	3-114
Syntax Diagram	3-114
Parameters	3-114
YMOVMM (Move Member)	3-115
Syntax Diagram	3-116
Parameters	3-117
YMOVOBJ (Move Objects)	3-119
Syntax Diagram	3-120
Parameters	3-121
YMOVOBJSRC (Move Objects & Source)	3-124
Syntax Diagram	3-124
Parameters	3-126
YMVY1DTA (Move User Data Objects)	3-131
Syntax Diagram	3-131
Parameters	3-131
YOPRLST (Operate on List)	3-132
Syntax Diagram	3-132

Parameters	3-133
YRMVLE (Remove Library List Entry)	3-136
Syntax Diagram	3-136
Parameters	3-137
YRMVM (Remove Member)	3-138
Syntax Diagram	3-139
Parameters	3-139
YRNMLIB (Rename Library)	3-140
Syntax Diagram	3-140
Parameters	3-140
YRNMLIBLST (Rename Library List)	3-142
Syntax Diagram	3-143
Parameters	3-143
YRNMLLE (Rename Library List Entry)	3-143
Syntax Diagram	3-144
Parameters	3-144
YRNMMNU (Rename Menu)	3-146
Syntax Diagram	3-146
Parameters	3-146
YRNMOBJSRC (Rename an Object and Source)	3-147
Syntax Diagram	3-147
Parameters	3-147
YRNMPNL (Rename Panel Design)	3-148
Syntax Diagram	3-149
Parameters	3-149
YRNM RTP (Rename Report Design)	3-150
Syntax Diagram	3-150
Parameters	3-150
YRNMUSRPRF (Rename User Profile)	3-151
Syntax Diagram	3-151
Parameters	3-151
YRSTSPLRTR (Restart Spooled File Router)	3-152
Syntax Diagram	3-153
Parameters	3-153
YRTVMSGF (Retrieve Message File)	3-154
Syntax Diagram	3-154
Parameters	3-154
YRTVOBJLIB (Retrieve Object Library)	3-159
Syntax Diagram	3-160
Parameters	3-160
YRTV PN L D S N (Retrieve Panel Design)	3-161
Syntax Diagram	3-162
Parameters	3-162

YRTVRPTDSN (Retrieve Report Design)	3-164
Syntax Diagram	3-164
Parameters	3-164
YRTVSPLFA (Retrieve Spooled File Attributes)	3-165
Parameters	3-166
YRTVUSRPF (Retrieve User Profile).....	3-167
Syntax Diagram	3-168
Parameters	3-168
YRUNSQL (Start Interactive SQL Session)	3-172
Parameters	3-172
YSCNRPLSRC (Scan/Replace Source Member)	3-175
Syntax Diagram	3-176
Parameters	3-177
YSCNSRC (Scan Source Member)	3-180
Syntax Diagram	3-180
Parameters	3-182
YSETBRKPGM (Set Break Program)	3-188
Syntax Diagram	3-188
Parameters	3-188
YSTRCVTPRT (Start Convert Print Key Output Mode)	3-190
Syntax Diagram	3-190
Parameters	3-190
YSTRDBG (Start Debug and Add Auto Breakpoints)	3-192
Syntax Diagram	3-192
Parameters	3-192
YSTRSPLRTR (Start spooled file router)	3-195
Parameters	3-195
YTDYRPFSCR (Tidy RPG III Source)	3-200
Syntax Diagram	3-200
Parameters	3-200
YTRNPF (Translate Physical File Data)	3-202
Syntax Diagram	3-203
Parameters	3-203
YTRNSRCF (Translate Source File Data)	3-204
Syntax Diagram	3-205
Parameters	3-206
YWRKF (Work with Database File Data)	3-208
Syntax Diagram	3-208
Parameters	3-209
YWRKLIBLST (Work with Library List)	3-212
Syntax Diagram	3-213
Parameters	3-213
YWRKMNU (Work with Menus).....	3-214

Syntax Diagram	3-214
Parameters	3-214
YWRKPNL (Work with Panel Design)	3-215
Syntax Diagram	3-216
Parameters	3-216
YWRKRPT (Work with Report Design)	3-217
Syntax Diagram	3-217
Parameters	3-217
YWRKSPLRTE (Work with Spooled File Routing Entries)	3-218
YWRUSRPRF (Work with User Profile)	3-218
Syntax Diagram	3-218
Parameters	3-218

Appendix A: Expanded Parameter Definitions

LST and LSTTYPE Parameters	A-1
OBJLST, MBRLST, FMTLST and DBFLST Parameters	A-2
Default List Names	A-2
Work Lists	A-2
SELECT Value for a List Name	A-3
LSTOPT Parameter	A-3
OUTLST Parameter	A-4
FLAGVAL, OUTFLAGVAL, and UPDLST Parameters	A-4
OUTFLAGVAL Parameter	A-5
UPDLST Parameter	A-6
BUILD and EDIT Parameters	A-7
RQSDTA, USROPT, OPTFIL and OPTMBR Parameters	A-7
User Option QAUOOPT File	A-8
Substitution Symbols	A-8
Internationalization of Special Substitution Symbols	A-9
MENU Parameter	A-9
Referring to Menus	A-9
PANEL Parameter	A-10
Referring to Panel Designs	A-10
REPORT Parameter	A-10
Referring to Report Designs	A-11
CHGPRTDFT Parameter	A-11

Appendix B: Required Application Objects

Obtain Required Object Lists	B-1
------------------------------------	-----

Appendix C: Long Running Commands

Appendix D: Monitored Error Messages

Commands Using a Database File List	D-1
Commands Using a Member List	D-2
Commands Using an CA 2E Object List	D-2
YADDDSNFM	D-2
YBLDLIBLST	D-2
YWRKF	D-3
YCHGOBJOWN	D-3
YCHKLSTE	D-3
YCHKLIBLST (And All CA 2E Library List Commands)	D-3
YCHKMNU (And All CA 2E Menu Commands)	D-4
YCHKPNL (And All CA 2E Panel Design Commands)	D-4
YCHKRPT (And All CA 2E Report Design Commands)	D-4
YCPYLIBLST	D-4
YCPYF	D-5
YCRTDSNF	D-5
YCRTDUPOBJ	D-5
YCRTOBJ	D-6
YDLTOBJ	D-6
YEXCDBFLST	D-6
YEXCMBRLST	D-6
YEXCOBJLST	D-6
YMOVOBJ	D-7
YMOVM	D-7
YxxxLLE	D-7

Appendix E: Command and Keyword Abbreviations

Index

Chapter 1: Introduction

CA 2E is an integrated package of software utilities for the IBM i. This chapter describes the arrangement of this manual and the documentation presentation standards. Related publications are also listed.

Documentation

The documentation for the CA 2E utilities are divided into two manuals:

- *CA 2E Concepts Guide*, giving a conceptual overview of the utilities, and how they link together.
- *Reference Guide*, containing detailed explanations of each of the CA 2E commands that run the utilities.

This manual is the *CA 2E Toolkit Reference Guide*. We recommend that you read the *CA 2E Concepts Guide* before referring to the reference guide.

Additional documentation is available in the form of *Online Help Text*. All of the interactive CA 2E utility programs have operating instructions, which display by selecting Help while using the programs.

Arrangement of this Manual

Each CA 2E command is documented separately. The manual is arranged in alphabetical order by command names.

There are technical appendices at the end of the manual, giving information that is common to all or many of the commands, as well as information that may be useful if you wish to make use of the commands in your own programs.

Related Publications

- CA 2E Concepts Guide
- CA Standards Guide
- *IBM i Programming: Control Language Programmer's Guide*
- *IBM i Programming: Control Language Reference*
 - Volume 1

- Volume 2
- Volume 3
- Volume 4
- Volume 5
- *IBM i Programming: Data Description Specifications*
- *IBM i Text Management/38 User's Guide and Reference Manual*

Documentation Presentation Standards

IBM has developed a documentation convention for the IBM i Control Language (CL) that, once learned, is both concise and precise. Central to the convention is a special notation for explaining i OS commands - the Command Syntax Diagram.

All of the CA 2E utilities are accessible via CL commands, and each command is documented strictly according to the i OS convention. For each command there are:

- Definition the function command
- Description of each parameter in the command
- Notes on any restrictions or extra considerations
- An example or examples of using the command

Some parameters, common to several commands, are described centrally in the Appendix "Expanded Parameter Definitions."

For those experienced with CA 2E, the command syntax diagrams and their parameters can be the most useful. For those are not familiar with command diagrams, refer to the *IBM i Control Language Reference Manual Volume 1, Chapter 2*, for a detailed description of the CL command syntax conventions.

Wherever possible, existing i OS parameter keywords and definitions have been used. This manual can be regarded as an extension of the i OS manual. Appendix E contains a list of all the CA 2E abbreviations and keywords that are additional to the i OS canon.

Calling CA 2E Commands

CA 2E commands can either be entered from any program or IBM display that allows the entry of commands, or via the CA 2E Help menus, which can be displayed using the CA 2E command Go To Menu YGO:

YGO MENU (*Y1)

All CA 2E commands have the CA 2E utility product library (NOCHG) as attribute Product Library. Use the i OS command Add Library List Entry (ADDLIBLE) to add the CA 2E library to your library list as follows:

ADDLIBLE LIB (Y1SY) POSITION (*LAST)

Alternatively, the product library can be added permanently to the library list using the i OS command Change System Value (CHGSYSVAL).

Data Object Commands

When running commands that use CA 2E data objects, ensure that the data objects reside in the library list. Data objects can be created with the command Create Data Objects (YCRTIDTA).

CA 2E Modules

The utilities are divided into four modules:

- *ACS: User access aids
- *DSN: Design aids
- *PGMR: Programmer aids
- *DOC: Documentation aids

The utilities for all four of the CA 2E modules are shipped, regardless of which module(s) purchased. Only the utilities in the modules purchased are authorized for use. Adding additional modules can be supplied on request.

Each CA 2E command syntax diagram includes an indication of which module the command belongs to.

CA 2E Object Names

The names of all CA 2E objects begin with the letter **Y** and reside in library Y1SY, the CA 2E utility library.

All execution message definitions are contained in a message file called YYYYMSG. The message file resides in the CA 2E product library, except for the help and menu utilities. Help and menu utilities can be run independently (see CA 2E *YDUPAPPOBJ* command).

Abbreviated Command Names

Frequently used commands are available in an abbreviated form. These abbreviations are listed below, but abbreviations can be added. Abbreviated commands are the only CA 2E objects that do not have names beginning with the letter Y.

Command	Abbreviated Form
YCHGLIBL	R
YCHGJOB DLL	RJ
YDSPABR	D
YEDTLIBLST	L
YGO	M
YSETBRKPGM	B

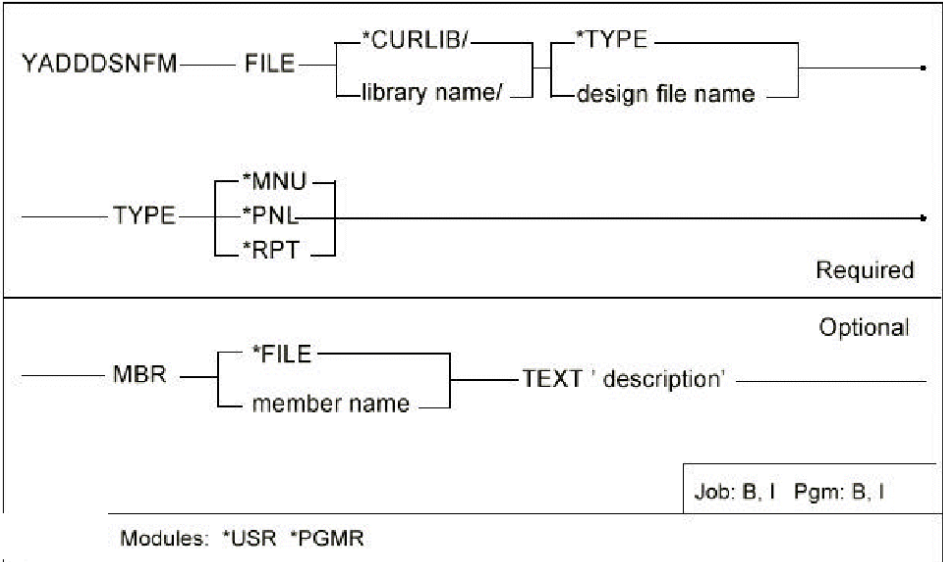
Chapter 2: 2E Commands, YA to YDOCMSGREF

This chapter lists the various commands available with CA2E from YADDDSNFM to YDOCMSGREF.

YADDDSNFM (Add Design File Member)

This command adds a member to a design file with the correct format and any required dependent logical views.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
FILE	Qualified name of file where members are added. If specified, the name must not exceed nine characters.	TYPE: use default file name for type; if type is: <ul style="list-style-type: none"> ■ *MNU: YDSNMNU ■ *PNL: YDSNPNL ■ *RPT: YDSNRPT, a library must be specified
TYPE	CA2E design file created	<ul style="list-style-type: none"> ■ *MNU: menu file ■ *PNL: panel design file ■ *RPT: report design file
MBR	Name of member in file	*FILE: (default) member name is the same as file name
TEXT	Descriptive text for member	

Notes

None

Example

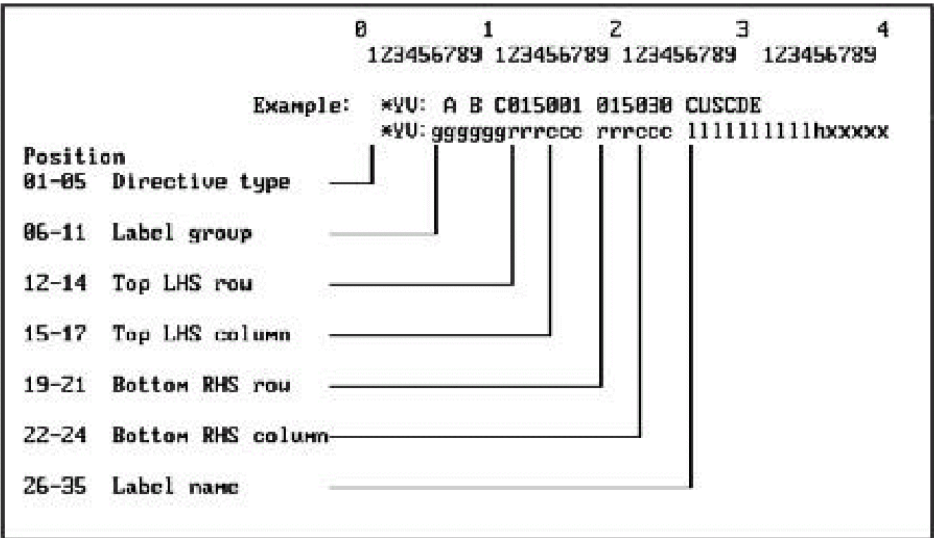
To add a new panel design member NEWMBR to panel file YDSNPNL in library CECIL:

```
YADDSDNFM FILE(CECIL/YDSNPNL) TYPE(*PNL)
MBR(NEWMBR) TEXT('New member')
```

YADDHLPTBL (Add Help Vector Table)

Builds a Help vector table from the DDS for a Display file and adds it to a source file member. The Help vector table can be used to drive the field sensitive help text facility of Display Help Text (YDSPHLP). Output from the YADDHLPTBL command is written to a source file that can be modified and included in the help text.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
SRCFILE	Qualified name of the source file containing the DDS of the display file for which a vector table is generated.	*LIBL/QDDSSRC: (default) Source file name MBRLST: The MBRLST parameter specifies the name of a member list containing the members to be processed.
SRCMBR	Generic or specific member's name in the source file selected for processing.	*ALL: (default) All members in the file are processed
TOFILE	Qualified name of the output file containing the member for the resulting vector table.	*CURLIB/QTXTSRC: (default) Output file name
TOMBR	The name of the member in the output file holding the derived vector table	*SRCMBR: (default) The name of the output member is the same as that of the source file member; if the member does not exist, it is created

Parameter	Definition	Value and Description
MBROPT	Determines whether output replaces or is appended to any existing data	<ul style="list-style-type: none"> ■ *REPLACE: (default) Replace existing member ■ *ADD: Add to the contents of any existing member
CRTFILE	Determines whether the output file is created. An error is displayed if the output file does not exist and CRTFILE(*NO) is specified	<ul style="list-style-type: none"> ■ *NO: (default) Do not create the output file. ■ *YES: If the output file does not exist, it is created
MBRLST	Qualified name of a member list	QTEMP/TEMPLST: (default) Member list name
EDIT	Edit list option	<ul style="list-style-type: none"> ■ *NO: (default) No editing is required ■ *YES: Invoke the Edit Member List function to edit the list before execution

Notes

- The YADDHLPTBL command is subject to a number of size considerations:
 - Not more than 10 format names in each embedded YDFNPNDLSN or YDFNSCRDSN command.
 - Not more than 30 YDFNxxxDSN commands.
 - Not more than 100 record formats in each DDS source file member.
- Additionally, further considerations when using the YADDHLPTBL command are:
 - The YADDHLPTBL command makes a compilation of the DDS source. It is possible for a DDS compilation to fail. The most common causes of failure are either an incorrect library list (that is, the source can refer to a field reference file not in the library list), or that the source member is not a display file. If the compilation fails, the compilation listing is left on an output queue and a warning message is issued to the panel or the job log.

In general, any keyword that can change a field location dynamically is ignored. To aid problem determination a warning message is added to the source file output where appropriate.
 - The following DDS keywords are ignored, and a warning message issued:
 - SFLDROP Subfile drop (fold or truncate subfile records)

- SFLIN Subfile line (used for a horizontal subfile)
- USRDFN User defined data stream
- The following DDS keywords are ignored, and a warning message is not issued:
 - ALWROL Allow roll
 - CLRL Clear line (erase lines)
 - DSPMOD Display mode
 - DSPSIZ Display size - always assumed to be 24 x 80
 - SLNO Starting line number
- Warning messages are also be inserted for:
 - Formats not containing fields
 - Message subfile record formats.
 - Message subfile control record formats.

Example

The following example shows some panel DDS and the vector tables generated from the YADDHLPTBL command.

DDS Display file source appears as follows:

A	R HEADER	TEXT('Header')
*	YDFNPNLDSN PANEL(ONE) FORMAT(HEADER SFLCTL SFL)	
A		Z 2' Description'
A	FIELD1	10A B 4 5
A N58	FIELD2	5A 0 4 22
A	FIELD3	5S 2H
A 58	FIELD4	5A I 4 23
=====		
A	R SFL	
A	SFL1	4A B 6 20
A	SFL2	1A B 6 30
=====		
A	R SFLCTL	SFLCTL(SFL)
A		SFLIN(5)
A		SFLPAG(20) SFLSIZ(21)
A 25		SFLDSP SFLDSPCTL
A N25		SFLINZ
A	CTL1	9A 0 5 20

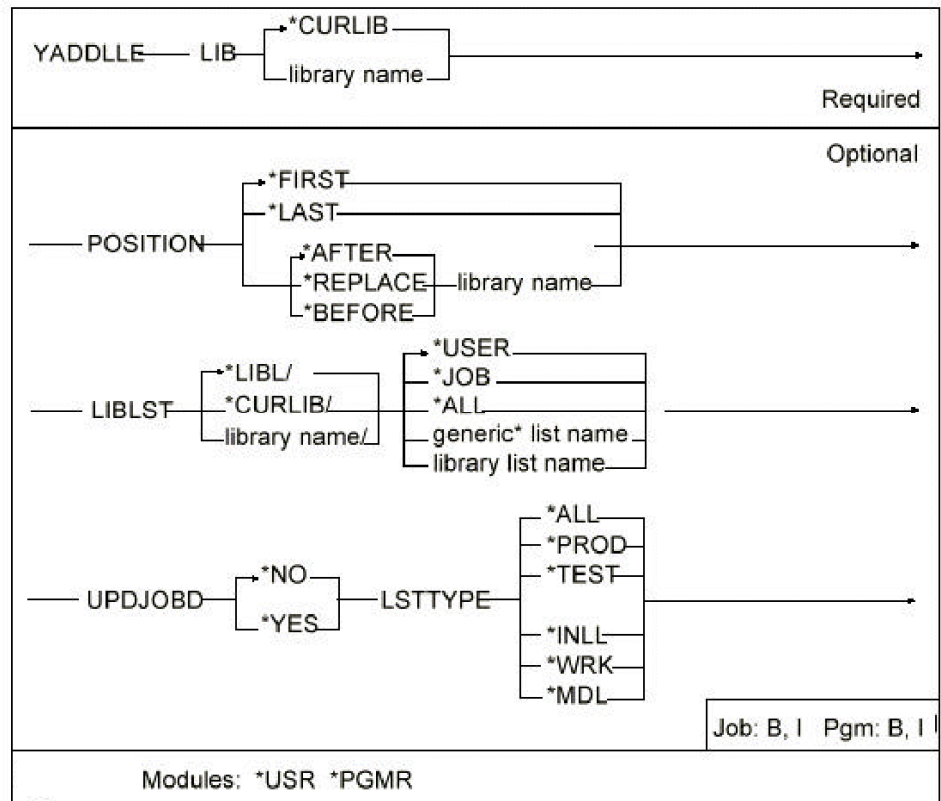
The Vector table (output of YADDHLPTBL) appears as follows:

```
* ****  
* Vector table for UUDEV/QDDSSRC member (HLP1ST)  
* ****  
* PANEL(ONE) FORMAT(HEADER SFLCTL SFL)  
*YU: X 004005 004014 LBL0001 Field-FIELD1 format-HEADER  
*YU: X 004022 004026 LBL0002 Field-FIELD2 format-HEADER  
*YU: X 004023 004027 LBL0003 Field-FIELD4 format-HEADER  
*YU: X 002001 004000 LBL0004 Format level entry-HEADER  
*YU: X 005020 005028 LBL0005 Field-CTL1 format-SFLCTL  
*YU: X 005001 005000 LBL0006 Format level entry-SFLCTL  
*YU: Warning: SFLIN keyword ignored. Subfile output may be incorrect  
*YU: X 006020 025023 LBL0007 Field-SFLF1 format-SFL  
*YU: X 006030 025030 LBL0008 Field-SFLF2 format-SFL  
*YU: X 006001 025000 LBL0009 Format level entry-SFL  
*YU: X 001001 024000 LBL0010 Panel level entry
```

YADDLLE (Add Library List Entry)

This command adds a command to a library or a library list or lists. If the library does not exist a warning message is issued, but the command is still added to the library list(s). If a library list already contains 25 libraries no additional libraries will be added. A completion message is returned indicating the number of library lists to which the library has been added.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
LIB	Name of the library to be added to the library list	*CURLIB: Add job's current library to all lists specified by LIBLST parameter
POSITION	The destination of the library list specified by the LIB parameter	<ul style="list-style-type: none"> ■ *FIRST: (default) Add library to the beginning of the specified library list ■ *LAST: Add library to the end of the specified library list <p>Otherwise POSITION is a list parameter made up of the following two elements:</p> <ul style="list-style-type: none"> ■ List position <ul style="list-style-type: none"> – *AFTER: (default) Add library after the reference library given in the second element – *BEFORE: Add library before the reference library – *REPLACE: Replace the reference library in the destination library list with the library specified by the LIB parameter ■ Reference library <ul style="list-style-type: none"> – Name specified by the LIB parameter that lists where to place the library list
LIBLST	Qualified generic name of library list(s) where the library is added	<ul style="list-style-type: none"> ■ *USER: (default) Library list name is the same as the current user profile ■ *JOB: Library list name is the same as the current job ■ *ALL: Add library to all library lists in specified library
UPDJOB	Update job description associated with the library list	<ul style="list-style-type: none"> ■ *YES: (default) Update initial library list of job description with libraries specified in the list. ■ *NO: Do not update job description

Parameter	Definition	Value and Description
LSTTYPE	Library list type	<ul style="list-style-type: none"> ■ *ALL: (default) Select all library lists ■ *PROD ■ *TEST ■ *INLL ■ *WRK ■ *MDL

Notes

If UPDJOB(*YES) is specified for a library list without an associated job description, the parameter is ignored.

Examples

To add library MORGAN to library list NY:

```
YADDLLE LIB(MORGAN) LIBLST(NY)
```

To add library QGPL to all library lists beginning with 'NY':

```
YADDLLE LIB(QGPL) LIBLST(NY*)
```

To add library ENGLISH before library FRENCH to all library lists in QGPL:

```
YADDLLE LIB(ENGLISH) POSITION(*BEFORE FRENCH) LIBLST(QGPL/*ALL)
```

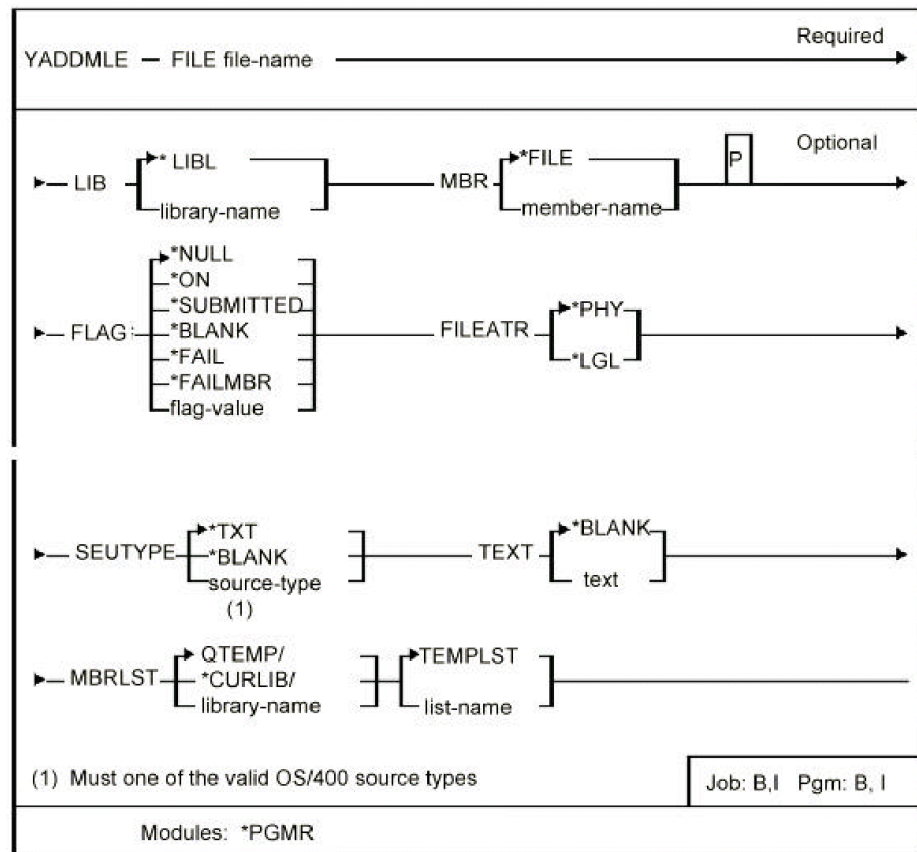
To add library GERMAN to all lists of type TEST, updating initial library list of all associated job descriptions to all library lists in QGPL:

```
YADDLLE LIB(GERMAN) LIBLST(QGPL/*ALL) UPDJOB(*YES) LSTTYPE(*TEST)
```

YADDMLE (Add Member List)

This command adds one entry to a member list and the entry details are taken from the command and not from any physical member.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
FILE	Name of file on the member list entry	
LIB	Name of library on the member list entry	*LIBL: (default) The library name is *LIBL on the member list entry
MBR	Name of member on the member list entry	*FILE: (default) The member name is the same as that given by the FILE parameter
FLAG	Flag value on the member list entry	Single character flag value or one of the special flag values

Parameter	Definition	Value and Description
FILEATR	File attribute on the member list entry	<ul style="list-style-type: none"> ■ *PHY: (default) Physical file ■ *LGL: Logical file
SEUTYPE	Source type on the member list entry	<ul style="list-style-type: none"> ■ *TXT: text ■ *BLANK: Source type is blank (enforced if FILEATR parameter is *LGL) ■ Or one of the valid i OS source types (native or S38E)
TEXT	Text on member list entry	*BLANK: (default) no text for member
MBRLST	Qualified name to which the entry is added	QTEMP/TEMPLST: (default) list name

Notes

1. Restriction: The entry added to the member list does not contain the following information:

- MLNOMB Number of members in file
- MLNRCD Number of active records
- MLNDTR Number of deleted records
- MLSIZE Current size of member
- MLCCEN Creation century
- MLCDAT Creation date
- MLCHGC Last changed century
- MLCHGD Last changed date
- MLCHGT Last changed time

If, using the command Filter Member List (YFLTMBRLST), you attempt to filter any above attributes, unpredictable results can occur.

2. The following information is derived:

- MLRDAT Retrieval Date from System Date
- MLRTIM Retrieval Time from System Time
- MLFTYP Type of File from File Attribute
- MLSEU Member Short Source Type from Long Source Type
- MLFATR From File Attribute ('PF' or 'LF')

Example

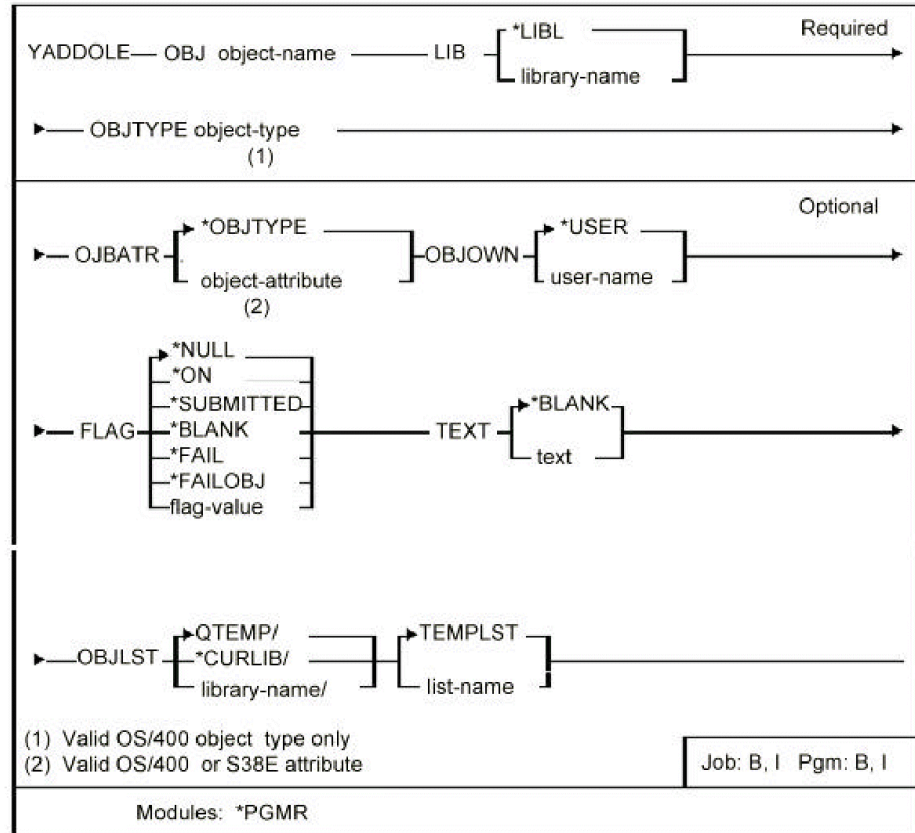
Add a single entry to member list FRED in library QTEMP:

```
YADDMLE FILE(QCLSRC) LIB(MYLIB) MBR(TEST) SEUTYPE(*CLP) MBRLST(QTEMP/FRED)
```

YADDOLE (Add Object List Entry)

Adds one entry to an object list and the entry details are taken from the command and not from any physical object.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
OBJ	Name of the object listed in the object list entry	
LIB	Name of library in the object list entry	*LIBL: the library name is *LIBL on the object list entry
OBJTYPE	Valid i OS object type in the object list entry	
OBJATR	i OS object attribute in object list entry	<ul style="list-style-type: none"> ■ *OBJTYPE: (default) Object attribute is derived as follows: <ul style="list-style-type: none"> – If object type is *PGM, then OBJATR is CLP. – If object type is *FILE, then OBJATR is PF. – Otherwise OBJATR is blank
OBJOWN	Owner in object list entry	*USER: (default) Owner is the current user
FLAG	Flag value in the object list entry	Single character flag value or one of the special flag values
TEXT	Text in object list entry	*BLANK: (default) No text for object list entry
OBJLST	Qualified name of object where entry is added	QTEMP/TEMPLST: (default) List name

Notes

- The added entry only contains the information specified by the command.
If using the command Filter Object List (YFLTOBJLST), and filtering any of the above attributes, unpredictable results can occur.
- The following information is derived:
 - ODDDAT Retrieval Date from System Date
 - ODDTIM Retrieval Time from System Time

Example

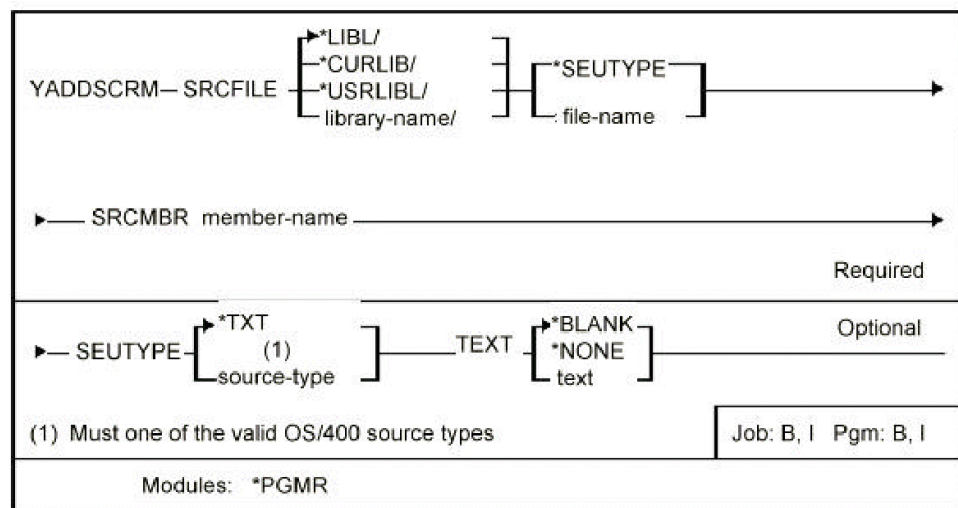
Add a single entry to object list FRED in library QTEMP:

```
YADDOLE OBJ(MYPGM) LIB(*LIBL) OBJTYPE(*PGM) OBJATR(*RPG) OBJLST(QTEMP/FRED)
```

YADDSRCM (Add Source Member)

This command adds a member to a source file; the source type of the member can be specified.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
SRCFILE	Qualified name of the source file where the member is added	*SEUTYPE: determines source file name from source type. If the source type is RPG use QRPGRSRC
SRCMBR	Name of added member	
SEUTYPE	Source type to be given to new member	Must be one of the valid i OS source types (native or S38E)
TEXT	Descriptive text for member	*BLANK: (default) no text for member

Notes

- 1. The nominated file must already exist.
- 2. If SRCFILE(*SEUTYPE) is specified, but an appropriate source file cannot be determined from the source type, an escape message is sent.

Example

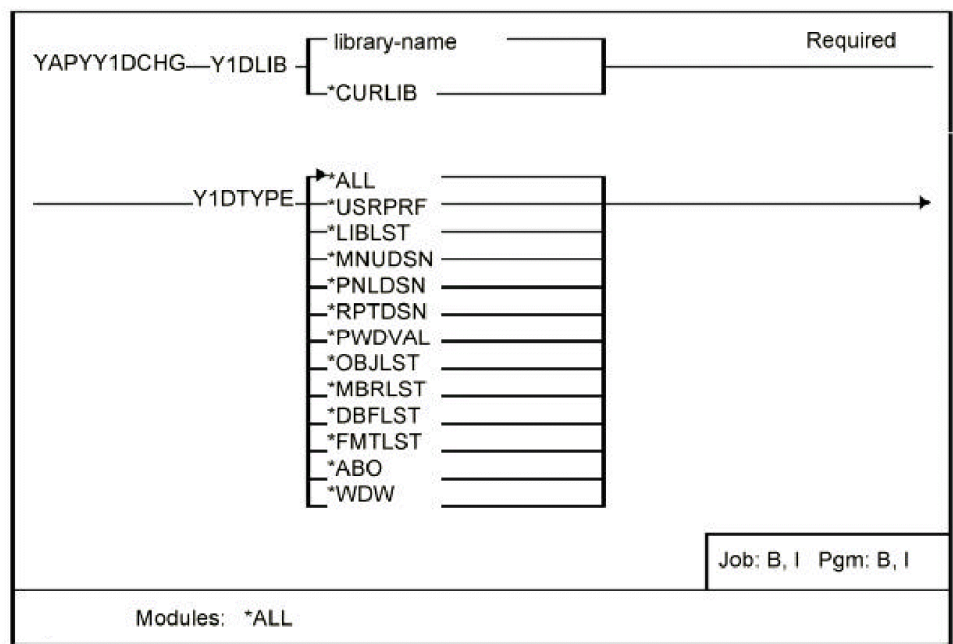
To add member ADDITION of type CL to the source file QCLSRC in library XTRA:

```
YADDSRCM FILE(XTRA/*SEUTYPE) SRCMBR(ADDITION) SEUTYPE(*CL)
```

YAPYY1DCHG (Apply Data Changes)

Updates data objects with any changes required by a new release. New release modifications must be applied to each data object before running some utilities on those data objects. This command affects any changes to named data objects in a named library.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
Y1DLIB	Name of the library containing the data objects to be changed.	*CURLIB: Use the current library for the job. Data objects that have already been converted, or that do not need to be converted are ignored.
Y1DTYPE	Type of data object to change in the given library	<ul style="list-style-type: none">■ *ALL: (default) Changes all data objects■ *USRPRF: Changes the user profile extension file■ *LIBLST: Changes the library list file■ *MNUDSN: Changes all menu design files■ *PNLDSN: Changes all panel design files■ *RPTDSN: Changes all report design files■ *PWDVAL: Changes the forbidden passwords file■ *OBJLST: Changes the object lists■ *MBRLST: Changes the member lists■ *DBFLST: Changes the database file lists■ *FMTLST: Changes all format lists■ *ABO: Changes all action bar objects■ *WDW: Changes all window source

Notes

1. Back up the data library before running this command.
2. Object existence rights are required for the objects being modified.
3. This command is cumulative. If objects are archived, and then restored, YAPYY1DCHG can be used to bring the objects up to date.

Example

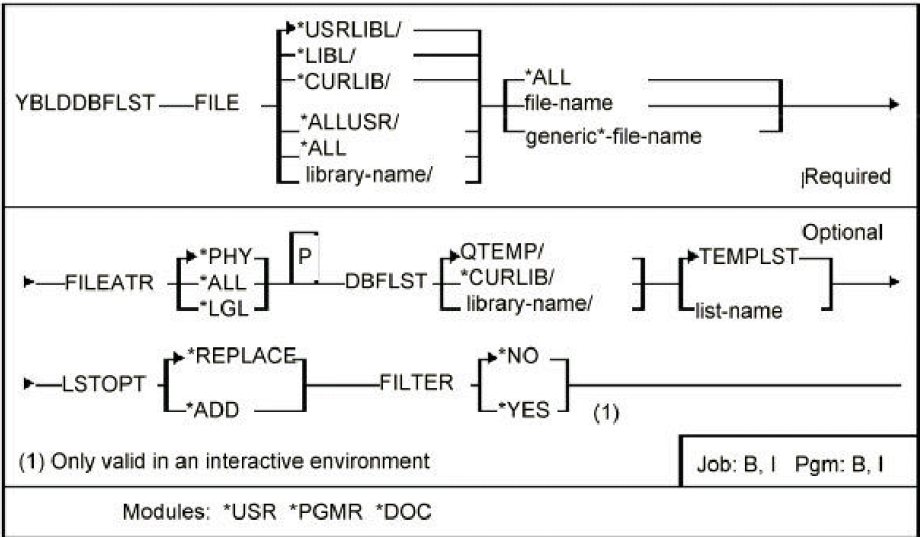
To update the library list file in library QGPL:

```
YAPYY1DCHG Y1DLIB(QGPL) Y1DTYPE(*LIBLST)
```

YBLDDBFLST (Build Database File List)

This command builds a list of database files that can be reused in other commands.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
FILE	Qualified generic file name of files included in the list.	*ALL: All files in nominated library or libraries
FILEATR	File attribute of database files which are included in the list.	<ul style="list-style-type: none">■ *PHY: (default) Includes physical files only■ *LGL: Includes logical files only■ *ALL: Includes both physical and logical files in list
DBFLST	Qualified name of dbf list that is to be built	QTEMP/TEMPLST: (default) List name
LSTOPT	List replacement option	<ul style="list-style-type: none">■ *REPLACE: (default) Creates a new list or replace the contents of any existing list■ *ADD: Add to any existing list's contents
FILTER	Filter option	<ul style="list-style-type: none">■ *NO: (default) Filter function is not invoked■ *YES: After building list invoke the command Filter Database File List (YFLTDBFLST)

Notes

If FILTER(*YES) is specified with FILEATR(*ALL) two prompts appear asking for the filter details: first for the filter to apply to the physical files, second for the filter to apply to the logical files.

Example

To build a new list of all physical files in library FRED whose names begin with RASP:

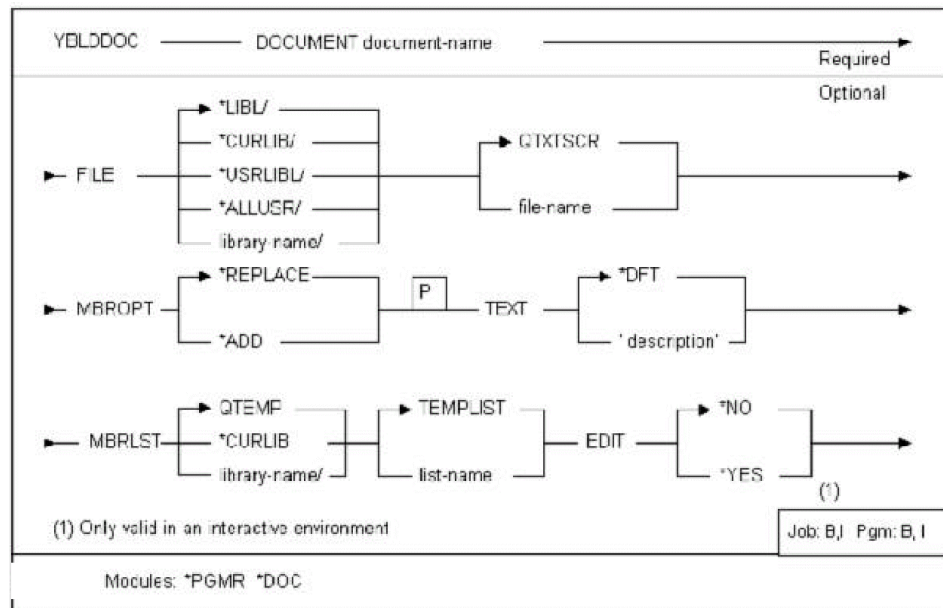
```
YBLDDBFLST FILE(FRED/RASP*) FILEATR(*PHY)
```

The list is called TEMPLST and resides in the library QTEMP.

YBLDDOC (Build Master Document)

Builds a source document from a member list. The document created contains Text Management/38 'imbed' references to all of the source members in the list, and therefore constitutes a 'master document' that can be used to run a print of all the referenced documents, using the i OS 'Text management/38' print function (QSYS38/PRTDOC).

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
DOCUMENT	Name of the master document (source file member) that is to be created/updated	
FILE	Qualified name of file to contain the master document that is created/updated	*LIBL/QXTSRC: (default) Name for file
MBROPT	Member update option	<ul style="list-style-type: none"> ■ *REPLACE: (default) Replace any existing document ■ *ADD: The new document references are appended to the contents of any existing document
TEXT	Title for master document	*DFT: (default) The title defaults to "Master source document"
MBRLST	Qualified name of the member list that is converted into a master document	QTEMP/TEMPLST: (default) List name
EDIT	Edit list option	<ul style="list-style-type: none"> ■ *NO: (default) The edit list function is not invoked ■ *YES: Before building the master document, invoke the edit member list function

Notes

1. The following lines are generated at the beginning of each document:

```
. *T: <Document title>
. * Table of contents.
. tc
```

2. The following lines are generated for each document referenced:

```
. pa <page advance>
. * <included document title>
. in <<document name> <source file><library>>
```

Example

A member list named BODY, contains the following three items:

QGPL/QTXTSRC ARM	Help text for BODY's arms
QGPL/QTXTSRC LEG	Help text for BODY's legs
QGPL/QTXTSRC KNEE	Help text for BODY's knees

The following command converts the member list into a new master document called BODY, in file QGPL/QTXTSRC:

```
YBLDDOC DOCUMENT(BODY)
FILE(QGPL/QTXTSRC) MBRLST(BODY) TEXT('Body members')
```

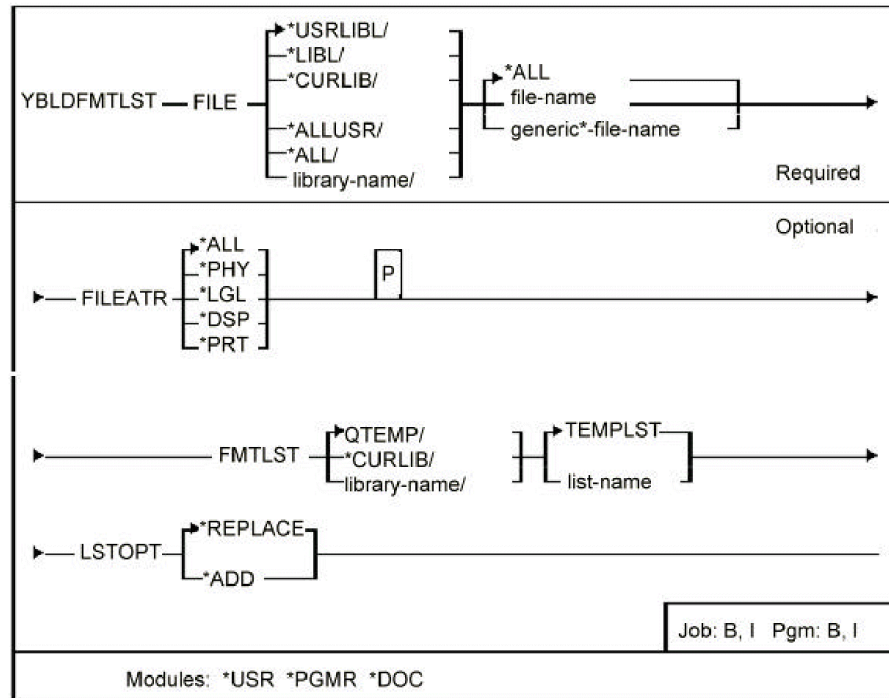
The following lines are generated in document BODY:

```
. *T: Body members
. * Table of contents.
. tc
. pa
. * Help text for BODY's arms
. in (ARM QTXTSRC.QGPL)
. pa
. * Help text for BODY's legs
. in (LEG QTXTSRC.QGPL)
. pa
. * Help text for BODY's knees
. in (KNEE QTXTSRC.QGPL)
```

YBLDFMTLST (Build Format List)

This command builds a list of the formats in a specified file or files. The list can then be used in other commands.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
FILE	Qualified generic file name of files containing formats which are included in the list	*ALL: All files in specified library
FILEATR	File attribute of files containing formats	<ul style="list-style-type: none"> ■ *ALL: (default) Include physical, logical, display and print files ■ *PHY: Include physical files only ■ *LGL: Include logical files only ■ *DSP: Include display files only ■ *PRT: Include print files only

Parameter	Definition	Value and Description
FMTLST	Qualified name of format list that is to be built	QTEMP/TEMPLST: (default) List name
LSTOPT	List replacement option	<ul style="list-style-type: none">■ *REPLACE: (default) creates a new list - or replace the contents of any existing list■ *ADD: adds to any existing list's contents

Notes

None

Example

To build a new list of all display file formats in library FRED:

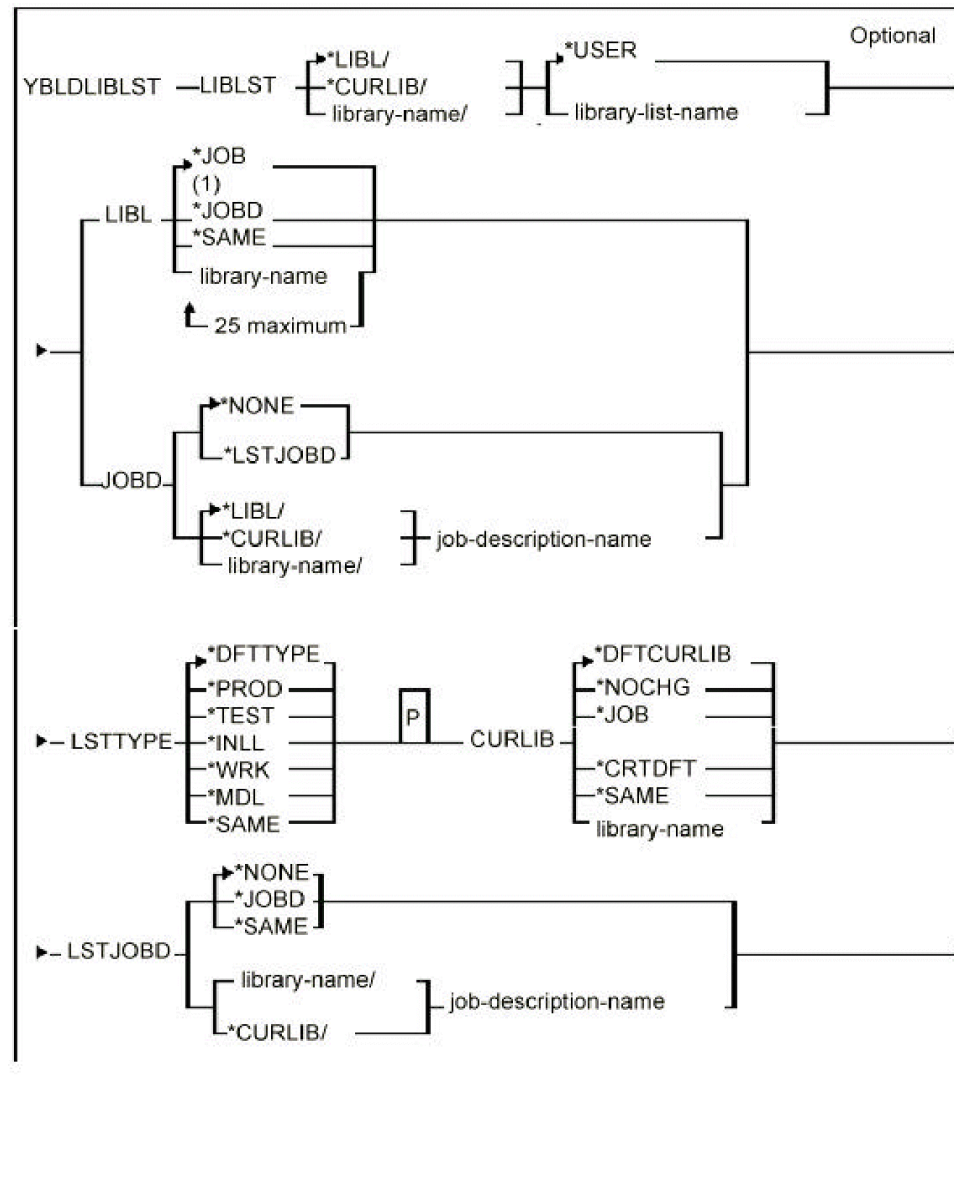
```
YBLDFMTLST FILE(FRED/*ALL) FILEATR(*DSP)
```

The list will be called TEMPLST and reside in QTEMP.

YBLDLIBLST (Build Library List)

This command builds or updates a library list from either a specified list of libraries, from the current job's library list, or from the initial library list of a specified job description. The library list can then be used in other commands.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
LIBLST	Qualified name of library list that is built or updated	*LIBL/*USER: (default) Library list has same name as the invoking job's user profile name. The library list is used to find the YLIBLST file
LIBL	Lists the libraries included in the built list (ignored if a value is specified for JOBD)	<ul style="list-style-type: none"> ■ *JOB: (default) Use current job's library list ■ *JOB: Use library list specified by the JOBD parameter ■ *SAME: Do not change list of libraries
JOBD	Qualified name of job description whose initial library list (as specified by the INLLIBL parameter on the job description) is used to build the list	<ul style="list-style-type: none"> ■ *NONE: (default) Use library list specified by the LIBL parameter ■ *LSTJOBD: Use job description specified by the LSTJOBD parameter
LSTTYPE	Library list type of the built library list	<ul style="list-style-type: none"> ■ *SAME: (default) Do not change the library list ■ *PROD ■ *TEST ■ *INLL ■ *WRK ■ *MDL ■ *DFTTYPE: Library list type is *INLL if library list is built from a JOBD or JOB; otherwise it is *WRK

Parameter	Definition	Value and Description
CURLIB	Name of the current library associated with the library list	<ul style="list-style-type: none"> ■ *DFTCURLIB: (default) If the library list is built from a JOB, use the current library; otherwise, *NOCHG ■ *NOCHG: Do not change current library when replacing with the built library list ■ *JOB: Use the current library in the current job's library list as the current library associated with the library list ■ *SAME: Do not change the current library list ■ *CRTDFT: The current library is *CRTDFT when replacing the built library list
LSTJOB	Qualified name of job description associated with the library list	<ul style="list-style-type: none"> ■ *NONE: (default) Do not associate any job description with the library list ■ *JOB: Associate the job description, specified in the JOB parameter, with the library list ■ *SAME: Do not change the library list job description
UPDJOB	Update library list's job description specified in the LSTJOB parameter	<ul style="list-style-type: none"> ■ *YES: (default) The initial library list with job descriptions is updated with the libraries specified in the list ■ *NO: Do not update the job description
TEXT	Descriptive text for list (up to 50 characters)	<ul style="list-style-type: none"> ■ *DFTTXT: (default) Provide default text ■ *NONE: Give descriptive text value *BLANK ■ *SAME: Do not change descriptive text

Notes

1. To obtain the initial library list of a job description the YBLDLIBLST command submits a job-to-job queue QINTER. This causes the job to enter the QINTER subsystem; QINTER must therefore be active for the command to function. There may be a slight delay while the job that builds the library lists executes and the list is created. A message appears on the workstation message queue when the list has been built.
2. Library lists are stored in a database file called YLIBLST in the specified library. The file must already exist in the specified library.
3. If a non-existent library is included in the library list, the list is still created/updated, but a warning message is issued.

Examples

To build a library list called FRED from the current job's library list:

```
YBLDLIBLST LIBLST(FRED) TEXT('Fred''s library list')
```

To build a library list called SEPP0 from a specified list of libraries:

```
YBLDLIBLST LIBLST(SEPP0) TEXT('Seppo''s library list') LIBL(QTEMP QGPL SEPP0  
QRPG QTXT)
```

To build a library list called BASIL from the initial library list of job description ALGERNON:

```
YBLDLIBLST LIBLST(BASIL) TEXT('Basil''s library list') OBD(ALGERNON)
```

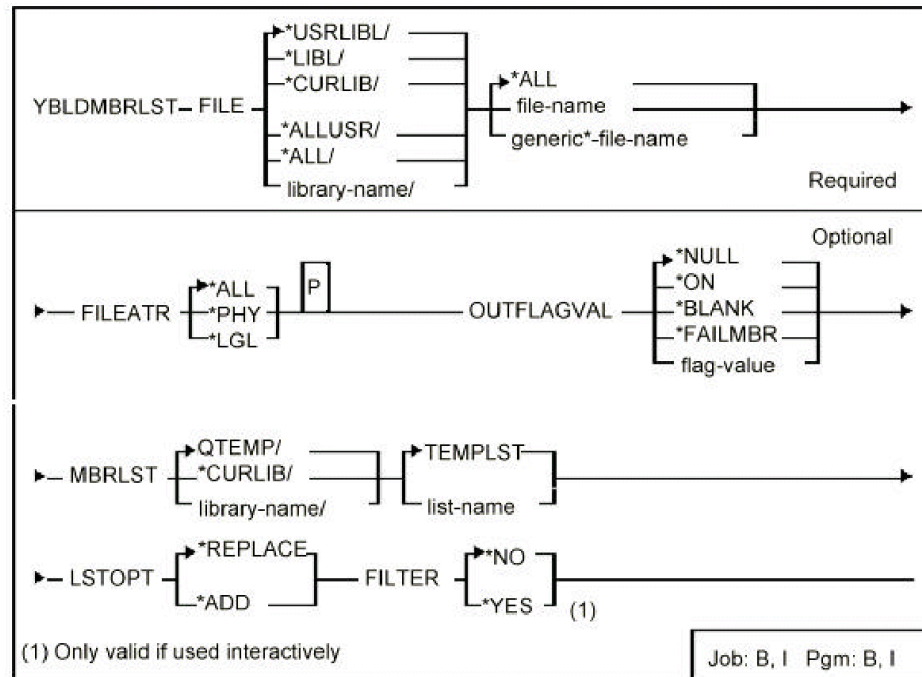
To build a library list named MIKE from job description MIKE in library QGPL, specifying that this job description is to be associated with this library list, to have current library CHRIS, and to be for a test environment:

```
YBLDLIBLST LIBLST(MIKE) JOBD(QGPL/MIKE) LSTTYPE(*TEST) CURLIB(CHRIS)  
LSTJOBD(*JOBD)
```

YBLDMBRLST (Build Member List)

This command builds a list of the members in a specified database file or files. The list can then be used in other commands.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
FILE	Qualified generic file name of database files containing members included in the list	<ul style="list-style-type: none"> *ALL: (default) All files in specified library
FILEATR	File attribute of files containing members	<ul style="list-style-type: none"> *ALL: (default) Include both logical & physical files *PHY: Include physical files only *LGL: Include logical files only
OUTFLAGVAL	Flag value given to entries in the list	*NULL: (default) Set flag to null value Single character flag value or one of the special flag values
MBRLST	Qualified name of member list that is built	QTEMP/TEMPLST: (default) List name

Parameter	Definition	Value and Description
LSTOPT	List replacement option	<ul style="list-style-type: none"> ■ *REPLACE: (default) Creates a new list or replace the contents of any existing list ■ *ADD: Add to any existing list's contents
FILTER	Filter option	<ul style="list-style-type: none"> ■ *NO: (default) Filter function is not invoked ■ *YES: After building the list invoke the command Filter Member List (YFLTMBRLST)

Notes

The Filter Member List (YFLTMBRLST) can be used to select the member name etc.

Example

To build a new list of all physical file members in library FRED, flagging each list entry with '*ON':

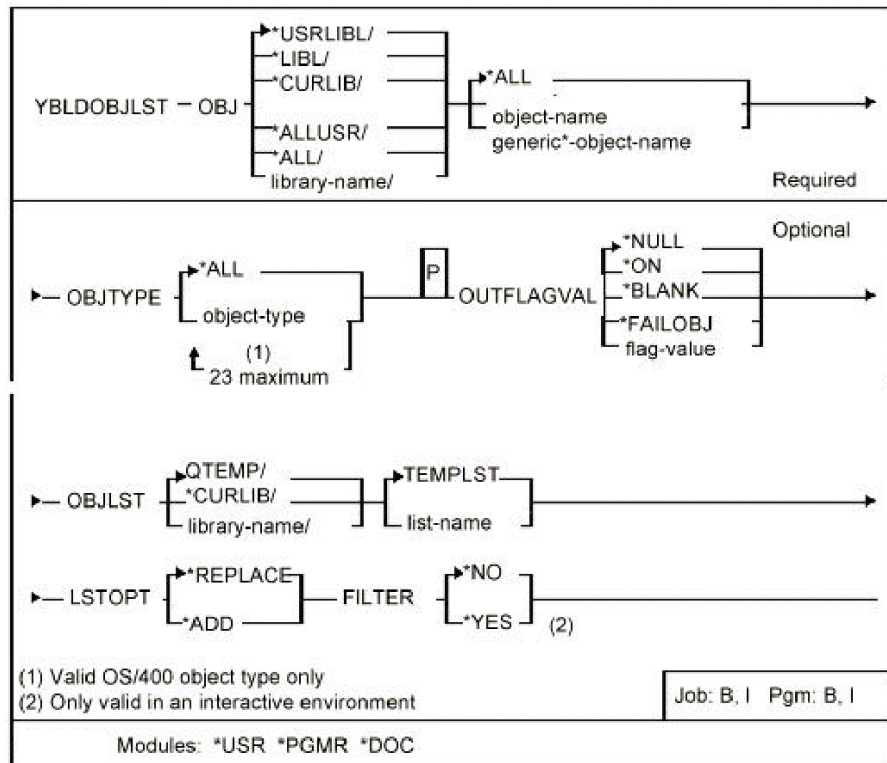
```
YBLDMBRLST FILE(FRED/*ALL) FILEATR(*PHY) +
OUTFLAGVAL(*ON)
```

The list will be called TEMPLST and reside in QTEMP.

YBLDOBJLST (Build Object List)

This command builds a list of objects that can then be used in other commands.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
OBJ	Qualified generic object name of the objects included in the list	*ALL: (default) Includes all objects in the specified libraries
OBJTYPE	List of i OS object types of objects which are to be included in object list	*ALL: (default) Includes all object types
OUTFLAGVAL	Flag value to be given to entries in the list	*NULL: (default) Set flag to null value Single character flag value or one of the special flag values
OBJLST	Qualified name of object list that is to be built	QTEMP/TEMPLST: (default) Name for list

Parameter	Definition	Value and Description
LSTOPT	List replacement option	<ul style="list-style-type: none"> ■ *REPLACE: (default) Create a new list, replacing any previous list's contents ■ *ADD: Add to any existing list's contents
FILTER	Filter option	<ul style="list-style-type: none"> ■ *NO: (default) Filter function is not invoked ■ *YES: After building the list, invoke the Filter Object List (YFLTOBJLST)

Notes

None

Example

To build a new list of all programs in library TVTIMES whose names begin with 'BBC', flagging each entry with 'B':

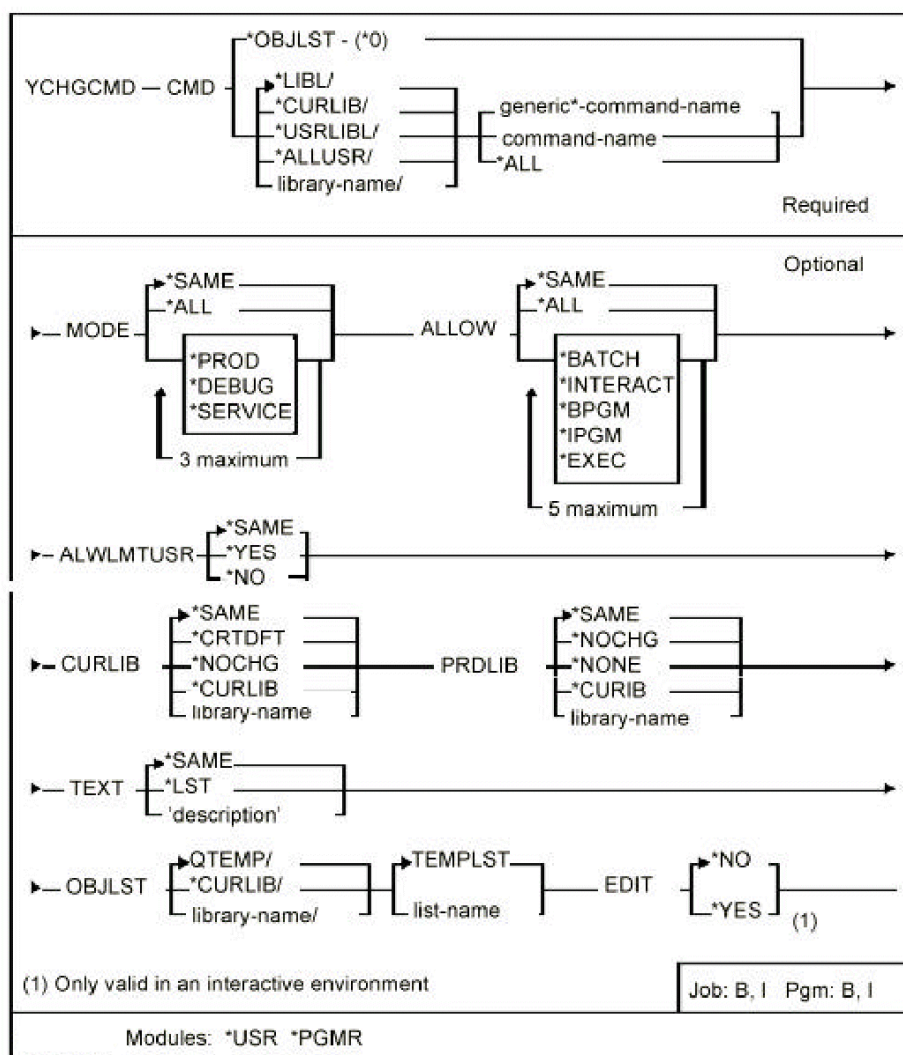
```
YBLDOBJLST OBJ(TVTIMES/BBC*) OBJTYPE(*PGM) OUTFLAGVAL(B)
```

The list will be called TEMPLST and reside in library QTEMP.

YCHGCMD (Change Command)

This command changes the attributes of one or more commands. The commands to be modified can either be specified by a generic name, or with the object list.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
CMD	Qualified generic name of commands with changed attributes	<ul style="list-style-type: none"> ■ *OBJLST: Commands are specified by a list whose name is given by the OBJLST parameter ■ *ALL: All commands in specified library

Parameter	Definition	Value and Description
MODE	Specifies the types of operating environment in which the command is used. One or more of the modes can be specified	<ul style="list-style-type: none">■ *SAME: (default) No change to the modes of operation for the commands■ *ALL: The commands are valid in all types of operation■ *PROD: The commands are valid for production mode■ *DEBUG: The commands are valid for debug mode■ *SERVICE: The commands are valid for service mode

Parameter	Definition	Value and Description
ALLOW	Specifies where the commands are processed. One or more of the options can be specified	<ul style="list-style-type: none">■ *SAME: (default) No change to where the commands are processed■ *ALL: The commands are valid in a batch input stream, in a CL program, or when processed interactively. The commands can also be passed to the system program QCMDEXC (or QCAEXEC) for processing■ *BATCH: The commands are valid in a batch input stream, external to a compiled CL program■ *BPGM: The commands can be included in a compiled CL program that runs in a batch input stream■ *IPGM: The commands can be included in a compiled CL program that runs interactively■ *EXEC: The commands can be included as a parameter on the CALL command and be passed as a character string to the system program QCMDEXC (or QCAEXEC) for processing■ *INTERACT: The commands are valid interactively

Parameter	Definition	Value and Description
ALWLMTUSR	Specifies whether the commands can be entered directly by a user whose profile is set for limited capabilities (as defined by the LMTCPB keyword on the i OS and/or Create User Profile (CRTUSRPRF, YCRTUSRPRF), and Change User Profile (CHGUSRPRF, YCHGUSRPRF) commands	<ul style="list-style-type: none"> ■ *SAME: (default) The limited user authority is not changed ■ *NO: The commands cannot be entered by a user whose profile is set for limited capabilities ■ *YES: The commands can be entered by a user whose profile is set for limited capabilities
CURLIB	Specifies the name of the library used as the current library during the processing of the commands	<ul style="list-style-type: none"> ■ *SAME: (default) The current library for the commands is not changed ■ *CRTDFT: There is no current library active during the processing of these commands ■ *NOCHG: The current library is not changed for the processing of the commands ■ *CURLIB: The current library of the commands is set to invoking the YCHGCMD command. If no current library exists in the library list of the job, QGPL is used

Parameter	Definition	Value and Description
PRDLIB	Specifies the name of the product library that is affected during the processing of the commands	<ul style="list-style-type: none"> ■ *SAME: (default) The product library for the commands is not changed ■ *NONE: There is no product library in the job's library list during processing of the commands ■ *NOCHG: The product library is not changed when the command process starts ■ *CURLIB: The product library of the commands is set to invoking the YCHGCMD command. If no current library exists in the library list of the job, QGPL is used
TEXT	Specifies the user-defined text on the commands	<ul style="list-style-type: none"> ■ *SAME: (default) The user-defined text is unchanged ■ *LST: The user-defined text is taken from the object list entry for the command being changed
OBJLST	Qualified name of object list specifying the commands for which the attributes are changed	<ul style="list-style-type: none"> ■ QTEMP/TEMPLST: (default) List name ■ If CMD(*OBJLST) is specified, an existing list is used. Otherwise a new list is created from the parameters specified
EDIT	Edit list option	<ul style="list-style-type: none"> ■ *NO: (default) The edit list function is not to be invoked ■ *YES: Invoke the edit list facility before proceeding with the command execution

Notes

- 1. The command does not work with commands in libraries QSYS or QSYS38.
- 2. Diagnostic messages are sent for commands whose attributes could not be changed, for instance, because they could not be allocated.

Example

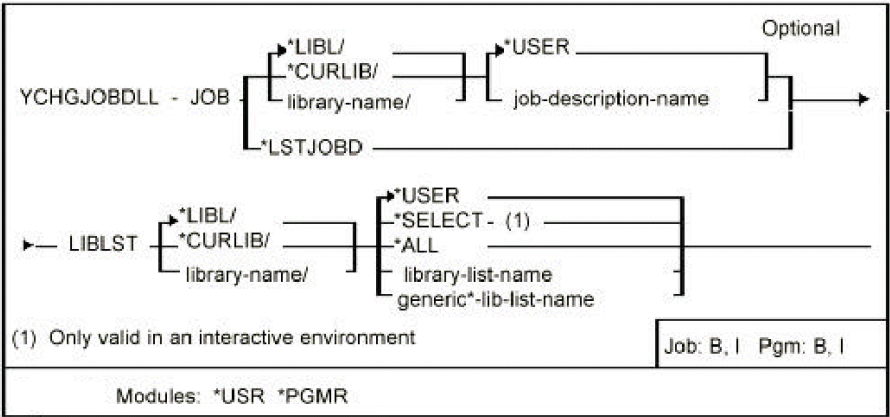
The following command change the product library attribute of all commands in library FRED to library LIVELIB:

```
YCHGCMD CMD(FRED/*ALL) PRDLIB(LIVELIB)
```

YCHGJOB DLL (Change Job Description Library List)

This command replaces the initial library list of a job description with the contents of a named library list. The library list must have been stored using the command Build Library List (YBLDLIBLST).

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
JOB	Qualified name of job description whose initial library list is changed	<ul style="list-style-type: none">■ *USER: (default) Job description has same name as current user■ *LSTJOB: Job description from library list entry
LIB	Qualified name of the library list whose contents are used to replace the initial library list on the specified job description	<ul style="list-style-type: none">■ *USER: (default) Library list has same name as user■ *SELECT: Display a list of available library lists■ *ALL: Select all library lists

Notes

1. The named library list(s) must be created using the commands Build Library List (YBLDLIBLST), Edit Library List (YEDTLIBLST) or Work with Library List (YWRKLIBLST).
2. If LIB(*ALL,generic*) is specified, the named job description must be JOB(*LSTJOB).

Examples

To change the initial library list on job description QBATCH in library QGPL to the contents of library list FRED:

```
YCHGJOB DLL JOB(QGPL/QBATCH) LIB(FRED)
```

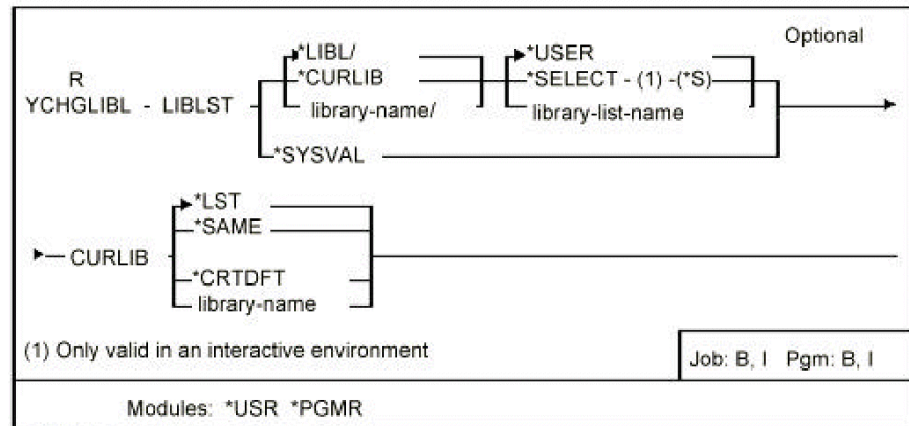
To synchronize the initial library lists of all job descriptions referenced in all library lists with the library lists:

```
YCHGJOB DLL JOB(*LSTJOB) LIB(*ALL)
```

YCHGLIBL (Change Library List)

This command replaces the current job's library list with the contents of a specified library list. The library list must have been stored using the command Build Library List (YBLDLIBLST).

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
LIBLST	Qualified name of the library list whose contents are used to replace the current job's library list	<ul style="list-style-type: none"> ■ *USER: (default) Use library list of same name as user ■ *SELECT: Display a list of available library lists ■ *SYSVAL: Use system user library list
CURLIB	Specifies the library to replace the current library in the current job's library list	<ul style="list-style-type: none"> ■ *LST: (default) Use the library list's current library entry to replace the current library in the current job's library list ■ *SAME: Do not change the current library in the job's library list ■ *CRTDFT: No library should be in the current entry of the job's library list. If objects are created into the current library, then use library QGPL as the default current library

Notes

Library lists are stored in a file called YLIBLST in the specified library.

Examples

To set up a library list FRED for use containing the libraries QTEMP, QGPL, FRED and FREDSRC:

```
YBLDLIBLST LIBLST(FRED) LIBL(QTEMP QGPL FRED FREDSRC)
```

To replace the current job's library list with the contents of FRED:

```
YCHGLIBL LIBLST(FRED)
```

Or simply:

```
R FRED
```

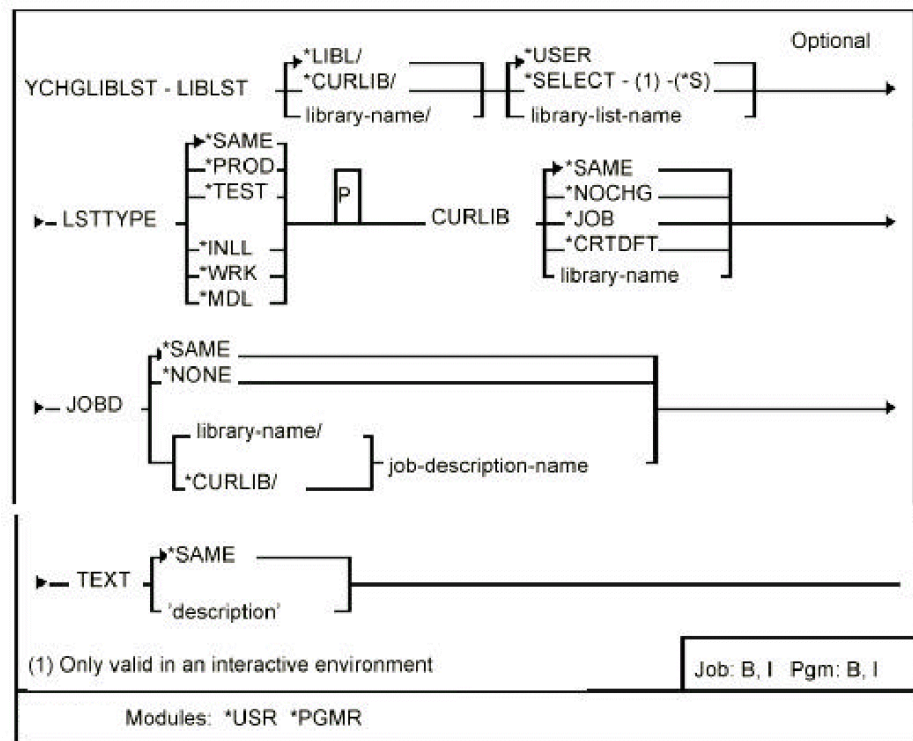
To replace the current job's library list with the contents of FRED, and change the current library to MYLIB:

```
YCHGLIBL LIBLST(FRED) CURLIB(MYLIB)
```

YCHGLIBLST (Change Library List Description)

This command changes a library list description.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
LIBLST	Qualified name of the library list whose contents are used to replace the current job's library list	<ul style="list-style-type: none"> ■ *USER: (default) Use library list with same name as user ■ *SELECT: Display list of available library lists
LSTTYPE	Library list type of the built library list	<ul style="list-style-type: none"> ■ *SAME: (default) Do not change the library list type ■ *PROD ■ *TEST ■ *INLL ■ *WRK ■ *MDL <p>Use chosen library list type</p>

Parameter	Definition	Value and Description
JOB	Qualified name of job description to be associated with the library list	<ul style="list-style-type: none">■ *SAME: (default) Do not change the job description■ *NONE: Do not associate any job description with the library list
CURLIB	Name of the current library to be associated with the library list	<ul style="list-style-type: none">■ *SAME: (default) Do not change the current library■ *NOCHG: Do not change the current library when replacing the library list with the built library list■ *JOB: Use the current library in the job's library list as the current library associated with the library list■ *CRTDFT: Current library is to be *CRTDFT when replacing the library list with the built library list
TEXT	Descriptive text for list (up to 50 characters)	*SAME: (default) Do not change the text

Notes

None

Example

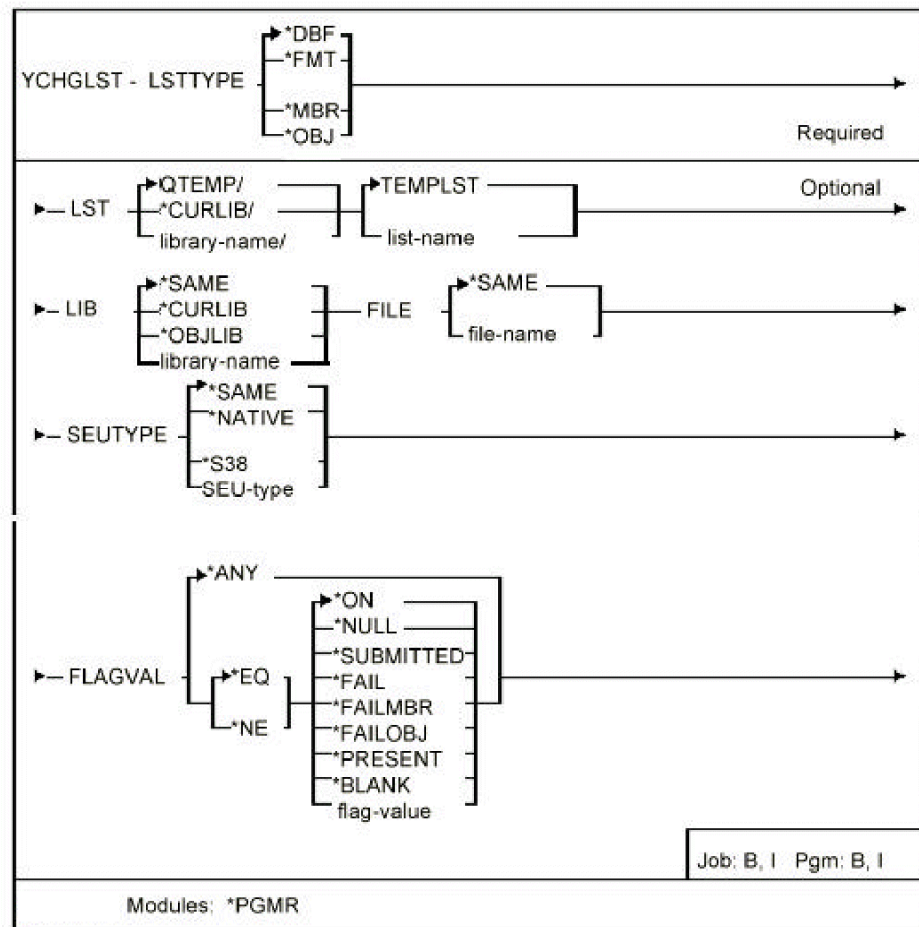
The following command would change the current library of library list FRED to *NOCHG:

```
YCHGLIBLST LIBLST(FRED) CURLIB(*NOCHG)
```

YCHGLST (Change List)

Allows changes to be made to selected fields for each item in a list.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
LSTTYPE	List type whose entries are changed	<ul style="list-style-type: none"> ■ *DBF: Database file list ■ *FMT: Format list ■ *MBR: Member list ■ *OBJ: Object list
LST	Qualified name of a list whose entries are changed	QTEMP/TEMPLST: (default) Name for list

Parameter	Definition	Value and Description
LIB	New value for library name on list items	<ul style="list-style-type: none">■ *SAME: (default) Make no change to library name■ *CURLIB: Change library name to current library of job■ *OBJLIB: Change the library name to the name of the first library in the current library list in which the object is found. If the object is not found, change library to *LIBL
FILE	New value for file name on member list items	<ul style="list-style-type: none">■ *SAME: (default) Make no change to file name Only applies if LSTTYPE(*MBR) or LSTTYPE(*FMT) are specified
SEUTYPE	New value for SEU source type on member list items if LSTTYPE(*MBR), or new value for object attribute on object list if LSTTYPE(*OBJ)	<ul style="list-style-type: none">■ *SAME: (default) make no change to SEU type or object attribute■ *S38: Change all source types (object attributes) to be IBM i S38E types (object attributes), that is add the characters '38' to SEU types (object attributes). For example, convert RPG to RPG38, DSPF to DSPF38■ *NATIVE: Change all source types (object attributes) to be native IBM i source types (object attributes), that is remove any instances of the characters '38' or '36' from SEU types (object attributes). For example, convert RPG38 to RPG, DSPF36 to DSPF' Only applies if LSTTYPE(*MBR) or LSTTYPE(*OBJ) specified

Parameter	Definition	Value and Description
FLAGVAL	Flag value of list entries to change	<ul style="list-style-type: none"> ■ *ANY: (default) Change all list entries Otherwise, FLAGVAL is a list parameter made up of the following two elements: ■ Relational operator for selection of flags <ul style="list-style-type: none"> – *EQ: (default) Equal to – *NE: Not equal to ■ Flag value <ul style="list-style-type: none"> – Single character flag value or one of the special flag values

Notes

List entries selected by the FLAGVAL parameter will be updated with the value or values specified.

List type	List item data updated			
	FILE	LIB	SEUTYPE	OBJATR
DBFLST	-	File library	-	-
FMTLST	File name	File library	-	-
MBRLST	File name	File library	SEU type	-
OBJLST	-	Object library	-	SEU type

Examples

To change the library name on all object entries in object list FRED to HORACE:

```
YCHGLST LSTTYPE(*OBJ) LST(FRED) LIB(HORACE)
```

To change the source type on all member entries in member list FRED to RPG:

```
YCHGLST LSTTYPE(*MBR) LST(FRED) SEUTYPE(RPG)
```

YCHGPWD (Change Password)

This command calls an interactive program that displays password expiry information for user profiles, and allows the password to change.

Syntax Diagram

YCHGPWD There are no parameters for this command	
	Job: I
Modules: *USR	

Parameters

None

Notes

1. Calls an interactive program to display the date the password was last changed, expiry date information (if specified on the user profile) and to allow the password to be changed. Press HELP while using the program for instructions.
2. This command does not work in debug mode.
3. At i OS Release V1R20 or above.

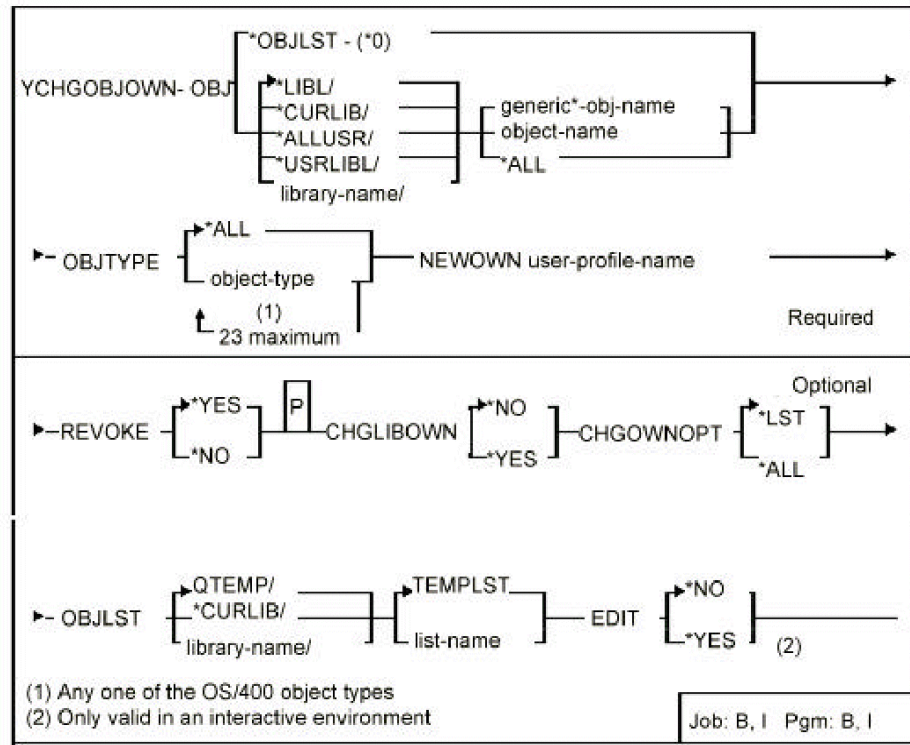
The YCHGPWD command is primarily used to display password expiry information. It calls the initial program if you sign on after the password has expired or when it is due to expire within a 'grace' period.

A function key is provided to invoke the i OS command Change Password (CHGPWD), which can be used to change your password. Password validation is effected via the CHGPWD command.

YCHGOBJOWN (Change Object Ownership)

This command changes the object ownership for a list of objects. The list can either be specified with a generic name, or an object list previously created with the build object list functions. The current owner's authorities can be revoked at the same time.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
OBJ	Qualified generic name of objects whose ownership is changed	<ul style="list-style-type: none"> ■ *OBJLST: Objects are specified by a list whose name is given by the OBJLST parameter ■ *ALL: All objects
OBJTYPE	List of types of object whose ownership is to be changed	*ALL: change ownership for all object types
NEWOWN	New owner and must be the name of an existing user profile	

Parameter	Definition	Value and Description
REVOKE	Revoke old owner's authorities	<ul style="list-style-type: none"> ■ *YES: (default) Authorities of the old owner are revoked, unless the old owner is the same as that specified by the NEWOWN parameter ■ *NO: Old owner is still authorized
CHGLIBOWN	Change ownership of library containing objects	<ul style="list-style-type: none"> ■ *NO: (default) Do not change ownership of library. ■ *YES: change ownership of library. CHGLIBOWN(*YES) can not be specified if OBJ(*OBJLST) is specified
CHGOWNOPT	Specifies if the command should ignore objects owned by the NEWOWN profile. (The list records the ownership of the object when the list is built.)	<ul style="list-style-type: none"> ■ *LST: (default) Changes ownership only of objects the list indicates are not already owned by the NEWOWN profile ■ *ALL: Change ownership of all objects in list regardless of their recorded ownership
OBJLST	Qualified name of object list specifying the object for which the ownership is to be changed.	QTEMP/TEMPLST: (default) List name If OBJ(*OBJLST) is specified, an existing list is used. Otherwise a new list is created from the parameters specified
EDIT	Edit list option.	<ul style="list-style-type: none"> ■ *NO: (default) The edit list function is not invoked ■ *YES: Invoke the edit list facility before proceeding with the command execution

Notes

1. The user must have object existence rights to objects whose ownership is being changed.
2. The command does not generally work on objects in a library whose name begins with the letter 'Q'. The following are exceptions:
 - Objects in libraries QTEMP, QGPL, QDOC and QPATOOL.
 - Library objects QTEMP, QGPL, QRCL, QRECOVERY, QUSRSYS and QRPLOBJ in library QSYS.
 - User profile objects QPGMR, QSYSOPR, QUSER and QDFTOWN in library QSYS.
3. Diagnostic messages are sent for objects whose ownership could not be changed, for instance because they could not be allocated.

Example

The following command change the ownership of all objects in library FRED to a new owner ALOYSIUS:

```
YCHGOBJOWN OBJ(FRED/*ALL) OBJTYPE(*ALL) NEWOWN(ALOYSIUS)
```

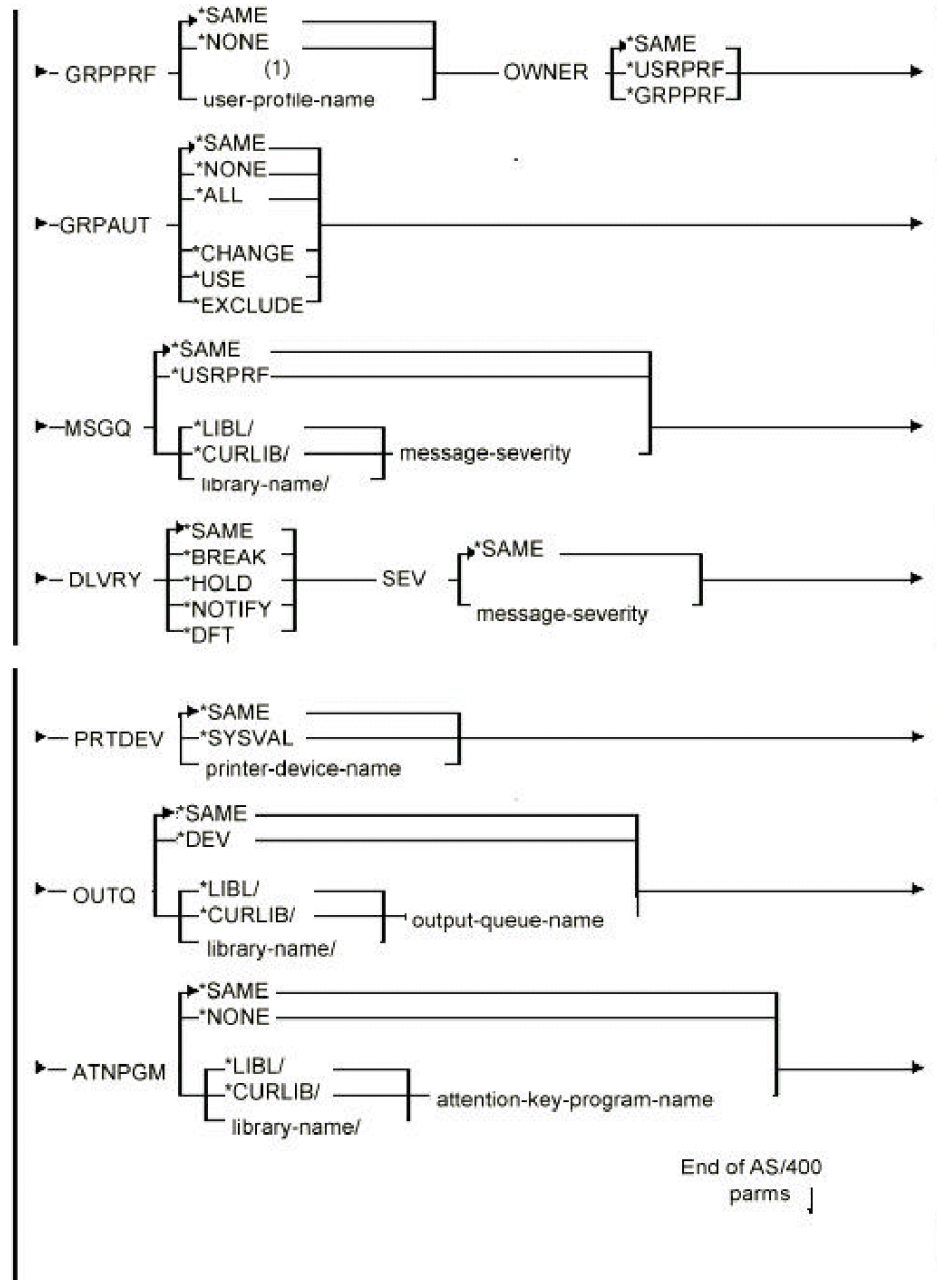
YCHGUSRPRF (Change User Profile)

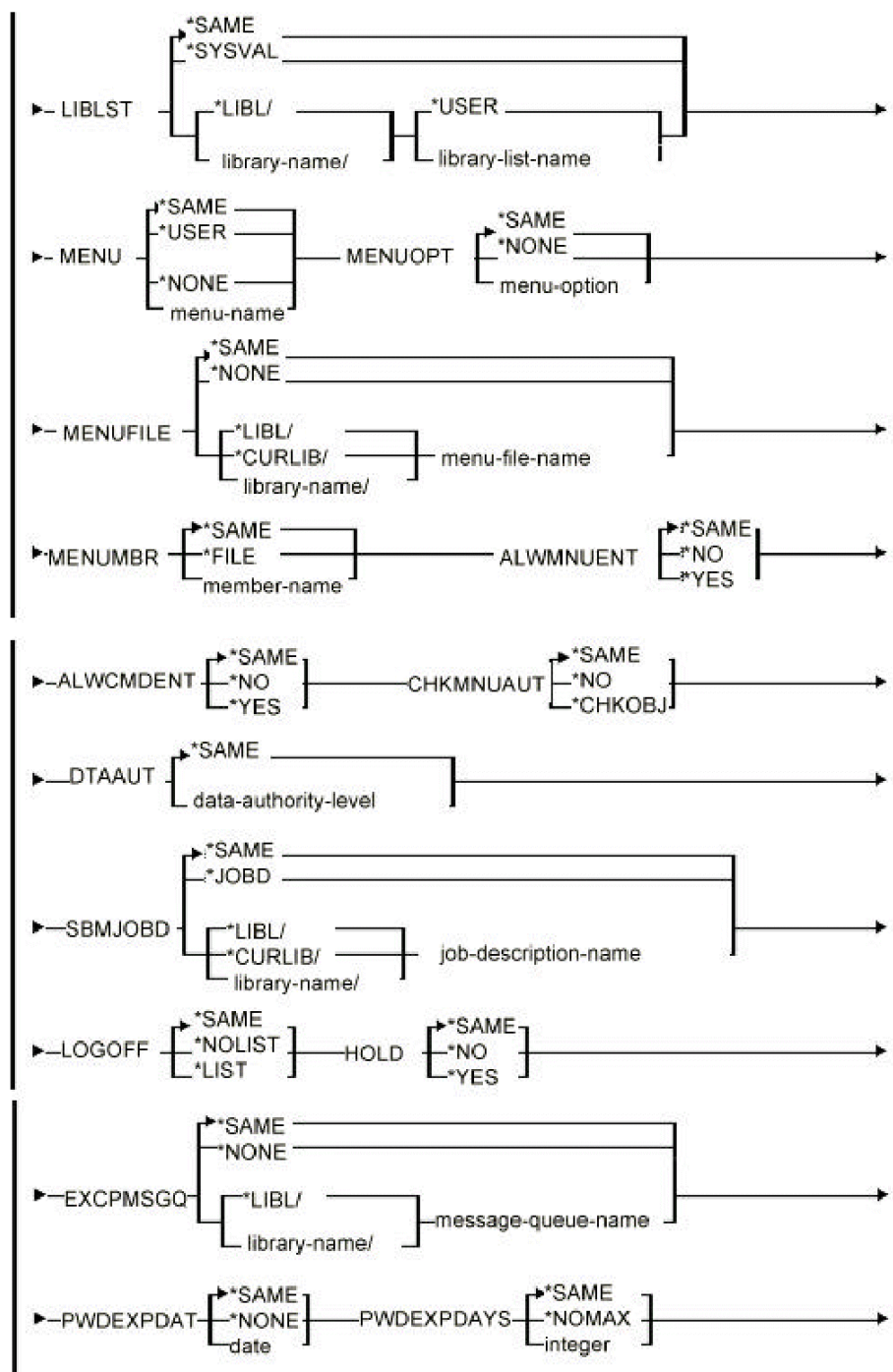
This command changes a specified user profile. Both i OS and CA2E user profile details can be changed.

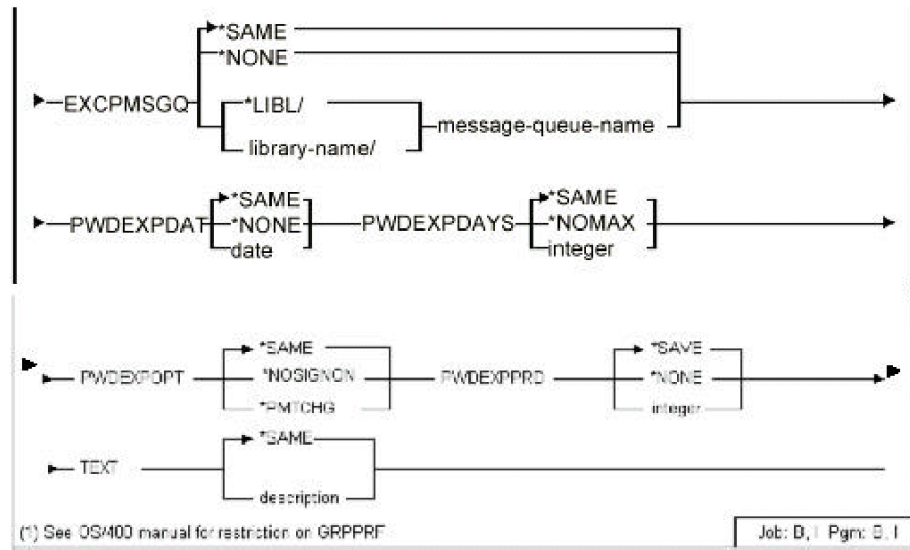
Syntax Diagram

The diagram appears on the following pages.

YCHGUSRPRF - USRPRF user-profile-name		Required
▶- PASSWORD <div> <div>*SAME</div> <div>*NONE</div> <div>password</div> </div>	<div>P</div> <div>USRCLS</div> <div> <div>*SAME</div> <div>*USER</div> <div>*SECOFR</div> <div>*SECADM</div> <div>*PGMR</div> <div>*SYSOPR</div> </div>	Optional
▶- CURLIB <div> <div>*SAME</div> <div>*CRTDFT</div> <div>library-name</div> </div>		
▶- INLPGM <div> <div>*SAME</div> <div>*NONE</div> <div>*LIBL/</div> <div>*CURLIB/</div> <div>library-name/</div> </div>	<div>program-name</div>	
▶- INLMNU <div> <div>*SAME</div> <div>*SIGNOFF</div> <div>*LIBL/</div> <div>*CURLIB/</div> <div>library-name/</div> </div>	<div>menu-name</div> <div>LMTCPB</div> <div> <div>*SAME</div> <div>*NO</div> <div>*PARTIAL</div> <div>*YES</div> </div>	
▶- SPCAUAT <div> <div>*SAME</div> <div>*NONE</div> <div>*USRCLS</div> <div>*SAVSYS</div> <div>*JOBCTL</div> <div>*ALLOBJ</div> <div>*SERVICE</div> <div>*SPLCTL</div> <div>*SECADM</div> <div>6 maximum</div> </div>		
▶- JOB <div> <div>*SAME</div> <div>*LIBL/</div> <div>*CURLIB/</div> <div>library-name/</div> </div>	<div>QDFTJOB</div> <div>job-description-name</div>	







Parameters

Parameter	Definition	Value and Description
OBJ	Qualified generic name of objects whose ownership is to be changed	<ul style="list-style-type: none"> ■ *OBJLST: Objects are specified by a list whose name is given by the OBJLST parameter ■ *ALL: all objects
USRPRF	User profile name	

Parameter	Definition	Value and Description
PASSWORD	A valid User password name	<ul style="list-style-type: none"> ■ SAME: (default) The previous value remains unchanged ■ *NONE: Profile has no password. USRCLS User class. One of the following values: ■ *USER: Defined as a user ■ *SECOFR: User has security officer class ■ *SECADM: User has administration class. ■ *PGMR: Defined as a programmer ■ *SYSOPR: User has system operator class
CURLIB	Default current library for this user	<ul style="list-style-type: none"> ■ *SAME: (default) The previous value remains unchanged ■ *CRTDFT: This user has no default create library. The library QGPL is used as the default library
INLPGM	Qualified name of initial program for profile. This program is the first program to execute when the user signs on	<ul style="list-style-type: none"> ■ *SAME: (default) The previous value remains unchanged ■ *NONE: There is no initial program for the profile ■ *LIBL/YINLPGM: (default) The initial program is the first program to execute
INLMNU	Qualified name of initial menu for profile. This menu is called if there is no initial program, and if an initial program executes a return	<ul style="list-style-type: none"> ■ *SAME: (default) The previous value remains unchanged ■ *SIGNOFF: Executes a signoff command if the initial program executes a return

Parameter	Definition	Value and Description
LMTCPB	Limit capability of user to change user profile	<ul style="list-style-type: none"> ■ *SAME: (default) The previous value remains unchanged ■ *NO: Do not limit capability ■ *YES: Limit capability ■ *PARTIAL: Partial limitation
SPCAUT	Special authorities. Either single value	<ul style="list-style-type: none"> ■ *SAME: (default) The previous value remains unchanged ■ *NONE: No special authorities are to be granted ■ *USRCLS: Authorities are defined by user class ■ *Or a list of up six special values: <ul style="list-style-type: none"> – *SAVSYS: Grant save system rights – *JOBCTL: Grant job control rights – *SECADM: Grant administrator rights – *ALLOBJ: Grant rights to all objects – *SERVICE: Grant service rights – *SPLCTL: Grant spool control rights
JOB	Qualified name of initial job description defining the initial job environment when the user signs on	<ul style="list-style-type: none"> ■ *SAME: (default) The previous value remains unchanged ■ QDFTJOB: The system-supplied job description in library QGPL is used
GRPPRF	Group profile name	<ul style="list-style-type: none"> ■ *SAME: (default) The previous value remains unchanged ■ *NONE: Profile is not a group profile member

Parameter	Definition	Value and Description
Owner	Owner of created objects for group members. If a value of GRPPRF(*NONE) is specified then *USRPRF must be specified	<ul style="list-style-type: none">■ *SAME: (default) The previous value remains unchanged■ *USRPRF: The user profile is to own any objects created by the profile■ *GRPPRF: The group profile is to own any created objects
GRPAUT	Authority given to the group profile for newly created objects. If OWNER (*GRPPRF) is specified *NONE is required	<ul style="list-style-type: none">■ *SAME: (default) The previous value remains unchanged■ *NONE: No authority is given to the group profile■ *ALL: All authority is given to the group profile■ *CHANGE: Change authority for the object type is given to the group profile■ *USE: Authority to use the object type is given to the group profile■ *EXCLUDE: User profile is excluded
MSGQ	Qualified name of message queue associated with the user profile	<ul style="list-style-type: none">■ *SAME: (default) The previous value remains unchanged■ *USRPRF: A message queue with the same specified name as in the USRPRF parameter and used as the message queue for the user

Parameter	Definition	Value and Description
DELVRV	Mode of delivery for messages sent to specified message queue	<ul style="list-style-type: none">■ *SAME: (default) The previous value remains unchanged■ *HOLD: Messages are to be held on the message queue■ *BREAK: Messages of the appropriate severity are displayed at the time of delivery. The severity level is determined by the value specified for SEV■ *NOTIFY: Messages of the appropriate severity cause the message waiting indicator to be displayed■ *DFT: All information messages are ignored and other messages are held. Any enquiry messages are given the default reply specified on the job description, or on the message description
SEV	Message severity to use when setting message queue delivery	<ul style="list-style-type: none">■ *SAME: (default) The previous value remains unchanged■ 00/99: Message severity. See i OS manual for message severity code meanings
PRTDEV	Name of the printer associated with the user	<ul style="list-style-type: none">■ *SAME: (default) The previous value remains unchanged■ *SYSVAL: Use the printer device specified by the system value QPRTDEV

Parameter	Definition	Value and Description
OUTQ	Qualified name of output queue associated with the user profile	<ul style="list-style-type: none"> ■ *SAME: (default) The previous value remains unchanged ■ *DEV: Output is to be directed to the output queue having the same name as the printer device specified on the PRTDEV parameter
ATNPGM	Qualified name of the attention key program associated with the user profile	<ul style="list-style-type: none"> ■ *SAME: (default) The previous value remains unchanged ■ *NONE: No attention key program is associated with the user profile <p>If an attention program is specified, the job automatically changes into a group job.</p>
AUT	Authority given to the public for the user profile	<ul style="list-style-type: none"> ■ *SAME: (default) The previous value remains unchanged ■ *ALL: All authorities are given to the public ■ *CHANGE: Change authority for the profile given to the public ■ *USE: Authority to display the profile is given to the public ■ *EXCLUDE: No authority is given to the public
LIBLST	Qualified name of initial library list set at start of sign-on	<ul style="list-style-type: none"> ■ *SAME: (default) The previous value remains unchanged ■ *SYSVAL: Uses the system default library list ■ *USER: Uses a library list with the same name as that specified in the USRPRF parameter

Parameter	Definition	Value and Description
MENU	Name of initial profile user menu	<ul style="list-style-type: none"> ■ *SAME: (default) The previous value remains unchanged ■ *USER: Uses an initial menu with the same name as that specified in the USRPRF parameter ■ *NONE: There is no initial menu
MENUOPT	The profile for the initial menu option	<ul style="list-style-type: none"> ■ *SAME: (default) The previous value remains unchanged ■ *NONE: The initial menu is displayed ■ menu-option: Executes the program/command specified by the initial menu option
MENUFILE	Qualified file name and must be a menu file	<ul style="list-style-type: none"> ■ *SAME: (default) The previous value remains unchanged ■ *LIBL/YDSNMNU: The default menu file name ■ *NONE: No initial menu is displayed
MENUMBR	Menu file member name	<ul style="list-style-type: none"> ■ *SAME: (default) The previous value remains unchanged ■ *FILE: The member name is the name specified in the FILE parameter
ALWMNUENT	Allow menu name entry from the menu display	<ul style="list-style-type: none"> ■ *SAME: (default) The previous value remains unchanged ■ *NO: The user is not permitted to enter menu names. The user is restricted to the displayed menu options ■ *YES: The user can enter menu names to transfer directly to the named menu

Parameter	Definition	Value and Description
ALWCMDENT	Allow command entry from the menu display	<ul style="list-style-type: none">■ *SAME: (default) The previous value remains unchanged■ *NO: The user is not permitted to enter commands and is restricted to the displayed menu options.■ *YES: The user may enter and execute commands directly from the menu
CHKMNUAUT	Checks user authorization to menu options while loading menus	<ul style="list-style-type: none">■ *SAME: (default) The previous value remains unchanged■ *NO: When loading a menu, do not check the user's authorization to use the program or command named for each menu option■ *CHKOBJ: When loading a menu, check the user's authorization to use the program or command named for each executable menu option
DTAAUT	Data authority level (1-high to 9-low)	<p>This value is available directly from the file YUSRPRF in the library, or via the YRTVUSRPRF command in a CL program. It can be used to provide a simple form of field level authorization, For example, levels 4 and above can view a salary field. Checking must be provided by user code</p> <p>*SAME: (default) The previous value remains unchanged</p>

Parameter	Definition	Value and Description
SBMJOB	Qualified job description for submitted jobs	<p>This job description is used by the menu display program for menu options having the submit option</p> <ul style="list-style-type: none"> ■ *SAME: (default) The previous value remains unchanged ■ *JOB: Use the value specified on the JOB parameter
LOGOFF	Sign off option	<ul style="list-style-type: none"> ■ *SAME: (default) The previous value remains unchanged ■ *NOLIST: No job log is to be created ■ *LIST: A job log is to be created
HOLD	Hold/release user profile option	<ul style="list-style-type: none"> ■ *SAME: (default) The previous value remains unchanged ■ *NO: The user may sign-on ■ *YES: The user is prevented from signing on
EXCPMSGQ	Qualified name of exception message queue. Profile associated with the user profile and a copy of any exception message received by the display menu program is sent to this queue	<ul style="list-style-type: none"> ■ *SAME: (default) The previous value remains unchanged ■ *NONE: No exception message is associated with the profile
PWDEXPDAT	Date password expires	<ul style="list-style-type: none"> ■ *SAME: (default) The previous value remains unchanged ■ *NONE: There is no expiry date

Parameter	Definition	Value and Description
PWDEXPDAYS	Number of days after date of last change that a password is to remain valid	<ul style="list-style-type: none"> ■ *SAME: (default) The previous value remains unchanged ■ *NOMAX: There is no limit to the number of days that the password remains valid 1-999: Number of days
PWDEXPOPT	Action that YINLPGM is to take if password expiry is detected for a user at sign-on	<ul style="list-style-type: none"> ■ *SAME: (default) The previous value remains unchanged ■ *NOSIGNON: Prevent user from signing on ■ *PMTCHG: Prompt user for new password, providing number of days specified by PWDEXPPRD parameter has not been exceeded
PWDEXPPRD	Number of days after password has expired that user is still allowed to sign on, providing the password is changed	<ul style="list-style-type: none"> ■ *SAME: (default) The previous value remains unchanged ■ *NONE: The user may not sign on after the password has expired. 1-999: Number of days
TEXT	Text associated with user profile	<ul style="list-style-type: none"> ■ *SAME: (default) The previous value remains unchanged

Notes

1. The YCHGUSRPRF command can also be conveniently executed via the command Display User Profile (YDSPUSRPRF), as the existing parameter values are prompted.
2. Initial YINLPGM program; YCHGUSRPRF command YINLPGMPWD Initial programs. For a profile to be able make use of the user profile extension attributes at sign-on, the initial program (YINLPGM), or a user modified variant of it, must be specified as the initial program on the INLPGM parameter. Password expiry checking is done by a subsidiary program YINLPGMPWD.

Example

To change a user profile called ATTILLA to have the initial menu option 1 in menu ATTMNU:

```
YCHGUSRPRF USRPRF(ATTILLA) MENU(ATTMNU) MENUOPT(1)
```

YCHKIFSOBJ (Check IFS Object)

The Check IFS Object (YCHKIFSOBJ) command checks the existence of an object in the Integrated File System (IFS) and verifies the user's authority to the object before trying to access it.

If the object exists and the user has the correct authority to the object, then no error messages will be sent to the user. If the object does not exist or the user does not have the correct authority to it, an error message will be sent to the user. The error message will be in the range CPE3401 through CPE3599.

Parameters

Parameter	Definition	Value and Description
OBJ	Specify the full path name of the object in the Integrated File System to be checked. An asterisk signifying a wildcard character is not allowed within either a directory or an object name. This is a required parameter	Valid examples are as follows: /webopt/adv2e.gif /QDLS/Y2WEBDOC/H0001043.HTM /qsys.LIB/y2syveng.LIB/yvalls trfp.FILE/yvallstrfp.MBR /QOpenSys/QIBM/UserData/HTTSPSVR/ProxyCache
ACCESS	Specifies the type of checking to be performed on the object specified in the OBJ parameter.	<ul style="list-style-type: none"> ■ *EXIST: Only the object's existence is checked. ■ *R: The command checks whether the user running this command has read authority to the object. A user with *R authority can view the contents of the object. *R authority provides object operational authority and

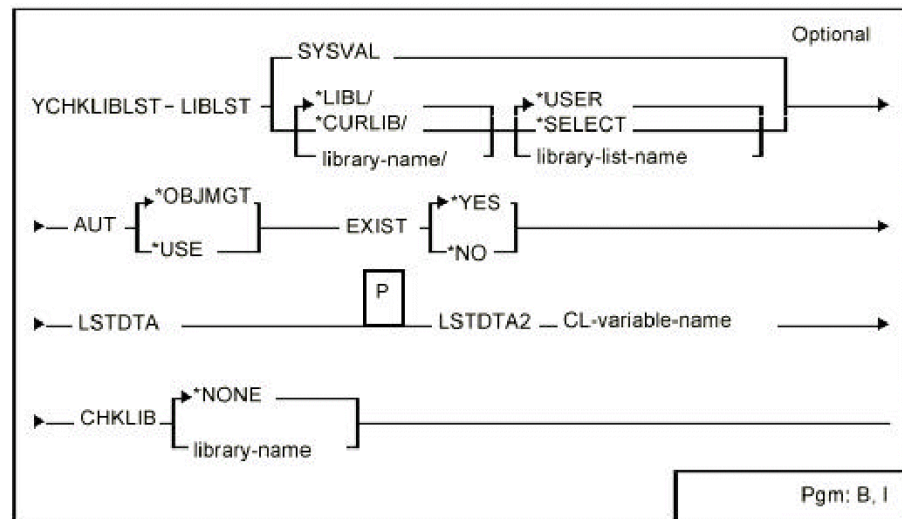
Parameter	Definition	Value and Description
		data read authority.
		■ *W: The command checks whether the user running this command has write authority to the object. A user with *W authority can change the contents of the object. *W authority provides object operational authority and data add, update, and delete authorities.
		■ *X: The command checks whether the user running this command has execute authority to the object. A user with *X authority can run a program or search a library or directory. *X authority provides object operational authority and data execute authority.
		■ *RW: The command checks whether the user running this command has read and write authority to the object. A user with *RW authority can view the contents of the object and change the contents of the object. *RW authority provides object operational authority and data read, add, update, and delete authorities.
		■ *RX: The command checks whether the user running this command has read and execute authority to the object. A user with *RX authority can perform basic operations on the object, such as run a program or display the contents of a

Parameter	Definition	Value and Description
		file. The user is prevented from changing the object. *RX authority provides object operational authority and read and execute authorities.
		■ *WX: The command checks whether the user running this command has write and execute authority to the object. A user with *WX authority can change the contents of the object and run a program or search a library or directory. *WX authority provides object operational authority and data add, update, delete, and execute authorities.
		■ *RWX: The command checks whether the user running this command has read, write and execute authority to the object. A user with *RWX authority can perform all operations on the object except those limited to the owner or controlled by object existence, object management, object alter, and object reference authority. The user can change the object and perform basic functions on the object. *RWX authority provides object operational authority and all the data authorities.

YCHKLIBLST (Check Library List)

Checks for a library list, and is intended for internal use by utilities. Returns the list contents to a variable.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
LIBLST	Qualified name of the checked library list	<ul style="list-style-type: none"> ■ *USER: (default) Checks the list with the name of the current user profile to ensure the name exists in the library ■ *SELECT: Provides a display of existing lists in the specified library ■ *SYSVAL: Returns the list contents of the system value library list QSYSLIBL

Parameter	Definition	Value and Description
AUT	Authorization rights to check	<ul style="list-style-type: none"> ■ *OBJMGT: (default) Corresponds to authority rights *OBJOPR, *ADD, *READ, *DLT, *UPD for the list file ■ *USE: Corresponds to authority rights *OBJOPR and *READ for the library list file
EXIST	Checks whether specified list exists	<ul style="list-style-type: none"> ■ *YES: (Default) Issues escape message if list specified by the LIBLST parameter is not found ■ *NO: Issues escape message if list is found
LSTDTA	Name of CL-variable into which the first part of the found list is to be returned. Must be a character variable, 322 bytes long	
LSTDTA2	Name of the CL variable into which the second part of the found list is to be returned. Must be a character variable 87 bytes long	
CHKLIB	Checks whether a library exists in the library list	*NONE: (default) do not check any library's existence in library list

Notes

1. If errors occur on the checks, i OS or escape messages are sent.
File, member, and authority checking is performed using the i OS command Check Object (CHKOBJ). See the i OS *Control Language Reference* manual for details of the escape messages sent by the CHKOBJ command.
The following additional messages can be sent:
 - *YLL0001 - Specified library list not found.
 - *YLL0003 - Specified library file list not found.

- *YLL0006 - Specified library does not exist in library list.
 - *YLL0007 - No selection made; command cancelled by user.
 - *YLL0014 - Specified library list already exists.
2. Library lists are stored in a file called YLIBLST in the specified library.
 3. The LSTDTA and LSTDTA2 parameters have the following formats:

Field	Type		Offset		Contents
LSTDTA					
YLLVN	10	A	01	- 10	Library list name
YLLTX	50	A	11	- 60	Library list text
YLJBUS	10		61	70	Library list creator
YLLLEL	2.0	B	71	72	No of libraries in list
YLLBLS	250	A	73	- 322	List contents: up to 25 x 10
LSTDTA2					
YLCLVN					
YLJDVN	10	A	1	- 10	Library list current library
YLDLB	10	A	11	- 20	Library list job description
YLLTP	10	A	21	- 30	Library list jobd library
	4	A	31	- 34	Library list type
	53	A	35	- 87	(reserved)

Example

To check whether the list FRED exists in file *LIBL/YLIBLST, and check if the correct rights are obtained (object management rights):

```
DCL VAR(&LIBLST) TYPE(*CHAR) LEN(10) /* List name */
DCL VAR(&LSTDTA) TYPE(CHAR) LEN(322) /* List contents */
DCL VAR(&LIB) TYPE(*CHAR) LEN(10) /* Library name */
CHGVAR VAR(&LIBLST) VALUE('FRED')
YCHKLIBLST LIBLST(&LIBLST) LSTDTA(&LSTDTA)
```

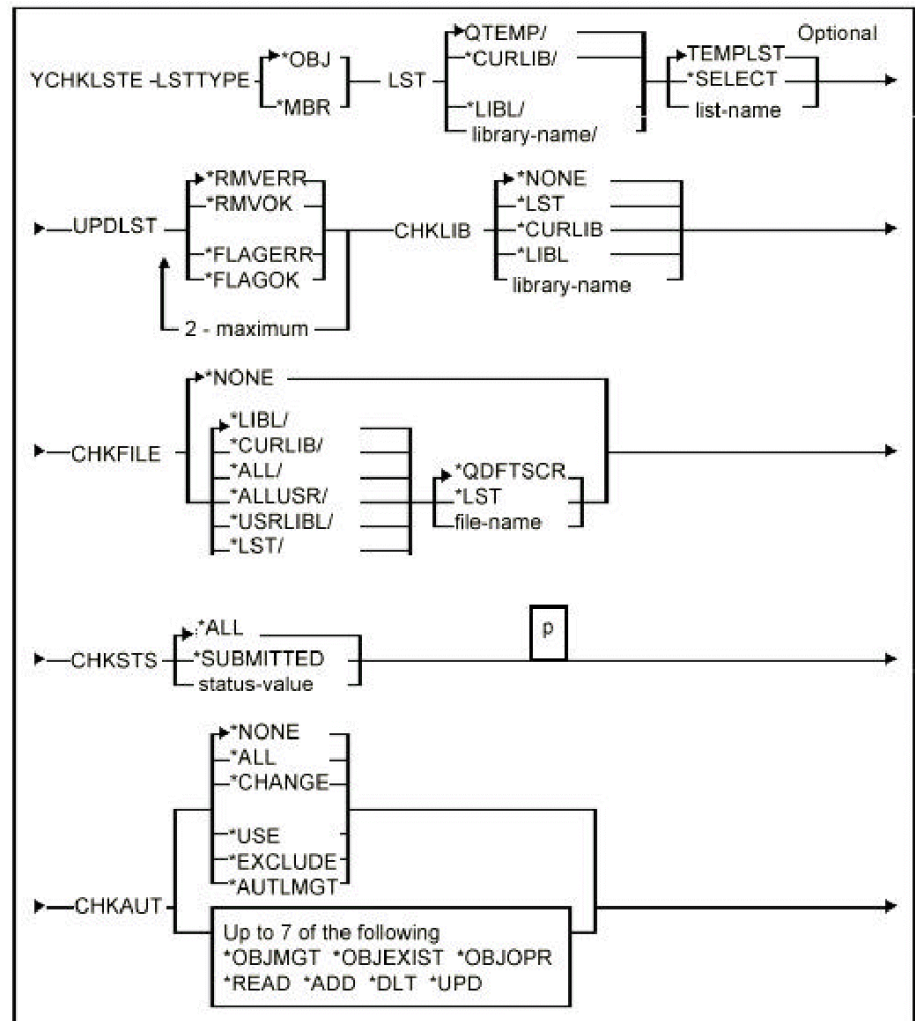
If list FRED is successful then you might obtain the name of the first library in the list as follows:

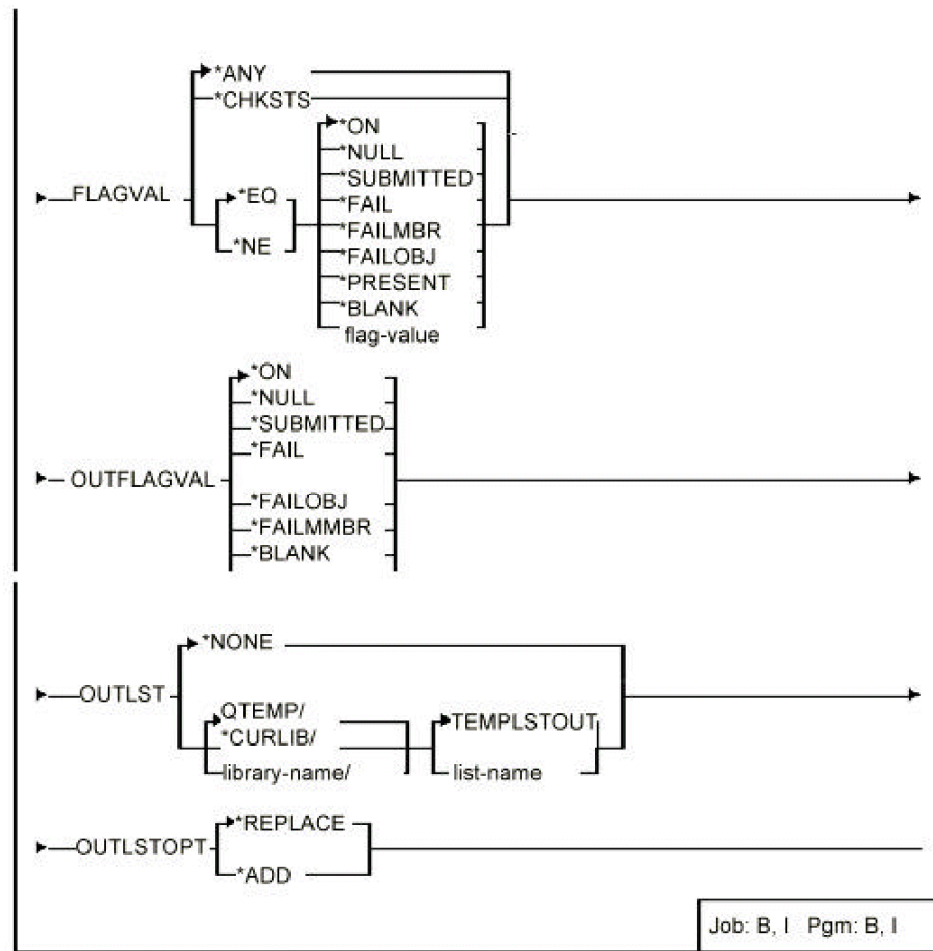
```
CHGVAR VAR(&LIB) VALUE(%SST(&LSTDTA 73 10))
```

YCHKLSTE (Check List Entries)

This command checks the entries in an object or member list.

Syntax Diagram





Parameters

Parameter	Definition	Value and Description
LSTTYPE	Type of list whose entries are checked	<ul style="list-style-type: none"> *OBJ: (default) Object list *MBR: Member list
LST	Qualified name of E list whose entries are checked	<ul style="list-style-type: none"> QTEMP/TEMPLST: (default) List name *SELECT: Display list of existing lists in specified library

Parameter	Definition	Value and Description
UPDLST	List update option	<p>Up to two of the following values:</p> <ul style="list-style-type: none"> ■ *RMVERR: (default) Removes items from list that are not verified ■ *RMVOK: Removes items from list if they are verified. For example, if both the specified source member and object are found ■ *FLAGERR: Flags items from list that are not verified. Flag with the flag specified by the OUTFLAGVAL parameter ■ *FLAGOK: Flag the items in the list if they are verified. Flag with the flag specified by the OUTFLAGVAL parameter <p>If more than one value is specified, *RMVERR + FLAGOK and *RMVOK + FLAGERR are the only valid combinations</p>
CHKLIB	Name of the library to use when checking the existence of objects	<ul style="list-style-type: none"> ■ *NONE: (default) Do not check object's existence ■ *LST: Use the list value. For object lists, the library specified for each object list entry is used; for member lists, an object with the same name as the member is looked for. The source type is determined by the object type. The check is ignored for source types that do not have corresponding objects, e.g. text ■ *LIBL: Use the current job's library list to locate objects when checking their existence ■ *CURLIB: Use the current library for the invoking job when checking for an object's existence

Parameter	Definition	Value and Description
CHKFILE	Qualified name of source file to use when checking for the existence of source	<ul style="list-style-type: none"> ■ *NONE: (default) do not check for the existence of a corresponding source member ■ *LST: use the source file name and member name specified for the list entry. For object lists, this will have been derived from the name held on the compiled object ■ *QDFTSRC: use the default source file name according to the source type, e.g. QRPGRSRC for RPG III source, QCLSRC for CL source. Use the object name as the source member name. (Note that YCHKLSTE assumes that Query and DFU execution objects do not have source, unless a source member name is present for the list entry.)
CHKSTS	Status of list entry values to check: entries must have the specified value to when checking for the existence of source	<ul style="list-style-type: none"> ■ *ALL: (default) do not check that list entries have a particular value ■ *SUBMITTED: Check that list entries have a value of 'S'
CHKAUT	Authority to check for object or member existence as specified by the CHKLIB or CHKFILE parameters	<p>*NONE: (default) Do not check for any particular authorities when checking existence of list items</p> <p>i OS object authorities to test. See the <i>IBM i Programmers guide</i> for information on these values</p>

Parameter	Definition	Value and Description
FLAGVAL	Flag value of list entries to check	<ul style="list-style-type: none"> ■ *ANY: (default) Check all list entries ■ *CHKSTS: Use value specified by CHKSTS parameter <p>Otherwise, FLAGVAL is a list parameter made up of the following two elements:</p> <p>Relational operator for selection of flags</p> <ul style="list-style-type: none"> ■ *EQ: (default) equal to ■ *NE: not equal to <p>Flag value</p> <ul style="list-style-type: none"> ■ Single character flag value or one of the special flag values
OUTFLAGVAL	Single character flag value or one of the special flag values	Flags value of entries when the UPDLST parameter is *FLAGERR or *FLAGOK
OUTLST	Qualified name of the list resulting entries are placed. The output list is the same type as the input list	<ul style="list-style-type: none"> ■ *NONE: (default) List entries are not copied to an output list ■ QTEMP/TEMPLSTOUT: Default list name
OUTLSTOPT	Output list replacement option for the list specified by the OUTLST parameter	<ul style="list-style-type: none"> ■ *REPLACE: (default) Creates a new output list, replacing any previous list's contents ■ *ADD: Adds to any existing list's contents

Notes

1. The YCHKLSTE command sends a completion message (YOL0026) that indicates the number of items in the original list, and the number of items removed or copied. If there are no items in the specified list an escape message (YYY0103) is sent. Thus the command can be used to check whether this list is empty.
2. For an entry to be verified as correct the conditions specified by both the CHKFILE and the CHKLIB parameters must be satisfied.
3. If no Output list is specified, the list specified by the LST parameter is updated according to the following table:

UPDLST	PASS	FAIL
*RMVERR	Leave	Remove
*RMVOK	Remove	Leave
*FLAGERR	No Chg	Flag
*FLAGOK	Flag	No Chg

If an Output list is specified, the list specified by the OUTLST parameter is updated according to the following table:

UPDLST	Input List		Output List	
	PASS	FAIL	PASS	FAIL
*RMVERR	Leave	Leave	Add	-
*RMVOK	Leave	Leave	-	Add
*FLAGERR	No Chg	No Chg	No Chg	Flag
*FLAGOK	No Chg	No Chg	Flag	No Chg

Examples

A member list example is as follows:

There is a member list containing five entries, and the corresponding source member and objects exist as follows:

```

List entry      : Source member      : Object
QRPGSRC FRED DICK RPG : FRED/QRPGSRC DICK RPG : QGPL/DICK *PGM
QDDSSRC QGPL CINDY PF : QGPL/QDDSSRC CINDY PF : QGPL/CINDY *PF
QDDSSRC QGPL JOE LF : QGPL/QDDSSRC JOE LF : none
QCLSRC QGPL JAKE CLP : none : QGPL/JAKE *PGM

```

The following command verifies that all the members named in list TEMPLST still exist in the files from which the list is build, and that for each member the corresponding compiled object exists in library QGPL:

```
YCHKLSTE LSTTYPE(*MBR) CHKLIB(QGPL) UPDLST(*RMVERR)
```

```

List entry
QRPGSRC FRED DICK RPG
QDDSSRC QGPL CINDY PF

```

Diagnostic messages are sent to indicate which entries are removed.

An object list example is as follows:

Verify that all objects named in list TEMPLST still exist in library QGPL and that source members still exist in the source files used at object compilation. Any objects in TEMPLST that do not exist in QGPL, or that do not have source members are to be removed from the list:

```
YCHKLSTE CHKLIB(QGPL) CHKFILE(*OBJLST) +  
UPDLST(*RMVERR)
```

Examples of use of output list and list flags are as follows:

Check that items in a member list which are flagged with an 'S' ('Submitted') have an object existing in the current library associated with the job. Any items which do not, are to be copied to a separate output list called FAILED in QTEMP. The copied items should be unflagged:

```
YCHKLSTE LSTTYPE(*MBR) UPDLST(*RMVOK)  
CHKLIB(*CURLIB) FLAGVAL(*SUBMITTED)  
OUTFLAGVAL(*NULL) OUTLST(QTEMP/FAILED)
```

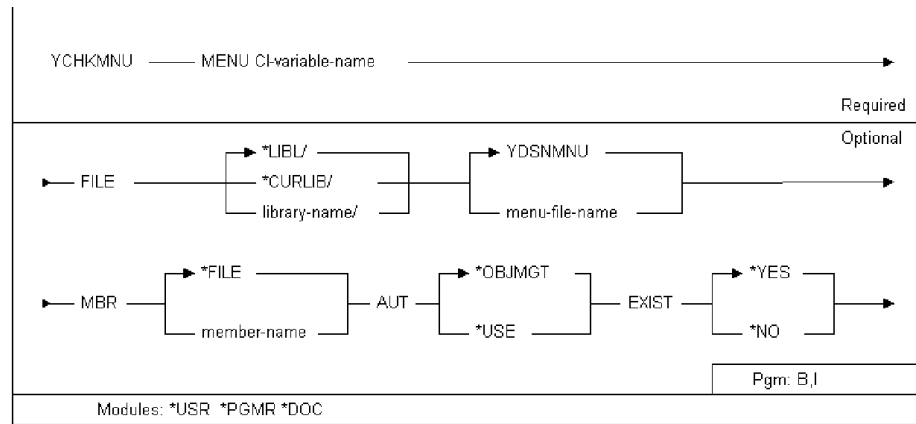
Check that the user has full authority to objects in list TEMPLST in QTEMP. any objects to which the user does not have authority are to be placed in an output list GRTOBJAUT in QTEMP, and flagged with an 'A':

```
YCHKLSTE LSTTYPE(*OBJ) UPDLST(*RMVOK  
*FLAGERR) CHKLIB(*ST)AUT(*ALL)  
OUTFLAGVAL(A) OUTLST(QTEMP/GRTOBJAUT)
```

YCHKMNU (Check Menu)

Checks for the existence of a menu file and is intended primarily for internal use by the utilities.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
MENU	CL variable containing the name of the checked menu or a special value. Must be a character variable, ten bytes long	If *NONE, *ALL, *SELECT, or a generic name is specified, then no menu existence check is made
FILE	Qualified name of menu file containing menu	*LIBL/YDSNMNU: (default) Menu file name
MBR	Name of member in file containing menu	*FILE: (default) Member name is same as file name

Parameter	Definition	Value and Description
AUT	Authorization rights to check	<ul style="list-style-type: none"> ■ *OBJMGT: (default) Corresponds to authority rights *OBJOPR, *ADD, *READ, *DLT, *UPD for the menu file ■ *USE: Corresponds to authority rights *OBJOPR AND *READ for the menu file
EXIST	Check whether specified menu exists	<ul style="list-style-type: none"> ■ *YES: (default) Issue escape message if the menu specified by the MENU parameter is not found ■ *NO: Issue escape message if menu is found

Notes

1. If errors occur on the checks, i OS or CA2E escape messages are issued. File, member, and authority checking is performed using the i OS command Check Object (CHKOBJ). See the *i OS Control Language Reference Manual* for details of the escape messages issued by the CHKOBJ command.

In addition to the escape messages sent by i OS, the following escape messages may be sent:

- *YMN0006 - Specified file is not a menu file
- *YMN0011 - Specified menu not found
- *YMN0012 - Specified menu already exists

Example

To check whether menu FRED exists in file *LIBL/YDSNMNU and whether you have object management rights to it:

```
DCL VAR(&MNU) TYPE(*CHAR) LEN(10) /*Menu name */
CHGVAR VAR(*MNU) VALUE('FRED')
YCHKMNU MENU(&MNU)
```


Parameter	Definition	Value and Description
AUT	Authority to check	<ul style="list-style-type: none"> ■ *OBJMGT: (default) Checks that the user has *OBJOPR, *ADD, *READ, *DLT and *UPD authority for the panel file ■ *USE: Checks that the user has *OBJOPR and *READ authority for the panel file
EXIST	Panel checking option; only applies if a panel design name is specified	<ul style="list-style-type: none"> ■ *YES: (default) The panel design must exist ■ *NO: The panel design cannot exist
PNLDTA	CL variable where the panel title is recorded	Must be a character variable 80 characters long. You may obtain a layout of the file that defines the subfields of this variable by using the command Document file (YDOCF)
YDOCF YDSNPNL		

Notes

In addition to the i OS escape messages generated by CHKOBJ, the following escape messages can be issued:

- *YDS0002 File is not panel file
- *YDS0004 Panel not found
- *YDS0005 Panel already exists

Example

To check whether panel design FRED exists in file *LIBL/YDSNSCR and whether you have object management rights to it:

```
DCL VAR(&PNL) TYPE(*CHAR) LEN(10)      /* Panel name */
DCL VAR(&PNLDTA) TYPE(*CHAR) LEN(80)    /* Panel record */
DCL VAR(&TEXT) TYPE(*CHAR) LEN(50)      /* Panel title */
CHGVAR VAR(&PNL) VALUE('FRED')
YCHKPNL PANEL(&PNL) PNLDTA(&PNLDTA)
```

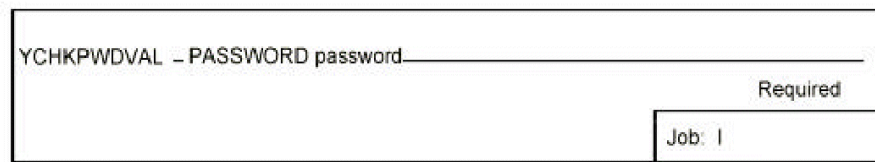
If panel design FRED is successfully found, then you might obtain its title as follows:

```
CHGVAR VAR(&TEXT) VALUE(%SST(&PNLDTA 15 50))
```

YCHKPWDVAL (Check New Password)

This command checks a password to ensure that it meets the specified criteria.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
PASSWORD	Password to be checked	

Notes

1. The nature of the password checking depends on the password control values specified on the Edit Password values display (YEDTPWDVAL command). According to the values specified the YCHKPWDVAL command tests:
 - Whether the password is the name of an existing user profile.
 - Whether the password is on a list of forbidden values.
 - If the new password fails any of the tests, an escape message is sent.

Controlled By	Escape Message
YALWPRFPWD YPWDCHKLVL	YYY0014 Password is name of an existing profile YYY0015 Password is not allowed
YALWPRFPWD YPWDCHKLVL	YYY0014 Password is name of an existing profile YYY0015 Password is not allowed

- The values required for i OS password validation (available in i OS release 2.0) are not checked by this command.

Example

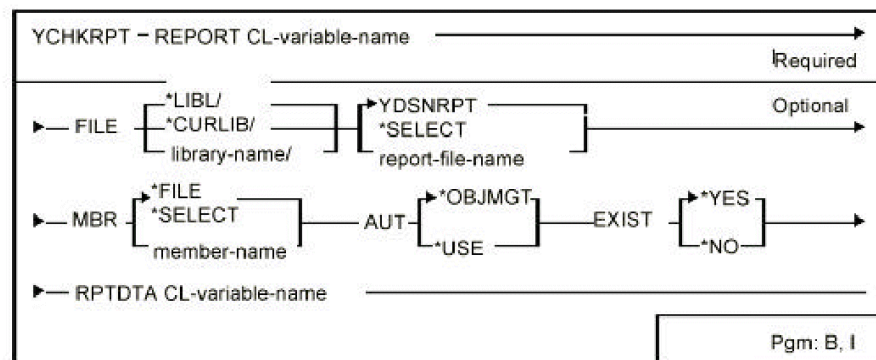
To check that password SESAME is allowed:

```
YCHKPWDVAL PASSWORD(SESAME)
```

YCHKRPT (Check Report Design)

This command checks for the existence of an E report design, and/or report design file. This command is intended primarily for internal use by the utilities.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
REPORT	CL variable containing the name of the checked report design or a special value. Must be a character variable ten characters long	<p>The following special values may be used:</p> <ul style="list-style-type: none"> ■ *SELECT: The report design selection display appears ■ *NONE, generic name, or *ALL: no report check is performed

Parameter	Definition	Value and Description
FILE	Qualified name of a file containing report design.	*LIBL/YDSNRPT: (default) Report file name file must have report design format
MBR	Name of member in the file containing report design.	*FILE: (default) The member has the same name as the file
EXIST	Report checking option; only applies if a report design name is specified.	<ul style="list-style-type: none"> ■ *YES: (default) The report design must exist ■ *NO: The report design must not exist
AUT	Authority to check	<ul style="list-style-type: none"> ■ *OBJMGT: (default) Checks that the user has *OBJOPR, *ADD, *READ, *DLT and *UPD authorities to the report file ■ *USE: Check that the user has *OBJOPR and *READ authorities to the report file
RPTDTA	CL Variable that have report title information returned to it. Must be a character variable 80 characters long You may obtain a layout of the report title record by using the command Document file (YDOCF):	
YDOCF YDSNRPT		

Notes

In addition to the i OS escape messages generated by CHKOBJ, the following escape messages can be issued:

- *YDS0103 'File is not report file.'
- *YDS0104 'Report not found.'
- *YDS0105 'Report already exists.'

Example

To check whether report design FRED exists in file *LIBL/YDSNRPT and whether you have object management rights to it:

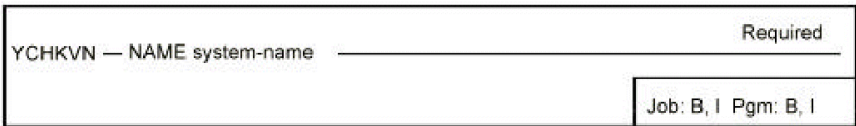
```
DCL VAR(&RPT) TYPE(*CHAR) LEN(10) /* Report name */
DCL VAR(&RPTDTA) TYPE(*CHAR) LEN(80) /* Report record */
DCL VAR(&TEXT) TYPE(*CHAR) LEN(50) /* Report title */
CHGVAR VAR(&RPT) VALUE('FRED')
YCHKRPT REPORT(&RPT) RPTDTA(&RPTDTA)
```

If report FRED is successfully found, then you might obtain its title as follows:

YCHKVN (Check System Name Is Valid)

This command checks that a character string is a valid system name. This command is intended primarily for internal use by the utilities.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
NAME	Name to be checked and cannot exceed ten characters or have embedded blanks.	

Notes

The YCHKVN command may be useful to help validate input into your interactive programs when the CHECK(VN) keyword cannot be used - for instance, when you have special values beginning with an '*' such as '*LIBL', '*ALL'.

If there are any errors a diagnostic message (CPD0078) and an escape message (CPF0001) is sent.

Example

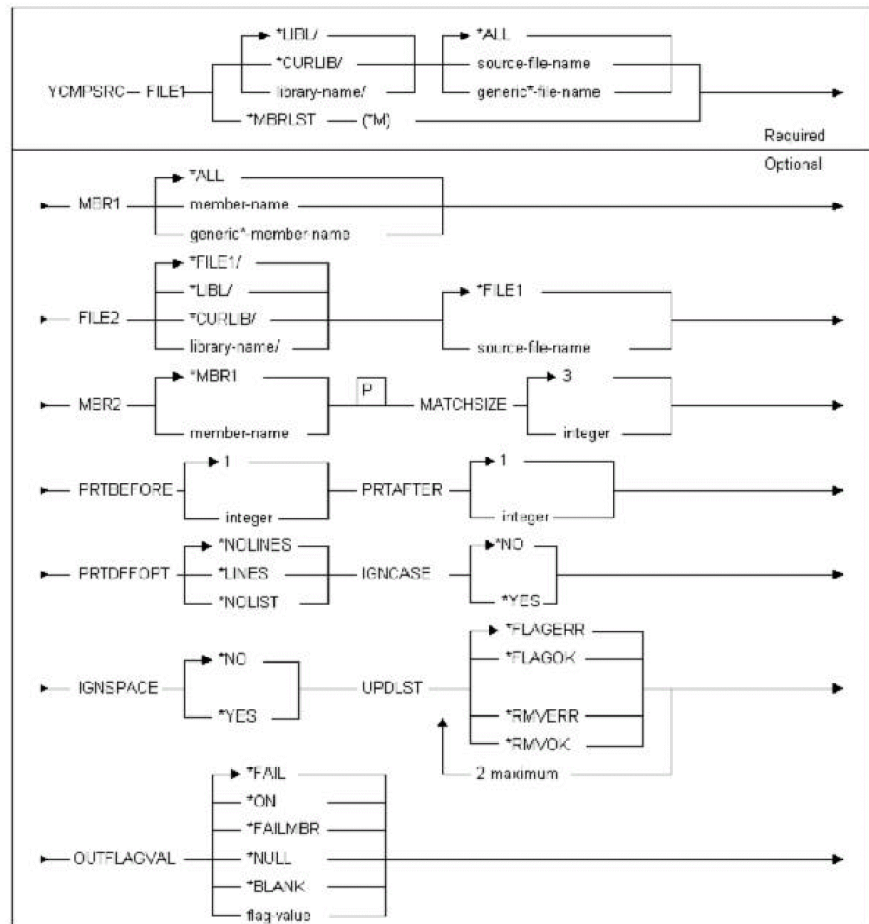
To check whether FRED is a valid system name:

```
YCHKVN NAME(FRED)
```

YCMPSRC (Compare Source)

Compares a pair of source file members and reports on any mismatches. Multiple members can be compared and the degree of detail reported can be controlled.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
FILE1	Qualified name of source file whose records are compared with those of the file specified in the FILE2 parameter	<ul style="list-style-type: none"> ■ *MBRLST: (default) The source file is obtained from the member list specified in the MBRLST parameter. *MBRLST can be abbreviated to *M ■ *ALL: All of the source files
MBR1	The name of source member specified by parameter FILE1 compared with member specified in parameter MBR2	*ALL: (default) All the members of file specified by FILE1 participate in the comparison
FILE2	Qualified name of source file whose records are compared with those of the file specified in the FILE1 parameter.	<p>*FILE1: (default) Second member compared is obtained from the file specified in parameter FILE1.</p> <p>If a generic file name is specified for the FILE1 parameter, or a value of FILE1(*ALL), or FILE1(*MBRLST), then FILE2(*FILE1) must be specified</p>
MBR2	Name of source member specified by parameter FILE2 compared with member specified in parameter MBR1	<p>*MBR1: (default) Name of second member is the same as that specified in the MBR1 parameter.</p> <p>If FILE1 is the same as FILE2, the MBR1 should not be the same as MBR2.</p>
MATCHSIZE	Number of lines required for a match	3: Three lines must match
PRTBEFORE	Number of preceding source lines to be printed when a mismatch is encountered	1: Print the line before the mismatch
PRTAFTER	Number of source lines following the mismatch that must be printed	1: Print the line after the mismatch

Parameter	Definition	Value and Description
PRTDFFOPT	Degree of detail to be reported	<ul style="list-style-type: none"> ■ *NOLINES: (default) Print the mismatched lines without the associated sequence numbers ■ *LINES: Print the mismatched lines with the associated sequence numbers ■ *NOLIST: No mismatch details were reported. Indicate whether or not the source file members match
IGNCASE	Case-sensitivity option	<ul style="list-style-type: none"> ■ *NO: (default) Treat upper and lower case characters as different for comparison purposes ■ *YES: Ignore any differences between upper and lower case for comparison purposes. Thus the character 'a' matches with both 'a' and 'A'
IGNSPACE	Blank character sensitivity option	<ul style="list-style-type: none"> ■ *NO: (default) All characters, including blanks, in both lines undergoing a match-check must be the same ■ *YES: Blanks are ignored when comparing source lines. The two lines shown above would be regarded as matching

Parameter	Definition	Value and Description
UPDLST	List update option	<p>Up to two of the following values:</p> <ul style="list-style-type: none">■ *FLAGERR: (default) If any line of the two members being compared do not match, then the list entry for MBR1 is flagged with the value specified by the OUTFLAGVAL■ *FLAGOK: If all lines of the two members being compared match, then the list entry for MBR1 is flagged with the value specified by the OUTFLAGVAL■ *RMVERR: If any line of the two members being compared does not match then the list entry for MBR1 is removed from the list■ *RMVOK: If all lines of the members being compared match, then the list entry for MBR1 is removed from the list <p>If more than one value is specified, *RMVERR + *FLAGOK and *RMVOK + *FLAGERR are the only valid combinations</p>
OUTFLAGVAL	Flag value used when the UPDLST parameter is *FLAGERR or *FLAGOK	<p>*FAIL: (default) Flag members do not match as 'failed'</p> <p>Single character flag value or one of the special flag values</p>
MBRLST	Qualified name of a listed member	QTEMP/TEMPLST: (default) List name
EDIT	List option	<ul style="list-style-type: none">■ *NO: (default) No editing is required■ *YES: Invoke the edit member list function to edit the list before executing the scan

Notes

1. If every mismatch between two members is reported then, to continue with the comparison after each mismatch, the utility must resynchronize the source members to the next possible match, if any. The utility considers the members resynchronized when the MATCHSIZE, number of consecutive lines, have matched, using the following method.

MBR1 is read one line at a time. For each line read from MBR1, a record is read from MBR2 and the two lines compared.

If the comparison is exact, then a match is made and the next record is read from MBR1 and compared with the next record in MBR2; this is repeated until the MATCHSIZE number of records match.

If the comparison fails, then if (a) a partial match has been achieved, the record from MBR2 is taken as the starting point and the process repeated from 1.2. Otherwise (b) if a partial match has not yet been achieved, then the next record from MBR2 is read, and tested until either a match is found or the end of the file is detected.

2. The use of a member list or a generic member name for FILE1 enables two versions of a source file to be compared, for instance all the RPG/400 source members in a development system can be compared with their corresponding RPG/400 members in the production system.

Examples

To compare source from member FONS in file FREDSRC in library QGPL with member ABINITIO in the same file, ignoring any case differences in characters as well as ignoring spaces between characters:

```
YCMPSRC FILE1(QGPL/FREDPRDSRC) MBR1(FONS) MBR2(ABINITIO)
IGNCASE(*YES) IGNSPACE(*YES)
```

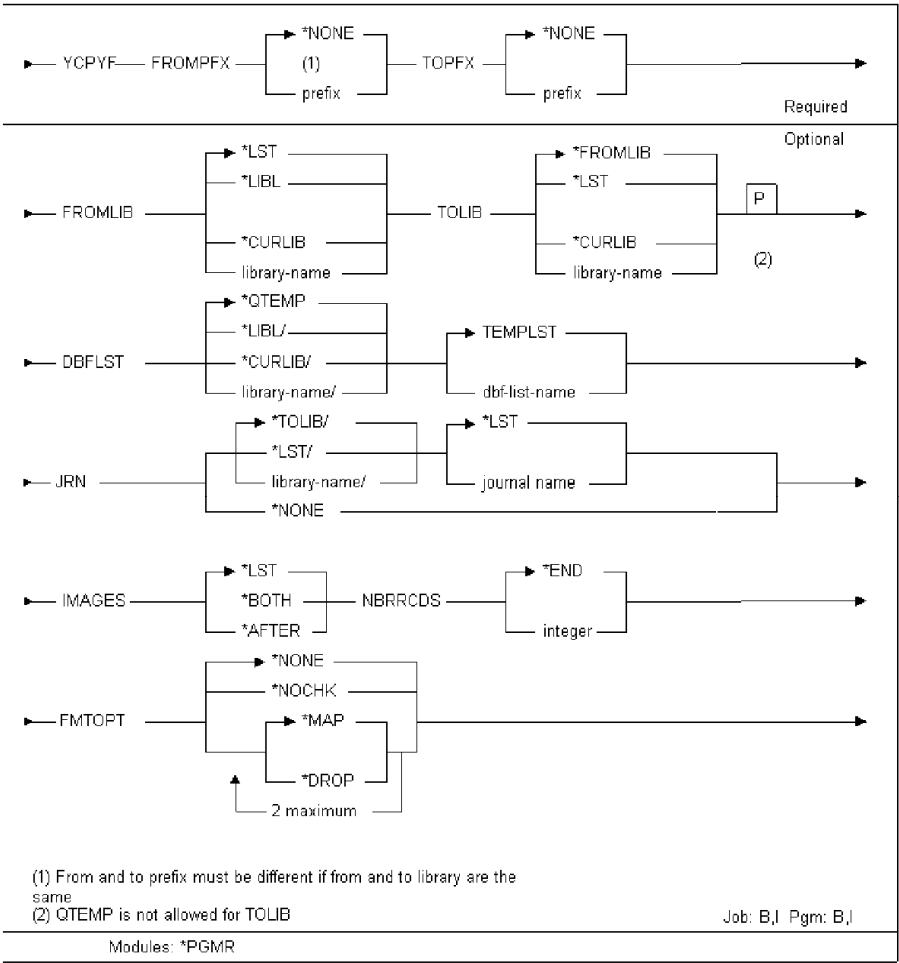
To compare source from members in a list HONMP in library QGPL with versions of the same members in file FREDPRDSRC in library QGPL:

```
YCMPSRC FILE1(*MBRLST) MBR1(*ALL)
FILE2(QGPL/FREDPRDSRC) MBR2(*MBR1)
MBRLST(QGPL/HONMP)
```

YCPYF (Copy Files)

This command copies a list of database physical files. This allows a synchronized "snapshot" of the contents of a list of files that can be used as a recovery, testing or debugging aid. A prefix facility enables many snapshots of the same database files to be stored in the same library.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
FROMPFX	Prefix part of the file name; and is replaced by the TOPFX value in the name of the copied version of the file	<p>*NONE: (default) No prefix is used</p> <p>Character, up to three digits</p>
TOPFX	Prefix that replaces the FROMPFX value in the name of the copied files	<p>*NONE: (default) Copied file names are to be the same as the file names</p> <p>Character, up to three digits, no imbedded blanks</p>
FROMLIB	Library containing copied files	<ul style="list-style-type: none"> ■ *LST: (default) Use library name or names given by the list entry for each file ■ *LIBL: Use library list to find 'from' file
TOLIB	Copies are placed in the library. If copies already exist in the 'to' library their contents are replaced. If copies do not exist in the 'to' library they are created	<ul style="list-style-type: none"> ■ *FROMLIB: (default) Same as value specified by FROMLIB parameter ■ *LST: Use library name or names given by the list entry for each file
DBFLST	Qualified name of a database file list that specifies the names of the copied files	QTEMP/TEMPLST: (default) List name

Parameter	Definition	Value and Description
JRN	The journal name used after the copy is made, if the file specified by the TOFILE parameter already exists as journal	<p>*NONE: Do not restart journal</p> <p>The following special value can be specified for the journal name:</p> <p>*LST: Use journal named for file in list</p> <p>The following special values can be specified for the journal library name:</p> <ul style="list-style-type: none"> ■ *TOLIB: (default) Journal library is the same as that specified for the TOLIB parameter ■ *LST: Use the journal library named for file in the database list <p>Journaling is restarted if the file was already processed</p>
IMAGES	Nature of journaling	<ul style="list-style-type: none"> ■ *LST: (default) Before or after images are kept according to the value specified for file in database list ■ *BOTH: Before and after images are kept ■ *AFTER: Only after images are kept
NBRRCDs	Number of records to copy	*END: (default) Copy until end of file
FMTOPT	Format mapping option used when copying to existing files	<ul style="list-style-type: none"> ■ *NONE: (default) Formats must match in both files ■ *NOCHK: Copy even if formats do not match ■ *MAP: If formats differ map fields by name ■ *DROP: If formats differ drop fields which cannot be mapped

Notes

The FROMPFX and TOPFX parameters enable the labeling of each set of database files with a unique identifier, and also to keep copies of the same database file in the same library. The names of the copied files are generated by taking each file name in the database list, and appending or substituting the prefixes as shown in the following table:

	FROMPFX	TOPFX	DBF list file name	From file name used	To file name used
Append	*NONE XX	XX *NONE	PCUSFIL PCUSFIL	PCUSFIL XXPCUSFI L	XXPCUSFI L PCUSFIL
Substitute	P X	X P	PCUSFIL PCUSFIL	PCUSFIL XCUSFIL	XCUSFIL PCUSFIL
Substitute & append	P XX	XX P	PCUSFIL PCUSFIL	PCUSFIL XXCUSFIL	XXCUSFIL PCUSFIL
No Change (1)					

Examples

A library UDTA that contains three application files, UORDFILE, UCUSFILE, and UORDLINE, and a "snapshot" to library UTST with a prefix of XX, and then later restore is needed:

A list is built of files that needed copying.

```
YBLDDBFLST FILE(UDTA/U*) FILEATR(*PHY) DBFLST(UDTA/TSTDTA)
```

To create a copy the files in the list in library UTST:

```
YCPYF FROMPFX(U) TOPFX(XX) FROMLIB(UDTA) TOLIB(UTST) DBFLST(UDTA/TSTDTA)
```

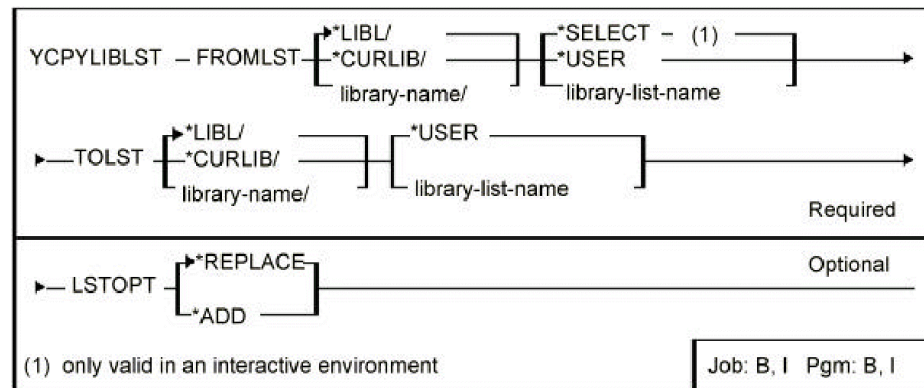
To restore a copy the files in the list from library UTST back to UDTA:

```
YCPYF FROMPFX(XX) TOPFX(U) FROMLIB(UTST) TOLIB(UDTA) DBFLST(UDTA/TSTDTA)
```

YCPYLIBLST (Copy Library List)

This command copies a library list.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
FROMLST	Qualified name of the copied library list	<ul style="list-style-type: none"> ■ *SELECT: Displays the existing lists ■ *USER: List has the same name as invoking job's user profile
TOLST	Qualified name of the new library list	*USER: List has same name as invoking job's user profile
LSTOPT	List replacement option	<ul style="list-style-type: none"> ■ *REPLACE: (default) Replaces an existing list; a list of the specified name must already exist ■ *ADD: Creates a new list; no list of the specified name can exist

Notes

Library lists are stored in file YLIBLST in the library specified by the LIBLST parameter.

Example

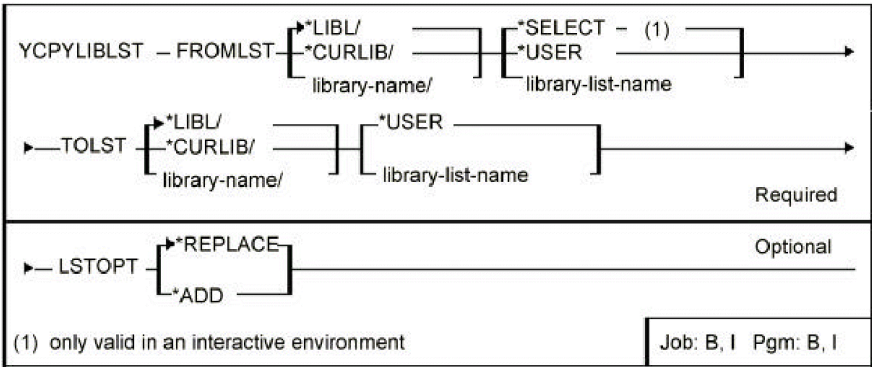
To copy library list DEWEY to CONGRESS:

YCPYLIBLST FROMLST(DEWEY) TOLST(CONGRESS)

YCPYLST (Copy List)

This command copies and optionally renames an object, member, format or database file list.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
LSTTYPE	Identifies the copied list type	<ul style="list-style-type: none">■ *DBF: Database file list■ *FMT: Format list■ *MBR: Member list■ *OBJ: Object list
FROMLST	Qualified name of the copied list	

Parameter	Definition	Value and Description
TOLST	Qualified name of the list to which is copied by FROMLST	
LSTOPT	List option	<ul style="list-style-type: none">■ *REPLACE: (default) Replaces any existing contents of list specified by TOLST■ *ADD: Adds to current TOLST contents

Notes

None

Example

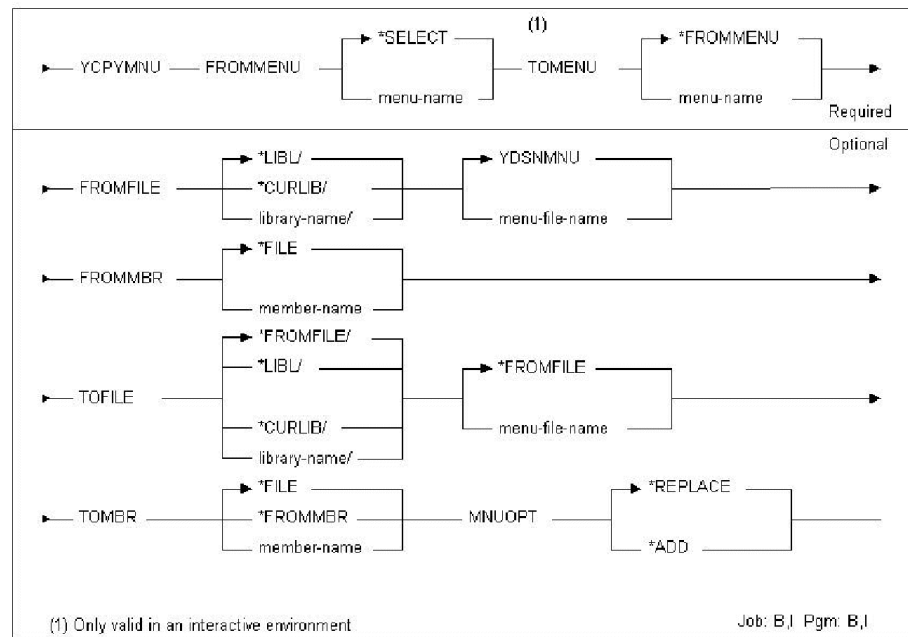
To copy object list QTEMP/TEMPLST to list QGPL/SHOPPING:

```
YCPYLST LSTTYPE(*OBJ) FROMLST(QTEMP/TEMPLST) TOLST(QGPL/SHOPPING)
```

YCPYMNU (Copy Menu)

This command copies a menu.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
FROMMENU	Name of the copied menu. The menu must exist in the FROMFILE and FROMMBR named below	*SELECT: The menu selection displays are presented
TOMENU	Name of the created menu. The menu must not already exist in the TOFILE and TOMBR named below	*FROMMENU: The 'to menu' name is the same as the 'from menu'. This is only permitted if the values of FROMFILE and FROMMBR are not the same as those of TOFILE and TOMBR
FROMFILE	Qualified name of file containing menu that is to be copied	*LIBL/YDSNMNU: (default) Menu file name. File must be created with the YCRTDSNF command, TYPE(*MNU)

Parameter	Definition	Value and Description
FROMMBR	Name of member in FROMFILE	*FILE: (default) Member name is same as FROMFILE
TOFILE	Qualified name of the file that contains the copied menu	*FROMFILE: (default) File name is same as FROMFILE. File must be created with the command YCRTDSNF, TYPE(*MNU)
TOMBR	Name of member in TOFILE	<ul style="list-style-type: none">■ *FILE: (default) Member name is same as TOFILE■ *FROMMBR: Member name is same as FROMMBR
MNUOPT	Menu option	<ul style="list-style-type: none">■ *REPLACE: (default) Replace any existing contents of menu specified by TOMNU■ *ADD: Only copy if TOMNU does not already exist

Notes

User must have the permission to add right to the TOFILE.

Example

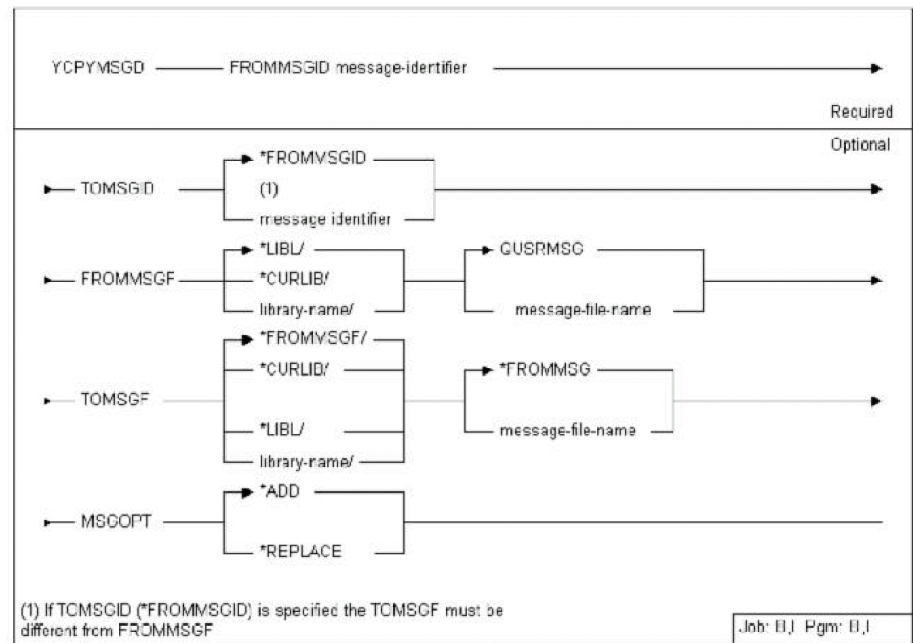
To copy menu FRED from file YDSNMNU in library QGPL to file YDSNMNU in library BORIS:

```
YCPYMNU FROMMENU(FRED) TOMENU(FRED) FROMFILE(QGPL/YDSNMNU) TOFILE(BORIS/YDSNMNU)
```

YCPYMSGD (Copy Message Description)

This command copies a message description.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
FROMMSGID	Identifies the copied message description. Must be valid i OS message identifier	
TOMSGI	Identifies the added message description	*FROMMSGID: (default) The same value as that specified by the FROMMSGID parameter. Must be valid i OS message identifier
FROMMSGF	Qualifying name of message file containing copied message description	QUSRMSG: (default) Message file name.
TOMSGF	Qualifying name of message file to which message description is to be copied	*FROMMSGF: (default) The TOMSGF is the same name as that specified by the FROMMSGF parameter

Parameter	Definition	Value and Description
MSGOPT	Option to replace any existing message in the TOMSGF or to add it if it is not already there	<ul style="list-style-type: none">■ *ADD: (default) Add the copied message to the TOMSGF; if the message already exists in the TOMSGF, do not copy■ *REPLACE: If the message is present in the TOMSGF, replace it with the copied message. If the message is not present in the TOMSGF, add it

Notes

The command uses the spooled output from the 'Display Message Description' command DSPMSGD.

Example

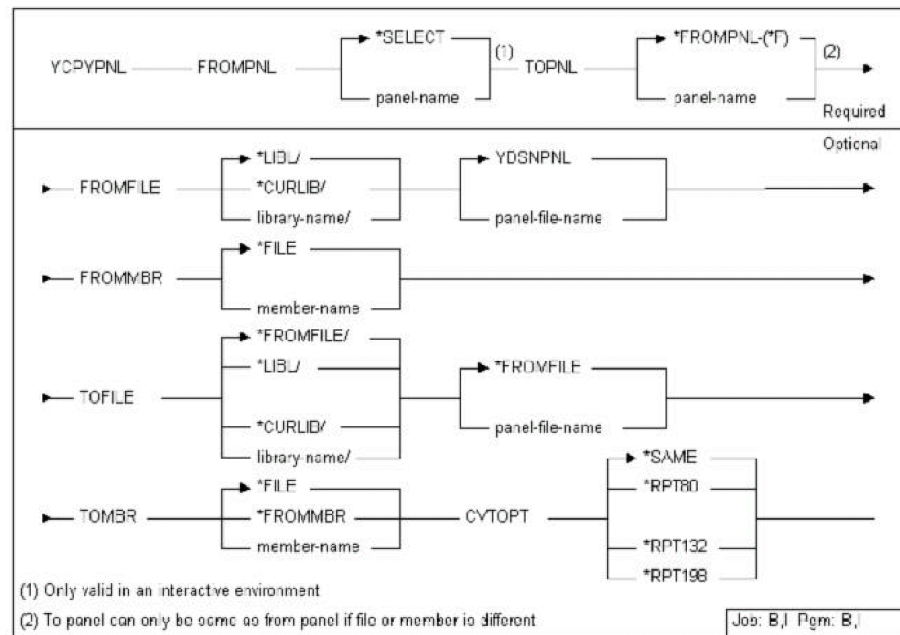
To copy the message description for USR0001 in UMSGF in library QGPL, to USR0443 in VMSGF in MYLIB:

```
YCPYMSGD FRMSGID(USR0001) TOMSGID(USR0443) FRMSGF(QGPL/UMSGF)
TOMSGF(MYLIB/VMSGF)
```

YCPYPNL (Copy Panel Design)

This command copies a panel design.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
FROMPNL	Name of the copied panel design	*SELECT: The panel selection display is presented
TOPNL	Name of new panel design	*FROMPNL: The new panel is to have the same name as the original. Only allowed if the values of the TOFILE or TOMBR parameters are different from those of the FROMFILE or FROMMBR parameters
FROMFILE	Qualified name of the file containing panel design	*LIBL/YDSNPNL: (default) Panel file name The file must have been created with the command Create Design File (YCRTDSNF)

Parameter	Definition	Value and Description
FROMMBR	Name of member in file containing panel design	*FILE: (default) The member has the same name as that specified by the FROMFILE parameter
TOFILE	Qualified name of panel design file to contain copied designs	*FROMFILE: (default) The file has the same name as the from file File must be design file unless CVTOPT(*RPT) is specified, in which case it must be a report design file. (Both panel and report designs can be created with the YCRTDSNF command)
TOMBR	Name of member in destination file	<ul style="list-style-type: none"> ■ *FILE: (default) The member has the same name as the destination file name ■ *FROMMBR: The member has the same name as FROMMBR
CVTOPT	Convert option	<ul style="list-style-type: none"> ■ *SAME: (default) Do not convert panel design ■ *RPT80: Convert panel design into a report design: 80 columns wide ■ *RPT132: Convert panel design into a report design: 132 columns wide ■ *RPT198: Convert panel design into a report design: 198 columns wide

Notes

1. User must have add rights to the file specified by the TOFILE parameter.
2. If a panel design is converted into a report design, the from file must be different from the to file.

Examples

To copy panel design FRED to panel PERCIVAL in the same file, YDSNPNL, in library BASIL:

```
YCPYPNL FROMPNL(FRED) TOPNL(PERCIVAL)
FROMFILE(BASIL/YDSNPNL)
```

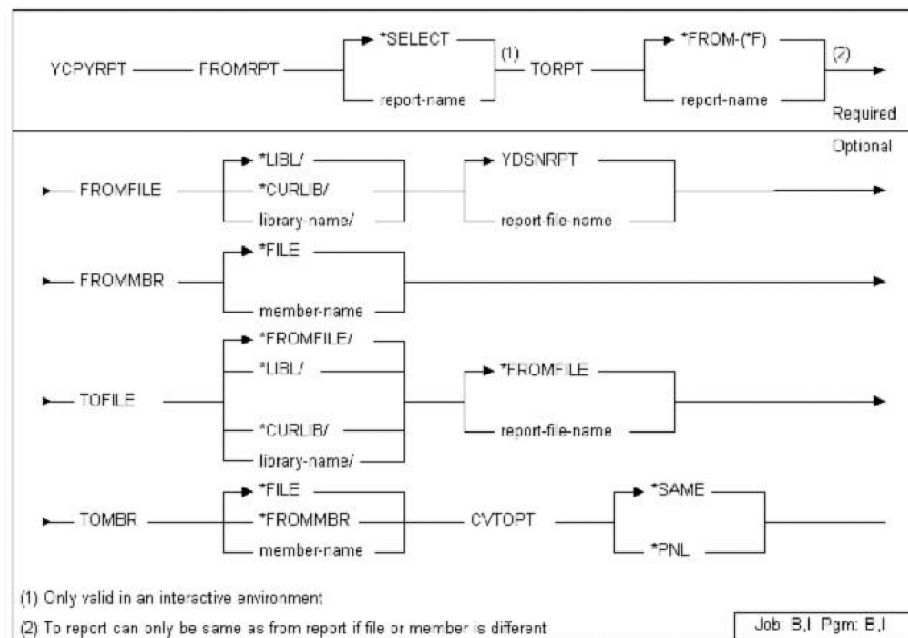

To copy panel design FRED from file YDSNPNL in library GEORGE to an 80 column report, PERCIVAL, in file YDSNRPT in the same library:

```
YCPYPNL FROMPNL(FRED) TOPNL(PERCIVAL) FROMFILE(GEORGE/YDSNPNL)
TOFILE(GEORGE/YDSNRPT) CYTOPT(*RPT80)
```

YCPYRPT (Copy Report Design)

This command copies a report design.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
FROMRPT	Name of report copied design	*SELECT: The report selection display is presented
TORPT	Name of new report design	*FROMRPT: The new report has the same name as the original. Only allowed if value of TOFILE or TOMBR is different from that of FROMFILE or FROMMBR
FROMFILE	Qualifying name of file containing report design	*LIBL/YDSNRPT: (default) Report file name. File must have report design format
FROMMBR	Name of member in file containing report design	*FILE: (default) The member has the same name as the file specified by the FROMFILE parameter
TOFILE	Qualifying name of report file to contain copied designs	<p>*FROMFILE: (default) The file has the same name as the file specified by the FROMFILE parameter</p> <p>File must be a report design file unless CVTOPT(*PNL) is specified, in which case it must be a panel design file (both can be created with the YCRTDSNF command)</p>
TOMBR	Name of member in destination file	<ul style="list-style-type: none"> ■ *FILE: (default) The member has the same name as the file specified by the TOFILE parameter ■ *FROMMBR: The member has the same name as FROMMBR
CVTOPT	Convert option	<ul style="list-style-type: none"> ■ *SAME: (default) Do not convert report design ■ *PNL: Convert report design into a panel design

Notes

1. User must have add rights to the file specified by the TOFILE parameter.
2. If a report design is converted into a screen design, the from and to files must be different. If a report design is larger than 24 lines x 80 columns, it is truncated.

Examples

To copy report design FRED from file YDSNRPT in library QGPL to file YDSNRPT in library GEORGE:

```
YCPYRPT FROMRPT(FRED) FROMFILE(QGPL/YDSNRPT) TOFILE(GEORGE/YDSNRPT)
```

To copy report design FRED from file YDSNRPT in library QGPL to panel FRED in panel design file YDSNPNL in library MORRIS:

```
YCPYRPT FROMRPT(FRED) FROMFILE(QGPL/YDSNRPT) TOFILE(MORRIS/YDSNPNL) CVTOPT(*PNL)
```

YCPYUSRPRF (Copy User Profile)

This command copies a specified user profile.

Parameter	Definition	Value and Description
AUT	Authority given to the public for the new user profile	<ul style="list-style-type: none"> ■ EXCLUDE: (default) No public authority is given ■ *ALL: All authorities are given to the public ■ *USE: Normal authority to use the profile is given to the public ■ *CHANGE: Normal authority to change the profile is given to the public
MSGQ	Qualified name of message queue associated with the new user profile	<ul style="list-style-type: none"> ■ *TOUSRPRF: (default) Use default message queue with same name as the new user profile ■ *FROMUSRPRF: Use default message queue with same name as the copied user profile
GRTUSRAUT	Specifies whether to grant the copied from profile's authorities to the new profile	<ul style="list-style-type: none"> ■ *NO: (default) Do not grant the authority ■ *YES: Grant the authority of the copied profile to the new profile

Notes

Security officer rights are required for this command.

Example

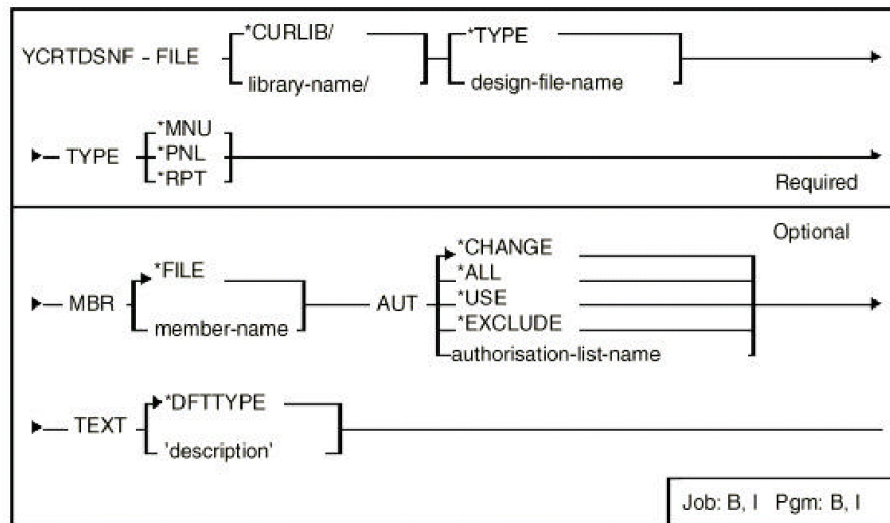
To copy a user profile called OGEE to form a new profile called CIMARECTA:

```
CPYUSRPRF FROMUSRPRF(OGEE) TOUSRPRF(CIMARECTA) TEXT('Cimarecta profile')
```

YCRTDSNF (Create Design File)

Creates database files to contain menu, report or panel designs. Files are created with the correct format and any required dependent logical views. This is the recommended way of creating design files.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
FILE	Qualified name of file that is to be created	<p>*TYPE: (default) Use default file name for type, for example if type is *MNU, YDSNMNU, if type is *PNL, YDSNPNL, if type is *RPT, YDSNRPT</p> <p>A library must be specified</p>
TYPE	Type of design file that is to be created	<ul style="list-style-type: none"> ■ *MNU: (default) Menu file ■ *PNL: Panel design file ■ *RPT: Report design file ■ MBR: Name of member in file ■ *FILE: (default) Member name is the same as file name

Parameter	Definition	Value and Description
AUT	Public authority of created file	<ul style="list-style-type: none"> ■ *CHANGE: (default) The user has object operational rights and all data rights to the file ■ *ALL: Everyone is authorized to do anything to the file, except transfer ownership ■ *USE: The user has object operational rights and read data rights ■ *EXCLUDE: Prevents the user from accessing the file ■ Authorization list name: The name of an authorization list whose authority is used for the file
TEXT	Descriptive text for member or file	*DFTTYPE: (default) Use default text for type; if type is *MNU, Menu design file, if type is *PNL, Panel design file, if type is *RPT, Report design file

Notes

1. The file name must be nine characters or less.
2. Further members can be added to the design file using the command Add Design File Member (YADDDSNFM).

Examples

To create a new panel design file in library CECIL:

```
YCRTDSNF TYPE(*PNL) FILE(CECIL/YDSNPNL) TEXT('Cecil B de Mille''s panel designs')
```

To create a new report design file in library ALVEY:

```
YCRTDSNF TYPE(*RPT) FILE(ALVEY/YDSNPNL) TEXT('Alvey''s report designs')
```

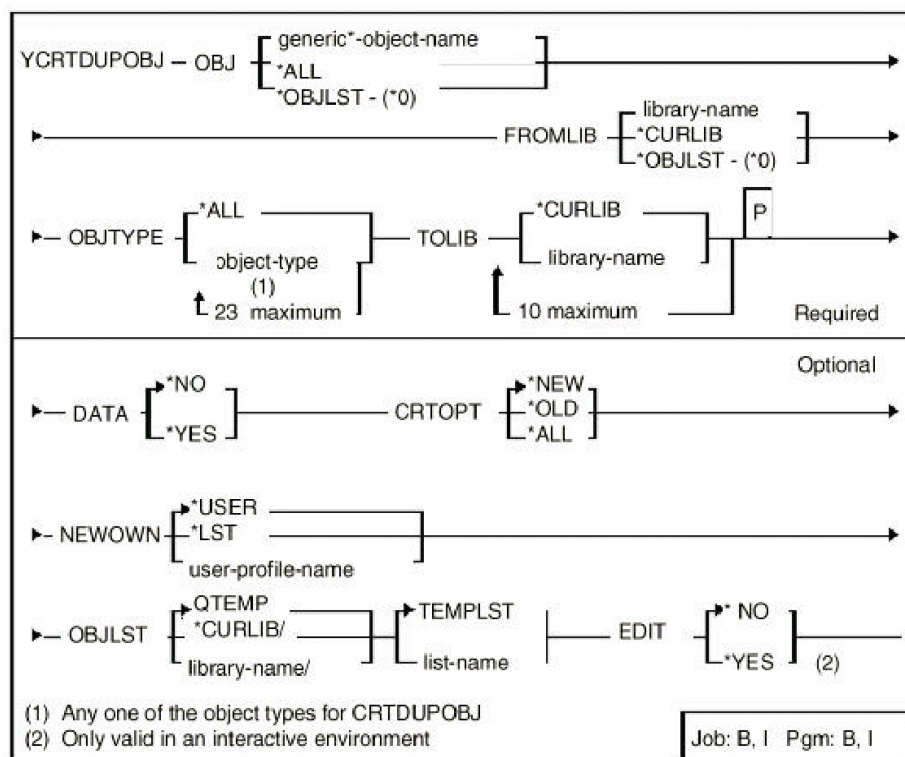
To create a new menu design file in library MAXIM:

```
YCRTDSNF TYPE(*MNU) FILE(MAXIM/YDSNMNU) TEXT('Maxim''s menus')
```

YCRTDUPOBJ (Create Duplicate Objects)

This command creates duplicate objects in several libraries. Objects to be duplicated can be specified generically, or via a list. This also provides a means of replicating a change made to a master library into several parallel libraries.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
OBJ	Generic name of duplicated objects	<ul style="list-style-type: none"> ■ *ALL: All objects ■ *OBJLST: Objects are specified by a list, whose name is given by the OBJLST parameter
FROMLIB	Library containing the objects that are to be duplicated	<ul style="list-style-type: none"> ■ *OBJLST: The name of the library containing the objects is specified by a list, whose name is given by the OBJLST parameter ■ *CURLIB: Duplicate the objects in the current library of the invoking job
OBJTYPE	List of types of object that are to be duplicated	*ALL: All object types that are eligible for duplication by the i OS CRTDUPOBJ command
TOLIB	List of up to ten libraries in which to place copies of the duplicated objects. The libraries specified with the TOLIB parameter must not be specified in the system part of the current job's library list	*CURLIB: Duplicate the objects to the current library of the invoking job
DATA	Duplicate data option	<ul style="list-style-type: none"> ■ *NO: (default) Do not duplicate data for data objects ■ *YES: Duplicate data for data objects

Parameter	Definition	Value and Description
CRTOPT	Duplicate existing objects option	<ul style="list-style-type: none">■ *NEW: (default) Only duplicate objects that do not already exist in the destination library■ *OLD: Only duplicate objects that already exist in the destination library■ *ALL: Duplicate all objects
NEWOWN	User profile to own duplicated objects	<ul style="list-style-type: none">■ *USER: (default) Owner is the current user■ *LST: Owner of each duplicated object is the same as value for owner in the list
OBJLST	Qualified name of object list of objects that are to be duplicated	QTEMP/TEMPLST: (default) List name If OBJ(*OBJLST) is specified the existing list is used. Otherwise a new list is created from the parameters specified
EDIT	Edit list option	<ul style="list-style-type: none">■ *NO: (default) The list edit function is not invoked■ *YES: Edit the list once built: valid only for interactive programs

Notes

1. The objects are duplicated in alphabetical order, but logical files are duplicated separately after all other objects have been duplicated. Note that if CRTOPT(*ALL) or CRTOPT(*OLD) is specified the order of duplication may be significant if there are existing physical files with dependent logical files.
2. Objects are duplicated in turn to each library. If the object type is *FILE or *ALL, the TOLIB library is placed at the front of the library list, before objects are duplicated. Thus, providing the based-on physical file exists in the TOLIB, any logical file created is based on the appropriate copy of the physical file.
3. If *OBJLST is specified for the OBJ parameter, but not for the FROMLIB parameter, then the command uses the list to obtain the object name, but uses the given FROMLIB value for the library name of the objects to be duplicated.

Examples

To duplicate objects whose names begin with 'PF' in LIVDTA into libraries TSTDALIB1, TSTDALIB2, and TSTDALIB3:

```
YCRTDUPOBJ OBJ(PF*) FROMLIB(LIVDTA) OBJTYPE(*ALL)
TOLIB(TSTDALIB1 TSTDALIB2 TSTDALIB3)
```

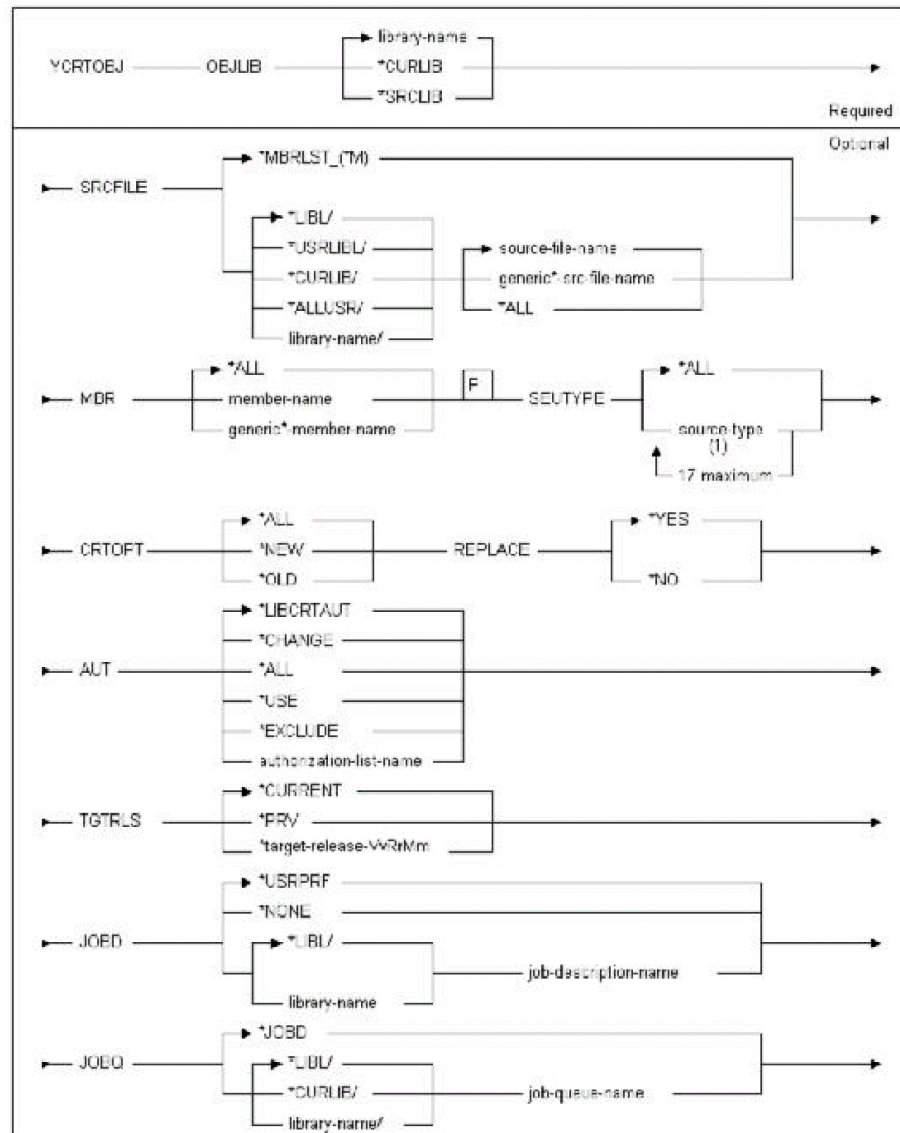
To duplicate all objects in object list FRED into libraries TSTDALIB1, TSTDALIB2, and TSTDALIB3:

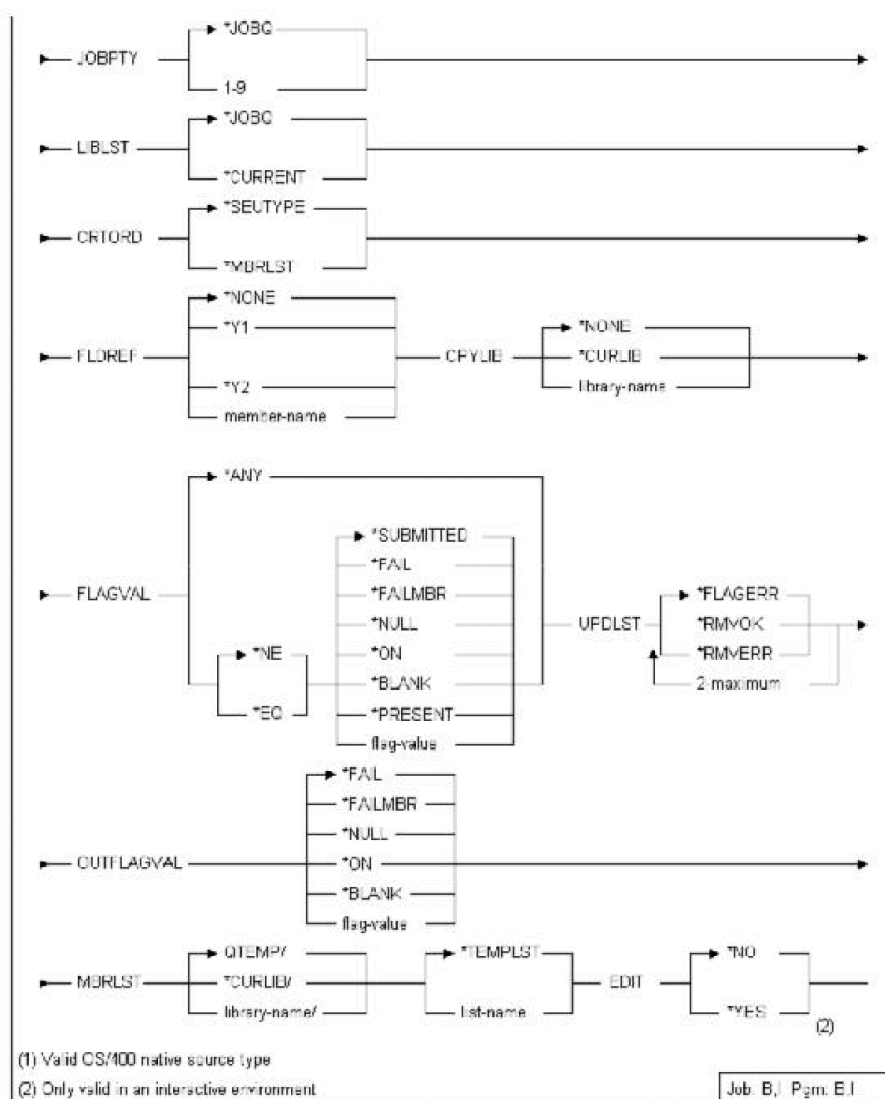
```
YCRTDUPOBJ OBJ(*OBJLST) FROMLIB(*OBJLST) OBJTYPE(*ALL)
TOLIB(TSTDALIB1 TSTDALIB2 TSTDALIB3) CRTOPT(*ALL) OBJLST(FRED)
```

YCRTOBJ (Create Objects)

This command compiles a list of source file members. The list is specified by a generic name, or the member list function.

Syntax Diagram





Parameters

Parameter	Definition	Value and Description
OBJLIB	Library where compiled objects are placed	<ul style="list-style-type: none"> ■ *CURLIB: (default) Place objects in current library for the invoking job ■ *SRCLIB: Place each object in the library from which the source member for that object was obtained
SRCFILE	Qualified generic name of the file containing source members	<ul style="list-style-type: none"> ■ *MBRLST: (default) Locate source members to be compiled by the member list specified by the MBRLST parameter ■ *ALL: all files
MBR	Generic member name of members that are to be compiled	*ALL: (default) Compile all members
SEUTYPE	List of SEU source types of members that are to be compiled	<p>*ALL: (default) All source types in the file are compiled, including certain non-native source types</p> <p>Must be a valid i OS native source type.</p>
CRTOPT	Create option	<ul style="list-style-type: none"> ■ *ALL: (default) All specified members are compiled ■ *OLD: Only members for objects that already exist in the create library are compiled ■ *NEW: Only members for objects that do not already exist in the create library are compiled

Parameter	Definition	Value and Description
REPLACE	Specifies replace option to use on create command	<ul style="list-style-type: none"> ■ *YES: (default) Specifies a new object are created and any existing object of the same name in the specified library are moved to library QRPLOBJQRPLOBJ i OS library YCRTOBJ. ■ *NO: Specifies a new object is not created if an object with the same name already exists in the specified library
TGTRLS	Specifies Target Release option to use on the create command in the form VvRrMm, where <i>v</i> is the i OS version, <i>r</i> is the i OS release number, and <i>m</i> is the i OS machine level; for example, V2R1M0, or one of the following special values	<ul style="list-style-type: none"> ■ *CURRENT: (default) Generate program objects to run on the current release of i OS ■ *PRV: Generate program objects to run on the previous release of i OS

Parameter	Definition	Value and Description
AUT	Public authority of compiled object	<ul style="list-style-type: none">■ LIBCRTAUT: (default) The authority is specified by the Create Authority (CRTAUT) attribute of the library where the object is compiled■ *CHANGE: The user has object operational rights and all data rights to the object■ *ALL: Everyone is authorized to do anything to the object, except transfer ownership■ *USE: The user has object operational rights and read data rights■ *EXCLUDE: Prevents the user from accessing the object■ Authorization list name: the name of an authorization list whose authority is used for the object
JOBQ	Qualified name of used job description	<ul style="list-style-type: none">■ *USRPRF: (default) Use default job description for user profile■ *NONE: Create objects interactively
JOBQ	Qualified name of complied job queue	*JOBQ: (default) Use the job queue specified by the job description named with the JOBQ parameter
JOBPTY	Job scheduling priority 1 (high), to 9 (low)	*JOBQ: (default) Use priority on job description

Parameter	Definition	Value and Description
LIBLST	Uses initial library list for compilation, either qualified name of a library list or one of the following special values	<ul style="list-style-type: none"> ■ *JOB: (default) Use library list specified by the job description named with the JOB parameter ■ *CURRENT: Use the current library list of the job which invokes the YCRTOBJ command
CRTORD	Order in which to create objects	<ul style="list-style-type: none"> ■ *SEUTYPE: (default) submit compilations in the following order: <ul style="list-style-type: none"> – Field reference file, if any: see FLDREF parameter below – Physical files – Logical files and Commands – Device files – All other objects ■ *MBRLST: (default) submit compiles in the order in which member names appear in the member list
FLDREF	Name of field reference file member; if a member of the specified name is found in the member list, it is to be compiled before all other source members	<ul style="list-style-type: none"> ■ *NONE: (default) no field reference file member is to be compiled ■ *Y1: retrieve the name of the field reference file from the design default value ■ *Y2: retrieve the name of the field reference file from the YFRFVNM model value of the first model found in the invoking job's library list
CPYLIB	Name of library from which to copy data to add to new physical files	<ul style="list-style-type: none"> ■ *NONE: (default) data is not copied for existing files ■ *CURLIB: copy data from current library for invoking job

Parameter	Definition	Value and Description
FLAGVAL	Flag value of list entries to select for creation	<p>A list parameter made up of the following two elements:</p> <ul style="list-style-type: none">■ Relational operator for selection of flags<ul style="list-style-type: none">– *EQ: (default) equal to– *NE: not equal to■ Flag value<ul style="list-style-type: none">Single character flag value or one of the special flag values. See Appendix A for further details on flag values. Otherwise:<ul style="list-style-type: none">– *ANY: Execute all list entries
UPDLST	List update option	<ul style="list-style-type: none">■ *FLAGERR: (default) List items which are not successfully submitted are to be flagged■ *RMVOK: List items which are successfully submitted are to be removed from the list■ *RMVERR: List items which are not successfully submitted are to be removed from the list <p>If *FLAGERR is specified, then *RMVERR is invalid</p>
OUTFLAGVAL	Flag value to be given to flagged entries when the UPDLST parameter is *FLAGERR	<p>*FAIL: Failed flag items when submitted</p> <p>Single character flag value or one of the special flag values. See Appendix A for further details on flag values.</p>
MBRLST	Qualified name of a member list	

Parameter	Definition	Value and Description
QTEMP/TEMPLST: (default) list name		
EDIT	Edit list option	<ul style="list-style-type: none"> ■ *NO: (default) List edit function is not invoked ■ *YES: Invoke the list edit function before proceeding with compilations

Notes

1. You must have object existence rights to delete any existing objects that are to be recompiled.
2. Source members must have a valid source type on them: you should correct the source types before using the Create Objects command. This can be done using the i OS utility Start SEU (STRSEU), the i OS utility Work with Members using PDM (WRKMBRPDM), or if you are using a member list, by means of the command Change List (YCHGLST). If a source type is missing, the CRTOBJ command will supply it in the following cases:
 - QRPGRSRC: Assumes default type RPG
 - QPLISRC: Assumes default type PLI
 - QCMDSRC: Assumes default type CMD
 - QCLSRC: Assumes default type CLP

Source members of type TXT will only be compiled into Spelling aide dictionaries using the i OS command Create Spelling Aid Dictionary (CRTSPADCT) if the source file name is QXTXSRC.
3. You should note that for the successful compilation of certain object types, the appropriate based-on objects must have been compiled previously. If CRTORD(*SEUTYPE) is specified, compilations of members of different source types will be submitted in an order that will give the maximum opportunity of establishing the relevant dependencies. Dependencies within a source type will not necessarily be successfully re-established.
4. The CPYLIB parameter can be used to preserve existing data when recompiling existing files: specifying a value other than *NONE for the CPYLIB parameter indicates that after a physical file has been successfully compiled, data is to be copied into the new file from any file with the same name that exists in the library specified by the CPYLIB parameter. Copy options *MAP and *DROP will be used when copying the data. The CPYF request is submitted as a separate job to execute after the compilation has run.
5. Items successfully submitted are flagged with an 'S' (*SUBMITTED).

6. Non-native Source Types are as follows:

S/38 source types will be compiled using the appropriate S/38 'Create' commands from library QSYS38.

The following S/36 source types can be compiled using this command:

- CBL36
- DSPF36
- RPG36
- RPT36

7. Library list for compilation

If compiling interactively by specifying JOBD(*NONE), then either LIBLST(*JOB) or LIBLST(*CURRENT) causes the user library list of the job invoking YCRTOBJ to be used when compiling the object(s).

The current library (YCURLIB) used when compiling the object(s), either in batch or interactively, will be the current library of the job invoking YCRTOBJ, unless the LIBLST parameter specifies a Synon/1E library list that either names the current library to use or has the special value *CRTDFT.

Example

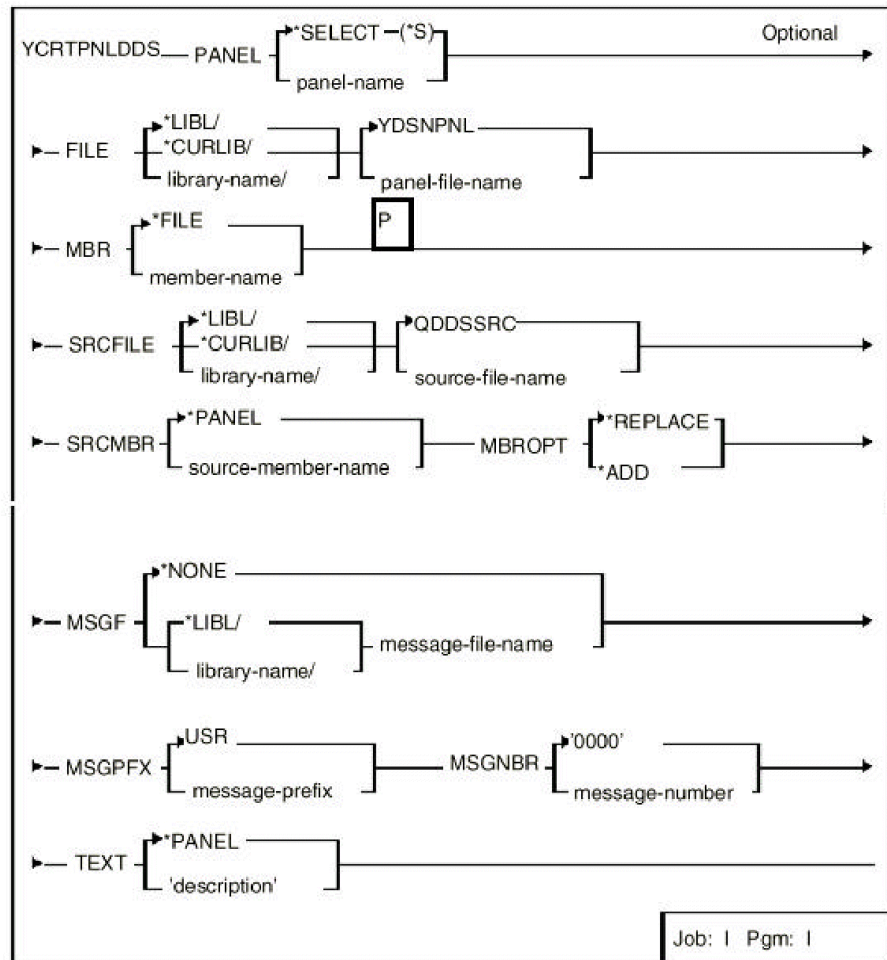
To recompile all physical file source members in FRED/QDDSSRC using job description QBATCH. The new files are to be placed in library CASSANDRA, replacing any existing versions:

```
YCRTOBJ OBJLIB(CASSANDRA) SRCFILE(QDDSSRC) MBR(*ALL)
SRCTYPE(*PF)
```

YCRTPNLDDS (Create DDS From Panel Design)

This command invokes an interactive utility to generate Database Design Specifications (DDS) from a named CA2E panel design.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
PANEL	Name of panel design from which DDS is to be generated	*SELECT: (default) The panel selection display will be presented
FILE	Qualified name of file containing panel design	YDSNPNL: (default) Panel file name

Parameter	Definition	Value and Description
MBR	Name of member in file containing panel design	*FILE: (default) the member has the same name as the file
SRCFILE	Qualified name of source file for DDS source	QDDSSRC: (default) The member is in the i OS default source file in the indicated library. The source file must exist before the Create Panel DDS command can be used
SRCMBR	Name of member to contain generate DDS source	*PANEL: (default) The member has the same name as the panel. The member is created if it does not exist already
MBROPT	Source member update option	<ul style="list-style-type: none"> ■ *REPLACE: (default) the generated DDS source replaces any existing source member. ■ *ADD: the new DDS is appended to any existing member contents
MSGF	Message file name into which message descriptions defining literals are to be placed	*NONE: (default) do not use message descriptions to define literals.
MSGPFX	Message prefix for the message identifiers of messages defining literals. This is used in conjunction with the MSGNBR parameter to generate message identifiers for the messages which are added to the file defined by the MSGF parameter	USR: (default) message prefix
MSGNBR	Starting number for allocating message identifier for messages which are added to message file	0000: (default) start number

Parameter	Definition	Value and Description
TEXT	DDS source member name text	*PANEL: (default) the member text is updated from the panel title.

Notes

This command calls an interactive program to generate DDS for a panel design. Press Help while using the program for information on how to generate the DDS. See the CA2E *Concepts Guide*, Part 3 under Panel Designs for details on panel designs.

Example

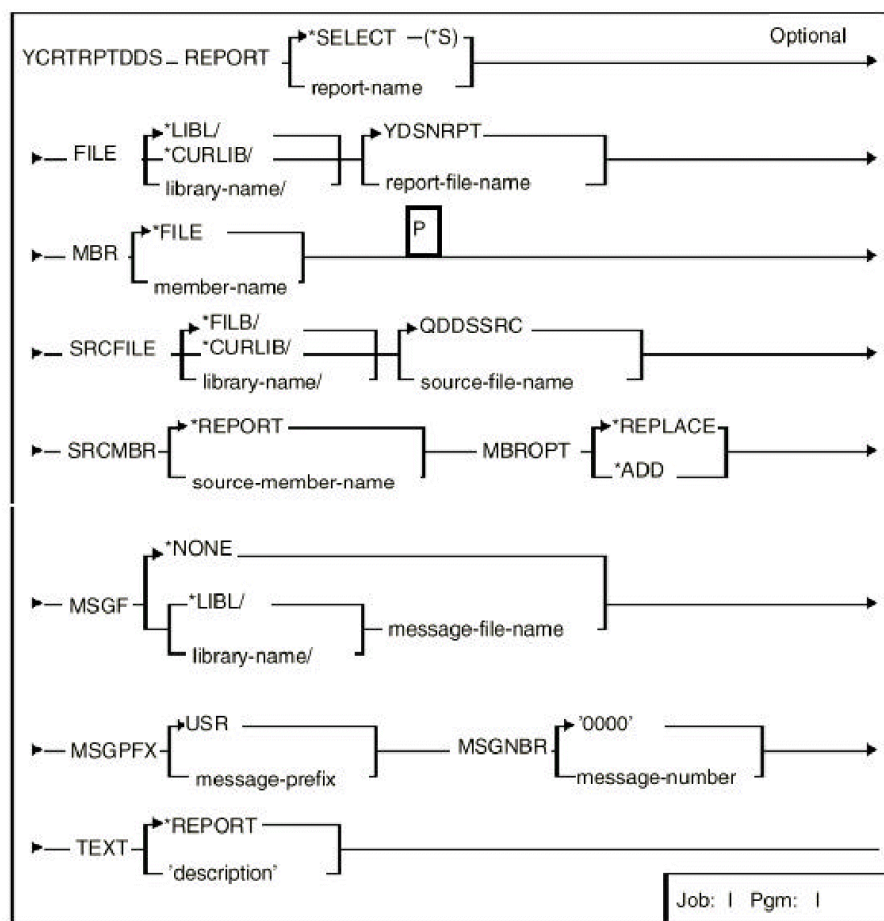
To call the utility to generate DDS for panel SCREAM, placing the generated DDS into member CRY in *LIBL/QDDSSRC:

```
YCRTPNLDDS PANEL(SCREAM) SRCMBR(CRY)
```

YCRTRPTDDS Create DDS From Report Design)

(This command invokes an interactive utility to generate Database Design Specifications (DDS) from a named report design.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
REPORT	Name of report design from which DDS is to be generated	*SELECT: (default) The report selection display will be presented
FILE	Qualified name of file containing report design	YDSNRPT: (default) Report file name
MBR	Name of member in file containing report design	*FILE: (default) The member has the same name as the file

Parameter	Definition	Value and Description
SRCFILE	Qualified name of source file to contain generated DDS source	QDDSSRC: (default) The source member is in the i OS default source file in the indicated library. The source file must exist before the 'Create report DDS' command can be used
SRCMBR	Name of member to contain generated DDS source	*REPORT: (default) The member will have the same name as the report. The member is created if it does not exist already
MBROPT	Source member update option	<ul style="list-style-type: none"> ■ *REPLACE: The generated DDS source will replace any existing source member. ■ *ADD: The new DDS is appended to any existing contents of the source member.
MSGF	Message file name into which message descriptions defining literals are to be placed	*NONE: (default) Do not use message descriptions to define literals
MSGPFX	Message prefix for message identifiers for messages defining literals. This is used in conjunction with the MSGNBR parameter to generate message identifiers for the messages which are added to the file defined by the MSGF parameter	USR: (default) Message prefix
MSGNBR	Starting number for allocating message identifier for messages which are added to message file	0000: (default) Start number
TEXT	DDS source member name text	*REPORT: (default) The member text is updated from the report title

Notes

This command calls an interactive program to generate DDS from a report design. Press Help while using the program for information on how to use this program. See the CA2E *Concepts Guide*, Part 3 under Concepts: Report Designs for details on report design.

Example

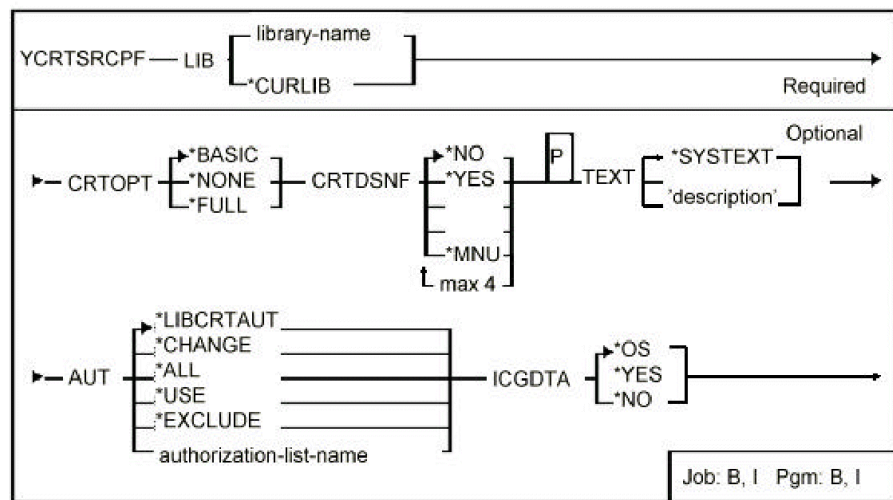
To call the utility to generate DDS for report RUMOUR, placing the generated DDS into member GOSSIP in *LIBL/QDDSSRC:

```
YCRTRPTDDS REPORT(RUMOUR) SRCMBR(GOSSIP)
```

YCRTSRCPF (Create Source Files)

This command creates one of each type of the i OS default source files, to common standards. Intended for setting up a programming environment, which can be used to create design files.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
LIB	Library in which to create source files. Only source files which do not already exist in the library is created	<ul style="list-style-type: none"> ■ *BASIC: (default) Only the following files are created: QCLSRC, QDDSSRC, QCMDSRC, QPNLSRC, QXTSRC, QRPGRS, QCBLSRC QLBLSRC, QPLISRC, QPLICPY. ■ *FULL: The following files area also created: QPASSRC, QBASSRC, QUDSSRC, QFMTSRC, QRMCSRC, QREXSRC, QQMQRYSRC, QQMFORMSRC, QFNTSRC, QCSRC and QCLDSRC. ■ *NONE: do not create source files
CRTOPT	Create option	
CRTDSNF	Create utility design files option	<ul style="list-style-type: none"> ■ *NO: (default) Do not create any design files. ■ *YES: Create files to contain menus, panel designs, and report designs in the named library. ■ *MNU: Create files to contain menus

Parameter	Definition	Value and Description
AUT	Public authority of files	<ul style="list-style-type: none"> ■ *LIBCRTAUT: (default) the user has authority determined by the Create Authority (CRTAUT) prompt of the i OS Create Library (CRTLIB) command or on the i OS Change Library (CHGLIB) command for the library containing the objects to be created. Once the objects are created, their authorities are not affected by changes to the Create Authority (CRTAUT) for the library in which the objects reside. ■ *CHANGE: The user has object operational rights and all data rights to the files. ■ *ALL: Everyone is authorized to do anything to the files, except transfer ownership. ■ *USE: The user has object operational rights and read data rights. ■ *EXCLUDE: Prevents the user from accessing the files. ■ Authorization list name: the name of an authorization list whose authority is used for the files
TEXT	Text to go on each file: up to thirty characters. The text placed on the file is of the form:	<ul style="list-style-type: none"> ■ "RPG source for 'Text '", ■ "CLP source for 'Text '" etc. ■ *SYSTEXT: (default) text is retrieved from data area *LIBL/YYSYTXA

Parameter	Definition	Value and Description
ICGDTA	Specifies whether the file can contain double-byte character set (DBCS) data.	<ul style="list-style-type: none"> ■ *OS: The ICGDTA attribute is specified by the default value on the ICGDTA parameter of the CRTSRCPF command in the installed National Language Version of i OS. ■ *YES: The file may contain DBCS data. ■ *NO: The file may not contain DBCS data

Notes

You must have authority to the i OS command Create Source File (CRTSRCPF) to be able to use this command.

Examples

The following command will create five source files and three sets of design files:

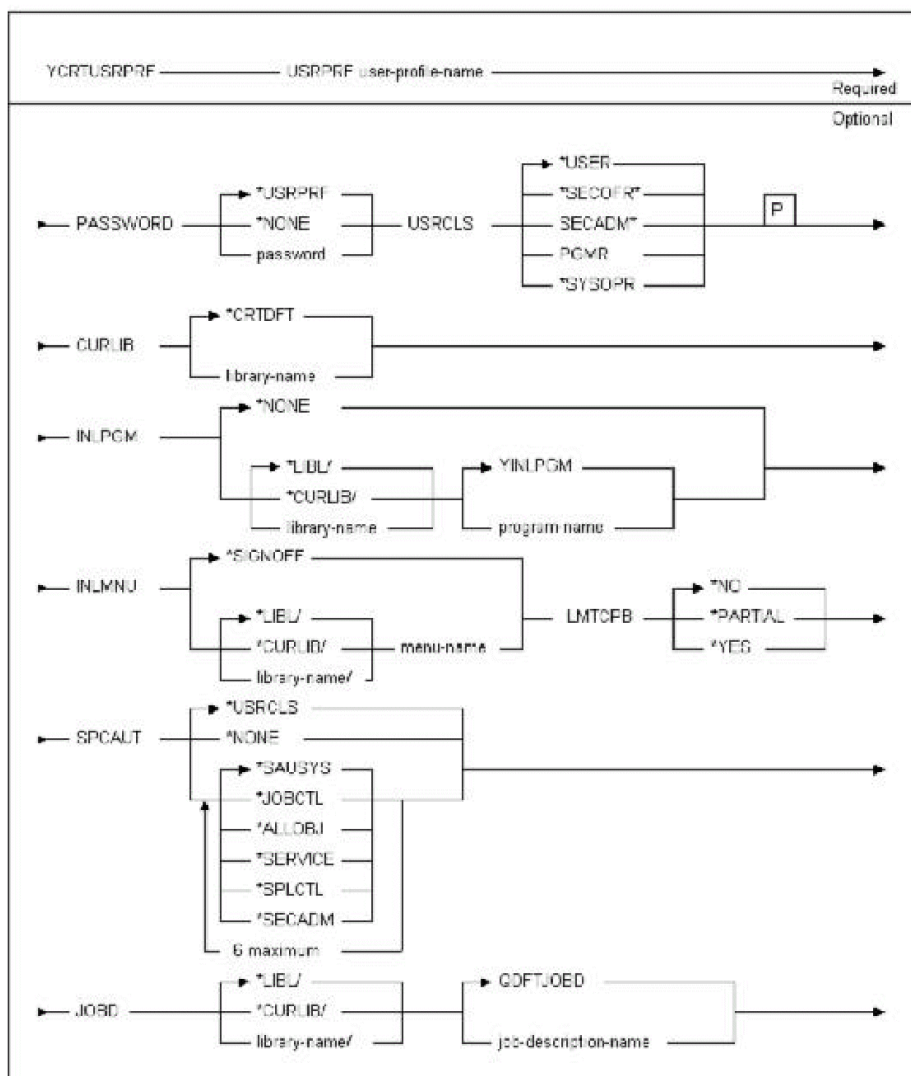
```
YCRTSRCPF LIB(QGPL) TEXT('The Widget System') CRTDSNF(*YES)
```

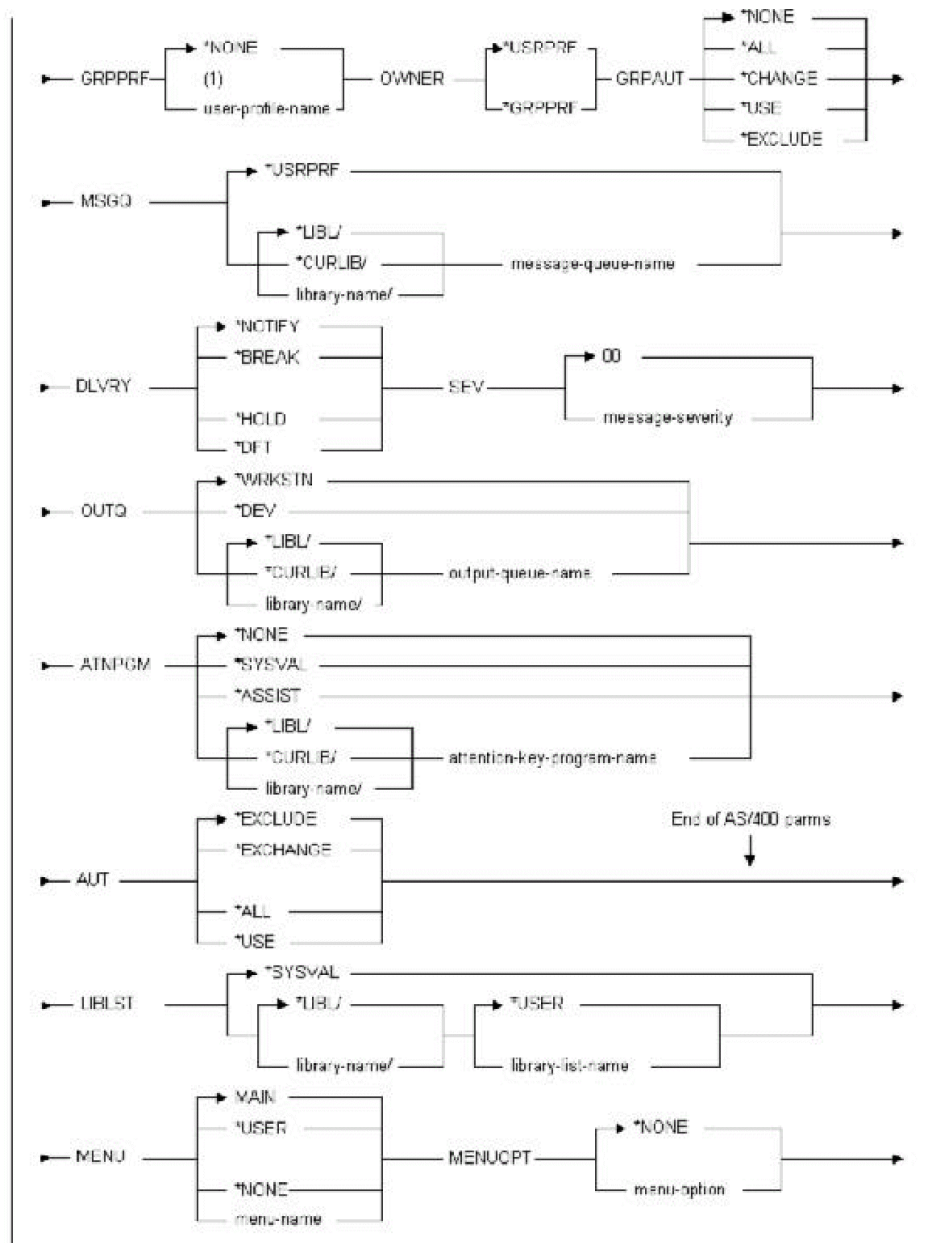
```
QGPL/QCLSRC  TEXT('CL  source for The Widget System')
QGPL/QCMDSRC TEXT('CMD  source for The Widget System')
QGPL/QDDSSRC TEXT('DDS  source for The Widget System')
QGPL/QTXTSRC TEXT('TXT  source for The Widget System')
QGPL/QRPGSRC TEXT('RPG  source for The Widget System')
.....
QGPL/YDSNMNU TEXT('Menus for The Widget System')
QGPL/YDSNPNL TEXT('Panel  designs for The Widget System')
QGPL/YDSNRPT TEXT('Report designs for The Widget System')
```

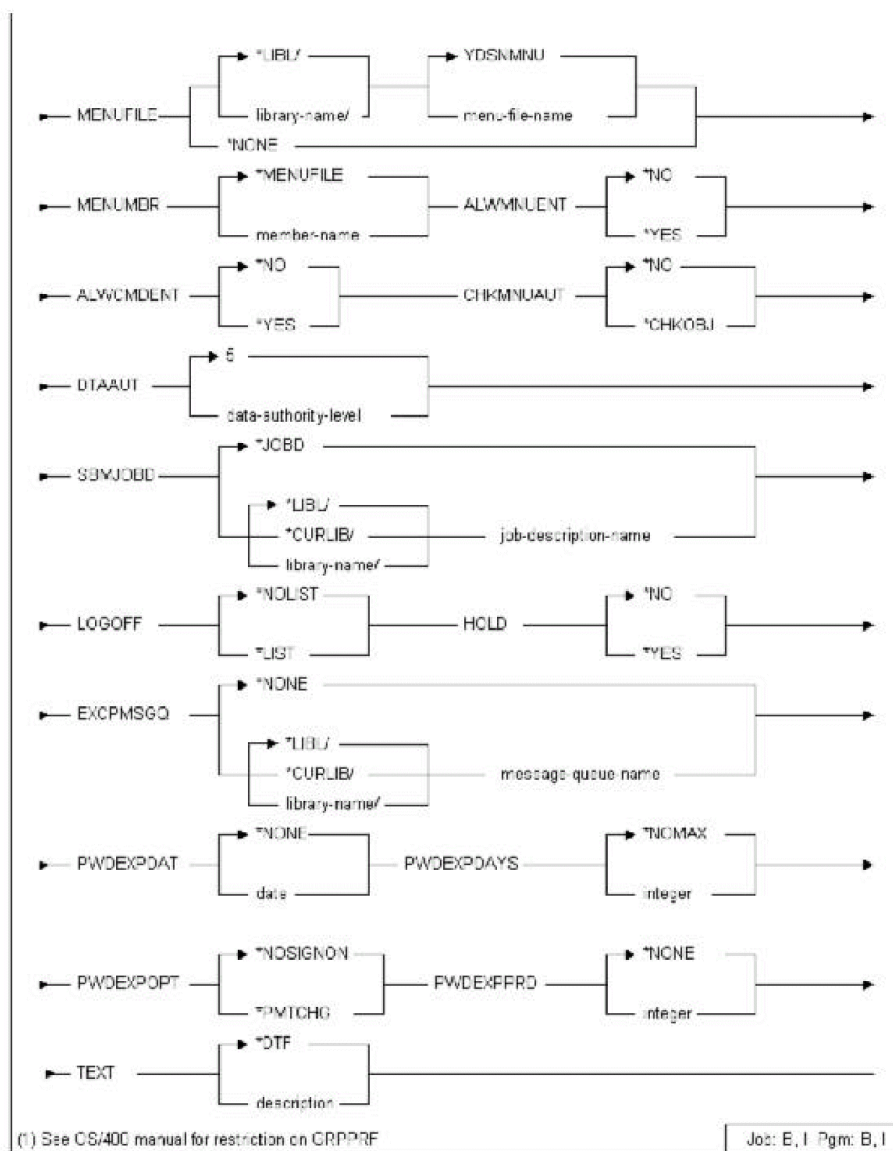
YCRTUSRPRF (Create User Profile)

This command creates a specified user profile. Both the extension profile details and certain i OS user profile details are maintained.

Syntax Diagram







Parameters

Parameter	Definition	Value and Description
USRPRF	User profile name	
PASSWORD	User password. The password must be a valid name	<ul style="list-style-type: none"> ■ *USRPRF: (default) Adopt the profile name as the password. ■ *NONE: Profile has no password
USRCLS	User class. One of the following values	<ul style="list-style-type: none"> ■ *USER: (default) Defined as a user. ■ *SECOFR: User has security officer class. ■ *SECADM: User has administration class. ■ *PGMR: Defined as a programmer. ■ *SYSOPR: User has system operator class
CURLIB	Default current library for this user	*CRTDFT: (default) This user has no default current library. The library QGPL is used as the default create library
INLPGM	Qualified name of initial program for profile. This program is the first program to execute when the user signs on	<ul style="list-style-type: none"> ■ *LIBL/YINLPGM: (default) YINLPGM program is the first program to execute. ■ *NONE: There is no initial program for the profile
INLMNU	Qualified name of initial menu for profile. This menu will be called if there is no initial program, and if an initial program executes a return	*SIGNOFF: (default) Execute a SIGNOFF command if the initial program executes a return
LMTCPB	Limit capability of user to change user profile	<ul style="list-style-type: none"> ■ *NO: (default) Do not limit capability. ■ *YES: Limit capability. ■ *PARTIAL: Partial limitation

Parameter	Definition	Value and Description
SPCAUT	Special authorities. Either a single value	<ul style="list-style-type: none"> ■ *USRCLS: (default) Authorities are defined by user class. ■ *NONE: No special authorities are to be granted. ■ *Or a list of up six special values: <ul style="list-style-type: none"> – *SAVSYS: Grant save system rights. – *JOBCTL: Grant job control rights. – *SECADM: Grant administrator rights. – *ALLOBJ: Grant rights to all objects. – *SERVICE: Grant service rights. – *SPLCTL: grant spool control rights
JOB	Qualified name of initial job description. This job description defines the initial job environment when the user signs on, and is used as the default job description for the user profile	*LIBL/QDFTJOB: (default) The system-supplied job description found in library QGPL is used
GRPPRF	Group profile name	*NONE: (default) Profile is not a group profile member
OWNER	Owner of created objects for group members. If a value of GRPPRF(*NONE) is specified then *USRPRF must be specified	<ul style="list-style-type: none"> ■ *USRPRF: (default) The user profile is to own any objects created by the profile. ■ *GRPPRF: The group profile is to own any created objects

Parameter	Definition	Value and Description
GRPAUT	Authority given to the group profile for newly-created objects for which the user profile is the owner. If OWNER(*GRPPRF) is specified, *NONE is required	<ul style="list-style-type: none"> ■ *NONE: (default) No authority is given to the group profile. ■ *ALL: All authority is given to the group profile. ■ *CHANGE: Change authority for the object type is given to the group profile. ■ *USE: Authority to use the object type is given to the group profile, i.e. read the file, or call the program. ■ *EXCLUDE: User profile is excluded.
MSGQ	Qualified name of message queue associated with the user profile	*USRPRF: (default) Use default message queue with same name as the user profile
DLVRY	Mode of delivery for messages sent to specified message queue	<ul style="list-style-type: none"> ■ *NOTIFY: (default) Messages of the appropriate severity cause the message waiting indicator to be set on the display. ■ *HOLD: Messages are to be held on the message queue. ■ *BREAK: Messages of the appropriate severity are displayed at the time of delivery. The severity level is determined by the value specified for SEV. ■ *DFT: All information messages will be ignored, other messages will be held. Any enquiry messages are given the default reply specified on the job description, or on the message description

Parameter	Definition	Value and Description
SEV	Message severity to use when setting message queue delivery	<ul style="list-style-type: none"> ■ 00: (default) Severity - show all messages. ■ 00-99: message severity. See i OS manual for message severity code meanings
PRTDEV	Name of printer which is to be associated with this user	<ul style="list-style-type: none"> ■ *SYSVAL: (default) The printer device specified by the QPRTDEV system value will be used. ■ *WRKSTN: Printer device associated with the work station
OUTQ	Qualified name of output queue associated with the user profile	<ul style="list-style-type: none"> ■ *DEV: (default) Output is to be directed to the output queue which has the same name as the printer device specified on the PRTDEV parameter. ■ *WRKSTN: Printer device associated with the work station
ATNPGM	Qualified name of attention key program associated with the user profile	<p>*NONE: (default) No attention key program is to be associated with the user profile.</p> <p>If an attention program is specified, the job automatically changes into a group job.</p>
AUT	Authority given to the public for the user profile	<ul style="list-style-type: none"> ■ *EXCLUDE: (default) No authority is given to the public. ■ *ALL: All authorities are given to the public. ■ *CHANGE: Change authority for the profile is given to the public. ■ *USE: Authority to display the profile is given to the public

Parameter	Definition	Value and Description
LIBLST	Qualified name of initial library list. Library list to be set at start of sign-on	*SYSVAL: (default) the system default library list is used
MENU	Name of initial menu for user profile	<ul style="list-style-type: none"> ■ MAIN: (default) Initial menu name. ■ *USER: An initial menu with the same name as that specified in the USRPRF parameter will be used. ■ *NONE: There is no initial menu
MENUOPT	Initial menu option for user profile	<ul style="list-style-type: none"> ■ *NONE: (default) The initial menu is to be displayed. ■ menu-option: The program/command specified by the initial menu option is to be executed
MENUFILE	Qualified menu file name. The menu file in which the initial menu is located. Must be a Synon/1E menu file	*LIBL/YDSNMNU: The default menu file name
MENUMBR	Menu file member name	*MENUFILE: The member name is the name specified in the MENUFILE parameter
ALWMNUENT	Allow menu name entry from the menu display	<ul style="list-style-type: none"> ■ *NO: (default) The user is not permitted to enter menu names. The user is restricted to the displayed menu options. ■ *YES: The user may enter menu names to transfer directly to the named menu

Parameter	Definition	Value and Description
ALWCMDENT	Allow command entry from the menu display	<ul style="list-style-type: none"> ■ *NO: (default) The user is not permitted to enter commands. The user is restricted to the displayed menu options. ■ *YES: The user may enter and execute commands directly from the menu
CHKMNUAUT	Check user authority to menu options while loading menus	<ul style="list-style-type: none"> ■ *NO: (default) When loading a menu, do not check the user's authority to use the program or command named for each menu option. ■ *CHKOBJ: When loading a menu, check the user's authority to use the program or command named for each executable menu option
DTAAUT	Data authority level (1-high to 9-low)	This value is available directly from the file YUSRPRF in the library, or via the YRTVUSRPRF command in a CL program. It may be used to provide a simple form of field level authorization, for examples levels 4 and above may view a salary field. Checking must be provided by user code
SBMJOB	Qualified job description for submitted jobs	<p>This job description is used by the menu display program for menu options having the submit option.</p> <p>*JOB: (default) Use the value specified on the JOB parameter</p>
LOGOFF	Sign off option	<ul style="list-style-type: none"> ■ *NOLIST: (default) No job log is to be created. ■ *LIST: A job log is to be created

Parameter	Definition	Value and Description
HOLD	Hold/release user profile option	<ul style="list-style-type: none"> ■ *NO: (default) The user may sign on. ■ *YES: The user is prevented from signing on
EXCPMSGQ	Qualified name of exception message queue associated with the user profile. A copy of any exception message received by the display menu program is to be sent to this queue	<ul style="list-style-type: none"> ■ *NONE: (default) No exception message is associated with this profile. ■ *USRPRF: Use default message queue with same name as the user profile
PWDEXPDAT	Date upon which password is to expire	*NONE: (default) There is no expiry date
PWDEXPDAYS	Number of days after date of last change that a password is to remain valid	<ul style="list-style-type: none"> ■ *NOMAX: (default) There is no limit to the number of days that the password remains valid. ■ 1-999: Number of days
PWDEXPOPT	Action that YINLPGM is to take if password expiry is detected for a user at sign-on	<ul style="list-style-type: none"> ■ *NOSIGNON: (default) Don't let user sign on. ■ *PMTCHG: Prompt user for new password, providing number of days specified by PWDEXPPRD parameter has not been exceeded
PWDEXPPRD	Number of days after password has expired that user may still be allowed to sign on, providing the changes his password	<ul style="list-style-type: none"> ■ *NONE: (default) The user may not sign on after the password has expired. ■ 1-999: Number of days
TEXT	User profile text	*DFT: The default text is used

Notes

1. See the i OS manuals for further details on the user profile parameters up to AUT. Note that the following i OS user profile parameters are not available on this command: SPCENV, MAXSTG, PTYLMT, AGCCDE, DOCPWD, USROPT, DSPSGNINF, PWDEXPITV and PWDEXP. To use these parameters, use the i OS command Change User Profile (CHGUSRPRF).

See the CA2E *Concepts Guide* for further details on the other parameters (user profile extension attributes), and the Go to Menu (YGO) command diagram.

2. Initial menus versus initial menu options: MNUOPT parameter. Parameters MNUOPT to ALWCMDEXT are used to control the initial menu display for a user.

If a menu option is present, the program or command for the option is executed as the user's initial program.

If a menu option is not present, the menu specified for the option is displayed as the user's initial menu.

3. Initial programs

For a profile to be able make use of the user profile extension attributes at sign-on, the initial program (YINLPGM), or a user modified variant of it, must be specified as the initial program on the INLPGM parameter.

Example

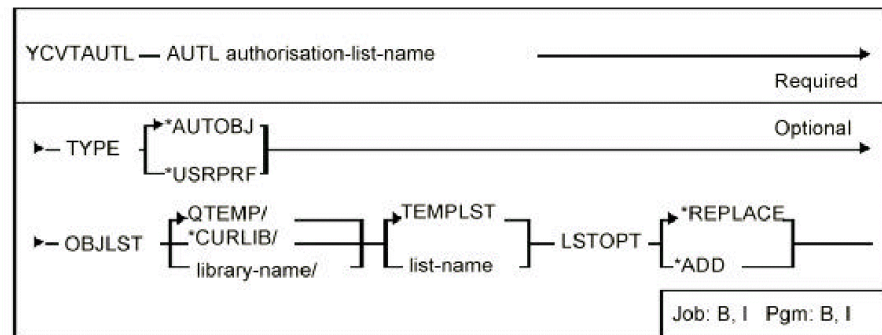
To create a user profile called ROMAN using the initial menu Option 1, in menu file CIVROMMNU:

```
YCRTUSRPRF USRPRF(ROMAN) TEXT('Roman profile') MENU(CIVROMMNU) MNUOPT(1)
```

YCVTAUTL (Convert Authorization List)

This command creates an object list of all the objects belonging to an Authorization List: either of all objects to which the list controls authorization, or all profiles which are included in the list.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
AUTL	Name of authorization list	
TYPE	Type of objects to be included in the object list	<ul style="list-style-type: none"> ■ *AUTOBJ: (default) Include objects to which authorization list controls authorization. ■ *USRPRF: Include user profiles contained in the authorization list
OBJLST	Qualified name of object list that is to be built	QTEMP/TEMPLST: (default) List name
LSTOPT	List replacement option	<ul style="list-style-type: none"> ■ *REPLACE: (default) Create a new list replacing any previous list's contents. ■ *ADD: Add to any existing list's contents

Notes

1. Restrictions are as follows:

The list entries in the resulting object list do not contain complete object information: only the object data shown in the table below is provided. The command makes use of the information from the outfiles provided by the DSPAUTL command.

List entry information added by YCVTAUTL command is as follows:

OBJLST	Field name	QUDALO	Field	QAOBJAUT
ODDCEN	Creation century	AODCEN		OADCEN
ODDDAT	Creation date	AODDAT		OADDAT
ODDTIM	Creation time	AODTIM		OADTIM
ODOBNM	Object name	AONAME		OAUSR
ODLBNM	Object library	AOLIB		'QSYS'
ODOBTP	Object type	AOTYPE		'*USRPRF'
ODOBAT	Object attribute	-		-
ODOBOW	Object owner	AOOWN		-
ODOBTX	Object text	AOTXT		-
ODSRCF	Object source	AOANAM	AUTL name	OAANAM
ODSRCL	file	AOALIB	AUTL lib	OALIB
ODSRCM	Object sour lib	AOAOWN	AUTL owner	-
	Object source member			

If, using the command Filter Object list (YFLTOBJLST), you attempt to filter the resultant object list on attributes other than those shown above (such as object size) unpredictable results may occur.

- Note that if you are converting an authorization list into a list of user profiles, *PUBLIC is not included in the resulting object list.

Examples

To build a list of all objects secured by authorization list FRED:

```
YCVTAUTL AUTL(FRED)
```

The list will be called TEMPLST and reside in library QTEMP.

YCVTBIN (Convert Binary To Decimal)

This command converts a binary number to the decimal equivalent. For use in CL programs.

Syntax Diagram

YCVTBIN	—	BIN binary-number	—	DEC CL-variable-name	Required
					Pgm: B, I

Parameters

Parameter	Definition	Value and Description
BIN	Binary number to be converted; 2 bytes long	
DEC	Decimal equivalent returned by command. Must be a five digit numeric packed CL variable	

Notes

This command is only valid in a CL program.

Example

To get the number of libraries present in a library list passed by a command.

```
PGM (&LIBL)

DCL VAR(&LIBL) TYPE(*CHAR) LEN(252) /*List: 2B + 25 x10*/

DCL VAR(&BIN) TYPE(*CHAR) LEN(2) /*Binary value*/

DCL VAR(&DEC) TYPE(*DEC) LEN(5 0) /*Decimal value*/

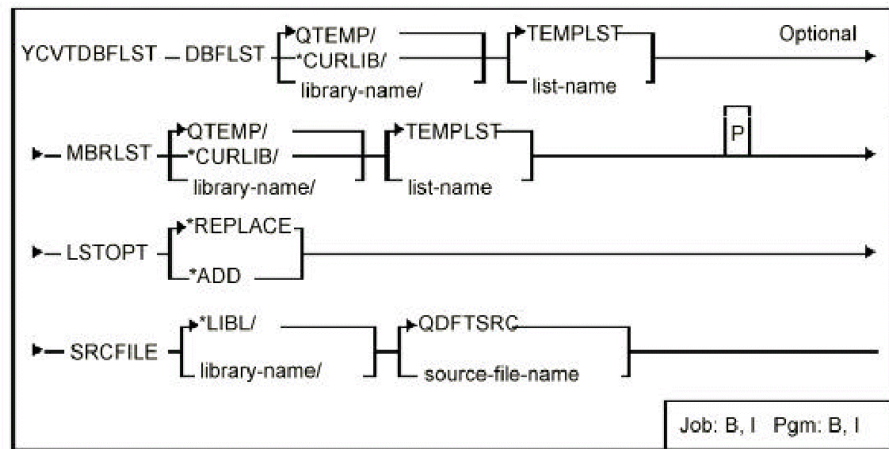
CHGVAR VAR(&BIN) VALUE(%SST(&LIBL 1 2))

YCVTBIN BIN(&BIN) DEC(&DEC)
```

YCVTDBFLST (Convert DBF List To Member List)

This command converts an existing database file list to a member list. The file name is converted to a source file member name.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
DBFLST	Qualified name of database file list which is to be converted	QTEMP/TEMPLST: (default) list name
MBRLST	Qualified name of member list to hold converted source member output	QTEMP/TEMPLST: (default) list name.
LSTOPT	List replacement option	<ul style="list-style-type: none"> ■ *REPLACE: (default) Create new list, replacing any previous list contents. ■ *ADD: add to any existing list contents
SRCFILE	Source file options	QDFTSRC: (default) Use QDDSSRC. Nominated source file and library

Notes

1. The entries in the resulting member list will not contain the following information:

- MLNOMB Number of members in file.
- MLNRCD Number of active records.
- MLNDTR Number of deleted records.
- MLSIZE Current size of member.
- MLCCEN Creation century.
- MLCDAT Creation date.
- MLCHGC Last changed century.
- MLCHGD Last changed date.
- MLCHGT Last changed time.

If, using the command Filter Member list (YFLTMBRLST), you attempt to filter the resultant member list on any of the above attributes, unpredictable results may occur.

2. Files not derived from source members are ignored by this command.

Example

Say we have a database file list HERMES containing the following two items:

Object	Type
OLYMPIA/CADUCEUS	*FILE
OLYMPIA/PYX	*FILE

The following command converts the database file list into a new member list APHRODITE:

```
YCVTDBFLST DBFLST(HERMES) MBRLST(APHRODITE) SRCFILE(DELPHI/QDDSSRC)
```

The new member list, APHRODITE, would contain the following two items:

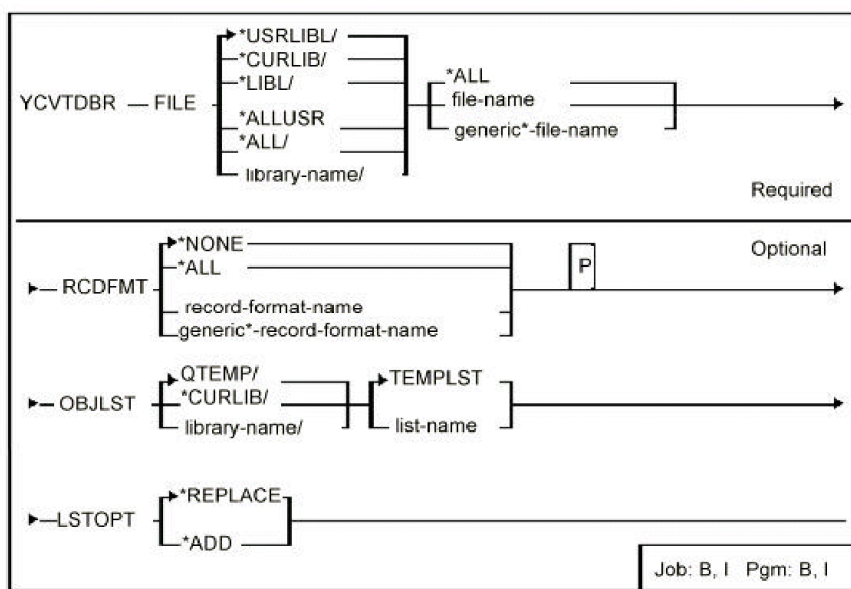
File	Member	SEU type
DELPHI/QDDSSRC	CADUCEUS	PF
DELPHI/QDDSSRC	PYX	PF

YCVTDBR (Convert Database Relations)

This command creates an object list from the output of the `i OS` command Display database relations (DSPDBR).

Enables you to obtain a list of all dependent files on a physical file or files; this can then be used to recompile the files via the Create Object command (**YCRTOBJ**).

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
FILE	Qualified generic file name of file for which database relations are to be determined	*ALL: (default) Include all files
RCDFMT	Generic record format name	<ul style="list-style-type: none"> ■ *NONE: (default) No specific record format is to be located. ■ *ALL: All record format usage is to be determined. <p>If a name is entered the files containing the specified format(s) are included</p>
OBJLST	Qualified name of object list that is to be built	QTEMP/TEMPLST: (default) List name
LSTOPT	List replacement option	<ul style="list-style-type: none"> ■ *REPLACE: (default) Create a new list replacing any previous list's contents. ■ *ADD: Add to any existing list's contents

Notes

1. Restrictions are as follows:

The list entries in the resulting object list do not contain complete object information. Only the following object data is provided (default values are shown underlined):

- (ODOBNM) Object name.
- (ODOBLB) Object library
- (ODOBTP) Object type - *FILE
- (ODOBAT) Object attribute - PHY or LGL
- (ODSRCF) Source file name - QDDSSRC
- (ODSRCL) Source file library - *LIBL

- (ODSRCM) Source member - Object name

When using the command Filter Object list (YFLT OBJLST) and attempting to filter the resultant object list on attributes, other than those shown above, such as object size, unpredictable results can occur.

Examples

To build a list of all files dependent on MYFILE in library MYLIB:

```
YCVTDBR FILE(MYLIB/MYFILE)
```

The list is called TEMPLST and reside in QTEMP.

To build a list of all files having record formats beginning with GL in MYLIB:

```
YCVTDBR FILE(MYFILE/*ALL) RCDFMT(GL*) LSTOPT(*ADD)
```

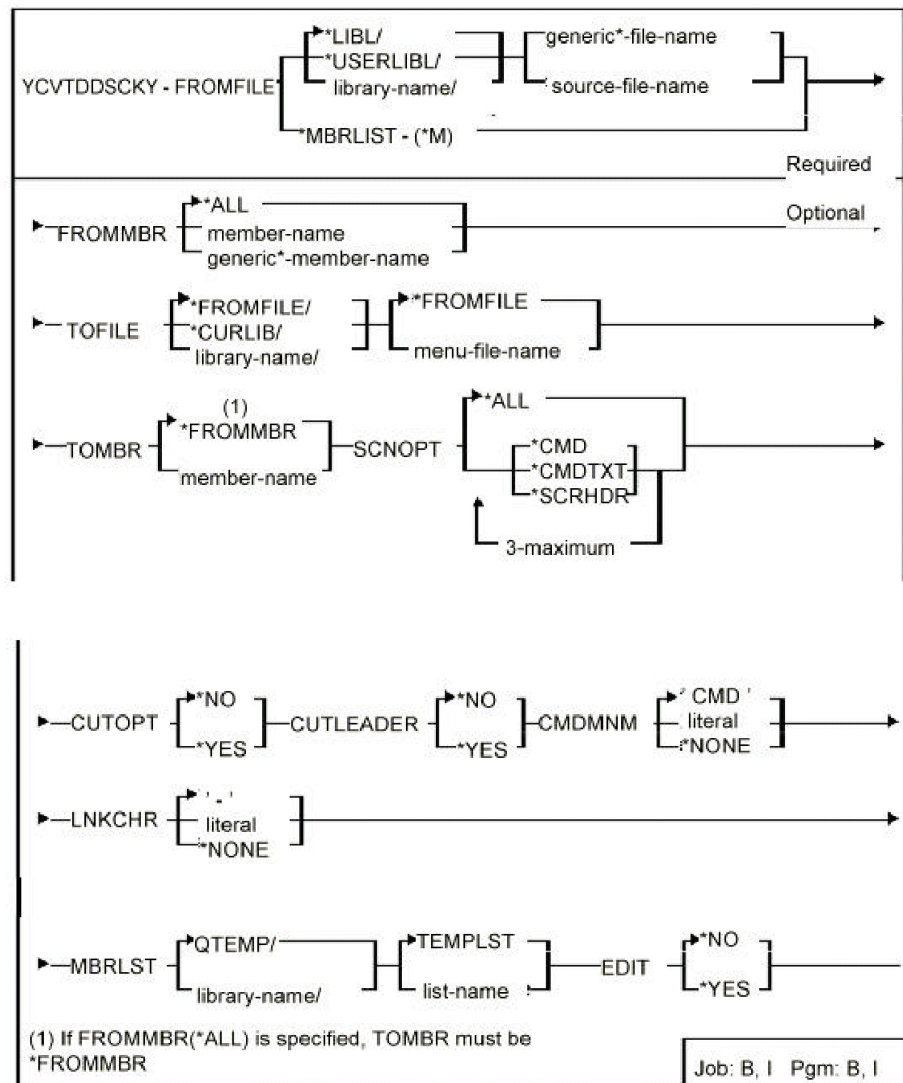
The list is called TEMPLST and reside in library QTEMP. The output is added to the previous list contents.

YCVTDDSKY (Convert DDS Command Keys)

This command analyses/converts a display file DDS source and can be used to modify DDS source to make panel designs conform to CUA standards. The command optionally converts:

- Command keys
- Literals providing command key explanations
- Text leaders within text literals
- Command key table
- A report is produced of all changes made

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
FROMFILE	Qualified generic name of source files containing converted DDS members	*MBRLST: (default) Use the member list specified by the MBRLST parameter
FROMMBR	Generic member name of converted DDS	*ALL: (default) Include all members

Parameter	Definition	Value and Description
TOFILE	Qualified name of source file into which the converted source members are to be placed	*FROMFILE: (default) Name is the same as FROMFILE
TOMBR	Member name of converted source	*FROMMBR: (default) Name is same as FROMMBR
SCNOPT	Analysis option; specifies those components of the DDS source for which the conversion program is to look	<ul style="list-style-type: none"> ■ *ALL: (default) Analyze source for use of command keys, command explanation literals and screen headers ■ *CMD: Scan for use of command keys ■ *CMDTXT: Scan for use of command explanation text ■ *SCRHDR: Scan for use of screen headers. The DDS for the first three lines of each display are shown on the report. This helps to identify the other items shown on the report. No conversion of the DDS for display headers are attempted
CVTOPT	Conversion option	<ul style="list-style-type: none"> ■ *NO: (default) Only analyze source and produce a report, do not convert source ■ *YES: Convert source
CVTLEADER	Indicates whether text leader literals are to be converted	<ul style="list-style-type: none"> ■ *NO: (default) Do not convert text leaders ■ *YES: Convert text literals ending in `.:`; the program changes these to conform to the SAA standard, that is have spaces between periods leading up to the colon, i.e. `(. . .:)`

Parameter	Definition	Value and Description
CMDMNM	Indicates whether the specified command mnemonic is to be removed when converting command explanation literals	<ul style="list-style-type: none"> ■ 'CMD:': (default) Remove indicated value if found in command key explanation text ■ *NONE: Do not remove any leading mnemonics on command key explanation text line
LNKCHR	Indicates whether the program is to look for a linking character when scanning for command key explanations, and if so, for what character it should search. This may be necessary to distinguish between a command key reference and the use of a numeric literal in the text	'-': Look for a hyphen within command explanation text. The value is replaced by an '=' character *NONE: Do not attach a linking character when scanning the command text line
MBRLST	Qualified name of a member list	QTEMP/TEMPLST: (default) List name
EDIT	List option	<ul style="list-style-type: none"> ■ *NO: (default) No editing is required. ■ *YES: Invoke the edit member list function to edit the list before executing the scan

Notes

1. The purpose of the command is to assist in the conversion of display files so that they conform to SAA CUA conventions. In particular the command helps the re-mapping of command keys.
2. Command key re-mapping is achieved by changing the assignments of command/function keys to HLL indicators, but still leaving the response indicator unchanged. The program source does not need modification, as the same response indicator is returned from the display file. It does not even necessary to recompile the program, as the format level is unchanged.

Comments are inserted by the utility into the converted source to indicate the changes, which have been made. This example displays DDS source before modification:

```

A          CA01<01 'Exit Program.'>
A          CF03<03 'Define.'>
A          CF05<05 'Re-load.'>
A          CF07<07 'Field Ref display'>
A          CF10<10 'field definitions'>

```

This example displays DDS source after modification:

```

/*M: FRED 02/06/88 Command key conversion from 01 to 03.
A          CA03<01 'Exit Program.'>
/*M: FRED 02/06/88 Command key conversion from 03 to 10.
A          CF10<03 'Define.'>
A          CF05<05 'Re-load.'>
A          CF07<07 'Field Ref display'>
/*M: FRED 02/06/88 Mandatory conversion from 10 to 04.
A          CF04<10 'Field definitions'>

```

The character specified by the LNKCHR parameter is used to identify command key explanation text. The program checks for command text from lines 20 to 27.

The modified command key explanation text is sorted into the revised command key order.

This example displays DDS source before modification:

```

A N40          Z3 Z'CMD: 1-Exit, 3-Define object, 5-Re-
A              load, 7-Fields, 9-Insert. -
A

```

This example displays DDS source after modification

```

/*M: FRED 02/06/88 command key conversion from 01 to F3.
/*M: FRED 02/06/88 Command key conversion from 03 to F10.
/*M: FRED 02/06/88 Command key conversion from 05 to F5.
/*M: FRED 02/06/88 Command key conversion from 07 to F7.
A          Z3 Z'F3=Exit F5=Reload F7=Fields F9=-
A              Insert F10=Define object -
A

```

3. A report is produced of all modifications which have been made or, if CVTOPT(*NONE) is specified, which might be made. The source line numbers shown refer to the originating source member.

Example

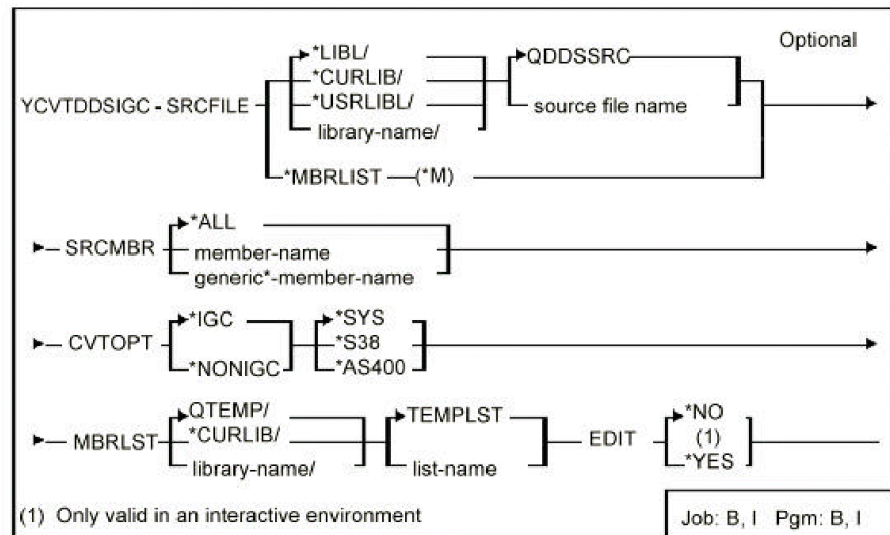
To convert all display files DDS in file QDDSSRC in library DEV38, converting screen leaders, removing a command mnemonic, and using a linking character:

```
YCVTDDSCY FROMFILE(DEV38/QDDSSRC) FROMMBR(*ALL) TOFILE(DEV400/QDDSSRC)
SCNOPT(*ALL) CVTOPT(*YES) CVTLEADER(*YES)
```

YCVTDDSIGC (Convert DDS To IGC)

This command converts DDS source from one version to another, according to directives coded in columns 1-5 of the source. The conversion can be used to change DDS source so that it is suitable for use on an IGC (ideographic) machine or for use on the IBM i to one suitable for use on the System/38, and vice versa.

Syntax Diagram

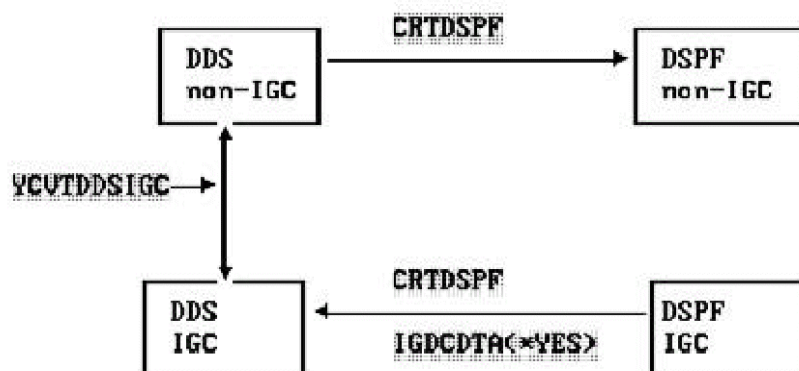


Parameters

Parameter	Definition	Value and Description
SRCFILE	Qualified generic name of file containing converted source members	<ul style="list-style-type: none"> ■ QDDSSRC: (default) Source file name. ■ *MBRLST: Convert members named in member list specified by the MBRLST parameter
SRCMBR	Generic name of the converted members	*ALL: (default) Convert all members in the specified file or files
CVTOPT	Specifies the conversion option to use	<p>List parameter made up of two elements:</p> <ul style="list-style-type: none"> ■ *IGC type ■ *IGC: (default) Convert DDS to IGC ■ *NONIGC: Converts DDS back to non-IGC <p>Machine type:</p> <ul style="list-style-type: none"> ■ *SYS: (default) Convert for current machine ■ *S38: Convert for System/38 ■ *AS400: Convert for IBM i
MBRLST	Qualified name of member list indicating the new converted members	<p>QTEMP/TEMPLST: (default) List name</p> <p>Specify FROMFILE(*MBRLST) to use an existing member list</p>
EDIT	Edit list required	<ul style="list-style-type: none"> ■ *NO: (default) No editing required ■ *YES: The edit member list function is invoked to edit the list before execution

Notes

1. This command makes it possible to use the same DDS source to build a system on both IGC and non-IGC machines. The command automatically makes the necessary changes to the source. The conversion is directed by extra information coded in columns 1-5 in the DDS source. Lines only applicable to the IGC version are coded as comments. The conversion is reversible: source may be converted both to and from IGC format:



2. Before you use this command you will need to 'mark up' the source with additional information: this is entered as comments and/or extra characters in columns 1-5 of the DDS statements.
 - Lines which are required only for one version of the DDS.
Some DDS keywords are only applicable to one particular version. For example, keywords such as LOWER, CHECK(FE) and DUP are not available on IGC machines, while the IGCCNV keyword is only applicable to IGC machines. Further, a different function key activates the IGC conversion on the System/38 and the lines containing keywords not required in the base version of the DDS should be included as comments in the source. The YCVTDDSIGC command will remove or add the '*' in column 7 as appropriate.
 - Lines which require modification on conversion from one version to another.
Some lines of the DDS will require modification for use on IGC machines, for example by the addition of a keyboard shift (E,O or J) to allow the entry of ideographic characters in a field. The YCVTDDSIGC command will modify such lines according to the directives entered in columns 1-5 of the source line, see below.
 - Directives for conversion in columns 1-5.
All statements to be examined by the YCVTDDSIGC command have a colon ':' in Column 2. Column 1 indicates the line type.

The following is an example of DDS source marked up for YCVTDDSIGC:


```

.....AAN01N02N03T.Name+++++RLen++RDpBLinPosFunctions+++++
:
: *=====*
: *:K A* IGCNU<CF18 18>
: *=====*
: 0:N A OBJ R B 5 9
: *:N A LOWER
: CHECK<FE>
: DUP<45>
: DSPATR<UL>
: DSPATR<HI>
: COLOR<TRQ>
: DSPATR<RI PC>
: Z2 1MSGCON<78 VVY000Z VMSG>
: Z2 1MSGCON<78 VVY110Z VMSG>
:

```

```

.....AAN01N02N03T.Name+++++RLen++RDpBLin PosFunctions+++++
:
: *=====*
: *:K A IGCNU<CF18 18>
: *=====*
: :K A OBJ R 0 B 5 9
: *:N A* LOWER
: *:N A* CHECK<FE>
: *:N A* 44 DUP<45>
: A N31N42 DSPATR<UL>
: A N31 DSPATR<HI>
: *:N A*N31 COLOR<TRQ>
: A 31 DSPATR<RI PC>
: *:N A* Z2 1MSGCON<78 VVY000Z VMSG>
: *:K A Z2 1MSGCON<78 VVY110Z VMSG>
:

```

A type of IGC/non-IGC specific is illustrated in the following examples. If column 1 contains an asterisk '*', then the line only applies to an IGC or a non-IGC version, and it will be made into a comment when the source is converted to the version to which it does not apply. For example:

```

.....AAN01N02N03T.Name+++++RLen++RDpBLinPosFunctions+++++
: *K A IGCNU(CF18 18)
:
: *N A LOWER
:
:
      YCVTDDSIGC CUTOPT(*IGC)      YCVTDDSIGC CUTOPT(*NONIGC)
      ↓                               ↑
.....AAN01N02N03T.Name+++++RLen++RDpBLinPosFunctions+++++
: *K A IGCNU(CF18 18)
:
: *N A LOWER
:
:

```

Note that when commenting out a statement, the YCVTDDSIGC command will preserve any existing value from column 7 by placing it in column 4. For example:

```

.....AAN01N02N03T.Name+++++RLen++RDpBLinPosFunctions+++++
: *N A 33N34N48
: *N A0 32 COLOR(WHT)
:
      YCVTDDSIGC CUTOPT(*IGC)      YCVTDDSIGC CUTOPT(*NONIGC)
      ↓                               ↑
.....AAN01N02N03T.Name+++++RLen++RDpBLinPosFunctions+++++
: *N A* 33N34N48
: *N0 A* 32 COLOR(WHT)
:
:

```

```

.....AAN01N02N03T.Name+++++RLen++RDpBLinPosFunctions+++++
: *K 1A* IGCNU(CF18 18)
: *K 0A IGCNU(CF24 24)
:
: *K 1A* MSGCON(78 YVY2207 YMSG):
: *K 0A MSGCON(78 YVY1107 YMSG):
:
VCUTDDSIGC CUTOPT(*IGC *AS400) VCUTDDSIGC CUTOPT(*IGC *S38)
.....AAN01N02N03T.Name+++++RLen++RDpBLinPosFunctions+++++
: *K 1A IGCNU(CF18 18)
: *K 0A IGCNU(CF24 24)
:
: *K 1A MSGCON(78 YVY2207 YMSG):
: *K 0A MSGCON(78 YVY1107 YMSG):
:

.....AAN01N02N03T.Name+++++RLen++RDpBLinPosFunctions+++++
: O:N A AFIELD 25 1 12
:
VCUTDDSIGC CUTOPT(*IGC) VCUTDDSIGC CUTOPT(*NONIGC)
.....AAN01N02N03T.Name+++++RLen++RDpBLinPosFunctions+++++
: K A AFIELD 250 1 12
:

```

Example

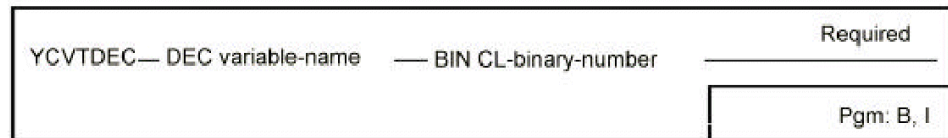
To convert all source members in source file QDDSSRC in library FRED to ideographic standards:

```
YCVTDDSIGC FILE(FRED/QDDSSRC) CVTOPT(*IGC)
```

YCVTDEC (Convert Decimal To Binary)

This command converts a decimal number to the binary equivalent.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
DEC	Decimal number to be converted. Maximum size of five digits, no decimal places	
BIN	Binary equivalent returned by command 2 bytes CL character variable	

Notes

This command is only valid in a CL program.

Example

To change the number of entries in a list to the length of the list:

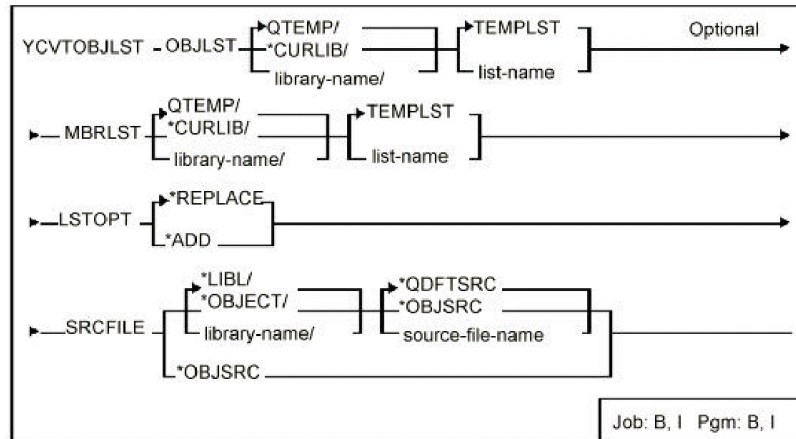
```
PGM (&LIBL)
DCL VAR(&LIBL) TYPE(*CHAR) LEN(252)  /*List: 2B + 25 x10*/
DCL VAR(&BIN)  TYPE(*CHAR) LEN(2)<P8%-2> /*Binary value*/
DCL VAR(&DEC)  TYPE(*DEC)  LEN(5 0)  /*Decimal value*/<B%-2>
CHGVAR VAR(&BIN) VALUE(%SST(&LIBL 1 2))
YCVTBIN BIN(&BIN) DEC(&DEC)
CHGVAR VAR(&DEC) VALUE(10 * &DEC)
```

```
YCVTDEC DEC(&DEC) BIN(&BIN)
CHGVAR VAR(%SST(&LIBL 1 2)) VALUE(&BIN)
```

YCVTOBJLST (Convert Object List to Member List)

This command converts an existing object list into a member list. The object name is converted to a source file member name.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
OBJLST	Qualified name of object list which is to be converted into a member list	QTEMP/TEMPLST: (default) List name
MBRLST	Qualified name of member list to hold converted source member output	QTEMP/TEMPLST: (default) List name
LSTOPT	List replacement option	<ul style="list-style-type: none"> ■ *REPLACE: (default) Create new list, replacing any previous list's contents ■ *ADD: Add to any existing list's contents

Parameter	Definition	Value and Description
SRCFILE	Qualified name of source file for members	<ul style="list-style-type: none">■ *QDFTSRC: (default) Use object type and attribute to assign i OS default source file name for object (*LIBL can be used for this option)■ *OBJSRC: Use source file and library held in object description

Notes

1. The entries in the resulting member list will not contain the following information:
 - MLNOMB Number of members in source file
 - MLNRCD Number of active records
 - MLNDTR Number of deleted records
 - MLSIZE Current size of member
 - MLCCEN Creation century
 - MLCDAT Creation date
 - MLCHGC Last changed century
 - MLCHGD Last changed date
 - MLCHGT Last changed time

When using the command Filter Member List (YFLTMBRLST) and attempting to filter the resultant member list, unpredictable results can occur.
2. Objects not derived from source members are ignored by this command.
3. The member list can have the same name as the object list.

Example

Say that an object list HERMES contains the following two items:

```

HERMES : .....
       : Object      Type  Attribute Compiled from :
       : -----
       : OLYMPIA/CADUCEUS *FILE  DSPF  PELION/QDDSSRC CADUCEUS :
       : OLYMPIA/PYX      *PGM   RPG   OSSIAN/QRPGSRC PYX   :
       : SPARTA/AMPHORA   *PGM   CLP   OSSIAN/QCLSRC AMPHORA :
       : .....

```

The following command would convert the object list into a new member list APHRODITE:

```
YCVTOBJLST OBJLST(HERMES) MBRLST(APHRODITE) SRCFILE(*OBJSRC)
```

```

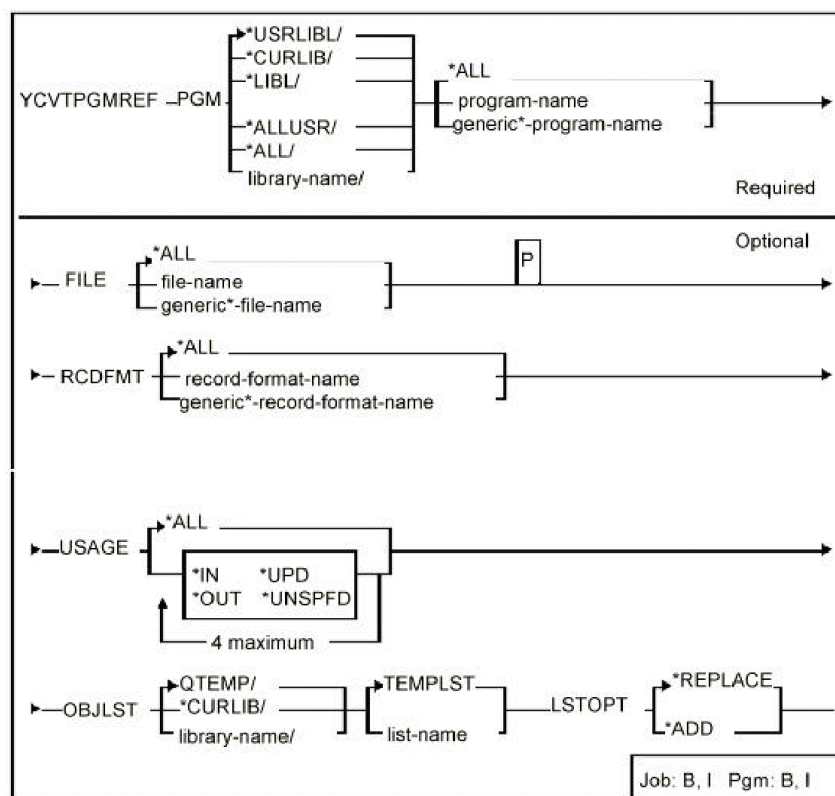
APHRODITE : .....
          : Source file      Member      Source type :
          : -----
          : PELION/QDDSSRC  CADUCEUS      DSPF      :
          : OSSIAN/QDDSSRC  PYX           RPG       :
          : OSSIAN/QCLSRC   AMPHORA       CLP       :
          : .....

```

YCVTPGMREF (Convert Program References)

This command creates an object list of the programs references and the list can be converted to a member list (by the YCVTOBJLST command), which may then be used to recompile the programs via the command Create Object (YCRTOBJ).

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
PGM	Qualified generic name of programs for which references are to be processed	*ALL: (default) Include all programs
FILE	Generic file name of referenced files. Only those programs which reference the specified file(s) are selected	*ALL: (default) Do not select on file name

Parameter	Definition	Value and Description
RCDFMT	Generic record format name of referenced formats. Only those programs which reference the specified record format(s) are selected	*ALL: (default) Do not select on record format
USAGE	File usage types. Only those programs which access the given file(s)/format(s) in the specified way are selected	Single value <ul style="list-style-type: none"> ■ *ALL: (default) Do not select on file usage ■ *Or list of up to four elements <ul style="list-style-type: none"> – *IN: Usage is read only – *OUT: Usage is output only – *UPD: Usage is update – *UNSPFD: Usage is unspecified
OBJLST	Qualified name of object list that is to be built	QTEMP/TEMPLST: (default) List name
LSTOPT	List replacement option	<ul style="list-style-type: none"> ■ *REPLACE: Create a new list, replacing any previous list's contents ■ *ADD: Add to any existing list's contents

Notes

None

Example

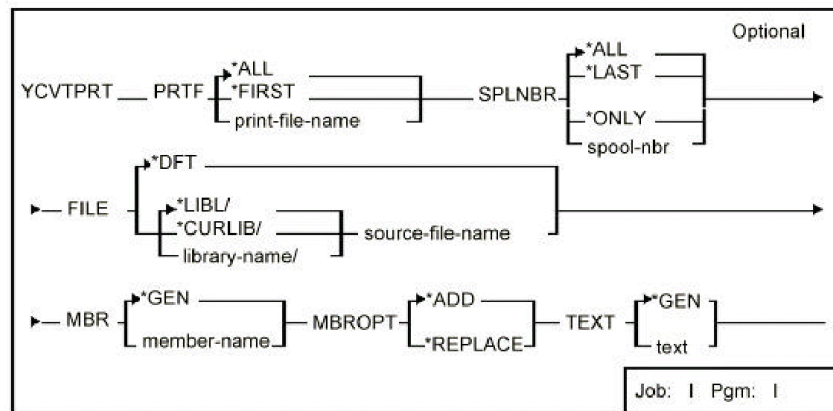
To build a list called TEMPLST in library QTEMP of all programs in library GLPGM which reference record format @GLPOST for output or update:

```
YCVTPGMREF PGM(GLPGM/*ALL) RCDFMT(@GLPOST) USAGE(*OUT *UPD)
```

YCVTPRT (Convert Print Output To Source Member)

This command converts spooled print file output into a dbf source member. Output is 'framed' to look like a screen image, suitable for inclusion as an illustration in your user instruction manuals.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
PRTF	Name of print file whose output is to be converted to a source member	<ul style="list-style-type: none"> ■ *ALL: (default) Convert the output of any print files previously named by the YSTRCVTPRT command ■ *FIRST: Convert the output of the first print file previously named by the YSTRCVTPRT command ■ Print file name: Convert the output of the named file. (File must have been previously named by the YSTRCVTPRT command.)

Parameter	Definition	Value and Description
SPLNBR	Spool number of print output which is to be converted. Output must have been created by current job subsequent to using the YSTRCVTPRT receive converted output	<ul style="list-style-type: none"> ■ *ALL: (default) Convert all output ■ *LAST: Convert the most recent spooled output ■ *ONLY: Convert the output only if there is only one unprocessed spooled file present ■ Spool file nbr: Convert the output for the numbered file
FILE	Name of source file containing member that is to receive converted print output	*DFT: (default) Use source file nominated by the YSTRCVTPRT command
MBR	Name of source member that is to receive converted print output	*GEN: (default) Generate a member name as follows: "PP + Job number + Spool number," for example: P010014002
MBROPT	Member update option	<ul style="list-style-type: none"> ■ *ADD: (default) Add to any existing member having the name specified by the MBR parameter ■ *REPLACE: Replace any existing member having the name specified by the MBR parameter
TEXT	Member text	*GEN: (default) Generate descriptive text as follows: "Print key for' PRTF + JOBNBR/USER/JOB SPLNBR"

Notes

1. The command Start Print Conversion Mode (YSTRCVTPRT) must have been invoked prior to using this command, and before creating the print output that is to be converted.
2. The print output is cancelled after successful conversion.
3. To end print key conversion, return to the invocation level at which you called the Start Print Conversion Mode command.

4. The characters used to form the frame of the screen illustration are retrieved from a data area YPBXCHA. You may modify the characters used by changing the contents of this data area. The data area is eight characters long and of type character, and its contents are as follows:

Character	Pos	Shipped Default	Hex
Left hand side	1	Vertical bar	4F
Right hand side	2	Vertical bar	4F
Top line	3	Underscore	6D
Bottom line	4	Overscore	A1
Top left corner	5	Period	4B
Top right corner	6	Period	4B
Bottom left corner	7	Apostrophe	7D
Bottom right corner	8	Apostrophe	7D

You may change the contents of the data area using the i OS command Change Data Area (CHGDTAARA), for example:

```
CHGDTAARA DTAARA(YPBXCHA)VALUE('||-..''')
```

Alternatively, you can change the frame characters using the command Edit Design Defaults (YEDTDSNDFT)

Example

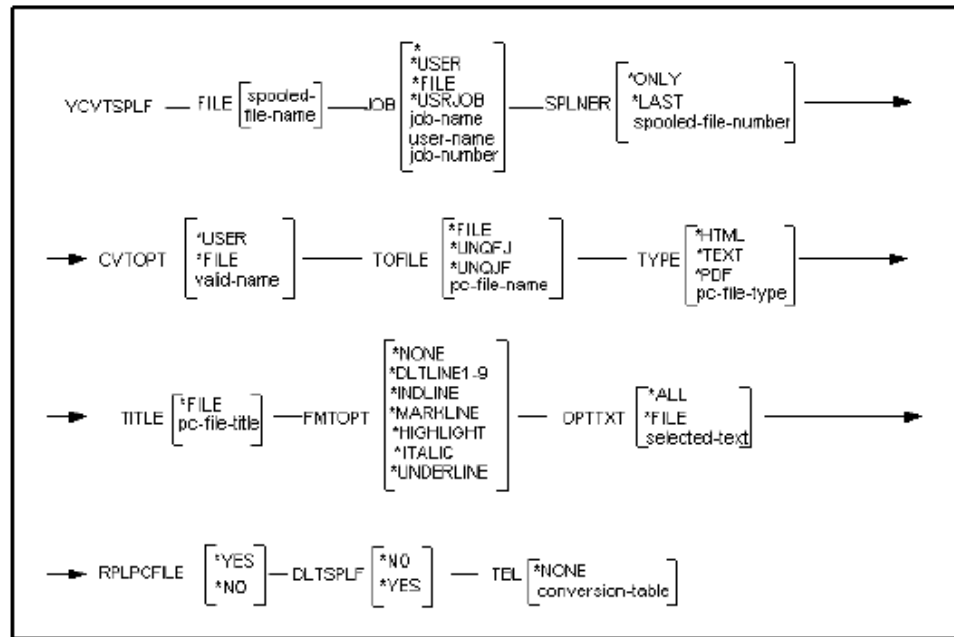
To convert all the print output from the two print files QSYSPRT and YPRTKEY to *LIBL/QTXTSRC:

```
YCVTPRT
```

YCVTSPLF (Convert Spooled File)

Converts an IBM i spooled file to a PC file.

- Allows conversion to a plain text file, an HTML file, or a file type of your choice
- Allows selection of a single spooled file from a list of duplicates
- Allows selection of folder on the IBM i that is accessible from a PC network.



Parameters

Parameter	Definition	Value and Description
FILE	Name of the spooled file that is being converted to a PC document. Required.	<ul style="list-style-type: none"> spooled-file-name

Parameter	Definition	Value and Description
JOB	<p>Name of the job that created the specified spooled file.</p> <p>You can enter one of the special values detailed below, the job name only, the job name and job user only or the job name, job user and job number.</p>	<ul style="list-style-type: none">■ *: Named spooled file from the job that issued this command.■ *USER: Named spooled file from any job run by the user executing this command.■ *FILE: Named spooled file from any job where the job has the same name as the spooled file.■ *USRJOB: Named spooled file from any job run by the user executing this command where the job has the same name as the spooled file.■ job-name: Name of the job that contains the spooled file.■ user-name: User name that identifies the user profile under which the job is run.■ job-number: Systemassigned job number.
SPLNBR	<p>Unique number of the spooled file in the job which is being converted.</p>	<ul style="list-style-type: none">■ *ONLY: Only one spooled file in the job has the specified file name; therefore, the number of the spooled file is not necessary.■ *LAST: If there is more than one spooled file with the specified file name the one with the highest number is the file which will be converted.■ spooled-file-number: Number of the spooled file that matches the file name which you wish to convert.

Parameter	Definition	Value and Description
CVTOPT	Options that will be used when converting the spooled file. These options include the name of the PC folder to which the document will be copied, and, if converting to an HTML document, an optional background, header and footer. Conversion options are held in the YCVTOPTP file. They can added or changed by using the YCHGCVTOPT command.	<ul style="list-style-type: none"> ■ *USER: If a record for the user running this command is found in the YCVTOPTP file, the details from that record are used to determine the conversion options for the spooled file. If no record is found for the user, the default record on the file ('*USER') is used. ■ *FILE: Use the conversion option record with the same name as the spooled file being converted. ■ valid-name: If a record is found in the YCVTOPTP file with a key matching the value entered, the details from that record are used to determine the conversion options for the spooled file. If no record is found, the default record on the file ('*USER') is used.
TOFILE	Name of the PC file into which the spooled file will be converted. If the conversion options chosen will convert the spooled file into a document in the Document Library System (QDLS) folder, the maximum length of this field is 8 characters.	<ul style="list-style-type: none"> ■ *FILE: PC file has the same name as the IBM i spooled file. ■ *UNQFJ: PC file has a unique name composed of the spooled file name concatenated with the spooled file number concatenated with the job name, with all variables separated by underscores. ■ *UNQJF: PC file has a unique name composed of the job name concatenated with the spooled file name concatenated with the spooled file name, with all variables separated by underscores. ■ pc-file-name: Name of the PC file, excluding the 3 character PC extension.

Parameter	Definition	Value and Description
TYPE	Type of PC file into which the spooled file will be converted.	<ul style="list-style-type: none">■ *HTML: PC file will be an HTML file with a PC file extension of '.HTM'.■ *TEXT: PC file will be a plain text file with a PC file extension of '.TXT'.■ *PDF: PC file will be a PDF file with a PC file extension of '.PDF'.■ pc-file-type: PC file type in the format .xxx.
TITLE	Title of the HTML file which is displayed in the top bar of the browser window.	<ul style="list-style-type: none">■ *FILE: Title is the same as the HTML file name (including the 3-character PC file extension).■ pc-file-title: Title of the HTML file, up to 50 characters.

Parameter	Definition	Value and Description
FMTOPT	The formatting options available when you convert a selected spooled file.	<ul style="list-style-type: none"> ■ *NONE: No formatting options should be used. ■ *DLTLINE1-9: Selected number of lines will be deleted from the converted spooled file, starting with the line containing the text selected in the OPTTXT parameter. ■ *INDLINE: Any line containing the selected text will be indented by three spaces. ■ *MARKLINE: Any line containing the selected text will be marked with '=>' in the first three spaces. ■ *HIGHLIGHT: Any text selected in the OPTTXT parameter will be highlighted, and will show as bold when the HTML file is opened and displayed. ■ *ITALIC: Any text selected in the OPTTXT parameter will be converted to italic, and will show as italic when the HTML file is opened and displayed. ■ *UNDERLINE: Any text selected in the OPTTXT parameter will be converted to underlined and will show as underlined when the HTML file is opened and displayed.
OPTTXT	Specifies the text within the spooled file that the options selected in the FMTOPT parameter should be applied to when the spooled file is converted.	<ul style="list-style-type: none"> ■ *ALL: The selected format options should be applied to all the text in the spooled file. ■ *FILE: The selected options should be applied to any instances of the spooled file name itself within the spooled file. ■ selected-text: The text to which the selected formatting options should be applied.

Parameter	Definition	Value and Description
DBFCCSID	<p>Specifies the effective CCSID that the spooled file should be treated as.</p> <p>Note: The YCVTSPLF processing will convert the spooled file to a stream file by converting FROM the CCSID specified in DBFCCSID, TO the CCSID specified in STMFCODPAG. Ideally the DBFCCSID (from CCSID) should map to the STMFCODPAG (to CCSID). See the following link for supported mappings: http://publib.boulder.ibm.com/infocenter/IBM/i/v5r3/index.jsp?topic=/nls/rbagsccsidmappings.htm</p>	<ul style="list-style-type: none">■ *YDBFCCS: The value of this argument will be derived from the YDBFCCS data area in the library list. If the YDBFCCS data area does not exist in the library list an error will be thrown. The YDBFCCS data area can be created with the YCRTY1DTA command.■ *JOB: The value of this argument will be derived from the "Default coded character set identifier" of the job that invokes the YCVTSPLF command.■ CCSID: Specify the effective CCSID.

Parameter	Definition	Value and Description
CNVALT	Specifies the internal "Conversion alternative" argument to the YCVTSPLF internal processing API	<ul style="list-style-type: none">■ *AUTO: The YCVTSPLF internal processing will determine an appropriate value for the conversion alternative. In the event that *AUTO does not provide successful conversion, specify 0, 57 or 102.■ 0: This will be passed to the YCVTSPLF internal processing API. The IBM-defined default conversion method and the associated conversion tables. Most of the default tables follow the round-trip conversion criterion. For the default tables that do not follow the round-trip conversion criterion, see the Globalization topic in the IBM i Information Center. See DBFCCSID and STMFCODPAG for further details.■ 57: This 57 will be passed to the YCVTSPLF internal processing API. The enforced subset match (substitution) criterion. For the CCSID conversion pairs that support this criterion, refer to the Globalization topic in the IBM i Information Center. See DBFCCSID and STMFCODPAG for further details.■ 102: This will be passed to the YCVTSPLF internal processing API. The best-fit conversion criterion for character mismatch. See DBFCCSID and STMFCODPAG for further details.

Parameter	Definition	Value and Description
STMFCODPAG	<p>Specifies the effective stream file code page of the resultant stream file.</p> <p>Note: The YCVTSPLF processing will convert the spooled file to a stream file by converting FROM the CCSID specified in DBFCCSID, TO the CCSID specified in STMFCODPAG.</p> <p>Ideally the DBFCCSID (from CCSID) should map to the STMFCODPAG (to CCSID). See the following link for supported mappings: http://publib.boulder.ibm.com/infocenter/IBM/i/v5r3/index.jsp?topic=/nls/rbagsccsidmappings.htm</p>	<ul style="list-style-type: none">■ *YSTMFCOD: The value of this argument will be derived from the YSTMFCOD data area in the library list. If the YSTMFCOD data area does not exist in the library list an error will be thrown. The YSTMFCOD data area can be created with the YCRTY1DTA command.■ *ASCII: Equivalent to passing 819 (ASCII).■ *WINDOWS: Equivalent to passing 1252 (Windows default encoding).■ stream-file-code-page: Specify the code page of the resultant stream file.

Parameter	Definition	Value and Description
PDFSKL	<p>Specifies the skeleton to be used for generation of the PDF format.</p> <p>Note: The PDFSKL parameter is only relevant when the PC File Type (TYPE) argument is *PDF.</p>	<ul style="list-style-type: none"> ■ *YPDFSKL: The value of this argument will be derived from the YPDFSKL data area in the library list. If the YPDFSKL data area does not exist in the library list an error will be thrown. The YPDFSKL data area can be created with the YCRTY1DTA command. ■ *WEST: The PDF file will be formatted to present non Japanese fonts. <p>For example, *WEST is appropriate when DBFCCSID/STMFCODPAG is 37/819 or 285/1252.</p> <p>*JPN</p> <p>The PDF file will be formatted to present Japanese fonts.</p> <p>E.G. *JPN is appropriate when DBFCCSID/STMFCODPAG is 5035/932 or 5026/932.</p>
FONTsiz	<p>Specifies the size of font to be used for generation of the PDF format.</p>	<p>font-size</p> <p>Specify the font size</p> <p>Note: The FONTsiz parameter is only relevant when the PC File Type (TYPE) argument is *PDF.</p> <p>Note: Specifying FONTsiz > 10 can cause overlap between lines of data in the PDF. Adequate spacing between lines in the spooled file can prevent this.</p>

Parameter	Definition	Value and Description
RPLPCFILE	Specifies whether an existing PC file in the selected directory should be replaced.	<ul style="list-style-type: none"> ■ *YES: If a PC file of the same name and type exists in the selected directory, it will be replaced with the PC file created by this command. ■ *NO: If a PC file of the same name and type exists in the selected directory, it will not be replaced.
DLTSPLF	Specifies whether the selected spooled file should be deleted from the IBM i after being converted.	<ul style="list-style-type: none"> ■ *NO: The selected spooled file will not be deleted after being converted. ■ *YES: The selected spooled file will be deleted after being converted.
TBL	Qualified name of the conversion table used to convert data from the spooled file to the PC file.	<ul style="list-style-type: none"> ■ *NONE: No conversion table will be used to convert the data. ■ conversion-table: The data is converted using a conversion table.

Notes

If you are using YCVTSPLF library with Japanese or Simplified Chinese DBFCCSID values, you must use the following suggested DBFCCSID and STMFCODPAG mappings while running YCVTSPLF.

Simplified Chinese

DBFCCSID(1388) STMFCODPAG(1386)

Japanese

DBFCCSID(5035) STMFCODPAG(932)

DBFCCSID(5026) STMFCODPAG(932)

DBFCCSID(1399) STMFCODPAG(943)

Examples

The following command converts spooled file *YCHKMDLL1\$* with File number 3, from job *QPADEV0005* with specified job number and user, to an HTML file (with the same name) in folder '/ysplf/coxpa02' on the IFS.

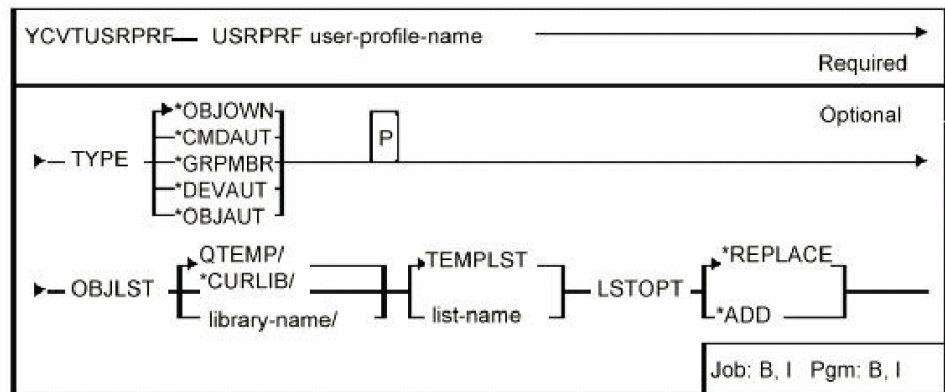
Additionally, by using the FMTOPT and OPTTXT parameters, we have specified that any instance of text 'YAPPFILRFP' in the document will be underlined.

```
YCVTSPLF FILE(YCHKMDLL1$) JOB(304841/COX4002/QPADEV0005) SPLNBR(3)
FMTOPT(*UNDERLINE) OPTTXT(YAPPFILRFP)
```

YCVTUSRPRF (Convert User Profile)

This command creates an object list of all the objects owned or accessible by a user, using the output of the i OS command Display User Profile (DSPUSRPRF).

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
USRPRF	Name of user profile whose objects are to be included in the list	
TYPE	Type of objects to be included in the list.	Can be one of the following types: <ul style="list-style-type: none">■ *OBJOWN: (default) Objects owned by the user profile■ *CMDAUT: Commands to which the user profile is explicitly authorized■ *GRPMBR: If the profile is a group profile, all the profiles belonging to the user profile group■ *DEVAUT: Device descriptions to which the user profile is explicitly authorized■ *OBJAUT: Objects to which the user profile is explicitly authorized
OBJLST	Qualified name of object list that is to be built	QTEMP/TEMPLST: (default) List name
LSTOPT	List replacement option	<ul style="list-style-type: none">■ *REPLACE: (default) Create a new list replacing any previous list's contents■ *ADD: Add to any existing list's contents

Notes

If object type is *CMD, set to QCMDSRC. If object type is *FILE, set to QDDSSRC, otherwise blank.

User authorities placed in field.

If, using the command Filter Object List (YFLTOBJLST), you attempt to filter the resultant object list on attributes other than those shown above (such as object size) unpredictable results may occur.

Examples

To build a list of all objects owned by FRED:

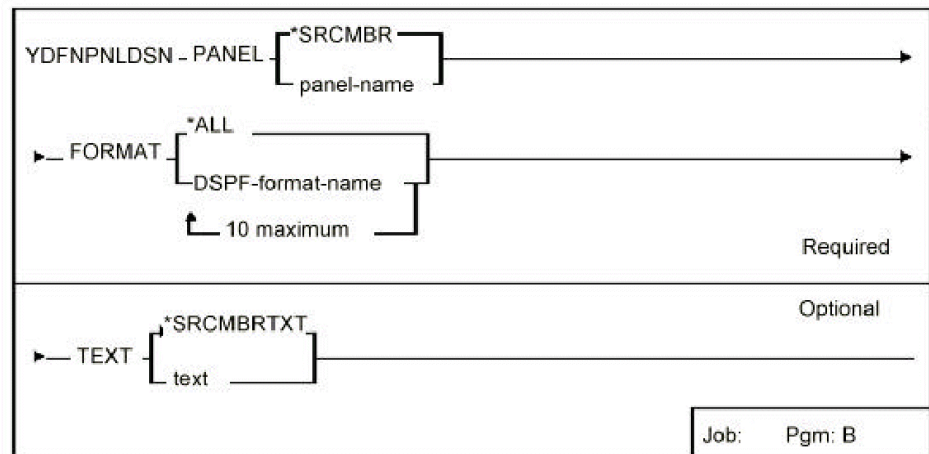
```
YCVTUSRPRF USRPRF (FRED)
```

The list is called TEMPLST and resides in library QTEMP.

YDFNPNLDSN (Define Panel Design)

This command defines a combination of DDS display formats for use by the command Retrieve Panel Design (YRTVPNLDSN). The command should be placed as a comment statement in the source of the Display file DDS which is retrieved into a panel design.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
PANEL	Name to be given to retrieved panel design. Any existing version of the design is replaced	*SRCMBR: (default) Panel design is given the same name as the source member containing the YDFNPNLDSN command

Parameter	Definition	Value and Description
FORMAT	List of names of up to ten display file formats in the source member which are to be combined to make a single panel design	*ALL: (default) All formats in the specified source member is retrieved
TEXT	Title to be given to panel design	*SRCMBRTXT: (default) Design is given the same text as the source member which contains the YDFNPNLDSN command

Notes

1. This command is only for use in conjunction with the command Retrieve panel design (YRTVPNLDSN). Command statements to define panel designs can be included in the DDS source of display files, which are retrieved as panel designs.
Restrictions are as follows:
 - Panel definition statements, that is, comment lines containing the YDFNPNLDSN statement, must be included in the first 30 lines of the source.
 - Each panel definition statement must be on a single line.
2. Formats are included in the panel design in the order they appear in the FORMAT parameter. If there is an overlap between the panel area covered by a format, then each subsequent format overlays any previous formats.

Example

If the DDS source for display file FRED contains five formats #KEY, #SFCT#1, #SFRC#1, #CMDKEY, and #DTLSA, then the following two statements would cause the YRTVPNLDSN command to retrieve the source of FRED as two panel designs:

- WIDGKEYS containing illustrations of #KEY and #CMDKEY combined as one panel
- WIDGDTLS containing illustrations of #SFCT#1, #SFRC#1, #CMDKEY

```

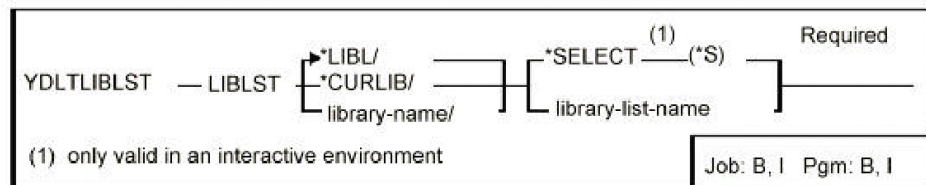
T*: Display widgets
Z* CRTDSPF RSTDSP(*NO)
* YDFNPNLDSN PANEL(WIDGKEYS) FORMAT(#KEY #CMDKEY) TEXT('KEYS')
* YDFNPNLDSN PANEL(WIDGDTLS) FORMAT(#SFCT#1 #SFRC#1 #CMDKEY)
.....
A          R #KEY                      TEXT('Key screen')
.....
A          R #DTLSA                     TEXT('Details  ')
.....
A          R #SFRC#1                    TEXT('SFL record')
.....
A          R #SFCT#1                    TEXT('SFL control')
.....
A          R #CMDKEY                     TEXT('Command keys')

```

YDLTLIBLST (Delete Library List)

This command deletes a library list.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
LIBLST	Qualified name of deleted library list	*SELECT: Display list of existing lists

Notes

Library lists are stored in file YLIBLST in the library specified by the LIBLST parameter.

Example

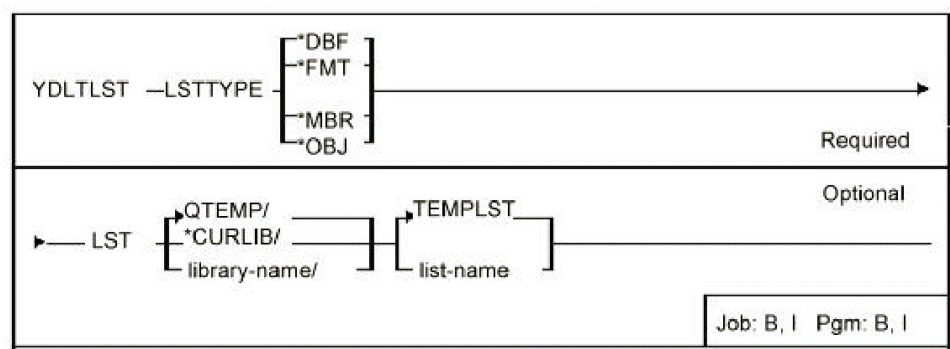
To delete library list DEWEY:

```
YDLTLIBLST LIBLST(DEWEY)
```

YDLTLST (Delete Library List)

This command deletes an object, member, format or database file list.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
LSTTYPE	List type of the deleted list	<ul style="list-style-type: none">*DBF: Database file list*FMT: Format list*MBR: Member list*OBJ: Object list
LST	Qualified name of the deleted list	QTEMP/TEMPLST: (default) List name

Notes

None

Example

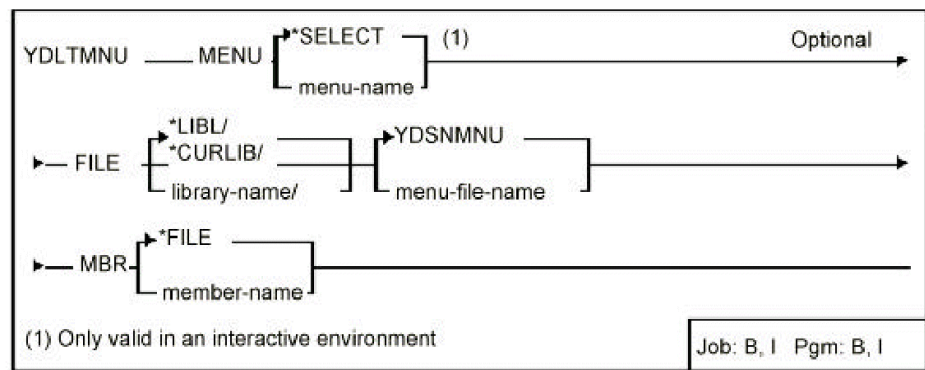
To delete object list TOWER in library PISA:

```
YDLTLST LSTTYPE(*OBJ) LST(PISA/TOWER)
```

YDLTMNU (Delete Menu)

This command deletes a menu.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
MENU	Name of the deleted menu	*SELECT: (default) Display list of existing menus
FILE	Qualified name of file containing menu	*LIBL/YDSNMNU: (default) Name for menu file
MBR	Name of member in file containing menu	*FILE: (default) Member name is same as file name

Notes

The Delete Menu command does not check for references to the deleted menu from any other menu.

Example

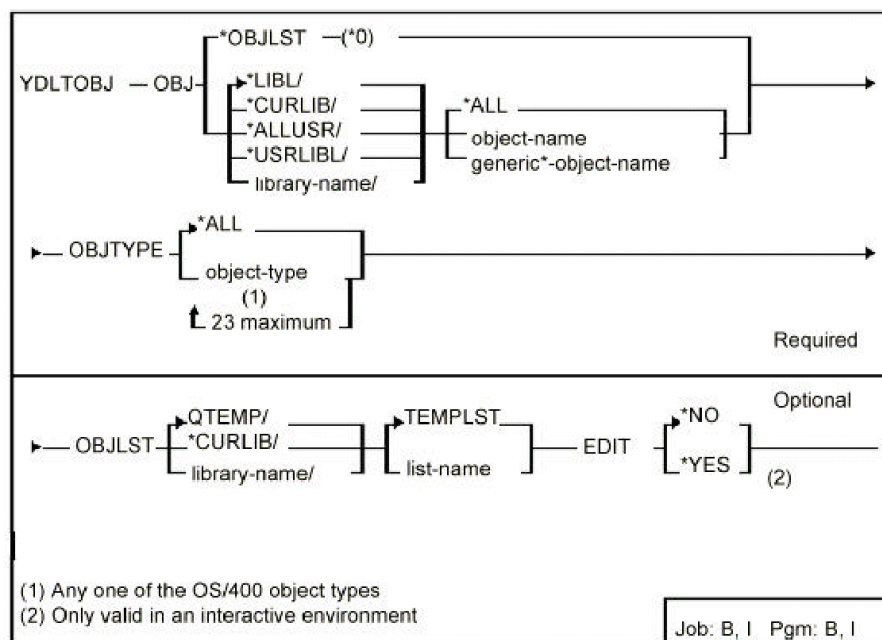
To delete menu ALACARTE in menu file PEREBISE:

```
YDLTMNU MENU(ALACARTE) FILE(PEREBISE)
```

YDLTOBJ (Delete Object)

This command deletes an object or objects.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
OBJ	Qualified generic name of objects which are to be deleted	<ul style="list-style-type: none"> ■ *OBJLST: (default) Objects are specified by a object list whose name is given by the OBJLST parameter ■ *ALL: All objects
OBJTYPE	List of types of objects which are to be deleted	*ALL: (default) All object types
OBJLST	Qualified name of deleted objects in the object list	QTEMP/TEMPLST: (default) Name for list If OBJ(*OBJLST) is specified an existing list is used. Otherwise a new list is created
EDIT	Edit list option	<ul style="list-style-type: none"> ■ *NO: (default) Edit list function is not to be invoked ■ *YES: Edit the list once built (valid only for interactive programs)

Notes

1. The user must have object existence rights to the objects for deletion.
2. System objects cannot be deleted with this command. There are three exceptions to this rule:
 - Objects of type *USPRF, *DEVD, *LIND which reside in library QSYS can be deleted.
 - Any objects in library QTEMP can be deleted.
 - Any objects in library QGPL whose names do not begin with the letter 'Q' can be deleted.

Examples

To delete all objects in library QGPL whose names begin with the letter 'Z':

```
YDLTOBJ OBJ(QGPL/Z*) OBJTYPE(*ALL)
```

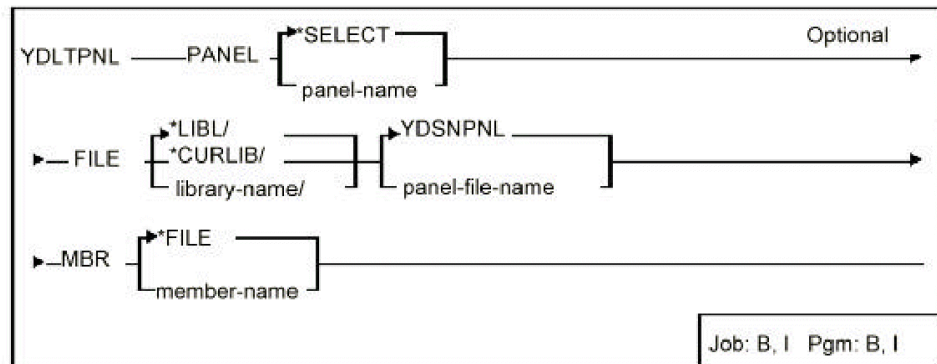
To delete all objects in list KILL in library QGPL:

```
YDLTOBJ OBJ(*OBJLST) OBJTYPE(*ALL) OBJLST(QGPL/KILL)
```

YDLTPNL (Delete Panel Design)

This command deletes a panel design.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
PANEL	Name of the deleted panel design	*SELECT: (default) The panel design selection display is presented
FILE	Qualified name of file containing panel design	YDSNPNL: (default) Panel file name
MBR	Name of member in file containing panel design	*FILE: (default) The member has the same name as the file

Notes

The command does not check for references to the deleted panel design from any other panel designs.

Example

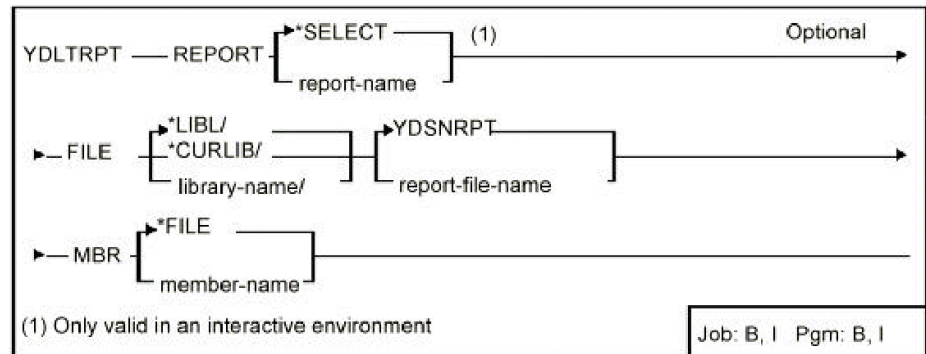
To delete panel design ODEON in panel file YDSNPNL:

```
YDLTPNL PANEL (ODEON)
```


YDLTRPT (Delete Report Design)

This command deletes a report design.

Syntax Parameters



Parameters

Parameter	Definition	Value and Description
REPORT	Name of deleted report design	*SELECT: (default) The report design selection display is presented
FILE	Qualified name of file containing report design	YDSNRPT: (default) Report design file name
MBR	Name of member in file containing report design	*FILE: (default) The member has the same name as the file

Notes

None

Example

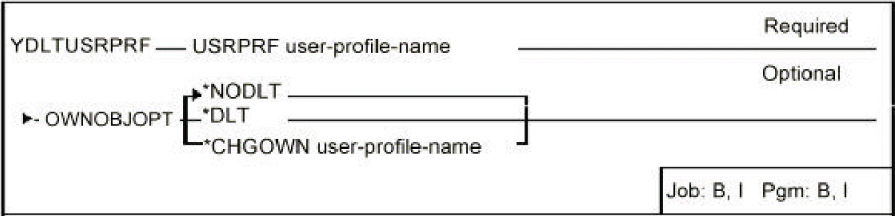
To delete report design BEVERIDGE in report file YDSNRPT:

```
YDLTRPT REPORT(BEVERIDGE)
```

YDLTUSRPRF (Delete User Profile)

This command deletes a specified user profile, where both the i OS user profile and the profile details are deleted.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
USRPRF	Name of deleted user profile	
OWNOBJOPT	Specifies the type of operations being performed on the owned objects of the user profile being deleted	<ul style="list-style-type: none"> ■ NODLT<M%-2>: (default) the owned objects for the user profile are not changed, and the user profile is not deleted if the user owns any objects ■ <M%-4>*DLT: The owned objects for the user profile are deleted. The user profile is also deleted if the deletion of all owned objects is successful ■ *CHGOWN: The owned objects for the user profile have ownership transferred to the specified user profile. The user profile is deleted if the transfer of all owned objects is successful <p>A user profile name must be specified for the new user profile. The new user profile owns all objects owned by the user profile specified by the USRPRF parameter</p>

Notes

1. You must have security officer rights to use this command.
2. Deletes both the i OS and the user profile details.
3. User profile details are stored in a file called YUSRPRF.

Example

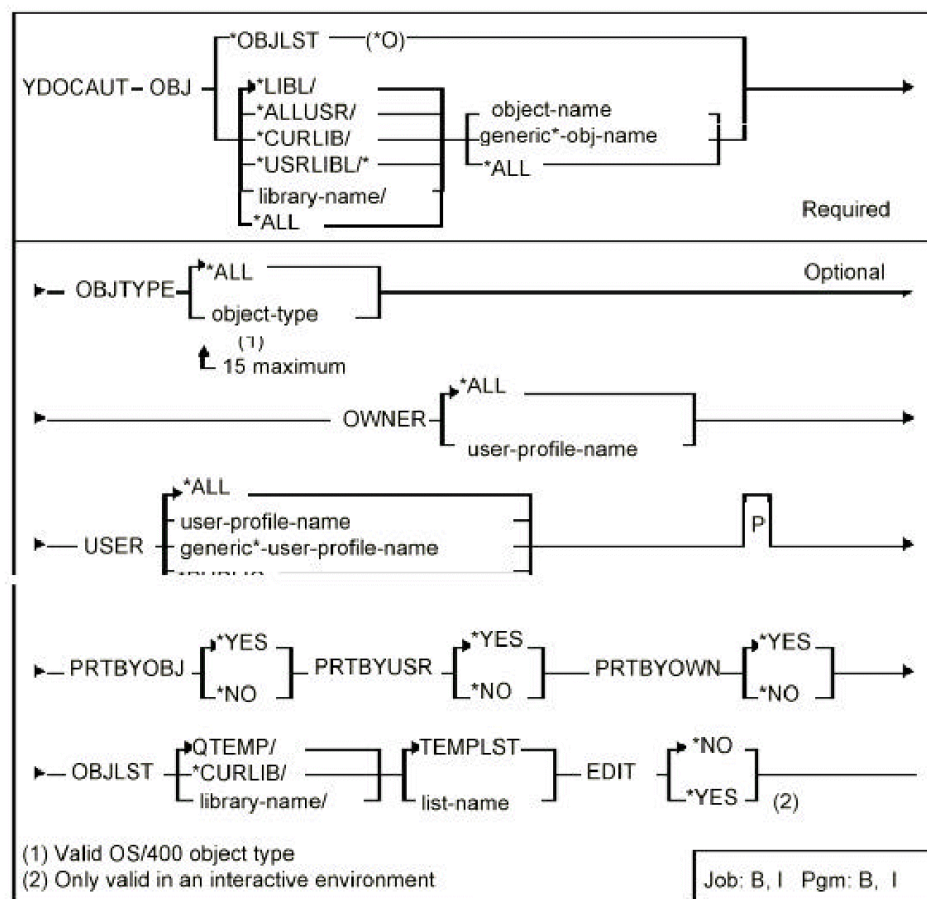
To delete user profile NOBLE:

YDLTUSRPRF USRPRF (NOBLE)

YDOCAUT (Document Object Authorities)

This command documents object authorities, by object, owner, or user profile names.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
OBJ	Qualified generic name of objects which are included in cross-reference listing	*OBJLST: (default) Use the list of objects specified by the OBJLST parameter
OBJTYPE	Lists object types included in the listing	*ALL: (default) All objects are to be included
OWNER	User profile name of owner of objects which are included in listing	*ALL: (default) Objects are to be include for all object owners
USER	Generic name of user profiles which are included in cross-reference listing	<ul style="list-style-type: none"> ■ *ALL: (default) Objects are to be included for all object users ■ *PUBLIC: Include objects for which user *PUBLIC is specified as an authorized user ■ *NONE: Include objects for which no authorized user is specified
PRTBYOBJ	Cross-reference by object option	<ul style="list-style-type: none"> ■ *YES: (default) List authorities in object order ■ *NO: No object order list is produced
PRTBYUSR	Cross-reference by user option	<ul style="list-style-type: none"> ■ *YES: (default) List authorities in object user order ■ *NO: No object user list is produced
PRTBYOWN	Cross-reference by owner option	<ul style="list-style-type: none"> ■ *YES: (default) List authorities in object owner order ■ *NO: No object owner list is produced
OBJLST	Qualified name of object list	QTEMP/TEMPLST: (default) List name

Parameter	Definition	Value and Description
EDIT	List operation	<ul style="list-style-type: none"> ■ *NO: (default) Do not invoke the edit list function ■ *YES: The edit list function is to be invoked on the list before the cross-reference is run

Notes

1. This command takes a long time to run.
2. Any combination of listing options may be made.

Example

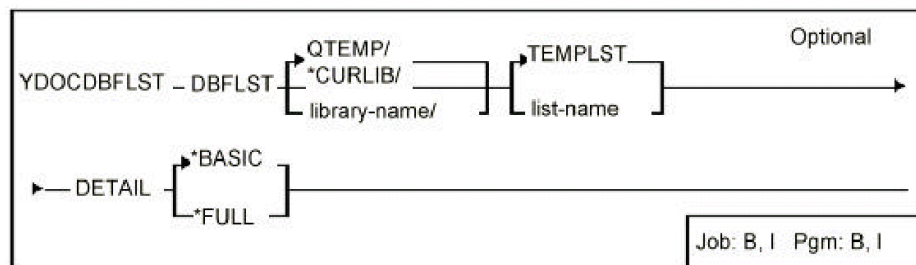
To run cross references by object, user and owner for all objects in library QGPL:

```
YDOCAUT OBJ(QGPL/*ALL) OBJTYPE(*ALL)
```

YDOCDBFLST (Document DBF List)

This command prints the contents of a database file list.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
DBFLST	Qualified name of the printed database file	QTEMP/TEMPLST: (default) List name
DETAIL	Level of detail to print	<ul style="list-style-type: none">■ *BASIC: (default) Print only basic details about each dbf file (name, library, type, text)■ *FULL: Print additional information about each dbf file

Notes

None

Example

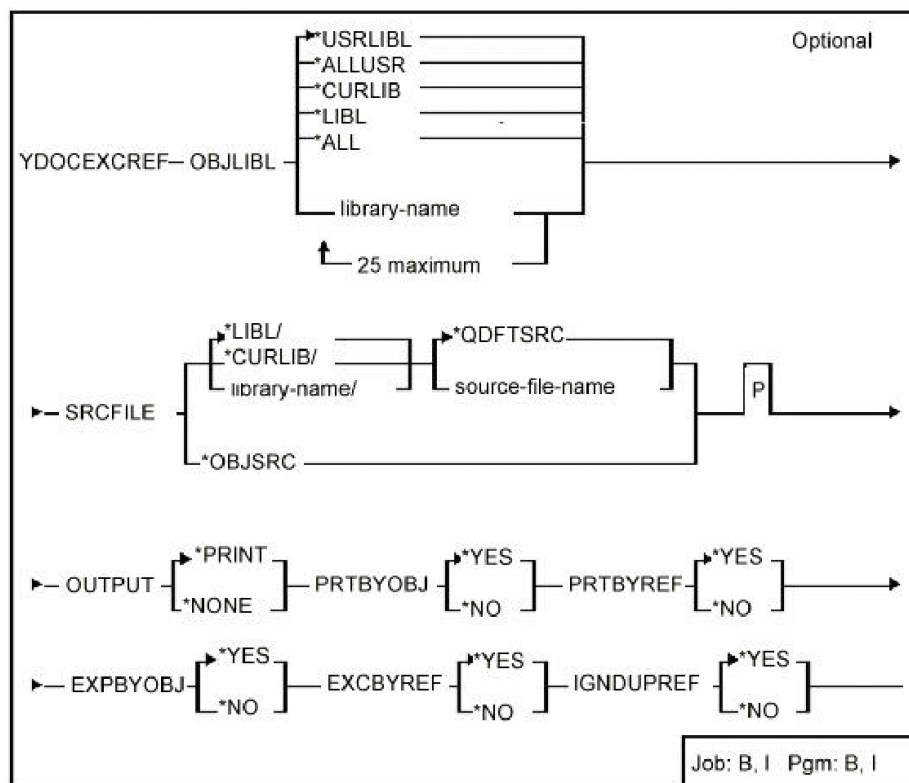
To print the contents of list TEMPLST in library QTEMP:

```
YDOCDBFLST
```

YDOCEXCREF (Document Execution References)

This command produces a cross-reference listing all source references to executable objects. Listings may be produced of referenced objects by calling object, or vice versa.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
OBJLIBL	List of libraries searched to locate objects which are included in listing	<ul style="list-style-type: none"> ■ *USRLIBL: (default) Use user portion of library list to locate objects ■ *LIBL: Use library list to locate objects ■ *ALL: Use all libraries on machine ■ *ALLUSR: Use all user libraries on machine to locate objects ■ *CURLIB: Use current library to locate objects ■ SRCFILE Qualified source file name for CL and RPG III source members containing references to be tabulated ■ *QDFTSRC: (default) The source file name defaults to QCLSRC or QRPGRSRC depending on program type ■ *OBJSRC: The source file name is taken from the source file name in the object description
OUTPUT	Output option	<ul style="list-style-type: none"> ■ *PRINT: (default) Printed output is to be produced ■ *NONE: No printed output is required
PRTBYOBJ	Produce a report of referenced objects by calling object	<ul style="list-style-type: none"> ■ *YES: (default) A listing is produced ■ *NO: A listing is not produced
PRTBYREF	Produce a report of calling objects by referenced object	<ul style="list-style-type: none"> ■ *YES: (default) A listing is produced ■ *NO: A listing is not produced

Parameter	Definition	Value and Description
EXPBYOBJ	Produce an explosion report of referenced objects by calling object	<ul style="list-style-type: none"> ■ *YES: (default) A listing is produced ■ *NO: A listing is not produced
EXPBYREF	Produce an explosion report of calling objects by referenced object	<ul style="list-style-type: none"> ■ *YES: (default) A listing is produced ■ *NO: a listing is not produced
IGNDUPREF	Ignore duplicate references option	<ul style="list-style-type: none"> ■ *YES: (default) Omit duplicate references to the same program by a given program ■ *NO: Include duplicate references to the same program by a given program

Notes

1. This command takes a long time to execute.
2. When OUTPUT(*NONE) is specified the following work files are created in library QTEMP:

Output file	File	Mbr
Execution references	YDEXRFP	YDEXRFP
Execution references -by calling object. (Only if PRTBYOBJ(*YES))		

3. The following considerations apply to the use of the command YDOCEXCREF if you have added libraries to the system part of your library list; for example, changed the system value QSYSLIBL.
 - When run as part of an interactive job, the YDOCEXCREF command may not be used to document libraries that are in the system part of the job's library list. If you wish to document such libraries interactively, first use the i OS command Change System Value (CHGSYSVAL) to remove the libraries from the system part of the job's library list.

- When run as part of a batch job, the YDOCEXCREF command always removes all libraries except QSYS from the system part of the job's library list. Thus the command may be used to document libraries that are in the system part of the library list. Note however that if you use the YDOCEXCREF command in your own programs, you may want to restore the system part of the library list after invoking the command.

Example

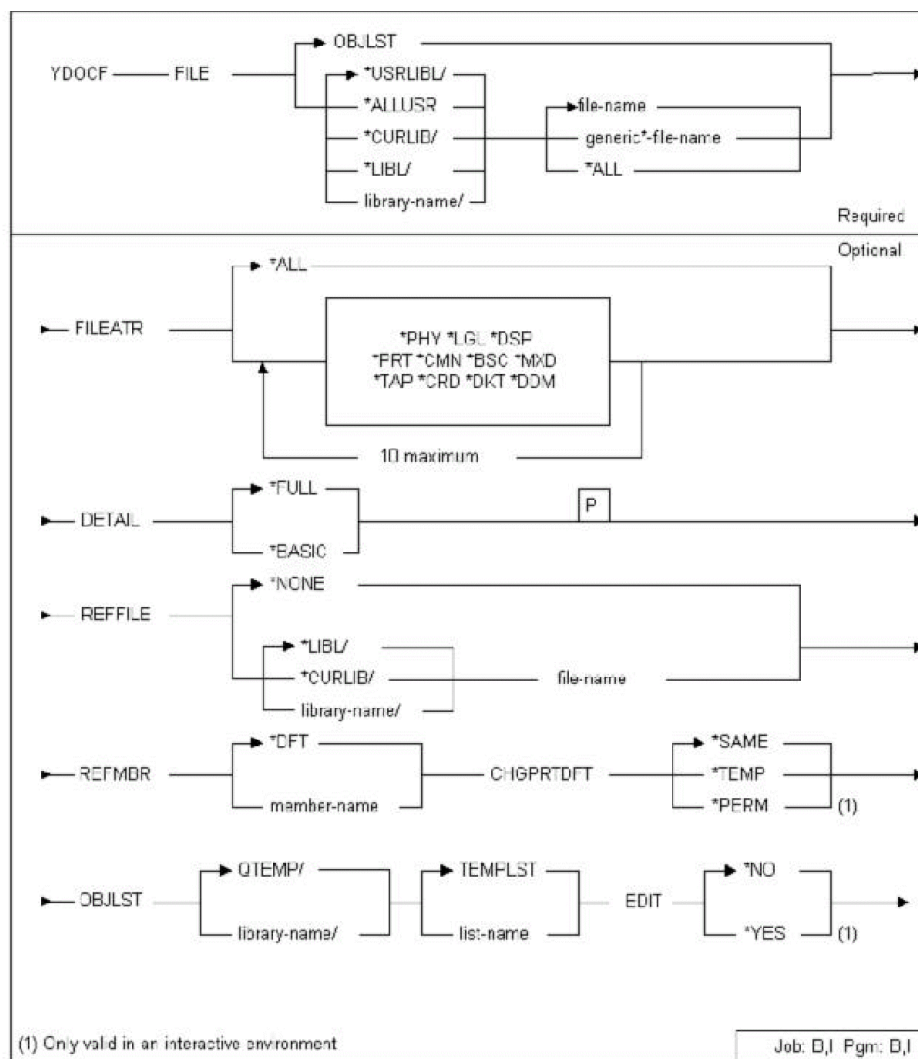
To print execution references for all objects in libraries KETCH and DERRICK:

```
YDOCEXCREF OBJLIBL(KETCH DERRICK)
```

YDOCF (Document File)

This command prints summary information about a specified file or files. Object, format, field, and access path information is included.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
FILE	Qualified generic name of files that are to be listed	<ul style="list-style-type: none"> ■ *OBJLST: List files specified in object list named by the OBJLST parameter. ■ *ALL: List all file types
FILEATR	List of attributes of files that are to be listed	*ALL: (default) List all types of file
DETAIL	Level of detail required	<ul style="list-style-type: none"> ■ *FULL: A 132 column report for documentation ■ *BASIC: An 80 column report suitable for a system specification
REFFILE	Qualified name of source file containing DDS for field reference file. See note below for further details	*NONE: (default) Text from the field reference source member is not to be printed
REFMBR	Name of source file member containing DDS for field reference file	*DFT: (default) The name of the field reference file is taken from the design defaults
CHGPRTDFT	Change print defaults	<ul style="list-style-type: none"> ■ *SAME: (default) The print defaults are not to be changed ■ *TEMP: The print defaults are to be changed for the current operation only ■ *PERM: The print defaults are to be changed permanently <p>N.B. only *SAME may be used for batch jobs</p>

Parameter	Definition	Value and Description
OBJLST	Qualified name of a object list to be used to identify the files to be documented	QTEMP/TEMPLST: (default) List name. To use an existing list, specify value of *OBJLST for the FILE parameter
EDIT	Edit object list option	<ul style="list-style-type: none"> ■ *NO: (default) Edit list function is not invoked ■ *YES: Invoke edit list function to edit list before executing print

Notes

- The information listed by the YDOCF command for each file includes:
 - Object information - from DSPOBJD.
 - Format information - from DSPFD *RCDFMT
 - Access path information - from DSPFD *ACCPH (if relevant)
 - Field information - from DSPFFD
 - Database dependency information - from DSPDBR (if relevant)
 - Join dependency information - from DSPFD *JOIN.
- An index is provided via the Document Object List command (YDOCOBJLST).
- The file documentor includes a facility for including additional explanatory text for a field, in addition to that provided by the DDS TEXT or COLHDG keywords.

Additional text may be associated with any field in the field reference dictionary. The text is printed by the file documenter as part of the field level information.

Additional text must be entered as comment lines within the DDS of the field reference file containing the referenced field. The field name associated with the text must be repeated on each comment line containing the extra explanatory text. The text starts in column 30, and may be up to 50 characters in length.

The following example shows this facility being used to annotate field values:

A	##BRIN	1	COLHDG('Bank' 'Reconciled'
A			'indicator')
A			VALUES('Y' ' ')
A*	##BRIN	Y	= posting has been reconciled.
A*	##BRIN	Blank	= posting not yet reconciled.

File DDS displays as follows:

A	WHBRIN	R	REFFLD(##BRIN)
---	--------	---	----------------

YDOCF output displays as follows:

Field Name	Type/ Use	Decimal Length	Field Length	Text
WHBRIN	A I		A I	Bank reconciled indicator Y .>posting has been reconciled blank =posting not yet reconciled.

Additional uses of extra text include:

- Foreign language field names for multi-national systems.
- Additional field validations and restrictions above those supported by i OS

Example

To print documentation for all physical files whose names begin with INDIAN* in library QGPL:

```
YDOCF FILE(QGPL/INDIAN*) FILEATR(*PHY)
```

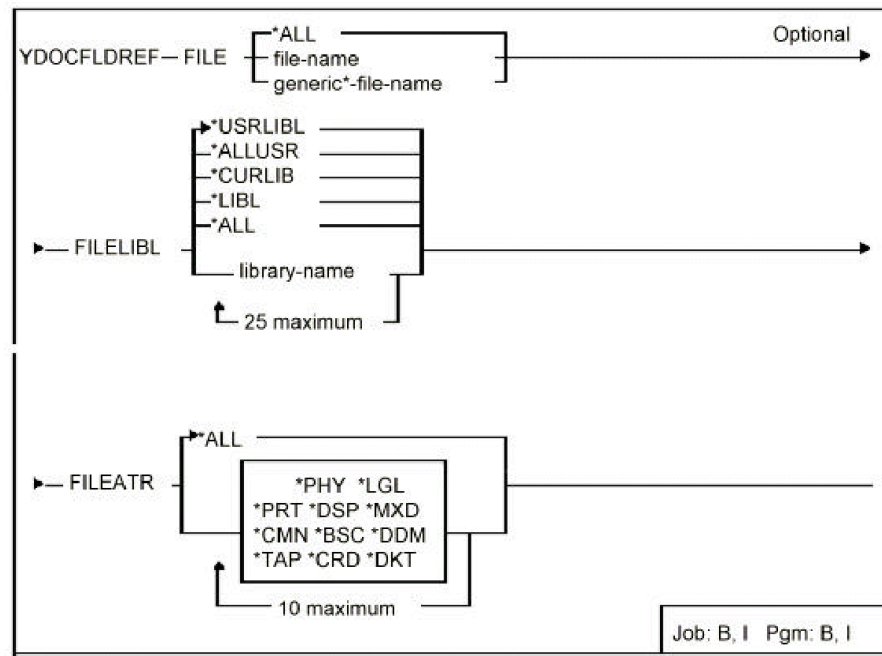
To print documentation for all files in object list QGPL/FRED, including additional explanatory text from source member (QGPL/QDDSSRC FLDREFP), which is the source member for the field reference file used by the files in the list:

```
YDOCF FILE(*OBJLST) REFFILE(QGPL/QDDSSRC) REFMBR(FLDREFP) OBJLST(QGPL/FRED)
```

YDOCFLDREF (Document Field References)

This command prints a cross-reference of fields for a specified group of files.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
FILE	Qualified generic name of files which are to be cross-referenced	*ALL: (default) List all files
FILELIBL	List of libraries to be used to locate files which are to be included in cross-reference listing	<ul style="list-style-type: none"> ■ *USRLIBL: (default) Use library list to locate files ■ *LIBL: Use library list to locate files ■ *CURLIB: Use current library to locate files ■ *ALL: Use all libraries on machine ■ *ALLUSR: Use all user libraries on machine to locate files
FILEATR	List of attributes of the files which are to be listed	*ALL: (default) List all file types

Notes

1. Special considerations apply to the use of the YDOCFLDREF command if you have added libraries to the system part of your library list.
2. When run as part of an interactive job, the YDOCFLDREF command may not be used to document libraries that are in the system part of the job's library list. If you wish to document such libraries interactively, first use the i OS Change system value command (CHGSYSVAL) to remove the libraries from the system part of the job's library list.
3. When run as part of a batch job, the YDOCFLDREF command always removes all libraries except QSYS from the system part of the job's library list. Thus the command may be used to document libraries that are in the system part of the library list. Note however that if you use the YDOCFLDREF command in your own programs, you may want to restore the system part of the library list after invoking the command.

Example

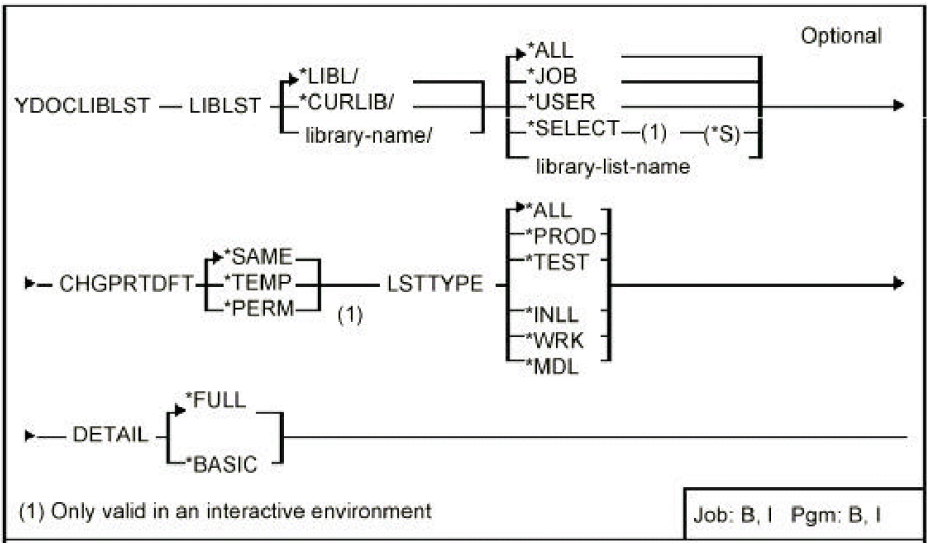
To list all field references in physical files whose names begin with X in libraries PASTURE, MEADOW and LEA:

```
YDOCFLDREF FILE(X*) FILELIBL(PASTURE MEADOW LEA) FILEATR(*PHY)
```


YDOCLIBLST (Document Library)

This command prints library lists.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
LIBLST	Qualified generic name of library lists that are documented	<ul style="list-style-type: none">■ *ALL: (default) Document all library lists■ *JOB: Use same library list name as that of current job■ *USER: Use same library list name as that of current user profile■ *SELECT: Display list of library lists contained in specified library, and allow one library list to be selected

Parameter	Definition	Value and Description
CHGPRTDFT	Change default print attributes of output	<ul style="list-style-type: none">■ *SAME: (default) The print defaults are not changed■ *TEMP: The print defaults are changed for the current operation only■ *PERM: The print defaults are changed permanently
LSTTYPE	Library list type	<ul style="list-style-type: none">■ *ALL: select all library lists.■ *PROD■ *TEST■ *INLL■ *WRK■ *MDL
DETAIL	Detail	<ul style="list-style-type: none">■ *FULL (default)■ *BASIC

Notes

See [Appendix A](#) of this manual for information on the special types for this parameter.

Examples

To print all library lists:

```
YDOCLIBLST
```

To print library list DEWEY:

```
YDOCLIBLST LIBLST(DEWEY)
```

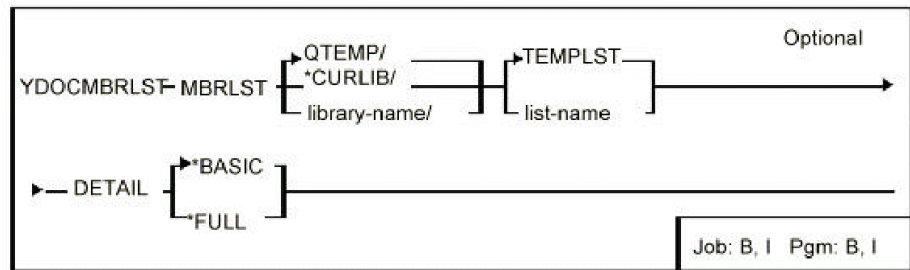
To print basic details of all library lists with type TEST in library QGPL:

```
YDOCLIBLST LIBLST(QGPL/*ALL) LSTTYPE(*TEST) DETAIL(*BASIC)
```

YDOCMBRLST (Document Member List)

This command prints the contents of a member list.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
MBRLST	Qualified name of the member list which is to be printed	QTEMP/TEMPLST: (default) List name
DETAIL	Level of detail to print	<ul style="list-style-type: none"> ■ *BASIC: (default) Print only basic details about each member (name, library, SEU type, text) ■ *FULL: Print additional information about each member

Notes

None

Example

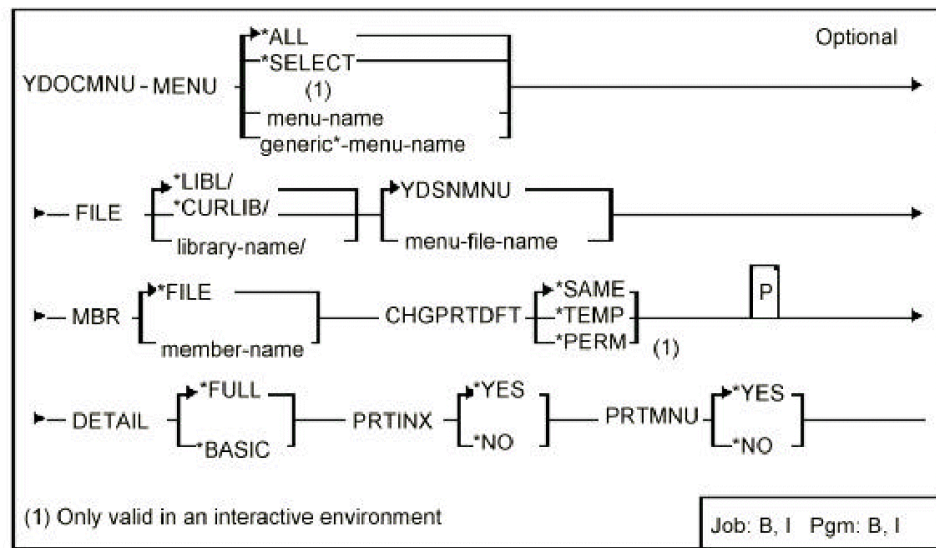
To print the contents of member list TEMPLST in library QTEMP:

```
YDOCMBRLST
```

YDOCMNU (Document Menu)

This command prints menus.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
MENU	Generic name of menu or menus which are to be documented	<ul style="list-style-type: none"> *ALL: (default) All menus in the file are to be documented *SELECT: A list of menus will be presented
FILE	Qualified name of file containing menus, which are documented. File must have been created with Create Design File (YCRTDSNF) command	*LIBL/YDSNMNU: (default) Menu file name
MBR	Name of member in menu file containing menus	*FILE: (default) The member has the same name as the file

Parameter	Definition	Value and Description
CHGPRTDFT	Change print defaults	<ul style="list-style-type: none"> ■ *SAME: (default) The print defaults are not changed ■ *TEMP: The print defaults are changed for the current operation only ■ *PERM: The print defaults are changed permanently <p>N.B. Only *SAME may be used for batch jobs</p>
DETAIL	Level of detail to be included	<ul style="list-style-type: none"> ■ *FULL: (default) Print full documentation ■ *BASIC: Only print menu screen images
PRTINX	Print menu index	<ul style="list-style-type: none"> ■ *YES: (default) The index is listed. ■ *NO: No index is listed
PRTMNU	Print menus	<ul style="list-style-type: none"> ■ *YES: (default) The menus are listed ■ *NO: Menus are not listed

Notes

None

Examples

To print all menus in file *LIBL/YDSNMNU:

```
YDOCMNU
```

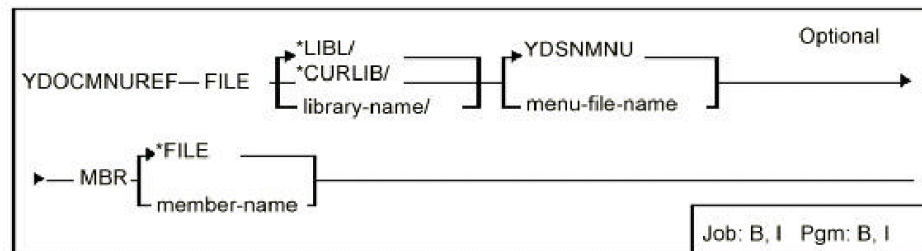
To print menu TAGESKARTE in file *LIBL/YDSNMNU:

```
YDOCMNU MENU(TAGESKARTE)
```

YDOCMNUREF (Document Menu References)

This command prints usage of menus by option.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
FILE	Qualified name of file containing menus to be cross-referenced	*LIBL/YDSNMNU: (default) Name
MBR	Name of member in menu file containing menus	*FILE: (default) The member has the same name as the file

Notes

None

Example

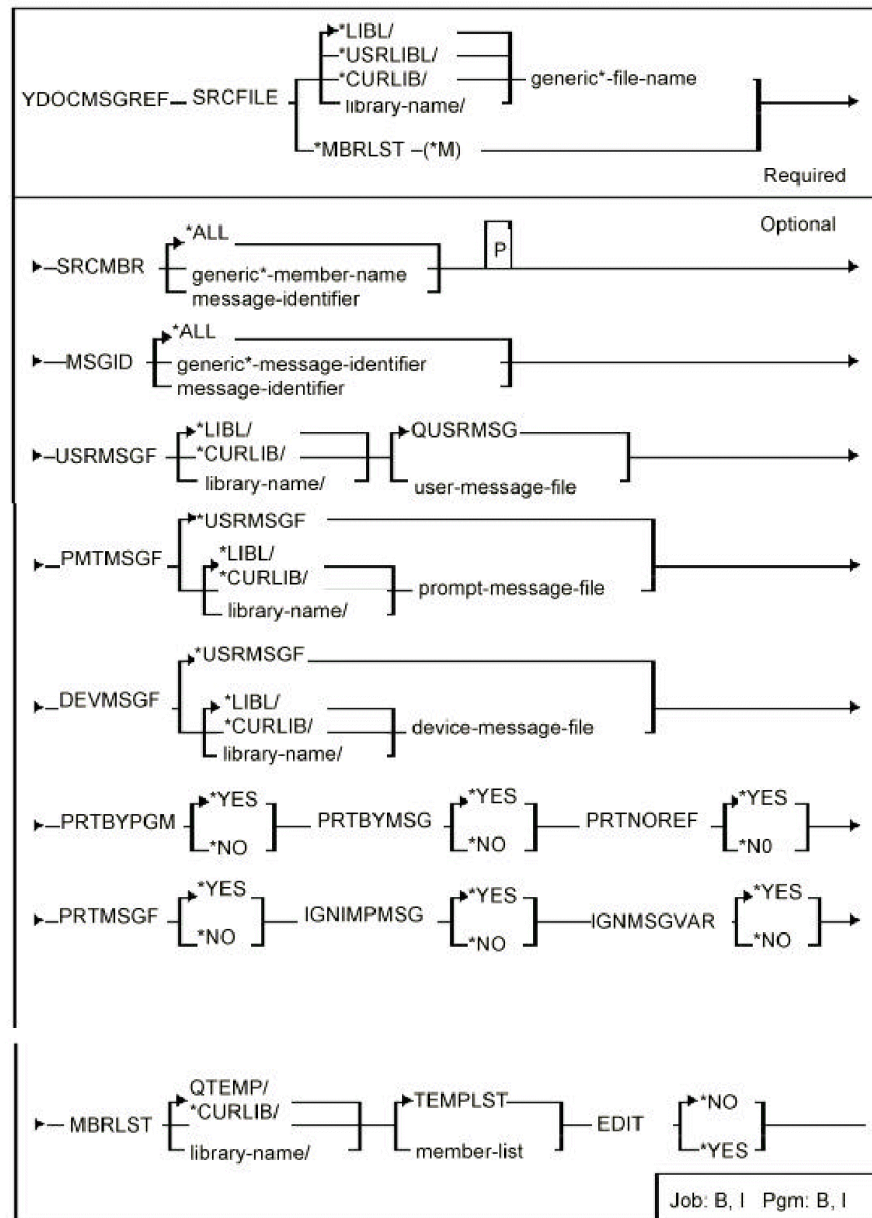
To print all menus in file *LIBL/YDSNMNU:

```
YDOCMNUREF
```

YDOCMSGREF (Document Message References)

This command prints cross-references of usage of messages for a specified program, a group of programs or a list of programs.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
SRCFILE	Qualified generic name of programs whose message references are documented	*MBRLST: (default) Use the member list named in the MBRLST parameter to identify the source members which are processed
SRCMBR	Generic name of members contained in the specified source file(s), which are included in the cross-referencing	*ALL: (default) Use all members in the files, or contained in the specified list
MSGID	Generic message identifier upon which the cross-referencing is based	*ALL: (default) Include all message identifiers
USRMSGF	Qualified name of the message file from which first level message text is to be retrieved for any found message identifiers. This file is also used for the listing of unreferenced messages, and for the listing of messages in text order	*LIBL/QUSRMSGF: (default) Message file name
PMTMSGF	Qualified name of the message file from which command prompt text is retrieved for any message identifiers found in command source. This file is also used for the listing of unreferenced messages, and for the listing of messages in text order	*USRMSGF: (default) Prompt messages are stored in the message file specified by the USRMSGF parameter

Parameter	Definition	Value and Description
DEVMSGF	Qualified name of the message file from which DDS prompt text (by use of the MSGCON keyword) is retrieved for any message identifiers found in DDS source. This file is also used for the listing of unreferenced messages, and for the listing of messages in text order	*USRMSGF: (default) Device messages are stored in the message file specified by the USRMSGF parameter
PRTBYPGM	Print messages by source member	<ul style="list-style-type: none"> ■ *YES: (default) A listing is produced ■ *NO: A listing is not produced, but the work files are not cleared (see notes)
PRTBYMSG	Print members by message identifier	<ul style="list-style-type: none"> ■ *YES: (default) A listing is produced ■ *NO: A listing is not produced
PRTNOREF	Print unreferenced messages from the file(s) specified on the USRMSGF and PMTMSGF parameters	<ul style="list-style-type: none"> ■ *YES: (default) a listing is produced ■ *NO: a listing is not produced
PRTMSGF	Print all messages from the file(s) specified on the USRMSGF and MSGF parameters, in ascending, first level text order	<ul style="list-style-type: none"> ■ *YES: (default) A listing is produced ■ *NO: A listing is not produced
IGNIMPMSG	Ignore impromptu messages	<ul style="list-style-type: none"> ■ *YES: (default) Impromptu messages are ignored. Only references to externally defined messages are documented ■ *NO: impromptu messages is included

Parameter	Definition	Value and Description
IGNMSGVAR	Ignore message variables	<ul style="list-style-type: none"> ■ *YES: (default) References to variables containing message identifiers is ignored ■ *NO: References to variables are included: for instance 'SNDPGMMSG MSGID(&MSGID)'
MBRLST	Qualified name of a member list	QTEMP/TEMPLST: (default) List name
EDIT	List edit option	<ul style="list-style-type: none"> ■ *NO: (default) No editing is required ■ *YES: Invoke the edit member list function to edit the list before processing

Notes

1. This command may take a long time to execute.
2. To identify which messages are used by a given program or command, the YDOCMMSGREF command searches the appropriate source for the message identifiers. It determines message identifier usage according to the source type, as follows:

- For CL program source, YDOCMMSGREF searches for use of the MSGID keyword in the following CL commands:

ADDMSGD, CHGMSGD, DSPMSGD, RTVMSG, SNDBRKMSG, SNDMSG, SNDNETMSG, SNDPGMMSG, SNDUSRMSG

For example:

```
SNDPGMMSG MSGID(CPF6801) MSGF(QCPFMSG) MSGDTA(&MSGDTA)
```

If the program fails to find a reference to a message identifier, it will scan for impromptu message text. For example, use of the MSG keyword (unless IGNIMPMSG(*NO) is specified):

```
SNDNETMSG MSG('System to close in 5 minutes') TOUSRID(&AUSER +
```

- For CMD source, YDOCMMSGREF searches for use of message identifiers in the PROMPT keyword in the following commands:

CMD, PARM, ELEM, QUAL, DEP

For example:

```

CMD      PROMPT(YVZ0101)
PARM     KWD(MBR) TYPE(*NAME) PROMPT(YVF1051)
PARM     KWD(FILE) TYPE(FLIB) PROMPT(YVF1001) FILE(*IN)
FLIB:    QUAL TYPE(*NAME) DFT(QTXTSRC)
          QUAL TYPE(*NAME) PROMPT(YVL0001) DFT(*LIBL)

```

If the program fails to find a reference to a message identifier, it will scan for impromptu message text, (unless IGNIMPMSG(*NO) is specified):

```

CMD      PROMPT('Edit Source')
PARM     KWD(MBR) TYPE(*NAME) PROMPT('Member name:')

```

- For RPG III source, YDOCMMSGREF searches for any of the following:
 - Use of a seven-character constant of the form XXXnnnn in factor two of an RPG III calculation specification line with an operation code of MOVE or PARM. For example:

C	CALL	'YYSNMSC'		60	
C	PARM	'CPF6801'	ZAMSID ?		Message Id.
C	PARM		ZAMSDA13Z		Message dat
.....					
C	MOVE	'CPF6801'	W?	?	Cmd cancel
.....					
C	MOVE	'USR0001'	MSGID ?		Cmd cancel
.....					
C*	MOVE	'CPF6801'	Command cancelled		

- Use of a variable called MSGID in the result field of an RPG III calculation specification line with an operation code of MOVE or PARM. For example:

C	MOVE @M,1	MSGID ?	Setup msg
C	MOVE LUX5412	MSGID ?	Setup msg
C	CALL 'VVSNSMC'	60	
C	PARM	MSGID ?	Message id
C	PARM	MSGDTA132	Message dat

- Use of the DSPLY operation code in an RPG III calculation specification line. YDOCMMSGREF will determine the message identifier according to the rules specified in the RPG III manual. For example:

C	*M1	DSPLY
C	W7	DSPLY

- For DDS Display file and Print file source, YDOCMMSGREF searches for use of the MSGCON, SFLMSGID or ERRMSGID keywords. For example:

A	6	MSGCON(19 YMN0001 Y1DEUPMT)
A		TEXT('** Start of menu **')

- For ALL source types, the command searches for comment lines containing references to message identifiers, indicated by the MSGID keyword. If IGNIMPMSG(*NO) is specified, the command searches for the use of impromptu messages, indicated by the MSG keyword.
- For CL, CMD or PLI source:

```

or          /* MSGID(CPF6801) Command cancelled */
or          /* MSGID(CPF0001) Error */
or          /* MSGID(US33001) MSGF(DEULIB/DEUMSGF) */
or          /* MSGID(USR0091) MSGF(DEUMSGF) MSGFLIB(DEULIB) */
or          /* MSGID('Hello sailor') */

```

- For RPG III or COBOL source:

```

* MSGID<USR0241> MSGF<QUSRMSG> MSGFLIB<PRLIB>
  MOVE 'USR0241' TO ERR-MESSAGE-ID
  PERFORM ZS-SND-PGM-MSG
.....
/* Exit to calling program: MSGID<RTNCDE>
  MOVE ERR-MESSAGE-ID TO W0-RTN-CODE
  PERFORM ZU-EXIT-PGM
.....
* Set return code: MSGID<&MSGID>
                      MOVE W0MSID      RTNCDE      Return code
.....
* MSGID<ERR1015> MSGF<DEULIB/ERRMSGF>
                      MOVE W0MSID      PRMSID      Error msg.
.....
* MSG<'Abnormal termination'>
                      MOVE @X,A        MSGTXT      Message txt

```

3. The YDOCMMSGREF command uses the i OS command Retrieve Message (RTVMSG) to obtain the first level text of messages. N.B. Use of the i OS command Override Message File (OVRMSGF) may cause it not to find the correct text.

The following rules are used to identify the message files containing the identifiers:

- For CL program source, the name of the message file containing any message identifiers is also extracted from the source statement. For example:

```
SNDPGMMSG MSGID<Y0L000Z> MSGF<YYYYMSG> MSGDTA<&MSGDTA>
```

Comment lines may be used to document use of message file names in variables, for example:

```

SNDPGMMSG MSGID<&MSGID> MSGF<&MSGF> MSGDTA<&MSGDTA>
.....
/* MSGID<USR1000> MSGF<PRODLIB/QUSRMSG> */
or      /* MSGID<USR1000> MSGF<QUSRMSG> */
or      /* MSGID<USR1000> MSGF<QUSRMSG> MSGFLIB<PRODLIB> */

```

- For CMD source, the name of the message file containing any message identifiers is determined from the PMTMSGF parameter on the YDOCMMSGREF command, except for parameter dependency checking when the default (USRMSGF) is used.

- For RPG III, PLI and COBOL source, the name of the message file containing any message identifiers is determined from the USRMSGF parameter on the YDOCMMSGREF command.
 - For all source types, the message file/library for messages referred to in comment lines may be specified explicitly.
4. The YDOCMMSGREF command creates the following outfiles in QTEMP.
- YDMSRFP - contains details of referenced messages found, based on the selection criteria specified.
 - YPUSMSP - contains the outfile data from the i OS command Display Message File (DSPMSGF). The file may have either one or two members depending on whether both message file parameters are used.

Output File	File	Mbr
Message references	YDMSRFP	YDMSRFP
Unreferenced messages (Created if LSTNOREF or LSTMSGF is *YES)	YPUSMSP	YPUSMSP
(Created if PMTMSGF not equal to USRMSGF)		YPUSMSP1
(Created if DEVMSGF not equal to USRMSGF)		YPUSMSP2

Example

To document all user messages found in all members from all source files beginning with 'Q' in production library PRLIB, using the user message file PRUSRMSGF:

```
YDOCMMSGREF SRCFILE(PRLIB/Q*) SRCMBR(*ALL) MSGID(U*) USRMSGF(PRUSRMSGF)
```

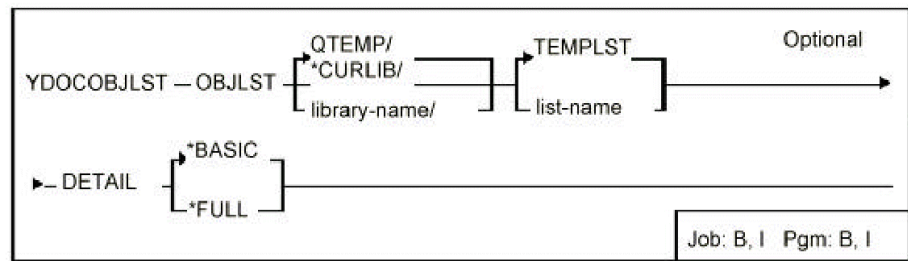

Chapter 3: CA 2E Commands, YD to YW

This chapter lists the various commands available with CA 2E from YDOCOBJLST to YWRUSRPRF.

YDOCOBJLST (Document Object List)

This command prints the contents of an object list.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
OBJLST	Qualified name of object list which is to be printed	QTEMP/TEMPLST (default) List name
DETAIL	Level of detail to print	<ul style="list-style-type: none">■ *BASIC: (default) Print only basic details about each object (name, library, type, text)■ *FULL: Print additional information about each object

Notes

None

Example

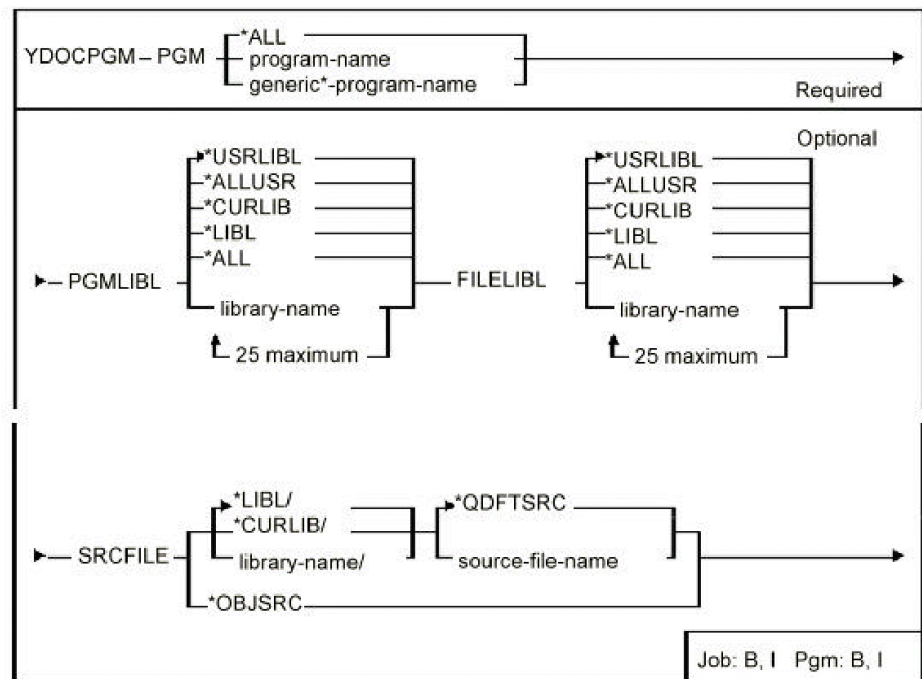
Prints the contents of list TEMPLST in library QTEMP:

```
YD0C0BJLST
```

YDOCPGM (Document Program command)

This command prints description of a program or programs: includes referenced objects and linkage details. Works for a command(s) and includes menu references.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
PGM	Qualified generic name of programs/commands that are to be documented	*ALL: Include all programs/commands specified by the library list
PGMLIBL	List of libraries to be used to locate programs which are to be documented	<ul style="list-style-type: none">■ *USRLIBL: (default) Use user portion of library list to locate programs/commands■ *LIBL: Use library list to locate programs■ *CURLIB: Use current library to locate programs■ *ALL: Use all libraries on machine■ *ALLUSR: Use all user libraries on machine to locate programs
FILELIBL	List of libraries to be used to locate referenced files to be included in documentation	<ul style="list-style-type: none">■ *USRLIBL: (default) Use user portion of library list to locate files■ *LIBL: Use library list to locate files■ *CURLIB: Use current library to locate file■ *ALL: Use all libraries on machine■ *ALLUSR: Use all user libraries on machine to locate files
SRCFILE	Qualified source file name for CL, RPG III and CMD source; used to obtain execution references and selected source comments	<ul style="list-style-type: none">■ *QDFTSRC: (default) The source file name defaults to QCLSRC, QRPGSRC or QCMDSRC depending on object type■ *OBJSRC: The source file name is taken from the source file name in the object description

Notes

1. This command takes a long time to execute: even if only a single program is specified the command must analyze the entire library to obtain information about where the program is called.
2. The report produced by the YDOCPGM command contains the following sections for each program, command, or menu:
 - Object header details and compiler options:
Compiler options as specified by Z*or /*Z: comment lines in source.
 - Synopsis/Function narrative:
As specified by H*or /*H:comment lines in source.
 - File usage:
 - Files and data areas called by the program.
 - Compile-time file overrides as specified by Y*or /*Y:comment lines in source.
 - Linkage details:
 - Entry parameters
 - Objects and menus called by the program
 - Objects and menus that call the program.
 - Maintenance notes:
As specified by M*or /*M:comment lines in source.
 - Warning notes:
As specified by W*or /*W:comment lines in source.
3. When run as part of an interactive job, the YDOCPGM command may not be used to document libraries that are in the system part of the job's library list. If you wish to document such libraries interactively, first use the i OS command Change System Value (CHGSYSVAL) to remove the libraries from the system part of the job's library list.
4. When run as part of a batch job, the YDOCPGM command always removes all libraries except QSYS from the system part of the job's library list. Thus the command may be used to document libraries that are in the system part of the library list. Note however, that if you use the YDOCPGM command in your own programs, you may want to restore the system part of the library list after invoking the command.

Example

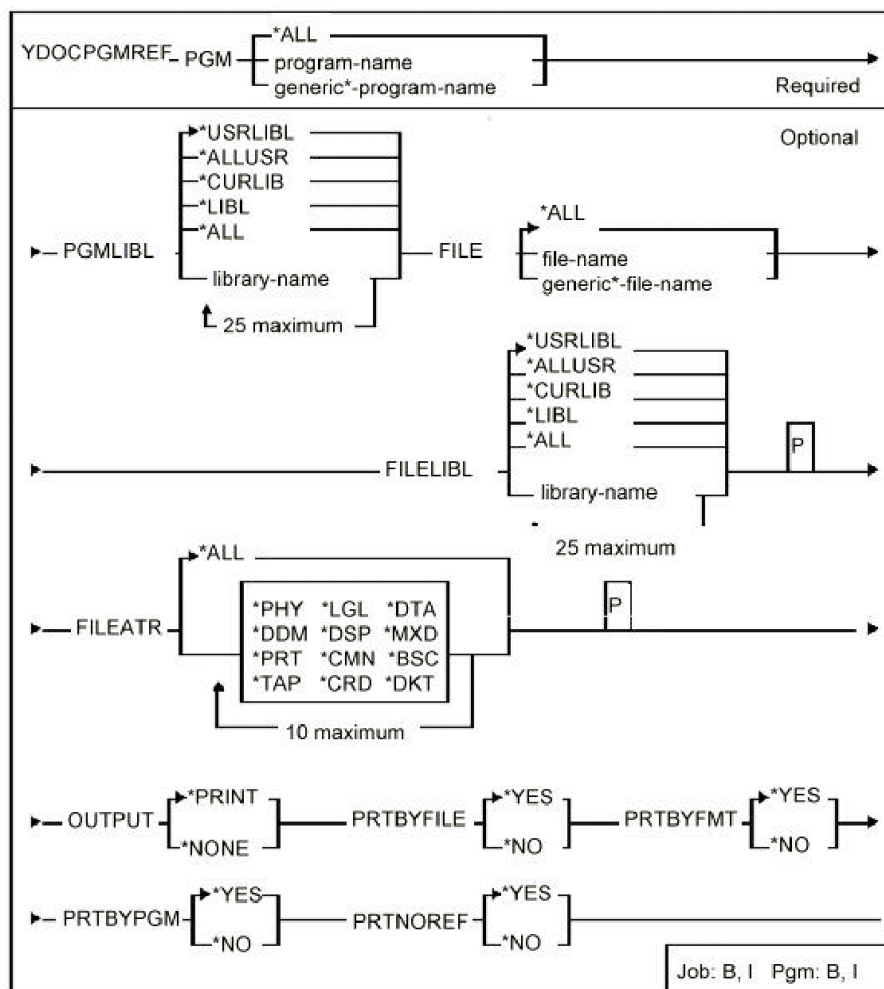
To document all programs in library APPPGM and APPDEV, using file library APPFIL. Source is in APPSRC:

```
YDOCPGM PGM(*ALL) PGMLIBL(APPGM APPDEV) FILELIBL(APPFIL) SRCFILE(APPSRC/*QDFTSRC)
```

YDOCPGMREF (Document Program References)

This command prints cross-references of usage data for a specified group of files and programs.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
PGM	Generic name of programs whose references are documented	*ALL: (default) Includes all programs in specified program library list, as specified by the PGMLIBL parameter
PGMLIBL	List of libraries to be used to locate programs that are included in cross-reference	<ul style="list-style-type: none"> ■ *USRLIBL: (default) Use user portion of library list to locate programs ■ *LIBL: Use library list to locate programs ■ *CURLIB: Use current library to locate programs ■ *ALL: Use all libraries on machine ■ *ALLUSR: Use all user libraries on machine to locate programs
FILE	Generic name of files that are cross-referenced	*ALL: (default) Include in listing all files in specified file library list, as specified by the FILELIBL parameter
FILELIBL	List of libraries used to locate files and other reference objects that are to be included in cross-reference	<ul style="list-style-type: none"> ■ *USRLIBL: (default) Use user portion of library list to locate files ■ *LIBL: Use library list to locate files ■ *CURLIB: Use current library to locate files ■ *ALL: Use all libraries on machine ■ *ALLUSR: Use all user libraries on machine to locate files
FILEATR	List of attributes of files that are cross-referenced	<p>*ALL: (default) List all file types.</p> <p>Must be a valid i OS file type, or *DTA: list data areas</p>

Parameter	Definition	Value and Description
OUTPUT	Output required	<ul style="list-style-type: none">■ *PRINT: (default) Printed output is to be produced■ *NONE: No printed output is required but the work files are not cleared
PRTBYFILE	Print programs by referenced file	<ul style="list-style-type: none">■ *YES: (default) Listing is produced■ *NO: Listing is not produced
PRTBYFMT	Print programs by referenced format	<ul style="list-style-type: none">■ *YES: (default) Listing is produced■ *NO: Listing is not produced
PRTBYPGM	Print files/formats by referenced program	<ul style="list-style-type: none">■ *YES: (default) Listing is produced■ *NO: Listing is not produced
PRTNOREF	Print files included by the file selection criteria which are not referenced by the programs included in the program selection criteria	<ul style="list-style-type: none">■ *YES: (default) Listing is produced■ *NO: Listing is not produced

Notes

1. This command takes a long time to execute.
2. If values other than *ALL are specified for the FILE or the FILEATR parameters, than a partial listing is produced. Only those files included by the selection criteria are listed, and any program for which no references remain is dropped from the listing.

3. When a file or data area attribute is shown as *??? on the report, the object definition was not available via the file selection criteria.
4. When OUTPUT(*NONE) is specified the following work files are created in library QTEMP:

Output file	File	Mbr
Program references	YDPGRFP	YDPGRFP
Program references by referenced file if PRTBYFIL(*YES))	YDPGRFL1	YDPGRFL1
Program references by referenced format if PRTBYFMT(*YES))		

Example

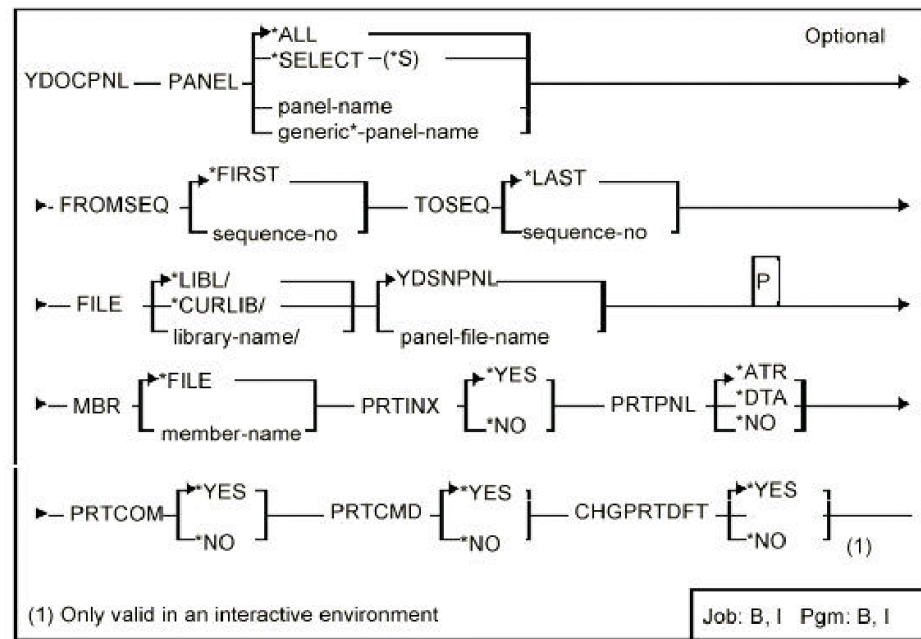
To document all programs in library XXXAPP, obtaining file references from libraries XXXAPP and XXXDTA:

```
YDOCPGMREF PGM(*ALL) PGMLIBL(XXXAPP) FILE(*ALL) FILELIBL(XXXAPP XXXDTA)
```

YDOCPNL (Document Panel Design)

This command prints panel designs. The panel designs are printed in sequence number order.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
PANEL	Generic name of panel designs which are documented	<ul style="list-style-type: none"> ■ *ALL: (default) All panel designs in the specified file and member are to be documented ■ *SELECT: Provide a display of available panel designs in the specified file, one of which may be selected

Parameter	Definition	Value and Description
FROMSEQ	Print sequence number (999.99) of first panel design which is documented	*FIRST: (default) Start listing from first panel
TOSEQ	Print sequence (999.99) number of last panel design which is documented	*LAST: (default) Finish listing at last panel design
FILE	Qualified name of file containing panel designs	*LIBL/YDSNPNL: (default)
MBR	Name of member in panel design file	*FILE: (default) The member has the same name as the file
PRTINX	Produce an index of panel designs	<ul style="list-style-type: none"> ■ *YES: (default) An index is printed ■ *NO: No index is to be printed
PRTPNL	Document the panel design's images	<ul style="list-style-type: none"> ■ *ATR: default) The panel images are to be documented with fields shown by the field representation characters. ■ *DTA: The panel images are documented with user data - data may entered using the command Display Panel Design (YDSPPNL) with OPTION(*CHGDTA). Fields where no data has been keyed is printed as blank. ■ *NO: No panel images are documented
PRTCOM	Document the panel design's comments	<ul style="list-style-type: none"> ■ *YES: (default) The panel comments are documented ■ *NO: No panel comments are documented

Parameter	Definition	Value and Description
PRTCMD	Document the panel design's command keys	<ul style="list-style-type: none">■ *YES: The command key specifications are documented■ *NO: No command key specifications are documented
CHGPRTDFT	Change print defaults	<ul style="list-style-type: none">■ *SAME: (default) The print defaults are not changed■ *TEMP: The print defaults are changed for the current operation only■ *PERM: The print defaults are changed permanently <p>N.B. Only *SAME may be used for batch jobs</p>

Notes

1. Panel designs are printed in the order of the print sequence numbers entered on the panel design header display. Panel designs with a sequence number of 999.99 are not printed.
2. A sequence range and generic panel design name may both be specified and will act cumulatively - for a panel design to be printed, it must satisfy both selection parameters.

Example

To print all panel designs with names starting with the letters GL in file YDSNPNL in library MYLIB, and having print sequence 100.00 to 100.99:

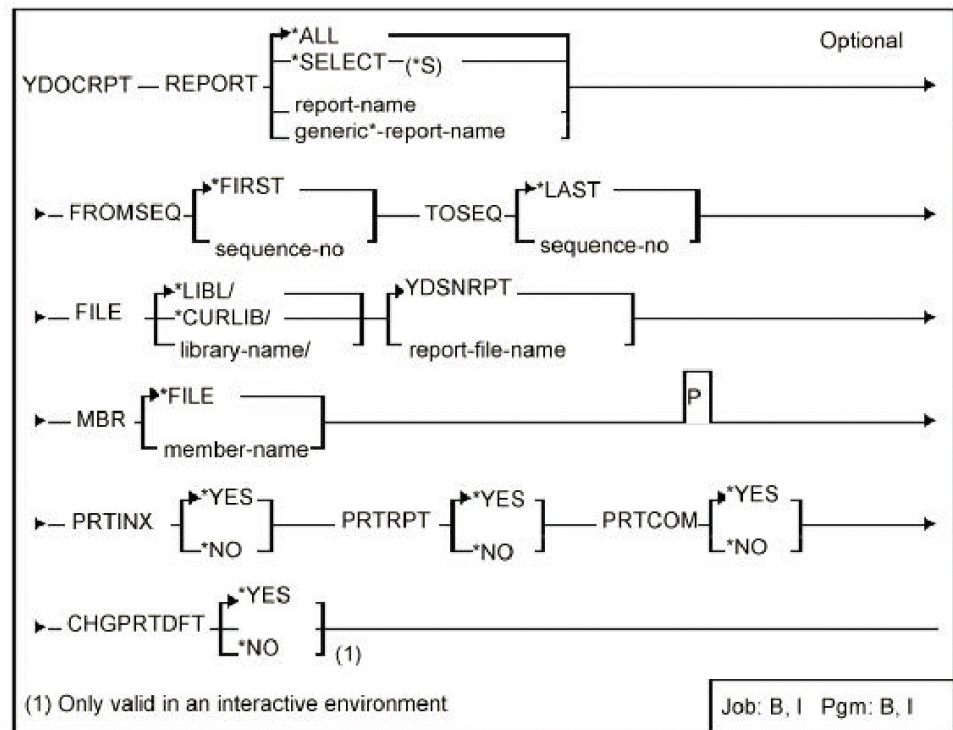
```
YDOCPNL PANEL(GL*) FROMSEQ(100.00) + TOSEQ(100.99) FILE(MYLIB/YDSNPNL) +  
PRTPNL(*DTA) PRTCMD(*NO)
```

The index of panel designs will be printed, the panel designs will contain merged user data, the panel design comments will be documented, and the command key specification will be suppressed

YDOCRPT (Document Report Design)

This command prints report designs in sequence number order.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
REPORT	Generic name of report designs which are documented	<ul style="list-style-type: none"> ■ *ALL: (default) All report designs in the file are documented. ■ *SELECT: Provide a display of report designs in the specified file, one of which may be selected

Parameter	Definition	Value and Description
FROMSEQ	Sequence number (999.99) of first report design which is documented	*FIRST: (default) Start listing from first report
TOSEQ	Sequence number (999.99) of last report design which is documented	*LAST: (default) Finish listing at last report design
FILE	Qualified name of file containing report designs	YDSNRPT: (default) Report design file name The file must have been created using the command Create Design File YCRTDSNF) with parameter TYPE(*RPT)
MBR	Name of member in report file	*FILE: (default) The member has the same name as the file
PRTINX	Produce an index of report designs	<ul style="list-style-type: none">■ *YES: (default) An index is to be printed■ *NO: An index is not to be printed
PRTRPT	Document the report design's images	<ul style="list-style-type: none">■ *YES: (default) The report images are to be documented■ *NO: Do not print the report design images
PRTCOM	Document the report design's comments	<ul style="list-style-type: none">■ *YES: The report comments are to be documented■ *NO: Do not print the comments
CHGPRTDFT	Change print defaults	<ul style="list-style-type: none">■ *SAME: (default) The print defaults are not to be changed■ *TEMP: The print defaults are to be changed for the current operation only■ *PERM: The print defaults are to be changed permanently <p>N.B. only *SAME may be used for batch jobs</p>

Notes

Report designs are printed in the sequence numbers entered on the report header screen. Report designs with a sequence number of 999.99 are not printed. A sequence range and a generic report design name may both be specified.

Example

To print all report designs with names starting with the letters GL in file YDSNRPT in library MYLIB, and having print sequence 100 to 100.99:

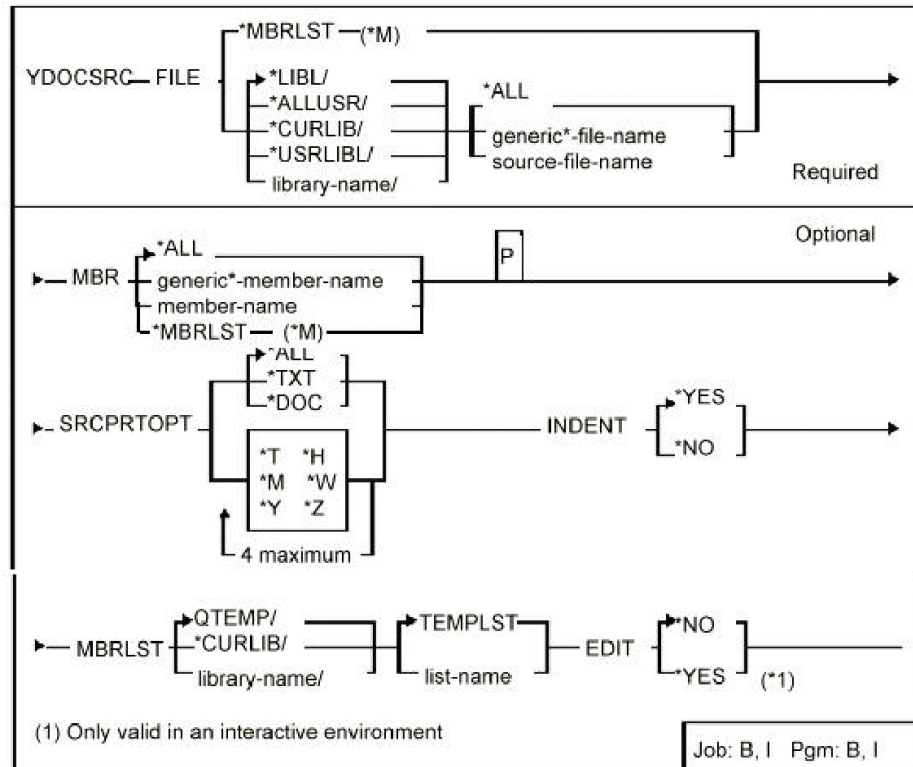
```
YDOCRPT REPORT(GL*) FROMSEQ(100.00) TOSEQ(100.99) FILE(MYLIB/YDSNRPT) PRTTXT(*NO)
```

The report design index will be printed, the report designs will be printed, and the report design comments will be suppressed

YDOCSRC (Document Source File Members)

This command documents specified source members and an index is included.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
FILE	Qualified generic name of source file containing the source members which are documented	<ul style="list-style-type: none"> ■ *MBRLST: The source members are located by an existing member list. ■ *ALL: All source files

Parameter	Definition	Value and Description
MBR	Generic name of members which are documented	Unless FILE(*MBRLST) is specified, the MBR parameter is used to filter the list of members given by the FILE parameter *ALL: (default) Document all source members
SRCPRTOPT	Source print option. Single value	<ul style="list-style-type: none"> ■ *ALL: (default) Print all source lines in each member ■ *TXT: Print as text document: omit source line number and change date ■ *DOC: Print only source lines with a documentation comment type (*T, *H, *W, *M, *Y, *Z) ■ Or a list of up to four of the following: <ul style="list-style-type: none"> – *T: Print only source lines with a title line – *H: Print only source lines with a header comment line – *W: Print only source lines with a warning comment line – *M: Print only source lines with a maintenance comment line – *Y: Print only source lines with a pre-compile directive line – *Z: Print only source lines with a compiler directive line
INDENT	Indent RPG III source option	<ul style="list-style-type: none"> ■ *NO: (default) Do not indent RPG source listings ■ *YES: Indent any listings of RPGIII source to indicate the structured programming constructs

Parameter	Definition	Value and Description
MBRLST	Qualified name of a member list specified that are documented. To use an existing list, specify FILE(*MBRLST)	QTEMP/TEMPLST: (default) Member list name
EDIT	Edit member list	<ul style="list-style-type: none">■ *NO: (default) Do not invoke the edit list function■ *YES: Invoke the edit member list function to review the member list before listing the members in it

Notes

If a value other than *ALL is specified for the SRCPRTOPT parameter, then the YDOCSRC utility will only print the comment lines from the source that meet the specified types.

Example

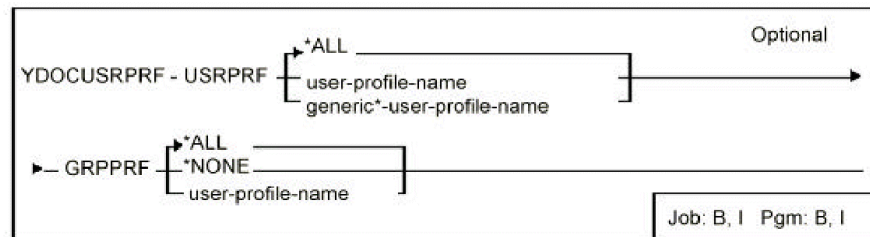
To print all source members in list SAUCY:

```
YDOCSRC FILE(*MBRLST) MBRLST(SAUCY)
```

YDOCUSRPRF (Document User Profile)

This command prints details of user profiles. i OS profiles for which no data is held are also listed.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
USRPRF	Generic user profile name of profiles which are to be documented	*ALL: (default) Print details of all user profiles
GRPPRF	Name of group profile of user profiles which are documented	<ul style="list-style-type: none"> *ALL: (default) Print details of all user profiles *NONE: Print details of all user profiles which do not belong to a group profile

Notes

Only those profiles to which the user has read authority is printed.

Example

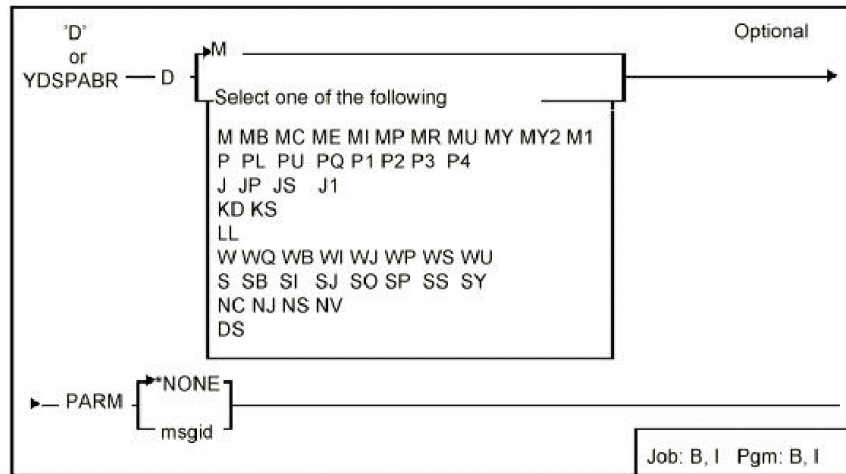
To print details for user profile TRON:

```
YDOCUSRPRF USRPRF(TRON)
```

YDSPABR (Display Abbreviated)

This command provides an abbreviated means of specifying most of the commonly required display functions.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
D	Type of display option	<ul style="list-style-type: none"> ■ M - messages <ul style="list-style-type: none"> – M: DSPMSG – MB: WRKMSGD MSGID (parm value) QCBL/QCBLMSG – MC: WRKMSGD MSGID (parm value) *LIBL/QCPFMSG – ME: WRKMSGD MSGID (parm value) *LIBL/QRPGMSGE – MI: WRKMSGD MSGID (parm value) QIDU/QIDUMSG – MP: WRKMSGD MSGID (parm value) *LIBL/YCMDPMT – MR: WRKMSGD MSGID (parm value) QRPG/QRPGMSG – MU: WRKMSGD MSGID (parm value) *LIBL/Y1USRMSG – MY: WRKMSGD MSGID (parm value) *LIBL/YYYYMSG – MY2: WRKMSGD MSGID (parm value) *LIBL/Y2MSG

Parameter	Definition	Value and Description
		<ul style="list-style-type: none">■ P - print queues<ul style="list-style-type: none">- P : WRKOUTQ- PL: WRKOUTQ (Output queue)- PQ: WRKOUTQ QGPL/QPRINT- P1: WRKOUTQ *LIBL/QPRINTP2: WRKOUTQ *LIBL/QPRINT2P3: WRKOUTQ *LIBL/QPRINT3- P4: WRKOUTQ *LIBL/QPRINT4■ J - job queues<ul style="list-style-type: none">- J : WRKJOBQ- JP: WRKJOBQ *LIBL/QPGMR- JS: WRKJOBQ QGPL/QSPL- J1: WRKJOBQ *LIBL/QBATCH■ K - diskette<ul style="list-style-type: none">- KD: DSPDKT DKT01- KS: DSPDKT DKT01 DATA(*SAVRST)

Parameter	Definition	Value and Description
		<ul style="list-style-type: none"> ■ L - library lists. <ul style="list-style-type: none"> - LL: WRKLIB ■ W - display job/submitted job. <ul style="list-style-type: none"> - W : WRKSBMJOB *WRKSTN - WA: WRKACTJOB - WB: WRKACTJOB SBS(QBATCH) - WI: WRKACTJOB SBS(QINTER) - WJ: WRKSBMJOB *JOB - WP: WRKACTJOB SBS(QPGMR) - WS: WRKJOB - WU: WRKSBMJOB *USER ■ S - subsystem & system status. <ul style="list-style-type: none"> - S : WRKSBS - SB: WRKSBSJOB QBATCH - SI: WRKSBSJOB QINTER - SJ: WRKSBSJOB *JOBQ - SO: WRKSBSJOB *OUTQ - SP: WRKSBSJOB QPGMR - SS: WRKSBSJOB QSPL - SY: WRKSYSSTS ■ N - display menu. <ul style="list-style-type: none"> - NC: GO CMDGRP - NJ: WRKJRN - NS: GO SUBJECT - NV: GO VERB ■ D - configuration status <ul style="list-style-type: none"> - DS: WRKCFGSTS *DEV

Notes

1. The command may be specified in an abbreviated form: D.
2. For the D PL option, the name of the output queue containing the job logs is retrieved from a data area: YPLGOQA in the library list.

- For the D Mn options, a partial value may be specified for the MSGID parameter; for example, a value of CPD causes i OS messages beginning at CPD0000 to be displayed.

Examples

To display CPF messages beginning at CPF9801:

```
D MC CPF9801
```

To display CPF messages beginning at CPF9000:

```
D MC CPF9
```

To display your user profile's message queue:

```
D
```

To work with all jobs submitted by the workstation:

```
D W
```

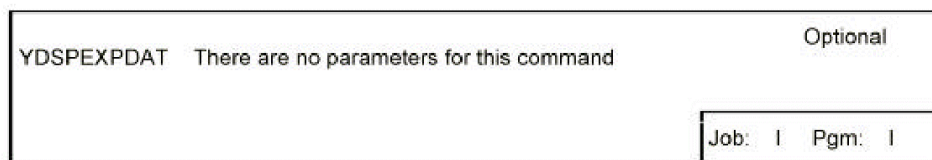
To work with the print queue QPRINT in library QGPL:

```
D PQ
```

YDSPEXPDAT (Display Expiry Date)

This command displays authorization data for your version of the utilities, including the expiry date and the serial number of the authorized machine.

Syntax Diagram



Parameters

None.

Notes

This command displays the expiry date if you have a trial copy of CA 2E.

Example

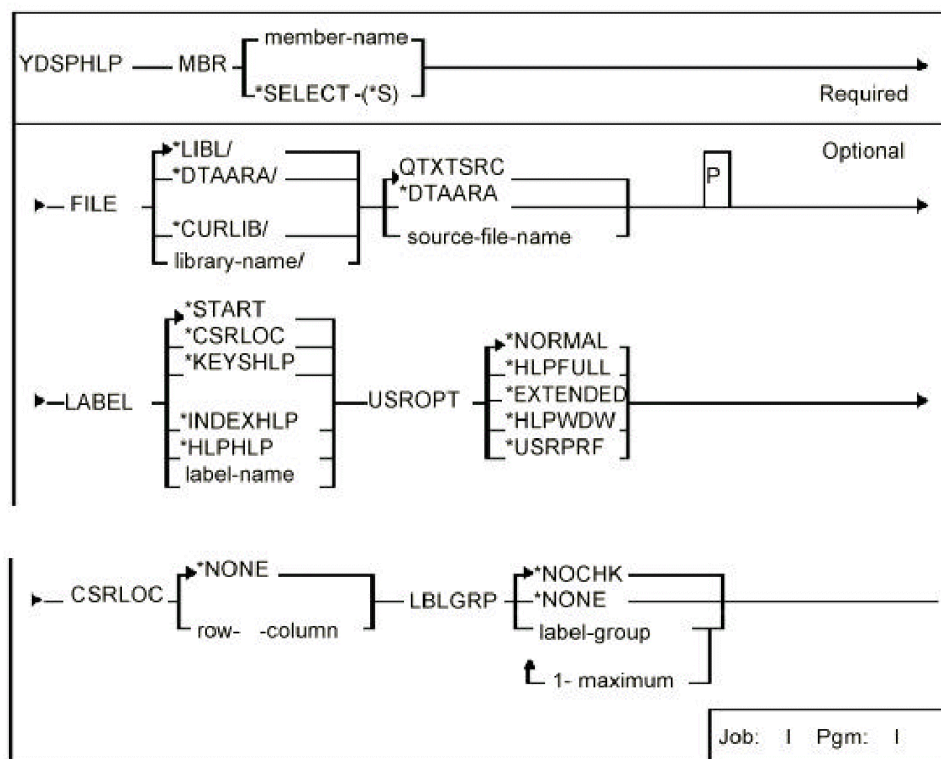
To display the authorization data for your copy of CA 2E:

```
YDSPEXPDAT
```

YDSPHLP (Display Help Text)

This command displays a source file member as help text. Additional pages of help text can be displayed using the ROLL keys. The control characters of the Text Management/38 utility can be used to format the text. A cursor location can be passed to the help display program, which uses a table of field locations (vector table) at the start of the text to determine at which point the help display is to commence. The vector table may be added manually, or by means of the YADDHLPTBL command.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
MBR	Name of source file member containing text which is to be displayed	*SELECT: A list of available members will be displayed, one of which may be selected
FILE	Qualified name of source file containing help text member.	<ul style="list-style-type: none"> *LIBL/QTXTSRC: (default) Text file name *DTAARA: Text file name is retrieved from data areas YMHPFLA (file name), and/or YMHPPLBA (library name)

Parameter	Definition	Value and Description
LABEL	Help text label at which the help text display is to commence	<ul style="list-style-type: none"> ■ *START: (default) Display from the beginning of the Help text (extended help when USROPT is *HLPWDW) ■ *CSRLOC: The CSRLOC parameter is to determine the label at which the help text display is to commence ■ *KEYSHLP: If USROPT is *HLPWDW, display Keys help defined in the help document ■ *INDEXHLP: If USROPT is *HLPWDW, display the help Index defined in the help document ■ *HLPHLP: If USROPT is *HLPWDW, display help for help
USROPT	Display option	<ul style="list-style-type: none"> ■ *NORMAL: (default) Display help with USROPT specified in data area YMHPOPA. If data area is not found, display with *HLPFULL ■ *HLPFULL: Display all help in full panels ■ *USRPRF: Retrieve USROPT from the i OS User Profile. If USROPT(*HLPFULL) is not specified on the i OS User Profile, the default is to display Help in a window
CSRLOC	Cursor location, entered as two values, row and column coordinates, which is used in conjunction with the LBLGRP parameter to derive a start position for displaying the Help text	*NONE: (default) No cursor location specified

Parameter	Definition	Value and Description
LBLGRP	List of up to ten label group names. Label group names are single characters, either alphabetic or numeric. This parameter is used when more than one label is appropriate for a particular cursor location	<ul style="list-style-type: none">■ *NOCHK: (default) Any label group conditioning specified in the help text vector table is to be ignored.■ *NONE: Any entries specified in the vector table in the help text will be ignored unless either the entry has no label group specified for it or the entry only has label group names prefixed by N.■ Character values: Only those vector table entries with label groups matching the supplied values are used

Notes

1. This command calls an interactive program to display the help text. For instructions on how to use the help utility, press HELP while using the program. When USROPT is *HLPWDW, the F1 key is also available for instructions.
2. If a value of *LIBL is specified for the help text file library, the help display program uses the invoking job's library list to find the text member: the first file containing a source member of the specified name is used. See the section on help text in the *CA 2E Concepts Guide* for further details.
3. Help text can be entered using either the Start SEU utility (STRSEU) or the Text Management/38 Utility (EDTTXT)-the latter is preferable. Text Management/38 is necessary for highlighting functions such as underline, and high intensity.

4. The YDSPHLP command interprets Text Management/38 control characters in the same manner as the print functions of Text Management/38, except in respect to the following points:

Symbol	-	YDSPHLP action
. *	-	Line is ignored except for: . *T: - Title line . *YV: AAAAABBBCCC CCCEEE label . *YI: XXXXXXXXXXXX label . *Y*: Index display only line. . *YH: XXXXXXXXXXXX label
. PA	-	Blank line : must be on a line by itself.
. TC	-	Ignored
. SK	-	Ignored
. PN	-	Ignored
. KP	-	Ignored

The help text directives are described in more detail in the *CA 2E Concepts Guide*.

5. The Help text facility can be called directly from within a HLL program, either by using the i OS QCMDEXC program to execute the YDSPHLP command, or by calling program YDDSHPR.

As shown in the following example, DDS source that enables the HELP key, and assign an indicator to the HELP key, appears as follows:

```

A                                HELPC25 'HELP KEY'>

```

The following example of RPG III source calls the help display program if the HELP key is pressed -without cursor-sensitive help. Help text will be displayed starting at the beginning of the member (or using the index, if one is present in the member).USROPT *HLPWDW results in the CUA Extended Help display:

```

* Display panel
C          EXFMT#DISPLAY
*.....
* If HELP key is pressed, display Help text
C 25      CALL 'YDDSHPR'          60  Display hlp
C          PARM          ##MBUN 10      I:MBR NAME
C          PARM          ##FLUN 10      I:FILE NAME
C          PARM          ##LBUN 10      I:LIB NAME
C          PARM *BLANK    ##RTUM 7      O:RETURN CD
* Return code values: 'YHP0003' : HELP TEXT NOT FOUND
*                    'YHP0002' : HELP TEXT DISPLAYED

```

The following example shows RPG III source to call the help display program if the HELP key is pressed -with an explicit label. Help text is displayed starting at the specified label (FRED) in the member. This results in a window when USROPT is *HLPWDW.

```

* Display panel
C          EXFMT#DISPLAY
*.....
* If HELP key is pressed, display Help text
C 25      CALL 'YDDSHPR'          60  Display hlp
C          PARM          ##MBUN 10      I:MBR NAME
C          PARM          ##FLUN 10      I:FILE NAME
C          PARM          ##LBUN 10      I:LIB NAME
C          PARM *BLANK    ##RTUM 7      O:RETURN CD
* Parameters after this point are optional
C          PARM 'FRED'    ##LBL 10      I:START POS

```

The following example shows RPG III source to call the Help display program if Help is pressed -with cursor-sensitive Help and label group conditioning. The vector table in the Help text will be used to determine the starting point for the Help display. Note that the cursor location is obtained from a file information feedback data structure associated with the display file:

```

FUUDSCUR#CF  E                                WORKSTN
F                                                    KINFDS INFDS#
*
IINFDS#      DS
I                                                    B 370 3710##RWCL

```

```

* Set Help label group conditioning
C          *IN09      IFEQ '1'
C                      MOVEL 'A'          ##LGRP
C                      ELSE
C                      MOVEL 'B'          ##LGRP
C                      END
C                      Z-ADD1          ##LGCT  50      Label count
*.....
* Display panel
C          EXFMT#DISPLAY
*.....

```

```

* IF HELP key is pressed, display Help text
C 25      DO
* Calculate row and column
C          ##RWCL      DIU 256      WOHPRW  50      Row
C                      MUR          WOHPCL  50      Col
*
C          CALL 'YDDSHPR'          60      Display hlp
C          PARM          ##MBUN 10      I:MBR NAME
C          PARM          ##FLUN 10      I:FILE NAME
C          PARM          ##LBUN 10      I:LIB NAME
C          PARM *BLANK      ##RTUM 7      O:RETURN CD
* Parameters after this point are optional
C          PARM '*CSRLOC' ##LBL 10      I:START POS
C          PARM '*NORMAL' ##USOP 9      I:USER OPT
C          PARM WOHPRW      ##ROW 50      I:CURSR ROW
C          PARM WOHPCL      ##COL 50      I:CURSR COL
C          PARM          ##LGCT 50      I:LBLGR CNT
C          PARM          ##LGRP 10      I:LBL GRPS
* Return code values: 'YHP0003' : HELP TEXT NOT FOUND
*                   'YHP0002' : HELP TEXT DISPLAYED
C                      END

```

The return codes are the message identifiers of the appropriate message in the shipped user message file, Y1USRMSG.

You may find it convenient to give the help text member the same name as the program: the member name can then be retrieved from the program status data structure.

If LBLGRP(*NOCHK) is required then \$\$LGCT should be zero and \$\$LGRP should be blank. If LBLGRP(*NONE) is required then \$\$LGCT should be set to -1 (minus one) and \$\$LGRP should be blank. If a list of label groups is specified then \$\$LGCT should contain the number of label groups (1-10) and \$\$LGRP should contain the label groups (left-justified).

The first four parameters are required. The remaining six are optional. If you specify LABEL(*CSRLOC) (parameter \$\$LBL), you must specify both row and column values and a display option. If you specify a label group, you must specify the number of label groups (\$\$LGCT) and a cursor location.

If none of the optional parameters are passed into the YDDSHPR program, default values (as supplied for the command YDSPHLP) are supplied by the program.

6. Because the HELP key is defined as a command action key, no data will be returned to a program when the HELP key is pressed, unless you specifically read the display format after detecting the HELP key. If you do not explicitly read the data, the data will be lost. The following DDS entries will ensure that the data do not have to be re-keyed:

DDS source additions are as follows:

- Create display file with RSTDSP(*YES). Do not display subfile records following return from HELP; for example, .condition SFLDSP by Help key indicator off.
- Include PUTOVR line at record level for non-subfile record formats, and condition this line by the HELP key indicator.

- If you use a message subfile to clear the screen before writing other formats (using the OVERLAY keyword), then include the OVERLAY keyword in the specifications for the message subfile control record - and condition it with the HELP key indicator:

```

1. →/*Z: CRTDSPF DFRWRT(*YES) RSTDSP(*YES)
A                                     HELP(25 'HELP KEY.')
A*=====
A      R #SFRC#1                      SFL
... SUBFILE RECORD.
A*=====
A      R #SFCT#1                      SFLCTL(#SFRC#1)
A                                     BLINK OVERLAY
A                                     SFLPAG(18) SFLSIZ(19)
A                                     INDXT(50 'CLEAR SUBFILE')
A                                     INDXT(51 'DSPLY SUBFILE')
A                                     SFLCLF
A 50                                  SFLDSPCTL
A M50                                SFLDSP
2. →A M50 51N25
3. →A 25                             PUTOUT
A*=====
...
* Error messages subfile Control
A      R #SFCT#Q                      SFLCTL(#SFRC#Q)
4. →A 25                             OVERLAY
A*=====

```

- The Text Management/38 edit function may be invoked directly from the Help display. If you wish to prevent a user from updating Help text, you should ensure that he or she does not have update rights to the Help text source file. For example:

```

RVKOBJAUT OBJ(file-name)OBJTYPE(*FILE)
USER(user-name)AUT(*ADD *UPD *DLT)

```

- You can use the Display Help function without CA 2E being present in your library list. See Appendix B for more information.

Example

To display document AUSECOURS from QTXTSRC starting at the first page:

```
YDSPHLP MBR(AUSECOURS) FILE(QTXTSRC)
```

To display document AUSECOURS from QTXTSRC starting the Help display at the entry corresponding to a cursor position of row five, column fifteen:

```
YDSPHLP MBR(AUSECOURS) FILE(QTXTSRC) LABEL(*CSRLC) CURSOR(5 15)
```

To display document AUSECOURS from QTXTSRC starting the Help display at the entry belonging to label group 'X' and corresponding to a cursor position of row five, column fifteen:

```
YDSPHLP MBR(AUSECOURS) FILE(QTXTSRC) LABEL(*CSRLOC) CURSOR(5 15) LBLGRP(X)
```

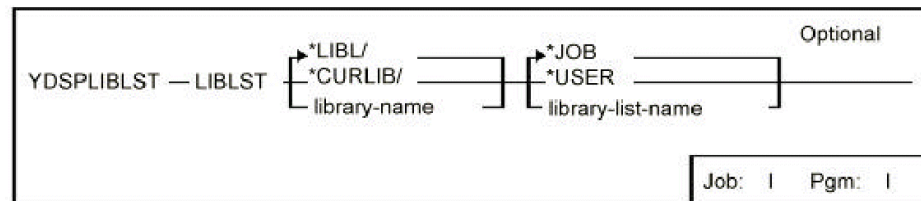
To display the Keys Help window for document STDKEYS in file and library stored in data areas YMHPFLA and YMHPLBA:

```
YDSPHLP MBR(STDKEYS) USROPT(*HLPWDW) LABEL(*KEYSHLP)
```

YDSPLIBLST (Display Library List)

This command displays a stored library list.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
LIBLST	Qualified name of library list which is displayed	*JOB: (default) Display list with same name as current job *USER: Display list with same name as the current user profile

Notes

Library lists are stored in file YLIBLST in the library specified by the **LIBLST** parameter.

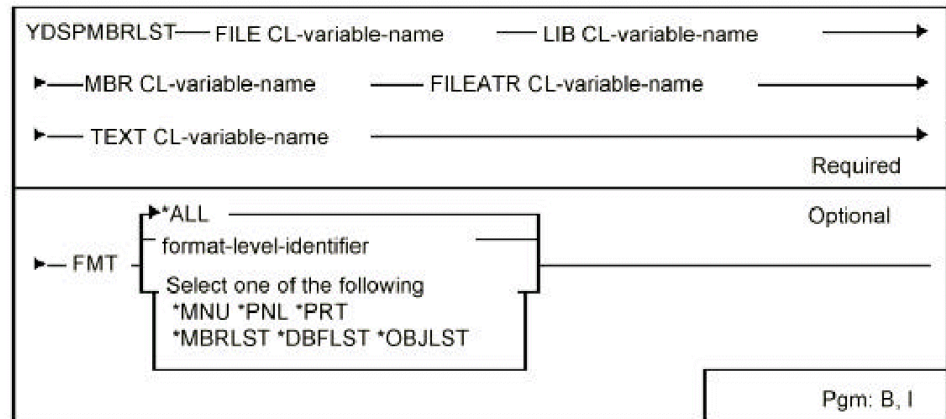
Example

To display library list DEWEY:

```
YDSPLIBLST LIBLST(DEWEY)
```

YDSPMBRLST (Display Member)

This command displays a list of members within one or more files, and allows one of the displayed member names to be selected and returned to the calling program.

Syntax Diagram

Parameters

Parameter	Definition	Value and Description
FILE	CL variable that (1) contains the name of the file(s) whose members are displayed, and (2) returns the name of the file selected by the user	<ul style="list-style-type: none">■ Character variable, ten bytes long■ Generic names are allowed■ *ALL: (default) display all files in the specified library
LIB	CL variable that (1) contains the name of the library containing the file(s) displayed - or a special value, and (2) returns the name of the library containing the file selected by the user Character variable, ten bytes long	<ul style="list-style-type: none">■ *LIBL: (default) display the first files found with the specified name in the current job's library list■ *USRLIBL: display the first files found with the specified name in the user part of the current job's library list■ *ALLUSR: display all files found with the specified name in all user libraries■ *ALL: display all files found with the specified name in all libraries
MBR	CL variable that (1) contains the name of the member(s) to be displayed, and (2) returns the name of the member selected by the user. Generic names and name masks are allowed Character variable, ten bytes long	<ul style="list-style-type: none">■ *ALL: (default) selects all files in the specified library
FILEATR	CL variable that (1) contains the attribute of the file to be displayed and (2) returns the attribute of the file selected by the user Character variable, four bytes long, restricted to the following two values	<ul style="list-style-type: none">■ *PHY:(default) display physical files■ *LGL: display logical files

Parameter	Definition	Value and Description	
TEXT	CL variable that (1) contains the title to be displayed on the second line of the screen and (2) returns the text associated with the member selected by the user Character variable, fifty bytes long If left blank, the title displayed will default to Display Member List *FMT: (default) use the appropriate default text for the format type specified by the FMT parameter	Fmt	MSG ID
		*ALL	YML0022
		*MNU	YMN0047
		*RPT	YRD0037
		*PNL	YSD0037
		*OBJLST	YOL0025
		*MBRLST	YML0021
		*DBFLST	YDL0021
FMT	Format level identifier of files to include in display. Only character variable, thirteen bytes long, corresponding to the i OS format level id	<ul style="list-style-type: none"> ■ *ALL: (default) display files regardless of format id ■ *MNU: display only menu files ■ *PNL: display only panel design files ■ *RPT: display only report design files ■ *OBJLST: display only object list files ■ *MBRLST: display only member list files ■ *DBFLST: display only database file list files 	

Notes

1. The command creates a work member called YDSPMBRLST in file QTEMP/YMBRLST. Any existing version of the file will be overridden.
2. A name mask is used to specify which names are to be selected. Refer to the description of the Filter Database File List command (YFLTDBFLST) for details of name masks and examples of their use.

Example

To make a CL program display a list of members in any source files in QGPL whose names begin with the letters Q*, so that a member name can be selected by the user and returned to the program:

```
DCL &FILE TYPE(*CHAR) LEN(10) 'Q*' /* File name */

DCL &LIB TYPE(*CHAR) LEN(10) 'QGPL' /* Library name */

DCL &MBR TYPE(*CHAR) LEN(10) '*ALL' /*Member Name*/

DCL &FILEATR TYPE(*CHAR) LEN(4) '*ALL' /* File attribute */

DCL &TEXT TYPE(*CHAR) LEN(50) 'Select a member' /* Text */

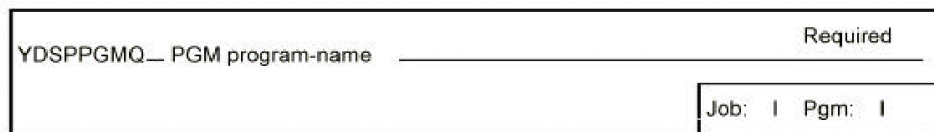
YDSPMBRLST FILE(&FILE) LIB(&LIB) MBR(&MBR) FILEATR(&FILEATR ) TEXT (&TEXT) /*
Provider selection */

CALL USERPROG(&FILE &LIB &LIB &TEXT)      /* Call usr pgm */
```

YDSPPGMQ (Display a Program's Message Queue)

This command displays an active program's message queue.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
PGM	Name of active program whose message queue is displayed	

Notes

Only active programs can have their program message queues displayed. If a program appears more than once in the invocation stack, the queue of the most recent invocation is displayed.

Example

To display the message queue of program CPP1:

```
YDSPPGMQ PGM(CPP1)
```

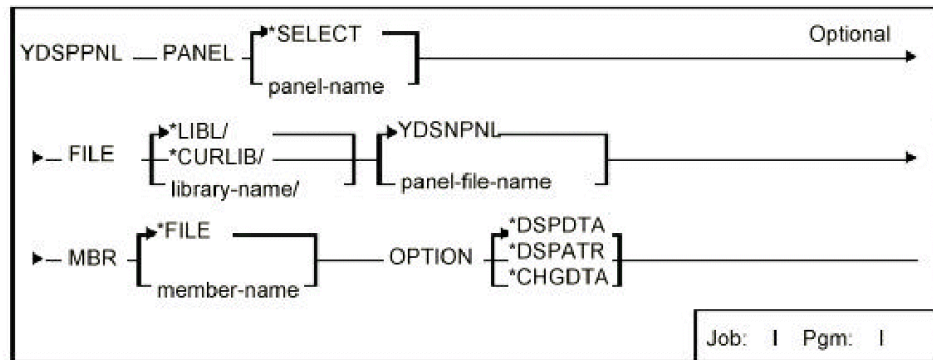
YDSPPNL (Display Panel Design)

This command displays a panel design as a prototype simulation of an actual display file and program. The panel is displayed with full attributes; for example, highlight, underline, input capable etc. Sample data may be shown in fields. The YDSPPNL command may be run in either of two modes:

Update mode is used to set up realistic sample data. Data may be entered into screen fields (both input and output), for storage with the panel design.

Display mode is used to present designs to a client. The design plus data may be displayed.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
PANEL	Name of the panel design which is displayed	*SELECT: (default) The panel selection display is presented
FILE	Qualified name of file containing panel design	YDSNPNL: (default) Panel design file name

Parameter	Definition	Value and Description
MBR	Name of member in file containing panel design	*FILE: (default) The member has the same name as the file
OPTION	Display option	<ul style="list-style-type: none">■ *DSPDTA: (default) Display panel design along with any previously stored data.■ *DSPATR: Display panel with field type attributes. These can be modified using the command Edit Design Defaults (YEDTDSNDFT)■ *CHGDTA: Display screen with first 128 fields (both input and output) as input fields. Any data keyed is stored with the panel design for later re-display. <p>N.B. If a field is moved then the stored data for that field will need to be re-keyed</p>

Notes

1. When you press the Help key while displaying a panel design, the narrative text associated with that panel design displays.
2. Panel designs are entered using the CA 2E utility Work with Panel (YWRKPNL).
3. When displaying a panel design, the command keys will be enabled according to the branching information associated with that panel design.
4. Use the HOME key to exit immediately from the display function. It will be necessary to use the HOME key to exit if no command keys are defined for a particular panel design.
5. If you want to blank out a data field when in *CHGDTA mode, you must enter at least one blank before pressing FIELD EXIT. Pressing FIELD EXIT by itself will not change the field

Examples

To display panel ODEON with data and attributes:

```
YDSPPNL  PANEL(ODEON)
```


To display panel ODEON to allow sample data to be changed:

```
YDSPPNL PANEL(ODEON) OPTION(*CHGDTA)
```

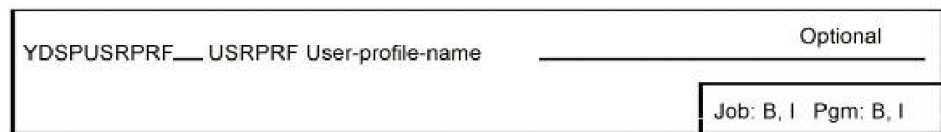
To display panel ODEON with attributes but no data:

```
YDSPPNL PANEL(ODEON) OPTION(*DSPATR)
```

YDSPUSRPRF (Display User Profile)

This command display user profiles. Both the i OS user profile details, and the extension attributes may be displayed.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
USRPRF	Name of user profile which is to be displayed	

Notes

1. This command calls an interactive display to display a user profile. Press Help while using the program for instructions.
2. Only those profiles known to CA 2E can be displayed. A profile is when the CA 2E commands Create User Profile (YCRTUSRPRF) or Change User Profile (YCHGUSRPRF) are used on it.
3. The profile may be changed using the CA 2E command Change User Profile (YCHGUSRPRF).

Example

To display user profile ATTILLA enter:

YDSPUSRPRF USRPRF(ATTILLA)

YEDTCKYTBL (Edit Command Key Table)

This command calls an interactive program to edit the conversion table used by the command Convert DDS Command Key (YCVTDDSCKY) to re-map command key usage in DDS source.

Syntax Diagram

YEDTCKYTBL There are no parameters for this command

Job: Pgm:

Parameters

None

Notes

1. This command calls an interactive display to add or modify command key mappings. Press HELP while using the program for instructions.
2. You can indicate if a command key enter in the table is to be mapped by an entry for the command key of the form *Cxnn*. The program will map *CFnn* to *CFmm* and *CAnn* to *CImm*, where *nn* represents the from command key and *mm* represents the to command key.

For each command key entered, a filter to indicate that is to be mapped is associated with a particular response indicator.

The type of mapping can be one of the following:

- Fixed: The mapping from key is not available for general use, and will not be automatically assigned by the program when mapping is mandatory; that is, when the from key is already in use in a given display file.
- Variable: When it is mandatory to map a command key the program will look for the first unassigned, variable key and use that to map to.

For example, the set of table values shown below causes YCVTDDSCKY to do the following:

- Map all instances of Cx01 to Cx03, Cx02 to Cx12, and Cx12 to Cx08, regardless of whether Cx03, or Cx12, or Cx08 are already in use. Any use of Cx08 will be remapped to use the first unassigned command key of Map type Variable.
- Map all instances of Cx03 associated with a response indicator of 03 to Cx10, regardless of whether Cx10 is already in use, but only if Cx03 is associated with the response indicator 03. If the conversion occurs then if Cx10 is in use it will also be remapped to use the first unassigned command key of Map type Variable.
- Map all instances of Cx14 to Cx07; however, if Cx07 is not to be used in a given display file. It will be considered available for general use.

Map from Keyword	Filter	Map to Keyword	Mapping
CMD01	03	CMD03	F
CMD02		CMD12	F
CMD03		CMD10	F
CMD12		CMD0	F
CMD14		8 CMD07	V
HELP		CMD01	F□

Example

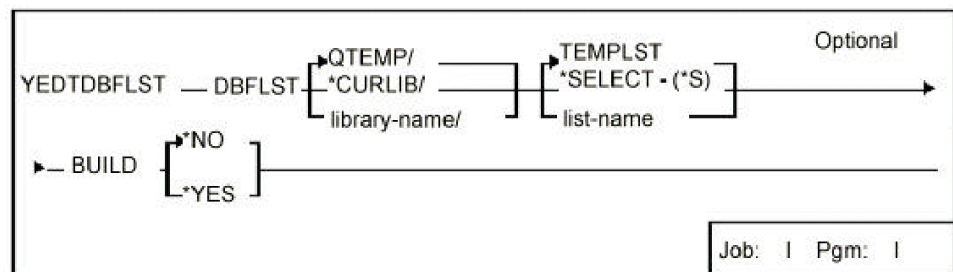
To invoke an interactive program to edit the command key table:

YEDTCKYTBL

YEDTDBFLST (Edit DBF List)

This command calls an interactive utility to edit a list of database files.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
DBFLST	Qualified name of database file list which is edited	<ul style="list-style-type: none">■ QTEMP/TEMPLST: (default) List name■ *SELECT: Display a list of existing database file lists, one of which is selected
BUILD	Build list option	<ul style="list-style-type: none">■ *NO: (default) Edit an existing list■ *YES: Invoke the build list function to build a new list before editing

Notes

None

Example

To edit database file list FRED in library QGPL:

```
YEDTDBFLST DBFLST(QGPL/FRED)
```

YEDTDSSNDFT (Design Default Data Areas)

This command calls an interactive program to edit the display, print, and DDS generation defaults for the screen and report design utilities, and also to edit the frame characters used in the command Convert Print (YCVTPRT). Enables design and presentation standards to be set centrally, and thereafter implemented automatically.

Syntax Diagram

YEDTDSSNDFT	There are no parameters for this command
Job:	Pgm:

Parameters

None

Notes

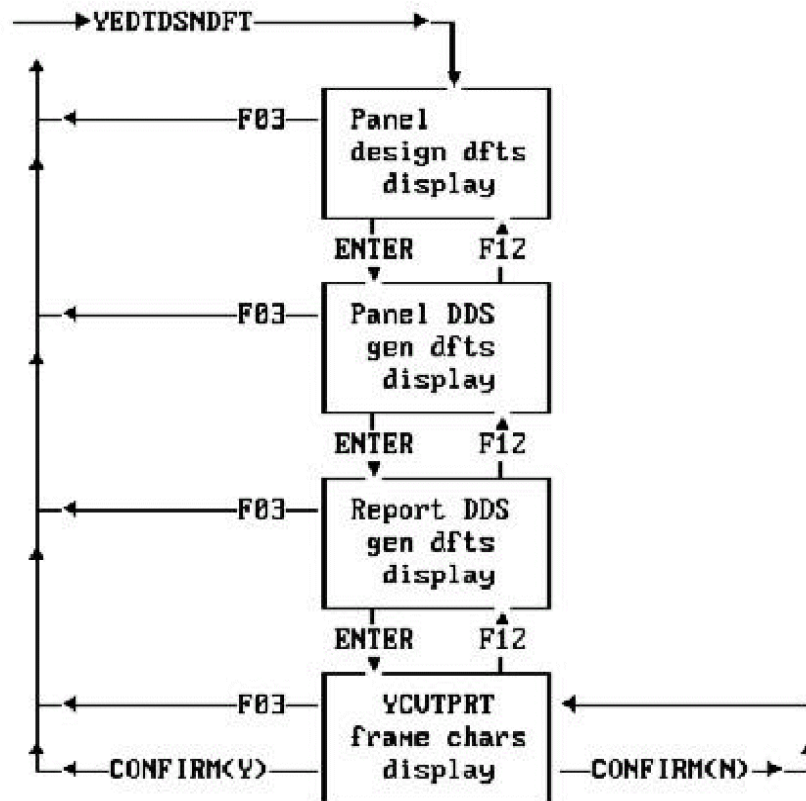
1. This command calls an interactive display to set the design defaults. Press Help while using the program for instructions.
2. The design defaults are stored in data areas. A set of the appropriate data area are supplied in the shipped utilities library (their names all begin with YD (YDSCDCA, YDSCDFA and YDLCLMA) or YP (YPBXCHA). If you need to keep several sets of standards current on the same machine, you may create extra copies of the data areas as follows.
 - CRTDUPOBJ OBJ(YD*) FROMLIB (utilities-product-library') OBJTYPE(*DTAARA) TOLIB (user-library-name)
 - CRTDUPOBJ OBJ(YPBXCHA) FROMLIB (utilities- product-library')OBJTYPE (*DTAARA) TOLIB(user-library-name)

Example

To call the program to change the design defaults:

YEDTDSNDFT

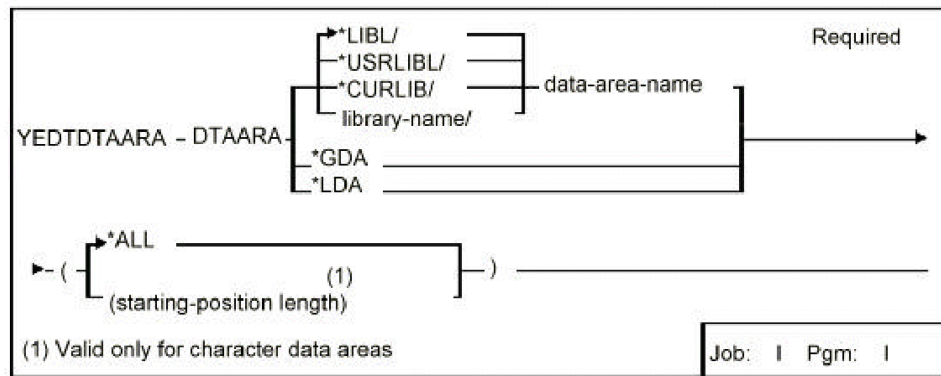
The following diagram display connection displays the main interconnections between the displays of the Design Default program:



YEDDTAARA (Edit Data Area Contents)

This command prompts the i OS command Change Data Area (CHGDTAARA) with the current contents of the data area already displayed.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
DTAARA	Simple list made up of the following elements	<ul style="list-style-type: none"> ■ * Qualified name of data area whose value is changed. Or a special value: ■ *LDA: Local data area for current job ■ *GDA: Group job data area for current job ■ Starting-position: Starting byte of data area character string that is changed ■ *ALL: (default) The entire data area is changed ■ * Length: Number of bytes of data area character string that are changed, beginning with the starting position <p>If used, both length and starting position must be specified, and must delimit a string within the bounds of the specified data area</p> <p>A maximum of 512 bytes may be changed during one execution of the command</p>

Notes

You must have data update rights to the data area.

Example

To change the contents of data area YYCOTXA in library QGPL between bytes 20 and 29:

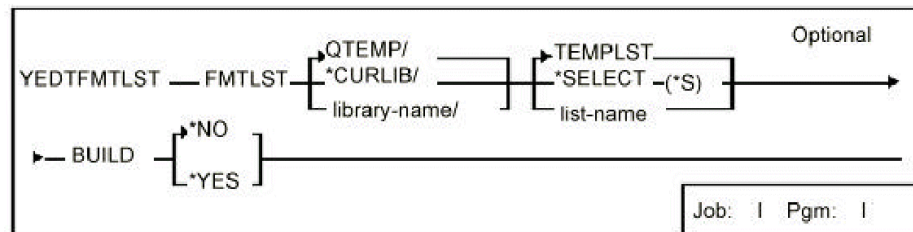
```
YEDDTAARA DTAARA(QGPL/YYCOTXA (20 10)
```

The existing contents of the data area will then be displayed. You can overwrite the existing values with a new value.

YEDTFMTLST (Edit Format List)

This command calls an interactive program to edit a list of file formats.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
FMTLST	Qualified name of format list which is to be edited	<ul style="list-style-type: none"> ■ QTEMP/TEMPLST: (default) List name ■ *SELECT: Display a list of existing format lists, one of which may be selected
BUILD	Build option	<ul style="list-style-type: none"> ■ *NO: (default) Edit an existing format list ■ *YES: Invoke the build list function to create a new format list before editing

Notes

None

Example

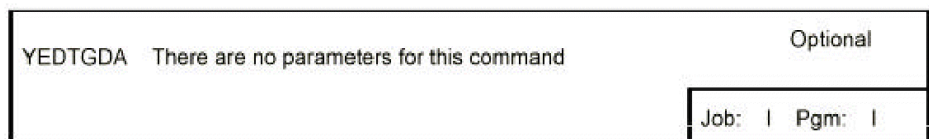
To edit format list RIK in library SHAW:

```
YEDTFMTLST FMTLST (SHAW/RIK)
```

YEDTGDA (Edit Group Data Area)

This command allows the interactive editing of a job's group data area.

Syntax Diagram



Parameters

None

Notes

This command calls an interactive display to edit the GDA. Press Help while using the program for instructions.

Example

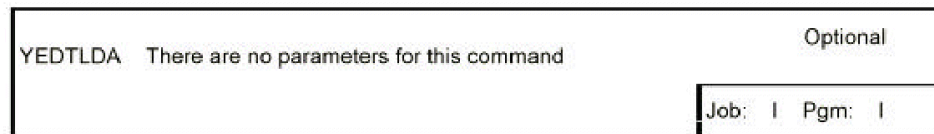
To edit the group data area for your job:

YEDTGDA

YEDTLDA (Edit Local Data Area)

This command allows the interactive editing of a job's local data area.

Syntax Diagram



Parameters

None

Notes

This command calls an interactive display to edit the LDA. Press Help while using the program for instructions.

Example

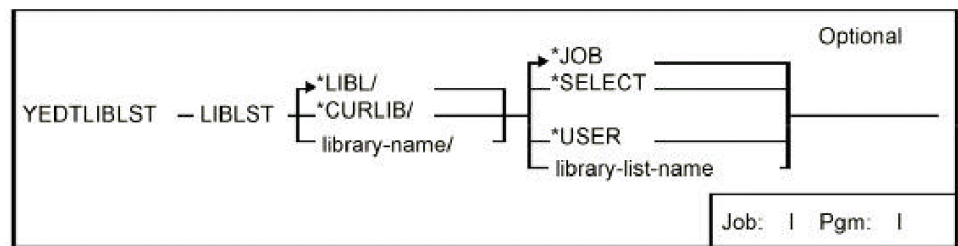
To edit the local data area for your job:

YEDTLDA

YEDTLIBLST (Edit Library List)

This command calls an interactive program to edit or change a library list. Library lists can be re-ordered, added to, or erased. The edited list can be stored away permanently as a library list. Also the command can be used to change the current job's library list.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
LIBLST	Qualified name of the edited library list	<ul style="list-style-type: none"> ■ *JOB: (default) Current job's library list ■ *SELECT: Display a list of existing lists, one of which may be selected ■ *USER: A stored library list having same name as the current job's user profile

Notes

1. Calls an interactive program to edit a member list. Press Help while using the program for instructions.
2. Library lists are stored in file YLIBLST in the library specified by the LIBLST parameter.

It is recommended that you have only one library list file per installation. However, additional files can be created as follows:

```

CRTDUPOBJ OBJ(YLIBLST) FROMLIB(utilities-product-library')
OBJTYPE(*FILE) TOLIB (library-name)

```

Examples

To edit the current job's library list:

```
YEDTLIBLST
```

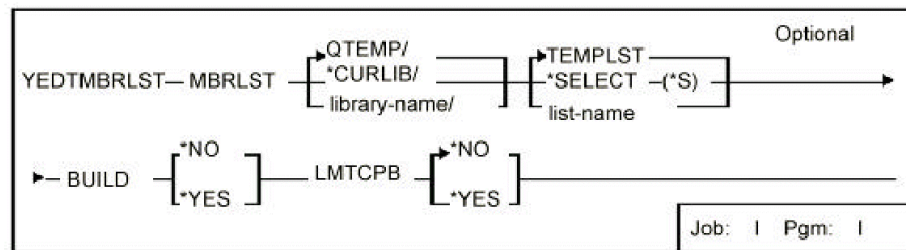
To edit a library list named BORGES:

```
YEDTLIBLST LIBLST(BORGES)
```

YEDTMBRLST (Edit Member List)

This command calls an interactive program to edit a file member.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
MBRLST	Qualified name of edited member list	<ul style="list-style-type: none"> ■ QTEMP/TEMPLST: (default) List name. ■ *SELECT: Display a list of existing member lists, one of which can be selected
BUILD	Build option	<ul style="list-style-type: none"> ■ *NO: (default) Edit an existing member list ■ *YES: Invoke the build list function to create a new member list before editing

Parameter	Definition	Value and Description
LMTCPB	Limit capabilities controls whether you can work with members and enter commands	<ul style="list-style-type: none"> ■ *NO: (default) Allow use of a command line and working with members ■ *YES: Do not supply a command line and disallow working with members

Notes

This command calls an interactive program to edit a member list. Press Help while using the program for instructions.

Example

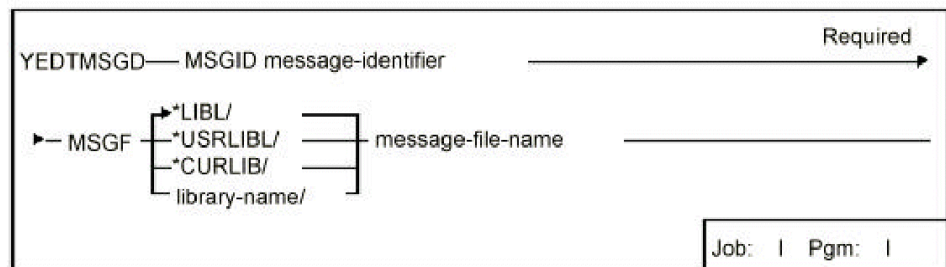
To edit member list HON in library WESTMINSTR:

```
YEDTMBRLST MBRLST(WESTMINSTR/HON)
```

YEDTMSGD (Edit Message Description)

This command retrieves an existing message description and displays it with the i OS command Change Message Description (CHGMSGD) so it can be changed.

Syntax Diagrams



Parameters

Parameter	Definition	Value and Description
MSGID	Message identifier of message description which is to be changed	Must be valid i OS message identifier
MSGF	Qualified name of message file containing changed message description	

Notes

The command also sends the command request to change the message to the program queue of the invoking job as a request message. This makes it possible for you to duplicate the request using the facility of the command entry program. Only the first 512 bytes can be duplicated.

Example

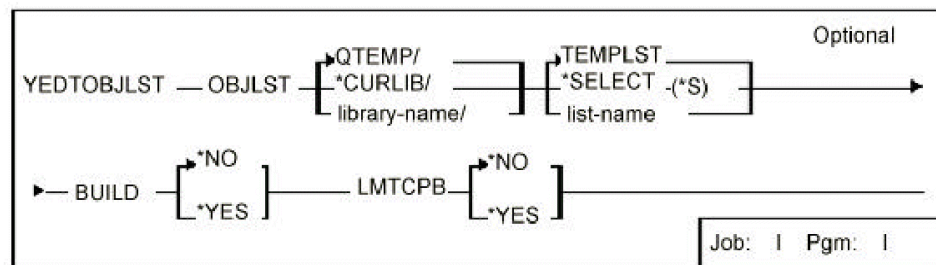
To edit the message description for USR0001 in message file UMSGF:

```
YEDTMSGD MSGID(USR0001) MSGF(UMSGF)
```

YEDTOBJLST (Edit Object List)

This command calls an interactive program to edit a list of objects.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
OBJLST	Qualified name of edited object list	<ul style="list-style-type: none"> ■ QTEMP/TEMPLST: (default) Object list name ■ *SELECT: Display a list of existing object lists, one of which may be selected
BUILD	Build list option	<ul style="list-style-type: none"> ■ *NO: (default) Edit existing list ■ *YES: Invoke the build list function to build a new list before editing it
LMTCPB	Limit capabilities controls whether you can work with objects and enter commands	<ul style="list-style-type: none"> ■ *NO: (default) Allows use of a command line and working with objects ■ *YES: Do not supply a command line and disallow working with objects

Notes

This command calls an interactive program to edit the object list. Press Help while using the program for instructions.

Example

To edit object list DART in QGPL:

```
YEDTOBJLST OBJLST(QGPL/DART)
```

YEDTPWDVAL (Edit Password Values)

This command calls an interactive program to edit the password control values and validation criteria. Maintains a list of forbidden password values, if required.

Syntax Diagram

YEDTPWDVAL There are no parameters for this command
Job: Pgm:

Parameters

None

Notes

1. This command calls an interactive program to edit the password control values, and a list of forbidden password values. Press Help while using the program for instructions.
2. You will need *CHANGE authority to the forbidden passwords file, YPWDVAL.

Example

To edit the password control values:

YEDTPWDVAL

YENDSPLRTR (End Spooled File Router)

The End spooled file router (YENDSPLRTR) command is used to end a spooled file router job which is monitoring the specified data queue.

See the command help for the Start spooled file router (YSTRSPLRTR) command for more information about the spooled file router job.

Parameters

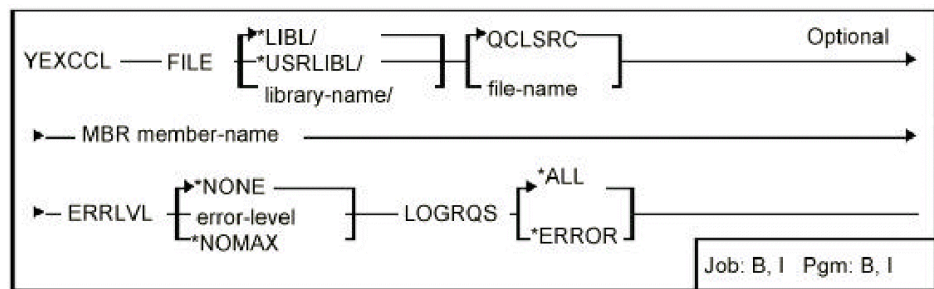
Parameter	Definition	Value and Description
-----------	------------	-----------------------

Parameter	Definition	Value and Description
DTAQ	<p>Specifies the data queue currently being monitored.</p> <p>Note: You must specify a data queue that is currently being monitored by a spooled file router job. Specifying a data queue that is not being monitored by a spooled file router job can result in serious errors within any programs using that data queue.</p>	<ul style="list-style-type: none"> ■ YSPLRTRQ: Use the default spooled router data queue YSPLRTRQ. ■ data-queue-name: Enter the name of an existing data queue which is being monitored by a spooled file router job. ■ *LIBL: All libraries in the user and system portions of the job's library list are searched until the first match is found for the specified data queue. ■ *CURLIB: The current library for the job is used to locate the specified data queue. If no library is specified as the current library for the job, the QGPL library is used. ■ library-name: Specify the name of the library to be searched for the specified data queue.

YEXCCL (Execute CL Member)

This command executes the CL statements in a source member.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
FILE	Qualified file name of source file contains the CL source which is to be executed	QCLSRC: (default) Source file
MEMBER	Source member name of member contains CL source	
ERRLVL	Number of execution errors allowed before the execution is abandoned	<ul style="list-style-type: none">■ *NONE: (default) No errors are allowed■ *NOMAX: There is no limit on the number of errors allowed
LOGRQS	Log request messages in log	<ul style="list-style-type: none">■ *ALL: (default) Logs all requests as executed■ *ERROR: Only logs error messages

Notes

CL members should be entered with SEU. Only those CL commands allow in readers.

Examples

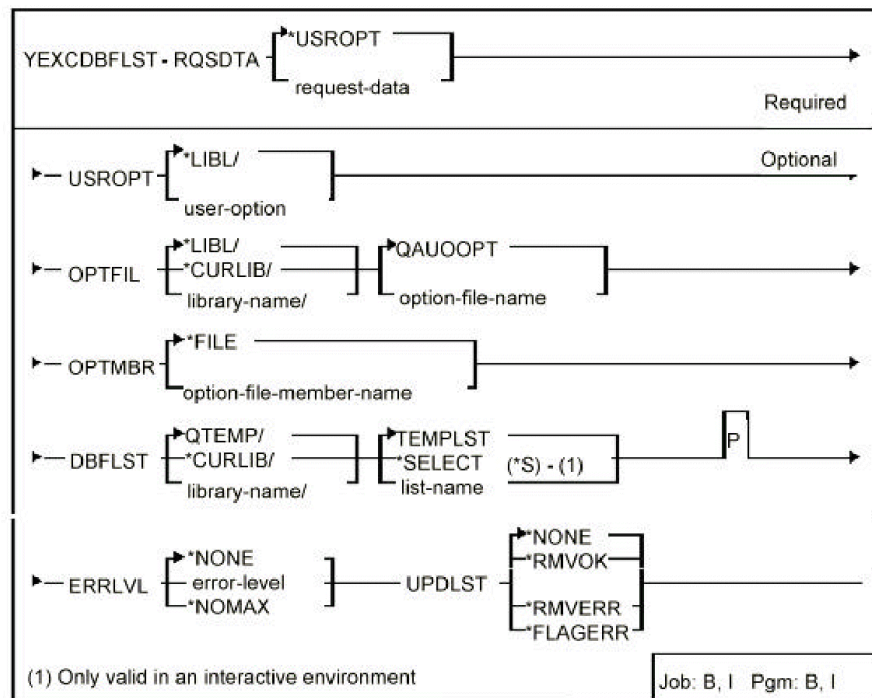
To execute the CL statements in CL member FRED:

```
YEXCCL FILE(QCLSRC) MBR(FRED)
```

YEXCDBFLST (Execute Database File List)

This command executes a specified command for each file in a specified database file list. A set of special substitution symbols indicates where the file, library names and other attributes are inserted.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
RQSDTA	Request data to execute for each item in database file list. Request string can be up to 256 characters (see the following details). To invoke the command prompter on the request string, place a ? as the first character	*USROPT: (default) Use the user option specified by the USROPT parameter
USROPT	Two character key identifying a record in the specified user-defined option file, containing user-defined option text for substitution in the RQSDTA string	*NONE: (default) No user option text is to be substituted

Parameter	Definition	Value and Description
OPTFIL	Qualified name of file containing the user-defined option text	QAUOOPT: (default) File name
OPTMBR	Name of member containing the user-defined option text	*FILE: (default) The member has the same name as the file
DBFLST	Qualified name of the executed database file list	<ul style="list-style-type: none">■ QTEMP/TEMPLST: (default) List name■ *SELECT: Display database file lists for selection
ERRLVL	Number of execution errors allowed before the list execution is abandoned	<ul style="list-style-type: none">■ *NONE: (default) No errors are allowed■ *NOMAX: There is no limit on the number of errors allowed <p>Note that execution does not take place if syntax errors are detected in the request string</p>
UPDLST	List update option	<ul style="list-style-type: none">■ *NONE: (default) The list is not to be updated■ *RMVOK: If the requested command completes without error for a list entry, then the list entry is removed from the list■ *RMVERR: If an error occurs when the requested command is executed for a list entry, then the list entry is removed from the list■ FLAGERR: If an error occurs when the requested command is executed for a list entry, then the list entry is to be flagged with an F

Notes

1. The DBF lists substitution symbols DBF lists executing request data is executed for each item.
2. User-defined options may be maintained using the i OS Programming Development Manager (PDM).

Example

To start journaling changes for all files in the database file list
QTEMP/TEMPLST:

```
YEXCDBFLST RQSDTA('JRNP FILE(&L/&F) JRN(TESTJRN) IMAGES(*BOTH)') UPDOPT(*RMVOK)
```

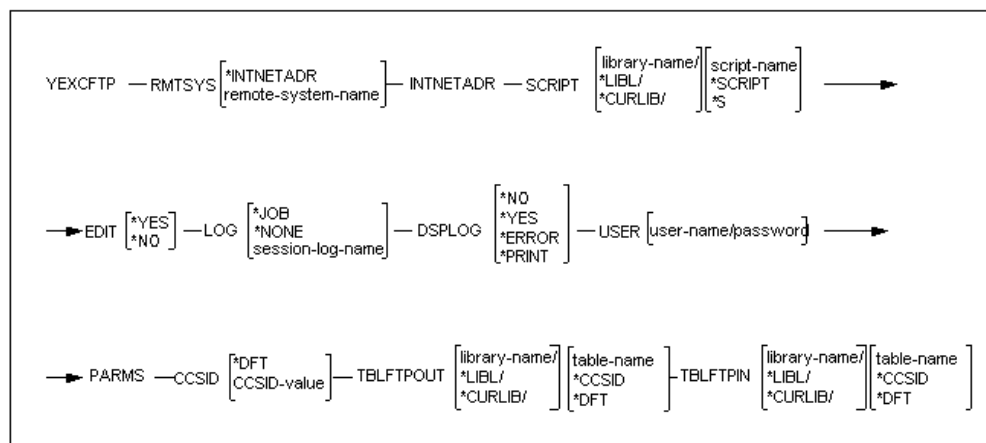
On completion, the list only contains entries for unsuccessfully commands

YEXCFTP (File Transfer Protocol)

Executes a pre-defined File Transfer Protocol (FTP) batch session.

- The FTP statements to be executed are stored as a 'script' member in a source file called YSCRIPT.
- The user can name an FTP script to use, or select one from a list of currently existing ones.
- The user can edit the FTP script before it is executed.
- The user can pass parameters to the script in the command, using substitution variables.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
RMTSYS	Remote system name to which or from which the files are transferred.	<ul style="list-style-type: none"> ■ *INTNETADR: The INTNETADR parameter is prompted. ■ remote-system-name: Remote system name to which or from which the file transfer takes place.
INTNETADR	Internet address of the remote system to which or from which the file transfer takes place.	

Parameter	Definition	Value and Description
SCRIPT	The script containing the FTP statements to execute. Each script is defined as a member in the YSCRIPT file. This parameter is required.	<ul style="list-style-type: none"> ■ *SELECT or *S: Script from the existing scripts in the selected YSCRIPT file. ■ script-name: Use an existing script in YSCRIPT. <p>Possible library values are:</p> <ul style="list-style-type: none"> ■ *LIBL: All libraries in the user and system portions of the job's library list are searched until the first match is found. ■ *CURLIB: The current library for the job is used to locate the YSCRIPT file. ■ library-name: Name of the library to be searched for the YSCRIPT file.
EDIT	Determines whether to edit the selected FTP session script prior to executing the batch FTP session. The FTP session script will be edited using Source Entry Utility (SEU).	<ul style="list-style-type: none"> ■ *YES: Edit the FTP session script prior to executing the FTP session. ■ *NO: Execute the FTP session without editing the FTP session script.
LOG	Name of member in YSCRIPT where run-time messages from the executed FTP session are stored.	<ul style="list-style-type: none"> ■ *JOB: A session log is created for this job in file YSCRIPT in the same library as the FTP session script. ■ *NONE: No permanent log file will be created. ■ session-log-name: Use the specified log.

Parameter	Definition	Value and Description
DSPLOG	Determines whether to display the FTP session log after the FTP session has executed.	<ul style="list-style-type: none">■ *NO: Do not display the FTP session log after the FTP session has ended■ *YES: Display the FTP session log after the FTP session has ended.■ *ERROR: Only display the FTP session log if it contains any FTP server replies that contain error reply codes.■ *PRINT: Prints the FTP session log.
USER	Determines the user name and password to use when connecting to the remote system.	<ul style="list-style-type: none">■ user-name/password: The user name and password are used to connect to the remote system.
PARMS	Specifies up to 9 substitution variables and related values that can be used in the specified script.	<ul style="list-style-type: none">■ In addition to the &1 - &9 variables allowed, the special variables &U, &P, and &M are also allowed.■ @1 - @9: These substitution variables can be used only once within a given execution of the command.
CCSID	ASCII coded character set identifier (CCSID) that is used for single-byte character set (SBCS) ASCII file transfers when the FTP TYPE mode is set to ASCII.	<ul style="list-style-type: none">■ *DFT: The CCSID value 00819 (ISO 8859-1 8-bit ASCII) is used.■ CCSID-value: The requested CCSID value is used.

Parameter	Definition	Value and Description
TBLFTPOUT	Table object that is to be used to map all outgoing data in the FTP client. Outgoing data is mapped from EBCDIC to ASCII.	<ul style="list-style-type: none"> ■ *CCSID: CCSID parameter used to determine outgoing mapping. ■ *DFT: CCSID parameter used to determine outgoing mapping: <ul style="list-style-type: none"> — *LIBL: All libraries in the user and system portions of the job's library list are searched until the first match is found. — *CURLIB: The current library for the job is searched. — library-name: Name of the library to be searched. ■ outgoing-mapping-table: Table object to be used by the FTP client for mapping
TBLFTPIN	Table object that is to be used to map all incoming data in the FTP client. Incoming data is mapped from ASCII to EBCDIC.	<ul style="list-style-type: none"> ■ *CCSID: CCSID parameter used to determine incoming mapping. ■ *DFT: CCSID parameter used to determine incoming mapping: <ul style="list-style-type: none"> — *LIBL: All libraries in the user and system portions of the job's library list are searched until the first match is found. — *CURLIB: The current library for the job is searched. — library-name: Name of the library to be searched. ■ incoming-mapping-table: Table object to be used by the FTP client for mapping

Notes

None.

Examples

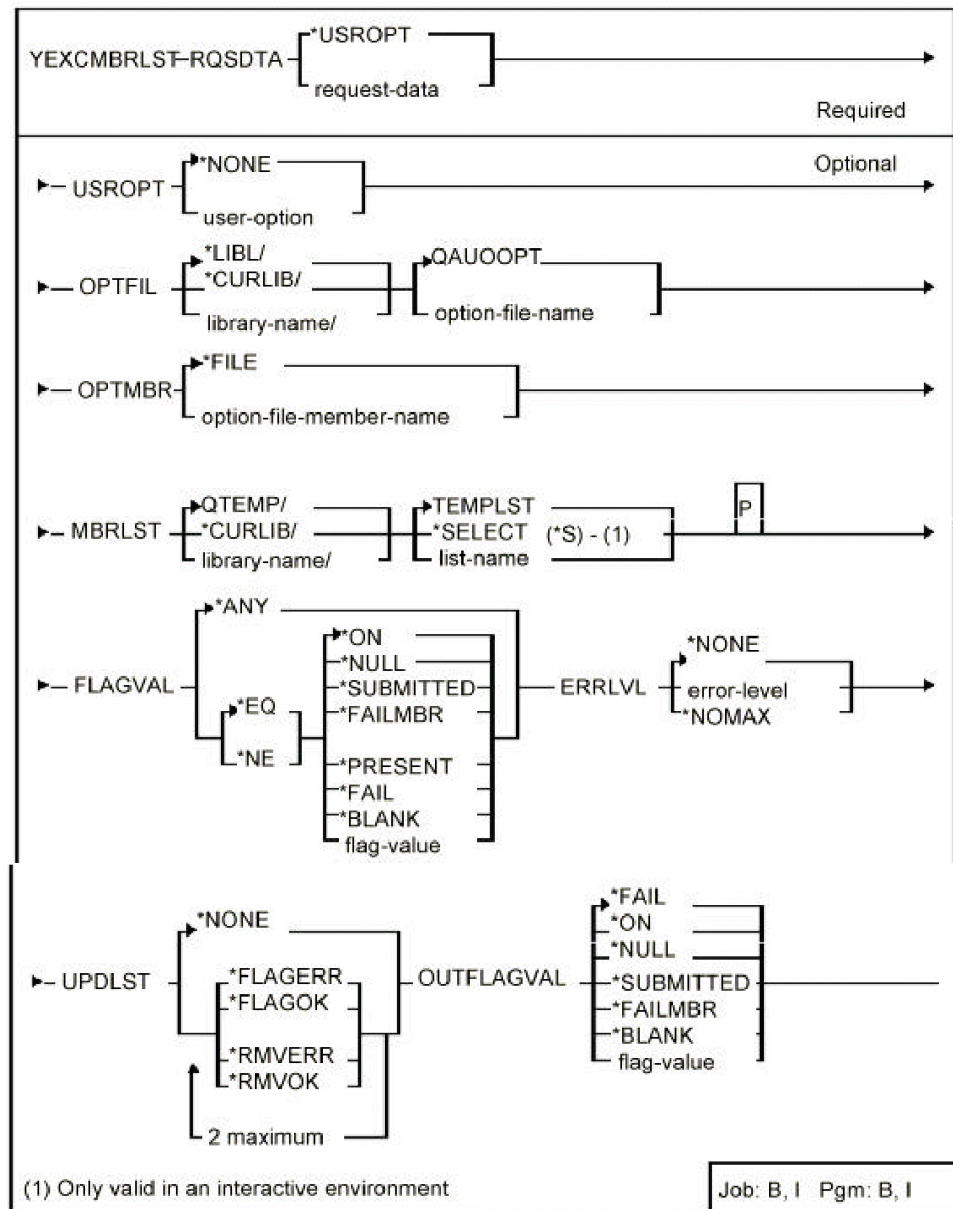
The following command is used to execute the FTP statements in Script **ABCSCRIPT** for remote system **SYSTEM1**. User name **User1** and password **Pass10** will be used to connect to machine **SYSTEM1**.

```
YEXCFTP RMTSYS(SYSTEM1) SCRIPT(ABCSCRIPT) USER(User1 Pass10)
```

YEXCMBRLST (Execute Member List)

This command executes a command or user-defined option for each member name in a specified member list. A set of special substitution symbols indicates where the member, file, library names and other attributes are to be inserted in the command request string. Entries may be removed or flagged following execution of the command.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
RQSDTA	Request data to execute upon the items in the specified member list. The request string can be up to 256 characters long (see below for further details). To invoke the command prompt on the request data, place a '?' as the first character	*USROPT: Use the user-defined option specified by the USROPT parameter
USROPT	Two character user-defined option identifying a record in the specified user-defined option file, containing user option text for substitution in the RQSDTA string	*NONE: (default) No user option text is to be substituted
OPTFIL	Qualified name of file containing the user-defined option text	QAUOOPT: (default) File name
OPTMBR	Name of member containing the user option text	*FILE: (default) The member has the same name as the file
MBRLST	Qualified name of executed member list	<ul style="list-style-type: none">■ QTEMP/TEMPLST: (default) List name■ *SELECT: Display member lists for selection

Parameter	Definition	Value and Description
FLAGVAL	Flag value of list entries to select for execution	<p>*ANY: (default) Execute all list entries</p> <p>Otherwise, FLAGVAL is a list parameter made up of the following two elements:</p> <p>Relational operator for selection of flags</p> <ul style="list-style-type: none"> ■ *EQ: (default) Equal to ■ *NE: Not equal to <p>Flag value</p> <ul style="list-style-type: none"> ■ Single character flag value or one of the special flag values. See Appendix A for further details on flag values
ERRLVL	Number of execution errors allowed before the list execution is abandoned	<ul style="list-style-type: none"> ■ *NONE: (default) No errors are allowed ■ *NOMAX: There is no limit on the number of errors allowed <p>Note that execution does not take place if syntax errors are detected in the request data</p>

Parameter	Definition	Value and Description
UPDLST	List update option	<ul style="list-style-type: none">■ *NONE: (default) The list is not to be updated Otherwise, UPDLST can be up to two of the following:■ *RMVOK: If the requested command completes without error for a list entry, then the list entry is removed from the list■ *RMVERR: If an error occurs when the requested command is executed upon a list entry, then the list entry is removed from the list■ *FLAGERR: If an error occurs when the requested command is executed upon a list entry, then the list entry is to be flagged with the flag specified by the OUTFLAGVAL parameter■ *FLAGOK: If the requested command completes without error for a list entry, then the list entry is flagged with the flag specified by the OUTFLAGVAL parameter <p>If more than one value is specified, *RMVERR + *FLAGOK and *RMVOK + *FLAGERR are the only valid combinations</p>
OUTFLAGVAL	Flag value given to flagged entries when the UPDLST parameter is *FLAGERR or *FLAGOK	Single character flag value or one of the special flag values

Notes

1. The member lists can be built and manipulated using the commands Build Member List (YBLDMBRLST) and Edit Member List (YEDTMBRLST).
2. The request data is executed for each member name in the specified member list. You can specify where in the request string you wish to have the member names inserted by using the substitution symbols given under the RQSDTA parameter in [Appendix A](#).
3. Note that items in the list without a member name are ignored.
4. Not all of the substitution variables need to be specified.
5. User-defined options may be maintained using the i OS Programming Development Manager (PDM).

Examples

To copy all members in member list QTEMP/TEMPLST (the default member list), from their existing locations, to a file called NEWSRC:

```
YEXCMBRLST RQSDTA('CPYSRCF FROMFILE(&L/&F) TOFILE(&L/NEWSRC) FROMMBR(&M)')
```

To make a member list (QTEMP/TEMPLST) created on one file drive processing on another:

```
YEXCMBRLST RQSDTA('CPYSRCF HARRY FRED @M @M *REPLACE') UPDOPT(*RMVOK)
```

On completion the list will only contain entries for which the command did not complete successfully.

Example of use of flags

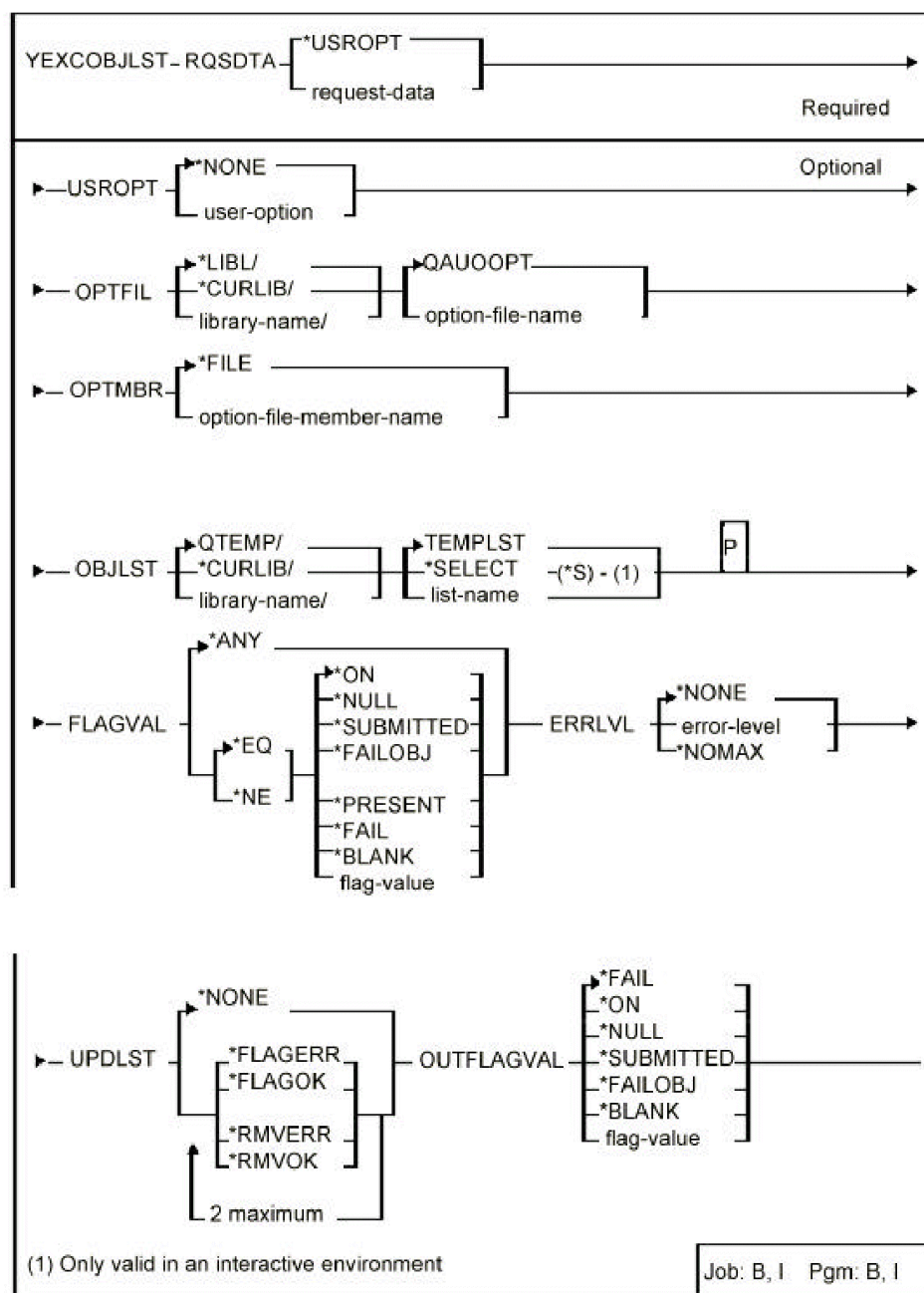
To process only those items which have a flag value of A, flagging those items for which the command failed with an F (*FAIL) and removing those for which the command succeeded:

```
YEXCMBRLST RQSDTA('DSPPFM &N &M') FLAGVAL(A) UPDOPT(*RMVOK *FLAGERR)  
OUTFLAGVAL(*FAIL)
```

YEXCOBJLST (Execute Object List)

This command executes a command or user-defined option for each object name in a specified object list. A set of special substitution symbols indicates where the object name, library name, object type and other attributes are to be inserted. Specific list entries may be executed by selecting on a given flag. Entries may be removed or flagged following execution of the command.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
RQSDTA	Request data to execute upon each item in the specified object list. The request string can be up to 256 characters long (see below for further details). To invoke the command prompt on the request string, place a ? as the first character in the request data.	*USROPT: (default) Use the user option specified by the USROPT parameter
USROPT	Two character user-defined option identifying a record in the specified user-defined option file, containing user option text for substitution in the RQSDTA string	*NONE: (default) No user-defined option text is to be substituted
OPTFIL	Qualified name of file containing the user-defined option text	QAUOOPT: (default) File name
OPTMBR	Name of member containing the user option text	*FILE:(default) The member has the same name as the file
OBJLST	Qualified name of Synon/1E object list which is to be executed	QTEMP/TEMPLST: (default) Object list name *SELECT: Display object lists for selection

Parameter	Definition	Value and Description
FLAGVAL	Flag value of list entries to select for execution	<ul style="list-style-type: none">■ *ANY: (default) Execute all list entries Otherwise, FLAGVAL is a list parameter made up of the following two elements: Relational operator for selection of flags <ul style="list-style-type: none">■ *EQ: (default) Equal to■ *NE: Not equal to Flag value <ul style="list-style-type: none">■ Single character flag value or one of the special flag values
ERRLVL	Number of execution errors allowed before the list execution is abandoned	<ul style="list-style-type: none">■ *NONE:(default) No errors are allowed■ *NOMAX: There is no limit on the number of errors allowed Note that execution does not take place if syntax errors are detected in the request string

Parameter	Definition	Value and Description
UPDLST	List update option	<ul style="list-style-type: none"> ■ *NONE: (default) The list is not to be updated. Otherwise, UPDLST can be up to two of the following: ■ *RMVOK: If the requested command completes without error for a list entry, then the list entry is removed from the list. ■ *RMVERR: If an error occurs when the requested command is executed upon a list entry, then the list entry is removed from the list. ■ *FLAGERR: If an error occurs when the requested command is executed upon a list entry, then the list entry is to be flagged with the flag specified by the OUTFLAGVAL parameter. ■ *FLAGOK: If the requested command completes without error for a list entry, then the list entry is flagged with the flag specified by the OUTFLAGVAL parameter. <p>If more than one value is specified, *RMVERR + *FLAGOK and *RMVOK + *FLAGERR are the only valid combinations</p>
OUTFLAGVAL	Flag value to be given to flagged entries when the UPDLST parameter is *FLAGERR or *FLAGOK	Single character flag value or one of the special flag values.

Notes

1. Object lists can be built and manipulated using the commands Build Object List (YBLDOBJLST) and Edit Object List (YEDTOBJLST).
2. The request data is executed for each object in the object list. You can specify where in the request string you wish to have the object names inserted by using the substitution symbols given under the RQSDTA parameter in [Appendix A](#).
3. Not all of the substitution symbols need be specified.
4. User-defined options may be maintained using the i OS Programming Development Manager (PDM).

Examples

To print a full object description for each item in the list QTEMP/TEMPLST:

```
YEXCOBJLSTRQSDTA('DSP0BJD &L/&O &T OUTPUT(*PRINT) DETAIL(*FULL)')
```

Example of special rule

To print a full object description for each item in the list QTEMP/TEMPLST, prompting first for the parameters:

```
YEXCOBJLST RQSDTA(' ?DSP0BJD @L/@O OBJTYPE(*PGM) OUTPUT(*PRINT) DETAIL(*FULL)')
```

Example of use of only one substitution symbol

To list all of the user profiles in list QTEMP/TEMPLST:

```
YEXCOBJLST RQSDTA('DSPUSRPRF QSYS/&O OUTPUT(*LIST)') UPDOPT(*RMVOK)
```

On completion the list will only contain entries for which the command did not complete successfully.

Example of use of flag

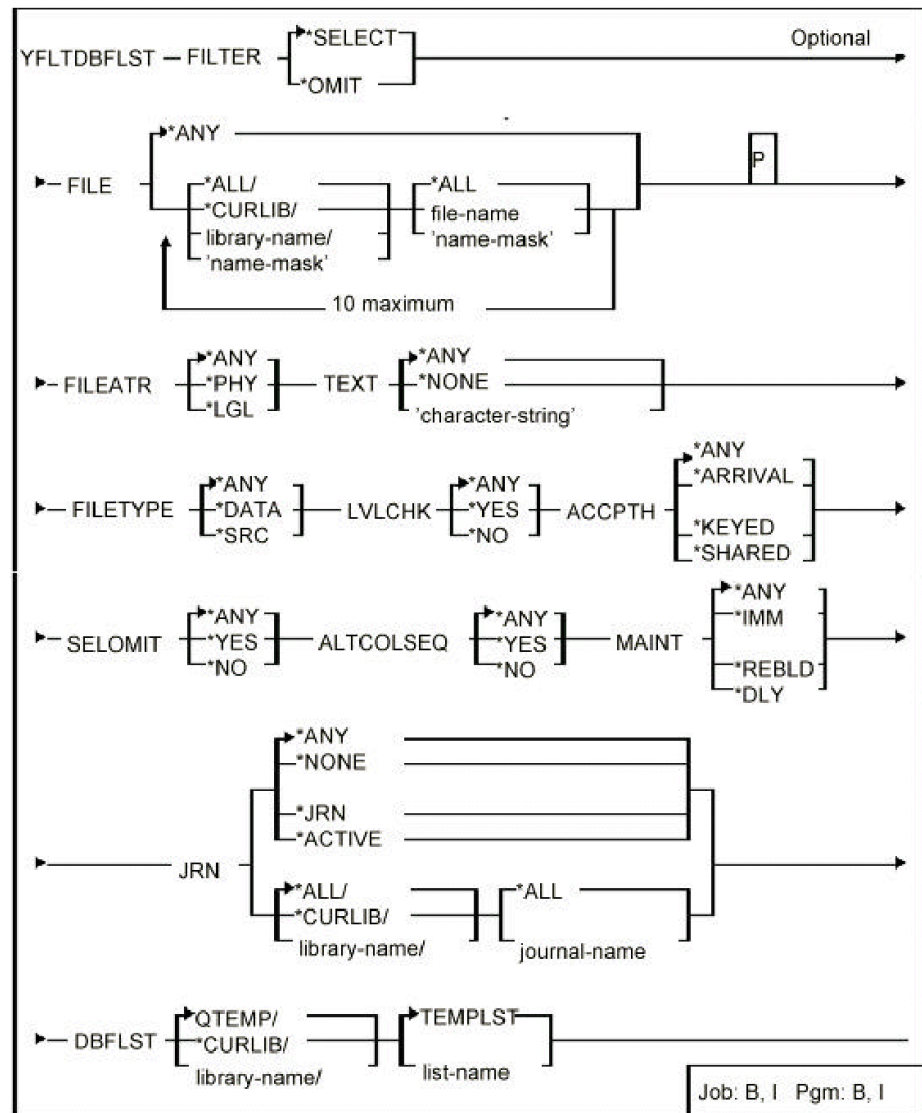
To process only those items which have a flag value of A, flagging those items for which the command failed with an F (*FAIL) and removing those for which the command succeeded:

```
YEXCOBJLST RQSDTA('DSP0BJD &N &T') FLAGVAL(A) UPDOPT(*RMVOK *FLAGERR)  
OUTFLAGVAL(*FAIL)
```

YFLTDBFLST (Filter Database List)

This command removes unwanted items from an existing database file list.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
FILTER	Type of filtering required	<ul style="list-style-type: none">■ *SELECT: (default) Keep items that satisfy the filtering conditions■ *OMIT: Reject items which satisfy the filtering conditions
FILE	List of up to ten qualified names or name masks of files on which to filter. Name masks may contain ? or *	<ul style="list-style-type: none">■ *ANY: (default) No filtering on file name or file library name■ '*ALL/name mask': Filter on file name but not on library name■ 'name mask/*ALL': Filter on library name but not on file name
FILEATR	File type on which to filter	<ul style="list-style-type: none">■ *ANY: (default) No filtering on file type■ *PHY: Select physical files only■ *LGL: Select logical files only
TEXT	File text on which to filter	<ul style="list-style-type: none">■ *ANY: (default) No filtering is done on file text■ *NONE: Only those files with no text are included■ Character string: Only those files whose text contains the specified string, starting at any position, are included in the list. Up to fifty characters of search string may be specified. Upper and lower case differences are ignored. The character ? may be used as a wild character in the search string

Parameter	Definition	Value and Description
FILETYPE	File data type on which to filter	<ul style="list-style-type: none"> ■ *ANY: (default) No filtering on file data type ■ *DATA: Select data files only ■ *SRC: Select source files only
LVLCHK	File level checking attribute on which to filter	<ul style="list-style-type: none"> ■ *ANY: (default) No filtering on file level checking ■ *YES: Select files for which LVLCHK(*YES) applies ■ *NO: Select files for which LVLCHK(*NO) applies
ACCPH	File access path type on which to filter	<ul style="list-style-type: none"> ■ *ANY: (default) No filtering on file access path type ■ *ARRIVAL: Select arrival sequence files ■ *KEYED: Select keyed sequence files ■ *SHARED: Select files sharing access paths
SELOMIT	File access path select/omit logic attribute on which to filter	<ul style="list-style-type: none"> ■ *ANY: (default) no filtering on select/omit logic ■ *YES: Select files for which select or omit logic has been specified ■ *NO: Select files for which no select or omit logic has been specified

Parameter	Definition	Value and Description
ALTCOLSEQ	Alternate collating sequence attribute on which to filter	<ul style="list-style-type: none">■ *ANY: (default) No filtering on alternate collating sequence■ *YES: Select files for which an alternate collating sequence has been specified■ *NO: Select files for which no alternate collating sequence has been specified■ N.B. This keyword is not available from the Edit Database File List command display (YEDTDBFLST) due to programming limitations
MAINT	File access path maintenance type on which to filter	<ul style="list-style-type: none">■ *ANY: (default) No filtering on file maintenance type■ *IMM: Select immediate maintenance files■ *REBLD: Select rebuild maintenance files■ *DLY: Select delayed maintenance files
JRN	Qualified journal name on which to filter	<ul style="list-style-type: none">■ *ANY: (default) No filtering on journal data■ *NONE: Select unjournalized files■ *JRN: Select all files that are being or have been journaled■ *ACTIVE: select files for which journaling is currently active <p>If a journal name or library is specified the special value *ALL may be used to match all values</p>
DBFLST	Qualified name of dbf list which is to be filtered	QTEMP/TEMPLST: (default) list name

Notes

1. A name mask is used to specify which names are to be selected.

The mask may contain the following special characters:

- '?' anywhere within the mask: match on any character.
- '**' anywhere within the mask: floating scan indicator.
- '*' at the end of the mask: generic name indicator

The mask characters may be combined together. Examples of name masking are as follows:

- A* would select all names beginning with 'A'. *A* would select any names containing 'A'. *A would select any name ending with 'A'. A?C would select any name beginning with 'A' and with 'C' in the third position.
- *A? would select any name with 'A' as the last character but one.
- A*B* would select any name starting with 'A' and containing 'B'.
- Q???SRC will match QRPGRS & QDDSSRC, but not QCLSRC.
- Q*SRC will match QRPGRS & QCLSRC, but not QRPGRS1.
- Q*S?C* will match all examples, plus names such as QSXC.

2. Rules for combining selection parameters include the following:

- If *ANY is specified for a parameter that parameter is not used for filtering.
- Different parameters are ANDed together.

The following would select only physical source files:

```
YFLTDBFLST FILEATR(*PHY)
FILETYPE(*SRC)
```

- Where a list of values may be specified for a parameter the list elements are ORed together.

The following would select any files whose names begin either with the letters ABC or with XYZ:

```
YFLTDBFLST FILE((ABC*)(XYZ*))
```

Each element of the file name list can consist of a qualified file name. For example, name plus library name. If both file and library name are specified they are ANDed together. The following would select any files whose names begin with the letters ABC and that are in any library having a name beginning with the letters XYZ:

```
YFLTDBFLST FILE((ABC*XYZ*))
```

Examples

To select all physical files in database file list QTEMP/TEMPLST which are being journaled:

```
YFLTDBFLST FILTER(*SELECT) JRN(*ACTIVE)
```

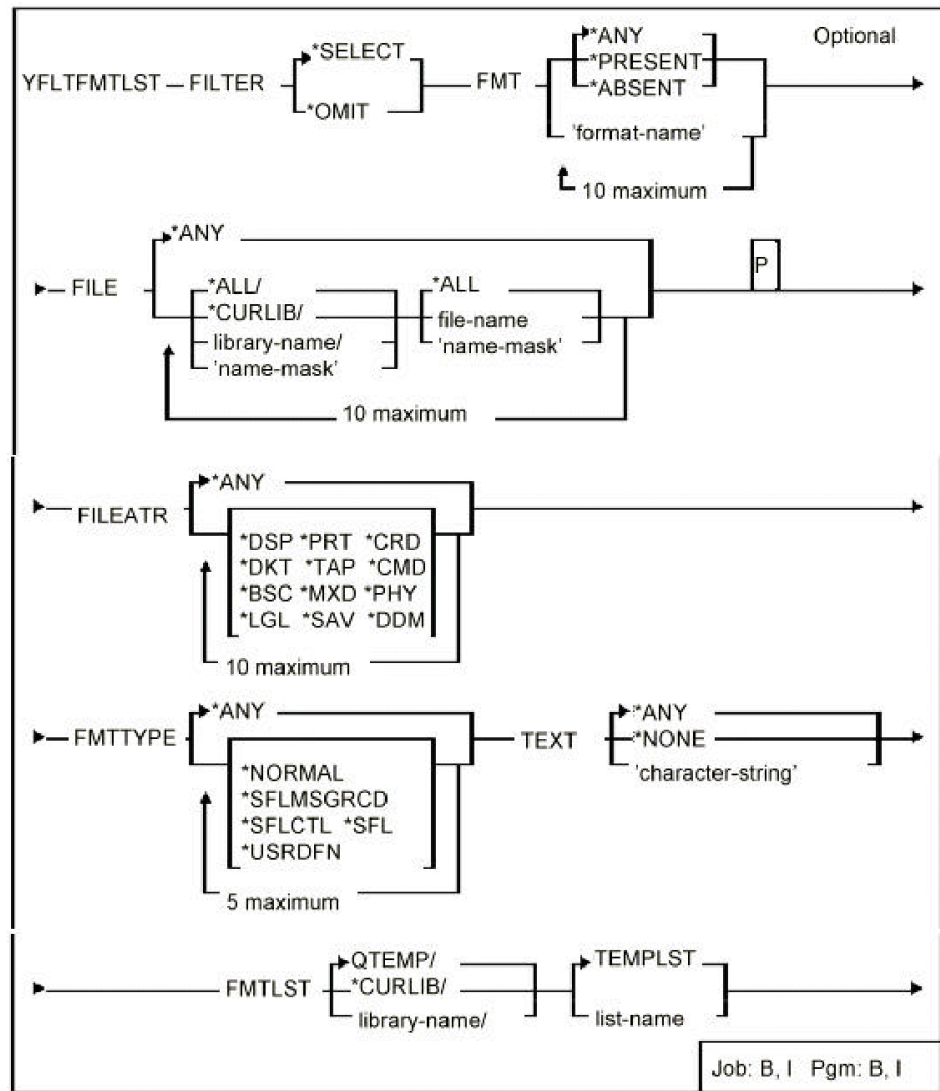
To select all files in database file list QTEMP/TEMPLST with names beginning with the letters GL, for which *REBLD maintenance has been specified:

```
YFLTDBFLST FILTER(*SELECT) FILE(*ALL/GL*) MAINT(*REBLD)
```

YFLTFMTLST (Filter Format List)

This command removes unwanted items from existing format list.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
FILTER	Type of filtering required	<ul style="list-style-type: none">■ *SELECT: (default) Keep items which satisfy the filtering conditions.■ *OMIT: Reject items which satisfy the filtering conditions
FMT	List of format names on which to filter. Either one of the following single values	<ul style="list-style-type: none">■ *ANY: (default) No filtering on format name■ *PRESENT: A non-blank format name must be present■ *ABSENT: Format name must be blank Or a list of up to ten generic format names
FILE	List of up to ten qualified names or name masks of files on which to filter. Name masks may contain ? or *	<ul style="list-style-type: none">■ *ANY: (default) No filtering on file name or file library name■ *ALL/name mask': Filter on file name but not on library name■ 'name mask/*ALL': Filter on library name but not on file name
FILEATR	File attribute on which to filter	<ul style="list-style-type: none">■ *ANY: (default) No filtering on file attribute Valid i OS or S38 file attribute

Parameter	Definition	Value and Description
FMTTYPE	Format type on which to filter	<ul style="list-style-type: none"> ■ *ANY: (default) No filtering on format type <p>Or a list of up to five format types:</p> <ul style="list-style-type: none"> ■ *NORMAL: Any format type except for subfile control, subfile record, subfile message record or user-defined types. ■ *SFLMSGRC: Subfile message record format types ■ *SFLCTL: Subfile control record format type ■ *SFL: Subfile record format type ■ *USRDFN: User-defined format types
TEXT	Format text on which to filter	<ul style="list-style-type: none"> ■ *ANY: (default) No filtering is done on format text ■ *NONE: (default) Only those formats with no text included ■ Character string: Only formats whose text contains the specified string, starting at any position, are included in the list. Up to fifty characters of search string may be specified. Upper and lower case differences are ignored. The character ? may be used as a wild character in the search string
FMTLST	Qualified name of format list that is to be filtered	QTEMP/TEMPLST: (default) list name

Notes

Rules for combining selection parameters are as follows:

- If *ANY is specified for a parameter that parameter is not used for filtering.
- Different parameters are ANDed together.
- Where a list of values may be specified for a parameter the list elements are ORed together.
- Each element of the file name list can consist of a qualified file name, i.e. file name plus library name. If both file and library name are specified they are ANDed together. The following would select any files whose names begin with the letters ABC and that are in any library having a name beginning with the letters XYZ:

```
YFLTFMTLST FILE((ABC* XYZ*))
```

Examples

To select all formats with blank text:

```
YFLTFMTLST FILTER(*SELECT) TEXT(*NONE)
```

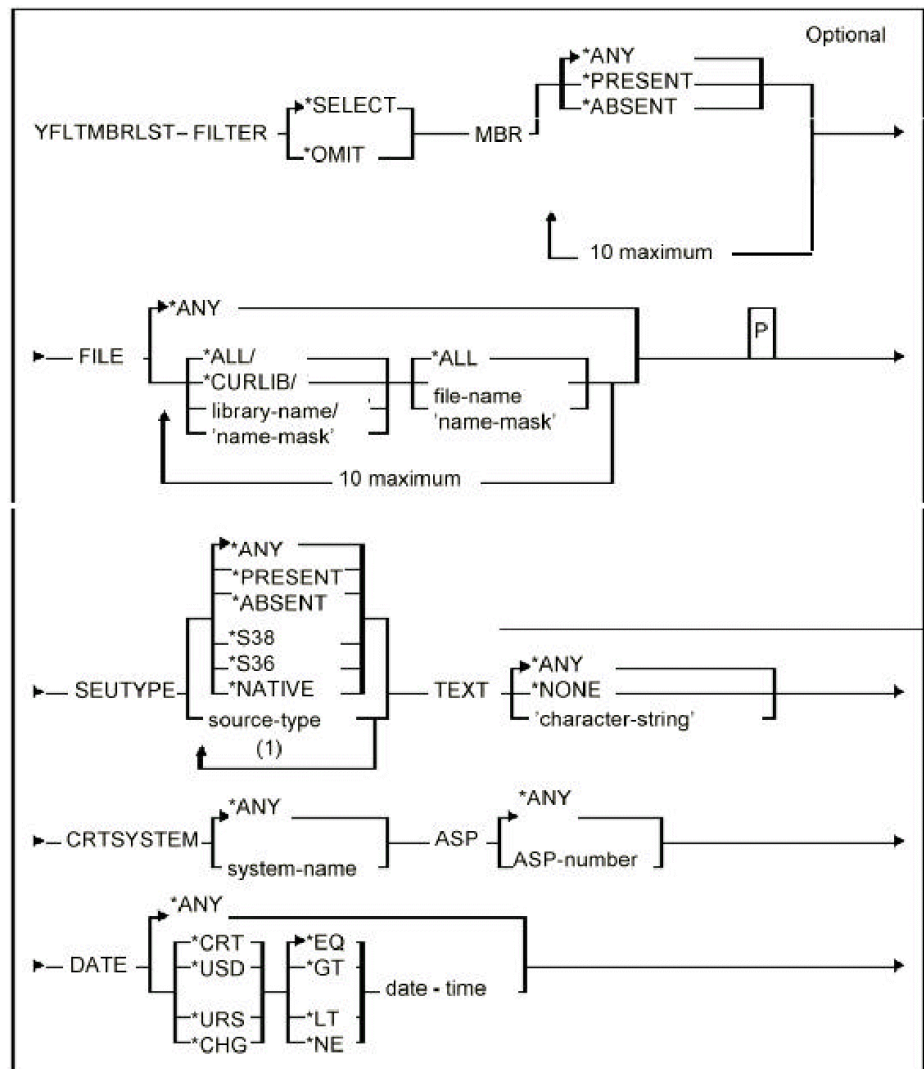
To select all formats with names beginning with the characters @L, which are subfile control record formats:

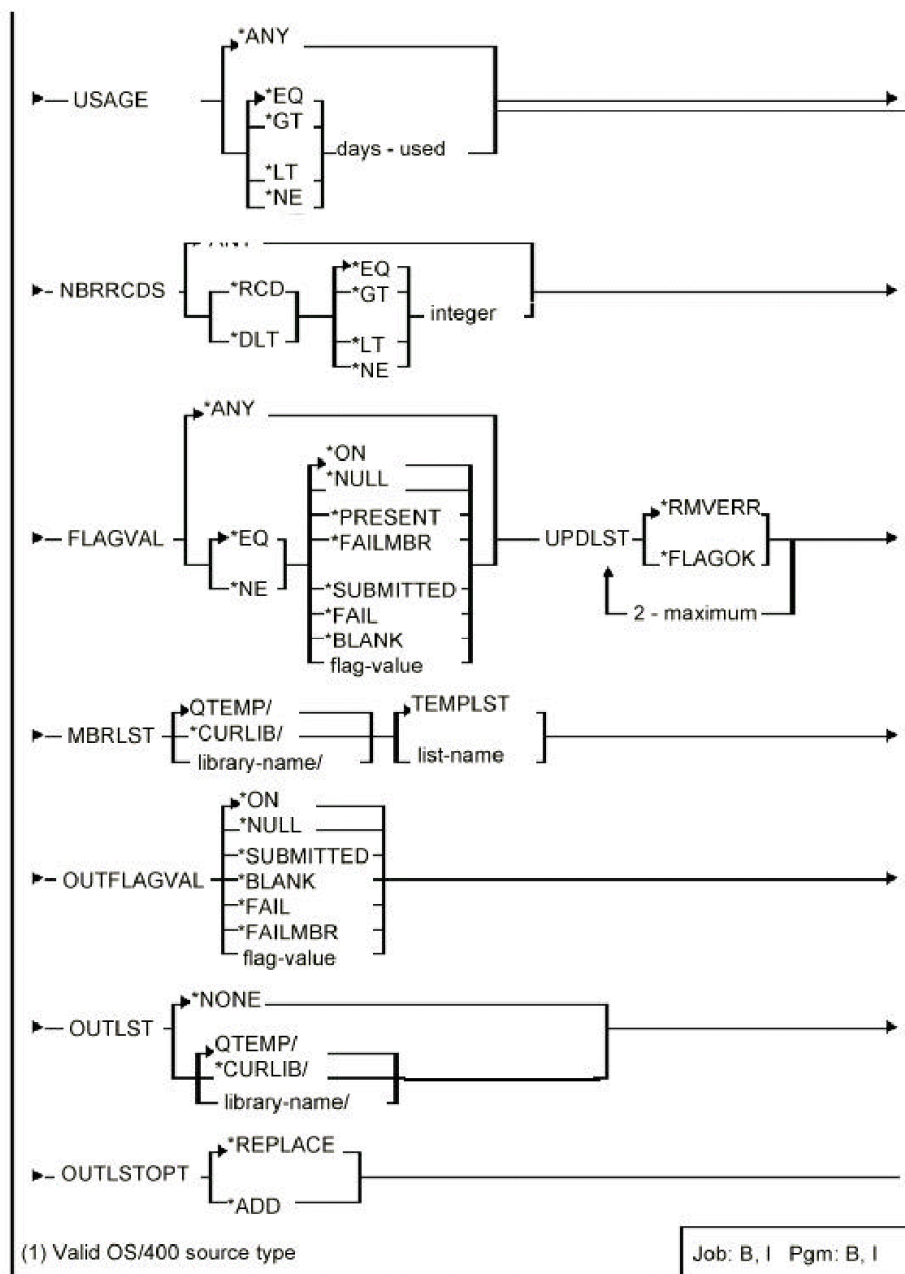
```
YFLTFMTLST FILTER(*SELECT) FMT(@L*) FMTPYPE(*SFLCTL)
```

YFLTMBRLST (Filter Member List)

This command removes unwanted items from an existing member list. Wanted items can be placed in a designated output member list.

Syntax Diagram





Parameters

Parameter	Definition	Value and Description
FILTER	Type of filtering required	<ul style="list-style-type: none"> ■ *SELECT: (default) Keep items that satisfy filtering conditions ■ *OMIT: Reject items that satisfy the filtering conditions
MBR	List of up to ten names or name masks of members on which to filter. Name masks may contain ? or *	<ul style="list-style-type: none"> ■ *ANY: (default) No filtering on member name ■ *PRESENT: Match any item having a member name ■ *ABSENT: Match any item not having a member name
FILE	List of up to ten names or name masks of the member's file and library on which to filter. Name masks may contain ? or *	<ul style="list-style-type: none"> ■ *ANY: (default) No filtering on file name ■ '*ALL/name mask': Filter on file name but not on library name ■ 'name mask/*ALL': Filter on library name but not on file name
SEUTYPE	List of attribute of member's SEU source types on which to filter	<ul style="list-style-type: none"> ■ *ANY: (default) No filtering on source type ■ *PRESENT: Match any item which has a source type other than blank ■ *ABSENT: Match any item which has no source type ■ *S38: Match any item which has an SEUTYPE containing the characters 38 ■ *S36: Match any item which has an SEUTYPE containing the characters 36 ■ *NATIVE: Match any item which does not have characters 38 or 36 in source type <p>Otherwise must be a valid i OS source type</p>

Parameter	Definition	Value and Description
TEXT	Member text on which to filter	<ul style="list-style-type: none">■ *ANY: (default) No filtering is done on member text■ *NONE: Only those members with no text are included■ Character string: Only those members whose text contains the specified string, starting at any position, are included in the list. Up to fifty characters of search string may be specified. Upper and lower case differences are ignored. The character ? can be used as a wild character in the search string
CRTSYSTEM	Filter on System where member was created	<p>*ANY: (default) No filtering on System where member was created</p> <p>Otherwise, specify an IBM i system name for filtering</p>
ASP	Filter on Auxiliary Storage Pool (ASP) where member resides	<p>*ANY: (default) No filtering on ASP</p> <p>Otherwise, specify an Auxiliary Storage Pool number</p>

Parameter	Definition	Value and Description
DATE	Member date on which to filter	<ul style="list-style-type: none"> ■ *ANY: (default) No filtering on date <p>Otherwise, DATE is a list parameter made up of the following four elements:</p> <ul style="list-style-type: none"> ■ Date type <ul style="list-style-type: none"> – *CRT: Filter on date member created – *CHG: Filter on date member last changed – *USD: Filter on date member last used – *URS: Filter on date usage counter was last resent ■ Date operator <ul style="list-style-type: none"> – *EQ: (default) Equal to – *NE: Not equal to – *LT: Less than – *GT: Greater than ■ Date: Entered in system date format (QDATFMT). ■ Time: Entered in HHMMSS format
USAGE	Number of days since Usage Counter was reset	<ul style="list-style-type: none"> ■ *ANY: (default) No filtering on number of days used <p>Otherwise, USAGE is a list of parameters made up of the following two elements:</p> <ul style="list-style-type: none"> ■ Days used operator <ul style="list-style-type: none"> – *EQ: (default) Equal to – *NE: Not equal to – *LT: Less than – *GT: Greater than ■ Days used entered as a number

Parameter	Definition	Value and Description
NBRRCDs	Number of records in member on which to filter	<ul style="list-style-type: none">■ *ANY: (default) No filtering on number of records <p>Otherwise, NBRRCDs is a list parameter made up of the following four elements:</p> <ul style="list-style-type: none">■ Record type<ul style="list-style-type: none">– *RCD: Filter on number of active records– *DLT: Filter on number of deleted records■ Date Operator<ul style="list-style-type: none">– *EQ: (default) Equal to– *NE: Not equal to– *LT: Less than– *GT: Greater than
FLAGVAL	Flag value on which to filter	<ul style="list-style-type: none">■ *ANY: (default) No filtering is done on flag value <p>Otherwise, FLAGVAL is a list parameter made up of the following two elements:</p> <ul style="list-style-type: none">■ Relational operator for selection of flags<ul style="list-style-type: none">– *EQ: (default) Equal to– *NE: Not equal to■ Flag value<ul style="list-style-type: none">Single character flag value or one of the special flag values
UPDLST	List update option	<p>Up to two of the following values</p> <ul style="list-style-type: none">■ *RMVERR:(default) Remove items from list that do not meet the filter criteria■ *FLAGOK: Flag the items in the list that meet the filter criteria. Flag with the flag specified by the OUTFLAGVAL parameter
MBRLST	Qualified name of member list which is to be filtered	QTEMP/TEMPLST: (default) List name

Parameter	Definition	Value and Description
OUTFLAGVAL	Flag value given to selected items in resulting list if *FLAGOK is specified for the UPDLST parameter	Single character flag value or one of the special flag values.
OUTLST	Qualified name of copied output list with selected entries	<ul style="list-style-type: none"> ■ *NONE: (default) Selected list entries are not copied to an output list ■ QTEMP/TEMPLSTOUT: (default) Output list name
OUTLSTOPT	Output list replacement option for the list specified by the OUTLST parameter	<ul style="list-style-type: none"> ■ *REPLACE: (default) Create a new output list, replacing any previous list's contents ■ *ADD: Add to any existing list's contents

Notes

- A name mask is used to specify which names are to be selected.
See the command diagram for the command Filter Database File List (YFLTDBFLST) for details of name mask selection.
- Rules for combining selection parameters are as follows:
 - If *ANY is specified for a parameter that parameter is not used for filtering.
 - Different parameters are ANDed together:
The following would select only PF source members in file QDDSSRC:
YFLTMBRLST FILE(QDDSSRC)SEUTYPE(*PF)
 - Where a list of values may be specified for a parameter the list elements are ORed together:
The following would select any members of type RPG or CLP:
YFLTMBRLST SEUTYPE(*RPG *CLP)
 - Each element of the file name list can consist of a qualified file name. For example, the file name plus library name. If both file and library name are specified they are ANDed together. The following would select files whose names begin with the letters ABC and are in any library having a name beginning with the letters XYZ:
YFLTMBRLST FILE(ABC*XYZ*)

3. The following table lists all valid combinations of the UPDLST and OUTLST parameters with the actions taken depending on the FILTER parameter

Parameter	Condition									
FILTER(*SELECT)	Y	Y	Y	Y	Y					
FILTER(*OMIT)						Y	Y	Y	Y	Y
OUTLST(*NONE)	Y	Y	Y			Y	Y	Y		
OUTLST(list-name)				Y	Y				Y	Y
UPDLST(*RMVERR)	Y	Y		Y		Y	Y			
UPDLST(*FLAGOK)		Y	Y		Y		Y	Y		Y
	:	:	:	:	:	:	:	:	:	:
	:	:	:	:	:	:	:	:	:	:
Action	:	:	:	:	:	:	:	:	:	:
Remove unselected from input	X	X								
Remove selected from input						X	X			
Flag unselected on input							X	X		
Flag selected on input		X	X							
Copy selected to output				X	X					X
Copy unselected to output					X				X	X
Flag copied items on output					X					X

4. See [Appendix A](#) for information on the special values for Special Flag Values for FLAGVAL and OUTFLAGVAL parameters.

Example

To select the RPG members in member list FRED whose names contain the letters 'ORD' and have been changed since 16/04/84:

```
YFLTMRLST MBR('ORD*') SEUTYPE(*RPG) DATE(*CHG *GT 160484 000000) MBRLST(FRED)
```

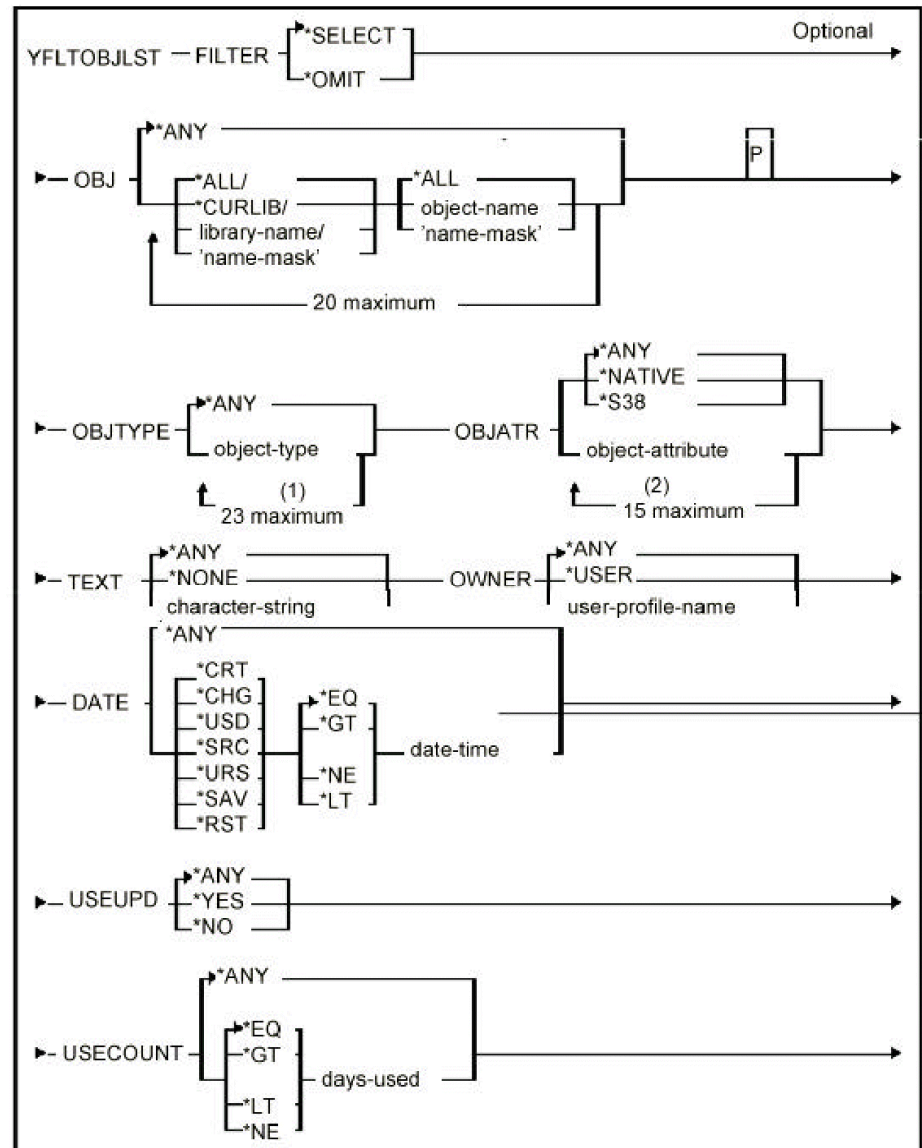
To copy all entries of type SQLRPG which do not have flag *SUBMITTED into member list SQLRECOMP in library QTEMP and set the flag to *OFF on copied entries:

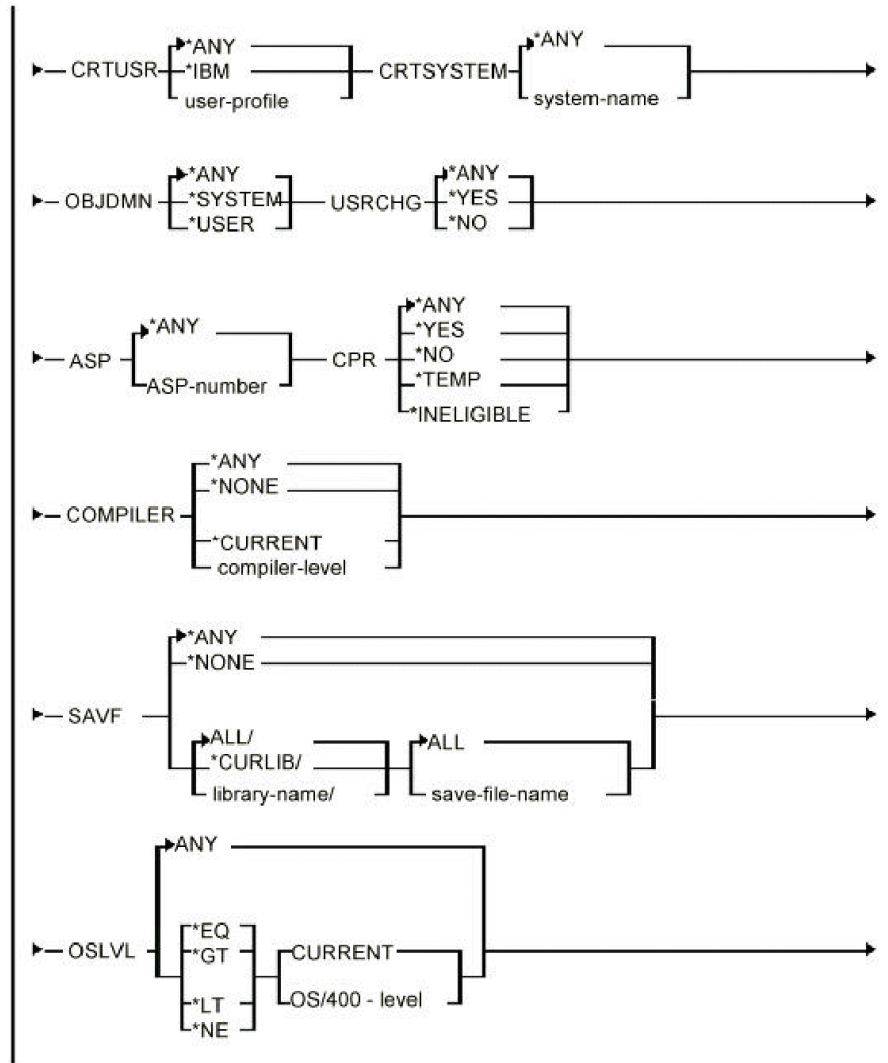
```
YFLTMRLST SEUTYPE(*SQLRPG) FLAGVAL(*SUBMITTED) UPDLST(*RMVERR *FLAGOK)
OUTLST(SQLRECOMP) OUTFLAGVAL
```

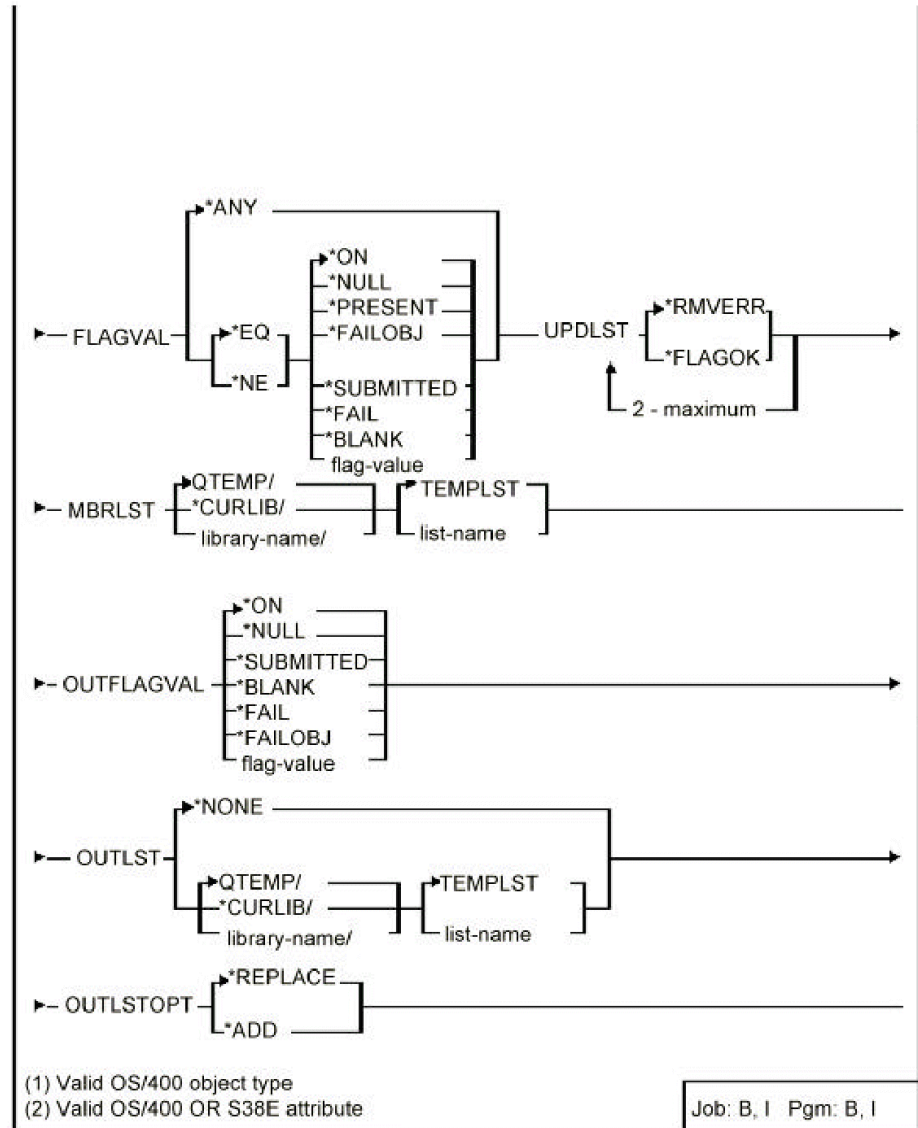
YFLTOBJLST (Filter Object List)

This command removes or flags unwanted items from an existing object list.

Syntax Diagram







Parameters

Parameter	Definition	Value and Description
FILTER	Type of filtering required	<ul style="list-style-type: none">■ *SELECT: (default) Keep items which satisfy the filtering conditions■ *OMIT: Reject items which satisfy the filtering conditions
OBJ	List of qualified names or name masks of objects and libraries on which to filter. Name masks may contain '*' and ? characters	<ul style="list-style-type: none">■ *ANY: (default) No filtering is done on object names■ '*ALL/name mask': Filter on object name but not on library name.■ 'name mask/*ALL': Filter on library name but not on object name
OBJTYPE	List of object types on which to filter	*ANY: (default) no filtering is done on type must be a valid i OS object type
OBJATR	List of object attributes on which to filter	Either one of the following single values <ul style="list-style-type: none">■ *ANY: (default) No filtering is done on attribute■ *NATIVE: Must be a valid native i OS object attribute■ *S38: Must be a valid i OS S38 object attribute Or a list of up to fifteen of the following elements: <ul style="list-style-type: none">■ Any valid i OS object attribute■ *PHY: Equivalent to using both *PF and *PF38■ *LGL: Equivalent to using both *LF and *LF38■ *DSP: Equivalent to using both *DSPF and *DSPF38■ *PRT: Equivalent to using both *PRTF and *PRTF38
OWNER	Object owner to filter	<ul style="list-style-type: none">■ *ANY:(default) No filtering is done on owner■ *USER: Filter on user profile of current job

Parameter	Definition	Value and Description
DATE	Date to filter	<ul style="list-style-type: none"> ■ *ANY: (default) No filtering is done on date <p>Otherwise, DATE is a list parameter made up of the following four elements:</p> <ul style="list-style-type: none"> ■ Date type on which to filter <ul style="list-style-type: none"> – *CRT: Filter on date object was last created – *CHG: Filter on date object was last changed – *SRC: Date source was last changed – *SAV: Date object was last saved – *RST: Date object was last restored – *USD: Date object last used – *URS: Date usage counter was last reset ■ Date operator. <ul style="list-style-type: none"> – *EQ: (default) Equal to – *NE: Not equal to – *LT: Less than – *GT: Greater than ■ Date, entered in system date format ■ Time, entered in HHMMSS format
USEUPD	Usage data collected	<ul style="list-style-type: none"> ■ *ANY: (default) No filtering on usage data collected ■ *YES: Filter on objects for which usage data is collected ■ *NO: Do not filter on objects for which usage data is collected

Parameter	Definition	Value and Description
USECOUNT	Number of days since usage counter was last reset	<ul style="list-style-type: none">■ *ANY: (default) No filtering number of days used Otherwise, USECOUNT is a list of parameters made up of the following two elements: <ul style="list-style-type: none">■ Days used operator<ul style="list-style-type: none">– *EQ: equal to– *NE: not equal to– *LT: less than– *GT: greater than■ Days used: entered as a number
CRTUSR	Name of creator user profile on which to filter	<ul style="list-style-type: none">■ *ANY: (default) No filtering is done on creator■ *IBM: Filter on objects which were not created by any user profile
CRTSYSTEM	Filter on system where object was created	*ANY: (default) N filtering on system where object was created Otherwise, specify an IBM i (IBM i) system name for filtering
OBJDMN	Object domain on which to filter	<ul style="list-style-type: none">■ *ANY: (default) No filtering on object domain■ *SYSTEM: The object is in the system domain■ *USER: The object is in the user domain
USRCHG	Filter on objects modified by a user	<ul style="list-style-type: none">■ *ANY: (default) No filtering on user modified■ *YES: Object is modified by a user■ *NO: Object is not modified by a user
ASP	Filter on auxiliary storage pool (ASP) where object resides	*ANY: (default) No filtering on ASP Otherwise, specify an auxiliary storage pool number

Parameter	Definition	Value and Description
CPR	Compression status on which to filter	<ul style="list-style-type: none"> ■ *ANY: (default) No filtering on compression status ■ *YES: Filter on permanently compressed objects ■ *NO: Filter on objects eligible for compression but not compressed. ■ *TEMP: Filter on objects temporarily compressed ■ *INELIGIBLE: Filter is ineligible for compression <p>Compiler</p> <p>Compiler level at object creation on which to filter</p> <ul style="list-style-type: none"> ■ *ANY: (default) No filtering is done on compiler level ■ *NONE: Filter on objects not compiled ■ *CURRENT: Filter on compiler level same as i OS level of machine executing the filter <p>Otherwise, COMPILER is a decimal number VRR.MM, where V is Version, RR is release, and MM is machine level</p>
SAVF	Qualified name of save file on which to filter	<ul style="list-style-type: none"> ■ *ANY: (default) Do not filter on save file ■ *NONE: Filter on objects which have not been saved to a save file ■ Save file *ALL: Do not filter on file name ■ Save library *ALL: Do not filter on library name

Parameter	Definition	Value and Description
OSLVL	i OS level at which object was created	<ul style="list-style-type: none">■ *ANY: (default) no filtering will be done on i OS level■ *CURRENT: filter on i OS level of machine executing the filter Otherwise, OSLVL is a decimal number VRR.MM, where V is Version, RR is release, and MM is machine level■ Relational operator for compiler level filtering<ul style="list-style-type: none">– *EQ: (default) equal to– *NE: not equal to– *LT: less than– *GT: greater than■ Four digit number with two decimal places where RR is the release level, and MM is the modification level<ul style="list-style-type: none">– *CURRENT: (default) Use the current i OS level
OBJSIZE	Object size on which to filter	<ul style="list-style-type: none">■ *ANY: (default) No filtering is done on object size Otherwise, OBJSIZE is a list parameter made up of the following two elements:■ Relational operator for object size filtering<ul style="list-style-type: none">– *EQ: (default) Equal to– *NE: Not equal to– *LT: Less than– *GT: Greater than■ Object size in bytes<ul style="list-style-type: none">– *ZERO: (default) Zero bytes

Parameter	Definition	Value and Description
OBJDAMAGE	Object damage status on which to filter	<ul style="list-style-type: none"> ■ *ANY: (default) No filtering is done on object damage status ■ *DAMAGED: Only damaged objects are to be included in the list ■ *UNDAMAGED: Only undamaged objects are to be included in the list
TEXT	Object text on which to filter	<ul style="list-style-type: none"> ■ *ANY: (default) No filtering is done on object text ■ *NONE: Only those objects with no text included ■ Character string: Only those objects whose text contains the specified string, starting at any position, is included in the list. Up to fifty characters of search string can be specified. Upper and lower case differences are ignored. The character ? can be used as a wild character in the search string
FLAGVAL	Flag value on which to filter	<ul style="list-style-type: none"> ■ *ANY: (default) No filtering is done on flag value <p>Otherwise, FLAGVAL is a list parameter made up of the following two elements:</p> <ul style="list-style-type: none"> ■ Relational operator for selection of flags <ul style="list-style-type: none"> – *EQ: (default) Equal to – *NE: Not equal to ■ Flag value <ul style="list-style-type: none"> – Single character flag value or one of the special flag values
UPDLST	List update option. Up to two of the following values	<ul style="list-style-type: none"> ■ RMVERR: Remove items from list that do not meet the filter criteria ■ *FLAGOK: Flag list items that meet the filter criteria. Flag with the flag specified by the OUTFLAGVAL parameter

Parameter	Definition	Value and Description
OBJLST	Qualified name of the filtered object list	QTEMP/TEMPLST: (default) List name
OUTFLAGVAL	Flag value given to selected items in resulting list	<ul style="list-style-type: none">■ *FLAGOK is specified for the UPDLST parameter■ Single character flag value or one of the special flag values
OUTLST	Qualified name of output list where selected entries are copied	<ul style="list-style-type: none">■ *NONE: (default) Selected list entries are not copied to an output list■ QTEMP/TEMPLST: (default) output list name
OUTLSTOPT	Output list replacement option specified by the OUTLST parameter	<ul style="list-style-type: none">■ *REPLACE: (default) Create a new output list, replacing any previous list's contents■ *ADD: Add to any existing list's contents

Notes

1. A name mask is used to specify which names are to be selected.
See the command diagram for the command Filter Database File List (YFLTDBFLST) for details of name mask selection.
2. Rules for combining selection parameters are as follows:
 - If *ANY is specified for a parameter, that parameter is not used for filtering.
 - Different parameters are ANDed together:
The following would select only programs in object list QTEMP/TEMPLST owned by the QSECOFR user profile:
`YFLT0BJLST OBJTYPE(*PGM) OWNER(QSECOFR)`
 - Where a list of values may be specified for a parameter the list elements are ORed together:
The following would select any programs type RPG or CL in object list QTEMP/TEMPLST:
`YFLT0BJLST OBJATR(*RPG *CL)`
 - Each element of the object name list can consist of a qualified object name, i.e. an object name plus a library name. If both an object and a library name are specified they are ANDed together. The following would select all objects whose names begin with the letters ABC and that are in any library having a name beginning with the letters XYZ:

YFLT0BJLST OBJ(ABC* XYZ*)

3. The table below lists all valid combinations of the UPDLST and OUTLST parameters with the actions taken depending on the FILTER parameter:

Parameter	Condition									
FILTER(*SELECT)	Y	Y	Y	Y	Y					
FILTER(*OMIT)						Y	Y	Y	Y	Y
OUTLST(*NONE)	Y	Y	Y			Y	Y	Y		
OUTLST(list-name)				Y	Y				Y	Y
UPDLST(*RMVERR)	Y	Y		Y		Y	Y			
UPDLST(*FLAGOK)		Y	Y		Y		Y	Y		Y
	:	:	:	:	:	:	:	:	:	:
	:	:	:	:	:	:	:	:	:	:
Action	:	:	:	:	:	:	:	:	:	:
Remove unselected from input	X	X								
Remove selected from input						X	X			
Flag unselected on input							X	X		
Flag selected on input		X	X							
Copy selected to output				X	X					X
Copy unselected to output					X				X	X
Flag copied items on output					X					X

4. Special Flag Values for FLAGVAL and OUTFLAGVAL parameters

See [Appendix A](#) for information on the special values for these parameters.

Example

To move all objects changed after a certain date, say 04/16/91, from library QGPL to library FRED you could:

Build an object list:

```
YBLD0BJLST OBJ(QGPL/*ALL) OBJTYPE(*ALL)
```

Filter the list:

```
YFLT0BJLST DATE(*CRT *GT 160484)
```

Use the list to drive the move object command:

```
YMOV0BJ OBJ(*OBJLST) OBJTYPE(*ALL) TOLIB(FRED)
```

To set off all flags in an object list FRED in library QGPL:

```
YFLT0BJLST OBJLST(FRED.QGPL) UPDLST(*FLAGOK)  
OUTFLAGVAL(*OFF)
```

To flag all entries in the object list FRED in library QGPL which have text containing the character string GO with the flag M, and simultaneously remove entries not containing the string GO:

```
YFLT0BJLST TEXT('GO') OBJLST(FRED.QGPL)  
UPDLST(*RMVERR *FLAGOK) OUTFLAGVAL(M)
```

To remove all entries in object list TEMPLST in library QTEMP which do not have flag value O (*FAILOBJ fail object), and which had owner JOHN, at the same time resetting those which were not removed to Y (*ON):

```
YFLT0BJLST FILTER(*OMIT) OWNER(JOHN) FLAGVAL(*NE *FAILOBJ) UPDLST(*RMVERR  
*FLAGOK) OUTFLAGVAL(*ON)
```

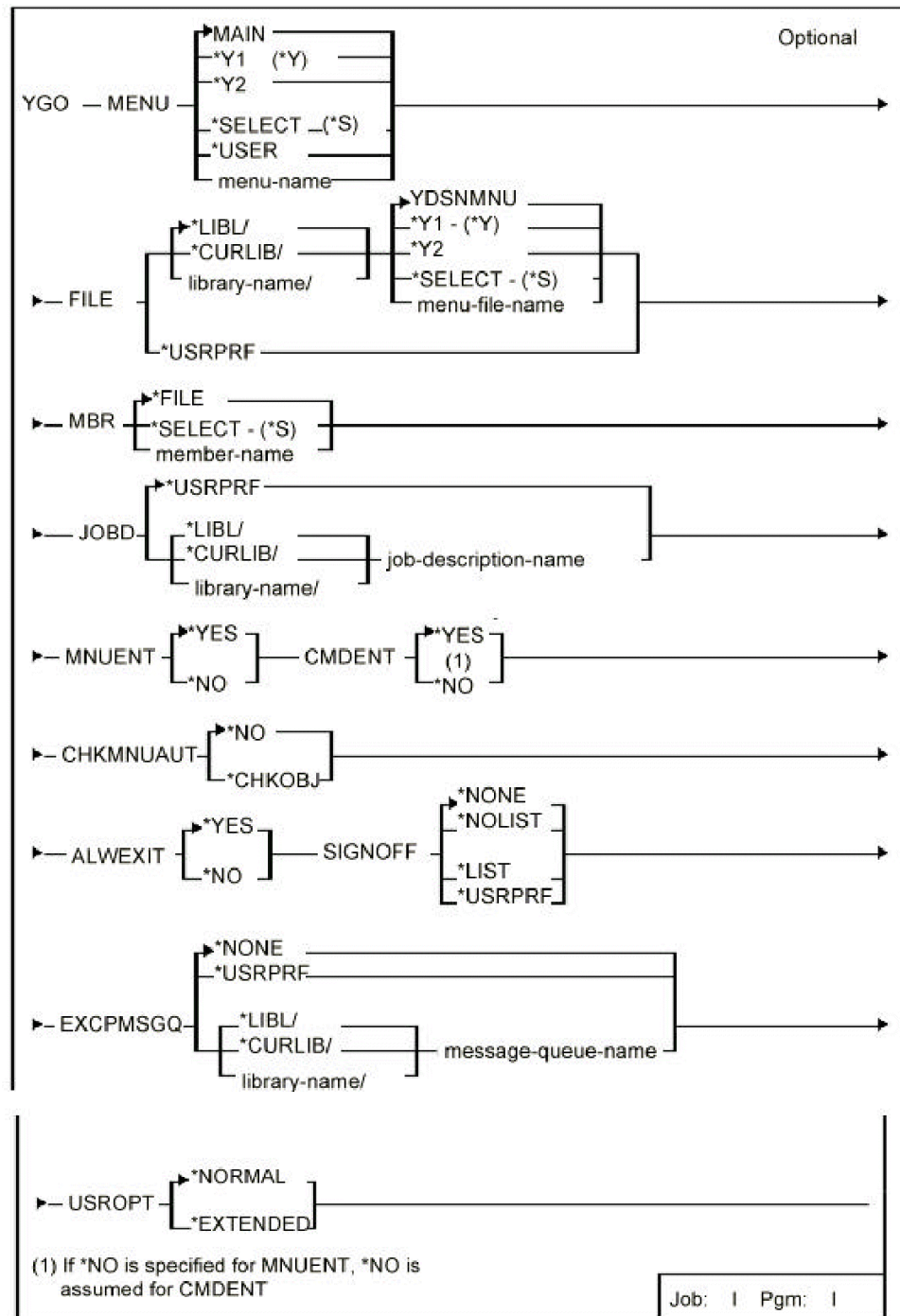
To add entries from object list SYSTEM in library QGPL which have object damaged into object list DAMAGED in library QGPL setting the flag on the copied entries to A:

```
YFLT0BJLST OBJDAMAGE(*DAMAGED) OBJLST(QGPL/SYSTEM)  
UPDLST(*FLAGOK *RMVERR) OUTFLAGVAL(A)  
  
OUTLST(QGPL/DAMAGED) OUTLSTOPT(*ADD)
```

YGO (Go to Menu)

This command displays type menus, starting with a specified menu.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
MENU	Name of first menu which is displayed	<ul style="list-style-type: none">■ MAIN: (default) Menu named MAIN is displayed■ *Y1: The master help menu for the utilities is displayed■ *Y2: The master help menu for the application generator is displayed■ *SELECT: A list of available menus are displayed, one of which may be selected■ *USER: A menu with the same name as the current user's user profile is displayed
FILE	Qualified name of the file containing menus	<ul style="list-style-type: none">■ YDSNMNU: (default) File name■ *USRPRF: The menu file specified for the user profile with the command Change User Profile (YCHGUSRPRF) is used■ *Y1: Help menu file■ *Y2: Help menu file■ *SELECT: A list of available menu files are displayed The file must be created with the YCRTDSNF command
MBR	Name of member in file containing the menus	<ul style="list-style-type: none">■ *FILE: (default) The member name will be assumed to be the same as the file name.■ *SELECT: A list of available menu file members are displayed
JOBID	Qualified name of job description to be used by the menu program when submitting jobs	*USRPRF: (default) Job description specified for the user profile is used

Parameter	Definition	Value and Description
MNUENT	Direct menu name entry allowed option	<ul style="list-style-type: none"> ■ *YES: (default) Allow entry of menu names ■ *NO: Do not allow menu name entry
CMDENT	Command entry allowed option	<ul style="list-style-type: none"> ■ *YES: (default) Allow entry of command requests ■ *NO: Do not allow command entry
CHKMNUAUT	Authority checking of menu options	<ul style="list-style-type: none"> ■ *NO: (default) No authority checking of options in menu ■ *CHKOBJ: Authority checking of options in menu to take place
ALWEXIT	Allow exit from top menu by pressing F03	<ul style="list-style-type: none"> ■ *YES: (default) Allow exit via F03 ■ *NO: Do not allow exit via F03
SIGNOFF	Effect of taking so from any menu display	<ul style="list-style-type: none"> ■ *NONE: (default) Return to caller ■ *NOLIST: Execute SIGNOFF(*NOLIST) ■ *LIST: Execute SIGNOFF(*LIST) ■ *USRPRF: Execute SIGNOFF using signoff option from user profile extension attributes
EXCPMSGQ	Qualified name of message queue to which a copy of any exception message received by the display menu program is to be sent	<ul style="list-style-type: none"> ■ *NONE: (default) Do not send a copy of exception messages ■ *USRPRF: The default message queue specified for the user profile is used
USROPT	Display option	<ul style="list-style-type: none"> ■ *NORMAL: (default) Display menu display with SAA standards ■ *EXTENDED: Display menu display with extended display options, including date and time

Notes

1. This command calls an interactive display to display the menus. Press Help on the selection or command field for instructions on how to use menus. Pressing Help with the cursor in the option or header area of the menu causes the context-sensitive help defined for those areas to be displayed.
2. Menu option requests will be logged according to the i OS job logging convention. For instance, to log everything:

`CHGJOB LOG(4 00 *SECLVL)`

Command requests will not be logged.
3. The ALWEXIT parameter enables you to make sure that a user does not inadvertently signoff by pressing the exit key.

The SIGNOFF parameter enables you to let the user signoff directly, even though the user may be several invocation levels deep in the menu system.
4. If an exception message queue is specified using the EXCPMSGQ parameter, the user must have data add rights to it.
5. Use of the Go to Menu utility is independent of CA 2E. The Go to Menu utility may be used without CA 2E being present in your library list. See Appendix B of this Guide for further details.

Example

To display menu FRED in file YDSNMNU:

```
YGO MENU(FRED)
```

To display the Synon/1E help menu:

```
YGO MENU(*Y1)
```

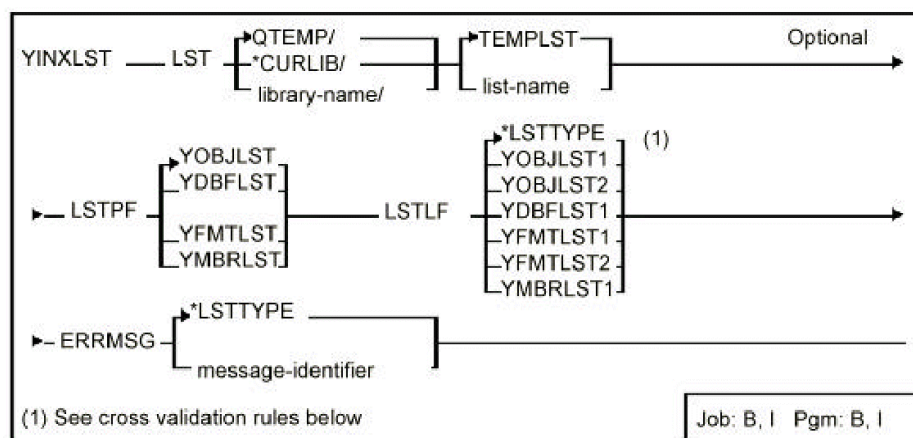
To display menu FRED in file YDSNMNU with the menu name and command request fields protected:

```
YGO MENU(FRED) FILE(YDSNMNU) MNUENT(*NO) CMDENT(*NO)
```

YGRTPRDAUT (Grant Product Authority)

This command allows you to enter a code to enable the products.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
LST	Qualified name of list which is to be indexed	QTEMP/TEMPLST: (default) List name
LSTPF	Name of physical file containing list	<ul style="list-style-type: none"> YOBJLST: (default) Object list (QADSPOBJ) YDBFLST: Database file list (QAFDMBR) YFMTLST: Format list (QAFDRFMT) YMBRLST: Member list (QAFDMBRL)

Parameter	Definition	Value and Description
LSTLF	Name of logical file to which member is to be added. See the following table	<ul style="list-style-type: none"> ■ *LSTTYPE: (default) Determine from LSTPF - see table ■ YOBJLST1: Object name/library index ■ YOBJLST2: Object name/type index ■ YFMTLST1: Format name/file/library index ■ YFMTLST2: Format name/format id ■ YDBFLST1: File name/library index ■ YMBRLST1: Member name/file/library index
ERRMSG	Message identifier of message which is to be sent as an escape message if the list is not found. The message must be in the message file, YYYYMSG	*LSTTYPE: (default) Determine from LSTPF

Notes

Allowed values and default values are as follows:

LSTPF	LSTLF	Key fields	ERRMSG
YOBJLST YOBJLST	YOBJLST1 YOBJLST2	ODOBNM, ODLBNM, ODOPTP, ODOBAT ODOBNM, ODLBTP	YOL0001 YOL0001
YDBFLST	YDBFLST1	PHFILE, PHLIB	YDL0001
YFMTLST YFMTLST	YFMTLST1 YFMTLST2	RFNAME, RFFILE, RFLIB RFNAME, RFID	YFL0001 YFL0001
YMBRLST	YMBRLST1	MLNAME, MLFILE, MLLIB	YML0001

Default values used if *LSTTYPE is specified are shown in bold.

Example

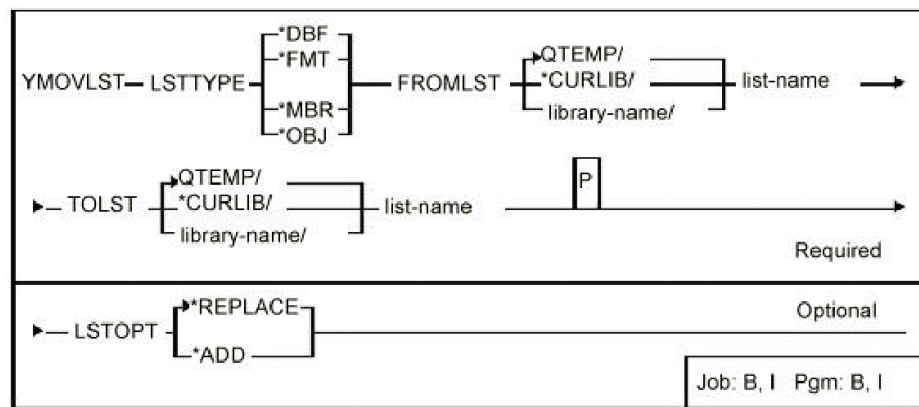
To add an index to object list FRED in library QTEMP. Index is to be same format as YOBJLST1 file. Message USR0101 is to be issued if the list is not added:

```
YINXLST LST(QTEMP/FRED) LSTPF(YOBJLST) LSTLF(YOBJLST1) ERRMSG(USR0101)
```

YMOVLST (Move List)

This command moves an object, member, format or database file list to another list file.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
LSTTYPE	Type of list moved	<ul style="list-style-type: none"> ■ *DBF: Database file list ■ *FMT: Format list ■ *MBR: Member list ■ *OBJ: Object list
FROMLST	Qualified name of moved list	

Parameter	Definition	Value and Description
TOLST	Qualified name of destination list	
LSTOPT	List replacement option	<ul style="list-style-type: none"> ■ *REPLACE: (default) Replace any existing contents of the list identified by TOLST parameter ■ *ADD: Add to any existing TOLST contents

Notes

None

Example

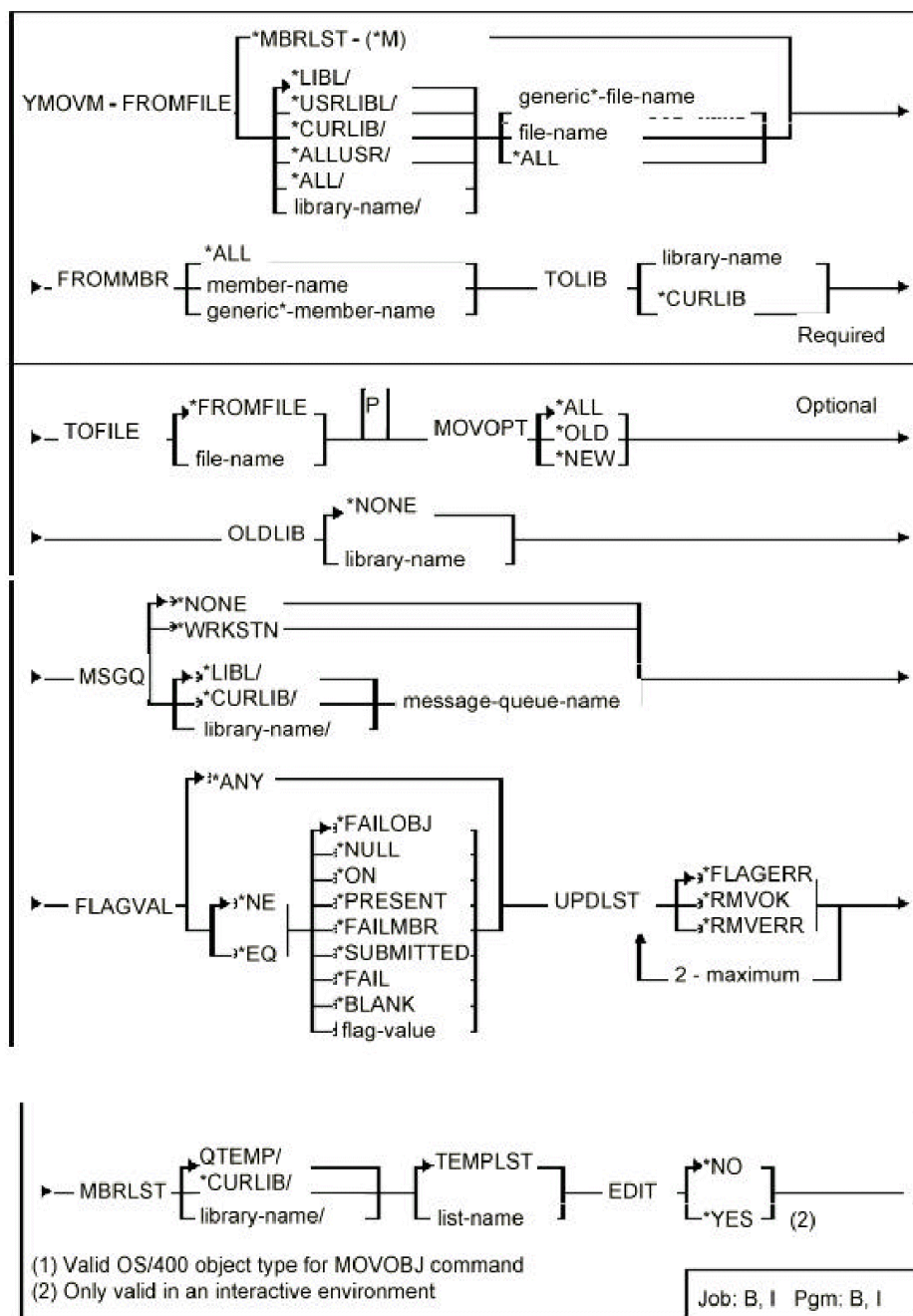
To move an object list PORT from one library to another:

```
YMOVLST LSTTYP(*OBJ) FROMLST(FRED/PORT) TOLST(QGPL/PORT)
```

YMOVM (Move Member)

This command moves a list of source members. The list can either be specified by a generic name, or by using a member list. Movement consists of copying the member to the destination file, then removing the original member. The old version of the member may also be archived.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
FROMFILE	Qualified generic name of file containing members which are to be moved	<ul style="list-style-type: none"> ■ *MBRLST: (default) Move members named in member list specified with the MBRLST parameter ■ *ALL: Move all files in the specified library
FROMMBR	Generic name of members to be moved	*ALL: Move all members in the specified file or files
TOLIB	Destination library to which members are to be moved. A copy of the from file must already exist in the to library	*CURLIB (default) Move members to a file in the invoking job's current library
TOFILE	Name of file in destination library	*FROMFILE: (default) Each member is moved to a file of the same name as the file in the from library containing the member
MOVOPT	Movement option	<ul style="list-style-type: none"> ■ *ALL: (default) All the found members are moved ■ *OLD: Only existing members in the destination file are to be moved ■ *NEW: Only found members that do not already exist in the destination file are moved
OLDLIB	Name of archive library	*NONE: (default) No archive copy is kept, If the member already exists in a file in the library specified by OLDLIB, it is replaced

Parameter	Definition	Value and Description
MSGQ	Qualified name of message queue to which a completion message is to be sent for each member moved. May be used to provide a control log of program implementation	<ul style="list-style-type: none">■ *NONE: (default) No message is sent.■ *WRKSTN: Messages is sent to the current workstation's message queue
FLAGVAL	Flag value of list items to move	<p>A list parameter made up of the following two elements:</p> <ul style="list-style-type: none">■ Relational operator for selection of flags<ul style="list-style-type: none">– *EQ: (default) Equal to– *NE: Not equal to■ Flag value<ul style="list-style-type: none">Single character flag value or one of the special flag values.Otherwise *ANY executes all list entries
UPDLST	List update option	<p>Up to two of the following values:</p> <ul style="list-style-type: none">■ *FLAGERR: (default) List items not successfully moved are flagged with M (*FAILMBR)■ *RMVOK: List items successfully moved are removed from the list■ *RMVERR: List items not successfully moved are removed from the list

Parameter	Definition	Value and Description
MBRLST	Qualified name of member list indicating members that are to be moved	QTEMP/TEMPLST: (default) List name Specify FROMFILE(*MBRLST) to use an existing member list
EDIT	Edit list required	<ul style="list-style-type: none"> ■ *NO: (default) No editing required ■ *YES: The edit member list function is invoked to edit the list before execution

Notes

If UPDLST(*FLAGERR) is specified, and a source member cannot be moved, the member list entry is flagged with an M (*FAILMBR). Member list entries for source members successfully moved is always flagged with *BLANK. This may help you when errors occur on some but not all members of the list.

Examples

To move all members beginning with the letter W from file QRPGRSRC in library FRED to file QRPGRSRC in library DERF:

```
YMMBMVOM FROMFILE(FRED/QRPGRSRC) FROR(W*) TOLIB(DERF)
```

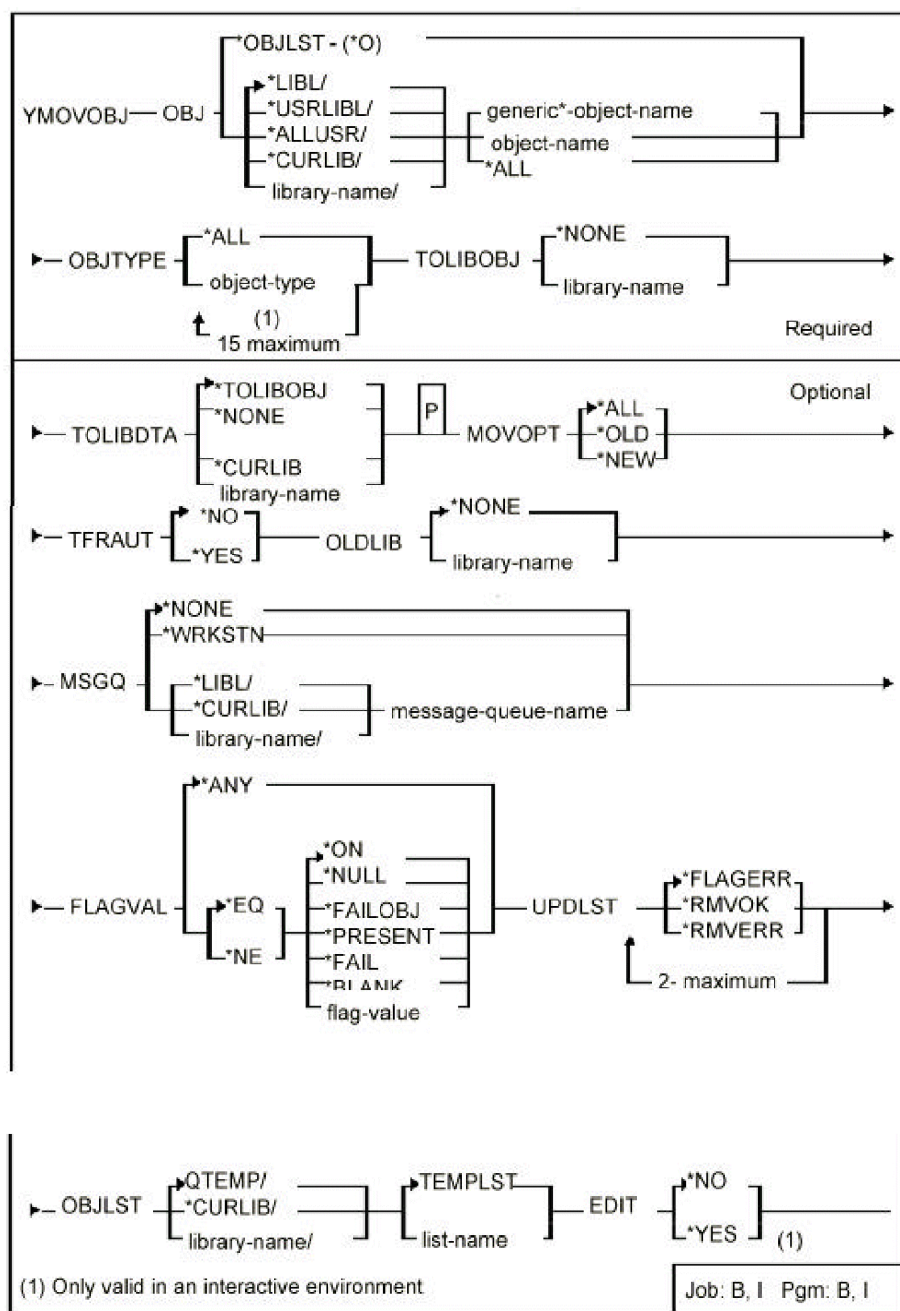
To move all members in member list SHEBA to library SOLOMON:

```
YMOVOM FROMFILE(*MBRLST) FROMMBR(*ALL) TOLIB(SOLOMON)MBRLST(SHEBA)
```

YMOV OBJ (Move Objects)

This command moves a list of objects. The list can either be specified by a generic name or by using a list. Data and application objects can be directed to different destination libraries, and a log kept of movements. Authorizations can be preserved.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
OBJ	Qualified generic name of moved objects	<ul style="list-style-type: none"> ■ *OBJLST: An object list specified by the OBJLST parameter, names the objects which are to be moved ■ *ALL: All objects in the specified libraries are to be moved
OBJTYPE	Lists object to be moved	<p>*ALL: All object types are to be moved.\</p> <p>If OBJ(*OBJLST) is specified then *ALL should be specified for OBJTYPE. Note that certain i OS object types may not be moved</p>
TOLIBOBJ	Library to which non-data objects are to be moved	<ul style="list-style-type: none"> ■ *NONE: Non-data objects are not moved ■ *CURLIB: Move non-data objects to current library
TOLIBDTA	Destination library for data objects. Data objects include physical and logical files, and data areas	<ul style="list-style-type: none"> ■ *TOLIBOBJ: (default) Move data objects to the library specified by the TOLIBOBJ parameter ■ *CURLIB: Move data objects to current library ■ *NONE: Data objects are not to be moved *NONE may not be specified for both TOLIBOBJ and TOLIBDTA
MOVOPT	Movement option	<ul style="list-style-type: none"> ■ *ALL: (default) All the found objects are to be moved ■ *OLD: Only found objects that already exist in the destination library are to be moved ■ *NEW: Only found objects that do not already exist in the destination library are to be moved

Parameter	Definition	Value and Description
TFRAUT	Transfer authorities	<ul style="list-style-type: none">■ *NO: (default) No authorities are to be transferred■ *YES: Any authorities that any existing version of the object in the library named by the TOLIB parameter possesses is granted to the object that replaces it
OLDLIB	Library to place versions of the objects that already exist in the destination library.	*NONE: (default) If a copy of the object already exists in the destination library, that copy is deleted
MSGQ	Qualified name of message queue to which a completion message is to be sent for each object moved. May be used to provide a control log of program implementation	<ul style="list-style-type: none">■ *NONE: (default) No message is sent■ *WRKSTN: Messages are sent to the current workstation's message queue
FLAGVAL	Flag value of list items to move	<ul style="list-style-type: none">■ *ANY: (default) Move objects regardless of the flag values of their list entries <p>Otherwise, FLAGVAL is a list parameter made up of the following two elements:</p> <ul style="list-style-type: none">■ Relational operator for selection of flags<ul style="list-style-type: none">– *EQ: (default) Equal to– *NE: Not equal to■ Flag value<ul style="list-style-type: none">– Single character flag value or one of the special flag values

Parameter	Definition	Value and Description
UPDLST	List update option. Up to two of the following values	<ul style="list-style-type: none"> ■ *FLAGERR: (default) Flag the list entries for those objects which are not successfully moved with O (*FAILOBJ) ■ *RMVOK: Remove the list entries for those objects which are successfully moved ■ *RMVERR: Remove the list entries for those objects items which are not successfully moved
OBJLST	Qualified name of object list that specifies objects that are to be moved	QTEMP/TEMPLST: (default) List name Specify OBJ(*OBJLST) to use an existing list
EDIT	Edit list	<ul style="list-style-type: none"> ■ *NO: (default) Do not invoke list edit function ■ *YES: The edit list function is to be invoked before executing the move

Notes

- To safeguard against the inadvertent loss of data, objects will not be moved under the following conditions:
 - The object being moved is a physical file or data area
 - And OLDLIB(*NONE) is specified
 - The physical file or data area already exists in the library specified by the TOLIBDTA parameter

For example, you cannot use YMOVOBJ to move and replace a physical file unless a value other than *NONE is specified for the OLDLIB parameter.
- The YMOVOBJ command will ignore any objects residing in libraries whose names begin with the letter Q. However objects in libraries QGPL and QTEMP will be moved.
- If UPDLST(*FLAGERR) is specified and an object could not be moved (for instance, due to insufficient authority), then the flag field in the object list will be flagged with an O,(*FAILOBJ). Objects that are moved successfully will be flagged with a *BLANK. This may help you to identify items in the list which still need moving.
- Note that if a name, or a generic name, is specified for the OBJ parameter, then a new object list will be built.

Examples

To move all programs whose names begin with the letter W from library FRED to library FERDINAND, saving any existing objects from FERDINAND in FREDOLD:

```
YMOVOBJ OBJ(FRED/W*) OBJTYPE(*PGM) TOLIBOBJ(FERDINAND) OLDLIB(FREDOLD)
```

To move all objects in object list MAHOMET to library MOUNTAIN:

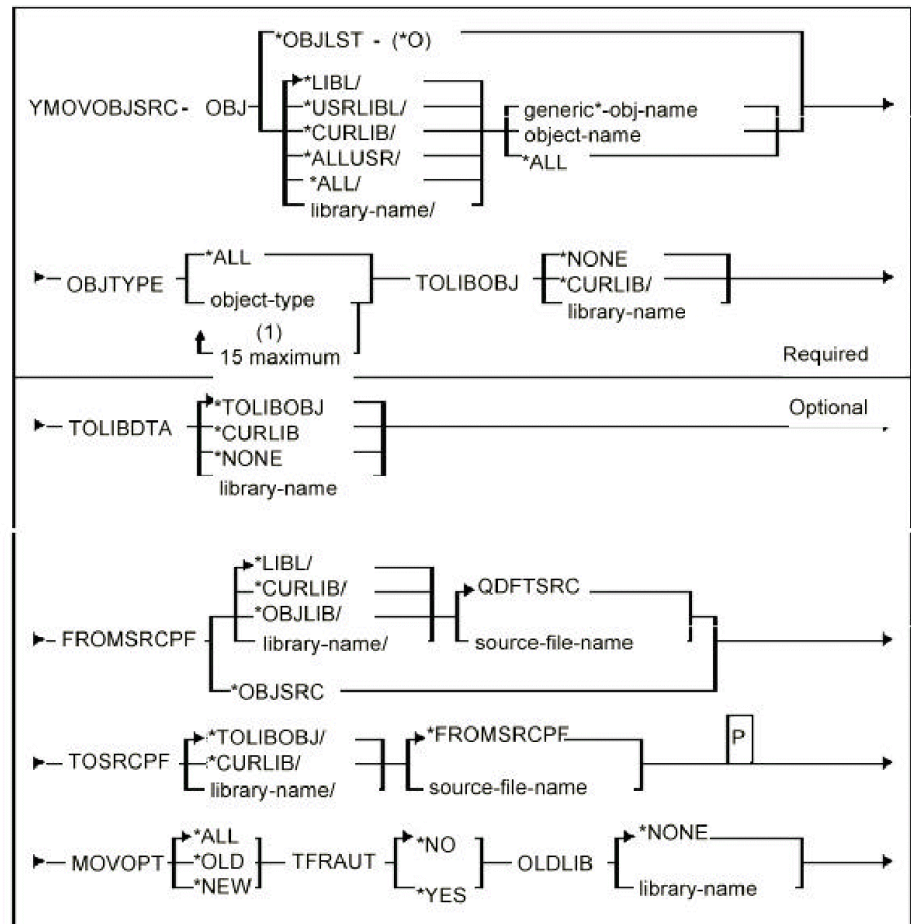
```
YMOVOBJ OBJ(*OBJLST) OBJTYPE(*ALL) TOLIBOBJ(MOUNTAIN) OBJLST(MAHOMET)
```

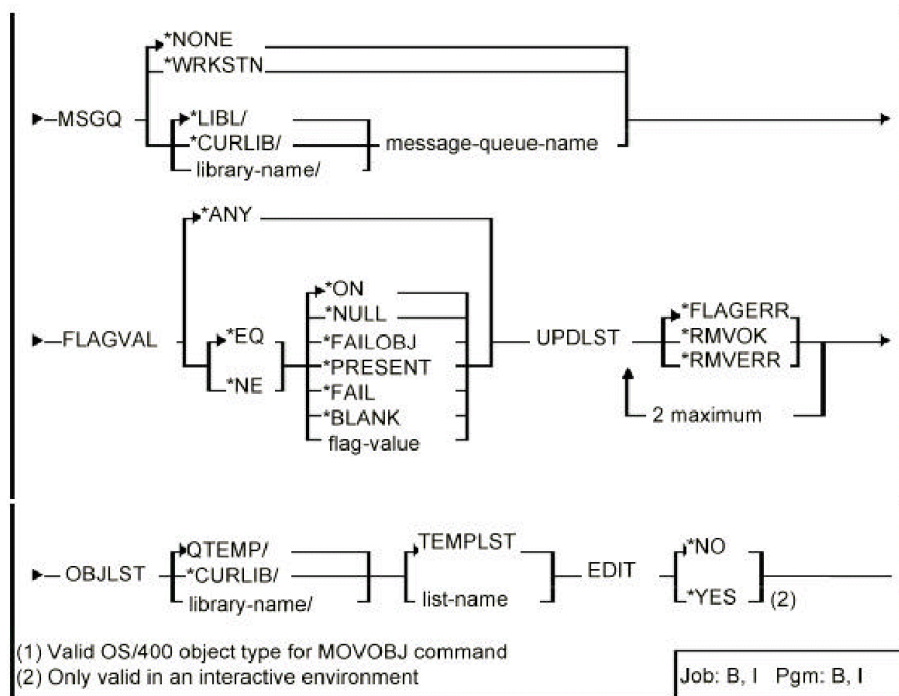
YMOVOBJSRC (Move Objects & Source)

This command moves a list of objects and their accompanying source members. The list may either be specified by a generic name, or by a object list. Data and application objects may be directed to different destination libraries, and a log kept of movements. Authorities can be preserved automatically.

Syntax Diagram

The diagram appears on the following page.





Parameters

Parameter	Definition	Value and Description
OBJ	Qualified generic name of objects moved	<ul style="list-style-type: none"> ■ *OBJLST: Object list specified by OBJLST parameter specifies objects to be moved ■ *ALL: All objects in specified libraries are to be moved
OBJTYPE	Type of object to be moved	*ALL: All object types are to be moved
TOLIBOBJ	Destination library to which non-data objects are to be moved	<ul style="list-style-type: none"> ■ *NONE: Non-data objects are not to be moved ■ *CURLIB: Move non-data objects to current library

Parameter	Definition	Value and Description
TOLIBDTA	Destination library to which data objects are to be moved. Data objects include physical and logical files, and data areas	<ul style="list-style-type: none"> ■ *TOLIBOBJ: (default) Use the library whose name is specified by the TOLIBOBJ parameter ■ *NONE: Data objects are not to be moved. *NONE May not be specified for both destination libraries ■ *CURLIB: Move data objects to current library
FROMSRCPF	Qualified name of source file from which source members are moved	<ul style="list-style-type: none"> ■ *QDFTSRC: (default) The source file defaulted to the appropriate default source file name for the object type and attribute ■ *OBJSRC: The source file and member where the object was created
TOSRCPF	Qualified name of destination source file	<p>*FROMSRCPF: (default) The file name in the destination library is the same as that of the file containing the member in the originating library</p> <p>The library name defaults to the library specified by the TOLIBOBJ parameter</p>
MOVOPT	Movement option	<ul style="list-style-type: none"> ■ *ALL: (default) All objects are moved ■ *OLD: Only found objects that already exist in the destination library are moved ■ *NEW: Only found objects that do not already exist in the destination library are moved

Parameter	Definition	Value and Description
TFAUT	Transfer authorities	<ul style="list-style-type: none">■ *NO: (default) No authorities are transferred■ *YES: Any existing object in the TOLIB library possesses are granted to the object that replaces it
OLDLIB	Library in which to place any old versions of the objects and source already exist in the destination library or libraries. (If a version already exists in the library specified by the OLDLIB parameter	*NONE: (default) Do not keep the previous copy; if a copy of the object already exists in the destination libraries, that copy is deleted (if it is not a data object)
MSGQ	Qualified name of a message queue to which a completion message is to be sent recording each object and each member moved. May be used to provide a control log of program implementation	<ul style="list-style-type: none">■ NONE: (default) No message is sent■ *WRKSTN: Messages are sent to the current workstation's message queue
FLAGVAL	Flag value of list items to move	<ul style="list-style-type: none">■ *ANY: (default) Move objects regardless of the flag values of their list entries <p>Otherwise, FLAGVAL is a list parameter made up of the following two elements:</p> <ul style="list-style-type: none">■ Relational operator for selection of flags<ul style="list-style-type: none">– *EQ: (default) Equal to– *NE: Not equal to■ Flag value<ul style="list-style-type: none">– Single character flag value or one of the special flag values

Parameter	Definition	Value and Description
UPDLST	List update option. Up to two of the following values	<ul style="list-style-type: none"> ■ *FLAGERR: (default) Flag the list entries for those objects not successfully moved with O (*FAILOBJ). The corresponding member list are flagged with M (*FAILMBR) ■ *RMVOK: Remove the list entries for those objects successfully moved ■ *RMVERR: Remove the list entries for those objects items not successfully moved
OBJLST	Qualified name of a object list specifying the objects that are to be moved	QTEMP/TEMPLST: (default) List name Specify OBJ(*OBJLST) to use an existing list
EDIT	Edit list option	<ul style="list-style-type: none"> ■ *NO: (default) Do not invoke the list edit function ■ *YES: The edit list function is to be invoked before executing the move

Notes

- To safeguard against the inadvertent loss of data, objects will not be moved under the following conditions:
 - The object being moved is a physical file or data area
 - OLDLIB(*NONE) is specified.
 - The physical file or data area already exists in the library specified by the TOLIBDTA parameter.
- The YMOVOBJSRC command will ignore any objects residing in libraries whose names begin with the letter Q; however objects in libraries QGPL and QTEMP will be moved.

3. The move command operates in four steps:

- A list of objects to be moved is built/found.
- All objects on the object list are moved. If an error occurs on moving an object and UPDLST(*FLAGERR) is specified, the entry for the object in the object list will be flagged with an O. If an object is moved successfully, the entry is flagged with a *BLANK.
- The object list is converted into a member list of the same name. (The command Convert Object List (YCVTOBJLST) is used to do this.)
- All members on the member list are moved. If an error occurs on moving a member and UPDLST(*FLAGERR) is specified, the entry for the member in the member list will be flagged with an M. If a member is moved successfully, the entry is flagged with a *BLANK.

A diagnostic message is sent for each error. Possible causes of error include lack of authorization, missing source, and inability to allocate objects.

If errors occur you will probably want to rerun the command to move just the objects or members for which errors occurred. Note that if the objects are moved successfully but the source is not, you could restart from step (d) by using the command Move Member (YMOVMM)

4. The transfer authority option saves you from having to regrant authorities. If the object's ownership is to be changed as well, you can run the generic command Change Ownership (YCHGOBJOWN) for the same list of objects before moving them:

```
YCHGOBJOWN OBJ(*OBJLST) OBJTYPE(*ALL) NEWOWN(BOYS) OBJLST(ALLMINE)
```

5. If you wish to update the object list to reflect the movement you may use the command Change List (YCHGLST) to alter the library name on the list entries.

Examples

To move all objects whose names begin with the letter W, together with their source, from library TEST to library MCCOY, saving any existing copies of the objects and source in library MCCOY to library FREDOLD:

```
YMOV0BJSRC OBJ(TEST/W*) OBJTYPE(*ALL) TOLIBOBJ(MCCOY) OLDLIB(FREDOLD)
```

To move all objects and their source in list LISTZ to library CHOPIN, flagging all objects and source not successfully moved:

```
YMOV0BJSRC OBJ(*OBJLST) OBJTYPE(*ALL) TOLIBOBJ(CHOPIN) OBJLST(LISTZ)
```

Let us say that the source members for two of the objects in the list are not moved because they are in use - we would get a diagnostic message for each member not moved.

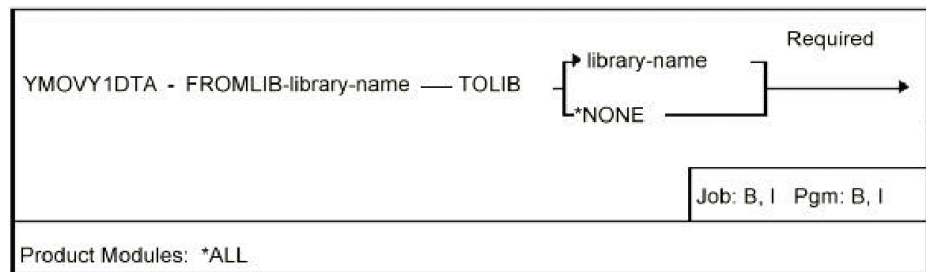
Once we have established that the members are free, we could rerun the move just on the member list:

```
YMOV FROMFILE(*MBRLST) FROMMBR(*ALL) TOLIB(CHOPIN) FLAGVAL(*EQ *FAILMBR)
MBRLST(LISTZ)
```

YMOVY1DTA (Move User Data Objects)

This command moves created user objects that might contain data from one library to another.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
FROMLIB	Name of library from which user data objects are moved	
TOLIB	Name of library to which user data objects are moved	*NONE: Special value: Only check the library to see if any objects exist. Do not move them

Notes

Sets of these objects are normally created as part of the product installation process or by using the YCRTY1DTA command.

Example

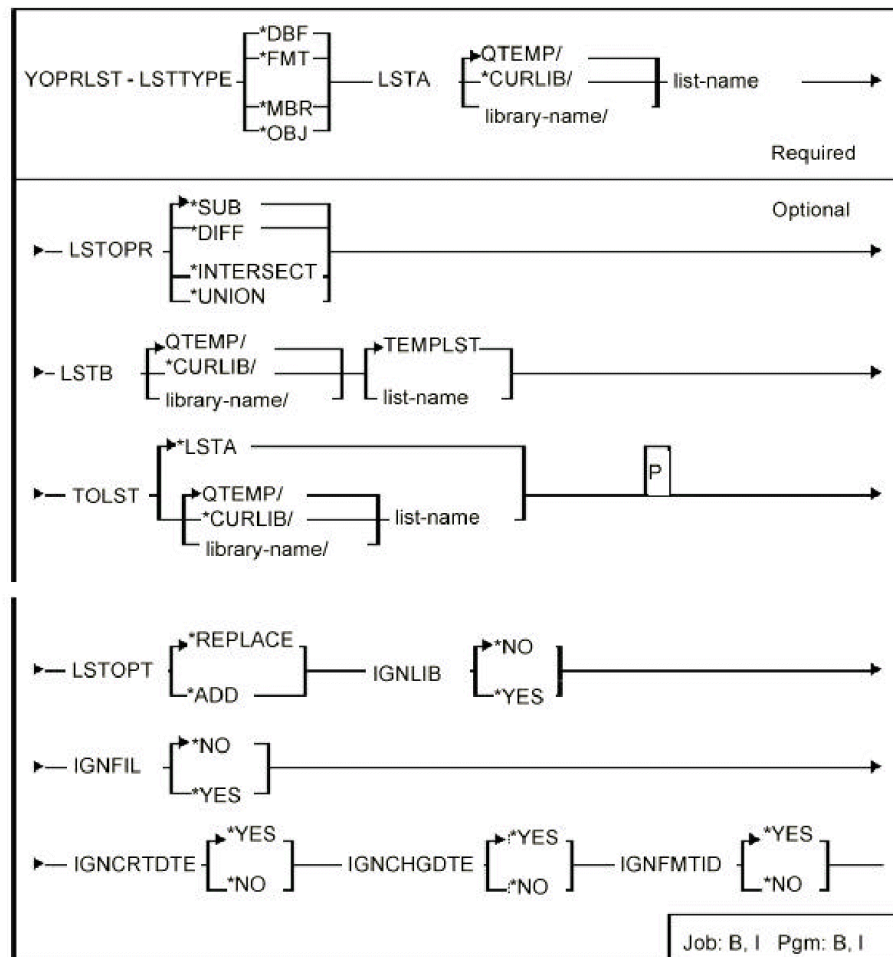
To move objects from library Y1SY to QGPL, enter:

```
YMOVY1DTA FROMLIB(Y1SY) TOLIB(QGPL)
```

YOPRLST (Operate on List)

This command performs a set operation on two input lists yielding a result list.

Syntax Diagram



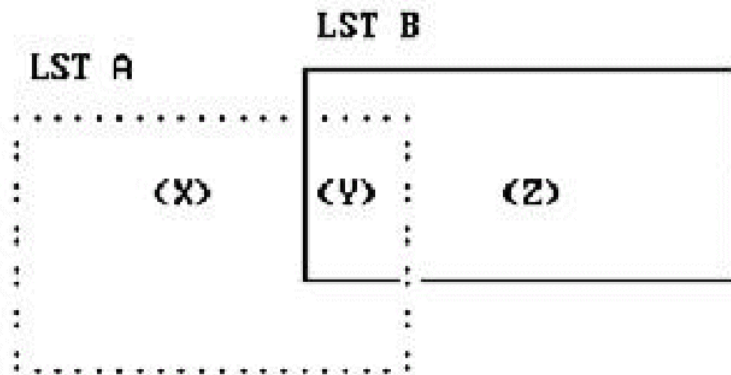
Parameters

Parameter	Definition	Value and Description
LSTTYPE	List type for operation	<ul style="list-style-type: none"> ■ *DBF: Database file list ■ *FMT: Format list. ■ *MBR: Member list. ■ *OBJ: Object list
LSTA	Qualified name of a list which is operand 1	
LSTOPR	List (set) operation	<ul style="list-style-type: none"> ■ *SUB: (default) Subtract contents of LSTB from LSTA ■ *DIFF: Calculate difference between two lists ■ *INTERSECT: Calculate intersection of two lists ■ *UNION: Calculate union of two lists
LSTB	Qualified name of a list which is operand 2	QTEMP/TEMPLST: (default) List name
TOLST	Qualified name of a list to contain the result	*LSTA: Place result of the operation in list named by LSTA parameter
LSTOPT	List replacement option	<ul style="list-style-type: none"> ■ *REPLACE: (default) Replace any existing contents of TOLST ■ *ADD: Add to current TOLST contents
IGNLIB	Ignore library names when comparing list items	<ul style="list-style-type: none"> ■ *NO: (default) Use library name when comparing list items ■ *YES: Ignore library name
IGNFIL	Ignore file names when comparing member list items	<ul style="list-style-type: none"> ■ *NO: (default) Use file name when comparing member list name ■ *YES: Ignore file name when comparing member list items

Parameter	Definition	Value and Description
IGNCRTDTE	Ignore creation dates when comparing list items (database file, member and object lists only)	<ul style="list-style-type: none">■ *YES: (default) Ignore creation dates■ *NO: Check that creation dates are the same when comparing list items
IGNCHGDTE	Ignore change dates when comparing list items (database, member and object list only)	<ul style="list-style-type: none">■ *YES: (default) Ignore change dates■ *NO: Check that change dates are the same when comparing list items
IGNFMTID	Ignore format ids when comparing list items (format lists only)	<ul style="list-style-type: none">■ *YES: (default) Ignore format id■ *NO: Check that format ids are the same when comparing list items

Notes

The following VENN diagram represents the contents of two intersecting lists:



The results of the list operations are:

Operand 1	LSTOPR	Operand 2	TOLST
LSTA	*SUB	LSTB	Area: X
LSTA	*DIFF	LSTB	Area: X and area Z
LSTA	*INTERSECT	LSTB	Area: Y
LSTA	*UNION	LSTB	Area: X and area Y and area Z

Note that list subtraction is not commutative; for example, (A - B) is the same as $-(B - A)$.

Examples

To build a list, C, of all the items in object list A that are not in object list B:

```
YOPRLST LSTTYPE(*OBJ) LSTA(QTEMP/A) LSTOPR(*SUB) LSTB(QTEMP/B) TOLST(QTEMP/C)
```

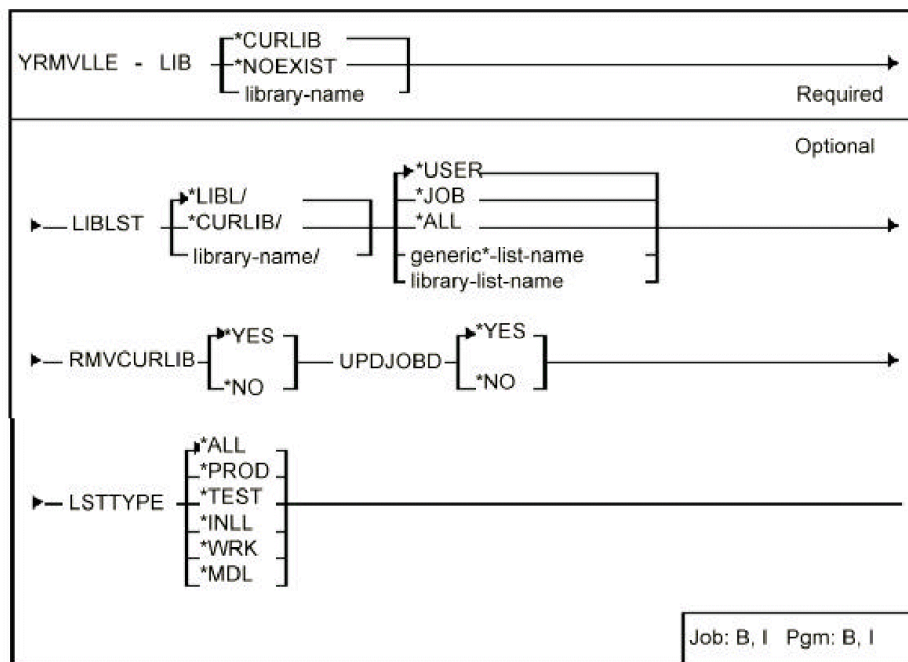
To build a list, C, of all items common to member lists A and B, ignoring the library name when comparing items:

```
YOPRLST LSTTYPE(*MBR) LSTA(QTEMP/A) LSTOPR(*INTERSECT) LSTB(QTEMP/B)
TOLST(QTEMP/C) IGNLIB(*YES)
```

YRMVLLE (Remove Library List Entry)

This command removes a library from a library list.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
LIB	Name of the removed library	<ul style="list-style-type: none"> ■ *CURLIB: Take the name of the library to be removed from the current library of the job running the YRMVLE command ■ *NOEXIST: For all library lists specified on LIBLST parameter, check the existence of each library in the user portion; and remove the entry if the library does not exist. If RMVCURLIB(*YES) is also specified, then the current library (CURLIB) of each library list is also checked and replaced with *NOCHG if a nonexistent library name was specified
LIBLST	Qualified generic name of library list(s) from which the library is removed	<ul style="list-style-type: none"> ■ *USER: (default) Library list name is same as that of the current user profile ■ *JOB: Library list name is same as that of the current job ■ *ALL: Remove library from all library lists in specified library
RMVCURLIB	Remove current library entry from library list	<ul style="list-style-type: none"> ■ *YES: (default) If the library specified on the LIB parameter matches the current library (CURLIB) entry on the library list, remove the library name and replace it with *NOCHG ■ *NO: Do not remove library list's current library entry

Parameter	Definition	Value and Description
UPDJOB	Update library list's job description	<ul style="list-style-type: none">■ *YES: (default) Updates the initial library list of the job description entry on the library list with the modified list of libraries specified on the list■ *NO: Do not update job description
LSTTYPE	Select library list type	*ALL: (default) Select all library lists

Notes

1. If the library does not exist a warning message is issued but the library is still removed from the library list(s).
2. If a library list contains only one library, a warning message is issued and the library is not removed from the list.
3. If you specify UPDJOB(*YES) for a library list without an associated job description, this parameter is ignored.

Examples

To remove library MORGAN from library list NY:

```
YRMVLE LIB(MORGAN) LIBLST(NY)
```

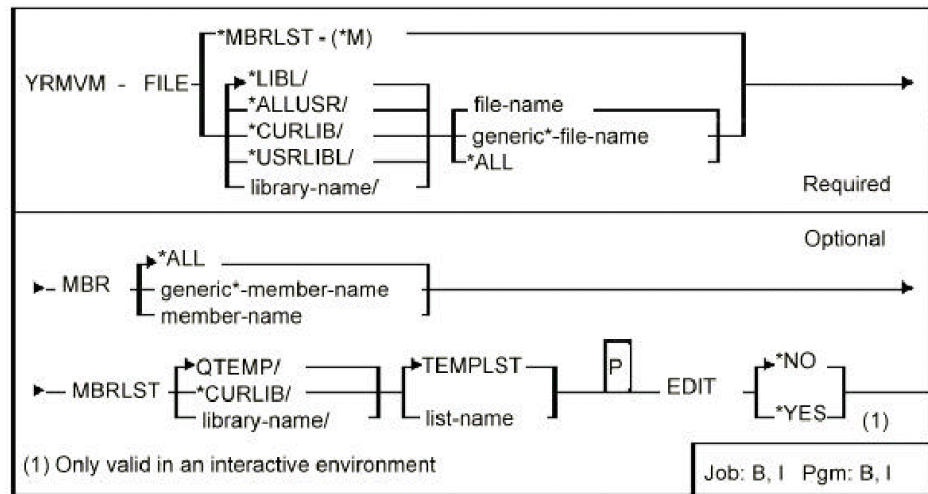
To remove all entries for nonexistent libraries from all library lists of type TEST:

```
YRMVLE LIB(*NOEXIST) LIBLST(*ALL) LSTTYPE(*TEST)
```

YRMVM (Remove Member)

This command removes members from a file. The members to be removed can be specified by a generic name or by a member list.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
FILE	Qualified generic name of file containing removed members	<ul style="list-style-type: none"> ■ *MBRLST: The names of removed members are specified by a member list whose name is given by the MBRLST parameter ■ *ALL: Members are removed from all files in specified libraries
MBR	Generic name of members which are removed	*ALL: (default) All members are removed
MBRLST	Qualified name of a member list specifying members that are to be removed	QTEMP/TEMPLST: (default) Specify FILE(*MBRLST) to use an existing list
EDIT	Edit list required	<ul style="list-style-type: none"> ■ *NO: (default) No editing required ■ *YES: The edit member list function is invoked to edit the list before execution

Notes

None

Example

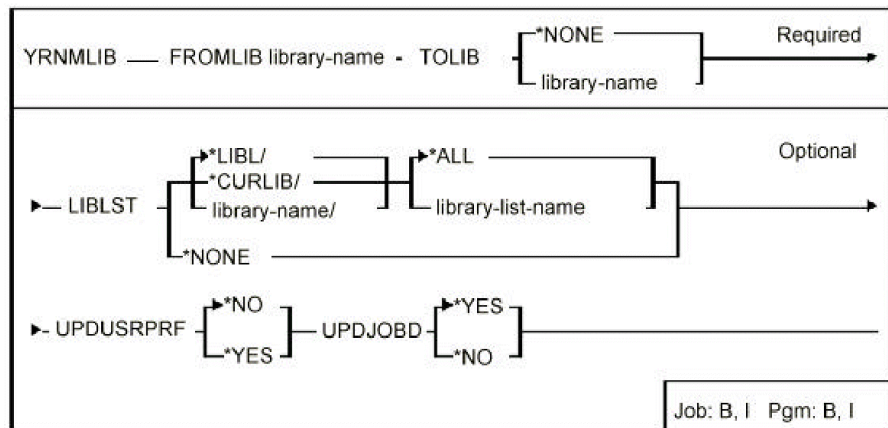
To remove all members in member list JRIPPER:

```
YRMVM FILE(*MBRLST) MBRLST(JRIPPER)
```

YRNMLIB (Rename Library)

This command renames a library and updates all references to the library in the library lists. Any references to the library made by a user profiles can also be updated.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
FROMLIB	Name of library which is to be renamed	
TOLIB	New name of library	*NONE: (default) Library is removed from all lists in which it occurs

Parameter	Definition	Value and Description
LIBLST	Qualified name of library list containing references to the library being renamed which should be updated	<ul style="list-style-type: none"> ■ *ALL: (default) All library lists are updated ■ *NONE: No library lists are updated
UPDUSRPRF	Update any references to the library made by user profiles	<ul style="list-style-type: none"> ■ *NO: (default) Do not update user profiles ■ *YES: Do update user profiles
UPDJOB	Update the job description associated with the library list	<ul style="list-style-type: none"> ■ *YES: (default) Update the initial library list of the job description with the libraries specified in the list. ■ *NO: Do not update job description

Notes

1. You must have object management rights to library.
2. If the library is a CA 2E model library, then the YMDLVNM model value is updated.
3. You must have security administrator special authority (*SECADM) on your own user profile and *USE authority to the i OS command CHGUSRPRF to update references made by user profiles to the library.
4. Update Job Description (UPDJOB) considerations are as follows:
UPDJOB(*YES) is invalid if LIBLST is *NONE.

If you specified UPDJOB(*YES) for a library list without an associated job description, this parameter is ignored.

If you specified UPDJOB(*YES), the library name of the JOBDs associated with the library lists are renamed, if matching the from library

5. If you specified UPDUSRPRF(*YES) any references to the library by individual user profiles will be updated. These may include:

Used by	Library
OS/400	Initial program library
OS/400	Job description library
OS/400	Output queue library
OS/400	Message queue library
OS/400	Current library, CURLIB
YINLPGM	Initial library list library
YINLPGM	Initial menu file library
YINLPGM	Attention key program library
YINLPGM	YGO Submitted job description library
YINLPGM	YGO Exception message queue library

Example

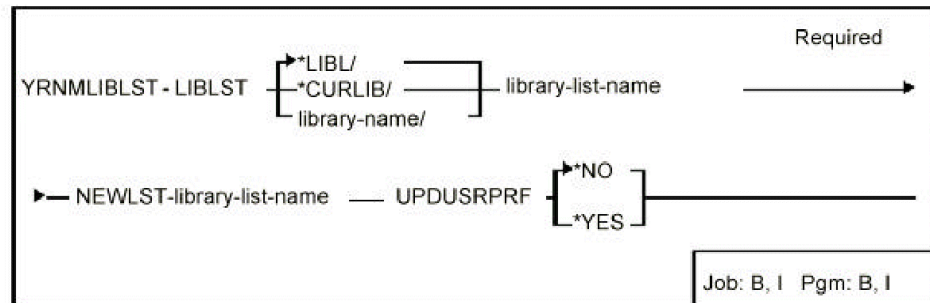
To rename library BOOTS to MUDIES, updating all references in library lists and user profiles and synchronizing initial library list of all referenced JOBDs:

```
YRNMLIB FROMLIB(BOOTS) TOLIB(MUDIES) UPDUSRPRF(*YES) UPDJOB(*YES)
```

YRNMLIBLST (Rename Library List)

This command renames a library list.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
LIBLST	Qualified name of library list to be renamed	
NEWLST	Name of new library list	
UPDUSRPRF	Update any reference to the library list in user profiles	<ul style="list-style-type: none"> ■ *NO: (default) Do not update user profiles ■ *YES: Update user profiles

Notes

None

Example

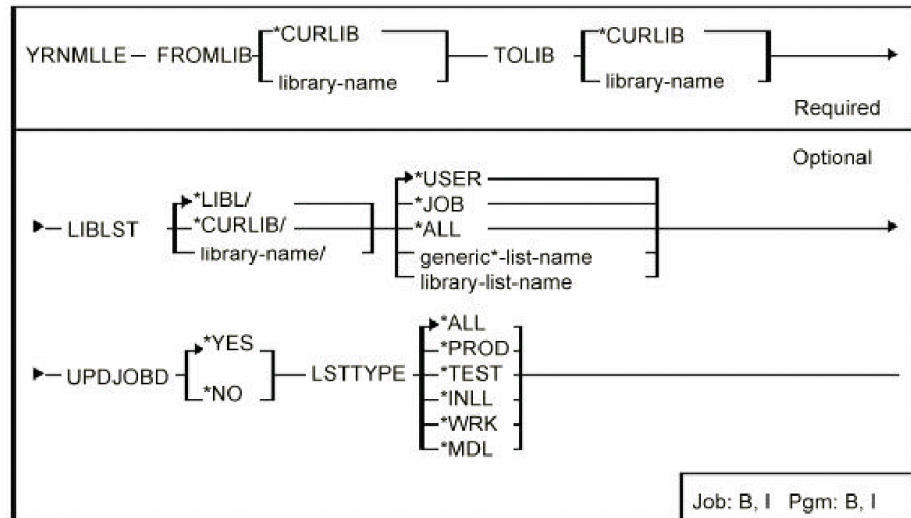
To rename library list NY to library list LA:

```
YRNMLIBLST LIBLST(NY) NEWLST(LA)
```

YRNMLLE (Rename Library List Entry)

This command renames a library list entry in a list(s).

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
FROMLIB	Name of library list entry to be renamed	*CURLIB: (default) Rename job's current library
TOLIB	New name of library	*CURLIB:(default) Rename library to job's current library
LIBLST	Qualified generic name of library list(s) in which the library list entry is renamed	<ul style="list-style-type: none"> ■ *USER: (default) Library list name is same as that of the current user profile ■ *JOB: Library list name is same as that of the current job ■ *ALL: Rename library in all lists in specified library

Parameter	Definition	Value and Description
UPDJOB	Update library list's job description	<ul style="list-style-type: none"> ■ *YES: (default) Update the initial library list of the job description associated with the library list with the libraries specified in the list ■ *NO: Do not update job description
LSTTYPE	Select library list type	*ALL: (default) Select all library lists

Notes

- The command may be used to rename a library list entry in a library list or lists. Note the following points:
 - If the TOLIB library does not exist, a warning message will be issued, but the library list entry will still be renamed in each list.
 - If the TOLIB library is already present as an entry in a list, the first duplicate reference will be removed.
- A completion message is returned indicating the number of lists in which the library list entry has been renamed.
- Update Job Description (UPDJOB) considerations: UPDJOB(*YES) is invalid if LIBLST is *NONE.
 - If you specified UPDJOB(*YES) for a library list without an associated job description, this parameter is ignored.
 - If you specified UPDJOB(*YES), the library name of the JOBDs associated with the library lists are renamed, if matching the from library. See [Appendix A](#) for special types for this parameter.

Examples

To rename library list entry MORGAN in library list NY to TRUMP:

```
YRNMLE FROMLIB(MORGAN) TOLIB(TRUMP) LIBLST(NY)
```

To rename library list entry CAPTAIN to TRUMP in library list LA and update any associated job description:

```
YRNMLE FROMLIB(CAPTAIN) TOLIB(TRUMP) LIBLST(LA) UPDJOB(*YES)
```


Notes

User must have update rights to menu file.

Example

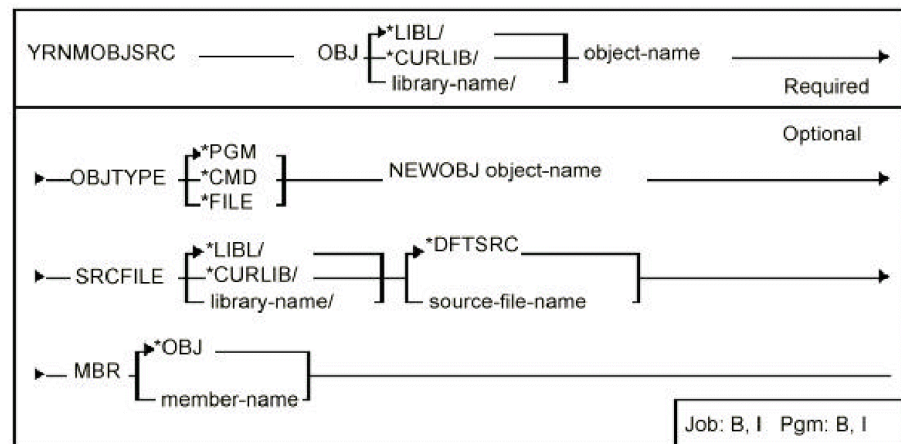
To rename menu ALACARTE as PLATDUJOUR:

```
YRNMNU MENU(ALACARTE) TOMENU(PLATDUJOUR)
```

YRNMOBJSRC (Rename an Object and Source)

This command renames an object and its corresponding source member.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
OBJ	Qualified name of the renamed object	
OBJTYPE	Type of renamed object	<ul style="list-style-type: none"> ■ *PGM: (default) Program ■ *CMD: Command ■ *FILE: File

Parameter	Definition	Value and Description
NEWOBJ	New name of object and source member	
SRCFILE	Qualified name of file containing source member	<ul style="list-style-type: none">■ *DFTSRC: (default) Use source file appropriate to source type:■ If OBJTYPE(*CMD) is specified, use QCMSRC■ If OBJTYPE(FILE) is specified, use QDDSSRC■ If OBJTYPE(*PGM) is specified, first try QCLSRC, then QRPGRSRC
MBR	Name of existing source member	*OBJ: Same as object name

Notes

1. This command is equivalent to the i OS commands Rename Object (RNMOBJ) and Rename Member (RNMM).
2. Renaming will only be done if both object and source can be renamed without errors. Renaming requires the exclusive allocation of the source member.
3. Note that PL/I programs cannot be renamed.

Example

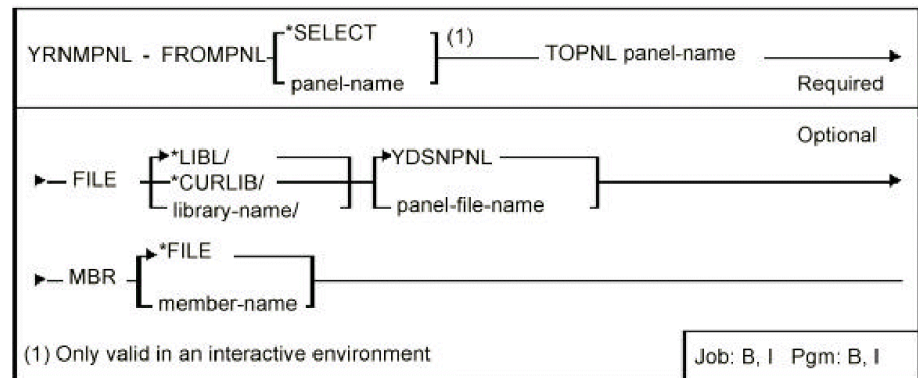
To rename HLL program TREAD and its source to RETREAD:

```
YRNMOBJSRC OBJ(TREAD) OBJTYPE(*PGM) NEWOBJ(RETREAD)
```

YRNMPNL (Rename Panel Design)

This command renames a panel design.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
FROMPNL	Name of a panel design which is renamed	*SELECT: (default) The panel design selection display is presented
TOPNL	New name for panel design	
FILE	Qualified name of file containing panel design	YDSNPNL: (default) Panel design file name
MBR	Name of member in file containing panel design	*FILE: (default) The member has the same name as the file

Notes

The command does not check for references to the renamed panel design from any other panel design.

Example

To rename a panel design called JOHN in file FRED/YDSNPNL to JAMES you would enter the following:

```
YRNMPNL FROMPNL(JOHN) TOPNL(JAMES) FILE(FRED/YDSNPNL)
```


Example

To rename report design KINSEY to FRQNCY:

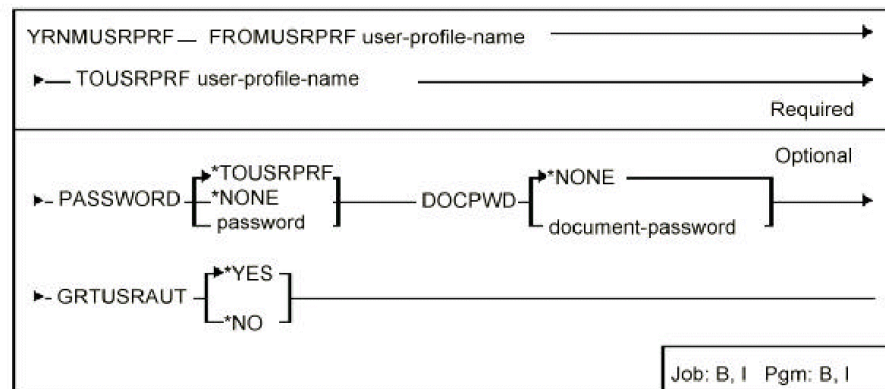
```
YRNMURPT FROMRPT(KINSEY) TORPT(FRQNCY)
```

```
YRNMUSRPRF
```

YRNMUSRPRF (Rename User Profile)

This command renames a user profile. You should ensure that the profile is not enrolled in Office/400 before you execute this command. The command does not update references to the user profile on existing authorization lists. Any such references are lost.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
FROMUSRPRF	Name of user profile which is to be renamed	
TOUSRPRF	New name of user profile	
PASSWORD	User password. The password must be a valid name	<ul style="list-style-type: none"> ■ *TOUSRPRF: (default) Adopt the new profile name as the password ■ *NONE: New profile has no password

Parameter	Definition	Value and Description
DOCPWD	Specifies a document password for DIA interchange	*NONE: (default) No document password is used
GRTUSRAUT	Specifies whether to grant the authorities of the original profile to the renamed profile	<ul style="list-style-type: none">■ *YES: (default) Grant the authorities of the original profile to the new profile■ *NO: Do not grant the authorities

Notes

This command renames a profile by creating a new profile, transferring all ownerships and authorities and then deleting the old one. It does this in the following stages:

- Creates the TOUSRPRF using the command Copy User Profile YCPYUSRPRF.
- Changes the ownership of all objects owned by the FROMUSRPRF to the TOUSRPRF.
- Changes the group profile of any sub-profiles belonging to the FROMUSRPRF to the TOUSRPRF.
- If no errors have occurred, deletes the FROMUSRPRF.
- If errors occur, you will need to delete the FROMUSRPRF manually.

Example

To rename library user profile ROMAN to GREEK, transferring all authorities:

```
YRNMUSRPRF FROMUSRPRF(ROMAN) TOUSRPRF(GREEK)
```

YRSTSPLRTR (Restart Spooled File Router)

The Restart Spooled File Router (YRSTSPLRTR) command is used to reset a spooled file router job which is monitoring the specified data queue.

A spooled file router should be reset when the contents of the YSPLRTRP file have changed (due to spooled file routing entries having been added, changed or deleted). The reset process will update the running spooled file router job with the changed YSPLRTRP data, without ending the job.

See the command help for the Start spooled file router (YSTRSPLRTR) command for more information about the spooled file router job.

Syntax Diagram

Parameters

Parameter	Definition	Value and Description

Data queue (DTAQ)

Specifies the data queue currently being monitored.

Note: You must specify a data queue that is currently being monitored by a spooled file router job. Specifying a data queue that is not being monitored by a spooled file router job can result in serious errors within any programs using that data queue.

YSPLRTRQ

Use the default spooled router data queue YSPLRTRQ.

data-queue-name

Enter the name of an existing data queue which is being monitored by a spooled file router job.

Possible library values are:

*LIBL

All libraries in the user and system portions of the job's library list are searched until the first match is found for the specified data queue.

*CURLIB

The current library for the job is used to locate the specified data queue. If no library is specified as the current library for the job, the QGPL library is used.

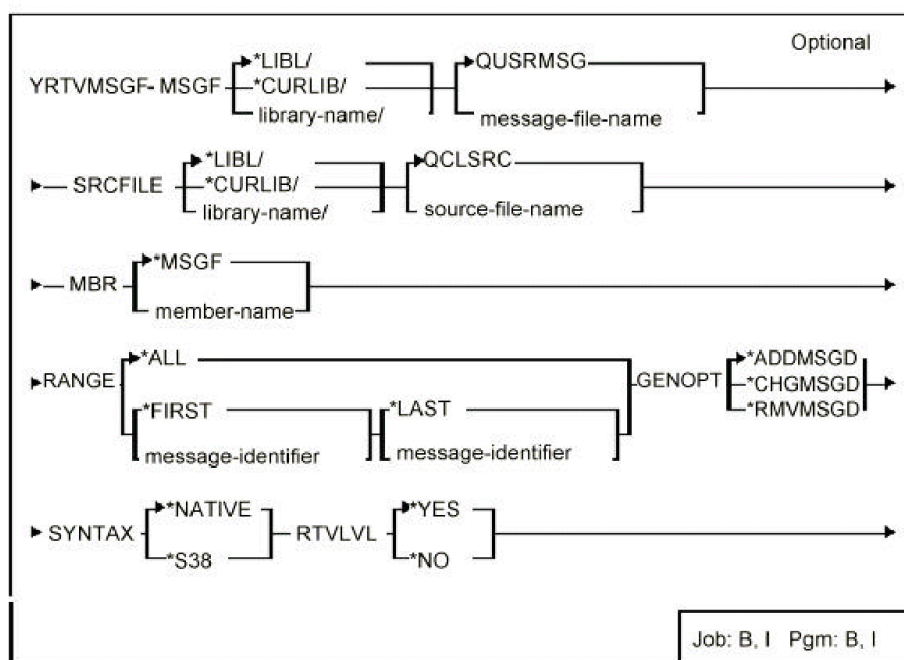
library-name

Specify the name of the library to be searched for the specified data queue.

YRTVMSGF (Retrieve Message File)

This command retrieves the message descriptions or some of the message descriptions from a message file into a CL source member. The CL source member contains the CL instructions necessary to add, change, or remove messages (or a range of messages) from the message file specified on a subsequent call to the compiled program.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
MSGF	Qualified name of message files whose message descriptions are to be retrieved	QUSRMSG: (default) Message file name

Parameter	Definition	Value and Description
SRCFILE	Qualified name of source file into which the CL source member is to be placed	QCLSRC: (default) Source file name
MBR	Member name of CL member containing message descriptions	*MSGF: (default) Name is same as MSGF
RANGE	Specifies a range of message identifiers within the specified message file for which statements are to be processed	<ul style="list-style-type: none"> ■ *ALL: (default) Generate CL for all the messages <p>Otherwise, RANGE is a list parameter made up of the following two elements:</p> <ul style="list-style-type: none"> ■ Starting message identifier: All message definitions with an identifier greater than or equal to this value will be retrieved. <ul style="list-style-type: none"> – *FIRST: Start at the first message description in the file. ■ Ending message identifier: All messages with an identifier less than or equal to this value are retrieved. <ul style="list-style-type: none"> – *LAST: Continue until the last message description in the file
GENOPT	Generation option: specifies whether the CL generation should be to add, change or remove the message descriptions	<ul style="list-style-type: none"> ■ *ADDMSGD: (default) Generate CL statements to add the message descriptions to the named message file ■ *CHGMSGD: Generate CL statements to change the message descriptions in the named message file ■ *RMVMSGD: Generate CL statements to remove the message descriptions from the named message file

Parameter	Definition	Value and Description
SYNTAX	Specifies the syntax of the CL program generated	<ul style="list-style-type: none">■ *NATIVE: (default) Generate i OS Native CL.■ *S38: Generate S38 CL
RTVLVL	Specifies whether to retrieve and include Creating level information (applicable to ADDMSGD only)	<ul style="list-style-type: none">■ *YES: (default) Retrieve level■ *NO: Do not retrieve level

Notes

1. The command outputs a CL source member which can be compiled and used to build a message file. This utility can be of use:
 - When you need to translate messages into other National languages, it is generally easier to carry out the text editing of messages using SEU rather than the command entry display. This is especially true for IGC versions of i OS, when ideographic message text must be entered using SEU.
 - For comparing messages in Message files.
 - When merging or reconciling messages files; for instance, when development has been taking place on two different machines. For example:
 - To copy a range of messages from one message file to another, you could use the YRTVMSGF command with a value of *ADDMSGD for the GENOPT parameter, and then run the resulting CL program to add the message definitions to the target message file.
 - To move a range of messages from one message file to another, you could use the YRTVMSGF command to create two separate programs; once with a value of *ADDMSGD for the GENOPT parameter, and once with a value of *RMVMSGD. Running the first CL program will add the message definitions to the target message file, running the second one will remove them from the originating file.

2. The CL source generated includes the following features:
 - A banner showing the date generated.
 - An entry parameter is declared; this is for the name of the message file whose descriptions are to be modified.
 - For each message identifier retrieved there is the necessary CL statement to modify the message description as specified (add, change or remove).
 - The length of the strings involved is shown as a comment which appears after each description. This may be useful information for translators when editing message text for Display file DDS prompts that is retrieved through the DDS MSGCON or MSGID keywords.
 - General error handling code is included.

CL source generated appears on the following page.

```

PGM      PARM(&ADDMSGF)
/*T:      Message descriptions from QGPL/DEMOMSG          */
/*Z:      CRTCLPGM LOG(*NO)                               */
/*H:      System      : Widget Processing System          */
/*H:      Programmer   : A.N.Other                        */
/*H:      Date        : 07/07/88                          */
/*H:      Job         : RTVPMT                            */
/*H:      Synopsis     :                                  */
/*H:                                   Creates a Message file from ADDMSGD commands. */
/*H:                                   1. Check for existence of Message file.      */
/*H:                                   Create Message file if it does not exist.    */
/*H:                                   2. Build messages from message descriptions.  */
/*H:                                   Monitor for message already existing in fl  */
/*H:                                   3. Send Completion message.                  */
/*M:      Maintenance  : *NONE                                                                    */

/* Passed parameters */
DCL &ADDMSGF *CHAR 20 /* Add Message file/library name. */
DCL &TOMSGF *CHAR 10 /* Add Message file name. */
DCL &TOLIB *CHAR 10 /* Add Message file library name. */

/* Work fields */
DCL &USER *CHAR 10 /* User ID */
DCL &TEXT *CHAR 50 /* Message file text */
DCL &MSGID *CHAR 7 /* Message ID */
DCL &MSGDTA *CHAR 132 /* Message data */
DCL &MSGF *CHAR 10 /* Message file */
DCL &MSGFLIB *CHAR 10 /* Message file library */

/* ===== */
/*H: 0.1 Global monitor for message description already exists. */
MONMSG MSGID(CPF2412) /* Do nothing. */
/*H: 0.2 Global monitor for escape messages. */
MONMSG MSGID(CPF0000) EXEC(GOTO CMDLBL(ERROR))
/* ===== */
/*H: 1. Process parameters. */
/* ----- */
CHGVAR VAR(&TOMSGF) VALUE(%SST(ADDMSGF 1 10))
CHGVAR VAR(&TOLIB) VALUE(%SST(ADDMSGF 11 10))
/* ----- */
/*H: 2. Check required objects. */
/* ----- */
/*H: 2.1 Check Message file exists. */
CHKOBJ OBJ(&TOLIB/&TOMSGF) OBJTYPE(*MSGF)
MONMSG MSGID(CPF9801) EXEC(DO)
RTVJOBA USER(&USER)
CHGVAR &TEXT VALUE('Generated by YRTVMSGF - User: ' +
|| &USER |<' ')
CRTMSGF MSGF(&TOMSGF.&TOLIB) TEXT(&TEXT)
ENDDO
/* ----- */
/*H: 3. Add Message descriptions for specified Message file. */
/* ----- */

```

```

ADDMSGD      MSGID(USE0001) MSGF(&TOLIB/&TOMSGF) +
              MSG('Good Morning &1') SEV(0) +
              DMPLST(*JOB) LVL('20/05/88' 01)
/* Length of 1st level text: 15 */

ADDMSGD      MSGID(USR0002) MSGF(&TOLIB/&TOMSGF) +
              MSG('Invalid response') SEV(0) DMPLST(*JOB) +
              LVL('20/05/88' 01)
/* Length of 1st level text: 16 */

etc etc ...

/*-----*/
/*H: 98. Send Completion message. */
      CHGVAR      VAR(&MSGDTA) VALUE(&TOMSGF || &TOLIB)
      SNDPGMMSG   MSGID(YMS0001) MSGF(YYYYMSG) +
      MSGDTA(&MSGDTA) MSGTYPE(*COMP)

ENDPGM: RETURN
/*=====*/
/*H: 99. Error Message processing. */
ERROR: RCVMSG     MSGTYPE(*EXCP) MSGDTA(&MSGDTA) MSGID(&MSGID) +
      MSGF(&MSGF) MSGFLIB(&MSGFLIB)
      MONMSG      MSGID(CPF0000)
      SNDPGMMSG   MSGID(&MSGID) MSGF(&MSGFLIB/&MSGF) +
      MSGDTA(&MSGDTA) MSGTYPE(*ESCAPE)
      MONMSG      MSGID(CPF0000)
      GOTO ENDPGM

ENDPGM

```

Example

To retrieve all message descriptions from message file MESSAGE into a member MESSAGE in file QCLSRC in library QGPL:

```
YRTVMSGF MSGF(MESSAGE) SRCFILE(QGPL/QCLSRC)
```

YRTVOBJLIB (Retrieve Object Library)

This command checks current job's library list for a given object and, if it is found, returns the name of the first library containing the object.

Parameter	Definition	Value and Description
OBJTYPE	Object type of object whose library is to be retrieved	Must be one of the valid i OS object types as allowed for the i OS command Check Object (CHKOBJ)
MBR	File member's name. If a member is specified, the YRTVOBJLIB command checks for a file containing the specified member	<ul style="list-style-type: none"> ■ *NONE: (default) Do not check for a member. ■ *FIRST: Check the existence of at least one member

Notes

If the object cannot be found either message CPF9801 (Object not found), or CPF9815 (Member not found) is sent.

Example

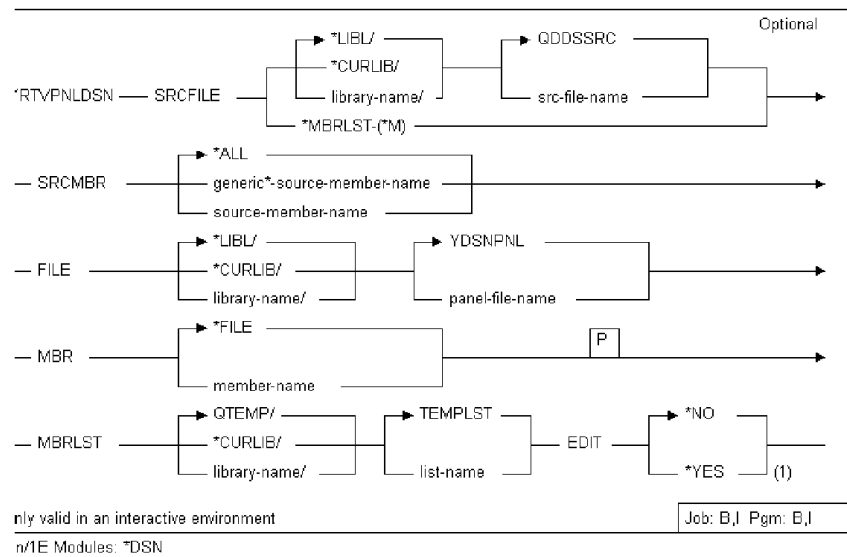
To specify that the user part of the invoking job's library list is to be searched for a file with the name FRED and that, if found, the name of the library containing the file is to be returned into variable &OBJLIB.

```
DCL &OBJLIB *CHAR 10 /* Library*/
CHGVAR VAR(&OBJLIB) VALUE(*USRLIBL)
YRTVOBJLIB OBJ(FRED) OBJLIB(&OBJLIB) OBJTYPE(*FILE)
```

YRTVPNLDSN (Retrieve Panel Design)

This command retrieves a panel design from the DDS source for an external display file or files. The command Define Panel Design (YDFPNLDSN) can be used in conjunction with this command to define how individual formats are combined into a panel design.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
SRCFILE	Qualified name of source file containing DDS source members for externally described display files which are to be retrieved into panel designs	<ul style="list-style-type: none"> ■ QDDSSRC: (default) DDS source file name ■ *MBRLST: locate source members to be compiled by the member list specified by the MBRLST parameter
SRCMBR	Generic name of source members to be retrieved	*ALL: All members of type DSPF in the specified source file will be retrieved
FILE	Qualified name of file to contain panel designs	YDSNPNL: (default) Panel file name
MBR	Name of member in file to contain panel designs	*FILE: (default) The member has the same name as the file

Parameter	Definition	Value and Description
MBRLST	Qualified name of a member list	QTEMP/TEMPLST: (default) List name
EDIT	Edit list option	<ul style="list-style-type: none"> ■ *NO: List edit function is not invoked ■ *YES: invoke the list edit function before proceeding with panel design retrieval

Notes

1. Execution of this command involves a dummy recompilation of the selected print files (that is, CRTDSPF with GENOPT(*NOGEN)). You should ensure that any field reference files necessary for the successful compilation of the print file are present in the library list of the job executing the YRTVNLDSN command. If the recompilation fails, the panel design will not be retrieved.
2. You may control which formats from the DDS are combined to make up a Panel Design by placing Define Panel Design (YDFNPNLDSN) statements as comments at the beginning of the DDS source member which is to be retrieved.

CA 2E builds one panel design for each command statement YDFNPNLDSN that it finds in a source member, using the formats named on the command statement. If the formats named by a single YDFNPNLDSN statement overlap, then they will be overlaid on top of each other in the order in which they appear in the DDS source. For example, each successive format will be overlaid on top of the previous one.

If no instance of the YDFNPNLDSN command is included in a DDS source member that is being retrieved, then CA 2E assumes the default values for the YDFNPNLDSN command. For example, all the display file formats from the source member are combined into a single panel design. If the panel formats overlap, they will be overlaid in order of specification.

See the command diagram Define Panel Design (YDFNPNLDSN) for further details.

Example

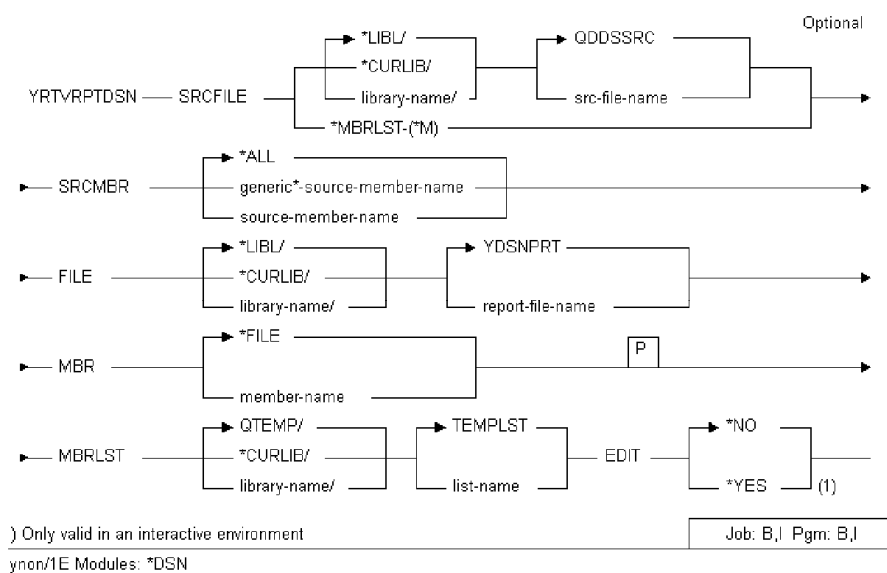
To retrieve all DDS members with names beginning GL into panel design file YPNLDSN in library ADLIB, member ADHOC:

```
YRTVNLDSN SRCFILE(ADLIB/QDDSSRC) SRCMBR(GL*) FILE(ADLIB/YPNLDSN) MBR(ADHOC)
```

YRTVRPTDSN (Retrieve Report Design)

This command retrieves a report design from the DDS source for an external print file or files.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
SRCFILE	Qualified name of source file containing DDS source for external print file which is to be retrieved to form report designs	<ul style="list-style-type: none"> ■ QDDSSRC: (default) DDS source file name ■ *MBRLST: Locate source members to be compiled by the member list specified by the MBRLST parameter
SRCMBR	Generic name of source members to be retrieved	*ALL: (default) All members in the specified source file are retrieved
FILE	Qualified name of file to contain report designs	YDSNPRT: (default) Report file name

Parameter	Definition	Value and Description
MBR	Name of member in file to contain report designs	*FILE: (default) The member has the same name as the file
MBRLST	Qualified name of a member list	QTEMP/TEMPLST: (default) List name
EDIT	Edit list option	<ul style="list-style-type: none"> ■ *NO: (default) List edit function will not be invoked ■ *YES: Invoke the list edit function before proceeding with report design retrieval

Notes

1. Execution of this command involves a dummy (that is, CRTPRTF with GENOPT(*NOGEN)) recompilation of the selected print files. You should ensure that any field reference files necessary for the successful compilation of the print file are present in the library list of the job executing the YRTVRPTDSN command. If the recompilation fails, the report design will not be retrieved.
2. Formats are included in the report design in the order in which they appear in the DDS source.

Example

To retrieve all DDS members with names beginning with the letters GL into report design file YRPTDSN in library ADLIB, member ADHOC:

```
YRTVRPTDSN SRCFILE(ADLIB/QDDSSRC) SRCMBR(GL*) FILE(ADLIB/YRPTDSN) MBR(ADHOC)
```

YRTVSPLFA (Retrieve Spooled File Attributes)

The Retrieve Spooled File Attributes (YRTVSPLFA) command is used in a CL program to retrieve the values of one or more spooled file attributes for a specified spooled file and place the values into the specified CL variable.

The CL prompt for this command lists the minimum length for retrieved variables next to the appropriate parameters. For character variables, a single number is shown. For decimal variables, two numbers are shown. The first number indicates the minimum variable length and the second number indicates the minimum number of decimal positions.

Parameters

Parameter	Definition	Value and Description
FILE	This is a required parameter. Specifies the name of the spooled file whose attributes are being retrieved.	<ul style="list-style-type: none">■ Job name (JOB): Specifies the name of the job that created the spooled file.■ *: The job that created the spooled file issued this command.■ job-name: Specify the name of the job that contains the spooled file.■ user-name: Specify the user name that identifies the user profile under which the job is run.■ job-number: Specify the system-assigned job number.
SPLNBR	Specifies the unique number of the spooled file in the job whose attributes are being retrieved.	<ul style="list-style-type: none">■ *ONLY: Only one spooled file in the job has the specified file name; therefore, the number of the spooled file is not necessary.■ *LAST: If there is more than one spooled file with the specified file name the one with the highest number is the file whose attributes are retrieved.■ spooled-file-number: Specify the number of the spooled file that matches the file name whose attributes you wish to retrieve.

Notes

This command is valid only within a CL program.

CL var for RTNJOBNAME (10) (RTNJOBNAME)

Specifies the name of a variable used to return the name of the job that created the spooled file. In control language (CL) programs, this should be a character variable of length (10). This parameter can be used with RTNJOBUSER and RTNJOBNBR to return the full job name when the special value * was passed in the JOB parameter.

CL var for RTNJOBUSER (10) (RTNJOBUSER)

Specifies the name of a variable used to return the user of the job that created the spooled file. In control language (CL) programs, this should be a character variable of length (10). This parameter can be used with RTNJOBNAME and RTNJOBNBR to return the full job name when the special value * was passed in the JOB parameter.

CL var for RTNJOBNBR (6) (RTNJOBNBR)

Specifies the name of a variable used to return the number of the job that created the spooled file. In control language (CL) programs, this should be a character variable of length (6). This parameter can be used with RTNJOBNAME and RTNJOBUSER to return the full job name when the special value * was passed in the JOB parameter.

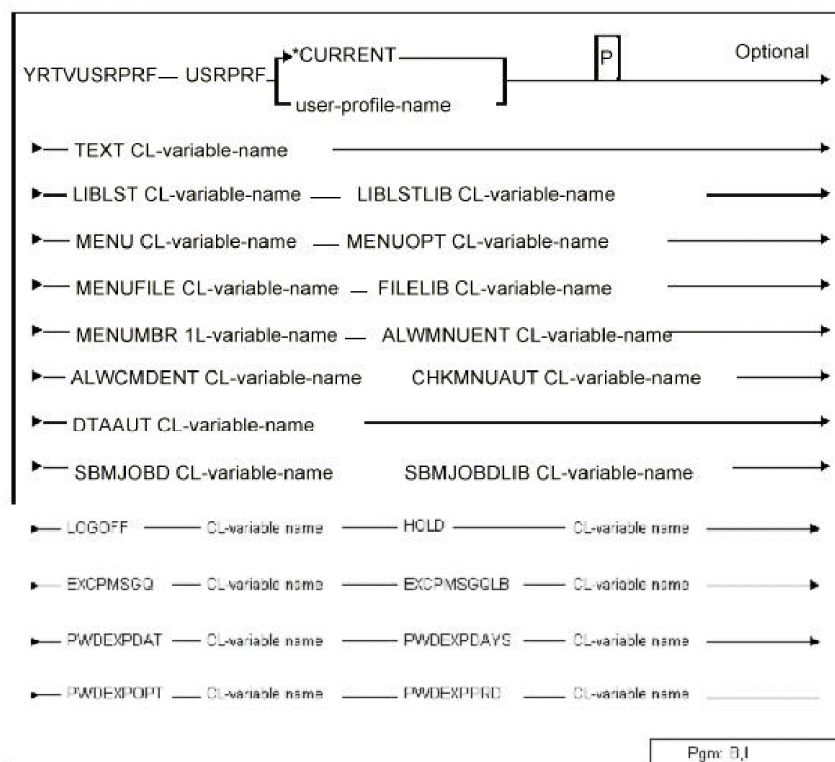
CL var for RTNSPLNBR (6 0) (RTNSPLNBR)

Specifies the name of a variable used to return the number of the spooled file. In control language (CL) programs, this should be a decimal variable of length (6 0). This parameter can be used to return the actual spooled file number when either of the special values *ONLY or *LAST were passed in the SPLNBR parameter.

YRTVUSRPF (Retrieve User Profile)

This command retrieves user profile attributes for a specified user profile.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
USRPRF	Name of user profile whose details are retrieved	*CURRENT: (default) Retrieve details for the current job's user profile
TEXT	CL variable that receives the name of the text for the user profile	Character variable fifty bytes long
LIBLST	CL variable that receives the name of the initial library list for the profile	Character variable ten bytes long

Parameter	Definition	Value and Description
LIBLSTLIB	CL variable that receives the name of the library containing the library list for the profile	Character variable ten bytes long
MENU	CL variable that receives the name of the initial menu for the profile	Character variable ten bytes long
MENUOPT	CL variable that receives the name of the initial menu option for the profile	Character variable two bytes long
MENUFILE	CL variable that receives the name of the menu file containing the initial menu for the profile	Character variable ten bytes long
FILELIB	CL variable that receives the name of the library containing the initial menu for the profile	Character variable ten bytes long
MENUMBR	CL variable that receives the name of the member containing the initial menu for the profile	Character variable ten bytes long
ALWMNUENT	CL variable that receives the current menu entry value for the profile	Character variable four bytes long <ul style="list-style-type: none"> ■ *NO: The user is not permitted to enter menu names from his menu, but is restricted to the displayed menu options ■ *YES: The user may enter menu names to transfer directly to a named menu
ALWCMDENT	CL variable that receives the current command entry value for the profile	Character variable four bytes long <ul style="list-style-type: none"> ■ *NO: (default) The user is not permitted to enter commands. The user is restricted to the displayed menu options ■ *YES: The user may enter and execute commands directly from a menu

Parameter	Definition	Value and Description
CHKMNUAUT	CL variable that receives the current menu option check value	Character variable seven bytes long
DTAAUT	CL variable that receives the current data authority level value for the profile	Character variable one byte long 1 to 9: (1-high, 9-low)
SBMJOBBD	CL variable that receives the current job description name to be used by the program Go to Menu (YGO) when submitting jobs for the profile	Character variable ten bytes long
SBMJOBDLIB	CL variable that receives the current name of the job description used by the program Go to Menu program (YGO) when submitting jobs for the profile	Character variable ten bytes long
LOGOFF	CL variable that receives the current LOGOFF option value for the profile	Character variable seven bytes long. <ul style="list-style-type: none">■ *NOLIST: (default) No job log is to be created■ *LIST: A job log is to be created
HOLD	CL variable that receives the current status of the user profile	Character variable four bytes long. <ul style="list-style-type: none">■ *YES: (default) The user is prevented from signing on■ *NO: The user may sign on
EXCPMSGQ	CL variable that receives the name of the exception message queue to be used by the program Go to Menu (YGO) when errors occur	Character variable ten bytes long
EXCPMSGQLB	CL variable that receives the name of the library containing the exception message queue	Character variable ten bytes long

Parameter	Definition	Value and Description
PWDEXPDAT	CL variable that receives the value of the password expiry date	Character variable six bytes long
PWDEXPDAYS	CL variable that receives the number of days between password changes	Decimal variable three digits packed
PWDEXPOPT	CL variable that receives the value of the password expiry option	Character variable ten bytes long
PWDEXPDAYS	CL variable that is to receive the number of days after a password has expired which the user may change it	Decimal variable three digits packed

Notes

None

Example

To retrieve the menu and library list user profile attributes from within a CL program:

```

_/* User profile name. */
_/*Initial library list. */
_/*Initial library list library. */
_/*Initial menu name. */
_/*Initial menu option. */
DCL &FILE *CHAR 10 /*Menu file. */
_/*Menu file library. */
DCL &MBR *CHAR 10 /*Menu member. */
_/*Allow menu entry (*YES/*NO). */
RTVJOB USER(&USRPRF)
YRTVUSRPRF USRPRF(&USRPRF) LIBLST(&LIBLST)
LIBLSTLIB(&LIBLSTLIB) MENU(&MENU) MENUOPT(&MENUOPT)
FILE(&FILE) FILELIB(&FILELIB) MBR(&MBR)
ALWMNUENT(&ALWMNUENT) ALWCMDENT(&ALWCMDENT)
CHKMNUAUT(&CHKMNUAUT)

```

YRUNSQL (Start Interactive SQL Session)

This command is used to execute an ad hoc SQL statement from a command line or CL program, irrespective of whether or not interactive SQL is installed on the IBM i. In addition, if interactive SQL is installed on the IBM i, you can start an interactive SQL session.

The user can override an unqualified file used in the SQL statement or the interactive SQL session to a specific member in the file in a specific library.

Parameters

Parameter	Definition	Value and Description
STMT	Specifies the SQL statement to be processed. This can be any SQL statement that is valid in an interactive SQL session, such as a SELECT or an UPDATE statement	<ul style="list-style-type: none">■ *STRSQL: Starts an interactive SQL session, using the default values specified for the Start SQL Interactive Session (STRSQL) command.■ sql-statement: Specifies a valid SQL statement to be processed.
IGNCAS	Specifies whether the statement should be treated as case-insensitive.	<ul style="list-style-type: none">■ *NO: Considers case-sensitivity.■ *YES Ignores case-sensitivity.

Parameter	Definition	Value and Description
FILE	Specifies the file to be overridden. This should be a file used in the SQL statement (or used in a statement to be supplied in the interactive SQL session.) any references to the unqualified file name in the SQL statement will be overridden to use this file and the member specified in the Member prompt (MBR parameter).	<ul style="list-style-type: none"> ■ *NONE: Specifies that no file is overridden. If the SQL statement (or a statement used in the interactive SQL session) refers to an unqualified file, the copy of the file highest in the library list is used. ■ file-name: Specifies a file that will be used to override the same-named unqualified file in the SQL statement. <p>The name of the file can be qualified by one of the following library-values:</p> <ul style="list-style-type: none"> ■ *LIBL: Searches all libraries in the job's library list until the first match is found. ■ *CURLIB: Searches the current library for the job. If no library is specified as the current library for the job, the QGPL library is used. ■ library-name: Specifies the name of the library to be searched.
MBR	Specifies the file member to be overridden to. This parameter is ignored if FILE (*NONE) is specified.	<ul style="list-style-type: none"> ■ *FIRST Specifies the file in the FILE parameter is overridden to use the *FIRST member. ■ member-name Specifies the file in the FILE parameter is overridden to use the specified member.

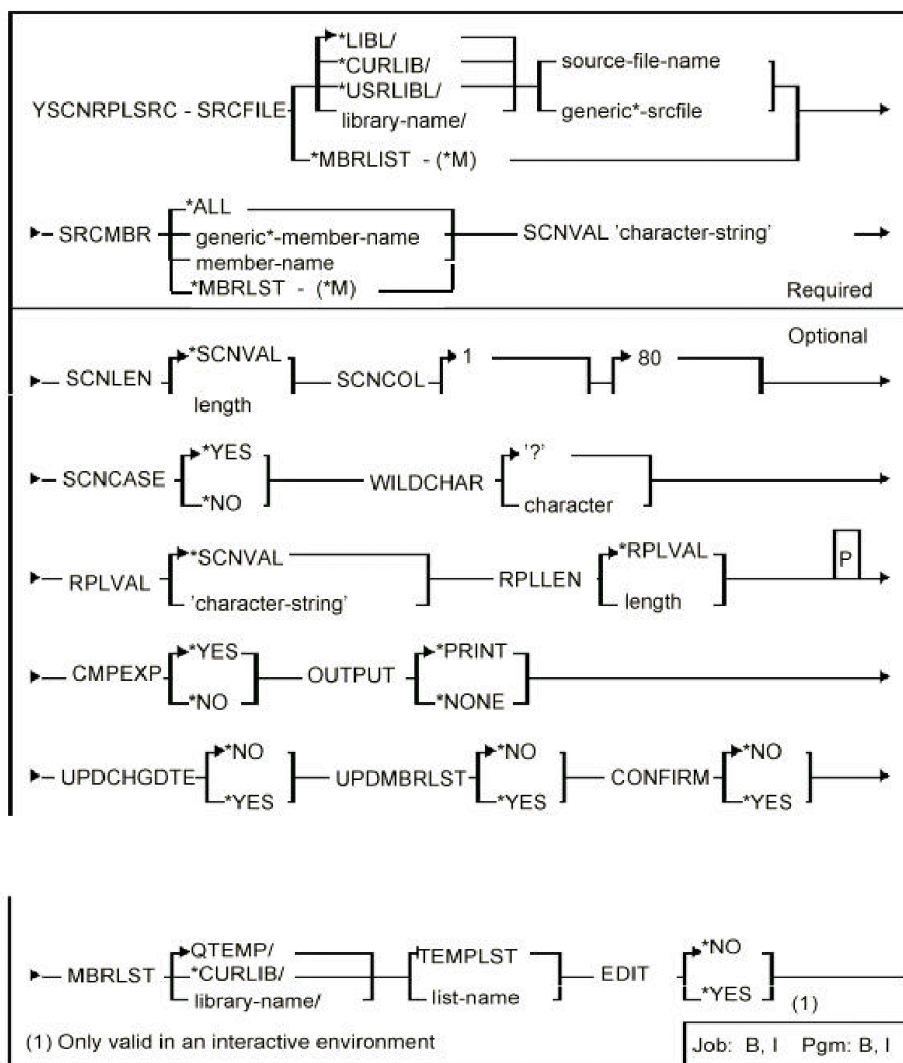
Parameter	Definition	Value and Description
OUTPUT	Specifies whether the output from the command is shown at the requesting work station, printed with the job's spooled output, or directed to a database file.	<ul style="list-style-type: none">■ *: Indicates the output produced by the query is formatted and, in interactive mode, sent to the work station that runs the command. If the command is run in batch mode, the output is sent to the default printer used by query management.■ *PRINT: Indicates the output produced by the query is formatted and sent to the default printer used by query management.■ *OUTFILE: Indicates the output produced by the query is written to the database file specified in the Output File prompt (OUTFILE parameter).
OUTFILE	Specifies the database file to receive the query output. If the file specified does not exist, the system creates it in the specified library as a table in a collection. If the file is created by this function, the authority for users without specific authority is *EXCLUDE.	<ul style="list-style-type: none">■ database-file-name: Specifies the name of the database file that receives the output of the command. The name of the database file can be qualified by one of the following library values:<ul style="list-style-type: none">■ *LIBL: Searches all libraries in the job's library list until the first match is found.■ *CURLIB: Searches the current library for the job. If no library is specified as the current library for the job, the QGPL library is used.■ library-name: Specifies the name of the library to be searched.

Parameter	Definition	Value and Description
OUTMBR	Specifies the name of the database file member to which the output is directed.	<ul style="list-style-type: none">■ *FIRST: Indicates that the first member in the file receives the output. If no members exist in the file, the system creates a member with the name of the file specified in the Output File prompt (OUTFILE parameter).■ member-name: Specifies the name of the file member that receives the output. If the name does not exist, the system creates it. <p>The possible actions to take values are:</p> <ul style="list-style-type: none">■ *REPLACE: Clears the file before new records are inserted.■ *ADD: Adds new records after any existing record.

YSCNRPLSRC (Scan/Replace Source Member)

This command scans a list of source file members searching for instances of a specified combination of characters. The list can be specified by a generic name or by the member list function. On finding an instance of the search character string, replaces it with another specified character string. The search and replacement strings may, optionally, be displayed for confirmation before replacement CONFIRM parameter.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
SRCFILE	Qualified generic name of source file or files which are to be scan/replaced	*MBRLST: (default) Use the named member list to obtain source member names
SRCMBR	Generic name of source members to scan/replace	*ALL: (default) Search all source members in specified file or files
SCNVAL	Search mask: Characters that are replaced	String of up to 80 characters. Strings that contain embedded blanks or lower case characters should be enclosed in quotes
SCNLEN	Length of search mask	<ul style="list-style-type: none"> ■ *SCNVAL: (default) Use search mask up to last non-blank character ■ 1-80: Use specified length
SCNCOL	Column range in source that is to be scanned for the search mask. Made up of two values	<ul style="list-style-type: none"> ■ Starting column in source line for search <ul style="list-style-type: none"> – 1: (default) Value – 1-240: Column number at which to start scanning ■ Ending column source line for search <ul style="list-style-type: none"> – 80: (default) Value – 1-240: Column number at which to stop scanning (must be greater than or equal to start column number)
SCNCASE	Case-sensitive option on search	<ul style="list-style-type: none"> ■ *YES: (default) Treat upper and lower case characters as different for comparison purposes when searching for occurrences of the specified string ■ *NO: Ignore differences between upper and lower case for comparison purposes

Parameter	Definition	Value and Description
WILDCHAR	Character to be used as a “dummy” in any search mask; any character counts as a match during the search	`?': (default) Value
RPLVAL	Replacement string of characters that is to replace the character string specified by the SCNVAL parameter	<ul style="list-style-type: none"> ■ *SCNVAL: (default) Replacement string is same as scan string Character string up to 80 characters long. If the string contains embedded blanks or lower case characters, it should be enclosed in quotes
RPLLEN	Length of replacement string	<ul style="list-style-type: none"> ■ *RPLVAL: (default) Use replacement string up to last non-blank character ■ 1-80: Use specified length
CMPEXP	Compress or expand line	<ul style="list-style-type: none"> ■ *YES: (default) If replace length is different from scan length, justify line to omit/or insert blanks. If there is insufficient space, line is not adjusted ■ *NO: Do not justify line. If replacement length is shorter than scan length, blanks will be inserted. If it is longer, it is shortened
OUTPUT	Print option	<ul style="list-style-type: none"> ■ *PRINT: (default) Print a report of lines changed ■ *NONE: Do not print report
UPDCHGDTE	Update source change dates.	<ul style="list-style-type: none"> ■ *NO: (default) Do not update the source change dates ■ *YES: Update the source change date for each line amended
UPDMBRLST	Update member list option.	<ul style="list-style-type: none"> ■ *NO: (default) Do not update the member list ■ *YES: Delete all members in the list for which no lines satisfy the search

Parameter	Definition	Value and Description
CONFIRM	Prompt for confirmation	<ul style="list-style-type: none"> ■ *NO: (default) No confirmation is required ■ *YES: Prompt for confirmation of each replacement of the search string
MBRLST	Qualified name of a member list	QTEMP/TEMPLST: (default) List name
EDIT	List option	<ul style="list-style-type: none"> ■ *NO: (default) No editing is required ■ *YES: Invoke the edit member list function to edit the list before executing the scan

Notes

1. The command works on a source record length of up to 240 characters.
2. If the replacement character string is longer than the scan pattern, some data may be lost. This can happen in any of the following three circumstances:
 - CMPEXP(*NO) is specified and the replacement string overrides source characters to the right of the found scan.
 - CMPEXP(*YES) is specified and the rightmost source characters are expanded off the end of the line.
 - There is not enough room for the replacement string and the replacement takes but is shortened.

Examples

To scan all members in file QTXTSRC for the characters 'Geneva', and replace all found instances with 'Des Moines':

```
YSCNRPLSRC SRCFILE(*LIBL/QTXTSRC) SCNVAL('Geneva') RPLVAL('Des Moines')
```

To scan all members in file QRPGRSRC in library TOMATO whose names begin with the letters MY for the characters STKCOD in columns 33-38, and replace any found instances of the search characters with the characters STKCDE. Print a list of changes made, and also build a member list, called STKCDELIST, of all the source members in which a replacement has taken place:

```
YSCNRPLSRC SRCFILE(TOMATO/QRPGSRRC) SRCMBR(MY*) SCNVAL(STKCOD) SCNCOL(33 38)
RPLVAL(STKCDE)
```

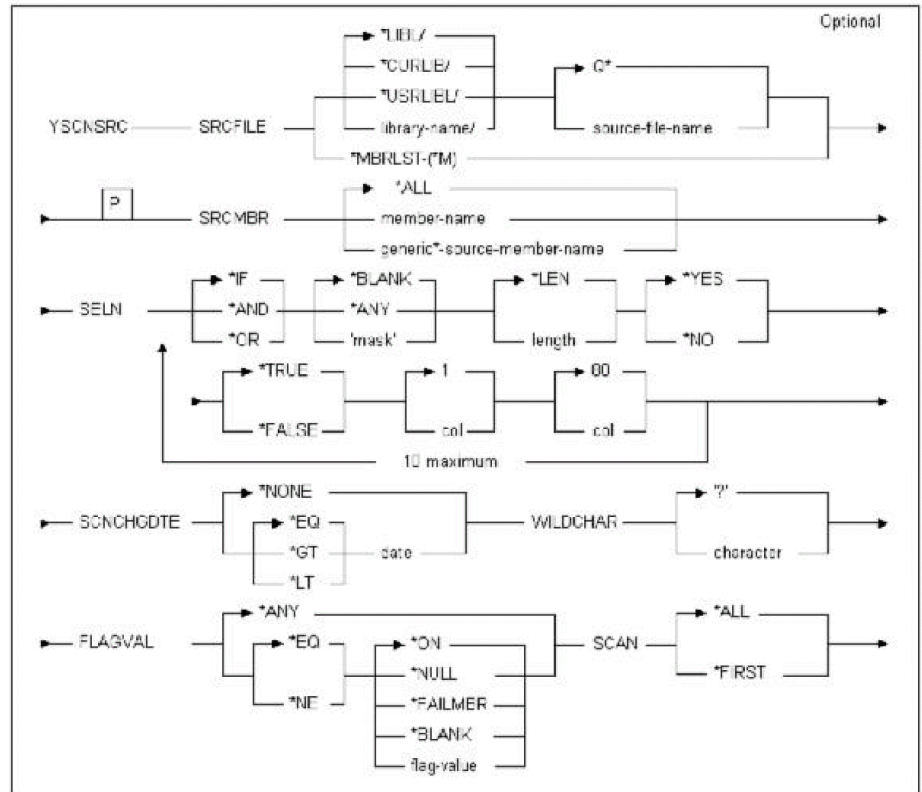
UPDMBRLST(*YES) MBRLST(QTEMP/STKCDELIST)

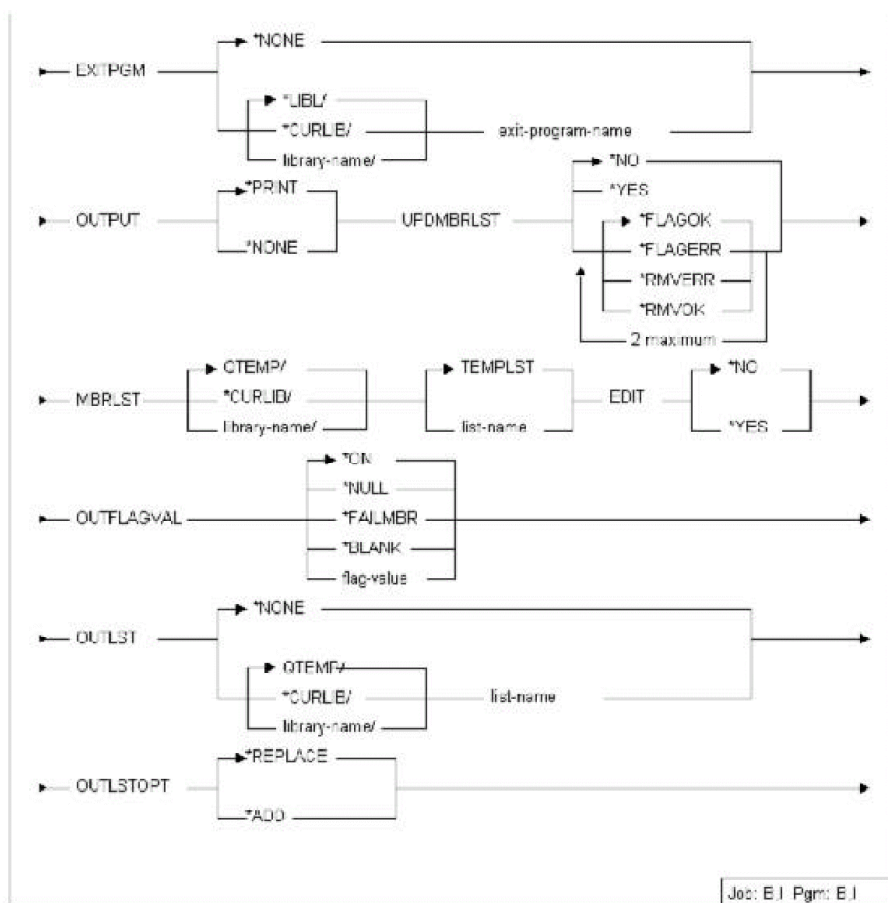
YSCNSRC (Scan Source Member)

This command scans a list of source file members searching for specified combinations of characters. The list can be specified by a generic name or by the member list function. Produces a report, and/or maintains a member list (input or output), of those members that contain source lines satisfying the search conditions. Source selection can further be controlled by means of a user-written exit program.

Syntax Diagram

This diagram appears on the following page.





Parameters

Parameter	Definition	Value and Description
SRCFILE	Qualified generic name of source file or files which are to be scanned	<ul style="list-style-type: none"> ■ *LIBL/Q*: (default) Value ■ *MBRLST: Use the named member list to obtain source member names
SRCMBR	Generic name of source members to scan	*ALL : (default) Search all source members

Parameter	Definition	Value and Description
SELN	A list of up to ten search elements. Each element in the list is made up of the following seven fields	<p>The logical operation for combining this element with other elements.</p> <ul style="list-style-type: none"> ■ *IF: (default) Only valid for the first element ■ *AND: This element combines with the previous element(s) in an *OR group ■ *OR: This element starts a new *OR group <p>Search mask: Up to 21 characters. Can contain any character. Lower case characters should be enclosed in quotes, for example, 'qtxt'. The mask may contain the wild card character specified by the WILDCHAR parameter.</p> <ul style="list-style-type: none"> ■ *BLANK: (default) Scan for blank characters ■ *ANY: All source records are selected. This option normally is used in combination with the SCNCHGDTE parameter <p>Length of search mask</p> <ul style="list-style-type: none"> ■ *LEN: (default) Use search mask up to last non-blank ■ 1-21: Use specified length <p>Case-Sensitive option</p> <ul style="list-style-type: none"> ■ *YES: (default) Treat upper and lower case characters as different for comparison purposes ■ *NO: Ignore differences between upper and lower case for comparison purposes. Thus a search mask value of 'a' is matched with both a and A

Parameter	Definition	Value and Description
		True/False option <ul style="list-style-type: none">■ *TRUE: Search mask must be in the line■ *FALSE: Search mask must NOT be in the line Starting column at which to begin search <ul style="list-style-type: none">■ 1: (default) Value■ 1-80: Start scanning at this column Ending column for search. <ul style="list-style-type: none">■ 80: (default) value.■ 1-80: stop scanning at this column
SCNCHGDTE	Source change date for which to scan. Only records which meet the specified date criteria (in addition to any search criteria specified by the SELN parameter) is selected	<ul style="list-style-type: none">■ *NONE: (default) Do not scan for source change dates Otherwise, SCNCHGDTE is a list parameter made up of the following two elements: <ul style="list-style-type: none">■ Relational operator for selection of changed source records<ul style="list-style-type: none">– *EQ: (default) Scan records which were changed on a specified date– *GT: Scan records which were changed after a specified date– *LT: Scan records which were changed before a specified date– Source change date for which to scan in QDATFMT format
WILDCHAR	Character used as a dummy in any search mask; will match any character during the scan	`?': (default) value

Parameter	Definition	Value and Description
FLAGVAL	Flag value of list items to be processed	<ul style="list-style-type: none"> ■ *ANY: (default) Scan all items in list <p>Otherwise, FLAGVAL is a list parameter which is used to determine whether the list item will be scanned. It is made up of the following two elements:</p> <ul style="list-style-type: none"> ■ Relational operator for selection of flags <ul style="list-style-type: none"> – *EQ: (default) Equal to – *NE: Not equal to ■ Flag value <ul style="list-style-type: none"> – Single character flag value or one of the special flag values
SCAN	Scan stop option: denotes whether to stop scanning a specific member after finding a line which satisfies the search conditions	<ul style="list-style-type: none"> ■ *ALL: (default) Scan and print all lines ■ *FIRST: Stop after finding one instance of the search string in the member
EXITPGM	Qualified name of exit program to be called for user- defined override of match selection	*NONE: Do not call an exit program
OUTPUT	Print option	<ul style="list-style-type: none"> ■ *PRINT: (default) Print a report of lines found ■ *NONE: Do not print a report

Parameter	Definition	Value and Description
UPDMBRLST	Update member list option	<ul style="list-style-type: none">■ *NO: (default) Do not update the member list■ *YES: Update the member list by removing members which do not contain the search string. This is equivalent to *RMVERR. <p>Or up to two of the following values:</p> <ul style="list-style-type: none">■ *FLAGERR: (default) If a member does not contain the specified search string, then its list entry is to be flagged with the value specified by the OUTFLAGVAL.■ *FLAGOK: If a member does contain the specified search string, then its list entry is to be flagged with the value specified by the OUTFLAGVAL■ *RMVERR: If a member does not contain the specified search string, then its list entry is to be removed from the list (equivalent to *YES)■ *RMVOK: If the member does contain the specified search string, then its list entry is to be removed from the list <p>If more than one value is specified, *RMVERR + *FLAGOK and *RMVOK + *FLAGERR are the only valid combinations</p>
MBRLST	Qualified name of a member list	QTEMP/TEMPLST: (default) List name
EDIT	List edit option	<ul style="list-style-type: none">■ *NO:(default) No editing is required■ *YES: Invoke the edit member list function to edit the list before executing the scan

Parameter	Definition	Value and Description
OUTFLAGVAL	Flag value to be given to selected items in resulting list if UPDMBRLST is *FLAGOK or *FLAGERR	Single character flag value or one of the special flag values
OUTLST	Qualified name of member list to which selected entries are copied	<ul style="list-style-type: none"> ■ *NONE: (default) Selected list entries are not copied to an output list ■ QTEMP/TEMPLSTOUT: Default output list name
OUTLSTOPT	Output list replacement option for the list specified by the OUTLST parameter	<ul style="list-style-type: none"> ■ *REPLACE: (default) Creates a new output list, replacing any previous list's contents ■ *ADD: Add to any existing list's contents

Notes

1. The SCAN(*FIRST) option normally results in a faster scan.
2. To use the Exit program function, you should write a program with a parameter interface as indicated by the following CL program fragment:

```
PGM PARM(&SRCDTA &SRCSEQ &SRCDAT &FILE &LIB &MEMBER &SEUTYPE &MATCH)
```

```

DCL &SRCDTA  *CHAR 80      /* INPUT: Matching source line */
DCL &SRCSEQ  *DEC(6 2)     /* INPUT: Source sequence number */
DCL &SRCDAT  <%-2>*DEC(6 0) /* INPUT: Source line change date */
DCL &FILE    *CHAR 10      /* INPUT: File name of member */
DCL &LIB     *CHAR 10      /* INPUT: Library name of member */
DCL &MEMBER  *CHAR 10      /* INPUT: Member name */
DCL &SEUTYPE *CHAR 8       /* INPUT: SEU type of member */
DCL &MATCH   *DEC(1 0)     /* OUTPUT: set to 0 for no match */
                        /*          set to 1 for match */
...
ENDPGM
```

Examples

To scan all members in file(s) QTXTSRC in the library list for the characters 'Geneva', and print all found lines:

```
YSCNSRC SRCFILE(*LIBL/QTXTSRC) SELN(( *IF 'Geneva'))
```

To scan all members in the file library TOMATO whose names begin with the letters 'MY' for the characters 'CALL' in columns 28-31, and 'C' in column 6, build a member list called PGMCALLS in library QTEMP of all members containing at least one occurrence of the specified strings:

```
YSCNSRC SRCFILE(TOMATO/QRPGSRC) SRCMBR(*ALL) SELN((*IF 'C' 1 *YES *TRUE 6 6)
(*AND 'CALL' *LEN *YES *TRUE 28 31)) SCAN(*FIRST) OUTPUT(*NONE) UPDMBRLST(*YES)

MBRLST(QTEMP/PGMCALLS)
```

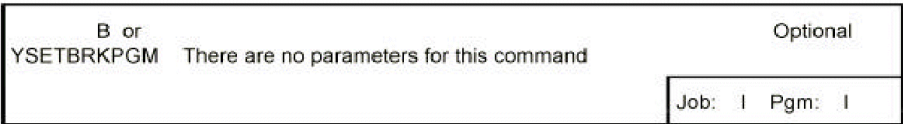
To scan the member list built in example B for those members containing the string 'QCAEXEC'. Place the found items in a new list called 'QCA':

```
YSCNSRC SRCFILE(*MBRLST) SELN(*IF 'QCAEXEC') MBRLST(QTEMP/PGMCALLS)
UPDMBRLST(*RMVERR)
OUTLST(QTEMP/QCA)
```

YSETBRKPGM (Set Break Program)

This command creates a message queue and sets a break program to serve it. The message queue can then be used to interrupt a job at any time with any request, using the system request menu.

Syntax Diagram



Parameters

None

Notes

The command creates a message queue called Y in library QTEMP, and sets a break program to receive and execute any messages sent to the queue. Option 5 on the system request display (send break message) can then be used to send any arbitrary request string to the message queue at any point during the execution of an interactive job.

Examples

To set the break program for the invoking job:

```
YSETBRKPGM
```

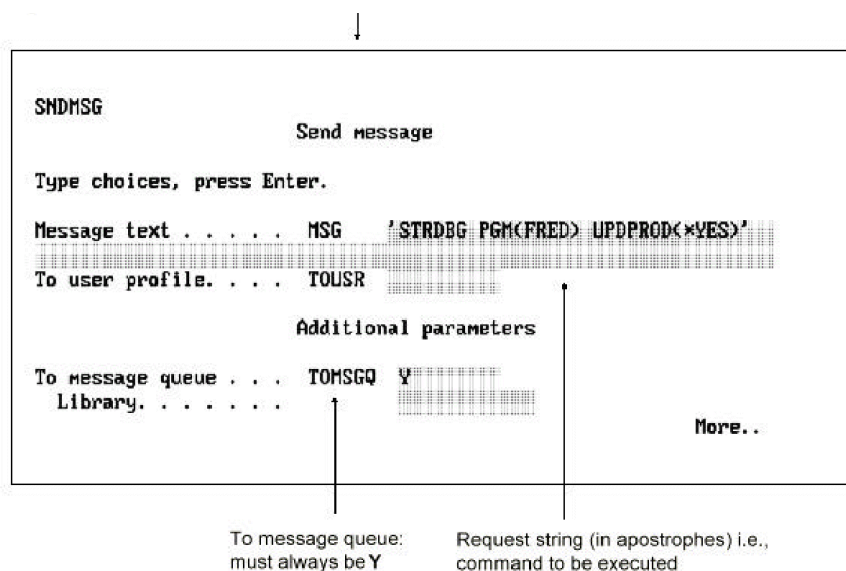
You might then call a user program FRED:

```
CALL FRED
```

Then at any point while running program FRED you wish to start debugging you can do so by transferring to the system request menu and sending a message to the Y queue:

System request		System: J70038
Select one of the following:		
<ul style="list-style-type: none">1. Display sign on for alternative job2. End previous request3. Display current job4. Display messages5. Send a message6. Display System operation messages		
90. Signoff		
Selection		
5		
F3=Exit F12=Previous		

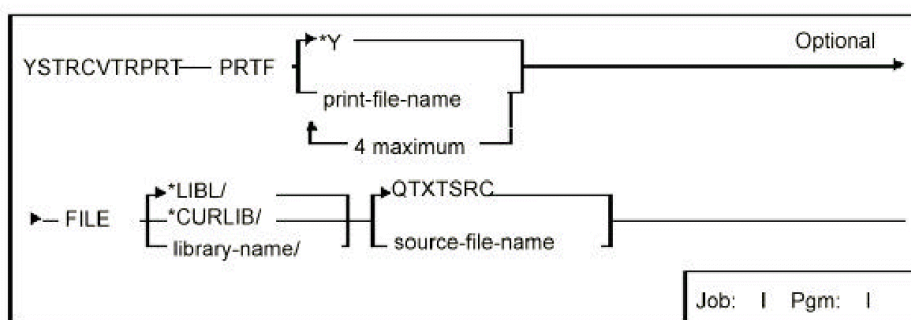
Option 5 displays as follows:



YSTRCVTPRT (Start Convert Print Key Output Mode)

This command starts print output conversion mode. Used in conjunction with the command Convert Print Output (YCVTPRT), this command provides an easy method of obtaining illustrations for documentation and instruction manuals.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
PRTF	List of names of up to four print files receiving print output which is to be converted	*Y: (default) QSYSPRT and YPRTKEY\$ are print file names
FILE	Name of source file to be used as default when subsequently using the YCVTPRT command	*LIBL/QTXTSRC: (default) Source file name

Notes

1. This command sets up overrides to direct all print output from the nominated print files to an output queue (YPRINT) to await subsequent conversion into source members using the YCVTPRT command.
2. One restriction is that the YSTRCVTPRT command will invoke the i OS command entry program (QCMD) after setting up the print file redirection: print file output must be created and converted at the same or a lower invocation level as the invocation of qcmd by YSTRCVTPRT.
3. Once the YSTRCVTPRT command has been invoked, the subsequent output to any of the nominated print files by the current job is redirected to a work output queue. In particular, *the output from the workstation print key can be redirected*.

The command Convert Print Output (YCVTPRT) can then be invoked to convert all or some of the print output into a source file member. See the YCVTPRT command diagram for further details.
4. To end print key conversion, press F03 from the i OS command entry display (QCMD).
5. The following print files are normally used for print key output:
 - i OS commands and displays QSYSPRT
 - CA 2E displays: YPRTKEY\$

Example

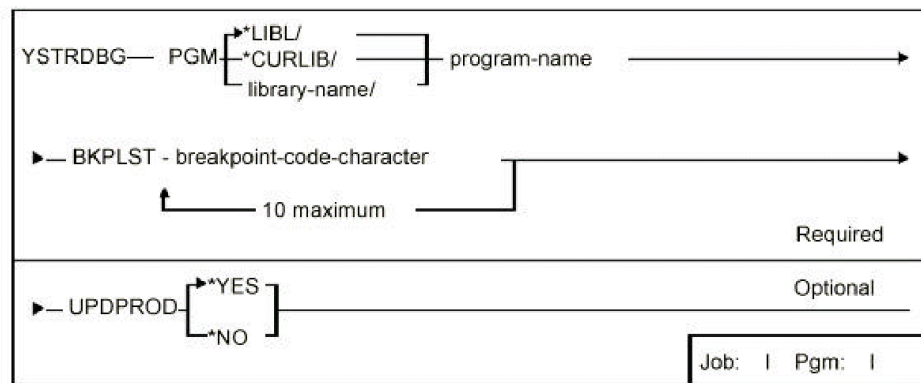
To set up a print key conversion environment for output to print files QSYSPRT and FRED to QTXTSRC in library QGPL:

```
YSTRCVTPRT PRTF(QSYSPRT FRED) FILE(QGPL/QTXTSRC)
```

YSTRDBG (Start Debug and Add Auto Breakpoints)

This command starts debug mode for a specified program, then retrieves and applies break point statements from the source member of the program. Up to ten different sets of breakpoints can be activated in one execution.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
PGM	Qualified name of the debugged program	
BKPLST	A list of character codes which identify the breakpoint sets to be activated	
UPDPROD	Update production files option	<ul style="list-style-type: none"> ■ *YES: (default) Data in production libraries can be updated ■ *NO: i OS prevents update of data in production libraries

Notes

1. The YSTRDBG command will work on both CL and RPG III programs.

2. The YSTRDBG command needs to access the source member for the program being debugged**D* Debug source directives. To find the source member of the program being debugged, the YSTRDBG command uses the source file member specified on the object description. (You can use the i OS command Display Object (DSPOBJD) with DETAIL(*FULL) to see which member is specified.) If the program source has been moved or renamed since the program was last compiled, a prompt will be issued for the current source member location.
3. To store a breakpoint in the source, insert a '*D**' comment line immediately before the executable line that you wish to have the breakpoint inserted.

For RPG III source the '*D**'comment line has the following format:

123456789 123456789 123456789

XXXXXD*field1 {field2...}

where

- XXXXX may be up to five single characters (A-Z, or #,\$,@).Each character can identify which breakpoint set(s)the '*D**'comment line belongs to. The characters should start in column one.
- '*D**'must be in columns six and seven.
- *Field X* is the name of a valid program variable that you wish to have displayed at the breakpoint.

For example; 'ACMX *D**...'belongs to breakpoint sets A, C, M and X, and will be activated if one or more of these letters is in the breakpoint list entered on the command.

For CL programs the comment line has the following format:

/*D:XXXXX '&field1'&field2'...*/

where

- '*/*D*:' must begin the comment.
- XXXXX may be up to 5 single characters (A-Z, or #,\$,@).Each character can identify which breakpoint set(s)the '*D**'comment line belongs to. The characters must be contiguous; for example, '*/*D*:ABC', not '*/*D*:A BC'.
- '&*FieldX*' is the name of a valid program variable that you wish to have displayed at the breakpoint. The field name(s) must be in single quotes. For example, '*/*D*:ACMX' belongs to breakpoint sets A,C,M and X, and will be activated if one or more of these letters is in the breakpoint list entered on the command.

If any code on a line prefix matches any of the code specified for the BKPLST parameter, then a breakpoint is added for the next statement within the source. Note that if you add lines to the source without recompiling the program, then the stored breakpoints may be invalid.

Examples

If, for instance, the following source lines are in the source member for an RPG/400 program called BLATTELA:

```

0100.00 A    D*
0101.00      C                      MOVE YQMNC D      XXMNC D 10
0102.00      C                      MOVE YQOPUN      XXOPUN 10
0103.00      C                      Z-ADD1          #R      30
0104.00 A B  D* XXMNC D XXOPUN
0105.00      C                      YQSBJB  COMP '0'                      99
0106.00 B    D* *IN99
0107.00      C                      YQMNC D  CASEQ' FRED'      BADSMN

```

Then:

```
YSTRDBG PGM(BLATELLA) BRKLST(A B)
```

would be equivalent to the following:

```
STRDBG BLATTELA UPDPROD(*YES)
```

```
ADDBKP BKP(10100)
```

```
ADDBKP BKP(10500) PGMVAR('XXMNC D' 'XXOPVN')
```

```
ADDBKP BKP(10700) PGMVAR('*IN99')
```

If, for instance, there are the following source lines in the source member for a CL program called AMERICANA:

```

0100.00 /*D: A  */
0101.00      RTUOBJA JOB(&JOB) USER(&USER)
0102.00 /*D: AB ' &JOB' ' &MBR' */
0103.00      CHGVAR VAR(&MBR) VAR(&MBR) VALUE(&JOB)
0104.00      ' &MBR'
0105.00      OURDBF FILE(YALL) TOFILE(HAVA/NICEDAY)

```

Then:

```
YSTRDBG PGM(AMERICANA) BRKLST(A B)
```

would be equivalent to the following:

```
STRDBG AMERICANA UPDPROD(*YES)

ADDBKP BKP(10100)

ADDBKP BKP(10300) PGMVAR('&JOB' '&USER')

ADDBKP BKP(10500) PGMVAR('&MBR')
```

YSTRSPLRTR (Start spooled file router)

The Start spooled file router (YSTRSPLRTR) command is used to start a spooled file router job to monitor a specified data queue which has previously been attached to one or more output queues. When spooled files on those output queues reach Ready (RDY) status, the spooled file router processes the spooled file (see below for details).

If this command is called interactively, it submits the spooled file router job to batch, using a job description called YSPLRTRJD. If this command is called in batch, the spooled file router job runs under the submitting job profile.

Note: The YSPLRTRJD job description is shipped with this command. You should ensure that it is changed to use a library list that contains all necessary libraries to execute all commands that you specify in the YSPLRTRP file.

The processing that takes place for each spooled file is controlled by the contents of the YSPLRTRP file. Spooled files are compared with records in YSPLRTRP and all matching records on YSPLRTRP (that is, all records on YSPLRTRP where the OUTQ, OUTQLIB, USER, JOB, FILE and USRDTA fields match the spooled file definition and attributes) are processed (see below for processing details) in PRIORITY order.

Note: If any of the OUTQ, OUTQLIB, USER, JOB, FILE or USRDTA fields on YSPLRTRP has the value '*ALL', then that field is assumed to match the equivalent value for the spooled file being processed. Thus a record with all the above fields set to '*ALL' will be considered to match every spooled file.

Note: If a matching record has PRCTYP = 'E' (Exclusive), then after that record has been processed, no further processing is performed on the spooled file. If a matching record has PRCTYP = 'D' (Deleted), then that record is not processed.

Parameters

Parameter	Definition	Value and Description
DTAQ	Specifies the data	

Parameter	Definition	Value and Description
	<p>queue to be monitored. This data queue should previously have been created with MAXLEN(128) and SEQ(*FIFO), for example: CRTDTAQ DTAQ(QGPL/YSPLRTRQ) MAXLEN(128) SEQ(*FIFO) And have been attached to one or more output queues by using the Create Output Queue (CRTOUTQ) or Change Output Queue (CHGOUTQ) command specifying this data queue on the DTAQ parameter: CHGOUTQ OUTQ(QGPL/QPRINT) DTAQ(QGPL/YSPLRTRQ)</p> <p>Note: The data queue can have any name, not just YSPLRTRQ, and can be created in any library, not just QGPL, as in the example above. However, a data queue called YSPLRTRQ must be created in QGPL as above, to act as a default data queue for this command.</p> <p>If a data queue is attached to an output queue, an entry will be added to the data queue every time a spooled file reaches Ready (RDY) status on the output queue. It is</p>	

Parameter	Definition	Value and Description
	<p>these entries which are monitored for by the spooled file router.</p> <p>Note: A single data queue can be attached to several output queues, or you can create different data queues for different output queues.</p>	
YSPLRTRQ	<p>Use the default spooled router data queue YSPLRTRQ. This data queue must already have been created as above.</p> <p>data-queue-name Enter the name of an existing data queue with the correct attributes (*FIFO, maximum length 128).</p>	<ul style="list-style-type: none"> ■ *LIBL (Default): All libraries in the user and system portions of the job's library list are searched until the first match is found for the specified data queue. ■ *CURLIB: The current library for the job is used to locate the specified data queue. If no library is specified as the current library for the job, the QGPL library is used. ■ library-name: Specify the name of the library to be searched for the specified data queue. ■ Errors allowed (ERRLVL): Specifies the maximum number of command errors that are tolerated during the running of the spooled file router. Command errors occur if the router processes a record in YSPLRTRP file and either the command does not exist, the parameters are incorrect or other errors occur (such as authority problems). If this value is greater than 0 (zero) and an error occurs whilst processing a command on a spooled file, but this value has not been reached, the spooled file router will not end, but no further records are processed for that spooled file (although any secondary commands will

Parameter	Definition	Value and Description
		still be processed). If this value is reached, the spooled file router will end. <ul style="list-style-type: none">■ *NOMAX (Default): No maximum number of errors is specified, and all command errors are tolerated.■ number-of-errors: Specify a value that specifies the maximum number of command errors that is tolerated.
LOG	Specifies whether commands processed for spooled files should be written to the job log of the spooled file router job.	<ul style="list-style-type: none">■ *NO (Default): Commands are not written to the job log unless an error occurs during the processing of the command, in which case the command is written to the job log following the error message.■ *YES: All commands are written to the job log of the spooled file router job. If an error occurs during the processing of a command, the error message is written to the job log following the command.

Notes

1. A primary command is built from the CMD and CMDLIB fields on YSPLRTRP, taking the FILE, JOB and SPLNBR parameters automatically from the spooled file.
2. Substitution variables in the PARMS field are replaced, as follows:
 - &DATE The current system date (in 'yymmdd' format)
 - &DAY The current system day
 - &DOC The document from which the spooled file was created (if applicable)
 - &FILE The spooled file name
 - &FLR The folder containing the document from which the spooled file was created (if applicable)
 - &HOUR The current system hour
 - &JOB The name of the job which produced the spooled file

- &JOBNNR The number of the job which produced the spooled file
 - &MINUTE The current system minute
 - &MONTH The current system month
 - &OUTQ The output queue name
 - &OUTQLIB The output queue library name
 - &PGM The program which created the spooled file
 - &PGMLIB The library containing the program which created the spooled file
 - &SECOND The current system second
 - &SPLNNR The number of the spooled file
 - &TIME The current system time (in 'hhmmss' format)
 - &USER The user of the job which produced the spooled file
 - &USRDTA The USRDTA attribute of the spooled file
 - &YEAR The current system year (in 'yy' format)
3. If a colon(';') is found within the PARMS field, any data up to the colon is considered primary command parameter data and is appended to the primary command above. Any data after the colon is considered secondary command data and is saved.
- If a colon(';') is not found within the PARMS field, all the data in the PARMS field is considered primary command parameter data and is appended to the primary command above.
4. The primary command is executed.
5. If any secondary command data exists, each secondary command is executed in turn.

Several secondary commands can exist, separated by colons(';'). Each secondary command must be a full command, including all necessary parameters. Any commands can be used as secondary commands and need not include any substitution variables, nor need they be related to the spooled file.

If errors occur during the processing of the primary command, any secondary commands will still be executed. However, no further primary commands will be executed for the spooled file.

Example

A spooled file QEZJOBLOG from job 123456/MYJOB/ME running on 17th December 2001 is matched to the following 2 records in YSPLRTRP:

```
CMD CMDLIB PARMS
```

```
CHGSPLFA QSYS USRDTA(JL_&DATE) PRTQLTY(*FASTDRAFT)
CPYSPLF *LIBL TOFILE(QGPL/JOBLOGS) TOMBR(JOB&JOBNNR);SNDMSG
MSG('Your file &FILE from job &JOB was copied
to QGPL/JOBLOGS - Go get it!') TOUSR(&USER)
```

This would result in the following 3 commands being executed:

```
QSYS/CHGSPLFA FILE(QEZJOBLOG) JOB(123456/MYJOB/ME) SPLNBR(00001)
USRDTA(JL_011217) PRTQLTY(*FASTDRAFT)
```

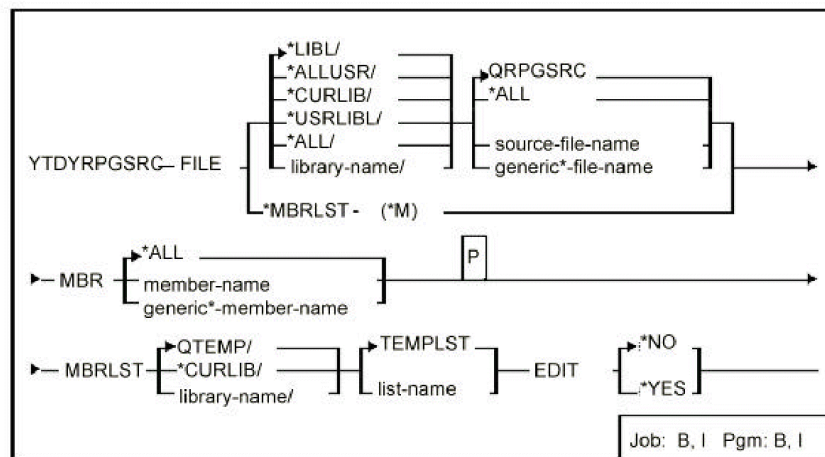
```
*LIBL/CPYSPLF FILE(QEZJOBLOG) JOB(123456/MYJOB/ME) SPLNBR(00001)
TOFILE(QGPL/JOBLOGS) TOMBR(JOB123456)
```

```
SNDMSG MSG('Your file QEZJOBLOG from job MYJOB was copied to QGPL/JOBLOGS - Go
get it!') TOUSR(ME)
```

YTDYRPFSCR (Tidy RPG III Source)

This command tidies up RPG/400 source to make it more readable. The beginning and end of structured programming constructs are labeled.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
-----------	------------	-----------------------

Parameter	Definition	Value and Description
FILE	Qualified generic name of source file containing RPG source members which are to be tidied	<ul style="list-style-type: none"> ■ *LIBL/QRPGSRC: (default) File name ■ *MBRLST: Use the named member list to obtain source member names. ■ *ALL: (default) Tidy all source files
MBR	Generic name of the members containing the RPG source which is to be tidied	*ALL: (default) Tidy all members in file
MBRLST	Qualified name of a member list	QTEMP/TEMPLST: (default) list name
EDIT	List option	<ul style="list-style-type: none"> ■ *NO: (default) No editing is required ■ *YES: Invoke the edit member list function to edit the list before executing the tidy

Notes

1. Places a label on the start and end statements of structured programming operations so that the nesting is clearer. The label on the END statement consists of the operation code backwards. The label is added as comment, starting in column 60.
2. ***WARNING!*** Any existing comments in columns 60-79 will be overwritten.

For instance:

For operation code:-				YTDYRPGSRC adds in columns 61-79:-			
C	1	DO	28	DO			
C	...	END		OD	1:28	I	

Also:

For operation code:			YTDYRPGSRC adds in columns 61-70:		
1	DO	20 I	END	OD	1:20 I
A	IFEQ	B	END	FI	A *EQ B
A	DOUGT	B	END	UOD	A *GT B
*IN	DOWN	'1'	END	WOD	*01 *NE '1'
A	CASEQ	B	END	SAC	A *EQ B

Example

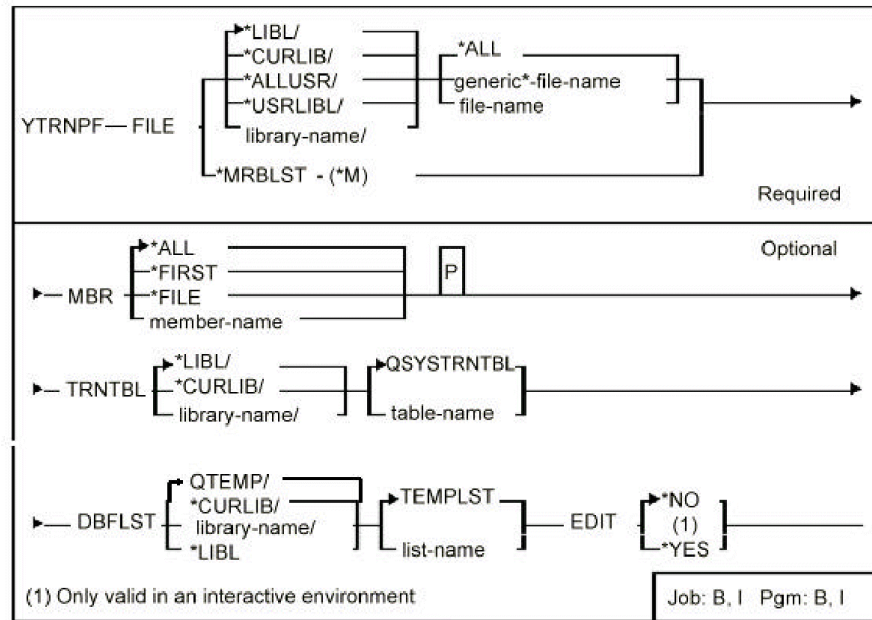
To tidy member FRED in QRPGRSRC in library QGPL:

```
YTDYRPGSRC FILE(QGPL/QRPGRSRC) MBR(FRED)
```

YTRNPF (Translate Physical File Data)

This command translates the alphanumeric data in a physical file member using a specified translation table. The physical file members can be specified generically, or a database file list can be used. Can be used to translate data into upper case, which is necessary for applications to run on ideographic workstations.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
FILE	Qualified generic name of file containing members whose alphanumeric data is to be translated	<ul style="list-style-type: none"> ■ *DBFLST: Translate data in files named in database file list specified with the DBFLST parameter ■ *ALL: Translate all physical files in the specified library
MBR	Generic name of members containing the data which is to be translated	<ul style="list-style-type: none"> ■ *ALL: (default) Translate data in all members in the specified file or files ■ *FIRST: Translate data in the first member in the specified file or files
TRNTBL	Qualified name of translation table to be used	QSYSTRNTBL: (default) Use the IBM-supplied system translation table

Parameter	Definition	Value and Description
DBFLST	Qualified name of database file list of files whose alphanumeric data is to be translated	QTEMP/TEMPLST: Default list name. Specify FROMFILE(*DBFLST) to use an existing database file list
EDIT	Edit list required	<ul style="list-style-type: none">■ *NO: (default) No editing required■ *YES: The edit database file list function will be invoked to edit the list before execution

Notes

This command converts contents of all alphanumeric fields in the specified files using the specified translation table.

Example

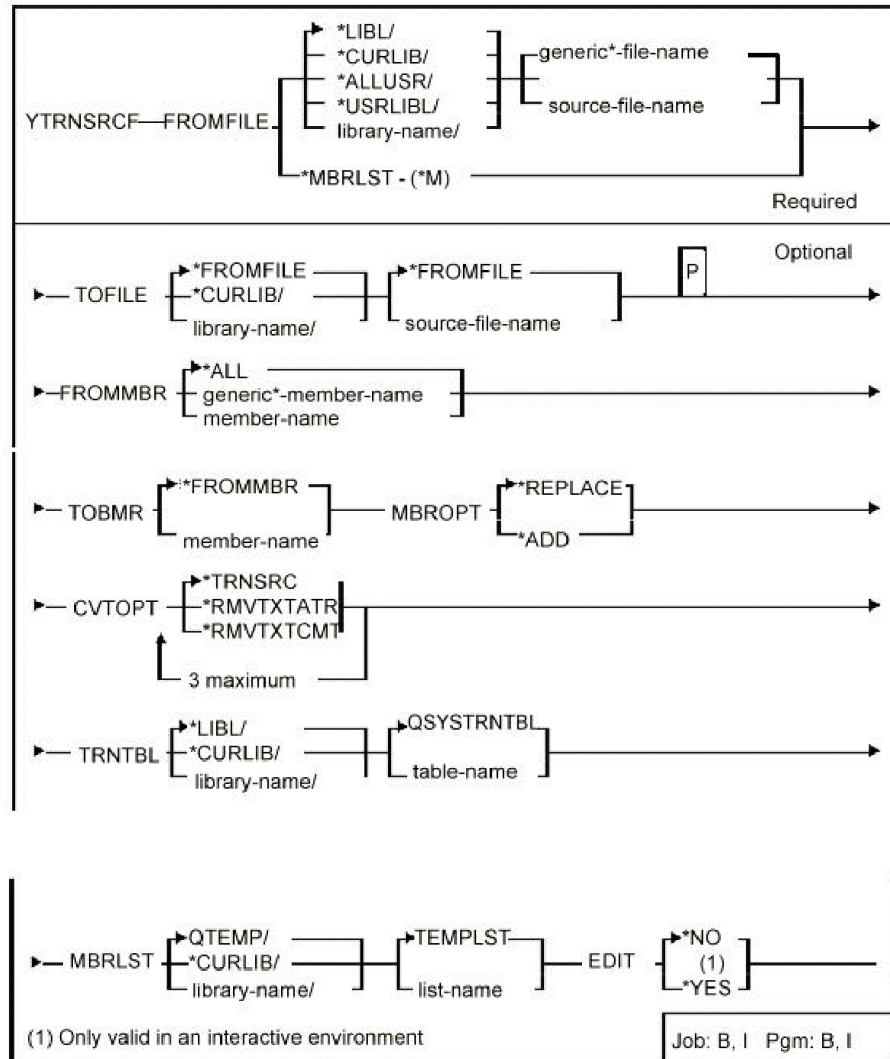
To convert the data in all members in file YUSRPRF in library QGPL to upper case:

```
YTRNPF FILE(QGPL/YUSRPRF)
```

YTRNSRCF (Translate Source File Data)

This command translates the data in source file members using a specified translation table. The source file members can be specified generically, or a member list can be used. Can be used to translate source into upper case, which is necessary for applications to run on ideographic workstations.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
FROMFILE	Qualified generic name of file containing source members which are to be translated	*MBRLST: Translate members named in member list specified by the MBRLST parameter
TOFILE	Qualified name of file into which translated source is to be placed	*FROMFILE: (default) Each member is output to a file of the same name as the file from which it was read
FROMMBR	Generic name of the members to be translated	*ALL: (default) Translate all members in the specified file or files
TOMBR	Name of member in TOFILE	*FROMMBR: (default) Member name is same as FROMMBR
MBROPT	Source member update option	<ul style="list-style-type: none"> ■ *REPLACE: (default) The translated source replaces any existing source member ■ *ADD: The translated source is appended to any existing source member contents
CVTOPT	Conversion options to use. Up to three values may be specified	<ul style="list-style-type: none"> ■ *TRNSRC: (default) Translate data according to table specified by the TRNTBL parameter ■ *RMVXTATR: Remove Text Management/38 hidden control information ■ *RMVXTTCMT: Remove Text Management/38 print control commands which are not generally used by the YDSPHLP program
TRNTBL	Qualified name of translation table to be used	QSYSTRNTBL: (default) QSYSTRNTBL i OS translation table, YTRNSRCF use the IBM-supplied system translation table

Parameter	Definition	Value and Description
MBRLST	Qualified name of member list of members to be translated	QTEMP/TEMPLST: (default) List name Specify FROMFILE(*MBRLST) to use an existing member list
EDIT	Edit list required	<ul style="list-style-type: none"> ■ *NO: (default) No editing required. ■ *YES: The edit member list function is invoked to edit the list before execution

Notes

1. This command translates the contents of the SRCDTA field in the specified source members using the specified translation table.
2. Specifying *RMVXTATR for the CVTOPT parameter will cause the following to be removed from members which are Text Management/38 documents:
 - Control characters for underlining and highlighting (stored as hidden extra lines).
 - Paragraph control information (which will be stored on the first line of a paragraph if there is no preceding blank line)
 - Paragraph definition, dictionary and other text document information residing at the end of the document.

Specifying *RMVXTTCMT for the CVTOPT parameter will cause any Text Management/38 comment lines and print control commands which are not generally used by the utility Display Help (YDSPHLP) to be removed from the document. The following control lines will not be removed using this option:

.Hn	.date	.*T
.Pn	.docid	.*YH
.SK n .		.*YI
KP xx		.*YV
.IM		.*Y

3. If the file name specified for the TOFILE parameter does not exist, it is created as a standard source file.

Example

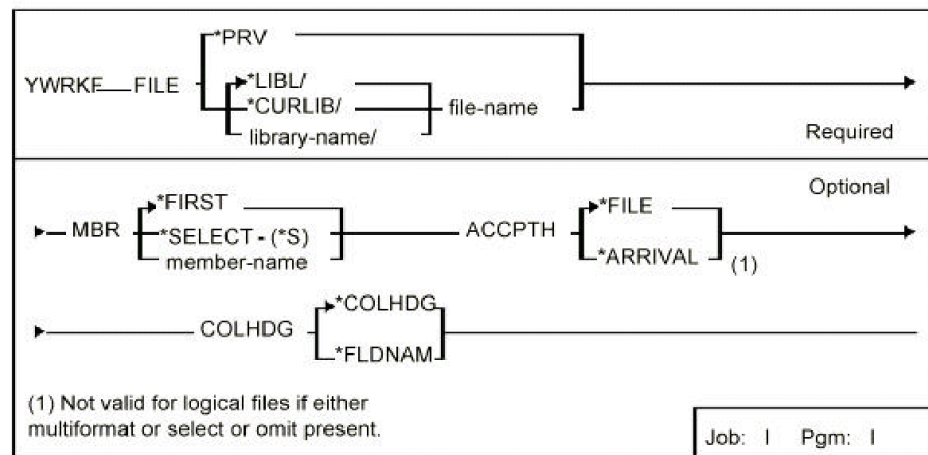
To translate all text members in source file QTXTSRC in library JAPAN to upper case:

```
YTRNSRCF FILE(JAPAN/QTXTSRC)
```

YWRKF (Work with Database File Data)

This command displays records from a specified database file. Records can be added, updated or deleted. The file can have either an arrival sequence, or a keyed access path that both allow records to be printed.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
FILE	Qualified name of file whose records are displayed and optionally changed. The file may be any database file	<p>*PRV: The file specified on the previous invocation of YWRKF within the current job is used. Using *PRV gives a substantially quicker response. Note that *PRV may be used when looking at a different member within the same file.</p>
MBR	Member containing data which is to be displayed	<ul style="list-style-type: none"> ■ *FIRST: (default) The first member in the file is used. ■ *SELECT: Display a list of the members within the specified file - one of which may be selected
ACCPH	Access path of file to be used	<ul style="list-style-type: none"> ■ *FILE: (default) Indexed files are accessed in key sequence. Arrival sequence files are accessed in record arrival sequence. ■ *ARRIVAL: Use arrival sequence path regardless

Parameter	Definition	Value and Description
COLHDG	Column heading option	<ul style="list-style-type: none">■ *COLHDG: (default) DDS column headings are used to identify data fields. If a field has no column headings, then the DDS name is used instead■ *FLDNAM: DDS field names are used to identify data fields• *NONE: For physical file and single-format logical files, no column headings will be used, allowing 15 lines of data records to be displayed. For multiple-format logical files, single separator lines will be used between data records of different record formats; each separator line will consist simply of the name of the record format. Note that F22 is ignored if COLHDG(*NONE) is specified.

Parameter	Definition	Value and Description
CFMPMT	Default Confirm value	<ul style="list-style-type: none">• *NO: When a record is added, updated or deleted in the file and the Enter key is pressed, a confirm prompt will appear defaulted to 'N'. If the Enter key is pressed again, the request will be cancelled. If the value is changed to 'Y', the request will be processed.• *YES: When a record is added, updated or deleted in the file and the Enter key is pressed, a confirm prompt will appear defaulted to 'Y'. If the Enter key is pressed again, the request will be processed. If the value is changed to 'N', the request will be cancelled.• *BLANK: When a record is added, updated or deleted in the file and the Enter key is pressed, no confirm prompt will appear and the change will be processed immediately. <p>e: If a 1-character data area called YWRKFDCA exists in the library list, the value of that data area will be used to determine the initial confirm prompt value, irrespective of the value entered for this parameter.</p>

Parameter	Definition	Value and Description
KEYOPT	Key Option	<ul style="list-style-type: none">• *NONE: No key options are used• *PROTECT: When an existing record is updated, the key fields are protected and cannot be changed. <p>*HIGHLIGHT: When a keyed record is displayed in multiple-record mode or is displayed, deleted, updated or created in single-record mode, the descriptive text for the key fields is highlighted.</p> <p>*BOTH: Processing for both the *PROTECT and *HIGHLIGHT parameter values will be used.</p>

Notes

1. This command calls an interactive display to update a file. Press Help while using the program for instructions.
2. You must have full data authority to the specified file, to update or add. If you do not have update or add authority, or if update or add is inapplicable, the utility operates in a display-only mode.

Example

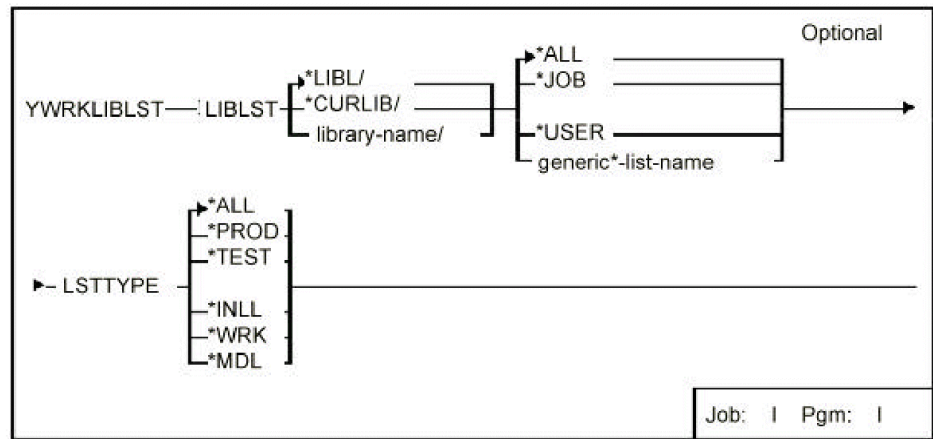
To display data from member FRED in file FRED in library QGPL:

```
YWRKF FILE(QGPL/FRED) MBR(FRED)
```

YWRKLIBLST (Work with Library List)

This command calls an interactive program to edit or change a library list or lists.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
LIBLST	Qualified generic name of library lists	<ul style="list-style-type: none"> ■ *ALL: (default) Displays a list of existing lists one or more of which can be selected ■ *JOB: Uses a stored library list having the same name as the current job ■ *USER: A stored library list having same name as the current job's user profile
LSTTYPE	Select library list type	*ALL: (default) Select all library lists

Notes

1. Library lists are stored in file YLIBLST in the library specified by the LIBLST parameter. It is recommended that you have only one library list file per installation. However, additional files can be created as follows:

```

CRTDUPOBJ OBJ(YLIBLST) FROMLIB('Product- library') OBJTYPE(*FILE)
TOLIB(library-name)

```

Examples

To work with all library lists:

YWRKLIBLST

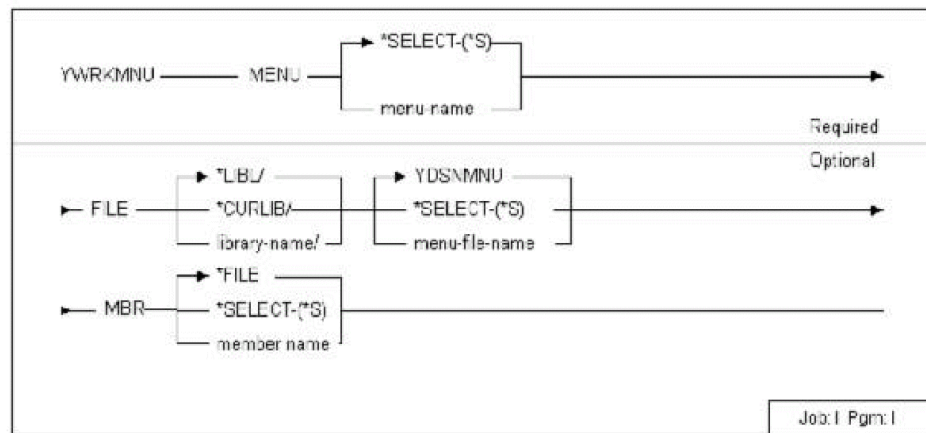
To work with all library lists of type TEST in library QGPL:

```
YWRKLIBLST LIBLST(QGPL/*ALL) LSTTYPE(*TEST)
```

YWRKMNU (Work with Menus)

This command calls an interactive program to work with menus.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
MENU	Name of menu which is to be changed or added	*SELECT: (default) A list of available menus is displayed. From this display it is possible to edit, copy, rename, delete, or print menus

Parameter	Definition	Value and Description
FILE	Qualified name of file containing menus	<ul style="list-style-type: none"> ■ YDSNMNU: (default) Menu file name ■ *SELECT: A list of the available menu files are displayed, one of which may be selected <p>Menu files should be created using the command Create Design File (YCRTDSNF)</p>
MBR	Name of member in file containing menus	<ul style="list-style-type: none"> ■ *FILE: (default) Member name is same as file name ■ *SELECT: A list of the existing members in the named menu file is displayed, one of which may be selected

Notes

This command calls an interactive program to edit menus. Press Help for instructions.

Example

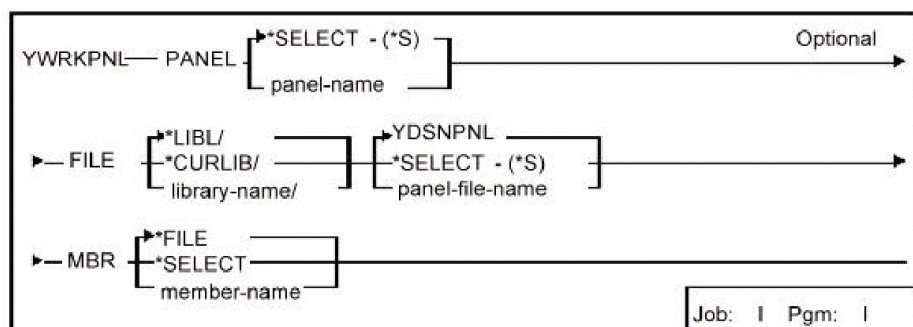
To work with menu DUJOUR in library MAXIMS:

```
YWRKMNU MENU(DUJOUR) FILE(MAXIMS/YDSNMNU)
```

YWRKPNL (Work with Panel Design)

This command calls an interactive utility to edit a panel design.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
PANEL	Name of panel design which is to be edited	*SELECT: (default) The panel selection display is presented
FILE	Qualified name of file containing panel design	<ul style="list-style-type: none"> YDSNPNL: (default) Panel file name *SELECT: Displays a list of panel design files, one of which can be selected <p>File must have been created with the command Create Design File (YCRTDSNF)</p>
MBR	Name of member in file containing panel design	<ul style="list-style-type: none"> *FILE: (default) The member has the same name as the file *SELECT: Displays a list of members, one of which can be selected

Notes

This command calls an interactive program to edit a panel design. Press Help for instructions.

Example

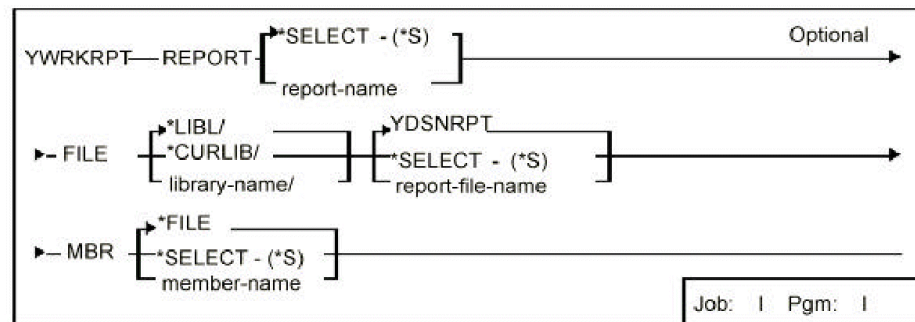
To work with panel MASK in the default design file:

YWRKP NL PANEL (MASK)

YWRKRPT (Work with Report Design)

This command calls an interactive utility to edit a report design.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
REPORT	Name of report design which is to be edited	*SELECT: (default) The report selection display is presented
FILE	Qualified name of file containing report design	<ul style="list-style-type: none"> YDSNRPT: (default) Report file name. *SELECT: Displays a list of report design file names, one of which can be selected. <p>File must have been created with the command Create Design File (YCRTDSNF)</p>
MBR	Name of member in file containing report design	<ul style="list-style-type: none"> *FILE: (default) The member has the same name as the file *SELECT: Displays a list of member names, one of which can be selected

Notes

This command calls an interactive program to edit a report design. Press Help for instructions.

Example

To work with report BACKFIRE in the default design file:

```
YWRKRPT REPORT(BACKFIRE)
```

YWRKSPLRTE (Work with Spooled File Routing Entries)

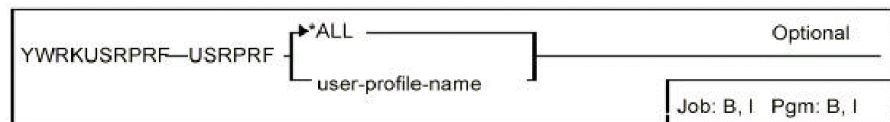
The W/W Spooled File Rtg. Entries (YWRKSPLRTE) command is used to add or change spooled file routing entries. Spooled file routing entries are used by spooled file router jobs to determine how spooled files should be processed.

There are no parameters for this command.

YWRUSRPRF (Work with User Profile)

This command works with the user profile and displays both the i OS user profile details, and the extension attributes.

Syntax Diagram



Parameters

Parameter	Definition	Value and Description
USRPRF	Name of user profile which is to be displayed	*ALL: (default) Display all user profiles

Notes

1. Only those profiles known are shown on the display. A profile becomes known when the commands Create User Profile (YCRTUSRPRF) or Change User Profile (YCHGUSRPRF) are applied to the profile.
2. Profiles can be selected from the display for change with the command Change User Profile (YCHGUSRPRF). Profiles can also be held or released directly from the display.

Example

To work with user profile ATTILLA enter:

```
YWRKUSRPRF USRPRF(ATTILLA)
```


Appendix A: Expanded Parameter Definitions

This appendix contains expanded descriptions of certain parameters which are common to several or many commands.

The CA 2E command parameter definitions meet one or both conditions:

- Have extensive information about how they are used
- Used in several commands and it is more efficient to describe them centrally

The list name parameters identify lists of objects, database files, or file members that are created or processed:

- LST
- OBJLST
- MBRLST
- FMTLST
- DBFLST

The list parameter is a qualified name made up of two elements:

- **Name of list:** Must be a valid i OS system name, ten characters long, begin with a letter, and contain no embedded blanks.
- **Name of library containing list:** Must be the name of an existing library, for example YEDTOBJLST OBJLST (QGPL/FRED)

N.B. Lists should not be stored in libraries whose names begin with the letter Q, apart from libraries QTEMP and QGPL.

LST and LSTTYPE Parameters

The LST parameter identifies the list name and the LSTTYPE parameter is specified to identify the type of list in commands that can process any type of list. For example, to delete an object list called FRED in library QTEMP:

YDLTLST LSTTYPE (OBJ) LST (QTEMP/FRED)

OBJLST, MBRLST, FMTLST and DBFLST Parameters

These parameters are used when commands can process only a particular type of list. The list name is specified with one of four list name keywords:

- OBJLST
- MBRLST
- YFMTLST
- DBFLST

For example, to print an object list called FRED in library QTEMP:

```
YD0C0BJLST OBJLST (QTEMP/FRED)
```

When indicating a generic command, indicate the list parameter, and the special value for the object. For example, to change the ownership of all objects in object list FRED:

```
YCHGOBJOWN OBJ (OBJLST) OBJTYPE (ALL) OBJLST (QTEMP/FRED)
```

To compile all members in member list FRED into library GEORGE:

```
YCRTOBJ OBJLIB (GEORGE) SRCFILE (MBRLST) MBR (ALL) MBRLST (QTEMP/FRED)
```

Default List Names

If no value is specified for the list name parameter, a default value of `TEMPLST` is used in the `QTEMP` library. This is a work list in the temporary library `QTEMP` that is deleted at the end of the job.

Work Lists

Most generic commands build a work list even if a list is not explicitly named. For instance, the following command results in the creation in `QTEMP` of a list containing all objects in library `QGPL` whose names begin with the letters `FR`:

```
YCHGOBJOWN OBJ (QGPL/FR) OBJTYPE (ALL)
```

SELECT Value for a List Name

A value of SELECT can be specified instead of a list name in many of the commands that use lists. This provides a display of existing lists. For example:

```
YEDTOBJLST OBJLST (QGPL/SELECT)
```

Lists can have qualified names. For instance, QGPL/FRED fact lists are stored as database file members, but the Object list QGPL/FRED is actually member FRED in file YOBJLST in library QGPL. The database file where a given list is stored has the same format as the output file of the related i OS display command, as shown in the following table:

List Type	Create CMD	DFT file	System related command	Sys output model file	Format name
DBF	YBLDDDBFLST	YDBFLST	DSPFD *ATR	QAFDPHY	QWHFDPHY
MBR	YBLDMBRLST	YMBRLST	*PHY	QAFDMBRL	QWHFDML
OBJ	YBLDOBJLST	YOBJLST	DSPFD	QADSPOBJD	QLIDOBJD
FMT	YBLDFMTLST	YFMTLSTYD	*MBRLIST	QAFDRFMT	QWHFDFMTQWHFDML
		BFLST file	DSPOBJD		format member lists
			DSPFD*RCDFMT		QWHFDPHY format
			DSPFD i OS command		DBF lists
			DSPOBJD i OS command		

The command Document File (YDOCF) can be used to obtain layouts of the formats of the above files.

LSTOPT Parameter

In many of the commands that manipulate lists, either an existing list can be added to, or a new list can be created. The list update option (LSTOPT) parameter denotes which. It may have the following values:

- REPLACE: A new list is to be created, replacing any existing list of the same name in the specified library.
- ADD: If a list of the same name and type already exists in the specified library it will be added to. If the list does not exist an error message is sent. Entries that already exist in the target list are not added or replaced.

The LSTOPT parameter can be used to build up lists from a number of sources, and to merge lists. For instance, the following two commands would result in a list of all programs in QGPL whose names begin with either AB or CD:

YBLDOBJLST OBJ (QGPL/AB) OBJTYPE (PGM) LSTOPT (REPLACE)

YBLDOBJLST OBJ (QGPL/CD) OBJTYPE (PGM) LSTOPT (ADD)

OUTLST Parameter

The OUTLST parameter allows the creation of a second list from a primary list. On such commands you will specify the name of the list to receive the output using an OUTLST parameter.

The OUTLSTOPT parameter specifies the list update action for the output list, that is, whether an existing list should be added to, or a new list should be created. It takes the same values as the list update option (LSTOPT) parameter: REPLACE or ADD. ADD will not add or replace existing entries in the target list.

Commands with OUTLST parameters include the following:

CA 2E Command	Output List Parameter	Rqd/Opt	Default Value	Output List Option
YFLTOBJLSTYFLT MBRLST	OUTLST OUTLST	Opt Opt	*NONE *NONE *NONE *NONE	OUTLSTOPT OUTLSTOPT
YCHKLSTE YSCNSRC	OUTLST OUTLST	Opt Opt	*NONE *NONE	OUTLSTOPT OUTLSTOPT
YOPRLST YOPYLST	TOLST TOLST	Opt Rqd	*LSTA *FROMLST	LSTOPT LSTOPT
YCVTOBJLST YCVTDNFLST	MBRLST MBRLST	Rqd Rqd	TEMPLST TEMPLST	LSTOPT LSTOPT

FLAGVAL, OUTFLAGVAL, and UPDLST Parameters

The FLAGVAL parameter controls which items in a list are processed. When a list command is executed with the FLAGVAL parameter specified, only those items with the specified value in the flag field are processed. The others are ignored.

The FLAGVAL parameter is a simple list made up of two terms:

- Relational operator: (EQ/NE) To specify whether the associated flag value is or is not to be selected.

- Flag value: Value upon which to select

A single value of ANY is allowed for the FLAGVAL parameter.

Examples:

- Select all items with a flag value other than F:
YFLT0BJLST FLAGVAL (NE FAIL)
- Process all items with a flag value of U:
YEXC0BJLST FLAGVAL (EQ 'U')
- Process all items:
YEXC0BJLST FLAGVAL (ANY)

The following table lists special flag values with equivalent flag values.

Special Value	Flag Storage Value
BLANK	"
ON	'Y'
NULL	'0', '1'
FAILOBJ	'O'
FAILMBR	'M'
FAIL	'F'
SUBMITTED	'S'
PRESENT	neither '1' nor '0'

Note: Values are stored in the century byte field as set by i OS DSPxxx commands with OUTFILE specified; therefore both 0 and 1 are valid values for *NULL. When NULL is specified as FLAGVAL, items with either 0 or 1 are selected.

OUTFLAGVAL Parameter

The OUTFLAGVAL parameter is used to specify a flag value to be given to the items in a list to indicate the result of processing the item. The UPDLST parameter controls the flagged items. The OUTFLAGVAL may be either a single character variable, or one of the values shown in the table above.

Examples:

- Set flag on all missing items to F (FAIL)

YCHKLSTE OUTFLAGVAL (FAIL) UPDLST (FLAGERR)

- Set flag on all items in an object list to Y (ON)

YFLT0BJLST OUTFLAGVAL (ON) UPDLST (FLAGOK)

Some commands allow the entry of more than one value for the UPDLST parameter.

UPDLST Parameter

If errors occur when processing one or more items in a list, you may want to re-process just the items for which errors occurred, or continue processing just those items for which errors did not occur. The UPDLST parameter provides a convenient way of achieving this: the parameter can be used to delete from the list the items that have been processed successfully. Alternatively it can be used to delete from the list the items for which errors occurred. The UPDLST parameter may have one the following values:

- NONE: Do not change list items for which errors occurred.
- FLAGERR: Flag only those list items for which errors occurred.
- FLAGOK: Flag only those list items for which no errors occurred.
- RMVERR: Remove list items for which errors occurred.
- RMVOK: Remove list items for which no errors occurred.

The flag value used if UPDLST (FLAGERR) or UPDLST (FLAGOK) is specified is determined by the OUTFLAGVAL parameter.

The following table shows the effect of the UPDLST values on the entries in a list:

Value	PASS	FAIL
NONE	Leave No Chg	Leave No Chg
RMVERR RMVOK	Leave Remove	Remove Leave
FLAGERR FLAGOK	No Chg Flag	Flag No Chg

With commands that support both input and output lists, the action specified by the UPDLST parameter is always applied to the output list.

BUILD and EDIT Parameters

Using a list will often be a several step process:

6. Build a list.
7. Edit the list.
8. Filter the list.
9. Execute the list.

Many list commands contain a facility to invoke more than one list processing function at a time, without having to re-enter the list name. The linking is done using the BUILD, EDIT, and FILTER parameters. For instance:

Using the Command Edit Object List (YEDTOBJLST) you could specify a value of BUILD (YES), you will then be prompted for the selection criteria that you wish to use to build the list. The list will be built and immediately presented for editing. Thus:

```
YEDTOBJLST OBJLST (QGPL/FRED) BUILD (YES)
```

is equivalent to:

```
YBLDOBJLST OBJLST (QGPL/FRED) /* build */  
YEDTOBJLST OBJLST (QGPL/FRED) /* edit */
```

Using the command Create Objects (YCRTOBJ) you could specify a value of EDIT (YES), you will then be given the opportunity to edit the list of members before they are submitted for compilation. Thus:

```
YCRTOBJ OBJLIB (QGPL) SRCFILE (QGPL/QCLSRC) OBJLST (QGPL/FRED) EDIT (YES)
```

is equivalent to:

```
YBLDMBRLST SRCFILE (QGPL/QCLSRC) MBRLST (QGPL/FRED) /*build*/  
YEDTMBRLST MBRLST (QGPL/FRED) /*edit */  
YCRTOBJ OBJLIB (QGPL) SRCFILE(MBRLIST) MBRLST(QGPL/FRED) /* use */
```

RQSDTA, USROPT, OPTFIL and OPTMBR Parameters

The commands Execute List YEXCOBJLST, YEXCMBRLST, and YEXCDBFLST invoke a specified request string upon a list of items. The request string is specified with the RQSDTA parameter, in one of two ways:

The request string may be specified explicitly:

```
YEXCOBJLST RQSDTA ('CHGPGM &L/&O')
```

Reference may be made to a predefined request string stored in a user-defined option file in which case a value of *USROPT should be specified for the RQSDTA parameter, and the actual request string indicated using the USROPT, OPTFIL and OPTMBR parameters.

```
YEXCOBJLST RQSDTA (USROPT) USROPT ('A1') OPTFIL(MYOPT)
```

Alternatively, the contents of a user-defined option can be substituted into the request string by means of the substitution symbol @C.

User Option QAUOOPT File

The user-defined option file should have the same format as the i OS QAUOOPT user-defined option file. New options can conveniently be added using the i OS PDM utility (STRPDM).

Substitution Symbols

The request data is executed for each item in the list. You can specify where in the request string you wish to have the item names, and other properties of the list entries, inserted by using the following substitution symbols:

		YEXCOBJLST	YEXCDBFLST	YEXCMBRLST
&A	Item attribute	Object attr	File attr	SEU type
&B	List type	Y	Y	Y
&C	User option	Y	Y	Y
&D	Last change	Object date	File date	Member date
&F	date	-	File name	File name
&O	File name	Y	-	-
&J	Object name	Y	Y	Y
&L	Job description	Object lib	File lib	File lib
&M	name	-	-	Member name
&N	Library name	Object name	File name	Member name
&S	Member name	Y	-	-
&T	List entry	Y	-	-
&X	name	Object text	File text	Member text
	Short type (OBJ)			
	Full type (*OBJ)			
	Object text			

The following alternative special substitution values are also supported. They can be used when invoking the command prompter upon your request string (the command prompter does not allow names beginning with &).

```
YEXCOBJLST RQSDTA ('?DSPOBJD OBJ(@L/@0) OBJTYPE(@T)')
```

		YEXCOBJLST	YEXCDBFLST	YEXCMBRLST
@@O	Object name	Object name	-	-
@F	File name	-	File	File
@M	Member name	-	-	Member
@L	Library name	Object lib.	File lib.	File lib.
@T	Object name	Object type		
@C	User option	Y	Y	Y

Internationalization of Special Substitution Symbols

If the @ character is not available on your keyboard you may either key the equivalent hex value (X'7C'), or else specify an alternative character with the YPEXCHA data area. For example, to specify # as an alternative substitution value (that is. '#O', '#L' '#T')

```
CHGDTAARA DTAARA (YPEXCHA) VALUE ('#')
```

MENU Parameter

The MENU parameter identifies a menu held in a menu file. Menu names must always be valid system names, that is: be up to ten characters long, begin with a letter, (or @ or #), and contain no embedded blanks. Menu names must be unique within a member. In many commands a special value of SELECT is allowed for the menu name. Specifying this value gives you a selection display of available menus.

```
YWRKMNU MENU (SELECT)
```

Menus must be stored in one or more menu files: a menu file is a database file having the same format as the file YDSNMNU in the utility library. The command Create Design File (YCRTDSNF) should be used to create any additional copies of the menu file required, as it will create a file of the correct format. A default copy of the menu, YDSNMNU is shipped with CA 2E.

Referring to Menus

Menus can be referred to simply by name, in which case the default menu file YDSNMNU is used:

YGO MENU (FRED)

Alternatively references to menus can be qualified by file name, in which case the override file name specified by the FILE parameter is used:

YGO MENU (FRED) FILE (MYMENUS)

PANEL Parameter

The PANEL parameter identifies a panel design held in a design file. Panel names must always be valid system names, that is, be up to ten characters long, begin with a letter, (or @ or #), and contain no embedded blanks. Panel names must be unique within a member. In many commands a special value of '*SELECT' is allowed for the panel name specifying this value will give you a selection display of available panel designs.

YWRKPNL PANEL (*SELECT)

Panel design files must have a specific format. The command Create Design File (YCRTDSNF) should be used to create any additional copies of the panel file that you require; it will create a file of the correct format, as well as the necessary logical view.

A default copy of the panel design file, YDSNPNL is shipped.

Referring to Panel Designs

Panel designs can be referred to simply by name, in which case the default panel design file YDSNPNL is used:

YDSPNL PANEL (FRED)

Alternatively, references to panel designs may be qualified by file name, in which case the override file name specified by the FILE parameter is used:

YDSPNL PANEL (FRED) FILE (MYPANELS)

REPORT Parameter

The REPORT parameter identifies a report design held in a CA 2E report design file. Report names must always be valid system names, that is, be up to ten characters long, begin with a letter, (or @ or #), and contain no embedded blanks. Report names must be unique within a member.

In many commands a special value of *SELECT is allowed for the report name: specifying this value will give you a selection display of available report designs.

Report design files must have a specific format (in fact the report design file is a logical file spanning three different physical files). The command Create Design File command (YCRTDSNF) should be used to create any additional copies of the report file that you require; it creates a file of the correct format, as well as the necessary logical views.

Referring to Report Designs

Report designs can be referred to simply by name, in which case the default report design file YDSNRPT is used:

```
YWRKRPT REPORT (FRED)
```

Alternatively, references to report designs may be qualified by file name, in which case the override file name specified by the FILE parameter is used:

```
YWRKRPT REPORT (FRED) FILE (MYREPORTS)
```

CHGPRTDFT Parameter

The change print default parameter (CHGPRTDFT) allows you to change the print file attributes of the output files generated by certain of the documentation commands.

One of three values may be specified for the parameter:

- SAME: The print file attributes are not to be changed.
- TEMP: The print file attributes are to be changed just for the current execution of the command.
- PERM: The print file attributes are to be changed permanently.

Values of TEMP and PERM are only valid in an interactive environment. If TEMP or PERM is specified, you will be prompted to supply override attributes. Refer to the i OS commands Override Print File (OVRPRTF) and Change Print File (CHGPRTF) for details on print file attribute values.

Appendix B: Required Application Objects

This appendix describes how to obtain a list of objects required to display menus and help text.

Obtain Required Object Lists

You can obtain a list of objects required to display menus and help text by doing the following from an i OS command line or CL program:

```
YBLDJOBLS Y1USR0BJ FILE OBJLST (Y1USR0BJ) CPYF FROM FILE(Y1USR0BJ)
TOFILE(QTEMP/YOBJLST)
FROMMBR(*ALL) TOMBR(*FROMMBR) MBROPT(*AD) FMTOPT(*MAP)
YBLDMBRLST QTEMP/YOBJLST
YFLTMBRLST MBR (Y1)
YEXCMBRLST RQDTA ('YD0C0BJLST QTEMP/@N)
```

This obtains files listing the objects and a set of object lists used in conjunction with the Create Duplicate Objects command to create the objects.

If you are running a generated application, you can use the YDUPAPPOBJ command to duplicate objects required for menus and help.

Appendix C: Long Running Commands

This appendix names the long running commands used with CA 2E.

Long running commands are listed in the following table. Be sure to use these commands with caution because they may require a large amount of system resources to run. A severity (1 is high) is shown against each command.

Command	Severity
YCHGCMD	3
YCHGOBJOWN	3
YCMPSRC	2
YCPYF	2
YCRTDUPOBJ	2
YCRTOBJ	3
YCVTDBR	2
YCVTPGMREF	2
YCVTUSRPRF	1
YDOCAUT	2
YDOCEXCREF	1
YDOCF	2
YDOCFLDREF	2
YDOCMSGREF	2
YDOCPGM	1
YDOCPGMREF	2
YDSPDBR	2
YDSPMBRLST	2
YMOVM	2
YMOVOBJ	3
YMOVOBJSRC	2
YRTVMSGF	1
YRTVPNLDSN	1
YRTVRPTDSN	1
YSCNSRC	2
YSCNRPLSRC	2
YWRKF	3

Appendix D: Monitored Error Messages

This appendix lists some of the escape messages which may be generated by the commands. Where possible, make use of the appropriate existing standard i OS messages.

The messages are listed with command; the commands are listed in alphabetical order. The following information is given for each message if applicable:

- n Message identifier
- n Message type (E – Escape)
- n Message text

All execution messages definitions are in a single message file: YYYYMSG in the utility library.

Commands Using a Database File List

Commands using a database file list generate the following messages:

YDL0001 E Database file list &1 not found

YYY0005 E SELECT option only valid for interactive jobs

YYY0103 E List &1 in &2 does not contain any entries.

Commands Using a Member List

Commands using a member list generate the following messages:

YML0001 E Member list &1 not found

YYY0005 E SELECT option only valid for interactive jobs

YYY0103 E List &1 in &2 does not contain any entries

Commands Using an CA 2E Object List

Commands using an CA 2E object list generate the following messages:

YOL0001 E Object list &1 not found

YOL0021 E IF OBJLST specified for OBJ, OBJTYPE must be ALL.

YYY0005 E SELECT option only valid for interactive jobs

YYY0103 E List &1 in &2 does not contain any entries

YADDDSNFM

Messages generated by YADDDSNFM include the following:

YDS0010 E Member required when option is ADDMBR

YDS0012 E Text required when option is ADDMBR

YBLDLIBLST

YBLDLIBLST generates the following message:

YLL0011 E IF job description name specified for JOB, LIBL must be JOB

YWRKF

YWRKF generates the following message:

YPR0034 E File PRV not established

YCHGOBJOWN

YCHGOBJOWN generates the following message:

YPG0001 E Ownership of &2 objects changed to &1. &3 not changed

YCHKLSTE

YCHKLSTE generates the following message:

YYY0103 E List &1 in &2 does not contain any entries.

YCHKLIBLST (And All CA 2E Library List Commands)

Messages generated by YCHKLIBLST (and all CA 2E Library List commands) include the following:

YLL0001 E Library list &1 in &2 not found

YLL0003 E Library list file &1 in &2 not found

YLL0007 E No selection made - command cancelled by user

YLL0014 E Library list &1 in &2 already exists

YLL0027 E SELECT not allowed unless interactive job

YCHKMNU (And All CA 2E Menu Commands)

Messages generated by YYCHKMNU (and all CA 2E Menu commands) include the following:

YMN0011 E Specified menu not found

YMN0012 E Specified menu already exists

YCHKPNL (And All CA 2E Panel Design Commands)

Messages generated by YCHKPNL (and all CA 2E Panel Design commands) include the following:

YDS0002 E Specified file is not a Panel file

YDS0004 E Specified Panel not found

YDS0005 E Specified Panel already exists

YCHKRPT (And All CA 2E Report Design Commands)

Messages generated by YCHKRPT (and all CA 2E report design commands) include the following:

YDS0103 E Specified file is not a report file

YDS0104 E Specified report not found

YDS0105 E Specified report already exists

YCPYLIBLST

Messages generated by YCPYLIBLST include the following:

YLL0001 E Library list &1 in &2 not found

YLL0014 E Library list &1 in &2 already exists

YLL0031 E FROMLIBLST must not be the same as TOLIBLST

YCPYF

Messages generated by YCPYF include the following:

YPG0051 E If FROMLIB is same as TOLIB, FROMPFX must not equal TOPFX

YPG0052 E &2 files copied to &1 &3 not copied.

YPG0056 E LST cannot be specified for both FROMLIB and TOLIB.

YPG0057 E IF FROMLIB (LIBL) specified, TOLIB (FROMLIB) is not allowed.

YPR0031 E QTEMP not allowed for TOLIB.

YCRTDSNF

Messages generated by YCRTDSNF include the following:

YDS0013 E Library required when option is CREATE.

YDS0018 E LIBL not allowed if file or member name left blank.

YCRTDUPOBJ

Messages generated by YCRTDUPOBJ include the following:

YLL0023 E If FROMLIB (OBJLST) specified, OBJ (FROMLST) must also be specified.

YPR0051 E YCRTDUPOBJ command cannot duplicate to library & 1.

YPR0055 E & 1 errors occurred when duplicating objects to & 2.

YPR0057 E Errors occurred while duplicating & 1 in & 2 to library & 3.

YCRTOBJ

YCRTOBJ generates the following message:

YYY7405 E & 2 errors occurred on compilation of group & 3.

YDLTOBJ

YDLTOBJ generates the following message:

YPG0053 E & 1 objects deleted. & 2 objects not deleted.

YEXCDBFLST

Messages generated by YEXCDBFLST include the following:

YDL0008 C Database file list executed, & 1 errors found.

YDL0009 E ERRLVL exceeded on YEXCDBFLST for list & 1 in & 2

YEXCMBRLST

Messages generated by YEXCMBRLST include the following:

YML0008 C Member list executed, & 1 errors found.

YML0009 E ERRLVL exceeded on YEXCMBRLST for list & 1 in & 2

YEXCOBJLST

Messages generated by YEXCOBJLST include the following:

YOL0008 C Object list executed, &1 errors found.

YOL0009 E ERRLVL exceeded on YEXCOBJLST for list &1 in &2

YMOVBJ

Messages generated by YMOVBJ include the following:

YLB0002 E Archive library must not be the same as To/from library.

YLB0003 E NONE may not be specified for both TOLIBOBJ and TOLIBDTA.

YPR0011 E &1 moved=&2, added=&3, replaced = &4, failed moves = &5.

YMOVM

Messages generated by YMOVM include the following:

YLB0001 E From and To libraries must not be the same

YLB0002 E Archive library must not be the same as To/from library

YPR0011 E &1 moved=&2, added=&3, replaced=&4, failed moves=&5

YMOVBJSRC (May also send YMOVBJ and YMOVM messages)

YPR0005 E TOLIB required when referenced from other parameters

YPR0011 E &1 moved=&2, added=&3, replaced=&4, failed moves=&5

YPR0025 Errors occurred on moving objects and or source

YxxxLLE

Messages generated by YxxxLLE include the following:

YLL0003 E Library list file &1 in &2 not found

YYY1215 E &6 errors occurred when updating lists. &5 lists updated, &7 lists ignored

Appendix E: Command and Keyword Abbreviations

This appendix contains an alphabetic list of all the abbreviations used in the AllFusion 2E command names and their parameter keywords, that are additional to the i OS standards. For the i OS standards see Appendix F in the *IBM i OS Programmer's Handbook*.

Command and keyword abbreviations include the following:

- ABR - Abbreviated
- BLD - Build
- FLT - Filter
- INX - Index
- LL - Library list
- SCN - Scan
- TDY - Tidy

Index

*

*D Debug source directives, 3-193

A

abbreviated command names, 1-4

abbreviations, E-1

Add Design File Member (YADDDSNFM), 2-1

Add Help Vector Table (YADDHLPTBL), 2-2

Add Library List Entry (YADDLLE), 2-6

Add Member List (YADDMLE), 2-9

Add Object List Entry (YADDOLE), 2-12

Add Source Member (YADDSRCM), 2-14

AllFusion 2E
modules, 1-3
object names, 1-4

Apply Data Changes (YAPYY1DCHG), 2-15

B

Build Database File List (YBLDDBFLST), 2-17

Build Format List (YBLDFMTLST), 2-21

Build Library List (YBLDLIBLST), 2-23

Build Master Document (YBLDDOC), 2-19

Build Member List (YBLDMBRLST), 2-27

Build Object List (YBLDOBJLST), 2-29

C

calling AllFusion 2E commands, 1-3

Change Command (YCHGCMD), 2-31

Change Job Description Library List
(YCHGJOBDDL), 2-37

Change Library List (YCHGLIBL), 2-38

Change Library List Description (YCHGLIBLST),
2-40

Change List (YCHGLST), 2-42, 2-46

Change Object Ownership (YCHGOBJOWN), 2-
46

Change User Profile (YCHGUSRPRF), 2-49

Check IFS Object (YCHKIFSOBJ), 2-63

Check Library List (YCHKLIBLST), 2-66

Check List Entries (YCHKLSTE), 2-68

Check Menu (YCHKMNU), 2-75

Check New Password (YCHKPWDVAL), 2-80

Check Panel Design (YCHKPNL), 2-78

Check Report Design (YCHKRPT), 2-81

Check System Name Is Valid (YCHKVN), 2-83

Compare Source (YCMPSRC), 2-84

Convert Authorization List (YCVTAUTL), 2-143

Convert Binary to Decimal (YCVTBIN), 2-145

Convert Database Relations (YCVTDBR), 2-149

Convert DBF List to Member List
(YCVTDBFLST), 2-147

Convert DDS Command Keys (YCVTDDSCKY),
2-151

Convert DDS to IGC (YCVTDDSIGC), 2-156

Convert Decimal To Binary (YCVTDEC), 2-162

Convert Object List to Member List
(YCVTOBJLST), 2-164

Convert Print Output To Source Member
(YCVTPRT), 2-169

Convert Program References (YCVTPGMREF),
2-166

Convert User Profile (YCVTUSRPRF), 2-182
Copy Files (YCPYF), 2-89
Copy Library List (YCPYLIBLST), 2-93
Copy List (YCPYLST), 2-95
Copy Menu (YCPYMNU), 2-96
Copy Message Description (YCPYMSGD), 2-98
Copy Panel Design (YCPYPNL), 2-100
Copy Report Design (YCPYRPT), 2-103
Copy User Profile (YCPYUSRPRF), 2-105
Create DDS From Panel Design (YCRTPNLDDS), 2-123
Create DDS from Report Design (YCRTRPTDDS), 2-126
Create Design File (YCRTDSNF), 2-108
Create Duplicate Objects (YCRTDUPOBJ), 2-110
Create Objects (YCRTOBJ), 2-114
Create Source Files (YCRTSRCPF), 2-129
Create User Profile (YCRTUSRPRF), 2-132

D

data object commands, 1-3
QWHFDPHY format, A-3
default list names, A-2
Define Panel Design (YDFNPNLDSN), 2-184
Delete Library List (YDLTLIBLST), 2-186
Delete Library List (YDLTLST), 2-187
Delete Menu (YDLTMNU), 2-188
Delete Object (YDLTOBJ), 2-189
Delete Panel Design (YDLTPNL), 2-191
Delete Report Design (YDLTRPT), 2-192
Delete User Profile (YDLTUSRPRF), 2-193
Design default data areas (YEDTDSSNDFT), 3-45

Display a Program's Message Queue (YDSPPGMQ), 3-38
Display Abbreviated (YDSPABR), 3-20
Display Expiry Date (YDSPEXPDAT), 3-24
Display Help Text (YDSPHLP), 3-25
Display Library List (YDSPLIBLST), 3-34
Display Member (YDSPMBRLST), 3-35
Display Panel Design (YDSPPNL), 3-39
Display User Profile (YDSPUSRPRF), 3-41
Document DBF List (YDOCDBFLST), 2-197
Document Execution References (YDOCEXCREF), 2-198
Document Field References (YDOCFLDREF), 2-207
Document File (YDOCF), 2-202
Document Format List (YDOCFMTLST), 2-209
Document Library (YDOCLIBLST), 2-210
Document Member List (YDOCMBRLST), 2-211
Document Menu (YDOCMNU), 2-212
Document Menu References (YDOCMNUREF), 2-214
Document Message References (YDOCMMSGREF), 2-215
Document Object Authorities (YDOCAUT), 2-195
Document Object List (YDOCOBJLST), 3-1
Document Panel Design (YDOCPNL), 3-10
Document Program (YDOCPGM), 3-3
Document Program References (YDOCPGMREF), 3-6
Document Report Design (YDOCRPT), 3-13
Document Source File Members (YDOCSRC), 3-16
Document User Profile (YDOCUSRPRF), 3-19
documentation, 1-1
 presentation standards, 1-2

DSPFD OS/400 command, A-3
DSPOBJD OS/400 command, A-3

E

Edit Command Key Table (YEDTCKYTBL), 3-42
Edit Data Area (YEDTDTAARA), 3-46
Edit DBF List (YEDTDBFLST), 3-43
Edit Format List (YEDTFMTLST), 3-48
Edit Group Data Area (YEDTGDA), 3-49
Edit Library List (YEDTLIBLST), 3-51
Edit Local Data Area (YEDTLDA), 3-50
Edit Member List (YEDTMBRLST), 3-52
Edit Message Description (YEDTMSGD), 3-53
Edit Object List (YEDTOBJLST), 3-54
Edit Password Values (YEDTPWDVAL), 3-55
End Spooled File Router (YENDSPLRTR), 3-56
Execute CL Member (YEXCCL), 3-57
Execute Database File List (YEXCDBFLST), 3-58
Execute Member List (YEXCMBRLST), 3-66
Execute Object List (YEXCOBJLST), 3-71

F

Filter Database List (YFLTDBFLST), 3-77
Filter Format List (YFLTFMTLST), 3-82
Filter Member List (YFLTMBRLST), 3-87
Filter Object List (YFLTOBJLST), 3-95

G

Go to Menu (YGO), 3-106
Grant Product Authority (YGRTPRDAUT), 3-110

I

Index List (YINXLST), 3-111
Initial programs, 2-62

L

long running commands, C-1
LSTOPT parameter, A-3

M

QWHFDML format, A-3
Move List (YMOVLST), 3-114
Move Member (YMOVM), 3-115
Move Objects & Source (YMOVOBJSRC), 3-124
Move Objects (YMOV OBJ), 3-119
Move User Data Objects (YMVY1DTA), 3-131

O

Operate on List (YOPRLST), 3-132

P

parameters
 BUILD and EDIT, A-7
 CHGPRTDFT, A-12
 FLAGVAL, OUTFLAGVAL, and UPDLST, A-5
 LST and LSTTYPE, A-1
 LSTOPT, A-3
 MENU, A-10
 OBJLST, MBRLST, FMTLST and DBFLST, A-2
 OUTFLAGVAL, A-6
 OUTLSOUTLST and OUTLSTOPT, A-4
 PANEL, A-10
 REPORT, A-11
 RQSDTA, USROPT, OPTFIL and OPTMBR, A-8

UPDLST, A-6

R

referring to menus, A-10
referring to Panel Designs, A-11
referring to Report Designs, A-11
Remove Library List Entry (YRMVLLE), 3-136
Remove Member (YRMVM), 3-138
Rename an Object and Source (YRNMBSRC), 3-147
Rename Library (YRNMLIB), 3-140
Rename Library List (YRNMLIBLST), 3-142
Rename Library List Entry (YRNMLLE), 3-143
Rename Menu (YRNMMNU), 3-146
Rename Panel Design (YRNMPNL), 3-148
Rename Report Design (YRNMRRPT), 3-150
Rename User Profile (YRNMUSRPRF), 3-151
required application objects, B-1
Restart Spooled File Router (YRSTSPLRTR), 3-152
Retrieve Message File (YRTVMSGF), 3-154
Retrieve Object Library (YRTVOBJLIB), 3-159
Retrieve Panel Design (YRTVPNLDSN), 3-161
Retrieve Report Design (YRTVRPTDSN), 3-164
Retrieve Spooled File Attributes (YRTVSPLFA), 3-165
Retrieve User Profile (YRTVUSRPF), 3-167

S

Scan Source Member (YSCNSRC), 3-180
Scan/Replace Source Member (YSCNRPLSRC), 3-175
SELECT Value for a List Name, A-3

Set Break Program (YSETBRKPGM), 3-188

Start Convert Print Key Output Mode (YSTRCVTPRT), 3-190

Start Debug and Add Auto Breakpoints (YSTRDBG), 3-192

Start spooled file router (YSTRSPLRTR), 3-195

substitution symbols, A-8
internationalization of, A-9

T

Tidy RPG III Source (YTDYRPFSCR), 3-200

Translate Physical File Data (YTRNPF), 3-202

Translate Source File Data (YTRNSRCF), 3-204

U

User Option QAUOOPT File, A-8

Exception message queue, 2-61

W

work lists, A-2

Work with Database File Data (YWRKF), 3-208

Work with Library List (YWRKLIBLST), 3-212

Work with Menus (YWRKMNU), 3-214

Work with Panel Design (YWRKPNLU), 3-215

Work with Report Design (YWRKRPT), 3-217

Work with Spooled File Routing Entries (YWRKSPLRTE), 3-218

Work with User Profile (YWRUSRPRF), 3-218

Y

YINLPGM program, 2-62

QRPLOBJ OS/400 library, 2-118

YDBFLST file, A-3
YDSNMNU file, 2-110
YINLPGMPWD, 2-62

