

Index

+ (addition operator), 156, 157
 & (ampersand), 37
 * (asterisk), 45
 - (continuation character), 79, 81
 + (continuation character)
 blanks preceding, 80
 for comments, 81
 defined, 79
 example use, 79–80
 SEU prompter placement, 81
 / (division operator), 156, 157
 > (greater than) symbol, 39
 < (less than) symbol, 40
 * (multiplication operator), 156, 157
 + (plus) symbol, 38–39
 / (qualified character), 20, 22
 ? (question mark), 37
 ?? (selective prompting character), 344
 ?- (selective prompting character), 344, 345
 ?* (selective prompting character), 344, 345
 - (subtraction operator), 156, 157

A

Activation groups, 393–396
 default, 394
 defined, 393
 file overrides and, 395–396
 options, 393
 option use, 395
 principles, 395
 scoping, 396
 system-named, 394
 user-named, 394
 See also Integrated Language Environment (ILE)
 Additional parameters function key (F10), 35
 ADDLIB (Add Library List Entry) command, 175, 201
 ADDPFM (Add Physical File Member) command, 10, 42, 250
 %ADDR function. *See* %ADDRESS function
 %ADDRESS function, 120, 123, 124
 Allocation/storage system values, 247
 All parameters function key (F9), 33
 Ampersand (&), 37
 *AND operator, 163, 164
 Application Development ToolSet (ADTS), 264
 Application programming interfaces (APIs), 279
 bindable, 359
 CEE, 359
 CEEDATE, 359–360
 CEEDAYS, 359
 IBM-supplied, 351–361
 ILE bindable, 359–360
 introduction to, 351–352

 list of, 351
 QCLSCAN, 358, 361
 QCMD, 352–354, 361
 QCMDCHK, 355–356, 361
 QCMDEXC, 356–358, 361, 431
 QDCXLATE, 358, 361
 QRCVDTAQ, 358–359
 QSNDDTAQ, 358–359
 QUSCMDLN, 354, 361
 summary, 361
 Arguments, 215
 Arithmetic expressions, 156–158
 operands, 156
 operators, 156
 summary, 166
 using, 156–158
 in VALUE specification, 118
 within COND parameter, 156
 within MAPFLD parameter, 156
 within VALUE parameter, 156
 See also Expressions
 Arithmetic operators, 156
 Asterisk (*), 45
 Attributes
 changing, 239–241
 display, 266
 LOG, 379–380
 LOGCLPGM, 380–381
 network, 248
 retrieving, 238
 system values, 241–247

B

Backups, 49
 Based variables
 declaring, 100–101
 defined, 100
 See also Program variables
 Batch jobs
 defined, 173
 submitting, 182
 See also Jobs
 *BCAT operator, 119, 158
 defined, 119
 symbol, 159
 %BIN function. *See* %BINARY function
 %BINARY function, 121
 length, 158
 as operand, 157
 in receiver variable, 123
 using, 157–158
 Bindable APIs, 359
 Bind-by-copy, 391, 392
 Bind-by-reference, 392
 Binding programs, 68–69
 Blank lines, 78
 Built-in functions
 %ADDRESS (%ADDR) function, 120, 123, 124

 %BINARY (%BIN) function, 121, 123
 %OFFSET (%OFS) function, 120, 124, 434
 with pointers, 123–124
 in receiver variable, 121–123
 %SUBSTRING (%SST) function, 120–121, 123
 %SWITCH, 120

C

CALL (Call Program) command, 49, 71, 175–179, 187
 called program, 176
 call stack, 178–179
 ILE considerations, 177
 parameters, 176
 in passing variables as parameters, 221
 PGM parameter, 175–176
 Called programs
 CALL command, 176
 defining parameters in, 216–217
 passing constants to, 217–221
 passing variables to, 221–222
 *CALLER activation group option, 395
 CALLPRC (Call Bound Procedure) command, 177
 for bindable APIs, 359
 defined, 392
 RTNVAL parameter, 392
 Call stack, 178–179
 defined, 178
 invocation levels, 178–179
 invocation levels (TFRCTL command), 180–181
 level scoping, 396
 LIFO, 179
 Call stack message queue
 defined, 290
 OPM programs, 299
 See also Message queues
 CALLSUBR (Call Subroutine) command, 146
 coding, 146
 defined, 146
 example use, 146
 form, 146
 RTNVAL parameter, 147
 *CAT operator, 119, 158
 defined, 119
 symbol, 159
 CEEDATE (Convert Lilian Date to Character Format) API, 359–360
 calling, 359–360
 defined, 359
 parameters, 359
 CEEDAYS (Convert Date to Lilian Format) API, 359
 Character constants
 defined, 217
 passing, 217–220
 sending as parameters, 219
 See also Constants
 Character string expressions, 158–160
 defined, 158

- operators, 158
- in parentheses, 158
- with SNDPGMMSG command, 159
- %SST function, 159–160
- summary, 166
- symbols, 159
- See also* Expressions
- Character type (*CHAR) variables
 - assigning values to, 97–98
 - CHGVAR with, 114
 - converting numeric data to, 116–117
 - converting to numeric data, 117
 - default initial value, 96
 - defined, 94
 - length, 218
 - length specification, 95
 - parameter definition, 216
 - %SUBSTRING function with, 121
- Character values, 19
- CHGCMD (Change Command) command, 370–371, 374
 - defined, 370
 - properties, 371
- CHGCMDDFT (Change Command Default) command, 372, 374
- CHGCURDIR (Change Current Directory) command, 11
- CHGCURLIB (Change Current Library) command, 175, 246
- CHGDTAARA (Change Data Area) command, 276, 281
- CHGJOB (Change Job) command, 239–241
 - defined, 239
 - form, 239
 - running, 240
 - *SAME default, 448
- CHGJOB (Change Job Description) command, 379
- CHGLIBL (Change Library List) command, 175
- CHGPFM (Change Physical File Member) command, 10
- CHGSYSLIBL (Change System Library List) command, 175
- CHGSYSVAL (Change System Value) command, 242
- CHGUSRPRF (Change User Profile) command, 248, 434
- CHGVAR (Change Variable) command, 80, 113–124
 - built-in functions and, 120–124
 - character length, 114
 - with character variables, 114
 - for data conversion, 114, 116–117, 266
 - defined, 113
 - with logical variables, 116
 - with numeric variables, 115
 - in procedure section, 113
 - receiver variable, 115
 - relational expressions with, 162
 - summary, 124
 - uses, 113–114
- VALUE parameter, 113, 115
- VALUE specification, arithmetic expression for, 118
- VALUE specification, character string expression for, 119–120
- VALUE specification, variable for, 118
- VAR parameter, 113
- CHKOBJ (Check Object) command, 196–197, 201
 - AUT parameter, 197
 - defined, 196
 - error messages, 203
 - file member existence test, 196–197
 - MBR parameter, 197
 - object authority test, 197
 - object existence test, 196
 - OBJTYPE parameter, 196
 - to test for file existence, 384
 - Choices function key (F11), 34
- CL
 - controlling workflow with, 3
 - control structures, 133–148
 - defined, 1–2
 - execution environment and restrictions, 3
 - file existence requirements, 259–260
 - file processing, 267–271
 - functions, 2
 - HLL in combination with, 2
 - limitations, 2
 - Original Program Model (OPM) and, 70
 - source, compiling, 56–67
 - SQL and, 259
 - structured programming and, 133–134
 - summary, 4
 - use of, 4
 - what it's not, 2
- CL compiler
 - defined, 56
 - options, 58–60
 - reports, 60–63
 - text description, 59
 - using, 56–57
- CLOF (Close File) command, 327
- CL programs
 - binding, 68–69
 - capabilities, 49–50
 - comments, entering, 79
 - as compiled program objects, 3
 - conditional logic, 3
 - creating, 51–70
 - creation steps, 71
 - defined, 2
 - example, 3–4
 - executing with CALL command, 175–179
 - executing with TFRCTL command, 179–181
 - execution order, altering, 3
 - file use in, 319
 - logic, 139
 - objects, 60
 - OPM, 177
- purposes, 1
 - for repetitive processes, 2
 - self-submitting, 185–186
 - structure, 77–89
 - summary, 71
- CMDADD menu, 45
- CMDFILE menu, 44, 45
- CMD statement, 406–407
 - ALLOW parameter, 406
 - ALWLMTUSR parameter, 407
 - CURLIB parameter, 407
 - defined, 406
 - MAXPOS parameter, 407
 - PRDLIB parameter, 407
 - PROMPT parameter, 406
 - TEXT parameter, 406
 - See also* Command definition statements
- CMD-type menus, 45
- Collections, 259
- Command complete function key (F16), 34
- Command continuation characters, 79–81, 89
 - +, 79
 - , 79
 - +, 80–81
 - , 81
- Command definition object (CDO)
 - defined, 404
 - parameter description, 425
- Command definition statements, 404–413
 - CMD statement, 406–407
 - defined, 404
 - DEP statement, 412
 - ELEM statement, 412
 - EXCCMD (Execute Commands), 405
 - PARM statement, 407–410
 - PMTCTL statement, 412–413
 - QUAL statement, 410–411
 - types of, 404
- Command-level MONMSG, 198–199
 - command evaluation, 201
 - defined, 198
 - example, 198–199
 - mixed with program-level MONMSG, 201–202
- Command line operations, 27
- Command list
 - ADDLIBL (Add Library List Entry), 175, 201
 - ADDPFM (Add Physical File Member), 10, 42, 250
 - CALL (Call Program), 49, 71, 175–179
 - CALLPRC (Call Bound Procedure), 177, 359, 392
 - CALLSUBR (Call Subroutine), 146
 - CHGCMD (Change Command), 370–371, 374
 - CHGCMDDFT (Change Command Default), 372, 374
 - CHGCURDIR (Change Current Directory), 11
 - CHGCURLIB (Change Current Library), 175, 246
 - CHGDTAARA (Change Data Area), 276, 281
 - CHGJOB (Change Job), 239–241, 448
 - CHGJOB (Change Job Description), 379

- CHGLIBL (Change Library List), 175
 CHGPFM (Change Physical File Member), 10
 CHGSYSLIBL (Change System Library List), 175
 CHGSYSVAL (Change System Value), 242
 CHGUSRPRF (Change User Profile), 248, 434
 CHGVAR (Change Variable), 80, 113–124
 CHKOBJ (Check Object), 196–197, 201, 384
 CLOF (Close File), 327
 COPYRIGHT, 85
 CPYF (Copy File), 16, 104
 CPYFRMQRYP (Copy from Query File), 331–332, 333
 CRTBNDCL (Create Bound CL Program), 35–36, 69, 389
 CRTCLMOD (Create CL Module), 56, 58–60, 71, 367, 381, 388
 CRTCLPGM (Create CL Program), 3, 70
 CRTCMD (Create Command), 9, 404, 413
 CRTDSPF (Create Display File), 10, 264
 CRTDTAARA (Create Data Area), 273, 274–275
 CRTDTAQ (Create Data Queue), 278
 CRTLF (Create Logical File), 10, 263–264
 CRTLIB (Create Library), 343
 CRTMSGQ (Create Message Queue), 290
 CRTPF (Create Physical File), 10, 262
 CRTPGM (Create Program), 68, 388
 CRTPNLGRP (Create Panel Group), 445
 CRTPRTF (Create Printer File), 10, 324
 CRTSRCPF (Create Source Physical File), 71
 CRTSRVPGM (Create Service Program), 392
 CRTUSRPRF (Create User Profile), 434
 CVTDAT (Convert Date), 249–250
 DCL (Declare CL Variable), 85, 93–103
 DCLF (Declare File), 85, 103–104, 266, 267–268, 280
 DCLPRCOPT (Declare Processing Options), 85, 147–148
 DLTDATQ (Delete Data Queue), 279
 DLTF (Delete File), 1
 DLTMOD (Delete Module), 389
 DLTOVR (Delete Override), 323, 396
 DLTPGM (Delete Program), 7
 DMPCLPGM (Dump CL Program), 122, 200, 201
 DO, 66, 139
 DOFOR (Do For), 141, 142–143
 DOUNTIL (Do Until), 142
 DOWHILE (Do While), 141–142
 DSPCMD (Display Command), 17, 367
 DSPDTAARA (Display Data Area), 275
 DSPFFD (Display File Field Descriptions), 273
 DSPJOB (Display Job), 9, 377
 DSPJOBLOG (Display Job Log), 204, 377–378
 DSPLIB (Display Library), 38, 39, 341
 DSPLIBL (Display Library List), 175
 DSPLOG (Display Log), 290
 DSPMSGD (Display Message Description), 194
 DSPPFM (Display Physical File Member), 10, 20
 DSPTAP (Display Tape), 8, 13, 345
 EDTF (Edit File), 7
 EDTLIBL (Edit Library List), 175
 ELSE, 137
 ENDDO, 66, 139
 ENDJOBABN (End Job Abnormally), 245
 ENDPGM (End Program), 3, 86
 ENDRCV (End Receive), 271
 ENDSELECT, 140
 ENDSUBR (End a Subroutine), 145, 147
 GENCMDDOC (Generate Command Documentation), 440
 GO, 26, 46
 GOTO, 3, 133, 135
 IF, 136
 INZTAP (Initialize Tape), 8
 ITERATE, 143–144
 LEAVE, 143–145
 MONMSG (Monitor Message), 86, 197–206
 MOVOBJ (Move Object), 273
 OPNDBF (Open Database File), 326
 OPNQRYP (Open Query File), 326, 328–331, 333
 OVRDBF (Override with Database File), 272, 320–323
 OVRPRTF (Override with Printer File), 323–325
 PGM (Program), 83, 84
 POSDBF (Position Database File), 326
 RCVF (Receive File), 270, 280
 RCVMSG (Receive Message), 206, 297–305
 RETURN, 89, 181
 RGZPFM (Reorganize Physical File Member), 10
 RMVLIBLE (Remove Library List Entry), 175
 RMVMSG (Remove Message), 305–306
 RTNSUBR (Return from a Subroutine), 147
 RTVDTAARA (Retrieve Data Area), 275–276, 281
 RTVJOBA (Retrieve Job Attributes), 186, 238–239
 RTVMBRD (Retrieve Member Description), 248
 RTVNETA (Retrieve Network Attributes), 248
 RTVOBJD (Retrieve Object Description), 249
 RTVSYSVAL (Retrieve System Value), 159, 241–242
 RTVUSRPRF (Retrieve User Profile), 247–248
 RUNSQLSTM (Run SQL Statement), 332
 SAVLIB (Save Library), 85, 342
 SBMDBJOB (Submit Database Job), 3
 SBMJOB (Submit Job), 3, 182–186, 225–226
 SELECT, 140
 SENDPGMMSG (Send Program Message), 206
 SNDBRKMSG (Send Break Message), 293
 SNDF (Send File), 269, 280
 SNDMSG (Send Message), 7, 292
 SNDPGMMSG (Send Program Message), 80, 159, 293–295
 SNDRCVF (Send/Receive File), 138, 268–269, 280, 319
 SNDUSMSG (Send User Message), 295–297
 STRSEU (Start Source Entry Utility), 3, 54
 SUBR (Begin a Subroutine), 145
 TFRCTL (Transfer Control), 42, 179–181, 224–225
 UPDPGM (Update Program), 390
 WAIT, 271
 WRKACTJOB (Work with Active Job), 9
 WRKJOB (Work with Job), 9
 WRKMBRPDM (Work with Members Using PDM), 53
 WRKMSGD (Work with Message Description), 194–195
 WRKSBMJOB (Work with Submitted Jobs), 182
 WRKTMZON (Work with Time Zone Objects), 244
 Command parameters. *See* Parameters
 Command processing programs (CPP), 368, 425–435
 analyzing, 430–431
 defined, 404, 425
 EXCCMDC, 426–428
 integer processing, 433
 parameters, 425–430
 processing list parameters in, 432–434
 QCMDEXC as, 434–435
 summary, 435
 writing, 425
 Command prompt display, 29
 additional parameters, 35–36
 allowable values, 30–32
 consistency, 29
 defined, 28
 invoking, 28
 prompt special characters, 37–40
 Command prompting, 341–346
 commands not allowing, 343
 selective, 344–346
 simple, 341–343
 summary, 346
 Command properties, 367–374
 changing, 370–371
 displaying, 367–370
 summary, 374
 Commands
 aliases, 11
 defined, 1
 entering into source member, 54–55
 finding, 41–45
 grouping, 3
 IBM-supplied, changing, 372–373
 labels, 82
 logging, 380–381
 modifiers, 9–11
 most-often-used, 457
 names, 7–11
 program looping, 141–145
 retrieving, 27
 in REXX language procedure, 3
 subjects, 7–9
 user-defined, 403–414
 using, 1–2
 verbs, 7–9
 Command string (*CMDSTR) parameter, 182–183
 Command string function key (F14), 34

- Comments
 - continuation character for, 81
 - entering, 79
- Common Execution Environment (CEE) APIs, 359
- *COMP messages, 291
- Compiled program objects, 3
- Compile-time errors
 - defined, 63
 - diagnosing, 64–66
- COMP keyword, 264
- Complex expressions, 156
- Complex lists, 21
- Conditional branching, 136
- Conditional logic, 3
- Conditional loops, 141
- Conditional processing groups, 139
- Constants
 - character, 217–220
 - decimal, 221
 - logical, 220–221
 - passing, 217–221
 - values, 93
- Contextual help
 - defined, 439
 - window, 445
- Control Language. *See* CL
- Control structures, 133–148
 - DO groups, 139–140
 - GOTO command, 133, 135
 - IF/ELSE, 136–138
 - SELECT/WHEN/OTHERWISE, 140–141
 - structured programming, 134–135
 - subroutines, 145–147
 - summary, 148–149
- COPYRIGHT command, 85
- Count-controlled loops, 141
- CPF prefix, 196
- CPYF (Copy File) command, 16, 104
 - parameter order, 16–17
 - positional parameters, 17
- CPYFRMQRVF (Copy from Query File) command, 331–332, 333
 - defined, 331
 - form, 332
 - parameters, 332
- Cross reference, 63
- CRTBNDCL (Create Bound CL Program)
 - command, 35–36, 69, 389
 - defined, 69
 - prompted, 35, 69
 - prompted with additional parameters, 36
- CRTCLMOD (Create CL Module) command, 56, 58–60, 71
 - compile options, 60
 - defined, 56
 - ILE CL source member compilation, 388
 - invoking, 56
 - LOG parameter, 381
 - prompt display, 58–59
 - property display, 367
 - sample display, 369
- CRTCLPGM (Create CL Program), 70
 - in OPM program creation, 391
 - use, 3
- CRTCMD (Create Command) command, 9, 404
 - CMD parameter, 413
 - defined, 413
 - form, 413
 - SRCFILE parameter, 413
 - SRCMBR parameter, 413
- CRTDSPF (Create Display File) command, 10, 264
- CRTDTAARA (Create Data Area) command, 273, 274–275
 - defined, 274
 - form, 274
 - LEN parameter, 274
 - sample, 275
 - TYPE parameter, 274
 - VALUE parameter, 274
- CRTDTAQ (Create Data Queue) command, 278
- CRTLFL (Create Logical File) command, 10, 263–264
- CRTLIL (Create Library) command, 343
- CRTMSGQ (Create Message Queue) command, 290
- CRTPF (Create Physical File) command, 10, 262
- CRTPGM (Create Program) command, 68
 - in binding modules into programs, 389
 - BNDSRVPGM parameter, 393
 - ILE binding process invocation, 388
 - PGM parameter, 388
 - prompted, 68
- CRTPNLGRP (Create Panel Group) command, 445
- CRTPRTF (Create Printer File) command, 10, 324
- CRTSRCPF (Create Source Physical File)
 - command, 71
- CRTSRVPGM (Create Service Program) command, 392
- CRTUSRPRF (Create User Profile) command, 434
- Current library (CURLIB), 175
 - changing, 246
 - defined, 175
- CVTDAT (Convert Date) command, 249–250
 - DATE parameter, 249
 - defined, 249
 - form, 249
 - FROMFMT parameter, 249–250
 - TOFMT parameter, 250
 - TOSEP parameter, 250
- D**
- Data areas, 273–277
 - CL commands for, 273
 - contents, changing, 276
 - contents, displaying, 275
 - contents, retrieving, 275–276
 - creating, 273, 274–275
 - defined, 273
 - deleting, 275
 - information change, 273
 - local, 277
 - summary, 281
- Database files, 103, 277
 - logical, 257, 258, 263–264
 - opening, 326
 - overriding with, 272, 320–323
 - physical, 257, 258, 261–263
 - positioning, 326
 - preopening, 326–327
 - See also* Files
- Data conversion
 - character to numeric, 117
 - with CHGVAR, 116–117
 - numeric to character, 116–117
- Data Definition Language (DDL), 259
- Data description specifications (DDS), 260–266
 - defined, 260
 - for display files, 264–266
 - files without, 266
 - for logical files, 263–264
 - for physical files, 261–263
 - record description, 260
 - source members, 50
- Data Manipulation Language (DML), 259
- Data queues, 278–281
 - clearing, 280
 - creating, 278
 - defined, 278
 - deleting, 279
 - reasons to use, 280–281
 - receiving entries from, 279–280
 - sending entries to, 279
 - undesirable characteristics, 281
- Data values, 19
- Dates
 - conversion, 249–250
 - system values, 243–245
- DCL (Declare CL Variable) command, 93–103
 - ADDRESS parameter, 100
 - BASPTR parameter, 101
 - defined, 85, 93
 - DEFVAR parameter, 101–102
 - examples, 102–103
 - LEN parameter, 94–95
 - with pointers, 100–101
 - STG parameter, 101
 - TYPE parameter, 94
 - VALUE parameter, 96, 99
 - VAR parameter, 94
- DCLF (Declare File) command, 85, 103–104, 280
 - defined, 85, 267
 - examples, 104
 - FILE parameter, 267
 - multiple, 104
 - open file identifier, 103

OPNID parameter, 103, 268
 program-described files, 266
 RCDFMT parameter, 267

DCLPRCOPT (Declare Processing Options)
 command, 85, 147–148
 compiler processing options, 148
 defined, 85, 147
 parameters, 147–148
 SUBRSTACK parameter, 148

Debugging view, 60

Decimal constants
 defined, 221
 passing, 221
See also Constants

Decimal type (*DEC) variables
 CHGVAR with, 115
 default initial value, 96
 defined, 94
 length specification, 95
 parameter definition, 216
See also Program variables

Declarations section, 84–85

Default activation group, 394

Default values, 14
 defined, 13
 in positional notation, 18

Defined variables
 declaring, 101–102
 defined, 101
See also Program variables

DEP statement, 412

*DIAG messages, 291
 *ESCAPE message relationship, 310
 received determination, 308, 309
 receiving, 306
 resending, 310
 sending, 307

Display attributes, 266

Display files, 103, 277
 DDS for, 264–266
 defined, 259
 indicators, 265
See also Files

Display function key (F13), 32

DLTDTAQ (Delete Data Queue) command, 279

DLTF (Delete File) command, 1

DLTMOD (Delete Module) command, 389

DLTOVR (Delete Override) command, 323, 396

DLTPGM (Delete Program) command, 7

DMPCLPGM (Dump CL Program) command,
 122, 200, 201

DO command, 66, 139

DOFOR (Do For) command, 141, 142–143
 defined, 142
 form, 142
 FROM parameter, 143
 loops, 433
 VAR parameter, 143

DO groups, 139–140
 conditional processing, 139

defined, 139
 inserting commands into, 140
 with MONMSG command, 204
 programming practice, 140

DOUNTIL (Do Until) command, 142
 defined, 142
 loops, 308, 431

DOWHILE (Do While) command, 141–142
 defined, 141
 trailing decision loop, 142

DSPCMD (Display Command) command, 17, 367

DSPDTAARA (Display Data Area) command, 275

DSPFFD (Display File Field Descriptions)
 command, 273

DSPJOB (Display Job) command, 9, 377

DSPJOBLOG (Display Job Log) command, 204,
 377–378
 defined, 377
 job log display, 378

DSPLIB (Display Library) command, 38, 39, 341
 defined, 38
 with expanded parameter list, 39
 LIB parameter, 344
 OUTPUT parameter, 344

DSPLIBL (Display Library List) command, 175

DSPLOG (Display Log) command, 290

DSPMSGD (Display Message Description)
 command, 194

DSPPFM (Display Physical File Member) command,
 10, 20

DSPTAP (Display Tape) command, 8, 345
 DATA parameter, 14
 DEV parameter, 13, 14, 155
 with extended device parameter, 37
 keyword notation, 15
 LABEL parameter, 14, 30, 31
 OUTPUT parameter, 14
 parameters, 13
 partially prompted, 29
 positional notation, 15
 prompted, 28
 SEQNBR parameter, 14

Dynamic program calls, 391

E

Editing system values, 247

EDTF (Edit File) command, 7

EDTLIBL (Edit Library List) command, 175

ELEM statement, 412

ELSE command, 137

ENDDO (End Do) command, 66
 defined, 139
 DO command and, 139

ENDJOBABN (End Job Abnormally) command,
 245

ENDPGM command
 defined, 86
 validity, 3

ENDRCV (End Receive) command, 271

ENDSELECT command, 140

ENDSUBR (End a Subroutine) command, 145, 147

Enlarge function key (F20), 32

*ENTMOD activation group option, 395

Entry modules, 390

Error handling, 193–207
 ILE, 396–397
 standard routines, 306–312
 VCP, 447

Error messages, 193–194
 CHKOBJ command-generated, 203
 defined, 193
 detailed display, 205
 function key (F15), 34
 generic, 202–203
 ignoring, 205
 monitoring for, 197–206
 program messages display, 193–194
 receiving, 206
 sending, 206

Errors
 compile-time, 63–66
 detection, 207
 diagnosing with WDS c, 67
 displaying, 34
 severity, 66
 types of, 193
 within MONMSG, 203–204

*ESCAPE messages, 291
 ambiguous, 306
 CPE, 307
 defined, 87
 *DIAG message relationship, 310
 function check, 396
 read order, 300
 receiving, 306, 311–312
 sending, 307
 sent to message queue, 308
 text, receiving, 205–206

Event files, 60, 67

EXCCMDC CPP, 426–428

EXCCMD (Execute Commands) command,
 405–406
 creating, 446
 decimal parameters, 429
 IFERROR parameter, 444

Exit function key (F3), 27

Expressions, 155–166
 arithmetic, 118, 156–158
 character string, 158–160
 complex, 156
 defined, 155–156
 logical, 162–165
 order of evaluation, 166
 relational, 160–162
 summary, 166
 types of, 155

Extended help
 defined, 439
 function key (F2), 31

window, 443, 444
See also Help

External message queues
 defined, 290
 display, 297
 sending information message to, 297
See also Message queues

F

Field-level specification, 262

File cursors, 321

File member existence test, 196–197

Files
 advanced techniques, 319–333
 closing, 327
 copying, 16
 copying without declaring, 104
 database, 103, 257–258, 277
 declaring, 266
 deleting, 1
 display, 103, 259, 264–266, 277
 event, 60
 existence requirements, 259–260
 I/O processing support, 280
 logical, 257, 258, 263–264
 message, 194–197
 moving, 272
 overrides, 319–325
 physical, 257, 258, 261–263
 printer, 323–325
 processing, 267–271
 source physical, 50–51
 summary, 277–281
 using, 319
 without DDS, 266
 working with, 257–259

Free-format statement entry, 77–78

Function checks, 308, 396

Function keys
 additional parameters, 35
 all parameters, 33
 choices, 34
 command complete, 34
 command-line, 27
 command-prompt, 32–34
 command string, 34
 defined, 27
 display, 32
 enlarge, 32
 error messages, 34
 exit, 27
 extended help, 31
 help, 31
 keywords, 34
 more keys, 33
 prompt, 27, 41
 refresh, 32

retrieve, 27
 rotating through groups of, 33
 summary table, 36

G

GENCMDDOC (Generate Command Documentation) command, 440

Generic parameters, 26

Global message monitor. *See* Program-level MONMSG

GO command, 26, 46

GOTO command, 133

CMDLBL parameter, 135
 defined, 135
 example, 135
 label specification, 135
 unconditional branching, 135
 validity, 3

Graphical user interface (GUI) displays, 370

Greater than (>) symbol, 39

Grouping, CL commands, 3

H

Help
 contextual, 439, 445
 extended, 439, 443, 444
 function key (F1), 31
 identifier, 444

Help panel groups
 defined, 439
 EXCCMDH, 442

Hexadecimal notation, 227

Hexadecimal values, 19

Hierarchical decomposition, 145

High-level languages (HLLs), 2, 183
 calling APIs from, 352
 programs, enabling, 351

I

i5/OS name, 82

IBM i
 CL as system-control language, 1
 command-line operations, 27
 Information Center, 16, 19
 Integrated File System (IFS), 11
 menus, 26
 menu system, 25–26
 messages, 289
 prompt facility, 28–40
 user interface, 25–46

IF command
 COND parameter, 136, 156
 defined, 136
 (&MSGID=), 308

nested, 136–137
 THEN parameter, 136

IF/ELSE structure
 conditional branching, 136
 ELSE command, 137–138
 IF command, 136
 nested IF, 136–137
 nested IF/THEN/ELSE, 138

ILE programs, 388
 calling, 391–392
 creating, 388–390
 entry procedure (PEP), 299
 modules, adding, 391
 modules, creating, 388–390
 multiple-module, 389–390
 rebinding, 391
 service, 392–393
 single entry point, 390
 updating, 390–391
See also Integrated Language Environment (ILE)

Indicators, 265

*INFO messages, 291

Integer (*INT) variables
 assigning values to, 99
 CHGVAR with, 115
 default initial value, 96
 defined, 94
 length specification, 95
 parameter definition, 216
See also Program variables

Integer values, 19

Integrated File System (IFS), 11

Integrated Language Environment (ILE), 70, 387–398
 activation groups, 393–396
 binding step, 388
 in calling procedures, 177
 defined, 387
 error handling, 396–397
 percolation, 397
 summary, 397–398
 unhandled exceptions and, 397
See also ILE programs

Interactive jobs
 defined, 173
 log, 204–205

Interpreted job streams, 3

INZTAP (Initialize Tape) command, 8

ITERATE command, 143–144
 CMDLBL parameter, 144
 defined, 143
 example, 143–144
 in nested loops, 144

Iteration commands, 141–145

J

- Job Control Language (JCL), 1
- Job definition parameters, 184
- Job logs, 377–384
 - analyzing, 381–383
 - columns, 383
 - commands not logged in, 380
 - defined, 377
 - displaying, 377–378
 - heading, 383
 - interactive, 204–205
 - lean, 384
 - spooled, 382–383
 - summary, 384
 - WDS display, 378
- Job queues
 - defined, 174
 - job priority, 240
- Jobs
 - attributes, changing, 239–241
 - attributes, retrieving, 238
 - batch, 173
 - defined, 173
 - descriptions, 174
 - identifiers, 174
 - interactive, 173
 - library list, 174–175
 - name, 174
 - priority, 240
 - scoping, 396
 - type, 185
- Job streams, 3

K

- Keyword notation
 - defined, 15
 - entering parameters with, 15
 - entering statements in, 79
 - positional notation with, 18–19
 - when to use, 18
 - See also* Positional notation
- Keywords
 - COMP, 264
 - display, 34
 - function key (F11), 34
 - MAX, 432
 - PARM, 215
 - PMTCTLPGM, 449
 - PROMPT, 410
 - RANGE, 409
 - REL, 410
 - SPCVL, 409
 - VALUES, 409

L

- Labels
 - entry point identification, 135
 - GOTO, 135
 - uses, 82
- Last-in-first-out (LIFO), 179
- Leading decision loops, 141
- LEAVE command, 143–145
 - CMDLBL parameter, 144
 - example, 143–144
 - in nested loop, 144
- Less than (<) symbol, 40
- Libraries
 - creating, 343
 - current, 175, 246
 - displaying, 341
 - product, 175, 246
 - QSYS, 321, 373
 - QTEMP, 271
 - saving, 342
- Library lists
 - commands, 175
 - defined, 174
 - parameters, 184
 - system search, 174
 - system (SYSLIBL), 175
 - system values, 246
 - user (USRLIBL), 175
- Limited users, 369
- Local data areas, 277
- Localized variables, 223
- LOG attribute, 379–380
- LOGCLPGM attribute, 380–381
- Logging, 380–381
- Logical constants, 164
 - defined, 220
 - passing, 220–221
 - See also* Constants
- Logical expressions, 162–165
 - defined, 162
 - evaluating, 163–164
 - logical constants, 164
 - logical variables, 164
 - more than two operands, 165
 - operators, 163
 - summary, 166
 - value evaluation, 162
 - See also* Expressions
- Logical files
 - DDS for, 263–264
 - DDS using select/omit criteria, 263
 - defined, 257
 - physical file relationship, 258
 - specifications, 263
 - See also* Database files; Files
- Logical type (*LGL) variables
 - assigning values to, 98–99
 - CHGVAR with, 116
 - default initial value, 96
 - defined, 94
 - indicator declaration, 265
 - length specification, 95
 - in logical expressions, 164
 - parameter definition, 216
 - See also* Program variables
- Logical values, 19
- Loops, 141–145
 - conditional, 141
 - count-controlled, 141
 - DOFOR, 433
 - DOUNTIL, 308, 431
 - early exits and, 143–145
 - leading decision, 141
 - trailing decision, 142
- Lower case, 78
- LPEX Editor, 54

M

- MAIN menu, 25–26, 46
- MAJOR menu, 41–42
 - defined, 42
 - function, 42
 - illustrated, 41, 42
- Menus, 26
 - accessing, 46
 - CMDADD, 45
 - CMDFILE, 44, 45
 - CMD-type, 45
 - MAIN, 25–26, 46
 - MAJOR, 41–42
 - SLTCMD, 42
 - SUBJECT, 43
 - VERB, 42
- Message files (*MSGF), 194–197
 - content display, 194–195
 - defined, 194
 - QCPFMSG, 194, 195
- Message key, 290
- Message queues, 289–290
 - call stack entry, 290, 299
 - creating, 290
 - defined, 289–290
 - external, 290
 - job. *See* Job logs
 - system history log (QHST), 290
 - system operator (QSYSOPR), 290
 - user, 290
 - workstation, 290
- Messages
 - advanced handling, 289–312
 - *COMP, 291
 - defined, 289
 - *DIAG, 291, 306–307, 309–310, 312
 - display, 193–194

- *ESCAPE, 291, 306–308, 310–312
- IBM i, 289
- *INFO, 291
- logging level, 379
- *NOTIFY, 291
- receiving, 297–305
- removing, 305–306
- *RPY, 291
- *RQS, 291
- sending, 291–297
- severity level, 379
- *STATUS, 291
- summary, 312
- text level, 380
- types of, 290–291
- Message summary, 63
- Messaging/logging system values, 247
- Modifiers, 9–11
- Modules
 - adding, 391
 - creating, 388–390
 - debugging, 60
 - entry, 390
 - programs, 389–390
- MONMSG (Monitor Message) command, 86, 197–206, 431
 - CMPDTA parameter, 198
 - command-level, 198–199
 - defined, 197–198
 - DO groups with, 204
 - errors within, 203–204
 - EXEC parameter, 198, 200, 203
 - for generic messages, 202–203
 - mixed command-/program-level, 201–202
 - MSGID (CPF0000), 207
 - MSGID (CPF9800), 207
 - MSGID (CPF9801), 207
 - MSGID parameter, 198
 - multiple commands, 203
 - program-level, 199–201
 - specific messages, 207
 - summary, 207
 - syntax, 198
 - use examples, 198
- More keys function key (F24), 33
- MOVOBJ (Move Object) command, 273
- Multiple entry point programs, 393
- MYPROGRAM sample source member, 60–63
 - compiler listing, 62–63
 - defined, 60
 - illustrated, 61
- MYPROGRAM2 sample source member, 64–66

N

- Name values, 19
- Nested IF/THEN/ELSE structures, 138
- Nest IF control structure, 136–137

- Network attributes, 248
- *N value, 18
- *NOTIFY messages, 291, 300
- *NOT operator, 163, 164
- Numeric variables
 - converting character data to, 117
 - converting to character data, 116–117
 - See also* Program variables

O

- Object authority test, 197
- Object existence test, 196
- Offset, 100
 - %OFS function. *See* %OFFSET function
 - %OFFSET function, 120, 124, 434
- Online documentation, 19
- Open data paths
 - defined, 322
 - sharing of, 326
- Open file identifiers, 103
- Operands, 156
 - %BIN function as, 157
 - logical expression, 165
 - relational expressions, 160
- Operating system messages, 196
- Operations
 - arithmetic, 156
 - order of, 157
- Operators
 - *AND, 163, 164
 - arithmetic expression, 156
 - *BCAT, 119, 158, 159
 - *CAT, 119, 158, 159
 - character string expression, 158
 - logical, 163, 164
 - relational, 161
 - *TCAT, 119, 158, 159
- OPNDBF (Open Database File) command, 326
- OPNQRYF (Open Query File) command, 326, 328–331, 333
 - defined, 328
 - example use, 330–331
 - FILE parameter, 328
 - form, 328
 - FORMAT parameter, 329
 - KEYFLD parameter, 330
 - OPTION parameter, 328
 - QRYSLT parameter, 329, 331
 - subject, 331
 - uses, 328
 - using, 330–331
- Optional parameters, 13
- Original Program Model (OPM) programs, 70, 117, 298
 - call stack entry message queue, 299
 - creating, 391
 - running, 387

- *OR operator, 163, 164
- Outfiles, 271
- OUTPUT
 - *OUTFILE, 271
 - support, 271–273
- Output parameters, 184–185
- Overrides, 319–325
 - activation groups and, 395–396
 - commands, 319
 - database file, 320–323
 - defined, 319
 - deleting, 323
 - printer file, 323–325
 - rules, 322
 - scope, 322
 - summary, 333
- OVRDBF (Override with Database File) command, 272, 320–323
 - defined, 320
 - FILE parameter, 320
 - form, 320
 - MBR parameter, 321, 326
 - OVRSCOPE parameter, 322
 - POSITION parameter, 321
 - SHARE parameter, 322
 - TOFILE parameter, 320
- OVRPRTF (Override with Printer File) command, 323–325
 - applying, 325
 - defined, 323
 - example use, 324–325
 - FILE parameter, 324
 - form, 324
 - purposes, 324

P

- Panel groups
 - associating with commands, 445–446
 - creating, 445
 - defined, 439
 - help, 439
 - summary, 449
- Parameter list, 17
 - entries, inserting, 39
 - entries, removing, 40
 - expanded, 38, 39
- Parameters, 13–22
 - additional, 35–36
 - arguments versus, 215
 - character, 19
 - choices, 34
 - command processing programs (CPP), 425–430
 - command string, 19
 - complex lists, 21
 - date, 19
 - decimal, 19
 - default values, changing, 372

- defined, 13, 215
- defining in called program, 216–217
- displaying all, 33
- entering, with keyword notation, 15
- entering, with positional notation, 15–18
- function, 215
- generic, 26
- hexadecimal, 19
- integer, 19
- job definition, 184
- keyword and positional notation mixture, 18–19
- library list, 184
- list, processing, 432–434
- logical, 19
- name, 19
- optional, 13
- order of, 16–17
- output, 184–185
- passing, 176–177, 215–227
- program variables as, 215
- required, 13
- scheduling, 185
- special values, 19–20
- summary, 21–22
- time, 19
- types, 19
- value lists, 20–21
- values, prompting for, 30
- PARM keyword, 215
- PARM statement, 407–410
 - CONSTANT parameter, 434
 - defined, 407
 - example, 408
 - KWD parameter, 408
 - LEN parameter, 408
 - MIN parameter, 410
 - PMTCTLPGM keyword, 449
 - PROMPT keyword, 410
 - RANGE keyword, 409
 - REL keyword, 410
 - RSTD parameter, 409
 - SPCVAl keyword, 409
 - TYPE parameter, 408
 - VALUES keyword, 409
 - See also* Command definition statements
- Pass by read-only reference, 223
- Pass by reference, 223
- Passing constants, 217–221
 - character, 217–220
 - decimal, 221
 - defined, 217
 - logical, 220–221
- Passing parameters, 176–177, 215–227
 - constants to called program, 217–221
 - illustrated, 177
 - by read-only reference, 223
 - by reference, 223
 - with SBMJOB command, 225–226
- steps, 176
- summary, 228
- TFRCTL for, 181, 224–225
- variables to called program, 221–224
- Passing variables, 221–224
 - CALL command, 221, 222
 - examples, 221
 - mechanics, 222–224
- Percolation, 397
- PGM (Program) command, 83, 84
 - as beginning boundary, 83
 - PARM parameter, 84, 100
- Physical files
 - DDS for, 261–263
 - defined, 257
 - field-level specification, 262
 - logical file relationship, 258
 - record format specification, 262
 - source, 50–51
 - specifications, 261
 - See also* Database files; files
- Place holders, 18
- Plus (+) symbol, 38–39
- PMTCTL statement, 412–413
- Pointer (*PTR) variables
 - built-in functions with, 123–124
 - DCL with, 100–101
 - defined, 94
 - parameter definition, 216
 - uses, 100
- POSDBF (Position Database File) command, 326
- Positional notation
 - compact nature, 16
 - default values, 18
 - defined, 15
 - entering parameters with, 15–18
 - entering statements in, 79
 - keyword limits, 17
 - keyword notation with, 18–19
 - maximum number of parameters, 17
 - parameter order, 16–17
 - programming style, 18
 - See also* Keyword notation
- Preopening database files, 326–327
- Printer files
 - creating, 324
 - overriding, 323–325
- Procedures
 - calling, 177, 391–392
 - defined, 177, 387
 - program entry, 388
 - return values, 392
 - static calls, 391–392
- Procedure section, 85–87
 - CHGVAR (Change Variable) command, 113
 - defined, 85
 - error procedure segment, 87
 - global MONMSG commands, 86
 - illustrated, 85–86
 - main procedure segment, 86
 - segments, 86
 - subroutine segment, 87
 - See also* Source members
- Product library (PRDLIB), 175
- Program Automatic Storage Area (PASA), 222
- Program entry procedures, 299, 388
- Program information section, 83
- Program-level MONMSG, 199–201
 - branching to error-handling routine, 200
 - defined, 198
 - EXEC parameter, 200
 - as global message monitor, 199
 - mixed with command-level MONMSG, 201–202
 - See also* MONMSG (Monitor Message) command
- Program linkage section, 83–84
- Program looping commands, 141–145
 - defined, 141
 - DOFOR, 142–143
 - DOUNTIL, 142
 - DOWHILE, 141–142
 - early exits, 143–145
- Programming Development Manager (PDM), 52, 544–550
 - adding source members with, 53–54
 - compiling CL source member with, 58
 - working with members with, 53
 - See also* Source Entry Utility (SEU)
- Programs. *See* CL programs; ILE programs
- Program testing and debugging, 551–560
- Program variables
 - assigning values to, 97–99
 - based, 100–101
 - character type (*CHAR), 94, 95, 96, 97–98
 - data types, 94, 105
 - with DCL command, 93–103
 - with DCLF command, 103–104
 - decimal type (*DEC), 94, 95, 96, 98–99
 - declaring, 93–105
 - defined, 93, 101–102
 - existence, 93
 - integer (*INT), 94, 95, 96, 99
 - localized, 223
 - logical type (*LGL), 94, 95, 96, 98
 - manipulating, 113–124
 - naming, 94
 - offset, 100
 - as parameters, 215
 - passing to called program, 221–224
 - pointer type (*PTR), 94, 100
 - size, specifying, 94–95
 - storage area for, 224
 - summary, 105
 - unsigned integer (*UINT), 94, 95, 96, 99
 - values, 93
 - values, initializing, 96
 - in VALUE specification, 118

Prompt choice programs, 448
 Prompt control programs, 449
 Prompt facility, 28–40
 additional parameters, 35–36
 allowable values, 30–32
 command prompt display, 29
 command prompt function keys, 32–34
 function key, 27
 prompt special characters, 37–40
 review, 341
 Prompt function key (F4), 27, 41
 Prompt override programs, 448

Q

QABNORMSW system value, 245
 QADSPJOB model file, 271, 272, 273, 321
 QCBLESRC source file, 50
 QCENTURY system value, 243
 QCLRDTAQ API, 280
 QCLSCAN API, 358, 361
 QCLSRC source file, 50
 QCMD API, 298, 352–353, 361
 command line, 353
 defined, 352
 using, 352–353
 QCMDCHK API, 355–356, 361
 defined, 355
 error detection, 356
 form, 355
 prompting with, 355–356
 QCMDEXC API, 356–358, 361, 431
 call levels and, 358
 as CPP, 434–435
 defined, 356
 format, 356
 parameter rules, 356–357
 program example, 357–358
 prompting with, 357
 QCPFMSG message file, 194, 195, 294
 defined, 194
 error condition messages, 195
 QDATE system value, 243
 QDATETIME system value, 243
 QDATFMT system value, 244
 QDAYOFWEEK system value, 243
 QDCXLATE API, 358, 361
 QDDSSRC source file, 50
 QIPLDATTIM system value, 246
 QIPLSTS system value, 246
 QMODEL system value, 246
 QPJOBLOG spooled file, 381
 QPRTDEV system value, 246
 QRCVDTAQ API, 279, 358–359
 QRPGLSRC source file, 50
 QSNDDTAQ API, 279, 358–359
 QSRLNBR system value, 246

QSYSLIBL system value, 246
 QSYS library, 321
 IBM commands in, 372
 user library ahead of, 373
 QTEMP library, 271
 QTIME system value, 243
 QTIMZON system value, 244
 Qualified character (/), 20, 22
 Qualified names, 410
 Qualified values
 defined, 20
 job name, 174
 QUAL statement, 410–411
 Question mark (?), 37
 QUSCMDLN API, 354, 361
 QUSRLIBL system value, 246

R

Rational Developer for System i (RDi), 50, 52
 RCVF (Receive File) command, 270, 280
 RCVMSG (Receive Message) command, 206, 297–305
 defined, 297
 examples, 304–305
 form, 297
 KEYVAR parameter, 302
 MSGDTALEN parameter, 302
 MSGDTA parameter, 302
 MSGF parameter, 304
 MSGID parameter, 302
 MSGKEY parameter, 301
 MSGLEN parameter, 302
 MSG parameter, 302
 MSGQ parameter, 301
 MSGTYPE parameter, 301
 PGMQ parameter, 301
 read operations, 300
 RMV parameter, 302
 RTNTYPE parameter, 303
 SENDER parameter, 302
 SEV parameter, 302
 SNDMSGFLIB parameter, 302, 304
 WAIT parameter, 301
 Record format specification, 262
 Refresh function key (F5), 32
 Relational expressions, 160–162
 character operands, 162
 with CHGVAR command, 162
 defined, 160
 evaluation, 161
 examples, 160
 operands, 160
 operators, 161
 summary, 166
 uses, 160
 See also Expressions
 Relational operators, 161
 Reports
 analyzing, 63
 CL compiler, 60–63
 compile-time errors, 63
 cross reference, 63
 message summary, 63
 source listing, 63
 Required parameters, 13
 Retrieve function key (F9), 27
 RETURN command, 89
 defined, 181
 implicit operation, 181
 Return values
 called procedure, 392
 parameter, 147
 using, 147
 Return variables, 237
 REXX language procedure, 3
 RGZPFM (Reorganize Physical File Member) command, 10
 RMVLIBL (Remove Library List Entry) command, 175
 RMVMSG (Remove Message) command, 305–306
 CLEAR parameter, 306
 defined, 305
 examples, 306
 form, 305
 MSGKEY parameter, 306
 MSGQ parameter, 305
 PGMQ parameter, 305, 306
 *RPY messages, 291
 *RQS messages, 291
 RTNSUBR (Return from a Subroutine) command, 147
 RTVDTAARA (Retrieve Data Area), 275–276, 281
 defined, 275
 DTAARA parameter, 276
 example, 276
 form, 275
 RTVJOBA (Retrieve Job Attributes) command, 186, 238–239
 defined, 238
 example use, 239
 form, 238
 parameters, 238
 using, 238–239
 RTVMBRD (Retrieve Member Description) command, 248
 RTVNETA (Retrieve Network Attributes) command, 248
 RTVOBJD (Retrieve Object Description) command, 249
 RTVSYSVAL (Retrieve System Value) command, 159, 241–242
 defined, 241
 example, 241–242
 form, 241
 RTVUSRPRF (Retrieve User Profile) command, 247–248
 RUNSQLSTM (Run SQL Statement) command, 332

S

- SAVLIB (Save Library) command, 85, 342
 - command prompt display, 342
 - defined, 85
- SBMDBJOB (Submit Database Job) command, 3
- SBMJOB (Submit Job) command, 3, 182–186, 187
 - caution, 192
 - CMD parameter, 182–183
 - CURLIB parameter, 184
 - defined, 182
 - HOLD parameter, 185
 - INLLIBL parameter, 184
 - job definition parameters, 184
 - JOB parameter, 184
 - JOBQ parameter, 184
 - OUTQ parameter, 185
 - parameters, 183–185
 - passing parameters with, 225–226
 - PRTDEV parameter, 184–185
 - RQSDTA parameter, 356
 - SCDDATE parameter, 185
 - SCDIME parameter, 185
 - self-submitting program, 185–186
 - SYSLIBL parameter, 184
- Scheduling parameters, 185
- Scope, override, 322
- Scoping
 - activation group, 396
 - call stack level, 396
 - defined, 393
 - job, 396
- Screen Design Aid (SDA), 264
- Security system values, 247
- SELECT command, 140
- Selective prompting, 344–346
 - characters, 344, 346
 - defined, 344
 - table summary, 345
 - uses, 346
 - See also* Command prompting
- SELECT/WHEN/OTHERWISE structure, 140–141
- Self-submitting program, 185–186
- SENDPGMMSG (Send Program Message) command, 206
- Service programs
 - benefits, 393
 - defined, 392
 - multiple entry point, 393
 - using, 392–393
- Shared open
 - defined, 326
 - methods, 327
- Simple command prompting, 341–343
- Simple lists, 432
- Single entry point programs, 390
- SLTCMD menu, 42
- SNDBRKMSG (Send Break Message) command, 293
- SNDF (Send File) command, 269, 280
- SNDMSG (Send Message) command, 7, 292
 - defined, 292
 - form, 292
 - MSG parameter, 292
 - MSGTYPE parameter, 292
 - RPYMSGQ parameter, 292
 - TOMSGQ parameter, 292, 342
 - TOUSR parameter, 292
- SENDPGMMSG (Send Program Message) command, 80, 159, 293–295
 - defined, 293
 - forms, 293
 - for immediate messages, 293
 - KEYVAR parameter, 295
 - MSGDTA parameter, 294
 - MSG parameter, 294
 - for predefined messages, 293
 - RPYMSGQ parameter, 294
 - TOPGMQ parameter, 294, 298–300
 - TOPGMQ parameter examples, 300
- SNDRCVF (Send/Receive File) command, 138, 280, 319
 - defined, 268
 - to display menu and process response, 269
 - form, 268
 - OPNID parameter, 268
 - RCDfmt parameter, 268
 - WAIT parameter, 268
- SNDUSRMSG (Send User Message) command, 295–297
 - defined, 295
 - DFT parameter, 296
 - forms, 296
 - for impromptu messages, 296
 - MSGRPY parameter, 296
 - for predefined messages, 296
 - TRNTBL parameter, 297
 - VALUES parameter, 296
- Source Entry Utility (SEU), 27
 - defined, 54
 - sample editing session, 55
- Source listing, 63
- Source members
 - adding, 52–54
 - CL, 50
 - Cobol, 50
 - compiling with PDM, 58
 - compiling with WDS_c, 57
 - complete illustrated, 87–88
 - DDS, 50
 - declarations section, 84–85
 - defined, 50
 - entering CL commands into, 54–55
 - procedure section, 85–87
 - program information section, 83
 - program linkage section, 83–84
- RPG, 50
 - sections, 82–87, 89
 - See also* Statements
- Source physical files
 - creating, 51
 - defined, 50
 - members, 50
 - using, 50–51
- Source statements, 50
- Special characters
 - ampersand (&), 37
 - greater than (>) symbol, 39
 - less than (<) symbol, 40
 - plus (+) symbol, 38–39
 - prompting, 37–40
 - question mark (?), 37
 - types of, 37
- Special values, 18
 - defined, 20
 - parameter, 19–20
- %SST function. *See* %SUBSTRING function
- Standard error-handling routines, 306–312
- Statements
 - blank lines, 78
 - command continuation, 79–81
 - comments, 79
 - entering, 77–82
 - free-format entry, 77–78
 - keyword notation, 79
 - labels, 82
 - lowercase entry, 78
 - positional notation, 79
 - uppercase entry, 78
- Static procedure calls, 391–392
 - bind-by-copy version, 392
 - defined, 391
- *STATUS messages, 291
- STRSEU (Start Source Entry Utility) command, 54
 - display, 54
 - prompts, 54
 - validity, 3
- Structured programming, 133–144
- Structured Query Language (SQL), 259
 - collections, 259
 - statement processing, 332
 - tables, 259
- SUBJECT menu, 43
- Subjects
 - common, 9
 - phrases, 9–11
 - verb compatibility, 9
- SUBR (Begin a Subroutine) command, 145
- Subroutines, 145–147
 - beginning/ending, 145
 - calling, 145–146
 - defined, 145
 - leaving, 147
 - return values, 147

stack, 147
See also Control structures

%SUBSTRING function, 120–121
 arguments, 120
 in character string expressions, 159–160
 with *CHAR variables, 121
 defined, 120
 in field extraction, 266
 in receiver variable, 121

%SWITCH function, 120

System control system values, 245–246

System history log message queue (QHST), 290

System i, 16, 25

System library list (SYSLIBL), 175

System-named activation group, 394

System operator message queue (QSYSOPR), 290

System values, 241–247
 allocation/storage, 247
 categories, 242–247
 changing, 242
 date/time, 243–245
 defined, 241
 editing, 247
 library list, 246
 messaging/logging, 247
 QABNORMSW, 245
 QCENTURY, 243
 QDATE, 243
 QDATETIME, 243
 QDATFMT, 244
 QDAYOFWEEK, 243
 QIPLDATTIM, 246
 QIPLSTS, 246
 QMODEL, 246
 QPRTDEV, 246
 QSRLNBR, 246
 QSYSLIBL, 246
 QTIME, 243
 QTIMZON, 244
 QUSRLIBL, 246
 retrieving, 241–242
 security, 247
 system control, 245–246

T

Tables, 259

*TCAT operator, 119, 158
 defined, 119
 symbol, 159

TFRCTL (Transfer Control) command, 42,
 179–181, 187
 call stack invocation levels, 180–181
 defined, 179
 form, 180
 invalid use, 224
 memory savings, 181
 passing parameters with, 181, 224–225

restrictions, 181

Threads, 370

Time
 system values, 243–245
 values, 19

Top-down design, 145

Trailing decision loops, 142

U

Unconditional branching, 135

Unhandled exceptions
 defined, 396
 ILE and, 397

Unsigned integer (*UINT) variables
 assigning values to, 99
 CHGVAR with, 115
 default initial value, 96
 defined, 94
 length specification, 95
 parameter definition, 216
See also Program variables

UPDPGM (Update Program) command, 390

Upper case, 78

User-defined commands, 403–414
 benefits, 403
 command definition statements, 404–413
 components, 404
 EXCCMD (Execute Commands), 405–406
 program design and, 403
 summary, 414
 system administration and, 403
 user interface and, 403

User Interface Manager (UIM)
 defined, 439
 :HELP./:EHELP., 443
 :HP1./:EHP1., 440, 443
 :PARML./:EPARML., 445
 :PNLGRP./:EPNLGRP., 443
 sample source, 441
 source creation, 440
 summary, 449
 tag language, 440–445

User library list (USRLIBL), 175

User message queue, 290

User-named activation group, 394

User profiles
 changing, 248
 retrieving, 247–248

Users, limited, 369

V

Validity checking programs (VCPs), 446–448
 defined, 446
 error-handling routine, 447
 EXCCMDV, 447

purpose, 446
 rules, 446
 running, 448
 summary, 449

Value lists, 20–21
 defined, 20
 example, 20–21

Variables. *See* Program variables

VERB menu, 42

Verbs
 common, 8
 subject compatibility, 9

W

WAIT command, 271

WebSphere Development Studio Client (WdSc),
 50, 532–543
 adding source members with, 52–53
 compiling CL source member with, 57
 defined, 71
 error diagnosis with, 67
 error list, 67
 event files, 67
 job log display, 378
 sample editing session, 55

Workflow control, 3, 173–187
 command batch execution, 182–186
 program execution with CALL command,
 175–179
 program execution with TFRCTL command,
 179–181
 RETURN command, 181–182
 summary, 187
 work management, 173–175

Work management, 173

Workstation message queue, 290

WRKACTJOB (Work with Active Job) command, 9

WRKJOB (Work with Job) command, 9

WRKMBRPDM (Work with Members Using PDM)
 command, 53

WRKMSGD (Work with Message Description)
 command, 194–195

WRKSBJOB (Work with Submitted Jobs)
 command, 182

WRKTIMZON (Work with Time Zone Objects)
 command, 244