

# Index

## Symbols

- 0: Unknown drive type, 196
- 1: Removable drive type, 196
- 2: Fixed drive type, 196
- 3: Network drive type, 196
- 4: CD-ROM drive type, 196
- 5: RAM Disk drive type, 196

## A

- Abort button, 107
- Abs( ) function, 126
- ACEs (access control entries), WMI (Windows Management Instrumentation), 300
- Active Directory Services Interface (ADSI). See ADSI (Active Directory Services Interface)
- Active Server Pages (ASPs), 384, 410
- ActiveX Data Objects (ADO). See ADO (ActiveX Data Objects)
- ActiveX scripting languages, 16-18
- AD (Active Directory)
  - attributes, 260
  - classes, 260
  - OUs (organizational units)
    - creating, 260-261
    - deleting, 263-264
    - modifying, 261-263
    - querying, 263
  - users and computers, targeting, 477-479
- AD (Active Directory) scripts
  - logoff scripts, 451-454
  - logon scripts, 440-446, 453-454
    - specific logon scripts, 446-451

**AddPrinterConnection method (Network object), 177****addresses**

- IP addresses, third octet, 63

- local IP addresses, attaining, 61-63

**AddUsers.vbs listing (28.1), 456-463****AddUsersFromXLS.vbs listing (3.2), 49-50****AddUsersFromXLS.vbs listing (3.3), 50-53****AddWindowsPrinterConnection method (Network object), 177****ADLogon1.vbs listing (27.3), 446-448****ADLogon2.vbs listing (27.4), 448-451****administrative scripts**

- domains, 455, 480

- inactive users, finding, 464-469

- network administrative scripts, 481, 495

- listing hot fixes, 492-495

- listing software, 492-495

- open file monitoring, 486-488

- remote computer shutdowns, 481-483

- remote MSI package uninstalls, 489-492

- remote share listings, 484-486

- system information, collecting, 469-474

- templates, 474-479

- user creation, automating, 455-463

**ADO (ActiveX Data Objects), 384, 455**

- Excel spreadsheets, queries, 455-463

**Adobe Dreamweaver, 410****ADOs (ActiveX Data Objects), 422****ADSI (Active Directory Services Interface), 168, 173, 247, 257, 335, 355, 497, 508**

- domains

- manipulating, 269

- OUs (organizational units), 260-264

- preloading, 264-269

- querying, 257-259

- reconfiguring, 259

- functions, writing, 340-342

- mass password changes, executing, 503-505

- object creation scripts, creating, 500-502

- object deletion scripts, creating, 502-503

- object query scripts, creating, 502

- objects, 247-249

- provider names, 248

- providers, 254-255

- GC (Global Catalog), 254

- IIS, 254

- LDAP, 248-254

- NDS (NetWare Directory Services), 254

- NWCOMPAT, 254

- OLE DB, 254

- WinNT, 248-252

- queries, debugging, 366-370

- remote shares, listing, 484-486

- scripts

- debugging, 357-358

- designing, 335-340

- testing, 351-355

- writing, 343-350

- subroutines, writing, 340-342

**ADSIDebug.vbs listing (21.2), 368-370****ADSystemInfo object, 179-180****advanced arithmetic, VBScript, 118-121****Advanced VBScript for Windows Administrators, 409, 420****American National Standards Institute (ANSI), 299****And Boolean operator, 156****ANSI (American National Standards Institute), 299****antivirus software, 406****AppActivate method (Shell object), 185**

**applications**

- graphical applications, 410
- HTAs (HTML Applications), 409-410, 420
  - building, 410
  - event handlers, 410, 418-420
  - event-driven scripting, 410
  - functional code, 411-412
  - HTML code conversion, 414-416
  - Mshsta.exe, 411
  - scheduled tasks, 410
  - script code, 416-418
  - security context, 409
  - user interface design, 412-413
  - writing, 411-420

**“Are you sure?” dialog boxes, 104**

**Argument method (Shortcut object), 187**

**Arithmetic, VBScript, 116-118**

- advanced arithmetic, 118-121
- Boolean math, 121-126

**arrays, 145, 148, 152**

- creating, 149-151
- dynamic arrays, 149
- manipulating, 149-151
- multidimensional arrays, 149
- numeric variables, array elements, 151-152
- VBScript, 149

**Asc( ) function, 118, 130**

**ASPs (Active Server Pages), 384, 410**

**assigning data to variables, 82-83**

**associations**

- WMI (Windows Management Instrumentation), 319
  - instances, 321-325
- WQL (WMI Query Language), 317

**associators, WQL (WMI Query Language), 303**

**ASSOCIATORS OF query, WMI instances, associating, 321-325**

**Atn( ) function, 118**

**attributes, AD (Active Directory), 260**

**Attributes property (Folder objects), 200-202**

**Attributes property (FSO File object), 204**

**automating**

- tasks, domains, 455, 480
- user creation, domains, 455-456, 459-463

**AvailableSpace property (Drive object), 195**

**B**

**batch files, 2, 13, 20**

**best practices, MsgBox statement, 106-107**

**Boolean math, VBScript, 121-126**

**Boolean operators, 156-157**

- WQL (WMI Query Language), 301-302

**bugs, scripts, 421-422, 435**

- logic errors, debugging, 425-435
- prevention, 31
- syntax errors, preventing, 422-425

**building HTAs (HTML Applications), 410**

**BuildPath function, FSO (FileSystemObject), 209-210**

**built-in commands, 75**

**built-in functions, 42**

- custom functions, compared, 88-92

**built-in functions. See intrinsic functions**

**built-in subroutines. See statements**

**buttons**

- Abort, 107
- dialog boxes, Cancel button, 105
- Ignore, 107
- message boxes, 103
- Retry, 107

**bytes, 145, 152**

- converting, 152
- VBScript, 152

## C

**Calculator, Scientific mode, 124**

**Cancel button, dialog boxes, 105**

**candidate modules, identifying, 223**

**CBool( ) function, 126**

**CByte( ) function, 152**

**CCur( ) function, 126**

**CD-ROM drive type, 196**

**CDbl( ) function, 126**

**certificate rules, 400**

### characters

double quotation marks, strings, 129-130

line continuation characters (\\_), 79, 337

underscore (\\_) character, 79, 106

**CheckFolder.vbs listing (12.3), 200-202**

**CheckGroupMembership.vbs listing (16.1), 276-277**

**checking group membership, 61**

**CheckRoutes.vbs listing (17.2), 297**

**Chr( ) function, 130**

**CIM (Common Information Model), WMI (Windows Management Information), 284**

**CInt( ) function, 126**

### classes

AD (Active Directory), 260

finding, WMI (Windows Management Instrumentation), 293-294

WMI (Windows Management Instrumentation), 284-285, 305

associations, 319-325

interrelated classes, 300, 317-320

Win32\_OperatingSystem class, 336-337, 349, 482-483

WQL (WMI Query Language), interrelated classes, 317

**Classid parameter (WSCs), 383**

**CLng( ) function, 126**

### code

bugs, 421-422, 435

logic errors, 425-435

syntax errors, 422-425

debug code, adding, 236, 239

debugging

logic errors, 425-435

syntax errors, 422-425

error-handling code, 70

functional code, HTAs (HTML Applications), 411-412

HTML code, converting for HTA (HTML Applications), 414-416

inline code, 45

pseudocode, 56

reusable code, 31

reusing, 424-425

script code, adding to HTA (HTML Applications), 416, 418

**code wizards, 32**

**coercion, numeric data types, 126-127**

**collections, COM objects, 96**

**CollectSysInfo.vbs listing (28.3), 469-474**

**COM (Component Object Model) objects, 18, 20, 76, 94-98, 439**

collections, 96

methods, 95-96

properties, 95

sample object (Tree object), 97-98

scripting with, 98-99

**command-line input, message boxes, asking for, 109-110**

**command-line parameters as input, 110-114**

**command-line scripts, 23-25**

execution of, 110

parameters, parsing, 111-114

running, 111

**command-line utilities, 110**

**commands**

- built-in commands, 75
- DatePart, 224
- Format, 223
- Len, 225
- MapPrinter, 67
- net, 60
- NET TIME, 445
- Option Explicit, 81
- Right, 224
- Sleep, 186
- WScript.Echo, 213-214

**comments, 48-53**

**Common Information Model (CIM), WMI (Windows Management Instrumentation), 284**

**comparison operators, WQL (WMI Query Language), 302-303**

**complex information**

- WMI (Windows Management Instrumentation), 300, 317-319
- WQL (WMI Query Language), 317

**complex information queries**

- testing, 325-326
- writing, 325
  - VBScript, 326-333

**Component Object Model (COM) objects. See COM (Component Object Model) objects**

**CompressAll.vbs listing (19.2), 330-333**

**computer programming, 75**

**ComputerName method (Network object), 178**

**computers, minimum requirements, 8**

**concatenating, strings, 135-139**

**conditional errors, 70**

**conditional execution statements**

- If/Then, 156-161, 170-171
- Select/Case, 161-162

**conditional executions, scripts, 155-162**

- If/Then conditional execution statement, 156-161, 170-171
- Select/Case conditional execution statement, 161-162

**connections, remote computers, WMI (Windows Management Instrumentation), 287-289**

**Const statement, 80**

**constants, 466**

- variables, compared, 80

**constructs**

- If/Then, 69
- Select/Case, 48

**consumers, WMI (Windows Management Instrumentation), 285-286**

**control-of-flow statements, 155**

**converting**

- bytes, 152
- data types
  - to numeric data, 127
  - to strings, 142-143
- date and time data, 147
- HTML code, HTAs (HTML Applications), 414-416
- numeric data types, 126-127

**Copy method (Folder objects), 198**

**Copy method (FSO File objects), 204**

**CopyFolder method (FSO), 197**

**copying-and-pasting, script, 373-377**

- WSCs (Windows Script Components), 378, 382-384, 387

**Cos( ) function, 118**

**cosines, 115**

**Create method, 260-261**

**Create method (objDomain), 248**

**CreateFolder method (FSO), 197**

**CreateObject, 247-248**

**CreateShortcut method (Shell object), 183, 187**

**CreateTextFile method (Folder objects), 198, 347**

**critical icon, 107**

**CScript.exe, 23-25, 110, 114**

command-line parameters, 114

**CScript.exe executable, deleting, 405-406**

**CStrng( ) function, 126**

**CStr( ) function, 130, 143**

**currency number types, 116**

**custom functions**

intrinsic functions, compared, 48, 88-92

main script, 45-46

**custom statements. See subroutines**

**custom subroutines**

creating, 92-94

main script, 45-46

statements, compared, 48

## D

**data coercion, 83-85**

**data types, 82, 145**

assumptions, 127

date and time, 145-148

converting, 147

numeric data, converting to, 127

numeric data types

converting, 126-127

converting to, 127

strings, converting to, 142-143

**Database Browser, 32**

**date and time data type, 145-148**

converting, 147

**Date( ) function, 42, 85, 146, 227**

**DateAdd( ) function, 147-148, 223**

**DateCreated property (FSO File object), 204**

**DateDiff( ) function, 87, 467**

**DateLastAccessed property (FSO File object), 204**

**DateLastModified property (FSO File object), 204**

**DatePart command, 224**

**DatePart( ) function, 118, 136, 146**

**dates**

future dates, working with, 147-148

getting, 146-147

past dates, working with, 147-148

VBScript, 145-148

**Day( ) function, 146**

**debug code, adding, 236, 239**

**debuggers, 434-435**

**debugging**

ADSI queries, 366-370

first script, 35-36

scripts, 357-358, 435

logic errors, 425-435

spaghetti debugging, 425-426

WMI queries, 358-370

**declaring variables, 79-81**

**default script extensions, Windows XP, 399**

**defaults, message boxes, 103**

**defining function input/output, 223**

**Delete method, 263**

**Delete method (Folder objects), 198**

**Delete method (FSO File objects), 204**

**Delete method (objDomain), 249**

**DeleteFolder method (FSO), 197**

**DeleteMail method, 96**

**deleting OUs (organizational units), 263-264**

**deployment, scripts, 17**

**Description method (Shortcut object), 187**

**Description parameter (WSCs), 383**

- designing**
  - scripts, 55, 219-220, 335-340
    - fact gathering, 220-221
    - script outlines, 222
    - task definitions, 221-222
  - user interfaces, HTAs (HTML Applications), 412-413
- designing scripts**
  - modules, creating, 63-69
  - task lists, creating, 56-58
  - tools, selecting, 58-63
- Desktop Management Task Force (DMTF), WMI (Windows Management Instrumentation), 283**
- DHCP (Dynamic Host Configuration Protocol), 325**
- dialog boxes**
  - “Are you sure?”, 104
  - Cancel buttons, 105
  - creating, 101
  - Script Package, 395
  - Starting, 22
- dialog builders, 101**
- digitally signing scripts, 402-405**
- Dim keyword, 89, 149**
- Dim statements, 80, 98**
- Dim sVariable( ) function, 149**
- directories, LDAP, fully qualified domain names, 253**
- DisableUser.vbs listing (5.1), 77-79**
- DisableUser2.vbs listing (5.2), 90-92**
- display options, MsgBox statement, 103**
- displaying messages, 59-60, 101-108**
  - complex messages, 105-106
  - WScript.Echo, 107-108
- DLLs (dynamic link libraries), 16, 76**
- DMTF (Desktop Management Task Force), WMI (Windows Management Instrumentation), 283**
- Do Until statement, 424**
- Do Until/Loop construct, 163-164**
- Do While/Loop construct, 162-163**
- Do/Loop Until construct, 163-164**
- Do/Loop While construct, 162-163**
- documentation, VBScript, downloading, 59**
- documentation (scripts), 48-53**
- domain attributes, LDAP provider, 258**
- domains, 257**
  - inactive users, finding, 464-469
  - inventorying script for, 343-355
  - lists of computers, targeting, 474-477
  - lockout policies, changing, 259
  - manipulating, 269
  - names, FQDNs (fully qualified domain names), 271
  - OUs (organizational units), 260
    - creating, 260-261
    - deleting, 263-264
    - modifying, 261-263
    - querying, 263
  - passwords, changing, 259
  - preloading, 264-269, 277-281
  - querying, 257-259
  - system information, collecting, 469-474
  - tasks, automating, 455, 480
  - time zones, 472
  - user creation, automating, 455-456, 459-463
  - users, targeting, 477-479
- DoObject subroutine, 476-477**
- DoObject\_User subroutine, 479**
- double quotation marks, strings, 129-130**
- doubles, 115**
- downloading VBScript documentation, 59**
- Dreamweaver, 410**
- Drive object (FSO), 194-197**
  - drive types, 196
  - properties, 195-196

**Drive property (FSO File object), 204**  
**drive types, Drive object (FSO), 196**  
**DriveLetter property (Drive object), 195**  
**drives, 196**  
    mapping, 60-61  
**DriveType property (Drive object), 195**  
**debugging scripts, syntax errors, 422-425**  
**Duncan, Larry, 61**  
**dynamic arrays, 149**  
**Dynamic Host Configuration Protocol (DHCP), 325**  
**dynamic link libraries (DLLs). See DLLs (dynamic link libraries)**

## E

**ECMAScript, 382**  
**editors, What-You-See-Is-What-You-Get editors, 410**  
**Elseif.vbs listing (10.1), 160-161**  
**Encapsulation, functions and subroutines, 373**  
**encoded script, 389**  
    detriments, 394  
    encrypted code, compared, 390  
    running, 393  
    writing, 390-393  
**encoding script, 397**  
**encrypted code, encoded code, compared, 390**  
**EnumNetworkDrive method (Network object), 176**  
**EnumPrinterConnection method (Network object), 177**  
**Environment method (Shell object), 183-184**  
**error-handling code, 70**  
**ErrorMsg Subroutine listing (5.3), 93**

**errors**  
    conditional errors, 70  
    logic errors, 70, 219  
    planning for, 70-72  
    scripts, identifying, 231-233  
    syntax errors, 70  
**errors, scripts, 421-422, 435**  
    logic errors, debugging, 425-435  
    syntax errors, preventing, 422-425  
**Escape( ) function, 130**  
**ESP (Evolved Script Packager), 394-397**  
**event handlers, 410**  
    HTAs (HTML Applications), wiring up, 418-420  
**event-driven programs, 410**  
**event-driven scripting, HTAs (HTML Applications), 410**  
**Evolved Script Packager (ESP), 394-397**  
**Excel spreadsheets, queries, ADO (Active Directory Objects), 455-463**  
**exclamation mark icon, 106**  
**Exec method (Shell object), 180-182**  
**ExecQuery( ) method, 285, 422**  
**Execute( ) method, 423**  
**executions, scripts, 155, 167-170**  
    conditional executions, 155-162  
    loops, 162-167  
**exiting loops, 166-167**  
**Exp( ) function, 119**  
**ExpandEnvironmentStrings method (Shell object), 184**  
**Expression, 410**  
**Extensible Markup Language (XML). See XML (Extensible Markup Language)**  
**extensions (file), WScript.exe associations, removing, 405**

## F

**file extensions, WScript.exe associations, removing, 405**

**File object (FSO), 194**

**File objects, FSO (FileSystemObject), 203-206**

methods, 204-206

properties, 204-206

**file servers, connecting to, WinNT provider, 249**

**file systems, FSO (FileSystemObject), 193, 209, 218**

BuildPath function, 209-210

Drive object, 194-197

File object, 194

File objects, 203-206

Folder object, 194

Folder objects, 198-203

folders, 197-198

GetTempName method, 210

log file scanner creation, 211-218

object libraries, 193-194

TaxStream object, 194

TextStream objects, 206-209, 212, 216

**File Transfer Protocol (FTP). See FTP (File Transfer Protocol)**

**FileProperties.vbs listing (12.4), 204**

**files**

batch files, 13, 20

CScript.exe executable, deleting, 405-406

FSO (FileSystemObject), 209, 218

BuildPath function, 209-210

File objects, 203-206

GetTempName method, 210

log file scanners, 211-218

TextStream objects, 206-209, 212, 216

open files, monitoring, 486-488

text files, 14

WScript.exe executable

deleting, 405-406

renaming, 406

**FileSystem property (Drive object), 195**

**FileSystemObject (FSO), 193, 209, 218, 247, 335, 347, 509**

BuildPath function, 209-210

Drive object, 194-197

drive types, 196

properties, 195-196

File objects, 194, 203-206

methods, 204-206

properties, 204-206

Folder objects, 194, 198-199

Attributes property, 200-202

methods, 198

properties as objects, 202-203

folders, 197-198

methods, 197

GetTempName method, 210

log file scanners, creating, 211-218

object libraries, 193-194

TaxStream object, 194

TextStream objects, 206-209, 212, 216

**FileSystemObject's GetFolder method, 99**

**FindOldUsers.vbs listing (28.2), 464-469**

**first script**

debugging, 35-36

running, 35

writing, 33-35

**Fixed drive type, 196**

**Folder objects, FSO (FileSystemObject), 194, 198-199**

Attributes property, 200-202

Folder objects, 198-203

methods, 197-198

properties as objects, 202-203

- FolderExists method (FSO), 197**
- For Each/Next construct, 165-166**
- For/Next construct, 164-165**
- forceLogoff domain attribute (LDAP provider), 258**
- Format commands, 223**
- FormatCurrency( ) function, 130, 142**
- FormatDateTime( ) function, 130, 141-142**
- FormatLogFileName function, writing, 223-227**
- FormatLogFileName Function listing (13.1), 225-227**
- FormatNumber( ) function, 130, 142**
- FormatPercent( ) function, 130, 141-142**
- formatting strings, 141-142**
- FQDNs (fully qualified domain names), 271**
- fractional values, numbers, 115**
- FreeSpace property (Drive object), 195**
- FROM property (WQL query), 290, 300**
- FrontPage, 410**
- FSO (FileSystemObject), 193, 209, 218**
  - BuildPath function, 209-210
  - Drive object, 194-197
    - drive types, 196
    - properties, 195-196
  - File object, 194
  - File objects, 203-206
    - methods, 204-206
    - properties, 204-206
  - Folder object, 194
  - Folder objects, 198-199
    - Attributes property, 200-202
    - methods, 198
    - properties as objects, 202-203
  - folders, 197-198
    - methods, 197
  - GetTempName method, 210
  - log file scanners, creating, 211-218
  - object libraries, 193-194
  - TaxStream object, 194
  - TextStream objects, 206-209, 212, 216
- FTP (File Transfer Protocol), 32**
- FullName method (Shortcut object), 187**
- fully qualified domain names, objects, LDAP directories, 253**
- fully qualified domain names (FQDNs). See FQDNs (fully qualified domain names)**
- Function keyword, 48**
- Function statement, 89, 118**
- functional code, HTAs (HTML Applications), writing, 411-412**
- functions, 39, 42-43, 76, 85, 99**
  - Abs( ), 126
  - as statements, 94
  - Asc( ), 118, 130
  - Atn( ), 118
  - BuildPath, 209-210
  - CBool( ), 126
  - CByte( ), 152
  - CCur( ), 126
  - CDbl( ), 126
  - Chr( ), 130
  - CInt( ), 126
  - CLng( ), 126
  - Cos( ), 118
  - CSng( ), 126
  - CStr( ), 130, 143
  - custom functions, 88-92
  - Date( ), 42, 85, 146, 227
  - DateAdd, 223
  - DateAdd( ), 147-148
  - DateDiff( ), 87, 467
  - DatePart( ), 118, 136, 146
  - Day( ), 146
  - Dim sVariable( ), 149

encapsulation, 373  
Escape( ), 130  
Exp( ), 119  
experimenting with, 139  
FormatCurrency( ), 130, 142  
FormatDateTime( ), 130, 141-142  
FormatNumber( ), 130, 142  
FormatPercent( ), 130, 141-142  
GetIP( ), 44, 67-68  
GetOSInfo( ), 340, 342, 344, 348, 351  
GetSPVer( ), 418  
Hour( ), 146  
input, defining, 223  
input parameters, 85-89  
InputBox( ), 108-110, 114, 216  
InStr( ), 44, 63, 67, 86, 131, 133-134, 143, 213  
InStrRev( ), 131, 134  
Int( ), 126  
intrinsic functions, 42, 88-92  
IsArray( ), 152  
IsMember( ), 46  
IsMemberOf function, 43  
IsMemberOf function( ), 43  
IsMemberOf( ), 43-44, 46, 67, 69  
Join( ), 151  
LCase( ), 131, 139, 217  
Left( ), 131, 133, 143  
Len( ), 131  
Log, 468  
Log( ), 119  
LTrim( ), 131  
main script  
    custom functions, 45-46  
    intrinsic functions, 47-48  
MakePW, 117  
Mid( ), 44, 63, 67, 118, 131, 143  
Minute( ), 146  
modular, 43  
Month( ), 146  
MonthName( ), 147  
MsgBox( ), 94, 101-110, 114, 491  
NameCase( ), 140  
Now( ), 146  
On Error Goto 0, 70  
output, defining, 223  
output values, 87-88  
Randomize, 119  
Replace( ), 131, 140-141, 143  
Right( ), 131, 134, 143  
Rnd( ), 119, 126  
Round( ), 509  
RTrim( ), 131  
Second( ), 146  
Sin( ), 119  
Space( ), 131  
Split( ), 150  
Sqr( ), 119  
StrComp( ), 131  
string manipulation, 130-131  
StrReverse( ), 131  
Tan( ), 119  
Time( ), 146  
Trim( ), 131  
UCase( ), 131, 136, 139-140  
Unescape( ), 131  
Weekday( ), 146  
WeekdayName( ), 147  
when to use, 45  
writing, 223-227, 340, 342  
Year( ), 146  
**future dates, working with, 147-148**

## G

**GC (Global Catalog) provider, ADSI, 254**  
**Get method, 263**  
     user information, querying, 273-274  
**Get method (objDomain), 248**  
**Get( ) method, 87**  
**GetEx method, 263**  
**GetFolder method (FSO), 197**  
**GetInfo method, 263**  
**GetInfo.vbs listing (6.1), 111-114**  
**GetIP( ) function, 44, 67-68**  
**GetMail( ) method, 96**  
**GetObject statement, 248**  
**GetOSInfo( ) function, 340-351**  
**GetOSInfo.vbs listing (20.2), 340-342**  
**GetParentFolderName method (FSO), 197**  
**GetSpecialFolder method (FSO), 197**  
**GetSPVer( ) function, 418**  
**GetTempName method, FSO (FileSystemObject), 210**  
**Go subroutine, 418**  
**graphical applications, 410**  
**graphical input, message boxes, asking for, 108-109**  
**Graphical User Interfaces (GUIs). See GUIs (Graphical User Interfaces)**  
**Group Policy object (GPO) Editor, 406**  
**groups, 271**  
     creating, 271  
         LDAP provider, 273  
         WinNT provider, 272  
     membership, checking, 61, 275-277  
**GUIs (graphical user interfaces), 409**  
     HTA (HTML Applications), designing, 412-413

## H

**hash rules, 400**  
**Hicks, Jeffrey, 409**  
**hierarchies, WMI (Windows Management Instrumentation), 283-285**  
     consumers, 285-286  
     providers, 285-286  
     versions, 286  
**hosts, script hosts, 14-16**  
**hot fixes, listing, 492-495**  
**HotKey method (Shortcut object), 187**  
**Hour( ) function, 146**  
**HTAs (HTML Applications), 23, 409-410, 420**  
     building, 410  
     event handlers, 410  
         wiring up, 418-420  
     event-driven scripting, 410  
     functional code, writing, 411-412  
     HTML code, converting, 414-416  
     Mshta.exe, 411  
     scheduled tasks, 410  
     script code, adding, 416, 418  
     security context, 409  
     user interface, designing, 412-413  
     writing, 411-416, 418-420  
**HTML (Hypertext Markup Language), 23**  
**HTML Applications (HTAs). See HTAs (HTML Applications)**  
**HTML code, HTAs (HTML Applications), converting, 414-416**  
**Hungarian notation, 85**

- |
- IconLocation method (Shortcut object), 187**
- icons**
  - critical icon, 107
  - exclamation mark icon, 106
  - message boxes, 103
  - question mark icon, 106
- Identifying possible errors listing (4.6), 71**
- IDEs (Integrated Development Environments), 29, 44**
- If/Then conditional execution statements, 156-161, 170-171**
  - nesting, 159-160
- If/Then constructs, 69**
- Ignore button, 107**
- IIS (Internet Information Services), 219**
- IIS log rotation tool, creating, 219**
  - candidate modules, identifying, 223
  - FormatLogFileName function, writing, 223-227
  - functions, writing, 223-227
  - main script, writing, 228-234
  - script, testing, 234-243
  - script design, 219-222
  - subroutines, writing, 223-227
- IIS provider, ADSI, 254**
- “impersonate” directives, queries, 315**
- inline code, 45**
- input**
  - command-line parameters, 110-114
  - functions, defining, 223
  - message boxes, asking for, 108-110
- input parameters, functions, 85-89**
- InputBox( ) function, 108-110, 114, 216**
- installation**
  - Script Encoder, 390
  - WMI (Windows Management Instrumentation), 290-291
- instances, WMI instances, associating, 321-325**
- InStr( ) function, 44, 63, 67, 86, 131-134, 143, 213**
- InStrRev( ) function, 131, 134**
- Int( ) function, 126**
- integers, 115**
- Integrated Development Environments (IDEs). See IDEs (Integrated Development Environments)**
- interaction, objects, 75**
- interfaces, 193**
- Internet Information Services (IIS), 219**
- Internet shortcuts, creating, 188**
- Internet Zone rules, 401**
- interpretation, scripting, 17**
- interpreted languages, VBScript, 14**
- interrelated classes**
  - WMI (Windows Management Instrumentation), 300, 317-320
  - WQL (WMI Query Language), 317
- intrinsic functions, 42**
  - custom functions, compared, 48, 88-92
  - main script, 47-48
- intrinsic subroutines, 44**
  - main script, 47-48
- inventories, domains, script for, 343-344, 347-352, 355**
- Inventory.vbs listing (29.6), 494-495**
- InventoryDomain.vbs listing (20.4), 344-350**
- InventoryDomain2.vbs listing (20.5), 352-355**
- IP addresses**
  - local IP addresses, attaining, 61-63
  - third octet, attaining, 63
- ipconfig command-line utility, 110**
- IsArray( ) function, 152**
- IsMember( ) function, 46**
- IsMemberOf( ) function, 43-46, 67, 69**
- IsReady property (Drive object), 195**

**J****Java Virtual Machines (JVMs), 17****JavaScript, 382****Join( ) function, 151****JSript, 382****JVMs (Java Virtual Machines), 17****K****keywords, 422**

Dim, 89, 149

Function, 48

Preserve, 150

Sub, 48

WHERE, 136

WQL (WMI Query Language), 303

**Kixtart, 439****L****languages**

strongly typed languages, 82

weakly typed languages, 82

**LCase( ) function, 131, 139, 217****LDAP (Lightweight Directory Access Protocol), 252**

directories, fully qualified domain names, 253

paths, 252

**LDAP provider, 271**

ADSI, connections, 248

domain attributes, 258

domains, querying, 257-259

group membership, checking, 275-277

groups, creating, 273

user settings, changing, 274-275

users

creating, 273

information queries, 273-274

**Left( ) function, 131-133, 143****Len command, 225****Len( ) function, 131****libraries**

object libraries, FSO (FileSystemObject), 193-194

script libraries, creating, 72

**Lightweight Directory Access Protocol (LDAP).****See LDAP (Lightweight Directory Access Protocol)****line continuation characters, 34, 79****line continuation characters (\\_), 337****listing**

hot fixes, 492, 494-495

remote shares, ADSI, 484-486

software, 492-495

**listings, 9**

2.1 (WhoHas.vbs Displays the User or Users Who Have a File Open), 33

3.1 (LoginScript.vbs Sample Logon Script), 40

3.2 (AddUsersFromXLS.vbs. Creates users from an Excel spreadsheet), 49-50

3.3 (AddUsersFromXLS.vbs. Creates users from an Excel spreadsheet), 50-53

4.1 (Logon script task list), 56

4.2 (Logon script task list v2), 56

4.3 (Logon script task list v3), 57

4.4 (Retrieve IP Address.vbs), 62

4.5 (LogonScript.vbs), 64-69

4.6 (Identifying possible errors), 71

5.1 (DisableUser.vbs), 77-79

5.2 (DisableUser2.vbs), 90-92

5.3 (ErrMsg Subroutine), 93

- 5.4 (TreeObject model script), 97
- 5.5 (RootFiles), 98-99
- 6.1 (GetInfo.vbs), 111-114
- 7.1 (MakePW.vbs), 117-118
- 7.2 (MakePW.vbs), 120-121
- 10.1 (Elseif.vbs), 160-161
- 10.2 (SelectCase.vbs), 161
- 10.3 (WhoHasFile.vbs), 167-170
- 11.1 (Logon.vbs), 178-179
- 11.2 (Shortcuts.vbs), 188-189
- 12.1 (NameDrives.vbs), 196
- 12.2 (Types.vbs), 199
- 12.3 (CheckFolder.vbs), 200-201
- 12.4 (FileProperties.vbs), 204
- 13.1 (FormatLogFileName Function), 225-227
- 13.2 (Log Rotation.vbs), 228-231
- 13.3 (LogRotation2.vbs), 232-234
- 13.4 (LogRotation3.vbs), 234-236
- 13.5 (LogRotation4.vbs), 237-239
- 13.6 (LogRotation5.vbs), 241-243
- 15.1 (PreLoad.vbs), 264-266
- 16.1 (CheckGroupMembership.vbs), 276-277
- 16.2 (PreloadDomain2.vbs), 278-280
- 17.1 (ShowNIC.vbs), 285
- 17.2 (CheckRoutes.vbs), 297
- 19.1 (SetDHCP.vbs), 328-330
- 19.2 (CompressAll.vbs), 330-333
- 20.1 (WizardCode.vbs), 337-339
- 20.2 (GetOSInfo.vbs), 340-342
- 20.3 (MainScript.vbs), 343-344
- 20.4 (InventoryDomain.vbs), 344-347
- 20.5 (InventoryDomain2.vbs), 352-355
- 21.1 (ShowProperties.vbs), 366-368
- 21.2 (ADSIDebug.vbs), 368-369
- 22.1 (WMIFunctions.wsc), 379-382
- 23.1 (ResetPW.vbs. An unencoded administrative script written in VBScript), 391-393
- 24.1 (Signer.wsf. This script signs another one), 402-404
- 25.1 (SampleHTA.hta), 419-420
- 27.1 (Logon1.vbs), 440-444
- 27.2 (Logon2.vbs), 444-445
- 27.3 (ADLogon1.vbs), 446-448
- 27.4 (ADLogon2.vbs), 448-451
- 27.5 (Logoff.vbs), 451-453
- 28.1 (AddUsers.vbs), 456-459
- 28.2 (FindOldUsers.vbs), 464-466
- 28.3 (CollectSysInfo.vbs), 469-471
- 28.4 (TargetingLists.vbs), 474-475
- 28.5 (TargetingAD.vbs), 477-479
- 29.1 (Shutdown.vbs), 482-483
- 29.2 (Shares.vbs), 484-485
- 29.3 (ListShares.vbs), 485-486
- 29.4 (WhoHas.vbs), 486-488
- 29.5 (Uninstall.vbs), 489-492
- 29.6 (Inventory.vbs), 494-495
- 30.1 (MassPass.vbs), 503-505
- 30.2 (UpdateServices.vbs), 505-508
- ListShares.vbs listing (29.3), 485-486**
- Live Syntax Checking, 422**
- loading domains, preloading, 264-269**
- local IP addresses, attaining, 61-63**
- Local name parameter (MapNetworkDrive method), 175**
- lockout policies, domains, changing, 259**
- lockoutDuration domain attribute (LDAP provider), 258**
- lockOutObservationWindow domain attribute (LDAP provider), 258**
- lockoutThreshold domain attribute (LDAP provider), 258**
- log file scanners, creating, FSO (FileSystemObject), 211-218**
- Log function, 468**

**log rotation tool, creating, 219**

- candidate modules, identifying, 223
- FormatLogFileName function, writing, 223-227
- functions, writing, 223-227
- main script, writing, 228-234
- script, testing, 234-236, 239-243
- script design, 219-222
- subroutines, writing, 223-227

**Log Rotation.vbs listing (13.2), 228-231****Log( ) function, 119****LogEvent method (Shell object), 184****logic errors, 70, 219****LoginScript.vbs sample logon script, 39, 42****LoginScript.vbs Sample Logon Script listing (3.1), 40****logoff scripts, 439**

- AD (Active Directory), 451-453

**Logoff.vbs listing (27.5), 451-453****Logon script task list listing (4.1), 56****Logon script task list v2 listing (4.2), 56****Logon script task list v3 listing (4.3), 57****logon scripts, 439**

- AD scripts, 440-446, 453-454
  - specific logon scripts, 446-451
- NT domains, calling, 446
- NT scripts, 440-446, 453-454

**Logon.vbs listing (11.1), 178-179****Logon1.vbs listing (27.1), 440-444****Logon2.vbs listing (27.2), 444-445****LogonScript.vbs listing (4.5), 64-69****LogRotation2.vbs listing (13.3), 232-234****LogRotation3.vbs listing (13.4), 234-236****LogRotation4.vbs listing (13.5), 237-239****LogRotation5.vbs listing (13.6), 241-243****long integers, 115****loops, 162-167**

- Do Until/Loop construct, 163-164
- Do While/Loop construct, 162-163
- Do/Loop Until construct, 163-164
- Do/Loop While construct, 162-163
- exiting, 166-167
- For Each/Next construct, 165-166
- For/Next construct, 164-165

**LTrim( ) function, 131**

## M

**main script, 44-45**

- custom functions and subroutines, 45-46
- decision making in, 48
- intrinsic functions, 47-48
- statements, 47-48
- testing, 234-243
- writing, 228-350

**main scripts, errors, identifying, 231-233****MainScript.vbs listing (20.3), 343-344****MakePW function, 117****MakePW.vbs listing (7.1), 117-118****MakePW.vbs listing (7.2), 120-121*****Managing Windows with VBScript and WMI, 2*****MapDrive routine, 68****MapDrive subroutine, 44, 67****MapNetworkDrive method (Network object), 175-176****mapping**

- drives, 60-61
- printers, 61

**MapPrinter command, 67****mass password changes, ADSI, performing, 503-505****MassPass.vbs listing (30.1), 503-505**

**mathematical operators, 116-118**

Boolean operators, 121, 156-157

**maxPwdAge domain attribute (LDAP provider), 258**

**Melissa virus, 399**

**membership, groups, checking, 61, 275-277**

**message boxes, 101**

buttons, 103

command-line input, asking for, 109-110

defaults, 103

graphical input, asking for, 108-109

icons, 103

input, asking for, 108-110

modality, 103

system icons, 102

**messages**

displaying, 59-60, 101-108

complex messages, 105-106

WScript.Echo, 107-108

nonerror messages, 106

nonwarning messages, 106

**methods**

COM objects, 95-96

Create, 260

Create method, 261

CreateShortcut, 187

CreateTextFile, 347

Delete, 263

DeleteMail, 96

ExecQuery( ), 285, 422

Execute( ), 423

FileSystemObject's GetFolder, 99

FSO (FileSystemObject)

CopyFolder, 197

CreateFolder, 197

DeleteFolder, 197

File objects, 204-206

Folder objects, 198

FolderExists, 197

folders, 197

GetFolder, 197

GetParentFolderName, 197

GetSpecialFolder, 197

MoveFolder, 197

Get, 263, 273-274

Get( ), 87

GetEx, 263

GetInfo, 263

GetMail( ), 96

Network object

AddPrinterConnection, 177

AddWindowsPrinterConnection, 177

ComputerName, 178

EnumNetworkDrive, 176

EnumPrinterConnection, 177

MapNetworkDrive, 175-176

RemoveNetworkDrive, 176

RemovePrinterConnection, 178

SetDefaultPrinter, 178

UserDomain, 178

UserName, 178

objDomain variable, 248

OpenTextFile, 468

Put, 263, 274-275

PutEx, 275

SetInfo, 260-263

Shell object

AppActivate method, 185

CreateShortcut method, 183

Environment method, 183-184

Exec method, 180-182

ExpandEnvironmentStrings method, 184

- LogEvent method, 184
- Popup method, 187
- RegDelete method, 184-185
- RegRead method, 184
- RegWrite method, 184-185
- Run method, 180-182
- SendKeys method, 185-187
- SpecialFolders method, 182-183
- Shortcut object, 187-188
- Shutdown, 308, 311, 483
- Uninstall, 492
- Microsoft Expression, 410**
- Microsoft FrontPage, 410**
- Microsoft TechNet, 2**
- Microsoft's MSDN Library, 420**
- Mid( ) function, 44, 63, 67, 118, 131, 143**
- minimum requirements, computers, 8**
- minPwdAge domain attribute (LDAP provider), 258**
- minPwdLength domain attribute (LDAP provider), 258**
- Minute( ) function, 146**
- modality, message boxes, 103**
- modifying scripts, 240-241**
- modular functions, 43**
- modularization, scripts, 373-377, 388**
  - detriments, 377
  - reasons for, 377
- modularization, scripts, WSCs (Windows Script Components), 378**
  - XML (Extensible Markup Language), 378-384, 387
- modules**
  - creating, 63-69
  - searching for, 373-377
- monospaced type, 8**
- Month( ) function, 146**
- MonthName( ) function, 147**

- Move method (Folder objects), 198**
- Move method (FSO File objects), 204**
- MoveFolder method (FSO), 197**
- MSDN Library (Microsoft), 420**
- MsgBox statement, 24-25, 84, 94, 101-108, 114, 170**
  - best practices, 106-107
  - display options, 103
  - message boxes, displaying, 101-108
  - return values, 104
- MsgBox( ) function, 94, 101-108, 114, 491**
  - input asking for, 108-110
  - messages, displaying, 101-108
- Mshhta.exe, HTAs (HTML Applications), 411**
- MSI packages, remote MSI packages, uninstalling, 489-492**
- multidimensional arrays, 149**

## N

- Name property (FSO File object), 204**
- NameCase( ) function, 140**
- NameDrives.vbs listing (12.1), 196**
- names, variables, 227**
- native scripts, 17**
- NDS (NetWare Directory Services) provider, ADSI, 254**
- nesting If/Then constructs, 159-160**
- net command, 60**
- .NET Framework, 18**
- NET TIME command, 445**
- network administrative scripts, 481, 495**
  - hot fixes, listing, 492-495
  - open files, monitoring, 486-488
  - remote computers, shutting down, 481, 483
  - remote MSI packages, uninstalling, 489-492

- remote shares, listing, 484-486
  - software, listing, 492-495
  - Network drive type, 196**
  - Network object (WSH), 173-175**
    - AddPrinterConnection method, 177
    - AddWindowsPrinterConnection method, 177
    - ComputerName method, 178
    - EnumNetworkDrive method, 176
    - EnumPrinterConnection method, 177
    - MapNetworkDrive method, 175-176
    - practical application, 178-179
    - RemoveNetworkDrive method, 176
    - RemovePrinterConnection method, 178
    - SetDefaultPrinter method, 178
    - UserDomain method, 178
    - UserName method, 178
  - Nimda virus, 399**
  - nonerror messages, 106**
  - nonwarning messages, 106**
  - notations, Hungarian notation, 85**
  - Notepad, 219**
    - scripting with, 25-26
  - Nothing statement, 251**
  - Now( ) functions, 146**
  - NT domains, VBScript logon scripts, calling, 446**
  - NT scripts, logon scripts, 440-446, 453-454**
  - numbers**
    - currency number types, 116
    - doubles, 115
    - fractional values, 115
    - integers, 115
    - singles, 115
    - VBScript, 115-116
    - whole numbers, 115
  - numeric data types, converting, 126-127**
  - numeric variables, arrays, 151-152**
  - NWCOMPAT provider, ADSI, 254**
- ## O
- objDomain variable, 248**
  - Object Browser (WMI), 291-292**
  - object creation scripts, ADSI object creation script, 500-502**
  - object deletion scripts, ADSI object creation script, 502-503**
  - object interfaces, 76**
  - object libraries, FSO (FileSystemObject), 193-194**
  - object query scripts, ADSI object creation script, 502**
  - object-oriented programming language, 75**
  - objects, 99, 189-190**
    - ADSI (Active Directory Services Interface), 247-249
    - ADSystem object, 179-180
    - COM objects, 94-98
      - collections, 96
      - methods, 95-96
      - properties, 95
      - sample object (Tree object), 97-98
      - scripting with, 98-99
    - FSO (FileSystemObject), 209, 218
      - Drive object, 194-197
      - File object, 194
      - File objects, 203-206
      - Folder object, 194
      - Folder objects, 197-203
      - TaxStream object, 194
      - TextStream objects, 206-209, 212, 216
    - fully qualified domain names, LDAP directories, 253
    - interaction, 75
    - names, 194
    - Network object, 173-175
      - AddPrinterConnection method, 177
      - AddWindowsPrinterConnection method, 177

ComputerName method, 178  
 EnumNetworkDrive method, 176  
 EnumPrinterConnection method, 177  
 MapNetworkDrive method, 175-176  
 practical application, 178-179  
 RemoveNetworkDrive method, 176  
 RemovePrinterConnection method, 178  
 SetDefaultPrinter method, 178  
 UserDomain method, 178  
 UserName method, 178  
**Shell object, 173, 180**  
 AppActivate method, 185  
 CreateShortcut method, 183  
 Environment method, 183-184  
 Exec method, 180-182  
 ExpandEnvironmentStrings method, 184  
 LogEvent method, 184  
 Popup method, 187  
 RegDelete method, 184-185  
 RegRead method, 184  
 RegWrite method, 184  
 Run method, 180  
 SendKeys method, 185-187  
 SpecialFolders method, 182-183  
**Shortcut object, 173, 187**  
 methods, 187  
 practical application, 188-189  
 properties, 187  
 TextStream, 347  
 WScript, 247  
 WScript object, 173-174  
**objTS variable, 424**  
**OLE DB provider, ADSI, 254**  
**On Error Goto 0 function, 70**  
**onClick event, 418**  
**oops resilience, 32**  
**open files, monitoring, WinNT provider, 486-488**

**OpenAsTextStream method (FSO File objects), 204**  
**OpenTextFile method, 468**  
**operators, 116-118**  
 Boolean operators, 121, 156-157  
 WQL (WMI Query Language)  
 Boolean operator, 301-302  
 comparison operator, 302-303  
**Option Explicit, syntax errors, preventing, 423-424**  
**Option Explicit command, 81**  
**Or Boolean operator, 156**  
**organizational units (OUs). See OUs (organizational units)**  
**OUs (organizational units), 248, 257, 260, 271**  
 creating, 260-261  
 deleting, 263-264  
 modifying, 261-263  
 querying, 263  
**oUser input parameter, 89**  
**outlining scripts, 222**  
**output, functions, defining, 223**  
**output values, functions, 87-88**

## P

**packaging script, 394-397**  
**parameters**  
 command-line parameters as input, 110-114  
 command-line scripts, parsing, 111-114  
 input parameters, functions, 85-87  
 MapNetworkDrive method, 175  
**ParentFolder property (FSO File object), 204**  
**parsing parameters, command-line scripts, 111-114**  
**Password parameter (MapNetworkDrive method), 175**

**passwords**

- creating, 117-121
- domains, changing, 259
- mass password changes, ADSI, 503-505
- scripts, placing in, 176

**past dates, working with, 147-148****Path property (Drive object), 195****Path property (FSO File object), 204****path rules, 401****paths, LDAP, 252****PCs (personal computers), minimum requirements, 8****PFE (Programmer's File Editor), 37**

- scripting with, 26-27

**ping command-line utility, 110****policies**

- software restriction policies, 400-401
- WSH trust policy, implementing, 406

**Popup method (Shell object), 187****PreLoad.vbs listing (15.1), 264-269****PreloadDomain2.vbs listing (16.2), 278-280****preloading domains, 264-269, 277-281****Preserve keyword, 150****PrimalScript, 29-31**

- Live Syntax Checking, 422
- Snippets, 377
- syntax errors, preventing, 422-423
- TLB Browser, 189
- WMI Wizard, 336-337, 340

**PrimalScript Wizard, WQL queries, writing, 308, 310-312****printers, mapping, 61****procedural programs, 410****programs, event-driven programs, 410****Progid parameter (WSCs), 383****programming**

- computer programming, 75
- typographical elements, 8-9

**programming books, 76****programming languages**

- strongly typed languages, 82
- weakly typed languages, 82

**programs, procedural programs, 410****properties**

- COM objects, 95
- Drive object (FSO), 195-196
- FSO (FileSystemObject)
  - File objects, 204-206
  - objects, 202-203
- WQL (WMI Query Language), 300-301

**providers**

- ADSI, 254-255
  - GC (Global Catalog) provider, 254
  - IIS provider, 254
  - LDAP provider, 248, 252-254
  - NWCOMPAT provider, 254
  - OLE DB provider, 254
  - WinNT provider, 248-252
- LDAP provider, 271
  - ADSI, connections, 248
  - domain attributes, 258
  - domains, querying, 257-259
  - group creation, 273
  - group membership, checking, 275-277
  - user creation, 273
  - user settings, changing, 274-275
- NDS (NetWare Directory Services) provider, 254
- WinNT provider
  - ADSI connections, 248-252
  - domains, reconfiguring, 259
  - group creation, 272
  - LDAP, 252-254
  - open file monitoring, 486-488
  - user creation, 272
- WMI (Windows Management Instrumentation), 285-286

**provider names, ADSI, 248**  
**pseudocode, 56**  
**Put method, 263**  
    user settings, changing, 274-275  
**Put method (objDomain), 248**  
**PutEx method, 275**  
**pwdHistoryLength domain attribute (LDAP provider), 258**

## Q

### queries

ADSI queries, debugging, 366-370  
Excel spreadsheets, ADO (Active Directory Objects), 455-463  
“impersonate” directive, 315  
testing, WMI (Windows Management Instrumentation), 294-295  
WMI queries  
    debugging, 358-370  
    testing, 325-326  
    writing, 325-333  
WQL (WMI Query Language), 299-301  
    associations, 317  
    associators, 303  
    Boolean operators, 301-302  
    comparison operators, 302-303  
    complex information, 317  
    determining, 304-305  
    keywords, 303  
    properties, 300-301  
    references, 303  
    results, 312-314  
    testing, 306-308  
    writing, 308-315  
WQL queries  
    ASSOCIATORS OF query, 321-325

REFERENCES OF query, 323-325

WMI (Windows Management Instrumentation), 289-290

writing, WMI (Windows Management Instrumentation), 294-295

**query scripts, WMI (Windows Management Instrumentation), creating, 497-499**

### querying

domains, 257-259

OUs (organizational units), 263

user information, Get method, 273-274

**question mark icon, 106**

## R

**RAM Disk drive type, 196**

**Randomize function, 119**

**RealTimePublishers.com, 2**

**ReDim statement, 149**

**references, WQL (WMI Query Language), 303**

REFERENCES OF query, WMI instances,  
    associating, 323-325

**RegDelete method (Shell object), 184-185**

**RegRead method (Shell object), 184**

**RegWrite method (Shell object), 184**

**Remotable parameter (WSCs), 383**

### remote computers

connecting to, WMI (Windows Management Instrumentation), 287-289

shutting down, WMI (Windows Management Instrumentation), 481, 483

**remote MSI packages, uninstalling, WMI (Windows Management Instrumentation), 489-492**

**Remote name parameter (MapNetworkDrive method), 175**

**remote shares, listing, ADSI, 484-486**

**Removable drive type, 196**

**RemoveNetworkDrive** method (Network object), 176

**RemovePrinterConnection** method (Network object), 178

renaming WScript.exe, 406

**Replace( )** function, 131, 140-141, 143

requirements, computers, 8

**ResetPW.vbs** listing (23.1), 391-393

results, script testing, analyzing, 236

**Retrieve IP Address.vbs** listing (4.4), 62

**Retry** button, 107

return values, MsgBox statement, 104

reusable code, 31

reusing code, 424-425

**Right** command, 224

**Right( )** function, 131, 134, 143

**Rnd( )** function, 119, 126

**RootFiles** listing (5.5), 98-99

**RootFolder** property (Drive object), 195

**Round( )** function, 509

**routines, subroutines**

- encapsulation, 373
- event handlers, 410
- writing, 223-227, 340-342

**RTrim( )** function, 131

**rules, software, 400**

**Run** method (Shell object), 180

**running**

- command-line scripts, 111
- encoded script, 393
- first script, 35
- scripts, WSH (Windows Script Host), 21-23
- signed scripts, 404-405

## S

**safe scripting, implementing, 405-406**

**SAM (Security Accounts Manager), 247, 271**

**SampleHTA.hta** listing (25.1), 419-420

**SAPIEN Primal Script**

- syntax errors, preventing, 422-423
- TLB Browser, 189

**ScanLog.vbs** listing (12.6), 211-214

**ScanLog2.vbs** listing (12.7), 215-218

**scheduled tasks, HTAs (HTML Applications), 410**

**Scientific mode, Calculator, 124**

**scripts**

- query scripts, WMI (Windows Management Instrumentation), 497-499
- update scripts, WMI (Windows Management Instrumentation), 499-500

**scr files, WScript.exe associations, removing, 405**

**script**

- complex WMI queries, writing, 325-333
- encoded script, 389-390
  - detriments, 394
  - running, 393
  - writing, 390-393
- encoding, 397
- encrypted code, 390
- script packaging, 394-397

**script editors**

- Notepad, 26
- PFE (Programmer's File Editor), 26-27
- VBScript editors, 27-28
- VBScript IDEs, 29-33
- What-You-See-Is-What-You-Get editors, 410

**Script Encoder, 389**

- encoded script
  - detriments, 394
  - running, 393
  - writing, 390-393
- installing, 390

**script extensions, Windows XP, 399**

**script hosts, 14-16**

**script libraries, creating, 72**

**Script Package dialog box, 395**

**script packagers, 394-397**

**SCRIPT tags, 392**

**script-based viruses, 399**

**scripting, 13-14**

- candidate modules, identifying, 223
- dangers of, 399-400
- errors, planning for, 70-72
- event-driven scripting, HTAs (HTML Applications), 410
- FormatLogFileName function, writing, 223-227
- functions, writing, 223-227
- interpretation, 17
- main script, writing, 228-234
- Notepad, 25-26
- PFE (Programmer's File Editor), 26-27
- safe scripting, implementing, 405-406
- security, 405-406
- security issues, 18-19
- strings, double quotation marks, 129-130
- subroutines, writing, 223-227
- typographical elements, 8-9
- VBScript Editors, 27-28
- VBScript IDEs, 29-33
- Windows administrators, 13
- with objects, 98-99

**scripting languages, ActiveX, 16-18**

**ScriptingAnswers.com, 2, 14, 420**

**ScriptingOutpost.com, 377**

**ScriptingTraining.com, 2**

**Scriptomatic, WQL queries, writing, 308-312**

**Scriptomatic tool, 340**

**Scriptomatic tool (WMI), 292-293**

**scripts, 2, 13**

- administrative scripts, 455, 480
  - inactive user discovery, 464-469
  - system information collection, 469-474
  - templates, 474-479
  - user creation automation, 455-463
- ADSI mass password changes, performing, 503-505
- ADSI object creation scripts, creating, 500-502
- ADSI object deletion scripts, creating, 502-503
- ADSI object query scripts, creating, 502
- bugs, 421-422, 435
  - logic errors, 425-435
  - syntax errors, 422-425
- checking, 14
- command-line scripts, 23-25
  - execution of, 110
  - parsing parameters, 111-114
  - running, 111
- comments, 48-53
- completing, 241
- debugging, 357-358
  - logic errors, 425-435
  - syntax errors, 422-425
- decision making in, 48
- deployment, 17
- designing, 55, 219-220, 335-340
  - fact gathering, 220-221
  - module creation, 63-69
  - script outlines, 222
  - task definitions, 221-222
  - task lists, 56-58
  - tool selection, 58-63
- digitally signing, 402-405
- documentation, 48-53
- errors, identifying, 231-233

- executions, 155, 167-170
  - conditional executions, 155-162
  - loops, 162-167
- first script
  - debugging, 35-36
  - running, 35
  - writing, 33-35
- functions, 39-43
- HTAs (HTML Applications)
  - adding, 416-418
  - writing, 411-420
- logoff scripts, 439
  - AD scripts, 451-454
- logon scripts, 439
  - AD scripts, 440-454
  - NT scripts, 440-454
- main script, 44-45
  - custom functions and subroutines, 45-46
  - intrinsic functions, 47-48
  - statements, 47-48
  - writing, 343-350
- modifying, 240-241
- modularization, 373-377, 388
  - detriments, 377
  - reason for, 377
  - WSCs (Windows Script Components), 378-384, 387
- modularizing, 63-69
- native scripts, 17
- network administrative scripts, 481, 495
  - listing hot fixes, 492-495
  - listing software, 492-495
  - open file monitoring, 486-488
  - remote computer shutdowns, 481-483
  - remote MSI package uninstalls, 489-492
  - remote share listings, 484-486
- passwords, placing in, 176
- polishing, 243
- pseudocode, 56
- reusing, 424-425
- running, WSH (Windows Script Host), 21-23
- script-based viruses, 399
- script hosts, 14-16
- service accounts, updating, 505-508
- shutdown scripts, 451
- signed scripts, 402-405
  - running only, 404-405
- statements, 44
- subroutines, 39, 43-44
- testing, 234-236, 239-241-243, 351-352, 355
- text-based scripts, 17
- VBScript, typical scripts, 39, 42
- Windows 9x scripts, execution, 440
- WMI (Windows Management Instrumentation) scripts
  - testing, 297
  - writing, 293-297
- WQL queries, writing, 308-312
- sDate input parameter, 89**
- SearchScripting.com, 2**
- Second( ) function, 146**
- security**
  - antivirus software, 406
  - scripting, 18-19
    - dangers of, 399-400
    - safe scripting, 405-406
  - scripts
    - digitally signing, 402-405
    - signed scripts, 404-405
  - software restriction policies, 400

- Windows Server 2003, improvements to, 400-402
- Windows XP, improvements to, 400-402
- security, HTAs (HTML Applications), security context, 409
- Security Accounts Manager (SAM). *See* SAM (Security Accounts Manager)
- security identifiers (SIDs). *See* SIDs (security identifiers)
- SELECT property (WQL), 300
- SELECT property (WQL query), 290
- Select/Case conditional execution statements, 161-162
- Select/Case construct, 48
- SelectCase.vbs listing (10.2), 161
- SendKeys method (Shell object), 185-187
- SerialNumber property (Drive object), 195
- Server 2003 (Windows), security improvements to, 400-402
- Server Message Blocks (SMBs), 445
- service accounts, updating, 505-508
- SetDefaultPrinter method (Network object), 178
- SetDHCP.vbs listing (19.1), 328-330
- SetInfo method, 260, 263
- SetInfo method (objDomain), 249
- settings, users, changing, 274-275
- ShareName property (Drive object), 195
- Shares.vbs listing (29.2), 484-485
- Shell object, 173, 180
  - AppActivate method, 185
  - CreateShortcut method, 183
  - Environment method, 183-184
  - ExpandEnvironmentStrings method, 184
  - LogEvent method, 184
  - Popup method, 187
  - RegDelete method, 184-185
  - RegRead method, 184
  - RegWrite method, 184
  - Run method, 180-182
  - SendKeys method, 185-187
  - SpecialFolders method, 182-183
- Shortcut object, 173, 187
  - methods, 187
  - practical application, 188-189
  - properties, 187
- shortcuts, creating, 188
- Shortcuts.vbs listing (11.2), 188-189
- ShortName property (FSO File object), 204
- ShowNIC.vbs listing (17.1), 285
- ShowProperties.vbs listing (21.1), 366-368
- Shutdown method, 308, 311, 483
- shutdown scripts, 451
- Shutdown.vbs listing (29.1), 482-483
- shutting down, remote computers, WMI (Windows Management Instrumentation), 481-483
- SIDs (security identifiers), 58
- signed scripts, running only, 404-405
- Signer.wsf. This script signs another one listing (24.1), 402-404
- signing scripts, 402-405
- Sin( ) function, 119
- singles, 115
- Size property (FSO File object), 204
- Sleep command, 186
- SMBs (Server Message Blocks), 445
- SMS (Systems Management Server), system information, collecting, 469-474
- Snippets, 377
- snippets, reusing, 424-425
- software
  - listing, 492-495
  - rules, 400
- software restriction policies, 400-401
- Space( ) function, 131
- spaghetti debugging, 425-426
- SpecialFolders method (Shell object), 182-183
- Split( ) function, 150
- SQL (Structured Query Language), 299

- Sqr( ) function, 119**
  - standard shortcuts, creating, 188**
  - Starting dialog box, 22**
  - statements, 44, 92**
    - conditional execution statements
      - If/Then, 156-161, 170-171
      - Select/Case, 161-162
    - Const, 80
    - control-of-flow statements, 155
    - custom subroutines, compared, 48
    - Dim, 80, 98
    - Do Until, 424
    - Function, 89, 118
    - functions, acting as, 94
    - GetObject, 248
    - main script, 47-48
    - MsgBox, 84, 94, 101-108, 114, 170
      - best practices, 106-107
      - display options, 103
      - return values, 104
    - Nothing, 251
    - ReDim, 149
    - Sub, 93
    - WinNT://, 248
  - StdIn, command-line input, 109-110**
  - StrComp( ) function, 131**
  - strComputer variable, 418**
  - strings, 129**
    - changing, 139-141
    - concatenating, 135-139
    - data types, converting to, 142-143
    - double quotation marks, 129-130
    - formatting, 141-142
    - manipulating, 143-144
      - functions, 130-131
    - substrings, 131-134
    - VBScript, 129-131
  - strName variable, 476**
  - strongly typed languages, 82**
  - strResult variable, 418**
  - StrReverse( ) function, 131**
  - Structured Query Language (SQL), 299**
  - Sub keyword, 48**
  - Sub statement, 93**
  - subroutines, 39, 43-44, 90-92, 99**
    - custom subroutines, creating, 92-94
    - DoObject, 476-477
    - DoObject\_User, 479
    - encapsulation, 373
    - event handlers, 410
      - HTA (HTML Applications), 418-420
    - intrinsic subroutines, 44
    - main script
      - custom subroutines, 45-46
      - intrinsic subroutines, 47-48
    - MapDrive, 44, 67
    - when to use, 45
    - WorkWithObject, 478
    - writing, 223-227, 340, 342
  - substrings, 131-134**
  - syntax, 75**
  - syntax errors, 70**
    - preventing, 422-425
  - system icons, message boxes, 102**
  - Systems Management Server (SMS). See SMS (System Management Server)**
- ## T
- Tan( ) function, 119**
  - tangents, 115**
  - TargetingAD.vbs listing (28.5), 477-479**
  - TargetingLists.vbs listing (28.4), 474-477**

**TargetPath method (Shortcut object), 187**

**task lists, creating, 56-58**

**tasks**

domains, automating, 455, 480

scripts, defining, 221-222

**TaxStream object (FSO), 194**

**TechNet (Microsoft), 2**

**templates, administrative scripts, 474-479**

**testing**

complex WMI queries, 325-326

queries, WMI (Windows Management Instrumentation), 294-295

scripts, 234-355

WMI (Windows Management Instrumentation) scripts, 297

WQL queries, 306-308

**text files, 14**

**text-based scripts, 17**

**TextStream objects, 347**

FSO (FileSystemObject), 206-209, 212, 216

**third octets, IP addresses, attaining, 63**

**time zones, domains, 472**

**Time( ) function, 146**

**times**

getting, 146-147

VBScript, 145-148

**TLB Browser, 189**

**tools**

Script Encoder, 390

selecting, script design, 58-63

WMI (Windows Management Instrumentation), 291-292

Object Browser, 291-292

Scriptomatic tool, 292-293

**TotalSize property (Drive object), 196**

**tracert command-line utility, 110**

**TreeObject model script listing (5.4), 97**

**Trim( ) function, 131**

**trust policies, WSH trust policy, implementing, 406**

**Type property (FSO File object), 204**

**Types.vbs listing (12.2), 199**

**typographical elements, programming, 8-9**

## U

**UCase( ) function, 131, 136, 139-140**

**UNC (Universal Naming Convention), 168**

**UNCs (universal naming conventions), 66**

**underscore (\_) character, 79**

**underscore characters, 106**

**Unescape( ) function, 131**

**Uninstall method, 492**

**Uninstall.vbs listing (29.5), 489-492**

**uninstalling remote MSI packages, 489-492**

**Universal (Greenwich) time, 472**

**Universal Naming Convention (UNCs), 66, 168**

**Unknown drive type, 196**

**Update profile parameter (MapNetworkDrive method), 175**

**update scripts, WMI (Windows Management Instrumentation), creating, 499-500**

**UpdateServices.vbs listing (30.1), 505-508**

**updating service accounts, WMI, 505-508**

**user input, validating, 69**

**user interfaces, HTAs (HTML Applications), designing, 412-413**

**UserDomain method (Network object), 178**

**UserName method (Network object), 178**

**Username parameter (MapNetworkDrive method), 175**

**users, 271**  
 creating, 271  
   automating, 455-463  
   LDAP provider, 273  
   WinNT provider, 272  
 inactive users, finding, 464-469  
 information, querying, 273-274  
 settings, changing, 274-275  
 targeting, AD (Active Directory), 477-479

**utilities, command-line utilities, 110**

## V

**validating, user input, 69**

**values, constants, 466**

**variables, 76, 79, 99**

  arrays, 145, 148, 152  
     creating, 149-151  
     manipulating, 149-151  
     multidimensional arrays, 149  
     numerical variables, 151-152  
     VBScript, 149  
 bytes, 152  
 constants, compared, 80  
 data, assigning to, 82-83  
 data coercion, 83-85  
 data types, 82  
 declaring, 79-81  
 dynamic arrays, 149  
 names, 227  
 objDomain, 248  
 objTS, 424  
 strComputer, 418  
 strName, 476  
 strResult, 418  
 vPwd, 118

**VBA (Visual Basic for Applications), 400**

**vbs files, WScript.exe associations, removing, 405**

**VBScript (Visual Basic, Scripting Edition), 1, 13, 16-18**

  advanced arithmetic, 118-121  
 arrays, 149  
 as interpreted language, 14  
 basic arithmetic, 116-118  
 Boolean math, 121-126  
 bytes, 152  
 complex queries, writing, 326-333  
 data types, converting, 127  
 dates, 145-148  
 documentation, downloading, 59  
 input, asking for, 108-110  
 interactive capabilities, limitations, 101  
 introduction of, 1  
 messages, displaying, 101-108  
 numbers, 115-116  
 numeric data types, converting, 126-127  
 procedural programs, 410  
 scripts, typical scripts, 39, 42  
 strings, 129-131  
 times, 145-148  
 vulnerabilities, 1  
 Windows PowerShell, compared, 19-20  
 WQL queries, writing, 308-312  
 WScript.exe, 37

**VBScript Editor**

  scripting with, 27-28  
 syntax errors, preventing, 422-423

**VBScript IDEs, scripting with, 29-33**

**VbsEdit, 27-28**

**Version parameter (WSCs), 383**

**versions, WMI (Windows Management Instrumentation), 286**

**virtual private network (VPN) adapters, 284**

**viruses**

- antivirus software, 406
- script-based viruses, 399

**Visual Basic, Scripting Edition (VBScript). See VBScript (Visual Basic, Scripting Edition)**

**Visual Basic for Applications (VBA), 400**

**Visual Query Builder, 32**

**Visual Studio, 101**

**VolumeName property (Drive object), 196**

**vPasswd variable, 118**

**VPN (virtual private network) adapters, 284**

**W**

**WBEM (Web-Based Enterprise Management), 286**

**Wbemtest.exe tool**

- complex queries, testing, 325-326
- WMI (Windows Management Instrumentation) queries, 286-289
  - debugging, 358-366
  - WQL queries, 289-290

**weakly typed languages, 82**

**Web-Based Enterprise Management (WBEM), 286**

**Weekday( ) function, 146**

**WeekdayName( ) function, 147**

**What-You-See-Is-What-You-Get editors, 410**

**Where clause, WMI instances, associating, 324-325**

**WHERE keyword, 136**

**WHERE property (WQL query), 290, 301**

**WhoHas.vbs**

- debugging, 35-36
- running, 35
- writing, 33-35

**WhoHas.vbs listing (2.1), 33**

**WhoHas.vbs listing (29.4), 486-488**

**WhoHasFile.vbs listing (10.3), 167-170**

**whole numbers, 115**

**Win32 classes, WMI (Windows Management Instrumentation), 284**

**Win32\_OperatingSystem class (WMI), 336-337, 349, 482-483**

**Windows 9x scripts, execution, 440**

**Windows administrators, scripting, 13**

**Windows Calculator, Scientific mode, 124**

**Windows domains**

- inactive users, finding, 464-469
- lists of computers, targeting, 474-477
- system information, collecting, 469-474
- tasks, automating, 455, 480
- user creation, automating, 455-456, 459-463
- users, targeting, 477-479

**Windows Management Instrumentation (WMI). See WMI (Windows Management Instrumentation)**

**Windows PowerShell, VBScript, compared, 19-20**

**Windows PowerShell: TFM, 19**

**Windows Script Components (WSCs). See WSCs (Windows Script Components)**

**Windows Script Host (WSH). See WSH (Windows Script Host)**

**Windows Server 2003, security improvements to, 400-402**

**Windows XP**

- default script extensions, 399
- security, rules, 400
- security improvements to, 400-402

**WindowState method (Shortcut object), 187**

**WinNT, 271**

**WinNT provider**

- ADSI, 249-252
  - connections, 248
  - file server connections, 249
- domains, reconfiguring, 259
- groups, creating, 272
- LDAP, 252-254
- open files, monitoring, 486-488
- users
  - creating, 272
  - information queries, 273-274

**WinNT:// statement, 248****WizardCode.vbs listing (20.1), 337, 339****wizards**

- code wizard, 32
- WMI Wizard, 336-337

**WMI (Windows Management Instrumentation), 14, 145, 173, 283, 286, 299, 335, 355, 497, 508**

- ACEs (access control entries), 300
- associations, 319
  - instances, 321-325
- CIM (Common Information Model), 284
- classes, 284-285, 305
  - finding, 293-294
  - interrelated classes, 300, 317-319
  - Win32\_OperatingSystem, 336-337, 482-483
  - Win32\_OperatingSystem class, 349
- complex information, 300, 317-319
- DMTF (Desktop Management Task Force), 283
- functions, writing, 340, 342
- hierarchy, 283-285
  - consumers, 285-286
  - providers, 285-286
  - versions, 286
- installing, 290-291

**queries**

- debugging, 358-366, 370
- testing, 294-295, 325-326
- writing, 294-295, 325-333
- query scripts, creating, 497-499
- remote computers
  - connecting to, 287-289
  - shutting down, 481-483
- remote MSI packages, uninstalling, 489-492
- scripts
  - debugging, 357-358
  - designing, 335-340
  - testing, 297, 351-355
  - writing, 293-297, 343-344, 347-350
- service accounts, updating, 505-508
- subroutines, writing, 340, 342
- tools, 291-292
  - Object Browser, 291-292
  - Scriptomatic tool, 292-293
- update scripts, creating, 499-500
- Wbemtest.exe tool, 286-289
  - WQL queries, 289-290
- Win32, 284
- WQL (WMI Query Language), 299-301
  - associations, 317
  - associators, 303
  - Boolean operators, 301-302
  - classes, 317
  - comparison operators, 302-303
  - complex information, 317
  - keywords, 303
  - properties, 300-301
  - queries, 304-315
  - references, 303

**WMI Wizard, 336-337****WMIFunctions.wsc listing (22.1), 379-387**

**WorkingDirectory method (Shortcut object), 187**

**WorkWithObject subroutine, 478**

**WQL (WMI Query Language), 299-301**

associations, 317

associators, 303

Boolean operators, 301-302

classes, interrelated classes, 317

comparison operators, 302-303

complex information, 317

keywords, 303

properties, 300-301

queries

debugging, 358-366

determining, 304-305

results, 312-314

testing, 306-308, 325-326

writing, 308-315, 325-333

references, 303

WMI instances, associating, 321-325

**WQL queries, WMI (Windows Management Instrumentation), 289-290**

**writing**

complex WMI queries, 325

VBScript, 326-333

encoded script, 390-393

first script, 33-35

functions, 223-227, 340-342

FormatLogFileName function, 223-227

HTAs (HTML Applications), 411-420

main script, 228-234, 343-350

queries, WMI (Windows Management Instrumentation), 294-295

scripts, WMI (Windows Management Instrumentation) scripts, 293-297

subroutines, 223-227, 340-342

WQL queries, 308-315

**WScript objects, 173-174, 247**

**WScript.ADSystemInfo object, 179-180**

**WScript.Echo, 114, 170, 213**

command-line output, 110

message boxes, displaying, 107-108

**WScript.Echo command, 213-214**

**WScript.exe (VBScript), 37**

deleting, 405-406

file associations, removing, 405

renaming, 406

**WScript.Network object, 174-175**

AddPrinterConnection method, 177

AddWindowsPrinterConnection method, 177

ComputerName method, 178

EnumNetworkDrive method, 176

EnumPrinterConnection method, 177

MapNetworkDrive method, 175-176

practical application, 178-179

RemoveNetworkDrive method, 176

RemovePrinterConnection method, 178

SetDefaultPrinter method, 178

UserDomain method, 178

UserName method, 178

**WScript.Shell object, 180**

AppActivate method, 185

CreateShortcut method, 183

Environment method, 183-184

Exec method, 180-182

ExpandEnvironmentStrings method, 184

LogEvent method, 184

Popup method, 187

RegDelete method, 184-185

RegRead method, 184

RegWrite method, 184

Run method, 180

SendKeys method, 185-187

SpecialFolders method, 182-183

**WScript.Shortcut object, 187**

methods, 187  
 practical application, 188-189  
 properties, 187

**WScript.StdIn, command-line input, 109-110****WSCs (Windows Script Components), 378, 388**

registration, 383  
 XML (Extensible Markup Language), 378,  
 382-384, 387

**wsf files, WScript.exe associations,  
removing, 405****WSH (Windows Script Host), 15-16, 20-23, 173,  
390**

ADSystemInfo object, 179-180  
 Network object, 173-175  
   AddPrinterConnection method, 177  
   AddWindowsPrinterConnection  
   method, 177  
   ComputerName method, 178  
   EnumNetworkDrive method, 176  
   EnumPrinterConnection method, 177  
   MapNetworkDrive method, 175-176  
   practical application, 178-179  
   RemoveNetworkDrive method, 176  
   RemovePrinterConnection method, 178  
   SetDefaultPrinter method, 178  
   UserDomain method, 178  
   UserName method, 178  
 Shell object, 173, 180  
   AppActivate method, 185  
   CreateShortcut method, 183  
   Environment method, 183-184  
   Exec method, 180-182  
   ExpandEnvironmentStrings method, 184  
   LogEvent method, 184  
   Popup method, 187

RegDelete method, 184-185  
 RegRead method, 184  
 RegWrite method, 184  
 run method, 180  
 SendKeys method, 185-187  
 SpecialFolders method, 182-183  
 Shortcut object, 173, 187  
   methods, 187  
   practical application, 188-189  
   properties, 187  
 trust policy, implementing, 406  
 WScript object, 173-174

**WshNamed object, parameters, parsing,  
111-114****X-Z****XML (Extensible Markup Language), 32**

WSCs (Windows Script Components), 378,  
 382-384, 387

**Year( ) function, 146**