INDEX

Symbols

- **# (octothorpes), 44**
- **.ini files, case study, 332-333**

A

- **aborting**
  - Do loop, 88
  - For...Next loop, 84-86
- **Abs() function, 48**
- **absolute value of a number, returning, 48**
- **Access object class modules, 15**
- **Access SQL. *See* SQL**
- **accessing VBE, 11**
- **AccessObject object, properties, 214-217**
- **ACE (Access Connective Engine), 229**
- **active control, 144**
- **adding**
  - to combo box, case study, 138-140
  - unlisted values to combo boxes, 159-167
- **AddNew method, data, inserting**
  - ADO object model, 292-293
  - DAO object model, 291-292
- **ADO (ActiveX Data Objects) object model**
  - AddNew method, inserting data, 292-293
  - connecting to data source, 275-278
  - Delete method, 307-308
  - Execute method, inserting data, 290-291
  - Filter property, 302-304
  - Find method, 299-301
  - object hierarchy, 232-234
  - Recordsets, opening, 284-288
  - Seek method, 301-302
- **ADOX (ActiveX Data Objects Extensions for Data Definition Language and Security), 234**
  - databases, creating, 243-246
  - fields, creating, 255-259
  - indexes, creating, 261-262
  - object hierarchy, 235-236
  - queries, creating, 269-271
  - relationships, creating, 265-266
  - tables, creating, 249-251
- **AllForms collection, 213**
- **annuities**
  - payments, calculating, 51
  - periodic interest, calculating, 51
- **ANSI (American National Standard Institute), 373**
- **API calls, 359-362**
  - files, browsing, 365-372
  - FindExecutable API, 364
  - FindWindow API, 362-363
  - Network Login ID, capturing, 363
- **application case study, tracking current users, 126-127**
- **application collections, 211-212**
- **application lifetime, 122**
- **arguments**
  - delimiting, 72
  - ExportXML method, 350-352
  - Open function, 324-325
  - OpenArgs, 133
  - OpenDatabase method, 273
  - OpenRecordset method, 280
  - OpenReport method, 188-189
    - OpenArgs, 190-191
  - optional, 71-72
  - passing by reference, 72
  - passing by value, 72-73
  - specifying, 135
  - syntax, 71
- **arithmetic, performing on date values, 42-43**
- **arrays, 93**
  - dynamic, 98
    - Erase statement, 98-99
    - ReDim statement, 98
  - elements, 93, 95
    - assigning, 95-96
    - specifying, 96
  - fixed-size, declaring, 94
  - index values, 94
    - Option Base statement, 95
  - multidimensional, 96-98
- **Asc() function, 52**

assigning
- array elements, 95-96
- data type to functions, 70
- values to object properties, 104

attachment fields, 229

audit trail case study, 184-185

automation, 335
- objects, creating
  - CreateObject function, 338
  - early binding, 339-340
  - GetObject function, 339
- required components, 335
- type libraries, 336
  - references, setting, 336-337

automation servers, 340
- transferring data to Excel, 340-342
- transferring data to Word, 342-345

B

backing up data, case study, 309-311

bands, events, 152

base table, 319

best practices for coding, 21-22
- documenting, 23-24
- indenting, 22

bonuses, calculating, 89-92

Boolean data type, 30
- controls, 181
- converting to, 39

branching, 77
- For..Next, 81-82
  - aborting, 84-86
  - counters, 83
  - nesting, 83-84
  - Step clause, 82
- If...Then...Else, 77-80

Break mode, 17

browsing files, 365-372

buffers, 151

button values, MsgBox() function, 62

Byte data type, 30

ByVal keyword, 72

C

calculating bonuses, case study, 89-92

cancelling events, 152-153

cascading combo boxes, 156

case studies
- adding to combo box, 138-140
- audit trail, 184-185
- backing up data, 309-311
- date functions, 64
- form references, 32-35
- .ini files, 332-333
- product catalog, 196
- updating database installation, 271-272
- version control, 221-223

CBool() function, 39

CDate() function, 40

charts (Excel), creating from Access data, 345-347

Chr() function, 53

CInt() function, 40

class properties, implementing, 112-113

clauses
- FROM, 377
- GROUP BY, 379
- HAVING, 380
- ORDER BY, 379
- SQL, 374
- WHERE, 378-379

Close method, 189-190

closing
- forms, 134-135
- reports, 189-190

cmdPreview button, 141

code
- debugging, 17-18
  - Break mode, 17
  - single step through, 18-19
- saving, 19

Code Builder, 114

coding
- best practices
  - documenting, 23-24
  - identing, 22
  - naming conventions, 21-22
- shortcuts, Intellisense, 20

collections, 105-106
- AllForms, 213
- Application, 211-212

combo boxes
- adding to, case study, 138-140
- cascading, 156
- unlisted values, adding, 159-167

comments, adding to code, 23-24

common action keywords (SQL), 374-375

comparing public and private keywords, 70

complete functions list, displaying, 38

complex data fields, 229

conditional branching, 77
- For...Next statement, 81-82
  - aborting, 84-86
  - counters, 83
  - nesting, 83-84
  - Step clause, 82
- If...Then...Else statement, 77-80

connecting to data source, 273
- using ADO object model, 275-278
- using DAO object model, 273-275

constants, 25
- data types, 28-30
- declaring, 28
- for LockTypeEnum method, 281

for OpenRecordset method, 280-281
intrinsic, 28
lifetime, measuring, 121
module-level, 119
public, 120-121
system-defined, 27

control-specific events, 148

controls, 32, 143
active control, 144
Boolean, 181
combo boxes
cascading, 156
unlisted values, adding, 159-167
event sequence, 144-146
focus, tracking, 179-180
list controls
populating, 155-156
records, filtering, 156-158
multiselect, 169-172
option groups, 167-169
subforms, 182
Tag property, 183
text boxes, 177
properties, 177-179

conversion functions, 38
CBool(), 39
CDate(), 40
CInt(), 40
CStr(), 41
CVar(), 41
example of, 46-48
Nz(), 41
syntax, 39
Val(), 39

creating
custom properties, 215-217
custom ribbons, 209
databases
ADOX object model, 243-246
DAO object model, 241-243
dates from individual parts, 45
dates from string values, 46
fields
ADOX object model, 255-259
DAO object model, 252-255
form-based menus, 199
indexes
ADOX object model, 261-262
DAO object model, 260-261
link tables, 313-314
objects, 102-104, 110-112
objects from automation servers
CreateObject function, 338
early binding, 339-340
GetObject function, 339
relationships
ADOX object model, 265-266, 269-271
DAO object model, 263-264, 267-269
tables
ADOX object model, 249-251
DAO object model, 246-249

crosstab queries, 383-384

CStr() function, 41

Currency data type, 30

current date, returning, 42

cursors, 279-280

custom class modules, 15

custom properties, creating, 215-217

custom ribbons, 208-209

CVar() function, 41

D

DAO (Data Access Objects) object model, 229
AddNew method, inserting data, 291-292
connecting to data source, 273-275
databases, creating, 241-243
Delete method, 307
Execute method, inserting data, 288-290
fields, creating, 252-255
Filter method, 298-299
Find methods, 295-297
indexes, creating, 260-261
object hierarchy, 229-232
queries, creating, 267-269
Recordsets, opening, 280-283
relationships, creating, 263-264
Seek method, 297-298
tables, creating, 246-249

data
backing up, 309-311
deleting with Delete() method, 306
ADO object model, 307-308
DAO object model, 307
inserting
ADO object model, 290-293
DAO object model, 288-292
updating, 304-306

data events, 146-151

data interface object models, 228-229

data sources, 227
connecting to, 273
ADO object model, 275-278
DAO object model, 273-275

data types, 28-30
assigning to functions, 70

data validation, case study, 153-154

Database Splitter, launching, 313

databases, creating
ADOX object model, 243-246
DAO object model, 241-243

How can we make this index more useful? Email us at indexes@quepublishing.com

Date data type, 30
  converting to, 40

date functions, 42
  case study, 64
  date component functions, 46
  Date(), 42
  DateAdd() function, 42-43
  DateDiff() function, 43-44
  DatePart() function, 44-45
  DateSerial() function, 45
  DateValue() function, 46
  example of, 46-48

Date$() function, 42

DateAdd() function, 42-43

DateDiff() function, 43-44

DatePart() function, 44-45

DateSerial() function, 45

DateValue() function, 46

Day() function, 46

DCount() function, 60

DdB() function, 50

DDL (Data Definition Language), 315-316, 374
  keywords, 315

debugging code, 17-18
  Break mode, 17
  single step through, 18-19

Decimal data type, 30

decimal values, removing from numbers, 48

declaring
  arrays, fixed-size, 94
  constants, 28
  variables, 25-26

DELETE statement (SQL), 383

deleting data, Delete method, 306
  ADO object model, 307-308
  DAO object model, 307

dependencies
  tracking, 219-220
  viewing, 217-218

depreciation of assets, calculating, 50

derived table, 319

difference between dates, determining, 43-45

Dim statement, 25

displaying complete functions list, 38

DLookup() function, 60

DMax() function, 60

DMin() function, 60

DML (Data Manipulation Language), 374-375

Do loop, 86-88

DoCmd.Close method, closing forms, 134-135

DoCmd.OpenForm method, opening forms, 131

documenting code, 23-24

domain aggregate functions, 59-60

Double data type, 30

double-declining balance method, 50

dynamic arrays, 98
  Erase statement, 98-99
  ReDim statement, 98

E

early binding, 339-340

elements, 93
  of arrays, 95
    assigning, 95-96
    specifying, 96
  specifying in multidimensional arrays, 96

embedded macros, 8

EOF function, 327

Erase statement, 98-99

error handling
  On Error Goto statement, 74-75
  On Error Resume Next statement, 73
  report-level errors, 194

event procedures, 67, 113

events, 113-114, 144-146, 151-152
  cancelling, 152-153
  control-specific, 148
  data events, 146-151
  data validation, case study, 153-154
  form events, 137-138
  navigation events, 149
  responding to, 141, 143

examples
  of conversion function, 46-48
  of date function, 46-48
  of financial function, 51
  of mathematical function, 49

Excel
  charts, creating from Access data, 345-347
  transferring data to, 340-342

Execute method, data, inserting
  ADO object model, 290-291
  DAO object model, 288-290

exporting
  data to XML files, 350-353
  HTML files, 353

ExportXML method, 350, 353
  arguments, 350-352
  example use of, 352-353

expressions, 38

extracting parts of dates, 44-46

F

fields, 143
  creating
    ADOX object model, 255-259
    DAO object model, 252-255

file handle, 323

file I/O, 323
- opening files, 324-326
- printing to files, 331
- reading from files, 326-329
- writing to files, 329-330

Filter method (DAO object model), 298-299

Filter property (ADO object model), 302-304

filtering
- list control records, 156-158
- report data, 193

financial functions, 50
- DdB() function, 50
- example of, 51
- Pmt() function, 51
- Rate() function, 51

Find method
- ADO object model, 299-301
- DAO object model, 295-297

FindExecutable API, 364

finding data in recordsets, 294
- Filter method
  - ADO object model, 302-304
  - DAO object model, 298-299
- Find method
  - ADO object model, 299-301
  - DAO object model, 295-297
- results, limiting, 294-295
- Seek method
  - ADO object model, 301-302
  - DAO object model, 297-298

FindWindow API, 362-363

fixed-size arrays, declaring, 94

flow of control statements, 77
- Do loop, 87-88
- For...Next, 81-82
  - aborting, 84-86
  - counters, 83
  - nesting, 83-84
  - Step clause, 82
- GoTo, 89
- If...Then...Else, 77-80

focus, 144

For...Next statements, 81-82
- aborting, 84-86
- counters, 83
- nesting, 83-84
- Step clause, 82

foreign key, 262

foreign table, 262

form events, 137-138

form modules, 135

form objects, setting properties, 136-137

form-based menus, 200-203
- creating, 199

Format() function, 55
- date/time named formats, 56
- numeric named formats, 56
- user-defined formats, 57-59

forms
- closing, 134-135
- controls, 32, 143
  - active control, 144
  - event sequence, 144-146
- events, 149
  - data events, 150-151
  - navigation events, 149
- multiple items, selecting, 172-174
- OpenArgs argument, 133
- opening, 131-133
- properties, 135
  - setting, 136-137
- referencing, case study, 32, 34-35
- subforms, 182

FROM clause, 377

functions, 37, 69-70
- API calls, 362
  - FindExecutable API, 364
  - FindWindow API, 362-363
  - Network Login ID, capturing, 363
- complete list of, displaying, 38
- conversion, 38
  - CBool(), 39
  - CDate(), 40
  - CInt(), 40
  - CStr(), 41
  - CVar(), 41
  - example of, 46-48
  - Nz(), 41
  - syntax, 39
  - Val(), 39
- CreateObject, 338
- data type, assigning, 70
- date functions, 42
  - case study, 64
  - date component functions, 46
  - Date(), 42
  - DateAdd(), 42-43
  - DateDiff(), 43-44
  - DatePart(), 44-45
  - DateSerial(), 45
  - DateValue(), 46
  - example of, 46-48
- domain aggregate functions, 59
  - DCount(), 60
  - DLookup(), 60
  - DMax(), 60
  - DMin(), 60
- EOF, 327
- financial functions, 50
  - DdB() function, 50
  - example of, 51
  - Pmt() function, 51
  - Rate() function, 51
- Format(), 55
  - date/time named formats, 56
  - numeric named formats, 56
  - user-defined formats, 57-59

GetObject, 339
Is() functions, 61
IsTable(), 319
LOF, 327
mathematical functions, 48
Abs(), 48
example of, 49
Int(), 48
Rnd(), 49
Open(), 324
arguments, 324-325
string functions
Asc(), 52
Chr(), 53
Lcase(), 53
Left(), 54
Len(), 53
Mid(), 54
Replace(), 54
Right(), 54
Split(), 55
Trim(), 55
Ucase(), 53
user interaction
InputBox(), 63-64
MsgBox(), 61-62

G-H

generating random numbers, 49

GoTo statement, 89

GROUP BY clause, 379

grouping objects within NavPane, 204-206

handling report errors, 194

HAVING clause, 380

hiding toolbars, 15

hierarchical structure
of access objects, 229-232
of ADO objects, 233-234
of ADOX objects, 235-236

Hour() function, 46

HTML files, exporting, 353

Hungarian convention, 21-22

I

I/O, 323
opening files, 324-326
printing to files, 331
reading from files, 326-329
writing to files, 329-330

icon constants, MsgBox() function, 62

identifiers, 31

If...Then...Else statements, 77-80

importing data into XML files, 354
example of, 355-357

ImportXML method, 354-357

in place filtering, 302

indenting code, 22

index values, 94
Option Base statement, 95

indexes, creating
ADOX object model, 261-262
DAO object model, 260-261

initiating procedures, 16-17

InputBox() function, 63-64

INSERT statement (SQL), 380-381

inserting
data
ADO object model, 290-293
DAO object model, 288-292
modules, 15
subroutines in modules, 68-69

Int() function, 48

Integer data type, 30
converting to, 40

Intellisense, 20

interaction functions
InputBox(), 63-64
MsgBox(), 61-62

intrinsic constants, 28

invoking methods, 105

Is() functions, 61

IsTable() function, 319

J-K-L

keywords
public and private, comparing, 70
SQL, 374-375
subqueries, syntax, 320

launching Database Splitter, 313

LCase() function, 53

Left() function, 54

Len() function, 53

lifetime, 117
measuring, 121
of module-level variables, measuring, 122-123
of public variables, measuring, 124

limiting results in searches, 294-295

linked tables, creating, 313-314

list controls
combo boxes, adding unlisted values, 159-167
populating, 155-156
record, filtering, 156-158

loans
payments, calculating, 51
periodic interest, calculating, 51

local variables, 117

LockTypeEnum method, constants, 281

LOF function, 327

Long data type, 30

loops, Do Loop, 86-87
aborting, 88

M

macros, 8

mathematical functions
- Abs() function, 48
- example of, 49
- Int() function, 48
- Rnd() function, 49

measuring lifetime, 121
- of module-level variables, 122-123
- of public variables, 124

menu bar options (VBE), 12-13

menus, 199
- form-based, 199-203

methods
- arguments, specifying, 135
- Close, 189-190
- DoCmd.Close, closing forms, 134-135
- DoCmd.OpenForm, opening forms, 131
- ExportXML, 350, 353
  - arguments, 350-352
  - example use of, 352-353
- ImportXML, 354-357
- invoking, 105
- OpenForm, OpenArgs argument, 133
- OpenReport, 188
  - arguments, 188-189
  - OpenArgs argument, 190-191
- syntax, 105

Microsoft Office Fluent UI, 208-209

Mid() function, 54

Minute() function, 46

module-level variables, 119
- lifetime, measuring, 122-123

modules, 15
- subroutines, inserting, 68-69

Month() function, 46

MsgBox() function, 61-62

multidimensional arrays, 96-98
- elements, specifying, 96

multiselect controls, 169-172

N

naming conventions
- applying, 21-22
- variables, 27

navigation events, 149

NavPane, 204
- methods, 206-208
- objects, grouping, 204-206

nesting, 83-84

Network Login ID, capturing, 363

Now() function, 42

Null data types, converting, 41

Nz() function, 41

O

Object Browser, 109-110

Object data type, 30

object lifetime, 122

object models, 106-107
- ADO, 232
  - object hierarchy, 233-234
- ADOX, 234
  - object hierarchy, 235-236
- DAO, 229
  - object hierarchy, 229-230, 232
- data interface object models, 228-229
- referencing, 108
- selecting, 236-237

objects, 101-102
- collections, 105-106
- controls, 32
- creating, 102-104, 110-112
- grouping within NavPane, 204-206
- properties
  - events, 113-114
  - values, assigning, 104
- referencing, 31-32
  - case study, 32-35
- UI objects, 227

octothorpes (#), 44

On Error Goto statement, 74-75

On Error Resume Next statement, 73

one-dimensional arrays, 96

Open function, 324
- arguments, 324-325

OpenArgs argument, 133

OpenArgs argument (OpenReport method), 190-191

OpenDatabase method, 274-275
- arguments, 273

OpenForm method, OpenArgs argument, 133

opening
- files, I/O method, 324-326
- forms, 131-133
- Recordsets, 278-280
  - ADO object model, 284-288
  - DAO object model, 280-283
- reports, 188-189
- schema recordsets, 317

OpenRecordset method
- arguments, 280
- constants, 280-281

OpenReport method, 188
- arguments, 188-189
  - OpenArgs, 190-191

operators, 31

Option Base statement, 95

Option Explicit statement, 26

- **option groups, 167-169**
- **optional arguments, 71-72**
- **ORDER BY clause, 379**

P

- **passing arguments**
  - by reference, 72
  - by value, 72-73
- **payments for annuities, calculating, 51**
- **periodic interest, calculating, 51**
- **Platform SDK, 360**
- **Pmt() function, 51**
- **populating**
  - list controls, 155-156
  - reports, 191-193
- **predefined formats, applying to data, 55-57**
- **predicates (SQL), 376-377**
- **primary key, 262**
- **primary table, 262**
- **printing to files, I/O method, 331**
- **private keywords, 70**
- **procedure-level variables, 117-119**
- **procedures, 67**
  - arguments, 71
    - optional, 71-72
    - passing by reference, 72
    - passing by value, 72-73
  - functions, 69-70
    - data type, assigning, 70
  - initiating, 16-17
  - public versus private, 70
  - stubs, 16
  - subroutines, 67
    - inserting in modules, 68-69
- **product catalog case study, 196**
- **properties**
  - AccessObject object, 214-215, 217
  - events, 113-114
  - of classes, implementing, 112-113
  - of controls, Tag property, 183
  - of form objects, setting, 136-137
  - of forms, 135
  - of objects, values, assigning, 104
  - of text boxes, 177-179
  - read/write, 105
  - write-only, 105
- **Public keyword, 70**
- **public keywords, 70**
- **public variables, 120-121**
  - lifetime, measuring, 124

Q-R

- **qualifiers, 32**
- **queries**
  - creating
    - ADOX object model, 269-271
    - DAO object model, 267-269
  - crosstab queries, 383-384

- **random numbers, generating, 49**
- **Rate() function, 51**
- **read/write properties, 105**
- **reading from files, I/O method, 326-329**
- **Recordsets**
  - cursors, 279-280
  - data, deleting, 306
    - Delete method (ADO object model), 307-308
    - Delete method (DAO object model), 307
  - data, finding, 294
    - Filter method (DAO object model), 298-299
    - Filter property (ADO object model), 302-304
    - Find method (ADO object model), 299, 301
    - Find methods (DAO object model), 295, 297
    - results, limiting, 294-295
    - Seek method (ADO object model), 301-302
    - Seek method (DAO object model), 297-298
  - data, updating, 304-306
  - opening, 278-280
    - ADO object model, 284-288
    - DAO object model, 280-283
- **ReDim statement, 98**
- **referencing object models, 108**
- **referencing objects, syntax, 31-35**
- **relationships, creating**
  - ADOX object model, 265-266
  - DAO object model, 263-264
- **Replace() function, 54**
- **replacing characters within a string, 54**
- **reports**
  - closing, 189-190
  - error handling, 194
  - event sequence, 151-152
  - filtering data, 193
  - opening, 188-189
  - populating, 191-193
  - subreports, 195
  - troubleshooting, 195
- **responding to events, 141-143**
- **ribbons, 208**
  - customizing, 209
- **Right() function, 54**
- **Rnd() function, 49**

S

- **saving code, 19**
- **schema, 227, 239**

schema recordsets, 317-319

scheme definition files, 350

scope, 117
- module-level, 119
- procedure-level, 117-119
- public, 120-121

Second() function, 46

Seek method
- ADO object model, 301-302
- DAO object model, 297-298

Select Case statement, 80-81

SELECT INTO statement (SQL), 382-383

SELECT statement (SQL), 376
- FROM clause, 377
- GROUP BY clause, 379
- HAVING clause, 380
- ORDER BY clause, 379
- predicates, 376-377
- WHERE clause, 378-379

selecting
- files from File, Open dialog, 365-371
- multiple form items, 172-174
- object models, 236-237

setting breakpoints, 18

shortcuts for coding, Intellisense, 20

Simonyi, Charles, 21

Single data type, 30

single stepping through code, 18-19

specifying
- arguments for methods, 135
- elements in multidimensional arrays, 96

specifying array elements, 96

split configuration, implementing, 313

Split() function, 55

SQL (Structured Query Language), 9
- clauses, 374
- crosstab queries, 383-384
- DDL, 315-316
  - keywords, 315
- keywords, 374
  - common action keywords, 374-375
- statements, 374
  - DELETE, 383
  - INSERT, 380-381
  - SELECT, 376-380
  - SELECT INTO, 382-383
  - UPDATE, 382
- subqueries, 319
  - keyword syntax, 320

standard modules, 15

Standard toolbar (VBE), 13-14

statements
- Dim, 25
- Option Explicit, 26
- SQL, 374
  - DELETE, 383
  - INSERT, 380-381
  - SELECT, 376-380
  - SELECT INTO, 382-383
  - UPDATE, 382

static variables, 124-125

Steele, Doug, 365

String data type, 30
- converting to, 41

string functions
- Asc(), 52
- Chr(), 53
- LCase(), 53
- Left(), 54
- Len(), 53
- Mid(), 54
- Replace(), 54
- Right(), 54
- Split(), 55
- Trim(), 55
- UCase(), 53

structured files, 350

stub, 16

subforms, 182

subqueries, 319
- keywords, syntax, 320

subreports, 195

subroutines, 67
- inserting in modules, 68-69

syntax, 25
- for Abs() function, 48
- for arguments, 71
- for Asc() function, 52
- for Chr() function, 53
- for conversion functions, 39
- for DateAdd() function, 42
- for DateDiff() function, 43
- for DatePart() function, 44
- for Dbd() function, 50
- for Format() function, 56
- for InputBox() function, 63
- for Int() function, 48
- for Is() functions, 61
- for LCase() function, 53
- for Left() function, 54
- for Len() function, 53
- for methods, 105
- for Mid() function, 54
- for MsgBox() function, 61
- for Pmt() function, 51
- for Rate() function, 51
- for referencing objects, 31-32
  - case study, 32-35
- for Replace() function, 54
- for Right() function, 54
- for Rnd() function, 49
- for Split() function, 55
- for static variables, 124
- for Trim() function, 55
- for UCase() function, 53

system-defined constants, 27

T

tables, creating
- ADOX object model, 249-251
- DAO object model, 246-249

Tag property, 183

text boxes, 177
  properties, 177-179

toolbars, hiding, 15

tracking
  current application users, case study, 126-127
  dependencies, 219-220

transferring data
  to Excel, 340-342
  to Word, 342-345

Trim() function, 55

troubleshooting reports, 195

two-dimensional arrays, 93

type libraries, 336
  references, setting, 336-337

U

UCase() function, 53

UDFs (user-defined functions), 69-70

UI objects, 227

UPDATE statement (SQL), 382

updating
  data, 304-306
  database installation, case study, 271-272

user-defined formats, applying to data, 57-59

Userforms, 14

V

Val() function, 39

values, formatting, 55
  date/time named formats, 56
  numeric named formats, 56
  user-defined formats, 57-59

variables, 25
  data types, 28-30
  declaring, 25-26
  lifetime, 117
    measuring, 121
  local, 117
  module-level, 119
    lifetime, measuring, 122-123
  naming, 27
  procedure-level, 117-119
  public, 120-121
    lifetime, measuring, 124
  scope, 117
  static, 124-125

Variant data type, 31
  converting to, 41

VBA, 10

VBE (Visual Basic Editor), 11
  menu bar options, 12-13
  Standard toolbar, 13-14

version control, case study, 221, 223

viewing dependencies, 217-218

W

WHERE clause, 378-379
  syntax, 374

window handles, 362

Windows API, 359
  API calls, 359-362
    FindExecutable API, 364
    FindWindow API, 362-363
    Network Login ID, capturing, 363
  files, browsing, 365-372
  when to use, 364

Word, transferring data to, 342-345

write-only properties, 105

writing to files, I/O method, 329-330

X-Y-Z

XML, 349
  exporting data to XML files, 350-353
    example of, 352-353
  importing, 354-357
  structured files, 350
  supported file types, 349-350

Y

Year() function, 46