

INDEX

Symbols & Numbers

- & (ampersand), 166, 167
- && (and operator), 85, 93
- * (asterisk), 251–252, 260
- *?, 260
- @ (at sign), 23, 100
- @@ (class variables), 23, 100
- \ (line continuation character), 297
- \B, 260
- \b, 260
- \D, 260
- \d, 260
- \S, 260
- \s, 260
- \W, 260
- \w, 260
- ^ (caret), 251, 260. *See also* regular expressions
- : (colon)
 - in case statements, 90
 - instead of then, 84
 - in symbols), 181
- :: (scope resolution operator), 30, 185, 193, 208–209, 330
- { } (curly brackets)
 - as delimiters, 161
 - in regular expressions, 260
- \$ (dollar sign), 251–252, 260. *See also* regular expressions; variables
- \$`, 255. *See also* regular expressions
- \$., 202
- #!, 141
- \$/ (record separator), 42
- \$', 255. *See also* regular expressions
- \$~, 256. *See also* regular expressions
- = (assignment operator), 6
- =begin, 4
- =begin rdoc, 345–346
- =end, 4, 345–346
- == (equal-to operator), 56, 136
- ===, 88, 92
- => (assoc operator), 141
- ~, 250. *See also* regular expressions
- ! (exclamation mark)
 - at end of methods, 131
 - as not operator, 85, 86, 93
- != (not-equals operator), 86
- // (delimiters), 250
- > (greater-than sign), 56
- # (hash mark), for comments, 4
- #{} , in strings, 2
- < (less-than sign), 56
 - indicating descent, 17–18
- <=> (comparison operator), 54, 55, 77, 207, 223
- <<
 - append operator, 37, 63, 129
 - concatenation method, 130–131
 - with heredocs, 45
 - and singletons, 109, 117
- <<- , with heredocs, 45
- <%%>. *See* Rails, tags
- <%= . *See* Rails, tags
- () , parentheses,
 - avoiding ambiguity, 85, 137, 162
 - optional, 2, 22
 - in regular expressions, 260
 - and super, 18
- %/ , 35
- %d, 43
- %f, 43
- %o, 43

`%p`, 43
`%Q`, 35
`%q`, 35
`%r`, 250. *See also* regular expressions
`%s`, 43
`%W`, 46, 49
`%w`, 49
`%x`, 36, 43
`.` (period), 260
`+` (plus sign), 252, 260
`+=`, 23
`??`, 260
`?` (question mark), 256, 260
`?..:`, as shorthand for `if..then..else`,
87, 269
`[]` (square brackets), 260
`|` (vertical pipe), 260
`||` (or operator), 85, 93
3rd Rail, 357

A

accessor methods, 19
Ada, xviii
Adobe Flex, 322
alias, 199–200
`all_symbols`, 190
ambiguity, xix. *See also*
 parentheses (`()`)
 avoiding, 123, 199
 hashes and blocks, 162
 line breaks, 285, 296
ampersand (`&`), 166, 167
ancestors. *See* class hierarchy
and operator (`&&`), 85, 93
Apache, 302, 358
Aptana RadRails, 358
arguments
 by reference, 129
 default, 124
 passing, 134–135
 by value, 134
arrays, 47–48
 `&`, 57
 `-`, 57
 `<<`, 57
 `+`, 57
 appending, 63–64

`clear`, 57
 `clone`, 53
 `collect`, 159, 190
 `compact`, 57, 58
 `compact!`, 57
 concatenating, 63–64
 copying, 53
 creating, 48–50
 `delete`, 57
 `delete_at`, 57
 each, 159
 expressions in, 49
 of filenames, 49
 `flatten`, 57, 58
 `flatten!`, 57
 indexing, 52–53
 iterators, 51–52, 159
 length, 47
 methods, 57–58
 multidimensional, 50
 nested, 50
 reverse, 57
 `reverse!`, 57
 sort, 57
 `sort!`, 57
 sorting, 54–55
 of strings, 49
 testing for equality, 53–54
 treating hashes as, 63
ASCII character codes, testing, 215
`assert`, 293, 295
`assert_block`, 295
`assert_equal`, 292, 295
`assert_in_delta`, 295
`assert_instance_of`, 295
`assert_kind_of`, 294
`assert_match`, 295
`assert_nil`, 295
`assert_no_match`, 295
`assert_not_equal`, 295
`assert_nothing_raised`, 296
`assert_nothing_thrown`, 296
`assert_not_nil`, 295
`assert_not_same`, 295
`assert_operator`, 296
`assert_raise`, 296
`assert_raises`, 296
`assert_respond_to`, 296

- assert_same, 296
- assert_send, 296
- assert_thrown, 296
- assignment operator (=), 6
- assignments, 135–136
 - multiple, 133
 - parallel, 133
- associative arrays. *See* hashes
- assoc operator (=>), 141
- asterisk (*), 251–252, 260
- at sign (@), 23, 100
- attr_accessor, 21, 327
- attributes, 20–23
- attr_reader, 20, 182
- attr_writer, 20, 182

B

- \B, 260
- \b, 260
- backquotes. *See* strings
- backtrace, 150. *See also* exceptions
- BasicObject, 29, 106, 142
- =begin, 4
- begin..end, 73, 152
- =begin rdoc, 345–346
- bindings, 334–336
- block_given?, 171
- block parameters, 70, 157, 168, 170, 179
- blocks, 70, 155–156. *See also* fibers
 - {}, 70, 161
 - call, 162
 - creating methods from, 332, 337
 - creating objects from, 162
 - do..end, 70, 158
 - and for loops, 180
 - and instance variables, 176–177
 - and iterating arrays, 159–161
 - as iterators, 172–174
 - and line breaks, 156–157
 - and local variables, 177–180
 - nested, 165
 - parameters, 70, 157, 168, 170, 179
 - passing as named arguments, 165–170
 - precedence, 170–171

- returning from methods, 175–176
- Ruby 1.9, 155
- scope of, 168, 177
- sorting with, 54
- blogs, 316–321
- books, on Ruby, 353–354
- Boolean operators, 85–86, 93–94
 - negation, 86
 - precedence, 93
 - side effects, 94
- break, 75, 90, 158–159
- breaking code over a line, 297. *See also* line breaks
- breakpoints, 289. *See also* debugging
- build_message, 296
- bytes, reading, 247

C

- call, 187
- camel case, xviii
- capitalization. *See* strings
- caret (^), 251, 260. *See also* regular expressions
- case sensitivity, 2
- case statements, 89
 - return values, 92
 - selectors, 91
- catch, 94–96, 152–153
- character codes. *See* strings
- chomp, 42
- chop, 42
- classes, 6
 - descending from Object, 106
 - vs. modules, 192, 205
 - as objects, 104
 - open, 30
 - partial, 30–31
 - singleton, 108
- class_eval, 329–331
- class hierarchy, 15–16
- class methods, 97, 98, 122. *See also* methods
- class_variable_get, 331–333
- class variables, 23, 99
- class_variables, 332
- class_variable_set, 331–333

- clone, 60
- closures, 163–164. *See also* blocks
- collect, 76, 273. *See also* arrays;
 - Enumerable
- colon (:)
 - in case statements, 90
 - instead of then, 84
 - in symbols, 181
- comments, 4
 - in code examples, xvii
 - multiline, 4
 - single line, 4
- Comparable, 55, 206, 207
- comparison operator (<=>), 54, 55,
 - 77, 207, 223
- conditional statements, 83. *See also* case statements;
 - if modifiers
- constants, 90
 - accessing with ::, 30, 207, 208
 - inside classes, 30
- const_get, 333
- constructors
 - initialize, 10–11, 104–105
 - new, 10–11, 104–105
 - overriding, 104–105
- const_set, 333
- Controller, 305–307, 323
- copying files, 217–218. *See also* files
- cp, 218
- curly brackets, {}
 - as delimiters, 161
 - in regular expressions, 260
- custom comparisons, 77

D

- \D, 260
- \d, 260
- %d, 43
- databases, 358–359
- data hiding, 128
- debuggers, visual, 226, 297, 357–358
- debugging, 286
 - breakpoints, 289
 - with Ruby Debugger, 289
 - watchpoints, 290
- def, 5
- default, 60
- default=, 60
- defined?, 179
- define_method, 188, 330, 337
- delete, 62
- descendants, 15–16
- destructors, 11
- dictionaries. *See* hashes
- Digging Deeper
 - arguments, passing, 134–135
 - arrays, 63
 - assertions, 295–296
 - assignments, 135–136
 - begin..end, 152
 - blocks and closures, 175–180
 - Boolean operators, 93–94
 - catch..throw, 94, 152–153
 - debuggers, 297
 - each, 81
 - Enumerable, 76–81
 - equality, testing for, 136
 - exceptions, 152–153
 - extending objects, 210
 - freezing objects, 211, 342–343
 - hashes, 63
 - heredocs, 45–46
 - line breaks, 296–297
 - Marshal version numbers,
 - 246–248
 - method accessibility, 114–116
 - modules, 205–210
 - MVC, 322–323
 - Rails folders, 323
 - recursions, 224–226
 - regular expressions, 260–261
 - Ruby frameworks, 324
 - scope resolution, 208–209
 - string literals, 46
 - superclasses, 25–30
 - symbols, 190, 261
 - threads, 278–281
 - unit testing, 295–296
 - YAML, 237–238
 - yield, 81
- directories, navigating in, 219. *See also* files
- Dir.foreach, 220
- Dir.getwd, 144

- do, 69, 156
- documenting Ruby. *See* RDoc
- do..end, 156, 158. *See also* blocks
- dollar sign (\$), 251–252, 260. *See also* regular expressions; variables
- downto, 70–71
- dynamic programming, 188, 325
 - adding methods, 337
 - adding methods to classes, 332
 - creating classes at runtime, 333
 - creating objects at runtime, 333
 - and embedded evaluation, 326
 - eval, 327
 - freezing objects, 342
 - monkey patching, 330
 - removing methods, 337
 - self-modifying programs, 326, 328
 - writing programs at runtime, 340

E

- each, 68, 79, 159, 234
 - vs. for, 68
 - and yield, 81
- ebooks, on Ruby, 355
- editors, Ruby, 357–358
- Eiffel, xviii, 6
- else, 145. *See also* exceptions
- Embedded Ruby (ERb), 313, 314
- encapsulation, 8, 128
 - breaking, 131, 333
- end, 5
- =end, 4, 345–346
- ensure, 145. *See also* exceptions
- Enumerable, 76–81, 206
 - collect, 76
 - include?, 76
 - including in a class, 79
 - max, 76
 - min, 76, 78
 - overriding methods, 80
- equal?, 135, 136, 183
- equality, testing for, 136
- equal-to operator (==), 56, 136
- ERb (Embedded Ruby), 313, 314
- Errno, 146–147. *See also* exceptions

- eval, 186, 327. *See also* dynamic programming
- example programs, xix
- exception hierarchy, 142
- exceptions, 140
 - and \$!, 141
 - associating with variable name, 141
 - backtrace, 150
 - begin..end, omitting, 152
 - creating custom messages for, 150–151
 - custom, raising, 151
 - displaying as strings, 148
 - else, 145–146
 - ensure, 144
 - Errno constants, 147
 - multiple, 142
 - NoMethodError, 142
 - raise, 149
 - rescue, 140
 - retry, 148
 - RuntimeError, 150
 - subclassing, 151
 - TypeError, 142, 342
 - ZeroDivisionError, 141
- exclamation mark (!)
 - at end of methods, 131
 - as not operator, 85, 86, 93
- extend, 210

F

- %f, 43
- fibers
 - alive?, 276
 - and blocks, 275
 - dead, 276
 - FiberError, 276
 - parameters, 276
 - resume, 275
 - and threads, 275
 - yield, 275
- File methods, 98
- files
 - backing up, 219
 - closing, 214–216
 - copying, 217

- files (*continued*)
 - and directories, 217
 - eof, 215
 - exist?, 217
 - foreach, 254, 258
 - handles, 215
 - modes, 214
 - opening, 214
 - processing using regular
 - expressions, 258–259
 - reading/writing text, 215, 259
 - saving data to YAML, 231
 - size of, 220
 - calculating, 219
 - sorting by, 222
 - as a subclass of IO, 214
- FileTest, 206
- FileUtils, 218
- Fixnums, 136. *See also* integers
- flunk, 296
- foreach, 213
- for...in loops, 51
- for loops, 67–70
 - and arrays, 71
 - and blocks, 180
 - and local variable scoping, 180
 - and multiple iterator
 - arguments, 71
 - and ranges, 69
- format strings, 43
- freeze, 211, 342–343
- frozen?, 211, 342
- functions, 157–158. *See also* methods
 - and module_function, 209

G

- garbage collection, 11
- GC, 206
- Gemfile, 319
- Gems, Ruby, 300
- generate, 306
- getc, 216, 248
- global variables, 5, 184. *See also*
 - variables
- goto, 94

H

- Hansson, David Heinemeier, 316
- hashes, 58
 - copying, 60
 - creating, 58–59
 - and curly brackets, 161–162
 - has_key?, 62
 - has_value?, 62
 - indexing into, 60
 - items in
 - deleting, 62
 - finding position of, 62
 - merging, 62
 - methods, 62
 - order of elements, 60, 229
 - sort, 223
 - sorting, 61
 - treating as arrays, 63
- hask mark (#), for comments, 4
- heredocs, 45–46
- HTML, tags, 309, 311. *See also* Rails

I

- id2name, 186
- IDEs (integrated development environments), 283, 357–358
- if...else, 84
- if...elsif, 86
- if modifiers, 88–89
- if...then, 4–5
- if...then...else, 84
- include, 194. *See also* modules
- include?, 76, 88. *See also* Enumerable
- index, 62
- information hiding, 128
- inheritance, 16
- initialize, 10–11
- input and output. *See* IO (input and output)
- inspect, 11–13
- instance_eval, 329
- instance methods, 121–122
- instance_variable_get, 103, 333
- instance variables, 7, 23, 100
- instance_variables, 197
- instance_variable_set, 332

- integers
 - object_id, 127
 - testing for equality, 136
- integrated development environments (IDEs), 283, 357–358
- Interactive Ruby (IRB), 283–286
- invert, 62
- IO (input and output)
 - class, 213
 - eof, 215
 - exceptions in, 144
 - foreach, 213
 - getc, 215, 248
 - gets, 2
 - print, 2
 - printf, 43, 221
 - putc, 215
 - puts, 1
 - readlines, 214
- IRB (Interactive Ruby), 283–286
- IronRuby, 359

J

- JRuby, 359

K

- Kernel, 193, 206, 330
- key, 62
- keys, 59, 62

L

- lambda, 162–163
- lambdas, 161
- less-than sign (<), 56
 - indicating descent, 17–18
- LightTPD, 302, 358
- line breaks, 285, 296–297
- line continuation character (\), 297
- Lisp, xviii, 163
- load, 203
- LoadError, 202
- LocalJumpError, 178
- local variables. *See* variables
- loop, 75, 158

- loops, 73
 - begin..end, 73
 - each, 68
 - executing at least once, 73
 - for. *See* for loops
 - until, 29, 74
 - while, 72

M

- Maglev, 359
- main object, 9
- map, 273
- Marshal library, 239
 - compared to YAML, 240
 - dump, 239
 - file formats, 240
 - load, 239, 241
 - marshal_dump, 240, 243
 - marshal_load, 241, 243
 - omitting variables, 240
 - and singletons, 243
 - TypeError, 240
 - version numbers, 246–248
- MatchData, 254–255. *See also* regular expressions
- Math, 193, 206, 207
- matrices, 64–65
- Matsumoto, Yukihiro (Matz), xviii
- max, 76–80. *See also* Enumerable
- messages, sending, 9, 64
- metaprogramming, 326. *See also* dynamic programming
- method, 185, 187
- method_defined?, 189
- method_missing, 339
- methods, 9, 97
 - adding at runtime, 188, 337
 - class, 97, 122
 - vs. instance methods, 102
 - vs. singleton methods, 105, 116–118
 - declaring, 5
 - default arguments, 124
 - instance, 121
 - vs. class methods, 102
 - and message sending, 64

- methods (*continued*)
 - missing, 338
 - module, 192
 - nameless, 157–158
 - naming, and capitalization, 119–120
 - nested, 118–119
 - overriding, 110
 - private, 111–113
 - in descendant classes, 115
 - invading private methods, 116
 - protected, 111–113, 114–116
 - public, 111–113, 115–116
 - removing at runtime, 337
 - returning multiple values, 124
 - returning values, 8, 123
 - singleton, 105, 122
 - vs. class methods, 105, 116–118
 - static, 97
 - migration, 318. *See also* Rails
 - min, 76–80. *See also* Enumerable
 - mixins, 194–197. *See also* modules
 - potential problems with, 200–201
 - vs. multiple inheritance, 201
 - mkdir, 218
 - Model, 322
 - Model-View-Controller (MVC), 301–302, 322–323
 - Controller, 305–307, 323
 - Model, 322
 - View, 310–313, 322–323
 - module_eval, 329
 - module_function, 209
 - modules, 191, 192
 - vs. classes, 192, 205
 - and class variables, 197
 - defined, 205
 - and extend, 210
 - and functions, 209
 - included. *See* mixins
 - including from files, 201
 - and instance methods, 192
 - and local variables, 195
 - as namespaces, 193
 - predefined, 205–207
 - resolving name conflicts with, 198
 - Mongrel, 302, 358
 - monkey patching. *See* dynamic programming
 - Mutex class, 272–275
 - mv, 218
 - MVC. *See* Model-View-Controller (MVC)
 - MySQL, 300, 317, 349, 358
 - configuring, 351
 - .dll file (Windows), 352
 - errors, 319, 351–352
 - no such file to load, 304, 351
 - rake aborted, 319
 - gem, installing, 351
 - installing, 350
 - username and password, 351
- ## N
- namespaces, 193. *See also* modules
 - nested methods, 118–119
 - NetBeans, 358
 - new, 10–11
 - Nginx, 358
 - nil
 - and array indexes, invalid, 48
 - displaying, 48, 251
 - nil?, 48
 - NilClass, 48
 - not-equals operator (!=), 86
 - not operator (!), 85, 86, 93
 - numbers, 3
- ## O
- %, 43
 - Object, 29, 106
 - methods, 185, 188
 - object_id, 39, 125, 126
 - objects, 6–7
 - creating at runtime, 333
 - extending, 210
 - freezing, 211, 342–343
 - identical, identifying, 136
 - inspecting, 11–13
 - ObjectSpace, 206
 - or operator (||), 85, 93
 - overriding methods, 110

P

p, 12
%p, 43
parallel assignments, 133
parameters. *See* arguments
parentheses, ()
 and super, 18
 avoiding ambiguity, 85, 137, 162
 optional, 2, 22
 in regular expressions, 260
partial classes, 30–31
Pascal, 5, 68
period (.), 260
PI, 207
plus sign (+), 252, 260
polymorphism, 9
PostgreSQL, 359
Precision, 206
private methods, 111–113
 in descendant classes, 115–116
 invading, 116
Proc, 162–163
 new, 162
 in Ruby 1.8, 163
 in Ruby 1.9, 163
Process, 206
procs, 161. *See* blocks
properties, 21. *See also* attributes
protected methods, 111–113,
 114–116
public methods, 111–113, 115–116

Q

%Q, 35
%q, 35
question mark (?), 256, 260

R

%r, 250. *See also* regular expressions
Rails
 and ApplicationController, 308
 blog application in, 316
 and class hierarchy, 312–313
 “configuration by convention”
 approach, 306, 312

 creating a View in, 310
 database.yml, 317
 and embedded Ruby, 312,
 314–315
 ERb files, 313–314, 322
 folders, 323
 generate, 306
 generate controller, 306, 308,
 309–310
 generating view methods in, 309
 and HTML tags
 in Controller, 311
 in strings, 312–313
 installing, 300–301
 and instance variables, 311–312
 migration, 318
 and MySQL, configuring, 317
 vs. other frameworks, 324
 partials, 318
 passing data from Controller to
 View, 311
 pluralization, 322
 and raw, 313
 and render, 306, 309, 311
 .rhtml, 314, 322
 routes, in Rails 3, 307
 running server, 320
 scaffolding, 317–318
 tags, 312, 313
 templates, 323
 updating, 301
Rails 2, commands, 303, 316
 generate controller, 305, 310
 generate scaffold, 317
 server, 304, 307, 320
Rails 3, commands, 303, 316
 script/generate controller,
 305, 310
 script/generate scaffold, 317
 script/server, 304, 307, 320
raise, 149–151. *See also* exceptions
rake, 318
Ramaze, 324
ranges, 43–44
RDoc, 345–347
 adding title with, 347
 excluding comments from, 347
 formatting, 347

- readlines, 214
- receiver objects, 64
 - modifying, 130–131
- record separator (\$/), 42
- recursions, 219–222, 224–226
- regular expressions (Regexp), 249
 - captures, 253, 255
 - custom delimiters, 250
 - elements, 260–261
 - *, 251–252, 260
 - ^, 251, 260
 - \$, 251–252, 260
 - \$`, 255
 - \$~, 256
 - \$', 255
 - // (delimiters), 250
 - %r, 250
 - +, 252
 - ?, 256
 - greedy, 256
 - groups, 253–254
 - match, 254–255
 - MatchData, 254–255
 - matching, 250–253
 - post_match, 255–256
 - pre_match, 255–256
 - processing paths, 256
 - quantifiers, 251
 - quick reference, 260
 - samples of, 260
 - and strings, 256
 - symbols, 261
 - to_a, 255
- remove_method, 189, 337
- require, 201
 - vs. load, 203
 - problems, 202
- require_relative, 202
- rescue, 140–144. *See also* exceptions
- respond_to?, 108
- retry, 148–149
- return, 8, 123
- RHTML, 314, 322. *See also* Rails
- rm, 218
- Rubinius, 359
- Ruby
 - books on, 353–354
 - defined, xviii
 - downloading, xx, 359
 - ebooks on, 355
 - editors, 357–358
 - frameworks, 324. *See also* Rails
 - IDEs, 357–358
 - implementations, xviii, 359
 - library documentation, xxi
 - one-click Installer, 288
 - running Ruby programs, xxi
 - software, 357–358
 - style conventions, xviii–xix
 - websites on, 355
- Ruby 1.8, xviii, xxi
- Ruby 1.9
 - BasicObject, 29, 106
 - binding, 336
 - block parameters, 168, 179
 - class variable scope, 335
 - deadlocks, 272
 - displaying arrays, 52
 - fibers, 275
 - getc, 248
 - handling Ruby 1.8
 - compatibility, 216
 - hash order, 60, 76, 78
 - indexing a hash, 62
 - instance_variables, 197
 - local variables, 179
 - Marshal version numbers,
 - 246–248
 - nil, 48, 251
 - not, 94
 - print, 48, 52
 - puts, 48, 52
 - regular expressions, symbols, 261
 - require, 202
 - require_relative, 202
 - singleton_methods, 110
 - sorting symbols, 190
 - string indexing, 39–41
 - then syntax, unsupported, 84, 90
 - Thread.pass, 279
 - threads, 265
 - priorities, 270, 272
 - statuses, 266–267
 - to_s, with exceptions, 148
- Ruby 2.0, xviii
- Ruby Gems, 300

rubygems.rb, 288
Ruby In Steel, 226, 297, 358. *See also*
 IDEs (integrated develop-
 ment environments)
RubyMine, 358
Ruby on Rails. *See* Rails
RUBYOPT, 288
RUBY_VERSION, 216

S

`\s`, 260
`\s`, 260
`%s`, 43
saving data. *See* Marshal
 library; YAML
scaffolding, 317. *See also* Rails
scope resolution operator (`::`), 30,
 185, 193, 208–209, 330
scopes, 5–6
send, 116, 332, 336
serialization, 240. *See also* Marshal
 library; YAML
server ports, 304
sets, 65
 adding new values to, 65
 merging, 65
 sorting, 66
 testing for equality, 66
setup. *See* unit testing
Signal, 206
Sinatra, 324
singleton classes, 108–110
singleton class methods, 116–118
singleton methods, 105–108, 122.
 See also methods
`singleton_methods`, 108, 110
singletons
 and Marshal library, 243–245
 saving, and reloading, 242
 and YAML, 242–243
sleep, 265
Smalltalk, xviii, 6, 9, 128
sort, 61, 77, 223
source code for this book, xxi
SQLite, 301, 359
 no such file to load, 304
 and Rails, 300

SQLite3, installaing, 301
SQL Server Express, 359
`sqrt`, 207
square brackets, `[]`, 260
static methods. *See* class methods
static variables. *See* class variables
string literals, 46
strings. *See also* regular expressions
 `=~`, 256
 assignments, 38–39
 backquotes, 36, 49
 and capitalize method, 39,
 41, 160
 and capitalize! method, 160–161
 character codes, 37
 and `chomp` method, 42
 and `chop` method, 42
 `chr`, 161
 concatenation, 37
 delimiters, user-defined, 35
 double-quoted, 2, 33–35
 downcase, 41
 `each_byte`, 161
 and embedded evaluation,
 2–3, 33
 format, 43
 formatting, 221
 `gsub`, 258
 indexing into, 39–41
 and `insert` method, 41
 length, 41
 matching
 methods of String class, 256
 regular expressions, 250–253
 and `ord` method, 216, 247
 record separator (`$/`), 42
 and `reverse` method, 41
 and `reverse!` method, 130
 `scan`, 256–257
 single-quoted, 34
 `slice`, 257
 `split`, 41, 160, 257
 `squeeze`, 41
 `sub`, 258
 `swapcase`, 41
 and `to_f` method, 3, 216
 and `to_s` method, 13
 `upcase`, 41

- super, 18, 23, 27, 104
- superclasses, 17–18, 25–30
- symbols, 20, 181
 - defined, 181, 190
 - and dynamic programming, 188
 - evaluating, 186
 - and regular expressions, 261
 - scope of, 184
 - sorting, 190
 - vs. strings, 182
 - uniqueness of, 182
 - usefulness of, 188–189
 - and variables, 186

T

- teardown, 294. *See also* unit testing
- TestCase, 292. *See also* unit testing
- TextMate, 358
- then, 4–5
- threads, 263
 - creating, 264
 - deadlocks, 272
 - ensuring execution of, 268–269
 - and fibers, 275
 - green, 265
 - and join method, 268
 - main, 266, 271
 - mutexes, 272–275
 - native, 265
 - and pass method, 269, 278–281
 - and preemptive multitasking, 265
 - priorities, 269
 - problems, 270
 - setting, 270
 - running, 264–265
 - scheduling, 281
 - and sleep method, 268
 - statuses, 266–268
 - and stop method, 272, 281
 - synchronizing, 273
 - time-slicing, 265
- throw, 94–96, 152–153
- times, 156, 178
- to_a, 44, 255
- to_f, 3, 216
- to_s, 13, 22, 148
- to_yaml, 228

U

- ubygems.rb*, 288
- undef_method, 338
- unit testing, 292
 - assertions available, 295–296
 - setup, 294
 - teardown, 294
 - TestCase, 292
- unless modifiers, 88–89
- unless tests, 88
- until loops, 74–75. *See also* loops
- upto, 70–71

V

- variables
 - class, 23
 - class instance, 102
 - class methods, 99
 - global, 5, 184
 - instance, 7, 100
 - local, 5
 - in modules, 195
 - static, 99
- vectors, 64–65
- vertical pipe (|), 260
- View, 310–313, 322–323

W

- \W, 260
- \w, 260
- %W, 46, 49
- %w, 49
- Waves, 324
- WEBrick, 302
 - running, 304
 - website, 358
- web servers, 302, 358
- websites, on Ruby, 355
- when, 89–91, 92
- while loops, 72–74
- while modifiers, 72–73, 89
- word counter, 258

X

- %x, 36, 43

Y

y, 229

YAML, 227

- adventure game, 236

- arrays, 230, 237–238

- documents, 228, 233

- and dump method, 231

- elements, 228

- formats, 237

- and hashes, 237

- and indentation, 231, 238

- and load_documents method, 234

- loading data, 231, 234

- and nested sequences, 231

- omitting variables, 232–233

- reconstructing complex

 - data, 236

- saving data, 231, 234

- and singletons, 242–243

- to_yaml, 228

- to_yaml_properties, 232

- y, 229

YARV (Yet Another Ruby Virtual-machine), 265

yield, 81, 164. *See also* blocks