INDEX

Symbols

= (assignment) operator, 18, 34
\ (backslash), 124, 151, 162, 174–175
    line continuation character, 93
^ (caret symbol), 162
    matching beginning of string, 159–160
    negative character classes, 159
: (colon), 38, 45, 54, 82, 127
{} (curly brackets), 105, 162
    greedy vs. nongreedy matching, 156–157
    matching specific repetitions with, 156
$ (dollar sign), 159–160, 162
. (dot character), 160–162
    using in paths, 175–176
    wildcard matches, 160–162
" (double quotes), 124
** (exponent) operator, 15
== (equal to) operator, 33, 34
/ (forward slash), 174–175
    division operator, 15, 88
> (greater than) operator, 33
>= (greater than or equal to) operator, 33
(hash character), 126
// (integer division/floored quotient) operator, 15
< (less than) operator, 33
<= (less than or equal to) operator, 33
% (modulus/remainder) operator, 15, 88
* (multiplication) operator, 15, 83, 88
!= (not equal to) operator, 33
() (parentheses), 96–97, 152–153
| (pipe character), 153–154, 164–165
+ (plus sign), 155–156, 162
    addition operator, 15, 17, 83, 88
? (question mark), 154–155, 162
' (single quote), 124
[] (square brackets), 80, 162
* (star), 162
    using with wildcard character, 161
    zero or more matches with, 155
- (subtraction) operator, 15, 88
''' (triple quotes), 125, 164
_ (underscore), 20

A

%A directive, 344
%a directive, 344
absolute paths, 175–179
abspath() function, 177
addition (+) operator, 15, 17, 83, 88
additive color model, 389
add_heading() method, 314
addPage() method, 299
add_paragraph() method, 313–314
add_picture() method, 315
add_run() method, 313–314
algebraic chess notation, 112–113
all_caps attribute, 311
ALL search key, 369
alpha, defined, 388
and operator, 35
ANSWERED search key, 370
API (application programming interface), 327–328
append() method, 89–90
application-specific passwords, 365
args keyword, 349
arguments, function, 23, 63
    keyword arguments, 65–66
    passing to processes, 354
    passing to threads, 348–349
assertions, 219–221
assignment (=) operator, 18, 34
AT&T mail, 363, 367
attributes, HTML, 241, 248
augmented assignment operators, 88–89

B

\b backspace escape character, 419
%B directive, 344
%b directive, 344
back() method, 261
backslash (\), 124, 151, 162, 174–175
BarChart() function, 290
basename() function, 178
BCC search key, 370
Beautiful Soup, 245. *See also* bs4 module
BeautifulSoup objects, 245–246
BEFORE search key, 369
binary files, 180–181, 184–185
binary operators, 35–37
bitwise or operator, 164–165
blank strings, 17
blocking execution, 337
blocks of code, 37–38
BODY search key, 369
bold attribute, 311
Boolean data type
    binary operators, 35–36
    flow control and, 32–33
    in operator, 87
    not in operator, 87
    "truthy" and "falsey" values, 53
    using binary and comparison operators together, 36–37
box tuples, 390
breakpoints, debugging using, 229–231
break statements
    overview, 49–50
    using in for loop, 55
browser, opening using webbrowser module, 234–236
bs4 module
    creating object from HTML, 245–246
    finding element with select() method, 246–247
    getting attribute, 248
    overview, 245
built-in functions, 57
bulleted list, creating in Wiki markup, 139–141
    copying and pasting clipboard, 139–140
    joining modified lines, 141
    overview, 139
    separating lines of text, 140

C

calling functions, 23
call stack, defined, 217
camelcase, 21
caret symbol (^), 162
    matching beginning of string, 159–160
    negative character classes, 159
Cascading Style Sheets (CSS)
    matching with selenium module, 258
    selectors, 246–247
case sensitivity, 21, 163
CC search key, 370
Cell objects, 268–269
cells, in Excel spreadsheets, 266
    accessing Cell object by its name, 268–269
    merging and unmerging, 286–287
    writing values to, 278–279
center() method, 133–134, 426
chaining method calls, 398
character classes, 158–159, 162
character styles, 310
charts, Excel, 288–290
chdir() function, 175
Chrome, developer tools in, 242–243
clear() method, 258
click() function, 420, 430, 431
clicking mouse, 420
click() method, 259
clipboard, using string from, 236
CMYK color model, 389
colon (:), 38, 45, 54, 82, 127
color values
    CMYK vs. RGB color models, 389
    RGBA values, 388–389
column_index_from_string() function, 270
columns, in Excel spreadsheets
    setting height and width of, 285–286
    slicing Worksheet objects to get Cell objects in, 270–272
Comcast mail, 363, 367
comma-delimited items, 80
command line arguments, 235
commentAfterDelay() function, 429
comments
    multiline, 126
    overview, 23

comparison operators
overview, 33–35
using binary operators with, 36–37
compile() function, 151, 152, 164–165
compressed files
backing up folder into, 209–212
creating ZIP files, 205–206
extracting ZIP files, 205
overview, 203–204
reading ZIP files, 204
computer screen
coordinates of, 415
resolution of, 416
concatenation
of lists, 83
string, 17–18
concurrency issues, 349
conditions, defined, 37
continue statements
overview, 50–53
using in for loop, 55
Coordinated Universal Time (UTC), 336
coordinates
of computer screen, 415
of an image, 389–390
copy() function, 100–101, 135, 198, 394
copytree() function, 198–199
countdown project, 357–358
counting down, 357
overview, 357
playing sound file, 357–358
cProfile.run() function, 337
crashes, program, 14
create_sheet() method, 278
CRITICAL level, 224
cron, 354
cropping images, 393–394
CSS (Cascading Style Sheets)
matching with selenium module, 258
selectors, 246–247
CSV files
defined, 319
delimeter for, 324
format overview, 320
line terminator for, 324
Reader objects, 321
reading data in loop, 322
removing header from, 324–327
looping through CSV files, 325
overview, 324–325
reading in CSV file, 325–326
writing out CSV file, 326–327
Writer objects, 322–323
curly brackets ({}), 105, 162
greedy vs. nongreedy matching, 156–157
matching specific repetitions with, 156
current working directory, 175

D

\D character class, 158
\d character class, 158
%d directive, 344
data structures
algebraic chess notation, 112–113
tic-tac-toe board, 113–117
data types
Booleans, 32
defined, 16
dictionaries, 105–106
floating-point numbers, 17
integers, 17
list() function, 97
lists, 80
mutable vs. immutable, 94–96
None value, 65
strings, 17
tuple() function, 97
tuples, 96–97
datetime module
arithmetic using, 343
converting objects to strings, 344–345
converting strings to objects, 345
fromtimestamp() function, 341
now() function, 341
overview, 341–342, 346
pausing program until time, 344
timedelta data type, 342–343
total_seconds() method, 342
datetime objects, 341–342
converting to strings, 344–345
converting from strings to, 345
debug() function, 222
debugging
assertions, 219–221
defined, 4
getting traceback as string, 217–218
in IDLE
overview, 225–227
stepping through program, 227–229
using breakpoints, 229–231

debugging (*continued*)
logging
disabling, 224–225
to file, 225
levels of, 223–224
`logging` module, 221–223
`print()` function and, 223
raising exceptions, 216–217
`DEBUG` level, 223
decimal numbers. *See* floating-point numbers
`decode()` method, 374–375
decryption, of PDF files, 297–298
deduplicating code, 62
`deepcopy()` function, 100–101
`def` statements, 62
with parameters, 63
`DELETED` search key, 370
`delete_messages()` method, 375
deleting files/folders
permanently, 200–201
using `send2trash` module, 201–202
`del` statements, 84
dictionaries
`copy()` function, 100–101
`deepcopy()` function, 100–101
`get()` method, 109
`in` operator, 109
`items()` method, 107–108
`keys()` method, 107–108
lists vs., 106–107
nesting, 117–119
`not in` operator, 109
overview, 105–106
`setdefault()` method, 110–111
`values()` method, 107–108
directories
absolute vs. relative paths, 175–176
backslash vs. forward slash, 174–175
copying, 198–199
creating, 176
current working directory, 175
defined, 173–174
deleting permanently, 200–201
deleting using `send2trash` module, 201–202
moving, 199–200
`os.path` module
absolute paths in, 177–179
file sizes, 179–180
folder contents, 179–180
overview, 177
path validity, 180
relative paths in, 177–179
renaming, 199–200
walking, 202–203
`dirname()` function, 178
`disable()` function, 224
division (`/`) operator, 15, 88
`Document` objects, 307–308
dollar sign (`$`), 159–160, 162
dot character (`.`), 160–162
using in paths, 175–176
wildcard matches, 160–162
dot-star character (`.*`), 161
`doubleClick()` function, 420, 430
double quotes (`"`), 124
`double_strike` attribute, 311
downloading
files from web, 239–240
web pages, 237–238
XKCD comics, 251–256, 350–352
`DRAFT` search key, 370
dragging mouse, 420–422
`dragRel()` function, 420, 422, 430
`dragTo()` function, 420, 430
drawing on images
ellipses, 407
example program, 407–408
`ImageDraw` module, 406
lines, 406–407
points, 406
polygons, 407
rectangles, 407
text, 408–410
`dumps()` function, 329
`duration` keyword arguments, 416

E

`ehlo()` method, 364, 379
elements, HTML, 240
`elif` statements, 40–45
`ellipse()` method, 407
`else` statements, 39–40
email addresses, extracting, 165–169
creating regex, 166–167
finding matches on clipboard, 167–168
joining matches into a string, 168
overview, 165–166
emails
deleting, 375
disconnecting from server, 375–376
fetching

folders, 368–369
getting message content, 372–373
logging into server, 368
overview, 366–367
raw messages, 373–375
gmail_search() method, 372
IMAP, 366
marking message as read, 372–373
searching, 368–371
sending
connecting to SMTP server, 363–364
disconnecting from server, 366
logging into server, 364–365
overview, 362
reminder, 376–380
sending "hello" message, 364
sending message, 365
TLS encryption, 364
SMTP, 362
emboss attribute, 311
encryption, of PDF files, 302–303
endswith() method, 131
epoch timestamps, 336, 341, 346
equal to (==) operator, 33, 34
ERROR level, 224
errors
crashes and, 14
help for, 8–9
escape characters, 124–125
evaluation, defined, 14
Excel spreadsheets
application support, 265–266
charts in, 288–290
column width, 285–286
converting between column letters and numbers, 270
creating documents, 277
creating worksheets, 278
deleting worksheets, 278
font styles, 282–284
formulas in, 284–285
freezing panes, 287–288
getting cell values, 268–269
getting rows and columns, 270–272
getting worksheet names, 268
merging and unmerging cells, 286–287
opening documents, 267
openpyxl module, 266
overview, 266–267
reading files
overview, 272–273
populating data structure, 274–275
reading data, 273–274
writing results to file, 275–276
and reminder emails project, 376–380
row height, 285–286
saving workbooks, 277
updating, 279–281
overview, 279–280
setup, 280
workbooks vs., 266
writing values to cells, 278–279
Exception objects, 217
exceptions
assertions and, 219–221
getting traceback as string, 217–218
handling, 72–74
raising, 216–217
execution, program
defined, 31
overview, 38
pausing until specific time, 344
terminating program with sys.exit(), 58
exists() function, 180
exit codes, 353–354
expand keyword, 398
exponent (**) operator, 15
expressions
conditions and, 37
in interactive shell, 14–16
expunge() method, 375
extensions, file, 173
extractall() method, 205
extracting ZIP files, 205
extract() method, 205

F

FailSafeException exception, 434
"falsey" values, 53
fetch() method, 371, 372–373
file editor, 21
file management
absolute vs. relative paths, 175–176
backslash vs. forward slash, 174–175
compressed files
backing up to, 209–212
creating ZIP files, 205–206

file management (*continued*)
compressed files (*continued*)
extracting ZIP files, 205
overview, 203–204
reading ZIP files, 204
creating directories, 176
current working directory, 175
multiclipboard project, 191–193
opening files, 181–182
`os.path` module
absolute paths in, 177–179
file sizes, 179–180
folder contents, 179–180
overview, 177
path validity, 180
relative paths in, 177–179
overview, 173–174
paths, 173–174
plaintext vs. binary files, 180–181
reading files, 182–183
renaming files, date styles, 206–209
saving variables with `pformat()` function, 185–186
`send2trash` module, 201–202
shelve module, 184–185
`shutil` module
copying files/folders, 198–199
deleting files/folders, 200–201
moving files/folders, 199–200
renaming files/folders, 199–200
walking directory trees, 202–203
writing files, 183–184
filenames, defined, 173
`File` objects, 182
`findall()` method, 157–158
`find_element_by_*` methods, 257–258
`find_elements_by_*` methods, 257–258
Firefox, developer tools in, 243
`FLAGGED` search key, 370
flipping images, 398–399
`float()` function, 25–28
floating-point numbers
integer equivalence, 27
overview, 17
rounding, 338
flow control
binary operators, 35–36
blocks of code, 37–38
Boolean values and, 32–33
`break` statements, 49–50
comparison operators, 33–35
conditions, 37
`continue` statements, 50–53
`elif` statements, 40–45
`else` statements, 39–40
`if` statements, 38–39
overview, 31–32
using binary and comparison operators together, 36–37
`while` loops, 45–49
folders
absolute vs. relative paths, 175–176
backing up to ZIP file, 209–212
creating new ZIP file, 211
figuring out ZIP filename, 210–211
walking directory tree, 211–212
backslash vs. forward slash, 174–175
copying, 198–199
creating, 176
current working directory, 175
defined, 173–174
deleting permanently, 200–201
deleting using `send2trash` module, 201–202
moving, 199–200
`os.path` module
absolute paths in, 177–179
file sizes, 179–180
folder contents, 179–180
overview, 177
path validity, 180
relative paths in, 177–179
renaming, 199–200
walking directory trees, 202–203
`Font` objects, 282–283
font styles, in Excel spreadsheets, 282–284
`for` loops
overview, 53–56
using dictionary items in, 108
using lists with, 86
`format` attribute, 392
`format_description` attribute, 392
`formData` list, 434
form filler project, 430–437
overview, 430–431
radio buttons, 435–436
select lists, 435–436
setting up coordinates, 432–434
steps in process, 431
submitting form, 436–437
typing data, 434–435

formulas, in Excel spreadsheets, 284–285
forward() method, 261
forward slash (/), 174–175
FROM search key, 370
fromtimestamp() function, 341, 346
functions. *See also names of individual functions*
  arguments, 23, 63
  as "black box", 72
  built-in, 57
  def statements, 63
  exception handling, 72–74
  keyword arguments, 65–66
  None value and, 65
  overview, 61–62
  parameters, 63
  return values, 63–65

G

get_active_sheet() method, 268
get_addresses() method, 374
get_attribute() method, 258
getcolor() function, 388–389, 393
get_column_letter() function, 270
getcwd() function, 175
get() function
  overview, 109
  requests module, 237
get_highest_column() method, 269, 377
get_highest_row() method, 269
get_payload() method, 374–375
getpixel() function, 400, 423, 424
get_sheet_by_name() method, 268
get_sheet_names() method, 268
getsize() function, 179
get_subject() method, 374
getText() function, 308–309
GIF format, 392
global scope, 70–71
Gmail, 363, 365, 367
gmail_search() method, 372
Google Maps, 234–236
graphical user interface automation. *See* GUI (graphical user interface) automation
greater than (>) operator, 33
greater than or equal to (>=) operator, 33
greedy matching
  dot-star for, 161
  in regular expressions, 156–157
group() method, 151, 152–153
groups, regular expression matching
  greedy, 156–157
  nongreedy, 157
  one or more, 155–156
  optional, 154–155
  specific reptitions, 156
  zero or more, 155
  using parentheses, 152–153
  using pipe character in, 153–154
Guess the Number program, 74–76
GUI (graphical user interface) automation. *See also* form filler project
  controlling keyboard, 426–429
    hotkey combinations, 429
    key names, 427–428
    pressing and releasing, 428–429
    sending string from keyboard, 426–427
  controlling mouse, 415–417, 419–423
    clicking mouse, 420
    dragging mouse, 420–422
    scrolling mouse, 422–423
  determining mouse position, 417–419
  image recognition, 425–426
  installing pyautogui module, 414
  logging out of program, 414
  overview, 413–414
  screenshots, 423–424
  stopping program, 414–415

H

%H directive, 344
hash character (#), 126
headings, Word document, 314–315
help
  asking online, 9–10
  for error messages, 8–9
hotkey combinations, 429
hotkey() function, 429, 430
Hotmail.com, 363, 367
HTML (Hypertext Markup Language)
  browser developer tools and, 242–243
  finding elements, 244
  learning resources, 240
  overview, 240–241
  viewing page source, 241–242

I

%I directive, 344
id attribute, 241
IDLE (interactive development environment)
  creating programs, 21–22
  debugging in
    overview, 225–227
    stepping through program, 227–229
    using breakpoints, 229–231
  expressions in, 14–16
  overview, 8
  running scripts outside of, 136
  starting, 7–8
if statements
  overview, 38–39
  using in while loop, 46–47
imageDraw module, 406
imageDraw objects, 406–408
ImageFont objects, 408–410
Image objects, 391–399
images
  adding logo to, 401–405
  attributes for, 392–393
  box tuples, 390
  color values in, 388–389
  coordinates in, 389–390
  copying and pasting in, 394–396
  cropping, 393–394
  drawing on
    example program, 407–408
    ellipses, 407
    ImageDraw module, 406
    lines, 406–407
    points, 406
    polygons, 407
    rectangles, 407
    text, 408–410
  flipping, 398–399
  opening with Pillow, 390–391
  pixel manipulation, 400
  recognition of, 425–426
  resizing, 397
  RGBA values, 388–389
  rotating, 398–399
  transparent pixels, 397
IMAP (Internet Message Access Protocol)
  defined, 366
  deleting messages, 375
  disconnecting from server, 375–376
  fetching messages, 372–375
  folders, 368–369
  logging into server, 368
  searching messages, 368–371
imapclient module, 366
IMAPClient objects, 367–368
immutable data types, 94–96
importing modules
  overview, 57–58
  pyautogui module, 417
imprint attribute, 311
im variable, 423
indentation, 93
indexes
  for dictionaries. *See* keys, dictionary
  for lists
    changing values using, 83
    getting value using, 80–81
    negative, 82
    removing values from list using, 84
  for strings, 126–127
IndexError, 106
index() method, 89
infinite loops, 49, 51, 418
INFO level, 223
in operator
  using with dictionaries, 109
  using with lists, 87
  using with strings, 127
input() function
  overview, 23–24, 89–90
  using for sensitive information, 365
installing
  openpyxl module, 266
  pyautogui module, 414
  Python, 6–7
  selenium module, 256
  third-party modules, 441–442
int, 17. See also integers
integer division/floored quotient (//) operator, 15
integers
  floating-point equivalence, 27
  overview, 17
interactive development environment. *See* IDLE (interactive development environment)
interactive shell. *See* IDLE
Internet Explorer, developer tools in, 242–243

Internet Message Access Protocol. *See* IMAP (Internet Message Access Protocol)
interpreter, Python, 7
`int()` function, 25–28
`isabs()` function, 177
`isalnum()` method, 129–131
`isalpha()` method, 129–130
`isdecimal()` method, 129–131
`isdir()` function, 180
`is_displayed()` method, 258
`is_enabled()` method, 258
`isfile()` function, 180
`islower()` method, 128–129
`is_selected()` method, 258
`isspace()` method, 130
`istitle()` method, 130
`isupper()` method, 128–129
`italic` attribute, 311
`items()` method, 107–108
`iter_content()` method, 239–240

J

`%j` directive, 344
`join()` method, 131–132, 174–175, 177, 352
JPEG format, 392
JSON files
  APIs for, 327–328
  defined, 319–320
  format overview, 327–328
  reading, 328–329
  and weather data project, 329–332
  writing, 329
justifying text, 133–134

K

keyboard
  controlling, with PyAutoGUI
    hotkey combinations, 429
    pressing and releasing keys, 428–429
    sending string from keyboard, 426–427
  key names, 427–428
`KeyboardInterrupt` exception, 340, 417, 418
`keyDown()` function, 428, 429, 430
keys, dictionary, 105
`keys()` method, 107–108
`keyUp()` function, 428, 429, 430
keyword arguments, 65–66

L

`LARGER` search key, 370
launchd, 354–355
launching programs
  and countdown project, 357–358
  opening files with default applications, 355–356
  opening websites, 355
  overview, 352–354
  passing command line arguments to processes, 354
  `poll()` method, 353
  running Python scripts, 355
  scheduling, 354–355
  `sleep()` function, 355
  `wait()` method, 354
`len()` function, 307–308
  finding number of values in list, 83
  overview, 24–25
less than (`<`) operator, 33
less than or equal to (`<=`) operator, 33
LibreOffice, 265, 306
line breaks, Word document, 315
`LineChart()` function, 290
line continuation character (`\`), 93
`line()` method, 406–407
linked styles, 310
Linux
  backslash vs. forward slash, 174–175
  cron, 354
  installing Python, 7
  installing third-party modules, 442
  launching processes from Python, 353
  logging out of automation program, 414
  opening files with default applications, 355
  pip tool on, 441–442
  Python support, 4
  running Python programs on, 445
  starting IDLE, 8
  Unix philosophy, 356
`listdir()` function, 179
`list_folders()` method, 368–369
`list()` function, 321, 426

lists
append() method, 89–90
augmented assignment operators, 88–89
changing values using index, 83
concatenation of, 83
copy() function, 100–101
deepcopy() function, 100–101
dictionaries vs., 106–107
finding number of values using len(), 83
getting sublists with slices, 82–83
getting value using index, 80–81
index() method, 89
in operator, 87
insert() method, 89–90
list() function, 97
Magic 8 Ball example program using, 92–93
multiple assignment trick, 87–88
mutable vs. immutable data types, 94–96
negative indexes, 82
nesting, 117–119
not in operator, 87
overview, 80
remove() method, 90–91
removing values from, 84
replication of, 83
sort() method, 91–92
storing variables as, 84–85
using with for loops, 86
ljust() method, 133–134
load_workbook() function, 267
loads() function, 328–329, 331
local scope, 67–70
locateAllOnScreen() function, 426
locateOnScreen() function, 425
location attribute, 258
logging
disabling, 224–225
to file, 225
levels of, 223–224
print() function and, 223
logging module, 221–223
logging out, of automation program, 414
login() method, 364, 368, 379
logo, adding to an image, 401–406
looping over files, 402–403
opening logo image, 401–402
overview, 404
resizing image, 403–404
logout() method, 375–376
LogRecord objects, 221
loops
break statements, 49–50
continue statements, 50–53
for loop, 53–56
range() function for, 56–57
reading data from CSV file, 322
using lists with, 86
while loop, 45–49
lower() method, 128–129
lstrip() method, 134–135

M

%M directive, 344
%m directive, 344
Mac OS X. *See* OS X
Magic 8 Ball example program, 92–93
makedirs() function, 176, 403
maps, open when location is copied, 234–236
figuring out URL, 234–235
handling clipboard content, 236
handling command line argument, 235–236
launching browser, 236
overview, 234
Match objects, 151
math
operators for, 15
programming and, 4
mergePage() method, 302
Message objects, 381–382
methods
chaining calls, 398
defined, 89
dictionary
get() method, 109
items() method, 107–108
keys() method, 107–108
setdefault() method, 110–111
values() method, 107–108
list
append() method, 89–90
index() method, 89
insert() method, 89–90
remove() method, 90–91
sort() method, 91–92
string
center() method, 133–134
copy() method, 135
endswith() method, 131

isalnum() method, 129–131
isalpha() method, 129–130
isdecimal() method, 129–131
islower() method, 128–129
isspace() method, 130
istitle() method, 130
isupper() method, 128–129
join() method, 131–132
ljust() method, 133–134
lower() method, 128–129
lstrip() method, 134–135
paste() method, 135
rjust() method, 133–134
rstrip() method, 134–135
split() method, 131–133
startswith() method, 131
strip() method, 134–135
upper() method, 128–129
Microsoft Windows. *See* Windows OS
middleClick() function, 420, 430
modules
importing, 57–58
third-party, installing, 442
modulus/remainder (%) operator, 15, 88
Monty Python, 4
mouse
controlling, 415–417, 419–423
clicking mouse, 420
dragging mouse, 420–422
scrolling mouse, 422–423
determining position of, 417–419
locating, 417–419
getting coordinates, 418–419
handling KeyboardInterrupt exception, 418
importing pyautogui module, 418
infinite loop, 418
overview, 417
and pixels, identifying colors of, 424–425
mouseDown() function, 420, 430
mouse.position() function, 418
mouseUp() function, 430
move() function, 199–200
moveRel() function, 416, 417, 420, 430
moveTo() function, 416, 420, 430
moving files/folders, 199–200
multiclipboard project, 191–193
listing keywords, 193
loading keyword content, 193
overview, 191
saving clipboard content, 192
setting up shelf file, 192
multiline comments, 126
multiline strings, 125–126
multiple assignment trick, 87–88
multiplication (*) operator, 15, 83, 88
multithreading
concurrency issues, 349
downloading multiple images, , 350–352
creating and starting threads, 351–352
using downloadXkcd() function, 350–351
waiting for threads to end, 352
join() method, 352
overview, 347–348
passing arguments to threads, 348–349
start() method, 348, 349
Thread() function, 347–348
mutable data types, 94–96

N

NameError, 84
namelist() method, 204
negative character classes, 159
negative indexes, 82
nested lists and dictionaries, 117–119
newline keyword argument, 322
None value, 65
nongreedy matching
dot, star, and question mark for, 161
in regular expressions, 157
not equal to (!=) operator, 33
not in operator
using with dictionaries, 109
using with lists, 87
using with strings, 127
not operator, 36
NOT search key, 370
now() function, 341, 346

O

ON search key, 369
open() function, 181–182, 234, 355–356, 391
opening files, 181–182
OpenOffice, 265, 306
open program, 355

openpyxl module, installing, 266
operators
augmented assignment, 88–89
binary, 35–36
comparison, 33–35
defined, 14
math, 15
using binary and comparison operators together, 36–37
order of operations, 15
or operator, 36
OR search key, 370
OS X
backslash vs. forward slash, 174–175
installing Python, 7
installing third-party modules, 442
launchd, 354
launching processes from Python, 356
logging out of automation program, 414
opening files with default applications, 355–356
pip tool on, 441–442
Python support, 4
running Python programs on, 445
starting IDLE, 8
Unix philosophy, 356
outline attribute, 311
Outlook.com, 363, 367

P

%p directive, 344
page breaks, Word document, 315
Page objects, 297
Paragraph objects, 307
paragraphs, Word document, 309–310
parameters, function, 63
parentheses (), 96–97, 152–153
parsing, defined, 245
passing arguments, 23
passing references, 100
passwords
application-specific, 365
managing project, 136–138
command-line arguments, 137
copying password, 137–138
data structures, 136–137
overview, 136
pastebin.com, 10
paste() method, 135, 394, 395
paths
absolute vs. relative, 175–176
backslash vs. forward slash, 174–175
current working directory, 175
overview, 173–174
os.path module
absolute paths in, 177–179
file sizes, 179–180
folder contents, 179–180
overview, 177
path validity, 180
relative paths in, 177–179
PAUSE variable, 415, 434
PdfFileReader objects, 297–298
PDF files
combining pages from multiple files, 303–306
adding pages, 303
finding PDF files, 304
opening PDFs, 304–305
overview, 303
saving results, 305–306
creating, 298–299
decrypting, 297–298
encrypting, 302–303
extracting text from, 296–297
format overview, 295–296
pages in
copying, 299–300
overlaying, 301–302
rotating, 300–301
PdfFileWriter objects, 298–299
pformat() function
overview, 111–112
saving variables in text files using, 185–186
phone numbers, extracting, 165–169
creating regex, 166
finding matches on clipboard, 167–168
joining matches into a string, 168
overview, 165–166
Pillow
copying and pasting in images, 394–396
cropping images, 393–394
drawing on images
ellipses, 407
example program, 407–408
ImageDraw module, 406
lines, 406–407

points, 406
polygons, 407
rectangles, 407
text, 408–410
flipping images, 398–399
image attributes, 392–393
module, 388
opening images, 390–391
pixel manipulation, 400
resizing images, 397
rotating images, 398–399
transparent pixels, 397
pipe character (|), 153–154, 164–165
pip tool, 441–442
`pixelMatchesColor()` function, 424, 435
pixels, 388, 400
plaintext files, 180–181
plus sign (+), 155–156, 162
PNG format, 392
`point()` method, 406
`poll()` method, 353
`polygon()` method, 407
`Popen()` function, 352–353
opening files with default applications, 355–356
passing command line arguments to, 354
`position()` function, 417, 419
`pprint()` function, 111–112
precedence of math operators, 15
`press()` function, 429, 430, 436
`print()` function, 435
logging and, 223
overview, 23
passing multiple arguments to, 66
using variables with, 24
processes
and countdown project, 357–358
defined, 352
opening files with default applications, 355–356
opening websites, 355
passing command line arguments to, 354
`poll()` method, 353
`Popen()` function, 352–353
`wait()` method, 354
profiling code, 336–337
programming
blocks of code, 37–38
comments, 23
creativity needed for, 5
deduplicating code, 62
defined, 3
exception handling, 72–74
execution, program, 38
functions as "black boxes", 72
global scope, 70–71
indentation, 93
local scope, 67–70
math and, 4
Python, 4
terminating program with `sys.exit()`, 58
projects
Adding Bullets to Wiki Markup, 139–141
Adding a Logo, 401–406
Automatic Form Filler, 430–437
Backing Up a Folder into a ZIP File, 209–212
Combining Select Pages from Many PDFs, 303–306
Downloading All XKCD Comics, 251–256
Extending the mouseNow Program, 424–425
Fetching Current Weather Data, 329–332
Generating Random Quiz Files, 186–191
"I'm Feeling Lucky" Google Search, 248–251
"Just Text Me" Module, 383
mapIt.py with the `webbrowser` Module, 234–236
Multiclipboard, 191–193
Multithreaded XKCD Downloader, 350–352
Password Locker, 136–138
Phone Number and Email Address Extractor, 165–169
Reading Data from a Spreadsheet, 272–276
Removing the Header from CSV Files, 324–327
Renaming Files with American-Style Dates to European-Style Dates, 206–209
Sending Member Dues Reminder Emails, 376–379
Simple Countdown Program, 357–358
Super Stopwatch, 338–340

projects (*continued*)
Updating a Spreadsheet, 279–281
"Where Is the Mouse Right Now?", 417–419
putpixel() method, 400
pyautogui.click() function, 431
pyautogui.click() method, 420
pyautogui.doubleClick() function, 420
pyautogui.dragTo() function, 420
pyautogui.FailSafeException exception, 415
pyautogui.hotkey() function, 429
pyautogui.keyDown() function, 428
pyautogui.keyUp() function, 428
pyautogui.middleClick() function, 420
pyautogui module
form filler project, 430–437
controlling keyboard, 426–429
hotkey combinations, 429
key names, 427–428
pressing and releasing keys, 428–429
sending string from keyboard, 426–427
controlling mouse, 415–417, 419–423
clicking mouse, 420
dragging mouse, 420–422
scrolling mouse, 422–423
documentation for, 414
fail-safe feature, 415
functions, 430
image recognition, 425–426
importing, 417
installing, 414
pausing function calls, 415
screenshots, 423–424
pyautogui.mouseDown() function, 420
pyautogui.moveRel() function, 416, 417
pyautogui.moveTo() function, 416
pyautogui.PAUSE variable, 415
pyautogui.position() function, 419
pyautogui.press() function, 436
pyautogui.rightClick() function, 420
pyautogui.screenshot() function, 423
pyautogui.size() function, 416
pyautogui.typewrite() function, 426, 427, 431
py.exe program, 444–445
pyobjc module, 442
PyPDF2 module
combining pages from multiple PDFs, 303–306
creating PDFs, 298–299
decrypting PDFs, 297–298
encrypting PDFs, 302–303
extracting text from PDFs, 296–297
format overview, 295–296
pages in PDFs
copying, 299–300
overlaying, 301–302
rotating, 300–301
pyperclip module, 135
Python
data types, 16–17
downloading, 6
example program, 21–22
help, 8–9
installing, 6–7
interactive shell, 8
interpreter, defined, 7
math and, 4
overview, 4
programming overview, 3
starting IDLE, 7–8
python-docx module, 306
pyzmail module, 366, 373–375
PyzMessage objects, 373–375

Q

question mark (?), 154–155, 162
quit() method, 261, 366, 379
quiz generator, 186–190
creating quiz file, 188
creating answer options, 189
overview, 186
shuffling question order, 188
storing quiz data in dictionary, 187
writing content to files, 189–191

R

radio buttons, 435–436
raise_for_status() method, 238
raise keyword, 216–217
range() function, 56–57
raw strings, 125, 151
Reader objects, 321–322
reading files, 182–183, 204
readlines() method, 182

read() method, 182
rectangle() method, 407
Reddit, 9
Reference objects, 289
references
    overview, 97–99
    passing, 100
refresh() method, 261
Regex objects
    creating, 150
    matching, 151
regular expressions
    beginning of string matches, 159–160
    case sensitivity, 163
    character classes, 158–159
    creating Regex objects, 150–151
    defined, 147–148
    end of string matches, 159–160
    extracting phone numbers and emails addresses, 165–169
    findall() method, 157–158
    finding text without, 148–150
    greedy matching, 156–157
    grouping
        matching specific repetitions, 156
        one or more matches, 155–156
        optional matching, 154–155
        using parentheses, 152–153
        using pipe character in, 153–154
        zero or more matches, 155
    HTML and, 243
    matching with, 151–152
    multiple arguments for compile() function, 164–165
    nongreedy matching, 157
    patterns for, 150
    spreading over multiple lines, 164
    substituting strings using, 163–164
    symbol reference, 162
    wildcard character, 160–162
relative paths, 175–179
relpath() function, 177, 178
remainder/modulus (%) operator, 15, 88
remove() method, 90–91
remove_sheet() method, 278
renaming files/folders, 199–200
    date styles, 206–209
        creating regex for dates, 206
        identifying dates in filenames, 207–208
        overview, 206
        renaming files, 209
replication
    of lists, 83
    string, 18
requests module
    downloading files, 239–240
    downloading pages, 237–238
resolution of computer screen, 416
Response objects, 237–238
return values, function, 63–65
reverse keyword, 91
RGBA values, 388–389
RGB color model, 389
rightClick() function, 420, 430
rjust() method, 133–134, 419
rmdir() function, 200
rmtree() function, 200
rotateClockwise() method, 300
rotateCounterClockwise() method, 300
rotating images, 398–399
rounding numbers, 338
rows, in Excel spreadsheets
    setting height and width of, 285–286
    slicing Worksheet objects to get Cell objects in, 270–272
rstrip() method, 134–135
rtl attribute, 311
Run objects, 310, 311–312
running programs
    on Linux, 445
    on OS X, 445
    overview, 443
    on Windows, 444–445
    shebang line, 443–444

S

\S character class, 158
\s character class, 158
%S directive, 344
Safari, developer tools in, 243
save() method, 391
scope
    global, 70–71
    local, 67–70
screenshot() function, 423, 430
screenshots
    analyzing, 424
    getting, 423
scripts
    running from Python program, 355
    running outside of IDLE, 136

scroll() function, 422, 423, 430
scrolling mouse, 422–423
searching
  email, 368–371
  the Web, 248–251
    finding results, 249–250
    getting command line arguments, 249
    opening web browser for results, 250–251
    overview, 248
    requesting search page, 249
search() method, 151
SEEN search key, 370
see program, 355
select_folder() method, 369
select lists, 435–436
select() method, bs4 module, 246–247
selectors, CSS, 246–247, 258
selenium module
  clicking buttons, 261
  finding elements, 257–259
  following links, 259
  installing, 256
  sending special keystrokes, 260–261
  submitting forms, 259–260
  using Firefox with, 256–257
send2trash module, 201–202
sending reminder emails, 376–379
  finding unpaid members, 378
  opening Excel file, 376–377
  overview, 376
  sending emails, 378–379
send_keys() method, 259–260
sendmail() method, 365, 379
sequence numbers, 373
sequences, 86
setdefault() method, 110–111
shadow attribute, 311
shebang line, 443–444
shelve module, 184–185
Short Message Service (SMS)
  sending messages, 381–382
  Twilio service, 380
shutil module
  deleting files/folders, 200–201
  moving files/folders, 199–200
  renaming files/folders, 199–200
SID (string ID), 382
Simple Mail Transfer Protocol. *See* SMTP (Simple Mail Transfer Protocol)
SINCE search key, 369
single quote ('), 124
single-threaded programs, 347
size() function, 416
sleep() function, 337–338, 344, 346, 355
slices
  getting sublists with, 82–83
  for strings, 126–127
small_caps attribute, 311
SMALLER search key, 370
SMS (Short Message Service)
  sending messages, 381–382
  Twilio service, 380
SMTP (Simple Mail Transfer Protocol)
  connecting to server, 363–364
  defined, 362
  disconnecting from server, 366
  logging into server, 364–365
  sending "hello" message, 364
  sending message, 365
  TLS encryption, 364
SMTP objects, 363–364
sort() method, 91–92
sound files, playing, 357–358
source code, defined, 3
split() method, 131–133, 178, 320
spreadsheets. *See* Excel spreadsheets
square brackets [], 80
Stack Overflow, 9
standard library, 57
star (*), 161, 162
  using with wildcard character, 161
  zero or more matches with, 155
start() method, 348, 349, 351
start program, 355
startswith() method, 131
starttls() method, 364, 379
step argument, 56
stopwatch project, 338–340
  overview, 338–339
  set up, 339
  tracking lap times, 339–340
strftime() function, 344–345, 346
str() function, 25–28, 97, 419
strike attribute, 311
string ID (SID), 382
strings
  center() method, 133–134
  concatenation, 17–18
  converting datetime objects to, 344–345
  converting to datetime objects, 345

copying and pasting, 135
double quotes for, 124
endswith() method, 131
escape characters, 124–125
extracting PDF text as, 296–297
getting traceback as, 217–218
indexes for, 126–127
in operator, 127
isalnum() method, 129–131
isalpha() method, 129–130
isdecimal() method, 129–131
islower() method, 128–129
isspace() method, 130
istitle() method, 130
isupper() method, 128–129
join() method, 131–132
literals, 124
ljust() method, 133–134
lower() method, 128–129
lstrip() method, 134–135
multiline, 125–126
mutable vs. immutable data types, 94–96
not in operator, 127
overview, 17
raw, 125
replication of, 18
rjust() method, 133–134
rstrip() method, 134–135
slicing, 126–127
split() method, 131–133
startswith() method, 131
strip() method, 134–135
substituting using regular expressions, 163–164
upper() method, 128–129
strip() method, 134–135
strptime() function, 345, 346
strs, 17, *See also* strings
Style objects, 282–283
SUBJECT search key, 369
sublists, getting with slices, 82–83
sub() method, 163–164
submitButtonColor variable, 432, 435
submitButton variable, 432
submit() method, 260
subprocess module, 335, 352–354
subtraction (-) operator, 15, 88
subtractive color model, 389
Sudoku puzzles, 4
sys.exit() function, 58

T

tag_name attribute, 258
Tag objects, 246–247
tags, HTML, 240
Task Scheduler, 354
termination, program, 22, 58
text attribute, 308, 311
text messaging
automatic notifications, 383
sending messages, 381–382
Twilio service, 380
text() method, 408–410
TEXT search key, 369
textsize() method, 409
third-party modules, installing, 441–442
Thread() function, 347–348, 351
threading module, 335, 347
Thread objects, 347–348
threads
concurrency issues, 349
join() method, 352
multithreading, 347–348
image downloader, 350–352
passing arguments to, 348–349
processes vs., 352
tic-tac-toe board, 113–117
timedelta data type, 342–343, 346
timedelta objects, 342–343
time module
overview, 346
sleep() function, 337–338, 344
stopwatch project, 338–340
time() function, 336–337
TLS encryption, 364
top-level domains, 167
TO search key, 370
total_seconds() method, 342, 346
traceback, getting from error, 217–218
transparency, 388, 397
transpose() method, 399
triple quotes ('''), 125, 164
truetype() function, 409
truth tables, 35–36
“truthy” values, 53
tuple data type
overview, 96–97
tuple() function, 97
twilio module, 380
TwilioRestClient objects, 381

Twilio service
automatic text messages, 383
overview, 380
sending text messages, 381–382
TypeError, 81, 94
typewrite() function, 426, 427, 430, 431, 435, 436

U

Ubuntu, 7
cron, 354–355
launching processes from Python, 353
opening files with default applications, 355
Unix philosophy, 356
UNANSWERED search key, 370
UNDELETED search key, 370
underline attribute, 311
underscore (_), 20
UNDRAFT search key, 370
UNFLAGGED search key, 370
Unicode encodings, 239
Unix epoch, 336, 341, 346
Unix philosophy, 356
unlink() function, 200
UNSEEN search key, 370
upper() method, 128–129
UTC (Coordinated Universal Time), 336

V

ValueError, 88, 345
values, defined, 14, 150
values() method, 107–108
variables. *See also* lists
assignment statements, 18–19
defined, 18
global, 70–71
initializing, 19
local, 67–70
naming, 20–21
None value and, 65
overwriting, 19–20
references, 97–99
saving with shelve module, 184–185
storing as list, 84–85
Verizon mail, 363, 367
volumes, defined, 174

W

\W character class, 158
\w character class, 158
%w directive, 344
walk() function, 202–203, 354
WARNING level, 223
weather data, fetching, 329–332
downloading JSON data, 330–331
getting location, 330
loading JSON data, 331–332
overview, 329
webbrowser module
open() function, 355
opening browser using, 234–236
WebDriver objects, 257
WebElement objects, 257–258
web scraping
bs4 module
creating object from HTML, 245–246
finding element with select() method, 246–247
getting attribute, 248
overview, 245
downloading
files, 239–240
images, 251–256
pages, 237–238
and Google maps project, 234–236
and Google search project, 248–251
HTML
browser developer tools and, 242–243
finding elements, 244
learning resources, 240
overview, 240–241
viewing page source, 241–242
overview, 233–234
requests module, 237–238
selenium module
clicking buttons, 261
finding elements, 257–259
following links, 259
installing, 256
sending special keystrokes, 260–261
submitting forms, 259–260
using Firefox with, 256–257
websites, opening from script, 355

while loops
getting and printing mouse coordinates using, 418
infinite, 418
overview, 45–49
whitespace, removing, 134–135
wildcard character (.), 160–162
Windows OS
backslash vs. forward slash, 174–175
installing Python, 6–7
installing third-party modules, 442
launching processes from Python, 353
logging out of automation program, 414
opening files with default applications, 355–356
pip tool on, 441–442
Python support, 4
running Python programs on, 444–445
starting IDLE, 7
Task Scheduler, 354
Word documents
adding headings, 314–315
creating documents with nondefault styles, 310–311
format overview, 306–307
getting text from, 308–309
line/page breaks, 315
pictures in, 315–316
python-docx module, 306
reading, 307–308
Run object attributes, 311–312
styling paragraphs, 309–310
writing to file, 312–314
Workbook objects, 267
workbooks, Excel, 266
creating worksheets, 278
deleting worksheets, 278
opening, 267
saving, 277
Worksheet objects, 268
write() method, 183–184
Writer objects, 322–323
writerow() method, 323

X

XKCD comics
downloading project, 251–256
designing program, 252–253
downloading web page, 253–254
overview, 251–252
saving image, 255–256
multithreaded downloading project, 350–352
creating and starting threads, 351–352
using downloadXkcd() function, 350–351
waiting for threads to end, 352

Y

%Y directive, 344
%y directive, 344
Yahoo! Mail, 363, 367

Z

zipfile module
creating ZIP files, 205–206
extracting ZIP files, 205
and folders, 209–212
overview, 203–204
reading ZIP files, 204
ZipFile objects, 204–205
ZipInfo objects, 204