

# CONTENTS IN DETAIL

**ACKNOWLEDGMENTS** **xxvii**

**INTRODUCTION** **xxix**

Who Is This Book For? . . . . .	xxx
What Can You Expect to Learn? . . . . .	xxx
Why Python? . . . . .	xxxi

## **PART I: BASICS** **1**

### **1** **GETTING STARTED** **3**

Setting Up Your Programming Environment . . . . .	3
Python 2 and Python 3. . . . .	4
Running Snippets of Python Code . . . . .	4
Hello World! . . . . .	4
Python on Different Operating Systems . . . . .	5
Python on Linux. . . . .	5
Python on OS X . . . . .	8
Python on Windows . . . . .	10
Troubleshooting Installation Issues . . . . .	15
Running Python Programs from a Terminal. . . . .	16
On Linux and OS X. . . . .	16
On Windows . . . . .	16
<i>Exercise 1-1: python.org</i> . . . . .	17
<i>Exercise 1-2: Hello World Typos</i> . . . . .	17
<i>Exercise 1-3: Infinite Skills</i> . . . . .	17
Summary . . . . .	17

### **2** **VARIABLES AND SIMPLE DATA TYPES** **19**

What Really Happens When You Run <code>hello_world.py</code> . . . . .	19
Variables . . . . .	20
Naming and Using Variables . . . . .	21
Avoiding Name Errors When Using Variables . . . . .	21
<i>Exercise 2-1: Simple Message</i> . . . . .	23
<i>Exercise 2-2: Simple Messages</i> . . . . .	23
Strings . . . . .	23
Changing Case in a String with Methods. . . . .	24
Combining or Concatenating Strings . . . . .	25
Adding Whitespace to Strings with Tabs or Newlines . . . . .	26
Stripping Whitespace . . . . .	26
Avoiding Syntax Errors with Strings . . . . .	28
Printing in Python 2 . . . . .	29
<i>Exercise 2-3: Personal Message</i> . . . . .	29
<i>Exercise 2-4: Name Cases</i> . . . . .	29
<i>Exercise 2-5: Famous Quote</i> . . . . .	29

<i>Exercise 2-6: Famous Quote 2</i> . . . . .	29
<i>Exercise 2-7: Stripping Names</i> . . . . .	29
Numbers . . . . .	30
Integers . . . . .	30
Floats . . . . .	30
Avoiding Type Errors with the <code>str()</code> Function . . . . .	31
Integers in Python 2 . . . . .	32
<i>Exercise 2-8: Number Eight</i> . . . . .	33
<i>Exercise 2-9: Favorite Number</i> . . . . .	33
Comments . . . . .	33
How Do You Write Comments? . . . . .	33
What Kind of Comments Should You Write? . . . . .	33
<i>Exercise 2-10: Adding Comments</i> . . . . .	34
The Zen of Python . . . . .	34
<i>Exercise 2-11: Zen of Python</i> . . . . .	36
Summary . . . . .	36

### **3**

## **INTRODUCING LISTS**

## **37**

What Is a List? . . . . .	37
Accessing Elements in a List . . . . .	38
Index Positions Start at 0, Not 1 . . . . .	39
Using Individual Values from a List . . . . .	39
<i>Exercise 3-1: Names</i> . . . . .	40
<i>Exercise 3-2: Greetings</i> . . . . .	40
<i>Exercise 3-3: Your Own List</i> . . . . .	40
Changing, Adding, and Removing Elements . . . . .	40
Modifying Elements in a List . . . . .	40
Adding Elements to a List . . . . .	41
Removing Elements from a List . . . . .	42
<i>Exercise 3-4: Guest List</i> . . . . .	46
<i>Exercise 3-5: Changing Guest List</i> . . . . .	46
<i>Exercise 3-6: More Guests</i> . . . . .	46
<i>Exercise 3-7: Shrinking Guest List</i> . . . . .	47
Organizing a List . . . . .	47
Sorting a List Permanently with the <code>sort()</code> Method . . . . .	47
Sorting a List Temporarily with the <code>sorted()</code> Function . . . . .	48
Printing a List in Reverse Order . . . . .	49
Finding the Length of a List . . . . .	49
<i>Exercise 3-8: Seeing the World</i> . . . . .	50
<i>Exercise 3-9: Dinner Guests</i> . . . . .	50
<i>Exercise 3-10: Every Function</i> . . . . .	50
Avoiding Index Errors When Working with Lists . . . . .	50
<i>Exercise 3-11: Intentional Error</i> . . . . .	52
Summary . . . . .	52

### **4**

## **WORKING WITH LISTS**

## **53**

Looping Through an Entire List . . . . .	53
A Closer Look at Looping . . . . .	54
Doing More Work Within a <code>for</code> Loop . . . . .	55
Doing Something After a <code>for</code> Loop . . . . .	56

Avoiding Indentation Errors . . . . .	57
Forgetting to Indent . . . . .	57
Forgetting to Indent Additional Lines . . . . .	58
Indenting Unnecessarily . . . . .	59
Indenting Unnecessarily After the Loop . . . . .	59
Forgetting the Colon . . . . .	60
<i>Exercise 4-1: Pizzas</i> . . . . .	60
<i>Exercise 4-2: Animals</i> . . . . .	60
Making Numerical Lists . . . . .	61
Using the range() Function . . . . .	61
Using range() to Make a List of Numbers . . . . .	62
Simple Statistics with a List of Numbers . . . . .	63
List Comprehensions . . . . .	63
<i>Exercise 4-3: Counting to Twenty</i> . . . . .	64
<i>Exercise 4-4: One Million</i> . . . . .	64
<i>Exercise 4-5: Summing a Million</i> . . . . .	64
<i>Exercise 4-6: Odd Numbers</i> . . . . .	64
<i>Exercise 4-7: Threes</i> . . . . .	64
<i>Exercise 4-8: Cubes</i> . . . . .	64
<i>Exercise 4-9: Cube Comprehension</i> . . . . .	64
Working with Part of a List . . . . .	65
Slicing a List . . . . .	65
Looping Through a Slice . . . . .	66
Copying a List . . . . .	67
<i>Exercise 4-10: Slices</i> . . . . .	69
<i>Exercise 4-11: My Pizzas, Your Pizzas</i> . . . . .	69
<i>Exercise 4-12: More Loops</i> . . . . .	69
Tuples . . . . .	69
Defining a Tuple . . . . .	69
Looping Through All Values in a Tuple . . . . .	70
Writing over a Tuple . . . . .	71
<i>Exercise 4-13: Buffet</i> . . . . .	71
Styling Your Code . . . . .	72
The Style Guide . . . . .	72
Indentation . . . . .	72
Line Length . . . . .	73
Blank Lines . . . . .	73
Other Style Guidelines . . . . .	73
<i>Exercise 4-14: PEP 8</i> . . . . .	74
<i>Exercise 4-15: Code Review</i> . . . . .	74
Summary . . . . .	74

## **5 IF STATEMENTS 75**

A Simple Example . . . . .	76
Conditional Tests . . . . .	76
Checking for Equality . . . . .	76
Ignoring Case When Checking for Equality . . . . .	77
Checking for Inequality . . . . .	78
Numerical Comparisons . . . . .	78
Checking Multiple Conditions . . . . .	79
Checking Whether a Value Is in a List . . . . .	80

Checking Whether a Value Is Not in a List . . . . .	81
Boolean Expressions . . . . .	81
<i>Exercise 5-1: Conditional Tests</i> . . . . .	82
<i>Exercise 5-2: More Conditional Tests</i> . . . . .	82
if Statements . . . . .	82
Simple if Statements . . . . .	82
if-else Statements . . . . .	83
The if-elif-else Chain . . . . .	84
Using Multiple elif Blocks . . . . .	86
Omitting the else Block . . . . .	86
Testing Multiple Conditions . . . . .	87
<i>Exercise 5-3: Alien Colors #1</i> . . . . .	88
<i>Exercise 5-4: Alien Colors #2</i> . . . . .	88
<i>Exercise 5-5: Alien Colors #3</i> . . . . .	89
<i>Exercise 5-6: Stages of Life</i> . . . . .	89
<i>Exercise 5-7: Favorite Fruit</i> . . . . .	89
Using if Statements with Lists . . . . .	89
Checking for Special Items . . . . .	90
Checking That a List Is Not Empty . . . . .	91
Using Multiple Lists . . . . .	92
<i>Exercise 5-8: Hello Admin</i> . . . . .	93
<i>Exercise 5-9: No Users</i> . . . . .	93
<i>Exercise 5-10: Checking Usernames</i> . . . . .	93
<i>Exercise 5-11: Ordinal Numbers</i> . . . . .	93
Styling Your if Statements . . . . .	94
<i>Exercise 5-12: Styling if statements</i> . . . . .	94
<i>Exercise 5-13: Your Ideas</i> . . . . .	94
Summary . . . . .	94

## **6**

### **DICTIONARIES** **95**

A Simple Dictionary . . . . .	96
Working with Dictionaries . . . . .	96
Accessing Values in a Dictionary . . . . .	97
Adding New Key-Value Pairs . . . . .	97
Starting with an Empty Dictionary . . . . .	98
Modifying Values in a Dictionary . . . . .	99
Removing Key-Value Pairs . . . . .	100
A Dictionary of Similar Objects . . . . .	100
<i>Exercise 6-1: Person</i> . . . . .	102
<i>Exercise 6-2: Favorite Numbers</i> . . . . .	102
<i>Exercise 6-3: Glossary</i> . . . . .	102
Looping Through a Dictionary . . . . .	102
Looping Through All Key-Value Pairs . . . . .	103
Looping Through All the Keys in a Dictionary . . . . .	104
Looping Through a Dictionary's Keys in Order . . . . .	106
Looping Through All Values in a Dictionary . . . . .	107
<i>Exercise 6-4: Glossary 2</i> . . . . .	108
<i>Exercise 6-5: Rivers</i> . . . . .	108
<i>Exercise 6-6: Polling</i> . . . . .	108
Nesting . . . . .	109
A List of Dictionaries . . . . .	109
A List in a Dictionary . . . . .	111

A Dictionary in a Dictionary . . . . .	113
<i>Exercise 6-7: People</i> . . . . .	114
<i>Exercise 6-8: Pets</i> . . . . .	115
<i>Exercise 6-9: Favorite Places</i> . . . . .	115
<i>Exercise 6-10: Favorite Numbers</i> . . . . .	115
<i>Exercise 6-11: Cities</i> . . . . .	115
<i>Exercise 6-12: Extensions</i> . . . . .	115
Summary . . . . .	115

## **7**

### **USER INPUT AND WHILE LOOPS** **117**

How the input() Function Works . . . . .	118
Writing Clear Prompts . . . . .	118
Using int() to Accept Numerical Input . . . . .	119
The Modulo Operator . . . . .	120
Accepting Input in Python 2.7 . . . . .	121
<i>Exercise 7-1: Rental Car</i> . . . . .	121
<i>Exercise 7-2: Restaurant Seating</i> . . . . .	121
<i>Exercise 7-3: Multiples of Ten</i> . . . . .	121
Introducing while Loops . . . . .	122
The while Loop in Action . . . . .	122
Letting the User Choose When to Quit . . . . .	122
Using a Flag . . . . .	124
Using break to Exit a Loop . . . . .	125
Using continue in a Loop . . . . .	126
Avoiding Infinite Loops . . . . .	126
<i>Exercise 7-4: Pizza Toppings</i> . . . . .	127
<i>Exercise 7-5: Movie Tickets</i> . . . . .	127
<i>Exercise 7-6: Three Exits</i> . . . . .	128
<i>Exercise 7-7: Infinity</i> . . . . .	128
Using a while Loop with Lists and Dictionaries . . . . .	128
Moving Items from One List to Another . . . . .	128
Removing All Instances of Specific Values from a List . . . . .	129
Filling a Dictionary with User Input . . . . .	130
<i>Exercise 7-8: Deli</i> . . . . .	131
<i>Exercise 7-9: No Pastrami</i> . . . . .	131
<i>Exercise 7-10: Dream Vacation</i> . . . . .	131
Summary . . . . .	131

## **8**

### **FUNCTIONS** **133**

Defining a Function . . . . .	134
Passing Information to a Function . . . . .	134
Arguments and Parameters . . . . .	135
<i>Exercise 8-1: Message</i> . . . . .	135
<i>Exercise 8-2: Favorite Book</i> . . . . .	135
Passing Arguments . . . . .	135
Positional Arguments . . . . .	136
Keyword Arguments . . . . .	137
Default Values . . . . .	138

Equivalent Function Calls . . . . .	139
Avoiding Argument Errors . . . . .	140
<i>Exercise 8-3: T-Shirt</i> . . . . .	141
<i>Exercise 8-4: Large Shirts</i> . . . . .	141
<i>Exercise 8-5: Cities</i> . . . . .	141
Return Values. . . . .	141
Returning a Simple Value . . . . .	142
Making an Argument Optional . . . . .	142
Returning a Dictionary . . . . .	144
Using a Function with a while Loop . . . . .	145
<i>Exercise 8-6: City Names</i> . . . . .	146
<i>Exercise 8-7: Album</i> . . . . .	146
<i>Exercise 8-8: User Albums</i> . . . . .	146
Passing a List . . . . .	147
Modifying a List in a Function . . . . .	147
Preventing a Function from Modifying a List . . . . .	149
<i>Exercise 8-9: Magicians</i> . . . . .	150
<i>Exercise 8-10: Great Magicians</i> . . . . .	150
<i>Exercise 8-11: Unchanged Magicians</i> . . . . .	150
Passing an Arbitrary Number of Arguments. . . . .	151
Mixing Positional and Arbitrary Arguments . . . . .	152
Using Arbitrary Keyword Arguments . . . . .	152
<i>Exercise 8-12: Sandwiches</i> . . . . .	154
<i>Exercise 8-13: User Profile</i> . . . . .	154
<i>Exercise 8-14: Cars</i> . . . . .	154
Storing Your Functions in Modules . . . . .	154
Importing an Entire Module . . . . .	154
Importing Specific Functions . . . . .	156
Using as to Give a Function an Alias . . . . .	156
Using as to Give a Module an Alias . . . . .	157
Importing All Functions in a Module . . . . .	157
Styling Functions . . . . .	158
<i>Exercise 8-15: Printing Models</i> . . . . .	159
<i>Exercise 8-16: Imports</i> . . . . .	159
<i>Exercise 8-17: Styling Functions</i> . . . . .	159
Summary . . . . .	159

## 9

### CLASSES

**161**

Creating and Using a Class. . . . .	162
Creating the Dog Class . . . . .	162
Making an Instance from a Class . . . . .	164
<i>Exercise 9-1: Restaurant</i> . . . . .	166
<i>Exercise 9-2: Three Restaurants</i> . . . . .	166
<i>Exercise 9-3: Users</i> . . . . .	166
Working with Classes and Instances. . . . .	167
The Car Class. . . . .	167
Setting a Default Value for an Attribute . . . . .	168
Modifying Attribute Values . . . . .	168
<i>Exercise 9-4: Number Served</i> . . . . .	171
<i>Exercise 9-5: Login Attempts</i> . . . . .	171

Inheritance . . . . .	172
The <code>__init__()</code> Method for a Child Class . . . . .	172
Inheritance in Python 2.7 . . . . .	173
Defining Attributes and Methods for the Child Class . . . . .	174
Overriding Methods from the Parent Class . . . . .	175
Instances as Attributes . . . . .	175
Modeling Real-World Objects . . . . .	177
<i>Exercise 9-6: Ice Cream Stand</i> . . . . .	178
<i>Exercise 9-7: Admin</i> . . . . .	178
<i>Exercise 9-8: Privileges</i> . . . . .	178
<i>Exercise 9-9: Battery Upgrade</i> . . . . .	178
Importing Classes . . . . .	179
Importing a Single Class . . . . .	179
Storing Multiple Classes in a Module . . . . .	180
Importing Multiple Classes from a Module . . . . .	181
Importing an Entire Module . . . . .	182
Importing All Classes from a Module . . . . .	182
Importing a Module into a Module . . . . .	183
Finding Your Own Workflow . . . . .	184
<i>Exercise 9-10: Imported Restaurant</i> . . . . .	184
<i>Exercise 9-11: Imported Admin</i> . . . . .	184
<i>Exercise 9-12: Multiple Modules</i> . . . . .	184
The Python Standard Library . . . . .	184
<i>Exercise 9-13: OrderedDict Rewrite</i> . . . . .	186
<i>Exercise 9-14: Dice</i> . . . . .	186
<i>Exercise 9-15: Python Module of the Week</i> . . . . .	186
Styling Classes . . . . .	186
Summary . . . . .	187

## **10 FILES AND EXCEPTIONS 189**

Reading from a File . . . . .	190
Reading an Entire File . . . . .	190
File Paths . . . . .	191
Reading Line by Line . . . . .	193
Making a List of Lines from a File . . . . .	194
Working with a File's Contents . . . . .	194
Large Files: One Million Digits . . . . .	195
Is Your Birthday Contained in Pi? . . . . .	196
<i>Exercise 10-1: Learning Python</i> . . . . .	197
<i>Exercise 10-2: Learning C</i> . . . . .	197
Writing to a File . . . . .	197
Writing to an Empty File . . . . .	197
Writing Multiple Lines . . . . .	198
Appending to a File . . . . .	199
<i>Exercise 10-3: Guest</i> . . . . .	199
<i>Exercise 10-4: Guest Book</i> . . . . .	199
<i>Exercise 10-5: Programming Poll</i> . . . . .	199
Exceptions . . . . .	200
Handling the <code>ZeroDivisionError</code> Exception . . . . .	200
Using <code>try-except</code> Blocks . . . . .	200
Using Exceptions to Prevent Crashes . . . . .	201

The else Block . . . . .	202
Handling the FileNotFoundError Exception . . . . .	203
Analyzing Text . . . . .	204
Working with Multiple Files . . . . .	205
Failing Silently . . . . .	206
Deciding Which Errors to Report . . . . .	207
<i>Exercise 10-6: Addition</i> . . . . .	207
<i>Exercise 10-7: Addition Calculator</i> . . . . .	208
<i>Exercise 10-8: Cats and Dogs</i> . . . . .	208
<i>Exercise 10-9: Silent Cats and Dogs</i> . . . . .	208
<i>Exercise 10-10: Common Words</i> . . . . .	208
Storing Data . . . . .	208
Using json.dump() and json.load(). . . . .	209
Saving and Reading User-Generated Data . . . . .	210
Refactoring . . . . .	212
<i>Exercise 10-11: Favorite Number</i> . . . . .	214
<i>Exercise 10-12: Favorite Number Remembered</i> . . . . .	214
<i>Exercise 10-13: Verify User</i> . . . . .	214
Summary . . . . .	214

**11 TESTING YOUR CODE 215**

Testing a Function . . . . .	216
Unit Tests and Test Cases . . . . .	217
A Passing Test . . . . .	217
A Failing Test . . . . .	218
Responding to a Failed Test . . . . .	219
Adding New Tests . . . . .	221
<i>Exercise 11-1: City, Country</i> . . . . .	222
<i>Exercise 11-2: Population</i> . . . . .	222
Testing a Class . . . . .	222
A Variety of Assert Methods . . . . .	222
A Class to Test . . . . .	223
Testing the AnonymousSurvey Class . . . . .	225
The setUp() Method . . . . .	227
<i>Exercise 11-3: Employee</i> . . . . .	228
Summary . . . . .	228

**PART II: PROJECTS 231**

**PROJECT 1: ALIEN INVASION**

**12 A SHIP THAT FIRES BULLETS 235**

Planning Your Project . . . . .	236
Installing Pygame . . . . .	236
Installing Python Packages with pip . . . . .	237
Installing Pygame on Linux . . . . .	238



Installing Pygame on OS X . . . . .	239
Installing Pygame on Windows . . . . .	240
Starting the Game Project . . . . .	240
Creating a Pygame Window and Responding to User Input . . . . .	241
Setting the Background Color . . . . .	242
Creating a Settings Class . . . . .	243
Adding the Ship Image . . . . .	244
Creating the Ship Class . . . . .	245
Drawing the Ship to the Screen . . . . .	246
Refactoring: the game_functions Module . . . . .	247
The check_events() Function . . . . .	247
The update_screen() Function . . . . .	248
<i>Exercise 12-1: Blue Sky</i> . . . . .	249
<i>Exercise 12-2: Game Character</i> . . . . .	249
Piloting the Ship . . . . .	249
Responding to a Keypress . . . . .	249
Allowing Continuous Movement . . . . .	250
Moving Both Left and Right . . . . .	252
Adjusting the Ship's Speed . . . . .	253
Limiting the Ship's Range . . . . .	255
Refactoring check_events() . . . . .	255
A Quick Recap . . . . .	256
alien_invasion.py . . . . .	256
settings.py . . . . .	256
game_functions.py . . . . .	256
ship.py . . . . .	257
<i>Exercise 12-3: Rocket</i> . . . . .	257
<i>Exercise 12-4: Keys</i> . . . . .	257
Shooting Bullets . . . . .	257
Adding the Bullet Settings . . . . .	257
Creating the Bullet Class . . . . .	258
Storing Bullets in a Group . . . . .	259
Firing Bullets . . . . .	260
Deleting Old Bullets . . . . .	261
Limiting the Number of Bullets . . . . .	262
Creating the update_bullets() Function . . . . .	263
Creating the fire_bullet() Function . . . . .	264
<i>Exercise 12-5: Sideways Shooter</i> . . . . .	264
Summary . . . . .	264

## 13 ALIENS!

**265**

Reviewing Your Project . . . . .	266
Creating the First Alien . . . . .	266
Creating the Alien Class . . . . .	267
Creating an Instance of the Alien . . . . .	268
Making the Alien Appear Onscreen . . . . .	268
Building the Alien Fleet . . . . .	269
Determining How Many Aliens Fit in a Row . . . . .	269
Creating Rows of Aliens . . . . .	270
Creating the Fleet . . . . .	271

Refactoring create_fleet()	273
Adding Rows	273
Exercise 13-1: Stars	276
Exercise 13-2: Better Stars	276
Making the Fleet Move	276
Moving the Aliens Right	276
Creating Settings for Fleet Direction	277
Checking to See Whether an Alien Has Hit the Edge	278
Dropping the Fleet and Changing Direction	278
Exercise 13-3: Raindrops	279
Exercise 13-4: Steady Rain	279
Shooting Aliens	280
Detecting Bullet Collisions	280
Making Larger Bullets for Testing	281
Repopulating the Fleet	282
Speeding Up the Bullets	283
Refactoring update_bullets()	283
Exercise 13-5: Catch	284
Ending the Game	284
Detecting Alien-Ship Collisions	284
Responding to Alien-Ship Collisions	285
Aliens that Reach the Bottom of the Screen	288
Game Over!	288
Identifying When Parts of the Game Should Run	289
Exercise 13-6: Game Over	290
Summary	290

## 14

### SCORING

**291**

Adding the Play Button	292
Creating a Button Class	292
Drawing the Button to the Screen	294
Starting the Game	295
Resetting the Game	296
Deactivating the Play Button	297
Hiding the Mouse Cursor	298
Exercise 14-1: Press P to Play	298
Exercise 14-2: Target Practice	298
Leveling Up	299
Modifying the Speed Settings	299
Resetting the Speed	300
Exercise 14-3: Challenging Target Practice	301
Scoring	301
Displaying the Score	301
Making a Scoreboard	303
Updating the Score as Aliens Are Shot Down	304
Making Sure to Score All Hits	305
Increasing Point Values	306
Rounding the Score	307
High Scores	308

Displaying the Level . . . . .	310
Displaying the Number of Ships . . . . .	313
<i>Exercise 14-4: All-Time High Score</i> . . . . .	317
<i>Exercise 14-5: Refactoring</i> . . . . .	317
<i>Exercise 14-6: Expanding Alien Invasion</i> . . . . .	317
Summary . . . . .	317

## PROJECT 2: DATA VISUALIZATION

<b>15</b>	
<b>GENERATING DATA</b>	<b>321</b>
Installing matplotlib . . . . .	322
On Linux . . . . .	322
On OS X . . . . .	322
On Windows . . . . .	323
Testing matplotlib . . . . .	323
The matplotlib Gallery . . . . .	323
Plotting a Simple Line Graph . . . . .	324
Changing the Label Type and Graph Thickness . . . . .	324
Correcting the Plot. . . . .	326
Plotting and Styling Individual Points with <code>scatter()</code> . . . . .	326
Plotting a Series of Points with <code>scatter()</code> . . . . .	328
Calculating Data Automatically. . . . .	328
Removing Outlines from Data Points . . . . .	329
Defining Custom Colors . . . . .	330
Using a Colormap. . . . .	330
Saving Your Plots Automatically . . . . .	331
<i>Exercise 15-1: Cubes</i> . . . . .	331
<i>Exercise 15-2: Colored Cubes</i> . . . . .	331
Random Walks . . . . .	331
Creating the <code>RandomWalk()</code> Class. . . . .	332
Choosing Directions . . . . .	332
Plotting the Random Walk . . . . .	333
Generating Multiple Random Walks . . . . .	334
Styling the Walk . . . . .	335
Coloring the Points . . . . .	335
Plotting the Starting and Ending Points. . . . .	336
Cleaning Up the Axes . . . . .	337
Adding Plot Points. . . . .	337
Altering the Size to Fill the Screen . . . . .	338
<i>Exercise 15-3: Molecular Motion</i> . . . . .	339
<i>Exercise 15-4: Modified Random Walks</i> . . . . .	339
<i>Exercise 15-5: Refactoring</i> . . . . .	339
Rolling Dice with Pygal . . . . .	339
Installing Pygal . . . . .	340
The Pygal Gallery . . . . .	340
Creating the Die Class. . . . .	340
Rolling the Die . . . . .	341
Analyzing the Results. . . . .	341

Making a Histogram . . . . .	342
Rolling Two Dice . . . . .	343
Rolling Dice of Different Sizes . . . . .	345
<i>Exercise 15-6: Automatic Labels</i> . . . . .	346
<i>Exercise 15-7: Two D8s</i> . . . . .	346
<i>Exercise 15-8: Three Dice</i> . . . . .	346
<i>Exercise 15-9: Multiplication</i> . . . . .	346
<i>Exercise 15-10: Practicing with Both Libraries</i> . . . . .	346
Summary . . . . .	347

## **16**

### **DOWNLOADING DATA** **349**

The CSV File Format . . . . .	350
Parsing the CSV File Headers . . . . .	350
Printing the Headers and Their Positions . . . . .	351
Extracting and Reading Data . . . . .	352
Plotting Data in a Temperature Chart . . . . .	353
The datetime Module . . . . .	354
Plotting Dates . . . . .	355
Plotting a Longer Timeframe . . . . .	356
Plotting a Second Data Series . . . . .	357
Shading an Area in the Chart . . . . .	358
Error-Checking . . . . .	359
<i>Exercise 16-1: San Francisco</i> . . . . .	362
<i>Exercise 16-2: Sitka-Death Valley Comparison</i> . . . . .	362
<i>Exercise 16-3: Rainfall</i> . . . . .	362
<i>Exercise 16-4: Explore</i> . . . . .	362
Mapping Global Data Sets: JSON Format . . . . .	362
Downloading World Population Data . . . . .	362
Extracting Relevant Data . . . . .	363
Converting Strings into Numerical Values . . . . .	364
Obtaining Two-Digit Country Codes . . . . .	365
Building a World Map . . . . .	367
Plotting Numerical Data on a World Map . . . . .	368
Plotting a Complete Population Map . . . . .	369
Grouping Countries by Population . . . . .	371
Styling World Maps in Pygal . . . . .	372
Lightening the Color Theme . . . . .	374
<i>Exercise 16-5: All Countries</i> . . . . .	375
<i>Exercise 16-6: Gross Domestic Product</i> . . . . .	375
<i>Exercise 16-7: Choose Your Own Data</i> . . . . .	375
<i>Exercise 16-8: Testing the country_codes Module</i> . . . . .	375
Summary . . . . .	375

## **17**

### **WORKING WITH APIS** **377**

Using a Web API . . . . .	378
Git and GitHub . . . . .	378
Requesting Data Using an API Call . . . . .	378
Installing Requests . . . . .	379

Processing an API Response . . . . .	379
Working with the Response Dictionary . . . . .	380
Summarizing the Top Repositories . . . . .	382
Monitoring API Rate Limits . . . . .	383
Visualizing Repositories Using Pygal . . . . .	384
Refining Pygal Charts . . . . .	386
Adding Custom Tooltips . . . . .	387
Plotting the Data . . . . .	388
Adding Clickable Links to Our Graph . . . . .	390
The Hacker News API . . . . .	390
<i>Exercise 17-1: Other Languages</i> . . . . .	393
<i>Exercise 17-2: Active Discussions</i> . . . . .	393
<i>Exercise 17-3: Testing python_repos.py</i> . . . . .	393
Summary . . . . .	393

## PROJECT 3: WEB APPLICATIONS

### 18

#### GETTING STARTED WITH DJANGO

**397**

Setting Up a Project . . . . .	398
Writing a Spec . . . . .	398
Creating a Virtual Environment . . . . .	398
Installing virtualenv . . . . .	399
Activating the Virtual Environment . . . . .	399
Installing Django . . . . .	400
Creating a Project in Django . . . . .	400
Creating the Database . . . . .	401
Viewing the Project . . . . .	401
<i>Exercise 18-1: New Projects</i> . . . . .	402
Starting an App . . . . .	403
Defining Models . . . . .	403
Activating Models . . . . .	404
The Django Admin Site . . . . .	406
Defining the Entry Model . . . . .	408
Migrating the Entry Model . . . . .	409
Registering Entry with the Admin Site . . . . .	409
The Django Shell . . . . .	410
<i>Exercise 18-2: Short Entries</i> . . . . .	412
<i>Exercise 18-3: The Django API</i> . . . . .	412
<i>Exercise 18-4: Pizzeria</i> . . . . .	412
Making Pages: The Learning Log Home Page . . . . .	412
Mapping a URL . . . . .	413
Writing a View . . . . .	414
Writing a Template . . . . .	415
<i>Exercise 18-5: Meal Planner</i> . . . . .	416
<i>Exercise 18-6: Pizzeria Home Page</i> . . . . .	416
Building Additional Pages . . . . .	416
Template Inheritance . . . . .	416
The Topics Page . . . . .	418
Individual Topic Pages . . . . .	421

<i>Exercise 18-7: Template Documentation</i> . . . . .	424
<i>Exercise 18-8: Pizzeria Pages</i> . . . . .	424
Summary . . . . .	425

## **19 USER ACCOUNTS 427**

Allowing Users to Enter Data . . . . .	428
Adding New Topics . . . . .	428
Adding New Entries . . . . .	432
Editing Entries . . . . .	435
<i>Exercise 19-1: Blog</i> . . . . .	438
Setting Up User Accounts . . . . .	439
The users App . . . . .	439
The Login Page . . . . .	440
Logging Out . . . . .	442
The Registration Page . . . . .	443
<i>Exercise 19-2: Blog Accounts</i> . . . . .	446
Allowing Users to Own Their Data . . . . .	446
Restricting Access with @login_required . . . . .	447
Connecting Data to Certain Users . . . . .	448
Restricting Topics Access to Appropriate Users . . . . .	451
Protecting a User's Topics . . . . .	451
Protecting the edit_entry Page . . . . .	452
Associating New Topics with the Current User . . . . .	453
<i>Exercise 19-3: Refactoring</i> . . . . .	454
<i>Exercise 19-4: Protecting new_entry</i> . . . . .	454
<i>Exercise 19-5: Protected Blog</i> . . . . .	454
Summary . . . . .	454

## **20 STYLING AND DEPLOYING AN APP 455**

Styling Learning Log . . . . .	456
The django-bootstrap3 App . . . . .	456
Using Bootstrap to Style Learning Log . . . . .	457
Modifying base.html . . . . .	458
Styling the Home Page Using a Jumbotron . . . . .	461
Styling the Login Page . . . . .	461
Styling the new_topic Page . . . . .	463
Styling the Topics Page . . . . .	463
Styling the Entries on the Topic Page . . . . .	464
<i>Exercise 20-1: Other Forms</i> . . . . .	466
<i>Exercise 20-2: Stylish Blog</i> . . . . .	466
Deploying Learning Log . . . . .	466
Making a Heroku Account . . . . .	466
Installing the Heroku Toolbelt . . . . .	466
Installing Required Packages . . . . .	466
Creating a Packages List with a requirements.txt File . . . . .	467
Specifying the Python Runtime . . . . .	468
Modifying settings.py for Heroku . . . . .	468
Making a Procfile to Start Processes . . . . .	469
Modifying wsgi.py for Heroku . . . . .	470

Making a Directory for Static Files . . . . .	470
Using the unicorn Server Locally . . . . .	470
Using Git to Track the Project's Files . . . . .	471
Pushing to Heroku . . . . .	473
Setting Up the Database on Heroku . . . . .	474
Refining the Heroku Deployment . . . . .	475
Securing the Live Project . . . . .	476
Committing and Pushing Changes . . . . .	477
Creating Custom Error Pages . . . . .	478
Ongoing Development . . . . .	480
The SECRET_KEY Setting . . . . .	481
Deleting a Project on Heroku . . . . .	481
<i>Exercise 20-3: Live Blog</i> . . . . .	482
<i>Exercise 20-4: More 404s</i> . . . . .	482
<i>Exercise 20-5: Extended Learning Log</i> . . . . .	482
Summary . . . . .	482

**AFTERWORD** **483**

**A**  
**INSTALLING PYTHON** **485**

Python on Linux . . . . .	485
Finding the Installed Version . . . . .	486
Installing Python 3 on Linux . . . . .	486
Python on OS X . . . . .	486
Finding the Installed Version . . . . .	486
Using Homebrew to Install Python 3 . . . . .	487
Python on Windows . . . . .	488
Installing Python 3 on Windows . . . . .	488
Finding the Python Interpreter . . . . .	488
Adding Python to Your Path Variable . . . . .	489
Python Keywords and Built-in Functions . . . . .	489
Python Keywords . . . . .	489
Python Built-in Functions . . . . .	490

**B**  
**TEXT EDITORS** **491**

Geany . . . . .	492
Installing Geany on Linux . . . . .	492
Installing Geany on Windows . . . . .	492
Running Python Programs in Geany . . . . .	493
Customizing Geany Settings . . . . .	493
Sublime Text . . . . .	494
Installing Sublime Text on OS X . . . . .	494
Installing Sublime Text on Linux . . . . .	494
Installing Sublime Text on Windows . . . . .	495
Running Python Programs in Sublime Text . . . . .	495
Configuring Sublime Text . . . . .	495
Customizing Sublime Text Settings . . . . .	496

IDLE . . . . .	496
Installing IDLE on Linux. . . . .	496
Installing IDLE on OS X . . . . .	496
Installing IDLE on Windows . . . . .	497
Customizing IDLE Settings . . . . .	497
Emacs and vim . . . . .	497

**C**  
**GETTING HELP** **499**

First Steps . . . . .	499
Try It Again . . . . .	500
Take a Break . . . . .	500
Refer to This Book’s Resources . . . . .	500
Searching Online . . . . .	501
Stack Overflow . . . . .	501
The Official Python Documentation . . . . .	501
Official Library Documentation . . . . .	502
r/learnpython . . . . .	502
Blog Posts . . . . .	502
IRC (Internet Relay Chat) . . . . .	502
Make an IRC Account . . . . .	502
Channels to Join . . . . .	503
IRC Culture. . . . .	503

**D**  
**USING GIT FOR VERSION CONTROL** **505**

Installing Git . . . . .	506
Installing Git on Linux . . . . .	506
Installing Git on OS X . . . . .	506
Installing Git on Windows . . . . .	506
Configuring Git. . . . .	506
Making a Project . . . . .	507
Ignoring Files. . . . .	507
Initializing a Repository. . . . .	507
Checking the Status . . . . .	508
Adding Files to the Repository . . . . .	508
Making a Commit . . . . .	509
Checking the Log . . . . .	509
The Second Commit . . . . .	510
Reverting a Change . . . . .	511
Checking Out Previous Commits . . . . .	512
Deleting the Repository . . . . .	513

**INDEX** **515**