

PRAISE FOR *PYTHON CRASH COURSE*

“It has been interesting to see No Starch Press producing future classics that should be alongside the more traditional programming books. *Python Crash Course* is one of those books.”

—GREG LADEN, SCIENCEBLOGS

“Deals with some rather complex projects and lays them out in a consistent, logical, and pleasant manner that draws the reader into the subject.”

—FULL CIRCLE MAGAZINE

“Well presented with good explanations of the code snippets. The book works with you, one small step at a time, building more complex code, explaining what’s going on all the way.”

—FLICKTHROUGH REVIEWS

“Learning Python with *Python Crash Course* was an extremely positive experience! A great choice if you’re new to Python.”

—MIKKE GOES CODING

“Does what it says on the tin, and does it really well. . . . Presents a large number of useful exercises as well as three challenging and entertaining projects.”

—REALPYTHON.COM

“A fast-paced but comprehensive introduction to programming with Python, *Python Crash Course* is another superb book to add to your library and help you finally master Python.”

—TUTORIALEDGE.NET

“A brilliant option for complete beginners without any coding experience. If you’re looking for a solid, uncomplicated intro to this very deep language, I have to recommend this book.”

—WHATPIXEL.COM

“Contains literally everything you need to know about Python and even more.”

—FIREBEARSTUDIO.COM

“While *Python Crash Course* uses Python to teach you to code, it also teaches clean programming skills that apply to most other languages.”

—GREAT LAKES GEEK

PYTHON CRASH COURSE

3RD EDITION

**A Hands-On, Project-Based
Introduction to Programming**

by **Eric Matthes**



**no starch
press**

San Francisco

PYTHON CRASH COURSE, 3RD EDITION. Copyright © 2023 by Eric Matthes.

All rights reserved. No part of this work may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage or retrieval system, without the prior written permission of the copyright owner and the publisher.

First printing

26 25 24 23 22 1 2 3 4 5

ISBN-13: 978-1-7185-0270-3 (print)

ISBN-13: 978-1-7185-0271-0 (ebook)

Publisher: William Pollock
Managing Editor: Jill Franklin
Production Editor: Jennifer Kepler
Developmental Editor: Eva Morrow
Cover Illustrator: Josh Ellingson
Interior Design: Octopod Studios
Technical Reviewer: Kenneth Love
Copyeditor: Doug McNair
Compositor: Jeff Lytle, Happenstance Type-O-Rama
Proofreader: Scout Festa

For information on distribution, bulk sales, corporate sales, or translations, please contact No Starch Press, Inc. directly at info@nostarch.com or:

No Starch Press, Inc.
245 8th Street, San Francisco, CA 94103
phone: 1.415.863.9900
www.nostarch.com

The Library of Congress has catalogued the first edition as follows:

Matthes, Eric, 1972-

Python crash course : a hands-on, project-based introduction to programming / by Eric Matthes.
pages cm

Includes index.

Summary: "A project-based introduction to programming in Python, with exercises. Covers general programming concepts, Python fundamentals, and problem solving. Includes three projects - how to create a simple video game, use data visualization techniques to make graphs and charts, and build an interactive web application"-- Provided by publisher.

ISBN 978-1-59327-603-4 -- ISBN 1-59327-603-6

1. Python (Computer program language) I. Title.

QA76.73.P98M38 2015

005.13'3--dc23

2015018135

No Starch Press and the No Starch Press logo are registered trademarks of No Starch Press, Inc. Other product and company names mentioned herein may be the trademarks of their respective owners. Rather than use a trademark symbol with every occurrence of a trademarked name, we are using the names only in an editorial fashion and to the benefit of the trademark owner, with no intention of infringement of the trademark.

The information in this book is distributed on an "As Is" basis, without warranty. While every precaution has been taken in the preparation of this work, neither the author nor No Starch Press, Inc. shall have any liability to any person or entity with respect to any loss or damage caused or alleged to be caused directly or indirectly by the information contained in it.

For my father, who always made time to
answer my questions about programming,
and for Ever, who is just beginning to ask
me his questions

About the Author

Eric Matthes was a high school math and science teacher for 25 years, and he taught introductory Python classes whenever he could find a way to fit them into the curriculum. Eric is a full-time writer and programmer now, and he is involved in a number of open source projects. His projects have a diverse range of goals, from helping predict landslide activity in mountainous regions to simplifying the process of deploying Django projects. When he's not writing or programming, he enjoys climbing mountains and spending time with his family.

About the Technical Reviewer

Kenneth Love lives in the Pacific Northwest with their family and cats. Kenneth is a longtime Python programmer, open source contributor, teacher, and conference speaker.

BRIEF CONTENTS

Preface to the Third Edition	xxvii
Acknowledgments	xxxix
Introduction	xxxiii
PART I: BASICS	1
Chapter 1: Getting Started	3
Chapter 2: Variables and Simple Data Types	15
Chapter 3: Introducing Lists	33
Chapter 4: Working with Lists	49
Chapter 5: if Statements	71
Chapter 6: Dictionaries	91
Chapter 7: User Input and while Loops	113
Chapter 8: Functions	129
Chapter 9: Classes	157
Chapter 10: Files and Exceptions	183
Chapter 11: Testing Your Code	209
PART II: PROJECTS	225
Chapter 12: A Ship That Fires Bullets	227
Chapter 13: Aliens!	255
Chapter 14: Scoring	277
Chapter 15: Generating Data	301
Chapter 16: Downloading Data	329
Chapter 17: Working with APIs	355

Chapter 18: Getting Started with Django	373
Chapter 19: User Accounts	403
Chapter 20: Styling and Deploying an App	433
Appendix A: Installation and Troubleshooting	463
Appendix B: Text Editors and IDEs	469
Appendix C: Getting Help	477
Appendix D: Using Git for Version Control	483
Appendix E: Troubleshooting Deployments	493
Index	503

CONTENTS IN DETAIL

PREFACE TO THE THIRD EDITION	xxvii
-------------------------------------	--------------

ACKNOWLEDGMENTS	xxxix
------------------------	--------------

INTRODUCTION	xxxiii
---------------------	---------------

Who Is This Book For?	xxxiv
What Can You Expect to Learn?	xxxiv
Online Resources	xxxv
Why Python?	xxxvi

PART I: BASICS	1
-----------------------	----------

1	
GETTING STARTED	3

Setting Up Your Programming Environment	3
Python Versions	4
Running Snippets of Python Code	4
About the VS Code Editor	4
Python on Different Operating Systems	5
Python on Windows	5
Python on macOS	7
Python on Linux	8
Running a Hello World Program	9
Installing the Python Extension for VS Code	9
Running <code>hello_world.py</code>	10
Troubleshooting	10
Running Python Programs from a Terminal	11
On Windows	12
On macOS and Linux	12
<i>Exercise 1-1: python.org</i>	13
<i>Exercise 1-2: Hello World Typos</i>	13
<i>Exercise 1-3: Infinite Skills</i>	13
Summary	13

2	
VARIABLES AND SIMPLE DATA TYPES	15

What Really Happens When You Run <code>hello_world.py</code>	15
Variables	16
Naming and Using Variables	17
Avoiding Name Errors When Using Variables	17
Variables Are Labels	18
<i>Exercise 2-1: Simple Message</i>	19
<i>Exercise 2-2: Simple Messages</i>	19

Strings	19
Changing Case in a String with Methods	20
Using Variables in Strings	20
Adding Whitespace to Strings with Tabs or Newlines	21
Stripping Whitespace	22
Removing Prefixes	23
Avoiding Syntax Errors with Strings	24
<i>Exercise 2-3: Personal Message</i>	25
<i>Exercise 2-4: Name Cases</i>	25
<i>Exercise 2-5: Famous Quote</i>	25
<i>Exercise 2-6: Famous Quote 2</i>	25
<i>Exercise 2-7: Stripping Names</i>	25
<i>Exercise 2-8: File Extensions</i>	25
Numbers	26
Integers	26
Floats	26
Integers and Floats	27
Underscores in Numbers	28
Multiple Assignment	28
Constants	28
<i>Exercise 2-9: Number Eight</i>	29
<i>Exercise 2-10: Favorite Number</i>	29
Comments	29
How Do You Write Comments?	29
What Kinds of Comments Should You Write?	29
<i>Exercise 2-11: Adding Comments</i>	30
The Zen of Python	30
<i>Exercise 2-12: Zen of Python</i>	31
Summary	32

3 INTRODUCING LISTS 33

What Is a List?	33
Accessing Elements in a List	34
Index Positions Start at 0, Not 1	34
Using Individual Values from a List	35
<i>Exercise 3-1: Names</i>	36
<i>Exercise 3-2: Greetings</i>	36
<i>Exercise 3-3: Your Own List</i>	36
Modifying, Adding, and Removing Elements	36
Modifying Elements in a List	36
Adding Elements to a List	37
Removing Elements from a List	38
<i>Exercise 3-4: Guest List</i>	41
<i>Exercise 3-5: Changing Guest List</i>	42
<i>Exercise 3-6: More Guests</i>	42
<i>Exercise 3-7: Shrinking Guest List</i>	42
Organizing a List	42
Sorting a List Permanently with the <code>sort()</code> Method	43
Sorting a List Temporarily with the <code>sorted()</code> Function	43
Printing a List in Reverse Order	44

Finding the Length of a List	44
<i>Exercise 3-8: Seeing the World</i>	45
<i>Exercise 3-9: Dinner Guests</i>	45
<i>Exercise 3-10: Every Function</i>	45
Avoiding Index Errors When Working with Lists	46
<i>Exercise 3-11: Intentional Error</i>	47
Summary	47

4 WORKING WITH LISTS 49

Looping Through an Entire List	49
A Closer Look at Looping	50
Doing More Work Within a for Loop	51
Doing Something After a for Loop	52
Avoiding Indentation Errors	53
Forgetting to Indent	53
Forgetting to Indent Additional Lines	54
Indenting Unnecessarily	54
Indenting Unnecessarily After the Loop	55
Forgetting the Colon	55
<i>Exercise 4-1: Pizzas</i>	56
<i>Exercise 4-2: Animals</i>	56
Making Numerical Lists	56
Using the range() Function	57
Using range() to Make a List of Numbers	58
Simple Statistics with a List of Numbers	59
List Comprehensions	59
<i>Exercise 4-3: Counting to Twenty</i>	60
<i>Exercise 4-4: One Million</i>	60
<i>Exercise 4-5: Summing a Million</i>	60
<i>Exercise 4-6: Odd Numbers</i>	60
<i>Exercise 4-7: Threes</i>	60
<i>Exercise 4-8: Cubes</i>	60
<i>Exercise 4-9: Cube Comprehension</i>	60
Working with Part of a List	61
Slicing a List	61
Looping Through a Slice	62
Copying a List	63
<i>Exercise 4-10: Slices</i>	65
<i>Exercise 4-11: My Pizzas, Your Pizzas</i>	65
<i>Exercise 4-12: More Loops</i>	65
Tuples	65
Defining a Tuple	65
Looping Through All Values in a Tuple	66
Writing Over a Tuple	67
<i>Exercise 4-13: Buffet</i>	67
Styling Your Code	68
The Style Guide	68
Indentation	68
Line Length	69
Blank Lines	69

Other Style Guidelines	69
<i>Exercise 4-14: PEP 8</i>	70
<i>Exercise 4-15: Code Review</i>	70
Summary	70

5

IF STATEMENTS **71**

A Simple Example	72
Conditional Tests	72
Checking for Equality	72
Ignoring Case When Checking for Equality	73
Checking for Inequality	74
Numerical Comparisons	74
Checking Multiple Conditions	75
Checking Whether a Value Is in a List	76
Checking Whether a Value Is Not in a List	76
Boolean Expressions	77
<i>Exercise 5-1: Conditional Tests</i>	77
<i>Exercise 5-2: More Conditional Tests</i>	78
if Statements	78
Simple if Statements	78
if-else Statements	79
The if-elif-else Chain	80
Using Multiple elif Blocks	81
Omitting the else Block	82
Testing Multiple Conditions	82
<i>Exercise 5-3: Alien Colors #1</i>	84
<i>Exercise 5-4: Alien Colors #2</i>	84
<i>Exercise 5-5: Alien Colors #3</i>	84
<i>Exercise 5-6: Stages of Life</i>	84
<i>Exercise 5-7: Favorite Fruit</i>	85
Using if Statements with Lists	85
Checking for Special Items	85
Checking That a List Is Not Empty	86
Using Multiple Lists	87
<i>Exercise 5-8: Hello Admin</i>	88
<i>Exercise 5-9: No Users</i>	88
<i>Exercise 5-10: Checking Usernames</i>	88
<i>Exercise 5-11: Ordinal Numbers</i>	88
Styling Your if Statements	89
<i>Exercise 5-12: Styling if Statements</i>	89
<i>Exercise 5-13: Your Ideas</i>	89
Summary	89

6

DICTIONARIES **91**

A Simple Dictionary	92
Working with Dictionaries	92
Accessing Values in a Dictionary	92
Adding New Key-Value Pairs	93
Starting with an Empty Dictionary	94

Modifying Values in a Dictionary	94
Removing Key-Value Pairs	96
A Dictionary of Similar Objects	96
Using get() to Access Values	97
<i>Exercise 6-1: Person</i>	98
<i>Exercise 6-2: Favorite Numbers</i>	98
<i>Exercise 6-3: Glossary</i>	99
Looping Through a Dictionary	99
Looping Through All Key-Value Pairs	99
Looping Through All the Keys in a Dictionary	101
Looping Through a Dictionary's Keys in a Particular Order	102
Looping Through All Values in a Dictionary	103
<i>Exercise 6-4: Glossary 2</i>	104
<i>Exercise 6-5: Rivers</i>	105
<i>Exercise 6-6: Polling</i>	105
Nesting	105
A List of Dictionaries	105
A List in a Dictionary	108
A Dictionary in a Dictionary	110
<i>Exercise 6-7: People</i>	111
<i>Exercise 6-8: Pets</i>	111
<i>Exercise 6-9: Favorite Places</i>	111
<i>Exercise 6-10: Favorite Numbers</i>	111
<i>Exercise 6-11: Cities</i>	111
<i>Exercise 6-12: Extensions</i>	111
Summary	111

7 USER INPUT AND WHILE LOOPS 113

How the input() Function Works	114
Writing Clear Prompts	114
Using int() to Accept Numerical Input	115
The Modulo Operator	116
<i>Exercise 7-1: Rental Car</i>	117
<i>Exercise 7-2: Restaurant Seating</i>	117
<i>Exercise 7-3: Multiples of Ten</i>	117
Introducing while Loops	117
The while Loop in Action	117
Letting the User Choose When to Quit	118
Using a Flag	120
Using break to Exit a Loop	121
Using continue in a Loop	122
Avoiding Infinite Loops	122
<i>Exercise 7-4: Pizza Toppings</i>	123
<i>Exercise 7-5: Movie Tickets</i>	123
<i>Exercise 7-6: Three Exits</i>	123
<i>Exercise 7-7: Infinity</i>	123
Using a while Loop with Lists and Dictionaries	124
Moving Items from One List to Another	124
Removing All Instances of Specific Values from a List	125
Filling a Dictionary with User Input	125
<i>Exercise 7-8: Deli</i>	127

<i>Exercise 7-9: No Pastrami</i>	127
<i>Exercise 7-10: Dream Vacation</i>	127
Summary	127

8

FUNCTIONS **129**

Defining a Function	130
Passing Information to a Function	130
Arguments and Parameters	131
<i>Exercise 8-1: Message</i>	131
<i>Exercise 8-2: Favorite Book</i>	131
Passing Arguments	131
Positional Arguments	132
Keyword Arguments	133
Default Values	134
Equivalent Function Calls	135
Avoiding Argument Errors	136
<i>Exercise 8-3: T-Shirt</i>	136
<i>Exercise 8-4: Large Shirts</i>	137
<i>Exercise 8-5: Cities</i>	137
Return Values	137
Returning a Simple Value	137
Making an Argument Optional	138
Returning a Dictionary	139
Using a Function with a while Loop	140
<i>Exercise 8-6: City Names</i>	141
<i>Exercise 8-7: Album</i>	142
<i>Exercise 8-8: User Albums</i>	142
Passing a List	142
Modifying a List in a Function	143
Preventing a Function from Modifying a List	145
<i>Exercise 8-9: Messages</i>	146
<i>Exercise 8-10: Sending Messages</i>	146
<i>Exercise 8-11: Archived Messages</i>	146
Passing an Arbitrary Number of Arguments	146
Mixing Positional and Arbitrary Arguments	147
Using Arbitrary Keyword Arguments	148
<i>Exercise 8-12: Sandwiches</i>	149
<i>Exercise 8-13: User Profile</i>	149
<i>Exercise 8-14: Cars</i>	149
Storing Your Functions in Modules	149
Importing an Entire Module	150
Importing Specific Functions	151
Using as to Give a Function an Alias	151
Using as to Give a Module an Alias	152
Importing All Functions in a Module	152
Styling Functions	153
<i>Exercise 8-15: Printing Models</i>	154
<i>Exercise 8-16: Imports</i>	154
<i>Exercise 8-17: Styling Functions</i>	154
Summary	154

9

CLASSES

157

Creating and Using a Class	158
Creating the Dog Class	158
The <code>__init__()</code> Method	159
Making an Instance from a Class	159
<i>Exercise 9-1: Restaurant</i>	162
<i>Exercise 9-2: Three Restaurants</i>	162
<i>Exercise 9-3: Users</i>	162
Working with Classes and Instances	162
The Car Class	162
Setting a Default Value for an Attribute	163
Modifying Attribute Values	164
<i>Exercise 9-4: Number Served</i>	166
<i>Exercise 9-5: Login Attempts</i>	167
Inheritance	167
The <code>__init__()</code> Method for a Child Class	167
Defining Attributes and Methods for the Child Class	169
Overriding Methods from the Parent Class	170
Instances as Attributes	170
Modeling Real-World Objects	172
<i>Exercise 9-6: Ice Cream Stand</i>	173
<i>Exercise 9-7: Admin</i>	173
<i>Exercise 9-8: Privileges</i>	173
<i>Exercise 9-9: Battery Upgrade</i>	173
Importing Classes	173
Importing a Single Class	174
Storing Multiple Classes in a Module	175
Importing Multiple Classes from a Module	176
Importing an Entire Module	176
Importing All Classes from a Module	177
Importing a Module into a Module	177
Using Aliases	178
Finding Your Own Workflow	179
<i>Exercise 9-10: Imported Restaurant</i>	179
<i>Exercise 9-11: Imported Admin</i>	179
<i>Exercise 9-12: Multiple Modules</i>	179
The Python Standard Library	179
<i>Exercise 9-13: Dice</i>	180
<i>Exercise 9-14: Lottery</i>	180
<i>Exercise 9-15: Lottery Analysis</i>	180
<i>Exercise 9-16: Python Module of the Week</i>	180
Styling Classes	181
Summary	181

10

FILES AND EXCEPTIONS

183

Reading from a File	184
Reading the Contents of a File	184
Relative and Absolute File Paths	186
Accessing a File's Lines	186

Working with a File's Contents	187
Large Files: One Million Digits	188
Is Your Birthday Contained in Pi?	189
<i>Exercise 10-1: Learning Python</i>	189
<i>Exercise 10-2: Learning C</i>	190
<i>Exercise 10-3: Simpler Code</i>	190
Writing to a File	190
Writing a Single Line	190
Writing Multiple Lines	191
<i>Exercise 10-4: Guest</i>	192
<i>Exercise 10-5: Guest Book</i>	192
Exceptions	192
Handling the ZeroDivisionError Exception	192
Using try-except Blocks	193
Using Exceptions to Prevent Crashes	193
The else Block	194
Handling the FileNotFoundError Exception	195
Analyzing Text	196
Working with Multiple Files	197
Failing Silently	198
Deciding Which Errors to Report	199
<i>Exercise 10-6: Addition</i>	200
<i>Exercise 10-7: Addition Calculator</i>	200
<i>Exercise 10-8: Cats and Dogs</i>	200
<i>Exercise 10-9: Silent Cats and Dogs</i>	200
<i>Exercise 10-10: Common Words</i>	200
Storing Data	201
Using json.dumps() and json.loads()	201
Saving and Reading User-Generated Data	202
Refactoring	204
<i>Exercise 10-11: Favorite Number</i>	206
<i>Exercise 10-12: Favorite Number Remembered</i>	206
<i>Exercise 10-13: User Dictionary</i>	206
<i>Exercise 10-14: Verify User</i>	206
Summary	207

11

TESTING YOUR CODE 209

Installing pytest with pip	210
Updating pip	210
Installing pytest	211
Testing a Function	211
Unit Tests and Test Cases	212
A Passing Test	212
Running a Test	213
A Failing Test	214
Responding to a Failed Test	215
Adding New Tests	216
<i>Exercise 11-1: City, Country</i>	217
<i>Exercise 11-2: Population</i>	217

Testing a Class	217
A Variety of Assertions	217
A Class to Test	218
Testing the AnonymousSurvey Class	220
Using Fixtures	221
<i>Exercise 11-3: Employee</i>	223
Summary	223

PART II: PROJECTS

225

12

A SHIP THAT FIRES BULLETS

227

Planning Your Project	228
Installing Pygame	228
Starting the Game Project	229
Creating a Pygame Window and Responding to User Input	229
Controlling the Frame Rate	230
Setting the Background Color	231
Creating a Settings Class	232
Adding the Ship Image	233
Creating the Ship Class	234
Drawing the Ship to the Screen	235
Refactoring: The <code>_check_events()</code> and <code>_update_screen()</code> Methods	237
The <code>_check_events()</code> Method	237
The <code>_update_screen()</code> Method	237
<i>Exercise 12-1: Blue Sky</i>	238
<i>Exercise 12-2: Game Character</i>	238
Piloting the Ship	238
Responding to a Keypress	238
Allowing Continuous Movement	239
Moving Both Left and Right	241
Adjusting the Ship's Speed	242
Limiting the Ship's Range	243
Refactoring <code>_check_events()</code>	244
Pressing Q to Quit	244
Running the Game in Fullscreen Mode	245
A Quick Recap	245
alien_invasion.py	246
settings.py	246
ship.py	246
<i>Exercise 12-3: Pygame Documentation</i>	246
<i>Exercise 12-4: Rocket</i>	246
<i>Exercise 12-5: Keys</i>	246
Shooting Bullets	247
Adding the Bullet Settings	247
Creating the Bullet Class	247
Storing Bullets in a Group	248
Firing Bullets	249
Deleting Old Bullets	250

Limiting the Number of Bullets	251
Creating the <code>_update_bullets()</code> Method	252
<i>Exercise 12-6: Sideways Shooter</i>	253
Summary	253

13

ALIENS! 255

Reviewing the Project	256
Creating the First Alien	256
Creating the Alien Class	257
Creating an Instance of the Alien	257
Building the Alien Fleet	259
Creating a Row of Aliens	259
Refactoring <code>_create_fleet()</code>	260
Adding Rows	261
<i>Exercise 13-1: Stars</i>	263
<i>Exercise 13-2: Better Stars</i>	263
Making the Fleet Move	263
Moving the Aliens Right	263
Creating Settings for Fleet Direction	264
Checking Whether an Alien Has Hit the Edge	265
Dropping the Fleet and Changing Direction	265
<i>Exercise 13-3: Raindrops</i>	266
<i>Exercise 13-4: Steady Rain</i>	266
Shooting Aliens	266
Detecting Bullet Collisions	267
Making Larger Bullets for Testing	268
Repopulating the Fleet	268
Speeding Up the Bullets	269
Refactoring <code>_update_bullets()</code>	269
<i>Exercise 13-5: Sideways Shooter Part 2</i>	270
Ending the Game	270
Detecting Alien-Ship Collisions	270
Responding to Alien-Ship Collisions	271
Aliens That Reach the Bottom of the Screen	273
Game Over!	274
Identifying When Parts of the Game Should Run	275
<i>Exercise 13-6: Game Over</i>	275
Summary	275

14

SCORING 277

Adding the Play Button	278
Creating a Button Class	278
Drawing the Button to the Screen	279
Starting the Game	281
Resetting the Game	281
Deactivating the Play Button	282
Hiding the Mouse Cursor	282
<i>Exercise 14-1: Press P to Play</i>	283
<i>Exercise 14-2: Target Practice</i>	283

Leveling Up	283
Modifying the Speed Settings	283
Resetting the Speed	285
<i>Exercise 14-3: Challenging Target Practice</i>	286
<i>Exercise 14-4: Difficulty Levels</i>	286
Scoring	286
Displaying the Score	286
Making a Scoreboard	287
Updating the Score as Aliens Are Shot Down	289
Resetting the Score	289
Making Sure to Score All Hits	290
Increasing Point Values	290
Rounding the Score	291
High Scores	292
Displaying the Level.	294
Displaying the Number of Ships	296
<i>Exercise 14-5: All-Time High Score</i>	299
<i>Exercise 14-6: Refactoring</i>	299
<i>Exercise 14-7: Expanding the Game</i>	299
<i>Exercise 14-8: Sideways Shooter, Final Version</i>	299
Summary	299

15

GENERATING DATA	301
Installing Matplotlib	302
Plotting a Simple Line Graph	302
Changing the Label Type and Line Thickness	303
Correcting the Plot.	305
Using Built-in Styles	306
Plotting and Styling Individual Points with scatter()	306
Plotting a Series of Points with scatter().	308
Calculating Data Automatically.	308
Customizing Tick Labels	309
Defining Custom Colors	310
Using a Colormap.	310
Saving Your Plots Automatically	311
<i>Exercise 15-1: Cubes</i>	311
<i>Exercise 15-2: Colored Cubes</i>	311
Random Walks	312
Creating the RandomWalk Class.	312
Choosing Directions	312
Plotting the Random Walk	313
Generating Multiple Random Walks	314
Styling the Walk	315
<i>Exercise 15-3: Molecular Motion</i>	319
<i>Exercise 15-4: Modified Random Walks</i>	319
<i>Exercise 15-5: Refactoring</i>	319
Rolling Dice with Plotly	319
Installing Plotly	320
Creating the Die Class.	320
Rolling the Die	320
Analyzing the Results.	321

Making a Histogram	322
Customizing the Plot	323
Rolling Two Dice	324
Further Customizations.	325
Rolling Dice of Different Sizes.	326
Saving Figures	327
<i>Exercise 15-6: Two D8s</i>	328
<i>Exercise 15-7: Three Dice</i>	328
<i>Exercise 15-8: Multiplication</i>	328
<i>Exercise 15-9: Die Comprehensions</i>	328
<i>Exercise 15-10: Practicing with Both Libraries</i>	328
Summary	328

16 DOWNLOADING DATA 329

The CSV File Format	330
Parsing the CSV File Headers	330
Printing the Headers and Their Positions.	331
Extracting and Reading Data	332
Plotting Data in a Temperature Chart	332
The datetime Module	333
Plotting Dates	334
Plotting a Longer Timeframe	336
Plotting a Second Data Series.	336
Shading an Area in the Chart.	337
Error Checking	338
Downloading Your Own Data.	341
<i>Exercise 16-1: Sitka Rainfall</i>	342
<i>Exercise 16-2: Sitka–Death Valley Comparison</i>	342
<i>Exercise 16-3: San Francisco</i>	342
<i>Exercise 16-4: Automatic Indexes</i>	342
<i>Exercise 16-5: Explore</i>	342
Mapping Global Datasets: GeoJSON Format	342
Downloading Earthquake Data	343
Examining GeoJSON Data.	343
Making a List of All Earthquakes.	345
Extracting Magnitudes	346
Extracting Location Data	346
Building a World Map	347
Representing Magnitudes	348
Customizing Marker Colors	349
Other Color Scales	350
Adding Hover Text	350
<i>Exercise 16-6: Refactoring</i>	352
<i>Exercise 16-7: Automated Title</i>	352
<i>Exercise 16-8: Recent Earthquakes</i>	352
<i>Exercise 16-9: World Fires</i>	352
Summary	352

17		
WORKING WITH APIS		355
Using an API		355
Git and GitHub		356
Requesting Data Using an API Call		356
Installing Requests		357
Processing an API Response		357
Working with the Response Dictionary		358
Summarizing the Top Repositories		361
Monitoring API Rate Limits		362
Visualizing Repositories Using Plotly		362
Styling the Chart		364
Adding Custom Tooltips		365
Adding Clickable Links		366
Customizing Marker Colors		367
More About Plotly and the GitHub API		368
The Hacker News API		368
<i>Exercise 17-1: Other Languages</i>		371
<i>Exercise 17-2: Active Discussions</i>		371
<i>Exercise 17-3: Testing python_repos.py</i>		372
<i>Exercise 17-4: Further Exploration</i>		372
Summary		372

18		
GETTING STARTED WITH DJANGO		373
Setting Up a Project		374
Writing a Spec		374
Creating a Virtual Environment		374
Activating the Virtual Environment		375
Installing Django		375
Creating a Project in Django		376
Creating the Database		376
Viewing the Project		377
<i>Exercise 18-1: New Projects</i>		378
Starting an App		379
Defining Models		379
Activating Models		380
The Django Admin Site		381
Defining the Entry Model		384
Migrating the Entry Model		385
Registering Entry with the Admin Site		385
The Django Shell		386
<i>Exercise 18-2: Short Entries</i>		387
<i>Exercise 18-3: The Django API</i>		388
<i>Exercise 18-4: Pizzeria</i>		388
Making Pages: The Learning Log Home Page		388
Mapping a URL		388
Writing a View		390
Writing a Template		390

<i>Exercise 18-5: Meal Planner</i>	392
<i>Exercise 18-6: Pizzeria Home Page</i>	392
Building Additional Pages	392
Template Inheritance	392
The Topics Page	394
Individual Topic Pages	397
<i>Exercise 18-7: Template Documentation</i>	400
<i>Exercise 18-8: Pizzeria Pages</i>	400
Summary	400

19

USER ACCOUNTS 403

Allowing Users to Enter Data	404
Adding New Topics	404
Adding New Entries	408
Editing Entries	412
<i>Exercise 19-1: Blog</i>	415
Setting Up User Accounts	415
The accounts App	415
The Login Page	416
Logging Out	419
The Registration Page	420
<i>Exercise 19-2: Blog Accounts</i>	423
Allowing Users to Own Their Data	423
Restricting Access with @login_required	423
Connecting Data to Certain Users	425
Restricting Topics Access to Appropriate Users	427
Protecting a User's Topics	428
Protecting the edit_entry Page	429
Associating New Topics with the Current User	429
<i>Exercise 19-3: Refactoring</i>	430
<i>Exercise 19-4: Protecting new_entry</i>	430
<i>Exercise 19-5: Protected Blog</i>	430
Summary	430

20

STYLING AND DEPLOYING AN APP 433

Styling Learning Log	434
The django-bootstrap5 App	434
Using Bootstrap to Style Learning Log	434
Modifying base.html	435
Styling the Home Page Using a Jumbotron	440
Styling the Login Page	441
Styling the Topics Page	442
Styling the Entries on the Topic Page	443
<i>Exercise 20-1: Other Forms</i>	445
<i>Exercise 20-2: Stylish Blog</i>	445
Deploying Learning Log	445
Making a Platform.sh Account	445
Installing the Platform.sh CLI	446
Installing platformshconfig	446

Creating a requirements.txt File	446
Additional Deployment Requirements	447
Adding Configuration Files	447
Modifying settings.py for Platform.sh	451
Using Git to Track the Project's Files	451
Creating a Project on Platform.sh	453
Pushing to Platform.sh	455
Viewing the Live Project	456
Refining the Platform.sh Deployment	456
Creating Custom Error Pages	459
Ongoing Development	460
Deleting a Project on Platform.sh	461
<i>Exercise 20-3: Live Blog</i>	<i>461</i>
<i>Exercise 20-4: Extended Learning Log</i>	<i>462</i>
Summary	462

A
INSTALLATION AND TROUBLESHOOTING **463**

Python on Windows	463
Using py Instead of python	463
Rerunning the Installer	464
Python on macOS	464
Accidentally Installing Apple's Version of Python	464
Python 2 on Older Versions of macOS	465
Python on Linux	465
Using the Default Python Installation	465
Installing the Latest Version of Python	465
Checking Which Version of Python You're Using	466
Python Keywords and Built-in Functions	466
Python Keywords	466
Python Built-in Functions	467

B
TEXT EDITORS AND IDES **469**

Working Efficiently with VS Code	470
Configuring VS Code	470
VS Code Shortcuts	473
Other Text Editors and IDEs	474
IDLE 474	
Geany	474
Sublime Text	474
Emacs and Vim	475
PyCharm	475
Jupyter Notebooks	475

C
GETTING HELP **477**

First Steps	477
Try It Again	478
Take a Break	478
Refer to This Book's Resources	478

Searching Online	479
Stack Overflow	479
The Official Python Documentation	479
Official Library Documentation	480
r/learnpython	480
Blog Posts	480
Discord	480
Slack	481

D
USING GIT FOR VERSION CONTROL **483**

Installing Git	484
Configuring Git	484
Making a Project	484
Ignoring Files	484
Initializing a Repository	485
Checking the Status	485
Adding Files to the Repository	486
Making a Commit	486
Checking the Log	487
The Second Commit	487
Abandoning Changes	488
Checking Out Previous Commits	489
Deleting the Repository	491

E
TROUBLESHOOTING DEPLOYMENTS **493**

Understanding Deployments	494
Basic Troubleshooting	494
Follow Onscreen Suggestions	495
Read the Log Output	496
OS-Specific Troubleshooting	497
Deploying from Windows	497
Deploying from macOS	499
Deploying from Linux	499
Other Deployment Approaches	500

INDEX **503**