

CONTENTS IN DETAIL

INTRODUCTION	xix
Who This Book Is For	xx
How Spreadsheets Can Hold You Back	xxi
Detecting Errors	xxii
Documenting Your Work	xxiii
Collaborating with Others	xxiv
Integrating Modern AI Tools	xxv
The Benefits of Using Python vs. Excel	xxv
Speed	xxvi
Modularity	xxviii
Automation	xxviii
Interoperability	xxx
Security	xxxi
Alternatives to Python	xxxii
VBA	xxxii
Python in Excel	xxxiii
What's in This Book	xxxiv

PART I GETTING STARTED WITH PYTHON

1	
SETTING UP YOUR CODING ENVIRONMENT	3
Why a Well-Configured Installation Matters	3
Using the Command Line	4
Navigating Directories and Subdirectories	5
Manipulating Files	9
Customizing and Storing System Settings	10
Working Within Corporate IT Policies	13
Installing Python	14
Running the Installer	14
Verifying the Installation	16
Running Some Code	17
Getting Started with Python Libraries	18

Tools to Simplify Writing Code	19
Important Features	20
Popular Python Editors	21
VS Code Installation	21
Conclusion	23

2	
CODING FUNDAMENTALS EXPLAINED THROUGH EXCEL	25
What Is Code?	26
Objects and Variables	27
Rules of Python Syntax	28
Case Sensitivity	28
Indentation	28
Comments	29
Statements and Assignments	29
Testing Examples in VS Code	30
Basic Data Types	31
Boolean Variables	31
Integers and Floats	31
Strings and Characters	32
None	34
Operators	34
Arithmetic Operators	34
Comparison Operators	35
Logical Operators	35
Composite Data Types	36
Lists	36
Tuples	37
Dictionaries	38
Functions and Methods	39
Common String Methods	40
Common List Methods	40
Common Dict Methods	41
Conditionals	41
if Statements	42
else Clauses	43
elif Clauses	43
Nested Conditionals	44
Representing Dates and Times	45
Internal Representation of Dates in Excel	45
The datetime Library	46
Conclusion	50

3	AUTOMATING TASKS WITH PYTHON SCRIPTS	51
What Are Scripts?		51
When to Use a Script		52
Parts of a Script		53
Loops		54
for Loops		54
List Comprehensions		56
Loop Controls		57
while Loops		59
User-Defined Functions		59
Arguments		60
Scope		62
Returns		62
Modifier Functions		64
Functionalizing Logic		65
Building and Running a Complete Script		66
Running a Script on the Command Line		69
Importing a Script as a Module		70
Using a Main Block		71
Passing System Arguments		72
Conclusion		73
 4	 TRACKING CHANGES WITH VERSION CONTROL	 75
Why Use Version Control?		76
What Is Git?		77
Why Git Works Best with Code		77
Git vs. GitHub		78
Working Within Git Repositories		78
Centralizing Your Work in a Single Folder		78
Navigating Local and Remote Repositories		80
Maintaining a .gitignore File		81
Setting Up a Git Repository		82
Installing Git		82
Configuring Git		82
Creating a New Git Repository		83
Tracking Changes to Files		84
Git's Staging Area		84
Commits		85
Syncing Local and Remote Repositories		86
fetch		86
pull		86
push		86

Managing Histories	87
The Main Branch	87
Feature Branches	87
Merges	88
Conflicts	88
Best Practices for Using Git	90
Troubleshooting with Status Messages	91
Integrating Git with VS Code	92
How Git Works	92
Directed Acyclic Graphs	92
Hashing	93
Conclusion	93

PART II WORKING WITH DATA

5 DATA ANALYSIS MADE INTERACTIVE 97

What Are Jupyter Notebooks?	98
Code Cells	98
Markdown Cells	99
Raw Text Cells	100
The JupyterLab Code Editor	101
Installing JupyterLab	101
Launching JupyterLab	101
Creating a Jupyter Notebook	102
Closing a Jupyter Notebook	103
Building a Jupyter Notebook	103
Scripts and Notebooks Working Together	105
Version Control with Jupyter Notebooks	107
Conclusion	107

6 ANALYZING AND TRANSFORMING DATA 109

Efficient Data Analysis with pandas	110
Series and DataFrames	110
pandas and Jupyter Notebooks	111
Getting Started with pandas	111
Creating a DataFrame from Scratch	112
Reading Data from a File into a DataFrame	114
Working with pandas Data Types	115
Moving Between Excel and pandas	116
Using the Clipboard	117
Using openpyxl	118

Manipulating and Transforming Data	118
Accessing Values	118
Updating Values	120
Sorting Datasets	121
Manipulating Data	121
Handling Missing Data	124
Combining DataFrames	125
Aggregating Data	128
Easily Reshaping Data with pandas	129
Grouping	129
Pivoting	130
Conclusion	131

7

WORKING WITH DATABASES

133

Why Use a Database?	134
How Databases Work	135
Incorporating Databases into Your Workflow	135
Where Databases Live	135
How to Access Corporate Databases	136
The Costs of a Database Connection	136
Relational Databases and SQL Workflows	137
Database Management Systems	137
Sandbox Environments	138
Creating a Practice Database	138
Installing PostgreSQL	138
Managing Users and Databases	140
Using PostgreSQL in VS Code	141
Creating and Editing Tables	142
Creating a Table	143
Altering a Table's Structure	144
Dropping a Table	145
Adding Rows	145
Updating Records	145
Deleting Data	146
Retrieving Database Data	146
Selections	146
Filters	148
Joins	150
Column Transformations	154
Aggregations	154
Groupings	155
Structuring Complex Queries	156
Automating Database Tasks with Python and SQL	157
Installing Psycopg 2	158
Connecting to PostgreSQL from Python	158

Running a Query from Python	159
Closing the Database Connection.....	160
Conclusion	160

8 RETRIEVING DATA FROM THE INTERNET 161

What Is an API?	162
REST APIs	162
Other Types of APIs	163
Making Requests and Handling Responses	163
Internet Communication Protocols	163
Request Types	165
Response Types	166
Simplifying Requests with the Requests Library	169
Making an API Request	169
Adding Arguments	170
Building in Error Handling	171
try...except Blocks	172
Common Errors	173
Breaking Large Requests into Smaller Parts	174
Handling Authentication.....	176
API Keys	177
OAuth	179
Streamlining Your Requests with Reusable Functions	180
Conclusion	183

9 CREATING CHARTS AND VISUALS 185

Charting with Excel vs. Python	186
Integration into a Broader Workflow	186
Reusability	186
Version Control	187
Customization	187
Standardization.....	187
Charting Libraries in Python	188
Using Plotly in Jupyter Notebooks	189
Installing and Importing Plotly	189
Creating Simple Charts	189
An Example Dataset.....	190
A Basic Line Chart	191
A Multi-Trace Line Chart	193
A Basic Bar Chart.....	194
A Multi-Trace Bar Chart.....	196
Saving Charts as Images	198
Charting Multidimensional Data.....	198

Elevating Charts with Custom Styling	200
Layout Properties	201
Themes	205
Using Plotly's Interactivity Features	209
Conclusion	210

10 BUILDING INTERACTIVE REPORTS 211

Why Use Interactive Reports	211
Creating Interactive Reports with Dash	212
When Interactive Reports Make Sense	212
How to Choose the Right Framework	213
Dash Basics	214
Installing Dash	215
Setting Up a Dashboard	215
Deploying a Dashboard Locally	216
Web Development Concepts and Dash	216
HTML Basics	217
HTML in Dash	219
CSS Basics	221
CSS in Dash	222
Integrating Plotly and Dash	223
Functionalizing Plotly Charts	223
Embedding a Chart in a Dashboard	225
Enabling User Selections	226
Updating Charts with Callbacks	226
Putting It All Together	228
Sharing Your Reports with Others	230
Conclusion	231

PART III WRITING BETTER CODE

11 ORGANIZING YOUR CODE WITH CLASSES 235

From Rows and Columns to Classes	236
Examples of Classes in Common Python Libraries	237
Custom Classes	237
Programming with Objects to Stay Organized	239
Reusability	243
Extendability	244
Encapsulation	245
Inheritance	249
Conclusion	255

12	
FINDING AND FIXING ERRORS	257
The Art of Debugging	257
Debugging Inside the VS Code Editor	258
Breakpoints	259
The Debug Panel	260
The Debug Console	262
Debugging Code with Errors	263
The Step Out Command	263
The Step Into Command	264
Pausing Code at the Right Moment	265
Debugging with Different Inputs	266
The Process of Testing Code	267
Assertions	268
The unittest Module	269
Identifying Test Cases	271
Conclusion	272
 13	
THREE GOOD CODING HABITS	273
Simpler Code Is Better Code	274
Too Many Conditions	274
Mixed Scopes	276
Inconsistency	277
Confusing Variable Names	277
Repetition	278
Making Code Manageable	279
The Right Amount of Modularization	280
Too Much Modularization	281
First Make It Run, Then Make It Better	282
Conclusion	283
 AFTERWORD	285
 INDEX	287