

BRIEF CONTENTS

Acknowledgments	xv
Introduction	1
Chapter 1: Starting Your Project.	5
Chapter 2: Modules, Libraries, and Frameworks	15
Chapter 3: Documentation and Good API Practice	33
Chapter 4: Handling Timestamps and Time Zones	49
Chapter 5: Distributing Your Software	57
Chapter 6: Unit Testing.	75
Chapter 7: Methods and Decorators	99
Chapter 8: Functional Programming.	119
Chapter 9: The Abstract Syntax Tree, Hy, and Lisp-like Attributes	135
Chapter 10: Performances and Optimizations	151
Chapter 11: Scaling and Architecture.	177
Chapter 12: Managing Relational Databases	187
Chapter 13: Write Less, Code More.	201
Index	215

CONTENTS IN DETAIL

ACKNOWLEDGMENTS

xv

INTRODUCTION

1

Who Should Read This Book and Why	2
About This Book	2

1

STARTING YOUR PROJECT

5

Versions of Python	5
Laying Out Your Project	7
What to Do	7
What Not to Do	8
Version Numbering	8
Coding Style and Automated Checks	10
Tools to Catch Style Errors	11
Tools to Catch Coding Errors	12
Joshua Harlow on Python	13

2

MODULES, LIBRARIES, AND FRAMEWORKS

15

The Import System	16
The sys Module	17
Import Paths	18
Custom Importers	18
Meta Path Finders	19
Useful Standard Libraries	20
External Libraries	22
The External Libraries Safety Checklist	23
Protecting Your Code with an API Wrapper	23
Package Installation: Getting More from pip	24
Using and Choosing Frameworks	26
Doug Hellmann, Python Core Developer, on Python Libraries	27

3

DOCUMENTATION AND GOOD API PRACTICE

33

Documenting with Sphinx	34
Getting Started with Sphinx and reST	35
Sphinx Modules	36
Writing a Sphinx Extension	39
Managing Changes to Your APIs	40
Numbering API Versions	41

Documenting Your API Changes	41
Marking Deprecated Functions with the warnings Module	43
Summary	45
Christophe de Vienne on Developing APIs.	45

4 HANDLING TIMESTAMPS AND TIME ZONES 49

The Problem of Missing Time Zones	50
Building Default datetime Objects	50
Time Zone–Aware Timestamps with dateutil.	52
Serializing Time Zone–Aware datetime Objects.	54
Solving Ambiguous Times	55
Summary	56

5 DISTRIBUTING YOUR SOFTWARE 57

A Bit of setup.py History	58
Packaging with setup.cfg	60
The Wheel Format Distribution Standard	61
Sharing Your Work with the World	64
Entry Points	67
Visualizing Entry Points	68
Using Console Scripts	69
Using Plugins and Drivers	71
Summary	73
Nick Coghlan on Packaging	74

6 UNIT TESTING 75

The Basics of Testing	76
Some Simple Tests.	76
Skipping Tests	78
Running Particular Tests	79
Running Tests in Parallel.	81
Creating Objects Used in Tests with Fixtures.	81
Running Test Scenarios	83
Controlled Tests Using Mocking	84
Revealing Untested Code with coverage	88
Virtual Environments	90
Setting Up a Virtual Environment.	91
Using virtualenv with tox	92
Re-creating an Environment	94
Using Different Python Versions	95
Integrating Other Tests.	95
Testing Policy.	96
Robert Collins on Testing.	97

7		
METHODS AND DECORATORS		99
Decorators and When to Use Them		100
Creating Decorators		100
Writing Decorators		101
Stacking Decorators		102
Writing Class Decorators		103
How Methods Work in Python		107
Static Methods		108
Class Methods		109
Abstract Methods		110
Mixing Static, Class, and Abstract Methods		112
Putting Implementations in Abstract Methods		114
The Truth About super		114
Summary		117

8		
FUNCTIONAL PROGRAMMING		119
Creating Pure Functions		120
Generators		121
Creating a Generator		121
Returning and Passing Values with yield		123
Inspecting Generators		124
List Comprehensions		125
Functional Functions Functioning		126
Applying Functions to Items with map()		127
Filtering Lists with filter()		127
Getting Indexes with enumerate()		127
Sorting a List with sorted()		128
Finding Items That Satisfy Conditions with any() and all()		128
Combining Lists with zip()		129
A Common Problem Solved		129
Useful itertools Functions		132
Summary		134

9		
THE ABSTRACT SYNTAX TREE, HY, AND LISP-LIKE ATTRIBUTES		135
Looking at the AST		136
Writing a Program Using the AST		137
The AST Objects		138
Walking Through an AST		139
Extending flake8 with AST Checks		140
Writing the Class		141
Ignoring Irrelevant Code		141
Checking for the Correct Decorator		142
Looking for self		143
A Quick Introduction to Hy		145
Summary		147
Paul Tagliamonte on the AST and Hy		147

10		
PERFORMANCES AND OPTIMIZATIONS		151
Data Structures		152
Understanding Behavior Through Profiling		154
cProfile		154
Disassembling with the dis Module		156
Defining Functions Efficiently		158
Ordered Lists and bisect		159
namedtuple and Slots		162
Memoization		167
Faster Python with PyPy		169
Achieving Zero Copy with the Buffer Protocol		170
Summary		174
Victor Stinner on Optimization		174
11		
SCALING AND ARCHITECTURE		177
Multithreading in Python and Its Limitations		178
Multiprocessing vs. Multithreading		179
Event-Driven Architecture		181
Other Options and asyncio		182
Service-Oriented Architecture		184
Interprocess Communication with ZeroMQ		185
Summary		186
12		
MANAGING RELATIONAL DATABASES		187
RDBMSs, ORMs, and When to Use Them		187
Database Backends		190
Streaming Data with Flask and PostgreSQL		190
Writing the Data-Streaming Application		191
Building the Application		193
Dimitri Fontaine on Databases		195
13		
WRITE LESS, CODE MORE		201
Using six for Python 2 and 3 Support		201
Strings and Unicode		202
Handling Python Modules Moves		203
The modernize Module		203
Using Python Like Lisp to Make a Single Dispatcher		203
Creating Generic Methods in Lisp		204
Generic Methods with Python		205
Context Managers		207
Less Boilerplate with attr		210
Summary		213
INDEX		215