Table of Contents

Preface.. xi

Part I. Getting Started

1. Picking an Interpreter.. 3
The State of Python 2 Versus Python 3 3
Recommendations 4
So…3? 4
Implementations 5
CPython 5
Stackless 5
PyPy 6
Jython 6
IronPython 6
PythonNet 7
Skulpt 7
MicroPython 7

2. Properly Installing Python.. 9
Installing Python on Mac OS X 9
Setuptools and pip 11
virtualenv 12
Installing Python on Linux 12
Setuptools and pip 13
Development Tools 13
virtualenv 15
Installing Python on Windows 15

Setuptools and pip 18
virtualenv 18
Commercial Python Redistributions 19

3. Your Development Environment. 23
Text Editors 24
Sublime Text 25
Vim 25
Emacs 27
TextMate 28
Atom 29
Code 29
IDEs 29
PyCharm/IntelliJ IDEA 31
Aptana Studio 3/Eclipse + LiClipse + PyDev 32
WingIDE 32
Spyder 33
NINJA-IDE 33
Komodo IDE 33
Eric (the Eric Python IDE) 34
Visual Studio 34
Enhanced Interactive Tools 35
IDLE 35
IPython 35
bpython 36
Isolation Tools 36
Virtual Environments 36
pyenv 38
Autoenv 39
virtualenvwrapper 39
Buildout 40
Conda 41
Docker 42

Part II. Getting Down to Business

4. Writing Great Code. 47
Code Style 47
PEP 8 48
PEP 20 (a.k.a. The Zen of Python) 49
General Advice 50

Conventions 56
Idioms 59
Common Gotchas 62
Structuring Your Project 65
Modules 66
Packages 69
Object-Oriented Programming 70
Decorators 72
Dynamic Typing 73
Mutable and Immutable Types 74
Vendorizing Dependencies 76
Testing Your Code 76
Testing Basics 78
Examples 81
Other Popular Tools 84
Documentation 87
Project Documentation 87
Project Publication 88
Docstring Versus Block Comments 89
Logging 89
Logging in a Library 90
Logging in an Application 91
Choosing a License 93
Upstream Licenses 93
Options 94
Licensing Resources 95

5. Reading Great Code. 97
Common Features 98
HowDoI 99
Reading a Single-File Script 99
Structure Examples from HowDoI 102
Style Examples from HowDoI 103
Diamond 105
Reading a Larger Application 106
Structure Examples from Diamond 111
Style Examples from Diamond 115
Tablib 118
Reading a Small Library 118
Structure Examples from Tablib 122
Style Examples from Tablib 130
Requests 132

Reading a Larger Library 132
Structure Examples from Requests 136
Style Examples from Requests 141
Werkzeug 146
Reading Code in a Toolkit 147
Style Examples from Werkzeug 154
Structure Examples from Werkzeug 155
Flask 162
Reading Code in a Framework 162
Style Examples from Flask 168
Structure Examples from Flask 169

6. **Shipping Great Code**.. 173
Useful Vocabulary and Concepts 174
Packaging Your Code 175
Conda 175
PyPI 176
Freezing Your Code 179
PyInstaller 181
cx_Freeze 182
py2app 184
py2exe 184
bbFreeze 185
Packaging for Linux-Built Distributions 186
Executable ZIP Files 187

Part III. Scenario Guide

7. **User Interaction**.. 193
Jupyter Notebooks 193
Command-Line Applications 194
GUI Applications 202
Widget Libraries 202
Game Development 208
Web Applications 209
Web Frameworks/Microframeworks 210
Web Template Engines 213
Web Deployment 219

8. **Code Management and Improvement**.. 223
Continuous Integration 223

System Administration 224
Server Automation 226
System and Task Monitoring 231
Speed 233
Interfacing with C/C++/FORTRAN Libraries 243

9. **Software Interfaces.** 249
Web Clients 250
Web APIs 250
Data Serialization 255
Distributed Systems 258
Networking 258
Cryptography 264

10. **Data Manipulation.** 271
Scientific Applications 272
Text Manipulation and Text Mining 276
String Tools in Python's Standard Library 277
Image Manipulation 280

11. **Data Persistence.** 283
Structured Files 283
Database Libraries 284

A. **Additional Notes.** 301

Index. 311