

Contents

Acknowledgments	xxix
Introduction	xxxix
Chapter 1: Programming Basics and Strings	1
How Programming Is Different from Using a Computer	1
Programming Is Consistency	2
Programming Is Control	2
Programming Copes with Change	2
What All That Means Together	3
The First Steps	3
Starting codeEditor	3
Using codeEditor's Python Shell	4
<i>Try It Out: Starting the Python Shell</i>	4
Beginning to Use Python — Strings	5
What Is a String?	5
Why the Quotes?	6
<i>Try It Out: Entering Strings with Different Quotes</i>	6
Understanding Different Quotes	6
Putting Two Strings Together	8
<i>Try It Out: Using + to Combine Strings</i>	8
Putting Strings Together in Different Ways	9
<i>Try It Out: Using a Format Specifier to Populate a String</i>	9
<i>Try It Out: More String Formatting</i>	9
Displaying Strings with Print	10
<i>Try It Out: Printing Text with Print</i>	10
Summary	10
Exercises	11
Chapter 2: Numbers and Operators	13
Different Kinds of Numbers	13
Numbers in Python	14
<i>Try It Out: Using Type with Different Numbers</i>	14
<i>Try It Out: Creating an Imaginary Number</i>	15

Contents

Program Files	15
<i>Try It Out: Using the Shell with the Editor</i>	16
Using the Different Types	17
<i>Try It Out Including Different Numbers in Strings</i>	18
<i>Try It Out: Escaping the % Sign in Strings</i>	18
Basic Math	19
<i>Try It Out Doing Basic Math</i>	19
<i>Try It Out: Using the Modulus Operation</i>	20
Some Surprises	20
<i>Try It Out: Printing the Results</i>	21
Using Numbers	21
Order of Evaluation	21
<i>Try It Out: Using Math Operations</i>	21
Number Formats	22
<i>Try It Out: Using Number Formats</i>	22
Mistakes Will Happen	23
<i>Try It Out: Making Mistakes</i>	23
Some Unusual Cases	24
<i>Try It Out: Formatting Numbers as Octal and Hexadecimal</i>	24
Summary	24
Exercises	25
Chapter 3: Variables — Names for Values	27
Referring to Data – Using Names for Data	27
<i>Try It Out: Assigning Values to Names</i>	28
Changing Data Through Names	28
<i>Try It Out: Altering Named Values</i>	29
Copying Data	29
Names You Can't Use and Some Rules	29
Using More Built-in Types	30
Tuples — Unchanging Sequences of Data	30
<i>Try It Out: Creating and Using a Tuple</i>	30
<i>Try It Out: Accessing a Tuple Through Another Tuple</i>	31
Lists — Changeable Sequences of Data	33
<i>Try It Out Viewing the Elements of a List</i>	33
Dictionaries — Groupings of Data Indexed by Name	34
<i>Try It Out: Making a Dictionary</i>	34
<i>Try It Out: Getting the Keys from a Dictionary</i>	35
Treating a String Like a List	36
Special Types	38

Other Common Sequence Properties	38
Referencing the Last Elements	38
Ranges of Sequences	39
<i>Try It Out: Slicing Sequences</i>	39
Growing Lists by Appending Sequences	40
Using Lists to Temporarily Store Data	40
<i>Try It Out: Popping Elements from a List</i>	40
Summary	41
Exercises	42
Chapter 4: Making Decisions	43
<hr/>	
Comparing Values — Are They the Same?	43
<i>Try It Out: Comparing Values for Sameness</i>	43
Doing the Opposite — Not Equal	45
<i>Try It Out: Comparing Values for Difference</i>	45
Comparing Values — Which One Is More?	45
<i>Try It Out: Comparing Greater Than and Less Than</i>	45
More Than or Equal, Less Than or Equal	47
Reversing True and False	47
<i>Try It Out: Reversing the Outcome of a Test</i>	47
Looking for the Results of More Than One Comparison	48
How to Get Decisions Made	48
<i>Try It Out: Placing Tests within Tests</i>	49
Repetition	51
How to Do Something — Again and Again	51
<i>Try It Out: Using a while Loop</i>	51
Stopping the Repetition	52
<i>Try It Out: Using else While Repeating</i>	54
<i>Try It Out: Using continue to Keep Repeating</i>	54
Handling Errors	55
Trying Things Out	55
<i>Try It Out: Creating an Exception with Its Explanation</i>	56
Summary	57
Exercises	58
Chapter 5: Functions	59
<hr/>	
Putting Your Program into Its Own File	59
<i>Try It Out: Run a Program with Python -i</i>	61

Contents

Functions: Grouping Code under a Name	61
<i>Try It Out: Defining a Function</i>	61
Choosing a Name	62
Describing a Function in the Function	63
<i>Try It Out: Displaying __doc__</i>	63
The Same Name in Two Different Places	64
Making Notes to Yourself	65
<i>Try It Out: Experimenting with Comments</i>	65
Asking a Function to Use a Value You Provide	66
<i>Try It Out Invoking a Function with Parameters</i>	67
Checking Your Parameters	68
<i>Try It Out: Determining More Types with the type Function</i>	69
<i>Try It Out: Using Strings to Compare Types</i>	69
Setting a Default Value for a Parameter — Just in Case	70
<i>Try It Out: Setting a Default Parameter</i>	70
Calling Functions from within Other Functions	71
<i>Try It Out: Invoking the Completed Function</i>	72
Functions Inside of Functions	72
Flagging an Error on Your Own Terms	73
Layers of Functions	74
How to Read Deeper Errors	74
Summary	75
Exercises	76
Chapter 6: Classes and Objects	79
Thinking About Programming	79
Objects You Already Know	79
Looking Ahead: How You Want to Use Objects	81
Defining a Class	81
How Code Can Be Made into an Object	81
<i>Try It Out: Defining a Class</i>	82
<i>Try It Out: Creating an Object from Your Class</i>	82
<i>Try It Out: Writing an Internal Method</i>	84
<i>Try It Out: Writing Interface Methods</i>	85
<i>Try It Out: Using More Methods</i>	87
Objects and Their Scope	89
<i>Try It Out: Creating Another Class</i>	89
Summary	92
Exercises	93

Chapter 7: Organizing Programs	95
Modules	96
Importing a Module So That You Can Use It	96
Making a Module from Pre-existing Code	97
<i>Try It Out: Creating a Module</i>	97
<i>Try It Out: Exploring Your New Module</i>	98
Using Modules — Starting With the Command Line	99
<i>Try It Out: Printing sys.argv</i>	100
Changing How Import Works — Bringing in More	101
Packages	101
<i>Try It Out: Making the Files in the Kitchen Class</i>	102
Modules and Packages	103
Bringing Everything into the Current Scope	103
<i>Try It Out: Exporting Modules from a Package</i>	104
Re-importing Modules and Packages	104
<i>Try It Out: Examining sys.modules</i>	105
Basics of Testing Your Modules and Packages	106
Summary	106
Exercises	107
Chapter 8: Files and Directories	109
File Objects	109
Writing Text Files	110
Reading Text Files	111
<i>Try It Out: Printing the Lengths of Lines in the Sample File</i>	112
File Exceptions	113
Paths and Directories	113
Paths	114
Directory Contents	116
<i>Try It Out: Getting the Contents of a Directory</i>	116
<i>Try It Out: Listing the Contents of Your Desktop or Home Directory</i>	118
Obtaining Information about Files	118
Recursive Directory Listings	118
Renaming, Moving, Copying, and Removing Files	119
Example: Rotating Files	120
Creating and Removing Directories	121
Globbing	122

Contents

Pickles	123
<i>Try It Out: Creating a Pickle File</i>	123
Pickling Tips	124
Efficient Pickling	125
Summary	125
Exercises	125
Chapter 9: Other Features of the Language	127
<hr/>	
Lambda and Filter: Short Anonymous Functions	127
Reduce	128
<i>Try It Out: Working with Reduce</i>	128
Map: Short-Circuiting Loops	129
<i>Try It Out: Use Map</i>	129
Decisions within Lists — List Comprehension	130
Generating Lists for Loops	131
<i>Try It Out: Examining an xrange Object</i>	132
Special String Substitution Using Dictionaries	133
<i>Try It Out: String Formatting with Dictionaries</i>	133
Featured Modules	134
Getopt — Getting Options from the Command Line	134
Using More Than One Process	137
Threads — Doing Many Things in the Same Process	139
Storing Passwords	140
Summary	141
Exercises	142
Chapter 10: Building a Module	143
<hr/>	
Exploring Modules	143
Importing Modules	145
Finding Modules	145
Digging through Modules	146
Creating Modules and Packages	150
<i>Try It Out: Creating a Module with Functions</i>	150
Working with Classes	151
Defining Object-Oriented Programming	151
Creating Classes	151
<i>Try It Out: Creating a Meal Class</i>	152
Extending Existing Classes	153

Finishing Your Modules	154
Defining Module-Specific Errors	154
Choosing What to Export	155
Documenting Your Modules	156
<i>Try It Out: Viewing Module Documentation</i>	157
Testing Your Module	162
Running a Module as a Program	164
<i>Try It Out: Running a Module</i>	164
Creating a Whole Module	165
<i>Try It Out: Finishing a Module</i>	165
<i>Try It Out: Smashing Imports</i>	169
Installing Your Modules	170
<i>Try It Out: Creating an Installable Package</i>	171
Summary	174
Exercises	174
Chapter 11: Text Processing	175
<hr/>	
Why Text Processing Is So Useful	175
Searching for Files	176
Clipping Logs	177
Sifting through Mail	178
Navigating the File System with the os Module	178
<i>Try It Out: Listing Files and Playing with Paths</i>	180
<i>Try It Out: Searching for Files of a Particular Type</i>	181
<i>Try It Out: Refining a Search</i>	183
Working with Regular Expressions and the re Module	184
<i>Try It Out: Fun with Regular Expressions</i>	186
<i>Try It Out: Adding Tests</i>	187
Summary	189
Exercises	189
Chapter 12: Testing	191
<hr/>	
Assertions	191
<i>Try It Out: Using Assert</i>	192
Test Cases and Test Suites	193
<i>Try It Out: Testing Addition</i>	194
<i>Try It Out: Testing Faulty Addition</i>	195
Test Fixtures	196
<i>Try It Out: Working with Test Fixtures</i>	197

Contents

Putting It All Together with Extreme Programming	199
Implementing a Search Utility in Python	200
<i>Try It Out: Writing a Test Suite First</i>	201
<i>Try It Out: A General-Purpose Search Framework</i>	203
A More Powerful Python Search	205
<i>Try It Out: Extending the Search Framework</i>	206
Formal Testing in the Software Life Cycle	207
Summary	208
Chapter 13: Writing a GUI with Python	209
GUI Programming Toolkits for Python	209
PyGTK Introduction	210
pyGTK Resources	211
Creating GUI Widgets with pyGTK	213
<i>Try It Out: Writing a Simple pyGTK Program</i>	213
GUI Signals	214
GUI Helper Threads and the GUI Event Queue	216
<i>Try It Out: Writing a Multithreaded pyGTK App</i>	219
Widget Packing	222
Glade: a GUI Builder for pyGTK	223
GUI Builders for Other GUI Frameworks	224
Using libGlade with Python	225
A Glade Walkthrough	225
Starting Glade	226
Creating a Project	227
Using the Palette to Create a Window	227
Putting Widgets into the Window	228
Glade Creates an XML Representation of the GUI	230
<i>Try It Out: Building a GUI from a Glade File</i>	231
Creating a Real Glade Application	231
Advanced Widgets	238
Further Enhancing PyRAP	241
Summary	248
Exercises	248
Chapter 14: Accessing Databases	249
Working with DBM Persistent Dictionaries	250
Choosing a DBM Module	250
Creating Persistent Dictionaries	251
<i>Try It Out: Creating a Persistent Dictionary</i>	251

Accessing Persistent Dictionaries	252
<i>Try It Out: Accessing Persistent Dictionaries</i>	253
Deciding When to Use DBM and When to Use a Relational Database	255
Working with Relational Databases	255
Writing SQL Statements	257
Defining Tables	259
Setting Up a Database	260
<i>Try It Out: Creating a Gadfly Database</i>	261
Using the Python Database APIs	262
Downloading Modules	263
Creating Connections	263
Working with Cursors	264
<i>Try It Out: Inserting Records</i>	264
<i>Try It Out: Writing a Simple Query</i>	266
<i>Try It Out: Writing a Complex Join</i>	267
<i>Try It Out: Updating an Employee's Manager</i>	269
<i>Try It Out: Removing Employees</i>	270
Working with Transactions and Committing the Results	271
Examining Module Capabilities and Metadata	272
Handling Errors	272
Summary	273
Exercises	274
Chapter 15: Using Python for XML	275
What Is XML?	275
A Hierarchical Markup Language	275
A Family of Standards	277
What Is a Schema/DTD?	278
What Are Document Models For?	278
Do You Need One?	278
Document Type Definitions	278
An Example DTD	278
DTDs Aren't Exactly XML	280
Limitations of DTDs	280
Schemas	280
An Example Schema	280
Schemas Are Pure XML	281
Schemas Are Hierarchical	281
Other Advantages of Schemas	281
Schemas Are Less Widely Supported	281

XPath	282
HTML as a Subset of XML	282
The HTML DTDs	283
HTMLParser	283
<i>Try It Out: Using HTMLParser</i>	283
htmllib	284
<i>Try It Out: Using htmllib</i>	284
XML Libraries Available for Python	285
Validating XML Using Python	285
What Is Validation?	286
Well-Formedness versus Validation	286
Available Tools	286
<i>Try It Out: Validation Using xmlproc</i>	286
What Is SAX?	287
Stream-based	288
Event-driven	288
What Is DOM?	288
In-memory Access	288
Why Use SAX or DOM	289
Capability Trade-Offs	289
Memory Considerations	289
Speed Considerations	289
SAX and DOM Parsers Available for Python	289
PyXML	290
xml.sax	290
xml.dom.minidom	290
<i>Try It Out: Working with XML Using DOM</i>	290
<i>Try It Out: Working with XML Using SAX</i>	292
Intro to XSLT	293
XSLT Is XML	293
Transformation and Formatting Language	293
Functional, Template-Driven	293
Using Python to Transform XML Using XSLT	294
<i>Try It Out: Transforming XML with XSLT</i>	294
Putting It All Together: Working with RSS	296
RSS Overview and Vocabulary	296
Making Sense of It All	296
RSS Vocabulary	297
An RSS DTD	297

A Real-World Problem	297
<i>Try It Out: Creating an RSS Feed</i>	298
Creating the Document	300
Checking It Against the DTD	301
Another Real-World Problem	301
<i>Try It Out: Creating An Aggregator</i>	301
Summary	303
Exercises	303
Chapter 16: Network Programming	305
<i>Try It Out: Sending Some E-mail</i>	305
Understanding Protocols	307
Comparing Protocols and Programming Languages	307
The Internet Protocol Stack	308
A Little Bit About the Internet Protocol	309
Internet Addresses	309
Internet Ports	310
Sending Internet E-mail	311
The E-mail File Format	311
MIME Messages	313
MIME Encodings: Quoted-printable and Base64	313
MIME Content Types	314
<i>Try It Out: Creating a MIME Message with an Attachment</i>	315
MIME Multipart Messages	316
<i>Try It Out: Building E-mail Messages with SmartMessage</i>	320
Sending Mail with SMTP and smtpplib	321
<i>Try It Out: Sending Mail with MailServer</i>	323
Retrieving Internet E-mail	323
Parsing a Local Mail Spool with mailbox	323
<i>Try It Out: Printing a Summary of Your Mailbox</i>	324
Fetching Mail from a POP3 Server with poplib	325
<i>Try It Out: Printing a Summary of Your POP3 Mailbox</i>	327
Fetching Mail from an IMAP Server with imaplib	327
<i>Try It Out: Printing a Summary of Your IMAP Mailbox</i>	329
IMAP's Unique Message IDs	330
<i>Try It Out: Fetching a Message by Unique ID</i>	330
Secure POP3 and IMAP	331
Webmail Applications Are Not E-mail Applications	331
Socket Programming	331
Introduction to Sockets	332
<i>Try It Out: Connecting to the SuperSimpleSocketServer with Telnet</i>	333

Contents

Binding to an External Hostname	334
The Mirror Server	335
<i>Try It Out: Mirroring Text with the MirrorServer</i>	336
The Mirror Client	336
SocketServer	337
Multithreaded Servers	339
The Python Chat Server	340
Design of the Python Chat Server	340
The Python Chat Server Protocol	341
Our Hypothetical Protocol in Action	341
Initial Connection	342
Chat Text	342
Server Commands	342
General Guidelines	343
The Python Chat Client	346
Single-Threaded Multitasking with select	348
Other Topics	350
Miscellaneous Considerations for Protocol Design	350
Trusted Servers	350
Terse Protocols	350
The Twisted Framework	351
Deferred Objects	351
The Peer-to-Peer Architecture	354
Summary	354
Exercises	354
Chapter 17: Extension Programming with C	355
<hr/>	
Extension Module Outline	356
Building and Installing Extension Modules	358
Passing Parameters from Python to C	360
Returning Values from C to Python	363
The LAME Project	364
The LAME Extension Module	368
Using Python Objects from C Code	380
Summary	383
Exercises	383
Chapter 18: Writing Shareware and Commercial Programs	385
<hr/>	
A Case Study: Background	385
How Much Python Should You Use?	386

Pure Python Licensing	387
Web Services Are Your Friend	388
Pricing Strategies	389
Watermarking	390
Other Models	394
Selling as a Platform, Rather Than a Product	395
Your Development Environment	395
Finding Python Programmers	396
Training non-Python Programmers	397
Python Employment Resources	397
Python Problems	397
Porting to Other Versions of Python	397
Porting to Other Operating Systems	398
Debugging Threads	399
Common Gotchas	399
Portable Distribution	400
Essential Libraries	401
Timeoutsocket	401
PyGTK	402
GEOip	402
Summary	403
Chapter 19: Numerical Programming	405
Numbers in Python	405
Integers	406
Long Integers	406
Floating-point Numbers	407
Formatting Numbers	408
Characters as Numbers	410
Mathematics	412
Arithmetic	412
Built-in Math Functions	414
The math Module	415
Complex Numbers	416
Arrays	418
The array Module	420
The numpy Package	422
Using Arrays	422
Computing the Standard Deviation	423
Summary	424
Exercises	425

Chapter 20: Python in the Enterprise **427**

Enterprise Applications	428
Document Management	428
The Evolution of Document Management Systems	429
What You Want in a Document Management System	430
People in Directories	431
Taking Action with Workflow	432
Auditing, Sarbanes-Oxley, and What You Need to Know	433
Auditing and Document Management	434
Working with Actual Enterprise Systems	435
Introducing the wftk Workflow Toolkit	435
<i>Try It Out: Very Simple Record Retrieval</i>	436
<i>Try It Out: Very Simple Record Storage</i>	438
<i>Try It Out: Data Storage in MySQL</i>	439
<i>Try It Out: Storing and Retrieving Documents</i>	441
<i>Try It Out: A Document Retention Framework</i>	446
The python-ldap Module	448
<i>Try It Out: Using Basic OpenLDAP Tools</i>	449
<i>Try It Out: Simple LDAP Search</i>	451
More LDAP	453
Back to the wftk	453
<i>Try It Out: Simple Workflow Trigger</i>	454
<i>Try It Out: Action Queue Handler</i>	456
Summary	458
Exercises	458

Chapter 21: Web Applications and Web Services **459**

REST: The Architecture of the Web	460
Characteristics of REST	460
A Distributed Network of Interlinked Documents	461
A Client-Server Architecture	461
Servers Are Stateless	461
Resources	461
Representations	462
REST Operations	462
HTTP: Real-World REST	463
<i>Try It Out: Python's Three-Line Web Server</i>	463
The Visible Web Server	464
<i>Try It Out: Seeing an HTTP Request and Response</i>	465
The HTTP Request	466
The HTTP Response	467

CGI: Turning Scripts into Web Applications	468
<i>Try It Out: Running a CGI Script</i>	469
The Web Server Makes a Deal with the CGI Script	470
CGI's Special Environment Variables	471
Accepting User Input through HTML Forms	473
The cgi Module: Parsing HTML Forms	474
<i>Try It Out: Printing Any HTML Form Submission</i>	478
Building a Wiki	480
The BittyWiki Core Library	481
Back-end Storage	481
WikiWords	481
Writing the BittyWiki Core	481
<i>Try It Out: Creating Wiki Pages from an Interactive Python Session</i>	483
The BittyWiki Web Interface	484
Resources	484
Request Structure	484
But Wait — There's More (Resources)	485
Wiki Markup	486
Web Services	493
How Web Services Work	494
REST Web Services	494
REST Quick Start: Finding Bargains on Amazon.com	495
<i>Try It Out: Peeking at an Amazon Web Services Response</i>	496
Introducing WishListBargainFinder	497
Giving BittyWiki a REST API	500
Wiki Search-and-Replace Using the REST Web Service	503
<i>Try It Out: Wiki Searching and Replacing</i>	507
XML-RPC	508
XML-RPC Quick Start: Get Tech News from Meerkat	509
The XML-RPC Request	511
Representation of Data in XML-RPC	512
The XML-RPC Response	513
If Something Goes Wrong	513
Exposing the BittyWiki API through XML-RPC	514
<i>Try It Out: Manipulating BittyWiki through XML-RPC</i>	517
Wiki Search-and-Replace Using the XML-RPC Web Service	518
SOAP	520
SOAP Quick Start: Surfing the Google API	520
The SOAP Request	522
The SOAP Response	524
If Something Goes Wrong	524

Contents

Exposing a SOAP Interface to BittyWiki	525
<i>Try It Out: Manipulating BittyWiki through SOAP</i>	526
Wiki Search-and-Replace Using the SOAP Web Service	527
Documenting Your Web Service API	529
Human-Readable API Documentation	529
The BittyWiki REST API Document	529
The BittyWiki XML-RPC API Document	529
The BittyWiki SOAP API Document	530
The XML-RPC Introspection API	530
<i>Try It Out: Using the XML-RPC Introspection API</i>	530
WSDL	531
<i>Try It Out: Manipulating BittyWiki through a WSDL Proxy</i>	533
Choosing a Web Service Standard	534
Web Service Etiquette	535
For Consumers of Web Services	535
For Producers of Web Services	535
Using Web Applications as Web Services	536
A Sampling of Publicly Available Web Services	536
Summary	538
Exercises	538
Chapter 22: Integrating Java with Python	539
Scripting within Java Applications	540
Comparing Python Implementations	541
Installing Jython	541
Running Jython	542
Running Jython Interactively	542
<i>Try It Out: Running the Jython Interpreter</i>	542
Running Jython Scripts	543
<i>Try It Out Running a Python Script</i>	543
Controlling the jython Script	544
Making Executable Commands	545
<i>Try It Out: Making an Executable Script</i>	546
Running Jython on Your Own	546
Packaging Jython-Based Applications	547
Integrating Java and Jython	547
Using Java Classes in Jython	548
<i>Try It Out: Calling on Java Classes</i>	548
<i>Try It Out: Creating a User Interface from Jython</i>	550
Accessing Databases from Jython	552
Working with the Python DB API	553

Setting Up a Database	554
<i>Try It Out: Create Tables</i>	555
Writing J2EE Servlets in Jython	558
Setting Up an Application Server	559
Adding the PyServlet to an Application Server	560
Extending HttpServlet	561
<i>Try It Out: Writing a Python Servlet</i>	562
Choosing Tools for Jython	564
Testing from Jython	565
<i>Try It Out: Exploring Your Environment with Jython</i>	565
Embedding the Jython Interpreter	566
Calling Jython Scripts from Java	566
<i>Try It Out: Embedding Jython</i>	567
Compiling Python Code to Java	568
Handling Differences between C Python and Jython	569
Summary	570
Exercises	571
Appendix A: Answers to Exercises	573
Appendix B: Online Resources	605
Appendix C: What's New in Python 2.4	609
Glossary	613
Index	623

