

Python

ESSENTIAL REFERENCE

Fourth Edition

David M. Beazley

vVAddison-Wesley

Upper Saddle River, NJ • Boston • Indianapolis • San Francisco
New York • Toronto • Montreal • London • Munich • Paris • Madrid
Cape Town • Sydney • Tokyo • Singapore • Mexico City

Table of Contents

Introduction	1
--------------	---

I: The Python Language

1 A Tutorial Introduction 5

Running Python	5
Variables and Arithmetic Expressions	7
Conditionals	9
File Input and Output	10
Strings	11
Lists	12
Tuples	14
Sets	15
Dictionaries	16
Iteration and Looping	17
Functions	18
Generators	19
Coroutines	20
Objects and Classes	21
Exceptions	22
Modules	23
Getting Help	24

2 Lexical Conventions and Syntax 25

Line Structure and Indentation	25
Identifiers and Reserved Words	26
Numeric Literals	26
String Literals	27
Containers	29
Operators, Delimiters, and Special Symbols	30
Documentation Strings	30
Decorators	30
Source Code Encoding	31

3 Types and Objects 33

Terminology	33
Object Identity and Type	33
Reference Counting and Garbage Collection	34
References and Copies	35

First-Class Objects	36
Built-in Types for Representing Data	37
The None Type	38
Numeric Types	38
Sequence Types	39
Mapping Types	44
Set Types	46
Built-in Types for Representing Program Structure	47
Callable Types	47
Classes, Types, and Instances	50
Modules	50
Built-in Types for Interpreter Internals	51
Code Objects	51
Frame Objects	52
Traceback Objects	52
Generator Objects	53
Slice Objects	53
Ellipsis Object	54
Object Behavior and Special Methods	54
Object Creation and Destruction	54
Object String Representation	55
Object Comparison and Ordering	56
Type Checking	57
Attribute Access	57
Attribute Wrapping and Descriptors	58
Sequence and Mapping Methods	58
Iteration	59
Mathematical Operations	60
Callable Interface	62
Context Management Protocol	62
Object Inspection and <code>dir()</code>	63
Operators and Expressions	65
Operations on Numbers	65
Operations on Sequences	67
String Formatting	70
Advanced String Formatting	72
Operations on Dictionaries	74
Operations on Sets	75
Augmented Assignment	75

	The Attribute (.) Operator	76
	The Function Call () Operator	76
	Conversion Functions	76
	Boolean Expressions and Truth Values	77
	Object Equality and Identity	78
	Order of Evaluation	78
	Conditional Expressions	79
5	Program Structure and Control Flow	81
	Program Structure and Execution	81
	Conditional Execution	81
	Loops and Iteration	82
	Exceptions	84
	Built-in Exceptions	86
	Defining New Exceptions	88
	Context Managers and the with Statement	89
	Assertions and <code>__debug__</code>	91
6	Functions and Functional Programming	93
	Functions	93
	Parameter Passing and Return Values	95
	Scoping Rules	96
	Functions as Objects and Closures	98
	Decorators	101
	Generators and <code>yield</code>	102
	Coroutines and <code>yield</code> Expressions	104
	Using Generators and Coroutines	106
	List Comprehensions	108
	Generator Expressions	109
	Declarative Programming	110
	The <code>lambda</code> Operator	112
	Recursion	112
	Documentation Strings	113
	Function Attributes	114
	<code>eval()</code> , <code>exec()</code> , and <code>compile()</code>	115
7	Classes and Object-Oriented Programming	117
	The <code>class</code> Statement	117
	Class Instances	118
	Scoping Rules	118
	Inheritance	119

Contents

Polymorphism Dynamic Binding and Duck Typing	122
Static Methods and Class Methods	123
Properties	124
Descriptors	126
Data Encapsulation and Private Attributes	127
Object Memory Management	128
Object Representation and Attribute Binding	131
<code>__slots__</code>	132
Operator Overloading	133
Types and Class Membership Tests	134
Abstract Base Classes	136
Metaclasses	138
Class Decorators	141
8 Modules, Packages, and Distribution	143
Modules and the import Statement	143
Importing Selected Symbols from a Module	145
Execution as the Main Program	146
The Module Search Path	147
Module Loading and Compilation	147
Module Reloading and Unloading	149
Packages	149
Distributing Python Programs and Libraries	152
Installing Third-Party Libraries	154
9 Input and Output	157
Reading Command-Line Options	157
Environment Variables	158
Files and File Objects	158
Standard Input, Output, and Error	161
The <code>print</code> Statement	162
The <code>print()</code> Function	163
Variable Interpolation in Text Output	163
Generating Output	164
Unicode String Handling	165
Unicode I/O	167
Unicode Data Encodings	168
Unicode Character Properties	170
Object Persistence and the pickle Module	171

10	Execution Environment	173
	Interpreter Options and Environment	173
	Interactive Sessions	175
	Launching Python Applications	176
	Site Configuration Files	177
	Per-user Site Packages	177
	Enabling Future Features	178
	Program Termination	179
11	Testing, Debugging, Profiling, and Tuning	181
	Documentation Strings and the doctest Module	181
	Unit Testing and the unittest Module	183
	The Python Debugger and the pdb Module	186
	Debugger Commands	187
	Debugging from the Command Line	189
	Configuring the Debugger	190
	Program Profiling	190
	Tuning and Optimization	191
	Making Timing Measurements	191
	Making Memory Measurements	192
	Disassembly	193
	Tuning Strategies	194
II:	The Python Library	199
12	Built-in Functions and Exceptions	201
	Built-in Functions and Types	201
	Built-in Exceptions	212
	Exception Base Classes	212
	Exception Instances	212
	Predefined Exception Classes	213
	Built-in Warnings	216
	future_builtins	217
13	Python Runtime Services	219
	atexit	219
	copy	219
	Notes	220

gc	220
Notes	222
inspect	222
marshal	226
Notes	226
pickle	226
Notes	229
sys	229
Variables	229
Functions	233
traceback	235
types	237
Notes	237
warnings	238
Notes	239
weakref	240
Example	242
Notes	242

14 Mathematics 243

decimal	243
Decimal Objects	244
Context Objects	244
Functions and Constants	247
Examples	248
Notes	249
fractions	250
math	251
Notes	252
numbers	252
Notes	253
random	254
Seeding and Initialization	254
Random Integers	254
Random Sequences	254
Real-Valued Random Distributions	255
Notes	256

15	Data Structures, Algorithms, and Code Simplification	257
	abc	257
	array	259
	Notes	261
	bisect	261
	collections	262
	deque and defaultdict	262
	Named Tuples	263
	Abstract Base Classes	265
	contextlib	267
	functools	268
	heapq	269
	itertools	270
	Examples	273
	operator	273
16	String and Text Handling	277
	codecs	277
	Low-Level codecs Interface	277
	I/O-Related Functions	279
	Useful Constants	279
	Standard Encodings	280
	Notes	280
	re	281
	Pattern Syntax	281
	Functions	283
	Regular Expression Objects	284
	Match Objects	285
	Example	286
	Notes	287
	string	287
	Constants	287
	Formatter Objects	288
	Template Strings	289
	Utility Functions	290
	struct	290
	Packing and Unpacking Functions	290
	Struct Objects	291

Format Codes	291
Notes	292
unicodedata	293

17 Python Database Access 297

Relational Database API Specification	297
Connections	297
Cursors	298
Forming Queries	300
Type Objects	301
Error Handling	302
Multithreading	302
Mapping Results into Dictionaries	303
Database API Extensions	303
sqlite3 Module	303
Module-Level Functions	304
Connection Objects	305
Cursors and Basic Operations	308
DBM-Style Database Modules	310
shelve Module	311

18 File and Directory Handling 313

bz2	313
filecmp	314
fnmatch	316
Examples	316
glob	317
Example	317
gzip	317
Notes	317
shutil	318
tarfile	319
Exceptions	322
Example	322
tempfile	323
zipfile	324
zlib	328

19 Operating System Services 331

- commands 331
 - Notes 332
- configParser, Configparser 332
 - The ConfigParser Class 332
 - Example 334
 - Notes 336
- datetime 336
 - date Objects 336
 - time Objects 338
 - datetime objects 339
 - timedelta objects 340
 - Mathematical Operations Involving Dates 341
 - tzinfo Objects 342
 - Date and Time Parsing 343
- errno 343
 - POSIX Error Codes 344
 - Windows Error Codes 346
- fcntl 347
 - Example 348
 - Notes 349
- io 349
 - Base I/O Interface 349
 - Raw I/O 350
 - Buffered Binary I/O 351
 - Text I/O 353
 - The open () Function 354
 - Abstract Base Classes 354
- logging 355
 - Logging Levels 355
 - Basic Configuration 355
 - Logger Objects 356
 - Handler Objects 362
 - Message Formatting 364
 - Miscellaneous Utility Functions 366
 - Logging Configuration 366
 - Performance Considerations 369
 - Notes 369

mmap	369
Notes	372
msvcrt	372
optparse	374
Example	377
Notes	378
os	378
Process Environment	379
File Creation and File Descriptors	381
Files and Directories	386
Process Management	390
System Configuration	395
Exceptions	396
os. path	396
signal	399
Example	401
Notes	401
subprocess	402
Examples	404
Notes	404
time	405
Notes	407
winreg	408
Notes	411

20 Threads and Concurrency 413

Basic Concepts	413
Concurrent Programming and Python	414
multiprocessing	415
Processes	415
Interprocess Communication	417
Process Pools	424
Shared Data and Synchronization	426
Managed Objects	428
Connections	433
Miscellaneous Utility Functions	434
General Advice on Multiprocessing	435
threading	436
Thread Objects	436
Timer Objects	437

Lock Objects	438
RLock	438
Semaphore and Bounded Semaphore	439
Events	440
Condition Variables	441
Working with Locks	442
Thread Termination and Suspension	443
Utility Functions	443
The Global Interpreter Lock	444
Programming with Threads	444
queue, Queue	444
Queue Example with Threads	445
Coroutines and Microthreading	446
21 Network Programming and Sockets	449
Network Programming Basics	449
asynchat	452
asyncore	455
Example	457
select	459
Advanced Module Features	460
Advanced Asynchronous I/O Example	460
When to Consider Asynchronous Networking	467
socket	469
Address Families	470
Socket Types	470
Addressing	471
Functions	473
Exceptions	485
Example	485
Notes	486
ssl	486
Examples	489
SocketServer	489
Handlers	490
Servers	491
Defining Customized Servers	492
Customization of Application Servers	494

22 Internet Application Programming 497

ftplib 497

Example 500

http Package 500

http.client (httplib) 502

http.server (BaseHTTPServer, CGIHTTPServer,
SimpleHTTPServer) 506

http.cookies (Cookie) 511

http.cookiejar (cookielib) 513

smtplib 513

Example 514

urllib Package 514

urllib.request (urllib2) 515

urllib.response 520

urllib.parse 520

urllib.error 523

urllib.robotparser (robotparser) 523

Notes 524

xmlrpc Package 524

xmlrpc.client (xmlrpclib) 524

xmlrpc.server (SimpleXMLRPCServer, DocXMLRPCServer)
527

23 Web Programming 531

cgi 533

CGI Programming Advice 537

Notes 538

cgitb 539

wsgiref 540

The WSGI Specification 540

wsgiref Package 542

webbrowser 544

24 Internet Data Handling and Encoding 545

base64 545

binascii 547

csv 548

Dialects 551

Example 551

email Package	552
Parsing Email	552
Composing Email	555
Notes	558
hashlib	559
hmac	559
Example	560
HTMLParser	561
Example	562
json	563
mimetypes	566
quopri	567
xml Package	568
XML Example Document	569
xml.dom.minidom	570
xml.etree.ElementTree	573
xml.sax	580
xml.sax.saxutils	583

25 Miscellaneous Library Modules 585

Python Services	585
String Processing	586
Operating System Modules	586
Network	587
Internet Data Handling	587
Internationalization	587
Multimedia Services	588
Miscellaneous	588

III: Extending and Embedding 589

26 Extending and Embedding Python 591

Extension Modules	591
An Extension Module Prototype	593
Naming Extension Modules	595
Compiling and Packaging Extensions	596
Type Conversion from Python to C	597
Type Conversion from C to Python	602

Adding Values to a Module	604
Error Handling	605
Reference Counting	607
Threads	607
Embedding the Python Interpreter	608
An Embedding Template	608
Compilation and Linking	608
Basic Interpreter Operation and Setup	608
Accessing Python from C	610
Converting Python Objects to C	611
ctypes	612
Loading Shared Libraries	612
Foreign Functions	612
Datatypes	613
Calling Foreign Functions	615
Alternative Type Construction Methods	616
Utility Functions	617
Example	618
Advanced Extending and Embedding	619
Jython and IronPython	620

Appendix Python 3 621

Who Should Be Using Python 3?	621
New Language Features	622
Source Code Encoding and Identifiers	622
Set Literals	622
Set and Dictionary Comprehensions	623
Extended Iterable Unpacking	623
Nonlocal Variables	624
Function Annotations	624
Keyword-Only Arguments	625
Ellipsis as an Expression	626
Chained Exceptions	626
Improved super()	627
Advanced Metaclasses	627
Common Pitfalls	629
Text Versus Bytes	629
New I/O System	631

print () and exec () Functions	631
Use of Iterators and Views	632
Integers and Integer Division	633
Comparisons	633
Iterators and Generators	633
File Names, Arguments, and Environment Variables	633
Library Reorganization	634
Absolute Imports	634
Code Migration and 2to3	634
Porting Code to Python 2.6	634
Providing Test Coverage	635
Using the 2to3 Tool	635
A Practical Porting Strategy	637
Simultaneous Python 2 and Python 3 Support	638
Participate	638
Index	639