

Preface

The Python programming language reconciles many apparent contradictions: elegant yet pragmatic, simple yet powerful, it's very high-level yet doesn't get in your way when you need to fiddle with bits and bytes, and it's suitable for novice programmers and great for experts, too.

This book is intended for programmers with some previous exposure to Python, as well as experienced programmers coming to Python for the first time from other languages. It provides a quick reference to Python itself, the most commonly used parts of its vast standard library, and a few of the most popular and useful third-party modules and packages. The Python ecosystem has grown so much in richness, scope, and complexity that a single volume can no longer reasonably hope to be encyclopedic. Still, the book covers a wide range of application areas, including web and network programming, XML handling, database interactions, and high-speed numeric computing. It also explores Python's cross-platform capabilities and the basics of extending Python and embedding it in other applications.

How To Use This Book

While you can read this volume linearly from the beginning, we also aim for it to be a useful reference for the working programmer. You may choose to use the index to locate items of interest, or to read specific chapters for coverage of their particular topics. However you use it, we sincerely hope you enjoy reading what represents the fruit of the best part of a year's work for the team.

The book has five parts, as follows.

Part I, Getting Started with Python

Chapter 1, “Introduction to Python”

Covers the general characteristics of the Python language, its implementations, where to get help and information, how to participate in the Python community, and how to obtain and install Python on your computer(s) or run it in your browser.

Chapter 2, “The Python Interpreter”

Covers the Python interpreter program, its command-line options, and how to use it to run Python programs and in interactive sessions. The chapter mentions text editors for editing Python programs and auxiliary programs for checking your Python sources, along with some full-fledged integrated development environments, including IDLE, which comes free with standard Python. The chapter also covers running Python programs from the command line.

Part II, Core Python Language and Built-ins

Chapter 3, “The Python Language”

Covers Python syntax, built-in data types, expressions, statements, control flow, and how to write and call functions.

Chapter 4, “Object-Oriented Python”

Covers object-oriented programming in Python.

Chapter 5, “Type Annotations”

Covers how to add type information to your Python code, to gain type hinting and autocomplete help from modern code editors and support static type checking from type checkers and linters.

Chapter 6, “Exceptions”

Covers how to use exceptions for errors and special situations, logging, and how to write code to automatically clean up when exceptions occur.

Chapter 7, “Modules and Packages”

Covers how Python lets you group code into modules and packages, how to define and import modules, and how to install third-party Python packages. This chapter also covers working with virtual environments to isolate project dependencies.

Chapter 8, “Core Built-ins and Standard Library Modules”

Covers built-in data types and functions, and some of the most fundamental modules in the Python standard library (roughly speaking, the set of modules supplying functionality that, in some other languages, is built into the language itself).

Chapter 9, “Strings and Things”

Covers Python's facilities for processing strings, including Unicode strings, bytestrings, and string literals.

Chapter 10, "Regular Expressions"

Covers Python's support for regular expressions.

Part III, Python Library and Extension Modules

Chapter 11, "File and Text Operations"

Covers dealing with files and text with many modules from Python's standard library and platform-specific extensions for rich text I/O. This chapter also covers issues regarding internationalization and localization.

Chapter 12, "Persistence and Databases"

Covers Python's serialization and persistence mechanisms and its interfaces to DBM databases and relational (SQL-based) databases, particularly the handy SQLite that comes with Python's standard library.

Chapter 13, "Time Operations"

Covers dealing with times and dates in Python, with the standard library and third-party extensions.

Chapter 14, "Customizing Execution"

Covers ways to achieve advanced execution control in Python, including execution of dynamically generated code and control of garbage collection. This chapter also covers some Python internal types, and the specific issue of registering "cleanup" functions to execute at program termination time.

Chapter 15, "Concurrency: Threads and Processes"

Covers Python's functionality for concurrent execution, both via multiple threads running within one process and via multiple processes running on a single machine.¹ This chapter also covers how to access the process's environment, and how to access files via memory-mapping mechanisms.

Chapter 16, "Numeric Processing"

Covers Python's features for numeric computations, both in standard library modules and in third-party extension packages; in particular, how to use decimal numbers or fractions instead of the default binary floating-point numbers. This chapter also covers how to get and use pseudorandom and truly random numbers, and how to speedily process whole arrays (and matrices) of numbers.

Chapter 17, "Testing, Debugging, and Optimizing"

Covers Python tools and approaches that help you make sure that your programs are correct (i.e., that they do what they're meant to do), find and fix errors in your programs, and check and enhance your programs' performance. This chapter also covers the concept of warnings and the Python library module that deals with them.

Part IV, Network and Web Programming

Chapter 18, “Networking Basics”

Covers the basics of networking with Python.

Chapter 19, “Client-Side Network Protocol Modules”

Covers modules in Python’s standard library to write network client programs, particularly for dealing with various network protocols from the client side, sending and receiving emails, and handling URLs.

Chapter 20, “Serving HTTP”

Covers how to serve HTTP for web applications in Python, using popular third-party lightweight Python frameworks leveraging Python’s WSGI standard interface to web servers.

Chapter 21, “Email, MIME, and Other Network Encodings”

Covers how to process email messages and other network-structured and encoded documents in Python.

Chapter 22, “Structured Text: HTML”

Covers popular third-party Python extension modules to process, modify, and generate HTML documents.

Chapter 23, “Structured Text: XML”

Covers Python library modules and popular extensions to process, modify, and generate XML documents.

Part V, Extending, Distributing, and Version Upgrade and Migration

Chapters 24 and 25 are included in summary form in the print edition of this book. You will find the full content of these chapters in the supporting online repository, described in **“How to Contact Us”**.

Chapter 24, “Packaging Programs and Extensions”

Covers tools and modules to package and share Python modules and applications.

Chapter 25, “Extending and Embedding Classic Python”

Covers how to code Python extension modules using Python’s C API, Cython, and other tools.

Chapter 26, “v3.7 to v3.n Migration”

Covers topics and best practices for planning and deploying version upgrades for Python users ranging from individuals to library maintainers to enterprise-wide deployment and support staff.

Appendix, “New Features and Changes in Python 3.7 Through 3.11”

Provides a detailed list of features and changes in Python language syntax and the standard library, by version.

Conventions Used in This Book

The following conventions are used throughout this book.

Reference Conventions

In the function/method reference entries, when feasible, each optional parameter is shown with a default value using the Python syntax *name=value*. Built-in functions need not accept named parameters, so parameter names may not be significant. Some optional parameters are best explained in terms of their presence or absence, rather than through default values. In such cases, we indicate that a parameter is optional by enclosing it in brackets ([]). When more than one argument is optional, brackets can be nested.

Version Conventions

This book covers changes and features in Python versions 3.7 through 3.11.

Python 3.7 serves as the base version for all tables and code examples, unless otherwise noted.² You will see these notations to indicate changes or features added and removed across the range of covered versions:

- **3.x+** marks a feature introduced in version 3.x, not available in prior versions.
- **-3.x** marks a feature removed in version 3.x, available only in prior versions.

Typographic Conventions

Please note that, for display reasons, our code snippets and samples may sometimes depart from [PEP 8](#). We do not recommend taking such liberties in your code. Instead, use a utility like [black](#) to adopt a canonical layout style.

The following typographical conventions are used in this book:

Italic

Used for file and directory names, program names, URLs, and to introduce new terms.

`Constant width`

Used for command-line output and code examples, as well as for code elements that appear in the text, including methods, functions, classes, and modules.

Constant width italic

Used to show text to be replaced with user-supplied values in code examples and commands.

Constant width bold

Used for commands to be typed at a system command line and to indicate code output in Python interpreter session examples. Also used for Python keywords.

TIP

This element signifies a tip or suggestion.

NOTE

This element signifies a general note.

WARNING

This element indicates a warning or caution.

Using Code Examples

This book is here to help you get your job done. In general, if example code is offered with this book, you may use it in your programs and documentation. You do not need to contact us for permission unless you're reproducing a significant portion of the code. For example, writing a program that uses several chunks of code from this book does not require permission. Selling or distributing examples from O'Reilly books does require permission. Answering a question by citing this book and quoting example code does not require permission. Incorporating a significant amount of example code from this book into your product's documentation does require permission.

We appreciate, but do not require, attribution. An attribution usually includes the title, author, publisher, and ISBN. For example: “*Python in a Nutshell*, 4th ed., by Alex Martelli, Anna Martelli Ravenscroft, Steve Holden, and Paul McGuire. Copyright 2023, 978-1-098-11355-1.”

If you feel your use of code examples falls outside fair use or the permission given above, feel free to contact us at permissions@oreilly.com.

O'Reilly Online Learning

NOTE

For more than 40 years, [O'Reilly Media](#) has provided technology and business training, knowledge, and insight to help companies succeed.

Our unique network of experts and innovators share their knowledge and expertise through books, articles, and our online learning platform. O'Reilly's online learning platform gives you on-demand access to live training courses, in-depth learning paths, interactive coding environments, and a vast collection of text and video from O'Reilly and 200+ other publishers. For more information, visit <https://oreilly-com.ezproxy.sfpl.org>.

How to Contact Us

Please address comments and questions concerning this book to the publisher:

- O'Reilly Media, Inc.
- 1005 Gravenstein Highway North
- Sebastopol, CA 95472
- 800-998-9938 (in the United States or Canada)
- 707-829-0515 (international or local)
- 707-829-0104 (fax)

The book has its own GitHub repository, where we list errata, examples, and any additional information. The repository also contains the full content of Chapters [24](#) and [25](#), for which there was insufficient space in the printed volume. You will find it at

<https://github.com/pynutshell/pynut4>.

We have tested and verified the information in this book to the best of our ability, but you may find that features have changed (or even that we have made mistakes!). Please let the publisher know about any errors you find, as well as your suggestions for future editions.

O'Reilly has a web page for this book, where they list errata, examples, and any additional information. You can access this page at

<https://oreil.ly/python-nutshell-4e>.

To comment or ask technical questions about this book, send email to

pynut4@gmail.com.

For news and information about our books and courses, visit

<https://oreilly-com.ezproxy.sfpl.org>.

Find us on LinkedIn: <https://linkedin.com/company/oreilly-media>.

Follow us on Twitter: <https://twitter.com/oreillymedia>.

Acknowledgments

Many thanks to O'Reilly editors and staff Amanda Quinn, Brian Guerin, Zan McQuade, and Kristen Brown. Special thanks to our editor, Angela Rufino, who did the heavy lifting to keep us on track to get this book completed on time! Also, thanks to our excellent copyeditor Rachel Head for helping us to seem more erudite than we are, and to our production editor, Christopher Faucher, for helping ensure the book looked its best in both printed and electronic formats.

Thanks to our hardworking tech reviewers, David Mertz, Mark Summerfield, and Pankaj Gaijar, who read through every explanation and example in the book's draft. Without them, this book would not have been as accurate.³ All errors that remain are entirely our own.

Thanks also to Luciano Ramalho, the whole PyPy team, Sebastián Ramírez, Fabio Pliger, Miguel Grinberg, and the Python Packaging Authority team for their help on selected portions of the book, and to Google for its useful Workspace online collaboration tools, without which our intense communication (and coordination among authors on different continents!) would have been far less easy and efficient.

Last but by no means least, the authors and all readers of this book owe a huge debt of thanks to the core developers of the Python language itself, without whose heroic efforts there would be no need for this book.

- ¹ The separate chapter on asynchronous programming in the third edition has been dropped in this edition, deferring to more thorough coverage of this growing topic in references found in [Chapter 15](#).
- ² For example, to accommodate the widespread changes in Python 3.9 and 3.10 in type annotations, most of [Chapter 5](#) uses Python 3.10 as the base version for features and examples.

3 Nor would it have so many footnotes!