

Ch 14 The Python Standard Library

- [The Python Language Reference](#) describes the exact syntax and semantics of the Python language
- [Python Standard Library Reference Manual](#) describes the standard library that is distributed with Python
- [Python's standard library is very extensive](#), offering a wide range of facilities as indicated by the long table of contents
- The Python installers for the Windows platform usually [include the entire standard library](#) and often also include many additional components.
- In addition to the standard library, [there is a growing collection of several thousand components](#)
 - from individual programs and modules to packages and entire application development frameworks,
 - available from [the Python Package Index \(PyPI\)](#)

Table of Contents [1/17]

- 1. Introduction
- 2. Built-in Functions
- 3. Built-in Constants
 - 3.1. Constants added by the site module
- 4. Built-in Types
 - 4.1. Truth Value Testing
 - 4.2. Boolean Operations — and, or, not
 - 4.3. Comparisons
 - 4.4. Numeric Types — int, float, complex
 - 4.5. Iterator Types
 - 4.6. Sequence Types — list, tuple, range
 - 4.7. Text Sequence Type — str
 - 4.8. Binary Sequence Types — bytes, bytearray, memoryview
 - 4.9. Set Types — set, frozenset
 - 4.10. Mapping Types — dict
 - 4.11. Context Manager Types
 - 4.12. Other Built-in Types
 - 4.13. Special Attributes
- 5. Built-in Exceptions
 - 5.1. Base classes
 - 5.2. Concrete exceptions
 - 5.3. Warnings
 - 5.4. Exception hierarchy

Table of Contents [2/17]

- 6. Text Processing Services
 - 6.1. string — Common string operations
 - 6.2. re — Regular expression operations
 - 6.3. difflib — Helpers for computing deltas
 - 6.4. textwrap — Text wrapping and filling
 - 6.5. unicodedata — Unicode Database
 - 6.6. stringprep — Internet String Preparation
 - 6.7. readline — GNU readline interface
 - 6.8. rlcompleter — Completion function for GNU readline
- 7. Binary Data Services
 - 7.1. struct — Interpret bytes as packed binary data
 - 7.2. codecs — Codec registry and base classes

Table of Contents [3/17]

- 8. Data Types
 - 8.1. datetime — Basic date and time types
 - 8.2. calendar — General calendar-related functions
 - 8.3. collections — Container datatypes
 - 8.4. `collections.abc` — Abstract Base Classes for Containers
 - 8.5. heapq — Heap queue algorithm
 - 8.6. bisect — Array bisection algorithm
 - 8.7. array — Efficient arrays of numeric values
 - 8.8. `weakref` — Weak references
 - 8.9. `types` — Dynamic type creation and names for built-in types
 - 8.10. copy — Shallow and deep copy operations
 - 8.11. `pprint` — Data pretty printer
 - 8.12. `reprlib` — Alternate `repr()` implementation
 - 8.13. `enum` — Support for enumerations

Table of Contents [4/17]

- 9. Numeric and Mathematical Modules
 - 9.1. `numbers` — Numeric abstract base classes
 - 9.2. `math` — Mathematical functions
 - 9.3. `cmath` — Mathematical functions for complex numbers
 - 9.4. `decimal` — Decimal fixed point and floating point arithmetic
 - 9.5. `fractions` — Rational numbers
 - 9.6. `random` — Generate pseudo-random numbers
 - 9.7. `statistics` — Mathematical statistics functions
- 10. Functional Programming Modules
 - 10.1. `itertools` — Functions creating iterators for efficient looping
 - 10.2. `functools` — Higher-order functions and operations on callable objects
 - 10.3. `operator` — Standard operators as functions

Table of Contents [5/17]

- 11. File and Directory Access
 - 11.1. `pathlib` — Object-oriented filesystem paths
 - 11.2. `os.path` — Common pathname manipulations
 - 11.3. `fileinput` — Iterate over lines from multiple input streams
 - 11.4. `stat` — Interpreting `stat()` results
 - 11.5. `filecmp` — File and Directory Comparisons
 - 11.6. `tempfile` — Generate temporary files and directories
 - 11.7. `glob` — Unix style pathname pattern expansion
 - 11.8. `fnmatch` — Unix filename pattern matching
 - 11.9. `linecache` — Random access to text lines
 - 11.10. `shutil` — High-level file operations
 - 11.11. `macpath` — Mac OS 9 path manipulation functions
- 12. Data Persistence
 - 12.1. `pickle` — Python object serialization
 - 12.2. `copyreg` — Register pickle support functions
 - 12.3. `shelve` — Python object persistence
 - 12.4. `marshal` — Internal Python object serialization
 - 12.5. `dbm` — Interfaces to Unix “databases”
 - 12.6. `sqlite3` — DB-API 2.0 interface for SQLite databases

Table of Contents [6/17]

- 13. Data Compression and Archiving
 - 13.1. `zlib` — Compression compatible with **gzip**
 - 13.2. `gzip` — Support for **gzip** files
 - 13.3. `bz2` — Support for **bzip2** compression
 - 13.4. `lzma` — Compression using the LZMA algorithm
 - 13.5. `zipfile` — Work with **ZIP** archives
 - 13.6. `tarfile` — Read and write tar archive files
- 14. File Formats
 - 14.1. `csv` — CSV File Reading and Writing
 - 14.2. `configparser` — Configuration file parser
 - 14.3. `netrc` — `netrc` file processing
 - 14.4. `xdrlib` — Encode and decode XDR data
 - 14.5. `plistlib` — Generate and parse Mac OS X `.plist` files

Table of Contents [7/17]

- 15. Cryptographic Services
 - 15.1. hashlib — Secure hashes and message digests
 - 15.2. hashlib — BLAKE2 hash functions
 - 15.3. Module
 - 15.4. Examples
 - 15.5. Credits
 - 15.6. hmac — Keyed-Hashing for Message Authentication
 - 15.7. secrets — Generate secure random numbers for managing secrets
- 16. Generic Operating System Services
 - 16.1. os — Miscellaneous operating system interfaces
 - 16.2. io — Core tools for working with streams
 - 16.3. time — Time access and conversions
 - 16.4. argparse — Parser for command-line options, arguments and sub-commands
 - 16.5. getopt — C-style parser for command line options
 - 16.6. logging — Logging facility for Python
 - 16.7. logging.config — Logging configuration
 - 16.8. logging.handlers — Logging handlers
 - 16.9. getpass — Portable password input
 - 16.10. curses — Terminal handling for character-cell displays
 - 16.11. curses.textpad — Text input widget for curses programs
 - 16.12. curses.ascii — Utilities for ASCII characters
 - 16.13. curses.panel — A panel stack extension for curses
 - 16.14. platform — Access to underlying platform's identifying data
 - 16.15. errno — Standard errno system symbols
 - 16.16. ctypes — A foreign function library for Python

Table of Contents [8/17]

- 17. Concurrent Execution
 - 17.1. threading — Thread-based parallelism
 - 17.2. multiprocessing — Process-based parallelism
 - 17.3. The concurrent package
 - 17.4. concurrent.futures — Launching parallel tasks
 - 17.5. subprocess — Subprocess management
 - 17.6. sched — Event scheduler
 - 17.7. queue — A synchronized queue class
 - 17.8. dummy_threading — Drop-in replacement for the threading module
 - 17.9. _thread — Low-level threading API
- 18. Interprocess Communication and Networking
 - 18.1. socket — Low-level networking interface
 - 18.2. ssl — TLS/SSL wrapper for socket objects
 - 18.3. select — Waiting for I/O completion
 - 18.4. selectors — High-level I/O multiplexing
 - 18.5. asyncio — Asynchronous I/O, event loop, coroutines and tasks
 - 18.6. asyncore — Asynchronous socket handler
 - 18.7. asynchat — Asynchronous socket command/response handler
 - 18.8. signal — Set handlers for asynchronous events
 - 18.9. mmap — Memory-mapped file support

Table of Contents [9/17]

- 19. Internet Data Handling
 - 19.1. email — An email and MIME handling package
 - 19.2. json — JSON encoder and decoder
 - 19.3. mailcap — Mailcap file handling
 - 19.4. mailbox — Manipulate mailboxes in various formats
 - 19.5. mimetypes — Map filenames to MIME types
 - 19.6. base64 — Base16, Base32, Base64, Base85 Data Encodings
 - 19.7. binhex — Encode and decode binhex4 files
 - 19.8. binascii — Convert between binary and ASCII
 - 19.9. quopri — Encode and decode MIME quoted-printable data
 - 19.10. uu — Encode and decode uuencode files
- 20. Structured Markup Processing Tools
 - 20.1. html — HyperText Markup Language support
 - 20.2. html.parser — Simple HTML and XHTML parser
 - 20.3. html.entities — Definitions of HTML general entities
 - 20.4. XML Processing Modules
 - 20.5. xml.etree.ElementTree — The ElementTree XML API
 - 20.6. xml.dom — The Document Object Model API
 - 20.7. xml.dom.minidom — Minimal DOM implementation
 - 20.8. xml.dom.pulldom — Support for building partial DOM trees
 - 20.9. xml.sax — Support for SAX2 parsers
 - 20.10. xml.sax.handler — Base classes for SAX handlers
 - 20.11. xml.sax.saxutils — SAX Utilities
 - 20.12. xml.sax.xmlreader — Interface for XML parsers
 - 20.13. xml.parsers.expat — Fast XML parsing using Expat

Table of Contents [10/17]

- 21. Internet Protocols and Support
 - 21.1. webbrowser — Convenient Web-browser controller
 - 21.2. `cgi` — Common Gateway Interface support
 - 21.3. `cgitb` — Traceback manager for CGI scripts
 - 21.4. `wsgiref` — WSGI Utilities and Reference Implementation
 - 21.5. `urllib` — URL handling modules
 - 21.6. `urllib.request` — Extensible library for opening URLs
 - 21.7. `urllib.response` — Response classes used by `urllib`
 - 21.8. `urllib.parse` — Parse URLs into components
 - 21.9. `urllib.error` — Exception classes raised by `urllib.request`
 - 21.10. `urllib.robotparser` — Parser for `robots.txt`
 - 21.11. `http` — HTTP modules
 - 21.12. `http.client` — HTTP protocol client
 - 21.13. `ftplib` — FTP protocol client
 - 21.14. `poplib` — POP3 protocol client
 - 21.15. `imaplib` — IMAP4 protocol client
 - 21.16. `nntplib` — NNTP protocol client
 - 21.17. `smtplib` — SMTP protocol client
 - 21.18. `smtpd` — SMTP Server
 - 21.19. `telnetlib` — Telnet client
 - 21.20. `uuid` — UUID objects according to RFC 4122
 - 21.21. `socketserver` — A framework for network servers
 - 21.22. `http.server` — HTTP servers
 - 21.23. `http.cookies` — HTTP state management
 - 21.24. `http.cookiejar` — Cookie handling for HTTP clients
 - 21.25. `xmlrpc` — XMLRPC server and client modules
 - 21.26. `xmlrpc.client` — XML-RPC client access
 - 21.27. `xmlrpc.server` — Basic XML-RPC servers
 - 21.28. `ipaddress` — IPv4/IPv6 manipulation library

Table of Contents [11/17]

- 22. Multimedia Services
 - 22.1. `audioop` — Manipulate raw audio data
 - 22.2. `aifc` — Read and write AIFF and AIFC files
 - 22.3. `sunau` — Read and write Sun AU files
 - 22.4. `wave` — Read and write WAV files
 - 22.5. `chunk` — Read IFF chunked data
 - 22.6. `colorsys` — Conversions between color systems
 - 22.7. `imghdr` — Determine the type of an image
 - 22.8. `sndhdr` — Determine type of sound file
 - 22.9. `ossaudiodev` — Access to OSS-compatible audio devices
- 23. Internationalization
 - 23.1. `gettext` — Multilingual internationalization services
 - 23.2. `locale` — Internationalization services
- 24. Program Frameworks
 - 24.1. `turtle` — Turtle graphics
 - 24.2. `cmd` — Support for line-oriented command interpreters
 - 24.3. `shlex` — Simple lexical analysis

Table of Contents [12/17]

- 25. Graphical User Interfaces with Tk
 - 25.1. tkinter — Python interface to Tcl/Tk
 - 25.2. `tkinter.ttk` — Tk themed widgets
 - 25.3. `tkinter.tix` — Extension widgets for Tk
 - 25.4. `tkinter.scrolledtext` — Scrolled Text Widget
 - 25.5. IDLE
 - 25.6. Other Graphical User Interface Packages
- 26. Development Tools
 - 26.1. `typing` — Support for type hints
 - 26.2. `pydoc` — Documentation generator and online help system
 - 26.3. `doctest` — Test interactive Python examples
 - 26.4. `unittest` — Unit testing framework
 - 26.5. `unittest.mock` — mock object library
 - 26.6. `unittest.mock` — getting started
 - 26.7. 2to3 - Automated Python 2 to 3 code translation
 - 26.8. `test` — Regression tests package for Python
 - 26.9. `test.support` — Utilities for the Python test suite

Table of Contents [13/17]

- 27. Debugging and Profiling
 - 27.1. `bdb` — Debugger framework
 - 27.2. `faulthandler` — Dump the Python traceback
 - 27.3. `pdb` — The Python Debugger
 - 27.4. The Python Profilers
 - 27.5. `timeit` — Measure execution time of small code snippets
 - 27.6. `trace` — Trace or track Python statement execution
 - 27.7. `tracemalloc` — Trace memory allocations
- 28. Software Packaging and Distribution
 - 28.1. `distutils` — Building and installing Python modules
 - 28.2. `ensurepip` — Bootstrapping the `pip` installer
 - 28.3. `venv` — Creation of virtual environments
 - 28.4. `zipapp` — Manage executable python zip archives

Table of Contents [14/17]

- 29. Python Runtime Services
 - 29.1. sys — System-specific parameters and functions
 - 29.2. `sysconfig` — Provide access to Python's configuration information
 - 29.3. `builtins` — Built-in objects
 - 29.4. `__main__` — Top-level script environment
 - 29.5. `warnings` — Warning control
 - 29.6. `contextlib` — Utilities for with-statement contexts
 - 29.7. `abc` — Abstract Base Classes
 - 29.8. `atexit` — Exit handlers
 - 29.9. `traceback` — Print or retrieve a stack traceback
 - 29.10. `__future__` — Future statement definitions
 - 29.11. `gc` — Garbage Collector interface
 - 29.12. `inspect` — Inspect live objects
 - 29.13. `site` — Site-specific configuration hook
 - 29.14. `fpectl` — Floating point exception control
- 30. Custom Python Interpreters
 - 30.1. `code` — Interpreter base classes
 - 30.2. `codeop` — Compile Python code

Table of Contents [15/17]

- 31. Importing Modules
 - 31.1. `zipimport` — Import modules from Zip archives
 - 31.2. `pkgutil` — Package extension utility
 - 31.3. `modulefinder` — Find modules used by a script
 - 31.4. `runpy` — Locating and executing Python modules
 - 31.5. `importlib` — The implementation of `import`
- 32. Python Language Services
 - 32.1. `parser` — Access Python parse trees
 - 32.2. `ast` — Abstract Syntax Trees
 - 32.3. `symtable` — Access to the compiler's symbol tables
 - 32.4. `symbol` — Constants used with Python parse trees
 - 32.5. `token` — Constants used with Python parse trees
 - 32.6. `keyword` — Testing for Python keywords
 - 32.7. `tokenize` — Tokenizer for Python source
 - 32.8. `tabnanny` — Detection of ambiguous indentation
 - 32.9. `pyclbr` — Python class browser support
 - 32.10. `py_compile` — Compile Python source files
 - 32.11. `compileall` — Byte-compile Python libraries
 - 32.12. `dis` — Disassembler for Python bytecode
 - 32.13. `pickletools` — Tools for pickle developers

Table of Contents [16/17]

- 33. Miscellaneous Services
 - 33.1. `formatter` — Generic output formatting
- 34. MS Windows Specific Services
 - 34.1. `msilib` — Read and write Microsoft Installer files
 - 34.2. `msvcrt` — Useful routines from the MS VC++ runtime
 - 34.3. `winreg` — Windows registry access
- 35. Unix Specific Services
 - 35.1. `posix` — The most common POSIX system calls
 - 35.2. `pwd` — The password database
 - 35.3. `spwd` — The shadow password database
 - 35.4. `grp` — The group database
 - 35.5. `crypt` — Function to check Unix passwords
 - 35.6. `termios` — POSIX style tty control
 - 35.7. `tty` — Terminal control functions
 - 35.8. `pty` — Pseudo-terminal utilities
 - 35.9. `fcntl` — The `fcntl` and `ioctl` system calls
 - 35.10. `pipes` — Interface to shell pipelines
 - 35.11. `resource` — Resource usage information
 - 35.12. `nis` — Interface to Sun's NIS (Yellow Pages)
 - 35.13. `syslog` — Unix syslog library routines

Table of Contents [17/17]

- 36. Superseded Modules
 - 36.1. `optparse` — Parser for command line options
 - 36.2. `imp` — Access the `import` internals
- 37. Undocumented Modules
 - 37.1. Platform specific modules