**Navigation**

* [index](mk:@MSITStore:C:\Python35\Doc\python352.chm::/genindex.html)
* [modules](mk:@MSITStore:C:\Python35\Doc\python352.chm::/py-modindex.html) |
* [next](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/tarfile.html) |
* [previous](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/lzma.html) |
* [Python](https://www.python.org/) »
* [3.5.2 Documentation](mk:@MSITStore:C:\Python35\Doc\python352.chm::/index.html) »
* [The Python Standard Library](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/index.html) »
* [13. Data Compression and Archiving](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/archiving.html) »

**13.5.** [**zipfile**](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#module-zipfile) **— Work with ZIP archives**

**Source code:** [Lib/zipfile.py](https://hg.python.org/cpython/file/3.5/Lib/zipfile.py)

The ZIP file format is a common archive and compression standard. This module provides tools to create, read, write, append, and list a ZIP file. Any advanced use of this module will require an understanding of the format, as defined in [PKZIP Application Note](https://pkware.cachefly.net/webdocs/casestudies/APPNOTE.TXT).ZIP文件格式是常见的文档和压缩标准，这个模块提供用于创建、读取、写入、添加和列示一个ZIP文件的工具。任何对本模块的高级使用都需要理解这种文档的格式，其定义在[PKZIP Application Note](https://pkware.cachefly.net/webdocs/casestudies/APPNOTE.TXT)

This module does not currently handle multi-disk ZIP files. It can handle ZIP files that use the ZIP64 extensions (that is ZIP files that are more than 4 GiB in size). It supports decryption of encrypted files in ZIP archives, but it currently cannot create an encrypted file. Decryption is extremely slow as it is implemented in native Python rather than C.这个模块目前不能够处理多磁盘ZIP文件，当压缩文件超过4GB 可以使用ZIP64扩展库。它支持解压压缩文件，但目前不能够创建压缩文件。压缩文件是非常慢的因为它是用Python原生实现的而不是用C。

The module defines the following items:

这个模块定义了以下项目：

*exception* zipfile.BadZipFile

The error raised for bad ZIP files.由坏的ZIP文件引起的异常

New in version 3.2.

*exception* zipfile.BadZipfile

Alias of [BadZipFile](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.BadZipFile), for compatibility with older Python versions.是BadZipFile的别名，为了维护老版本python的兼容性

Deprecated since version 3.2.

*exception* zipfile.LargeZipFile

The error raised when a ZIP file would require ZIP64 functionality but that has not been enabled.大文件异常

*class* zipfile.ZipFile

The class for reading and writing ZIP files. See section [ZipFile Objects](mk:@MSITStore:C:\\Python35\\Doc\\python352.chm::/library/zipfile.html" \l "zipfile-objects) for constructor details.

用于读写ZIP的类。预知详情，见[ZipFile Objects](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile-objects)部分

*class* zipfile.PyZipFile

Class for creating ZIP archives containing Python libraries.

用于创建包含Python库的ZIP文档。

*class* zipfile.ZipInfo(*filename='NoName'*, *date\_time=(1980*, *1*, *1*, *0*, *0*, *0)*)

Class used to represent information about a member of an archive. Instances of this class are returned by the [getinfo()](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZipFile.getinfo) and [infolist()](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZipFile.infolist) methods of [ZipFile](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZipFile) objects. Most users of the [zipfile](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#module-zipfile) module will not need to create these, but only use those created by this module. *filename* should be the full name of the archive member, and *date\_time* should be a tuple containing six fields which describe the time of the last modification to the file; the fields are described in section [ZipInfo Objects](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipinfo-objects).

此类用于表示压缩文档的一个成员的信息

zipfile.is\_zipfile(*filename*)

Returns True if *filename* is a valid ZIP file based on its magic number, otherwise returns False. *filename* may be a file or file-like object too.

判断一个文件名filename 是否是一个合法的ZIP文件，基于其魔法数值，文件名需要是一个文件或类文件对象。

Changed in version 3.1: Support for file and file-like objects.

zipfile.ZIP\_STORED

The numeric constant for an uncompressed archive member.

未压缩文件数量的数值常量

zipfile.ZIP\_DEFLATED

The numeric constant for the usual ZIP compression method. This requires the [zlib](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zlib.html#module-zlib) module.

数值常量对应于常用的压缩方法，这需要zlib方法。

zipfile.ZIP\_BZIP2

The numeric constant for the BZIP2 compression method. This requires the [bz2](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/bz2.html#module-bz2) module.

New in version 3.3.

zipfile.ZIP\_LZMA

The numeric constant for the LZMA compression method. This requires the [lzma](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/lzma.html#module-lzma) module.

New in version 3.3.

Note

The ZIP file format specification has included support for bzip2 compression since 2001, and for LZMA compression since 2006. However, some tools (including older Python releases) do not support these compression methods, and may either refuse to process the ZIP file altogether, or fail to extract individual files.

See also

[PKZIP Application Note](https://pkware.cachefly.net/webdocs/casestudies/APPNOTE.TXT)

Documentation on the ZIP file format by Phil Katz, the creator of the format and algorithms used.

[Info-ZIP Home Page](http://www.info-zip.org/)

Information about the Info-ZIP project’s ZIP archive programs and development libraries.

**13.5.1. ZipFile Objects**

*class* zipfile.ZipFile(*file*, *mode='r'*, *compression=ZIP\_STORED*, *allowZip64=True*)

Open a ZIP file, where *file* can be either a path to a file (a string) or a file-like object. The *mode* parameter should be 'r' to read an existing file, 'w' to truncate and write a new file, 'a' to append to an existing file, or 'x' to exclusively create and write a new file. If *mode* is 'x' and *file* refers to an existing file, a [FileExistsError](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/exceptions.html#FileExistsError) will be raised. If *mode* is 'a' and *file* refers to an existing ZIP file, then additional files are added to it. If *file* does not refer to a ZIP file, then a new ZIP archive is appended to the file. This is meant for adding a ZIP archive to another file (such as python.exe). If *mode* is a and the file does not exist at all, it is created. If *mode* is r or a, the file should be seekable. *compression* is the ZIP compression method to use when writing the archive, and should be [ZIP\_STORED](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZIP_STORED), [ZIP\_DEFLATED](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZIP_DEFLATED), [ZIP\_BZIP2](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZIP_BZIP2) or [ZIP\_LZMA](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZIP_LZMA); unrecognized values will cause [RuntimeError](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/exceptions.html#RuntimeError) to be raised. If [ZIP\_DEFLATED](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZIP_DEFLATED), [ZIP\_BZIP2](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZIP_BZIP2) or [ZIP\_LZMA](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZIP_LZMA) is specified but the corresponding module ([zlib](mk:@MSITStore:C:\\Python35\\Doc\\python352.chm::/library/zlib.html" \l "module-zlib" \o "zlib: Low-level interface to compression and decompression routines compatible with gzip.), [bz2](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/bz2.html#module-bz2) or [lzma](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/lzma.html#module-lzma)) is not available, [RuntimeError](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/exceptions.html#RuntimeError) is also raised. The default is [ZIP\_STORED](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZIP_STORED). If *allowZip64* is True (the default) zipfile will create ZIP files that use the ZIP64 extensions when the zipfile is larger than 2 GiB. If it is false [zipfile](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#module-zipfile) will raise an exception when the ZIP file would require ZIP64 extensions.

If the file is created with mode 'w', 'x' or 'a' and then [closed](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZipFile.close) without adding any files to the archive, the appropriate ZIP structures for an empty archive will be written to the file.

ZipFile is also a context manager and therefore supports the [with](mk:@MSITStore:C:\Python35\Doc\python352.chm::/reference/compound_stmts.html#with) statement. In the example, *myzip* is closed after the [with](mk:@MSITStore:C:\Python35\Doc\python352.chm::/reference/compound_stmts.html#with) statement’s suite is finished—even if an exception occurs:

with ZipFile('spam.zip', 'w') as myzip:

myzip.write('eggs.txt')

New in version 3.2: Added the ability to use [ZipFile](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZipFile) as a context manager.

Changed in version 3.3: Added support for [bzip2](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/bz2.html#module-bz2) and [lzma](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/lzma.html#module-lzma) compression.

Changed in version 3.4: ZIP64 extensions are enabled by default.

Changed in version 3.5: Added support for writing to unseekable streams. Added support for the 'x' mode.

ZipFile.close()

Close the archive file. You must call [close()](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZipFile.close) before exiting your program or essential records will not be written.调用close方法关闭文档文件，退出程序前必须调用close方法，否则重要的记录将不会被写入

ZipFile.getinfo(*name*)

Return a [ZipInfo](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZipInfo) object with information about the archive member *name*. Calling [getinfo()](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZipFile.getinfo) for a name not currently contained in the archive will raise a [KeyError](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/exceptions.html#KeyError).返回一个ZipInfo对象，包含文档成员name的信息，如果name不在当前文档中，会引发一个KeyError。

ZipFile.infolist()

Return a list containing a [ZipInfo](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZipInfo) object for each member of the archive. The objects are in the same order as their entries in the actual ZIP file on disk if an existing archive was opened.返回一个列表，包含文档每个成员的ZipInfo 。

ZipFile.namelist()

Return a list of archive members by name.

返回一个包含文档成员名的列表

ZipFile.open(*name*, *mode='r'*, *pwd=None*)

Extract a member from the archive as a file-like object (ZipExtFile). *name* is the name of the file in the archive, or a [ZipInfo](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZipInfo) object. The *mode* parameter, if included, must be one of the following: 'r' (the default), 'U', or 'rU'. Choosing 'U' or 'rU' will enable [universal newlines](mk:@MSITStore:C:\Python35\Doc\python352.chm::/glossary.html#term-universal-newlines) support in the read-only object. *pwd* is the password used for encrypted files. Calling [open()](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZipFile.open) on a closed ZipFile will raise a [RuntimeError](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/exceptions.html#RuntimeError).

[open()](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZipFile.open) is also a context manager and therefore supports the [with](mk:@MSITStore:C:\Python35\Doc\python352.chm::/reference/compound_stmts.html#with) statement:

open（）也是个上下文管理器，因此也支持with语法。

with ZipFile('spam.zip') as myzip:

with myzip.open('eggs.txt') as myfile:

print(myfile.read())

Note

The file-like object is read-only and provides the following methods: [read()](mk:@MSITStore:C:\\Python35\\Doc\\python352.chm::/library/io.html" \l "io.BufferedIOBase.read" \o "io.BufferedIOBase.read), [readline()](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/io.html#io.IOBase.readline), [readlines()](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/io.html#io.IOBase.readlines), [\_\_iter\_\_()](mk:@MSITStore:C:\Python35\Doc\python352.chm::/reference/datamodel.html#object.__iter__), [\_\_next\_\_()](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/stdtypes.html#iterator.__next__).

类文件对象是只读的，而且提供以下方法：[read()](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/io.html#io.BufferedIOBase.read), [readline()](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/io.html#io.IOBase.readline), [readlines()](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/io.html#io.IOBase.readlines), [\_\_iter\_\_()](mk:@MSITStore:C:\Python35\Doc\python352.chm::/reference/datamodel.html#object.__iter__), [\_\_next\_\_()](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/stdtypes.html#iterator.__next__).

Note

Objects returned by [open()](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZipFile.open) can operate independently of the ZipFile.

Note

The [open()](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZipFile.open), [read()](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZipFile.read) and [extract()](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZipFile.extract) methods can take a filename or a [ZipInfo](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZipInfo) object. You will appreciate this when trying to read a ZIP file that contains members with duplicate names.

Deprecated since version 3.4, will be removed in version 3.6: The 'U' or 'rU' mode. Use [io.TextIOWrapper](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/io.html#io.TextIOWrapper) for reading compressed text files in [universal newlines](mk:@MSITStore:C:\Python35\Doc\python352.chm::/glossary.html#term-universal-newlines) mode.

ZipFile.extract(*member*, *path=None*, *pwd=None*)

Extract a member from the archive to the current working directory; *member* must be its full name or a [ZipInfo](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZipInfo) object). Its file information is extracted as accurately as possible. *path* specifies a different directory to extract to. *member* can be a filename or a [ZipInfo](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZipInfo) object. *pwd* is the password used for encrypted files.

Returns the normalized path created (a directory or new file).

Note

If a member filename is an absolute path, a drive/UNC sharepoint and leading (back)slashes will be stripped, e.g.: ///foo/bar becomes foo/bar on Unix, and C:\foo\bar becomes foo\bar on Windows. And all ".." components in a member filename will be removed, e.g.: ../../foo../../ba..r becomes foo../ba..r. On Windows illegal characters (:, <, >, |, ", ?, and \*) replaced by underscore (\_).

ZipFile.extractall(*path=None*, *members=None*, *pwd=None*)

Extract all members from the archive to the current working directory. *path* specifies a different directory to extract to. *members* is optional and must be a subset of the list returned by [namelist()](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZipFile.namelist). *pwd* is the password used for encrypted files.

Warning

Never extract archives from untrusted sources without prior inspection. It is possible that files are created outside of *path*, e.g. members that have absolute filenames starting with "/" or filenames with two dots "..". This module attempts to prevent that. See [extract()](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZipFile.extract) note.

ZipFile.printdir()

Print a table of contents for the archive to sys.stdout.

在标准输出中打印一个关于文档内容的表

ZipFile.setpassword(*pwd*)

Set *pwd* as default password to extract encrypted files.

将pwd设置为提取压缩文件的密码

ZipFile.read(*name*, *pwd=None*)

Return the bytes of the file *name* in the archive. *name* is the name of the file in the archive, or a [ZipInfo](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZipInfo) object. The archive must be open for read or append. *pwd* is the password used for encrypted files and, if specified, it will override the default password set with [setpassword()](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZipFile.setpassword). Calling [read()](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZipFile.read) on a closed ZipFile will raise a [RuntimeError](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/exceptions.html#RuntimeError). Calling [read()](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZipFile.read) on a ZipFile that uses a compression method other than [ZIP\_STORED](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZIP_STORED), [ZIP\_DEFLATED](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZIP_DEFLATED), [ZIP\_BZIP2](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZIP_BZIP2) or [ZIP\_LZMA](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZIP_LZMA) will raise a [NotImplementedError](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/exceptions.html#NotImplementedError). An error will also be raised if the corresponding compression module is not available.

ZipFile.testzip()

Read all the files in the archive and check their CRC’s and file headers. Return the name of the first bad file, or else return None. Calling [testzip()](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZipFile.testzip) on a closed ZipFile will raise a [RuntimeError](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/exceptions.html#RuntimeError).

ZipFile.write(*filename*, *arcname=None*, *compress\_type=None*)

Write the file named *filename* to the archive, giving it the archive name *arcname* (by default, this will be the same as *filename*, but without a drive letter and with leading path separators removed). If given, *compress\_type* overrides the value given for the *compression* parameter to the constructor for the new entry. The archive must be open with mode 'w', 'x' or 'a' – calling [write()](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZipFile.write) on a ZipFile created with mode 'r' will raise a [RuntimeError](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/exceptions.html#RuntimeError). Calling [write()](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZipFile.write) on a closed ZipFile will raise a [RuntimeError](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/exceptions.html#RuntimeError).

Note

There is no official file name encoding for ZIP files. If you have unicode file names, you must convert them to byte strings in your desired encoding before passing them to [write()](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZipFile.write). WinZip interprets all file names as encoded in CP437, also known as DOS Latin.

Note

Archive names should be relative to the archive root, that is, they should not start with a path separator.

Note

If arcname (or filename, if arcname is not given) contains a null byte, the name of the file in the archive will be truncated at the null byte.

ZipFile.writestr(*zinfo\_or\_arcname*, *bytes*[, *compress\_type*])

Write the string *bytes* to the archive; *zinfo\_or\_arcname* is either the file name it will be given in the archive, or a [ZipInfo](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZipInfo) instance. If it’s an instance, at least the filename, date, and time must be given. If it’s a name, the date and time is set to the current date and time. The archive must be opened with mode 'w', 'x' or 'a' – calling [writestr()](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZipFile.writestr) on a ZipFile created with mode 'r' will raise a [RuntimeError](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/exceptions.html#RuntimeError). Calling [writestr()](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZipFile.writestr) on a closed ZipFile will raise a [RuntimeError](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/exceptions.html#RuntimeError).

If given, *compress\_type* overrides the value given for the *compression* parameter to the constructor for the new entry, or in the *zinfo\_or\_arcname* (if that is a [ZipInfo](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZipInfo) instance).

Note

When passing a [ZipInfo](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZipInfo) instance as the *zinfo\_or\_arcname* parameter, the compression method used will be that specified in the *compress\_type* member of the given [ZipInfo](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZipInfo) instance. By default, the [ZipInfo](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZipInfo) constructor sets this member to [ZIP\_STORED](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZIP_STORED).

Changed in version 3.2: The *compress\_type* argument.

The following data attributes are also available:

ZipFile.debug

The level of debug output to use. This may be set from 0 (the default, no output) to 3 (the most output). Debugging information is written to sys.stdout.

ZipFile.comment

The comment text associated with the ZIP file. If assigning a comment to a [ZipFile](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZipFile) instance created with mode 'w', 'x' or 'a', this should be a string no longer than 65535 bytes. Comments longer than this will be truncated in the written archive when [close()](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZipFile.close) is called.

**13.5.2. PyZipFile Objects**

The [PyZipFile](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.PyZipFile) constructor takes the same parameters as the [ZipFile](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZipFile) constructor, and one additional parameter, *optimize*.

*class* zipfile.PyZipFile(*file*, *mode='r'*, *compression=ZIP\_STORED*, *allowZip64=True*, *optimize=-1*)

New in version 3.2: The *optimize* parameter.

Changed in version 3.4: ZIP64 extensions are enabled by default.

Instances have one method in addition to those of [ZipFile](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZipFile) objects:

writepy(*pathname*, *basename=''*, *filterfunc=None*)

Search for files \*.py and add the corresponding file to the archive.

If the *optimize* parameter to [PyZipFile](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.PyZipFile) was not given or -1, the corresponding file is a \*.pyc file, compiling if necessary.

If the *optimize* parameter to [PyZipFile](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.PyZipFile) was 0, 1 or 2, only files with that optimization level (see [compile()](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/functions.html#compile)) are added to the archive, compiling if necessary.

If *pathname* is a file, the filename must end with .py, and just the (corresponding \*.py[co]) file is added at the top level (no path information). If *pathname* is a file that does not end with .py, a [RuntimeError](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/exceptions.html#RuntimeError) will be raised. If it is a directory, and the directory is not a package directory, then all the files \*.py[co] are added at the top level. If the directory is a package directory, then all \*.py[co] are added under the package name as a file path, and if any subdirectories are package directories, all of these are added recursively.

*basename* is intended for internal use only.

*filterfunc*, if given, must be a function taking a single string argument. It will be passed each path (including each individual full file path) before it is added to the archive. If *filterfunc* returns a false value, the path will not be added, and if it is a directory its contents will be ignored. For example, if our test files are all either in test directories or start with the string test\_, we can use a *filterfunc* to exclude them:

>>> zf = PyZipFile('myprog.zip')

>>> def notests(s):

... fn = os.path.basename(s)

... return (not (fn == 'test' or fn.startswith('test\_')))

>>> zf.writepy('myprog', filterfunc=notests)

The [writepy()](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.PyZipFile.writepy) method makes archives with file names like this:

string.pyc # Top level name

test/\_\_init\_\_.pyc # Package directory

test/testall.pyc # Module test.testall

test/bogus/\_\_init\_\_.pyc # Subpackage directory

test/bogus/myfile.pyc # Submodule test.bogus.myfile

New in version 3.4: The *filterfunc* parameter.

**13.5.3. ZipInfo Objects**

Instances of the [ZipInfo](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZipInfo) class are returned by the [getinfo()](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZipFile.getinfo) and [infolist()](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZipFile.infolist) methods of [ZipFile](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/zipfile.html#zipfile.ZipFile) objects. Each object stores information about a single member of the ZIP archive.

Instances have the following attributes:

ZipInfo.filename

Name of the file in the archive.

ZipInfo.date\_time

The time and date of the last modification to the archive member. This is a tuple of six values:

| **Index** | **Value** |
| --- | --- |
| 0 | Year (>= 1980) |
| 1 | Month (one-based) |
| 2 | Day of month (one-based) |
| 3 | Hours (zero-based) |
| 4 | Minutes (zero-based) |
| 5 | Seconds (zero-based) |

Note

The ZIP file format does not support timestamps before 1980.

ZipInfo.compress\_type

Type of compression for the archive member.

ZipInfo.comment

Comment for the individual archive member.

ZipInfo.extra

Expansion field data. The [PKZIP Application Note](https://pkware.cachefly.net/webdocs/casestudies/APPNOTE.TXT) contains some comments on the internal structure of the data contained in this string.

ZipInfo.create\_system

System which created ZIP archive.

ZipInfo.create\_version

PKZIP version which created ZIP archive.

ZipInfo.extract\_version

PKZIP version needed to extract archive.

ZipInfo.reserved

Must be zero.

ZipInfo.flag\_bits

ZIP flag bits.

ZipInfo.volume

Volume number of file header.

ZipInfo.internal\_attr

Internal attributes.

ZipInfo.external\_attr

External file attributes.

ZipInfo.header\_offset

Byte offset to the file header.

ZipInfo.CRC

CRC-32 of the uncompressed file.

ZipInfo.compress\_size

Size of the compressed data.

ZipInfo.file\_size

Size of the uncompressed file.

**Navigation**

* [index](mk:@MSITStore:C:\Python35\Doc\python352.chm::/genindex.html)
* [modules](mk:@MSITStore:C:\Python35\Doc\python352.chm::/py-modindex.html) |
* [next](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/tarfile.html) |
* [previous](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/lzma.html) |
* [Python](https://www.python.org/) »
* [3.5.2 Documentation](mk:@MSITStore:C:\Python35\Doc\python352.chm::/index.html) »
* [The Python Standard Library](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/index.html) »
* [13. Data Compression and Archiving](mk:@MSITStore:C:\Python35\Doc\python352.chm::/library/archiving.html) »

© [Copyright](mk:@MSITStore:C:\Python35\Doc\python352.chm::/copyright.html) 2001-2016, Python Software Foundation.   
The Python Software Foundation is a non-profit corporation. [Please donate.](https://www.python.org/psf/donations/)   
Last updated on Jun 25, 2016. [Found a bug](mk:@MSITStore:C:\Python35\Doc\python352.chm::/bugs.html)?   
Created using [Sphinx](http://sphinx.pocoo.org/) 1.3.1.