

:mod:`zipfile` --- Work with ZIP archives

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 1); [backlink](#)

Unknown interpreted text role "mod".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 4)

Unknown directive type "module".

```
.. module:: zipfile
   :synopsis: Read and write ZIP-format archive files.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 7)

Unknown directive type "moduleauthor".

```
.. moduleauthor:: James C. Ahlstrom <jim@interet.com>
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 8)

Unknown directive type "sectionauthor".

```
.. sectionauthor:: James C. Ahlstrom <jim@interet.com>
```

Source code: `:source:`Lib/zipfile.py``

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 10); [backlink](#)

Unknown interpreted text role "source".

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](#).

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.

The module defines the following items:

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 28)

Unknown directive type "exception".

```
.. exception:: BadZipFile
   The error raised for bad ZIP files.

.. versionadded:: 3.2
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 35)

Unknown directive type "exception".

```
.. exception:: BadZipfile
   Alias of :exc:`BadZipFile`, for compatibility with older Python versions.
```

```
.. deprecated:: 3.2
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 42)

Unknown directive type "exception".

```
.. exception:: LargeZipFile
```

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

The class for reading and writing ZIP files. See section [ref`zipfile-objects`](#) for constructor details.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 51); [backlink](#)

Unknown interpreted text role "ref".

A pathlib-compatible wrapper for zip files. See section [ref`path-objects`](#) for details.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 58); [backlink](#)

Unknown interpreted text role "ref".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 61)

Unknown directive type "versionadded".

```
.. versionadded:: 3.8
```

Class for creating ZIP archives containing Python libraries.

Class used to represent information about a member of an archive. Instances of this class are returned by the [meth` .getinfo`](#) and [meth` .infolist`](#) methods of [class` ZipFile`](#) objects. Most users of the [mod` 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 [ref`zipinfo-objects`](#).

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 72); [backlink](#)

Unknown interpreted text role "meth".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 72); [backlink](#)

Unknown interpreted text role "meth".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 72); [backlink](#)

Unknown interpreted text role "class".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 72); [backlink](#)

Unknown interpreted text role "mod".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 72); [backlink](#)

Unknown interpreted text role "ref".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 81)

Unknown directive type "function".

```
.. function:: 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.

```
.. versionchanged:: 3.1
    Support for file and file-like objects.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 90)

Unknown directive type "data".

```
.. data:: ZIP_STORED
```

The numeric constant for an uncompressed archive member.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 95)

Unknown directive type "data".

```
.. data:: ZIP_DEFLATED
```

The numeric constant for the usual ZIP compression method. This requires the :mod:`zlib` module.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 101)

Unknown directive type "data".

```
.. data:: ZIP_BZIP2
```

The numeric constant for the BZIP2 compression method. This requires the :mod:`bz2` module.

```
.. versionadded:: 3.3
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 108)

Unknown directive type "data".

```
.. data:: ZIP_LZMA
```

The numeric constant for the LZMA compression method. This requires the :mod:`lzma` module.

```
.. versionadded:: 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.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 124)

Unknown directive type "seealso".

```
.. seealso::
```

```
`PKZIP Application Note`_
    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.
```

ZipFile Objects

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 141)

Invalid class attribute value for "class" directive: "ZipFile(file, mode='r', compression=ZIP_STORED, allowZip64=True, \compresslevel=None, *, strict_timestamps=True, metadata_encoding=None)".

```
.. class:: ZipFile(file, mode='r', compression=ZIP_STORED, allowZip64=True, \
               compresslevel=None, *, strict_timestamps=True,
               metadata_encoding=None)
```

Open a ZIP file, where *file* can be a path to a file (a string), a file-like object or a `:term:`path-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 `:exc:`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 `:file:`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 `:const:`ZIP_STORED``, `:const:`ZIP_DEFLATED``, `:const:`ZIP_BZIP2`` or `:const:`ZIP_LZMA``; unrecognized values will cause `:exc:`NotImplementedError`` to be raised. If `:const:`ZIP_DEFLATED``, `:const:`ZIP_BZIP2`` or `:const:`ZIP_LZMA`` is specified but the corresponding module (`:mod:`zlib``, `:mod:`bz2`` or `:mod:`lzma``) is not available, `:exc:`RuntimeError`` is raised. The default is `:const:`ZIP_STORED``.

If *allowZip64* is `True` (the default) zipfile will create ZIP files that use the ZIP64 extensions when the zipfile is larger than 4 GiB. If it is `false` `:mod:`zipfile`` will raise an exception when the ZIP file would require ZIP64 extensions.

The *compresslevel* parameter controls the compression level to use when writing files to the archive. When using `:const:`ZIP_STORED`` or `:const:`ZIP_LZMA`` it has no effect. When using `:const:`ZIP_DEFLATED`` integers `'0'` through `'9'` are accepted (see `:class:`zlib` <zlib.compressobj>` for more information). When using `:const:`ZIP_BZIP2`` integers `'1'` through `'9'` are accepted (see `:class:`bz2` <bz2.BZ2File>` for more information).

The *strict_timestamps* argument, when set to `False`, allows to zip files older than 1980-01-01 at the cost of setting the timestamp to 1980-01-01. Similar behavior occurs with files newer than 2107-12-31, the timestamp is also set to the limit.

When mode is `'r'`, *metadata_encoding* may be set to the name of a codec, which will be used to decode metadata such as the names of members and ZIP comments.

If the file is created with mode `'w'`, `'x'` or `'a'` and then `:meth:`closed` <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 :keyword:`with` statement. In the example, *myzip* is closed after the :keyword:`!with` statement's suite is finished---even if an exception occurs::

```
with ZipFile('spam.zip', 'w') as myzip:
    myzip.write('eggs.txt')
```

.. note::

metadata_encoding is an instance-wide setting for the ZipFile. It is not currently possible to set this on a per-member basis.

This attribute is a workaround for legacy implementations which produce archives with names in the current locale encoding or code page (mostly on Windows). According to the .ZIP standard, the encoding of metadata may be specified to be either IBM code page (default) or UTF-8 by a flag in the archive header. That flag takes precedence over *metadata_encoding*, which is a Python-specific extension.

.. versionadded:: 3.2

Added the ability to use :class:`ZipFile` as a context manager.

.. versionchanged:: 3.3

Added support for :mod:`bz2` <bz2>` and :mod:`lzma` compression.

.. versionchanged:: 3.4

ZIP64 extensions are enabled by default.

.. versionchanged:: 3.5

Added support for writing to unseekable streams.
Added support for the ``'x'`` mode.

.. versionchanged:: 3.6

Previously, a plain :exc:`RuntimeError` was raised for unrecognized compression values.

.. versionchanged:: 3.6.2

The *file* parameter accepts a :term:`path-like object`.

.. versionchanged:: 3.7

Add the *compresslevel* parameter.

.. versionadded:: 3.8

The *strict_timestamps* keyword-only argument

.. versionchanged:: 3.11

Added support for specifying member name encoding for reading metadata in the zipfile's directory and file headers.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 246)

Unknown directive type "method".

```
.. method:: ZipFile.close()
```

Close the archive file. You must call :meth:`close` before exiting your program or essential records will not be written.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 252)

Unknown directive type "method".

```
.. method:: ZipFile.getinfo(name)
```

Return a :class:`ZipInfo` object with information about the archive member *name*. Calling :meth:`getinfo` for a name not currently contained in the archive will raise a :exc:`KeyError`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 259)

Unknown directive type "method".

```
.. method:: ZipFile.infolist()
```

Return a list containing a :class:`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.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 266)

Unknown directive type "method".

```
.. method:: ZipFile.namelist()
```

Return a list of archive members by name.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 271)

Unknown directive type "method".

```
.. method:: ZipFile.open(name, mode='r', pwd=None, *, force_zip64=False)
```

Access a member of the archive as a binary file-like object. *name* can be either the name of a file within the archive or a :class:`ZipInfo` object. The *mode* parameter, if included, must be ``'r'`` (the default) or ``'w'``. *pwd* is the password used to decrypt encrypted ZIP files.

:meth:`~ZipFile.open` is also a context manager and therefore supports the :keyword:`with` statement::

```
with ZipFile('spam.zip') as myzip:
    with myzip.open('eggs.txt') as myfile:
        print(myfile.read())
```

With *mode* ``'r'`` the file-like object (:class:`ZipExtFile`) is read-only and provides the following methods: :meth:`~io.BufferedIOBase.read`, :meth:`~io.IOBase.readline`, :meth:`~io.IOBase.readlines`, :meth:`~io.IOBase.seek`, :meth:`~io.IOBase.tell`, :meth:`~__iter__`, :meth:`~iterator.__next__`. These objects can operate independently of the ZipFile.

With *mode* ``'w'``, a writable file handle is returned, which supports the :meth:`~io.BufferedIOBase.write` method. While a writable file handle is open, attempting to read or write other files in the ZIP file will raise a :exc:`ValueError`.

When writing a file, if the file size is not known in advance but may exceed 2 GiB, pass ``force_zip64=True`` to ensure that the header format is capable of supporting large files. If the file size is known in advance, construct a :class:`ZipInfo` object with :attr:`~ZipInfo.file_size` set, and use that as the *name* parameter.

.. note::

The :meth:`~open`, :meth:`~read` and :meth:`~extract` methods can take a filename or a :class:`ZipInfo` object. You will appreciate this when trying to read a ZIP file that contains members with duplicate names.

.. versionchanged:: 3.6
Removed support of ``mode='U'``. Use :class:`io.TextIOWrapper` for reading compressed text files in :term:`universal newlines` mode.

.. versionchanged:: 3.6
:meth:`~open` can now be used to write files into the archive with the ``mode='w'`` option.

.. versionchanged:: 3.6
Calling :meth:`~open` on a closed ZipFile will raise a :exc:`ValueError`. Previously, a :exc:`RuntimeError` was raised.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 322)

Unknown directive type "method".

```
.. method:: 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 `:class:`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 `:class:`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 (```_```).

```
.. versionchanged:: 3.6
```

Calling `:meth:`extract`` on a closed `ZipFile` will raise a `:exc:`ValueError``. Previously, a `:exc:`RuntimeError`` was raised.

```
.. versionchanged:: 3.6.2
```

The **path** parameter accepts a `:term:`path-like object``.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 350)

Unknown directive type "method".

```
.. method:: 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 `:meth:`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 `:meth:`extract`` note.

```
.. versionchanged:: 3.6
```

Calling `:meth:`extractall`` on a closed `ZipFile` will raise a `:exc:`ValueError``. Previously, a `:exc:`RuntimeError`` was raised.

```
.. versionchanged:: 3.6.2
```

The **path** parameter accepts a `:term:`path-like object``.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 373)

Unknown directive type "method".

```
.. method:: ZipFile.printdir()
```

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

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-

main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 378)

Unknown directive type "method".

```
.. method:: ZipFile.setpassword(pwd)

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

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 383)

Unknown directive type "method".

```
.. method:: 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 :class:`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 :meth:`setpassword`. Calling
:meth:`read` on a ZipFile that uses a compression method other than
:const:`ZIP_STORED`, :const:`ZIP_DEFLATED`, :const:`ZIP_BZIP2` or
:const:`ZIP_LZMA` will raise a :exc:`NotImplementedError`. An error will also
be raised if the corresponding compression module is not available.

.. versionchanged:: 3.6
    Calling :meth:`read` on a closed ZipFile will raise a :exc:`ValueError`.
    Previously, a :exc:`RuntimeError` was raised.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 399)

Unknown directive type "method".

```
.. method:: 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``.

.. versionchanged:: 3.6
    Calling :meth:`testzip` on a closed ZipFile will raise a
    :exc:`ValueError`. Previously, a :exc:`RuntimeError` was raised.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 409)

Unknown directive type "method".

```
.. method:: ZipFile.write(filename, arcname=None, compress_type=None, \
                           compresslevel=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. Similarly, *compresslevel* will override the constructor if
given.
The archive must be open with mode ``'w'``, ``'x'`` or ``'a'``.

.. note::

    The ZIP file standard historically did not specify a metadata encoding,
    but strongly recommended CP437 (the original IBM PC encoding) for
    interoperability. Recent versions allow use of UTF-8 (only). In this
    module, UTF-8 will automatically be used to write the member names if
    they contain any non-ASCII characters. It is not possible to write
    member names in any encoding other than ASCII or UTF-8.

.. 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.

.. note::

A leading slash in the filename may lead to the archive being impossible to open in some zip programs on Windows systems.

.. versionchanged:: 3.6

Calling :meth:`write` on a ZipFile created with mode ``'r'`` or a closed ZipFile will raise a :exc:`ValueError`. Previously, a :exc:`RuntimeError` was raised.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 450)

Unknown directive type "method".

.. method:: ZipFile.writestr(zinfo_or_arcname, data, compress_type=None, \compresslevel=None)

Write a file into the archive. The contents is **data**, which may be either a :class:`str` or a :class:`bytes` instance; if it is a :class:`str`, it is encoded as UTF-8 first. **zinfo_or_arcname** is either the file name it will be given in the archive, or a :class:`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'``.

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 :class:`ZipInfo` instance). Similarly, **compresslevel** will override the constructor if given.

.. note::

When passing a :class:`ZipInfo` instance as the **zinfo_or_arcname** parameter, the compression method used will be that specified in the **compress_type** member of the given :class:`ZipInfo` instance. By default, the :class:`ZipInfo` constructor sets this member to :const:`ZIP_STORED`.

.. versionchanged:: 3.2

The **compress_type** argument.

.. versionchanged:: 3.6

Calling :meth:`writestr` on a ZipFile created with mode ``'r'`` or a closed ZipFile will raise a :exc:`ValueError`. Previously, a :exc:`RuntimeError` was raised.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 481)

Unknown directive type "method".

.. method:: ZipFile.mkdir(zinfo_or_directory, mode=511)

Create a directory inside the archive. If **zinfo_or_directory** is a string, a directory is created inside the archive with the mode that is specified in the **mode** argument. If, however, **zinfo_or_directory** is a :class:`ZipInfo` instance then the **mode** argument is ignored.

The archive must be opened with mode ``'w'``, ``'x'`` or ``'a'``.

.. versionadded:: 3.11

The following data attributes are also available:

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 495)

Unknown directive type "attribute".

```
.. attribute:: ZipFile.filename
```

Name of the ZIP file.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 499)

Unknown directive type "attribute".

```
.. attribute:: 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``.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 505)

Unknown directive type "attribute".

```
.. attribute:: ZipFile.comment
```

The comment associated with the ZIP file as a :class:`bytes` object. If assigning a comment to a :class:`ZipFile` instance created with mode ``'w'``, ``'x'`` or ``'a'``, it should be no longer than 65535 bytes. Comments longer than this will be truncated.

Path Objects

Construct a Path object from a `root` zipfile (which may be a :class:`ZipFile` instance or file suitable for passing to the :class:`ZipFile` constructor).

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 521); [backlink](#)

Unknown interpreted text role "class".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 521); [backlink](#)

Unknown interpreted text role "class".

`at` specifies the location of this Path within the zipfile, e.g. 'dir/file.txt', 'dir/', or ''. Defaults to the empty string, indicating the root.

Path objects expose the following features of :mod:`pathlib.Path` objects:

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 529); [backlink](#)

Unknown interpreted text role "mod".

Path objects are traversable using the `/` operator or `joinpath`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 534)

Unknown directive type "attribute".

```
.. attribute:: Path.name
```

The final path component.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 538)

Unknown directive type "method".

```
.. method:: Path.open(mode='r', *, pwd, **)  
  
Invoke :meth:`ZipFile.open` on the current path.  
Allows opening for read or write, text or binary  
through supported modes: 'r', 'w', 'rb', 'wb'.  
Positional and keyword arguments are passed through to  
:class:`io.TextIOWrapper` when opened as text and  
ignored otherwise.  
``pwd`` is the ``pwd`` parameter to  
:meth:`ZipFile.open`.  
  
.. versionchanged:: 3.9  
   Added support for text and binary modes for open. Default  
   mode is now text.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 553)

Unknown directive type "method".

```
.. method:: Path.iterdir()  
  
Enumerate the children of the current directory.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 557)

Unknown directive type "method".

```
.. method:: Path.is_dir()  
  
Return ``True`` if the current context references a directory.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 561)

Unknown directive type "method".

```
.. method:: Path.is_file()  
  
Return ``True`` if the current context references a file.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 565)

Unknown directive type "method".

```
.. method:: Path.exists()  
  
Return ``True`` if the current context references a file or  
directory in the zip file.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 570)

Unknown directive type "data".

```
.. data:: Path.suffix  
  
The file extension of the final component.  
  
.. versionadded:: 3.11  
   Added :data:`Path.suffix` property.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 577)

Unknown directive type "data".

```
.. data:: Path.stem
```

The final path component, without its suffix.

```
.. versionadded:: 3.11
   Added :data:`Path.stem` property.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 584)

Unknown directive type "data".

```
.. data:: Path.suffixes
```

A list of the path's file extensions.

```
.. versionadded:: 3.11
   Added :data:`Path.suffixes` property.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 591)

Unknown directive type "method".

```
.. method:: Path.read_text(*, **)
```

Read the current file as unicode text. Positional and keyword arguments are passed through to :class:`io.TextIOWrapper` (except ``buffer``, which is implied by the context).

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 598)

Unknown directive type "method".

```
.. method:: Path.read_bytes()
```

Read the current file as bytes.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 602)

Unknown directive type "method".

```
.. method:: Path.joinpath(*other)
```

Return a new Path object with each of the *other* arguments joined. The following are equivalent::

```
>>> Path(...).joinpath('child').joinpath('grandchild')
>>> Path(...).joinpath('child', 'grandchild')
>>> Path(...) / 'child' / 'grandchild'
```

```
.. versionchanged:: 3.10
   Prior to 3.10, ``joinpath`` was undocumented and accepted
   exactly one parameter.
```

The [zipp](#) project provides backports of the latest path object functionality to older Pythons. Use `zipp.Path` in place of `zipfile.Path` for early access to changes.

PyZipFile Objects

The :class:`PyZipFile` constructor takes the same parameters as the :class:`ZipFile` constructor, and one additional parameter, *optimize*.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-

main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 625); [backlink](#)

Unknown interpreted text role "class".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 625); [backlink](#)

Unknown interpreted text role "class".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 628)

Invalid class attribute value for "class" directive: "PyZipFile(file, mode='r', compression=ZIP_STORED, allowZip64=True, \optimize=-1)".

```
.. class:: PyZipFile(file, mode='r', compression=ZIP_STORED, allowZip64=True, \
    optimize=-1)

.. versionadded:: 3.2
    The *optimize* parameter.

.. versionchanged:: 3.4
    ZIP64 extensions are enabled by default.

Instances have one method in addition to those of :class:`ZipFile` objects:

.. method:: PyZipFile.writepy(pathname, basename='', filterfunc=None)

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

    If the *optimize* parameter to :class:`PyZipFile` was not given or ``-1``,
    the corresponding file is a :file:`*.pyc` file, compiling if necessary.

    If the *optimize* parameter to :class:`PyZipFile` was ``0``, ``1`` or
    ``2``, only files with that optimization level (see :func:`compile`) are
    added to the archive, compiling if necessary.

    If *pathname* is a file, the filename must end with :file:`.py`, and
    just the (corresponding :file:`*.pyc`) file is added at the top level
    (no path information). If *pathname* is a file that does not end with
    :file:`.py`, a :exc:`RuntimeError` will be raised. If it is a directory,
    and the directory is not a package directory, then all the files
    :file:`*.pyc` are added at the top level. If the directory is a
    package directory, then all :file:`*.pyc` are added under the package
    name as a file path, and if any subdirectories are package directories,
    all of these are added recursively in sorted order.

    *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 :meth:`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

.. versionadded:: 3.4
    The *filterfunc* parameter.

.. versionchanged:: 3.6.2
    The *pathname* parameter accepts a :term:`path-like object`.
```

```
.. versionchanged:: 3.7
    Recursion sorts directory entries.
```

ZipInfo Objects

Instances of the `class:ZipInfo` class are returned by the `meth:.getinfo` and `meth:.infolist` methods of `class:ZipFile` objects. Each object stores information about a single member of the ZIP archive.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 701); [backlink](#)

Unknown interpreted text role "class".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 701); [backlink](#)

Unknown interpreted text role "meth".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 701); [backlink](#)

Unknown interpreted text role "meth".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 701); [backlink](#)

Unknown interpreted text role "class".

There is one classmethod to make a `class:ZipInfo` instance for a filesystem file:

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 705); [backlink](#)

Unknown interpreted text role "class".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 708)

Unknown directive type "classmethod".

```
.. classmethod:: ZipInfo.from_file(filename, arcname=None, *, \
                                strict_timestamps=True)
```

Construct a `class:ZipInfo` instance for a file on the filesystem, in preparation for adding it to a zip file.

`*filename*` should be the path to a file or directory on the filesystem.

If `*arcname*` is specified, it is used as the name within the archive.

If `*arcname*` is not specified, the name will be the same as `*filename*`, but with any drive letter and leading path separators removed.

The `*strict_timestamps*` argument, when set to `False`, allows to zip files older than 1980-01-01 at the cost of setting the timestamp to 1980-01-01.

Similar behavior occurs with files newer than 2107-12-31, the timestamp is also set to the limit.

```
.. versionadded:: 3.6
```

```
.. versionchanged:: 3.6.2
```

The `*filename*` parameter accepts a `term:path-like object`.

```
.. versionadded:: 3.8
```

The `*strict_timestamps*` keyword-only argument

Instances have the following methods and attributes:

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 737)

Unknown directive type "method".

```
.. method:: ZipInfo.is_dir()

Return ``True`` if this archive member is a directory.

This uses the entry's name: directories should always end with ``/``.

.. versionadded:: 3.6
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 746)

Unknown directive type "attribute".

```
.. attribute:: ZipInfo.filename

Name of the file in the archive.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 751)

Unknown directive type "attribute".

```
.. attribute:: 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.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 777)

Unknown directive type "attribute".

```
.. attribute:: ZipInfo.compress_type

Type of compression for the archive member.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 782)

Unknown directive type "attribute".

```
.. attribute:: ZipInfo.comment
```

Comment for the individual archive member as a :class:`bytes` object.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 787)

Unknown directive type "attribute".

```
.. attribute:: ZipInfo.extra
```

Expansion field data. The `PKZIP Application Note`_ contains some comments on the internal structure of the data contained in this :class:`bytes` object.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 794)

Unknown directive type "attribute".

```
.. attribute:: ZipInfo.create_system
```

System which created ZIP archive.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 799)

Unknown directive type "attribute".

```
.. attribute:: ZipInfo.create_version
```

PKZIP version which created ZIP archive.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 804)

Unknown directive type "attribute".

```
.. attribute:: ZipInfo.extract_version
```

PKZIP version needed to extract archive.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 809)

Unknown directive type "attribute".

```
.. attribute:: ZipInfo.reserved
```

Must be zero.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 814)

Unknown directive type "attribute".

```
.. attribute:: ZipInfo.flag_bits
```

ZIP flag bits.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 819)

Unknown directive type "attribute".

```
.. attribute:: ZipInfo.volume  
  
    Volume number of file header.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 824)

Unknown directive type "attribute".

```
.. attribute:: ZipInfo.internal_attr  
  
    Internal attributes.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 829)

Unknown directive type "attribute".

```
.. attribute:: ZipInfo.external_attr  
  
    External file attributes.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 834)

Unknown directive type "attribute".

```
.. attribute:: ZipInfo.header_offset  
  
    Byte offset to the file header.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 839)

Unknown directive type "attribute".

```
.. attribute:: ZipInfo.CRC  
  
    CRC-32 of the uncompressed file.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 844)

Unknown directive type "attribute".

```
.. attribute:: ZipInfo.compress_size  
  
    Size of the compressed data.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 849)

Unknown directive type "attribute".

```
.. attribute:: ZipInfo.file_size  
  
    Size of the uncompressed file.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 855)

Unknown directive type "program".

```
.. program:: zipfile
```

Command-Line Interface

The `mod:zipfile` module provides a simple command-line interface to interact with ZIP archives.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 860); [backlink](#)

Unknown interpreted text role "mod".

If you want to create a new ZIP archive, specify its name after the `option:-c` option and then list the filename(s) that should be included:

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 863); [backlink](#)

Unknown interpreted text role "option".

```
$ python -m zipfile -c monty.zip spam.txt eggs.txt
```

Passing a directory is also acceptable:

```
$ python -m zipfile -c monty.zip life-of-brian_1979/
```

If you want to extract a ZIP archive into the specified directory, use the `option:-e` option:

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 876); [backlink](#)

Unknown interpreted text role "option".

```
$ python -m zipfile -e monty.zip target-dir/
```

For a list of the files in a ZIP archive, use the `option:-l` option:

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 883); [backlink](#)

Unknown interpreted text role "option".

```
$ python -m zipfile -l monty.zip
```

Command-line options

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 893)

Unknown directive type "cmdoption".

```
.. cmdoption:: -l <zipfile>
               --list <zipfile>
```

List files in a zipfile.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 898)

Unknown directive type "cmdoption".

```
.. cmdoption:: -c <zipfile> <source1> ... <sourceN>
```

```
--create <zipfile> <source1> ... <sourceN>

Create zipfile from source files.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 903)

Unknown directive type "cmdoption".

```
.. cmdoption:: -e <zipfile> <output_dir>
               --extract <zipfile> <output_dir>
```

Extract zipfile into target directory.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 908)

Unknown directive type "cmdoption".

```
.. cmdoption:: -t <zipfile>
               --test <zipfile>
```

Test whether the zipfile is valid or not.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library) zipfile.rst, line 913)

Unknown directive type "cmdoption".

```
.. cmdoption:: --metadata-encoding <encoding>
```

Specify encoding of member names for :option:`-l`, :option:`-e` and :option:`-t`.

```
.. versionadded:: 3.11
```

Decompression pitfalls

The extraction in zipfile module might fail due to some pitfalls listed below.

From file itself

Decompression may fail due to incorrect password / CRC checksum / ZIP format or unsupported compression method / decryption.

File System limitations

Exceeding limitations on different file systems can cause decompression failed. Such as allowable characters in the directory entries, length of the file name, length of the pathname, size of a single file, and number of files, etc.

Resources limitations

The lack of memory or disk volume would lead to decompression failed. For example, decompression bombs (aka [ZIP bomb](#)) apply to zipfile library that can cause disk volume exhaustion.

Interruption

Interruption during the decompression, such as pressing control-C or killing the decompression process may result in incomplete decompression of the archive.

Default behaviors of extraction

Not knowing the default extraction behaviors can cause unexpected decompression results. For example, when extracting the same archive twice, it overwrites files without asking.