

:mod:`bz2` --- Support for :program:`bzip2` compression

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]bz2.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]bz2.rst, line 1); [backlink](#)

Unknown interpreted text role "program".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]bz2.rst, line 4)

Unknown directive type "module".

```
.. module:: bz2
:synopsis: Interfaces for bzip2 compression and decompression.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]bz2.rst, line 7)

Unknown directive type "moduleauthor".

```
.. moduleauthor:: Gustavo Niemeyer <niemeyer@conectiva.com>
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]bz2.rst, line 8)

Unknown directive type "moduleauthor".

```
.. moduleauthor:: Nadeem Vawda <nadeem.vawda@gmail.com>
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]bz2.rst, line 9)

Unknown directive type "sectionauthor".

```
.. sectionauthor:: Gustavo Niemeyer <niemeyer@conectiva.com>
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]bz2.rst, line 10)

Unknown directive type "sectionauthor".

```
.. sectionauthor:: Nadeem Vawda <nadeem.vawda@gmail.com>
```

Source code: :source:`Lib/bz2.py`

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]bz2.rst, line 12); [backlink](#)

Unknown interpreted text role "source".

This module provides a comprehensive interface for compressing and decompressing data using the bzip2 compression algorithm. The :mod:`bz2` module contains:

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]bz2.rst, line 19); [backlink](#)

Unknown interpreted text role "mod".

- The `.func:'.open'` function and `:class:'BZ2File'` class for reading and writing compressed files.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]bz2.rst, line 21); [backlink](#)

Unknown interpreted text role "func".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]bz2.rst, line 21); [backlink](#)

Unknown interpreted text role "class".

- The `:class:'BZ2Compressor'` and `:class:'BZ2Decompressor'` classes for incremental (de)compression.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]bz2.rst, line 23); [backlink](#)

Unknown interpreted text role "class".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]bz2.rst, line 23); [backlink](#)

Unknown interpreted text role "class".

- The `.func:'compress'` and `.func:'decompress'` functions for one-shot (de)compression.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]bz2.rst, line 25); [backlink](#)

Unknown interpreted text role "func".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]bz2.rst, line 25); [backlink](#)

Unknown interpreted text role "func".

(De)compression of files

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]bz2.rst, line 32)

Unknown directive type "function".

```
.. function:: open(filename, mode='rb', compresslevel=9, encoding=None, errors=None, newline=None)
```

Open a bzip2-compressed file in binary or text mode, returning a `:term:'file object'`.

As with the constructor for `:class:'BZ2File'`, the `*filename*` argument can be an actual filename (a `:class:'str'` or `:class:'bytes'` object), or an existing file object to read from or write to.

The `*mode*` argument can be any of `''r''`, `''rb''`, `''w''`, `''wb''`, `''x''`, `''xb''`, `''a''` or `''ab''` for binary mode, or `''rt''`, `''wt''`, `''xt''`, or `''at''` for text mode. The default is `''rb''`.

The `*compresslevel*` argument is an integer from 1 to 9, as for the `:class:'BZ2File'` constructor.

For binary mode, this function is equivalent to the `:class:'BZ2File'` constructor: ```BZ2File(filename, mode, compresslevel=compresslevel)```. In this case, the `*encoding*`, `*errors*` and `*newline*` arguments must not be provided.

For text mode, a `:class:`BZ2File`` object is created, and wrapped in an `:class:`io.TextIOWrapper`` instance with the specified encoding, error handling behavior, and line ending(s).

```
.. versionadded:: 3.3

.. versionchanged:: 3.4
    The ``'x'`` (exclusive creation) mode was added.

.. versionchanged:: 3.6
    Accepts a :term:`path-like object`.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]bz2.rst, line 66)

Invalid class attribute value for "class" directive: "BZ2File(filename, mode='r', *, compresslevel=9)".

```
.. class:: BZ2File(filename, mode='r', *, compresslevel=9)
```

Open a bzip2-compressed file in binary mode.

If `*filename*` is a `:class:`str`` or `:class:`bytes`` object, open the named file directly. Otherwise, `*filename*` should be a `:term:`file object``, which will be used to read or write the compressed data.

The `*mode*` argument can be either ```'r'``` for reading (default), ```'w'``` for overwriting, ```'x'``` for exclusive creation, or ```'a'``` for appending. These can equivalently be given as ```'rb'```, ```'wb'```, ```'xb'``` and ```'ab'``` respectively.

If `*filename*` is a file object (rather than an actual file name), a mode of ```'w'``` does not truncate the file, and is instead equivalent to ```'a'```.

If `*mode*` is ```'w'``` or ```'a'```, `*compresslevel*` can be an integer between ```1``` and ```9``` specifying the level of compression: ```1``` produces the least compression, and ```9``` (default) produces the most compression.

If `*mode*` is ```'r'```, the input file may be the concatenation of multiple compressed streams.

`:class:`BZ2File`` provides all of the members specified by the `:class:`io.BufferedIOBase``, except for `:meth:`detach`` and `:meth:`truncate``. Iteration and the `:keyword:`with`` statement are supported.

`:class:`BZ2File`` also provides the following method:

```
.. method:: peek([n])
```

Return buffered data without advancing the file position. At least one byte of data will be returned (unless at EOF). The exact number of bytes returned is unspecified.

```
.. note:: While calling :meth:`peek` does not change the file position of the :class:`BZ2File`, it may change the position of the underlying file object (e.g. if the :class:`BZ2File` was constructed by passing a file object for *filename*).
```

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

```
.. versionchanged:: 3.1
    Support for the :keyword:`with` statement was added.
```

```
.. versionchanged:: 3.3
    The :meth:`fileno`, :meth:`readable`, :meth:`seekable`, :meth:`writable`, :meth:`readl` and :meth:`readinto` methods were added.
```

```
.. versionchanged:: 3.3
    Support was added for *filename* being a :term:`file object` instead of an actual filename.
```

```
.. versionchanged:: 3.3
    The ``'a'`` (append) mode was added, along with support for reading multi-stream files.
```

```
.. versionchanged:: 3.4
    The ``'x'`` (exclusive creation) mode was added.
```

```

.. versionchanged:: 3.5
   The :meth:`~io.BufferedIOBase.read` method now accepts an argument of
   ``None``.

.. versionchanged:: 3.6
   Accepts a :term:`path-like object`.

.. versionchanged:: 3.9
   The *buffering* parameter has been removed. It was ignored and deprecated
   since Python 3.0. Pass an open file object to control how the file is
   opened.

   The *compresslevel* parameter became keyword-only.

.. versionchanged:: 3.10
   This class is thread unsafe in the face of multiple simultaneous
   readers or writers, just like its equivalent classes in :mod:`gzip` and
   :mod:`lzma` have always been.

```

Incremental (de)compression

Create a new compressor object. This object may be used to compress data incrementally. For one-shot compression, use the `:func:`compress`` function instead.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ [cpython-main] [Doc] [library]bz2.rst, line 152); [backlink](#)

Unknown interpreted text role "func".

compresslevel, if given, must be an integer between 1 and 9. The default is 9.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ [cpython-main] [Doc] [library]bz2.rst, line 159)

Unknown directive type "method".

```
.. method:: compress(data)
```

Provide data to the compressor object. Returns a chunk of compressed data if possible, or an empty byte string otherwise.

When you have finished providing data to the compressor, call the :meth:`flush` method to finish the compression process.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ [cpython-main] [Doc] [library]bz2.rst, line 168)

Unknown directive type "method".

```
.. method:: flush()
```

Finish the compression process. Returns the compressed data left in internal buffers.

The compressor object may not be used after this method has been called.

Create a new decompressor object. This object may be used to decompress data incrementally. For one-shot compression, use the `:func:`decompress`` function instead.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ [cpython-main] [Doc] [library]bz2.rst, line 178); [backlink](#)

Unknown interpreted text role "func".

Note

This class does not transparently handle inputs containing multiple compressed streams, unlike `:func:`decompress`` and `:class:`BZ2File``. If you need to decompress a multi-stream input with `:class:`BZ2Decompressor``, you must use a new decompressor for each stream.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]bz2.rst, line 183); [backlink](#)

Unknown interpreted text role "func".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]bz2.rst, line 183); [backlink](#)

Unknown interpreted text role "class".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]bz2.rst, line 183); [backlink](#)

Unknown interpreted text role "class".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]bz2.rst, line 188)

Unknown directive type "method".

```
.. method:: decompress(data, max_length=-1)
```

Decompress **data** (a :term:`bytes-like object`), returning uncompressed data as bytes. Some of **data** may be buffered internally, for use in later calls to :meth:`decompress`. The returned data should be concatenated with the output of any previous calls to :meth:`decompress`.

If **max_length** is nonnegative, returns at most **max_length** bytes of decompressed data. If this limit is reached and further output can be produced, the :attr:`~.needs_input` attribute will be set to ``False``. In this case, the next call to :meth:`~.decompress` may provide **data** as ``b''`` to obtain more of the output.

If all of the input data was decompressed and returned (either because this was less than **max_length** bytes, or because **max_length** was negative), the :attr:`~.needs_input` attribute will be set to ``True``.

Attempting to decompress data after the end of stream is reached raises an `EOFError`. Any data found after the end of the stream is ignored and saved in the :attr:`~.unused_data` attribute.

```
.. versionchanged:: 3.5
   Added the *max_length* parameter.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]bz2.rst, line 215)

Unknown directive type "attribute".

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

``True`` if the end-of-stream marker has been reached.

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

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]bz2.rst, line 222)

Unknown directive type "attribute".

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

Data found after the end of the compressed stream.

If this attribute is accessed before the end of the stream has been reached, its value will be ``b''`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ [cpython-main] [Doc] [library]bz2.rst, line 229)

Unknown directive type "attribute".

```
.. attribute:: needs_input

    ``False`` if the :meth:`.decompress` method can provide more
    decompressed data before requiring new uncompressed input.

.. versionadded:: 3.5
```

One-shot (de)compression

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ [cpython-main] [Doc] [library]bz2.rst, line 240)

Unknown directive type "function".

```
.. function:: compress(data, compresslevel=9)

    Compress *data*, a :term:`bytes-like object` <bytes-like object>`.

    *compresslevel*, if given, must be an integer between ``1`` and ``9``. The
    default is ``9``.

    For incremental compression, use a :class:`BZ2Compressor` instead.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ [cpython-main] [Doc] [library]bz2.rst, line 250)

Unknown directive type "function".

```
.. function:: decompress(data)

    Decompress *data*, a :term:`bytes-like object` <bytes-like object>`.

    If *data* is the concatenation of multiple compressed streams, decompress
    all of the streams.

    For incremental decompression, use a :class:`BZ2Decompressor` instead.

.. versionchanged:: 3.3
    Support for multi-stream inputs was added.
```

Examples of usage

Below are some examples of typical usage of the `mod:'bz2'` module.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ [cpython-main] [Doc] [library]bz2.rst, line 267); [backlink](#)

Unknown interpreted text role "mod".

Using `func:'compress'` and `func:'decompress'` to demonstrate round-trip compression:

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ [cpython-main] [Doc] [library]bz2.rst, line 269); [backlink](#)

Unknown interpreted text role "func".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ [cpython-main] [Doc] [library]bz2.rst, line 269); [backlink](#)

Unknown interpreted text role "func".

```
>>> import bz2
>>> data = b"""\
... Donec rhoncus quis sapien sit amet molestie. Fusce scelerisque vel augue
... nec ullamcorper. Nam rutrum pretium placerat. Aliquam vel tristique lorem,
... sit amet cursus ante. In interdum laoreet mi, sit amet ultrices purus
... pulvinar a. Nam gravida euismod magna, non varius justo tincidunt feugiat.
... Aliquam pharetra lacus non risus vehicula rutrum. Maecenas aliquam leo
... felis. Pellentesque semper nunc sit amet nibh ullamcorper, ac elementum
... dolor luctus. Curabitur lacinia mi ornare consectetur vestibulum."""
>>> c = bz2.compress(data)
>>> len(data) / len(c) # Data compression ratio
1.513595166163142
>>> d = bz2.decompress(c)
>>> data == d # Check equality to original object after round-trip
True
```

Using `class:'BZ2Compressor'` for incremental compression:

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]bz2.rst, line 287); [backlink](#)

Unknown interpreted text role "class".

```
>>> import bz2
>>> def gen_data(chunks=10, chunksize=1000):
...     """Yield incremental blocks of chunksize bytes."""
...     for _ in range(chunks):
...         yield b"z" * chunksize
...
>>> comp = bz2.BZ2Compressor()
>>> out = b""
>>> for chunk in gen_data():
...     # Provide data to the compressor object
...     out = out + comp.compress(chunk)
...
>>> # Finish the compression process. Call this once you have
>>> # finished providing data to the compressor.
>>> out = out + comp.flush()
```

The example above uses a very "nonrandom" stream of data (a stream of `b"z"` chunks). Random data tends to compress poorly, while ordered, repetitive data usually yields a high compression ratio.

Writing and reading a bzip2-compressed file in binary mode:

```
>>> import bz2
>>> data = b"""\
... Donec rhoncus quis sapien sit amet molestie. Fusce scelerisque vel augue
... nec ullamcorper. Nam rutrum pretium placerat. Aliquam vel tristique lorem,
... sit amet cursus ante. In interdum laoreet mi, sit amet ultrices purus
... pulvinar a. Nam gravida euismod magna, non varius justo tincidunt feugiat.
... Aliquam pharetra lacus non risus vehicula rutrum. Maecenas aliquam leo
... felis. Pellentesque semper nunc sit amet nibh ullamcorper, ac elementum
... dolor luctus. Curabitur lacinia mi ornare consectetur vestibulum."""
>>> with bz2.open("myfile.bz2", "wb") as f:
...     # Write compressed data to file
...     unused = f.write(data)
>>> with bz2.open("myfile.bz2", "rb") as f:
...     # Decompress data from file
...     content = f.read()
>>> content == data # Check equality to original object after round-trip
True
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]bz2.rst, line 329)

Unknown directive type "testcleanup".

```
.. testcleanup::

    import os
    os.remove("myfile.bz2")
```