Python Development Mode

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]devmode.rst, line 6)

Unknown directive type "versionadded".

```
.. versionadded:: 3.7
```

The Python Development Mode introduces additional runtime checks that are too expensive to be enabled by default. It should not be more verbose than the default if the code is correct; new warnings are only emitted when an issue is detected.

It can be enabled using the :option:`-X dev <-X>` command line option or by setting the :envvar:`PYTHONDEVMODE` environment variable to `1`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]devmode.rst, line 13); *backlink*

Unknown interpreted text role "option".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]devmode.rst, line 13); *backlink*

Unknown interpreted text role "envvar".

See also :ref:`Python debug build <debug-build>`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]devmode.rst, line 16); *backlink*

Unknown interpreted text role "ref".

Effects of the Python Development Mode

Enabling the Python Development Mode is similar to the following command, but with additional effects described below:

```
PYTHONMALLOC=debug PYTHONASYNCIODEBUG=1 python3 -W default -X faulthandler
```

Effects of the Python Development Mode:

- Add `default` :ref:`warning filter <describing-warning-filters>`. The following warnings are shown:

  System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]devmode.rst, line 28); *backlink*

  Unknown interpreted text role "ref".

  - :exc:`DeprecationWarning`

    System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]devmode.rst, line 31); *backlink*

    Unknown interpreted text role "exc".

  - :exc:`ImportWarning`

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

    Unknown interpreted text role "exc".

  - :exc:`PendingDeprecationWarning`

    System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]devmode.rst, line 33); *backlink*

Unknown interpreted text role "exc".

- :exc:`ResourceWarning`

  System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]devmode.rst`, line 34); *backlink*

  Unknown interpreted text role "exc".

Normally, the above warnings are filtered by the default :ref:`warning filters <describing-warning-filters>`.

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]devmode.rst`, line 36); *backlink*

Unknown interpreted text role "ref".

It behaves as if the :option:`-W default <-W>` command line option is used.

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]devmode.rst`, line 39); *backlink*

Unknown interpreted text role "option".

Use the :option:`-W error <-W>` command line option or set the :envvar:`PYTHONWARNINGS` environment variable to `error` to treat warnings as errors.

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]devmode.rst`, line 41); *backlink*

Unknown interpreted text role "option".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]devmode.rst`, line 41); *backlink*

Unknown interpreted text role "envvar".

- Install debug hooks on memory allocators to check for:
  - Buffer underflow
  - Buffer overflow
  - Memory allocator API violation
  - Unsafe usage of the GIL

  See the :c:func:`PyMem_SetupDebugHooks` C function.

  System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]devmode.rst`, line 52); *backlink*

  Unknown interpreted text role "c:func".

  It behaves as if the :envvar:`PYTHONMALLOC` environment variable is set to `debug`.

  System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]devmode.rst`, line 54); *backlink*

  Unknown interpreted text role "envvar".

  To enable the Python Development Mode without installing debug hooks on memory allocators, set the :envvar:`PYTHONMALLOC` environment variable to `default`.

  System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]devmode.rst`, line 57);

backlink

Unknown interpreted text role "envvar".

- Call :func:`faulthandler.enable` at Python startup to install handlers for the :const:`SIGSEGV`, :const:`SIGFPE`, :const:`SIGABRT`, :const:`SIGBUS` and :const:`SIGILL` signals to dump the Python traceback on a crash.

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]devmode.rst`, line 61); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]devmode.rst`, line 61); *backlink*

Unknown interpreted text role "const".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]devmode.rst`, line 61); *backlink*

Unknown interpreted text role "const".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]devmode.rst`, line 61); *backlink*

Unknown interpreted text role "const".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]devmode.rst`, line 61); *backlink*

Unknown interpreted text role "const".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]devmode.rst`, line 61); *backlink*

Unknown interpreted text role "const".

It behaves as if the :option:`-X faulthandler <-X>` command line option is used or if the :envvar:`PYTHONFAULTHANDLER` environment variable is set to `1`.

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]devmode.rst`, line 65); *backlink*

Unknown interpreted text role "option".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]devmode.rst`, line 65); *backlink*

Unknown interpreted text role "envvar".

- Enable :ref:`asyncio debug mode <asyncio-debug-mode>`. For example, :mod:`asyncio` checks for coroutines that were not awaited and logs them.

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]devmode.rst`, line 69); *backlink*

Unknown interpreted text role "ref".

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

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]devmode.rst, line 69); backlink

Unknown interpreted text role "mod".

It behaves as if the :envvar:`PYTHONASYNCIODEBUG` environment variable is set to `1`.

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

Unknown interpreted text role "envvar".

- Check the *encoding* and *errors* arguments for string encoding and decoding operations. Examples: :func:`open`, :meth:`str.encode` and :meth:`bytes.decode`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]devmode.rst, line 75); backlink

Unknown interpreted text role "func".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]devmode.rst, line 75); backlink

Unknown interpreted text role "meth".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]devmode.rst, line 75); backlink

Unknown interpreted text role "meth".

By default, for best performance, the *errors* argument is only checked at the first encoding/decoding error and the *encoding* argument is sometimes ignored for empty strings.

- The :class:`io.IOBase` destructor logs `close()` exceptions.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]devmode.rst, line 83); backlink

Unknown interpreted text role "class".

- Set the :attr:`~sys.flags.dev_mode` attribute of :attr:`sys.flags` to `True`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]devmode.rst, line 84); backlink

Unknown interpreted text role "attr".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]devmode.rst, line 84); backlink

Unknown interpreted text role "attr".

The Python Development Mode does not enable the :mod:`tracemalloc` module by default, because the overhead cost (to performance and memory) would be too large. Enabling the :mod:`tracemalloc` module provides additional information on the origin of some errors. For example, :exc:`ResourceWarning` logs the traceback where the resource was allocated, and a buffer overflow error logs the traceback where the memory block was allocated.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]devmode.rst, line 87); backlink

Unknown interpreted text role "mod".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]devmode.rst, line 87); *backlink*

Unknown interpreted text role "mod".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]devmode.rst, line 87); *backlink*

Unknown interpreted text role "exc".

The Python Development Mode does not prevent the :option:`-O` command line option from removing :keyword:`assert` statements nor from setting :const:`__debug__` to `False`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]devmode.rst, line 94); *backlink*

Unknown interpreted text role "option".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]devmode.rst, line 94); *backlink*

Unknown interpreted text role "keyword".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]devmode.rst, line 94); *backlink*

Unknown interpreted text role "const".

The Python Development Mode can only be enabled at the Python startup. Its value can be read from :data:`sys.flags.dev_mode <sys.flags>`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]devmode.rst, line 98); *backlink*

Unknown interpreted text role "data".

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

Unknown directive type "versionchanged".

```
.. versionchanged:: 3.8
   The :class:`io.IOBase` destructor now logs ``close()`` exceptions.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]devmode.rst, line 104)

Unknown directive type "versionchanged".

```
.. versionchanged:: 3.9
   The *encoding* and *errors* arguments are now checked for string encoding
   and decoding operations.
```

ResourceWarning Example

Example of a script counting the number of lines of the text file specified in the command line:

```
import sys

def main():
    fp = open(sys.argv[1])
    nlines = len(fp.readlines())
    print(nlines)
    # The file is closed implicitly

if __name__ == "__main__":
    main()
```

The script does not close the file explicitly. By default, Python does not emit any warning. Example using README.txt, which has 269 lines:

```
$ python3 script.py README.txt
269
```

Enabling the Python Development Mode displays a :exc:`ResourceWarning` warning:

> **System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]devmode.rst, line 134); *backlink***
>
> Unknown interpreted text role "exc".

```
$ python3 -X dev script.py README.txt
269
script.py:10: ResourceWarning: unclosed file <_io.TextIOWrapper name='README.rst' mode='r' encoding='UTF-
  main()
ResourceWarning: Enable tracemalloc to get the object allocation traceback
```

In addition, enabling :mod:`tracemalloc` shows the line where the file was opened:

> **System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]devmode.rst, line 144); *backlink***
>
> Unknown interpreted text role "mod".

```
$ python3 -X dev -X tracemalloc=5 script.py README.rst
269
script.py:10: ResourceWarning: unclosed file <_io.TextIOWrapper name='README.rst' mode='r' encoding='UTF-
  main()
Object allocated at (most recent call last):
  File "script.py", lineno 10
    main()
  File "script.py", lineno 4
    fp = open(sys.argv[1])
```

The fix is to close explicitly the file. Example using a context manager:

```
def main():
    # Close the file explicitly when exiting the with block
    with open(sys.argv[1]) as fp:
        nlines = len(fp.readlines())
    print(nlines)
```

Not closing a resource explicitly can leave a resource open for way longer than expected; it can cause severe issues upon exiting Python. It is bad in CPython, but it is even worse in PyPy. Closing resources explicitly makes an application more deterministic and more reliable.

Bad file descriptor error example

Script displaying the first line of itself:

```
import os

def main():
    fp = open(__file__)
    firstline = fp.readline()
    print(firstline.rstrip())
    os.close(fp.fileno())
    # The file is closed implicitly

main()
```

By default, Python does not emit any warning:

```
$ python3 script.py
import os
```

The Python Development Mode shows a :exc:`ResourceWarning` and logs a "Bad file descriptor" error when finalizing the file object:

> **System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]devmode.rst, line 196); *backlink***
>
> Unknown interpreted text role "exc".

```
$ python3 script.py
import os
script.py:10: ResourceWarning: unclosed file <_io.TextIOWrapper name='script.py' mode='r' encoding='UTF-8
  main()
ResourceWarning: Enable tracemalloc to get the object allocation traceback
Exception ignored in: < io.TextIOWrapper name='script.py' mode='r' encoding='UTF-8'>
Traceback (most recent call last):
```

```
  File "script.py", line 10, in <module>
    main()
OSError: [Errno 9] Bad file descriptor
```

`os.close(fp.fileno())` closes the file descriptor. When the file object finalizer tries to close the file descriptor again, it fails with the `Bad file descriptor` error. A file descriptor must be closed only once. In the worst case scenario, closing it twice can lead to a crash (see :issue:`18748` for an example).

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]devmode.rst`, line 212); *backlink*
>
> Unknown interpreted text role "issue".

The fix is to remove the `os.close(fp.fileno())` line, or open the file with `closefd=False`.