:mod:`test` --- Regression tests package for Python

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

Unknown directive type "module".

```
.. module:: test
   :synopsis: Regression tests package containing the testing suite for Python.
```

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

Unknown directive type "sectionauthor".

```
.. sectionauthor:: Brett Cannon <brett@python.org>
```

Note

The :mod:`test` package is meant for internal use by Python only. It is documented for the benefit of the core developers of Python. Any use of this package outside of Python's standard library is discouraged as code mentioned here can change or be removed without notice between releases of Python.

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

Unknown interpreted text role "mod".

The :mod:`test` package contains all regression tests for Python as well as the modules :mod:`test.support` and :mod:`test.regrtest`. :mod:`test.support` is used to enhance your tests while :mod:`test.regrtest` drives the testing suite.

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst`, line 18); *backlink*

Unknown interpreted text role "mod".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst`, line 18); *backlink*

Unknown interpreted text role "mod".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst`, line 18); *backlink*

Unknown interpreted text role "mod".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst`, line 18); *backlink*

Unknown interpreted text role "mod".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst`, line 18); *backlink*

Unknown interpreted text role "mod".

Each module in the :mod:`test` package whose name starts with `test_` is a testing suite for a specific module or feature. All new tests should be written using the :mod:`unittest` or :mod:`doctest` module. Some older tests are written using a "traditional" testing style that compares output printed to `sys.stdout`; this style of test is considered deprecated.

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

Unknown interpreted text role "mod".

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

Unknown interpreted text role "mod".

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

Unknown interpreted text role "mod".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 30)

Unknown directive type "seealso".

```
.. seealso::

   Module :mod:`unittest`
      Writing PyUnit regression tests.

   Module :mod:`doctest`
      Tests embedded in documentation strings.
```

Writing Unit Tests for the :mod:`test` package

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

Unknown interpreted text role "mod".

It is preferred that tests that use the :mod:`unittest` module follow a few guidelines. One is to name the test module by starting it with `test_` and end it with the name of the module being tested. The test methods in the test module should start with `test_` and end with a description of what the method is testing. This is needed so that the methods are recognized by the test driver as test methods. Also, no documentation string for the method should be included. A comment (such as `# Tests function returns only True or False`) should be used to provide documentation for test methods. This is done because documentation strings get printed out if they exist and thus what test is being run is not stated.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 44); *backlink*

Unknown interpreted text role "mod".

A basic boilerplate is often used:

```
import unittest
from test import support

class MyTestCase1(unittest.TestCase):

    # Only use setUp() and tearDown() if necessary

    def setUp(self):
        ... code to execute in preparation for tests ...

    def tearDown(self):
        ... code to execute to clean up after tests ...

    def test_feature_one(self):
        # Test feature one.
```

```
            ... testing code ...

        def test_feature_two(self):
            # Test feature two.
            ... testing code ...

        ... more test methods ...

    class MyTestCase2(unittest.TestCase):
        ... same structure as MyTestCase1 ...

    ... more test classes ...

    if __name__ == '__main__':
        unittest.main()
```

This code pattern allows the testing suite to be run by :mod:`test.regrtest`, on its own as a script that supports the :mod:`unittest` CLI, or via the `python -m unittest` CLI.

> **System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 88); *backlink***
>
> Unknown interpreted text role "mod".

> **System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 88); *backlink***
>
> Unknown interpreted text role "mod".

The goal for regression testing is to try to break code. This leads to a few guidelines to be followed:

- The testing suite should exercise all classes, functions, and constants. This includes not just the external API that is to be presented to the outside world but also "private" code.
- Whitebox testing (examining the code being tested when the tests are being written) is preferred. Blackbox testing (testing only the published user interface) is not complete enough to make sure all boundary and edge cases are tested.
- Make sure all possible values are tested including invalid ones. This makes sure that not only all valid values are acceptable but also that improper values are handled correctly.
- Exhaust as many code paths as possible. Test where branching occurs and thus tailor input to make sure as many different paths through the code are taken.
- Add an explicit test for any bugs discovered for the tested code. This will make sure that the error does not crop up again if the code is changed in the future.
- Make sure to clean up after your tests (such as close and remove all temporary files).
- If a test is dependent on a specific condition of the operating system then verify the condition already exists before attempting the test.
- Import as few modules as possible and do it as soon as possible. This minimizes external dependencies of tests and also minimizes possible anomalous behavior from side-effects of importing a module.
- Try to maximize code reuse. On occasion, tests will vary by something as small as what type of input is used. Minimize code duplication by subclassing a basic test class with a class that specifies the input:

  ```
  class TestFuncAcceptsSequencesMixin:

      func = mySuperWhammyFunction

      def test_func(self):
          self.func(self.arg)

  class AcceptLists(TestFuncAcceptsSequencesMixin, unittest.TestCase):
      arg = [1, 2, 3]

  class AcceptStrings(TestFuncAcceptsSequencesMixin, unittest.TestCase):
      arg = 'abc'

  class AcceptTuples(TestFuncAcceptsSequencesMixin, unittest.TestCase):
      arg = (1, 2, 3)
  ```

  When using this pattern, remember that all classes that inherit from :class:`unittest.TestCase` are run as tests. The :class:`Mixin` class in the example above does not have any data and so can't be run by itself, thus it does not inherit from :class:`unittest.TestCase`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 145); *backlink*

Unknown interpreted text role "class".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 145); *backlink*

Unknown interpreted text role "class".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 145); *backlink*

Unknown interpreted text role "class".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 151)

Unknown directive type "seealso".

```
.. seealso::

   Test Driven Development
      A book by Kent Beck on writing tests before code.
```

Running tests using the command-line interface

The :mod:`test` package can be run as a script to drive Python's regression test suite, thanks to the :option:`-m` option: :program:`python -m test`. Under the hood, it uses :mod:`test.regrtest`; the call :program:`python -m test.regrtest` used in previous Python versions still works. Running the script by itself automatically starts running all regression tests in the :mod:`test` package. It does this by finding all modules in the package whose name starts with `test_`, importing them, and executing the function :func:`test_main` if present or loading the tests via unittest.TestLoader.loadTestsFromModule if `test_main` does not exist. The names of tests to execute may also be passed to the script. Specifying a single regression test (:program:`python -m test test_spam`) will minimize output and only print whether the test passed or failed.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 162); *backlink*

Unknown interpreted text role "mod".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 162); *backlink*

Unknown interpreted text role "option".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 162); *backlink*

Unknown interpreted text role "program".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 162); *backlink*

Unknown interpreted text role "mod".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 162); *backlink*

Unknown interpreted text role "program".

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

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 162); *backlink*

Unknown interpreted text role "mod".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 162); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 162); *backlink*

Unknown interpreted text role "program".

Running :mod:`test` directly allows what resources are available for tests to use to be set. You do this by using the `-u` command-line option. Specifying `all` as the value for the `-u` option enables all possible resources: :program:`python -m test -uall`. If all but one resource is desired (a more common case), a comma-separated list of resources that are not desired may be listed after `all`. The command :program:`python -m test -uall,-audio,-largefile` will run :mod:`test` with all resources except the `audio` and `largefile` resources. For a list of all resources and more command-line options, run :program:`python -m test -h`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 175); *backlink*

Unknown interpreted text role "mod".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 175); *backlink*

Unknown interpreted text role "program".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 175); *backlink*

Unknown interpreted text role "program".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 175); *backlink*

Unknown interpreted text role "mod".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 175); *backlink*

Unknown interpreted text role "program".

Some other ways to execute the regression tests depend on what platform the tests are being executed on. On Unix, you can run :program:`make test` at the top-level directory where Python was built. On Windows, executing :program:`rt.bat` from your :file:`PCbuild` directory will run all regression tests.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 186); *backlink*

Unknown interpreted text role "program".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 186); *backlink*

Unknown interpreted text role "program".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 186); *backlink*

Unknown interpreted text role "file".

:mod:`test.support` --- Utilities for the Python test suite

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 193); *backlink*

Unknown interpreted text role "mod".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 196)

Unknown directive type "module".

```
.. module:: test.support
   :synopsis: Support for Python's regression test suite.
```

The :mod:`test.support` module provides support for Python's regression test suite.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 200); *backlink*

Unknown interpreted text role "mod".

Note

:mod:`test.support` is not a public module. It is documented here to help Python developers write tests. The API of this module is subject to change without backwards compatibility concerns between releases.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 205); *backlink*

Unknown interpreted text role "mod".

This module defines the following exceptions:

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 212)

Unknown directive type "exception".

```
.. exception:: TestFailed

   Exception to be raised when a test fails. This is deprecated in favor of
   :mod:`unittest`\ -based tests and :class:`unittest.TestCase`'s assertion
   methods.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 219)

Unknown directive type "exception".

```
.. exception:: ResourceDenied

   Subclass of :exc:`unittest.SkipTest`. Raised when a resource (such as a
   network connection) is not available. Raised by the :func:`requires`
   function.
```

The :mod:`test.support` module defines the following constants:

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 226); *backlink*

Unknown interpreted text role "mod".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 228)

Unknown directive type "data".

```
.. data:: verbose

   ``True`` when verbose output is enabled. Should be checked when more
   detailed information is desired about a running test. *verbose* is set by
   :mod:`test.regrtest`.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 235)

Unknown directive type "data".

```
.. data:: is_jython

   ``True`` if the running interpreter is Jython.
```

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

Unknown directive type "data".

```
.. data:: is_android

   ``True`` if the system is Android.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 245)

Unknown directive type "data".

```
.. data:: unix_shell

   Path for shell if not on Windows; otherwise ``None``.
```

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

Unknown directive type "data".

```
.. data:: LOOPBACK_TIMEOUT

   Timeout in seconds for tests using a network server listening on the network
   local loopback interface like ``127.0.0.1``.

   The timeout is long enough to prevent test failure: it takes into account
   that the client and the server can run in different threads or even
   different processes.

   The timeout should be long enough for :meth:`~socket.socket.connect`,
   :meth:`~socket.socket.recv` and :meth:`~socket.socket.send` methods of
   :class:`socket.socket`.

   Its default value is 5 seconds.

   See also :data:`INTERNET_TIMEOUT`.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 268)

Unknown directive type "data".

```
.. data:: INTERNET_TIMEOUT

   Timeout in seconds for network requests going to the internet.

   The timeout is short enough to prevent a test to wait for too long if the
   internet request is blocked for whatever reason.

   Usually, a timeout using :data:`INTERNET_TIMEOUT` should not mark a test as
   failed, but skip the test instead: see
   :func:`~test.support.socket_helper.transient_internet`.

   Its default value is 1 minute.

   See also :data:`LOOPBACK_TIMEOUT`.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 284)

Unknown directive type "data".

```
.. data:: SHORT_TIMEOUT

   Timeout in seconds to mark a test as failed if the test takes "too long".

   The timeout value depends on the regrtest ``--timeout`` command line option.

   If a test using :data:`SHORT_TIMEOUT` starts to fail randomly on slow
   buildbots, use :data:`LONG_TIMEOUT` instead.

   Its default value is 30 seconds.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 296)

Unknown directive type "data".

```
.. data:: LONG_TIMEOUT

   Timeout in seconds to detect when a test hangs.

   It is long enough to reduce the risk of test failure on the slowest Python
   buildbots. It should not be used to mark a test as failed if the test takes
   "too long".  The timeout value depends on the regrtest ``--timeout`` command
   line option.

   Its default value is 5 minutes.

   See also :data:`LOOPBACK_TIMEOUT`, :data:`INTERNET_TIMEOUT` and
   :data:`SHORT_TIMEOUT`.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 311)

Unknown directive type "data".

```
.. data:: PGO

   Set when tests can be skipped when they are not useful for PGO.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 316)

Unknown directive type "data".

```
.. data:: PIPE_MAX_SIZE

   A constant that is likely larger than the underlying OS pipe buffer size,
   to make writes blocking.
```

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

Unknown directive type "data".

```
.. data:: SOCK_MAX_SIZE

   A constant that is likely larger than the underlying OS socket buffer size,
   to make writes blocking.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 328)

Unknown directive type "data".

```
.. data:: TEST_SUPPORT_DIR

   Set to the top level directory that contains :mod:`test.support`.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 333)

Unknown directive type "data".

```
.. data:: TEST_HOME_DIR

   Set to the top level directory for the test package.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 338)

Unknown directive type "data".

```
.. data:: TEST_DATA_DIR

   Set to the ``data`` directory within the test package.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 343)

Unknown directive type "data".

```
.. data:: MAX_Py_ssize_t

   Set to :data:`sys.maxsize` for big memory tests.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 348)

Unknown directive type "data".

```
.. data:: max_memuse

   Set by :func:`set_memlimit` as the memory limit for big memory tests.
   Limited by :data:`MAX_Py_ssize_t`.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 354)

Unknown directive type "data".

```
.. data:: real_max_memuse

   Set by :func:`set_memlimit` as the memory limit for big memory tests.  Not
   limited by :data:`MAX_Py_ssize_t`.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 360)

Unknown directive type "data".

```
.. data:: MISSING_C_DOCSTRINGS

   Return ``True`` if running on CPython, not on Windows, and configuration
   not set with ``WITH_DOC_STRINGS``.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 366)

Unknown directive type "data".

```
.. data:: HAVE_DOCSTRINGS

   Check for presence of docstrings.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 371)

Unknown directive type "data".

```
.. data:: TEST_HTTP_URL

   Define the URL of a dedicated HTTP server for the network tests.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 376)

Unknown directive type "data".

```
.. data:: ALWAYS_EQ

   Object that is equal to anything.  Used to test mixed type comparison.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 381)

Unknown directive type "data".

```
.. data:: NEVER_EQ

   Object that is not equal to anything (even to :data:`ALWAYS_EQ`).
   Used to test mixed type comparison.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 387)

Unknown directive type "data".

```
.. data:: LARGEST

   Object that is greater than anything (except itself).
   Used to test mixed type comparison.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 393)

Unknown directive type "data".

```
.. data:: SMALLEST

   Object that is less than anything (except itself).
   Used to test mixed type comparison.
```

The :mod:`test.support` module defines the following functions:

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 399); *backlink*

Unknown interpreted text role "mod".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 401)

Unknown directive type "function".

```
.. function:: is_resource_enabled(resource)

   Return ``True`` if *resource* is enabled and available. The list of
   available resources is only set when :mod:`test.regrtest` is executing the
   tests.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 408)

Unknown directive type "function".

```
.. function:: python_is_optimized()

   Return ``True`` if Python was not built with ``-O0`` or ``-Og``.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 413)

Unknown directive type "function".

```
.. function:: with_pymalloc()

   Return :data:`_testcapi.WITH_PYMALLOC`.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 418)

Unknown directive type "function".

```
.. function:: requires(resource, msg=None)

   Raise :exc:`ResourceDenied` if *resource* is not available. *msg* is the
   argument to :exc:`ResourceDenied` if it is raised. Always returns
   ``True`` if called by a function whose ``__name__`` is ``'__main__'``.
   Used when tests are executed by :mod:`test.regrtest`.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 426)

Unknown directive type "function".

```
.. function:: system_must_validate_cert(f)
```

```
   Raise :exc:`unittest.SkipTest` on TLS certification validation failures.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 431)

Unknown directive type "function".

```
.. function:: sortdict(dict)

   Return a repr of *dict* with keys sorted.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 436)

Unknown directive type "function".

```
.. function:: findfile(filename, subdir=None)

   Return the path to the file named *filename*. If no match is found
   *filename* is returned. This does not equal a failure since it could be the
   path to the file.

   Setting *subdir* indicates a relative path to use to find the file
   rather than looking directly in the path directories.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 446)

Unknown directive type "function".

```
.. function:: match_test(test)

   Match *test* to patterns set in :func:`set_match_tests`.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 451)

Unknown directive type "function".

```
.. function:: set_match_tests(patterns)

   Define match test with regular expression *patterns*.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 456)

Unknown directive type "function".

```
.. function:: run_unittest(*classes)

   Execute :class:`unittest.TestCase` subclasses passed to the function. The
   function scans the classes for methods starting with the prefix ``test_``
   and executes the tests individually.

   It is also legal to pass strings as parameters; these should be keys in
   ``sys.modules``. Each associated module will be scanned by
   ``unittest.TestLoader.loadTestsFromModule()``. This is usually seen in the
   following :func:`test_main` function::

      def test_main():
          support.run_unittest(__name__)

   This will run all tests defined in the named module.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 473)

Unknown directive type "function".

```
.. function:: run_doctest(module, verbosity=None, optionflags=0)

   Run :func:`doctest.testmod` on the given *module*.  Return
   ``(failure_count, test_count)``.

   If *verbosity* is ``None``, :func:`doctest.testmod` is run with verbosity
   set to :data:`verbose`.  Otherwise, it is run with verbosity set to
   ``None``.  *optionflags* is passed as ``optionflags`` to
   :func:`doctest.testmod`.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 484)

Unknown directive type "function".

```
.. function:: setswitchinterval(interval)

   Set the :func:`sys.setswitchinterval` to the given *interval*.  Defines
   a minimum interval for Android systems to prevent the system from hanging.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 490)

Unknown directive type "function".

```
.. function:: check_impl_detail(**guards)

   Use this check to guard CPython's implementation-specific tests or to
   run them only on the implementations guarded by the arguments::

      check_impl_detail()               # Only on CPython (default).
      check_impl_detail(jython=True)    # Only on Jython.
      check_impl_detail(cpython=False)  # Everywhere except CPython.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 500)

Unknown directive type "function".

```
.. function:: set_memlimit(limit)

   Set the values for :data:`max_memuse` and :data:`real_max_memuse` for big
   memory tests.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 506)

Unknown directive type "function".

```
.. function:: record_original_stdout(stdout)

   Store the value from *stdout*.  It is meant to hold the stdout at the
   time the regrtest began.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 512)

Unknown directive type "function".

```
.. function:: get_original_stdout
```

```
   Return the original stdout set by :func:`record_original_stdout` or
   ``sys.stdout`` if it's not set.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 518)

Unknown directive type "function".

```
.. function:: args_from_interpreter_flags()

   Return a list of command line arguments reproducing the current settings
   in ``sys.flags`` and ``sys.warnoptions``.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 524)

Unknown directive type "function".

```
.. function:: optim_args_from_interpreter_flags()

   Return a list of command line arguments reproducing the current
   optimization settings in ``sys.flags``.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 530)

Unknown directive type "function".

```
.. function:: captured_stdin()
              captured_stdout()
              captured_stderr()

   A context managers that temporarily replaces the named stream with
   :class:`io.StringIO` object.

   Example use with output streams::

      with captured_stdout() as stdout, captured_stderr() as stderr:
          print("hello")
          print("error", file=sys.stderr)
      assert stdout.getvalue() == "hello\n"
      assert stderr.getvalue() == "error\n"

   Example use with input stream::

      with captured_stdin() as stdin:
          stdin.write('hello\n')
          stdin.seek(0)
          # call test code that consumes from sys.stdin
          captured = input()
      self.assertEqual(captured, "hello")
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 555)

Unknown directive type "function".

```
.. function:: disable_faulthandler()

   A context manager that replaces ``sys.stderr`` with ``sys.__stderr__``.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 560)

Unknown directive type "function".

```
.. function:: gc_collect()

   Force as many objects as possible to be collected.  This is needed because
   timely deallocation is not guaranteed by the garbage collector.  This means
   that ``__del__`` methods may be called later than expected and weakrefs
   may remain alive for longer than expected.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 568)

Unknown directive type "function".

```
.. function:: disable_gc()

   A context manager that disables the garbage collector upon entry and
   reenables it upon exit.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 574)

Unknown directive type "function".

```
.. function:: swap_attr(obj, attr, new_val)

   Context manager to swap out an attribute with a new object.

   Usage::

      with swap_attr(obj, "attr", 5):
          ...

   This will set ``obj.attr`` to 5 for the duration of the ``with`` block,
   restoring the old value at the end of the block.  If ``attr`` doesn't
   exist on ``obj``, it will be created and then deleted at the end of the
   block.

   The old value (or ``None`` if it doesn't exist) will be assigned to the
   target of the "as" clause, if there is one.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 592)

Unknown directive type "function".

```
.. function:: swap_item(obj, attr, new_val)

   Context manager to swap out an item with a new object.

   Usage::

      with swap_item(obj, "item", 5):
          ...

   This will set ``obj["item"]`` to 5 for the duration of the ``with`` block,
   restoring the old value at the end of the block. If ``item`` doesn't
   exist on ``obj``, it will be created and then deleted at the end of the
   block.

   The old value (or ``None`` if it doesn't exist) will be assigned to the
   target of the "as" clause, if there is one.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 610)

Unknown directive type "function".

```
.. function:: flush_std_streams()

   Call the ``flush()`` method on :data:`sys.stdout` and then on
```

```
   :data:`sys.stderr`. It can be used to make sure that the logs order is
   consistent before writing into stderr.

   .. versionadded:: 3.11
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 619)

Unknown directive type "function".

```
.. function:: print_warning(msg)

   Print a warning into :data:`sys.__stderr__`. Format the message as:
   ``f"Warning -- {msg}"``. If *msg* is made of multiple lines, add
   ``"Warning -- "`` prefix to each line.

   .. versionadded:: 3.9
```

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

Unknown directive type "function".

```
.. function:: wait_process(pid, *, exitcode, timeout=None)

   Wait until process *pid* completes and check that the process exit code is
   *exitcode*.

   Raise an :exc:`AssertionError` if the process exit code is not equal to
   *exitcode*.

   If the process runs longer than *timeout* seconds (:data:`SHORT_TIMEOUT` by
   default), kill the process and raise an :exc:`AssertionError`. The timeout
   feature is not available on Windows.

   .. versionadded:: 3.9
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 643)

Unknown directive type "function".

```
.. function:: calcobjsize(fmt)

   Return :func:`struct.calcsize` for ``nP{fmt}0n`` or, if ``gettotalrefcount``
   exists, ``2PnP{fmt}0P``.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 649)

Unknown directive type "function".

```
.. function:: calcvobjsize(fmt)

   Return :func:`struct.calcsize` for ``nPn{fmt}0n`` or, if ``gettotalrefcount``
   exists, ``2PnPn{fmt}0P``.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 655)

Unknown directive type "function".

```
.. function:: checksizeof(test, o, size)

   For testcase *test*, assert that the ``sys.getsizeof`` for *o* plus the GC
   header size equals *size*.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 661)

Unknown directive type "decorator".

```
.. decorator:: anticipate_failure(condition)

   A decorator to conditionally mark tests with
   :func:`unittest.expectedFailure`. Any use of this decorator should
   have an associated comment identifying the relevant tracker issue.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 668)

Unknown directive type "decorator".

```
.. decorator:: run_with_locale(catstr, *locales)

   A decorator for running a function in a different locale, correctly
   resetting it after it has finished.  *catstr* is the locale category as
   a string (for example ``"LC_ALL"``).  The *locales* passed will be tried
   sequentially, and the first valid locale will be used.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 676)

Unknown directive type "decorator".

```
.. decorator:: run_with_tz(tz)

   A decorator for running a function in a specific timezone, correctly
   resetting it after it has finished.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 682)

Unknown directive type "decorator".

```
.. decorator:: requires_freebsd_version(*min_version)

   Decorator for the minimum version when running test on FreeBSD.  If the
   FreeBSD version is less than the minimum, raise :exc:`unittest.SkipTest`.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 688)

Unknown directive type "decorator".

```
.. decorator:: requires_linux_version(*min_version)

   Decorator for the minimum version when running test on Linux.  If the
   Linux version is less than the minimum, raise :exc:`unittest.SkipTest`.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 694)

Unknown directive type "decorator".

```
.. decorator:: requires_mac_version(*min_version)

   Decorator for the minimum version when running test on macOS.  If the
   macOS version is less than the minimum, raise :exc:`unittest.SkipTest`.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 700)

Unknown directive type "decorator".

```
.. decorator:: requires_IEEE_754

   Decorator for skipping tests on non-IEEE 754 platforms.
```

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

Unknown directive type "decorator".

```
.. decorator:: requires_zlib

   Decorator for skipping tests if :mod:`zlib` doesn't exist.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 710)

Unknown directive type "decorator".

```
.. decorator:: requires_gzip

   Decorator for skipping tests if :mod:`gzip` doesn't exist.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 715)

Unknown directive type "decorator".

```
.. decorator:: requires_bz2

   Decorator for skipping tests if :mod:`bz2` doesn't exist.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 720)

Unknown directive type "decorator".

```
.. decorator:: requires_lzma

   Decorator for skipping tests if :mod:`lzma` doesn't exist.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 725)

Unknown directive type "decorator".

```
.. decorator:: requires_resource(resource)

   Decorator for skipping tests if *resource* is not available.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 730)

Unknown directive type "decorator".

```
.. decorator:: requires_docstrings

   Decorator for only running the test if :data:`HAVE_DOCSTRINGS`.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 735)

Unknown directive type "decorator".

```
.. decorator:: cpython_only(test)

   Decorator for tests only applicable to CPython.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 740)

Unknown directive type "decorator".

```
.. decorator:: impl_detail(msg=None, **guards)

   Decorator for invoking :func:`check_impl_detail` on *guards*.  If that
   returns ``False``, then uses *msg* as the reason for skipping the test.
```

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

Unknown directive type "decorator".

```
.. decorator:: no_tracing(func)

   Decorator to temporarily turn off tracing for the duration of the test.
```

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

Unknown directive type "decorator".

```
.. decorator:: refcount_test(test)

   Decorator for tests which involve reference counting.  The decorator does
   not run the test if it is not run by CPython.  Any trace function is unset
   for the duration of the test to prevent unexpected refcounts caused by
   the trace function.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 759)

Unknown directive type "decorator".

```
.. decorator:: bigmemtest(size, memuse, dry_run=True)

   Decorator for bigmem tests.

   *size* is a requested size for the test (in arbitrary, test-interpreted
   units.)  *memuse* is the number of bytes per unit for the test, or a good
   estimate of it.  For example, a test that needs two byte buffers, of 4 GiB
   each, could be decorated with ``@bigmemtest(size=_4G, memuse=2)``.

   The *size* argument is normally passed to the decorated test method as an
   extra argument.  If *dry_run* is ``True``, the value passed to the test
   method may be less than the requested value.  If *dry_run* is ``False``, it
   means the test doesn't support dummy runs when ``-M`` is not specified.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 774)

Unknown directive type "decorator".

```
.. decorator:: bigaddrspacetest(f)

   Decorator for tests that fill the address space.  *f* is the function to
   wrap.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 780)

Unknown directive type "function".

```
.. function:: check_syntax_error(testcase, statement, errtext='', *, lineno=None, offset=None)

   Test for syntax errors in *statement* by attempting to compile *statement*.
   *testcase* is the :mod:`unittest` instance for the test.  *errtext* is the
   regular expression which should match the string representation of the
   raised :exc:`SyntaxError`.  If *lineno* is not ``None``, compares to
   the line of the exception.  If *offset* is not ``None``, compares to
   the offset of the exception.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 790)

Unknown directive type "function".

```
.. function:: open_urlresource(url, *args, **kw)

   Open *url*.  If open fails, raises :exc:`TestFailed`.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 795)

Unknown directive type "function".

```
.. function:: reap_children()

   Use this at the end of ``test_main`` whenever sub-processes are started.
   This will help ensure that no extra children (zombies) stick around to
   hog resources and create problems when looking for refleaks.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 802)

Unknown directive type "function".

```
.. function:: get_attribute(obj, name)

   Get an attribute, raising :exc:`unittest.SkipTest` if :exc:`AttributeError`
   is raised.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 808)

Unknown directive type "function".

```
.. function:: catch_unraisable_exception()

   Context manager catching unraisable exception using
   :func:`sys.unraisablehook`.

   Storing the exception value (``cm.unraisable.exc_value``) creates a
   reference cycle. The reference cycle is broken explicitly when the context
```

```
manager exits.

Storing the object (``cm.unraisable.object``) can resurrect it if it is set
to an object which is being finalized. Exiting the context manager clears
the stored object.

Usage::

    with support.catch_unraisable_exception() as cm:
        # code creating an "unraisable exception"
        ...

        # check the unraisable exception: use cm.unraisable
        ...

    # cm.unraisable attribute no longer exists at this point
    # (to break a reference cycle)

.. versionadded:: 3.8
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 836)

Unknown directive type "function".

```
.. function:: load_package_tests(pkg_dir, loader, standard_tests, pattern)

   Generic implementation of the :mod:`unittest` ``load_tests`` protocol for
   use in test packages.  *pkg_dir* is the root directory of the package;
   *loader*, *standard_tests*, and *pattern* are the arguments expected by
   ``load_tests``.  In simple cases, the test package's ``__init__.py``
   can be the following::

      import os
      from test.support import load_package_tests

      def load_tests(*args):
          return load_package_tests(os.path.dirname(__file__), *args)
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 851)

Unknown directive type "function".

```
.. function:: detect_api_mismatch(ref_api, other_api, *, ignore=())

   Returns the set of attributes, functions or methods of *ref_api* not
   found on *other_api*, except for a defined list of items to be
   ignored in this check specified in *ignore*.

   By default this skips private attributes beginning with '_' but
   includes all magic methods, i.e. those starting and ending in '__'.

   .. versionadded:: 3.5
```

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

Unknown directive type "function".

```
.. function:: patch(test_instance, object_to_patch, attr_name, new_value)

   Override *object_to_patch.attr_name* with *new_value*.  Also add
   cleanup procedure to *test_instance* to restore *object_to_patch* for
   *attr_name*.  The *attr_name* should be a valid attribute for
   *object_to_patch*.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 871)

Unknown directive type "function".

```
.. function:: run_in_subinterp(code)

   Run *code* in subinterpreter.  Raise :exc:`unittest.SkipTest` if
   :mod:`tracemalloc` is enabled.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 877)

Unknown directive type "function".

```
.. function:: check_free_after_iterating(test, iter, cls, args=())

   Assert that *iter* is deallocated after iterating.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 882)

Unknown directive type "function".

```
.. function:: missing_compiler_executable(cmd_names=[])

   Check for the existence of the compiler executables whose names are listed
   in *cmd_names* or all the compiler executables when *cmd_names* is empty
   and return the first missing executable or ``None`` when none is found
   missing.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 890)

Unknown directive type "function".

```
.. function:: check__all__(test_case, module, name_of_module=None, extra=(), not_exported=())

   Assert that the ``__all__`` variable of *module* contains all public names.

   The module's public names (its API) are detected automatically
   based on whether they match the public name convention and were defined in
   *module*.

   The *name_of_module* argument can specify (as a string or tuple thereof) what
   module(s) an API could be defined in order to be detected as a public
   API. One case for this is when *module* imports part of its public API from
   other modules, possibly a C backend (like ``csv`` and its ``_csv``).

   The *extra* argument can be a set of names that wouldn't otherwise be automatically
   detected as "public", like objects without a proper ``__module__``
   attribute. If provided, it will be added to the automatically detected ones.

   The *not_exported* argument can be a set of names that must not be treated
   as part of the public API even though their names indicate otherwise.

   Example use::

      import bar
      import foo
      import unittest
      from test import support

      class MiscTestCase(unittest.TestCase):
          def test__all__(self):
              support.check__all__(self, foo)

      class OtherTestCase(unittest.TestCase):
          def test__all__(self):
              extra = {'BAR_CONST', 'FOO_CONST'}
              not_exported = {'baz'}  # Undocumented name.
              # bar imports part of its API from _bar.
              support.check__all__(self, bar, ('bar', '_bar'),
                                   extra=extra, not_exported=not_exported)
```

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

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 931)

Unknown directive type "function".

```
.. function:: skip_if_broken_multiprocessing_synchronize()

   Skip tests if the :mod:`multiprocessing.synchronize` module is missing, if
   there is no available semaphore implementation, or if creating a lock raises
   an :exc:`OSError`.

   .. versionadded:: 3.10
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 940)

Unknown directive type "function".

```
.. function:: check_disallow_instantiation(test_case, tp, *args, **kwds)

   Assert that type *tp* cannot be instantiated using *args* and *kwds*.

   .. versionadded:: 3.10
```

The :mod:`test.support` module defines the following classes:

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 947); *backlink*

Unknown interpreted text role "mod".

A context manager used to try to prevent crash dialog popups on tests that are expected to crash a subprocess.

On Windows, it disables Windows Error Reporting dialogs using SetErrorMode.

On UNIX, :func:`resource.setrlimit` is used to set :attr:`resource.RLIMIT_CORE`'s soft limit to 0 to prevent coredump file creation.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 958); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 958); *backlink*

Unknown interpreted text role "attr".

On both platforms, the old value is restored by :meth:`__exit__`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 962); *backlink*

Unknown interpreted text role "meth".

Class to save and restore signal handlers registered by the Python signal handler.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 973)

Unknown directive type "method".

```
.. method:: matches(self, d, **kwargs)

   Try to match a single dict with the supplied arguments.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 978)

Unknown directive type "method".

```
.. method:: match_value(self, k, dv, v)

   Try to match a single stored value (*dv*) with a supplied value (*v*).
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 985)

Unknown directive type "method".

```
.. method:: run(test)

   Run *test* and return the result.
```

:mod:`test.support.socket_helper` --- Utilities for socket tests

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 990); *backlink*

Unknown interpreted text role "mod".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 993)

Unknown directive type "module".

```
.. module:: test.support.socket_helper
   :synopsis: Support for socket tests.
```

The :mod:`test.support.socket_helper` module provides support for socket tests.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 997); *backlink*

Unknown interpreted text role "mod".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 999)

Unknown directive type "versionadded".

```
.. versionadded:: 3.9
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1002)

Unknown directive type "data".

```
.. data:: IPV6_ENABLED

    Set to ``True`` if IPv6 is enabled on this host, ``False`` otherwise.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1007)

Unknown directive type "function".

```
.. function:: find_unused_port(family=socket.AF_INET, socktype=socket.SOCK_STREAM)
```

```
   Returns an unused port that should be suitable for binding.  This is
   achieved by creating a temporary socket with the same family and type as
   the ``sock`` parameter (default is :const:`~socket.AF_INET`,
   :const:`~socket.SOCK_STREAM`),
   and binding it to the specified host address (defaults to ``0.0.0.0``)
   with the port set to 0, eliciting an unused ephemeral port from the OS.
   The temporary socket is then closed and deleted, and the ephemeral port is
   returned.

   Either this method or :func:`bind_port` should be used for any tests
   where a server socket needs to be bound to a particular port for the
   duration of the test.
   Which one to use depends on whether the calling code is creating a Python
   socket, or if an unused port needs to be provided in a constructor
   or passed to an external program (i.e. the ``-accept`` argument to
   openssl's s_server mode).  Always prefer :func:`bind_port` over
   :func:`find_unused_port` where possible.  Using a hard coded port is
   discouraged since it can make multiple instances of the test impossible to
   run simultaneously, which is a problem for buildbots.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1030)

Unknown directive type "function".

```
.. function:: bind_port(sock, host=HOST)

   Bind the socket to a free port and return the port number.  Relies on
   ephemeral ports in order to ensure we are using an unbound port.  This is
   important as many tests may be running simultaneously, especially in a
   buildbot environment.  This method raises an exception if the
   ``sock.family`` is :const:`~socket.AF_INET` and ``sock.type`` is
   :const:`~socket.SOCK_STREAM`, and the socket has
   :const:`~socket.SO_REUSEADDR` or :const:`~socket.SO_REUSEPORT` set on it.
   Tests should never set these socket options for TCP/IP sockets.
   The only case for setting these options is testing multicasting via
   multiple UDP sockets.

   Additionally, if the :const:`~socket.SO_EXCLUSIVEADDRUSE` socket option is
   available (i.e. on Windows), it will be set on the socket.  This will
   prevent anyone else from binding to our host/port for the duration of the
   test.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1049)

Unknown directive type "function".

```
.. function:: bind_unix_socket(sock, addr)

   Bind a unix socket, raising :exc:`unittest.SkipTest` if
   :exc:`PermissionError` is raised.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1055)

Unknown directive type "decorator".

```
.. decorator:: skip_unless_bind_unix_socket

   A decorator for running tests that require a functional ``bind()`` for Unix
   sockets.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1061)

Unknown directive type "function".

```
.. function:: transient_internet(resource_name, *, timeout=30.0, errnos=())

   A context manager that raises :exc:`~test.support.ResourceDenied` when
   various issues with the internet connection manifest themselves as
   exceptions.
```

:mod:`test.support.script_helper` --- Utilities for the Python execution tests

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1068); *backlink*

Unknown interpreted text role "mod".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1071)

Unknown directive type "module".

```
.. module:: test.support.script_helper
   :synopsis: Support for Python's script execution tests.
```

The :mod:`test.support.script_helper` module provides support for Python's script execution tests.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1075); *backlink*

Unknown interpreted text role "mod".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1078)

Unknown directive type "function".

```
.. function:: interpreter_requires_environment()

   Return ``True`` if ``sys.executable interpreter`` requires environment
   variables in order to be able to run at all.

   This is designed to be used with ``@unittest.skipIf()`` to annotate tests
   that need to use an ``assert_python*()`` function to launch an isolated
   mode (``-I``) or no environment mode (``-E``) sub-interpreter process.

   A normal build & test does not run into this situation but it can happen
   when trying to run the standard library test suite from an interpreter that
   doesn't have an obvious home with Python's current home finding logic.

   Setting :envvar:`PYTHONHOME` is one way to get most of the testsuite to run
   in that situation.  :envvar:`PYTHONPATH` or :envvar:`PYTHONUSERSITE` are
   other common environment variables that might impact whether or not the
   interpreter can start.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1097)

Unknown directive type "function".

```
.. function:: run_python_until_end(*args, **env_vars)

   Set up the environment based on *env_vars* for running the interpreter
   in a subprocess.  The values can include ``__isolated``, ``__cleanenv``,
   ``__cwd``, and ``TERM``.

   .. versionchanged:: 3.9
      The function no longer strips whitespaces from *stderr*.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1107)

Unknown directive type "function".

```
.. function:: assert_python_ok(*args, **env_vars)

   Assert that running the interpreter with *args* and optional environment
   variables *env_vars* succeeds (``rc == 0``) and return a ``(return code,
   stdout, stderr)`` tuple.

   If the ``__cleanenv`` keyword is set, *env_vars* is used as a fresh
   environment.

   Python is started in isolated mode (command line option ``-I``),
   except if the ``__isolated`` keyword is set to ``False``.

   .. versionchanged:: 3.9
      The function no longer strips whitespaces from *stderr*.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1123)

Unknown directive type "function".

```
.. function:: assert_python_failure(*args, **env_vars)

   Assert that running the interpreter with *args* and optional environment
   variables *env_vars* fails (``rc != 0``) and return a ``(return code,
   stdout, stderr)`` tuple.

   See :func:`assert_python_ok` for more options.

   .. versionchanged:: 3.9
      The function no longer strips whitespaces from *stderr*.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1135)

Unknown directive type "function".

```
.. function:: spawn_python(*args, stdout=subprocess.PIPE, stderr=subprocess.STDOUT, **kw)

   Run a Python subprocess with the given arguments.

   *kw* is extra keyword args to pass to :func:`subprocess.Popen`. Returns a
   :class:`subprocess.Popen` object.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1143)

Unknown directive type "function".

```
.. function:: kill_python(p)

   Run the given :class:`subprocess.Popen` process until completion and return
   stdout.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1149)

Unknown directive type "function".

```
.. function:: make_script(script_dir, script_basename, source, omit_suffix=False)

   Create script containing *source* in path *script_dir* and *script_basename*.
   If *omit_suffix* is ``False``, append ``.py`` to the name.  Return the full
   script path.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1156)

Unknown directive type "function".

```
.. function:: make_zip_script(zip_dir, zip_basename, script_name, name_in_zip=None)

   Create zip file at *zip_dir* and *zip_basename* with extension ``zip`` which
   contains the files in *script_name*. *name_in_zip* is the archive name.
   Return a tuple containing ``(full path, full path of archive name)``.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1163)

Unknown directive type "function".

```
.. function:: make_pkg(pkg_dir, init_source='')

   Create a directory named *pkg_dir* containing an ``__init__`` file with
   *init_source* as its contents.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1169)

Unknown directive type "function".

```
.. function:: make_zip_pkg(zip_dir, zip_basename, pkg_name, script_basename, \
                           source, depth=1, compiled=False)

   Create a zip package directory with a path of *zip_dir* and *zip_basename*
   containing an empty ``__init__`` file and a file *script_basename*
   containing the *source*.  If *compiled* is ``True``, both source files will
   be compiled and added to the zip package.  Return a tuple of the full zip
   path and the archive name for the zip file.
```

:mod:`test.support.bytecode_helper` --- Support tools for testing correct bytecode generation

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1179); *backlink*

Unknown interpreted text role "mod".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1182)

Unknown directive type "module".

```
.. module:: test.support.bytecode_helper
   :synopsis: Support tools for testing correct bytecode generation.
```

The :mod:`test.support.bytecode_helper` module provides support for testing and inspecting bytecode generation.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1185); *backlink*

Unknown interpreted text role "mod".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1188)

Unknown directive type "versionadded".

```
.. versionadded:: 3.9
```

The module defines the following class:

This class has custom assertion methods for inspecting bytecode.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1196)

Unknown directive type "method".

```
.. method:: BytecodeTestCase.get_disassembly_as_string(co)

   Return the disassembly of *co* as string.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1201)

Unknown directive type "method".

```
.. method:: BytecodeTestCase.assertInBytecode(x, opname, argval=_UNSPECIFIED)

   Return instr if *opname* is found, otherwise throws :exc:`AssertionError`.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1206)

Unknown directive type "method".

```
.. method:: BytecodeTestCase.assertNotInBytecode(x, opname, argval=_UNSPECIFIED)

   Throws :exc:`AssertionError` if *opname* is found.
```

:mod:`test.support.threading_helper` --- Utilities for threading tests

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1211); *backlink*

Unknown interpreted text role "mod".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1214)

Unknown directive type "module".

```
.. module:: test.support.threading_helper
   :synopsis: Support for threading tests.
```

The :mod:`test.support.threading_helper` module provides support for threading tests.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1217); *backlink*

Unknown interpreted text role "mod".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1219)

Unknown directive type "versionadded".

```
.. versionadded:: 3.10
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1222)

Unknown directive type "function".

```
.. function:: join_thread(thread, timeout=None)

   Join a *thread* within *timeout*.  Raise an :exc:`AssertionError` if thread
   is still alive after *timeout* seconds.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1228)

Unknown directive type "decorator".

```
.. decorator:: reap_threads(func)

   Decorator to ensure the threads are cleaned up even if the test fails.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1233)

Unknown directive type "function".

```
.. function:: start_threads(threads, unlock=None)

   Context manager to start *threads*.  It attempts to join the threads upon
   exit.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1239)

Unknown directive type "function".

```
.. function:: threading_cleanup(*original_values)

   Cleanup up threads not specified in *original_values*.  Designed to emit
   a warning if a test leaves running threads in the background.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1245)

Unknown directive type "function".

```
.. function:: threading_setup()

   Return current thread count and copy of dangling threads.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1250)

Unknown directive type "function".

```
.. function:: wait_threads_exit(timeout=None)

   Context manager to wait until all threads created in the ``with`` statement
   exit.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1256)

Unknown directive type "function".

```
.. function:: catch_threading_exception()

   Context manager catching :class:`threading.Thread` exception using
   :func:`threading.excepthook`.

   Attributes set when an exception is caught:

   * ``exc_type``
   * ``exc_value``
   * ``exc_traceback``
   * ``thread``

   See :func:`threading.excepthook` documentation.

   These attributes are deleted at the context manager exit.

   Usage::

       with threading_helper.catch_threading_exception() as cm:
           # code spawning a thread which raises an exception
           ...

           # check the thread exception, use cm attributes:
           # exc_type, exc_value, exc_traceback, thread
           ...

       # exc_type, exc_value, exc_traceback, thread attributes of cm no longer
       # exists at this point
       # (to avoid reference cycles)

   .. versionadded:: 3.8
```

:mod:`test.support.os_helper` --- Utilities for os tests

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1289); *backlink*

Unknown interpreted text role "mod".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1292)

Unknown directive type "module".

```
.. module:: test.support.os_helper
   :synopsis: Support for os tests.
```

The :mod:`test.support.os_helper` module provides support for os tests.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1295); *backlink*

Unknown interpreted text role "mod".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1297)

Unknown directive type "versionadded".

```
.. versionadded:: 3.10
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1300)

Unknown directive type "data".

```
.. data:: FS_NONASCII

   A non-ASCII character encodable by :func:`os.fsencode`.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1305)

Unknown directive type "data".

```
.. data:: SAVEDCWD

   Set to :func:`os.getcwd`.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1310)

Unknown directive type "data".

```
.. data:: TESTFN

   Set to a name that is safe to use as the name of a temporary file.  Any
   temporary file that is created should be closed and unlinked (removed).
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1316)

Unknown directive type "data".

```
.. data:: TESTFN_NONASCII

   Set to a filename containing the :data:`FS_NONASCII` character.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1321)

Unknown directive type "data".

```
.. data:: TESTFN_UNENCODABLE

   Set to a filename (str type) that should not be able to be encoded by file
   system encoding in strict mode.  It may be ``None`` if it's not possible to
   generate such a filename.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1328)

Unknown directive type "data".

```
.. data:: TESTFN_UNDECODABLE

   Set to a filename (bytes type) that should not be able to be decoded by
   file system encoding in strict mode.  It may be ``None`` if it's not
   possible to generate such a filename.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1335)

Unknown directive type "data".

```
.. data:: TESTFN_UNICODE

    Set to a non-ASCII name for a temporary file.
```

Class used to temporarily set or unset environment variables. Instances can be used as a context manager and have a complete

dictionary interface for querying/modifying the underlying `os.environ`. After exit from the context manager all changes to environment variables done through this instance will be rolled back.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1348)

Unknown directive type "versionchanged".

```
.. versionchanged:: 3.1
   Added dictionary interface.
```

Simple :term:`path-like object`. It implements the :meth:`__fspath__` method which just returns the *path* argument. If *path* is an exception, it will be raised in :meth:`!__fspath__`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1354); *backlink*

Unknown interpreted text role "term".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1354); *backlink*

Unknown interpreted text role "meth".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1354); *backlink*

Unknown interpreted text role "meth".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1359)

Unknown directive type "method".

```
.. method:: EnvironmentVarGuard.set(envvar, value)

   Temporarily set the environment variable ``envvar`` to the value of
   ``value``.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1365)

Unknown directive type "method".

```
.. method:: EnvironmentVarGuard.unset(envvar)

   Temporarily unset the environment variable ``envvar``.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1370)

Unknown directive type "function".

```
.. function:: can_symlink()

   Return ``True`` if the OS supports symbolic links, ``False``
   otherwise.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1376)

Unknown directive type "function".

```
.. function:: can_xattr()
```

```
   Return ``True`` if the OS supports xattr, ``False``
   otherwise.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1382)

Unknown directive type "function".

```
.. function:: change_cwd(path, quiet=False)

   A context manager that temporarily changes the current working
   directory to *path* and yields the directory.

   If *quiet* is ``False``, the context manager raises an exception
   on error.  Otherwise, it issues only a warning and keeps the current
   working directory the same.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1392)

Unknown directive type "function".

```
.. function:: create_empty_file(filename)

   Create an empty file with *filename*.  If it already exists, truncate it.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1397)

Unknown directive type "function".

```
.. function:: fd_count()

   Count the number of open file descriptors.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1402)

Unknown directive type "function".

```
.. function:: fs_is_case_insensitive(directory)

   Return ``True`` if the file system for *directory* is case-insensitive.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1407)

Unknown directive type "function".

```
.. function:: make_bad_fd()

   Create an invalid file descriptor by opening and closing a temporary file,
   and returning its descriptor.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1413)

Unknown directive type "function".

```
.. function:: rmdir(filename)

   Call :func:`os.rmdir` on *filename*.  On Windows platforms, this is
   wrapped with a wait loop that checks for the existence of the file.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1419)

Unknown directive type "function".

```
.. function:: rmtree(path)

   Call :func:`shutil.rmtree` on *path* or call :func:`os.lstat` and
   :func:`os.rmdir` to remove a path and its contents.  On Windows platforms,
   this is wrapped with a wait loop that checks for the existence of the files.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1426)

Unknown directive type "decorator".

```
.. decorator:: skip_unless_symlink

   A decorator for running tests that require support for symbolic links.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1431)

Unknown directive type "decorator".

```
.. decorator:: skip_unless_xattr

   A decorator for running tests that require support for xattr.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1436)

Unknown directive type "function".

```
.. function:: temp_cwd(name='tempcwd', quiet=False)

   A context manager that temporarily creates a new directory and
   changes the current working directory (CWD).

   The context manager creates a temporary directory in the current
   directory with name *name* before temporarily changing the current
   working directory.  If *name* is ``None``, the temporary directory is
   created using :func:`tempfile.mkdtemp`.

   If *quiet* is ``False`` and it is not possible to create or change
   the CWD, an error is raised.  Otherwise, only a warning is raised
   and the original CWD is used.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1451)

Unknown directive type "function".

```
.. function:: temp_dir(path=None, quiet=False)

   A context manager that creates a temporary directory at *path* and
   yields the directory.

   If *path* is ``None``, the temporary directory is created using
   :func:`tempfile.mkdtemp`.  If *quiet* is ``False``, the context manager
   raises an exception on error.  Otherwise, if *path* is specified and
   cannot be created, only a warning is issued.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1462)

Unknown directive type "function".

```
.. function:: temp_umask(umask)

   A context manager that temporarily sets the process umask.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1467)

Unknown directive type "function".

```
.. function:: unlink(filename)

   Call :func:`os.unlink` on *filename*.  On Windows platforms, this is
   wrapped with a wait loop that checks for the existence of the file.
```

:mod:`test.support.import_helper` --- Utilities for import tests

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1473); *backlink*

Unknown interpreted text role "mod".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1476)

Unknown directive type "module".

```
.. module:: test.support.import_helper
   :synopsis: Support for import tests.
```

The :mod:`test.support.import_helper` module provides support for import tests.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1479); *backlink*

Unknown interpreted text role "mod".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1481)

Unknown directive type "versionadded".

```
.. versionadded:: 3.10
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1484)

Unknown directive type "function".

```
.. function:: forget(module_name)

   Remove the module named *module_name* from ``sys.modules`` and delete any
   byte-compiled files of the module.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1490)

Unknown directive type "function".

```
.. function:: import_fresh_module(name, fresh=(), blocked=(), deprecated=False)

   This function imports and returns a fresh copy of the named Python module
   by removing the named module from ``sys.modules`` before doing the import.
   Note that unlike :func:`reload`, the original module is not affected by
   this operation.

   *fresh* is an iterable of additional module names that are also removed
   from the ``sys.modules`` cache before doing the import.

   *blocked* is an iterable of module names that are replaced with ``None``
   in the module cache during the import to ensure that attempts to import
   them raise :exc:`ImportError`.

   The named module and any modules named in the *fresh* and *blocked*
   parameters are saved before starting the import and then reinserted into
   ``sys.modules`` when the fresh import is complete.

   Module and package deprecation messages are suppressed during this import
   if *deprecated* is ``True``.

   This function will raise :exc:`ImportError` if the named module cannot be
   imported.

   Example use::

      # Get copies of the warnings module for testing without affecting the
      # version being used by the rest of the test suite. One copy uses the
      # C implementation, the other is forced to use the pure Python fallback
      # implementation
      py_warnings = import_fresh_module('warnings', blocked=['_warnings'])
      c_warnings = import_fresh_module('warnings', fresh=['_warnings'])

   .. versionadded:: 3.1
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1526)

Unknown directive type "function".

```
.. function:: import_module(name, deprecated=False, *, required_on())

   This function imports and returns the named module. Unlike a normal
   import, this function raises :exc:`unittest.SkipTest` if the module
   cannot be imported.

   Module and package deprecation messages are suppressed during this import
   if *deprecated* is ``True``.  If a module is required on a platform but
   optional for others, set *required_on* to an iterable of platform prefixes
   which will be compared against :data:`sys.platform`.

   .. versionadded:: 3.1
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1540)

Unknown directive type "function".

```
.. function:: modules_setup()

   Return a copy of :data:`sys.modules`.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1545)

Unknown directive type "function".

```
.. function:: modules_cleanup(oldmodules)

   Remove modules except for *oldmodules* and ``encodings`` in order to
   preserve internal cache.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1551)

Unknown directive type "function".

```
.. function:: unload(name)

   Delete *name* from ``sys.modules``.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1556)

Unknown directive type "function".

```
.. function:: make_legacy_pyc(source)

   Move a :pep:`3147`/:pep:`488` pyc file to its legacy pyc location and return the file
   system path to the legacy pyc file.  The *source* value is the file system
   path to the source file.  It does not need to exist, however the PEP
   3147/488 pyc file must exist.
```

A context manager to force import to return a new module reference. This is useful for testing module-level behaviors, such as the emission of a DeprecationWarning on import. Example usage:

```
with CleanImport('foo'):
    importlib.import_module('foo')  # New reference.
```

A context manager to temporarily add directories to sys.path.

This makes a copy of :data:`sys.path`, appends any directories given as positional arguments, then reverts :data:`sys.path` to the copied settings when the context ends.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1578); *backlink*

Unknown interpreted text role "data".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1578); *backlink*

Unknown interpreted text role "data".

Note that *all* :data:`sys.path` modifications in the body of the context manager, including replacement of the object, will be reverted at the end of the block.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1582); *backlink*

Unknown interpreted text role "data".

:mod:`test.support.warnings_helper` --- Utilities for warnings tests

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1587); *backlink*

Unknown interpreted text role "mod".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1590)

Unknown directive type "module".

```
.. module:: test.support.warnings_helper
   :synopsis: Support for warnings tests.
```

The :mod:`test.support.warnings_helper` module provides support for warnings tests.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1593); *backlink*

Unknown interpreted text role "mod".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1595)

Unknown directive type "versionadded".

```
.. versionadded:: 3.10
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1598)

Unknown directive type "function".

```
.. function:: check_no_resource_warning(testcase)

   Context manager to check that no :exc:`ResourceWarning` was raised.  You
   must remove the object which may emit :exc:`ResourceWarning` before the
   end of the context manager.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1605)

Unknown directive type "function".

```
.. function:: check_syntax_warning(testcase, statement, errtext='', *, lineno=1, offset=None)

   Test for syntax warning in *statement* by attempting to compile *statement*.
   Test also that the :exc:`SyntaxWarning` is emitted only once, and that it
   will be converted to a :exc:`SyntaxError` when turned into error.
   *testcase* is the :mod:`unittest` instance for the test.  *errtext* is the
   regular expression which should match the string representation of the
   emitted :exc:`SyntaxWarning` and raised :exc:`SyntaxError`.  If *lineno*
   is not ``None``, compares to the line of the warning and exception.
   If *offset* is not ``None``, compares to the offset of the exception.

   .. versionadded:: 3.8
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1619)

Unknown directive type "function".

```
.. function:: check_warnings(*filters, quiet=True)

   A convenience wrapper for :func:`warnings.catch_warnings()` that makes it
   easier to test that a warning was correctly raised.  It is approximately
   equivalent to calling ``warnings.catch_warnings(record=True)`` with
   :meth:`warnings.simplefilter` set to ``always`` and with the option to
   automatically validate the results that are recorded.

   ``check_warnings`` accepts 2-tuples of the form ``("message regexp",
   WarningCategory)`` as positional arguments. If one or more *filters* are
   provided, or if the optional keyword argument *quiet* is ``False``,
   it checks to make sure the warnings are as expected:  each specified filter
   must match at least one of the warnings raised by the enclosed code or the
   test fails, and if any warnings are raised that do not match any of the
   specified filters the test fails.  To disable the first of these checks,
   set *quiet* to ``True``.

   If no arguments are specified, it defaults to::

      check_warnings(("", Warning), quiet=True)

   In this case all warnings are caught and no errors are raised.
```

```
On entry to the context manager, a :class:`WarningRecorder` instance is
returned. The underlying warnings list from
:func:`~warnings.catch_warnings` is available via the recorder object's
:attr:`warnings` attribute.  As a convenience, the attributes of the object
representing the most recent warning can also be accessed directly through
the recorder object (see example below).  If no warning has been raised,
then any of the attributes that would otherwise be expected on an object
representing a warning will return ``None``.

The recorder object also has a :meth:`reset` method, which clears the
warnings list.

The context manager is designed to be used like this::

   with check_warnings(("assertion is always true", SyntaxWarning),
                       ("", UserWarning)):
       exec('assert(False, "Hey!")')
       warnings.warn(UserWarning("Hide me!"))

In this case if either warning was not raised, or some other warning was
raised, :func:`check_warnings` would raise an error.

When a test needs to look more deeply into the warnings, rather than
just checking whether or not they occurred, code like this can be used::

   with check_warnings(quiet=True) as w:
       warnings.warn("foo")
       assert str(w.args[0]) == "foo"
       warnings.warn("bar")
       assert str(w.args[0]) == "bar"
       assert str(w.warnings[0].args[0]) == "foo"
       assert str(w.warnings[1].args[0]) == "bar"
       w.reset()
       assert len(w.warnings) == 0


Here all warnings will be caught, and the test code tests the captured
warnings directly.

.. versionchanged:: 3.2
   New optional arguments *filters* and *quiet*.
```

Class used to record warnings for unit tests. See documentation of :func:`check_warnings` above for more details.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]test.rst, line 1687); *backlink*

Unknown interpreted text role "func".