
sharedmock Documentation

Release 0.1

RelEng Team

Aug 31, 2017

Contents:

| | | |
|----------|---------------------------------------|-----------|
| 1 | Getting started | 3 |
| 2 | Source | 5 |
| 3 | Table of Contents | 7 |
| 3.1 | sharedmock package | 7 |
| 3.1.1 | Subpackages | 7 |
| 3.1.1.1 | sharedmock.test package | 7 |
| 3.1.2 | Submodules | 7 |
| 3.1.3 | sharedmock.asserters module | 7 |
| 3.1.4 | sharedmock.mock module | 8 |
| 3.1.5 | Module contents | 8 |
| 4 | Indices and tables | 9 |
| | Python Module Index | 11 |

A multiprocessing-friendly Python mock object

CHAPTER 1

Getting started

The `SharedMock` object has an interface similar to Python's own `unittest.mock.Mock`. The main difference is that the state of a `SharedMock` object is shared among subprocesses. This allows you to easily test interactions of subprocesses with your mock instance.

```
from sharedmock.mock import SharedMock

sharedmock = SharedMock()

subprocess = mp.Process(target=sharedmock,
                        args=('fancyArg',))
subprocess.start()
subprocess.join()
subprocess.terminate()

expected_calls = [call('fancyArg')]
sharedmock.assert_has_calls(expected_calls,
                             same_order=False)
```

If the `SharedMock` were to be replaced by a `unittest.mock.Mock` in the example above, the assertion would fail. The interaction with the `unittest.mock.Mock` object would not get propagated to your test code:

```
E           AssertionError: Calls not found.
E           Expected: [call('fancyArg')]
E           Actual: []
```


CHAPTER 2

Source

The entire source code is available on [GitHub](#).

sharedmock package

Subpackages

sharedmock.test package

Submodules

sharedmock.test.test_asserters module

sharedmock.test.test_mock module

Module contents

Submodules

sharedmock.asserters module

sharedmock.asserters.**assert_calls_equal** (*expected, actual*)

Check whether the given mock object (or mock method) calls are equal and return a nicely formatted message.

sharedmock.asserters.**assert_calls_equal_unsorted** (*expected, actual*)

Raises an AssertionError if the two iterables do not contain the same items.

The order of the items is ignored

sharedmock.asserters.**raise_calls_differ_error** (*expected, actual*)

Raise an AssertionError with pretty print format for the given expected and actual mock calls in order to ensure consistent print style for better readability.

sharedmock.mock module

Module contents

CHAPTER 4

Indices and tables

- `genindex`
- `modindex`
- `search`

S

`sharedmock`, 8

`sharedmock.asserters`, 7

`sharedmock.test`, 7

A

`assert_calls_equal()` (in module `sharedmock.asserters`), 7
`assert_calls_equal_unsorted()` (in module `sharedmock.asserters`), 7

R

`raise_calls_differ_error()` (in module `sharedmock.asserters`), 7

S

`sharedmock` (module), 8
`sharedmock.asserters` (module), 7
`sharedmock.test` (module), 7