PyTest cheat sheet

Installation

Run the following on the Terminal:

`pip3 install pytest (Mac)`

Or

`pip install pytest (Windows)`

 # Nomenclature

- Add suffix 'test_' to the file that needs to be tested.
- Add suffix 'test_' to the functions to be tested.

Running pytest

This is the command that has to be executed on the Terminal prompt:

`python3 -m pytest test_file.py`

Alternative method

py.test will look for the keyword test and run the tests over those files and functions automatically.

`py.test test_file.py`

When you run pytest for a specific function add :: to run a specific function in a given file.

Flags used

For example, -v is the flag:

`python3 -m pytest abc.py -v`

Some other flag options are:

-v for verbose
-q quiet mode
-s allows the print statement inside the functions to be executed
-x is to flag the tests to stop execution after first failure
-m is used to mark a specific function
-k is a flag for searching and running tests with a specific keyword
--tb is to disable the traceback code of errors
--maxfail n specifies maximum number of test fails allowed

Tips

- The rule of thumb is that the assert statement looks for a Boolean result. You can use in, not in, is, <, >, other than == to check Boolean values.
- You can add multiple assert statements inside a single test function.

Additional reading

Fixtures

Fixtures are a type of function that is applied to functions to be tested. These functions must run before that test is executed. The purpose of fixtures is to supply data from multiple sources including URLs and databases to the test before running the test. Fixtures are used in cases where code repeats initialization.

Format:

```
@pytest.fixture
```

Markers

Markers are used to 'mark' specific functions to be executed by letting users create special names. There are many built-in markers such as xfail, xpass, skip and so on.

They follow a format such as:

```
@pytest.mark.<markername>
```

For example:

```
@pytest.mark.alpha
```

Running the specific marked test in the command line can be done with the following command:

```
pytest -m <markername> -v
```

Which will be as follows for a marker called alpha.

```
pytest -m alpha -v
```