

Unit Testing 1 cheatsheet

Testing

Test functions should begin with the word `test_`

`assert` validates if an expression is True

```
def add_two_numbers(a, b):  
    return a + b  
  
def test_add_two_numbers():  
    # Arrange  
    a = 5  
    b = 5  
    expected = 10  
  
    # Act  
    actual = add_two_numbers(a, b)  
  
    # Assert - pass  
    assert expected == actual  
  
test_add_two_numbers()
```

Pytest

Installation

```
$ pip install pytest
```

1. File names should begin or end with `test_`, as in `test_example.py` or `example_test.py`.
2. Function names should begin with `test_`. So for instance: `test_example`.
3. If tests are defined as methods on a class, the class should start with `Test`, as in `TestExample`.
4. You can run `pytest --collect-only` to see which tests `pytest` will discover, without running them.

```
# test_additions.py  
def add_two_numbers(a, b):  
    return a + b  
  
def test_add_two_numbers():  
    expected = 5  
    actual = add_two_numbers(4, 1)  
    assert expected == actual
```

Copy the code to a Python file, run `python -m pytest` and watch the output. Hopefully you should see some information about 1 test passing.

Testing Terms and Definitions

- Unit : The smallest testable chunk of code.
- TDD : Test Driven Development. The process of writing tests first.
- Happy Path : Successful test scenarios.
- Unhappy Path : Unsuccessful test scenarios.
- Corner Case : Outside normal parameters.
- Edge Case : Extreme min/max parameters.