Testing Guidelines

NJRB2 | MVS9 | FA296

This document includes the frameworks and libraries we are planning to use to assist in testing our code.

# PYTEST

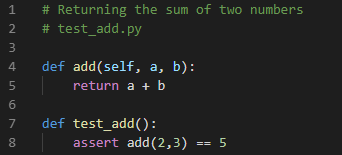
We have decided to use the ‘pytest’ framework, as it is not only the most popular, but the easiest to use.

## Pytest vs Unittest

We are aware that ‘unittest’ is available using the standard Python library, however decided on ‘pytest’ due to the below reasons:

1. Easier to write and understand the test code
2. Can store values that have failed
3. We won’t need to create classes to create functions, we can just write test functions
4. Allows tests to have less errors due to readable code

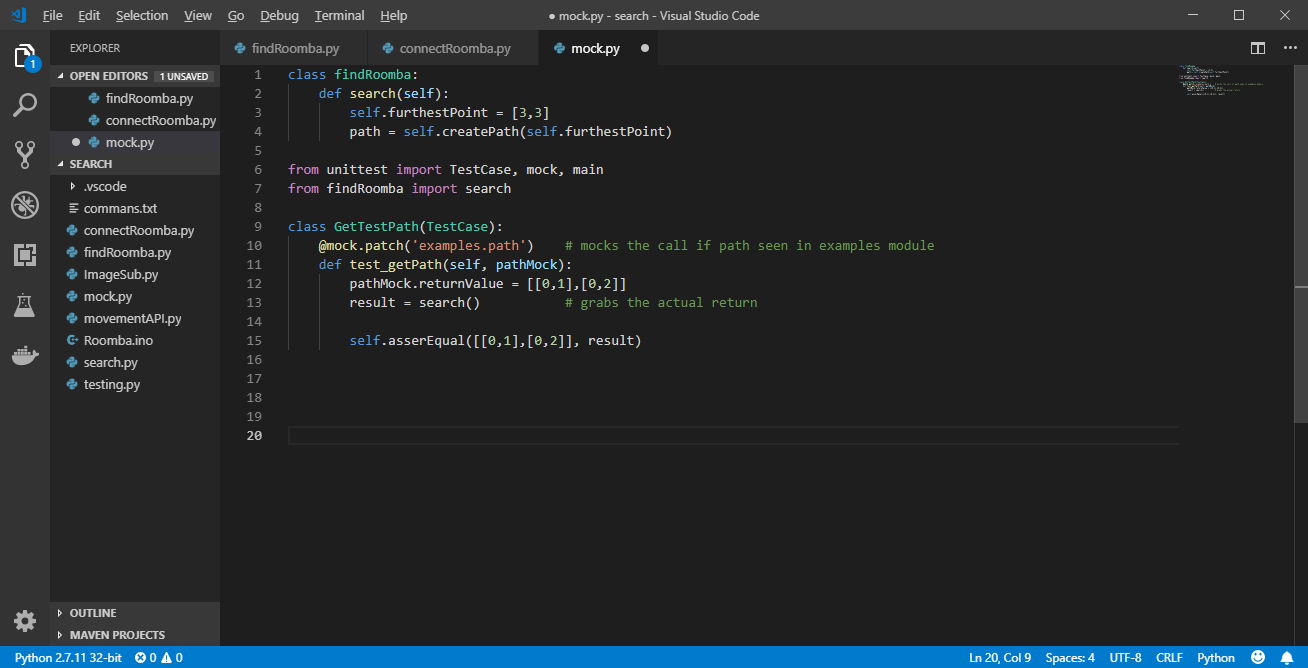
## Pytest Example

Below shows a simple example of what a pytest looks like:

# Mock Library

We have planned to use Mock alongside pytest to assist in out testing. This allows us to imitate a behaviour of an object. This means we won’t need to access the actual object and instead can use the mock object. Using Mock is beneficial as we can access one of the object’s attributes if necessary. This means we can isolate some parts of the object. Further to this, Mock is known for running tests quickly as we are not using a lot of resources.

## Mock Example

Below is a simple example of a mock test: