**Question: Which testing framework is good and why?**

For unit testing framework I will prefer “unittest” module which is also known as pyUnit. This testing framework is similar to xUnit, Junit framework and it can be easily relatable to even those who do not have any prior knowledge of python. Unit testing with python focuses mostly on testing a particular component without accessing and dependent objects in simple words I would say that it means “Test should be isolated and it should not rely on another test i.e. independent of other test”. This is often difficult to do and a bit unrealistic as code is inherently dependent on other piece of code. Different techniques can be used like stubs and mocks to separate code into “units” and run level testing on the individual pieces.

Test driven development TDD.

TDD means that you write the test before you write the code, basically the concept is that you think about what you want your code to do and write the test implementing that behaviors and watch the test fail since it doesn’t have a code to run against, and now write a code in a way that gets the test to pass and that is what Test-driven development means.

Stubs and Mocks

They are used to fake methods being tested so they think they are using a real system resource. A stub is used to fill in some dependency that is required for a unit to run correctly. So, in order to test an application component, which may be dependent on other component not yet developed you will need to write stubs, which basically are fake implementation of various components interfaces that give correct responses in cases needed to test other units. A mock on the other hand is used more for assisting in verifying that the unit did what it was designed to,