jUnit & Mockito

* Software Testing
* What is unit testing
* How does exactly it works

Software Testing 🡪

It is a process used to identify correctness, completeness and quality of our application. It includes set of activities conducted for finding errors or bugs in our applications so because our correct and complete application will b released.

For testing there are broadly four layers.

1. **Unit testing**
2. Integration testing
3. System Testing
4. Acceptance Testing – User acceptance testing (UAT)
5. **Unit testing** 🡪 It is micro-level of testing or it is first level of testing because for achieving this we will have to create / divide our application in smaller parts rater can say unit.
   1. Unit testing can be refer to any smaller unit it can be functionality, it can be program or any procedure in your application.
6. **Integration testing** 🡪 It is second level of testing, this level will be done just after unit testing. In this we will have to group our unit tests by combining them we can perform integration testing.
7. **System Testing** 🡪 It is third level of testing, It is mostly refer to whether all components of our application or our application as a system is it working perfectly fine or not. It is very important because it is going to check/test from technical, functional and business perspective.

Note-Mostly testing team is going to assign for this task.

1. **Acceptance Testing** 🡪 It is fourth level of testing, Mostly in this level we will have to test whether this application/project is ready to release or not, it means we will have to check with user’s requirements, whether all requirements are going to fulfil or not. It is also called as final level of testing because if it’s successful we can put this project for production.

**Unit Testing** 🡪

Unit testing is a way of diving our application in small units and test them. It is mostly focus on functional correctness.

**How does unit test works?**

1. Start – create our test
2. Run Test 🡪
   1. Pass, I will add it to test suit
   2. Failed, I will have to make some changes and then again run test
3. Passed then I can add it to add suit

Benefits of Unit Testing

1. Improve code quality
2. Helps to find bugs very quickly
3. Helps to design software well
4. Reduces testing cost.

Before learning jUnit we should know on which version of jUnit we are working.

Recent version of jUnit is 5

What is JUnit 5?

**JUnit 5 = JUnit Platform + JUnit Jupiter + JUnit Vintage**

* The **JUnit Platform** serves as a foundation for [launching testing frameworks](https://junit.org/junit5/docs/current/user-guide/#launcher-api) on the JVM. It also defines the [TestEngine](https://junit.org/junit5/docs/current/api/org.junit.platform.engine/org/junit/platform/engine/TestEngine.html) API for developing a testing framework that runs on the platform. Furthermore, the platform provides a [Console Launcher](https://junit.org/junit5/docs/current/user-guide/#running-tests-console-launcher) to launch the platform from the command line.
* **JUnit Jupiter** is the combination of the new [programming model](https://junit.org/junit5/docs/current/user-guide/#writing-tests) and [extension model](https://junit.org/junit5/docs/current/user-guide/#extensions) for writing tests and extensions in JUnit 5. The Jupiter sub-project provides a TestEngine for running Jupiter based tests on the platform.
* **JUnit Vintage** provides a TestEngine for running JUnit 3 and JUnit 4 based tests on the platform. It requires JUnit 4.12 or later to be present on the class path or module path.

WE have seen that unit test s mostly focus on functional correctness.

**1. Scenario ->** You are responsible for developing a calculator and using jUint you will have to test it’s functional correctness.

* 1. Addition of three numbers

2. We have to create a method which will return full name by concatenating two string

Full\_name =First\_name+” “+lastName

Bhushan Kumar

1. We will have to write a unit test for checking whether the given string is empty or not