**Junit in Java (Unit Testing)**

**Assertions in Junit:**

assertTrue assertNull fail

assertFalse assertNotNull

assertEquals assertSame

assertNotEquals assertNotSame

Example:

ArrayIntList list=new ArrayIntList();

list.add(40);

list.add(-3);

list.add(10);

list.add(13);

assertEquals(4,list.size());

|  |  |
| --- | --- |
| **Annotation** | **Description** |
| @Test | The annotated method is to be run as a test method. |
| @Before | The annotated method is to be run before EACH of the test method. |
| @After | The annotated method is to be run after EACH of the test method. |
| @BeforeClass | The annotated method is to be run ONCE before any of the test method. |
| @AfterClass | The annotated method is to be run ONCE after all the test methods. |
| @Ignore | Ignore (don't run) the test method. This is a convenient way to mark out a test method (e.g. after some code changes that fail this test.) |
| @Rule | [TODO] |

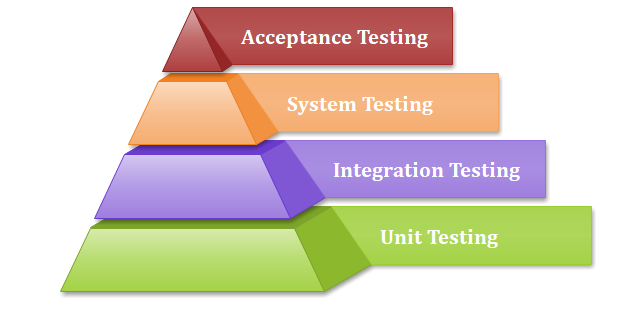
**What Is Unit Testing?**

Unit Testing is a methodology of testing source code for its fitment of use in production.

We start out writing unit tests by creating various test cases to verify the behaviors of an individual unit of source code.

**Why Unit Testing?**

In a testing level hierarchy, unit testing is the first level of testing done before integration and other remaining levels of the testing. It uses modules for the testing process which reduces the dependency of waiting for Unit testing frameworks, stubs, drivers and mock objects are used for assistance in unit testing.



## Unit Testing Tools

* NUnit
* JUnit
* PHPunit
* ParasoftJtest
* EMMA

## Advantages and disadvantages of unit testing

The pros and cons of unit testing are as follows:

### **Advantages**

* Unit testing uses module approach due to that any part can be tested without waiting for completion of another parts testing.
* The developing team focuses on the provided functionality of the unit and how functionality should look in unit test suits to understand the unit API.
* Unit testing allows the developer to refactor code after a number of days and ensure the module still working without any defect.

### **Disadvantages**

* It cannot identify integration or broad level error as it works on units of the code.
* In the unit testing, evaluation of all execution paths is not possible, so unit testing is not able to catch each and every error in a program.
* It is best suitable for conjunction with other testing activities.

## JUnit

Java provides a framework called [**JUnit**](https://www.javatpoint.com/junit-tutorial) to perform the unit testing of our Java code. In the development of **test-driven** development, JUnit is very important.

# **JUnit - Writing a Test**

https://www.tutorialspoint.com/junit/junit\_writing\_tests.htm

**Junit – Assertions and Annotations**

<https://www.tutorialspoint.com/junit/junit_using_assertion.htm> A