**What is JUnit?**

JUnit is a simple, open source framework to write and run repeatable tests. JUnit features include:

* Assertions for testing expected results.
* Test feature for sharing common test data.
* Test suites for easily organizing and running tests.
* Graphical and textual test runners.

JUnit is a unit testing framework for Java programming language. JUnit has been important in the development of test-driven development and is one of a family of unit testing frameworks collectively known as xUnit, that originated with Junit.

To run Junit test from the command line:

The **org.junit.runner.JUnitCore** class provides **runClasses()** method, this allows to run one or several tests classes/

As a return parameter we receive an object of the type org.junit.runner.Result, this object can be used to retrieve information about the tests.

* org.junit.runner.JunitCore

@BeforeClass – Run once and before all the test methods in the class, public static void

@AfterClass – Run once and after all the test methods in the class, public static void

@Before – Run before @Test, public void

@After – Run after @Test, public void

@Test – This is the test method to run, public void

@Test(timeout=5000) – which causes a test to fail if it takes longer then specified time[milliseconds]

@Ignore – to ignore test.

[ test order is based upon alphabetic order of the test cases]

**Junit assertions:**

void assertEquals(“message”, expected value, actual value);

Assets two values are equls, value can be any type.

void assertTrue(“message”, boolean condition);

Asserts that condition is true.

void assertFalse(“message”, boolean condition);

Asserts that condition is false.

void assertNotNull(“message”, java.lang.object object);

Asserts that an object is not null.

void assertNull(“message”, java.lang.object object);

Asserts that an object is null.

void assertSame(“message”, java.lang.object expected, java.lang.object expected);

Asserts that two objects refer to the same.

void assertNotSame(“message”, java.lang.object expected, java.lang.object expected));

Asserts that two objects do not refer to the same object.

void assertArraysEquals(“message”, java.lang.object object);

Asserts expected and result array are equal.

**Creating suite tests:**

A test suite is a collection of some test cases from different classes that can be run all together using **@RunWith** and **@Suite** annotations.

This is very helpful if we have many test classes

Example:

@RunWith(Suite.class)

@Suite.SuiteClasses({ className.class, className.class})

Public class SampleTest{

}

**Creating parameterized tests:**

The class is annotated with @RunWith(Parameterized.class)

The class constructor that stores the test data.

The class has a static method that generates and returns test data and is annotated with @Parameters annotation.

The class has a test with a method annotated with the @Test.

**Note: Junit 5,** we can **order test execution** by using **@Order(1)** and can use **@OrderAll** annotations.