**JUnit Notes**

**Junit:**

[JUnit](http://junit.org/) is a test framework which uses annotations to identify methods that specify a test. JUnit is an open source project hosted at [Github](https://github.com/junit-team/junit).

[**How to define a test in JUnit?**](http://www.vogella.com/tutorials/JUnit/article.html#unittesting_junit_test)

A JUnit test is a method contained in a class which is only used for testing. This is called a Test class. To define that a certain method is a test method, annotate it with the @Test annotation.

This method executes the code under test. You use an assertmethod, provided by JUnit or another assert framework, to check an expected result versus the actual result. These method calls are typically called assertsor assert statements.

## [Defining test methods](http://www.vogella.com/tutorials/JUnit/article.html#usingjunit_annotations)

JUnit uses annotations to mark methods as test methods and to configure them.

import org.junit.\*- Import statement for using the following annotations.

@Test- Identifies a method as a test method.

@Before- Executed before each test. It is used to prepare the test environment (e. class).

@After- Executed after each test. It is used to cleanup the test environment (e.g. defaults).

@BeforeClass- Executed once, before the start of all tests. It is used to perform time in connect to a database. Methods marked with this annotation need to be JUnit.

@AfterClass- Executed once, after all tests have been finished. It is used to perform c disconnect from a database. Methods annotated with this annotation ne with JUnit.

@Ignore or @Ignore("Why disabled")-Marks that the test should be disabled. This is useful when the underly test case has not yet been adapted. Or if the execution time of this test practice to provide the optional description, why the test is disabled

@Test(timeout=100)- Fails if the method takes longer than 100 milliseconds

## [Assert statements](http://www.vogella.com/tutorials/JUnit/article.html#usingjunit_asserts)

JUnit provides static methods to test for certain conditions via the Assert class. These assert statementstypically start with assert. They allow you to specify the error message, the expected and the actual result. An *assertion method* compares the actual value returned by a test to the expected value. It throws an AssertionException if the comparison fails.