Annotations for Junit testing:

**1. @Test:** It is used to specify the test method.

**2. @BeforeClass:** It is used to specify that method will be called only once, before starting all the test cases.

**3. @AfterClass:** It is used to specify that method will be called only once, after finishing all the test cases.

**4. @Before:** It is used to specify that method will be called before each test case.

**5. @After:** It is used to specify that method will be called after each test case.

**6. @Ignore:** It is used to ignore the test case.

In Junit, test suite allows us to aggregate all test cases from multiple classes in one place and run it together.

To run the suite test, you need to annotate a class using below-mentioned annotations:

1. @Runwith(Suite.class)
2. @SuiteClasses(test1.class,test2.class……) or

@Suite.SuiteClasses ({test1.class, test2.class……})

With above annotations, all the test classes in the suite will start

import org.junit.After;

import org.junit.AfterClass;

import org.junit.Before;

import org.junit.BeforeClass;

import org.junit.Ignore;

import org.junit.Test;

public class DivisionTestCase {

//called only once, before starting all the test cases.

@BeforeClass

public static void beforeClass() {

System.out.println("In beforeClass method.");

}

//called only once, after finishing all the test cases.

@AfterClass

public static void afterClass() {

System.out.println("In afterClass method");

}

//called before each test case.

@Before

public void before() {

System.out.println("In before method");

}

//called after each test case.

@After

public void after() {

System.out.println("In after method");

}

//define the test case.

@Test

public void testCase1() {

System.out.println("In testCase1");

}

//define the test case.

@Test

public void testCase2() {

System.out.println("In testCase2");

}

//ignore the test case.It will not execute.

@Ignore

@Test

public void testCase3() {

System.out.println("In testCase3");

}

}

**DivisionTestCase.java**

import com.w3spoint.business.\*;

import static org.junit.Assert.\*;

import org.junit.Test;

public class DivisionTestCase {

//DivisionTest class objects

DivisionTest divisionTest1 = new DivisionTest(10, 2);

DivisionTest divisionTest2 = new DivisionTest(10, 0);

//Test case for division

@Test

public void test() {

assertEquals(5, divisionTest1.division());

}

//Test case for expected ArithmeticException,

//As in this case ArithmeticException

// is the expected exception so JUnit

//will pass this unit test.

@Test(expected = ArithmeticException.class)

public void testException() {

assertEquals(5, divisionTest2.division());

}

}

**DivisionTest.java**

public class DivisionTest {

//data members

int num1, num2;

//parameterised constructor

public DivisionTest(int num1, int num2){

this.num1 = num1;

this.num2 = num2;

}

//division method

public int division() throws ArithmeticException{

return num1/num2;

}

}

TestSuite.java

import org.junit.runner.RunWith;

import org.junit.runners.Suite;

@RunWith(Suite.class)

@Suite.SuiteClasses({

DivisionTestCase1.class,

DivisionTestCase2.class

})

public class TestSuite {

}

**DivisionTestCase1.java**

import com.w3spoint.business.\*;

import static org.junit.Assert.\*;

import org.junit.Test;

public class DivisionTestCase1 {

//DivisionTest class objects

DivisionTest divisionTest1 = new DivisionTest(10, 2);

DivisionTest divisionTest2 = new DivisionTest(10, 0);

//Test case for division

@Test

public void test() {

System.out.println("In DivisionTestCase1.test");

assertEquals(5, divisionTest1.division());

}

//Test case for expected ArithmeticException,

//As in this case ArithmeticException

// is the expected exception so JUnit

//will pass this unit test.

@Test(expected = ArithmeticException.class)

public void testException() {

System.out.println("In DivisionTestCase1.testException");

assertEquals(5, divisionTest2.division());

}

}

**DivisionTestCase2.java**

import com.w3spoint.business.\*;

import static org.junit.Assert.\*;

import org.junit.Test;

public class DivisionTestCase2 {

//DivisionTest class objects

DivisionTest divisionTest = new DivisionTest(10, 5);

//Test case for division

@Test

public void test() {

System.out.println("In DivisionTestCase2.test");

assertEquals(2, divisionTest.division());

}

}

**DivisionTest.java**

public class DivisionTest {

//data members

int num1, num2;

//parameterised constructor

public DivisionTest(int num1, int num2){

this.num1 = num1;

this.num2 = num2;

}

//division method

public int division() throws ArithmeticException{

return num1/num2;

}

}

**TestRunner.java**

import org.junit.runner.JUnitCore;

import org.junit.runner.Result;

import org.junit.runner.notification.Failure;

public class TestRunner {

public static void main(String[] args) {

Result result = JUnitCore.runClasses(DivisionTestCase.class);

for (Failure failure : result.getFailures()) {

System.out.println(failure.toString());

}

System.out.println(result.wasSuccessful());

}

}