# JUnit Testing

Exercise 1: Setting Up Junit:

## JunitTest1.java:

package sample;

public class JunitTest {

public int add(int a, int b) {

return a + b;

}

}

## JunitTestCase.java:

package sample;

import static org.junit.Assert.\*;

import org.junit.Test;

public class JunitTestCase {

@Test

public void test() {

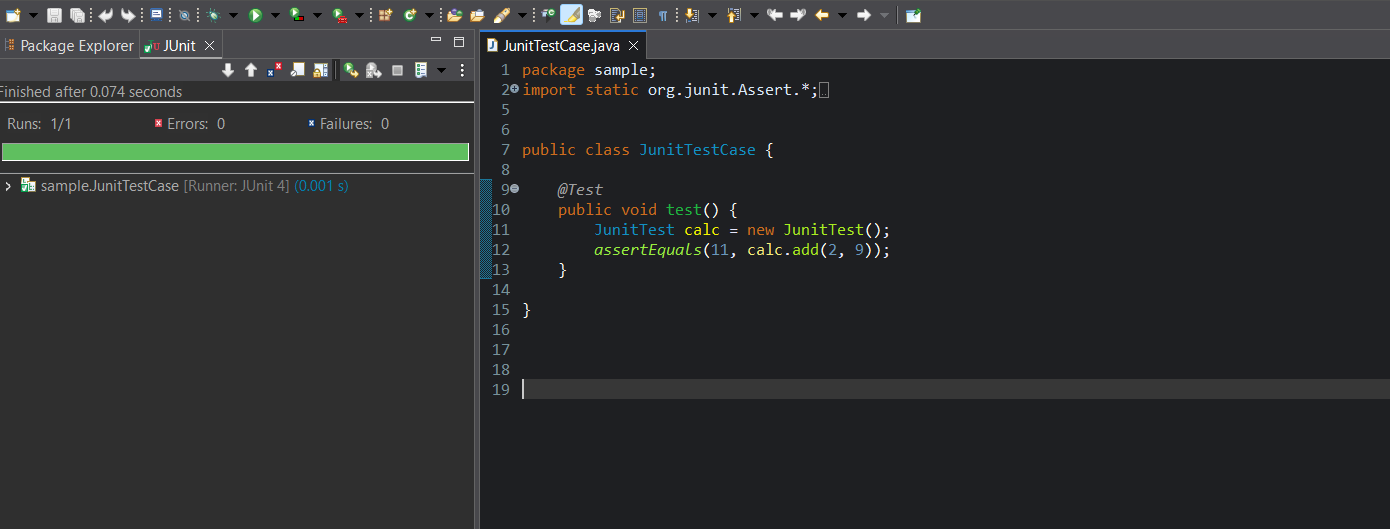
JunitTest calc = new JunitTest();

assertEquals(11, calc.add(2, 9));

}

}

## Output:



Exercise 3: Assertions in Junit:

## JunitExercise3TestCase.java:

package sample;

import static org.junit.Assert.\*;

import org.junit.Test;

public class JunitExercise3TestCase {

@Test

public void test() {

assertEquals(5,2+3);

assertTrue(5 > 3);

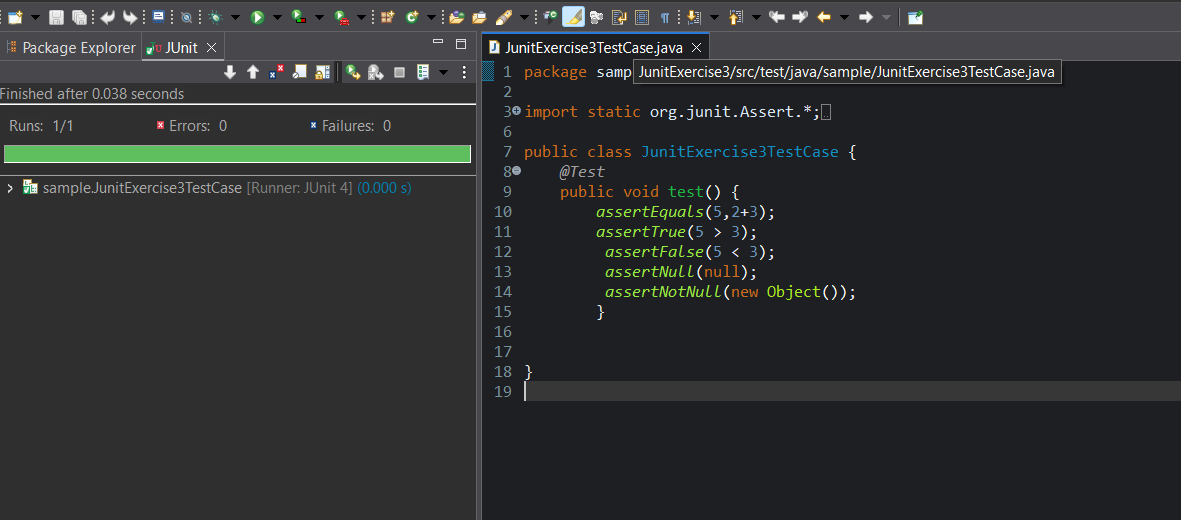
assertFalse(5 < 3);

assertNull(null);

assertNotNull(new Object());

} }

## Output:



Exercise 4: Arrange-Act-Assert (AAA) Pattern, Test Fixtures, Setup and Teardown Methods in Junit:

## Printer.java:

package sample;

public class Printer {

public String welcome() {

return "Welcome!";

}

public String goodbye() {

return "Goodbye!";

}

}

## PrinterTest.java:

package sample;

import org.junit.Before;

import org.junit.After;

import org.junit.Test;

import static org.junit.Assert.\*;

public class Printertest {

private Printer printer;

@Before

public void setUp() {

printer = new Printer();

System.out.println("Setup done.");

}

@After

public void tearDown() {

printer = null;

System.out.println("Teardown done.");

}

@Test

public void testWelcome() {

String result = printer.welcome();

assertEquals("Welcome!", result);

}

@Test

public void testGoodbye() {

String result = printer.goodbye();

assertEquals("Goodbye!", result);

}

}

## Output:

