**Ques 1 : Setting Up Junit**

Code :

**--Pom.xml(MavenProject) :**

<?xml version="1.0" encoding="UTF-8"?>  
<project xmlns="http://maven.apache.org/POM/4.0.0"  
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
 xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">  
 <modelVersion>4.0.0</modelVersion>  
  
 <groupId>Junit.project</groupId>  
 <artifactId>MavenProject</artifactId>  
 <version>1.0-SNAPSHOT</version>  
  
 <properties>  
 <maven.compiler.source>22</maven.compiler.source>  
 <maven.compiler.target>22</maven.compiler.target>  
 <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>  
 </properties>  
 <dependencies>  
 <dependency>  
 <groupId>junit</groupId>  
 <artifactId>junit</artifactId>  
 <version>4.13.2</version>  
 <scope>test</scope>  
 </dependency>  
 <dependency>  
 <groupId>junit</groupId>  
 <artifactId>junit</artifactId>  
 <version>4.13.2</version>  
 <scope>compile</scope>  
 </dependency>  
 </dependencies>  
  
</project>

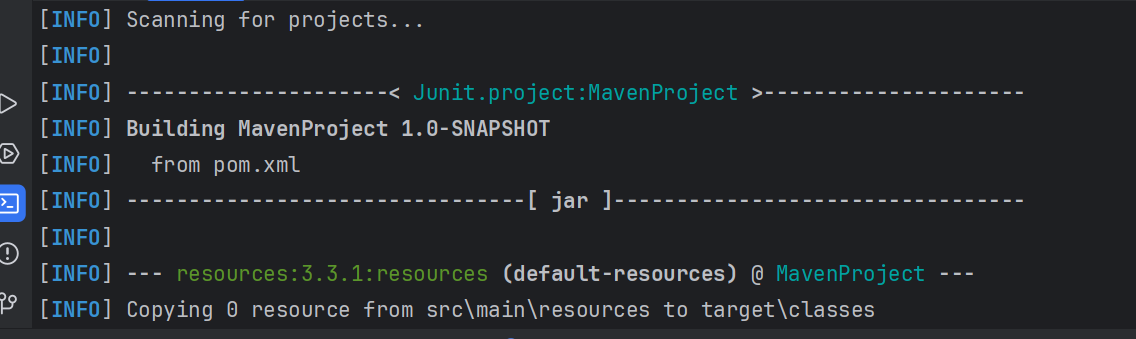
**--Calculator.java**

package Junit.project;  
  
public class Calculator {  
 public int add(int a, int b) {  
 return a + b;  
 }  
}

**--CalculatorTest.java**

package Junit.project;  
  
import org.junit.Test;  
import static org.junit.Assert.\*;  
  
public class CalculatorTest {  
  
 @Test  
 public void testAdd() {  
 Calculator calc = new Calculator();  
 int result = calc.add(2, 3);  
 *assertEquals*(5, result);  
 }  
}

Output :



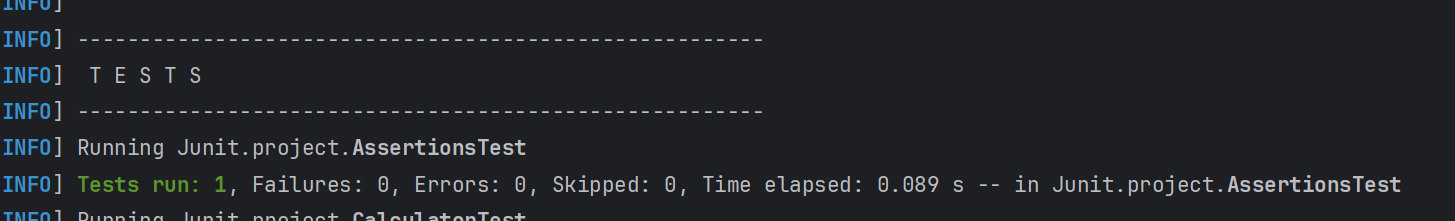
**Ques 2 : Assertions in Junit**

Code :

**--AssertionsTest.java**

package Junit.project;  
  
import org.junit.Test;  
import static org.junit.Assert.\*;  
  
public class AssertionsTest {  
 @Test  
 public void testAssertions() {  
 // Assert equals  
 *assertEquals*(5, 2 + 3); // Passes if 2 + 3 equals 5  
  
 // Assert true  
 *assertTrue*(5 > 3); // Passes if condition is true  
  
 // Assert false  
 *assertFalse*(5 < 3); // Passes if condition is false  
  
 // Assert null  
 *assertNull*(null); // Passes if value is null  
  
 // Assert not null  
 *assertNotNull*(new Object()); // Passes if value is NOT null  
 }  
}

Output :



**Ques 3 : Arrange-Act-Assert (AAA) Pattern, Test Fixtures, Setup and Teardown Methods in Junit.**

Code :

**--CalculatorWithFixtureTest.java**

package Junit.project;  
  
import org.junit.After;  
import org.junit.Before;  
import org.junit.Test;  
import static org.junit.Assert.\*;  
  
public class CalculatorWithFixtureTest {  
  
 private Calculator calculator;  
  
 // Setup method: runs before each test  
 @Before  
 public void setUp() {  
 calculator = new Calculator(); // Arrange  
 System.*out*.println("Setup: Calculator initialized.");  
 }  
  
 // Teardown method: runs after each test  
 @After  
 public void tearDown() {  
 calculator = null;  
 System.*out*.println("Teardown: Calculator cleaned up.");  
 }  
  
 @Test  
 public void testAdd() {  
 // Act  
 int result = calculator.add(2, 3);  
  
 // Assert  
 *assertEquals*(5, result);  
 }  
  
 @Test  
 public void testSubtract() {  
 int result = calculator.subtract(7, 2);  
 *assertEquals*(5, result);  
 }  
  
 @Test  
 public void testMultiply() {  
 int result = calculator.multiply(3, 4);  
 *assertEquals*(12, result);  
 }  
  
 @Test  
 public void testDivide() {  
 double result = calculator.divide(10, 2);  
 *assertEquals*(5.0, result, 0.001);  
 }  
}

**--Calculator.java**

package Junit.project;  
  
public class Calculator {  
 public int add(int a, int b) {  
 return a + b;  
 }  
 public int subtract(int a, int b) {  
 return a - b;  
 }  
 public int multiply(int a, int b) {  
 return a \* b;  
 }  
 public double divide(int a, int b) {  
 return a / b;  
 }  
}

Output :

