**1. Setting Up Junit**:

**ProjectName: Calculator**

**Code:**

**pom.xml**

<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>com.cts</groupId>

    <artifactId>calculator</artifactId>

    <version>1.0-SNAPSHOT</version>

    <dependencies>

        <dependency>

            <groupId>junit</groupId>

            <artifactId>junit</artifactId>

            <version>4.13.2</version>

            <scope>test</scope>

        </dependency>

    </dependencies>

</project>

**Calculator.java**

public class Calculator {

public int add(int a, int b) {

return a + b;

}

public int subtract(int a, int b) {

return a - b;

}

}

**CalculatorTest.java**

import org.junit.Test;

import static org.junit.Assert.\*;

public class CalculatorTest {

@Test

public void testAddition() {

Calculator calc = new Calculator();

int result = calc.add(10, 5);

assertEquals(15, result);

}

@Test

public void testSubtraction() {

Calculator calc = new Calculator();

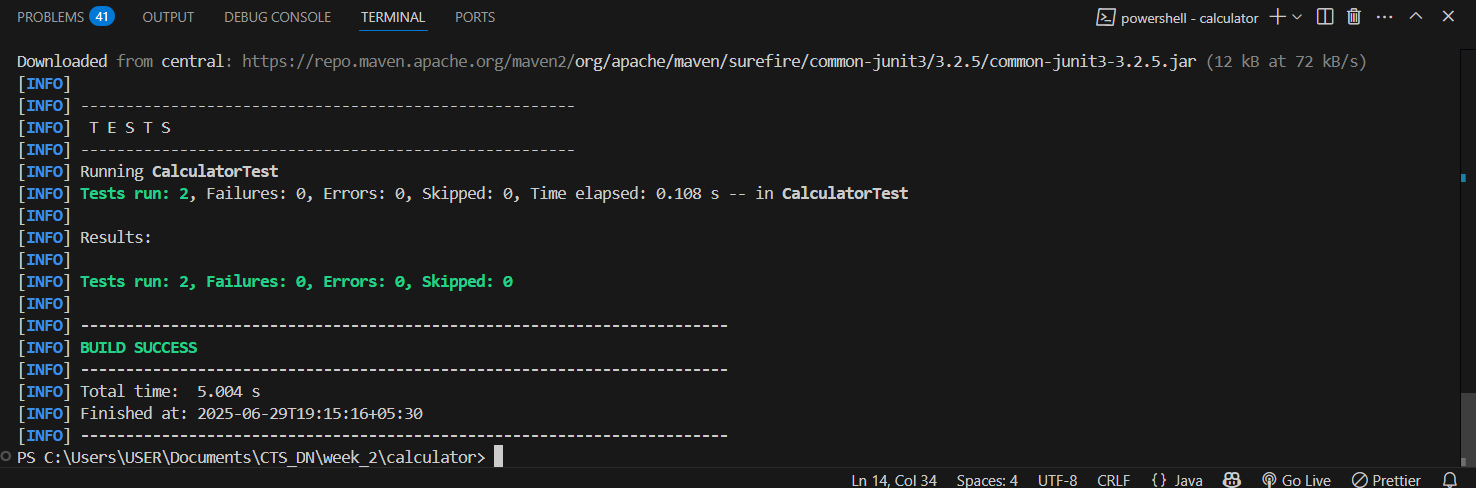
int result = calc.subtract(10, 3);

assertEquals(7, result);

}

}

**Output:**

****

**2. Assertions in Junit:**

**Code:**

**AssertionsTest.java**

import org.junit.Test;

import static org.junit.Assert.\*;

public class AssertionsTest {

@Test

public void testVariousAssertions() {

int actualSum = 2 + 3;

int expectedSum = 5;

assertEquals(expectedSum, actualSum);

boolean isGreater = 5 > 3;

assertTrue(isGreater);

boolean isLess = 5 < 3;

assertFalse(isLess);

String userName = null;

assertNull(userName);

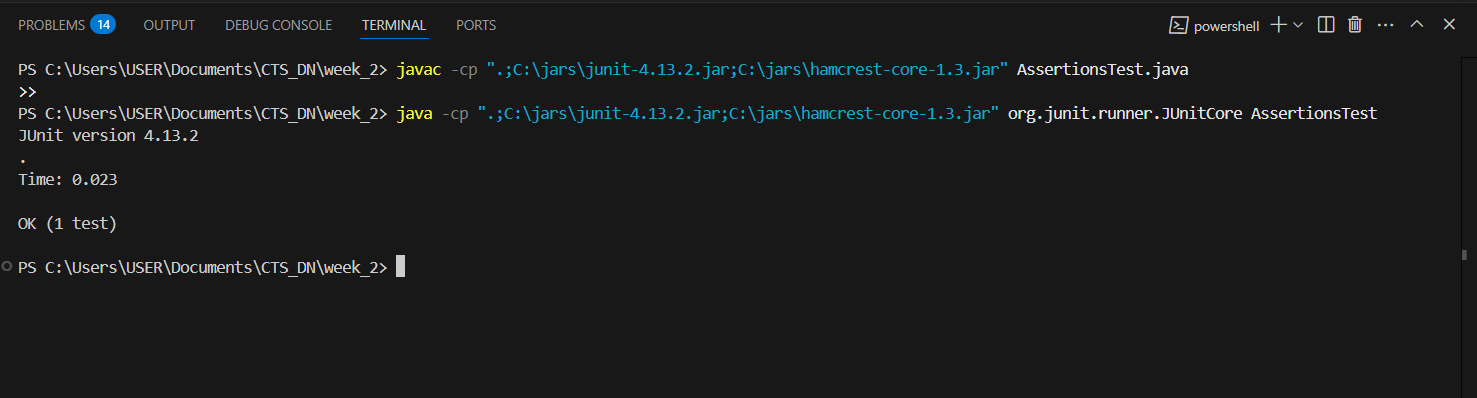
Object student = new Object();

assertNotNull(student);

}

}

**Output:**

****

**3. Arrange-Act-Assert (AAA) Pattern, Test Fixtures, Setup and**

**Teardown Methods in Junit**

**ProjectName: JUnitBasics**

**Code:**

**Calculator.java**

public class Calculator {

public int multiply(int a, int b) {

return a \* b;

}

public int divide(int numerator, int denominator) {

return numerator / denominator;

}

}

**CalculatorTest.java**

import org.junit.Before;

import org.junit.After;

import org.junit.Test;

import static org.junit.Assert.\*;

public class CalculatorTest {

private Calculator calculator;

@Before

public void initializeCalculator() {

calculator = new Calculator();

}

@After

public void cleanUpCalculator() {

calculator = null;

}

@Test

public void testMultiplication() {

int firstNumber = 4;

int secondNumber = 5;

int actualResult = calculator.multiply(firstNumber, secondNumber);

int expectedResult = 20;

assertEquals(expectedResult, actualResult);

}

@Test

public void testDivision() {

int numerator = 10;

int denominator = 2;

int actualResult = calculator.divide(numerator, denominator);

int expectedResult = 5;

assertEquals(expectedResult, actualResult);

}

}

**Output:**

