**JAVA FSE WEEK 2**

**Junit\_Basic\_Testing**

**Mandatory Hands-On Exercises**

**Exercise 1: Setting Up Junit**

**Senario:** You need to set up JUnit in your Java project to start writing unit tests.

SampleTest.java:

package example;

import org.junit.Test;

import static org.junit.Assert.\*;

public class SampleTest {

@Testpublic void testAddition() {

        int a = 2;

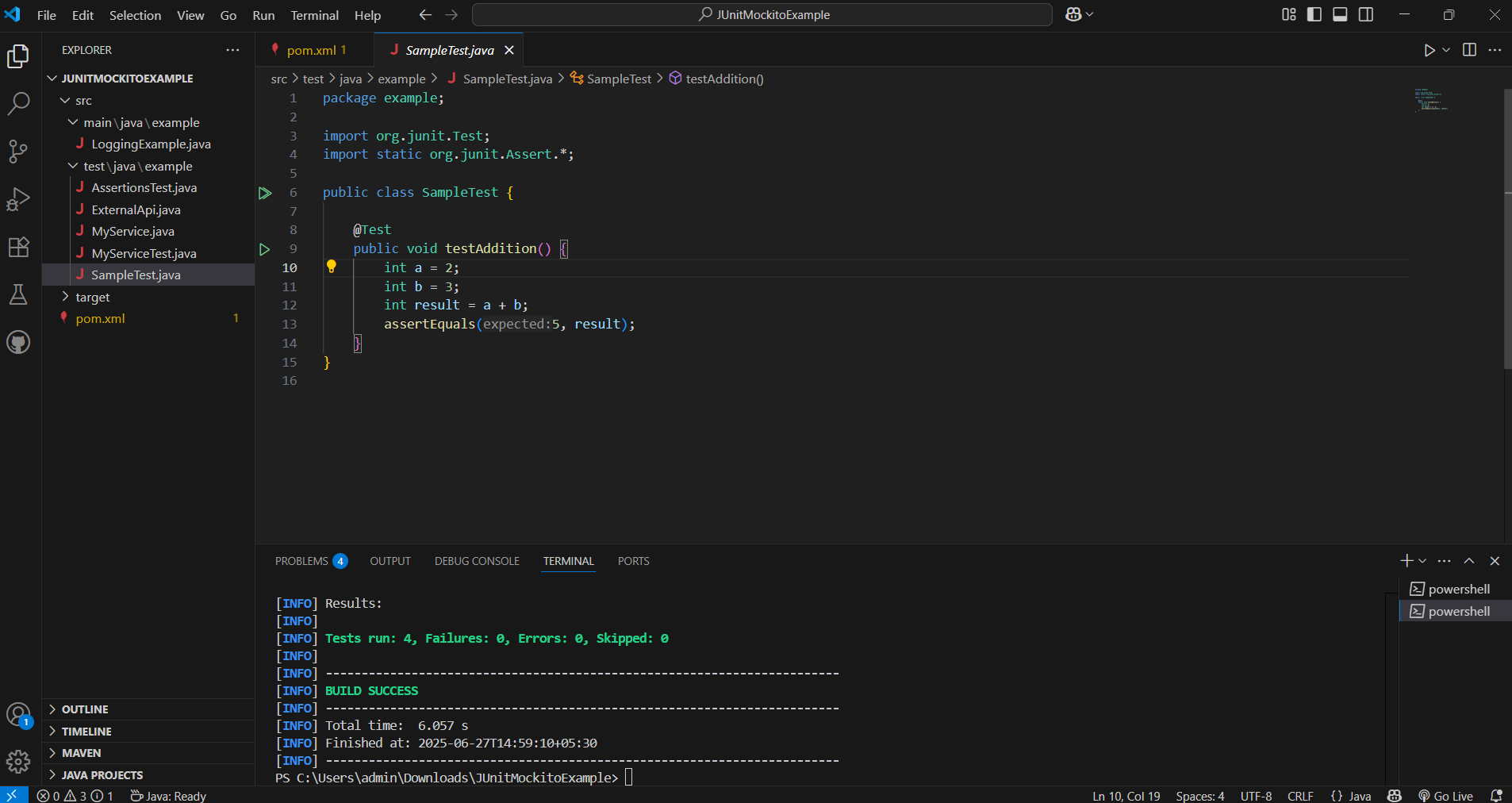
        int b = 3;

        int result = a + b;

        assertEquals(5, result);

    }

}

**OUTPUT:**

**Exercise 3: Assertions in JUnit**

**Scenario:** You need to use different assertions in JUnit to validate your test results

AssertionsTest.java:

package example;

import org.junit.Test;

import static org.junit.Assert.\*;

public class AssertionsTest {

@Test

public void testAssertions() {

// Assert equals

assertEquals(5, 2 + 3);

// Assert true

assertTrue(5 > 3);

// Assert false

assertFalse(5 < 3);

// Assert null

assertNull(null);

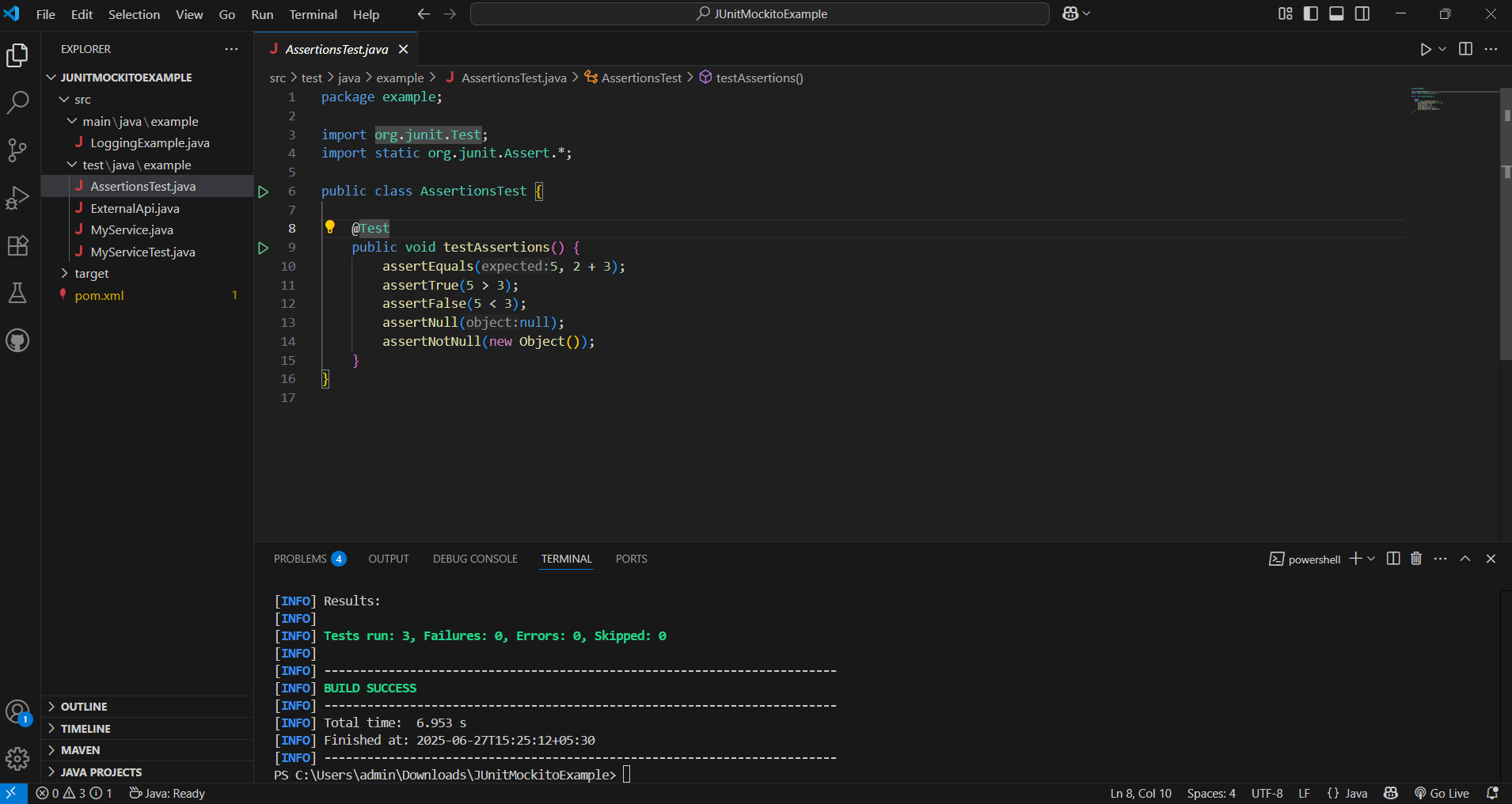
// Assert not null

assertNotNull(new Object());

}

}

**OUTPUT:**

****

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

**Scenario:** You need to organize your tests using the Arrange-Act-Assert (AAA) pattern and use setup and teardown methods

CalculatorTest.java:

package example;

import org.junit.After;

import org.junit.Before;

import org.junit.Test;

import static org.junit.Assert.\*;

// Class to be tested

class Calculator {

public int add(int a, int b) {

return a + b;

}

public int multiply(int a, int b) {

return a \* b;

}

}

public class CalculatorTest {

private Calculator calculator;

// 🔧 Setup before each test

@Before

public void setUp() {

calculator = new Calculator(); // Arrange

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

}

// 🧹 Cleanup after each test

@After

public void tearDown() {

calculator = null;

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

}

@Test

public void testAddition() {

// Act

int result = calculator.add(2, 3);

// Assert

assertEquals(5, result);

}

@Test

public void testMultiplication() {

// Act

int result = calculator.multiply(4, 5);

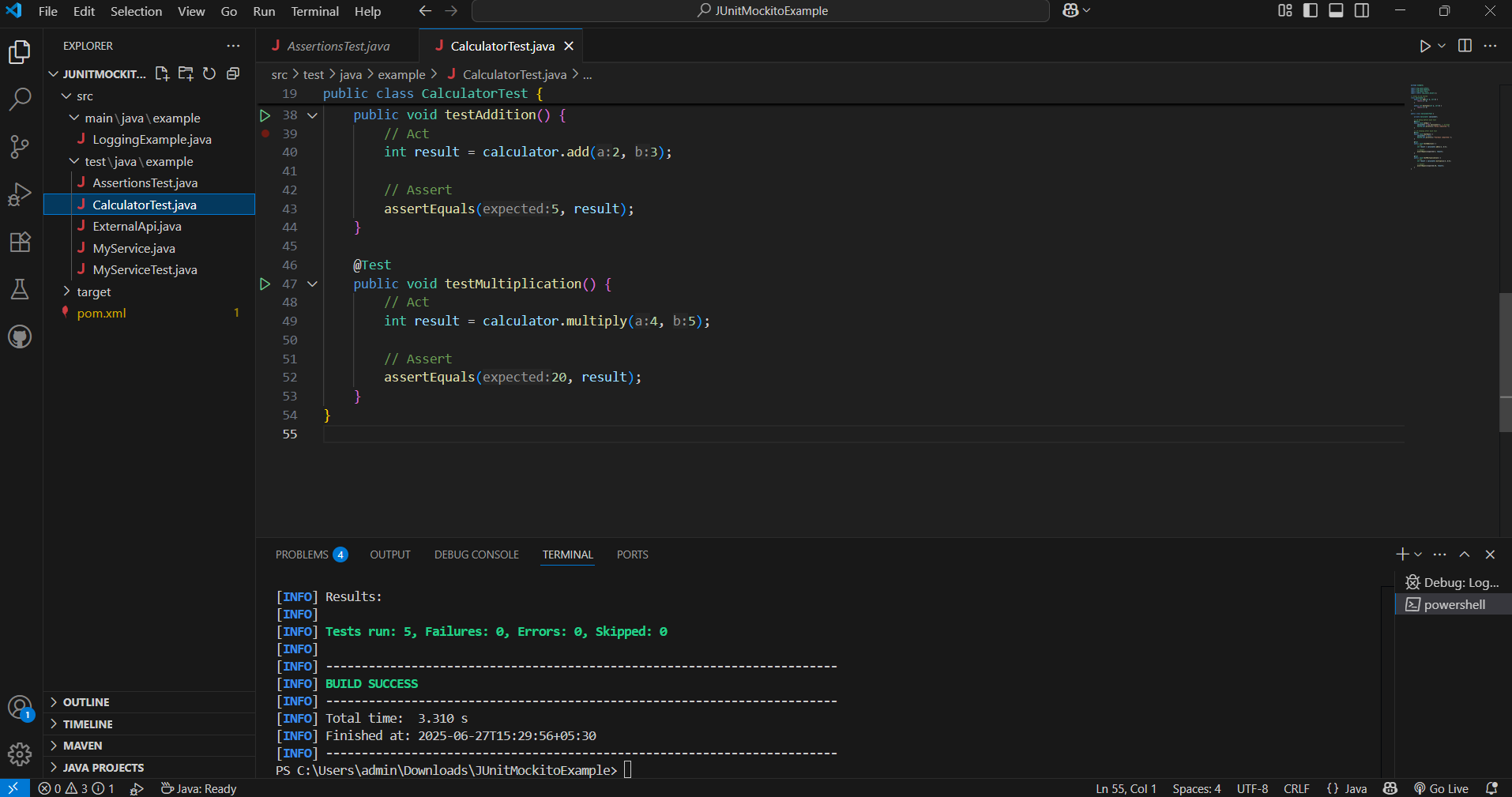
// Assert

assertEquals(20, result);

}

}

**OUTPUT:**

****