**TDD using Junit5 and Mockito**

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

**CODE**

// Calculator.java

package com.example;

public class Calculator {

public int add(int a, int b) {

return a + b;

}

public int divide(int a, int b) {

if (b == 0) throw new ArithmeticException("Cannot divide by zero");

return a / b;

}

}

// CalculatorTest.java

package com.example;

import org.junit.After;

import org.junit.Before;

import org.junit.Test;

import static org.junit.Assert.\*;

public class CalculatorTest {

private Calculator calculator;

@Before

public void setUp() {

System.out.println("Setting up Calculator instance...");

calculator = new Calculator();

}

@After

public void tearDown() {

System.out.println("Tearing down Calculator instance...");

calculator = null;

}

@Test

public void testAddition() {

// Arrange

int a = 10;

int b = 5;

// Act

int result = calculator.add(a, b);

// Assert

assertEquals(15, result);

}

@Test

public void testDivision() {

// Arrange

int a = 20;

int b = 4;

// Act

int result = calculator.divide(a, b);

// Assert

assertEquals(5, result);

}

@Test(expected = ArithmeticException.class)

public void testDivisionByZero() {

// Arrange

int a = 10;

int b = 0;

// Act

calculator.divide(a, b);

// Assert (handled by @Test expected)

}

}

**OUTPUT**

