**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.

Steps:

1. Write tests using the AAA pattern.

2. Use @Before and @After annotations for setup and teardown methods.

**Source Code**

**File: Calculator.java**

**package** myproject.calci;

**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** **int** divide(**int** a, **int** b) {

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

**return** a / b;

}

}

**File: CalculatorTest.java**

**package** myproject.calci;

**import** **static** org.junit.Assert.\*;

**import** org.junit.After;

**import** org.junit.Before;

**import** org.junit.Test;

**public** **class** CalculatorTest {

**private** Calculator calculator;

@Before

**public** **void** setUp() {

calculator = **new** Calculator();

System.***out***.println("Setup for test executed.");

}

@After

**public** **void** tearDown() {

calculator = **null**;

System.***out***.println("Teardown after test executed.\n");

}

@Test

**public** **void** testAddition() {

**int** a = 10;

**int** b = 5;

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

*assertEquals*(15, result);

System.***out***.println("Addition Result: " + result);

}

@Test

**public** **void** testSubtraction() {

**int** a = 10;

**int** b = 5;

**int** result = calculator.subtract(a, b);

*assertEquals*(5, result);

System.***out***.println("Subtraction Result: " + result);

}

@Test

**public** **void** testMultiplication() {

**int** a = 4;

**int** b = 5;

**int** result = calculator.multiply(a, b);

*assertEquals*(20, result);

System.***out***.println("Multiplication Result: " + result);

}

@Test

**public** **void** testDivision() {

**int** a = 20;

**int** b = 4;

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

*assertEquals*(5, result);

System.***out***.println("Division Result: " + result);

}

@Test(expected = IllegalArgumentException.**class**)

**public** **void** testDivisionByZero() {

**int** a = 10;

**int** b = 0;

calculator.divide(a, b);

System.***out***.println("Division by zero test passed with exception.");

}

}

**Output:**

