**Exercise 1**: Setting Up JUnit Scenario: You need to set up JUnit in your Java project to start writing unit tests.

Public class Test{

}

**Exercise 3**: Assertions in Junit

**import** **static** org.junit.jupiter.api.Assertions.\*; // JUnit 5 Assertions

**import** org.junit.jupiter.api.Test; // JUnit 5 Test

**public** **class** AssertionsTest {

@Test

**public** **void** testAssertions() {

assertEquals(5, 2 + 3);

assertTrue(5 > 3);

assertFalse(5 < 3);

assertNull(**null**);

assertNotNull(**new** Object());

}

}

Output:

Tests run: 1, Failures: 0, Errors: 0, Skipped: 0

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

public class Calculator {

public int add(int a, int b) {

return a + b;

}

public int subtract(int a, int b) {

return a - b;

}

}

import static org.junit.Assert.\*;

import org.junit.Before;

import org.junit.After;

import org.junit.Test;

public class CalculatorTest {

private Calculator calculator;

// Setup method: runs BEFORE each test

@Before

public void setUp() {

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

calculator = new Calculator();

}

// Teardown method: runs AFTER each test

@After

public void tearDown() {

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

calculator = null;

}

@Test

public void testAddition() {

// Arrange

int a = 5;

int b = 3;

// Act

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

// Assert

assertEquals(8, result);

}

@Test

public void testSubtraction() {

// Arrange

int a = 5;

int b = 3;

// Act

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

// Assert

assertEquals(2, result);

}

}

**Output:**

Setting up Calculator instance...

Cleaning up Calculator instance...

Setting up Calculator instance...

Cleaning up Calculator instance...