**Exercise3: Arrange – Act – Assert (AAA) Pattern, Setup and Teardown in Junit**

**Objective:** To understand how to organize unit tests using the AAA pattern and utilize setup..

(@Before) and teardown(@After) methods in Junit.

Key Concepts:

1. **AAA(Arrange-Act-Assert) Pattern**

|  |  |
| --- | --- |
| **Step** | **Purpose** |
| Arrange | Prepare objects and set initial state |
| Act | Perform the operation to be tested |
| Assert | Check the result is as expected |

1. **@Before Annotation:**

Runs before each test. Used to initialize objects or resources.

1. **@After Annotation:**

Runs after each test. Used to clean up or reset state.

Create a class under Test Calculator.java:

|  |
| --- |
| package harshita;  public class Calculator {  public int add (int a, int b) {  return a + b;  }  public int subtract (int a, int b) {  return a - b;  }  } |

Create a Test Class with AAA:

|  |
| --- |
| package harshita;  import org.junit.After;  import org.junit.Before;  import org.junit.Test;  import static org.junit.Assert.assertEquals;  public class CalculatorTest {  private Calculator calculator;  // Setup: Runs before each test  @Before  public void setUp() {  System.out.println("Setting up...");  calculator = new Calculator (); // Arrange  }  // Teardown: Runs after each test  @After  public void teardown () {  System.out.println("Cleaning up...");  calculator = null;  }  @Test  public void testAdd() {  // Act  int result = calculator.add(3, 7);  // Assert  assertEquals(10, result);  }  @Test  public void testSubtract() {  // Act  int result = calculator.subtract(10, 4);  // Assert  assertEquals(6, result);  }  } |

Expected output:

