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

//Calculator.java

package myapp;

public class Calculator {

private int total;

public void add(int value) {

total += value;

}

public void reset() {

total = 0;

}

public int getTotal() {

return total;

}

}

**//CalculatorTest.java**

package myapptest;

import org.junit.Before;

import org.junit.After;

import org.junit.Test;

import myapp.Calculator;

import static org.junit.Assert.\*;

public class CalculatorTest{

private Calculator calculator;

*@Before*

public void setUp() {

calculator = new Calculator();

}

*@After*

public void tearDown() {

calculator.reset();

}

*@Test*

public void testAddition() {

// Act

calculator.add(5);

calculator.add(3);

// Assert

*assertEquals*("Sum should be 8", 8, calculator.getTotal());

}

*@Test*

public void testReset() {

// Arrange

calculator.add(10);

// Act

calculator.reset();

// Assert

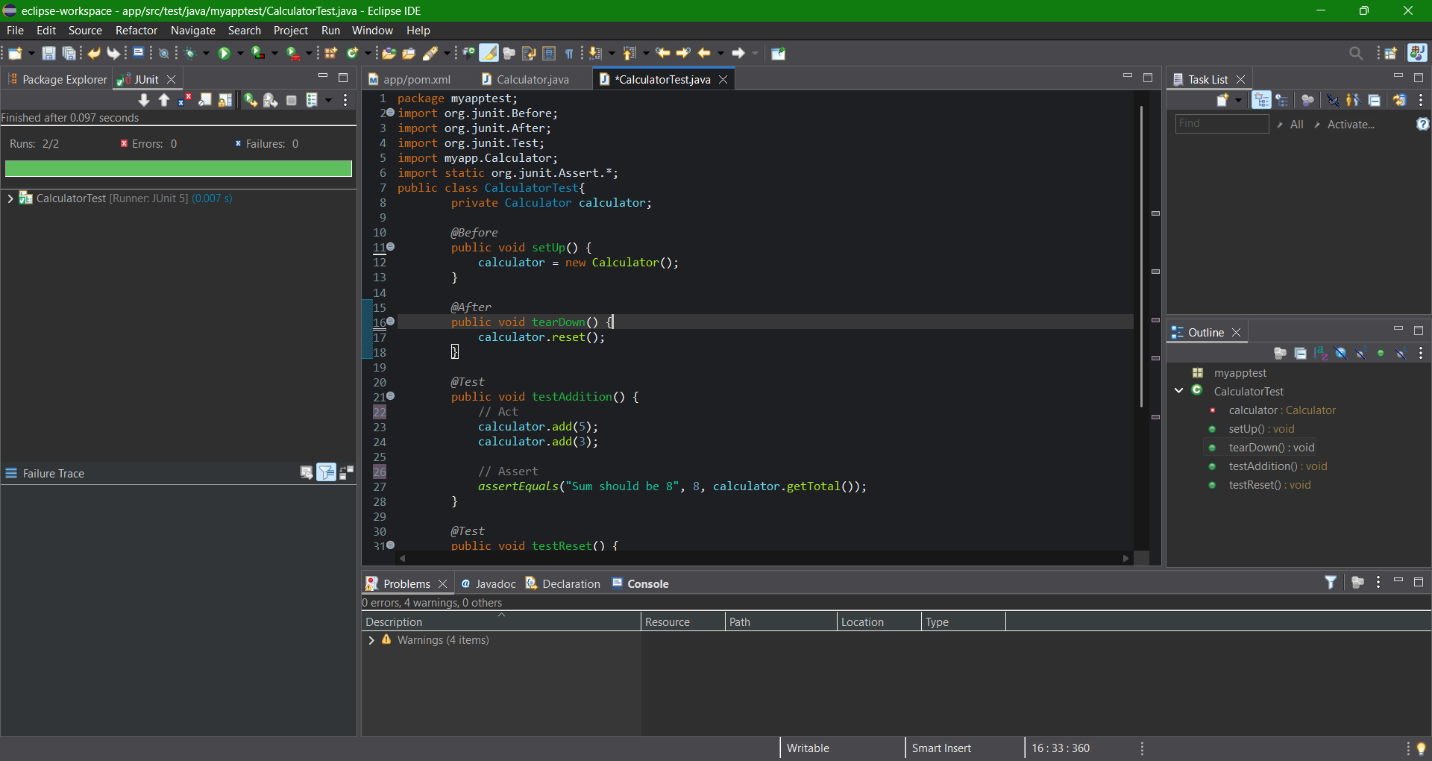
*assertEquals*("Total should be zero after reset", 0, calculator.getTotal());

*assertTrue*("Total should equal 0", calculator.getTotal() == 0);

}

}

**Output:**

****