**Week 2\_JUnit Testing Exercises\_HandsOn**

**Exercise 1: Setting Up JUnit**

**Calculator.java**

package junit;

public class Calculator {

public int add(int a, int b) {

return a + b;

}

}

**CalculatorTest.java**

package junit;

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

import org.junit.Test;

public class CalculatorTest {

@Test

public void testAdd() {

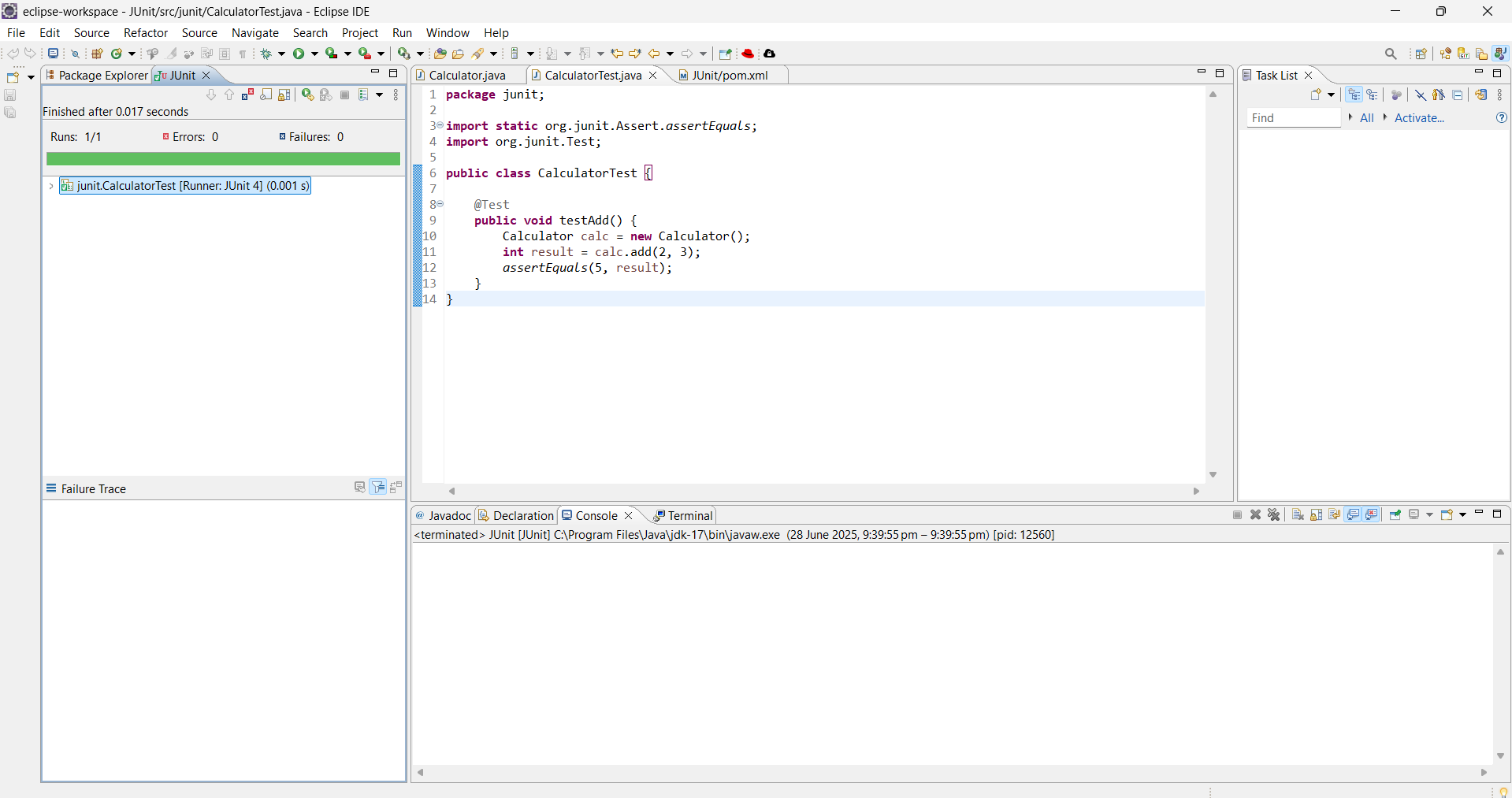
Calculator calc = new Calculator();

int result = calc.add(2, 3);

*assertEquals*(5, result);

}

}

****

**Exercise 3: Assertions in Junit**

package junit;

import static org.junit.Assert.\*;

import org.junit.Test;

public class AssertionsTest {

@Test

public void testAssertions() {

// Assert equals

*assertEquals*(5, 2 + 3);

// Assert true

*assertTrue*(5 > 3);

// Assert false

*assertFalse*(5 < 3);

// Assert null

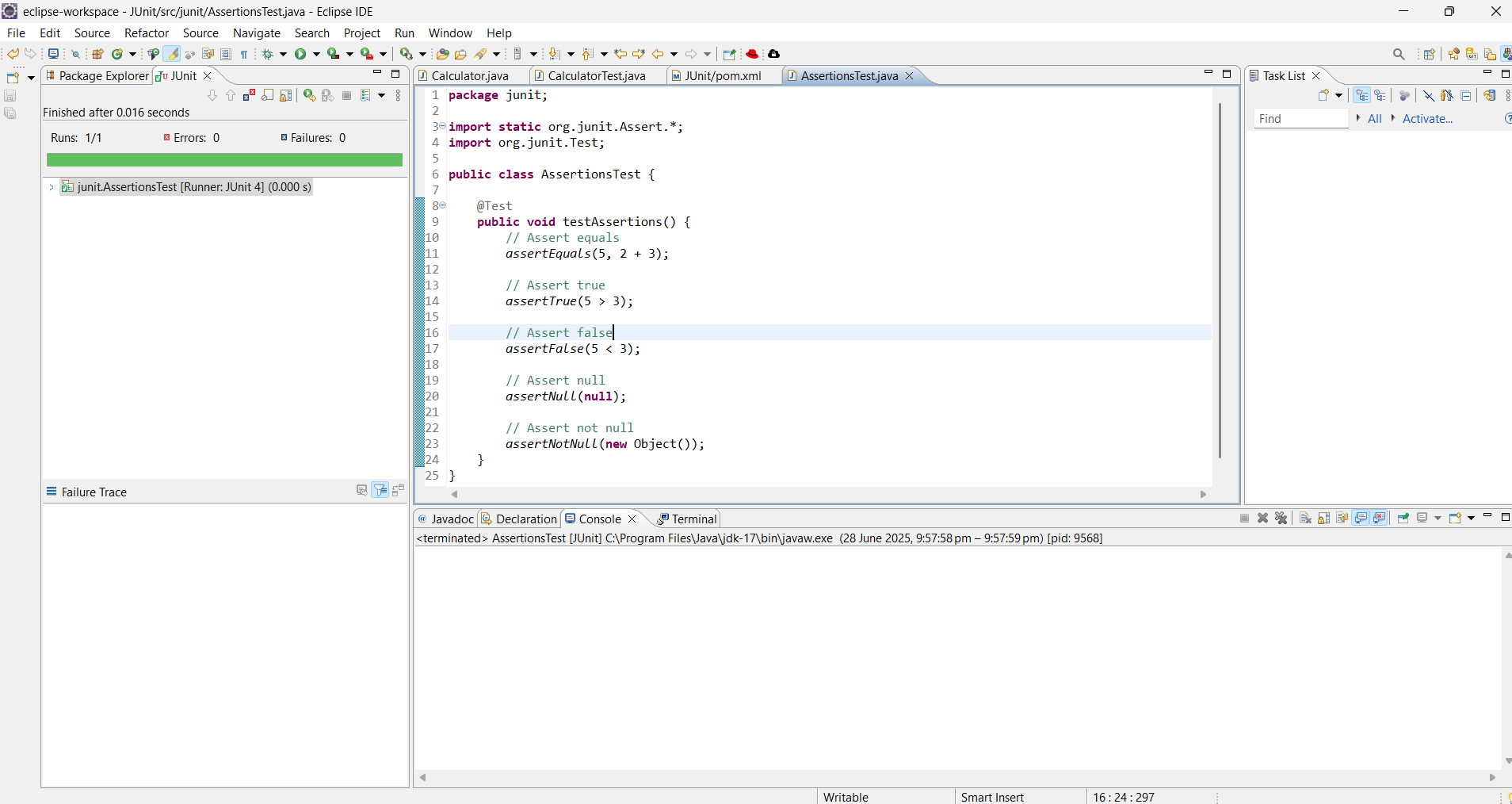
*assertNull*(null);

// Assert not null

*assertNotNull*(new Object());

}

}

****

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

package junit;

import org.junit.After;

import org.junit.Before;

import org.junit.Test;

import static org.junit.Assert.\*;

public class CalculatorTest {

private Calculator calc;

@Before

public void setUp() {

calc = new Calculator();

System.*out*.println("Setup completed");

}

@After

public void tearDown() {

calc = null;

System.*out*.println("Teardown completed");

}

@Test

public void testAddition() {

int result = calc.add(2, 3);

*assertEquals*(5, result);

}

@Test

public void testAdditionWithNegative() {

int result = calc.add(-2, 3);

*assertEquals*(1, result);

}

}

