Week 2: TDD using JUnit5 and Mockito (Implemented in Eclipse)

JUnit_Basic Testing

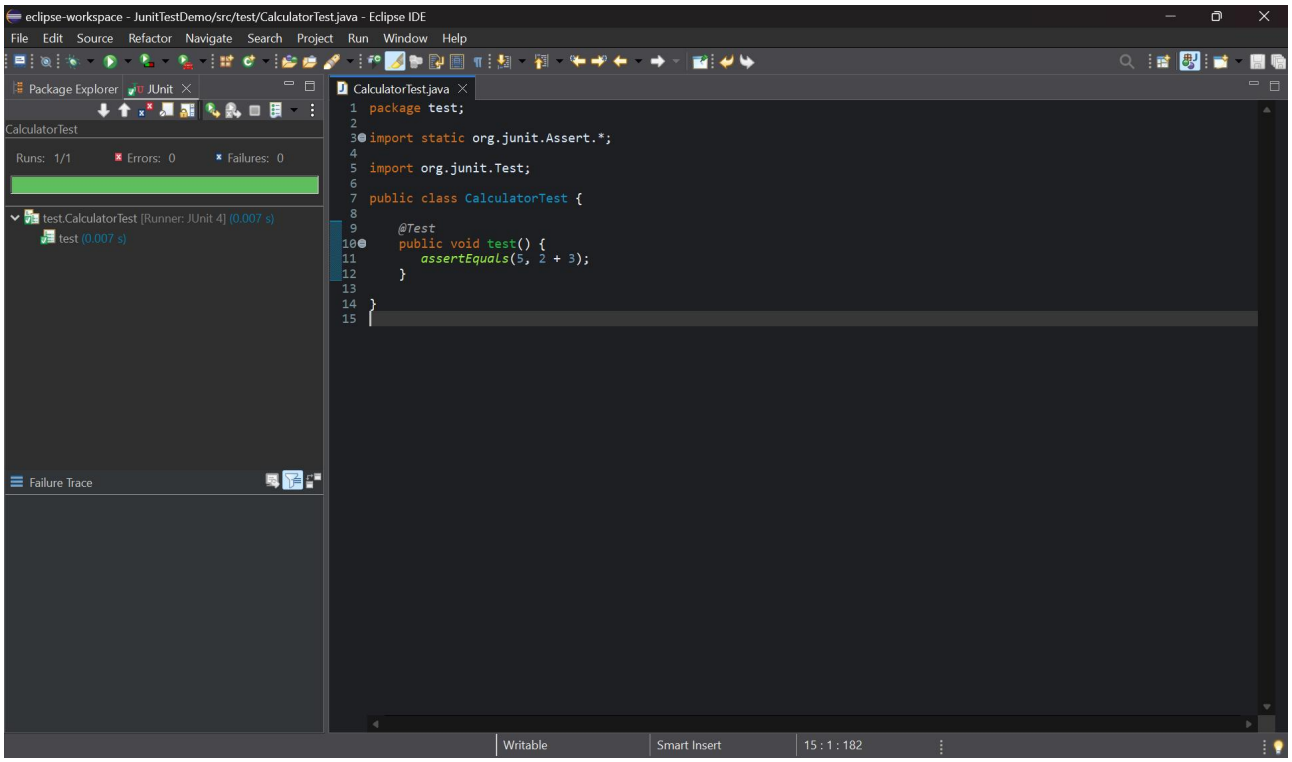
Exercise 1: Setting Up JUnit

IMPLEMENTATION:

CalculatorTest.java

```
package test;
import static org.junit.Assert.assertEquals;
import org.junit.Test;
public class CalculatorTest {
@Test
public void testAddition() {
assertEquals(5, 2 + 3);
}
}
```

Output:

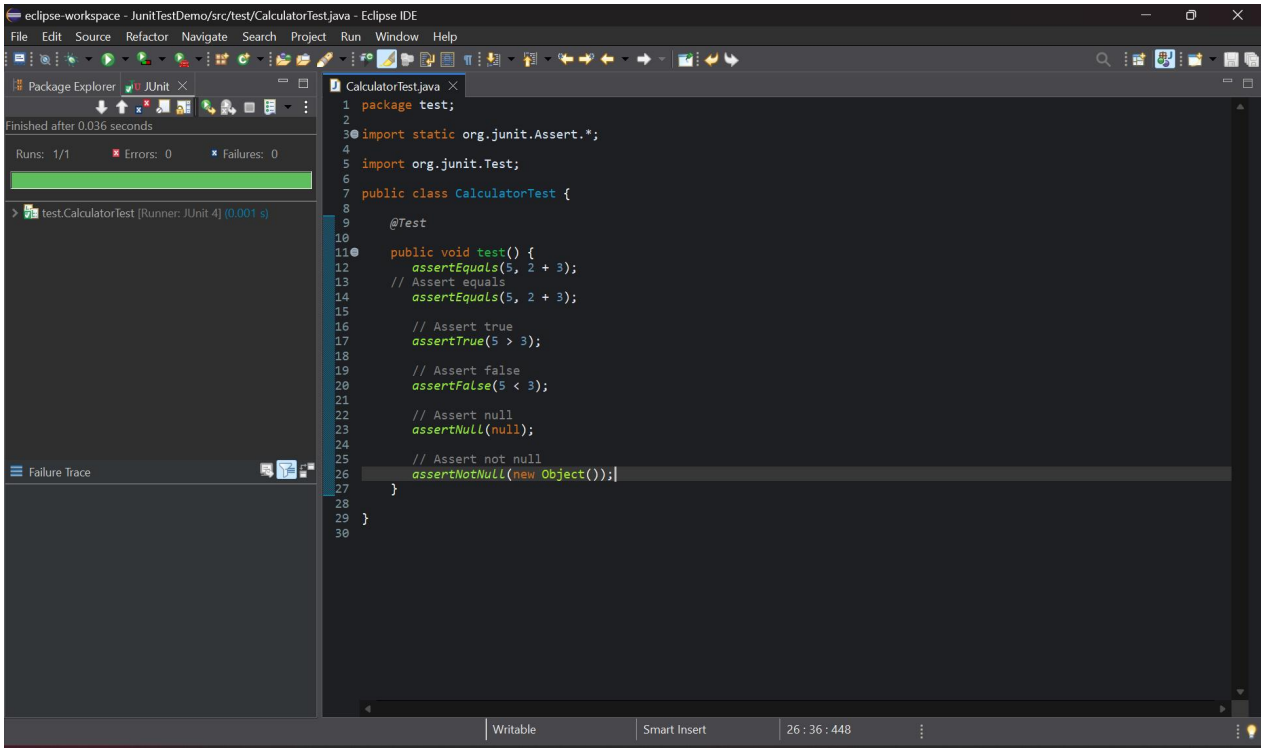


Exercise 3: Assertions in JUnit

AssertionsTest.java:

```
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());
}
}
```

Output:



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

CalculatorTest.java

```
import static org.junit.Assert.*;
import org.junit.Before;
import org.junit.After;
import org.junit.Test;
public class CalculatorTest {
private Calculator calculator;
// Setup - runs before each test
@Before
public void setUp() {
calculator = new Calculator();
System.out.println("Setup called");
}
// Teardown - runs after each test
@After
public void tearDown() {
calculator = null;
System.out.println("Teardown called");
}
@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 = 10;
int b = 4;
// Act
int result = calculator.subtract(a, b);
// Assert
assertEquals(6, result);
```

```
}
}
```

Calculator,java:

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

Output: