

# Sri Lanka Institute of Information Technology



## **Software Engineering Process & Quality Management**

**SE3010**

**Lab 2 – JUnit**

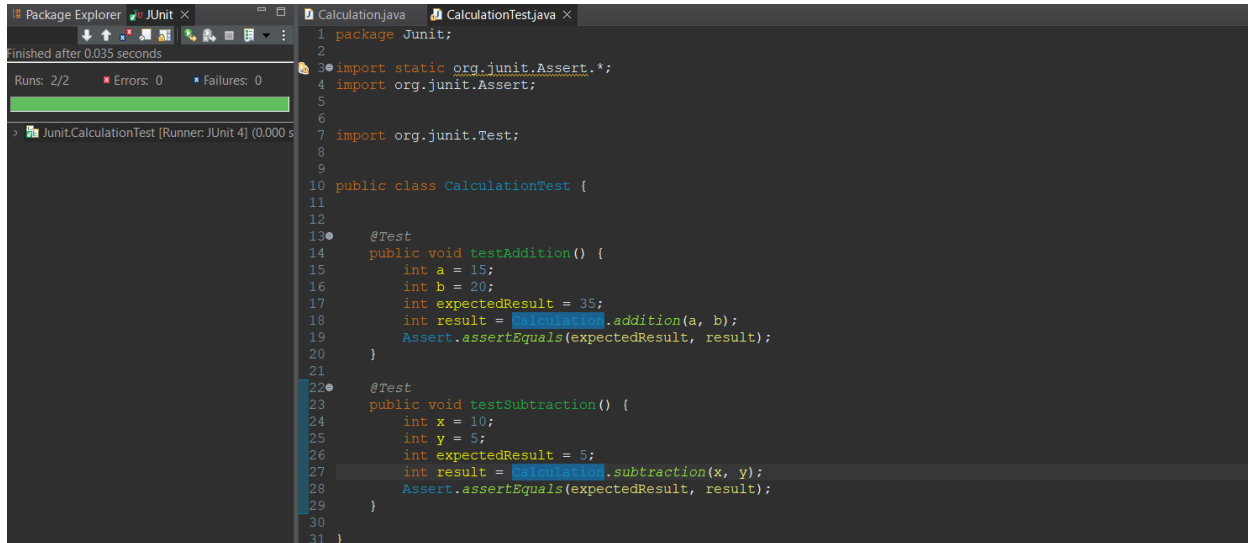
IT22251114 JINAD INDUWITHWA A.G

# Creating a Java Project and Calculation Class

```
Calculation.java ×
1 package Junit;
2
3 public class Calculation {
4
5     public int addition(int a, int b) {
6         int total = a+b;
7         return total;
8     }
9
10    public int subtraction(int x, int y) {
11        int result = x-y;
12        return result;
13    }
14 }
15 }
16 |
```

# Adding a JUnit Test Cases

## Test case 01

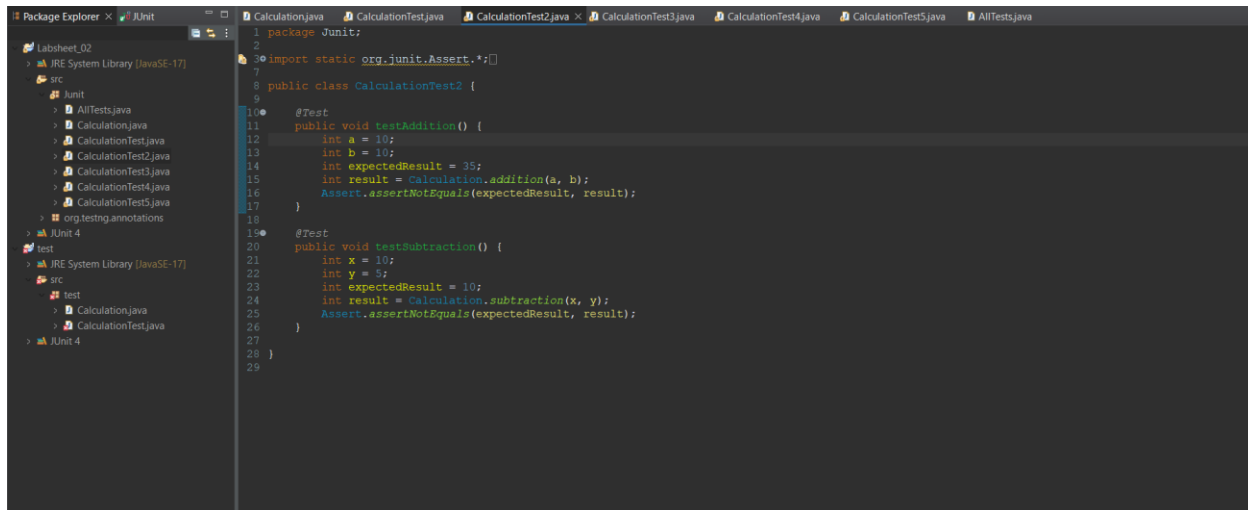


```

1 package JUnit;
2
3 import static org.junit.Assert.*;
4 import org.junit.Assert;
5
6
7 import org.junit.Test;
8
9
10 public class CalculationTest {
11
12
13     @Test
14     public void testAddition() {
15         int a = 15;
16         int b = 20;
17         int expectedResult = 35;
18         int result = Calculation.addition(a, b);
19         Assert.assertEquals(expectedResult, result);
20     }
21
22     @Test
23     public void testSubtraction() {
24         int x = 10;
25         int y = 5;
26         int expectedResult = 5;
27         int result = Calculation.subtraction(x, y);
28         Assert.assertEquals(expectedResult, result);
29     }
30
31 }

```

## Test case 02

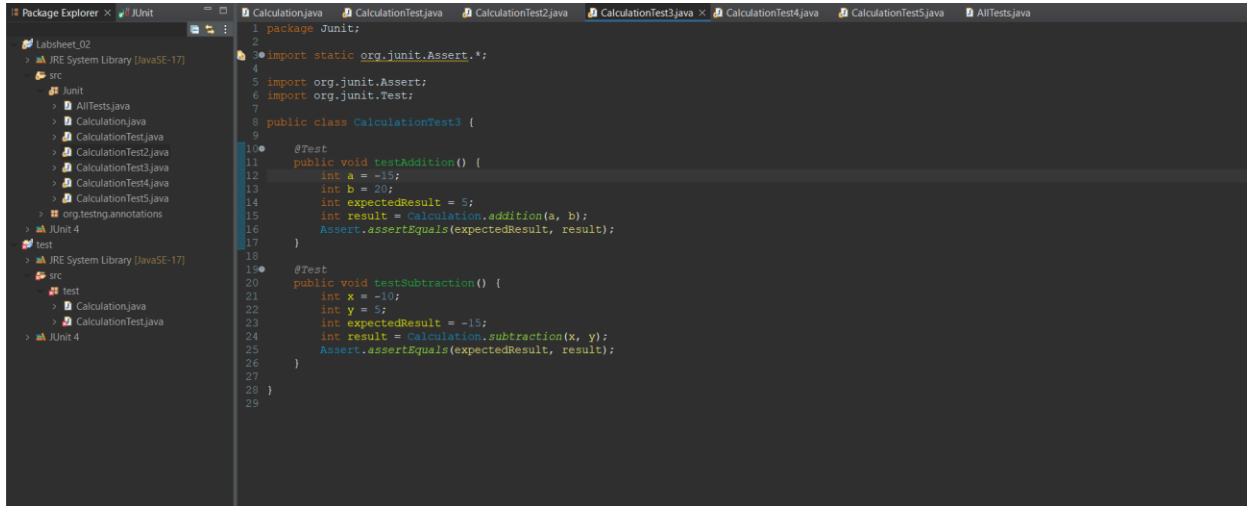


```

1 package JUnit;
2
3 import static org.junit.Assert.*;
4
5 public class CalculationTest2 {
6
7
8     @Test
9     public void testAddition() {
10         int a = 10;
11         int b = 10;
12         int expectedResult = 35;
13         int result = Calculation.addition(a, b);
14         Assert.assertNotEquals(expectedResult, result);
15     }
16
17
18     @Test
19     public void testSubtraction() {
20         int x = 10;
21         int y = 5;
22         int expectedResult = 10;
23         int result = Calculation.subtraction(x, y);
24         Assert.assertNotEquals(expectedResult, result);
25     }
26
27
28 }
29

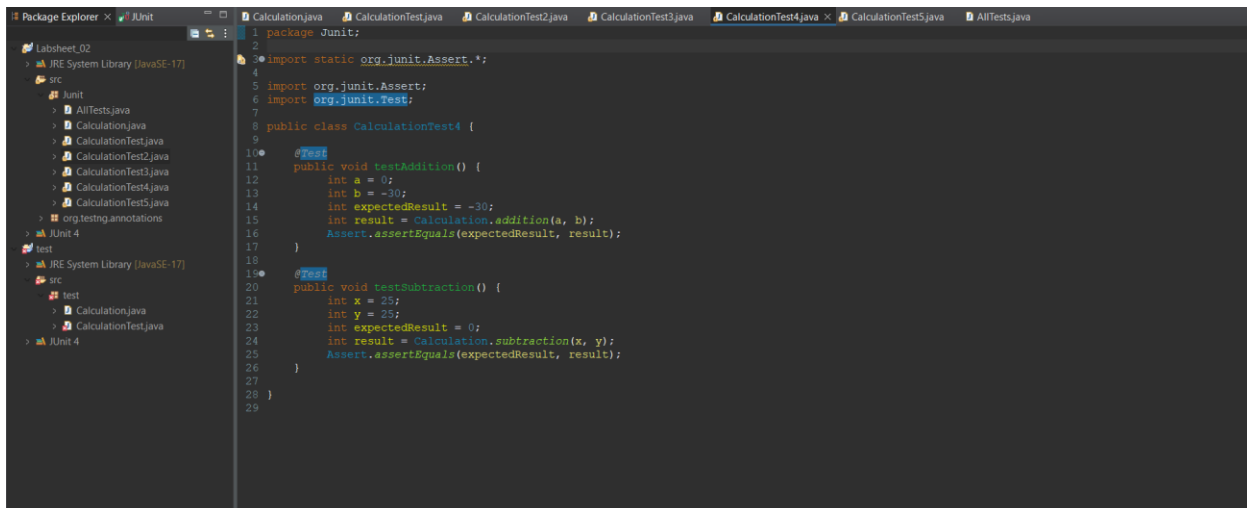
```

## Test case 03



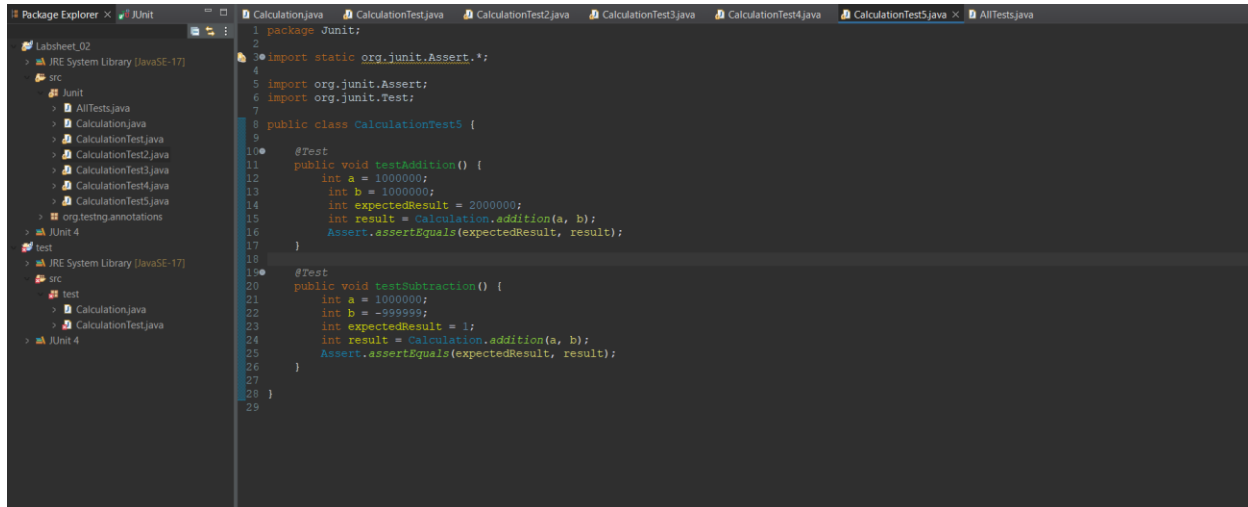
```
1 package JUnit;
2
3 import static org.junit.Assert.*;
4
5 import org.junit.Assert;
6 import org.junit.Test;
7
8 public class CalculationTest3 {
9
10     @Test
11     public void testAddition() {
12         int a = -15;
13         int b = 20;
14         int expectedResult = 5;
15         int result = Calculation.addition(a, b);
16         Assert.assertEquals(expectedResult, result);
17     }
18
19     @Test
20     public void testSubtraction() {
21         int x = -10;
22         int y = 5;
23         int expectedResult = -15;
24         int result = Calculation.subtraction(x, y);
25         Assert.assertEquals(expectedResult, result);
26     }
27
28 }
29
```

## Test case 04



```
1 package JUnit;
2
3 import static org.junit.Assert.*;
4
5 import org.junit.Assert;
6 import org.junit.Test;
7
8 public class CalculationTest4 {
9
10     @Test
11     public void testAddition() {
12         int a = 0;
13         int b = -30;
14         int expectedResult = -30;
15         int result = Calculation.addition(a, b);
16         Assert.assertEquals(expectedResult, result);
17     }
18
19     @Test
20     public void testSubtraction() {
21         int x = 25;
22         int y = 25;
23         int expectedResult = 0;
24         int result = Calculation.subtraction(x, y);
25         Assert.assertEquals(expectedResult, result);
26     }
27
28 }
29
```

## Test case 05



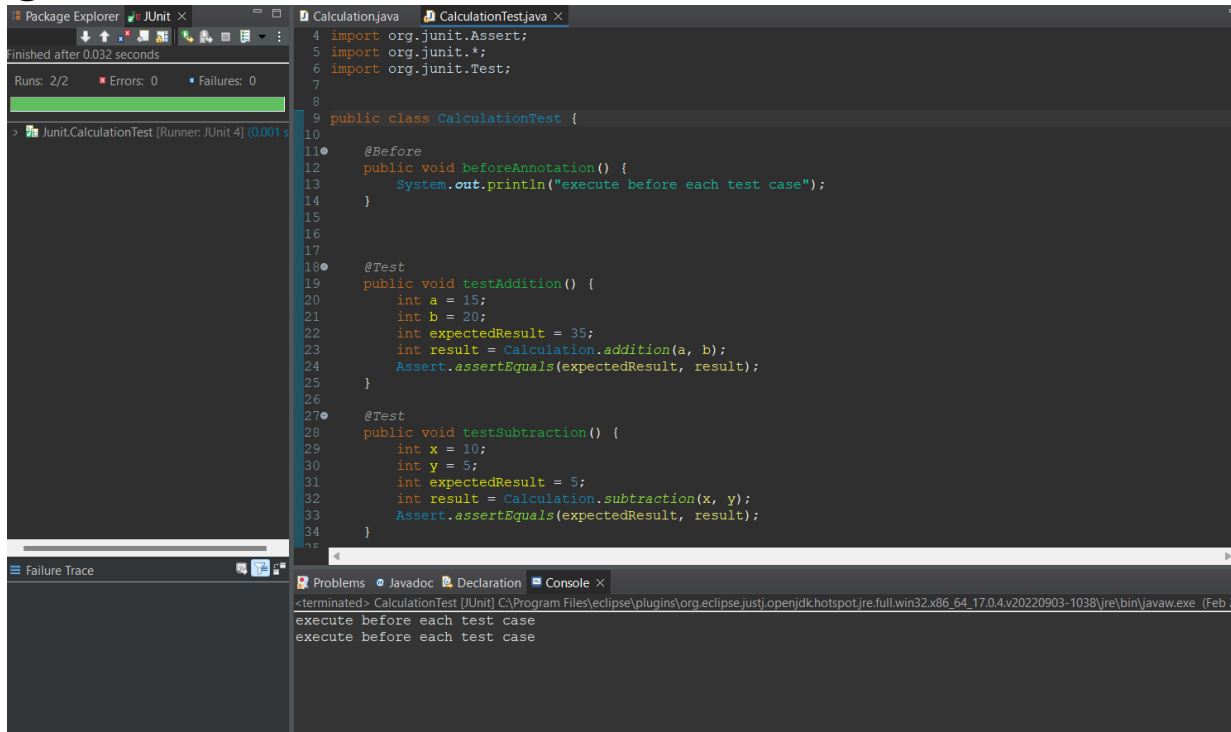
The screenshot shows an IDE with the following components:

- Package Explorer (Left):** Displays a project structure for 'Labsheet\_02'. It includes a 'src' folder with 'JUnit' and 'AllTests.java'. There are also several 'CalculationTest' files (CalculationTest.java, CalculationTest2.java, CalculationTest3.java, CalculationTest4.java, CalculationTest5.java) and a 'test' folder containing 'org.testing.annotations', 'JUnit 4', and 'test'.
- Code Editor (Right):** Shows the code for 'CalculationTest5.java'. The code is as follows:

```
1 package JUnit;
2
3 import static org.junit.Assert.*;
4
5 import org.junit.Assert;
6 import org.junit.Test;
7
8 public class CalculationTest5 {
9
10     @Test
11     public void testAddition() {
12         int a = 1000000;
13         int b = 1000000;
14         int expectedResult = 2000000;
15         int result = Calculation.addition(a, b);
16         Assert.assertEquals(expectedResult, result);
17     }
18
19     @Test
20     public void testSubtraction() {
21         int a = 1000000;
22         int b = -999999;
23         int expectedResult = 1;
24         int result = Calculation.addition(a, b);
25         Assert.assertEquals(expectedResult, result);
26     }
27 }
28
29
```

# JUnit Annotations

## @Before annotation



The screenshot shows the Eclipse IDE with a JUnit test runner. The Package Explorer on the left shows the project structure. The main editor displays the `CalculationTest.java` file. The console at the bottom shows the output of the test run.

```

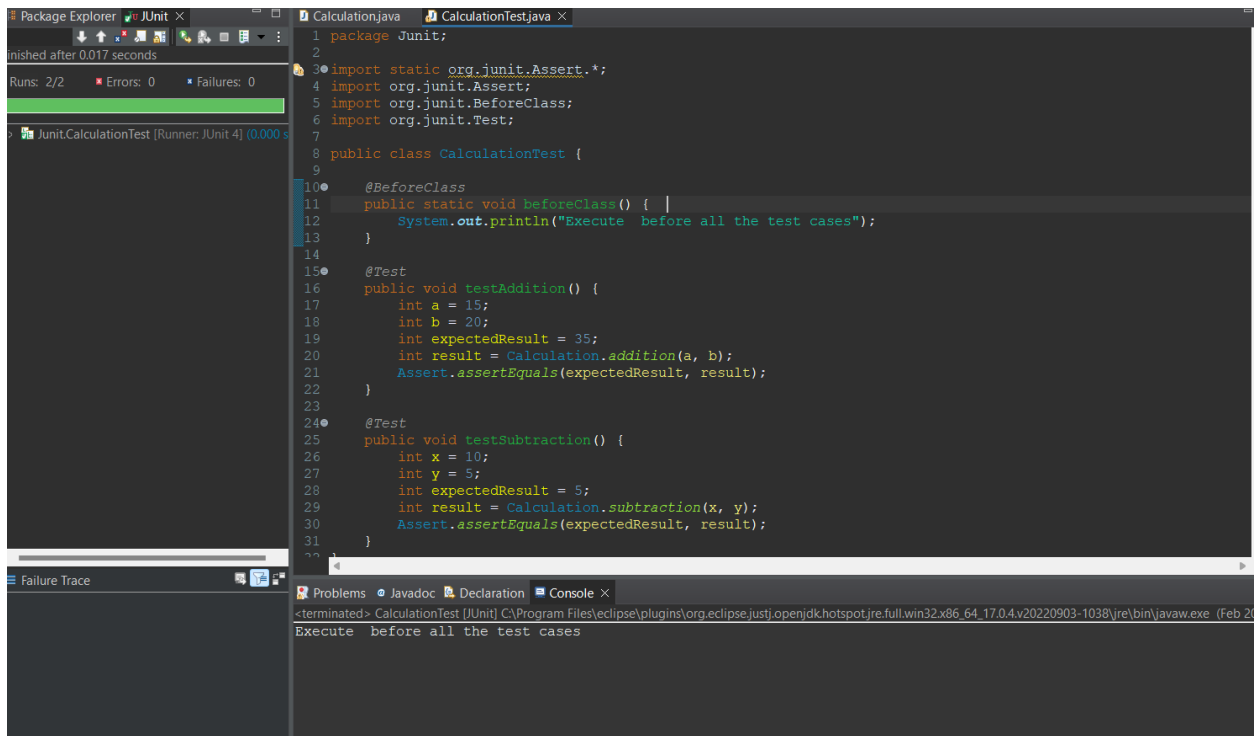
4 import org.junit.Assert;
5 import org.junit.*;
6 import org.junit.Test;
7
8
9 public class CalculationTest {
10
11     @Before
12     public void beforeAnnotation() {
13         System.out.println("execute before each test case");
14     }
15
16
17
18     @Test
19     public void testAddition() {
20         int a = 15;
21         int b = 20;
22         int expectedResult = 35;
23         int result = Calculation.addition(a, b);
24         Assert.assertEquals(expectedResult, result);
25     }
26
27     @Test
28     public void testSubtraction() {
29         int x = 10;
30         int y = 5;
31         int expectedResult = 5;
32         int result = Calculation.subtraction(x, y);
33         Assert.assertEquals(expectedResult, result);
34     }
35 }
  
```

Console Output:

```

<terminated> CalculationTest [JUnit] C:\Program Files\eclipse\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64.17.0.4.v20220903-1038\jre\bin\javaw.exe (Feb 2022)
execute before each test case
execute before each test case
  
```

## @BeforeClass annotation



The screenshot shows the Eclipse IDE with a JUnit test runner. The Package Explorer on the left shows the project structure. The main editor displays the `CalculationTest.java` file. The console at the bottom shows the output of the test run.

```

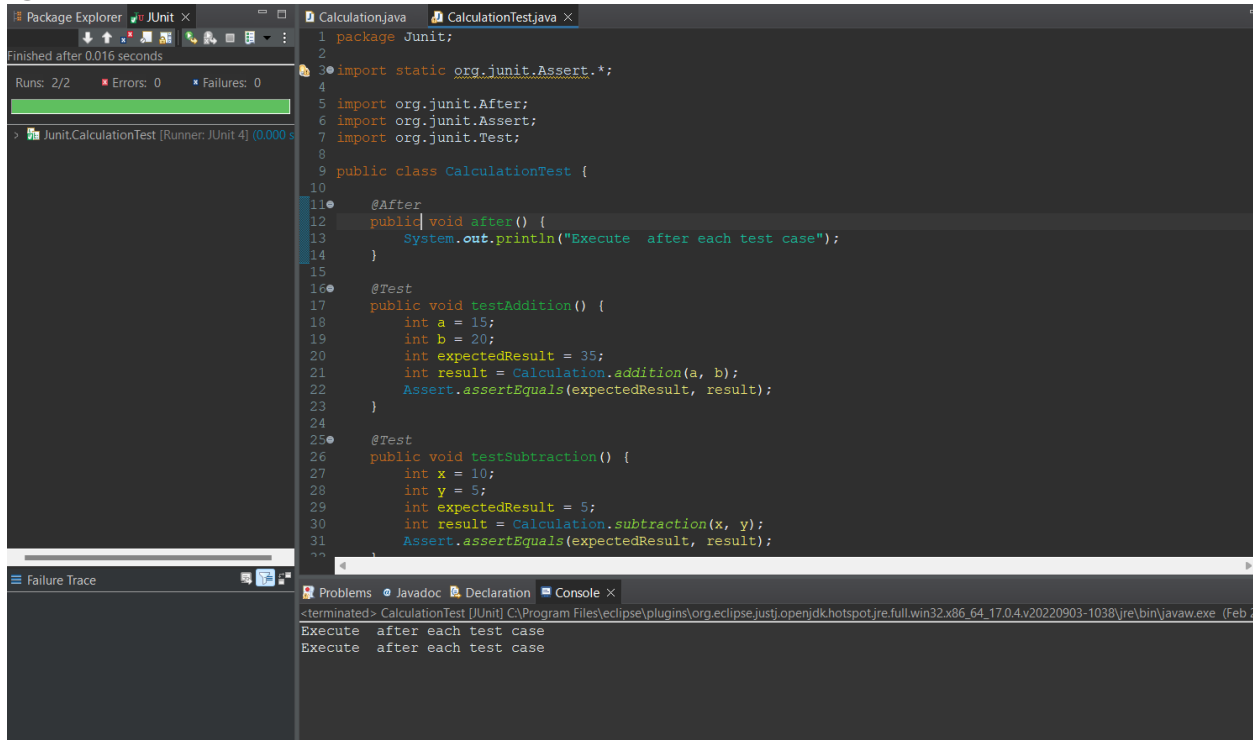
1 package junit;
2
3 import static org.junit.Assert.*;
4 import org.junit.Assert;
5 import org.junit.BeforeClass;
6 import org.junit.Test;
7
8 public class CalculationTest {
9
10     @BeforeClass
11     public static void beforeClass() {
12         System.out.println("Execute before all the test cases");
13     }
14
15     @Test
16     public void testAddition() {
17         int a = 15;
18         int b = 20;
19         int expectedResult = 35;
20         int result = Calculation.addition(a, b);
21         Assert.assertEquals(expectedResult, result);
22     }
23
24     @Test
25     public void testSubtraction() {
26         int x = 10;
27         int y = 5;
28         int expectedResult = 5;
29         int result = Calculation.subtraction(x, y);
30         Assert.assertEquals(expectedResult, result);
31     }
32 }
  
```

Console Output:

```

<terminated> CalculationTest [JUnit] C:\Program Files\eclipse\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64.17.0.4.v20220903-1038\jre\bin\javaw.exe (Feb 2022)
Execute before all the test cases
  
```

## @After annotation



The screenshot shows the Eclipse IDE with a JUnit test project. The Package Explorer on the left shows the project structure. The main editor displays the `CalculationTest.java` file. The code uses the `@After` annotation to execute a method after each test case. The console output shows the execution of the test cases and the message printed by the `after` method.

```

1 package junit;
2
3 import static org.junit.Assert.*;
4
5 import org.junit.After;
6 import org.junit.Assert;
7 import org.junit.Test;
8
9 public class CalculationTest {
10
11     @After
12     public void after() {
13         System.out.println("Execute after each test case");
14     }
15
16     @Test
17     public void testAddition() {
18         int a = 15;
19         int b = 20;
20         int expectedResult = 35;
21         int result = Calculation.addition(a, b);
22         Assert.assertEquals(expectedResult, result);
23     }
24
25     @Test
26     public void testSubtraction() {
27         int x = 10;
28         int y = 5;
29         int expectedResult = 5;
30         int result = Calculation.subtraction(x, y);
31         Assert.assertEquals(expectedResult, result);
32     }
33 }

```

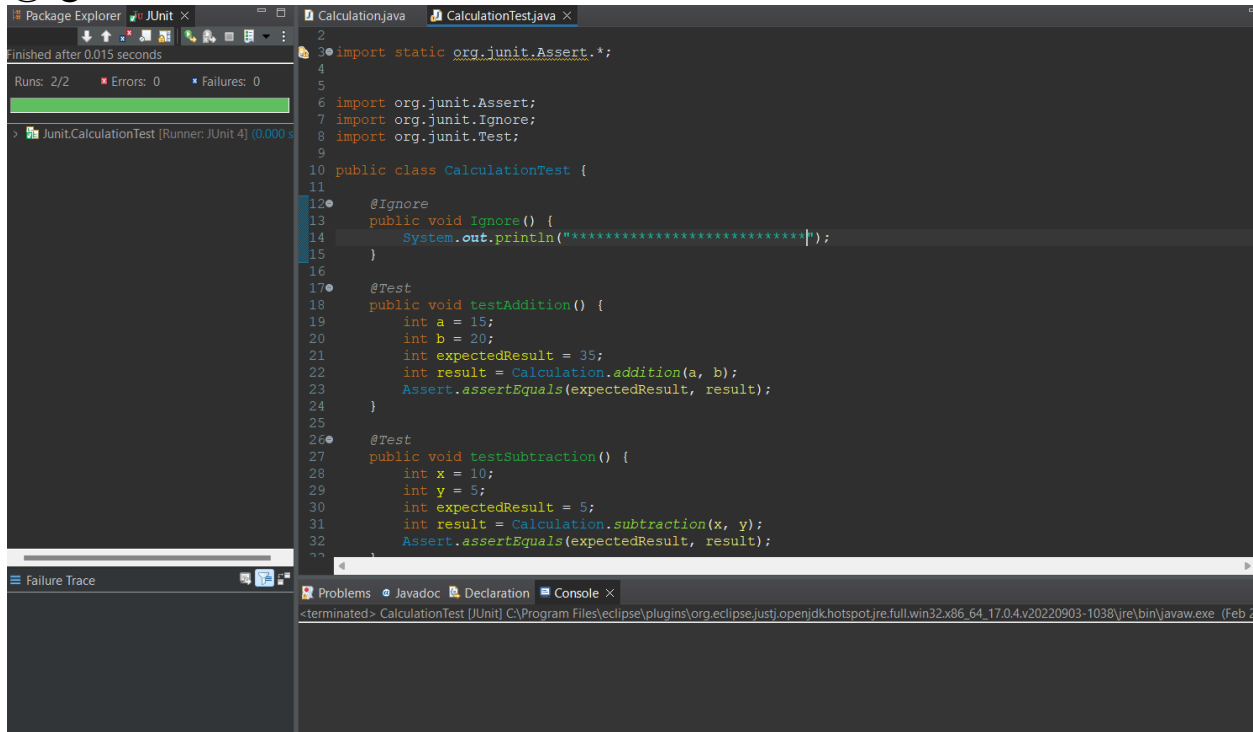
Console Output:

```

<terminated> CalculationTest [JUnit] C:\Program Files\eclipse\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64_17.0.4.v20220903-1038\jre\bin\javaw.exe (Feb 2
Execute after each test case
Execute after each test case

```

## @Ignore annotation



The screenshot shows the Eclipse IDE with a JUnit test project. The Package Explorer on the left shows the project structure. The main editor displays the `CalculationTest.java` file. The code uses the `@Ignore` annotation to skip a test method. The console output shows the execution of the test cases and the message printed by the `Ignore` method.

```

1 package junit;
2
3 import static org.junit.Assert.*;
4
5 import org.junit.Assert;
6 import org.junit.Ignore;
7 import org.junit.Test;
8
9 public class CalculationTest {
10
11     @Ignore
12     public void Ignore() {
13         System.out.println("*****");
14     }
15
16     @Test
17     public void testAddition() {
18         int a = 15;
19         int b = 20;
20         int expectedResult = 35;
21         int result = Calculation.addition(a, b);
22         Assert.assertEquals(expectedResult, result);
23     }
24
25     @Test
26     public void testSubtraction() {
27         int x = 10;
28         int y = 5;
29         int expectedResult = 5;
30         int result = Calculation.subtraction(x, y);
31         Assert.assertEquals(expectedResult, result);
32     }
33 }

```

Console Output:

```

<terminated> CalculationTest [JUnit] C:\Program Files\eclipse\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64_17.0.4.v20220903-1038\jre\bin\javaw.exe (Feb 2
*****

```

# Creating a Test Suite

