

JUnit Testing Exercises

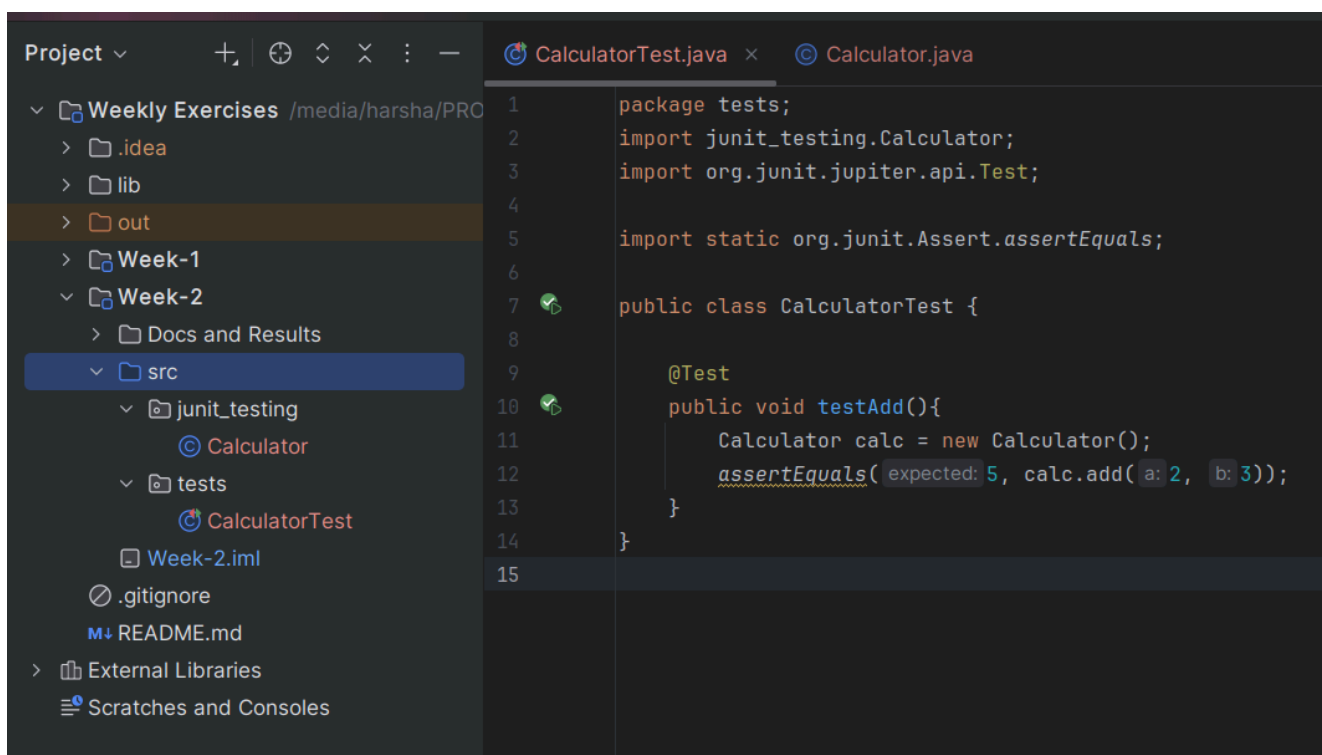
Source Code: [Here](#)

Mandatory Exercises

Exercise-1: Setting Up JUnit

Setting up JUnit in your Java project and create a Test class.

Results:



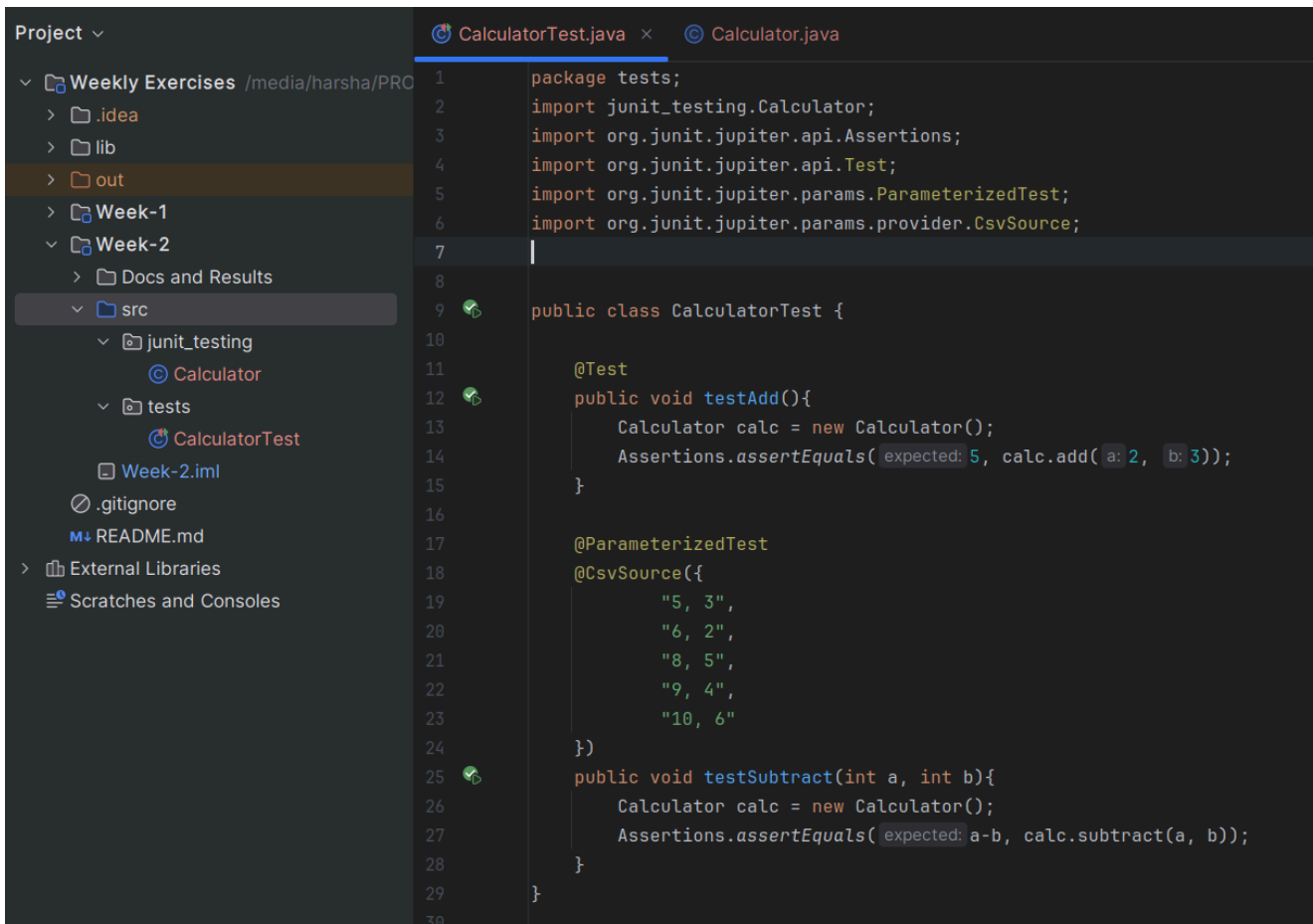
The screenshot shows an IDE with a project named 'Weekly Exercises' located at '/media/harsha/PRO'. The project structure on the left includes folders for '.idea', 'lib', 'out', 'Week-1', 'Week-2', 'src', 'junit_testing', 'tests', and files like 'Calculator', 'CalculatorTest', 'Week-2.iml', '.gitignore', and 'README.md'. The 'src' folder is expanded, showing 'junit_testing' and 'tests'. The 'tests' folder contains 'CalculatorTest'. The 'CalculatorTest.java' file is open in the editor, showing the following code:

```
1 package tests;
2 import junit_testing.Calculator;
3 import org.junit.jupiter.api.Test;
4
5 import static org.junit.Assert.assertEquals;
6
7 public class CalculatorTest {
8
9     @Test
10     public void testAdd(){
11         Calculator calc = new Calculator();
12         assertEquals( expected: 5, calc.add( a: 2, b: 3));
13     }
14 }
15
```

Exercise-3: Assertions in JUnit

Tested basic calculator operations like Add and Subtract

Source Code:



```
package junit_testing;

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

    public int subtract(int a, int b){
        return a-b;
    }
}
```

```
package tests;
import junit_testing.Calculator;
import org.junit.jupiter.api.Assertions;
import org.junit.jupiter.api.Test;
import org.junit.jupiter.params.ParameterizedTest;
import org.junit.jupiter.params.provider.CsvSource;

public class CalculatorTest {

    @Test
    public void testAdd(){
        Calculator calc = new Calculator();
```

```

        Assertions.assertEquals(5, calc.add(2, 3));
    }

    @ParameterizedTest
    @CsvSource({
        "5, 3",
        "6, 2",
        "8, 5",
        "9, 4",
        "10, 6"
    })
    public void testSubtract(int a, int b){
        Calculator calc = new Calculator();
        Assertions.assertEquals(a-b, calc.subtract(a, b));
    }

    @Test
    public void givenTestSourceCode(){
        // Assert equals
        Assertions.assertEquals(5, 2 + 3);
        // Assert true
        Assertions.assertTrue(5 > 3);
        // Assert false
        Assertions.assertTrue(5 < 3);
        // Assert null
        Assertions.assertNull(null);
        // Assert not null
        Assertions.assertNotNull(new Object());
    }
}

```

Results:

Run CalculatorTest x

1 test failed, 6 passed 7 tests total, 89 ms

Test Name	Duration	Status
CalculatorTest (tests)	89 ms	Failed
testAdd()	38 ms	Passed
testSubtract(int, int)	45 ms	Passed
[1] 5, 3	40 ms	Passed
[2] 6, 2	2 ms	Passed
[3] 8, 5	1 ms	Passed
[4] 9, 4	1 ms	Passed
[5] 10, 6	1 ms	Passed
givenTestSourceCode()	6 ms	Failed

```

org.opentest4j.AssertionFailedError:
Expected :true
Actual   :false
<Click to see difference>

> <6 internal lines>
>   at tests.CalculatorTest.givenTestSourceCode(CalculatorTest.java:37) <1 internal line>
> <67 folded frames>

Process finished with exit code 255

```

Other Exercises

Exercise-2: Writing Basic JUnit Tests

Created `Calclater` class with methods like `add` and `subtract` and these are tested with JUnit test cases.

Source code:

```
package junit_testing;

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

    public int subtract(int a, int b){
        return a-b;
    }
}
```

```
package tests;
import junit_testing.Calculator;
import org.junit.jupiter.api.Assertions;
import org.junit.jupiter.api.Test;
import org.junit.jupiter.params.ParameterizedTest;
import org.junit.jupiter.params.provider.CsvSource;

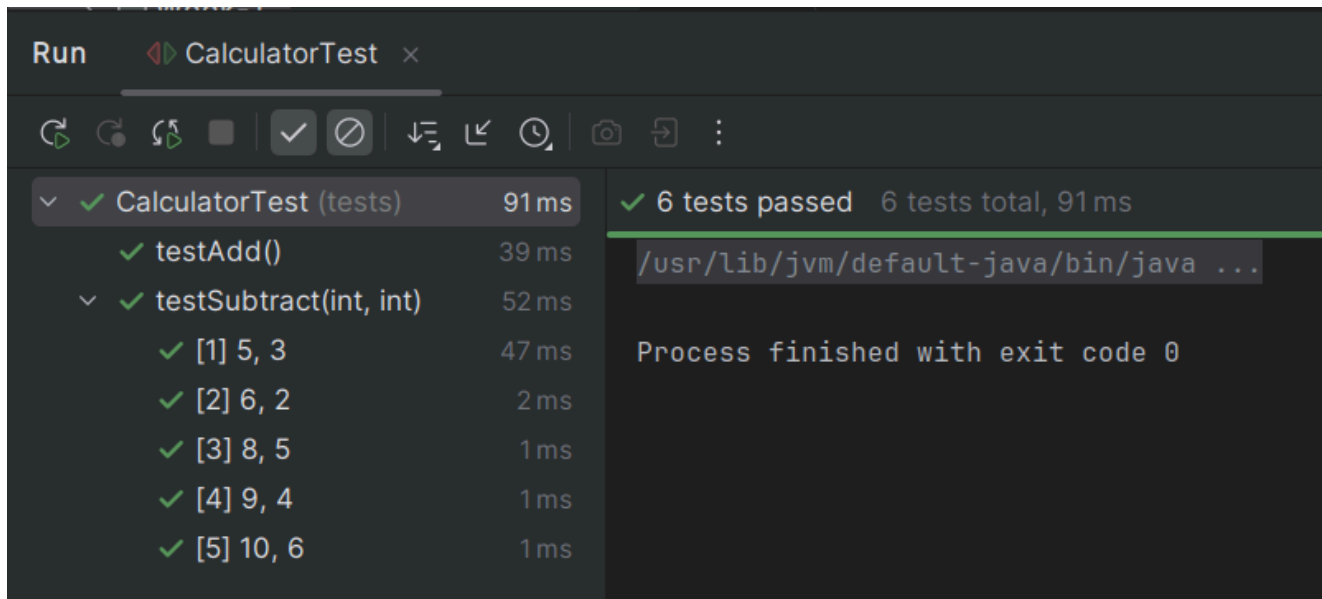
public class CalculatorTest {

    @Test
    public void testAdd(){
        Calculator calc = new Calculator();
        Assertions.assertEquals(5, calc.add(2, 3));
    }

    @ParameterizedTest
    @CsvSource({
        "5, 3",
        "6, 2",
        "8, 5",
        "9, 4",
        "10, 6"
    })
    public void testSubtract(int a, int b){
        Calculator calc = new Calculator();
    }
```

```
        Assertions.assertEquals(a-b, calc.subtract(a, b));  
    }  
}
```

Results:



The screenshot shows the 'Run' window of an IDE, specifically for a test class named 'CalculatorTest'. The window is divided into two main sections. The left section displays a tree view of the test results, and the right section shows the output of the test execution.

Run Window: CalculatorTest

Test Results (Left Panel):

Test Name	Duration
✓ CalculatorTest (tests)	91 ms
✓ testAdd()	39 ms
✓ testSubtract(int, int)	52 ms
✓ [1] 5, 3	47 ms
✓ [2] 6, 2	2 ms
✓ [3] 8, 5	1 ms
✓ [4] 9, 4	1 ms
✓ [5] 10, 6	1 ms

Output (Right Panel):

```
✓ 6 tests passed 6 tests total, 91 ms  
/usr/lib/jvm/default-java/bin/java ...  
  
Process finished with exit code 0
```