JUnit Testing Exercises

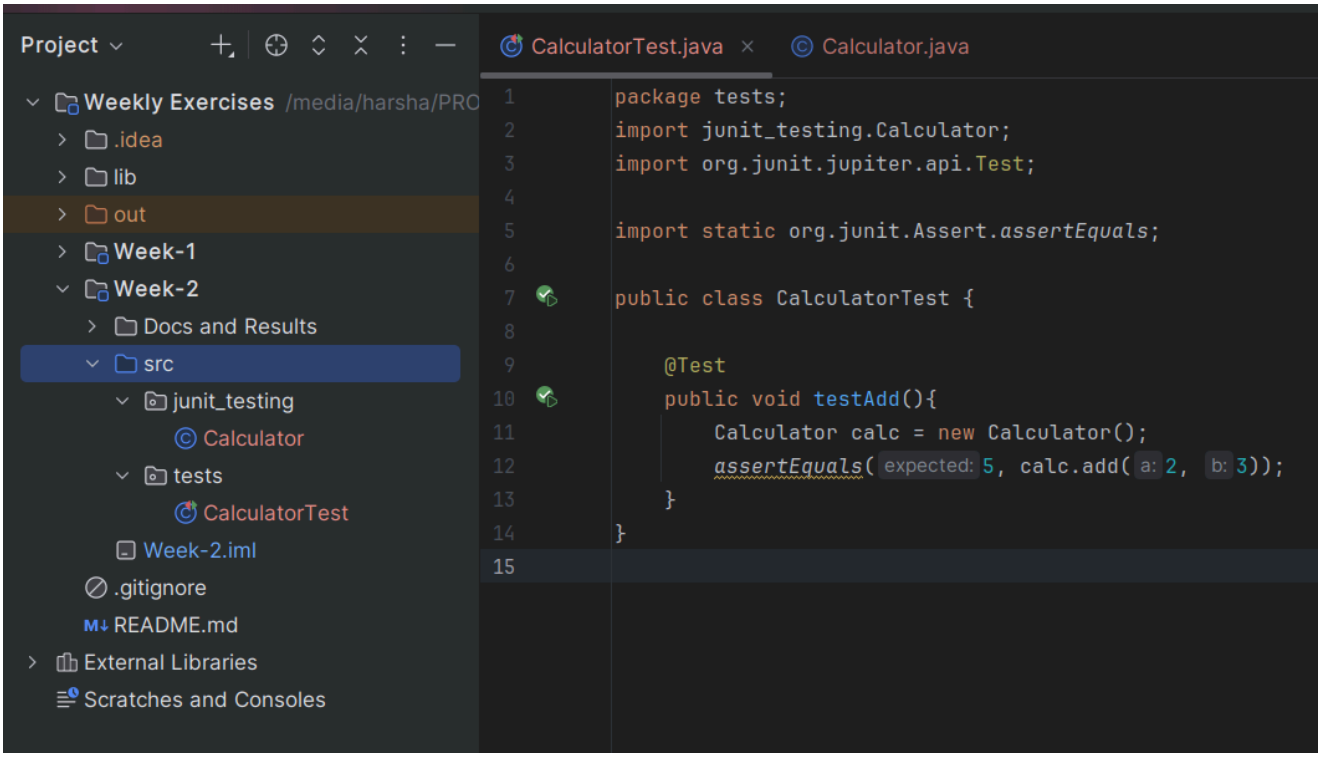
Source Code: Here

Mandatory Exercises

Exercise-1: Setting Up JUnit

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

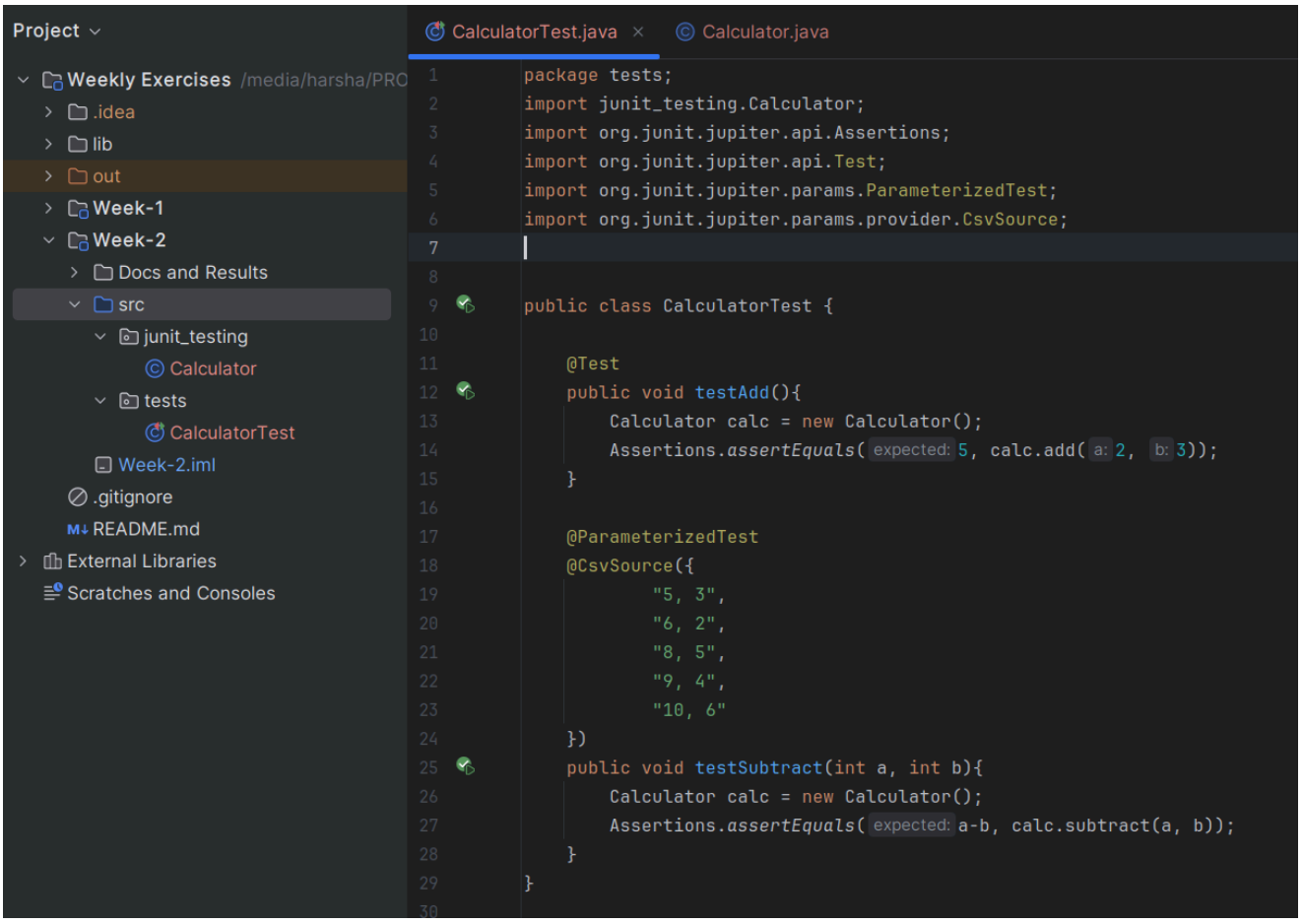
Results:



Exercise-3: Assertions in JUnit

Tested basic calculator operations like Add and Subtract

Source Code:

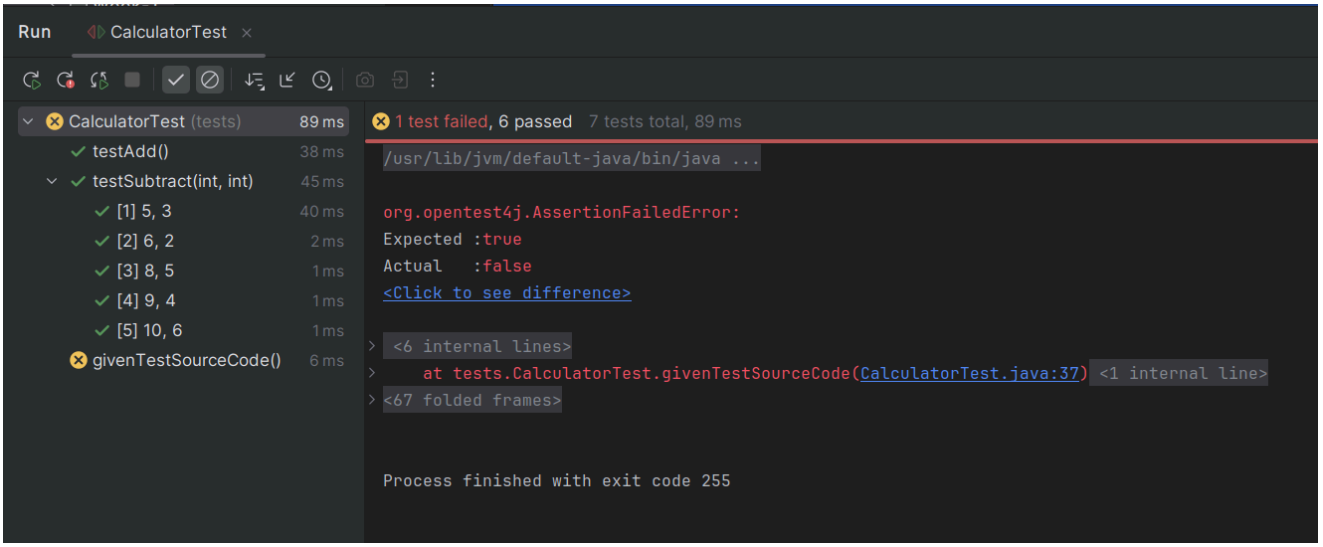



```java
package junit_testing;

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

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

```java
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:



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

Here I have created two `static` methods to initiate and destroy the object of `Calculator` class.

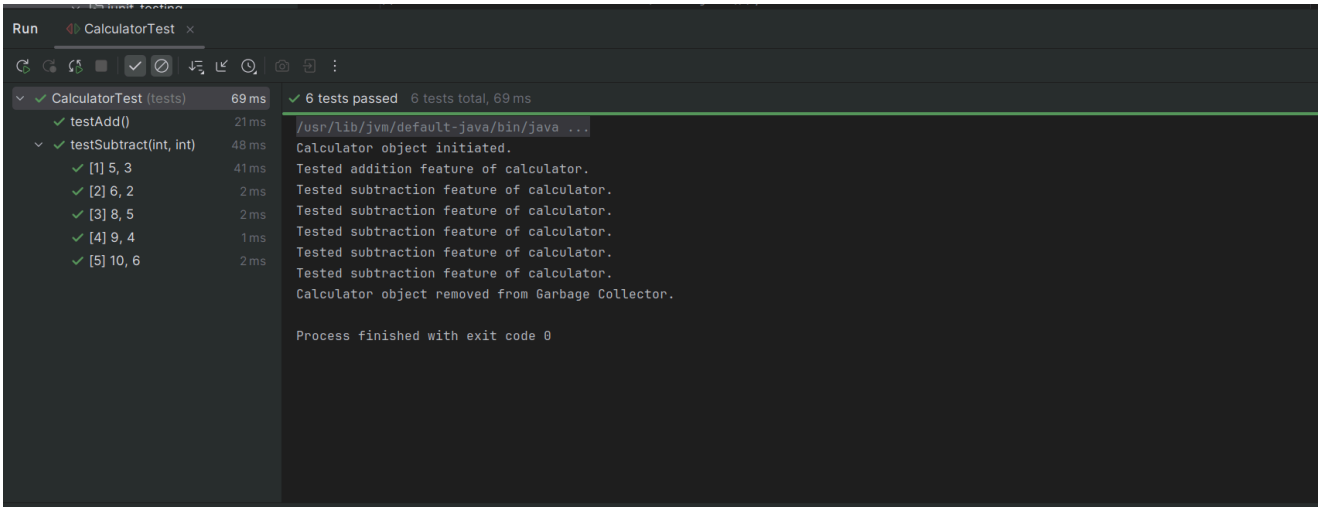
Source Code:

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


public class CalculatorTest {
    private static Calculator calc;

    @BeforeAll
    public static void initiateCalcObject(){
        if(calc == null){
            System.out.println("Calculator object initiated.");
            calc = new Calculator();
        }
    }

    @AfterAll
    public static void destroyCalculatorObject(){
        if(calc!=null){
            calc = null;
            System.gc();
            System.out.println("Calculator object removed from Garbage
Collector.");
        }
    }

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

        System.out.println("Tested addition feature of calculator.");
    }

    @ParameterizedTest
    @CsvSource({
            "5, 3",
            "6, 2",
            "8, 5",
```

```
            "9, 4",
            "10, 6"
    })
    public void testSubtract(int a, int b){
        Assertions.assertEquals(a-b, calc.subtract(a, b));
        System.out.println("Tested subtraction feature of calculator.");
    }
}
```

Here `initiateCalcObject` and `destroyCalculatorObject` methods will be invoked before and after executing all the test cases.

Results:



Other Exercises

Exercise-2: Writing Basic JUnit Tests

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

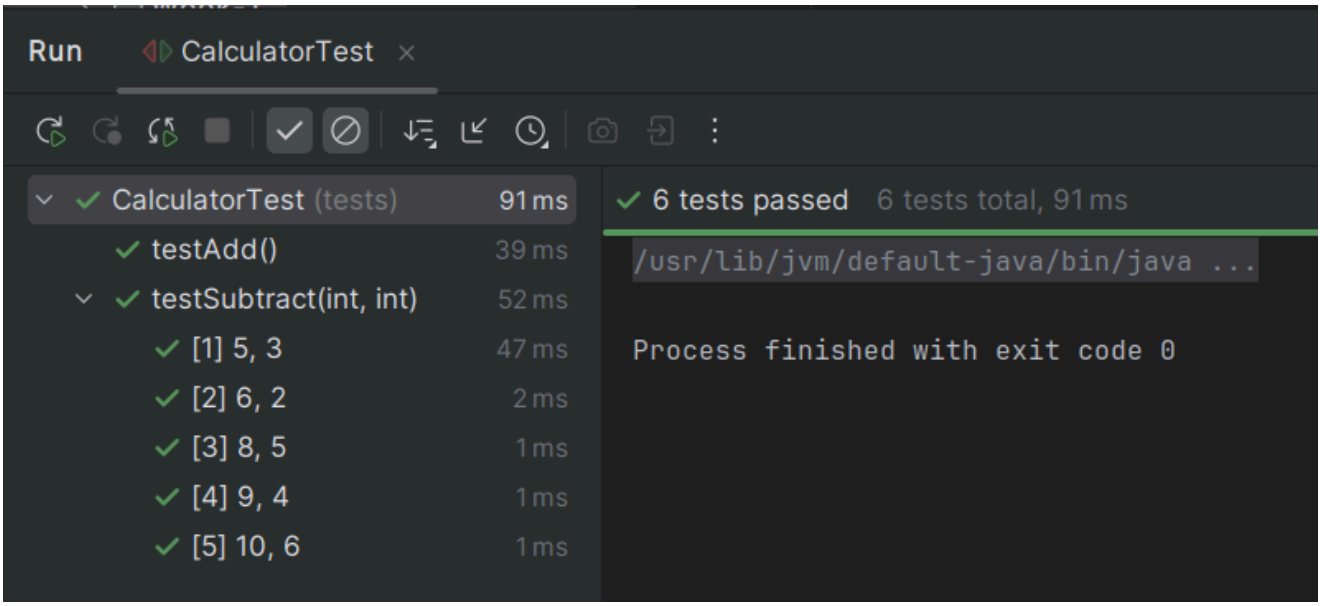
Source code:

```
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:


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