Junit Testing , Mockito and Sl4j Logging

// File: 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;

    }

    public int multiply(int a, int b) {

        return a \* b;

    }

    public int divide(int a, int b) {

        if (b == 0) throw new IllegalArgumentException("Cannot divide by zero");

        return a / b;

    }

}

// File: CalculatorTest.java

import org.junit.jupiter.api.Test;

import static org.testng.Assert.assertThrows;

import static org.testng.AssertJUnit.assertEquals;

public class CalculatorTest {

    Calculator calculator = new Calculator();

    @Test

    void testAddition() {

        assertEquals(10, calculator.add(7, 3));

    }

    @Test

    void testSubtraction() {

        assertEquals(4, calculator.subtract(10, 6));

    }

    @Test

    void testMultiplication() {

        assertEquals(20, calculator.multiply(4, 5));

    }

    @Test

    void testDivision() {

        assertEquals(2, calculator.divide(10, 5));

    }

    @Test

    void testDivisionByZero() {

        Exception exception = assertThrows(IllegalArgumentException.class, () -> {

            calculator.divide(5, 0);

        });

        assertEquals("Cannot divide by zero", exception.getMessage());

    }

// File: Main.java

    public static void main(String[] args) {

        Calculator calc = new Calculator();

        System.out.println("Add: " + calc.add(10, 5));

        System.out.println("Subtract: " + calc.subtract(10, 5));

    }

}

//File: CalcTest.java

import org.junit.jupiter.api.AfterEach;

import org.junit.jupiter.api.BeforeEach;

import org.junit.jupiter.api.Test;

import static org.junit.jupiter.api.Assertions.\*;

public class CalcTest {

    private Calculator calculator;

    @BeforeEach

    void setUp() {

        calculator = new Calculator();

        System.out.println("Setting up test...");

    }

    @AfterEach

    void tearDown() {

        System.out.println("Tearing down...");

    }

    @Test

    void testAdd() {

        assertEquals(5, calculator.add(2, 3));

    }

}

//File: AssertionTest.java

import static org.junit.jupiter.api.Assertions.\*;

import org.junit.jupiter.api.Test;

public class AssertionTest {

    @Test

    public void testAssertions() {

        // Assert equals

        assertEquals(5, 2 + 3);

        // Assert true

        assertTrue(5 > 3);

        // Assert false

        assertFalse(5 < 3);

        // Assert null

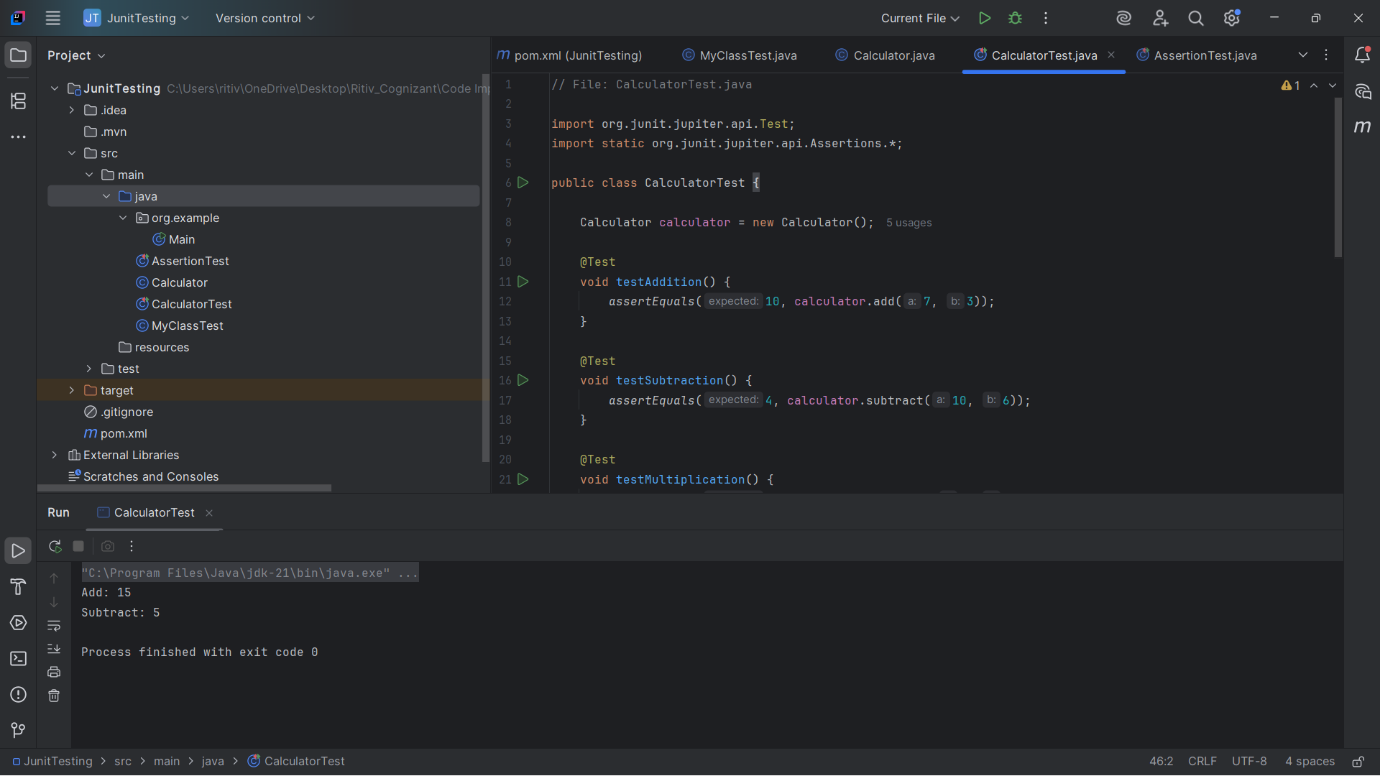
        assertNull(null);

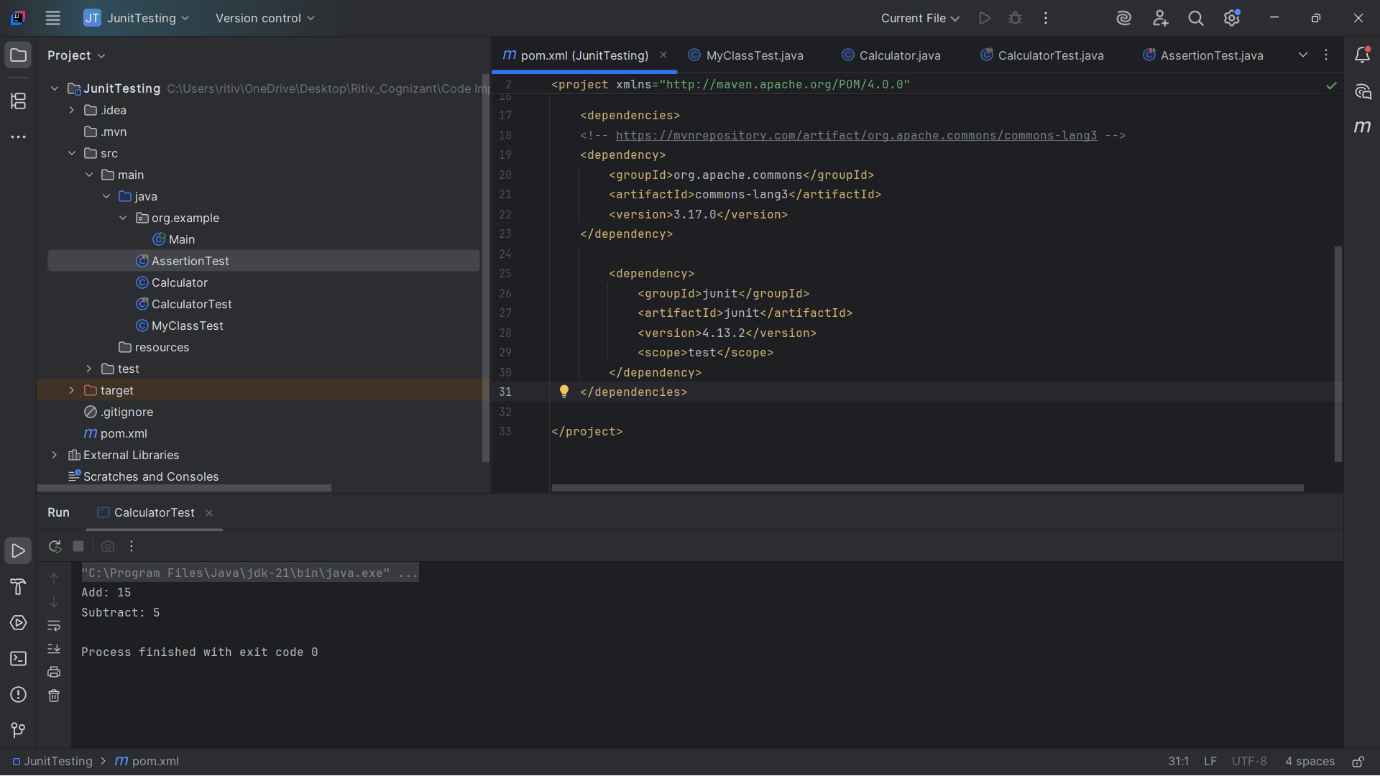
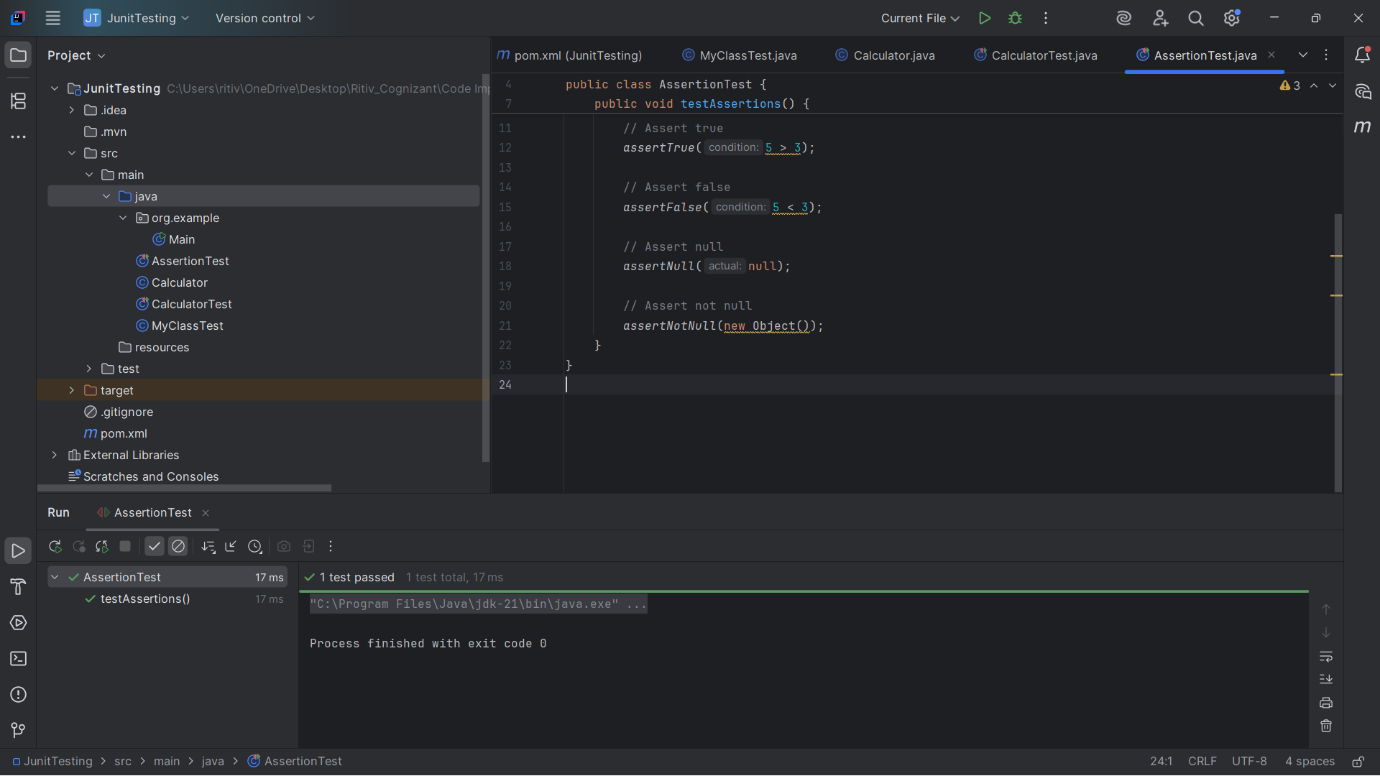
        // Assert not null

        assertNotNull(new Object());

    }

}





//Mockito

File:MyService.java

public class MyService {  
 private ExternalApi api;  
  
 public MyService(ExternalApi api) {  
 this.api = api;  
 }  
  
 public String fetchData() {  
 return api.getData("user123");  
 }  
  
  
 public interface ExternalApi {  
 String getData(String user123);  
 }  
}

//File:MyServiceTest.java

import static org.mockito.Mockito.\*;  
import static org.junit.jupiter.api.Assertions.\*;  
  
import org.junit.jupiter.api.Test;  
import org.mockito.Mockito;  
  
public class MyServiceTest {  
  
 @Test  
 public void testExternalApi() {  
 MyService.ExternalApi mockApi = Mockito.*mock*(MyService.ExternalApi.class);  
 *when*(mockApi.getData("user123")).thenReturn("Mock Data");  
  
 MyService service = new MyService(mockApi);  
 String result = service.fetchData();  
  
 *assertEquals*("Mock Data", result);  
 *verify*(mockApi).getData("user123");  
 }  
}

//File:LoggingExample.java

import org.slf4j.Logger;  
import org.slf4j.LoggerFactory;  
  
public class LoggingExample {  
 private static final Logger *logger* = LoggerFactory.*getLogger*(LoggingExample.class);  
  
 public static void main(String[] args) {  
 *logger*.error("This is an error message");  
 *logger*.warn("This is a warning message");  
 *logger*.info("This is an info message");  
 *logger*.debug("This is a debug message");  
 *logger*.trace("This is a trace message");  
 }  
}

