**TDD using JUnit5 and Mockito, SL4**

**JUNIT5**

**Exercise 1: Setting Up Junit**

* Add below dependency to depandancies present in pom.xml

<dependency>

<groupId>junit</groupId>

<artifactId>junit</artifactId>

<version>4.13.2</version>

<scope>test</scope>

</dependency>

**HelloWorldTest.java**

package com.example;

import org.junit.Test;

public class HelloWorldTest {

*@Test*

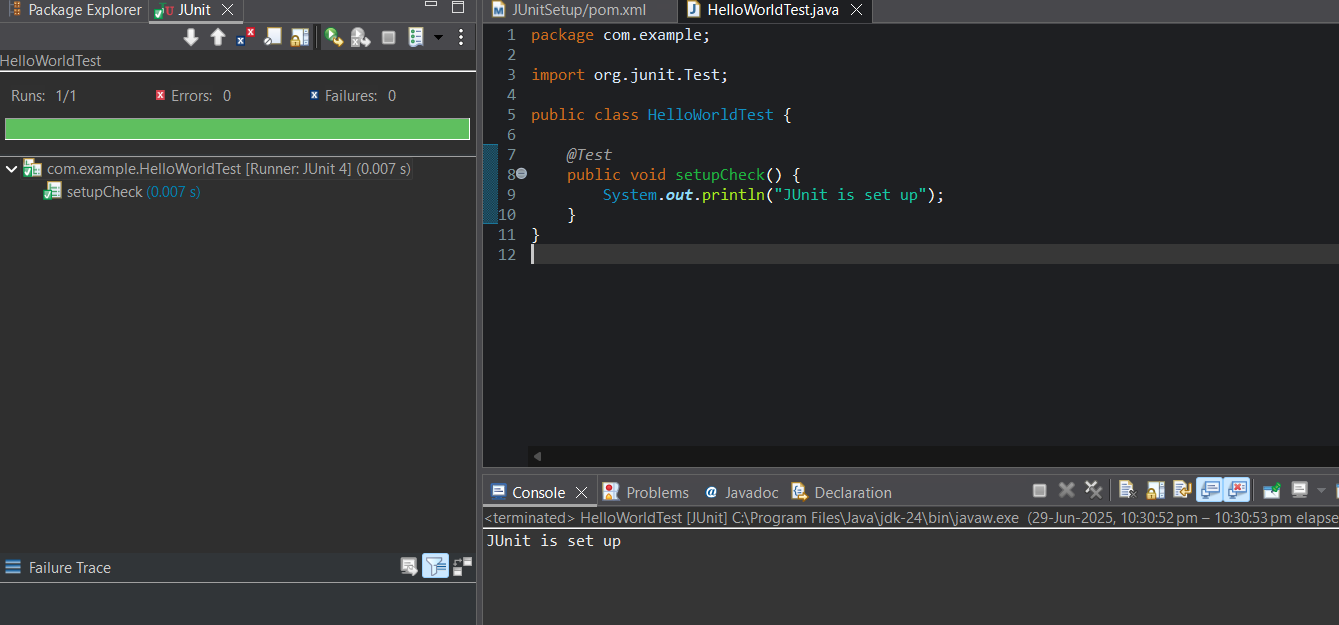
public void setupCheck() {

System.***out***.println("JUnit is set up");

}

}

**Output:**



**Exercise 3: Assertions in Junit**

package com.example;

import org.junit.Test;

import static org.junit.Assert.\*;

public class AssertionsTesting {

*@Test*

public void testAssertions() {

*assertEquals*("Sum should be 5", 5, 2 + 3);

*assertTrue*("5 is greater than 3", 5 > 3);

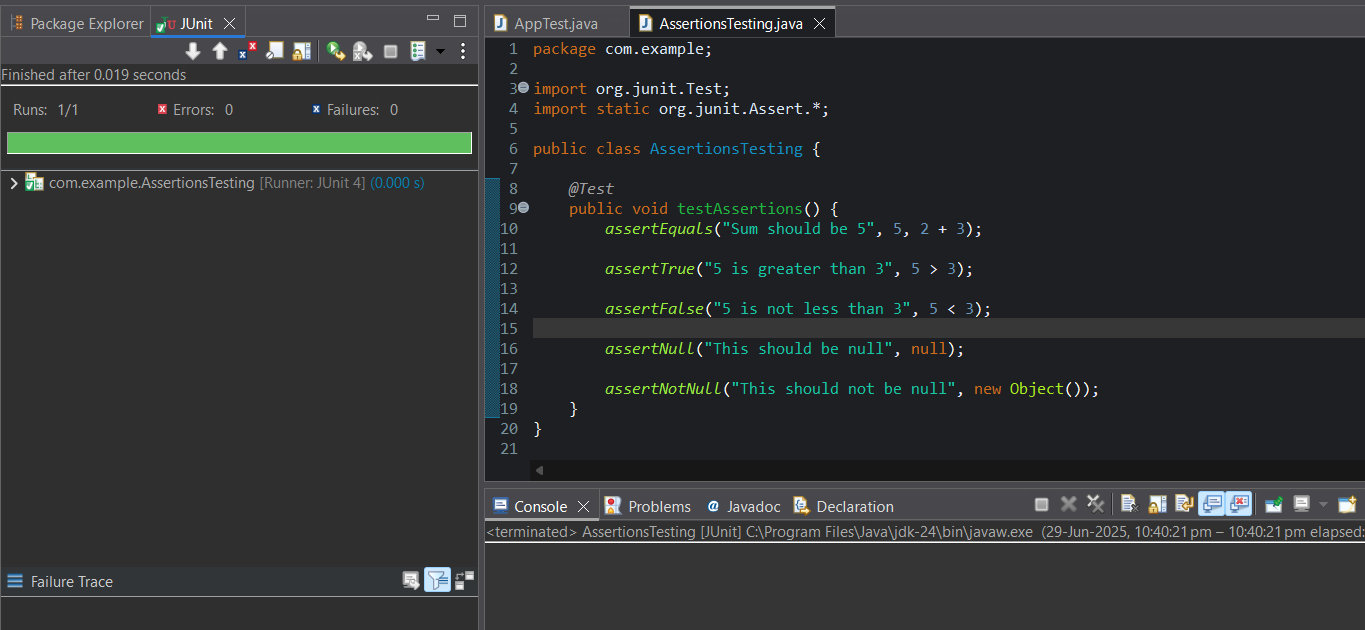
*assertFalse*("5 is not less than 3", 5 < 3);

*assertNull*("This should be null", null);

*assertNotNull*("This should not be null", new Object());

}}

**OUTPUT:**



**Exercise 4: Arrange-Act-Assert, Setup, Teardown**

*package com.example;*

*import static org.junit.Assert.\*;*

*import org.junit.Test;*

*import org.junit.After;*

*import org.junit.Before;*

*public class CalculatorTest {*

*private Calculator calculator;*

*@Before*

*public void setUp() {*

*calculator = new Calculator(); // Arrange*

*System.****out****.println(">> setUp() called: Calculator created");*

*}*

*@After*

*public void tearDown() {*

*calculator = null; // Cleanup*

*System.****out****.println("<< tearDown() called: Calculator destroyed");*

*}*

*@Test*

*public void testAddition() {*

*int result = calculator.add(2, 3); // Act*

*assertEquals("Sum of 2 + 3 must be 5", 5, result); // Assert*

*}*

*@Test*

*public void testSubtraction() {*

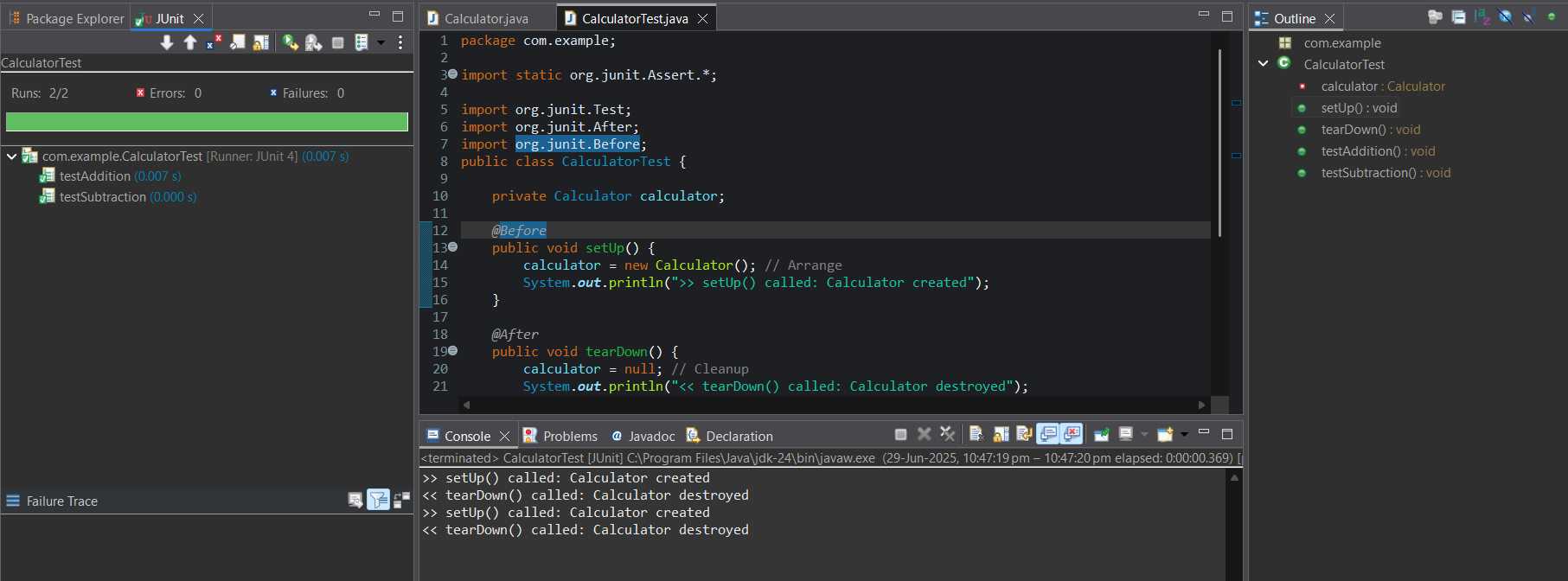
*int result = calculator.subtract(10, 4);*

*assertEquals("10 - 4 must be 6", 6, result);*

*}*

*}*

**Output:**

****

**Mockito**

**Exercise 1: Mocking and Stubbing**

**ExternalApi.java**

package com.example.mokito\_demo;

public class ExternalApi {

public String getData() {

return "Real Data";

}

}

**MyService.java**

package com.example.mokito\_demo;

public class MyService {

private ExternalApi api;

public MyService(ExternalApi api) {

this.api = api;

}

public String fetchData() {

return api.getData();

}

}

**MyServiceTest.java**

package com.example.mokito\_demo;

import org.junit.jupiter.api.Test;

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

import static org.mockito.Mockito.\*;

public class MyServiceTest {

*@Test*

public void testExternalApi() {

ExternalApi mockApi = *mock*(ExternalApi.class);

*when*(mockApi.getData()).thenReturn("Mock Data");

MyService service = new MyService(mockApi);

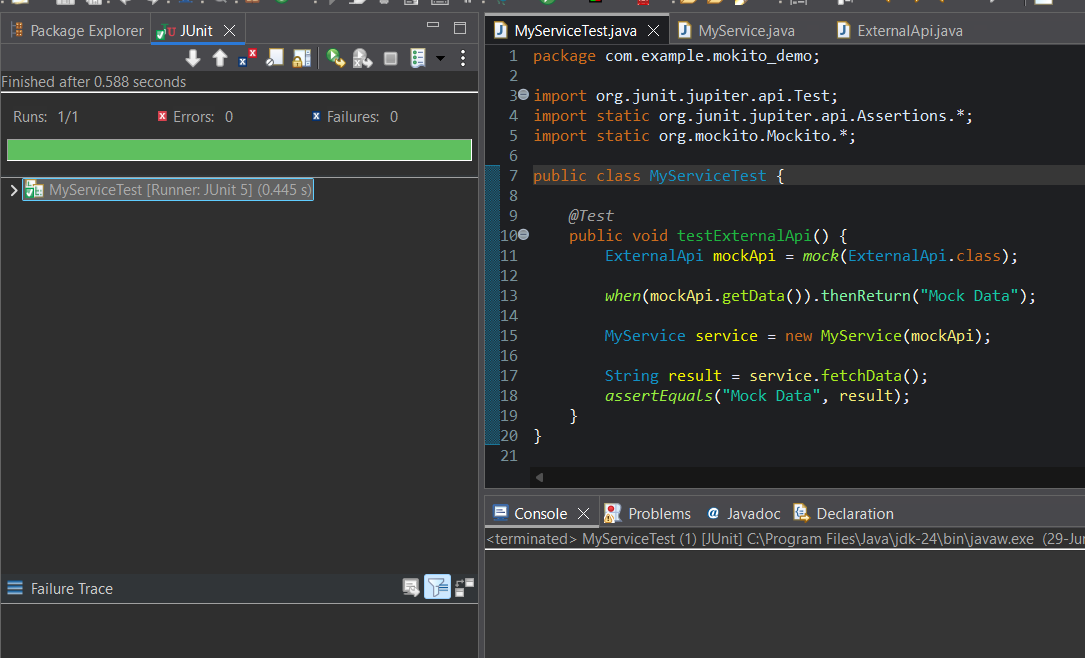
String result = service.fetchData();

*assertEquals*("Mock Data", result);

}

}

**OUTPUT:**

****

**Exercise 2: Verifying Interactions**

**ExternalApi.java**

package com.example.mokito\_demo;

public class ExternalApi {

public String getData() {

return "Real Data";

}

}

**MyService.java**

package com.example.mokito\_demo;

public class MyService {

private ExternalApi api;

public MyService(ExternalApi api) {

this.api = api;

}

public String fetchData() {

return api.getData();

}

}

**MyServiceTest.java**

package com.example.mokito\_demo;

import org.junit.jupiter.api.Test;

import static org.mockito.Mockito.\*;

public class MyServiceTest {

*@Test*

public void testExternalApi() {

ExternalApi mockApi = *mock*(ExternalApi.class);

MyService service = new MyService(mockApi);

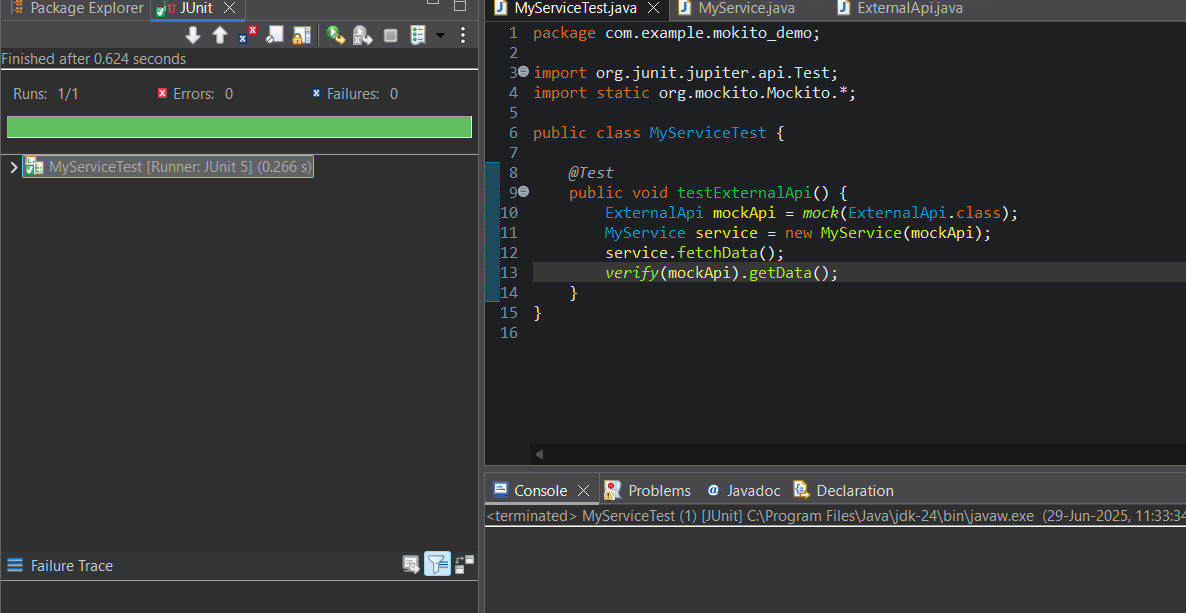
service.fetchData();

*verify*(mockApi).getData();

}

}

**Output:**

****

**SLF4J**

**Exercise 1: Logging Error Messages and Warning Levels**

package com.example;

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");

}

}

Output:

