**WEEK 2 TDD using JUnit5 and Mockito**

**HANDS ON EXERCISE**

**MANDATORY**

**Exercise 1: Mocking and Stubbing:**

Scenario: You need to test a service that depends on an external API. Use Mockito to mock the external API and stub its methods.

**CODE:**

**ExternalApi.java**

package com.example.JunitDemo;

public interface ExternalApi {

String getData();

}

**MyService.java**

package com.example.JunitDemo;

public class MyService {

private final ExternalApi api;

public MyService(ExternalApi api) {

this.api = api;

}

public String fetchData() {

return api.getData();

}

}

**MyServiceTest.java**

package com.example.JunitDemo;

import org.junit.jupiter.api.Test;

import static org.mockito.Mockito.\*;

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

public class MyServiceTest {

@Test

public void testExternalApi() {

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

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

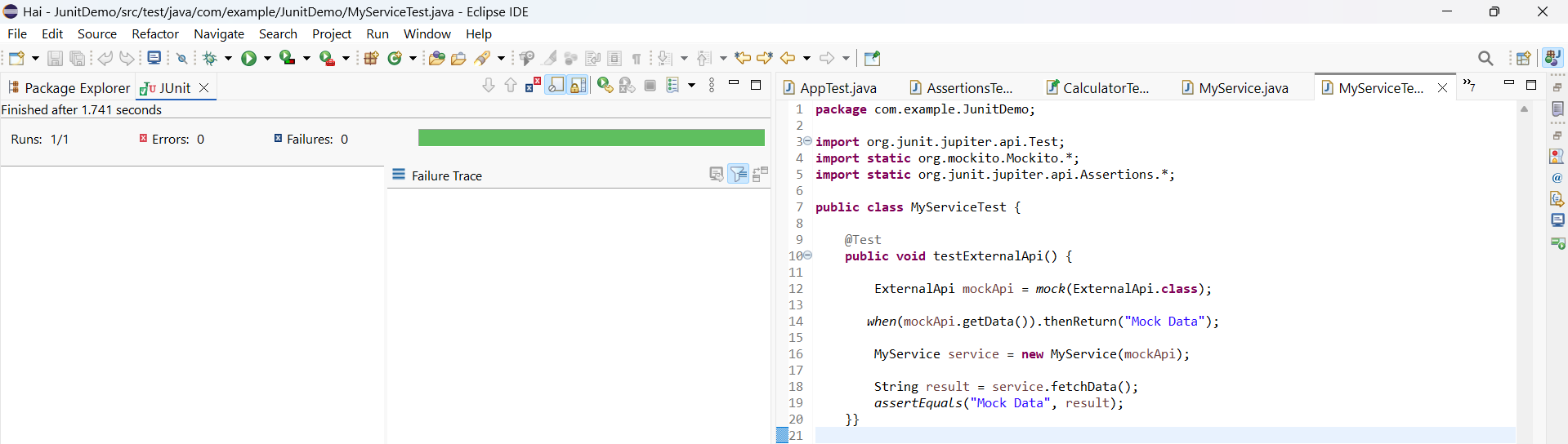
MyService service = new MyService(mockApi);

String result = service.fetchData();

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

}}

**OUTPUT for exercise 1: Mocking and Stubbing:**

****

**Exercise 2: Verifying Interactions**

Scenario: You need to ensure that a method is called with specific arguments.

**CODE:**

**ExternalApi.java**

package com.example.JunitDemo;

public interface ExternalApi {

String getData();

}

**MyService.java**

package com.example.JunitDemo;

public class MyService {

private final ExternalApi api;

public MyService(ExternalApi api) {

this.api = api;

}

public String fetchData() {

return api.getData();

}

}

**MyServiceTest.java**

package com.example.JunitDemo;

import static org.mockito.Mockito.\*;

import org.junit.jupiter.api.Test;

import org.mockito.Mockito;

public class MyServiceTest {

@Test

public void testVerifyInteraction() {

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

MyService service = new MyService(mockApi);

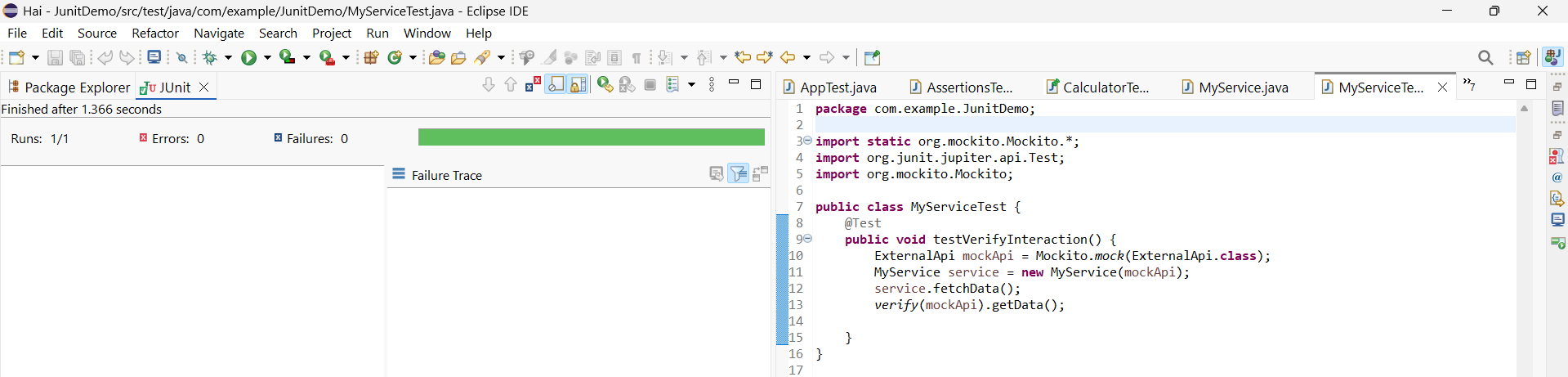
service.fetchData();

*verify*(mockApi).getData();

}

}

**OUTPUT for exercise 2: Verifying Interactions:**

****