# TDD using JUnit5 and Mockito

## Exercise 1: Mocking and Stubbing

### CODE:

ExternalApi.java

package com.example.demo;

public interface ExternalApi {

String getData();

}

MyService.java

package com.example.demo

public class MyService {

private ExternalApi externalApi;

public MyService(ExternalApi externalApi) {

this.externalApi = externalApi;

}

public String fetchData() {

return externalApi.getData();

}

}

MyServiceTest.java

package com.example.demo;

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

import static org.mockito.Mockito.\*

import org.junit.jupiter.api.Test;

import org.mockito.Mockito;

public class MyServiceTest {

@Test

public void testExternalApi() {

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

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

MyService service = new MyService(mockApi);

String result = service.fetchData();

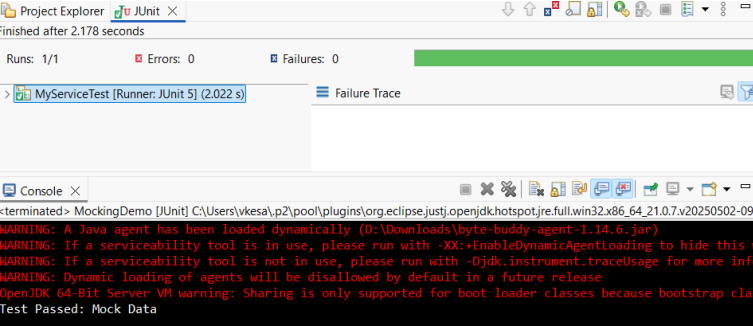
*assertEquals*("Mock Data", result);

System.*out*.println("Test Passed: " + result);

}

}

### OUTPUT:



## Exercise 2: Verifying Interactions

### CODE:

ExternalApi.java

package VERIFYINTERACTION;

public interface ExternalApi {

String getData();

}

MyService.java

package VERIFYINTERACTION;

public class MyService {

private ExternalApi externalApi;

public MyService(ExternalApi externalApi) {

this.externalApi = externalApi;

}

public String fetchData() {

return externalApi.getData();

}

}

MyServiceTest.java

package VERIFYINTERACTION;

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();

System.*out*.println("Verified: getData() was called on mockApi");

}

}

### OUTPUT:

