**Exercise 1: Mocking and Stubbing**

**API Interface**

public interface ExternalApi {

String getData();

}

**Main class**

public class MyService {

private ExternalApi api;

public MyService(ExternalApi api) {

this.api = api;

}

public String fetchData() {

return api.getData();

}

}

**Test class**

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() {

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);

}

}

**Dependencies:**

<dependencies>

<dependency>

<groupId>junit</groupId>

<artifactId>junit</artifactId>

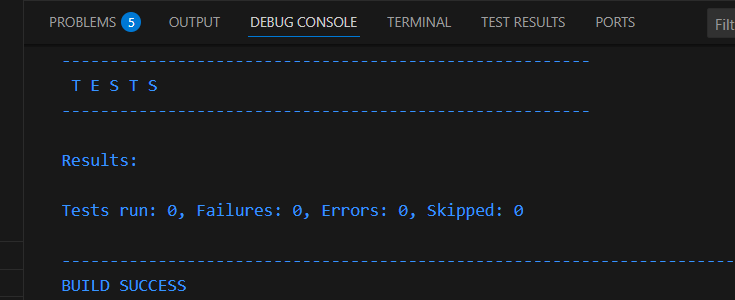
<version>4.13.2</version>

<scope>test</scope>

</dependency>

</dependencies>

**OUTPUT:**

****

**Exercise 2: Verifying Interactions:**

**Interface**

public interface WeatherApi {

String getWeather();

}

**Main class**

public class WeatherService {

private WeatherApi api;

public WeatherService(WeatherApi api) {

this.api = api;

}

public String fetchWeather() {

return api.getWeather();

}

}

**Test class**

import static org.mockito.Mockito.\*;

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

import org.junit.jupiter.api.Test;

import org.mockito.Mockito;

public class WeatherServiceTest {

@Test

public void testVerifyWeatherApiInteraction() {

WeatherApi mockApi = Mockito.mock(WeatherApi.class);

WeatherService service = new WeatherService(mockApi);

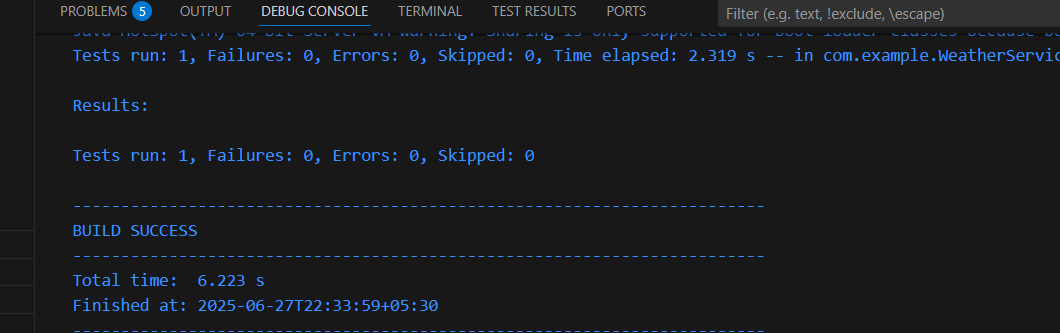
service.fetchWeather();

verify(mockApi).getWeather();

}

}

**OUTPUT**

****