**Exercise 2: Verifying Interactions**

**Scenario**:  
 You need to ensure that a method is called with specific argument  
 Code:

**Step1**:

Created a Maven project named Mockito-demo  
**Step2**:

Added required dependencies in pom.xml

**Step** **3**:

Created Interface named ExternalApi inside src/main/java  
package com.Mockito.Mockito\_Demo;

public interface ExternalApi {

String getData(String key);

}

**Step 4:**

Created java class named MyService inside src/main/java  
MyService.java  
package com.Mockito.Mockito\_Demo;

public class MyService {

private ExternalApi api;

public MyService(ExternalApi api)

{

this.api=api;

}

public String fetchdata(String key)

{

return api.getData(key);

}

}

**Step 5:**

Created MyserviceTest.java inside src/main/test  
package com.Mockito.Mockito\_Demo;

import static org.mockito.Mockito.*verify*;

import static org.mockito.Mockito.*when*;

import org.junit.jupiter.api.Test;

import org.mockito.Mockito;

public class MyServiceTest {

*@Test*

public void verifyingInteractions()

{

//step1:we are creating a mock object

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

//step 2:now we need to write a stub method that returns a value

*when*(mockapi.getData("user49")).thenReturn("Mock Data");

//step 3: inject and call the method

MyService service=new MyService(mockapi);

String result=service.fetchdata("user49");

//step 4:verifying interaction with specific argument

*verify*(mockapi).getData("user49");

//assertion

assert result.equals("Mock Data");

}

}

**Step 5:**

Below is the image of an output after execution of MyServiceTest

