**Exercise – 2 Verifying Interactions**

**ExternalApi.java**

package com.example;

public interface ExternalApi {

    String getData();

}

**MyService.java**

package com.example;

public class MyService {

    private ExternalApi api;

    // Constructor injection

    public MyService(ExternalApi api) {

        this.api = api;

    }

    public String fetchData() {

        // Calls the real or mocked API

        return api.getData();

    }

}

**MyServiceTest.java**

package com.example;

import static org.mockito.Mockito.\*;

import org.junit.Test;

import org.mockito.Mockito;

public class MyServiceTest {

    @Test

    public void testVerifyInteraction() {

        //Create a mock object

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

        //Use the mock in your service

        MyService service = new MyService(mockApi);

        service.fetchData();  // This should internally call mockApi.getData()

        //Verify that mockApi.getData() was actually called

        verify(mockApi).getData();

    }

}

**Result after verifying the interaction**

Verify(mockData).getData(); This is used to :

Verify that mockApi.getData() was actually called.

