**Mockito Hands-On Exercises**

**Exercise 1: Mocking and Stubbing**

**ExternalApi.java:**

package mock;

public interface ExternalApi {

String getData();

}

**MyService.java:**

package mock;

public class MyService {

private ExternalApi externalApi;

public MyService(ExternalApi externalApi) {

this.externalApi = externalApi;

}

public String fetchData() {

return externalApi.getData();

}

}

**MyServiceTest.java:**

package test;

import mock.ExternalApi;

import mock.MyService;

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);

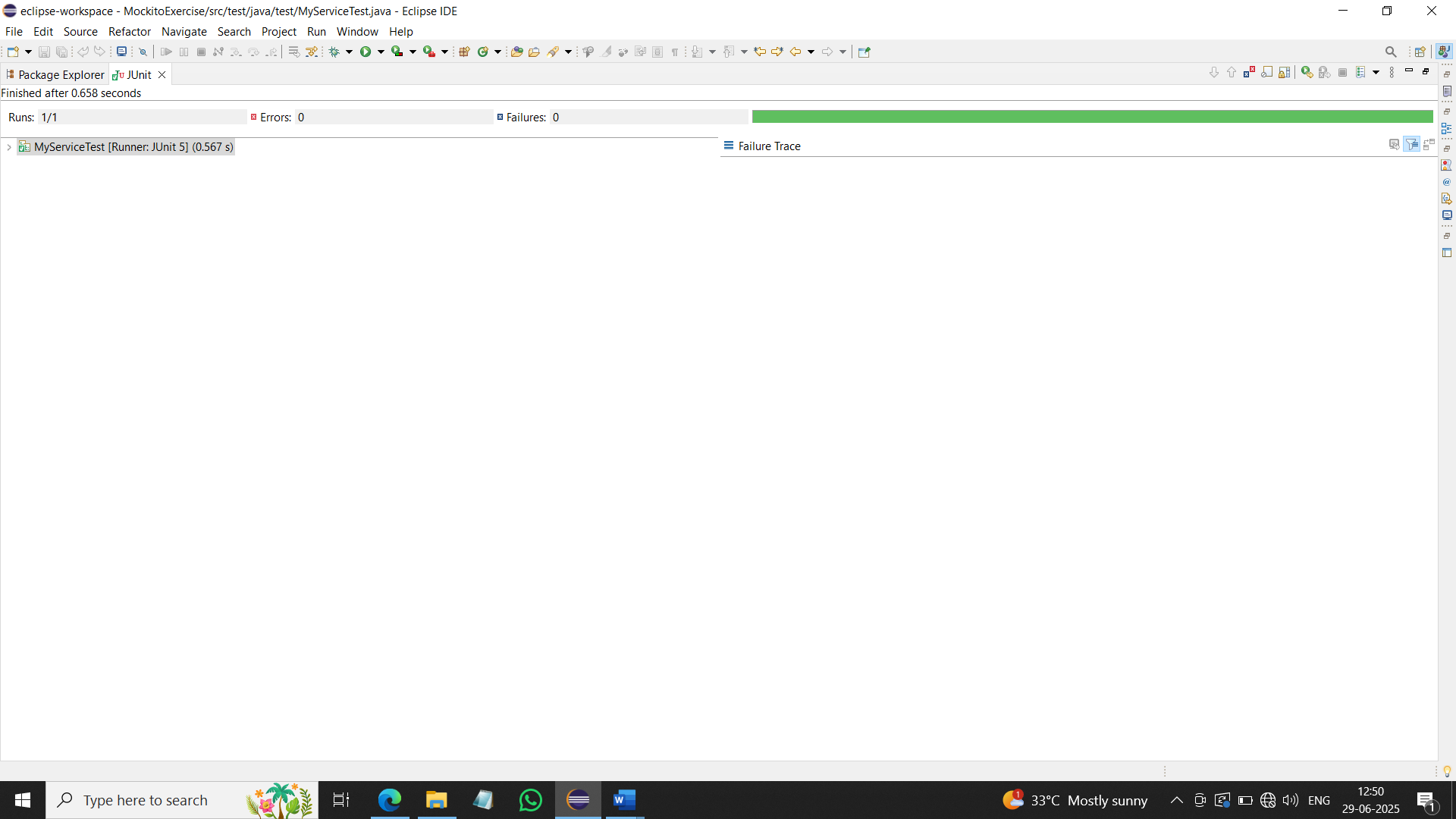
String result = service.fetchData();

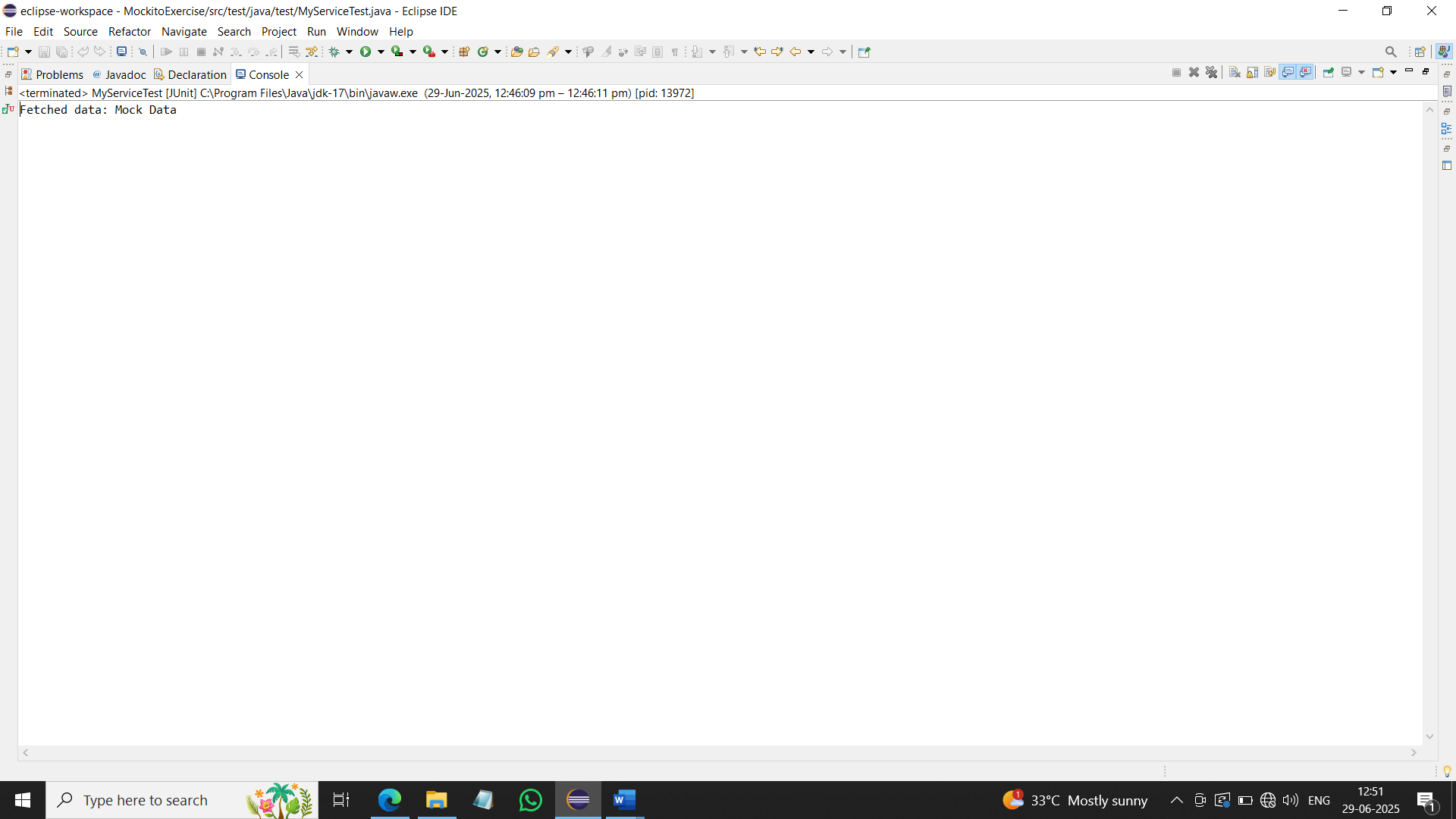
System.*out*.println("Fetched data: " + result);

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

}

}





**Exercise 2: Verifying Interactions**

**MyServiceVerificationTest.java:**

package test;

import mock.ExternalApi;

import mock.MyService;

import static org.mockito.Mockito.\*;

import org.junit.jupiter.api.Test;

import org.mockito.Mockito;

public class MyServiceVerificationTest {

@Test

public void testVerifyInteraction() {

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

MyService service = new MyService(mockApi);

service.fetchData();

*verify*(mockApi).getData();

}

}

