**Exercise 5 – Mocking and Stubbing**

**Objective:**

To implement mocking and stubbing using Mockito and test the interaction between a service class and an external dependency.

**Steps Followed:**

1. Created an interface ExternalApi to simulate external service behavior.
2. Created a class MyService that depends on ExternalApi.
3. Wrote a unit test MyServiceTest using Mockito to:
   * Create a mock of ExternalApi
   * Stub the getData() method to return mock data
   * Assert the response from MyService.fetchData()

**Java Code:**

**ExternalApi.java**

public interface ExternalApi {

String getData();

}

**MyService.java**

public class MyService {

private ExternalApi api;

public MyService(ExternalApi api) {

this.api = api;

}

public String fetchData() {

return api.getData();

}

}

**MyServiceTest.java**

import org.junit.jupiter.api.Test;

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

import static org.mockito.Mockito.\*;

public class MyServiceTest {

@Test

public void testFetchData() {

ExternalApi mockApi = mock(ExternalApi.class);

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

MyService service = new MyService(mockApi);

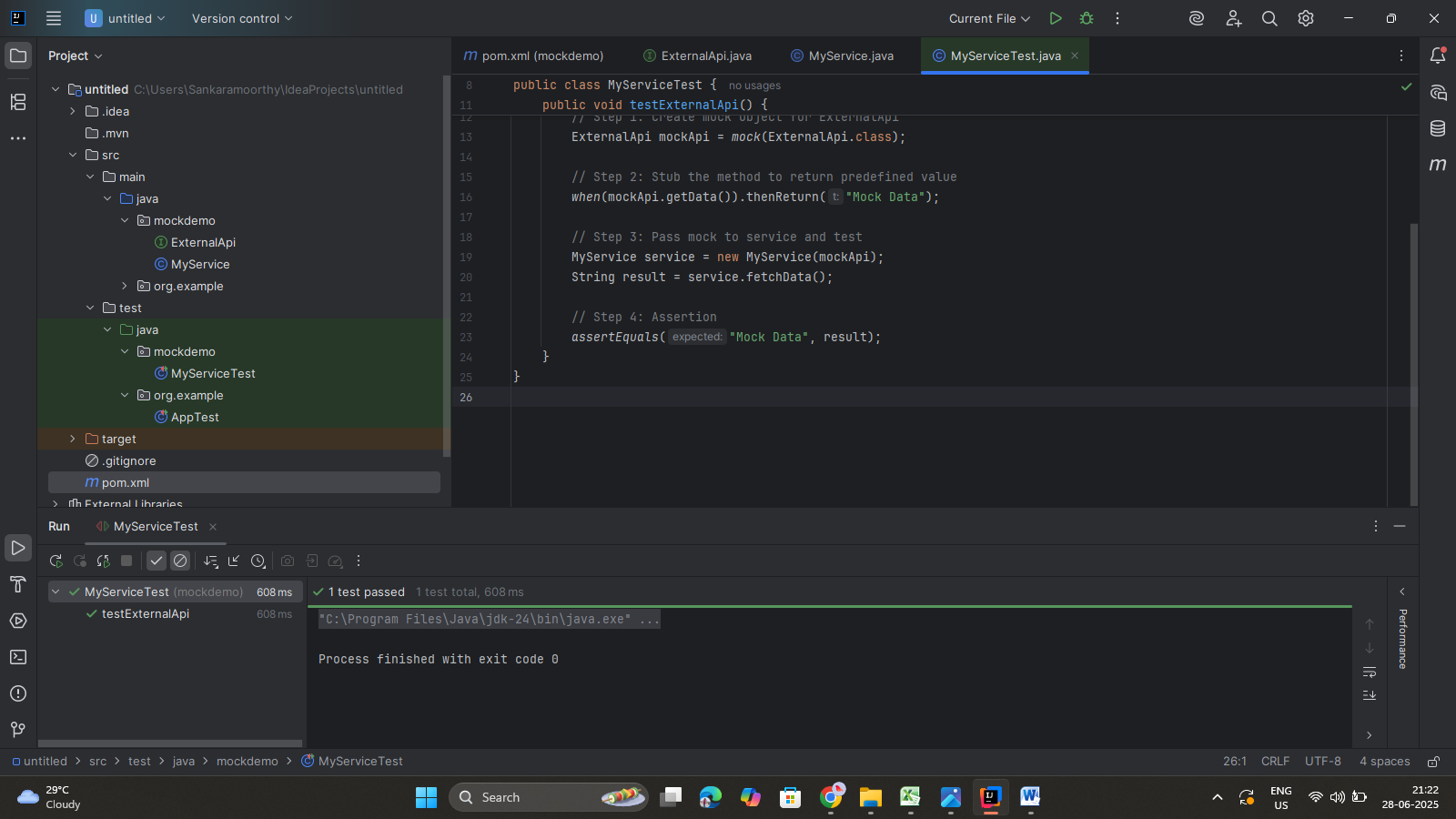
String result = service.fetchData();

assertEquals("Mock Data", result);

}

}

**Output Screenshot:**

****

**Tools Used:**

* IntelliJ IDEA
* JUnit 5
* Mockito (mocking + stubbing)