# **Exercise 1: Mocking and Stubbing**

## **Scenario**

You need to test a service that depends on an external API. To do this without calling the real API, you use Mockito to create a mock object and stub its methods to return predefined values.

### **Create a Mock Object**

I used Mockito to create a mock object for the ExternalApi interface.

### **Stub Methods**

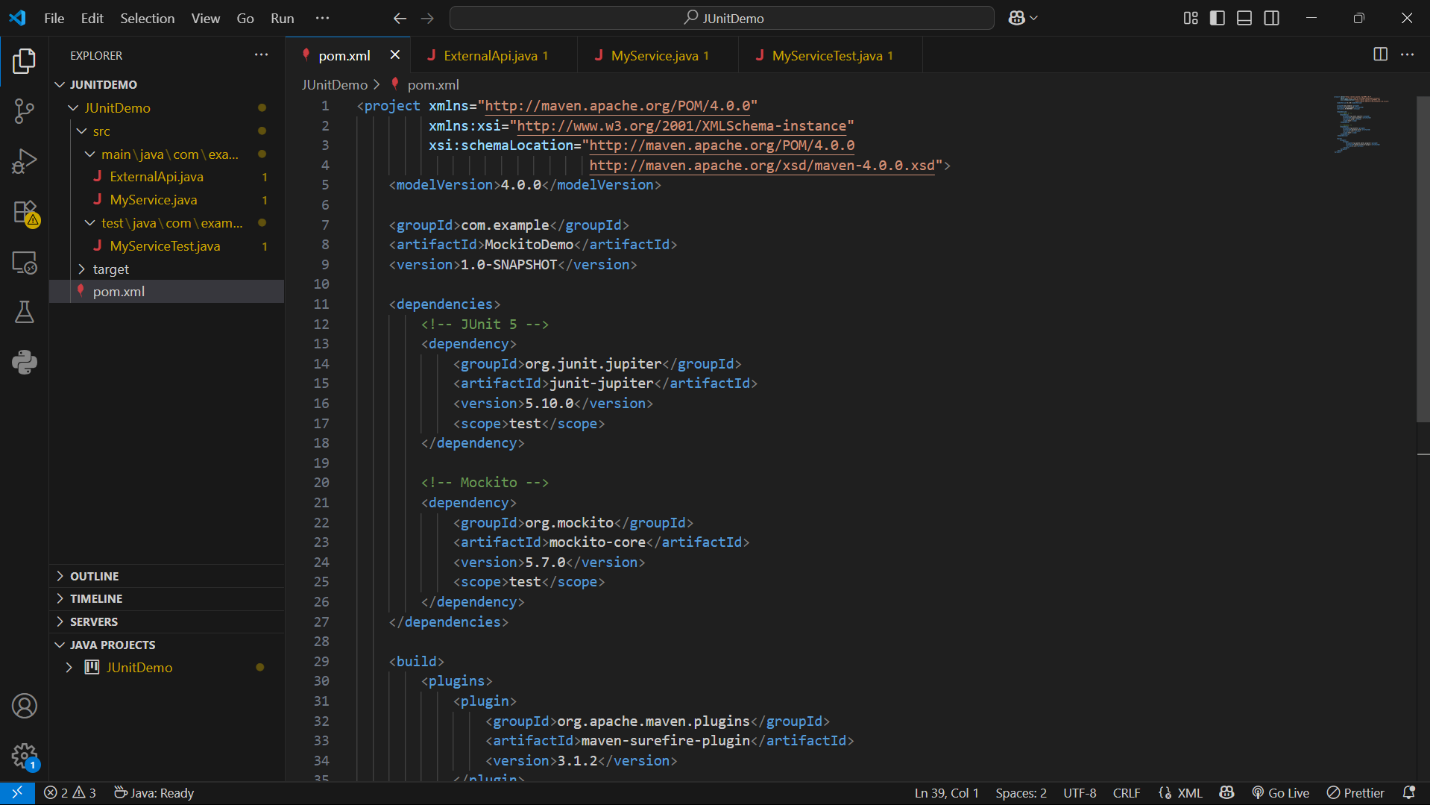
I used when(...).thenReturn(...) to define what the mock API should return when its method is called.

### **Write the Test**

I wrote a test case that uses the mock object instead of a real API and verifies the result.

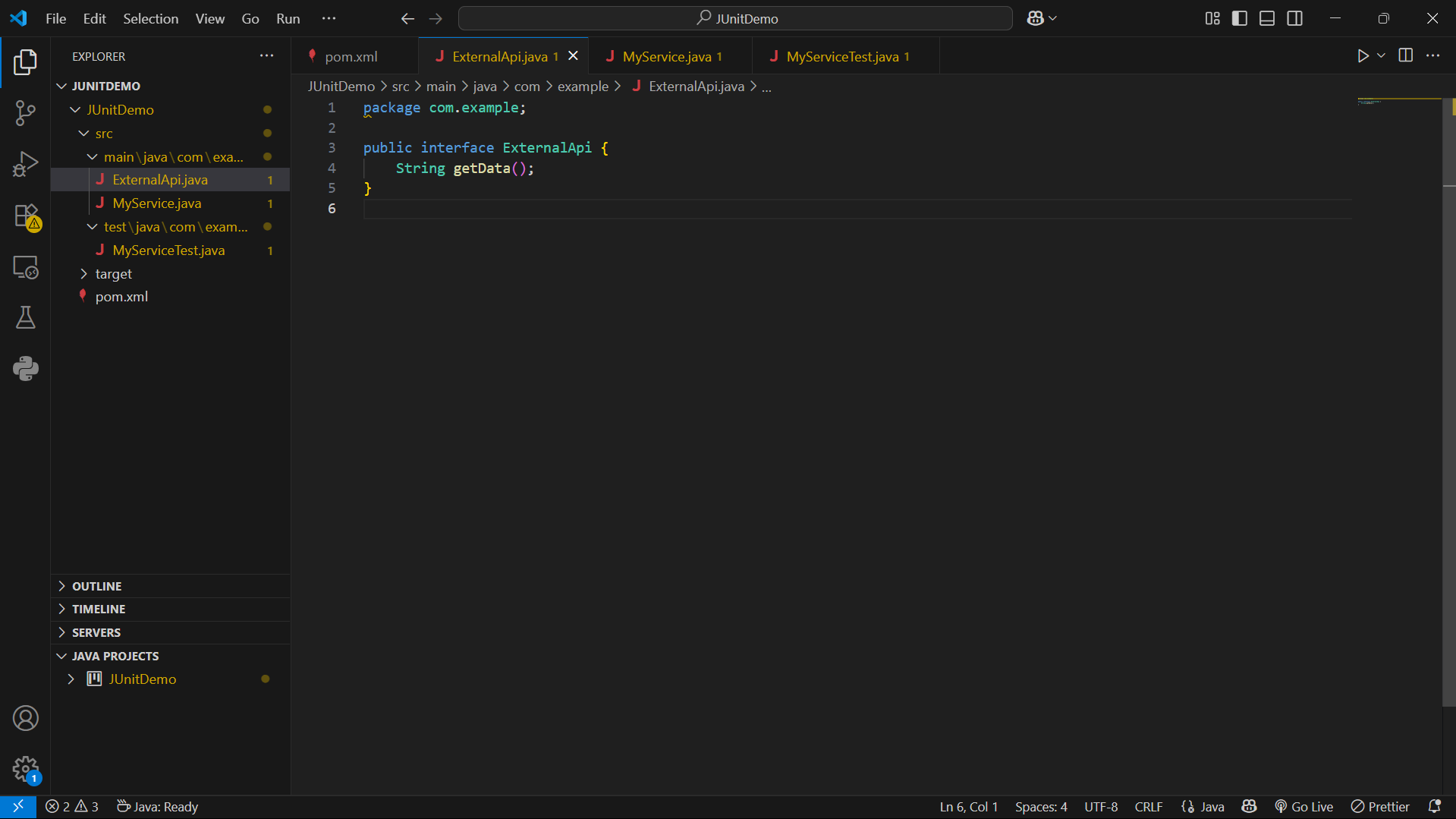
## **pom.xml**

<project xmlns="http://maven.apache.org/POM/4.0.0"  
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
 xsi:schemaLocation="http://maven.apache.org/POM/4.0.0  
 http://maven.apache.org/xsd/maven-4.0.0.xsd">  
 <modelVersion>4.0.0</modelVersion>  
  
 <groupId>com.example</groupId>  
 <artifactId>MockitoDemo</artifactId>  
 <version>1.0-SNAPSHOT</version>  
  
 <dependencies>  
 <!-- JUnit 5 -->  
 <dependency>  
 <groupId>org.junit.jupiter</groupId>  
 <artifactId>junit-jupiter</artifactId>  
 <version>5.10.0</version>  
 <scope>test</scope>  
 </dependency>  
  
 <!-- Mockito -->  
 <dependency>  
 <groupId>org.mockito</groupId>  
 <artifactId>mockito-core</artifactId>  
 <version>5.7.0</version>  
 <scope>test</scope>  
 </dependency>  
 </dependencies>  
  
 <build>  
 <plugins>  
 <plugin>  
 <groupId>org.apache.maven.plugins</groupId>  
 <artifactId>maven-surefire-plugin</artifactId>  
 <version>3.1.2</version>  
 </plugin>  
 </plugins>  
 </build>  
</project>



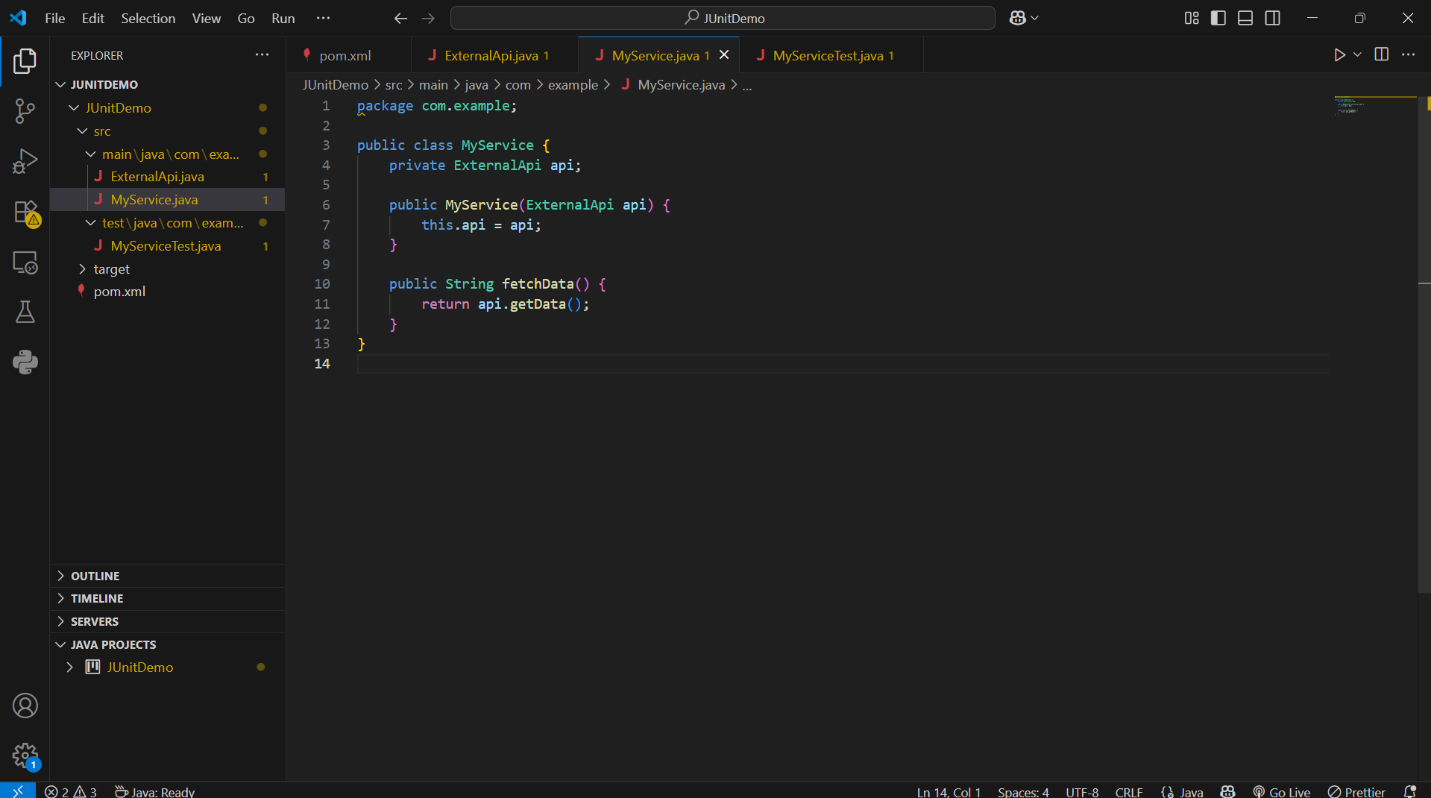
## **ExternalApi.java**

package com.example;  
  
public interface ExternalApi {  
 String getData();  
}



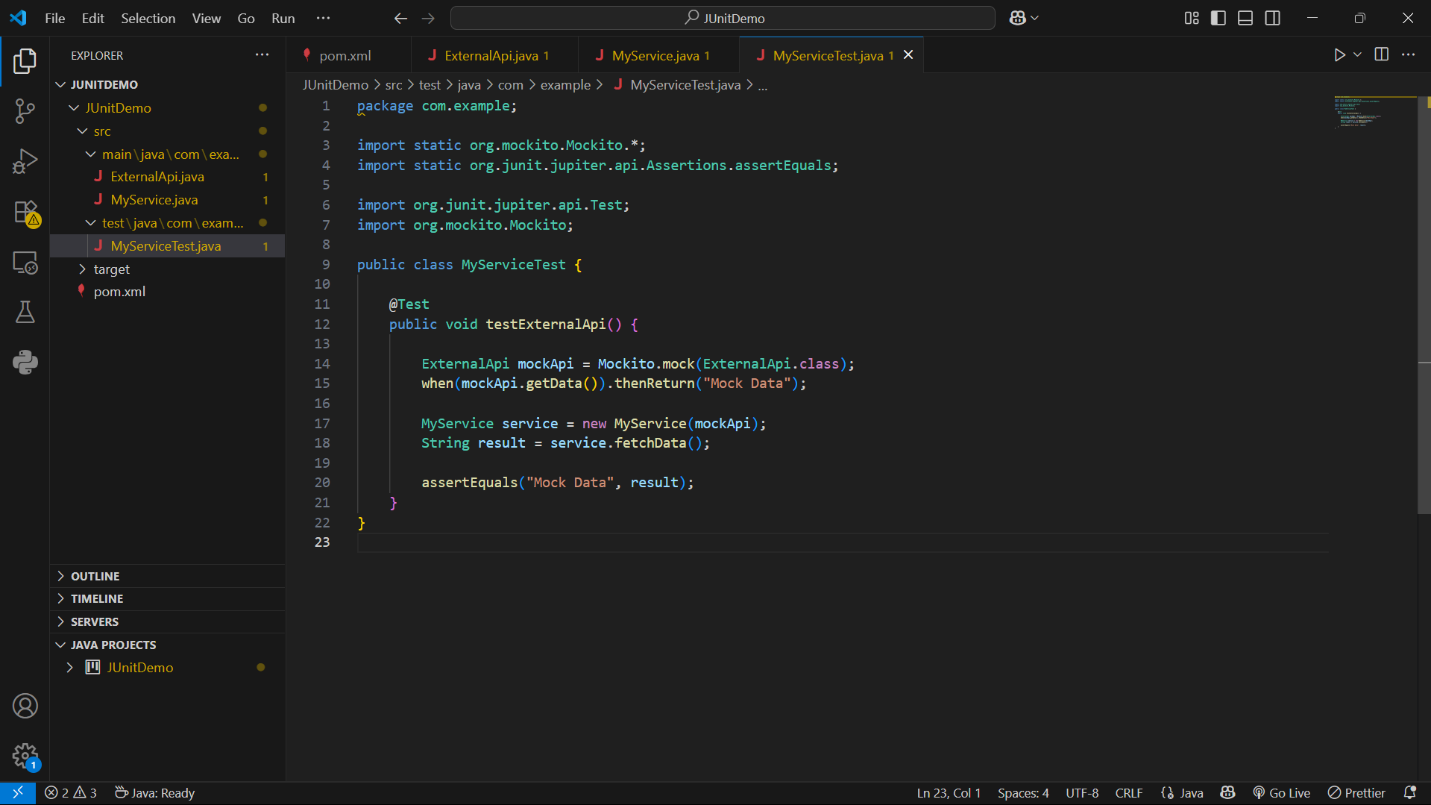
## **MyService.java**

package com.example;  
  
public class MyService {  
 private ExternalApi api;  
  
 public MyService(ExternalApi api) {  
 this.api = api;  
 }  
  
 public String fetchData() {  
 return api.getData();  
 }  
}



## **MyServiceTest.java**

package com.example;  
  
import static org.mockito.Mockito.\*;  
import static org.junit.jupiter.api.Assertions.assertEquals;  
  
import org.junit.jupiter.api.Test;  
import org.mockito.Mockito;  
  
public class MyServiceTest {  
  
 @Test  
 public void testExternalApi() {  
 // Arrange: Create mock & stub  
 ExternalApi mockApi = Mockito.mock(ExternalApi.class);  
 when(mockApi.getData()).thenReturn("Mock Data");  
  
 // Act: Use the mock in service  
 MyService service = new MyService(mockApi);  
 String result = service.fetchData();  
  
 // Assert: Verify result  
 assertEquals("Mock Data", result);  
 }  
}



## **How to Run**

In the VS Code terminal, inside the project folder, I ran:

mvn test

## 

## **Sample Output**

-------------------------------------------------------  
 T E S T S  
-------------------------------------------------------  
Running com.example.MyServiceTest  
Tests run: 1, Failures: 0, Errors: 0, Skipped: 0  
  
Results :  
  
Tests run: 1, Failures: 0, Errors: 0, Skipped: 0  
  
[INFO] BUILD SUCCESS

