Exercise 1: Mocking and Stubbing
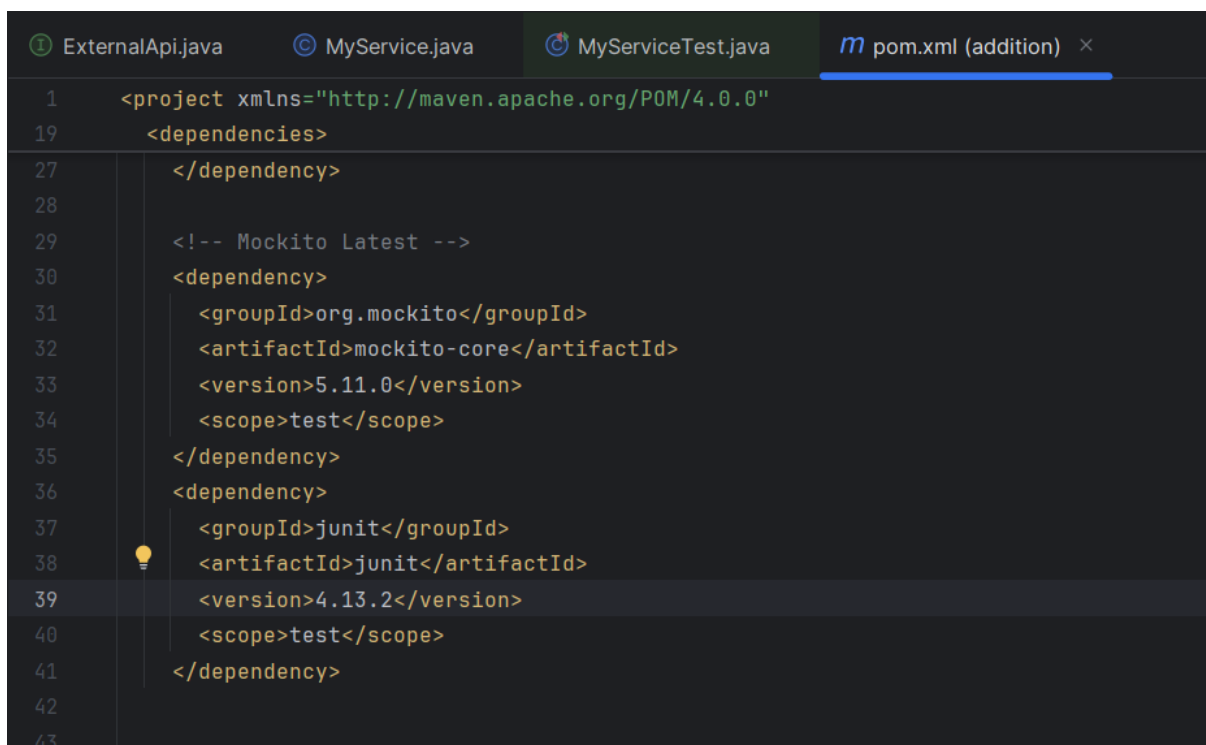
FROM FILE Mockito Exercises

Pom.xml:
Dependency:

```xml
<dependency>
  <groupId>org.mockito</groupId>
  <artifactId>mockito-core</artifactId>
  <version>5.11.0</version>
  <scope>test</scope>
</dependency>
```
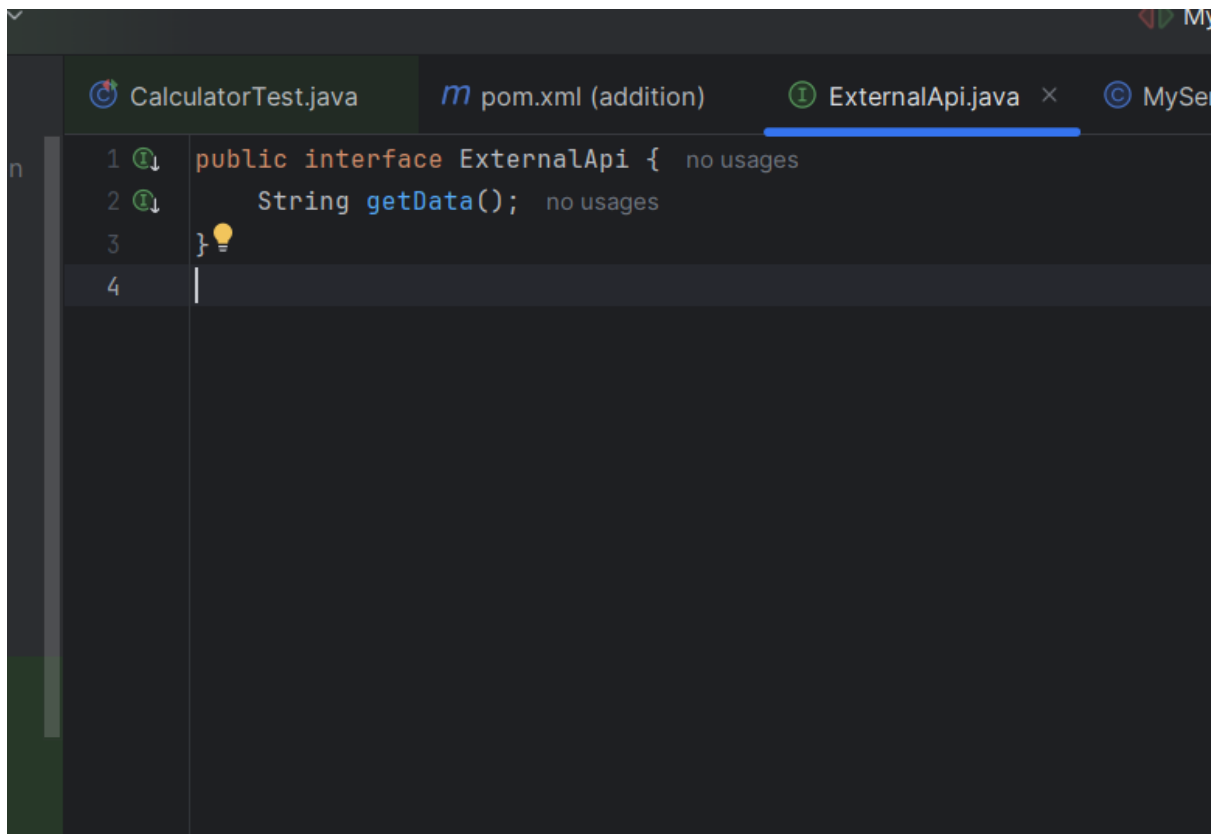
```xml
① ExternalApi.java      ⓒ MyService.java      MyServiceTest.java      m pom.xml (addition)  ×

1       <project xmlns="http://maven.apache.org/POM/4.0.0"
19        <dependencies>
27          </dependency>
28
29          <!-- Mockito Latest -->
30          <dependency>
31            <groupId>org.mockito</groupId>
32            <artifactId>mockito-core</artifactId>
33            <version>5.11.0</version>
34            <scope>test</scope>
35          </dependency>
36          <dependency>
37            <groupId>junit</groupId>
38            <artifactId>junit</artifactId>
39            <version>4.13.2</version>
40            <scope>test</scope>
41          </dependency>
42
43
```

ExternalApi.java class:

Code:
```java
public interface ExternalApi {
    String getData();}
```

Practice Screen from intellij:


```java
public interface ExternalApi {   no usages
    String getData();   no usages
}
```

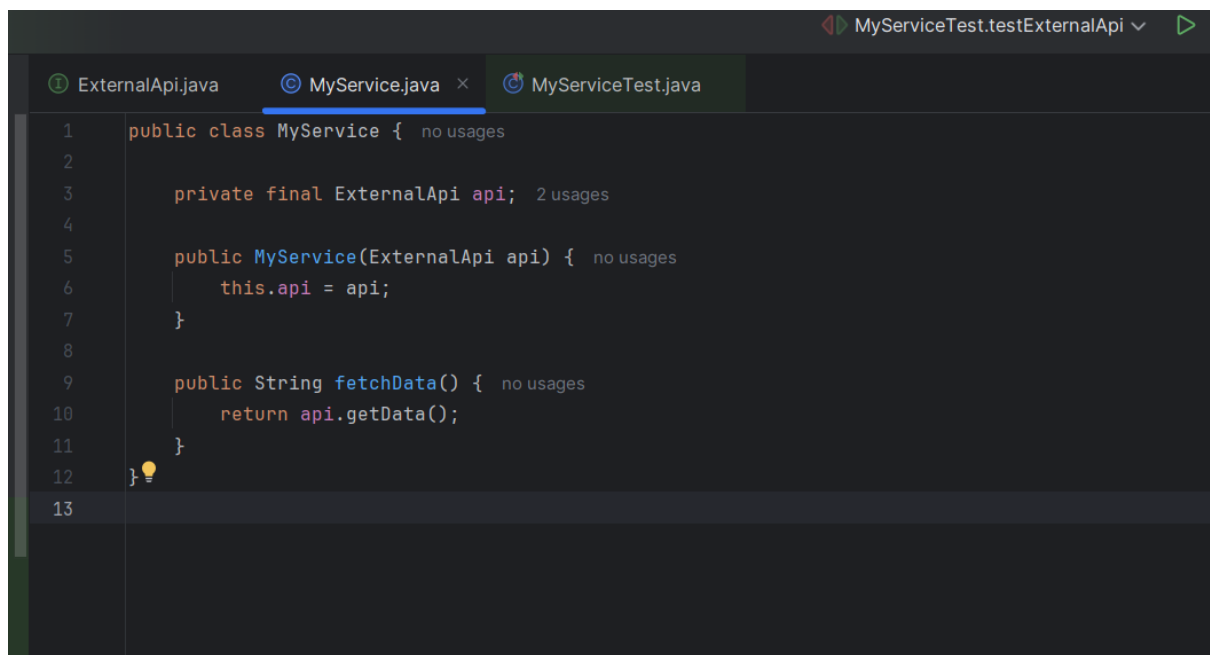**MyService.java class:**

Code:

```java
public class MyService {
    private final ExternalApi api;

    public MyService(ExternalApi api) {
        this.api = api;
    }

    public String fetchData() {
        return api.getData();
    }
}
```

Practice Screen from intellij:


```
                                                    ◀▷ MyServiceTest.testExternalApi ⌄    ▷
  ① ExternalApi.java        © MyService.java ✕      ◉ MyServiceTest.java
   1     public class MyService {  no usages
   2
   3         private final ExternalApi api;   2 usages
   4
   5         public MyService(ExternalApi api) {  no usages
   6             this.api = api;
   7         }
   8
   9         public String fetchData() {  no usages
  10             return api.getData();
  11         }
  12     }💡
  13
```

MyServiceTest.java class:

Code:

```java
import static org.mockito.Mockito.*;
import org.junit.jupiter.api.Test;
import org.mockito.Mockito;
import static org.junit.jupiter.api.Assertions.assertEquals;

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();
```
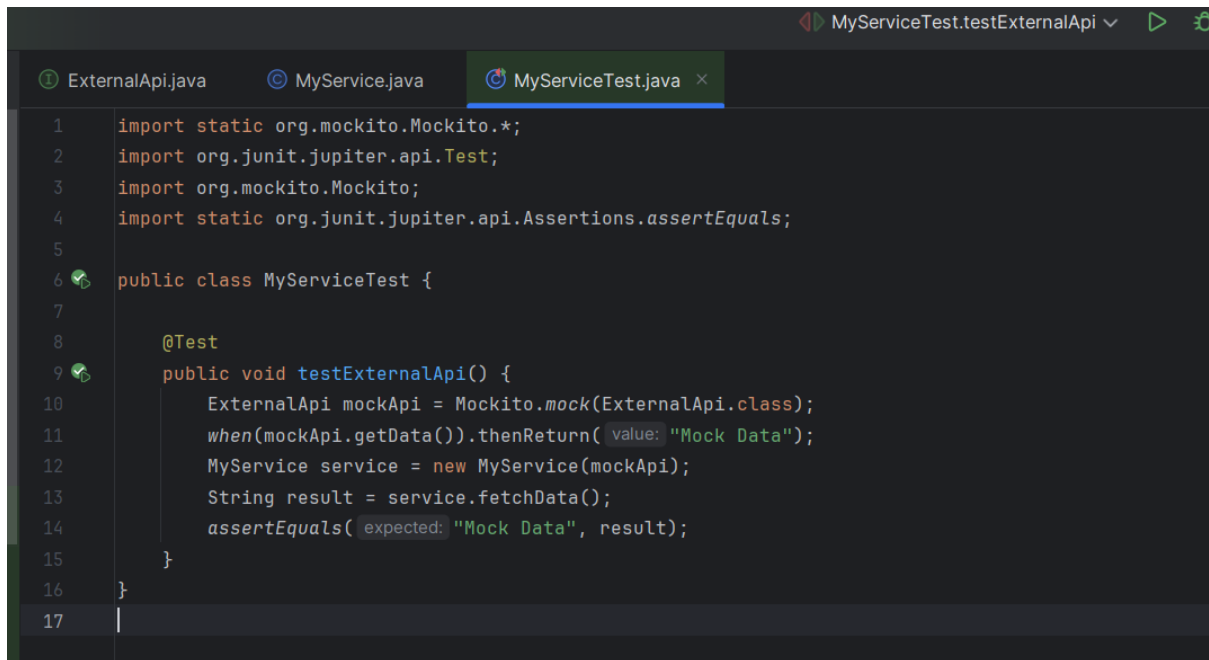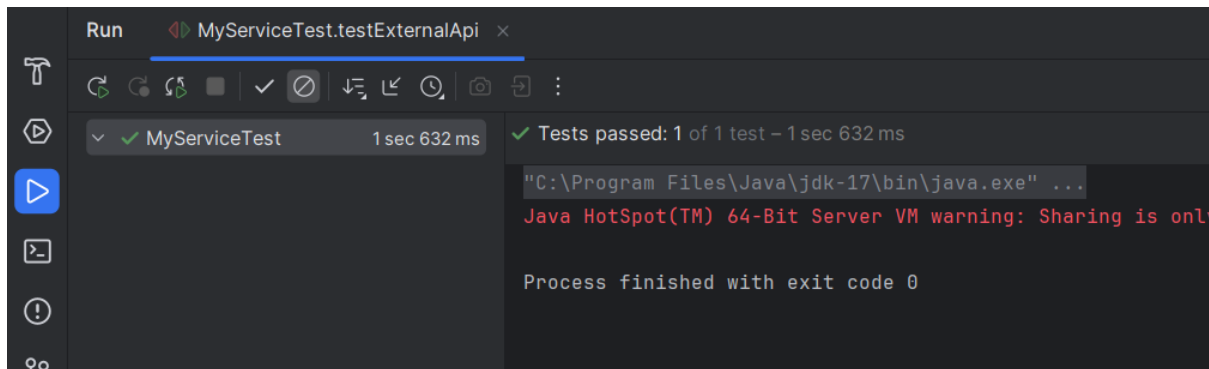
```
    assertEquals("Mock Data", result);
  }
}
```

Practice Screen from intellij:

Output after Run:

Process finished with exit code 0