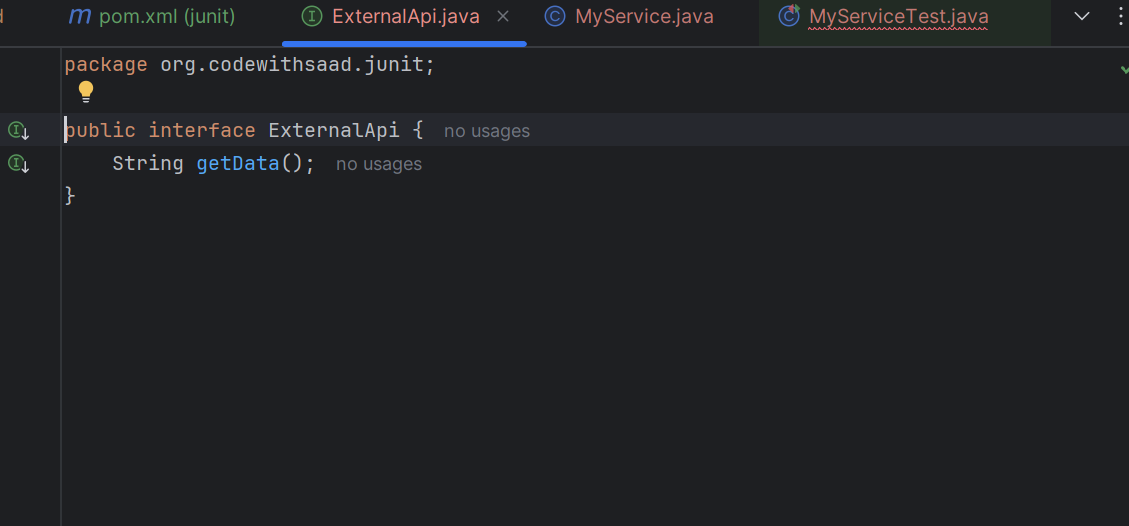
**Week 2: MOCKITO EXERCISES**

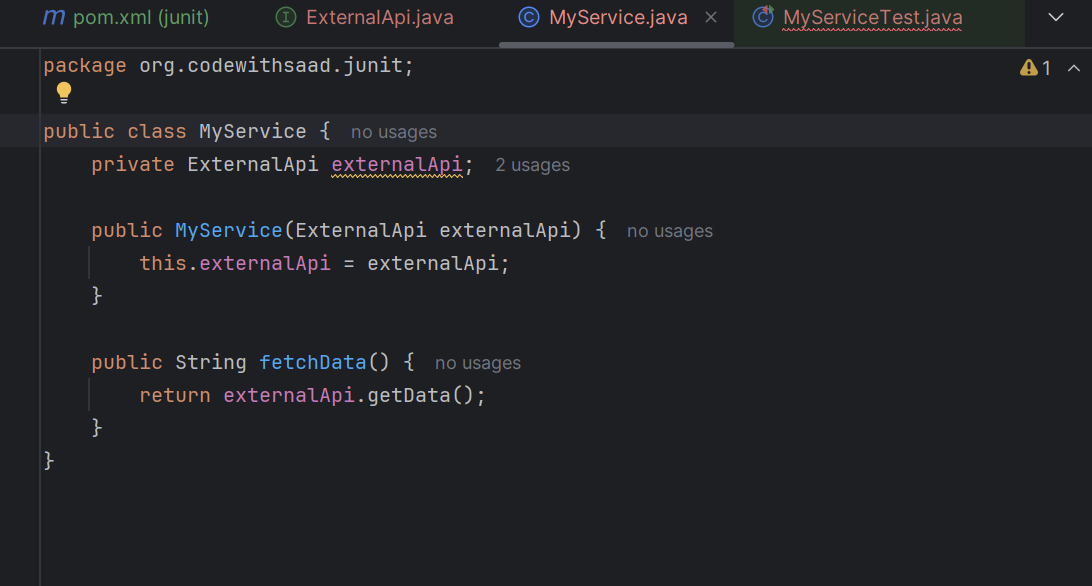
**Exercise 1: Mocking and Stubbing**

**Scenario:** You need to test a service that depends on an external API. Use Mockito to mock the external API and stub its methods.

**ExternalApi.java**



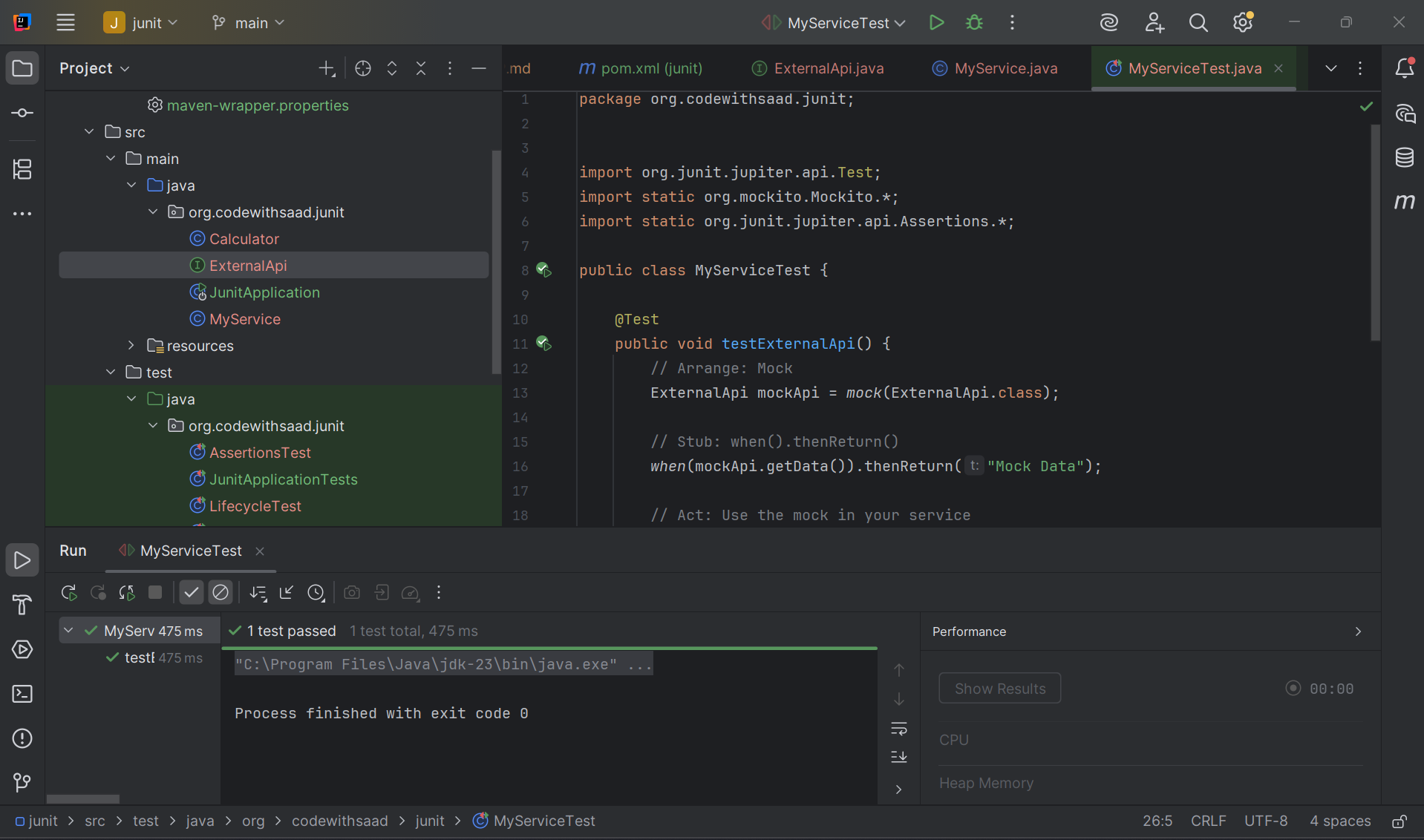
**MyServiceTest.java**



**MyServiceTest.java (testing code)**

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

**Output:**



**Exercise 2: Verifying Interactions**

**Scenario:** You need to ensure that a method is called with specific arguments.

**In the above testing code we add another piece of code,**

@Test

public void testVerifyInteraction() {

// Arrange: Mock

ExternalApi mockApi = mock(ExternalApi.class);

// Act: Use the mock

MyService service = new MyService(mockApi);

service.fetchData();

// Assert: Verify that getData() was called

verify(mockApi).getData();

}

**OUTPUT:**

