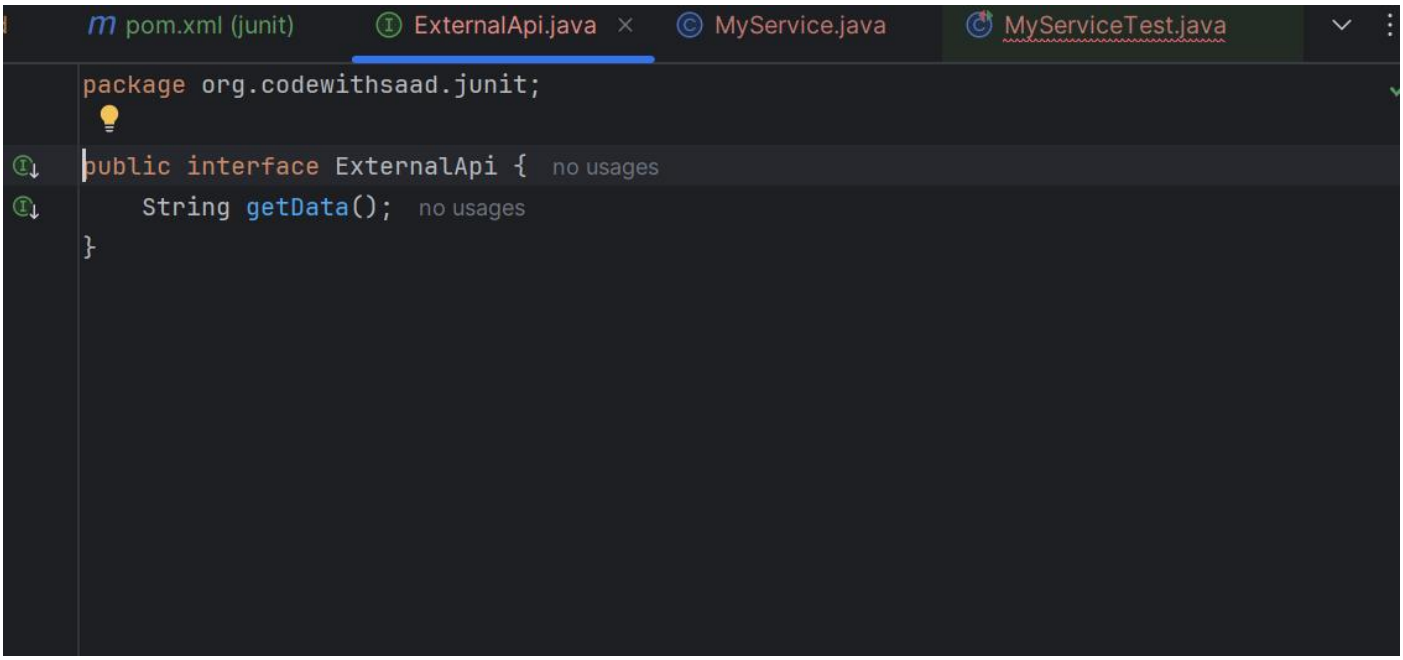
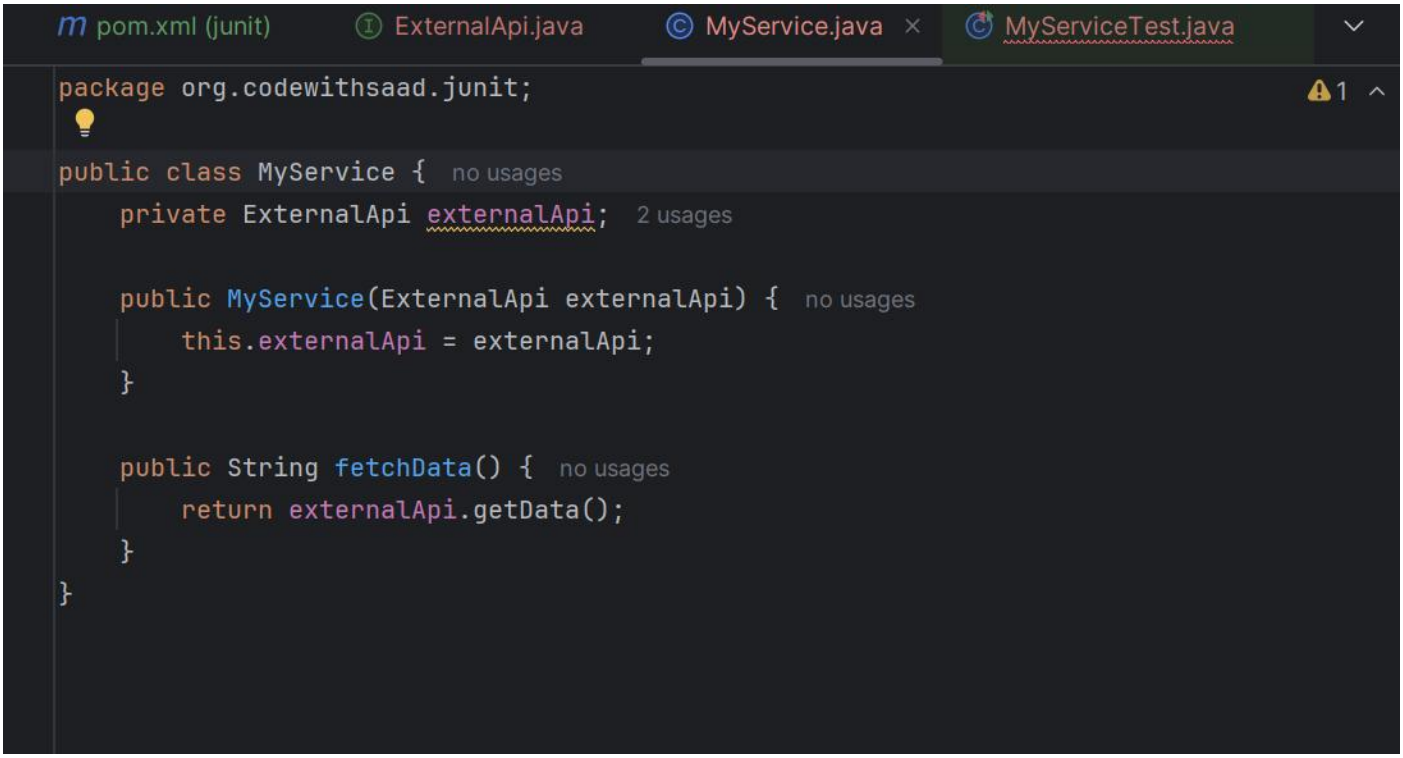
Mockito

Exercise 1: Mocking and Stubbing

ExternalApi.java



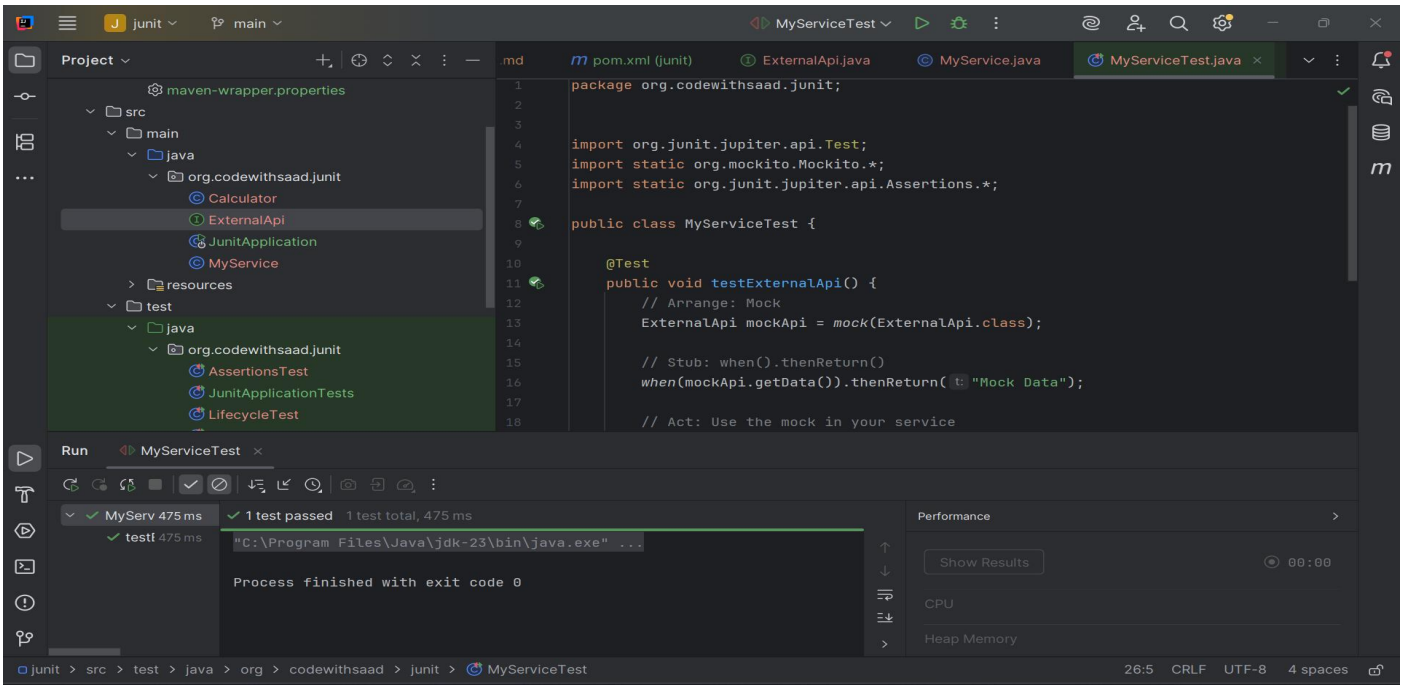
MyServiceTest.java



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:

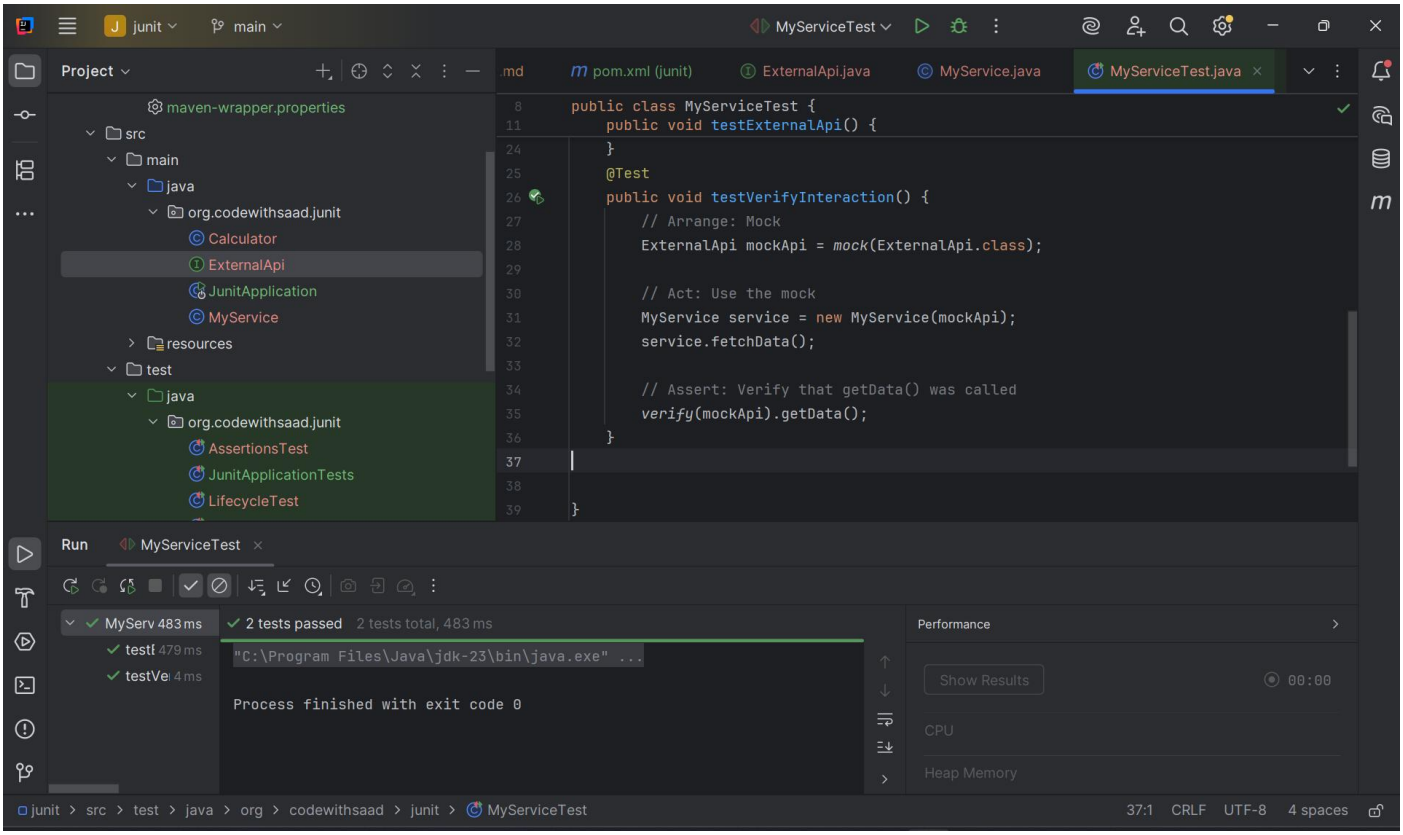


Exercise 2: Verifying Interactions

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:



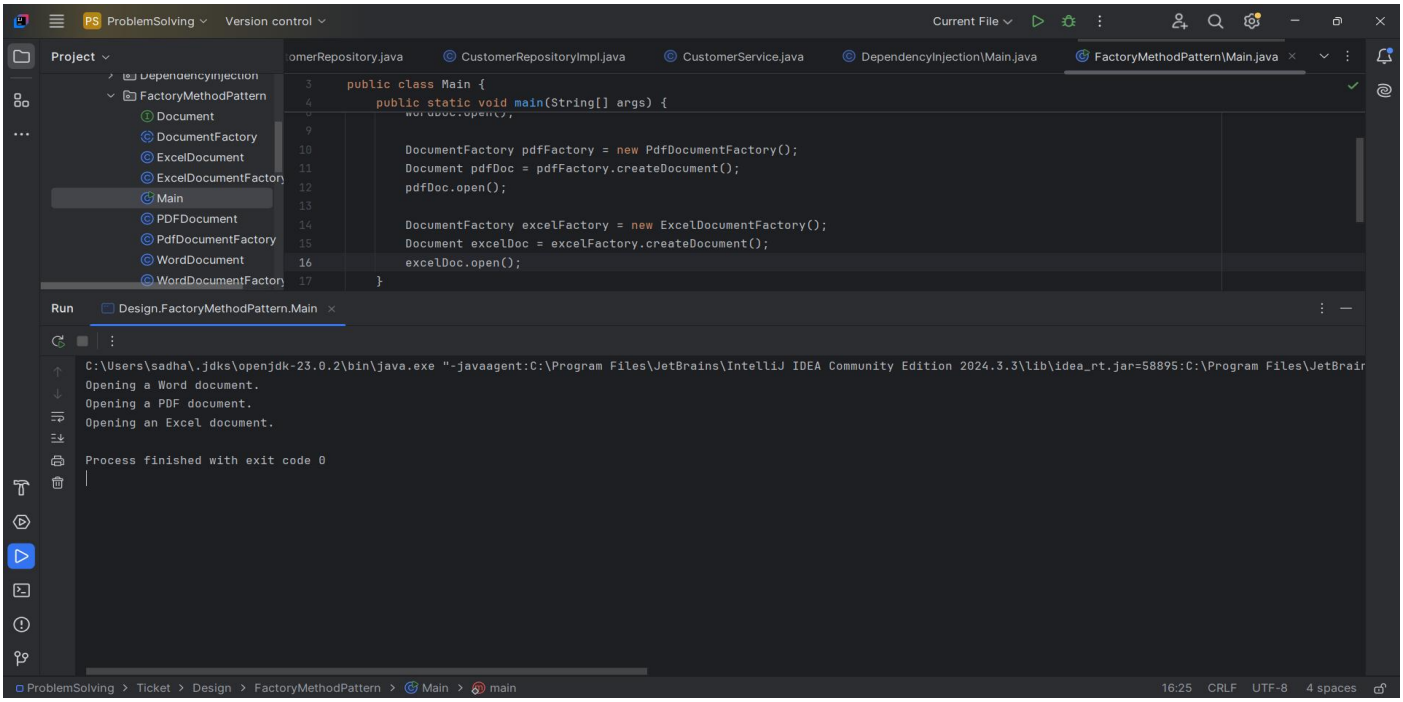
Worddocument.java:

```
package Design.FactoryMethodPattern;
public class WordDocument implements Document {
    @Override
    public void open() {
        System.out.println("Opening a Word document.");
    }
}
```

Worddocumentfactory.java:

```
package Design.FactoryMethodPattern;
public class WordDocumentFactory extends DocumentFactory {
    @Override
    public Document createDocument() {
        return new WordDocument();
    }
}
```

OUTPUT:


ProblemSolving
Version control
Current File
Project
FactoryMethodPattern
Document
DocumentFactory
ExcelDocument
ExcelDocumentFactory
Main
PDFDocument
PdfDocumentFactory
WordDocument
WordDocumentFactory
omerRepository.java
CustomerRepositoryImpl.java
CustomerService.java
DependencyInjection\Main.java
FactoryMethodPattern\Main.java
public class Main {
public static void main(String[] args) {
DocumentFactory pdfFactory = new PdfDocumentFactory();
Document pdfDoc = pdfFactory.createDocument();
pdfDoc.open();
DocumentFactory excelFactory = new ExcelDocumentFactory();
Document excelDoc = excelFactory.createDocument();
excelDoc.open();
}
Run
Design.FactoryMethodPattern.Main
C:\Users\sadha\.jdks\openjdk-23.0.2\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2024.3.3\lib\idea_rt.jar=58895:C:\Program Files\JetBrain
Opening a Word document.
Opening a PDF document.
Opening an Excel document.
Process finished with exit code 0
ProblemSolving
Ticket
Design
FactoryMethodPattern
Main
main
16:25
CRLF
UTF-8
4 spaces