**Module 4 – Test driven development and Logging framework**

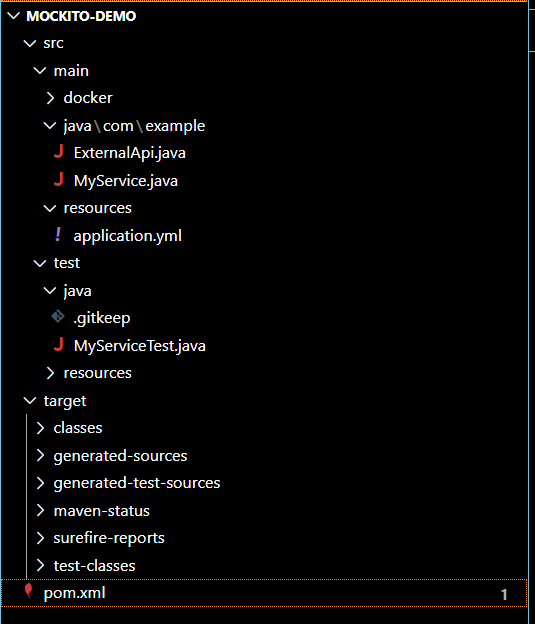
**Mockito Hands-On Exercises**

**Exercise 1: Mocking and Stubbing**

**Aim:**

To test a service that depends on an external API and by using Mockito to mock the external API and stub its methods.

**Structure:**



**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>mockito-demo</artifactId>

  <version>1.0-SNAPSHOT</version>

  <properties>

    <maven.compiler.source>11</maven.compiler.source>

    <maven.compiler.target>11</maven.compiler.target>

  </properties>

  <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>5.11.0</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> package com.example;

**ExternalApi.java:**

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 org.junit.jupiter.api.Test;

import static org.junit.jupiter.api.Assertions.\*;

import static org.mockito.Mockito.\*;

public class MyServiceTest {

    @Test

    public void testExternalApi() {

        // Arrange

        ExternalApi mockApi = mock(ExternalApi.class);

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

        // Act

        MyService service = new MyService(mockApi);

        String result = service.fetchData();

        // Assert

        assertEquals("Mock Data", result);

    }

}

  </build>

</project>

**Output:**

