**WEEK – 2 TDD USING JUNIT5 AND MOCKITO**

**EXERCISE 1: Mocking and Stubbing**

**DEPENDENCY:**

<dependencies>

<dependency>

<groupId>org.junit.jupiter</groupId>

<artifactId>junit-jupiter</artifactId>

<version>5.10.0</version>

<scope>test</scope>

</dependency>

<dependency>

<groupId>org.mockito</groupId>

<artifactId>mockito-core</artifactId>

<version>5.11.0</version>

<scope>test</scope>

</dependency>

<dependency>

<groupId>junit</groupId>

<artifactId>junit</artifactId>

<version>4.13.2</version>

<scope>test</scope>

</dependency>

</dependencies>

This dependency setup enables modern unit testing with **JUnit 5** and **Mockito**, while also supporting **JUnit 4 legacy test cases** — giving full flexibility for writing clean, isolated, and maintainable tests.

**DATAFETCHER.JAVA**

package com.assignment;

public interface DataFetcher {

String retrieveInfo();

}

**DATAHANDLER.JAVA**

java

CopyEdit

package com.assignment;

public class DataHandler {

private final DataFetcher dataFetcher;

public DataHandler(DataFetcher dataFetcher) {

this.dataFetcher = dataFetcher;

}

public String processData() {

return dataFetcher.retrieveInfo();

}

}

This class DataHandler depends on DataFetcher and uses **constructor injection** to receive it.  
The processData() method delegates the call to dataFetcher.retrieveInfo() and returns the result.

**DATAHANDLERTEST.JAVA**

package com.assignment;

import org.junit.jupiter.api.Test;

import static org.junit.jupiter.api.Assertions.assertEquals;

import static org.mockito.Mockito.\*;

public class DataHandlerTest {

@Test

public void testRetrieveInfo() {

DataFetcher mockFetcher = mock(DataFetcher.class);

when(mockFetcher.retrieveInfo()).thenReturn("Stubbed Response");

DataHandler handler = new DataHandler(mockFetcher);

String output = handler.processData();

assertEquals("Stubbed Response", output);

}

}

This test class verifies that DataHandler correctly calls DataFetcher and returns the expected output.  
It uses **Mockito** to mock the interface and **JUnit 5** to assert the value returned by processData().