**SLF4J logging framework**

**Exercise 1: Logging Error Messages and Warning Levels**

**Code:**

<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>SLF4JLoggerDemo</artifactId>

<version>1.0.0</version>

<properties>

<maven.compiler.source>17</maven.compiler.source>

<maven.compiler.target>17</maven.compiler.target>

</properties>

<dependencies>

<!-- SLF4J API -->

<dependency>

<groupId>org.slf4j</groupId>

<artifactId>slf4j-api</artifactId>

<version>2.0.9</version>

</dependency>

<!-- Simple backend implementation for SLF4J -->

<dependency>

<groupId>org.slf4j</groupId>

<artifactId>slf4j-simple</artifactId>

<version>2.0.9</version>

</dependency>

</dependencies>

</project>

**--------------------------------------------------------------------------------------------**

package com.example;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

public class LoggerExample {

private static final Logger ***logger*** = LoggerFactory.*getLogger*(LoggerExample.class);

public void processInput(String input) {

***logger***.info("Received input: {}", input);

if (input == null) {

***logger***.warn("Input is null. Defaulting to empty string.");

input = "";

}

if (input.isEmpty()) {

***logger***.error("Processing failed: Input cannot be empty.");

return;

}

***logger***.info("Processing input: {}", input.toUpperCase());

}

public static void main(String[] args) {

LoggerExample app = new LoggerExample();

app.processInput(null); // logs warn

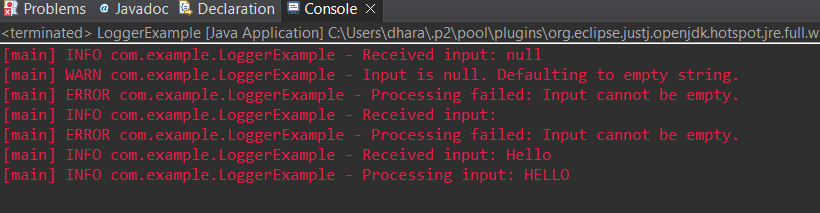
app.processInput(""); // logs error

app.processInput("Hello"); // logs info

}

}

**Output:**

****