**WEEK-02 HANDS ON SOLUTIONS**

**Logging using SLF4J**

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

Write a Java application that demonstrates logging error messages and warning levels using SLF4J.

Step-by-Step Solution:

1. Add SLF4J

and Logback dependencies to your `pom.xml` file: org.slf4j slf4j-api 1.7.30 ch.qos.logback logback-classic 1.2.3 2.

Create a Java class that uses SLF4J for logging:

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

public class LoggingExample

{

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

public static void main(String[] args)

{ logger.error("This is an error message");

logger.warn("This is a warning message");

}

}

**EXPLANATION:**

In this exercise, the goal was to demonstrate logging in a Java application using the SLF4J (Simple Logging Facade for Java) API, backed by the Logback implementation.

To begin, I added the required dependencies to the Maven project’s pom.xml file.

These included slf4j-api (version 1.7.30) for the logging interface and logback-classic (version 1.2.3) as the underlying implementation.

Maven automatically handled downloading and linking these libraries to my project.

Next, I created a Java class named LoggingExample where I initialized a logger instance using SLF4J’s LoggerFactory.getLogger() method.

Inside the main method, I used the logger to print an error message and a warning message to the console using logger.error() and logger.warn() respectively.

After running the program, the output in the console clearly displayed the log messages with the correct severity levels, along with the class name and logger prefix, which confirmed successful integration.

**OUTPUT**

