

# Handson 10 : Logging Error Messages and Warning Levels

This exercise is all about learning how to log messages in your Java application — especially errors and warnings — using a popular logging library called SLF4J + Logback.

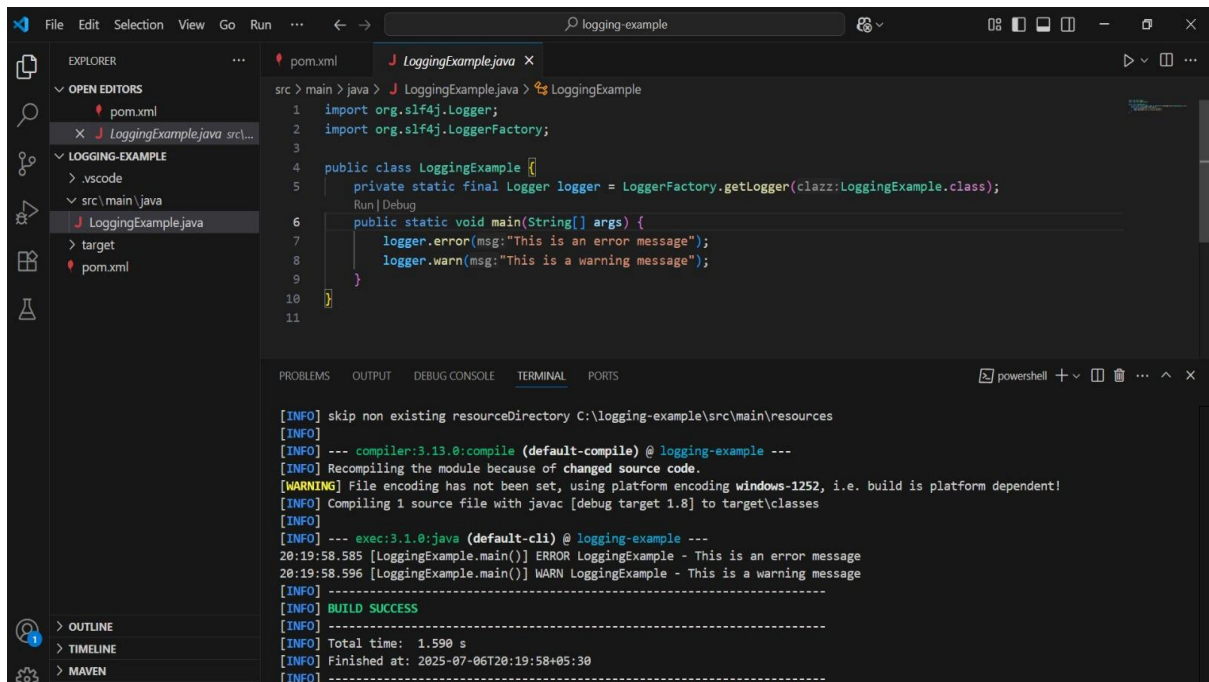
SLF4J is a interface or a middleman that lets you use any logging backend (like Logback, Log4J, etc.).

Logback is ,It's the actual logging engine here.

## **understanding from this :**

- ☐ Logging helps you track problems, understand app behavior, and debug faster.
- ☐ Using different levels (like error, warn) makes logs way more readable and useful.
- ☐ This is literally must-know stuff for every backend dev, no matter the size of the project.

# OUTPUT SNAPSHOTS:



The screenshot displays the Visual Studio Code interface for a project named 'logging-example'. The Explorer sidebar on the left shows the project structure, including 'pom.xml', 'src/main/java', and 'target'. The main editor window shows the 'LoggingExample.java' file with the following code:

```
src > main > java > J LoggingExample.java > LoggingExample
1 import org.slf4j.Logger;
2 import org.slf4j.LoggerFactory;
3
4 public class LoggingExample {
5     private static final Logger logger = LoggerFactory.getLogger(clazz::LoggingExample.class);
6     Run | Debug
7     public static void main(String[] args) {
8         logger.error(msg:"This is an error message");
9         logger.warn(msg:"This is a warning message");
10    }
11 }
```

The Terminal panel at the bottom shows the output of the build and execution process:

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
[INFO] skip non existing resourceDirectory C:\logging-example\src\main\resources
[INFO] --- compiler:3.13.0:compile (default-compile) @ logging-example ---
[INFO] Recompiling the module because of changed source code.
[WARNING] File encoding has not been set, using platform encoding windows-1252, i.e. build is platform dependent!
[INFO] Compiling 1 source file with javac [debug target 1.8] to target\classes
[INFO] --- exec:3.1.0:java (default-cli) @ logging-example ---
20:19:58.585 [LoggingExample.main()] ERROR LoggingExample - This is an error message
20:19:58.596 [LoggingExample.main()] WARN LoggingExample - This is a warning message
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 1.590 s
[INFO] Finished at: 2025-07-06T20:19:58+05:30
[INFO] -----
```