**SL4J Logging exercises**

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

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 https://maven.apache.org/xsd/maven-4.0.0.xsd">

<modelVersion>4.0.0</modelVersion>

<groupId>com.example</groupId>

<artifactId>LoggingDemo</artifactId>

<version>0.0.1-SNAPSHOT</version>

<dependencies>

<!-- SLF4J API -->

<dependency>

<groupId>org.slf4j</groupId>

<artifactId>slf4j-api</artifactId>

<version>1.7.30</version>

</dependency>

<!-- Logback Classic Implementation -->

<dependency>

<groupId>ch.qos.logback</groupId>

<artifactId>logback-classic</artifactId>

<version>1.2.3</version>

</dependency>

</dependencies>

</project>

LoggingExample.java

package com.example;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

public class LoggingExample {

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

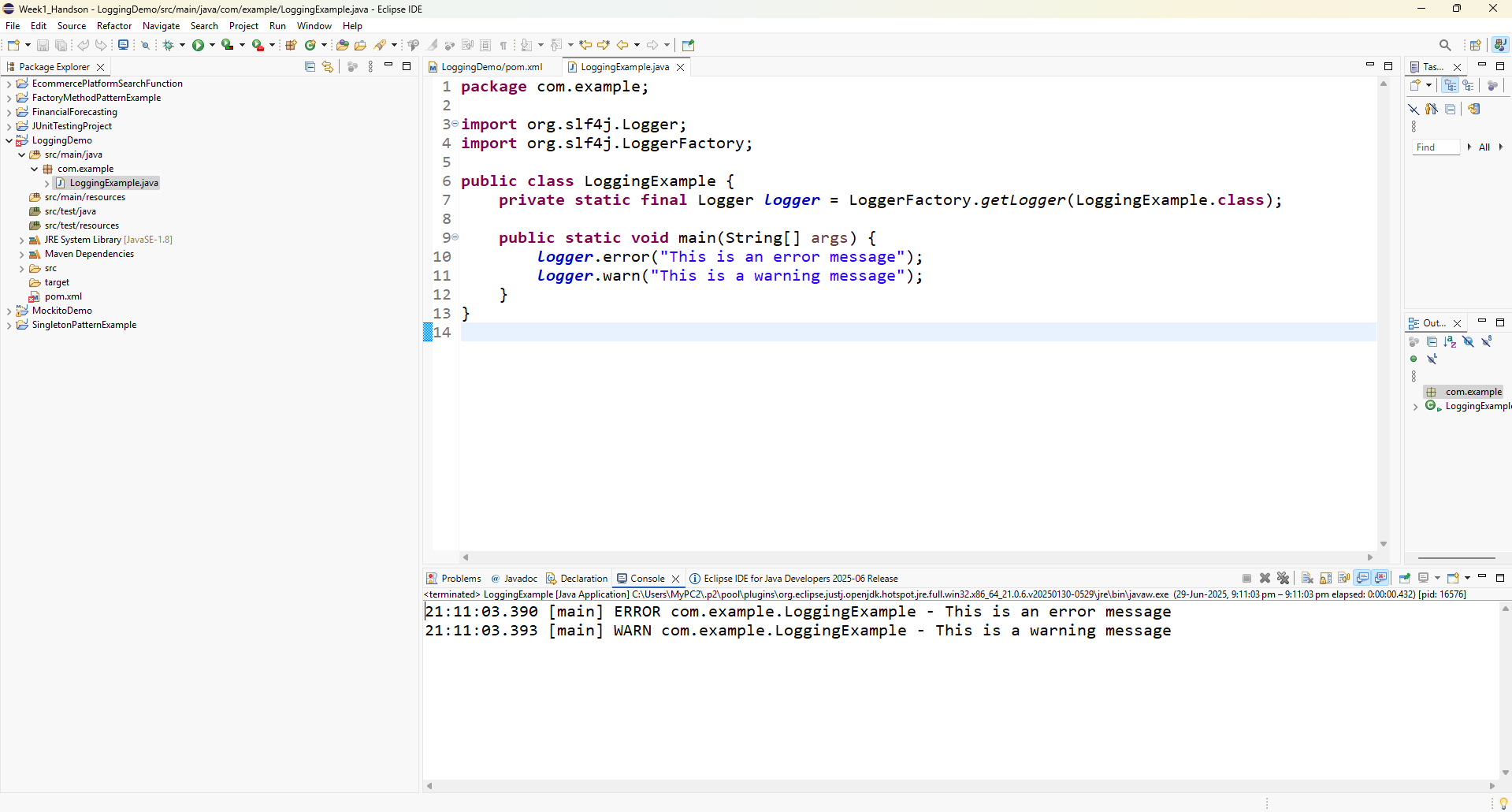
public static void main(String[] args) {

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

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

}

}



**Exercise 2: Parameterized Logging**

ParameterizedLoggingExample.java

package com.example;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

public class ParameterizedLoggingExample {

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

public static void main(String[] args) {

String username = "Raima";

int loginAttempts = 3;

boolean isLocked = true;

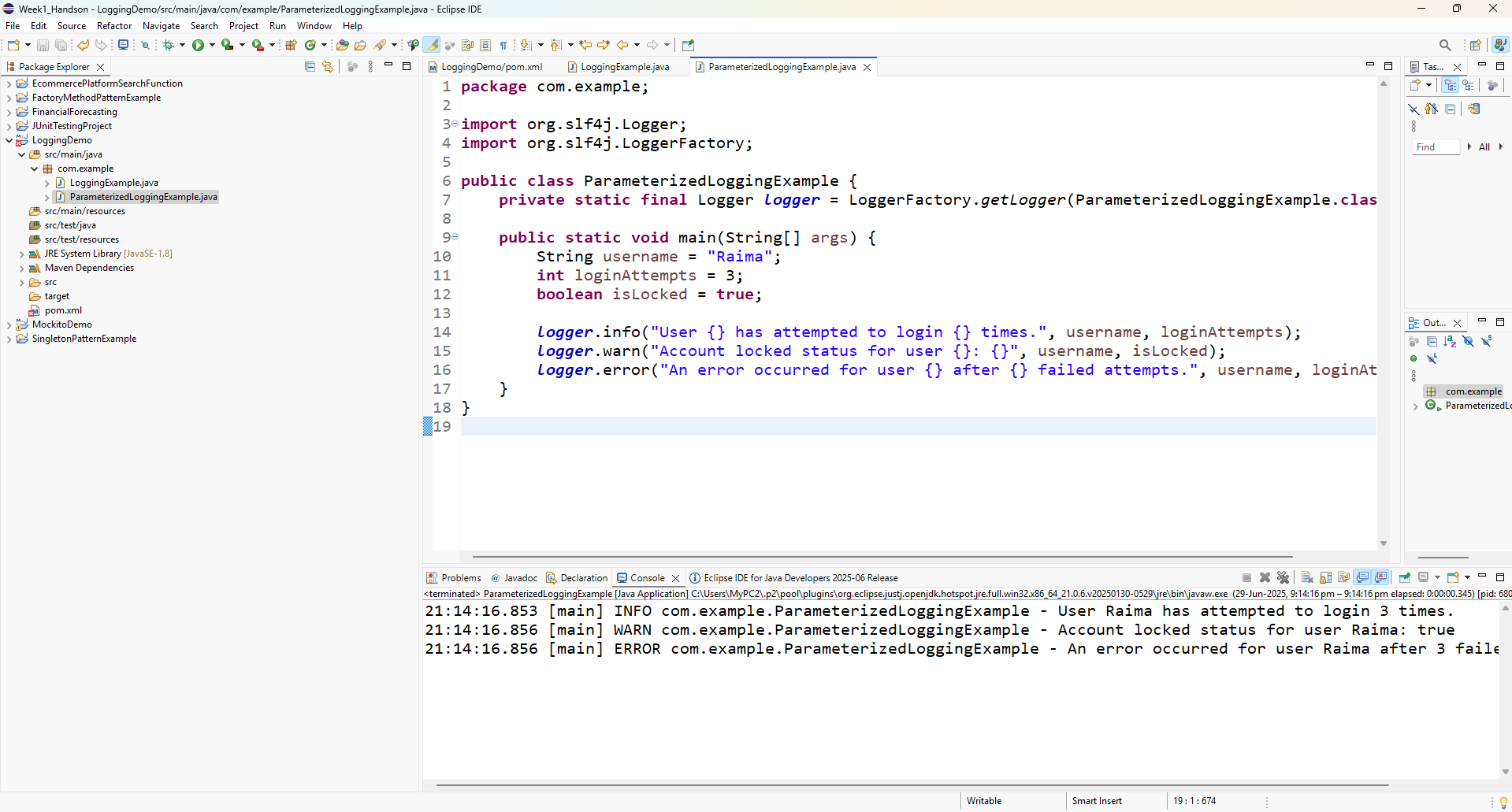
logger.info("User {} has attempted to login {} times.", username, loginAttempts);

logger.warn("Account locked status for user {}: {}", username, isLocked);

logger.error("An error occurred for user {} after {} failed attempts.", username, loginAttempts);

}

}



**Exercise 3: Using Different Appenders**

logback.xml

<configuration>

<appender name="console" class="ch.qos.logback.core.ConsoleAppender">

<encoder>

<pattern>%d{HH:mm:ss.SSS} [%thread] %-5level %logger{36} - %msg%n</pattern>

</encoder>

</appender>

<appender name="file" class="ch.qos.logback.core.FileAppender">

<file>app.log</file>

<encoder>

<pattern>%d{yyyy-MM-dd HH:mm:ss} [%thread] %-5level %logger{36} - %msg%n</pattern>

</encoder>

</appender>

<root level="debug">

<appender-ref ref="console" />

<appender-ref ref="file" />

</root>

</configuration>

MultipleAppendersExample.java

package com.example;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

public class MultipleAppendersExample {

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

public static void main(String[] args) {

logger.info("This message goes to both console and file.");

logger.debug("This is a debug message.");

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

logger.error("This is an error log entry.");

}

}

