**“Logging using SLF4J”**

**Exercise 1: Logging Error Messages and Warning Level , Write a Java application that demonstrates logging error messages and warning levels using SLF4J.**

**SOLUTION:**

package com.example;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

public class LoggingDemo {

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

public static void main(String[] args) {

int diskUsage = 93;

if (diskUsage > 90) {

logger.warn("Warning: Disk usage is critically high at {}%", diskUsage);

}

try {

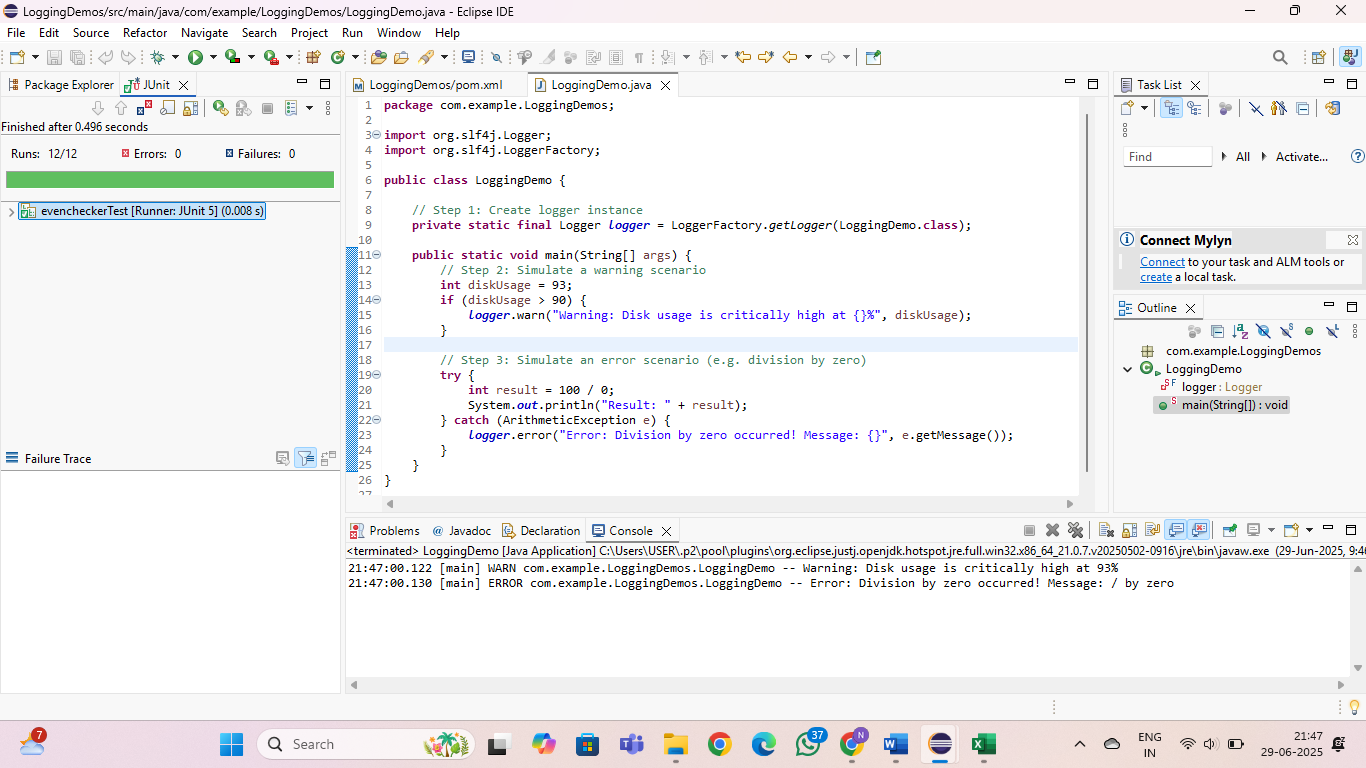
int result = 100 / 0;

System.out.println("Result: " + result);

} catch (ArithmeticException e) {

logger.error("Error: Division by zero occurred! Message: {}", e.getMessage());

}}

}  
****

**Exercise 2: Parameterized Logging Task: Write a Java application that demonstrates parameterized logging using SLF4J.**

**SOLUTION:**

package com.example;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

import java.time.LocalDateTime;

public class LoggingExample {

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

public static void main(String[] args) {

String username = "Nisha";

String operation = "Login";

LocalDateTime timestamp = LocalDateTime.now();

logger.info("User {} performed {} at {}", username, operation, timestamp);

logger.warn("Low disk space on server {}: {}% used", "Server-1", 92.5);

try {

String value = null;

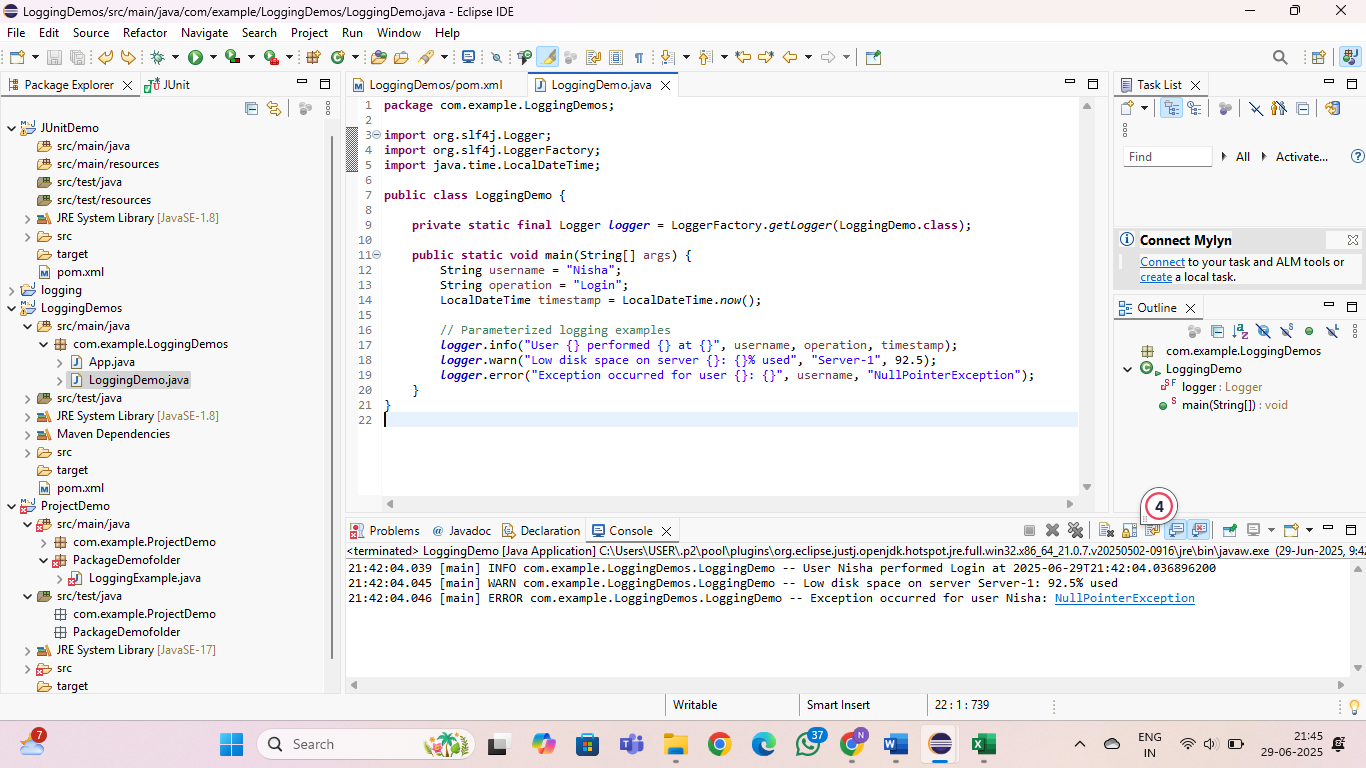
value.length(); // NullPointerException

} catch (Exception e) {

logger.error("Exception for user {}: {}", username, e.toString());

}

}}

****

**Exercise 3: Using Different Appenders Task: Write a Java application that demonstrates using different appenders with SLF4J.**

**SOLUTION:**

package com.example.LoggingDemos;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

public class LoggingDemo {

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

public static void main(String[] args) {

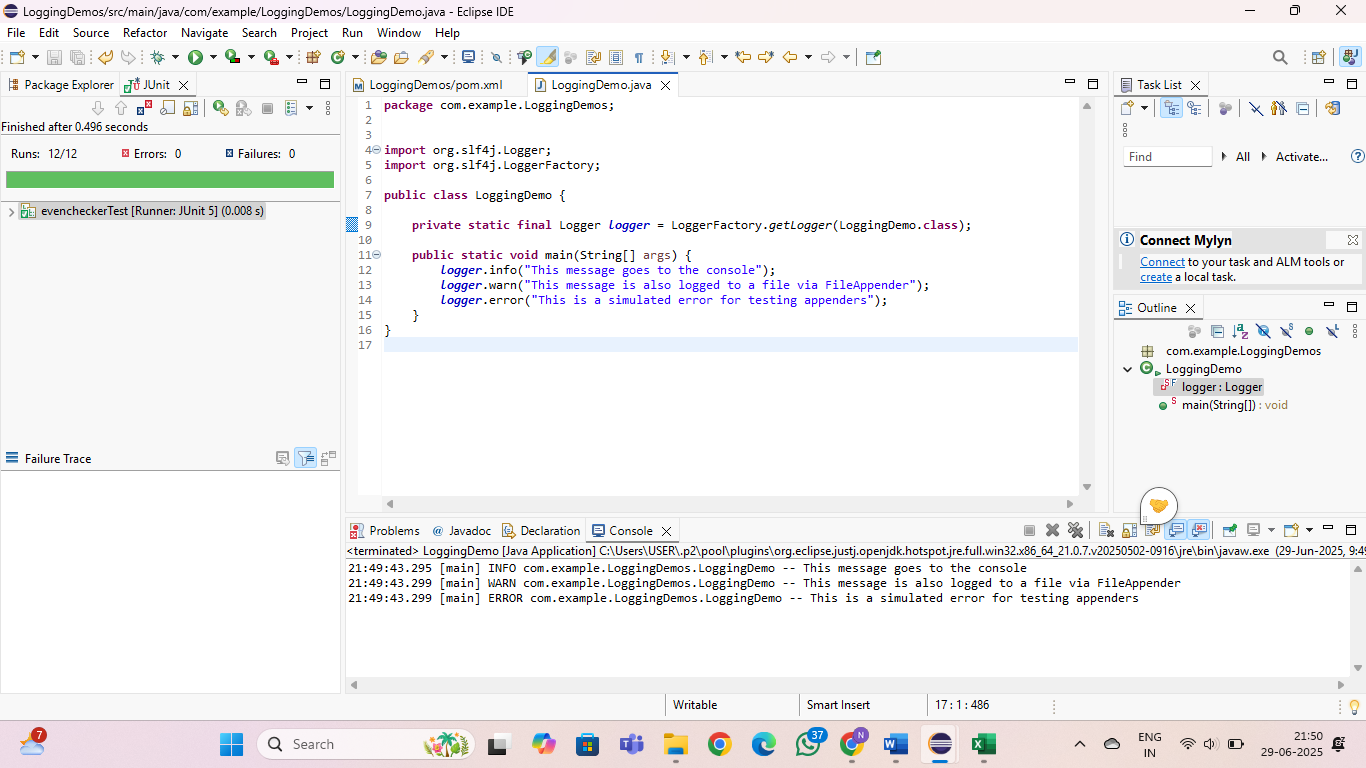
logger.info("This message goes to the console");

logger.warn("This message is also logged to a file via FileAppender");

logger.error("This is a simulated error for testing appenders");

}

}

****

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