**Exercise 1: Implementing the Singleton Pattern**

**Logger.java**

package javapro;

public class Logger {

private static Logger *instance*;

private Logger() {

System.***out***.println("Logger instance created.");

}

public static Logger getInstance() {

if (*instance* == null) {

*instance* = new Logger();

}

return *instance*;

}

public void log(String message) {

System.***out***.println("Log: " + message);

}

}

**Main.java**

package javapro;

public class Main {

public static void main(String[] args) {

Logger logger1 = Logger.*getInstance*();

Logger logger2 = Logger.*getInstance*();

logger1.log("This is the first log message.");

logger2.log("This is the second log message.");

if (logger1 == logger2) {

System.***out***.println("Both logger instances are the same (Singleton confirmed).");

} else {

System.***out***.println("Logger instances are different (Singleton failed).");

}

}

}

**OUTPUT:**

Logger instance created.

Log: This is the first log message.

Log: This is the second log message.

Both logger instances are the same (Singleton confirmed).