**Singleton Pattern Implementation**

**Implementation**

**Logger.java**

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**

public class Main {

    public static void main(String[] args) {

        Logger logger1=Logger.getInstance();

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

        Logger logger2=Logger.getInstance();

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

        if(logger1==logger2) {

            System.out.println("logger1 and logger2 are the same instance.");

        }else {

           System.out.println("Singleton failed. Different instances exist.");

        }

    }

}

**Output**

Logger instance created.

[LOG]: This is the first log message.

[LOG]: This is the second log message.

logger1 and logger2 are the same instance.

