

Exercise 1: Implementing the Singleton Pattern

```
public class main{  
    Run | Debug  
    public static void main(String[] args) {  
        logger logger1 = logger.getInstance();  
        logger logger2 = logger.getInstance();  
  
        logger1.log(message:"This is the first log message.");  
        logger2.log(message:"This is the second log message.");  
  
        if (logger1 == logger2) {  
            System.out.println(x:"Both logger1 and logger2 refer to the same instance.");  
        } else {  
            System.out.println(x:"Different instances exist!");  
        }  
    }  
}
```

```
public class logger {  
    private static logger singleInstance;  
  
    private logger() {  
        System.out.println(x:"Logger initialized");  
    }  
  
    public static logger getInstance() {  
        if (singleInstance == null) {  
            singleInstance = new logger();  
        }  
        return singleInstance;  
    }  
  
    public void log(String message) {  
        System.out.println("LOG: " + message);  
    }  
}
```

Output:

```
Logger initialized  
LOG: This is the first log message.  
LOG: This is the second log message.  
Both logger1 and logger2 refer to the same instance.
```