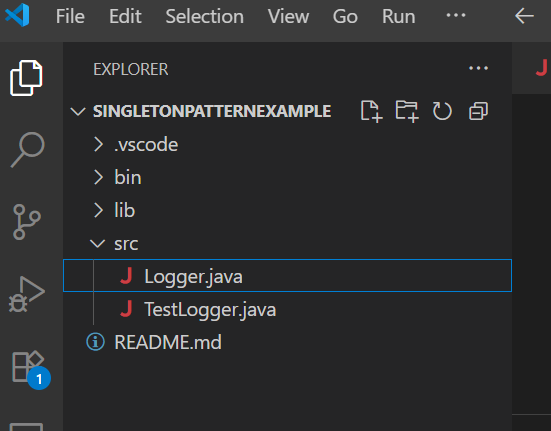
**Implementing the Singleton Pattern**

## **Singleton Pattern**

**Singleton** means:

* Only one object is created.
* Everyone uses the same object.

**CODE:**



Logger.java

## public class Logger {

## private static Logger instance = new Logger();

## 

## private Logger() {

## System.out.println("Logger initialized...");

## }

## public static Logger getInstance() {

## return instance;

## }

## public void log(String message) {

## System.out.println("[LOG] " + message);

## }

## }

## 

TestLogger.java:

public class TestLogger {

public static void main(String args[]) {

Logger log1 = Logger.getInstance();

Logger log2 = Logger.getInstance();

log1.log("This is the first log message.");

log2.log("This is the second log message.");

if (log1 == log2) {

System.out.println("Both Logger instance are same. Singleton confirmed.");

} else {

System.out.println("Logger instance are Different. Singleton pattern failed.");

}

}

}

## **Output :**

