**EXERCISE 1:**

SingletonPatternExample

class Logger {  
 private static Logger *instance*;  
 private Logger() {  
 // Private constructor to prevent instantiation  
 }  
 public static Logger getInstance() {  
 if (*instance* == null) {  
 *instance* = new Logger();  
 }  
 return *instance*;  
 }  
 public void log(String message) {  
 System.*out*.println("Log: " + message);  
 }  
}  
public class SingletonPatternExample {  
 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.");  
  
 // Check if both instances are the same  
 if (logger1 == logger2) {  
 System.*out*.println("Both logger instances are the same.");  
 } else {  
 System.*out*.println("Logger instances are different.");  
 }  
 }  
}