**Logger.java**

public class Logger {

    // Private static instance of the class

    private static Logger instance;

    // Private constructor to prevent instantiation from other classes

    private Logger() {

        // Initialize resources here if needed

    }

    // Public static method to get the single instance of the class

    public static Logger getInstance() {

        if (instance == null) {

            synchronized (Logger.class) {

                if (instance == null) {

                    instance = new Logger();

                }

            }

        }

        return instance;

    }

    // Example method for logging

    public void log(String message) {

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

    }

}

**SingletonTest.java**

public class SingletonTest {

    public static void main(String[] args) {

        // Get the single instance of Logger

        Logger logger1 = Logger.getInstance();

        Logger logger2 = Logger.getInstance();

        System.out.println();

        // Check if both references point to the same instance

        if (logger1 == logger2) {

            System.out.println("Logger instances are the same.");

        } else {

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

        }

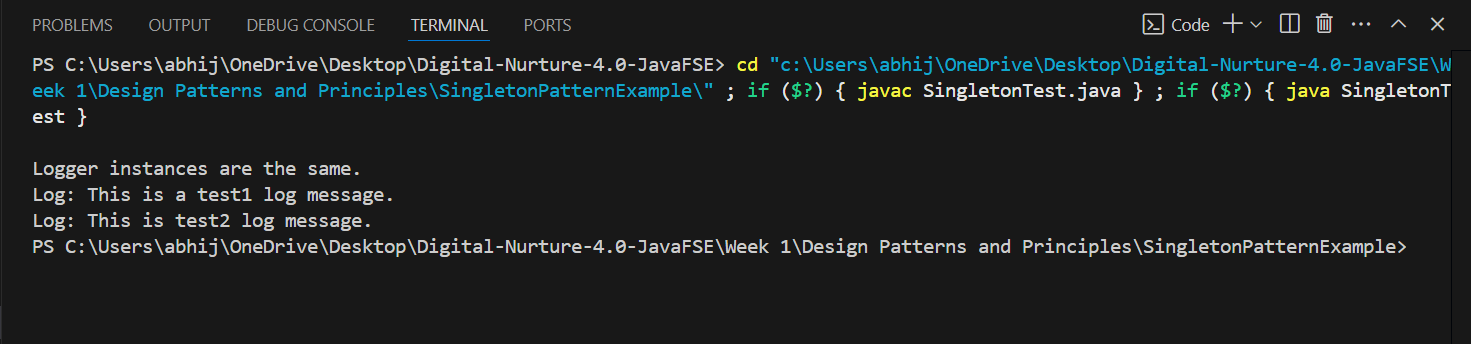
        // Test logging

        logger1.log("This is a test log message.");

        logger2.log("This is another test log message.");

    }

}

**Output:**