**Week 1 Design Patterns And Principles**

**Exercise 1: Implementing the Singleton Pattern**

using System;

namespace SingletonPatternExample

{

    public class Logger

    {

        private static Logger? instance;

        // Private constructor

        private Logger()

        {

            Console.WriteLine("Logger instance created.");

        }

        // Public method to get the singleton instance

        public static Logger GetInstance()

        {

            if (instance == null)

            {

                instance = new Logger();

            }

            return instance;

        }

        public void Log(string message)

        {

            Console.WriteLine("Log: " + message);

        }

    }

    class Program

    {

        static void Main(string[] args)

        {

            Logger logger1 = Logger.GetInstance();

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

            Logger logger2 = Logger.GetInstance();

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

            if (logger1 == logger2)

            {

                Console.WriteLine("Both logger instances are the same (singleton verified).");

            }

            else

            {

                Console.WriteLine("Different instances (singleton failed).");

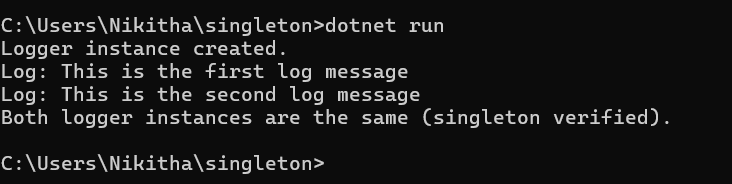
            }

        }

    }

}

**Output Screenshot:**

****