**COGNIZANT Deep\_Skilling**

**Design Pattern and Principles**

Question 1: Implementing the Singleton Pattern

Code:

using System;

namespace SingletonPatternExample

{

    public class Logger

    {

        private static readonly Logger \_instance = new Logger();

        private Logger()

        {

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

        }

        public static Logger GetInstance()

        {

            return \_instance;

        }

        public void Log(string message)

        {

            Console.WriteLine($"[LOG]: {message}");

        }

    }

    class Program

    {

        static void Main(string[] args)

        {

            Console.WriteLine("Testing Singleton Logger...\n");

            Logger logger1 = Logger.GetInstance();

            Logger logger2 = Logger.GetInstance();

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

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

            Console.WriteLine($"\nlogger1 and logger2 are the same instance: {object.ReferenceEquals(logger1, logger2)}");

        }

    }

}

**Output:**

