**Week-1: Design Principles And Patterns**

**Exercise 1: Implementing the Singleton Pattern**

**Program.cs**

**using System;**

**class Program**

**{**

**static void Main()**

**{**

**Logger logger1 = Logger.GetInstance();**

**logger1.Log("First log message");**

**Logger logger2 = Logger.GetInstance();**

**logger2.Log("Second log message");**

**Console.WriteLine(object.ReferenceEquals(logger1, logger2)**

**? "Both logger instances are the same (Singleton working)"**

**: "Different instances (Singleton failed)");**

**}**

**}**

**Logger.cs**

**using System;**

**public class Logger**

**{**

**private static Logger \_instance;**

**private static readonly object \_lock = new object();**

**private Logger()**

**{**

**Console.WriteLine("Logger Initialized");**

**}**

**public static Logger GetInstance()**

**{**

**if (\_instance == null)**

**{**

**lock (\_lock)**

**{**

**if (\_instance == null)**

**{**

**\_instance = new Logger();**

**}**

**}**

**}**

**return \_instance;**

**}**

**public void Log(string message)**

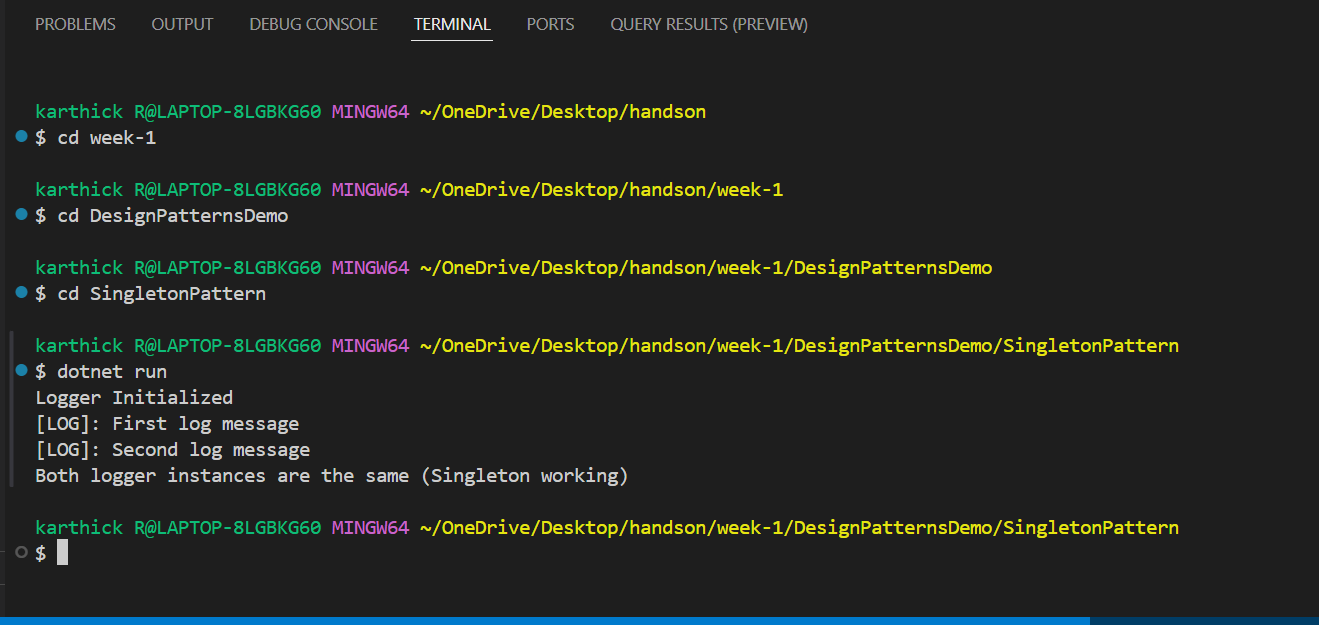
**{**

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

**}**

**}**

**OUTPUT:**

****

**Exercise 2: Implementing the Factory Method Pattern**

**Program.cs:**

**using System;**

**class Program**

**{**

**static void Main()**

**{**

**DocumentFactory wordFactory = new WordFactory();**

**IDocument wordDoc = wordFactory.CreateDocument();**

**wordDoc.Open();**

**DocumentFactory pdfFactory = new PdfFactory();**

**IDocument pdfDoc = pdfFactory.CreateDocument();**

**pdfDoc.Open();**

**DocumentFactory excelFactory = new ExcelFactory();**

**IDocument excelDoc = excelFactory.CreateDocument();**

**excelDoc.Open();**

**}**

**}**

**IDocument.cs**

**public interface IDocument**

**{**

**void Open();**

**}**

**WordDocument.cs**

**using System;**

**public class WordDocument : IDocument**

**{**

**public void Open()**

**{**

**Console.WriteLine("Opening Word Document");**

**}**

**}**

**PdfDocument.cs**

**using System;**

**public class PdfDocument : IDocument**

**{**

**public void Open()**

**{**

**Console.WriteLine("Opening PDF Document");**

**}**

**}**

**ExcelDocument.cs**

**using System;**

**public class ExcelDocument : IDocument**

**{**

**public void Open()**

**{**

**Console.WriteLine("Opening Excel Document");**

**}**

**}**

**DocumentFactory.cs**

**public abstract class DocumentFactory**

**{**

**public abstract IDocument CreateDocument();**

**}**

**WordFactory.cs**

**public class WordFactory : DocumentFactory**

**{**

**public override IDocument CreateDocument()**

**{**

**return new WordDocument();**

**}**

**}**

**PdfFactory.cs**

**public class PdfFactory : DocumentFactory**

**{**

**public override IDocument CreateDocument()**

**{**

**return new PdfDocument();**

**}**

**}**

**ExcelFactory.cs**

**public class ExcelFactory : DocumentFactory**

**{**

**public override IDocument CreateDocument()**

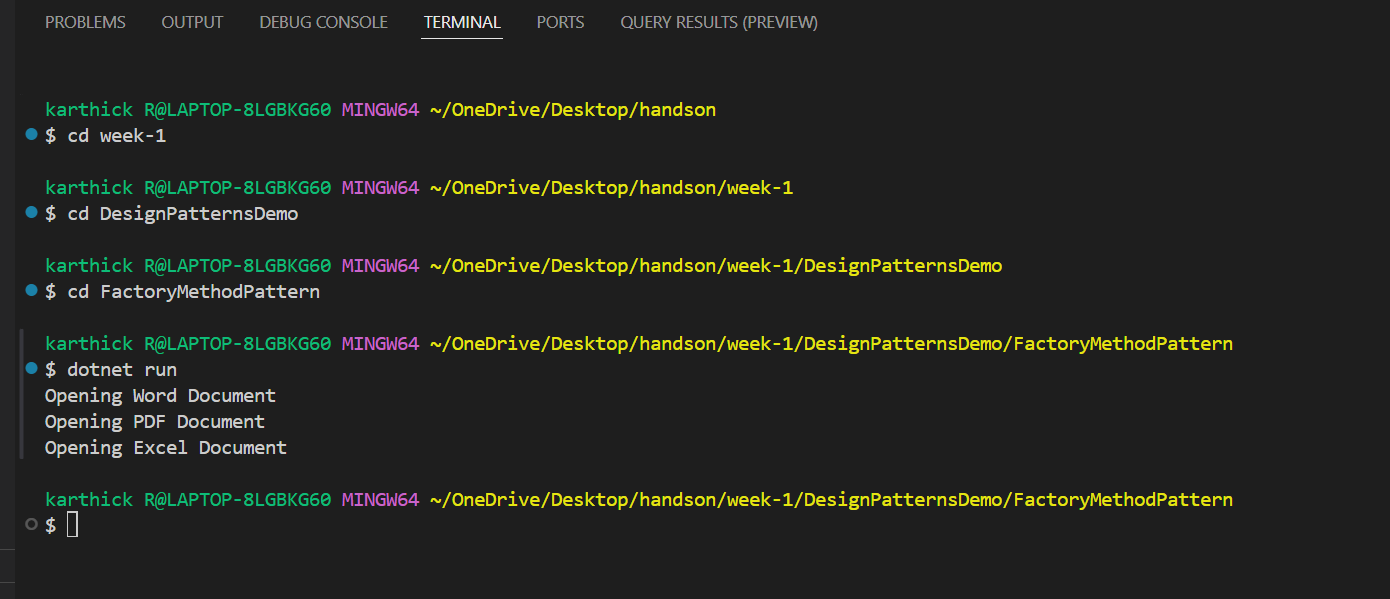
**{**

**return new ExcelDocument();**

**}**

**}**

**OUTPUT:**

****