Exercise 1: Implementing the Singleton Pattern

Code:

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace SingletonPatternExample

{

public sealed class Logger

{

private static Logger \_instance;

private Logger() { }

public static Logger Instance

{

get

{

if (\_instance == null)

{

\_instance = new Logger();

}

return \_instance;

}

}

private void Log(string message)

{

Console.WriteLine($"{DateTime.Now}:{message}");

}

}

public class Program

{

public static void Main(string[] args)

{

Logger logger1 = Logger.Instance;

Logger logger2 = Logger.Instance;

if (ReferenceEquals(logger1, logger2))

{

Console.WriteLine("logger1 and logger2 are the same instance");

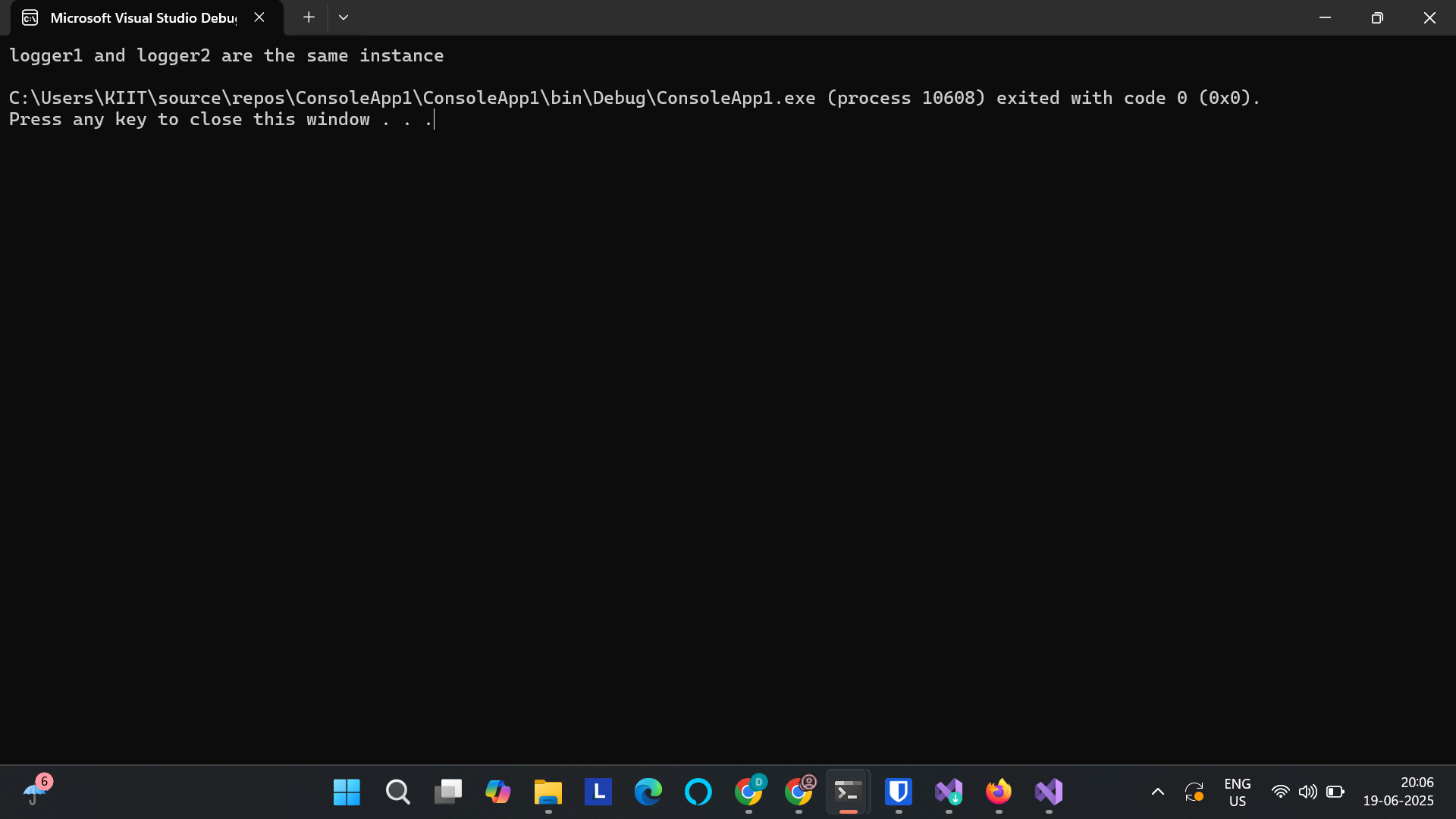
}

}

}

}

Output:



Exercise 2: Implementing the Factory Method Pattern

Code:

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System;

namespace FactoryMethodPatternExample

{

// Product interface

public interface IDocument

{

void DisplayInfo();

}

// Concrete products

public class WordDocument : IDocument

{

public void DisplayInfo()

{

Console.WriteLine("This is a Word document.");

}

}

public class PdfDocument : IDocument

{

public void DisplayInfo()

{

Console.WriteLine("This is a PDF document.");

}

}

public class ExcelDocument : IDocument

{

public void DisplayInfo()

{

Console.WriteLine("This is an Excel document.");

}

}

// Abstract Factory

public abstract class DocumentFactory

{

public abstract IDocument CreateDocument();

}

// Concrete Factories

public class WordDocumentFactory : DocumentFactory

{

public override IDocument CreateDocument()

{

return new WordDocument();

}

}

public class PdfDocumentFactory : DocumentFactory

{

public override IDocument CreateDocument()

{

return new PdfDocument();

}

}

public class ExcelDocumentFactory : DocumentFactory

{

public override IDocument CreateDocument()

{

return new ExcelDocument();

}

}

// Client code

class Program

{

static void Main(string[] args)

{

DocumentFactory wordFactory = new WordDocumentFactory();

DocumentFactory pdfFactory = new PdfDocumentFactory();

DocumentFactory excelFactory = new ExcelDocumentFactory();

IDocument wordDoc = wordFactory.CreateDocument();

IDocument pdfDoc = pdfFactory.CreateDocument();

IDocument excelDoc = excelFactory.CreateDocument();

wordDoc.DisplayInfo(); // Output: This is a Word document.

pdfDoc.DisplayInfo(); // Output: This is a PDF document.

excelDoc.DisplayInfo(); // Output: This is an Excel document.

}

}

}

Output:

