**Factory Method Pattern**

**Project Structure**

Week1-Mandatory-Hands-On/

└── FactoryMethodPattern/

├── Documents/

│ ├── Document.cs // Interface

│ ├── WordDocument.cs // Concrete Word class

│ ├── PdfDocument.cs // Concrete PDF class

│ └── ExcelDocument.cs // Concrete Excel class

├── Factories/

│ ├── DocumentFactory.cs // Abstract factory

│ ├── WordDocumentFactory.cs // Concrete Word factory

│ ├── PdfDocumentFactory.cs // Concrete PDF factory

│ └── ExcelDocumentFactory.cs // Concrete Excel factory

├── FactoryMethodPattern.csproj

└── Program.cs // Test class

**1. Interface: IDocument.cs**

namespace FactoryMethodPattern.Documents

{

    public interface Document

    {

        void Open();

    }

}

**2. Concrete Document Classes**

**WordDocument.cs**

using System;

namespace FactoryMethodPattern.Documents

{

    public class WordDocument : Document

    {

        public void Open()

        {

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

        }

    }

}

**PdfDocument.cs**

using System;

namespace FactoryMethodPattern.Documents

{

    public class PdfDocument : Document

    {

        public void Open()

        {

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

        }

    }

}

**ExcelDocument.cs**

using System;

namespace FactoryMethodPattern.Documents

{

    public class ExcelDocument : Document

    {

        public void Open()

        {

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

        }

    }

}

**3. Abstract Factory: DocumentFactory.cs**

using FactoryMethodPattern.Documents;

namespace FactoryMethodPattern.Factories

{

    public abstract class DocumentFactory

    {

        public abstract Document CreateDocument();

    }

}

**4. Concrete Factories**

**WordDocumentFactory.cs**

using FactoryMethodPattern.Documents;

namespace FactoryMethodPattern.Factories

{

    public class WordDocumentFactory : DocumentFactory

    {

        public override Document CreateDocument()

        {

            return new WordDocument();

        }

    }

}

**PdfDocumentFactory.cs**

using FactoryMethodPattern.Documents;

namespace FactoryMethodPattern.Factories

{

    public class PdfDocumentFactory : DocumentFactory

    {

        public override Document CreateDocument()

        {

            return new PdfDocument();

        }

    }

}

**ExcelDocumentFactory.cs**

using FactoryMethodPattern.Documents;

namespace FactoryMethodPattern.Factories

{

    public class ExcelDocumentFactory : DocumentFactory

    {

        public override Document CreateDocument()

        {

            return new ExcelDocument();

        }

    }

}

**5. Main Program: Program.cs**

using System;

using FactoryMethodPattern.Documents;

using FactoryMethodPattern.Factories;

namespace FactoryMethodPattern

{

    class Program

    {

        static void Main(string[] args)

        {

            DocumentFactory wordFactory = new WordDocumentFactory();

            Document wordDoc = wordFactory.CreateDocument();

            wordDoc.Open();

            DocumentFactory pdfFactory = new PdfDocumentFactory();

            Document pdfDoc = pdfFactory.CreateDocument();

            pdfDoc.Open();

            DocumentFactory excelFactory = new ExcelDocumentFactory();

            Document excelDoc = excelFactory.CreateDocument();

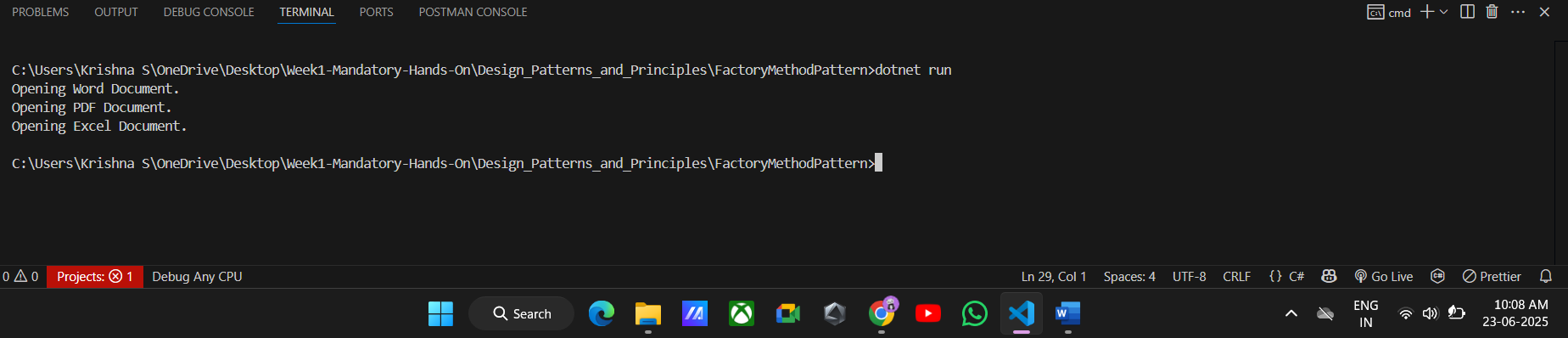
            excelDoc.Open();

        }

    }

}

**7. Output**

****