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();

    }

}

Program.cs

using System;

class Program

{

    static void Main(string[] args)

    {

        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();

    }

}

Output:

