

Factory Method Pattern in C#

Project Structure

```
FactoryMethodPatternExample/  
├── IDocument.cs  
├── WordDocument.cs  
├── PdfDocument.cs  
├── ExcelDocument.cs  
├── DocumentFactory.cs  
├── WordDocumentFactory.cs  
├── PdfDocumentFactory.cs  
├── ExcelDocumentFactory.cs  
├── DocumentFactorySelector.cs  
└── Program.cs
```

1. Interface: IDocument.cs

```
public interface IDocument  
{  
    void Open();  
}
```

2. Concrete Document Classes

WordDocument.cs

```
public class WordDocument : IDocument  
{  
    public void Open()  
    {  
        Console.WriteLine("Opening Word Document");  
    }  
}
```

PdfDocument.cs

```
public class PdfDocument : IDocument  
{  
    public void Open()  
    {  
        Console.WriteLine("Opening PDF Document");  
    }  
}
```

```
}  
}
```

ExcelDocument.cs

```
public class ExcelDocument : IDocument  
{  
    public void Open()  
    {  
        Console.WriteLine("Opening Excel Document");  
    }  
}
```

3. Abstract Factory: DocumentFactory.cs

```
public abstract class DocumentFactory  
{  
    public abstract IDocument CreateDocument();  
}
```

4. Concrete Factories

WordDocumentFactory.cs

```
public class WordDocumentFactory : DocumentFactory  
{  
    public override IDocument CreateDocument()  
    {  
        return new WordDocument();  
    }  
}
```

PdfDocumentFactory.cs

```
public class PdfDocumentFactory : DocumentFactory  
{  
    public override IDocument CreateDocument()  
    {  
        return new PdfDocument();  
    }  
}
```

ExcelDocumentFactory.cs

```
public class ExcelDocumentFactory : DocumentFactory
{
    public override IDocument CreateDocument()
    {
        return new ExcelDocument();
    }
}
```

5. Factory Selector: DocumentFactorySelector.cs

```
public static class DocumentFactorySelector
{
    public static DocumentFactory GetFactory(string type)
    {
        switch (type.ToUpper())
        {
            case "WORD":
                return new WordDocumentFactory();
            case "PDF":
                return new PdfDocumentFactory();
            case "EXCEL":
                return new ExcelDocumentFactory();
            default:
                throw new ArgumentException("Invalid document type: " + type);
        }
    }
}
```

6. Main Program: Program.cs

```
using System;

class Program
{
    static void Main(string[] args)
    {
        Console.Write("Enter document type (WORD, PDF, EXCEL): ");
        string? type = Console.ReadLine()?.Trim();

        try
        {
            DocumentFactory factory = DocumentFactorySelector.GetFactory(type
?? "");
```

```

        IDocument doc = factory.CreateDocument();
        doc.Open();
    }
    catch (Exception ex)
    {
        Console.WriteLine(ex.Message);
    }
}
}

```

7. Output

```

FactoryMethodPatternExample > Program.cs > ...
1 using System;
2
3 references
4 class Program
5 {
6     references
7     static void Main(string[] args)
8     {
9         Console.WriteLine("Enter document type (WORD, PDF, EXCEL): ");
10        string? type = Console.ReadLine()?.Trim();
11
12        try
13        {
14            DocumentFactory factory = DocumentFactorySelector.GetFactory(type ?? "");
15            IDocument doc = factory.CreateDocument();
16            doc.Open();
17        }
18        catch (Exception ex)
19        {
20            Console.WriteLine(ex.Message);
21        }
22    }
23 }

```

PS D:\Cognizant\Week 1\Hand_On> cd FactoryMethodPatternExample
 PS D:\Cognizant\Week 1\Hand_On\FactoryMethodPatternExample> dotnet run
 >>
 Enter document type (WORD, PDF, EXCEL): WORD
 Opening Word Document
 PS D:\Cognizant\Week 1\Hand_On\FactoryMethodPatternExample> dotnet run
 >>
 Enter document type (WORD, PDF, EXCEL): PDF
 Opening PDF Document
 PS D:\Cognizant\Week 1\Hand_On\FactoryMethodPatternExample> dotnet run
 >>
 Enter document type (WORD, PDF, EXCEL): EXCEL
 Opening Excel Document
 PS D:\Cognizant\Week 1\Hand_On\FactoryMethodPatternExample> dotnet run
 >>
 Enter document type (WORD, PDF, EXCEL): PPF
 Invalid document type: PPF
 PS D:\Cognizant\Week 1\Hand_On\FactoryMethodPatternExample>