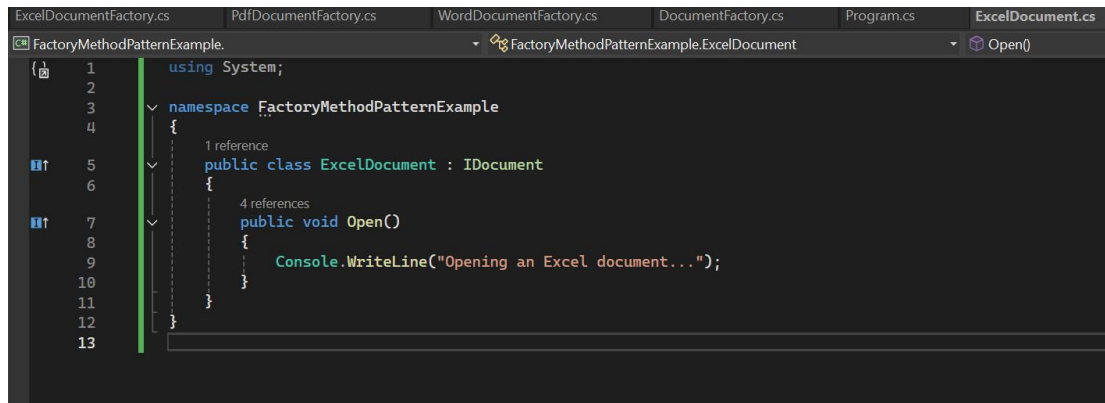


## Exercise 2: Implementing the Factory Method Pattern

### Scenario:

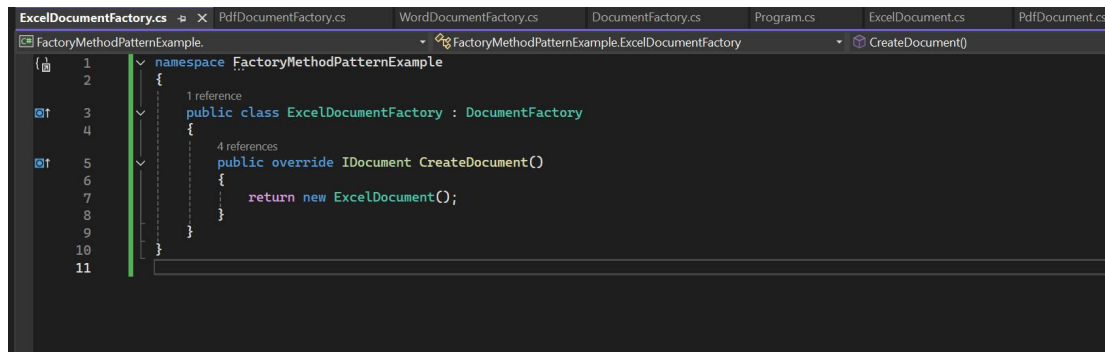
You are developing a document management system that needs to create different types of documents (e.g., Word, PDF, Excel). Use the Factory Method Pattern to achieve this.

### EXCELDOCUMENT.CS



```
ExcelDocumentFactory.cs PdfDocumentFactory.cs WordDocumentFactory.cs DocumentFactory.cs Program.cs ExcelDocument.cs
FactoryMethodPatternExample. FactoryMethodPatternExample.ExcelDocument Open()
1 using System;
2
3 namespace FactoryMethodPatternExample
4 {
5     1 reference
6     public class ExcelDocument : IDocument
7     {
8         4 references
9         public void Open()
10        {
11            Console.WriteLine("Opening an Excel document...");
12        }
13    }
```

### IDocument.cs



```
ExcelDocumentFactory.cs PdfDocumentFactory.cs WordDocumentFactory.cs DocumentFactory.cs Program.cs ExcelDocument.cs PdfDocument.cs
FactoryMethodPatternExample. FactoryMethodPatternExample.ExcelDocumentFactory CreateDocument()
1 namespace FactoryMethodPatternExample
2 {
3     1 reference
4     public class ExcelDocumentFactory : DocumentFactory
5     {
6         4 references
7         public override IDocument CreateDocument()
8         {
9             return new ExcelDocument();
10        }
11    }
```

```
ternExample. FactoryMethodPatternExample.IDocument Open()
{
    // Base interface for all document types
    10 references
    public interface IDocument
    {
        6 references
        void Open();
    }
}
```

## PdfDocument.cs

```
ExcelDocumentFactory.cs WordDocumentFactory.cs DocumentFactory.cs Program.cs ExcelDocument.cs PdfDocument.cs IDocument.cs
FactoryMethodPatternExample. FactoryMethodPatternExample.PdfDocument Open()
{
    1 using System;
    2
    3     namespace FactoryMethodPatternExample
    4     {
    5         1 reference
    6         public class PdfDocument : IDocument
    7         {
    8             4 references
    9             public void Open()
10             {
11                 Console.WriteLine("Opening a PDF document...");
12             }
13         }
    }
```

## Documentaryfactor.cs

```
ExcelDocumentFactory.cs PdfDocumentFactory.cs WordDocumentFactory.cs DocumentFactory.cs
FactoryMethodPatternExample. FactoryMethodPatternExample.DocumentFactory
{
    1 namespace FactoryMethodPatternExample
    2 {
    3     // Abstract Factory
    4     6 references
    5     public abstract class DocumentFactory
    6     {
    7         6 references
    8         public abstract IDocument CreateDocument();
    9     }
}
```

## Program.cs

```
ExcelDocumentFactory.cs | DocumentFactory.cs | Program.cs* | ExcelDocument.cs | PdfDocument.cs | IDocument.cs | PdfDocumentFactory.cs
FactoryMethodPatternExample | FactoryMethodPatternExample.Program | Main(string[] args)
1 using System;
2
3 namespace FactoryMethodPatternExample
4 {
5     0 references
6     class Program
7     {
8         0 references
9         static void Main(string[] args)
10         {
11             // Word Document
12             DocumentFactory wordFactory = new WordDocumentFactory();
13             IDocument wordDoc = wordFactory.CreateDocument();
14             wordDoc.Open();
15
16             // PDF Document
17             DocumentFactory pdfFactory = new PdfDocumentFactory();
18             IDocument pdfDoc = pdfFactory.CreateDocument();
19             pdfDoc.Open();
20
21             // Excel Document
22             DocumentFactory excelFactory = new ExcelDocumentFactory();
23             IDocument excelDoc = excelFactory.CreateDocument();
24             excelDoc.Open();
25         }
26     }
27 }
```

## WordDocument.cs

```
ExcelDocumentFactory.cs | Program.cs* | ExcelDocument.cs | PdfDocument.cs | IDocument.cs | PdfDocumentFactory.cs | wordDocument.cs
FactoryMethodPatternExample | FactoryMethodPatternExample.WordDocument | Open()
1 using System;
2
3 namespace FactoryMethodPatternExample
4 {
5     1 reference
6     public class WordDocument : IDocument
7     {
8         4 references
9         public void Open()
10         {
11             Console.WriteLine("Opening a Word document...");
12         }
13     }
14 }
```

## WordDocumentFactory

```
ExcelDocumentFactory.cs | Program.cs* | ExcelDocument.cs | PdfDocument.cs | IDocument.cs | PdfDocumentFactory.cs | wordDocument.cs | WordDocumentFactory.cs
FactoryMethodPatternExample | FactoryMethodPatternExample.WordDocumentFactory | CreateDocument()
1 namespace FactoryMethodPatternExample
2 {
3     1 reference
4     public class WordDocumentFactory : DocumentFactory
5     {
6         4 references
7         public override IDocument CreateDocument()
8         {
9             return new WordDocument();
10         }
11     }
12 }
```

## PdfDocumentFactory.cs

```
1 namespace FactoryMethodPatternExample
2 {
3     public class PdfDocumentFactory : DocumentFactory
4     {
5         public override IDocument CreateDocument()
6         {
7             return new PdfDocument();
8         }
9     }
10 }
11
```

Output:

```
C:\Users\KIIT\source\repos\F
Opening a Word document...
Opening a PDF document...
Opening an Excel document...

```