**Exercise 2: Implementing the Factory Method Pattern**

* **Create a New Java Project**

In your VS Code IDE, create a new Java project named *FactoryMethodPatternExample*. This will serve as the base for implementing the Factory Method Pattern.

* **Define Document Classes**

Create an interface named Document which will define the basic operations that any document should have, such as open(), save(), and close().

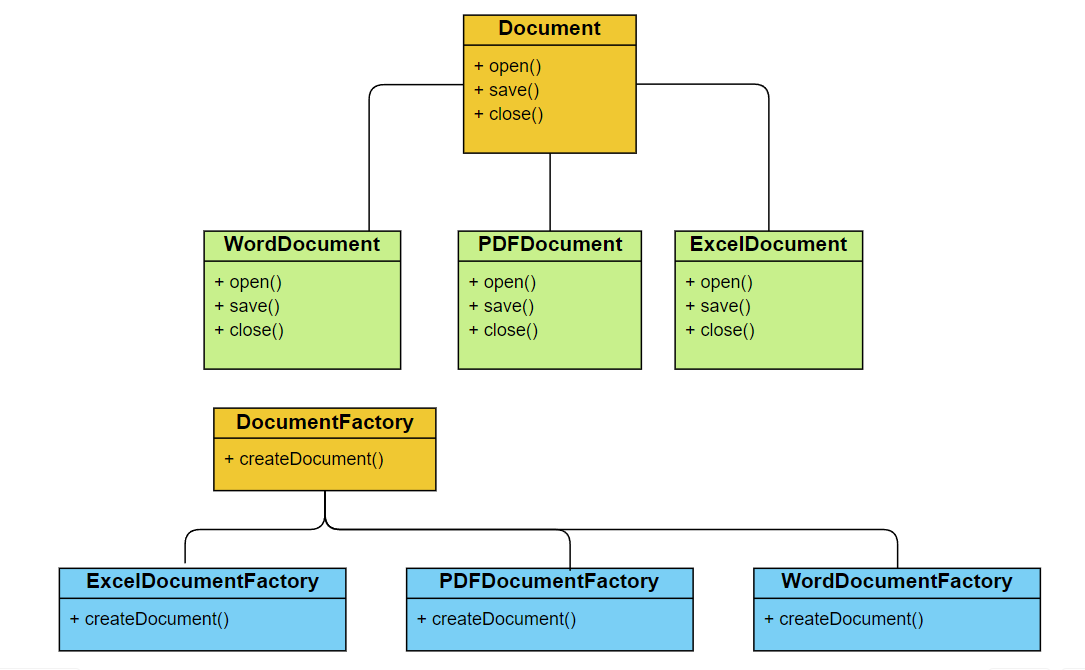
* **Create Concrete Document Classes**

Create concrete classes for each document type (*WordDocument*, *PdfDocument*, and *ExcelDocument*) that implement the Document interface. Each class will provide specific implementations of the open(), save(), and close() methods.

* **Implement the Factory Method**

Create an abstract class named *DocumentFactory* with an abstract method *createDocument().* This method will be implemented by concrete factory classes (*WordDocumentFactory*, *PdfDocumentFactory*, and *ExcelDocumentFactory*), each of which will instantiate and return an instance of the corresponding document type.

Here a class diagram for better understand :



* **Class Descriptions**

1. **Document (Interface)**: Defines the basic operations for any document (open(), save(), close()).
2. **WordDocument (Concrete Class)**: Implements the Document interface and provides specific behaviour for Word documents.
3. **PdfDocument (Concrete Class)**: Implements the Document interface and provides specific behaviour for PDF documents.
4. **ExcelDocument (Concrete Class)**: Implements the Document interface and provides specific behaviour for Excel documents.
5. **DocumentFactory (Abstract Class)**: Declares the factory method createDocument() which will be implemented by concrete factory classes.
6. **WordDocumentFactory (Concrete Class)**: Extends DocumentFactory and implements createDocument() to create and return a WordDocument instance.
7. **PdfDocumentFactory (Concrete Class)**: Extends DocumentFactory and implements createDocument() to create and return a PdfDocument instance.
8. **ExcelDocumentFactory (Concrete Class)**: Extends DocumentFactory and implements createDocument() to create and return an ExcelDocument instance

* **Test the Factory Method Implementation**

Create a test class to demonstrate the creation of different document types using the factory method. The test class will use the different factory classes to create and manipulate instances of the various document types.

The Output of the program

