

Week 1: Design Principles and Patterns:

Exercise 2: Implementing the Factory Method Pattern

Factory Pattern Example: JAVA Project

The Project structured into the following files and folders(packages):

```
FactoryMethodPatternExample/
├── document/
│   ├── Document.java
│   ├── ExcelDocument.java
│   ├── PdfDocument.java
│   └── WordDocument.java
├── factory/
│   ├── DocumentFactory.java
│   ├── WordDocFactory.java
│   ├── ExcelDocFactory.java
│   └── PdfDocFactory.java
└── main/
    └── Main.java
```

- The Code is segregated into two packages – document package and factory package
- The Main.java file consists of main function, containing code to test the data
- Document.java is the single interface used to implement all types of documents (word, excel, ppt in this case) making it elegant to use
- DocumentFactory.java is an abstract class that contains abstract createDocument() method to extend to different document type builders

Why Interface for Document?

To keep the implementations of document functions consistent for different file types (open, print, close) while making it elegant to create references to different types of documents (we use references of “Document” object type for any type keeping it organized).

Why Abstract Class for Factory?

To extend the factory (document building) methods by adding document type specific instructions, while allowing to define default methods common to all documents (in future)

Document Package

Document.java

```
package document;
```

```
public interface Document {
    void open();
    void print();
    void close();
}
```

WordDocument.java

```
package document;

public class WordDocument implements Document{
    public void open() {
        System.out.println("Word Document Opened");
    }
    public void print() {
        System.out.println("Printing Word Document...");
    }
    public void close() {
        System.out.println("Closing Word Document...");
    }
}
```

ExcelDocument.java

```
package document;

public class ExcelDocument implements Document{
    public void open() {
        System.out.println("Excel Document Opened");
    }
    public void print() {
        System.out.println("Printing Excel Document...");
    }
    public void close() {
        System.out.println("Closing Excel Document...");
    }
}
```

PdfDocument.java

```
package document;

public class PdfDocument implements Document{
    public void open() {
        System.out.println("PDF Document Opened");
    }
    public void print() {
        System.out.println("Printing PDF Document...");
    }
    public void close() {
        System.out.println("Closing PDF Document...");
    }
}
```

Factory Package

DocumentFactory.java

```
package factory;
import document.Document;

public abstract class DocumentFactory {
    public abstract Document createDocument();
}
```

WordDocFactory.java

```
package factory;
import document.*;

public class WordDocFactory extends DocumentFactory{
    public Document createDocument() {
        return new WordDocument();
    }
}
```

ExcelDocFactory.java

```
package factory;
import document.*;

public class ExcelDocFactory extends DocumentFactory {
    public Document createDocument() {
        return new ExcelDocument();
    }
}
```

PdfDocFactory.java

```
package factory;
import document.*;

public class PdfDocFactory extends DocumentFactory{
    public Document createDocument() {
        return new PdfDocument();
    }
}
```

Main.java

```
package main;

import document.*;
import factory.*;

public class Main {
    public static void main(String[] args) {
        DocumentFactory wordFactory = new WordDocFactory();
        DocumentFactory pdfFactory = new PdfDocFactory();
        DocumentFactory excelFactory = new ExcelDocFactory();

        Document d1 = wordFactory.createDocument();
        Document d2 = excelFactory.createDocument();
        Document d3 = pdfFactory.createDocument();

        Document [] documents = {d1, d2, d3};

        for(Document doc : documents) {
            doc.open();
            doc.print();
            doc.close();
        }
    }
}
```

Output

```
307fd719\bin main.Main "
Word Document Opened
Printing Word Document...
Closing Word Document...
Excel Document Opened
Printing Excel Document...
Closing Excel Document...
PDF Document Opened
Printing PDF Document...
Closing PDF Document...
```