# **Assignment: Exercise 2 – Factory Method Pattern: Document Management System**

## **2. Setup: Creating a Document Interface**

Document.java

public interface Document {

void open();

}

## **3. Implementation: Concrete Document Classes**

WordDocument.java

public class WordDocument implements Document {

@Override

public void open() {

System.out.println("Opening Word Document...");

}

}

PdfDocument.java

public class PdfDocument implements Document {

@Override

public void open() {

System.out.println("Opening PDF Document...");

}

}

ExcelDocument.java

public class ExcelDocument implements Document {

@Override

public void open() {

System.out.println("Opening Excel Document...");

}

}

## **4. Factory Method: DocumentFactory and Subclasses**

DocFact.java

public abstract class DocFact {

public abstract Document createDocument();

}

WordDocFact.java

public class WordDocFact extends DocFact {

@Override

public Document createDocument() {

return new WordDocument();

}

}

PdfDocFact.java

public class PdfDocFact extends DocFact {

@Override

public Document createDocument() {

return new PdfDocument();

}

}

ExcelDocFact.java

public class ExcelDocFact extends DocFact {

@Override

public Document createDocument() {

return new ExcelDocument();

}

}

## **5. Testing the Factory Pattern**

Main.java

public class Main {

public static void main(String[] args) {

DocumentFactory wordFactory = new WordDocumentFactory();

Document wordDoc = wordFactory.createDocument();

wordDoc.open();

DocumentFactory pdfFactory = new PdfDocumentFactory();

Document pdfDoc = pdfFactory.createDocument();

pdfDoc.open();

DocumentFactory excelFactory = new ExcelDocumentFactory();

Document excelDoc = excelFactory.createDocument();

excelDoc.open();

}

}

## **6. Output**

