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

**OBJECTIVE:**

To create a **Document Management System** that can generate different document types (Word, PDF, Excel) using the **Factory Method Pattern**, ensuring scalability and clean code separation.

**CODE:**

public interface FileType {

void render();

}

public class WordFile implements FileType {

public void render() {

System.out.println("Rendering content in a Word document.");

}

}

public class PdfFile implements FileType {

public void render() {

System.out.println("Rendering content in a PDF file.");

}

}

public class ExcelSheet implements FileType {

public void render() {

System.out.println("Rendering data in an Excel sheet.");

}

}

public abstract class FileGenerator {

public abstract FileType generateFile();

}

public class WordGenerator extends FileGenerator {

public FileType generateFile() {

return new WordFile();

}

}

public class PdfGenerator extends FileGenerator {

public FileType generateFile() {

return new PdfFile();

}

}

public class ExcelGenerator extends FileGenerator {

public FileType generateFile() {

return new ExcelSheet();

}

}

public class Document {

public static void main(String[] args) {

FileGenerator wordGen = new WordGenerator();

FileType word = wordGen.generateFile();

word.render();

FileGenerator pdfGen = new PdfGenerator();

FileType pdf = pdfGen.generateFile();

pdf.render();

FileGenerator excelGen = new ExcelGenerator();

FileType excel = excelGen.generateFile();

excel.render();

}

}

**OUTPUT:**

