Implementing the Factory Method Pattern :-

1. Document.java :-

public interface Document {

void open();

void save();

void close();

}

2. WordDocument.java:-

public class WordDocument implements Document {

@Override

public void open() {

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

}

@Override

public void save() {

System.out.println("Saving Word document...");

}

@Override

public void close() {

System.out.println("Closing Word document...");

}

}

3. ExcelDocument.java :-

public class ExcelDocument implements Document {

@Override

public void open() {

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

}

@Override

public void save() {

System.out.println("Saving Excel document...");

}

@Override

public void close() {

System.out.println("Closing Excel document...");

}

}

4. PdfDocument.java

public class PdfDocument implements Document {

@Override

public void open() {

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

}

@Override

public void save() {

System.out.println("Saving PDF document...");

}

@Override

public void close() {

System.out.println("Closing PDF document...");

}

}

5. DocumentFactory.java :-

public abstract class DocumentFactory {

public abstract Document createDocument();

}

6. WordDocumentFactory.java :-

/public class WordDocumentFactory extends DocumentFactory {

public WordDocumentFactory() {

}

public Document createDocument() {

return new WordDocument();

}

}

7. ExcelDocumentFactory.java :-

public class ExcelDocumentFactory extends DocumentFactory {

@Override

public Document createDocument() {

return new ExcelDocument();

}

}

8. PdfDocumentFactory.java :-

public class PdfDocumentFactory extends DocumentFactory {

public PdfDocumentFactory() {

}

public Document createDocument() {

return new PdfDocument();

}

}

Output: -

