# FACTORY DESIGN PATTERN

Code :

public class DocumentFactoryPattern {

interface Document {

void open();

}

static class WordDocument implements Document {

public void open() {

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

}

}

static class PDFDocument implements Document {

public void open() {

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

}

}

static class ExcelDocument implements Document {

public void open() {

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

}

}

static class DocumentFactory {

public static Document createDocument(String type) {

if (type.equalsIgnoreCase("word")) {

return new WordDocument();

} else if (type.equalsIgnoreCase("pdf")) {

return new PDFDocument();

} else if (type.equalsIgnoreCase("excel")) {

return new ExcelDocument();

} else {

return null;

}

}

}

public static void main(String[] args) {

Document doc1 = DocumentFactory.createDocument("word");

if (doc1 != null) doc1.open();

Document doc2 = DocumentFactory.createDocument("pdf");

if (doc2 != null) doc2.open();

Document doc3 = DocumentFactory.createDocument("excel");

if (doc3 != null) doc3.open();

Document doc4 = DocumentFactory.createDocument("text");

if (doc4 == null) {

System.out.println("Unknown document type");

}

}

}

Output :

