# FactoryMethodPatternExample:

Document.java:

public interface Document {

void open();

}

DocumentFactory.java:

public abstract class DocumentFactory {

public abstract Document createDocument();

}

ExcelDocument.java:

public class ExcelDocument implements Document {

public void open() {

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

}

}

ExcelDocumentFactory.java:

public class ExcelDocumentFactory extends DocumentFactory {

public Document createDocument() {

return new ExcelDocument();

}

}

FactoryPatternTest.java:

public class FactoryPatternTest {

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();

}

}

PdfDocument.java:

public class PdfDocument implements Document {

public void open() {

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

}

}

PdfDocumentFactory.java:

public class PdfDocumentFactory extends DocumentFactory {

public Document createDocument() {

return new PdfDocument();

}

}

WordDocument.java:

public class WordDocument implements Document {

public void open() {

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

}

}

WordDocumentFactory.java:

public class WordDocumentFactory extends DocumentFactory {

public Document createDocument() {

return new WordDocument();

}

}

