Exercise 2: Factory Method Pattern Implementation

public interface Document {

void open();

void save();

void close();

}

public class WordDocument implements Document {

public void open() {

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

}

public void save() {

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

}

public void close() {

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

}

}

public class PdfDocument implements Document {

public void open() {

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

}

public void save() {

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

}

public void close() {

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

}

}

public class ExcelDocument implements Document {

public void open() {

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

}

public void save() {

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

}

public void close() {

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

}

}

public abstract class DocumentFactory {

public abstract Document createDocument();

public void processDocument() {

Document doc = createDocument();

doc.open();

doc.save();

doc.close();

}

}

public class WordDocumentFactory extends DocumentFactory {

public Document createDocument() {

return new WordDocument();

}

}

public class PdfDocumentFactory extends DocumentFactory {

public Document createDocument() {

return new PdfDocument();

}

}

public class ExcelDocumentFactory extends DocumentFactory {

public Document createDocument() {

return new ExcelDocument();

}

}

public class FactoryMethodTest {

public static void main(String[] args) {

DocumentFactory wordFactory = new WordDocumentFactory();

Document wordDoc = wordFactory.createDocument();

wordDoc.open();

DocumentFactory pdfFactory = new PdfDocumentFactory();

pdfFactory.processDocument();

DocumentFactory excelFactory = new ExcelDocumentFactory();

Document excelDoc = excelFactory.createDocument();

excelDoc.save();

}

}