**Design Patterns and Principles – Exercise 2**

**Factory Method Pattern Example**

**Main.java**

public class Main {

    public static void main(String[] args) {

        DocumentFactory word = new WordDocumentFactory();

        Document wordDoc = word.createDocument();

        wordDoc.open();

        DocumentFactory pdf = new PdfDocumentFactory();

        Document pdfDoc = pdf.createDocument();

        pdfDoc.open();

        DocumentFactory excel = new ExcelDocumentFactory();

        Document excelDoc = excel.createDocument();

        excelDoc.open();

    }

}

**Document.java**

public interface Document {

    void open();

}

**PdfDocument.java**

public class PdfDocument implements Document{

    public void open(){

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

    }

}

**WordDocument.java**

public class WordDocument implements Document{

    public void open(){

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

    }

}

**ExcelDocument.java**

public class ExcelDocument implements Document{

    public void open(){

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

    }

}

**DocumentFactory.java**

public abstract class DocumentFactory {

    public abstract Document createDocument();

}

**PdfDocumentFactory.java**

public class PdfDocumentFactory extends DocumentFactory{

    public Document createDocument(){

        return new PdfDocument();

    }

}

**WordDocumentFactory.java**

public class WordDocumentFactory extends DocumentFactory{

    public Document createDocument(){

        return new WordDocument();

    }

}

**ExcelDocumentFactory.java**

public class ExcelDocumentFactory extends DocumentFactory{

    public Document createDocument(){

        return new ExcelDocument();

    }

}

**Output**

