Design Principles and Patterns:

Question-2: (FactoryPatternExample)

public class Q2 {

    public interface Document {

    void open();

    }

    public static class WordDocument implements Document {

        @Override

        public void open() {

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

        }

    }

    public static class PDFDocument implements Document {

        @Override

        public void open() {

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

        }

    }

    public static class ExcelDocument implements Document {

        @Override

        public void open() {

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

        }

    }

    public abstract class DocumentFactory {

        public abstract Document createDocument();

    }

    public class WordDocumentFactory extends DocumentFactory {

        @Override

        public Document createDocument() {

            return new WordDocument();

        }

    }

    public class PDFDocumentFactory extends DocumentFactory {

        @Override

        public Document createDocument() {

            return new PDFDocument();

        }

    }

    public class ExcelDocumentFactory extends DocumentFactory {

        @Override

        public Document createDocument() {

            return new ExcelDocument();

        }

    }

    public static void main(String[] args) {

        DocumentFactory wordFactory = new Q2().new WordDocumentFactory();

        Document wordDoc = wordFactory.createDocument();

        wordDoc.open();

        DocumentFactory pdfFactory = new Q2().new PDFDocumentFactory();

        Document pdfDoc = pdfFactory.createDocument();

        pdfDoc.open();

        DocumentFactory excelFactory = new Q2().new ExcelDocumentFactory();

        Document excelDoc = excelFactory.createDocument();

        excelDoc.open();

    }

}

Output:

