Exercise 2: Factory Method Pattern

Code:

public interface Document {  
 void open();  
}

public class WordDocument implements Document {  
 @Override  
 public void open() {  
 System.*out*.println("Opening Word document...");  
 }  
}

public class PdfDocument implements Document {  
 @Override  
 public void open() {  
 System.*out*.println("Opening PDF document...");  
 }  
}

public 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 class Main {  
 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();  
 }  
}

Output:

A screen shot of a computer

AI-generated content may be incorrect.