**Exercise 2: Implementing Factory Method Pattern**

**Document.java**

public interface Document {

void open();

}

**WordDocument.java**

public class WordDocument implements Document {

public void open() {

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

}

}

**ExceLDocument.java**

public class ExceLDocument implements Document {

public void open() {

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

}

}

**PdfDocument.java**

public class PdfDocument implements Document {

public void open() {

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

}

}

**DocumentFactory.java**

public abstract class DocumentFactory {

public abstract Document createDocument();

}

**WordDocumnetFactory.java**

public class WordDocumentFactory extends DocumentFactory {

public Document createDocument() {

return new WordDocument();

}

}

**PdfDocumentFactory.java**

public class PdfDocumentFactory extends DocumentFactory {

public Document createDocument() {

return new PdfDocument();

}

}

**ExceLDocumentFactory.java**

public class ExceLDocumentFactory extends DocumentFactory {

public Document createDocument() {

return new ExceLDocument();

}

}

**Main2.java**

public class Main2 {

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();

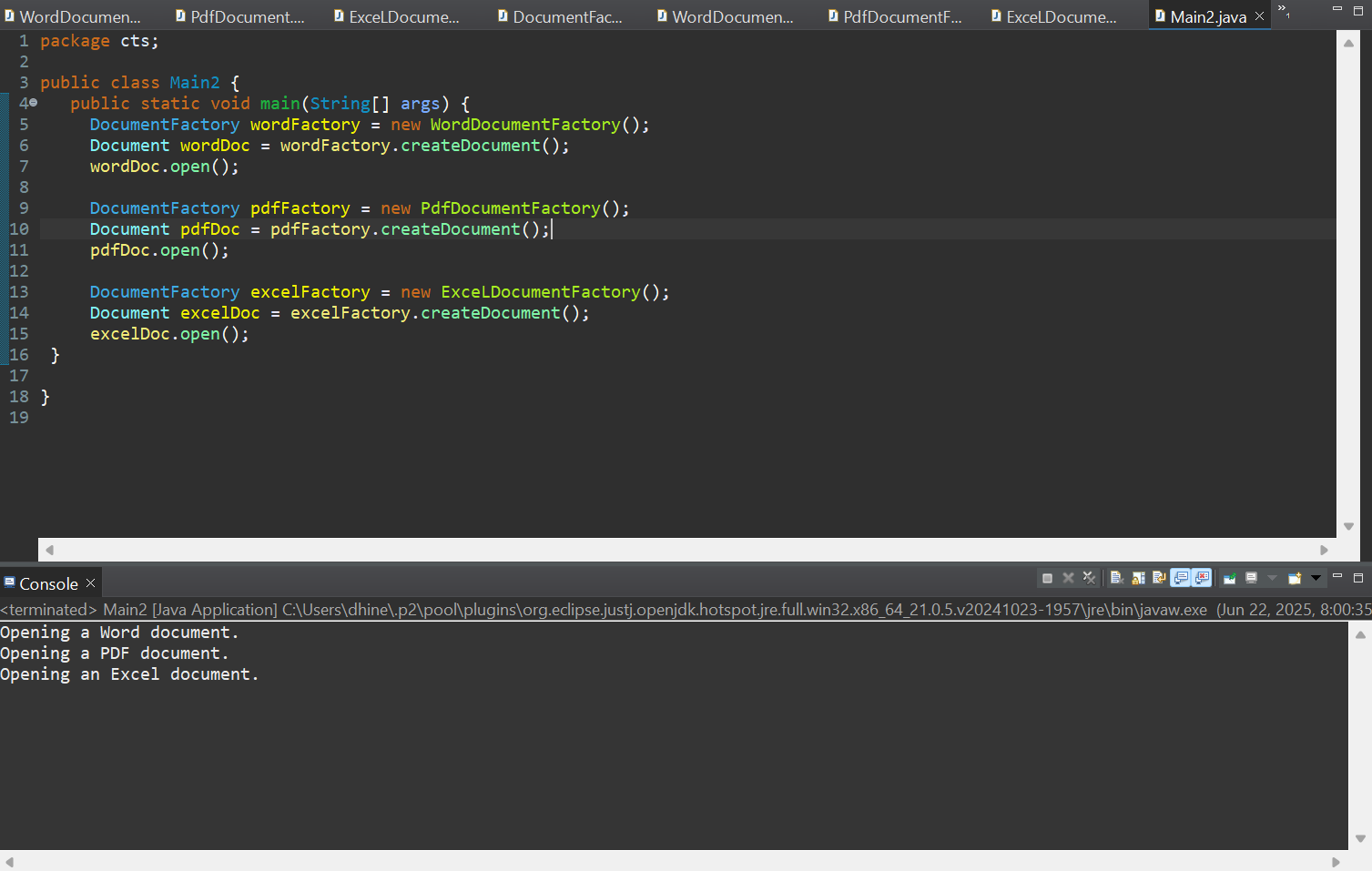
Document excelDoc = excelFactory.createDocument();

excelDoc.open();

}

}

**Output:**

****