**TASK 2**

**IMPLEMENTING THE FACTORY METHOD PATTERN**

**Document.java**

package factoryMethodPatternExample;

public interface document {

void open();

}

**WordDocument.java**

package factoryMethodPatternExample;

public class WordDocument implements document {

@Override

public void open() {

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

}

}

**PdfDocument.java**

package factoryMethodPatternExample;

public class PdfDocument implements document {

@Override

public void open() {

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

}

}

**ExcelDocument.java**

**package** factoryMethodPatternExample;

**public** **class** ExcelDocument **implements** document {

@Override

**public** **void** open() {

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

}

}

**WordDocumentFactory.java**

package factoryMethodPatternExample;

public class WordDocumentFactory extends DocumentFactory {

@Override

public document createDocument() {

return new WordDocument();

}

}

**PdfDocumentFactory.java**

package factoryMethodPatternExample;

public class PdfDocumentFactory extends DocumentFactory {

@Override

public document createDocument() {

return new PdfDocument();

}

}

**ExcelDocumentFactory.java**

package factoryMethodPatternExample;

public class ExcelDocumentFactory extends DocumentFactory {

@Override

public document createDocument() {

return new ExcelDocument();

}

}

**DocumentFactoryTest.java**

**package** factoryMethodPatternExample;

**public** **class** DocumentFactoryTest {

**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();

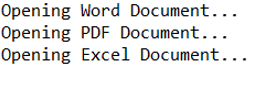
document excelDoc = excelFactory.createDocument();

excelDoc.open();

}

}

**OUTPUT:**

****