**Exercise 2: Implementing the Factory Method Pattern**

**Code:**

**Document Interface: -**

public interface Document {  
 void write();  
}

**Document Factory: -**

public class DocumentFactory {  
 public Document createDocument(String type){  
 if(type.equalsIgnoreCase("word")){  
 return new WordDocument();  
 }  
 else if(type.equalsIgnoreCase("pdf")){  
 return new PdfDocument();  
 }  
 return new ExcelDocument();  
  
 }  
}

**WordDocument class: -**

public class WordDocument implements Document{  
 @Override  
 public void write() {  
 System.*out*.println("I am a Word Document");  
 }  
}

**PdfDocument class: -**

public class PdfDocument implements Document{  
 @Override  
 public void write() {  
 System.*out*.println("I am a Pdf Document");

}  
}

**ExcelDocument class: -**

public class ExcelDocument implements Document{  
 @Override  
 public void write() {  
 System.*out*.println("I am a Excel Document");  
 }  
}

**TestClass (Main class): -**

public class TestClass {  
 public static void main(String[] args){  
 DocumentFactory factory=new DocumentFactory();  
 Document obj1=factory.createDocument("Word");  
 obj1.write();  
 Document obj2=factory.createDocument("PDF");  
 obj2.write();  
 Document obj3=factory.createDocument("Excel");  
 obj3.write();  
 Document obj4= factory.createDocument("xxxy");  
 obj4.write();  
 }  
}

**Output:**

**A screenshot of a computer program

AI-generated content may be incorrect.**