Design Patterns and Principles

Exercise 1: Implementing the Singleton Pattern

CODE:

Main.java | Logger.java

src > Logger.java > ...

```java
public class Logger {
    // Step 1: private static instance (eager initialization)
    private static Logger instance = new Logger();

    // Step 2: private constructor
    private Logger() {
        System.out.println(x:"Logger instance created.");
    }

    // Step 3: public method to return the same instance
    public static Logger getInstance() {
        return instance;
    }

    // Logger functionality
    public void log(String message) {
        System.out.println("[LOG] " + message);
    }
}
```

Main.java | Logger.java

src > Logger.java > ...

```java
public class Logger {
    // Step 1: private static instance (eager initialization)
    private static Logger instance = new Logger();

    // Step 2: private constructor
    private Logger() {
        System.out.println(x:"Logger instance created.");
    }

    // Step 3: public method to return the same instance
    public static Logger getInstance() {
        return instance;
    }

    // Logger functionality
    public void log(String message) {
        System.out.println("[LOG] " + message);
    }
}
```


Output:

```
[Running] cd "c:\Users\KIIT\Desktop\DotNet FSE\Engineering concepts\Design patterns and principles\SingletonPatternExample\src\" &&
javac Main.java && java Main
Logger instance created.
[LOG] Application started.
[LOG] Performing some operation.
Both logger1 and logger2 are the same instance.

[Done] exited with code=0 in 0.85 seconds
```

Exercise 2: Implementing the Factory Method Pattern

CODE:

Main.java

src > Main.java > ...

```java
public class Main {
    Run | Debug
    public static void main(String[] args) {
        // Word Document
        DocumentFactory wordFactory = new WordDocumentFactory();
        MyDocument wordDoc = wordFactory.createDocument();
        wordDoc.open();

        // PDF Document
        DocumentFactory pdfFactory = new PdfDocumentFactory();
        MyDocument pdfDoc = pdfFactory.createDocument();
        pdfDoc.open();

        // Excel Document
        DocumentFactory excelFactory = new ExcelDocumentFactory();
        MyDocument excelDoc = excelFactory.createDocument();
        excelDoc.open();
    }
}
```

WordDocument.java

src > WordDocument.java > ...

```java
public class WordDocument implements MyDocument {
    @Override
    public void open() {
        System.out.println(x:"Opening a Word document...");
    }
}
```

PdfDocument.java

src > PdfDocument.java > ...

```java
public class PdfDocument implements MyDocument {
    @Override
    public void open() {
        System.out.println(x:"Opening a PDF document...");
    }
}
```

ExcelDocument.java

src > ExcelDocument.java > ...

```java
public class ExcelDocument implements MyDocument {
    @Override
    public void open() {
        System.out.println(x:"Opening an Excel document...");
    }
}
```

DocumentFactory.java

src > DocumentFactory.java > ...

```java
public abstract class DocumentFactory {
    public abstract MyDocument createDocument();
}
```

PdfDocumentFactory.java

src > PdfDocumentFactory.java > ...

```java
public class PdfDocumentFactory extends DocumentFactory {
    @Override
    public MyDocument createDocument() {
        return new PdfDocument();
    }
}
```

WordDocumentFactory.java

src > WordDocumentFactory.java > ...

```java
public class WordDocumentFactory extends DocumentFactory {
    @Override
    public MyDocument createDocument() {
        return new WordDocument();
    }
}
```

ExcelDocumentFactory.java

src > ExcelDocumentFactory.java > ...

```java
public class ExcelDocumentFactory extends DocumentFactory {
    @Override
    public MyDocument createDocument() {
        return new ExcelDocument();
    }
}
```

OUTPUT:

```
PROBLEMS   OUTPUT   DEBUG CONSOLE   TERMINAL   PORTS        Filter        Code

[Running] cd "c:\Users\KIIT\Desktop\DotNet FSE\Engineering concepts\Design patterns and principles\FactoryMethodPatternExample\src\" && javac Main.java &&
java Main
Opening a Word document...
Opening a PDF document...
Opening an Excel document...

[Done] exited with code=0 in 0.806 seconds
```