**Exercise 1: Implementing the Singleton Pattern**

Codes :

Logger.java:

public class Logger {

private static Logger instance;

private Logger() {

System.out.println("Logger instance created.");

}

public static Logger getInstance() {

If (instance == null) {

instance = new Logger();

}

return instance;

}

public void log(String message) {

System.out.println("Log: " + message);

}

}

Main.java:

public class Main {

public static *void* main(String[] *args*) {

Logger logger1 = Logger.getInstance();

        logger1.log("First log message");

Logger logger2 = Logger.getInstance();

        logger2.log("Second log message");

if (logger1 == logger2) {

System.out.println("Both logger instances are the same (Singleton works correctly).");

        } else {

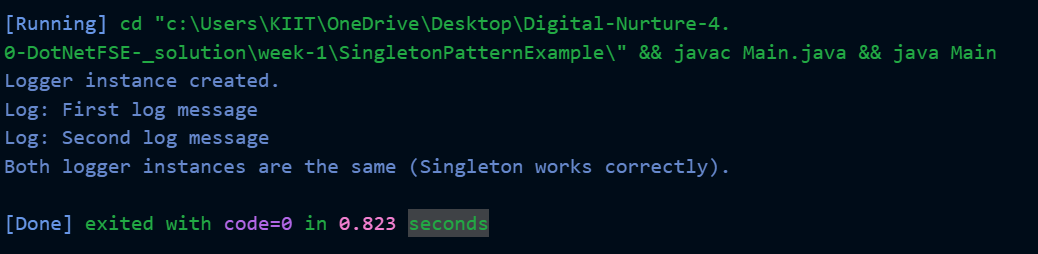
System.out.println("Logger instances are different (Singleton implementation failed).");

        }

    }

}

Output:



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

Code:

Main.java:

public class Main {

// Document Interface

    interface Document {

*void* open();

    }

// Word Document

    static class WordDocument implements Document {

        public *void* open() {

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

        }

    }

// PDF Document

    static class PdfDocument implements Document {

        public *void* open() {

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

        }

    }

// Excel Document

    static class ExcelDocument implements Document {

        public *void* open() {

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

        }

    }

// Abstract Factory

    abstract static class DocumentFactory {

        public abstract Document createDocument();

    }

// Word Factory

    static class WordFactory extends DocumentFactory {

        public Document createDocument() {

            return new WordDocument();

        }

    }

// PDF Factory

    static class PdfFactory extends DocumentFactory {

        public Document createDocument() {

            return new PdfDocument();

        }

    }

// Excel Factory

static class ExcelFactory extends DocumentFactory {

        public Document createDocument() {

            return new ExcelDocument();

        }

    }

// Main Method

    public static *void* main(String[] *args*) {

// Create and open Word Document

        DocumentFactory wordFactory = new WordFactory();

        Document wordDoc = wordFactory.createDocument();

        wordDoc.open();

// Create and open PDF Document

        DocumentFactory pdfFactory = new PdfFactory();

        Document pdfDoc = pdfFactory.createDocument();

        pdfDoc.open();

// Create and open Excel Document

        DocumentFactory excelFactory = new ExcelFactory();

        Document excelDoc = excelFactory.createDocument();

        excelDoc.open();

    }

}

Output:

