**Exercise 1: Implementing the Singleton Pattern**

**Code:**  
public class SingletonPatternExample {

    static class Logger {

        private static Logger instance;

        private Logger() {

            System.out.println("Logger initialized.");

        }

        public static Logger getInstance() {

            if (instance == null) {

                instance = new Logger();

            }

            return instance;

        }

        public void log(String message) {

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

        }

    }

    public static void main(String[] args) {

        Logger logger1 = Logger.getInstance();

        logger1.log("This is the first log message.");

        Logger logger2 = Logger.getInstance();

        logger2.log("This is the second log message.");

        if (logger1 == logger2) {

            System.out.println("Only one instance of Logger is used.");

        } else {

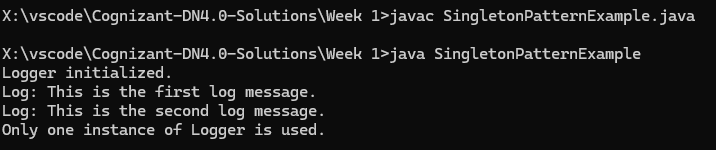
            System.out.println("Different instances exist! Singleton failed.");

        }

    }

}

**Output:**



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

**Code:**  
public class FactoryMethodPatternExample {

    abstract static class Document {

        public abstract void open();

    }

    static class WordDocument extends Document {

        @Override

        public void open() {

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

        }

    }

    static class PdfDocument extends Document {

        @Override

        public void open() {

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

        }

    }

    static class ExcelDocument extends Document {

        @Override

        public void open() {

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

        }

    }

    abstract static class DocumentFactory {

        public abstract Document createDocument();

    }

    static class WordDocumentFactory extends DocumentFactory {

        @Override

        public Document createDocument() {

            return new WordDocument();

        }

    }

    static class PdfDocumentFactory extends DocumentFactory {

        @Override

        public Document createDocument() {

            return new PdfDocument();

        }

    }

    static class ExcelDocumentFactory extends DocumentFactory {

        @Override

        public Document createDocument() {

            return new ExcelDocument();

        }

    }

    public static void main(String[] args) {

        DocumentFactory wordFactory = new WordDocumentFactory();

        DocumentFactory pdfFactory = new PdfDocumentFactory();

        DocumentFactory excelFactory = new ExcelDocumentFactory();

        Document wordDoc = wordFactory.createDocument();

        Document pdfDoc = pdfFactory.createDocument();

        Document excelDoc = excelFactory.createDocument();

        wordDoc.open();

        pdfDoc.open();

        excelDoc.open();

    }

}

**Output:**  
