**Exercise 1: Implementing the Singleton Pattern**

public class Main {

    static class Logger {

        private static Logger singleInstance;

        private Logger() {

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

        }

        public static Logger getInstance() {

            if (singleInstance == null) {

                singleInstance = new Logger();

            }

            return singleInstance;

        }

        public void log(String message) {

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

        }

    }

    public static void main(String[] args) {

        Logger logger1 = Logger.getInstance();

        Logger logger2 = Logger.getInstance();

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

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

        if (logger1 == logger2) {

            System.out.println("Both logger1 and logger2 refer to the SAME instance.");

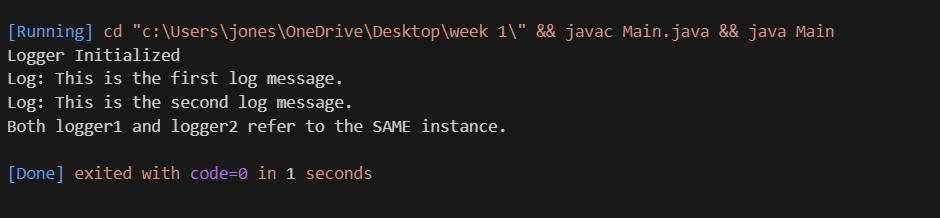
        } else {

            System.out.println("Different instances were created. Singleton failed.");

        }

    }

}

**Output :**  


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

**Document.java**

public interface Document {

    void open();

}

**DocumentFactory.java**public abstract class DocumentFactory {

    public abstract Document createDocument();

}

**ExcelDocument.java**  
public class ExcelDocument implements Document {

    public void open() {

        System.out.println("Excel document opened.");

    }

}

**ExcelFactory.java**  
public class ExcelFactory extends DocumentFactory {

    public Document createDocument() {

        return new ExcelDocument();

    }

}

**Main.java**  
public class Main {

    public static void main(String[] args) {

        DocumentFactory wordFactory = new WordFactory();

        Document wordDoc = wordFactory.createDocument();

        wordDoc.open();

        DocumentFactory pdfFactory = new PdfFactory();

        Document pdfDoc = pdfFactory.createDocument();

        pdfDoc.open();

        DocumentFactory excelFactory = new ExcelFactory();

        Document excelDoc = excelFactory.createDocument();

        excelDoc.open();

    }

}

**PdfDocument.java**

public class PdfDocument implements Document {

    public void open() {

        System.out.println("PDF document opened.");

    }

}

**PdfFactory.java**  
public class PdfFactory extends DocumentFactory {

    public Document createDocument() {

        return new PdfDocument();

    }

}

**WordDocument.java**

public class WordDocument implements Document {

    public void open() {

        System.out.println("Word document opened.");

    }

}

**WordFactory.java**

public class WordFactory extends DocumentFactory {

    public Document createDocument() {

        return new WordDocument();

    }

}

**Output:**  
