**Exercise 1: Implementing the Singleton Pattern**

Program:

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 class Exercise1 {

    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("Only one Logger instance exists.");

        } else {

            System.out.println("Multiple Logger instances found!");

        }

    }

}

Output:

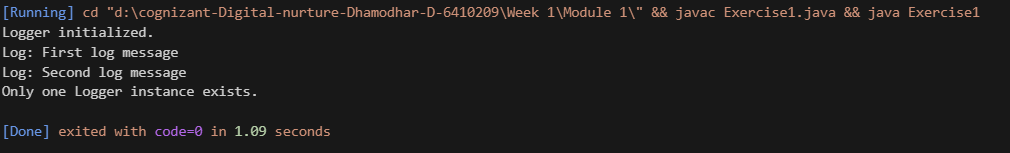
Logger initialized.

Log: First log message

Log: Second log message

Only one Logger instance exists.

Screenshot:



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

interface Document

{

    void open();

    void save();

    void print();

}

class WordDocument implements Document

{

    public void open()

    {

        System.out.println("Opening the word document");

    }

    public void save()

    {

        System.out.println("Saving the word document");

    }

    public void print()

    {

        System.out.println("Printing the word document");

    }

}

class PdfDocument implements Document

{

    public void open()

    {

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

    }

    public void save()

    {

        System.out.println("Saving the pdf document");

    }

    public void print()

    {

        System.out.println("Printing the pdf document");

    }

}

class ExcelDocument implements Document

{

    public void open()

    {

        System.out.println("Opening the excel document");

    }

    public void save()

    {

        System.out.println("Saving the excel document");

    }

    public void print()

    {

        System.out.println("Printing the excel document");

    }

}

class Factory

{

    public Document createDocument(String type)

    {

        if(type.equalsIgnoreCase("WORD"))

        {

            return new WordDocument();

        }

        else if(type.equalsIgnoreCase("PDF"))

        {

            return new PdfDocument();

        }

        else if(type.equalsIgnoreCase("EXCEL"))

        {

            return new ExcelDocument();

        }

        return null;

    }

}

public class FactoryMethodPattern {

    public static void main(String[] args) {

        Factory factory=new Factory();

        Document word=factory.createDocument("word");

        word.open();

        word.save();

        word.print();

        Document pdf=factory.createDocument("pdf");

        pdf.open();

        pdf.save();

        pdf.print();

        Document excel=factory.createDocument("excel");

        excel.open();

        excel.save();

        excel.print();

    }

}

Output :

