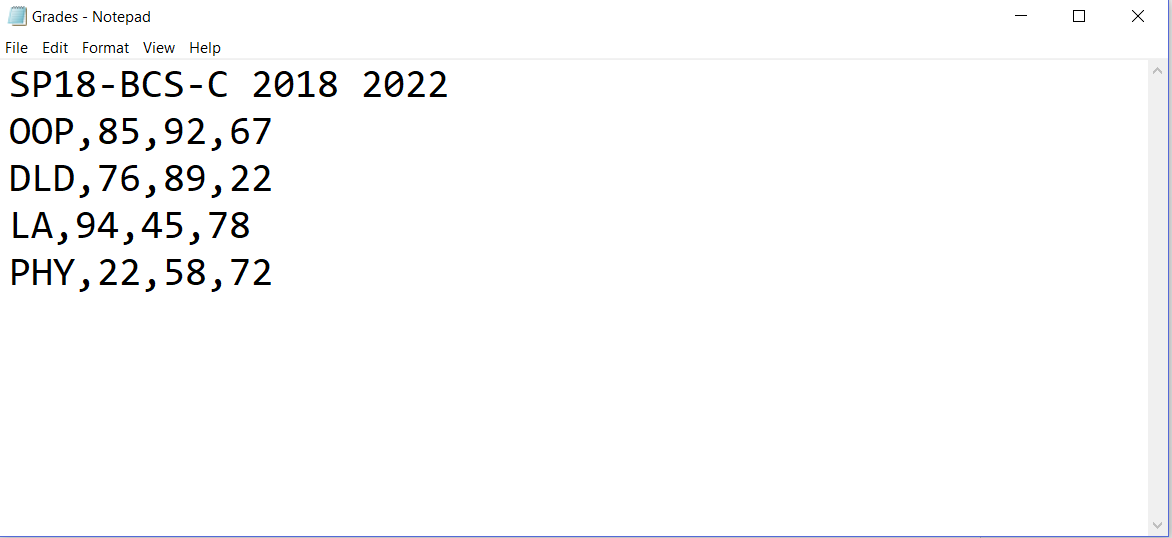
**File: **

Consider the above text file. Read it using Scanner class (used for reading “text” files) and show the output in the following format.

**Expected Output:**

Batch: SP18-BCS-C (4-year degree)

Result Summary:

OOP (Average: 78.5%)

DLD (Average: 62.0%)

LA (Average: 66.5%)

PHY (Average: 45.0%)

Sample Code:

S

**import java.io.FileReader;**

**import java.io.IOException;**

**import java.io.PrintWriter;**

**import java.util.Scanner;**

**public class LineNumberer{**

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

**Scanner console = new Scanner(System.in);**

**System.out.print("Input file: ");**

**String inFile = console.next();**

**System.out.print("Output file: ");**

**String outFile = console.next();**

**try{**

**FileReader reader = new FileReader(inFile);**

**Scanner in = new Scanner(reader);**

**PrintWriter out = new PrintWriter(outFile);**

**int lineNumber = 1;**

**while (in.hasNextLine()){**

**String line = in.nextLine();**

**out.println("/\* " + lineNumber +**

**" \*/ " + line);**

**lineNumber++;**

**}**

**out.close();**

**} catch (IOException exception){**

**System.out.println("Error processing file: " + exception);**

**}**

**}**

**}**

**Problem 2.** (Using object serialization to read/write data from/to a file)

Create a batch class having the following attributes:

**package** com.company;  
  
**import** java.io.\*;  
**import** java.util.Arrays;  
**import** java.util.Scanner;  
  
**public class** BatchClass {  
 **private** String **batchName**;  
 **private int startYear**, **endYear**;  
 **private final int numSubjects**;  
 **private** String[] **subjects**;  
 **private int**[][] **marks**;  
  
 **public** BatchClass(String batchName, **int** startYear, **int** endYear, **int** numSubjects, String[] subjects, **int**[][] marks) {  
 **this**.**batchName** = batchName;  
 **this**.**startYear** = startYear;  
 **this**.**endYear** = endYear;  
 **this**.**numSubjects** = numSubjects;  
 **this**.**subjects** = subjects;  
 **this**.**marks** = marks;  
 }  
  
 **public** String getBatchName() {  
 **return batchName**;  
 }  
  
 **public void** setBatchName(String batchName) {  
 **this**.**batchName** = batchName;  
 }  
  
 **public int** getStartYear() {  
 **return startYear**;  
 }  
  
 **public void** setStartYear(**int** startYear) {  
 **this**.**startYear** = startYear;  
 }  
  
 **public int** getEndYear() {  
 **return endYear**;  
 }  
  
 **public void** setEndYear(**int** endYear) {  
 **this**.**endYear** = endYear;  
 }  
  
 **public int** getNumSubjects() {  
 **return numSubjects**;  
 }  
  
 **public** String[] getSubjects() {  
 **return subjects**;  
 }  
  
 **public void** setSubjects(String[] subjects) {  
 **this**.**subjects** = subjects;  
 }  
  
 **public int**[][] getMarks() {  
 **return marks**;  
 }  
  
 **public void** setMarks(**int**[][] marks) {  
 **this**.**marks** = marks;  
 }  
  
 **public static void** main(String[] args) {  
 **final int** nS = 4;  
 String[] subs = {**"OOP"**, **"DLD"**, **"LA"**, **"PHY"**};  
 **int**[][] marks = {{85, 92, 67}, {76, 89, 22}, {94, 45, 78}, {22, 58, 72}};  
 BatchClass b1 = **new** BatchClass(**"SP18-BCS-C"**, 2018,  
 2022, nS, subs, marks);  
 }  
  
 @Override  
 **public** String toString() {  
 **return "BatchClass{"** +  
 **"batchName='"** + **batchName** + **'\''** +  
 **", startYear="** + **startYear** +  
 **", endYear="** + **endYear** +  
 **", numSubjects="** + **numSubjects** +  
 **", subjects="** + Arrays.*toString*(**subjects**) +  
 **", marks="** + Arrays.*toString*(**marks**) +  
 **'}'**;  
 }  
}

In the main method, an object of batch is created named b1 using fully parameterized constructor, where instance variables have the values as shown in the “grades.txt” file above.

Write the object to a file named “grades.dat” using object serialization. (Hint: Use ObjectOutputStream and FileOutputStream classes). Handle all checked exceptions as reported by the compiler.

Once the object is written to the file, in a separate java file, read back the Batch object from this file and display the output as shown in the expected output below (Hint: Use ObjectInputStream and FileInputStream classes). Handle all checked exceptions as reported by the compiler.

**Expected Output:**

Batch: SP18-BCS-C (4-year degree)

Result Summary:

OOP (Average: 78.5%)

DLD (Average: 62.0%)

LA (Average: 66.5%)

PHY (Average: 45.0%)

Sample Code:

**import** java.util.Scanner;

**import** javax.naming.directory.InvalidAttributeValueException;

**import** java.time.LocalDate;

**import** java.io.ObjectOutput;

**import** java.io.ObjectOutputStream;

**import** java.io.ObjectInput;

**import** java.io.ObjectInputStream;

**import** java.io.FileInputStream;

**import** java.io.FileNotFoundException;

**import** java.io.FileOutputStream;

**import** java.io.IOException;

**public** **class** A\_binary **implements** java.io.Serializable {

**private** **int** x, y;

**private** String s;

**public** **int** getX() {

**return** x;

}

**public** **void** setX(**int** x) {

**this**.x = x;

}

**public** **int** getY() {

**return** y;

}

**public** **void** setY(**int** y) {

**this**.y = y;

}

**public** String getS() {

**return** s;

}

**public** **void** setS(String s) {

**this**.s = s;

}

**public** A\_binary(**int** x, **int** y, String s) {

**super**();

**this**.x = x;

**this**.y = y;

**this**.s = s;

}

**public** **static** **void** writeA(String fileName)

{

A\_binary a1 = **new** A\_binary(786,420,"OOP");

**try** {

ObjectOutputStream oos = **new** ObjectOutputStream(**new** FileOutputStream(fileName));

oos.writeObject(a1);

**if**(oos!=**null**)

oos.close();

} **catch** (FileNotFoundException e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

} **catch** (IOException e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}

}

**public** **static** A\_binary readA(String fileName)

{

A\_binary a1=**null**;

**try**

{

ObjectInputStream ois = **new** ObjectInputStream(**new** FileInputStream(fileName));

a1 = (A\_binary) ois.readObject();

**if**(ois!=**null**)

ois.close();

**return** a1;

} **catch** (FileNotFoundException e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

} **catch** (IOException e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

} **catch** (ClassNotFoundException e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}

**finally**

{

}

**return** a1;

}

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

{

*writeA*("MyFile1.txt");

A\_binary a1 = *readA*("MyFile1.txt");

System.***out***.println(a1);

}

@Override

**public** String toString() {

**return** "A [getX()=" + getX() + ", getY()=" + getY() + ", getS()=" + getS() + "]";

}

} // class D ends