

COP-3337 Programming II

Programming Assignment 3: Input / Output, Interface, & Polymorphism

Florida International University

In this assignment, you implement a program for storing the personal information of students and teachers of a school in a comma-separated values (.csv) file.

1 Problem Specification

In order to implement the system, you need to write a java program that gets the personal information of individuals from the console and store it in different rows of the output .csv file.

1.1 Input Format

User enters personal information of n students and teachers using the console in the following format:

```
 $n$ 
Position1 Name1 StudentID1 TeacherID1 Phone1
Position2 Name2 StudentID2 TeacherID2 Phone2
Position3 Name3 StudentID3 TeacherID3 Phone3
...
Position $n$  Name $n$  StudentID $n$  TeacherID $n$  Phone $n$ 
```

Please note that the first line contains only an integer counting the number of lines following the first line.

In each of the n given input lines,

- Position must be one of the following three strings “student”, “teacher”, or “TA”.
- Name must be a string of two words separated by a single comma only.
- StudentID and TeacherID must be either “0” or a string of 5 digits. If Position is “teacher”, StudentID is zero, but TeacherID is not zero. If Position is “student”, TeacherID is zero, but StudentID is not zero. If Position is “TA”, neither StudentID nor TeacherID are zero.
- Phone is a string of 10 digits.

If the user enters information in a way that is not consistent with the mentioned format, your program must gracefully handle the situation by printing a message on the screen asking the user to partially/completely re-enter the information that was previously entered in a wrong format.

1.2 Data Structure, Interface and Classes

Your program must have an interface called “CSVPrintable” containing the following three methods:

- String getName ();
- int getID ();
- void csvPrintln (PrintWriter out);

You need to have two classes called “Student” and “Teacher” implementing CSVPrintable interface and another class called “TA” extending Student class. Both Student and Teacher classes must have appropriate variables to store Name and ID.

In order to store Phone, Student class must have a *phone* variable of type *long* that can store a 10-digit integer; while the Teacher class must have a *phone* variable of type *int* to store only the 4-digit postfix of the phone number.

Method getName has to be implemented by both Student and Teacher classes in the same way. Class Student must implement getID in a way that it returns the StudentID and ignores the TeacherID given by the input. Class Teacher must implement getID in a way that it returns the TeacherID and ignores the StudentID given by the input. Class TA must override the Student implementation of getID so that it returns the maximum value of StudentID and TeacherID.

Method csvPrintln has to be implemented by Student and Teacher classes (and overridden by TA class) so that it writes the following string followed by a new line on the output stream out:

getName() + “,” + getID() + “,” + phone

1.3 Output .csv File

The program must store the personal information of students, teachers and TAs in a comma-separated values (.csv) file called “out.csv”. You need to construct the output file by repetitively calling the csvPrintln method of every CSVPrintable object instantiated in your program. The output .csv file stores the information of every individual in a separate row; while each column of the file stores different type of information regarding the students and teachers (i.e. Name, ID and phone columns). Please note that you should be able to open the output file of your program using MS-Excel and view it as a table.

2 Sample Input/Output

Assume that the user enters the following four lines in console:

3

Teacher Alex,Martinez 0 98765 3053489999

Student Rose,Gonzales 56789 0 9876543210

TA John,Cruz 88888 99999 1234567890

The program must write the following content in the out.csv file.

Alex Martinez,98765,9999

Rose Gonzales,56789,9876543210

John Cruz,99999,1234567890

3 Submissions

You need to submit a *.zip* file compressing the Java source file(s) of your program (*.java* files).