# **Lab: Objects and Classes**

Test your tasks in the Judge system: https://judge.softuni.org/Contests/4482

## 1. Songs

Define a class called **Song** that will hold the following information about some songs:

- Type List
- Name
- Time

### **Input / Constraints**

- On the first line, you will receive the **number of songs N**.
- On the next N lines, you will be receiving data in the following format: "{typeList}\_{name}\_{time}".
- On the last line, you will receive **Type List** or "all".

### Output

- If you receive Type List as an input on the last line, print out only the names of the songs, which are from that Type List.
- If you receive the "all" command, print out the names of all the songs.

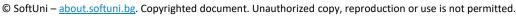
### **Examples**

Input	Output
<pre>favourite_DownTown_3:14 favourite_Kiss_4:16 favourite_Smooth Criminal_4:01 favourite</pre>	DownTown Kiss Smooth Criminal
favourite_DownTown_3:14 listenLater_Andalouse_3:24 favourite_In To The Night_3:58 favourite_Live It Up_3:48 listenLater	Andalouse
2 like_Replay_3:15 ban_Photoshop_3:48 all	Replay Photoshop

## 2. Students

Define a class called **Student**, which will hold the following information about some students:

















- first name string
- last name string
- age int
- home town string

#### Input

Read information about some students, until you receive the "end" command.

After that, you will receive a city name.

### Output

Print the students who are from the given city in the following format:

"{firstName} {lastName} is {age} years old."

### **Examples**

Input	Output
John Smith 15 Sofia Peter Ivanov 14 Plovdiv Linda Bridge 16 Sofia Simon Stone 12 Varna end Sofia	John Smith is 15 years old. Linda Bridge is 16 years old.
Anthony Taylor 15 Chicago David Anderson 16 Washington Jack Lewis 14 Chicago David Lee 14 Chicago end Chicago	Anthony Taylor is 15 years old. Jack Lewis is 14 years old. David Lee is 14 years old.

#### 3. Store Boxes

Define a class Item, which contains these properties: Name and Price.

Define a class Box, which contains these properties: Serial Number, Item, Item Quantity and Price for a Box.

Until you receive "end", you will be receiving data in the following format: "{Serial Number} {Item Name} {Item Quantity} {itemPrice}"

The Price of one box has to be calculated: itemQuantity \* itemPrice.

Print all the boxes ordered descending by price for a box, in the following format:

#### {boxSerialNumber}

- -- {boxItemName} \${boxItemPrice}: {boxItemQuantity}
- -- \${boxPrice}

The price should be formatted to the 2<sup>nd</sup> digit after the decimal separator.

### **Examples**

Input	Output
19861519 Dove 15 2.50	37741865













```
86757035 Butter 7 3.20
                                -- Samsung - $1000.00: 10
39393891 Orbit 16 1.60
                                -- $10000.00
37741865 Samsung 10 1000
                               19861519
                                -- Dove - $2.50: 15
end
                                -- $37.50
                               39393891
                                -- Orbit - $1.60: 16
                                -- $25.60
                               86757035
                                -- Butter - $3.20: 7
                                -- $22.40
48760766 Alcatel 8 100
                               97617240
97617240 Intel 2 500
                               -- Intel - $500.00: 2
83840873 Milka 20 2.75
                               -- $1000.00
35056501 SneakersXL 15 1.50
                               48760766
end
                               -- Alcatel - $100.00: 8
                                -- $800.00
                               83840873
                               -- Milka - $2.75: 20
                               -- $55.00
                               35056501
                               -- SneakersXL - $1.50: 15
                                -- $22.50
```

## 4. \*Vehicle Catalogue

Your task is to create a Vehicle catalog, which contains only Trucks and Cars.

Define a class **Truck** with the following properties: **Brand, Model, and Weight**.

Define a class **Car** with the following properties: **Brand, Model, and Horse Power**.

Define a class Catalog with the following properties: Collections of Trucks and Cars.

You must read the input, until you receive the "end" command. It will be in following format:

"{type}/{brand}/{model}/{horse power / weight}"

In the end, you have to print all of the vehicles ordered alphabetical by brand, in the following format:

"Cars:

{Brand}: {Model} - {Horse Power}hp

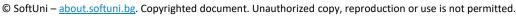
Trucks:

{Brand}: {Model} - {Weight}kg"

#### **Examples**

Input	Output
Car/Audi/A3/110 Car/Maserati/Levante/350 Truck/Mercedes/Actros/9019 Car/Porsche/Panamera/375 end	Cars: Audi: A3 - 110hp Maserati: Levante - 350hp Porsche: Panamera - 375hp Trucks: Mercedes: Actros - 9019kg
Car/Subaru/Impreza/152 Car/Peugeot/307/109	Cars: Peugeot: 307 - 109hp

















Subaru: Impreza - 152hp end













