CSE222 / BİL505
Data Structures and Algorithms
Homework #1

In this homework, you will design a customer tracking system using Java. You are expected to upload your files to Teams as a **zip** file.

Scenario

You are asked to write a complete customer tracking program. First, you should read the data you need from a text file. There will be 3 main types of data to read from the text file: order, customer, operator. Each operator may or may not have customers and each customer may or may not have orders. The customers can be either a retail customer or a corporate customer. Once you read the data from the text file, you should save this data in your program. Finally, you should take an input from the user: ID. If this ID corresponds to an operator/customer, you should print that operator's / customer's information. If not, you should inform the user.

Class Structure

You should have the following classes in your homework. These are must. However you can add new classes if you want (for example, a class to read the text file and a class to process the data you read).

1) person
2) operator (extends person)
3) customer (extends person)
4) retail_customer (extends customer)
5) corporate_customer (extends customer)
6) order

In the last page, you see the class diagram for this homework. You should stick into this format. The required variables and functions for each class are also written in the diagram. Again, the given variables/functions are must. However, you can use extra variables/functions to make your work easier/more readable.

Reading From The Text File

The text file you will read consists of multiple lines, each line describes a record of an order, a customer or an operator. Each line will be seperated by semicolon. Below, you see the expected format of a line in the text file for each class.

order:

order;product_name;count;total_price;status;customer_id

The first item in this line, "order", describe the type of data this line represents. So, for each order record, first item should be the same: "order". The following 5 items are the essential variables of the *order class*.

PS: The *status variable* of *order class* is an integer. However, while printing it as an output of the program, you should print the corresponding string. You can see the int-string matchings in the table below:

int value	corresponding string
0	Initialized
1	Processing
2	Completed
3	Cancelled

retail_customer:

retail_customer;name;surname;address;phone;ID;operator_ID

The first item in this line, "retail_customer", describe the type of data this line represents. So, for each retail_customer record, first item should be the same: "retail_customer". The following 6 items are the essential variables of the retail_customer class. As you can see, there is no item for *orders variable*. You should define *orders variable* of a customer (whether it's a retail_customer or a corporate_customer) by using the *define_orders()* function of the *customer class*. To do that, you will be checking the customer_id variable of each order you have read.

corporate_customer:

corporate_customer;name;surname;address;phone;ID;operator_ID;company_name

There are only 2 difference from a retail_customer record:

1) The first item should be "corporate_customer" instead of "retail_customer"
2) There should be an extra item for *company_name variable*.

Anything else said for retail_customer applies to corporate_customer records as well.

operator:

operator;name;surname;address;phone;ID;wage

The first item in this line, "operator", describe the type of data this line represents. So, for each operator record, first item should be the same: "operator". The following 6 items are the essential variables of the operator class. As can be seen, there is no item for *customers variable* of *operator class*. You should define the *customers variable* of an operator by using the *define_customers()* function of the *operator class*. To do that, you will be checking the operator_ID variable of each customer you have read (whether it's a retail_customer or a corporate_customer).

Input and Output

Other than the text file, you will also get an input from the user directly. This input will be an ID number.

- First, you will check if there is an operator with this ID. If so, you will print the personal information of that customer and also all of the customers of that operator (if any) including the orders of each customer (if any). If the operator has customer(s) you should also print the type of that customer (retail or corporate).

- If there is no operator with that ID number, you will check if there is a customer with that ID. If so, you will print the personal information of that customer, including all of the orders of that customer (if any).
- If there is no operator and no customer with the given ID number, you should inform the user.

After printing the output, the program terminates. No loop/menu is needed.

Error Handling:

You should check the inputs (both in text file and also in console) to make sure they are valid.

- All strings should have at least 1 character.
- All integers should be positive and small enough to be stored in an integer variable.
- There cannot be more than one operator with the same ID.
- There cannot be more than one customer with the same ID.
- An operator and a customer cannot have the same ID.
- A line in the text file should have all of the items described above, according to its type (order, retail_customer, corporate_customer or operator).
- A line in the text file cannot have more items than it should have.

You can assume:

- In each line of the text file: the items corresponding to a string variable will not have any special characters or numbers.
- There will be a text file with the given path.
    - The text file should be in the same folder where your .java files are.
    - The name of the file is “content.txt”.
- There will be at least one valid customer record in the text file.

Q: What to do if there is an invalid item in a line or if a line has missing/extra items?

A: Ignore that line. You don’t have to print anything about it to inform the user.

Q: What to do if the input given from the console (ID number) is not valid?

A: Inform the user and terminate.

Permittions and Restrictions:

- You cannot use any library other than the ones defined below:
    - java.io.File
    - java.util.Scanner (*not java.util.**)
- You can keep public variables in your main to keep the read data in arrays (basic arrays, not List or ArrayList etc.).
- You can define the size of the arrays you use as 100.
- You cannot change given variables, you should define and use them as described. You can use additional variables but you cannot unfit the given design.

Examples:

You can use the given text file to run and test your program. By using this text file, your program should give the exact same outputs for the inputs given in the examples below.

```
Please enter your ID...
500
*** Operator Screen ***
----------------------------
Name & Surname: gokhan kaya
Address: istanbul
Phone: +902626050004
ID: 500
Wage: 2000
----------------------------
Customer #1 (a retail customer):
Name & Surname: yakup genc
Address: kocaeli
Phone: +902626052201
ID: 1500
Operator ID: 500
Order #1 => Product name: tv - Count: 2 - Total price: 2000 - Status: Initialized.
Order #2 => Product name: computer - Count: 4 - Total price: 8000 - Status: Processing.
----------------------------
Customer #2 (a corporate customer):
Name & Surname: ibrahim sogukpinar
Address: kocaeli
Phone: +902626052202
ID: 1501
Operator ID: 500
Company name: gebze technical university
Order #1 => Product name: computer - Count: 2 - Total price: 5000 - Status: Completed.
----------------------------
```

```
Please enter your ID...
501
*** Operator Screen ***
----------------------------
Name & Surname: burcu yilmaz
Address: kocaeli
Phone: +902626050005
ID: 501
Wage: 1900
----------------------------
Customer #1 (a retail customer):
Name & Surname: yusuf sinan akgul
Address: kocaeli
Phone: +902626052203
ID: 1502
Operator ID: 501
Order #1 => Product name: smartphone - Count: 2 - Total price: 3000 - Status: Cancelled.
----------------------------
```

```
Please enter your ID...
502
*** Operator Screen ***
----------------------------
Name & Surname: didem gozupek kocaman
Address: istanbul
Phone: +902626050006
ID: 502
Wage: 2100
----------------------------
This operator doesn't have any customer.
----------------------------
```

```
Please enter your ID...
503
No operator/customer was found with ID 503. Please try again.
```

```
Please enter your ID...
1500
*** Customer Screen ***
Name & Surname: yakup genc
Address: kocaeli
Phone: +902626052201
ID: 1500
Operator ID: 500
Order #1 => Product name: tv - Count: 2 - Total price: 2000 - Status: Initialized.
Order #2 => Product name: computer - Count: 4 - Total price: 8000 - Status: Processing.
```

```
Please enter your ID...
1501
*** Customer Screen ***
Name & Surname: ibrahim sogukpinar
Address: kocaeli
Phone: +902626052202
ID: 1501
Operator ID: 500
Company name: gebze technical university
Order #1 => Product name: computer - Count: 2 - Total price: 5000 - Status: Completed.
```

```
Please enter your ID...
1502
*** Customer Screen ***
Name & Surname: yusuf sinan akgul
Address: kocaeli
Phone: +902626052203
ID: 1502
Operator ID: 501
Order #1 => Product name: smartphone - Count: 2 - Total price: 3000 - Status: Cancelled.
```

```
Please enter your ID...
1503
No operator/customer was found with ID 1503. Please try again.
```

General Information About the Homework:

- Cheating is not permitted. The students who cheat will receive NA from the course.
- You will be asked to make a demo to be graded. If you don't attend to demo or cannot give proper answers to the questions you were asked during the demo, you will get 0.
- The Problem Session on February 27 will describe the homework in detail. If you have any questions, you should ask them in PS. Any other questions before/after the PS will be ignored unless there is a mistake or missing information in this PDF. In such a case, the announcement will be made on Teams, you are responsible for reading them in time.
- Late submissions will not allowed. The due date will not postponed.

Grading:

To be graded, you should read the file successfully and print the information according to input (ID).

Associating orders with customers and printing a customer's info (including orders, if any).	30 pts (No partial grading)
Associating customers with operators and printing an operator's info (including customers, if any).	35 pts (No partial grading)
Error Handling	35 pts
Total	100 pts

Additional grading details:

Code is not compilable	-100 pts (You get 0 directly)
Not attending demo / not providing acceptable explanations during demo	-100 pts (You get 0 directly)
Not a proper OOP design as described above	-100 pts (You get 0 directly)
Cheating	You get NA from the course
Order status is not printed as string	-10 pts
Additional library usage (other than the ones described above)	-30 pts
Printing all customers/operators instead of reading an input from the console	-30 pts
Using different data types other than String, int, and arrays.	-50 pts

PS: Additional grading criteria may apply.

Class Diagram