Due Date: 16.11.2022 23:55

CENG211 – Programming Fundamentals
Homework #2

In this homework you are expected to implement a "Vehicle Price Calculator" in Java.
You should fulfill the concepts of:

- Inheritance
- Polymorphism
- ArrayLists

In this application there are sold vehicles which has vehicle ID, month of sale, city of sale and production year. The sold vehicles are automobile, pickup truck, and bicycle. An automobile could be either a sedan, a hatchback, or a minivan.

A file that consists of different sold vehicles' information is given: HW2_SoldVehicles.csv. The columns of this file is given in the following for each different sold vehicle.

Vehicles:

Sedan: ***vehicle ID, month of sale, city of sale, production year, roof type, engine volume, VAT (%)***

Hatchback: ***vehicle ID, month of sale, city of sale, production year, city mode, engine volume, VAT (%)***

Minivan: ***vehicle ID, month of sale, city of sale, production year, number of seats, engine volume, VAT (%)***

Pickup Truck: ***vehicle ID, month of sale, city of sale, production year, cab type, truck bed type, VAT (%)***

Bicycle: ***vehicle ID, month of sale, city of sale, production year, chain type, seat post, VAT (%)***

NOTE: VAT = (Value-added tax)

The vehicle ID's of sold vehicles start with prefix 'S' for sedan, 'H' for hatchback and so on. The information that are given in the attached file will be used to calculate the SCT (special consumption tax) of each sold vehicle. Each column of information, such as number roof type in a sedan, might affect the final value of a sold vehicle SCT. The following tables show the SCT values of different columns such as the roof type of a sedan.

Month of sale	January	May	August	October	December
SCT value	0.3	0.4	0.5	0.6	0.7

City of sale	Izmir	Istanbul	Ankara
SCT value	0.1	0.3	0.2

Production Year	2001-2008	2012-2017	2018-2022
SCT value	1.0	1.2	1.6

Roof type	Regular	Moonroof	Sunroof
SCT value	0.5	0.6	0.8

City mode	yes	no
SCT value	0.15	0.1

Number of seats	4	5	6	7
SCT value	0.1	0.4	0.6	0.8

Cab type	regular	extended	crew
SCT value	2.5	2.8	3

Chain type	derailleur	onechain	doublechain
SCT value	1.1	1.2	1.3

Truck bed type	regular	tanker	trailer
SCT value	0.5	0.8	1.0

Seat post	carbonfiber	steel	aluminum	titanium
SCT value	0.8	0.7	0.9	0.6

The following formulas show how the SCT of each sold vehicle is calculated. Note that, the meaning of for example, city of sale, in formula is the SCT value of city of sale in table above. However, the value of for example, engine volume could be obtained from the attached file.

SCT calculations:

SCT of sedan = (engine volume * 0.2 * roof type) / production year

SCT of hatchback = (engine volume * 0.3 * production year) + (city mode)

SCT of minivan = (0.6 * production year) / (engine volume + number of seats)

SCT of pickup truck = (truck bed type * production year) / cab type

SCT of bicycle = (chain type * seat post * 0.1) + month of sale

After the calculation of SCTs of vehicles, you will calculate the total price paid by the buyer for each sold vehicle. In order to calculate prices, the calculations are given below.

Total price calculations:

Total price paid for automobile = (Base Price) * (1 + SCT * 0.8) + (1 + VAT/100)

Total price paid for pickup truck = (Base Price) * (1 + (SCT * 0.6)) + (1 + VAT/100)

Total price paid for bicycle = (Base Price * 0.9) * (1 + SCT) + (1 + VAT/100)

Category	Automobile	Pickup Truck	Bicycle
Base Price (TL)	200 000	250 000	10 000

As addition of these, in this application, there is a sales record class which holds an array list of the sold vehicles. There should be methods in this class which are explained in the following:

- A method that returns an array list of all the sold sedan object
- A method that returns an array list of all the sold hatchback object
- A method that returns an array list of all the sold minivan object
- A method that returns an array list of all the sold pickup truck object
- A method that returns an array list of all the sold bicycle object

Additional methods could be added to the sales record class if necessary.

In this application, you are expected to read the given files automatically and to create the necessary objects. You should create a sales record that holds all these created objects and could classify these into categories such as sedan or minivan etc. In this application the only input that is taken from the user is an integer. Please do not ask **any other input**. With the help of sales record, the sold vehicles' price should be calculated and outputted on the console as in the following:

```
********************************************************************

Please press,

1 to see all sold vehicles list

2 to see sold sedan list

3 to see sold hatchback list

4 to see sold minivan list

5 to see sold pickup truck list

6 to see sold bicycle list

Please enter your choice: 1

Vehicle: Minivan Vehicle ID: M235 Month: August City: Izmir Production Year: 2022 SCT: 0.***

The total price paid for M235 is: ............ TL

Vehicle: Bicycle Vehicle ID: B154 Month: January City: Izmir Production Year: 2020 SCT: 0.***

The total price paid for B154 is: ............ TL

********************************************************************
```

Note that, all of the calculations and SCT values are made up and are not taken from real life.

Important Notes:

1. You can use standard **java.io** packages to read files. Do NOT use other 3rd party libraries.

2. You should use **relative** paths (e.g. Files/sample.csv) instead of **absolute** paths (e.g. C:\\user\\eclipse-workspace\\MyProject\\Files\\sample.csv). Please be sure of it, otherwise there will be **no output** of your application and you certainly will **lose points**.

3. To support **Turkish characters,** you may need to change your project's text file encoding to UTF8: Right click on your project (in package explorer) → Properties → Text file encoding → Other → UTF8 → Apply.

4. You are expected to write clean, readable, and tester-friendly code. Please try to maximize reusability and prevent from redundancy in your methods.

Assignment Rules:

1. In this lecture's homework, there are no cheating allowed. If any cheating has been detected, they will be graded as 0 and there will be no further discussion on this.
2. You are expected to submit your homework in groups. Therefore, only one of you will be sufficient to submit your homework.
3. Make sure you export your homework as an Eclipse project. You can use other IDEs as well, however, you must test if it **can be executed** in Eclipse.
4. Submit your homework through Cloud-LMS.
5. Your exported Java Project should have the following naming format with your assigned group ID (which will be announced on MS Teams) as the given below:

   G05_CENG211_HW2

   Also the zip folder that your project in should have the same name

   G05_CENG211_HW2.zip
6. Please beware that if you do not follow the assignment rules for exporting and naming conventions, you will lose points.
7. Please be informed that your submissions may be anonymously used in software testing and maintenance research studies. Your names and student IDs will be replaced with non-identifying strings. If you do not want your submissions to be used in research studies, please inform the instructor (Dr. Tuglular) via e-mail.