CSE212: SOFTWARE DEVELOPMENT METHODOLOGIES

YEDITEPE UNIVERSITY

SPRING 2022

ASSIGNMENT 1 – DUE DATE MARCH 18TH, 2022

This semester, you will be implementing a *Fuel Station Management System* as part of your assignment series. In this context, you are a large corporate fuel company that wants to streamline management on its fuel stations.

For the first assignment, you are asked to implement the following classes and their respective instance variables:

Gasoline:
`origin (String), pricePerLiter (double), totalLiters (double)`

Diesel:
`origin (String), pricePerLiter (double), totalLiters (double)`

Station:
`stationName (String), ID (int), gasolineArray (Gasoline[]), dieselArray (Diesel[])`

Also, you should maintain the following array in your main method:

```
Station[] stationArray
```

(For simplicity the array sizes should be set to 10)

Your application should be able to handle the following scenario and requirements:

- You have to create a Menu to display the menu items. When a user starts your application, it should prompt the following menu:

```
1. Create a new station
2. Add gasoline to a station inventory
3. Add diesel to a station inventory
4. Display a station inventory
5. Exit
```

- When the user selects the 1st option, the program should create a new `Station` object and add it to the `stationArray`. So firstly, the program should ask the station name and station ID and by using this information, it should create the `Station` object.
- When the user selects the 2nd option, the program should ask for a station ID to search the `stationArray` and find the `Station` object that has the given ID. When the object is found, then the program should create a `Gasoline` object by requesting the information from the user. This `Gasoline` object then should be added to the `gasolineArray` of the corresponding `Station` object. If there are no `Stations` in the `stationArray` with the given ID, then the program should print "No station found with the given ID!".
- When the user selects the 3rd option, the program should ask for a station ID to search the `stationArray` and find the `Station` object that has the given ID. When the object is found, then the program should create a `Diesel` object by requesting the information from the user. This `Diesel` object then should be added to the `dieselArray` of the corresponding `Station` object. If there are no `Stations` in the `stationArray` with the given ID, then the program should print "No station found with the given ID!".
- When the user selects the 4th option, the program should ask for a station ID to search the `stationArray` and find the `Station` object that has the given ID. When the object is found, then the program should print all of the `Gasoline` and `Diesel` inventory by iterating over the `gasolineArray` and `dieselArray` of the corresponding `Station` object. If there are no `Stations` in the `stationArray` with the given ID, then the program should print "No station found with the given ID!".
- When a user selects the 5th option, your application should terminate.

Rules that you must follow

1. You are given a MainClass.java file. Here you can find the main method and the general outline of the menu structure. You can also find the necessary method calls that perform the 1st, 2nd, 3rd, and 4th options of the menu.
    a. *public static Station createStation()*
    b. *public static void findStationAndAddGasoline(Station[] stationArray)*

c. *public static void findStationAndAddDiesel(Station[] stationArray)*
d. *public static void displayStationInventory(Station[] stationArray)*

2. These methods listed above must be defined in the Station class and should be static such that you can access them from the main method directly. What they do is self explanatory as it can be understood from their name.
3. You are **not allowed** to make any changes to the MainClass.java file.
4. Your class, variable, method, etc. names should be defined as indicated in this PDF file unless stated otherwise.
5. **If your code does not compile you won't get any credit!**
6. Your program output should **match** the example given below. Keep in mind the spaces, empty lines, asking the user input in the same line, etc.
7. Write each class in a separate .java file.

Application Walkthrough

```
1. Create a new station
2. Add gasoline to a station inventory
3. Add diesel to a station inventory
4. Display a station inventory
5. Exit
1
Please enter the name of the Station: JavaPetrol
Please enter the Station ID: 1234

1. Create a new station
2. Add gasoline to a station inventory
3. Add diesel to a station inventory
4. Display a station inventory
5. Exit
1
Please enter the name of the Station: NeoPetrol
Please enter the Station ID: 4321

1. Create a new station
2. Add gasoline to a station inventory
3. Add diesel to a station inventory
4. Display a station inventory
5. Exit
2
Please enter the ID of the Station you want to search: 1111

No station found with the given ID!
```

```
1. Create a new station
2. Add gasoline to a station inventory
3. Add diesel to a station inventory
4. Display a station inventory
5. Exit
2
Please enter the ID of the Station you want to search: 1234

Please enter the origin of gasoline: Middle East
Please enter the price per liter: 1.8
Please enter the total shipment volume in liter: 1500

1. Create a new station
2. Add gasoline to a station inventory
3. Add diesel to a station inventory
4. Display a station inventory
5. Exit
2
Please enter the ID of the Station you want to search: 1234

Please enter the origin of gasoline: Venezuela
Please enter the price per liter: 2.0
Please enter the total shipment volume in liter: 1000

1. Create a new station
2. Add gasoline to a station inventory
3. Add diesel to a station inventory
4. Display a station inventory
5. Exit
2
Please enter the ID of the Station you want to search: 1234

Please enter the origin of gasoline: Russia
Please enter the price per liter: 2.5
Please enter the total shipment volume in liter: 500

1. Create a new station
2. Add gasoline to a station inventory
3. Add diesel to a station inventory
4. Display a station inventory
5. Exit
3
Please enter the ID of the Station you want to search: 1234

Please enter the origin of Diesel: Middle East
Please enter the price per liter: 1.5
Please enter the total shipment volume in liter: 2500

1. Create a new station
2. Add gasoline to a station inventory
3. Add diesel to a station inventory
```

```
4. Display a station inventory
5. Exit
3
Please enter the ID of the Station you want to search: 4321

Please enter the origin of Diesel: Middle East
Please enter the price per liter: 1.5
Please enter the total shipment volume in liter: 5000

1. Create a new station
2. Add gasoline to a station inventory
3. Add diesel to a station inventory
4. Display a station inventory
5. Exit
4
Please enter the ID of the Station you want to display: 1000

No station found with the given ID!

1. Create a new station
2. Add gasoline to a station inventory
3. Add diesel to a station inventory
4. Display a station inventory
5. Exit
4
Please enter the ID of the Station you want to display: 1234

Displaying the inventory of Station #1234

Gasoline...
The origin is: Middle East
Price per liter is: 1.8
Total liters of this gasoline is: 1500.0

Gasoline...
The origin is: Venezuela
Price per liter is: 2.0
Total liters of this gasoline is: 1000.0

Gasoline...
The origin is: Russia
Price per liter is: 2.5
Total liters of this gasoline is: 500.0

Diesel...
The origin is: Middle East
Price per liter is: 1.5
Total liters of this Diesel is: 2500.0

1. Create a new station
2. Add gasoline to a station inventory
```

```
3. Add diesel to a station inventory
4. Display a station inventory
5. Exit
4
Please enter the ID of the Station you want to display: 4321

Displaying the inventory of Station #4321

Diesel...
The origin is: Middle East
Price per liter is: 1.5
Total liters of this Diesel is: 5000.0

1. Create a new station
2. Add gasoline to a station inventory
3. Add diesel to a station inventory
4. Display a station inventory
5. Exit
5
```

Submit your assignments in a zip file, which has your student number as name, through the YULEARN (https://yulearn.yeditepe.edu.tr) latest by the end of Friday, March 18th, 2022.