

### Question 1:

Make a Command Line System for Vehicle Showroom. As it is a vehicle showroom, so it will have some vehicles in it. A "vehicle" has following descriptions:

1. model number
2. engine type

There are three types of engines

- a. oil
- b. gas
- c. diesel

3. engine power
4. tire size

In the vehicle showroom there are 3 types of vehicles

- Normal Vehicles, Sports Vehicles & Heavy Vehicles.

For "Normal Vehicle" There is no extra property, so it will have all the descriptions as above described descriptions of a "vehicle".

For "Sports Vehicle" There are two extra properties or attributes you can say. And these are :

1. turbo
2. engine type is oil only

For "Heavy Vehicle" There are also two extra properties or attributes you can say. And these are :

1. weight
2. engine type is diesel only

Now, your system should be able to have following functionalities:

- a. Add any type of vehicle in showroom.
- b. Remove any of the vehicles from showroom.
- c. Show the vehicle lists with details currently have in showroom.

Here comes the exciting part. Without visitors a vehicle showroom is not good, right?

The regular expected visitor count of the vehicle showroom is 30.

But whenever we add a new "Sports Vehicle" in the showroom the expected visitor count increase by 20.

So here comes a new functionality to your system:

- d. Show the list of vehicles with current expected visitor count

There is no boundary regarding the console panel representation. You should represent it in your creative way.

You can also add extra properties to your vehicle details if you need and even the data types for the attributes are all to you.

But whatever described above is a must have.

Enjoy yourself making Vehicle Showroom System. :)

### Question 2:

I know, I know you are bored with the command line. So, here comes an exciting task for you.

You have to build an "Web Application" which contains two pages. One is the "Login" Page and Other is "Landing" Page. The details are described below :

- a. Login Page:

It is basically the typical login screen you see in almost every websites you visit.

Where user will enter "Email" and "Password" and submit (authentication).

Please note that the input for Email should accept Email only.

b. Landing Page:

There is basically two page. One is for "Success" and another is for "Error".

Based on result of the authentication app will show another page with "Error" or "Success".

To make you excited, there is also some bonus factors. These are not mandatory. But if you have it, it will be act as some bonus points for you.

These are:

1. Responsive UI
2. Framework
3. Dependency Management
4. Version Control such as git.
5. Database
6. Object Relation Mapping

You can use any Language you want to build the application. It's totally up to you.

Languages Preferred : JAVA, JavaScript( NodeJS Environment ), Python.

Enjoy making systems. Wish you good luck. :)

Question 1 is mandatory. Feel free to ask any question. Use Github public repository to submit your solution(s).

For further question and to provide your answer please contact: [nahid.rahman@dsinnovators.com](mailto:nahid.rahman@dsinnovators.com)