Auto Shop

I would like to create a program which simulates a car tuning shop. I heard a lot of talk about doing something like a library or grocery store, and the overall idea sounded fun, but I wanted to change it a little as I figured many people would’ve had similar ideas. The idea is that you will have a car which is yours, and it will have stats. You can shop around in a tuner shop and look through car parts to buy, to upgrade your car. You go through checkout and now you own the parts. In the garage you can view your car’s parts and stats. I could possibly decide to add an actual economy where you earn money by doing a drag race and have to manage money. But I will prioritize the previous things first.

The intended user is a person into cars who wants to chew some time, or someone who wants a little bit of light knowledge on car parts. I may include descriptions on the upgraded car parts which talk about how they are better. And how the part in general works, in a way for a user new to cars would understand.

The project is trying to be something someone might use for a few minutes to entertain themselves and learn a bit on how cars work and how parts affect them.

I will use classes and objects, object serialization, java, it may console based, and I will use ArrayLists for storing data. If I have time and effort to spare, I might try to make a GUI but I’d rather have the program working in a console based setting first.

Use Case

Console:

Current car: none

1. Buy car
2. Auto Shop
3. Exit
4. Garage

User:

1

Console:

1. BMW M2
2. Ferrari F8 Tributo
3. Lexus LFA
4. BMW M5
5. Back

User:

3

Console:

Lexus LFA

Engine: 4.8L V10

Power: 552 HP @ 8700 rpm

Torque: 488 ft-lb @ 6800 rpm

Weight: 1,480 kg

0-60 Mph: 3.7 seconds

Top Speed: 202 Mph

Would you like to buy the Lexus LFA?

1. Confirm
2. Cancel

User:

1

Console:

Lexus LFA acquired.

Console:

Current car: Lexus LFA

1. Buy car
2. Auto Shop
3. Exit
4. Garage

User:

4

Console:

1. Lexus LFA
2. Back

User:

1

Console:

Lexus LFA

Engine: 4.8L V10

Power: 552 HP @ 8700 rpm

Torque: 488 ft-lb @ 6800 rpm

Weight: 1,480 kg

0-60 Mph: 3.7 seconds

Top Speed: 202 Mph

1. Get in car
2. Back

User:

1

Console:

Lexus LFA selected.

Console:

1. Lexus LFA
2. Back

User:

2

Console:

Current car: Lexus LFA

1. Buy car
2. Auto Shop
3. Exit
4. Garage

User:

2

Console:

Welcome to the auto shop.

1. Back
2. Engine Stage 1
3. Weight Reduction Stage 1
4. Racing Transmission

User:

1

Console:

Engine upgrade bought.

1. Back
2. Engine stage 2
3. Weight reductio stage 1
4. Racing Transmission

User:

0

Console:

Current car: Lexus LFA

1. Buy car
2. Auto Shop
3. Exit
4. Garage

User:

4

Console:

1. Lexus LFA
2. Back

User:

1

Console:

Lexus LFA

Engine: 4.8L V10 Stage 1

Power: 623 HP @ 8700 rpm

Torque: 520 ft\_lb @ 6800 rpm

Weight: 1,480 kg

0-60 Mph: 3.4 seconds

Top Speed: 202 Mph

1. Get in car
2. Back

You get the gist of the user flow. There wouldn’t be any extra special users with extra access, like admins for example.

Data Design

The data is mainly about the cars themselves, particularly the stats attached to each of them. Each upgraded tuning part will have their own stats, which overwrite the car stats when equipped. A car class which lays out the main properties and methods which every car would contain. Then a car object with stats inherited from the car class, with the stat values depending on what the car is. There would be classes for each kind of car part, such as the exhaust and suspension. Then an exhaust object with a horsepower number that overwrites the car’s current horsepower when equipped.

Each car object which the user has in their garage would probably be stored in an ArrayList. The data will be need to be persistent across navigating menus and closing/opening the program, which will happen through object serialization. The user’s garage will be stored in an ArrayList, and the 0th index will be the user’s equipped car. Dealership will have it’s own ArrayList which is left untouched.

UI Design

Graphical user interface, application, Teams

Description automatically generated

Algorithm

Create a program which simulates the experience of owning a car and tuning it up at a shop, as well as servicing.

Input will be user choices based on the menu

Output will be text printed, stats, cars, parts, choices.

Make car class, garage class, menu class, tuner shop class, dealership class, engine class, car body class, transmission class. Menu class will contain car objects which are initialized, and program will check for save data. The cars and their characteristics will be created with setters and getters, and the car upgrade objects will be upgraded. All the objects will be organized into their respective ArrayLists. Print out the main menu and take the user input. If the user goes to the dealership, it will load in the dealership ArrayList and print out the car choices. Take the user input, print out the car stats and take confirmation of purchase. If the user buys, add the car to the initially empty garage ArrayList. The 0th index in the ArrayList will be the user’s selected car. Back in the menu, if the user goes to the garage, show the garage ArrayList. When a user selects a car, it will be moved to the 0th index of the ArrayList. In the tuning shop, each category of car parts for each car will be put into their own ArrayLists. Depending on the car selected, the according ArrayList of car parts will be shown to the user. When the user selects a part, a setter will overwrite the respective stats of the car affected by the part.

UML

Graphical user interface, application, Teams

Description automatically generated