2017

Sarang Kharche

150740395



Introduction to Object Oriented Programming by Mustafa Bozkurt

**Table of Contents**

[Introduction 2](#_Toc501403794)

[Approach 2](#_Toc501403795)

[Issues Faced 3](#_Toc501403796)

[Functions 3](#_Toc501403797)

[Limitations 4](#_Toc501403798)

[Testing 4](#_Toc501403799)

# **Introduction**

This topic of this project is Car Rental System, being said it helps the admin to create fleet, rent out cars, display rented cars, display available fleet and other minimal functions. This program is written in C++ and compiled with g++ in Linux environment (Ubuntu 17.04).

The way this system is designed is very straight forward to use, the user of the system who will be the admin and access all the available functions available. In a car rental system, it’s important that user can create fleet as in input available car in the system, then assign the customer to specific car, should be able to see the rented cars and the available cars.

This system will allow the user to carry all of these functionalities. To access these main functions user can use 1,2,3,4,5 numerical.

Following is the description of the functions:

1st Function: User can create fleet.

2nd Function: Rent out the car.

3rd Function: Check the fleet that is rented.

4th Function: Check the available fleet.

5th Function: Exit the system.

These functionalities will help the user to perform the car rental procedures.

# **Approach**

Firstly, I worked on the required functions by the system in accordance with that I created the Class rentalSystem. For the functions of the display I used the i\*j grid to form 2\*8 grip of available cars.

I made rentalsSystems the class and used it as the main class to implement other functions, after deciding the no. of functions to be made, I then worked on the main program function which displays the main menu at the start of the program, from this bit user can access other functions.

I believe I took a conceptual approach to develop this program, with rough design, class diagram and then implementing it in C++.

# **Issues Faced**

The main issue I faced was to come up with system procedure because with car rental there are many ways in which it could work. With many different methods of car rental management, I chose what I think is an effective one, as it’s procedural use makes it easy to use and effective as well.

Another issue I got stuck at was to make the 2D display grid, as the car needs to be assigned to specific bay, when I was implementing this function it would often generate error and the assigned car would not be assigned.

# **Functions**

**Create Fleet:** Create a fleet that can be rented by the customer. To store the fleet details like Brand name, model, registration, no. of seats in the car, pick up location, drop off location, these details will then be used to create fleet.

**Rent Cars:** This function will rent the available fleet to the customer. When the user will enter the brand name, if the fleet is available the function will ask for the bay to choose from the available 8 bays of car. If the bay is available the function will assign the car to the customer, if not available the appropriate responses will be displayed.

**Assign:** This function will assign the car from the available bays in the specific brand and display the message calculating the available bays in the fleet.

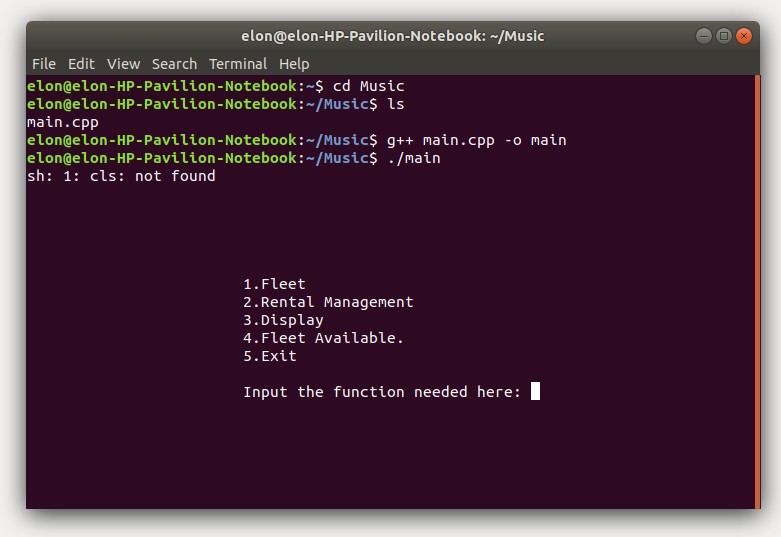
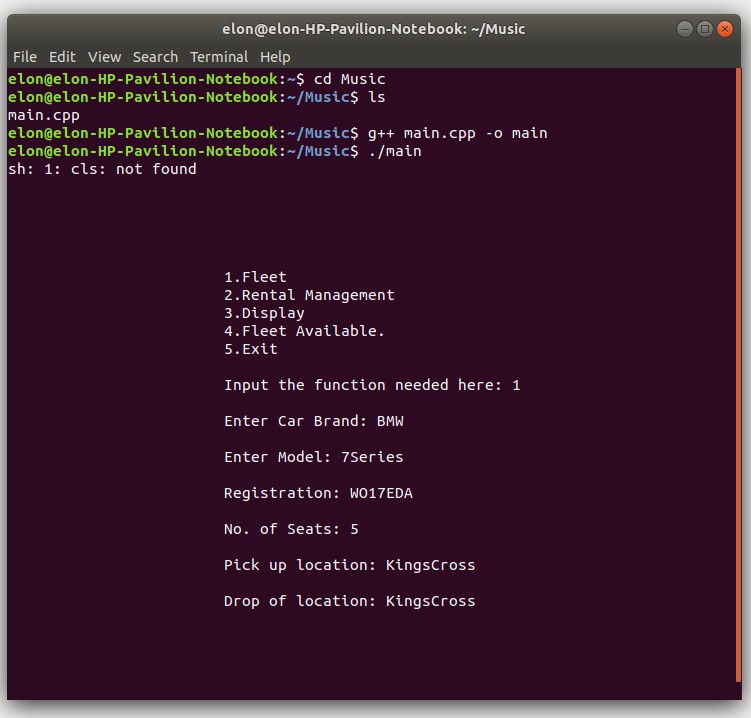
**Display**: This function will display the rented car in the console. The function will first ask for the brand name to check the availability of the cars in the specific brand, after entering the brand name the function will display complete detailed info of the rented car.

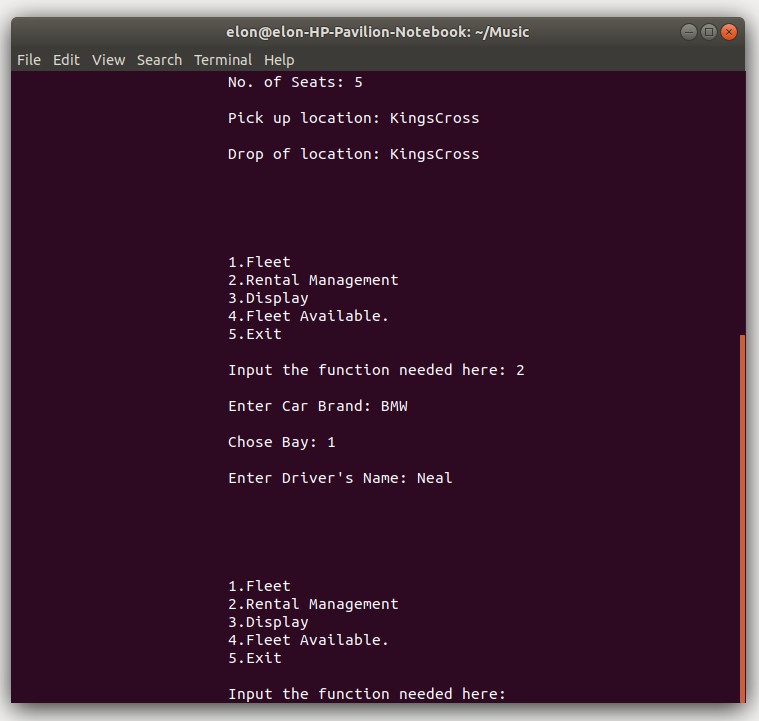
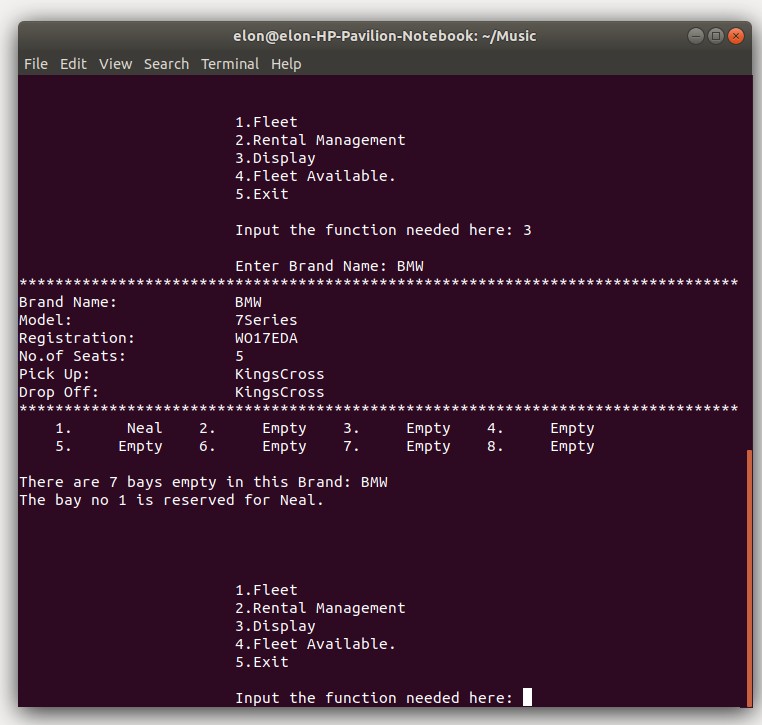
**Avail (Availability):** This function will display all the available fleet. When user creates fleet with the create fleet the fleet gets available to rent, this function will then check the rented fleet and the available fleet and display the available fleet in tabular form.

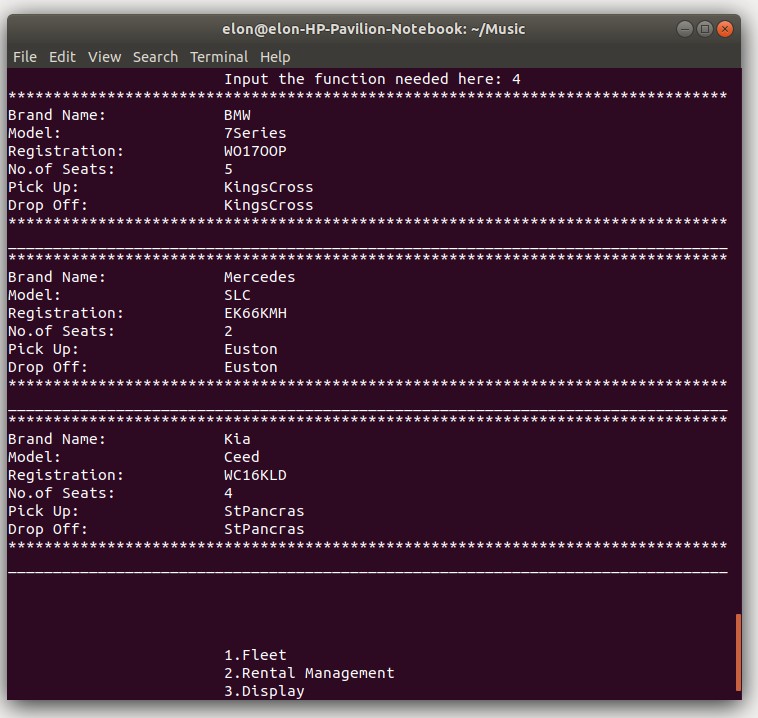
# **Limitations**

There are a few limitations with this system, I believe the use of mysql or .dat would have enabled me to add one more functionality where a driver profile can be created, to store all the personal details and access them when needed.

# **Testing**

I testes this program personally and asked 2 of my friends to try it and the results are posted below. The pictures below are minimized to save the space, can be right clicked and download to view.





---------------- Thank You --------------