

**Activity Report**

**On**

**Car Rental System**

***(Object Oriented Programming)***

**Submitted by**

**Sadhana Das (1930037)**

**B.Tech Programme in Electronics and Computer Science Engineering**

**School of Electronics Engineering**

**Kalinga Institute of Industrial Technology, Deemed to be University**

**Bhubaneswar, India**

CONTENTS

1. Acknowledgment
2. Problem Definition and Proposed Solution
3. Identify Function & Users of the Application
4. Identify Classes, data members and member functions
5. Class diagrams with relationships
6. The files required with attributes
7. Implementation
8. Screen Shots of the Output and the files used
9. Test Cases
10. Troubleshooting Common Problems
11. Limitations
12. Future Scope
13. Conclusion
14. Reference

ACKNOWLEDGEMENT

I sincerely thank all the team members who participated in the case study, thanks to their efforts for making the project-report a success. I also thank the teachers who have provided us the opportunity to present our skills in the report.

Thanks to Ravi Prakash Mishra, 1930032, for writing the report and making the class diagram along with the inheritance chart and ppt for presentation.

Thanks to Arunavo Dutta, 1930071, for doing the verification and needed correction in the code and report.

Thanks to Sampat Parimanik.1930040, for the researching and collecting information for the project.

Thanks to Sadhana Das, 1930037 for writing the code for the project and adding the outputs and file used in the project.

Hope the report has cleared all the required strata.

Yours Sincerely

Sadhana Das (1930037)

Problem Definition and Proposed Solution

A new up and coming car rental service is wishing to have a user interface that will allow their customers to view the models, descriptions and prices of different cars available. The user has the ability to register and log in to the system and track their rental plan. The program code will be responsive, allowing for the customer to view it. The administrator will also be able to login through the same form but have the ability to add/remove new car rentals, change prices, and so on.

Car Rental System is based on a concept to rent cars and generate rental invoice of a rental company. Before stepping into the main system a user has to pass through a login system to get access, then only the user can select cars with a different model and rent for certain days.

Talking about the features of the Car Rental System, after logging in as a user he/she has to provide a name then the user can select available cars. After selecting a car, the system displays selected car details which contain maximum power, mileage and many more. Then the user has to provide the information such as Car number and number of days to rent the car. After all these procedures, the system calculates rent and displays Customer Invoice presenting invoice number, customer’s name, car model, number, number of days and total rental amount.

Identify Function & Users of the Application

1. void dispAvailCar();
2. int countAvail();
3. int rate(int, int);
4. int carCount();

Belongs to class “car”

1. void availCar();
2. void displayCar();
3. void carData();
4. void customerData();
5. int custCount();

Belongs to class “customer”

1. void existingCust();
2. void customerMenu();
3. void newCarData();
4. void delCar();

Belongs to class “Admin”

1. void showCarData();
2. void adminMenu();
3. void welcome();
4. void tNc();

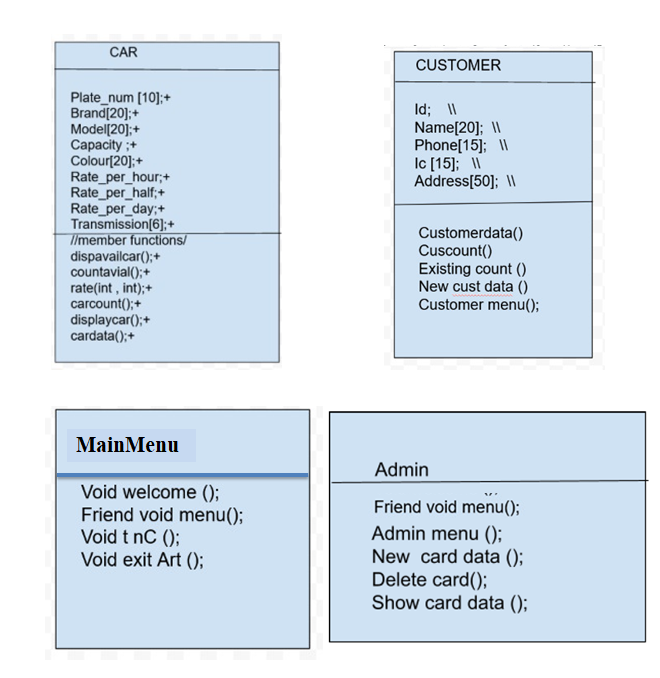
Belongs to class “MainMenu”

1. void exitArt();
2. void menu();

Friend function to class “customer”, “Admin” and ”MainMenu”

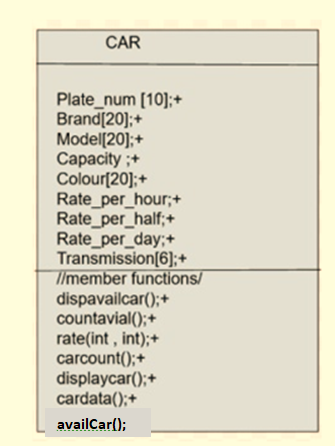
1. Void dispAvailCar() : the car rental file is opened, where the details of the car are copied to another temporary file ” available.txt” here the customer can read and select the car needed
2. Int countAvail() : the function is supposed to count the touple number and the price of the car renting calculation of the rented car is acquired present in “available.txt” file
3. Int rate(int, int) : rental price Is classified as 1 hour, 12 hour and 24 hour basis, hour required is given by customer, hence the car chosen is rented after the calculation is done
4. Int CarCount() : while reading the ‘car rental.txt’ file, the string word checks foe an empty row, until then ‘No. of car’ is incremented, after we reach the end of file, the counting stops and file is closed
5. Void availCar() : it basically reads the “available.txt” file , so no abnormality is caused during the car renting process
6. Void displayCar() : function is to display the car data from “car rental.txt”
7. Void carData() : it reads the “car rental.txt “ file
8. Void customerData() : the customer.txt file is opened and customer details like name, ic, address is stored
9. Int custCount() : counts the number of rows in customer.txt
10. Void existingCust(): function allows to enter the id and then procced to car selection and then to payment
11. CustomerMenu(): the menu portal for customer
12. Void newCarData() : allows admin to add new car details and the result is stored in carrental.txt
13. Void delCar () : allows to delete the car details from its plate number, a temp.txt is created where the details from car rental.txt file are copied except the car whose details are to deleted from record, the temp.txt is renamed as car rental.txt
14. Void showCarData(0: the admin function to display the car details
15. Void adminMenu() : the menu portal for admin
16. Void welcome (): to open welcome.txt
17. Void tNc() : to show the terms and conditions added to the project
18. Void exitArt() : to open the exitArt.txt file
19. Void menu(): it shows to enter the next menu portal of customer or admin , otherwise to read the terms and condition or simply log off

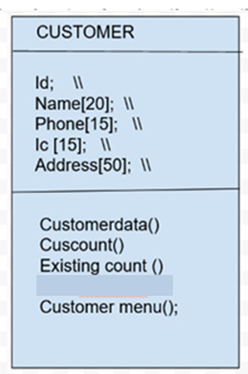
Identify Classes, Data Members and Member Functions

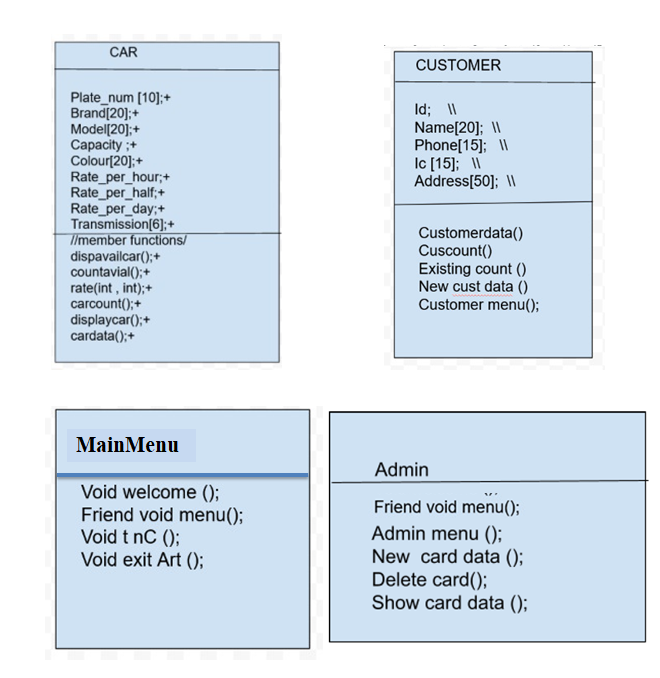


**availCar(); +**

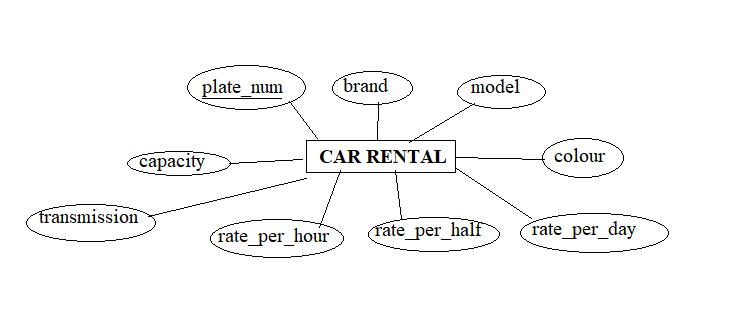
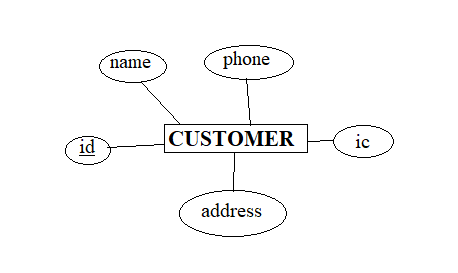
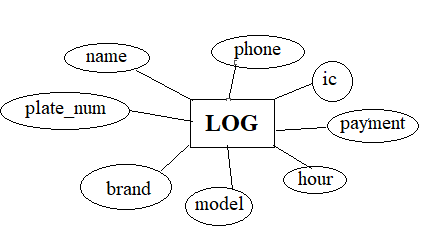
**MainMenu**

Class Diagrams with Relationships

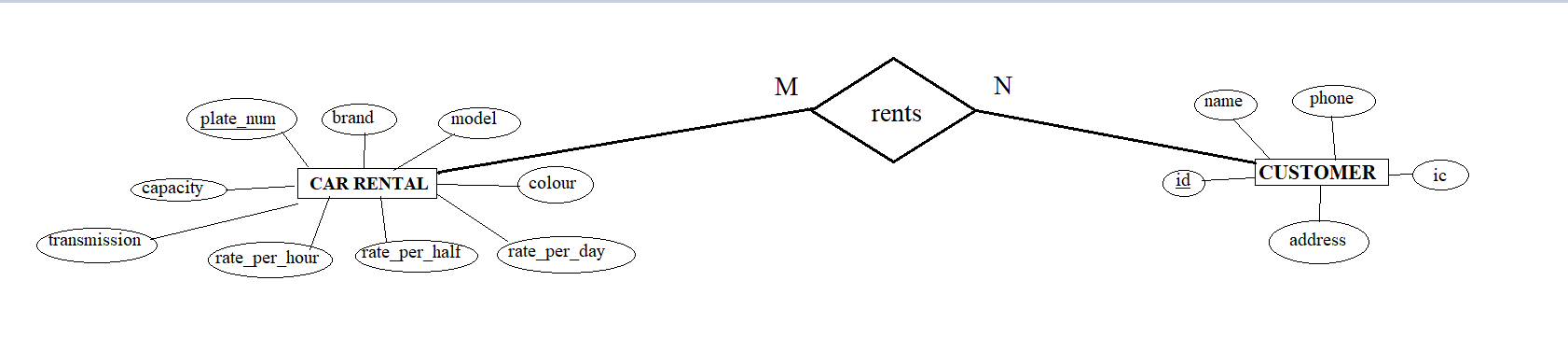




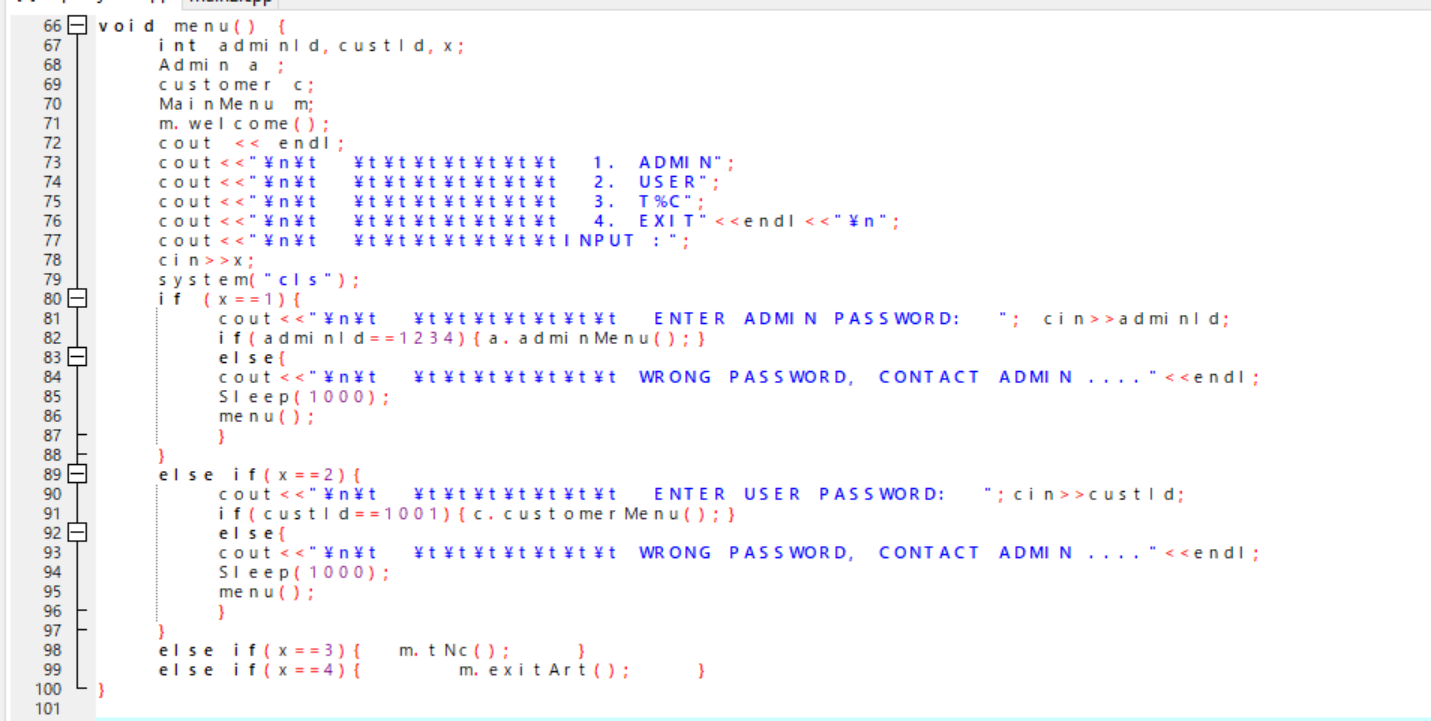
The Files Required with Attributes

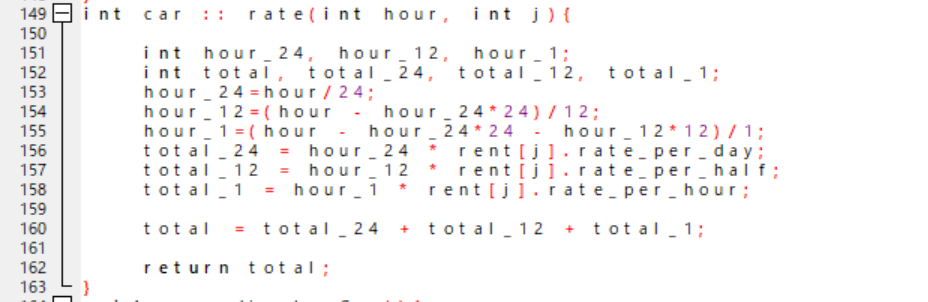
1. **Car Rental .txt**
   * char plate\_num [10];
   * char brand [20};char model [20];
   * float capacity;
   * char colour [20];
   * float rate\_per\_hour;
   * float rate\_per\_half;
   * float rate\_per\_day;
   * char transmission;
2. **Customer.txt**
   * int id;
   * char name [20];
   * char phone [15];
   * char ic [15];
   * char address [50];
3. **log.txt**
   * char name [20];
   * ****char phone [15];
   * char ic [15];
   * char plate\_num [10];
   * char brand [20};
   * char model [20];
   * int payment

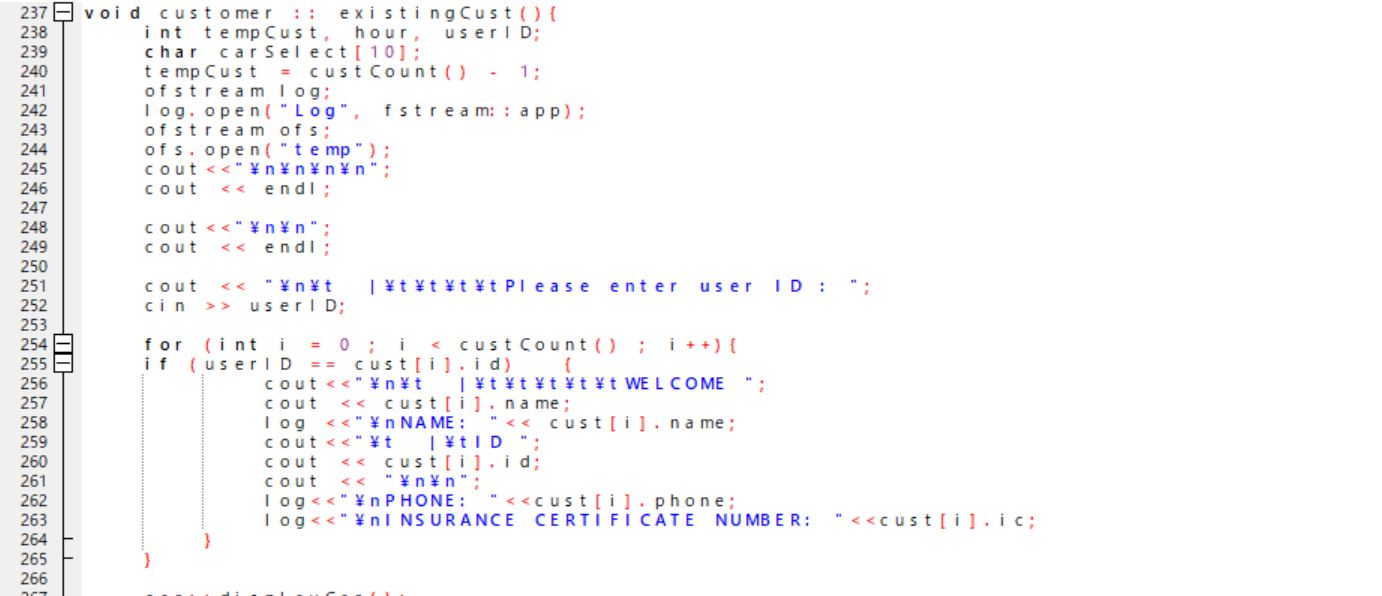
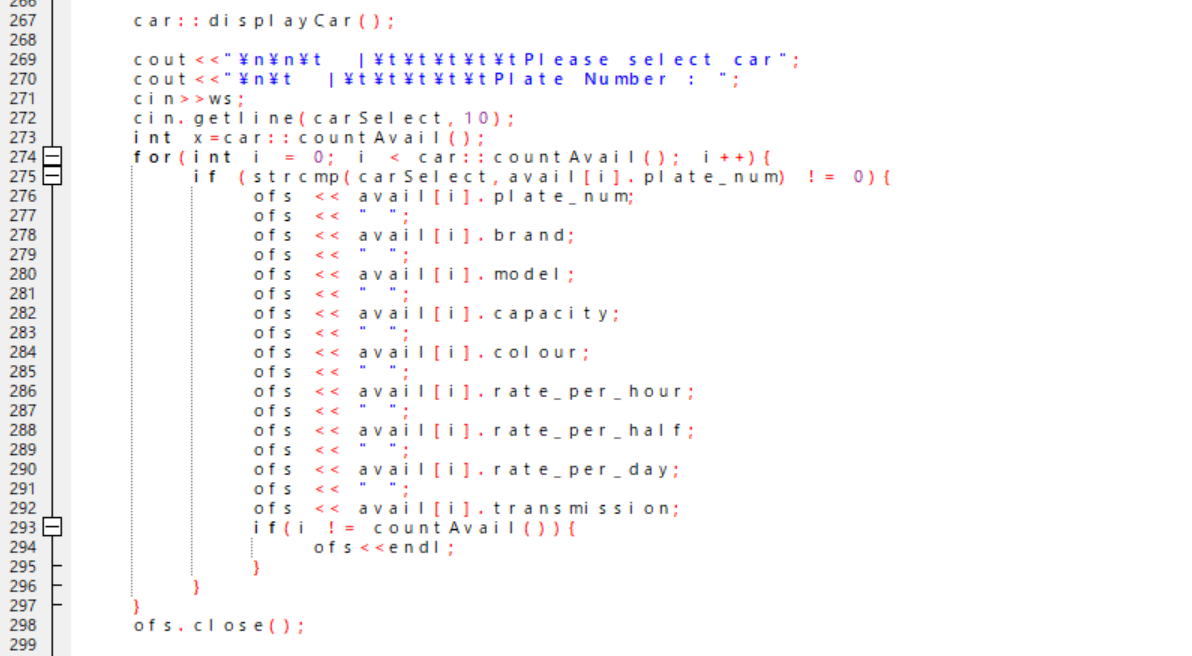
1. **welcome.txt**
2. **exit art.txt**

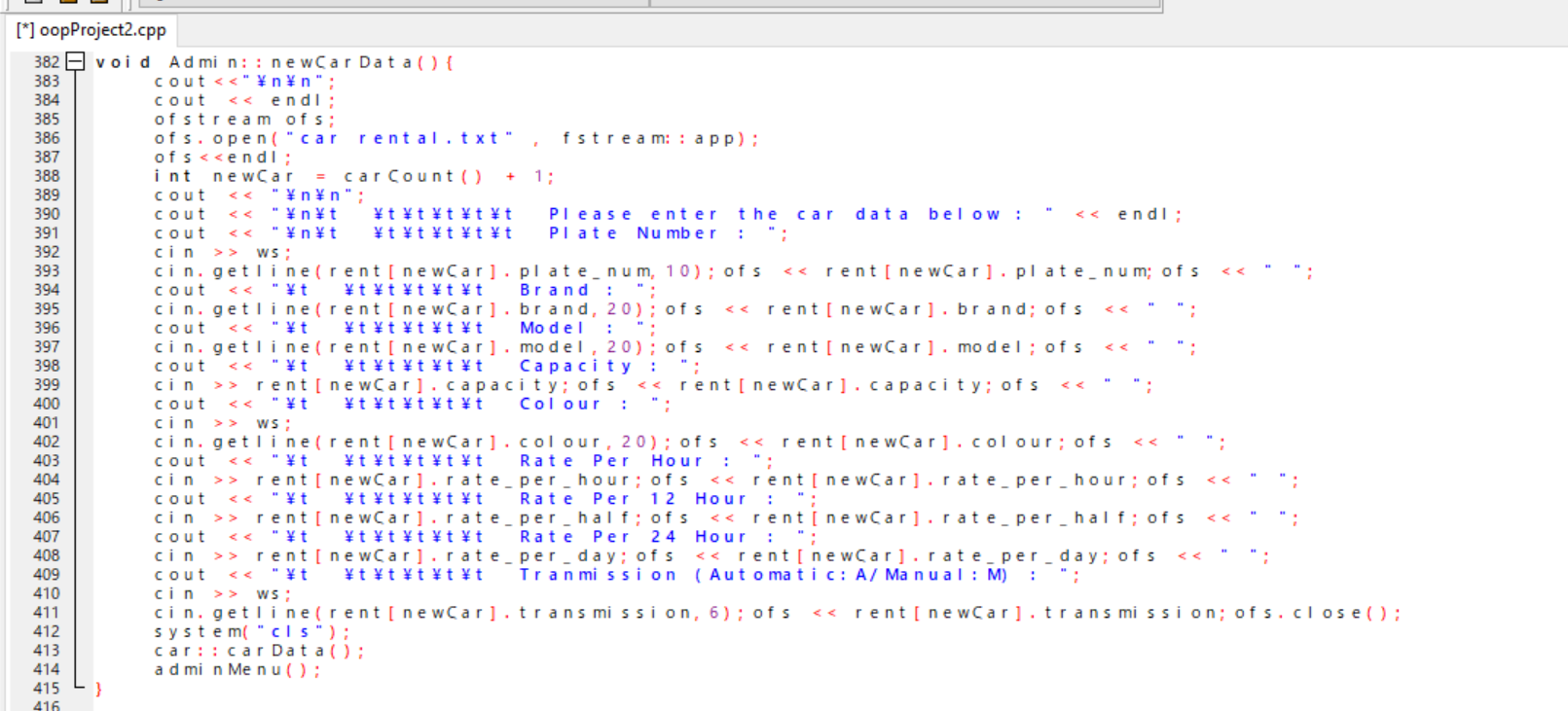
****

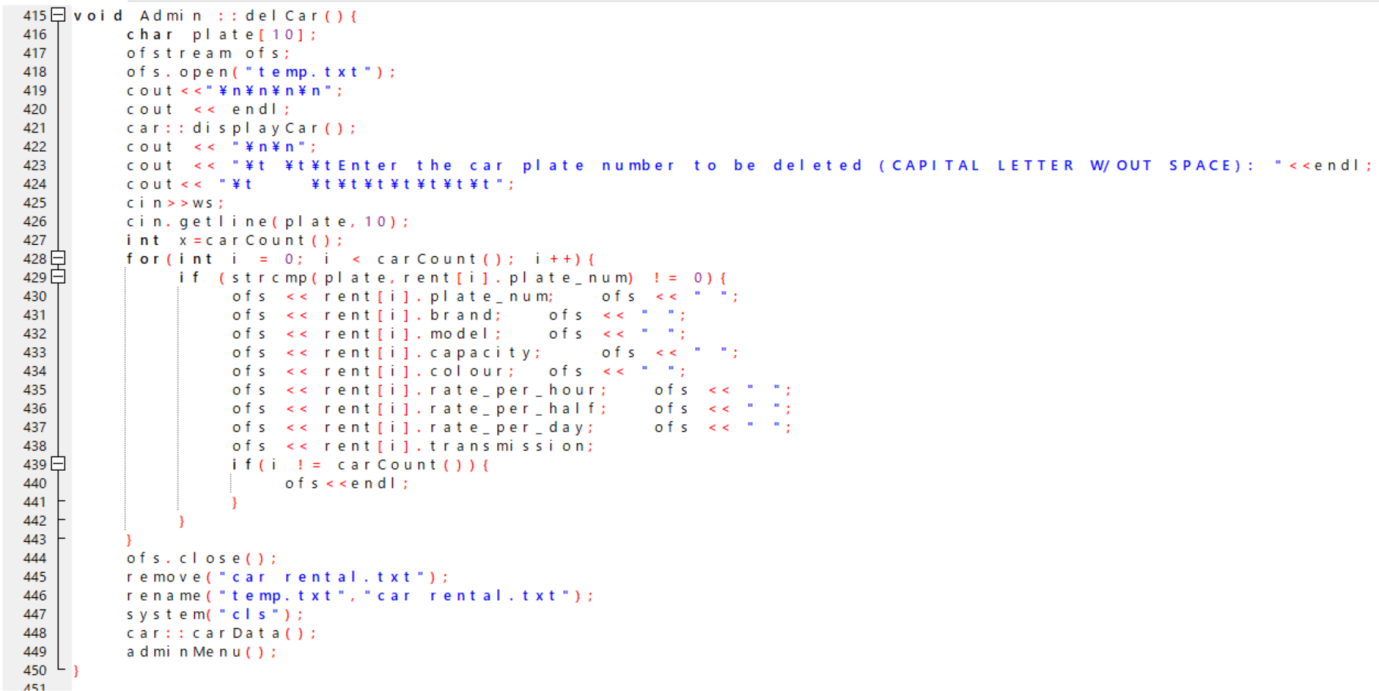
Implementation







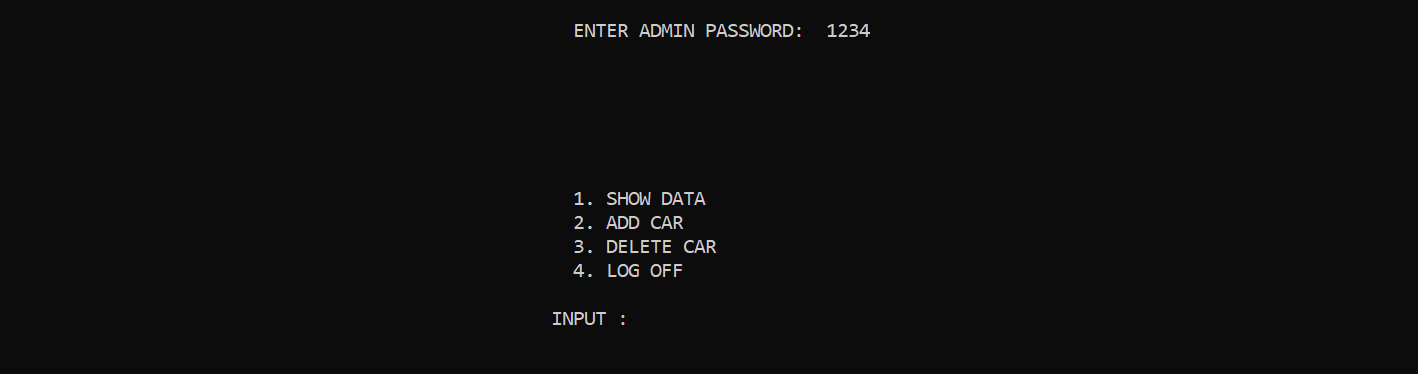




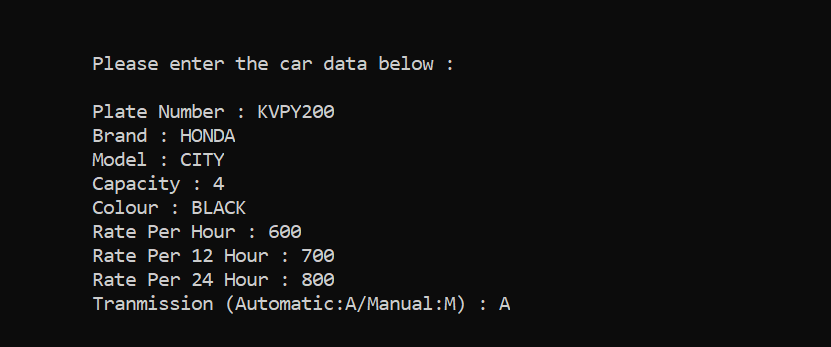
Screen Shots Of The Output And The Files Used



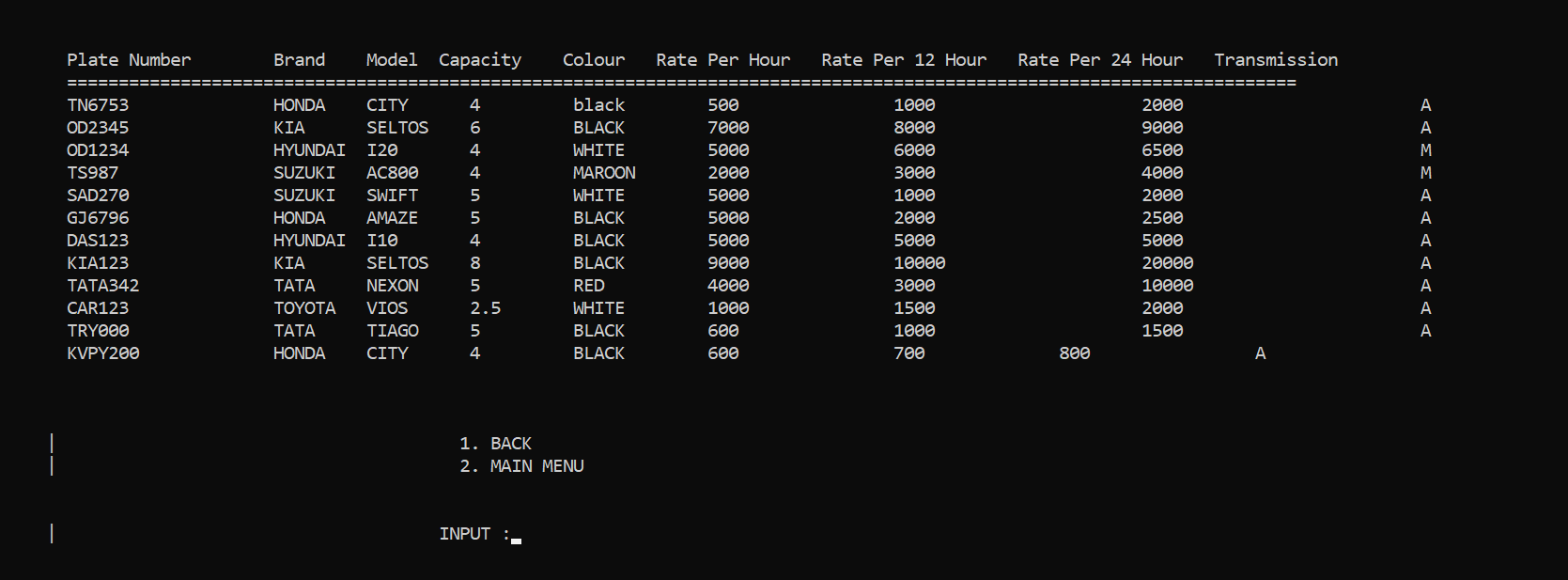
ADMIN MENU:



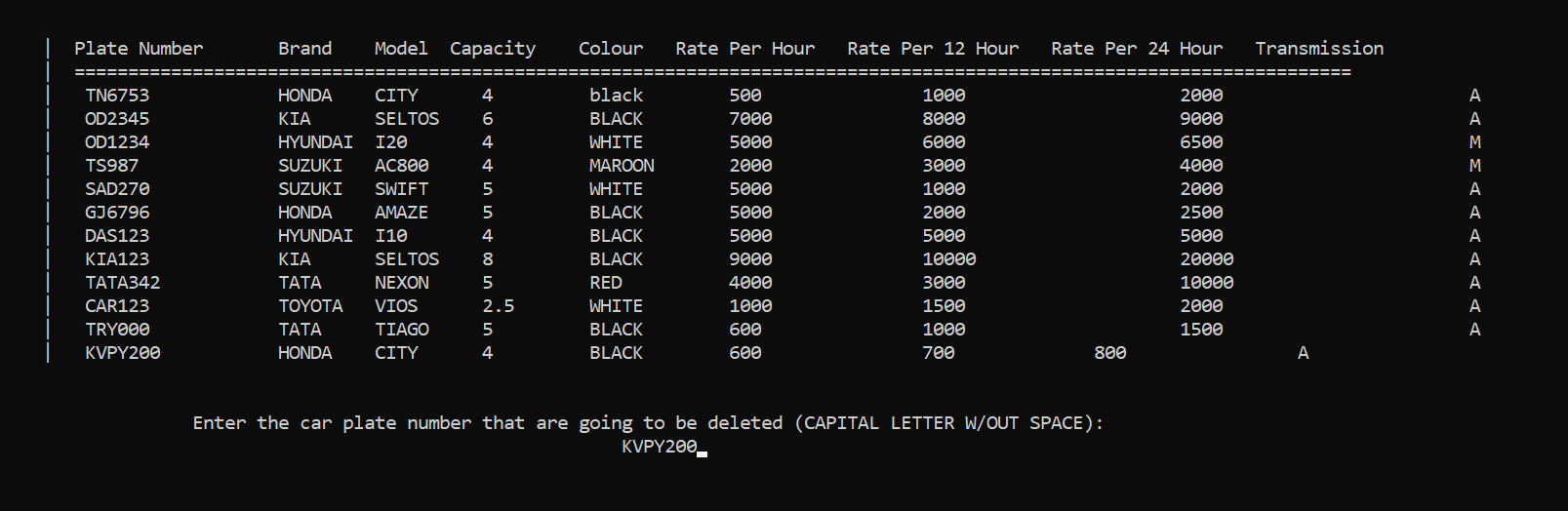
ADMIN >> ADD CAR()



ADMIN>>> SHOW DATA()

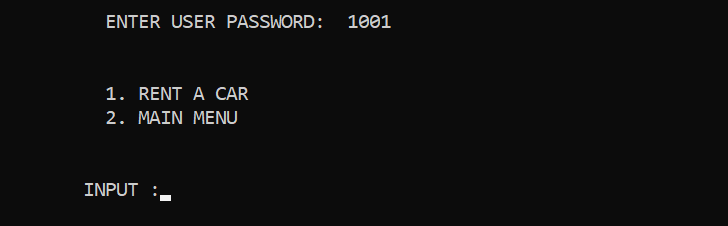


ADMIN >>> DELETE CAR ()

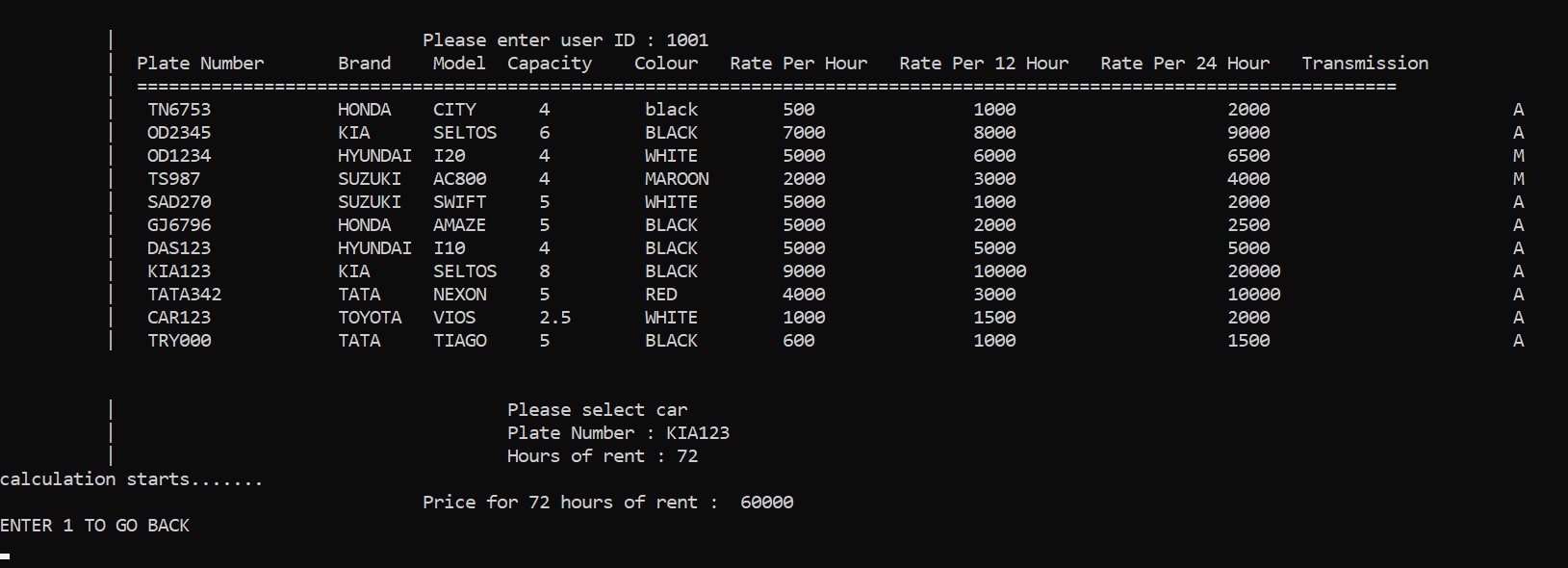


ADMIN >> SHOWDATA () AFTER DELETECAR ()

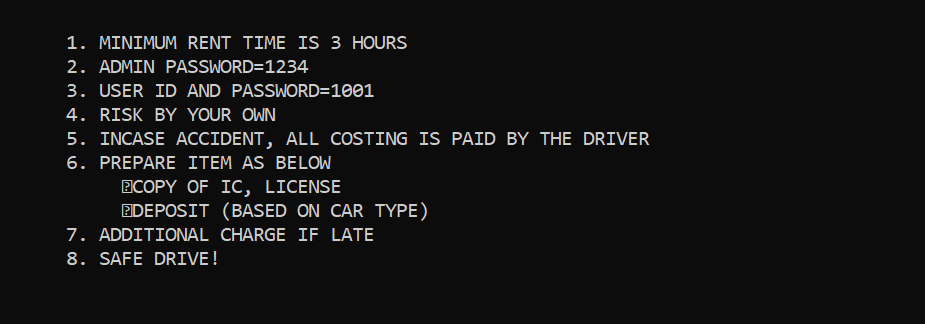
USER/CUSTOMER MENU:

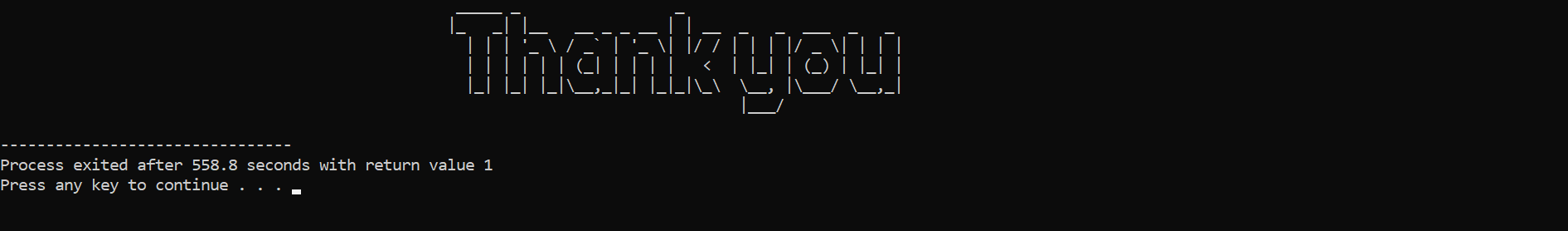


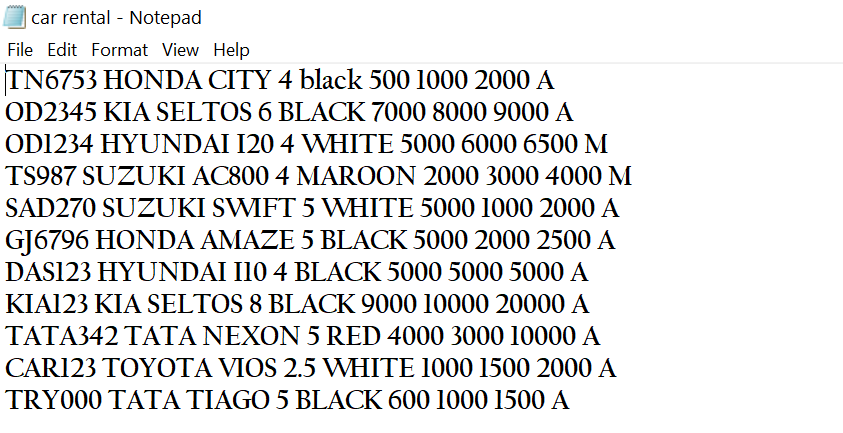
USER >>> RENT A CAR()

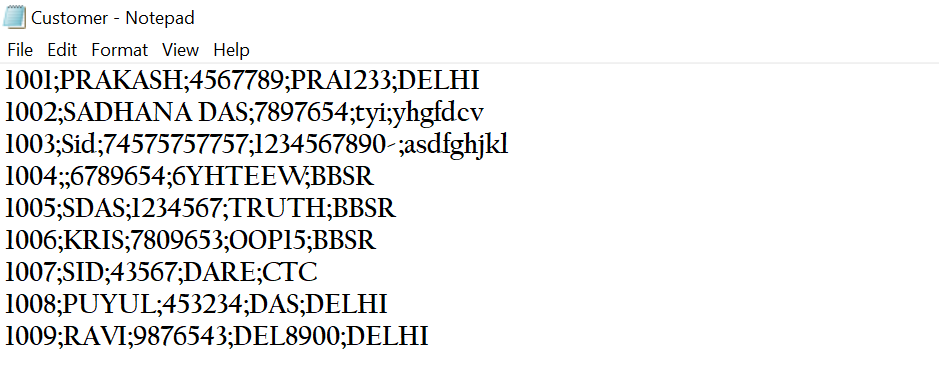


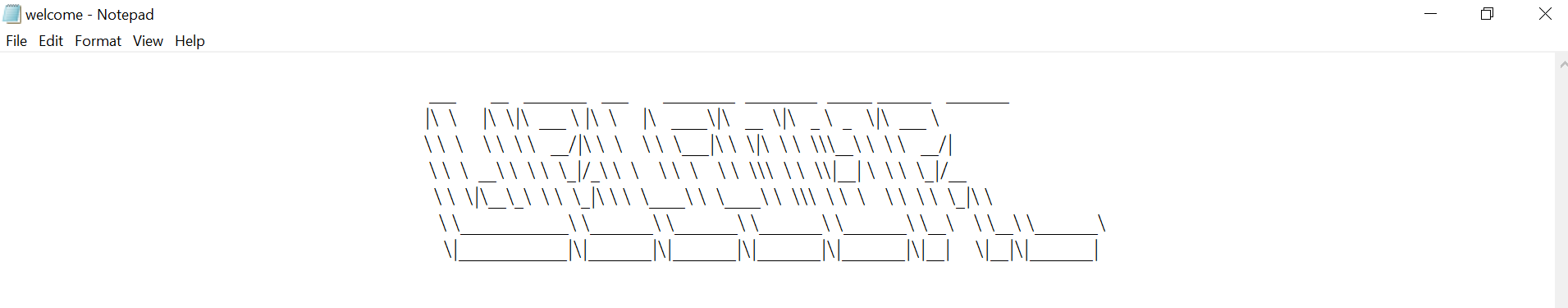
T&C

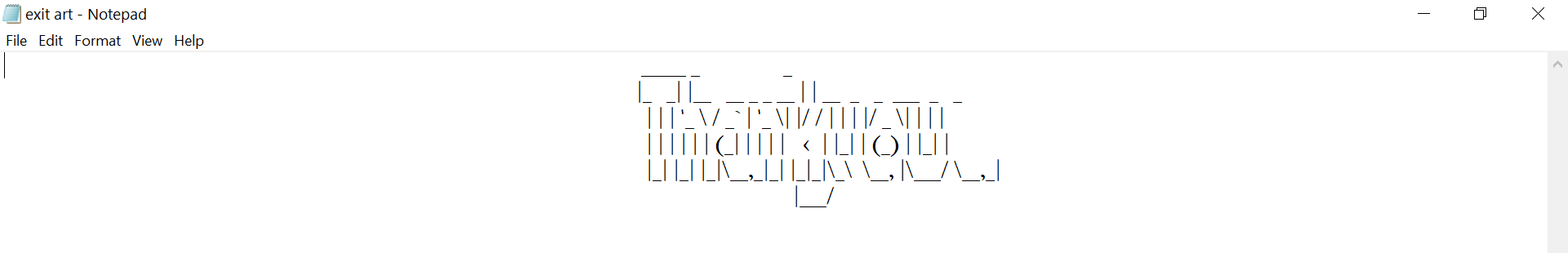


EXIT

**FILES USED**









Test Cases

The following are examples of test cases we implemented:

* Amount values should be displayed with correct calculated results
* The “Welcome” and “Thankyou” graphics to display properly
* Smooth navigations from one screen to another
* The file contents implementation to run smooth
* The password protected system and message display if the input doesn’t match
* The table of the object entered to be displayed properly
* Check all pages for broken images
* Check all pages for broken links

Troubleshooting Common Problems

Problem: Invalid Login

* The credentials used to log in was not found in the database, input the credentials again in case of mistype
* If invalid login persists, pursue the lost password option to obtain a new password. Look for ID input of “customer.txt “ or the “ T&C ” page

Problem: Page Not Found

* Make sure you have entered the text files correctly.
* Make sure the code file and the text file are present in same folder.

Problem: Customer/Vehicle Not Found

* Upon searching through the software, this means that the customer or vehicle is not currently in the database or may have been removed. Review recent changes to the

Database by the company to see if any changes had occurred.

Limitations

1. In order to perform the rent a car capability;-Customers must login to their own profiles.Guests must create their own accounts which in the hand of admin.

2. Unless the payment phase is completed, the system will not allow to perform the booking and renting capabilities

Future Scope

Car Rental Business in the whole world growing rapidly. Now rental companies are focusing on customer support, training, sales and marketing to stand out clearly and to communicate better with the customers. This will increase the customer footfalls and sales.

* Easy Access to buyers & Chart and Graph analysis of Car
* Multi User Account System
* Free Listings and inquiries
* Zero loss of customers
* Invoice Management
* Monitoring the whole Car Management System
* Responsive User Interface
* Management of all type of users’ account
* Car report management

Conclusion

Hence, the Car Rental System is properly studied and implemented. The outputs and files used are attached to the report.

Reference

1. [https://youtu.be/MPJT9xwIBUI](https://youtu.be/MPJT9xwIBUI%20)
2. [https://code-projects.org/car-rental-system-in-c-with-source-code/](https://code-projects.org/car-rental-system-in-c-with-source-code/%20)
3. <https://www.coursehero.com/file/p1h93uf/14-Scope-and-limitations-of-the-system-The-developer-is-going-to-develop/>
4. OOP with ANSI &Turbo C++, Kamthane, Pearson
5. OOP with C++, E Balagurusamy, 8th Edition, Mc Graw Hill