UECS1104/1144 Object-Oriented Application Development

****

Group Assignment

Name : Cheah Ho Ching 1801494 (Leader), Siew Shun Yao 1905076, Lu Jason 1906449, Teh Jia Rong 1704975, Beh Ze Xin 1701165

Course : SE

Practical group : Cheah Ho Ching-P4 , Siew Shun Yao-P3, Lu Jason-P3, Teh Jia Rong-P4, Beh Ze Xin-P2

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Component** | **Missing** | **Poor** | **Average** | **Good** | **Excellent** | **Max Marks** | **Marks Obtained** |
| **Part 2** | System Requirements Specification |  |  |  |  |  | **10** |  |
| **Part 3** | Class Diagram (Notation used, relationship, multiplicities) |  |  |  |  |  | **20** |  |
| **Part 4** | Implementation/ Functionalities |  |  |  |  |  | **40** |  |
| File Handling |  |  |  |  |  | **5** |  |
| Exception Handling |  |  |  |  |  | **5** |  |
| Sample of Input data |  |  |  |  |  | **5** |  |
| Sample Output (Test Cases) |  |  |  |  |  | **5** |  |
| Presentation / General Effort |  |  |  |  |  | **10** |  |
| **Total** | | | | | | | **100** |  |
| Remarks (for Examiner only): | | | | | | | | |

**System requirements:**

**Assumptions and Dependencies**

1. Assume a car model can only be owner by a manager at a time, while a manager can own many car models.

2. Assume the manager refer to car list and find a car by number plate car.

3. Assume the number plate car is unique.

4. Assume user can always go back menu by the menu button on right top side.

5. Assume user after enter and value and press enter, user need to delete the number manual and type in a new number on the text box.

**System Features and Requirements**

**Class Functional Requirements**

1. Car class should display the Plate number, model, cost and status rented or unrented for a car from file.

2.Add class should able to add a car information such as plate number for the car, company name of the car, the name or model of the car, the cost of the car, the first name and last name of manager of key in the information of the car.

3. Rent class should able to let users to find a car by plat number to rent. User need to enter 1 to conform their decision. User need to enter their information such as first name, last name and system will updated the status. If user want to return a car, just need enter car plat number and press 1 to conform.

4. Search class should able to search a car from CarData.txt though the plate number for a car.

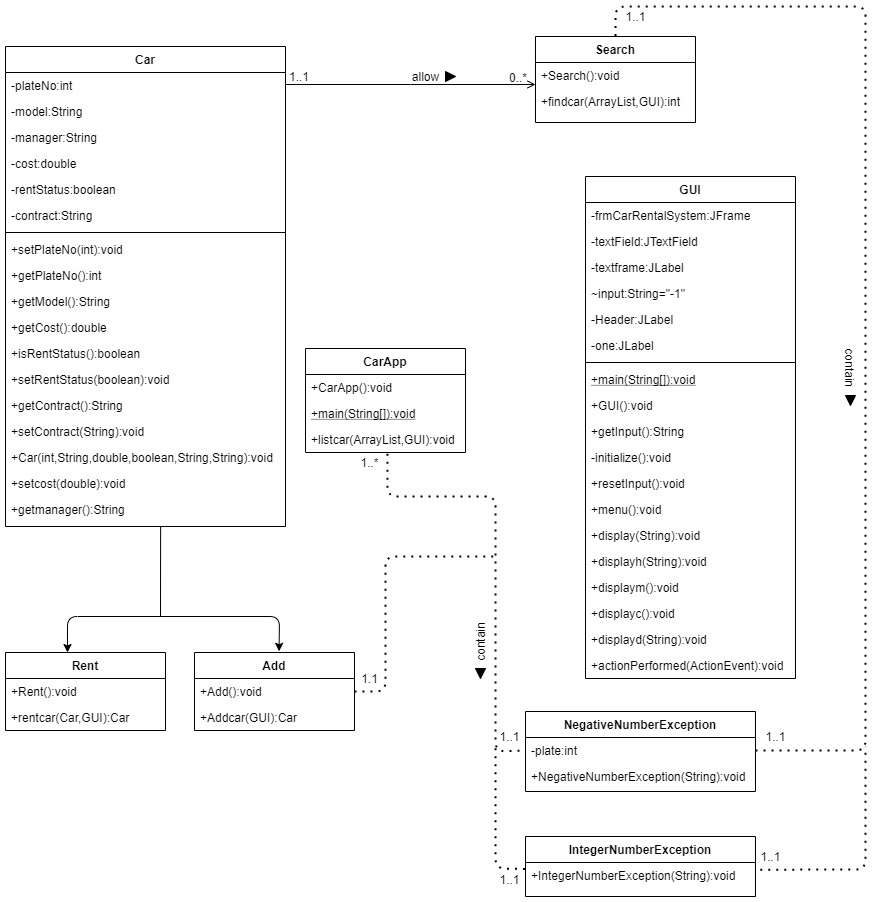
5. CarApp class should able to search plate number of the car to delete it and conform by 1. Next, CarApp should able to save the update text file when user enter 6.

6. Class NegativeNumberException should able to notice user when user key in a negative value and notice user retype again.

7. Class IntegerNumberException should able to notice user when user key in a string data type on integer data type column.

**UML Class Diagram**

**UML Tool: draw io website**



|  |  |
| --- | --- |
| Tools | description |
|  | Association- To use the arrow to point the relationship between one class to another, it is used for bi-directional. From UML class diagram above, One Car is allow to search 0 to many times, One Car will show one list car in CarApp. |
|  | The class text. |
|  | The arrow to point to the next diagram. |

**Source Code:**

***Car class:***

**import java.util.ArrayList;**

**public class Car {**

**private int plateNo;**

**private String model;**

**private String manager;**

**private double cost;**

**private boolean rentStatus = false;**

**private String contract = "unrenter";**

**public void setplateno(int plate) {**

**if(plate<1000 || plate>9999) {**

**throw new NegativeNumberException("Negative plate is not allow");**

**}**

**else plateNo= plate;**

**}**

**public void setcost(double acost) {**

**if(acost<0) {**

**throw new NegativeNumberException("Negative cost is not allow");**

**}**

**else cost = acost;**

**}**

**public int getPlateNo() {**

**return plateNo;**

**}**

**public String getModel() {**

**return model;**

**}**

**public String getmanager() {**

**return manager;**

**}**

**public double getCost() {**

**return cost;**

**}**

**public boolean isRentStatus() {**

**return rentStatus;**

**}**

**public void setRentStatus(boolean rentStatus) {**

**this.rentStatus = rentStatus;**

**}**

**public String getContract() {**

**return contract;**

**}**

**public void setContract(String contract) {**

**this.contract = contract;**

**}**

**public Car(int no, String mod, double co, boolean stats, String rent,String m)**

**{**

**setplateno(no);**

**model = mod;**

**setcost(co);**

**rentStatus = stats;**

**contract = rent;**

**manager =m;**

**}**

**}**

***Add class:***

**public class Add {**

**public Car Addcar(GUI gui) {**

**gui.resetInput();**

**Car add= null;**

**try {**

**gui.display("Please enter the plate no of the car :");**

**while(gui.getInput().equalsIgnoreCase("-1"))**

**{**

**gui.getInput();**

**}**

**int plate = Integer.parseInt(gui.getInput());**

**gui.resetInput();**

**gui.display("Please enter the company of the car :");**

**while(gui.getInput().equalsIgnoreCase("-1"))**

**{**

**gui.getInput();**

**}**

**String m1 = gui.getInput();**

**gui.resetInput();**

**gui.display("Please enter the name of the car :");**

**while(gui.getInput().equalsIgnoreCase("-1"))**

**{**

**gui.getInput();**

**}**

**String m2 = gui.getInput();**

**gui.resetInput();**

**gui.display("Please enter the cost of the car :");**

**while(gui.getInput().equalsIgnoreCase("-1"))**

**{**

**gui.getInput();**

**}**

**double cost = Double.parseDouble(gui.getInput());**

**gui.resetInput();**

**gui.display("First name of manager of this car:");**

**while(gui.getInput().equalsIgnoreCase("-1"))**

**{**

**gui.getInput();**

**}**

**String n1= gui.getInput();**

**gui.resetInput();**

**gui.display("Last name of manager of this car:");**

**while(gui.getInput().equalsIgnoreCase("-1"))**

**{**

**gui.getInput();**

**}**

**String n2 = gui.getInput();**

**gui.resetInput();**

**String x = m1 +" "+ m2;**

**String y = n1 +" "+ n2;**

**add = new Car(plate,x,cost,false,"unrented",y);**

**gui.displayh("Car added!");**

**gui.displaym();**

**}catch(NegativeNumberException ex) {**

**gui.display(ex.getMessage());**

**}**

**return add;**

**}**

**}**

***Rent class:***

**public class Rent {**

**public Car rentcar(Car car,GUI gui) {**

**if(car.isRentStatus())**

**{**

**gui.displayh("Is this the car to return? (enter 1 to comfirm)");**

**gui.resetInput();**

**while(gui.getInput().equalsIgnoreCase("-1"))**

**{**

**gui.getInput();**

**}**

**int choice = Integer.parseInt(gui.getInput());**

**if (choice == 1)**

**{**

**car.setRentStatus(false);**

**car.setContract("unrented");**

**gui.display("changes made!");**

**}**

**else**

**gui.display("No changes made!");**

**}**

**else**

**{**

**gui.displayh("Is this the car to rent? (enter 1 to comfirm)");**

**gui.resetInput();**

**while(gui.getInput().equalsIgnoreCase("-1"))**

**{**

**gui.getInput();**

**}**

**int choice = Integer.parseInt(gui.getInput());**

**if (choice == 1)**

**{**

**car.setRentStatus(true);**

**gui.displayh("Please enter the renter of this car");**

**gui.display("First name :");**

**gui.resetInput();**

**while(gui.getInput().equalsIgnoreCase("-1"))**

**{**

**gui.getInput();**

**}**

**String n1= gui.getInput();**

**gui.display("Last name:");**

**gui.resetInput();**

**while(gui.getInput().equalsIgnoreCase("-1"))**

**{**

**gui.getInput();**

**}**

**String n2 = gui.getInput();**

**car.setContract(n1+" "+n2);**

**gui.display("changes made!");**

**}**

**else**

**gui.display("No changes made!");**

**}**

**gui.resetInput();**

**gui.displaym();**

**return car;**

**}**

**}**

***Search class:***

**import java.util.ArrayList;**

**public class Search {**

**public int findcar(ArrayList<Car> car,GUI gui) {**

**int x=1;**

**int comf=-1;**

**do {**

**try {**

**gui.display("Enter the plate no of the car :");**

**gui.resetInput();**

**while(gui.getInput().equalsIgnoreCase("-1"))**

**{**

**gui.getInput();**

**}**

**int num = Integer.parseInt(gui.getInput());**

**gui.resetInput();**

**int count = 0;**

**for( Car a :car) {**

**a.setplateno(num);**

**if(num == a.getPlateNo()) {**

**String showcar = "<html>" + "<br/>PlateNo :" +a.getPlateNo() +"<br/>Model :"+a.getModel()+"<br/>Cost :"+"RM"+a.getCost()+**

**"<br/>Rent :"+a.getContract();**

**gui.display(showcar);**

**gui.displaym();**

**comf=count;**

**}**

**count++;**

**}**

**if(comf == -1)**

**{**

**gui.displayh("No Car found! return to menu>>>");**

**}**

**x=2;**

**}**

**catch(NegativeNumberException ex) {**

**gui.display("Negative plate not allow or Not in the range of the value 1000-9999 is not allow as well");**

**}**

**return comf;**

**}while(x==1);}**

**}**

***CarApp class:***

**import java.io.\*;**

**import java.util.ArrayList;**

**public class CarApp {**

**public static void main(String[] args)throws IOException{**

**ArrayList<Car>car = new ArrayList<Car>();**

**Search finder = new Search(); //Search function**

**Add adder = new Add(); //Add car function**

**Rent renter = new Rent(); //Delete car function**

**GUI gui = new GUI(); //GUI**

**BufferedReader bReader = null; //get car data**

**try {**

**bReader = new BufferedReader(new FileReader("CarData.txt"));**

**String line = null;**

**while((line= bReader.readLine()) !=null)**

**{**

**String [] carinfo = line.split(",");**

**int no = Integer.parseInt(carinfo[0]);**

**String model = carinfo[1];**

**double cos = Double.parseDouble(carinfo[2]);**

**boolean stats =Boolean.parseBoolean(carinfo[3]);**

**String rent = carinfo[4];**

**String name = carinfo[5];**

**car.add(new Car(no,model,cos,stats,rent,name));**

**}**

**} catch (FileNotFoundException e) {**

**System.out.println("CarData.txt not found!");;**

**}**

**bReader.close();**

**int option =0; //main menu**

**int choice;**

**gui.menu();**

**int x=1;**

**do {**

**try{**

**gui.displayd(" ");**

**choice = Integer.parseInt(gui.getInput());**

**switch(choice) {**

**case 1:finder.findcar(car,gui);**

**break;**

**case 2:car.add(adder.Addcar(gui));**

**break;**

**case 3:{**

**int comf = 0; //Confirmation**

**int found = 0; //Placement of the car found**

**while (comf != 1)**

**{**

**gui.displayh("Please search the car to delete!");**

**found = finder.findcar(car,gui);**

**if(found != -1)**

**{**

**gui.displayh("Is this the car to delete? (enter 1 to comfirm)");**

**gui.resetInput();**

**while(gui.getInput().equalsIgnoreCase("-1"))**

**{**

**gui.getInput();**

**}**

**int inc = Integer.parseInt(gui.getInput());**

**if (inc == 1)**

**comf++;**

**}**

**}**

**car.remove(found);**

**gui.displayh("Car deleted! retrun to menu >>>");**

**gui.resetInput();**

**}**

**break;**

**case 4:{**

**int found = -1;**

**Car rent = null;**

**gui.displayh("Please search the car to rent/return!");**

**found = finder.findcar(car,gui);**

**if (found != -1)**

**{**

**rent = renter.rentcar(car.get(found),gui);**

**car.set(found, rent);**

**}**

**else**

**{**

**gui.display("No car found!!");**

**gui.displaym();**

**}**

**gui.resetInput();**

**}**

**break;**

**case 5:listcar(car,gui);**

**break;**

**case 6:{**

**PrintWriter output = new PrintWriter("CarData.txt"); //save array list to text file**

**for( Car a :car) {**

**output.println(a.getPlateNo() + "," + a.getModel() + "," + a.getCost() + "," + a.isRentStatus() + "," + a.getContract() + "," + a.getmanager());**

**}**

**output.close();**

**gui.displayh("changes saved!");**

**};**

**break;**

**case 7:gui.display(gui.getInput());**

**break;**

**}**

**}**

**catch(Exception ex) {**

**gui.display("Invalid input, enter any numbers and click 'OK', then click on 'MENU' to return to page");**

**}**

**}while(option !=1);**

**}**

**public static void listcar(ArrayList<Car> car,GUI gui) {**

**String showcar = "<html>";**

**for( Car a :car) {**

**String newshowcar = "<br/>PlateNo :" +a.getPlateNo() +"<br/>Model :"+a.getModel()+"<br/>Cost :"+"RM"+a.getCost()+"<br/>Rent :"+a.getContract() + "<br/>";**

**showcar = showcar + newshowcar ;**

**}**

**gui.display(showcar);**

**gui.displaym();**

**gui.resetInput();**

**}**

**}**

***GUI:***

**import java.awt.EventQueue;**

**import javax.swing.JButton;**

**import javax.swing.JFrame;**

**import javax.swing.JTextField;**

**import java.awt.Button;**

**import java.awt.event.ActionListener;**

**import java.awt.event.ActionEvent;**

**import javax.swing.JLabel;**

**public class GUI implements ActionListener{**

**private JFrame frmCarRentalSystem;**

**private JTextField textField;**

**private JLabel textframe;**

**String input = "-1";**

**private JLabel Header;**

**private JLabel one;**

**public static void main(String[] args) {**

**EventQueue.invokeLater(new Runnable() {**

**public void run() {**

**try {**

**GUI window = new GUI();**

**window.frmCarRentalSystem.setVisible(true);**

**} catch (Exception e) {**

**e.printStackTrace();**

**}**

**}**

**});**

**}**

**public GUI() {**

**initialize();**

**}**

**private void initialize() {**

**frmCarRentalSystem = new JFrame();**

**frmCarRentalSystem.setTitle("Car Rental System");**

**frmCarRentalSystem.setBounds(100, 100, 624, 738);**

**frmCarRentalSystem.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);**

**frmCarRentalSystem.getContentPane().setLayout(null);**

**Button Menu\_1 = new Button("Menu");**

**Menu\_1.addActionListener(new ActionListener() {**

**public void actionPerformed(ActionEvent e) {**

**menu();**

**displayc();**

**}**

**});**

**Menu\_1.setBounds(524, 10, 74, 27);**

**frmCarRentalSystem.getContentPane().add(Menu\_1);**

**textField = new JTextField();**

**textField.setBounds(10, 660, 507, 31);**

**frmCarRentalSystem.getContentPane().add(textField);**

**textField.setColumns(10);**

**JButton button = new JButton("OK");**

**button.addActionListener(new ActionListener() {**

**public void actionPerformed(ActionEvent e) {**

**input = new String(textField.getText());**

**}**

**});**

**button.setBounds(524, 660, 74, 31);**

**frmCarRentalSystem.getContentPane().add(button);**

**textframe = new JLabel("");**

**textframe.setBounds(0, 46, 598, 604);**

**frmCarRentalSystem.getContentPane().add(textframe);**

**Header = new JLabel("");**

**Header.setBounds(0, 10, 518, 27);**

**frmCarRentalSystem.getContentPane().add(Header);**

**frmCarRentalSystem.getRootPane().setDefaultButton(button);**

**one = new JLabel("");**

**one.setBounds(562, 223, 46, 14);**

**frmCarRentalSystem.getContentPane().add(one);**

**frmCarRentalSystem.setVisible(true);**

**}**

**public String getInput()**

**{**

**return input;**

**}**

**public void resetInput()**

**{**

**input = "-1";**

**}**

**public void menu()**

**{**

**display("<html>1. Find car<br/>2. Add car<br/>3. Delete car<br/>4. Rent/Return car<br/>5. Show car list<br/>"**

**+ "6. Update text file<br/><br/>Please enter your option :<br/><br/>(Update text file before closing the programe!)");**

**}**

**public void display(String string) {**

**textframe.setText(string);**

**}**

**public void displayh(String string) {**

**Header.setText(string);**

**}**

**public void displaym() {**

**Header.setText("return to menu >>>");**

**}**

**public void displayc() {**

**Header.setText(" ");**

**}**

**public void displayd(String string) {**

**one.setText(string);**

**}**

**@Override**

**public void actionPerformed(ActionEvent e) {**

**input = new String(textField.getText());**

**}**

**}**

***IntegerNumberException class:***

**public class IntegerNumberException extends RuntimeException {**

**public IntegerNumberException(String excpmsg) {**

**super(excpmsg);**

**}**

**}**

***NegativeNumberException class:***

**public class NegativeNumberException extends RuntimeException{**

**private int plate;**

**public NegativeNumberException(String exceptionmsg) {**

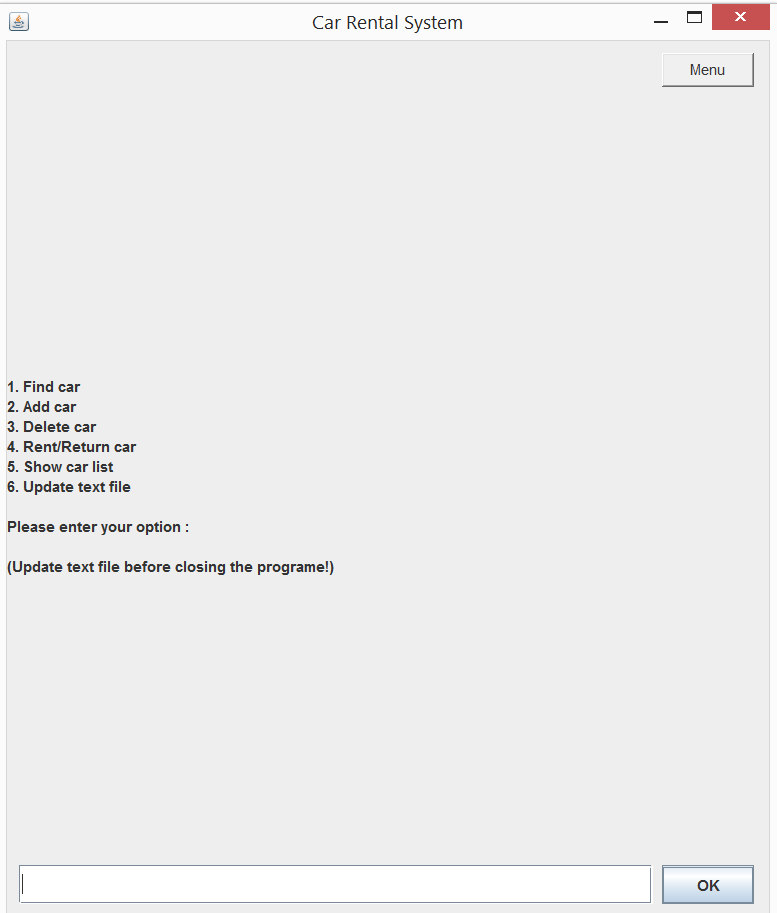
**super(exceptionmsg);**

**}**

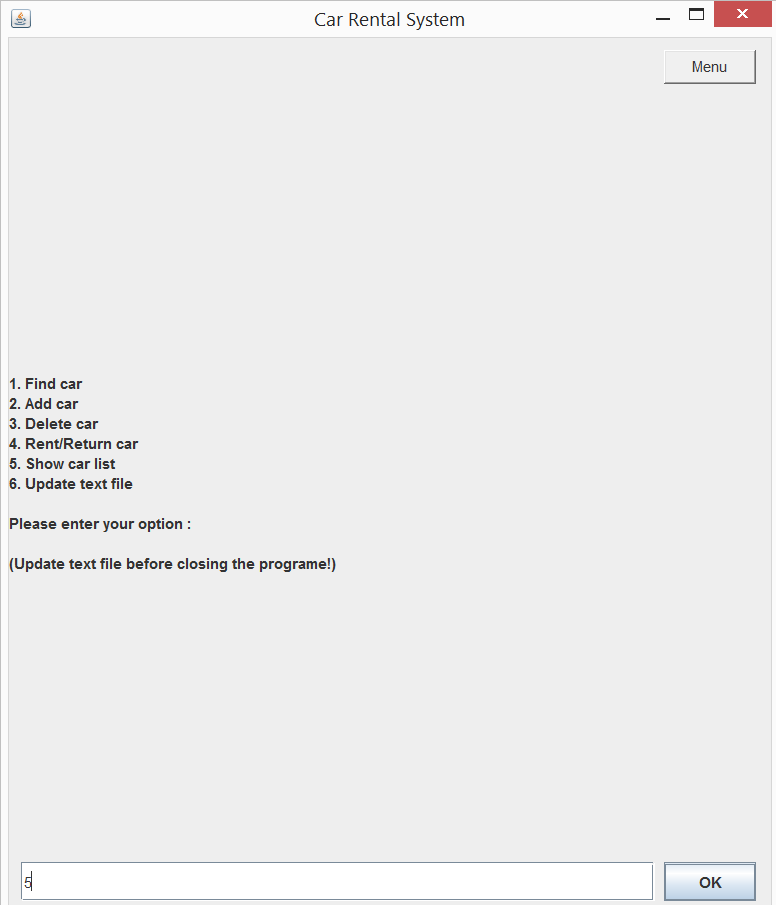
**}**

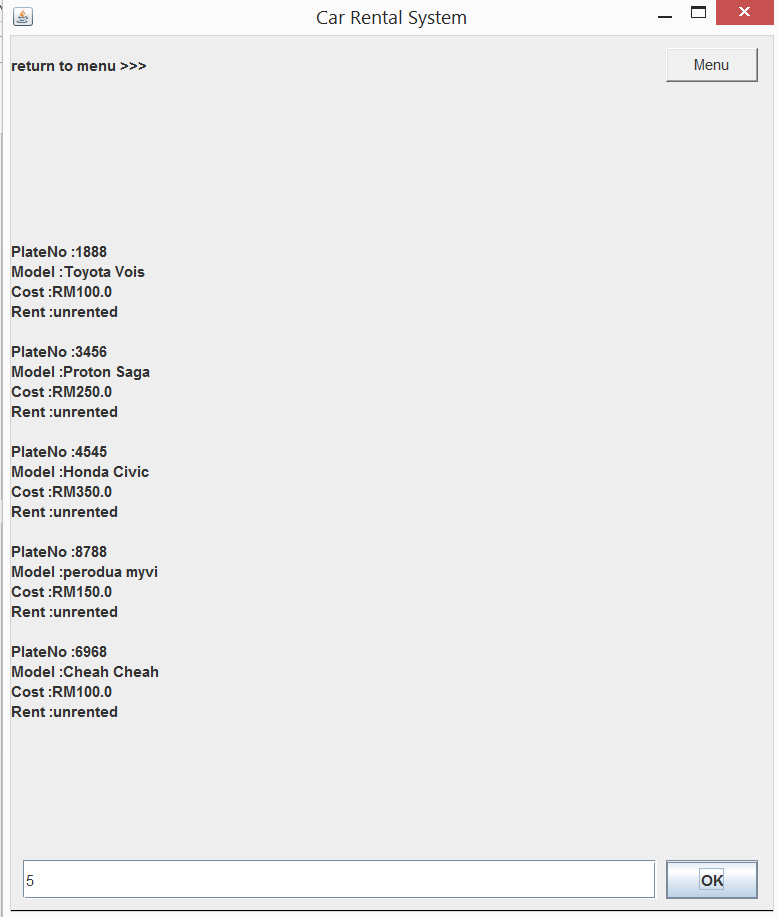
**Sample input:**

Menu:

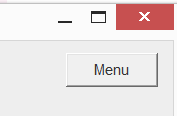


**Select 5 to show car list.**

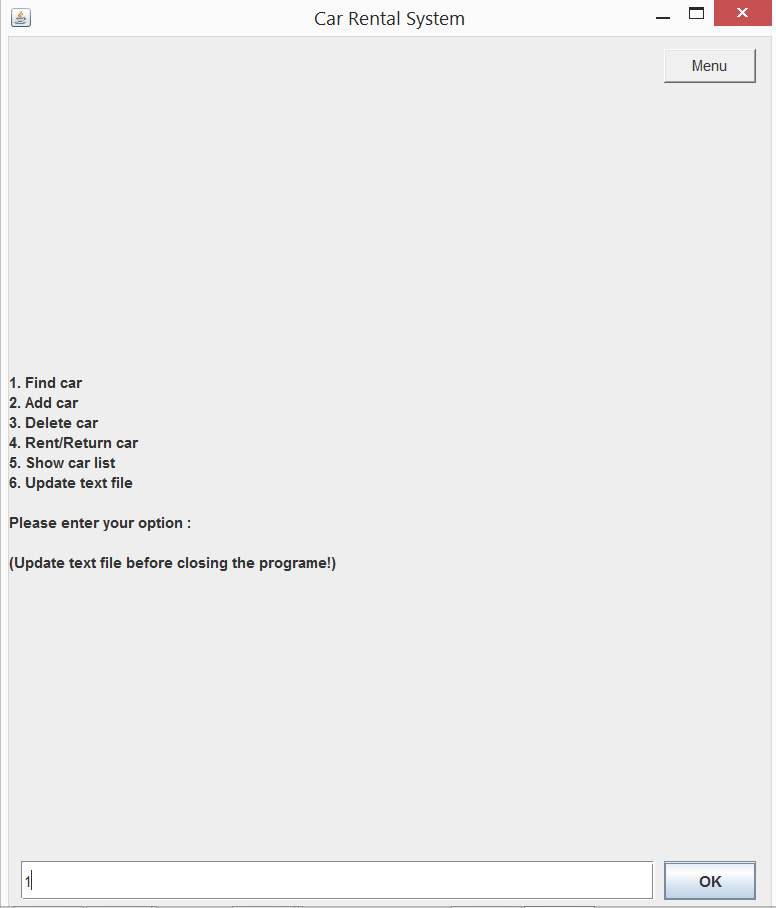




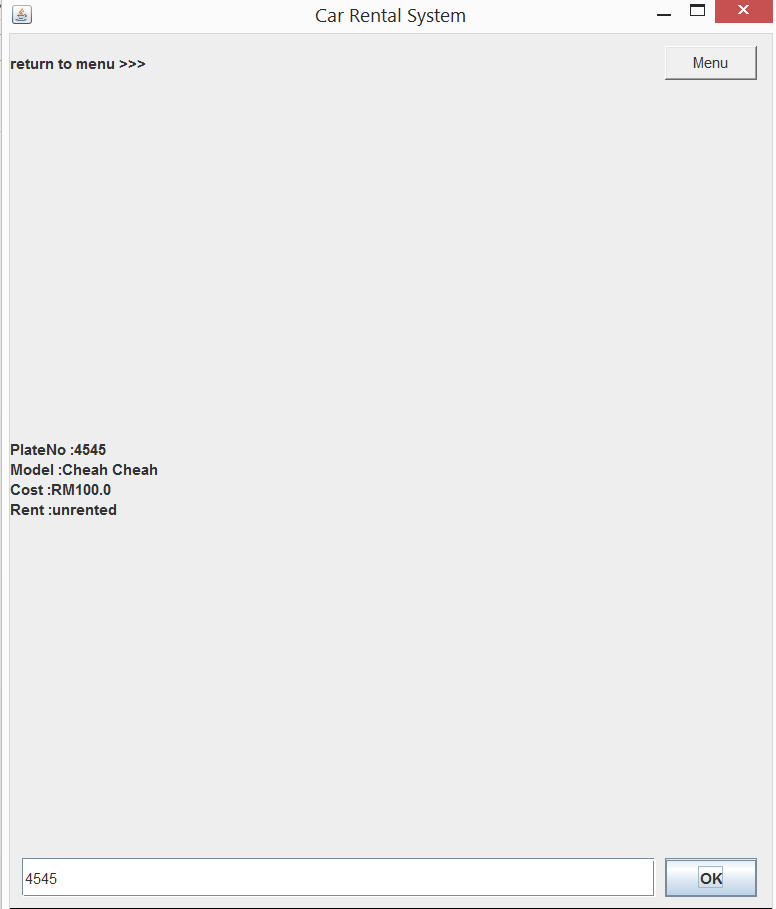
Click the Menu button go back menu.



**Insert 1 to find car**

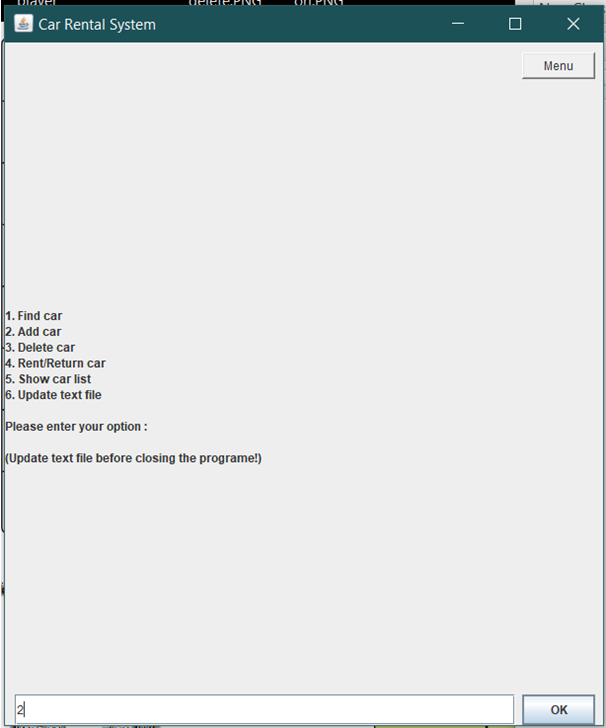


Insert 4545 to find a car

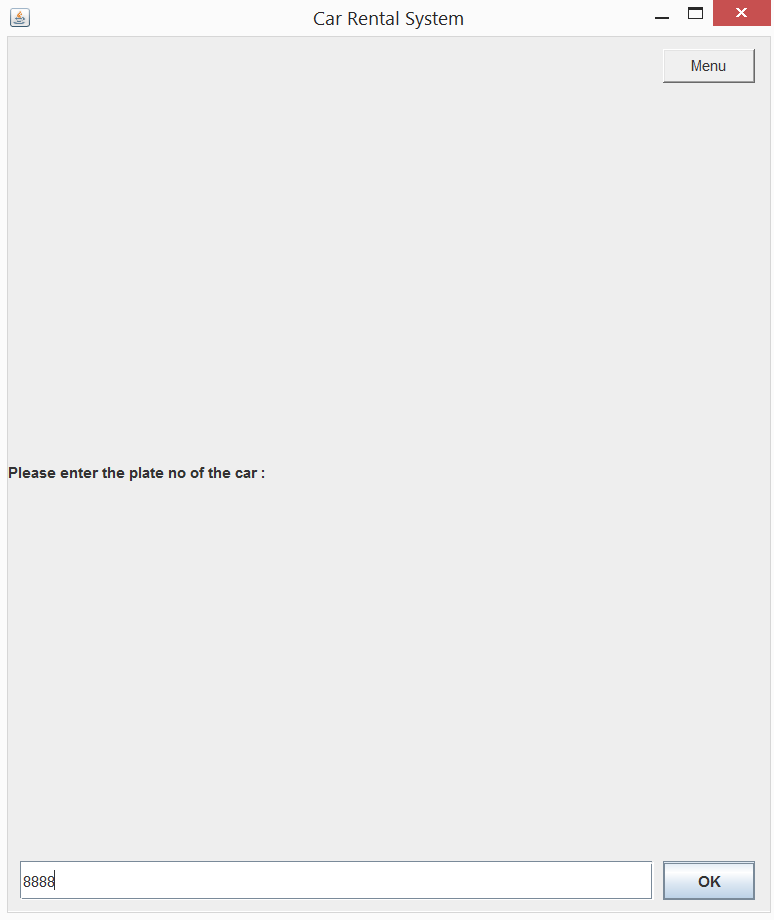


Return to menu by button

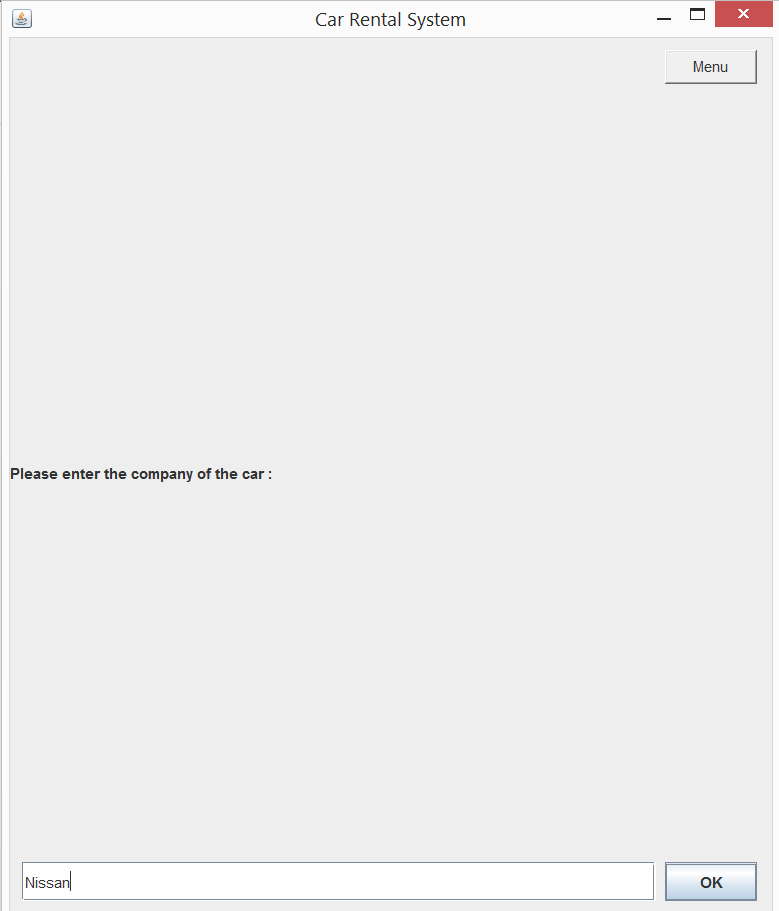
**Insert 2 to Add car**



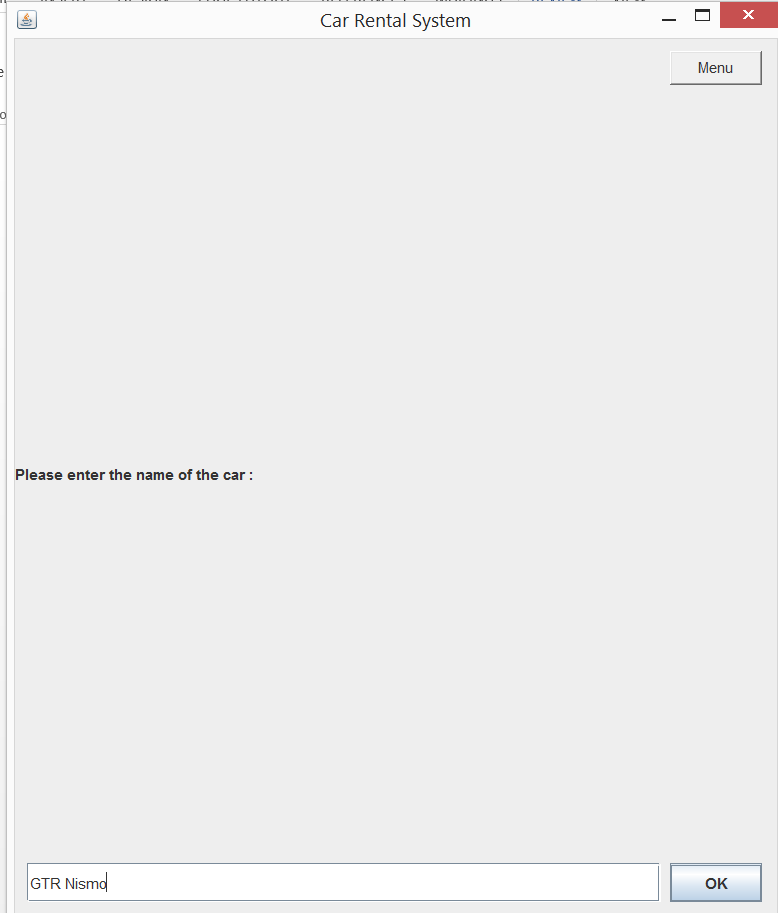
Insert plate no for the car



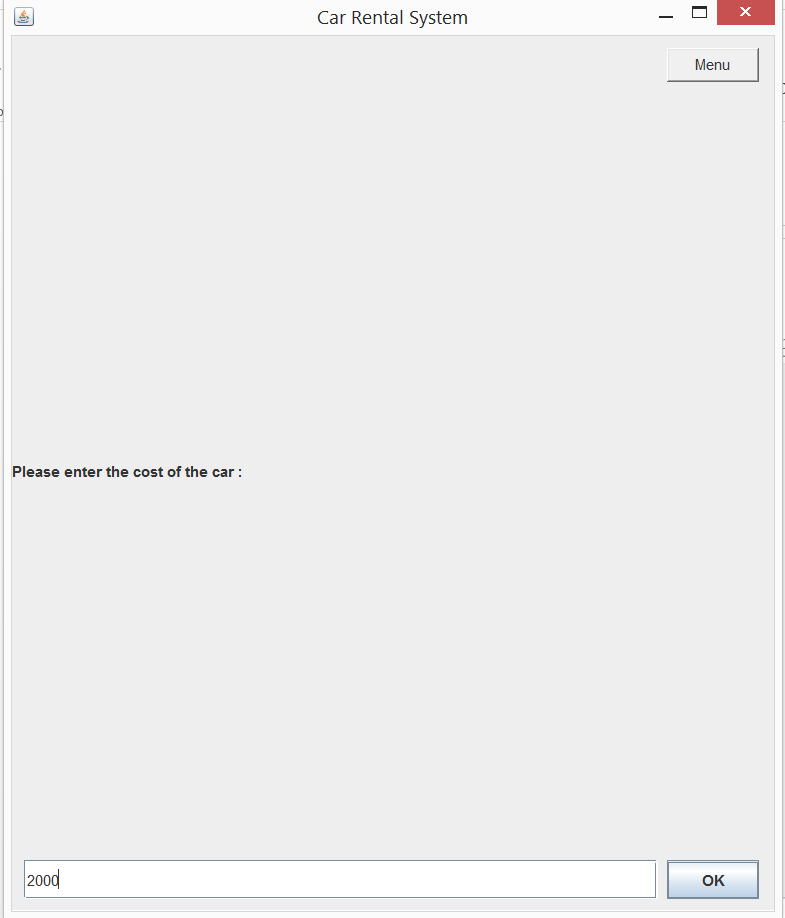
Key in company name



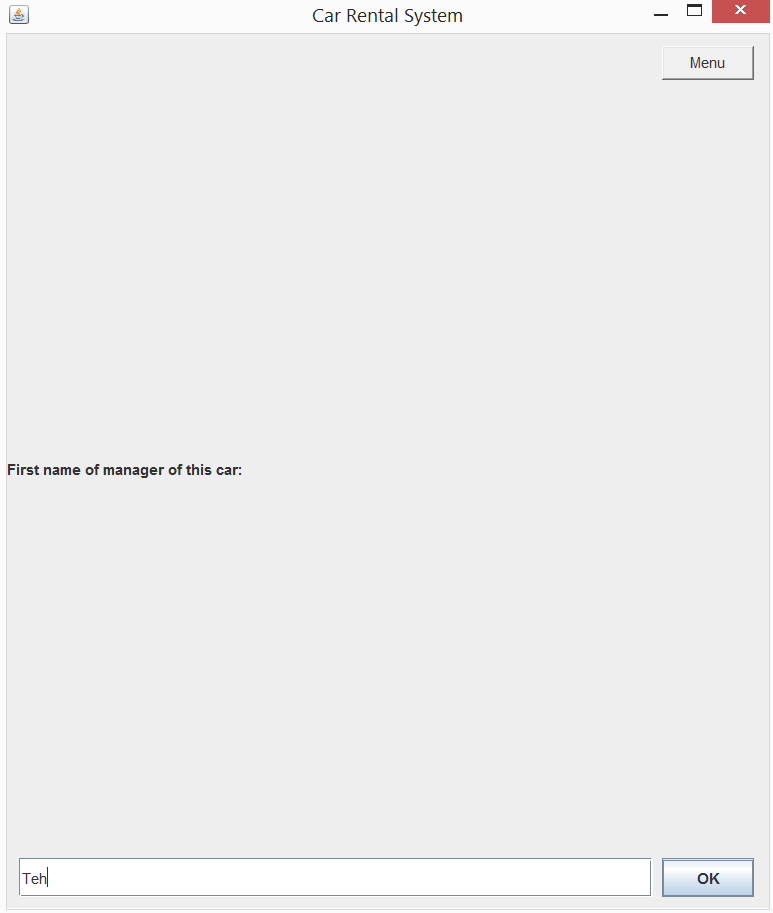
Key in car name



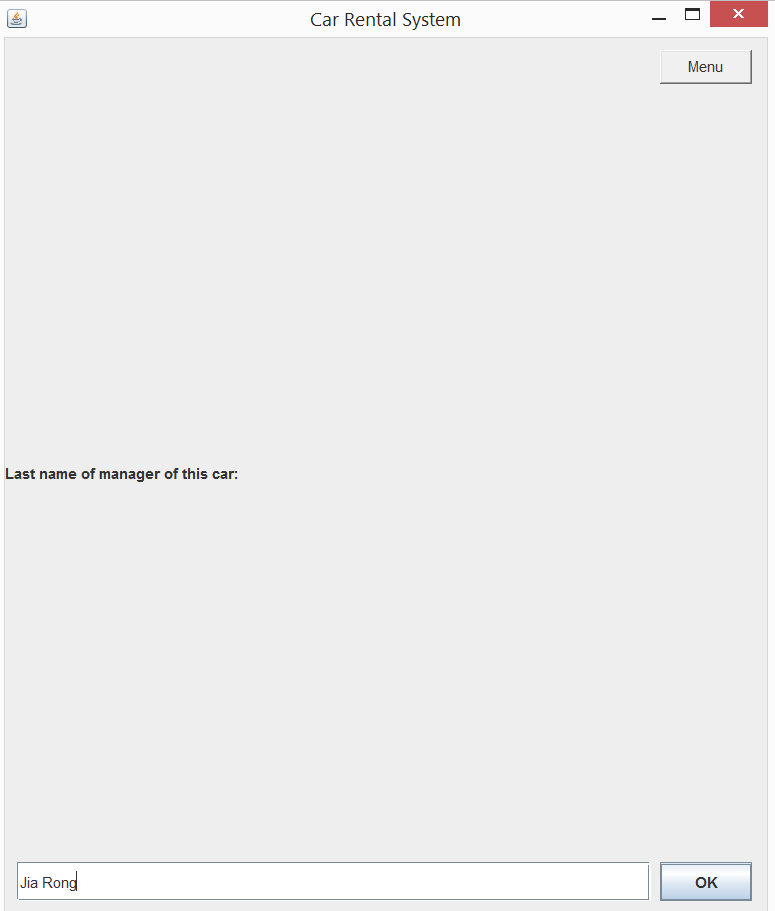
Key in cost



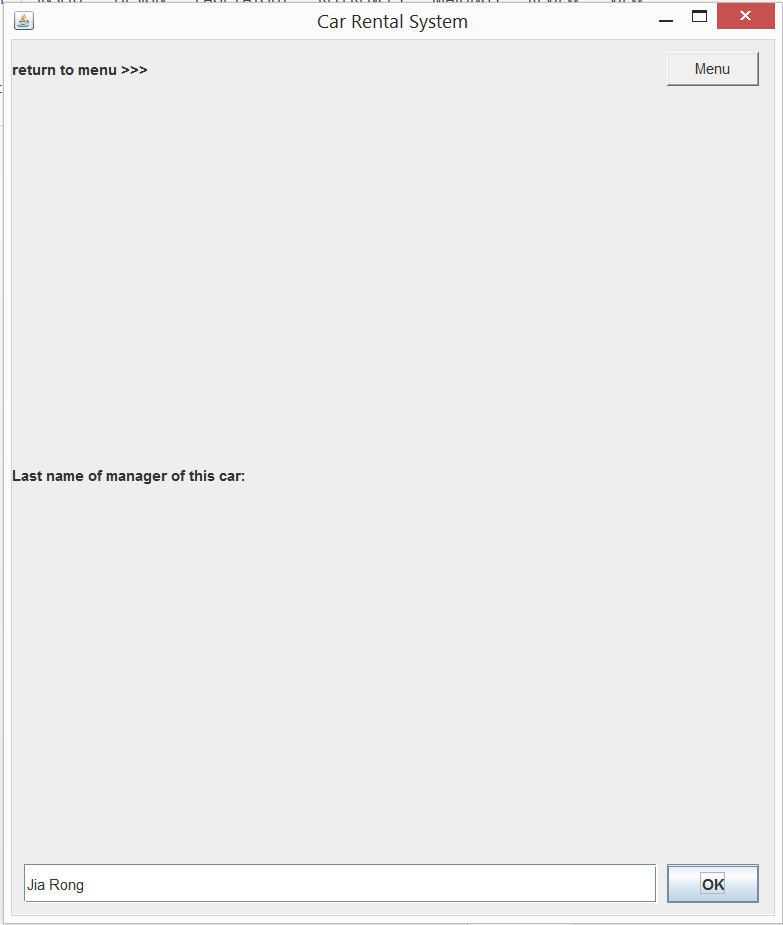
Key in manager first name



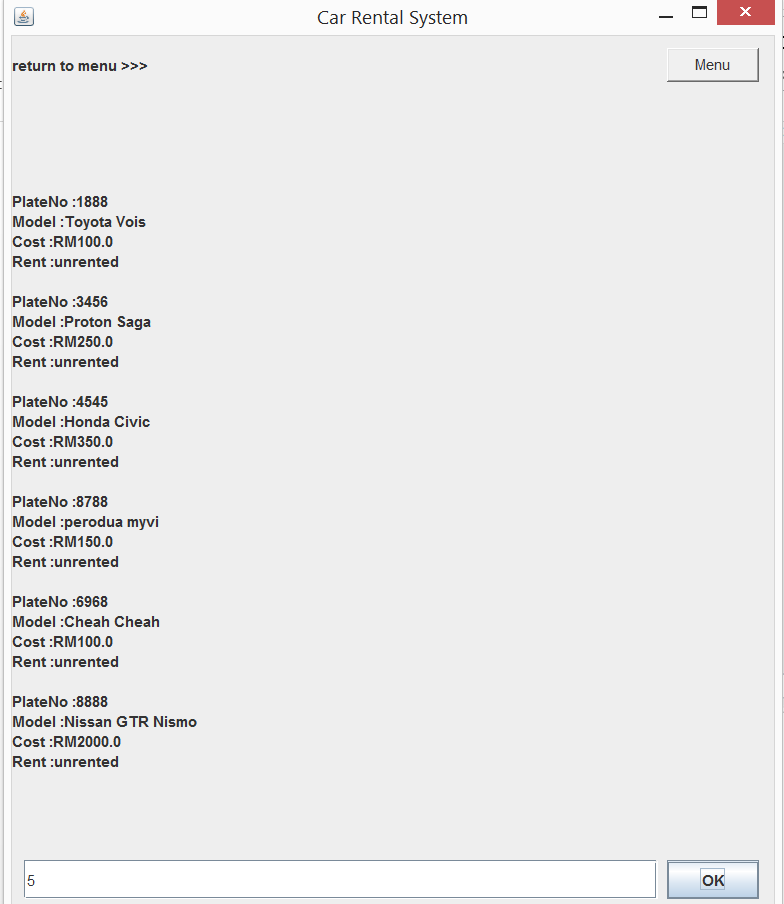
Key in manager last name



Data save and return to menu

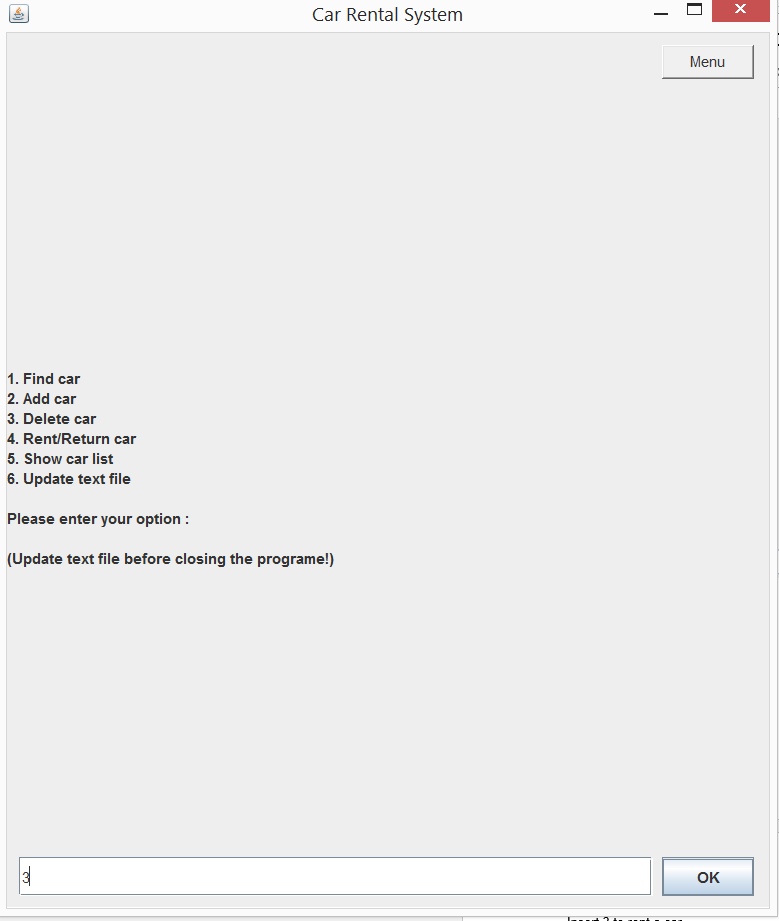


**Insert 5 to show car list**

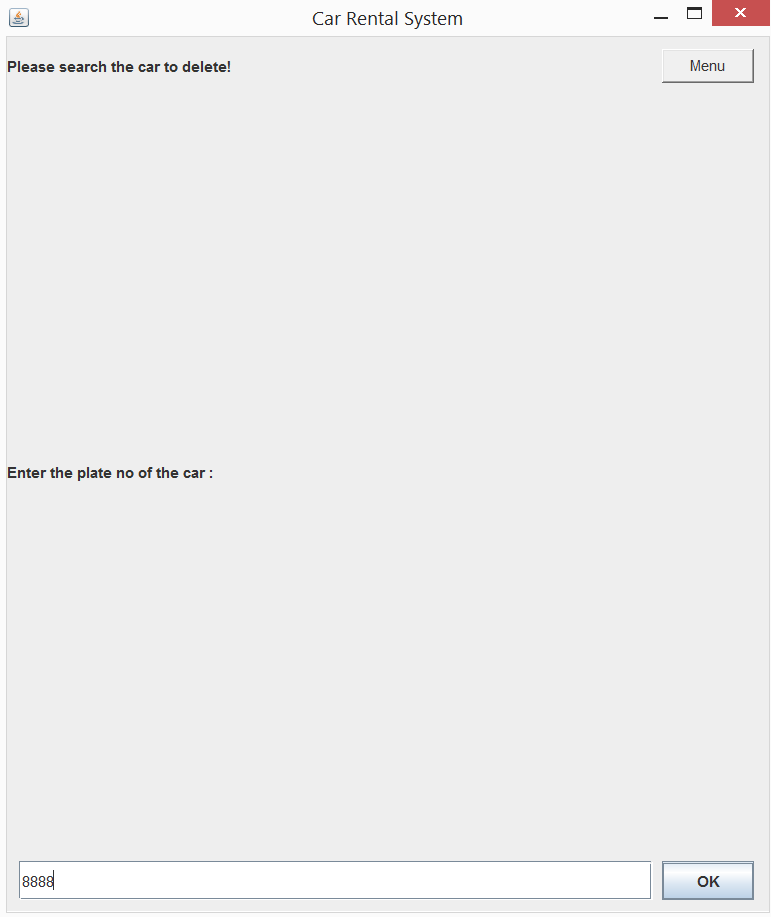


Car List updated

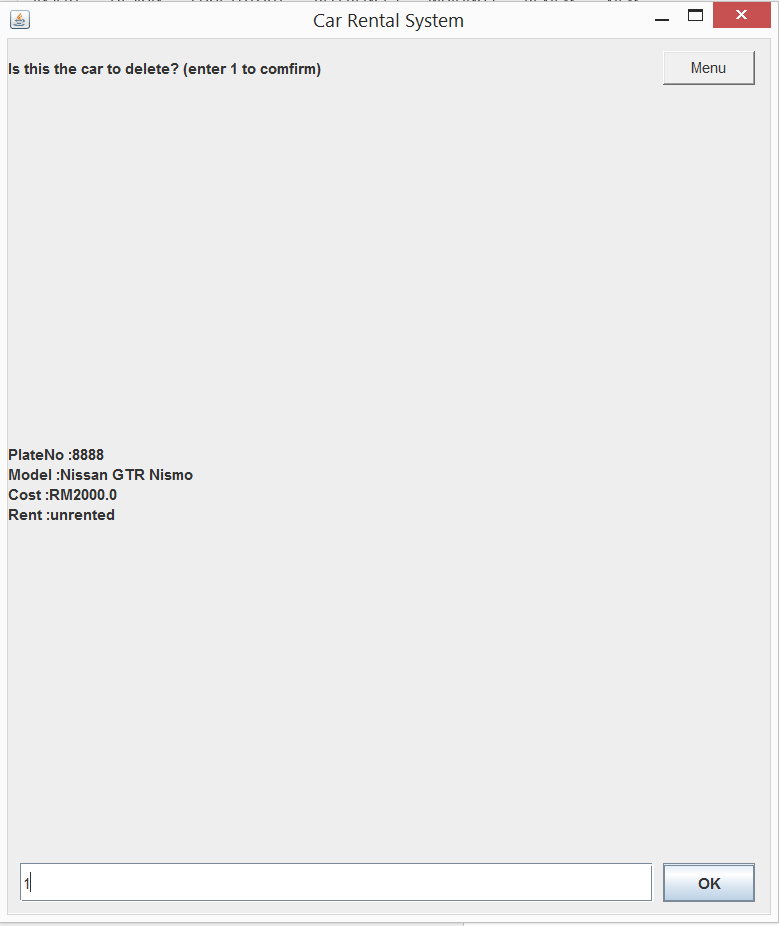
**Insert 3 to rent a car**



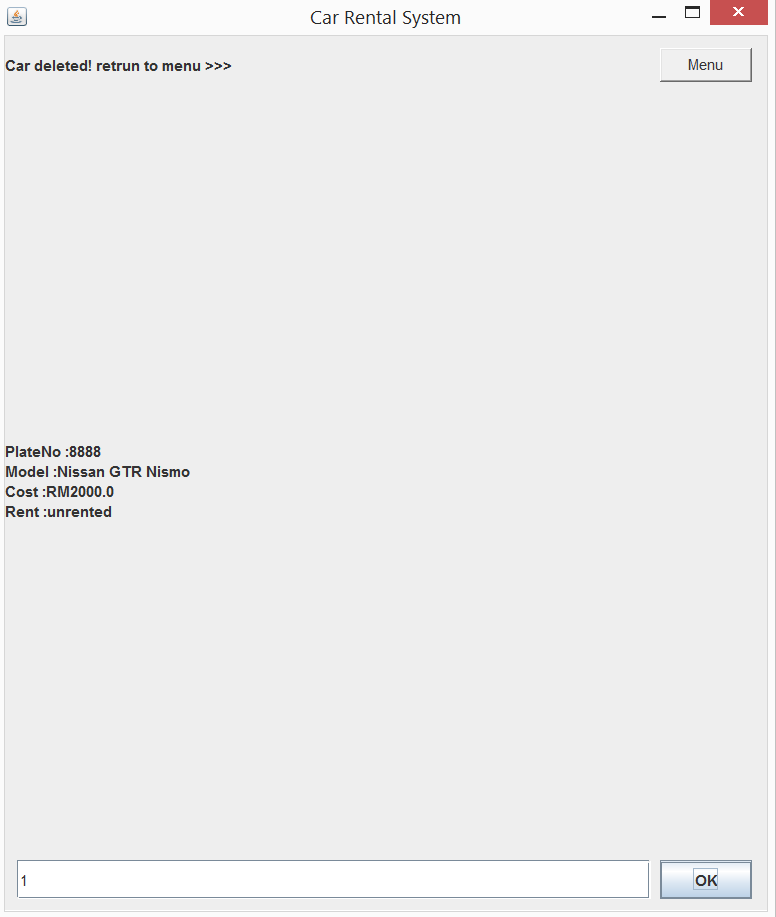
Enter u plate Number to delete the car



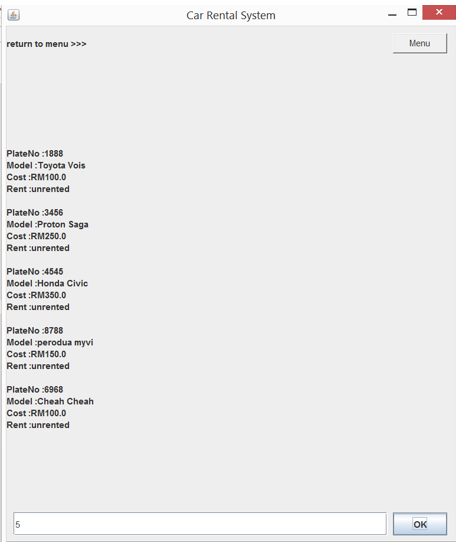
Enter 1 to conform delete



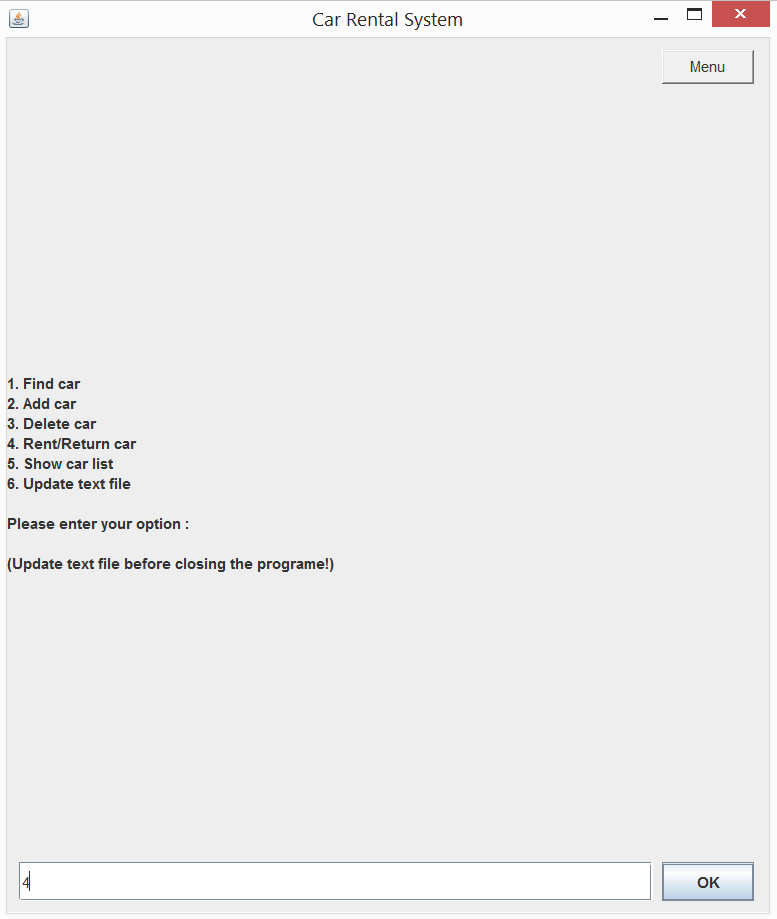
Car Delete



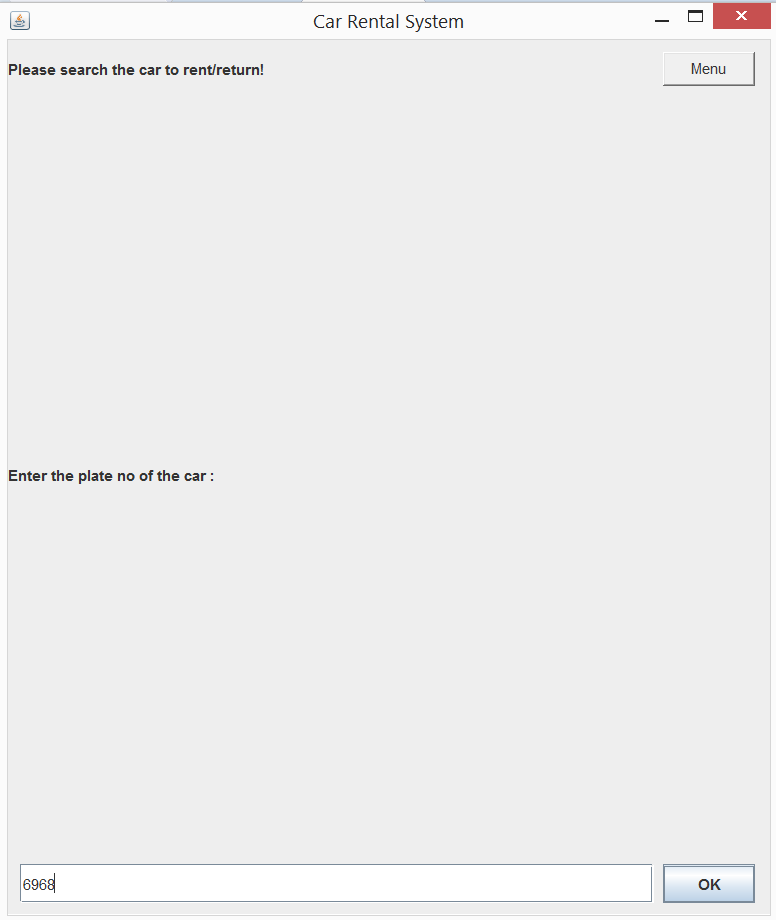
Return to menu by menu button and insert 5check car list.



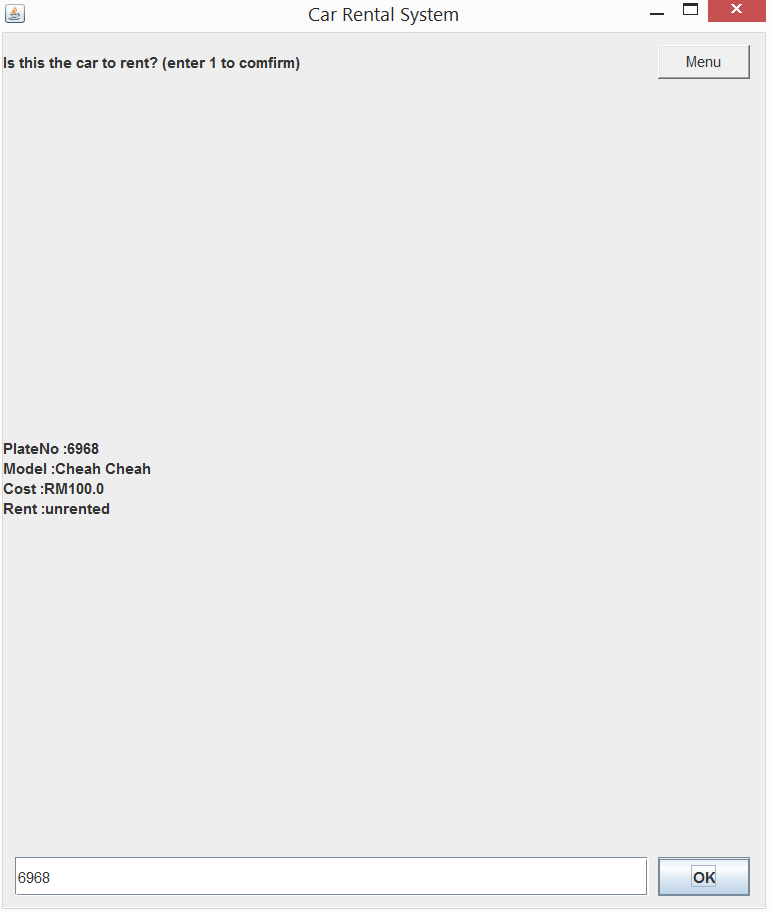
**Insert 4 to rent a car**



Enter the plat No for the car.



Enter 1 to conform rent 1

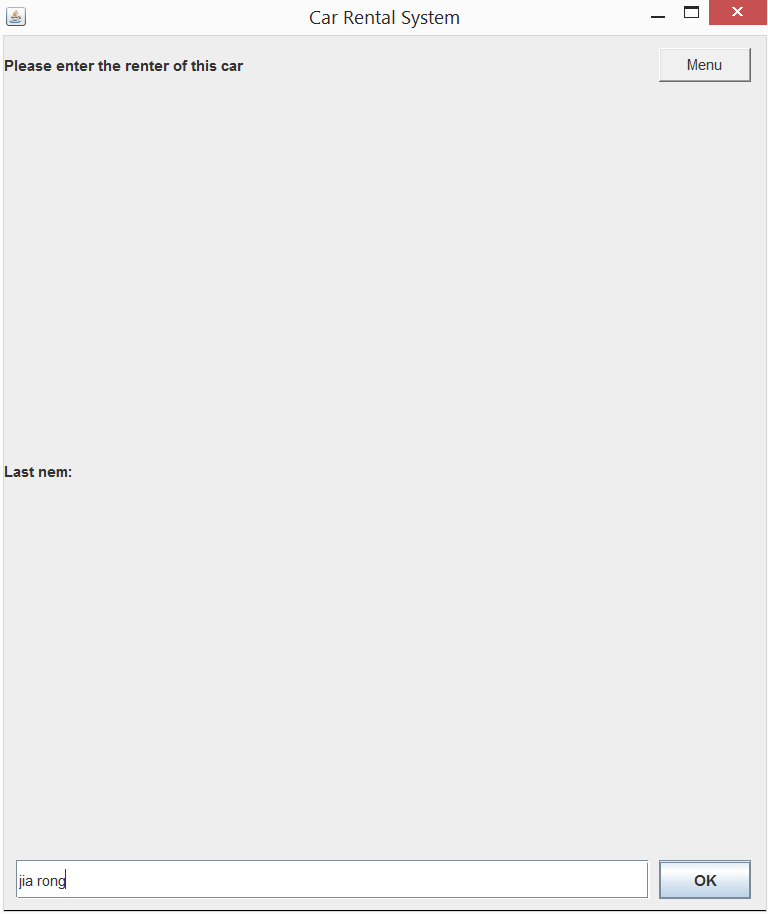


Click one to confirm

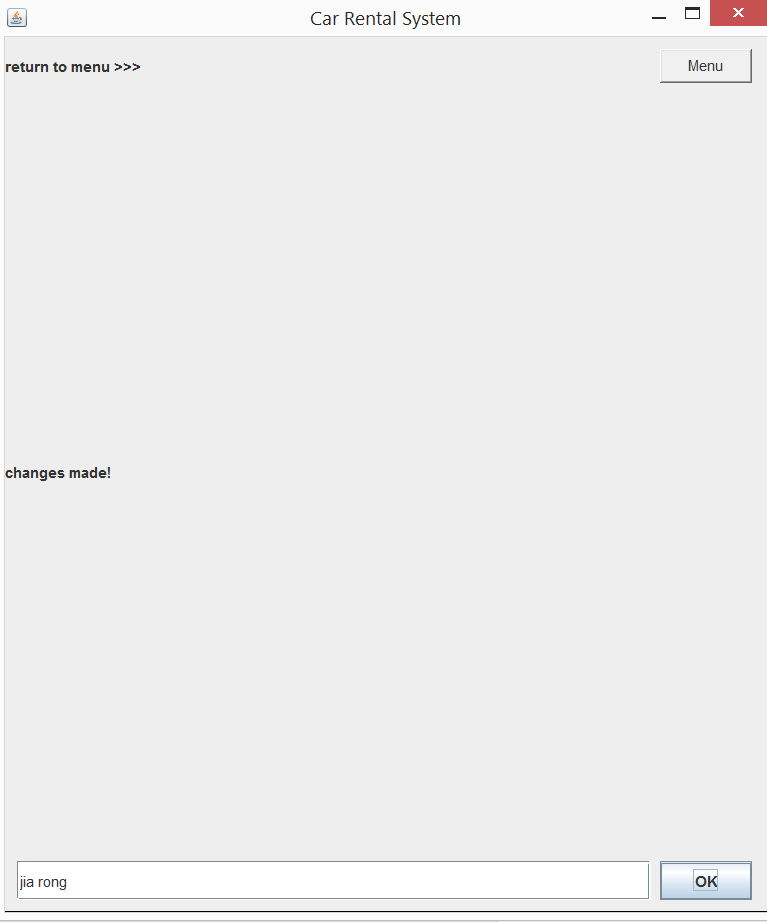
Enter the first name



Enter Last name.



Rent successful



Go back menu by menu button

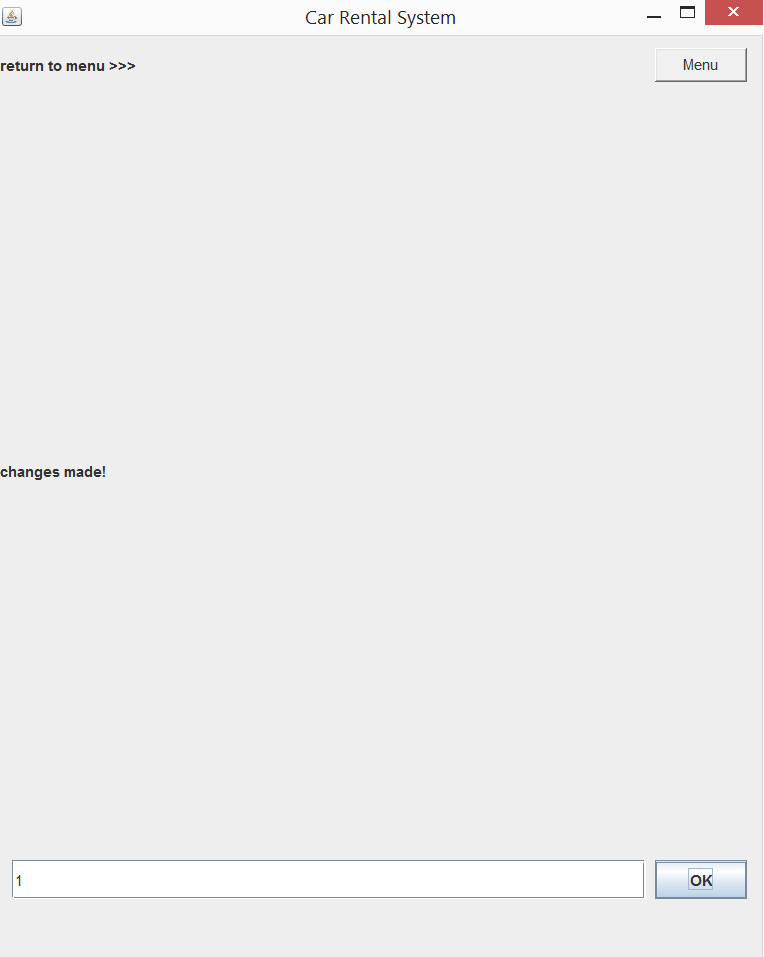
Insert 4 and to return car



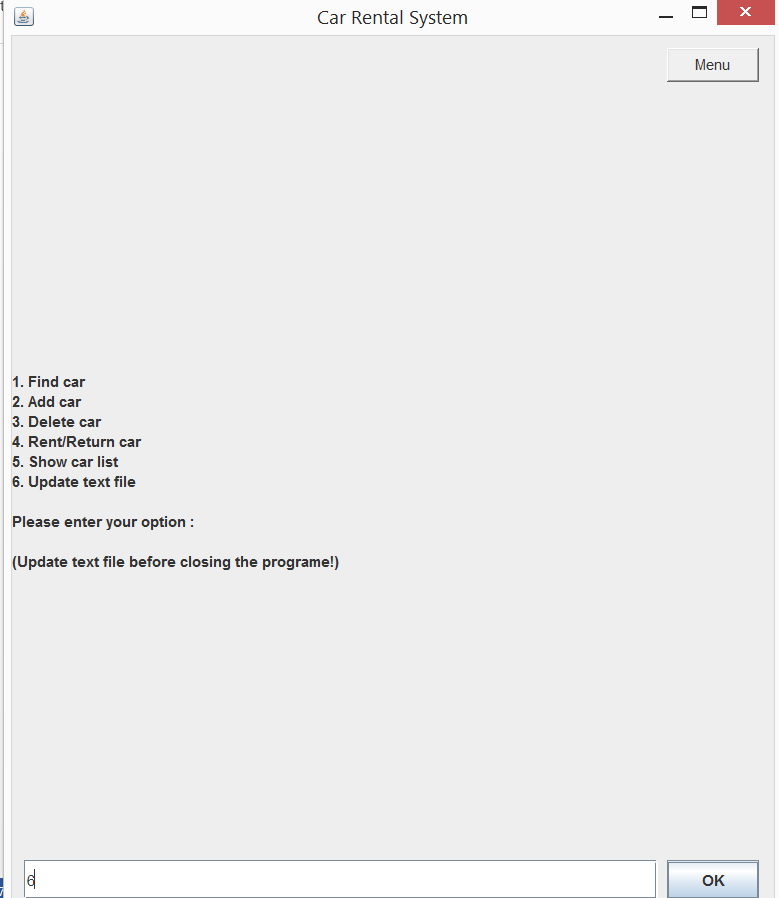
Enter the plat no car you want to return



Enter 1 to confirm



Insert 6 to updated the text file



Changed save.

