# PROGRAM CAR-MANAGER

NGUYỄN HOÀNG DANH – SE63365.

[PROGRAM CAR-MANAGER 1](#_Toc509186062)

[**I.** **INTRODUCTION.** 2](#_Toc509186063)

[**Data structure:** 3](#_Toc509186064)

[ *Car.java :* 3](#_Toc509186065)

[ *CarList.java:* 3](#_Toc509186066)

[ *InputValid.java:* 3](#_Toc509186067)

[ *Menu.java:* 3](#_Toc509186068)

[ *Manager.java:* 3](#_Toc509186069)

[ *Main.java:* 3](#_Toc509186070)

[**II.Function.** 4](#_Toc509186071)

[**A.** **Add a car.** Choice 1. 4](#_Toc509186072)

[**** **Interface:** 4](#_Toc509186073)

[**** **How it works:** 4](#_Toc509186074)

[**** **Constraint:** 4](#_Toc509186075)

[**B.** **Remove a car: Choice 2.** 5](#_Toc509186076)

[**** **Inteface:** 5](#_Toc509186077)

[**** **How it works**: 5](#_Toc509186078)

[**** **Constraint:** 5](#_Toc509186079)

[**C.** **Modify car: Choice 3.** 5](#_Toc509186081)

[**** **Interface**: 6](#_Toc509186082)

[**** **How it work:** 6](#_Toc509186083)

[**** **Constraint:** 7](#_Toc509186084)

[**D.** **Search car: choice 4.** 7](#_Toc509186085)

[**** **Interface:** 7](#_Toc509186086)

[**** **How it work:** 8](#_Toc509186087)

[E. **Sort list car and display:** Choice 5. 8](#_Toc509186088)

[**** **Interface :** 8](#_Toc509186089)

[ **How it work:** 8](#_Toc509186090)

[**F.** **Display list: Choice 6.** 8](#_Toc509186091)

[**** Interface: 8](#_Toc509186092)

[**** **How it work:** 9](#_Toc509186093)

[**G.** **Read file store data. Choice 7:** 9](#_Toc509186094)

[ Interface: 9](#_Toc509186095)

[**** How it work: 9](#_Toc509186096)

[**** Constraint: 9](#_Toc509186097)

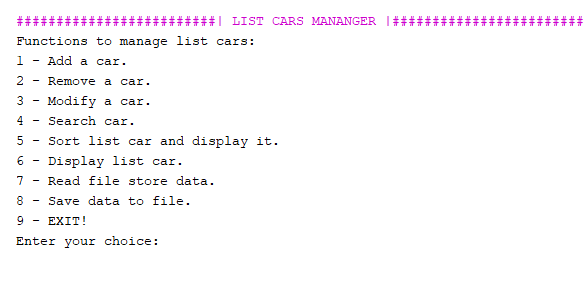
[**H.** **Save to file. Choie 8:** 9](#_Toc509186098)

[**** Interface: 9](#_Toc509186099)

[**** How it work: 9](#_Toc509186100)

1. **INTRODUCTION.**

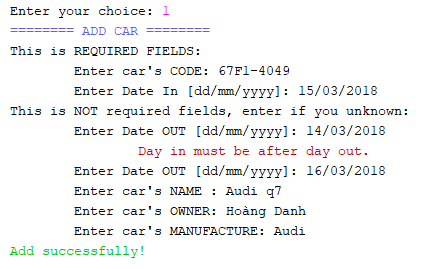
This is a car management program in the parking lot.

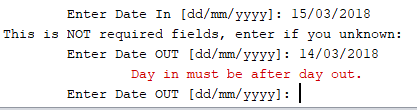
Main interface: We have 8 choices function and 1 choice to EXIT.

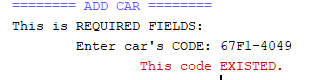
**Data structure:**

* *Car.java :*
* Include 6 field \***code**(license plate - mandatory)**, name, owner, manufacture, \*dateIn and dateOut** and some method comparator, contructor.
* *CarList.java:*
* Include method **(add, search, sort and check exist in list).**
* *InputValid.java:*
* This is class contain method **check valid input** of each data and constraints value.
* *Menu.java:*
* This class display **menu function and notification show Error, Success, Search menu, Sort menu.**
* *Manager.java:*
* All the main method of the program are written in this class( **ChoiceAdd, ChoiceRemove, ChoiceSort, ChoiceSearch, ChoiceModify, Display, Read from file and Write to file).**
* *Main.java:*
* Include command call method in Manager.java to perform on screen.

**II.Function.**

Picture A1: Interface add.

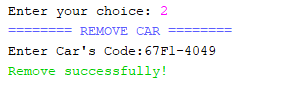
**Picture A2: Add date out Error.

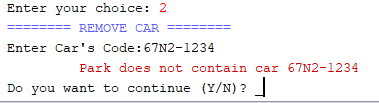
**Pictrue A3: Add code existed.

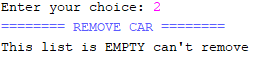
* 1. **Add a car.** Choice 1.
     + **Interface:**
     + **How it works:**

This function accept for user input information of car: code , Name, Owner, Manufacture, DateIn, DateOut.

* + - **Constraint:**
      * 1. **Code** –Must be filled and can’t input code existed in list.
        2. **Date In**- Can’t empty.
        3. **Date out**- Can empty but if you fill that must be after date in.
  1. **Remove a car: Choice 2.**
     + **Inteface:**

Pictrue B1: Interface REMOVE .

Pictrue B2: REMOVE car not exist.

Pictrue B3: List empty.

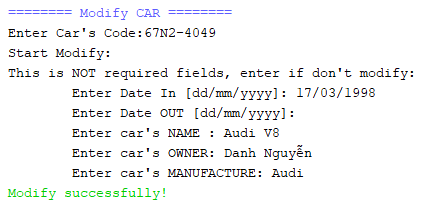
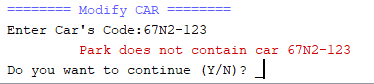
* + - **How it works**:

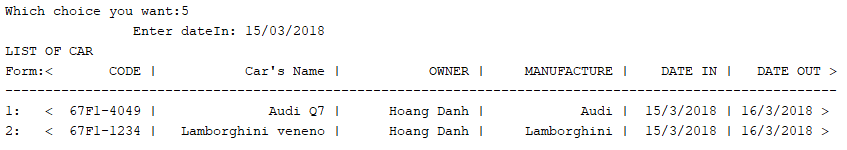
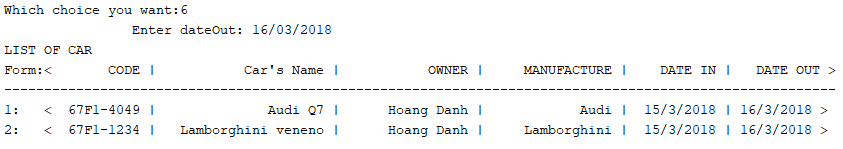
User just input car’s CODE to remove the car in the list , and the program check the code user input Exist in list if true -> remove else notify it not contain in list and ask user continue remove or do orther work.

* + - **Constraint:**

User just can input code to remove, input name, owner….. can’t remove.

* 1. **Modify car: Choice 3.**
     + **Interface**:

 Picture C1: Interface ModifyPicture C2: Code modify not exist.

 Picture D5: Search by Date In.  Picture D6: Search by Date Out.

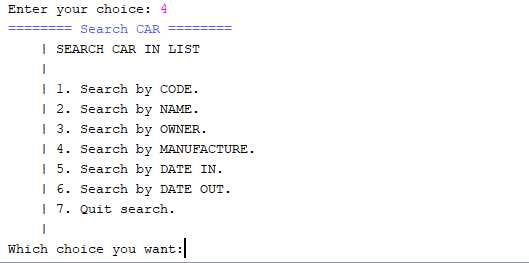
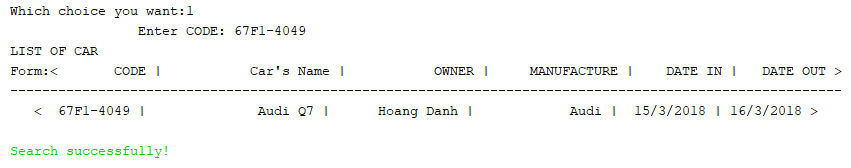
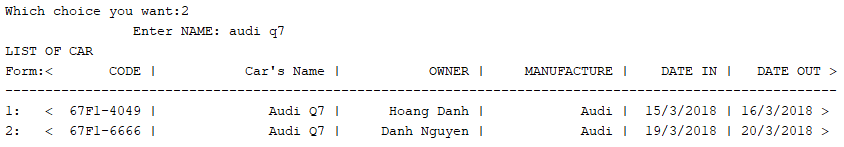
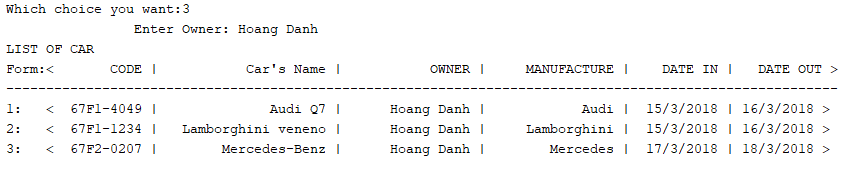
* + - **How it work:**

User input code of car, program will accept user input new data of car if any information user don’t want to modify user can press ENTER to pass it. If user modify Date In after Date Out, Date out be going to empty.

* + - **Constraint:**

User must input correctly code of the car to modify it cant input name or anything to modify.

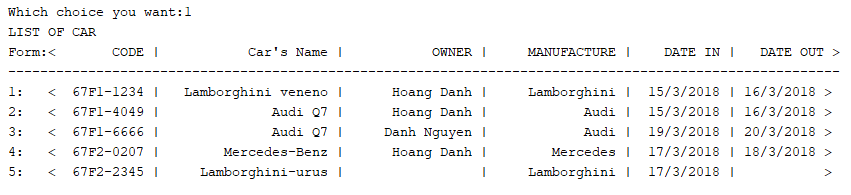
* 1. **Search car: choice 4.**
     + **Interface:**

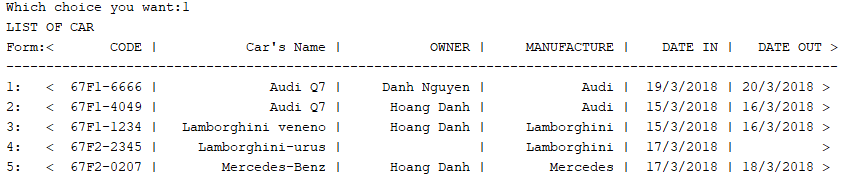
 Picture D1. Interface Menu search. Picture D2. Search by code.Picture D3: Search By name.Pictrue D4: Search By Owner.

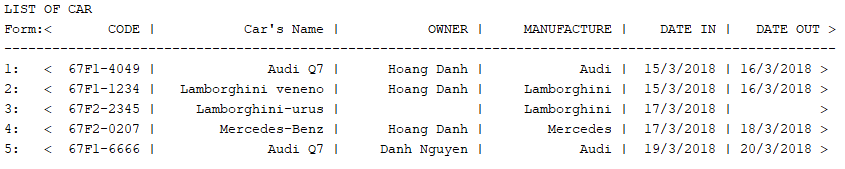
* + - **How it work:**

This choice accept user search a car in list by Code, Name, Owner, Manufacture, DateIn, DateOut.

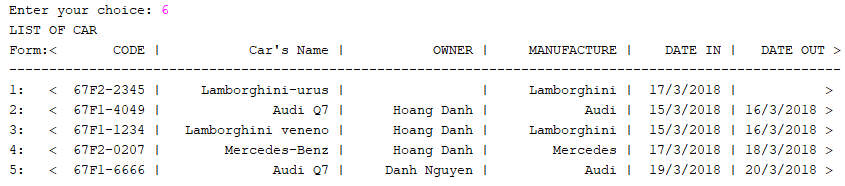
* 1. **Sort list car and display:** Choice 5.
     + **Interface :**

 Picture E2.Sort by code Ascending.

Picture E3. Sort by Name.

Picture E4. Sort by DateIn.

* **How it work:**
* when user choice sort program will show menu sort user can choice sort by Code, Name, Owner, Manufacture, DateIn or Date Out. After that a new menu will be showed with 2 choice Ascending or Descending.
  1. **Display list: Choice 6.**
     + Interface:



* + - **How it work:**

Choosen 6 is the function help user display list of car on screen.

* 1. **Read file store data. Choice 7:**
     + Interface:
     + How it work:

The program allows users to read data from a file. After that program will show notification Successfully.

* + - Constraint:

Data just be read to program from file only if data in file correct format:

* + - 1. Code
      2. Name
      3. Owner
      4. Manufacture
      5. Date In
      6. Date out.
  1. **Save to file. Choie 8:**
     + Interface:
     + How it work:

The program allows users save data to file.