Driving school C++ management system

“Driving school management system” has been design to avail the registry process in any Driving school around the globe and assert the applicability of C++ coding in problem solving and in automated systems.

The system helps the clients to choose their preferred car type, time of practice and the captain responsible of teaching them. The friendly interface and its ease of use provide the clients and the employees with an organized data and functions to maintain the registry process.

Hierarchy of the system:

Once the program is opened, the user faces a **Main Menu** with the following options:

1. Customer entry
2. Sessions Management
3. Print full data base
4. Clear data
5. Car Management
6. Close program

Each of these options contain other sub – menus and embedded modules; the following is an explanation of each option.

1- Customer entry:

This option contains the following sub menus:

1- Add member
2- List Records
3- Modify Records
4- Delete member
5- Return to main menu

Option 1: Requests the user to enter each client’s Name, ID and phone number.

Option 2: Shows the data of the registered clients.

Option 3: Asks the user of the Client to-be-edited ID the enable the user to edit other client’s data.

Option 4: Asks the user to enter the ID of the client to be removed then.

2- Sessions Management:

This option contains the following sub menus:

1- Add Session
2- Filter by Day & Type
3- Filter by Time & Type
4- Cancel a session
5- Return to Main Menu

Option 1: Allows the user to choose his preferred timing of practice of the week (Day/hour) and His preferred Car type (Manual / Automatic) then provide the user with the available Captain.

Option 2: User chooses his preferred day of practicing and car type then the program prompts all the available hours of practicing and Captains considering User's choice. After that the user can book his session.

Option 3: User chooses his preferred hour of practicing and car type then the program prompts all the available days of practicing and Captains considering User's choice. After that the user can book his session.

Option 4: Program asks the user to enter the ID deletes his/her sessions.

3- Print full data:

By selecting this option; a 2d array looking like an excel sheet is printed containing all the data and stating the booked sessions by providing the Booker's ID and the empty sessions.

4- Clear data:

Selecting this option will clear all the booked sessions and resets it to empty.

5- Car management:

This option contains the following sub menus:

1- Add car
2- History of cars
3- Return to main menu

Option 1: Allows the user to add a new car by entering its characteristics (name, model, color and type)

Option 2: Displays the data of all the cars and their data.

6- Close program:

Terminates the program.

Data Handling in the system

The user mainly interacts, store and retrieve data from three txt files respectively, each file is associated with a main module.

Customer entry → interacts with users.txt

Sessions Management, Print full Data and Clear Data → interacts with Book2.txt

Car Management → interact with cars.txt

Modules in the Program:

The program has three main modules, each containing sub modules carry out specific tasks.

The three main modules are:

- **Customer entry modules** that always checks if the ID entered is unique to prevent any duplication of user's data
- **Session management Modules** that always checks if the session is already booked or not to prevent the user from booking already taken sessions.
- **Car management Module**

Credits

This project was made by the members of group 101.

- Abdelrahman El badawy
- Abdelrahman Ashraf
- Abdelmoez Ashraf
- Ahmed Khaled Ramadan
- Omar Ashraf Helmy
- Mahmoud Ashraf Gad

Project code line by line

Start	End	Name	Type	Purpose
1	9		Header statements	
11	17	Customer	struct	Contain customer data Global declaration of object customer of type object
19	23	Car	struct	Contain Car data Global declaration of object Car of type Car
27	40	Read_file	Void function	Function containing a while loop to read cars.txt
42	60	AddCar	Void function	Function that write Car data to cars.txt file
63	71	checkID	Boolean function	Function to check if the id entered by user in custom entery menu already exists
73	80	savefile	Void function	Write/append/read data from users.txt
81	94	DataToArr	Void function	Function containing while loop to access data from users.txt file and store it in an array
96	105	No0Items		Calculates the number of users entered in the users.txt file by calculating the number of lines in that file
107	118	printData	Void function	Prints the data stored in 2d array arr that contains sessions data from Book2.txt
120	127	writeData	Void function	Writes data to Book2.txt that contains all the sessions data
129	175	addSession	Void function	Function containing two for loops to loop over columns and rows of data in Book2 text and a Boolean to check if the session is booked or not. Then a few if conditions used to verify that the for loop is located on the right data the user has chosen to make him book the right day/time/ car and captain
177	194	Bookfree	Void fucntion	Function containing for loop and if conditions to loop over the data present in array arr containing data present in Book2.txt The purpose is to change the cell containing none to user's ID to verify that the session is booked
196	216	FilterDayType	Void function	User enter the preferred day and type then for loops are used to loop over array containing Book2.txt data to see all the sessions available. The user then chooses one session

216	235	FilterDayType	Void function	After the user make a choice, if conditions check if the sessions is booked or not then prompts to the user.
237	276	FilterTimeType	Void function	User enter the preferred time and type then for loops are used to loop over array containing Book2.txt data to see all the sessions available. The user then chooses one session After the user make a choice, if conditions check if the sessions is booked or not then prompts to the user.
277	283	ClearData	voidfunction	Clears all the ids in arr[][4] by using for loop
285	303	deleteSession	voidfunction	For loop is used to loop over the user IDs present in arr[][4] Then if condition verify if the ID exists or not to delete it
305	568		Main function	
305	310			Opening user.txt and book2.txt
310	315			Two for loops for reading data from text 2
319	341			Writing to the console the Customer entry options
343	368		Switch Statement	to add members by getting ID from the user and calling checkID function. Then writing data to users.txt
369	380		Switch Statement	to view all records DatatoArr function is called
381	422		Switch Statement	User enters id to console checkId functions is called to verify its uniqueness while loop breaks only when user enter unique ID savefile function is called to save the modified data
424	446		Switch statement	User enters id to console Data is demolished using for loop DataToArr function is called Savefile function is called
456	462		If statement	Console outputs session management sub menu
464	472		If statement	Add session module addSession and writeData functions are called
473	481		If statement	FilterDayType and write Data functions are called
482	490		If statement	FilterTimeType and write Data functions are called
491	499		If statement	deleteSession and writeData functions are called
500	508		If statement	printData function is called

509	528		If statement	Cleardata function is called All the bookings get cleared
539	554		If statement	Car Module Addcar and readfile functions are called
556	566		If statement	Group credits outputted to the console