**Mates-Rates Rent-a-Car User Manual**

Jack Gate-Leven 2020

This program is for Mates-Rates Rent-a-Car employees to use to perform work operations. It runs on command line and requires user keyboard input. Use this program to perform all necessary operations.

**Running**

The program can be run by opening the project in visual studio and pressing run or building the program and then executing the executable \MRCC\bin\Debug\MRCC.exe.

Data files can be supplied looking at the below section , or new files will be created upon start-up of the program if none are supplied.

The program can be run through command line by moving to the \MRCC\bin\Debug folder and entering ./MRCC

**Folders**

The MRCC folder holds the VS solution and executable files while the MRRCManagement folder holds the class manager and all other files for the program.

**Files**

The data files for this program are expected to be held in the Data folder of the program solution and should be .csv files.

* The customers.csv file contains the following 6 columns: ID, Title, FirstName, LastName, Gender, DOB
* The fleet.csv file contains the following 12 columns: Registration, Grade, Make, Model, Year, NumSeats, Transmission, Fuel, GPS, SunRoof, DailyRate, Colour
* The rentals.csv file contains the following 2 columns: Registration, CustomerID

The files hold all the data to be used by the program, CRM and Fleet load and save to the files. If files do not exist when the program is started, then the CRM and Fleet will create news ones with the same names as above.

The user will be meet by a prompt for each files name and location relative to the data folder. If enter is pressed, then the default file names will be provided. If any of the files do not exist, then new ones will be created with just the file headers in them. If either the customers or fleet files are not properly loaded, then a new rentals file will be created.

**Program Usage**

The program is to be used by Mates-Rates Rent-a-Car employees for all work-based operations. The program starts and loads into the home menu where the user can navigate to the appropriate menu for the operation they wish to perform. Keyboard input is required, and key presses are used for all the menus. H and q are special and can be used to return to the home menu or to exit the program from any menu. For instance, user can navigate to the customer menu by pressing the a key and then can display all customers by pressing the a key again.

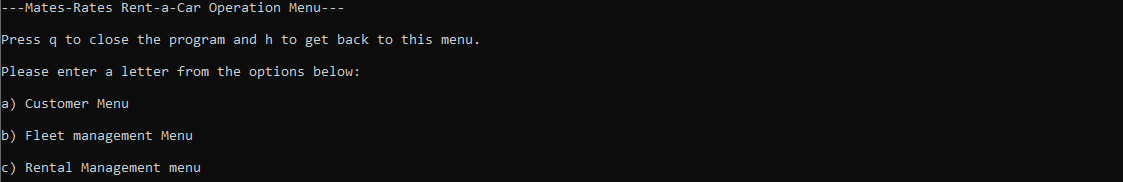
Users need to know what operation they wish to perform before using the program, but then once they start the program it is straight forward to use. If the program runs into an error or the user inputs an incorrect value, then they are prompted by the system to return to a menu or retry the activity. The program alerts the user as to what it was, they did incorrectly.

In order to properly save all changes to file the user must enter exit while in any menu, otherwise changes will not be recorded.

**Main Menu**

This menu is displayed when the user starts the program and directs them to option a, b, c from which operations can be performed. This menu tells the user that the h and q keys perform special actions. The user only needs to enter a single key and the program reads that as the input.

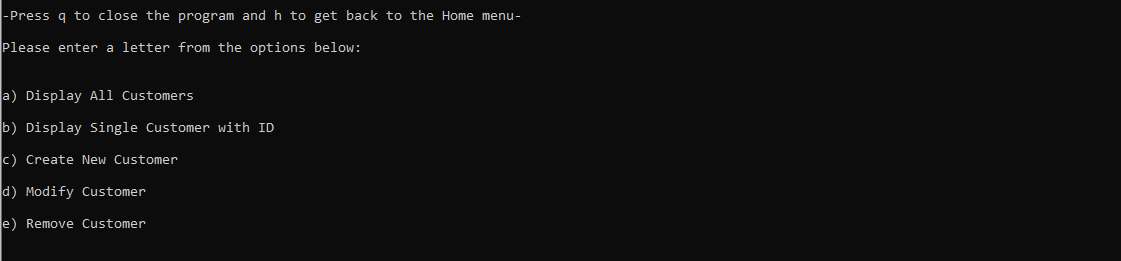
1. Customer menu
2. Fleet management menu
3. Rental management menu



**Customer Resource Management Menu**

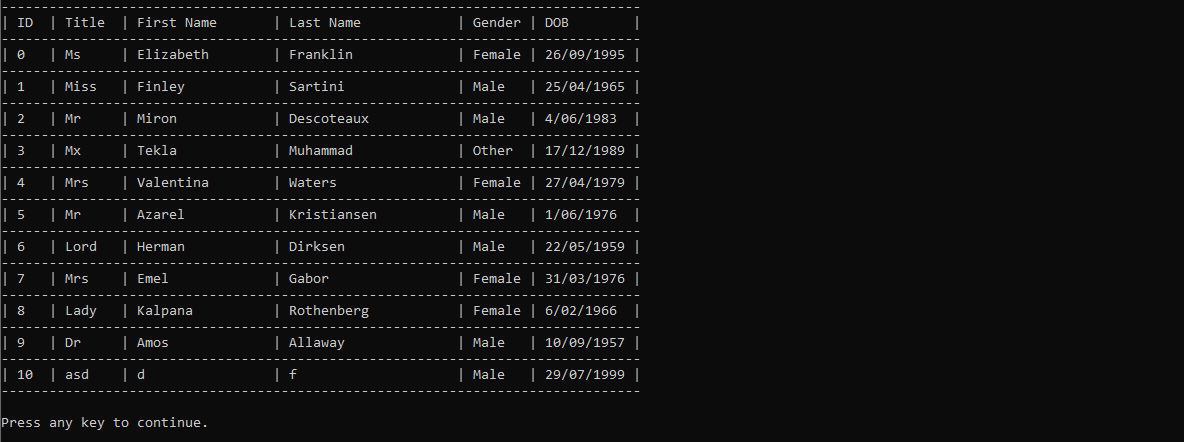
This menu is displayed when the user enters the customer menu and directs them to option a, b, c, d, e from which operations can be performed. The special keys are still displayed. A single key input is read as the user input.

1. Display all customers
2. Display single customer with ID
3. Create new customer
4. Modify customer
5. Remove customer



**Customers Display**

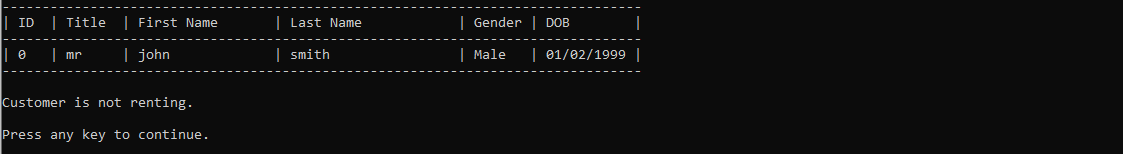
This operation displays a table of all customers in the CRM, with all properties of the customers. If no customers exist, then the user is told so.

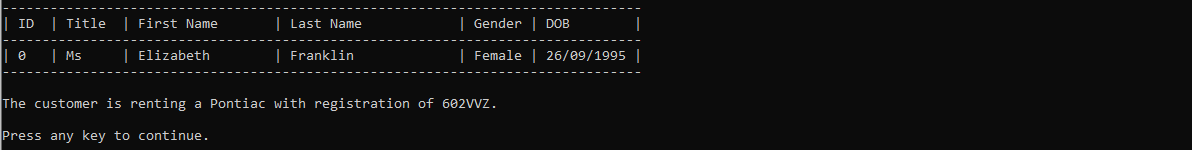




**Individual Customer Display**

The user inputs a customer ID and the operation displays a table of all the details of that individual customer. The rental status of the customer is also displayed underneath.





**Create Customer**

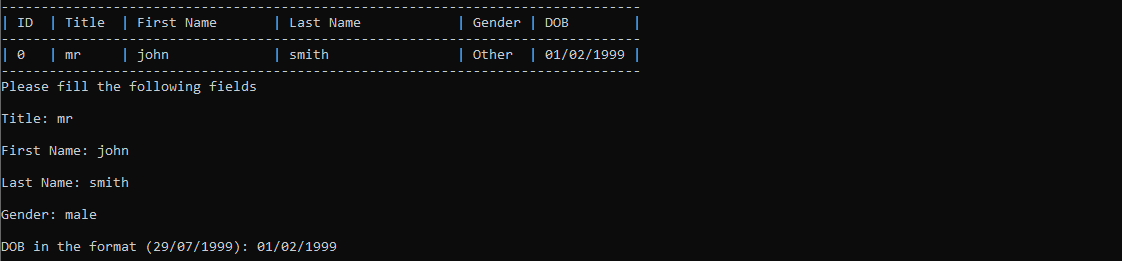
This process allows for the creation of a new customer. The user needs to give input for 6 options, all are mandatory, in order to create customer. The title, first name and last name can be given any input but must not be given just a blank input. The gender needs to be either male, female or other. The DOB needs to be in the format of dd/MM/yyyy and needs to be a valid date from 1900 to 2099.



**Modify Customer**

In order to modify a customer, the user must go through the same process as when creating a customer, the customers properties are supplied for comparison when changing anything. All properties can be changed as the customer ID is not editable.

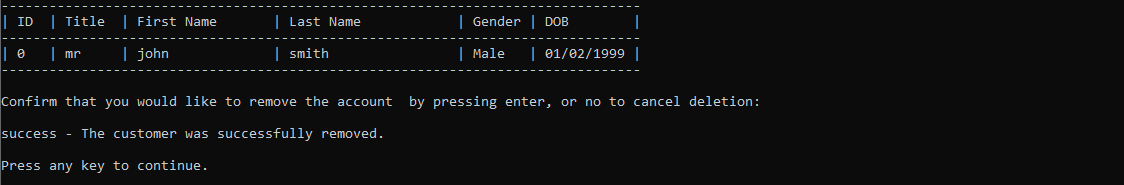




**Remove Customer**

The user can remove a customer from the CRM here. The user only needs to enter the ID of the customer and if that customer is not renting then they can confirm the removal. If the customer is renting, then the removal cannot be performed, and the user is told so. The customer is displayed as the user is asked for confirmation to confirm they supplied the correct customer.

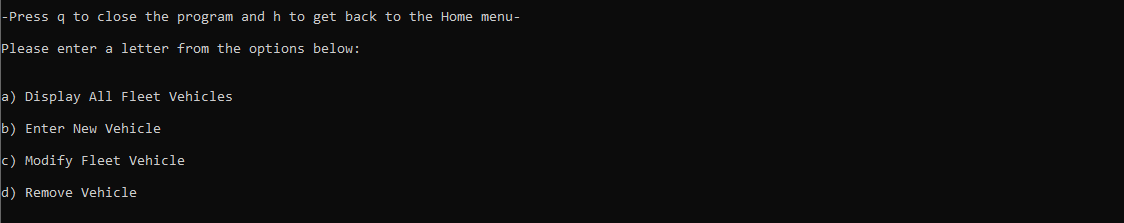




**Fleet management Menu**

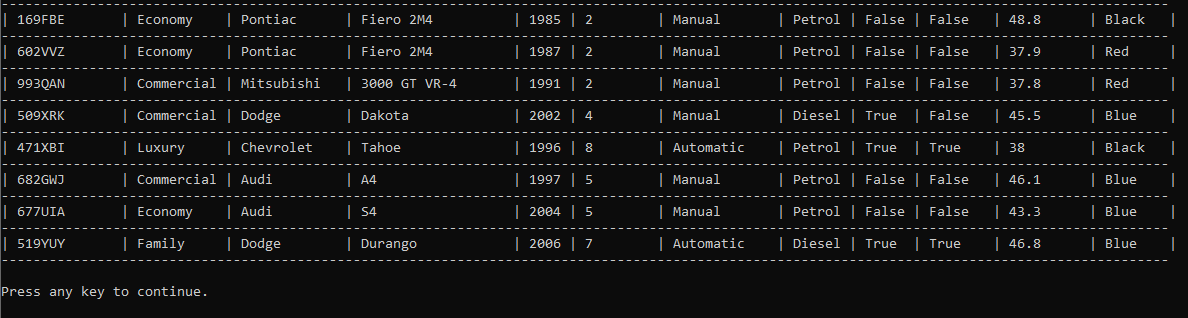
This menu is displayed when the user enters the fleet menu and directs them to option a, b, c and d from which operations can be performed. The special keys are still displayed. A single key input is read as the user input.

1. Display all fleet vehicles
2. Enter new vehicle
3. Modify fleet vehicle
4. Remove vehicle



**Display fleet**

This operation displays a table of all vehicles in the fleet, with all properties of the vehicles. If no vehicles exist, then the user is told so.





**Create Vehicle**

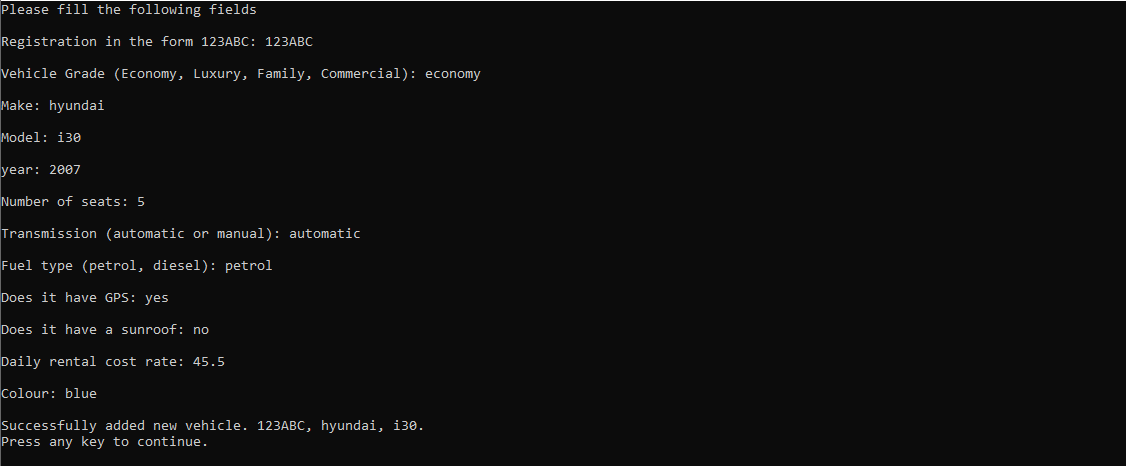
This process allows for the creation of a new customer. The user needs to give input for 5 mandatory properties and the other 7 optional properties can be given by the user or are given defaults based of the vehicle grade. The user is asked for either 1 if they have all the vehicles details or 2 if they only know the basics. Option 1 allows the user to input the mandatory properties and option 2 asks for all mandatory and optional properties.

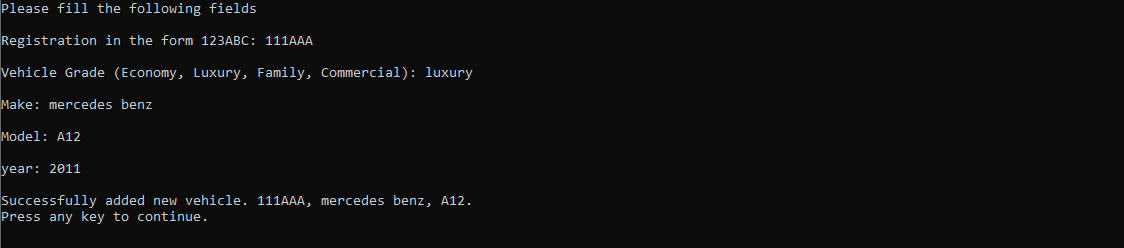
The 5 mandatory properties are as follows:

1. Registration which needs to be in the format 123ABC and cannot be the registration of a vehicle already in the fleet.
2. The vehicle grade must match Economy, Luxury, Commercial or Family. Each grade gives the vehicle multiple default properties e.g. whether the car has a sunroof or not. Unless the user chooses to input all the vehicle’s properties.
3. Make can be any input but cannot be blank.
4. Model can be any input but cannot be blank.
5. Year must be 4 digits between 1900 and 2099

The optional properties are as follows:

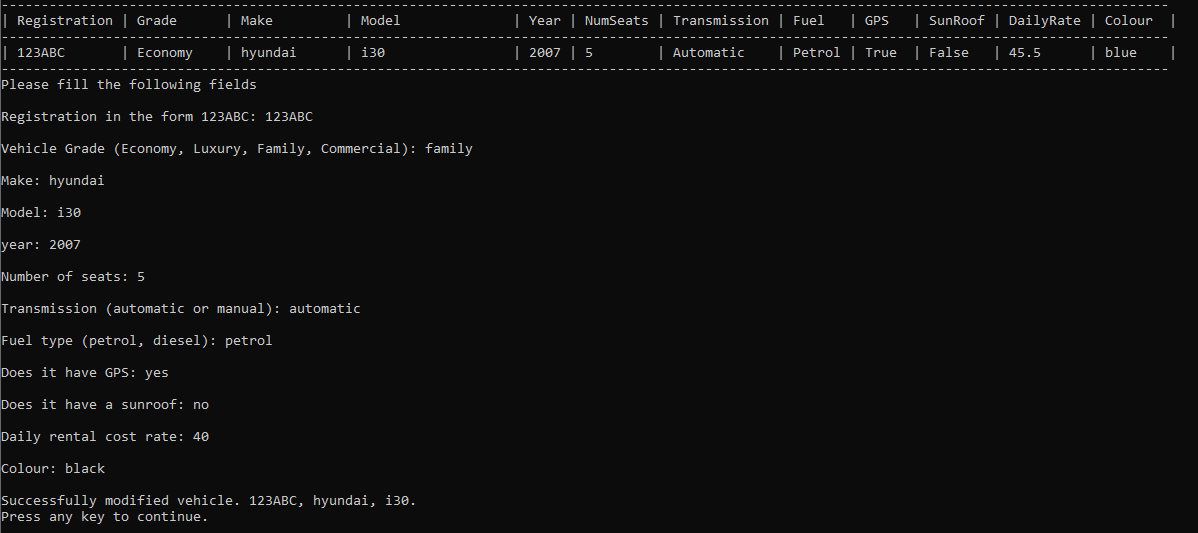
1. Number of seats must just be a number.
2. Transmission must match either automatic or manual.
3. Fuel type must match either petrol or diesel.
4. GPS must match either yes or no.
5. Sunroof must match either yes or no.
6. Daily rental cost must be number, either 50 or 50.55 work.
7. Colour can be any input but can’t be blank.

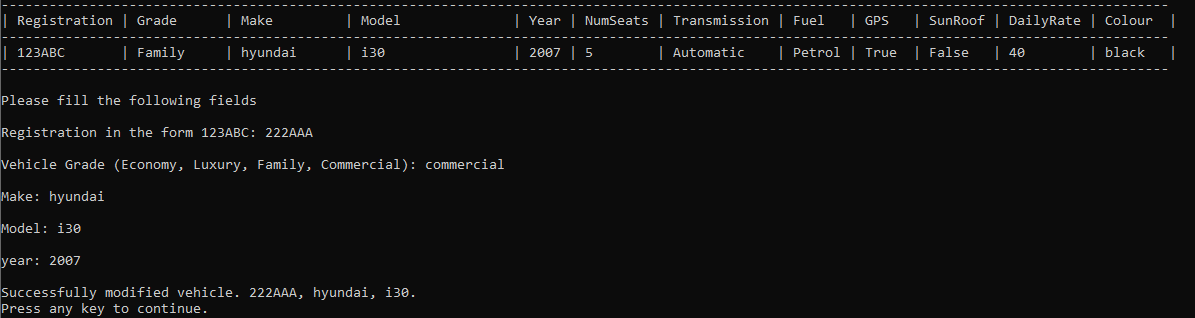




**Modify Vehicle**

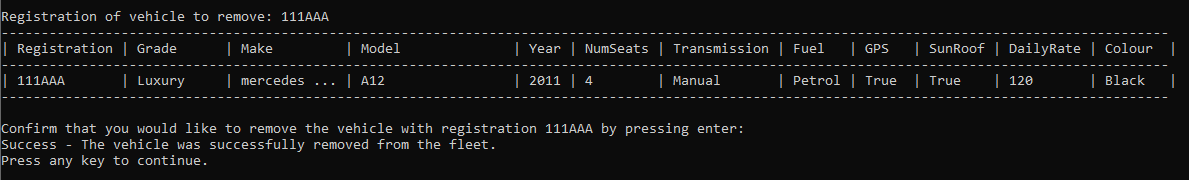
In order to modify a vehicle, the user must go through the same process as when creating a vehicle, however they are able to change from inputting all properties to only the mandatory ones. The registration can be changed but only if the vehicle is not being rented and all other properties can be changed.





**Remove Vehicle**

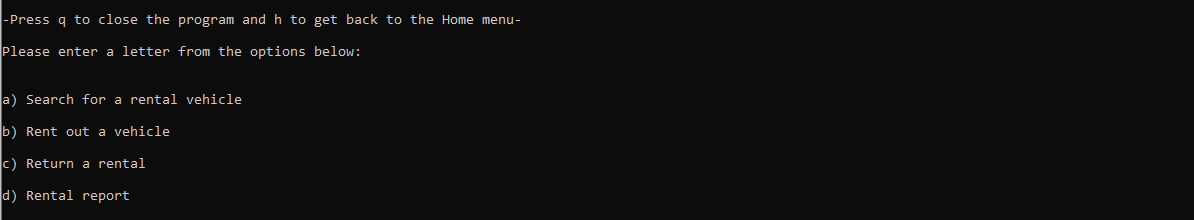
The user can remove a vehicle from the fleet here. The user only needs to enter the registration of the vehicle and if that vehicle is not being rented then they can confirm the removal. If the vehicle is being rented, then the removal cannot be performed, and the user is told so. The vehicles is displayed as the user is asked for confirmation.



**Rental management Menu**

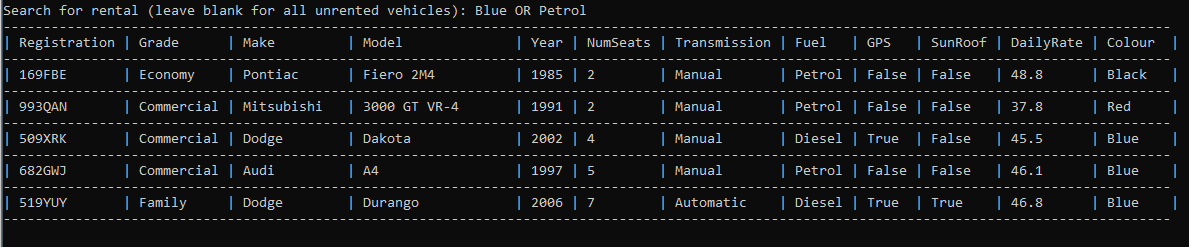
This menu is displayed when the user enters the customer menu and directs them to option a, b, c and d from which operations can be performed. The special keys are still displayed. A single key input is read as the user input.

1. Search for a rental vehicle
2. Rent out a vehicle
3. Return a rental
4. Rental report



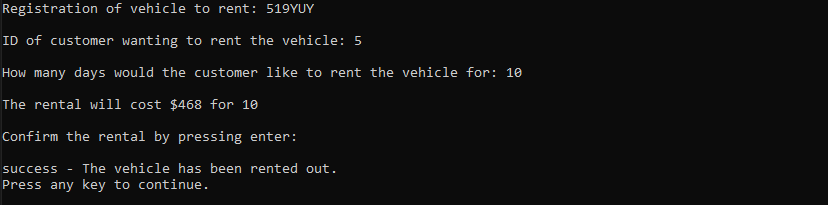
**Rental Search**

This operation allows the user to search for available rentals and displays them in the same format as display fleet. The user can press enter without any input and be given all available rentals, or they can provide a search query to return a more refined set of vehicles. Some queries that would work are Red OR Blue, Petrol AND Manual, 4 AND Diesel OR Manual. AND and OR must be typed in capitals and search filters need to match the properties exactly.



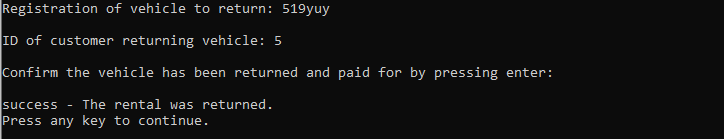
**Rent Vehicle**

In order to rent a vehicle, the user must supply a registration of vehicle to rent, id of customer wanting to rent and the number of days for rental. The cost of the rental for that number of days is displayed and the user can confirm the rental.



**Return Vehicle**

When a rental is being returned the registration and customer id must be supplied. The user is asked to confirm that the rental has been returned and paid for.



**Rental Report**

This operation displays a table of all rentals, with registration of vehicle being rented, ID of customer rented that vehicle and daily cost of that rental vehicle. If no rentals exist, then the user is told so.





**Error Handling**

The program has been designed to be able to handle most expected or reasonable errors. This includes:

* Non-numeric inputs where numeric values are expected
* Unexpected inputs that don’t match the required format, such as supplying 1A1A1 for a vehicle registration when something like 123ABC is expected, or giving the date 40-30-1500 where a date such as 31/12/2001 is expected
* Blank input when a value is required
* A given value needs to be unique and is already being used by another vehicle, such as registration
* A specific value input is expected but an unknown value is supplied, such as supplying yeah for sunroof when yes or no is expected, or M is given for gender but male, female or other is required

The program has tried to accommodate for errors and supply the user with the error and how they could fix it. In some cases, the user may cancel a process by instead of confirming by pressing enter, they enter a value, this is considered purposeful and will not be considered an error. Most inputs are intuitive, and it should be clear as to what input is expected however for some cases the expected input is not as intuitive and for these cases the expected input has been shown to the user. For instance, male, female or other are shown as valid, not M or F or O.



