MRRCManagement User Manual

Following the instructions in this manual will ensure successful use of the program. This program has many methods and classes that may not work as expected when initiated in the incorrect way.

## Building

To build the code for the program, run the .sln file in the root folder in Visual Studio. Once open, right-click on the MRRC solution in the solution explorer and select “build”. After this, right-click on the MRRCManagement solution and select “build”. This will automatically happen once the program is run.

## Running

To run the program, select the program and press the f5 key, opening a new window from which the program will run.

## CSV Files

The program stores and accesses data to and from two CSV files:

* Customers.csv
* Fleet.csv

These files are stored in the Data folder. Customers.csv stores the customer information data and fleet.csv stores the vehicle information data. Rentals.csv stores the information data pertaining to the renting of each vehicle (which vehicle is rented, and by who). **The user can enter their own file paths in the command line arguments of the MRRC class. Right click on the class and select ‘properties’ to enter the path. Enter each argument inside double quotes (“ “) and separated by spaces.**

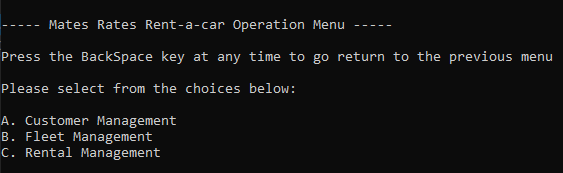
# Program use

This program is used for the management of customers and vehicles in a vehicle rental system. The user navigates through subsets of menus depending on their choices. The menus each have several choices that correspond to methods in the program (e.g. selecting “Delete Customer” in the customer management menu will initiate the Menu.Delete\_Customer method). These methods will create other menus, navigating through menus until the user has completed the desired task. The user can at any time (beside some areas of the creation and modification of customers and vehicles) press the BACKSPACE key to return to the previous menu and the ESC key to formally exit the program entirely. The areas in which the user does not have access to these functions have been noted with a message in that menu:



# Main Menu

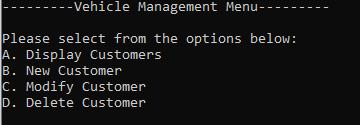
Located in the “UI” class, the main menu is what is immediately displayed as the program is run. It is the point from which all sub-menus can be navigated to.



Inputting a, b, or c (upper or lower case) will proceed to the corresponding sub-menu. Only one letter needs to be inputted, and there is no need to press ENTER, as the program only looks for a one-character input.

# Customer Management Menu

Selecting ‘A’ displays the customer management sub-menu, displaying, similar to the main menu, four options, that initiate a corresponding customer management method:



Inputting a choice takes you to a sequence of prompts relating to the chosen method.

# Display Customers

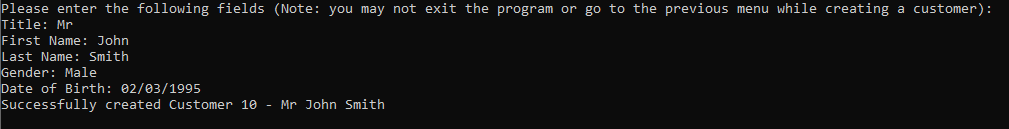
Selecting ‘A’ for Display Customers displays a table of each customer in the CRM:



This table updates dynamically based on the amount of characters in the largest entry of each column (an empty space character is put after the largest entry for formatting purposes).

# New Customer

Selecting ‘B’ for New Customer takes the user to a series of prompts relating to each aspect of a customer (seen above in the display customers table). As these prompts are expecting more than a single character, the user cannot exit out of the program formally, or press BACKSPACE to return to the previous menu during this time (this is noted at the top of the screen). A standard customer creation sequence will look as followed:

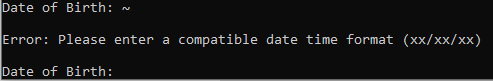


As seen above, the program requests input in the form of strings. These strings are checked against different evaluation methods and deemed either valid or invalid. If deemed invalid, an error message is shown. These are as follows (to account for different languages, no validation is made on the title, first name, and last name fields).

* Gender:



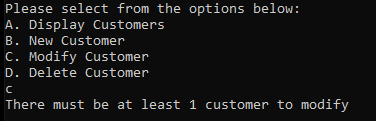
* Date of Birth:



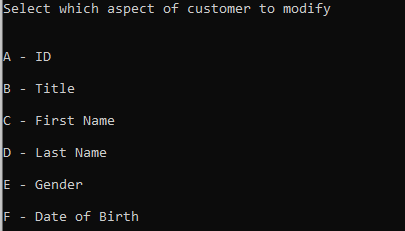
These error messages do not stop the sequence of prompts but appear and display the current prompt again. Once the program has valid inputs, a new object of the CRM class is made, under a temporary name “customer”. This object is also given a unique ID number one higher than the highest current one in the list of customers such that no two customers have the same ID number. After successfully creating a customer, the program displays a message conveying this, then returns to the customer management main menu.

# Modify Customers

The Modify customers function allows the user to select an existing customer and modify any number of aspects of it. If no customers exist in the program, the following error message is displayed:



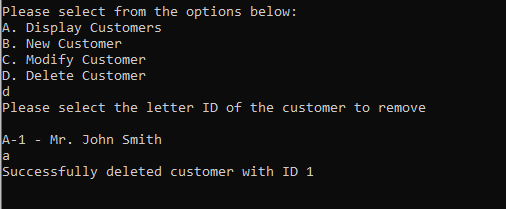
If there is a customer to be modified, the program asks for a letter input corresponding to each customer (A for ID 1, B for ID 2, C for ID 3, etc). Following this, another prompt is requested with letters corresponding to each of the customers aspects seen below:



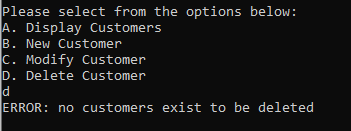
Selecting one of these aspects, the program then requests a string input signifying what the user will change it to, after validating if these are correct (the same validation seen in create\_customer method), the input is turned into its proper format and used to modify the customer. Each aspect of the customer will have different validation methods, which will reject the user’s input if deemed invalid (e.g. ID must be a positive integer that is unique to any of the existing customers). The table of customers is then displayed for the user to see their modification before the program returns to the customer management menu.

# Delete Customer

Selecting the delete customer option allows the user to delete any number of chosen customers. This function prompts a letter input corresponding to each customer and removes the chosen customer from the list of customers, as well as from the csv file of customers. Shown below is the normal sequence for the Delete Customer method:

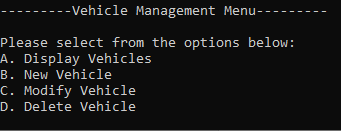


If there are no customers to delete, the following error message is displayed before returning to the previous menu:



# Vehicle Management Menu

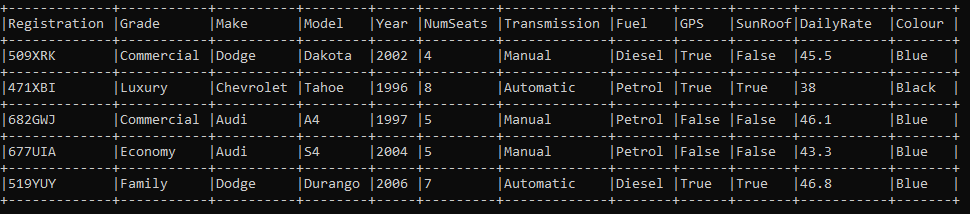
Selecting ‘B’ from the main menu takes the user to the vehicle management sub-menu. This menu layout is very similar to the customer management menu:



Selecting A, B, C, or D takes the user to the corresponding sub-menu.

## Display Vehicles

Selecting ‘A’ for display vehicles displays a list of each vehicle in a tabular format:

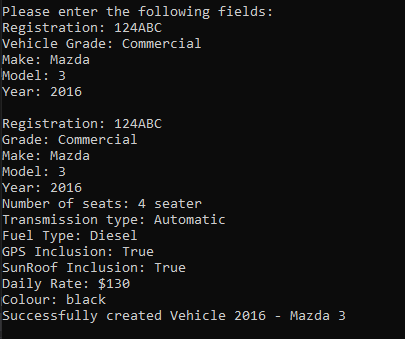


(If this table is not formatting correctly make the window full-screen and display it again. This is an issue with the window and how text is automatically wrapped)

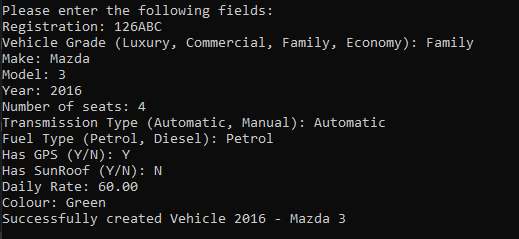
As with the customer management display vehicles function, this method displays an error to the screen if no vehicles exist to be displayed.

## Create Vehicle

Selecting this option prompts the user to either create a vehicle using default values for some aspects or create values for each aspect of the vehicle. Should the first option be chosen, the user will only need to enter the fields for every aspect up to Year (Where the first 5 fields are ones entered by the user):



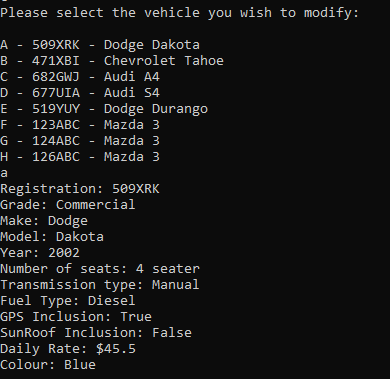
The vehicles aspects are based on the vehicles grade (e.g. luxury cars will have a higher daily rate). Choosing the fully intensive aspects settings looks as followed:

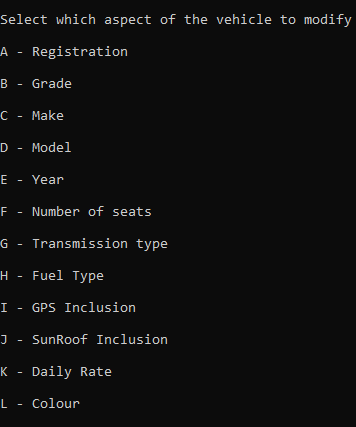


Every aspect in this sequence was chosen by the user. Most aspects in this sequence require validation before being formally entered. Ending this method causes the csv file of vehicles to be updated with the new vehicle. The vehicle is also added to the in-app list of vehicles.

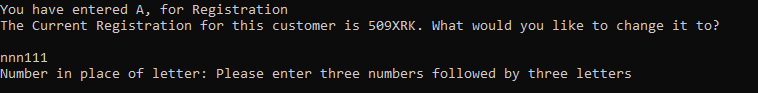
## Modify Vehicle

Selecting this option allows the user to modify a vehicle of their choice in the same format as the Modify Customer method.

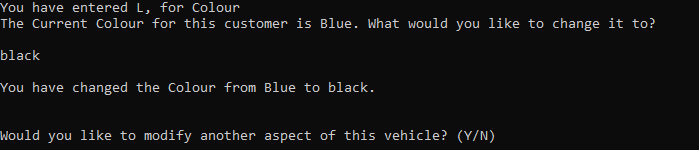




Entering a letter will cause the method to display the current value of the corresponding aspect, and prompt the user for what they would like to change it to:



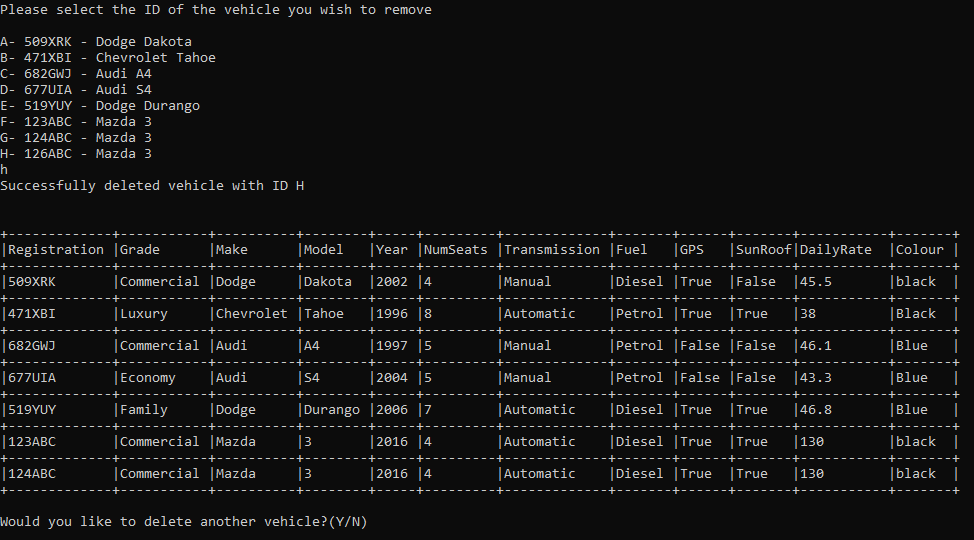
In the above case, the user has entered A for registration, then entered an invalid registration format, causing the error message to be displayed. After this, the program returns to the previous menu, and prompts for a modification again. A successfully validated modification will display the following, before asking the user if they would like to use the method again or return to the vehicle management menu:



This method updates the csv file of vehicles, as well as the in-application list of vehicles, with the modification.

## Delete Vehicle

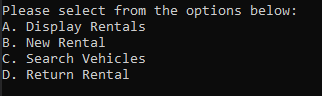
This method prompts the user for a letter corresponding to the vehicle they wish to be deleted, before displaying the vehicle table and asking if they would like to run the method again.



This method removes the chosen vehicle from the fleet.csv file and the in-app list of vehicles.

# Rental management menu

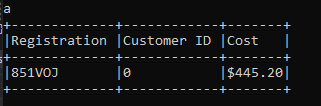
Selecting ‘C’ on the main menu takes the user to the rental management sub-menu:



Selecting ‘A’, ‘B’, ‘C’, or ‘D’ will take the user to the corresponding sub-menu.

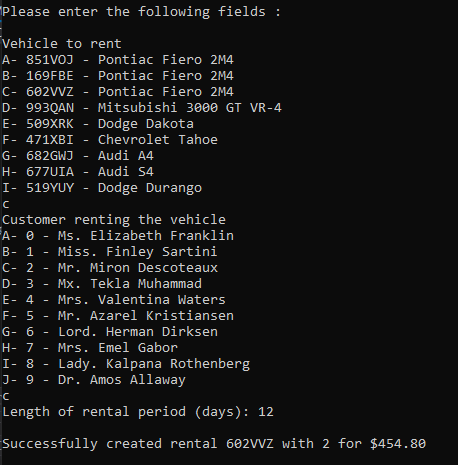
## Display rentals

Selecting ‘A’ in the sub-menu shows a list of rentals in a table format. If there are no rentals to display, a message is shown describing this instead.



## New Rental

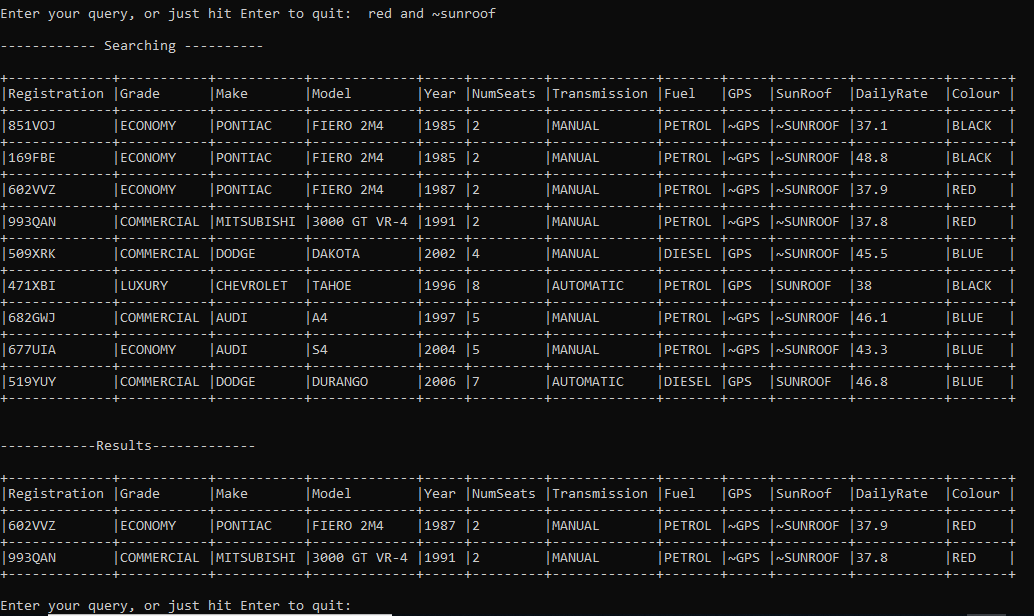
Selecting ‘B’ in the sub-menu directs the user to the new rental menu, where they can create a new rental of a car by a customer of their choice. An example of the process is shown below:



The user is queried for a vehicle to rent from the list of vehicles, then for a customer from the list of customers. If the car is being rented, or the customer is renting a vehicle already, the user is presented with an error message, and asked to select the option again. The length of the rental period is then asked for, and this time is used to calculate the total cost of the rental (based on the vehicle’s daily rate).

## Search vehicle

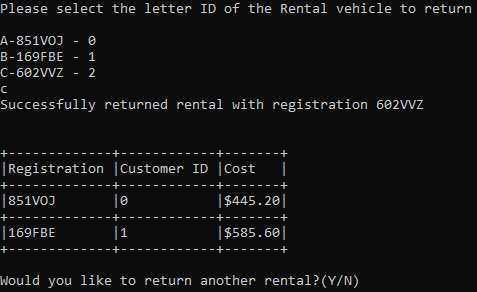
Selecting ‘C’ on the sub-menu takes the user to the search menu, where the user can search the list of vehicles for certain aspects such as colour and GPS inclusion.



Above is an example of a simple search for a red car with no sunroof. Sunroof and GPS inclusion can be searched for by entering ‘GPS’ or ‘Sunroof’, or, searching for cars without a GPS or Sunroof can be done by entering ‘~GPS’ or ‘~Sunroof’. This is explained in a message displayed when the user begins the search process. Complex queries using ‘and’ and ‘or’ and parentheses can also be done, with appropriate error messages being displayed for unbalanced parentheses or operators being used without subsequent queries afterwards (e.g. ‘red AND ‘).

## Return vehicle

Selecting ‘D’ in the sub-menu will take the user to the return vehicle sub-menu, where the user is prompted for a rental to return.



An updated list of rentals is then shown to the screen, and the user is prompted to return another rental.