RMIT / OUA
COSC2288 / COSC2684/ CPT222
Assignment 1 Online Booking System for Vehicle Rentals
Jack Harris | s3893749@student.rmit.edu.au

How to compile:

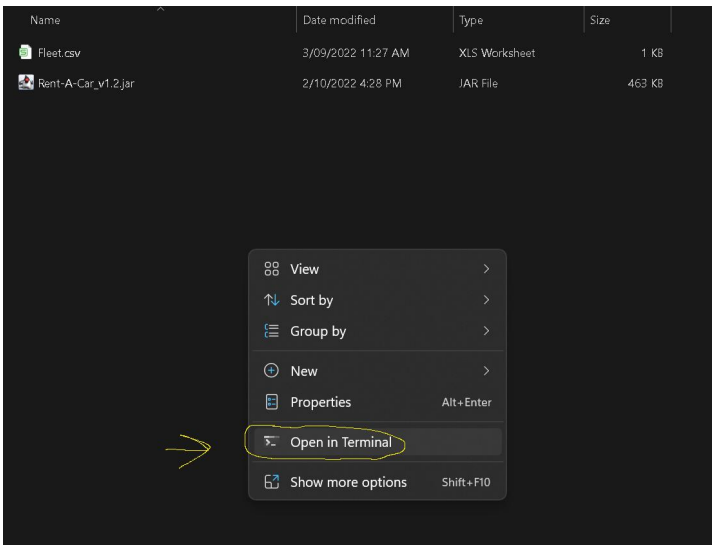
To compile the source code into runnable code, ensure you are in the folder containing the "src" directory, then enter the following command:

```
javac src/com/jackgharris/cosc2288/mycar/core/*.java
src/com/jackgharris/cosc2288/mycar/models/*.java
src/com/jackgharris/cosc2288/mycar/controllers/*.java
src/com/jackgharris/cosc2288/mycar/views/*.java
src/com/jackgharris/cosc2288/mycar/enums/*.java
src/com/jackgharris/cosc2288/mycar/services/*.java
```

How to run:

As the program uses a standard java legalized package name "com.jackgharris.cosc2288.mycar" I am currently unsure how to manually run this from the commandline, running programs with only one main package is easy how ever I ran into extreme difficulty running this one. Issues running the program are not present when running the jar file, jar files can also be unzipped and decoded to validate that the code inside is the same as a source code provided as part of my submission.

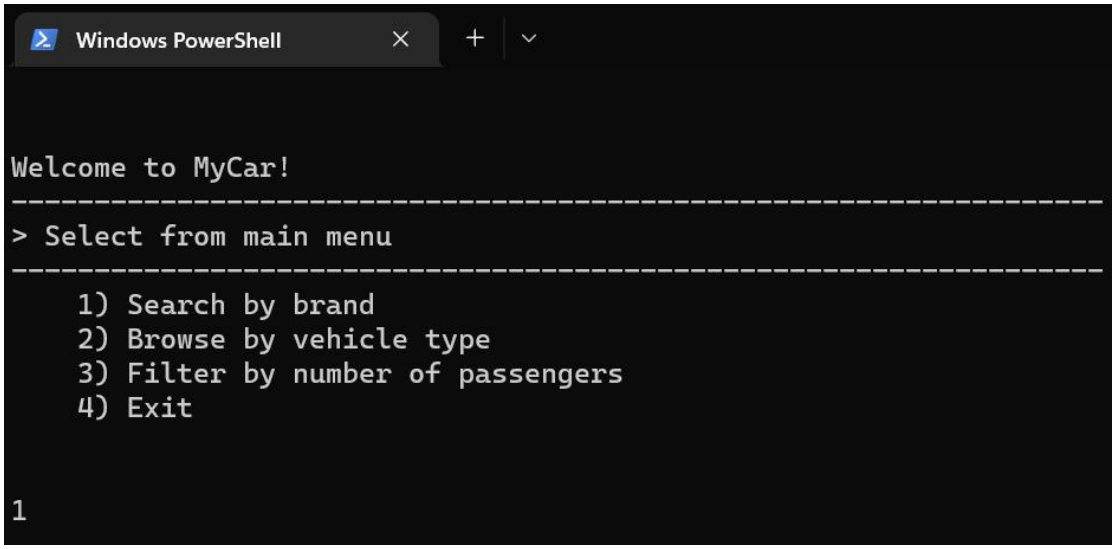
To run the program simply head to https://github.com/s3893749/RMIT-Further-Programming-A1/tree/master/builds/v1.2 and download both the contains jar file and csv file. Once you have them downloaded place them in a folder and then right click to open a windows terminal at that location.



Once you have the terminal open then simply type the following command to start the program:

```
java -jar "Rent-A-Car_v1.2.jar"
```

You will now see the program start! Its important to note that you must run the java program from the directory that contains both the jar file and the “Fleet.csv” file. In the event the jar is run from a different folder location then the “Fleet.csv” will not be detected and you will see no results when searching for vehicles.



You can now use the program via the provided menus!

Documentation:

All code is commented with effective justification, in addition to this I have created a number of documentation md files that are hosted in the GitHub repository, please refer to these in conjunction with the code commenting for understanding how specific methods or classes work.

GitHub:

The Github link is https://github.com/s3893749/RMIT-Further-Programming-A1, As was the case with Milestone 1, the tutor is assumed to be the marker and currently has access to this private github repository. Note: the repository will not be made public untill marks are released for this assessment.

In the event a different markers requries access please email me at my email address below and I will invite you via GitHub.

s3893749@student.rmit.edu.au

Unit Testing:

The assignment has good unit testing coverage with a total of 66 tests spanning 5 classes with each class mapping to a direct object to test. Unit Testing has not been conducted on application loop objects such as Controllers or Views as this would require some spoofing of the System.in user input and was out of scope for this assignment.

CarModelAndCollectionTests.java -> Tests all methods inside the car model and car collection including static getter methods. (19 UNIT TESTS)

FileServiceTests.java -> Tests all methods located in the FileService class. (2 UNIT TESTS)

RequestObjectTests.java -> Tests all methods relating to the programs request object. (14 UNIT TESTS)

ResponseObjectTests.java -> Tests all methods relating to the programs response object. (22 UNIT TESTS)

ReservationModelTests.java -> Tests all methods inside our reservation model. (9 UNIT TESTS)

Additional Notes:

The feedfoward feedback to clear the screen has been attempted using ANSI escape codes to clear and change the position of the screen. In the development I found that attempting to use “clear” or “cls” was not effective and required further logic to determine what OS was running.

The last and only section of the program that would require further validation is the number passenger selection input at the final step of the booking confirmation, due to time constrains I was not able to constrict the number to be validated agents the car and instead have just allowed it to be validated as a integer only!