Instructions

IUT Lyon 1, SAE C/Algo

Summary

Note to colleagues

If you are the duo which is evaluating this project, please notice that to see the execution time of each operation you need to provide options to the commands. If you don't, the execution time will include the time when the program is asking you for the options. Follow the instructions below or the help command to see the options.

Usage

To run the project, you need to compile it or run it into an IDE. You can run it with Clion or Visual Studio Code. The main concept of this project is the usage of a customers listing. The listing is stored in a file. To manipulate a listing you need to load it from a file (or add by yourself the customers). Then you can perform all actions that you want, but it will only affect the loaded listing. To store your changes you need to save the listing into a file (the same as loading or another). When you start the program, it will ask you the commands that you want to perform :

```
  ~~~ Customers manager ~~~

 => Enter a command : _
```

To exit the program you can enter the command `exit` , `quit` , `stop` , `end` or `q` :

```
  ~~~ Customers manager ~~~

 => Enter a command : exit
```

```
~~~ Customers manager ~~~
```

⚠️ Please notice that there isn't any search command. We have chosen that because we are conviced that the search is a particular case of filter. As you will see in the dedicated section below, the filter command works with regex and the search is nothing else rather than a regex with the full specific searched field. For example if we are looking for the Customer with the name `Alex` and living in the city `LONDON`, we just have to provide the regex `Alex` for the name and `LONDON` for the city.

Here is the list of the available commands :

Help command

Displays the list of the available commands with their description and parameters.

Add command

With this command you can add a customer to the listing. You can perform the command without any parameter, so it will ask you to enter the customer's name, surname, city, phone number and email :

```
 => Enter a command : add
Enter the name of the new customer to add : John
Enter the surname of the new customer : Doe
Enter the city of the new customer : New York
Enter the postal code of the new customer : 11111
Enter the phone number of the new customer : 11.11.11.11.11
Enter the email of the new customer : john.doe@gmail.com
Enter the job of the new customer : student
Customer has been added successfully
```

You can also provide the customer's information using the parameters :

- `name` or `n` for the name

- `surname` or `s` for the surname

- `city` or `c` for the city

- `postal`, `postalCode` or `po` for the postal code

- `phone` or `ph` for the phone number

- `email` or `e` for the name

- `job` or `j` for the job

After a parameter, you need to enter the value of the parameter. Here is an example :

```
 => Enter a command : add -name John -surname Doe -city NEW YORK -postal 11111 -phone 11.11.11.11.11 -email j
ohn.doe@gmail.com -job student
 Customer has been added successfully
```

⚠️ The postal code and phone number must match a specific pattern. If you don't provide the correct pattern, the program will ask you to enter the correct pattern again. The postal code must be an integer between 11111 and 99999 (includes). The phone number must follow the pattern : XX.XX.XX.XX.XX where X is an integer between 0 and 9 (includes).

Delete command

This command allows you to delete a list of customer from the listing.

⚠️ This command will delete all the customer in the listing if you don't give any parameter be careful about that.

You can provide the following parameters to delete a customer (notice that what you provide will be interpreted as regex and will be used to delete all the customers matching with all the given regex) :

- `name` or `n` for the name
- `surname` or `s` for the surname
- `city` or `c` for the city
- `postal` , `postalCode` or `po` for the postal code
- `phone` or `ph` for the phone number
- `email` or `e` for the name
- `job` or `j` for the job

Here is an example :

```
=> Enter a command : delete -name ie
6 customers deleted successfully.
```

This command will delete all customer with the string `ie` in their name from the listing. The program will display the number of customers deleted. If no customer is deleted, the program will display a message :

```
=> Enter a command : delete -name ie
No customer found.
```

Edit command

With this command you can edit a list customer from the listing. To edit customers, the program need to know which customers you want to edit and which information you want to change. If you don't provide any parameter to specify customers to edit, the program will edit all the customers in the listing. You can provide the following parameters to specify which customers you want to edit (notice that they will be interpreted as regex) :

- `sname` or `sn` for the name
- `ssurname` or `ss` for the surname

- `scity` or `sc` for the city

- `spostal`, `spostalCode` or `spo` for the postal code

- `sphone` or `sph` for the phone number

- `semail` or `se` for the name

- `sjob` or `sj` for the job

You can also provide the following parameters to specify which information you want to change :

- `name` or `n` for the name

- `surname` or `s` for the surname

- `city` or `c` for the city

- `postal`, `postalCode` or `po` for the postal code

- `phone` or `ph` for the phone number

- `email` or `e` for the name

- `job` or `j` for the job

If you don't provide any parameter to specify which information you want to change, the program will ask you to enter the information you want to change. For each parameter, you can enter the new value or press `Enter` to keep the old value.

⚠️ If you provide a new value for some fields, the program will edit all the customers (matching with the search regex) with the same values. If you choose to keep the old value, the program will keep old value for each customer independently.

Here is an example :

```
=> Enter a command : edit -sname ie -sname ^Sa -city London
Enter the new name of the customer (enter to keep current value) :
Enter the new surname of the customer (enter to keep current value) :
Enter the new postal code of the customer (enter to keep current value) :
Enter the new phone number of the customer (enter to keep current value) :
Enter the new email of the customer (enter to keep current value) :
Enter the new job of the customer (enter to keep current value) : Student
2 customers edited successfully.
```

Notice that since we provide the parameter `-city` to specify the new value of the field `city`, the program won't ask us to enter the new value for the field `city` for each matching customer. This command will search for all the customers with the string `ie` in their and surname beginning with `Sa`. For each of these, the program will edit the city to `London` and the job to `Student`. The program will display the number of customers edited. If no customer is edited, the program will tell you it.

Filter command

With this command you can filter the customers keeping only the customers matching with regex that you will give. You can provide the following parameters to filter the customers (notice that they will be interpreted as regex) :

- `name` or `n` for the name

- `surname` or `s` for the surname

- `city` or `c` for the city

- `postal` , `postalCode` or `po` for the postal code

- `phone` or `ph` for the phone number

- `email` or `e` for the name

- `job` or `j` for the job

Here is an example :

```
=> Enter a command : filter -name ie
Customer list successfully filtered.
```

Load command

You can load a list of customers from a csv file. There is only one available parameter for this command :

- `file` or `f` for the file name

Here is an example :

```
=> Enter a command : load -f resources/test.csv
Customers successfully loaded.
```

If the file doesn't exist, the program will display a message :

```
 => Enter a command : load -f a.csv
File not found.
```

Save command

You can save a list of customers into a csv file. There is only one available parameter for this command :

- `file` or `f` for the file name

Here is an example :

```
 => Enter a command : save -f resources/test.csv
Customers list successfully saved in file resources/test.csv.
```

⚠️ As said in the introduction, the modification of the customers list will be lost if you don't save it in a file.

Show command

Maybe the most useful command of this program. The show command will display the list of customers. You can perform some action on the listing before displaying it. You can sort it, filter it and display only the

customers which have incomplete data or only the customers which have complete date. This command has following available parameters :

- `filter` or `f` for the field of the filter
- `value` or `v` for the regex of the filter
- `sort` or `s` for the field to sort the customers by
- `revert` or `r` for the descending order of the sort
- `incomplete` or `i` for the customers which have incomplete data
- `complete` or `c` for the customers which have complete data

Here are some examples :

Simple show :

```
 => Enter a command : show
Here is the customers list :

      Name                   Surname                  City              Postal code   Phone
Email                                       Job
1     Adelaide               Pottier-Leleu            MARSEILLE         13015         04.7
0.89.89.12   adelaide.pottier-leleu@server.fr
2     Adrien                 Guillaume                LYON              69008         04.1
8.14.57.97   adrien.guillaume@server.fr
3     Agathe                 Boucher                  BORDEAUX          33000         05.5
3.39.48.61   agathe.boucher@server.fr
4     Lucie                  Sanchez                                                  01.0
1.11.75.95   lucie.sanchez@server.fr                  boulanger
5     Pauline                Marin                    LYON              69004         04.5
7.30.95.93   pauline.marin@server.fr
6     Stephanie              Dufour                   VILLEURBANNE      69100         04.9
8.90.39.16   stephanie.dufour@server.fr               journaliste
7     Thierry                Sanchez                                                  05.1
6.43.80.37   thierry.sanchez@server.fr
8     Virginie               Gaudin                   NANTES            44100         02.4
3.66.83.82
9     Zacharie               Georges                                                  04.4
8.81.83.92   zacharie.georges@server.fr               chef d'entreprise
```

Filtered show :

```
 => Enter a command : show -filter name -value ie
Here is the filtered customers list :

      Name                   Surname                  City              Postal code   Phone
Email                                       Job
1     Adrien                 Guillaume                LYON              69008         04.1
8.14.57.97   adrien.guillaume@server.fr
2     Lucie                  Sanchez                                                  01.0
1.11.75.95   lucie.sanchez@server.fr                  boulanger
3     Stephanie              Dufour                   VILLEURBANNE      69100         04.9
8.90.39.16   stephanie.dufour@server.fr               journaliste
4     Thierry                Sanchez                                                  05.1
6.43.80.37   thierry.sanchez@server.fr
5     Virginie               Gaudin                   NANTES            44100         02.4
3.66.83.82
6     Zacharie               Georges                                                  04.4
8.81.83.92   zacharie.georges@server.fr               chef d'entreprise
```

Descending sorted show :

```
 => Enter a command : show -sort name -revert
Here is the sorted customers list :

       Name                    Surname                        City               Postal code   Phone
Email                                          Job
1     Zacharie              Georges                                                             04.4
8.81.83.92   zacharie.georges@server.fr               chef d'entreprise
2     Virginie              Gaudin                         NANTES             44100         02.4
3.66.83.82
3     Thierry               Sanchez                                                           05.1
6.43.80.37   thierry.sanchez@server.fr
4     Stephanie             Dufour                         VILLEURBANNE       69100         04.9
8.90.39.16   stephanie.dufour@server.fr               journaliste
5     Pauline               Marin                          LYON               69004         04.5
7.30.95.93   pauline.marin@server.fr
6     Lucie                 Sanchez                                                           01.0
1.11.75.95   lucie.sanchez@server.fr                  boulanger
7     Agathe                Boucher                        BORDEAUX           33000         05.5
3.39.48.61   agathe.boucher@server.fr
8     Adrien                Guillaume                      LYON               69008         04.1
8.14.57.97   adrien.guillaume@server.fr
9     Adelaide              Pottier-Leleu                  MARSEILLE          13015         04.7
0.89.89.12   adelaide.pottier-leleu@server.fr
```

Incomplete show :

```
 => Enter a command : show -incomplete
8 customers have missing data (88.89%).
Here is the list of customers with missing values :

       Name                    Surname                        City               Postal code   Phone
Email                                          Job
1     Adelaide              Pottier-Leleu                  MARSEILLE          13015         04.7
0.89.89.12   adelaide.pottier-leleu@server.fr
2     Adrien                Guillaume                      LYON               69008         04.1
8.14.57.97   adrien.guillaume@server.fr
3     Agathe                Boucher                        BORDEAUX           33000         05.5
3.39.48.61   agathe.boucher@server.fr
4     Lucie                 Sanchez                                                           01.0
1.11.75.95   lucie.sanchez@server.fr                  boulanger
5     Pauline               Marin                          LYON               69004         04.5
7.30.95.93   pauline.marin@server.fr
6     Thierry               Sanchez                                                           05.1
6.43.80.37   thierry.sanchez@server.fr
7     Virginie              Gaudin                         NANTES             44100         02.4
3.66.83.82
8     Zacharie              Georges                                                           04.4
8.81.83.92   zacharie.georges@server.fr               chef d'entreprise
```

Complete show :

```
 => Enter a command : show -complete
1 customers have no missing data (11.11%).
Here is the list of customers without missing values :

       Name                    Surname                        City               Postal code   Phone
Email                                          Job
1     Stephanie             Dufour                         VILLEURBANNE       69100         04.9
8.90.39.16   stephanie.dufour@server.fr               journaliste
```

⚠️ Please notice that the filters and sorts perform in the show command doesn't affect the loaded customers list. If you perform a sorted show and then a simple show, you won't get the same output (except if the list is already sorted).

Sort command

This command will perform a sort on the customers. To do it, the program need to know which field you want to sort the customers by. You may also want to sort the customers descending order. You can provide the following parameters to specify this information :

- `field` or `f` for the field to sort by

- `revert` or `r` for the descending order

If you don't provide the field parameter, the program will ask you to enter it. Notice that if you provide a field which doesn't exist, the program won't display any specific message but the customers list won't be changed.

Here is an example :

```
 => Enter a command : sort -f name -r
Customer list successfully sorted by name.
```