Rohith Ramesh

1001518031

Input Validation Assignment

This project is written in python 3 using pycharm IDE. It represents a phone storage system which reads,deletes and writes to a persistent storage system(in this case a CSV file called records.csv). It writes data based on given set of rules and format. The program deletes data by using name and also by using phone number. It also lists the available data entries on the data store. The audit function works in linux platform and enters log entries with timestamp if the program is run by a privileged user.

The Regex module of python is used for input validation. The regex functions(check\_name and check\_phone) validate the given user input(Name,Phone Number). For the regex check to pass the user has to input a valid name and phone number according to the given constraints and formats. If either of the check for name or phone number fails the program will throw an message stating to enter the wrong data correctly(name or phone number) and exit with a code of 0.

**Compilation details :**

* Technology used Python 3
* Download and install pycharm.
* Run the main.py file by ckicking the run(play) button on the IDE.
* Follow the onscreen options(ADD,DELETE,LIST,EXIT)
* For Audit function- run the code using pycharm in linux environment as a root user(EUID=0) and the audit log will be written

**Code Description :**

The code performs 5 basic operations, add,delete,list,exit and audit log. The functions are described below

**ADD**- the add operation is performed by the create() function in the code. The function takes a name and phone number from the user and calls the check\_phone and check\_name functions to check if the phone and name are according to the given format. If yes then the data is written to the csv file. If no then an appropriate error message is displayed.

**DELETE**- the delete function is split into two sub functions delete1 and delete2. The delete1 is used to delete a record based on the name. if the record is not present then an error with exit code 1 is thrown and the program stops execution. The delete 2 function performs the same functionality but with the key search element as the phone number. The code behaves in a similar way throwing error withs appropriate exit codes when we try to delete a record which doesn’t exist.

**LIST**-The list function lists the given records in the data store. If no records are present it displays an empty string(no display) and the control goes back to the main function loop.

**EXIT**- the exit option exits the code and stops execution

**Audit**-The log option works only with linux environment. This does not stop the code from being executed in a windows environment. The log function checks if the EUID of the user is 0 if so then log.txt is written with the name of the person who has been added/deleted , their RUID and the timestamp. A known limitation to the Audit function is that it only writes to the log when the delete operation is performed by name and does not write to the log when the delete operation is performed using phone number.

**Assumptions:**

The database already has records present in it if we choose to run the List operation first.

The database file records.csv is already present in the same folder as that of the code.

The names and phone numbers entered also match the rules given according to the regex

**Advantages**:

* The code can be run on any environment in terms of operating system. If the operating system is not Linux the code will still execute but the log() function which writes to the persistent datastore will not work. This is done by checking if the operating system is windows or not using the python module “platform”.
* The code is light weight and easy to move into different systems without having to install any expensive and heavy software.

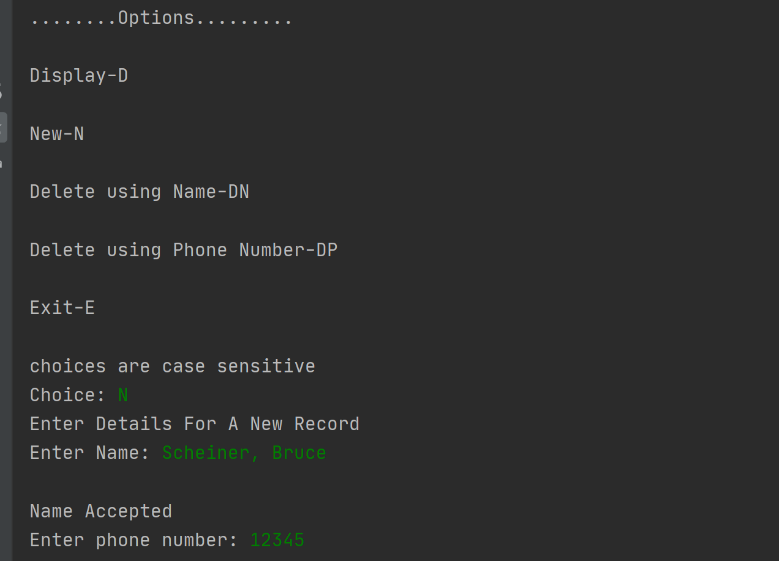
**Disadvantage:**

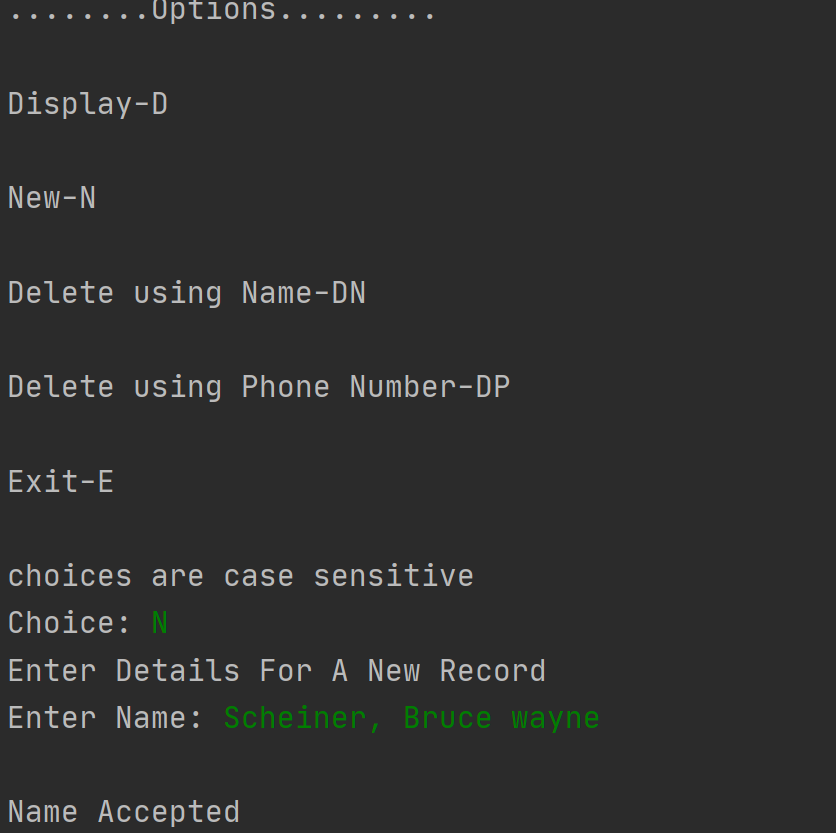
* The filtering of numbers is limited to the scope of this project. Other international or numbers with a different format will not work

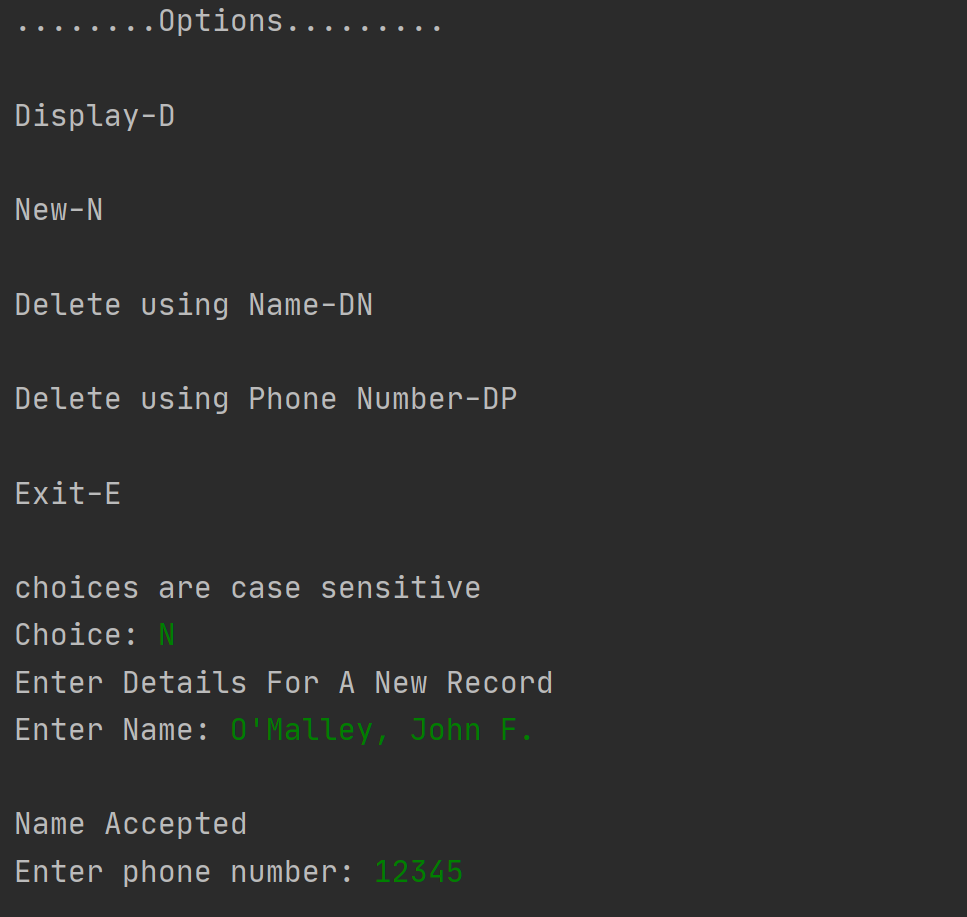
**Screenshots**:

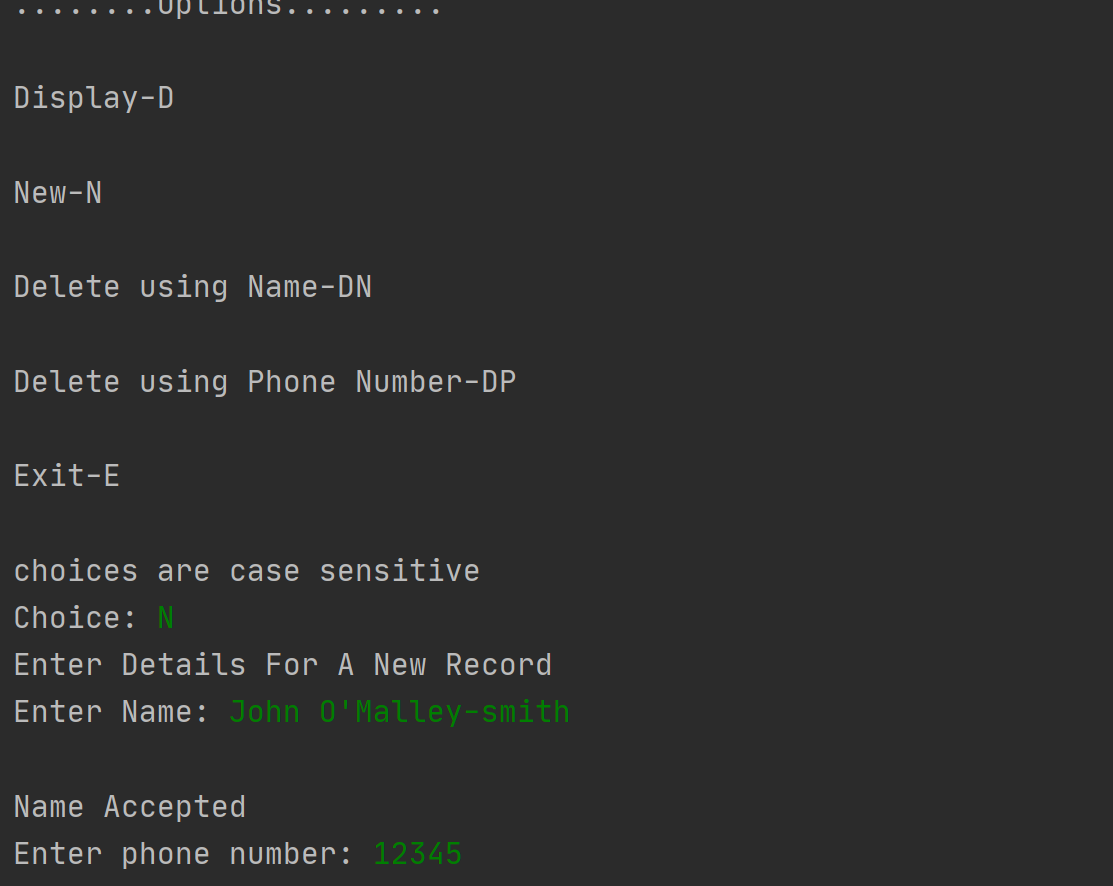
**ADD**:

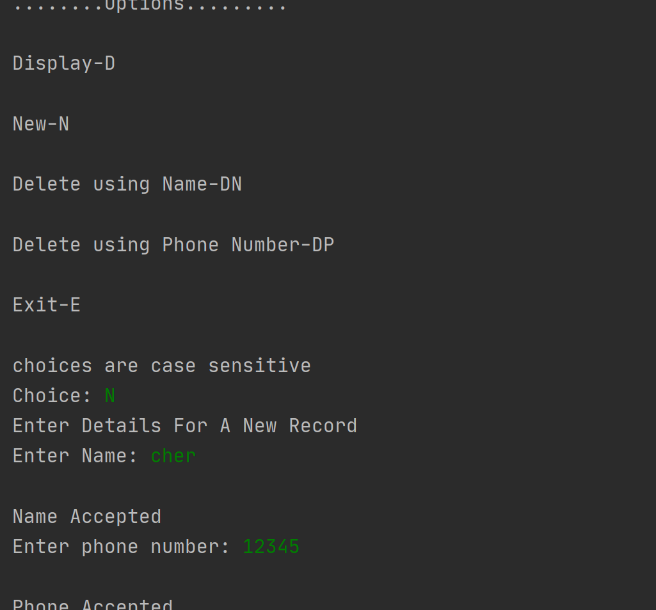


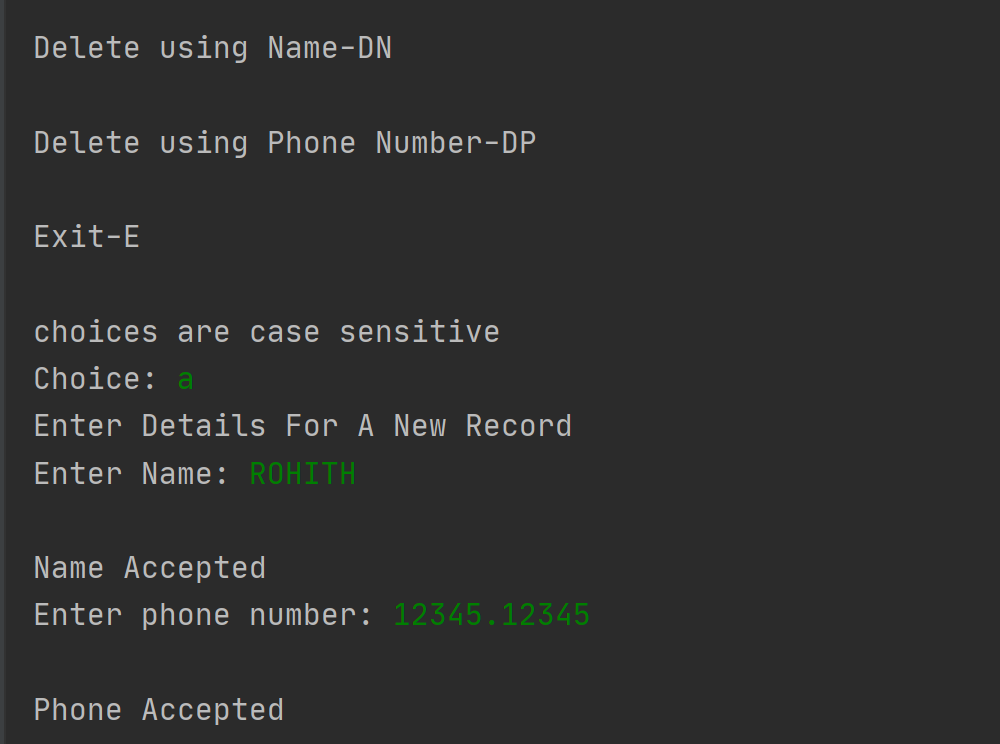
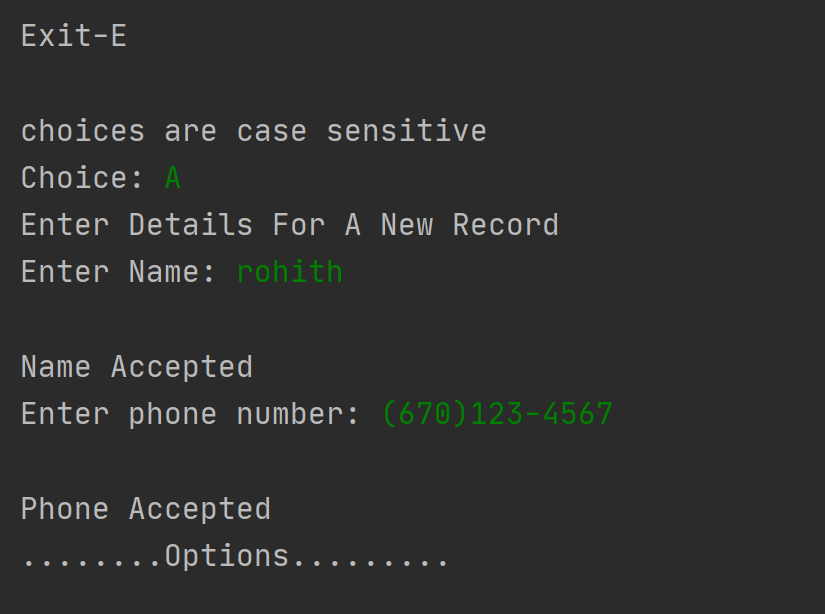
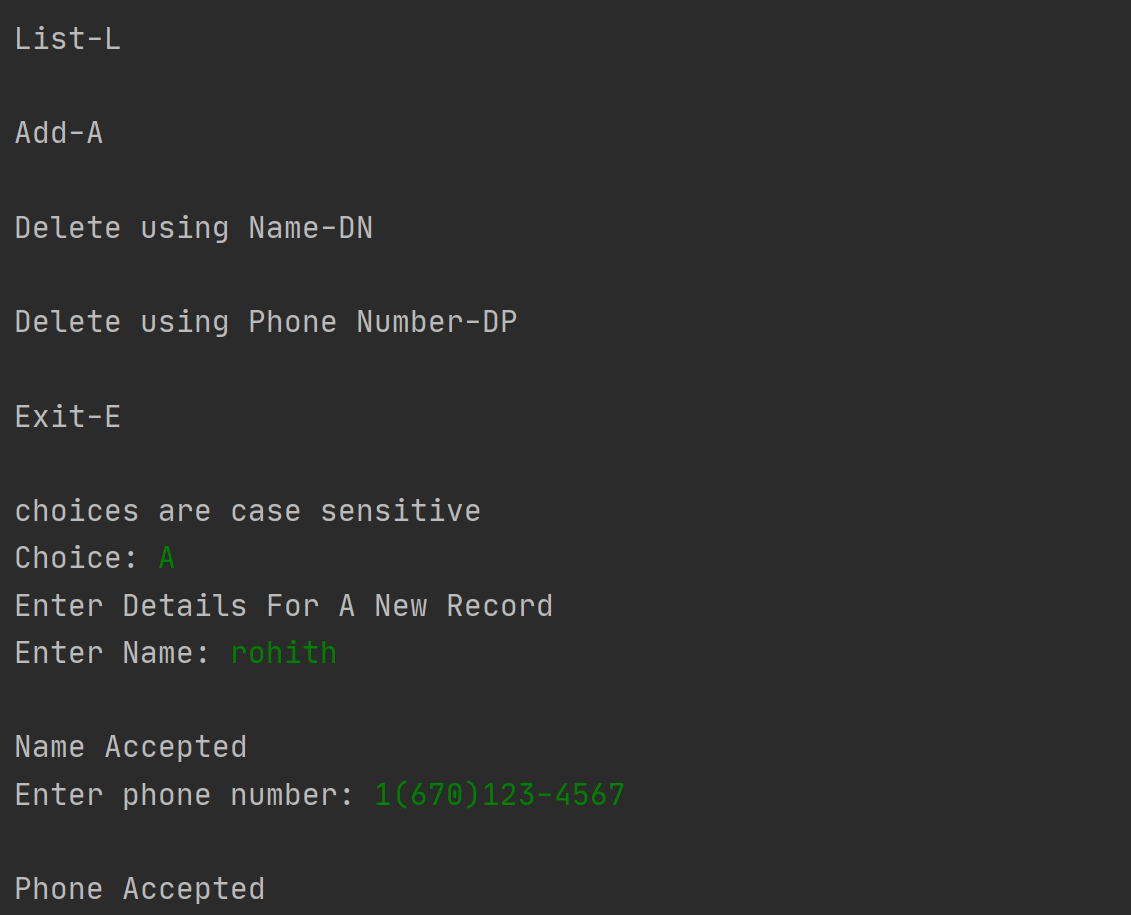




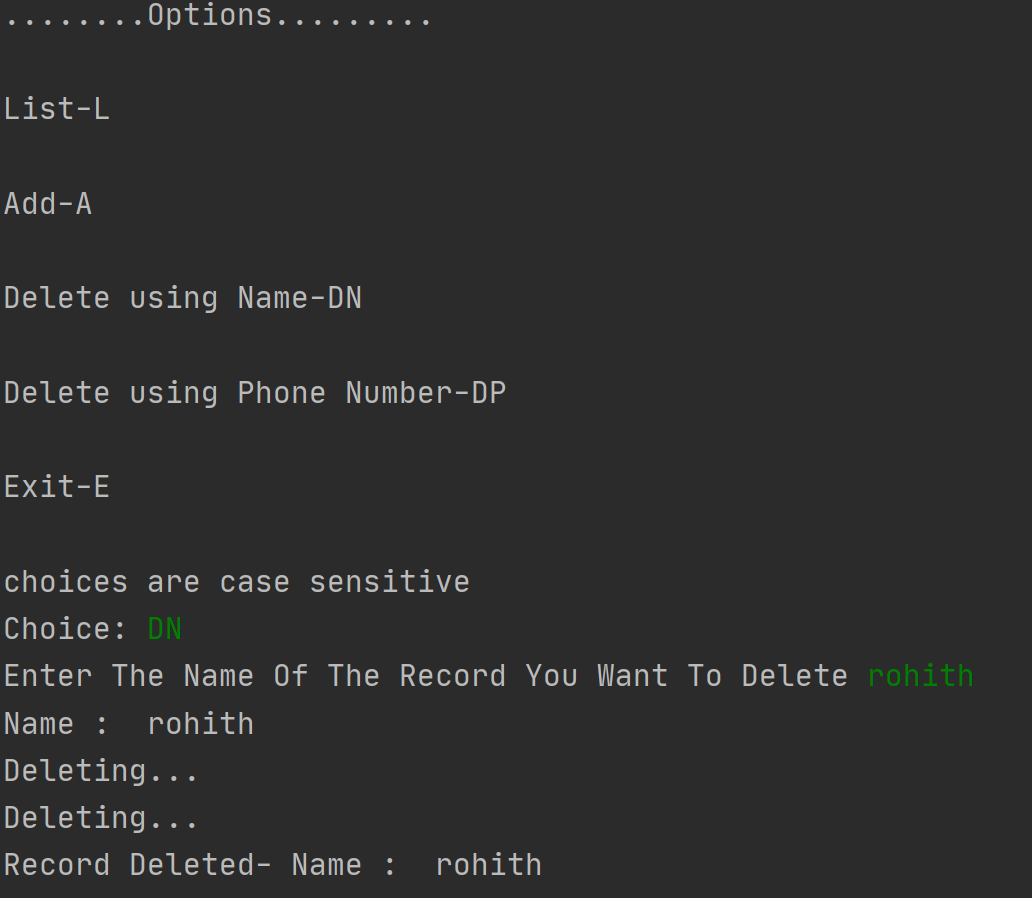




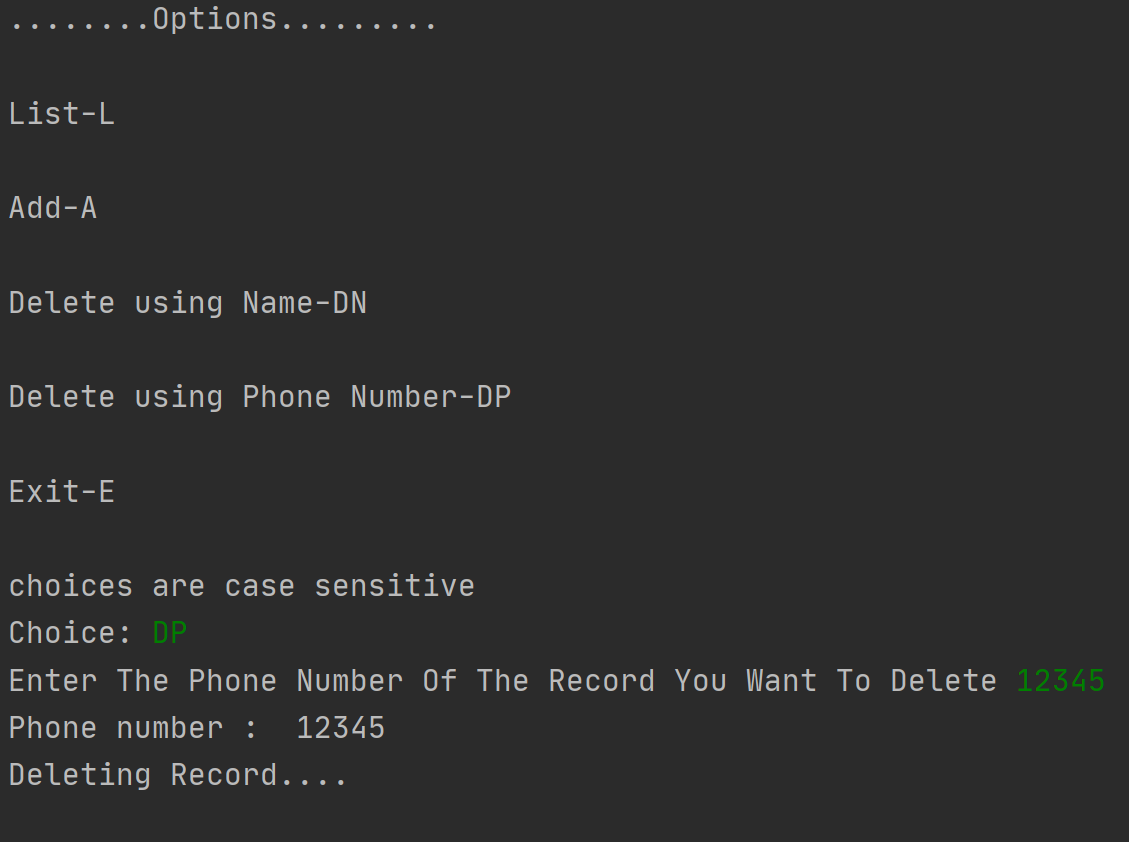




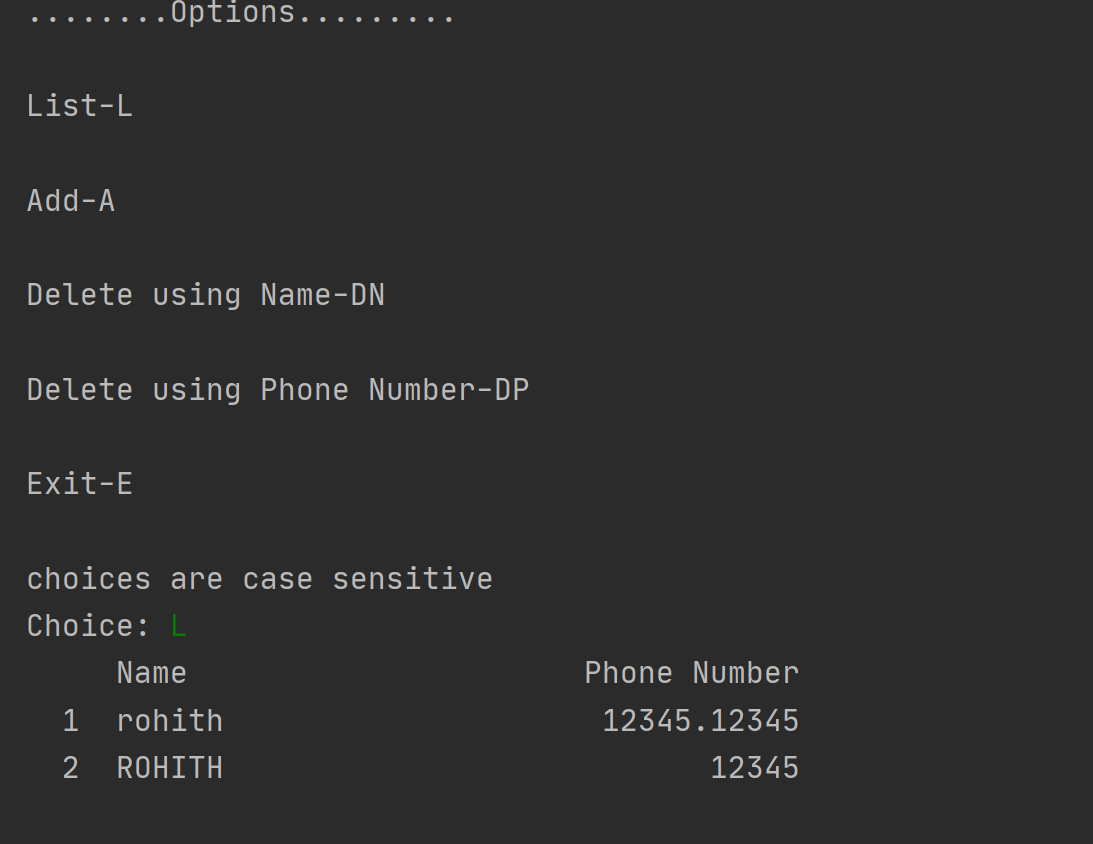
**Delete(Name):**



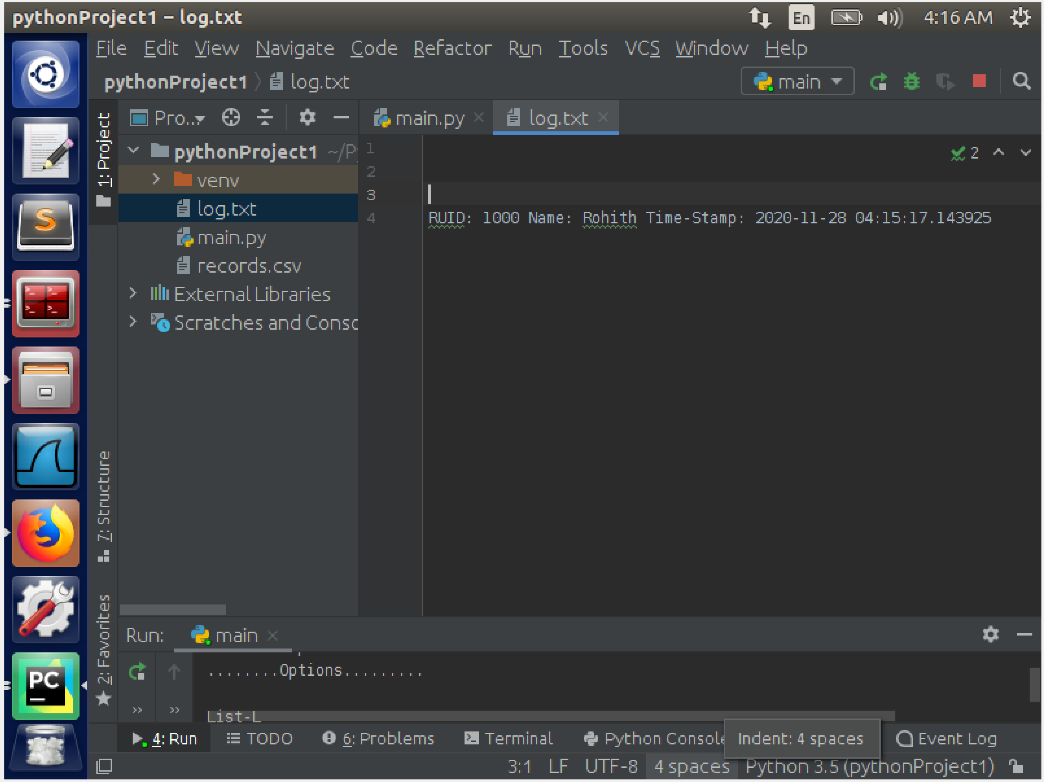
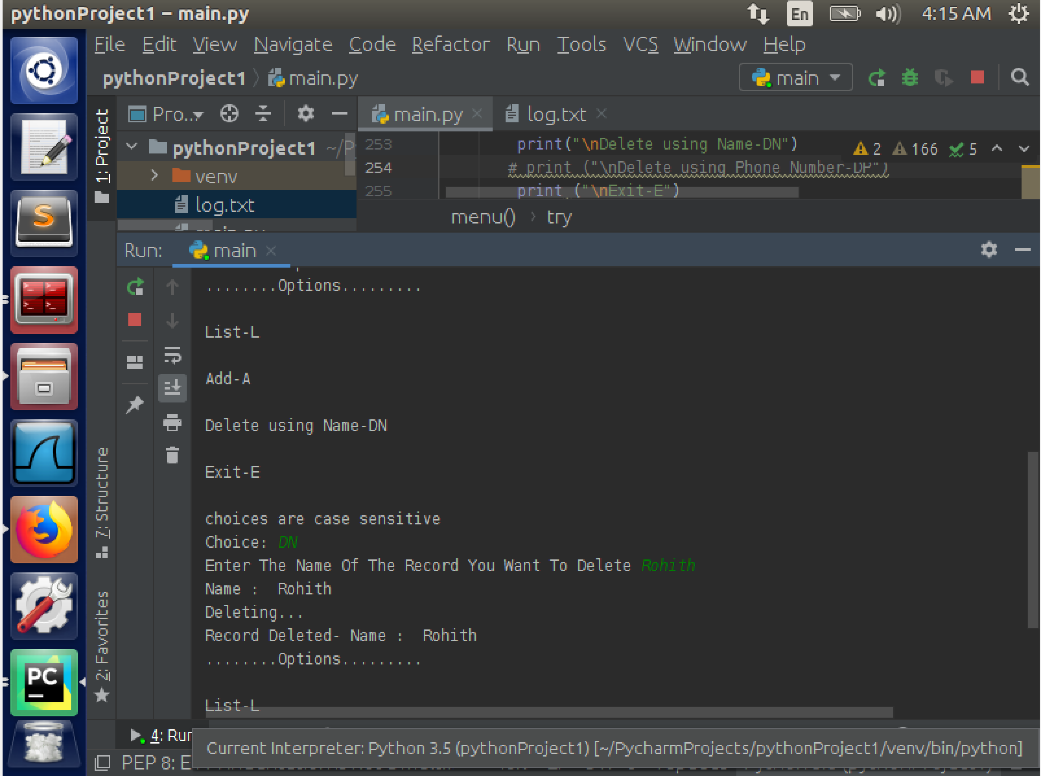
**Delete(Phone):**



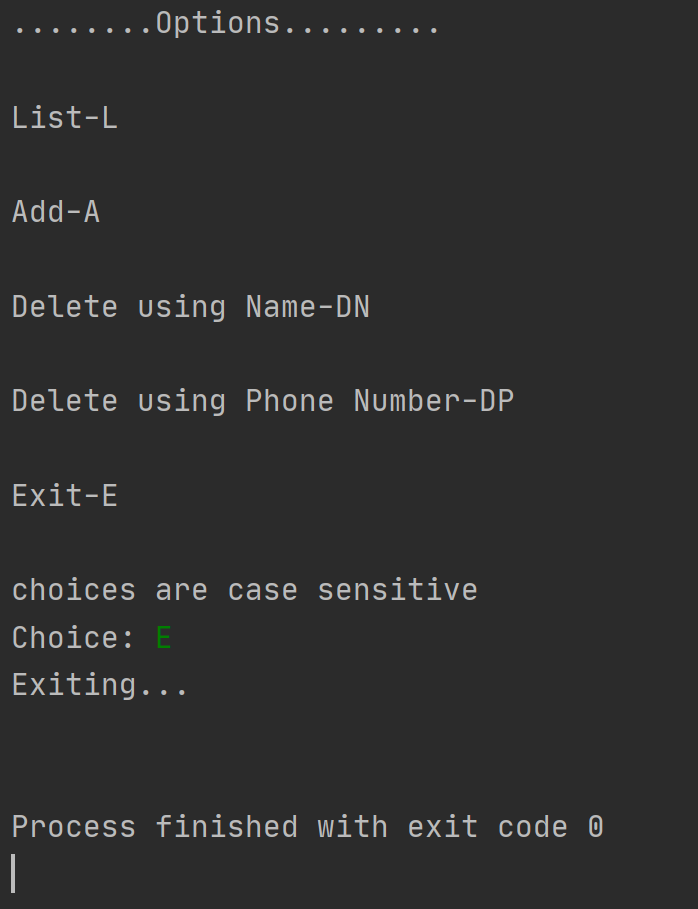
**LIST**:



**Audit log:**



**EXIT**:



**References**:

<https://github.com/Bogro/phone_email_verifier>

<https://stackoverflow.com/questions/16699007/regular-expression-to-match-standard-10-digit-phone-number>

<https://stackoverflow.com/questions/2385701/regular-expression-for-first-and-last-name>