**Assignment 11: Input Validation**

**Submitted to: Prof. Thomas Lawson Jones**

**Submitted by: Amitesh Singh Baghel (1001721186)**

This project is written in python language. The project uses sqlite3 and python3 so make sure you get these two installed before running the project. The project does not use anything additional that needs to be installed. The main file here is called “assignment\_11.py” which imports “api.py” to perform the operations on sqlite3 database. “api.py” imports connect.py.

The assignment us asks us to make input validation program that lets user to ADD, LIST and DELETE the entries present in Table in database. The input validation in this program which accepts valid cases and rejects all other invalid cases is written using regex. When the input is validated the function is called to perform the ADD/DEL operation. The input validation prevents sql injection and makes our program secure.

To Run the Program:-

1. Have Python Installed on your PC
2. Have SQLite3 installed
3. Open terminal
4. Go to the location where the folder exists
5. Type in your terminal “python assignment\_11.py” Hit Enter
6. The instructions on how to run the program will be displayed to you
7. To Add a value:- python assignment\_11.py ADD “<<Name of the Person>>” “<<Phone Number>>”
8. To List values in db: python assignment\_11.py LIST
9. To Delete: python assignment\_11.py DEL “<<Phone-Number>>”

OR

python assignment\_11.py DEL “<<Name>>”

1. When the program will first run, it will create database “phonebook.db” which will have “phonedict” table that will store name and phone number
2. The program will also make a log file phonebook.log which will contain Operation, Values(Name, Phone Number)Given, Result of the operation (Success/Fail), Timestamp.

Below I will be writing detailed description of each task, how the program was implemented, things to improve on.

**Add Operation:-**

The program collects the values given by user using sys.argv and if the second value in the list matches with ADD it goes forward to perform ADD operation. To do that operation it calls “add\_info” function by providing it Name and Phone Number and this function validates Name and Phone Number against the regex for acceptable cases. If it matches then it calls the function “insert” in api.py and pass it name and phone number and this function Inserts the name and phone number using connect.py. The function insert is only called when the regex case matches otherwise the program prints “Input could not be validated you either gave name in incorrect format or phone number in incorrect format”. In case with ADD only name/phone number is given and not both then message “Input given cannot be accepted as more or less arguments are given. Please follow the format” is given. Log entry for each case is done.

**DEL Operation(PhoneNumber/Name):-**

Similar to ADD function DEL function does the regex operation to sanitize the input and make sure no SQL Injection can happen. Here, in case no name/phone is given by user the message “Input given cannot be accepted as more or less arguments are given. Please follow the format” is shown to user. In case the phone number/name is provided to “delete\_info”function. Then regex is used to make sure the input is valid and also identifies if phone number or name is given. In case name is provided and is valid, then the delete\_by\_name function is called which runs the query and returns 1 in case the DELETE Operation is successful, otherwise returns 0. Same way the delete\_by\_phone function works. Based on the response the messages "Entry Successfully Deleted"/"PhoneNumber or User Name provided does not exist" are displayed to user.

**LIST Operation:**

Its simple operation. Here when user provides the value LIST, we directly call the get function in api.py which performs SELECT Query and retrieves all the entries of the table and displays it to user even if it is empty.

**Log:**

Here, we use the logger package in python and for any operation we call the function log and pass sys.argv and “Success” or “Fail”. The log checks for the command ADD/DEL/LIST and number of elements sys.argv contains. Based on the input it logs the response in the logfile along with TimeStamp.

**Assumptions:-**

We have assumed that user does not require GUI and will understand how to run the code.

**Advantages:-**

The regex part of the project is good and protects the db from sql injection attack

The code is light, clean and has medium cognitive complexity

It can run on any operation system.

**Disadvantages:-**

The code is limited to certain cases only. By using regex and making input limited to certain cases only, we have limited the scope of the project.

A certain functionalities are not implemented efficiently and with a person with more expertise could have handled the implementation of certain properties better.

**Screenshots of Operation with logs:-**

**ADD:-**

**Text

Description automatically generated**

**Text, letter

Description automatically generated**

**Text, letter

Description automatically generated**

**Text, letter

Description automatically generated**

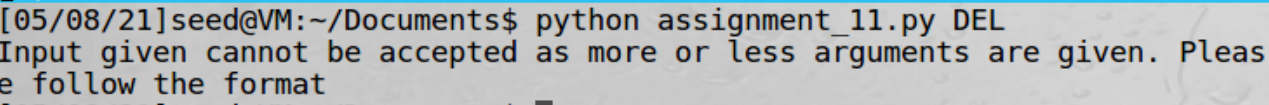
**Text, letter

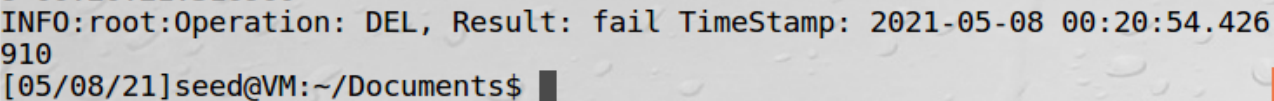
Description automatically generated**

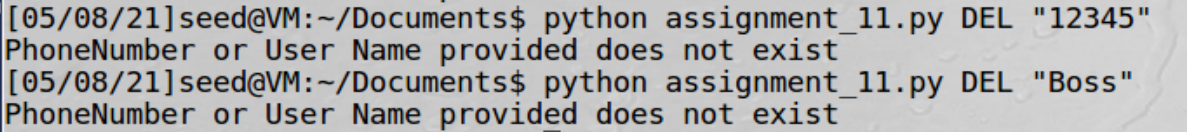
**DEL:-**

**Text, letter

Description automatically generated**

****

****

****

**A picture containing text, people

Description automatically generated**

**LIST:-**

**Text

Description automatically generated**

**Text

Description automatically generated**

**Case Where nothing is passed after assignment\_11.py:-**

**Text, letter

Description automatically generated**