***Computer Engineering Department***

***CMPE 30052 Lab Activity No. 1 – Application of Python List***

***Due on November 16, 2022 3:00PM***

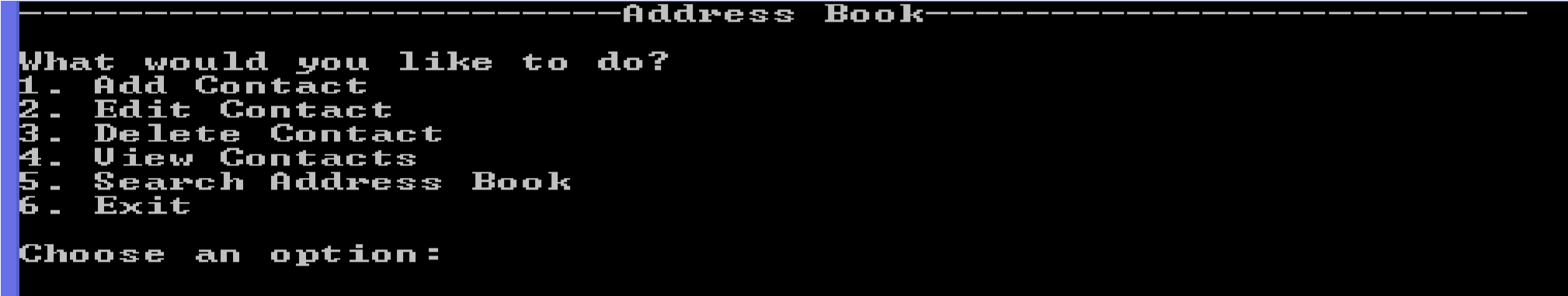
|  |  |  |
| --- | --- | --- |
| Name: Viernes, Michael E. | Date Performed: | November 16, 2022 |
| Course/Yr: BSCOE 2-1 | Date Submitted: | November 16, 2022 |
| Class Schedule 3 -6:00 PM |W | Score: | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |

**LAB ACTIVITY #1 Application of Python List**

**Instructions:**

Create an Address Book Python program that can contain 50 entries. Provide data validation for both data and responses.

Below is a sample output.



1. **Add contact** 
   1. Prompt the user for the first name, last name, address, and contact number. Please use Python Function.
2. **Edit Contact** 
   1. Prompt the user for the entry number he wants to edit. Please use Python Function
3. **Delete Contact** 
   1. Prompt the user to enter the entry number to be deleted.
   2. After deleting a record, all succeeding entries will move forward.
   3. Please use Python Function
4. **View Contacts** 
   1. Display all the entries.
   2. Please use Python Functions
5. **Search address book** 
   1. Prompt the user to search the address book **(a)** **by first name, (b) by last name, (c) by address or (d) by contact number.**
   2. Display all the entries that matched the query. Else, notify the user that the entry doesn’t exist.
   3. Please use Python Functions
6. **Exit** 
   1. Closes the program.
   2. The Main Menu is program also using Python Function.

Lab Challenge (ADDITIONAL BONUS POINTS)

If you can present the project using GUI rendition via Tktinter GUI Library.

**LAB ACTIVITY #1 Application of Python List**

Source Code: <https://github.com/MichaelViernes271/CMPE-30052>

**SCREENSHOTS:**

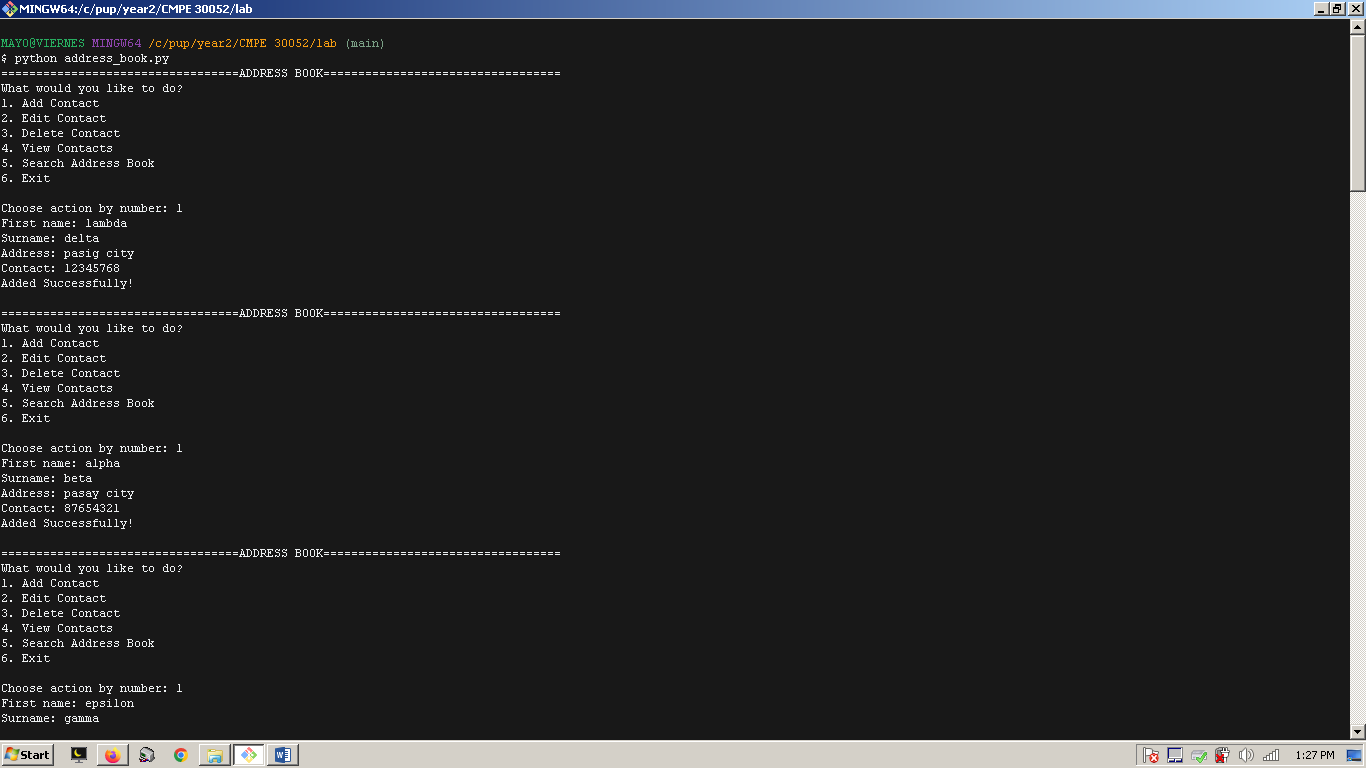


Figure 1. Adding contacts to the address book (a)

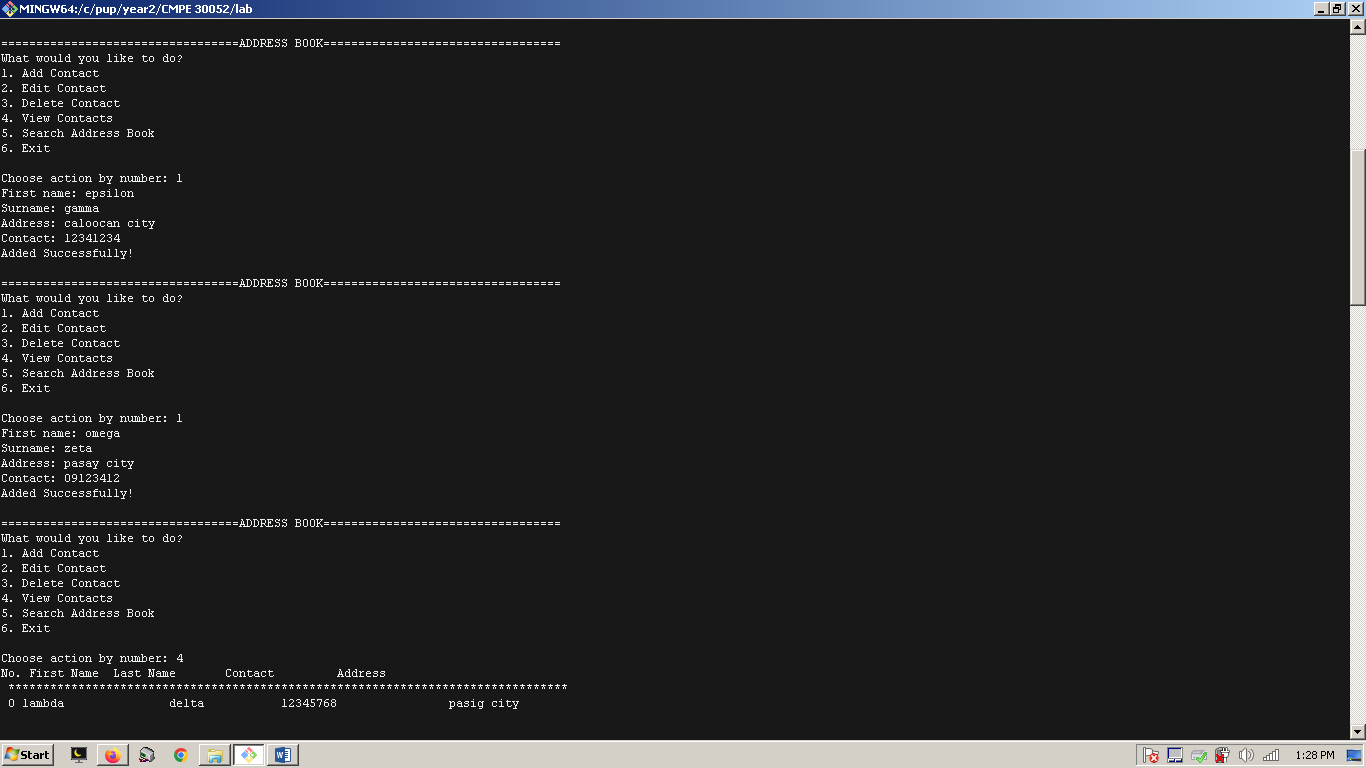
Z

Figure 2. Adding contacts to the address book (b)

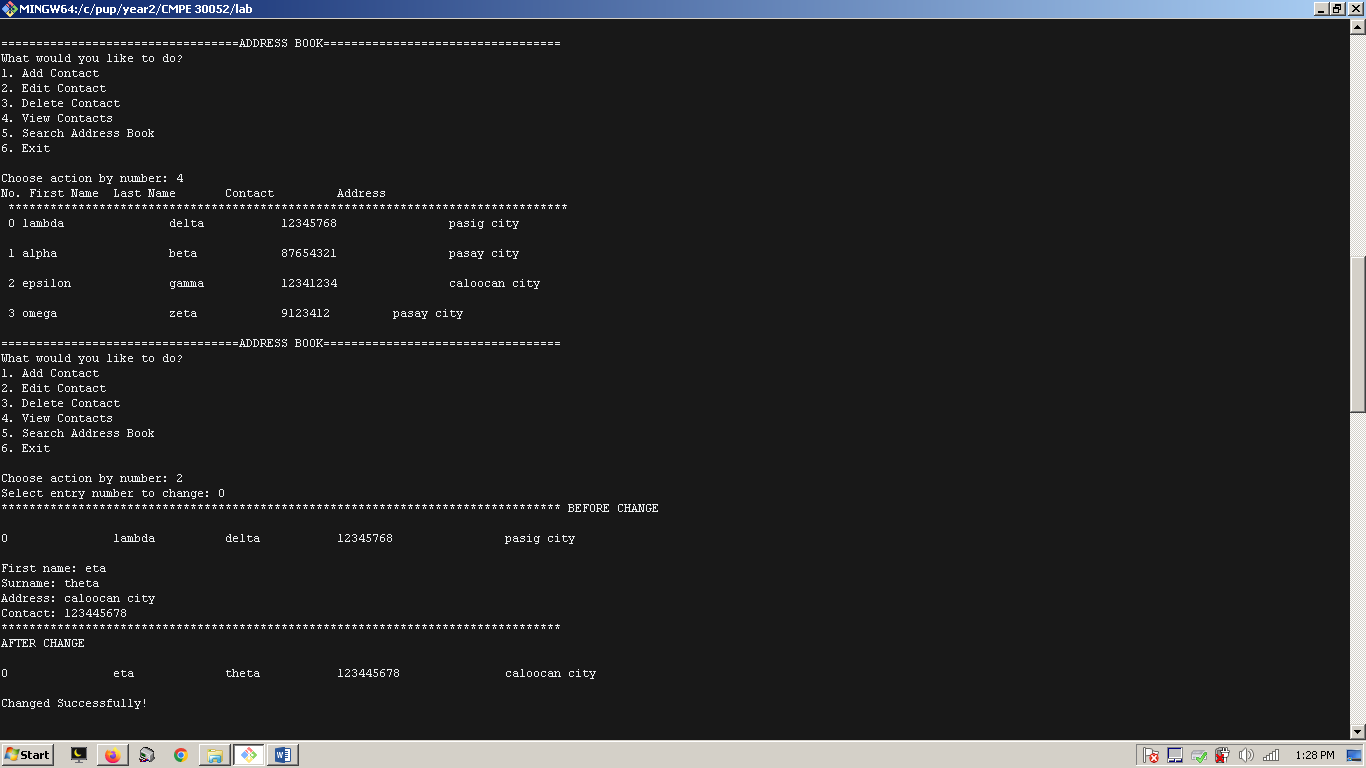


Figure 3. Viewing the contents of the address book. Editing the first entry in the address book, and showing the changes made thereafter.

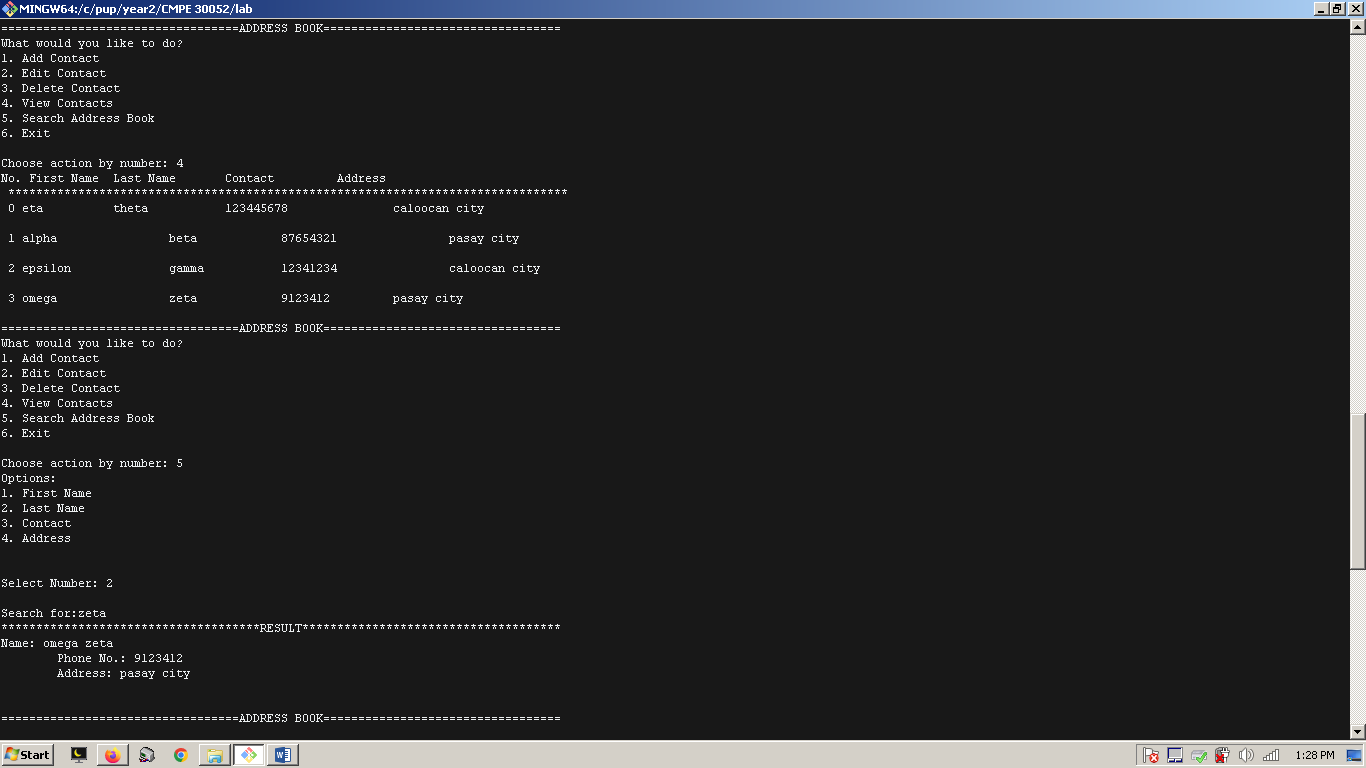


Figure 4. Searching for a keyword in the address book to search for the name *zeta.*

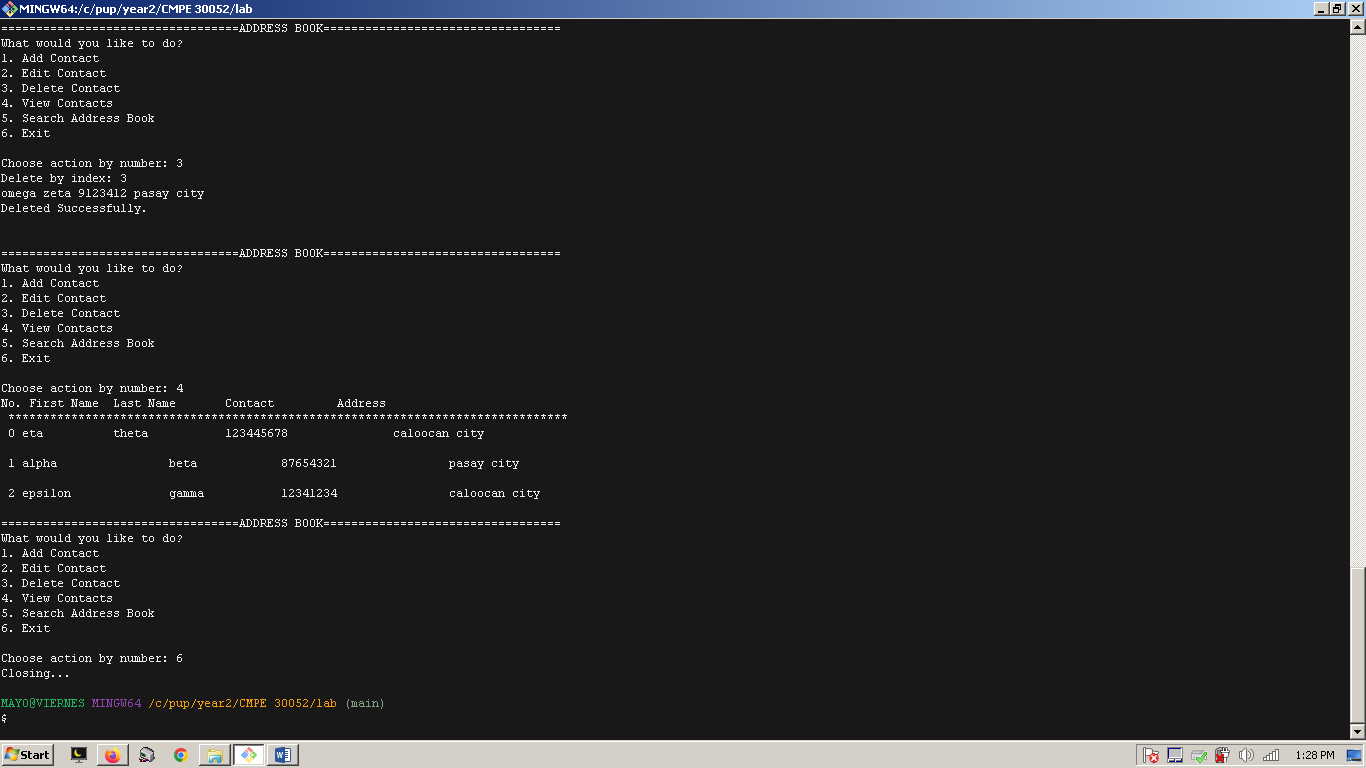


Figure 5. Deleting the fourth entry (3rd index). Reviewing the contents of address book before closing the main application.

REFLECTION

The list data type in Python offers a multitude of methods and use cases, and frequently solves most of the problems that needs immediate receptacle for variables. In this regard, the benefits outweighs any of the very few demerits it has for specific circumstances but it shall.

To start with, I am relieved to say that my code was made out to be organized and simple like how python tries to approach both its developer and user. Using the list data type, there is not a shadow of doubt wherein its usage has come to hand in terms of flexibility and mixed data in structuring quick solutions. No need for excruciating efforts to check whether type is exactly as the developer encoded it the way it was intended because the most important part is that it gives resolution to the problem.

The methods in the list are the cream of the crop when combined with other python expressions. Just in my case, I was measuring the length of the list *fname* so that I can use it to iterate the values of lists so it displays the data using the *view function*; limiting up to where it needs to stop iterating. Moreover, it helped me in implementing the code for *edit* function by simply assigning the values by position the user requested. Voila! Short solutions make powerful effect.

Its fascinating features, however, also come with great consequences. For an instance, as I was trying to manage the search results it had been cumbersome to delineate which position, the index, of the list (e.g. *fname, lname, contact, address*) to match the results in searching for the desired keyword of the user. Therefore, as it may looked organized from afar there is a necessity for a ideas by thinking outside of the box. Still, it was just one of the incompatibilities among the many advantages of the list.

Overall, utilizing list to its maximum potential can make big differences anytime a sequence of data has to be contained and manipulated. The list is the answer, for most of the time.