**Documentation**

**What our project is and what it does?**

Our project is a contact book GUI application that allows users to add contacts, delete contacts, and search for contacts through a database. When the application is launched, the user is presented with their contacts listed out. From there, they can search for a specific contact and view their address or contact information. They could also add additional contacts as they choose or they could delete contacts that are not needed anymore.

**How to run the program via command line:**

To run our program via command, type in the following into the command line from the path of our application:

‘python3 contact\_book.py’

**How to use the program:**

To use the program, the user may start off by adding contacts to their liking. Insert a value into all 4 entry boxes with the phone number being 10-digits. When finished adding contacts, the user may search for contacts by first\_name, last\_name, etc. They may also select a contact with their cursor to delete a selected contact. If the user chooses to view all the contacts, they may click the view buttonand will be shown of all the contacts in the database.

**What to install:**

If not already installed, our application requires the following modules:

Tkinter

SqLite3

Re (Regex)

**Bibliography**

<https://docs.python.org/3/library/sqlite3.html>

The above link is a sqlite3 tutorial/table of contents which taught us about the different commands in the sqlite3 package. This was implemented throughout our Database class to create our database, create our table, execute, and commit.

<https://www.code4example.com/python/python3-get-selected-value-from-listbox-in-tkinter/>

The above link is a tutorial on how to get a selected value from the listbox. We used this source to help implement our get\_selected function. We learned to use the widget and set an index of our selected item. We could then take our desired content using our index.

<https://subscription.packtpub.com/book/application-development/9781785283758/1/ch01lvl1sec10/preventing-the-gui-from-being-resized>

The above link showed us how to make a Tkinter window so that it cannot be re-sized. We implemented this where we created the GUI window by adding the following code: root.resizable(0, 0)