

This is the report for Cmpu291 Winter 18 Assignment 4.
Group Member: Shiyao Chen; Yuanxi Li; Xinyue Xiang

(a) A general overview of the system with a small user guide.

We simply developed an interface using Python and can lead the user to go through those tasks.

1) When a user opens our program, a user interface with all 5 options show up as following:

```
Rexs-MacBook:cmpu291-a4 xinyuexiang$ python3 assignment4.py
1: Q1
2: Q2
3: Q3
4: Q4
E: Exit
Enter your choice: █
```

Users can enter either 1,2,3,4 or E and the functionality for each task has been shown above. Once a task is finished, the user is presented with the 5 options again.

2)Task1:

When 1 is entered, the program allows the user to enter 4 inputs: a start year, an end year and the crime type. Once the user enters the 4 valid inputs, the program will give a bar plot that shows the month-wise total count of the given crime type and also save the plot in the following format: {Q-1}-{count}.png.

3)Task2:

When 2 is entered, the program allows the user to enter 1 input: the number of locations. Once the user enters the value, the program will create a html file in the following format: {Q-2}-{count}.html. When the user opens the file, the user will see a map with N-most populous and N-least populous neighborhoods with their population count.

4)Task3:

When the choice of 3 is entered, the program allows the user to enter 4 inputs: start year, end year, crime type and the number of the neighborhoods. Once the user enters these 4 inputs and the program got that, the program will create a "Q3.html" file. When the user opens this html, the user will see a map with the "Top-N neighborhoods and their crime count" where the crime type user given occurs most during the provided periods.

5)Task4:

When the choice of 4 is entered, the program allows the user to enter 3 inputs: start year, end year, and integer N. Once the user entered the valid inputs, the program will create "Q4.html" file

that contains Top-N neighborhoods with the highest crimes to population ratio within the provided range and the output of the most frequent crime types in each neighborhoods. When the user opens the html file, the user will see the circled map.

(b) A detailed design of the software with a focus on the components required to deliver the major functions of the application.

We took the assignment into two parts. The one is the query, and the other one is the interface.

1) Interface (main function)

We used the imported library sqlite3 to connect to the database.

We have two while loops in the main function.

The first while loop prints out all the tasks with descriptions. If the input is not 'E', the program will never end and requires the users to enter a valid input to go to a specific task.

The second while loop connects the input task with the corresponding function with a parameter 'connection'.

2) tasks (4 define functions)

- task1

The function task_1 has one parameter, connection, that enables the following instructions to be connected with the database (a4.db). This function has 3 parts. The first part is to inquire the valid inputs from the user. The second part is to create the correct query and put the output to the dataframe. The third part is to show and save the bar plot according to the dataframe. Once the task has done the program will go back to the main page, allowing the user to make another choice.

- task2

The function task_1 has one parameter, connection, that enables the following instructions to be connected with the database (a4.db). This function has 4 parts. The first part is to inquire the valid inputs from the user. The second part is to create the query without using limit and offset and put the output to the dataframe. After that, we convert the dataframe into list. The third part of the function is to get the most and least N element from it by comparing. We use this method to deal with the case of a tie. The last part is to show all the selected information on a map. Once the task has done the program will go back to the main page, allowing the user to make another choice.

- task3

The function task_3 has one parameter, connection, that enables the following instructions to be connected with the database(a4.db). The task is mainly to do 2 parts.

The first part is to do the query, get the data frame(neighborhood_name, crime_count, latitude, longitude) from the database. In this task, we transfer the dataframe into list in order to better draw the map. Also, a trick part is to consider with the tie things. We got all the data frame without limit and using for-loop to make a rank so that deal with it. The second part is to do with the map. We use the folium in Python to achieve it. After instantiating a map, we put all the information we want inside the map. Once this task has done, the program will go back to the main page, allowing the user to do another task.

- task4

The function task_4 has one parameter, connection. The function has two parts, the first part is to do the query, get the data frame(neighborhood_name, ratio, latitude, longitude, crime type) from the database. Then we transfer the dataframe into list in order to be better draw the map. For checking the correctness of the input, I used the while loop. For drawing the map, I used the folium. Once this task has done, the program will go back to the main page, allowing the user to do another task.

(c) Testing strategy

- 1) In the development of our program, we used the sample database for simple tests. We follow the sample picture on the eClass and test the basic function of the program.
- 2) After we mainly have done the basic function, we considered some special cases. Like empty entry, like tie cases.
- 3) Once we get the special cases, we found those cases inside the database and use those to test our program.
- 4) We revise our program lots of time because of so many cases we tested and make sure we considered everything carefully.
- 5) Also, we make some communication with other groups about testing, and they provided a lot of great suggestion.

(d) Group work break-down strategy.

1)Cooperation strategy:

We had a group chat room and used the Github to edit our source code.

Also, we set 3 group meeting in order to collaborate better with each other.

2)Group meeting:

First meeting Mar 22nd: done the task distribution.

Second meeting Mar 28th: merged each part into the main program and also done the debug things.

Third meeting Mar 30th: done the README.txt and report.pdf part.

3)Task distribution:

Yuanxi Li: Task4

Shiyao Chen: Task1,2

Xinyue Xiang: Task3 and interface

4)Debug, README.txt and report.pdf:

We talked about the organization of these two documents and wrote them together. Also, we checked each other's part in order to finish this assignment better.