About the Project

The data collection project is to collect data from multiple sources in a streamline manner while allow for people to easily search across all the data platforms. The project allows for a standard to be created for the Smart Cities Project to where data can be utilized by algorithms as this first requires that the data is downloaded unto the lab servers. The visualization of the data is the next important step to where the data can be quickly displayed prior to downloads.

In Summary:

* Collect data from government websites
* Create a standardization to download data for the Smart City Project
* Allow for quick visualization of the data

Getting Started

The program is mainly created in python, and html. There are two main parts of the program “mainapp.py” and “datacollectionapp.py”. The “mainapp.py” connects all the results from the “datacollectionapp.py” and transforms it into a viewable HTML result via the python library Flask. The “datacolectionapp.py” is the original python script that used the Socrata API and python library to allow for the retrieval of data from a user’s requirements of city, state, and topic. The important aspect is knowing the API endpoints which is the primary reason the CSV file was creating to link the city API’s URL for quick searching.

The “test.db” is currently a place holder and the name can be changed. Currently, the test.db is being used to hold an SQL database to generate the selection windows for the data selection in the main screen.

The templates folder contains the different HTML subpages for which Flask uses to create the results of the web site from the program. The ‘base.html’ is the default webpage settings for Flask to render the HTML code. The ‘index.html’ is the first and initial webpage to where the user can select the appropriate city and state to which they want to search datasets for. The ‘dataresults.html’ is a file that allows for the display of the results in a table, and the ability to type in the index numbers for that data to be selected. The ‘map.html’ is used to create a map using the Folium and Leaflet python libraries.

Once the “mainapp.py” program is running, type ‘localhost:5000’ on a web browser to view the web application.

*Perquisites*

The requirement.txt has all the python libraries for the program to work. It is important to update the requirements file after working on it for easy movement between lab mates and the upload to the server.

*GITHUB Link (Last Upload December 3, 2021)*

*https://github.com/degenathan/DataCollection*

Future Steps and Goals

1. Creating a CSV master list
   1. The “city\_api\_list.csv” contains a list of a city and state government that use the Socrata API. This list needs to be edited and expand on over time; more cities will be using the Socrata API however there is the possibility that a city might remove functionality from their open data portal. In the future, the CSV list to be used to generate the SQL database for the selection windows on the initial web page. Furthermore, if possible, it would be better if the SQL database (test.db) would not be needed and the selection windows could be generated straight from the CSV file.
2. City file classification system
   1. If a new download occurs a copy of the data would be download unto the server to an appropriate state and city folder. Additionally, if this is the first time that the city or state is being downloaded then the folders would be created automatically. Furthermore, the possibility of creating tertiary folder for the topics should be highly considerable for better organization.
3. Selection System to Download Data
   1. A checkmark system to where selected rows from the HTML results table can be selected to download on a user’s computers as well as a copy unto the website server.
   2. Allow for a user to preview the data set before downloading. The user should be able to see the geographical locations of points from the data set as well as basic statistics.