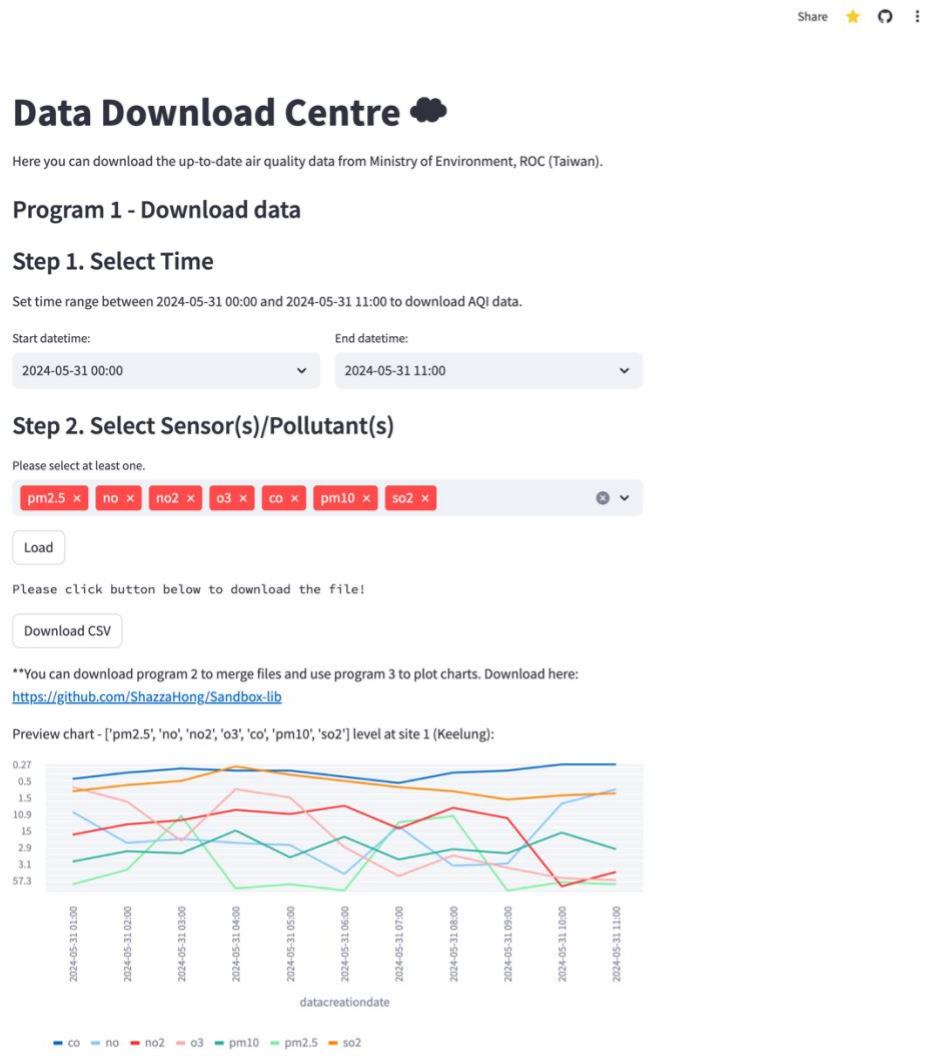
COSC480-24S1(C) – Project – Final Report

Project: Data Download Centre



Data Download Centre

Here you can download the up-to-date air quality data from Ministry of Environment, ROC (Taiwan).

Program 1 - Download data

Step 1. Select Time

Set time range between 2024-05-31 00:00 and 2024-05-31 11:00 to download AQI data.

Start datetime: 2024-05-31 00:00

End datetime: 2024-05-31 11:00

Step 2. Select Sensor(s)/Pollutant(s)

Please select at least one.

pm2.5 × no × no2 × o3 × co × pm10 × so2 ×

```
Please click button below to download the file!
```

**You can download program 2 to merge files and use program 3 to plot charts. Download here: https://github.com/ShazzaHong/Sandbox-lib

Preview chart - ['pm2.5', 'no', 'no2', 'o3', 'co', 'pm10', 'so2'] level at site 1 (Keelung):

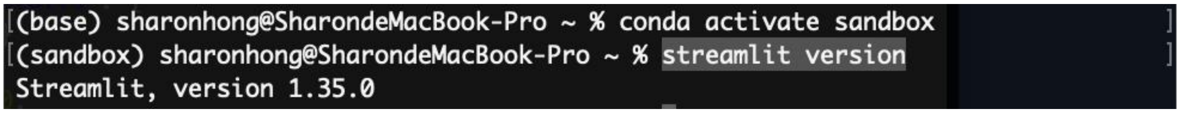
1. Preface

Use case: People from the Ministry of Environment-related department need to analyse the air quality and pollutants, so they can do further inspection or make decisions on policies. They often need up-to-date data and need to download data quite often. Although the government provides public APIs, not everyone knows how to use them or has the tools to access them. Therefore, I build this micro-app to showcase the feasibility of my project and my capability of coding in Python.

The project consists of three programs. Program 1 (name: **Hello.py**) is a Streamlit application deployed publicly (app website) for downloading air quality and pollutants data by using API. Before getting into codes, you can download the data first from the website to have a better understanding of the data and the process. Program 2 (name: **Program 2 - Merge Files.py**) is optional for users as the function is to merge multiple files that have required columns. I created this program because there is an upper limit to data extraction from the data source. If users wish to plot for a longer time with more data, they can consider running Program 2 which uses outer join to merge and excludes the duplication between files. Lastly, Program 3 (name: **Program 3 - Plot Graphs.py**) is mainly to plot the data as a time series chart.

2. How to use my programs

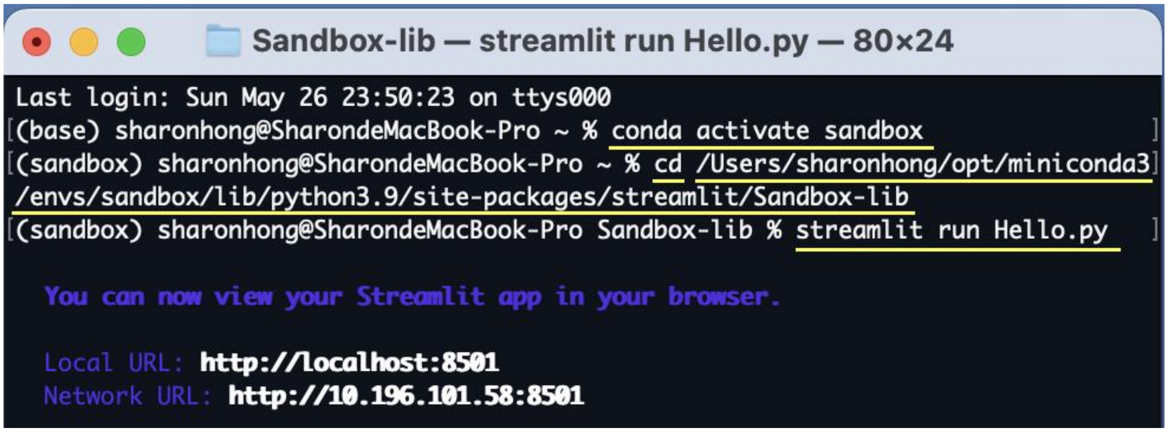
Before running the program, please make sure your system environment (e.g. environment in conda) has (1) installed Python, (2) pip, and s(3) Streamlit. If you haven't installed them yet, please install Python from this website first and read and follow the Streamlit installation guide. You can check the streamlit version you are using by typing `streamlit version` in your terminal or other command prompts. To run program 1 smoothly, I recommend upgrading Python to version 3.9 and Streamlit to version 1.35.0. (Guidance from Streamlit).



```
[(base) sharonhong@SharondeMacBook-Pro ~ % conda activate sandbox                ]
[(sandbox) sharonhong@SharondeMacBook-Pro ~ % streamlit version                  ]
 Streamlit, version 1.35.0
```

Program 1:

- If you already know which virtual environment you've installed Streamlit, please activate the virtual environment where Streamlit is installed. The example below is me activating a conda environment called sandbox, and changing directory (cd) to "Sandbox-lib" folder. Then run the program Hello.py by typing `streamlit run Hello.py`



```
Last login: Sun May 26 23:50:23 on ttys000
[(base) sharonhong@SharondeMacBook-Pro ~ % conda activate sandbox                ]
[(sandbox) sharonhong@SharondeMacBook-Pro ~ % cd /Users/sharonhong/opt/miniconda3]
/envs/sandbox/lib/python3.9/site-packages/streamlit/Sandbox-lib
[(sandbox) sharonhong@SharondeMacBook-Pro Sandbox-lib % streamlit run Hello.py   ]

  You can now view your Streamlit app in your browser.

  Local URL: http://localhost:8501
  Network URL: http://10.196.101.58:8501
```

- Browse Local URL (http://localhost:8501). The page will be the same as the app website.
- After setting the filters, click the "Load" button and wait for a few seconds. Click "Download CSV" to download the CSV files to your computer.

- In summary, run the following commands in CMD after installing :

  001 cd to your directory containing these three Python programs.

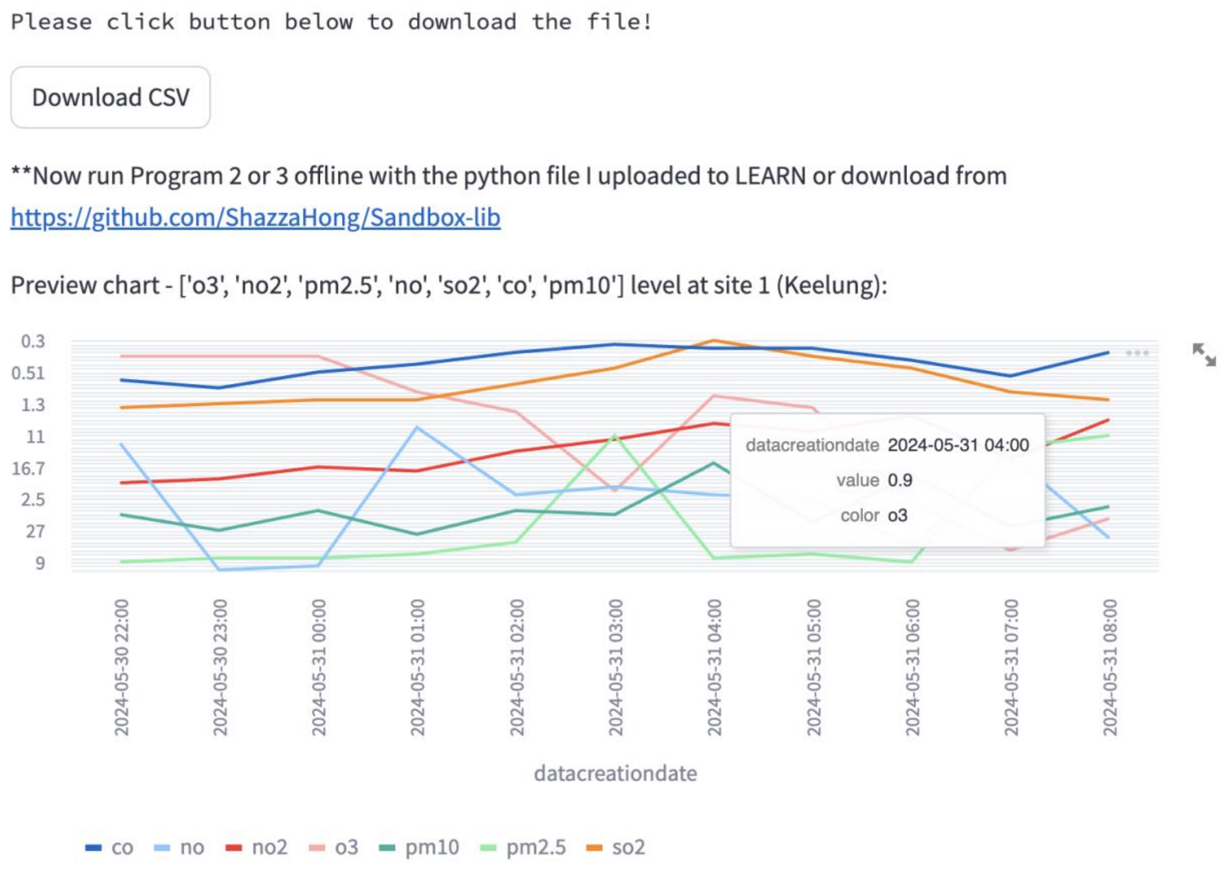
  002 streamlit run **Hello.py**

  If you skip 001, in 002 replace the python file name with the pathnames like **/Users/sharonhong/Downloads/Shazza's Programs/Hello.py**

  003 Browse the webpage by using the Local URL and follow the instructions given on the webpage.

  004 Download the file you need. (Tutors should test Program 2 and 3 with the files downloaded from this webpage.)

At the end of the page, users will see the message hinting them there are two other programs for use. The line chart is a preview for users to hover over lines to quickly read the number and the trend in the last few hours.



```
Please click button below to download the file!
```

**Now run Program 2 or 3 offline with the python file I uploaded to LEARN or download from https://github.com/ShazzaHong/Sandbox-lib

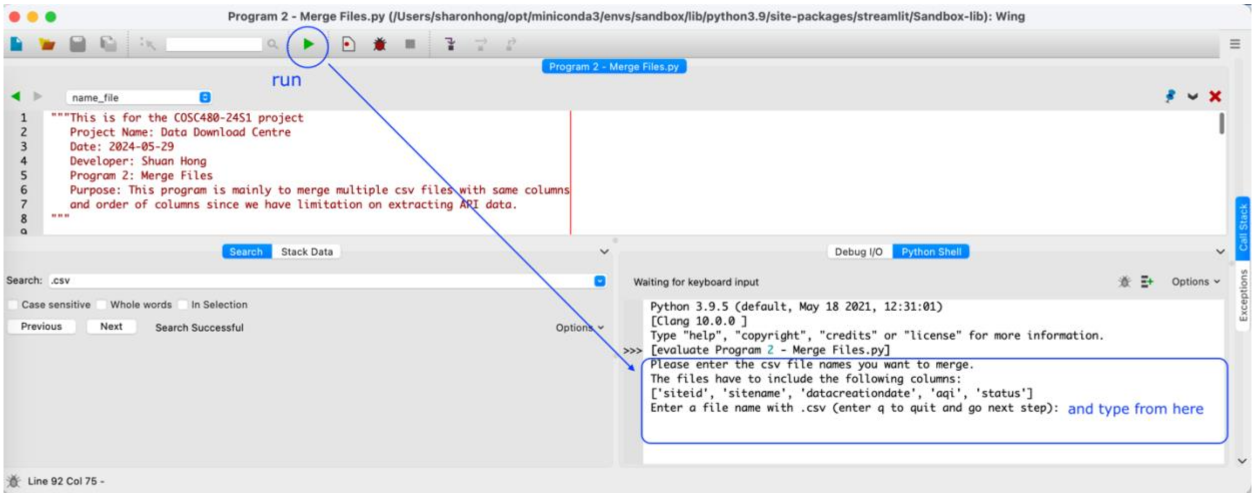
Preview chart - ['o3', 'no2', 'pm2.5', 'no', 'so2', 'co', 'pm10'] level at site 1 (Keelung):

datacreationdate 2024-05-31 04:00
value 0.9
color o3

Program 2:

- Make sure you have downloaded the files from LEARN or the link provided on the Program 1 app website and keep every document (including downloaded CSV) in the **same folder** as the Python programs.
- Open the programs with a code editor (e.g. Wing 101) and click Run.
- When running "Program 2 – Merge Files.py", the message will prompt in the Python Shell to tell users the period of current available data and ask users to enter the period they like to download. Users will need to enter with correct format and time in the given range.

The prompt message after clicking Run hints to users that this program only merges the files with necessary columns (See figure below).



Program 2 - Merge Files.py (/Users/sharonhong/opt/miniconda3/envs/sandbox/lib/python3.9/site-packages/streamlit/Sandbox-lib): Wing
Program 2 - Merge Files.py
run
name_file
Search
Stack Data
Case sensitive
Whole words
In Selection
Previous
Next
Search Successful
Options
Debug I/O
Python Shell
Waiting for keyboard input
Call Stack
Exceptions
and type from here
Line 92 Col 75 -

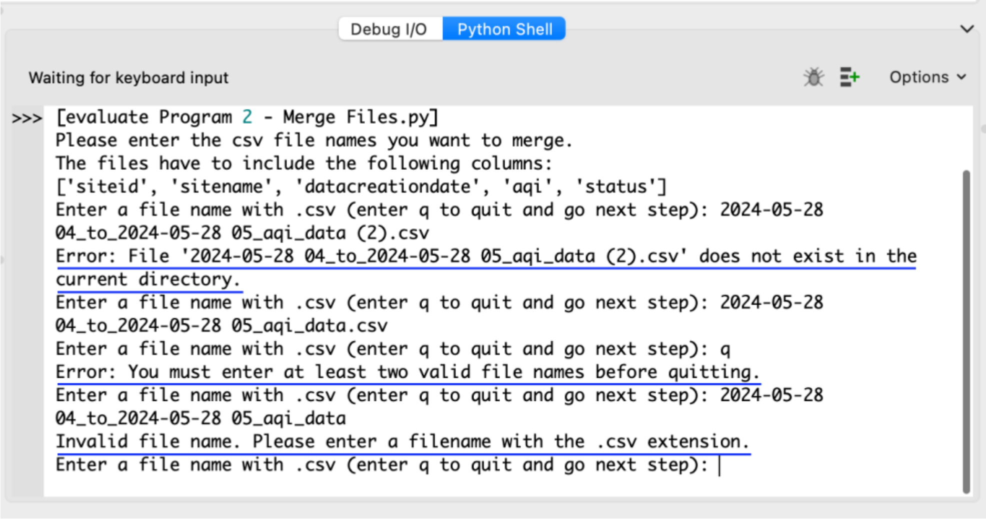
```
"""This is for the COSC480-24S1 project
   Project Name: Data Download Centre
   Date: 2024-05-29
   Developer: Shuan Hong
   Program 2: Merge Files
   Purpose: This program is mainly to merge multiple csv files with same columns
   and order of columns since we have limitation on extracting API data.
"""
```

Search: .csv

```
Python 3.9.5 (default, May 18 2021, 12:31:01)
[Clang 10.0.0 ]
Type "help", "copyright", "credits" or "license" for more information.
>>> [evaluate Program 2 - Merge Files.py]
Please enter the csv file names you want to merge.
The files have to include the following columns:
['siteid', 'sitename', 'datacreationdate', 'aqi', 'status']
Enter a file name with .csv (enter q to quit and go next step):
```

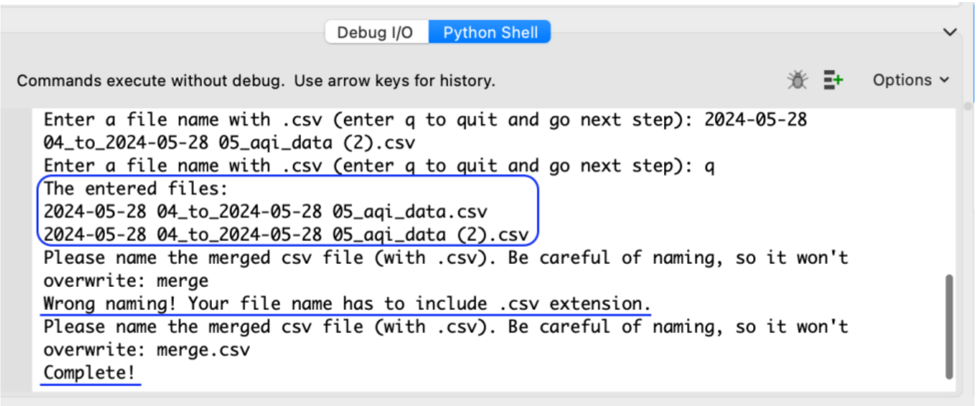
If the filename doesn't have a .csv extension, not in the same directory as the program, the error message will appear to remind users. Example in the figure below.

Users can enter the file names one by one until they enter more than two valid names and want to quit. Error messages will prompt if the file name doesn't include a .csv extension, the file is not found in the current directory, and the user wants to stop before entering two valid answers.



```
>>> [evaluate Program 2 - Merge Files.py]
    Please enter the csv file names you want to merge.
    The files have to include the following columns:
    ['siteid', 'sitename', 'datacreationdate', 'aqi', 'status']
    Enter a file name with .csv (enter q to quit and go next step): 2024-05-28
    04_to_2024-05-28 05_aqi_data (2).csv
    Error: File '2024-05-28 04_to_2024-05-28 05_aqi_data (2).csv' does not exist in the
    current directory.
    Enter a file name with .csv (enter q to quit and go next step): 2024-05-28
    04_to_2024-05-28 05_aqi_data.csv
    Enter a file name with .csv (enter q to quit and go next step): q
    Error: You must enter at least two valid file names before quitting.
    Enter a file name with .csv (enter q to quit and go next step): 2024-05-28
    04_to_2024-05-28 05_aqi_data
    Invalid file name. Please enter a filename with the .csv extension.
    Enter a file name with .csv (enter q to quit and go next step): |
```

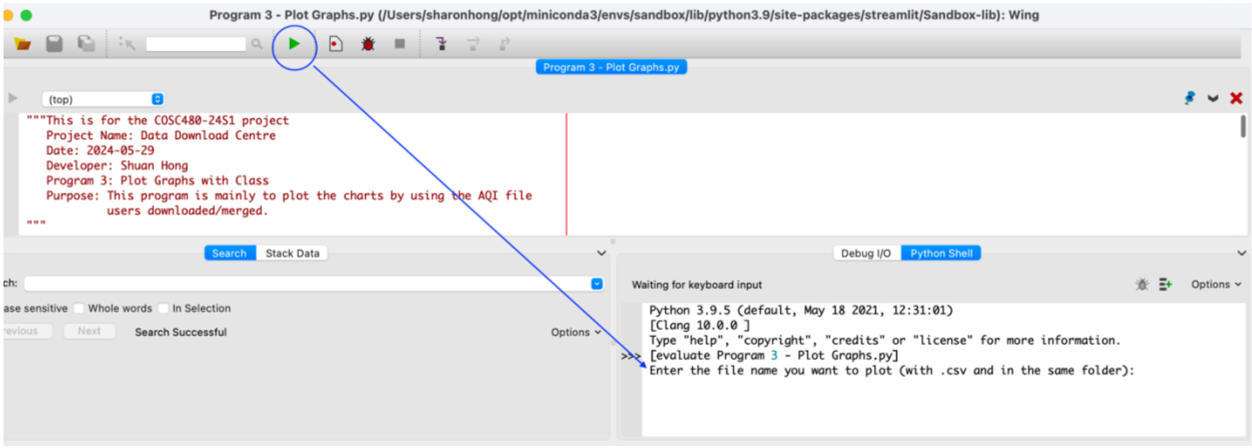
After quitting and before the data is saved as a CSV file, users are allowed to name it as they want. The error message will appear when users don't include a .csv extension.



```
Enter a file name with .csv (enter q to quit and go next step): 2024-05-28
04_to_2024-05-28 05_aqi_data (2).csv
Enter a file name with .csv (enter q to quit and go next step): q
The entered files:
2024-05-28 04_to_2024-05-28 05_aqi_data.csv
2024-05-28 04_to_2024-05-28 05_aqi_data (2).csv
Please name the merged csv file (with .csv). Be careful of naming, so it won't
overwrite: merge
Wrong naming! Your file name has to include .csv extension.
Please name the merged csv file (with .csv). Be careful of naming, so it won't
overwrite: merge.csv
Complete!
```

Program 3

- Because there are full-width left parenthesis characters in the file and that is not available in the default font Matplotlib configurated. We added the **'Heiti TC' font** to ensure that the warning no longer appears. Please make sure your system has this font. If you don't have it installed, please download it from your Font Book or other sources, and keep it in the same directory as the program 3.
- Open the programs with a code editor (e.g. Wing 101) and click Run. Alternatively, you can run the program in the Python command prompt.
- When running "Program 3 – Plot Graphs.py", the message will prompt in the Python Shell to tell users the period of current available data and ask users to enter the period they like to download. Users will need to enter with correct format and time in the given range.

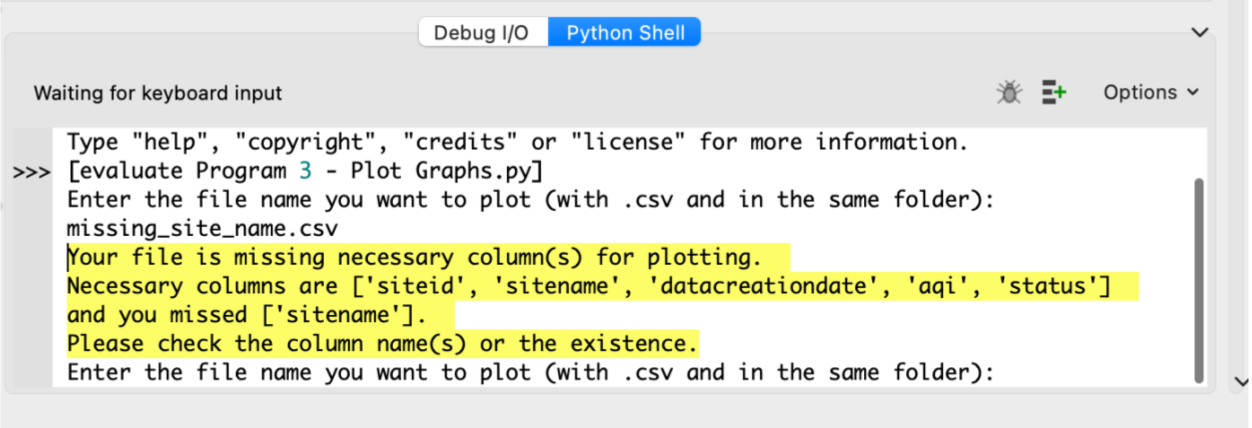


Program 3 - Plot Graphs.py (/Users/sharonhong/opt/miniconda3/envs/sandbox/lib/python3.9/site-packages/streamlit/Sandbox-lib): Wing
Program 3 - Plot Graphs.py
(top)
Search
Stack Data
ch:
ase sensitive
Whole words
In Selection
revious
Next
Search Successful
Options
Debug I/O
Python Shell
Waiting for keyboard input
Options

```
"""This is for the COSC480-24S1 project
   Project Name: Data Download Centre
   Date: 2024-05-29
   Developer: Shuan Hong
   Program 3: Plot Graphs with Class
   Purpose: This program is mainly to plot the charts by using the AQI file
            users downloaded/merged.
"""
```

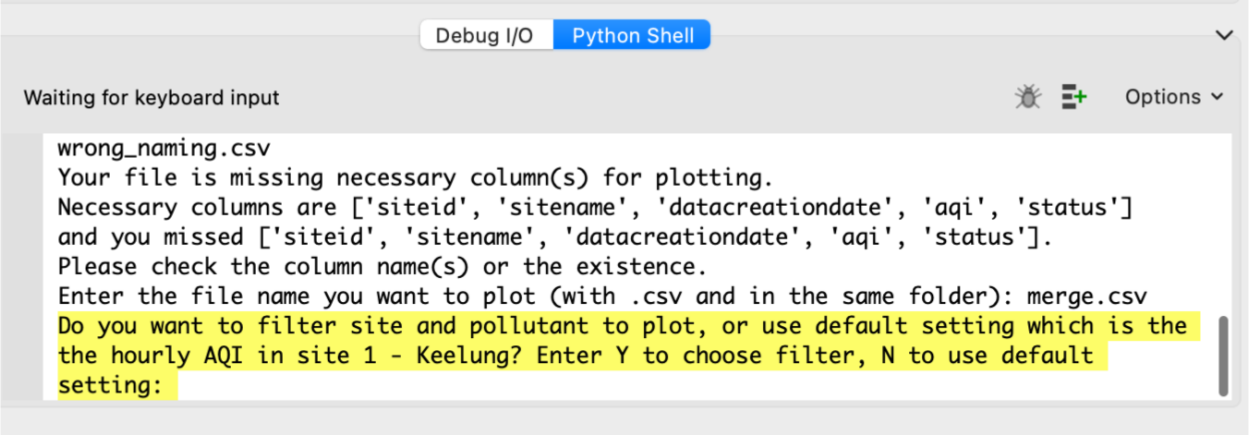
```
    Python 3.9.5 (default, May 18 2021, 12:31:01)
    [Clang 10.0.0 ]
    Type "help", "copyright", "credits" or "license" for more information.
>>> [evaluate Program 3 - Plot Graphs.py]
    Enter the file name you want to plot (with .csv and in the same folder):
```

Error messages will show up when it detects that there are missing columns and list out the missing ones. As shown in the figure below.



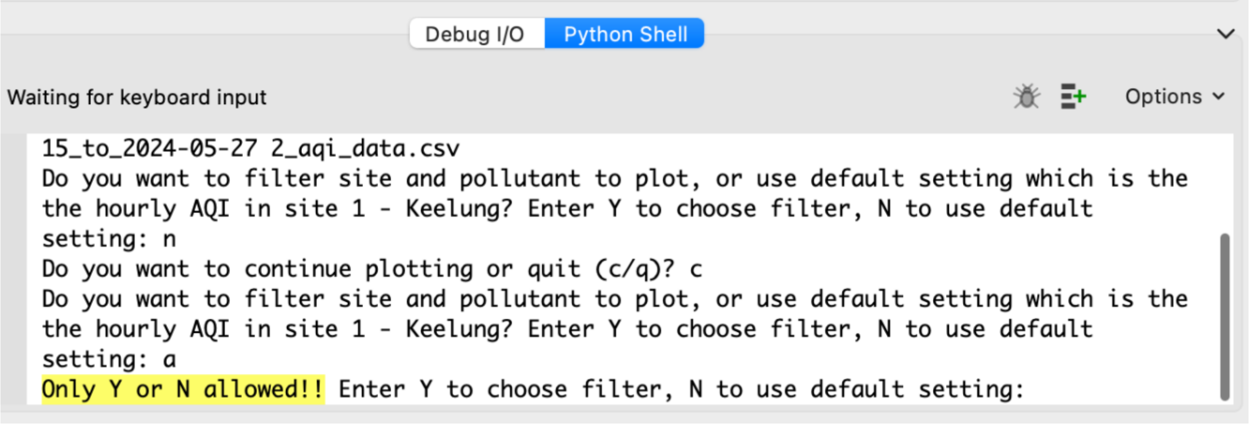
```
    Type "help", "copyright", "credits" or "license" for more information.
>>> [evaluate Program 3 - Plot Graphs.py]
    Enter the file name you want to plot (with .csv and in the same folder):
    missing_site_name.csv
    Your file is missing necessary column(s) for plotting.
    Necessary columns are ['siteid', 'sitename', 'datacreationdate', 'aqi', 'status']
    and you missed ['sitename'].
    Please check the column name(s) or the existence.
    Enter the file name you want to plot (with .csv and in the same folder):
```

If the file passes the examination, it will ask if you want to filter or just plot the default setting. Messages are shown in the figure below.



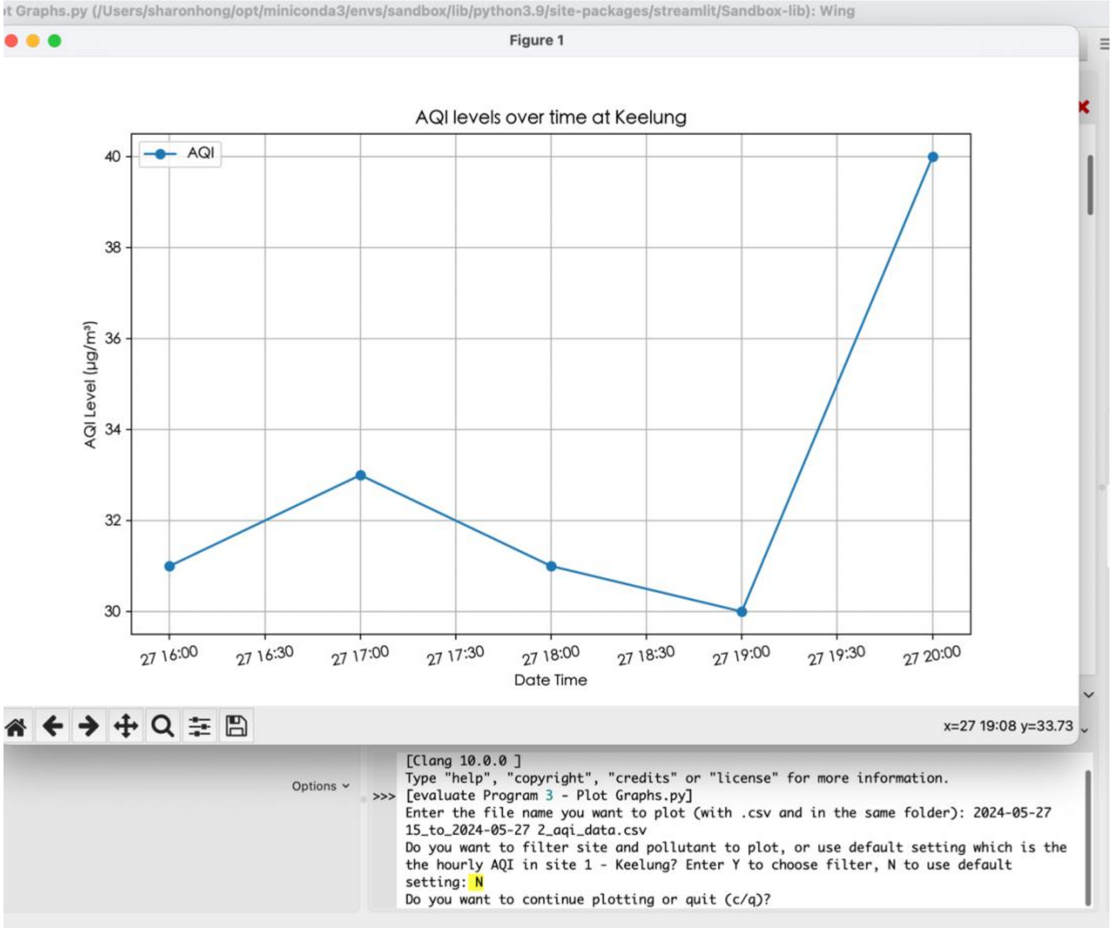
```
wrong_naming.csv
Your file is missing necessary column(s) for plotting.
Necessary columns are ['siteid', 'sitename', 'datacreationdate', 'aqi', 'status']
and you missed ['siteid', 'sitename', 'datacreationdate', 'aqi', 'status'].
Please check the column name(s) or the existence.
Enter the file name you want to plot (with .csv and in the same folder): merge.csv
Do you want to filter site and pollutant to plot, or use default setting which is the
the hourly AQI in site 1 - Keelung? Enter Y to choose filter, N to use default
setting:
```

The program will check the input type and value, case insensitive. The error message is shown in the figure below.



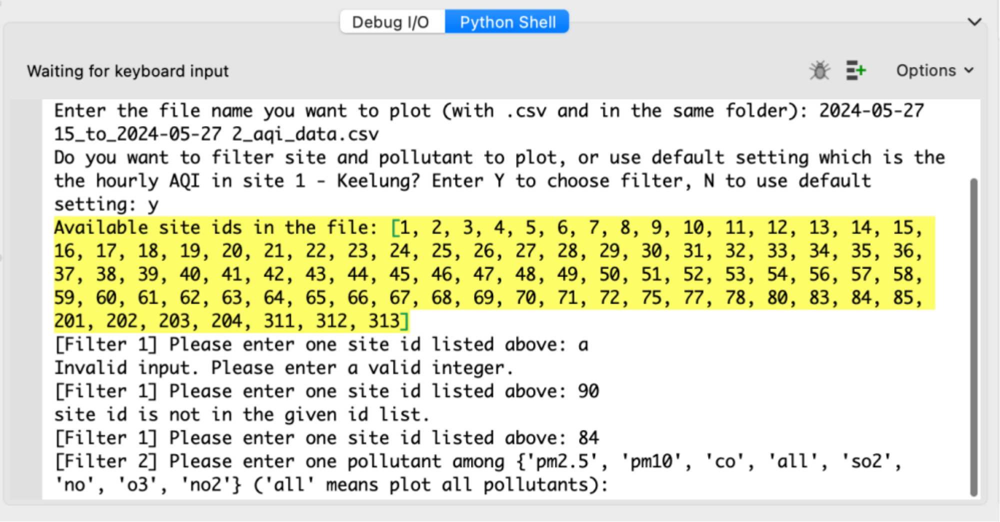
```
15_to_2024-05-27 2_aqi_data.csv
Do you want to filter site and pollutant to plot, or use default setting which is the
the hourly AQI in site 1 - Keelung? Enter Y to choose filter, N to use default
setting: n
Do you want to continue plotting or quit (c/q)? c
Do you want to filter site and pollutant to plot, or use default setting which is the
the hourly AQI in site 1 - Keelung? Enter Y to choose filter, N to use default
setting: a
Only Y or N allowed!! Enter Y to choose filter, N to use default setting:
```

If you enter N or n, the graph will pop up and the default is AQI level over time at Keelung (site).


t Graphs.py (/Users/sharonhong/opt/miniconda3/envs/sandbox/lib/python3.9/site-packages/streamlit/Sandbox-lib): Wing
Figure 1
AQI levels over time at Keelung
AQI
40
38
36
34
32
30
AQI Level (µg/m³)
27 16:00
27 16:30
27 17:00
27 17:30
27 18:00
27 18:30
27 19:00
27 19:30
27 20:00
Date Time
x=27 19:08 y=33.73
Options

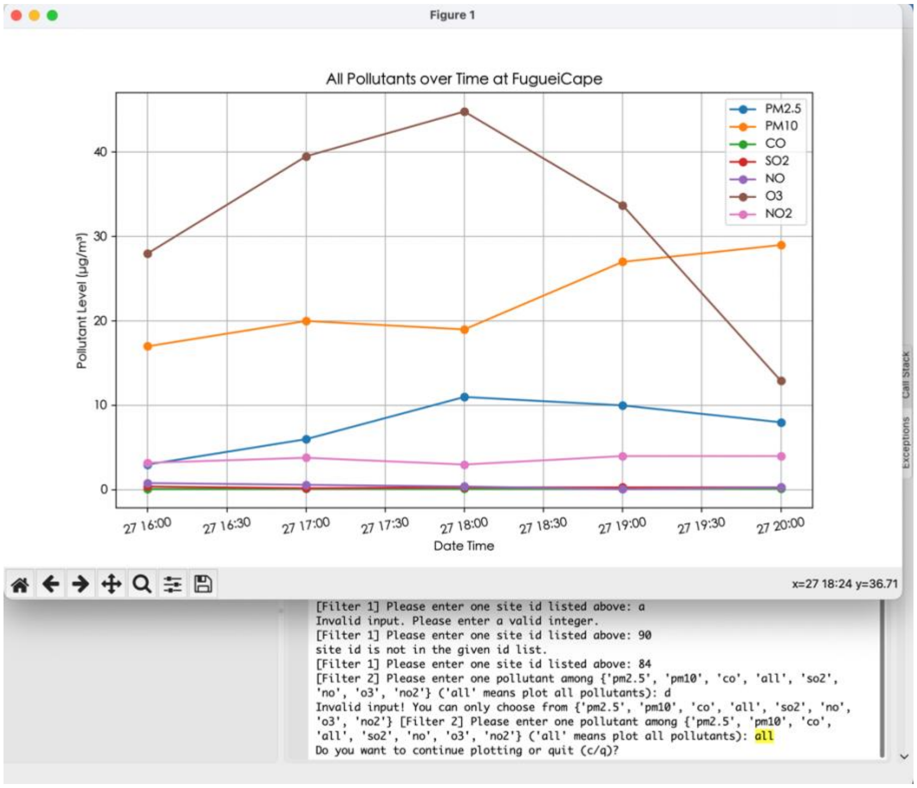
```
    [Clang 10.0.0 ]
    Type "help", "copyright", "credits" or "license" for more information.
>>> [evaluate Program 3 - Plot Graphs.py]
    Enter the file name you want to plot (with .csv and in the same folder): 2024-05-27
    15_to_2024-05-27 2_aqi_data.csv
    Do you want to filter site and pollutant to plot, or use default setting which is the
    the hourly AQI in site 1 - Keelung? Enter Y to choose filter, N to use default
    setting: N
    Do you want to continue plotting or quit (c/q)?
```

If users enter Y or y, it will list out the available site IDs for them to filter. Error messages will show up when users enter invalid input like non-integer or integer that's not on the list. If users enter a valid integer, the second filter will show up to ask users to enter the pollutant they want to plot.



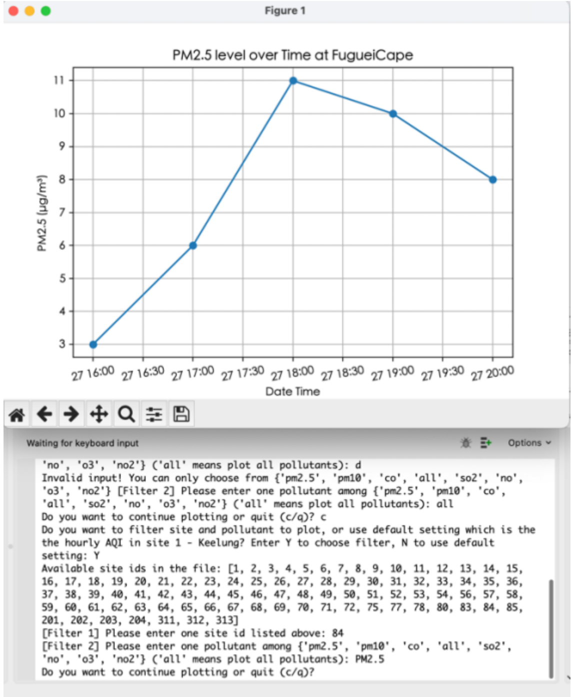
```
Enter the file name you want to plot (with .csv and in the same folder): 2024-05-27
15_to_2024-05-27 2_aqi_data.csv
Do you want to filter site and pollutant to plot, or use default setting which is the
the hourly AQI in site 1 - Keelung? Enter Y to choose filter, N to use default
setting: y
Available site ids in the file: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15,
16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36,
37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 56, 57, 58,
59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 75, 77, 78, 80, 83, 84, 85,
201, 202, 203, 204, 311, 312, 313]
[Filter 1] Please enter one site id listed above: a
Invalid input. Please enter a valid integer.
[Filter 1] Please enter one site id listed above: 90
site id is not in the given id list.
[Filter 1] Please enter one site id listed above: 84
[Filter 2] Please enter one pollutant among {'pm2.5', 'pm10', 'co', 'all', 'so2',
'no', 'o3', 'no2'} ('all' means plot all pollutants):
```

If users enter all, they will see all the pollutants in the file on the plot.



```
[Filter 1] Please enter one site id listed above: a
Invalid input. Please enter a valid integer.
[Filter 1] Please enter one site id listed above: 90
site id is not in the given id list.
[Filter 1] Please enter one site id listed above: 84
[Filter 2] Please enter one pollutant among {'pm2.5', 'pm10', 'co', 'all', 'so2',
'no', 'o3', 'no2'} ('all' means plot all pollutants): d
Invalid input! You can only choose from {'pm2.5', 'pm10', 'co', 'all', 'so2', 'no',
'o3', 'no2'} [Filter 2] Please enter one pollutant among {'pm2.5', 'pm10', 'co',
'all', 'so2', 'no', 'o3', 'no2'} ('all' means plot all pollutants): all
Do you want to continue plotting or quit (c/q)?
```

Users can decide if they want to continue by entering c or q (continue or quit). If continues, it will confirm again with the user about the need for a filter. If this time the user only filters one pollutant, the plot will only show one. (See figure below)



```
'no', 'o3', 'no2'} ('all' means plot all pollutants): d
Invalid input! You can only choose from {'pm2.5', 'pm10', 'co', 'all', 'so2', 'no',
'o3', 'no2'} [Filter 2] Please enter one pollutant among {'pm2.5', 'pm10', 'co',
'all', 'so2', 'no', 'o3', 'no2'} ('all' means plot all pollutants): all
Do you want to continue plotting or quit (c/q)? c
Do you want to filter site and pollutant to plot, or use default setting which is the
the hourly AQI in site 1 - Keelung? Enter Y to choose filter, N to use default
setting: Y
Available site ids in the file: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15,
16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36,
37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 56, 57, 58,
59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 75, 77, 78, 80, 83, 84, 85,
201, 202, 203, 204, 311, 312, 313]
[Filter 1] Please enter one site id listed above: 84
[Filter 2] Please enter one pollutant among {'pm2.5', 'pm10', 'co', 'all', 'so2',
'no', 'o3', 'no2'} ('all' means plot all pollutants): PM2.5
Do you want to continue plotting or quit (c/q)?
```

3. The development process

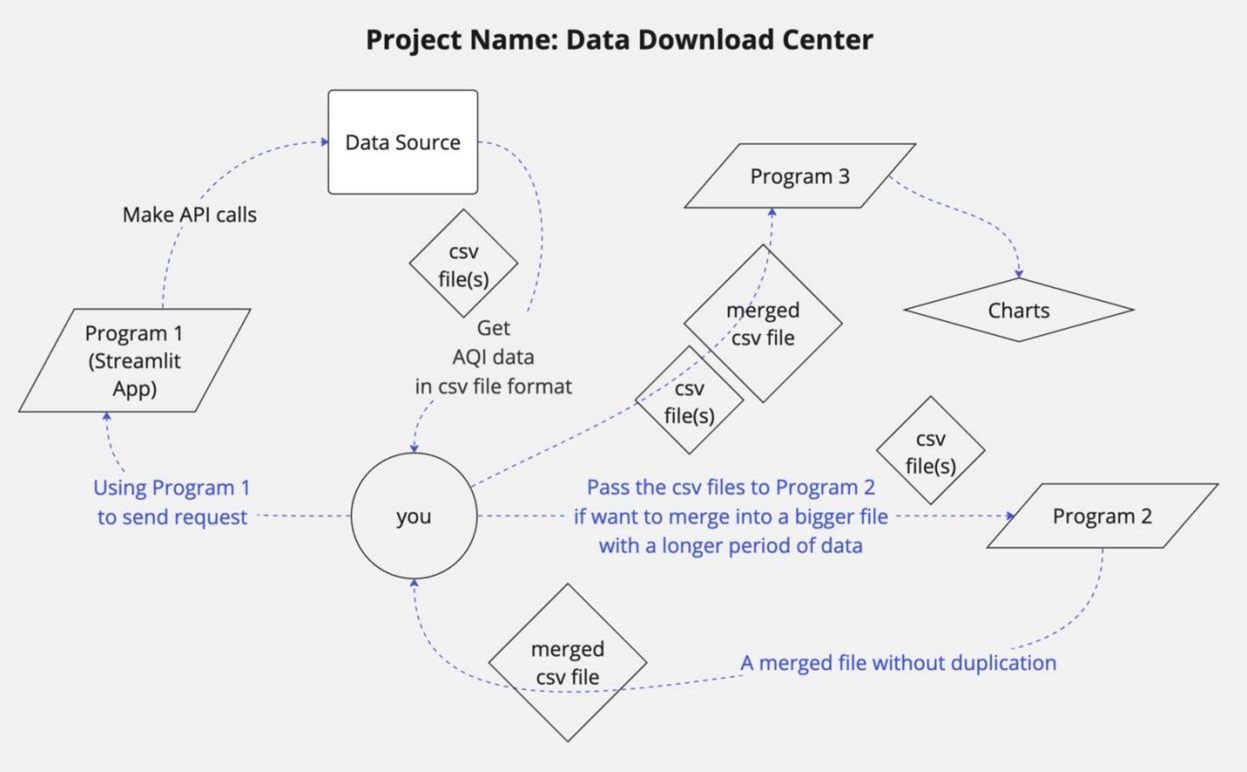
Before developing the application, I had to know what data I could access. Therefore, I went to the public website to check the available APIs. Using the API key I registered, I can preview the response body on the website. Knowing the structure of the data is important for programming as it matters to the list slicing, parameters, naming, etc. After deciding on the data source, I started writing code to call API. The website provides the format and explanation of parameters in the URL, so I only need to replace some parts with variables.

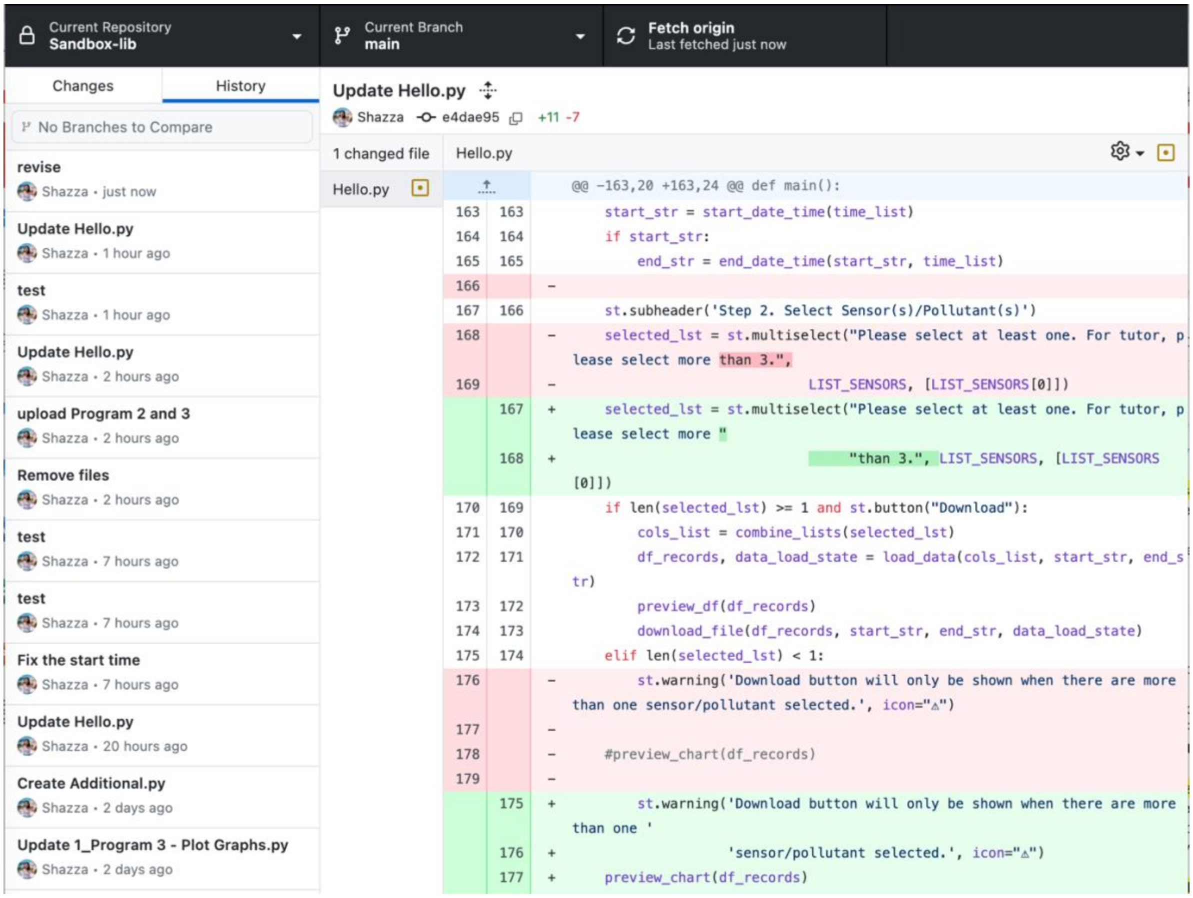
After successfully retrieving the data, I started thinking about what functions would be needed and would be feasible to program. It's more like a brainstorming for me and I was only planning to create filters at the beginning. Later, I recall the experience of downloading data from similar websites to see what users usually need and want. The process of downloading and viewing data visualisation is roughly drafted by pen and paper. However, when I gained more knowledge and skills in programming, I found that there was more I could build. For example, plotting, calculating, using other modules, libraries, etc. After the semester break and receiving feedback from the tutor, I started using the string methods, writing files, handling exceptions, errors, and other things learnt from LM8~LM11. In the programming process, I debugged very often, whenever I created a new function or added variables to several places, as I know it will be harder to trace back if we did too many changes at once.

Lastly, since I want users to download the data from a public website, I connected my Python program with Streamlit and developed the GUI using the Streamlit library. This took me many days to revise the codes to match the functions and features Streamlit provides. However, Streamlit is much more flexible to users as they can go back to the previous step and change the setting.

4. Things that went well

During the break, I successfully linked to the API and stored the data as CSV files. I came out with a better plan for the project and drew the process and data flow as shown in the figure below. When I found errors, I used simple and smaller test sets to test the logic and the reason for blocking. Normally I can understand or find the meaning of error messages. I used quite a lot of True-False, while and for loops to run the program and became familiar with it and can quickly come up with the element to put in those functions. Moreover, from my experience of internship, I know how to use GitHub and I found that GitHub is really helpful in version control as it will record all the changes I did every time I did a push. If I can't recall the code I wrote before, I can go back to check.


Project Name: Data Download Center
Data Source
Make API calls
csv file(s)
Get AQI data in csv file format
Program 1 (Streamlit App)
Using Program 1 to send request
you
Program 3
merged csv file
csv file(s)
Charts
Pass the csv files to Program 2 if want to merge into a bigger file with a longer period of data
csv file(s)
Program 2
merged csv file
A merged file without duplication



revise
Shazza • just now

Update Hello.py
Shazza • 1 hour ago

test
Shazza • 1 hour ago

Update Hello.py
Shazza • 2 hours ago

upload Program 2 and 3
Shazza • 2 hours ago

Remove files
Shazza • 2 hours ago

test
Shazza • 7 hours ago

test
Shazza • 7 hours ago

Fix the start time
Shazza • 7 hours ago

Update Hello.py
Shazza • 20 hours ago

Create Additional.py
Shazza • 2 days ago

Update 1_Program 3 - Plot Graphs.py
Shazza • 2 days ago

Update Hello.py
Shazza e4dae95 +11 -7

```
@@ -163,20 +163,24 @@ def main():
        start_str = start_date_time(time_list)
        if start_str:
            end_str = end_date_time(start_str, time_list)
-
        st.subheader('Step 2. Select Sensor(s)/Pollutant(s)')
-       selected_lst = st.multiselect("Please select at least one. For tutor, please select more than 3.",
-                                     LIST_SENSORS, [LIST_SENSORS[0]])
+       selected_lst = st.multiselect("Please select at least one. For tutor, please select more "
+                                     "than 3.", LIST_SENSORS, [LIST_SENSORS[0]])
        if len(selected_lst) >= 1 and st.button("Download"):
            cols_list = combine_lists(selected_lst)
            df_records, data_load_state = load_data(cols_list, start_str, end_str)
            preview_df(df_records)
            download_file(df_records, start_str, end_str, data_load_state)
        elif len(selected_lst) < 1:
-           st.warning('Download button will only be shown when there are more than one sensor/pollutant selected.', icon="⚠")
-
-       #preview_chart(df_records)
-
+           st.warning('Download button will only be shown when there are more than one '
+                      'sensor/pollutant selected.', icon="⚠")
+       preview_chart(df_records)
```

5. Challenges that you faced

The biggest challenge I encountered is that I found the Streamlit library quite hard to use and for me, it seems like a different language. I want users to have a great user interface to run the program but I can't modify my code to fit into Streamlit's requirement easily. Program 1 has been largely changed/modified after I decided to deploy it on Streamlit.

Handling errors is also one of the biggest challenges for me. It's quite frustrating in debugging as even when I find the reason causing bugs, I still can't find the solution. Sometimes I need hours to come out with a solution. And whenever I feel there are no errors, new errors appear. I'm not only being a developer but also a tester who's running a test script, simulating different kinds of scenarios.

The second challenge is that the available data is different from what I accessed before, so I had to change the features I planned to build in my program(s). For example, I was planning to calculate hourly value from data recorded per minute. However, the data I'm able to access now is already recorded in hours. Also, the time limitation and unfamiliarity with class (OOP) stopped me from adding features that needed complicated calculations. To keep my project complex, I created another program to merge files.

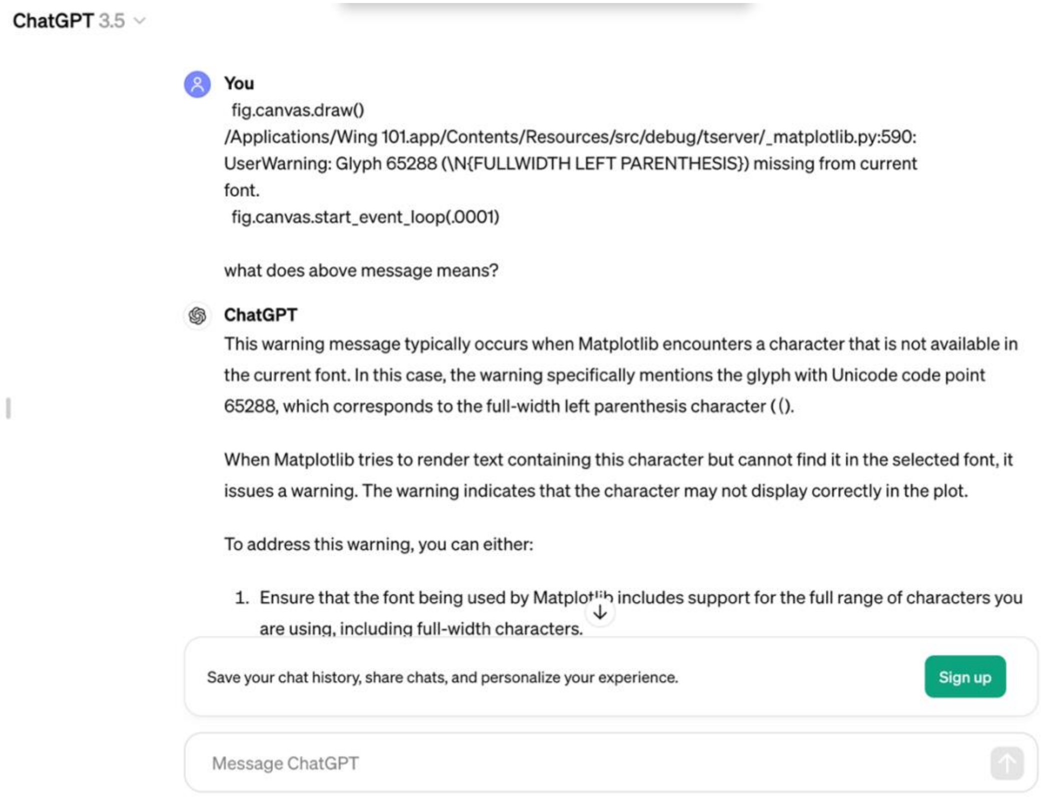
Other issues I faced were that since I was not familiar with programming, I spent lots of time on technical issues such as installing Streamlit on my laptop, connecting GitHub with Streamlit, setting up a programming environment, and so on.

6. Any future developments/work that can be done

Since this project is for me to practice basic programming skills, I was just planning to learn how to use APIs to retrieve real-time data and meet the course requirements. However, later in my current data analyst part-time job/internship, I need to do some analysis of confidential financial data. Therefore, I'll modify the codes to meet the requirement of building a KPI dashboard which shows trends and changes in customer numbers, revenue, and other product performance insights. This needs a more complicated calculation of rates and retention time. My company has data from different platforms, so if I can retrieve data from those platforms and provide APIs with one click (running program), it will save a lot of time and manual work. As we know, retrieving up-to-date data on demand is crucial for data science projects that require real-time data to make accurate predictions or decisions (Custer, 2024). My company can benefit from saving time on wandering around different data sources to join the data manually and to get more timely data analysis to support decision-making on marketing.

7. Appendix

The screenshot below is one of the examples of a conversation between me and ChatGPT. I used ChatGPT as a guide for explaining the errors.


ChatGPT 3.5
Save your chat history, share chats, and personalize your experience.
Sign up
Message ChatGPT

You

```
 fig.canvas.draw()
/Applications/Wing 101.app/Contents/Resources/src/debug/tserver/_matplotlib.py:590:
UserWarning: Glyph 65288 (\N{FULLWIDTH LEFT PARENTHESIS}) missing from current
font.
 fig.canvas.start_event_loop(.0001)
```

what does above message means?

ChatGPT

This warning message typically occurs when Matplotlib encounters a character that is not available in the current font. In this case, the warning specifically mentions the glyph with Unicode code point 65288, which corresponds to the full-width left parenthesis character (（).

When Matplotlib tries to render text containing this character but cannot find it in the selected font, it issues a warning. The warning indicates that the character may not display correctly in the plot.

To address this warning, you can either:

1. Ensure that the font being used by Matplotlib includes support for the full range of characters you are using, including full-width characters.

References

Custer, C. (2024, March 19). *Python API Tutorial: Getting Started with APIs. DataQuest.* https://www.dataquest.io/blog/python-api-tutorial/

Ministry for the Environment, ROC (Taiwan). (2024). *Air quality index (AQI)(historical data)* [Data set]. https://data.moenv.gov.tw/swagger/en/#/air/get_aqx_p_488

Connecting to data. (n.d.). Retrieved from https://docs.streamlit.io/develop/concepts/connections/connecting-to-data

Pandas. (n.d.). *Pandas.* Retrieved May 20, 2024, from https://pandas.pydata.org/