Ozone Layer Project

Start Assignment

Due Monday by 11:59pm **Points** 100 **Submitting** a file upload **File Types** zip

Please watch the lecture walkthrough on this. It will help you.

https://uvu365.sharepoint.com/sites/Course_cs_3270_x01_2023_fall_full_term-1Mrv2v868J1GF/Shared%20Documents/General/Recordings/110723-20231107_153043-Meeting%20Recording.mp4?web=1
(https://uvu365.sharepoint.com/sites/Course_cs_3270_x01_2023_fall_full_term-1Mrv2v868J1GF/Shared%20Documents/General/Recordings/110723-20231107_153043-Meeting%20Recording.mp4?web=1)

Background

You are aware that air pollution is a major problem in today's world, causing respiratory issues for many individuals as well as contributing to global warming. One indicator of pollution is high ozone levels, caused in part by automobile hydrocarbon emissions.

The U.S. Environmental Protection Agency (EPA) maintains ozone detection stations in various locations throughout the country. A key measure is the Air Quality Index (AQI). In the Canvas folder for this assignment is an 80mb (250,000+ lines) file named *daily_44201_2021.csv*, a CSV file containing all AQI data for 2021. (This is raw data downloaded directly from EPA's website.) The first line of the file is a header row naming each column. The fields in each column of each data row are described below.

daily_44201_2021.csv (https://uvu.instructure.com/courses/583149/files/118761057?wrap=1)
(https://uvu.instructure.com/courses/583149/files/118761057/download?download_frd=1)

Field Position	Field Name	Description
1	State Code	The FIPS code of the state in which the monitor resides.
2	County Code	The FIPS code of the county in which the monitor resides.
3	Site Num	A unique number within the county identifying the site.

Field Position	Field Name	Description
4	Parameter Code	The AQS code corresponding to the parameter measured by the monitor.
		(*many fields omitted*)
20	AQI	The Air Quality Index for the day for the pollutant, if applicable.
21	Method Code	An internal system code indicating the method (processes, equipment, and protocols) used in gathering and measuring the sample. The method name is in the next column.
22	Method Name	A short description of the processes, equipment, and protocols used in gathering and measuring the sample.
23	Local Site Name	The name of the site (if any) given by the State, local, or tribal air pollution control agency that operates it.
24	Address	The approximate street address of the monitoring site.
25	State Name	The name of the state where the monitoring site is located.
26	County Name	The name of the county where the monitoring site is located.
27	City Name	The name of the city where the monitoring site is located. This represents the legal incorporated boundaries of cities and not urban areas.
28	CBSA Name	The name of the core bases statistical area (metropolitan area) where the monitoring site is located.
29	Date of Last Change	The date the last time any numeric values in this record were updated in the AQS data system.

You will retrieve data by the names of the state and county (columns 25 and 26).

Requirements

In this project you will plot on demand all AQI data for a given county. The number of AQI entries varies for each county and are ordered chronologically. You just need to retrieve the AQI data (column 20) in the order it appears in the file and plot it. Annotate the plot with the state and county names, as well as the average AQI, as shown below.

Once the state and county have been selected, offer a choice of two destinations for the plot:

1. **Display** (the plot is shown on the screen)
2. **Save to File** (the plot is saved to a file)

Submit your source code along with plot files for

- Utah County, Utah
- Santa Clara County, California
- Wayne County, Michigan
- Pima County, Arizona

Implementation Notes

I spent quite a bit of time extracting which counties are in which state, so I will save you the trouble by giving you that information in the file *counties.txt* (also in the Canvas folder). However, you will need to make that information available at runtime. That should be the first thing your code does. Also, I have included a file, *state_codes.txt*, that associates the two-letter code for each state with their names as they appear in the data file. **(https://uvu.instructure.com/courses/583149/files/118761061?wrap=1)**

state_codes-1.txt (https://uvu.instructure.com/courses/583149/files/118761064?wrap=1)
(https://uvu.instructure.com/courses/583149/files/118761064/download?download_frd=1)

counties.txt (https://uvu.instructure.com/courses/583149/files/118761066?wrap=1)
(https://uvu.instructure.com/courses/583149/files/118761066/download?download_frd=1)

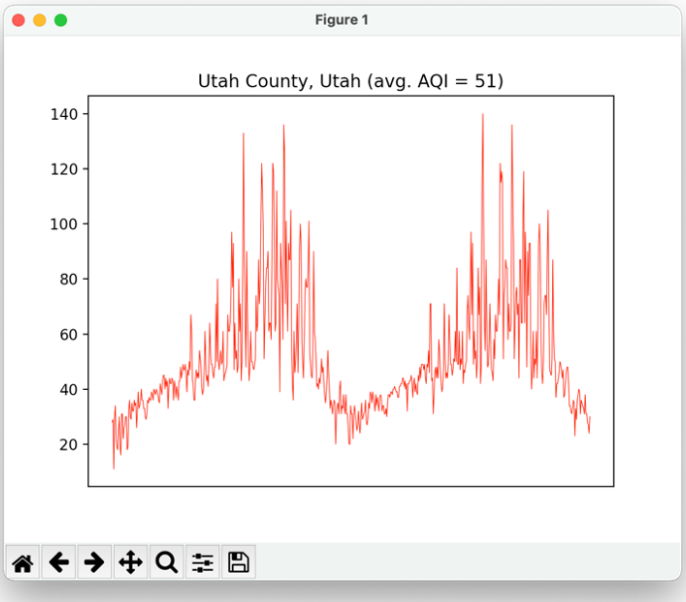
You will also need to locate the data for a specific county in a state at runtime. Here is a sample execution:

$ **python3 ozone.py**

```
Loading data...
Enter 2-letter state code (Q to quit): ut
1: Box Elder
2: Cache
3: Carbon
4: Davis
5: Duchesne
6: Garfield
```

```
7: Iron
8: Salt Lake
9: San Juan
10: Tooele
11: Uintah
12: Utah
13: Washington
14: Weber
Enter number for county: 12
Choose destination for plot:
     1    Screen
     2    File
```

1




```
Another Utah county? (y/n): y
1: Box Elder
2: Cache
3: Carbon
4: Davis
```

```
5: Duchesne
6: Garfield
7: Iron
8: Salt Lake
9: San Juan
10: Tooele
11: Uintah
12: Utah
13: Washington
14: Weber
Enter number for county: 12
Choose destination for plot:
     1    Screen
     2    File
```

2

Enter file with extension of jpg|png|pdf: **utah.png. (The plot is saved in utah.png)**

```
Another Utah county? (y/n): n
Enter 2-letter state code (Q to quit): ca
1: Alameda
2: Amador
3: Butte
4: Calaveras
5: Colusa
6: Contra Costa
7: El Dorado
8: Fresno
9: Glenn
10: Humboldt
11: Imperial
12: Inyo
```

```
13: Kern
14: Kings
15: Lake
16: Los Angeles
17: Madera
18: Marin
19: Mariposa
20: Monterey
21: Napa
22: Nevada
23: Orange
24: Placer
25: Riverside
26: Sacramento
27: San Benito
28: San Bernardino
29: San Diego
30: San Francisco
31: San Joaquin
32: San Luis Obispo
33: San Mateo
34: Santa Barbara
35: Santa Clara
36: Santa Cruz
37: Shasta
38: Solano
39: Sonoma
40: Stanislaus
41: Sutter
42: Tehama
43: Tulare
```

```
44: Tuolumne

45: Ventura

46: Yolo

Enter number for county: 10

Choose destination for plot:

     1    Screen

     2    File
```

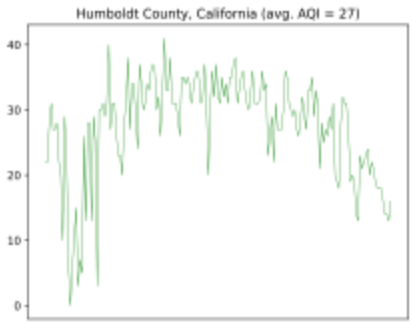
2

```
Enter file with extension of jpg|png|pdf: humboldt.pdf

Another California county? (y/n): n

Enter 2-letter state code (Q to quit): q
```

$



The file *humboldt.pdf* now contains the following plot:

Notice the green vs the red. Plots with an average above 50 should be in red; all others should be green. Round the average to the nearest integer for display.

By the way, check for invalid state codes:

```
$ python3 ozone.py

Loading data...

Enter 2-letter state code (Q to quit): ab

Invalid state code!

Enter 2-letter state code (Q to quit): al

1: Baldwin

2: DeKalb
```

```
3: Elmore
4: Etowah
5: Jefferson
6: Madison
7: Mobile
8: Montgomery
9: Morgan
10: Russell
11: Shelby
12: Sumter
13: Tuscaloosa
Enter number for county:…
```

Use **pandas** for this project. Read the entire EPA file into a Pandas **DataFrame** with **pandas.read_csv** once (you do *not* need Python's **csv** module). Use **Dataframe** indexing operations (either with brackets or the **loc** method) to locate the data needed for a selected plot. Pass the retrieved data to **matplotlib** for plotting.

This is one of the shortest projects (45 lines for me). It's a project instead of an exercise because you're dealing with some new concepts that'll take a little thought to get going with.

Submit your source code along with plot files for

Utah County, Utah
Santa Clara County, California
Wayne County, Michigan
Pima County, Arizona

Ozone Project Rubric

Criteria	Ratings		Pts
Code simulates the correct output of the test case above	**40 pts** **Full Marks**	**0 pts** **No Marks**	40 pts
Creates correct graph	**20 pts** **Full Marks**	**0 pts** **No Marks**	20 pts
Saves graph image to a file	**20 pts** **Full Marks**	**0 pts** **No Marks**	20 pts
Submits 4 graphs as required	**20 pts** **Full Marks**	**0 pts** **No Marks**	20 pts
			Total Points: 100