SI 206 - Final Project Report

By Tawseef Rahman and Evelyn Nguyen

1. Goals - Our Plans

The goals of this project are to analyze the correlation between average high temperatures in degrees Fahrenheit and average rainfall in inches for September 1, 2023 for the top 100 cities in the United States by population. To extract the data, we will use BeautifulSoup to extract the top 100 cities in the United States by population, use an API to gather the average high temperature data on September 1, 2023, and use an API to gather the average rainfall data on September 1, 2023.

2. Goals - Our Achievements

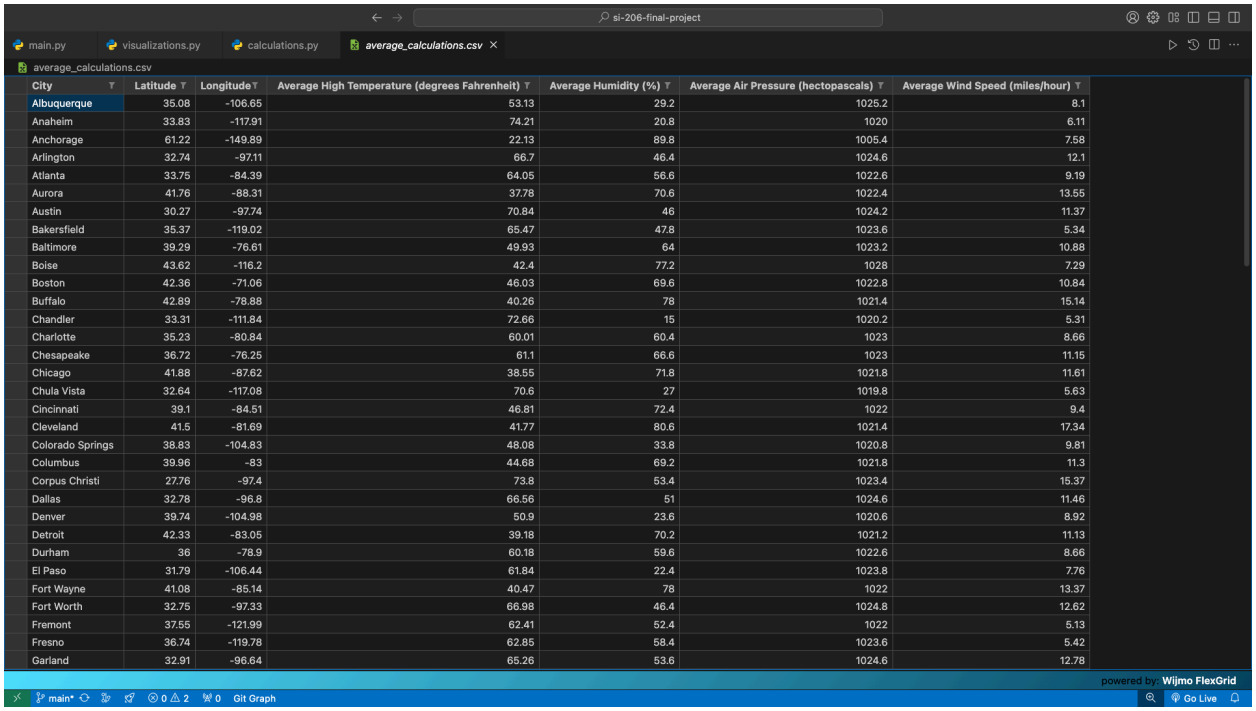
We had to change our original goals above because we felt that just getting the data for the average high temperature and average rainfall for one day in 2023 for the top 100 cities in the United States by population was insufficient for data analysis. Instead, we collected the average high temperature (in degrees Fahrenheit), average humidity (in percentage), average air pressure (in hectopascals), and average wind speed (in miles per hour) over a period of five days (December 16, 2024 to December 20, 2024 for this project). Collecting more data for the top 100 cities by population in the United States over more days instead of just looking at one day allowed us to make more meaningful data analysis by examining trends in the data with visualizations.

3. Problems We Faced

We ran out of API uses for the WeatherStack API that we listed in the “8. Resources” section of this report, so we decided to use the OpenWeather API to gather all the necessary weather data for the period of five days (December 16, 2024 to December 20, 2024) for this project, not just rainfall data for September 1, 2023.

4. Calculations from the Data in the Database

For each of the top 100 cities in the United States by population, we calculated the average high temperature, average humidity, average air pressure, and average wind speed over a period of five days (December 16, 2024 to December 20, 2024). Below are the results of the aforementioned calculations that were written to a CSV file:



si-206-final-project

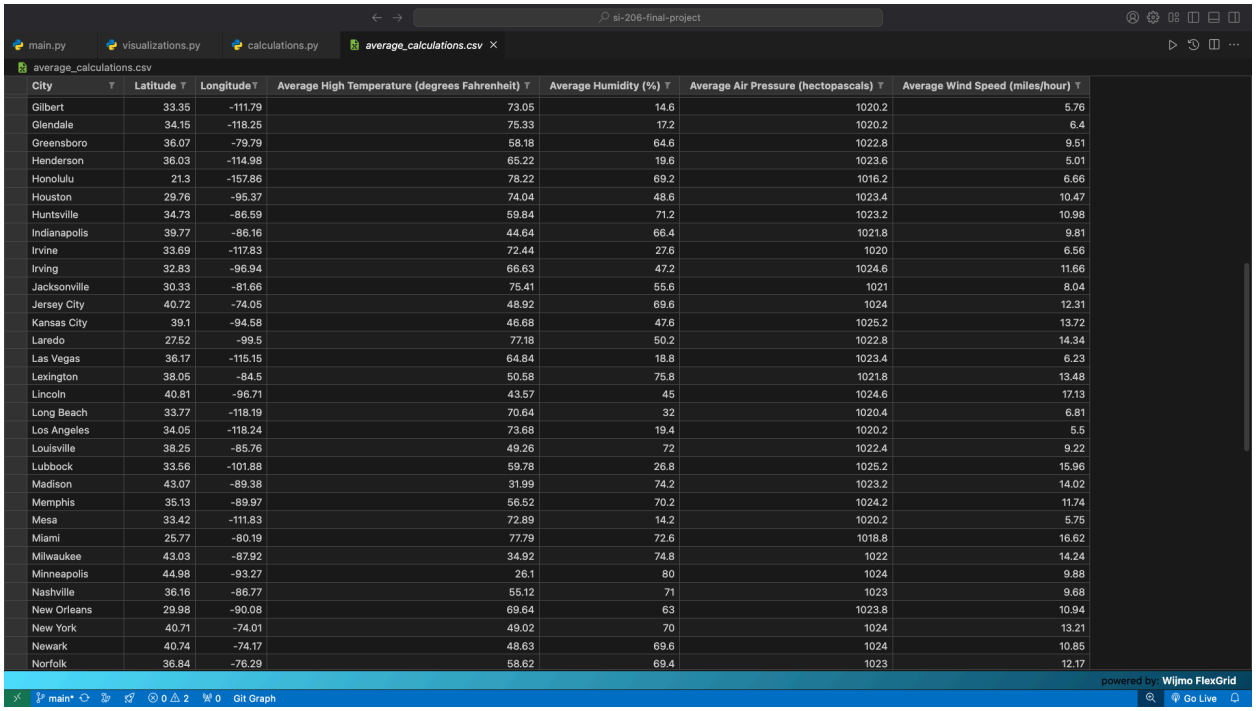
main.py | visualizations.py | calculations.py | average_calculations.csv

average_calculations.csv

City	Latitude	Longitude	Average High Temperature (degrees Fahrenheit)	Average Humidity (%)	Average Air Pressure (hectopascals)	Average Wind Speed (miles/hour)
Albuquerque	35.08	-106.65	53.13	29.2	1025.2	8.1
Anaheim	33.83	-117.91	74.21	20.8	1020	6.11
Anchorage	61.22	-149.89	22.13	89.8	1005.4	7.58
Arlington	32.74	-97.11	66.7	46.4	1024.6	12.1
Atlanta	33.75	-84.39	64.05	56.6	1022.6	9.19
Aurora	41.76	-88.31	37.78	70.6	1022.4	13.55
Austin	30.27	-97.74	70.84	46	1024.2	11.37
Bakersfield	35.37	-119.02	65.47	47.8	1023.6	5.34
Baltimore	39.29	-76.61	49.93	64	1023.2	10.88
Boise	43.62	-116.2	42.4	77.2	1028	7.29
Boston	42.36	-71.06	46.03	69.6	1022.8	10.84
Buffalo	42.89	-78.88	40.26	78	1021.4	15.14
Chandler	33.31	-111.84	72.66	15	1020.2	5.31
Charlotte	35.23	-80.84	60.01	60.4	1023	8.66
Chesapeake	36.72	-76.25	61.1	66.6	1023	11.15
Chicago	41.88	-87.62	38.55	71.8	1021.8	11.61
Chula Vista	32.64	-117.08	70.6	27	1019.8	5.63
Cincinnati	39.1	-84.51	46.81	72.4	1022	9.4
Cleveland	41.5	-81.69	41.77	80.6	1021.4	17.34
Colorado Springs	38.83	-104.83	48.08	33.8	1020.8	9.81
Columbus	39.96	-83	44.68	69.2	1021.8	11.3
Corpus Christi	27.76	-97.4	73.8	53.4	1023.4	15.37
Dallas	32.78	-96.8	66.56	51	1024.6	11.46
Denver	39.74	-104.98	50.9	23.6	1020.6	8.92
Detroit	42.33	-83.05	39.18	70.2	1021.2	11.13
Durham	36	-78.9	60.18	59.6	1022.6	8.66
El Paso	31.79	-106.44	61.84	22.4	1023.8	7.76
Fort Wayne	41.08	-85.14	40.47	78	1022	13.37
Fort Worth	32.75	-97.33	66.98	46.4	1024.8	12.62
Fremont	37.55	-121.99	62.41	52.4	1022	5.13
Fresno	36.74	-119.78	62.85	58.4	1023.6	5.42
Garland	32.91	-96.64	65.26	53.6	1024.6	12.78

powered by Wijmo FlexGrid

main* | Git Graph | Go Live



si-206-final-project

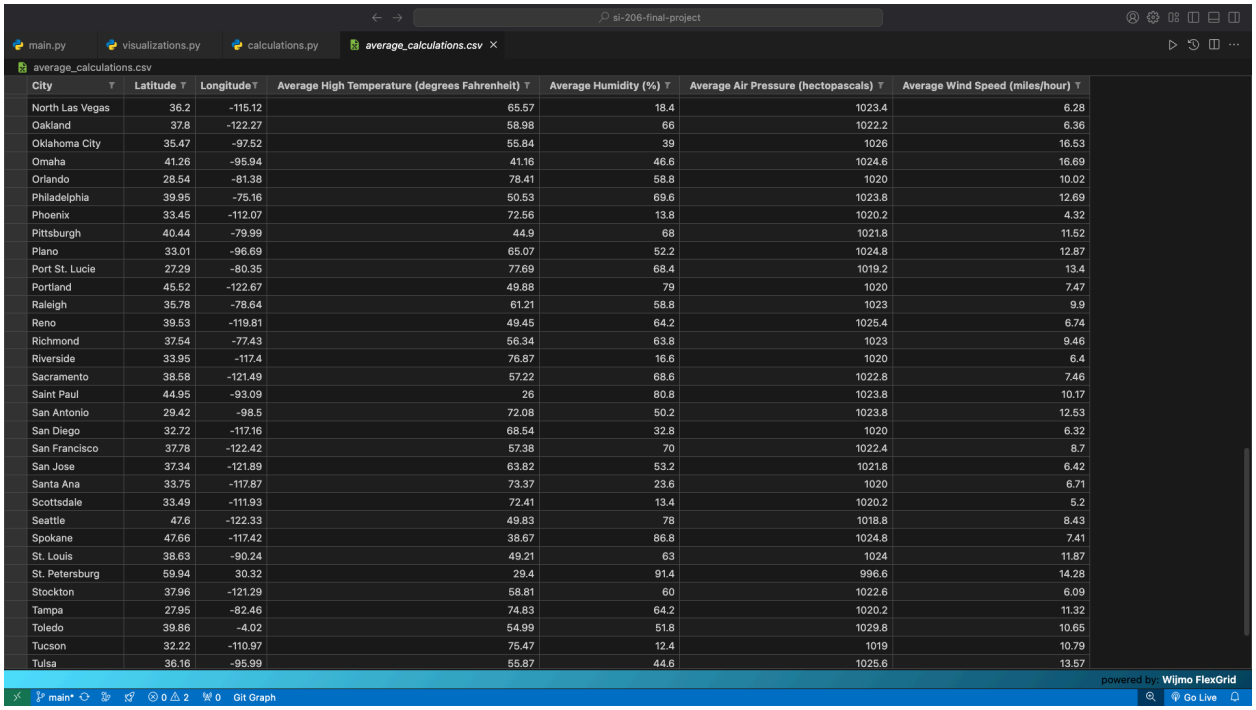
main.py | visualizations.py | calculations.py | average_calculations.csv

average_calculations.csv

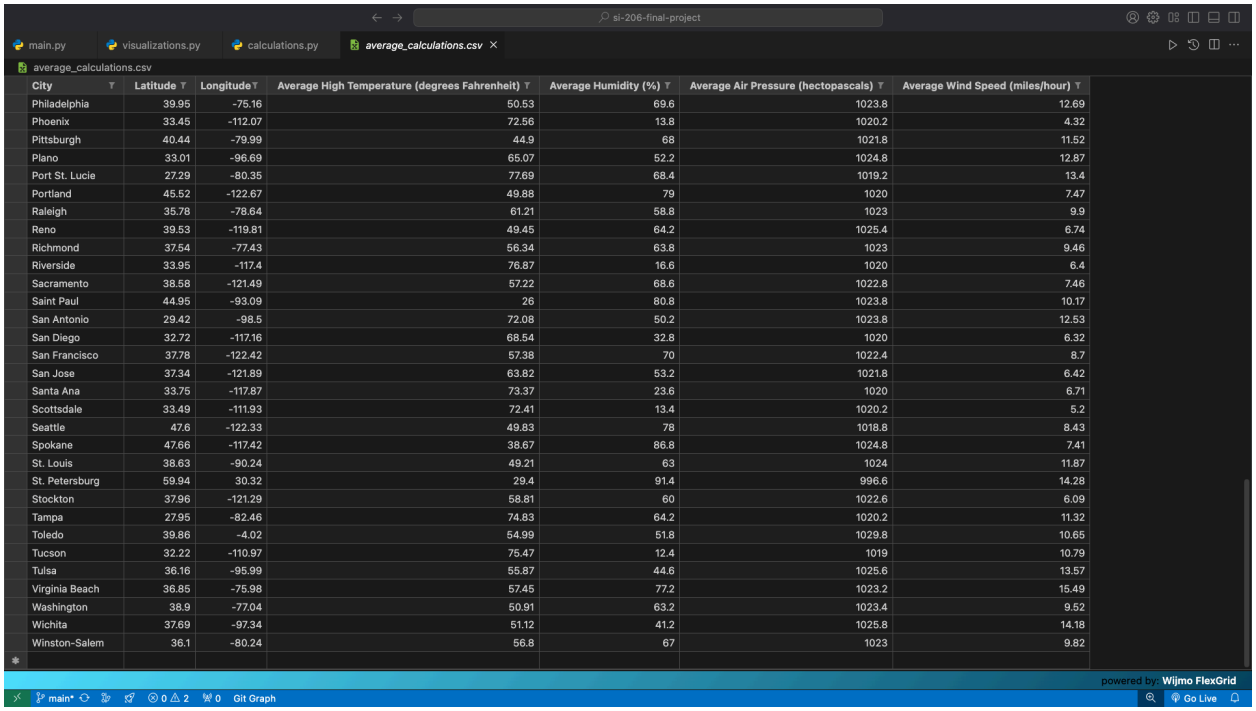
City	Latitude	Longitude	Average High Temperature (degrees Fahrenheit)	Average Humidity (%)	Average Air Pressure (hectopascals)	Average Wind Speed (miles/hour)
Gilbert	33.35	-111.79	73.05	14.6	1020.2	5.76
Glendale	34.15	-118.25	75.33	17.2	1020.2	6.4
Greensboro	36.07	-79.79	58.18	64.6	1022.8	9.51
Henderson	36.03	-114.98	65.22	19.6	1023.6	5.01
Honolulu	21.3	-157.86	78.22	69.2	1016.2	6.66
Houston	29.76	-95.37	74.04	48.6	1023.4	10.47
Huntsville	34.73	-86.59	59.84	71.2	1023.2	10.98
Indianapolis	39.77	-86.16	44.64	66.4	1021.8	9.81
Irvine	33.69	-117.83	72.44	27.6	1020	6.56
Irving	32.83	-96.94	66.63	47.2	1024.6	11.66
Jacksonville	30.33	-81.66	75.41	55.6	1021	8.04
Jersey City	40.72	-74.05	48.92	69.6	1024	12.31
Kansas City	39.1	-94.58	46.68	47.6	1025.2	13.72
Laredo	27.52	-99.5	77.18	50.2	1022.8	14.34
Las Vegas	36.17	-115.15	64.84	18.8	1023.4	6.23
Lexington	38.05	-84.5	50.58	75.8	1021.8	13.48
Lincoln	40.81	-96.71	43.57	45	1024.6	17.13
Long Beach	33.77	-118.19	70.64	32	1020.4	6.81
Los Angeles	34.05	-118.24	73.68	19.4	1020.2	5.5
Louisville	38.25	-85.76	49.26	72	1022.4	9.22
Lubbock	33.56	-101.88	59.78	26.8	1025.2	15.96
Madison	43.07	-89.38	31.99	74.2	1023.2	14.02
Memphis	35.13	-89.97	56.52	70.2	1024.2	11.74
Mesa	33.42	-111.83	72.89	14.2	1020.2	5.75
Miami	25.77	-80.19	77.79	72.6	1018.8	16.62
Milwaukee	43.03	-87.92	34.92	74.8	1022	14.24
Minneapolis	44.98	-93.27	26.1	80	1024	9.88
Nashville	36.16	-86.77	55.12	71	1023	9.68
New Orleans	29.98	-90.08	69.64	63	1023.8	10.94
New York	40.71	-74.01	49.02	70	1024	13.21
Newark	40.74	-74.17	48.63	69.6	1024	10.85
Norfolk	36.84	-76.29	58.62	69.4	1023	12.17

powered by Wijmo FlexGrid

main* | Git Graph | Go Live



City	Latitude	Longitude	Average High Temperature (degrees Fahrenheit)	Average Humidity (%)	Average Air Pressure (hectopascals)	Average Wind Speed (miles/hour)
North Las Vegas	36.2	-115.12	65.57	18.4	1023.4	6.28
Oakland	37.8	-122.27	58.98	66	1022.2	6.36
Oklahoma City	35.47	-97.52	55.84	39	1026	16.53
Omaha	41.26	-95.94	41.16	46.6	1024.6	16.69
Orlando	28.54	-81.38	78.41	58.8	1020	10.02
Philadelphia	39.95	-75.16	50.53	69.6	1023.8	12.69
Phoenix	33.45	-112.07	72.56	13.8	1020.2	4.32
Pittsburgh	40.44	-79.99	44.9	68	1021.8	11.52
Plano	33.01	-96.69	65.07	52.2	1024.8	12.87
Port St. Lucie	27.29	-80.35	77.69	68.4	1019.2	13.4
Portland	45.52	-122.67	49.88	79	1020	7.47
Raleigh	35.78	-78.64	61.21	58.8	1023	9.9
Reno	39.53	-119.81	49.45	64.2	1025.4	6.74
Richmond	37.54	-77.43	56.34	63.8	1023	9.46
Riverside	33.95	-117.4	76.87	16.6	1020	6.4
Sacramento	38.58	-121.49	57.22	68.6	1022.8	7.46
Saint Paul	44.95	-93.09	26	80.8	1023.8	10.17
San Antonio	29.42	-98.5	72.08	50.2	1023.8	12.53
San Diego	32.72	-117.16	68.54	32.8	1020	6.32
San Francisco	37.78	-122.42	57.38	70	1022.4	8.7
San Jose	37.34	-121.89	63.82	53.2	1021.8	6.42
Santa Ana	33.75	-117.87	73.37	23.6	1020	6.71
Scottsdale	33.49	-111.93	72.41	13.4	1020.2	5.2
Seattle	47.6	-122.33	49.83	78	1018.8	8.43
Spokane	47.66	-117.42	38.67	86.8	1024.8	7.41
St. Louis	38.63	-90.24	49.21	63	1024	11.87
St. Petersburg	59.94	30.32	29.4	91.4	996.6	14.28
Stockton	37.96	-121.29	58.81	60	1022.6	6.09
Tampa	27.95	-82.46	74.83	64.2	1020.2	11.32
Toledo	39.86	-4.02	54.99	51.8	1029.8	10.65
Tucson	32.22	-110.97	75.47	12.4	1019	10.79
Tulsa	36.16	-95.99	55.87	44.6	1025.6	13.57

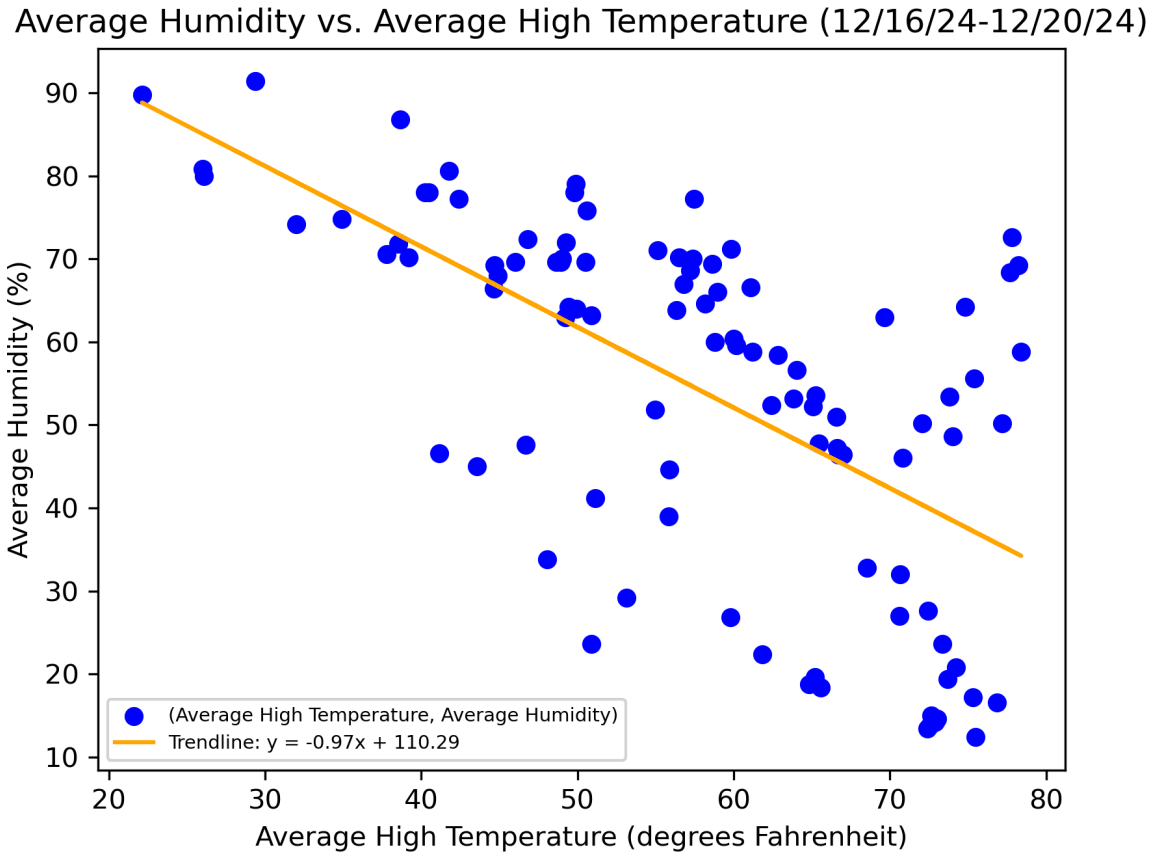


City	Latitude	Longitude	Average High Temperature (degrees Fahrenheit)	Average Humidity (%)	Average Air Pressure (hectopascals)	Average Wind Speed (miles/hour)
Philadelphia	39.95	-75.16	50.53	69.6	1023.8	12.69
Phoenix	33.45	-112.07	72.56	13.8	1020.2	4.32
Pittsburgh	40.44	-79.99	44.9	68	1021.8	11.52
Plano	33.01	-96.69	65.07	52.2	1024.8	12.87
Port St. Lucie	27.29	-80.35	77.69	68.4	1019.2	13.4
Portland	45.52	-122.67	49.88	79	1020	7.47
Raleigh	35.78	-78.64	61.21	58.8	1023	9.9
Reno	39.53	-119.81	49.45	64.2	1025.4	6.74
Richmond	37.54	-77.43	56.34	63.8	1023	9.46
Riverside	33.95	-117.4	76.87	16.6	1020	6.4
Sacramento	38.58	-121.49	57.22	68.6	1022.8	7.46
Saint Paul	44.95	-93.09	26	80.8	1023.8	10.17
San Antonio	29.42	-98.5	72.08	50.2	1023.8	12.53
San Diego	32.72	-117.16	68.54	32.8	1020	6.32
San Francisco	37.78	-122.42	57.38	70	1022.4	8.7
San Jose	37.34	-121.89	63.82	53.2	1021.8	6.42
Santa Ana	33.75	-117.87	73.37	23.6	1020	6.71
Scottsdale	33.49	-111.93	72.41	13.4	1020.2	5.2
Seattle	47.6	-122.33	49.83	78	1018.8	8.43
Spokane	47.66	-117.42	38.67	86.8	1024.8	7.41
St. Louis	38.63	-90.24	49.21	63	1024	11.87
St. Petersburg	59.94	30.32	29.4	91.4	996.6	14.28
Stockton	37.96	-121.29	58.81	60	1022.6	6.09
Tampa	27.95	-82.46	74.83	64.2	1020.2	11.32
Toledo	39.86	-4.02	54.99	51.8	1029.8	10.65
Tucson	32.22	-110.97	75.47	12.4	1019	10.79
Tulsa	36.16	-95.99	55.87	44.6	1025.6	13.57
Virginia Beach	36.85	-75.98	57.45	77.2	1023.2	15.49
Washington	38.9	-77.04	50.91	63.2	1023.4	9.52
Wichita	37.69	-97.34	51.12	41.2	1025.8	14.18
Winston-Salem	36.1	-80.24	56.8	67	1023	9.82

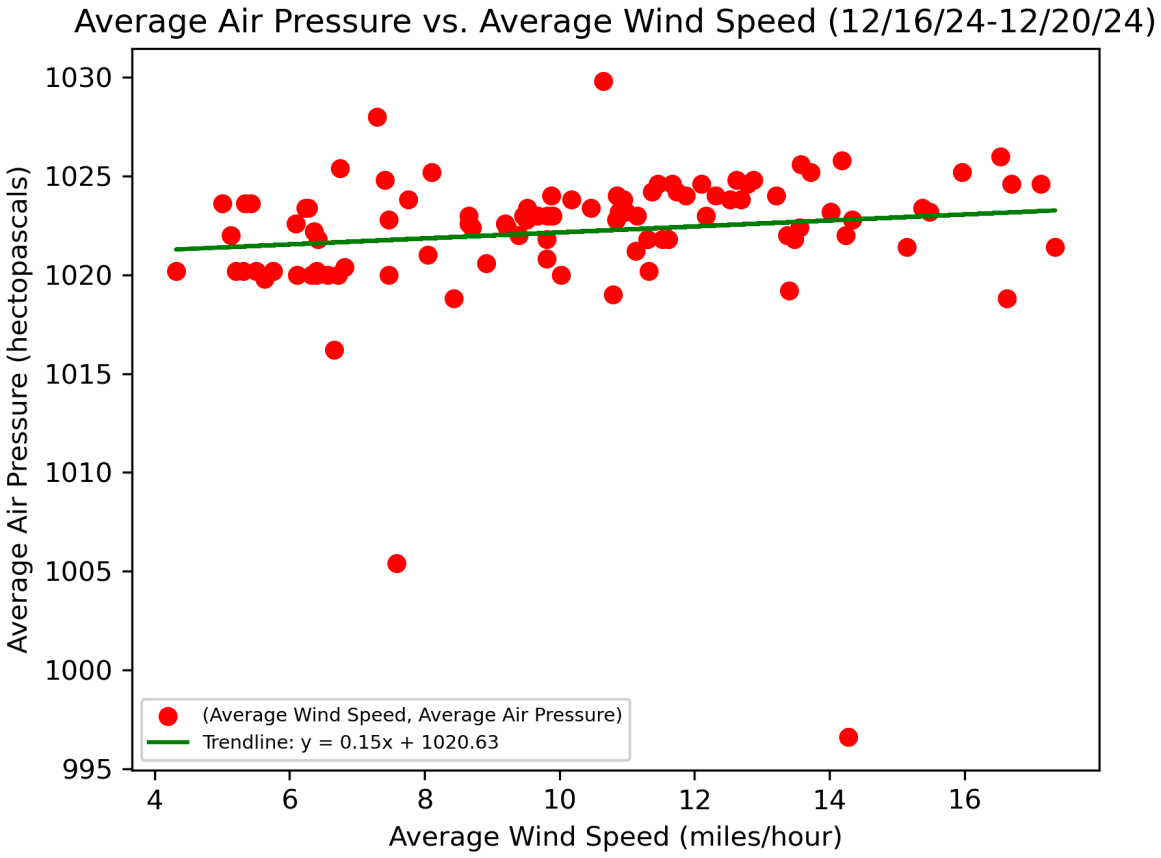
5. Visualizations We Created

We created three visualizations for the project. The visualizations and the descriptions of the respective visualizations are below:

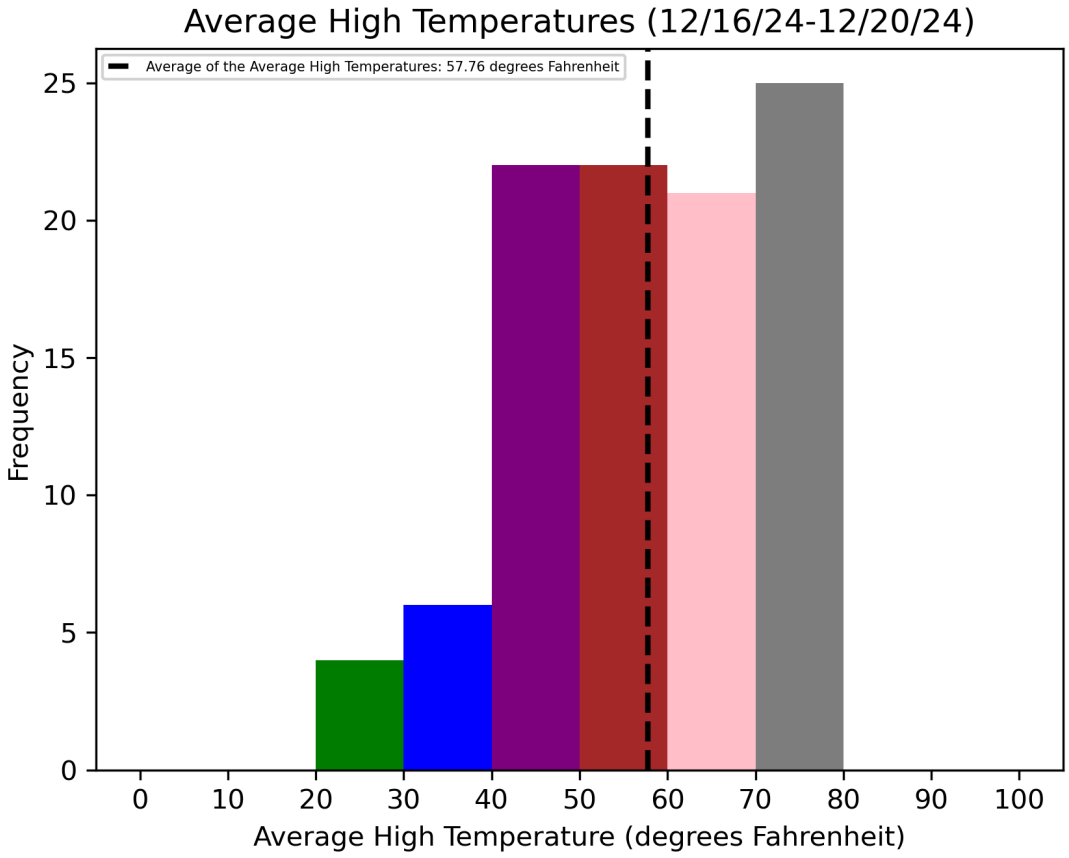
Visualization 1: A scatter plot, with a trendline, showing the relationship between average humidity and average high temperature over a period of five days (December 16, 2024 to December 20, 2024)



Visualization 2: A scatter plot, with a trendline, showing the relationship between average air pressure and average wind speed over a period of five days (December 16, 2024 to December 20, 2024)



Visualization 3: A histogram, with an indicator marker to mark the average of the average high temperatures, showing a frequency distribution of the average high temperatures over a period of five days (December 16, 2024 to December 20, 2024)



6. Instructions for Running the Code

NOTES:
Method 1 instructions work for any Anaconda environment for Python on any operating system.

Method 2 instructions work for a Mac that is running the Anaconda environment for Python. When you are instructed to type in a command in the terminal, wherever you see `username`, replace `username` with your actual computer user name.

`main.py`
The `main.py` file allows you to insert the top 100 cities in the United States by population to the Cities table of the database, insert the five dates for data analysis to the Dates table of the database, and the weather information for the five dates for the top 100 cities for analysis to the Weather table of the database.

Method 1: To run the `main.py` file, ensure the `main.py` file, the `visualizations.py` file, and the `calculations.py` file are in your working directory. Then, make sure that the `main.py` file is the currently open file in Visual Studio Code. Then, click on the "Play" button that is at the upper right corner of Visual Studio Code.

Method 2: To run the `main.py` file, ensure the `main.py` file, the `visualizations.py` file, and the `calculations.py` file are in your working directory.
Then, in the terminal, make sure you are currently in your working directory. In the terminal, type the following command:

```
/opt/anaconda3/bin/python
/Users/username/si206/project/si-206-final-project/main.py
```

The above command only inserts 25 cities to the Cities table of the database at a time. You need to run the above command an additional three times to add all the 100 cities to the Cities table of the database. The first time you run the above command, a file named `cities_weather_dates.db` will be created in the working terminal, which is the database file for the Cities table, Dates table, and Weather table. Every time you run the above command, you will get messages saying that the weather information for each city ID, the city ID's respective latitude, and the city ID's respective longitude has been added to the Weather table of the database.

`visualizations.py`

The `visualizations.py` file allows you to produce the visualizations for the database that were mentioned in the "5. Visualizations We Created" section of this report and save these produced visualizations to your working directory.

Method 1: To run the `visualizations.py` file, ensure the `main.py` file, the `visualizations.py` file, the `calculations.py` file, and the `cities_weather_dates.db` file are in your working directory. Then, make sure that the `visualizations.py` file is the currently open file in Visual Studio Code. Then, click on the "Play" button that is at the upper right corner of Visual Studio Code.

Method 2: To run the `visualizations.py` file, ensure that the `main.py` file, the `visualizations.py` file, the `calculations.py` file, and the `cities_weather_dates.db` file are in your working directory.
Then, in the terminal, make sure you are currently in your working directory. In the terminal, type the following command:

```
/opt/anaconda3/bin/python
/Users/username/si206/project/si-206-final-project/visualizations.py
```

The above command will produce the three visualizations and add them to your working directory. When you run the above command, you will get a message for each visualization that was added to your working directory.

`calculations.py`

The `calculations.py` file allows you to compute the average high temperature, average humidity, average air pressure, and average wind speed for the top 100 cities by population in the United States for the period of five days (December 16, 2024 to December 20, 2024) and save all the averages calculations to a CSV file.

Method 1: To run the `calculations.py` file, ensure the `main.py` file, the `visualizations.py` file, the `calculations.py` file, and the `cities_weather_dates.db` file are in your working directory. Then, make sure that the `calculations.py` file is the currently open file in Visual Studio Code. Then, click on the "Play" button that is at the upper right corner of Visual Studio Code.

Method 2: To run the `calculations.py` file, ensure that the `main.py` file, the `visualizations.py` file, the `calculations.py` file, and the `cities_weather_dates.db` file are in your working directory.
Then, in the terminal, make sure you are currently in your working directory. In the terminal, type the following command:

```
/opt/anaconda3/bin/python
/Users/username/si206/project/si-206-final-project/calculations.py
```

The above file will produce a CSV file called `average_calculations.csv` that contains all the average calculations for the top 100 cities in the United States by population, and the CSV file will be added to your working directory. When you run the above command, you will get a message that the CSV file for all the average calculations has been added to your working directory.

7. Documentation for Each Function

Functions in the `main.py` file:

`def set_up_database(database_name):`

Input(s):

`database_name`: The name of the database file that will be created and then added to the working directory

Function Description:

This function sets up the database and the necessary `cursor` and `conn` variables that will be used for the later functions in the `main.py` file.

Output(s):

`cursor`: The cursor variable for the database in the program

`conn`: The connection variable for the database in the program

`def get_top_one_hundred_cities(url):`

Input(s):

`url`: The link of the URL that will be used to extract the top 100 cities in the United States by population using BeautifulSoup

Function Description:

This function retrieves the top 100 cities in the United States by population using BeautifulSoup and inserts these top 100 cities to a list.

Output(s):

`cities_list`: A list of the top 100 cities in the United States by population

`def find_latitude_longitude(cities_list, key):`

Input(s):

`cities_list`: A list of the top 100 cities in the United States by population

`key`: The API key that is used to get the latitude and longitude data for the top 100 cities in the United States by population with the Geocoding API from API-Ninjas.

Function Description:

This function finds the latitude and longitude data for the top 100 cities in the United States by

population and inserts the respective latitude and longitude data for each city in a tuple; all the tuples are added to a list.
Output(s):
`latitude_longitude_list`: A list of tuples, where each tuple contains a city from the top 100 cities in the United States by population, the city's respective latitude value, and the city's respective longitude value

```
def cities_latitude_longitude_operation(cursor, conn, cities_list,
latitude_longitude_list):
```

Input(s):
`cursor`: The cursor variable for the database in the program
`conn`: The connection variable for the database in the program
`cities_list`: A list of the top 100 cities in the United States by population
`latitude_longitude_list`: A list of tuples, where each tuple contains a city from the top 100 cities in the United States by population, the city's respective latitude value, and the city's respective longitude value
Function Description:
This function adds all the top 100 cities in the United States by population, each city's respective latitude value, and each city's respective longitude value to the Cities table of the database
Output(s):
Nothing

```
def weather_forecast_operation(cursor, conn, key):
```

Input(s):
`cursor`: The cursor variable for the database in the program
`conn`: The connection variable for the database in the program
`key`: The API key that is used to get the weather data for the top 100 cities in the United States by population with the OpenWeather API.
Function Description:
This function adds all the five days (December 16, 2024 to December 20, 2024) for weather data analysis to the Dates table of the database. This function also adds all the weather data (high temperature, humidity, air pressure, and wind speed) for each of the five days for the 100 cities to the Weather table of the database.
Output(s):
Nothing

Functions in the `visualizations.py` file:

`def average_humidity_average_high_temperature_operation(database, output_file):`

Input(s):

`database`: The database file that is in the working directory

`output_file`: The name of the output file for the scatter plot

Function Description:

This function creates a scatter plot with a trendline that shows the relationship between the average high temperature and average humidity for the top 100 cities in the United States by population over a series of five days (December 16, 2024 to December 20, 2024).

Output(s):

Nothing

`def average_air_pressure_average_wind_speed_operation(database, output_file):`

Input(s):

`database`: The database file that is in the working directory

`output_file`: The name of the output file for the scatter plot

Function Description:

This function creates a scatter plot with a trendline that shows the relationship between the average wind speed and the average air pressure for the top 100 cities in the United States by population over a series of five days (December 16, 2024 to December 20, 2024).

Output(s):

Nothing

`def average_high_temperatures_operation(database, output_file):`

Input(s):

`database`: The database file that is in the working directory

`output_file`: The name of the output file for the histogram

Function Description:

This function creates a histogram that shows a frequency distribution of the average temperature for the top 100 cities in the United States by population over a series of five days (December 16, 2024 to December 20, 2024).

Output(s):

Nothing

Functions in the `calculations.py` file:
`def calculate_averages(database, output_file):`
Input(s):
`database`: The database file that is in the working directory
`output_file`: The name of the output file for the histogram
Function Description:
This function calculates the average values for all the weather data for each of the top 100 cities in the United States by population; all the average values for all 100 cities are written to a CSV file.
Output(s):
Nothing

8. Resources

Date	Issue Description	Location of Resource	Result (did it solve the issue?)
December 8, 2024	Use an API to obtain the respective latitude value and longitude value for each of the top 100 cities in the United States by population	API Ninjas - Geocoding https://api-ninjas.com/api/geocoding	Yes. We were able to obtain the respective latitude value and longitude value for each of the top 100 cities in the United states by population.
December 8, 2024	Use an API to obtain the high temperature, humidity, air pressure, and wind speed weather data for each of the 100 cities over a period of five days (December 16, 2024 to December 20, 2024)	OpenWeather API - Daily Forecast 16 Days https://openweathermap.org/forecast16	Yes. We were able to obtain the respective weather data for each of the 100 cities over a period of five days.
December 2, 2024	Use BeautifulSoup to gather the top 100 cities in the United States by population	Wikipedia - List of United States Cities by Population https://en.wikipedia.org/wiki/List_of_United_States_cities_by_population	Yes. We were able to obtain the top 100 cities in the United States by population.

December 2, 2024	Use an API to gather the average high temperature in degrees Fahrenheit on September 1, 2023 for each of the top 100 cities by population	WeatherStack API - https://weatherstack.com/?utm_source=Github&utm_medium=Referral&utm_campaign=Public-apis-repo-Best-sellers	No. We ran out of API uses for this API, and we had to use a different API to get the weather data.
December 2, 2024	Use an API to gather the average rainfall in inches on September 1, 2023 for each of the top 100 cities by population	OpenWeather API - https://developer.accuweather.com/accuweather-forecast-api/apis	No. We needed to get more weather data, not just rainfall data.