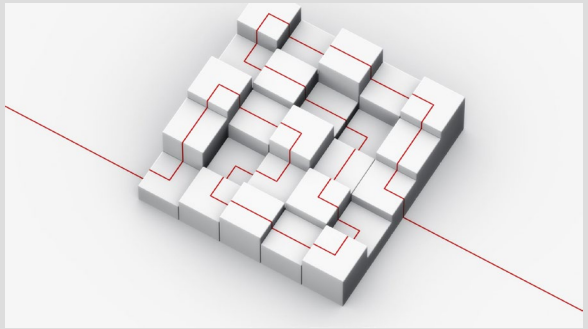
Developing a Project Using Python

John O'Leary

Objectives



- This project involves several steps including defining the input, processing, and output of a program.
- Create a flowchart to plan the steps of the process.
- Create a program that uses decisions and iteration with logical and relational expressions to process data.
- The project required installing and utilizing software including Anaconda, Spyder, and Linux.

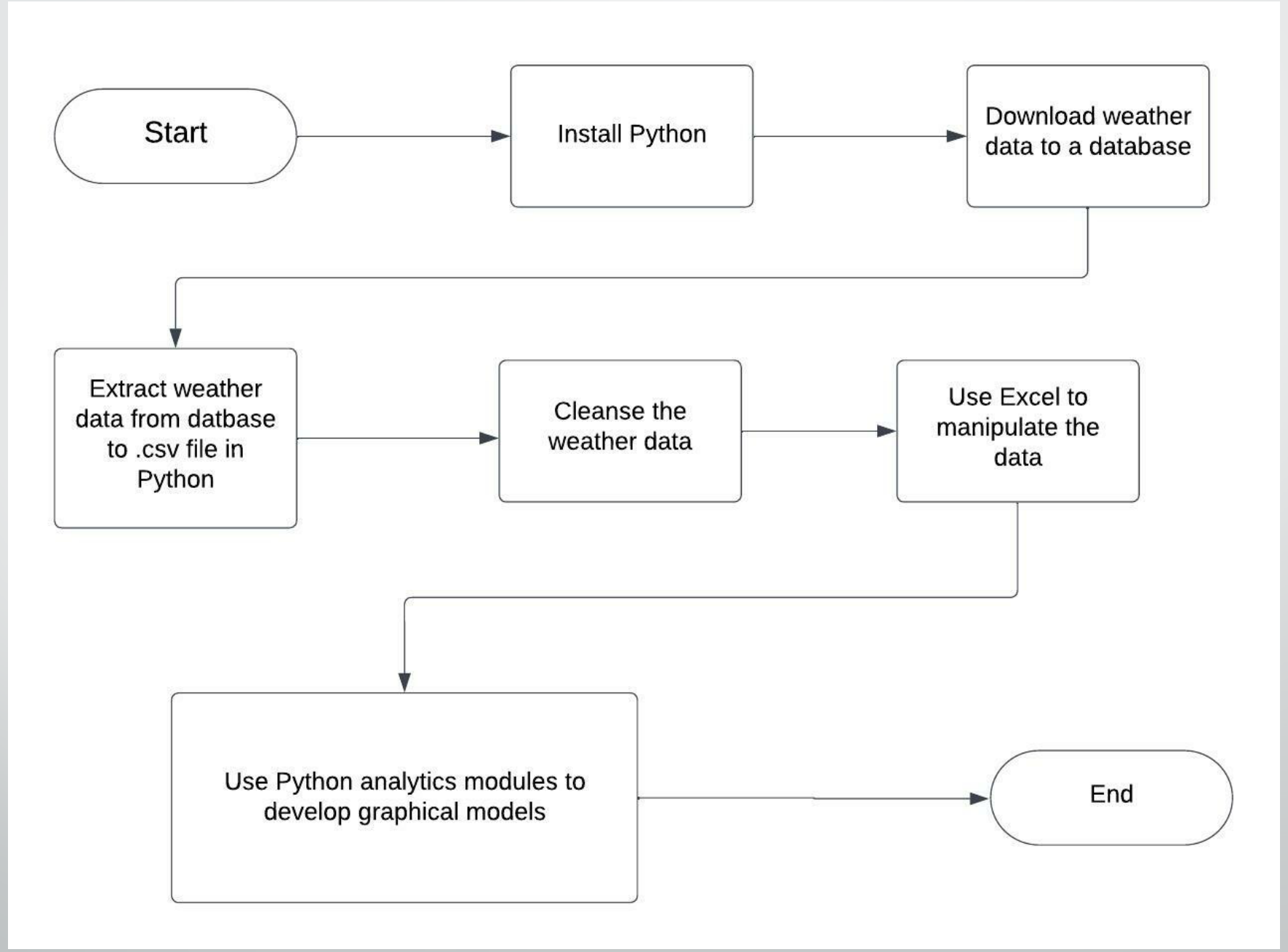
The initial steps to building a program in any programming language are first:

- Planning the steps required to accomplish the objectives. A common form of planning is the use of flowcharts.
- Once a plan is developed, the next step is to retrieve the data needed to be processed.

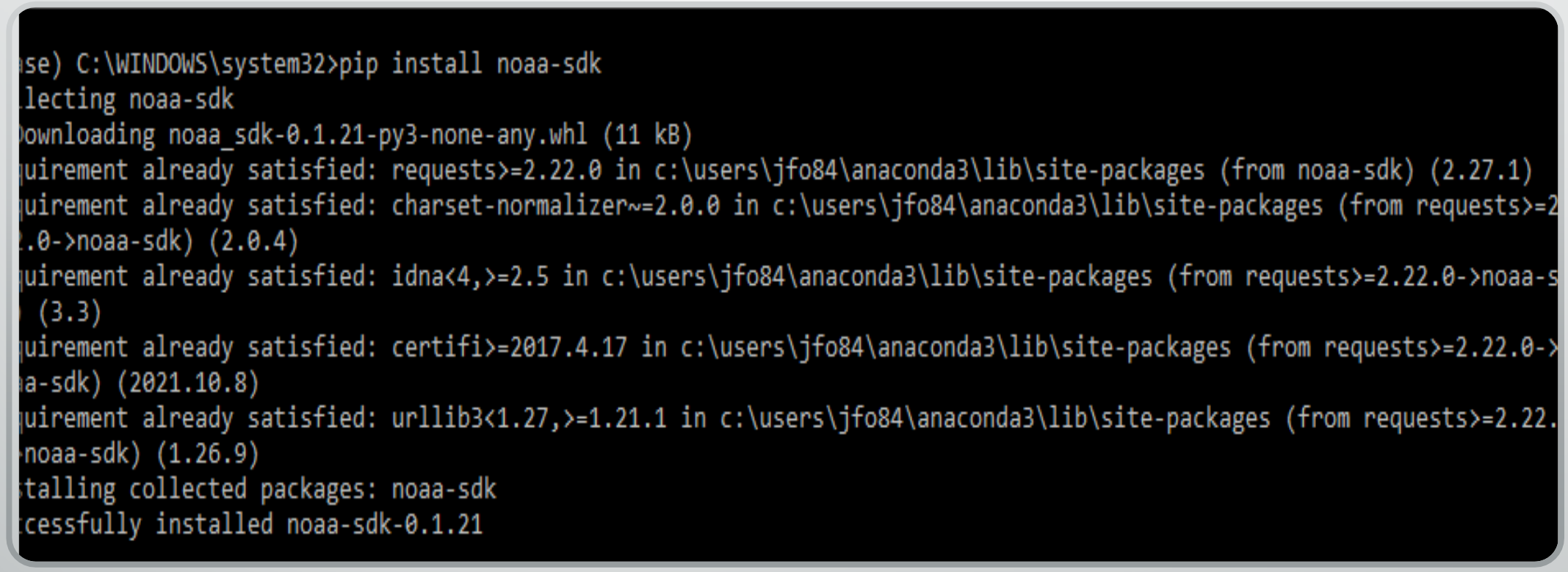
Using a Flowchart

The following slide contains a flowchart which includes the following processes:

- Install python
- Download weather data to a database.
- Extract weather data from database into a comma separated file with python
- Cleanse weather data
- Use Excel to manipulate data
- Use python data analytics modules to develop graphical models


Start
Install Python
Download weather data to a database
Extract weather data from datbase to .csv file in Python
Cleanse the weather data
Use Excel to manipulate the data
Use Python analytics modules to develop graphical models
End

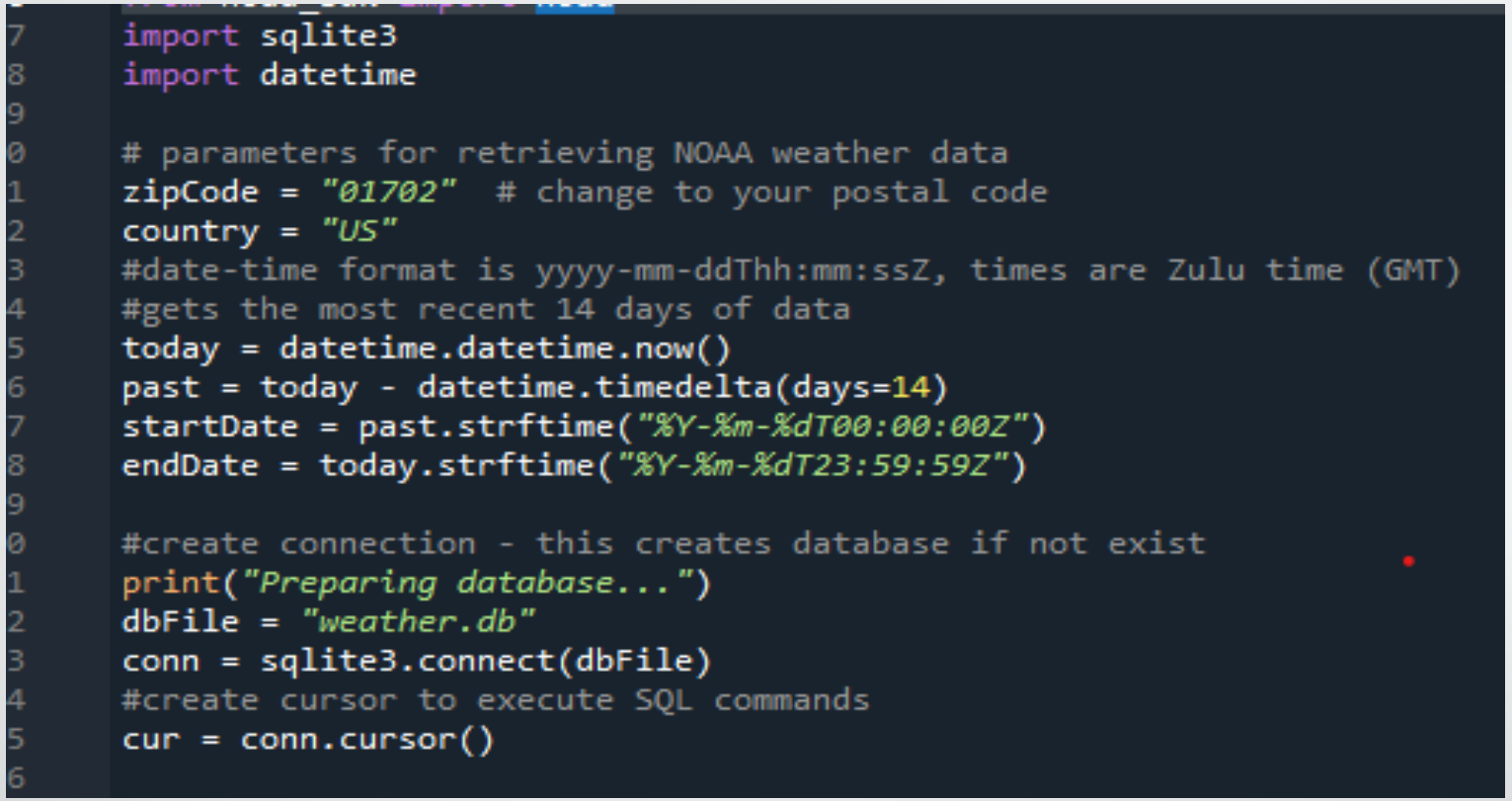
Screenshot of libraries from noaa-sdk being downloaded successfully.



Note: The libraries are necessary in order to use data downloaded from noaa in the next steps.

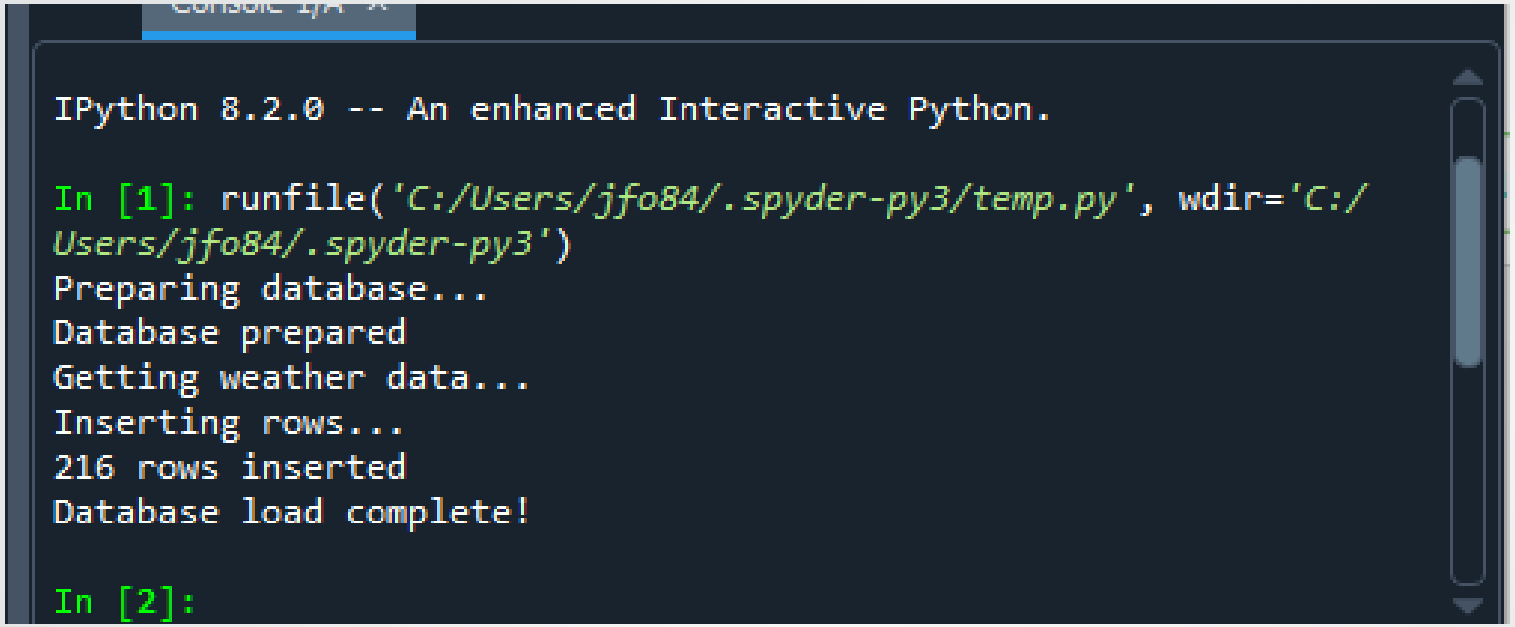
This step of the process involves downloading weather data with Python and building a weather database from that data.

The name of the download file is BuildWeatherDb.py and the screenshot includes a zip code to localize the data.



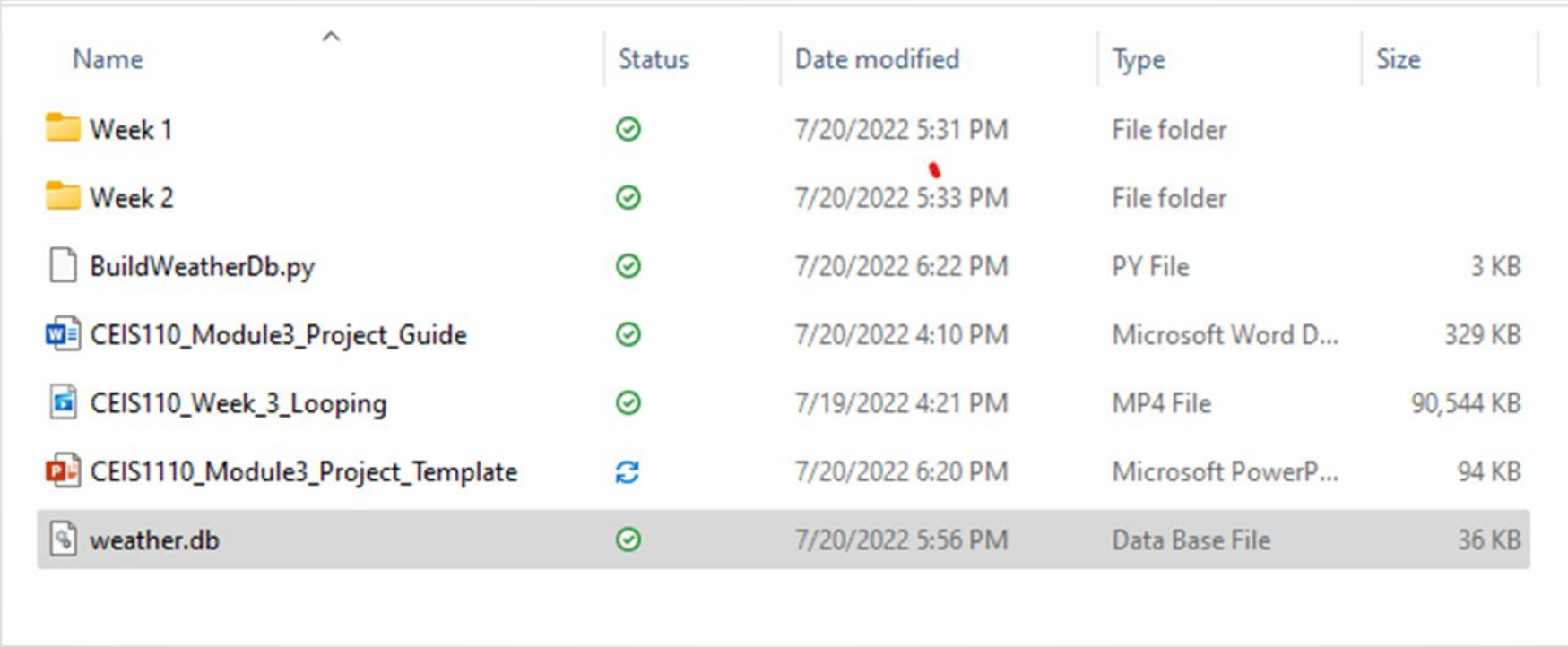
```
import sqlite3
import datetime

# parameters for retrieving NOAA weather data
zipCode = "01702"  # change to your postal code
country = "US"
#date-time format is yyyy-mm-ddThh:mm:ssZ, times are Zulu time (GMT)
#gets the most recent 14 days of data
today = datetime.datetime.now()
past = today - datetime.timedelta(days=14)
startDate = past.strftime("%Y-%m-%dT00:00:00Z")
endDate = today.strftime("%Y-%m-%dT23:59:59Z")

#create connection - this creates database if not exist
print("Preparing database...")
dbFile = "weather.db"
conn = sqlite3.connect(dbFile)
#create cursor to execute SQL commands
cur = conn.cursor()
```

BuildWeatherDb.py Code



Python Console

The screenshot shows that the code executed successfully.

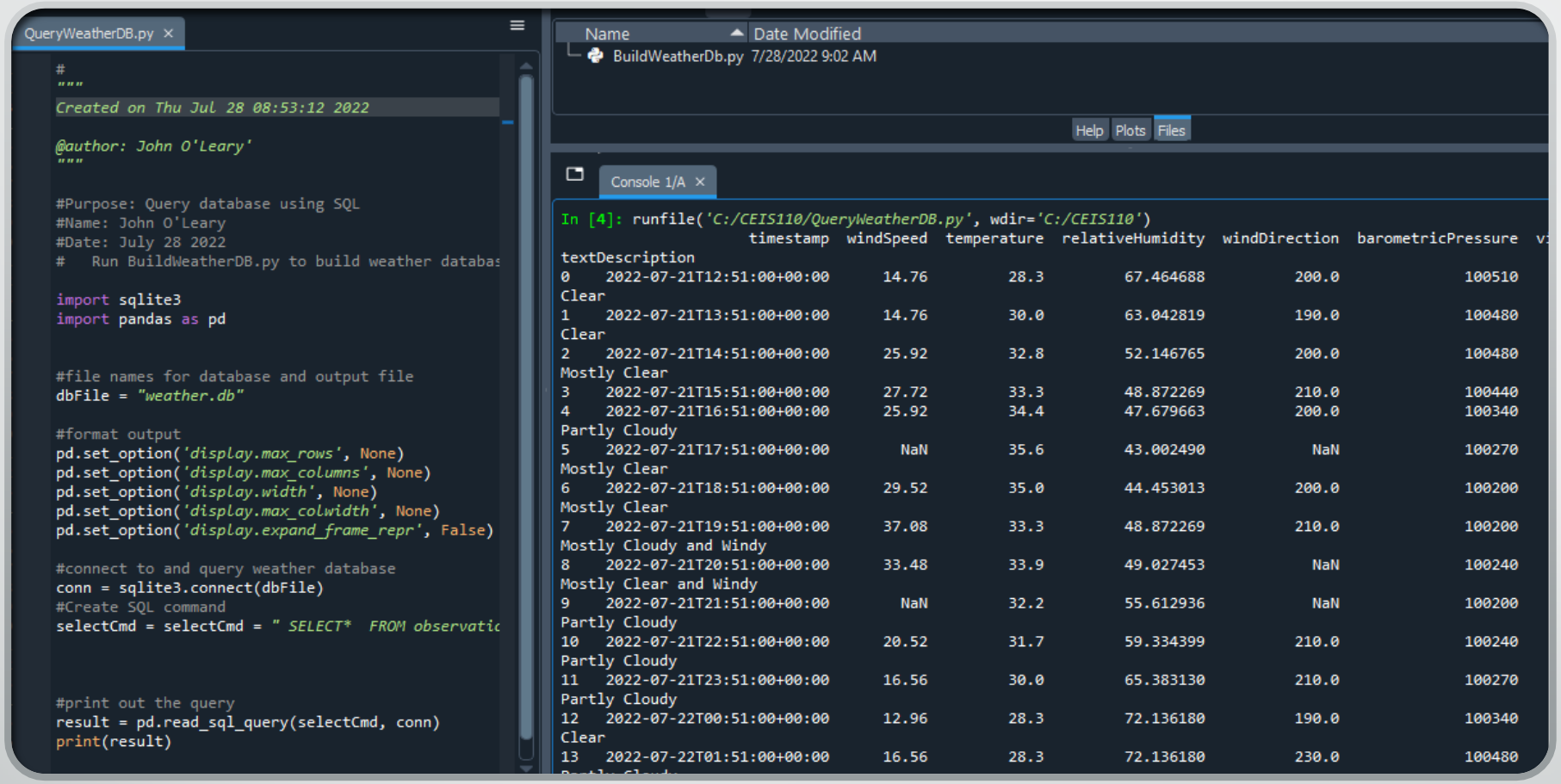


Name	Status	Date modified	Type	Size
Week 1	✓	7/20/2022 5:31 PM	File folder	
Week 2	✓	7/20/2022 5:33 PM	File folder	
BuildWeatherDb.py	✓	7/20/2022 6:22 PM	PY File	3 KB
CEIS110_Module3_Project_Guide	✓	7/20/2022 4:10 PM	Microsoft Word D...	329 KB
CEIS110_Week_3_Looping	✓	7/19/2022 4:21 PM	MP4 File	90,544 KB
CEIS1110_Module3_Project_Template	↻	7/20/2022 6:20 PM	Microsoft PowerP...	94 KB
weather.db	✓	7/20/2022 5:56 PM	Data Base File	36 KB

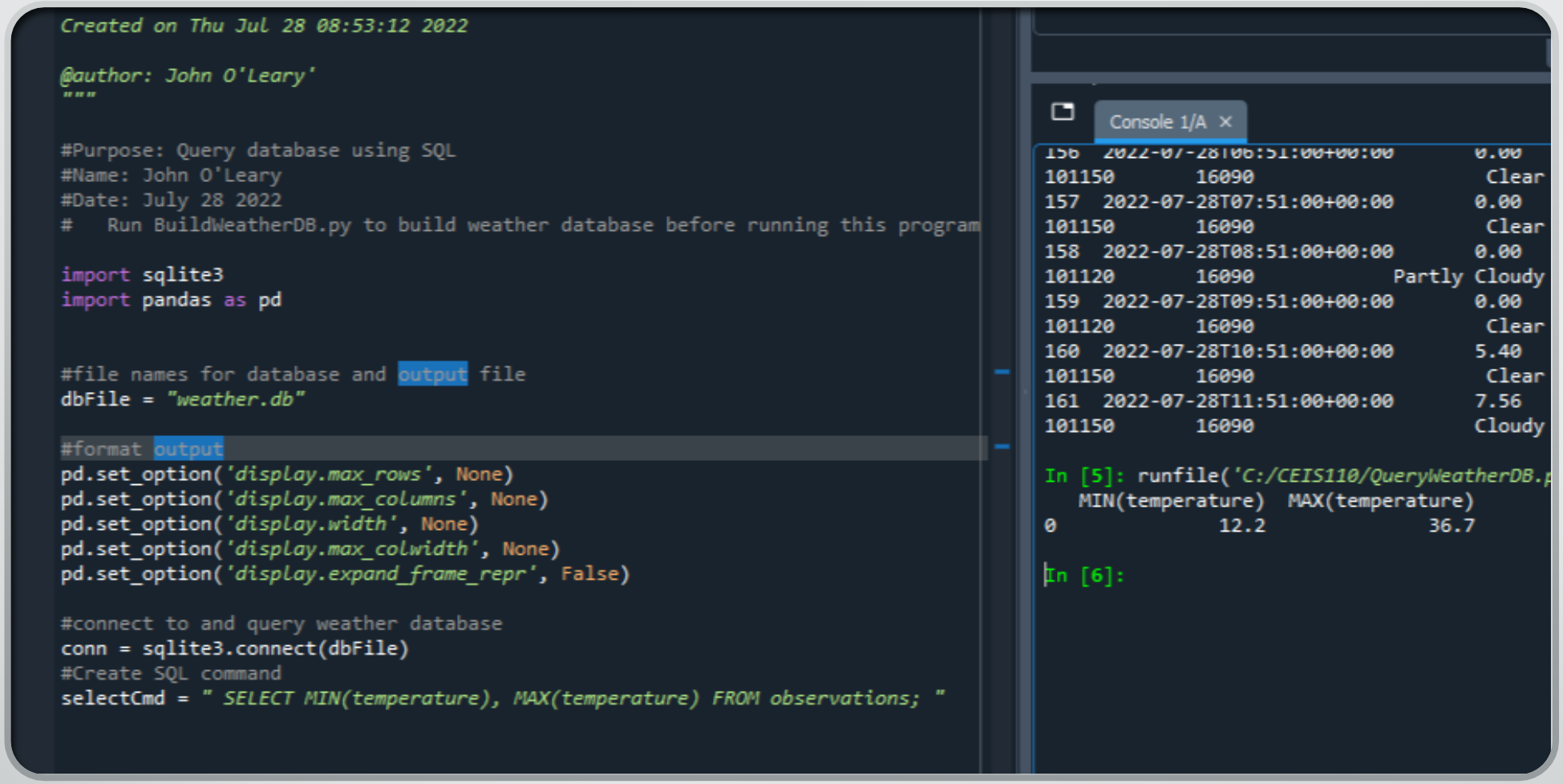
Weather.db file created

This part of the process includes using Python and SQL to create queries for retrieving data based on certain criteria.

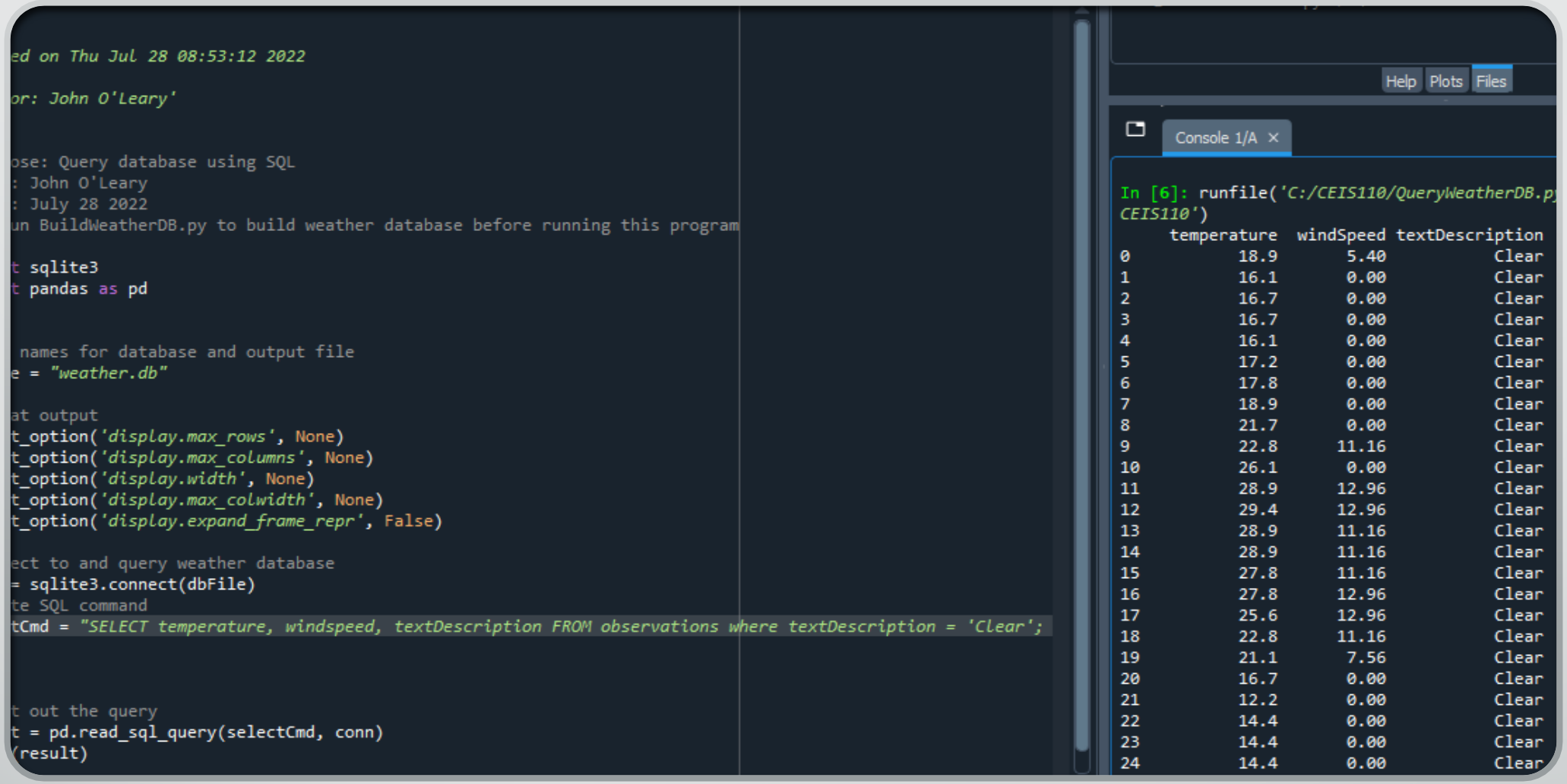
Purpose
➢ Query to retrieve all columns and rows
➢ Query to retrieve lowest and highest temperature
➢ Query to find all Clear days



Query to retrieve all columns and all rows



Query to retrieve lowest and highest temperatures



Query to retrieve data for all clear days

This step of building the program involves using Python to extract data from the weather database and use SQL to query and the manipulate the data. Then create a graphical representation of the extracted data in Excel.

```
#Purpose: Extract temperature, humidity data from weather database into CSV file
#Name: John O'Leary
#Date: August 1, 2022
#    Run BuildWeatherDB.py to build weather database before running this program

import sqlite3

#convert Celsius temperature to Fahrenheit
def convertCtoF(tempC):
    return (tempC*9.0/5.0) + 32.0

#file names for database and output file
dbFile = "weather.db"
output_file_name='formatdata.csv'

#connect to and query weather database and
dbFile = "weather.db"
conn = sqlite3.connect(dbFile)
#create cursor to execute SQL commands
cur = conn.cursor()
selectCmd = """ SELECT temperature, relativeHumidity FROM observations
                ORDER BY timestamp; """
cur.execute(selectCmd)
allRows = cur.fetchall()
#limit the number of rows output to half
rowCount = len(allRows)//2 # double slash does integer division
rows = allRows[:rowCount]

#write data to output file
with open(output_file_name,"w+") as outf:
    outf.write('Celsius,Fahrenheit,Humidity')
    outf.write('\n')
    for row in rows:
```

Python code used to extract the data.

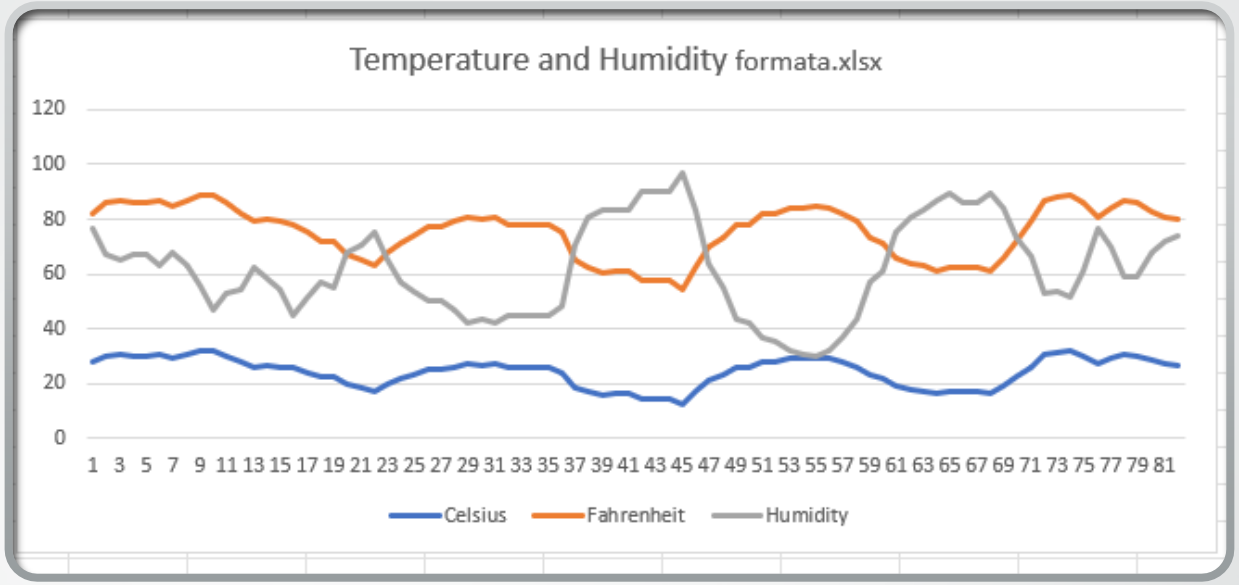
22.2	71.96	96.99591743
22.2	71.96	93.49564967
21.1	69.98	96.9712047
20.6	69.08	96.36167032
20	68	96.34534192
22.2	71.96	87.36520778
25	77	76.58907524
27.2	80.96	64.79910978
30	86	41.75007988
31.7	89.06	35.07281996
31.7	89.06	33.955371

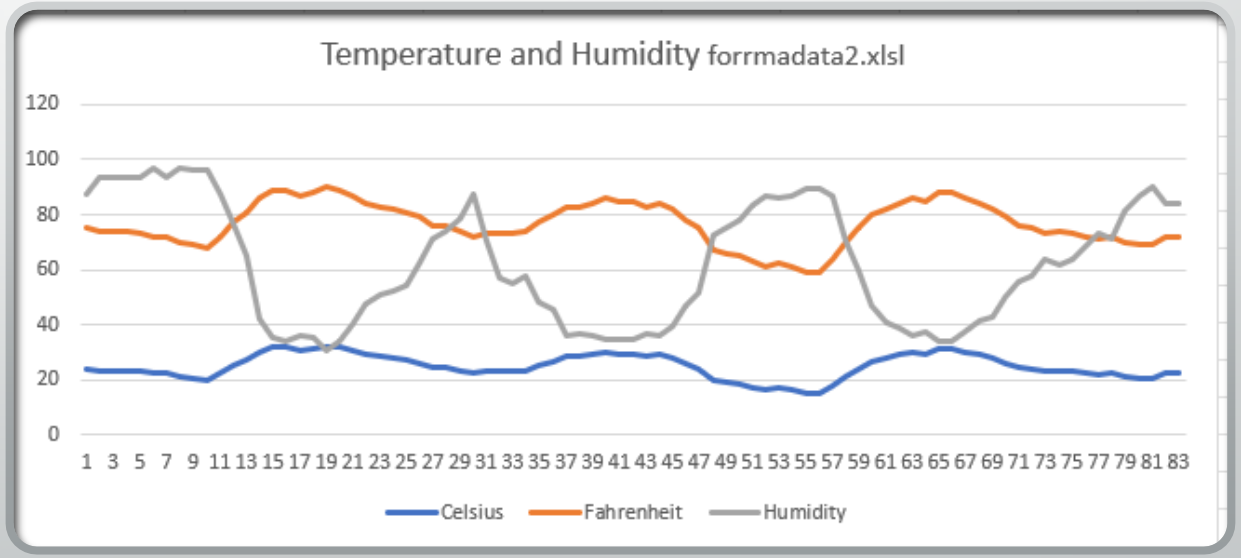
30	86	67.391164
30	86	67.391164
30.6	87.08	63.172227
29.4	84.92	67.681664
30.6	87.08	63.172227
31.7	89.06	55.491942
31.7	89.06	46.678267
30	86	53.064596
27.8	82.04	54.534039
26.1	78.98	62.186891
26.7	80.06	58.167242
26.1	78.98	54.108554
25.6	78.08	45.033206

Retrieve and Convert Data to CSV Format

Temperature and Humidity (Line Graphs)

- Excel charts based on temperature and humidity data from database



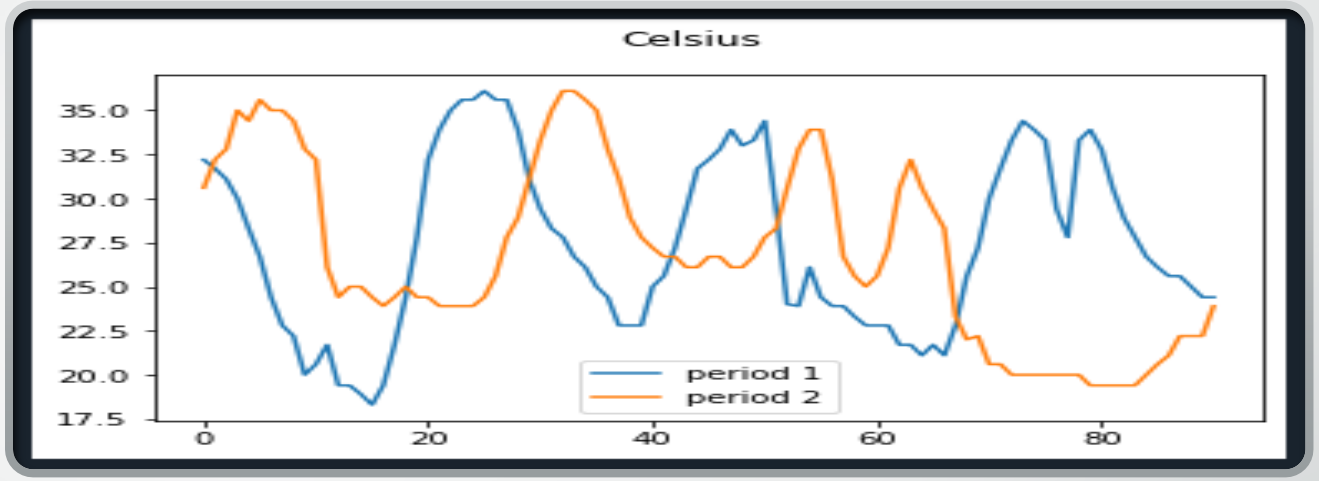


This section contains plots, graphs, Python code, and data analytic questions which were a part of the process.

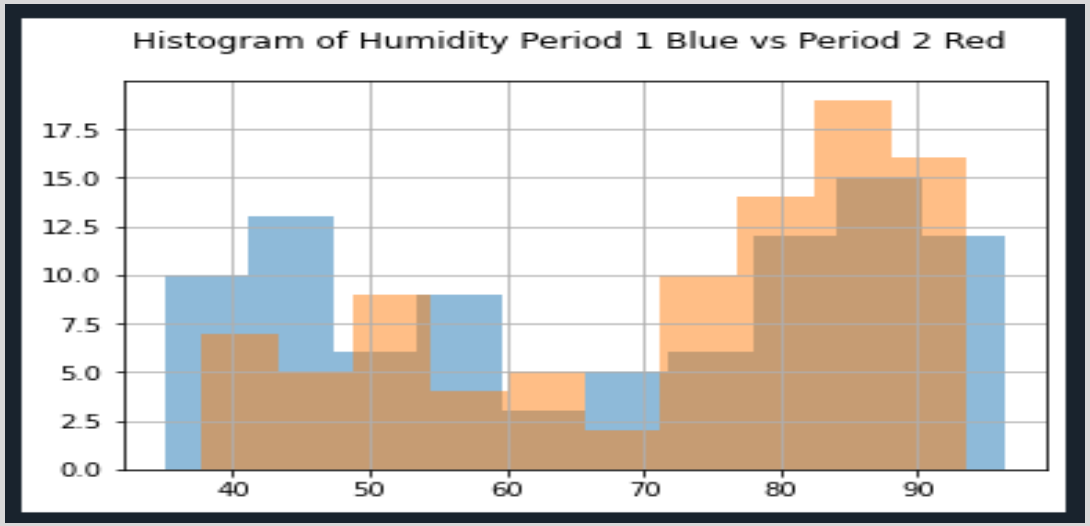
Plot #1

- This is a line chart depicting the variations in temperature in Celsius from one week to the next.

- The code is what was used to generate the output using Python.




```
#Purpose: Create Celsius plot comparing period 1 and period 2
#Name: John O'Leary
#Date: August 9, 2022
import pandas as pd
import matplotlib.pyplot as plt
df1 = pd.read_csv("formatdata.csv") #baseline data is period 1 (older)
df2 = pd.read_csv("formatdata2.csv") #data for period 2 (more recent)
plt.figure(); df1.Celsius.plot(label = 'period 1');
df2.Celsius.plot(label = 'period 2'); plt.legend(loc='best'); plt.suptitle('Celsius')
plt.show()
```



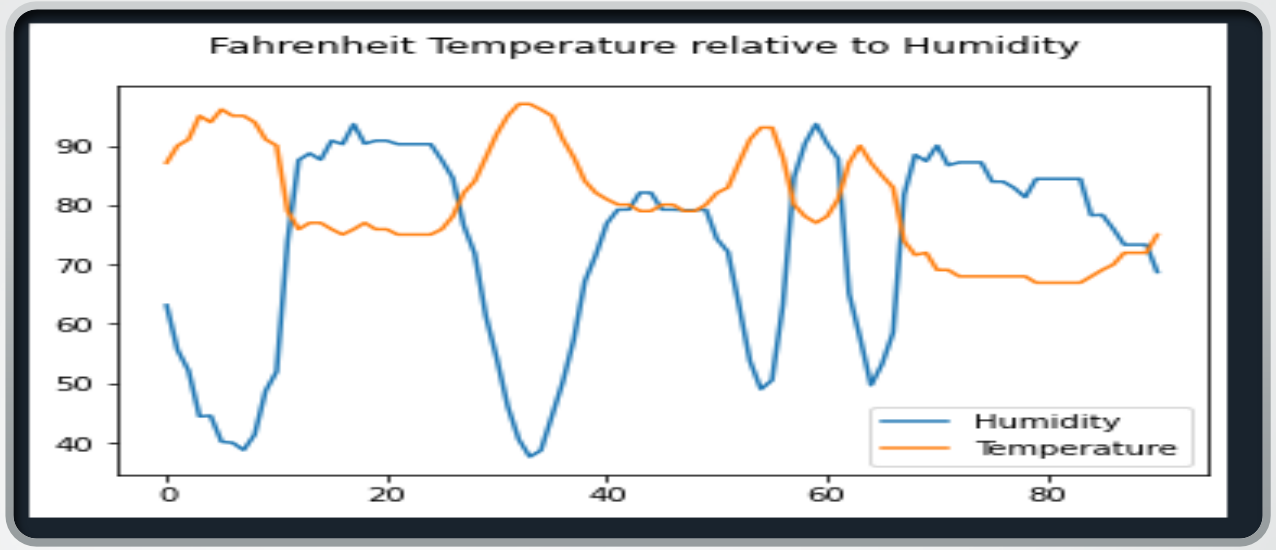

```
#Purpose: Create a histogram of humidity data comparing both periods
#Name: John O'Leary
#Date: August 11, 2022
import pandas as pd
import matplotlib.pyplot as plt
df1 = pd.read_csv("formatdata.csv")
df1['Humidity'].hist(bins=10, alpha=0.5)
df2 = pd.read_csv("formatdata2.csv")
df2['Humidity'].hist(bins=10, alpha=0.5);
plt.suptitle('Histogram of Humidity (Period 1 Blue vs. Period 2 Red)');
```

Plot #2

- Included is a histogram showing difference in humidity between week one and week two.
- The code is what was used to generate the graphical representation in Python.

Analysis

- My own question:
- What is the correlation between temperatures and humidity?
- Answer supported by Chart:
- Higher temperatures most often coincide with low humidity ranges. Spikes in temperature are usually preceded by dropping humidity.




```
#Purpose: Create a line graph to see the correlation between
#temperature and humidity in period 2
#Name: John O'Leary
#Date: August 11, 2022
import pandas as pd
import matplotlib.pyplot as plt
df1 = pd.read_csv("formatdata.csv") #data for period 1 (older)
df2 = pd.read_csv("formatdata2.csv") #data for period 2 (more recent)
plt.figure(); df2.Humidity.plot(label = 'Humidity ')
df2.Fahrenheit.plot(label = 'Temperature')
plt.legend(loc='best')
plt.suptitle('Fahrenheit Temperature relative to Humidity'); plt.show()
```

Prediction

Based on the current conditions, and if they follow the pattern displayed in the line graph, I expect that in the next few days, the humidity will continue to rise. It's currently at 52% and the heat index is at 73 degrees.

Another factor I considered in my prediction is based on my geographical location. Wind speed and direction, as well as barometric pressure have an impact on temperature and humidity levels. Therefore, here in New England the usual pattern does not always follow that when temperatures are high, humidity is low.

Challenges

The challenges to this project were the initial introduction to Python and learning the many syntax rules and commands.

At times, the error codes which were generated were difficult to understand and find the reason for the error message.

Learning to develop skills to plan and create programs is something which will take time.

The more I learned about using Python, the more I found I needed to learn.

Career Skills

Adding Linux to my skillset is a tremendous advantage for opening new doors in the IT job market.

Critical thinking is an important aspect of planning and creating programs and was very much exercised in this project.

Using Anaconda and Spyder are new skills I gained from this project.

Using Excel to create graphical models based on data is another new skill I acquired during this project.

My communications skills were improved every week during discussion postings and live lectures.

Conclusion

- In conclusion I want to thank Professor Boulware for his excellent and readily available guidance and sharing his wealth of knowledge and expertise on the subject. I also want to thank the other Professors, many of whom I gained some valuable information from. This course is one of the most difficult yet rewarding courses I've taken so far.

- This project is a representation of everything I learned during this course. I hope to continue learning about Python and have opportunities to put the knowledge to good use.

Thank you for watching!