Puja Roy 3/18/22

CET 4900 - OL60

Internship Journal Entry #8

Throughout this week of my internship, I worked on processing the large-scale datasets of the New York Urban Hydro-Meteorological Testbed (NY-uHMT) weather station. Since it's commonly challenging to process multiple files at once using Python, I conducted research on the best practices of processing large data files. My research mostly comprised of reading various technical articles and videos based on Python. As you can see in Fig 1-6 below, I wrote Python scripts to process 29+ NY-uHMT files. In Fig 2, I wrote code that specifies the location of where the NY-uHMT files are in my local machine and assigning them to a variable called path. This line of coe stores all the file names in a list. Then, I assigned a variable called filelist to empty brackets and wrote a for loop. These lines of code append the file names to the list and prints all the file names. There are various types of Python code that helps process large datasets. Throughout my research, I learned that a few lines of code using any type of programming language can solve a problem so easily. It helps in avoiding manual processes of workflow.

```
In [3]: import os

        path = 'C:\\Users\\prati\\UHMTProject\\Data'
        # Store all the file names in this list
        filelist = []

        for root, dirs, files in os.walk(path):
            for file in files:
                # Append the file name to the list
                filelist.append(os.path.join(root,file))

        #print all the file names
        for name in filelist:
            print(name)
```

Figure 1 – Python code for processing multiple files

```
C:\Users\prati\UHMTProject\Data\Astoria.txt
C:\Users\prati\UHMTProject\Data\Baisley_Park.txt
C:\Users\prati\UHMTProject\Data\Bay_View.txt
C:\Users\prati\UHMTProject\Data\Dyckman.txt
C:\Users\prati\UHMTProject\Data\East_River.txt
C:\Users\prati\UHMTProject\Data\Far_Rockaway.txt
C:\Users\prati\UHMTProject\Data\Haber_Coney_Island.txt
C:\Users\prati\UHMTProject\Data\Middletown.txt
C:\Users\prati\UHMTProject\Data\Polo_Ground.txt
C:\Users\prati\UHMTProject\Data\QueensBG.txt
C:\Users\prati\UHMTProject\Data\QueensboroughCC.txt
C:\Users\prati\UHMTProject\Data\RonaldEdmondsMS.txt
C:\Users\prati\UHMTProject\Data\Site10_BayView_Fifteen.txt
C:\Users\prati\UHMTProject\Data\Site11_Baisley_Park_Fifteen.dat.txt
C:\Users\prati\UHMTProject\Data\Site12_East_River_Fifteen.dat.txt
C:\Users\prati\UHMTProject\Data\Site13_Astoria_Fifteen.dat.txt
C:\Users\prati\UHMTProject\Data\Site14_Haber_Coney_Island_Fifteen.dat.txt
C:\Users\prati\UHMTProject\Data\Site15_Walt_Whitman_MS_Fifteen.dat.txt
C:\Users\prati\UHMTProject\Data\Site16_JHS_High_School_Fifteen.dat.txt
C:\Users\prati\UHMTProject\Data\Site16_JHS_High_School_Table2.dat.txt
C:\Users\prati\UHMTProject\Data\Site1_Queens_Botanical_Garden_Fifteen.dat.txt
C:\Users\prati\UHMTProject\Data\Site2_Queensborough_Community_College_Fifteen.dat.txt
C:\Users\prati\UHMTProject\Data\Site3_Ronald_Edmonds_Learning_Center_Fifteen.dat.txt
C:\Users\prati\UHMTProject\Data\Site5_Middletown_Houses_Fifteen.dat.txt
C:\Users\prati\UHMTProject\Data\Site5_Middletown_Plaza_Table2.dat.txt
C:\Users\prati\UHMTProject\Data\Site6_Dyckman_Houses_Fifteen.dat.txt
C:\Users\prati\UHMTProject\Data\Site7_Williamsburg_Houses_Fifteen.dat.txt
C:\Users\prati\UHMTProject\Data\Site8_Polo_Ground_Fifteen.dat.txt
C:\Users\prati\UHMTProject\Data\Site9_Far_Rockaway_Fifteen.dat.txt
C:\Users\prati\UHMTProject\Data\Williamsburg.txt
```

Figure 2 – Output of the Python code for processing multiple files

In [6]:
```
import os

def get_filepaths(directory):
    """
    This function will generate the file names in a directory
    tree by walking the tree either top-down or bottom-up. For each
    directory in the tree rooted at directory top (including top itself),
    it yields a 3-tuple (dirpath, dirnames, filenames).
    """
    file_paths = []  # List which will store all of the full filepaths.

    # Walk the tree.
    for root, directories, files in os.walk(directory):
        for filename in files:
            # Join the two strings in order to form the full filepath.
            filepath = os.path.join(root, filename)
            file_paths.append(filepath)  # Add it to the list.

    return file_paths  # Self-explanatory.

# Run the above function and store its results in a variable.
full_file_paths = get_filepaths("C:\\Users\\prati\\UHMTProject\\Data")
full_file_paths
```

Out[6]:
```
['C:\\Users\\prati\\UHMTProject\\Data\\Astoria.txt',
 'C:\\Users\\prati\\UHMTProject\\Data\\Baisley_Park.txt',
 'C:\\Users\\prati\\UHMTProject\\Data\\Bay_View.txt',
 'C:\\Users\\prati\\UHMTProject\\Data\\Dyckman.txt',
 'C:\\Users\\prati\\UHMTProject\\Data\\East_River.txt',
 'C:\\Users\\prati\\UHMTProject\\Data\\Far_Rockaway.txt',
 'C:\\Users\\prati\\UHMTProject\\Data\\Haber_Coney_Island.txt',
 'C:\\Users\\prati\\UHMTProject\\Data\\Middletown.txt',
 'C:\\Users\\prati\\UHMTProject\\Data\\Polo_Ground.txt',
 'C:\\Users\\prati\\UHMTProject\\Data\\QueensBG.txt',
 'C:\\Users\\prati\\UHMTProject\\Data\\QueensboroughCC.txt',
 'C:\\Users\\prati\\UHMTProject\\Data\\RonaldEdmondsMS.txt',
 'C:\\Users\\prati\\UHMTProject\\Data\\Site10_BayView_Fifteen.txt',
 'C:\\Users\\prati\\UHMTProject\\Data\\Site11_Baisley_Park_Fifteen.dat.txt',
 'C:\\Users\\prati\\UHMTProject\\Data\\Site12_East_River_Fifteen.dat.txt',
 'C:\\Users\\prati\\UHMTProject\\Data\\Site13_Astoria_Fifteen.dat.txt',
 'C:\\Users\\prati\\UHMTProject\\Data\\Site14_Haber_Coney_Island_Fifteen.dat.txt',
 'C:\\Users\\prati\\UHMTProject\\Data\\Site15_Walt_Whitman_MS_Fifteen.dat.txt',
 'C:\\Users\\prati\\UHMTProject\\Data\\Site16_JHS_High_School_Fifteen.dat.txt',
 'C:\\Users\\prati\\UHMTProject\\Data\\Site16_JHS_High_School_Table2.dat.txt',
 'C:\\Users\\prati\\UHMTProject\\Data\\Site1_Queens_Botanical_Garden_Fifteen.dat.txt',
 'C:\\Users\\prati\\UHMTProject\\Data\\Site2_Queensborough_Community_College_Fifteen.dat.txt',
 'C:\\Users\\prati\\UHMTProject\\Data\\Site3_Ronald_Edmonds_Learning_Center_Fifteen.dat.txt',
 'C:\\Users\\prati\\UHMTProject\\Data\\Site5_Middletown_Houses_Fifteen.dat.txt',
 'C:\\Users\\prati\\UHMTProject\\Data\\Site5_Middletown_Plaza_Table2.dat.txt',
 'C:\\Users\\prati\\UHMTProject\\Data\\Site6_Dyckman_Houses_Fifteen.dat.txt',
 'C:\\Users\\prati\\UHMTProject\\Data\\Site7_Williamsburg_Houses_Fifteen.dat.txt',
 'C:\\Users\\prati\\UHMTProject\\Data\\Site8_Polo_Ground_Fifteen.dat.txt',
 'C:\\Users\\prati\\UHMTProject\\Data\\Site9_Far_Rockaway_Fifteen.dat.txt',
 'C:\\Users\\prati\\UHMTProject\\Data\\Williamsburg.txt']
```

Figure 3 & 4 – Python code and output for processing multiple files in a directory

In [7]:
```
[pd.read_csv(file, delimiter='\t', encoding='UTF-8') for file in filelist]
```

Out[7]:
```
[     AST_date_time_1,AST_date_time_2,AST_AirTF,AST_RH,AST_Rainfall_Tot,AST_VWC1,AST_VWC2,AST_VWC3,AST_VWC4
0                    NaT,NaT,NaN,NaN,NaN,NaN,NaN,NaN,NaN
1                    NaT,NaT,NaN,NaN,NaN,NaN,NaN,NaN,NaN
2       22-Jan-2018,15:45:00,41.69786,81.72929,0,0,0,0,0
3       22-Jan-2018,16:00:00,41.70884,81.87734,0,0,0,0,0
4        22-Jan-2018,16:15:00,40.21433,82.8053,0,0,0,0,0
...                                                   ...
8241    18-Apr-2018,14:30:00,51.42863,37.65265,0,0,0,0,0
8242    18-Apr-2018,14:45:00,52.87369,36.37976,0,0,0,0,0
8243    18-Apr-2018,15:00:00,52.86819,37.29856,0,0,0,0,0
8244    18-Apr-2018,15:15:00,54.53577,35.68073,0,0,0,0,0
8245    18-Apr-2018,15:30:00,53.96159,35.65631,0,0,0,0,0

[8246 rows x 1 columns],
      BPK_date_time_1,BPK_date_time_2,BPK_AirTF,BPK_RH,BPK_Rainfall_Tot,BPK_VWC1,BPK_VWC2,BPK_VWC3,BPK_VWC4
0                    NaT,NaT,NaN,NaN,NaN,NaN,NaN,NaN,NaN
1                    NaT,NaT,NaN,NaN,NaN,NaN,NaN,NaN,NaN
2       03-Apr-2017,11:15:00,62.29402,57.56261,0,NaN,N...
3       03-Apr-2017,11:30:00,-10.74998,33.02047,0,NaN,...
```

Figure 5 – Python code that reads all the files in the filelist

In [10]:
```
df = pd.concat([pd.read_csv(file, delimiter='\t', encoding='UTF-8') for file in filelist])
df
```

Out[10]:

	AST_date_time_1,AST_date_time_2,AST_AirTF,AST_RH,AST_Rainfall_Tot,AST_VWC1,AST_VWC2,AST_VWC3,AST_VWC4	BPK_date_time_1,BPK_date_time_2,BPI
0	NaT,NaT,NaN,NaN,NaN,NaN,NaN,NaN,NaN	
1	NaT,NaT,NaN,NaN,NaN,NaN,NaN,NaN,NaN	
2	22-Jan-2018,15:45:00,41.69786,81.72929,0,0,0,0,0	
3	22-Jan-2018,16:00:00,41.70884,81.87734,0,0,0,0,0	
4	22-Jan-2018,16:15:00,40.21433,82.8053,0,0,0,0,0	
...	...	
22399	NaN	
22400	NaN	
22401	NaN	
22402	NaN	
22403	NaN	

1360518 rows × 30 columns

Figure 6 – Python Dataframe displaying concatenated (combined) files in the filelist