000_part_A

November 13, 2022

[1]:
```
from urlpath import URL
from pathlib import Path
import pandas as pd
import numpy as np
from urllib.request import urlopen
import csv
import os.path, time


# Write a function to access and save the data
def data_in(year):
    '''show the pandas dataframe, including the day of year, average␣
 ↪temperatures and stream
    save these into csv file '''

    #create a value called year, all the data should be basd on this
    year = year

    # for spacing
    dash = '\n----------------------------------'

    #firstly, get the date and tempreture data

    #Temperature url
    url = 'https://raw.githubusercontent.com/UCL-EO/geog0111/master/notebooks/
 ↪data/delNorteT.dat'

    #check error code: 200 is good
    print(f'trying to access the data from {URL(url).as_posix()}')
    if URL(url).get().status_code == 200:
        print(f'access well {dash}')

    else:
         return('fail to access')

    '''
    The code in line30-line39 is based on
```

```
    "Python's urllib.request for HTTP Requests"
    by Ian Currie
    https://realpython.com/urllib-request/

    My modifications have been to make the process more efficient by reducing a␣
↪temporary value.
    '''

    #open the url and read the data
    #store the data into response
    response = urlopen(url).read()

    #convert data to utf-8 format, which means that we can edit the data
    temp = response.decode('utf-8')

    #first use .splitline() to split the data
    #access CSV data by csv.reader()
    csvfile_T = csv.reader(temp.splitlines())

    #create a null list called urlData to store the tempreture data
    urlData = []

    #use for loop to store the data
    for r in csvfile_T:
        #r is each rows in csvfile_T
        #print(type(row))
        #it is necessary to split it and create a new list to store the data␣
↪temporarily
        R = str(r).split('\\t') #since the gap in each line is \\t
        urlData.append(R)

    #create lists maxT and minT to calculate avrT
    maxT = []
    minT = []
    avrT = []

    #create list date to represent the day of year
    date = []

    for u in urlData[1:]:
        #find the data of the typical year we want
        if str(year) not in u[0]:
            continue

        #u[0] is the day of year
        date.append(u[0])
```

```
        #u[1] and u[2] are maxt and mint respectively
        maxT.append(u[1])
        minT.append(u[2])



        #calculate average tempreture, using try and except since some maxt or␣
↪mint are missing
        try:
            #since the maxt and mint are string, we should change their type␣
↪first


            avrT.append((int(u[1])+int(u[2]))/2)

        except ValueError:
            #when it meets a value error, this means that u[1] or u[2] could␣
↪not change the type
            #because in the data set, we could see that some maxt and mint are M
            avrT.append('N/A')

        #break the loop if date reaches the end of a year.
        if '12-31' in u[0]:
            break

    #test
    # print(avrT[-10:])
    # print(date[-10:])


    # put in DataFrame
    df_date = pd.DataFrame(date)
    df_averageT = pd.DataFrame(avrT)









    #now we begin to access the stream data from the USGS

    #url2 is stream discharge from the USGS
    #we could access the data about a typical year we want, corresponding to␣
↪the value year we create before
```

```
    url2 = 'https://waterservices.usgs.gov/nwis/dv/?
↪sites=08220000&format=rdb&startDT='+str(year)+'-01-01&endDT='+str(year)+'-12-31&parameterCd

    #check error code: 200 is good
    print(f'trying to access the data from {URL(url2).as_posix()}')
    if URL(url2).get().status_code == 200:
        print(f'access well {dash}')

    # else:
    #     return('fail to access')

    #same method as line30-39 above, access from 'https://realpython.com/
↪urllib-request/'
    response2 = urlopen(url2).read()

    temp2 = response2.decode('utf-8')

    csvfile_S = csv.reader(temp2.splitlines())

    #create a null list called urlData2 to store the whole stream data
    urlData2 = []

    #the logic here is same as line36 - line41 above
    for r2 in csvfile_S:
        #print(type(r))
        R2 = str(r2).split('\\t')
       # print(R)
        urlData2.append(R2)

    #create a list called stream
    stream = []

    #it is necessary to get the data after NO.29 line in the document
    for u in urlData2[30:]:

        #use try and except store the stream data into the list called stream,␣
↪avoiding the value error
        try:
            stream.append(u[3])
        except ValueError:
            stream.append('N/A')

    #test
    #print(stream[:20])

    # put in DataFrame
    df_stream = pd.DataFrame(stream)
```

```
    #create some additional columns
    df = pd.DataFrame(date,columns=["the day of year"])
    df['average tempreture (Fahrenheit)'] = df_averageT
    df['stream discharge (ml/day)'] = df_stream
    # print(df)

    # save as csv without the index
    df.to_csv(Path('work/delNorte'+str(year)+'.csv'),index=False)
    print('finish saving as a csv document')


    # setup Path object for output file
    filename = Path('work/delNorte'+str(year)+'.csv')

    # check size:
    size = filename.stat().st_size

    # report
    print(f'file {filename} written: {size} bytes')

    #show modification time of the file
    '''
    The code in line195 is based on
    "How do I get file creation and modification date/times?"
    by Peter Mortensen and Bryan Oakley
    https://stackoverflow.com/questions/237079/
↪how-do-i-get-file-creation-and-modification-date-times
    '''
    print("last modified: %s" % time.ctime(os.path.getmtime(filename)),dash)


    return df


#print(data_in(2014))

# running help() for this function
help(data_in)
```

```
Help on function data_in in module __main__:

data_in(year)
    show the pandas dataframe, including the day of year, average temperatures
and stream
    save these into csv file
```

[2]:
```
#Demonstrate running the function to access and save the data (2016-2019)
# including showing the file size, modification date of the CSV files and the␣
 ↪pandas dataframe
for i in range(2016,2020):
    print(data_in(i))
```

```
trying to access the data from https://raw.githubusercontent.com/UCL-
EO/geog0111/master/notebooks/data/delNorteT.dat
access well
----------------------------------
trying to access the data from https://waterservices.usgs.gov/nwis/dv/?sites=082
20000&format=rdb&startDT=2016-01-01&endDT=2016-12-31&parameterCd=00060
access well
----------------------------------
finish saving as a csv document
file work/delNorte2016.csv written: 8207 bytes
last modified: Sun Nov 13 23:02:52 2022
----------------------------------
    the day of year average tempreture (Fahrenheit) stream discharge (ml/day)
0      ['2016-01-01                              13                        165
1      ['2016-01-02                            13.5                        170
2      ['2016-01-03                            15.5                        180
3      ['2016-01-04                            19.5                        190
4      ['2016-01-05                            20.5                        185
..              …                               …                          …
361    ['2016-12-27                              16                        175
362    ['2016-12-28                              14                        175
363    ['2016-12-29                              19                        180
364    ['2016-12-30                              21                        180
365    ['2016-12-31                              17                        180

[366 rows x 3 columns]
trying to access the data from https://raw.githubusercontent.com/UCL-
EO/geog0111/master/notebooks/data/delNorteT.dat
access well
----------------------------------
trying to access the data from https://waterservices.usgs.gov/nwis/dv/?sites=082
20000&format=rdb&startDT=2017-01-01&endDT=2017-12-31&parameterCd=00060
access well
----------------------------------
finish saving as a csv document
file work/delNorte2017.csv written: 8185 bytes
last modified: Sun Nov 13 23:02:54 2022
----------------------------------
    the day of year average tempreture (Fahrenheit) stream discharge (ml/day)
```

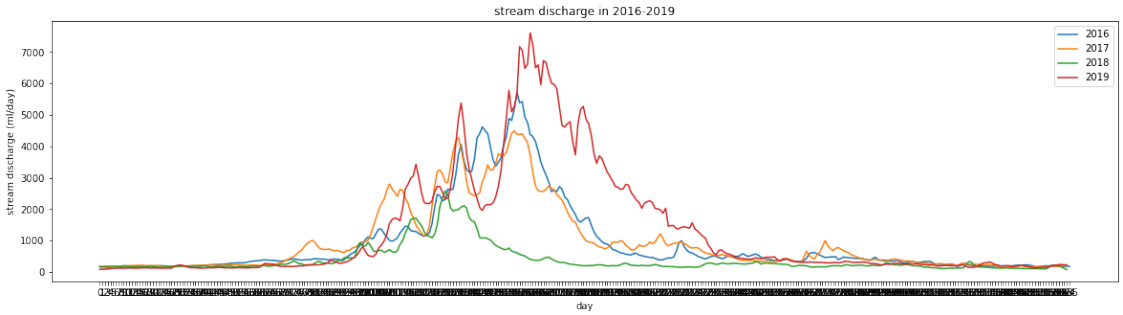
```
0      ['2017-01-01                                18.5                          185
1      ['2017-01-02                                20.5                          185
2      ['2017-01-03                                  18                          185
3      ['2017-01-04                                  19                          185
4      ['2017-01-05                                24.5                          190
..              …                                   …                            …
360    ['2017-12-27                                  30                          180
361    ['2017-12-28                                  30                          185
362    ['2017-12-29                                32.5                          190
363    ['2017-12-30                                31.5                          200
364    ['2017-12-31                                  36                          205

[365 rows x 3 columns]
trying to access the data from https://raw.githubusercontent.com/UCL-
EO/geog0111/master/notebooks/data/delNorteT.dat
access well
---------------------------------
trying to access the data from https://waterservices.usgs.gov/nwis/dv/?sites=082
20000&format=rdb&startDT=2018-01-01&endDT=2018-12-31&parameterCd=00060
access well
---------------------------------
finish saving as a csv document
file work/delNorte2018.csv written: 8136 bytes
last modified: Sun Nov 13 23:02:55 2022
---------------------------------
    the day of year  average tempreture (Fahrenheit) stream discharge (ml/day)
0      ['2018-01-01                              36.0                         175
1      ['2018-01-02                              36.0                         160
2      ['2018-01-03                              24.0                         155
3      ['2018-01-04                              28.5                         150
4      ['2018-01-05                              28.0                         155
..              …                                 …                           …
360    ['2018-12-27                              26.5                         210
361    ['2018-12-28                              24.0                         200
362    ['2018-12-29                               7.5                         160
363    ['2018-12-30                               6.5                         115
364    ['2018-12-31                               7.5                        80.0

[365 rows x 3 columns]
trying to access the data from https://raw.githubusercontent.com/UCL-
EO/geog0111/master/notebooks/data/delNorteT.dat
access well
---------------------------------
trying to access the data from https://waterservices.usgs.gov/nwis/dv/?sites=082
20000&format=rdb&startDT=2019-01-01&endDT=2019-12-31&parameterCd=00060
access well
---------------------------------
finish saving as a csv document
```

```
file work/delNorte2019.csv written: 8218 bytes
last modified: Sun Nov 13 23:02:56 2022
---------------------------------
    the day of year  average tempreture (Fahrenheit) stream discharge (ml/day)
0      ['2019-01-01                             13.0                       86.0
1      ['2019-01-02                              9.0                       88.0
2      ['2019-01-03                              3.5                       92.0
3      ['2019-01-04                              4.0                        100
4      ['2019-01-05                             14.0                        105
..              …                                …                          …
360    ['2019-12-27                             14.0                        220
361    ['2019-12-28                             16.0                        240
362    ['2019-12-29                             13.5                        240
363    ['2019-12-30                              2.5                        230
364    ['2019-12-31                             -0.5                        230

[365 rows x 3 columns]
```

[3]:
```python
# Visualise the data (2016-2019)
import pandas as pd
import matplotlib.pyplot as plt

# create a value as start year
y = 2016

# plot size
fig, axs = plt.subplots(1,1,figsize=(20,5))


# #use for loop
for i in range(0,4,1):

    # find and read the corresponding csv file
    dfPlot = pd.read_csv(Path('work/delNorte'+str(y+i)+'.csv'))

    #insert a column named row in dfPlot pandas dataframe
    #the total day in 2016 is different
    if i+y == 2016:
        row = [f'{m}' for m in range(0,366)]
    else:
        row = [f'{m}' for m in range(0,365)]

    dfPlot['row'] = row

    # plot y-data and set the label
    axs.plot(dfPlot['row'],dfPlot['stream discharge (ml/day)'],label= str(y+i))
```

```
# set legend now
axs.legend(loc='best')
# set the subplot title
axs.set_title('stream discharge in 2016-2019')
# y-label
axs.set_ylabel(f'stream discharge (ml/day)')
# x-label
axs.set_xlabel(f'day')



#help(plt.subplots())
```

[3]: Text(0.5, 0, 'day')



[ ]: