Data Acquisition : From File system

In [1]:

```
import pandas as pd
!dir
```

```
 C 드라이브의 볼륨에는 이름이 없습니다.
 볼륨 일련 번호: 3253-81AA

 C:₩Users₩Eunchae₩Desktop₩빅데이터응용 디렉터리

2021-05-02  오후 08:01    <DIR>          .
2021-05-02  오후 08:01    <DIR>          ..
2021-05-02  오후 07:43    <DIR>          .ipynb_checkpoints
2021-04-13  오후 06:04            78,679 1주차 201815069조은채 - Jupyter Notebook.pd
f
2021-04-13  오후 06:04           443,063 2주차 201815069 조은채 - Jupyter Notebook.p
df
2021-04-13  오후 05:59           499,230 3주차 - Jupyter Notebook.pdf
2021-04-12  오후 07:15           964,666 4주차 201815069 조은채 - Jupyter Notebook.p
df
2021-04-12  오후 08:48           127,071 5주차 - Jupyter Notebook.pdf
2021-04-13  오후 07:06           139,119 5주차 실습문제 - Jupyter Notebook.pdf
2021-04-13  오후 07:05             7,732 5주차 실습문제.ipynb
2021-04-19  오후 08:10           607,967 6주차 201815069조은채 - Jupyter Notebook.pd
f
2021-04-19  오후 08:09         1,221,451 6주차 201815069조은채.ipynb
2021-04-24  오후 09:46           202,709 6주차 실습문제 - Jupyter Notebook.pdf
2021-04-24  오후 09:46           212,049 6주차 실습문제.ipynb
2021-04-24  오후 11:01           535,901 7주차 201815069 조은채 - Jupyter Notebook.p
df
2021-04-24  오후 10:59           969,347 7주차 201815069 조은채.ipynb
2021-05-02  오후 07:35            92,065 7주차 실습문제 - Jupyter Notebook.pdf
2021-05-02  오후 07:35           109,896 7주차 실습문제.ipynb
2021-05-02  오후 08:01               706 8주차 201815069.ipynb
2021-04-26  오후 05:10             3,730 accident.csv
2021-04-12  오후 06:32         2,084,696 census.csv
2021-04-26  오후 05:10                58 ex1.csv
2021-04-26  오후 05:10                42 ex2.csv
2021-04-26  오후 05:10               163 ex4.csv
2021-04-26  오후 05:10                78 ex5.csv
              22개 파일           8,300,418 바이트
               3개 디렉터리  425,797,394,432 바이트 남음
```

In [2]:

```
!type ex1.csv
```

```
a,b,c,d,message
1,2,3,4,hello
5,6,7,8,world
9,10,11,12,foo
```

In [3]:

```
df = pd.read_csv('ex1.csv')
df
```

Out[3]:

	a	b	c	d	message
0	1	2	3	4	hello
1	5	6	7	8	world
2	9	10	11	12	foo

In [5]:

```
df = df.set_index('message')
#df.set_indet('message', inplace = True) 랑 위랑 같음
df
```

Out[5]:

	a	b	c	d
message				
hello	1	2	3	4
world	5	6	7	8
foo	9	10	11	12

In [7]:

```
dft = pd.read_table('ex1.csv')
dft
```

Out[7]:

	a,b,c,d,message
0	1,2,3,4,hello
1	5,6,7,8,world
2	9,10,11,12,foo

In [8]:

```
dft = pd.read_table('ex1.csv', sep = ',') #sep == 구분자
dft
```

Out[8]:

	a	b	c	d	message
0	1	2	3	4	hello
1	5	6	7	8	world
2	9	10	11	12	foo

In [9]:

```
!type ex2.csv
```

```
1,2,3,4,hello
5,6,7,8,world
9,10,11,12,foo
```

In [10]:

```
df2 = pd.read_csv('ex2.csv')
```

In [11]:

```
df2
```

Out[11]:

	1	2	3	4	hello
0	5	6	7	8	world
1	9	10	11	12	foo

In [12]:

```
df2 = pd.read_csv('ex2.csv', header = None)
df2
```

Out[12]:

	0	1	2	3	4
0	1	2	3	4	hello
1	5	6	7	8	world
2	9	10	11	12	foo

In [13]:

```
df2 = pd.read_csv('ex2.csv', names = ['a', 'b', 'c', 'd', 'message'])
df2
```

Out[13]:

	a	b	c	d	message
0	1	2	3	4	hello
1	5	6	7	8	world
2	9	10	11	12	foo

In [14]:

```
name = ['a', 'b', 'c', 'd', 'message']

df2 = pd.read_csv('ex2.csv', names = name, index_col = 'message')
df2
```

Out[14]:

	a	b	c	d
message				
hello	1	2	3	4
world	5	6	7	8
foo	9	10	11	12

In [15]:

```
!type ex4.csv
```

```
# hey!
a,b,c,d,message
# just wanted to make things more difficult for you
# who reads CSV files with computers, anyway?
1,2,3,4,hello
5,6,7,8,world
9,10,11,12,foo
```

In [18]:

```
df4 = pd.read_csv('ex4.csv', skiprows = [0, 2, 3])
df4
```

Out[18]:

	a	b	c	d	message
0	1	2	3	4	hello
1	5	6	7	8	world
2	9	10	11	12	foo

In [19]:

```
!type ex5.csv
```

```
something,a,b,c,d,message
one,1,2,3,4,NA
two,5,6,,8,world
three,9,10,11,12,foo
```

In [20]:

```
df5 = pd.read_csv('ex5.csv')
df5
```

Out[20]:

	something	a	b	c	d	message
0	one	1	2	3.0	4	NaN
1	two	5	6	NaN	8	world
2	three	9	10	11.0	12	foo

In [23]:

```
pd.isnull(df5)
```

Out[23]:

	something	a	b	c	d	message
0	False	False	False	False	False	True
1	False	False	False	True	False	False
2	False	False	False	False	False	False

In [24]:

```
sentinels = {'something' : ['two'], 'message' : ['world', 'NA']}
pd.read_csv('ex5.csv', na_values = sentinels)
```

Out[24]:

	something	a	b	c	d	message
0	one	1	2	3.0	4	NaN
1	NaN	5	6	NaN	8	NaN
2	three	9	10	11.0	12	foo

Writing Data to Text format

In [25]:

```
df5
```

Out[25]:

	something	a	b	c	d	message
0	one	1	2	3.0	4	NaN
1	two	5	6	NaN	8	world
2	three	9	10	11.0	12	foo

In [26]:

```
!dir
```

```
 C 드라이브의 볼륨에는 이름이 없습니다.
 볼륨 일련 번호: 3253-81AA

 C:₩Users₩Eunchae₩Desktop₩빅데이터응용 디렉터리

2021-05-02  오후 08:17    <DIR>          .
2021-05-02  오후 08:17    <DIR>          ..
2021-05-02  오후 07:43    <DIR>          .ipynb_checkpoints
2021-04-13  오후 06:04            78,679 1주차 201815069조은채 - Jupyter Notebook.pd
f
2021-04-13  오후 06:04           443,063 2주차 201815069 조은채 - Jupyter Notebook.p
df
2021-04-13  오후 05:59           499,230 3주차 - Jupyter Notebook.pdf
2021-04-12  오후 07:15           964,666 4주차 201815069 조은채 - Jupyter Notebook.p
df
2021-04-12  오후 08:48           127,071 5주차 - Jupyter Notebook.pdf
2021-04-13  오후 07:06           139,119 5주차 실습문제 - Jupyter Notebook.pdf
2021-04-13  오후 07:05             7,732 5주차 실습문제.ipynb
2021-04-19  오후 08:10           607,967 6주차 201815069조은채 - Jupyter Notebook.pd
f
2021-04-19  오후 08:09         1,221,451 6주차 201815069조은채.ipynb
2021-04-24  오후 09:46           202,709 6주차 실습문제 - Jupyter Notebook.pdf
2021-04-24  오후 09:46           212,049 6주차 실습문제.ipynb
2021-04-24  오후 11:01           535,901 7주차 201815069 조은채 - Jupyter Notebook.p
df
2021-04-24  오후 10:59           969,347 7주차 201815069 조은채.ipynb
2021-05-02  오후 07:35            92,065 7주차 실습문제 - Jupyter Notebook.pdf
2021-05-02  오후 07:35           109,896 7주차 실습문제.ipynb
2021-05-02  오후 08:17            27,027 8주차 201815069.ipynb
2021-04-26  오후 05:10             3,730 accident.csv
2021-04-12  오후 06:32         2,084,696 census.csv
2021-04-26  오후 05:10                58 ex1.csv
2021-04-26  오후 05:10                42 ex2.csv
2021-04-26  오후 05:10               163 ex4.csv
2021-04-26  오후 05:10                78 ex5.csv
              22개 파일           8,326,739 바이트
               3개 디렉터리  425,353,093,120 바이트 남음
```

In [27]:

```
df5.to_csv('out.csv')
```

In [33]:

```
!dir
```

```
 C 드라이브의 볼륨에는 이름이 없습니다.
 볼륨 일련 번호: 3253-81AA

 C:₩Users₩Eunchae₩Desktop₩빅데이터응용 디렉터리

2021-05-02  오후 08:21    <DIR>          .
2021-05-02  오후 08:21    <DIR>          ..
2021-05-02  오후 07:43    <DIR>          .ipynb_checkpoints
2021-04-13  오후 06:04            78,679 1주차 201815069조은채 - Jupyter Notebook.pd
f
2021-04-13  오후 06:04           443,063 2주차 201815069 조은채 - Jupyter Notebook.p
df
2021-04-13  오후 05:59           499,230 3주차 - Jupyter Notebook.pdf
2021-04-12  오후 07:15           964,666 4주차 201815069 조은채 - Jupyter Notebook.p
df
2021-04-12  오후 08:48           127,071 5주차 - Jupyter Notebook.pdf
2021-04-13  오후 07:06           139,119 5주차 실습문제 - Jupyter Notebook.pdf
2021-04-13  오후 07:05             7,732 5주차 실습문제.ipynb
2021-04-19  오후 08:10           607,967 6주차 201815069조은채 - Jupyter Notebook.pd
f
2021-04-19  오후 08:09         1,221,451 6주차 201815069조은채.ipynb
2021-04-24  오후 09:46           202,709 6주차 실습문제 - Jupyter Notebook.pdf
2021-04-24  오후 09:46           212,049 6주차 실습문제.ipynb
2021-04-24  오후 11:01           535,901 7주차 201815069 조은채 - Jupyter Notebook.p
df
2021-04-24  오후 10:59           969,347 7주차 201815069 조은채.ipynb
2021-05-02  오후 07:35            92,065 7주차 실습문제 - Jupyter Notebook.pdf
2021-05-02  오후 07:35           109,896 7주차 실습문제.ipynb
2021-05-02  오후 08:21            38,448 8주차 201815069.ipynb
2021-04-26  오후 05:10             3,730 accident.csv
2021-04-12  오후 06:32         2,084,696 census.csv
2021-04-26  오후 05:10                58 ex1.csv
2021-04-26  오후 05:10                42 ex2.csv
2021-04-26  오후 05:10               163 ex4.csv
2021-04-26  오후 05:10                78 ex5.csv
2021-05-02  오후 08:21                92 out.csv
              23개 파일           8,338,252 바이트
               3개 디렉터리  425,352,675,328 바이트 남음
```

In [29]:

```
!type out.csv
```

```
,something,a,b,c,d,message
0,one,1,2,3.0,4,
1,two,5,6,,8,world
2,three,9,10,11.0,12,foo
```

In [30]:

```
df5.to_csv('out.csv', na_rep = 'NULL')
!type out.csv
```

```
,something,a,b,c,d,message
0,one,1,2,3.0,4,NULL
1,two,5,6,NULL,8,world
2,three,9,10,11.0,12,foo
```

In [31]:

```
df5.to_csv('out.csv', index = False, header = False)
!type out.csv
```

```
one,1,2,3.0,4,
two,5,6,,8,world
three,9,10,11.0,12,foo
```

In [32]:

```
df5.to_csv('out.csv', sep = '&')
!type out.csv
```

```
&something&a&b&c&d&message
0&one&1&2&3.0&4&
1&two&5&6&&8&world
2&three&9&10&11.0&12&foo
```

Data Acquisition from JSON Data

In [36]:

```
obj = """
{"name": "Wes",
 "places_lived": ["United States", "Spain", "Germany"],
 "pet": null,
 "siblings": [{"name": "Scott", "age": 30, "pets": ["Zeus", "Zuko"]},
              {"name": "Katie", "age": 38, "pets": ["Sixes", "Stache", "Cisco"]}]
}
"""
obj
```

Out[36]:

```
'₩n{"name": "Wes",₩n "places_lived": ["United States", "Spain", "Germany"],₩n "pet":
null,₩n "siblings": [{"name": "Scott", "age": 30, "pets": ["Zeus", "Zuko"]},₩n
{"name": "Katie", "age": 38, "pets": ["Sixes", "Stache", "Cisco"]}]₩n}₩n'
```

In [34]:

```
import json
```

In [37]:

```
json.loads(obj)
```

Out[37]:

```
{'name': 'Wes',
 'places_lived': ['United States', 'Spain', 'Germany'],
 'pet': None,
 'siblings': [{'name': 'Scott', 'age': 30, 'pets': ['Zeus', 'Zuko']},
  {'name': 'Katie', 'age': 38, 'pets': ['Sixes', 'Stache', 'Cisco']}]}
```

In [39]:

```
result = json.loads(obj)
type(result)
```

Out[39]:

```
dict
```

In [40]:

```
df6 = pd.DataFrame(result['places_lived'], columns = ['place'])
df6
```

Out[40]:

	place
0	United States
1	Spain
2	Germany

In [41]:

```
df7 = pd.DataFrame(result['siblings'], columns = ['name', 'age', 'pets'])
df7
```

Out[41]:

	name	age	pets
0	Scott	30	[Zeus, Zuko]
1	Katie	38	[Sixes, Stache, Cisco]

Getting Data using API

In [55]:

```
ne0ne?key=573&ServiceKey=LYh8NWE962mcBlarGf%2Frapx7pe0VWy9Zsw50%2BC2n7nbCTeeHM23YlE0HRrAVtjnsSXTkyUc8
```

In [56]:

```
json_str
```

Out[56]:

'{WnWt"header" : {WnWtWt"description" : "소상공인시장진흥공단 주요상권"WnWtWt,"colum
ns" : ["상권번호","상권명","시도코드","시도명","시군구코드","시군구명","상권면
적","좌표개수","좌표값","데이터기준일자"]WnWtWt,"resultCode" : "00"WnWtWt,"resultMs
g" : "NORMAL SERVICE"WnWt},WnWt"body" : {WnWtWtWt"items" : [WnWtWtWtWt{WnWtWtWtWt"tr
arNo" : 573WnWtWtWtWt,"mainTrarNm" : "부산 금정구 구서동역"WnWtWtWtWt,"ctprvnCd" :
"26"WnWtWtWtWt,"ctprvnNm" : "부산광역시"WnWtWtWtWt,"signguCd" : "26410"WnWtWtWtWt,"s
ignguNm" : "금정구"WnWtWtWtWt,"trarArea" : 66080.5WnWtWtWtWt,"coordNum" : 16WnWtWtWt
Wt,"coords" : "POLYGON ((129.09092 35.248544, 129.090527 35.248755, 129.089265 35.24
8054, 129.089212 35.247334, 129.088914 35.247291, 129.088755 35.24522, 129.089828 3
5.244913, 129.08994 35.246723, 129.091246 35.246667, 129.091377 35.246602, 129.09146
6 35.247503, 129.091906 35.247535, 129.092248 35.247604, 129.091985 35.248508, 129.0
91776 35.248483, 129.09092 35.248544))"WnWtWtWtWt,"stdrDt" : "2015-12-17"WnWtWtWtWt}
WnWtWtWt]WnWt}Wn}Wn'

In [57]:

```
json_object = json.loads(json_str)
json_object
```

Out[57]:


```
{'header': {'description': '소상공인시장진흥공단 주요상권',
  'columns': ['상권번호',
   '상권명',
   '시도코드',
   '시도명',
   '시군구코드',
   '시군구명',
   '상권면적',
   '좌표개수',
   '좌표값',
   '데이터기준일자'],
  'resultCode': '00',
  'resultMsg': 'NORMAL SERVICE'},
 'body': {'items': [{'trarNo': 573,
    'mainTrarNm': '부산 금정구 구서동역',
    'ctprvnCd': '26',
    'ctprvnNm': '부산광역시',
    'signguCd': '26410',
    'signguNm': '금정구',
    'trarArea': 66080.5,
    'coordNum': 16,
    'coords': 'POLYGON ((129.09092 35.248544, 129.090527 35.248755, 129.089265 35.24
8054, 129.089212 35.247334, 129.088914 35.247291, 129.088755 35.24522, 129.089828 3
5.244913, 129.08994 35.246723, 129.091246 35.246667, 129.091377 35.246602, 129.09146
6 35.247503, 129.091906 35.247535, 129.092248 35.247604, 129.091985 35.248508, 129.0
91776 35.248483, 129.09092 35.248544))',
    'stdrDt': '2015-12-17'}]}}
```


In [58]:

```
body = [json_object['body']['items']]
body
```

Out[58]:


```
[[{'trarNo': 573,
   'mainTrarNm': '부산 금정구 구서동역',
   'ctprvnCd': '26',
   'ctprvnNm': '부산광역시',
   'signguCd': '26410',
   'signguNm': '금정구',
   'trarArea': 66080.5,
   'coordNum': 16,
   'coords': 'POLYGON ((129.09092 35.248544, 129.090527 35.248755, 129.089265 35.248
054, 129.089212 35.247334, 129.088914 35.247291, 129.088755 35.24522, 129.089828 35.
244913, 129.08994 35.246723, 129.091246 35.246667, 129.091377 35.246602, 129.091466
35.247503, 129.091906 35.247535, 129.092248 35.247604, 129.091985 35.248508, 129.091
776 35.248483, 129.09092 35.248544))',
   'stdrDt': '2015-12-17'}]]
```

In [62]:

```
from pandas.io.json import json_normalize
json_normalize(json_object['body']['items'])
```

```
<ipython-input-62-1b94e20cec6a>:2: FutureWarning: pandas.io.json.json_normalize is d
eprecated, use pandas.json_normalize instead
  json_normalize(json_object['body']['items'])
```

Out[62]:

	trarNo	mainTrarNm	ctprvnCd	ctprvnNm	signguCd	signguNm	trarArea	coordNum	cc
0	573	부산 금정구 구서동역	26	부산광역시	26410	금정구	66080.5	16	POLY ((129.0 35.24 129.09 3

In [64]:

```
pip install pandas_datareader
```

```
Collecting pandas_datareader
  Downloading pandas_datareader-0.9.0-py3-none-any.whl (107 kB)
Requirement already satisfied: pandas>=0.23 in c:₩users₩eunchae₩anaconda3₩lib₩site-p
ackages (from pandas_datareader) (1.1.3)
Requirement already satisfied: requests>=2.19.0 in c:₩users₩eunchae₩anaconda3₩lib₩si
te-packages (from pandas_datareader) (2.24.0)
Requirement already satisfied: lxml in c:₩users₩eunchae₩anaconda3₩lib₩site-packages
(from pandas_datareader) (4.6.1)
Requirement already satisfied: pytz>=2017.2 in c:₩users₩eunchae₩anaconda3₩lib₩site-p
ackages (from pandas>=0.23->pandas_datareader) (2020.1)
Requirement already satisfied: python-dateutil>=2.7.3 in c:₩users₩eunchae₩anaconda3
₩lib₩site-packages (from pandas>=0.23->pandas_datareader) (2.8.1)
Requirement already satisfied: numpy>=1.15.4 in c:₩users₩eunchae₩anaconda3₩lib₩site-
packages (from pandas>=0.23->pandas_datareader) (1.19.2)
Requirement already satisfied: idna<3,>=2.5 in c:₩users₩eunchae₩anaconda3₩lib₩site-p
ackages (from requests>=2.19.0->pandas_datareader) (2.10)
Requirement already satisfied: chardet<4,>=3.0.2 in c:₩users₩eunchae₩anaconda3₩lib₩s
ite-packages (from requests>=2.19.0->pandas_datareader) (3.0.4)
Requirement already satisfied: urllib3!=1.25.0,!=1.25.1,<1.26,>=1.21.1 in c:₩users₩e
unchae₩anaconda3₩lib₩site-packages (from requests>=2.19.0->pandas_datareader) (1.25.
11)
Requirement already satisfied: certifi>=2017.4.17 in c:₩users₩eunchae₩anaconda3₩lib
₩site-packages (from requests>=2.19.0->pandas_datareader) (2020.6.20)
Requirement already satisfied: six>=1.5 in c:₩users₩eunchae₩anaconda3₩lib₩site-packa
ges (from python-dateutil>=2.7.3->pandas>=0.23->pandas_datareader) (1.15.0)
Installing collected packages: pandas-datareader
Successfully installed pandas-datareader-0.9.0
Note: you may need to restart the kernel to use updated packages.
```

In [65]:

```
import pandas_datareader as pdr
```

In [66]:

```
df = pdr.get_data_yahoo('005930.KS')
df
```

Out[66]:

Date	High	Low	Open	Close	Volume	Adj Close
2016-05-03	25400.0	25120.0	25340.0	25220.0	7903300.0	21832.212891
2016-05-04	25800.0	25240.0	25440.0	25800.0	14702750.0	22334.296875
2016-05-09	26000.0	25700.0	25800.0	25980.0	13718100.0	22490.121094
2016-05-10	26000.0	25760.0	25980.0	25920.0	8559550.0	22438.179688
2016-05-11	25980.0	25740.0	25920.0	25840.0	8834400.0	22368.929688
...	...	...	...	...	...	...
2021-04-26	83500.0	82600.0	82900.0	83500.0	15489938.0	83500.000000
2021-04-27	83300.0	82500.0	83200.0	82900.0	12941533.0	82900.000000
2021-04-28	83200.0	82100.0	83200.0	82100.0	15596759.0	82100.000000
2021-04-29	82500.0	81500.0	82400.0	81700.0	20000973.0	81700.000000
2021-04-30	82100.0	81500.0	81900.0	81500.0	18673197.0	81500.000000

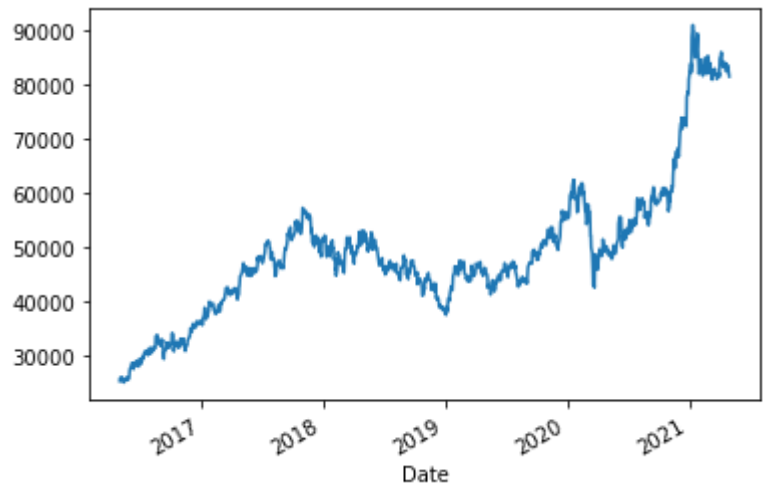
1221 rows × 6 columns

In [67]:

```
df['Close'].plot()
```

Out[67]:

<AxesSubplot:xlabel='Date'>



In [71]:

```
import matplotlib.pyplot as plt

df2 = pdr.get_data_yahoo('035420.KS')
plt.figure()
df2['Close'].plot()
df['Close'].plot()
```

Out[71]:

<AxesSubplot:xlabel='Date'>