# pandas

March 16, 2022

## 1 pandas tutorial day-11

this notebook explain..

## 2 How to install libraries

## 3 How to import libraries

```
[ ]: import pandas as pd
     import numpy as np
```

```
[ ]: # Object creation
     S = pd.Series([1,3,5, np.nan,7,8,9])
     S
```

```
[ ]: 0    1.0
     1    3.0
     2    5.0
     3    NaN
     4    7.0
     5    8.0
     6    9.0
     dtype: float64
```

```
[ ]: dates = pd.date_range("20220316", periods=8)
     dates
```

```
[ ]: DatetimeIndex(['2022-03-16', '2022-03-17', '2022-03-18', '2022-03-19',
                    '2022-03-20', '2022-03-21', '2022-03-22', '2022-03-23'],
                   dtype='datetime64[ns]', freq='D')
```

```
[ ]: dates = pd.date_range("20220316", periods=8)
     dates
     df = pd.DataFrame(np.random.randn(8,4), index=dates, columns=list("ABCD"))
     df
```

```
[ ]:                    A         B         C         D
     2022-03-16 -0.914245  0.959356  1.049520 -0.711187
     2022-03-17  0.882764  1.734137  0.584789 -0.461343
     2022-03-18 -1.659196  0.612527 -0.124615 -0.831812
     2022-03-19  1.184533 -0.531528  0.228300  0.409700
     2022-03-20  0.098057 -0.629789 -1.411908 -1.168739
     2022-03-21 -0.930665 -1.650199 -1.428300 -1.127338
     2022-03-22 -0.475620 -0.401701 -0.812811 -0.784462
     2022-03-23 -0.185733  2.125463  0.104172  0.552904
```

```
[ ]: df2 = pd.DataFrame(
         {
             "A": 1.0,
             "B": pd.Timestamp("20220316"),
             "C": pd.Series(1, index=list(range(4)), dtype="float32"),
             "D": np.array([3] * 4, dtype="int32"),
             "E": pd.Categorical(["girl", "woman", "girl", "woman"]),
             "F": "females"
         })
     df2
```

```
[ ]:      A          B    C  D      E        F
     0  1.0 2022-03-16  1.0  3   girl  females
     1  1.0 2022-03-16  1.0  3  woman  females
     2  1.0 2022-03-16  1.0  3   girl  females
     3  1.0 2022-03-16  1.0  3  woman  females
```

```
[ ]: df2.dtypes
```

```
[ ]: A           float64
     B    datetime64[ns]
     C           float32
     D             int32
     E          category
     F            object
     dtype: object
```

```
[ ]: df.head(2)
```

```
[ ]:                    A         B         C         D
     2022-03-16 -0.914245  0.959356  1.049520 -0.711187
     2022-03-17  0.882764  1.734137  0.584789 -0.461343
```

```
[ ]: df.tail(2)
```

```
[ ]:                    A         B         C         D
     2022-03-22 -0.475620 -0.401701 -0.812811 -0.784462
     2022-03-23 -0.185733  2.125463  0.104172  0.552904
```

```
[ ]: df.index
```

```
[ ]: DatetimeIndex(['2022-03-16', '2022-03-17', '2022-03-18', '2022-03-19',
                    '2022-03-20', '2022-03-21', '2022-03-22', '2022-03-23'],
                   dtype='datetime64[ns]', freq='D')
```

```
[ ]: df2.index
```

```
[ ]: Int64Index([0, 1, 2, 3], dtype='int64')
```

```
[ ]: df.to_numpy()
```

```
[ ]: array([[-0.91424523,  0.95935575,  1.04951987, -0.71118743],
            [ 0.88276369,  1.73413701,  0.58478911, -0.46134259],
            [-1.65919617,  0.61252719, -0.12461514, -0.83181165],
            [ 1.18453343, -0.53152797,  0.2283003 ,  0.40970008],
            [ 0.09805743, -0.6297892 , -1.41190787, -1.16873893],
            [-0.93066465, -1.65019875, -1.42829967, -1.12733786],
            [-0.47562024, -0.40170085, -0.81281118, -0.78446167],
            [-0.18573333,  2.12546308,  0.10417246,  0.55290433]])
```

```
[ ]: df2.to_numpy()
```

```
[ ]: array([[1.0, Timestamp('2022-03-16 00:00:00'), 1.0, 3, 'girl', 'females'],
            [1.0, Timestamp('2022-03-16 00:00:00'), 1.0, 3, 'woman',
             'females'],
            [1.0, Timestamp('2022-03-16 00:00:00'), 1.0, 3, 'girl', 'females'],
            [1.0, Timestamp('2022-03-16 00:00:00'), 1.0, 3, 'woman',
             'females']], dtype=object)
```

```
[ ]: df.describe()
```

```
[ ]:               A         B         C         D
     count  8.000000  8.000000  8.000000  8.000000
     mean  -0.250013  0.277283 -0.226357 -0.515284
     std    0.956749  1.296642  0.910750  0.655980
     min   -1.659196 -1.650199 -1.428300 -1.168739
     25%   -0.918350 -0.556093 -0.962585 -0.905693
     50%   -0.330677  0.105413 -0.010221 -0.747825
     75%    0.294234  1.153051  0.317423 -0.243582
     max    1.184533  2.125463  1.049520  0.552904
```

```
[ ]: # data transpose
     df2.T
```

```
[ ]:                      0                    1                    2  \
     A                  1.0                  1.0                  1.0
     B  2022-03-16 00:00:00  2022-03-16 00:00:00  2022-03-16 00:00:00
```

```
C                  1.0                  1.0                  1.0
D                    3                    3                    3
E                 girl                woman                 girl
F              females              females              females

                     3
A                  1.0
B  2022-03-16 00:00:00
C                  1.0
D                    3
E                woman
F              females
```

```
[ ]: df.sort_index(axis=1, ascending=False)
```

```
[ ]:                    D         C         B         A
     2022-03-16 -0.711187  1.049520  0.959356 -0.914245
     2022-03-17 -0.461343  0.584789  1.734137  0.882764
     2022-03-18 -0.831812 -0.124615  0.612527 -1.659196
     2022-03-19  0.409700  0.228300 -0.531528  1.184533
     2022-03-20 -1.168739 -1.411908 -0.629789  0.098057
     2022-03-21 -1.127338 -1.428300 -1.650199 -0.930665
     2022-03-22 -0.784462 -0.812811 -0.401701 -0.475620
     2022-03-23  0.552904  0.104172  2.125463 -0.185733
```

```
[ ]: df.sort_index(axis=1, ascending=True)
```

```
[ ]:                    A         B         C         D
     2022-03-16 -0.914245  0.959356  1.049520 -0.711187
     2022-03-17  0.882764  1.734137  0.584789 -0.461343
     2022-03-18 -1.659196  0.612527 -0.124615 -0.831812
     2022-03-19  1.184533 -0.531528  0.228300  0.409700
     2022-03-20  0.098057 -0.629789 -1.411908 -1.168739
     2022-03-21 -0.930665 -1.650199 -1.428300 -1.127338
     2022-03-22 -0.475620 -0.401701 -0.812811 -0.784462
     2022-03-23 -0.185733  2.125463  0.104172  0.552904
```

```
[ ]: df.sort_values(by="B")
```

```
[ ]:                    A         B         C         D
     2022-03-21 -0.930665 -1.650199 -1.428300 -1.127338
     2022-03-20  0.098057 -0.629789 -1.411908 -1.168739
     2022-03-19  1.184533 -0.531528  0.228300  0.409700
     2022-03-22 -0.475620 -0.401701 -0.812811 -0.784462
     2022-03-18 -1.659196  0.612527 -0.124615 -0.831812
     2022-03-16 -0.914245  0.959356  1.049520 -0.711187
     2022-03-17  0.882764  1.734137  0.584789 -0.461343
```

```
2022-03-23 -0.185733  2.125463  0.104172  0.552904
```

[ ]:
```
df.sort_values(by="B", ascending=True)
```

[ ]:
```
                   A         B         C         D
2022-03-21 -0.930665 -1.650199 -1.428300 -1.127338
2022-03-20  0.098057 -0.629789 -1.411908 -1.168739
2022-03-19  1.184533 -0.531528  0.228300  0.409700
2022-03-22 -0.475620 -0.401701 -0.812811 -0.784462
2022-03-18 -1.659196  0.612527 -0.124615 -0.831812
2022-03-16 -0.914245  0.959356  1.049520 -0.711187
2022-03-17  0.882764  1.734137  0.584789 -0.461343
2022-03-23 -0.185733  2.125463  0.104172  0.552904
```

[ ]:
```
df.sort_values(by="B", ascending=False)
```

[ ]:
```
                   A         B         C         D
2022-03-23 -0.185733  2.125463  0.104172  0.552904
2022-03-17  0.882764  1.734137  0.584789 -0.461343
2022-03-16 -0.914245  0.959356  1.049520 -0.711187
2022-03-18 -1.659196  0.612527 -0.124615 -0.831812
2022-03-22 -0.475620 -0.401701 -0.812811 -0.784462
2022-03-19  1.184533 -0.531528  0.228300  0.409700
2022-03-20  0.098057 -0.629789 -1.411908 -1.168739
2022-03-21 -0.930665 -1.650199 -1.428300 -1.127338
```

[ ]:
```
# to select ay specific column
df["A"]
```

[ ]:
```
2022-03-16   -0.914245
2022-03-17    0.882764
2022-03-18   -1.659196
2022-03-19    1.184533
2022-03-20    0.098057
2022-03-21   -0.930665
2022-03-22   -0.475620
2022-03-23   -0.185733
Freq: D, Name: A, dtype: float64
```

[ ]:
```
df["B"]
```

[ ]:
```
2022-03-16    0.959356
2022-03-17    1.734137
2022-03-18    0.612527
2022-03-19   -0.531528
2022-03-20   -0.629789
2022-03-21   -1.650199
2022-03-22   -0.401701
```

```
2022-03-23    2.125463
Freq: D, Name: B, dtype: float64
```

[ ]:

```
# row wise selection
df[0:2]
```

[ ]:

```
                   A         B         C         D
2022-03-16 -0.914245  0.959356  1.049520 -0.711187
2022-03-17  0.882764  1.734137  0.584789 -0.461343
```

[ ]:

```
df[0:3]
```

[ ]:

```
                   A         B         C         D
2022-03-16 -0.914245  0.959356  1.049520 -0.711187
2022-03-17  0.882764  1.734137  0.584789 -0.461343
2022-03-18 -1.659196  0.612527 -0.124615 -0.831812
```

[ ]:

```
df[0:5]
```

[ ]:

```
                   A         B         C         D
2022-03-16 -0.914245  0.959356  1.049520 -0.711187
2022-03-17  0.882764  1.734137  0.584789 -0.461343
2022-03-18 -1.659196  0.612527 -0.124615 -0.831812
2022-03-19  1.184533 -0.531528  0.228300  0.409700
2022-03-20  0.098057 -0.629789 -1.411908 -1.168739
```

[ ]:

```
# select by labels
df.loc[dates[0]]
```

[ ]:

```
A   -0.914245
B    0.959356
C    1.049520
D   -0.711187
Name: 2022-03-16 00:00:00, dtype: float64
```

[ ]:

```
# row wise selection
df.loc[dates[7]]
```

[ ]:

```
A   -0.185733
B    2.125463
C    0.104172
D    0.552904
Name: 2022-03-23 00:00:00, dtype: float64
```

[ ]:

```
# column wise selection
# "A" and "B" means columns
df.loc[:, ["A", "B"]]
```

```
[ ]:                    A         B
     2022-03-16 -0.914245  0.959356
     2022-03-17  0.882764  1.734137
     2022-03-18 -1.659196  0.612527
     2022-03-19  1.184533 -0.531528
     2022-03-20  0.098057 -0.629789
     2022-03-21 -0.930665 -1.650199
     2022-03-22 -0.475620 -0.401701
     2022-03-23 -0.185733  2.125463
```

```
[ ]: # select specific data
     df.loc["20220320": "20220323", ["A", "B"]]
```

```
[ ]:                    A         B
     2022-03-20  0.098057 -0.629789
     2022-03-21 -0.930665 -1.650199
     2022-03-22 -0.475620 -0.401701
     2022-03-23 -0.185733  2.125463
```

```
[ ]: # different dimension
     df.loc["20220317", ["A", "B"]]
```

```
[ ]: A    0.882764
     B    1.734137
     Name: 2022-03-17 00:00:00, dtype: float64
```

```
[ ]: # value of "A" on specific dates
     # Scalar value
     df.at[dates[0], "A"]
```

```
[ ]: -0.9142452301694447
```

```
[ ]: df.at[dates[5], "A"]
```

```
[ ]: -0.9306646546011359
```

```
[ ]: # position base
     df.iloc[3]
```

```
[ ]: A    1.184533
     B   -0.531528
     C    0.228300
     D    0.409700
     Name: 2022-03-19 00:00:00, dtype: float64
```

```
[ ]: df.iloc[0:5]
[ ]:                    A         B         C         D
     2022-03-16 -0.914245  0.959356  1.049520 -0.711187
     2022-03-17  0.882764  1.734137  0.584789 -0.461343
     2022-03-18 -1.659196  0.612527 -0.124615 -0.831812
     2022-03-19  1.184533 -0.531528  0.228300  0.409700
     2022-03-20  0.098057 -0.629789 -1.411908 -1.168739
```

```
[ ]: df.iloc[0:5, 0:4]
```

```
[ ]:                    A         B         C         D
     2022-03-16 -0.914245  0.959356  1.049520 -0.711187
     2022-03-17  0.882764  1.734137  0.584789 -0.461343
     2022-03-18 -1.659196  0.612527 -0.124615 -0.831812
     2022-03-19  1.184533 -0.531528  0.228300  0.409700
     2022-03-20  0.098057 -0.629789 -1.411908 -1.168739
```

```
[ ]: # if need all column then use this command
     df.iloc[0:8, :]
```

```
[ ]:                    A         B         C         D
     2022-03-16 -0.914245  0.959356  1.049520 -0.711187
     2022-03-17  0.882764  1.734137  0.584789 -0.461343
     2022-03-18 -1.659196  0.612527 -0.124615 -0.831812
     2022-03-19  1.184533 -0.531528  0.228300  0.409700
     2022-03-20  0.098057 -0.629789 -1.411908 -1.168739
     2022-03-21 -0.930665 -1.650199 -1.428300 -1.127338
     2022-03-22 -0.475620 -0.401701 -0.812811 -0.784462
     2022-03-23 -0.185733  2.125463  0.104172  0.552904
```

```
[ ]: # column with string method
     df.iloc[:, 0:1]
```

```
[ ]:                    A
     2022-03-16 -0.914245
     2022-03-17  0.882764
     2022-03-18 -1.659196
     2022-03-19  1.184533
     2022-03-20  0.098057
     2022-03-21 -0.930665
     2022-03-22 -0.475620
     2022-03-23 -0.185733
```

```
[ ]: df.iloc[:, 0:2]
```

```
[ ]:                    A         B
     2022-03-16 -0.914245  0.959356
     2022-03-17  0.882764  1.734137
     2022-03-18 -1.659196  0.612527
```

```
2022-03-19  1.184533 -0.531528
2022-03-20  0.098057 -0.629789
2022-03-21 -0.930665 -1.650199
2022-03-22 -0.475620 -0.401701
2022-03-23 -0.185733  2.125463
```

```
[ ]: # Booleans true or false
     # print values greater than 0
     df[df["A"]>0]
```

```
[ ]:                    A         B         C         D
     2022-03-17  0.882764  1.734137  0.584789 -0.461343
     2022-03-19  1.184533 -0.531528  0.228300  0.409700
     2022-03-20  0.098057 -0.629789 -1.411908 -1.168739
```

```
[ ]: # specific value filtration
     df[df["A"]>0]
```

```
[ ]:                    A         B         C         D
     2022-03-17  0.882764  1.734137  0.584789 -0.461343
     2022-03-19  1.184533 -0.531528  0.228300  0.409700
     2022-03-20  0.098057 -0.629789 -1.411908 -1.168739
```

```
[ ]: # whole value filtration
     df[df>0]
```

```
[ ]:                    A         B         C         D
     2022-03-16       NaN  0.959356  1.049520       NaN
     2022-03-17  0.882764  1.734137  0.584789       NaN
     2022-03-18       NaN  0.612527       NaN       NaN
     2022-03-19  1.184533       NaN  0.228300  0.409700
     2022-03-20  0.098057       NaN       NaN       NaN
     2022-03-21       NaN       NaN       NaN       NaN
     2022-03-22       NaN       NaN       NaN       NaN
     2022-03-23       NaN  2.125463  0.104172  0.552904
```

```
[ ]: # IS IN method
     # to add column to the existing data
     df2=df.copy()
     df2["babag ka column"]= ["one", "two", "three", "four", "three","one", "two",␣
      ↪"three"]
     df2
```

```
[ ]:                    A         B         C         D babag ka column
     2022-03-16 -0.914245  0.959356  1.049520 -0.711187             one
     2022-03-17  0.882764  1.734137  0.584789 -0.461343             two
     2022-03-18 -1.659196  0.612527 -0.124615 -0.831812           three
     2022-03-19  1.184533 -0.531528  0.228300  0.409700            four
```

```
2022-03-20  0.098057 -0.629789 -1.411908 -1.168739          three
2022-03-21 -0.930665 -1.650199 -1.428300 -1.127338            one
2022-03-22 -0.475620 -0.401701 -0.812811 -0.784462            two
2022-03-23 -0.185733  2.125463  0.104172  0.552904          three
```

[ ]:

```
# to add column to the existing data
df2["babag ka column"]= ["one", "two", "three", "four", "three","one", "two",␣
 ↪"three"]
df2
```

[ ]:

```
                   A         B         C         D babag ka column
2022-03-16 -0.914245  0.959356  1.049520 -0.711187             one
2022-03-17  0.882764  1.734137  0.584789 -0.461343             two
2022-03-18 -1.659196  0.612527 -0.124615 -0.831812           three
2022-03-19  1.184533 -0.531528  0.228300  0.409700            four
2022-03-20  0.098057 -0.629789 -1.411908 -1.168739           three
2022-03-21 -0.930665 -1.650199 -1.428300 -1.127338             one
2022-03-22 -0.475620 -0.401701 -0.812811 -0.784462             two
2022-03-23 -0.185733  2.125463  0.104172  0.552904           three
```

[ ]:

```
df2["new"]=[1.2, 2.3, 3.4,4.5,5.6,6.7,7.8,8.9]
df2
df2=df2.iloc[:, 0:1]
df2
```

[ ]:

```
                   A
2022-03-16 -0.914245
2022-03-17  0.882764
2022-03-18 -1.659196
2022-03-19  1.184533
2022-03-20  0.098057
2022-03-21 -0.930665
2022-03-22 -0.475620
2022-03-23 -0.185733
```

[ ]:

```
df2=df2.iloc[:, 0:3]
df2
```

[ ]:

```
                   A
2022-03-16 -0.914245
2022-03-17  0.882764
2022-03-18 -1.659196
2022-03-19  1.184533
2022-03-20  0.098057
2022-03-21 -0.930665
2022-03-22 -0.475620
2022-03-23 -0.185733
```

[ ]:

```
```