

PANDAS TUTORIAL DAY-11

- Jupyter Notebook
- Installing Libraries

```
In [ ]: # pip install pandas
        # pip install numpy
```

- Importing Libraries

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

- Object Creation

```
In [ ]: s= pd.Series([1,2,4,9,7,np.nan,3])
        s
```

```
Out[ ]: 0    1.0
        1    2.0
        2    4.0
        3    9.0
        4    7.0
        5    NaN
        6    3.0
        dtype: float64
```

- Creating Dates

```
In [ ]: # Format DDMMYY
        dates=pd.date_range("20220110",periods=4)
        dates
```

```
Out[ ]: DatetimeIndex(['2022-01-10', '2022-01-11', '2022-01-12', '2022-01-13'], dtype='datetime64[ns]', freq='D')
```

```
In [ ]: dates=pd.date_range("20220110",periods=4)
        dates
        df= pd.DataFrame(np.random.randn(4,5), index=dates,columns=list("ABCDE"))
        df
```

```
Out[ ]:
```

	A	B	C	D	E
2022-01-10	-0.487246	2.265057	1.165897	-1.152563	0.595654
2022-01-11	0.962376	0.992439	0.356797	-0.590903	0.990162
2022-01-12	-0.712390	-0.833559	0.199270	1.202913	-0.030350

	A	B	C	D	E
2022-01-13	0.904374	-1.026988	0.931431	0.993088	-1.639472

In []:

```
df.head(2)
```

Out[]:

	A	B	C	D	E
2022-01-10	-0.487246	2.265057	1.165897	-1.152563	0.595654
2022-01-11	0.962376	0.992439	0.356797	-0.590903	0.990162

In []:

```
df.tail(2)
```

Out[]:

	A	B	C	D	E
2022-01-12	-0.712390	-0.833559	0.199270	1.202913	-0.030350
2022-01-13	0.904374	-1.026988	0.931431	0.993088	-1.639472

In []:

```
df2= pd.DataFrame(
    {
        "A": 1.0,
        "B": pd.Timestamp("20220110"),
        "C": pd.Series(1,index=list(range(4)),dtype="float32"),
        "D": np.array([3]*4,dtype="int32"),
        "E": pd.Categorical(["test","train","test","train"]),
        "F": "foo"
    }
)
```

In []:

```
df2
```

Out[]:

	A	B	C	D	E	F
0	1.0	2022-01-10	1.0	3	test	foo
1	1.0	2022-01-10	1.0	3	train	foo
2	1.0	2022-01-10	1.0	3	test	foo
3	1.0	2022-01-10	1.0	3	train	foo

In []:

```
df3= pd.DataFrame(
    {
        "A": 1.0,
        "B": pd.Timestamp("20220110"),
        "C": pd.Series(2,index=list(range(5)),dtype="int32"),
        "D": np.array([20]*5,dtype="float32"),
        "E": pd.Categorical(["tana","bana","tana","bana","mana"]),
        "F": "too"
    }
)
```

```
)
df3
```

```
Out[ ]:
```

	A	B	C	D	E	F
0	1.0	2022-01-10	2	20.0	tana	too
1	1.0	2022-01-10	2	20.0	bana	too
2	1.0	2022-01-10	2	20.0	tana	too
3	1.0	2022-01-10	2	20.0	bana	too
4	1.0	2022-01-10	2	20.0	mana	too

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

```
Out[ ]:
```

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

dtype: object

```
In [ ]: df2.dtypes["A"]
```

```
Out[ ]: dtype('float64')
```

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

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

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

```
Out[ ]: array([[ -0.48724647,  2.26505745,  1.16589703, -1.15256321,  0.5956542 ],
        [ 0.96237638,  0.99243875,  0.3567971 , -0.59090292,  0.99016233],
        [-0.7123897 , -0.83355876,  0.19927026,  1.20291281, -0.0303501 ],
        [ 0.90437372, -1.0269883 ,  0.93143091,  0.99308793, -1.63947183]])
```

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

```
Out[ ]: array([[1.0, Timestamp('2022-01-10 00:00:00'), 1.0, 3, 'test', 'foo'],
        [1.0, Timestamp('2022-01-10 00:00:00'), 1.0, 3, 'train', 'foo'],
        [1.0, Timestamp('2022-01-10 00:00:00'), 1.0, 3, 'test', 'foo'],
        [1.0, Timestamp('2022-01-10 00:00:00'), 1.0, 3, 'train', 'foo']],
        dtype=object)
```

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

```
Out[ ]:
```

	A	B	C	D	E
count	4.000000	4.000000	4.000000	4.000000	4.000000

	A	B	C	D	E
mean	0.166778	0.349237	0.663349	0.113134	-0.021001
std	0.890264	1.568128	0.459625	1.163270	1.157908
min	-0.712390	-1.026988	0.199270	-1.152563	-1.639472
25%	-0.543532	-0.881916	0.317415	-0.731318	-0.432631
50%	0.208564	0.079440	0.644114	0.201093	0.282652
75%	0.918874	1.310593	0.990047	1.045544	0.694281
max	0.962376	2.265057	1.165897	1.202913	0.990162

In []: `df2.describe()`

Out[]:

	A	C	D
count	4.0	4.0	4.0
mean	1.0	1.0	3.0
std	0.0	0.0	0.0
min	1.0	1.0	3.0
25%	1.0	1.0	3.0
50%	1.0	1.0	3.0
75%	1.0	1.0	3.0
max	1.0	1.0	3.0

In []: `df2.T`

Out[]:

	0	1	2	3
A	1.0	1.0	1.0	1.0
B	2022-01-10 00:00:00	2022-01-10 00:00:00	2022-01-10 00:00:00	2022-01-10 00:00:00
C	1.0	1.0	1.0	1.0
D	3	3	3	3
E	test	train	test	train
F	foo	foo	foo	foo

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

Out[]:

	A	B	C	D	E
2022-01-10	-0.487246	2.265057	1.165897	-1.152563	0.595654
2022-01-11	0.962376	0.992439	0.356797	-0.590903	0.990162

	A	B	C	D	E
2022-01-12	-0.712390	-0.833559	0.199270	1.202913	-0.030350
2022-01-13	0.904374	-1.026988	0.931431	0.993088	-1.639472

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

Out[]:

	E	D	C	B	A
2022-01-10	0.595654	-1.152563	1.165897	2.265057	-0.487246
2022-01-11	0.990162	-0.590903	0.356797	0.992439	0.962376
2022-01-12	-0.030350	1.202913	0.199270	-0.833559	-0.712390
2022-01-13	-1.639472	0.993088	0.931431	-1.026988	0.904374

In []: `df`

Out[]:

	A	B	C	D	E
2022-01-10	-0.487246	2.265057	1.165897	-1.152563	0.595654
2022-01-11	0.962376	0.992439	0.356797	-0.590903	0.990162
2022-01-12	-0.712390	-0.833559	0.199270	1.202913	-0.030350
2022-01-13	0.904374	-1.026988	0.931431	0.993088	-1.639472

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

Out[]:

	A	B	C	D	E
2022-01-10	-0.487246	2.265057	1.165897	-1.152563	0.595654
2022-01-11	0.962376	0.992439	0.356797	-0.590903	0.990162
2022-01-12	-0.712390	-0.833559	0.199270	1.202913	-0.030350
2022-01-13	0.904374	-1.026988	0.931431	0.993088	-1.639472

In []: `df["C"]`

Out[]:

```

2022-01-10    1.165897
2022-01-11    0.356797
2022-01-12    0.199270
2022-01-13    0.931431
Freq: D, Name: C, dtype: float64

```

In []: `df[:]`

Out[]:

	A	B	C	D	E
--	---	---	---	---	---

	A	B	C	D	E
2022-01-10	-0.487246	2.265057	1.165897	-1.152563	0.595654
2022-01-11	0.962376	0.992439	0.356797	-0.590903	0.990162
2022-01-12	-0.712390	-0.833559	0.199270	1.202913	-0.030350
2022-01-13	0.904374	-1.026988	0.931431	0.993088	-1.639472

- Row wise Selection

In []: `df[1:2]`

Out[]:

	A	B	C	D	E
2022-01-11	0.962376	0.992439	0.356797	-0.590903	0.990162

In []: `df.loc[dates[1]]`

Out[]:

A	0.962376
B	0.992439
C	0.356797
D	-0.590903
E	0.990162

Name: 2022-01-11 00:00:00, dtype: float64

In []: `dates`

Out[]: DatetimeIndex(['2022-01-10', '2022-01-11', '2022-01-12', '2022-01-13'], dtype='datetime64[ns]', freq='D')

In []: `df`

Out[]:

	A	B	C	D	E
2022-01-10	-0.487246	2.265057	1.165897	-1.152563	0.595654
2022-01-11	0.962376	0.992439	0.356797	-0.590903	0.990162
2022-01-12	-0.712390	-0.833559	0.199270	1.202913	-0.030350
2022-01-13	0.904374	-1.026988	0.931431	0.993088	-1.639472

In []: `df.loc[dates[0]]`

Out[]:

A	-0.487246
B	2.265057
C	1.165897
D	-1.152563
E	0.595654

Name: 2022-01-10 00:00:00, dtype: float64

- Column Wise Selection

```
In [ ]: df.loc[:,["A","B"]]
```

```
Out[ ]:
```

	A	B
2022-01-10	-0.487246	2.265057
2022-01-11	0.962376	0.992439
2022-01-12	-0.712390	-0.833559
2022-01-13	0.904374	-1.026988

```
In [ ]: df.loc[:,["C"]]
```

```
Out[ ]:
```

	C
2022-01-10	1.165897
2022-01-11	0.356797
2022-01-12	0.199270
2022-01-13	0.931431

```
In [ ]: df.loc["20220110":"20220112",["A","B"]]
```

```
Out[ ]:
```

	A	B
2022-01-10	-0.487246	2.265057
2022-01-11	0.962376	0.992439
2022-01-12	-0.712390	-0.833559

```
In [ ]: df.loc[["20220110","20220112"],["A","B"]]
```

```
Out[ ]:
```

	A	B
2022-01-10	-0.487246	2.265057
2022-01-12	-0.712390	-0.833559

```
In [ ]: df.loc["20220110",["A","B"]]
```

```
Out[ ]:
```

A	-0.487246
B	2.265057

Name: 2022-01-10 00:00:00, dtype: float64

```
In [ ]: df.at[dates[1],"A"]
```

```
Out[ ]: 0.9623763835896663
```

```
In [ ]: df.at[dates[0], "A"]
```

```
Out[ ]: -0.4872464684539594
```

```
In [ ]: df.iloc[0:3, 0:2] # rows, col
```

```
Out[ ]:
```

	A	B
2022-01-10	-0.487246	2.265057
2022-01-11	0.962376	0.992439
2022-01-12	-0.712390	-0.833559

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

```
Out[ ]:
```

	A	B
2022-01-10	-0.487246	2.265057
2022-01-11	0.962376	0.992439
2022-01-12	-0.712390	-0.833559
2022-01-13	0.904374	-1.026988

```
In [ ]: df[df["A"] > 0]
```

```
Out[ ]:
```

	A	B	C	D	E
2022-01-11	0.962376	0.992439	0.356797	-0.590903	0.990162
2022-01-13	0.904374	-1.026988	0.931431	0.993088	-1.639472

```
In [ ]: df
```

```
Out[ ]:
```

	A	B	C	D	E
2022-01-10	-0.487246	2.265057	1.165897	-1.152563	0.595654
2022-01-11	0.962376	0.992439	0.356797	-0.590903	0.990162
2022-01-12	-0.712390	-0.833559	0.199270	1.202913	-0.030350
2022-01-13	0.904374	-1.026988	0.931431	0.993088	-1.639472

```
In [ ]: df[df["B"] > 1.5]
```

```
Out[ ]:
```

	A	B	C	D	E
2022-01-10	-0.487246	2.265057	1.165897	-1.152563	0.595654


```
In [ ]: i=df[df>0]
        i
```

```
Out[ ]:
```

	A	B	C	D	E
2022-01-10	NaN	2.265057	1.165897	NaN	0.595654
2022-01-11	0.962376	0.992439	0.356797	NaN	0.990162
2022-01-12	NaN	NaN	0.199270	1.202913	NaN
2022-01-13	0.904374	NaN	0.931431	0.993088	NaN

```
In [ ]: f=df.copy()
        f
```

```
Out[ ]:
```

	A	B	C	D	E
2022-01-10	-0.487246	2.265057	1.165897	-1.152563	0.595654
2022-01-11	0.962376	0.992439	0.356797	-0.590903	0.990162
2022-01-12	-0.712390	-0.833559	0.199270	1.202913	-0.030350
2022-01-13	0.904374	-1.026988	0.931431	0.993088	-1.639472

```
In [ ]: f["E"]=["one","two","three","four"]
```

```
In [ ]: f
```

```
Out[ ]:
```

	A	B	C	D	E
2022-01-10	-0.487246	2.265057	1.165897	-1.152563	one
2022-01-11	0.962376	0.992439	0.356797	-0.590903	two
2022-01-12	-0.712390	-0.833559	0.199270	1.202913	three
2022-01-13	0.904374	-1.026988	0.931431	0.993088	four

```
In [ ]: f=f.iloc[:,0:2]
        f
```

```
Out[ ]:
```

	A	B
2022-01-10	-0.487246	2.265057
2022-01-11	0.962376	0.992439
2022-01-12	-0.712390	-0.833559
2022-01-13	0.904374	-1.026988

- Assignment

```
In [ ]: df["mean"]=df.mean(axis='columns')
df
```

```
Out[ ]:
```

	A	B	C	D	E	mean
2022-01-10	-0.487246	2.265057	1.165897	-1.152563	0.595654	0.477360
2022-01-11	0.962376	0.992439	0.356797	-0.590903	0.990162	0.542174
2022-01-12	-0.712390	-0.833559	0.199270	1.202913	-0.030350	-0.034823
2022-01-13	0.904374	-1.026988	0.931431	0.993088	-1.639472	0.032486

```
In [ ]: df["min"]=df.min(axis='columns')
df
```

```
Out[ ]:
```

	A	B	C	D	E	mean	min
2022-01-10	-0.487246	2.265057	1.165897	-1.152563	0.595654	0.477360	-1.152563
2022-01-11	0.962376	0.992439	0.356797	-0.590903	0.990162	0.542174	-0.590903
2022-01-12	-0.712390	-0.833559	0.199270	1.202913	-0.030350	-0.034823	-0.833559
2022-01-13	0.904374	-1.026988	0.931431	0.993088	-1.639472	0.032486	-1.639472

```
In [ ]: df["max"]=df.max(axis='columns')
df
```

```
Out[ ]:
```

	A	B	C	D	E	mean	min	max
2022-01-10	-0.487246	2.265057	1.165897	-1.152563	0.595654	0.477360	-1.152563	2.265057
2022-01-11	0.962376	0.992439	0.356797	-0.590903	0.990162	0.542174	-0.590903	0.992439
2022-01-12	-0.712390	-0.833559	0.199270	1.202913	-0.030350	-0.034823	-0.833559	1.202913
2022-01-13	0.904374	-1.026988	0.931431	0.993088	-1.639472	0.032486	-1.639472	0.993088

```
In [ ]: df["median"]=df.median(axis='columns')
df
```

```
Out[ ]:
```

	A	B	C	D	E	mean	min	max	median
2022-01-10	-0.487246	2.265057	1.165897	-1.152563	0.595654	0.477360	-1.152563	2.265057	0.536507
2022-01-11	0.962376	0.992439	0.356797	-0.590903	0.990162	0.542174	-0.590903	0.992439	0.752275
2022-01-12	-0.712390	-0.833559	0.199270	1.202913	-0.030350	-0.034823	-0.833559	1.202913	-0.032587

	A	B	C	D	E	mean	min	max	median
2022-01-13	0.904374	-1.026988	0.931431	0.993088	-1.639472	0.032486	-1.639472	0.993088	0.468430

```
In [ ]: df.rename(
        columns={
            "mean": "means"
        },
        inplace=True
    )
```

```
In [ ]: df
```

```
Out[ ]:
```

	A	B	C	D	E	means	min	max	median
2022-01-10	-0.487246	2.265057	1.165897	-1.152563	0.595654	0.477360	-1.152563	2.265057	0.536507
2022-01-11	0.962376	0.992439	0.356797	-0.590903	0.990162	0.542174	-0.590903	0.992439	0.752275
2022-01-12	-0.712390	-0.833559	0.199270	1.202913	-0.030350	-0.034823	-0.833559	1.202913	-0.032587
2022-01-13	0.904374	-1.026988	0.931431	0.993088	-1.639472	0.032486	-1.639472	0.993088	0.468430

```
In [ ]: df.drop("median", axis=1, inplace=True)
df
```

```
Out[ ]:
```

	A	B	C	D	E	means	min	max
2022-01-10	-0.487246	2.265057	1.165897	-1.152563	0.595654	0.477360	-1.152563	2.265057
2022-01-11	0.962376	0.992439	0.356797	-0.590903	0.990162	0.542174	-0.590903	0.992439
2022-01-12	-0.712390	-0.833559	0.199270	1.202913	-0.030350	-0.034823	-0.833559	1.202913
2022-01-13	0.904374	-1.026988	0.931431	0.993088	-1.639472	0.032486	-1.639472	0.993088