Pandas

In [88]:
```
#importing library
import pandas as pd
```

In [2]:
```
#checking pandas version
pd.__version__
```

Out[2]:
```
'1.4.2'
```

In [5]:
```
#series
l=[1,2,3,4,5]
s1=pd.Series(l)
s1
```

Out[5]:
```
0    1
1    2
2    3
3    4
4    5
dtype: int64
```

In [17]:
```
import numpy as np
n=np.random.randn(5)
order="a","b","c","d","e"
s2=pd.Series(n,index=order)
```

In [18]:
```
s2
```

Out[18]:
```
a    0.585978
b   -0.005449
c   -1.228617
d   -0.133466
e   -0.681429
dtype: float64
```

In [24]:
```
#modifying the index of series
s1
s1.index=["A","B","C","D","E"]
s1
```

Out[24]:
```
A    1
B    2
C    3
D    4
E    5
dtype: int64
```

In [29]:
```
#slicing
a=s1[:3]
a
```

Out[29]:
```
A    1
B    2
C    3
dtype: int64
```

In [33]:
```
s3=s1.append(s2)
```

```
C:\Users\user\AppData\Local\Temp\ipykernel_10864\4069742347.py:1: FutureWarnin
g: The series.append method is deprecated and will be removed from pandas in a
future version. Use pandas.concat instead.
  s3=s1.append(s2)
```

In [34]:
```
s3
```

Out[34]:
```
A    1.000000
B    2.000000
C    3.000000
D    4.000000
E    5.000000
a    0.585978
b   -0.005449
c   -1.228617
d   -0.133466
e   -0.681429
dtype: float64
```

In [35]:
```
s3.drop("e")
```

Out[35]:
```
A    1.000000
B    2.000000
C    3.000000
D    4.000000
E    5.000000
a    0.585978
b   -0.005449
c   -1.228617
d   -0.133466
dtype: float64
```

In [36]:
```
1 s3
```

Out[36]:
```
A    1.000000
B    2.000000
C    3.000000
D    4.000000
E    5.000000
a    0.585978
b   -0.005449
c   -1.228617
d   -0.133466
e   -0.681429
dtype: float64
```

In [ ]:
```
1
```

In [ ]:
```
1
```

In [41]:
```
1 #operations
2 s4=[1,2,3,4,5,6,7]
3 s5=[8,9,10]
4 s4=pd.Series(s4)
5 s5=pd.Series(s5)
```

In [42]:
```
1 s4
```

Out[42]:
```
0    1
1    2
2    3
3    4
4    5
5    6
6    7
dtype: int64
```

In [43]:
```
1 s5
```

Out[43]:
```
0     8
1     9
2    10
dtype: int64
```

In [44]:
```
1 s4.add(s5)
```

Out[44]:
```
0     9.0
1    11.0
2    13.0
3     NaN
4     NaN
5     NaN
6     NaN
dtype: float64
```

In [46]:
```
1 s4.sub(s5)
```

Out[46]:
```
0   -7.0
1   -7.0
2   -7.0
3    NaN
4    NaN
5    NaN
6    NaN
dtype: float64
```

In [48]:
```
1 s4.mul(s5)
```

Out[48]:
```
0     8.0
1    18.0
2    30.0
3     NaN
4     NaN
5     NaN
6     NaN
dtype: float64
```

In [49]:
```
1 s4.div(s5)
```

Out[49]:
```
0    0.125000
1    0.222222
2    0.300000
3         NaN
4         NaN
5         NaN
6         NaN
dtype: float64
```

In [58]:
```
1 print("min",s4.min())
2 print("max",s4.max())
3 print("meidan",s4.median())
```

```
min 1
max 7
meidan 4.0
```

In [69]:
```
#creating dataframe
dates=pd.date_range("today",periods=6)
dates
num=np.random.rand(6,4)
num
columns=["A","B","C","D"]
columns
df1=pd.DataFrame(num,index=dates,columns=columns)
df1
```

Out[69]:

	A	B	C	D
2022-11-23 11:19:34.846846	0.457223	0.941358	0.381401	0.916826
2022-11-24 11:19:34.846846	0.980106	0.581302	0.232135	0.652914
2022-11-25 11:19:34.846846	0.238521	0.299744	0.034483	0.116437
2022-11-26 11:19:34.846846	0.341801	0.002434	0.530699	0.045640
2022-11-27 11:19:34.846846	0.235148	0.968091	0.660727	0.217395
2022-11-28 11:19:34.846846	0.816060	0.177028	0.220581	0.285347

In [71]:
```
#checking datatypes
df1.dtypes
```

Out[71]:
```
A    float64
B    float64
C    float64
D    float64
dtype: object
```

In [79]:
```
#checking top 5
df1.head()
```

Out[79]:

	A	B	C	D
2022-11-23 11:19:34.846846	0.457223	0.941358	0.381401	0.916826
2022-11-24 11:19:34.846846	0.980106	0.581302	0.232135	0.652914
2022-11-25 11:19:34.846846	0.238521	0.299744	0.034483	0.116437
2022-11-26 11:19:34.846846	0.341801	0.002434	0.530699	0.045640
2022-11-27 11:19:34.846846	0.235148	0.968091	0.660727	0.217395

In [80]:
```
#checking bottom 5
df1.tail(3)
```

Out[80]:

	A	B	C	D
2022-11-26 11:19:34.846846	0.341801	0.002434	0.530699	0.045640
2022-11-27 11:19:34.846846	0.235148	0.968091	0.660727	0.217395
2022-11-28 11:19:34.846846	0.816060	0.177028	0.220581	0.285347

In [81]:
```
1 df1.index
```

Out[81]:
```
DatetimeIndex(['2022-11-23 11:19:34.846846', '2022-11-24 11:19:34.846846',
               '2022-11-25 11:19:34.846846', '2022-11-26 11:19:34.846846',
               '2022-11-27 11:19:34.846846', '2022-11-28 11:19:34.846846'],
              dtype='datetime64[ns]', freq='D')
```

In [82]:
```
1 df1.columns
```

Out[82]:
```
Index(['A', 'B', 'C', 'D'], dtype='object')
```

In [84]:
```
1 df1.values
```

Out[84]:
```
array([[0.45722342, 0.94135781, 0.38140119, 0.91682595],
       [0.98010565, 0.58130245, 0.2321345 , 0.65291411],
       [0.23852054, 0.29974433, 0.03448304, 0.11643678],
       [0.34180088, 0.00243436, 0.53069859, 0.04563965],
       [0.23514824, 0.9680914 , 0.66072725, 0.21739501],
       [0.81605962, 0.17702781, 0.22058061, 0.28534735]])
```

In [87]:
```
1 #statistical data
2 df1.describe()
```

Out[87]:

	A	B	C	D
count	6.000000	6.000000	6.000000	6.000000
mean	0.511476	0.494993	0.343338	0.372426
std	0.314636	0.403076	0.228086	0.340289
min	0.235148	0.002434	0.034483	0.045640
25%	0.264341	0.207707	0.223469	0.141676
50%	0.399512	0.440523	0.306768	0.251371
75%	0.726351	0.851344	0.493374	0.561022
max	0.980106	0.968091	0.660727	0.916826

In [91]:
```
1 df1.sort_values(by="A")
```

Out[91]:

	A	B	C	D
2022-11-27 11:19:34.846846	0.235148	0.968091	0.660727	0.217395
2022-11-25 11:19:34.846846	0.238521	0.299744	0.034483	0.116437
2022-11-26 11:19:34.846846	0.341801	0.002434	0.530699	0.045640
2022-11-23 11:19:34.846846	0.457223	0.941358	0.381401	0.916826
2022-11-28 11:19:34.846846	0.816060	0.177028	0.220581	0.285347
2022-11-24 11:19:34.846846	0.980106	0.581302	0.232135	0.652914

In [92]:
```
1 df1[1:3]
```

Out[92]:

	A	B	C	D
2022-11-24 11:19:34.846846	0.980106	0.581302	0.232135	0.652914
2022-11-25 11:19:34.846846	0.238521	0.299744	0.034483	0.116437

In [98]:
```
1 df1[["A","B","C"]]
```

Out[98]:

	A	B	C
2022-11-23 11:19:34.846846	0.457223	0.941358	0.381401
2022-11-24 11:19:34.846846	0.980106	0.581302	0.232135
2022-11-25 11:19:34.846846	0.238521	0.299744	0.034483
2022-11-26 11:19:34.846846	0.341801	0.002434	0.530699
2022-11-27 11:19:34.846846	0.235148	0.968091	0.660727
2022-11-28 11:19:34.846846	0.816060	0.177028	0.220581

In [114]:
```
1 df1.iloc[1:3]
```

Out[114]:

	A	B	C	D
2022-11-24 11:19:34.846846	0.980106	0.581302	0.232135	0.652914
2022-11-25 11:19:34.846846	0.238521	0.299744	0.034483	0.116437

In [116]:
```
1 df1[["A"]].mean()
```

Out[116]:
```
A    0.511476
dtype: float64
```

In [117]:
```
1 df1["B"].sum()
```

Out[117]:
```
2.969958161962862
```

In [121]:
```
1 #file operations
2 df1.to_csv("data")
```

In [128]:
```
1 df1.to_excel("data.xlsx",sheet_name="sheet1")
```