

```
import numpy as np
import pandas as pd

dict={
    "name":["Yash","ajay","aditya"],
    "marks":[40,55,90],
    "occ":["HOD","CEO","HR"]
}
df=pd.DataFrame(dict)
print(df)
df.tail(2)
df.head(3)
print(df.tail(2))
print(df.head(3))
df.describe()
print(df.describe())
```

```

name marks occ
0 Yash 40 HOD
1 ajay 55 CEO
2 aditya 90 HR
name marks occ
1 ajay 55 CEO
2 aditya 90 HR
name marks occ
0 Yash 40 HOD
1 ajay 55 CEO
2 aditya 90 HR
marks
count 3.000000
mean 61.666667
std 25.658007
min 40.000000
25% 47.500000
50% 55.000000
75% 72.500000
max 90.000000
```

```
import numpy as np
import pandas as pd
arr1=pd.Series(np.random.random(10))
print(arr1)
```

```

0 0.708409
1 0.352534
2 0.580726
3 0.822135
4 0.470066
5 0.857800
6 0.643252
7 0.345560
8 0.623484
9 0.213174
dtype: float64
```

```
import numpy as np
import pandas as pd
newdf=pd.DataFrame(np.random.rand(3,5))
print(newdf)
print(newdf.describe())
print(newdf.index)
print(newdf.T)
print(newdf.tail(5))
print(newdf.head(100 ))
type(newdf)
newdf2=newdf
newdf2[0][0]=9989512263
print(newdf2)
newdf[0][0]=9618190176
print(newdf)
newdf.head(2)
newdf.tail(300)
newdf[0]
```

```

0 1 2 3 4
0 0.327392 0.322582 0.306516 0.439228 0.301024
1 0.940217 0.604687 0.874119 0.180599 0.785740
```

```
2 0.866797 0.388542 0.898028 0.335748 0.893676
count 3.000000 3.000000 3.000000 3.000000 3.000000
mean 0.711469 0.438603 0.692888 0.318525 0.660147
std 0.334639 0.147565 0.334821 0.130172 0.315657
min 0.327392 0.322582 0.306516 0.180599 0.301024
25% 0.597094 0.355562 0.590318 0.258173 0.543382
50% 0.866797 0.388542 0.874119 0.335748 0.785740
75% 0.903507 0.496614 0.886074 0.387488 0.839708
max 0.940217 0.604687 0.898028 0.439228 0.893676
RangeIndex(start=0, stop=3, step=1)
0 0.327392 0.940217 0.866797
1 0.322582 0.604687 0.388542
2 0.306516 0.874119 0.898028
3 0.439228 0.180599 0.335748
4 0.301024 0.785740 0.893676
0 0.327392 0.322582 0.306516 0.439228 0.301024
1 0.940217 0.604687 0.874119 0.180599 0.785740
2 0.866797 0.388542 0.898028 0.335748 0.893676
0 0.327392 0.322582 0.306516 0.439228 0.301024
1 0.940217 0.604687 0.874119 0.180599 0.785740
2 0.866797 0.388542 0.898028 0.335748 0.893676
0 9.989512e+09 0.322582 0.306516 0.439228 0.301024
1 9.402167e-01 0.604687 0.874119 0.180599 0.785740
2 8.667966e-01 0.388542 0.898028 0.335748 0.893676
0 9.618190e+09 0.322582 0.306516 0.439228 0.301024
1 9.402167e-01 0.604687 0.874119 0.180599 0.785740
2 8.667966e-01 0.388542 0.898028 0.335748 0.893676
0 9.618190e+09
1 9.402167e-01
2 8.667966e-01
Name: 0, dtype: float64
```

```
import numpy as np
import pandas as pd
```

```
newdf[0]#first column
```

```
0 9.618190e+09
1 9.402167e-01
2 8.667966e-01
Name: 0, dtype: float64
```

```
type(newdf)#checking what is it
```

```
pandas.core.frame.DataFrame
```

```
newdf.columns="A","B","C","D","E" # changing the top of the column
print(newdf)
```

```
      A      B      C      D      E
0 9.618190e+09 0.322582 0.306516 0.439228 0.301024
1 9.402167e-01 0.604687 0.874119 0.180599 0.785740
2 8.667966e-01 0.388542 0.898028 0.335748 0.893676
```

```
newdf.loc[0,'A']=420#replacing the int value
newdf.head(2)
```

	A	B	C	D	E
0	420.000000	0.322582	0.306516	0.439228	0.301024
1	0.940217	0.604687	0.874119	0.180599	0.785740

```
#newdf.loc[[1,2],['C','D']]#rows and colmns
print(newdf)
```

```
      A      B      C      D      E
0 420.000000 0.322582 0.306516 0.439228 0.301024
1 0.940217 0.604687 0.874119 0.180599 0.785740
2 0.866797 0.388542 0.898028 0.335748 0.893676
```

```
newdf.loc[(newdf['A']<0.3)]#follows the condition
```

```
newdf.loc[(newdf['A']<0.3)&(newdf['C']>0.3)]#follows the two conditions
```

A B C D E 

```
newdf.iloc[0,3]#locations
newdf.iloc[1,2]#locations
newdf.iloc[2,1]#locations
```

```
0.38854186513667766
```

```
newdf.loc[:,['B']]=100
print(newdf)
newdf.loc[:,['A','C','D','E']]=20
print(newdf)
```

	A	B	C	D	E
0	420.000000	100.0	0.306516	0.439228	0.301024
1	0.940217	100.0	0.874119	0.180599	0.785740
2	0.866797	100.0	0.898028	0.335748	0.893676

	A	B	C	D	E
0	20.0	100.0	20.0	20.0	20.0
1	20.0	100.0	20.0	20.0	20.0
2	20.0	100.0	20.0	20.0	20.0

```
newdf.mean()
```

```
A    20.0
B   100.0
C    20.0
D    20.0
E    20.0
dtype: float64
```

```
newdf.max()
```

```
A    20.0
B   100.0
C    20.0
D    20.0
E    20.0
dtype: float64
```

```
newdf.median()
```

```
A    20.0
B   100.0
C    20.0
D    20.0
E    20.0
dtype: float64
```

```
newdf.std()
```

```
A    0.0
B    0.0
C    0.0
D    0.0
E    0.0
dtype: float64
```

```
newdf.count()
```

```
A    3
B    3
C    3
D    3
E    3
dtype: int64
```