```
In [1]: import numpy as np
import pandas as pd
```

1. Create any Series and print the output

```
In [3]: a=pd.Series([1,2,3,4,5,6])
    print(a)

0    1
    1    2
    2    3
    3    4
    4    5
    5    6
    dtype: int64
```

2. Create any dataframe of 10x5 with few nan values and print the output

```
In [8]: df=pd.DataFrame(np.random.rand(10,5))
    df[4][4]=np.nan
    df[2][6]=np.nan
    df[0][3]=np.nan
    df
```

Out[8]:

	0	1	2	3	4
0	0.703089	0.349944	0.239923	0.537871	0.715836
1	0.099337	0.632751	0.735349	0.810469	0.599917
2	0.018779	0.078738	0.795391	0.090249	0.140877
3	NaN	0.115318	0.597116	0.049245	0.829471
4	0.358033	0.660131	0.883482	0.062877	NaN
5	0.382389	0.988966	0.834214	0.378425	0.549848
6	0.353578	0.609429	NaN	0.705653	0.533679
7	0.527556	0.307957	0.567367	0.997163	0.901904
8	0.017539	0.059505	0.571829	0.579723	0.923660
9	0.548897	0.391605	0.469035	0.455152	0.400070

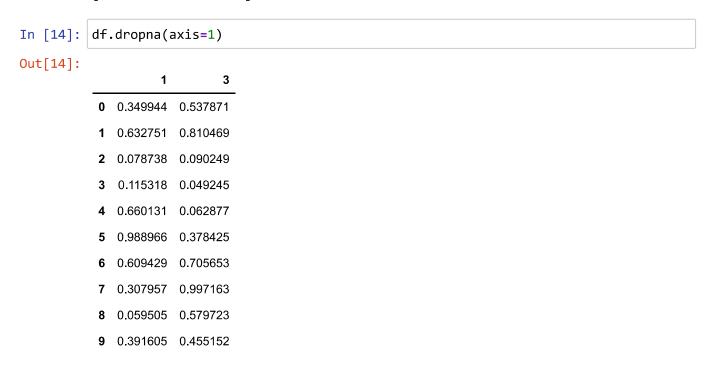
3.Display top 7 and last 6 rows and print the output

```
In [10]:
           df.head(7)
Out[10]:
                     0
              0.703089
                       0.349944 0.239923 0.537871
                                                    0.715836
               0.099337
                        0.632751
                                 0.735349
                                          0.810469
               0.018779
                       0.078738 0.795391 0.090249
                                                    0.140877
            3
                   NaN
                       0.115318 0.597116 0.049245
                                                    0.829471
               0.358033
                        0.660131 0.883482 0.062877
                                                         NaN
                                 0.834214 0.378425
                                                    0.549848
              0.382389
                        0.988966
               0.353578
                       0.609429
                                      NaN 0.705653 0.533679
In [11]:
           df.tail(6)
Out[11]:
                     0
                                                           4
                        0.660131 0.883482 0.062877
              0.358033
                                                         NaN
              0.382389
                        0.988966
                                  0.834214
                                           0.378425
              0.353578
                       0.609429
                                      NaN 0.705653
                                                    0.533679
                        0.307957 0.567367
              0.527556
                                          0.997163
                                                    0.901904
                        0.059505
               0.017539
                                 0.571829
                                           0.579723
                                                     0.923660
                        0.391605  0.469035  0.455152  0.400070
               0.548897
```

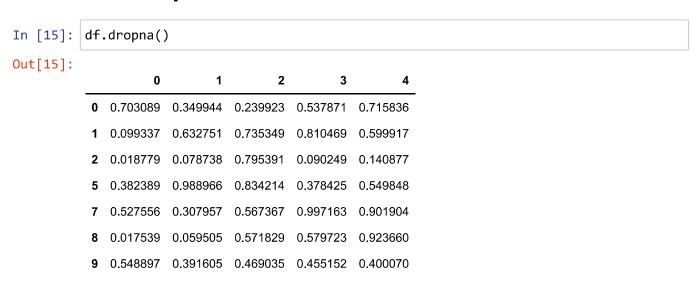
4. Fill with a constant value and print the output

```
In [16]: | df.fillna("8")
Out[16]:
              0.703089
                       0.349944 0.239923 0.537871
                                                     0.715836
               0.099337
                       0.632751
                                  0.735349 0.810469
               0.018779 0.078738
                                 0.795391 0.090249
                                                     0.140877
            3
                     8 0.115318
                                  0.597116 0.049245
                                                    0.829471
               0.358033
                       0.660131
                                  0.883482 0.062877
                        0.988966
                                 0.834214 0.378425 0.549848
              0.382389
              0.353578
                        0.609429
                                           0.705653
                                                    0.533679
              0.527556
                        0.307957
                                  0.567367 0.997163
                                                    0.901904
               0.017539
                                           0.579723
                        0.059505
                                  0.571829
                                                      0.92366
                        0.391605  0.469035  0.455152
                                                      0.40007
               0.548897
```

5. Drop the column with missing values and print the output



6. Drop the row with missing values and print the output



7. To check the presence of missing values in your dataframe

```
In [17]:
           df.isna()
Out[17]:
                   0
                          1
               False False
                            False False
                                          False
               False
                      False
                            False
                                  False
               False
                     False
                            False False
                                         False
                True
                      False
                             False
                                   False
                                          False
               False
                     False
                            False False
                                           True
               False
                      False
                             False False
                                          False
               False
                      False
                             True False
                                          False
                     False
                            False False
                                          False
               False
               False
                     False
                             False False
                                          False
               False False
                            False False
                                          False
```

8. Use operators and check the condition and print the output

```
In [18]:
           df1=df[df>0.5]
           df1
Out[18]:
                     0
                                                            4
              0.703089
                            NaN
                                      NaN 0.537871
                                                     0.715836
                   NaN
                        0.632751
                                  0.735349 0.810469
                                                     0.599917
            2
                                  0.795391
                   NaN
                            NaN
                                               NaN
                                                         NaN
                   NaN
                            NaN
                                  0.597116
                                               NaN
                                                     0.829471
                   NaN
                        0.660131
                                  0.883482
                                               NaN
                                                         NaN
                        0.988966
                                  0.834214
                                               NaN
                                                     0.549848
                   NaN
                        0.609429
                                      NaN 0.705653
                                                     0.533679
               0.527556
                                 0.567367
                                           0.997163
                                                     0.901904
                            NaN
                   NaN
                                  0.571829
                                           0.579723
                                                     0.923660
                            NaN
              0.548897
                                                         NaN
                            NaN
                                      NaN
                                               NaN
```

9. Display your output using loc and iloc, row and column heading

```
In [19]: df.loc[:2]
Out[19]:
                  0
            0.703089  0.349944  0.239923  0.537871
                                            0.715836
            0.099337
                    0.632751 0.735349
                                    0.810469
           0.018779  0.078738  0.795391  0.090249  0.140877
In [20]: df.iloc[:2]
Out[20]:
                  0
                          1
                                  2
                                          3
            0.703089
                    0.349944
                            0.239923 0.537871
                                            0.715836
```

10. Display the statistical summary of data \P

