Pandas

```
- It means panel data
- It is one of the data science package
- It is used for data anlaysis,manuplation and cleaning
- It contains 2 data structures in pandas
    - series - sequence of data
    - DataFrames :-  Data frame is a data with row and column
```

installation

```
* pip install pandas

# how to import pandas
import pandas

import pandas as pd
```

pandas series

```
- syntax :-  pd.Series(data,index,dtype)

s = "hello"
pd.Series(s)

0    hello
dtype: object

s1 = "hi hello gd afternoon".split()
pd.Series(s1)

0           hi
1        hello
2           gd
3    afternoon
dtype: object

pd.Series(s1,index=['a','b','c','d'])

a           hi
b        hello
c           gd
d    afternoon
dtype: object
```

```
li = [3,4,6,'a','n',7.8]
print(li)
pd.Series(li)
```

```
[3, 4, 6, 'a', 'n', 7.8]
```

```
0      3
1      4
2      6
3      a
4      n
5    7.8
dtype: object
```

```
t = (45,67,'f',6.8)
print(type(t))
r = pd.Series(t)
print(r)
print(type(r))
```

```
<class 'tuple'>
0     45
1     67
2      f
3    6.8
dtype: object
<class 'pandas.core.series.Series'>
```

```
d = {'key1':123,'key2':'hello','key3':'hiii'}
pd.Series(d)
```

```
key1      123
key2    hello
key3     hiii
dtype: object
```

pandas DataFrames

```
s = "welcome to python programming internship".split()
pd.DataFrame(s)
```

```
              0
0       welcome
1            to
2        python
3   programming
4    internship
```

```
li = [[1,2,3],[4,5,6]]
j = pd.DataFrame(li,index=['list1','list2'])
print(j)
print(type(j))
```

```
       0  1  2
list1  1  2  3
list2  4  5  6
<class 'pandas.core.frame.DataFrame'>
```

```
import pandas as pd
d1 = {'k1':'h1','k2':'h2','k3':'h3'}
pd.DataFrame(d1,index=[1,2,3])
```

```
   k1  k2  k3
1  h1  h2  h3
2  h1  h2  h3
3  h1  h2  h3
```

```
dic = {'Names':['raju','ravi','rani','vamsi'],
       'Marks':[90,89,94,100],'ID':[101,102,103,104]}
df1 = pd.DataFrame(dic,index=['std1','std2','std3','std4'])
df1
```

```
      Names  Marks   ID
std1   raju     90  101
std2   ravi     89  102
std3   rani     94  103
std4  vamsi    100  104
```

```
# creation of dataframes using numpy array
import numpy as np
import pandas as pd
k = np.array([[23,45,77],[44,78,28]])
print(k,type(k))
j = pd.DataFrame(k)
print(j,type(j))
```

```
[[23 45 77]
 [44 78 28]] <class 'numpy.ndarray'>
    0   1   2
0  23  45  77
1  44  78  28 <class 'pandas.core.frame.DataFrame'>
```

```
print(df1)
```

```
      Names  Marks   ID
std1   raju     90  101
std2   ravi     89  102
std3   rani     94  103
std4  vamsi    100  104
```

```
# to get only columns
df1.columns
```

```
Index(['Names', 'Marks', 'ID'], dtype='object')
```

```
# to get the no of records
print(len(df1.values))
```

```
4
```

```
# to get the row values
df1.values
```

```
array([['raju', 90, 101],
       ['ravi', 89, 102],
       ['rani', 94, 103],
       ['vamsi', 100, 104]], dtype=object)
```

```
# to get the index positions
print(df1.index)
```

```
Index(['std1', 'std2', 'std3', 'std4'], dtype='object')
```

```
# to get the no of rows and no of columns
print(df1.shape)
```

```
(4, 3)
```

```
print(df1['Names'])
```

```
std1     raju
std2     ravi
std3     rani
std4    vamsi
Name: Names, dtype: object
```

```
print(df1['Marks'])
```

```
std1     90
std2     89
std3     94
std4    100
Name: Marks, dtype: int64
```

```
print(df1[['Marks','ID']])
```

```
      Marks   ID
std1     90  101
std2     89  102
std3     94  103
std4    100  104
```

```
# head() - to get first 5 records
df1.head()
```

```
      Names  Marks   ID
std1   raju     90  101
std2   ravi     89  102
std3   rani     94  103
std4  vamsi    100  104
```

```
df1.head(1)
```

```
     Names  Marks   ID
std1  raju     90  101
```

```
df1.head(2)
```

```
     Names  Marks   ID
std1  raju     90  101
std2  ravi     89  102
```

```
df1.tail()
```

```
      Names  Marks   ID
std1   raju     90  101
std2   ravi     89  102
std3   rani     94  103
std4  vamsi    100  104
```

```
df1.tail(1)
```

```
      Names  Marks   ID
std4  vamsi    100  104
```

```
df1.tail(2)
```

```
      Names  Marks   ID
std3   rani     94  103
std4  vamsi    100  104
```

```
# To add new record to the data frame
df1.loc[4]=['sony',78,105]
print(df1)
```

```
      Names  Marks   ID
std1   raju     90  101
std2   ravi     89  102
std3   rani     94  103
std4  vamsi    100  104
4      sony     78  105
```

```
# to add new column
df1['Branch']=['ece','cse','mech','civil','eee']
df1
```

```
      Names  Marks   ID Branch
std1   raju     90  101    ece
std2   ravi     89  102    cse
std3   rani     94  103   mech
std4  vamsi    100  104  civil
4      sony     78  105    eee
```

```
# particular value update in DATAFRAME
df1.loc['std4','Branch']='Arts'
df1
```

```
      Names  Marks     ID Branch
std1   raju   90.0  101.0    ece
std2   ravi   89.0  102.0    cse
std3   rani   94.0  103.0   mech
std4  vamsi  100.0  104.0   Arts
4      sony   78.0  105.0    eee
3       NaN    NaN    NaN   Arts
```

```
# to get particular record details
df1.loc['std2']
```

```
Names      ravi
Marks      89.0
ID        102.0
Branch      cse
Name: std2, dtype: object
```

```
df1.loc[4]
```

```
Names      sony
Marks      78.0
ID        105.0
Branch      eee
Name: 4, dtype: object
```

```
# how to delete particular row or column - drop()
# row wise --- axis = 0
# column wise -  axis - 1
print(df1)
```

```
      Names  Marks     ID Branch
std1   raju   90.0  101.0    ece
std2   ravi   89.0  102.0    cse
std3   rani   94.0  103.0   mech
std4  vamsi  100.0  104.0   Arts
```

```
4      sony   78.0  105.0    eee
3       NaN    NaN    NaN   Arts
```

```
# particular row delete
df1 = df1.drop(3,axis=0)
df1
```

```
      Names  Marks     ID Branch
std1   raju   90.0  101.0    ece
std2   ravi   89.0  102.0    cse
std3   rani   94.0  103.0   mech
std4  vamsi  100.0  104.0   Arts
4      sony   78.0  105.0    eee
```

```
# particular column delte
df1 = df1.drop('ID',axis=1)
df1
```

```
      Names  Marks Branch
std1   raju   90.0    ece
std2   ravi   89.0    cse
std3   rani   94.0   mech
std4  vamsi  100.0   Arts
4      sony   78.0    eee
```

```
# To rename the particular single column name
df1.rename(columns={'Marks':'MARKS'},inplace=True)
df1
```

```
      Names  MARKS Branch
std1   raju   90.0    ece
std2   ravi   89.0    cse
std3   rani   94.0   mech
std4  vamsi  100.0   Arts
4      sony   78.0    eee
```

```
# Rename multiple column names
df1.columns=['NAMES','MARKS','BRANCH']
df1
```

```
      NAMES  MARKS BRANCH
std1   raju   90.0    ece
std2   ravi   89.0    cse
std3   rani   94.0   mech
std4  vamsi  100.0   Arts
4      sony   78.0    eee
```

```
# To delete all row records
df1.drop(df1.index,inplace=True)
df1
```

```
Empty DataFrame
Columns: [NAMES, MARKS, BRANCH]
Index: []
```

```
# To delete all columns
df1.drop(df1.columns,inplace=True,axis=1)
df1
```

```
Empty DataFrame
Columns: []
Index: []
```