

## Pandas Libraries using dataframe

In [13]:

```
##Series Object in Pandas
import pandas as pd
data=[1,2,3,4]
series1=pd.Series(data)
series1
```

Out[13]:

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

In [14]:

```
type(series1)
```

Out[14]:

```
pandas.core.series.Series
```

In [15]:

```
##Chianging the index of a series Object
series1=pd.Series(data,index=['a','b','c','d'])
series1
```

Out[15]:

```
a    1
b    2
c    3
d    4
dtype: int64
```

In [16]:

```
## How to create a dataframe
##Creating a dataframe using a list
import pandas as pd
data=[1,2,3,4,5]
df=pd.DataFrame(data)
df
```

Out[16]:

	0
0	1
1	2
2	3
3	4
4	5

In [18]:

```
##Creating a dataframe using a dictionary
dictionary={'fruits':['apples','banana','mangoes'], 'count':[10,20,15]}
df=pd.DataFrame(dictionary)
df
```

Out[18]:

	fruits	count
0	apples	10
1	banana	20
2	mangoes	15

In [19]:

```
##Creating a dataframe using a series
series=pd.Series([6,12],index=['a','b'])
df=pd.DataFrame(series)
df
```

Out[19]:

	0
a	6
b	12

In [20]:

```
##Creating a dataframe using a numpy array
import numpy as np
numpyarray=np.array([[50000,60000],['john','james']])
df=pd.DataFrame({'name':numpyarray[1], 'salary':numpyarray[0]})
df
```

Out[20]:

	name	salary
0	john	50000
1	james	60000

## How to perform a merge Operation?

In [22]:

```
player=['Player1','Player2','Player3']
point=[8,9,6]
title=['Game1','Game2','Game3']
df1=pd.DataFrame({'Player':player,'points':point,'Title':title})
df1
```

Out[22]:

	Player	points	Title
0	Player1	8	Game1
1	Player2	9	Game2
2	Player3	6	Game3

In [26]:

```
player=['Player1','Player5','Player6']
power=['punch','kick','Elbow']
title=['Game1','Game5','Game6']
df2=pd.DataFrame({'Player':player,'Power':power,'Title':title})
df2
```

Out[26]:

	Player	Power	Title
0	Player1	punch	Game1
1	Player5	kick	Game5
2	Player6	Elbow	Game6

In [28]:

```
##Inner Merge
df1.merge(df2,on='Title',how='inner')
```

Out[28]:

	Player_x	points	Title	Player_y	Power
0	Player1	8	Game1	Player1	punch

In [30]:

```
df1.merge(df2)
```

Out[30]:

	Player	points	Title	Power
0	Player1	8	Game1	punch

In [32]:

```
##left merge
df1.merge(df2,on='Player',how='left')
```

Out[32]:

	Player	points	Title_x	Power	Title_y
0	Player1	8	Game1	punch	Game1
1	Player2	9	Game2	NaN	NaN
2	Player3	6	Game3	NaN	NaN

In [34]:

```
##Right merge
df1.merge(df2,on='Player',how='right')
```

Out[34]:

	Player	points	Title_x	Power	Title_y
0	Player1	8.0	Game1	punch	Game1
1	Player5	NaN	NaN	kick	Game5
2	Player6	NaN	NaN	Elbow	Game6

In [36]:

```
##outer merge
df1.merge(df2,on='Player',how='outer')
```

Out[36]:

	Player	points	Title_x	Power	Title_y
--	--------	--------	---------	-------	---------

0	Player	points	Title_x	Power	Title_y
	Player1	8.0	Game1	punch	Game1
1	Player2	9.0	Game2	NaN	NaN
2	Player3	6.0	Game3	NaN	NaN
3	Player5	NaN	NaN	kick	Game5
4	Player6	NaN	NaN	Elbow	Game6

## How to perform join Operation in Pandas

In [54]:

```
player1=['Player1','Player2','Player3']
point=[8,9,6]
title=['Game1','Game2','Game3']
df3=pd.DataFrame({'Player':player1,'Points':point,'Title':title},index=['L1','L2','L3'])
df3
```

Out[54]:

	Player	Points	Title
L1	Player1	8	Game1
L2	Player2	9	Game2
L3	Player3	6	Game3

In [55]:

```
player=['Player1','Player5','Player6']
power=['punch','kick','Elbow']
title=['Game1','Game5','Game6']
df4=pd.DataFrame({'Player':player,'Power':power,'Titles':title},index=['L2','L3','L4'])
df4
```

Out[55]:

	Player	Power	Titles
L2	Player1	punch	Game1
L3	Player5	kick	Game5
L4	Player6	Elbow	Game6

In [57]:

```
##inner join
##df3.join(df4,how='inner')
```

In [58]:

```
##df3.join(df4,how='left')
```

In [59]:

```
##df3.join(df4,how='right')
```

In [60]:

```
##df3.join(df4,how='outer')
```

## How to perform concatenate two dataFrame

In [63]:

```
pd.concat([df1,df2])
```

C:\Users\ganesh chaurasiya\Desktop\python3.7\lib\site-packages\ipykernel\_launcher.py:2:  
FutureWarning: Sorting because non-concatenation axis is not aligned. A future version  
of pandas will change to not sort by default.

To accept the future behavior, pass 'sort=False'.

To retain the current behavior and silence the warning, pass 'sort=True'.

Out[63]:

	Player	Power	Title	points
0	Player1	NaN	Game1	8.0
1	Player2	NaN	Game2	9.0
2	Player3	NaN	Game3	6.0
L2	Player1	punch	Game1	NaN
L3	Player5	kick	Game5	NaN
L4	Player6	Elbow	Game6	NaN

In [65]:

```
import pandas as pd  
data = pd.read_csv("Iris.csv")  
data
```

Out[65]:

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	1	5.1	3.5	1.4	0.2	Iris-setosa
1	2	4.9	3.0	1.4	0.2	Iris-setosa
2	3	4.7	3.2	1.3	0.2	Iris-setosa
3	4	4.6	3.1	1.5	0.2	Iris-setosa
4	5	5.0	3.6	1.4	0.2	Iris-setosa
5	6	5.4	3.9	1.7	0.4	Iris-setosa
6	7	4.6	3.4	1.4	0.3	Iris-setosa
7	8	5.0	3.4	1.5	0.2	Iris-setosa
8	9	4.4	2.9	1.4	0.2	Iris-setosa
9	10	4.9	3.1	1.5	0.1	Iris-setosa
10	11	5.4	3.7	1.5	0.2	Iris-setosa
11	12	4.8	3.4	1.6	0.2	Iris-setosa
12	13	4.8	3.0	1.4	0.1	Iris-setosa
13	14	4.3	3.0	1.1	0.1	Iris-setosa
14	15	5.8	4.0	1.2	0.2	Iris-setosa
15	16	5.7	4.4	1.5	0.4	Iris-setosa
16	17	5.4	3.9	1.3	0.4	Iris-setosa
17	18	5.1	3.5	1.4	0.3	Iris-setosa
18	19	5.7	3.8	1.7	0.3	Iris-setosa
19	20	5.1	3.8	1.5	0.3	Iris-setosa
20	21	5.4	3.4	1.7	0.2	Iris-setosa
21	22	5.1	3.7	1.5	0.4	Iris-setosa
22	23	4.6	3.6	1.0	0.2	Iris-setosa
23	24	5.1	3.3	1.7	0.5	Iris-setosa

24	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
25	26	5.0	3.0	1.6	0.2	Iris-setosa
26	27	5.0	3.4	1.6	0.4	Iris-setosa
27	28	5.2	3.5	1.5	0.2	Iris-setosa
28	29	5.2	3.4	1.4	0.2	Iris-setosa
29	30	4.7	3.2	1.6	0.2	Iris-setosa
...	...	...	...	...	...	...
120	121	6.9	3.2	5.7	2.3	Iris-virginica
121	122	5.6	2.8	4.9	2.0	Iris-virginica
122	123	7.7	2.8	6.7	2.0	Iris-virginica
123	124	6.3	2.7	4.9	1.8	Iris-virginica
124	125	6.7	3.3	5.7	2.1	Iris-virginica
125	126	7.2	3.2	6.0	1.8	Iris-virginica
126	127	6.2	2.8	4.8	1.8	Iris-virginica
127	128	6.1	3.0	4.9	1.8	Iris-virginica
128	129	6.4	2.8	5.6	2.1	Iris-virginica
129	130	7.2	3.0	5.8	1.6	Iris-virginica
130	131	7.4	2.8	6.1	1.9	Iris-virginica
131	132	7.9	3.8	6.4	2.0	Iris-virginica
132	133	6.4	2.8	5.6	2.2	Iris-virginica
133	134	6.3	2.8	5.1	1.5	Iris-virginica
134	135	6.1	2.6	5.6	1.4	Iris-virginica
135	136	7.7	3.0	6.1	2.3	Iris-virginica
136	137	6.3	3.4	5.6	2.4	Iris-virginica
137	138	6.4	3.1	5.5	1.8	Iris-virginica
138	139	6.0	3.0	4.8	1.8	Iris-virginica
139	140	6.9	3.1	5.4	2.1	Iris-virginica
140	141	6.7	3.1	5.6	2.4	Iris-virginica
141	142	6.9	3.1	5.1	2.3	Iris-virginica
142	143	5.8	2.7	5.1	1.9	Iris-virginica
143	144	6.8	3.2	5.9	2.3	Iris-virginica
144	145	6.7	3.3	5.7	2.5	Iris-virginica
145	146	6.7	3.0	5.2	2.3	Iris-virginica
146	147	6.3	2.5	5.0	1.9	Iris-virginica
147	148	6.5	3.0	5.2	2.0	Iris-virginica
148	149	6.2	3.4	5.4	2.3	Iris-virginica
149	150	5.9	3.0	5.1	1.8	Iris-virginica

150 rows × 6 columns

In [66]:

```
type(data)
```

Out[66]:

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

In [67]:

```
data.head(10)
```

Out[67]:

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	1	5.1	3.5	1.4	0.2	Iris-setosa
1	2	4.9	3.0	1.4	0.2	Iris-setosa
2	3	4.7	3.2	1.3	0.2	Iris-setosa
3	4	4.6	3.1	1.5	0.2	Iris-setosa
4	5	5.0	3.6	1.4	0.2	Iris-setosa
5	6	5.4	3.9	1.7	0.4	Iris-setosa
6	7	4.6	3.4	1.4	0.3	Iris-setosa
7	8	5.0	3.4	1.5	0.2	Iris-setosa
8	9	4.4	2.9	1.4	0.2	Iris-setosa
9	10	4.9	3.1	1.5	0.1	Iris-setosa

In [68]:

```
data.tail()
```

Out[68]:

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
145	146	6.7	3.0	5.2	2.3	Iris-virginica
146	147	6.3	2.5	5.0	1.9	Iris-virginica
147	148	6.5	3.0	5.2	2.0	Iris-virginica
148	149	6.2	3.4	5.4	2.3	Iris-virginica
149	150	5.9	3.0	5.1	1.8	Iris-virginica

In [69]:

```
data.shape
```

Out[69]:

```
(150, 6)
```

In [71]:

```
data.info(null_counts=True)
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 150 entries, 0 to 149
Data columns (total 6 columns):
Id                150 non-null int64
SepalLengthCm     150 non-null float64
SepalWidthCm      150 non-null float64
PetalLengthCm     150 non-null float64
PetalWidthCm      150 non-null float64
Species           150 non-null object
dtypes: float64(4), int64(1), object(1)
memory usage: 6.5+ KB
```

In [72]:

```
data.mean()
```

Out[72]:

```
Id                75.500000
SepalLengthCm     5.843333
SepalWidthCm      3.054000
PetalLengthCm     3.758667
PetalWidthCm      1.198667
dtype: float64
```

In [73]:

```
data.median()
```

Out[73]:

```
Id                75.50
SepalLengthCm     5.80
SepalWidthCm      3.00
PetalLengthCm     4.35
PetalWidthCm      1.30
dtype: float64
```

In [74]:

```
data.max()
```

Out[74]:

```
Id                150
SepalLengthCm     7.9
SepalWidthCm      4.4
PetalLengthCm     6.9
PetalWidthCm      2.5
Species           Iris-virginica
dtype: object
```

In [75]:

```
data.min()
```

Out[75]:

```
Id                1
SepalLengthCm     4.3
SepalWidthCm      2
PetalLengthCm     1
PetalWidthCm      0.1
Species           Iris-setosa
dtype: object
```

In [76]:

```
data.count()
```

Out[76]:

```
Id                150
SepalLengthCm     150
SepalWidthCm      150
PetalLengthCm     150
PetalWidthCm      150
Species           150
dtype: int64
```

In [78]:

```
data.describe()
```

Out[78]:

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm
count	150.000000	150.000000	150.000000	150.000000	150.000000
mean	75.500000	5.843333	3.054000	3.758667	1.198667
std	43.445368	0.828066	0.433594	1.764420	0.763161
min	1.000000	4.300000	2.000000	1.000000	0.100000
25%	38.250000	5.100000	2.800000	1.600000	0.300000



50%	75.500000	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm
75%	112.750000		6.400000	3.300000	5.100000	1.800000
max	150.000000		7.900000	4.400000	6.900000	2.500000

In [81]:

```
##data cleaning from dataset :df.rename()
data=data.rename(columns={'Unnamed: 1':'model'})
data
```

Out[81]:

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	1	5.1	3.5	1.4	0.2	Iris-setosa
1	2	4.9	3.0	1.4	0.2	Iris-setosa
2	3	4.7	3.2	1.3	0.2	Iris-setosa
3	4	4.6	3.1	1.5	0.2	Iris-setosa
4	5	5.0	3.6	1.4	0.2	Iris-setosa
5	6	5.4	3.9	1.7	0.4	Iris-setosa
6	7	4.6	3.4	1.4	0.3	Iris-setosa
7	8	5.0	3.4	1.5	0.2	Iris-setosa
8	9	4.4	2.9	1.4	0.2	Iris-setosa
9	10	4.9	3.1	1.5	0.1	Iris-setosa
10	11	5.4	3.7	1.5	0.2	Iris-setosa
11	12	4.8	3.4	1.6	0.2	Iris-setosa
12	13	4.8	3.0	1.4	0.1	Iris-setosa
13	14	4.3	3.0	1.1	0.1	Iris-setosa
14	15	5.8	4.0	1.2	0.2	Iris-setosa
15	16	5.7	4.4	1.5	0.4	Iris-setosa
16	17	5.4	3.9	1.3	0.4	Iris-setosa
17	18	5.1	3.5	1.4	0.3	Iris-setosa
18	19	5.7	3.8	1.7	0.3	Iris-setosa
19	20	5.1	3.8	1.5	0.3	Iris-setosa
20	21	5.4	3.4	1.7	0.2	Iris-setosa
21	22	5.1	3.7	1.5	0.4	Iris-setosa
22	23	4.6	3.6	1.0	0.2	Iris-setosa
23	24	5.1	3.3	1.7	0.5	Iris-setosa
24	25	4.8	3.4	1.9	0.2	Iris-setosa
25	26	5.0	3.0	1.6	0.2	Iris-setosa
26	27	5.0	3.4	1.6	0.4	Iris-setosa
27	28	5.2	3.5	1.5	0.2	Iris-setosa
28	29	5.2	3.4	1.4	0.2	Iris-setosa
29	30	4.7	3.2	1.6	0.2	Iris-setosa
...	...	...	...	...	...	...
120	121	6.9	3.2	5.7	2.3	Iris-virginica
121	122	5.6	2.8	4.9	2.0	Iris-virginica
122	123	7.7	2.8	6.7	2.0	Iris-virginica
123	124	6.3	2.7	4.9	1.8	Iris-virginica
124	125	6.7	3.3	5.7	2.1	Iris-virginica
125	126	7.2	3.2	6.0	1.8	Iris-virginica
126	127	6.2	2.8	4.8	1.8	Iris-virginica
127	128	6.1	3.0	4.9	1.8	Iris-virginica

		SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Iris-Species
128	129					
129	130	7.2	3.0	5.8	1.6	Iris-virginica
130	131	7.4	2.8	6.1	1.9	Iris-virginica
131	132	7.9	3.8	6.4	2.0	Iris-virginica
132	133	6.4	2.8	5.6	2.2	Iris-virginica
133	134	6.3	2.8	5.1	1.5	Iris-virginica
134	135	6.1	2.6	5.6	1.4	Iris-virginica
135	136	7.7	3.0	6.1	2.3	Iris-virginica
136	137	6.3	3.4	5.6	2.4	Iris-virginica
137	138	6.4	3.1	5.5	1.8	Iris-virginica
138	139	6.0	3.0	4.8	1.8	Iris-virginica
139	140	6.9	3.1	5.4	2.1	Iris-virginica
140	141	6.7	3.1	5.6	2.4	Iris-virginica
141	142	6.9	3.1	5.1	2.3	Iris-virginica
142	143	5.8	2.7	5.1	1.9	Iris-virginica
143	144	6.8	3.2	5.9	2.3	Iris-virginica
144	145	6.7	3.3	5.7	2.5	Iris-virginica
145	146	6.7	3.0	5.2	2.3	Iris-virginica
146	147	6.3	2.5	5.0	1.9	Iris-virginica
147	148	6.5	3.0	5.2	2.0	Iris-virginica
148	149	6.2	3.4	5.4	2.3	Iris-virginica
149	150	5.9	3.0	5.1	1.8	Iris-virginica

150 rows × 6 columns

In [83]:

```
'''data.qsec=data.qsec.fillna(data.mean())
data'''
```

Out[83]:

```
'data.qsec=data.qsec.fillna(data.mean())\ndata'
```

In [85]:

```
##drop unwanted column
'''data=data.drop(columns=['S.NO'])
data'''
```

Out[85]:

```
"data=data.drop(columns=['S.NO'])\ndata"
```

In [86]:

```
data.iloc[:,1]
```

Out[86]:

```
0      5.1
1      4.9
2      4.7
3      4.6
4      5.0
5      5.4
6      4.6
7      5.0
8      4.4
9      4.9
10     5.4
11     4.8
```

```
12      4.8
13      4.3
14      5.8
15      5.7
16      5.4
17      5.1
18      5.7
19      5.1
20      5.4
21      5.1
22      4.6
23      5.1
24      4.8
25      5.0
26      5.0
27      5.2
28      5.2
29      4.7
...
120     6.9
121     5.6
122     7.7
123     6.3
124     6.7
125     7.2
126     6.2
127     6.1
128     6.4
129     7.2
130     7.4
131     7.9
132     6.4
133     6.3
134     6.1
135     7.7
136     6.3
137     6.4
138     6.0
139     6.9
140     6.7
141     6.9
142     5.8
143     6.8
144     6.7
145     6.7
146     6.3
147     6.5
148     6.2
149     5.9
Name: SepalLengthCm, Length: 150, dtype: float64
```

In [89]:

```
##see all the record of SepalLengthCm columns
data.loc[:, 'SepalLengthCm']
```

In [ ]: