

DATAFRAME

September 24, 2019

```
In [51]: print("\nSANJEEV SHARMA")
         print("\nInformatics Practices(New)\nCLASS XII\nCode No. 065 - 2019-20\nDataFrame")
```

SANJEEV SHARMA

Informatics Practices(New)
CLASS XII
Code No. 065 - 2019-20
DataFrame

```
In [57]: # PYTHON PANDAS BASIC DATAFRAME OPERATIONS
import pandas as pd
import numpy as np

print ("\nPYTHON PANDAS BASIC DATAFRAME OPERATIONS ")

data = {'Name': ['Amba', 'Radha', 'Sharada', 'Ayan'],
        'Score': [90,76,88,89]}

df=pd.DataFrame(data)
print("\nINITIAL DATAFRAME IS :\n")
print(df)

print("\nMaximum Value in The Entire DataFrame is :",D1.values.max())

print("\nThe Minimum Value in Score Column is :",df['Score'].min())
print("\nThe Maximum Value in Score Column is :",df['Score'].max())
print("\nThe Mean of Score Column is :",df['Score'].mean())
print("\nNumber of Records in the DataFrame are:\n",df.count())
print("\nSum of all the Values under Score Column is :",df['Score'].sum())

print("\nSorting of Dataframe with Name column \n",df.sort_values('Name'))

print("\nUse of loc to Change the Value of Second Row and Score Column")
df.loc[1, 'Score'] =85
print(df)
```

```

print("\nUse of loc to Print Fourth Row \n",df.loc[3, ['Name','Score']])

print("\nAdd New Column in the DataFrame")
df['position'] = [1,2,3,4]
print("\nAfter Adding New Column in the DataFrame\n",df)

print("\nAppend New Record at the End of the DataFrame")
newrow={'name':'Gauri','score':92,'position':5}
df=df.append(newrow,ignore_index=True)
print("\nAfter Adding New Record at the End of the DataFrame\n",df)

print("\nUse of fillna to Replace NaN With 0\n")
df=df.fillna(0)
print(df)

print("\nUse of where condition in DataFrame \n")
print("\nRows where Score value > 80\n" ,df.where(df>88))
print("\nReplace 90 with 92 Under Score column\n",df.where(df==90,92))
print("\nAfter Replacing 90 with 92 \n ",df)

print("\nUse of reindex in DataFrame \n")
print(df.reindex([4,3,2,1,0]))
print("\nUse of reindex in DataFrame if index do not match a NaN is filled\n")
print(df.reindex([40,3,2,1,0]))

print("\nDelete Row from DataFrame at Index Number 1")
df=df.drop(1)
print(df)

print("\nCreating Dataframe with Nested List and Custom Index and Heading")
lst=[[ 'S1', 'Himachal',17],[ 'S1', 'Punjab', 16],[ 'S3', 'Haryana',18]]
df2=pd.DataFrame(lst , index = [1,2,3], columns =['State Code','State Name','Rank'])
print(df2)

print("\nPerforming Basic Arithmetic Operations in DataFrame")
L1=[[10,20,30],[40,50,60]]
L2=[[15,25,35],[10,60,70]]

D1=pd.DataFrame(L1)
D2=pd.DataFrame(L2)
print("\nINITIAL DATAFRAME D1 :\n")
print(D1)
print("\nINITIAL DATAFRAME D2 :\n")
print(D2)
print("\nADDITION -> D1+D2 :\n")
print(D1+D2)
print("\nSUBTRACTION -> D1-D2 :\n")

```

```

print(D1-D2)
print("\nMULTIPLICATION -> D1*D2 :\n")
print(D1 * D2)
print("\nDIVISION -> D2/D1 :\n")
print(D2 // D1)

print("\n DATAFRAME D1 :\n")
print(D1)
print("\nMean/Average of DataFrame D1 Column Second - Third of all Rows :\n")
print(D1.loc[0:3,1:3].mean())

```

PYTHON PANDAS BASIC DATAFRAME OPERATIONS

INITIAL DATAFRAME IS :

	Name	Score
0	Amba	90
1	Radha	76
2	Sharada	88
3	Ayan	89

Maximum Value in The Entire DataFrame is : 60

The Minimum Value in Score Column is : 76

The Maximum Value in Score Column is : 90

The Mean of Score Column is : 85.75

Number of Records in the DataFrame are:

```

Name      4
Score     4
dtype: int64

```

Sum of all the Values under Score Column is : 343

Sorting of Dataframe with Name column

	Name	Score
0	Amba	90
3	Ayan	89
1	Radha	76
2	Sharada	88

Use of loc to Change the Value of Second Row and Score Column

	Name	Score
0	Amba	90
1	Radha	85

```

2 Sharada      88
3   Ayan      89

```

Use of loc to Print Fourth Row

```

Name      Ayan
Score     89
Name: 3, dtype: object

```

Add New Column in the DataFrame

After Adding New Column in the DataFrame

```

   Name  Score  position
0  Amba    90         1
1  Radha   85         2
2  Sharada 88         3
3   Ayan   89         4

```

Append New Record at the End of the DataFrame

After Adding New Record at the End of the DataFrame

```

   Name  Score  position  name  score
0  Amba  90.0         1   NaN   NaN
1  Radha 85.0         2   NaN   NaN
2  Sharada 88.0         3   NaN   NaN
3   Ayan 89.0         4   NaN   NaN
4   NaN  NaN         5  Gauri  92.0

```

Use of fillna to Replace NaN With 0

```

   Name  Score  position  name  score
0  Amba  90.0         1     0   0.0
1  Radha 85.0         2     0   0.0
2  Sharada 88.0         3     0   0.0
3   Ayan 89.0         4     0   0.0
4     0   0.0         5  Gauri  92.0

```

Use of where condition in DataFrame

Rows where Score value > 80

```

   Name  Score  position  name  score
0  Amba  90.0         NaN     0   NaN
1  Radha   NaN         NaN     0   NaN
2  Sharada  NaN         NaN     0   NaN
3   Ayan 89.0         NaN     0   NaN
4     0   NaN         NaN  Gauri  92.0

```

Replace 90 with 92 Under Score column

	Name	Score	position	name	score
0	92	90.0	92	92	92.0
1	92	92.0	92	92	92.0
2	92	92.0	92	92	92.0
3	92	92.0	92	92	92.0
4	92	92.0	92	92	92.0

After Replacing 90 with 92

	Name	Score	position	name	score
0	Amba	90.0	1	0	0.0
1	Radha	85.0	2	0	0.0
2	Sharada	88.0	3	0	0.0
3	Ayan	89.0	4	0	0.0
4	0	0.0	5	Gauri	92.0

Use of reindex in DataFrame

	Name	Score	position	name	score
4	0	0.0	5	Gauri	92.0
3	Ayan	89.0	4	0	0.0
2	Sharada	88.0	3	0	0.0
1	Radha	85.0	2	0	0.0
0	Amba	90.0	1	0	0.0

Use of reindex in DataFrame if index do not match a NaN is filled

	Name	Score	position	name	score
40	NaN	NaN	NaN	NaN	NaN
3	Ayan	89.0	4.0	0	0.0
2	Sharada	88.0	3.0	0	0.0
1	Radha	85.0	2.0	0	0.0
0	Amba	90.0	1.0	0	0.0

Delete Row from DataFrame at Index Number 1

	Name	Score	position	name	score
0	Amba	90.0	1	0	0.0
2	Sharada	88.0	3	0	0.0
3	Ayan	89.0	4	0	0.0
4	0	0.0	5	Gauri	92.0

Creating Dataframe with Nested List and Custom Index and Heading

	State	Code	State	Name	Rank
1	S1		Himachal		17
2	S1		Punjab		16
3	S3		Haryana		18

Performing Basic Arithmetic Operations in DataFrame

INITIAL DATAFRAME D1 :

```
   0  1  2
0  10 20 30
1  40 50 60
```

INITIAL DATAFRAME D2 :

```
   0  1  2
0  15 25 35
1  10 60 70
```

ADDITION -> D1+D2 :

```
   0  1  2
0  25 45 65
1  50 110 130
```

SUBSTRACTION -> D1-D2 :

```
   0  1  2
0  -5 -5 -5
1  30 -10 -10
```

MULTIPLICATION -> D1*D2 :

```
   0  1  2
0  150 500 1050
1  400 3000 4200
```

DIVISION -> D2+D1 :

```
   0  1  2
0  1  1  1
1  0  1  1
```

DATAFRAME D1 :

```
   0  1  2
0  10 20 30
1  40 50 60
```

Mean/Average of DataFrame D1 Column Second - Third of all Rows :

```
1    35.0
2    45.0
dtype: float64
```

In []: