**DataFrame Operations**

**Column operations:**

**Selecting columns:**

import pandas as pd

x = {'one' : pd.Series([1, 2, 3], index=['a', 'b', 'c']),

'two' : pd.Series([1, 2, 3, 4], index=['a', 'b', 'c', 'd'])}

df = pd.DataFrame(x)

print df ['one']

o/p:

a 1.0

b 2.0

c 3.0

d NaN

Name: one, dtype: float64

**Adding Columns:**

import pandas as pd

x = {'one' : pd.Series([1, 2, 3], index=['a', 'b', 'c']),

'two' : pd.Series([1, 2, 3, 4], index=['a', 'b', 'c', 'd'])}

df = pd.DataFrame(x)

# Adding a new column

df['three']=pd.Series([10,20,30],index=['a','b','c'])

print(df)

one two three

a 1.0 1 10.0

b 2.0 2 20.0

c 3.0 3 30.0

d NaN 4 NaN

#Adding a new column using the existing columns in DataFrame:

df['four']=df['one']+df['three']

print(df)

o/p:

one two three four

a 1.0 1 10.0 11.0

b 2.0 2 20.0 22.0

c 3.0 3 30.0 33.0

d NaN 4 NaN NaN

**Deleting a Column:**

Columns can be deleted or popped

x = {'one' : pd.Series([1, 2, 3], index=['a', 'b', 'c']),

'two' : pd.Series([1, 2, 3, 4], index=['a', 'b', 'c', 'd']),

'three' : pd.Series([10,20,30], index=['a','b','c'])}  
df = pd.DataFrame(x)

Print(df)

o/p:

one three two

a 1.0 10.0 1

b 2.0 20.0 2

c 3.0 30.0 3

d NaN NaN 4

# using del function

Del(df['one'])

Print(df)

o/p:

three two

a 10.0 1

b 20.0 2

c 30.0 3

d NaN 4

# using pop function

df.pop('two')

print(df)

o/p:

three

a 10.0

b 20.0

c 30.0

d NaN

**Row operations:**

***Selecting Rows:***

**Rows in a Dataframe can be selected in two ways:**

**(1) row label ; (2) integer location**

***Selection by Label***

Rows can be selected by passing row label to a **loc** function.

import pandas as pd

x = {'one' : pd.Series([1, 2, 3], index=['a', 'b', 'c']),

'two' : pd.Series([1, 2, 3, 4], index=['a', 'b', 'c', 'd'])}

df = pd.DataFrame(x)

print df.loc['b']

o/p:

one 2.0

two 2.0

Name: b, dtype: float64

The result is a series with labels as column names of the DataFrame. And, the Name of

the series is the label with which it is retrieved.

***Selection by integer location***

Rows can be selected by passing integer location to an **iloc** function.

x = {'one' : pd.Series([1, 2, 3], index=['a', 'b', 'c']),

'two' : pd.Series([1, 2, 3, 4], index=['a', 'b', 'c', 'd'])}

df = pd.DataFrame(x)

print df.iloc[2]

o/p:

one 3.0

two 3.0

Name: c, dtype: float64

**Adding Rows:**

Add new rows to a DataFrame using the **append** function. This function will append the

rows at the end.

import pandas as pd

df1 = pd.DataFrame([[1, 2], [3, 4]], columns=['a','b'])

df2 = pd.DataFrame([[5, 6], [7, 8]], columns=['a','b'])

df1 = df1.append(df2)

print(df1)

o/p:

a b

0 1 2

1 3 4

0 5 6

1 7 8

**Slicing Rows:**

Multiple rows can be selected using ‘ : ’ operator.

import pandas as pd

x = {'one' : pd.Series([1, 2, 3], index=['a', 'b', 'c']),

'two' : pd.Series([1, 2, 3, 4], index=['a', 'b', 'c', 'd'])}

df = pd.DataFrame(x)

print df[2:4]

o/p:

one two

c 3.0 3

d NaN 4

**Deleting Rows:**

Use index label to delete or drop rows from a DataFrame. If label is duplicated, then

multiple rows will be dropped.

If you observe, in the above example, the labels are duplicated. Let us drop a label and will see how many rows will get dropped.

df1 = pd.DataFrame([[1, 2], [3, 4]], columns=['a','b'])

df2 = pd.DataFrame([[5, 6], [7, 8]], columns=['a','b'])

df1 = df1.append(df2)

# Drop rows with label 0

df1 = df1.drop(0)

print(df1)

o/p:

a b

1 3 4

1 7 8

In the above example, two rows were dropped because those two contain the same label 0.