ITERATION ON DATAFRAME

The behavior of basic iteration over Pandas objects depends on the type. When iterating over a Series, it is regarded as array-like, and basic iteration produces the values. Other data structures, like DataFrame and Panel, follow the **dict-like** convention of iterating over the **keys** of the objects.

To iterate over the rows of the DataFrame, we can use the following functions −

- **iteritems()** − to iterate over the (key,value) pairs
- **iterrows()** − iterate over the rows as (index,series) pairs
- **itertuples()** − iterate over the rows as namedtuples

iteritems() − to iterate over the (key,value) pairs

Iterates over the DataFrame columns, returning a tuple with the column name and the content as a Series.

```
import pandas as pd
import numpy as np
disales={2015:{'Qtr1':34500,'Qtr2':56000,'Qtr3':47000,'Qtr4':49000},\
        2016:{'Qtr1':44500,'Qtr2':461000,'Qtr3':57000,'Qtr4':59000},\
        2017:{'Qtr1':54500,'Qtr2':51000,'Qtr3':47000,'Qtr4':58500}}
df1=pd.DataFrame(disales)
for(col,colseries) in df1.iteritems():
   print("Column Index:",col)
   print("Containing:")
   print(colseries)
```

Example:

```
df = pd.DataFrame({'species': ['mammal', 'mammal', 'fish'],
                   'population': [3948, 4000, 6000]},
                  index=['tiger', 'fox', 'shark'])
print(df)
for index, content in df.iteritems():
    print('label:', index)
    print("******************")
    print('content:', content, sep='\n')
```

iterrows() − iterate over the rows as (index,series) pairs

Iterates over the DataFrame columns, returning a tuple with the column name and the content as a Series.

```
import pandas as pd
import numpy as np
disales={2015:{'Qtr1':34500,'Qtr2':56000,'Qtr3':47000,'Qtr4':49000},\
        2016:{'Qtr1':44500,'Qtr2':461000,'Qtr3':57000,'Qtr4':59000},\
        2017:{'Qtr1':54500,'Qtr2':51000,'Qtr3':47000,'Qtr4':58500}}
df1=pd.DataFrame(disales)
print(df1)
#ITERROWS() FUNCTION
for(row,rowseries) in df1.iterrows():
    print("ROW Index:",row)
    print("Containing:")
```

print(rowseries)

Example:

Method : Using **iterrows()** method of the Dataframe.

```
import pandas package as pd
import pandas as pd

# Define a dictionary containing students data
data = {'Name': ['Ankit', 'Amit', 'Aishwarya', 'Priyanka'],
                'Age': [21, 19, 20, 18],
                'Stream': ['Math', 'Commerce', 'Arts', 'Biology'],
                'Percentage': [88, 92, 95, 70]}

# Convert the dictionary into DataFrame
df = pd.DataFrame(data, columns = ['Name', 'Age', 'Stream', 'Percentage'])

print("Given Dataframe :\n", df)

print("\nIterating over rows using iterrows() method :\n")

# iterate through each row and select
# 'Name' and 'Age' column respectively.
for index, row in df.iterrows():
    print (row["Name"], row["Age"])
```

Method ## : Using **itertuples()** method of the Dataframe.

Syntax:`DataFrame.itertuples(self, index=True, name='Pandas')`

```
# import pandas package as pd
import pandas as pd

# Define a dictionary containing students data
data = {'Name': ['Ankit', 'Amit', 'Aishwarya', 'Priyanka'],
                'Age': [21, 19, 20, 18],
                'Stream': ['Math', 'Commerce', 'Arts', 'Biology'],
                'Percentage': [88, 92, 95, 70]}

# Convert the dictionary into DataFrame
df = pd.DataFrame(data, columns = ['Name', 'Age', 'Stream', 'Percentage'])

print("Given Dataframe :\n", df)

print("\nIterating over rows using itertuples() method :\n")

# iterate through each row and select
# 'Name' and 'Percentage' column respectively.
for row in df.itertuples(index = True, name ='Pandas'):
    print (getattr(row, "Name"), getattr(row, "Percentage"))
```

EXAMPLE:

```
df = pd.DataFrame({'num_legs': [4, 2], 'num_wings': [0, 2]},
                  index=['fox', 'eagle'])
print(df)
for row in df.itertuples():
    print(row)
```

OUTPUT

```
Pandas(Index='fox', num_legs=4, num_wings=0)
Pandas(Index='eagle', num_legs=2, num_wings=2)
```

```
for row in df.itertuples(index=False):
    print(row)
```

OUTPUT

```
Pandas (num_legs=4, num_wings=0)
Pandas (num_legs=2, num_wings=2)
```

```
for row in df.itertuples(name='Animal'):
    print(row)
```

OUTPUT

```
Animal (Index='fox', num_legs=4, num_wings=0)
Animal (Index='eagle', num_legs=2, num_wings=2)
```

Method #1 : Using **index** attribute of the Dataframe .

```
# import pandas package as pd
import pandas as pd

# Define a dictionary containing students data
data = {'Name': ['Ankit', 'Amit', 'Aishwarya', 'Priyanka'],
                'Age': [21, 19, 20, 18],
                'Stream': ['Math', 'Commerce', 'Arts', 'Biology'],
                'Percentage': [88, 92, 95, 70]}

# Convert the dictionary into DataFrame
df = pd.DataFrame(data, columns = ['Name', 'Age', 'Stream', 'Percentage'])

print("Given Dataframe :\n", df)

print("\nIterating over rows using index attribute :\n")

# iterate through each row and select
# 'Name' and 'Stream' column respectively.
for ind in df.index:
     print(df['Name'][ind], df['Stream'][ind])
```

Method #2 : Using **loc[]** function of the Dataframe.

The **loc() function** is used to access a group of rows and columns by label(s)

```
# import pandas package as pd
import pandas as pd

# Define a dictionary containing students data
data = {'Name': ['Ankit', 'Amit', 'Aishwarya', 'Priyanka'],
                'Age': [21, 19, 20, 18],
                'Stream': ['Math', 'Commerce', 'Arts', 'Biology'],
                'Percentage': [88, 92, 95, 70]}

# Convert the dictionary into DataFrame
df = pd.DataFrame(data, columns = ['Name', 'Age', 'Stream', 'Percentage'])

print("Given Dataframe :\n", df)
```

```
print("\nIterating over rows using loc function :\n")

# iterate through each row and select
# 'Name' and 'Age' column respectively.
for i in range(len(df)) :
  print(df.loc[i, "Name"], df.loc[i, "Age"])
```

Method #3 : Using **iloc[]** function of the DataFrame.
iloc returns a Pandas Series when one row is selected, and a Pandas DataFrame when multiple rows are selected, or if any column in full is selected.

```
# import pandas package as pd
import pandas as pd

# Define a dictionary containing students data
data = {'Name': ['Ankit', 'Amit', 'Aishwarya', 'Priyanka'],
                'Age': [21, 19, 20, 18],
                'Stream': ['Math', 'Commerce', 'Arts', 'Biology'],
                'Percentage': [88, 92, 95, 70]}

# Convert the dictionary into DataFrame
df = pd.DataFrame(data, columns = ['Name', 'Age', 'Stream', 'Percentage'])

print("Given Dataframe :\n", df)

print("\nIterating over rows using iloc function :\n")

# iterate through each row and select
# 0th and 2nd index column respectively.
for i in range(len(df)) :
  print(df.iloc[i, 0], df.iloc[i, 2])
```

x----------x----------x