5. Giới thiệu dataframe

Ý tưởng cơ bản của một dataframe dựa trên bảng tính. Chúng ta có thể thấy cấu trúc dữ liệu của một dataframe như dạng bảng và bảng tính. Nó chứa một bộ các cột. Mỗi cột bao gồm một loại dữ liệu duy nhất, nhưng các cột khác nhau có thể có các loại khác nhau, ví dụ: Cột đầu tiên có thể bao gồm số nguyên, trong khi cột thứ hai bao gồm các giá trị boolean etc.

Dataframe có một chỉ mục hàng và cột; Nó giống như một dictionary với key là tên cột còn giá trị tương ứng là dữ liệu kiểu Series và chúng có chung chỉ mục hàng.

pandas cung cấp cho ta phương thức **pandas.DataFrame(data, index, columns, dtype, copy)** để tạo ra một dataframe từ nhiều cấu trúc dữ liệu khác nhau một cách linh hoạt.

Trong đó:

‘data’ sẽ nhận giá trị từ nhiều kiểu khác nhau như ndarray, series, map, lists, dict, constants and also another DataFrame.

‘index’ là nhãn chỉ mục hàng của dataframe.

‘columns’ nhãn chỉ mục cột của dataframe.

‘dtype’ là kiểu dữ liệu cho mỗi cột.

‘copy’ nhận giá trị True/False để chỉ rõ dữ liệu có được copy sang vùng nhớ mới không, mặc định sẽ nhận giá trị False.

Chú ý: Có thể kiểm tra dtype của một cột với **df.nameOfColumn.dtype**

1. Tạo một dataframe rỗng.

```
>>> import pandas as pd
>>> df = pd.DataFrame()
>>> df
Empty DataFrame
Columns: []
Index: []
>>>
```

2. Tạo một dataframe từ một list

```
>>> import pandas as pd
>>> names = ['MIT',"stanford","HUST"]
>>> df = pd.DataFrame(names)
>>> df
         0
0      MIT
1  stanford
2     HUST
>>> names_rank = [['MIT',1],["stanford",2],["HUST",2000]]
>>> df = pd.DataFrame(names_rank)
>>> df
         0     1
0      MIT     1
1  stanford     2
2     HUST  2000
>>>
```

3. Tạo dataframe từ dictionary của list hoặc ndarrays.

```
>>> crimes_rates = {"year":[1960,1961,1962,1963,1964],"Population":[179323175,182992000,185771000,188483000,191141000],"Total":[3384200,3488000,3752200,4109500,4564600],"Viol
>>> crimes_dataframe = pd.DataFrame(crimes_rates)
>>> crimes_dataframe
   Population    Total         Violent  year
0   179323175  3384200   288460  1960
1   182992000  3488000   289390  1961
2   185771000  3752200   301510  1962
3   188483000  4109500   316970  1963
4   191141000  4564600   364220  1964
>>>
```

Tạo dataframe từ list của dictionary

Ví dụ 1:

```
>>> data = [{'MIT': 5000, 'Stanford': 4500, "HUST":6000},{'MIT': 1, 'Stanford': 2, "HUST":2000}]
>>> df = pd.DataFrame(data, index=['NumOfStudents', "ranking"])
>>> df
                    HUST   MIT  Stanford
NumOfStudents  6000  5000      4500
ranking             2000     1        2
>>> df.HUST.dtype
dtype('int64')
```

Ví dụ 2.

```
>>> data = [{'MIT': 5000, 'Stanford': 4500, "HUST":6000},{'MIT': 'first', 'Stanford': 'second', "HUST":'third'}]
>>> df = pd.DataFrame(data, index=['NumOfStudents', "ranking"])
>>> df
                     HUST    MIT Stanford
NumOfStudents    6000   5000     4500
ranking               third  first   second
>>> df.HUST.dtype
dtype('O')
```

Có thể thấy sự khác biệt về kiểu dữ liệu của cột HUST trong ví dụ 2 là kiểu object.

4. Tạo dataframe từ dict của các series

```
>>> data = {"one": pd.Series([1,23,45],index = [1,2,3]), "two":pd.Series([1000,2400,1132,3434],index = [1,2,3,4])}
>>> df = pd.DataFrame(data)
>>> df
   one   two
1   1.0  1000
2  23.0  2400
3  45.0  1132
4   NaN  3434
>>>
```

Trong ví dụ trên 2 Pandas Series có chỉ mục khác nhau, nhưng chúng được merge với nhau, giá trị thiếu sẽ thay thế bằng NaN.

Tạo dataframe từ một dataframe khác, chú ý tham số copy. Mặc định copy = False

Ví dụ dưới đây đã tạo ra shared_df cùng chia sẻ bộ nhớ với df.

```
>>> shared_df = pd.DataFrame(df)
>>> shared_df['three'] = pd.Series(["MIT","Stanford","HUST"])
>>> df
   one   two     three
1   1.0  1000  Stanford
2  23.0  2400      HUST
3  45.0  1132       NaN
4   NaN  3434       NaN
>>> shared_df
   one   two     three
1   1.0  1000  Stanford
2  23.0  2400      HUST
3  45.0  1132       NaN
4   NaN  3434       NaN
>>>
```

Sau khi thêm cột ‘three’ vào DataFrame shared_df. Index sẽ tuân theo DataFrame gốc, shared_df lấy index từ 1. Dẫn đến giá trị ‘MIT’ với index 0 không được thêm vào.

Ví dụ sau khi copy = True, new_df là độc lập với df.

```
>>> data = [{'MIT': 5000, 'Stanford': 4500, "HUST":6000},{'MIT': 1, 'Stanford': 2, "HUST":2000}]
```

```
>>> data = {"one": pd.Series([1,23,45],index = [1,2,3]), "two":pd.Series([1000,2400,1132,3434],index = [1,2,3,4])}
>>> df = pd.DataFrame(data)
>>> df
   one   two
1   1.0  1000
2  23.0  2400
3  45.0  1132
4   NaN  3434
>>> new_df = pd.DataFrame(df,copy= True)
>>> new_df['three'] = pd.Series(["MIT","Stanford","HUST"])
>>> df
   one   two
1   1.0  1000
2  23.0  2400
3  45.0  1132
4   NaN  3434
>>> new_df
   one   two     three
1   1.0  1000  Stanford
2  23.0  2400      HUST
3  45.0  1132       NaN
4   NaN  3434       NaN
>>>
```

Kết Luận

Như vậy một Dataframe có cấu trúc dạng bảng, 2 chiều và có thể được khởi tạo với cú pháp pandas.DataFrame(). Có nhiều cách để tạo ra một Dataframe đã được chúng tôi giới thiệu với từng ví dụ cụ thể như tạo một dataframe từ một list, từ dictionary của list, từ list của dictionary, từ dictionary của series hay từ chính một dataframe khác. Trong bài tiếp theo một số thao tác cơ bản trên dataframe sẽ được chúng tôi đề cập tới.