



Data Application Lab

Pandas Basis

```
In [1]: # import necessary package
import pandas as pd
import numpy as np
```

```
In [2]: # generate a dataset and convert to pandas dataframe

data = {'name': ['Jack', 'Annie', 'Bob', 'Jake', 'Jasper'],
        'year': [2012, 2012, 2013, 2014, 2014],
        'grades': [4, 20, 15, 7, 24],
        'coverage': [25, 94, 57, 62, 70]}

df = pd.DataFrame(data, index = ['Cochic', 'Pima', 'Santa Cruz', 'Maricopa', 'Yuma'])
df
```

```
Out[2]:
```

	coverage	grades	name	year
Cochic	25	4	Jack	2012
Pima	94	20	Annie	2012
Santa Cruz	57	15	Bob	2013
Maricopa	62	7	Jake	2014
Yuma	70	24	Jasper	2014

1. Show the first three rows and last two rows.

```
In [3]: df.head(3)
```

```
Out[3]:
```

	coverage	grades	name	year
Cochic	25	4	Jack	2012
Pima	94	20	Annie	2012
Santa Cruz	57	15	Bob	2013

```
In [4]: df.tail(2)
```

```
Out[4]:
```

	coverage	grades	name	year
Maricopa	62	7	Jake	2014
Yuma	70	24	Jasper	2014

2. Generate a new dataframe including county, year and reports from above datafram (county and reports can be NaN)

```
In [6]: dfreordered = pd.DataFrame(data, columns=['county', 'year', 'reports'])
dfreordered
```

```
Out[6]:
```

	county	year	reports
0	NaN	2012	NaN
1	NaN	2012	NaN
2	NaN	2013	NaN
3	NaN	2014	NaN
4	NaN	2014	NaN

3. Add a new column attribute called "New", which is NaN.

```
In [7]: dfreordered['New'] = pd.Series([])
dfreordered
```

```
Out[7]:
```

	county	year	reports	New
0	NaN	2012	NaN	NaN
1	NaN	2012	NaN	NaN
2	NaN	2013	NaN	NaN
3	NaN	2014	NaN	NaN
4	NaN	2014	NaN	NaN

4. Delete the column "New".

```
In [8]: # hint: use del dfreordered['desired column']
del dfreordered['New']
dfreordered
```

```
Out[8]:
```

	county	year	reports
0	NaN	2012	NaN
1	NaN	2012	NaN
2	NaN	2013	NaN
3	NaN	2014	NaN
4	NaN	2014	NaN

5. Show the index and columns name.

```
In [9]: df.index
Out[9]: Index(['Cochic', 'Pima', 'Santa Cruz', 'Maricopa', 'Yuma'], dtype='object')

In [10]: df.columns
Out[10]: Index(['coverage', 'grades', 'name', 'year'], dtype='object')
```

6. Use df.iloc to select the first row

```
In [11]: df.iloc[0]
Out[11]: coverage      25
          grades       4
          name      Jack
          year     2012
          Name: Cochic, dtype: object
```

```
In [12]: df.loc['Cochic']
Out[12]: coverage      25
          grades       4
          name      Jack
          year     2012
          Name: Cochic, dtype: object
```

7. Slice the column of "coverage" and "grades".

```
In [14]: df[['coverage', 'grades']]
```

Out[14]:

	coverage	grades
Cochic	25	4
Pima	94	20
Santa Cruz	57	15
Maricopa	62	7
Yuma	70	24

8. select subtable of column name and year, row of Pima and Yuma

```
In [15]: df[['name', 'year']].loc[['Pima', 'Yuma']]
```

Out[15]:

	name	year
Pima	Annie	2012
Yuma	Jasper	2014

9. Use df.iloc to select the first two columns

```
In [16]: df.iloc[:, :2]
```

Out[16]:

	coverage	grades
Cochic	25	4
Pima	94	20
Santa Cruz	57	15
Maricopa	62	7
Yuma	70	24

10. Select by condition where grades greater than 10

```
In [17]: df[df['grades']>10]
```

Out[17]:

	coverage	grades	name	year
Pima	94	20	Annie	2012
Santa Cruz	57	15	Bob	2013
Yuma	70	24	Jasper	2014

11. Drop Rows of Cochic and Pima

```
In [18]: df.drop(['Cochic', 'Pima'])
```

Out[18]:

	coverage	grades	name	year
Santa Cruz	57	15	Bob	2013
Maricopa	62	7	Jake	2014
Yuma	70	24	Jasper	2014

12. Use df.drop to drop the column grades.

```
In [19]: df.drop('grades', axis =1)
```

Out[19]:

	coverage	name	year
Cochic	25	Jack	2012
Pima	94	Annie	2012
Santa Cruz	57	Bob	2013
Maricopa	62	Jake	2014
Yuma	70	Jasper	2014

13. Use describe function to show some statistics of column "coverage".

```
In [20]: df['coverage'].describe()
```

```
Out[20]: count      5.000000  
         mean     61.600000  
         std      24.905823  
         min      25.000000  
         25%      57.000000  
         50%      62.000000  
         75%      70.000000  
         max      94.000000  
         Name: coverage, dtype: float64
```

Pandas Fast Visualization

Read the following reading materials:

<https://pandas.pydata.org/pandas-docs/stable/visualization.html>