

▼ pandas

- Panel Data: 금융데이터 계량 분석
- Python Data Analysis Library
- <https://pandas.pydata.org>

```
import warnings
warnings.filterwarnings('ignore')
```

▼ File Upload to Colab

- [Colab 가상환경에 파일 올리기](#)
- Colab 종료 시 파일은 삭제됨
- Local\_Disk to Colab\_Linux\_File\_System
  - [PII.csv](#) & [PII.xlsx](#)
- 업로드된 파일 확인

```
!!ls -l

total 20
-rw-r--r-- 1 root root 723 Mar 12 00:44 PII.csv
-rw-r--r-- 1 root root 9907 Mar 12 00:44 PII.xlsx
drwxr-xr-x 1 root root 4096 Mar 9 18:58 sample_data
```

▼ I. pandas - DataFrame

- [Label Index](#)를 가진 2차원 구조
- 일반적으로 **다양한 데이터 타입의 열들**로 구성

▼ 1) pandas Package [import ~ as](#)

```
import pandas as pd
```

- Version Check

```
pd.__version__

'1.3.5'
```

▼ 2) [.read\\_csv\(\)](#)

- CSV(Comma Separated Values)
- [https://pandas.pydata.org/docs/reference/api/pandas.read\\_csv.html](https://pandas.pydata.org/docs/reference/api/pandas.read_csv.html)

```
DF1 = pd.read_csv('PII.csv')
```

- DataFrame

```
type(DF1)

pandas.core.frame.DataFrame
```

	Name	Gender	Age	Grade	Picture	BloodType	Height	Weight
0	송태섭	남자	21	3	무	B	179.1	63.9
1	최유정	여자	23	1	유	A	177.1	54.9
2	이한나	여자	20	1	무	A	167.9	50.2
3	김소혜	여자	23	3	무	O	176.1	53.5
4	서태웅	남자	24	4	무	B	176.1	79.8
5	정대만	남자	24	2	유	B	175.2	61.7
6	이정환	남자	22	4	무	B	169.1	69.8
7	채소연	여자	22	2	유	AB	169.9	52.7
8	강백호	남자	23	3	무	O	165.5	68.5
9	전소미	여자	22	2	유	O	161.9	52.3
10	변덕규	남자	21	1	무	A	163.2	55.5
11	정채연	여자	22	2	무	B	157.8	44.9
12	권준호	남자	24	4	유	O	166.9	61.7
13	채치수	남자	23	3	무	AB	181.8	85.9
14	윤대협	남자	22	2	유	AB	180.3	76.2
15	김세정	여자	21	1	무	O	155.5	44.9
16	신준섭	남자	23	1	무	A	168.9	62.7

▼ 3) [.to\\_csv\(\)](#)

- CSV 파일로 저장
  - [https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.to\\_csv.html#pandas.DataFrame.to\\_csv](https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.to_csv.html#pandas.DataFrame.to_csv)

```
DF1[0:3].to_csv('myCSV.csv', index = None)
```

- 생성된 CSV 파일 확인

```
!!ls -l
```

```
total 24
-rw-r--r-- 1 root root 171 Mar 12 00:45 myCSV.csv
-rw-r--r-- 1 root root 723 Mar 12 00:44 P11.csv
-rw-r--r-- 1 root root 9907 Mar 12 00:44 P11.xlsx
drwxr-xr-x 1 root root 4096 Mar 9 18:58 sample_data
```

- .read\_csv() 결과 확인

```
pd.read_csv('myCSV.csv')
```

	Name	Gender	Age	Grade	Picture	BloodType	Height	Weight
0	송태섭	남자	21	3	무	B	179.1	63.9
1	최유정	여자	23	1	유	A	177.1	54.9
2	이한나	여자	20	1	무	A	167.9	50.2

#### ▼ 4) .read\_excel()

- [https://pandas.pydata.org/docs/reference/api/pandas.read\\_excel.html](https://pandas.pydata.org/docs/reference/api/pandas.read_excel.html)

```
DF2 = pd.read_excel('P11.xlsx')
```

- DataFrame

```
type(DF2)
```

```
pandas.core.frame.DataFrame
```

```
DF2
```

	Name	Gender	Age	Grade	Picture	BloodType	Height	Weight
0	송태섭	남자	21.0	3.0	무	B	179.1	63.9
1	최유정	여자	23.0	1.0	유	A	177.1	54.9
2	이한나	여자	20.0	1.0	무	A	167.9	50.2
3	김소혜	여자	23.0	3.0	무	O	176.1	53.5
4	서태웅	남자	24.0	4.0	무	B	176.1	79.8
5	정대만	남자	24.0	2.0	유	B	175.2	61.7
6	이정환	남자	22.0	4.0	무	B	169.1	69.8
7	채소연	여자	22.0	2.0	유	AB	169.9	52.7
8	강백호	남자	23.0	3.0	무	O	165.5	68.5
9	전소미	여자	22.0	2.0	유	O	161.9	52.3
10	변덕규	남자	21.0	1.0	무	A	163.2	55.5
11	정채연	여자	22.0	2.0	무	B	157.8	44.9
12	권준호	남자	24.0	4.0	유	O	166.9	61.7
13	채치수	남자	23.0	3.0	무	AB	181.8	85.9
14	윤대협	남자	22.0	2.0	유	AB	180.3	76.2
15	김세정	여자	21.0	1.0	무	O	155.5	44.9
16	신준섭	남자	23.0	1.0	무	A	168.9	62.7

#### ▼ 5) .to\_excel()

- Excel 파일로 저장
  - [https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.to\\_excel.html](https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.to_excel.html)

```
DF2[14:].to_excel('myExcel.xlsx', index = None)
```

- 생성된 Excel 파일 확인

```
!!ls -l
```

```
total 32
-rw-r--r-- 1 root root 171 Mar 12 00:45 myCSV.csv
-rw-r--r-- 1 root root 5177 Mar 12 00:45 myExcel.xlsx
-rw-r--r-- 1 root root 723 Mar 12 00:44 P11.csv
-rw-r--r-- 1 root root 9907 Mar 12 00:44 P11.xlsx
drwxr-xr-x 1 root root 4096 Mar 9 18:58 sample_data
```

- .read\_excel() 결과 확인

```
pd.read_excel('myExcel.xlsx')
```

	Name	Gender	Age	Grade	Picture	BloodType	Height	Weight
0	윤대협	남자	22	2	유	AB	180.3	76.2
1	김세정	여자	21	1	무	O	155.5	44.9
2	신준섭	남자	23	1	무	A	168.9	62.7

#### ▼ 6) DataFrame Information

- [.info\(\)](#)

```
DF1.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 17 entries, 0 to 16
Data columns (total 8 columns):
#   Column      Non-Null Count  Dtype
---  ---
0   Name        17 non-null    object
1   Gender      17 non-null    object
2   Age         17 non-null    int64
3   Grade       17 non-null    int64
4   Picture     17 non-null    object
5   BloodType   17 non-null    object
6   Height      17 non-null    float64
7   Weight      17 non-null    float64
dtypes: float64(2), int64(2), object(4)
memory usage: 1.2+ KB
```

- [.index - 행\\_정보](#)

```
DF1.index

RangeIndex(start=0, stop=17, step=1)
```

- [.columns - 열\\_정보](#)

```
DF1.columns

Index(['Name', 'Gender', 'Age', 'Grade', 'Picture', 'BloodType', 'Height',
      'Weight'],
      dtype='object')
```

- [.values - Array](#)

```
DF1.values

array([[ '송태섭', '남자', 21, 3, '무', 'B', 179.1, 63.9],
       [ '최유정', '여자', 23, 1, '유', 'A', 177.1, 54.9],
       [ '이한나', '여자', 20, 1, '무', 'A', 167.9, 50.2],
       [ '김소혜', '여자', 23, 3, '무', 'O', 176.1, 53.5],
       [ '서태웅', '남자', 24, 4, '무', 'B', 176.1, 79.8],
       [ '정대만', '남자', 24, 2, '유', 'B', 175.2, 61.7],
       [ '이정환', '남자', 22, 4, '무', 'B', 169.1, 69.8],
       [ '채소연', '여자', 22, 2, '유', 'AB', 169.9, 52.7],
       [ '강백호', '남자', 23, 3, '무', 'O', 165.5, 68.5],
       [ '전소미', '여자', 22, 2, '유', 'O', 161.9, 52.3],
       [ '변덕규', '남자', 21, 1, '무', 'A', 163.2, 55.5],
       [ '정채연', '여자', 22, 2, '무', 'B', 157.8, 44.9],
       [ '권준호', '남자', 24, 4, '유', 'O', 166.9, 61.7],
       [ '채치수', '남자', 23, 3, '무', 'AB', 181.8, 85.9],
       [ '윤대협', '남자', 22, 2, '유', 'AB', 180.3, 76.2],
       [ '김세정', '여자', 21, 1, '무', 'O', 155.5, 44.9],
       [ '신준섭', '남자', 23, 1, '무', 'A', 168.9, 62.7]], dtype=object)
```

▼ 7) Function

- [.head\(\)](#)
  - 위에서 5행 출력

	Name	Gender	Age	Grade	Picture	BloodType	Height	Weight
0	송태섭	남자	21	3	무	B	179.1	63.9
1	최유정	여자	23	1	유	A	177.1	54.9
2	이한나	여자	20	1	무	A	167.9	50.2
3	김소혜	여자	23	3	무	O	176.1	53.5
4	서태웅	남자	24	4	무	B	176.1	79.8

- [.tail\(\)](#)
  - 밑에서 5행 출력

	Name	Gender	Age	Grade	Picture	BloodType	Height	Weight
12	권준호	남자	24	4	유	O	166.9	61.7
13	채치수	남자	23	3	무	AB	181.8	85.9
14	윤대협	남자	22	2	유	AB	180.3	76.2
15	김세정	여자	21	1	무	O	155.5	44.9
16	신준섭	남자	23	1	무	A	168.9	62.7

- [.sort\\_values\(\)- 오름차순](#)

	Name	Gender	Age	Grade	Picture	BloodType	Height	Weight
15	김세정	여자	21	1	무	O	155.5	44.9
11	정채연	여자	22	2	무	B	157.8	44.9
9	전소미	여자	22	2	유	O	161.9	52.3
10	변덕규	남자	21	1	무	A	163.2	55.5
8	강백호	남자	23	3	무	O	165.5	68.5

- [.sort\\_values\( ascending = False \) - 내림차순](#)

```
DF1.sort_values(by = 'Height', ascending = False).head()
```

	Name	Gender	Age	Grade	Picture	BloodType	Height	Weight
13	채치수	남자	23	3	무	AB	181.8	85.9
14	윤대협	남자	22	2	유	AB	180.3	76.2
0	송태섭	남자	21	3	무	B	179.1	63.9
1	최유정	여자	23	1	유	A	177.1	54.9
3	김소혜	여자	23	3	무	O	176.1	53.5

- `.describe()`
  - 숫자 데이터 통계량

```
DF1.describe()
```

	Age	Grade	Height	Weight
count	17.000000	17.000000	17.000000	17.000000
mean	22.352941	2.294118	170.135294	61.123529
std	1.169464	1.104802	7.853896	11.867894
min	20.000000	1.000000	155.500000	44.900000
25%	22.000000	1.000000	165.500000	52.700000
50%	22.000000	2.000000	169.100000	61.700000
75%	23.000000	3.000000	176.100000	68.500000
max	24.000000	4.000000	181.800000	85.900000

- `.mean()`

```
DF1.mean()
```

```
Age      22.352941
Grade     2.294118
Height   170.135294
Weight   61.123529
dtype: float64
```

- `.median()`

```
DF1.median()
```

```
Age      22.0
Grade     2.0
Height   169.1
Weight   61.7
dtype: float64
```

- `.var()`

```
DF1.var()
```

```
Age      1.367647
Grade     1.220588
Height    61.683676
Weight   140.846912
dtype: float64
```

- `.std()`

```
DF1.std()
```

```
Age      1.169464
Grade     1.104802
Height    7.853896
Weight   11.867894
dtype: float64
```

## ▼ 8) Indexing & Slicing

- Lable(Index) 변경

```
LABEL = ['No_1', 'No_2', 'No_3', 'No_4', 'No_5', 'No_6', 'No_7', 'No_8', 'No_9', 'No_10', 'No_11', 'No_12', 'No_13', 'No_14', 'No_15', 'No_16', 'No_17']
```

```
DF1.index = LABEL
```

```
DF1
```

	Name	Gender	Age	Grade	Picture	BloodType	Height	Weight
No_1	송태섭	남자	21	3	무	B	179.1	63.9
No_2	최유정	여자	23	1	유	A	177.1	54.9
No_3	이한나	여자	20	1	무	A	167.9	50.2

▼ (1) loc(Location)

- Label 값 기준

No_7	이정화	남자	22	4	무	B	169.1	69.8
------	-----	----	----	---	---	---	-------	------

- loc - indexing

- .loc[행\_label, 열\_label]

No_10	전수민	여자	22	3	유	O	161.0	52.2
-------	-----	----	----	---	---	---	-------	------

DF1.loc['No\_3', 'Age']

20

- loc - Slicing

- .loc[행\_label\_시작:행\_label\_끝, 열\_label\_시작:열\_label\_끝]
  - '행\_label\_시작' 이상 '행\_label\_끝' 이하
  - '열\_label\_시작' 이상 '열\_label\_끝' 이하

No_17	신순섭	남자	23	1	무	A	168.9	62.7
-------	-----	----	----	---	---	---	-------	------

DF1.loc['No\_3':'No\_6', 'Age':'BloodType']

	Age	Grade	Picture	BloodType
No_3	20	1	무	A
No_4	23	3	무	O
No_5	24	4	무	B
No_6	24	2	유	B

▼ (2) iloc(Integer Location)

- Index 값 기준

- iloc - indexing

DF1.iloc[2, 2]

20

- iloc - Slicing

- .iloc[행\_index\_시작:행\_index\_끝, 열\_index\_시작:열\_index\_끝]
  - '행\_index\_시작' 이상 '행\_index\_끝' 미만
  - '열\_index\_시작' 이상 '열\_index\_끝' 미만

DF1.iloc[2:5, 2:5]

	Age	Grade	Picture
No_3	20	1	무
No_4	23	3	무
No_5	24	4	무

▼ II. pandas - Series

- Label Index를 가지는 1차원 구조
- 일반적으로 동일한 데이터 타입으로 구성

▼ 1) Series from DataFrame

- 'Height' Column Series - 1

- DataFrame\_Name['Column\_Name']

DF1['Height']

No\_1 179.1  
No\_2 177.1  
No\_3 167.9  
No\_4 176.1  
No\_5 176.1  
No\_6 175.2  
No\_7 169.1  
No\_8 169.9  
No\_9 165.5  
No\_10 161.9  
No\_11 163.2  
No\_12 157.8  
No\_13 166.9  
No\_14 181.8  
No\_15 180.3  
No\_16 155.5  
No\_17 168.9  
Name: Height, dtype: float64

- 'Height' Column Series - 2

- DataFrame\_Name.Column\_Name

DF1.Height

No\_1 179.1  
No\_2 177.1  
No\_3 167.9

```
No_4      176.1
No_5      176.1
No_6      175.2
No_7      169.1
No_8      169.9
No_9      165.5
No_10     161.9
No_11     163.2
No_12     157.8
No_13     166.9
No_14     181.8
No_15     180.3
No_16     155.5
No_17     168.9
Name: Height, dtype: float64
```

- `type()` - Series

```
type(DF1['Height'])
```

```
pandas.core.series.Series
```

- 'Height' & 'Age' Columns
  - DataFrame Structure

```
DF1[['Height', 'Age']]
```

	Height	Age
<b>No_1</b>	179.1	21
<b>No_2</b>	177.1	23
<b>No_3</b>	167.9	20
<b>No_4</b>	176.1	23
<b>No_5</b>	176.1	24
<b>No_6</b>	175.2	24
<b>No_7</b>	169.1	22
<b>No_8</b>	169.9	22
<b>No_9</b>	165.5	23
<b>No_10</b>	161.9	22
<b>No_11</b>	163.2	21
<b>No_12</b>	157.8	22
<b>No_13</b>	166.9	24
<b>No_14</b>	181.8	23
<b>No_15</b>	180.3	22
<b>No_16</b>	155.5	21
<b>No_17</b>	168.9	23

- `type()` - DataFrame

```
type(DF1[['Height', 'Age']])
```

```
pandas.core.frame.DataFrame
```

## 2) Function

- `.count()`

```
DF1['Height'].count()
```

```
17
```

- `.sum()`

```
DF1['Height'].sum()
```

```
2892.2999999999997
```

- `.mean()`

```
DF1['Height'].mean()
```

```
170.13529411764705
```

- `.median()`

```
DF1['Height'].median()
```

```
169.1
```

- `.var()`

```
DF1['Height'].var()
```

```
61.68367647058823
```

- `.std()`

```
DF1['Height'].std()
```

```
7.853895623866428
```

- `.min()`

DF1.Height.min()

155.5

- .max()

DF1.Height.max()

181.8

- .idxmin()
  - Series [최솟값의 인덱스 주소](#)를 반환

DF1.Height.idxmin()

'No\_16'

- .idxmax()
  - Series [최댓값의 인덱스 주소](#)를 반환

DF1.Height.idxmax()

'No\_14'

- .sort\_values()- [오름차순](#)

DF1.Height.sort\_values(ascending = True).head()

```
No_16    155.5
No_12    157.8
No_10    161.9
No_11    163.2
No_9     165.5
Name: Height, dtype: float64
```

- .sort\_values( ascending = False ) - [내림차순](#)

DF1.Height.sort\_values(ascending = False).head()

```
No_14    181.8
No_15    180.3
No_1     179.1
No_2     177.1
No_4     176.1
Name: Height, dtype: float64
```

### ▼ 3) Indexing & Slicing

#### ▼ (1) loc(Location)

- Label 값 기준
- loc - indexing

DF1.Height.loc[ 'No\_6' ]

175.2

- loc - [Slicing](#)
  - .loc[label\_시작, label\_끝]
  - 'label\_시작' 이상 'label\_끝' [이하](#)

DF1.Height.loc[ 'No\_6': 'No\_10' ]

```
No_6     175.2
No_7     169.1
No_8     169.9
No_9     165.5
No_10    161.9
Name: Height, dtype: float64
```

#### ▼ (2) iloc(Integer Location)

- Index 값 기준

DF1.Height.iloc[5]

175.2

- .iloc 생략 가능

DF1.Height[5]

175.2

- iloc - [Slicing](#)
  - .iloc[index\_시작, index\_끝]
  - 'index\_시작' 이상 'index\_끝' [미만](#)

DF1.Height.iloc[5:10]

```
No_6     175.2
No_7     169.1
No_8     169.9
No_9     165.5
```

No\_10 161.9  
Name: Height, dtype: float64

- .iloc 생략 가능

DF1.Height[5:10]

```
No_6      175.2
No_7      169.1
No_8      169.9
No_9      165.5
No_10     161.9
Name: Height, dtype: float64
```

### III. pandas - Visualization

- <https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.plot.html>

#### 1) 선 그래프

- kind = 'line'
- style: '-', '-', 's-', 'o-', '^'
- linewidth: 숫자
- color: 16진수 RGB
- figsize: inch

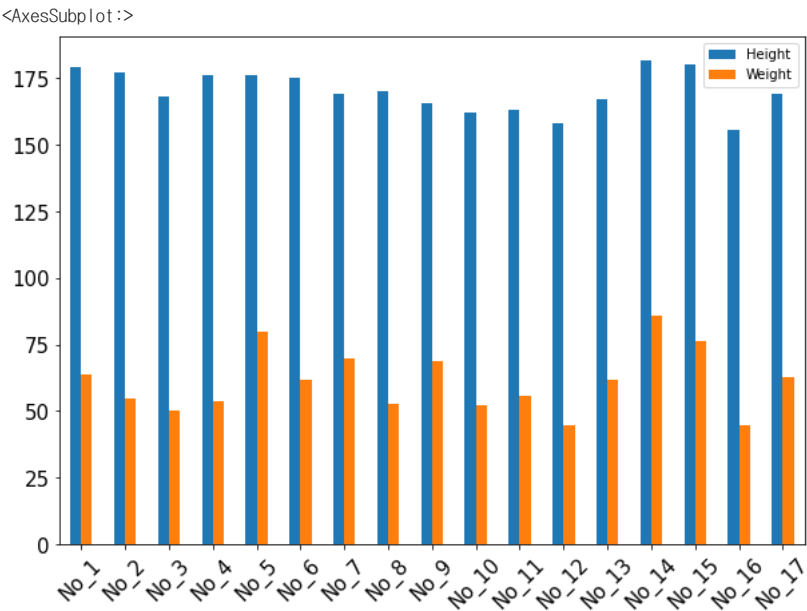
```
DF1[['Height']].plot(kind = 'line',
                    style = '-',
                    linewidth = 2,
                    color = '#FF0000',
                    grid = True,
                    figsize = (10, 7))
```



#### 2) 막대 그래프

- kind = 'bar'
- width: 0~1
- rot: 0~360
- fontsize: 숫자

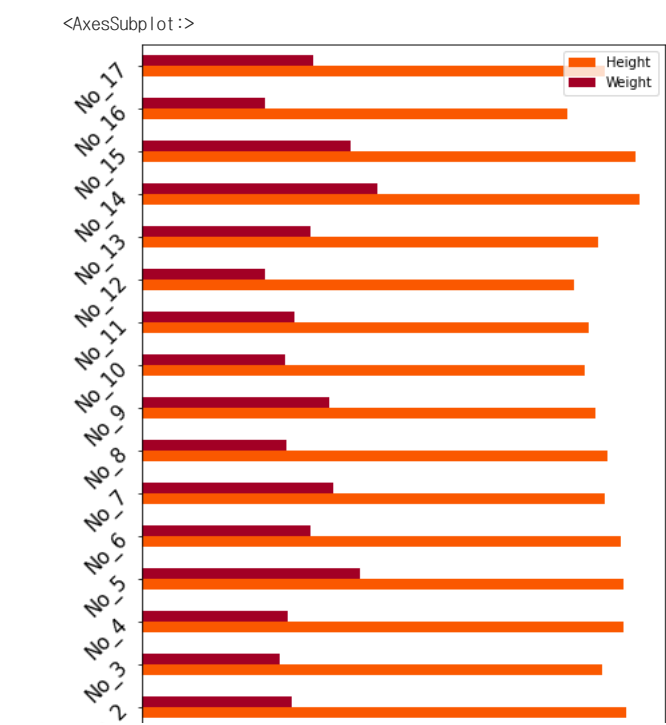
```
DF1[['Height', 'Weight']].plot(kind = 'bar',
                              width = 0.5,
                              rot = 45,
                              fontsize = 15,
                              figsize = (10, 7))
```



- kind = 'barh'

```
DF1[['Height', 'Weight']].plot(kind = 'barh',
                              rot = 45,
                              color = ['#FA5800', '#A20025'],
                              fontsize = 15,
                              figsize = (7, 10))
```

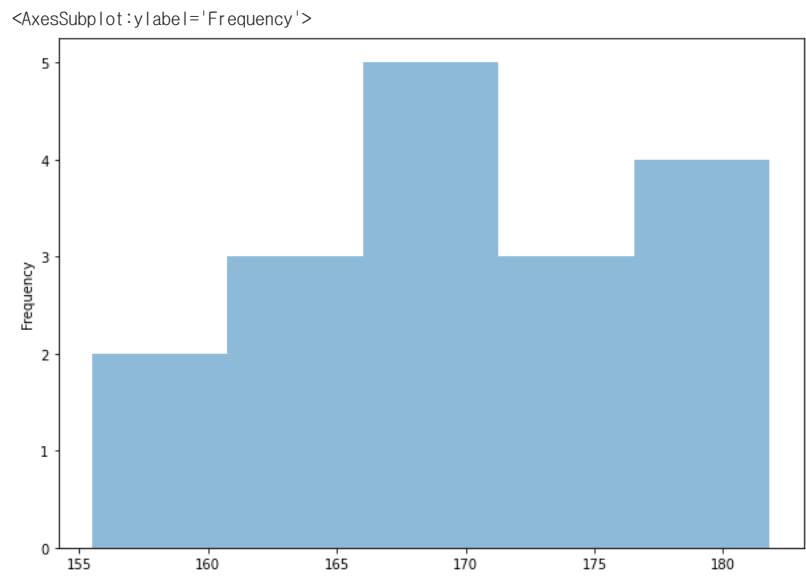




### 3) 히스토그램

- `kind = 'hist'`
- bins: 구간 개수
- alpha: 0~1

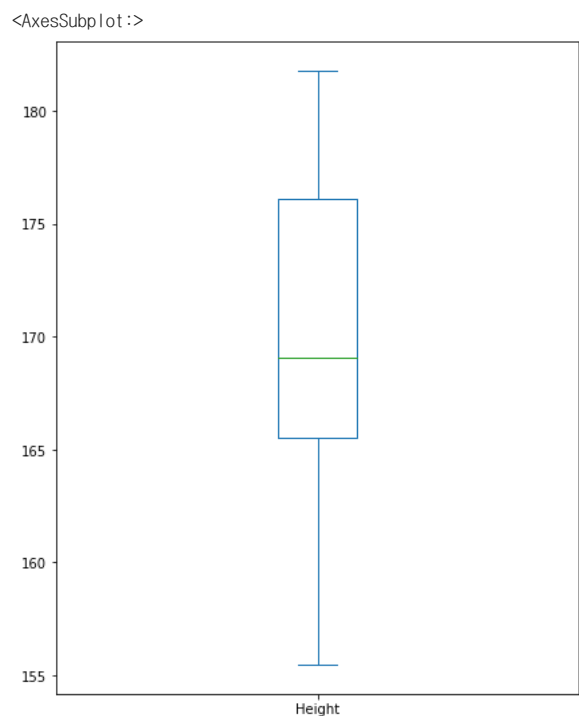
```
DF1['Height'].plot(kind = 'hist',  
                  bins = 5,  
                  alpha = 0.5,  
                  figsize = (10, 7))
```



### 4) 상자 그래프

- `kind = 'box'`

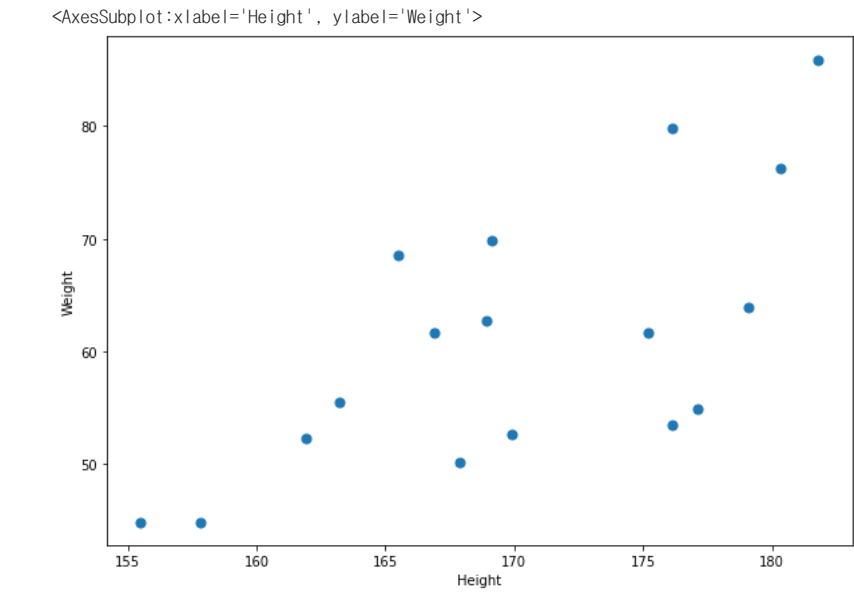
```
DF1['Height'].plot(kind = 'box',  
                  figsize = (7, 9))
```



### 5) 산점도

- `kind = 'scatter'`
- s: 숫자

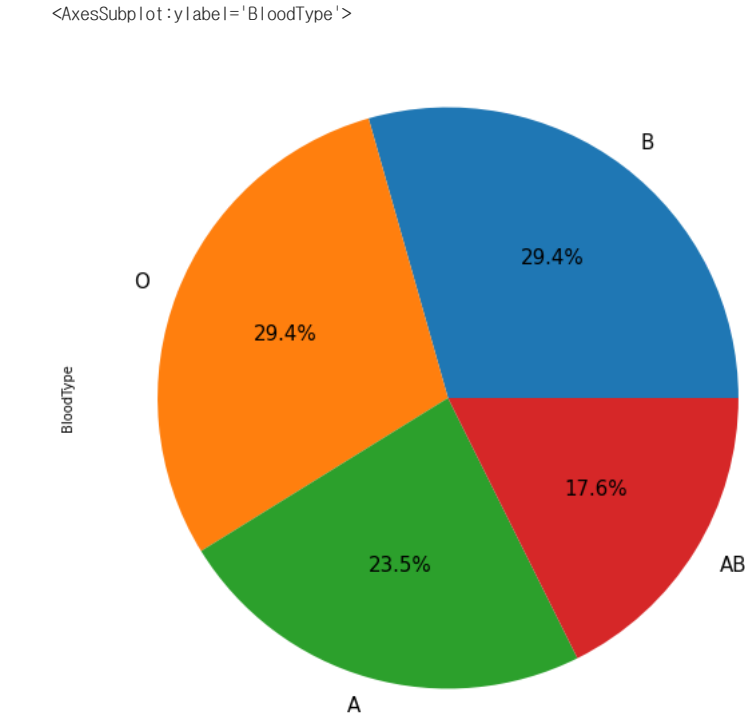
```
DF1[['Height', 'Weight']].plot(kind = 'scatter',  
                              x = 'Height',  
                              y = 'Weight',  
                              s = 50,  
                              figsize = (10, 7))
```



6) 파이 그래프

- kind = 'pie'

```
DF1.BloodType.value_counts().plot(kind = 'pie',
                                   autopct = '%.1f%%',
                                   fontsize = 15,
                                   figsize = (10, 10))
```



#  
#  
#

The End

#  
#  
#