

서울사이버대학교 데이터리터러시 12주차 실습 코드

pandas의 1차원 자료형 - Series

- 1차원적인 데이터 형식
- 파이썬의 딕셔너리와 유사함

```
In [1]: import pandas as pd
```

```
In [2]: a = {'a': 10, 'b': 20, 'c': 30, 'd': 40, 'e': 50}
```

```
In [3]: a
```

```
Out[3]: {'a': 10, 'b': 20, 'c': 30, 'd': 40, 'e': 50}
```

```
In [4]: type(a)
```

```
Out[4]: dict
```

```
In [5]: b = pd.Series(a)
```

```
In [6]: b
```

```
Out[6]: a    10  
       b    20  
       c    30  
       d    40  
       e    50  
       dtype: int64
```

```
In [7]: type(b)
```

```
Out[7]: pandas.core.series.Series
```

```
In [8]: b.shape
```

```
Out[8]: (5,)
```

```
In [9]: import numpy as np
```

```
In [10]: aa = np.arange(3, 10)
```

```
In [11]: aa
```

```
Out[11]: array([3, 4, 5, 6, 7, 8, 9])
```

```
In [12]: bb = pd.Series(aa)
```

```
In [13]: bb
```

```
Out[13]: 0    3  
       1    4  
       2    5  
       3    6  
       4    7  
       5    8
```

```
6    9
dtype: int32
```

```
In [14]: type(bb)
```

```
Out[14]: pandas.core.series.Series
```

```
In [15]: bb.shape
```

```
Out[15]: (7,)
```

```
In [16]: a
```

```
Out[16]: {'a': 10, 'b': 20, 'c': 30, 'd': 40, 'e': 50}
```

```
In [17]: a.keys()
```

```
Out[17]: dict_keys(['a', 'b', 'c', 'd', 'e'])
```

```
In [18]: a.values()
```

```
Out[18]: dict_values([10, 20, 30, 40, 50])
```

```
In [19]: b
```

```
Out[19]: a    10
         b    20
         c    30
         d    40
         e    50
         dtype: int64
```

```
In [20]: b.index
```

```
Out[20]: Index(['a', 'b', 'c', 'd', 'e'], dtype='object')
```

```
In [22]: b.values
```

```
Out[22]: array([10, 20, 30, 40, 50], dtype=int64)
```

```
In [23]: c = [10, 20, 30, 40, 50]
```

```
In [24]: d = pd.Series(c)
```

```
In [25]: d
```

```
Out[25]: 0    10
         1    20
         2    30
         3    40
         4    50
         dtype: int64
```

```
In [26]: type(d)
```

```
Out[26]: pandas.core.series.Series
```

```
In [27]: d.shape
```

```
Out[27]: (5,)
```

```
In [28]: d.index
```

```
Out[28]: RangeIndex(start=0, stop=5, step=1)
```

```
In [29]: d.values
```

```
Out[29]: array([10, 20, 30, 40, 50], dtype=int64)
```

```
In [30]: e = ['a', 'b', 'c', 'd', 'e']
```

```
In [31]: f = [10, 20, 30, 40, 50]
```

```
In [32]: g = pd.Series(f, index = e)
```

```
In [33]: g
```

```
Out[33]: a    10  
        b    20  
        c    30  
        d    40  
        e    50  
        dtype: int64
```

```
In [34]: g['a']
```

```
Out[34]: 10
```

```
In [35]: g[0]
```

```
Out[35]: 10
```

```
In [36]: g['a':'d']
```

```
Out[36]: a    10  
        b    20  
        c    30  
        d    40  
        dtype: int64
```

```
In [37]: g[0:4]
```

```
Out[37]: a    10  
        b    20  
        c    30  
        d    40  
        dtype: int64
```

pandas의 2차원 자료형 - DataFrame 1

- 2차원적인 데이터 형식
- 행렬 구조를 가짐

```
In [1]: import pandas as pd
```

```
In [2]: a = {'a': [1, 2, 3], 'b': [1, 3, 5], 'c': [2, 4, 6], 'd': [3, 6, 9]}
```

```
In [3]: a
```

```
Out[3]: {'a': [1, 2, 3], 'b': [1, 3, 5], 'c': [2, 4, 6], 'd': [3, 6, 9]}
```

```
In [4]: df1 = pd.DataFrame(a)
```

```
In [5]: df1
```

```
Out[5]:
```

	a	b	c	d
0	1	1	2	3
1	2	3	4	6
2	3	5	6	9

```
In [6]: type(df1)
```

```
Out[6]: pandas.core.frame.DataFrame
```

```
In [7]: df1.shape
```

```
Out[7]: (3, 4)
```

```
In [8]: df1.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3 entries, 0 to 2
Data columns (total 4 columns):
#   Column  Non-Null Count  Dtype
---  ---
0    a         3 non-null      int64
1    b         3 non-null      int64
2    c         3 non-null      int64
3    d         3 non-null      int64
dtypes: int64(4)
memory usage: 224.0 bytes
```

```
In [9]: import numpy as np
```

```
In [10]: b = np.arange(12).reshape(3, 4)
```

```
In [11]: b
```

```
Out[11]: array([[ 0,  1,  2,  3],
                [ 4,  5,  6,  7],
                [ 8,  9, 10, 11]])
```

```
In [12]: df2 = pd.DataFrame(b)
```

```
In [13]: df2
```

```
Out[13]:
```

	0	1	2	3
0	0	1	2	3
1	4	5	6	7
2	8	9	10	11

```
In [14]: type(df2)
```

```
Out[14]: pandas.core.frame.DataFrame
```

```
In [15]: df2.shape
```

```
Out[15]: (3, 4)
```

```
In [16]: df2.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3 entries, 0 to 2
Data columns (total 4 columns):
#   Column  Non-Null Count  Dtype
---  -
0    0      3 non-null      int32
1    1      3 non-null      int32
2    2      3 non-null      int32
3    3      3 non-null      int32
dtypes: int32(4)
memory usage: 176.0 bytes
```

```
In [17]: df = pd.DataFrame([[1, 20, '서울'], [2, 25, '대전']],
                           index = ['홍길동', '홍길순'],
                           columns = ['순서', '나이', '지역'])
```

```
In [18]: df
```

```
Out[18]:
```

	순서	나이	지역
홍길동	1	20	서울
홍길순	2	25	대전

```
In [19]: type(df)
```

```
Out[19]: pandas.core.frame.DataFrame
```

```
In [20]: df.shape
```

```
Out[20]: (2, 3)
```

```
In [21]: df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 2 entries, 홍길동 to 홍길순
Data columns (total 3 columns):
#   Column  Non-Null Count  Dtype
---  -
0   순서    2 non-null      int64
1   나이    2 non-null      int64
2   지역    2 non-null      object
```

```
dtypes: int64(2), object(1)
memory usage: 64.0+ bytes
```

```
In [22]: df.index
```

```
Out[22]: Index(['홍길동', '홍길순'], dtype='object')
```

```
In [23]: df.columns
```

```
Out[23]: Index(['순서', '나이', '지역'], dtype='object')
```

```
In [24]: df.values
```

```
Out[24]: array([[1, 20, '서울'],
                [2, 25, '대전']], dtype=object)
```

```
In [25]: df.index = ['사람 1', '사람 2']
```

```
In [26]: df
```

```
Out[26]:
```

	순서	나이	지역
사람 1	1	20	서울
사람 2	2	25	대전

```
In [27]: df.columns = ['연번', '연령', '도시']
```

```
In [28]: df
```

```
Out[28]:
```

	연번	연령	도시
사람 1	1	20	서울
사람 2	2	25	대전

```
In [29]: df = df.rename(index = {'사람 1': '홍길동'})
```

```
In [30]: df
```

```
Out[30]:
```

	연번	연령	도시
홍길동	1	20	서울
사람 2	2	25	대전

```
In [31]: df = df.rename(columns = {'연번': '순서', '연령': '나이'})
```

```
In [32]: df
```

```
Out[32]:
```

	순서	나이	도시
홍길동	1	20	서울
사람 2	2	25	대전

pandas의 2차원 자료형 - DataFrame 2

- 행 추가
- 열 추가
- 값 변경
- 행 삭제
- 열 삭제

In [34]: df

Out[34]:

	순서	나이	도시
홍길동	1	20	서울
사람 2	2	25	대전

In [35]: df.loc['사람 3'] = [3, 30, '부산']

In [36]: df

Out[36]:

	순서	나이	도시
홍길동	1	20	서울
사람 2	2	25	대전
사람 3	3	30	부산

In [37]: df.loc['사람 4'] = 0

In [38]: df

Out[38]:

	순서	나이	도시
홍길동	1	20	서울
사람 2	2	25	대전
사람 3	3	30	부산
사람 4	0	0	0

In [39]: df.loc['사람 1'] = df.loc['홍길동']

In [40]: df

Out[40]:

	순서	나이	도시
홍길동	1	20	서울
사람 2	2	25	대전
사람 3	3	30	부산
사람 4	0	0	0
사람 1	1	20	서울

In [41]: df['나이대'] = ['20대', '20대', '30대', '0대', '20대']

In [42]: df

Out [42]:

	순서	나이	도시	나이대
홍길동	1	20	서울	20대
사람 2	2	25	대전	20대
사람 3	3	30	부산	30대
사람 4	0	0	0	0대
사람 1	1	20	서울	20대

In [43]: `df['직업'] = '학생'`

In [44]: `df`

Out [44]:

	순서	나이	도시	나이대	직업
홍길동	1	20	서울	20대	학생
사람 2	2	25	대전	20대	학생
사람 3	3	30	부산	30대	학생
사람 4	0	0	0	0대	학생
사람 1	1	20	서울	20대	학생

In [45]: `df.loc['사람 4', '순서'] = 4`

In [46]: `df`

Out [46]:

	순서	나이	도시	나이대	직업
홍길동	1	20	서울	20대	학생
사람 2	2	25	대전	20대	학생
사람 3	3	30	부산	30대	학생
사람 4	4	0	0	0대	학생
사람 1	1	20	서울	20대	학생

In [47]: `df.iloc[3, 1] = 10`

In [48]: `df`

Out [48]:

	순서	나이	도시	나이대	직업
홍길동	1	20	서울	20대	학생
사람 2	2	25	대전	20대	학생
사람 3	3	30	부산	30대	학생
사람 4	4	10	0	0대	학생
사람 1	1	20	서울	20대	학생

In [49]: `df.loc['사람 4', ['도시', '나이대']] = '광주', '10대'`

In [50]: `df`

Out[50]:

	순서	나이	도시	나이대	직업
홍길동	1	20	서울	20대	학생
사람 2	2	25	대전	20대	학생
사람 3	3	30	부산	30대	학생
사람 4	4	10	광주	10대	학생
사람 1	1	20	서울	20대	학생

In [51]: `df.iloc[3, [1, 3]] = 40, '40대'`

In [52]: `df`

Out[52]:

	순서	나이	도시	나이대	직업
홍길동	1	20	서울	20대	학생
사람 2	2	25	대전	20대	학생
사람 3	3	30	부산	30대	학생
사람 4	4	40	광주	40대	학생
사람 1	1	20	서울	20대	학생

In [53]: `df = df.drop('사람 4', axis = 0)`

In [54]: `df`

Out[54]:

	순서	나이	도시	나이대	직업
홍길동	1	20	서울	20대	학생
사람 2	2	25	대전	20대	학생
사람 3	3	30	부산	30대	학생
사람 1	1	20	서울	20대	학생

In [55]: `df = df.drop(['나이대', '직업'], axis = 1)`

In [56]: `df`

Out[56]:

	순서	나이	도시
홍길동	1	20	서울
사람 2	2	25	대전
사람 3	3	30	부산
사람 1	1	20	서울

pandas의 2차원 자료형 - DataFrame 3

- 전체 복사
- 부분 복사

```
In [57]: df2 = df
```

```
In [58]: df2
```

```
Out[58]:
```

	순서	나이	도시
홍길동	1	20	서울
사람 2	2	25	대전
사람 3	3	30	부산
사람 1	1	20	서울

```
In [59]: df2.loc['사람 1', '순서'] = 4
```

```
In [60]: df2
```

```
Out[60]:
```

	순서	나이	도시
홍길동	1	20	서울
사람 2	2	25	대전
사람 3	3	30	부산
사람 1	4	20	서울

```
In [61]: df
```

```
Out[61]:
```

	순서	나이	도시
홍길동	1	20	서울
사람 2	2	25	대전
사람 3	3	30	부산
사람 1	4	20	서울

```
In [62]: df3 = df.copy()
```

```
In [63]: df3
```

```
Out[63]:
```

	순서	나이	도시
홍길동	1	20	서울
사람 2	2	25	대전
사람 3	3	30	부산
사람 1	4	20	서울

```
In [64]: df3.loc['사람 1', '순서'] = 5
```

```
In [65]: df3
```

```
Out[65]:
```

	순서	나이	도시
홍길동	1	20	서울

	순서	나이	도시
사람 2	2	25	대전
사람 3	3	30	부산
사람 1	5	20	서울

In [66]: df

Out[66]:

	순서	나이	도시
홍길동	1	20	서울
사람 2	2	25	대전
사람 3	3	30	부산
사람 1	4	20	서울

In [67]: hong = df.loc['홍길동']

In [68]: hong

Out[68]:

```

순서      1
나이     20
도시     서울
Name: 홍길동, dtype: object

```

In [69]: type(hong)

Out[69]: pandas.core.series.Series

In [70]: hong.shape

Out[70]: (3,)

In [71]: gil = df.iloc[0]

In [72]: gil

Out[72]:

```

순서      1
나이     20
도시     서울
Name: 홍길동, dtype: object

```

In [73]: type(gil)

Out[73]: pandas.core.series.Series

In [74]: gil.shape

Out[74]: (3,)

In [75]: df4 = df.loc[['홍길동', '사람 1']]

In [76]: df4

Out[76]:

	순서	나이	도시
--	----	----	----

	순서	나이	도시
홍길동	1	20	서울
사람 1	4	20	서울

In [77]: `type(df4)`

Out[77]: `pandas.core.frame.DataFrame`

In [78]: `df5 = df.iloc[[0, 3]]`

In [79]: `df5`

Out[79]:

	순서	나이	도시
홍길동	1	20	서울
사람 1	4	20	서울

In [80]: `df6 = df.loc['홍길동':'사람 3']`

In [81]: `df6`

Out[81]:

	순서	나이	도시
홍길동	1	20	서울
사람 2	2	25	대전
사람 3	3	30	부산

In [82]: `df7 = df.iloc[0: 3]`

In [83]: `df7`

Out[83]:

	순서	나이	도시
홍길동	1	20	서울
사람 2	2	25	대전
사람 3	3	30	부산

In [84]: `df`

Out[84]:

	순서	나이	도시
홍길동	1	20	서울
사람 2	2	25	대전
사람 3	3	30	부산
사람 1	4	20	서울

In [85]: `df8 = df[['나이']]`

In [86]: `df8`

Out[86]:

	나이
홍길동	20
사람 2	25
사람 3	30
사람 1	20

In [87]: df9 = df['나이']

In [88]: df9

Out[88]:

홍길동	20
사람 2	25
사람 3	30
사람 1	20

Name: 나이, dtype: int64

In [89]: type(df8)

Out[89]: pandas.core.frame.DataFrame

In [90]: type(df9)

Out[90]: pandas.core.series.Series

In [91]: df10 = df[['나이', '도시']]

In [92]: df10

Out[92]:

	나이	도시
홍길동	20	서울
사람 2	25	대전
사람 3	30	부산
사람 1	20	서울

In [93]: df

Out[93]:

	순서	나이	도시
홍길동	1	20	서울
사람 2	2	25	대전
사람 3	3	30	부산
사람 1	4	20	서울

In [94]: df11 = df.loc[['홍길동', '사람 1'], ['나이', '도시']]

In [95]: df11

Out[95]:

	나이	도시
홍길동	20	서울

	나이	도시
--	----	----

사람 1	20	서울
------	----	----

```
In [96]: df12 = df.iloc[[0, 3], [1, 2]]
```

```
In [97]: df12
```

```
Out[97]:
```

	나이	도시
--	----	----

홍길동	20	서울
-----	----	----

사람 1	20	서울
------	----	----

pandas의 2차원 자료형 - DataFrame 4

- 인덱스
- 정렬
- 전치
- 치환

```
In [98]: a = {'이름': ['홍길동', '사람 1', '사람 2'], '나이': [20, 30, 25], '도시': ['서울', '서울', '대전']}
```

```
In [99]: df = pd.DataFrame(a)
```

```
In [100]: df
```

```
Out[100]:
```

	이름	나이	도시
--	----	----	----

0	홍길동	20	서울
---	-----	----	----

1	사람 1	30	서울
---	------	----	----

2	사람 2	25	대전
---	------	----	----

```
In [101]: df2 = df.set_index(['이름'])
```

```
In [102]: df2
```

```
Out[102]:
```

	나이	도시
--	----	----

이름

홍길동	20	서울
-----	----	----

사람 1	30	서울
------	----	----

사람 2	25	대전
------	----	----

```
In [103]: df3 = df2.set_index(['도시'])
```

```
In [104]: df3
```

```
Out[104]:
```

나이

도시

서울	20
----	----

나이	
도시	
서울	30
대전	25

In [105]: df2

Out[105]:

나이		도시
이름		
홍길동	20	서울
사람 1	30	서울
사람 2	25	대전

In [106]: df3 = df2.reset_index()

In [107]: df3

Out[107]:

	이름	나이	도시
0	홍길동	20	서울
1	사람 1	30	서울
2	사람 2	25	대전

In [108]: df3 = df3.set_index(['도시'])

In [109]: df3

Out[109]:

이름		나이
도시		
서울	홍길동	20
서울	사람 1	30
대전	사람 2	25

In [110]: df2

Out[110]:

나이		도시
이름		
홍길동	20	서울
사람 1	30	서울
사람 2	25	대전

In [111]: df2 = df2.sort_index(ascending = True)

In [112]: df2

Out[112]:

	나이	도시
이름		
사람 1	30	서울
사람 2	25	대전
홍길동	20	서울

In [113]: df2 = df2.sort_index(ascending = False)

In [114]: df2

Out[114]:

	나이	도시
이름		
홍길동	20	서울
사람 2	25	대전
사람 1	30	서울

In [115]: df2 = df2.sort_values(by = '나이', ascending = True)

In [116]: df2

Out[116]:

	나이	도시
이름		
홍길동	20	서울
사람 2	25	대전
사람 1	30	서울

In [117]: df2 = df2.sort_values(by = '나이', ascending = False)

In [118]: df2

Out[118]:

	나이	도시
이름		
사람 1	30	서울
사람 2	25	대전
홍길동	20	서울

In [119]: df2 = df2.T

In [120]: df2

Out[120]:

이름	사람 1	사람 2	홍길동
나이	30	25	20
도시	서울	대전	서울


```
In [121]: df2 = df2.T
```

```
In [122]: df2
```

```
Out[122]:
```

	나이	도시
	이름	
	사람 1	30 서울
	사람 2	25 대전
	홍길동	20 서울

```
In [123]: df2['나이대'] = ['30대', '20대', '20대']
```

```
In [124]: df2
```

```
Out[124]:
```

	나이	도시	나이대
	이름		
	사람 1	30 서울	30대
	사람 2	25 대전	20대
	홍길동	20 서울	20대

```
In [125]: df2 = df2.reset_index()
```

```
In [126]: df2
```

```
Out[126]:
```

	이름	나이	도시	나이대
0	사람 1	30	서울	30대
1	사람 2	25	대전	20대
2	홍길동	20	서울	20대

```
In [127]: b = {'20대': 2, '30대': 3}
```

```
In [128]: df2['나이대'] = df2['나이대'].map(b)
```

```
In [129]: df2
```

```
Out[129]:
```

	이름	나이	도시	나이대
0	사람 1	30	서울	3
1	사람 2	25	대전	2
2	홍길동	20	서울	2

```
In [130]: df2.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3 entries, 0 to 2
Data columns (total 4 columns):
#   Column  Non-Null Count  Dtype
---  -

```

0	이름	3 non-null	object
1	나이	3 non-null	object
2	도시	3 non-null	object
3	나이대	3 non-null	int64

dtypes: int64(1), object(3)
memory usage: 224.0+ bytes

pandas의 기본 연산 - Series

```
In [1]: import pandas as pd
```

```
In [2]: a = {'a': 10, 'b': 20, 'c': 30, 'd': 40, 'e': 50}
```

```
In [3]: a
```

```
Out[3]: {'a': 10, 'b': 20, 'c': 30, 'd': 40, 'e': 50}
```

```
In [4]: b = pd.Series(a)
```

```
In [5]: b
```

```
Out[5]: a    10  
       b    20  
       c    30  
       d    40  
       e    50  
       dtype: int64
```

```
In [6]: type(b)
```

```
Out[6]: pandas.core.series.Series
```

```
In [7]: c = b / 10
```

```
In [8]: c
```

```
Out[8]: a    1.0  
       b    2.0  
       c    3.0  
       d    4.0  
       e    5.0  
       dtype: float64
```

```
In [9]: c = b // 10
```

```
In [10]: c
```

```
Out[10]: a    1  
        b    2  
        c    3  
        d    4  
        e    5  
        dtype: int64
```

```
In [11]: type(c)
```

```
Out[11]: pandas.core.series.Series
```

```
In [12]: d = b + c
```

```
In [13]: d
```

```
Out[13]: a    11  
        b    22  
        c    33  
        d    44
```

```
e    55  
dtype: int64
```

```
In [14]: e = {'a': 10, 'c': 20, 'e': 30, 'g': 40}
```

```
In [15]: e
```

```
Out[15]: {'a': 10, 'c': 20, 'e': 30, 'g': 40}
```

```
In [16]: f = pd.Series(e)
```

```
In [17]: f
```

```
Out[17]: a    10  
         c    20  
         e    30  
         g    40  
dtype: int64
```

```
In [18]: d + f
```

```
Out[18]: a    21.0  
         b     NaN  
         c    53.0  
         d     NaN  
         e    85.0  
         g     NaN  
dtype: float64
```

```
In [19]: d.add(f, fill_value = 0)
```

```
Out[19]: a    21.0  
         b    22.0  
         c    53.0  
         d    44.0  
         e    85.0  
         g    40.0  
dtype: float64
```

```
In [20]: d.sub(f, fill_value = 0)
```

```
Out[20]: a     1.0  
         b    22.0  
         c    13.0  
         d    44.0  
         e    25.0  
         g   -40.0  
dtype: float64
```

```
In [21]: d.mul(f, fill_value = 0)
```

```
Out[21]: a    110.0  
         b     0.0  
         c   660.0  
         d     0.0  
         e  1650.0  
         g     0.0  
dtype: float64
```

```
In [22]: d * f
```

```
Out[22]: a    110.0  
         b     NaN  
         c   660.0  
         d     NaN  
         e  1650.0
```

```
g      NaN
dtype: float64
```

```
In [23]: d.mul(f, fill_value = 1)
```

```
Out[23]: a      110.0
b       22.0
c      660.0
d       44.0
e     1650.0
g       40.0
dtype: float64
```

```
In [24]: d.div(f, fill_value = 0)
```

```
Out[24]: a      1.100000
b         inf
c      1.650000
d         inf
e      1.833333
g      0.000000
dtype: float64
```

```
In [25]: d.div(f, fill_value = 1)
```

```
Out[25]: a      1.100000
b     22.000000
c      1.650000
d     44.000000
e      1.833333
g      0.025000
dtype: float64
```

pandas의 기본 연산 - DataFrame 1

```
In [34]: a = {'점수 1': [10, 20, 30], '점수 2': [20, 40, 60]}
```

```
In [35]: df1 = pd.DataFrame(a)
```

```
In [36]: df1
```

```
Out[36]:
```

	점수 1	점수 2
0	10	20
1	20	40
2	30	60

```
In [37]: b = {'점수 1': [100, 200, 300], '점수 3': [200, 400, 600]}
```

```
In [38]: df2 = pd.DataFrame(b)
```

```
In [39]: df2
```

```
Out[39]:
```

	점수 1	점수 3
0	100	200
1	200	400
2	300	600

```
In [40]: df3 = df1 + df2
```

```
In [41]: df3
```

```
Out[41]:
```

	점수 1	점수 2	점수 3
0	110	NaN	NaN
1	220	NaN	NaN
2	330	NaN	NaN

```
In [42]: df4 = df1.add(df2, fill_value = 0)
```

```
In [43]: df4
```

```
Out[43]:
```

	점수 1	점수 2	점수 3
0	110	20.0	200.0
1	220	40.0	400.0
2	330	60.0	600.0

```
In [44]: df5 = df1.sub(df2, fill_value = 0)
```

```
In [45]: df5
```

```
Out[45]:
```

	점수 1	점수 2	점수 3
0	-90	20.0	-200.0
1	-180	40.0	-400.0
2	-270	60.0	-600.0

```
In [46]: df6 = df1.mul(df2, fill_value = 0)
```

```
In [47]: df6
```

```
Out[47]:
```

	점수 1	점수 2	점수 3
0	1000	0.0	0.0
1	4000	0.0	0.0
2	9000	0.0	0.0

```
In [48]: df6 = df1.mul(df2, fill_value = 1)
```

```
In [49]: df6
```

```
Out[49]:
```

	점수 1	점수 2	점수 3
0	1000	20.0	200.0
1	4000	40.0	400.0
2	9000	60.0	600.0

```
In [50]: df7 = df1.div(df2, fill_value = 0)
```

```
In [51]: df7
```

```
Out[51]:
```

	점수 1	점수 2	점수 3
0	0.1	inf	0.0
1	0.1	inf	0.0
2	0.1	inf	0.0

```
In [52]: df7 = df1.div(df2, fill_value = 1)
```

```
In [53]: df7
```

```
Out[53]:
```

	점수 1	점수 2	점수 3
0	0.1	20.0	0.005000
1	0.1	40.0	0.002500
2	0.1	60.0	0.001667

pandas의 기본 연산 - DataFrame 2

```
In [54]: a = {'이름': ['홍길동', '사람 1', '사람 2'], '나이': [20, 30, 25]}
```

```
In [55]: df1 = pd.DataFrame(a)
```

```
In [56]: df1
```

```
Out[56]:
```

	이름	나이
0	홍길동	20
1	사람 1	30
2	사람 2	25

```
In [57]: b = {'이름': ['홍길동', '사람 1', '사람 2'], '도시': ['서울', '서울', '대전']}
```

```
In [58]: df2 = pd.DataFrame(b)
```

```
In [59]: df2
```

```
Out[59]:
```

	이름	도시
0	홍길동	서울
1	사람 1	서울
2	사람 2	대전

```
In [60]: df = pd.merge(left = df1,
                        right = df2,
                        how = 'left',
                        left_on = '이름',
                        right_on = '이름')
```

```
In [61]: df
```

```
Out[61]:
```

	이름	나이	도시
0	홍길동	20	서울
1	사람 1	30	서울
2	사람 2	25	대전

```
In [62]: x1 = (df['도시'] == '서울')
```

```
In [63]: df[x1]
```

```
Out[63]:
```

	이름	나이	도시
0	홍길동	20	서울
1	사람 1	30	서울

```
In [64]: x2 = (df['도시'] != '서울')
```

```
In [65]: df[x2]
```

```
Out[65]:
```

	이름	나이	도시
2	사람 2	25	대전

```
In [66]: x3 = (df['나이'] < 30)
```

```
In [67]: df[x3]
```

```
Out[67]:
```

	이름	나이	도시
0	홍길동	20	서울
2	사람 2	25	대전

```
In [68]: x4 = (df['도시'] == '서울') & (df['나이'] < 30)
```

```
In [69]: df[x4]
```

```
Out[69]:
```

	이름	나이	도시
0	홍길동	20	서울

```
In [70]: x5 = (df['도시'] == '서울') | (df['나이'] < 30)
```

```
In [71]: df[x5]
```

```
Out[71]:
```

	이름	나이	도시
0	홍길동	20	서울
1	사람 1	30	서울
2	사람 2	25	대전