

19강

pandas1

19-1 pandas 개요 및 자료구조1

- pandas는 데이터 조작 및 분석을 위해 고수준 자료구조를 지원하는 소프트웨어 라이브러리이다.
- pandas의 특징은 아래와 같다.
 - ✓ 축의 정보를 이용하여 데이터를 추출 및 정렬할 수 있는 자료구조를 제공한다.
 - ✓ 데이터 그룹화를 이용한 데이터 처리 기능을 제공한다.
 - ✓ 시계열 데이터 처리 기능을 제공한다.
 - ✓ 누락된 데이터를 유연하게 처리할 수 있다.
 - ✓ SQL과 같은 일반 데이터베이스처럼 데이터를 합치고 관계 연산을 수행할 수 있다.



19-1 pandas 개요 및 자료구조1

● Series

- ✓ 일련의 객체를 담을 수 있는 1차원 배열 같은 자료 구조이다.
- ✓ index라고 하는 배열 데이터에 연관된 이름을 가지고 있다.
- ✓ Series 객체의 문자열 표현은 왼쪽에 색인을 보여주고 오른쪽에 해당 색의 값을 보여준다.

Series	
Index	Value
0	92600
1	92400
2	92100
3	94300
4	92300

19-1 pandas 개요 및 자료구조1

● Series

- ✓ Series 객체를 생성할 때는 index를 지정하지 않으면 자동으로 생성해준다.

```
import pandas as pd
import numpy as np

obj = pd.Series([4, 7, -5, 3])
print(obj)
print(obj.values)

obj2 = pd.Series([4, 7, -5, 3], index=['a', 'b', 'a', 'c'])
print(obj2)

print(obj2['a'])
obj2['d'] = 6
print(obj2[['c', 'a', 'd']])

print(obj2 > 0)
print(obj2[obj2 > 0])
print(obj2 * 2)
print(np.exp(obj2))

sdata = {'Ohio': 35000, 'Texas': 71000, 'Oregon': 16000, 'Utah': 5000}
obj3 = pd.Series(sdata)
print(obj3)
```

```
[결과]
0    4
1    7
2   -5
3     3
dtype: int64
[ 4  7 -5  3]

a    4
b    7
a   -5
c     3
dtype: int64

a    4
a   -5
dtype: int64

c     3
a     4
a   -5
dtype: int64

a    True
b    True
a    False
c     True
d     True
dtype: bool
```

```
[결과]
a    4
b    7
c     3
d     6
dtype: int64

a     8
b    14
a   -10
c     6
d    12
dtype: int64

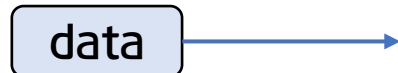
a    54.598150
b   1096.633158
a     0.006738
c    20.085537
d   403.428793
dtype: float64

Ohio    35000
Texas   71000
Oregon   16000
Utah     5000
dtype: int64
```

19-1 pandas 개요 및 자료구조1

● Dataframe

- ✓ 표 같은 스프레드시트 형식의 자료 구조로 여러 개의 칼럼이 있는데 서로 다른 종류의 값을 담을 수 있다.
- ✓ DataFrame은 색인의 모양이 같은 여러 개의 Series 객체를 담고 있다고 생각하면 된다.



DataFrame			
	Series ('col0')	Series ('col1')	Series ('col2')
Index	Value	Value	Value
0	1	10	100
1	2	20	200
2	3	30	300
3	4	40	400

19-2 pandas 개요 및 자료구조2

● Dataframe

```
import pandas as pd
import numpy as np

frame1 = pd.DataFrame(np.arange(6).reshape(2,3),
                      index=['first', 'second'],
                      columns=['one', 'two', 'three'])
print(frame1)

data = {'state': ['Ohio', 'Ohio', 'Ohio', 'Nevada', 'Nevada', 'Nevada'],
        'year': [2000, 2001, 2002, 2001, 2002, 2003],
        'pop': [1.5, 1.7, 3.6, 2.4, 2.9, 3.2]}

print(pd.DataFrame(data, columns=['year', 'state', 'pop']))

frame2 = pd.DataFrame(data, columns=['year', 'state', 'pop'],
                      index=['one', 'two', 'three', 'four', 'five', 'six'])
print(frame2)

print(frame2.columns)
print(frame2['state'])
print(frame2.year) #frame2['year']
print(frame2.loc['three'])
```

[결과]

```
one two three
first 0 1 2
second 3 4 5
```

```
   year state pop
0  2000  Ohio  1.5
1  2001  Ohio  1.7
2  2002  Ohio  3.6
3  2001 Nevada  2.4
4  2002 Nevada  2.9
5  2003 Nevada  3.2
```

```
   year state pop
one  2000  Ohio  1.5
two  2001  Ohio  1.7
three 2002  Ohio  3.6
four  2001 Nevada  2.4
five  2002 Nevada  2.9
six   2003 Nevada  3.2
```

```
Index(['year', 'state', 'pop'],
      dtype='object')
```

[결과]

```
one    Ohio
two    Ohio
three  Ohio
four   Nevada
five   Nevada
six    Nevada
Name: state, dtype:
object
```

```
   one    2000
two    2001
three   2002
four    2001
five    2002
six     2003
Name: year, dtype:
int64
```

```
   year    2002
state   Ohio
pop     3.6
Name: three, dtype:
object
```

19-2 pandas 개요 및 자료구조2

● 인덱싱

- ✓ Series와 Dataframe의 색인은 index와 column 정보 외에도 numpy 배열의 색인과 유사하게 배열 인덱스를 이용할 수 있다.

```
import pandas as pd
import numpy as np

obj = pd.Series([4,2,6,9], index=list('abcd'))
print(obj)
print(obj['b'])
print(obj[1])
print(obj[1:3])
print(obj['b':'c'])

frame = pd.DataFrame(np.arange(16).reshape(4,4),
                      index=['Ohio', 'Colorado', 'Utah', 'NewYork'],
                      columns=['one', 'two', 'three', 'four'])
print(frame)
#print('Wn', frame.iloc[2, [3,0,1]])

print(frame[frame['three'] > 5])
print(frame.loc[:, 'Utah', 'two'])
print(frame.iloc[:, :3][frame.three>5])
```

[결과]

```
a  4
b  2
c  6
d  9
dtype: int64
2
2
```

```
b  2
c  6
dtype: int64
```

```
b  2
c  6
dtype: int64
```

	one	two	three	four
Ohio	0	1	2	3
Colorado	4	5	6	7
Utah	8	9	10	11
NewYork	12	13	14	15

[결과]

	one	two	three	four
Colorado	4	5	6	7
Utah	8	9	10	11
NewYork	12	13	14	15

```
Ohio      1
Colorado   5
Utah       9
Name: two, dtype: int32
```

	one	two	three
Colorado	4	5	6
Utah	8	9	10
NewYork	12	13	14