

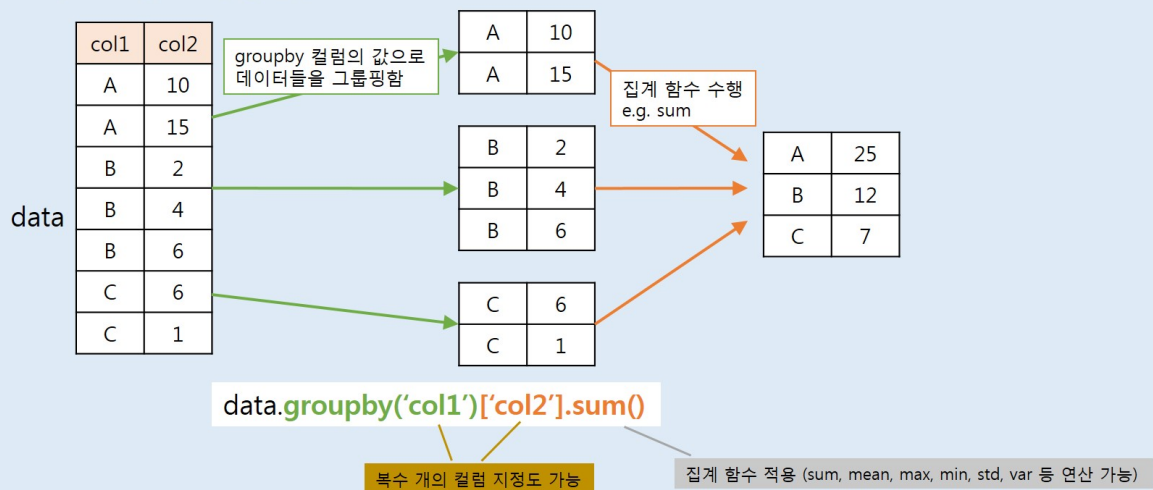
## pandas, numpy module 불러오기

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

### 그룹 분석

- SQL 구문의 GROUP BY와 집계 연산과 유사
- 데이터를 특정 속성의 값으로 그룹핑하여 집계를 수행함.
- 집계는 sum, max, min, mean, std 등 다양한 연산 가능
- Pandas에서 그룹 분석을 하는 방법은 크게 방법이 있다.

- ① Groupby 활용 활용
- ② Pivot\_table 함수 활용



### 그룹 분석 - pivot\_table()

구문 `data.pivot_table(index = '연도', columns = '도시', aggfunc = 'sum', values = '총인구')`

결과

	도시	부산	서울
연도			
2013	699	1306	
2014	448	1401	
2015	776	1355	
2016	567	1141	
2017	603	1642	

index : 로우 인덱스로 활용할 컬럼  
columns: 컬럼 인덱스로 활용할 컬럼  
aggfunc: 적용할 집계 함수  
values: 집계함수를 적용할 컬럼  
\* 모든 인자는 List의 형태로 복수 개 지정 가능

```
In [2]: # 실습 데이터 적재
data = pd.read_excel('data/인구수예제.xlsx')
data.head()
```

```
Out[2]:
```

	도시	자치구	연도	남자인구	여자인구	총인구
0	서울	강남구	2013	73	92	165
1	서울	강남구	2014	139	55	194
2	서울	강남구	2015	123	83	206
3	서울	강남구	2016	147	150	297
4	서울	강남구	2017	57	133	190

## 1. groupby

**(dataframe).groupby('집단별로 묶고 싶은 기준 column')['이후에 연산을 하고 싶은 column'].함수**

```
In [3]: # 연도별 총인구의 합
# 집단별로 묶고 싶은 기준 : '연도'
# 연산을 하고 싶은 column : '총인구'
# 사용할 함수 : 'sum()'
data.groupby('연도')['총인구'].sum()
```

```
Out[3]: 연도
2013    2130
2014    2256
2015    1848
2016    2045
2017    2128
Name: 총인구, dtype: int64
```

```
In [4]: pd.DataFrame(data.groupby('연도')['총인구'].sum())
```

```
Out[4]:
```

	총인구
연도	
2013	2130
2014	2256
2015	1848
2016	2045
2017	2128

```
In [5]: # reset_index() : index의 값을 column으로 이동
pd.DataFrame(data.groupby('연도')['총인구'].sum()).reset_index()
```

```
Out[5]:
```

	연도	총인구
0	2013	2130
1	2014	2256
2	2015	1848
3	2016	2045
4	2017	2128

```
In [7]: # 자치구별 총인구의 평균
data.groupby('자치구')['총인구'].mean()
```

```
Out[7]:
```

자치구	
강남구	210.4
도봉구	207.0
동래구	207.2
동작구	207.2
서대문구	206.2
송파구	194.8
수영구	212.2
영등포구	238.2
종로구	171.2
해운대구	227.0

Name: 총인구, dtype: float64

## 2. pivot\_table

**(dataframe).pivot\_table(index="", aggfunc="", values="")**

**index='집단별로 묶고 싶은 기준 column',**

**aggfunc=함수,**

**values='이후에 연산을 하고 싶은 column'**

```
In [8]: # 연도별 총인구의 합
data.pivot_table(index = '연도', aggfunc = 'sum', values = '총인구')
```

```
Out[8]:
```

	총인구
연도	
2013	2130
2014	2256
2015	1848
2016	2045
2017	2128

```
In [9]: # 자치구별 총인구의 평균
data.pivot_table(index = '자치구', aggfunc = 'mean', values = '총인구')
```

Out[9]:

	총인구
자치구	
강남구	210.4
도봉구	207.0
동래구	207.2
동작구	207.2
서대문구	206.2
송파구	194.8
수영구	212.2
영등포구	238.2
종로구	171.2
해운대구	227.0

## group, pivot\_table 예제

### 1) 연도별 전체의 남자인구, 여자인구, 총인구 수 합 구하기

```
In [10]: # groupby
data.groupby('연도')[['남자인구', '여자인구', '총인구']].sum()
```

Out[10]:

	남자인구	여자인구	총인구
연도			
2013	1099	1031	2130
2014	1196	1060	2256
2015	872	976	1848
2016	947	1098	2045
2017	1063	1065	2128

```
In [11]: # pivot_table
data.pivot_table(index = '연도', aggfunc = 'sum', values = ['남자인구', '여자인구'],
```

Out[11]:

	남자인구	여자인구	총인구
연도			
2013	1099	1031	2130
2014	1196	1060	2256
2015	872	976	1848
2016	947	1098	2045
2017	1063	1065	2128

## 2) 도시/자치구별 평균 총인구수

```
In [12]: # groupby
data.groupby(['도시', '자치구'])['총인구'].mean()
```

Out[12]:

도시	자치구	
부산	동래구	207.2
	수영구	212.2
	해운대구	227.0
서울	강남구	210.4
	도봉구	207.0
	동작구	207.2
	서대문구	206.2
	송파구	194.8
	영등포구	238.2
	종로구	171.2

Name: 총인구, dtype: float64

```
In [15]: # pivot_table
data.pivot_table(index = ['도시', '자치구'], aggfunc = 'mean', values = ['총인구'])
```

Out[15]:

		총인구
도시	자치구	
	동래구	207.2
부산	수영구	212.2
	해운대구	227.0
	강남구	210.4
	도봉구	207.0
	동작구	207.2
서울	서대문구	206.2
	송파구	194.8
	영등포구	238.2
	종로구	171.2

## 3) 도시별, 연도별로 총인구수 합 출력

```
In [19]: # groupby
pd.DataFrame(data.groupby(['도시', '연도'])['총인구'].sum()).unstack('도시')
```

Out[19]:

총인구		
도시	부산	서울
연도		
2013	603	1527
2014	683	1573
2015	597	1251
2016	652	1393
2017	697	1431

```
In [17]: # pivot_table
data.pivot_table(index = '연도', columns = '도시', aggfunc = 'sum', values = '총인
```

Out[17]:

도시	부산	서울
연도		
2013	603	1527
2014	683	1573
2015	597	1251
2016	652	1393
2017	697	1431

## 실습

### 1. 자치구별로 평균 총 인구수

```
In [ ]: # groupby 사용
```

```
In [ ]: # pivot_table 사용
```

### 2. 도시/자치구별 평균 남자인구와 여자인구수 구하기

```
In [ ]: # groupby 사용
```

```
In [ ]: # pivot_table 사용
```

### 3. 도시/자치구 별로 남자인구의 평균을 구한 후, 남자인구가 가장 많은 도시/자치구 5개를 출력하시오.

```
In [ ]: # groupby 사용
```

```
In [ ]: # pivot_table 사용
```

## 함수를 여러개 적용하고 싶은 경우

```
In [4]: # 연도, 도시별로 총인구의 평균과, 표준편차 구해보기
```

```
Out[4]:
```

	mean		std	
도시	부산	서울	부산	서울
연도				
2013	201.000000	218.142857	45.902070	33.982488
2014	227.666667	224.714286	16.563011	52.184654
2015	199.000000	178.714286	57.297469	31.731763
2016	217.333333	199.000000	49.541229	62.372537
2017	232.333333	204.428571	12.662280	32.315410

```
In [ ]: # groupby 함수에서는 agg() 함수를 적용하고 안에 list형식으로 넣어준다
```

```
In [ ]: # 실습. 도시/자치구별 기준으로 남녀차이의 평균이 가장 많이 나는 도시/자치구 3개를 찾아보기
# Hint: abs() = 절대값을 구하는 함수
```