

02

데이터 결합 (concat, merge)

< 행결합 >

00:09

1.5 데이터 결합

R과 다른점 : 결합하려는 데이터프레임의 행과 열의 개수가 맞지 않아도 NaN값이 채워짐

1.5.1 행결합

1.5.1.1 데이터프레임 붙이기 : pd.concat()

```
In [35]: import pandas as pd

customer1 = pd.DataFrame({'id': ['c01', 'c02', 'c03', 'c04'],
                          'last_name': ['Lee', 'Kim', 'Choi', 'Park']},
                          index = [0, 1, 2, 3])

customer2 = pd.DataFrame({'id': ['c05', 'c06', 'c07'],
                          'Last_name': ['Lim', 'Bae', 'Kim'],
                          'age': [23, 34, 45]},
                          index = [0, 1, 2])

In [ ]:
```

< 열결합 >

05:18

1.5.2 열결합

1.5.2.1 데이터프레임 붙이기 : pd.concat()

행인덱스와 열인덱스가 다른 데이터프레임 합치기

- `pd.concat([df1, df2, ...], axis=1)`

`axis=1`은 열방향으로 데이터프레임을 붙임
`customer2`에는 index 3행이 없으므로 NaN값이 채워짐

```
In [ ]: result2 = pd.concat([customer2, customer1], axis=1)
result2
```

1.5.2.2 시리즈를 데이터프레임에 붙이기 : pd.concat()

시리즈 객체를 생성할 때 `name`은, 이 시리즈가 데이터프레임에 결합되었을 때의 열이름을 나타냄

```
In [ ]: grade = pd.Series(['A', 'B', 'C', 'A', 'D', 'C', 'B'], name='grade')
grade2 = pd.Series(['A', 'B', 'C', 'A', 'D', 'C'], name='grade')
```

[merge 데이터 결합]

07:46

SQL과 같이 on = key값을 가지고 데이터를 병합하는 것이 매우 중요합니다!

```

1 e1
2 e2
3 e3
4 g0
5 g1
6 g2
7 g3
dtype: object
<class 'pandas.core.series.Series'>

```

1.5.3 merge

두 데이터프레임에 존재하는 고유값(Key)을 기준으로 병합

- `pd.merge(df_left, df_right, how='inner', on=None)` 이 default

how에 들어갈 수 있는 매개변수 : 'inner', 'outer', 'left', 'right'

```

In [ ]: id_name = pd.DataFrame({'id' : ['c01', 'c02', 'c03', 'c04', 'c05', 'c06', 'c07'],
                                'last_name': ['Lee', 'Kim', 'Choi', 'Park', 'Lim', 'Bae', 'Kim']})
id_number = pd.DataFrame({'id' : ['c03', 'c04', 'c05', 'c06', 'c07', 'c08', 'c09'],
                            'number' : [3, 1, 0, 7, 3, 4, 1]})

In [ ]: id_name

```

다음 강의는 3챕터 데이터 핸들링(2) 가 진행됩니다. `groupby`를 이용하여 원하는 변수로 그룹을 묶어서 데이터를 요약하는 방법을 배워볼 거예요

다음 강의에서 봐요~