
세상의 속도를
따라잡고 싶다면
Do it!
데이터 분석을 위한
판다스 입문
7일 완성!
테슬라 주식 • 우버 택시 • 빌보드 차트 • 예볼라 바이러스 데이터 등
현실 데이터 기반의 86개 예제로 판다스 실무 기본기 완성!
Chen, Daniel Y. 지음 | 강경덕 옮김
이지스퍼블리싱
파이썬 3.9 판다스
최신 버전 반영!
데이터 분석을 위한 판다스 입문

5장 데이터 연결하기

주피터 노트북 테마 설정하기

1. cmd로 console창으로 이동
2. pip install jupyterthemes
3. jt –l (소문자 L)
4. jt –t grade3

```
chesterish
grade3
gruvboxd
gruvboxl
monokai
oceans16
onedork
solarizedd
solarizedl
```

05-1 분석하기 좋은 데이터

분석하기 좋은 데이터란?

① 데이터 분석 목적에 맞는 데이터 모으기
② 측정한 값은 행(row)를 구성
③ 변수는 열(column)을 구성

분석하기 좋은 데이터 = 깔끔한 데이터(Tidy Data)

05-2 데이터 연결 기초

데이터 연결하기(1)

Concat 메서드로 데이터 연결(인덱스 유지되는 점 주의)

```
import pandas as pd

df1 = pd.read_csv('../data/concat_1.csv')
df2 = pd.read_csv('../data/concat_2.csv')
df3 = pd.read_csv('../data/concat_3.csv')
```

```
row_concat = pd.concat([df1, df2, df3])
print(row_concat)
```

```
     A    B    C    D
0   a0   b0   c0   d0
1   a1   b1   c1   d1
2   a2   b2   c2   d2
...
2  a10  b10  c10  d10
3  a11  b11  c11  d11
```

인덱스도 그대로 유지됩니다.

데이터 연결하기(2)

append 메서드로 데이터 연결

(연결할 데이터 프레임이 한 개일 때만!!)

일치하는 column 에 맞게 추가됨

```
new_row_df = pd.DataFrame([['n1', 'n2', 'n3', 'n4']], columns=['A', 'B', 'C', 'D'])
print(new_row_df)
```

```
df1.append(new_row_df)
```

데이터 연결하기(3)

ignore_index 인자 사용

원래의 인덱스를 무시하고 0부터 다시 지정

```
row_concat_i = pd.concat([df1, df2, df3], ignore_index=True)
print(row_concat_i)
```

```
    A   B   C   D
0  a0  b0  c0  d0
1  a1  b1  c1  d1
2  a2  b2  c2  d2
3  a3  b3  c3  d3
4  a4  b4  c4  d4
5  a5  b5  c5  d5
```

데이터 연결하기(4)

열 방향으로 데이터 연결하기

인자 axis = 1
default는 axis = 0이고 행으로 연결된다.

```
row_concat_i = pd.concat([df1, df2, df3], ignore_index=True)
print(row_concat_i)
```

```
    A   B   C   D
0  a0  b0  c0  d0
1  a1  b1  c1  d1
2  a2  b2  c2  d2
3  a3  b3  c3  d3
4  a4  b4  c4  d4
5  a5  b5  c5  d5
```

데이터 연결하기(5)

공통 열만 연결하기

```
row_concat = pd.concat([df1, df2, df3])
print(row_concat)
```

```
     A    B    C    D    E    F    G    H
0   a0   b0   c0   d0  NaN  NaN  NaN  NaN
1   a1   b1   c1   d1  NaN  NaN  NaN  NaN
2   a2   b2   c2   d2  NaN  NaN  NaN  NaN
3   a3   b3   c3   d3  NaN  NaN  NaN  NaN
0  NaN  NaN  NaN  NaN   a4   b4   c4   d4
1  NaN  NaN  NaN  NaN   a5   b5   c5   d5
2  NaN  NaN  NaN  NaN   a6   b6   c6   d6
3  NaN  NaN  NaN  NaN   a7   b7   c7   d7
0   a8  NaN   b8  NaN  NaN   c8  NaN   d8
1   a9  NaN   b9  NaN  NaN   c9  NaN   d9
2  a10  NaN  b10  NaN  NaN  c10  NaN  d10
3  a11  NaN  b11  NaN  NaN  c11  NaN  d11
```

데이터 연결하기(5)

공통 열만 연결하기

#inner join? 내부조인은 둘 이상의 데이터프레임에서 조건에 맞는 행을 연결하는 것입니다.

```
print(pd.concat([df1,df3], ignore_index=False, join='inner'))
```

```
    A   C
0  a0  c0
1  a1  c1
2  a2  c2
3  a3  c3
0  a8  b8
1  a9  b9
2 a10 b10
3 a11 b11
```

데이터 연결하기(5)

공통 인덱스만 연결하기

```
col_concat = pd.concat([df1, df2, df3], axis=1)
print(col_concat)
```

```
     A    B    C    D    E    F    G    H    A    C    F    H
0   a0   b0   c0   d0  NaN  NaN  NaN  NaN   a8   b8   c8   d8
1   a1   b1   c1   d1  NaN  NaN  NaN  NaN  NaN  NaN  NaN  NaN
2   a2   b2   c2   d2  NaN  NaN  NaN  NaN   a9   b9   c9   d9
3   a3   b3   c3   d3  NaN  NaN  NaN  NaN  NaN  NaN  NaN  NaN
4  NaN  NaN  NaN  NaN   a4   b4   c4   d4  NaN  NaN  NaN  NaN
5  NaN  NaN  NaN  NaN   a5   b5   c5   d5  a10  b10  c10  d10
6  NaN  NaN  NaN  NaN   a6   b6   c6   d6  NaN  NaN  NaN  NaN
7  NaN  NaN  NaN  NaN   a7   b7   c7   d7  a11  b11  c11  d11
```

```
print(pd.concat([df1, df3], axis=1, join='inner'))
```

```
    A   B   C   D   A   C   F   H
0  a0  b0  c0  d0  a8  b8  c8  d8
2  a2  b2  c2  d2  a9  b9  c9  d9
```

05-3 데이터 연결 마무리

merge 메서드 사용하기 (1)

1. 데이터 불러오기

```
person = pd.read_csv('../data/survey_person.csv')
site = pd.read_csv('../data/survey_site.csv')
survey = pd.read_csv('../data/survey_survey.csv')
visited = pd.read_csv('../data/survey_visited.csv')
```

merge 메서드 사용하기 (2)

2. merge 메서드 => default: 내부 조인
메서드를 사용한 데이터프레임 (site)이 merge된 DF의 왼쪽에 옴

```
o2o_merge = site.merge(visited_subset, left_on='name', right_on='site')
```

```
   name   lat    long
0   DR-1 -49.85 -128.57
1   DR-3 -47.15 -126.72
2  MSK-4 -48.87 -123.40

   ident  site       dated
0    619   DR-1  1927-02-08
2    734   DR-3  1939-01-07
6    837  MSK-4  1932-01-14

    name    lat    long  ident   site       dated
0   DR-1 -49.85 -128.57    619   DR-1  1927-02-08
1   DR-3 -47.15 -126.72    734   DR-3  1939-01-07
2  MSK-4 -48.87 -123.40    837  MSK-4  1932-01-14
```

merge 메서드 사용하기 (3)

3. left_on 과 right_on에 여러 개의 값 전달 가능

```
ps_vs = ps.merge(vs, left_on=['ident', 'taken', 'quant', 'reading'],
                 right_on=['person', 'ident', 'quant', 'reading'])
```

indent – person
taken – indent
.
.
다음과 같이 대응