

03

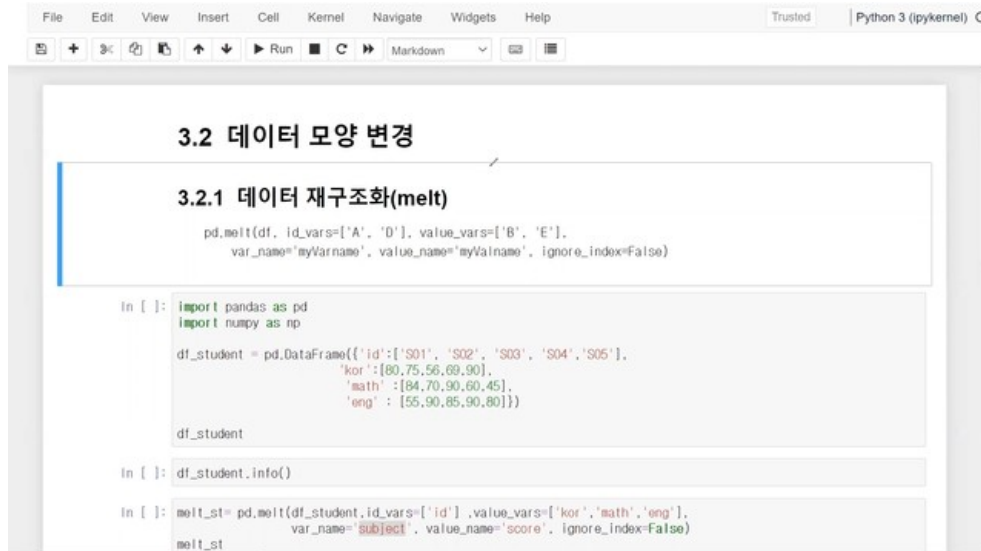
## 데이터 모양 변경 (pivot\_table)

[ melt, pivot table 설명 ]

# 테이블 재구성은 EDA, 특정 알고리즘 혹은 통계분석에서

데이터를 사용할 수 있도록 변환해주는 역할을 합니다.

00:25



```

3.2 데이터 모양 변경

3.2.1 데이터 재구조화(melt)

pd.melt(df, id_vars=['A', 'D'], value_vars=['B', 'E'],
        var_name='myVarname', value_name='myValname', ignore_index=False)

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

df_student = pd.DataFrame({'id': ['S01', 'S02', 'S03', 'S04', 'S05'],
                           'kor': [80, 75, 56, 69, 90],
                           'math': [84, 70, 90, 60, 45],
                           'eng': [55, 90, 85, 90, 80]})

df_student

In [ ]: df_student.info()

In [ ]: melt_st = pd.melt(df_student, id_vars=['id'], value_vars=['kor', 'math', 'eng'],
                        var_name='subject', value_name='score', ignore_index=False)
melt_st
  
```

[ melt 예제 ]

01:50

# value\_vars 를 variance라고 잘못 말씀드렸네요

value variavbles입니다.

File Edit View Insert Cell Kernel Navigate Widgets Help Trusted Python 3 (ipykernel)

3.2 데이터 모양 변경

3.2.1 데이터 재구조화(melt)

*Handwritten notes:* Pivot, 원문 - Melt -> Customizing, 녹여먹다, Pivot

```
pd.melt(df, id_vars='A', 'D', value_vars='B', 'E',
        var_name='myVarname', value_name='myValname', ignore_index=False)
```

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

df_student = pd.DataFrame({'id': ['S01', 'S02', 'S03', 'S04', 'S05'],
                           'kor': [80, 75, 56, 69, 90],
                           'math': [84, 70, 90, 60, 45],
                           'eng': [55, 90, 85, 90, 80]})

df_student

In [ ]: df_student.info()

In [ ]: melt_st = pd.melt(df_student, id_vars='id', value_vars=['kor', 'math', 'eng'],
                        var_name='subject', value_name='score', ignore_index=False)
melt_st
```

[ pivot table예제 ]

# 꼭 melt를 하지 않더라도 pivot\_table을 구성하실 수 있습니다.

하지만 melt 후에 재구성을 해주는 것이 편할 때가 많아요!

05:53

File Edit View Insert Cell Kernel Navigate Widgets Help Trusted Python 3 (ipykernel)

3.2.2 데이터 재구조화(pivot\_table)

```
pd.pivot_table(data, index, columns, values, aggfunc, margins)
```

```
In [ ]: pd.pivot_table(melt_st, index='id', columns='subject', values='score').head(10)

In [ ]: df_fruits = pd.read_csv('../data/Fruits.csv')
df_fruits

In [ ]: melt_fruits = pd.melt(df_fruits, ['Fruit'], value_vars=['Sales', 'Expenses', 'Profit'])
melt_fruits

In [ ]: melt_fruits.pivot_table(index='Fruit', columns='variable', values='value', aggfunc='mean', margins=False)

In [ ]: melt_fruits.pivot_table(index='Fruit', columns='variable', values='value', aggfunc='sum', margins=True)
```

3.2.3 데이터 행, 열 변경(transpose)

- 데이터의 행, 열 변경

```
In [ ]: df_fruits
```

다음 강의에서는 문자열 데이터를 핸들링 하는 방법을 배워볼 거예요. 문자열 데이터에서 에러가 나는 경우가 많습니다. 꼭 숙달해야 하는 부분이에요~

다음 강의에서 만나요~