1. 결측 처리

```
import pandas as pd
import numpy as np

df.dropna(column,how='all'/'any')

dic={column:value,column:value,....}
df.fillna(dic, method='bfill'/'ffill')

df[column]=df[column].fillna(df.groupby(column)[column].transfrom('func'))
```

2. datetime

```
from datetime import datetime

datetime.strptime(str, '%Y-%m-%d %H:%M:%S') # 글자 -> 시간
time_var.strftime('%Y-%m-%d %H:%M:%S')      # 시간 -> 글자

datetime.today()

pd.date_range(start_date, end_date, freq='D'/'MS'/'M'.....)

df[column].dt.year
df[column].dt.month
df[column].dt.day
df[column].dt.hour
df[column].dt.minute
df[column].dt.second
df[column].dt.dayofweek  # 0:월~ 6:일

#--------날짜 타입 변환 ------------
pd.to_datetime(df[column])
df[column].astype('datetime64[ns]')
```

3. 인코딩

```
# 1. LabelEncoder
from sklearn.preprocessing import LabelEncoder
le=LabelEncoder()
le.fit(df[column])
res=le.transform(df[column])

# 2. OneHotEncoder
from sklearn.preprocessing import OneHotEncoder
oe=OneHotEncoder()
oe.fit(df[column].values.reshape(-1,1))
res=oe.transform(df[column].values.reshape(-1,1))

oe.categores_  # Onehot카테고리 값

# 3. get_dummies
pd.get_dummies(df, columns=[column])

# 4. astype('category')
df[column].astype('category')
df[column].cat.codes        # 카테고리성 코드
df[column].cat.categories   # 원본 카테고리 보기
```

4. 바이닝

```
# 1. cut
pd.cut(data, range)

# 2. qcut
pd.qcut(data, 갯수)

# 3. nlargest
df[column].nlargest(n) # 상위 n개

# 4. nsmallest
df[column].nsmallest(n) # 하위 n개

# 5. 백분위수
pd.quantile(q=0.25)
np.percentile(df[column],q=25)
 = df[column].quantile(q=0.25)
```

4. scaling

```
# 1. StandardScaler
from sklearn.preprocessing import StandardScaler
sc=StandardScaler()
sc.fit(df[column].values.reshape(-1,1))
res=sc.transform(df[column].values.reshape(-1,1))

# 2. MinMaxScaler
from sklearn.preprocessing import MinMaxScaler
mm=MinMaxScaler()
mm.fit(df[column].values.reshape(-1,1))
res=mm.transform(df[column].values.reshape(-1,1))

# 3. RobustScaler
from sklearn.preprocessing import RobustScaler
rs=RobustScaler()
rs.fit(df[column].values.reshape(-1,1))
res=rs.transform(df[column].values.reshape(-1,1))
```