

Pandas_matplot_seaborn

```
In [1]: import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
import seaborn as sns
```

```
In [2]: # 한글폰트를 설정합니다.
# 마이너스 설정
from IPython.display import set_matplotlib_formats
set_matplotlib_formats('retina')
plt.rc('font',family='Malgun Gothic')
plt.rc('axes',unicode_minus=False)
```

```
In [3]: # 레티나 디스플레이로 폰트가 선명하게 표시되도록 합니다.
from IPython.display import set_matplotlib_formats
set_matplotlib_formats("retina")
```

```
In [4]: df = pd.read_csv('소상공인시장진흥공단_상가(상권)정보_서울_202012.csv', sep='|')
df.head()
```

Out[4]:

	상가업소 번호	상호 명	지점 명	상 권 업 종 대 분 류 코 드	상 권 업 종 대 분 류 명	상 권 업 종 중 분 류 코 드	상 권 업 종 중 분 류 명	상 권 업 종 소 분 류 코 드	상 권 업 종 소 분 류 명	표 준 업 분 류 코 드	...	건물관리번호
0	17163092	도전 최강 달인 왕만 두	NaN	Q	음 식	Q01	한 식	Q01A01	한 식/ 백 반/ 한 정 식	I56111	...	1174010200102170000018014
1	17120456	이때	NaN	Q	음 식	Q01	한 식	Q01A01	한 식/ 백 반/ 한 정 식	I56111	...	1144012400103900067027687
2	17175350	L.A.D	NaN	Q	음 식	Q01	한 식	Q01A01	한 식/ 백 반/ 한 정 식	I56111	...	1144012000103950112010755

	상가업소 번호	상호 명	지점 명	상권업종 중대분류코드	상권업종 중대분류명	상권업종 중분류코드	상권업종 중분류명	상권업종 소분류코드	상권업종 소분류명	표준 산업 분류 코드	...	건물관리번호
3	17175311	제이 씨에스 푸드	NaN	Q	음식	Q01	한식	Q01A01	한식/ 백반/ 한정식	I56111	...	1162010200101180033018722
4	22767534	BYC 상신점	상신 점	D	소매	D05	의복의류	D05A07	셔츠/ 내의/ 속옷	NaN	...	1171011300100360000022458

5 rows × 39 columns



모든 컬럼이 표시되도록 max_columns 의 수를 지정

- `pd.options.display.max_columns`

```
In [5]: pd.options.display.max_columns = 40
```

```
In [6]: df.head()
```

```
Out[6]:
```

	상가업소 번호	상호 명	지점 명	상권업종 대분류코드	상권업종 대분류명	상권업종 중분류코드	상권업종 중분류명	상권업종 소분류명	표준 분류코드	표준 업종명	시 도 코드	시 도 명	시 군 구 코드	시 군 구 명	행정동
0	17163092	도전 최강 달인 왕만 두	NaN	Q	음식	Q01	한식	Q01A01	한식/ 백반/ 한정식	I56111	11	서울 특별시	11740	강 동 구	117405
1	17120456	이때	NaN	Q	음식	Q01	한식	Q01A01	한식/ 백반/ 한정식	I56111	11	서울 특별시	11440	마 포 구	114407
2	17175350	L.A.D	NaN	Q	음식	Q01	한식	Q01A01	한식/ 백반/ 한정식	I56111	11	서울 특별시	11440	마 포 구	114406

	상가업소 번호	상호 명	지점 명	상권업종 대분류코드	상권업종 대분류명	상권업종 중분류코드	상권업종 중분류명	상권업종 소분류코드	상권업종 소분류명	표준 산업 분류 코드	표준 산업 분류 명	시 도 코드	시 도 명	시 군 구 코드	시 군 구 명	행정 동
3	17175311	제이씨에스푸드	NaN	Q	음식	Q01	한식	Q01A01	한식/백반/한정식	I56111	한식 음식 점업	11	서울 특별 시	11620	관악 구	116206
4	22767534	BYC상신점	상신점	D	소매	D05	의복의류	D05A07	셔츠/내의/속옷	NaN	NaN	11	서울 특별 시	11710	송파 구	117105

```
In [7]: df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 346577 entries, 0 to 346576
Data columns (total 39 columns):
#   Column                Non-Null Count  Dtype
---  -
0   상가업소번호          346577 non-null  int64
1   상호명                346576 non-null  object
2   지점명                58969 non-null   object
3   상권업종대분류코드    346577 non-null  object
4   상권업종대분류명      346577 non-null  object
5   상권업종중분류코드    346577 non-null  object
6   상권업종중분류명      346577 non-null  object
7   상권업종소분류코드    346577 non-null  object
8   상권업종소분류명      346577 non-null  object
9   표준산업분류코드      326327 non-null  object
10  표준산업분류명         326327 non-null  object
11  시도코드              346577 non-null  int64
12  시도명                346577 non-null  object
13  시군구코드            346577 non-null  int64
14  시군구명              346577 non-null  object
15  행정동코드            346577 non-null  int64
16  행정동명              346442 non-null  object
17  법정동코드            346577 non-null  int64
18  법정동명              346577 non-null  object
19  지번코드              346577 non-null  int64
20  대지구분코드          346577 non-null  int64
21  대지구분명            346577 non-null  object
22  지번본번지            346577 non-null  int64
23  지번부번지            278754 non-null  float64
24  지번주소              346577 non-null  object
25  도로명코드            346577 non-null  int64
26  도로명                346577 non-null  object
27  건물본번지            346577 non-null  int64
28  건물부번지            41957 non-null   float64
29  건물관리번호          346577 non-null  object
30  건물명                166570 non-null  object
31  도로명주소            346577 non-null  object
32  구우편번호            346577 non-null  int64
33  신우편번호            346575 non-null  float64
34  동정보                27395 non-null   object
35  층정보                216374 non-null  object
36  호정보                55894 non-null   object
37  경도                  346577 non-null  float64
38  위도                  346577 non-null  float64
dtypes: float64(5), int64(11), object(23)
memory usage: 103.1+ MB
```

결측치 확인

```
In [8]: df.isnull().sum()
```

```
Out[8]: 상가업소번호      0
상호명      1
지점명    287608
상권업종대분류코드      0
상권업종대분류명      0
상권업종중분류코드      0
상권업종중분류명      0
상권업종소분류코드      0
상권업종소분류명      0
표준산업분류코드    20250
표준산업분류명    20250
시도코드      0
시도명      0
시군구코드      0
시군구명      0
행정동코드      0
행정동명    135
법정동코드      0
법정동명      0
지번코드      0
대지구분코드      0
대지구분명      0
지번본번지      0
지번부번지    67823
지번주소      0
도로명코드      0
도로명      0
건물본번지      0
건물부번지    304620
건물관리번호      0
건물명    180007
도로명주소      0
구우편번호      0
신우편번호      2
동정보    319182
층정보    130203
호정보    290683
경도      0
위도      0
dtype: int64
```

missingno 로 결측치 시각화 하기

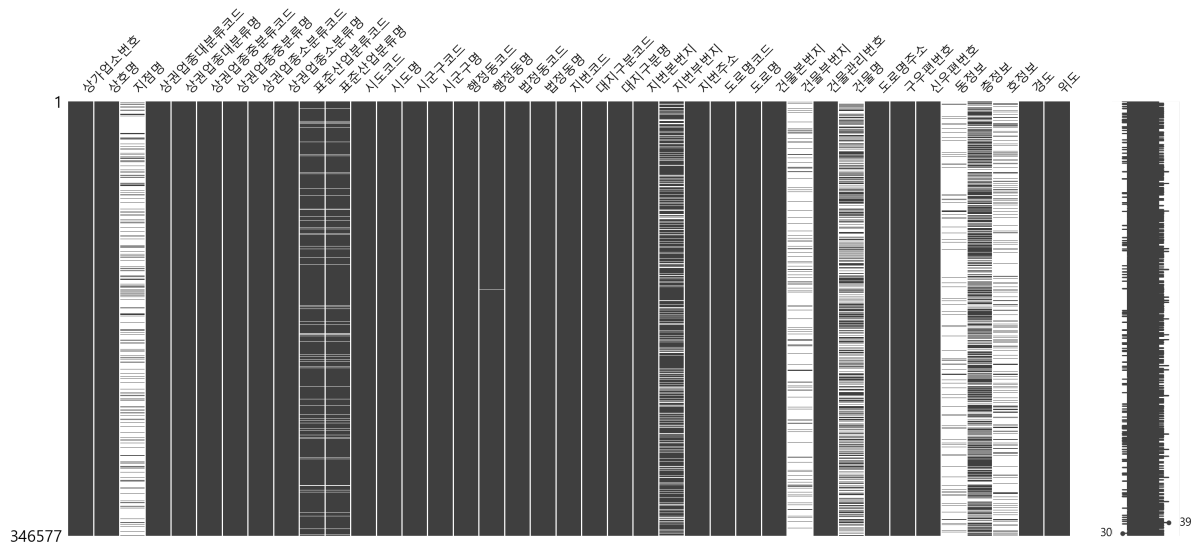
```
In [9]: #pip install missingno
```

```
In [10]: import missingno as msno
```

missingno.matrix()


```
In [11]: msno.matrix(df)
```

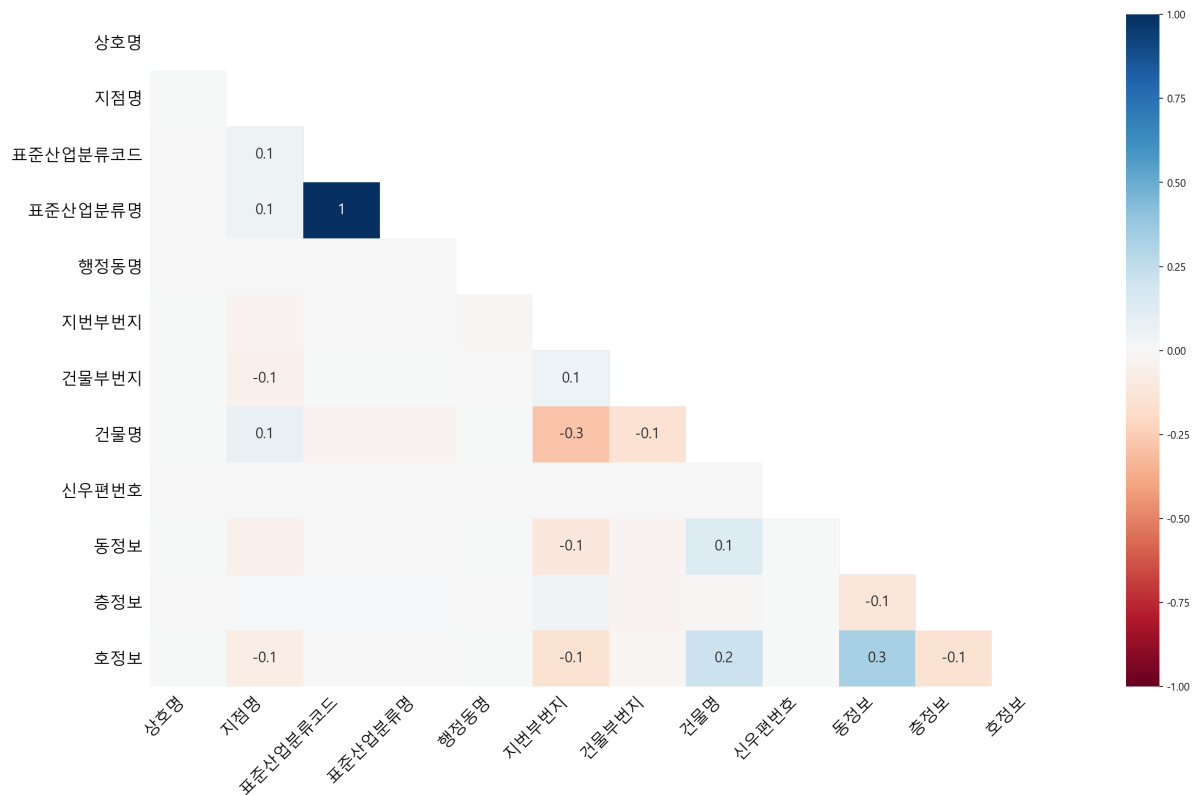
```
Out[11]: <AxesSubplot:>
```



missingno.heatmap

```
In [12]: msno.heatmap(df)
```

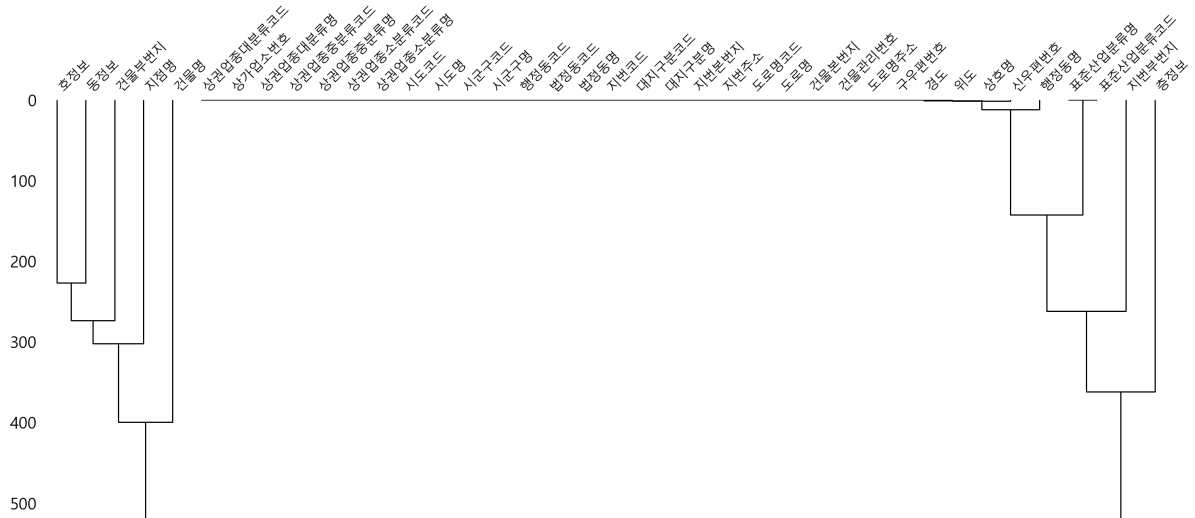
```
Out[12]: <AxesSubplot:>
```



missingno.dendrogram()

```
In [13]: msno.dendrogram(df)
```

```
Out[13]: <AxesSubplot:>
```



컬럼 제거

```
In [14]: cols = df.columns
cols
```

```
Out[14]: Index(['상가업소번호', '상호명', '지점명', '상권업종대분류코드', '상권업종대분류명',
'상권업종중분류코드',
'상권업종중분류명', '상권업종소분류코드', '상권업종소분류명', '표준산업분류코
드', '표준산업분류명', '시도코드',
'시도명', '시군구코드', '시군구명', '행정동코드', '행정동명', '법정동코드',
'법정동명', '지번코드',
'대지구분코드', '대지구분명', '지번본번지', '지번부번지', '지번주소', '도로명
코드', '도로명', '건물본번지',
'건물부번지', '건물관리번호', '건물명', '도로명주소', '구우편번호', '신우편번
호', '동정보', '총정보',
'호정보', '경도', '위도'],
dtype='object')
```

```
In [15]: cols_not = cols[cols.str.contains('코드|번호')]
cols_not
```

```
Out[15]: Index(['상가업소번호', '상권업종대분류코드', '상권업종중분류코드', '상권업종소분류코
드', '표준산업분류코드', '시도코드',
'시군구코드', '행정동코드', '법정동코드', '지번코드', '대지구분코드', '도로명
코드', '건물관리번호', '구우편번호',
'신우편번호'],
dtype='object')
```

```
In [16]: df = df.drop(cols_not, axis=1).copy()
```

```
In [17]: df.columns
```

```
Out[17]: Index(['상호명', '지점명', '상권업종대분류명', '상권업종중분류명', '상권업종소분류명', '표준산업분류명', '시도명', '시군구명', '행정동명', '법정동명', '대지구분명', '지번본번지', '지번부번지', '지번주소', '도로명', '건물본번지', '건물부번지', '건물명', '도로명주소', '동정보', '층정보', '호정보', '경도', '위도'], dtype='object')
```

```
In [18]: df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 346577 entries, 0 to 346576
Data columns (total 24 columns):
#   Column          Non-Null Count  Dtype
---  -
0   상호명          346576 non-null object
1   지점명          58969 non-null  object
2   상권업종대분류명 346577 non-null object
3   상권업종중분류명 346577 non-null object
4   상권업종소분류명 346577 non-null object
5   표준산업분류명  326327 non-null object
6   시도명          346577 non-null object
7   시군구명        346577 non-null object
8   행정동명        346442 non-null object
9   법정동명        346577 non-null object
10  대지구분명      346577 non-null object
11  지번본번지      346577 non-null int64
12  지번부번지      278754 non-null float64
13  지번주소        346577 non-null object
14  도로명          346577 non-null object
15  건물본번지      346577 non-null int64
16  건물부번지      41957 non-null  float64
17  건물명          166570 non-null object
18  도로명주소      346577 non-null object
19  동정보          27395 non-null  object
20  층정보          216374 non-null object
21  호정보          55894 non-null  object
22  경도            346577 non-null float64
23  위도            346577 non-null float64
dtypes: float64(4), int64(2), object(18)
memory usage: 63.5+ MB
```

```
In [19]: n = df.isnull().sum()
cols_not = n.sort_values(ascending=False).head(8)
cols_not = cols_not.index
df = df.drop(cols_not, axis=1).copy()
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 346577 entries, 0 to 346576
Data columns (total 16 columns):
#   Column                Non-Null Count  Dtype
---  -
0   상호명                346576 non-null  object
1   상권업종대분류명      346577 non-null  object
2   상권업종중분류명      346577 non-null  object
3   상권업종소분류명      346577 non-null  object
4   시도명                346577 non-null  object
5   시군구명              346577 non-null  object
6   행정동명              346442 non-null  object
7   법정동명              346577 non-null  object
8   대지구분명            346577 non-null  object
9   지번본번지            346577 non-null  int64
10  지번주소               346577 non-null  object
11  도로명                 346577 non-null  object
12  건물본번지            346577 non-null  int64
13  도로명주소             346577 non-null  object
14  경도                   346577 non-null  float64
15  위도                   346577 non-null  float64
dtypes: float64(2), int64(2), object(12)
memory usage: 42.3+ MB
```

```
In [20]: df.to_csv('seoul_store.csv', index=False)
```

```
In [22]: df = pd.read_csv("seoul_store.csv")
df
```

Out[22]:

상호명	상권업종대분류명	상권업종중분류명	상권업종소분류명	시도명	시군구명	행정동명	법정동명	대지구분명	지번본번지	지주소	도로명	건물본번지	도로주소	경도	위도
0	도전 달인만두	음식	한식	한식/백반/한정식	서울특별시	강동구	고덕2동	고덕동	대지	693	서울특별시강동구고덕로 서울특별시강동구고덕동 693	333	서울특별시강동구고덕로 333	127.159471	37.556197
1	이때	음식	한식	한식/백반/한정식	서울특별시	마포구	연남동	연남동	대지	390	서울특별시마포구동교로 38안길 7 서울특별시마포구연남동 390-67	7	서울특별시마포구동교로 38안길 7	126.924660	37.562176
2	L.A.D	음식	한식	한식/백반/한정식	서울특별시	마포구	서교동	서교동	대지	395	서울특별시마포구서교동 395-112	23	서울특별시마포구잔다리로 3안길 23	126.919845	37.550689

상호명	상권업종대분류명	상권업종중분류명	상권업종소분류명	시도명	시군구명	행정동명	법정동명	대지구분명	지번본번지	지주소	도로명	건물본번지	도로주소	경도	위도
3	제이씨에스푸드	음식	한식	한식/백반/한정식	서울특별시 관악구	서림동	신림동	대지	118	서울특별시 관악구 신림로 118-33	서울특별시 관악구 신림로 14길 3	3	서울특별시 관악구 신림로 14길 3	126.937790	37.471190
4	BYC상신점	소매	의복의류	셔츠/내의/속옷	서울특별시 송파구	거여2동	거여동	대지	36	서울특별시 송파구 거여동 36	서울특별시 송파구 오금로 527	527	서울특별시 송파구 오금로 527	127.147321	37.493054
...
346572	현대기림	소매	건강/미용식품	건강식품판매	서울특별시 성동구	성수1가1동	성수동1가	대지	276	서울특별시 성동구 성수동1가 276-17	서울특별시 성동구 똑섬로 366-72	366	서울특별시 성동구 똑섬로 366-72	127.051154	37.539464
346573	본현대미아점	소매	건강/미용식품	건강식품판매	서울특별시 성북구	길음2동	길음동	대지	20	서울특별시 성북구 길음동 20-1	서울특별시 성북구 동소문로 315	315	서울특별시 성북구 동소문로 315	127.028726	37.608392

상호명	상권업종대분류명	상권업종중분류명	상권업종소분류명	시도명	시군구명	행정동명	법정동명	대지구분명	지번본번지	지주소	도로명	건물본번지	도로주소	경도	위도
346574	오피스알파	소매	건강/미용식품	서울특별시	강북구	수유3동	수유동	대지	36	서울특별시강북구수유동36-37	서울특별시강북구노해로17길	62	서울특별시강북구노해로17길62-1	127.018733	37.640485
346575	클라인옥드백	소매	가방/신발/액세서리	서울특별시	구로구	구로5동	구로동	대지	573	서울특별시구로구로동573	서울특별시구로구로중양로	152	서울특별시구로구로중양로152	126.882409	37.501378
346576	피움테라피	생활서비스	이/미용/건강	서울특별시	노원구	중계2.3동	중계동	대지	504	서울시노원구중계동504-1	서울특별시노원구섬밭로	285	서울특별시노원구섬밭로285	127.062556	37.640913

346577 rows × 16 columns

describe 로 기술통계값확인

```
In [ ]: df.head()
```

```
In [ ]: df.describe()
```

```
In [ ]: df[['위도', '경도']].describe()
```

count:

- 결측치를 제외한 (NaN이 아닌) 값의 갯수를 계산

```
In [ ]: df['위도'].count()
```

quantile(q=0.25)

- 1사분위 수(25%)

```
In [ ]: df['위도'].quantile(q=0.25)
```

상관계수

- 좌표평면을 펼쳐놓고 x 축은 변인 x 를, y 축은 변인 y 를 나타내게 한 뒤 각각의 관찰값들을 산점도 형태로 찍어놓으면, 그 결과 두 변인이 어떤 관계가 있는지 시각적으로 나타나게 된다.

```
In [ ]: corr = df.corr()  
corr
```

```
In [ ]: sns.heatmap(corr)
```

빈도수 구하기

```
In [ ]: df['상권업종대분류명'].unique()
```

```
In [ ]: df['상권업종대분류명'].nunique()
```

```
In [ ]: df['상권업종대분류명'].value_counts()
```



```
In [ ]: ##### 업종 음식인 데이터 불러옴
df_food = df[df['상권업종대분류명'] == '음식'].copy()
df_food
```

```
In [ ]: food_seoul_gu = df_food.groupby(['시군구명', '상권업종중분류명'])['상호명'].count()
food_seoul_gu
```

```
In [ ]: food_seoul_gu.unstack().plot.bar(figsize=(12, 10))
```

reset_index()

- 데이터 프레임 형태로 표현하게 변경시켜줌

```
In [ ]: food_frame = food_seoul_gu.reset_index()
food_frame
```

rename

- 컬럼명 변경

```
In [ ]: food_frame = food_frame.rename(columns={'상호명': '상호수'})
food_frame
```

seaborn.barplot()

```
In [ ]: plt.figure(figsize=(15, 7))
plt.xticks(rotation=30)
sns.barplot(data=food_frame, x="시군구명", y="상호수")
```

서브플롯

- sns.catplot()

```
In [ ]: # 시군구별로 4개씩 출력
sns.catplot(data=food_frame, x="상권업종중분류명", y="상호수",
            kind="bar", col="시군구명", col_wrap=4)
```

```
In [ ]: ### 서울의 스타벅스와 투썸 위치 분석
```

```
In [ ]: df.head()
```

비교를 위한 컬럼값 통일작업

- Star / star / 스타 등 다양한 이름 하나로 통일 필요

```
In [ ]: df_seoul = df.copy()
df_seoul
```

```
In [ ]: df_seoul['상호명_lower'] = df['상호명'].str.lower()
df_seoul
```

```
In [ ]: df_seoul = df_seoul[~df_seoul['상호명'].isnull()]
df_seoul.head()
```

```
In [ ]: df_seoul[df_seoul['상호명_lower'].str.contains('starbuck|스타벅')].head()
```

```
In [ ]: df_seoul[df_seoul['상호명_lower'].str.contains('starbuck|스타벅')].head()
```

```
In [ ]: df_seoul[df_seoul['상호명_lower'].str.contains('twosome|투썸플')].head()
```

```
In [ ]: df_twostar_cafe = df_seoul[df_seoul['상호명_lower'].str.contains('twosome|투썸플|star
df_twostar_cafe.head()
```

```
In [ ]: df_twostar_cafe.loc[df_twostar_cafe['상호명_lower'].str.contains('twosome|투썸플'), '
df_twostar_cafe['cafe']
```

```
In [ ]: df_twostar_cafe['cafe'] = df_twostar_cafe['cafe'].fillna('스타벅스')
df_twostar_cafe.head(10)
```

상권이 음식이 아닌것은 제외 시켜줘야함

```
In [ ]: df_twostar_cafe['상권업종대분류명'].value_counts()
```

```
In [ ]: brand_count = df_twostar_cafe['cafe'].value_counts()
brand_count
```

```
In [ ]: df_twostar_cafe['cafe'].value_counts(normalize=True).plot.barh()
```

```
In [ ]: vs = sns.countplot(data=df_twostar_cafe, x="cafe")
for i, var in enumerate(brand_count.index):
    vs.text(x=i, y=brand_count[i], s=brand_count[i])
```

시군구별 빈도수

```
In [ ]: plt.figure(figsize=(12,10))
plt.xticks(rotation=30)
sns.countplot(data=df_twostar_cafe, x='시군구명', hue='cafe')
```

scatterplot()

- 좌표에 점을 찍어주는 플롯

```
In [ ]: sns.scatterplot(data=df_twostar_cafe,x='경도',y='위도',hue='cafe')
```

folium

- 지도활용

```
In [ ]: !pip install folium
```

```
In [ ]: import folium
```

```
In [ ]: latitude = df_twostar_cafe['위도'].mean()
longitude = df_twostar_cafe['경도'].mean()
```

```
In [ ]: m = folium.Map([latitude, longitude], zoom_start=11)
for i in df_twostar_cafe.index:
    sub_lat = df_twostar_cafe.loc[i, '위도']
    sub_long = df_twostar_cafe.loc[i, '경도']
    tooltip = df_twostar_cafe.loc[i, '상호명'] + ' / ' + df_twostar_cafe.loc[i, '도로명']

    folium.Marker([sub_lat, sub_long], tooltip=tooltip, icon=folium.Icon(color='purple'))
m
```

```
In [ ]: from folium.plugins import MarkerCluster

m = folium.Map([latitude, longitude], zoom_start=11)
marker_cluster = MarkerCluster().add_to(m)

for i in df_twostar_cafe.index:
    sub_lat = df_twostar_cafe.loc[i, '위도']
    sub_long = df_twostar_cafe.loc[i, '경도']
    tooltip = df_twostar_cafe.loc[i, '상호명'] + ' / ' + df_twostar_cafe.loc[i, '도로명']

    folium.Marker([sub_lat, sub_long], tooltip=tooltip, icon=folium.Icon(color='purple'))
m
```

```
In [ ]: m = folium.Map([latitude, longitude], zoom_start=11)
for i in df_twostar_cafe.index:
    sub_lat = df_twostar_cafe.loc[i, '위도']
    sub_long = df_twostar_cafe.loc[i, '경도']
    tooltip = df_twostar_cafe.loc[i, '상호명'] + ' / ' + df_twostar_cafe.loc[i, '도로명']

    color = 'hotpink'
    if df_twostar_cafe.loc[i, 'cafe'] == '투썸플레이스':
        color = 'blue'

    folium.CircleMarker([sub_lat, sub_long], tooltip=tooltip, color = color, radius=3).add_to(m)
```

In []: