안녕하세요^^

AIVLE '서울시 생활정보 기반 대중교통 수요 분석 ' 과정에 오신 여러분을 환영합니다.

- 본 과정에서는 실제 사례와 데이터를 기반으로 문제를 해결하는 전체 과정을 자기 주도형 실습으로 진행해볼 예정입니다.
- 앞선 교육과정을 정리하는 마음과 지금까지 배운 내용을 바탕으로 문제 해결을 해볼게요!
- 미니 프로젝트를 통한 문제 해결 과정 'A에서 Z까지', 지금부터 시작합니다!

데이터 분석부터 먼저 시작해보겠습니다.

"구별 등록인구 데이터" 를 확인해 보도록 하겠습니다

In [84]:
```
# 필요 라이브러리부터 설치합니다.
%pip install pandas seaborn
```

```
Requirement already satisfied: pandas in c:\users\user\anaconda3\lib\site-packages (2.0.3)
Requirement already satisfied: seaborn in c:\users\user\anaconda3\lib\site-packages (0.13.2)
Requirement already satisfied: python-dateutil>=2.8.2 in c:\users\user\anaconda3\lib\site-packages (from pandas) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in c:\users\user\anaconda3\lib\site-packages (from pandas) (2023.3.post1)
Requirement already satisfied: tzdata>=2022.1 in c:\users\user\anaconda3\lib\site-packages (from pandas) (2023.3)
Requirement already satisfied: numpy>=1.21.0 in c:\users\user\anaconda3\lib\site-packages (from pandas) (1.24.3)
Requirement already satisfied: matplotlib!=3.6.1,>=3.4 in c:\users\user\anaconda3\lib\site-packages (from seaborn) (3.8.3)
Requirement already satisfied: contourpy>=1.0.1 in c:\users\user\anaconda3\lib\site-packages (from matplotlib!=3.6.1,>=3.4->seaborn) (1.0.5)
Requirement already satisfied: cycler>=0.10 in c:\users\user\anaconda3\lib\site-packages (from matplotlib!=3.6.1,>=3.4->seaborn) (0.11.0)
Requirement already satisfied: fonttools>=4.22.0 in c:\users\user\anaconda3\lib\site-packages (from matplotlib!=3.6.1,>=3.4->seaborn) (4.25.0)
Requirement already satisfied: kiwisolver>=1.3.1 in c:\users\user\anaconda3\lib\site-packages (from matplotlib!=3.6.1,>=3.4->seaborn) (1.4.4)
Requirement already satisfied: packaging>=20.0 in c:\users\user\anaconda3\lib\site-packages (from matplotlib!=3.6.1,>=3.4->seaborn) (23.1)
Requirement already satisfied: pillow>=8 in c:\users\user\anaconda3\lib\site-packages (from matplotlib!=3.6.1,>=3.4->seaborn) (10.0.1)
Requirement already satisfied: pyparsing>=2.3.1 in c:\users\user\anaconda3\lib\site-packages (from matplotlib!=3.6.1,>=3.4->seaborn) (3.0.9)
Requirement already satisfied: six>=1.5 in c:\users\user\anaconda3\lib\site-packages (from python-dateutil>=2.8.2->pandas) (1.16.0)
Note: you may need to restart the kernel to use updated packages.
```

기본전제

- 처음에 제공되는 데이터는 '에이블러용' 폴더에 있습니다.

[기본 데이터]

- 1.3 seoul_people_202401.csv

[데이터 소개]

- 서울 시 주민 등록 데이터

[변수 소개]

- 한국인 / 등록 외국인 / 합계 / 세대수 / 고령인구수

1.데이터 불러오기

모든 미니 프로젝트의 시작은 '데이터 불러오기' 부터라고 할 수 있습니다.

- KeyPoint : 불러오고자 하는 데이터에 따라 자유롭게 변수로 지정할 수 있다.

데이터 프레임을 불러오고 변수로 저장(여기서는 CSV 기준으로 진행)

- csv : pd.read_csv("파일이름. csv")
- txt : pd.read_csv("파일이름. csv", sep="구분자")
- xlsx : pd.read_excel('파일이름.xlsx')
- pickle : pd.read_pickle("파일이름.pkl")
  [참고] pickle은 파이썬의 모든 객체를 파일로 저장할 수 있는 방법으로 DataFrame,List,Dict 등 모든 객체 저장 가능(특히 sklearn라이브러리를 통해 모델을 학습시키고, 저장할 때 많이 사용)

[실습문제1] 데이터 로딩

- 'seoul_people_202401.csv'파일을 'seoul_people' 변수에 저장하고 그 Shape을 확인하세요.
  - 데이터 파일 로딩시 참고 사항
    - 구분자(sep)는 '\t' 입니다
    - cp949 인코더를 사용해 주세요

In [85]:
```
# 아래에 실습코드를 작성하고 결과를 확인합니다.
import pandas as pd

seoul_people = pd.read_csv('1.3 seoul_people_202401.csv', sep='\t', encoding='cp949')
```

In [86]:
```
# 데이터 프레임의 Shape을 확인합니다.
seoul_people.shape
```

Out[86]: (28, 14)

2.기본 정보 확인 및 클렌징

- 데이터 클렌징 : 결측치, 이상치 등을 제거하여 데이터 분석 결과가 왜곡 되는 문제를 방지하기 위한 정제 과정

[실습문제2] 기본 정보 확인하기

- 'seoul_people' 데이터의 정보를 확인해보세요.
- 'describe', 'info', 'head' 등 전부 활용해 보겠습니다.

In [87]:
```
# 아래에 실습코드를 작성하고 결과를 확인합니다.
# info()
seoul_people.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 28 entries, 0 to 27
Data columns (total 14 columns):
 #   Column       Non-Null Count  Dtype 
---  ------       --------------  ----- 
 0   동별(1)        28 non-null     object
 1   동별(2)        28 non-null     object
 2   2024         28 non-null     object
 3   2024    .1   28 non-null     object
 4   2024    .2   28 non-null     object
 5   2024    .3   28 non-null     object
 6   2024    .4   28 non-null     object
 7   2024    .5   28 non-null     object
 8   2024    .6   28 non-null     object
 9   2024    .7   28 non-null     object
 10  2024    .8   28 non-null     object
 11  2024    .9   28 non-null     object
 12  2024    .10  28 non-null     object
 13  2024    .11  28 non-null     object
dtypes: object(14)
memory usage: 3.2+ KB
```

In [88]:
```
# 아래에 실습코드를 작성하고 결과를 확인합니다.
# describe()
seoul_people.describe().T
```

Out[88]:

	count	unique	top	freq
동별(1)	28	2	합계	26
동별(2)	28	27	동별(2)	2
2024	28	28	세대 (세대)	1
2024 .1	28	28	계 (명)	1
2024 .2	28	28	계 (명)	1
2024 .3	28	28	계 (명)	1
2024 .4	28	28	한국인 (명)	1
2024 .5	28	28	한국인 (명)	1
2024 .6	28	28	한국인 (명)	1
2024 .7	28	28	등록외국인 (명)	1
2024 .8	28	28	등록외국인 (명)	1
2024 .9	28	27	3225	2
2024 .10	28	23	1.97	2
2024 .11	28	28	65세이상고령자 (명)	1

In [89]:
```
# 아래에 실습코드를 작성하고 결과를 확인합니다.
# head()
seoul_people.head()
```

Out[89]:

	동별 (1)	동별 (2)	2024	2024 .1	2024 .2	2024 .3	2024 .4	2024 .5	2024 .6	2024 .7	2024 .8	2024 .9
0	동별 (1)	동별 (2)	세대 (세대)	계 (명)	계 (명)	계 (명)	한국인 (명)	한국인 (명)	한국인 (명)	등록외국인 (명)	등록외국인 (명)	등록외국인 (명)
1	동별 (1)	동별 (2)	소계	소계	남자	여자	소계	남자	여자	소계	남자	여자
2	합계	소계	4469417	9638799	4649446	4989353	9386034	4540031	4846003	252765	109415	143350
3	합계	종로구	72067	150453	71890	78563	139417	67306	72111	11036	4584	6452
4	합계	중구	64714	131793	63495	68298	121312	58659	62653	10481	4836	5645

In [90]:
```
# 아래에 실습코드를 작성하고 결과를 확인합니다.
# tail()
seoul_people.tail()
```

Out[90]:

	동별(1)	동별(2)	2024	2024.1	2024.2	2024.3	2024.4	2024.5	2024.6	2024.7	2024.8	2024.9	2024.10	
23	합계	관악구	284578	497883	249026	248857	481956	242651	239305	15927	6375	9552	1.69	8
24	합계	서초구	169884	412078	196391	215687	407664	194291	213373	4414	2100	2314	2.4	6
25	합계	강남구	239775	550282	262991	287291	544873	260520	284353	5409	2471	2938	2.27	8
26	합계	송파구	285927	660025	316981	343044	654166	314347	339819	5859	2634	3225	2.29	10
27	합계	강동구	203734	463318	226237	237081	459167	224423	234744	4151	1814	2337	2.25	8

[실습문제3] 데이터 확인 및 처리

- head 와 tail 을 보았을때, 어느 데이터만 가져와야 할지 생각 해 봅시다.
- 데이터가 세번째 줄부터 시작된다
- 서울시의 각 자치구별 남성, 여성 인구 수와 그 합계를 나타내는 데이터프레임 만들기

In [91]:
```
# 아래에 실습코드를 작성하고 결과를 확인합니다.
seoul_people = pd.read_csv('1.3 seoul_people_202401.csv', sep="\t", encoding = "cp949", header
seoul_people
```

Out[91]:

	동별(1)	동별(2)	소계	소계.1	남자	여자	소계.2	남자.1	여자.1	소계.3	남자.2	여자
0	합계	소계	4469417	9638799	4649446	4989353	9386034	4540031	4846003	252765	109415	14335
1	합계	종로구	72067	150453	71890	78563	139417	67306	72111	11036	4584	645
2	합계	중구	64714	131793	63495	68298	121312	58659	62653	10481	4836	564
3	합계	용산구	107825	227106	109826	117280	213151	102312	110839	13955	7514	644
4	합계	성동구	133089	284766	137620	147146	277361	134519	142842	7405	3101	430
5	합계	광진구	170077	351180	167562	183618	335554	161277	174277	15626	6285	934
6	합계	동대문구	172801	359873	174120	185753	341149	167346	173803	18724	6774	1195
7	합계	중랑구	188097	387470	189462	198008	382155	187372	194783	5315	2090	322
8	합계	성북구	196800	438168	208682	229486	425602	204171	221431	12566	4511	805
9	합계	강북구	143560	292977	141185	151792	288113	139514	148599	4864	1671	319
10	합계	도봉구	138261	309494	149675	159819	306948	148796	158152	2546	879	166
11	합계	노원구	217904	502925	241099	261826	498213	239117	259096	4712	1982	273
12	합계	은평구	215721	470869	223330	247539	466770	221725	245045	4099	1605	249
13	합계	서대문구	146845	320629	149879	170750	306231	145404	160827	14398	4475	992
14	합계	마포구	181090	375162	174073	201089	363697	169990	193707	11465	4083	738

	동별(1)	동별(2)	소계	소계.1	남자	여자	소계.2	남자.1	여자.1	소계.3	남자.2	여자
15	합계	양천구	180695	439252	214161	225091	436028	212835	223193	3224	1326	189
16	합계	강서구	274084	568826	272338	296488	563058	269822	293236	5768	2516	325
17	합계	구로구	184096	415651	204715	210936	392405	192341	200064	23246	12374	1087
18	합계	금천구	120381	241105	121592	119513	227481	114414	113067	13624	7178	644
19	합계	영등포구	190737	397800	195493	202307	374794	183726	191068	23006	11767	1123
20	합계	동작구	186675	389714	187623	202091	378769	183153	195616	10945	4470	647
21	합계	관악구	284578	497883	249026	248857	481956	242651	239305	15927	6375	955
22	합계	서초구	169884	412078	196391	215687	407664	194291	213373	4414	2100	231
23	합계	강남구	239775	550282	262991	287291	544873	260520	284353	5409	2471	293
24	합계	송파구	285927	660025	316981	343044	654166	314347	339819	5859	2634	322
25	합계	강동구	203734	463318	226237	237081	459167	224423	234744	4151	1814	233

In [92]:
```
# 아래에 실습코드를 작성하고 결과를 확인합니다.
# 동별(2),남자,여자,소계.1 데이터만 가져오기
seoul_people_dong = seoul_people[['동별(2)', '남자', '여자', '소계.1']]
```

In [93]:
```
seoul_people_dong
```

Out[93]:

	동별(2)	남자	여자	소계.1
0	소계	4649446	4989353	9638799
1	종로구	71890	78563	150453
2	중구	63495	68298	131793
3	용산구	109826	117280	227106
4	성동구	137620	147146	284766
5	광진구	167562	183618	351180
6	동대문구	174120	185753	359873
7	중랑구	189462	198008	387470
8	성북구	208682	229486	438168
9	강북구	141185	151792	292977
10	도봉구	149675	159819	309494
11	노원구	241099	261826	502925
12	은평구	223330	247539	470869
13	서대문구	149879	170750	320629
14	마포구	174073	201089	375162
15	양천구	214161	225091	439252
16	강서구	272338	296488	568826
17	구로구	204715	210936	415651
18	금천구	121592	119513	241105
19	영등포구	195493	202307	397800
20	동작구	187623	202091	389714
21	관악구	249026	248857	497883
22	서초구	196391	215687	412078
23	강남구	262991	287291	550282
24	송파구	316981	343044	660025
25	강동구	226237	237081	463318

In [94]:
```
# 아래에 실습코드를 작성하고 결과를 확인합니다.
# 첫 번째 행 제거
seoul_people_dong = seoul_people_dong.iloc[1:] # 첫번째 행을 제거
```

In [95]:
```
seoul_people_dong
```

Out[95]:

	동별(2)	남자	여자	소계.1
1	종로구	71890	78563	150453
2	중구	63495	68298	131793
3	용산구	109826	117280	227106
4	성동구	137620	147146	284766
5	광진구	167562	183618	351180
6	동대문구	174120	185753	359873
7	중랑구	189462	198008	387470
8	성북구	208682	229486	438168
9	강북구	141185	151792	292977
10	도봉구	149675	159819	309494
11	노원구	241099	261826	502925
12	은평구	223330	247539	470869
13	서대문구	149879	170750	320629
14	마포구	174073	201089	375162
15	양천구	214161	225091	439252
16	강서구	272338	296488	568826
17	구로구	204715	210936	415651
18	금천구	121592	119513	241105
19	영등포구	195493	202307	397800
20	동작구	187623	202091	389714
21	관악구	249026	248857	497883
22	서초구	196391	215687	412078
23	강남구	262991	287291	550282
24	송파구	316981	343044	660025
25	강동구	226237	237081	463318

In [96]:
```
# 아래에 실습코드를 작성하고 결과를 확인합니다.
# 동별(2)->자치구, 소계.1->합계로 이름 대체
seoul_people_dong.columns = ['자치구', '남자', '여자', '합계']
```

In [97]:
```
seoul_people_dong
```

Out[97]:

	자치구	남자	여자	합계
1	종로구	71890	78563	150453
2	중구	63495	68298	131793
3	용산구	109826	117280	227106
4	성동구	137620	147146	284766
5	광진구	167562	183618	351180
6	동대문구	174120	185753	359873
7	중랑구	189462	198008	387470
8	성북구	208682	229486	438168
9	강북구	141185	151792	292977
10	도봉구	149675	159819	309494
11	노원구	241099	261826	502925
12	은평구	223330	247539	470869
13	서대문구	149879	170750	320629
14	마포구	174073	201089	375162
15	양천구	214161	225091	439252
16	강서구	272338	296488	568826
17	구로구	204715	210936	415651
18	금천구	121592	119513	241105
19	영등포구	195493	202307	397800
20	동작구	187623	202091	389714
21	관악구	249026	248857	497883
22	서초구	196391	215687	412078
23	강남구	262991	287291	550282
24	송파구	316981	343044	660025
25	강동구	226237	237081	463318

In [98]:
```
# 해당 데이터프레임을 csv 파일로 저장하세요.

seoul_people_dong.to_csv('seoul_people_dong.csv', index=False)
```

3.데이터 분석하기

- KeyPoint : 데이터의 형태를 살펴보고 다양한 분석기법을 통해 모델링에 적합하도록 정제요소를 선별할 수 있다.
  - 데이터들의 패턴 탐색

- 변수들간의 관계 파악

In [100…
```
# 시각화 한글폰트 설정
import seaborn as sns
import matplotlib.pyplot as plt

plt.rc('font', family='Malgun Gothic')
sns.set(font="Malgun Gothic",#"NanumGothicCoding",
        rc={"axes.unicode_minus":False}, # 마이너스 부호 깨짐 현상 해결
        style='darkgrid')
```

[실습문제4] 데이터 분포 알아보기

- 다양한 변수를 기준으로 그래프를 그려보고 인사이트를 도출해보세요.

In [103…
```
# 아래에 실습코드를 작성하고 결과를 확인합니다.

seoul_peoples = pd.read_csv('seoul_people_dong.csv')
```

In [104…
```
seoul_peoples
```

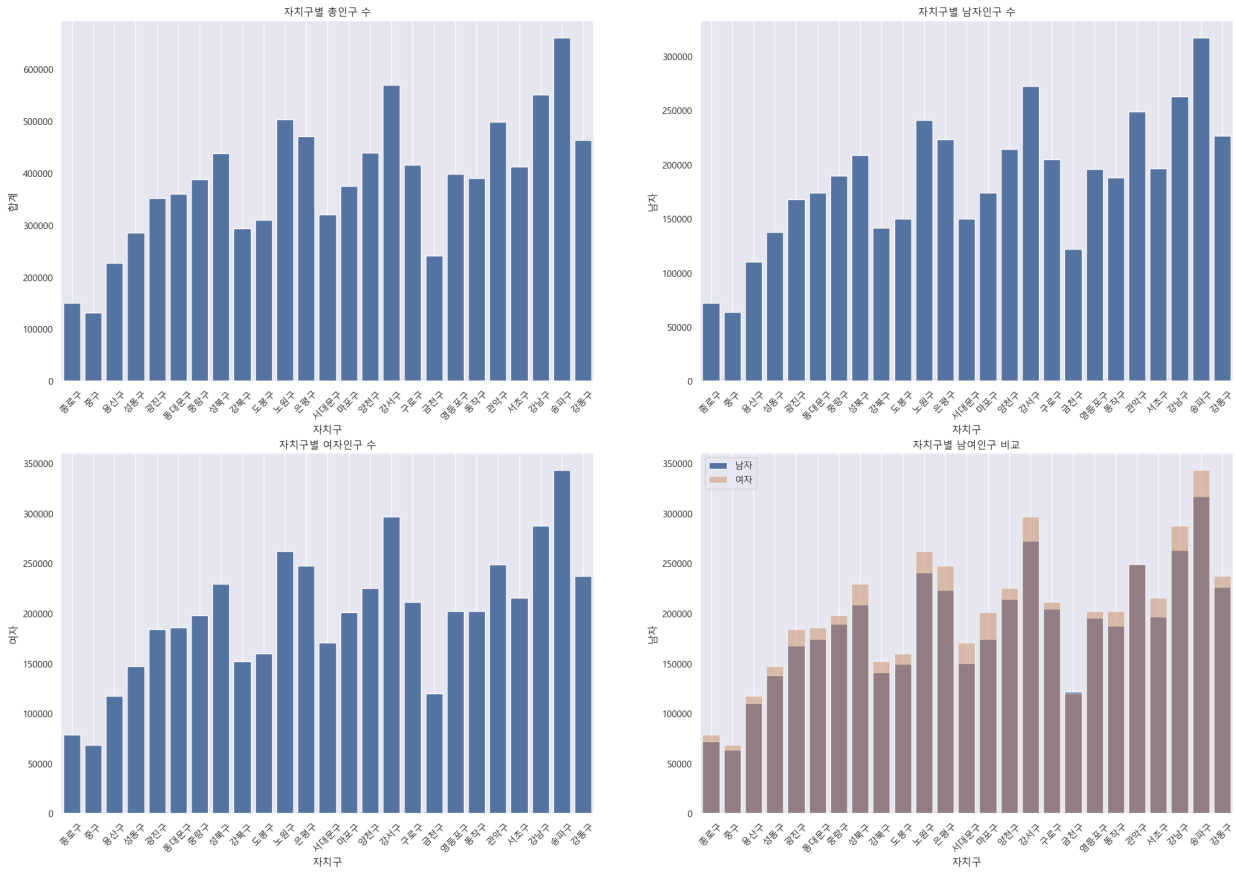
Out[104]:

	자치구	남자	여자	합계
0	종로구	71890	78563	150453
1	중구	63495	68298	131793
2	용산구	109826	117280	227106
3	성동구	137620	147146	284766
4	광진구	167562	183618	351180
5	동대문구	174120	185753	359873
6	중랑구	189462	198008	387470
7	성북구	208682	229486	438168
8	강북구	141185	151792	292977
9	도봉구	149675	159819	309494
10	노원구	241099	261826	502925
11	은평구	223330	247539	470869
12	서대문구	149879	170750	320629
13	마포구	174073	201089	375162
14	양천구	214161	225091	439252
15	강서구	272338	296488	568826
16	구로구	204715	210936	415651
17	금천구	121592	119513	241105
18	영등포구	195493	202307	397800
19	동작구	187623	202091	389714
20	관악구	249026	248857	497883
21	서초구	196391	215687	412078
22	강남구	262991	287291	550282
23	송파구	316981	343044	660025
24	강동구	226237	237081	463318

In [105…

```
plt.figure(figsize=(25, 17))

plt.subplot(2, 2, 1)
plt.title('자치구별 총인구 수')
sns.barplot(x='자치구', y='합계', data=seoul_peoples)
plt.xticks(rotation=45)
plt.grid()

plt.subplot(2, 2, 2)
plt.title('자치구별 남자인구 수')
sns.barplot(x='자치구', y='남자', data=seoul_peoples)
plt.xticks(rotation=45)
plt.grid()
```

```
plt.subplot(2, 2, 3)
plt.title('자치구별 여자인구 수')
sns.barplot(x='자치구', y='여자', data=seoul_peoples)
plt.xticks(rotation=45)
plt.grid()

plt.subplot(2, 2, 4)
plt.title('자치구별 남여인구 비교')
sns.barplot(x='자치구', y='남자', data=seoul_peoples, label='남자')
sns.barplot(x='자치구', y='여자', alpha=0.5, data=seoul_peoples, label='여자')
plt.xticks(rotation=45)
plt.legend()
plt.grid()


plt.show()
```



In [ ]:
```
# 위 차트를 통해 알게된 사실을 정리해봅시다.
# 1. 총 인구수, 남자의 수, 여자의 수가 가장 많은 지역은 송파구다 (강남구와 가까우며 강남3구에
# 2. 구별로 대체적으로 여자 인구가 많음 (아마 딸을 선호해서?) (공장일 및 3D업종은 대부분 남성이
# 3.
```