DAYS

데이터 분석 프로젝트

정보사회미디어 학과 20학번

정은진

INDEX

목 차 소 개

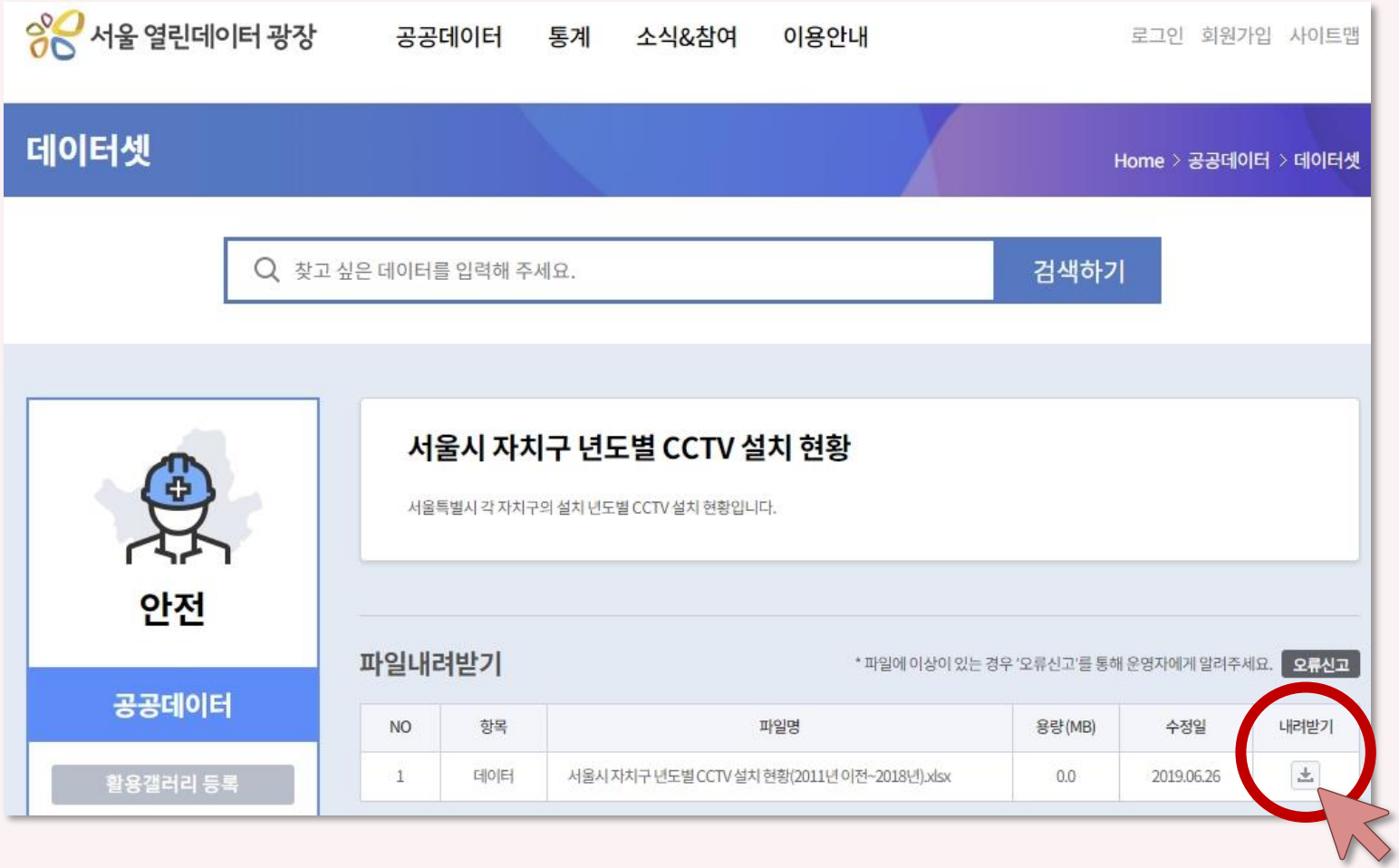
DAYS
정은진

주제 및 데이터 소개

❖ 주제: 서울시 인구 수와 CCTV 수 간의 상관관계 분석

주제 및 데이터 소개

❖ CCTV 데이터: CCTV_in_Seoul.csv



DAYS
정은진

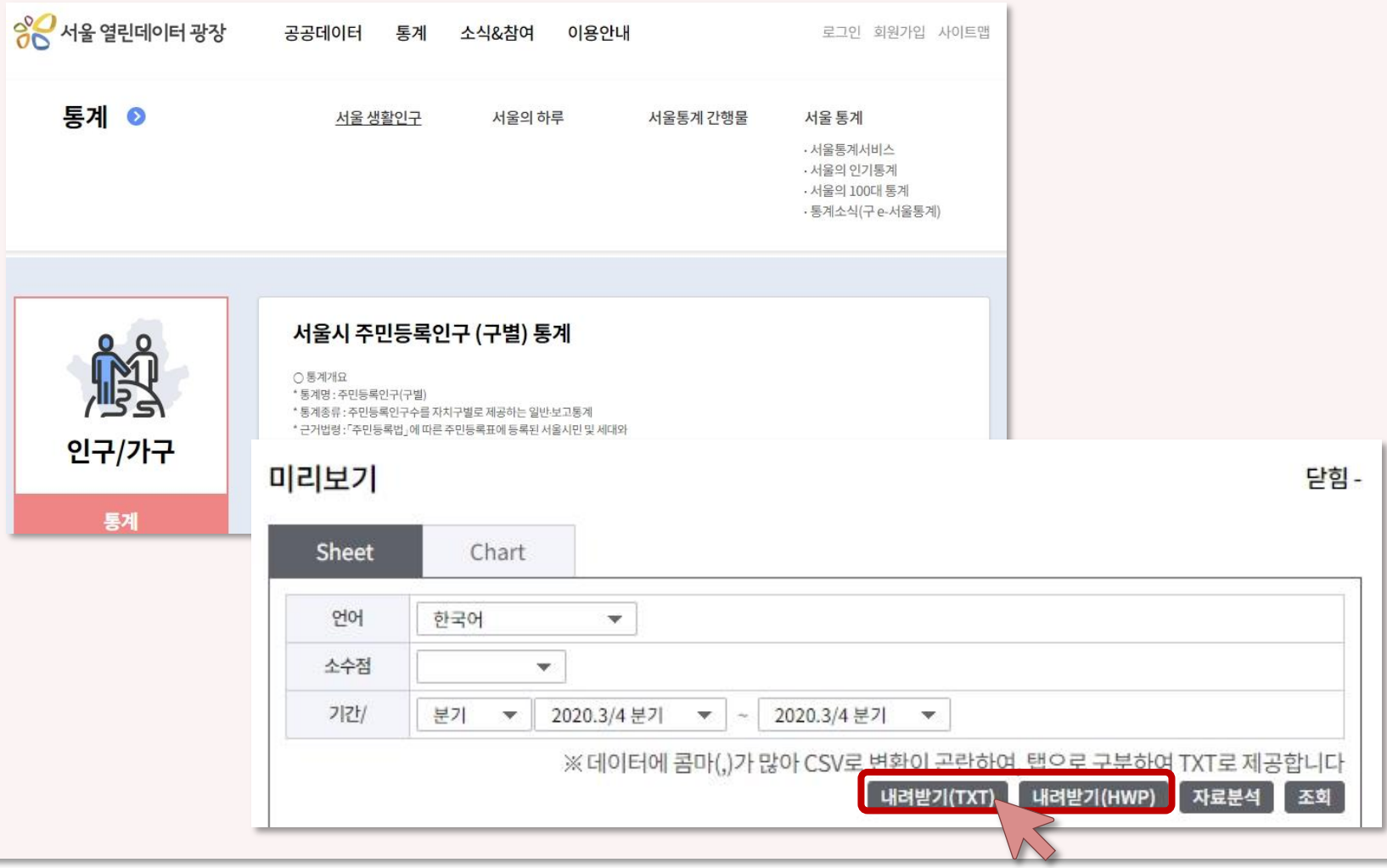
주제 및 데이터 소개

❖ CCTV 데이터: CCTV_in_Seoul.csv

	A	B	C	D	E	F
1	기관명	소계	2013년도	2014년	2015년	2016년
2	강남구	2780	1292	430	584	932
3	강동구	773	379	99	155	377
4	강북구	748	369	120	138	204
5	강서구	884	388	258	184	81
6	관악구	1496	846	260	390	613
7	광진구	707	573	78	53	174
8	구로구	1561	1142	173	246	323
9	금천구	1015	674	51	269	354
10	노원구	1265	542	57	451	516
11	도봉구	485	238	159	42	386
12	동대문구	1294	1070	23	198	579
13	동작구	1091	544	341	103	314
14	마포구	574	314	118	169	379
15	서대문구	962	844	50	68	292
16	서초구	1930	1406	157	336	398
17	성동구	1062	730	91	241	265
18	성북구	1464	1009	78	360	204
19	송파구	618	529	21	68	463
20	양천구	2034	1843	142	30	467
21	영등포구	904	495	214	195	373
22	용산구	1624	1368	218	112	398
23	은평구	1873	1138	224	278	468
24	종로구	1002	464	314	211	630
25	중구	671	413	190	72	348
26	중랑구	660	509	121	177	109

주제 및 데이터 소개

❖ 인구 데이터: population_in_Seoul.xls



DAYS
정은진

주제 및 데이터 소개

❖ 인구 데이터: population_in_Seoul.xls

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	기간	자치구	세대	인구	인구	인구	인구	인구	인구	인구	인구	인구	세대당인구	5세이상고령자
2	기간	자치구	세대	합계	합계	합계	한국인	한국인	한국인	등록외국인	등록외국인	등록외국인	세대당인구	5세이상고령자
3	기간	자치구	세대	계	남자	여자	계	남자	여자	계	남자	여자	세대당인구	5세이상고령자
4	2017.1/4	합계	4,202,888	10,197,604	5,000,005	5,197,599	9,926,968	4,871,560	5,055,408	270,636	128,445	142,191	2.36	1,321,458
5	2017.1/4	종로구	72,654	162,820	79,675	83,145	153,589	75,611	77,978	9,231	4,064	5,167	2.11	25,425
6	2017.1/4	중구	59,481	133,240	65,790	67,450	124,312	61,656	62,656	8,928	4,134	4,794	2.09	20,764
7	2017.1/4	용산구	106,544	244,203	119,132	125,071	229,456	111,167	118,289	14,747	7,965	6,782	2.15	36,231
8	2017.1/4	성동구	130,868	311,244	153,768	157,476	303,380	150,076	153,304	7,864	3,692	4,172	2.32	39,997
9	2017.1/4	광진구	158,960	372164	180992	191172	357211	174599	182612	14953	6393	8560	2.25	42214
10	2017.1/4	동대문구	159839	369496	182932	186564	354079	177021	177058	15417	5911	9506	2.22	54173
11	2017.1/4	중랑구	177548	414503	206102	208401	409882	204265	205617	4621	1837	2784	2.31	56774
12	2017.1/4	성북구	188512	461260	224076	237184	449773	219545	230228	11487	4531	6956	2.39	64692
13	2017.1/4	강북구	141554	330192	161686	168506	326686	160353	166333	3506	1333	2173	2.31	54813
14	2017.1/4	도봉구	136613	348646	171026	177620	346629	170289	176340	2017	737	1280	2.54	51312
15	2017.1/4	노원구	219957	569384	276823	292561	565565	275211	290354	3819	1612	2207	2.57	71941
16	2017.1/4	은평구	201869	494388	240220	254168	489943	238337	251606	4445	1883	2562	2.43	72334
17	2017.1/4	서대문구	137207	327163	156765	170398	314982	152613	162369	12181	4152	8029	2.3	48161
18	2017.1/4	마포구	169404	389649	185889	203760	378566	181346	197220	11083	4543	6540	2.23	48765
19	2017.1/4	양천구	176921	479978	237117	242861	475949	235278	240671	4029	1839	2190	2.69	52975
20	2017.1/4	강서구	247696	603772	294433	309339	597248	291249	305999	6524	3184	3340	2.41	72548
21	2017.1/4	구로구	172272	447874	224436	223438	416487	207114	209373	31387	17322	14065	2.42	56833
22	2017.1/4	금천구	105146	255082	130558	124524	236353	120334	116019	18729	10224	8505	2.25	32970
23	2017.1/4	영등포구	165462	402985	202573	200412	368072	183705	184367	34913	18868	16045	2.22	52413
24	2017.1/4	동작구	173033	412520	201217	211303	400456	195775	204681	12064	5442	6622	2.31	56013
25	2017.1/4	관악구	253826	525515	264763	260752	507203	256090	251113	18312	8673	9639	2	68082
26	2017.1/4	서초구	173856	450310	216264	234046	445994	214036	231958	4316	2228	2088	2.57	51733
27	2017.1/4	강남구	234107	570500	273301	297199	565550	270726	294824	4950	2575	2375	2.42	63167
28	2017.1/4	송파구	259883	667483	325040	342443	660584	321676	338908	6899	3364	3535	2.54	72506
29	2017.1/4	강동구	179676	453233	225427	227806	449019	223488	225531	4214	1939	2275	2.5	54622

데이터 전처리

❖ CCTV 데이터 불러오기

In [1]:
```
import pandas as pd
cctv = pd.read_csv('CCTV_in_Seoul.csv', encoding = 'utf-8')
cctv
```

Out[1]:

	기관명	소계	2013년도 이전	2014년	2015년	2016년
0	강남구	2780	1292	430	584	932
1	강동구	773	379	99	155	377
2	강북구	748	369	120	138	204
3	강서구	884	388	258	184	81
4	관악구	1496	846	260	390	613
5	광진구	707	573	78	53	174
6	구로구	1561	1142	173	246	323
7	금천구	1015	674	51	269	354
8	노원구	1265	542	57	451	516
9	도봉구	485	238	159	42	386
10	동대문구	1294	1070	23	198	579
11	동작구	1091	544	341	103	314
12	마포구	574	314	118	169	379
13	서대문구	962	844	50	68	292
14	서초구	1930	1406	157	336	398
15	성동구	1062	730	91	241	265
16	성북구	1464	1009	78	360	204
17	송파구	618	529	21	68	463
18	양천구	2034	1843	142	30	467
19	영등포구	904	495	214	195	373
20	용산구	1624	1368	218	112	398
21	은평구	1873	1138	224	278	468
22	종로구	1002	464	314	211	630
23	중구	671	413	190	72	348
24	중랑구	660	509	121	177	109

데이터 전처리

❖ CCTV 데이터 columns 이름 변경

In [1]:
```
import pandas as pd
cctv = pd.read_csv('CCTV_in_Seoul.csv', encoding = 'utf-8')
cctv
```

Out[1]:

	기관명	소계	2013년도 이전	2014년	2015년	2016년
0	강남구	2780	1292	430	584	932
1	강동구	773	379	99	155	377
2	강북구	748	369	120	138	204
3	강서구	884	388	258	184	81
4	관악구	1496	846	260	390	613

In [2]:
```
cctv.rename(columns = {cctv.columns[0] : '자치구',
                       cctv.columns[1] : 'CCTV수'}, inplace = True)
cctv.head()
```

Out[2]:

	자치구	CCTV수	2013년도 이전	2014년	2015년	2016년
0	강남구	2780	1292	430	584	932
1	강동구	773	379	99	155	377
2	강북구	748	369	120	138	204
3	강서구	884	388	258	184	81
4	관악구	1496	846	260	390	613

데이터 전처리

❖ 'CCTV수 증가율 ' 데이터 생성

$$*\text{CCTV 수 증가율} = \frac{2014\text{년} + 2015\text{년} + 2016\text{년마다 새로 설치된 CCTV 개수}}{2013\text{년도 이전에 설치된 CCTV 개수}} \times 100$$

```
In [3]: cctv['CCTV수 증가율'] = (cctv['2016년'] + cctv['2015년'] + cctv['2014년']) / cctv['2013년도 이전'] * 100
        cctv.head()
```

Out [3]:

	자치구	CCTV수	2013년도 이전	2014년	2015년	2016년	CCTV수 증가율
0	강남구	2780	1292	430	584	932	150.619195
1	강동구	773	379	99	155	377	166.490765
2	강북구	748	369	120	138	204	125.203252
3	강서구	884	388	258	184	81	134.793814
4	관악구	1496	846	260	390	613	149.290780

데이터 전처리

❖ CCTV 데이터 삭제

In [4]:
```
del cctv['2013년도 이전']
del cctv['2014년']
del cctv['2015년']
del cctv['2016년']
cctv.head()
```

Out[4]:

	자치구	CCTV수	CCTV수 증가율
0	강남구	2780	150.619195
1	강동구	773	166.490765
2	강북구	748	125.203252
3	강서구	884	134.793814
4	관악구	1496	149.290780

데이터 전처리

❖ 인구 데이터 선별하여 불러오기

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	기간	자치구	세대	인구	인구	인구	인구	인구	인구	인구	인구	인구	세대당인구	5세이상고령자
2	기간	자치구	세대	합계	합계	합계	한국인	한국인	한국인	등록외국인	등록외국인	등록외국인	세대당인구	5세이상고령자
3	기간	자치구	세대	계	남자	여자	계	남자	여자	계	남자	여자	세대당인구	5세이상고령자
4	2017.1/4	합계	4,202,888	10,197,604	5,000,005	5,197,599	9,926,968	4,871,560	5,055,408	270,636	128,445	142,191	2.36	1,321,458
5	2017.1/4	종로구	72,654	162,820	79,675	83,145	153,589	75,611	77,978	9,231	4,064	5,167	2.11	25,425
6	2017.1/4	중구	59,481	133,240	65,790	67,450	124,312	61,656	62,656	8,928	4,134	4,794	2.09	20,764
7	2017.1/4	용산구	106,544	244,203	119,132	125,071	229,456	111,167	118,289	14,747	7,965	6,782	2.15	36,231
8	2017.1/4	성동구	130,868	311,244	153,768	157,476	303,382	150,076	153,304	7,864	3,692	4,172	2.32	39,997

데이터 전처리

❖ 인구 데이터 선별하여 불러오기

```
In [5]: population = pd.read_excel('population_in_Seoul.xls',
                                   header = 2, # 엑셀의 세번째 행부터 읽어옴
                                   usecols = 'B, D, G, J, N') # B,D,G,J,N열을 읽어옴

        population
```

	자치구	계	계.1	계.2	65세이상고령자
0	합계	10197604.0	9926968.0	270636.0	1321458.0
1	종로구	162820.0	153589.0	9231.0	25425.0
2	중구	133240.0	124312.0	8928.0	20764.0
3	용산구	244203.0	229456.0	14747.0	36231.0
4	성동구	311244.0	303380.0	7864.0	39997.0
5	광진구	372164.0	357211.0	14953.0	42214.0
6	동대문구	369496.0	354079.0	15417.0	54173.0
7	중랑구	414503.0	409882.0	4621.0	56774.0
8	성북구	461260.0	449773.0	11487.0	64692.0
9	강북구	330192.0	326686.0	3506.0	54813.0
10	도봉구	348646.0	346629.0	2017.0	51312.0
11	노원구	569384.0	565565.0	3819.0	71941.0
12	은평구	494388.0	489943.0	4445.0	72334.0
13	서대문구	327163.0	314982.0	12181.0	48161.0
14	마포구	389649.0	378566.0	11083.0	48765.0
15	양천구	479978.0	475949.0	4029.0	52975.0
16	강서구	603772.0	597248.0	6524.0	72548.0
17	구로구	447874.0	416487.0	31387.0	56833.0
18	금천구	255082.0	236353.0	18729.0	32970.0
19	영등포구	402985.0	368072.0	34913.0	52413.0
20	동작구	412520.0	400456.0	12064.0	56013.0
21	관악구	525515.0	507203.0	18312.0	68082.0
22	서초구	450310.0	445994.0	4316.0	51733.0
23	강남구	570500.0	565550.0	4950.0	63167.0
24	송파구	667483.0	660584.0	6899.0	72506.0
25	강동구	453233.0	449019.0	4214.0	54622.0

데이터 전처리

❖ 인구 데이터 columns 이름 변경

	자치구	계	계.1	계.2	65세이상고령자
0	합계	10197604.0	9926968.0	270636.0	1321458.0
1	종로구	162820.0	153589.0	9231.0	25425.0
2	중구	133240.0	124312.0	8928.0	20764.0
3	용산구	244203.0	229456.0	14747.0	36231.0
4	성동구	311244.0	303380.0	7864.0	39997.0

In [6]:
```
population.rename(columns={population.columns[1] : '인구수',
                           population.columns[2] : '한국인',
                           population.columns[3] : '외국인',
                           population.columns[4] : '고령자'}, inplace = True)
population.head()
```

Out[6]:

	자치구	인구수	한국인	외국인	고령자
0	합계	10197604.0	9926968.0	270636.0	1321458.0
1	종로구	162820.0	153589.0	9231.0	25425.0
2	중구	133240.0	124312.0	8928.0	20764.0
3	용산구	244203.0	229456.0	14747.0	36231.0
4	성동구	311244.0	303380.0	7864.0	39997.0

데이터 전처리

❖ 두 데이터 병합, index 변경

```
In [7]: df = pd.merge(cctv, population, on='자치구')
        df.head()
```

Out [7]:

	자치구	CCTV수	CCTV수 증가율	인구수	한국인	외국인	고령자
0	강남구	2780	150.619195	570500.0	565550.0	4950.0	63167.0
1	강동구	773	166.490765	453233.0	449019.0	4214.0	54622.0
2	강북구	748	125.203252	330192.0	326686.0	3506.0	54813.0
3	강서구	884	134.793814	603772.0	597248.0	6524.0	72548.0
4	관악구	1496	149.290780	525515.0	507203.0	18312.0	68082.0

```
In [8]: df.set_index('자치구', inplace=True)
        df.head()
```

Out [8]:

자치구	CCTV수	CCTV수 증가율	인구수	한국인	외국인	고령자
강남구	2780	150.619195	570500.0	565550.0	4950.0	63167.0
강동구	773	166.490765	453233.0	449019.0	4214.0	54622.0
강북구	748	125.203252	330192.0	326686.0	3506.0	54813.0
강서구	884	134.793814	603772.0	597248.0	6524.0	72548.0
관악구	1496	149.290780	525515.0	507203.0	18312.0	68082.0

결측값 확인하기

❖ Info()함수를 이용하여 결측값 확인하기

```
In [9]: df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 25 entries, 강남구 to 중랑구
Data columns (total 6 columns):
 #   Column      Non-Null Count  Dtype
---  ------      --------------  -----
 0   CCTV수       25 non-null     int64
 1   CCTV수 증가율  25 non-null     float64
 2   인구수         25 non-null     float64
 3   한국인         25 non-null     float64
 4   외국인         25 non-null     float64
 5   고령자         25 non-null     float64
dtypes: float64(5), int64(1)
memory usage: 1.4+ KB
```

DAYS
정은진

결측값 확인하기

❖ Matplotlib을 위한 폰트 변경

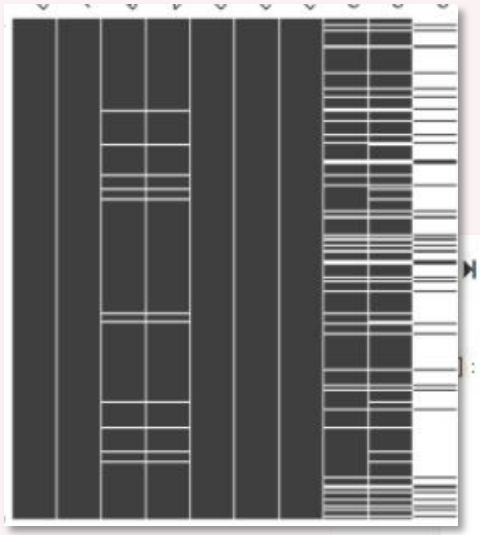
In [11]:

```
import platform
import matplotlib.pyplot as plt

from matplotlib import font_manager, rc
plt.rcParams['axes.unicode_minus'] = False
# Apple
if platform.system() == 'Darwin':
    rc('font', family='AppleGothic')
# Windows
elif platform.system() == 'Windows':
    path = 'c:/Windows/Fonts/malgun.ttf'
    font_name = font_manager.FontProperties(fname=path).get_name()
    rc('font', family=font_name)
else:
    print('Unknown system... Sorry.')
```

DAYS
정은진

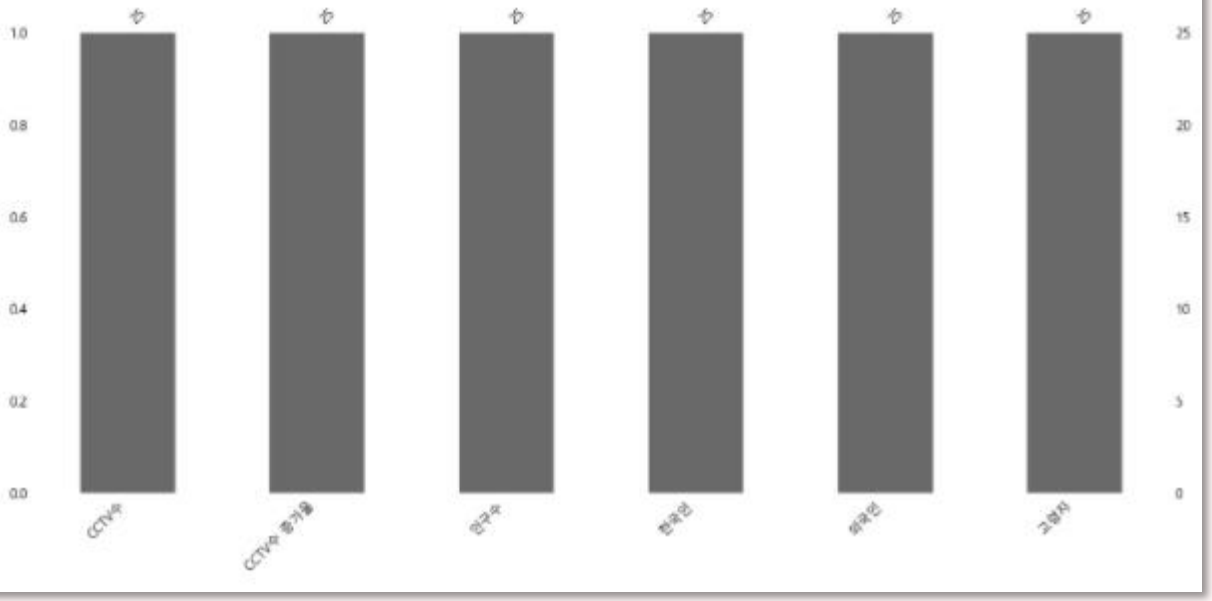
결측값 확인하기

❖ Missingno 모듈을 이용한 결측값 시각화



```
import missingno as msno

msno.bar(df)
```

```
<AxesSubplot:>
```



DAYS 정은진

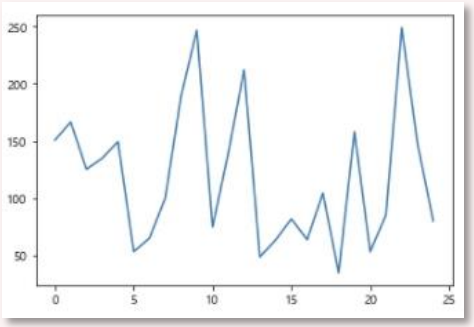
EDA

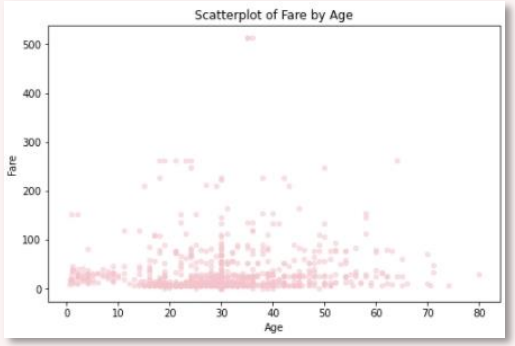
❖ EDA란?

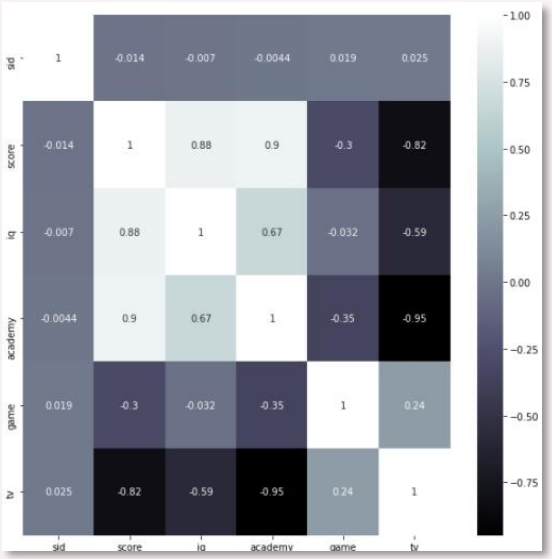
EDA는 *Exploratory Data Analysis*의 약자로, ' 탐색적 데이터 분석 ' 을 의미합니다.

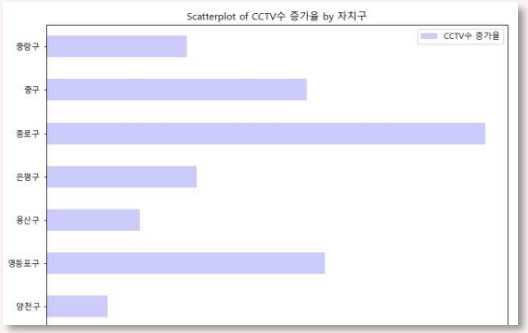
EDA는 단순 선형 회귀 분석 등을 위해 데이터 모델링에 들어가기에 앞서,
데이터의 분포나, 변수 간의 관계를 파악하기 위해 데이터를 시각화하는 과정을 말합니다.

EDA 과정에서는 히스토그램, 산점도, 상관관계표 등의 다양한 데이터 시각화 방법이 동원됩니다.



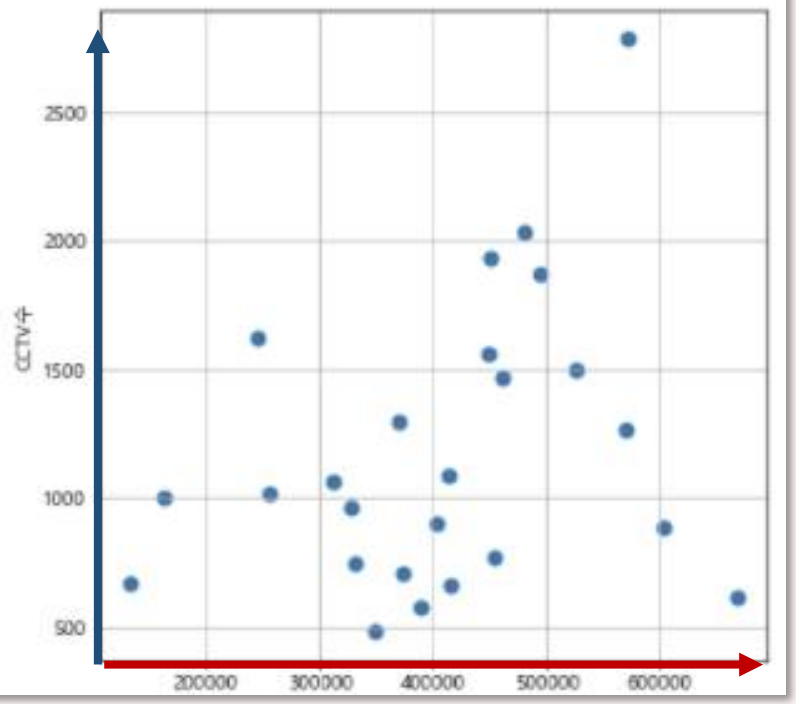






❖ 산점도를 이용하여 두 변수의 상관관계를 시각화하기

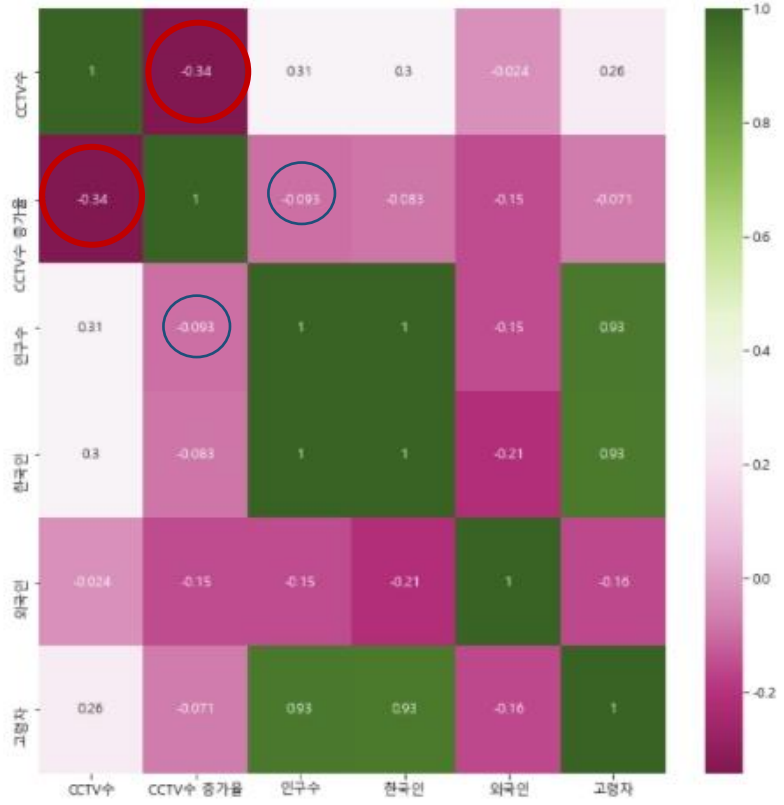
In [14]:
```
plt.figure(figsize=(6,6))
plt.scatter(df['인구수'], df['CCTV수'], s=50)
plt.xlabel('인구수')
plt.ylabel('CCTV수')
plt.grid()
plt.show()
```



인구수

❖ Heatmap 을 이용하여 상관계수를 시각화하기

```
In [14]: import seaborn as sns

         df.corr()

         plt.figure(figsize = (10, 10))
         sns.heatmap(df.corr(), cmap = 'PiYG', annot = True)

Out[14]: <AxesSubplot:>
```



* **상관계수:** 두 변수가 **'연관된'** 정도
두 변수가 ' **같이 일어나는** ' 강도
두 변수 간의 인과관계를 설명**X**
-1 ~ 1 사이의 값을 지님
절댓값이 높을수록 상관성이 높음
상관계수가 1이라는 것은 같은 변수라는 뜻
상관계수가 **0.4 이상**이면 **'상관성이 있다** ' 고 봄

* **인구수와 CCTV수 간의 상관계수: |0.34|**
* **인구수와 CCTV수 증가율 간의 상관계수: |0.093|**

가설 검정

❖ 가설 검정이란?

표본을 통해 모집단에 대한 가설의 옳고 그름을 판정하는 단계입니다.
가설 검정을 통해 **귀무 가설의 기각 여부를 확인합니다**.

- **귀무 가설:** 가설 검증에서, 표본에 의해 그 진위가 검증되어야 하는 가설입니다.
- **대립 가설:** 귀무 가설에 대립하는 가설입니다.

❖ 이번 분석에서 설정한 귀무가설

"인구수와 CCTV수는 상관관계가 없다"

❖ 대립가설

"인구수와 CCTV수는 상관관계가 있다"

```
In [16]: H0 = "인구수와 CCTV수는 상관관계가 없다"
         H1 = "인구수와 CCTV수는 상관관계가 있다"

         print("귀무가설은 ", H0)
         print("대립가설은 ", H1)

         귀무가설은  인구수와 CCTV수는 상관관계가 없다
         대립가설은  인구수와 CCTV수는 상관관계가 있다
```

DAYS
정은진

가설 검정

❖ 귀무 가설과 대립 가설, 유의 수준, 임계치 설정

평균이 0이고, 표준편차가 1인 정규분포표는 주로 **신뢰수준 95%, 신뢰수준 99%**를 기준으로 가설 검정을 진행합니다.

표본이 **신뢰수준 95%를 벗어나는 정도로** 귀무 가설이 틀렸다면, 귀무 가설을 기각한다.
(= 검정 통계량이 **유의수준 5%에 속한다면** 귀무 가설을 기각한다.)
(신뢰수준 95%의 **유의수준은 0.05, 임계치는 -1.96, 1.96**)

표본이 **신뢰수준 99%를 벗어나는 정도로** 귀무 가설이 틀렸다면, 귀무 가설을 기각한다.
(= 검정 통계량이 **유의수준 1%에 속한다면** 귀무 가설을 기각한다.)
(신뢰수준 95%의 **유의수준은 0.01, 임계치는 -2.58, 2.58**)

DAYS
정은진

가설 검정

❖ 검정 통계량 z 구하기

$$\text{검정 통계량 } z = \frac{\text{표본 평균} - \text{모평균}}{\text{표준 편차}/\text{루트}n}$$

❖ 모평균 구하기

시계열조회

지표명	공공기관 CCTV 설치 및 운영		
통계표명	공공기관 CCTV 설치 및 운영대수 초기화		
주기	연도	기간	2008 ~ 2019 조회

○ 통계표

[단위 : 대] 엑셀저장

	2008	2009	2010	2011	2012	2013	2014	2015	2016	2017	2018	2019
총 CCTV 설치대수(대)	157,197	241,415	309,227	364,302	461,746	565,723	655,030	739,232	845,136	954,261	1,032,879	1,148,770
전년대비 증가대수(대)	57,240	84,170	67,812	55,075	97,444	103,977	89,307	84,202	105,904	109,125	78,618	115,891
전년대비 증감비(%)	57.3	53.6	28.1	17.8	26.7	22.5	15.8	12.9	14.3	12.9	8.2	11.2

출처 : 실태조사 및 개인정보보호종합지원시스템 현황자료

주석 : 공공기관 CCTV 설치 및 운영현황은 2011년(2010년 자료)까지 공공기관별 조사 수합하였으나, 2012년(2011년 자료)부터 공공기관이 매년 3월 말까지 개인정보보호종합지원시스템에 현황을 등록함

가설 검정

❖ 검정 통계량 z 구하기

In [17]:

```
import numpy as np

# (전국CCTV수)모평균 구하기
#2016년 : 105904 / 2015년 : 84202, 2014년 : 89307
#2013년이전 : 256496
all_mean = (105904 + 84202 + 89307) / 256496 * 100
print(all_mean)

# 표본평균
cctv_mean = np.mean(df['CCTV수'])
print(cctv_mean)
# 표준편차
cctv_std = np.std(df['CCTV수'])
print(cctv_std)
# n
sample_n = 25

sample_error = cctv_mean - all_mean
SE = cctv_std / np.sqrt(sample_n)

# 검정 통계량 Z
z = sample_error / SE

print("검정통계량 Z는 ", z)
```

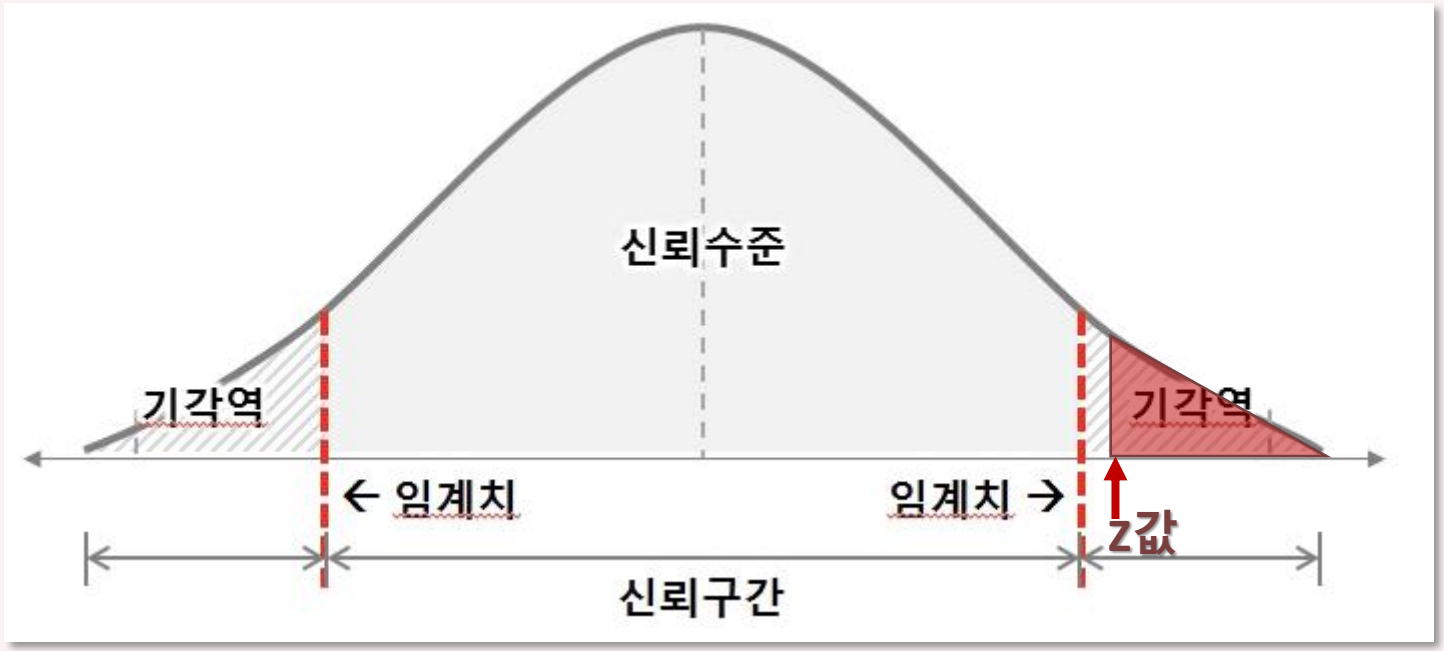
```
108.93464225562973
1179.08
545.4807728967172
검정통계량 Z는  9.8091941175257
```

가설 검정

❖ 검정 통계량과 임계치를 비교하여 가설 기각 여부 확인

```
In [18]: if abs(z) <= 1.96 :
             print("귀무가설 ", H0, " ", "-> 기각 실패")
         else :
             print("귀무가설 ", H0, " ", "-> 기각 성공")
```

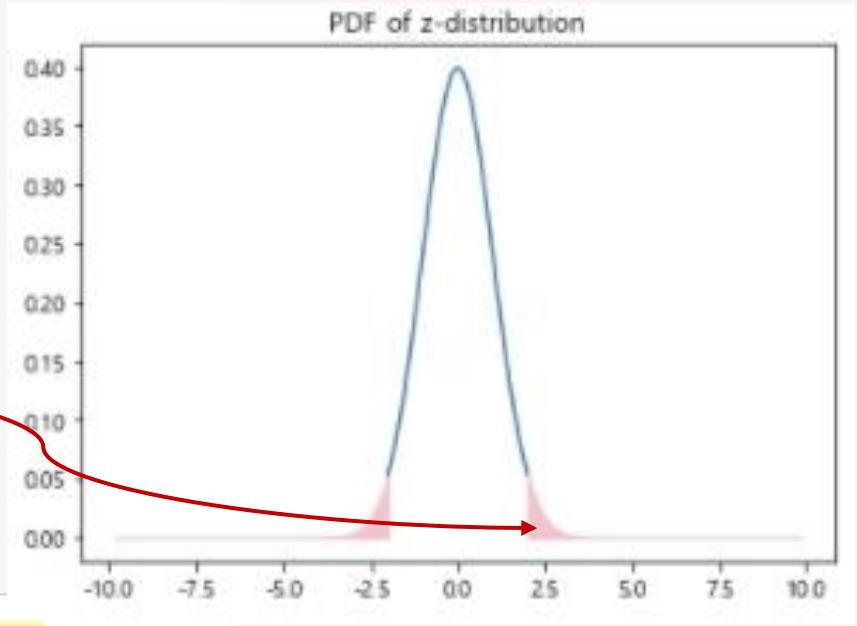
```
귀무가설  인구수와 CCTV수는 상관관계가 없다  -> 기각 성공
```



가설 검정

❖ P value 값과 유의 수준을 비교하여 가설 기각 여부 확인

In [19]:
```
from scipy import stats

# 정규분포 그리기
z_dist = stats.norm(loc = 0, scale = 1) # 평균이0이고 표준편차가 1인 정규분포
x = np.linspace(-2, 2, 100)
y = z_dist.pdf(x)

plt.plot(x, y)
plt.title("PDF of z-distribution")

# 오른쪽
right = np.linspace(abs(z), x[-1], 50)
plt.fill_between(right, z_dist.pdf(right), color = "pink")

# 왼쪽
left = np.linspace(-x[-1], -abs(z), 50)
plt.fill_between(left, z_dist.pdf(left), color = "pink")

p = z_dist.sf(abs(z))
p = p * 2
print("p value는", p)

if p >= 0.05 :
    print("결과:", H0, "-> 귀무가설 기각 실패")

else :
    print("결과:", H1, "-> 귀무가설 기각 성공")
```




```
p value는 1.0278613237537776e-22
결과: 인구수와 CCTV수는 상관관계가 없다. -> 귀무가설 기각 성공
```

DAYS
정은진

단순 선형회귀분석

❖ 단순 선형회귀분석이란?

종속변수와 독립변수 간에는 선형적인 관계가 성립합니다. (종속변수와 독립변수가 비례 / 혹은 반비례)

독립변수 간에는 상호 관련성이 없어야 합니다.

종속변수의 변동을 하나의 독립변수의 변동으로 **설명하는 분석 과정**입니다.

'종속변수의 변동(종속변수가 평균값과 얼마나 다른가)'을, 독립변수를 통해 설명할 수 있습니다.

독립변수와 종속변수 간의 관계를 잘 나타내는 직선을 찾아내는 분석입니다.

❖ 단순 선형회귀분석을 통해서...

두 변수(독립변수와 종속변수) 간에 인과관계가 존재하는지 확인할 수 있습니다.

인과 관계의 방향, 인과 관계의 정도와 데이터에 대한 수학적 모델을 확인할 수 있습니다.

❖ 이번 분석의 독립 변수와 종속 변수

독립변수: 인구수
종속변수: CCTV수

DAYS
정은진

단순 선형회귀분석

❖ Polyfit 함수를 이용하여 다항식 만들기

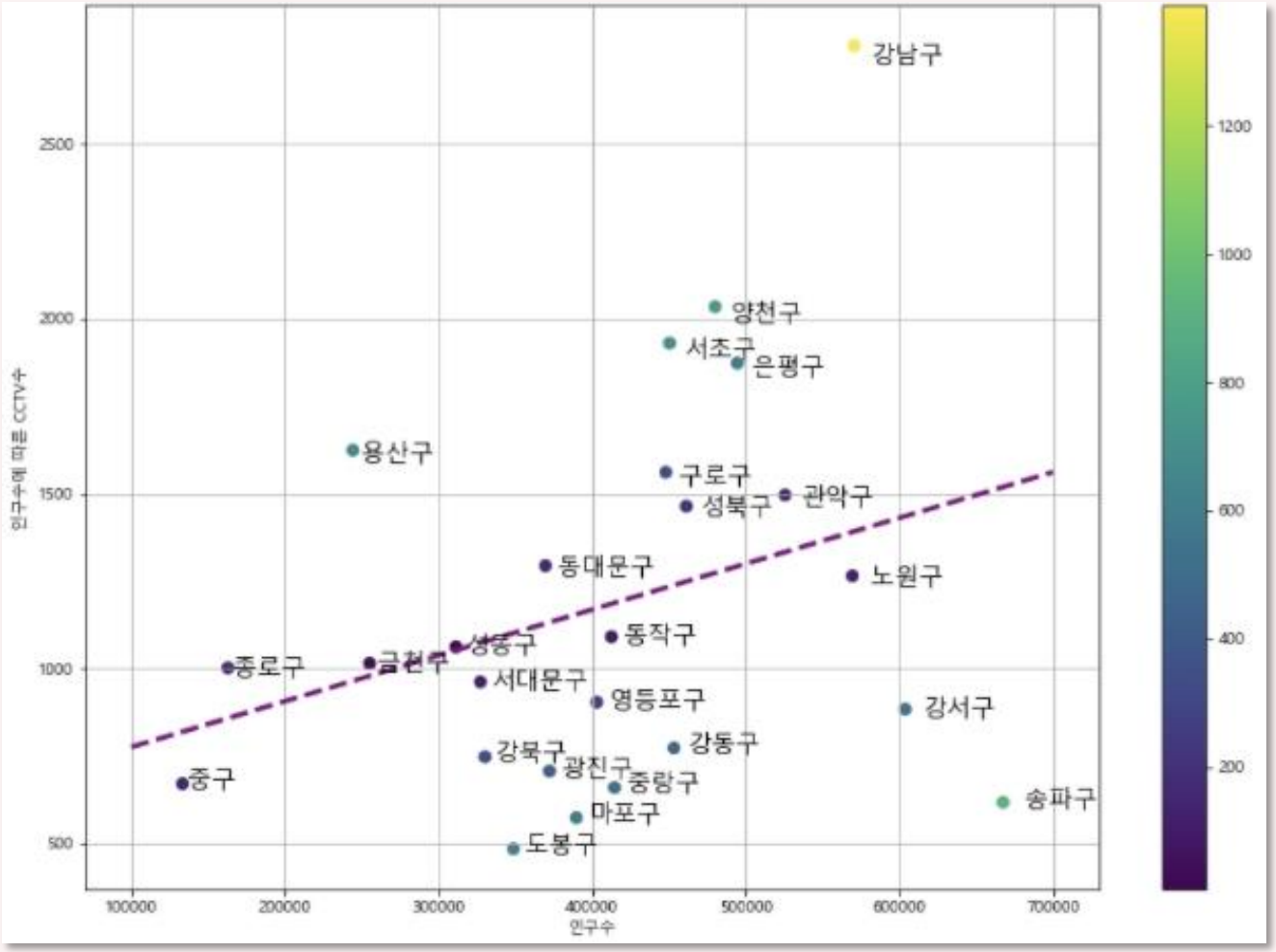
In [20]:
```
# 예측모델 만들기: x축-'인구수', y축-'CCTV수', 모델의 차수-'1차'
fp1 = np.polyfit(df['인구수'], df['CCTV수'], 1)

# poly1d()함수를 이용하여 fp1 모델을 다항식으로 변환하고, 그래프 그리기
f1 = np.poly1d(fp1)

# 10만부터 70만까지 100의 구간으로 나눠 그래프 그리기
fx = np.linspace(100000, 700000, 100)

# 오차값(종속변수의 변동정도)을 계산하여, df에 '오차' column을 추가
df['오차'] = np.abs(df['CCTV수'] - f1(df['인구수']))

# 오차값을 기준으로 내림차순 정렬
df_sort = df.sort_values(by='오차', ascending=False)
df_sort.head()
```

❖ 회귀 분석 그래프 그리기

In [21]:
```
plt.figure(figsize=(14,10))
plt.scatter(df['인구수'], df['CCTV수'],
            c=df['오차'], s=50)
plt.plot(fx, f1(fx), ls='dashed', lw=3, color='purple')

for n in range(25):
    plt.text(df_sort['인구수'][n]*1.02, df_sort['CCTV수'][n]*0.98,
             df_sort.index[n], fontsize=15
            )

plt.xlabel('인구수')
plt.ylabel('인구수에 따른 CCTV수')
plt.colorbar()
plt.grid()
plt.show()
```

단순 선형회귀분석

❖ 선형회귀분석에 따른 그래프



분석의 의의, 시사점

❖ 분석 결과의 의의, 그리고 이번 분석의 시사점

데이터를 전체적으로 살펴보면, 인구수와 CCTV수가 비례하나,
직선 아래에 있는 자치구는 인구수에 비해 CCTV수가 적다는 것을 알 수 있습니다.
그리고, 그러한 자치구의 수가 꽤 많습니다. (특히 송파구, 도봉구와 같은 경우에는 CCTV수가 다른 자치구에 비해 현저히 적습니다.)

각 자치구의 다양한 특성을 고려하면, 다른 구에 비해 CCTV수가 상대적으로 적은 이유가 있을 수 있겠지만,
그럼에도 불구하고,
범죄예방의 차원에서 직선 아래에 위치한 해당 자치구들은 CCTV를 증설해야 한다고 생각합니다.

회귀분석의 시각화를 통해 CCTV가 더 설치되어야 하는 자치구를 한 눈에 확인하여,
문제점을 파악하고, CCTV 증설의 필요성을 주장할 수 있었으므로,
이번 분석이 유의미한 결과를 도출하였다고 생각합니다.

감사합니다