DB 2차 프로젝트

DB연동 및 **시각화**

데이터분석과 구본성

CONTENTS

CONTENTS

분석주제

✓ **지역별 회원 수**

- ✓ 데이터를 시도, 광역시, 특별시 급으로 나누어
- ✓ 어느 지역의 회원이 가장 많은가에 대해 차트로 표현

01 EXCEL
지역별 회원 수

분석과정

✓ QUERY

```
SELECT ADDRESS, COUNT(ADDRESS) AS COUNT_ADDRESS FROM
(SELECT REPLACE(SUBSTR(ADDRESS1,1,2),'uC', '인천') AS ADDRESS FROM CUSTOMER)
GROUP BY ADDRESS
ORDER BY COUNT_ADDRESS;
```

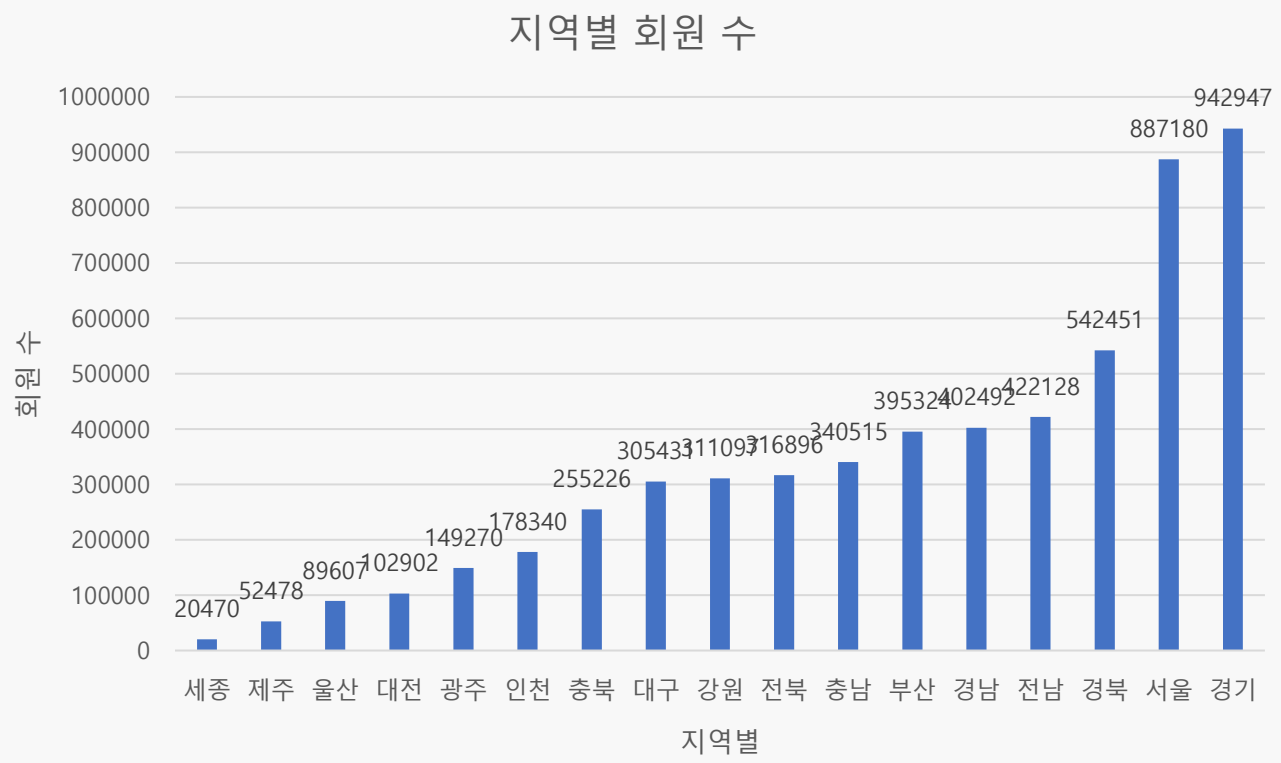
✓ CUSTOMER의 ADDRESS1 컬럼에 유니코드 값이 잘못 들어간 데이터 존재

→ REPLACE를 통해 제거

✓ 특별시, 광역시, 도에 따라 회원 수 데이터 추출

01 EXCEL
지역별 회원 수

- ✓ 서울, 경기권이 가장 뚜렷하게 많음
  - ✓ 서울 : 94만
  - ✓ 경기 : 89만
- ✓ 세종, 제주는 가장 적은 회원수를 보유하고 있음
  - ✓ 제주 : 5만
  - ✓ 세종 : 2만



분석주제

- ✓ **지역별 평균 신용 한도**
  - ✓ 데이터를 시도, 광역시, 특별시 급으로 나누어
  - ✓ 지역별로 평균 신용 한도의 차이가 얼마나 나는지 확인
  - ✓ 막대그래프를 통해 지역간 비교
- ✓ **지역별 가입 성비**
  - ✓ 데이터를 시도, 광역시, 특별시 급으로 나누어
  - ✓ 지역별로 가입자에 대한 성비가 어떻게 되는지 확인
  - ✓ 원그래프를 통해 지역간 비교

02 R
지역별 평균 신용 한도

분석과정

✓ QUERY

```
SELECT REGION, CREDIT_LIMIT FROM
(SELECT REPLACE(SUBSTR(ADDRESS1,1,2),'uC', '인천') AS REGION, CREDIT_LIMIT
FROM CUSTOMER);
```

✓ CUSTOMER의 ADDRESS1 컬럼에 유니코드 값이 잘못 들어간 데이터 존재

→ REPLACE를 통해 제거

분석과정

✓ R을 통한 전처리 (address_data : DB에서 가져온 데이터)

```
data<-address_data %>%
  group_by(REGION) %>% # 지역별 그룹핑
  summarise(count=n(), mean = mean(CREDIT_LIMIT)) %>% # count, 평균 계산
  arrange(mean) # 평균 순서대로 정렬
```

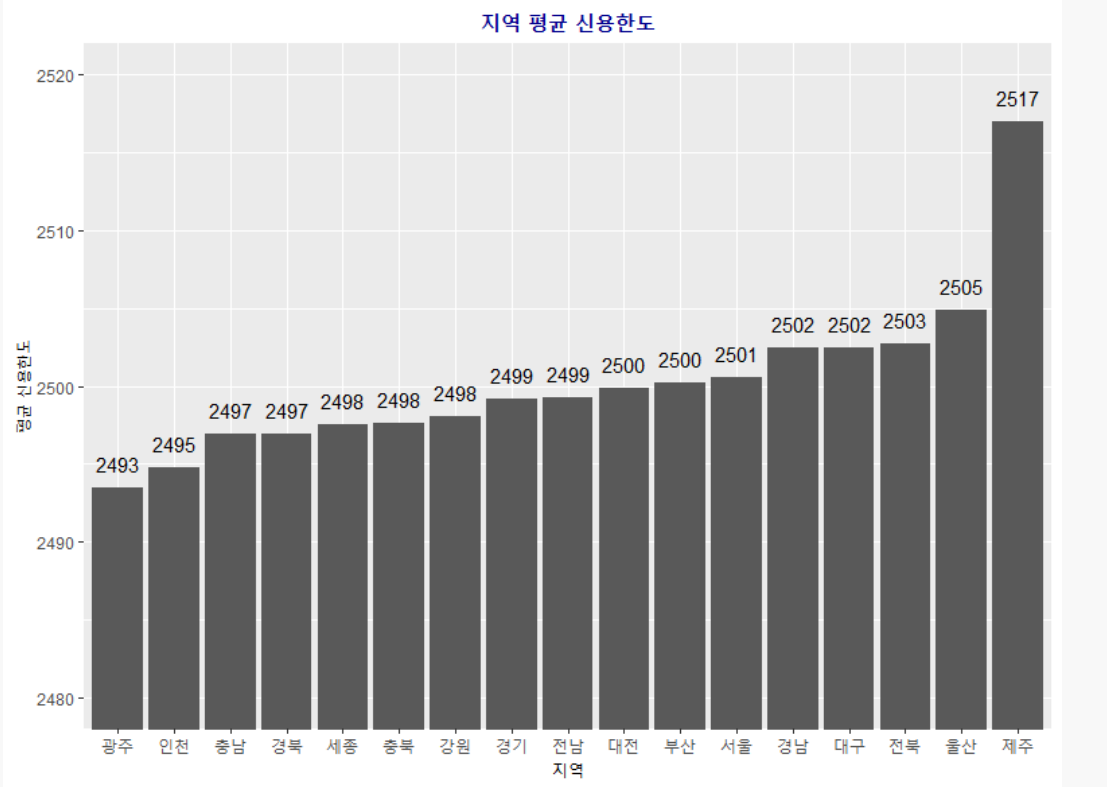
```
> head(data)
# A tibble: 6 x 3
  REGION  count  mean
  <chr>   <int> <dbl>
1 광주   149270 2493.
2 인천   178340 2495.
3 충남   340515 2497.
4 경북   542451 2497.
5 세종    20470 2498.
6 충북   255226 2498.
```

✓ ggplot2을 통한 시각화

→ 막대 그래프를 통한 시각화

02 R
지역별 평균 신용 한도

- ✓ 제주 가입자의 평균 신용한도가 뚜렷하게 높음
  - ✓ 제주 : 2517
- ✓ 나머지 지역에 대해서는 고른 분포를 보여주고 있음
  - ✓ 2493 ~ 2505 에 대해 분포하고 있음
  - ✓ 광주 : 2493 (최저 지역)



분석과정

✓ QUERY

```
SELECT REGION, GENDER FROM
(SELECT REPLACE(SUBSTR(ADDRESS1,1,2),'uC', '인천') AS REGION, GENDER  FROM CUSTOMER);
```

✓ CUSTOMER의 ADDRESS1 컬럼에 유니코드 값이 잘못 들어간 데이터 존재

→ REPLACE를 통해 제거

분석과정

✓ R을 통한 전처리 (gender_data : DB에서 가져온 데이터)

```
data2<-gender_data%>%
  group_by(REGION, GENDER)%>% # 지역, 성별에 대한 그룹핑
  summarise(n=n())%>% # 그룹에 대해 데이터의 개수 카운트
  mutate(freq=n/sum(n)) # 전체에 대한 데이터의 비율
```

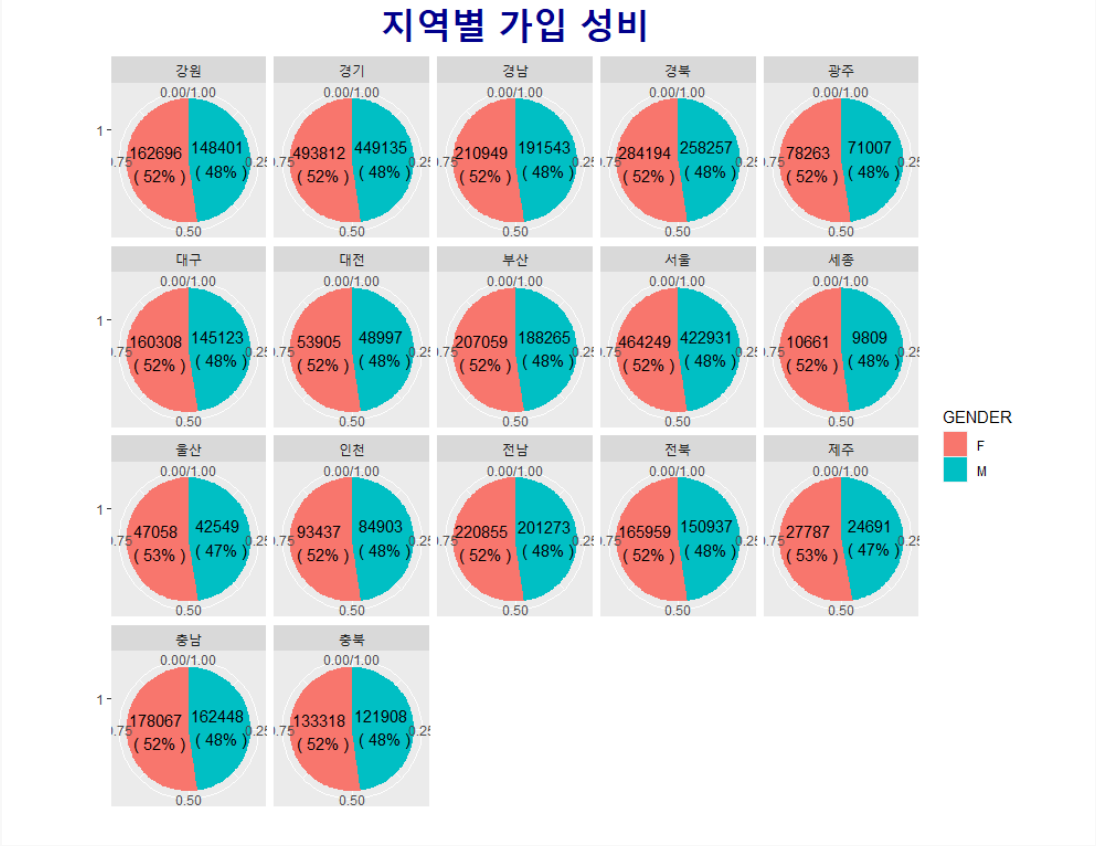
✓ ggplot2을 통한 시각화

→ 원 그래프를 통한 시각화

```
> data2
# A tibble: 34 x 4
# Groups:   REGION [17]
   REGION GENDER      n  freq
   <chr>  <chr>   <int> <dbl>
 1 강원   F      162696 0.523
 2 강원   M      148401 0.477
 3 경기   F      493812 0.524
 4 경기   M      449135 0.476
 5 경남   F      210949 0.524
 6 경남   M      191543 0.476
 7 경북   F      284194 0.524
 8 경북   M      258257 0.476
 9 광주   F       78263 0.524
10 광주   M       71007 0.476
# ... with 24 more rows
```

02 R
지역별 평균 신용 한도

- ✓ 지역별 원그래프 내에 절대 수치, 상대 수치 표현
- ✓ 지역별 고른 성비를 보여주고 있음
  - ✓ 남자 : 47% ~ 48%
  - ✓ 여자 : 52% ~ 53%
- ✓ 해당 데이터에 대해서는 의미 없는 그래프
  - ✓ 만약, 마케팅 결과 데이터라면
  - ✓ 어느 지역에서 더 관심을 가지는지 알 수 있음



분석주제

✓ **지역별 상세 정보**

- ✓ 지도에 지역별 회원 수, 평균 나이, 평균 신용한도, 성비 등 자세한 정보 표현
  → folium과 OpenStreetMap을 통한 표현
- ✓ 마커를 통한 위치 표현
- ✓ 버블맵을 통한 가장 회원수의 밀도가 높은 지역 표현

분석과정

✓ QUERY

```
query = '''SELECT ADDRESS, ROUND(AVG(CREDIT_LIMIT),2) AS AVG_CREDIT_LIMIT, ROUND(AVG(AGE),2) AS AVG_AGE,
    count(GENDER) AS COUNT_ALL,
    CONCAT(TO_CHAR(ROUND(count(case when GENDER = 'F' then 1 END)/count(GENDER),4)*100),'%') AS COUNT_FEMALE,
    CONCAT(TO_CHAR(ROUND(count(case when GENDER = 'M' then 1 END)/count(GENDER),4)*100),'%') AS COUNT_MALE,
    CASE WHEN COUNT(GENDER)<=20000 THEN '회원 수 적음'
        WHEN COUNT(GENDER)<=50000 THEN '회원 수 보통'
        WHEN COUNT(GENDER)>50000 THEN '회원 수 많음'
    END AS COUNT_STAT,
    CASE WHEN COUNT(GENDER)<=20000 THEN 'red'
        WHEN COUNT(GENDER)<=50000 THEN 'orange'
        WHEN COUNT(GENDER)>50000 THEN 'green'
    END AS COUNT_COLOR
FROM( SELECT REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(RTRIM(SUBSTR(ADDRESS1, 0, INSTR(ADDRESS1, ' ', 1, 2))),
    'uC778천 강화군', '인천 강화군'), '강원 홍천uAD70', '강원 홍천군'), '강원 삼uCC99시', '강원 삼척시'),  '충북 uCDA9주시','충북 충주시'),  '경기 안
uC591시','경기 안양시') AS ADDRESS, CREDIT_LIMIT, TRUNC((SYSDATE - BIRTH_DT) / 365) AS AGE, GENDER
    FROM CUSTOMER
) GROUP BY ADDRESS ORDER BY ADDRESS;'''
```

분석과정

✓ QUERY에 대한 결과를 데이터프레임으로 변환

```
df.head()
```

	ADDRESS	AVG_CREDIT_LIMIT	AVG_AGE	COUNT_ALL	COUNT_FEMALE	COUNT_MALE	COUNT_STAT	COUNT_COLOR
0	강원 강릉시	2497.27	42.08	33171.0	52.12%	47.88%	회원 수 보통	orange
1	강원 고성군	2483.74	42.00	12128.0	51.68%	48.32%	회원 수 적음	red
2	강원 동해시	2494.28	42.03	14226.0	51.83%	48.17%	회원 수 적음	red
3	강원 삼척시	2500.71	41.89	21882.0	52.5%	47.5%	회원 수 보통	orange
4	강원 속초시	2491.17	41.94	6575.0	52.03%	47.97%	회원 수 적음	red

✓ 각 컬럼에 대한 데이터를 마커에 표현하고자 함

분석과정

✓ 각 시군구 좌표를 가지고 있는 데이터와 병합

✓ 각 시군구에 대한 위도, 경도 데이터를 이용해 지도에 표현할 마커의 위치를 지정함

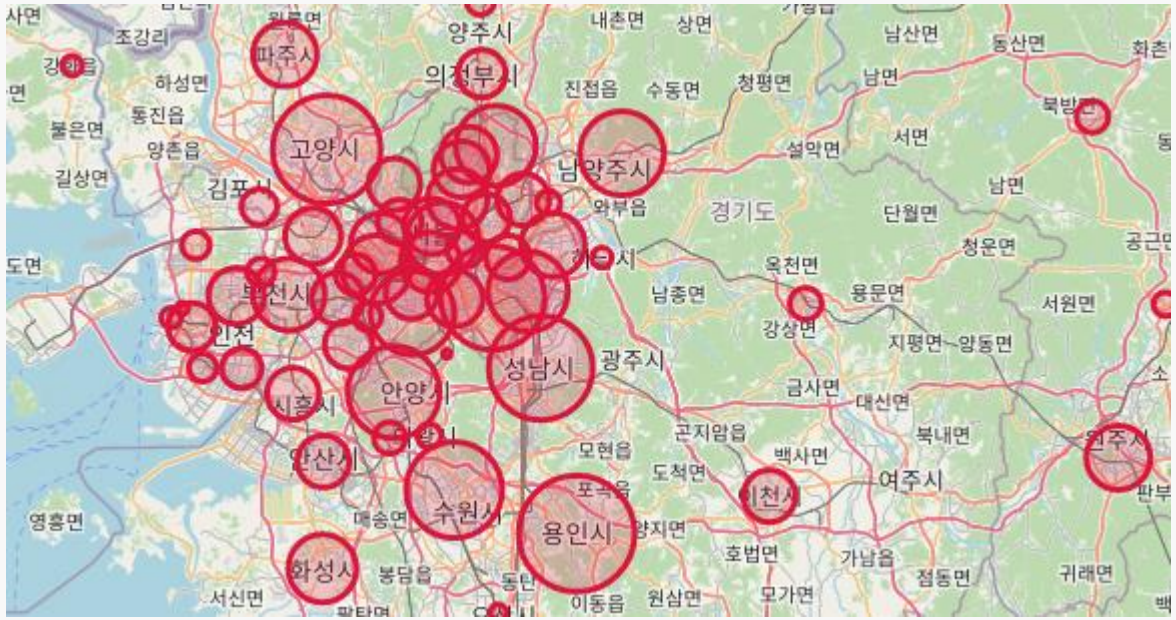
```
location[['ADDRESS','lat','long']]
```

	ADDRESS	lat	long
0	서울 강남구	37.4951	127.06278
1	서울 강동구	37.55274	127.14546
2	서울 강북구	37.6349	127.02015
3	서울 강서구	37.56227	126.81622
4	서울 관악구	37.47876	126.95235
...	...	...	...
223	전남 진도군	34.41018	126.1688
224	전남 곡성군	35.21449	127.2628
225	전남 구례군	35.20944	127.46444
226	제주 제주시	33.50972	126.52194
227	제주 서귀포시	33.29307	126.49748

228 rows × 3 columns

03 Python
지역별 상세 정보

- ✓ 각 지역에 대한 회원 수에 비례하여 원의 크기 지정
- ✓ 어느 지역의 회원 수가 가장 많은 지 지도를 통해 확인할 수 있음
- ✓ 서울, 경기권에 회원이 많다는 것을 알 수 있음



03 Python
지역별 상세 정보

- ✓ 각 지역에 대한 회원 수에 대해 마커의 색으로 회원 범례 표현
- ✓ 마커를 클릭했을 때, 지역에 대한 자세한 정보를 알 수 있음
- ✓ 좀 더 상세한 정보가 있을 경우, 더 구체적인 마케팅에 활용할 수 있음