

# SQL Server와 ODBC와 JDBC를 연동하여 고객 데이터 분석 & 시각화

데이터 분석과 조세진

2021. 04. 30

CONTACT

whtpwls777@naver.com



# 목차



## 프로젝트 개요

프로젝트의 설명

## EXCEL과 Oracle 연동

ODBC를 사용하여  
Oracle DBMS와 Excel을  
연동한 후 데이터 분석 및 시각화

## R과 Oracle 연동

JDBC를 사용하여  
R과 Oracle DBMS를  
연동한 후 데이터 분석 및 시각화

## Python과 Oracle 연동

JDBC를 사용하여  
Python과 Oracle DBMS를  
연동한 후 데이터 분석 및 시각화

# 1. 프로젝트 개요

## 각 직원이 담당하는 고객의 수와 연봉의 상관관계

고객의 정보가 담겨있는 customer 테이블과 직원의 정보가 담겨있는 emp 테이블을 이용하여  
직원이 관리하는 고객의 수와 직원의 연봉 간에 상관관계가 있는지 알아봄

EMP TABLE			CUSTOMER TABLE			DEPT TABLE		
이름	널?	유형	이름	널?	유형	이름	널?	유형
EMPNO	NOT NULL	NUMBER (4)	ID	NOT NULL	VARCHAR2 (20)	DEPTNO		NUMBER (2)
ENAME		VARCHAR2 (10)	PWD	NOT NULL	VARCHAR2 (20)	DNAME		VARCHAR2 (14)
JOB		VARCHAR2 (9)	NAME	NOT NULL	VARCHAR2 (20)	LOC		VARCHAR2 (13)
MGR		NUMBER (4)	ZIPCODE		VARCHAR2 (7)			
HIREDATE		DATE	ADDRESS1		VARCHAR2 (100)			
SAL		NUMBER (7, 2)	ADDRESS2		VARCHAR2 (100)			
COMM		NUMBER (7, 2)	MOBILE_NO		VARCHAR2 (14)			
DEPTNO		NUMBER (2)	PHONE_NO		VARCHAR2 (14)			
			CREDIT_LIMIT		NUMBER (9, 2)			
			EMAIL		VARCHAR2 (20)			
			ACCOUNT_MGR		NUMBER (4)			
			BIRTH_DT		DATE			
			ENROLL_DT		DATE			
			GENDER		VARCHAR2 (1)			

PROJECT.1

# ODBC를 사용하여 SQL SERVER 연동 후 데이터 분석 및 시각화

- EXCEL

01

---

ODBC를 사용하여  
Oracle DBMS와 Excel을  
연동한 후 데이터 분석 및 시각화

## 2. Excel - Oracle

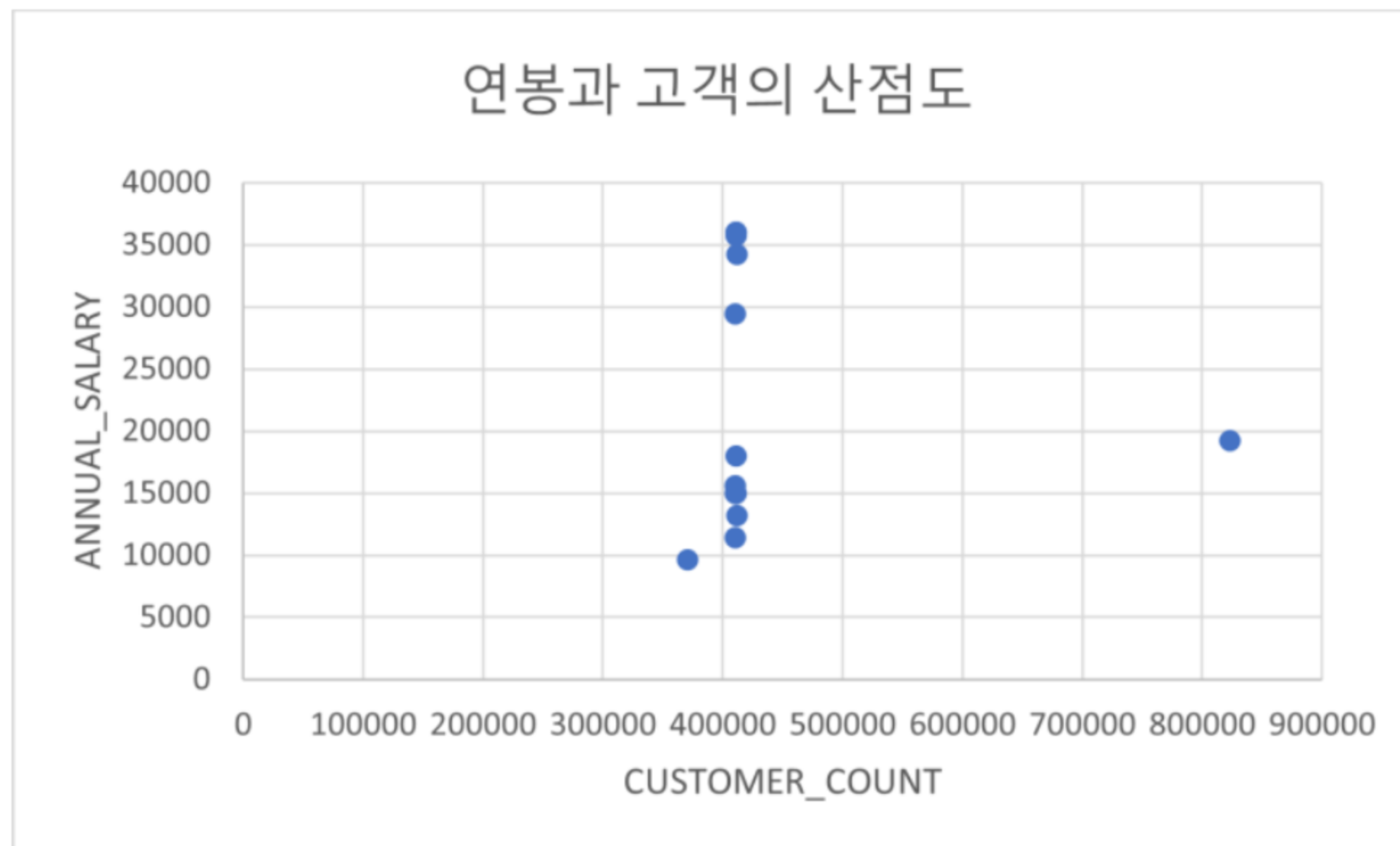
- 데이터 분석 및 시각화

### 사원별 고객의 수와 연봉

- customer table과 emp table을 join한 후

```
select e.ename, count(c.id) as customer_count, avg(sal)*12 as annual_salary  
from CUSTOMER c, emp e  
where c.ACCOUNT_MGR = e.EMPNO  
group by e.ENAME;
```

- 산점도를 봤을 때 고객의 수와 연봉은  
상관이 없다는 것을 확인할 수 있다.



## 2. Excel - Oracle

- 데이터 분석 및 시각화

### 연령대별 고객 비율(주제 외에 고객 테이블 분석)

select case

when birth\_dt between '50/01/01' and '59/12/31' then 50 --50년생

when birth\_dt between '60/01/01' and '69/12/31' then 60 --60년생

when birth\_dt between '70/01/01' and '79/12/31' then 70 --70년생

when birth\_dt between '80/01/01' and '89/12/31' then 80 --80년생

when birth\_dt between '90/01/01' and '99/12/31' then 90 --90년생

end as age\_group, count(\*) as 고객수 from customer group by case

when birth\_dt between '50/01/01' and '59/12/31' then 50

when birth\_dt between '60/01/01' and '69/12/31' then 60

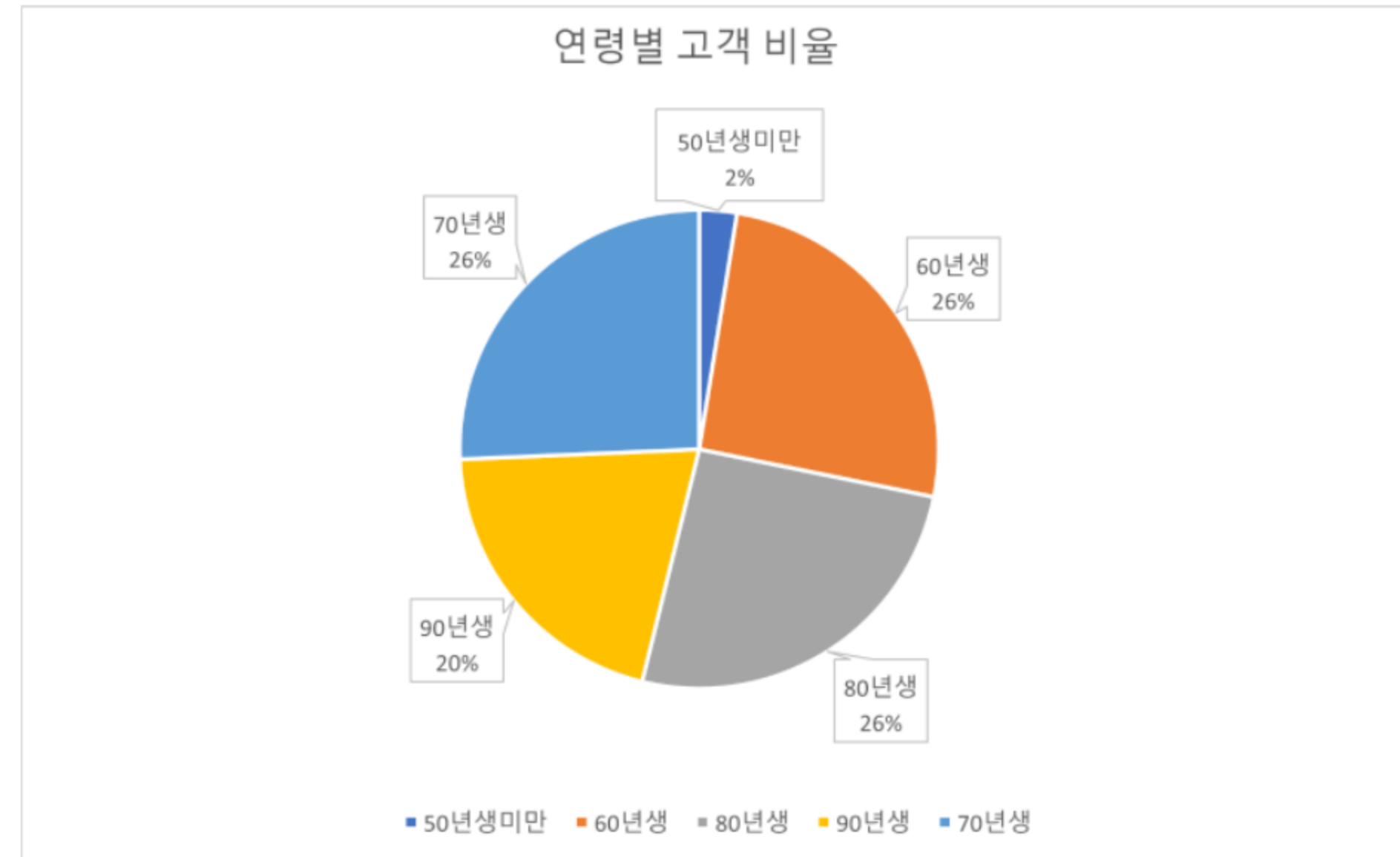
when birth\_dt between '70/01/01' and '79/12/31' then 70

when birth\_dt between '80/01/01' and '89/12/31' then 80

when birth\_dt between '90/01/01' and '99/12/31' then 90

end order by age\_group ;

- 60-80년생이 가장 많은 비율을 차지하고 있다.



## 2. Excel - Oracle

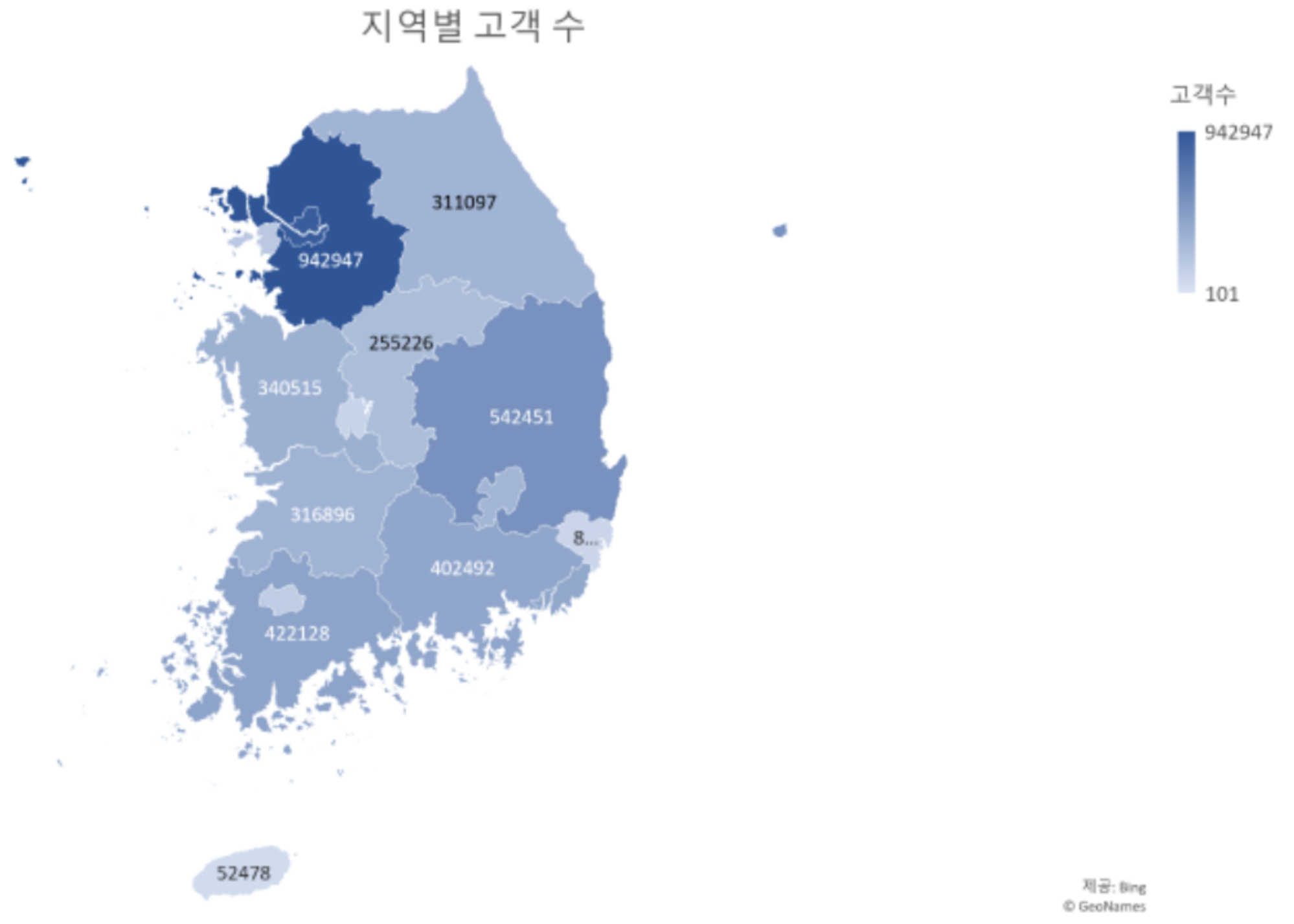
- 데이터 분석 및 시각화

### 지역별 고객 비율(주제 외에 고객 테이블 분석)

```
select decode(substr(address1,1,instr(address1,',', 1, 1) -1 ),  
'uC778천','인천광역시',substr(address1,1,instr(address1,',', 1, 1) -1 ))  
as local, count(*) as "고객수" from customer group by  
substr(address1,1,instr(address1,',', 1, 1) -1 );
```

- 지역은 ADDRESS1 컬럼을 이용하여 데이터 추출  
- 인천광역시의 데이터는 'uC778천'로 나와있어서  
데이터 처리 후 데이터 출력

- 경기와 서울에 많은 인원이 분포되어 있는 것을  
확인할 수 있다.



PROJECT.1

# JDBC를 사용하여 SQL SERVER 연동 후 데이터 분석 및 시각화

- R

02

---

JDBC를 사용하여  
R과 Oracle DBMS를  
연동한 후 데이터 분석 및 시각화



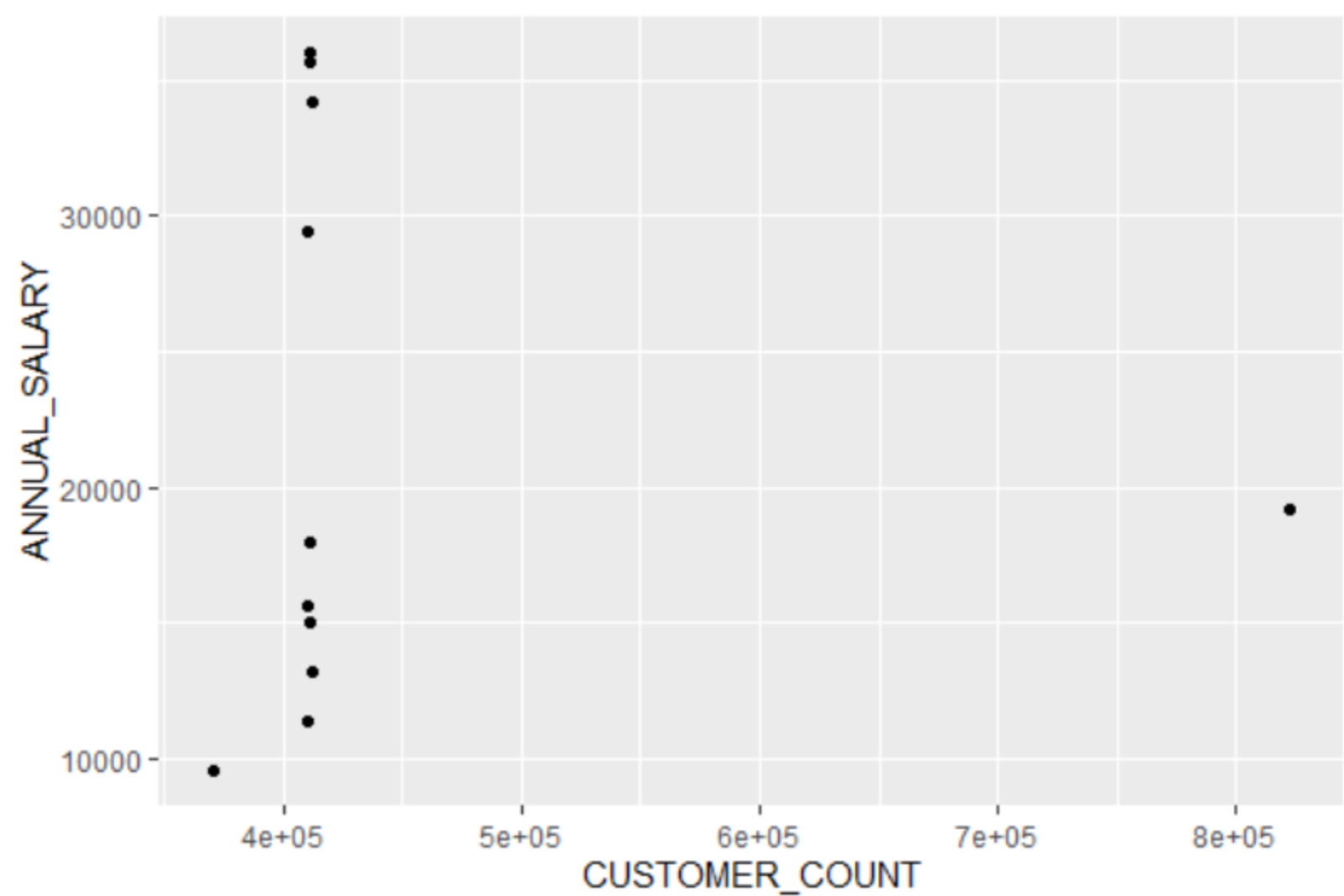
### 3. R - Oracle

- 데이터 분석 및 시각화

#### 사원별 고객의 수와 연봉

```
query <- "select e.ename, count(c.id) as customer_count, avg(sal)*12  
as annual_salary from CUSTOMER c, emp e  
where c.ACCOUNT_MGR = e.EMPNO  
group by e.ENAME"  
a <- dbGetQuery(conn, query)  
library(ggplot2)  
ggplot(data=a,aes(x=CUSTOMER_COUNT, y=ANNUAL_SALARY))+geom_point()
```

- 산점도를 봤을 때 고객의 수와 연봉은 상관이 없다는 것을 확인할 수 있다.



### 3. R - Oracle

- 데이터 분석 및 시각화

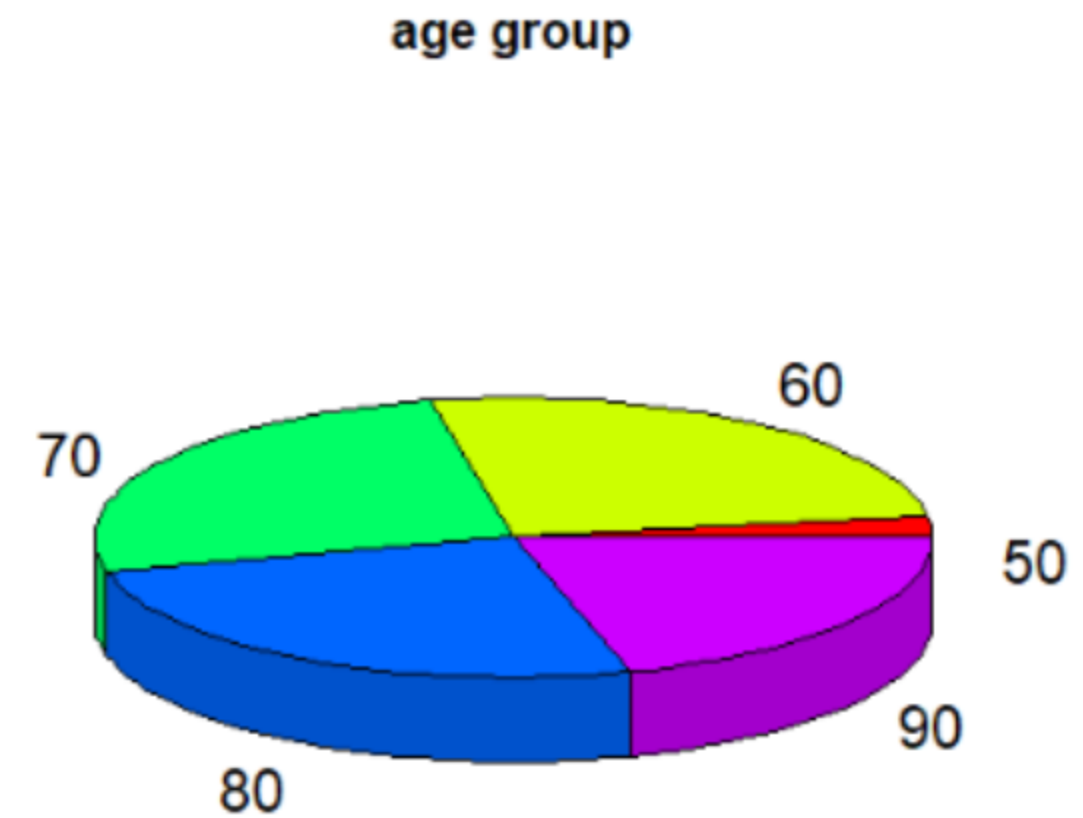
#### 연령대별 고객 비율(주제 외에 고객 테이블 분석)

```
query <- "select case when birth_dt between '50/01/01' and '59/12/31' then 50 when birth_dt between '60/01/01' and '69/12/31' then 60 when birth_dt between '70/01/01' and '79/12/31' then 70 when birth_dt between '80/01/01' and '89/12/31' then 80 when birth_dt between '90/01/01' and '99/12/31' then 90 end as age_group, count(*) as customer_count from customer group by case when birth_dt between '50/01/01' and '59/12/31' then 50 when birth_dt between '60/01/01' and '69/12/31' then 60 when birth_dt between '70/01/01' and '79/12/31' then 70 when birth_dt between '80/01/01' and '89/12/31' then 80 when birth_dt between '90/01/01' and '99/12/31' then 90 end order by age_group"
```

```
library(plotrix)
```

```
pie3D(a$CUSTOMER_COUNT, labels=a$AGE_GROUP, main = "age group")
```

- 60-80년생이 가장 많은 비율을 차지하고 있다.



PROJECT.1

# ODBC를 사용하여 SQL SERVER 연동 후 데이터 분석 및 시각화

## - PYTHON

03

---

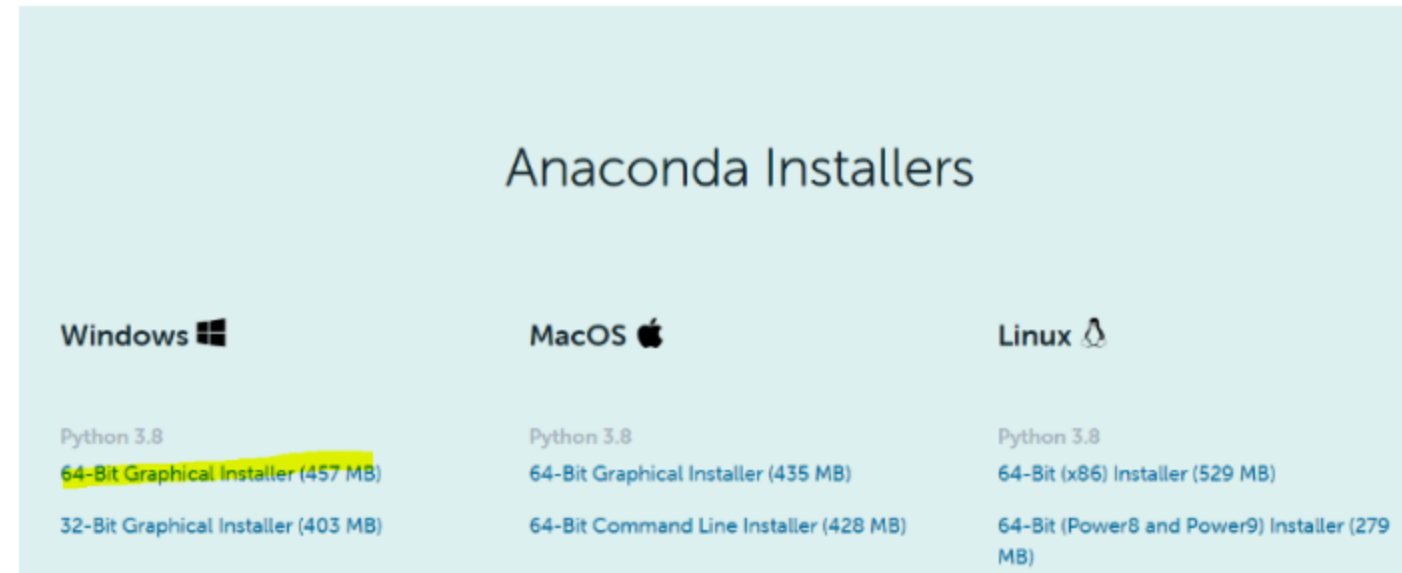
ODBC를 사용하여  
ORACLE DBMS와 EXCLE을 연동한 후  
ORACLE DBMS에 존재하는  
고객 데이터 분석 및 시각화

## 4. Python - Oracle

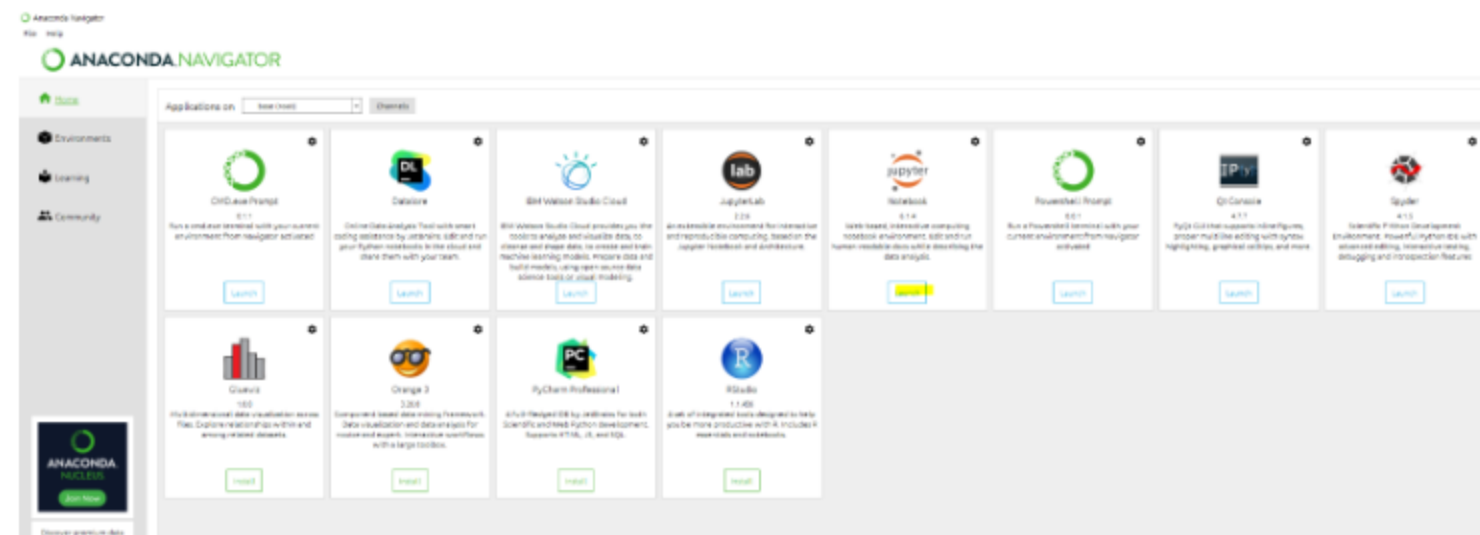
- 데이터 분석 및 시각화

<https://www.anaconda.com/products/individual#download-section>

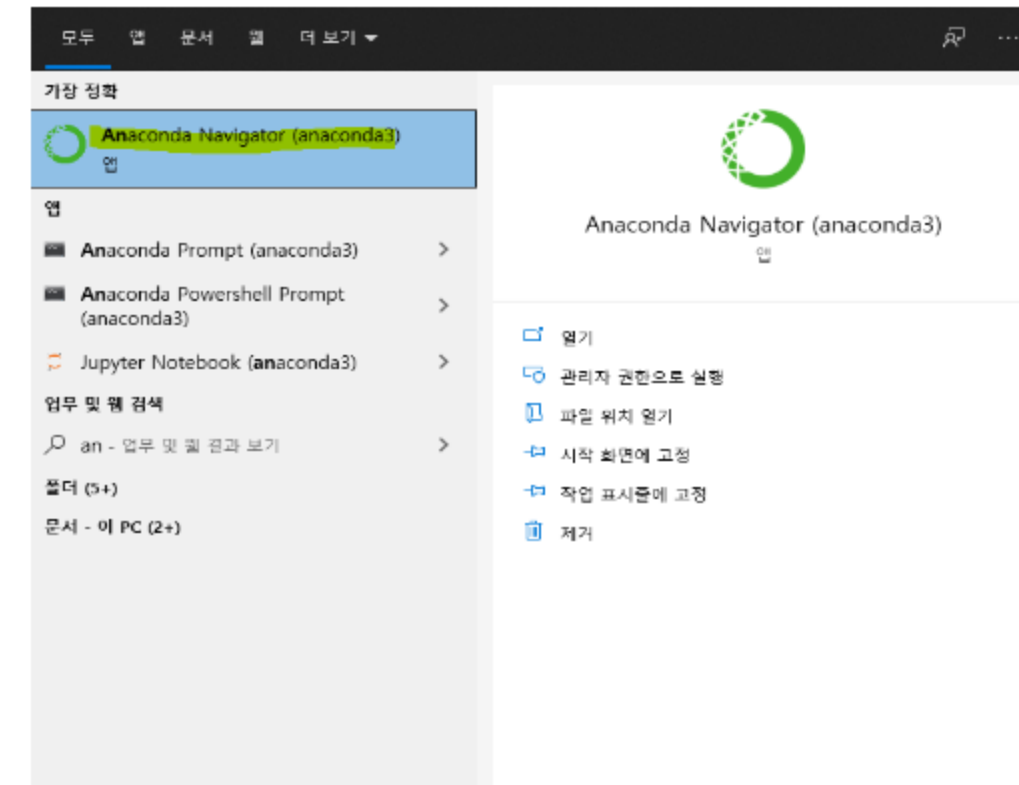
아나콘다 설치



Jupyter를 이용하여 Oracle과 연동



아나콘다 실행



Unitted.ipynb 선택



## 4. Python - Oracle

- 데이터 분석 및 시각화

### 사원별 고객의 수와 연봉

```
import pyodbc
```

```
import pandas as pd
```

```
import numpy as np
```

```
import numpy as np
```

```
import matplotlib.pyplot as plt
```

```
cnxn = pyodbc.connect("DSN=ORCL; uid=scott; pwd=tiger")
```

```
query = "SELECT e.ENAME AS ENAME, COUNT(c.ID) AS customer_count, AVG(SAL)*12 AS annual_salary FROM CUSTOMER c, EMP e WHERE c.ACCOUN
```

```
T_MGR = e.EMPNO GROUP BY e.ENAME;"
```

```
df = pd.read_sql(query, cnxn)
```

```
plt.figure(figsize=(20, 10))
```

```
plt.rc('font', family='Malgun Gothic')
```

```
plt.title("사원별 고객의 수와 연봉")
```

```
plt.scatter(df.CUSTOMER_COUNT, df.ANNUAL_SALARY)
```

```
plt.xlabel('고객수', fontsize=20)
```

```
plt.ylabel('연봉', fontsize=20)
```

```
plt.show()
```

```
jupyter Untitled Last Checkpoint: 33분 전 (unsaved changes) Logout
File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 C
In [1]: import pyodbc
import pandas as pd
import numpy as np
import numpy as np
import matplotlib.pyplot as plt

In [19]: cnxn = pyodbc.connect("DSN=ORCL; uid=scott; pwd=tiger")

query = "SELECT e.ENAME AS ENAME, COUNT(c.ID) AS customer_count, AVG(SAL)*12 AS annual_salary FROM CUSTOMER c, EMP e WHERE c.ACCOUN
df = pd.read_sql(query, cnxn)

plt.figure(figsize=(20, 10))

plt.rc('font', family='Malgun Gothic')

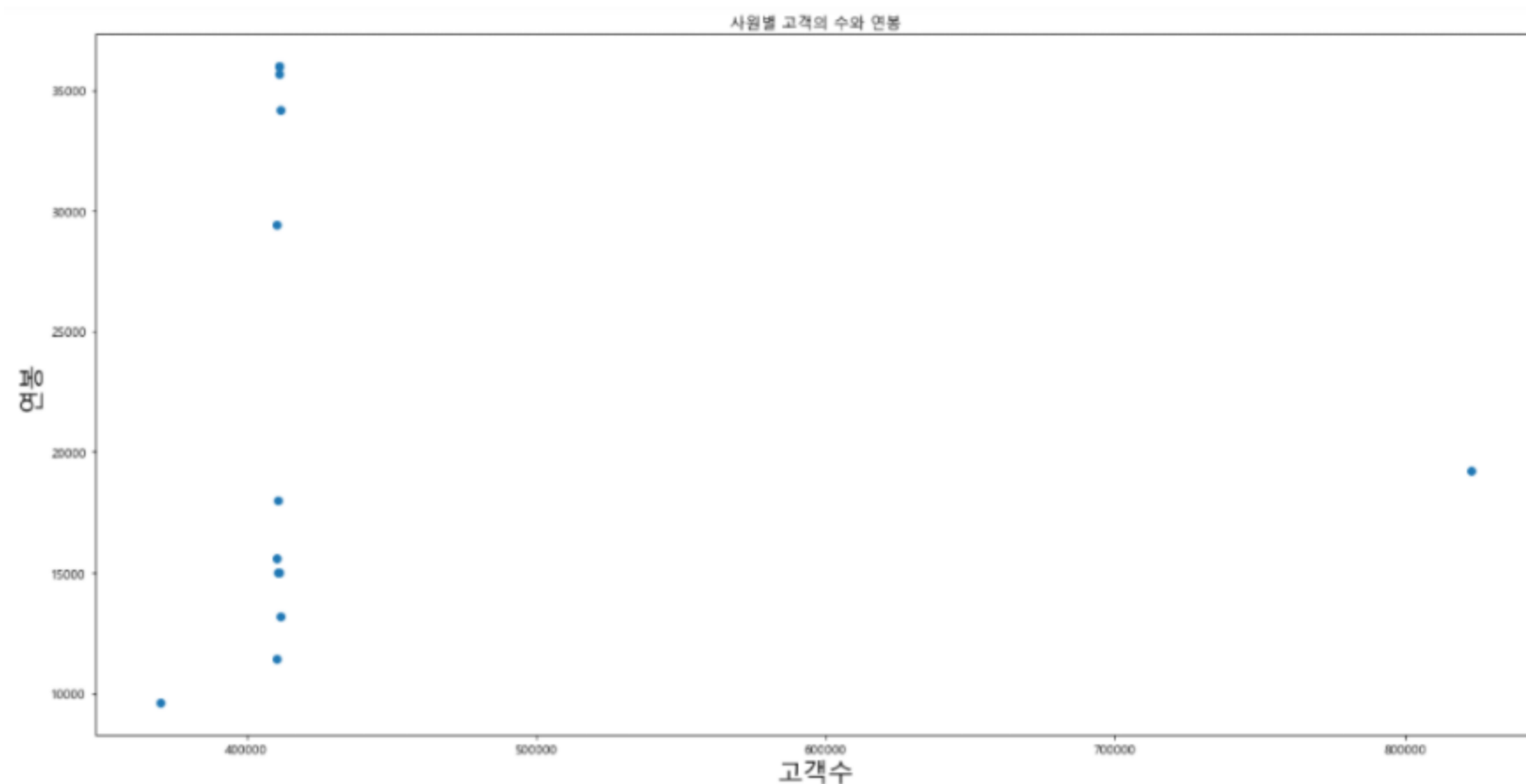
plt.title("사원별 고객의 수와 연봉")

plt.scatter(df.CUSTOMER_COUNT, df.ANNUAL_SALARY)

plt.xlabel('고객수', fontsize=20)

plt.ylabel('연봉', fontsize=20)

plt.show()
```



## 4. Python - Oracle

- 데이터 분석 및 시각화

분석된 Source 데이터를 JSON로 생성

```
js = df.to_json(orient = 'table')
```

```
with open('DATA.json', 'w', encoding='utf-8') as file:
```

```
    df.to_json(file, force_ascii=False)
```

실행하면 DATA.json 파일이 생성

jupyter	Quit	Logout
<input type="checkbox"/> Downloads	37분 전	
<input type="checkbox"/> eclipse	2달 전	
<input type="checkbox"/> eclipse-workspace	7일 전	
<input type="checkbox"/> Favorites	2달 전	
<input type="checkbox"/> git	2달 전	
<input type="checkbox"/> Intel	6달 전	
<input type="checkbox"/> Links	2달 전	
<input type="checkbox"/> Music	2달 전	
<input type="checkbox"/> OneDrive	9시간 전	
<input type="checkbox"/> Oracle	5일 전	
<input type="checkbox"/> Pictures	2달 전	
<input type="checkbox"/> Saved Games	2달 전	
<input type="checkbox"/> Searches	2달 전	
<input type="checkbox"/> test	2달 전	
<input type="checkbox"/> Videos	2달 전	
<input type="checkbox"/> Untitled.ipynb	Running 1분 전	18.9 kB
<input type="checkbox"/> DATA.json	몇 초 전	538 B

jupyter DATA.json	Logout
File Edit View Language	JSON
1	{ "ENAME": { "0": "ALLEN", "1": "JONES", "2": "FORD", "3": "CLARK", "4": "MILLER", "5": "SMITH", "6": "WARD", "7": "MARTIN", "8": "SCOTT", "9": "TURNER", "10": "ADAMS", "11": "BLAKE", "12": "JAMES"}, "CUSTOMER_COUNT": { "0": 822901.0, "1": 411274.0, "2": 411221.0, "3": 410620.0, "4": 410460.0, "5": 370474.0, "6": 411269.0, "7": 410791.0, "8": 411123.0, "9": 410658.0, "10": 411644.0, "11": 411699.0, "12": 410420.0}, "ANNUAL_SALARY": { "0": 19200.0, "1": 35700.0, "2": 36000.0, "3": 29400.0, "4": 15600.0, "5": 9600.0, "6": 15000.0, "7": 15000.0, "8": 36000.0, "9": 18000.0, "10": 13200.0, "11": 34200.0, "12": 11400.0}}

# 감사합니다!

# 잘 부탁드립니다!

2021.04. 30

**CONTACT**

whtpwls777@naver.com

