



# 데이터, 정보, 지식

데이터 : 관찰의 결과로 나타난 정량적 혹은 정성적인 실제 값

정보 : 데이터에 의미를 부여한 것

지식 : 사물이나 현상에 대한 이해

8848m의 에베레스트 -> 데이터

세계에서 가장 높은 산 -> 정보

에베레스트 관련 보고서 -> 지식



# 데이터베이스와 SQL

데이터 :

컴퓨터가 처리할 수 있는 문자, 숫자, 소리, 그림 따위의 형태로 된 자료.

데이터베이스(DB) :

데이터의 집합.

'데이터의 저장 공간' 자체.

논리적으로 연관된 데이터를 모아 일정한 형태로 저장해 놓은 것.

여러 명의 사용자나 응용프로그램이 공유하는 데이터 (동시 접근 가능)

응용 시스템들이 공용으로 사용하기 위해 통합, 저장한 데이터 집합





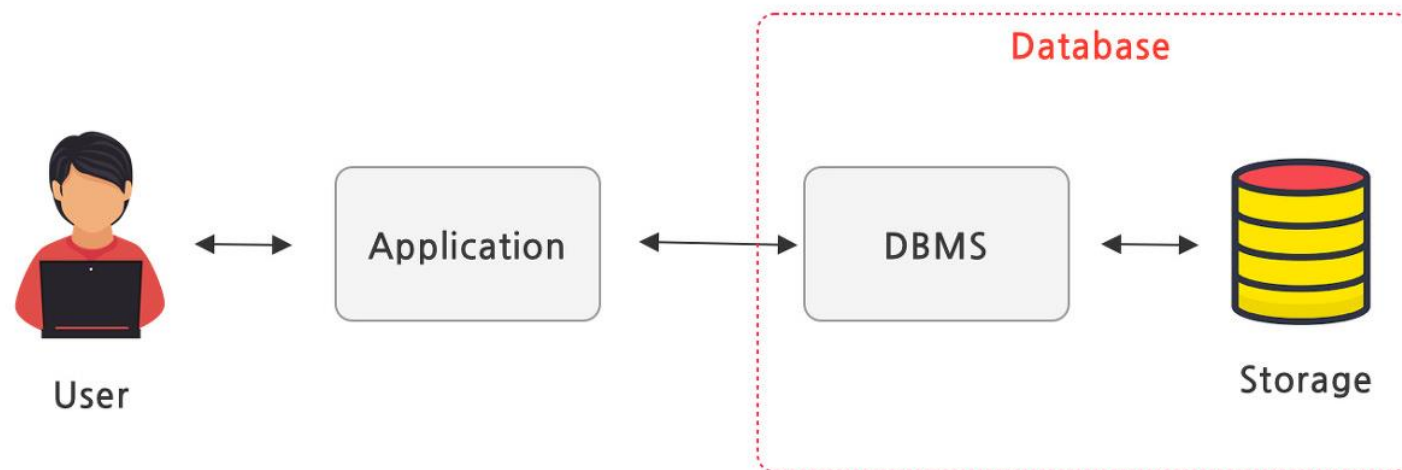
# 데이터베이스와 SQL

DBMS(Database Management System) :

데이터베이스를 관리 및 운영하는 소프트웨어. (데이터베이스 관리 프로그램)

대부분의 DBMS가 테이블로 구성된 관계형 DBMS(RDBMS)형태로 사용  
(MySQL, 오라클(Oracle), SQL 서버, MariaDB 등.)

DBMS를 이용하여 데이터의 입력, 수정, 삭제 등의 기능을 제공





# 데이터베이스와 SQL

DBMS(Database Management System) :

DBMS가 설치되어 데이터를 가진 쪽을 서버(Server)  
데이터를 요청하는 쪽을 클라이언트(Client)라고 한다.

데이터의 일관성 유지, 복구, 동시 접근 제어 등의 기능을 수행



# 데이터베이스와 SQL

## 일반 사용자

- 은행의 창구 혹은 관공서의 민원 접수처 등에서 데이터를 다루는 업무를 하는 사람
- 프로그래머가 개발한 프로그램을 이용해 데이터베이스에 접근하는 사용자

## 응용 프로그래머

- 일반 사용자가 사용할 수 있도록 프로그램을 만드는 사람
- 프로그래밍 언어와 SQL을 사용해 일반 사용자를 위한 UI와 데이터 로직을 개발

## SQL 사용자

- SQL을 이용해 업무를 처리하는 IT부서 담당자
- 응용 프로그램으로 구현되어 있지 않은 업무를 SQL을 사용해 처리

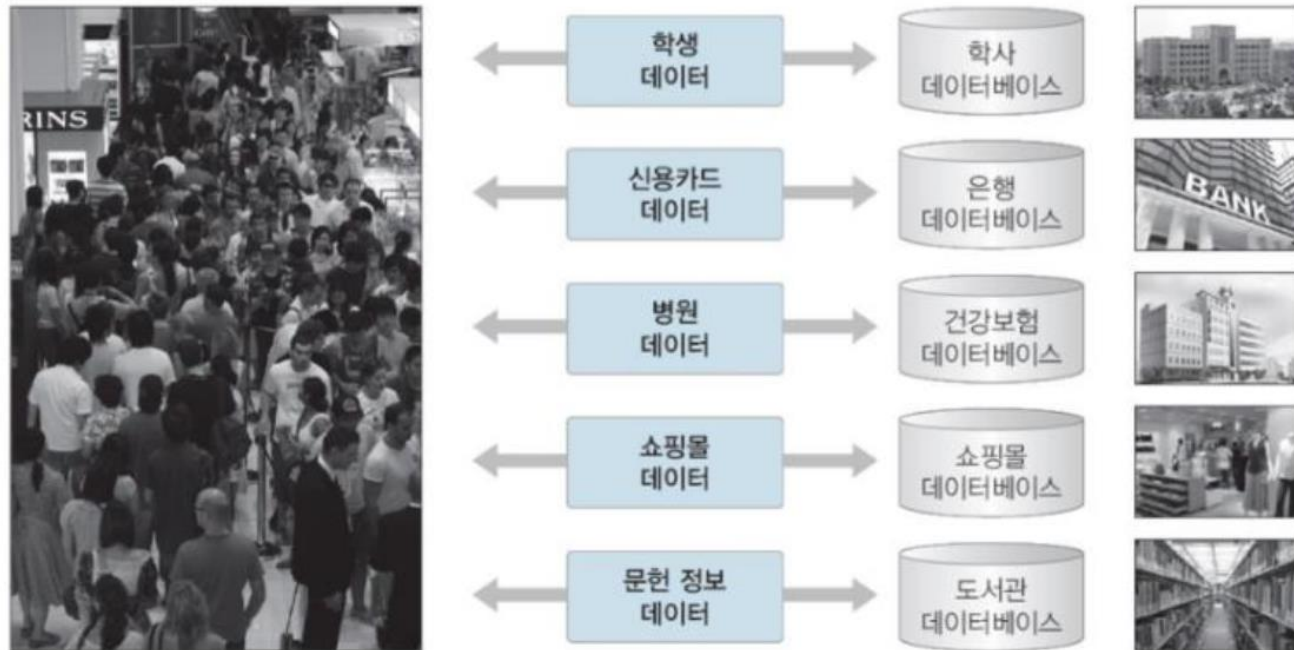
## 데이터베이스 관리자(DBA)

- DataBase Administrator, 데이터베이스 운영 조직의 시스템을 총괄
- 데이터 설계, 구현, 유지보수의 전 과정을 담당



# 데이터베이스와 SQL

데이터베이스 시스템은 데이터의 검색(select)과 변경(insert, delete, update 등) 작업을 수행  
우리에게 필요한 정보들을 구조적으로 모아서 관리





# 데이터베이스와 SQL

## 데이터베이스의 특징:

1. 실시간 접근성 : DB는 실시간으로 서비스된다.  
사용자가 데이터를 요청하면 수 초 내에 결과를 서비스한다.
2. 계속적인 변화 : DB에 저장된 내용은 한 순간의 상태이지만,  
데이터 값은 시간에 따라 항상 바뀐다.
3. 동시 공유 : DB는 서로 다른 업무, 여러 사용자에게 동시에 공유된다.
4. 내용에 따른 참조 : DB에 저장된 데이터는 물리적인 위치가 아니라 값에 따라 참조된다.



# 데이터베이스와 SQL

## 데이터베이스의 개념:

1. 통합된 데이터 : 데이터를 통합하는 개념.  
각자 사용하던 데이터의 중복을 최소화하여 중복으로 인한 데이터 불일치 제거
2. 저장된 데이터 : 문서로 보관된 데이터가 아니라 컴퓨터 저장장치에 저장된 데이터
3. 운영 데이터 : 조직의 목적을 위해 사용되는 데이터
4. 공용 데이터 : 한 사람 또는 한 업무를 위해 사용되는 데이터가 아니라 공동으로 사용되는 데이터





# 데이터베이스와 SQL

데이터베이스의 종류:

## 1. 계층형 데이터베이스 (HDBMS)

- 폴더와 파일 등의 계층(트리) 구조
- 상하종속 관계(1:N 관계)로 수정이 어렵고, 데이터 중복 가능성 높음
- 현재는 거의 사용되지 않음

## 2. 네트워크형 데이터베이스 (NDBMS)

- 데이터의 구조를 네트워크 상의 노드 형태로 논리적이게 표현한 데이터 모델
- 노드는 서로 대등한 관계
- 1:1, 1:N, N:M 관계 지원으로 효과적이고 빠른 데이터 추출 가능
- 구조가 복잡하여 변경이나 운영이 어려움

## 3. 관계형 데이터베이스 (RDBMS)

- 행과 열을 가지는 표 형식의 데이터 모델, SQL을 이용
- 다양한 용도로 사용 가지며, 높은 신뢰성(데이터 무결성) 보장
- 데이터 분류, 정렬, 탐색 속도가 빠름



# 데이터베이스와 SQL

데이터베이스의 종류:

## 4. 객체지향 데이터베이스 (ODBMS)

- 객체 그대로를 데이터베이스의 데이터에 저장
- 멀티미디어 데이터의 원활한 처리

## 5. 키-값 데이터베이스 (NoSQL)

- 키와 값을 한 쌍으로 갖는 형태로 대표적인 NoSQL(Not Only SQL)
- 단순한 구조로 속도가 빠르며, 분산 저장 시 용이
- 데이터를 자주 변경하지 않고, 읽기를 자주하는 경우 주로 사용

## 6. 시계열 데이터베이스(TSDB)

- 시간에 따라 데이터를 저장
- 시간 경과에 따른 데이터 변화를 추적하는데 용이



# 데이터베이스와 SQL

관계형 DBMS(R-DBMS) :

데이터베이스를 테이블(릴레이션, 엔티티)이라는 최소 단위로 구성.

\* 테이블은 하나 이상의 열(column)로 이루어져 있다.

<테이블(릴레이션, 엔티티)>

열 (컬럼, 필드, 속성, 애트리뷰트)

행 (로우, 튜플, 레코드)

아이디	이름	연락처
flower	화사	010-1111-1111
finetree	솔라	010-2222-2222
moon	문별	010-3333-3333
whee	휘인	010-4444-4444



# 데이터베이스와 SQL

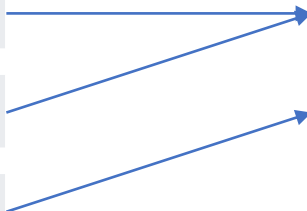
관계형 DBMS(R-DBMS) :

상품 (품번, 품명, 카테고리번호)

101	아메리카노	1
102	카페라떼	1
103	티라미수	2

카테고리 (카테고리번호, 카테고리명)

1	음료
2	푸드





# 데이터베이스와 SQL

SQL(Structured Query Language) :  
관계형 데이터베이스에서 데이터 조작과 정의를 위해 사용되는 언어

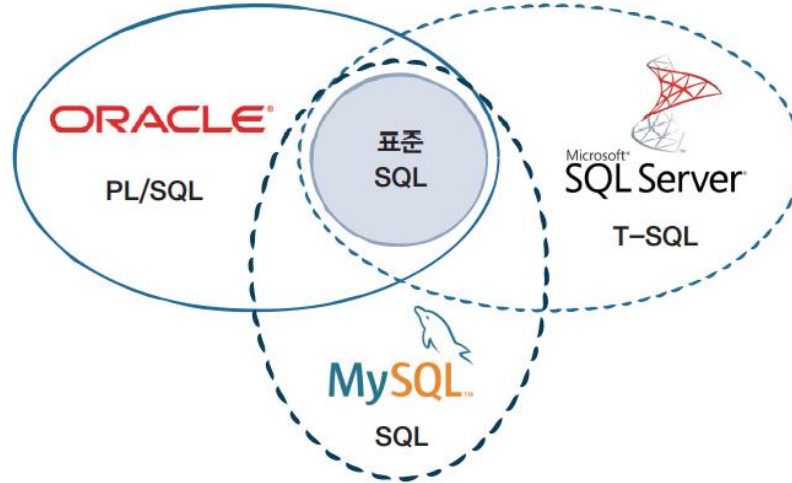


관계형 DBMS를 배우려면 SQL을 필수로 익혀야 한다.



# 데이터베이스와 SQL

문제는 SQL을 사용하는 DBMS를 만드는 회사가 여러 곳이기 때문에 표준 SQL이 각 회사 제품의 특성을 모두 포용하지 못한다.



국제표준화기구에서 SQL에 대한 표준인 [표준 SQL]을 정해 발표하고 있으며,

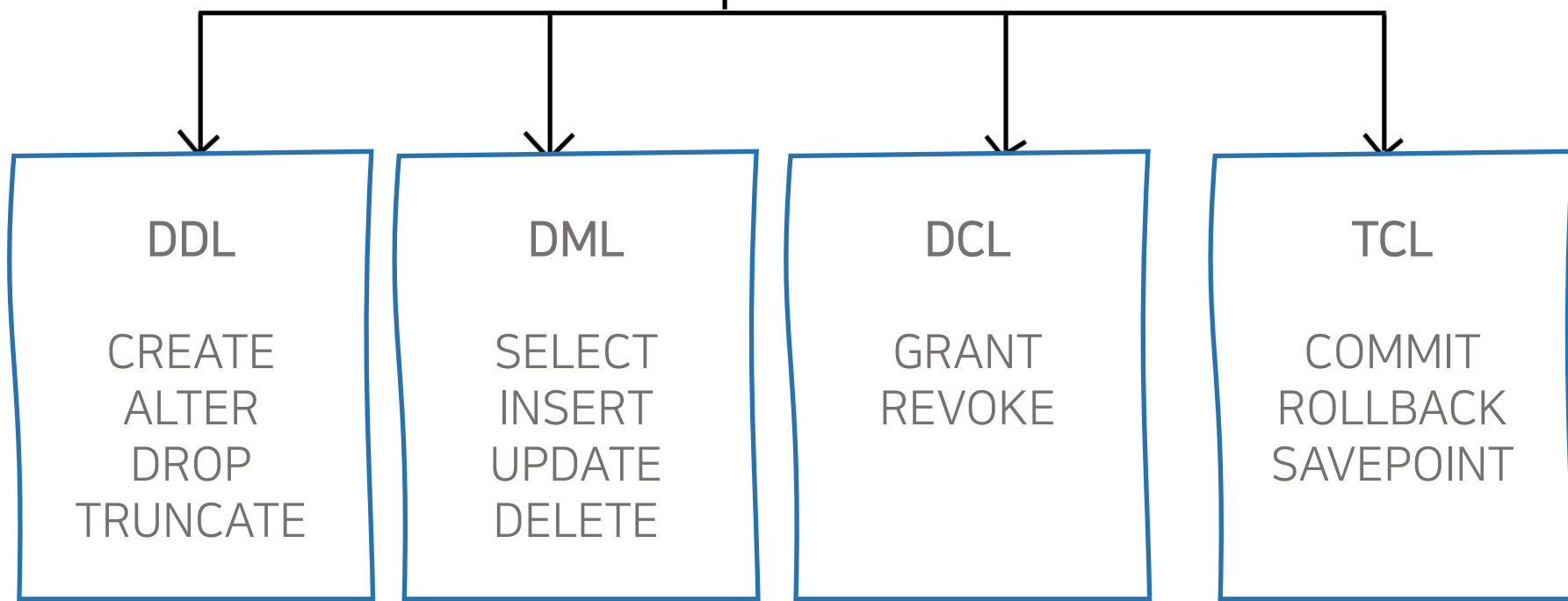
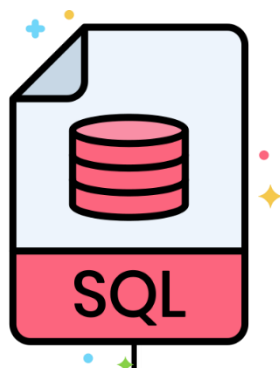
DBMS를 만드는 회사에서는 되도록 표준 SQL을 준수하되,  
각 제품의 특성을 반영한 SQL을 사용하고 있다.



# 데이터베이스와 SQL

SQL(Structured Query Language) :

- DBMS 제작 회사와 독립적
- 다른 시스템으로 이식성이 좋음
- 표준이 계속 발전 중
- 대화식 언어
- 분산형 클라이언트/서버 구조







# 데이터베이스와 SQL

DDL(Data Definition Language) :

- 데이터를 정의하는 명령어
- 데이터를 담는 그릇(객체)를 만들거나 삭제할 때 사용

DML(Data Manipulation Language) :

- 데이터를 조작하는 명령어
- 데이터의 추가, 삭제, 갱신과 같은 작업 시 사용

DCL(Data Control Language) :

- 데이터를 제어하는 명령어
- 트랜잭션을 제어하거나 데이터 접근권한을 제어할 때 사용
- \* 트랜잭션을 제어하는 명령만을 분리해 TCL이라 한다.



# 데이터베이스와 SQL

## DDL:

- CREATE : 데이터베이스 또는 테이블을 생성
- ALTER : 테이블을 수정
- DROP : 데이터베이스 또는 테이블 삭제
- TRUNCATE : 테이블을 초기화

## DML :


- SELECT : 데이터를 검색
- INSERT : 데이터를 입력
- UPDATE : 데이터를 수정
- DELETE : 데이터를 삭제

## DCL :

- GRANT : 특정 수행 권한을 부여
- REVOKE : 특정 수행 권한을 삭제
- COMMIT : 트랜잭션 작업을 완료하는 역할
- ROLLBACK : 트랜잭션 작업을 취소하거나 이전 상태로 복구하는 역할



# 데이터베이스와 SQL

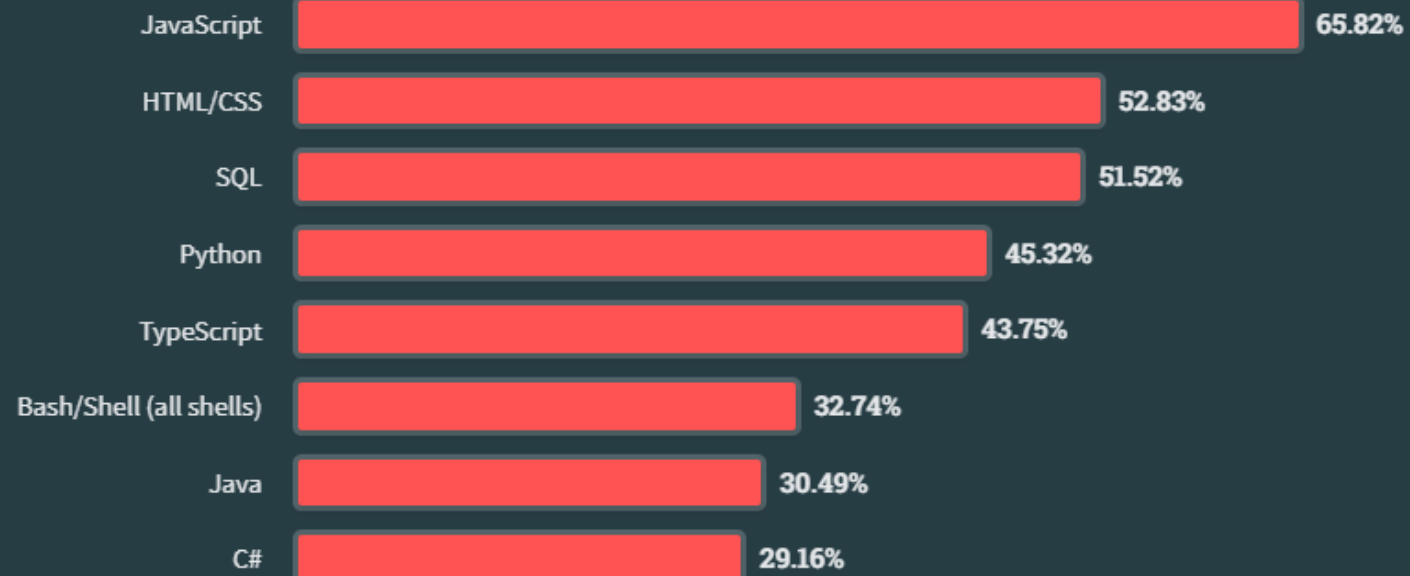
 [survey.stackoverflow.co/2023/#most-popular-technologies-language-prof](https://survey.stackoverflow.co/2023/#most-popular-technologies-language-prof)

All Respondents

**Professional Developers**

Learning to Code

Other Coders





# 데이터베이스와 SQL

## RELATIONAL DATABASE



## NON-RELATIONAL DATABASE





# 데이터베이스와 SQL

Rank			DBMS	Database Model	Score		
Feb 2024	Jan 2024	Feb 2023			Feb 2024	Jan 2024	Feb 2023
1.	1.	1.	Oracle +	Relational, Multi-model ⓘ	1241.45	-6.05	-6.08
2.	2.	2.	MySQL +	Relational, Multi-model ⓘ	1106.67	-16.79	-88.78
3.	3.	3.	Microsoft SQL Server +	Relational, Multi-model ⓘ	853.57	-23.03	-75.52
4.	4.	4.	PostgreSQL +	Relational, Multi-model ⓘ	629.41	-19.55	+12.90
5.	5.	5.	MongoDB +	Document, Multi-model ⓘ	420.36	+2.88	-32.41
6.	6.	6.	Redis +	Key-value, Multi-model ⓘ	160.71	+1.33	-13.12
7.	7.	↑ 8.	Elasticsearch	Search engine, Multi-model ⓘ	135.74	-0.33	-2.86
8.	8.	↓ 7.	IBM Db2	Relational, Multi-model ⓘ	132.23	-0.18	-10.74
9.	9.	↑ 12.	Snowflake +	Relational	127.45	+1.53	+11.80
10.	↑ 11.	↓ 9.	SQLite +	Relational	117.28	+2.08	-15.38



# MySQL

- SQL에 기반을 둔 관계형 DBMS
- 3개의 상용 에디션(Standard, Enterprise, Cluster CGE)과 1개의 무료 에디션(Community)
- 리눅스, 유닉스, 윈도우 등 거의 모든 운영체제에서 사용 가능
- 처리가 빠르고 대용량 데이터 처리에 용이



# MySQL 설치하기



<https://dev.mysql.com/downloads/installer/>

<b>Windows (x86, 32-bit), MSI Installer</b> (mysql-installer-web-community-8.0.36.0.msi)	8.0.36	2.1M	<a href="#">Download</a>
	MD5: 81061532541f716cf6c6e2c4881a154c   <a href="#">Signature</a>		
<b>Windows (x86, 32-bit), MSI Installer</b> (mysql-installer-community-8.0.36.0.msi)	8.0.36	285.3M	<a href="#">Download</a>
	MD5: d63232c190d0c9c294a2f8d776ed1c20   <a href="#">Signature</a>		



# MySQL 설치하기



**Login »**

using my Oracle Web account

**Sign Up »**

for an Oracle Web account

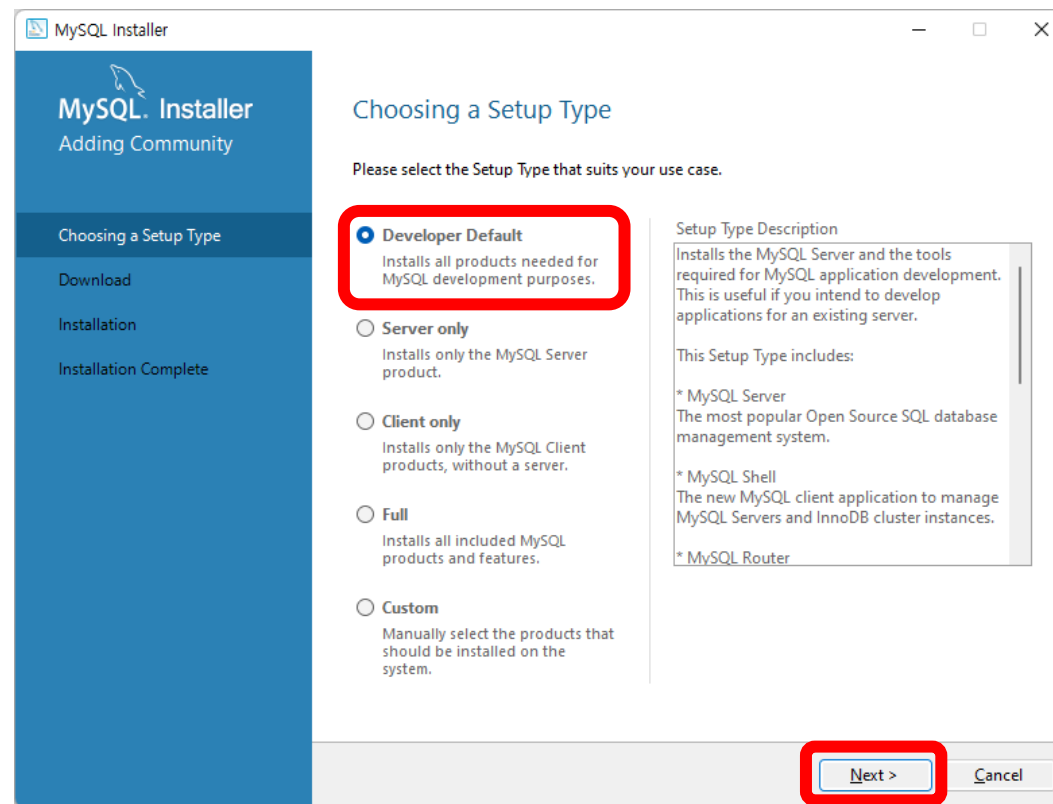
MySQL.com is using Oracle SSO for authentication. If you already have an Oracle Web account, click the Login link. Otherwise, you can signup for a free account by clicking the Sign Up link and following the instructions.

**No thanks, just start my download.**





# MySQL 설치하기





# MySQL 설치하기



MySQL Installer

MySQL Server 8.0.31

Type and Networking

Authentication Method

Accounts and Roles

Windows Service

Server File Permissions

Apply Configuration

Server Configuration Type

Choose the correct server configuration type for this MySQL Server installation. This setting will define how much system resources are assigned to the MySQL Server instance.

Config Type: Development Computer

Connectivity

Use the following controls to select how you would like to connect to this server.

☒ TCP/IP Port: 3306 X Protocol Port: 33060

☒ Open Windows Firewall ports for network access

☐ Named Pipe Pipe Name: MYSQL

☐ Shared Memory Memory Name: MYSQL

Advanced Configuration

Select the check box below to get additional configuration pages where you can set advanced and logging options for this server instance.

☐ Show Advanced and Logging Options

Next > Cancel

MySQL Installer

MySQL Server 8.0.31

Type and Networking

Authentication Method

Accounts and Roles

Windows Service

Server File Permissions

Apply Configuration

Authentication Method

☒ Use Strong Password Encryption for Authentication (RECOMMENDED)

MySQL 8 supports a new authentication based on improved stronger SHA256-based password methods. It is recommended that all new MySQL Server installations use this method going forward.

Attention: This new authentication plugin on the server side requires new versions of connectors and clients which add support for this new 8.0 default authentication (caching\_sha2\_password authentication).

Currently MySQL 8.0 Connectors and community drivers which use libmysqlclient 8.0 support this new method. If clients and applications cannot be updated to support this new authentication method, the MySQL 8.0 Server can be configured to use the legacy MySQL Authentication Method below.

☐ Use Legacy Authentication Method (Retain MySQL 5.x Compatibility)

Using the old MySQL 5.x legacy authentication method should only be considered in the following cases:

- If applications cannot be updated to use MySQL 8 enabled Connectors and drivers.
- For cases where re-compilation of an existing application is not feasible.
- An updated, language specific connector or driver is not yet available.

Security Guidance: When possible, we highly recommend taking needed steps towards upgrading your applications, libraries, and database servers to the new stronger authentication. This new method will significantly improve your security.

< Back Next > Cancel



# MySQL 설치하기



MySQL Installer

MySQL Server 8.0.31

- Type and Networking
- Authentication Method
- Accounts and Roles
- Windows Service
- Server File Permissions
- Apply Configuration

### Accounts and Roles

**Root Account Password**  
Enter the password for the root account. Please remember to store this password in a secure place.

MySQL Root Password:

Repeat Password:

Password strength: **Weak**

**1234**

**MySQL User Accounts**  
Create MySQL user accounts for your users and applications. Assign a role to the user that consists of a set of privileges.

MySQL User Name	Host	User Role
-----------------	------	-----------

[Add User](#)  
[Edit User](#)  
[Delete](#)

[< Back](#) [Next >](#) [Cancel](#)



# MySQL 설치하기



MySQL Installer

MySQL. Installer  
Samples and Examples

Connect To Server

Apply Configuration

### Connect To Server

Select the MySQL server instances from the list to receive sample schemas and data.

Server	Port	Arch...	Type	Status
<input checked="" type="checkbox"/> MySQL Server 8.0.31	3306	X64	Stand-alone Server	Connection succeeded.

Provide the credentials that should be used (requires root privileges).  
Click "Check" to ensure they work.

User name:  Credentials provided in Server configuration

Password:

root / 1234



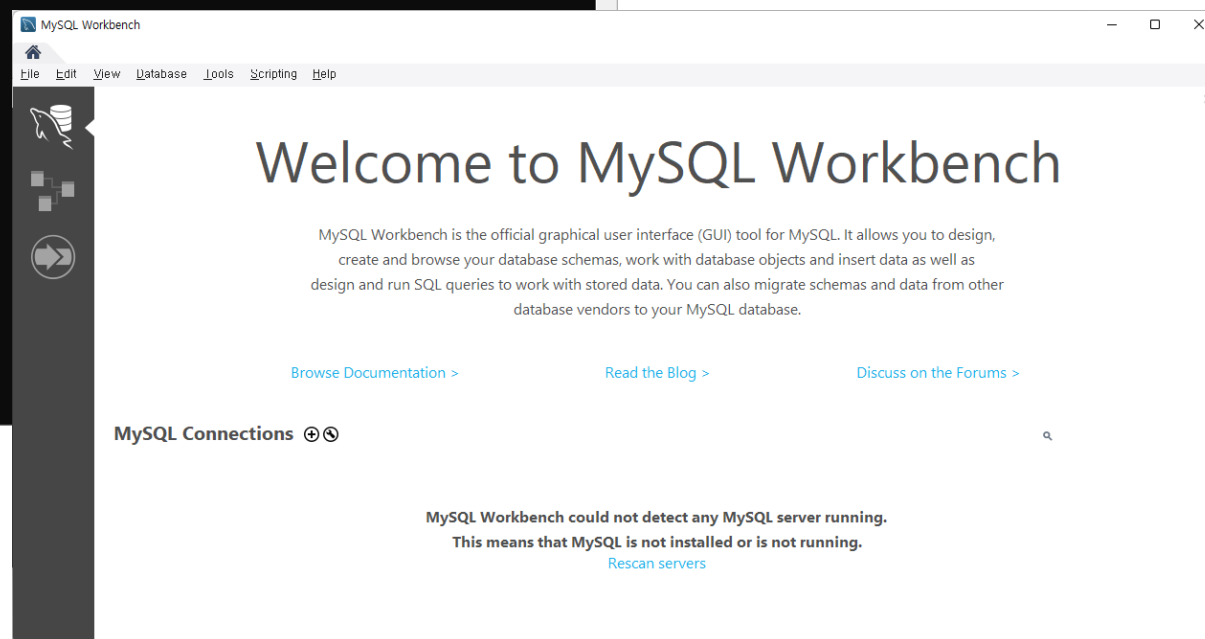
# MySQL 설치하기



```
C:\Program Files\MySQL\MySQL Shell 8.0\bin>mysqlsh.exe
MySQL Shell 8.0.31

Copyright (c) 2016, 2022, Oracle and/or its affiliates.
Oracle is a registered trademark of Oracle Corporation and/or its affiliates.
Other names may be trademarks of their respective owners.

Type 'help' or '?' for help; 'quit' to exit.
MySQL JS>
```





# MySQL 설치하기

## 1. homebrew 설치 여부 확인

\* homebrew 명령어 한 줄로 프로그램을 설치/제거할 수 있는 프로그램

```
brew --version
```

```
brew: command not found
```

2번으로

```
Homebrew 4.0.3
```

3번으로



# MySQL 설치하기

## 2. homebrew 설치

\* homebrew 명령어 한 줄로 프로그램을 설치/제거할 수 있는 프로그램

```
/bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
```

\* 실리콘칩(M1 이상)인 경우에만 아래 내용 실행

```
echo 'export PATH=/opt/homebrew/bin:$PATH' >> ~/.zshrc  
source ~/.zshrc
```



# MySQL 설치하기

## 3. MySQL 설치

```
brew install mysql
```

설치 이후, 버전 확인

```
mysql -V
```

MySQL 서버 켜기

```
mysql.server start
```





# MySQL 설치하기

## 4. MySQL 초기 설정

```
sudo mysql_secure_installation
```



# MySQL 설치하기

## 1) 암호를 복잡하게 설정할 것인가? (N)

VALIDATE PASSWORD COMPONENT can be used to test passwords and improve security. It checks the strength of password and allows the users to set only those passwords which are secure enough. Would you like to setup VALIDATE PASSWORD component?

Press y|Y for Yes, any other key for No:

## 2) 암호를 설정 (1234) (1234)

New password: 🔑

Re-enter new password: 🔑



# MySQL 설치하기

## 3) 기본으로 설정된 익명의 사용자를 삭제할 것인가? (Y)

By default, a MySQL installation has an anonymous user, allowing anyone to log into MySQL without having to have a user account created for them. This is intended only for testing, and to make the installation go a bit smoother. You should remove them before moving into a production environment.

Remove anonymous users?

(Press y|Y for Yes, any other key for No) :

## 4) 원격 접속을 불허할 것인가? (Y)

Normally, root should only be allowed to connect from 'localhost'. This ensures that someone cannot guess at the root password from the network.

Disallow root login remotely?

(Press y|Y for Yes, any other key for No) :



# MySQL 설치하기

## 5) testDB를 삭제할 것인가? (Y)

By default, MySQL comes with a database named 'test' that anyone can access. This is also intended only for testing, and should be removed before moving into a production environment.

Remove test database and access to it?  
(Press y|Y for Yes, any other key for No) :

## 6) 변경 사항을 지금 바로 적용할 것인가? (Y)

Reloading the privilege tables will ensure that all changes made so far will take effect immediately.


Reload privilege tables now?  
(Press y|Y for Yes, any other key for No) :



# MySQL 설치하기

## 5. workbench 설치

<https://dev.mysql.com/downloads/workbench/>

 Packages support latest version Sonoma (14) and deemed compatible with Ventura (13)

<b>macOS (x86, 64-bit), DMG Archive</b> (mysql-workbench-community-8.0.36-macos-x86_64.dmg)	8.0.36	115.1M	<a href="#">Download</a>
<b>macOS (ARM, 64-bit), DMG Archive</b> (mysql-workbench-community-8.0.36-macos-arm64.dmg)	8.0.36	114.2M	<a href="#">Download</a>

intel CPU

실리콘 CPU (m1, m2, m3)



# MySQL 설치하기

**Login »**

using my Oracle Web account

**Sign Up »**

for an Oracle Web account

MySQL.com is using Oracle SSO for authentication. If you already have an Oracle Web account, click the Login link. Otherwise, you can signup for a free account by clicking the Sign Up link and following the instructions.

**No thanks, just start my download.**



# MySQL 설치하기





# MySQL 설치하기 (한글 사용자명)



<https://dev.mysql.com/downloads/mysql/>

[General Availability \(GA\) Releases](#) [Archives](#) [Info](#)

## MySQL Community Server 8.0.36

Select Version:  
8.0.36

Select Operating System:  
Microsoft Windows

**Recommended Download:**

### MySQL Installer

for Windows

All MySQL Products. For All Windows Platforms.  
In One Package.

Starting with MySQL 5.6 the MySQL Installer package replaces the standalone MSI packages.

**Windows (x86, 32 & 64-bit), MySQL Installer MSI** [Go to Download Page >](#)

**Other Downloads:**

<b>Windows (x86, 64-bit), ZIP Archive</b> <small>(mysql-8.0.36-winx64.zip)</small>	8.0.36	232.3M	<a href="#">Download</a>
<small>MD5: a2f95f3625440e61d9012c5e2364bd79   <a href="#">Signature</a></small>			
<b>Windows (x86, 64-bit), ZIP Archive</b> <b>Debug Binaries &amp; Test Suite</b> <small>(mysql-8.0.36-winx64-debug-test.zip)</small>	8.0.36	663.5M	<a href="#">Download</a>
<small>MD5: 2f94487448f0227fc333d34c23ec8e81   <a href="#">Signature</a></small>			

We suggest that you use the [MD5 checksums](#) and [GnuPG signatures](#) to verify the integrity of the packages you download.





# MySQL 설치하기 (한글 사용자명)



**Login »**

using my Oracle Web account

**Sign Up »**

for an Oracle Web account

MySQL.com is using Oracle SSO for authentication. If you already have an Oracle Web account, click the Login link. Otherwise, you can signup for a free account by clicking the Sign Up link and following the instructions.

**No thanks, just start my download.**



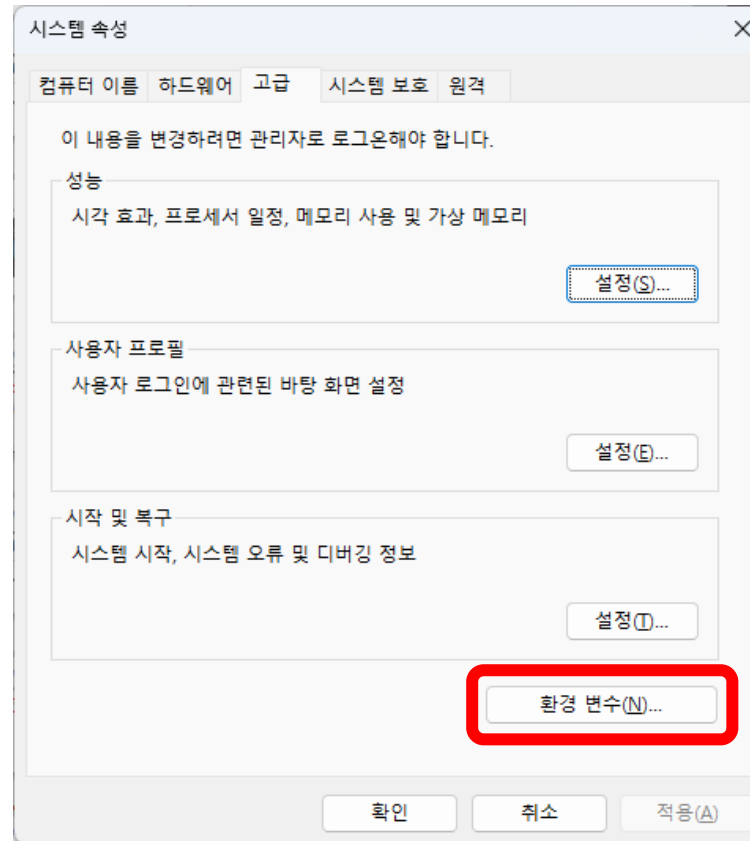
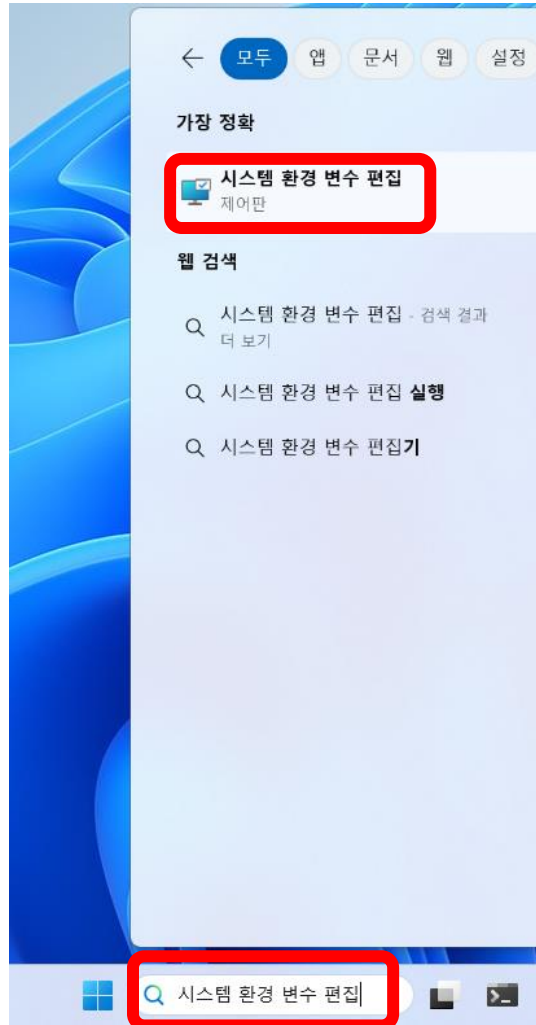
## MySQL 설치하기 (한글 사용자명)

mysql-8.0.36-winx64.zip을 C드라이브 하위에 압축풀기

C:\mysql-8.0.36-winx64\bin 이 존재하도록!

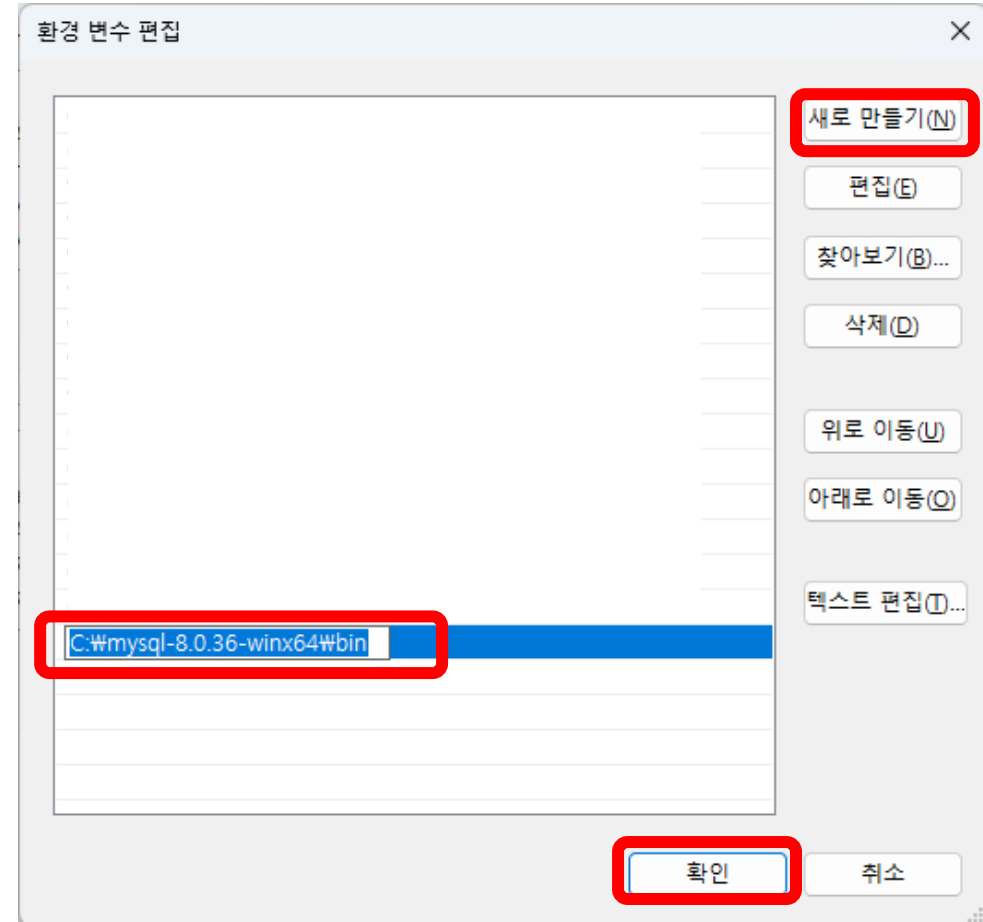
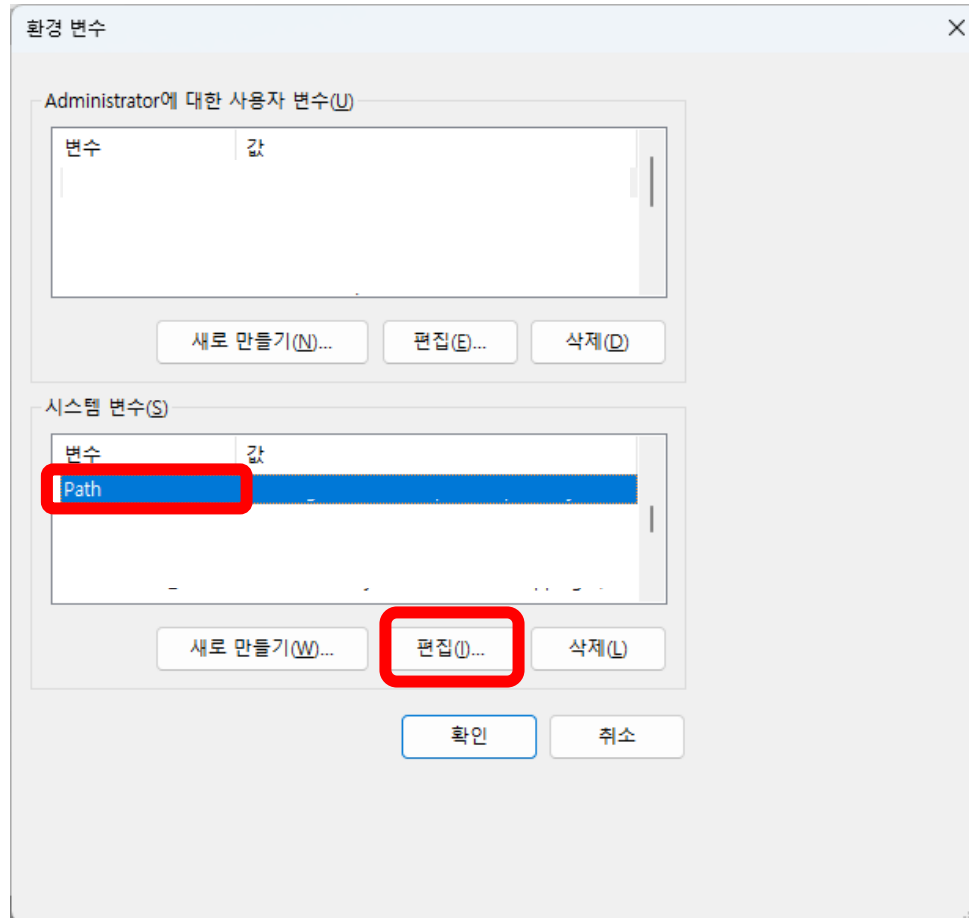


# MySQL 설치하기 (한글 사용자명)





# MySQL 설치하기 (한글 사용자명)





# MySQL 설치하기 (한글 사용자명)

명령 프롬프트(cmd)를 관리자 모드로 실행

mysql 경로로 이동

```
cd C:/mysql-8.0.36-winx64/bin
```

초기화

```
C:/mysql-8.0.36-winx64/bin> mysqld --initialize-insecure
```

설치

```
C:/mysql-8.0.36-winx64/bin> mysql --install
```



# MySQL 설치하기 (한글 사용자명)

mysql 실행

```
C:/mysql-8.0.36-winx64/bin> net start mysql
```

mysql 접속 (패스워드 요구 시 enter)

```
C:/mysql-8.0.36-winx64/bin> mysql -u root -p
```

비밀번호 변경

```
alter user 'root'@'localhost' identified by '1234'
```

mysql 종료

```
exit
```



# MySQL 설치하기 (한글 사용자명)



workbench 설치

<https://dev.mysql.com/downloads/workbench/>

[General Availability \(GA\) Releases](#) [Archives](#) [Info](#)

## MySQL Workbench 8.0.36

Select Operating System:


**Recommended Download:**

### MySQL Installer for Windows

All MySQL Products. For All Windows Platforms.  
In One Package.

Starting with MySQL 5.6 the MySQL Installer package replaces the standalone MSI packages.


**Windows (x86, 32 & 64-bit), MySQL Installer MSI**

[Go to Download Page >](#)

**Other Downloads:**

<b>Windows (x86, 64-bit), MSI Installer</b> <small>(mysql-workbench-community-8.0.36-winx64.msi)</small>	8.0.36	42.0M	<a href="#">Download</a>
---	--------	-------	--------------------------

MD5: 2156fe0cb6f5ed83908e463666b350a | Signature

 We suggest that you use the [MD5 checksums](#) and [GnuPG signatures](#) to verify the integrity of the packages you download.



# MySQL 설치하기 (한글 사용자명)



**Login »**

using my Oracle Web account

**Sign Up »**

for an Oracle Web account

MySQL.com is using Oracle SSO for authentication. If you already have an Oracle Web account, click the Login link. Otherwise, you can signup for a free account by clicking the Sign Up link and following the instructions.

**No thanks, just start my download.**





## 명령 프롬프트로 mysql 실행하기

MySQL 루트 계정 접속 :

```
C:\Users\Administrator> mysql -u root -p  
Enter password : 1234
```

데이터베이스 목록 보기 :

```
MySQL [(none)]> SHOW DATABASES;
```

MySQL 접속 종료 :

```
MySQL [(none)]> EXIT;
```



## 실습을 위한 예제 데이터 세팅

예제 데이터 파일 다운로드 : sample.sql

sample.sql이 위치한 경로에서 예제 데이터 파일 실행(명령 프롬프트):

```
C:\Users\Administrator> mysql -u root -p  
Enter password : 1234
```

```
MySQL [(none)]> source sample.sql
```



## 실습을 위한 예제 데이터 세팅

데이터베이스 목록 보기 :

```
MySQL [(none)]> SHOW DATABASES;
```

데이터베이스 선택하기 :

```
MySQL [(none)]> USE emp_db;
```

테이블 목록 보기 :

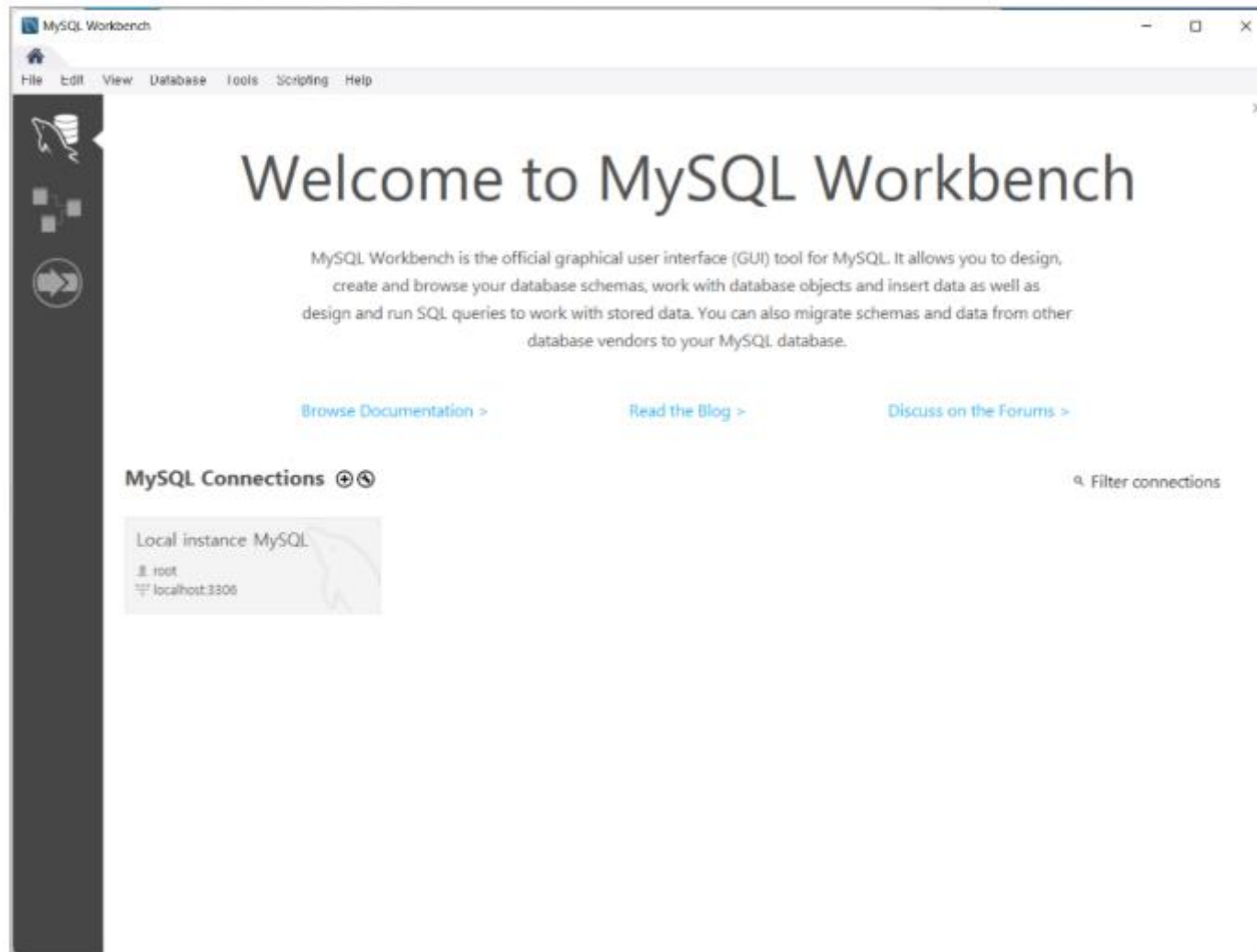
```
MySQL [emp_db]> SHOW TABLES;
```

MySQL 접속 종료 :

```
MySQL [emp_db]> EXIT;
```

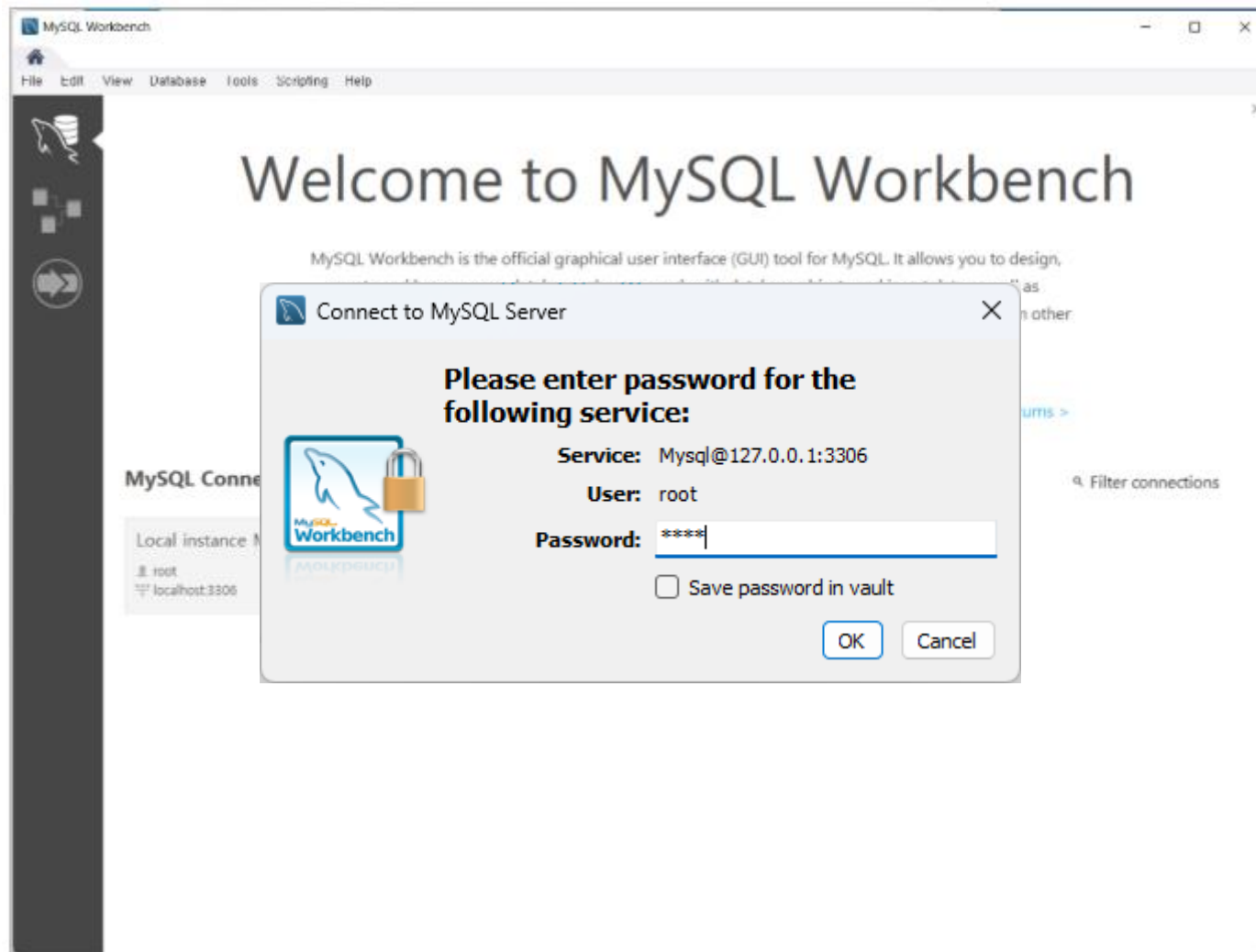


# Workbench 실행



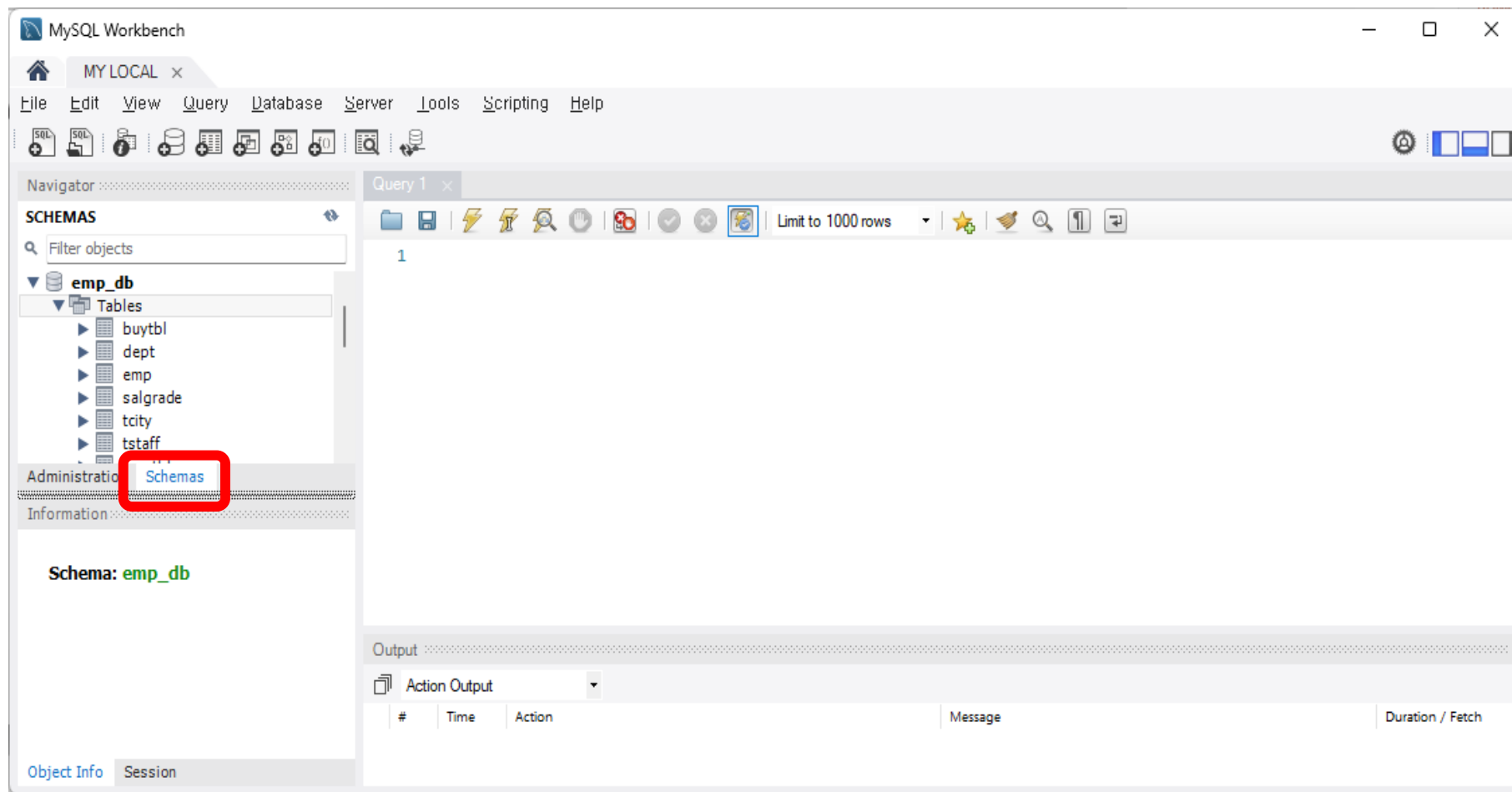


# Workbench 실행





# Workbench 실행





## Workbench 실행

Workbench 메뉴의 [Edit] >> [Preferences] 선택

왼쪽에서 [SQL Editor] 선택

오른쪽 제일 아래 <"Safe Updates...">체크 해제

Workbench 종료



## SQL의 특징

다른 프로그래밍 언어들과 다르게 예약어의 대소문자를 구분하지 않는다.  
(단, 데이터베이스에 저장된 내용은 대소문자를 구분)

SQL에서 '='는 "같다"의 의미로 사용된다.

명령문의 끝맺음은 ';' (세미콜론)을 반드시 사용한다.

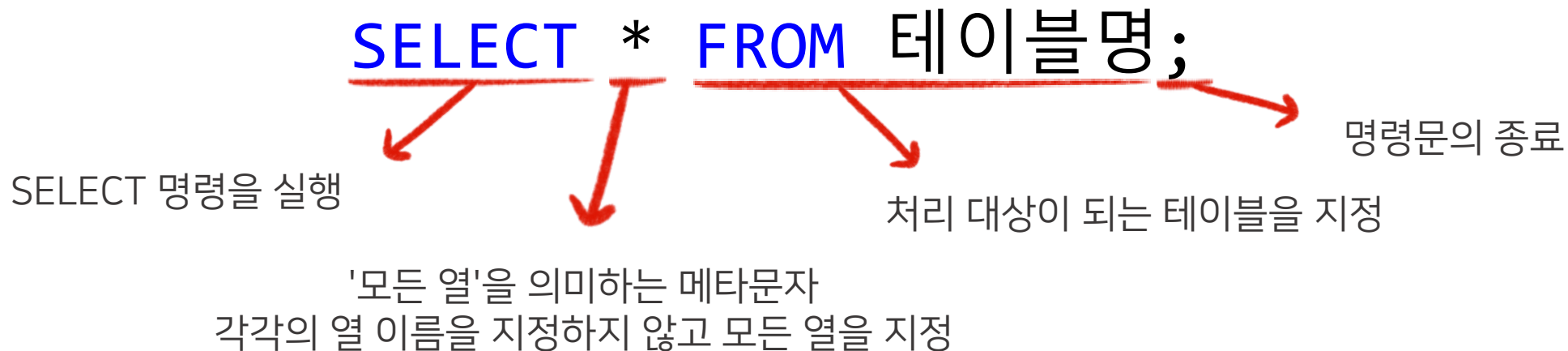




## 테이블 조회 - SELECT

SELECT :

DML에 속하는 명령으로 데이터베이스의 데이터를 읽어올 수 있다.  
이를 '조회', '질의' 또는 '쿼리'라 부르기도 한다.  
SQL에서 가장 자주 사용된다.





## 테이블 조회 - SELECT

```
SELECT [ALL | DISTINCT] 열이름  
FROM 테이블명  
[WHERE 조건]  
[GROUP BY 열이름]  
[HAVING 검색조건]  
[ORDER BY [열이름] [ASC | DESC]];
```

실행 순서 : FROM -> WHERE -> GROUP BY -> HAVING -> SELECT -> ORDER BY



## 테이블 조회 - SELECT

```
C:\ 명령 프롬프트
Microsoft Windows [Version 10.0.22621.3155]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Administrator>chcp 65001|
```

```
C:\ 명령 프롬프트 - mysql -u root
Active code page: 65001

C:\Users\Administrator>mysql -u root -p
Enter password: ****
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 109
Server version: 8.0.31 MySQL Community Server - GPL

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MySQL [(none)]> USE emp_db;
Database changed
```



# 테이블 조회 - SELECT

```
MySQL [emp_db]> SHOW TABLES;
```

Tables_in_emp_db
buytbl
dept
emp
salgrade
tcity
tstaff
usertbl

```
7 rows in set (0.002 sec)
```

```
MySQL [emp_db]> SELECT * FROM tcity;
```

name	area	popu	metro	region
부산	765	342	y	경상
서울	605	974	y	서울
순천	910	27	n	전라
오산	42	21	n	경기
전주	205	65	n	전라
청주	940	83	n	충청
춘천	1116	27	n	강원
홍천	1819	7	n	강원

```
8 rows in set (0.001 sec)
```



## 테이블 구조와 자료형

열 (컬럼/필드/속성/애트리뷰트)

name	area	popu	metro	region
부산	765	342	y	경상
서울	605	974	y	서울
순천	910	27	n	전라
오산	42	21	n	경기
전주	205	65	n	전라
청주	940	83	n	충청
춘천	1116	27	n	강원
홍천	1819	7	n	강원

SELECT 명령을 실행하면,  
표(테이블) 형태의 데이터가 출력된다.

각각의 열은 오직 하나의 자료형만 가지게 된다.

자료형은 숫자, 문자, 날짜 등이 있으며,  
데이터를 식별하는 분류체계이다.

행 (튜플/레코드)



## 테이블 구조와 자료형

테이블의 구조와 속성 파악 (Describe) :

MySQL [emp\_db]> DESC tcity;

- Field : 열 이름
- Type : 자료형
- Null : NULL 허용 여부
- Key : PK 또는 Unique 여부
- Default : 기본값
- Extra : 기타 정보



## 테이블 구조와 자료형

INTEGER, INT(BIGINT, MEDIUMINT, SMALLINT, TINYINT) DECIMAL, FLOAT, DOUBLE, REAL, CHAR, VARCHAR, NCHAR, NVARCHAR, TEXT(LONGTEXT, MEDIUMTEXT, TINYTEXT), DATE, TIME, DATETIME, TIMESTAMP, YEAR, BLOB(LONGBLOB, MEDIUMBLOB, TINYBLOB), BINARY, VARBINARY, GEOMETRY, GEOMETRYCOLLECTION, LINESTRING, MULTILINESTRING, POINT, MULTIPOINT, POLYGON, MULTIPOLYGON, JSON, BIT, BOOLEAN, ENUM, SET...

- INTEGER 형 : 수치 자료형 중 하나로 정수값을 저장할 수 있는 자료형
- CHAR : 고정된 길이의 문자열을 저장할 수 있는 자료형
- VARCHAR : 데이터의 크기에 맞춰 저장공간의 크기가 변경되는 가변 길이 문자열 자료형
- DATE : 2022년 11월 7일과 같이 연월일의 데이터를 저장할 수 있는 날짜 자료형
- TIME : 7시 10분 25초와 같이 시분초의 데이터를 저장할 수 있는 시간 자료형



## 테이블 검색

- 열 지정 검색

`SELECT` 열이름1, 열이름2, 열이름3, ... `FROM` 테이블명;





## 테이블 검색

- tcity 테이블에서 데이터 조회 (region, name, area 컬럼만 조회)



## 테이블 검색

- 별명 (alias)

SELECT문으로 조회하게 되면, 테이블 형태의 결과가 나타난다.

가장 위에는 컬럼명이 나타나는데,  
간혹 테이블의 컬럼명이 사용자가 읽기에는 직관적이지 못한 경우가 있다.  
이런 경우에는 별명을 지정해서, 컬럼명 대신 별명을 출력할 수 있다.

별명은 고유해야 하며, AS는 생략 가능하다.

만약 별명에 공백, 특수문자, 대문자가 있거나  
한글로 지정하는 경우에는 백틱(`)을 사용해 지정하는 것을 권장한다.



# 테이블 검색

- 별명 (alias)

```
SELECT      name AS `도시명`, popu AS `인구(만명)`  
FROM        tcity  
WHERE       metro = 'n';
```

```
SELECT      name `도시명`, popu `인구(만명)`  
FROM        tcity  
WHERE       metro = 'n';
```



# 테이블 검색

## - 계산

데이터가 조회되는 방식을 수정하거나 계산을 수행하고자 할 때, 산술표현식을 사용한다.

SELECT 문에서 산술표현식은 FROM절을 제외한 모든 절에서 사용 가능하다.

```
SELECT      name `도시명`, popu*10000 `인구수`  
FROM        tcity  
WHERE       metro = 'n';
```



## 테이블 검색

- tcity 테이블에서 데이터 조회 (name, area, popu, 인구밀도 조회)  
인구밀도는 인구수를 면적으로 나눈 값이다.
- SELECT문을 통해 1년은 몇 초인지 계산



# 테이블 검색

- 컬럼 연결(concat)

concat(str1, str2, ...)

매개변수를 연결한 문자열을 반환하는 함수  
한 개 이상의 매개변수를 가질 수 있다.

```
SELECT      concat(name, " ", grade)
FROM        tstaff;
```



# 테이블 검색

## - DISTINCT

특별히 명시되지 않는다면 SELECT문은 중복되는 행을 제거하지 않고, 결과를 반환한다.

결과에서 중복되는 행을 제거하기 위해서는 SELECT 뒤에 DISTINCT를 입력해야 한다.

```
SELECT  
    region  
FROM  
    tcity;
```

```
SELECT DISTINCT  
    region  
FROM  
    tcity;
```



# 테이블 검색

## - 정렬 (ORDER BY)

SELECT 문에서 순서에 대한 지정이 없는 경우에는 DBMS의 순서를 따른다.  
(Oracle은 데이터의 입력 순서, MySQL은 기본키에 대한 오름차순 정렬)

기본형식 : ORDER BY 컬럼명 [ASC | DESC]

오름차순인 경우에는 ASC, 내림차순인 경우에는 DESC를 지정한다.  
기본값은 ASC로 생략이 가능하다.

오름차순 :

숫자는 작은 값이 먼저 조회

날짜는 빠른 값이 먼저 조회

문자는 알파벳 순서, 가나다 순서 조회

NULL 값은 제일 나중에 조회





## 정렬 - ORDER BY

### - 오름차순 정렬

```
SELECT 열이름1, 열이름2, ...  
FROM 테이블명  
WHERE 조건식  
ORDER BY 열이름;
```

```
SELECT 열이름1, 열이름2, ...  
FROM 테이블명  
WHERE 조건식  
ORDER BY 열이름 ASC;
```

### - 내림차순 정렬

```
SELECT 열이름1, 열이름2, ...  
FROM 테이블명  
WHERE 조건식  
ORDER BY 열이름 DESC;
```



# 테이블 검색

## - 정렬 (ORDER BY)

2개 이상의 컬럼으로 정렬을 지정할 수도 있다.

이 경우에는 첫번째 기준 컬럼의 값이 같으면 두번째 기준 컬럼을 통해 정렬을 하게 된다.

```
SELECT * FROM tcity ORDER BY popu;
```

```
SELECT * FROM tcity ORDER BY popu DESC;
```

```
SELECT * FROM tcity ORDER BY region, name DESC;
```

```
SELECT * FROM tcity ORDER BY 5, 1 DESC;
```

```
SELECT name AS `도시명`, popu AS `인구` FROM tcity ORDER BY `인구` DESC;
```



## 테이블 검색

- tstaff 테이블에서 salary가 적은 사람부터 순서대로 출력하되, salary가 같다면 score가 높은 사람을 먼저 조회



# 테이블 검색

## - 조건문 (WHERE)

WHERE 절을 이용해 조건을 지정할 수 있다.

WHERE 절은 SELECT 문 뿐만 아니라 삭제를 위한 DELETE, 수정을 위한 UPDATE 절에도 쓰인다.

### 행 지정 검색

**SELECT \* FROM** 테이블명 **WHERE** 조건식;

```
metro = 'y';  
region <> '전라';  
  
area > 1000;  
popu <= 100;  
  
score IS NULL;  
score IS NOT NULL;
```



# 테이블 검색

## - 조건문 (WHERE)

### 비교 연산자

- $A = B$
- $A > B, A < B$
- $A \geq B, \text{NOT } (A < B)$
- $A \leq B, \text{NOT } (A > B)$
- $A \neq B, A \neq B, \text{NOT } (A = B)$  [같지 않다]

문자열 또는 날짜의 비교는 따옴표로 감싸주어야 한다.



## 테이블 검색

- tcity 테이블에서 metro가 n인 데이터 조회 (name, popu 컬럼만 조회)
- tstaff 테이블에서 joindate가 2015년 이전인 데이터 조회 (name, depart, grade 컬럼만 조회)



## 테이블 검색

### - 조건문 (WHERE)

NULL은 값이 입력되어 있지 않은 특수한 상태로,  
값을 알 수 없거나, 아직 결정할 수 없다는 의미이기에 0, " "과는 다르다.

NULL 여부를 표현할 때는 = NULL, != NULL 이 아닌 IS NULL, IS NOT NULL을 사용한다.

```
SELECT * FROM tstaff WHERE score = NULL;
```

```
SELECT * FROM tstaff WHERE score IS NULL;
```

```
SELECT * FROM tstaff WHERE score != NULL;
```

```
SELECT * FROM tstaff WHERE score IS NOT NULL;
```



## 테이블 검색

### - 조건문 (WHERE)

#### 논리 연산자

두 개 이상의 조건을 동시에 점검할 때는 AND, OR 연산자를 사용한다.

AND는 두 조건이 모두 참인 결과를 반환하며,

OR는 두 조건 중 하나라도 참인 결과를 반환한다.

NOT 연산자를 통해 진위 여부를 반대로 변경할 수 있다.

```
SELECT `a`, `b`, `and` FROM logic_operation WHERE a=1 AND b=1;
```

```
SELECT `a`, `b`, `or` FROM logic_operation WHERE a=1 OR b=1;
```

```
SELECT `a` FROM logic_operation WHERE NOT a=1;
```





## 테이블 검색

- tstaff 테이블에서 salary가 300 미만이면서 score는 60 이상인 직원이 누구인지 조회
- tstaff 테이블에서 인사과 남자 직원과 영업부 여자 직원을 모두 조회



# 테이블 검색

## - 조건문 (WHERE)

### LIKE

등호(=)를 통한 비교 연산은 완전히 일치하는 조건식을 표현하는데 비해, LIKE 연산자는 와일드 카드를 사용한 패턴으로 부분 문자열을 검색

### 와일드카드

- % : 길이에 구애받지 않는 임의 문자열
- \_ : 한 개의 임의 문자

와일드카드에 사용되는 문자를 검색할 때는 \ 또는 ESCAPE를 이용



## 테이블 검색

- tcity 테이블에서 '천'이라는 글자가 들어가는 도시를 검색
- tcity 테이블에서 '천'이라는 글자가 들어가지 않는 도시를 검색
- tcity 테이블에서 '천'으로 끝나는 도시를 검색
- tcity 테이블에서 '천'으로 시작하는 도시를 검색
- promotion\_tbl 테이블에서 promotion\_msg에 '30%'라는 글자가 들어가는 상품 검색



## 테이블 검색

### - 조건문 (WHERE)

#### BETWEEN ~ AND

두 사이의 범위를 제한할 때는 BETWEEN ~ AND 구문을 사용한다.  
범위 조건은 수치값에 대해 사용하지만, 문자열이나 날짜 등에도 사용할 수 있다.

```
SELECT * FROM tCity WHERE popu BETWEEN 50 AND 100;
```

```
SELECT * FROM tCity WHERE popu >= 50 AND popu <= 100;
```

```
SELECT * FROM tStaff WHERE name BETWEEN '가' AND '사';
```

```
SELECT * FROM tStaff WHERE joindate BETWEEN '20150101' AND '20180101';
```



# 테이블 검색

## - 조건문 (WHERE)

IN

IN 연산자는 불연속적인 값 여러 개의 목록을 가지고, 목록에 요소와 일치하는 데이터를 조회한다.

NOT과 결합해서 사용 가능하다.

```
SELECT * FROM tcity WHERE region IN ('경상', '전라');
```

```
SELECT * FROM tcity WHERE region = '경상' OR region = '전라';
```



## 테이블 검색

- tcity 테이블에서 area가 500~1000 사이의 도시 목록을 조회
- tcity 테이블에서 region 필드가 경상 또는 전라가 아닌 모든 도시를 조회
- tstaff 테이블에서 성이 '이'씨 이거나 '안'씨인 직원을 조회
- tstaff 테이블에서 총무부나 영업부에 근무하는 직원을 조회
- tstaff 테이블에서 인사과나 영업부에 근무하는 대리를 조회



## 테이블 검색

- 행의 개수 제한

SELECT 문 맨 뒤에 LIMIT를 작성하여 행의 개수를 제한할 수 있다.

MySQL에서는 LIMIT를 이용하지만,  
ORACLE에서는 ROWNUM이라는 개념을 이용하고,  
MS-SQL에서는 TOP를 이용한다.

기본 형식 : LIMIT [건너뛸 개수,] 조회할 개수

건너뛸 개수를 생략하면 0이 기본값이다.

```
SELECT * FROM tcity ORDER BY area DESC LIMIT 4;
```



## 테이블 검색

- tstaff 테이블에서 salary가 높은 상위 5명의 직원을 조회
- tcity 테이블에서 area가 넓은 도시 중 앞의 2개는 건너뛰고 이후 3개의 도시를 조회
- tstaff 테이블에서 salary 순으로 내림차순 정렬한 후 12위에서 16위까지 조회





## 수치 연산

- WHERE 구에서 연산하기  
tcity 테이블에서 인구밀도( $\text{popu} * 10000 / \text{area}$ )가 1000 미만인 행 조회

```
SELECT * FROM tcity WHERE popu * 10000 / area < 1000;
```

```
SELECT *, popu * 10000 / area AS `인구밀도`  
FROM tcity  
WHERE popu * 10000 / area <= 1000;
```



**NO**

```
SELECT *, popu * 10000 / area AS `인구밀도`  
FROM tcity  
WHERE `인구밀도` < 1000;
```



## 수치 연산

- ORDER BY 구에서 연산하기  
tcity 테이블에서 인구밀도( $\text{popu} * 10000 / \text{area}$ )가 높은 순으로 데이터 조회

```
SELECT * FROM tcity ORDER BY popu * 10000 / area DESC;
```



YES

```
SELECT *, popu * 10000 / area AS `인구밀도`  
FROM tcity  
ORDER BY `인구밀도` DESC;
```



## 수치 연산

### - 연산 시 주의사항

일반 프로그래밍 연산과 달리, SQL에서 NULL의 연산은 모두 NULL이 된다.

```
SELECT  
  NULL * 1,  
  NULL - 1,  
  NULL + 1,  
  NULL / 1,  
  1 / NULL;
```

```
SELECT  
  NULL * NULL,  
  NULL - NULL,  
  NULL + NULL,  
  NULL / NULL,  
  NULL / NULL;
```



# 함수

## - 산술 함수

ABS(x): 인수 x의 절대값을 반환

CEILING(x): x보다 크거나 같은 최소 정수를 반환

FLOOR(x): x보다 작거나 같은 최대 정수를 반환

ROUND(x): x를 가장 가까운 정수로 반올림

CONV(x, y, z): x를 y진수에서 z 진수로 변환

PI(): 원주율  $\pi$  값을 반환

MOD(x, y): x를 y로 나눈 나머지를 반환

POW(x, y): x의 y 거듭제곱 값을 반환

SQRT(x): x의 제곱근을 반환

RAND(): 0과 1 사이의 무작위 숫자를 반환

SIGN(x): x가 양수면 1, 음수면 -1, 0이면 0을 반환

TRUNC(x): x의 소수점 이하를 잘라내어 정수 부분만 반환



# 함수

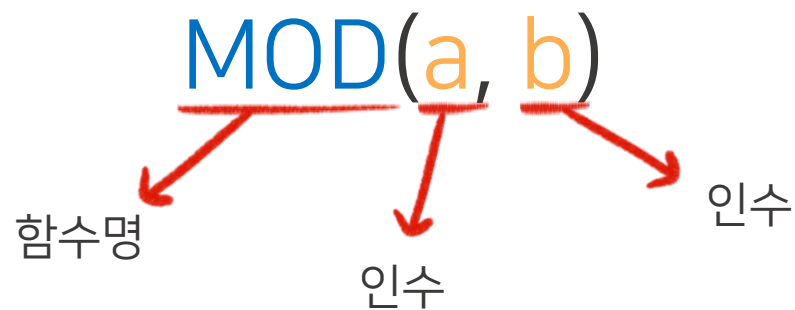
## - 함수 연산

함수명 뒤에 있는 괄호 안의 인수들이 연산 대상이 되어 함수명에 따른 연산을 진행합니다.

SELECT

10 % 3 AS `연산자 연산`,

MOD(10, 3) AS `함수 연산`;





# 함수

- 산술 함수 (ROUND)

$\text{ROUND}(a), \text{ROUND}(a, b)$

ROUND 함수는 두번째 인수에서 지정하는 자리에 맞춰 첫번째 인수를 반올림합니다.

두번째 인수를 생략하면 0이 되어, 소수점 첫째자리에서 반올림을 합니다.

두번째 인수에 음수를 넣으면 정수 자리(1의 자리, 10의 자리 등)의 반올림이 가능합니다.

\* 같은 방법으로 TRUNC 함수를 사용하면 버림 또한 가능합니다.



## 함수

- emp 테이블에서 EMPNO가 홀수인 데이터의 모든 컬럼을 조회
- tstaff 테이블에서 score를 소수점 첫째자리에서 반올림하여 데이터를 조회



# 함수

## - 문자열 함수

LENGTH(str): 문자열의 길이를 문자 수로 반환

CONCAT(str1, str2, ...): 여러 문자열을 결합하여 하나의 문자열로 반환

CONCAT\_WS(separator, str1, str2, ...): 여러 문자열을 결합하여 하나의 문자열로 반환

INSTR(str, substr), LOCATE(substr, str): 문자열에서 서브 문자열의 위치를 찾아 반환

REPEAT(str, count): 문자열을 지정된 횟수만큼 반복하여 반환

REPLACE(str, oldstr, newstr): 문자열에서 oldstr을 newstr로 모두 치환

REVERSE(str): 문자열을 뒤집어 반환

UPPER(str), LOWER(str): 문자열을 대문자 또는 소문자로 변환

LTRIM(str), RTRIM(str), TRIM(str): 문자열의 왼쪽, 오른쪽, 또는 양쪽의 공백을 제거

LEFT(str, len), RIGHT(str, len): 문자열의 왼쪽 또는 오른쪽에서 len만큼의 문자를 반환

SUBSTRING(str, start, length) : 문자열에서 일부를 추출 (start는 1부터 시작)





# 함수

- 문자열 결합 함수 (CONCAT)

CONCAT(열이름, 열이름)

CONCAT 함수는 문자열을 결합합니다.

```
SELECT CONCAT(quantity, unit) FROM product;
```



## 함수

- usertbl 테이블에서 name과 name의 길이를 조회
- emp 테이블의 hiredate 컬럼을 통해 4자리 연도, 2자리 달을 출력



# 함수

## - 날짜 함수

ADDDATE(date, INTERVAL expr unit) : 지정된 날짜(date)에 특정 기간(expr unit)을 더함

SUBDATE(date, INTERVAL expr unit) : 지정된 날짜(date)에 특정 기간(expr unit)을 뺌

ADDTIME(time, INTERVAL expr unit) : 지정된 시간(time)에 특정 기간(expr unit)을 더함

SUBTIME(time, INTERVAL expr unit): 지정된 시간(time)에 특정 기간(expr unit)을 뺌

CURRENT\_DATE(), CURDATE(): 현재 날짜를 반환

CURRENT\_TIME(), CURTIME(): 현재 시간을 반환

NOW(), LOCALTIME(), LOCALTIMESTAMP(), CURRENT\_TIMESTAMP(): 현재 날짜와 시간을 반환

YEAR(date), MONTH(date), DAY(date) : 주어진 날짜에서 각각의 구성 요소를 반환

hour(time), MINUTE(time), SECOND(time), 주어진 시간에서 각각의 구성 요소를 반환



# 함수

## - 날짜 함수

DAYOFWEEK(date): 주어진 날짜의 요일을 반환 (1 = 일요일, 2 = 월요일, ..., 7 = 토요일)

QUARTER(date): 주어진 날짜의 분기(quarter)를 반환

STR\_TO\_DATE(str, format): 문자열로 표시된 날짜를 날짜 형식으로 변환

DATEDIFF(date1, date2): 두 날짜 사이의 일 수 차이를 반환

TIMEDIFF(time1, time2): 두 시간 사이의 시간 차이를 반환

LAST\_DAY(date): 주어진 날짜의 해당 월의 마지막 날짜를 반환



# 함수

- 날짜의 덧셈과 뺄셈 (INTERVAL)

```
SELECT CURRNET_TIME + INTERVAL 3600 SECOND;  
SELECT CURRNET_TIME + INTERVAL 60 MINUTE;  
SELECT CURRNET_TIME + INTERVAL 1 HOUR;
```

```
SELECT CURRNET_DATE - INTERVAL 365 DAY;  
SELECT CURRNET_DATE - INTERVAL 12 MONTH;  
SELECT CURRNET_DATE - INTERVAL 1 YEAR;
```



## 함수

- tstaff 테이블에서 오늘 날짜까지 입사 후 며칠이 지났는지 조회
- usertbl 테이블에서 birthyear와 mdate 컬럼을 사용해, 만 나이 조회



# 함수

## - NULL 관련 함수

IFNULL(x, y) : x 값이 NULL이 아니면 x를 반환하고, x 값이 NULL이면 y를 반환

NULLIF(x, y) : x 값과 y 값이 같으면 NULL을 반환하고, 다르면 x를 반환

COALESCE(x, y, z, ...) : 인자 값들 중 최초 NULL이 아닌 값을 반환, 모두 NULL이면 NULL 반환

```
SELECT IFNULL(10,20);  
SELECT IFNULL(NULL,20);  
SELECT IFNULL(10/'S',20);
```

```
SELECT NULLIF(10,10);  
SELECT NULLIF(10,NULL);
```

```
SELECT COALESCE(NULL,1);
```



# 함수

## - 제어 흐름 함수

IF 함수 : 수식이 참 또는 거짓인지 결과에 따라서 2중 분기

```
SELECT IF(10 > 20, '참', '거짓');
```

CASE ~ WHEN ~ ELSE ~ END 연산

```
SELECT  
    CASE 10  
    WHEN 1 THEN '일'  
    WHEN 5 THEN '오'  
    WHEN 10 THEN '십'  
    ELSE '모름'  
END;
```





## 함수

- tstaff 테이블에서  
salary가 300 이상이면 salary의 30%  
그렇지 않으면 salary의 50%로 계산한 성과급 컬럼을 추가하여 데이터 조회
- tstaff 테이블에서  
grade가 사원이면 100, 대리이면 200, 나머지는 300으로 성과급 컬럼을 추가하여 데이터 조회



# CASE

## - CASE

CASE문은 조건에 따른 값을 지정하기 위해 사용합니다.  
이를 통해 기존 연산자와 함수로 처리가 불가능한 것들 중 간단한 것들을 처리할 수 있습니다.

CASE 문은 '검색 CASE'와 '단순 CASE' 로 나눌 수 있습니다.

CASE

```
WHEN 조건1 THEN 값1,  
WHEN 조건2 THEN 값2,  
...  
ELSE 값3
```

END

CASE 식1

```
WHEN 식2 THEN 식3,  
WHEN 식4 THEN 식5,  
...  
ELSE 식6
```

END



# CASE

## - CASE

tstaff 테이블에서 score가 null인 경우, 0으로 바꾸어 출력

```
SELECT
    *,
    CASE
        WHEN score IS NULL THEN 0
        ELSE score
    END
FROM tstaff;
```

```
SELECT
    *,
    IFNULL(score, 0)
FROM tstaff;
```



# CASE

## - CASE

tstaff 테이블에서 성별(gender)가 '남'인 경우 1, '여'인 경우 2를 출력

```
SELECT
    *,
    CASE gender
        WHEN '남' THEN 1
        WHEN '여' THEN 2
        ELSE '알 수 없음'
    END AS `성별`
FROM tstaff;
```



# CASE

## - CASE

SELECT구 뿐만 아니라 WHERE구와 ORDER BY구에서도 사용할 수 있습니다.

ELSE를 생략하면 자동적으로 ELSE null로 처리됩니다.

단순 CASE문에서의 NULL 값 비교는 불가능합니다.



# CASE

```
SELECT *  
FROM tstaff  
WHERE  
    CASE grade  
        WHEN '사원' THEN 1  
        WHEN '대리' THEN 2  
        WHEN '차장' THEN 3  
        WHEN '과장' THEN 4  
        WHEN '부장' THEN 5  
        WHEN '이사' THEN 6  
        ELSE NULL  
    END = 6;
```

```
SELECT *  
FROM tstaff  
ORDER BY  
    CASE grade  
        WHEN '사원' THEN 1  
        WHEN '대리' THEN 2  
        WHEN '차장' THEN 3  
        WHEN '과장' THEN 4  
        WHEN '부장' THEN 5  
        WHEN '이사' THEN 6  
        ELSE NULL  
    END;
```



# CASE



YES

SELECT

\*

,

CASE

WHEN score IS NULL THEN 0

ELSE score

END AS `점수(null is 0)`

FROM tstaff;



NO

SELECT

\*

,

CASE score

WHEN NULL THEN 0

ELSE score

END AS `점수(null is 0)`

FROM tstaff;



## 그룹화 - GROUP BY

GROUP BY :

집계함수는 그룹화와 함께 사용되어, SELECT 문의 활용범위를 넓힐 수 있다.

GROUP BY구에는 그룹화할 열을 지정하며, 복수의 컬럼 지정도 가능하다.

```
SELECT grade, AVG(salary) AS `직급별 평균 임금`  
FROM tstaff  
GROUP BY grade  
ORDER BY `직급별 평균 임금` ASC;
```

```
SELECT depart, gender, COUNT(*)  
FROM tStaff  
GROUP BY depart, gender  
ORDER BY depart, gender;
```





## 데이터 집계

- 집계함수 : 복수의 값에서 하나의 값을 계산해내는 함수

데이터를 그룹화해서 통계를 계산해주는 함수로 숫자나 날짜 데이터에 사용된다.  
문자열 데이터는 최소값과 최대값 조회만 가능하다.

SUM() : 그룹의 누적 합계를 반환

AVG() : 그룹의 평균을 반환

COUNT() : 그룹의 총 개수를 반환

MAX() : 그룹의 최대값을 반환

MIN() : 그룹의 최소값을 반환

STDDEV(): 그룹의 표준편차를 반환

VARIANCE(): 그룹의 분산을 반환



## 데이터 집계

- 집계함수 (SUM)

### SUM(수치 자료형의 열 이름)

SUM 함수는 수치형 데이터의 합계를 계산합니다.

\* 문자열 자료형 열의 합계와 날짜 자료형 열의 합계를 구해봅시다.



## 데이터 집계

- 집계함수 (AVG)

### AVG(수치 자료형의 열 이름)

AVG 함수는 수치형 데이터의 평균을 계산합니다.

\* 문자열 자료형 열의 합계와 날짜 자료형 열의 평균을 구해봅시다.



## 데이터 집계

- 집계함수 (COUNT)

### COUNT(열이름)

집계함수 : 복수의 값에서 하나의 값을 계산해내는 함수

COUNT 함수는 주어진 집합의 '개수'를 구해 반환합니다. (행의 개수)

\* NULL은 개수에 포함되지 않습니다.



## 데이터 집계

- 집계함수 (MIN, MAX)

MIN(열이름), MAX(열이름)

MIN 함수와 MAX 함수는 집합에서 최소값과 최대값을 구합니다.

\* NULL 값은 무시됩니다. 즉, 최소값도, 최대값도 아닙니다.



## 데이터 집계

- tstaff 테이블이 가진 데이터 개수(총 직원수) 조회
- tstaff 테이블에서 급여가 400 이상인 직원 수 조회
- tstaff 테이블에서 직급별 직원수 조회
- tstaff 테이블에서 직급 종류 개수 조회
- tstaff 테이블에서 score 값이 없는 직원 수 조회
- tstaff 테이블에서 인사과의 평균 급여 조회
- tcity 테이블에서 인구의 총합과 평균을 조회
- tcity 테이블에서 면적의 최소값과 최대값을 조회



## 그룹화 조건 - HAVING

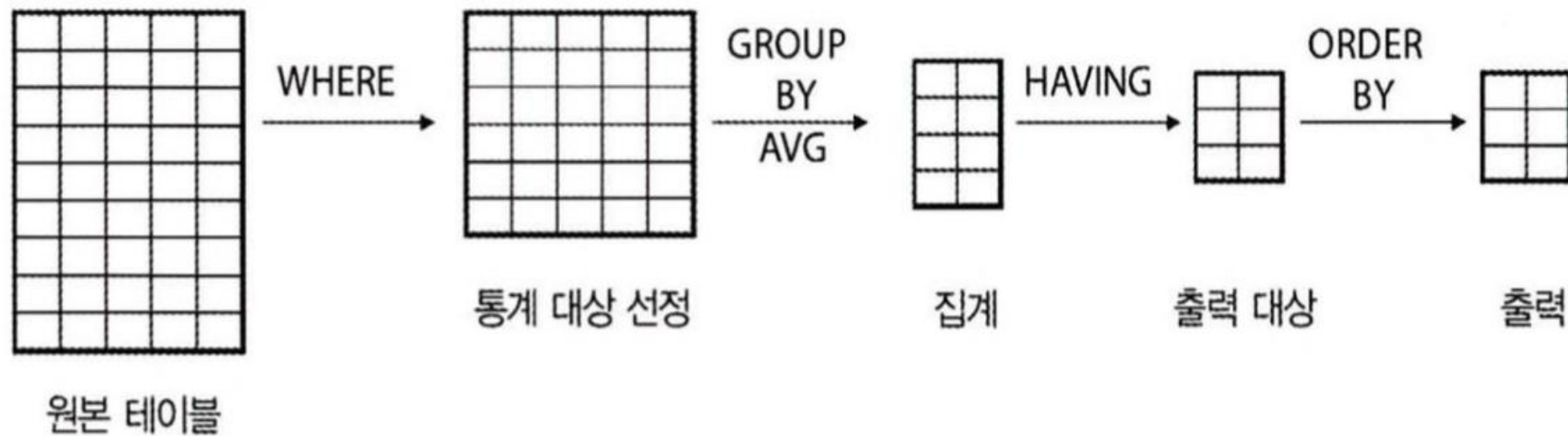
HAVING:

GROUP BY와 함께 쓰이며, 출력할 그룹의 조건을 지정한다.

```
SELECT
    depart, AVG(salary)
FROM
    tStaff
GROUP BY
    depart
HAVING
    AVG(salary) >= 350
ORDER BY
    AVG(salary);
```



## SELECT문이 실행되는 과정



실행 순서 : FROM -> WHERE -> GROUP BY -> HAVING -> SELECT -> ORDER BY





## 데이터 집계

- EMP 테이블에서 인원수, 최대 급여(sal), 최소 급여, 급여의 합을 계산하여 출력
- EMP 테이블에서 각 업무별(job)로 최대 급여(sal), 최소 급여, 급여의 합을 출력
- EMP 테이블에서 업무별(job) 인원수를 구하여 출력
- EMP 테이블에서 최고 급여(sal)와 최소 급여의 차이는 얼마인지 출력