

## Chapter 3

# MySQL 기초

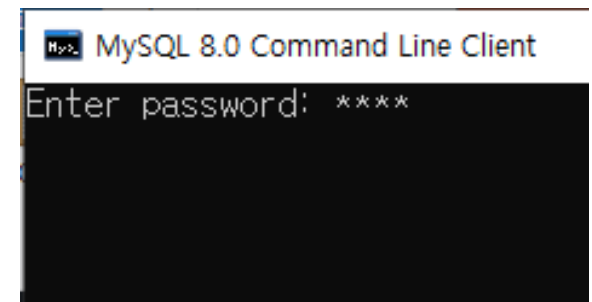
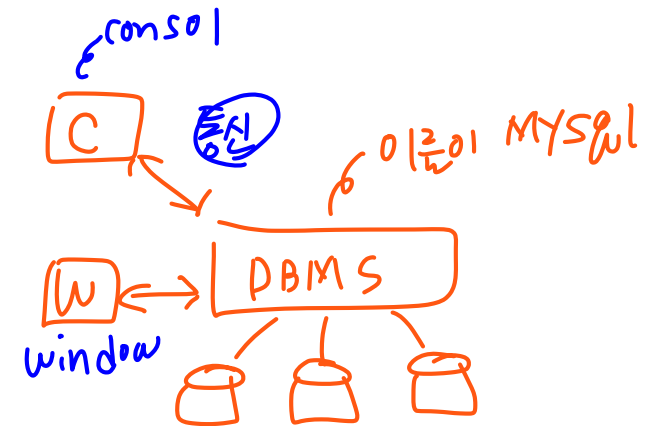
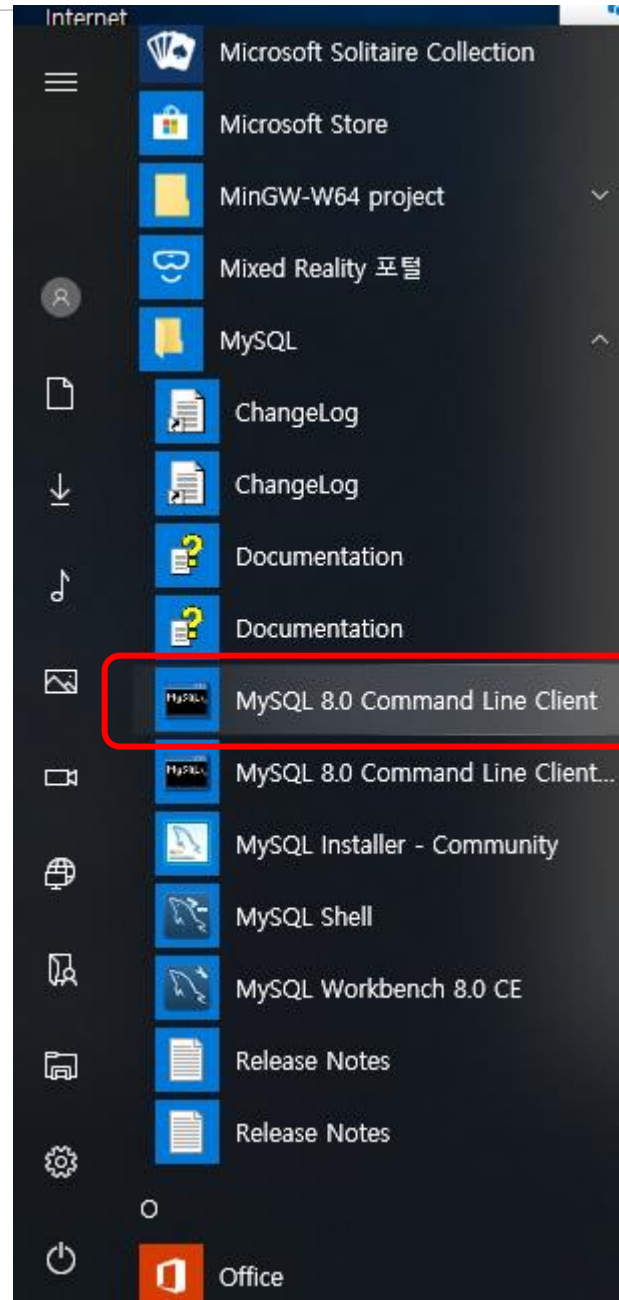
오 세 종

# 1. MySQL 특징

- SQL에 기반을 둔 관계형 DBMS 중 하나
- Oracle, IBM, Infomix 등의 데이터베이스는 고가이지만, MySQL 데이터베이스는 무료
- Oracle 에 인수된 이후에 Maria DB 등장
- 리눅스, 유닉스, 윈도우 등 거의 모든 운영체제에서 사용가능
- 처리 속도가 상당히 빠르고 대용량에 데이터도 처리 용이
- 설치 방법이 쉽고 초보자도 익히기 쉬움
- 보안성이 우수

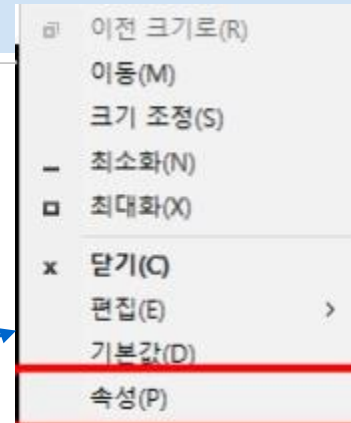
## 2. MySQL – 콘솔에서 사용하기

- 콘솔 열기



## 2. MySQL – 콘솔에서 사용하기

마우스 오른쪽 버튼 클릭  
(글씨크기, 바탕색등 조절)



```
MySQL 8.0 Command Line Client
Enter password: ****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 140
Server version: 8.0.15 MySQL Community Server - GPL

Copyright (c) 2000, 2019, Oracle and/or its affiliates. All rights reserved.

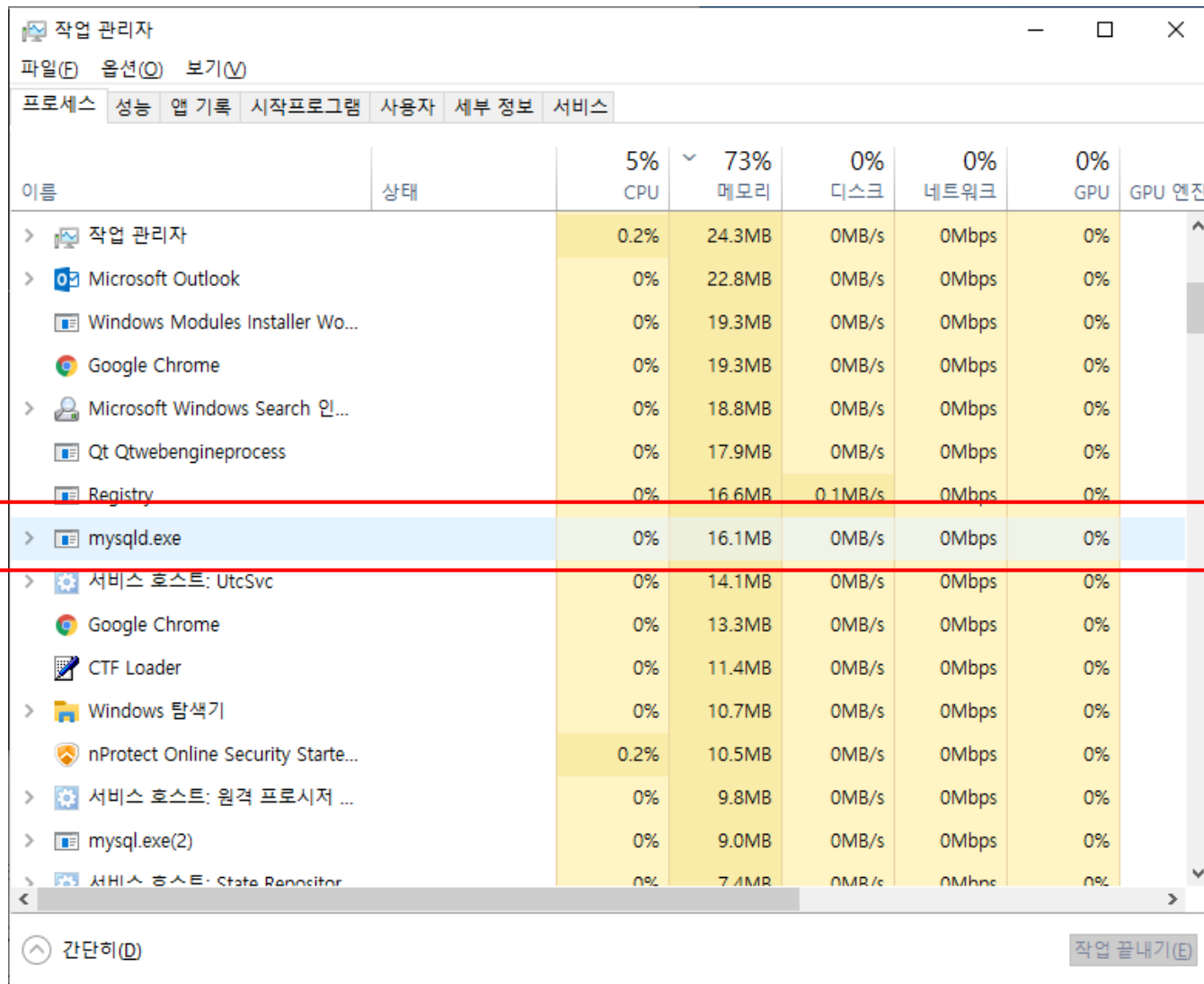
Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

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

mysql> _
```

## 2. MySQL – 콘솔에서 사용하기

- Mysql 서버가 실행중임



작업 관리자

파일(F) 옵션(O) 보기(V)

프로세스 성능 열 기록 시작프로그램 사용자 세부 정보 서비스

| 이름                                 | 상태 | 5% CPU | 73% 메모리 | 0% 디스크  | 0% 네트워크 | 0% GPU | GPU 엔진 |
|------------------------------------|----|--------|---------|---------|---------|--------|--------|
| > 작업 관리자                           |    | 0.2%   | 24.3MB  | 0MB/s   | 0Mbps   | 0%     |        |
| > Microsoft Outlook                |    | 0%     | 22.8MB  | 0MB/s   | 0Mbps   | 0%     |        |
| Windows Modules Installer Wo...    |    | 0%     | 19.3MB  | 0MB/s   | 0Mbps   | 0%     |        |
| Google Chrome                      |    | 0%     | 19.3MB  | 0MB/s   | 0Mbps   | 0%     |        |
| > Microsoft Windows Search 인...    |    | 0%     | 18.8MB  | 0MB/s   | 0Mbps   | 0%     |        |
| Qt Qtwebengineprocess              |    | 0%     | 17.9MB  | 0MB/s   | 0Mbps   | 0%     |        |
| Registry                           |    | 0%     | 16.6MB  | 0.1MB/s | 0Mbps   | 0%     |        |
| > mysql.exe                        |    | 0%     | 16.1MB  | 0MB/s   | 0Mbps   | 0%     |        |
| > 서비스 호스트: UtcSvc                  |    | 0%     | 14.1MB  | 0MB/s   | 0Mbps   | 0%     |        |
| Google Chrome                      |    | 0%     | 13.3MB  | 0MB/s   | 0Mbps   | 0%     |        |
| CTF Loader                         |    | 0%     | 11.4MB  | 0MB/s   | 0Mbps   | 0%     |        |
| > Windows 탐색기                      |    | 0%     | 10.7MB  | 0MB/s   | 0Mbps   | 0%     |        |
| nProtect Online Security Starte... |    | 0.2%   | 10.5MB  | 0MB/s   | 0Mbps   | 0%     |        |
| > 서비스 호스트: 원격 프로시저 ...             |    | 0%     | 9.8MB   | 0MB/s   | 0Mbps   | 0%     |        |
| > mysql.exe(2)                     |    | 0%     | 9.0MB   | 0MB/s   | 0Mbps   | 0%     |        |
| > 서비스 호스트: State Repositor         |    | 0%     | 7.4MB   | 0MB/s   | 0Mbps   | 0%     |        |

간단히(D) 작업 끝내기(E)

## 2. MySQL – 콘솔에서 사용하기

- 설치된 데이터베이스 보기

```
mysql> show databases ;
+-----+
| Database |
+-----+
| information_schema |
| mysql      |
| performance_schema |
| sakila     |
| sys        |
| world      |
+-----+
6 rows in set (0.00 sec)

mysql>
```

내 컴퓨터 내 database 목록

## 2. MySQL – 콘솔에서 사용하기

- world 데이터베이스 내용보기

```
mysql> use world ;
Database changed
mysql> show tables ;
+-----+
| Tables_in_world |
+-----+
| city             |
| country          |
| countrylanguage |
+-----+
3 rows in set (0.01 sec)

mysql>
```

사용할 데이터베이스 선택  
→ working space가 world DB를 선택

## 2. MySQL – 콘솔에서 사용하기

- city 테이블 구조 보기

```
mysql> desc city ;
```

| Field       | Type     | Null | Key | Default | Extra          |
|-------------|----------|------|-----|---------|----------------|
| ID          | int(11)  | NO   | PRI | NULL    | auto_increment |
| Name        | char(35) | NO   |     |         |                |
| CountryCode | char(3)  | NO   | MUL |         |                |
| District    | char(20) | NO   |     |         |                |
| Population  | int(11)  | NO   |     | 0       |                |

5 rows in set (0.01 sec)

```
mysql>
```

나머지 2개의 테이블 구조도 확인해 보자



## 2. MySQL – 콘솔에서 사용하기

- city 테이블의 내용 보기 (앞쪽 10개만)

(필요어) 조회 명령어 모든 column 앞에서 10개만

```
mysql> select * from city limit 10;
```

| ID | Name           | CountryCode | District      | Population |
|----|----------------|-------------|---------------|------------|
| 1  | Kabul          | AFG         | Kabul         | 1780000    |
| 2  | Qandahar       | AFG         | Qandahar      | 237500     |
| 3  | Herat          | AFG         | Herat         | 186800     |
| 4  | Mazar-e-Sharif | AFG         | Balkh         | 127800     |
| 5  | Amsterdam      | NLD         | Noord-Holland | 731200     |
| 6  | Rotterdam      | NLD         | Zuid-Holland  | 593321     |
| 7  | Haag           | NLD         | Zuid-Holland  | 440900     |
| 8  | Utrecht        | NLD         | Utrecht       | 234323     |
| 9  | Eindhoven      | NLD         | Noord-Brabant | 201843     |
| 10 | Tilburg        | NLD         | Noord-Brabant | 193238     |

10 rows in set (0.00 sec)

```
mysql>
```

## 2. MySQL – 콘솔에서 사용하기

- Seoul 에 대한 정보를 찾아보기

mysql> select \* from city where Name='Seoul' ;

| ID   | Name  | CountryCode | District | Population |
|------|-------|-------------|----------|------------|
| 2331 | Seoul | KOR         | Seoul    | 9981619    |

1 row in set (0.00 sec)

mysql>

경험은 대수론자  
조건 표시 / 구분  
같은 대수론자 구분

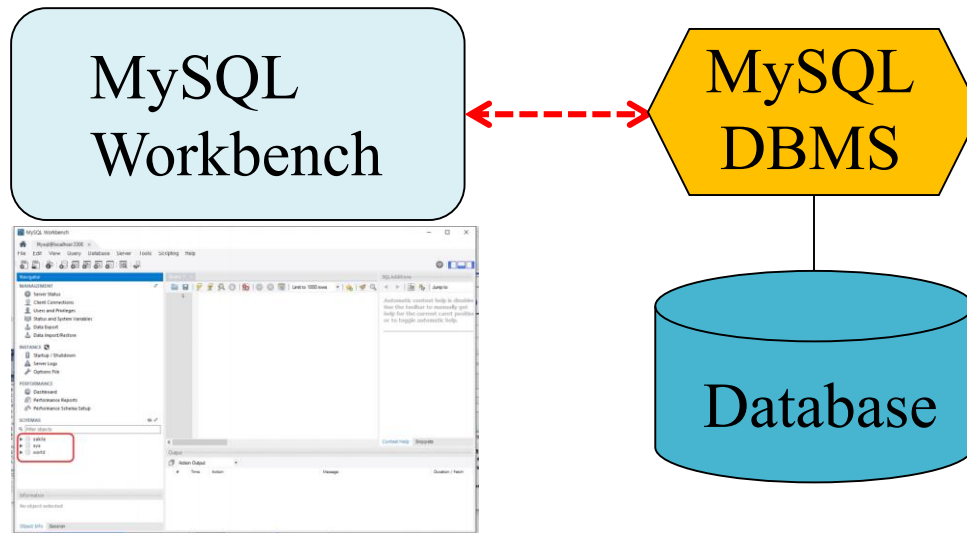
## 2. MySQL – 콘솔에서 사용하기

- mysql 종료하기

```
mysql> quit
```

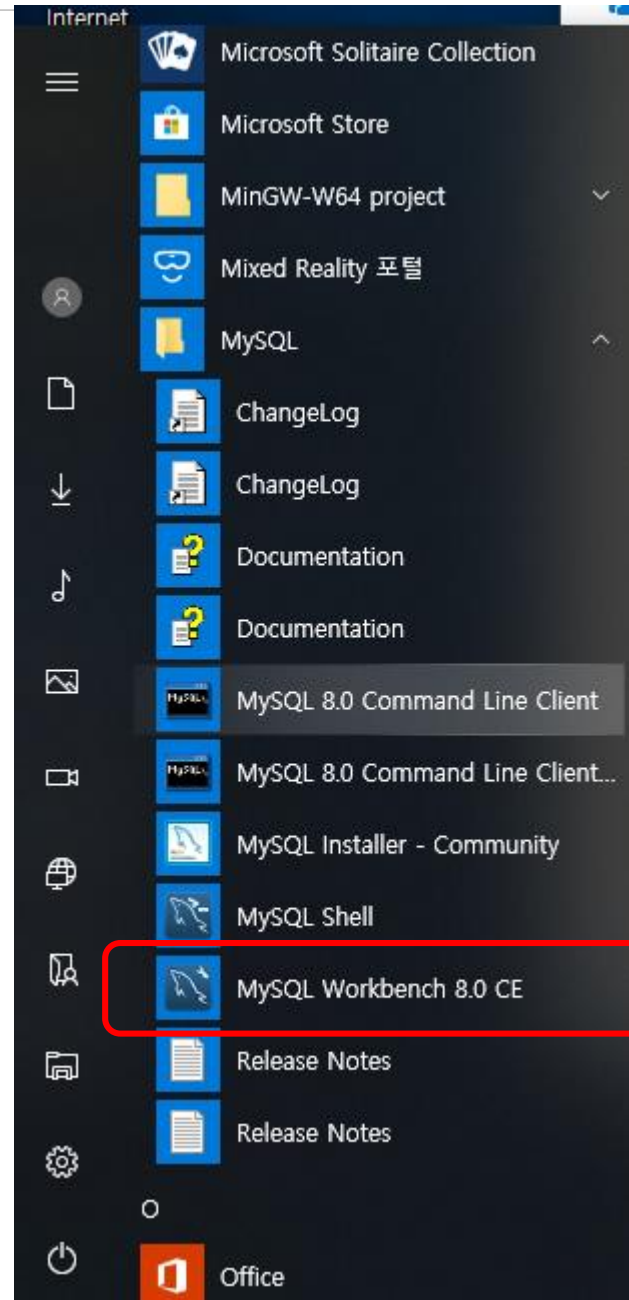
### 3. MySQL Workbench

- mysql 작업을 편하게 할 수 있도록 도와주는 보조 SW
- Console 에서 할 수 있는 모든 작업을 Window 환경에서 진행할 수 있다.



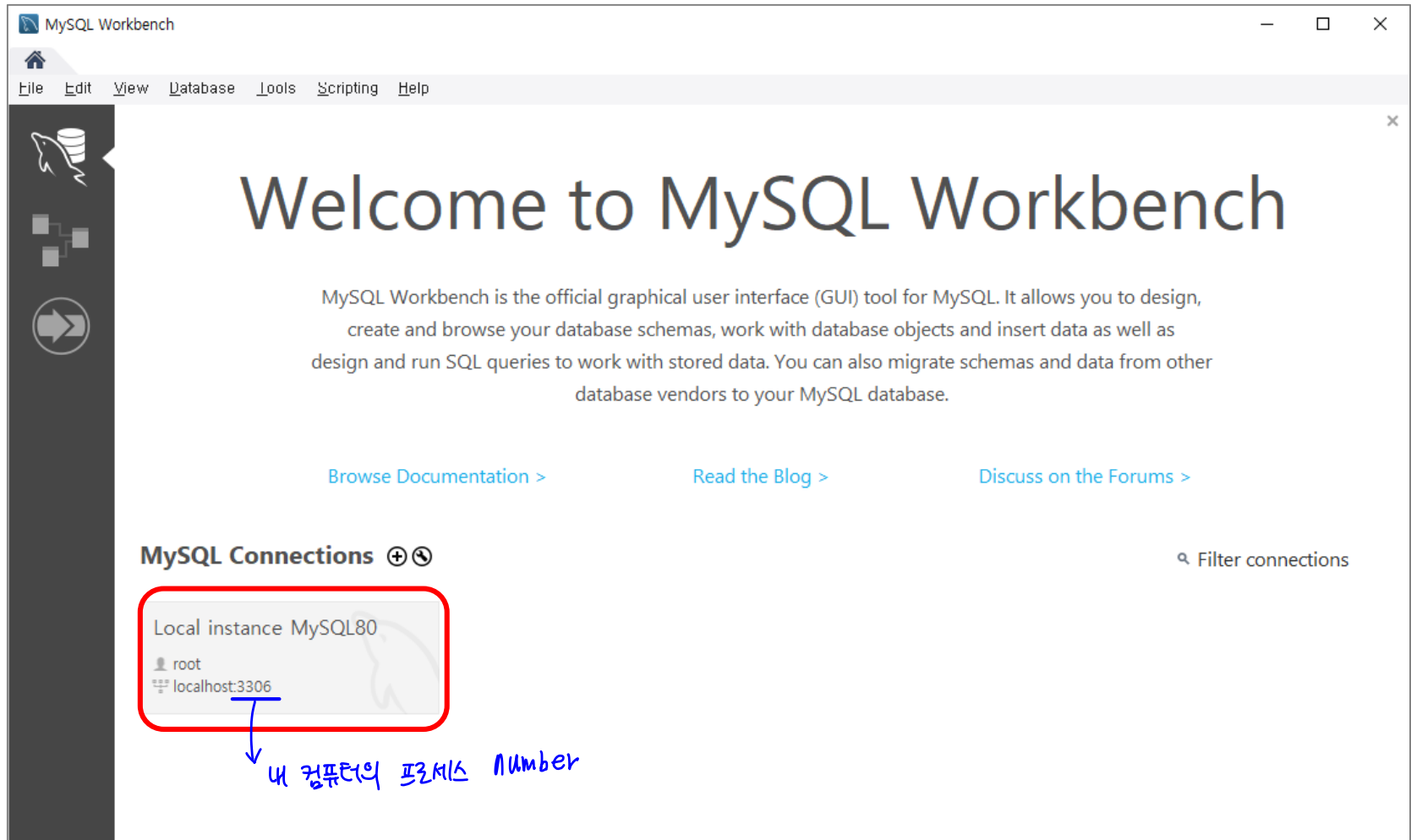
# 3. MySQL Workbench

- 시작



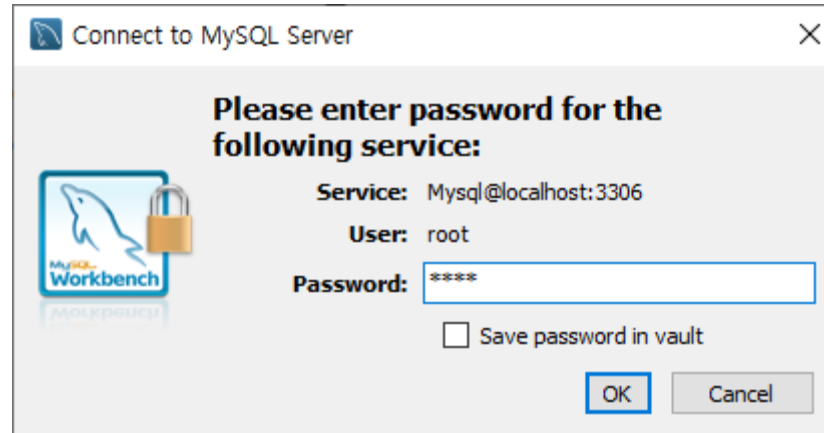
# 3. MySQL Workbench

- Open connection

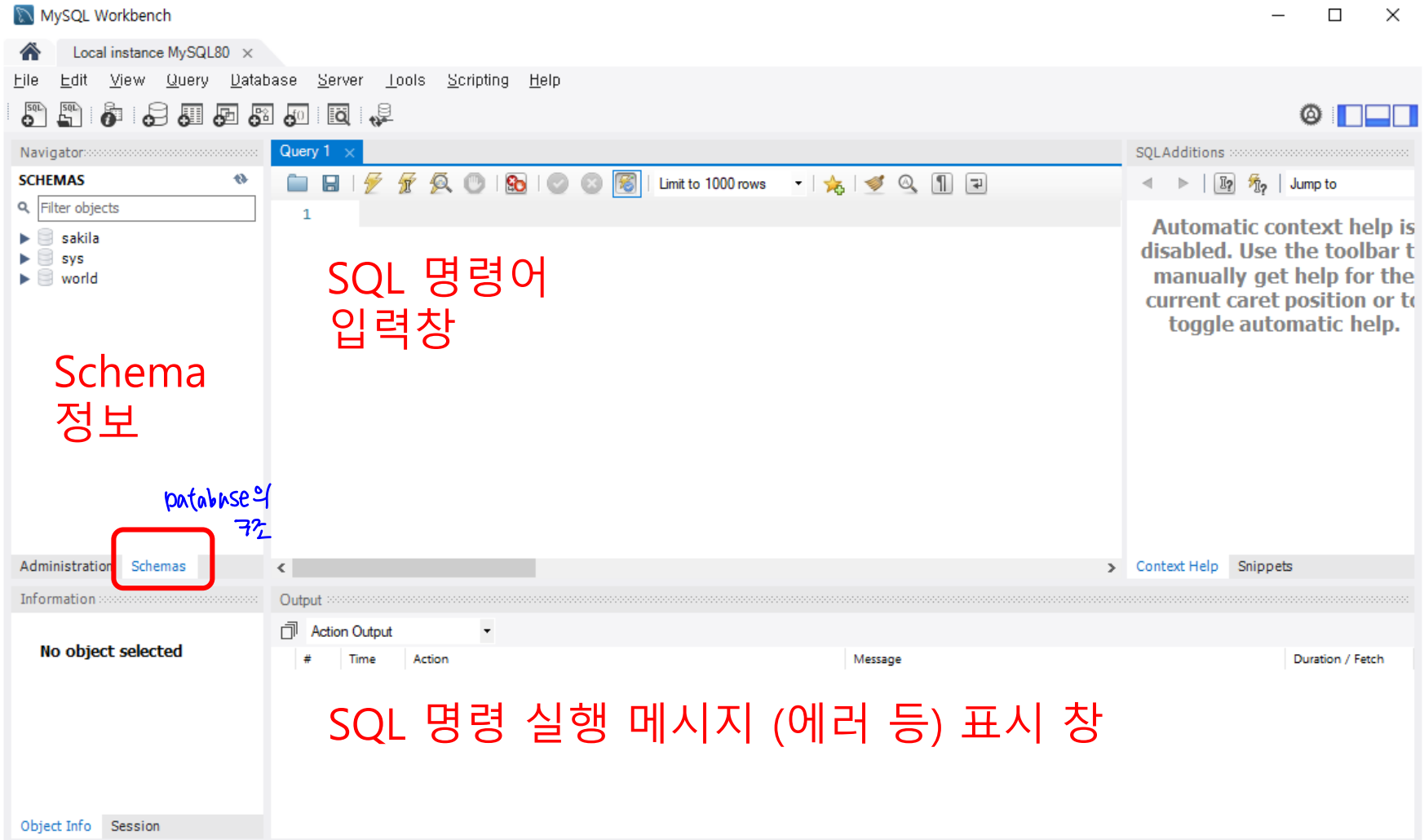


# 3. MySQL Workbench

- 로그인



# 3. MySQL Workbench





### 3. MySQL Workbench

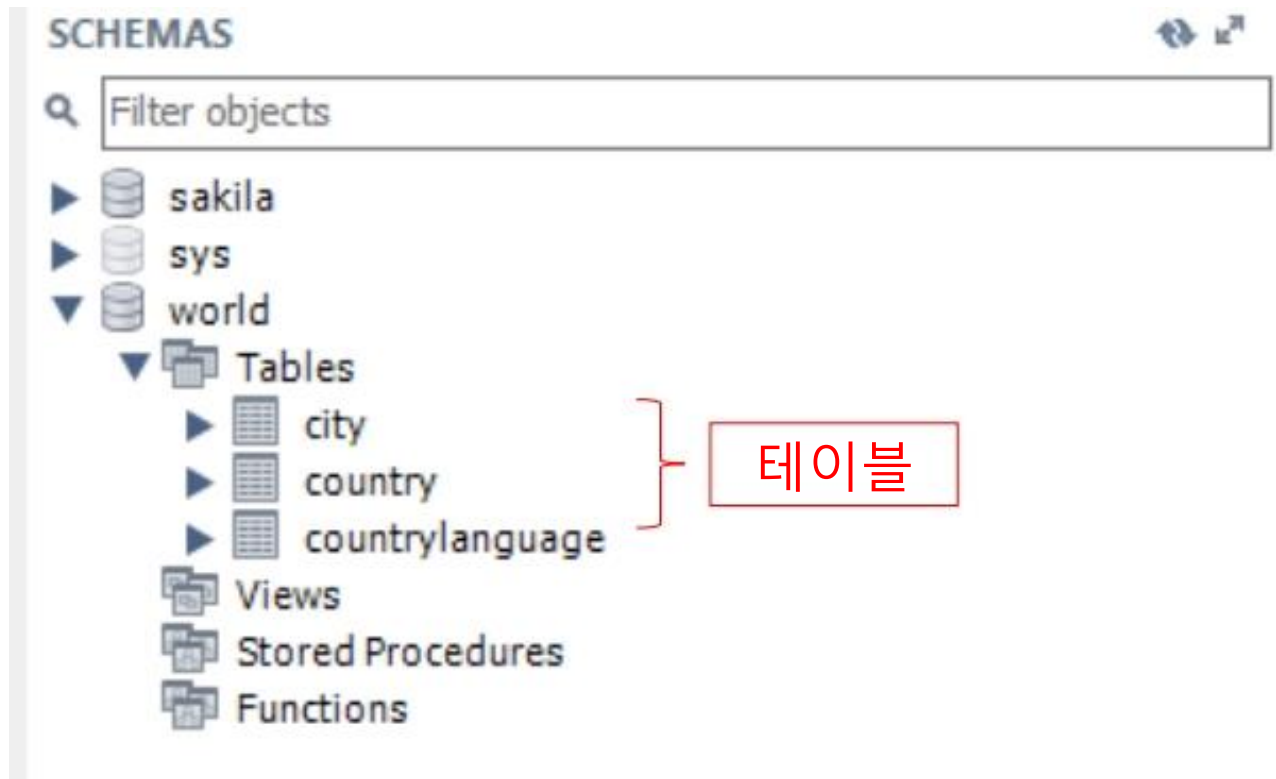
- 초기에 설치되어 있는 데이터베이스



- sakila : 영화 관련 데이터베이스
- sys : 시스템 데이터베이스 (수정하지 않는다)
- world : 도시,국가,언어 데이터베이스

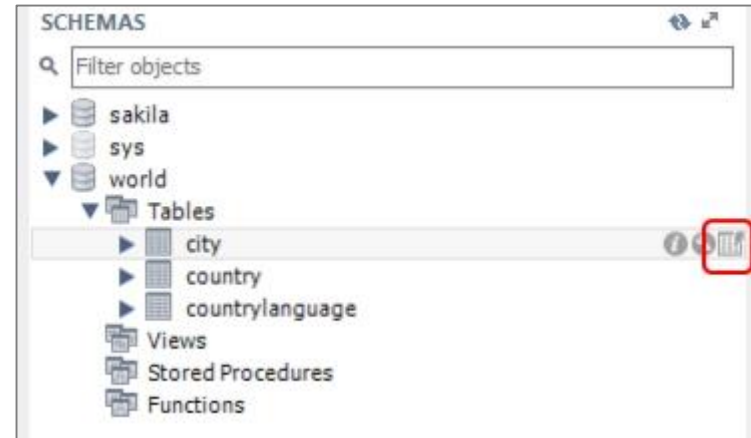
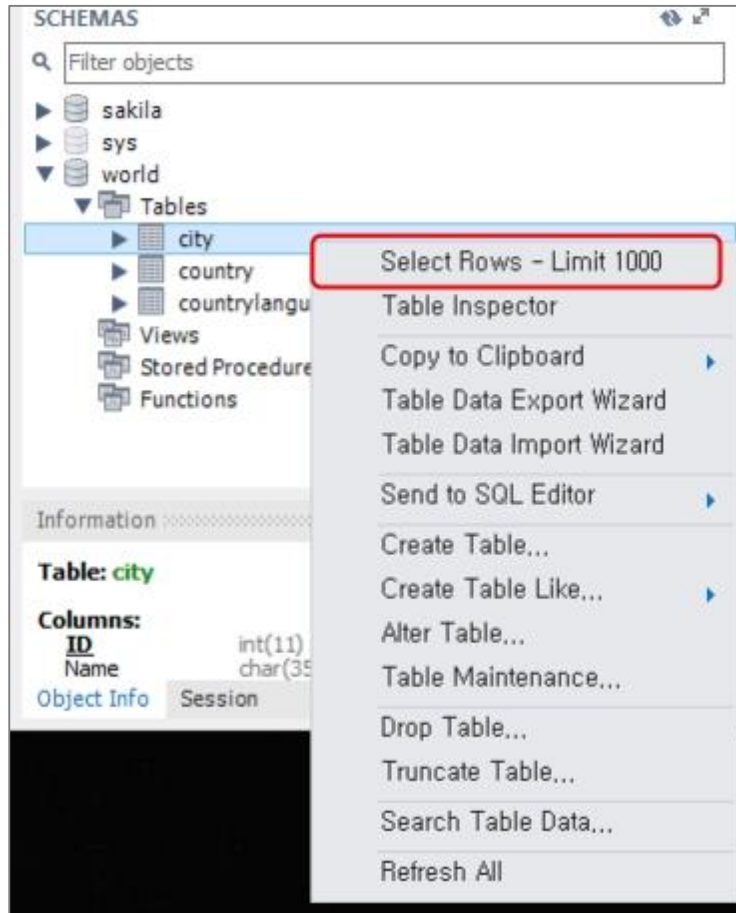
### 3. MySQL Workbench

- World 데이터베이스 살펴보기



### 3. MySQL Workbench

- World 데이터베이스 살펴보기
  - city 테이블 내용 보기 (두가지 방법)



# 3. MySQL Workbench

The screenshot displays the MySQL Workbench interface for a local instance of MySQL 8.0. The main query editor shows a query: `SELECT * FROM world.city;`. The result grid below shows the first five rows of data from the `world.city` table. The table information panel on the left shows the structure of the `city` table. The bottom panel shows the action output, which includes the execution time and the number of rows returned for each query.

**Query 1** city

```
1 • SELECT * FROM world.city;
```

**Result Grid**

| ID | Name           | CountryCode | District      | Population |
|----|----------------|-------------|---------------|------------|
| 1  | Kabul          | AFG         | Kabul         | 1780000    |
| 2  | Qandahar       | AFG         | Qandahar      | 237500     |
| 3  | Herat          | AFG         | Herat         | 186800     |
| 4  | Mazar-e-Sharif | AFG         | Balkh         | 127800     |
| 5  | Amsterdam      | NLD         | Noord-Holland | 731200     |

**Table: city**

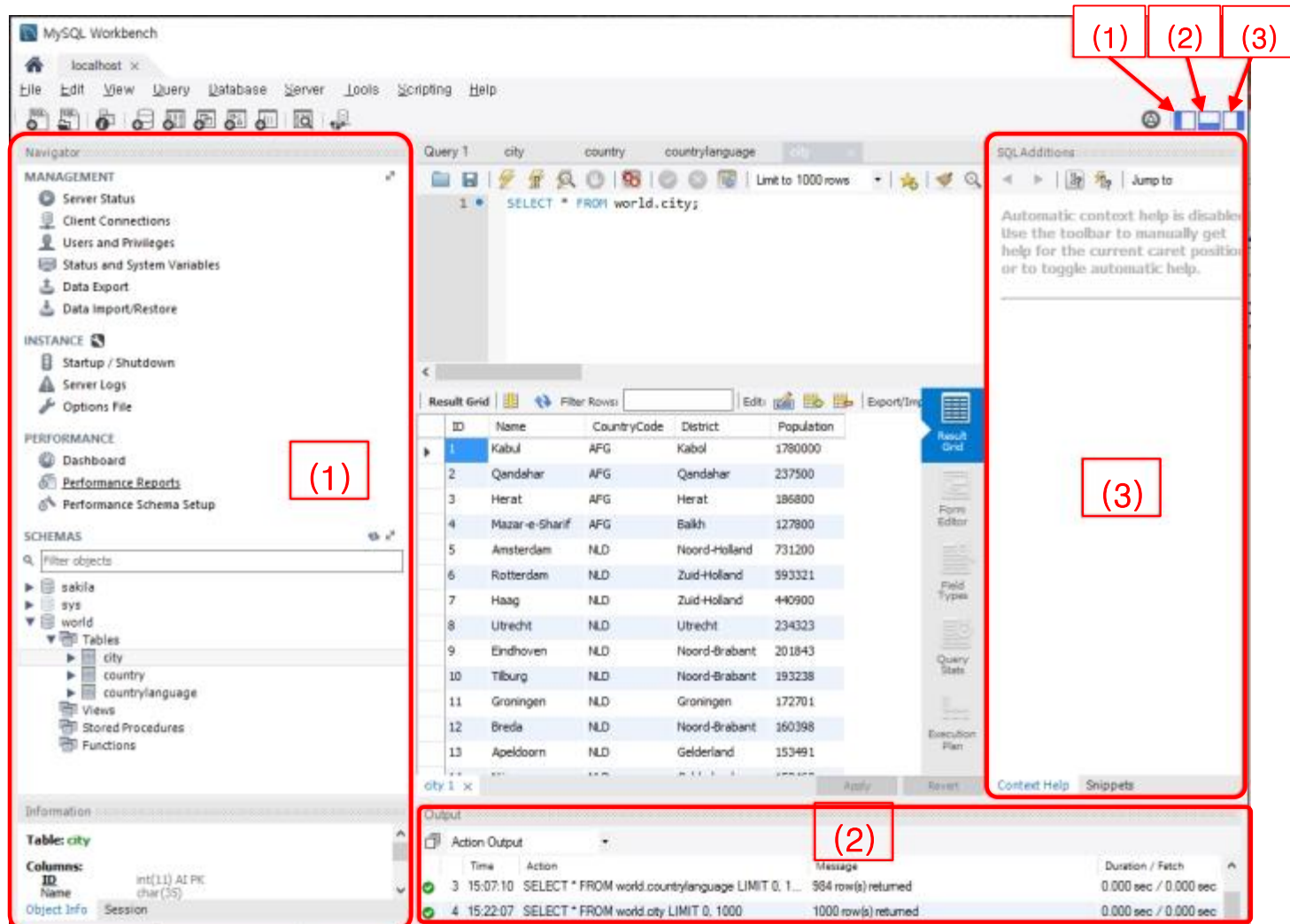
**Columns:**

- ID int(11) AI PK
- Name char(3)
- CountryCode char(3)
- District char(2)
- Population int(11)

**Action Output**

| # | Time     | Action                                 | Message              | Duration / Fetch      |
|---|----------|--|----------------------|-----------------------|
| 1 | 13:11:24 | SELECT * FROM world.city               | 1000 row(s) returned | 0.000 sec / 0.000 sec |
| 2 | 13:12:23 | SELECT * FROM world.city LIMIT 0, 1000 | 1000 row(s) returned | 0.015 sec / 0.000 sec |

# 3. MySQL Workbench



The screenshot displays the MySQL Workbench interface. The left sidebar contains the 'Navigator' pane with sections for 'MANAGEMENT', 'INSTANCE', 'PERFORMANCE', and 'SCHEMAS'. The 'SCHEMAS' section is expanded, showing a tree view of databases (sakila, sys, world) and their contents (Tables, Views, Stored Procedures, Functions). The 'city' table is selected under the 'world' database. The main query editor shows a query: `SELECT * FROM world.city;`. The 'Result Grid' displays the query results in a table format. The bottom right pane shows the 'SQLAdditions' section with a message about automatic context help. The bottom left pane shows the 'Information' section for the selected table. The bottom right pane shows the 'Output' section with a table of query execution results.

(1) points to the left sidebar (Navigator pane).

(2) points to the bottom right pane (Output section).

(3) points to the top right toolbar.

| ID | Name           | CountryCode | District      | Population |
|----|----------------|-------------|---------------|------------|
| 1  | Kabul          | AFG         | Kabul         | 1780000    |
| 2  | Qandahar       | AFG         | Qandahar      | 237500     |
| 3  | Herat          | AFG         | Herat         | 186800     |
| 4  | Mazar-e-Sharif | AFG         | Balkh         | 127800     |
| 5  | Amsterdam      | NLD         | Noord-Holland | 731200     |
| 6  | Rotterdam      | NLD         | Zuid-Holland  | 593321     |
| 7  | Haag           | NLD         | Zuid-Holland  | 440900     |
| 8  | Utrecht        | NLD         | Utrecht       | 234323     |
| 9  | Eindhoven      | NLD         | Noord-Brabant | 201843     |
| 10 | Tilburg        | NLD         | Noord-Brabant | 193238     |
| 11 | Groningen      | NLD         | Groningen     | 172701     |
| 12 | Breda          | NLD         | Noord-Brabant | 160398     |
| 13 | Apeldoorn      | NLD         | Gelderland    | 153491     |

| Time       | Action   | Message              | Duration / Fetch      |
|------------|--|----------------------|-----------------------|
| 3 15:07:10 | SELECT * FROM world.country/language LIMIT 0, 1... | 984 row(s) returned  | 0.000 sec / 0.000 sec |
| 4 15:22:07 | SELECT * FROM world.city LIMIT 0, 1000             | 1000 row(s) returned | 0.000 sec / 0.000 sec |

# 3. MySQL Workbench

- 네덜란드의 도시들만 보기

sql 실행 버튼

The screenshot shows the MySQL Workbench interface. At the top, a red box highlights the lightning bolt icon in the toolbar, which is the SQL execution button. A red arrow points from the text 'sql 실행 버튼' to this icon. Below the toolbar, the SQL editor contains the following query:

```
1 • SELECT * FROM world.city
2 Where CountryCode = 'NLD' ;
```

Below the query editor, the 'Result Grid' tab is active, displaying a table of results:

|   | ID | Name      | CountryCode | District      | Population |
|---|----|-----------|-------------|---------------|------------|
| ▶ | 5  | Amsterdam | NLD         | Noord-Holland | 731200     |
|   | 6  | Rotterdam | NLD         | Zuid-Holland  | 593321     |
|   | 7  | Haag      | NLD         | Zuid-Holland  | 440900     |
|   | 8  | Utrecht   | NLD         | Utrecht       | 234323     |
|   | 9  | Eindhoven | NLD         | Noord-Brabant | 201843     |

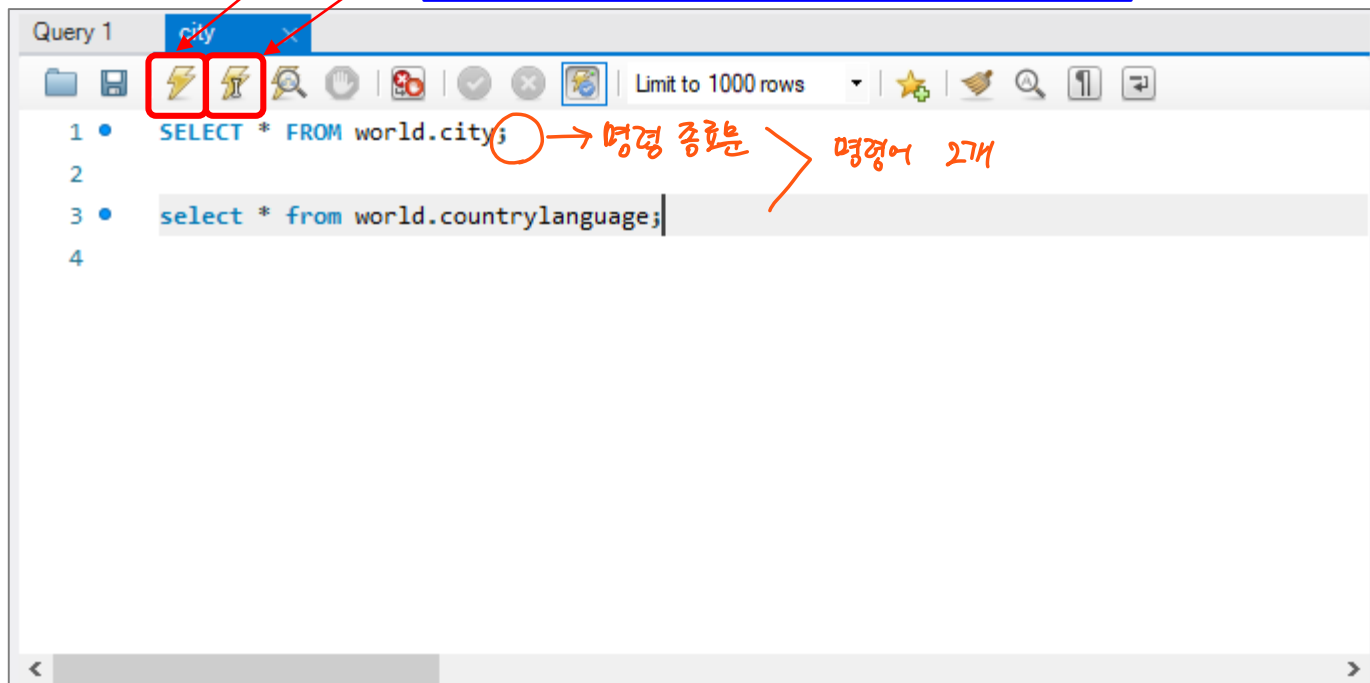
At the bottom, the 'Output' tab is active, showing the 'Action Output' log:

| #   | Time     | Action  | Message              |
|-----|----------|---|----------------------|
| ✓ 1 | 13:11:24 | SELECT * FROM world.city LIMIT 0, 1000                            | 1000 row(s) returned |
| ✓ 2 | 13:12:23 | SELECT * FROM world.city LIMIT 0, 1000                            | 1000 row(s) returned |
| ✓ 3 | 13:18:54 | SELECT * FROM world.city Where CountryCode = 'NLD' LIMIT 0, 10... | 28 row(s) returned   |

### 3. MySQL Workbench

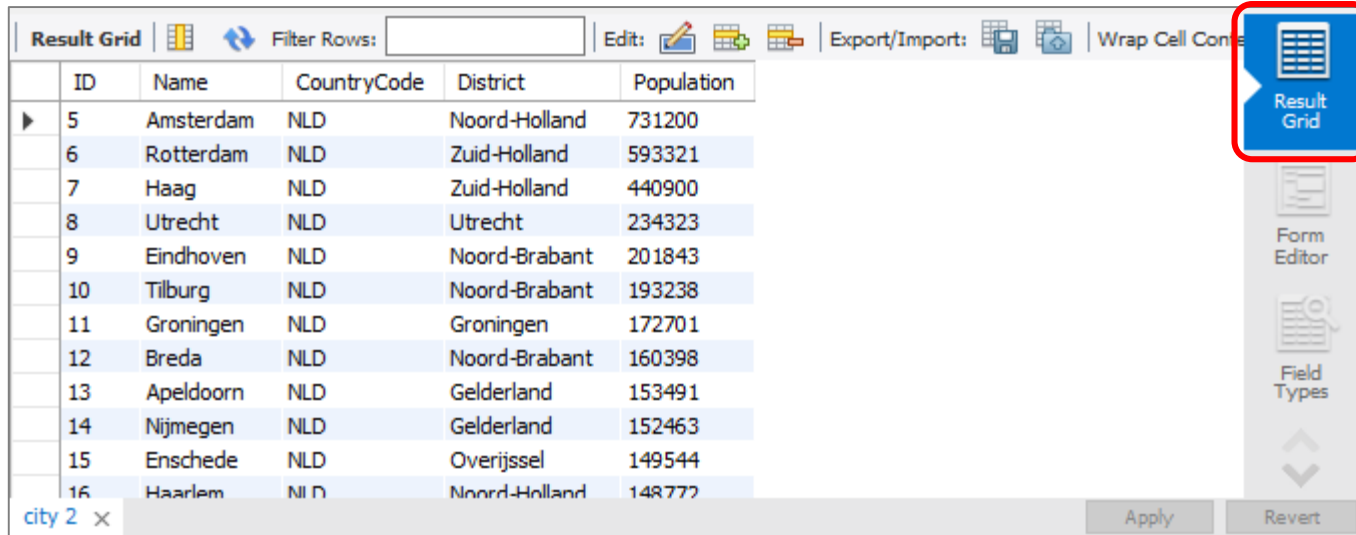
- SQL 명령어가 여러 개인 경우 선택하여 실행하기

명령창에 있는 모든 명령어 실행  
현재 커서 위치의 명령어 실행



### 3. MySQL Workbench

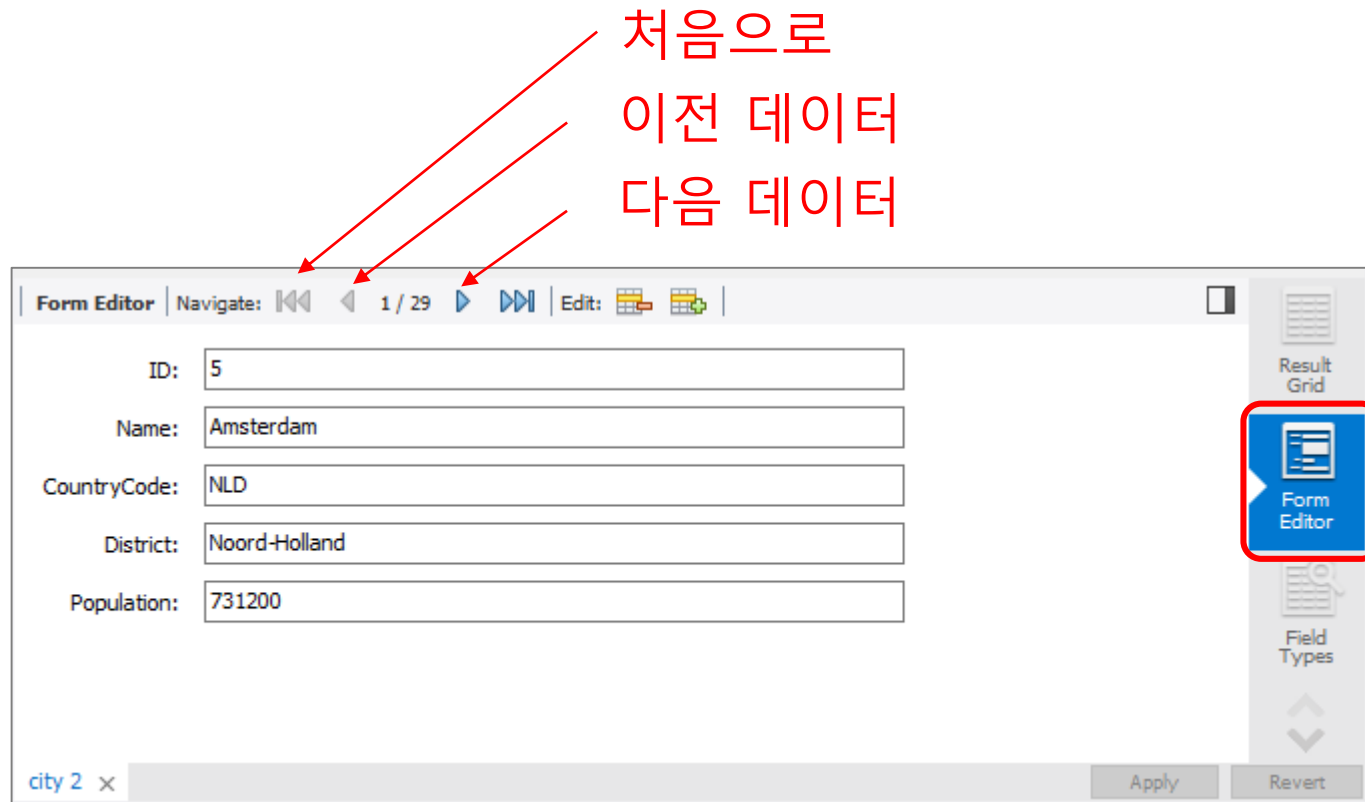
- 질의 결과를 격자 형태로 보기 (default)





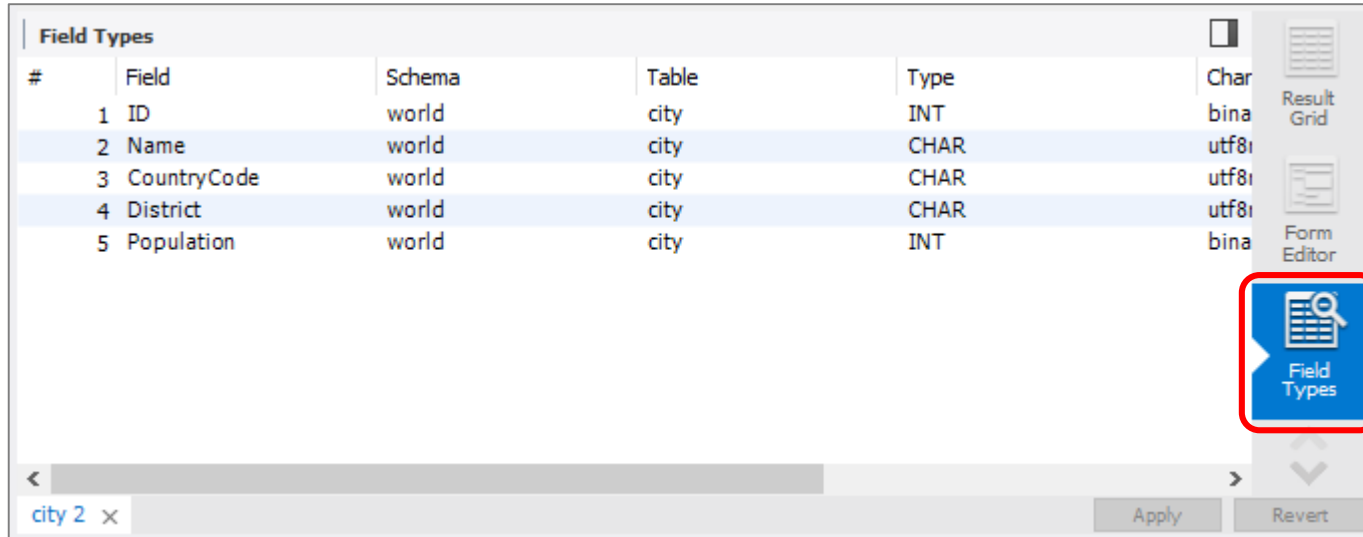
### 3. MySQL Workbench

- 질의 결과를 폼 형태로 보기



# 3. MySQL Workbench

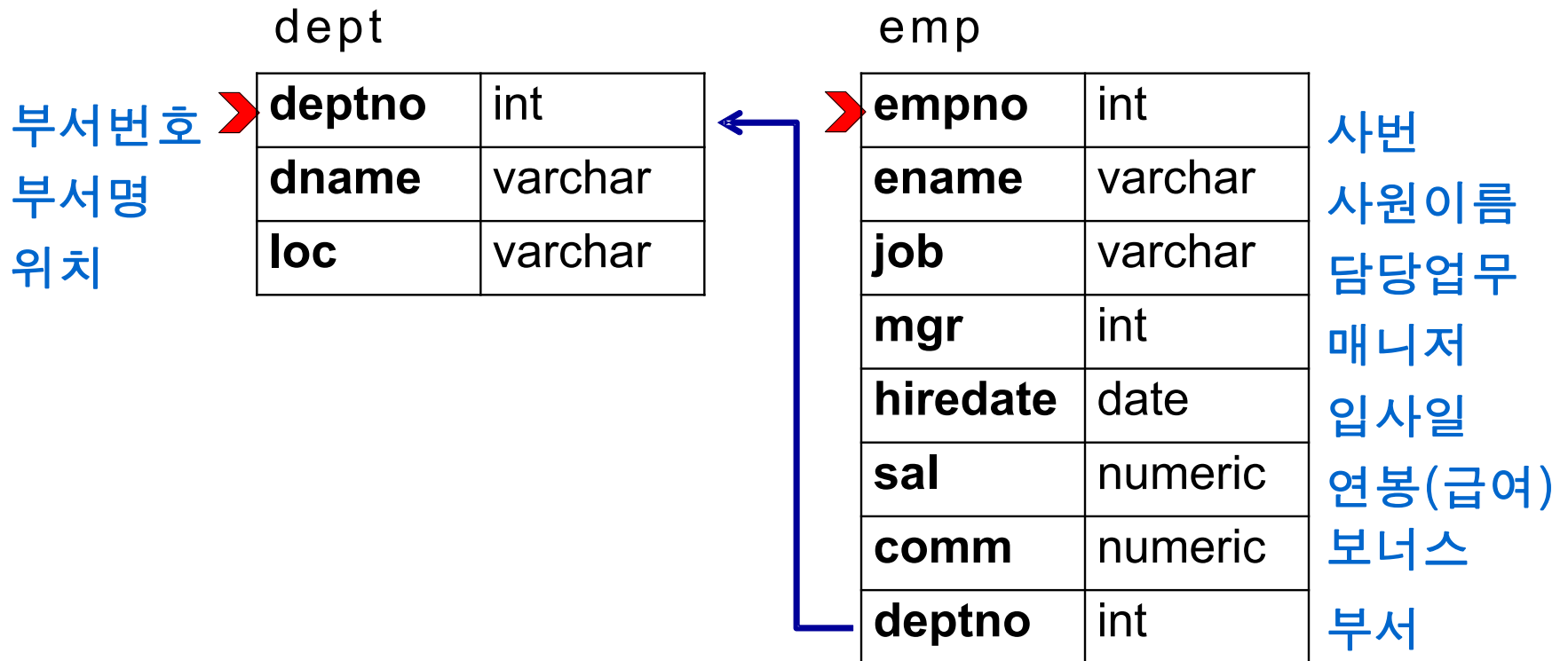
- 테이블 구조 정보 보기



Field = Column

## 4. 실습용 데이터베이스 생성하기

- 부서-사원 데이터베이스 (mydb)



## 4. 실습용 데이터베이스 생성하기

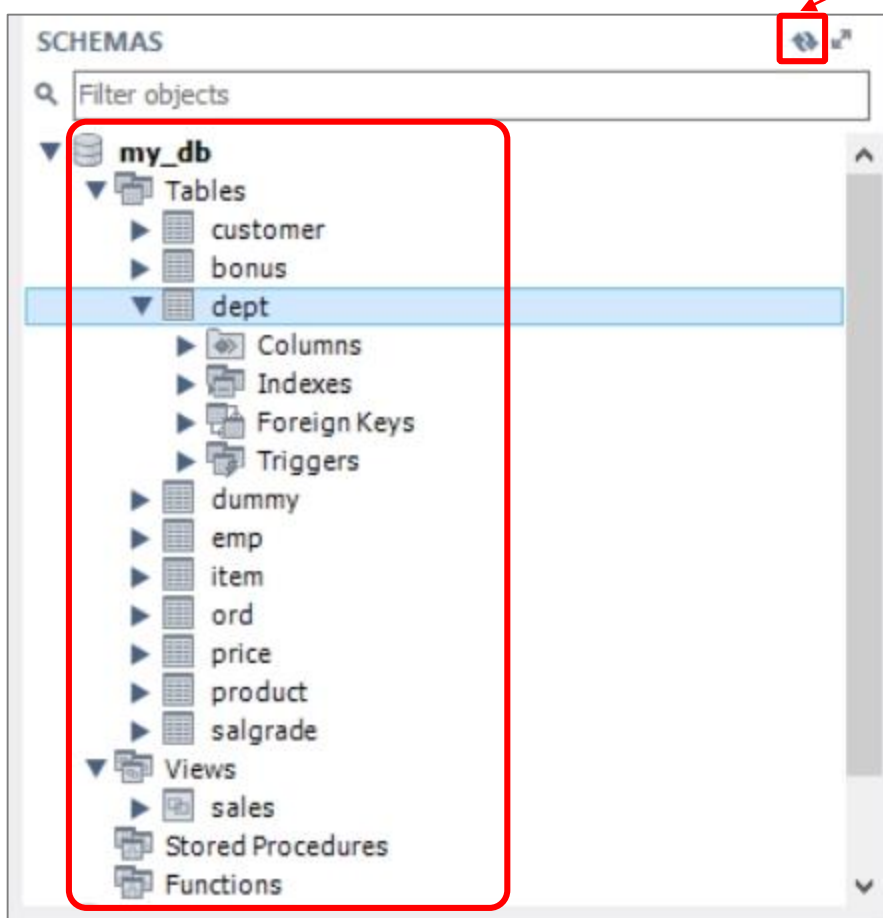
- create\_table.sql 의 내용을 SQL 명령어 실행창에 붙여 넣기  
한 후 실행

```
1 CREATE DATABASE my_db;
2 USE my_db;
3
4 CREATE TABLE DEPT (
5     DEPTNO          DECIMAL(2) NOT NULL,
6     DNAME           VARCHAR(14),
7     LOC             VARCHAR(13),
8     CONSTRAINT DEPT_PRIMARY_KEY PRIMARY KEY (DEPTNO));
9
10 INSERT INTO DEPT VALUES (10, 'ACCOUNTING', 'NEW YORK');
11 INSERT INTO DEPT VALUES (20, 'RESEARCH', 'DALLAS');
12 INSERT INTO DEPT VALUES (30, 'SALES', 'CHICAGO');
13 INSERT INTO DEPT VALUES (40, 'OPERATIONS', 'BOSTON');
14
15 CREATE TABLE EMP (
16     EMPNO           DECIMAL(4) NOT NULL,
17     ENAME           VARCHAR(10),
18     JOB             VARCHAR(9),
19     MGR             DECIMAL(4),
20     HIREDATE        DATE,
21     SAL             DECIMAL(7,2),
22     COMM            DECIMAL(7,2),
23     DEPTNO          DECIMAL(2) NOT NULL,
24     CONSTRAINT EMP_SELF_KEY FOREIGN KEY (MGR) REFERENCES EMP(EMPNO),
25     CONSTRAINT EMP_FOREIGN_KEY FOREIGN KEY (DEPTNO) REFERENCES DEPT (DEPTNO),
26     CONSTRAINT EMP_PRIMARY_KEY PRIMARY KEY (EMPNO));
27
28 INSERT INTO EMP VALUES (7839, 'KING', 'PRESIDENT', NULL, '81-11-17', 5000, NULL, 10);
29 INSERT INTO EMP VALUES (7698, 'BLAKE', 'MANAGER', 7839, '81-5-1', 2850, NULL, 30);
```

## 4. 실습용 데이터베이스 생성하기

- my\_db 가 정상적으로 생성되었는지 확인

새로고침



The Schemas window displays the following structure for my\_db:

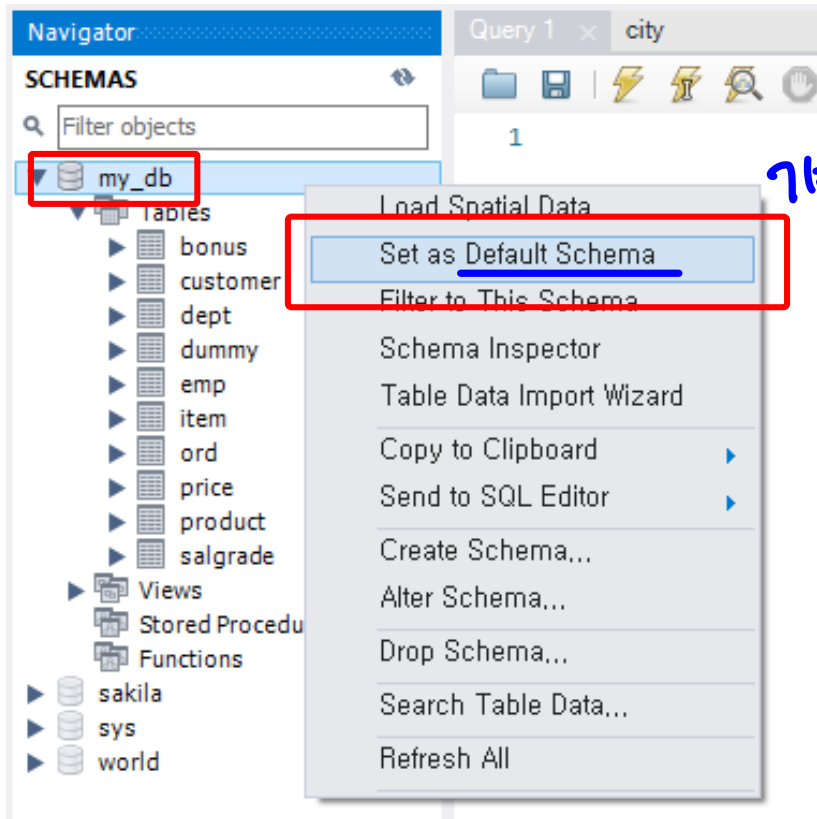
- my\_db
  - Tables
    - customer
    - bonus
    - dept
    - Columns
    - Indexes
    - Foreign Keys
    - Triggers
    - dummy
    - emp
    - item
    - ord
    - price
    - product
    - salgrade
  - Views
    - sales
  - Stored Procedures
  - Functions

Result Grid

|   | DEPTNO | DNAME      | LOC      |
|---|--------|------------|----------|
| ▶ | 10     | ACCOUNTING | NEW YORK |
|   | 20     | RESEARCH   | DALLAS   |
|   | 30     | SALES      | CHICAGO  |
|   | 40     | OPERATIONS | BOSTON   |
| * | NULL   | NULL       | NULL     |

## 4. 실습용 데이터베이스 생성하기

- 작업 데이터베이스 설정하기

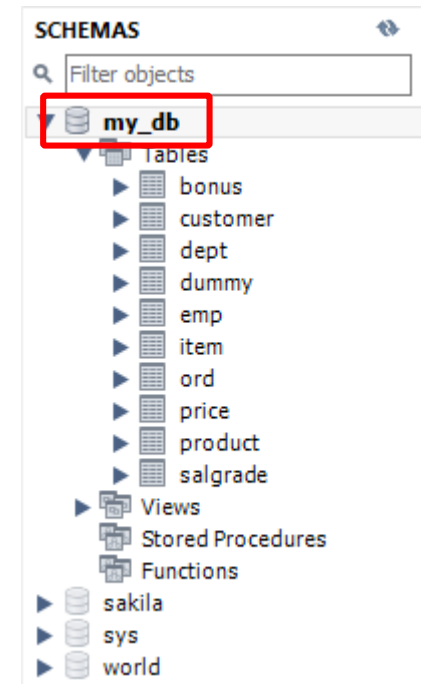


설정 전

```
select * from my_db.emp ;
```

설정 후

```
select * from emp ;
```



# Note

- SQL에서는 명령어, 테이블이름, 컬럼 이름 등에 대해 대소문자를 구별하지 않는다 '데이터 값은 구별'

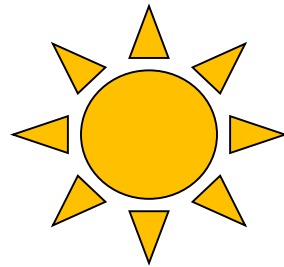
```
SELECT * FROM world.city;
```

```
select * from world.COUNTRYLANGUAGE;
```

- 한 명령어를 여러줄에 걸쳐 작성해도 된다

```
1 • select *  
2   from emp  
3   where empno = 7369 ;  
4
```

- 명령어와 명령어는 ; 으로 구분한다



수고했습니다