Chapter 3

MySQL 기초

오 세 종

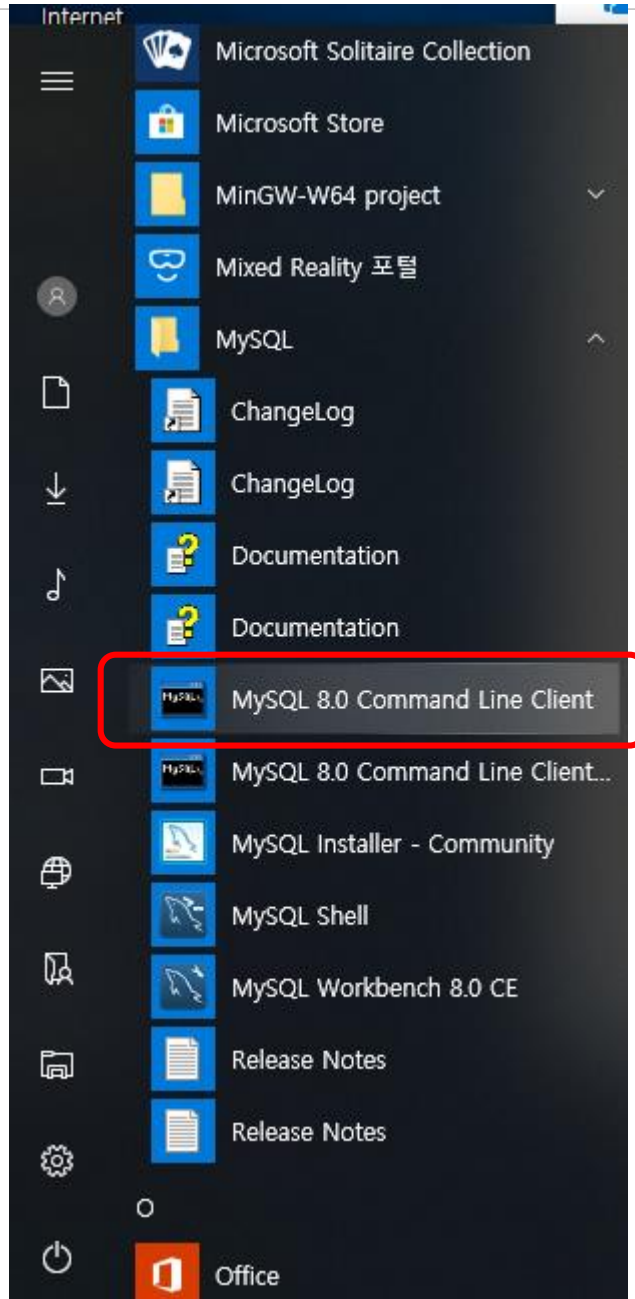
DANKOOK UNIVERSITY

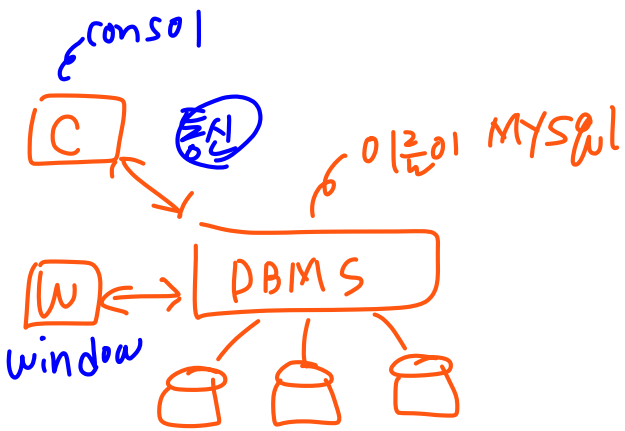
1. MySQL 특징

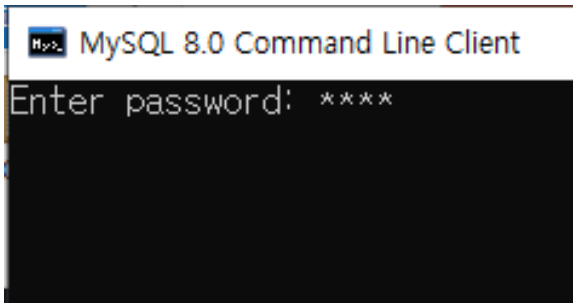
- SQL에 기반을 둔 관계형 DBMS 중 하나
- Oracle, IBM, Infomix 등의 데이터베이스는 고가이지만, MySQL 데이터베이스는 무료
- Oracle 에 인수된 이후에 Maria DB 등장
- 리눅스, 유닉스, 윈도우 등 거의 모든 운영체제에서 사용가능
- 처리 속도가 상당히 빠르고 대용량에 데이터도 처리 용이
- 설치 방법이 쉽고 초보자도 익히기 쉬움
- 보안성이 우수

2. MySQL – 콘솔에서 사용하기

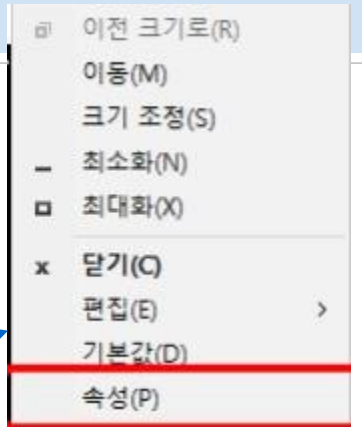
- 콘솔 열기



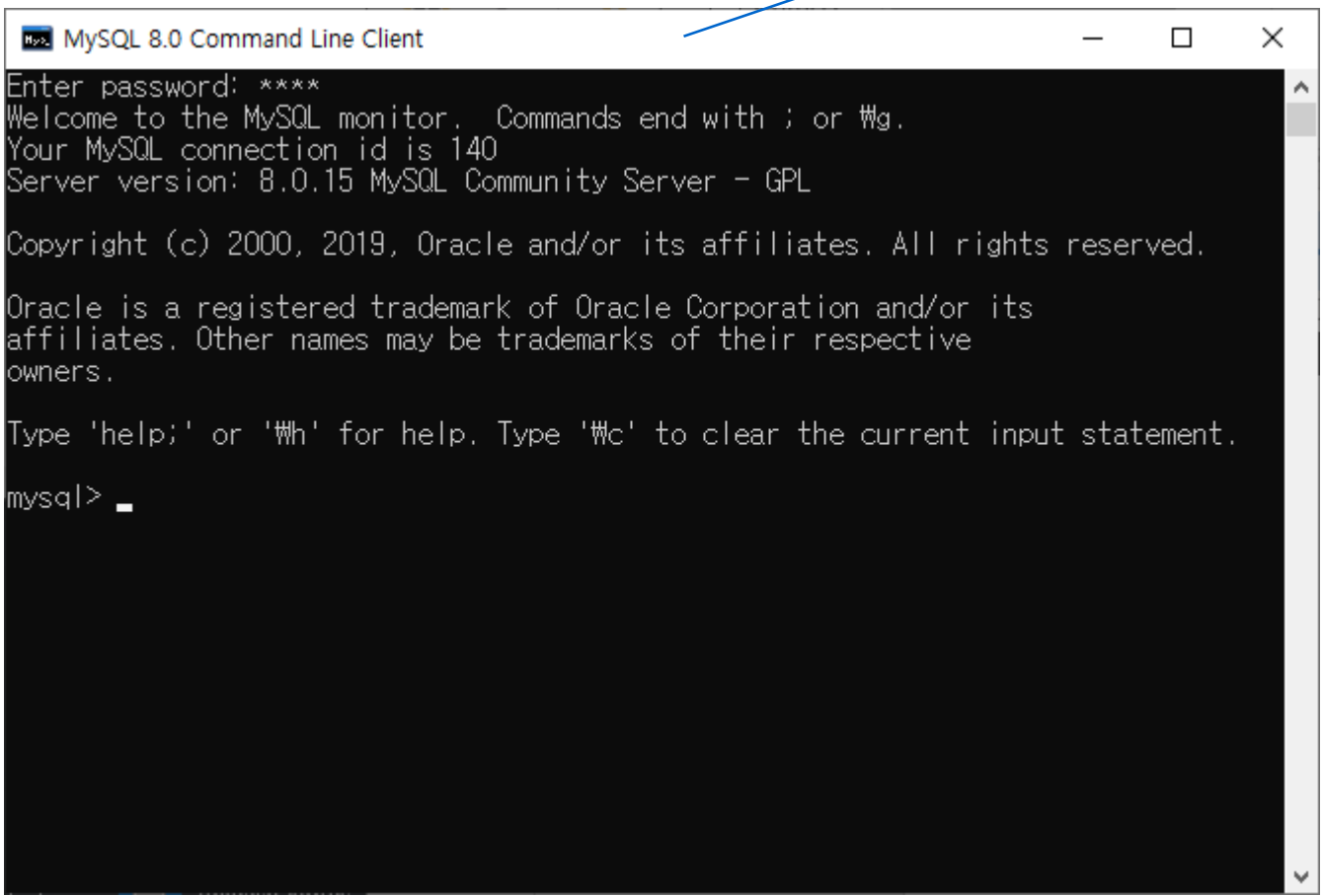




2. MySQL – 콘솔에서 사용하기

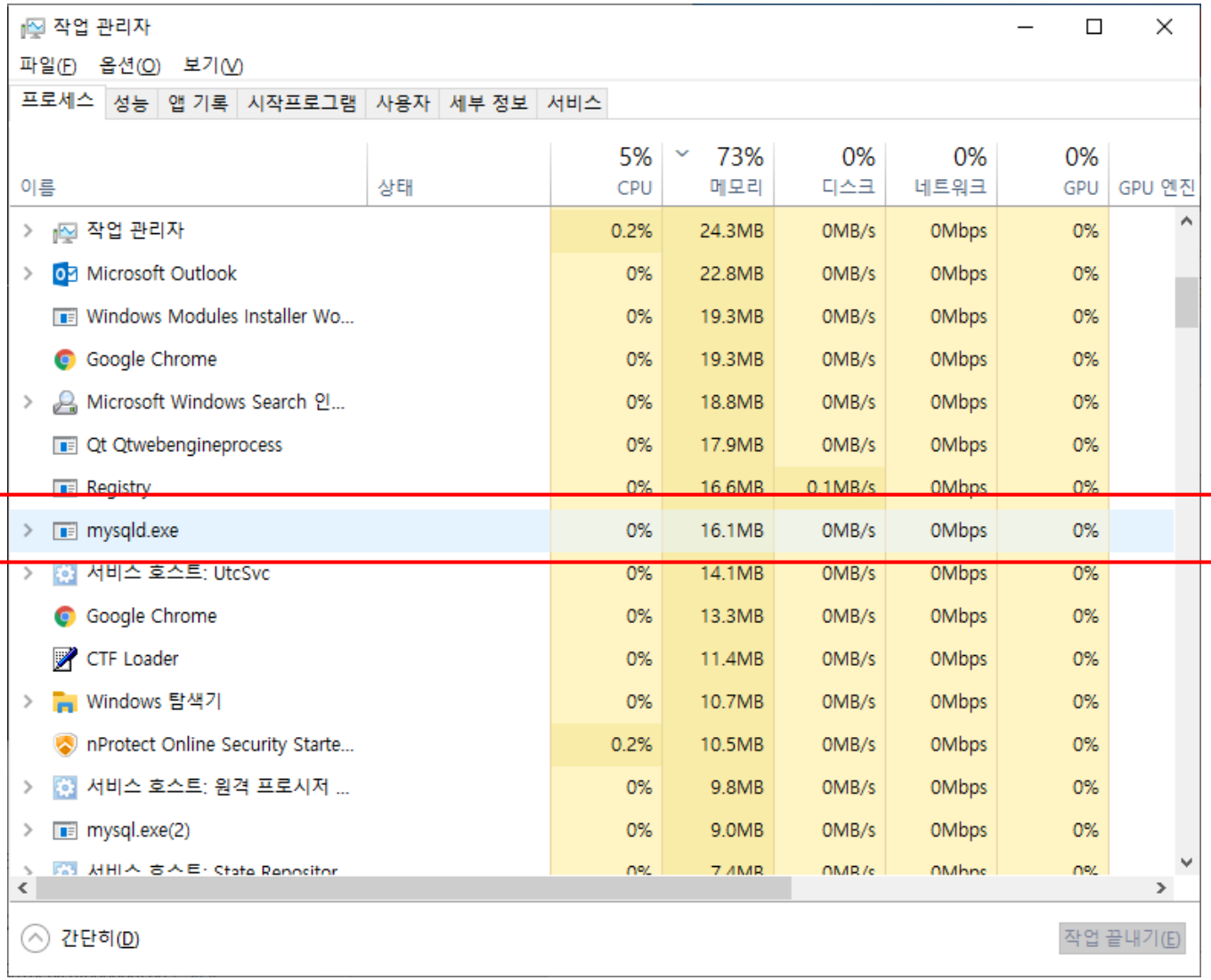


마우스 오른쪽 버튼 클릭
(글씨크기, 바탕색등 조절)



2. MySQL – 콘솔에서 사용하기

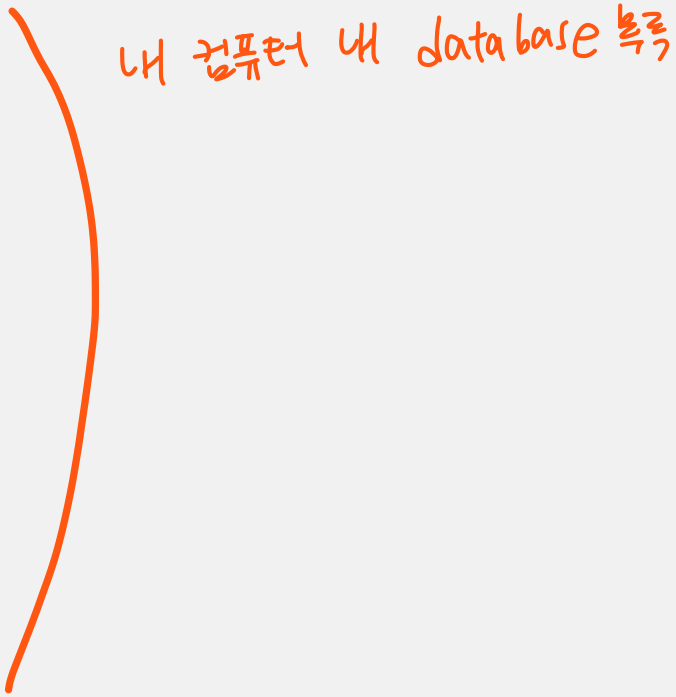
- Mysql 서버가 실행중임



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>
```



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 DB로 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)

(질의어) 조회 명령어 / 모든 column / 앞에서 10개만

```
mysql> select * from city limit 10;
+----+----------------+-------------+---------------+------------+
| ID | Name           | CountryCode | District      | Population |
+----+----------------+-------------+---------------+------------+
|  1 | Kabul          | AFG         | Kabol         |    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

조건 표시 / 명령어는 대소문자 구분X / 값은 대소문자 구분

```
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>
```

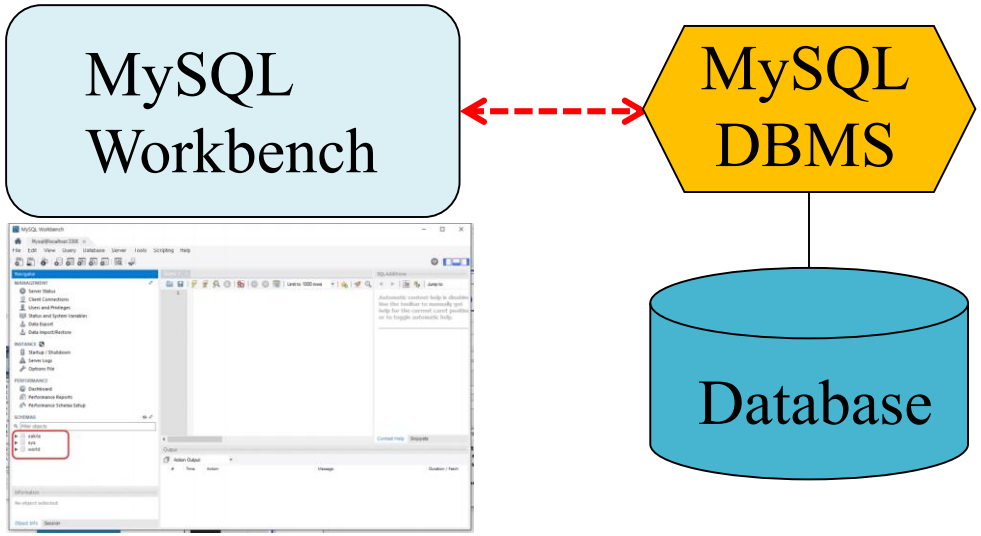
2. MySQL – 콘솔에서 사용하기

- mysql 종료하기

```
mysql> quit
```

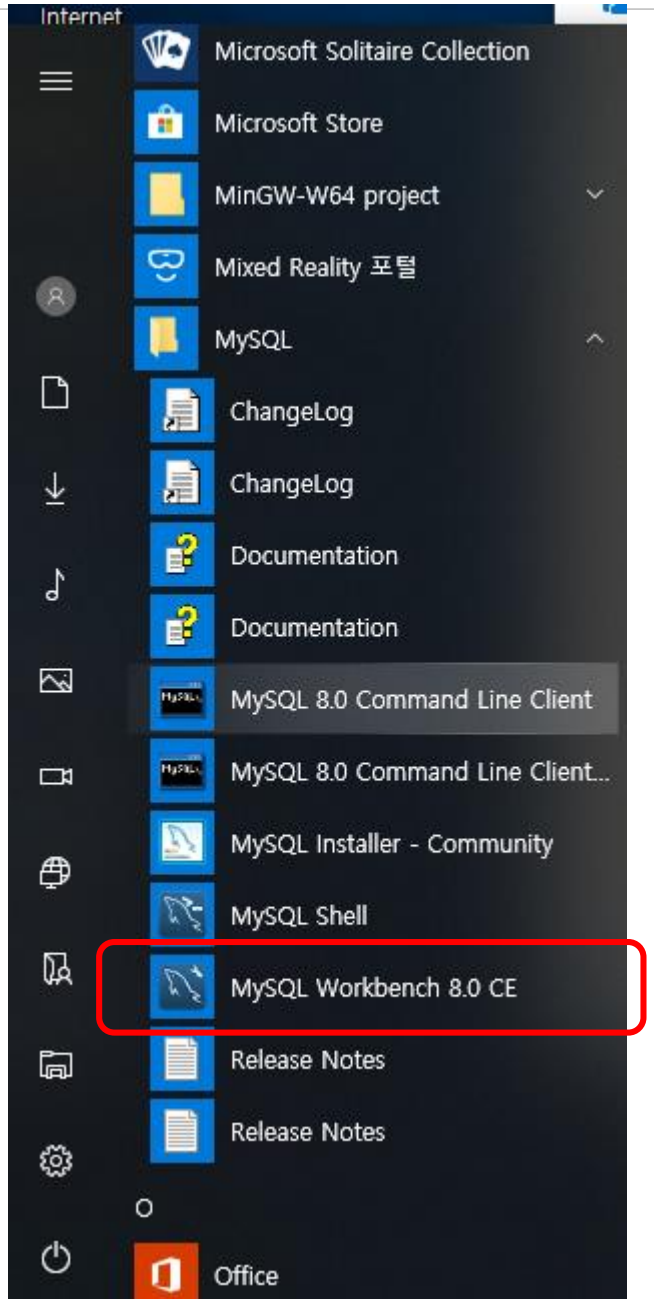
3. MySQL Workbench

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



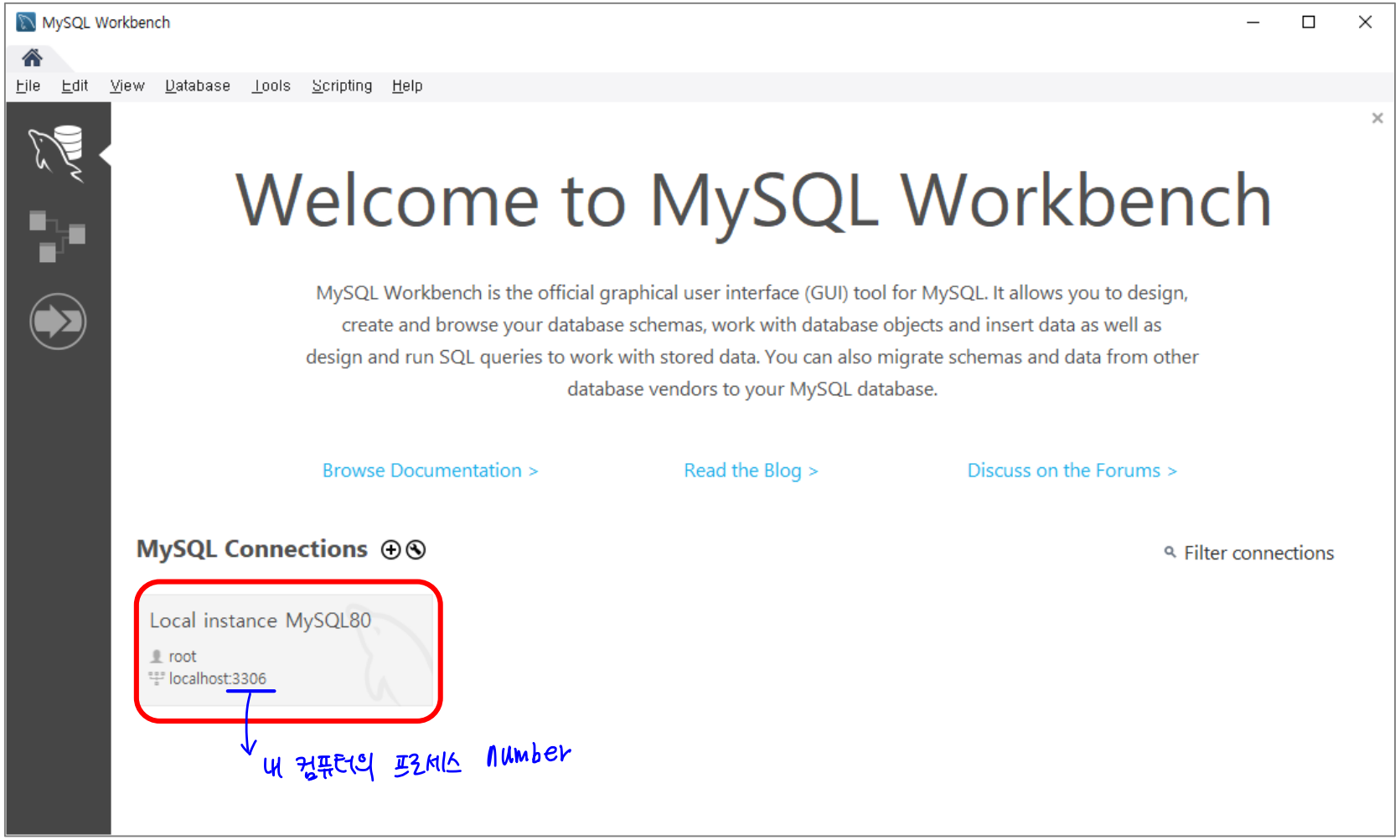
3. MySQL Workbench

- 시작



3. MySQL Workbench

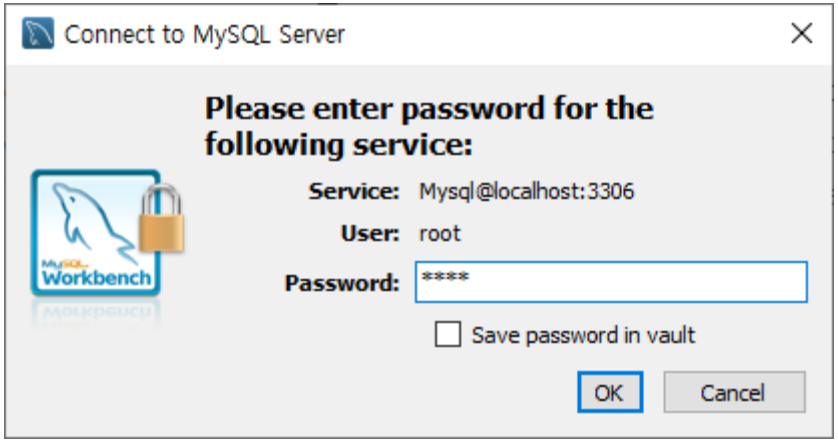
- Open connection



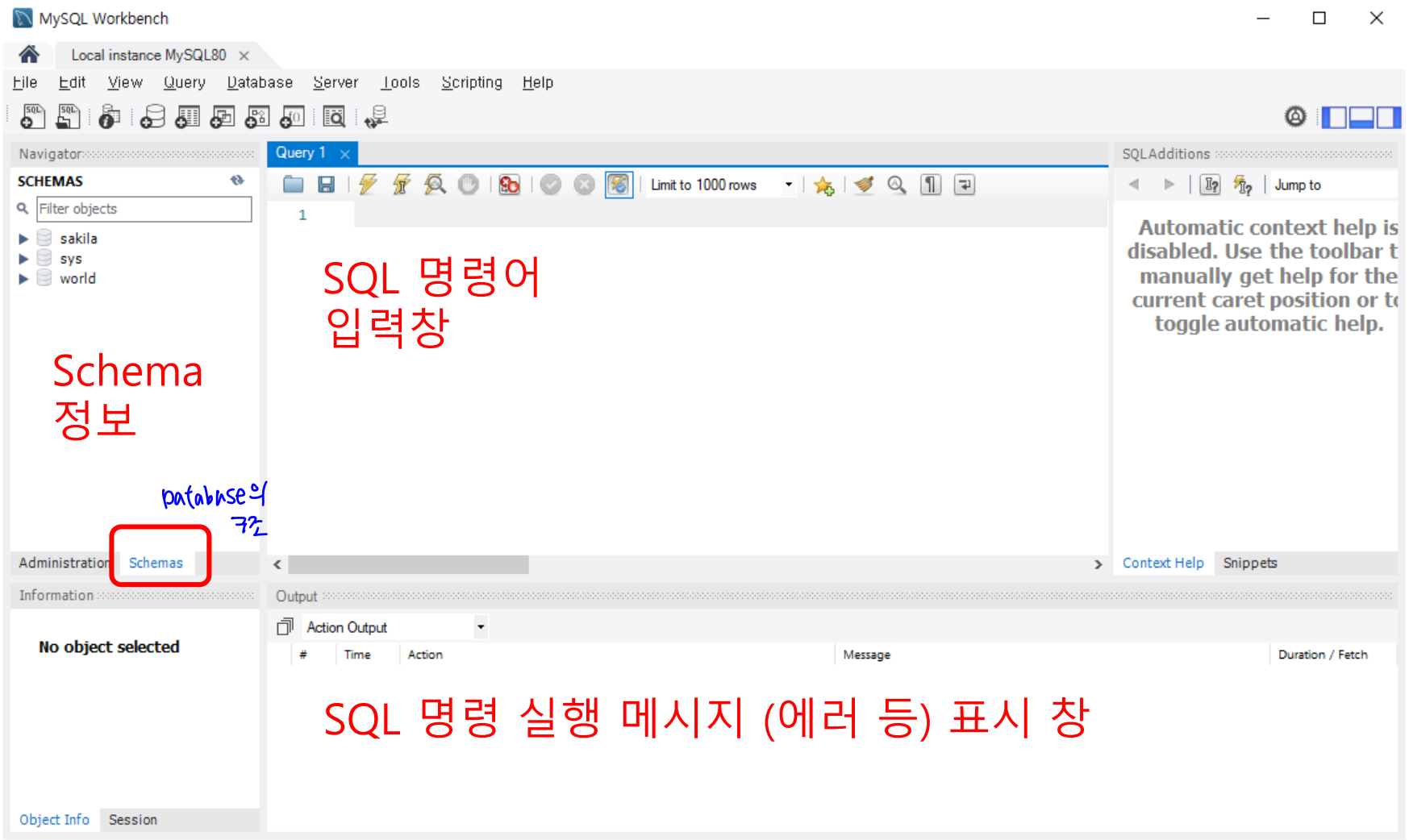
Connection 을 계정별로 여러 개 만들 수 있다

3. MySQL Workbench

- 로그인



3. MySQL Workbench



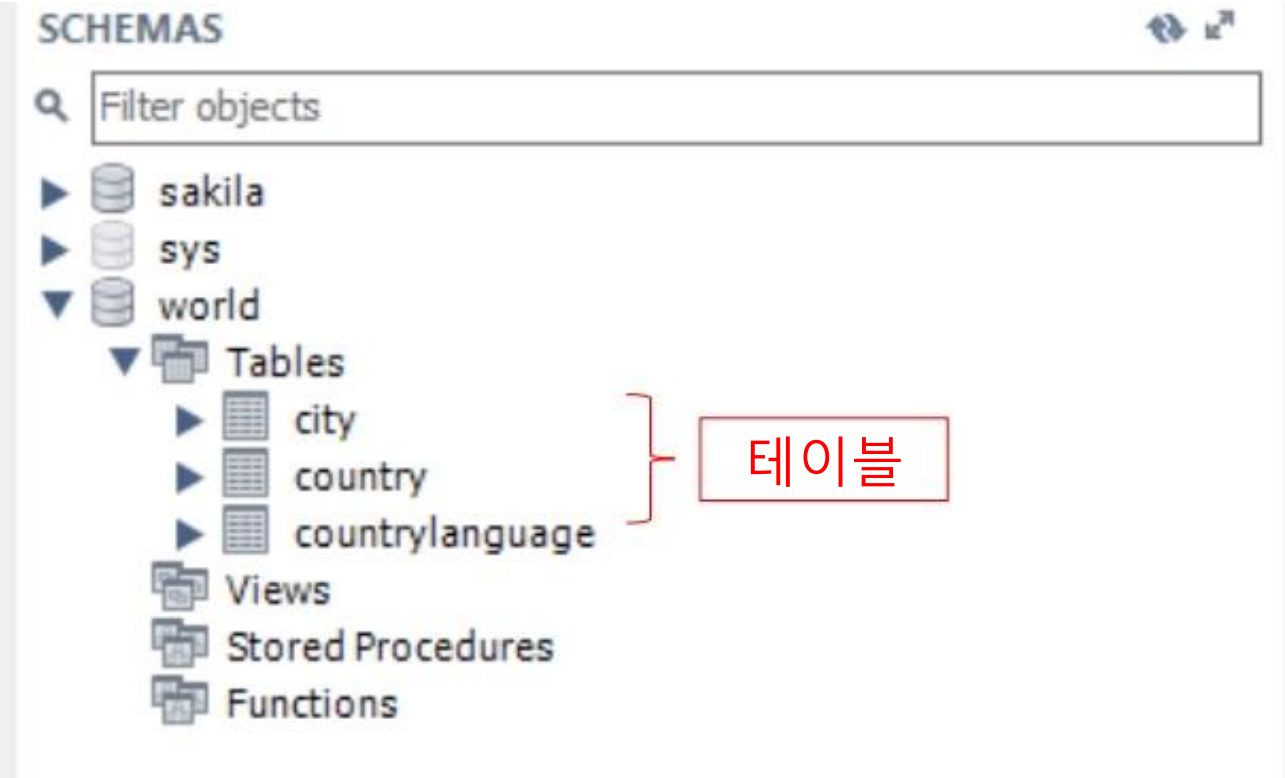
3. MySQL Workbench

- 초기에 설치되어 있는 데이터베이스

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

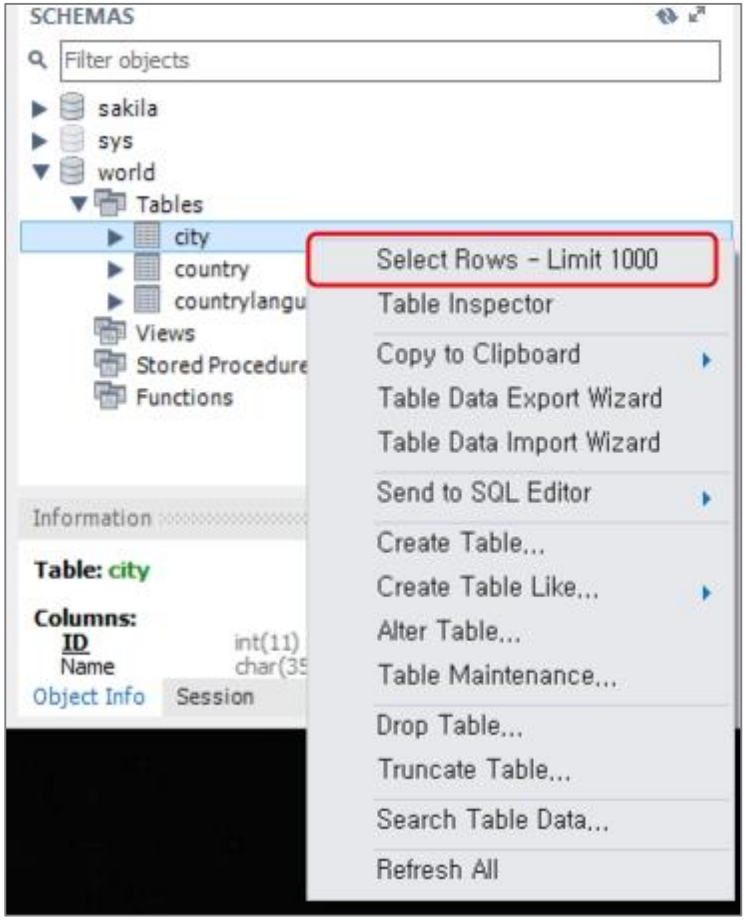
3. MySQL Workbench

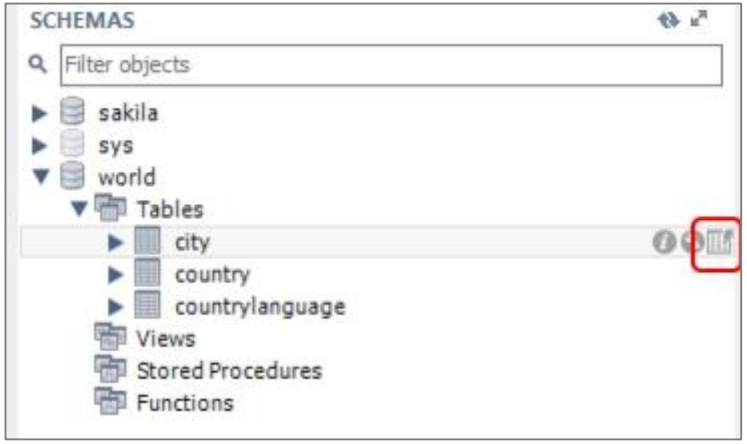
- World 데이터베이스 살펴보기



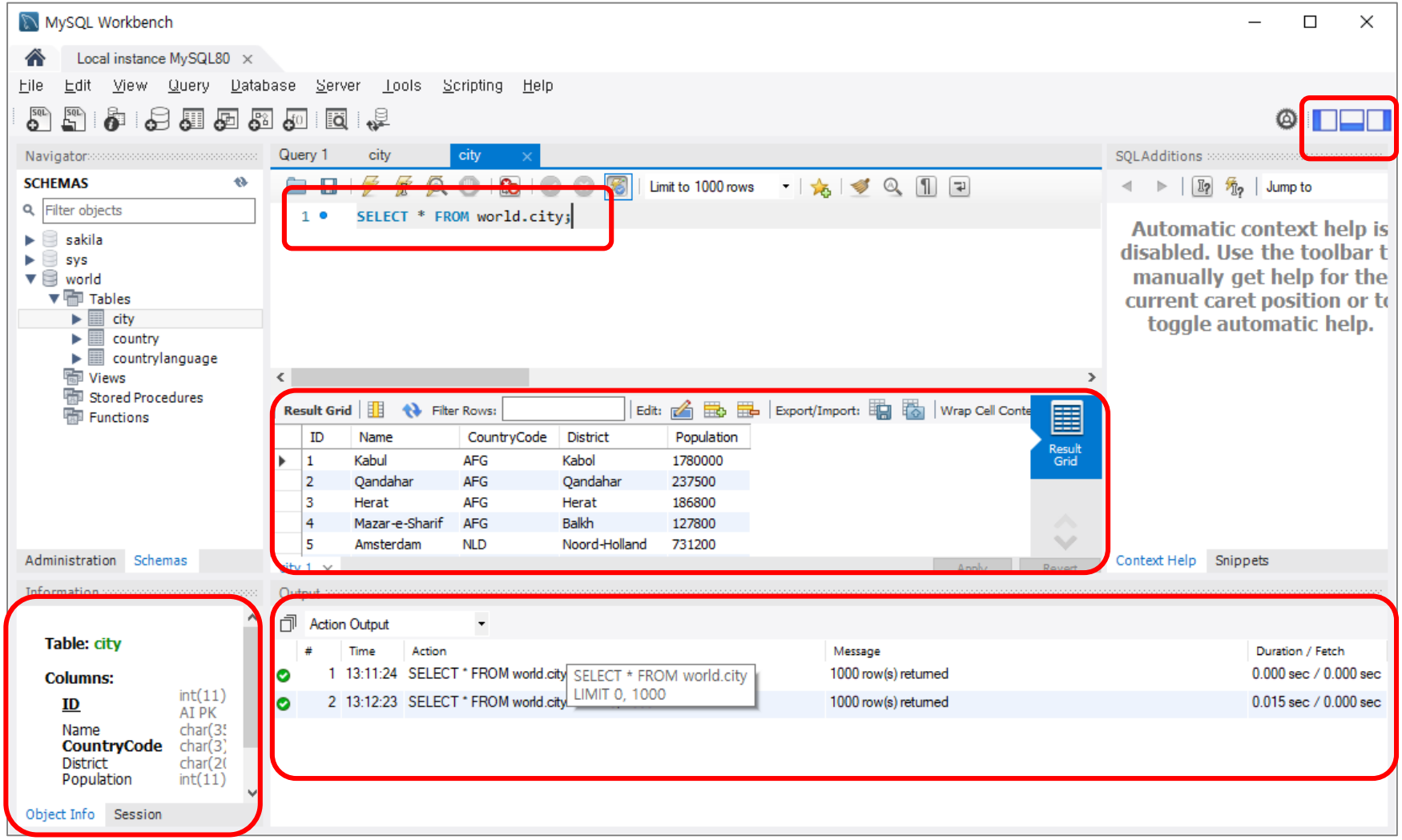
3. MySQL Workbench

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

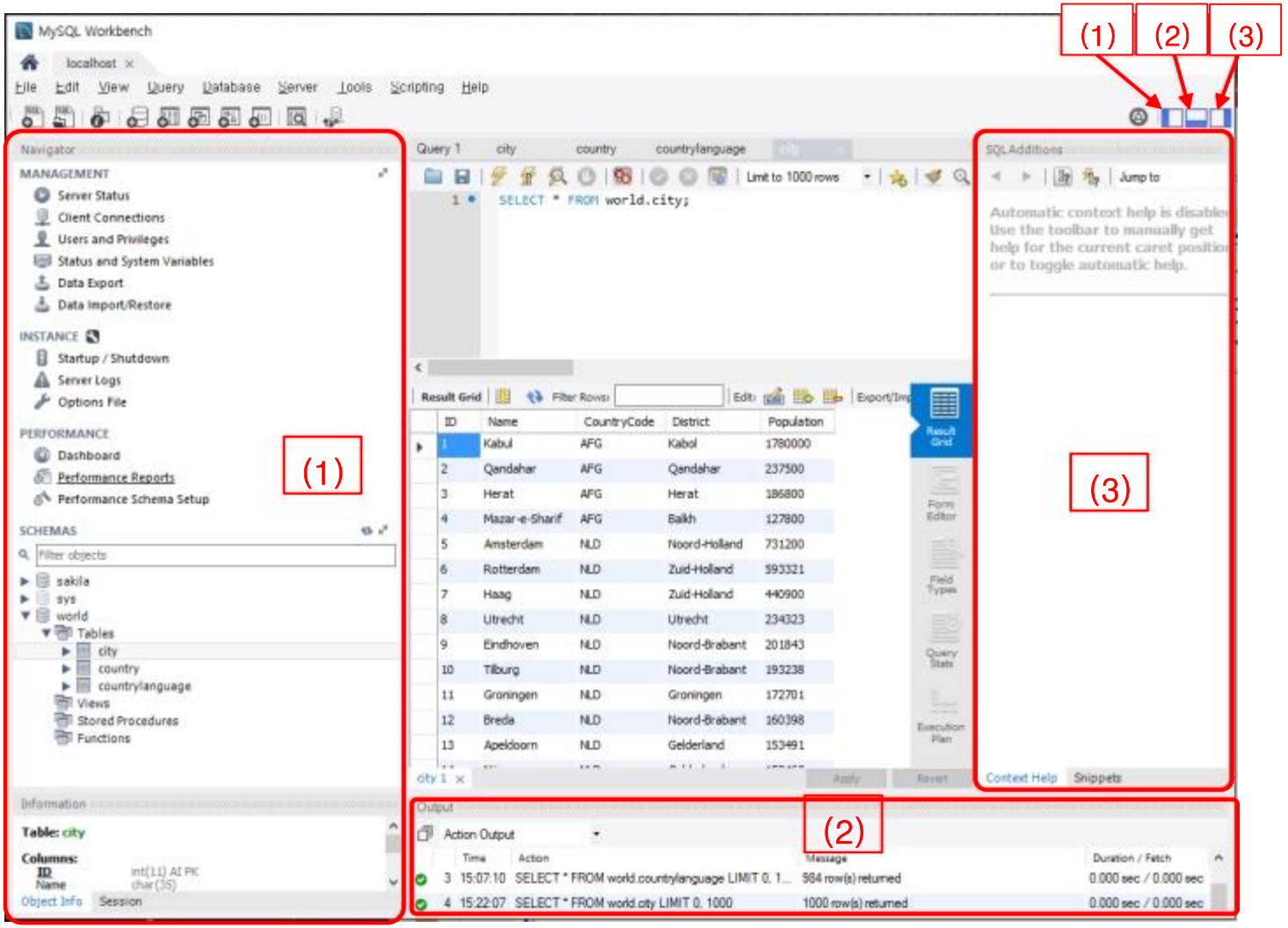




3. MySQL Workbench

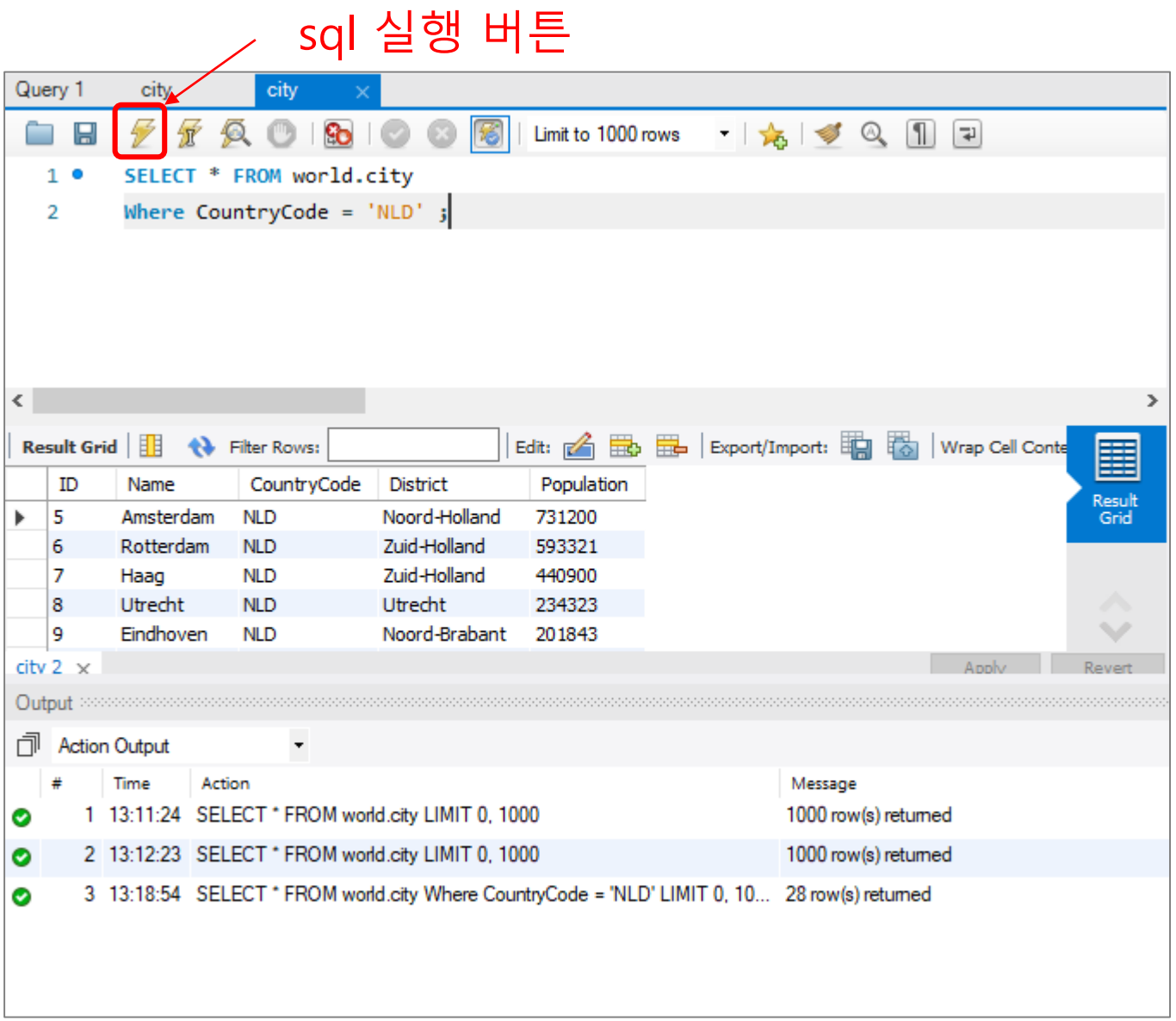


3. MySQL Workbench



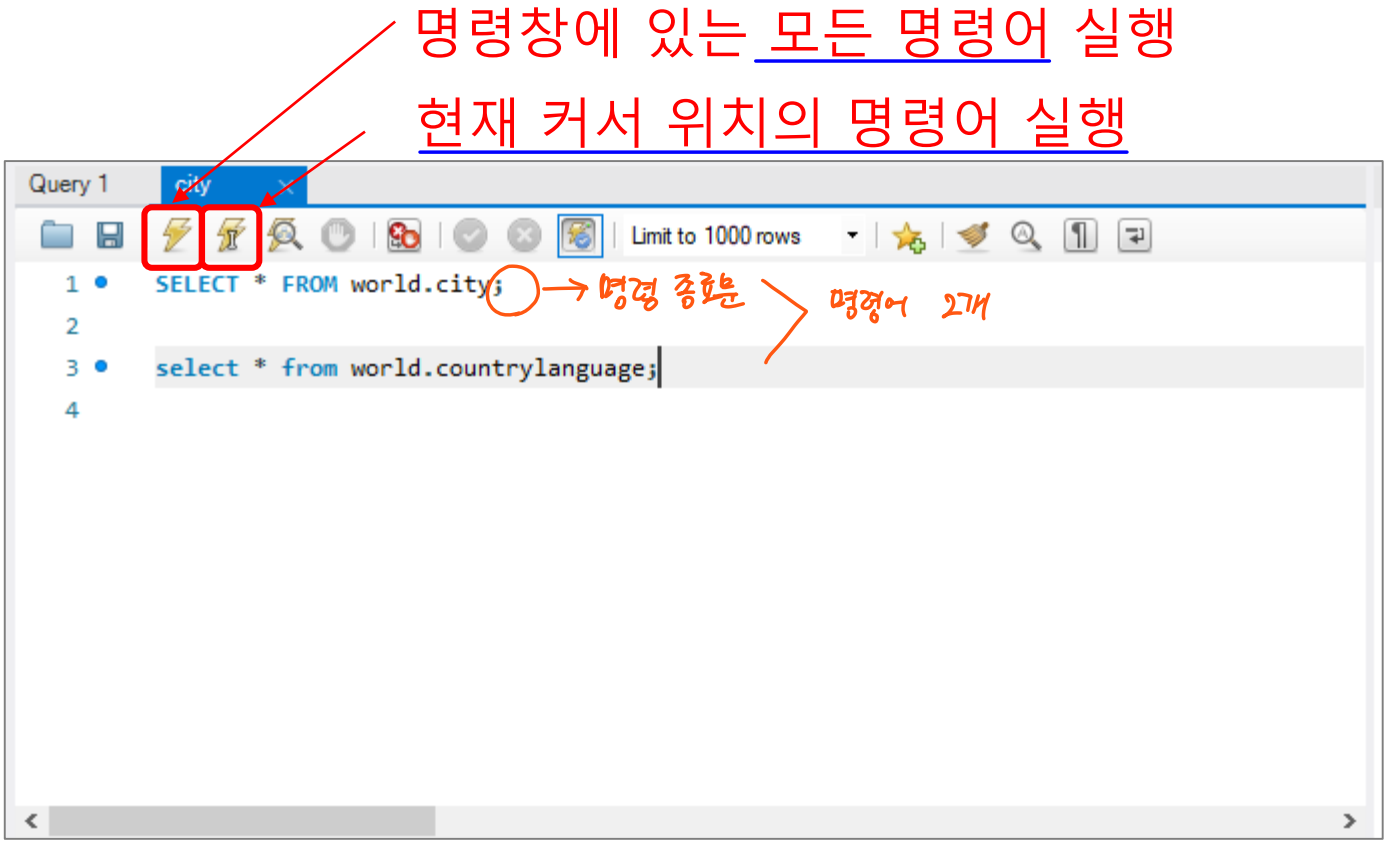
3. MySQL Workbench

- 네덜란드의 도시들만 보기



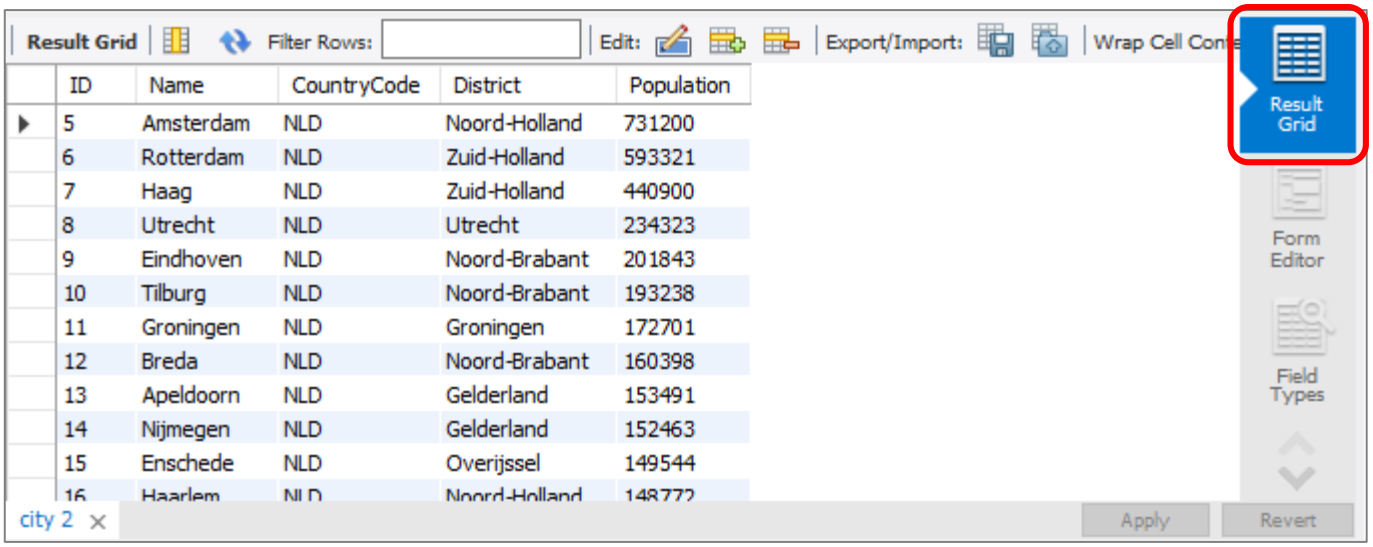
3. MySQL Workbench

- SQL 명령어가 여러 개인 경우 선택하여 실행하기



3. MySQL Workbench

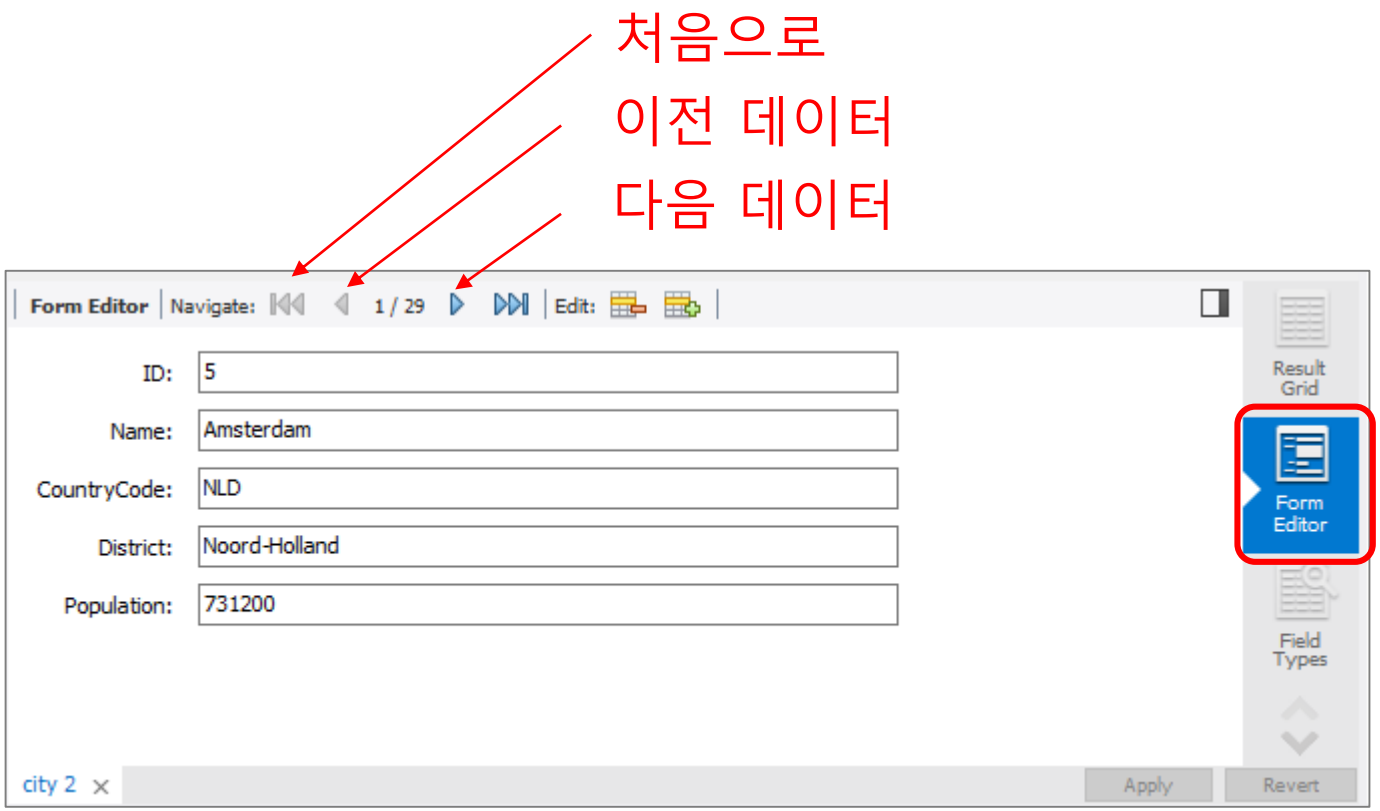
- 질의 결과를 격자 형태로 보기 (default)



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
10	Tilburg	NLD	Noord-Brabant	193238
11	Groningen	NLD	Groningen	172701
12	Breda	NLD	Noord-Brabant	160398
13	Apeldoorn	NLD	Gelderland	153491
14	Nijmegen	NLD	Gelderland	152463
15	Enschede	NLD	Overijssel	149544
16	Haarlem	NLD	Noord-Holland	148772

3. MySQL Workbench

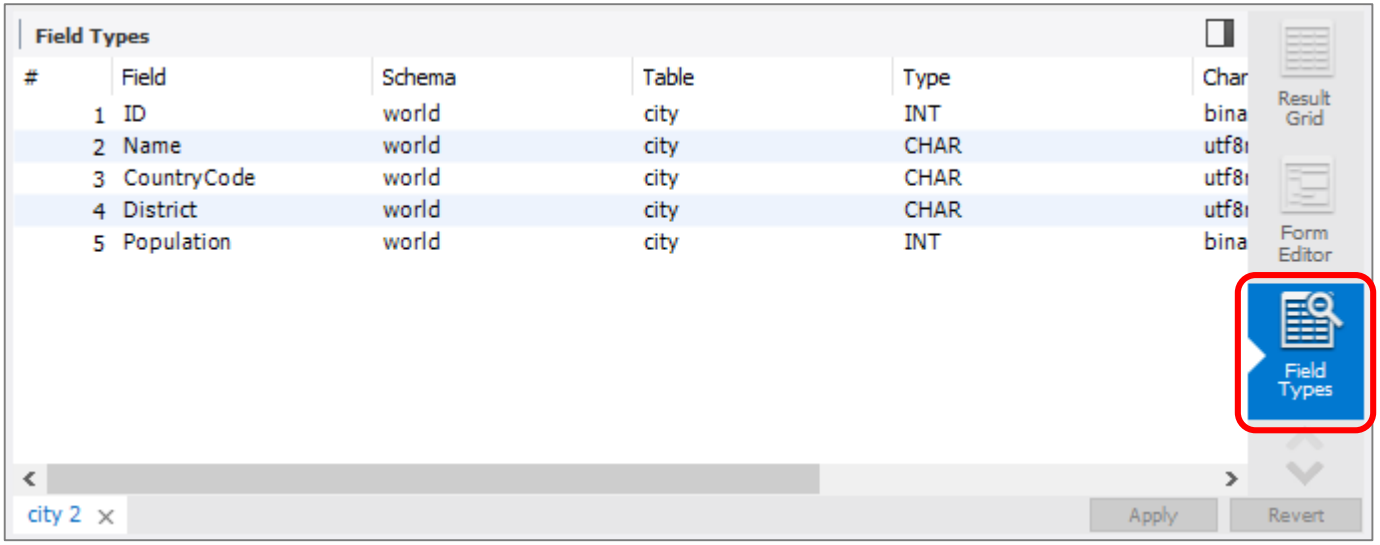
- 질의 결과를 폼 형태로 보기



편집 가능!!

3. MySQL Workbench

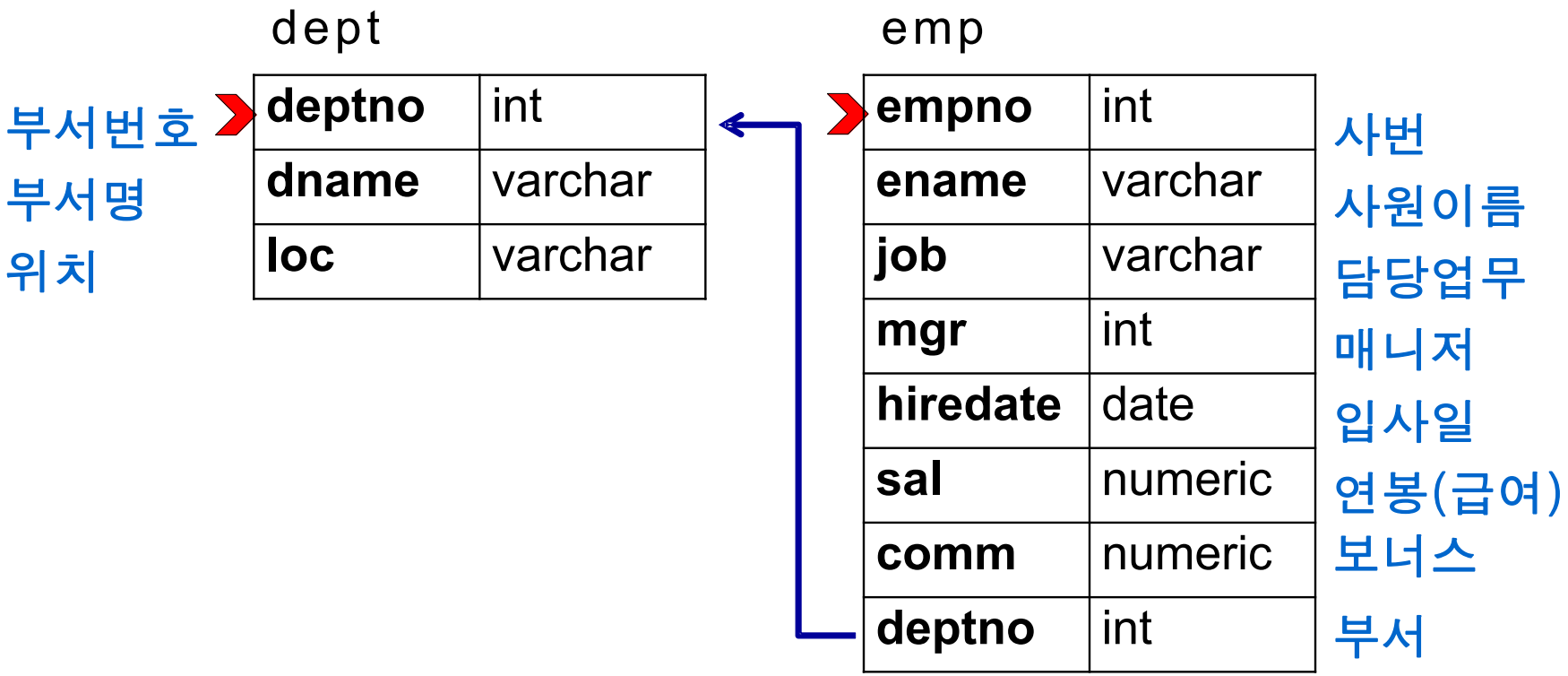
- 테이블 구조 정보 보기



Field = Column

4. 실습용 데이터베이스 생성하기

- 부서-사원 데이터베이스 (mydb)



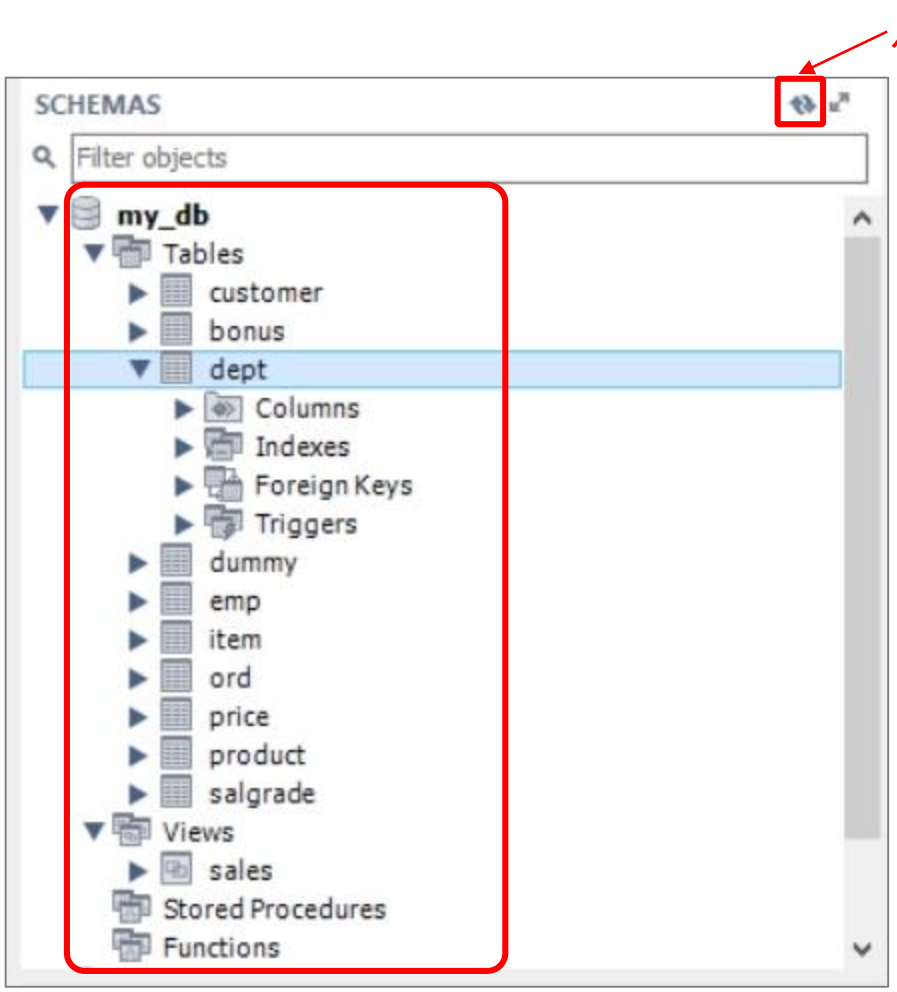
4. 실습용 데이터베이스 생성하기

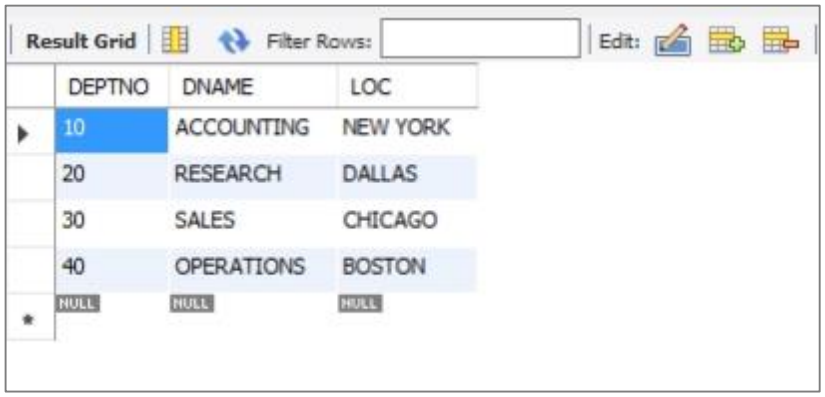
- create_table.sql 의 내용을 SQL 명령어 실행창에 붙여 넣기 한 후 실행

```
CREATE DATABASE my_db;
USE my_db;

CREATE TABLE DEPT (
 DEPTNO                DECIMAL(2) NOT NULL,
 DNAME                 VARCHAR(14),
 LOC                   VARCHAR(13),
 CONSTRAINT DEPT_PRIMARY_KEY PRIMARY KEY (DEPTNO));

INSERT INTO DEPT VALUES (10,'ACCOUNTING','NEW YORK');
INSERT INTO DEPT VALUES (20,'RESEARCH','DALLAS');
INSERT INTO DEPT VALUES (30,'SALES','CHICAGO');
INSERT INTO DEPT VALUES (40,'OPERATIONS','BOSTON');

CREATE TABLE EMP (
 EMPNO                 DECIMAL(4) NOT NULL,
 ENAME                 VARCHAR(10),
 JOB                   VARCHAR(9),
 MGR                   DECIMAL(4),
 HIREDATE              DATE,
 SAL                   DECIMAL(7,2),
 COMM                  DECIMAL(7,2),
 DEPTNO                DECIMAL(2) NOT NULL,
 CONSTRAINT EMP_SELF_KEY       FOREIGN KEY  (MGR) REFERENCES EMP(EMPNO),
 CONSTRAINT EMP_FOREIGN_KEY FOREIGN KEY (DEPTNO) REFERENCES DEPT (DEPTNO),
 CONSTRAINT EMP_PRIMARY_KEY PRIMARY KEY (EMPNO));

INSERT INTO EMP VALUES (7839,'KING','PRESIDENT',NULL,'81-11-17',5000,NULL,10);
INSERT INTO EMP VALUES (7698,'BLAKE','MANAGER',7839,'81-5-1',2850,NULL,30);
```

4. 실습용 데이터베이스 생성하기

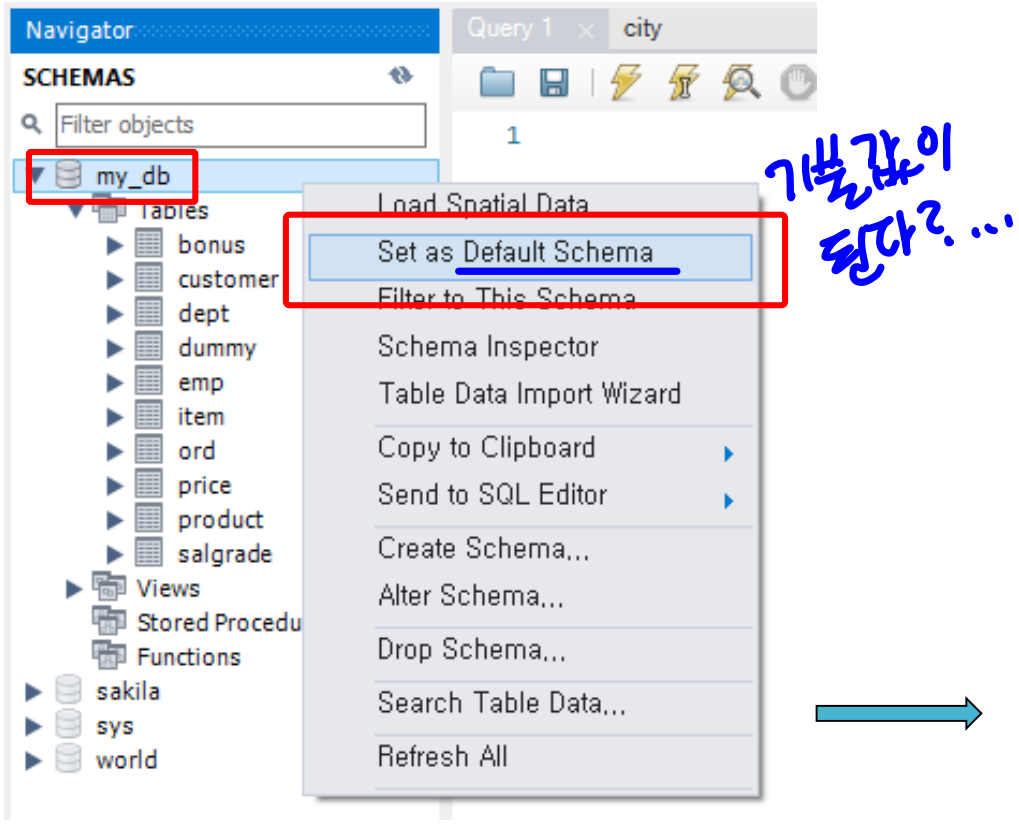
- my_db 가 정상적으로 생성되었는지 확인





4. 실습용 데이터베이스 생성하기

- 작업 데이터베이스 설정하기

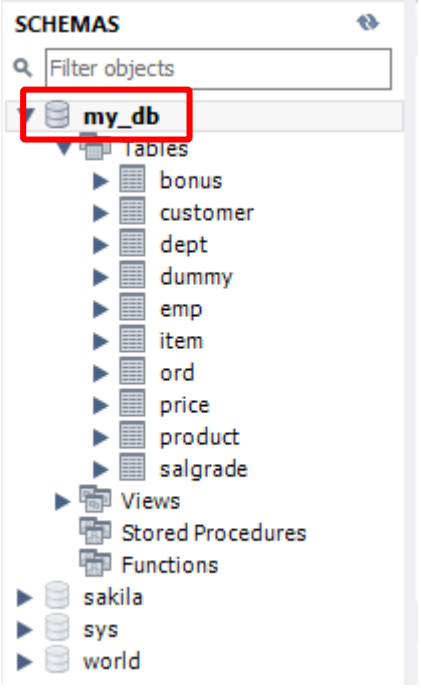


설정전

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

설정후

```
select * from emp ;
```



Note

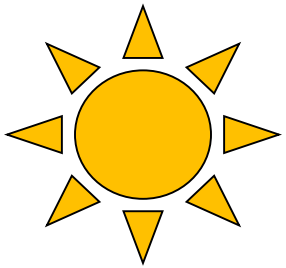
- SQL 에서는 명령어, 테이블이름, 컬럼 이름 등에 대해 대소문자를 구별하지 않는다 '데이터 값은 구별'

```
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