

04

CHAPTER

MySQL 툴과 유틸리티 사용법



Contents

- 01** MySQL Workbench의 기본 사용법
- 02** 데이터베이스 사용자 관리
- 03** 데이터베이스 백업과 복원

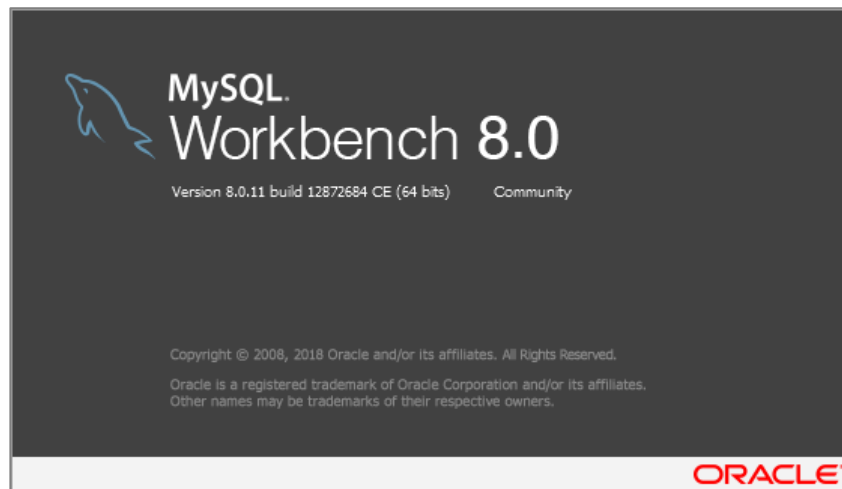
1-1 MySQL Workbench의 개요

- Workbench의 주요 기능
 - 데이터베이스 연결
 - 인스턴스 관리
 - 마법사(Wizard)를 이용한 MySQL 동작
 - 통합된 기능의 SQL 편집기 제공
 - 데이터베이스 모델링 기능 제공
 - 포워드/리버스 엔지니어링 기능 제공
 - 데이터베이스 인스턴스 시작/종료
 - 데이터베이스 내보내기/가져오기
 - 데이터베이스 계정 관리

1-2 MySQL Workbench 실행과 MySQL 서버 연결

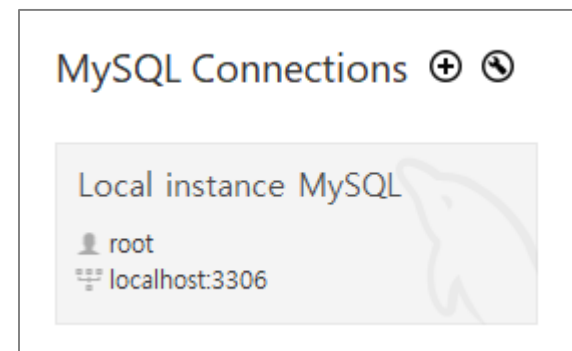
■ Workbench 실행

- 윈도우의 [시작]-[모든 앱]-[MySQL]-[MySQL Workbench 8.0 CE] 선택



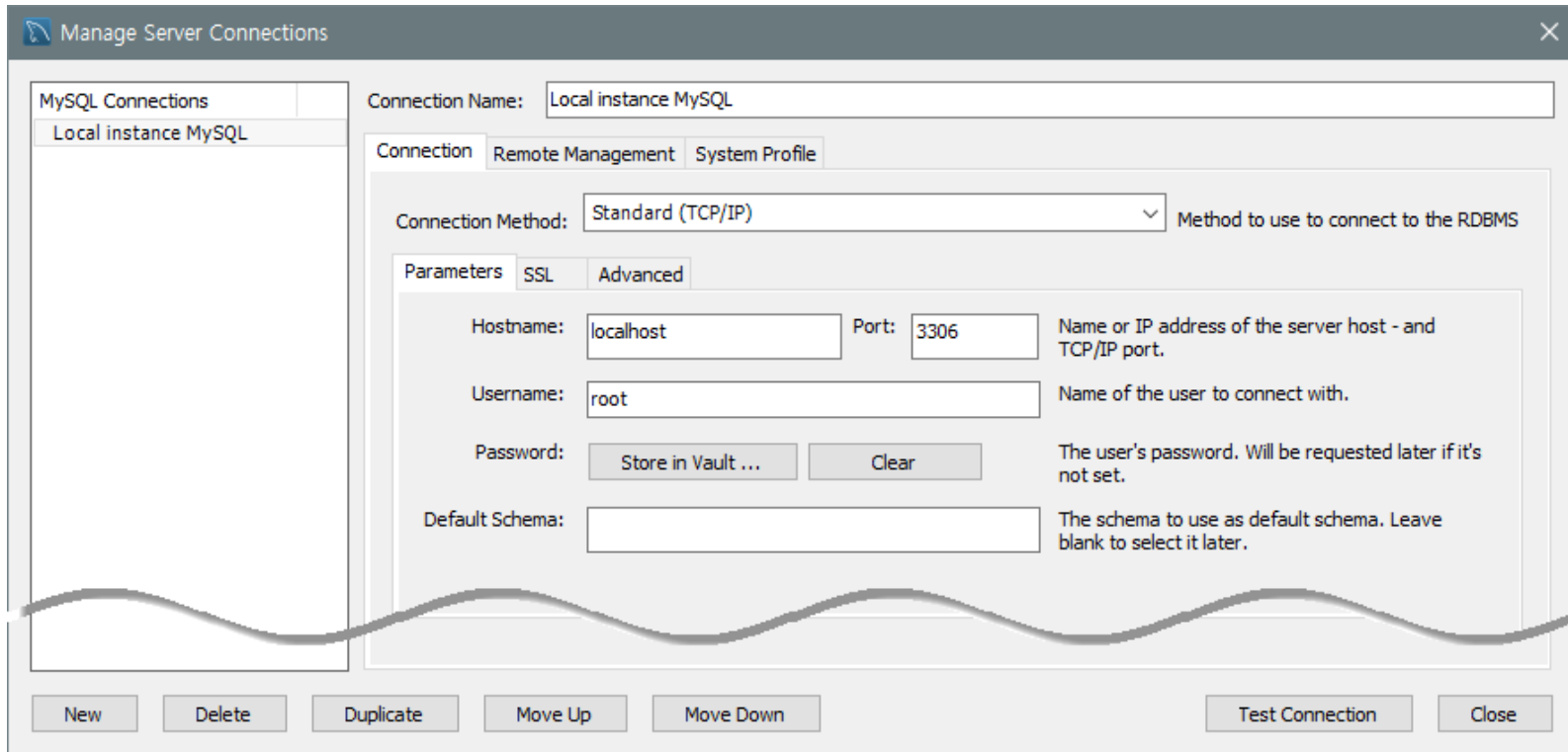
■ MySQL 연결

- Workbench를 실행하면 [MySQL Connections] 창이 뜬
- 기본 값으로 MySQL 관리자인 root 사용자, 서버는 자신의 컴퓨터를 의미하는 localhost, 포트는 3306번으로 접속하도록 설정되어 있음
- 다른 서버로 접속하려면 'Local instance MySQL'에서 마우스 오른쪽 버튼을 클릭하고 [Edit Connection]선택



1-2 MySQL Workbench 실행과 MySQL 서버 연결

- [Edit Connection]을 선택하면 뜨는 창
 - [Connection], [Remote Management], [System Profile] 탭이 있음



1-2 MySQL Workbench 실행과 MySQL 서버 연결

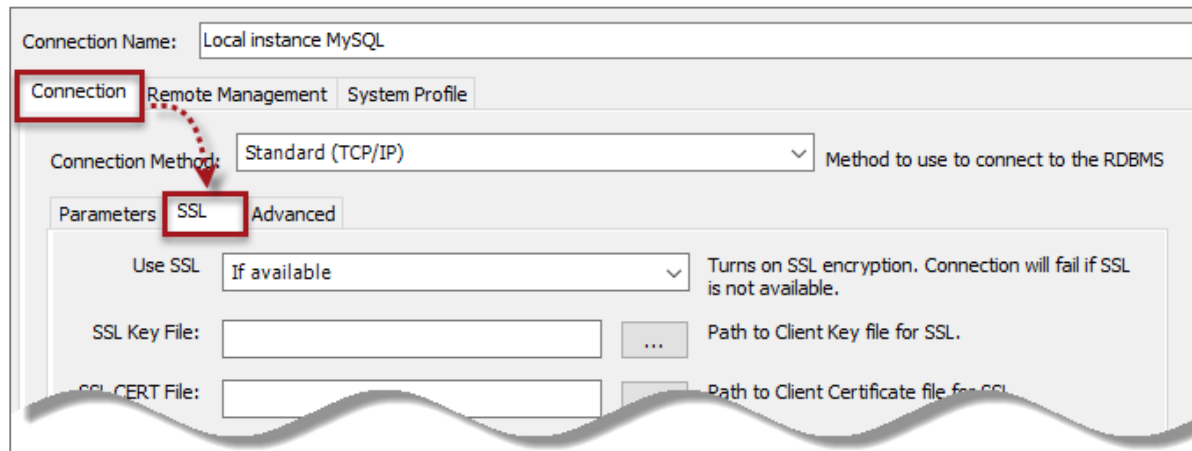
■ [Connection] 탭

■ [Parameters] 탭

- Hostname이 localhost 또는 127.0.0.1이라면 자신의 컴퓨터(Workbench가 설치된 컴퓨터)를 의미
- Port에는 접속할 MySQL의 포트 번호를 넣는데 특별한 경우가 아니면 3306을 사용
- Username에는 접속할 MySQL의 사용자를 넣음
- Password에서 <Store in Vault>를 클릭하면 미리 사용자의 비밀번호를 저장할 수 있음
- Default Schema는 접속 후에 기본적으로 선택되는 데이터베이스의 이름

■ [SSL] 탭

- SSL(Secure Socket Layer)은 보안을 위한 암호 규약으로, 서버와 클라이언트가 통신할 때 암호화를 통해 비밀을 유지하고 보안을 강화함
- 서버에서 특별히 설정하지 않았다면 그대로 둠

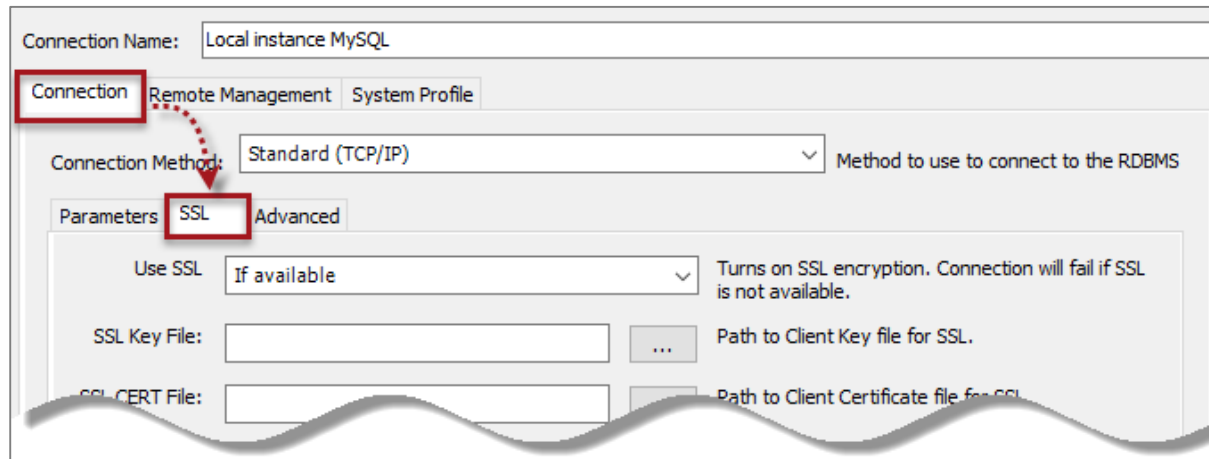


1-2 MySQL Workbench 실행과 MySQL 서버 연결

■ [Connection] 탭

■ [Advanced] 탭

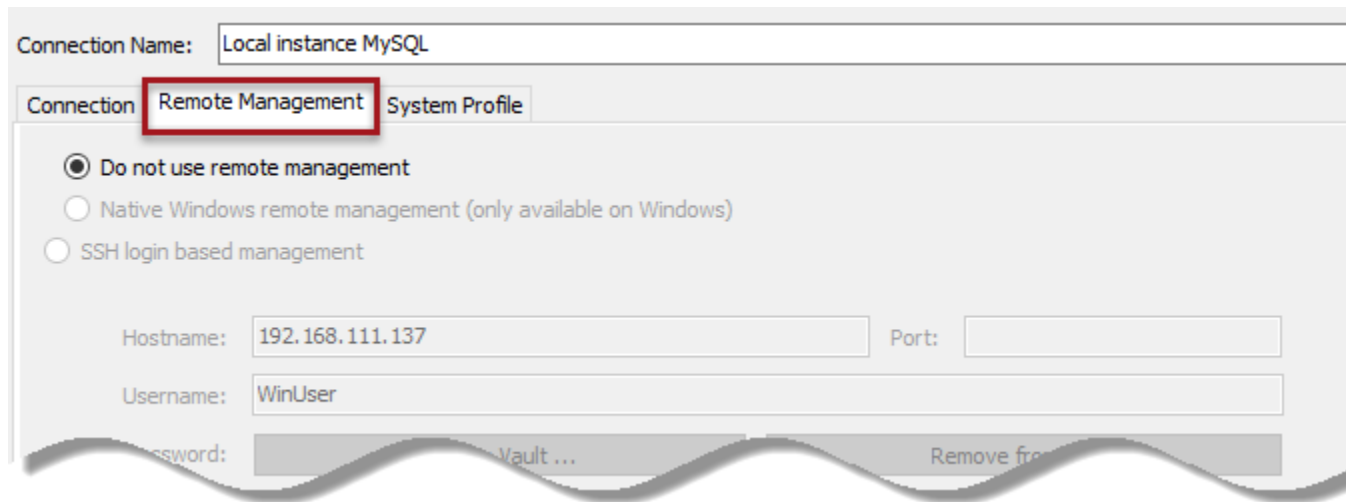
- 프로토콜의 압축, 인증 방식 등을 설정
- 특별한 경우가 아니라면 변경할 필요 없음



1-2 MySQL Workbench 실행과 MySQL 서버 연결

■ [Remote Management] 탭

- 원격 관리를 위한 여러 가지 설정을 함
- 이 부분이 활성화되려면 Hostname이 IP 주소로 설정되어 있어야 함
- Native Windows remote management를 선택하면 MySQL 서버가 설치된 운영체제가 윈도우인 경우에만 설정할 수 있음
- SSH login based management는 SSH 서버 기반으로 원격 접속이 됨



1-2 MySQL Workbench 실행과 MySQL 서버 연결

■ [System Profile] 탭

- 접속할 서버의 운영체제 종류와 MySQL 설정 파일의 경로 등을 설정
- System Type은 FreeBSD, Linux, MacOS X, OpenSolaris, Windows 중에서 선택
- Configuration File에서는 MySQL의 설정 파일이 경로와 함께 지정됨

The screenshot shows the 'System Profile' tab in the MySQL Workbench connection configuration window. The 'Connection Name' is 'Local instance MySQL'. The 'System Profile' tab is selected and highlighted with a red box. Below the tabs, there is a descriptive text: 'Information about the server and MySQL configuration, such as path to the configuration file, command to start or stop it etc. You may pick a preset configuration profile or customize one for your needs.' The configuration fields are as follows:

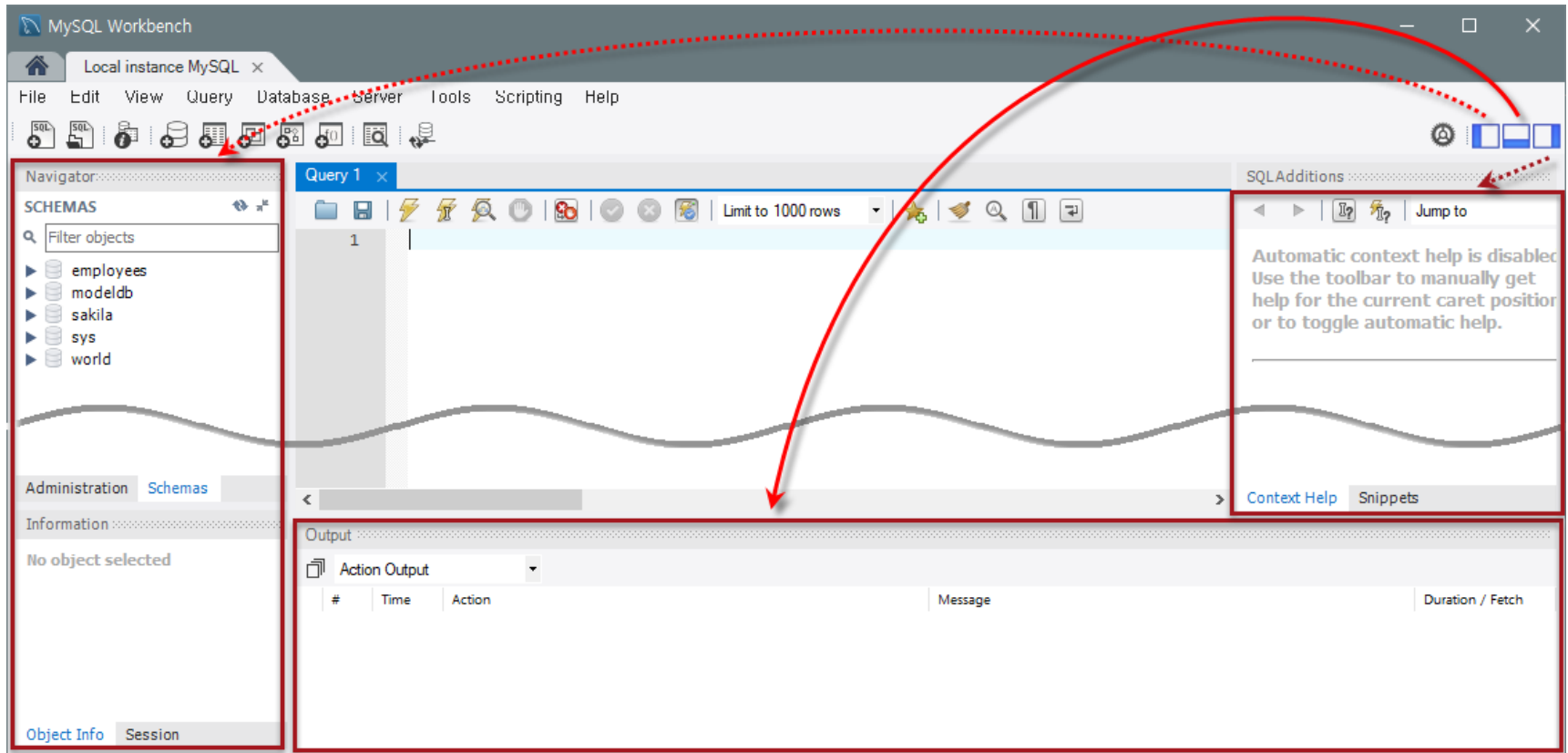
- System Type:** Windows (dropdown menu)
- Installation Type:** Custom (dropdown menu)
- Configuration File:** C:\ProgramData\MySQL\MySQL Server 8.0\my.ini (text field with a browse button '...')
- Configuration File Section:** mysqld (text field)
- Windows Service Name:** MySQL (text field)

At the bottom, there is a 'Windows Hint:' section with a wavy border, containing the text 'MySQL is configured for...'.

1-3 MySQL Workbench의 화면 구성

■ 패널

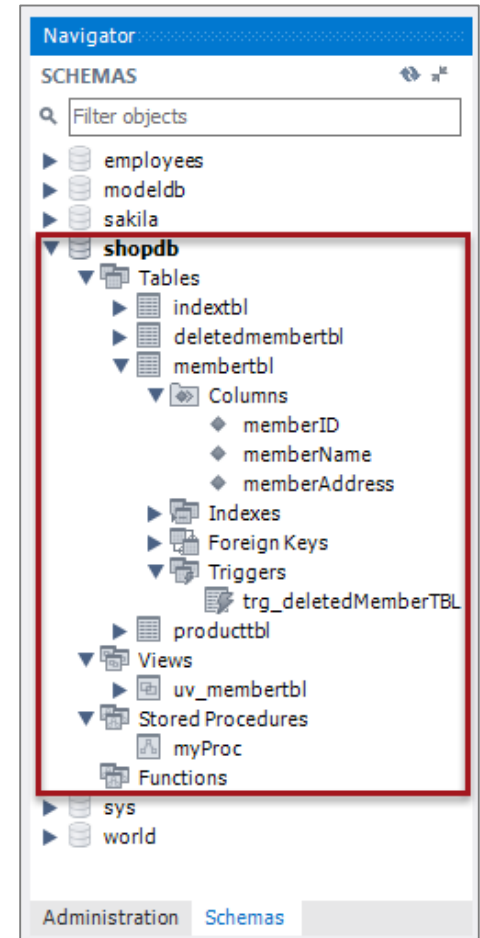
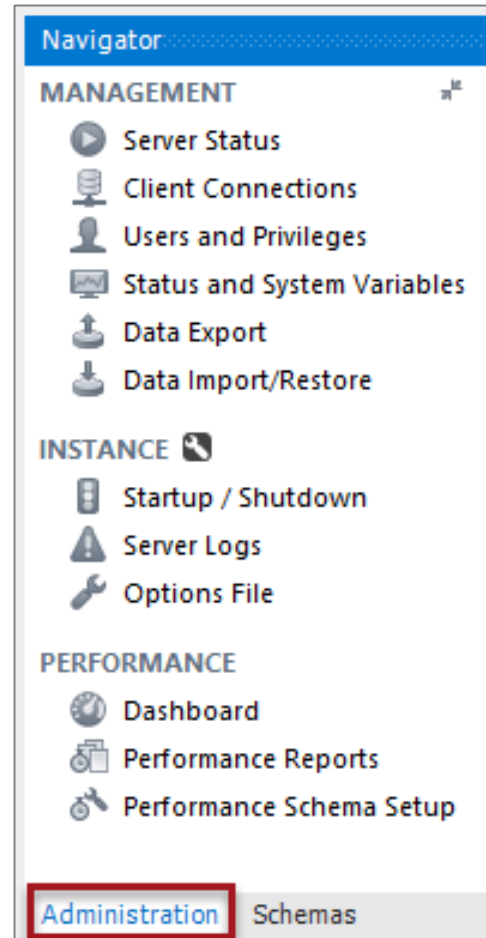
■ Workbench의 화면 구성



1-3 MySQL Workbench의 화면 구성

■ 내비게이터

- MySQL의 관리 및 운영을 위한 도구
- 각 항목에서 ►를 클릭하면 확장할 수 있음



1-3 MySQL Workbench의 화면 구성

■ 내비게이터가 각 탭에서 제공하는 기능

표 4-1 내비게이터의 기능

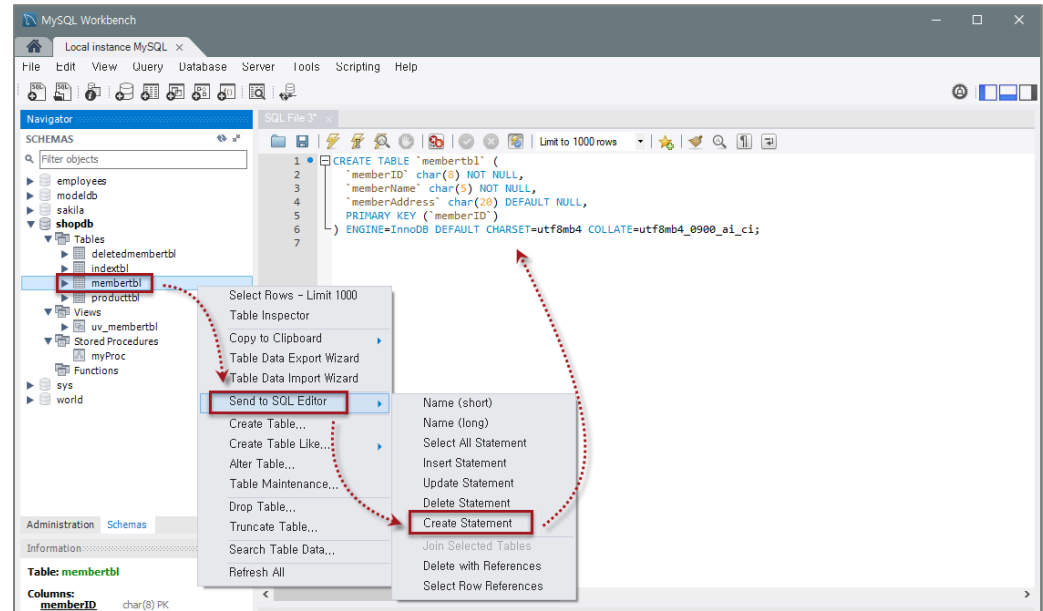
| 탭 | 기능 |
|--------------------|---|
| [Schemas] 탭 | <ul style="list-style-type: none">• 데이터베이스(스키마) 생성 및 삭제• 데이터베이스 개체(테이블, 뷰, 인덱스, 스토어드 프로시저, 함수 등) 생성 및 관리• 데이터베이스 속성 조회 |
| [Administration] 탭 | <ul style="list-style-type: none">• MANAGEMENT<ul style="list-style-type: none">– MySQL 서버의 가동 상태, 설치된 폴더 등 확인– MySQL 서버에 연결되어 있는 클라이언트의 정보 확인– 사용자 생성, 삭제 및 권한 관리– 서버 변수 값 확인– 데이터 내보내기/가져오기• INSTANCE<ul style="list-style-type: none">– MySQL 연결 정보 관리– MySQL 인스턴스 중지 및 시작– 서버에 기록된 로그 조회– MySQL 옵션 파일의 설정 정보 확인 및 변경• PERFORMANCE<ul style="list-style-type: none">– 네트워크 상태 및 MySQL의 성능 상태 확인– 성능 상태 보고서 작성– 성능 구성 설정 |

[실습 4-1] 내비게이터에서 SQL 문 자동으로 생성하기

교재 119~120p 참고

1 테이블을 만드는 SQL 문 자동 생성하기

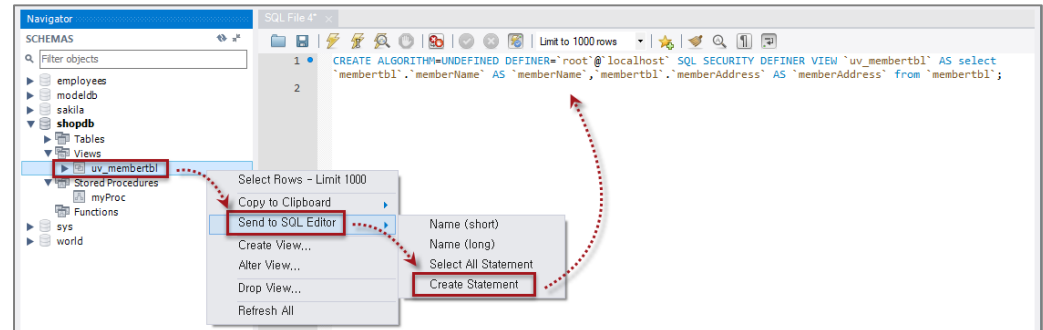
1-1 [Create Statement] 선택



2 뷰를 만드는 SQL 문 자동으로 생성하기

2-1 uv_memberTBL 뷰의

CREATE 문 생성

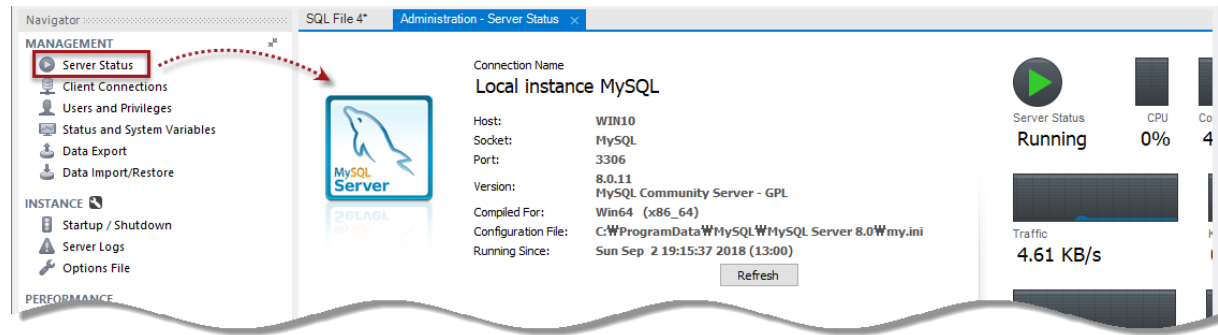


[실습 4-2] 내비게이터에서 MySQL 서버 관리하기

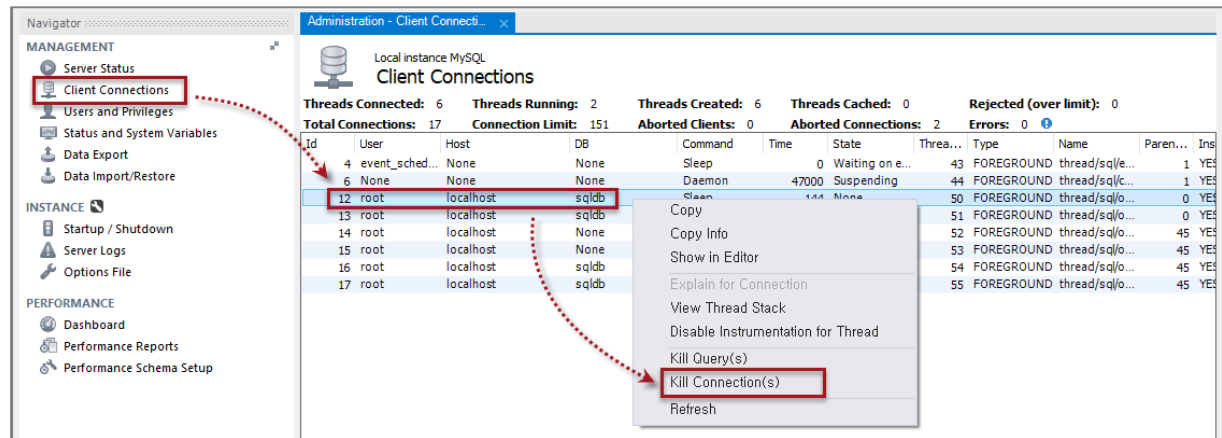
교재 120~124p 참고

1 [MANAGEMENT] 탭

1-1 [Administration] 탭에서
[MANAGEMENT]의
[Server Status]를 클릭



1-2 [Client Connections] 클릭



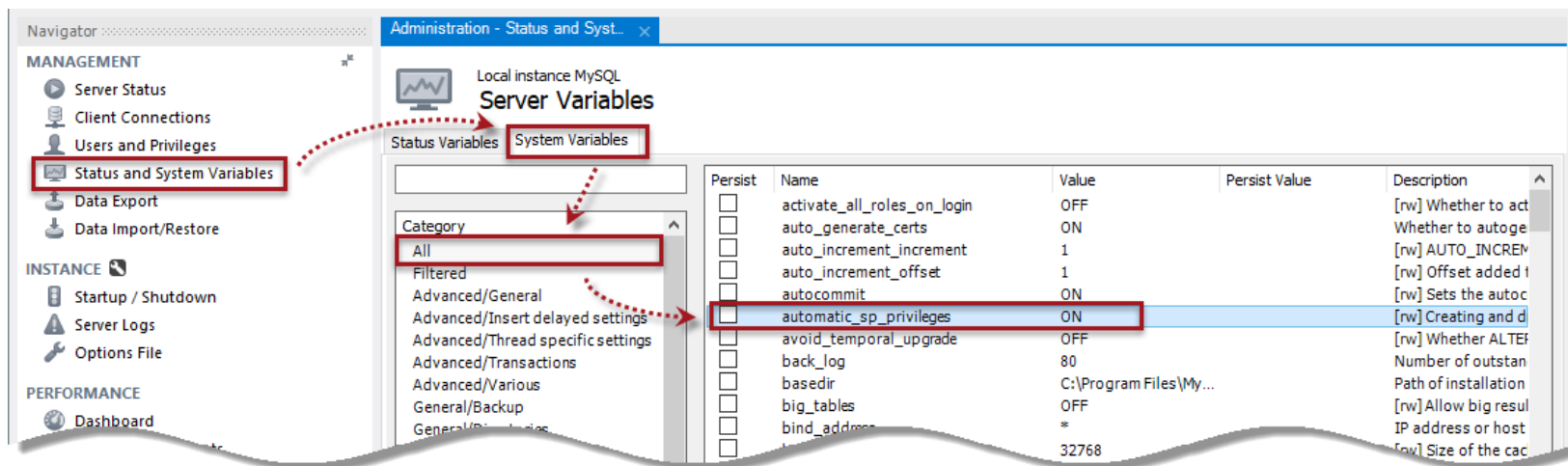
[실습 4-2] 내비게이터에서 MySQL 서버 관리하기

교재 120~124p 참고

1-3 [Users and Privileges] 클릭

1-4 [MANAGEMENT]의

[Status and System Variables] 클릭



1-5 [Data Export], [Data Import/Restore] 클릭

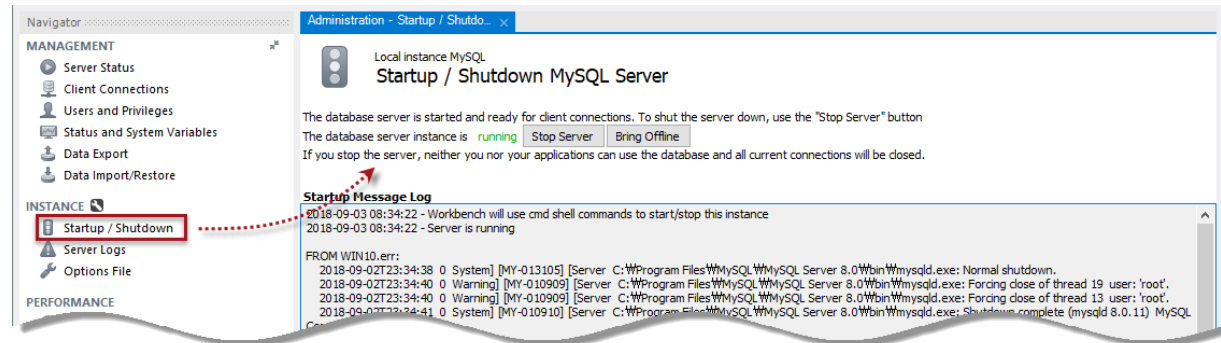
[실습 4-2] 내비게이터에서 MySQL 서버 관리하기

교재 120~124p 참고

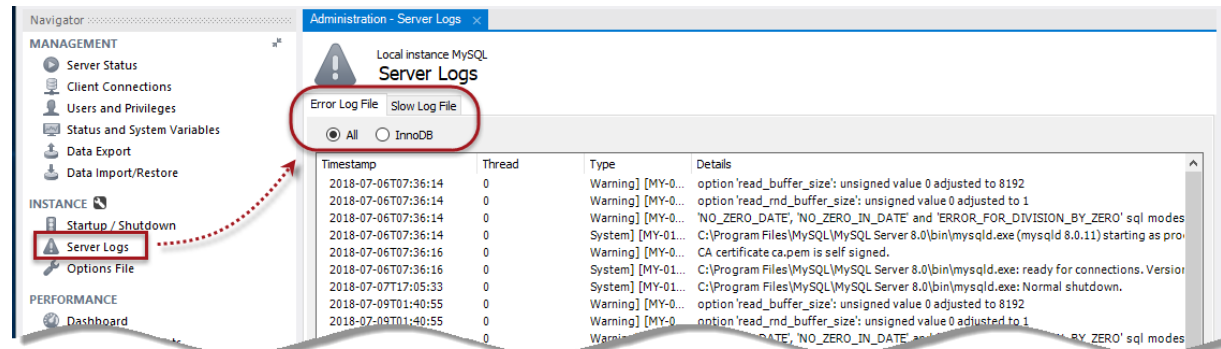
2 [INSTANCE] 탭

2-1 [INSTANCE]의

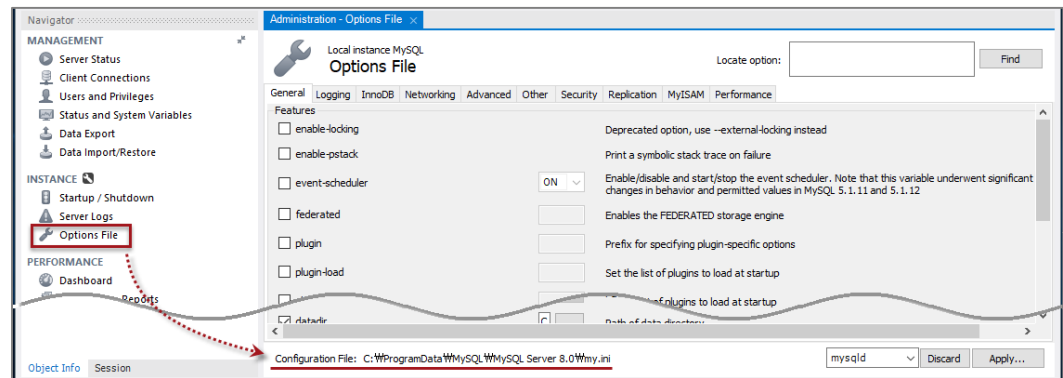
[Startup / Shutdown] 클릭



2-2 [Server Logs] 클릭

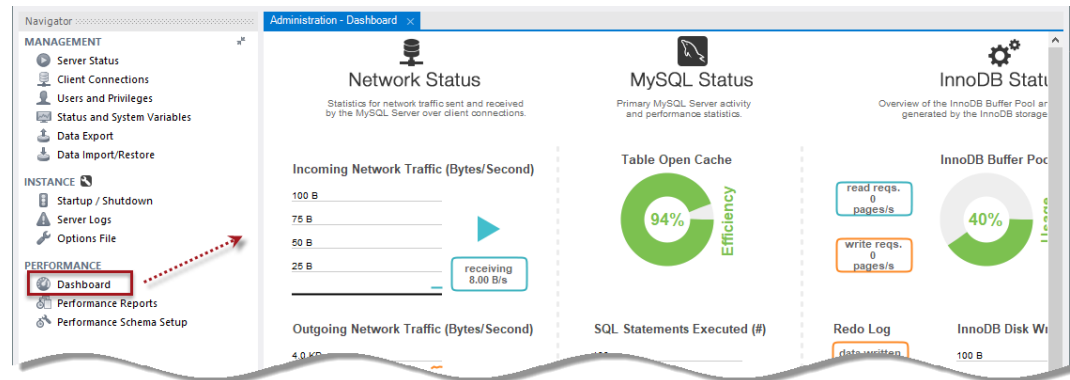


2-3 [Options File] 클릭



3 [PERFORMANCE] 탭

3-1 [PERFORMANCE]의 [Dashboard] 클릭



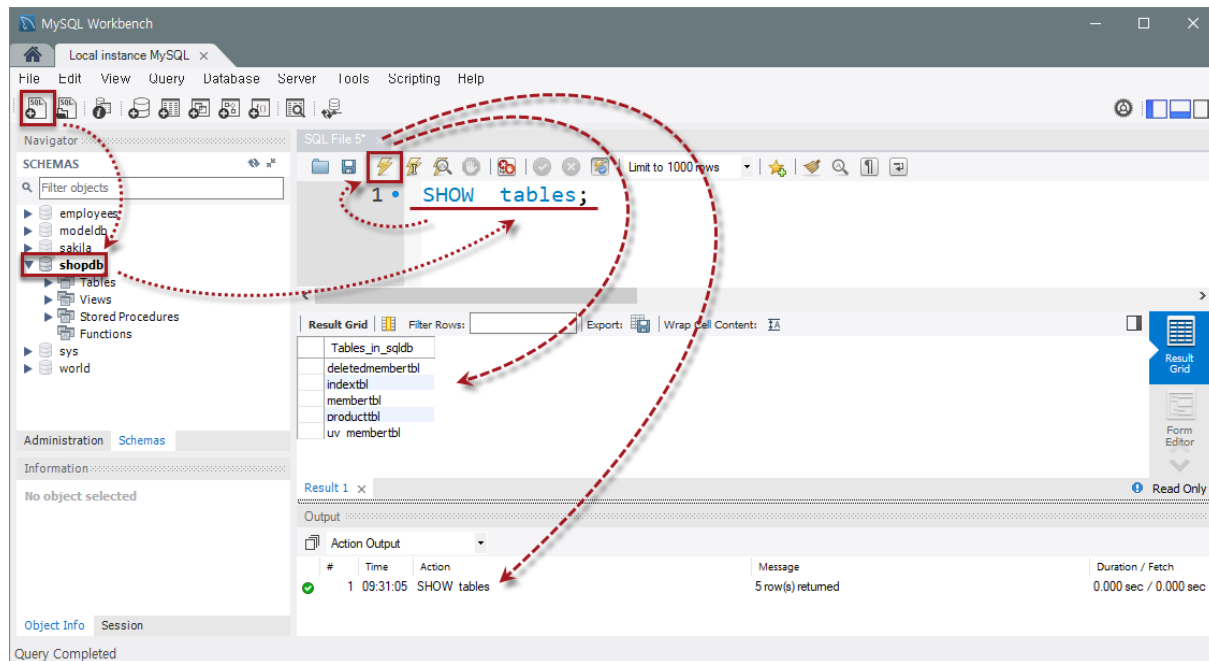
3-2 [Performance Reports] 클릭

3-3 [Performance Schema Setup] 클릭

4 쿼리 창 활용

■ 쿼리 창에서 SQL 문 실행 순서

- ① Workbench 상단 왼쪽의 Create a new SQL tab for executing queries 아이콘을 클릭 또는 메뉴의 [File]-[New Query Tab] 선택
- ② 작업할 데이터베이스를 [SCHEMAS] 탭에서 더블클릭
- ③ SQL 문을 문법에 맞게 입력
- ④ SQL 문에 이상이 없으면 툴바의 Execute the selected portion ... 아이콘을 클릭 또는 [Ctrl]+[Shift]+[Enter]를 눌러 SQL 문을 실행
- ⑤ [Output] 창에서 실행 결과 확인

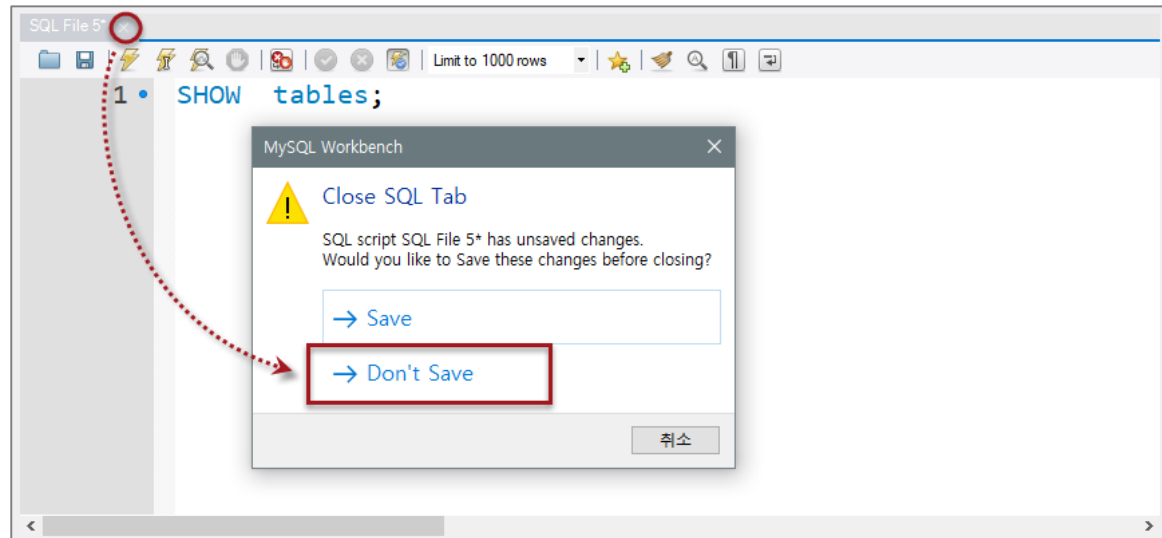


[실습 4-3] 쿼리 창 활용하기

교재 124~132p 참고

1 열려 있는 쿼리 창 닫기

1-1 열려 있는 쿼리 창 모두 닫기



2 SQL 문 실행하기

2-1 새로운 쿼리 창 열기

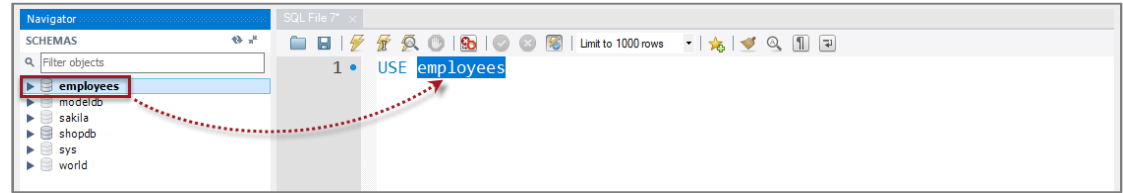
2-2 다음 SQL 문 입력 후 실행

```
USE employees;  
SELECT * FROM employees;
```

[실습 4-3] 쿼리 창 활용하기

교재 124~132p 참고

2-3 내비게이터에서 개체 이름을
마우스로 드래그하여 입력



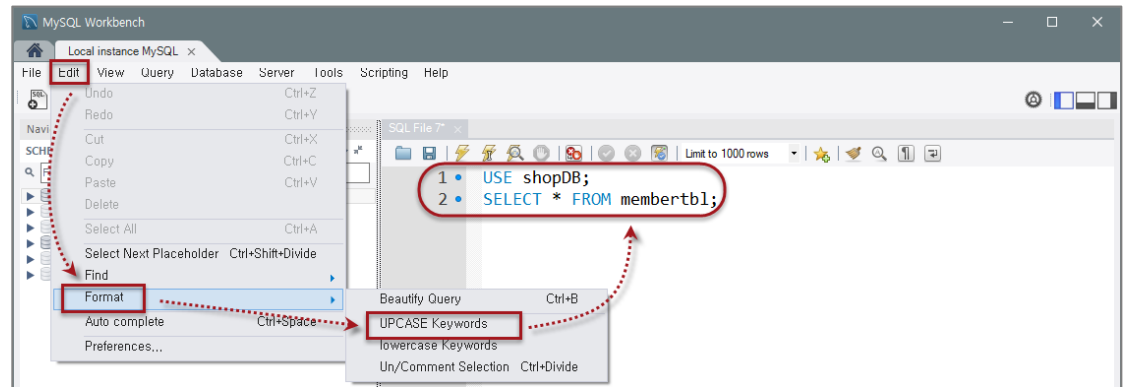
3 예약어로 대문자 또는 소문자로 변경하기

3-1 다음 SQL 문을 쿼리 창에 입력(이때 예약어(Keyword)는 모두 소문자로 입력)

```
use shopDB;  
select * from membertbl;
```

3-2 메뉴의 [Edit]-[Format]

-[UPCASE Keywords] 선택

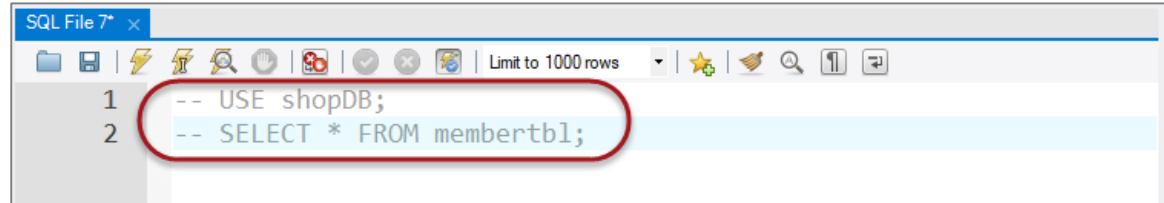


[실습 4-3] 쿼리 창 활용하기

교재 124~132p 참고

4 선택한 부분을 일괄적으로 주석 처리하거나 해제하기

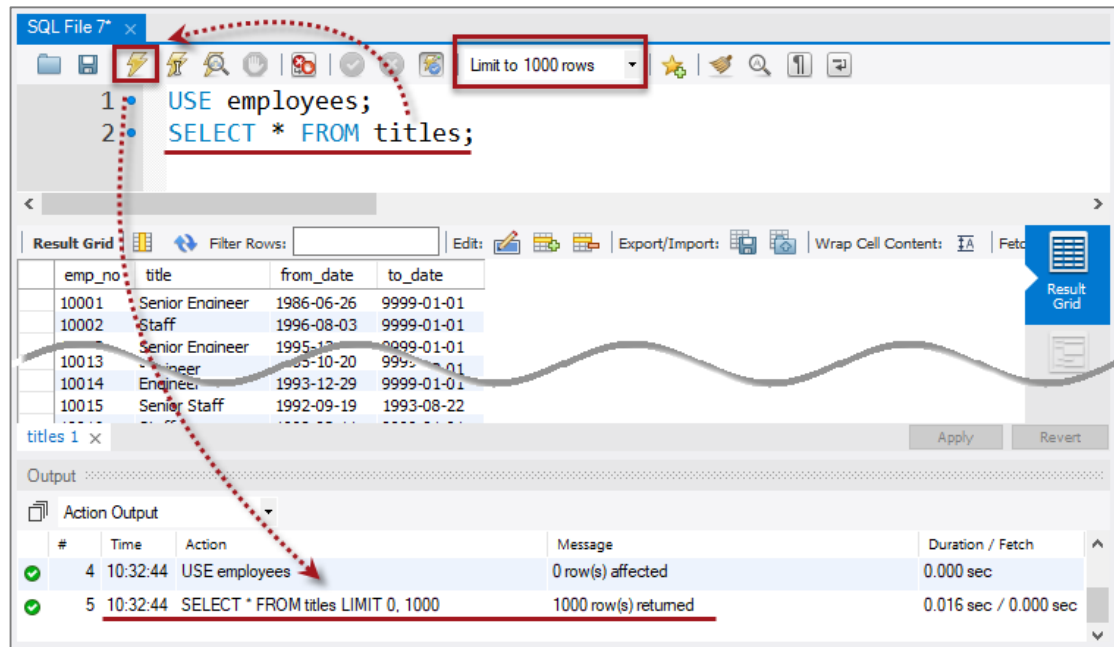
- 4-1 SQL 문 2줄을 마우스로 드래그
한 후 메뉴의 [Edit]-[Format]
-[Un/Comment Selection] 선택



5 SQL 문 실행 결과 행수 조절하기

- 5-1 다음 SQL 문 입력 후 실행

```
USE employees;  
SELECT * FROM titles;
```

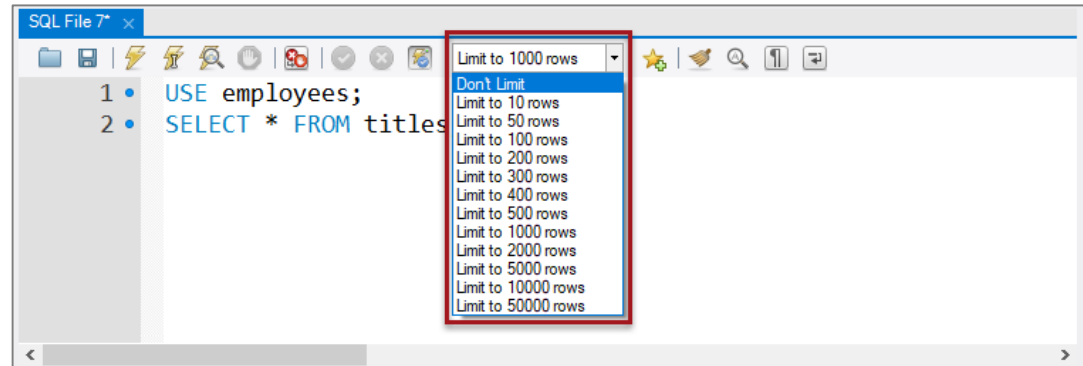


[실습 4-3] 쿼리 창 활용하기

교재 124~132p 참고

5-2 SQL 문의 실행 결과 행수는

'제한 없음(Don't Limit)'으로 설정



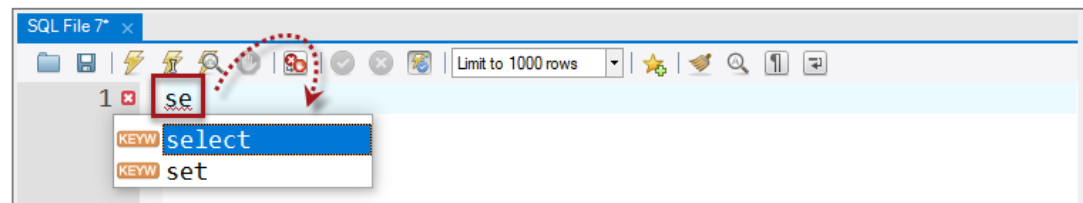
5-3 다시 SQL 문을 실행하면 40만 개가 넘는 결과가 나오는 것을 확인할 수 있음

| Output | | | | |
|---------------|----------|----------------------|------------------------|-----------------------|
| Action Output | | | | |
| # | Time | Action | Message | Duration / Fetch |
| ✓ 6 | 14:49:19 | USE employees | 0 row(s) affected | 0.000 sec |
| ✓ 7 | 14:49:19 | SELECT * FROM titles | 443308 row(s) returned | 0.016 sec / 0.969 sec |

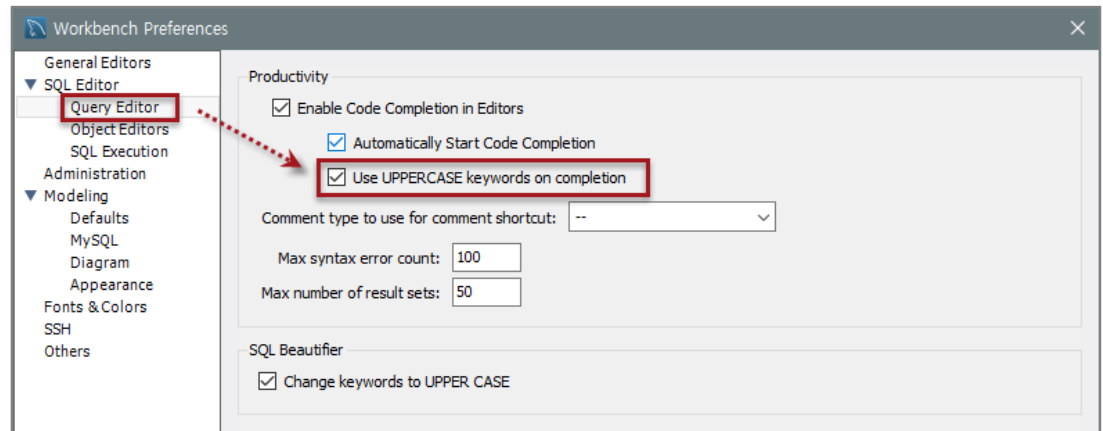
6 소문자로 자동 완성되는 예약어를 대문자로 설정하기

6-1 자동 완성되는 예약어는

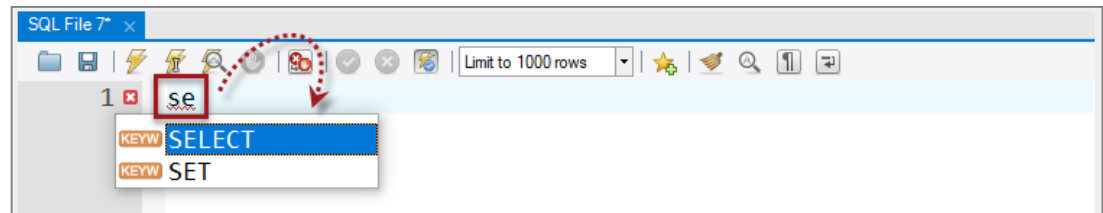
기본적으로 소문자로 입력됨



6-2 'USE UPPERCASE keywords on completion'에 체크 표시를 하고 <OK>를 클릭

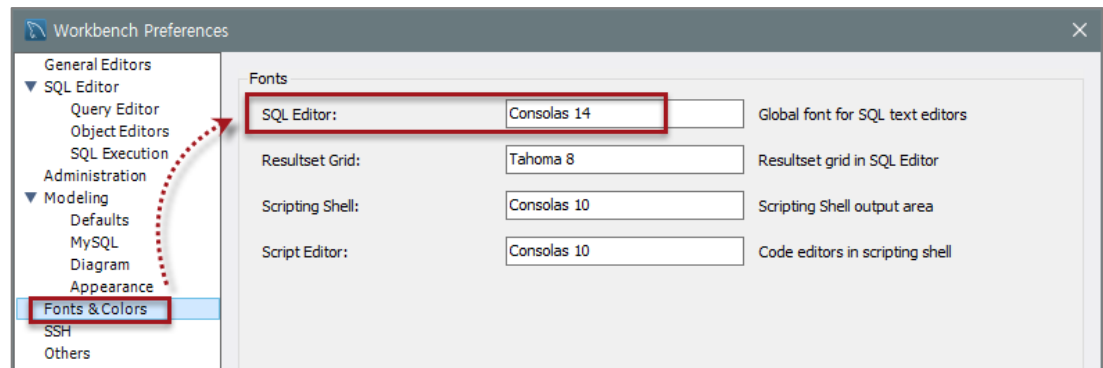


6-3 자동 완성되는 예약어가 대문자로 제시됨



7 쿼리 창의 글꼴 변경하기

7-1 메뉴의 [Edit]-[Preferences] 선택 후 [Fonts & Colors]를 선택



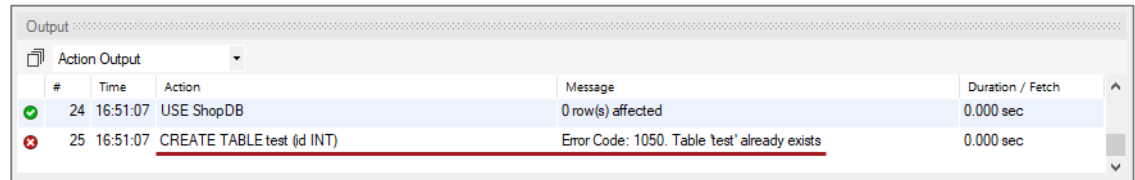
8 여러 SQL 문 실행하기

8-1 다음 구문 입력 후 Execute the selected portion ... 아이콘 클릭

```
USE ShopDB;  
CREATE TABLE test (id INT);
```

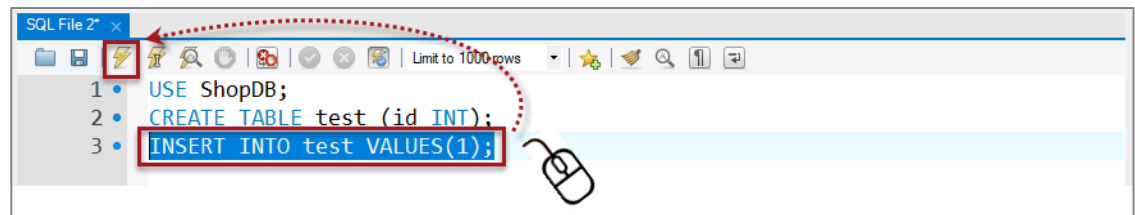
8-2 앞에서 입력한 구문을 지우지 말고 그 아래에 INSERT 문 추가 후 실행

```
USE ShopDB;  
CREATE TABLE test (id INT);  
INSERT INTO test VALUES(1);
```



| # | Time | Action | Message | Duration / Fetch |
|----|----------|----------------------------|---|------------------|
| 24 | 16:51:07 | USE ShopDB | 0 row(s) affected | 0.000 sec |
| 25 | 16:51:07 | CREATE TABLE test (id INT) | Error Code: 1050. Table 'test' already exists | 0.000 sec |

8-3 실행할 부분만 마우스로 드래그 하여 선택한 후 실행



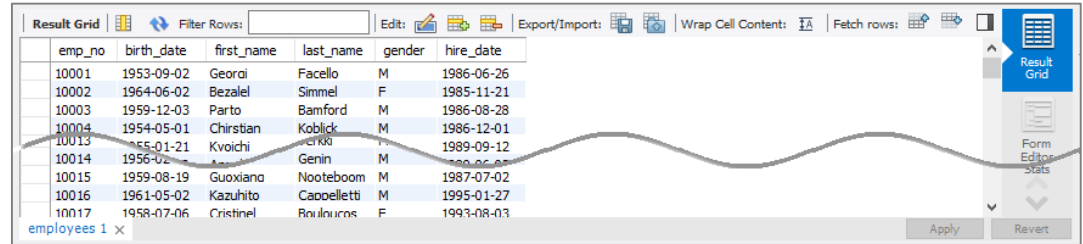
[실습 4-3] 쿼리 창 활용하기

교재 124~132p 참고

9 실행 결과를 필터링한 후 저장하기

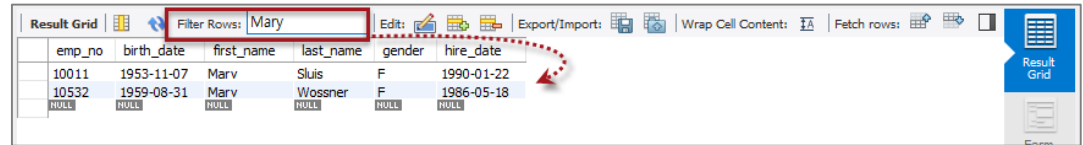
9-1 실행할 SQL 문만 마우스로 드래그하여 실행

```
USE employees;  
SELECT * FROM employees;
```



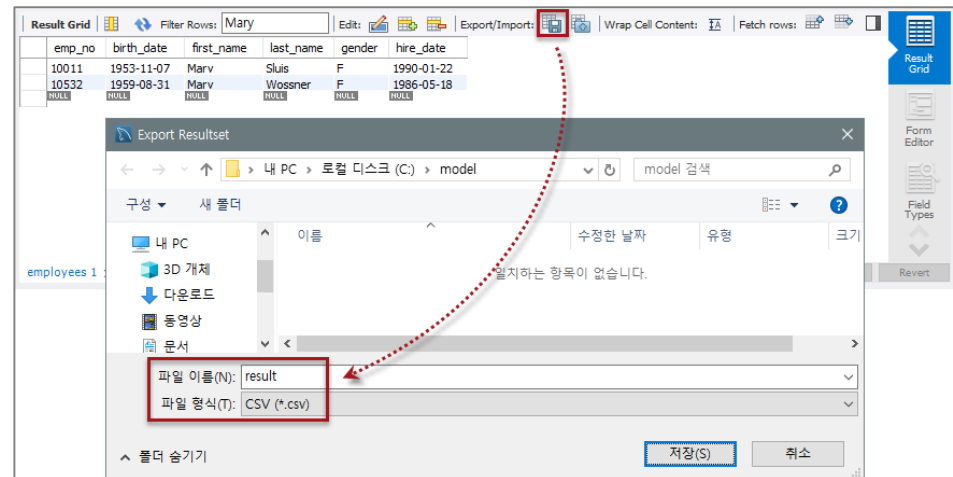
| emp_no | birth_date | first_name | last_name | gender | hire_date |
|--------|------------|------------|-------------|--------|------------|
| 10001 | 1953-09-02 | Geordi | Facello | M | 1986-06-26 |
| 10002 | 1964-06-02 | Bezalel | Simmel | F | 1985-11-21 |
| 10003 | 1959-12-03 | Parto | Bamford | M | 1986-08-28 |
| 10004 | 1954-05-01 | Chirstian | Koblick | M | 1986-12-01 |
| 10013 | 1955-01-21 | Kvoichi | Chiro | F | 1989-09-12 |
| 10014 | 1956-02-02 | Genin | Genin | M | 1989-06-03 |
| 10015 | 1959-08-19 | Guoxiana | Nootboom | M | 1987-07-02 |
| 10016 | 1961-05-02 | Kazuhiro | Capoelletti | M | 1995-01-27 |
| 10017 | 1958-07-06 | Cristinel | Roulours | F | 1993-08-03 |

9-2 Filter Rows: 옆의 텍스트 상자에 'Mary' 입력하고 [Enter] 누르기



| emp_no | birth_date | first_name | last_name | gender | hire_date |
|--------|------------|------------|-----------|--------|------------|
| 10011 | 1953-11-07 | Marv | Sluis | F | 1990-01-22 |
| 10532 | 1959-08-31 | Marv | Wossner | F | 1986-05-18 |

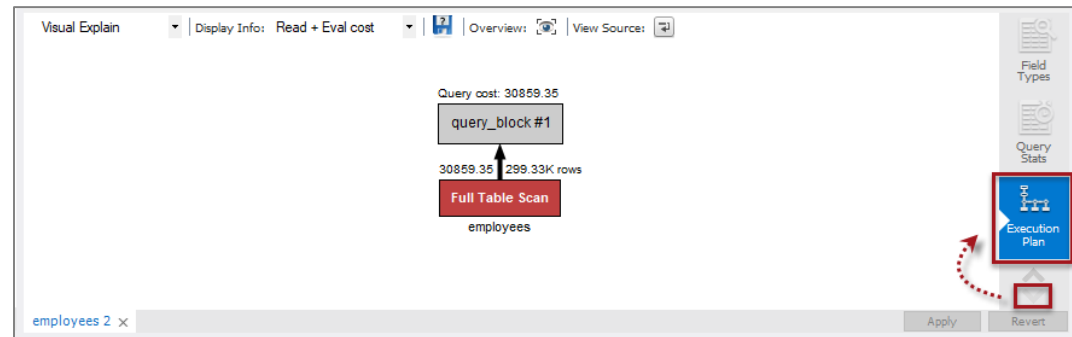
9-3 실행 결과 엑셀 파일로 저장



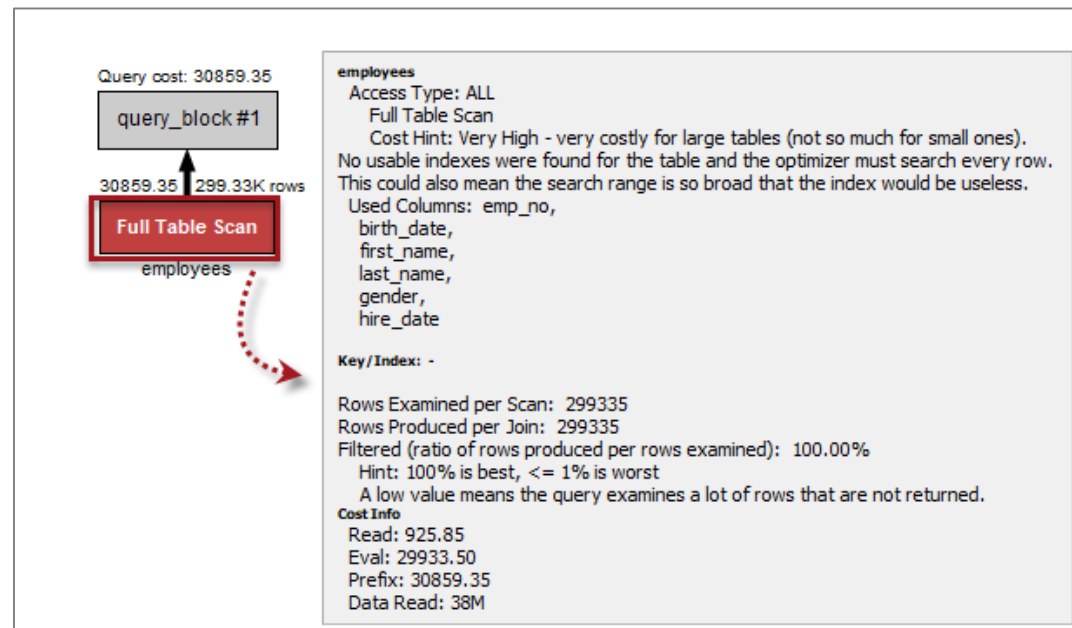
9-4 저장된 CSV 파일 확인

10 SQL 문의 실행 계획 확인하기

10-1 실행 계획(Execution Plan) 확인



10-2 실행 계획에 마우스 대기



10-3 Workbench 종료

2-1 MySQL 사용자 권한

- 일반적인 회사의 사용자와 권한의 예

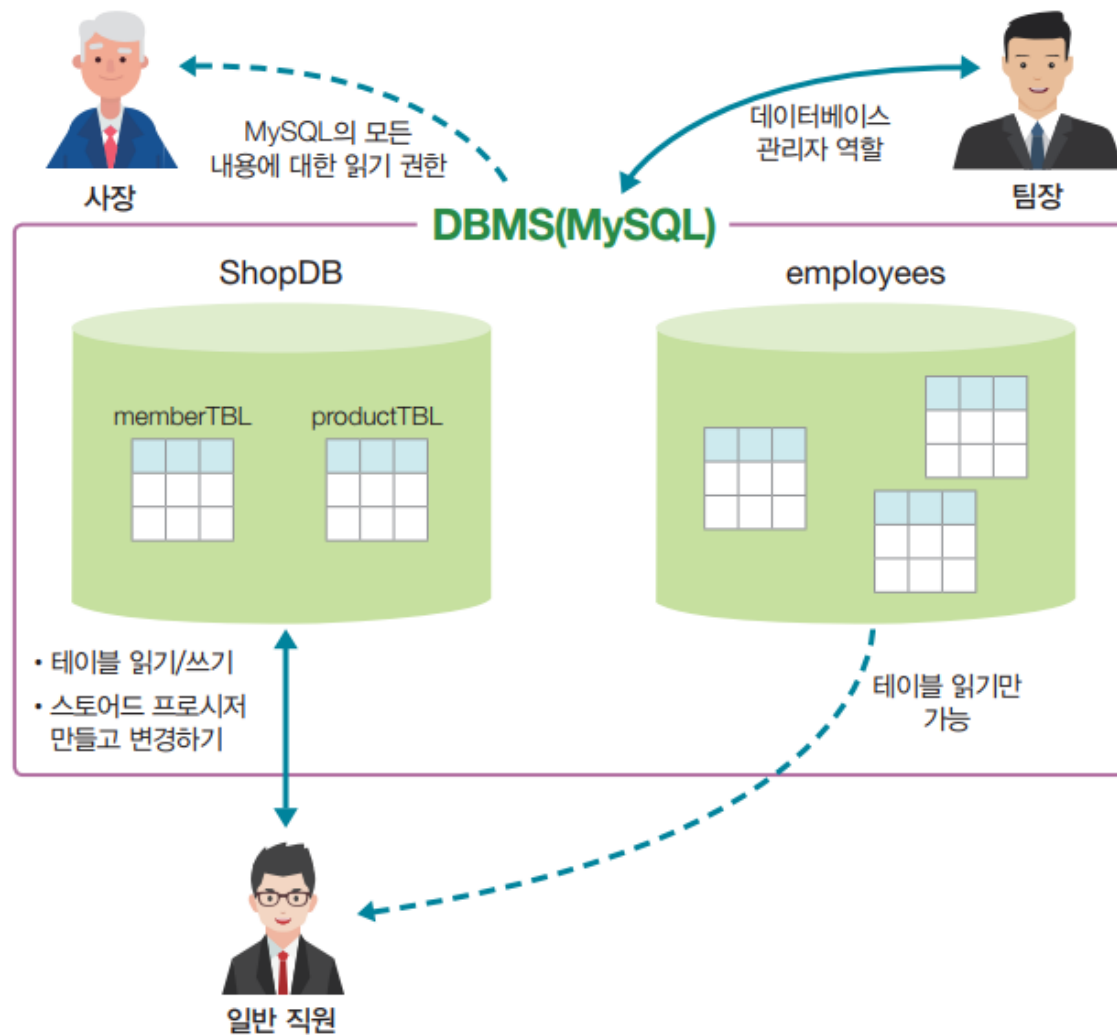


그림 4-40 사용자에게 부여된 다양한 권한

[실습 4-4] MySQL 사용자 생성하고 권한 부여하기

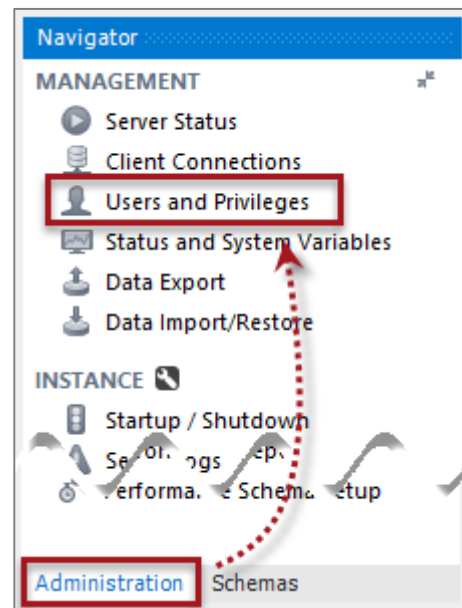
교재 134~141p 참고

1 MySQL 서버에 접속하기

1-1 root 사용자로 MySQL에 접속

2 팀장 생성하고 권한 부여하기

2-1 [Administration] 탭 선택 후 [Users and Privileges] 클릭



[실습 4-4] MySQL 사용자 생성하고 권한 부여하기

교재 134~141p 참고

2-2 [Login] 탭의 Login Name에 'director',
비밀번호는 기억하기 쉽게 'director'
를 입력 후 <Apply> 클릭

The screenshot shows the 'Administration - Users and Privileges' window for a local instance of MySQL. The 'Users and Privileges' section is active. On the left, the 'User Accounts' table lists several users, with 'director' highlighted. The 'Details for account newuser@%' window is open, showing the 'Login' tab. The 'Login Name' field is set to 'director'. The 'Authentication Type' is 'Standard'. The 'Limit to Hosts Matching' is '%'. The 'Password' and 'Confirm Password' fields are both set to '*****', with a red warning 'Weak password.' displayed below the password field. The 'Add Account' button is highlighted in red. The 'Apply' button is also highlighted in red. Red dotted arrows indicate the flow from the 'Add Account' button to the 'Login' tab, then to the 'Login Name' field, the 'Password' field, and finally to the 'Apply' button.

| User | From Host |
|------------------|-----------|
| mysql.infoschema | localhost |
| mysql.session | localhost |
| mysql.sys | localhost |
| root | localhost |
| director | % |

Details for account newuser@%

Login Account Limits Administrative Roles Schema Privileges

Login Name: director

Authentication Type: Standard

Limit to Hosts Matching: %

Password: *****

Confirm Password: *****

Weak password.

Expire Password

Add Account Delete Refresh Revert Apply

2-3 [Account Limits] 탭 클릭

The screenshot shows the 'Details for account director@%' window, with the 'Account Limits' tab selected. The 'Account Limits' section contains four rows of settings:

| Setting | Value | Description |
|-------------------------|-------|--|
| Max. Queries: | 0 | Number of queries the account can execute within one hour. |
| Max. Updates: | 0 | Number of updates the account can execute within one hour. |
| Max. Connections: | 0 | The number of times the account can connect to the server per hour. |
| Concurrent Connections: | 0 | The number of simultaneous connections to the server the account can have. |

Details for account director@%

Login Account Limits Administrative Roles Schema Privileges

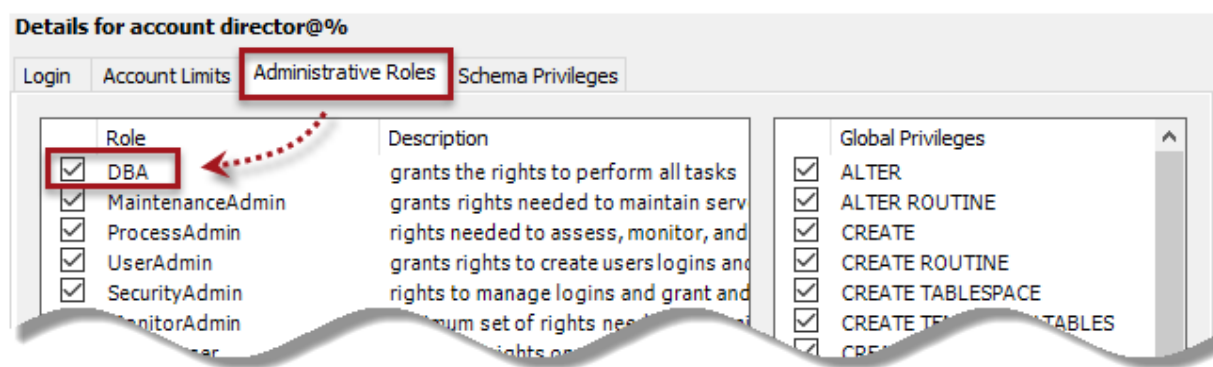
Max. Queries: 0

Max. Updates: 0

Max. Connections: 0

Concurrent Connections: 0

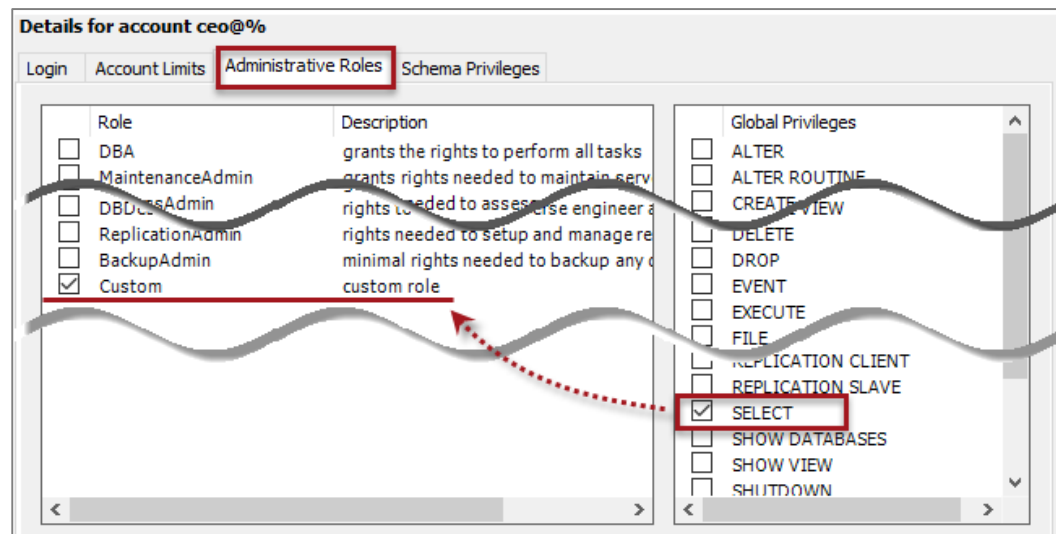
2-4 DBA에 체크 표시



3 사장 생성하고 권한 부여하기

3-1 [Users and Privileges] 창에서 왼쪽 아래의 <Add Account>를 클릭하고 [Login] 탭의 Login Name에 'ceo', 비밀번호는 기억하기 쉽게 'ceo' 입력

3-2 Global Privileges의 SELECT에 체크 표시



4 일반 직원 생성하고 권한 부여하기

4-1 [Users and Privileges] 창에서 왼쪽 아래의 <Add Account>를 클릭하고 [Login] 탭의 Login Name에 'staff', 비밀번호는 기억하기 쉽게 'staff' 입력

4-2 각 데이터베이스(스키마)에 대한 권한 부여

The screenshot shows the 'Details for account staff@%' window in MySQL. The 'Schema Privileges' tab is selected. A table lists the schema 'shopdb' with privileges: ALTER ROUTINE, CREATE ROUTINE, DELETE, INSERT, SELECT, UPDATE. Below the table, there are three sections: Object Rights, DDL Rights, and Other Rights. In the Object Rights section, SELECT, INSERT, UPDATE, and DELETE are checked. In the DDL Rights section, CREATE ROUTINE and ALTER ROUTINE are checked. In the Other Rights section, GRANT OPTION, CREATE TEMPORARY TABLES, and LOCK TABLES are unchecked. The 'Add Entry...' button is highlighted. A 'New Schema Privilege Definition' dialog is open, showing the 'Selected schema' as 'shopdb'.

Details for account staff@%

Schema Privileges

| Schema | Privileges |
|--------|---|
| shopdb | ALTER ROUTINE, CREATE ROUTINE, DELETE, INSERT, SELECT, UPDATE |

Schema and Host fields may use % and _ wildcards.
The server will match specific entries before wildcarded ones.

The user 'staff'@'%' will have the following access rights to the schema 'shopdb':

Object Rights

- ☒ SELECT
- ☒ INSERT
- ☒ UPDATE
- ☒ DELETE
- ☐ EXECUTE
- ☐ SHOW VIEW

DDL Rights

- ☐ CREATE
- ☐ ALTER
- ☐ REFERENCES
- ☐ INDEX
- ☐ CREATE VIEW
- ☒ CREATE ROUTINE
- ☒ ALTER ROUTINE
- ☐ EVENT
- ☐ DROP
- ☐ TRIGGER

Other Rights

- ☐ GRANT OPTION
- ☐ CREATE TEMPORARY TABLES
- ☐ LOCK TABLES

Buttons: Revoke All Privileges, Delete Entry, Add Entry...

New Schema Privilege Definition

Select the Schema for which the user 'staff' will have the privileges you want to define.

Schema

- ☐ All Schema (%)
- ☐ Schemas matching pattern:
- ☒ Selected schema: shopdb

This rule will apply to any schema name.

This rule will apply to schemas that match the given name or pattern. You may use _ and % as wildcards in a pattern. Escape these characters with \ in case you want their literal value.

Select a specific schema name for the rule to apply to.

4-3 employees 데이터베이스의 SELECT 권한만 부여

The image shows the MySQL 'Details for account staff@%' window with the 'Schema Privileges' tab selected. A table lists privileges for the 'employees' schema, with 'SELECT' highlighted. Below the table, the 'Object Rights' section has 'SELECT' checked. The 'Add Entry...' button is highlighted. A 'New Schema Privilege Definition' dialog box is open, showing 'Selected schema: employees' in the dropdown menu.

Details for account staff@%

Schema Privileges

| Schema | Privileges |
|-----------|---|
| shopdb | ALTER ROUTINE, CREATE ROUTINE, DELETE, INSERT, SELECT, UPDATE |
| employees | SELECT |

Schema and Host fields may use % and _ wildcards.
The server will match specific entries before wildcarded ones.

The user 'staff@%' will have the following access rights to the schema 'employees':

Object Rights:

- ☒ SELECT
- ☐ INSERT
- ☐ UPDATE
- ☐ DELETE
- ☐ EXECUTE
- ☐ SHOW VIEW

DDL Rights:

- ☐ CREATE
- ☐ ALTER
- ☐ REFERENCES
- ☐ INDEX
- ☐ CREATE VIEW
- ☐ CREATE ROUTINE
- ☐ ALTER ROUTINE
- ☐ EVENT
- ☐ DROP
- ☐ TRIGGER

Other Rights:

- ☐ GRANT OPTION
- ☐ CREATE TEMPORARY TABLES
- ☐ LOCK TABLES

Buttons: Revoke All Privileges, Delete Entry, Add Entry...

New Schema Privilege Definition

Select the Schema for which the user 'staff' will have the privileges you want to define.

Schema:

- ☐ All Schema (%)
- ☐ Schemas matching pattern:
- ☒ Selected schema: employees

This rule will apply to any schema name.

This rule will apply to schemas that match the given name or pattern. You may use _ and % as wildcards in a pattern. Escape these characters with \ in case you want their literal value.

Select a specific schema name for the rule to apply to.

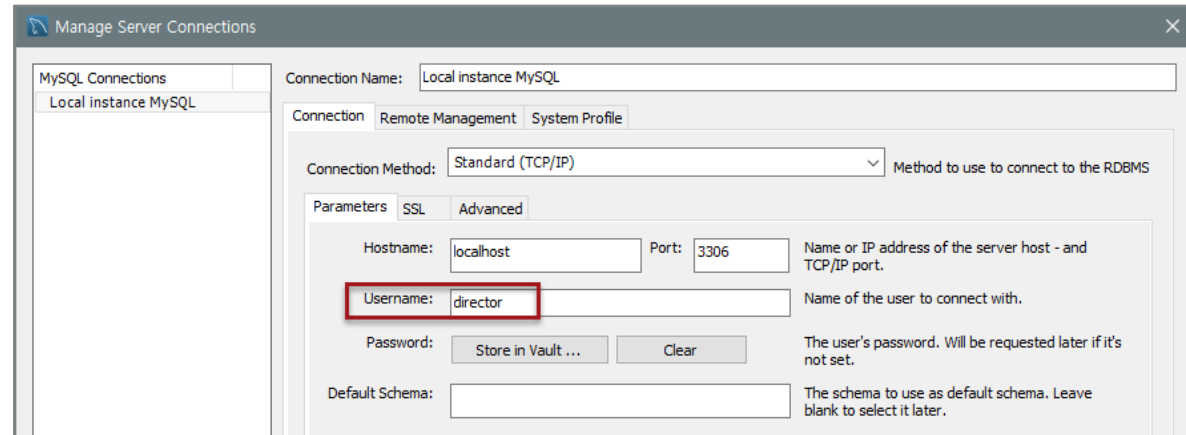
[실습 4-4] MySQL 사용자 생성하고 권한 부여하기

교재 134~141p 참고

5 팀장으로 접속하여 권한 확인하기

5-1 Workbench 실행 후 'Local instance MySQL'에서 마우스 오른쪽 버튼을 클릭하여 [Edit Connection] 선택

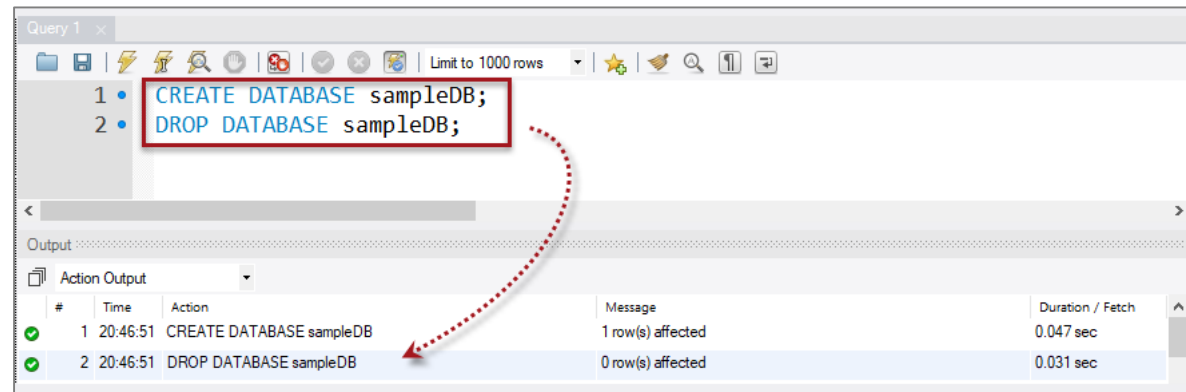
5-2 Username을 root에서
director로 변경



5-3 'Local instance MySQL'을 클릭하여 접속(비밀번호에 'director' 입력)

5-4 다음 SQL 문을 입력하여 데이터베이스를 하나 생성한 후 삭제

```
CREATE DATABASE sampleDB;  
DROP DATABASE sampleDB;
```



5-5 접속 종료

[실습 4-4] MySQL 사용자 생성하고 권한 부여하기

교재 134~141p 참고

6 사장으로 접속하여 권한 확인하기

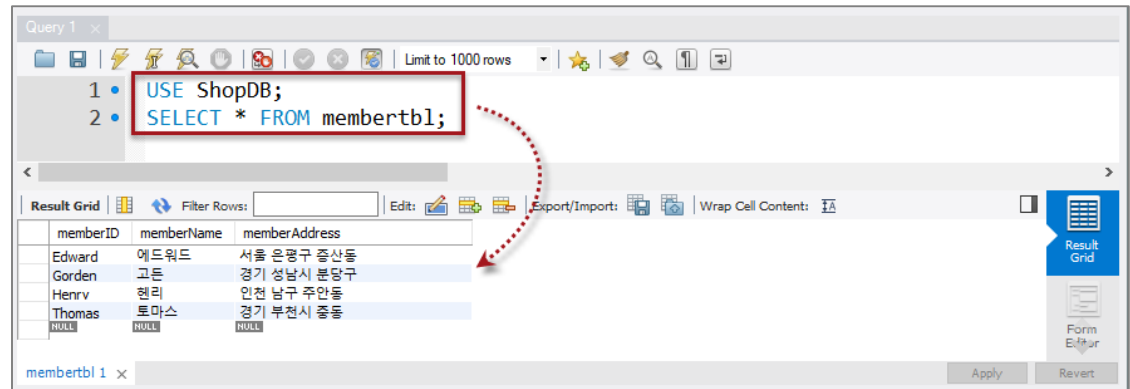
6-1 사장(ceo)으로 접속

6-2 Username을 ceo로 변경

6-3 'Local instance MySQL'을 클릭하여 접속(비밀번호에 'ceo' 입력)

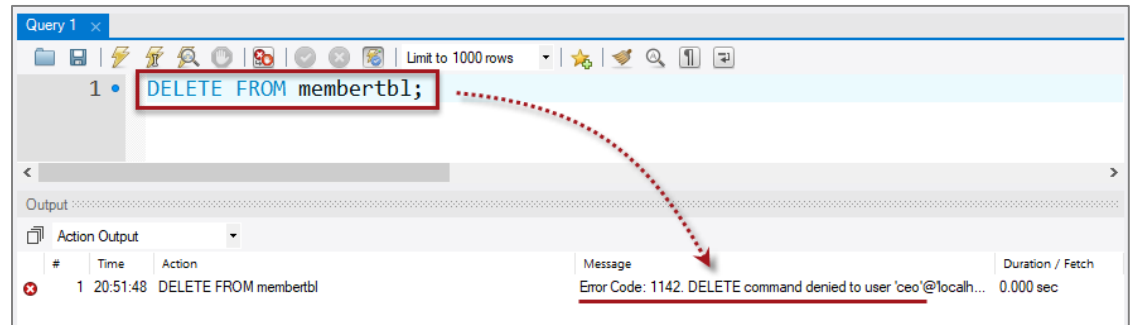
6-4 다음 SQL 문을 입력하여 제대로 읽기가 되는지 확인

```
USE ShopDB;  
SELECT * FROM membertbl;
```



6-5 다음 SQL 문을 입력하여 데이터가 삭제되는지 확인(실행되지 않아야 함)

```
DELETE FROM membertbl;
```



6-6 접속 종료

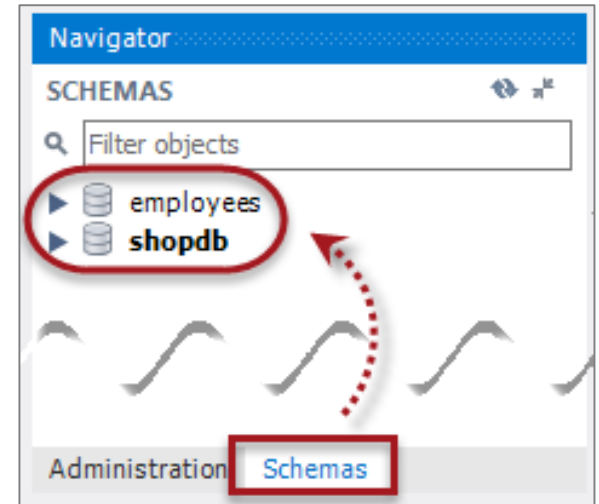
[실습 4-4] MySQL 사용자 생성하고 권한 부여하기

교재 134~141p 참고

7 일반 직원으로 접속하여 권한 확인하기

7-1 일반 직원(staff)으로 접속

7-2 [Navigator]의 [Schemas] 탭에 데이터베이스 확인



7-3 다음 SQL 문을 입력하여 실행되는지 확인

```
USE ShopDB;  
SELECT * FROM memberTBL;  
DELETE FROM memberTBL WHERE memberID = 'Gorden';
```

7-4 다음 SQL 문을 입력하여 테이블이 삭제되는지 확인(삭제되지 않아야 함)

```
DROP TABLE memberTBL;
```

[실습 4-4] MySQL 사용자 생성하고 권한 부여하기

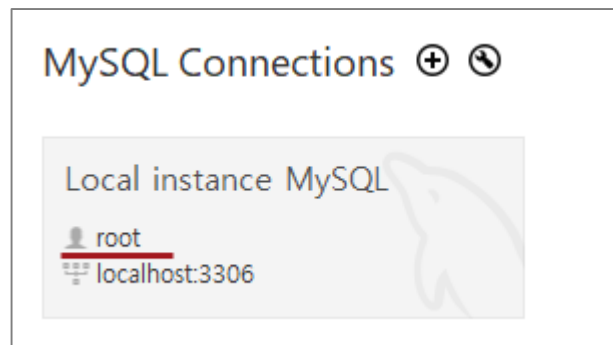
교재 134~141p 참고

7-5 다음 SQL 문을 입력하여 employees 데이터베이스 조회 확인

```
USE employees;  
SELECT * FROM employees;
```

7-6 접속 종료

7-7 'Local instance MySQL'을 원래의 root로 변경하고
Workbench 종료



3-1 백업과 복원의 개요

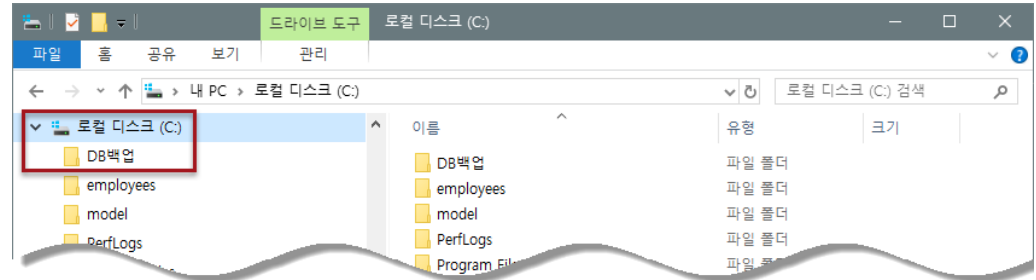
- 백업과 복원은 데이터베이스 관리자(DBA)가 해야 할 가장 중요한 일
- 백업
 - 데이터 베이스를 다른 매체에 보관하는 작업
- 복원
 - 데이터베이스에 문제가 발생했을 때 다른 매체 에 백업된 데이터를 이용하여 원상태로 돌려놓는 작업

[실습 4-5] 쇼핑몰 데이터베이스 백업 후 복원하기

교재 142~146p 참고

1 백업 준비하기

- 1-1 C:\WDB백업\ 폴더를 만들어
이 폴더를 별도의 디스크라고 가정

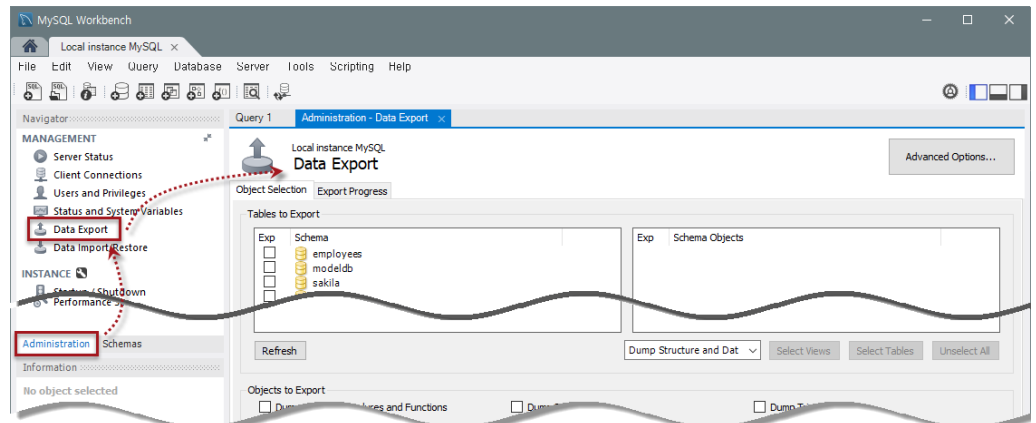


- 1-2 root 사용자(비밀번호는 1234)로 접속
- 1-3 다음 SQL 문을 입력하여 데이터 확인

```
USE ShopDB;  
SELECT * FROM productTBL;
```

2 쇼핑몰 데이터베이스(ShopDB) 백업하기

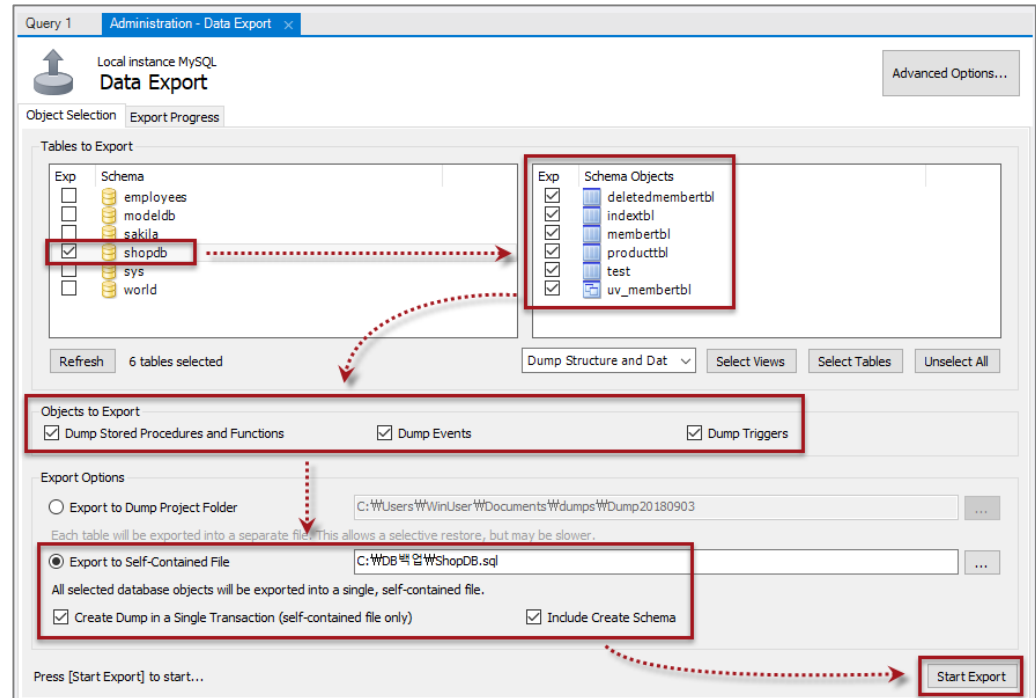
- 2-1 [Data Export] 클릭



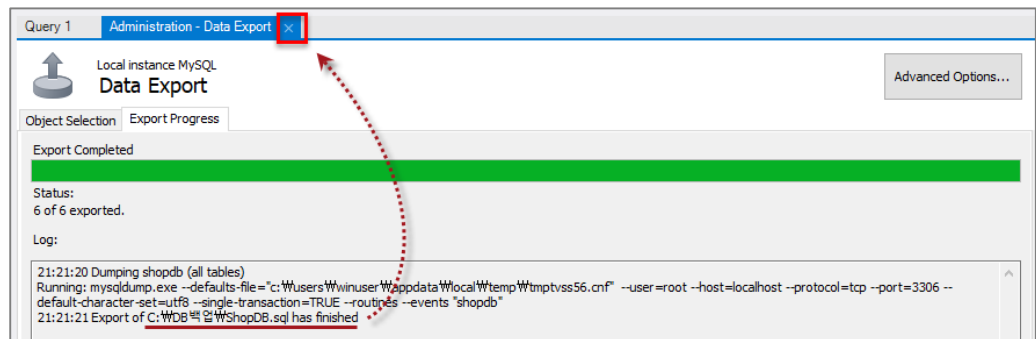
[실습 4-5] 쇼핑몰 데이터베이스 백업 후 복원하기

교재 142~146p 참고

2-2 ShopDB의 모든 내용을 백업하는
설정을 마치고 <Start Export> 클릭

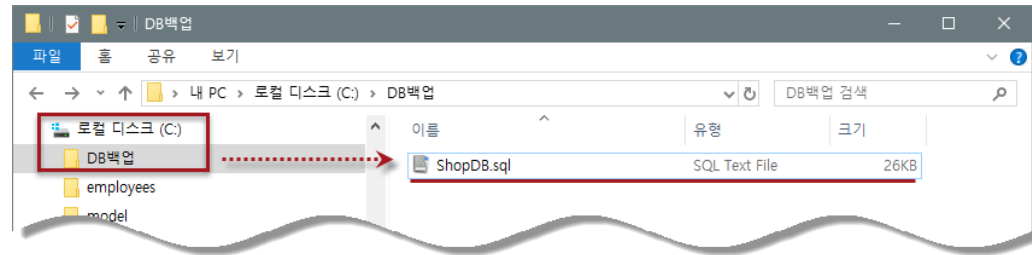


2-3 백업이 종료되면 닫기(x) 버튼 클릭



2-4 C:\WDB백업\ 폴더 확인

(백업된 파일 ShopDB.sql을 볼 수 있음)



3 고의로 모든 데이터 삭제하기

3-1 productTBL의 모든 데이터를 삭제

```
DELETE FROM productTBL;
```

3-2 다음 SQL 문을 실행(당연히 0개의 데이터가 조회됨)

```
SELECT * FROM productTBL;
```

4 복원하기

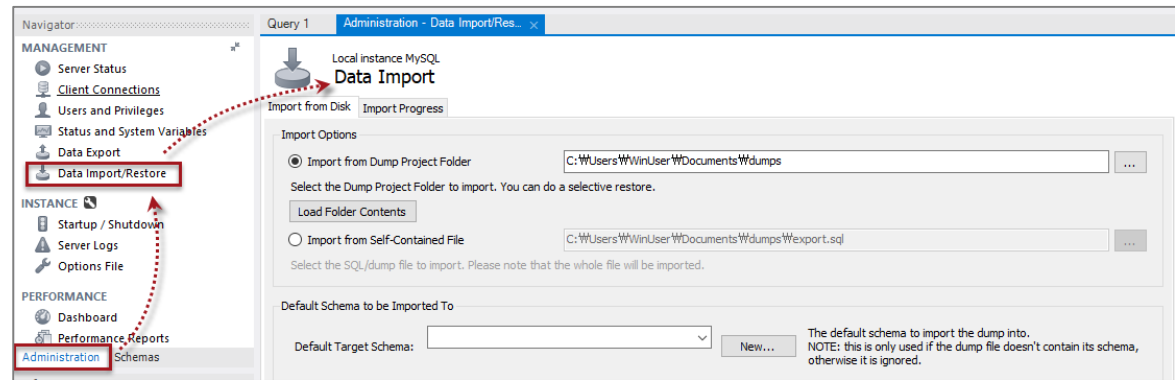
4-1 다음 SQL 문을 입력하여 실행

```
USE sys; -- 일단 다른 DB를 선택
```

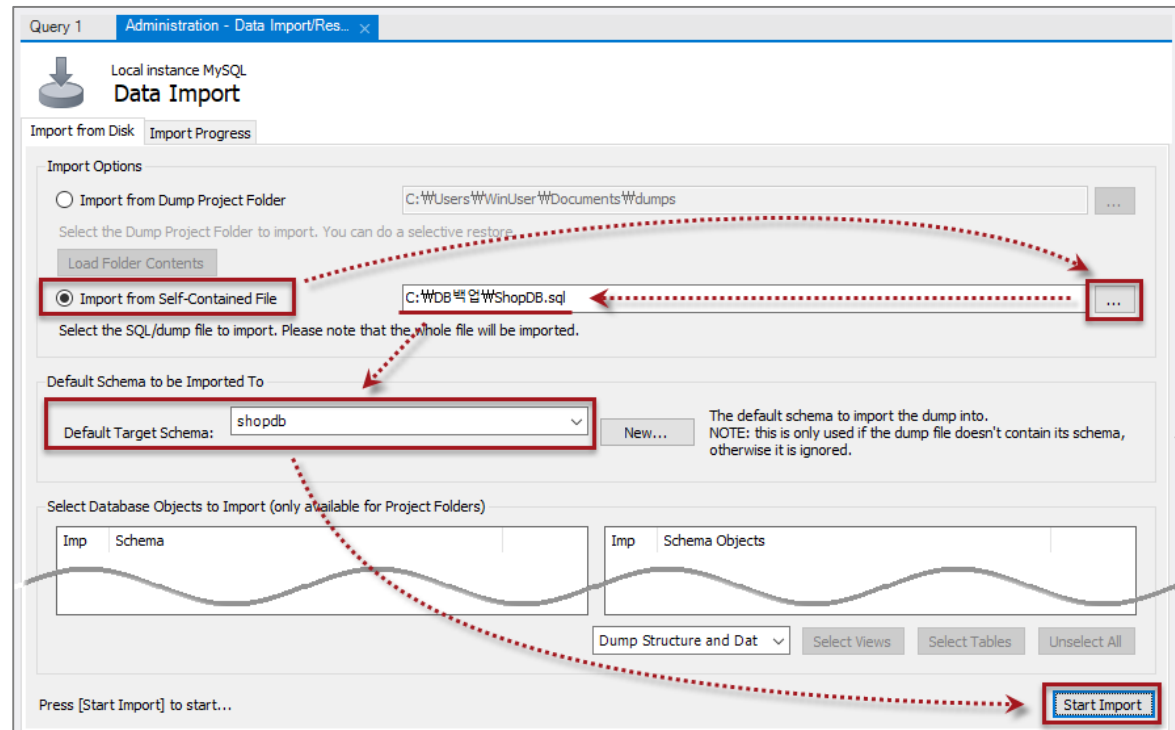

[실습 4-5] 쇼핑몰 데이터베이스 백업 후 복원하기

교재 142~146p 참고

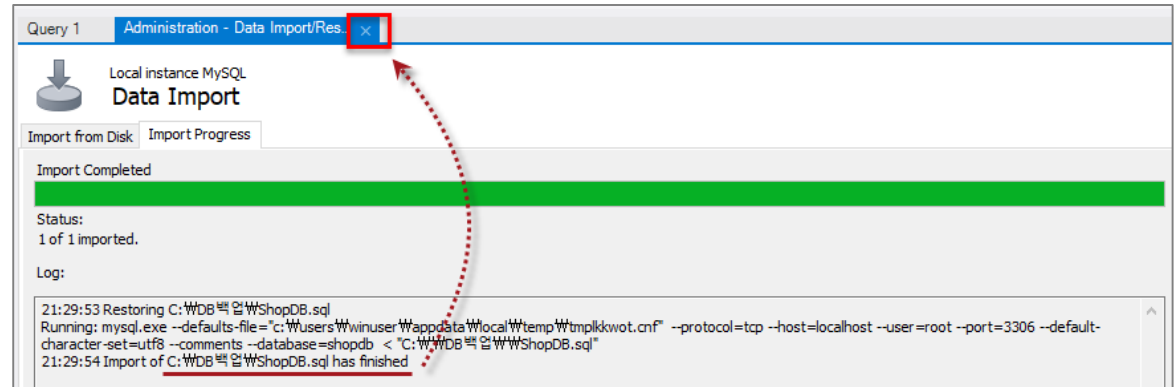
4-2 [Data Import/Restore] 클릭



4-3 모든 설정 후 <Start Import> 클릭



4-4 제대로 복원된 것 확인 후
[Data Import] 창 닫기



4-5 다음 SQL 문을 입력하여 데이터가 잘 복원되었는지 확인

```
USE ShopDB;  
SELECT * FROM productTBL;
```