

# DB 템프로젝트 과제-최종

2014180048 김민지  
데이터베이스 C반

# 텀프로젝트 수행 과정

- 게임선정

- 'Deemo' 를 선정한 이유 : Primary Key를 설정하기 수월하고, 평소 즐겨 하던 게임이기 때문입니다.

- 데이터베이스 요구사항 분석서 작성

- ER 다이어그램 작성

- 데이터베이스 요구사항 분석서를 바탕으로 작성했습니다.
  - '설정', '조정' 항목이 필요 없는 항목인 것 같아 삭제했습니다.

- 관계형데이터 모델 작성

- ER 다이어그램을 바탕으로 작성했습니다.

- MySQL Test 및 테이블 작성, 데이터 입력

- Test 결과 잘 실행되는 것을 확인한 후,  
관계형데이터 모델을 바탕으로 테이블을 작성하고, 데이터를 입력했습니다.

- 정규화

- 제 1, 2, 3 정규형을 통해 수정, 관리했습니다.

- MySQL 쿼리문 작성

- MySQL이 서버와 연결되어 있지 않다는 에러를 해결하기 위해 연결 정보를 수정하여 오류를 해결했습니다.
  - 쿼리문을 작성했으나 쿼리문을 실행하면 column만 출력이 되었고, 콘솔을 실행하여 데이터를 다시 입력하여 오류를 해결했습니다.

# 선정 게임 : Deemo by Rayark

---

## \* 데이터베이스 요구사항 분석서

### - 사용자

- 계정 연동을 통해 게임을 플레이할 수 있다.
- 사용자는 콜렉션을 플레이한다.
- 사용자는 사용자번호화 닉네임을 가지고 레벨이 표시된다.

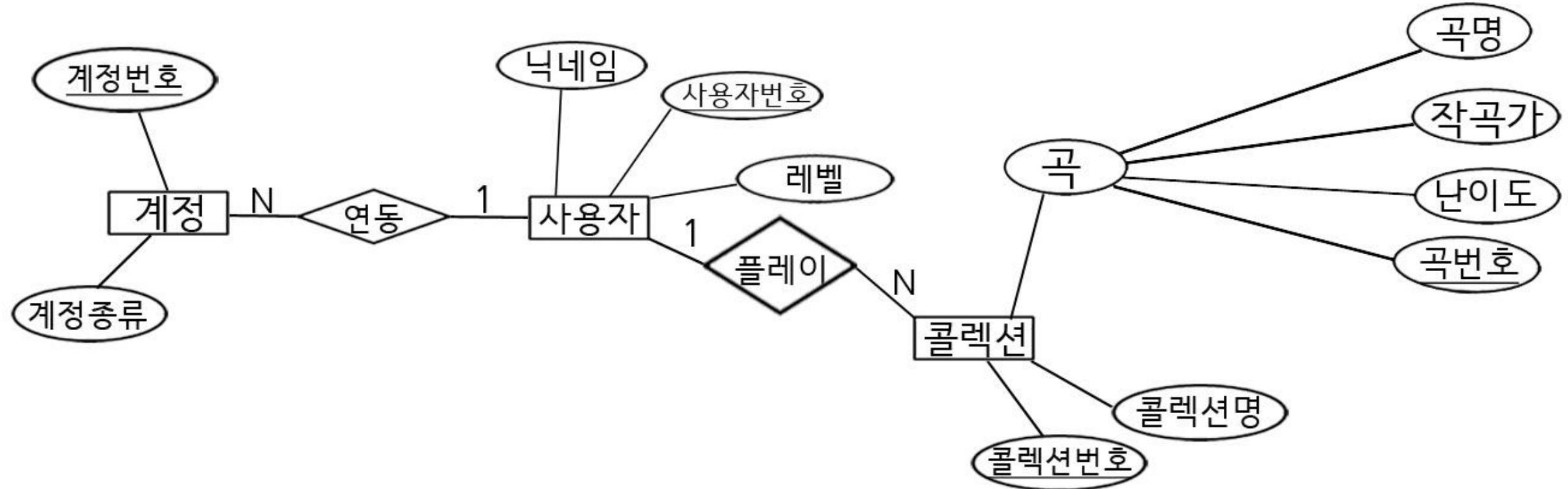
### - 콜렉션

- 콜렉션에는 콜렉션 명이 표시되어지고 각 콜렉션에는 각각 다른 여러 개의 곡들이 존재한다.
- 곡에는 곡 명과 작곡가, 난이도, 곡 번호가 표시되어진다.

### - 계정

- 계정에는 계정번호와 계정종류가 표시된다.
  - 사용자가 연동할 수 있는 계정에는 facebook과 google이 있다.
-

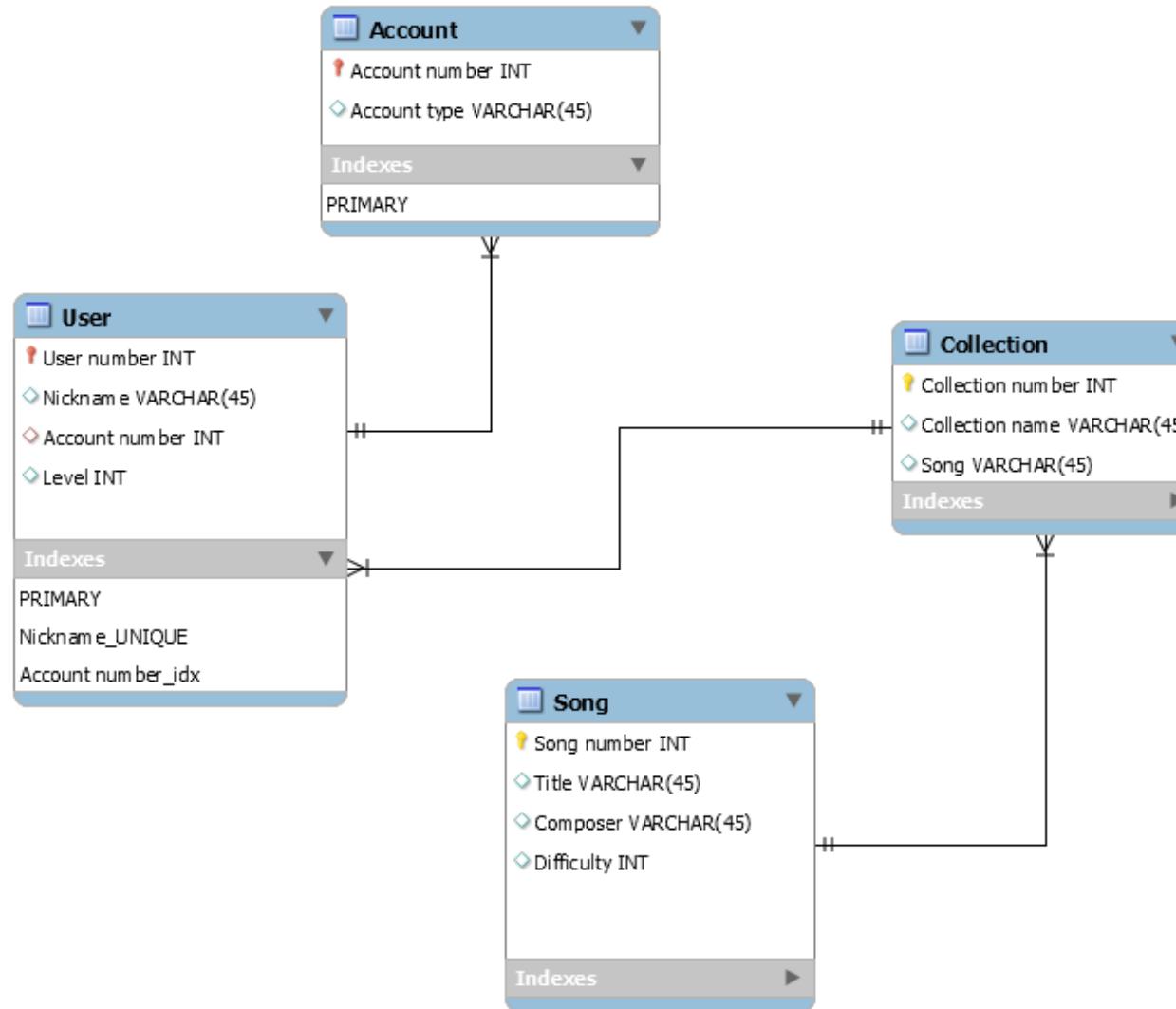
# Deemo ER 다이어그램



# Deemo 관계형데이터 모델



# Deemo My SQL 적용 - 1



# Deemo My SQL 적용 - 2

song	song number	title	composer	difficulty
	30001	Quartet1	nonamed1	5
	30002	Quartet2	nonamed2	6
	30003	Quartet3	nonamed3	8
	30004	Quartet4	nonamed4	2
	30005	Quartet5	nonamed5	3
	30006	piano1	nonamed6	9
	30007	piano2	nonamed7	1
	30008	piano3	nonamed8	10
	30009	piano4	nonamed9	4
	30010	piano5	nonamed10	7

account	accountnum	accounttype	user_usernum
	1001	facebook	20141
	1002	facebook	20142
	1003	facebook	20143
	1004	facebook	20144
	1005	facebook	20145
	2001	google	20146
	2002	google	20147
	2003	google	20148
	2004	google	20149
	2005	google	20150

collection	collectionnum	collectionnam	song	user_usernum
	2010001	Ver. 1	Quartet1	20141
	2010002	Ver. 2	Quartet2	20142
	2010003	Ver. 3	Quartet3	20143
	2010004	Ver. 4	Quartet4	20144
	2010005	Ver. 5	Quartet5	20145
	2010006	Ver. 6	piano1	20146
	2010007	Ver. 7	piano2	20147
	2010008	Ver. 8	piano3	20148
	2010009	Ver. 9	piano4	20149
	2010010	Ver. 10	piano5	20150

user	usernum	nickname	accountnum	lev
	20141	red	1001	1
	20142	orange	1002	21
	20143	yellow	1003	9
	20144	green	1004	15
	20145	blue	1005	8
	20146	purple	2001	10
	20147	black	2002	17
	20148	white	2003	32
	20149	gray	2004	2
	20150	ink	2005	11

# Deemo MySQL 질의문 - 1

The screenshot shows the MySQL Workbench interface. In the top navigation bar, there are tabs for Local instance MySQL57 (my...), Local instance MySQL57 (new\_sc...), MySQL Model\* (database 1.mwb), and EER Diagram. Below the tabs is a menu bar with File, Edit, View, Query, Database, Server, Tools, Scripting, and Help. The main area has a toolbar with icons for SQL, DDL, Data, Scripts, and other functions. On the left, there's a Navigator pane with sections for MANAGEMENT (Server Status, Client Connections, Users and Privileges, Status and System Variables, Data Export, Data Import/Restore), INSTANCE (Startup / Shutdown, Server Logs, Options File), and PERFORMANCE (Dashboard, Performance Reports, Performance Schema Setup). The central part of the screen shows a SQL editor window titled '22\*' containing the following code:

```
89 SET SQL_MODE=@OLD_SQL_MODE;
90 SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS;
91 SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS;
92
93
94 select usernum
95 from user;
96
97 select nickname, lev
98 from user
99 where lev >=10 and 20>=lev;
100
101
```

The code at line 97 is highlighted with a red box. Below the editor is a 'Result Grid' showing the output of the query:

nickname	lev
green	15
purple	10
black	17
pink	11

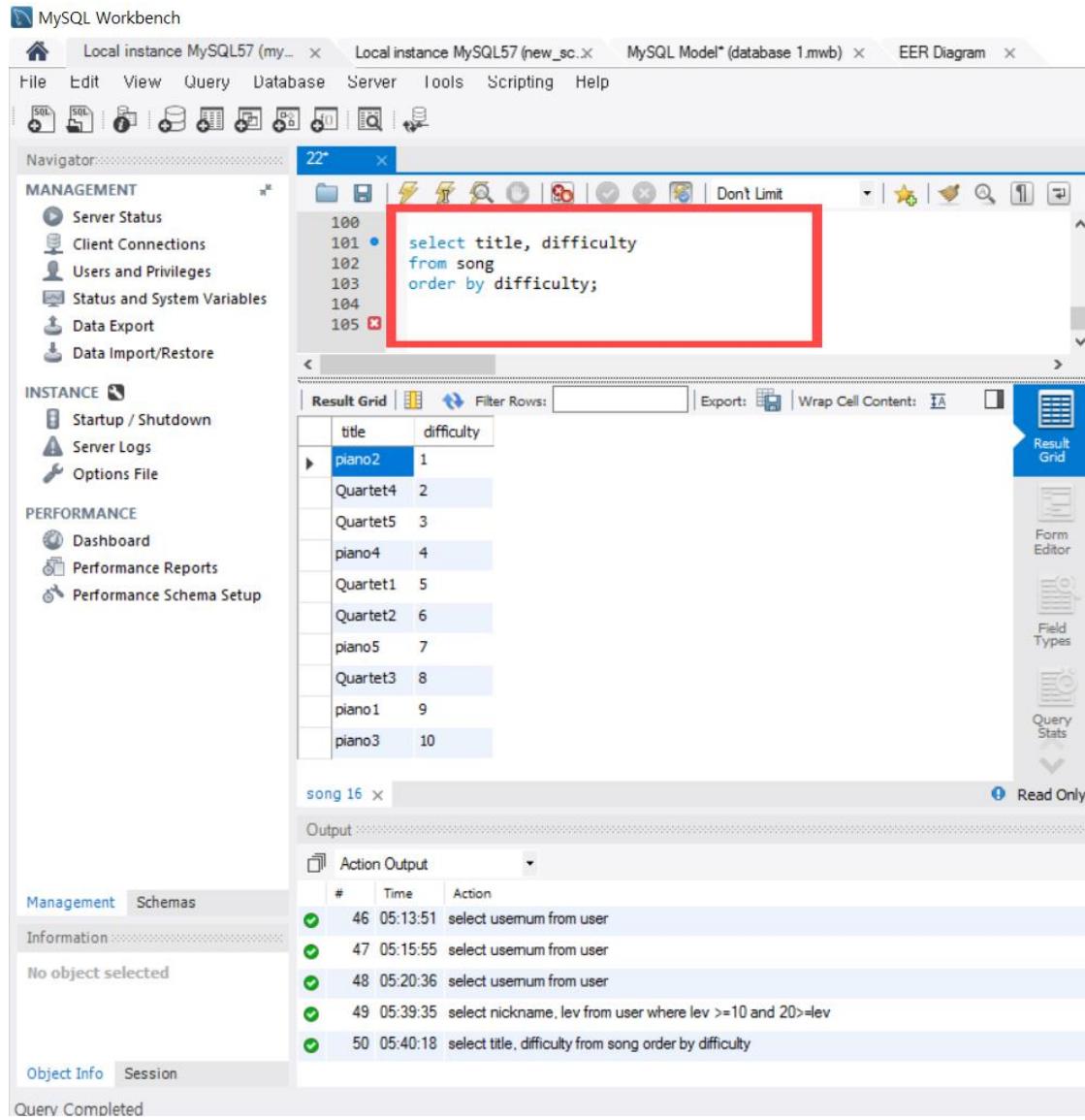
At the bottom, there's a 'user 15 x' tab and an 'Output' section with an 'Action Output' table:

#	Time	Action
45	05:13:30	select usenum from user
46	05:13:51	select usenum from user
47	05:15:55	select usenum from user
48	05:20:36	select usenum from user
49	05:39:35	select nickname, lev from user where lev >=10 and 20>=lev;

\* 레벨이 10부터 20사이인  
사용자들의 닉네임과 레벨을 출력.

select nickname, lev  
from user  
where lev >=10 and 20>=lev;

# Deemo MySQL 질문 - 2



\* 난이도가 낮은 곡부터 높은 곡 순으로 제목과 난이도를 나열.

select title, difficulty  
from song  
order by difficulty;

# Deemo MySQL 질의문 - 3

The screenshot shows the MySQL Workbench interface. In the top navigation bar, there are tabs for Local instance MySQL57 (my...), Local instance MySQL57 (new\_sc...), MySQL Model\* (database 1.mwb), and EER Diagram. The main window has a toolbar with various icons. On the left, there's a Navigator pane with sections for MANAGEMENT (Server Status, Client Connections, Users and Privileges, Status and System Variables, Data Export, Data Import/Restore), INSTANCE (Startup / Shutdown, Server Logs, Options File), and PERFORMANCE (Dashboard, Performance Reports, Performance Schema Setup). The central area contains a SQL editor with the following query:

```
111  
112  
113  
114 • select usernum, nickname, accountnum, lev  
from user  
where lev between 6 and 20;  
115  
116 ✘
```

The Result Grid shows the following data:

usernum	nickname	accountnum	lev
20143	yellow	1003	9
20144	green	1004	15
20145	blue	1005	8
20146	purple	2001	10
20147	black	2002	17
20150	pink	2005	11

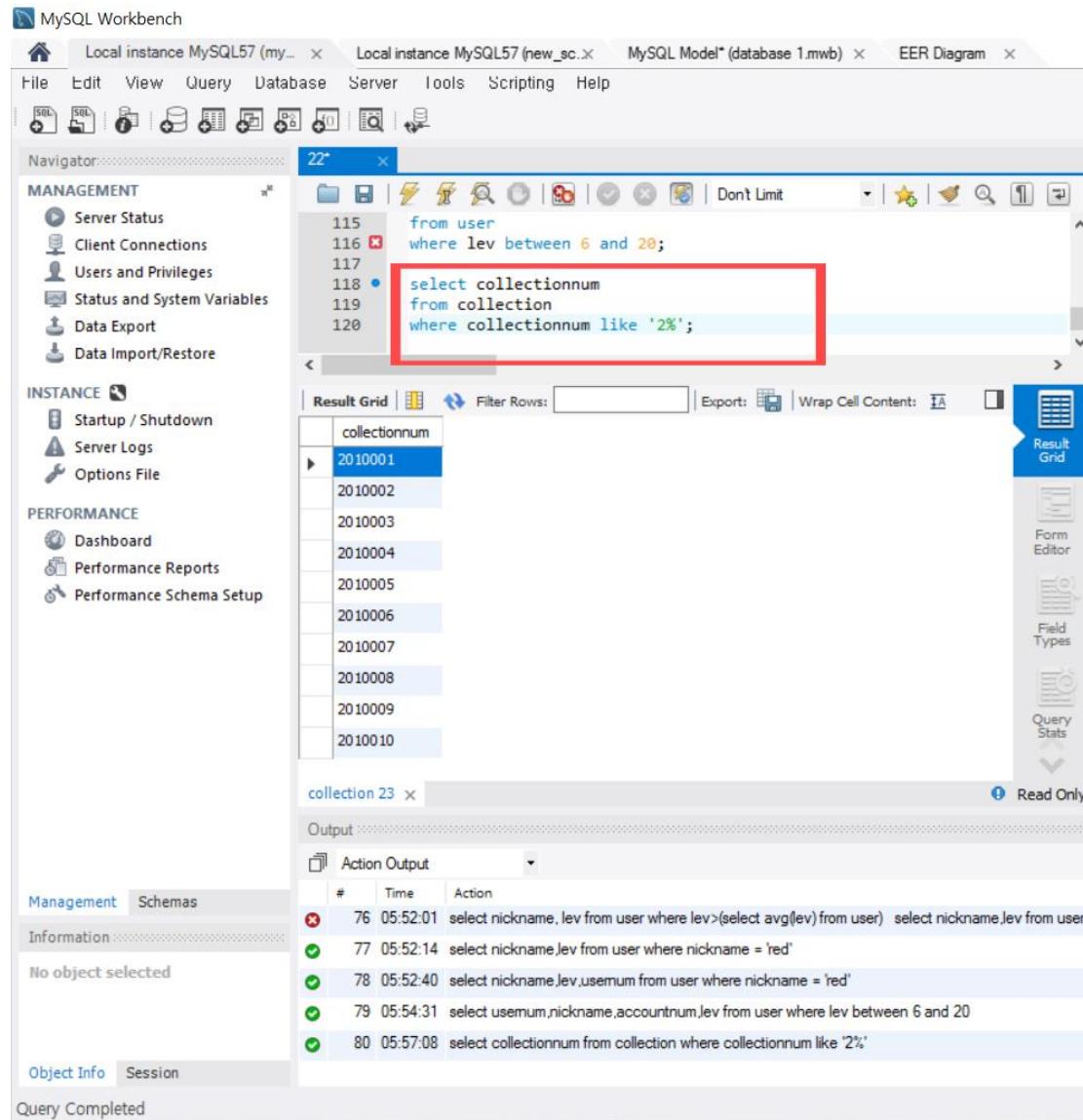
Below the result grid, there's a log titled "user 22" showing several actions:

#	Time	Action
75	05:51:34	select nickname, lev from user where lev > (select avg(lev) from user) order by lev desc
76	05:52:01	select nickname, lev from user where lev > (select avg(lev) from user) select nickname,lev from user \n
77	05:52:14	select nickname,lev from user where nickname = 'red'
78	05:52:40	select nickname,lev,useum from user where nickname = 'red'
79	05:54:31	select useum,nickname,accountnum,lev from user where lev between 6 and 20;

\* 레벨이 6에서 20사이인  
사용자의 사용자번호와 닉네임,  
계정번호, 레벨을 출력.

select usernum, nickname,  
accountnum, lev  
from user  
where lev between 6 and 20;

# Deemo MySQL 질의문 - 4



\* 콜렉션번호가 2로 시작하는 콜렉션의 콜렉션번호를 출력.

select collectionnum  
from collection  
where collectionnum like '2%';

# Deemo MySQL 질문 - 5

The screenshot shows the MySQL Workbench interface. In the top navigation bar, there are tabs for Local instance MySQL57, Local instance MySQL57 (new\_sc), MySQL Model (database 1.mwb), and EER Diagram. The main area has a toolbar with icons for SQL, DDL, Data, Scripts, and Help. On the left, the Navigator pane shows sections for MANAGEMENT (Server Status, Client Connections, Users and Privileges, Status and System Variables, Data Export, Data Import/Restore), INSTANCE (Startup / Shutdown, Server Logs, Options File), and PERFORMANCE (Dashboard, Performance Reports, Performance Schema Setup). The central part of the screen contains a query editor window titled '22\*' with the following SQL code:

```
from collection  
where collectionnum like '2%';  
  
select composer, title  
from song  
where composer like '__n%';
```

The code from line 122 to 124 is highlighted with a red box. Below the code, the 'Result Grid' shows the following data:

composer	title
nonamed1	Quartet1
nonamed2	Quartet2
nonamed3	Quartet3
nonamed4	Quartet4
nonamed5	Quartet5
nonamed6	piano1
nonamed7	piano2
nonamed8	piano3
nonamed9	piano4
nonamed10	piano5

At the bottom, the 'Output' pane shows the following log entries:

#	Time	Action
81	05:58:24	select title from song where title like '__a%'
82	05:58:37	select title from song where title like '__r%'
83	05:59:15	select title, songnum, composer from song where title like '__r%'
84	05:59:21	select title, songnum, composor from song where title like '__r%'
85	06:00:09	select composer, title from song where composer like '__n%'

\* 이름의 세 번째 철자가  
'n'인 작곡가의 이름과  
그 작곡가의 곡 제목을 출력.

select composer, title  
from song  
where composer like '\_\_n%';

# 학기 중, 과제 수행 중 느낀 점

\*

MySQL을 설치하고서 무작정 데이터를 입력, 수정하려 하니 막막했습니다.

대출한 책과 구글링을 통해서 쿼리문 작성에 대한 요령을 터득하고,

PPT 강의안을 보면서 쿼리문 작성을 해 나갔습니다.

계속되는 오류로 해결하기까지 장장 6시간이 걸렸지만, 데이터가 조회되는 화면을 보고쾌감을 느꼈습니다.

\*

데이터베이스 수업을 한 학기 동안 수강하면서 가장 기억에 남는 것은,

어려운 시험과 복잡한 MySQL 사용법인 것 같습니다.

그러나 MySQL로 쿼리문을 작성해보는 경험은 제게 자신감을 심어주었습니다.

항상 열성적인 강의를 해주시는 교수님께서는

어려운 내용도 흥미롭게 느껴지도록 설명을 잘 해주셔서 강의 시간이 즐거웠던 것 같습니다.

내용은 어려웠지만 말입니다.

데이터베이스 공부를 따로 해보고 싶다는 생각이 듭니다. 앞으로 남은 시험도 잘 마치고 싶습니다.

한 학기 동안 정말 감사했습니다.

**감사합니다!**