

Hello,



KDT 웹 개발자 양성 프로젝트

5기!

with





개발자	오라클
발표일	1995년 5월 23일
라이선스	GPL v2 (커뮤니티 에디션) 상용 라이선스 (표준, 엔터프라이즈, 클러스터)
링크	   



## 매직스테이션 II PLUS

- 166~100MHz 펜티엄®칩 탑재
- 28,800~14,400bps 고속팩스모뎀 장착
- 8~4배속 CD롬 내장
- 매직랜드로 그림, 음성안내 지원
- 음성명령 및 스케줄러 기능 제공

- SPC8260P-MM21QH (펜티엄칩-100, 8MB, 1.27GB, 14,400bps, 4배속 CD롬) W 2,618,000
- SPC8260P-AM21QH (펜티엄칩-100, 16MB, 1.27GB, 14,400bps, 4배속 CD롬) W 2,816,000
- SPC8260P-MM21QJ (펜티엄칩-120, 16MB, 1.27GB, 14,400bps, 4배속 CD롬) W 2,959,000
- SPC8260P-AM21GN (펜티엄칩-133, 16MB, 1.6GB, 14,400bps, 6배속 CD롬) W 3,234,000
- SPC8260P-MM21GV (펜티엄칩-150, 16MB, 1.6GB, 28,800bps, 6배속 CD롬) W 3,652,000
- SPC8260P-AM21GT (펜티엄칩-166, 16MB, 1.6GB, 28,800bps, 6배속 CD롬) W 4,081,000
- SPC8260P-DM22GT (펜티엄칩-166, 32MB, 2.1GB, 28,800bps, 8배속 CD롬) W 4,719,000

### 자매품 매직스테이션X

- SPC8220P-MW412H (펜티엄칩-100, 8MB, 1.27GB, 4배속 CD롬) W 2,079,000
- SPC8220P-MW412J (펜티엄칩-120, 8MB, 1.27GB, 4배속 CD롬) W 2,277,000
- SPC8220P-MW612N (펜티엄칩-133, 8MB, 1.27GB, 6배속 CD롬) W 2,453,000





[https://www.w3schools.com/sql/trysql.asp?filename=trysql\\_select\\_all](https://www.w3schools.com/sql/trysql.asp?filename=trysql_select_all)

---

### SQL Statement:

```
SELECT * FROM Customers;
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Run SQL »

### Result:

Click "**Run SQL**" to execute the SQL statement above.

W3Schools has created an SQL database in your browser.

The menu to the right displays the database, and will reflect any changes.

Feel free to experiment with any SQL statement.

You can restore the database at any time.

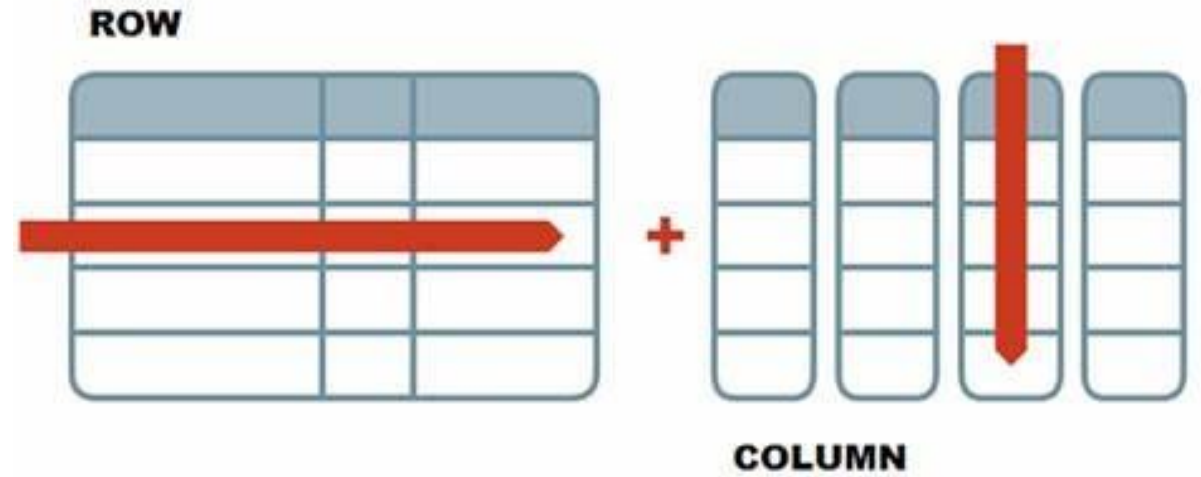




Diagram illustrating a table structure with labels and arrows:

- 행(row)**: Points to the rows of the table.
- 열 이름**: Points to the header row.
- 열(column)**: Points to the columns of the table.

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





# SELECT



# SELECT, DB 에서 원하는 데이터 가져오기



- 갓춰진 DB 에서 데이터를 뽑아 올 때 사용하는 구문 입니다!
- 일단
- `SELECT * FROM Customer;`



CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Maria Anders	Obere Str. 57	Berlin	12209	Germany
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Avda. de la Constitución 2222	México D.F.	05021	Mexico
3	Antonio Moreno Taquería	Antonio Moreno	Mataderos 2312	México D.F.	05023	Mexico
4	Around the Horn	Thomas Hardy	120 Hanover Sq.	London	WA1 1DP	UK
5	Berglunds snabbköp	Christina Berglund	Berguvsvägen 8	Luleå	S-958 22	Sweden
6	Blauer See Delikatessen	Hanna Moos	Forsterstr. 57	Mannheim	68306	Germany
7	Blondel père et fils	Frédérique Citeaux	24, place Kléber	Strasbourg	67000	France
8	Bólido Comidas preparadas	Martín Sommer	C/ Araquil, 67	Madrid	28023	Spain
9	Bon app'	Laurence Lebihans	12, rue des Bouchers	Marseille	13008	France
10	Bottom-Dollar Marketse	Elizabeth Lincoln	23 Tsawassen Blvd.	Tsawassen	T2F 8M4	Canada
11	B's Beverages	Victoria Ashworth	Fauntleroy Circus	London	EC2 5NT	UK
12	Cactus Comidas para llevar	Patricio Simpson	Cerrito 333	Buenos Aires	1010	Argentina
13	Centro comercial Moctezuma	Francisco Chang	Sierras de Granada 9993	México D.F.	05022	Mexico
14	Chop-suey Chinese	Yang Wang	Hauptstr. 29	Bern	3012	Switzerland
15	Comércio Mineiro	Pedro Afonso	Av. dos Lusíadas, 23	São Paulo	05432-043	Brazil



# 원하는 컬럼의 값만 가지고 올 때?

- 모든 컬럼 값이 필요한 상황이 아니라면!?
- `SELECT 컬럼명 FROM table`
- `SELECT City, Country FROM Customers`

## SQL Statement:



```
SELECT City, Country FROM Customers;
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Run SQL »

## Result:

Number of Records: 91

City	Country
Berlin	Germany
México D.F.	Mexico
México D.F.	Mexico
London	UK
Luleå	Sweden
Mannheim	Germany
Strasbourg	France

# DB에는 영향 없이 원하는 데이터 추가 할 때!



- 필요한 값을 임의로 추가해서 가져오고 싶을 땐?
- `SELECT` 컬럼명, 원하는 데이터 `FROM` table
- `SELECT` City, 1, '원하는 문자열', NULL `FROM` Customers

# SQL Statement:

SELECT City, 1, '원하는 문자열', NULL FROM Customers

Edit the SQL Statement, and click "Run SQL" to see the result.

Run SQL »

# Result:

Number of Records: 91

1	City	'원하는 문자열'	NULL
1	Berlin	원하는 문자열	null
1	México D.F.	원하는 문자열	null
1	México D.F.	원하는 문자열	null
1	London	원하는 문자열	null
1	Luleå	원하는 문자열	null
1	Mannheim	원하는 문자열	null
1	Strasbourg	원하는 문자열	null

# 원하는 조건을 충족하는 Row 만 가져 올 때!



- 원하는 조건을 충족 하는 값을 찾고 싶을 땐? **WHERE**
- **SELECT \* FROM table WHERE 조건**
- **SELECT \* FROM OrderDetails WHERE Quantity > 5;**

SQL Statement:

SELECT \* FROM OrderDetails WHERE Quantity > 5;

Edit the SQL Statement, and click "Run SQL" to see the result.

Run SQL »

Result:

Number of Records: 476

OrderDetailID	OrderID	ProductID	Quantity
1	10248	11	12
2	10248	42	10
4	10249	14	9
5	10249	51	40
6	10250	41	10
7	10250	51	35
8	10250	65	15
9	10251	22	6





# Row 를 정렬해서 가져 올 때!

- 선택한 값들을 정렬해서 가지고 올 때는? ORDER BY
- SELECT \* FROM Customers ORDER BY ContactName;
- SELECT \* FROM Customers ORDER BY ContactName DESC;
- SELECT \* FROM Customers ORDER BY CustomerName ASC, ContactName DESC;

SQL Statement:

```
SELECT * FROM Customers ORDER BY ContactName;
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Run SQL »

Result:

Number of Records: 91

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
69	Romero y tomillo	Alejandra Camino	Gran Vía, 1	Madrid	28001	Spain
52	Morgenstern Gesundkost	Alexander Feuer	Heerstr. 22	Leipzig	04179	Germany
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Avda. de la Constitución 2222	México D.F.	05021	Mexico
81	Tradição Hipermercados	Anabela Domingues	Av. Inês de Castro, 414	São Paulo	05634-030	Brazil
31	Gourmet Lanchonetes	André Fonseca	Av. Brasil, 442	Campinas	04876-786	Brazil
19	Eastern Connection	Ann Devon	35 King George	London	WX3 6FW	UK
41	La maison d'Asie	Annette Roulet	1 rue Alsace-Lorraine	Toulouse	31000	France
3	Antonio Moreno Taquería	Antonio Moreno	Mataderos 2312	México D.F.	05023	Mexico
21	Familia Arquibaldo	Aria Cruz	Rua Orós, 92	São Paulo	05442-030	Brazil
75	Split Rail Beer & Ale	Art Braunschweiger	P.O. Box 555	Lander	82520	USA

SQL Statement:

```
SELECT * FROM Customers ORDER BY ContactName DESC;
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Run SQL »

Result:

Number of Records: 91

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
91	Wolski	Zbyszek	ul. Filtrowa 68	Walla	01-012	Poland
54	Océano Atlántico Ltda.	Yvonne Moncada	Ing. Gustavo Moncada 8585 Piso 20-A	Buenos Aires	1010	Argentina
42	Laughing Bacchus Wine Cellars	Yoshi Tannamuri	1900 Oak St.	Vancouver	V3F 2K1	Canada
36	Hungry Coyote Import Store	Yoshi Latimer	City Center Plaza 516 Main St.	Elgin	97827	USA



# Row 의 개수를 지정 하고 싶을 때!

- 가지고 오는 ROW의 수를 지정하고 싶을 때? **LIMIT**
- **SELECT \* FROM Customers LIMIT 10;**
- **SELECT \* FROM Customers LIMIT 30, 10;**
  - 원하는 Row 순번, 자르는 개수

## SQL Statement:

```
SELECT * FROM Customers LIMIT 10;
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Run SQL »

## Result:

Number of Records: 10

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Maria Anders	Obere Str. 57	Berlin	12209	Germany
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Avda. de la Constitución 2222	México D.F.	05021	Mexico
3	Antonio Moreno Taquería	Antonio Moreno	Mataderos 2312	México D.F.	05023	Mexico
4	Around the Horn	Thomas Hardy	120 Hanover Sq.	London	WA1 1DP	UK
5	Berglunds snabbköp	Christina Berglund	Berguvsvägen 8	Luleå	S-958 22	Sweden
6	Blauer See Delikatessen	Hanna Moos	Forsterstr. 57	Mannheim	68306	Germany
7	Blondel père et fils	Frédérique Citeaux	24, place Kléber	Strasbourg	67000	France
8	Bólide Comidas preparadas	Martín Sommer	C/ Araquil, 67	Madrid	28023	Spain
9	Bon app'	Laurence Lebihans	12, rue des Bouchers	Marseille	13008	France
10	Bottom-Dollar Marketse	Elizabeth Lincoln	23 Tsawassen Blvd.	Tsawassen	T2F 8M4	Canada

# Column 명을 변경해서 가지고 오고 싶을 때?



- Column 명을 변경해서 가지고 올 때? **AS**
- **SELECT** CustomerId **AS** id, CustomerName **AS** name **FROM**  
Customers;



## SQL Statement:

```
SELECT CustomerId AS id, CustomerName AS name FROM Customers;
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Run SQL »

## Result:

Number of Records: 91

id	name
1	Alfreds Futterkiste
2	Ana Trujillo Emparedados y helados
3	Antonio Moreno Taquería
4	Around the Horn
5	Berglunds snabbköp
6	Blauer See Delikatessen







Local 에서  
연습 시작!

# 이제 Local 에 Mysql DB 를 구축해 봅시다!



- 터미널 실행!

- Mysql -V

```
C:\Users\#tetz>mysql -V  
mysql Ver 8.0.28 for Win64 on x86_64 (MySQL Community Server - GPL)
```

- 이게 안 뜨면 시스템 환경 변수 설정
  - <https://dog-developers.tistory.com/21>

# 이제 Local 에 Mysql DB 를 구축해 봅시다!



- `mysql -u root -p`
- 설정한 비밀 번호를 치고 들어가기

-u : 아이디를 ~ 로 쓰겠다  
-p : 비밀번호로 인증 하겠다

```
Welcome to the MySQL monitor.  Commands end with ; or \q.
Your MySQL connection id is 53
Server version: 8.0.28 MySQL Community Server - GPL

Copyright (c) 2000, 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 '\h' for help. Type '\c' to clear the current input statement.

mysql> _
```

# 이제 Local 에 Mysql DB 를 구축해 봅시다!



- show databases;

MySQL 서버에서 관리하는  
모든 DB 보기

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

# 이제 Local 에 Mysql DB 를 구축해 봅시다!



- use sakila;

쿼리문을 사용할  
DB 선택하기

- show tables;

해당 DB의 테이블 보기

```
mysql> show tables
-> ;
+-----+
| Tables_in_employees |
+-----+
| current_dept_emp     |
| departments          |
| dept_emp             |
| dept_emp_latest_date |
| dept_manager         |
| employees            |
| salaries             |
| titles               |
+-----+
8 rows in set (0.00 sec)
```

# 여기서 부터는 자유롭게 Query 를 사용!



- 다만 아무래도 CLI 는 불편하기 때문에 GUI 를 사용하는 편이 좋겠죠?
- 명령어가 익숙하면 상관이 없지만, 그렇지 않다면 아무래도 Workbench 를 사용하는 편이 좋습니다!
- MySQL Workbench 실행





# MySQL Workbench

# Workbench 사용!



- 먼저 schema 선택 하기

Navigator

MANAGEMENT

- Server Status
- Client Connections
- Users and Privileges
- Status and System Variables
- Data Export
- Data Import/Restore

INSTANCE

- Startup / Shutdown
- Server Logs
- Options File

PERFORMANCE

- Dashboard
- Performance Reports
- Performance Schema Setup

Administration Schemas

Information

Table: **answer**

Columns:

- \_id** int AI PK

Query 1

question answer ans

1 • **SELECT \* FROM mydb.answer**

Result Grid

	_id	question_id	answer_text
▶	1	0	그런 모임을 왜 이제서
	2	0	1년 전에 알려줬어도
	3	1	원소리여, 그냥 하던
	4	1	오호? 그런게 있어? 들
	5	2	무슨 버그가 발생한 거
	6	2	아... 내일도 야근 각
	7	3	일단 빠르게 개발 완
	8	3	그거 내일 아침에 와
*	NULL	NULL	NULL





# Schema



- 만들어진 DB 를 확인 할 수 있습니다!

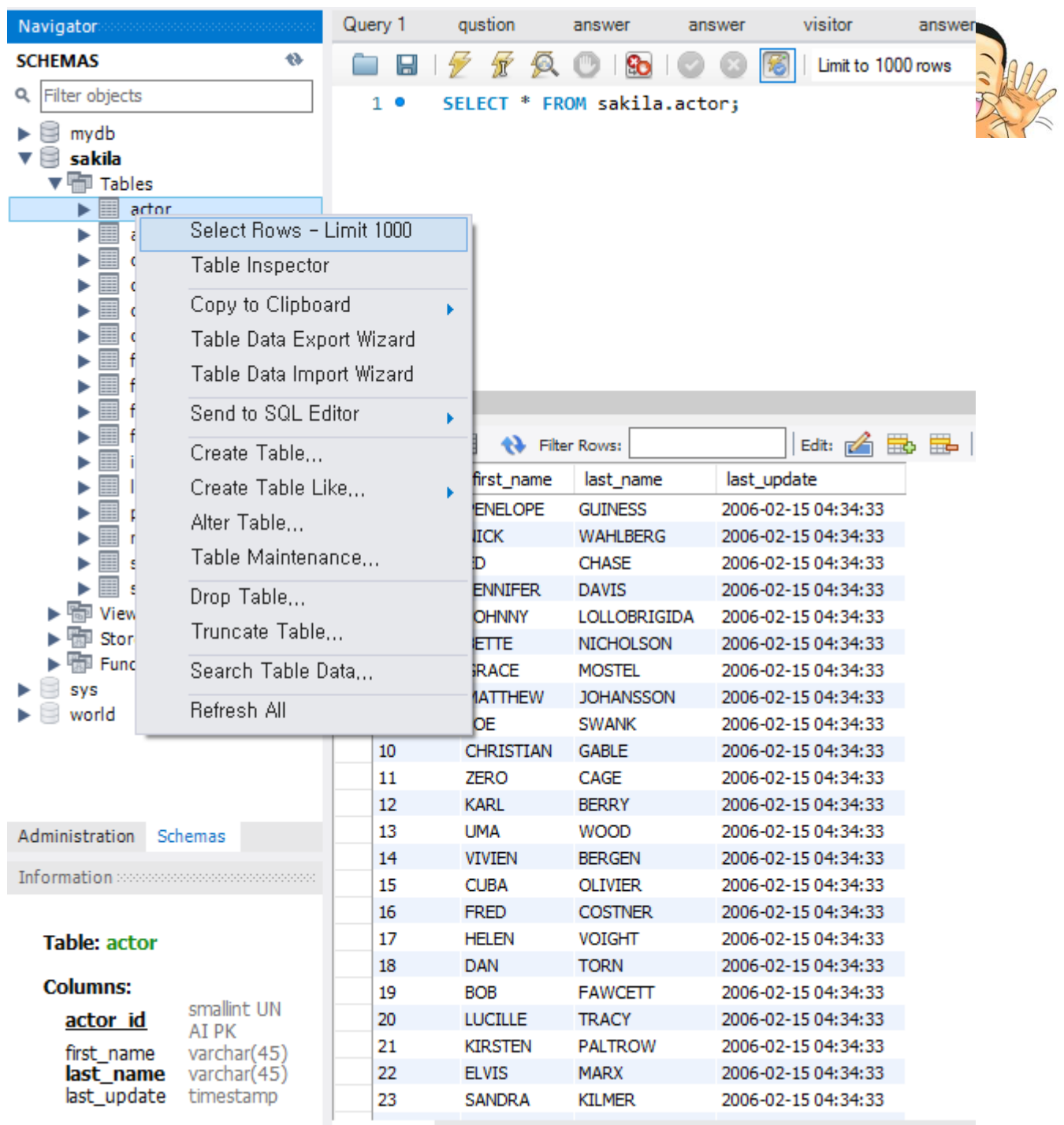
## SCHEMAS



- ▶  employees
- ▶  shopdb
- ▶  **sqlldb**
- ▶  sys

# Schema

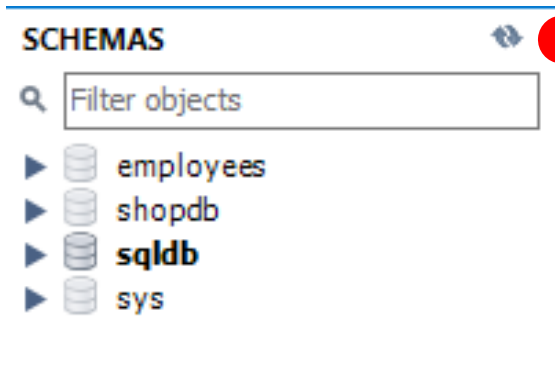
- 각각의 DB 에 있는 테이블과 데이터 확인 가능



The screenshot displays a database management interface. On the left, the 'Navigator' pane shows the 'sakila' database structure, including the 'actor' table. A context menu is open over the 'actor' table, listing various actions such as 'Select Rows - Limit 1000', 'Table Inspector', 'Copy to Clipboard', 'Table Data Export Wizard', 'Table Data Import Wizard', 'Send to SQL Editor', 'Create Table...', 'Create Table Like...', 'Alter Table...', 'Table Maintenance...', 'Drop Table...', 'Truncate Table...', 'Search Table Data...', and 'Refresh All'. The main pane shows the 'actor' table data, with columns: first\_name, last\_name, last\_update. The data is displayed in a table format, with rows numbered 1 to 23. The bottom pane shows the 'Table: actor' information, including columns: actor\_id (smallint UNSIGNED, AI, PK), first\_name (varchar(45)), last\_name (varchar(45)), and last\_update (timestamp).

	first_name	last_name	last_update
1	ENVELOPE	GUINNESS	2006-02-15 04:34:33
2	WICK	WAHLBERG	2006-02-15 04:34:33
3	D	CHASE	2006-02-15 04:34:33
4	ENNIFER	DAVIS	2006-02-15 04:34:33
5	JOHNNY	LOLLOBRIGIDA	2006-02-15 04:34:33
6	ETTE	NICHOLSON	2006-02-15 04:34:33
7	GRACE	MOSTEL	2006-02-15 04:34:33
8	MATTHEW	JOHANSSON	2006-02-15 04:34:33
9	OE	SWANK	2006-02-15 04:34:33
10	CHRISTIAN	GABLE	2006-02-15 04:34:33
11	ZERO	CAGE	2006-02-15 04:34:33
12	KARL	BERRY	2006-02-15 04:34:33
13	UMA	WOOD	2006-02-15 04:34:33
14	VIVIEN	BERGEN	2006-02-15 04:34:33
15	CUBA	OLIVIER	2006-02-15 04:34:33
16	FRED	COSTNER	2006-02-15 04:34:33
17	HELEN	VOIGHT	2006-02-15 04:34:33
18	DAN	TORN	2006-02-15 04:34:33
19	BOB	FAWCETT	2006-02-15 04:34:33
20	LUCILLE	TRACY	2006-02-15 04:34:33
21	KIRSTEN	PALTROW	2006-02-15 04:34:33
22	ELVIS	MARX	2006-02-15 04:34:33
23	SANDRA	KILMER	2006-02-15 04:34:33

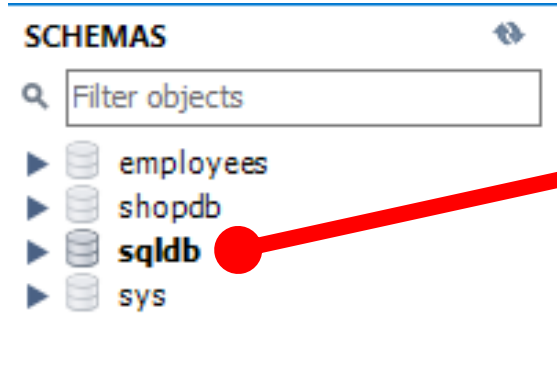
# 주의점!



이건 옛날 프로그램이라서  
자동 새로고침이 안됩니다!

따라서, 변경 사항이 있으면  
refresh 를 눌러줘야 합니다!

# 주의점!



현재 선택 중인 DB 는 DB 명이 BOLD(두껍게)  
처리 됩니다!

선택하는 방법은 더블 클릭하면 됩니다!

따라서, 자신이 선택한 DB 가  
무엇인지 잘 확인을 해야 합니다!

# 데이터 임포트






images	2024-07-10 오전 9:18	파일 폴더	
sakila	2024-07-10 오전 9:18	파일 폴더	
Changelog	2024-07-10 오전 9:18	파일	1KB
employees	2024-07-10 오전 9:18	SQL Text File	5KB
employees_partitioned	2024-07-10 오전 9:18	SQL Text File	7KB
employees_partitioned_5.1	2024-07-10 오전 9:18	SQL Text File	8KB
load_departments.dump	2024-07-10 오전 9:18	DUMP 파일	1KB
load_dept_emp.dump	2024-07-10 오전 9:18	DUMP 파일	13,829KB
load_dept_manager.dump	2024-07-10 오전 9:18	DUMP 파일	2KB
load_employees.dump	2024-07-10 오전 9:18	DUMP 파일	17,308KB
load_salaries1.dump	2024-07-10 오전 9:18	DUMP 파일	38,874KB
load_salaries2.dump	2024-07-10 오전 9:18	DUMP 파일	38,874KB
load_salaries3.dump	2024-07-10 오전 9:18	DUMP 파일	38,165KB
load_titles.dump	2024-07-10 오전 9:18	DUMP 파일	21,200KB
objects	2024-07-10 오전 9:18	SQL Text File	5KB
README	2024-07-10 오전 9:18	Markdown 원본 ...	5KB
show_elapsed	2024-07-10 오전 9:18	SQL Text File	1KB
sql_test	2024-07-10 오전 9:18	Shell Script	2KB
test_employees_md5	2024-07-10 오전 9:18	SQL Text File	5KB
test_employees_sha	2024-07-10 오전 9:18	SQL Text File	5KB
test_versions	2024-07-10 오전 9:18	Shell Script	2KB

sql 은 쿼리 구문 파일입니다!  
실제 데이터는 보통 DUMP 파일에 존재!

따라서, DUMP 파일이 존재 하는지를  
꼭 확인해야 합니다!

```
112 • SELECT 'LOADING departments' as 'INFO';
113 ✖ source load_departments.dump ;
114 • SELECT 'LOADING employees' as 'INFO';
115 ✖ source load_employees.dump ;
116 • SELECT 'LOADING dept_emp' as 'INFO';
117 ✖ source load_dept_emp.dump ;
118 • SELECT 'LOADING dept_manager' as 'INFO';
119 ✖ source load_dept_manager.dump ;
120 • SELECT 'LOADING titles' as 'INFO';
121 ✖ source load_titles.dump ;
122 • SELECT 'LOADING salaries' as 'INFO';
123 ✖ source load_salaries1.dump ;
124 ✖ source load_salaries2.dump ;
125 ✖ source load_salaries3.dump ;
126
127 ✖ source show_elapsed.sql ;
```



workbench 의 경우  
dump 파일의 경로 문제로인해서  
데이터 파일 적용이 안되는 상태

```
C:\test_db-master>mysql -u root -p
Enter password: ****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 48
Server version: 8.2.0 MySQL Community Server - GPL
```

```
Copyright (c) 2000, 2023, 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 '\h' for help. Type '\c' to clear the current input statement.
```

```
mysql> source employees.sql;
Query OK, 0 rows affected (0.00 sec)
```

덤프 파일이 있는 폴더로 직접 가서  
mysql 서버에 접속

employees.sql 에 작성된 쿼리를  
실행해서 새로운 DB 추가!

Query OK, 24924 rows affected (0.32 sec)  
Records: 24924 Duplicates: 0 Warnings: 0

Query OK, 24920 rows affected (0.32 sec)  
Records: 24920 Duplicates: 0 Warnings: 0

Query OK, 24920 rows affected (0.30 sec)  
Records: 24920 Duplicates: 0 Warnings: 0

Query OK, 24919 rows affected (0.33 sec)  
Records: 24919 Duplicates: 0 Warnings: 0

Query OK, 7671 rows affected (0.10 sec)  
Records: 7671 Duplicates: 0 Warnings: 0

data_load_time_diff
00:00:51

1 row in set (0.01 sec)

데이터 반영 완료!

```
mysql> use employees;
Database changed
mysql> show tables;
+-----+
| Tables_in_employees |
+-----+
| current_dept_emp     |
| departments          |
| dept_emp             |
| dept_emp_latest_date |
| dept_manager         |
| employees            |
| salaries             |
| titles               |
+-----+
8 rows in set (0.00 sec)

mysql> select * from employees;
```

emp_no	birth_date	first_name	last_name	gender	hire_date
10001	1953-09-02	Georgi	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

반영된 DB 선택

employees 테이블의 데이터 출력



**SCHEMAS**

Filter objects

- ▼ **employees**
  - ▼ Tables
    - departments
    - dept\_emp
    - dept\_manager
    - employees**
    - salaries
    - titles
  - Views
  - Stored Procedures
  - Functions
- shopdb
- sqldb
- sys

Limit to 1000 rows

1 • `SELECT * FROM employees.employees;`

**Result Grid** | Filter Rows: | Edit: | Export

	emp_no	birth_date	first_name	last_name	gender	hire_date
▶	10001	1953-09-02	Georgi	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
	10005	1955-01-21	Kyoichi	Maliniak	M	1989-09-12
	10006	1953-04-20	Anneke	Preusig	F	1989-06-02
	10007	1957-05-23	Tzvetan	Zielinski	F	1989-02-10
	10008	1958-02-19	Saniya	Kalloufi	M	1994-09-15
	10009	1952-04-19	Sumant	Peac	F	1985-02-18
	10010	1963-06-01	Duangkaew	Piveteau	F	1989-08-24
	10011	1953-11-07	Mary	Sluis	F	1990-01-22
	10012	1960-10-04	Patricio	Bridgland	M	1992-12-18
	10013	1963-06-07	Eberhardt	Terkki	M	1985-10-20
	10014	1956-02-12	Berni	Genin	M	1987-03-11

workbench 에서도  
확인 가능!



# 연습을 위한

# DB 만들기

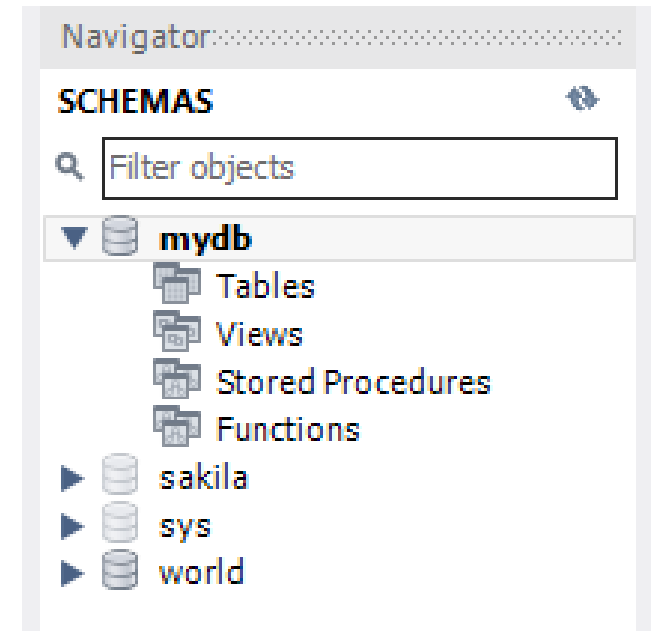
# DB 생성!



- `CREATE SCHEMA `mydb` DEFAULT CHARACTER SET utf8mb4 COLLATE utf8mb4_general_ci;`

A screenshot of a SQL query editor window. The title bar says "Query 1". Below the title bar is a toolbar with various icons for file operations, execution, and search. The main area shows a single line of SQL code: `CREATE SCHEMA `mydb` DEFAULT CHARACTER SET utf8mb4 COLLATE utf8mb4_general_ci;`. The line number "1" is visible on the left.

```
Query 1 x
1 CREATE SCHEMA `mydb` DEFAULT CHARACTER SET utf8mb4 COLLATE utf8mb4_general_ci;
```







# TABLE 생성!



# Table 이름 짓기 규칙! (필수 X)

- 소문자로 만들기
- 컬럼명은 백틱(`)으로 감싸줘야 합니다
- 단어와 단어 사이는 \_ 로 구분
- 누구나 컬럼명만 봐도 해당 컬럼이 어떤 데이터인지 알 수 있게 이름 짓기
- Primary Key 역할을 하는 단어가 최우선으로 온다
- Primary Key 컬럼은 뒤에 \_pk 를 추가
- Foreign Key 컬럼은 뒤에 \_fk 를 추가



### 3. 테이블 생성시 제약 넣기

```
CREATE TABLE people (  
  person_id INT AUTO_INCREMENT PRIMARY KEY,  
  person_name VARCHAR(10) NOT NULL,  
  nickname VARCHAR(10) UNIQUE NOT NULL,  
  age TINYINT UNSIGNED,  
  is_married TINYINT DEFAULT 0  
);
```

제약	설명
AUTO_INCREMENT	새 행 생성시마다 자동으로 1씩 증가
PRIMARY KEY	중복 입력 불가, NULL(빈 값) 불가
UNIQUE	중복 입력 불가
NOT NULL	NULL(빈 값) 입력 불가
UNSIGNED	(숫자일시) 양수만 가능
DEFAULT	값 입력이 없을 시 기본값



# DB 의 TABLE 생성!

```
1 • CREATE SCHEMA `mydb` DEFAULT CHARACTER SET utf8mb4 COLLATE utf8mb4_general_ci;
2 • CREATE TABLE user (
3     `id_pk` INT NOT NULL PRIMARY KEY AUTO_INCREMENT,
4     `name` VARCHAR(10) NOT NULL,
5     `userId` VARCHAR(10) NOT NULL UNIQUE,
6     `password` VARCHAR(10) NOT NULL,
7     `address` VARCHAR(10) NOT NULL,
8     `age` TINYINT UNSIGNED,
9     `membership` TINYINT DEFAULT 0,
10    `register_date` DATETIME DEFAULT CURRENT_TIMESTAMP,
11    `update_date` DATETIME DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP
12 )
```



# DATA 삽입하기

# 생성한 TABLE 에 DATA 삽입!



```
INSERT INTO user (name, userId, password, address, age)
VALUES ('이호석', 'xenosign', '1234', '서울 서대문구 북아현', 40);
```





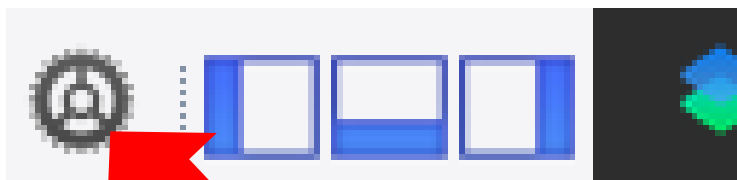
# DATA 수정 및 삭제



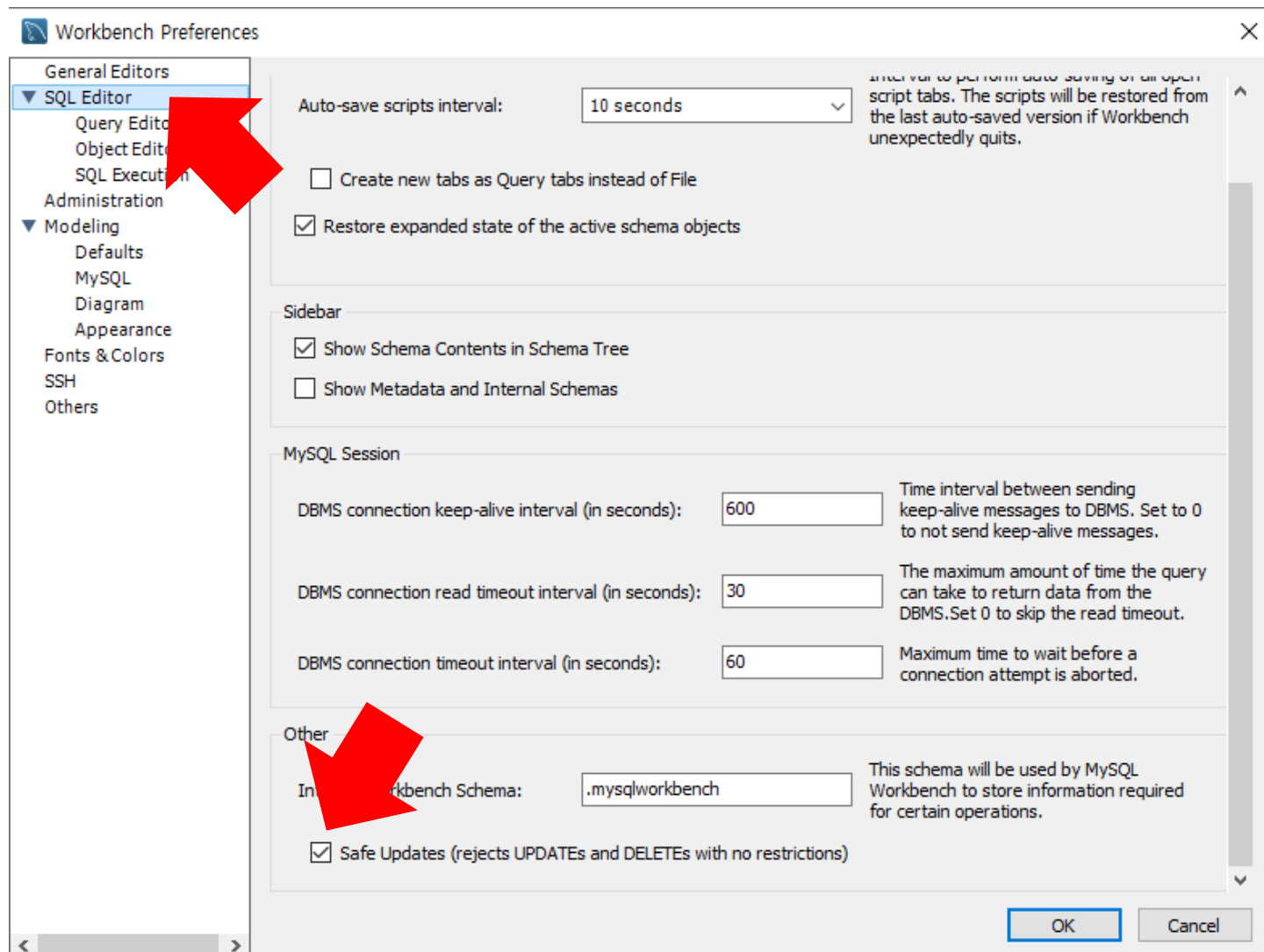
# 먼저 설정 변경이 필요합니다!



- 환경 설정 > SQL Editor



- 체크 박스를 해제





# DATA 삭제



- [illegible]



# DATA 수정



- [illegible]

22

[illegible]



# Table 수정 및 삭제하기



## ALTER TABLE - 테이블 변경

```
-- 테이블명 변경
ALTER TABLE people RENAME TO friends,
-- 컬럼 자료형 변경
CHANGE COLUMN person_id person_id TINYINT,
-- 컬럼명 변경
CHANGE COLUMN person_name person_nickname VARCHAR(10),
-- 컬럼 삭제
DROP COLUMN birthday,
-- 컬럼 추가
ADD COLUMN is_married TINYINT AFTER age;
```



## DROP TABLE - 테이블 삭제

```
DROP TABLE friends;
```



WorkBench 로  
수행하기!





# TABLE 생성!



Navigator

SCHEMAS

Filter objects

- mydb
  - Tables
    - answer
    - explanation
    - new\_table
      - Columns
      - Indexes
      - Foreign Keys
      - Triggers
    - question
    - timestamps
    - visitor
  - Views
  - Stored Procedures
  - Functions
- sakila
- sys
- world

Administration Schemas

Information

Schema: mydb

question answer answer visitor answer actor new\_table timestamps new\_table - Table x



Table Name: new\_table

Schema: mydb

Charset/Collation: Default Charset

Default Collation

Engine: InnoDB

Comments:

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

Column Name:

Data Type:

Charset/Collation:

Default:

Comments:

Storage:

☐ Virtual

☐ Stored

☐ Primary Key

☐ Not Null

☐ Unique

☐ Binary

☐ Unsigned

☐ Zero Fill

☐ Auto Increment

☐ Generated

Columns

Indexes

Foreign Keys

Triggers

Partitioning

Options

Apply

Revert





# TABLE 수정!



Navigator

SCHEMAS

Filter objects

mydb

Tables

- answer
- explain
- new\_t
- Co
- In
- For
- Tri
- questi
- timest
- visitor

Views

Stored Pro

Functions

sakila

sys

world

Administration

Sch

Information

question

answer

ansi

Table Name

Charset/Collatio

Comments:

Select Rows - Limit 1000

Table Inspector

Copy to Clipboard

Table Data Export Wizard

Table Data Import Wizard

Send to SQL Editor

Create Table...

Create Table Like...

Alter Table...

Table Maintenance...

Drop Table...

Truncate Table...

Search Table Data...

Refresh All



# Data 수정 및 삭제



Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: |

	id_pk	name	userId	password	address	age	membership	register_date	update_date
...	1	이호석	xenosign	1234	서울 서대문구 북아현	5	0	2024-07-10 11:38:22	2024-07-10 11:42:24
	3	이호석	xenosign1	1234	서울 서대문구 북아현	40		2024-07-10 11:40:09	2024-07-10 11:40:09
	5	이호석	xenosign2	1234	서울 서대문구 북아현	40		2024-07-10 11:43:53	2024-07-10 11:43:53
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Export/Import: | Wrap Cell Content: |

membership	register_date	update_date
0	2024-07-10 11:38:22	2024-07-10 11:42:24
0	2024-07-10 11:40:09	2024-07-10 11:40:09
0	2024-07-10 11:43:53	2024-07-10 11:43:53
NULL	NULL	NULL

Result Grid  
Form Editor  
Field Types

Apply Revert



## Review SQL Script

Apply SQL Script

## Review the SQL Script to be Applied on the Database

```
1 UPDATE `mydb`.`user` SET `NAME` = '이효' WHERE (`ID_PK` = '1');  
2
```



Back

Apply

Cancel



## Review SQL Script

## Apply SQL Script

## Applying SQL script to the database

The following tasks will now be executed. Please monitor the execution.  
Press Show Logs to see the execution logs.

☒ Execute SQL Statements

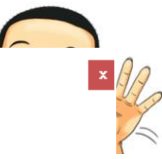
SQL script was successfully applied to the database.

Show Logs

Back

Finish

Cancel





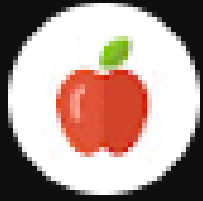
# 정규화





# 정규화?

- DB 설계에 있어서 중복을 최소화 하기 위해 데이터를 구조화 하는 과정
- 크고 조직화 되지 않은 테이블 → 작고, 잘 조직된 테이블로 변경
- 이렇게 작성을 해야만, 데이터 추가 및 삭제 시에 이상 현상(Abnormal)을 예방 할 수 있습니다!



코딩애플

구독자 14.9만명

<https://www.youtube.com/watch?v=Y1FbowQRcml>



# 제 1 정규형(1NF)

- 하나의 컬럼은 반드시 하나의 속성만을 가져야 하는 법칙

회원번호	회원이름	프로그램
101	강호동	스쿼시초급
102	손흥민	헬스
103	김민수	헬스, <u>골프초급</u>

회원번호	회원이름	프로그램
101	강호동	스쿼시초급
102	손흥민	헬스
103	김민수	헬스
103	김민수	골프초급



## 제 2 정규형(2NF)

- 모든 컬럼에 대한 부분 종속이 없어야 한다 → 현 테이블의 주제(?)와 필요 없는 친구는 빼준다

회원번호	회원이름	프로그램	가격	납부여부
101	강호동	스쿼시초급	5000	0
102	손흥민	헬스	6000	1
103	김민수	헬스	6000	1
103	김민수	골프초급	8000	0

수강등록현황 table				프로그램 table	
회원번호	회원이름	프로그램	납부여부	프로그램	가격
101	강호동	스쿼시초급	0	스쿼시초급	5000
102	손흥민	헬스	1	헬스	6000
103	김민수	헬스	1	골프초급	8000
103	김민수	골프초급	0		



## 제 3 정규형(3NF)

- 이행 종속성( $A = B$ ,  $B = C$ 여서  $A = C$ 인 경우)이 없어야 한다
- 일반 컬럼에만 종속된 컬럼은 다른 테이블로 빼기

프로그램	가격	강사	출신대학
스쿼시	5000	김을용	서울대
헬스	6000	박덕팔	연세대
골프	8000	이상구	고려대
골프중급	9000	이상구	고려대
개인피티	6000	박덕팔	연세대

프로그램	가격	강사	강사	출신대학
스쿼시	5000	김을용	김을용	서울대
헬스	6000	박덕팔	박덕팔	연세대
골프	8000	이상구	이상구	고려대
골프중급	9000	이상구	이상구	
개인피티	6000	박덕팔	박덕팔	



# Foreign Key

(외래 키)



# 외래 키(Foreign Key)

- 정규화를 하게 되면 테이블은 최소한의 단위로 쪼개지게 됩니다!
- 하지만 데이터를 불러 들일 때에는 한꺼번에 많은 값을 가지는 테이블을 JOIN 해서 가지고 와야 하는 경우가 많습니다.
- 앞서 배운 JOIN 을 사용하여 테이블을 합치게 되는데요. 보통 기준이 되는 값을 통해서 테이블을 합쳐 주게 됩니다! → 이 때 기준이 되는 값(서로 공유하고 있는 값)을 외래 키라고 합니다!



기본키(Primary Key)

외래키(Foreign Key)

기본키(Primary Key)

선수번호	이름	팀 코드	포지션	등번호	키
1	김남일	K03	DF	33	177
2	박지성	K07	MF	7	178
3	이영표	K02	MF	22	176

〈선수 테이블〉

팀 코드	팀 명	연고지
K03	스틸러스	포항
K07	드래곤즈	전남
K02	블루윙즈	수원

〈구단 테이블〉

선수번호	이름	팀 코드	포지션	등번호	키	팀 명	연고지
1	김남일	K03	DF	33	177	스틸러스	포항
2	박지성	K07	MF	7	178	드래곤즈	전남
3	이영표	K02	MF	22	176	블루윙즈	수원