

CSE235

데이터베이스 시스템

(Database Systems)

**Practice: SQL & Relational
Algebra**

담당교수: 전강욱(컴퓨터공학부)

kw.chon@koreatech.ac.kr

실습 목적

- 사용자 질의를 SQL로 표현하여, 데이터베이스 활용에 대한 이해를 높이는 것을 목적으로 함
- 사용자 질의에 대해 관계 대수 연산식이 어떻게 표현되는지를 풀어 봄으로써, 데이터베이스의 질의 처리에 대한 이해를 높이는 것을 목적으로 함

개발환경

- Windows

- SQL

- MySQL mysql-8.0.32
- MySQL Workbench 8.0.32

- Relational Algebra


- RADB
 - <https://users.cs.duke.edu/~junyang/ra2/>
 - Python으로 구현된 관계 대수 번역기
 - 사전설치 툴: Anaconda3 설치

개발환경 구축: MySQL

- 아래 사이트 접속 (GPL 라이선스 다운로드 URL)
 - <https://dev.mysql.com/downloads/mysql/>

개발환경 구축: MySQL

■ MSI Installer 다운로드

[General Availability \(GA\) Releases](#) [Archives](#) 


MySQL Installer 5.7.41

Select Operating System:

Microsoft Windows ▼

Looking for the latest GA version?

Windows (x86, 32-bit), MSI Installer (mysql-installer-web-community-5.7.41.0.msi)	5.7.41	2.4M	Download
Windows (x86, 32-bit), MSI Installer (mysql-installer-community-5.7.41.0.msi)	5.7.41	517.1M	Download

 We suggest that you use the [MD5 checksums and GnuPG signatures](#) to verify the integrity of the packages you download.

개발환경 구축: MySQL

■ 로그인 건너뛰기

MySQL Community Downloads

Login Now or Sign Up for a free account.

An Oracle Web Account provides you with the following advantages:

- Fast access to MySQL software downloads
- Download technical White Papers and Presentations
- Post messages in the MySQL Discussion Forums
- Report and track bugs in the MySQL bug system

Login »

using my Oracle Web account

Sign Up »

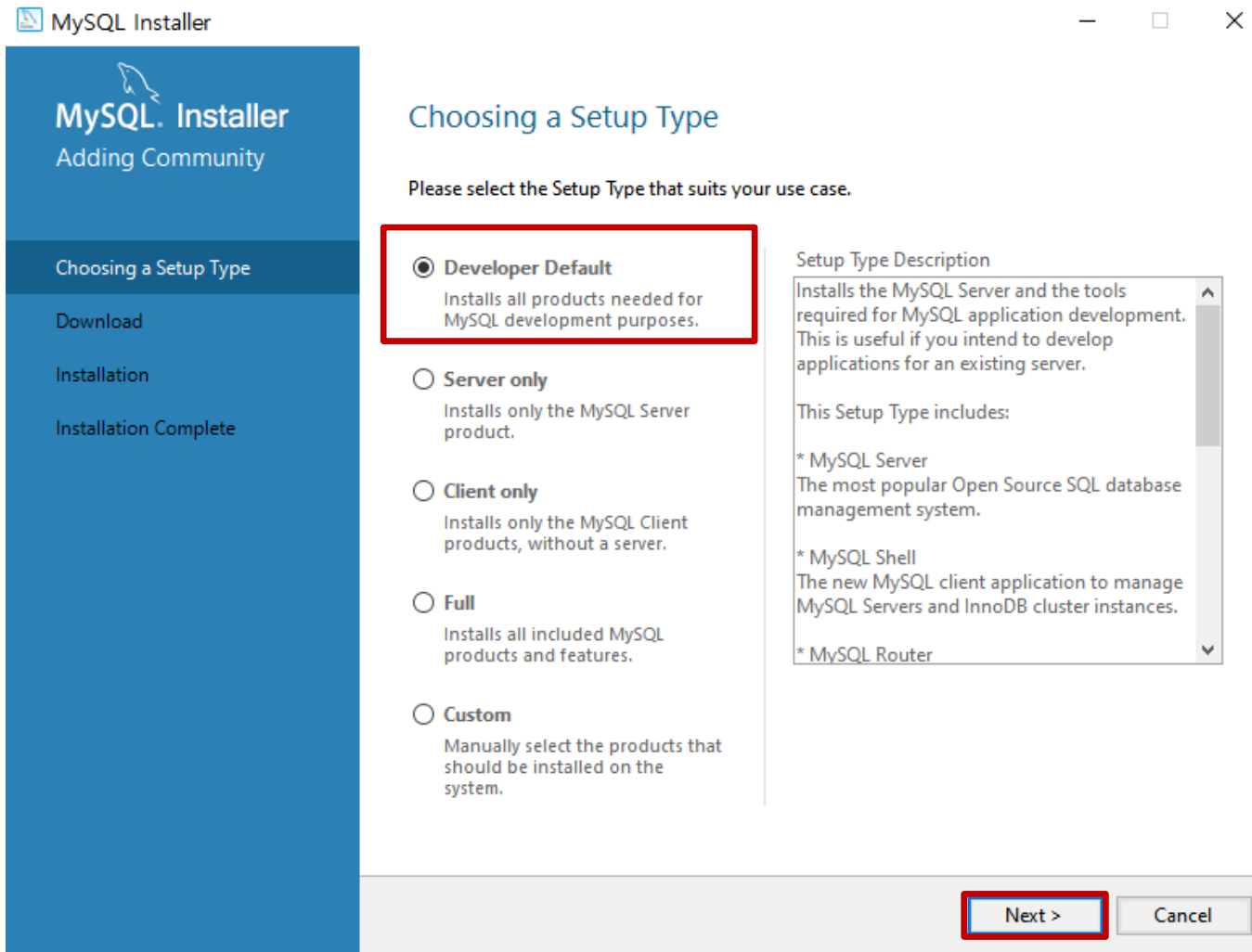
for an Oracle Web account

MySQL.com is using Oracle SSO for authentication. If you already have an Oracle Web account, click the Login link. Otherwise, you can signup for a free account by clicking the Sign Up link and following the instructions.

No thanks, just start my download.

개발환경 구축: MySQL

■ 개발자 모드 선택 후 설치



개발환경 구축: MySQL

MySQL Installer

MySQL[®] Installer
Adding Community

Choosing a Setup Type

Path Conflicts

Installation

Product Configuration

Installation Complete

Installation

The following products will be installed.

	Product	Status	Progress	Notes
✓	MySQL Server 8.0.32	Complete		
✓	MySQL Workbench 8.0.32	Complete		
✓	MySQL Shell 8.0.32	Complete		
✓	MySQL Router 8.0.32	Complete		
✓	Connector/ODBC 8.0.32	Complete		
✓	Connector/C++ 8.0.32	Complete		
✓	Connector/J 8.0.32	Complete		
✓	Connector/NET 8.0.32	Complete		
✓	Connector/Python 8.0.32	Complete		
✓	MySQL Documentation 8.0.32	Complete		
✓	Samples and Examples 8.0.32	Complete		

Show Details >

< Back

Next >

Cancel

개발환경 구축: MySQL

■ Port 번호 설정 (default: 3306)

- 혹시, 포트 충돌이 날 경우 변경

MySQL Installer

MySQL. Installer
MySQL Server 8.0.32

Type and Networking

Authentication Method

Accounts and Roles

Windows Service

Apply Configuration

Type and Networking

Server Configuration Type

Choose the correct server configuration type for this MySQL Server installation. This setting will define how much system resources are assigned to the MySQL Server instance.

Config Type: Development Computer

Connectivity

Use the following controls to select how you would like to connect to this server.

☒ TCP/IP Port: 3305 X Protocol Port: 33060

☒ Open Windows Firewall ports for network access

☐ Named Pipe Pipe Name: MYSQL

☐ Shared Memory Memory Name: MYSQL

Advanced Configuration

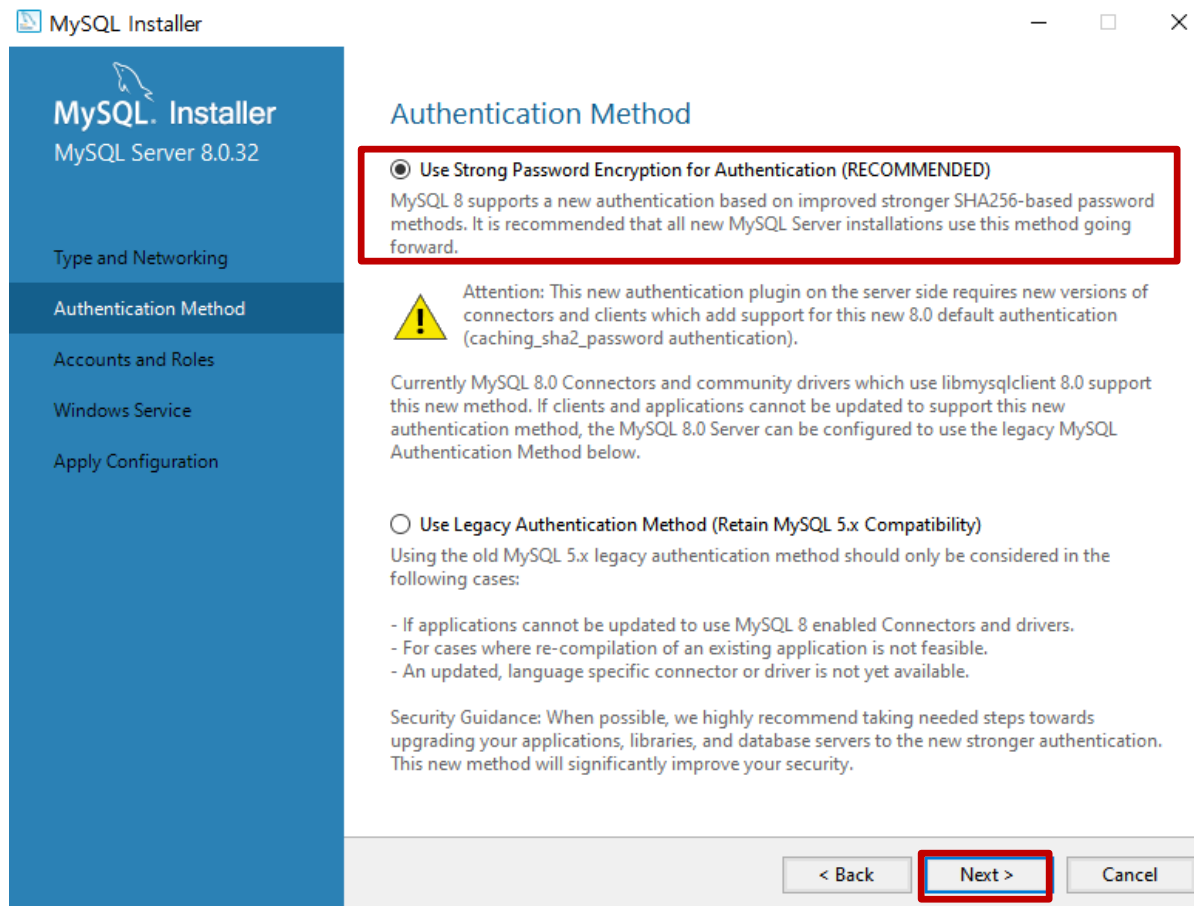
Select the check box below to get additional configuration pages where you can set advanced and logging options for this server instance.

☐ Show Advanced and Logging Options

Next > Cancel

개발환경 구축: MySQL

- MySQL 5.x 버전과 호환성이 필요 없는 경우 추천(RECOMMENDED) 선택



개발환경 구축: MySQL

■ 비밀번호 설정

- 비밀번호는 추후 DB 접속 시 계속 필요하므로 기억

MySQL Installer

MySQL Installer
MySQL Server 8.0.32

Type and Networking
Authentication Method
Accounts and Roles
Windows Service
Server File Permissions
Apply Configuration

Accounts and Roles

Root Account Password
Enter the password for the root account. Please remember to store this password in a secure place.

MySQL Root Password:

Repeat Password:

Password strength: Strong

MySQL User Accounts
Create MySQL user accounts for your users and applications. Assign a role to the user that consists of a set of privileges.

MySQL User Name	Host	User Role
-----------------	------	-----------

Add User
Edit User
Delete

< Back **Next >** Cancel

개발환경 구축: MySQL

■ Windows Service 이름 지정

MySQL Installer

MySQL. Installer
MySQL Server 8.0.32

Type and Networking

Authentication Method

Accounts and Roles

Windows Service

Server File Permissions

Apply Configuration

Windows Service

☒ Configure MySQL Server as a Windows Service

Windows Service Details

Please specify a Windows Service name to be used for this MySQL Server instance.
A unique name is required for each instance.

Windows Service Name:

☒ Start the MySQL Server at System Startup

Run Windows Service as ...

The MySQL Server needs to run under a given user account. Based on the security requirements of your system you need to pick one of the options below.

☒ Standard System Account
Recommended for most scenarios.

☐ Custom User
An existing user account can be selected for advanced scenarios.

< Back Next > Cancel

개발환경 구축: MySQL

■ 이후 Next 및 Finish 버튼 클릭 (4번 제외)

1 Server File Permissions

MySQL Installer can secure the server's data directory by updating the permissions of files and folders located at:

C:\ProgramData\MySQL\MySQL Server 8.0\Data

Do you want MySQL Installer to update the server file permissions for you?

☒ Yes, grant full access to the user running the Windows Service (if applicable) and the administrators group only. Other users and groups will not have access.

☐ Yes, but let me review and configure the level of access.

☐ No, I will manage the permissions after the server configuration.

2 Apply Configuration

Click [Execute] to apply the changes

Configuration Steps | Log

- ☐ Writing configuration file
- ☐ Updating Windows Firewall rules
- ☐ Adjusting Windows service
- ☐ Initializing database (may take a long time)
- ☐ Updating permissions for the data folder and related server files
- ☐ Starting the server
- ☐ Applying security settings
- ☐ Updating the Start menu link

3 Apply Configuration

The configuration operation has completed.

Configuration Steps | Log

- ☒ Writing configuration file
- ☒ Updating Windows Firewall rules
- ☒ Adjusting Windows service
- ☒ Initializing database (may take a long time)
- ☒ Updating permissions for the data folder and related server files
- ☒ Starting the server
- ☒ Applying security settings
- ☒ Updating the Start menu link

The configuration for MySQL Server 8.0.32 was successful. Click Finish to continue.

4 Connect To Server

Select the MySQL server instances from the list to receive sample schemas and data.

Server	Port	Arch...	Type	Status
<input checked="" type="checkbox"/> MySQL Server 8.0.32	3306	X64	Stand-alone Server	Connection succeeded.

Provide the credentials that should be used (requires root privileges). Click "Check" to ensure they work.

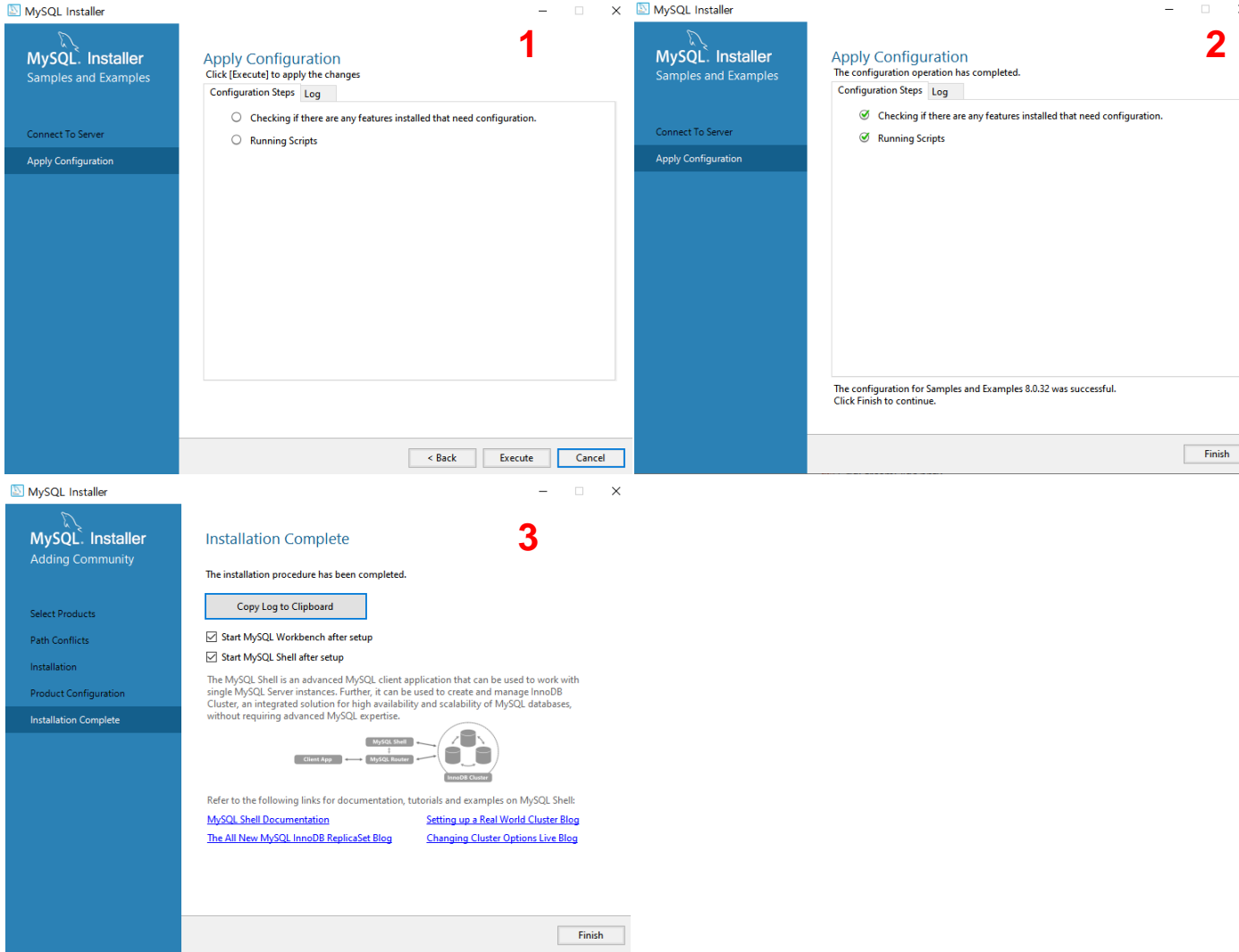
User name: root Credentials provided in Server configuration

Password:

Password 입력

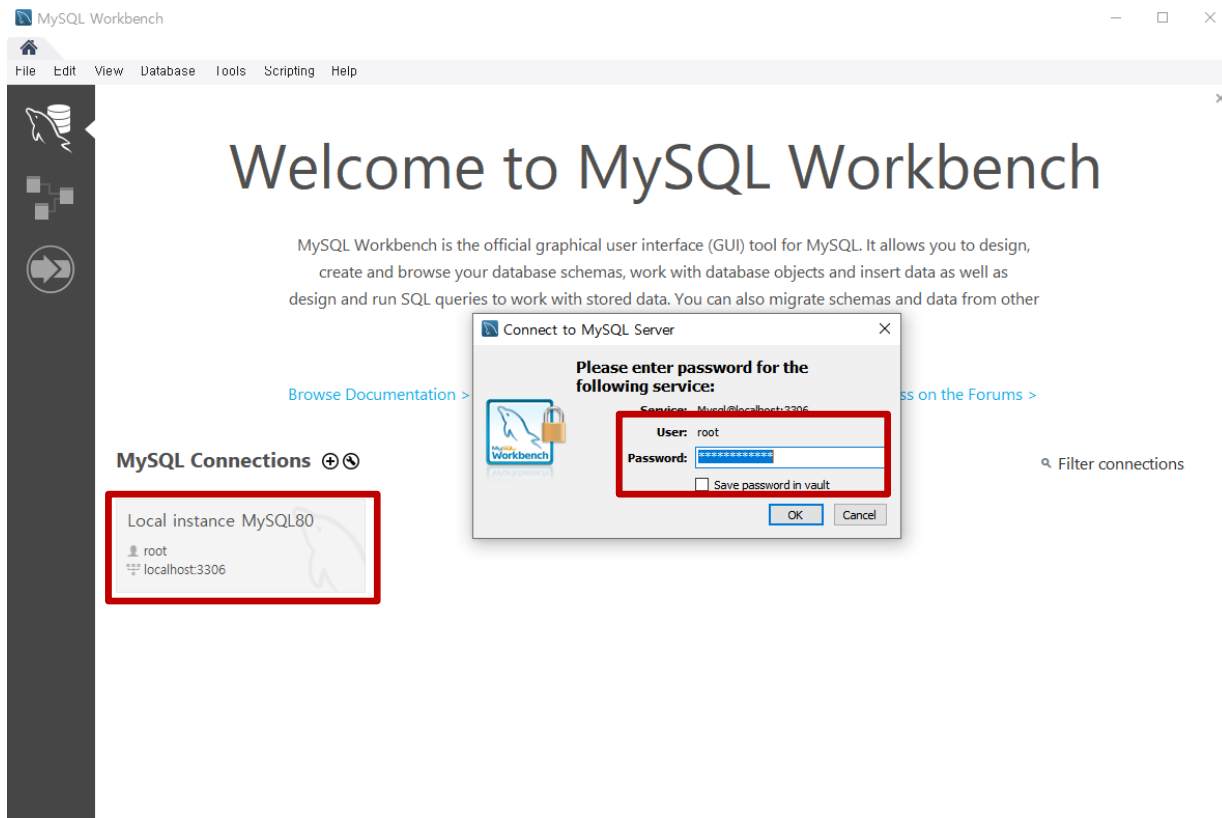
개발환경 구축: MySQL

■ 이후 Next 및 Finish 버튼 클릭



개발환경 구축: MySQL

- MySQL Workbench 실행
- 설치된 MySQL에 접근 비밀번호 입력



개발환경 구축: MySQL

MySQL Workbench

Local instance MySQL80 x MySQL Model* x EER Diagram x

File Edit View Query Database Server Tools Scripting Help

Menu

Navigator

Query 1 x

SCHEMAS

Filter objects

- book
- sakila
- sys
- world

1 • `select version();`

Query 작성을 위한 공간

Limit to 1000 rows

Result Grid

version()
8.0.32

Query 결과 출력

개발환경 구축: RADB

- Anaconda Prompt (anaconda3) 설치 후 아래 명령어 실행 후 설치 확인

```
pip install radb
```

```
conda list
```

qtconsole	5.3.0	pyhd3eb1b0_0	
qtpy	2.0.1	pyhd3eb1b0_0	
queuelib	1.5.0	py39haa95532_0	
radb	3.0.4	pypi_0	pypi
regex	2022.3.15	py39h2bbtf1b_0	
requests	2.27.1	pyhd3eb1b0_0	
requests-file	1.5.1	pyhd3eb1b0_0	
requests-oauthlib	1.3.1	pypi_0	pypi

개발환경 구축: RADB

- Conda 실행 후, 아래와 같이 RADB 실행

```
(base) C:\Users\82109\RA>radb beers.db
```

- 아래와 같이 RADB 내 SQL명령어 실행 가능

- 테이블 생성: `₩sqlxec_{ CREATE TABLE Bar(name VARCHAR(20) NOT NULL PRIMARY KEY, address VARCHAR(20))};`

- 값 입력: `₩sqlxec_{INSERT INTO Bar VALUES('Down Under Pub', '802 W. Main Street')};`

- 조회: `₩sqlxec_{SELECT * FROM Bar};`

- 스크립트 파일 생성하여 한번에 여러 개 명령어 실행 가능

```
(base) radb -i beers.ra beers.db
```

스크립트 파일명 DB명
(EL 시스템 내 첨부파일 확인)

개발환경 구축: RADB

■ SQL

```
❑ SELECT name FROM Bar WHERE name like 'Jame%';
```

■ 관계대수

```
ra> #select_{name like 'Jame%'} Bar;  
(name:string, address:string)
```

```
-----  
James Joyce Pub, 912 W. Main Street  
-----
```

```
1 tuple returned  
ra> #project_{name} Drinker;  
(name:string)  
-----
```

개발환경 구축: RADB

■ SQL

```
❑ SELECT name FROM Drinker;
```

■ 관계대수

```
ra> #project_{name} Drinker;  
(name:string)
```

```
-----  
Amy  
Ben  
Coy  
Dan  
Eve  
-----
```

```
5 tuples returned
```

개발환경 구축: RADB

■ SQL

```
❑ SELECT drinker, address  
FROM drinker  
JOIN likes  
ON drinker.name = likes.drinker  
AND beer = 'corona');
```

■ 관계대수

```
ra> ⋈project_{drinker, address}(⋈project_{drinker} ⋈select_{beer='Corona'} likes ⋈join_{name=drinker} drinker);  
(drinker:string, address:string)
```

```
-----  
Amy, 100 W. Main Street  
Dan, 300 N. Duke Street  
Eve, 100 W. Main Street  
-----
```

Sample DB 구축

■ Sample DB 구축

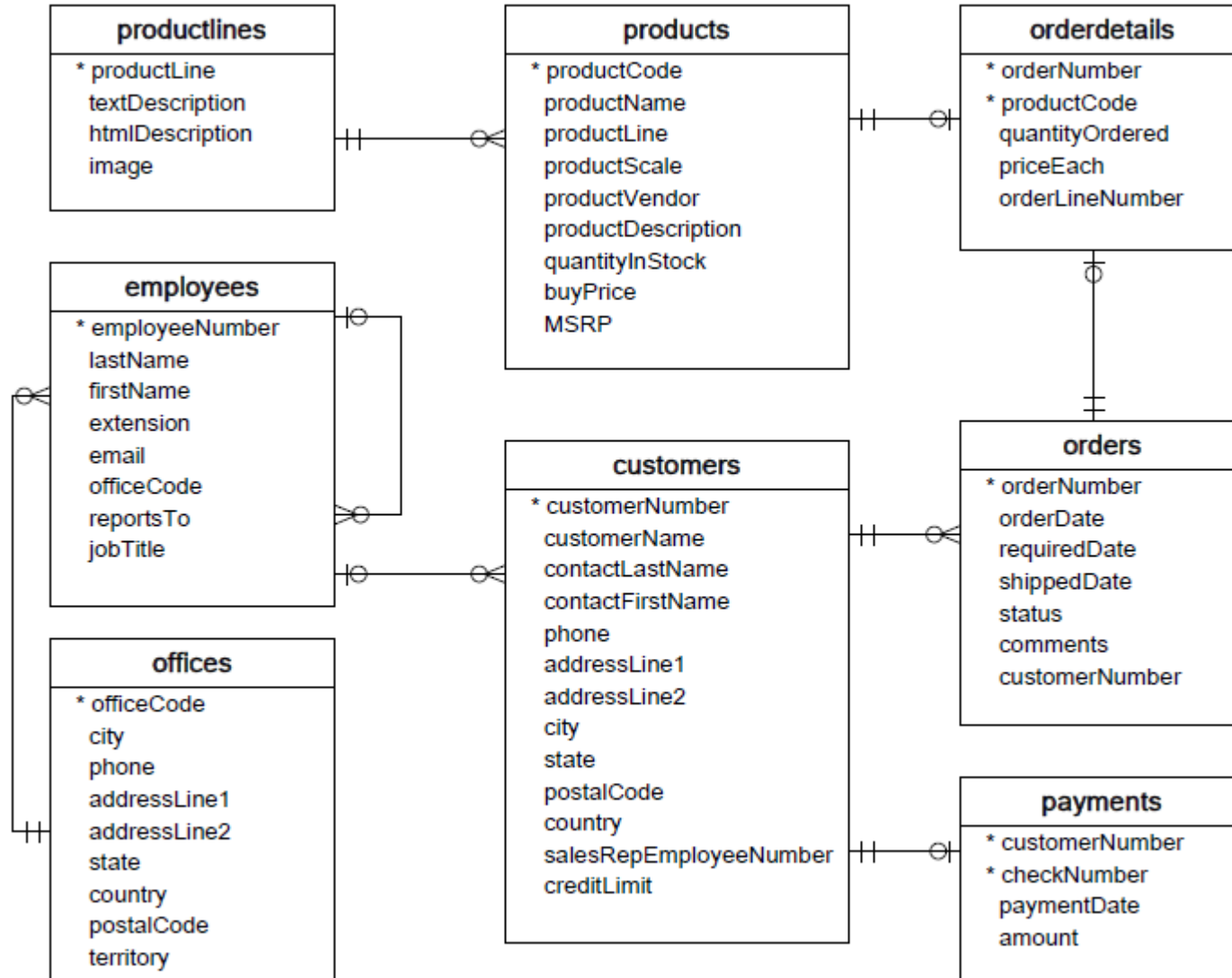
- `mysqlsampledatabase.sql` 파일 다운로드

- 출처: <https://github.com/hhorak/mysql-sample-db>

- 데이터 상세

- Customers: stores customer's data
 - Products: stores a list of scale model cars
 - ProductLines: stores a list of product line categories
 - Orders: stores sales orders placed by customers
 - OrderDetails: stores sales order line items for each sales order
 - Payments: stores payments made by customers based on their accounts
 - Employees: stores all employee information as well as the organization structure such as who reports to whom
 - Offices: stores sales office data.

Sample DB 구축 (계속)



출처: <https://www.mysqltutorial.org/mysql-sample-database.aspx>.

Sample DB 구축 (계속)

- Command Line Client 실행
- Password 입력 후 하기 명령어 수행
 - source \$PATH/mysqlsampledatabase.sql
 - \$PATH: mysqlsampledatabase.sql 파일을 포함하는 디렉토리

실습

- 아래 Query에 대해서 SQL문 및 관계대수문을 작성하시고, 출력 화면을 캡처하여 제출(SQL 코드 및 RADB 코드는 파일로 저장하여 제출)
 - employees의 lastname, firstname, jobtitle의 정보를 검색
 - customers중 country가 USA이면서 NYC라는 city에 살고 있으며 creditlimit이 200미만인 사람의 이름(customername)을 검색
 - buyprice가 50이하이거나 100이상인 products의 productcode, productname, buyprice를 검색
 - lastname에 am이 들어가고 firstname이 er으로 끝나는 employees의 employeenumber와 lastname, firstname을 검색
 - customers의 contactlastname은 사전의 역순으로, 고객의 contactfirstname은 사전순으로 정렬될 수 있도록 검색
 - buyprice가 가장 높은 products의 productname과 buyprice를 10위까지 추출
 - priceeach가 가장 작은 products의 productname과 priceeach를 검색
 - amount가 40000이상인 customers가 사는 city를 모두 나열

실습 (계속)

- 아래 Query에 대해서 SQL문 및 관계대수문을 작성하시고, 출력 화면을 캡처하여 제출(SQL 코드 및 RADB 코드는 파일로 저장하여 제출)
 - 고객테이블에서 고객번호, 고객 Lastname정보를 얻은 결과와 직원테이블에서 직원번호, 직원 Lastname정보를 합치고 번호에 대하여 오름차순으로 정렬한 결과를 보시오
 - 주문의 상태와 그 상태를 갖는 주문의 수를 출력하여라
 - 상태를 갖는 주문의 수가 많은 순으로 출력
 - 상세주문 테이블에서 주문번호별 총 주문수, 총 개당 가격을 추출하시오
 - 총 주문수는 1000건 이상이고 총 개당 가격은 600이상인 ROW만 추출
 - INNER JOIN을 활용하여 priceEach가 가장 작은 물품의 이름과 priceEach를 검색하시오
 - 주문 테이블과 상세 주문 테이블로부터 주문번호 별로 주문번호, 상태, 총 주문금액을 검색하시오
 - 하나의 주문번호에는 여러 개의 상품주문이 있기 때문에 총 주문금액은 quantityOrdered*priceEach들의 Sum이 되어야 함
 - 모든 고객의 주문 정보를 검색하시오(추출해야 할 정보는 고객번호, 고객이름, 주문번호, 상품상태)
 - Q: **SELECT** c.customerNumber, c.customerName, orderNumber, o.status **FROM** customers c **LEFT JOIN** orders o **ON** c.customerNumber=o.customerNumber;
 - Q의 결과에서 NULL값이 포함되었다면, 이유를 설명하시오
 - 고객 중 주문을 하지 않은 고객의 고객번호와 고객이름을 추출하시오

실습 (계속)

- 아래 Query에 대해서 SQL문을 작성하시고, 출력 화면을 캡처하여 제출 (SQL 코드는 파일로 저장하여 제출) 아래 데이터베이스 및 테이블을 구축 하시오
- empdb라는 이름의 database를 만드시오
 - Database 안에 merit이라는 테이블을 생성하시오(아래는 테이블 정보)
 - performance INT(11) NOT NULL
 - percentage FLOAT NOT NULL
 - PRIMARY KEY(performance)
 - Database 안에 employees라는 테이블을 만드시오
 - emp_id INT(11) NOT NULL AUTO_INCREMENT,
 - emp_name VARCHAR(255) NOT NULL,
 - performance INT(11) DEFAULT NULL,
 - salary FLOAT DEFAULT NULL,
 - PRIMARY KEY (emp_id) CONSTRAINT fk_performance FOREIGN KEY (performance) REFERENCES merits(performance)
 - merit 테이블에 실적(performance)별 보너스 percentage를 입력하시오
 - 1 → 0, 2 → 0.01, 3 → 0.03, 4 → 0.05, 5 → 0.08
 - employees 테이블에 직원 데이터를 입력하시오
 - ('Mary Doe', 1, 50000),
 - ('Cindy Smith', 3, 65000),
 - ('Sue Greenspan', 4, 75000),
 - ('Grace Dell', 5, 125000),
 - ('Nancy Johnson', 3, 85000),
 - ('John Doe', 2, 45000),
 - ('Lily Bush', 3, 55000);

실습 (계속)

■ buildings 테이블 생성 및 데이터 입력

- **CREATE TABLE** buildings (
 building_no **INT(11) NOT NULL AUTO_INCREMENT**,
 building_name **VARCHAR(255) NOT NULL**,
 address **VARCHAR(355) NOT NULL**,
 PRIMARY KEY (building_no));
- **INSERT INTO** buildings(building_name, address)
 VALUES('ACME Headquarters', '3950 North 1st Street CA 95134'),
 ('ACME Sales', '5000 North 1st Street CA 95134');

■ rooms 테이블 생성 및 데이터 입력

- **CREATE TABLE** rooms(
 room_no **INT(11) NOT NULL AUTO_INCREMENT**,
 room_name **VARCHAR(255) NOT NULL**,
 building_no **INT(11) NOT NULL**,
 PRIMARY KEY (room_no),
 KEY building_no(building_no),
 CONSTRAINT rooms_ibfk_1
 FOREIGN KEY(building_no)
 REFERENCES buildings (building_no)
 ON DELETE CASCADE);
- **INSERT INTO** rooms(room_name,building_no) **VALUES**('Amazon',1), ('War Room',1),
 ('Office of CEO',1), ('Marketing', 2), ('Showroom',2);

실습 (계속)

- Employee Table에 employeeNumber는 1901번, lastName은 hong, firstName은 jin, extension은 x5000, email은 nike@google.com, officeCode는 10, reportsTo는 null, jobTitle은 Research인 사원의 정보를 입력하시오
- Jobtitle이 research인 직원이 속한 office의 city를 Jeonju로 변경하시오

실습 (계속)

- 실적 별 월급은 실적이 올라갈 때 마다 월급=월급+월급
*percentage로 오른다고 가정하고, 현재 employees 테이블에 들어
가 있는 월급 데이터는 실적정보가 반영되어 있지 않다고 할 때 실
적을 반영한 월급으로 갱신 시키시오
- Employees와 offices Table에서 officeCode가 1인 Row들을 삭제
하시오 (삭제 후 결과 화면 스크린 샷)
- 27 page에서 입력시킨 사원을 삭제 하시오
- room 테이블을 캡처하고, 2번 building을 삭제 한 후에 room 테이
블을 캡처 하시오

감사합니다!

담당교수: 전강욱(컴퓨터공학부)

kw.chon@koreatech.ac.kr

Appendix

담당교수: 전강욱(컴퓨터공학부)

kw.chon@koreatech.ac.kr

SELECT문

- DB 내 데이터를 검색할 경우 사용
- 사용법
 - **SELECT** column명 **FROM** table명;
 - e.g., **SELECT * FROM** employees;
 - Employees내 모든 속성 값들을 출력

WHERE절

- **SELECT로 검색한 결과들에 조건을 제시하여, 조건에 맞는 행만을 추출하기 위한 절**
- **사용법**
 - **SELECT** column명 **FROM** table명 **WHERE** 조건;
 - 조건절에는 논리연산자(AND, OR, NOT 등)나 비교연산자(=, <, >, <=, >=, <> 등) 등이 들어갈 수 있음
 - e.g., **SELECT** employeeNumber, lastName, firstName **FROM** employee **WHERE** firstName= "Leslie";

BETWEEN 연산자

- BETWEEN 연산자를 이용하여 특정 범위 지정
- 사용법
 - **SELECT** column명 **FROM** table명 **WHERE** column **BETWEEN** start **AND** end;
 - $\text{start} \leq \text{column} \leq \text{end}$ 인 튜플들 모두 출력
 - e.g., **SELECT** employeeNumber, lastName, firstName **FROM** employee **WHERE** employeeNumber **BETWEEN** 1 **AND** 10;

LIKE 연산자

- 특정 패턴의 데이터를 검색 가능

- 사용법

- **SELECT** column명 **FROM** table명 **WHERE** column **LIKE** % 또는 _;

- column 값이,

- LIKE 'n%': n으로 시작하는 행들이 출력
 - LIKE '%n': n으로 끝나는 행들이 출력
 - LIKE '%n%': n을 포함하는 행들이 출력
 - LIKE 'n_w': n과 w 사이에 어떤 문자가 오든 검색

IN 연산자

- 리스트의 요소들을 쉽게 순회
- 사용법
 - **SELECT** column명 **FROM** table명 **WHERE** column **IN** ('VAL_1','VAL_2',...,'VAL_N');
 - e.g., **SELECT** * **FROM** company **WHERE** companyName **IN** ('microsoft','Apple','SK');
 - e.g., **SELECT** * **FROM** company **WHERE** companyName = 'microsoft' **OR** companyName = 'Apple' **OR** companyName = 'SK';

ORDER BY절

- **SELECT문의 출력 값을 특정한 조건으로 정렬**
- **사용법**
 - **SELECT column명 FROM table명 ORDER BY column [ASC|DESC];**
 - ASC: 사전 순으로 정렬
 - DESC: 사전의 역순으로 정렬
 - **SELECT column명 FROM table명 ORDER BY FIELD (column, 'value1', 'value2');**
 - 특정 순서로 정렬

DISTINCT 연산자

- 중복되는 행 제거

- 사용법

- **SELECT DISTINCT** column명 **FROM** table명
WHERE 조건;

LIMIT 절

- 추출되는 행들의 범위 및 개수를 정해주는 기능

- 사용법

- **SELECT** column명 **FROM** table명 **LIMIT** [start,] count;
 - start 번째 행부터 count개의 행 출력
 - start를 입력하지 않을 경우 0번째 행부터 출력

Aggregate 함수

- 여러 행들의 정보를 요약하여 통계 값을 계산하는데 사용
- 종류: AVG, COUNT, SUM, MAX, MIN
- 사용법
 - **SELECT** [AVG|COUNT|SUM|MAX|MIN]column명 **FROM** table명;
 - e.g., **SELECT** MAX(salary) maxSalary **FROM** employees;

Subquery

- WHERE절에 다른 SELECT 문이 존재하는 질의
- 사용법
 - 사무실이 영국에 있는 직원의 성과 이름을 추출하라
 - `SELECT lastName, firstName`
`FROM employees`
`WHERE officeCode IN (SELECT officeCode`
`FROM offices`
`WHERE country = 'ENGLAND');`

Subquery (계속)

- Subquery는 comparison 연산자와 함께 사용 가능
- e.g., salary가 가장 높은 직원의 성과 이름을 추출하라
 - **SELECT** lastName, firstName
FROM employees
WHERE salary = (**SELECT MAX**(salary)
FROM employees);

UNION 연산자

- 여러 개의 테이블을 합치는 연산자 (a.k.a., 합집합)
- 사용법
 - **SELECT** column1, column2 **FROM** table명 **UNION**
SELECT column1, column2 **FROM** table명;
 - UNION 연산자는 기본적으로 중복제거 기능이 있으며, 중복된 결과를 보고싶을 경우 UNION ALL 사용

GROUP BY절

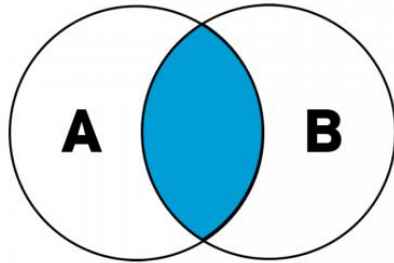
- 동일한 행들을 묶어서 그룹을 만드는데 사용
- GROUP BY는 Aggregate 함수(AVG, COUNT, SUM, MAX, MIN)와 함께 사용
 - i.e., 어떤 그룹의 평균값, 최대값 등
- 사용법
 - **SELECT** column_1, column_2,...,column_n
AGGREGATE_FUNCTION(expression)
FROM table
WHERE where_condition
GROUP BY column_1, column_2,...,column_n;
 - e.g., **SELECT** orders.status, **count**(orderNumber) **AS** count
FROM orders
GROUP BY orders.status
ORDER BY count desc;

HAVING 절

- **SELECT절의 WHERE처럼 조건을 제약하여 Row를 추출하기 위한 절**
 - 보통 GROUP BY와 함께 활용
 - WHERE절과의 차이점은 SELECT뒤에 오는 Aggregate Function이 적용된 column에 대하여 제약을 둘 수 있음
- **사용법**
 - **SELECT** column_1, column_2,...,column_n
FROM Table
GROUP BY column_1
HAVING column_2 > 1000;

INNER JOIN절

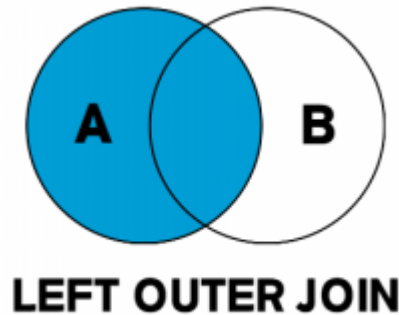
- Join절: 둘 이상의 테이블을 연결해서 데이터를 검색하는 방법
 - 연결하려는 테이블들이 적어도 하나의 column을 공유하는 것이 필요
- 여러 개의 테이블에서 교집합을 만족하는 Row들을 추출하는 절



- 사용법
 - **SELECT** column_1, column_2,...,column_n
FROM table_1
INNER JOIN table_2 **ON** join_condition_1
INNER JOIN table_3 **ON** join_condition_2
...
WHERE conditions;

LEFT JOIN절

- Join절: 둘 이상의 테이블을 연결해서 데이터를 검색하는 방법
 - 연결하려는 테이블들이 적어도 하나의 column을 공유하는 것이 필요
- 여러 개의 테이블에서 아래 그림을 만족하는 Row들을 추출하는 절



- 사용법
 - **SELECT** column_1, column_2,...,column_n
FROM table_1 A
LEFT JOIN table_2 B **ON** A.column_1 = B.column_1
WHERE conditions;

SELF JOIN절

- **Join절: 둘 이상의 테이블을 연결해서 데이터를 검색하는 방법**
 - 연결하려는 테이블들이 적어도 하나의 column을 공유하는 것이 필요
- **조인되어야 할 데이터들이 동일 테이블에 있을 경우 사용**
- **사용법**
 - **SELECT** column_1, column_2,...,column_n
FROM table_1 A1
LEFT JOIN table_1 A2 **ON** A.column_1 = A2.column_1
WHERE conditions;

JOIN 보충

S	A	B1
	1	가
	2	나
	3	다

T	B2	C
	나	#
	다	%
	라	\$
	마	@

조인

$$V \leftarrow S \bowtie_{B1=B2} T$$

V	A	B1	B2	C
2	가	나	나	#
3	다	다	다	%

왼쪽 외부조인

$$V \leftarrow S \ltimes_{B1=B2} T$$

V	A	B1	B2	C
1	가	가	null	null
2	나	나	나	#
3	다	다	다	%

오른쪽 외부조인

$$V \leftarrow S \bowtie_{B1=B2} T$$

V	A	B1	B2	C
2	가	나	나	#
3	다	다	다	%
null		null	라	\$
null		null	마	@

양쪽 외부조인

$$V \leftarrow S \ltimes_{B1=B2} T$$

V	A	B1	B2	C
1	가	가	null	null
2	나	나	나	#
3	다	다	다	%
null		null	라	\$
null		null	마	@

INSERT & UPDATE

■ INSERT문

- 데이터베이스 테이블에 데이터를 넣기 위한 문장

- 사용법

- **INSERT INTO** table(column_1, column_2,...) **VALUES** (value_1, value_2,...);

- 만약 테이블에 있는 column 전체에 데이터를 넣는 경우에는 column명을 나열할 필요 없음

■ UPDATE문

- 데이터베이스 테이블에 데이터를 갱신하기 문장

- 사용법

- **UPDATE** table **SET** column_1 = "new data", column_2="new data",...
WHERE update_condition;

UPDATE JOIN

- Join을 이용하여 양쪽 혹은 한쪽의 테이블을 갱신하기 위한 문장
- 사용법
 - **UPDATE** table_1, table_2,...
 [INNER JOIN | LEFT JOIN] table_1.column_1 = table_2.column_1
 SET table_1.column_2 = table_2.column_2
 WHERE update_condition;

DELETE

- 데이터베이스 테이블에서 데이터를 삭제 시키기 위한 문장
- 사용법
 - **DELETE FROM** table [**WHERE** delete_condition]
 - 예: **DELETE FROM** employees **WHERE** officeCode=4;
 - employees 테이블 안의 모든 행들을 지우기 위해서는 **DELETE FROM** employees;

DELETE CASCADE

- 데이터베이스 테이블에서 하나의 행이 삭제될 때, 그 행과 연관된 테이블의 데이터도 함께 삭제되도록 하기 위한 문장
 - 테이블 생성시 설정
- 예제
 - 아래 테이블에서 building이 하나 삭제되면 그 빌딩이 가지고 있는 room들도 함께 삭제되는 것이 필요함(building이 없으면 room이 없음)
 - 하나의 DELETE 연산이 다른 테이블에 영향을 미치는 경우

buildings		rooms
building_no INT(11)		room_no INT(11)
building_name VARCHAR(255)	←	room_name VARCHAR(255)
Address VARCHAR(355)		building_no VARCHAR(355)