



# 제9장

## 데이터베이스개요 및 개발환경 구축

# 1. 데이터 베이스의 정의와 특징

- 데이터베이스
  - 여러 사용자나 응용 프로그램이 공유하고 동시에 접근 가능한 '데이터의 집합'
  - '데이터의 저장 공간' 자체를 의미하기도 함
- DBMS(DataBase Management System)
  - 데이터베이스를 관리·운영하는 소프트웨어
  - 사용자나 응용 프로그램은 DBMS가 관리하는 데이터에 동시에 접속하여 데이터를 공유함

표 1-1 많이 사용되는 DBMS(2018년 기준)

DBMS	제작사	운영체제	최신 버전	비고
MySQL	오라클	유닉스, 리눅스, 윈도우, 맥	8.0	오픈 소스(무료), 상용
MariaDB	마리아DB	유닉스, 리눅스, 윈도우	10.3	오픈 소스(무료)
PostgreSQL	PostgreSQL	유닉스, 리눅스, 윈도우, 맥	10.4	오픈 소스(무료)
Oracle	오라클	유닉스, 리눅스, 윈도우	18c	상용 시장 점유율 1위
SQL Server	마이크로소프트	리눅스, 윈도우	2017	
DB2	IBM	유닉스, 리눅스, 윈도우	10	메인프레임 시장 점유율 1위
Access	마이크로소프트	윈도우	2017	PC용
SQLite	SQLite	안드로이드, iOS	3.24	모바일 전용, 오픈 소스(무료)

# 1. 데이터 베이스의 정의와 특징

## ■ 데이터베이스 개념도

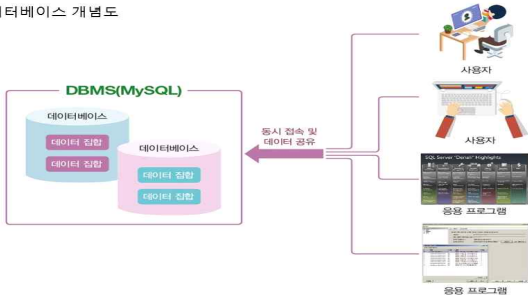


그림 1-1 DBMS의 개념

# 1. 데이터 베이스의 정의와 특징

## ■ 데이터베이스의 특징

- **데이터의 무결성:** 데이터베이스 안의 데이터는 어떤 경로를 통해 들어왔든 오류가 있어서는 안 됨
- **데이터의 독립성:** 데이터베이스와 응용 프로그램은 서로 의존적인 관계가 아니라 독립적인 관계임
- **보안:** 데이터베이스 안의 데이터는 데이터를 소유한 사람이나 데이터에 접근이 허가된 사람만 접근할 수 있음
- **데이터 중복 최소화:** 데이터베이스에서는 동일한 데이터가 여러 군데 중복 저장되는 것을 방지함
- **응용 프로그램 제작 및 수정 용이:** 데이터베이스를 이용하면 통일된 방식으로 응용 프로그램을 작성할 수 있고 유지·보수 또한 쉬움
- **데이터의 안전성 향상:** 데이터가 손상되는 문제가 발생하더라도 원래의 상태로 복원 또는 복구할 수 있음

## 2. DBMS의 분류

### ■ 계층형 DBMS

- 각 계층이 트리 형태를 띠고 1:N 관계를 가짐
- 한번 구축하면 구조를 변경하기 까다로움
- 접근의 유연성이 부족하여 임의 검색 시 어려움

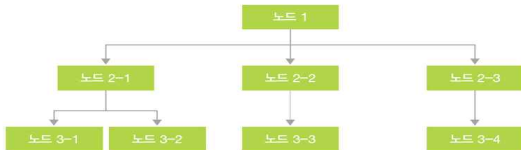


그림 1-2 계층형 구조

## 2. DBMS의 분류

### ■ 망형 DBMS

- 1:1, 1:N, N:M(다대다) 관계가 지원되어 효과적이고 빠른 데이터 추출이 가능
- 매우 복잡한 내부 포인터 사용
- 프로그래머가 모든 구조를 이해해야 만 프로그램을 작성할 수 있음

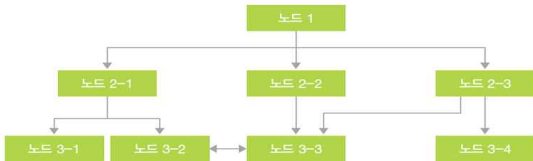


그림 1-3 망형 구조

## 2. DBMS의 분류

### ■ 관계형 DBMS

- 모든 데이터는 테이블에 저장
- 테이블 간의 관계는 기본키(PK)와 외래키(FK)를 사용하여 맺음(부모-자식 관계)
- 다른 DBMS에 비해 업무 변화에 따라 바로 순응할 수 있고 유지·보수 측면에서도 편리
- 대용량 데이터를 체계적으로 관리할 수 있음
- 데이터의 무결성도 잘 보장됨
- 시스템 자원을 많이 차지하여 시스템이 전반적으로 느려지는 단점이 있음



그림 1-4 관계형 구조

# 3. SQL의 개요

- SQL
  - 데이터베이스를 조작하는 언어
- SQL의 특징
  - DBMS 제작 회사와 독립적임
  - 다른 시스템으로의 이식성이 좋음
  - 표준이 계속 발전함
  - 대화식 언어임
  - 클라이언트/서버 구조 지원함
- 표준 SQL과 각 회사의 SQL
  - 많은 회사가 되도록 표준 SQL을 준수하려고 노력하지만 각 회사의 DBMS마다 특징이 있기 때문에 현실적으로 완전히 통일되기는 어려움
  - 각 회사의 제품은 모두 표준 SQL을 공통으로 사용하면서 자기 제품의 특성에 맞춘 호환되지 않는 SQL 문 사용

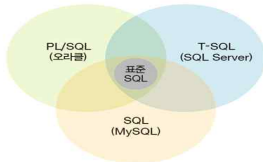


그림 1-5 표준 SQL과 각 회사의 SQL



## 4. MySQL의 개요

### ■ MySQL

- 오라클에서 제작한 DBMS 소프트웨어
- 오픈소스로 제공됨

표 1-2 MySQL의 변천사

시기	버전	기타
1994년		마이클 와이드니우스, 데이비드 액스마크가 개발
1995년 5월		최초 국제 버전 공개
1996년 말	3.19	
1997년 1월	3.20	
1998년 1월		윈도우용 릴리스
1998년	3.21	상용 제품 릴리스, <a href="http://www.mysql.com">www.mysql.com</a> 사이트 구축
2003년 3월	4.0	
2004년 10월	4.1	R-트리, B-트리, 서브쿼리 등 지원
2005년 10월	5.0	커서, 스토어드 프로시저, 트리거, 뷰 등 지원
2008년		선마이크로시스템에 인수됨
2008년 11월	5.1	파티셔닝, 복제 등 지원
2010년 1월		오라클이 선마이크로시스템을 인수
2010년 12월	5.5	InnoDB를 기본 엔진으로 사용하는 등 큰 변화가 있었음
2013년 2월	5.6	성능 개선 NoSQL 지원 등
2015년 10월	5.7	보안 강화, InnoDB 확장, JSON 지원 등
2018년 4월	8.0	성능 향상, 보안 강화, 장애 대비 능력 강화 등

## 5. MySQL 에디션

- 상용 에디션
  - Standard < Enterprise < Cluster CGE 순으로 비싸거나 기능이 높음
- 무료 에디션
  - Community 에디션은 Enterprise 에디션과 기능상 차이가 거의 없고 사용 허가에 대한 라이선스 차이가 있음
  - 이 수업에서는 무료 에디션인 Community 에디션 사용

## 6. MySQL 설치 전 준비 사항

### ■ 내 컴퓨터 운영체제 확인

표 1-3 MySQL Community 8.0을 설치하기 위한 운영체제 스펙

서버 운영체제(64bit)	PC 운영체제(64bit)
Windows Server 2019	윈도우 10
Windows Server 2016	윈도우 7(공식적으로는 지원하지 않음)
Windows Server 2012 R2	

### ■ 설치 프로그램 소개(초록색으로 표시된 것만 설치)

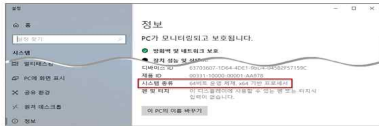
표 1-4 MySQL 관련 소프트웨어와 각 소프트웨어를 사용하기 위한 필수 요구 사항

관련 소프트웨어	요구 사항	요구 사항 설치	비고
MySQL Server	Visual C++ 2015 Runtime	자동	서버 프로그램
MySQL Client			클라이언트 프로그램
MySQL Workbench	Visual C++ 2015 Runtime	자동	GUI 지원 통합 관리 툴
Sample Database			샘플 데이터베이스
MySQL Notifier			서비스 알림 기능

# 7. MySQL 설치하기

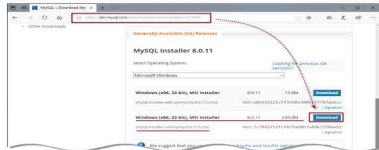
## 1 내 컴퓨터의 운영체제가 64비트인지 확인하기

### 1-1 시스템의 종류 확인



## 2 MySQL Community 다운로드하기

### 2-1 MySQL Community 8.0 다운로드 1



# 7. MySQL 설치하기

## 2-2 MySQL Community 8.0 다운로드 2

전속사이트 : <https://dev.mysql.com/downloads/installer/>



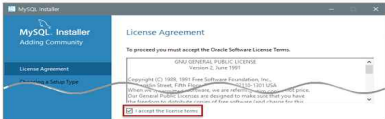
## 3 다운로드한 MySQL 설치하기

### 3-1 MySQL Community 8.0 설치 1



# 7. MySQL 설치하기

## 3-2 MySQL Community 8.0 설치 2



## 3-3 MySQL Community 8.0 설치 3



# 7. MySQL 설치하기

## 3-4 MySQL Community 8.0 설치 4

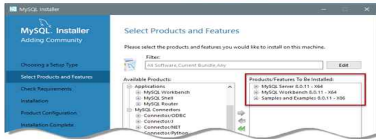


## 3-5 다음 2개 항목 추가

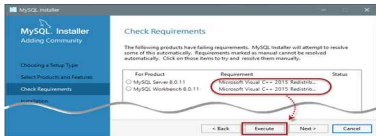
- ① [Applications]-[MySQL Workbench]-[MySQL Workbench 8.0]-[MySQL Workbench 8.0.11 - X64]
- ② [Documentation]-[Samples and Examples]-[Samples and Examples 8.0]-[Samples and Examples 8.0.11 - X86]

# 7. MySQL 설치하기

## 3-6 MySQL Community 8.0 설치 5



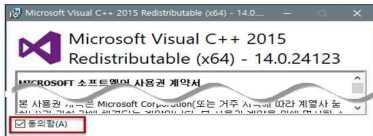
## 3-7 MySQL Community 8.0 설치 6



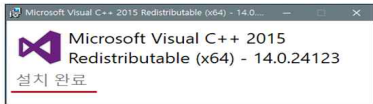


# 7. MySQL 설치하기

3-8 MySQL Community 8.0 설치 7



3-9 MySQL Community 8.0 설치 8



# 7. MySQL 설치하기

## 3-10 MySQL Community 8.0 설치 9



## 3-11 MySQL Community 8.0 설치 10



# 7. MySQL 설치하기

## 3-12 MySQL Community 8.0 설치 11



## 4 MySQL 환경 설정하기

### 4-1 MySQL Community 8.0 환경 설정 1

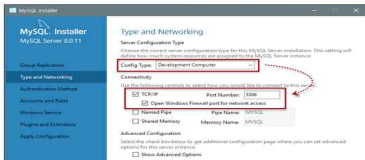


# 7. MySQL 설치하기

## 4-2 MySQL Community 8.0 환경 설정 2



## 4-3 MySQL Community 8.0 환경 설정 3



# 7. MySQL 설치하기

## 4-4 MySQL Community 8.0 환경 설정 4



## 4-5 MySQL Community 8.0 환경 설정 5



# 7. MySQL 설치하기

## 4-6 MySQL Community 8.0 환경 설정 6

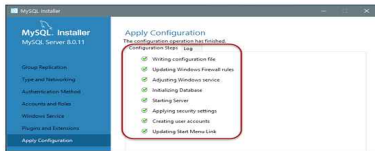


## 4-7 MySQL Community 8.0 환경 설정 7



# 7. MySQL 설치하기

## 4-8 MySQL Community 8.0 환경 설정 8

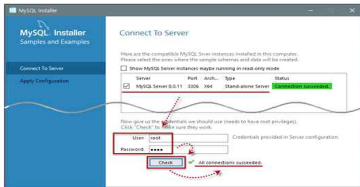


## 4-9 MySQL Community 8.0 환경 설정 9



# 7. MySQL 설치하기

## 4-10 MySQL Community 8.0 환경 설정 10



## 4-11 MySQL Community 8.0 환경 설정 11





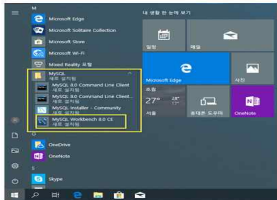
# 7. MySQL 설치하기

4-12 [Product Configuration] 창이 나오면 <Next> 클릭

4-13 MySQL Community 8.0 환경 설정 12



4-14 MySQL 등록 확인



# 7. MySQL 설치하기

## 5 MySQL에 접속하기

5-1 윈도우의 [시작]-[MySQL]-[MySQL Workbench 8.0 CE] 선택

5-2 MySQL Server에 연결

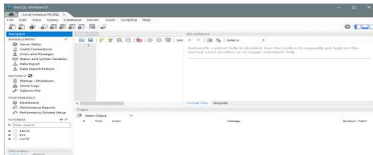


5-3 MySQL Server에 접속

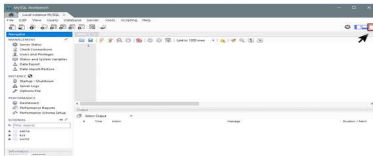


## 7. MySQL 설치하기

#### 5-4 MySQL Workbench 화면 확인



### 5-5 MySQL Workbench 화면 확인



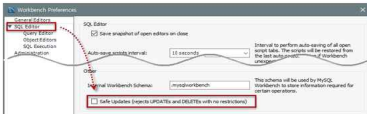
# 7. MySQL 설치하기

## 6 MySQL Workbench 설정 확인하기

### 6-1 메뉴에서 [Edit]-[Preferences] 선택



### 6-2 'Safe Updates'의 체크 해제



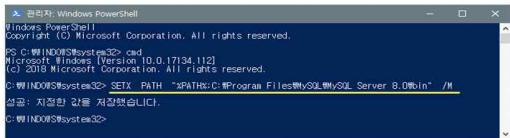
### 6-3 MySQL Workbench 닫기

# 7. MySQL 설치하기

7 MySQL의 실행 파일이 있는 경로를 Path에 추가하기

7-1 관리자 권한으로 파워셸 실행

7-2 C:\Program Files\MySQL\MySQL Server 8.0\bin 폴더를 Path에 추가



```
관리자: Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS C:\WINDOWS\system32> cmd
Microsoft Windows [Version 10.0.17134.112]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\WINDOWS\system32> SETX PATH "%PATH%;C:\Program Files\MySQL\MySQL Server 8.0\bin" /M
성공: 지정된 값을 저장했습니다.

C:\WINDOWS\system32>
```

7-3 shutdown -r -t 0 명령으로 컴퓨터 재부팅

## 8. MySQL 기본적 쿼리문 익히기

### ❖ <SELECT... FROM>

- 원하는 데이터를 가져와 주는 기본적인 구문
- 가장 많이 사용되는 구문
- 데이터베이스 내 테이블에서 원하는 정보 추출하는 기능

## 8. MySQL 기본적 쿼리문 익히기

### ❖ SELECT의 구문 형식

- 복잡한 형식을 실제 사용되는 형태로 요약

```
SELECT select_expr  
  [FROM table_references]  
  [WHERE where_condition]  
  [GROUP BY {col_name | expr | position}]  
  [HAVING where_condition]  
  [ORDER BY {col_name | expr | position}]
```



```
SELECT 열이름  
FROM 테이블이름  
WHERE 조건
```

## 8. MySQL 기본적 쿼리문 익히기

### ❖ USE 구문

- **SELECT문 학습 위해 사용할 데이터베이스 지정**
- 지정해 놓은 후 특별히 다시 USE문 사용하거나 다른 DB를 사용하겠다고 명시하지 않는 이상 모든 SQL문은 지정 DB에서 수행

```
USE 데이터베이스_이름;
```

- **Workbench 에서 직접 선택해서 사용도 가능**
  - [Navigator]의 [Schemas] 탭
    - **employees 데이터베이스를 더블 클릭하면 진한 글자 전환**
    - 왼쪽 아래 'Active schema changed to employees' 메시지



## 8. MySQL 기본적 쿼리문 익히기

### ❖ SELECT와 FROM

#### ■ SELECT \*

- 선택된 DB가 employees 라면 다음 두 쿼리는 동일

```
SELECT * FROM employees.titles;  
SELECT * FROM titles;
```

#### ■ SELECT 열 이름

- 테이블에서 필요로 하는 열만 가져오기 가능
- 여러 개의 열을 가져오고 싶을 때는 콤마로 구분
- 열 이름의 순서는 출력하고 싶은 순서대로 배열 가능

## 8. MySQL 기본적 쿼리문 익히기

❖ DB, TABLE, 열의 이름이 확실하지 않을 때 조회하는 법

- 현재 서버에 어떤 DB가 있는지 보기

- SHOW DATABASES;

- 현재 서버에 어떤 TABLE이 있는지 보기

- 데이터베이스 employees 에 있는 테이블 정보 조회

- SHOW TABLE STATUS;

- 테이블 이름만 간단히 보기

- SHOW TABLES;

- employees 테이블의 열이 무엇이 있는지 확인

- DESCRIBE employees;

- DESC employees;

## 8. MySQL 기본적 쿼리문 익히기

### ❖ 문제가 생긴 DB 초기화하기

#### ■ DB가 존재한다면 지우고 다시 만들기

```
DROP DATABASE IF EXISTS sqlDB; -- 만약 sqlDB가 존재하면 우선 삭제한다.  
CREATE DATABASE sqlDB;
```

- 계속 사용할 쿼리는 SQL 파일로 저장해서 재사용 가능하게 만들기
- 파일 내용을 불러다 쓰기 전에 모든 쿼리 창을 닫도록 함

#### ■ DB 내용을 입력하는 예제(실습)

## 8. MySQL 기본적 쿼리문 익히기

### ❖ 특정 조건의 데이터만 조회 - <SELECT FROM WHERE>

#### ■ 기본적인 **WHERE**절

- 조회하는 결과에 특정한 조건 줘서 원하는 데이터만 보고 싶을 때 사용
- SELECT 필드이름 FROM 테이블이름 WHERE 조건식;
- 조건이 없을 경우 테이블의 크기가 클수록 찾는 시간과 노력이 증가

#### ■ 관계 연산자의 사용

- '...했거나', '... 또는' - **OR** 연산자
- '...하고', '...면서', '... 그리고' - **AND** 연산자
- 조건 연산자(=, <, >, <=, >=, <>, != 등)와 관계 연산자(NOT, AND, OR 등)의 조합으로 알맞은 데이터를 효율적으로 추출

# 8. MySQL 기본적 쿼리문 익히기

## ❖ 테이블 관련 명령어

### ■ 테이블 생성하기

```
CREATE TABLE 테이블 이름(필드 이름1 자료형1[, 필드 이름2 자료형2, ...]);
```

[CREATE 명령어 사용 예]

```
CREATE TABLE member(  
  id int NOT NULL AUTO_INCREMENT,  
  
  name VARCHAR(100) not null,  
  passwd VARCHAR(50) not null,  
  PRIMARY KEY (id)  
);
```

## 8. MySQL 기본적 쿼리문 익히기

### ❖ 테이블 관련 명령어

#### ■ 테이블 구조 변경하기

```
ALTER TABLE 테이블 이름  
[ADD 필드 이름 자료형 |  
DROP COLUMN 필드 이름 |  
CHANGE COLUMN 기존 필드 이름 새 필드 이름 자료형];
```

- 기존 테이블에 열을 추가하는 경우: 기존 테이블 이름이 Member이고 여기에 새로운 열 이름 phone을 추가하는 경우는 다음과 같습니다.

```
ALTER TABLE Member ADD phone varchar(100);
```

## 8. MySQL 기본적 쿼리문 익히기

- INSERT 문

- 테이블에 데이터를 삽입하는 명령어

```
INSERT [INTO] 테이블이름[(열1, 열2, ...)] VALUES (값1, 값2, ...)
```

- INSERT 문에서 테이블 이름 다음에 나오는 열 생략 가능(단 열의 순서 및 개수는 동일해야 함)

```
USE sqldb;  
CREATE TABLE testTBL1 (id int, userName char(3), age int);  
INSERT INTO testTBL1 VALUES (1, '뿌로로', 16);
```

- id와 이름만 입력하고 나이는 입력하고 싶지 않다면

```
INSERT INTO testTBL1 (id, userName) VALUES (2, '크롱');
```

- 열의 순서를 바꾸어 입력하고 싶을 때

```
INSERT INTO testTBL1 (userName, age, id) VALUES ('루피', 14, 3);
```

## 8. MySQL 기본적 쿼리문 익히기

### ■ AUTO\_INCREMENT 키워드

- 자동으로 1부터 증가하는 값을 입력하는 키워드
- 특정 열을 AUTO\_INCREMENT로 지정할 때는 반드시 **PRIMARY KEY(기본키)** 또는 **UNIQUE(유일한 값)**로 설정해야 함
- 데이터 형식이 숫자인 열에만 사용 가능
- AUTO\_INCREMENT로 지정된 열은 INSERT 문에서 NULL 값으로 지정하면 자동으로 값이 입력됨

```
USE sqldb;
CREATE TABLE testTBL2
( id int AUTO_INCREMENT PRIMARY KEY,
  userName char(3),
  age int
);
INSERT INTO testTBL2 VALUES (NULL, '에디', 15);
INSERT INTO testTBL2 VALUES (NULL, '포비', 12);
INSERT INTO testTBL2 VALUES (NULL, '통통이', 11);
SELECT * FROM testTBL2;
```

	id	userName	age
▶	1	에디	15
	2	포비	12
	3	통통이	11
•	<b>NULL</b>	<b>NULL</b>	<b>NULL</b>

### ■ AUTO\_INCREMENT 입력 값을 100부터 시작하도록 변경하고 싶다면

```
ALTER TABLE testTBL2 AUTO_INCREMENT=100;
INSERT INTO testTBL2 VALUES (NULL, '패티', 13);
SELECT * FROM testTBL2;
```

	id	userName	age
▶	1	에디	15
	2	포비	12
	3	통통이	11
•	<b>100</b>	<b>패티</b>	<b>13</b>



## 8. MySQL 기본적 쿼리문 익히기

- **UPDATE 문**

- 테이블에 입력되어 있는 값을 수정하는 명령어

```
UPDATE 테이블이름  
SET 열1=값1, 열2=값2, ...  
WHERE 조건;
```

- 'Kyoichi'의 Lname을 '없음'으로 수정

```
USE sqldb;  
UPDATE testTBL4  
SET Lname = '없음'  
WHERE Fname = 'Kyoichi';
```

- 전체 테이블의 내용을 수정하고 싶을 때는 WHERE 절 생략

```
UPDATE buyTBL  
SET price = price * 1.5;
```

## 8. MySQL 기본적 쿼리문 익히기

- **DELETE 문**

- 테이블에 데이터를 행 단위로 삭제하는 명령어

```
DELETE FROM 테이블이름 WHERE 조건;
```

- **DELETE 문에서 WHERE 절을 생략하면 테이블에 저장된 전체 데이터가 삭제**

```
USE cookDB;  
DELETE FROM testTBL4 WHERE Fname = 'Aamer';
```

## 9. MySQL 기본적 쿼리문 익히기

- 데이터베이스 생성하기: 워크벤치로 MySQL에 접속하여 TestDB와 WebStoreDB라는 데이터베이스를 생성하도록 하자.
- 실습에 사용할 member테이블을 아래와 같이 생성하자.

```
drop table if exists member;  
CREATE TABLE member (  
    id varchar(20) not null primary key,  
    pw varchar(20) not null,  
    name varchar(30) not null,  
    phone varchar(30)  
);
```

감사합니다.

