
Database

컴퓨터개론

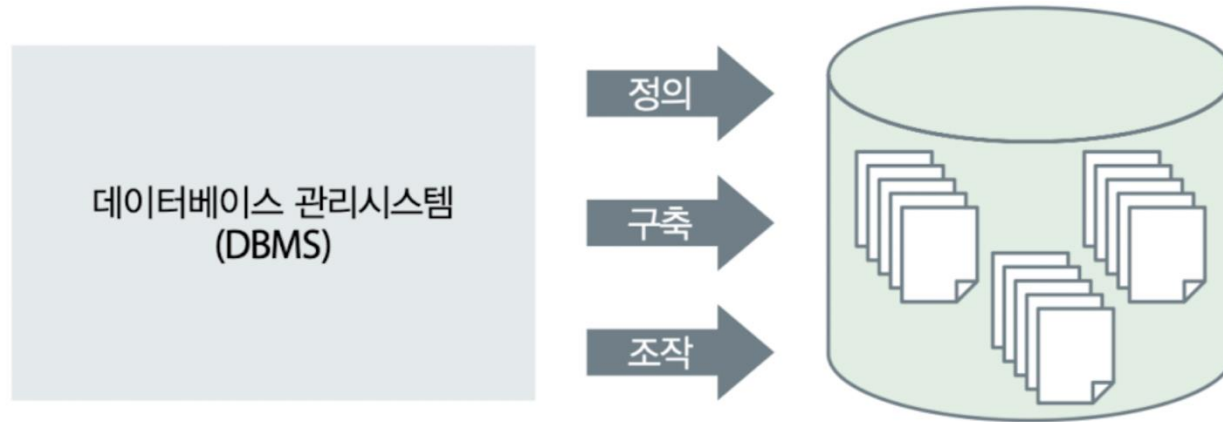
(Introduction to Computer Systems)

GEN1030

데이터베이스 관리시스템

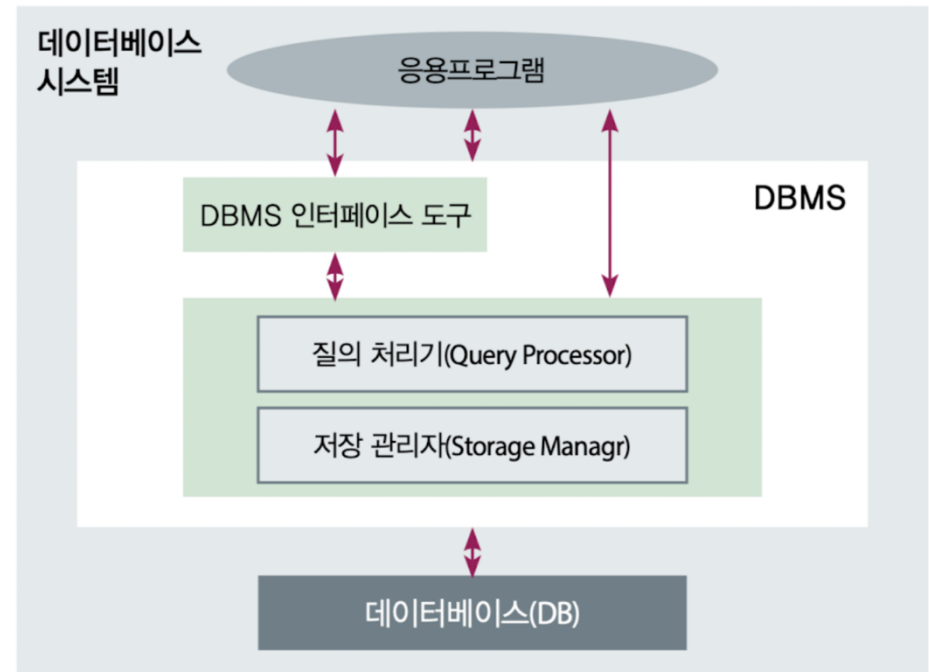
데이터베이스 관리시스템

- DataBase Management System (DBMS)
 - 데이터베이스를 정의/구축/조작/제어하여 데이터베이스에서 정보를 쉽게 활용할 수 있도록 만든 프로그램



DBMS 구성

- DBMS interface tool
 - 사용자가 데이터베이스 자료와 장치에 쉽게 접근할 수 있는 수단 제공
- Query processor
 - 사용자/응용프로그램이 요청하는 질의문(Query) 해석 및 결과 도출
- Storage manager
 - 실제 저장장치에 자료 저장



DBMS 종류

- 관계형 DBMS
 - MySQL
 - SQLite
 - MariaDB
 - Sybase
 - Oracle
 - Microsoft SQL Server
 - ...
- NoSQL DBMS
 - MongoDB (Document DB)
 - Redis (Key-Value DB)
 - Cassandra (Column-family DB)
 - Neo4j (Graph DB)
 - ...

DBMS 종류

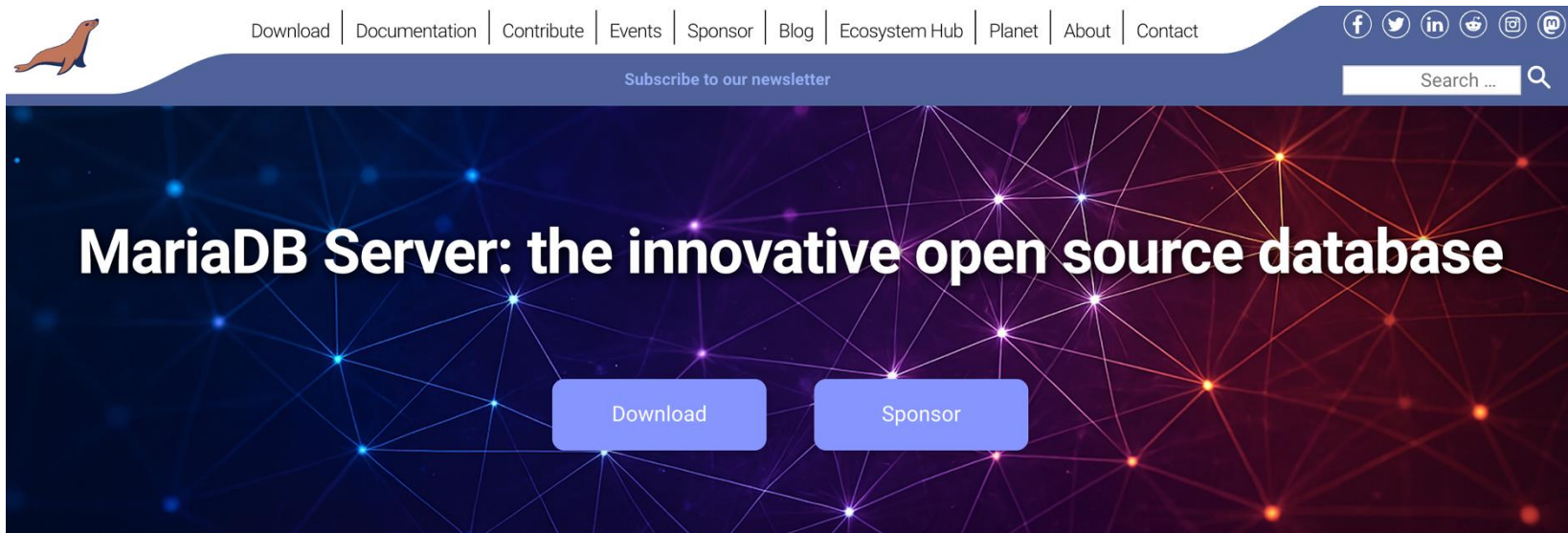
- MySQL

- 대표적인 무료 오픈 소스 관계형 DBMS (Relational DBMS, RDBMS)
- Oracle이 현재는 소유

The screenshot shows the MySQL website homepage. At the top left is the MySQL logo with the tagline "The world's most popular open source database". Below the logo is a navigation menu with links for "MYSQL.COM", "DOWNLOADS", "DOCUMENTATION", and "DEVELOPER ZONE". A secondary navigation bar contains links for "Products", "Services", "Partners", "Customers", "Why MySQL?", "News & Events", and "How to Buy". The main content area features a large banner for "MySQL 9.7 LTS" with the text "The Stability of a Long-Term Support Release, the Speed of Innovation". A circular badge on the right indicates "LTS LONG-TERM SUPPORT". Below the banner are four key features: "NOW LTS Long-term stability you can count on", "BUILT FOR INNOVATION Faster, smarter, more efficient", "STRONGER TOGETHER New community features and contributions", and "ENTERPRISE POWER New Data Distribution Management (DDM)". A "Read the Blog Announcement" button is at the bottom left of the banner.

DBMS 종류

- MariaDB
 - 오픈 소스 관계형 데이터베이스
 - MySQL과 사용방법 및 구조가 유사

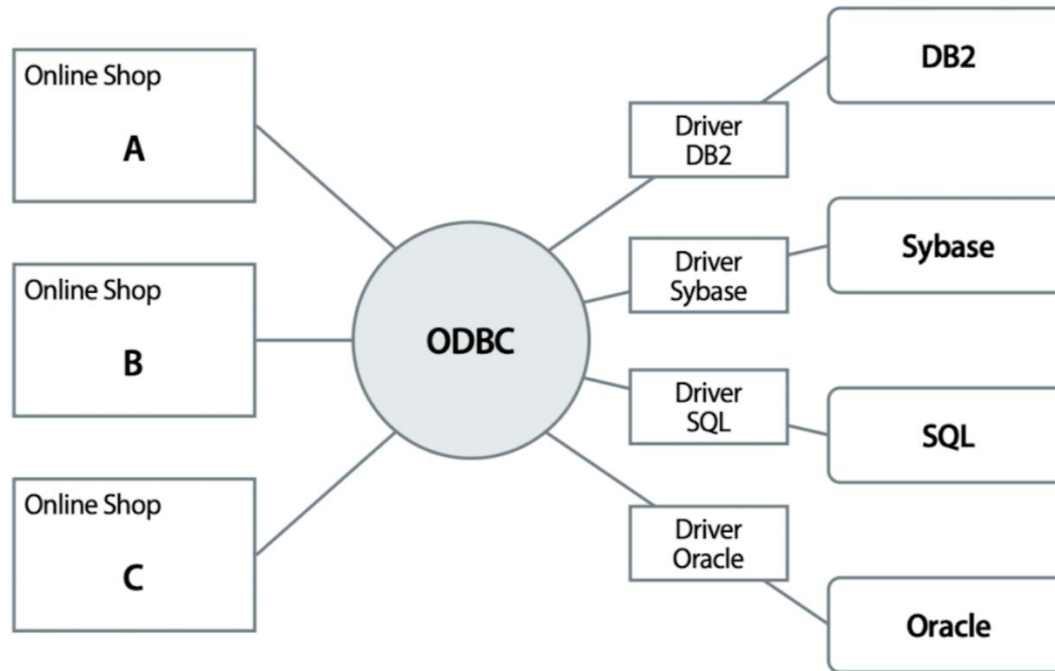


DBMS 종류

- Oracle
 - 1977년 설립
 - 대표적인 기업용 관계형 DBMS
 - 기업용 버전 유료
 - 무료 버전 존재하지만 제한적
- Microsoft SQL Server
 - 서버용 컴퓨터에 널리 사용되는 관계형 DBMS
 - 무료 버전, 유료 버전 모두 존재
 - ODBC (Open DataBase Connectivity)

ODBC (Open DataBase Connectivity)

- DBMS 종류에 관계없이 어떤 응용프로그램에서나 모두 접근해 사용할 수 있도록 하기 위한 DB 표준 interface
 - 여러 종류의 데이터베이스 함께 사용 가능



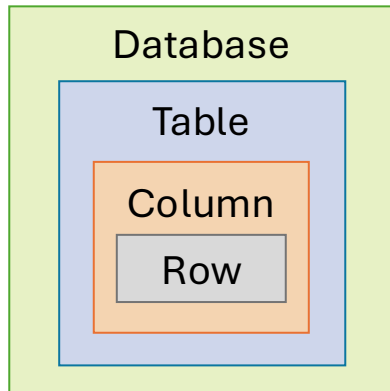
- 기존 데이터베이스 교체하더라도 응용프로그램은 변경 없이 기능 그대로 사용 가능

DBMS 종류

- MongoDB

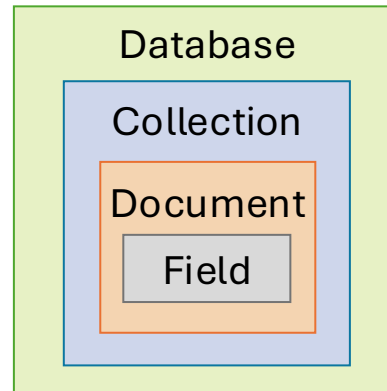
- Document 데이터베이스
- NoSQL 중 널리 사용되는 DBMS 중 하나
- Mongo: humongous 줄인 말 (매우 큰)
- 주로 JSON 형식으로 데이터 저장

관계형 데이터베이스



id	name	major
1	Kim	CS
2	Park	EE

NoSQL 데이터베이스



```
[  
  {  
    "id": 1,  
    "name": "Kim",  
    "major": "CS"  
  },  
  {  
    "id": 2,  
    "name": "Lee",  
    "major": "EE"  
  }  
]
```

Embedded DBMS

- 라이브러리와 같은 형태로 프로그램 자체에 통합되어 동작하는 경량 DBMS
 - cf. 일반적인 DBMS는 별도의 데이터베이스 서버가 작동하고 응용 프로그램이 서버에 접근하는 방식
 - 스마트폰 등 다양한 모바일 기기 보급으로 사용 증가
- SQLite
 - 대표적인 오픈 소스 임베디드 관계형 데이터베이스
 - 하나의 (로컬) 파일에 모든 데이터베이스 정보가 저장되
응용프로그램이 이를 사용
 - C로 구현되어 용량이 작고 가벼움
 - 스마트폰(안드로이드/아이폰) 등에서 사용

Structured Query Language

- 구조적 질의 언어(Structured Query Language, SQL)
 - 관계형 데이터베이스에서 데이터 정의/조작/제어 위해 사용하는 언어
- 목적에 따라 명령어 카테고리가 나뉨
 - Data Definition Language (DDL)
 - Data Query Language (DQL)
 - Data Manipulation Language (DML)
 - Transaction Control Language (TCL)
 - Data Control Language (DCL)

Structured Query Language

- 구조적 질의 언어(Structured Query Language, SQL)
 - 관계형 데이터베이스에서 데이터 정의/조작/제어 위해 사용하는 언어
- 목적에 따라 명령어 카테고리가 나뉨

명령어 분류	명령어	내용
DDL 데이터 정의어 (Data Definition Language)	create	테이블 등의 객체 생성
	alter	객체 변경
	drop	객체 삭제
	rename	객체 이름 변경
DQL 데이터 질의어 (Data Query Language)	select	데이터 조회
DML 데이터 조작어 (Data Manipulation Language)	insert	데이터 입력
	update	데이터 수정
	delete	데이터 삭제
TCL 트랜잭션 처리어 Transaction Control Language	commit	트랜잭션 정상적인 처리
	rollback	트랜잭션 취소
DCL 데이터 제어어 Data Control Language	grant	데이터베이스의 권한 부여
	revoke	데이터베이스의 권한 취소

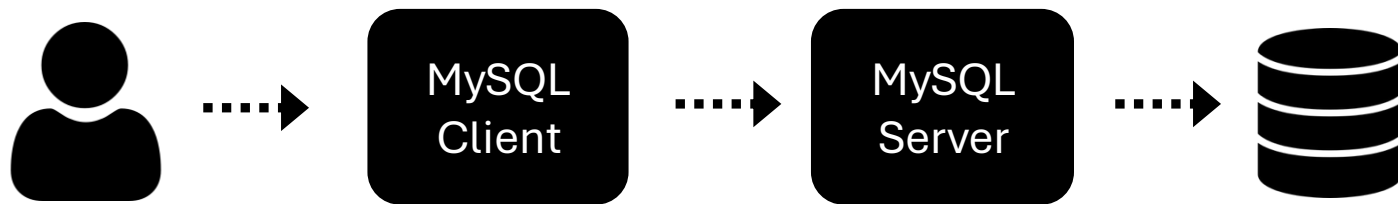
Example: MySQL

MySQL

- 대표적인 관계형 DBMS
- 사용
 - MySQL server 실행
 - MySQL client 접속
 - Database 생성/선택
 - SQL 명령어 실행



- 실제 Database 저장 및 관리
- 사용자의 SQL 명령 처리



- 사용자가 SQL 명령 입력
- 명령을 Server로 보냄

MySQL 사용

- MySQL 접속
 - `mysql -u (사용자) -p`
-u: 사용자 지정 -p: 비밀번호 입력 사용

MySQL Query

- 데이터베이스

- SHOW DATABASES; # 현재 서버에 있는 데이터베이스 목록 보여줌
- CREATE DATABASE 데이터베이스_이름; # 새 데이터베이스 생성
- USE 데이터베이스_이름; # 데이터베이스 선택

- 테이블

- SHOW TABLES; # 테이블 목록 보여줌
- CREATE TABLE 테이블_이름 (열_이름 데이터타입 ...) # 테이블 생성
- DESCRIBE 테이블_이름; # 테이블 구조 확인
- DROP TABLE 테이블_이름; # 테이블 삭제

MySQL Query

- 데이터

- INSERT INTO 테이블_이름 VALUES (데이터, ...) # 테이블에 데이터 추가
- SELECT * FROM 테이블_이름; # 테이블의 모든 행/열 조회
- SELECT 열_이름 FROM 테이블_이름; # 특정 열만 조회
- UPDATE 테이블_이름 SET ... WHERE 조건; # 기존 데이터 수정
- DELETE FROM ... WHERE 조건; # 기존 데이터 삭제

MySQL Example

DB 생성

```
CREATE DATABASE schoolDB;  
USE schoolDB;
```

학과 테이블 생성

```
CREATE TABLE departments (  
    dept_id INT PRIMARY KEY,  
    dept_name VARCHAR(50) NOT NULL  
);
```

데이터 추가

```
INSERT INTO departments VALUES (1, 'CS');  
INSERT INTO departments VALUES (2, 'EE');
```

데이터 조회

```
SELECT * FROM departments;
```

MySQL Example

학생 테이블 생성

```
CREATE TABLE students (  
    student_id INT PRIMARY KEY,  
    name VARCHAR(50) NOT NULL,  
    age INT,  
    dept_id INT,  
    FOREIGN KEY (dept_id) REFERENCES departments(dept_id)  
);
```

데이터 추가

```
INSERT INTO students VALUES (1, 'Kim', 20, NULL);  
INSERT INTO students VALUES (2, 'Park', 21, 1);
```

데이터 조회

```
SELECT student_id, name, age FROM students;
```

데이터 수정

```
UPDATE students SET age = 23 WHERE student_id = 1;
```

데이터 삭제

```
DELETE FROM students WHERE name = 'kim';
```

Summary

- DataBase Management System (DBMS)
 - 구성
 - 종류
- MySQL example