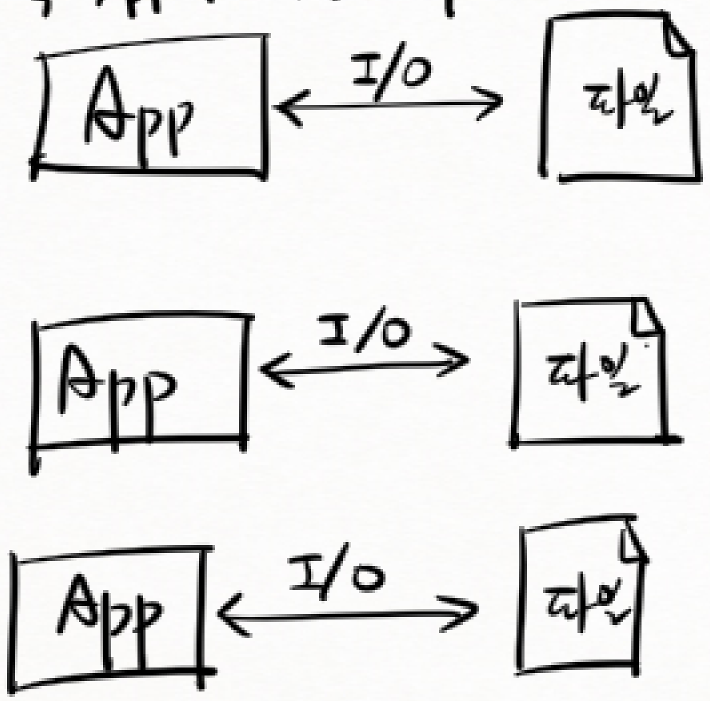
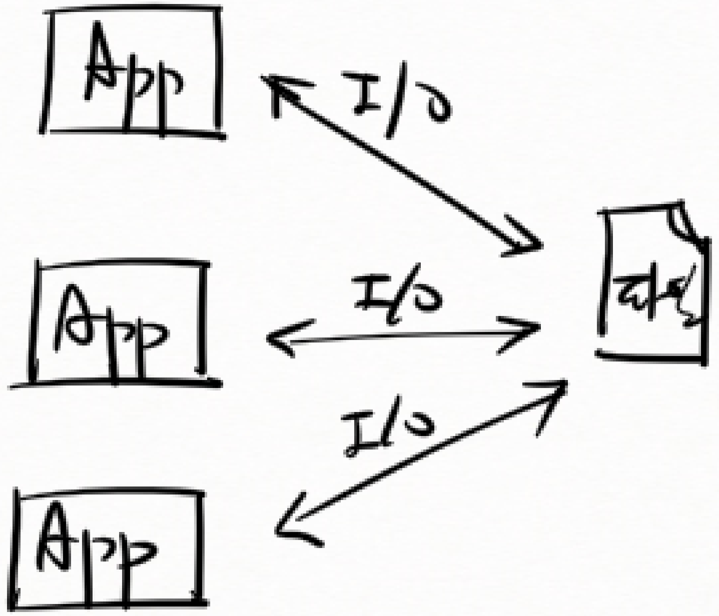
* 파일관리자 등장

① 각 App이 파일 관리



* App 끼리
데이터 공유가
힘들다.

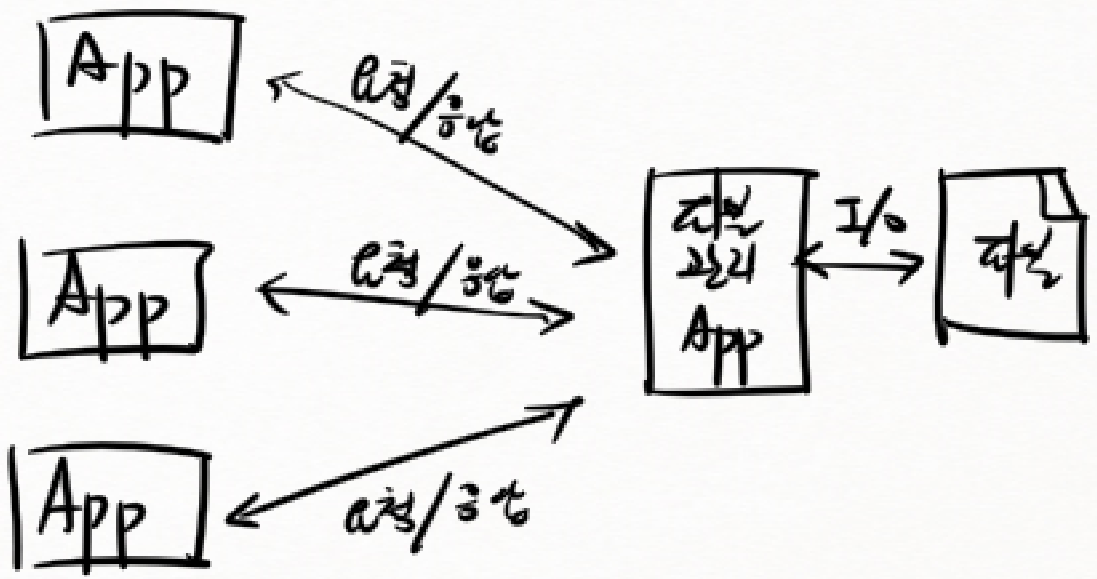
② App 끼리 파일 공유



* 다른 App의 데이터를
덮어 쓸 수 있다.

접근 권한 제어 곤란

③ 파일 관리자 도입

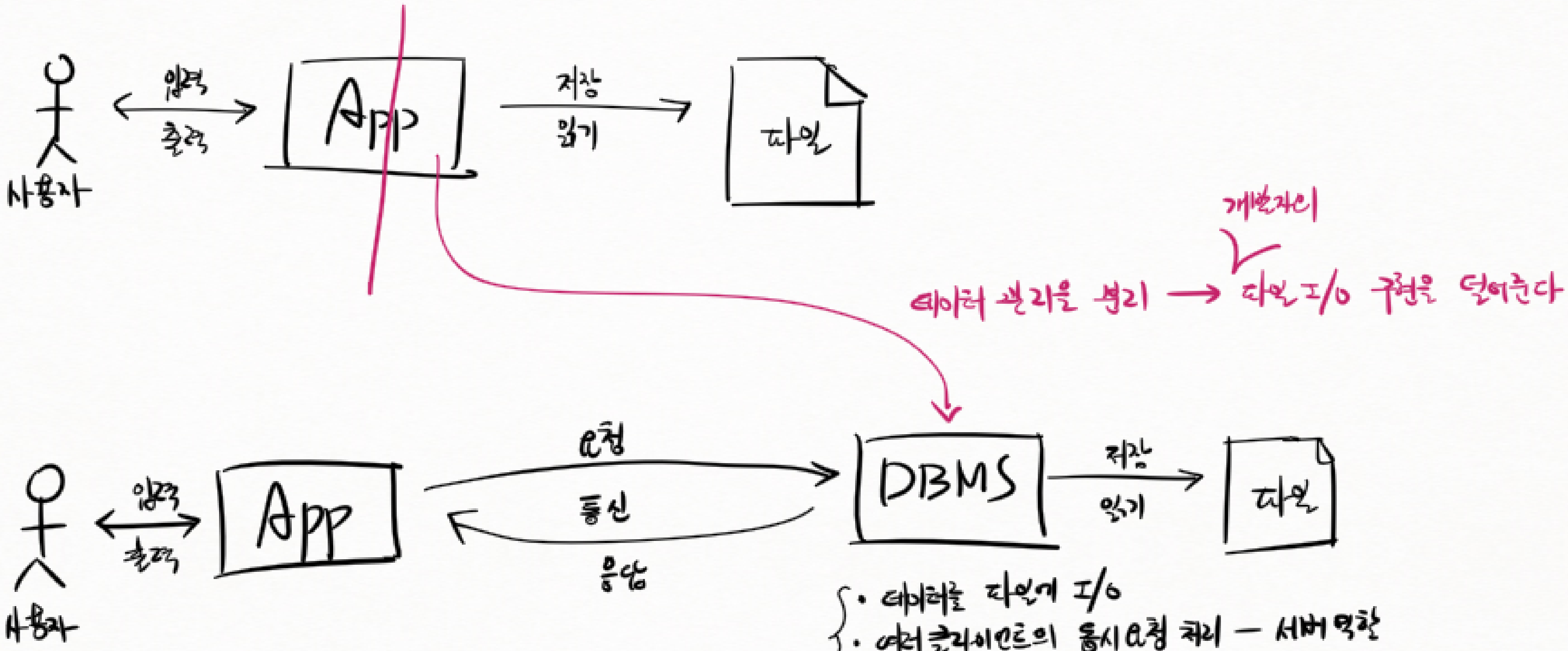


* App 개발을 할 때마다
파일관리 App도 개발해야 한다

⇓ 문제점

- 중복 개발
- 네트워킹 프로그래밍
- 접근 권한 제어

* DBMS의 등장

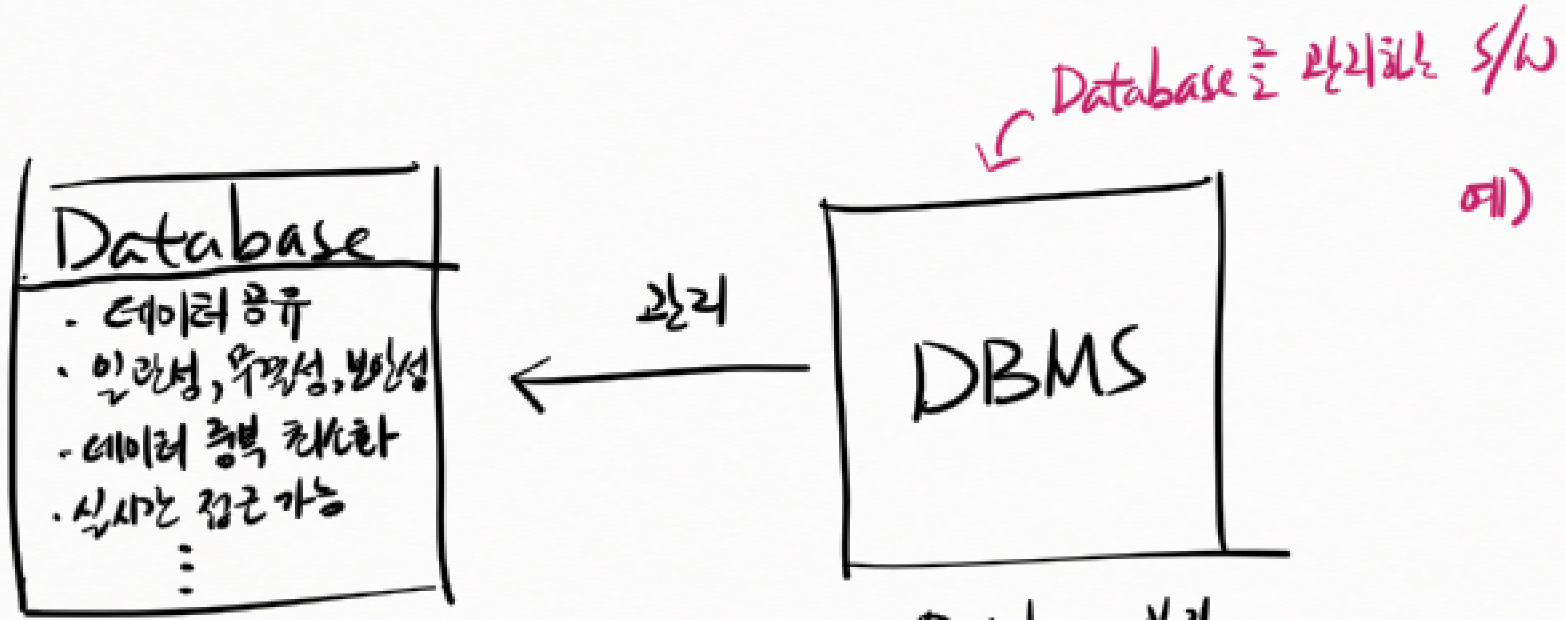


데이터 관리를 분리 → 개발자의 파일 I/O 구현을 덜어준다

- 데이터를 파일에 I/O
- 여러 클라이언트의 동시 요청 처리 — 서버 역할
- 접속자 인증 및 권한 검사
- 데이터 무결성 지원 (integrity)
- 데이터 공유

⋮

* Database 와 DBMS

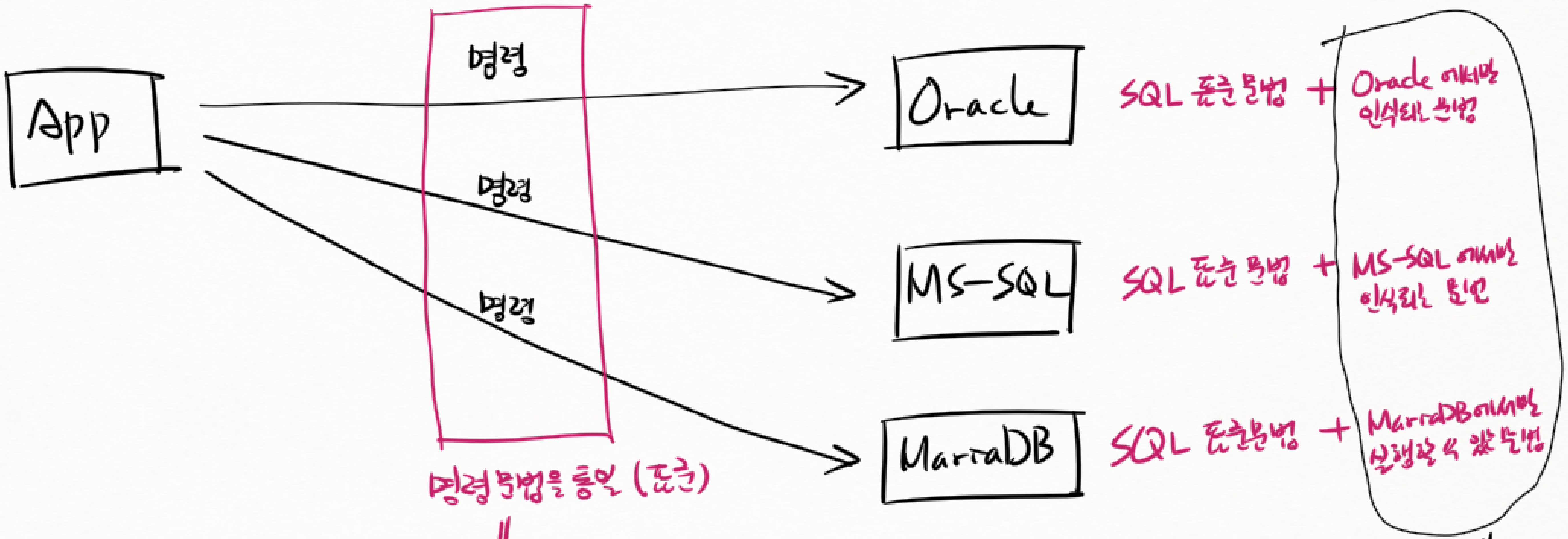


- Database 관리
- 다중 Client 요청 처리
- 접속자 인증/권한 검사

⋮

예) Oracle → Oracle DBMS
MS → MS-SQL
IBM → DB2
Oracle → MySQL.
↓ 오픈소스
개발커뮤니티 → MariaDB *
한국 → Cubrid
Altibase
Tibero
⋮

* DBMS 와 SQL



* SQL 표준?

SQL 표준 문법 + Oracle 에서만 인식되는 문법

SQL 표준 문법 + MS-SQL 에서만 인식되는 문법

SQL 표준문법 + MariaDB에서만 실행할 수 있는 문법

명령 문법을 통일 (표준)

↓

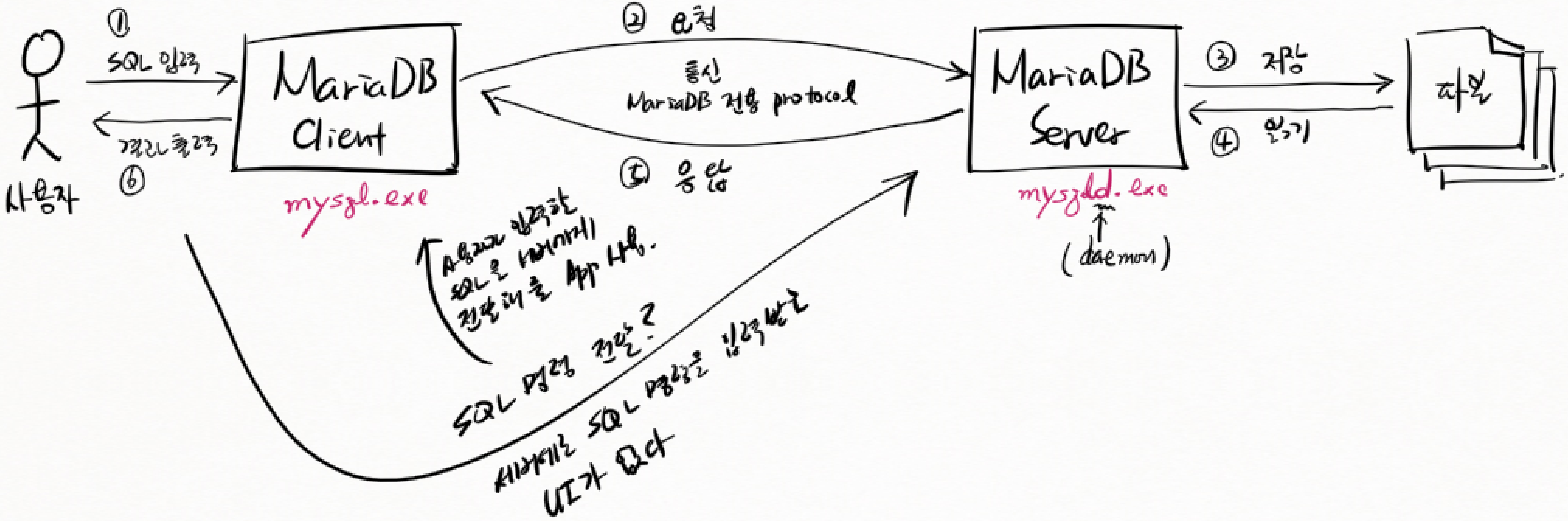
SQL (Structured Query Language)

구조를 갖춘, DBMS에 명령을 내릴때, 사용하는 문법

왜 DBMS 마다
전용 문법이 있는가?

DBMS 만의 특별기능을
사용할 문법이 필요.

* MariaDB Server et Client



* 사용자 등록, Database 생성, 권한 설정

① 사용자 등록

```
create user 'study'@'localhost' identified by '1111';
```

사용자ID

접근 컴퓨터 주소

암호

② 데이터베이스 생성

```
create database studydb character set utf8 collate utf8_general_ci;
```

데이터베이스 명

문자를 인코딩/디코딩 할 때
사용할 문자집합

정렬기준

③ 사용자가 접근할 수 있는 데이터베이스 권한 부여

```
grant all on studydb.* to 'study'@'localhost'
```

권한

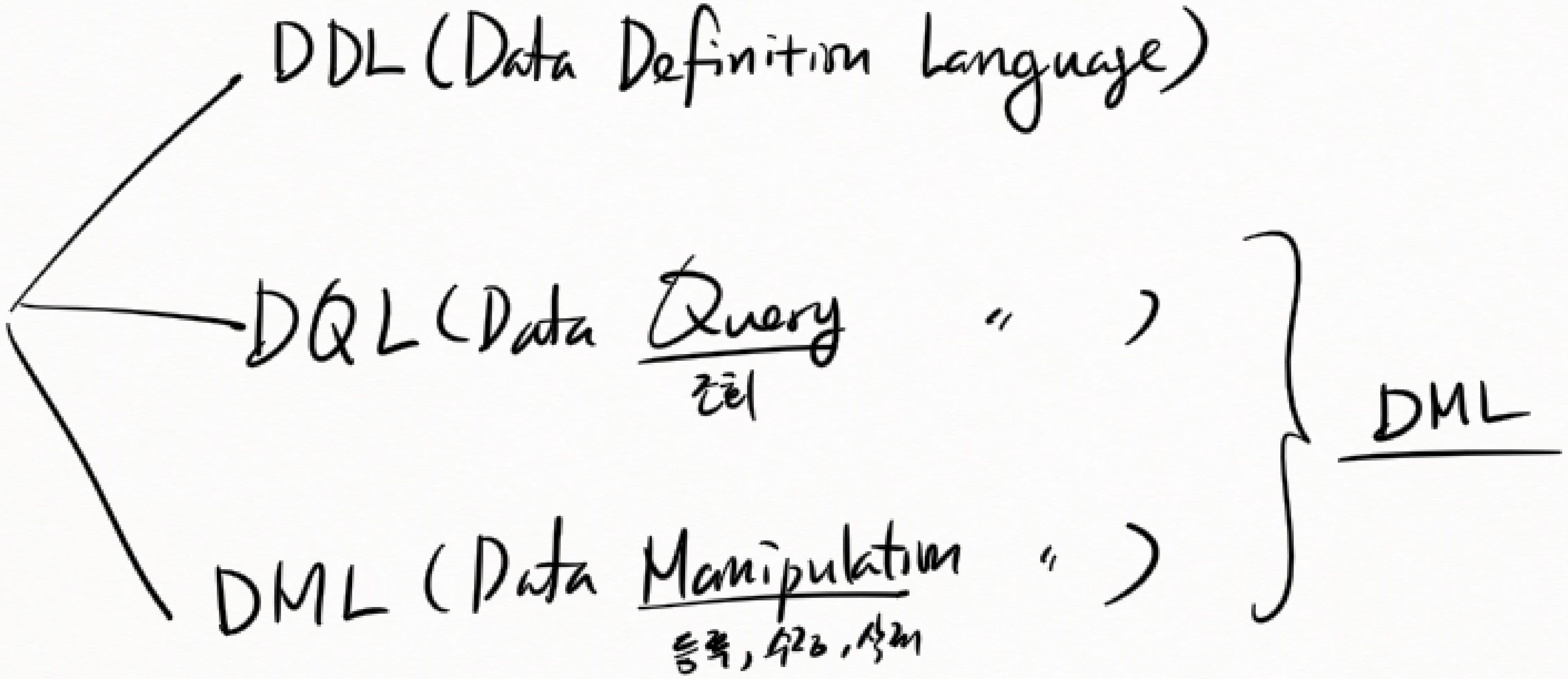
데이터베이스명

항목
예) table, View,
procedure 등

사용자ID

접속 컴퓨터

* SQL



DDL (Data Definition Language)

DQL (Data Query ")

조회

DML (Data Manipulation ")

등록, 수정, 삭제

DML

* SQL : DDL

↳ 테이블, 뷰, 프로시저 등 정의, 변경, 삭제

① 테이블 정의

저장할 데이터 구조 정의

```
create table test01 (
    name varchar(50) not null,
    :
    :
);
```

test01 : 테이블 명

name : 컬럼명

컬럼명
column(열)
||
항목

varchar(50) : 데이터 타입

not null : 옵션

* 사용할 데이터베이스 지정

```
use studydb;
```

studydb : 데이터베이스명

② 테이블에 데이터 입력

```
insert into test1(no, name) values(1, 'aaa');
```

test1 : 테이블명

no, name : 컬럼명
순서는 상관없다

1, 'aaa' : 컬럼에 들어갈 값
컬럼을 지정한 순서로
작성

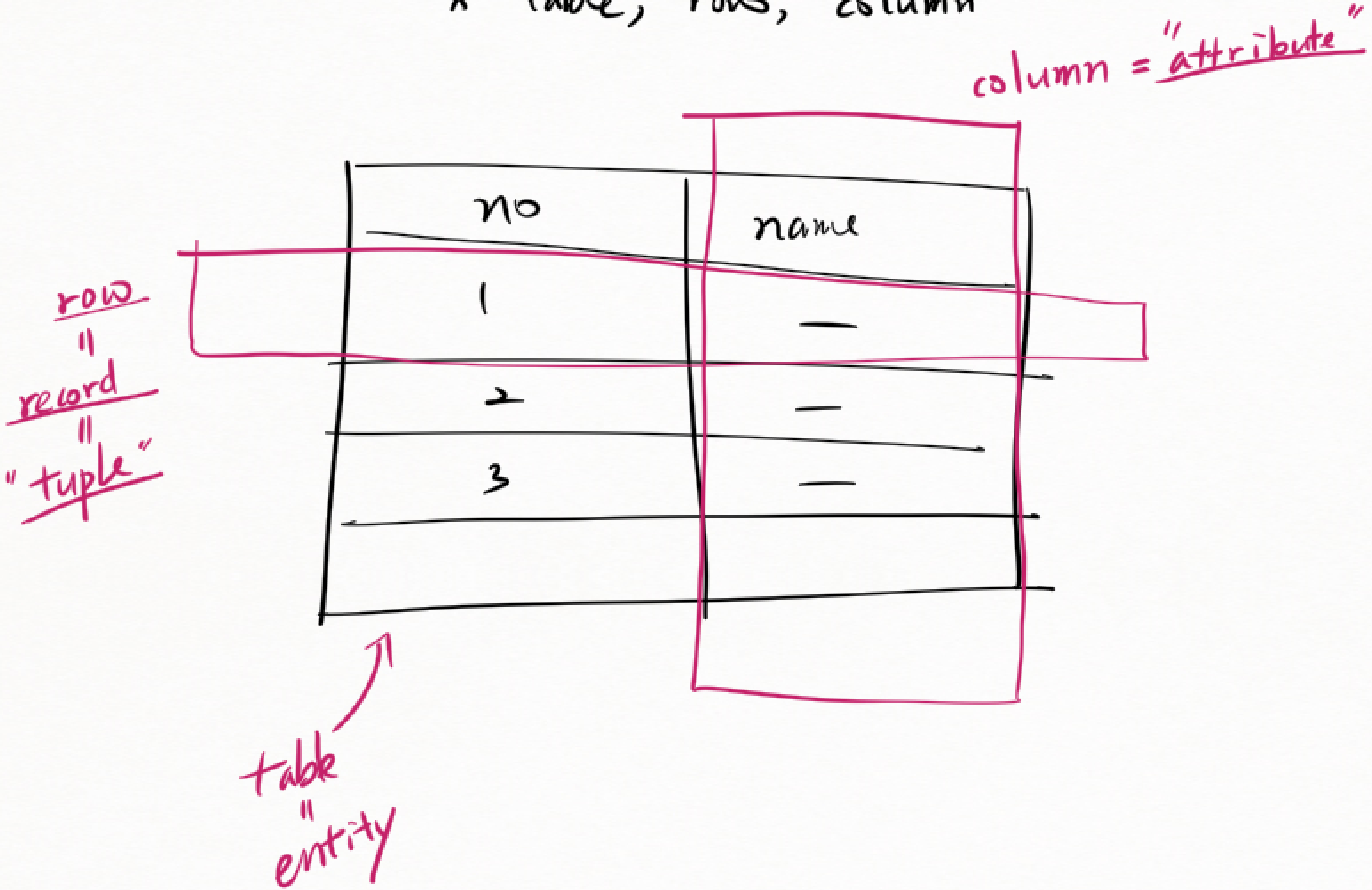
③ 테이블에 들어있는 데이터 조회

```
select no, name from test1;
```

no, name : 컬럼명
||
나열된 순서로
데이터를 추출한다.

test1 : 테이블명

* table, row, column



no	name
1	—
2	—
3	—