2021. 10.05 (67일차) 11:00   * DBMS
                                └→ <u>D</u>atabase   <u>M</u>anagement   <u>S</u>ystem

* DBMS
- 데이터 베이스를 조작하는 S/W
- 예)
  Oracle, MySQL, DBL
  MS-SQL, Altibase,
  Tibero, Cubrid 등

* Database
  - 실시간 접근 가능   :   데이터 처리 요청에 즉시 응답
  - 동시 공유 가능   :   여러 사용자가 공유
  - 데이터의 독립성   :   애플리케이션과 비종속
  - 일관성   :   데이터 처리 작업이 끝난 후 데이터의 같은 <u>유효한 상태를</u> 유지해야 한다.
  - 무결성   :   <u>유효하지 않은 데이터의 등록과 변경, 삭제를 제한</u>   └→예) 탈퇴 회원 데이터 삭제
  - 보안성   :   <u>사용자 인증과 권한검사</u>   └→예) 게시물 작성자를 삭제할 수 없다.   삭제한 회원이 글도 삭제

          ⓐuth   ↓   ↓
                    (<u>authentication</u>)   (<u>authorisation</u>)   삭제하려면 프로젝트로   삭제해야함
                    ID/PWD 일치   기능의 이용권한 검사
                        ↓   ↓
                    유효한 사용자   등록 허용?
                                         변경 허용?
                                           ⋮

DBMS가 애개 반방인 것

SQL 언어로 명령어 작성

DBMS로
MariaDB 설차함 (0/5)

데이터 작업의 요청

명령어 → DBMS
client

DBMS 통신규칙

DBMS → Data I/O → 파일

결과 출력 ←

요청처리 결과

MariaDB
Client
App 은 MariaDB 있지에
같이 설치됨.

- 데이터를 파일에 저장
- 파일에 저장된 데이터를
  조회 / 변경 / 삭제
- 데이터베이스의 기능을 제공

2021 . 10 . 05 (67일차) 13:30   * DBMS 와   SQL   "create user …"   "(서버기준말)
                                            2층   ↳ MariaDB   client에서 만든거 아님
                                                                                           client에는 서버에서 받은응답결과만있음

                              ② 요청

┌──────────┐   ←──────────────────────   ┌──────────┐   ←── SQL입력 ① ○
│ Maria DB │   │ Maria DB │   ㅅ
└──────────┘   ──────────────────────→   │ Client   │   ──────────→   사용자
                 MariaDB protocol   └──────────┘   처리결과
(mysqld . exe)   (mysql exe)
                      응답

                  ┌───────────────────────┐
예) SeverApp ←────┼───────────────────────┼──→ ClientApp
              ←────┼───────────────────────┼────   wwwwww
              └────┼───────────────────────┼──→   ↑
                   └───────────────────────┘   prompt

                    LTS   8 → upgrade ✗
                          ┌─────────────────────────┐
                          │ 11 →   O   │
                          └─────────────────────────┘
                                    ✗
                          ┌─────────────────────────┐
                          │ 17 →   O   │ graalvm 21.3
                          └─────────────────────────┘

* <u>SQL</u>

DBMS 에 공통으로 사용할 수 있는 문법
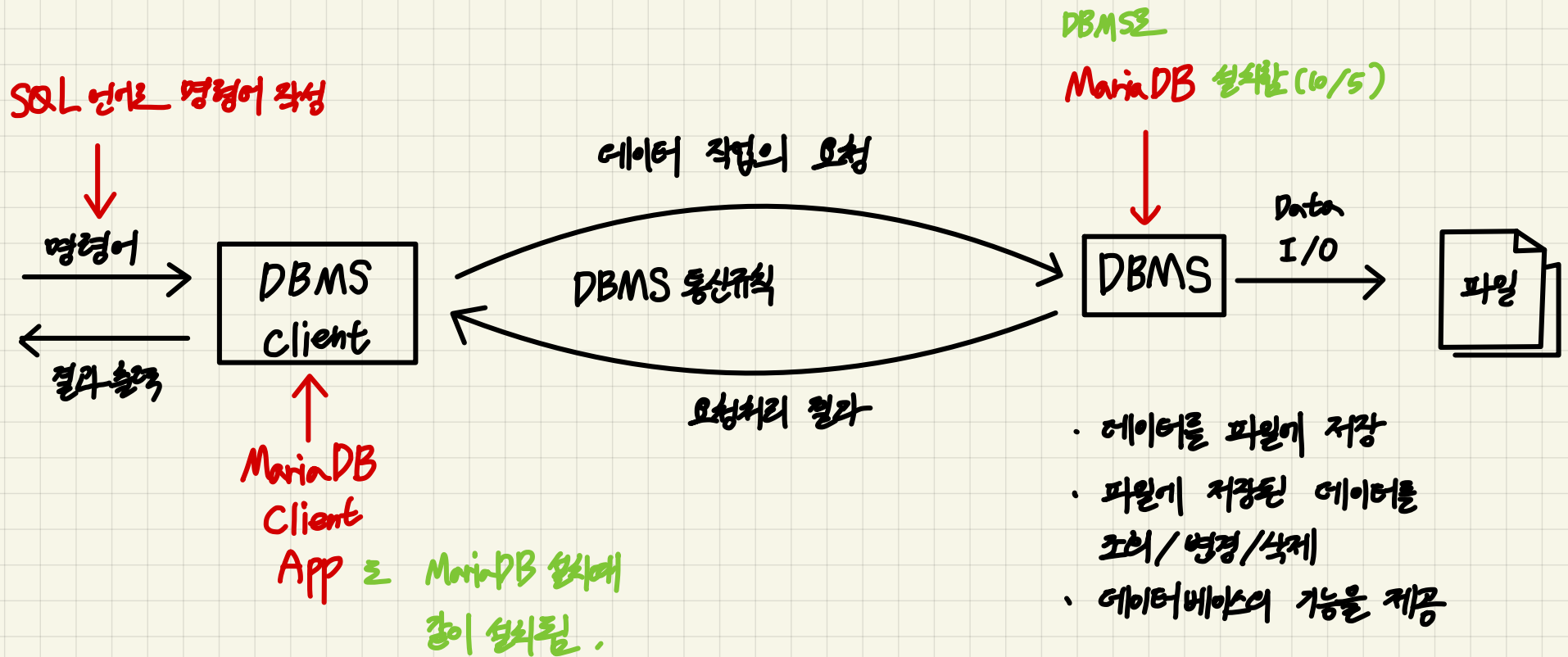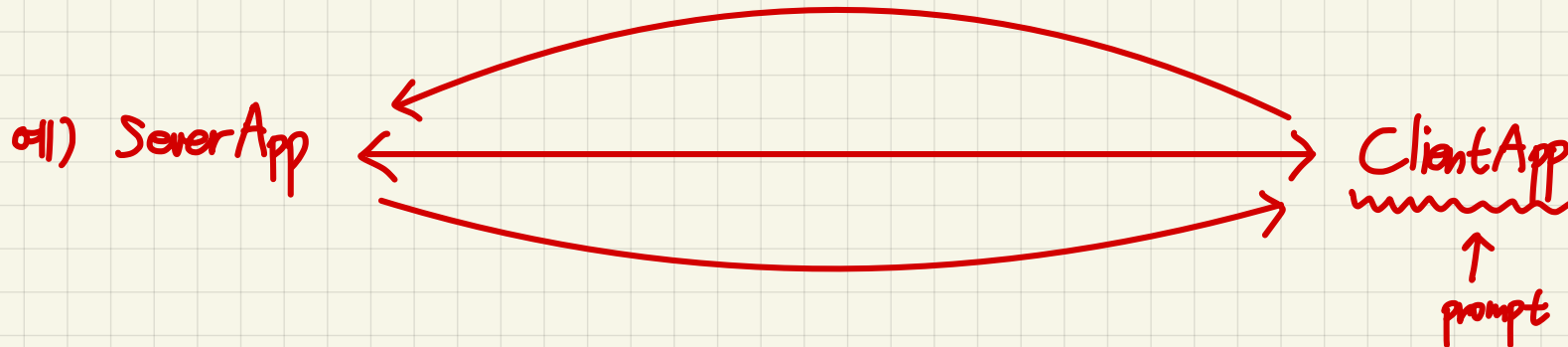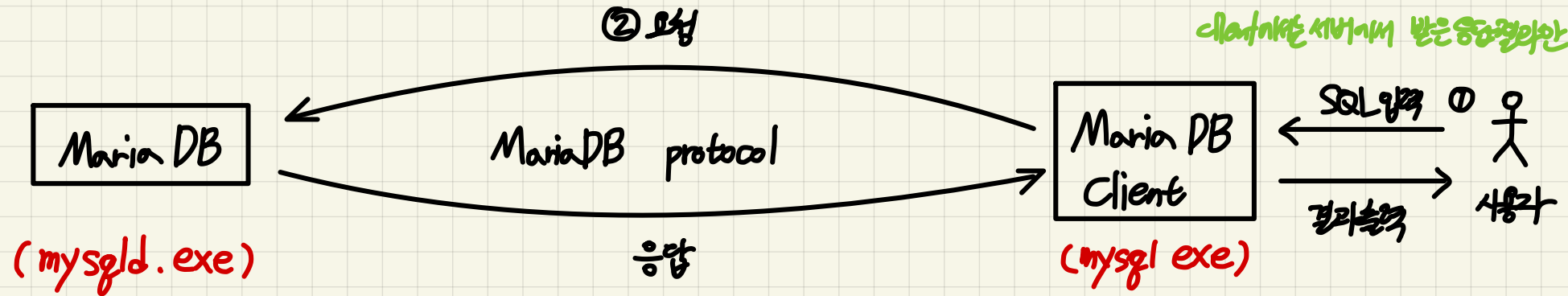
애플리케이션에서
사용하는 SQL은
DBMS 마다
약간씩 다르다

⇓

그래서 실무에서
DBMS에 따라
코드를 변경해야 한다.

App ──── 표준 SQL 문법 ────→ DBMS

App ──── + 오라클 <u>전용문법</u> ────→ Oracle

App ──── + MySQL <u>전용문법</u> ────→ MySQL

App ──── + MS-SQL 전용문법 ────→ MS-SQL
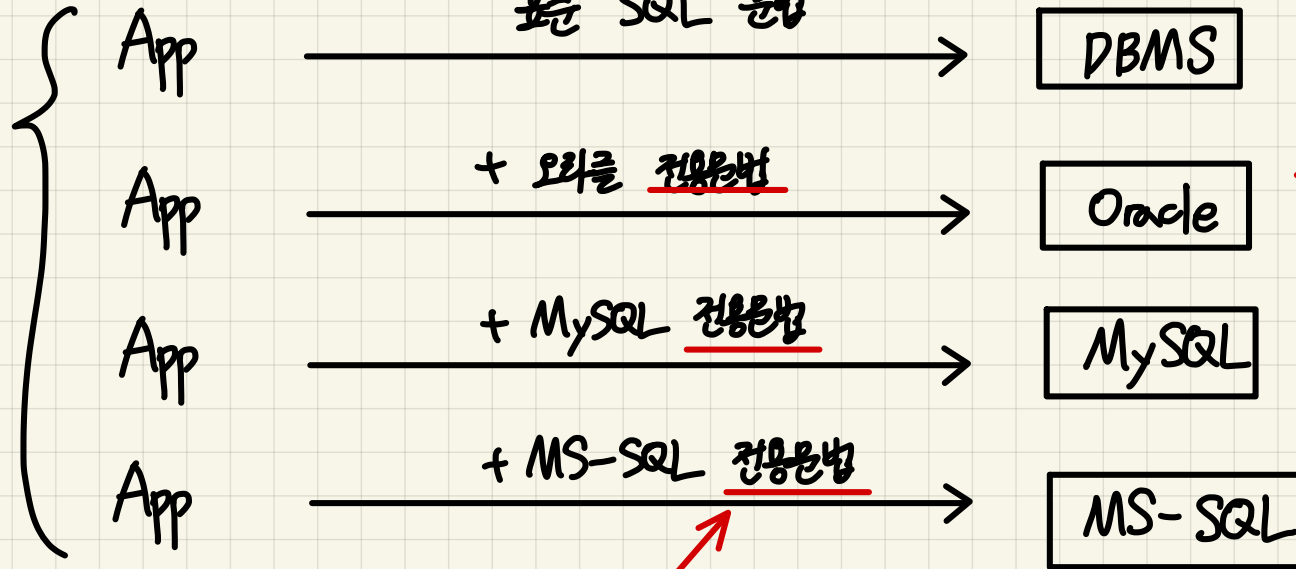
특정 DBMS 에서만
사용할 수 있는 문법

DBMS 제품마다
강점/약점이 있다.

↓

DBMS를
제대로 활용하려면
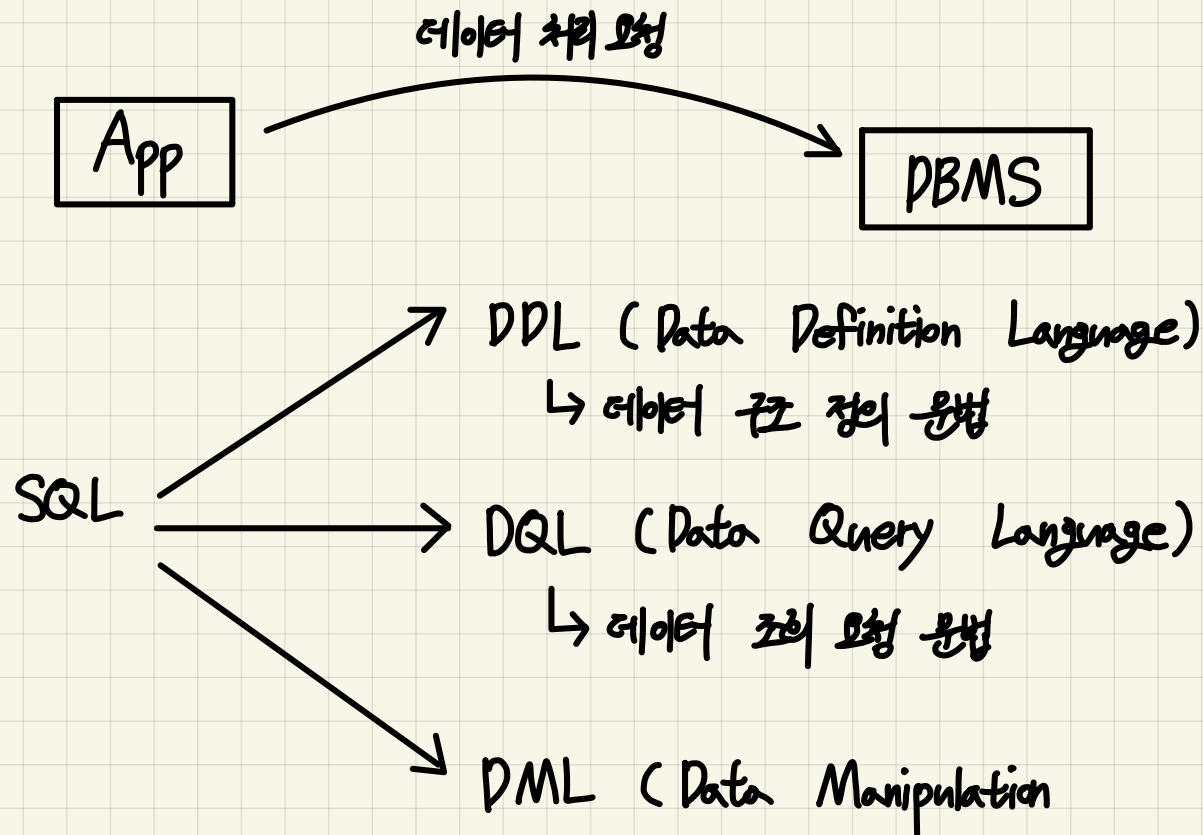DBMS 전용
SQL 문법을
사용할 수밖에 없다.

다음의 대가를 치뤄야 한다.

2021. 10. 05 (67일차)   11:30   * SQL (Structured Query Language)

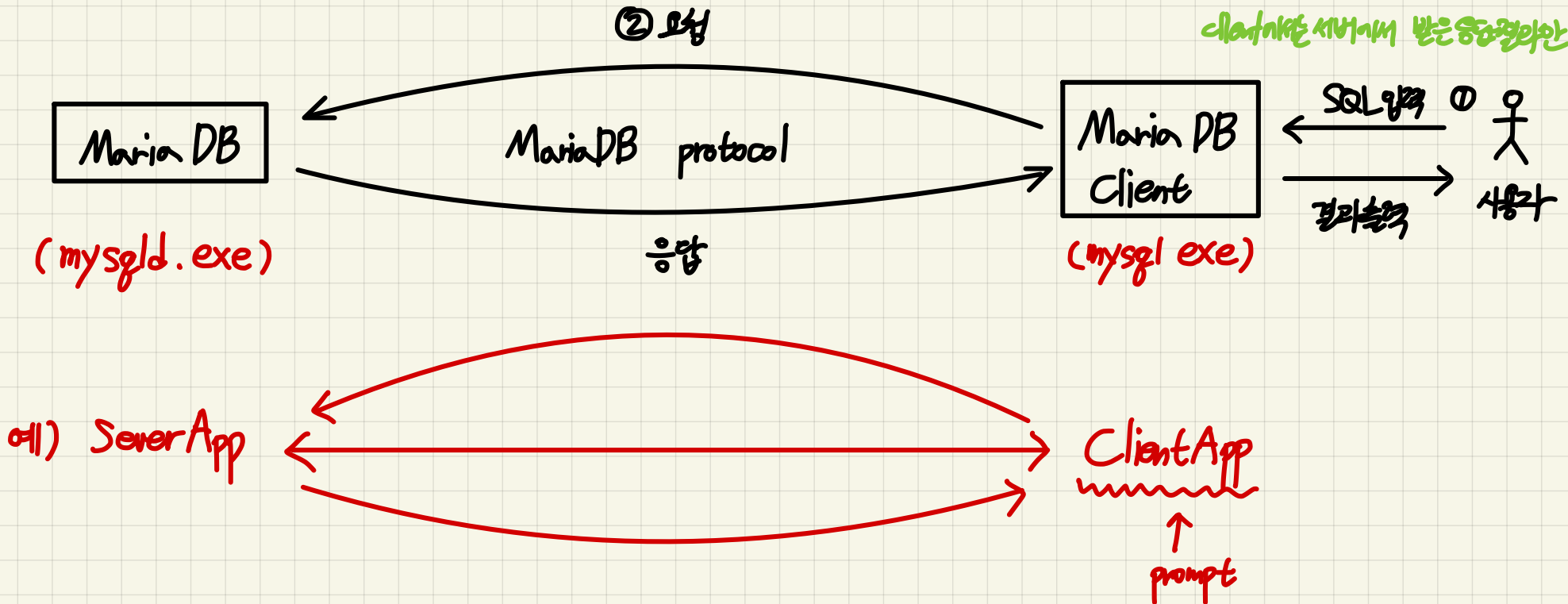DBMS에 상관없이
요청 명령을 작성하는 문법 = SQL

데이터 처리 요청

```
┌───────┐                      ┌───────────┐
│  App  │  ───────────────▶    │   DBMS    │
└───────┘                      └───────────┘
```

```
                    ↗  DDL (Data Definition Language)
                          ↳ 데이터 구조 정의 문법

    SQL  ─────────────▶  DQL (Data Query Language)
                          ↳ 데이터 조회 요청 문법

                    ↘  DML (Data Manipulation
```

MariaDB는 SQL의 오픈 소스임.

2021. 10. 05 (67번째) 13:30   * DBMS 와 SQL

"create user ..."

(서버기준말)
client에서 만든거 아님
client에게 서버에서 받은응답이 아님

② 요청

Maria DB ⟷ MariaDB protocol ⟷ Maria DB Client

(mysqld.exe)   응답   (mysql exe)

SQL입력 ① &
정리출력 사용자

예) ServerApp ⟷ ClientApp
                        ↑
                      prompt

LTS   8 → upgrade X

┌─────────────────────┐
│ 11 →   0 │
└─────────────────────┘
  X
┌─────────────────────┐
│ 17 →   0 │ graalvm 21.3
└─────────────────────┘

2021. 10. 05 (67일차) 13:56   * SQL 테스트 준비

                                        root만 사용자 create 가능   슈거 연속
                    > mysql -u (root) -p ↵   `   "   "
                              아이디   비번   backtic singlequtation double

    ① 사용자 추가
        > create user `study`@`localhost` identified by `1111`; ↵
                          ↑   ↑   ↑
                        사용자ID   접속을 허락할 PC 주소   사용자 암호

              예) localhost 에서 study 아이디로 접속하는 경우에 허락하겠다는 의미

        > create user `study`@`%` identified by `1111`; ↵

    ② 데이터를 저장할 데이터 베이스 생성

        > create database studydb CHARCTER SET utf8 collate utf8_general_ci; ↵
                             ↑   ↑
                           데이터베이스 이름   정렬
                           : `Id`+db

    ③ 데이터 베이스를 사용할 사용자의 권한을 지정

        > grant all on studydb.* to `study`@`localhost`; ↵
          부여하다   알어오든항목

    ④   > use studydb   ····>   MariaDB [studydb]

2021. 10. 05 (6주차) 15:13

* DDL (Data Definition Language)

2021.10.06 (68일차) 9:30
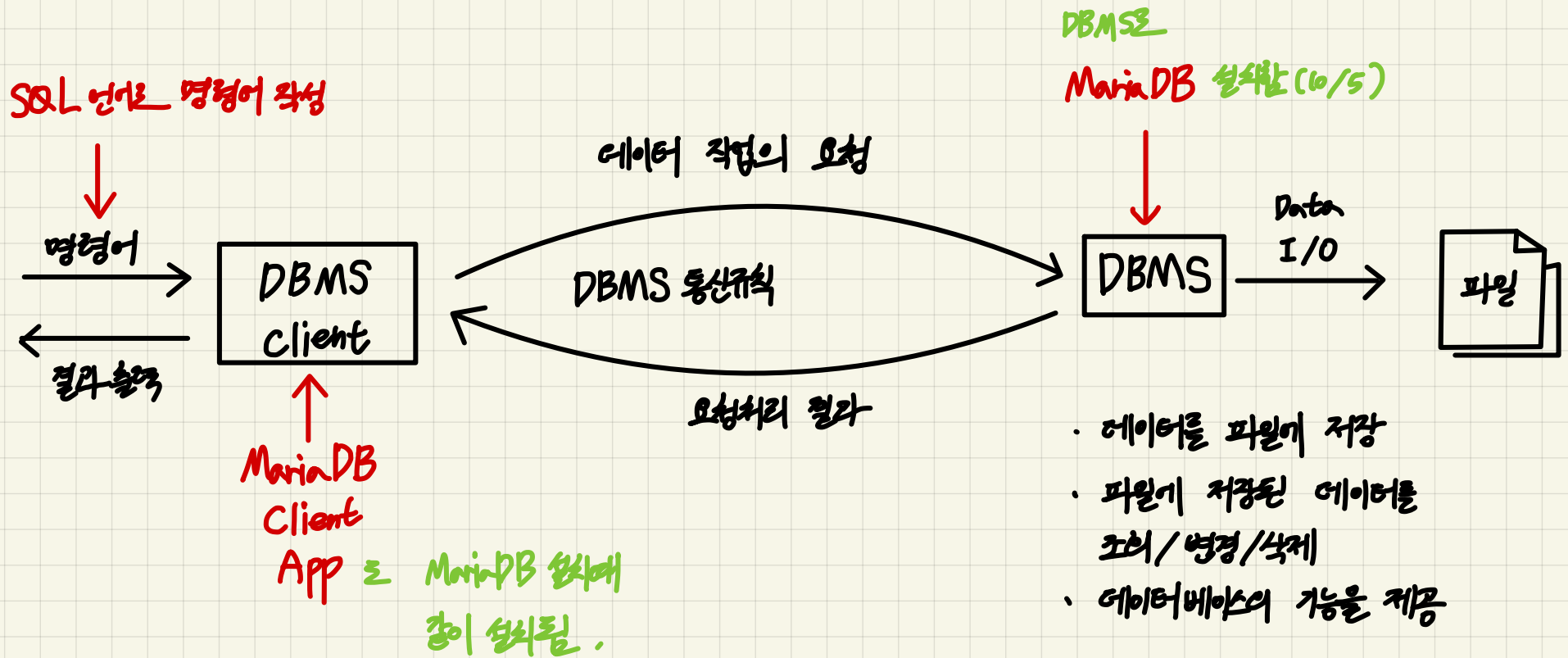
SQL 언어로 명령어 작성

DBMS로
MariaDB 설치함 (0/5)

데이터 작업의 요청

명령어 → DBMS Client ← DBMS 통신규칙 → DBMS → Data I/O → 파일

결과 출력 ← 요청처리 결과

MariaDB Client App 로 MariaDB 설치때 같이 설치됨.

- 데이터를 파일에 저장
- 파일에 저장된 데이터를 조회/변경/삭제
- 데이터베이스의 기능을 제공

* SQL

① DDL → 데이터를 저장하고 다룰 <u>DB객체를</u> 정의한다.

② DQL ⎫
   DML ⎭ → DB객체의
            데이터를 다룬다   ⇒ ⎧ Table
                                 ⎪ View
                                 ⎨ Procedure
                                 ⎪ Function
                                 ⎩   ;

* 테이블

create table   테이블명 (컬럼, ...)   ←── 테이블 정의

drop   table   테이블명   ←── 테이블 삭제

alte   table   테이블명   변경사항   ←── 테이블 변경

describle 테이블명   ←── 테이블 정보조회
   ||
desc

테이블은 꼭 만들어야 데이터 저장 가능
   ⤶ 데이터로 저장한 틀

7절로 데이터 저장? 데이믈 이름? 어떻게 틀가상목은 ? 정의해야구 Data저장값
                                            컬럼

* insert

컬럼명

insert   into   test1 (cno, name)   values (1, 'aaa')

테이블명

컬럼에 들어갈 값

컬럼을 나열한   순서와 일치해야 된다.
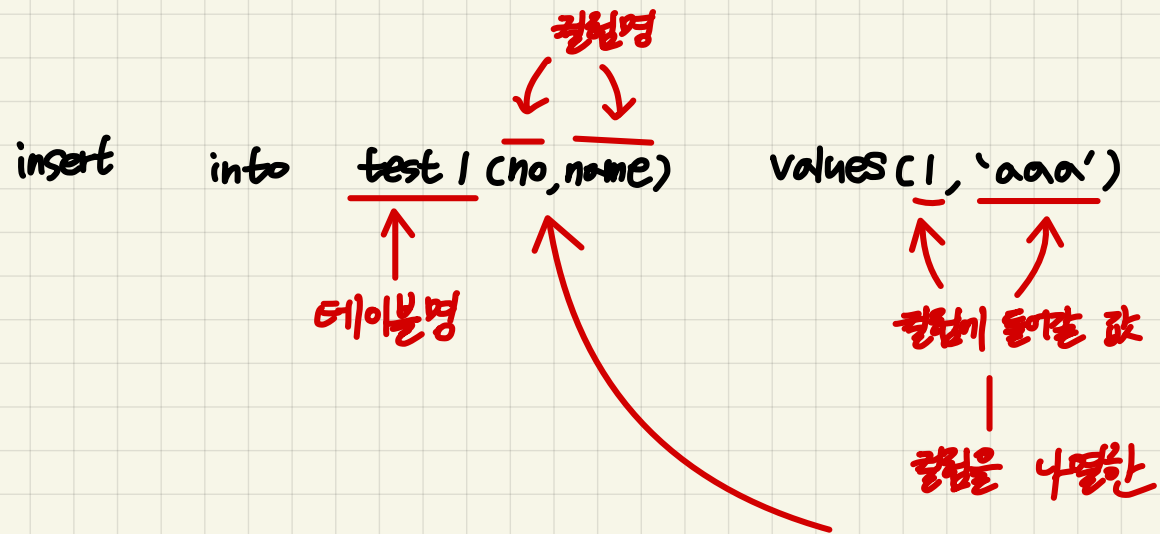
* select

select   no,name   from   test1

*

~~select~~

* key, candidate key, primary key / alternate key, artificial key

=

최소키

key ← 데이터 구분 에 사용할 컬럼들 * (중복X 없는만 가능)

↑ 1개 이상의 컬럼으로
데이터구별가능한 것도 key임
ex) [이름, 전화번호]

[이메일]

[이름, 전화번호]

[이메일, 이름]

[이름, 아이디, 전화번호]

[아이디]

[아이디, 전화번호]

[주민번호, 이름]

[주민번호]

[이메일, 주민번호]

[이름, 우편번호]

* key, candidate key, primary key / alternate key, artificial key

=

최소키

에이터를 구분할 에 사용할 컬럼들 * (중복 X 값들만 가능)

key ←

* candidate key
↳ 최소키

↑ 1개 이상의 컬럼으로
에이터 구별가능한 것도 key임
ex) [이름, 전화번호]

[이메일]

[아이디]

[주민번호]

[이름, 전화번호]

[아이디, 전화번호]

[이메일, 주민번호]

[이메일, 이름]

[주민번호, 이름]

[이름, 아이디, 전화번호]

2021.10.06 (68일차) 11:01 * key, candidate key, primary key / alternate key, artificial key

‖

최소키

주키로 선정되지 않은
나머지 후보키는
'대안키'가 된다.

`alternate key`

데이터로 구분할 때 사용할 컬럼들* (중복X 없는값 가능)

key

* candidate key
↳ 최소키

[이메일]

[이름, 전화번호]

[아이디]

[주민번호]

DB 관리자가
주 키로 결정한 key
= "Primary key"

게시물 :   제목 , 내용 , 작성일 , 작성자 , 조회수 , 좋아요수   게시글 번호

Primary key ?   마땅히 사용할 key가 없을경우

↓

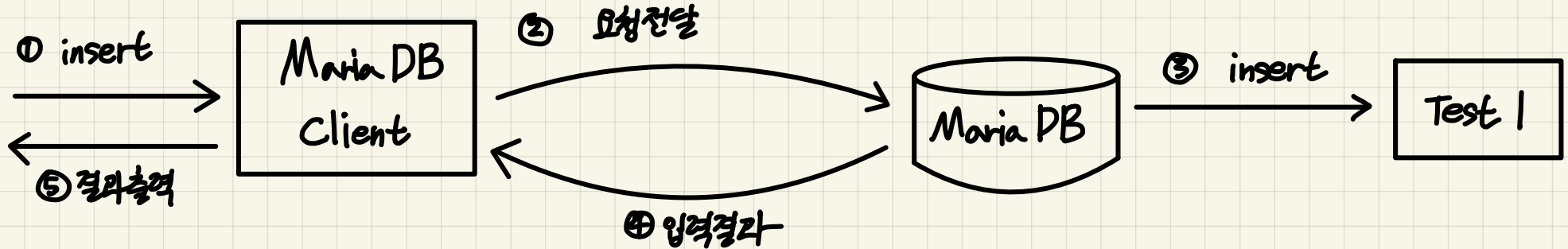Artificial key (인공키) =   임의의 컬럼을 만들어 PK로 각정한다.

예) 일련번호
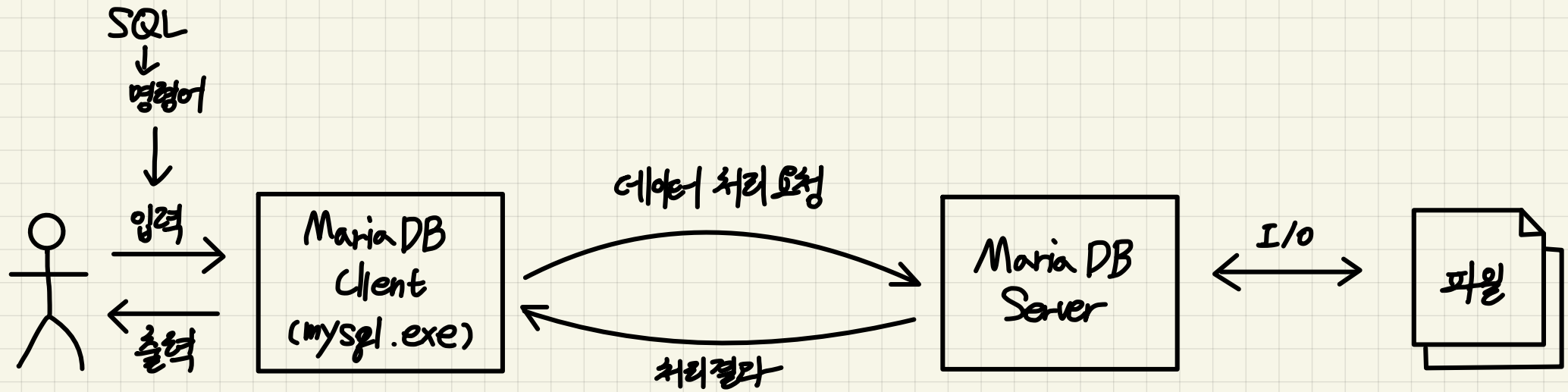
2021.10.06 (68일차)   15:36   * commit

① auto commit = true

① insert →   [Maria DB Client]   ② 요청전달 ⟶   (Maria DB)   ③ insert →   [Test 1]
⑤ 결과출력 ←   ④ 입력결과 ⟵

② auto commit = false

① insert →   [Maria DB Client]   ② 요청전달
insert   ⑤ commit 요청   ⟶   (Maria DB)   ③ insert →   [Test 1]
update   ⑤ rollback 요청
insert
  ↓   ③ 실행 명령 저장
④ commit   // commit 하기 전까지 실제 table에 저장 X   ↓
④ rollback   ~~insert~~
                                                    ~~insert~~   ⑥ 실제 테이블에 적용
                                                    ~~update~~
                        ⑥ 입력값 취소   ~~insert~~
                                                    임시 DB

SQL
↓
명령어
↓
입력 →
출력 ←

Maria DB
Client
(mysql.exe)

데이터 처리요청

처리결과

Maria DB
Server

I/O

파일
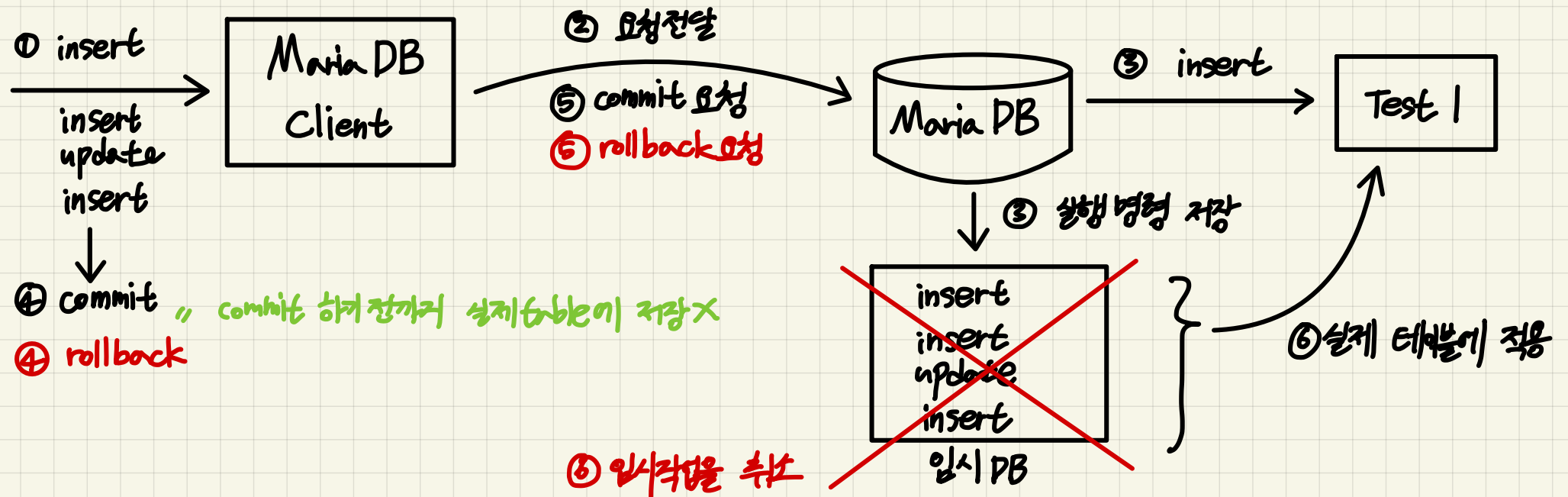
PDL (① 테이블 정의 / 변경 / 삭제)
         뷰

DML (PML ② 입력 / 변경 / 삭제
       DQL ③ 조회 / 검색

* selection / projection

select no, name from test1 where working = `Y´

selection

no	name	class	working	tel
1	aaa			
2	bbb			
3	ccc			
4	ddd			
5	eee			
6	fff			
7	ggg			

* Foreign Key

① 자바 객체

Board ──writer──→ Member

Board ──files──→ * Attached File →

- no
- title
- content
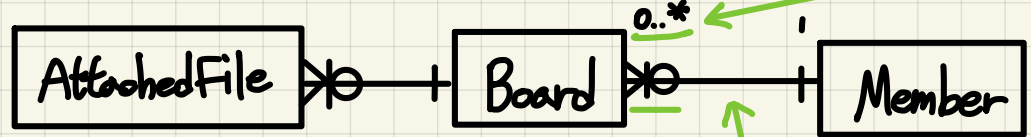- registered Date
- viewCount
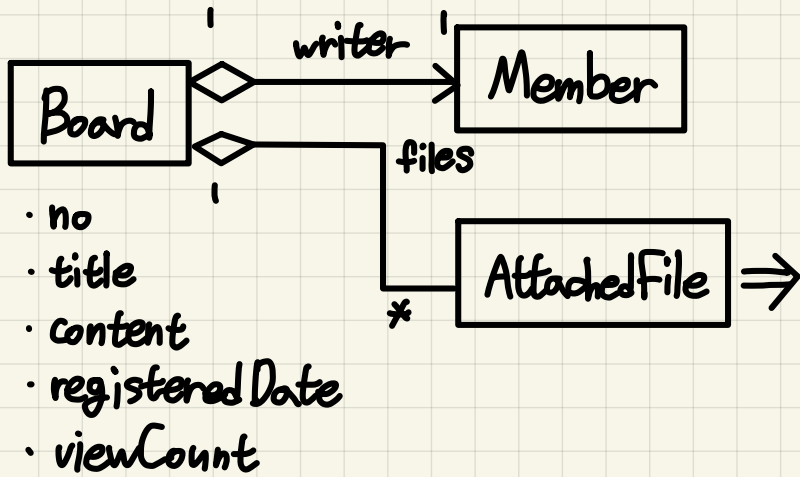
```
class Board {
    int no;
    String title;
    String content;
    Date registeredDate;
    int viewCount;
    Member writer;
    List<AttachedFile> files;
        ;
}
```

② 테이블 간 관계 (ER- Diagram ; ERD)

Entity → Relation

0 또는 2 이상

0..*

AttachedFile ─○<─┼─ Board ─>○─┼─ Member

Board table 이 Member table을 참조한다는 의미

1번 게시글 ────── 100번 홍길동
2번 게시글 ────── 101번 임꺽정
3번 게시글 ───── 102번 유관순

ERD 표기법

① Information Engineering Notation

* 객체간의 관계와 테이블간의 관계는 다르다!

프로그래밍 하기쉬운 객체간의 관계를 설정

데이터 중복을 최소화시키는 쪽으로 관계를 설정.

문학
↓
경험으로 얻은 사실을
학문으로 체계화시켜서
연구 하는 것
⇓
해당 분야에 종사하는 사람들이
시행착오를 줄이게 도와준다.

2021.10.07 (69일차) 11:54   * Foreign Key (외부키) - 다른 테이블의 PK 값

② 테이블 간 관계

| AttachedFile |---○<------------------|- Board -○<--------------------| Member |
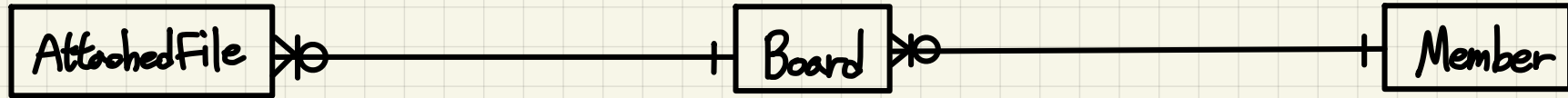
AttachedFile   PK를 가리킴   Board   PK를 가리킴   Member
                                  (참조)   (참조)

(pk)번호   파일명   게시글(FK)   (pk)번호   제목   작성자(FK)   (pk)번호   이름
11   ooa.gif   101 ——————— 101   oooo   1   1   홍길동
12   bbb.gif   102 ⎤   102   bbbb   1   2   임꺽정
13   ccc.gif   102 ⎦   103   cccc   3   3   유관순
14   ddd.gif   103 ⎤   104   dddd   4   4   안중근
15   eee.gif   103 ⎦

자식 테이블 •————————————————————→ 부모 테이블
                                              참조

자식 테이블 •————————————————————→ 부모 테이블

2021.10.07 (69일차) 13:25 　　　　* 컬럼 중복

Board

no	title	content	rdt	f1	f2	f3	f4	f5
1	aaa	——	—	a.gif	b.gif			
2	bbb	——	—	x.gif	y.gif	z.gif	k.gif	
3	ccc	——	—	○	○	○	○	○
4	ddd	——	—	m1.gif	m2.gif	m3.gif	m4.gif	m5.gif ○
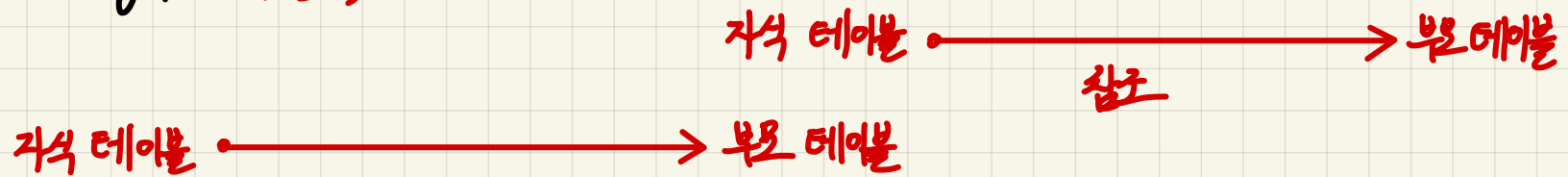
이렇게 같은 종류의 데이터를 여러개 저장하기 위해
컬럼이 중복으로 선언된 경우

다음에 문제 발생

첨부파일이 없음에도
5개의 컬럼이 존재하기 때문에
메모리 낭비

컬럼이 부족해서
첨부파일을 더 저장할 수 없다.

이런 컬럼을 검색하서
데이터 중복을 야기

정규화 실행!

* 컬럼 중복

게시글 데이터 보관

Board

no	title	content	rdt	f1	f2	f3	f4	f5
1	aaa	_____	__	a.gif	b.gif			
2	bbb	_____	__	x.gif	y.gif	z.gif	k.gif	
3	ccc	_____	__	⬭	⬭	⬭	⬭	⬭
4	ddd	_____	__	m1.gif	m2.gif	m3.gif	m4.gif	m5.gif	⬭

⇒

Board

no	title	content	rdt
1	aaa	_____	__
2	bbb	_____	__
3	ccc	_____	__
4	ddd	_____	__

Attached File

정확한 값 저장

no	filepath	bno (FK)
1	a.gif	1
2	b.gif	1
3	x.gif	2
4	y.gif	2
5	z.gif	2
6	k.gif	2
7	m1.gif	4
8	m2.gif	4

* ERD : foward / reverse   engineering

자동툴 : ER-Win, exerd, ...

DBMS

ER-Diagram

Table
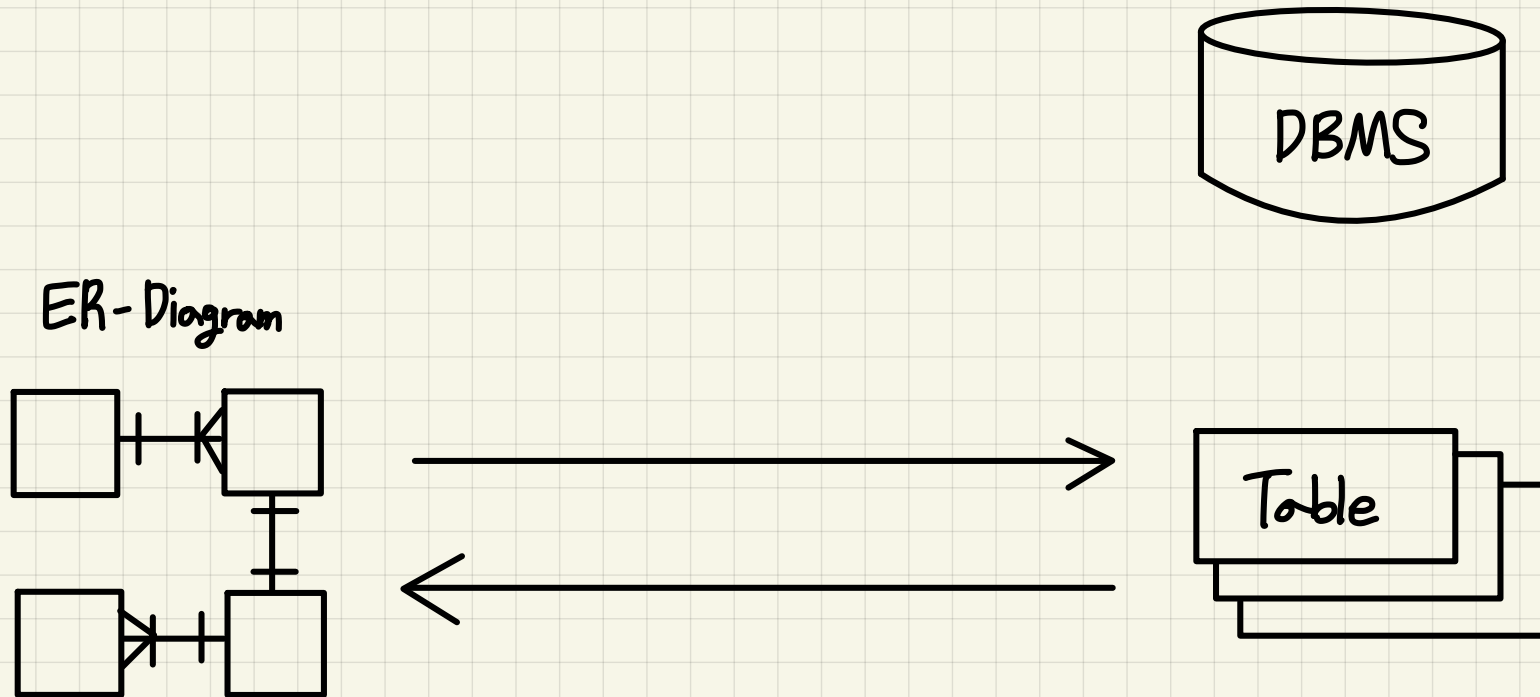
2021.10.07(69일차) 14:50 * SQL 조인 테스트를 위한 테이블의 ERD)
예) 교육센터 관리 시스템

회원기본정보 | memb | — | addr
매니저 5 | mgr
학생 8 | stnt
강사 7 | tcher

1 홍
2 김

강의
100 java
101 C#

lect

강의실 | room
강의실 사진 | room-photo

lect_tcher ← 1개의 강의에 여러 강사가 담당함

1 : 多

1 - 100
1 - 101
2 - 101

수강신청
lect_appl
← 多:多 →

memb ← Oracle —— room-phot
member ← MySQL —— room-photo
                   roomPhoto