

데이터베이스

인공지능소프트웨어학과

담당교수: 김희숙
(jasmin11@hanmail.net)

데이터베이스

4주차

담당교수: 김희숙
(jasmin11@hanmail.net)

Quiz

MySQL

담당교수: 김희숙
(jasmin11@hanmail.net)

[Quiz] 테이블 생성

데이터베이스: testdb

테이블: 학과

[Quiz 1-1] SQL 문법을 사용하여 테이블 생성하고 데이터 입력하시오 (studept-k.sql)

--학과(학과번호, 학과명)

테이블이름: 학과

| 테이블 이름 | 열 이름 | 데이터 형식 | 제약조건 | 기본키 | 외 레키 | FK테이블 | FK열 |
|--------|------|-------------|----------|-----|------|-------|-----|
| 학과 | 학과번호 | int | NOT NULL | PK | | | |
| | 학과명 | varchar(50) | | | | | |

drop table 학과;

--학과(학과번호, 학과명)

```
CREATE TABLE 학과 (  
    학과번호 int ,  
    학과명 varchar(50),  
     KEY(학과번호)  
);
```

select * from 학과;

데이터 입력: 3건의 데이터를 입력하시오

[Quiz] 테이블 생성

데이터베이스: testdb

테이블: member_tbl

member_tbl(c_no, c_name, phone, address, grade)

1. 데이터베이스 생성: testdb
2. 테이블 생성: member_tbl
3. 데이터 입력: 5건의 데이터를 입력하시오

| 컬럼ID | | 형태 | 길이 | NULL | 비고 |
|---------|------|---------|----|----------|-------------|
| C_NO | 회원번호 | CHAR | 5 | NOT NULL | PRIMARY KEY |
| C_NAME | 회원명 | VARCHAR | 15 | | |
| PHONE | 전화번호 | VARCHAR | 11 | | |
| ADDRESS | 주소 | VARCHAR | 50 | | |
| GRADE | 등급 | VARCHAR | 6 | | |

| 회원번호 | 회원명 | 전화번호 | 주소 | 등급 |
|-------|-----|-------------|---------|-----|
| 10001 | 홍길동 | 01011112222 | 서울시 강남구 | 일반 |
| 10002 | 장발장 | 01022223333 | 성남시 분당구 | 일반 |
| 10003 | 임꺽정 | 01033334444 | 대전시 유성구 | 일반 |
| 20001 | 성춘향 | 01044445555 | 부산시 서구 | VIP |
| 20002 | 이몽룡 | 01055556666 | 대구시 북구 | VIP |

[실습] 테이블 생성

데이터베이스: testdb

테이블: 제품

제품

| 제품번호 | 제품명 | 재고량 | 단가 | 제조업체 |
|------|-------|------|------|------|
| p01 | 그냥만두 | 5000 | 4500 | 대한식품 |
| p02 | 매운짬면 | 2500 | 5500 | 민국푸드 |
| p03 | 쿵떡파이 | 3600 | 2600 | 한빛제과 |
| p04 | 맛난초콜렛 | 1250 | 2500 | 한빛제과 |
| p05 | 얼큰라면 | 2200 | 1200 | 대한식품 |

```
USE testdb;
```

```
DROP TABLE if exists 제품;
```

```
CREATE TABLE 제품 (  
    제품번호    char(3)          NOT NULL ,  
    제품명      varchar(20)      ,  
    재고량      int ,  
    단가        int ,  
    제조업체    varchar(20)      ,  
    PRIMARY KEY(제품번호)  
);
```

[실습] 테이블 생성

데이터베이스: testdb

테이블: 제품

제품

| 제품번호 | 제품명 | 재고량 | 단가 | 제조업체 |
|------|-------|------|------|------|
| p01 | 그냥만두 | 5000 | 4500 | 대한식품 |
| p02 | 매운짬면 | 2500 | 5500 | 민국푸드 |
| p03 | 콩떡파이 | 3600 | 2600 | 한빛제과 |
| p04 | 맛난초콜렛 | 1250 | 2500 | 한빛제과 |
| p05 | 얼큰라면 | 2200 | 1200 | 대한식품 |

-- 제품(제품번호, 제품명, 재고량, 단가, 제조업체)

INSERT INTO 제품 VALUES ('p01', '그냥만두', 5000, 4500, '대한식품');

INSERT INTO 제품 VALUES ('p02', '매운짬면', 2500, 5500, '민국푸드');

INSERT INTO 제품 VALUES ('p03', '콩떡파이', 3600, 2600, '한빛제과');

INSERT INTO 제품 VALUES ('p04', '맛난초콜렛', 1250, 2500, '한빛제과');

INSERT INTO 제품 VALUES ('p05', '얼큰라면', 2200, 1200, '대한식품');

select * from 제품;

관계 데이터 모델

담당교수: 김희숙
(jasmin11@hanmail.net)

키(Key)

□기본키(Primary Key)

- ✓ 각 튜플을 유일하게 구별할 수 있는 하나 이상의 속성의 집합
- ✓ 중복불가
- ✓ 필수입력

테이블
관계

학생

필드 (속성, 열, 애트리뷰트) column

| 번호 | 이름 | 학년 | 분반 | 학과번호 |
|----|-----|----|----|------|
| 1 | 한지혜 | 1 | YB | 1 |
| 2 | 이정우 | 1 | YA | 1 |
| 3 | 오지영 | 2 | J1 | 2 |
| 4 | 강재미 | 1 | YB | 1 |
| 5 | 박철호 | 2 | J1 | 2 |

레코드
(튜플, 행)
row

키(Key)



□외래키(Foreign Key)

- ✓한 릴레이션의 기본키를 참조하는 키
- ✓참조되는 릴레이션의 기본키 값에 반드시 존재해야 한다

(외래키는 널 값 입력허용)

- ✓참조무결성 제약조건

고객 테이블

| 고객 ID | 이름 | 주민등록번호 | 전화번호 |
|-------|-----|----------------|--------------|
| 1 | 홍길동 | 111111-1111111 | 011-234-5678 |
| 2 | 원도우 | 222222-2222222 | 012-546-8526 |
| 3 | 김한빛 | 333333-3333333 | 014-654-8752 |

주문 테이블

| 물품 ID | 물품 이름 | 고객 ID |
|-------|-------|-------|
| 1 | A | 1 |
| 2 | B | 1 |
| 3 | C | 2 |

실습

기본키, 외래키

담당교수: 김희숙
(jasmin11@hanmail.net)

[실습] (MySQL)

- 1단계: 데이터베이스 생성(스키마 생성)
- 사용할 데이터베이스 선택
- 2단계: 테이블 생성
- 3단계: 데이터 입력

[테이블 명세서] ↗

테이블이름: 학과

| 테이블 이름 | 열 이름 | 데이터 형식 | 제약조건 | 기본키 | 외래키 | FK테이블 | FK열 |
|--------|------|-------------|----------|-----|-----|-------|-----|
| 학과 | 학과번호 | int | NOT NULL | PK | | | |
| | 학과명 | varchar(50) | | | | | |

학과(학과번호, 학과명)

학과 테이블 <<출력 형태>>

학과

| 학과번호 | 학과명 |
|------|-------------|
| 1 | 컴퓨터소프트웨어공학과 |
| 2 | 컴퓨터정보공학과 |
| 3 | 인공지능소프트웨어과 |

[실습] (MySQL)



-- MySQL Workbench 실행

-- (MySQL)

drop database studydb;

```
CREATE DATABASE `studydb`  
  DEFAULT CHARACTER SET utf8mb4;
```

use studydb;

1단계: 스키마/데이터베이스 생성 studydb

2단계: 테이블 생성

학과, 학생

3단계: 데이터 입력

-- 1단계: 스키마 생성

```
CREATE SCHEMA `studydb`  
  DEFAULT CHARACTER SET utf8mb4 ;
```

-- 사용할 데이터베이스 선택

```
USE studydb;
```

-- 데이터베이스 삭제

-- (MySQL)

```
drop database studydb;
```

[실습] 테이블 생성

테이블 명세서

[실습] 테이블 생성하고
데이터 입력하시오

| 테이블명 | 열 이름 | 데이터 형식 | NULL 유무 | 기본키 | 외래키 | FK 테이블명 | FK 열 이름 | 비고 |
|------|------|-------------|----------|-----|-----|---------|---------|----|
| 학과 | 학과번호 | int | NOT NULL | PK | | | | |
| | 학과명 | varchar(50) | | | | | | |

학과

| 학과번호 | 학과명 |
|------|-------------|
| 1 | 컴퓨터소프트웨어공학과 |
| 2 | 컴퓨터정보공학과 |
| 3 | 인공지능소프트웨어과 |

[실습] 테이블 생성

[테이블 명세서] ↗

테이블이름: 학과

| 테이블 이름 | 열 이름 | 데이터 형식 | 제약조건 | 기본키 | 외래키 | FK테이블 | FK열 |
|--------|------|-------------|----------|-----|-----|-------|-----|
| 학과 | 학과번호 | int | NOT NULL | PK | | | |
| | 학과명 | varchar(50) | | | | | |

-- 학과 테이블 생성

```
CREATE TABLE 학과 (
    학과번호 int NOT NULL ,
    학과명 varchar(50) ,
    PRIMARY KEY(학과번호)
);
```

-- 학과(학과번호,학과명)

```
INSERT INTO 학과 VALUES(1,'컴퓨터소프트웨어공학과');
INSERT INTO 학과 VALUES(2,'컴퓨터정보공학과');
INSERT INTO 학과 VALUES(3,'인공지능소프트웨어과');
```

학과

| 학과번호 | 학과명 |
|------|-------------|
| 1 | 컴퓨터소프트웨어공학과 |
| 2 | 컴퓨터정보공학과 |
| 3 | 인공지능소프트웨어과 |

```
select * from 학과;
```

[예제] SQL: 테이블 생성(기본키, 외래키)

```
CREATE TABLE 학과 (  
    학과번호 int NOT NULL ,  
    학과명 varchar(50) ,  
    PRIMARY KEY(학과번호)  
);
```

학과

| 학과번호 | 학과명 |
|------|-------------|
| 1 | 컴퓨터소프트웨어공학과 |
| 2 | 컴퓨터정보공학과 |
| 3 | 인공지능소프트웨어과 |

--학과(학과번호, 학과명)

insert into 학과 values(1,'컴퓨터소프트웨어공학과');

insert into 학과 values(2,'컴퓨터정보공학과');

insert into 학과 values(3,'정보통신과');

[실습] 테이블 생성

[실습] 기본키, 외래키

테이블 2개

학과

| 학과번호 | 학과명 |
|------|-------------|
| 1 | 컴퓨터소프트웨어공학과 |
| 2 | 컴퓨터정보공학과 |
| 3 | 인공지능소프트웨어과 |

(주의)

1. 테이블 생성 순서
2. 테이블 삭제 순서
3. 데이터 입력 순서

학생

| 번호 | 이름 | 학년 | 분반 | 학과번호 |
|----|-----|----|----|------|
| 1 | 한지혜 | 1 | YB | 1 |
| 2 | 이정우 | 1 | YA | 1 |
| 3 | 오지영 | 2 | J1 | 2 |
| 4 | 강재미 | 1 | YB | 1 |
| 5 | 박철호 | 2 | J1 | 2 |

[예제] 테이블 생성(기본키, 외래키)

테이블 명세서

| 테이블명 | 열 이름 | 데이터 형식 | NULL 유무 | 기본키 | 외래키 | FK 테이블명 | FK 열 이름 | 비고 |
|------|------|-------------|----------|-----|-----|---------|---------|----|
| 학과 | 학과번호 | int | NOT NULL | PK | | | | |
| | 학과명 | varchar(50) | | | | | | |

| 테이블명 | 열 이름 | 데이터 형식 | NULL 유무 | 기본키 | 외래키 | FK 테이블명 | FK 열 이름 | 비고 |
|------|------|-------------|----------|-----|-----|---------|---------|----|
| 학생 | 번호 | int | NOT NULL | PK | | | | |
| | 이름 | varchar(12) | | | | | | |
| | 학년 | int | | | | | | |
| | 분반 | char(2) | | | | | | |
| | 학과번호 | int | | | FK | 학과 | 학과번호 | |

학과

| 학과번호 | 학과명 |
|------|-------------|
| 1 | 컴퓨터소프트웨어공학과 |
| 2 | 컴퓨터정보공학과 |
| 3 | 인공지능소프트웨어과 |

학생

| 번호 | 이름 | 학년 | 분반 | 학과번호 |
|----|-----|----|----|------|
| 1 | 한지혜 | 1 | YB | 1 |
| 2 | 이정우 | 1 | YA | 1 |
| 3 | 오지영 | 2 | J1 | 2 |
| 4 | 강재미 | 1 | YB | 1 |
| 5 | 박철호 | 2 | J1 | 2 |

[예제] 테이블 생성(기본키, 외래키)

```
CREATE TABLE 학과 (  
    학과번호 int NOT NULL ,  
    학과명 varchar(50) ,  
    PRIMARY KEY(학과번호)  
);
```

학과

| 학과번호 | 학과명 |
|------|-------------|
| 1 | 컴퓨터소프트웨어공학과 |
| 2 | 컴퓨터정보공학과 |
| 3 | 인공지능소프트웨어과 |

학생

| 번호 | 이름 | 학년 | 분반 | 학과번호 |
|----|-----|----|----|------|
| 1 | 한지혜 | 1 | YB | 1 |
| 2 | 이정우 | 1 | YA | 1 |
| 3 | 오지영 | 2 | J1 | 2 |
| 4 | 강재미 | 1 | YB | 1 |
| 5 | 박철호 | 2 | J1 | 2 |

```
CREATE TABLE 학생 (  
    번호 int NOT NULL ,  
    이름 varchar(12) ,  
    학년 int ,  
    분반 char(2) ,  
    학과번호 int ,  
    PRIMARY KEY(번호) ,  
    FOREIGN KEY (학과번호)  
        REFERENCES 학과(학과번호)  
);
```

--학과(학과번호, 학과명)

insert into 학과 values(1,'컴퓨터소프트웨어공학과');

insert into 학과 values(2,'컴퓨터정보공학과');

insert into 학과 values(3,'인공지능소프트웨어과');

--학생(번호, 이름, 학년, 분반, 학과번호)

insert into 학생 values(1,'한지혜',1,'YB',1);

insert into 학생 values(2,'이정우',1,'YA',1);

insert into 학생 values(3,'오지영',2,'J1',2);

insert into 학생 values(4,'강재미',1,'YB',1);

insert into 학생 values(5,'박철호',2,'Y1',2);

commit;

-- (Oracle)

select * from 학과;

select * from 학생;

| 테이블명 | 열 이름 | 데이터 형식 | NULL 유무 | 기본키 | 외래키 | FK 테이블명 | FK 열 이름 | 비고 |
|------|------|-------------|----------|-----|-----|---------|---------|----|
| 학과 | 학과번호 | int | NOT NULL | PK | | | | |
| | 학과명 | varchar(50) | | | | | | |

| 테이블명 | 열 이름 | 데이터 형식 | NULL 유무 | 기본키 | 외래키 | FK 테이블명 | FK 열 이름 | 비고 |
|------|------|-------------|----------|-----|-----|---------|---------|----|
| 학생 | 번호 | int | NOT NULL | PK | | | | |
| | 이름 | varchar(12) | | | | | | |
| | 학년 | int | | | | | | |
| | 분반 | char(2) | | | | | | |
| | 학과번호 | int | | | FK | 학과 | 학과번호 | |

학과

| 학과번호 | 학과명 |
|------|-------------|
| 1 | 컴퓨터소프트웨어공학과 |
| 2 | 컴퓨터정보공학과 |
| 3 | 인공지능소프트웨어과 |

학생

| 번호 | 이름 | 학년 | 분반 | 학과번호 |
|----|-----|----|----|------|
| 1 | 한지혜 | 1 | YB | 1 |
| 2 | 이정우 | 1 | YA | 1 |
| 3 | 오지영 | 2 | J1 | 2 |
| 4 | 강재미 | 1 | YB | 1 |
| 5 | 박철호 | 2 | J1 | 2 |

데이터베이스 시스템

담당교수: 김희숙
(jasmin11@hanmail.net)



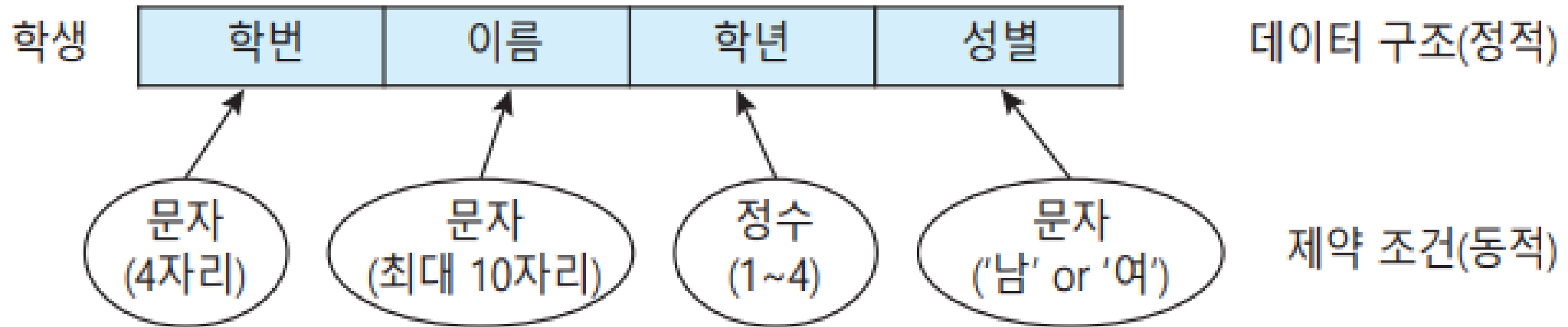
스키마

인스턴스: 데이터베이스에 실제 저장된 값

* 스키마(Schema):

데이터베이스를 구성하는 구조와 제약조건에 대한 명세를 기술한 것

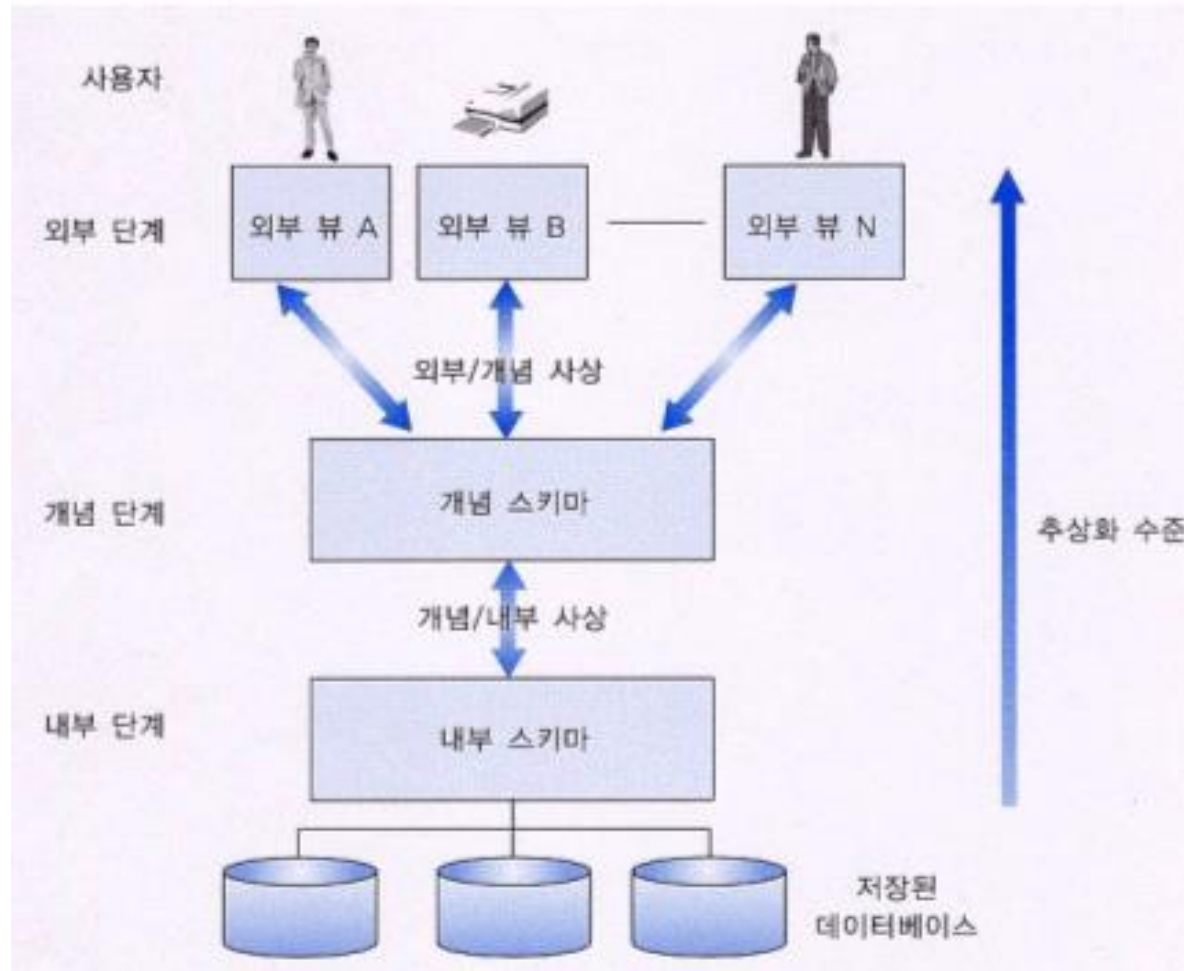
예) 학생 스키마



학생(학번, 이름, 학년, 성별)

[요약] 3단계 구조

ANSI/SPARC 아키텍처 (스키마 3단계 구조)



(그림 출처: "데이터베이스배움터", 홍의경 저, 생능, 2012)





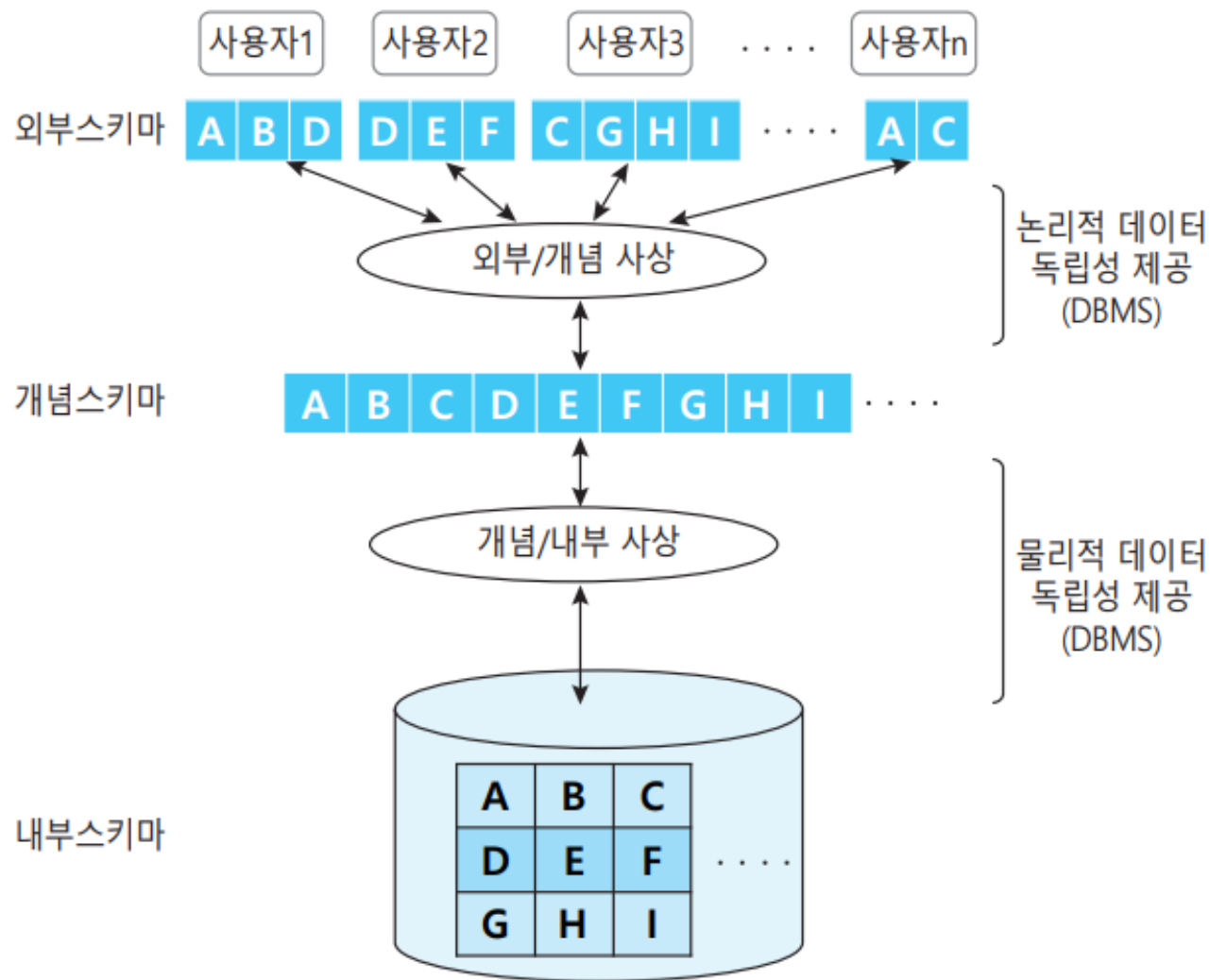
데이터 독립성

* 3단계 스키마 구조:

- 1) 외부 스키마
- 2) 개념 스키마
- 3) 내부 스키마

* 데이터 독립성:

논리적 데이터 독립성
물리적 데이터 독립성

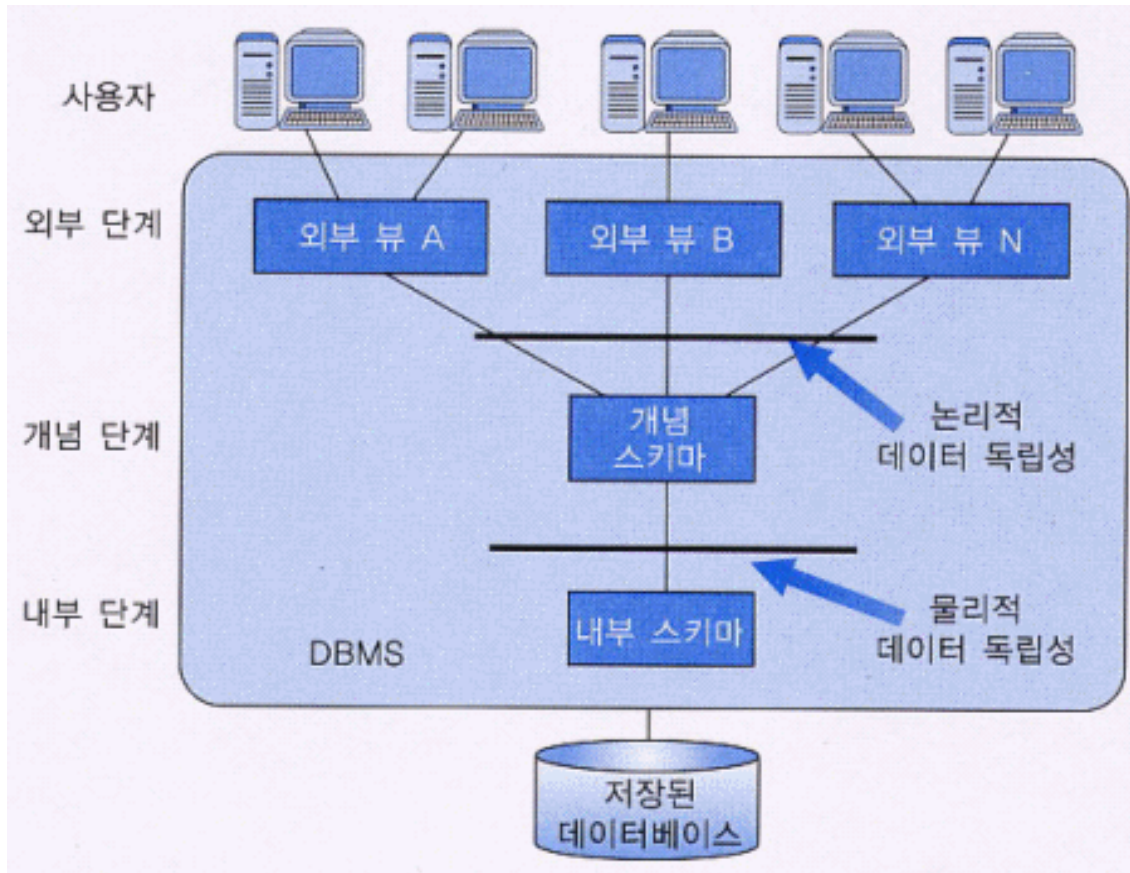


(그림 출처: "SQL과 NoSQL 기반의데이터베이스 입문", 박성진, 생능, 2023)

[요약] 데이터 독립성

* 3단계 스키마 구조:

- 1) 외부 스키마
- 2) 개념 스키마
- 3) 내부 스키마



- **논리적 데이터 독립성:**

개념스키마가 변화해도
외부스키마는 영향 받지 않는다

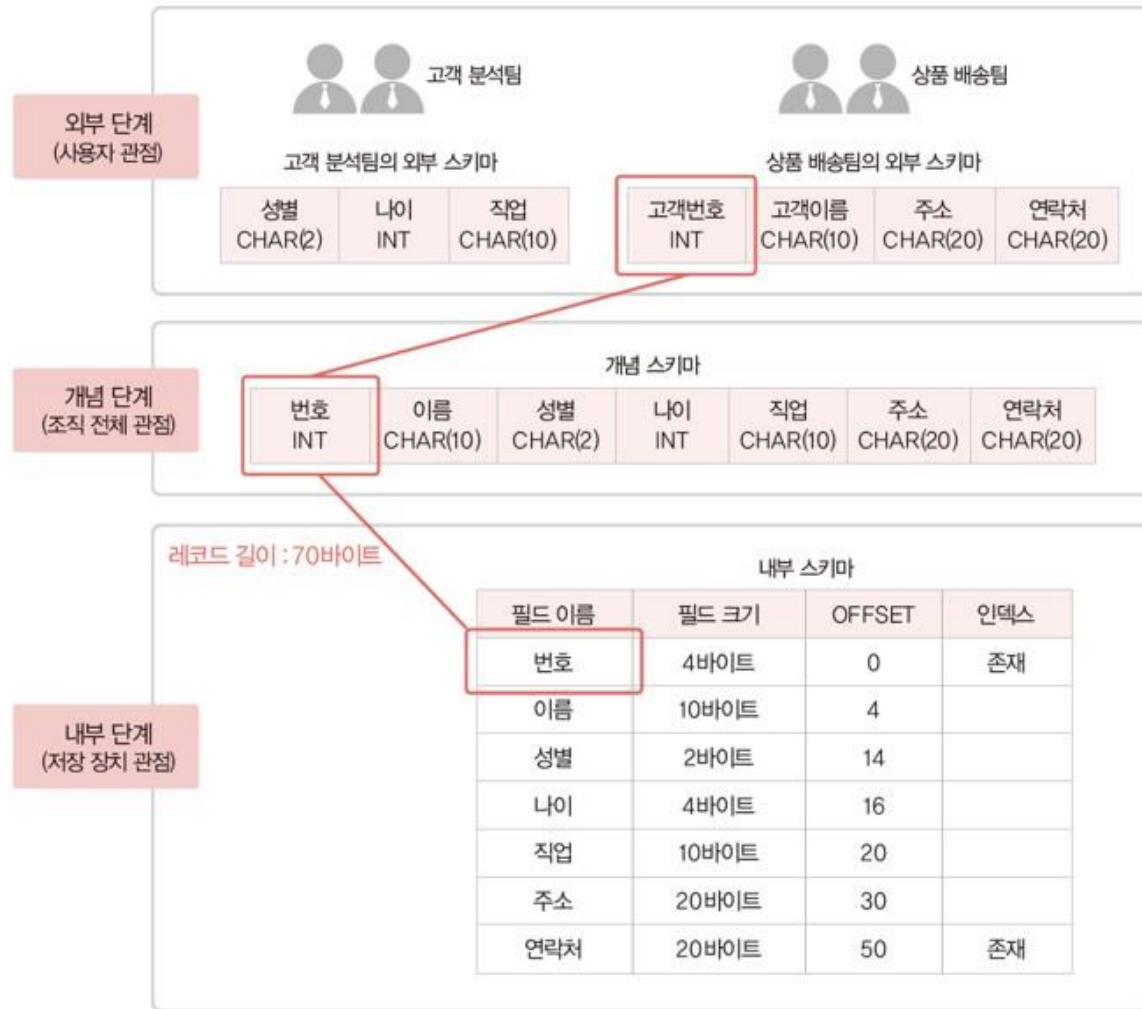
- **물리적 데이터 독립성:**

내부스키마가 변화해도
개념스키마, 외부스키마는 영향 받지 않는다

(그림 출처: "데이터베이스배움터", 홍의경 저, 생능, 2012)

[실습] 3단계 구조(외부스키마, 개념스키마, 내부스키마)

[실습] 외부 스키마(schema_view-ex.sql)



외부 뷰(view)

35 • `select * from 고객분석팀;`

| 성별 | 나이 | 직업 |
|----|----|----|
| 남 | 20 | 학생 |
| 남 | 22 | 학생 |
| 여 | 20 | 학생 |

외부 뷰(view)

47 • `select * from 상품배송팀;`

| 고객 번호 | 고객 이름 | 주소 | 연락처 |
|-------|-------|----|---------------|
| 1 | 홍길동 | 서울 | NULL |
| 2 | 임꺽정 | 인천 | 010-1111-1111 |
| 3 | 신아로미 | 서울 | 010-2222-2222 |

23 • `select * from 가입고객;`

| 번호 | 이름 | 성별 | 나이 | 직업 | 주소 | 연락처 |
|------|------|------|------|------|------|---------------|
| 1 | 홍길동 | 남 | 20 | 학생 | 서울 | NULL |
| 2 | 임꺽정 | 남 | 22 | 학생 | 인천 | 010-1111-1111 |
| 3 | 신아로미 | 여 | 20 | 학생 | 서울 | 010-2222-2222 |
| NULL | NULL | NULL | NULL | NULL | NULL | NULL |

(그림 출처: "데이터베이스 개론 2판", 김연희 저, 한빛아카데미, 2019)

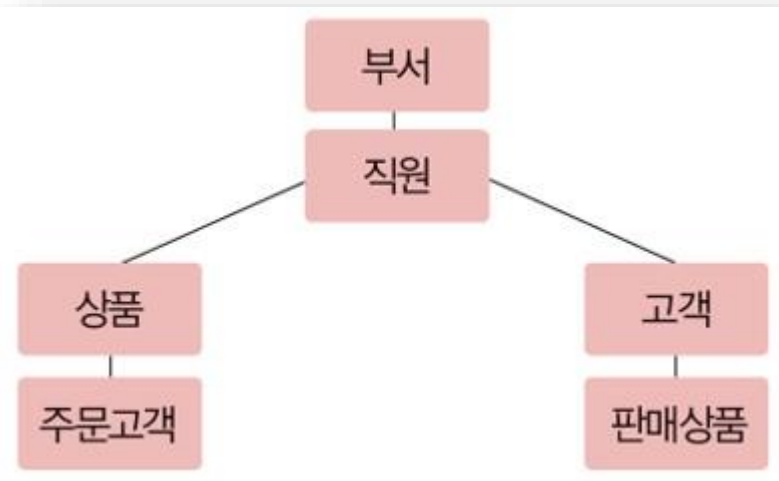
데이터베이스 시스템 발전 과정

• 데이터베이스 관리 시스템

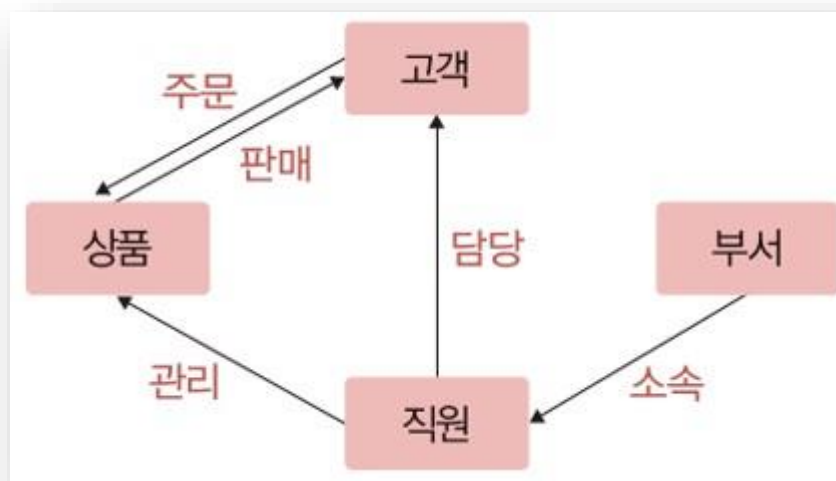
- 계층형 데이터 모델: HDBMS
- 네트워크형 데이터 모델: NDBMS
- 관계형 데이터 모델: RDBMS
- 객체지향형 데이터 모델: OODBMS
- 객체관계형 데이터 모델: ORDBMS

| 아이디 | 비밀번호 | 이름 | 연락처 | 주소 | 적립금 |
|--------|------|-----|-------------|---------|------|
| apple | 1234 | 정소화 | 02-111-1111 | 서울시 마포구 | 1000 |
| banana | 9876 | 김선우 | 02-222-2222 | 경기도 부천시 | 500 |

관계형 데이터 모델



계층형 데이터 모델

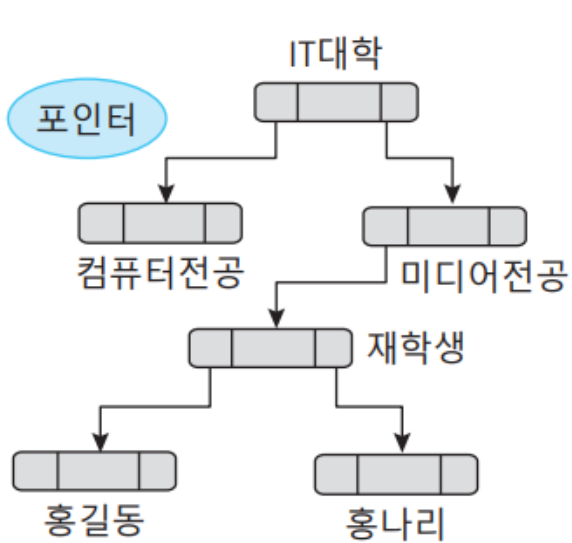


네트워크형 데이터 모델

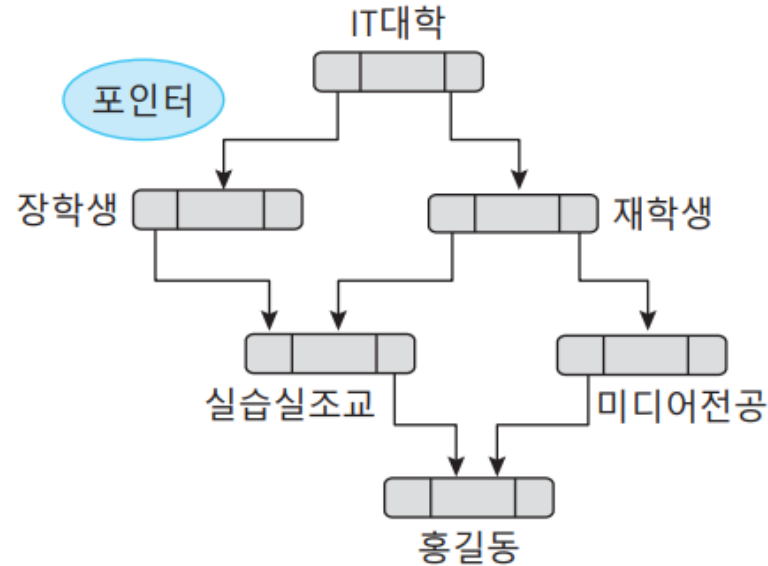
데이터 모델

* 데이터 모델(Data Model):

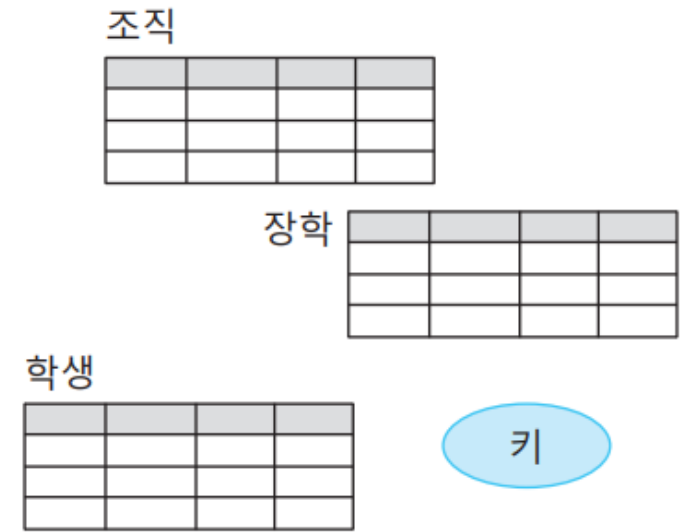
데이터베이스 종류에 따라서 다양한 저장 구조를 갖는다
데이터베이스 구조를 명세하기 위한 개념



계층형 데이터 모델



네트워크형 데이터 모델

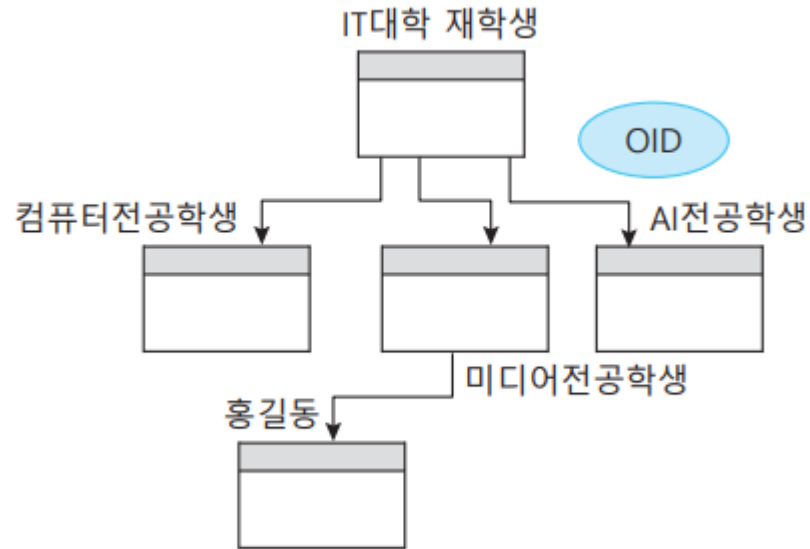


관계형 데이터 모델

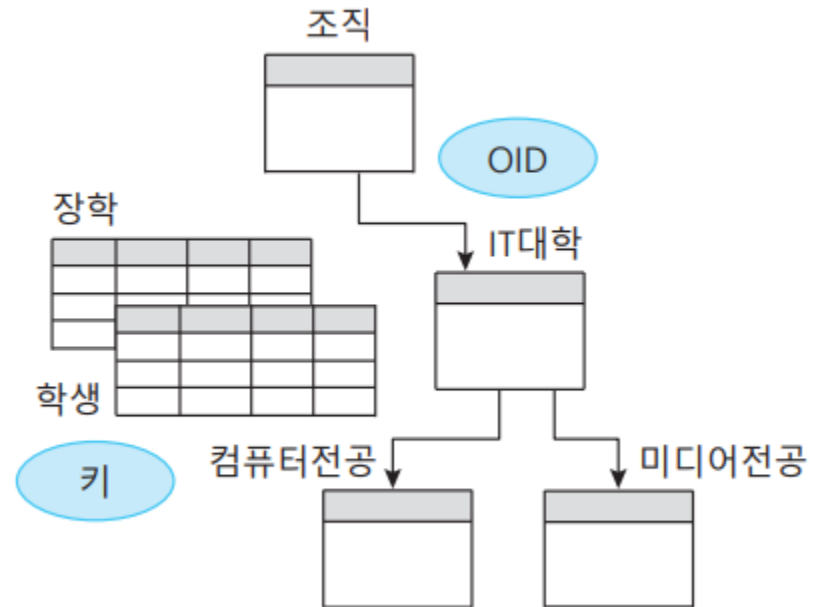
데이터 모델

* 데이터 모델(Data Model):

데이터베이스 종류에 따라서 다양한 저장 구조를 갖는다
데이터베이스 구조를 명세하기 위한 개념



객체 지향형 데이터 모델



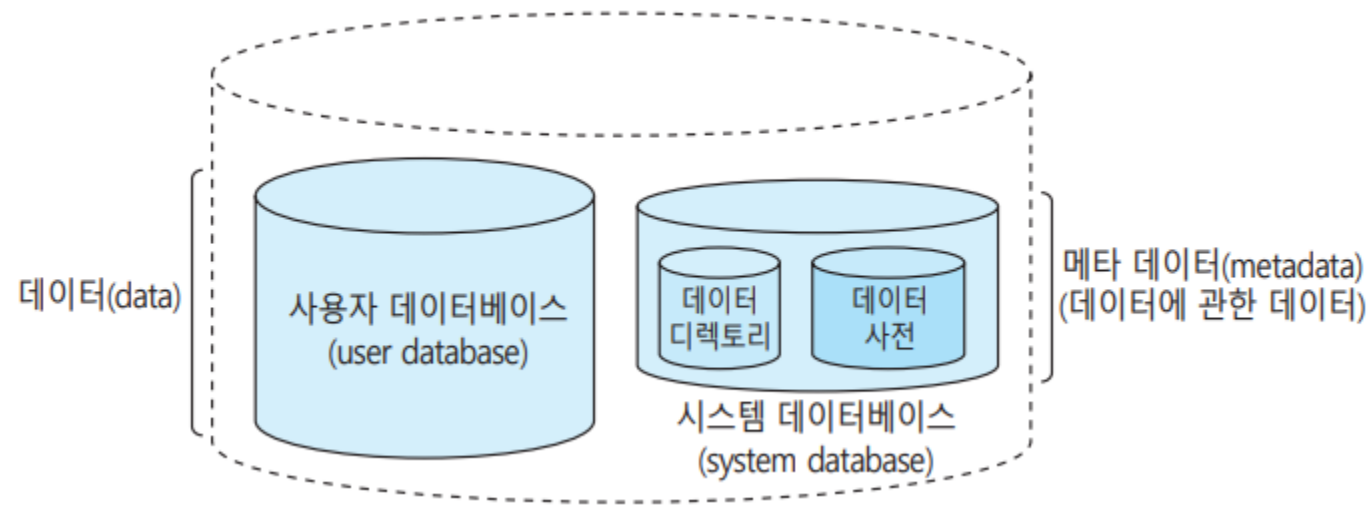
객체 관계형 데이터 모델

데이터 사전

데이터 사전

- * 데이터 사전(data dictionary)
- * 데이터 디렉토리(data directory)
- * 메타 데이터: 데이터에 관한 데이터

데이터 사전



(그림 출처: "'SQL과 NoSQL 기반의 데이터베이스 입문', 박성진, 생능, 2023)