

cse



# 데이터베이스 개론

## 2장 데이터베이스 관리시스템

김혜숙

## • 파일 시스템(file system)

- 데이터를 파일로 관리하기 위해 파일을 생성·삭제·수정·검색하는 기능을 제공하는 소프트웨어
- 응용 프로그램마다 필요한 데이터를 별도의 파일로 관리함
- 데이터 중복성과 데이터 종속성 문제가 발생

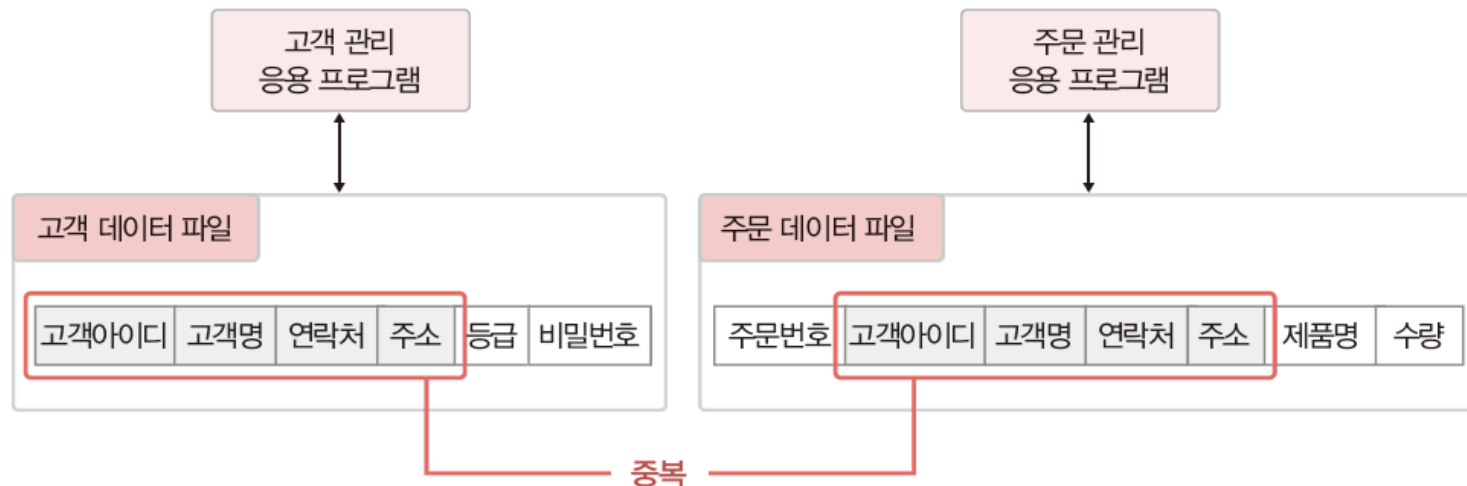


그림 2-1 파일 시스템에서의 데이터 관리

- 파일 시스템의 주요 문제점

- 같은 내용의 데이터가 여러 파일에 중복 저장된다 → 데이터 중복성
  - 저장 공간의 낭비는 물론 데이터 일관성과 데이터 무결성을 유지하기 어려움

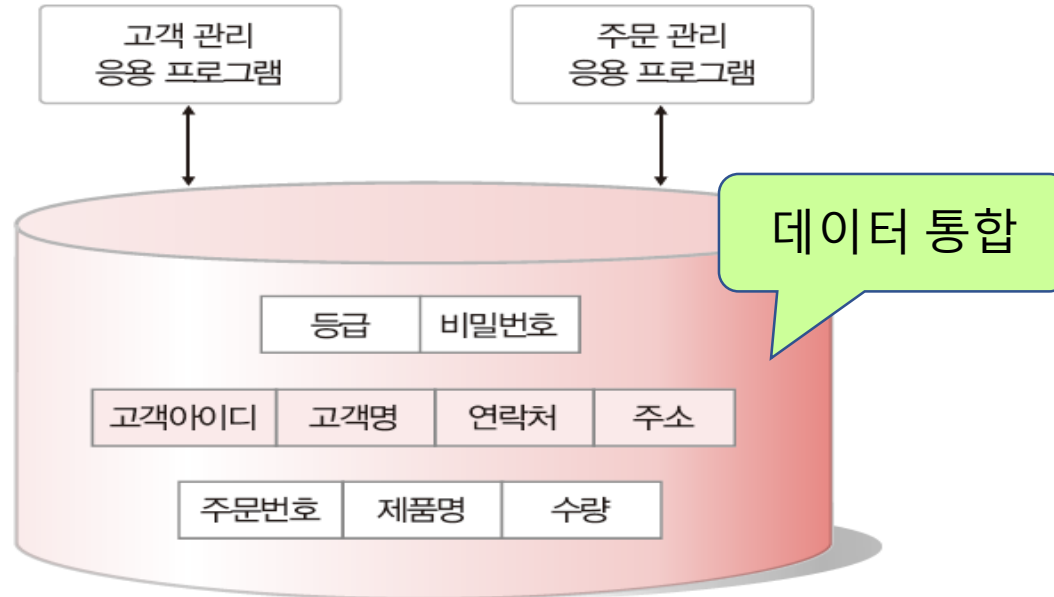


그림 2-2 파일 시스템의 데이터 중복성 문제를 해결하는 1차 방안

- 파일 시스템의 주요 문제점

- 응용 프로그램이 데이터 파일에 종속적이다 → 데이터 종속성
  - 사용하는 파일의 구조를 변경하면 응용 프로그램도 함께 변경해야 함

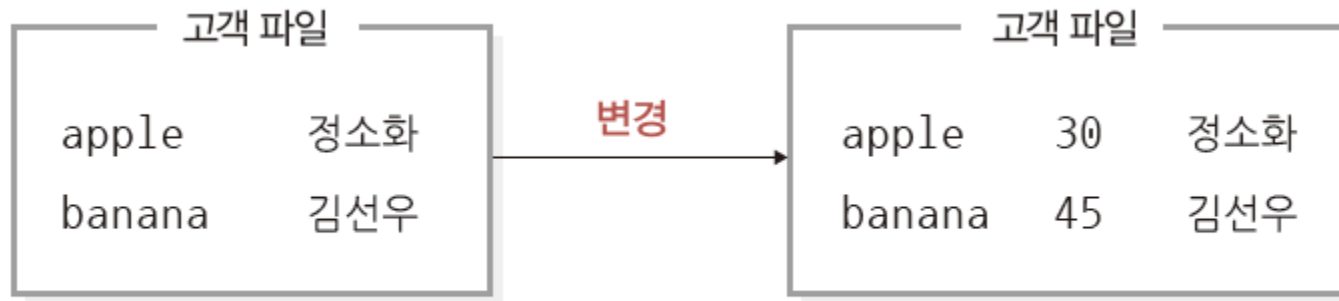


그림 2-3 파일 구조 변경 예

- 파일 시스템의 문제점

- 데이터 파일에 대한 동시 공유, 보안, 회복 기능이 부족하다
- 응용 프로그램 개발이 쉽지 않다

## • 데이터베이스 관리 시스템

- DBMS(DataBase Management System)
- 파일 시스템의 문제를 해결하기 위해 제시된 소프트웨어
- 조직에 필요한 데이터를 데이터베이스에 통합하여 저장하고 관리함

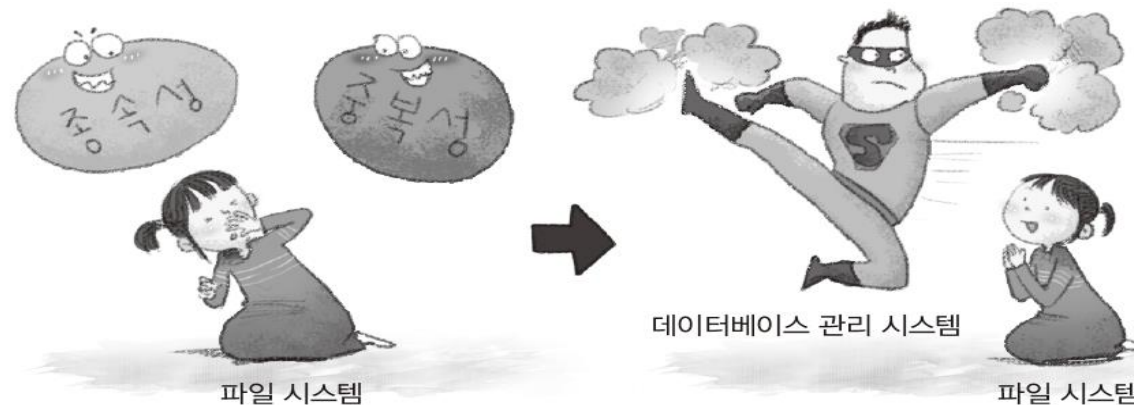


그림 2-4 파일 시스템과 데이터베이스 관리 시스템

- 데이터베이스 관리 시스템에서의 데이터 관리

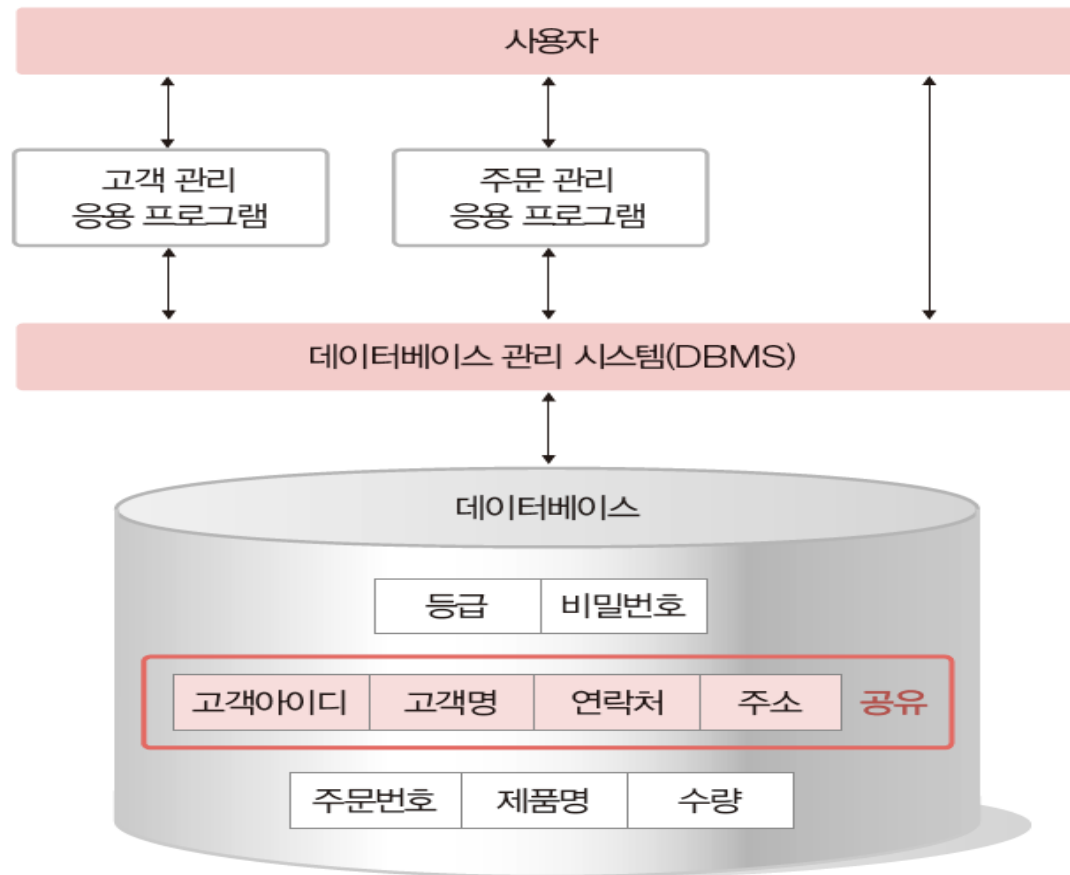


그림 2-5 데이터베이스 관리 시스템에서의 데이터 관리



## • 데이터베이스 관리 시스템의 주요 기능

정의 기능

데이터베이스 구조를 정의하거나 수정할 수 있다.

조작 기능

데이터를 삽입·삭제·수정·검색하는 연산을 할 수 있다.

제어 기능

데이터를 항상 정확하고 안전하게 유지할 수 있다.



# 03 데이터베이스 관리 시스템의 장단점

cse

## 장점

- ☐ 데이터 중복을 통제할 수 있다
- ☐ 데이터 독립성이 확보된다
- ☐ 데이터를 동시 공유할 수 있다
- ☐ 데이터 보안이 향상된다
- ☐ 데이터 무결성을 유지할 수 있다
- ☐ 표준화할 수 있다
- ☐ 장애 발생 시 회복이 가능하다
- ☐ 응용 프로그램 개발 비용이 줄어든다

## 단점

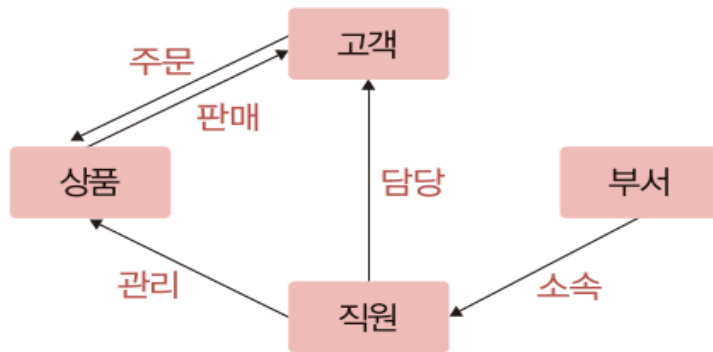
- ☐ 비용이 많이 든다
- ☐ 백업과 회복 방법이 복잡하다
- ☐ 중앙 집중 관리로 인한 취약점이 존재한다

그림 2-7 데이터베이스 관리 시스템의 장점과 단점

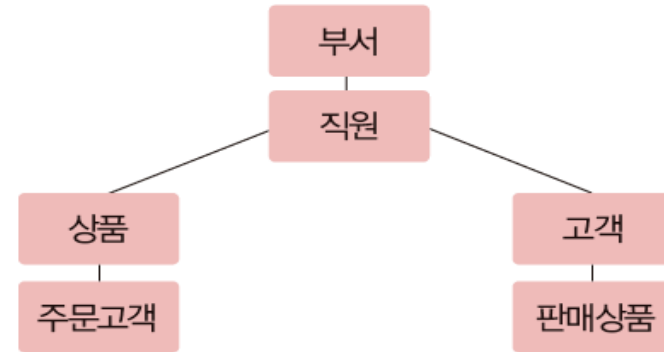
# 04 데이터베이스 관리 시스템의 발전 과정

## • 1세대(1960년대) : 네트워크 DBMS, 계층 DBMS

- 네트워크 DBMS : 데이터베이스를 그래프 형태로 구성
  - 예) IDS(Integrated Data Store)
- 계층 DBMS : 데이터베이스를 트리 형태로 구성
  - 예) IMS(Information Management System)



(a) 네트워크 DBMS



(b) 계층 DBMS

그림 2-8 1세대 DBMS 구조의 예

## • 2세대 : 관계 DBMS

- 관계 DBMS : 데이터베이스를 테이블 형태로 구성
  - 예) 오라클(Oracle), MS SQL 서버, 액세스(Access),  
인포믹스(Informix), MySQL

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

그림 2-9 관계 DBMS의 테이블 예 : 고객 테이블

- **3세대(1990) : 객체지향 DBMS, 객체관계 DBMS**

- 객체지향 DBMS : 객체를 이용해 데이터베이스를 구성
  - 예) 오투(O2), 온투스(ONTOS), 젬스톤(GemStone)
- 객체관계 DBMS : 객체 DBMS + 관계 DBMS

## • 4세대 : NoSQL • NewSQL DBMS

- NoSQL DBMS : 비정형 데이터를 처리하는데 적합하고 확장성이 뛰어남
  - 안정성과 일관성 유지를 위한 복잡한 기능을 포기
  - 데이터 구조를 미리 정해두지 않는 유연성
  - 확장성이 뛰어나 여러 대의 서버 컴퓨터에 데이터를 분산하여 저장하고 처리하는 환경에서 주로 사용
  - 예) 몽고디비(MongoDB), H베이스(HBase), 카산드라(Cassandra), 레디스(Redis), 네오포제이(Neo4j), 오리엔트DB(OrientDB) 등
- NewSQL DBMS: 관계 DBMS의 장점+ NoSQL의 확장성 및 유연성
  - 예) 구글 스패너(Spanner), 볼트DB(VoltDB), 누오DB(NuoDB)

cse



# 데이터베이스 개론

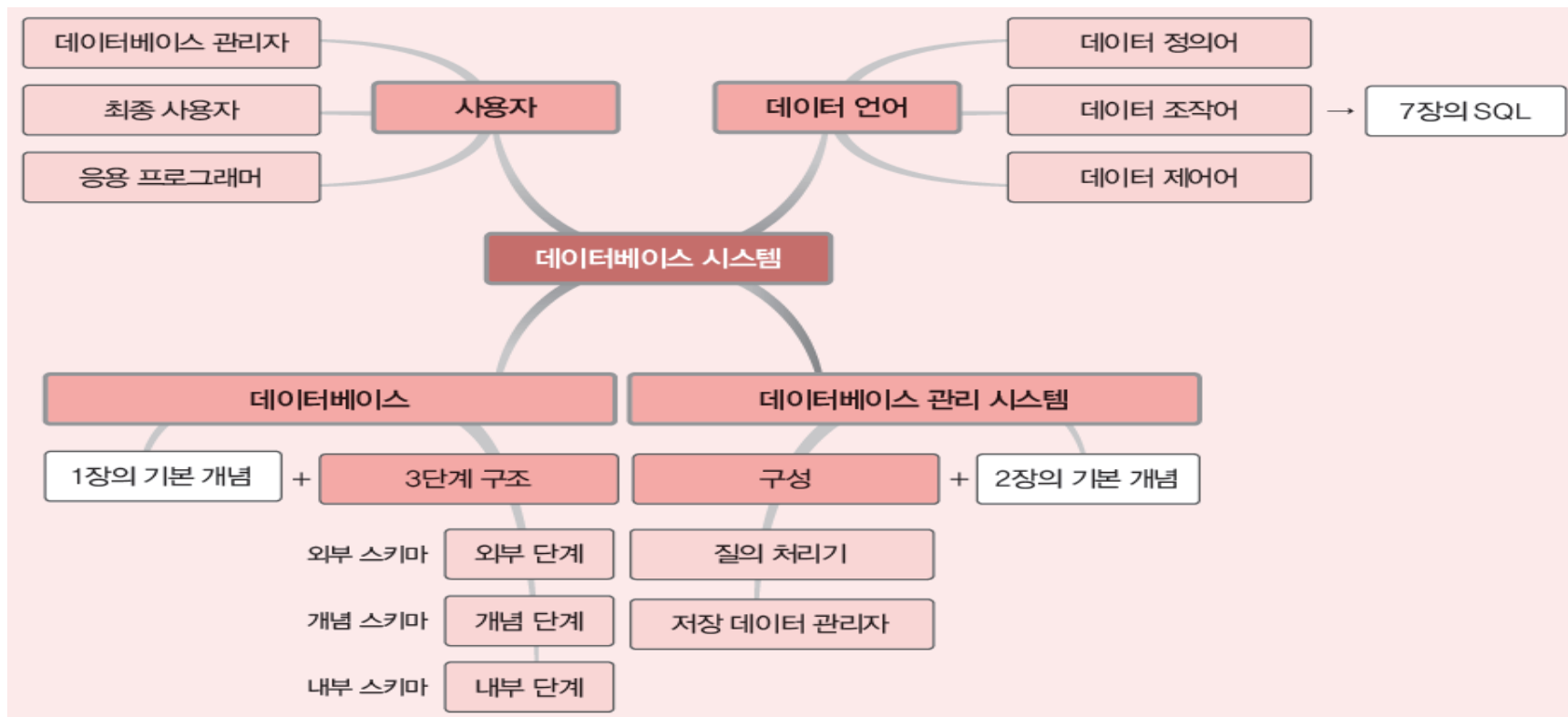
## 3장. 데이터베이스 시스템

김혜숙



## 학습내용

- 데이터베이스 시스템의 정의
- 데이터베이스의 구조
- 데이터베이스 사용자
- 데이터 언어
- 데이터베이스 관리 시스템의 구성



- ❖ 데이터베이스, 데이터베이스 관리 시스템, 데이터베이스 시스템의 차이를 이해한다.
- ❖ 데이터베이스 시스템의 구성 요소를 살펴본다.
- ❖ 데이터베이스 3단계 구조에서 데이터 독립성의 개념을 실현하는 방법을 이해한다.
- ❖ 데이터 언어별 특징을 알아본다.
- ❖ 데이터베이스 사용자별 특징을 알아본다.
- ❖ 데이터베이스 관리 시스템의 구성을 알아본다.





- **데이터**

- 현실 세계에서 단순히 관찰하거나 측정하여 수집한 사실이나 값

- **데이터베이스**

- 특정 조직의 여러 사용자가 공유하여 사용할 수 있도록 통합해서 저장한 운영 데이터의 집합

- **데이터 베이스 관리 시스템**

- 조직에 필요한 데이터를 데이터베이스에 통합하여 저장하고 관리하는 소프트웨어 패키지

- **데이터베이스 시스템**

- 데이터베이스에 데이터를 저장하고, 이를 관리하여 조직에 필요한 정보를 생성해주는 시스템

## • 데이터베이스 시스템(DBS; DataBase System)

- 데이터베이스에 데이터를 저장하고, 이를 관리하여 조직에 필요한 정보를 생성해주는 시스템

### • 구성요소

- 사용자
  - 데이터베이스 관리자, 최종 사용자, 응용 프로그래머
- 데이터 언어
  - 데이터 정의어, 데이터 조작어, 데이터 제어어
- 데이터베이스 관리 시스템
- 데이터베이스
- 컴퓨터

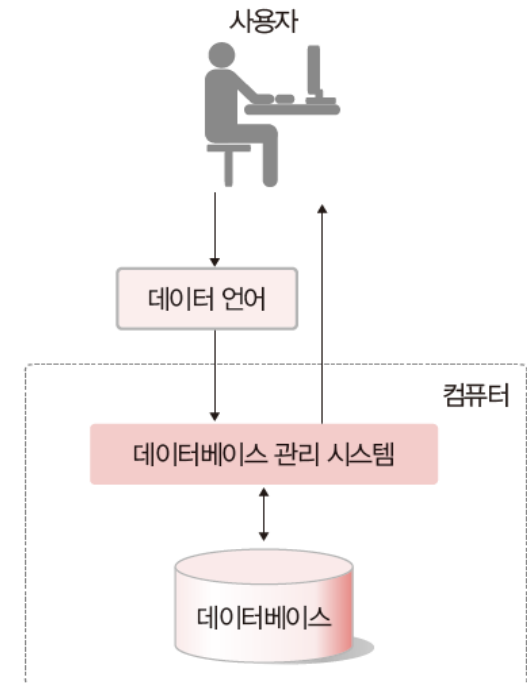


그림 3-1 데이터베이스 시스템의 구성

- 스키마와 인스턴스

- 스키마(schema)
  - 데이터베이스에 저장되는 데이터 구조와 제약조건을 정의한 것
- 인스턴스(instance)
  - 스키마에 따라 데이터베이스에 실제로 저장된 값

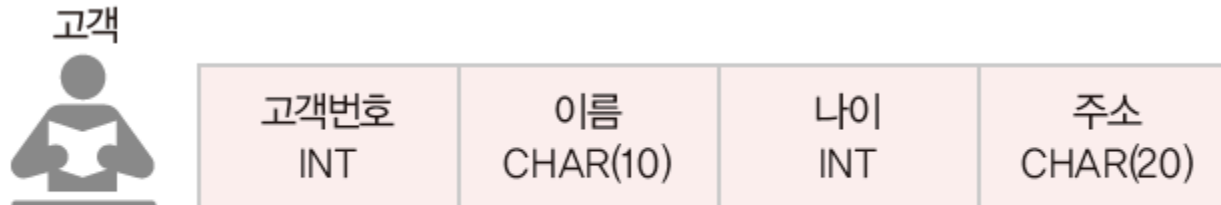


그림 3-2 스키마의 예

# 02 데이터베이스의 구조



## ■ 회원

스키마

회원ID	이름	주민등록번호	패스워드	우편번호	주소	휴대폰
eun4814	은수정	7909251645678	eun4814	706-130	대구시 수성구 대흥동	010-2564-8984
gdyoo	유경동	5810061454321	gdyoo	320-050	충남 논산시 대교동	017-954-1564
geunsugi	황근의	7812252846512	gugu	443-400	경기도 수원시 영통구 망포동	011-897-7613
jungh24	박정희	7302161846621	jungh24	140-120	서울시 강남구 신사동	01-08654-9876
kkamwl	신미원	7508232462035	kkam1234	121-020	서울시 마포구 공덕동	019-745-4139
ksy0416	김수영	7003011564328	blog8834	464-050	경기도 광주시 송정동	010-7462-5461
miyoung	이미영	7605082645731	my76	441-460	경기도 수원시 권선구 금곡동	010-5256-1358
quffid10	김창용	8106241346454	dreamy	135-240	서울시 강남구 개포동	010-8965-4135
sharon98	정은상	7811192126461	worth19	655-150	광주시 북구 일곡동	011-4436-8562
tcbnukyo	김윤수	6507141845234	kys65	330-280	충남 천안시 신당동	016-8461-2846
yunhui6262	정윤희	6811302648756	yun6262	413-050	경기도 파주시 검산동	010-8452-1878

인스턴스

## • 3단계 데이터베이스 구조

- 미국 표준화 기관인 ANSI/SPARC에서 제안
  - 데이터베이스의 구조는 복잡하나 내부구조와 동작방식을 쉽게 이해할 수 있는 개념
- 데이터베이스를 쉽게 이해하고 이용할 수 있도록 하나의 데이터베이스를 관점에 따라 세 단계로 나눈 것
  - 외부 단계(external level) : 개별 사용자 관점
  - 개념 단계(conceptual level) : 조직 전체의 관점
  - 내부 단계(internal level) : 물리적인 저장 장치의 관점
- 각 단계별로 다른 추상화(abstraction) 제공
  - 내부 단계에서 외부 단계로 갈수록 추상화 레벨이 높아짐

# 02 데이터베이스의 구조

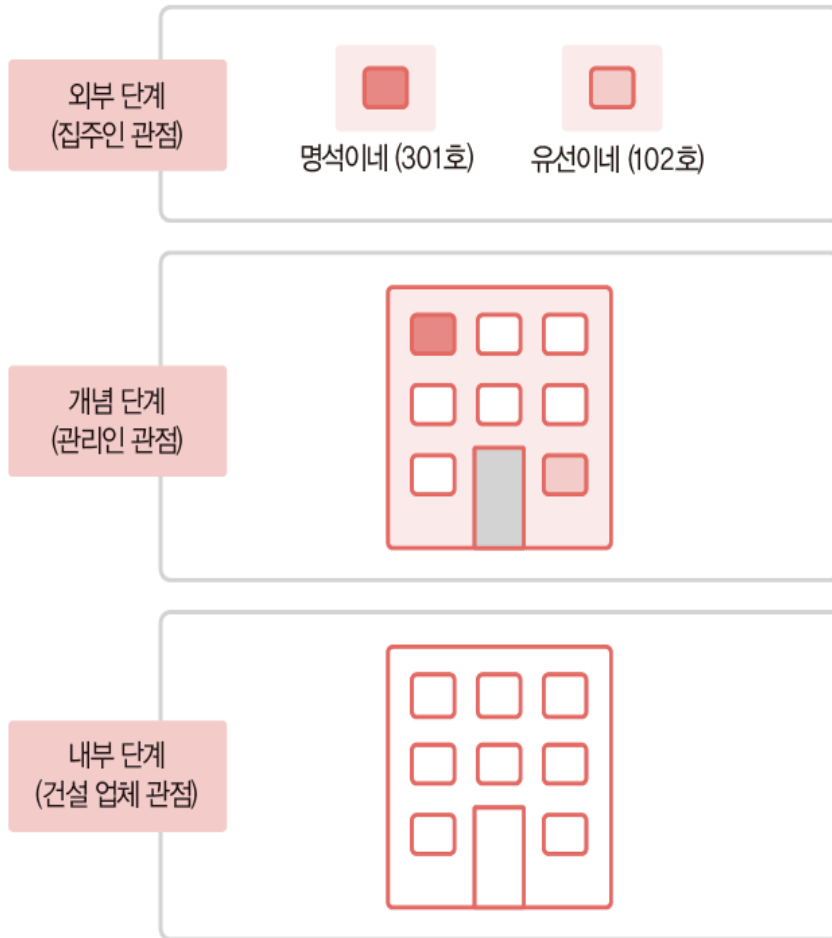


그림 3-3 3단계 데이터베이스 구조의 개념

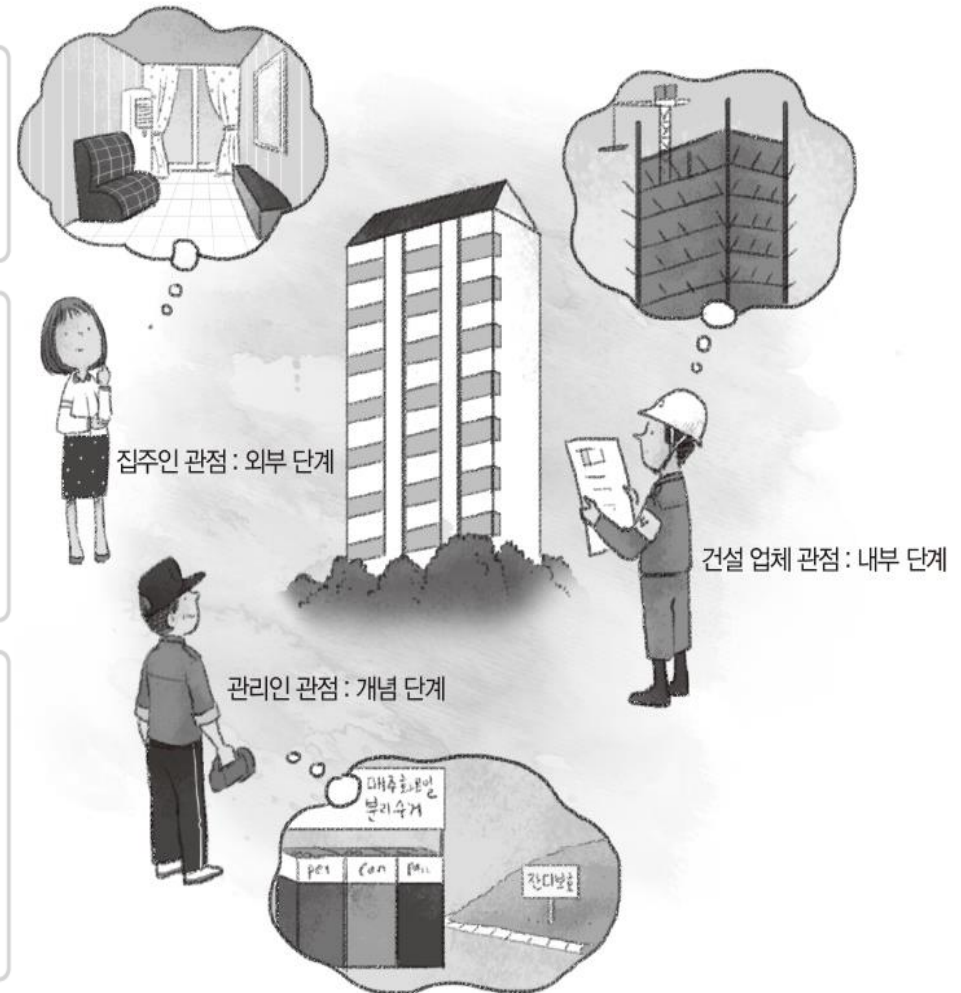


그림 3-4 3단계 데이터베이스 구조의 이해

## • 3단계 데이터베이스 구조 : 외부 단계

- 데이터베이스를 개별 사용자 관점에서 이해하고 표현하는 단계
- 외부 스키마(external schema)
  - 외부 단계에서 사용자에게 필요한 데이터베이스를 정의한 것
  - 각 사용자가 생각하는 데이터베이스의 모습, 즉 논리적 구조로 사용자마다 다름
  - 전체 데이터베이스 중 사용자가 관심을 갖는 일부분임
  - 서브 스키마(sub schema)라고도 함
- 데이터베이스 하나에 외부 스키마가 여러 개 존재할 수 있음

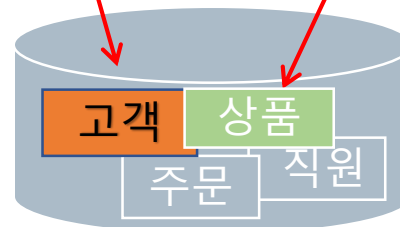
이름	나이	성별	등급
김마리	22	여	GOLD

고객  
관리팀

상품  
관리팀

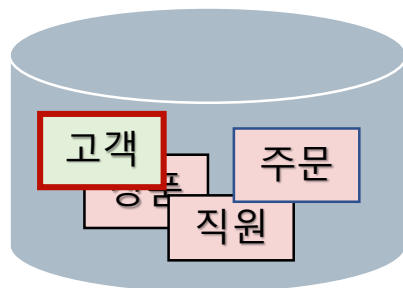
상품명	용량	단가	제조사
참치캔	135G	800	동원산업

쇼핑몰 운영



## • 3단계 데이터베이스 구조 : 개념 단계

- 데이터베이스를 조직 전체의 관점에서 이해하고 표현하는 단계
- 개념 스키마(conceptual schema)
  - 개념 단계에서 데이터베이스 전체의 논리적 구조를 정의한 것
  - 조직 전체의 관점에서 생각하는 데이터베이스의 모습
  - 전체 데이터베이스에 어떤 데이터가 저장되는지, 데이터들 간에는 어떤 관계가 존재하고 어떤 제약조건이 있는지에 대한 정의뿐만 아니라, 데이터에 대한 보안 정책이나 접근 권한에 대한 정의도 포함
- 데이터베이스 하나에 개념 스키마가 하나만 존재함





## • 3단계 데이터베이스 구조 : 내부 단계

- 데이터베이스를 저장 장치의 관점에서 이해하고 표현하는 단계
- 내부 스키마(internal schema)
  - 전체 데이터베이스가 저장 장치에 실제로 저장되는 방법을 정의한 것
  - 레코드 구조, 필드 크기, 레코드 접근 경로 등 물리적 저장 구조를 정의
  - 개념 스키마에 대한 물리적인 저장 구조를 표현한 것
- 데이터베이스 하나에 내부 스키마가 하나만 존재함

레코드 길이 : 70바이트

내부 단계  
(저장 장치 관점)

내부 스키마

필드 이름	필드 크기	OFFSET	인덱스
번호	4바이트	0	존재
이름	10바이트	4	
성별	2바이트	14	
나이	4바이트	16	
직업	10바이트	20	
주소	20바이트	30	
연락처	20바이트	50	존재

# 02 데이터베이스의 구조

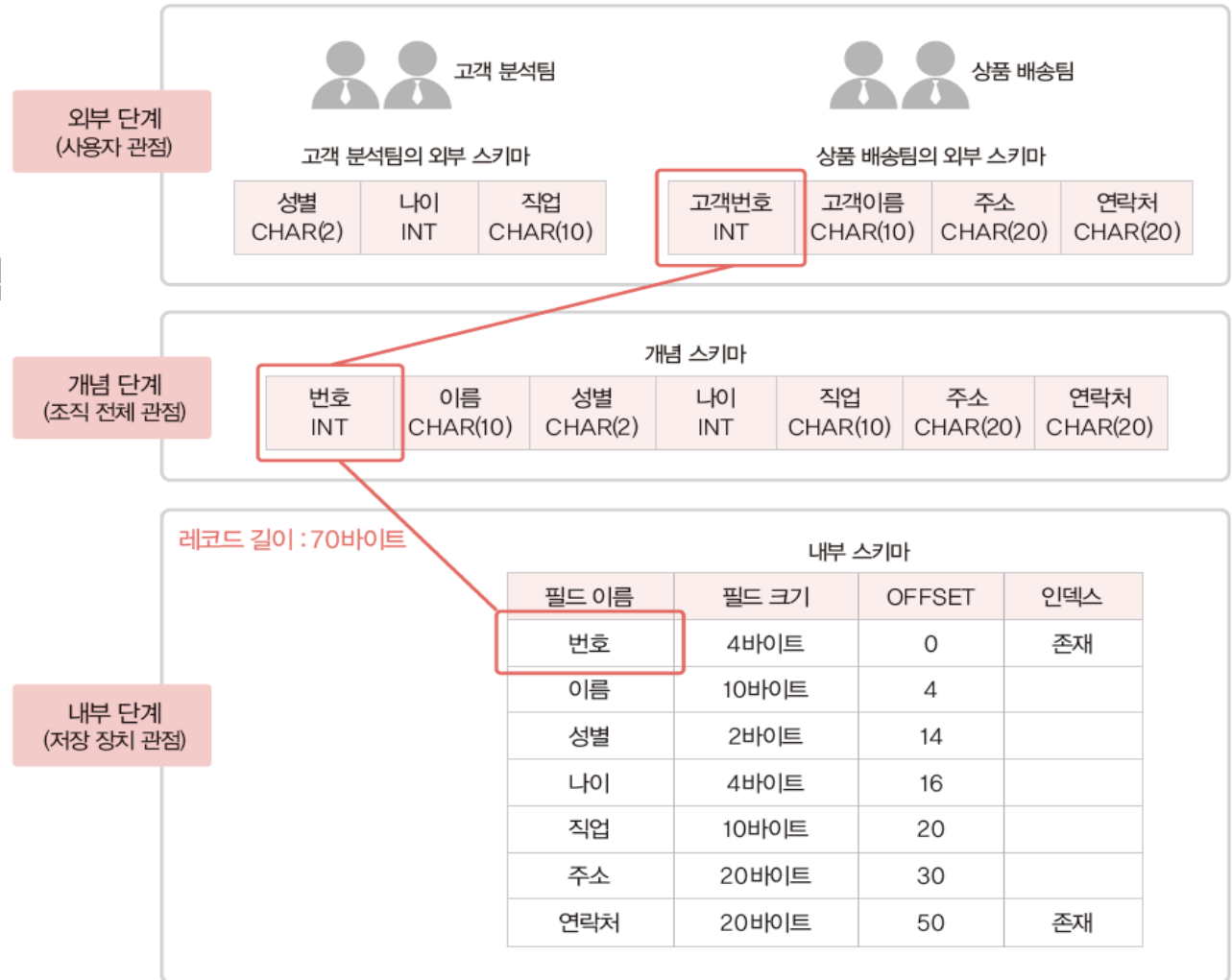
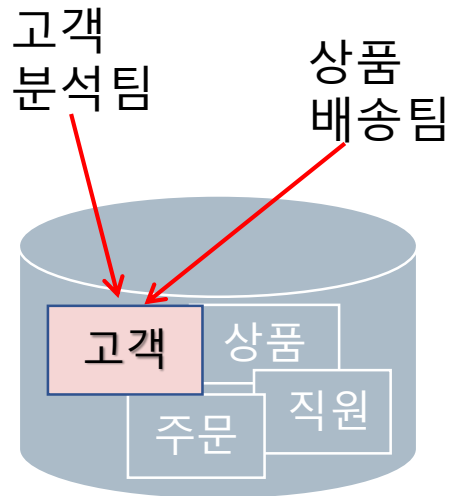


그림 3-5 3단계 데이터베이스 구조의 예



- **데이터 독립성(data independency)**

- 하나의 데이터베이스에는 세 가지 유형의 스키마가 존재
- 데이터베이스를 바라보는 관점이 다른데 모두 같은 데이터베이스를 표현
- 사용자가 자신의 외부 스키마를 통해 데이터를 얻으려면 내부 스키마에 따라 저장된 데이터베이스에 접근
- 스키마 사이에는 유기적 대응 관계가 성립해야 함
- 사상 또는 매핑

- 데이터 독립성(data independency)

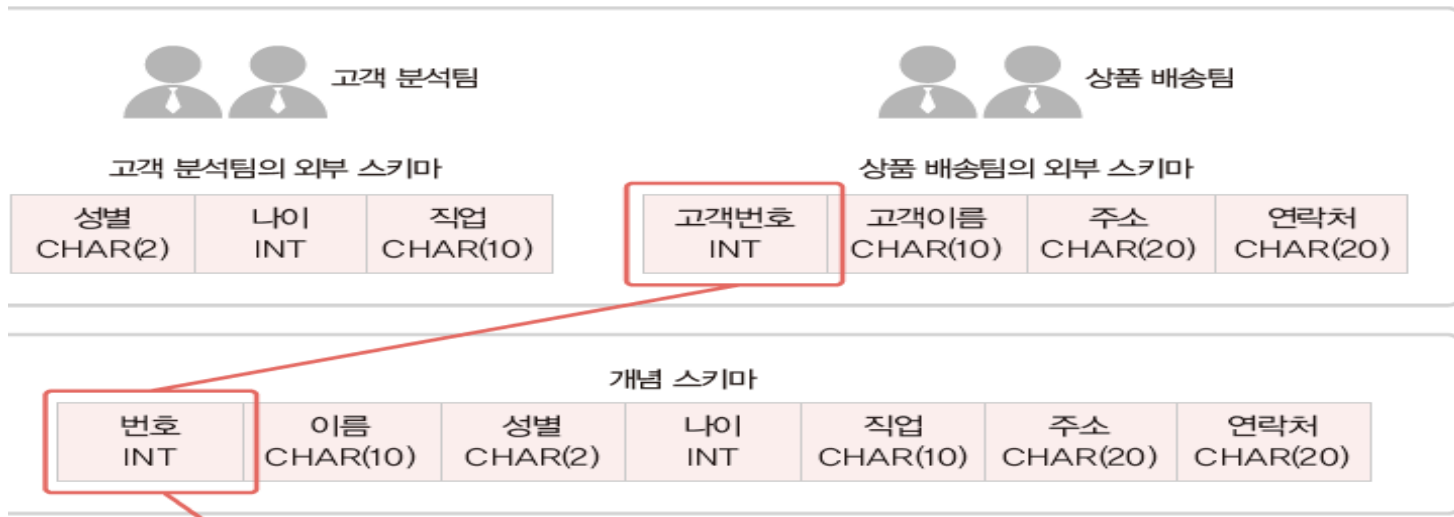
- 3단계 데이터베이스 구조의 사상 또는 매핑
- 스키마 사이의 대응 관계
  - 외부/개념 사상 : 외부 스키마와 개념 스키마의 대응 관계
    - 응용 인터페이스(application interface)라고도 함
  - 개념/내부 사상 : 개념 스키마와 내부 스키마의 대응 관계
    - 저장 인터페이스(storage interface)라고도 함
- 미리 정의된 사상 정보를 이용하여 사용자가 원하는 데이터에 접근

데이터베이스를 3단계 구조로 나누고 단계별로 스키마를 유지하며  
스키마 사이의 대응 관계를 정의하는 궁극적인 목적은

➔ **데이터 독립성의 실현**

## • 데이터 독립성(data independency)

- 데이터베이스 관리 시스템(DBMS)의 중요한 장점이나 필요한 이유
- 하위 스키마를 변경하더라도 상위 스키마가 영향을 받지 않는 특성
- **논리적 데이터 독립성**
  - 개념 스키마가 변경되어도 외부 스키마는 영향을 받지 않음
  - 개념 스키마가 변경되면 관련된 외부/개념 사상만 정확하게 수정해주면 됨



- 데이터 독립성(data independency)

- 물리적 데이터 독립성

- 내부 스키마가 변경되어도 개념 스키마는 영향을 받지 않음
    - 내부 스키마가 변경되면 관련된 개념/내부 사상만 정확하게 수정해주면 됨

개념 스키마						
번호 INT	이름 CHAR(10)	성별 CHAR(2)	나이 INT	직업 CHAR(10)	주소 CHAR(20)	연락처 CHAR(20)

≡ 길이 : 70바이트

내부 스키마			
필드 이름	필드 크기	OFFSET	인덱스
번호	4바이트	0	존재
이름	10바이트	4	
성별	2바이트	14	
나이	4바이트	16	
직업	10바이트	20	
주소	20바이트	30	
연락처	20바이트	50	존재

## 02 데이터베이스의 구조

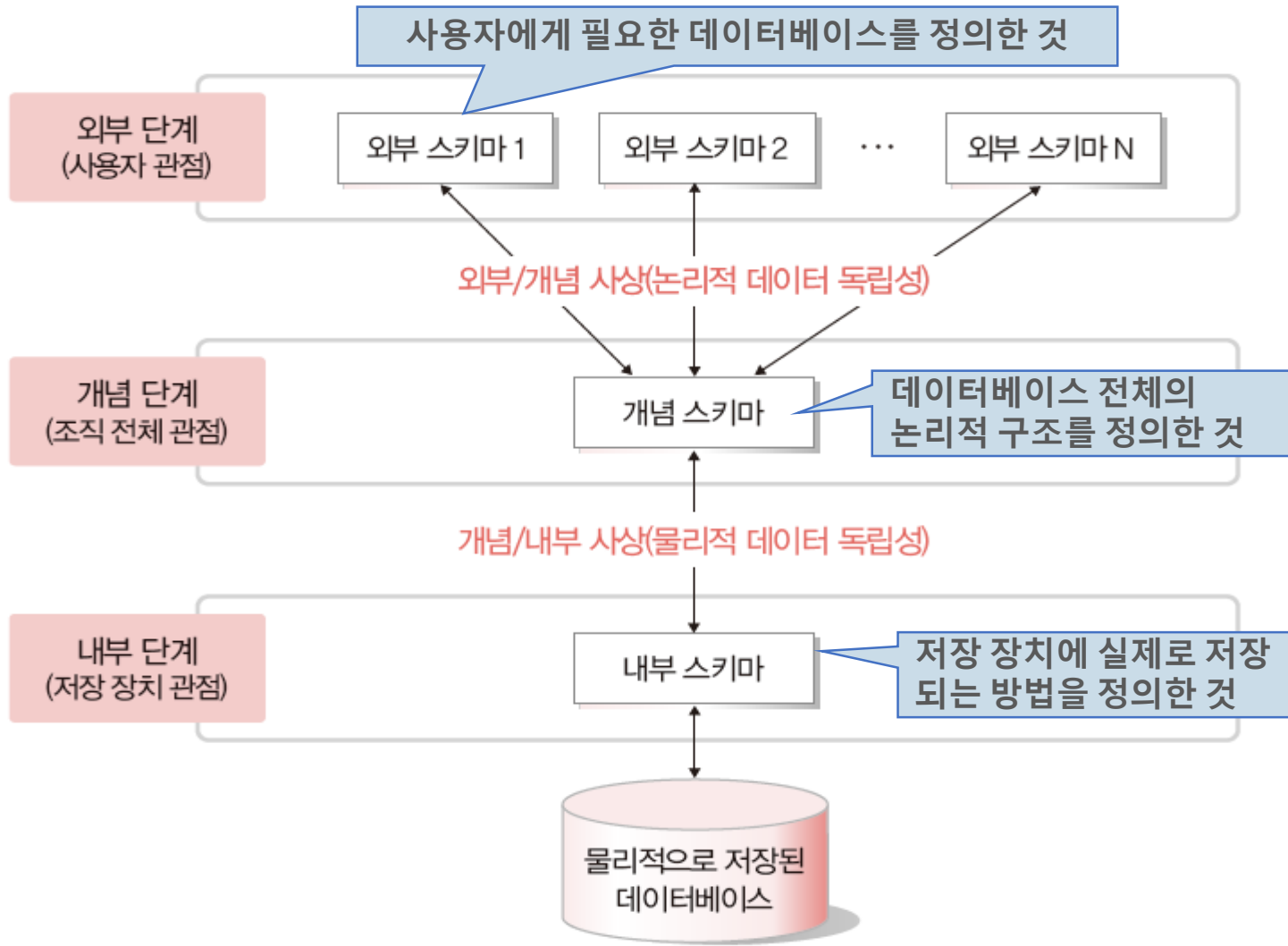


그림 3-6 3단계 데이터베이스 구조에서 스키마 간의 사상

# 02 데이터베이스의 구조

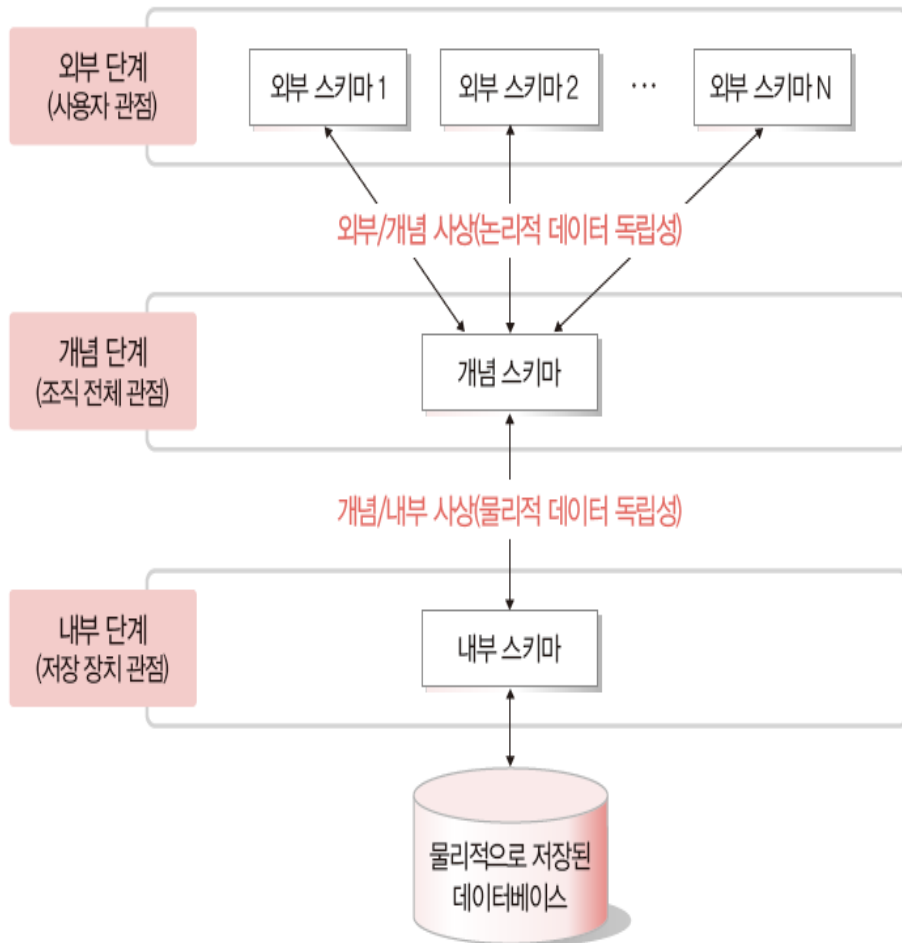


그림 3-6 3단계 데이터베이스 구조에서 스키마 간의 사상

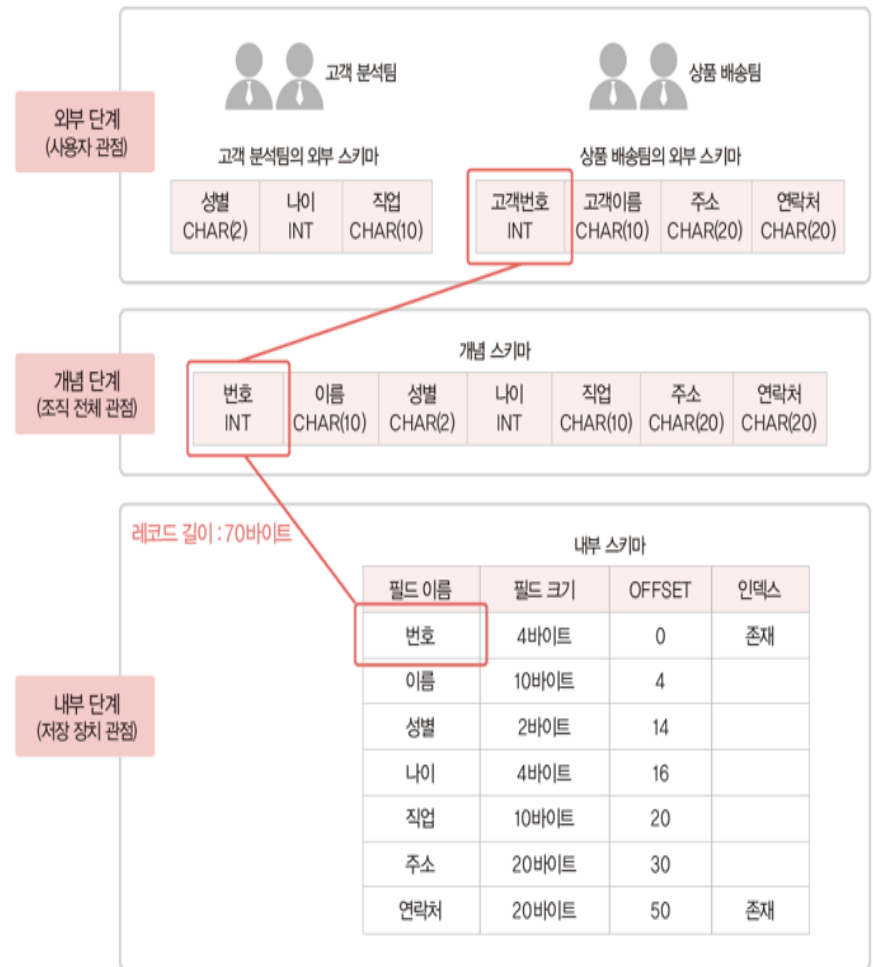


그림 3-5 3단계 데이터베이스 구조의 예





- **데이터베이스**

- 실제 데이터외에 저장된 데이터를 관리하기 위한 부가 정보도 저장해야 함
- 실제 data + 부가 data(스키마, 사상 정보)

- **사용자 데이터베이스 (user database)**

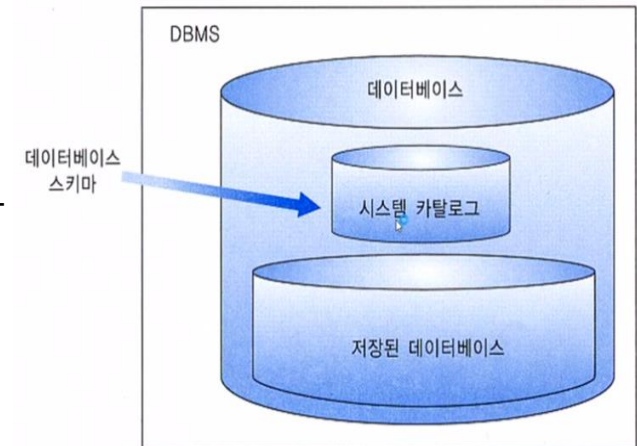
- 사용자가 실제로 이용하는 데이터가 저장되어 있는 일반 데이터베이스

- **시스템 데이터베이스(system database)**

- 데이터베이스 관리 시스템이 사용하기 위한 부가 데이터가 저장되어 있는 데이터베이스
- 데이터 사전, 데이터 디렉터리

## • 데이터 사전(data dictionary)

- 시스템 카탈로그(system catalog)라고도 함
- 데이터베이스에 저장되는 데이터에 관한 정보, 즉 메타 데이터를 유지하는 시스템 데이터베이스
  - 메타 데이터(meta data) : 데이터에 대한 데이터
- 스키마, 사상 정보, 다양한 제약조건 등을 저장
- 데이터베이스 관리 시스템이 스스로 생성하고 유지함
- 일반 사용자도 접근이 가능하지만 저장 내용을 검색만 할 수 있음



시스템 카탈로그와 저장된 데이터베이스

## • 데이터 디렉터리(data directory)

- 데이터 사전에 있는 데이터에 실제로 접근하는 데 필요한 위치 정보를 저장하는 시스템 데이터베이스
- 일반 사용자의 접근은 허용되지 않음



1. 다음 설명에서 ㉠과 ㉡가 각각 무엇인지 답하시오.

( ㉠ )는 데이터베이스에 저장되는 데이터 구조와 제약조건을 정의한 것이다.  
그리고 ( ㉠ )에 따라 데이터베이스에 실제로 저장된 값을 ( ㉡ )라고 한다.

2. 다음 설명에서 ㉠가 무엇인지 답하시오.

3단계 데이터베이스 구조에서 ( ㉠ )는 개념 스키마와 내부 스키마의 대응 관계를 정의한 것으로, 저장 인터페이스라고도 한다.

3. 다음 설명에서 ㉠과 ㉡가 각각 무엇인지 답하시오.

( ㉠ )는 데이터베이스에 저장되는 데이터에 관한 정보를 저장하는 곳으로, 스키마, 사상 정보, 다양한 제약조건 등을 저장한다. 그리고 ( ㉠ )에 저장되어 있는 정보에 실제로 접근하는 데 필요한 위치 정보는 ( ㉡ )에서 관리한다.

4. 데이터 독립성의 의미를 설명하고, 3단계 데이터베이스 구조에서 데이터 독립성을 실현하는 방법을 설명하시오.

## • 데이터베이스 사용자

- 데이터베이스를 이용하기 위해 접근하는 모든 사람
- 이용 목적에 따라 데이터베이스 관리자, 최종 사용자, 응용 프로그래머로 구분

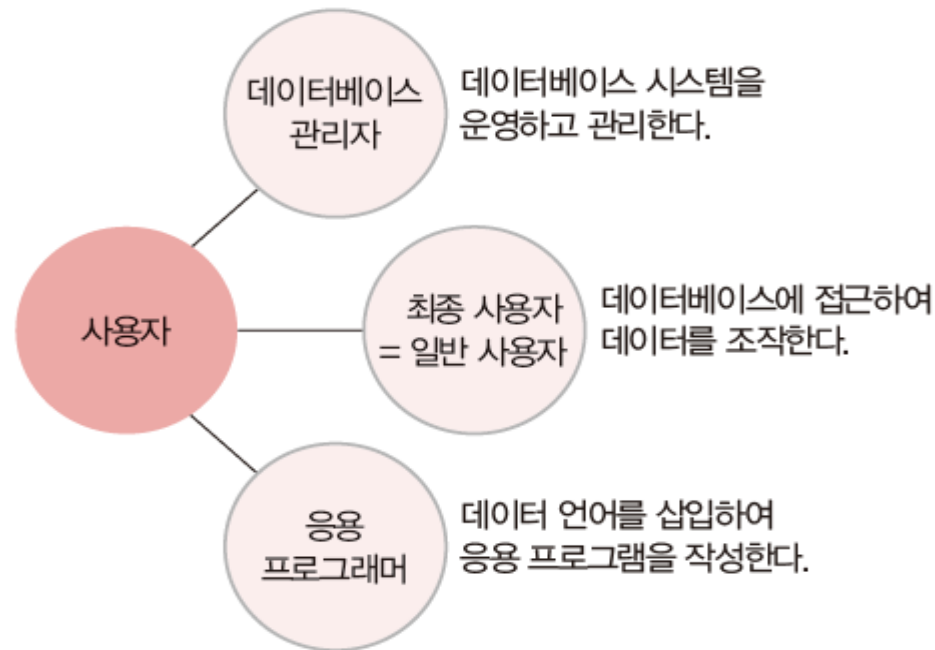


그림 3-7 데이터베이스 사용자

## • 데이터베이스 관리자(DBA; DataBase Administrator)

- 데이터베이스 시스템을 운영 및 관리하는 사람
- 주로 데이터 정의어와 데이터 제어어를 이용
- 주요 업무
  - 데이터베이스 구성 요소 선정
  - 데이터베이스 스키마 정의
  - 물리적 저장 구조와 접근 방법 결정
  - 무결성 유지를 위한 제약조건 정의
  - 보안 및 접근 권한 정책 결정
  - 백업 및 회복 기법 정의
  - 시스템 데이터베이스 관리
  - 시스템 성능 감시 및 성능 분석
  - 데이터베이스 재구성

- **최종 사용자(end user)**

- 데이터베이스에 접근하여 데이터를 조작(삽입·삭제·수정·검색)하는 사람
- 주로 데이터 조작어를 사용
- 캐주얼 사용자와 초보 사용자로 구분

- **응용 프로그래머(application programmer)**

- 데이터 언어를 삽입하여 응용 프로그램을 작성하는 사람
- 주로 데이터 조작어를 사용

응용 프로그래머



최종 사용자

그림 3-8 최종 사용자와 응용 프로그래머의 예

## • 데이터 언어

데이터베이스 관리 시스템의 세 가지 주요 기능:

데이터베이스 구조를 정의하거나 수정할 수 있는 **정의 기능**,  
데이터를 검색·삽입·삭제·수정하는 연산을 할 수 있는 **조작 기능**,  
데이터를 항상 정확하고 안전하게 유지할 수 있는 **제어 기능**을 제공한다.

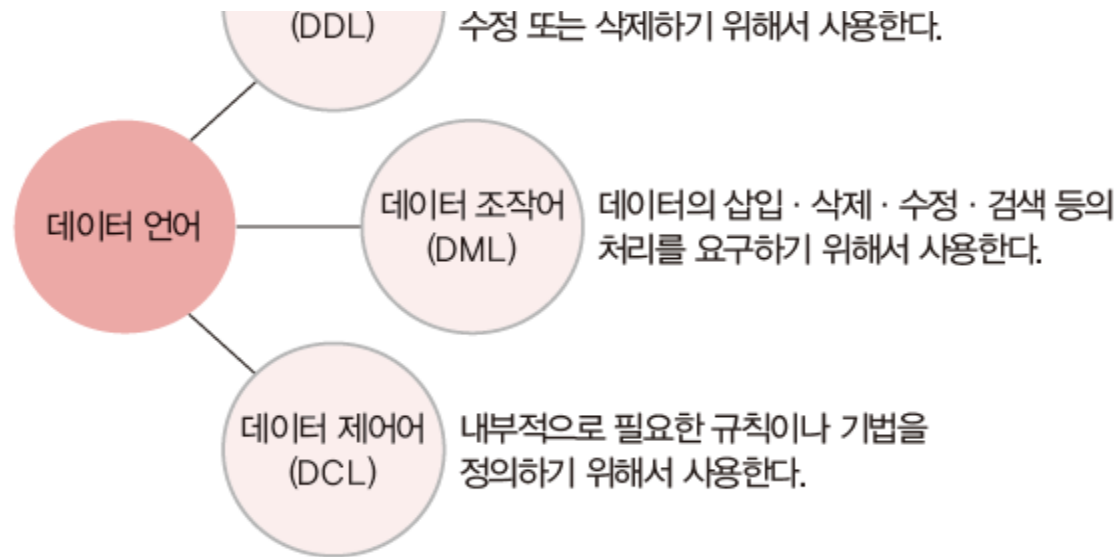


그림 3-9 데이터 언어의 종류와 용도



## • 데이터 정의어(DDL; Data Definition Language)

- 새로운 데이터 베이스를 구축하기 위해 스키마의를 정의
- 기존 스키마의 정의를 수정 또는 삭제하기 위해 사용
- 데이터 정의어로 명시된 문장이 입력되면 DBMS는 사용자가 정의한 스키마에 대한 명세를 데이터 사전에 저장
- 데이터 정의어의 기본적인 기능
  - 데이터 모델에서 지원하는 데이터 구조를 생성 : SQL에서 CREATE TABLE
  - 데이터 구조의 변경 : SQL에서 ALTER TABLE
  - 데이터 구조의 삭제 : SQL에서 DROP TABLE
  - 데이터 접근을 위해 특정 속성에 인덱스를 정의 : SQL에서 CREATE INDEX

- **데이터 조작어(DML; Data Manipulation Language)**

- 데이터의 삽입·삭제·수정·검색 등의 처리를 요구하기 위해 사용
- 절차적 데이터 조작어와 비절차적 데이터 조작어로 구분
  - 절차적 데이터 조작어(procedural DML)
    - 사용자가 어떤(what) 데이터를 원하고 그 데이터를 얻기 위해 어떻게(how) 처리해야 하는지도 설명
  - 비절차적 데이터 조작어(nonprocedural DML)
    - 사용자가 어떤(what) 데이터를 원하는지만 설명
    - 선언적 언어(declarative language)라고도 함



두부를 사오는 방법까지 구체적으로 알려주는 심부름 상황 : **절차적 조작어**



단순히 두부를 사오라고만 지시하는 심부름 상황 : **비절차적 조작어**

그림 3-10 절차적 데이터 조작어와 비절차적 데이터 조작어의 이해



- **데이터 제어어(DCL; Data Control Language)**

- 내부적으로 필요한 규칙이나 기법을 정의하기 위해 사용
- 사용 목적
  - 무결성 : 정확하고 유효한 데이터만 유지
  - 보안 : 허가받지 않은 사용자의 데이터 접근 차단, 허가된 사용자에게 권한 부여
  - 회복 : 장애가 발생해도 데이터 일관성 유지
  - 동시성 제어 : 같은 데이터를 동시 공유할 수 있도록 지원



- **데이터베이스 관리 시스템**

- 데이터베이스 관리와 사용자의 데이터 처리 요구 수행
- 주요 구성 요소
  - 질의 처리기(query processor)
    - 사용자의 데이터 처리 요구를 해석하여 처리
    - DDL 컴파일러, DML 프리 컴파일러, DML 컴파일러, 런타임 데이터베이스 처리기, 트랜잭션 관리자 등을 포함
  - 저장 데이터 관리자(stored data manager)
    - 디스크에 저장된 데이터베이스와 데이터 사전을 관리하고 접근함

# 05 데이터베이스 관리 시스템의 구성

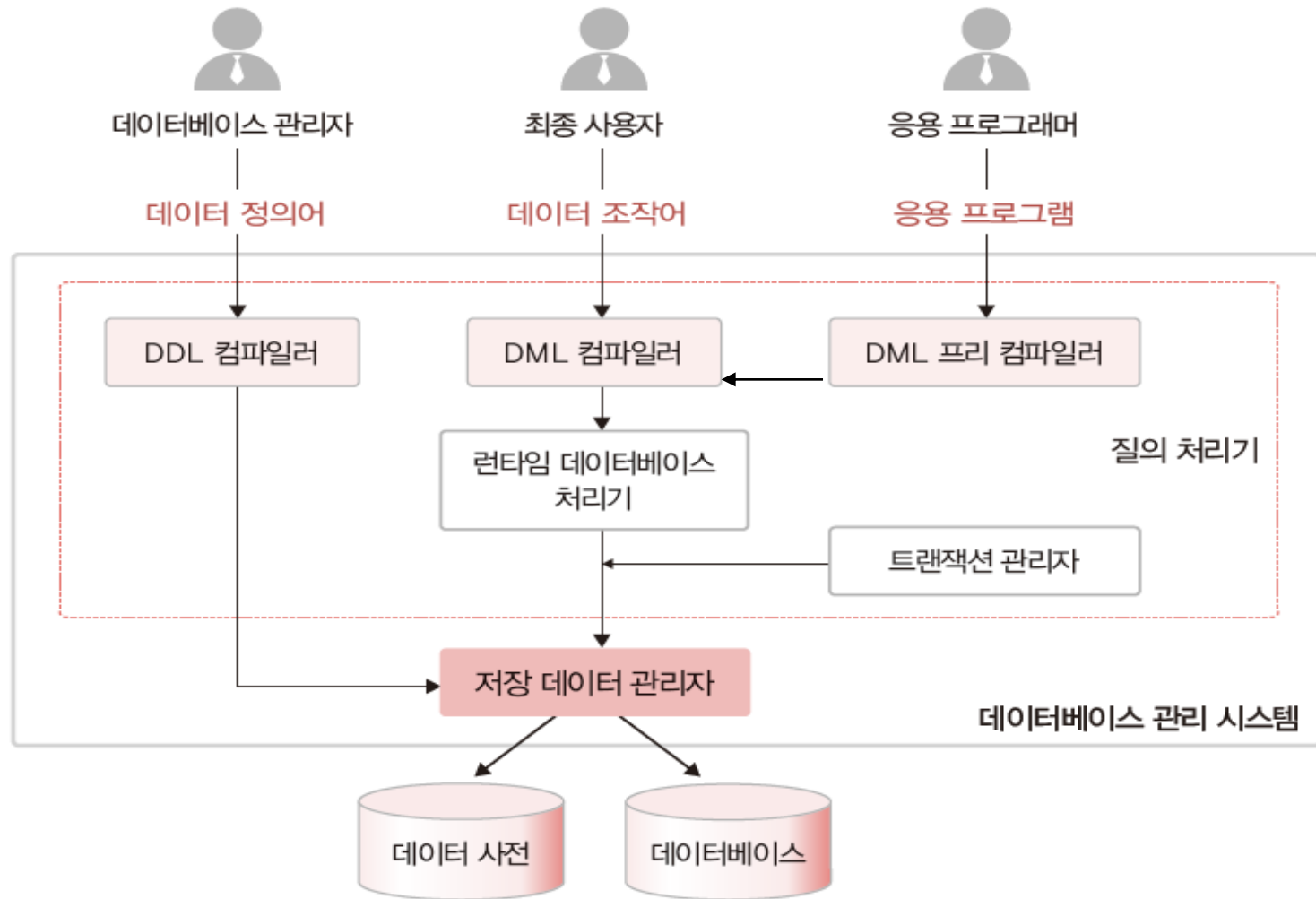


그림 3-11 데이터베이스 관리 시스템의 구성

## • 질의 처리기

### • DDL 컴파일러

- 데이터 정의어로 작성된 스키마의 정의를 해석
- 저장 데이터 관리자의 도움을 받아 새로운 데이터베이스 구축, 스키마의 정의를 데이터사전에 저장, 기존 스키마의 삭제나 수정 요청도 처리하여 변경된 내용을 데이터 사전에 적용

### • DML 프리 컴파일러

- 응용 프로그램에 삽입된 데이터 조작어를 추출하여 DML 컴파일러에 전달

### • DML 컴파일러

- 데이터 조작어로 작성된 데이터의 처리 요구를 분석하여 런타임 데이터베이스 처리기가 이해할 수 있도록 해석

### • 런타임 데이터베이스 처리기

- 저장 데이터 관리자를 통해 데이터베이스에 접근, DML 컴파일러로부터 받은 데이터 처리요구를 데이터베이스에서 실제로 실행

### • 트랜잭션 관리자

- 데이터베이스에 접근하는 과정에서 사용자의 접근 권한의 유효성 검사, 제약조건 위반 여부 확인, 회복, 병행과 관련한 작업 수행

# cse

## 수고하셨습니다

