



1 데이터베이스

학습목표

- 데이터베이스의 정의와 기본 개념을 이해한다.
- 데이터베이스의 종류를 살펴보고 장점과 단점을 비교한다.
- 데이터베이스의 구성 요소를 살펴본다



목차

- ✓ 데이터와 정보
- ✓ 데이터베이스와 데이터베이스 관리 시스템
- ✓ 데이터베이스 특징
- ✓ 데이터베이스 사용자
- ✓ 데이터베이스 언어
- ✓ 데이터베이스 관리 시스템
- ✓ 데이터베이스 관리 시스템의 종류
- ✓ 계층형 데이터베이스
- ✓ 네트워크형 데이터베이스
- ✓ 관계형 데이터베이스
- ✓ 객체 지향형 데이터베이스와 객체 관계형 데이터베이스
- ✓ 관계형 데이터베이스 용어

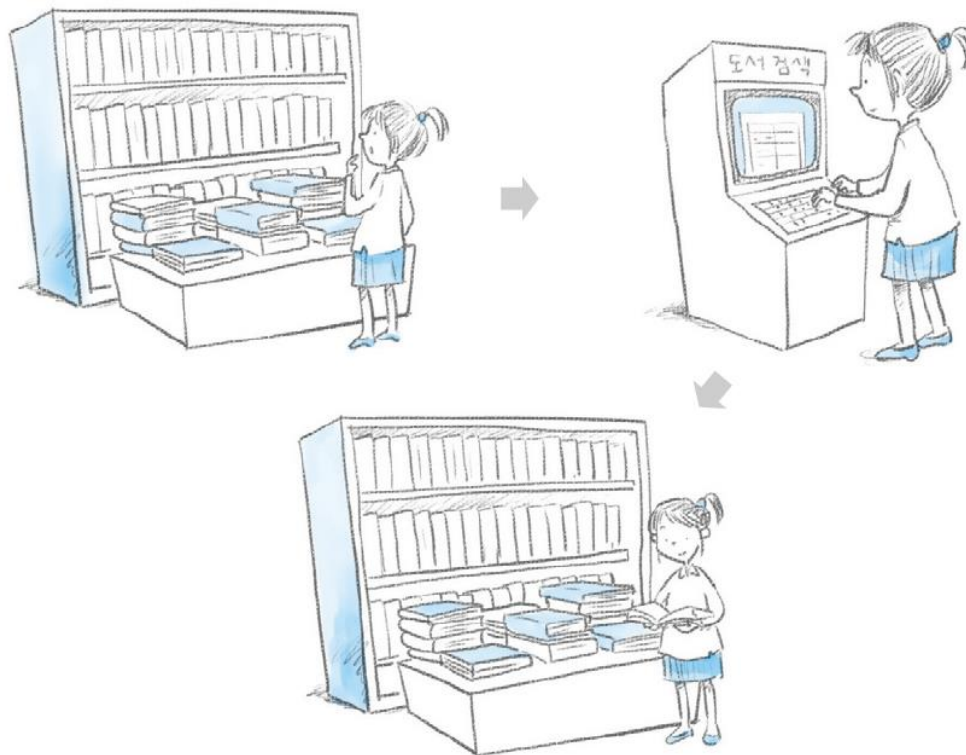


□ 데이터

- 컴퓨터를 사용할 때 불규칙하게 만들어지는 다양하고 많은 값들

□ 정보

- 체계적이고 조직적으로 관리하고 사용되는 데이터
- 여러 가지 값들로 표현되는 데이터를 의미 있고 가치 있는 형태로 가공해 놓은 것



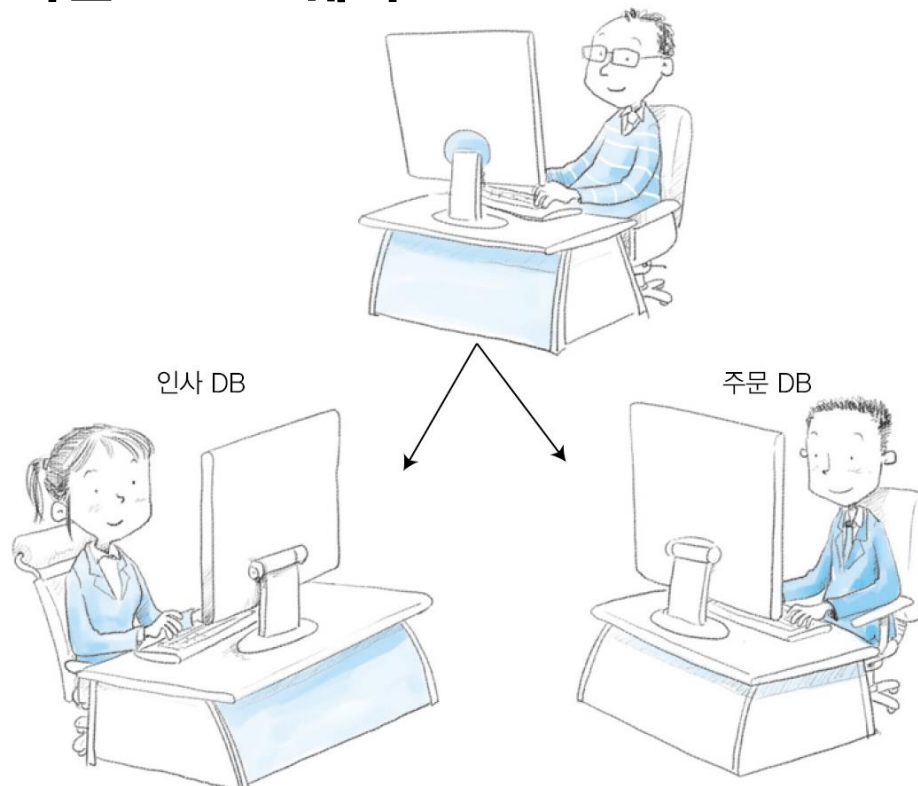
데이터베이스와 데이터베이스 관리 시스템

□ 데이터베이스

- 특정 조직 내에서 다수의 사용자들이 공유해 사용할 수 있도록 통합하고 저장한 운영 데이터의 집합체

□ 데이터베이스 관리 시스템(DBMS)

- 데이터베이스를 생성하여 안정적이고 효율적으로 운영하는 데 필요한 기능들을 제공하는 소프트웨어



데이터베이스 특징

□ 데이터의 중복성을 최소화할 수 있다.

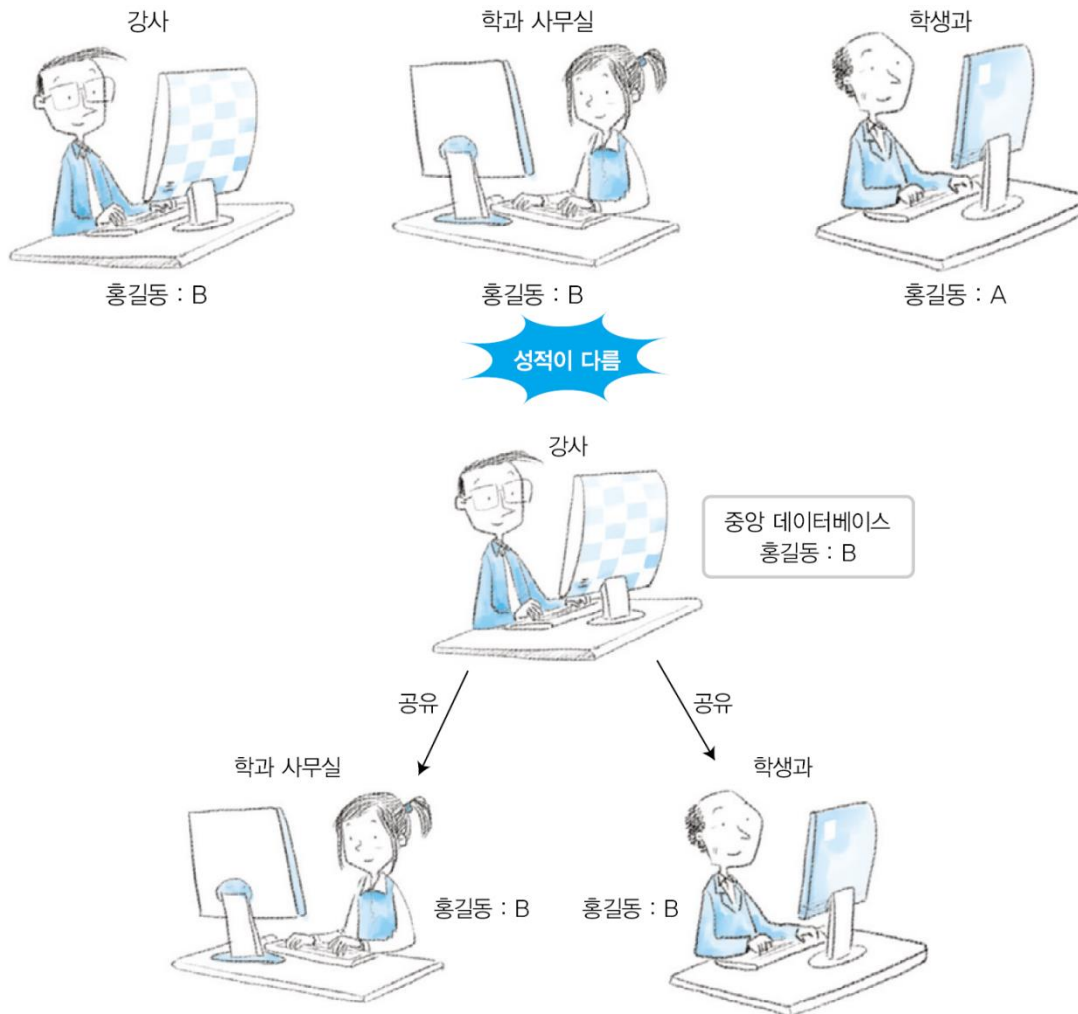
- 이용하면 업무의 흐름에 따라 데이터를 통합 및 분리하여 관리할 수 있게 되므로 이러한 중복성을 줄일 수 있음.



데이터베이스 특징

□ 데이터의 일관성을 유지할 수 있다.

- 데이터의 불일치성(Inconsistency)을 미리 방지하여 데이터를 정확하게 만들고 사용자에게 신뢰할 만한 정보를 제공할 수 있음.



데이터베이스 특징

□ 데이터의 무결성을 유지할 수 있다.

- 무결성(Integrity)이란 데이터베이스에 정확한 데이터가 유지되고 있음을 보장하는 것으로 데이터베이스에서 가장 중요한 개념임.
- 제약조건에 맞지 않는 데이터는 아예 입력되지 않도록 방지하는 기능을 제공해 데이터베이스의 무결성을 유지함.



데이터베이스 특징

□ 데이터의 독립성을 유지할 수 있다.

- 데이터의 표현 방법이나 저장 위치가 변하더라도 응용 프로그램에는 아무런 영향을 미치지 않는 것을 데이터의 독립성이라고 함.
- 데이터베이스를 사용하면 데이터의 독립성을 유지할 수 있으며, 관계형 데이터 베이스의 경우 데이터를 테이블 구조로 관리하기 때문에 독립성을 더욱 강화할 수 있음.



데이터베이스 특징

□ 데이터의 공유성을 최대화할 수 있다.

- 데이터베이스는 공동작업에 맞게 구조적으로 설계되며 통합된 체계에 의해 저장된 값들이 유지·관리되어야 하며, 조직의 구성원들은 응용 프로그램을 통해 데이터베이스에 저장된 공유 데이터들로부터 각자의 업무에 필요한 정보를 생성할 수 있음.

□ 데이터의 보안성을 보장할 수 있다.

- 데이터베이스 시스템의 성능을 평가하는 중요한 요소로, 최근 대부분의 데이터베이스 시스템은 이러한 보안 기능을 제공함.

□ 데이터를 표준화하여 관리할 수 있다.

- 데이터를 사용 목적 등의 유형별로 분류해 데이터의 형식이나 길이, 이름 등을 설정할 수 있음

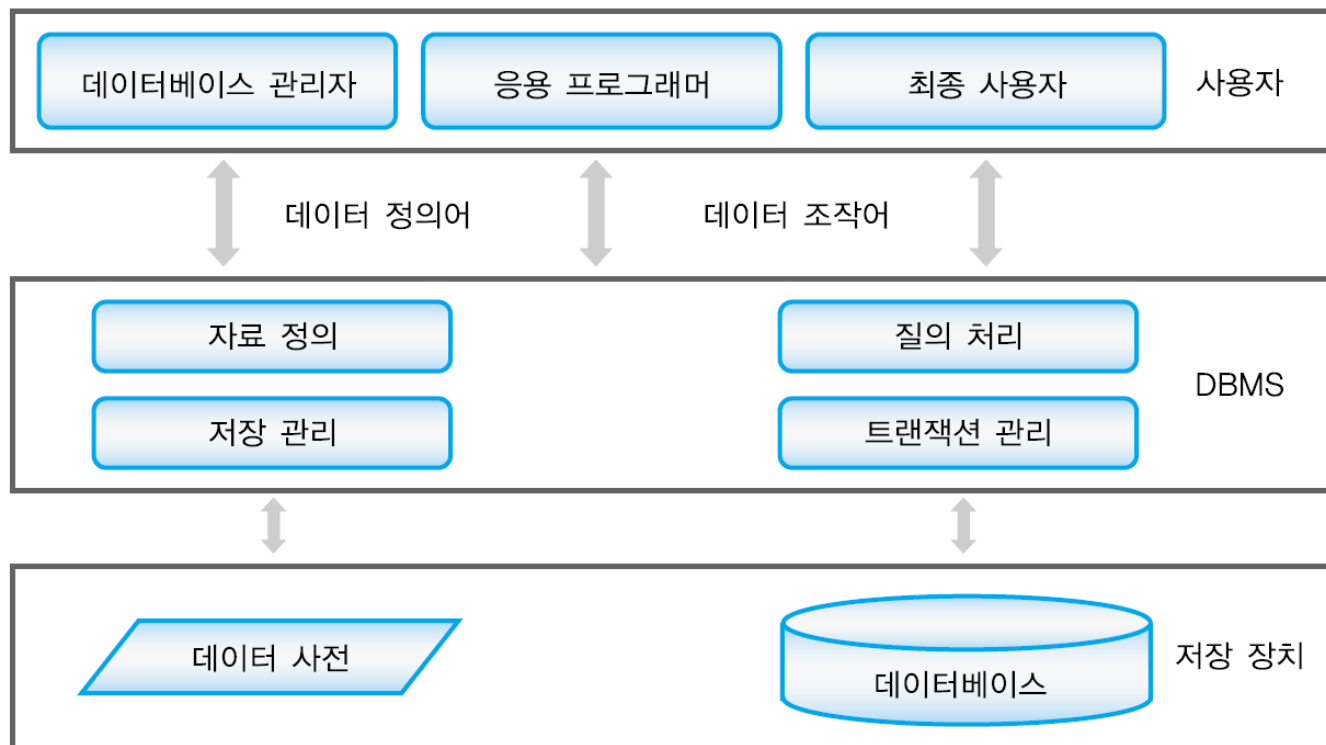


데이터베이스 구조

□ 데이터베이스 시스템(Database System)

- 하드웨어 장치에 저장되는 데이터와 사용자, 데이터베이스 관리를 위한 소프트웨어 등의 요소들을 총괄한 개념

□ 데이터베이스 시스템의 구성 요소와 관계



데이터베이스 사용자

- 데이터베이스 관리자(DBA, Database Administrator)
 - 데이터베이스 설계 · 정의, 효율적인 관리 · 운영 등 데이터베이스 시스템을 전체적으로 총괄 관리, 제어함

- 응용 프로그래머(Application Programmer)
 - 데이터베이스의 물리적인 저장 구조를 설계 및 정의
 - 최종 사용자들의 요구에 맞는 적합한 인터페이스와 응용 프로그램을 개발

- 최종 사용자(End User)
 - 데이터베이스를 실질적으로 사용하는 사람



- 데이터 정의어(DDL, Data Definition Language)
 - 데이터베이스 관리자나 응용 프로그래머가 데이터베이스의 논리적 구조를 정의하기 위해 사용하는 언어
 - 설계 모델에 따라 데이터 구조를 만들고 이미 만들어진 구조를 변경 · 삭제할 수 있음
 - 정의된 내용은 데이터 사전(Data Dictionary)에 저장
 - CREATE, DROP, ALTER, TRUNCATE 구문
- 데이터 조작어(DML, Data Manipulation Language)
 - 데이터베이스에 저장되어 있는 데이터를 처리하기 위해 사용하는 언어
 - 데이터 검색(Selection), 추가(Insert), 삭제>Delete), 갱신(Update) 작업 수행



□ DCL (Data Control Language)

- 데이터의 보안(**security**), 무결성(**integrity**), 데이터 회복(**recovery**), 병행 수행(**concurrency**) 등을 정의하는 데 사용
- 데이터베이스 관리자가 데이터 관리를 목적으로 사용함
- 데이터 제어어(**DCL**)의 종류

| 명령어 | 기능 |
|----------|--|
| COMMIT | 데이터베이스 조작 작업이 정상적으로 완료되었음을 관리자에게 알려줌 |
| ROLLBACK | 데이터베이스 조작 작업이 비정상적으로 종료되었을 때 원래의 상태로 복구함 |
| GRANT | 데이터베이스 사용자에게 사용권한을 부여함 |
| REVOKE | 데이터베이스 사용자의 사용권한을 취소함 |

데이터베이스 관리 시스템

□ 정의

- 데이터베이스를 생성해 운영하는 데 필요한 모든 기능들을 제공하는 소프트웨어

□ 기능

- 자료 정의 및 저장 관리 기능
 - 디스크 기억 장치에 데이터베이스 공간을 할당하여 물리적인 저장 구조를 만들고 이에 접근하는 방법을 제공
- 질의 처리 및 트랜잭션 기능
 - 데이터베이스에 저장되어 있는 데이터를 처리하기 위해 사용자가 입력한 요구 사항들을 번역하여 실행



데이터베이스 관리 시스템의 종류

□ 대표적인 DBMS

| DBMS | 개발사 | 특징 |
|--------|-----------|---|
| 액세스 | Microsoft | 윈도우즈 플랫폼으로 중소 규모 데이터베이스를 위한 데스크톱용 DBMS |
| SQL 서버 | Microsoft | 저렴한 제품 가격으로 Windows NT 플랫폼에서 최적의 성능을 발휘 |
| 인포믹스 | IBM | 성능이 뛰어나며 병렬 처리를 위한 멀티스레드(Multithread)를 지원 |
| DB2 | IBM | 다수 사용자가 다수 관계형 데이터베이스를 동시에 접근할 수 있는 대형 데이터베이스를 위한 시스템 |
| 오라클 | Oracle | PC급에서 메인프레임급까지 모두 설치할 수 있으며 분산 처리 지원 기능이 우수 |
| MySQL | MySQL AB | 다양한 플랫폼과 API를 지원하는 비상업용 DBMS |



데이터베이스 종류

□ 데이터베이스의 종류 및 발전 과정

1960년대

계층형 데이터베이스

네트워크형 데이터베이스

1970년대

관계형 데이터베이스

1980년대

객체 지향형 데이터베이스

1990년대

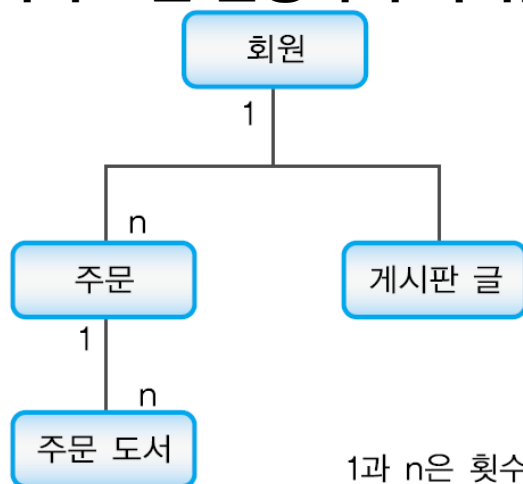
객체 관계형 데이터베이스



계층형 데이터베이스

□ 계층형 데이터베이스(HBD)의 특징

- 계층형 데이터 모델 사용
- 데이터는 트리 형태로 구성
- 각 데이터 요소(개체)들은 상하 관계를 나타내는 링크로 구성
- 현실 세계에서 하나의 조직이 여러 구성원으로 이루어지는 형태를 표현하기에 적합
- 각 개체 간의 관계를 명확하게 기술해야 함
- 개체들이 링크로 연결되어 있어 개체 간의 관계가 변경되거나 추가, 삭제되는 경우 기존의 구조를 변경하기 어려움



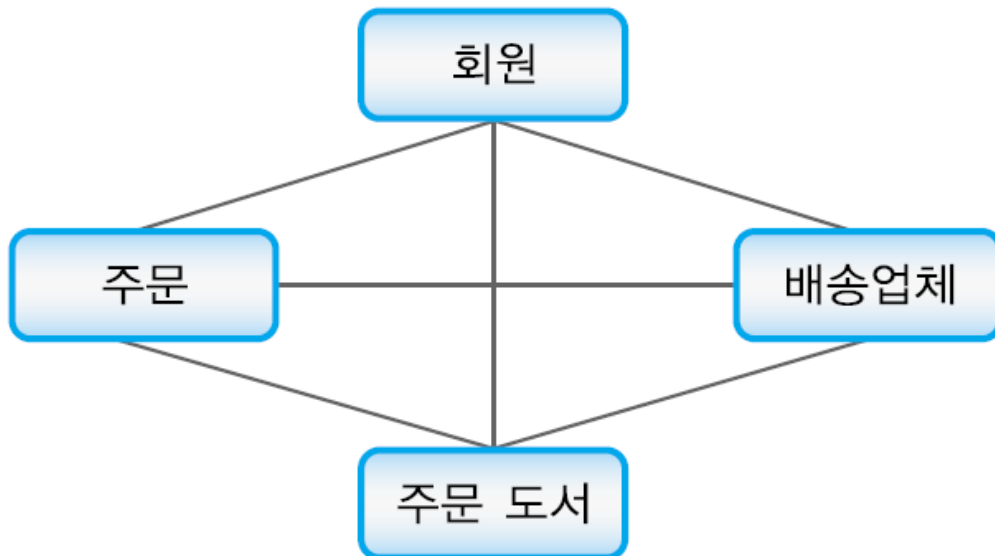
1과 n은 횟수를 의미하며 n은 반복되는 횟수를 나타낸다.



네트워크형 데이터베이스

□ 네트워크형 데이터베이스(NDB)의 특징

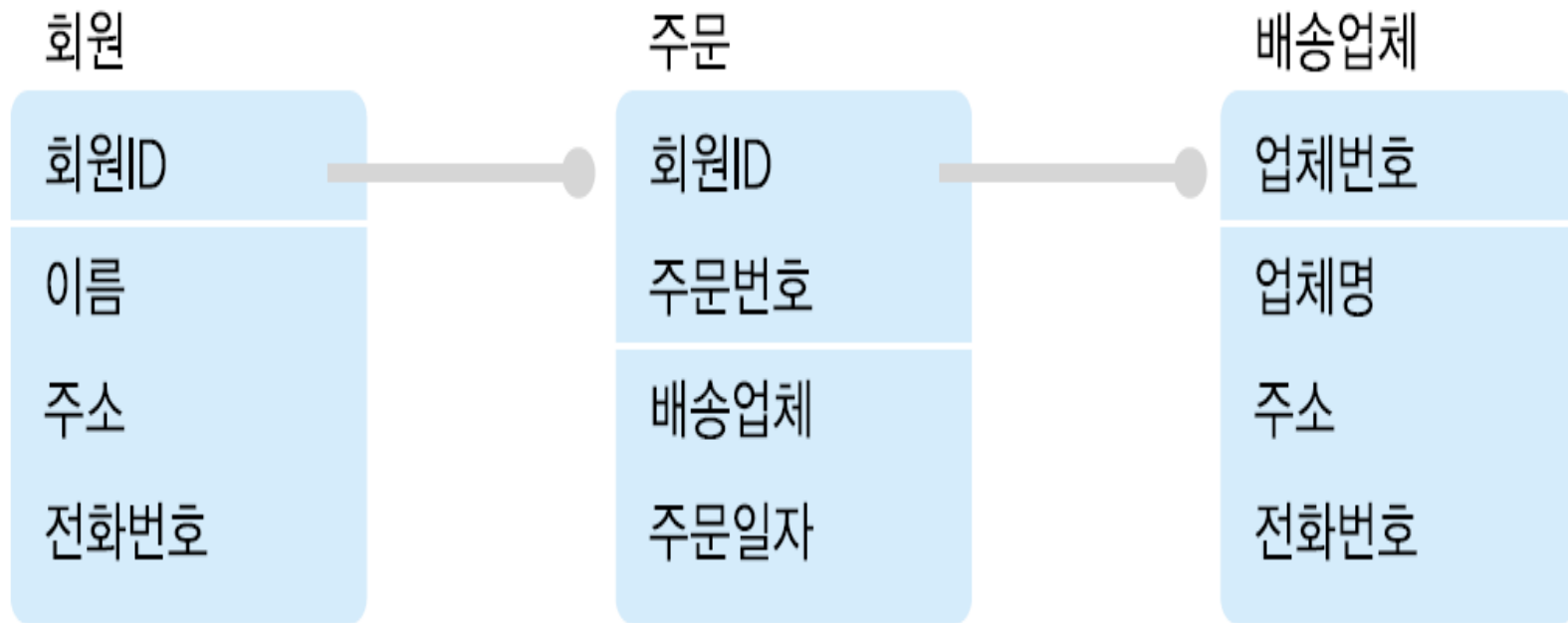
- 네트워크형 데이터 모델 사용
- 개체와 개체 관계를 그래프 구조로 연결
- 계층형 데이터 모델과 유사하나 부모(상위 개체)를 여러 개 가질 수 있다는 점이 다름
- 네트워크 구조로 진행되는 업무를 표현할 때 편리함
- 계층형 데이터베이스에 비해 구조가 복잡해지는 단점이 있음



관계형 데이터베이스

□ 관계형 데이터베이스(RDB)의 특징

- 관계형 데이터 모델 사용
- 현재까지 가장 안정적이고 효율적인 데이터베이스로 알려져 있음
- 개체를 테이블로 사용하고 개체들 간의 공통 속성을 이용해 서로 연결
- 자료의 구조가 단순한 업무에 적합



객체 지향형 데이터베이스와 객체 관계형 데이터베이스

□ 객체 지향형 데이터베이스(OODB)의 특징

- 1980년대 후반에 등장한 데이터베이스
- 객체 지향 프로그래밍 개념에 기반
- 데이터와 프로그램을 독립적인 객체의 형태로 구성
- 복잡한 데이터 유형을 처리하기 쉽고 객체들을 이해하기 쉬움
- 개념적인 형태로 실제 사용하기는 어렵기 때문에 아직까지 완전한 형태의 객체 지향형 데이터베이스를 구현하지 못함



□ 객체 관계형 데이터베이스의 특징

- 1990년대 후반에 등장
- 멀티미디어 데이터 처리가 많아짐에 따라 사용되기 시작
- 관계형 데이터베이스에 객체 지향 개념을 도입
- 비교적 널리 사용되고 있음



관계형 데이터베이스 용어

□ 테이블 형태의 데이터 구조

회원 테이블 ← 테이블(릴레이션, 엔티티)

필드(열, 속성)

| 필드(열, 속성) | | | | ← 필드명 |
|-----------|-----|---------------|----|------------------|
| 회원ID | 이름 | 주민등록번호 | 주소 | |
| eun4814 | 은현철 | 7909251645678 | 대구 | ← 레코드 (행, 튜플) |
| gdyoo | 유경동 | 5810061454321 | 충남 | |
| geunsugi | 황근의 | 7812252846512 | 경기 | |
| jungh24 | 박정희 | 7302161846621 | 서울 | |
| kkamwl | 신미원 | 7508232462035 | 서울 | |
| ksy0416 | 김수영 | 7003011564328 | 경기 | |



□테이블의 특징

- 릴레이션(Relation) 혹은 엔티티(Entity, 개체)라고도 함
- 열(Column)과 행(Row)으로 구성됨
- 테이블 구성 조건
 - 테이블에 포함된 행들은 유일해야 하고 중복된 행이 존재하지 않아야 함.
 - 테이블에 포함된 행들 간에는 순서가 존재하지 않음.
 - 테이블을 구성하는 열들 간에는 순서가 존재하지 않음.



□ 필드의 특징

- 속성(Attribute) 혹은 컬럼(Column)이라고도 함
- 테이블의 열에 해당
- 데이터 값을 기억하는 기억 단위
 - 예) 회원 테이블의 필드 : 회원ID, 이름, 주민등록번호, 주소
- 차수(degree) : 테이블에 포함된 필드의 개수
 - 예) 회원 테이블의 차수 : 4



관계형 데이터베이스 용어-레코드(Record)/도메인(Domain)

□ 레코드의 특징

- 튜플(Tuple)이라고도 함
- 테이블의 행에 해당
 - 예) 회원 테이블의 레코드 : 6개 /
- 카디널리티(Cardinality)
 - 테이블에 포함된 레코드의 개수

□ 도메인의 특징

- 필드가 가질 수 있는 값들의 집합
- 필드를 생성할 때는 데이터 형식(Data Type)을 정의하는데, 데이터 형식에 맞는 값들의 집합을 의미
 - 예) 회원 테이블에서 주민등록번호 필드의 도메인 : 13자리의 숫자
- 도메인의 역할
 - 데이터베이스에 입력되는 자료의 오류를 방지하기 위해 필요



04_릴레이션 스키마와 인스턴스

| | | | | | | |
|-----------|---------|-----|-------|----|----|------|
| 릴레이션 스키마 | 학번 | 성명 | 학과 | 주소 | 성별 | 출생년도 |
| 릴레이션 인스턴스 | 0801001 | 박명철 | 컴퓨터정보 | 서울 | 남 | 1980 |
| | 0801002 | 김한빛 | 컴퓨터정보 | 진주 | 여 | 1985 |
| | 0806002 | 하주희 | 유아교육 | 원주 | 여 | 1981 |
| | 0805120 | 박석규 | 사회복지 | 강릉 | 남 | 1989 |

[그림 2-13] 릴레이션의 스키마와 인스턴스

튜플은 삽입이나 삭제에 따라 가변적이기에 릴레이션 인스턴스도 정적이지 않고 시간의 흐름에 따라 변하는 특성이 있다.



04_릴레이션의 키

- 릴레이션에 있는 튜플은 특정 속성으로 식별할 수 있어야 함.
- 두 릴레이션 간의 관계를 정의할 때는 상징적 속성이 필요한 데, 이를 키 (key) 라 함.
- 릴레이션은 키를 이용해 중복을 피하고 독립적인 튜플과 릴레이션을 구성함.



관계형 데이터베이스 용어-키(Key)

□ 키의 특징

- 테이블 내의 레코드들을 서로 구별할 수 있는 필드의 집합
- 키 값은 유일한 값을 가져야 함
 - 예) 회원 테이블의 키 : 회원ID 필드

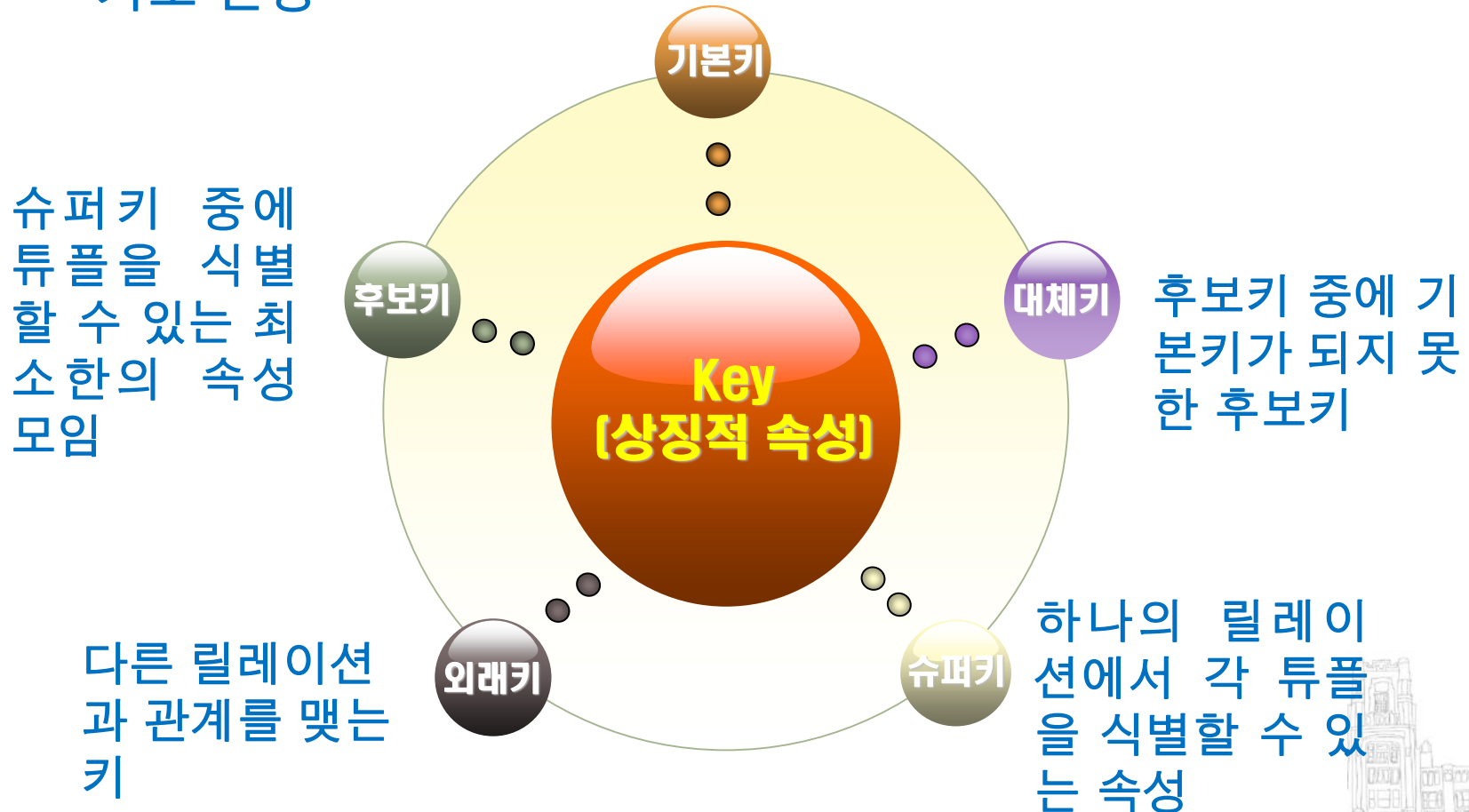
□ 키의 종류

| 이름 | 설명 |
|--------------|--------------------------------------|
| 후보 키 | 한 테이블 내에서 데이터 레코드들을 고유하게 식별할 수 있는 필드 |
| 기본 키 | 후보 키 중에서 대표로 선정된 키 |
| 수퍼 키 또는 복합 키 | 두 개 이상의 필드로 구성된 기본 키 |
| 외래 키 또는 참조 키 | 관계가 설정된 다른 테이블의 기본 키로 사용되는 필드 |



04_릴레이선의 키

후보키가 하나면 후보키가 기본키(Primary Key)가 되
지만 둘 이상의 후보키가 있을 때는 그 중 하나를 기본
키로 선정



04_키를 이용한 무결성 유지

학생

| 학번 | 성명 | 학과코드 |
|---------|-----|------|
| 0801001 | 박명철 | 05 |
| 0801002 | 김한빛 | 05 |
| 0806002 | 하주희 | 06 |
| 0805120 | 박석규 | 06 |

기본키

학과코드

| 학과코드 | 학과명 | 학과장 |
|------|-------|-----|
| 05 | 컴퓨터정보 | 박세현 |
| 06 | 유아교육 | 박재호 |
| 07 | 사회복지 | 하평수 |

기본키

외래키

[그림 2-14] 릴레이션의 기본키와 외래키

기본키와 개체 무결성 규칙

기본적으로 기본키는 튜플을 식별하는 데 사용하므로 중복될 수 없다. 이 원칙에 기본키로 지정된 속성에는 널값이 있을 수 없다는 조건을 부여하면 개체 무결성 규칙이 된다. 물론, 개체 무결성 규칙은 기본키에만 적용되고 다른 키에는 적용되지 않는다.

04_키를 이용한 무결성 유지

학생

| 학번 | 성명 | 학과코드 |
|---------|-----|------|
| 0801001 | 박명철 | 05 |
| 0801002 | 김한빛 | 05 |
| 0806002 | 하주희 | 06 |
| 0805120 | 박석규 | 06 |

기본키

외래키

학과코드

| 학과코드 | 학과명 | 학과장 |
|------|-------|-----|
| 05 | 컴퓨터정보 | 박세현 |
| 06 | 유아교육 | 박재호 |
| 07 | 사회복지 | 하평수 |

기본키

[그림 2-14] 릴레이션의 기본키와 외래키

외래키와 참조 무결성 규칙

외래키는 다른 릴레이션을 참조하려는 목적에서 만든 키다. 참조하려고 접근하는 릴레이션은 반드시 기본키값과 비교해야 하고, 상대 릴레이션의 기본키값을 갖지 못하면 널값을 갖게 된다는 제약조건이 참조 무결성 규칙이다.

04_키를 이용한 무결성 유지

고객 릴레이션

| <u>고객아이디</u> | 고객이름 | 나이 | 등급 | 직업 | 적립금 |
|--------------|------|----|--------|-----|------|
| apple | 김현준 | 20 | gold | 학생 | 1000 |
| banana | 정소화 | 25 | vip | 간호사 | 2500 |
| carrot | 원유선 | 28 | gold | 교사 | 4500 |
| orange | 정지영 | 22 | silver | 학생 | 0 |

고객 릴레이션의
기본키

주문 릴레이션

| <u>주문번호</u> | 주문고객 | 주문제품 | 수량 | 단가 | 주문일자 |
|-------------|--------|-------|----|------|------------|
| 1001 | cherry | 진짜우동 | 10 | 2000 | 2013-01-01 |
| 1002 | carrot | 맛있는파이 | 5 | 500 | 2013-01-10 |
| 1003 | banana | 그대로만두 | 11 | 4500 | 2013-01-11 |

주문 릴레이션의
기본키

주문 릴레이션의 외래키

✓ 참조 무결성 제약조건을 위반한 릴레이션의 예



04_키를 이용한 무결성 유지

고객 릴레이션

| 고객아이디 | 고객이름 | 나이 | 등급 | 직업 | 적립금 |
|--------|------|----|--------|-----|------|
| apple | 김현준 | 20 | gold | 학생 | 1000 |
| banana | 정소화 | 25 | vip | 간호사 | 2500 |
| carrot | 원유선 | 28 | gold | 교사 | 4500 |
| orange | 정지영 | 22 | silver | 학생 | 0 |

고객 릴레이션의
기본키

주문 릴레이션

| 주문번호 | 주문고객 | 주문제품 | 수량 | 단가 | 주문일자 |
|------|--------|-------|----|------|------------|
| 1001 | apple | 진짜우동 | 10 | 2000 | 2013-01-01 |
| 1002 | carrot | 맛있는파이 | 5 | 500 | 2013-01-10 |
| 1003 | NULL | 그대로만두 | 11 | 4500 | 2013-01-11 |

주문 릴레이션의
기본키

주문 릴레이션의
외래키

✓ 외래키가 NULL 값인 릴레이션의 예



1

데이터베이스의 가장 큰 목적은 데이터의 중복을 최소화하고 논리적, 물리적 독립성을 보장하는 것이다.

2

데이터베이스를 유지·관리하는 정책이나 규약을 스키마라고 한다. 물리적 저장장치의 관점에서 내부 스키마와 사용자 공동체를 위한 범 기관적 관점에서 내부 스키마, 외부의 각 사용자를 위한 외부 스키마로 나눌 수 있다.

3

데이터베이스 언어는 데이터베이스의 다양한 스키마를 정의하는 데이터 정의어(DDL)와 사용자가 원하는 정보를 얻는 데이터 조작어(DML), 데이터베이스의 트랜잭션을 명시하고 수행 권한을 부여하는 데이터 제어어(DCL)로 구성되어 있다.

4

대표적인 데이터베이스의 논리적 모델에는 트리 형태의 계층형, 그래프 형태의 네트워크형, 테이블 형태의 관계형이 있다.

5

관계형 데이터베이스를 구성하는 릴레이션의 가장 기본적인 요소는 행 단위의 튜플과 열 단위의 속성이다.

6

릴레이션의 각 튜플을 식별할 수 있는 속성을 기본키라고 하는데, 기본키의 특징으로 최소성과 유일성을 들 수 있다. 그리고 기본키에는 어떤 경우라도 널값이 있을 수 없다(개체 무결성).

7

외래키는 다른 릴레이션과의 관계를 맺는 속성으로, 참조하는 릴레이션의 기본키 속성값이 있어야 한다. 기본키와 달리 널값이 있을 수 있다(참조 무결성).