

6주차 1차시

데이터베이스(1)

자료의 조직적 집합체_데이터베이스 시스템의 이해

1 데이터베이스의 개요

2 데이터 모델

3 관계형 데이터베이스



데이터베이스(1)

◆ 학습목표

- 파일 처리 시스템의 문제점과 그 대안인 데이터베이스에 대해 알아본다.
- 데이터 모델의 종류를 알아보고 각 데이터 모델의 데이터 표현 방법을 알아본다.
- 관계형 데이터베이스의 구조와 관계형 대수를 알아본다.

1. 데이터베이스의 개요

1) 데이터베이스의 개념

- ◆ 데이터베이스란 방대한 데이터를 효율적으로 관리하기 위해 컴퓨터에 통합·저장한 것
- ◆ 데이터베이스 관리 시스템(DBMS)이라는 프로그램을 이용하여 관리



(a) 학교에서 데이터베이스의 활용



(b) 병원에서 데이터베이스의 활용



그림 6-1 일상생활에서 데이터베이스의 활용

1. 데이터베이스의 개요

2) 파일 처리 시스템

◆ 각 응용 프로그램마다 별도의 데이터 파일을 유지



그림 6-2 파일 처리 시스템

◆ 문제점

- 데이터 종속
 - 각 응용프로그램과 데이터 파일이 서로 의존적
- 데이터 중복

1. 데이터베이스의 개요

3) 데이터베이스의 특징

◆ 데이터베이스에 저장된 데이터의 특징

- 통합된 데이터(integrated data) : 연관된 데이터끼리 통합되어 저장됨
- 저장된 데이터(stored data) : 자기디스크나 광디스크 같은 보조기억장치에 저장
- 운영 데이터(operational data) : 필요에 따라 삽입, 삭제, 변경됨
- 공용 데이터(shared data) : 여러 사용자와 응용 프로그램이 공유함

◆ 데이터베이스의 특징

- 실시간 접근성
- 지속적인 변화
- 동시 공유
- 내용에 의한 참조

1. 데이터베이스의 개요

4) 데이터베이스 관리 시스템

◆ 데이터베이스 관리 시스템

- 사용자와 응용 프로그램에 편리하고 효율적인 데이터베이스 사용 환경을 제공하는 소프트웨어

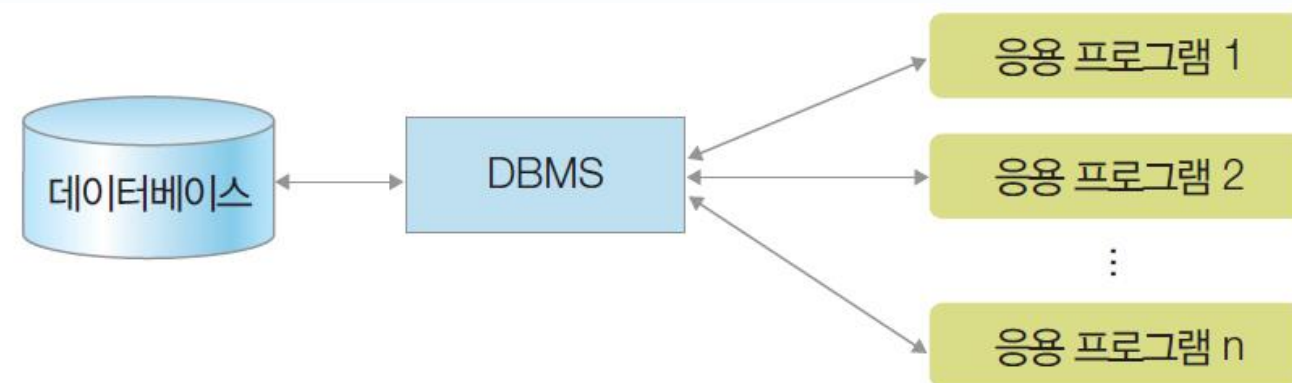


그림 6-3 DBMS를 통한 데이터베이스 관리

◆ 데이터베이스의 특징



그림 6-4 DBMS의 종류

1. 데이터베이스의 개요

4) 데이터베이스 관리 시스템

◆ 데이터베이스 관리 시스템

- ① 데이터의 중복과 불일치 감소
- ② 사용자에게 더 나은 서비스 제공
- ③ 시스템의 융통성이 향상됨
- ④ 시스템 개발 및 유지 비용 감소
- ⑤ 표준화 시행이 용이
- ⑥ 보안이 향상됨
- ⑦ 데이터 무결성이 향상됨
- ⑧ 조직체의 요구 사항을 파악하여 조정할 수 있음
- ⑨ 시스템의 고장으로부터 데이터베이스를 복구할 수 있음
- ⑩ 데이터베이스의 공유와 동시 접근이 가능함

1. 데이터베이스의 개요

5) 데이터베이스 언어

◆ 데이터 정의어

- 데이터 저장 구조, 데이터 접근 방법, 데이터 형식 등 데이터베이스를 구축하거나 수정할 때 사용하는 언어

◆ 데이터 조작어

- 데이터베이스에 저장된 데이터를 검색, 수정, 삽입, 삭제할 때 사용하는 언어

◆ 데이터 제어어

- 데이터를 보호하고 관리하는 언어
- 데이터베이스의 무결성 유지, 보안 및 접근 제어, 시스템 장애로부터의 복구, 병행 수행 제어 기능 등을 수행

1. 데이터베이스의 개요

6) 데이터베이스 사용자

◆ 응용 프로그래머

- 프로그래밍 언어(C, COBOL 등)를 사용하여 특정 응용 프로그램이나 인터페이스를 구현하는 사람

◆ 최종 사용자

- 데이터의 검색, 삽입, 삭제, 갱신 등을 위해 DBMS를 사용하는 사람

◆ 데이터베이스 관리자

- 데이터 정의어(DDL)와 데이터 제어어(DCL)를 사용하여 데이터베이스 스키마를 생성하고 관리하는 사람

2. 데이터 모델

1) 데이터 모델의 개념

- ◆ 데이터베이스 구조의 근간을 이루는 것
- ◆ 데이터베이스 설계 과정에서 데이터의 논리적인 구조를 표현하기 위해 사용하는 도구
- ◆ 대표적인 데이터 모델
 - 계층적 데이터 모델, 네트워크형 데이터 모델, 관계형 데이터 모델, 객체 지향형 데이터 모델

2. 데이터 모델

2) 계층형 데이터 모델

- ◆ 초창기 메인 프레임에서 널리 사용
- ◆ 데이터는 레코드와 링크로 구성된 트리 형태
- ◆ 링크로 연결된 레코드 집합은 부모-자식 관계를 표현
- ◆ 부모 노드와 자식 노드 간의 관계는 일 대 다 관계
- ◆ ADABAS, IMS, DMS-II 등

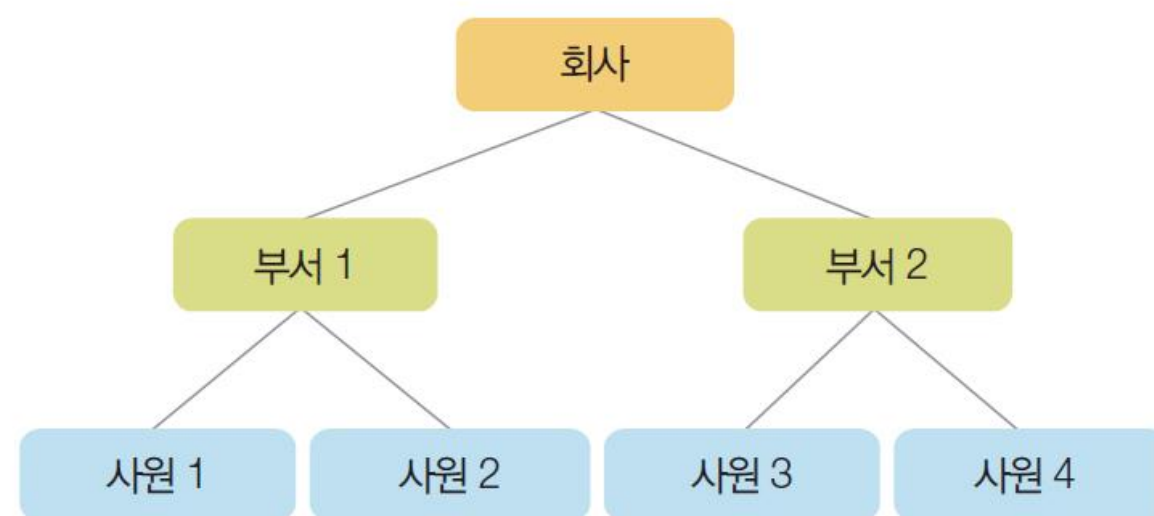


그림 6-5 계층형 데이터 모델의 예 : 회사의 조직 구조

2. 데이터 모델

3) 네트워크형 데이터 모델

- ◆ 그래프 구조에 기반
- ◆ 레코드를 노드로, 레코드와 레코드 간의 관계는 간선(edge)으로 나타냄
- ◆ 레코드와 레코드 간의 관계는 다 대 다 관계
- ◆ 레코드들이 간선으로 연결되어 구조를 변경하기 어려워 확장성이 떨어짐
- ◆ 코다실(CODASYL)

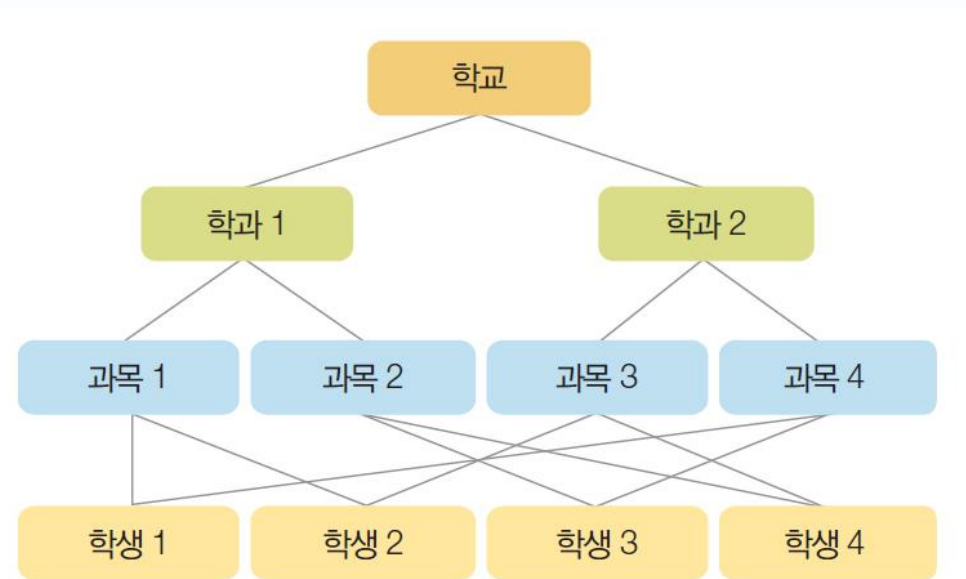


그림 6-6 네트워크형 데이터 모델의 예 : 학교의 수강 신청

2. 데이터 모델

4) 관계형 데이터 모델

- ◆ 1970년에 IBM연구소에서 일하던 에드가 F. 코드가 제안한 모델.
- ◆ 다른 모델에 비해 개념이 단순하여 현재까지도 가장 널리 사용됨
- ◆ 행과 열로 구성된 2차원 테이블에 데이터를 저장
- ◆ 테이블은 데이터베이스에서 표현하는 하나의 엔티티에 관한 정보를 저장
- ◆ DB2, Oracle, MySQL, MS SQL Server, Access 등

고객

학번	성명	주소	전화번호
C1000	홍길동	서울	02-123-4567
C1001	전우치	부산	051-234-5678

그림 6-7 관계형 데이터 모델 : 학생 정보 관리

2. 데이터 모델

5) 객체 지향형/관계형 데이터 모델

◆ 객체 지향형 데이터 모델

- 객체 지향 프로그래밍의 패러다임을 기반으로 함
- 오브젝티비티(Objectivity), 젬스톤(GemStone), O2, 온토스(Ontos), 버산트(Versant) 등

◆ 객체 관계형 데이터 모델

- 1990년대 후반에 등장
- 관계형 데이터 모델에 객체 지향 개념을 더한 모델
- 인포믹스(Informix), 유니버설 서버(Universal Server), 오라클 9i(Oracle 9i) 등

3. 관계형 데이터베이스

1) 관계형 데이터베이스의 구조

- ◆ 릴레이션(테이블)으로 구성된 데이터 항목들의 집합
- ◆ 릴레이션은 행과 열로 구성된 2차원 테이블
- ◆ 하나의 릴레이션은 하나의 엔티티에 관한 데이터를 저장
- ◆ 각 릴레이션은 고유한 이름을 가짐

속성					
학번	성명	학과명	학년	연락처	이메일
12120001	홍길동	컴퓨터공학과	2	010-1234-5678	hong@naver.com
11120010	김재철	전자공학과	3	011-2345-4321	kim@hanmail.net
13130021	박은영	정보통신공학과	3	016-987-6543	park@nate.com
14140101	홍길동	의용공학과	2	019-456-7890	gildong@yahoo.com

튜플
(레코드)

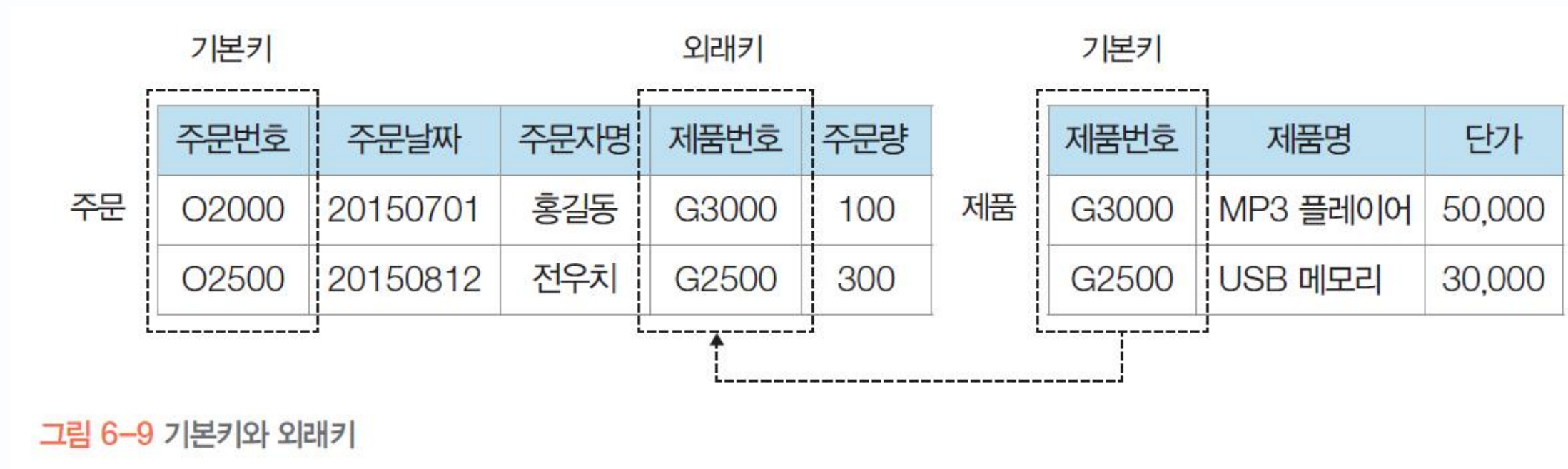
그림 6-8 Student 릴레이션

3. 관계형 데이터베이스

1) 관계형 데이터베이스의 구조

◆ 용어

- 튜플: 테이블의 행(레코드 단위)를 말함
- 속성: 열에 해당, 도메인은 속성들의 값의 범위를 의미함
- 도메인: 속성 하나가 가질 수 있는 집합
- 차수: 튜플을 구성하는 속성의 개수를 의미
- 카디널리티: 튜플의 개수를 의미
- 널 값: 속성 값이 아직 정해지지 않았거나 해당되는 값이 없을 경우에 사용
- 키: 각 튜플을 유일하게 구별 할 수 있는 하나 이상의 속성 모임을 의미
 - 후보키, 기본키, 대체키, 외래키



3. 관계형 데이터베이스

2) 무결성 제약 조건

◆ 도메인 제약 조건

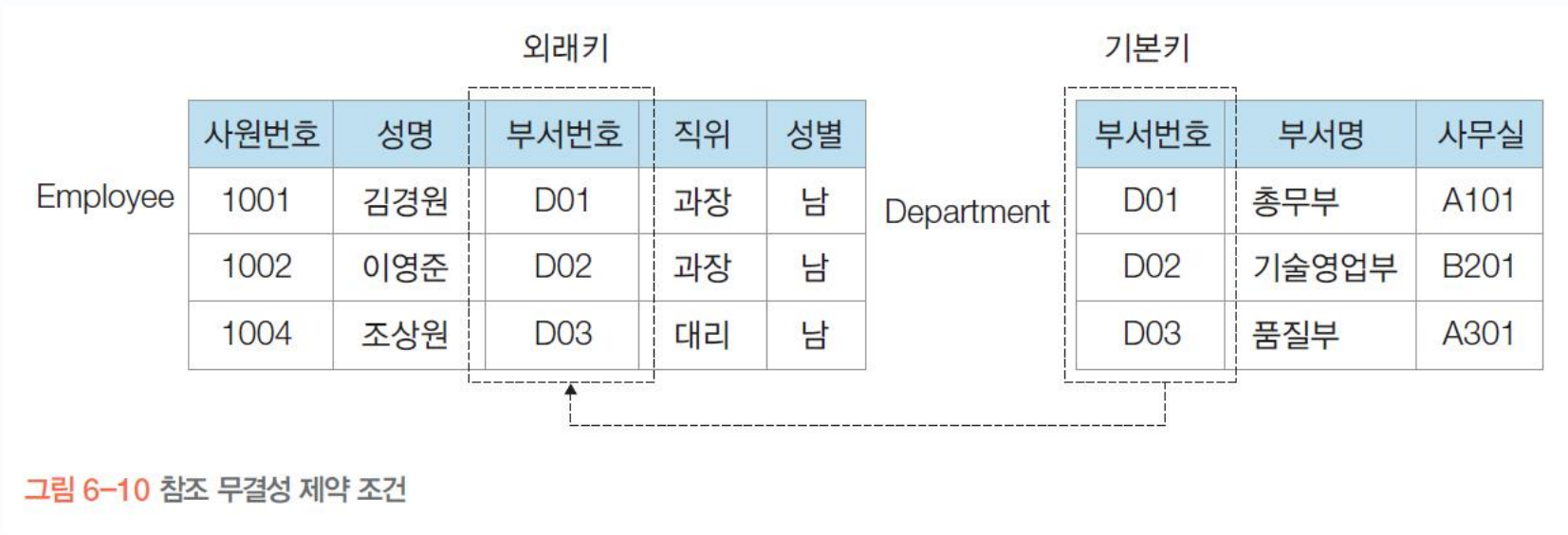
- 각 속성의 값은 반드시 도메인에 속한 하나의 값이어야 함
- 속성의 기본 값과 가능한 값들의 범위, 널 값의 허용 여부 등을 지정

◆ 개체 무결성 제약 조건

- 기본키를 구성하는 어떤 속성도 널 값을 가질 수 없음

◆ 참조 무결성 제약 조건

- 외래키의 값은 참조된 릴레이션의 기본키 값과 같아야 함



3. 관계형 데이터베이스

3) 관계형 대수

- ◆ 관계형 데이터베이스에 데이터를 삽입하고, 저장된 데이터를 검색, 수정, 삭제하는 등의 기능을 수행하는 언어
- ◆ 사용자가 원하는 데이터가 무엇(what)인지와 검색 방법(how)까지 기술하므로 절차적 언어

(예) Employee 릴레이션과 Department 릴레이션을 이용

사원번호	성명	부서번호	직위	성별
1001	김경원	D01	과장	남
1002	이영준	D02	과장	남
1003	최효정	D02	사원	여
1004	조상원	D03	대리	남
1005	박민혜	D01	계장	여

그림 6-11 Employee 릴레이션

3. 관계형 데이터베이스

3) 관계형 대수

◆ 선택 연산

- 주어진 릴레이션에서 특정 조건을 만족하는 튜플을 구하는 연산

$\sigma_{\text{조건식}}$ (릴레이션)

- Employee 릴레이션에서 직위가 '과장'인 튜플을 선택하는 연산

$\sigma_{\text{직위='과장'}}$ (Employee)

사원번호	성명	부서번호	직위	성별
1001	김경원	D01	과장	남
1002	이영준	D02	과장	남

그림 6-13 선택 연산의 결과

3. 관계형 데이터베이스

3) 관계형 대수

◆ 추출 연산

- 주어진 릴레이션에서 원하는 속성만 발췌하는 연산

$\pi_{\text{속성 리스트}}(\text{릴레이션})$

- Employee 릴레이션에서 '직위' 속성만 추출 연산

$\pi_{\text{직위}}(\text{Employee})$



그림 6-14 추출 연산의 결과

3. 관계형 데이터베이스

3) 관계형 대수

◆ 합집합 연산

- 두 개의 릴레이션 중 어느 한쪽 또는 양쪽에 모두 존재하는 튜플을 구하는 연산

$$(\pi_{\text{속성 리스트}}(\text{릴레이션})) \cup (\pi_{\text{속성 리스트}}(\text{릴레이션}))$$

- Employee 릴레이션에서 직위가 '과장'인 튜플의 부서번호와 Department 릴레이션에서 사무실 위치가 'B201'인 튜플의 부서번호를 모두 구하는 연산

$$(\pi_{\text{부서번호}}(\sigma_{\text{직위='과장'}}(\text{Employee}))) \cup (\pi_{\text{부서번호}}(\sigma_{\text{사무실='B201'}}(\text{Department})))$$

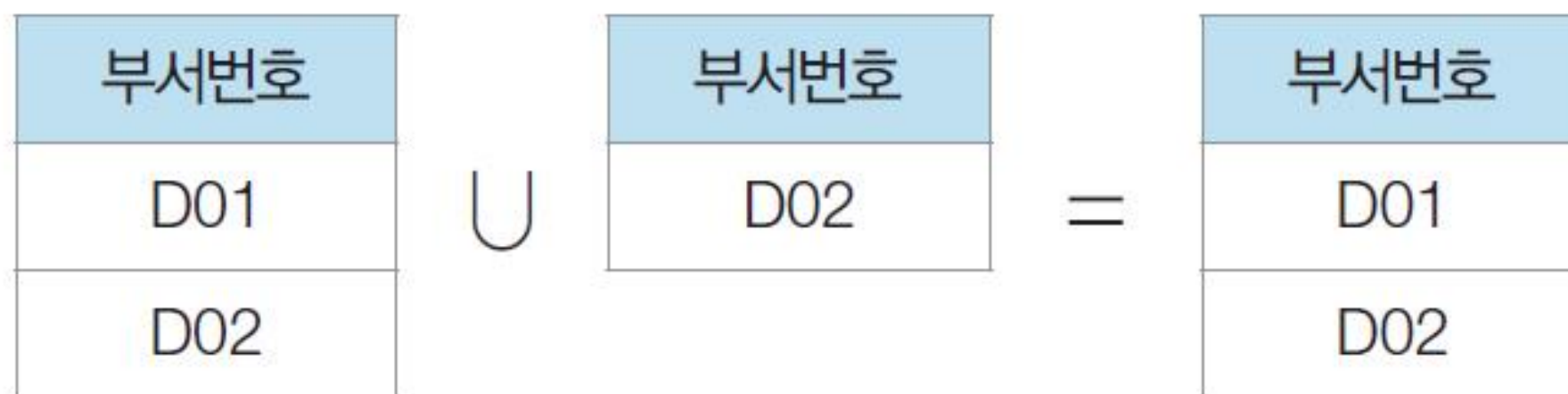


그림 6-15 합집합 연산의 결과

3. 관계형 데이터베이스

3) 관계형 대수

◆ 교집합 연산

- 두 개의 릴레이션에 모두 존재하는 튜플을 구하는 연산

$$(\pi_{\text{속성 리스트}}(\text{릴레이션})) \cap (\pi_{\text{속성 리스트}}(\text{릴레이션}))$$

- Employee 릴레이션에서 과장과 계장이 함께 근무하는 부서번호를 구하는 연산

$$(\pi_{\text{부서번호}}(\sigma_{\text{직위}='과장'}(\text{Employee}))) \cap (\pi_{\text{부서번호}}(\sigma_{\text{직위}='계장'}(\text{Employee})))$$

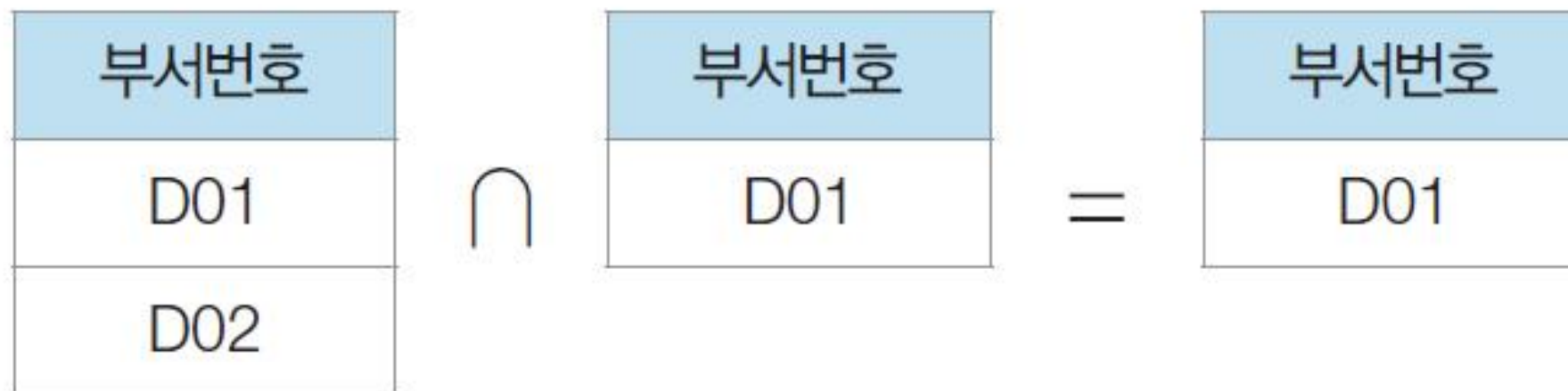


그림 6-16 교집합 연산의 결과

3. 관계형 데이터베이스

3) 관계형 대수

◆ 차집합 연산

- 릴레이션에 속하지만 다른 릴레이션에는 속하지 않는 튜플을 구하는 연산

$$(\pi_{\text{속성 리스트}}(\text{릴레이션})) - (\pi_{\text{속성 리스트}}(\text{릴레이션}))$$

- 소속된 직원이 한 명도 없는 부서의 부서번호를 구하는 연산

$$(\pi_{\text{부서번호}}(\text{Department})) - (\pi_{\text{부서번호}}(\text{Employee}))$$

부서번호		부서번호		부서번호
D01	—	D01	=	D04
D02		D02		
D03		D03		
D04				

그림 6-17 차집합 연산의 결과

3. 관계형 데이터베이스

3) 관계형 대수

◆ 카티전 곱 연산

- 두 개의 릴레이션에 존재하는 모든 튜플의 조합을 구하는 연산

릴레이션1 X 릴레이션2

- Product 릴레이션과 Store 릴레이션에 대한 카티전 곱을 구하는 연산

Product X Store

제품번호	제품명	단가
G3000	MP3 플레이어	50,000
G3001	USB 메모리	30,000

그림 6-18 Product 릴레이션

영업점 코드	영업점
S1000	하이마트
S2000	전자랜드

그림 6-19 Store 릴레이션

제품번호	제품명	단가	영업점 코드	영업점
G3000	MP3 플레이어	50,000	S1000	하이마트
G3000	MP3 플레이어	50,000	S2000	전자랜드
G3001	USB 메모리	30,000	S1000	하이마트
G3001	USB 메모리	30,000	S2000	전자랜드

그림 6-20 카티전 곱 연산의 결과

3. 관계형 데이터베이스

3) 관계형 대수

◆ 조인 연산

- 두 개의 릴레이션에서 특정 조건을 만족하는 튜플을 결합하여 하나의 튜플로 만드는 연산

릴레이션1 ▷ ◁_{조건식} 릴레이션2

- Employee 릴레이션과 Department 릴레이션을 조인하는 연산

Employee ▷ ◁_{Employee.부서번호=Department.부서번호} Department

3. 관계형 데이터베이스

3) 관계형 대수

◆ 조인 연산



그림 6-21 조인 연산의 결과

3. 관계형 데이터베이스

3) 관계형 대수

◆ 조인 연산

- 조인 연산 결과를 이용하여 총무부에서 근무하는 사원의 이름을 검색

$$\pi_{\text{성명}}(\sigma_{\text{부서명}='총무부'}(\text{Employee} \bowtie \text{Department}))$$

성명
김경원
박민혜

그림 6-22 조인 연산의 활용 결과

3. 관계형 데이터베이스

3) 관계형 대수

◆ 나누기 연산

- 두 개의 릴레이션 R(X, Y)와 S(Y)에 대한 나누기 연산
- $R(X, Y) \div S(Y)$ 의 결과는 릴레이션 S의 모든 Y값에 관련된 릴레이션 R의 X값 출력

릴레이션1 ÷ 릴레이션 2

- Product2 릴레이션을 Store 릴레이션으로 나누는 연산

$Product2 \div \pi_{\text{영업점}}(\sigma_{\text{영업점 코드}='S1000'}(Store))$





요약

◆ 데이터베이스 관리 시스템

- 사용자와 응용 프로그램에게 편리하고 효율적인 데이터베이스 사용환경을 제공하는 소프트웨어이다.

◆ 데이터베이스 언어

- 데이터베이스 시스템을 구축하고 관리하는데 사용되는 언어로, 데이터 정의어, 데이터 조작어, 데이터 제어어로 구분된다.

◆ 데이터베이스 사용자

- 데이터베이스 시스템의 활용 형태에 따라 응용 프로그래머, 최종 사용자, 데이터베이스 관리자로 구분된다.

◆ 데이터베이스 모델

- 데이터베이스 설계과정에서 데이터의 논리적인 구조를 표현하기 위해 사용되는 도구이다.