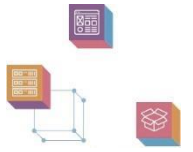


[강의교안 이용 안내]

- 본 강의교안의 저작권은 한빛아카데미(주)에 있습니다.
- 이 자료를 무단으로 전재하거나 배포할 경우 저작권법 136조에 의거하여 최고 5년 이하의 징역 또는 5천만원 이하의 벌금에 처할 수 있고 이를 병과(併科)할 수도 있습니다.



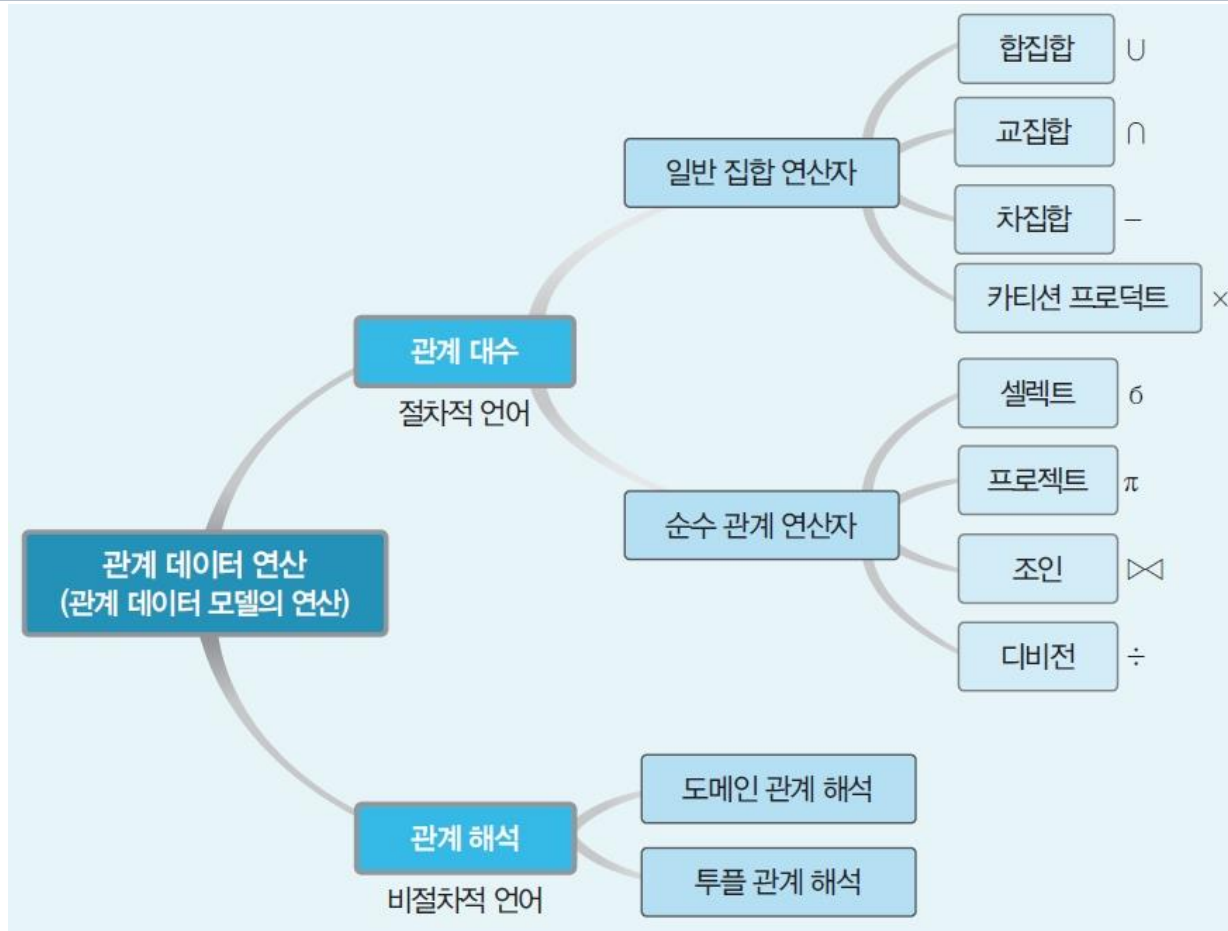
관계 데이터 연산

01 관계 데이터 연산의 개념

02 관계 대수

03 관계 해석

학습목표



- 관계 데이터 연산의 개념과 종류를 알아본다.
- 일반 집합 연산자와 순수 관계 연산자의 차이를 이해한다.
- 일반 집합 연산자와 순수 관계 연산자를 이용해 질의를 표현하는 방법을 익힌다.
- 관계 해석의 개념을 간단히 정리해본다.

01 관계 데이터 연산의 개념



◆ 데이터 모델 = 데이터 구조 + 연산 + 제약조건

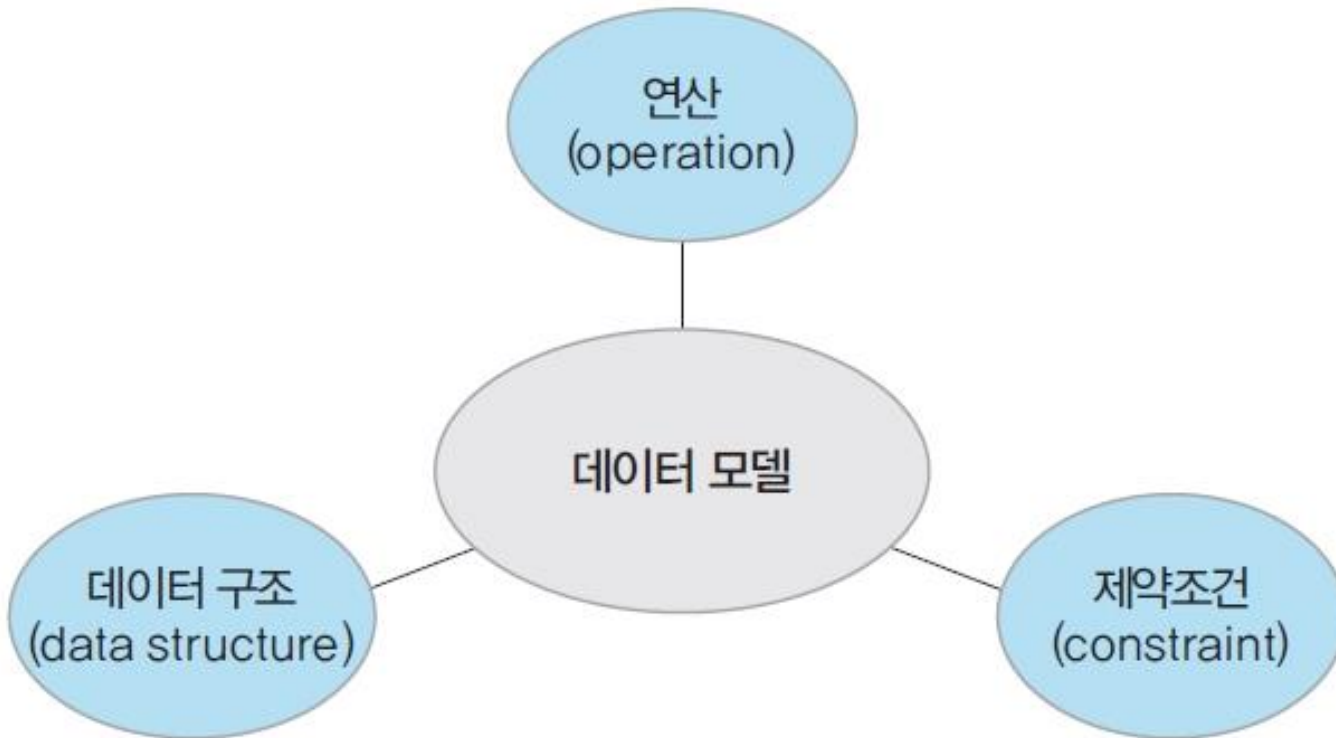


그림 6-1 데이터 모델의 구성

01 관계 데이터 연산의 개념



◆ 관계 데이터 연산(relational data operation)

- 관계 데이터 모델의 연산
- 원하는 데이터를 얻기 위해 릴레이션에 필요한 처리 요구를 수행하는 것
- 관계 대수와 관계 해석
 - 기능과 표현력 측면에서 동등한 능력



그림 6-2 관계 데이터 연산의 종류

01 관계 데이터 연산의 개념



◆ 관계 대수와 관계 해석의 역할

- 데이터 언어의 유용성을 검증하는 기준
- 관계 대수나 관계 해석으로 기술할 수 있는 모든 질의를 기술할 수 있는 데이터 언어 → 관계적으로 완전(relationally complete)하다고 판단
 - 질의(query) : 데이터에 대한 처리 요구



◆ 관계 대수(relational algebra)의 개념

- 절차 언어(procedural language)
 - 원하는 결과를 얻기 위해 릴레이션의 처리 과정을 순서대로 기술하는 언어
- 릴레이션을 처리하는 연산자들의 모임
 - 대표 연산자 8개
 - 일반 집합 연산자와 순수 관계 연산자로 분류
- 폐쇄 특성(closure property)
 - 피연산자도 릴레이션이고 연산의 결과도 릴레이션

02 관계 대수



◆ 관계 대수의 연산자

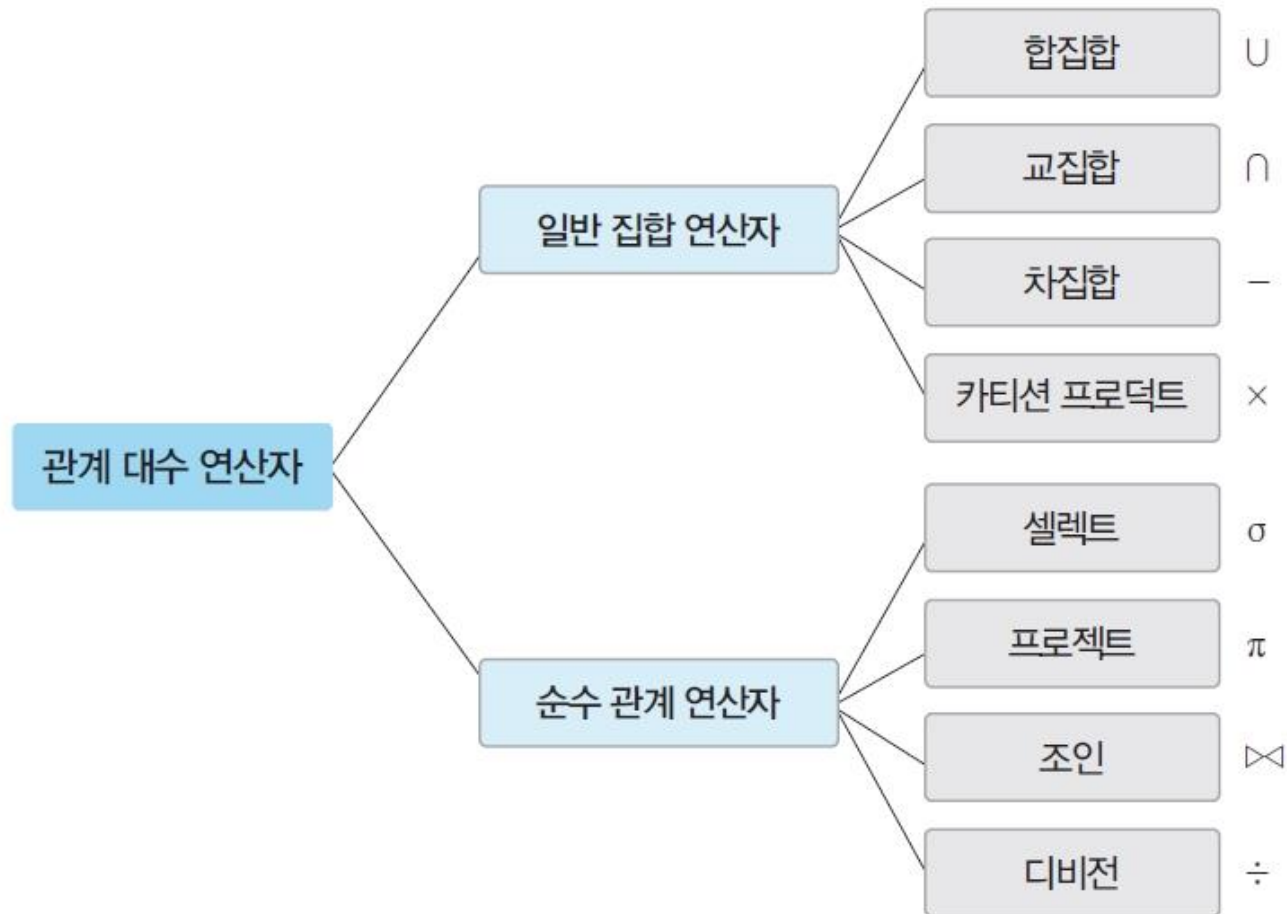


그림 6-3 관계 대수 연산자의 종류

02 관계 대수



◆ 일반 집합 연산자(set operation)

- 릴레이션이 튜플의 집합이라는 개념을 이용하는 연산자

| 연산자 | 기호 | 표현 | 의미 |
|----------|----------|--------------|---|
| 합집합 | \cup | $R \cup S$ | 릴레이션 R과 S의 합집합을 반환 |
| 교집합 | \cap | $R \cap S$ | 릴레이션 R과 S의 교집합을 반환 |
| 차집합 | $-$ | $R - S$ | 릴레이션 R과 S의 차집합을 반환 |
| 카티션 프로덕트 | \times | $R \times S$ | 릴레이션 R의 각 튜플과 릴레이션 S의 각 튜플을 모두 연결하여 만든 새로운 튜플을 반환 |

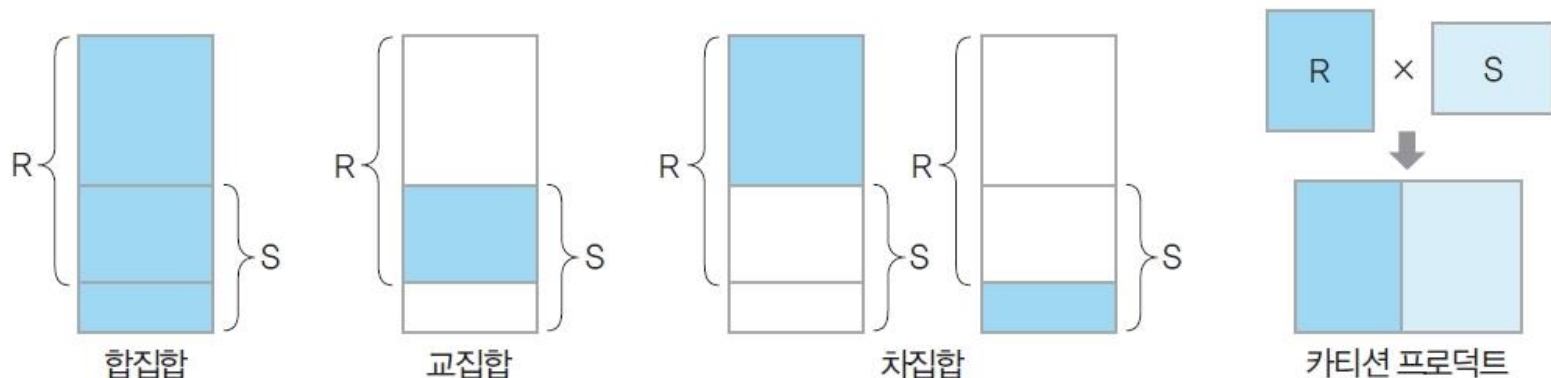


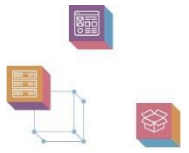
그림 6-4 일반 집합 연산자의 종류와 기능



◆ 일반 집합 연산자의 특성

- 피연산자가 2개 필요
 - 2개의 릴레이션을 대상으로 연산 수행
- 합집합, 교집합, 차집합은 피연산자인 두 릴레이션이 합병 가능해야 함
 - 합병 가능(union-compatible) 조건
 - 두 릴레이션의 차수가 같아야 함
 - 두 릴레이션에서 서로 대응되는 속성의 도메인이 같아야 함

02 관계 대수



고객 릴레이션

| 고객번호 | 고객이름 | 나이 |
|------|----------|-----|
| INT | CHAR(20) | INT |
| 100 | 정소화 | 20 |
| 200 | 김선우 | 35 |
| 300 | 고명석 | 24 |

직원 릴레이션

| 직원번호 | 직원이름 | 직위 |
|------|----------|----------|
| INT | CHAR(20) | CHAR(20) |
| 10 | 김용욱 | 부장 |
| 20 | 채광주 | 과장 |
| 30 | 김수진 | 대리 |

그림 6-6 합병이 불가능한 예

고객 릴레이션

| 고객번호 | 고객이름 | 나이 |
|------|----------|-----|
| INT | CHAR(20) | INT |
| 100 | 정소화 | 20 |
| 200 | 김선우 | 35 |
| 300 | 고명석 | 24 |

직원 릴레이션

| 직원번호 | 직원이름 | 나이 |
|------|----------|-----|
| INT | CHAR(20) | INT |
| 10 | 김용욱 | 40 |
| 20 | 채광주 | 32 |
| 30 | 김수진 | 28 |

그림 6-7 합병이 가능한 예



◆ 일반 집합 연산자 – 합집합(union)

- 합병 가능한 두 릴레이션 R과 S의 합집합 : $R \cup S$
 - 릴레이션 R에 속하거나 릴레이션 S에 속하는 모든 튜플로 결과 릴레이션 구성
- 결과 릴레이션의 특성
 - 차수는 릴레이션 R과 S의 차수와 같음
 - 카디널리티는 릴레이션 R과 S의 카디널리티를 더한 것과 같거나 적어짐
- 교환적 특징이 있음
 - $R \cup S = S \cup R$
- 결합적 특징이 있음
 - $(R \cup S) \cup T = R \cup (S \cup T)$

02 관계 대수



R

| 번호 | 이름 |
|-----|-----|
| 100 | 정소화 |
| 200 | 김선우 |
| 300 | 고명석 |

S

| 번호 | 이름 |
|-----|-----|
| 100 | 정소화 |
| 101 | 채광주 |
| 102 | 김수진 |

↓ 합집합 연산

RUS

| 번호 | 이름 |
|-----|-----|
| 100 | 정소화 |
| 200 | 김선우 |
| 300 | 고명석 |
| 101 | 채광주 |
| 102 | 김수진 |

그림 6-8 합집합 연산의 예



◆ 일반 집합 연산자 – 교집합(intersection)

- 합병 가능한 두 릴레이션 R과 S의 교집합 : $R \cap S$
 - 릴레이션 R과 S에 공통으로 속하는 튜플로 결과 릴레이션 구성
- 결과 릴레이션의 특성
 - 차수는 릴레이션 R과 S의 차수와 같음
 - 카디널리티는 릴레이션 R과 S의 어떤 카디널리티보다 크지 않음(같거나 적음)
- 교환적 특징이 있음
 - $R \cap S = S \cap R$
- 결합적 특징이 있음
 - $(R \cap S) \cap T = R \cap (S \cap T)$

02 관계 대수



R

| 번호 | 이름 |
|-----|-----|
| 100 | 정소화 |
| 200 | 김선우 |
| 300 | 고명석 |

S

| 번호 | 이름 |
|-----|-----|
| 100 | 정소화 |
| 101 | 채광주 |
| 102 | 김수진 |

↓ 교집합 연산

$R \cap S$

| 번호 | 이름 |
|-----|-----|
| 100 | 정소화 |

그림 6-9 교집합 연산의 예



◆ 일반 집합 연산자 – 차집합(difference)

- 합병 가능한 두 릴레이션 R과 S의 차집합 : $R-S$
 - 릴레이션 R에는 존재하지만 릴레이션 S에는 존재하지 않는 튜플로 결과 릴레이션 구성
- 결과 릴레이션의 특성
 - 차수는 릴레이션 R과 S의 차수와 같음
 - $R-S$ 의 카디널리티는 릴레이션 R의 카디널리티와 같거나 적음
 - $S-R$ 의 카디널리티는 릴레이션 S의 카디널리티와 같거나 적음
- 교환적, 결합적 특징이 없음

02 관계 대수



R

| 번호 | 이름 |
|-----|-----|
| 100 | 정소화 |
| 200 | 김선우 |
| 300 | 고명석 |

S

| 번호 | 이름 |
|-----|-----|
| 100 | 정소화 |
| 101 | 채광주 |
| 102 | 김수진 |

차집합 연산

R-S

| 번호 | 이름 |
|-----|-----|
| 200 | 김선우 |
| 300 | 고명석 |

S-R

| 번호 | 이름 |
|-----|-----|
| 101 | 채광주 |
| 102 | 김수진 |

그림 6-10 차집합 연산의 예



◆ 일반 집합 연산자 – 카티션 프로덕트(cartesian product)

- 두 릴레이션 R과 S의 카티션 프로덕트 : $R \times S$
 - 릴레이션 R에 속한 각 튜플과 릴레이션 S에 속한 각 튜플을 모두 연결하여 만들어진 새로운 튜플로 결과 릴레이션을 구성
- 결과 릴레이션의 특성
 - 차수는 릴레이션 R과 S의 차수를 더한 것과 같음
 - 카디널리티는 릴레이션 R과 S의 카디널리티를 곱한 것과 같음
- 교환적 특징이 있음
 - $R \times S = S \times R$
- 결합적 특징이 있음
 - $(R \times S) \times T = R \times (S \times T)$

02 관계 대수

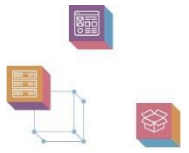
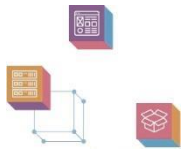


그림 6-11 카티션 프로덕트 연산의 예

02 관계 대수



◆ 순수 관계 연산자(relational operation)

- 릴레이션의 구조와 특성을 이용하는 연산자

| 연산자 | 기호 | 표현 | 의미 |
|------|-----------|------------------|--|
| 선택 | σ | $\sigma_{조건}(R)$ | 릴레이션 R에서 조건을 만족하는 튜플들을 반환 |
| 프로젝트 | π | $\pi_{속성리스트}(R)$ | 릴레이션 R에서 주어진 속성들의 값으로만 구성된 튜플들을 반환 |
| 조인 | \bowtie | $R \bowtie S$ | 공통 속성을 이용해 릴레이션 R과 S의 튜플들을 연결하여 만든 새로운 튜플들을 반환 |
| 디비전 | \div | $R \div S$ | 릴레이션 S의 모든 튜플과 관련이 있는 릴레이션 R의 튜플들을 반환 |

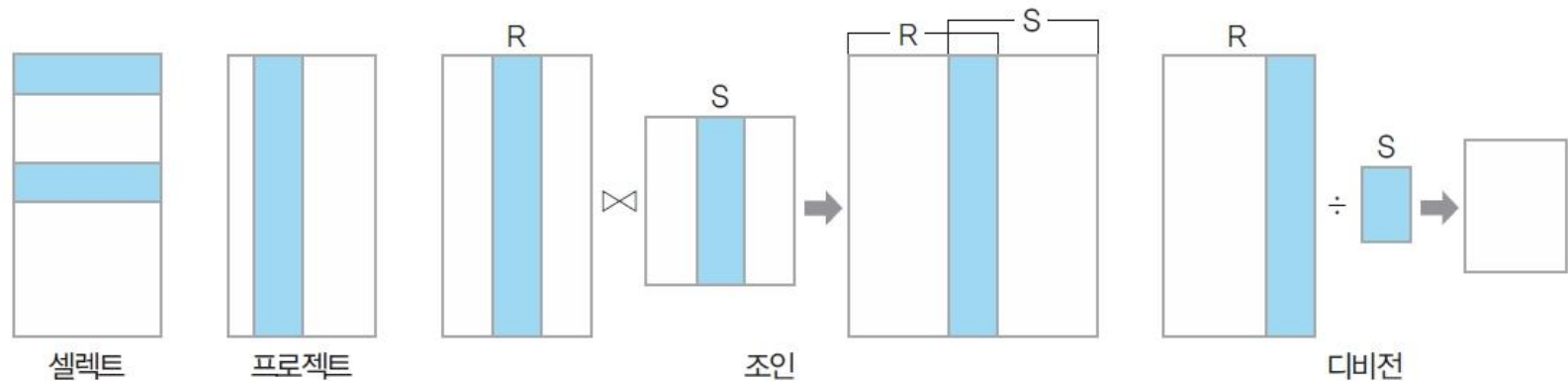


그림 6-5 순수 관계 연산자의 종류와 기능



◆ 순수 관계 연산자 – 선택(select)

- 릴레이션에서 조건을 만족하는 튜플만 선택하여 결과 릴레이션을 구성
 - 수평적 연산자 : 결과 릴레이션은 연산 대상 릴레이션의 수평적 부분집합
- 하나의 릴레이션을 대상으로 연산을 수행
- 수학적 표현 : $\sigma_{\text{조건식}}(\text{릴레이션})$
- 데이터 언어적 표현 : 릴레이션 where 조건식
- 조건식
 - 비교식, 프레디킷(predicate)이라고도 함
 - 속성과 상수의 비교나 속성들 간의 비교로 표현
 - 비교 연산자(>, ≥, <, ≤, =, ≠)와 논리 연산자(∧, ∨, ¬)를 이용해 작성

02 관계 대수



◆ 순수 관계 연산자 – 선택(select)

| 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 |

그림 6-12 선택 연산을 적용할 릴레이션 예 : 고객 릴레이션

02 관계 대수



◆ 순수 관계 연산자 – 선택(select)

예제 6-1

고객 릴레이션에서 등급이 gold인 튜플을 검색하시오.

▶▶ $\sigma_{\text{등급}='gold'}(\text{고객})$ 또는 고객 where 등급 = 'gold'

결과 릴레이션

| 고객아이디 | 고객이름 | 나이 | 등급 | 직업 | 적립금 |
|--------|------|----|------|----|------|
| apple | 김현준 | 20 | gold | 학생 | 1000 |
| carrot | 원유선 | 28 | gold | 교사 | 4500 |

02 관계 대수



◆ 순수 관계 연산자 – 선택(select)

예제 6-2

고객 릴레이션에서 등급이 gold이고, 적립금이 2000 이상인 튜플을 검색하시오.

▶▶ $\sigma_{\text{등급}='gold' \wedge \text{적립금} \geq 2000}(\text{고객})$ 또는 고객 where 등급 = 'gold' and 적립금 ≥ 2000

결과 릴레이션

| 고객아이디 | 고객이름 | 나이 | 등급 | 직업 | 적립금 |
|--------|------|----|------|----|------|
| carrot | 원유선 | 28 | gold | 교사 | 4500 |

02 관계 대수



◆ 순수 관계 연산자 – 선택(select)

| 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 |

등급 = 'gold'

선택 연산

| 고객아이디 | 고객이름 | 나이 | 등급 | 직업 | 적립금 |
|--------|------|----|------|----|------|
| apple | 김현준 | 20 | gold | 학생 | 1000 |
| carrot | 원유선 | 28 | gold | 교사 | 4500 |

그림 6-13 선택 연산의 수행 과정 : 고객 릴레이션

결과 릴레이션은 연산 대상 릴레이션의 수평적 부분집합

02 관계 대수



◆ 순수 관계 연산자 – 선택

- 교환적 특징이 있음

$$\sigma_{\text{조건식1}}(\sigma_{\text{조건식2}}(\text{릴레이션})) = \sigma_{\text{조건식2}}(\sigma_{\text{조건식1}}(\text{릴레이션})) = \sigma_{\text{조건식1} \wedge \text{조건식2}}(\text{릴레이션})$$

<고객 릴레이션에서 등급이 gold이고, 적립금이 2000 이상인 튜플 검색>

$$\sigma_{\text{적립금} \geq 2000}(\sigma_{\text{등급}='gold'}(\text{고객})) = \sigma_{\text{등급}='gold'}(\sigma_{\text{적립금} \geq 2000}(\text{고객})) = \sigma_{\text{등급}='gold' \wedge \text{적립금} \geq 2000}(\text{고객})$$



◆ 순수 관계 연산자 – 프로젝트(project)

- 릴레이션에서 선택한 속성의 값으로 결과 릴레이션을 구성
 - 수직적 연산자 : 결과 릴레이션은 연산 대상 릴레이션의 수직적 부분집합
- 하나의 릴레이션을 대상으로 연산을 수행
- 수학적 표현 : $\pi_{\text{속성리스트}}(\text{릴레이션})$
- 데이터 언어적 표현 : 릴레이션[속성리스트]

02 관계 대수

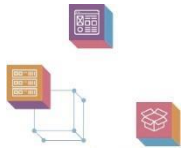


◆ 순수 관계 연산자 – 프로젝트

| 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 |

그림 6-14 프로젝트 연산을 적용할 릴레이션 예 : 고객 릴레이션

02 관계 대수



◆ 순수 관계 연산자 – 프로젝트

예제 6-3

고객 릴레이션에서 고객이름, 등급, 적립금을 검색하시오.

▶▶ $\pi_{\text{고객이름, 등급, 적립금}}(\text{고객})$ 또는 $\text{고객}[\text{고객이름, 등급, 적립금}]$

결과 릴레이션

| 고객이름 | 등급 | 적립금 |
|------|--------|------|
| 정소화 | gold | 1000 |
| 김선우 | vip | 2500 |
| 고명석 | gold | 4500 |
| 김용욱 | silver | 0 |

02 관계 대수



◆ 순수 관계 연산자 – 프로젝트

예제 6-4

고객 릴레이션에서 등급을 검색하시오.

▶▶ $\pi_{\text{등급}}(\text{고객})$ 또는 고객[등급]

결과 릴레이션

| 등급 |
|--------|
| gold |
| vip |
| silver |

결과 릴레이션에서 동일한 튜플은 중복되지 않고 한 번만 나타남

02 관계 대수



◆ 순수 관계 연산자 – 프로젝트

$\pi_{\text{고객이름, 등급, 적립금}}(\text{고객})$

| 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 |

프로젝트 연산

| 고객이름 | 등급 | 적립금 |
|------|--------|------|
| 김현준 | gold | 1000 |
| 정소화 | vip | 2500 |
| 원유선 | gold | 4500 |
| 정지영 | silver | 0 |

결과 릴레이션은
연산 대상 릴레이션의
수직적 부분집합

그림 6-15 프로젝트 연산의 수행 과정 예 : 고객 릴레이션



◆ 순수 관계 연산자 – 조인(join)

- 조인 속성을 이용해 두 릴레이션을 조합하여 결과 릴레이션을 구성
 - 조인 속성의 값이 같은 튜플만 연결하여 생성된 튜플을 결과 릴레이션에 포함
 - 조인 속성 : 두 릴레이션이 공통으로 가지고 있는 속성
- 표현법 : 릴레이션1 \bowtie 릴레이션2
- 동등 조인(equi-join)이라고도 함

02 관계 대수



◆ 순수 관계 연산자 – 조인

고객 릴레이션

| 고객아이디 | 고객이름 | 나이 | 등급 |
|--------|------|----|--------|
| apple | 김현준 | 20 | gold |
| banana | 정소화 | 25 | vip |
| carrot | 원유선 | 28 | gold |
| orange | 정지영 | 22 | silver |

주문 릴레이션

| 주문번호 | 고객아이디 | 주문제품 | 수량 |
|------|--------|-------|----|
| 1001 | apple | 진짜우동 | 10 |
| 1002 | carrot | 맛있는파이 | 5 |
| 1003 | banana | 그대로만두 | 11 |

주문 릴레이션의 외래키

만약, 정소화 고객이 주문한
제품을 검색하고 싶다면
조인 연산이 필요!

[조인 속성]

고객 릴레이션의 고객아이디,
주문 릴레이션의 고객아이디

그림 6-16 조인 연산을 적용할 예제 릴레이션들의 관계

02 관계 대수



◆ 순수 관계 연산자 – 조인

고객 ⋈ 주문

고객 릴레이션

| 고객아이디 | 고객이름 | 나이 | 등급 |
|--------|------|----|--------|
| apple | 김현준 | 20 | gold |
| banana | 정소화 | 25 | vip |
| carrot | 원유선 | 28 | gold |
| orange | 정지영 | 22 | silver |

주문 릴레이션

| 주문번호 | 고객아이디 | 주문제품 | 수량 |
|------|--------|-------|----|
| 1001 | apple | 진짜우동 | 10 |
| 1002 | carrot | 맛있는파이 | 5 |
| 1003 | banana | 그대로만두 | 11 |

어느 릴레이션의 소속인지
구분하기 위해
'릴레이션.속성이름'
형식으로 표기

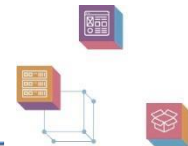
조인 연산

고객 ⋈ 주문

| 고객.고객아이디 | 고객이름 | 나이 | 등급 | 주문번호 | 주문.고객아이디 | 주문제품 | 수량 |
|----------|------|----|------|------|----------|-------|----|
| apple | 김현준 | 20 | gold | 1001 | apple | 진짜우동 | 10 |
| banana | 정소화 | 25 | vip | 1003 | banana | 그대로만두 | 11 |
| carrot | 원유선 | 28 | gold | 1002 | carrot | 맛있는파이 | 5 |

그림 6-17 조인 연산의 수행 과정 예 : 고객과 주문 릴레이션

02 관계 대수



◆ 세타 조인(theta-join, θ -join)

- 동등 조인에 비해 더 일반화된 조인
- 주어진 조인 조건을 만족하는 두 릴레이션의 모든 튜플을 연결하여 생성된 새로운 튜플로 결과 릴레이션을 구성
- 결과 릴레이션의 차수는 두 릴레이션의 차수를 더한 것과 같음
- 표현법 : 릴레이션1 $\bowtie_{A\theta B}$ 릴레이션2
 - θ 는 비교 연산자(>, \geq , <, \leq , =, \neq)를 의미

◆ 동등 조인(equi-join)

- θ 연산자가 “=”인 세타 조인을 의미
- 표현법 : 릴레이션1 $\bowtie_{A=B}$ 릴레이션2

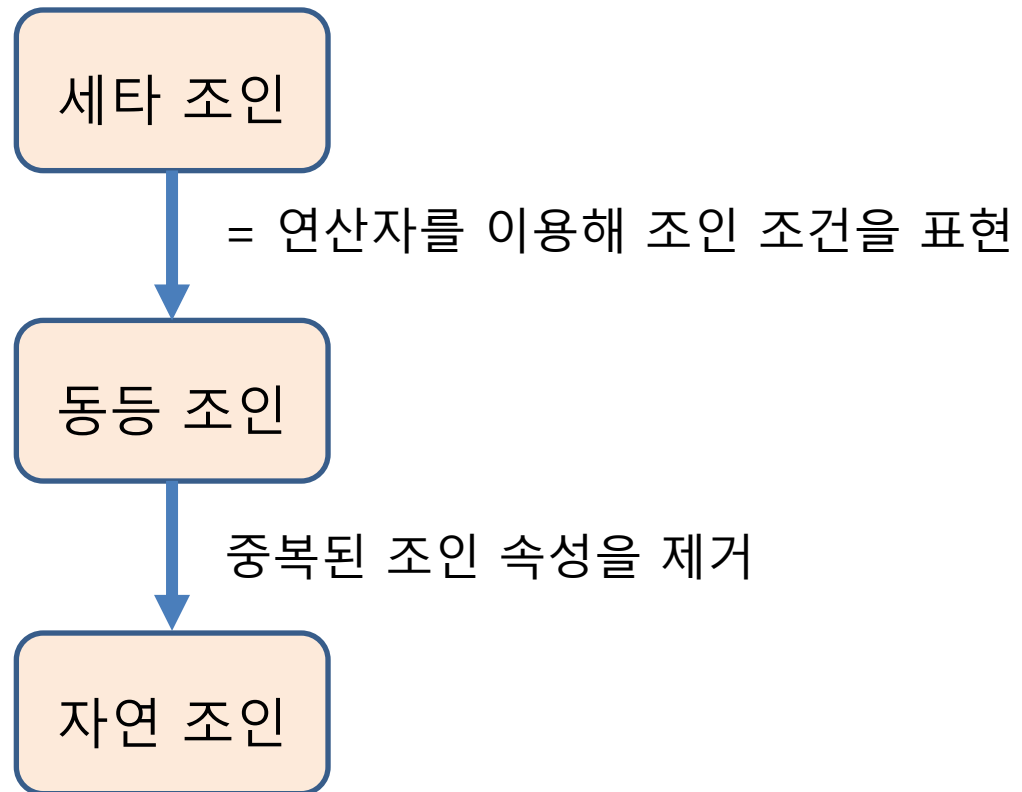
예) 고객 $\bowtie_{\text{고객.고객아이디}=\text{주문.고객아이디}}$ 주문

02 관계 대수

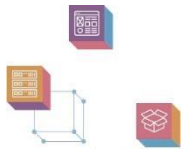


◆ 자연 조인(natural join)

- 동등 조인의 결과 릴레이션에서 조인 속성이 한 번만 나타나게 하는 연산
- 표현법 : 릴레이션1 \bowtie_N 릴레이션2



02 관계 대수



◆ 순수 관계 연산자 – 조인

고객 릴레이션

| 고객아이디 | 고객이름 | 나이 | 등급 |
|--------|------|----|--------|
| apple | 김현준 | 20 | gold |
| banana | 정소화 | 25 | vip |
| carrot | 원유선 | 28 | gold |
| orange | 정지영 | 22 | silver |

주문 릴레이션

| 주문번호 | 고객아이디 | 주문제품 | 수량 |
|------|--------|-------|----|
| 1001 | apple | 진짜우동 | 10 |
| 1002 | carrot | 맛있는파이 | 5 |
| 1003 | banana | 그대로만두 | 11 |

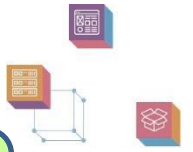
↓ 자연 조인 연산

고객 \bowtie_N 주문

| 고객아이디 | 고객이름 | 나이 | 등급 | 주문번호 | 주문제품 | 수량 |
|--------|------|----|------|------|-------|----|
| apple | 김현준 | 20 | gold | 1001 | 진짜우동 | 10 |
| banana | 정소화 | 25 | vip | 1003 | 그대로만두 | 11 |
| carrot | 원유선 | 28 | gold | 1002 | 맛있는파이 | 5 |

그림 6-18 자연 조인 연산의 예 : 고객과 주문 릴레이션

02 관계 대수



◆ 순수 관계 연산자 – 조인

조인 속성이 B1과 B2로 구성됨

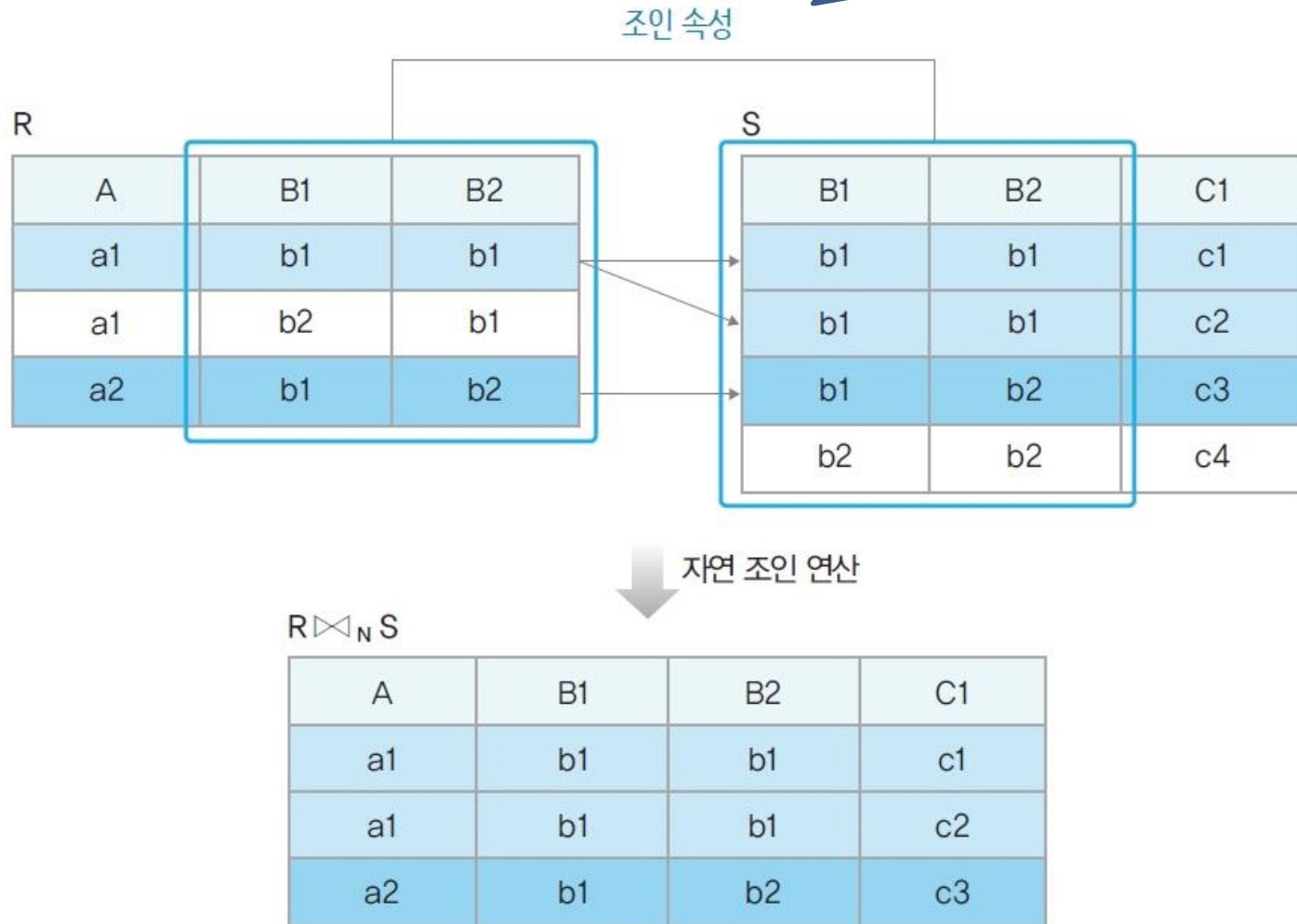


그림 6-19 2개의 속성으로 이루어진 조인 속성을 이용하는 자연 조인 연산의 예 : R과 S 릴레이션

02 관계 대수



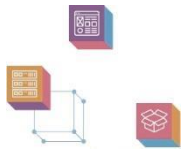
◆ 순수 관계 연산자 – 디비전(division)

- 표현법 : 릴레이션1 \div 릴레이션2
- 릴레이션2의 모든 튜플과 관련이 있는 릴레이션1의 튜플로 결과 릴레이션을 구성
 - 단, 릴레이션1이 릴레이션2의 모든 속성을 포함하고 있어야 연산이 가능함
 - 도메인이 같아야 한다는 의미

정수의 나눗셈과 유사

$$12 \div 2 = (6 \times 2) \div 2 = 6$$

02 관계 대수



◆ 순수 관계 연산자 – 디비전(division)

고객 릴레이션

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

우수등급 릴레이션

| 등급 |
|------|
| gold |

↓ 디비전 연산

고객 ÷ 우수등급

| 고객아이디 | 고객이름 | 나이 | 직업 | 적립금 |
|--------|------|----|----|------|
| apple | 김현준 | 20 | 학생 | 1000 |
| carrot | 원유선 | 28 | 교사 | 4500 |

그림 6-20 디비전 연산의 예 1 : 고객과 우수등급 릴레이션

02 관계 대수



◆ 순수 관계 연산자 – 디비전(division)

주문내역 릴레이션

| 주문고객 | 제품이름 | 제조업체 |
|--------|-------|------|
| apple | 진짜우동 | 한빛식품 |
| carrot | 맛있는파이 | 마포과자 |
| banana | 그대로만두 | 한빛식품 |
| apple | 그대로만두 | 한빛식품 |
| carrot | 그대로만두 | 한빛식품 |

제품1 릴레이션

| 제품이름 |
|-------|
| 진짜우동 |
| 그대로만두 |

제품2 릴레이션

| 제품이름 | 제조업체 |
|-------|------|
| 그대로만두 | 한빛식품 |

↓ 디비전 연산

주문내역 ÷ 제품1

| 주문고객 | 제조업체 |
|-------|------|
| apple | 한빛식품 |

주문내역 ÷ 제품2

| 주문고객 |
|--------|
| banana |
| apple |
| carrot |

그림 6-21 디비전 연산의 예 2 : 주문내역, 제품1, 제품2 릴레이션

02 관계 대수



◆ 관계 대수를 이용한 질의 표현 예



그림 6-22 질의 표현에 사용할 예제 릴레이션들 : 고객과 주문 릴레이션

02 관계 대수



◆ 관계 대수를 이용한 질의 표현 예

예제 6-5

등급이 gold인 고객의 이름과 나이를 검색하시오.

▶▶ $\pi_{\text{고객이름, 나이}}(\sigma_{\text{등급='gold'}}(\text{고객}))$

결과 릴레이션

| 고객이름 | 나이 |
|------|----|
| 김현준 | 20 |
| 원유선 | 28 |

02 관계 대수



◆ 관계 대수를 이용한 질의 표현 예

예제 6-6

고객이름이 원유선인 고객의 등급과, 원유선 고객이 주문한 주문제품, 수량을 검색하시오.

▶▶ $\pi_{\text{등급, 주문제품, 수량}}(\sigma_{\text{고객이름='원유선'}}(\text{고객} \bowtie \text{주문}))$

결과 릴레이션

| 등급 | 주문제품 | 수량 |
|------|-------|----|
| gold | 맛있는파이 | 5 |

02 관계 대수



◆ 관계 대수를 이용한 질의 표현 예

예제 6-7

주문수량이 10개 미만인 주문 내역을 제외하고 검색하시오.

▶▶ 주문 - ($\sigma_{\text{수량} < 10}(\text{주문})$)

결과 릴레이션

| 주문번호 | 주문고객 | 주문제품 | 수량 |
|------|--------|-------|----|
| 1001 | apple | 진짜우동 | 10 |
| 1003 | banana | 그대로만두 | 11 |

02 관계 대수



◆ 확장된 관계 대수 연산자

- 기본 연산자를 확장한 연산자들이 제안됨
- 예) 자연 조인 연산을 확장한 세미 조인과 외부 조인



그림 6-23 세미 조인과 외부 조인 연산을 적용할 릴레이션 예 : 고객과 주문 릴레이션

02 관계 대수



◆ 확장된 관계 대수 연산자

고객 릴레이션

| 고객아이디 | 고객이름 | 나이 |
|--------|------|----|
| apple | 김현준 | 20 |
| banana | 정소화 | 25 |
| carrot | 원유선 | 28 |
| orange | 정지영 | 22 |

주문 릴레이션

| 주문번호 | 고객아이디 | 주문제품 |
|------|--------|-------|
| 1001 | apple | 진짜우동 |
| 1002 | carrot | 맛있는파이 |
| 1003 | banana | 그대로만두 |

자연 조인 연산

고객 \bowtie_N 주문

| 고객아이디 | 고객이름 | 나이 | 주문번호 | 주문제품 |
|--------|------|----|------|-------|
| apple | 김현준 | 20 | 1001 | 진짜우동 |
| banana | 정소화 | 25 | 1003 | 그대로만두 |
| carrot | 원유선 | 28 | 1002 | 맛있는파이 |

그림 6-24 고객과 주문 릴레이션의 자연 조인 연산

02 관계 대수



◆ 확장된 관계 대수 연산자 – 세미 조인(semi-join)

- 조인 속성으로 프로젝트 연산을 수행한 릴레이션을 이용하는 조인
- 표현법 : 릴레이션1 \bowtie 릴레이션2
- 릴레이션2를 조인 속성으로 프로젝트 연산한 후, 릴레이션1에 자연 조인하여 결과 릴레이션을 구성
- 불필요한 속성을 미리 제거하여 조인 연산 비용을 줄이는 장점이 있음
- 교환적 특징이 없음
 - $R \bowtie S \neq S \bowtie R$

02 관계 대수



◆ 확장된 관계 대수 연산자 – 세미 조인(semi-join)

고객 릴레이션

| 고객아이디 | 고객이름 | 나이 |
|--------|------|----|
| apple | 김현준 | 20 |
| banana | 정소화 | 25 |
| carrot | 원유선 | 28 |
| orange | 정지영 | 22 |

주문 릴레이션

| 주문번호 | 고객아이디 | 주문제품 |
|------|--------|-------|
| 1001 | apple | 진짜우동 |
| 1002 | carrot | 맛있는파이 |
| 1003 | banana | 그대로만두 |

π 고객아이디(주문)

| 고객아이디 |
|--------|
| apple |
| carrot |
| banana |

자연 조인 연산

고객 릴레이션에서
자연 조인 연산에 참여할 수
있는 튜플만 선택됨
→ 주문한 적이 있는 고객의
데이터만 검색됨

고객 \bowtie 주문

| 고객아이디 | 고객이름 | 나이 |
|--------|------|----|
| apple | 김현준 | 20 |
| banana | 정소화 | 25 |
| carrot | 원유선 | 28 |

그림 6-25 고객과 주문 릴레이션의 세미 조인 연산

02 관계 대수



◆ 확장된 관계 대수 연산자 – 외부 조인(outer-join)

- 자연 조인 연산에서 제외되는 튜플도 결과 릴레이션에 포함시키는 조인
 - 결과 릴레이션에서 속성 값이 없는 경우는 널 값으로 처리
- 분류
 - 왼쪽(left) 외부 조인 / 오른쪽(right) 외부 조인 / 완전(full) 외부 조인
 - 모든 튜플을 결과 릴레이션으로 가져오는 릴레이션이 무엇이냐에 따라 분류
 - 자연 조인 연산에서 제외되는 튜플도 포함

02 관계 대수



◆ 확장된 관계 대수 연산자 – 외부 조인(outer-join)

■ 왼쪽 외부 조인

- 표현법 : 릴레이션1 \bowtie 릴레이션2
- 왼쪽에 있는 릴레이션1에 존재하는 모든 튜플을 결과 릴레이션에 포함시킴

02 관계 대수



◆ 확장된 관계 대수 연산자 – 외부 조인(outer-join)

고객 릴레이션

| 고객아이디 | 고객이름 | 나이 |
|--------|------|----|
| apple | 김현준 | 20 |
| banana | 정소화 | 25 |
| carrot | 원유선 | 28 |
| orange | 정지영 | 22 |

주문 릴레이션

| 주문번호 | 고객아이디 | 주문제품 |
|------|--------|-------|
| 1001 | apple | 진짜우동 |
| 1002 | carrot | 맛있는파이 |
| 1003 | banana | 그대로만두 |



왼쪽 외부 조인 연산

고객 ⋈ 주문

| 고객아이디 | 고객이름 | 나이 | 주문번호 | 주문제품 |
|--------|------|----|------|-------|
| apple | 김현준 | 20 | 1001 | 진짜우동 |
| banana | 정소화 | 25 | 1003 | 그대로만두 |
| carrot | 원유선 | 28 | 1002 | 맛있는파이 |
| orange | 정지영 | 22 | NULL | NULL |

고객 릴레이션에 있는
모든 튜플이
결과 릴레이션에 포함됨

그림 6-26 고객과 주문 릴레이션의 왼쪽 외부 조인 연산

02 관계 대수



◆ 확장된 관계 대수 연산자 – 외부 조인(outer-join)

■ 오른쪽 외부 조인

- 표현법 : 릴레이션1 \bowtie 릴레이션2
- 오른쪽에 있는 릴레이션2에 존재하는 모든 튜플을 결과 릴레이션에 포함시킴

주문 릴레이션에는 조인 연산에 참여하지 않는 튜플이 없음
→ 자연 조인한 결과와 같음

고객 \bowtie 주문

| 고객아이디 | 고객이름 | 나이 | 주문번호 | 주문제품 |
|--------|------|----|------|-------|
| apple | 김현준 | 20 | 1001 | 진짜우동 |
| banana | 정소화 | 25 | 1003 | 그대로만두 |
| carrot | 원유선 | 28 | 1002 | 맛있는파이 |

02 관계 대수



◆ 확장된 관계 대수 연산자 – 외부 조인(outer-join)

■ 완전 외부 조인

- 표현법 : 릴레이션1 \bowtie 릴레이션2
- 두 릴레이션에 있는 모든 튜플을 결과 릴레이션에 포함시킴

고객 릴레이션만 조인 연산에 참여하지 않는 튜플이 있음
→ 왼쪽 외부 조인한 결과와 같음

고객 \bowtie 주문

| 고객아이디 | 고객이름 | 나이 | 주문번호 | 주문제품 |
|--------|------|----|------|-------|
| apple | 김현준 | 20 | 1001 | 진짜우동 |
| banana | 정소화 | 25 | 1003 | 그대로만두 |
| carrot | 원유선 | 28 | 1002 | 맛있는파이 |
| orange | 정지영 | 22 | NULL | NULL |

02 관계 대수



◆ 확장된 관계 대수 연산자 – 외부 조인(outer-join)

<주문한 고객아이디는 모르는 새로운 주문 내역 튜플이 주문 릴레이션에 추가된 상황>

고객 릴레이션

| 고객아이디 | 고객이름 | 나이 |
|--------|------|----|
| apple | 김현준 | 20 |
| banana | 정소화 | 25 |
| carrot | 원유선 | 28 |
| orange | 정지영 | 22 |

주문 릴레이션

| 주문번호 | 고객아이디 | 주문제품 |
|------|--------|-------|
| 1001 | apple | 진짜우동 |
| 1002 | carrot | 맛있는파이 |
| 1003 | banana | 그대로만두 |
| 1004 | NULL | 얼큰라면 |

그림 6-27 세 가지 외부 조인 결과 비교 : 고객과 주문 릴레이션

02 관계 대수



◆ 확장된 관계 대수 연산자 – 외부 조인(outer-join)

(1) 왼쪽 외부 조인 연산: 고객 \bowtie 주문

| 고객아이디 | 고객이름 | 나이 | 주문번호 | 주문제품 |
|--------|------|----|------|-------|
| apple | 김현준 | 20 | 1001 | 진짜우동 |
| banana | 정소화 | 25 | 1003 | 그대로만두 |
| carrot | 원유선 | 28 | 1002 | 맛있는파이 |
| orange | 정지영 | 22 | NULL | NULL |

주문 내역이 없는 고객에 대한 튜플도 결과 릴레이션에서 확인 가능

02 관계 대수



◆ 확장된 관계 대수 연산자 – 외부 조인(outer-join)

(2) 오른쪽 외부 조인 연산: 고객 \bowtie 주문

| 고객아이디 | 고객이름 | 나이 | 주문번호 | 주문제품 |
|--------|------|------|------|-------|
| apple | 김현준 | 20 | 1001 | 진짜우동 |
| banana | 정소화 | 25 | 1003 | 그대로만두 |
| carrot | 원유선 | 28 | 1002 | 맛있는파이 |
| NULL | NULL | NULL | 1004 | 얼큰라면 |

주문한 고객아이디를 알지 못하는 주문 내역에 대한 튜플도
결과 릴레이션에서 확인 가능

02 관계 대수



◆ 확장된 관계 대수 연산자 – 외부 조인(outer-join)

(3) 완전 외부 조인 연산: 고객 \bowtie 주문

| 고객아이디 | 고객이름 | 나이 | 주문번호 | 주문제품 |
|--------|------|------|------|-------|
| apple | 김현준 | 20 | 1001 | 진짜우동 |
| banana | 정소화 | 25 | 1003 | 그대로만두 |
| carrot | 원유선 | 28 | 1002 | 맛있는파이 |
| orange | 정지영 | 22 | NULL | NULL |
| NULL | NULL | NULL | 1004 | 얼큰라면 |

모든 고객 튜플과 모든 주문 튜플을 결과 릴레이션에서 확인 가능

03 관계 해석



◆ 관계 해석(relational calculus)

- 처리를 원하는 데이터가 무엇인지만 기술하는 언어
 - 비절차 언어(nonprocedural language)
- 수학의 프레디킷 해석(predicate calculus)에 기반을 두고 있음
- 기능과 표현력 측면에서 관계 대수와 동등한 능력을 가짐
- 분류
 - 튜플 관계 해석(tuple relational calculus)
 - 도메인 관계 해석(domain relational calculus)



Thank You
