

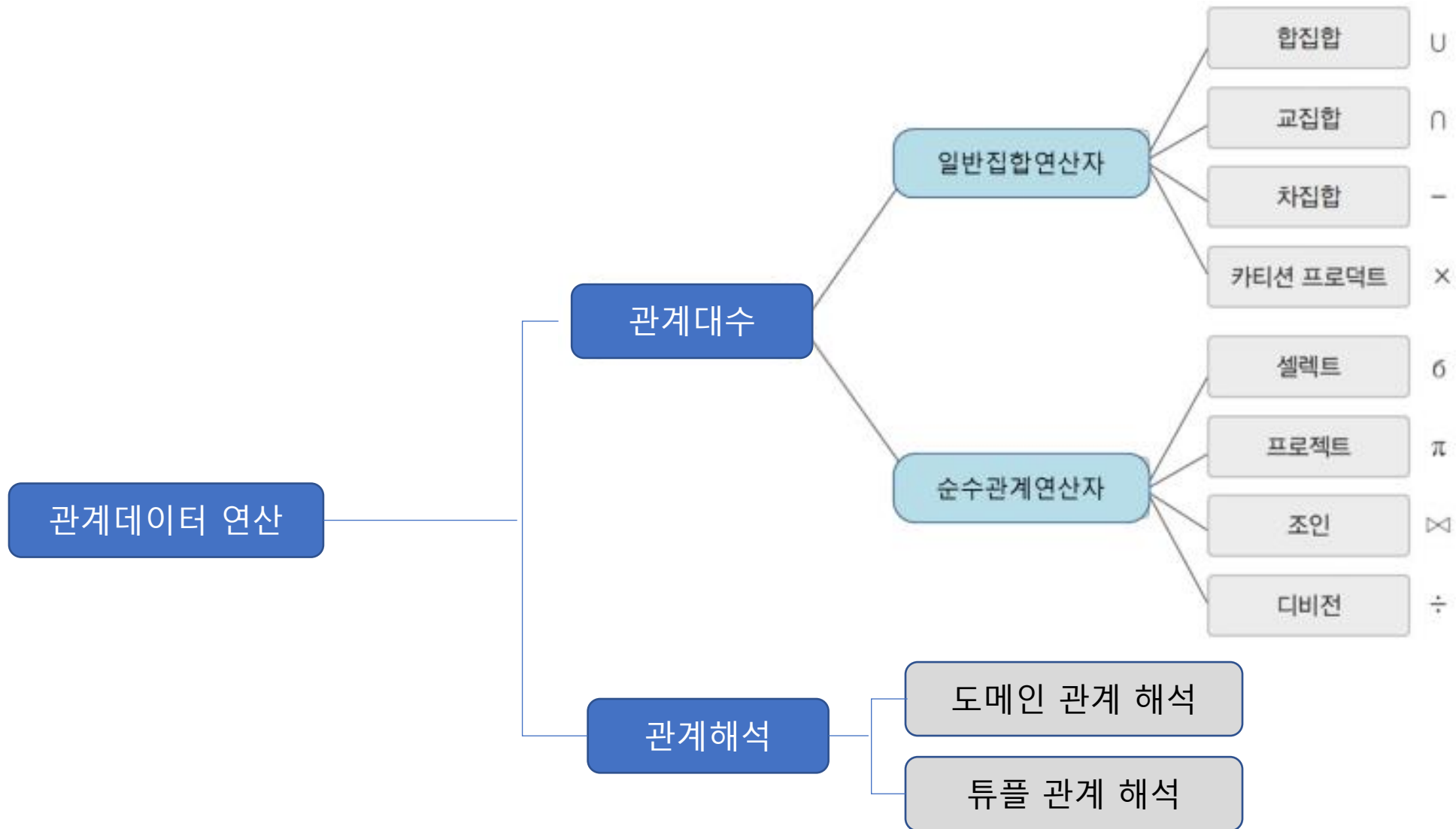
관계 데이터 연산

문혜영

학습 목차

- 01 관계 데이터 연산
- 02 관계 대수
- 03 관계 해석

01 관계 데이터 연산

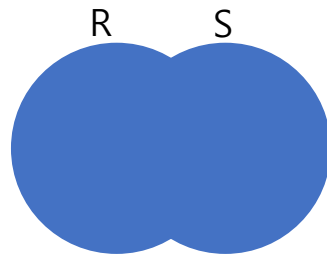


01 관계 데이터 연산

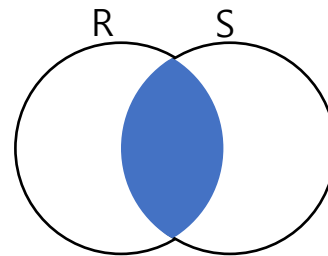
- 개념
 - 관계형 데이터베이스에서 데이터를 관리하기 위해 사용되는 연산
- 관계대수
 - 원하는 결과를 얻기 위해 데이터의 처리 과정을 순서대로 기술
 - 절차적 언어
- 관계해석
 - 원하는 결과를 얻기 위해 처리를 원하는 데이터가 무엇인지만 기술
 - 비절차적 언어

02 관계대수(일반 집합 연산자)

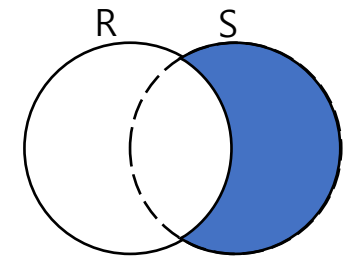
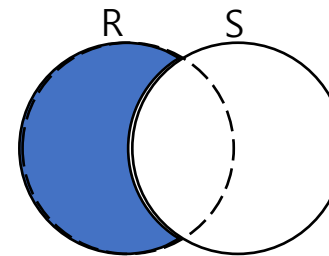
- 일반 집합 연산자의 특징
 - 2개의 피연산자 필요(2개의 릴레이션 대상으로 수행)
- 합집합, 교집합, 차집합은 두 릴레이션이 합병가능해야 함
 - 두 릴레이션의 차수가 동일해야 함
 - 두 릴레이션에서 서로 대응되는 속성의 도메인이 같아야 함



[합집합]



[교집합]



[차집합]

- 합병이 불가능한 예

R			S		
고객번호	이름	나이	회원번호	이름	주소
A1001	이다이	19	A1001	이다이	서울
A1002	박도도	20	A1002	박도도	제주
A1003	최라라	23	A1003	최라라	광주

- 합병이 가능한 예

R			S		
고객번호	이름	나이	회원번호	이름	나이
A1001	이다이	19	B1001	김아라	22
A1002	박도도	20	B1002	하마음	21
A1003	최라라	23	B1003	나오라	23



RUS		
고객번호	이름	나이
A1001	이다이	19
A1002	박도도	20
A1003	최라라	23
B1001	김아라	22
B1002	하마음	21
B1003	나오라	23

02 관계대수(일반 집합 연산자)

- ① 합집합(union)

- 두 릴레이션 R과 S의 합집합은 $R \cup S$ 로 나타낸다.
- 릴레이션 R에 속하거나 릴레이션 S에 속하는 모든 튜플로 결과 릴레이션이 구성된다.




- 결과 릴레이션의 특성
 - 차수는 릴레이션 R과 S의 차수와 같음
 - 카디널리티는 릴레이션 R과 S의 카디널리티를 더한 것과 같거나 적음
- $R \cup S = S \cup R$, $(R \cup S) \cup T = R \cup (S \cup T)$

02 관계대수(일반 집합 연산자)

- ② 교집합(intersection)

- 두 릴레이션 R과 S의 교집합을 $R \cap S$ 로 나타낸다.
- 릴레이션 R과 릴레이션 S에 모두 속하는 튜플로 결과 릴레이션이 구성된다.

R		S		$R \cap S$		
번호	이름	번호	이름		번호	이름
100	정수현	100	정수현		100	정수현
200	김치국	101	윤서윤			
300	고아라	102	이수진			

- 결과 릴레이션의 특성
 - 차수는 릴레이션 R과 S의 차수와 같음
 - 카디널리티는 릴레이션 R과 S의 카디널리티를 더한 것과 같거나 적음
- $R \cap S = S \cap R$, $(R \cap S) \cap T = R \cap (S \cap T)$

02 관계대수(일반 집합 연산자)

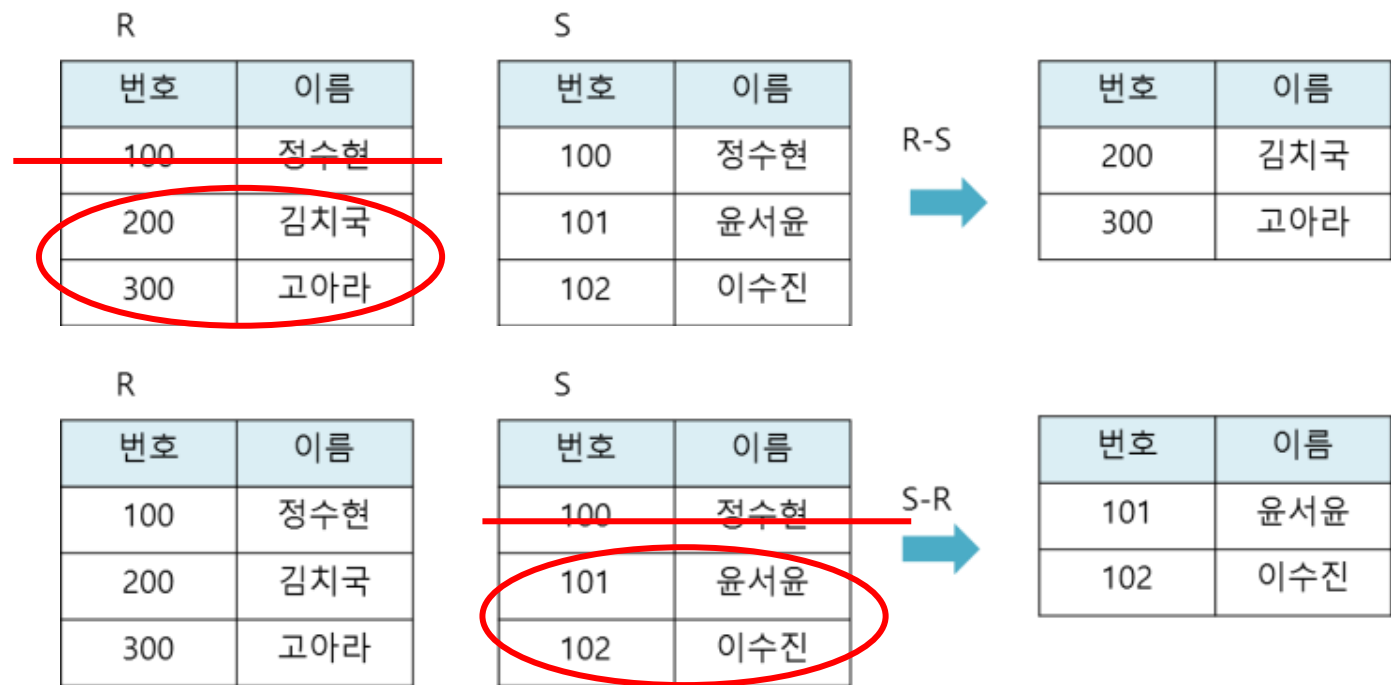
- ③ 차집합(difference)

- 두 릴레이션 R과 S의 차집합은 $R-S$ 로 나타낸다.
- 릴레이션 R에는 존재하고 릴레이션 S에는 존재하지 않는 튜플로 결과 릴레이션이 구성된다.

- 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 관계대수(일반 집합 연산자)

- 카티션 프로덕트(cartesian product)
 - 두 릴레이션 R과 S의 카티션 프로덕트는 $R \times S$ 로 나타낸다.
 - 릴레이션 R에 속한 각 튜플과 릴레이션 S에 속한 각 튜플을 모두 연결하여 만들어진 새로운 튜플로 결과 릴레이션이 구성된다.

- 결과 릴레이션의 특성

- 차수는
릴레이션 R과 S의 차수를 더한 것과 같음
- 카디널리티는
릴레이션 R과 S의 카디널리티를 곱한 것과 같음

R		S	
번호	이름	번호	수량
100	정수현	100	25
200	김치국	101	35
300	고아라	102	45

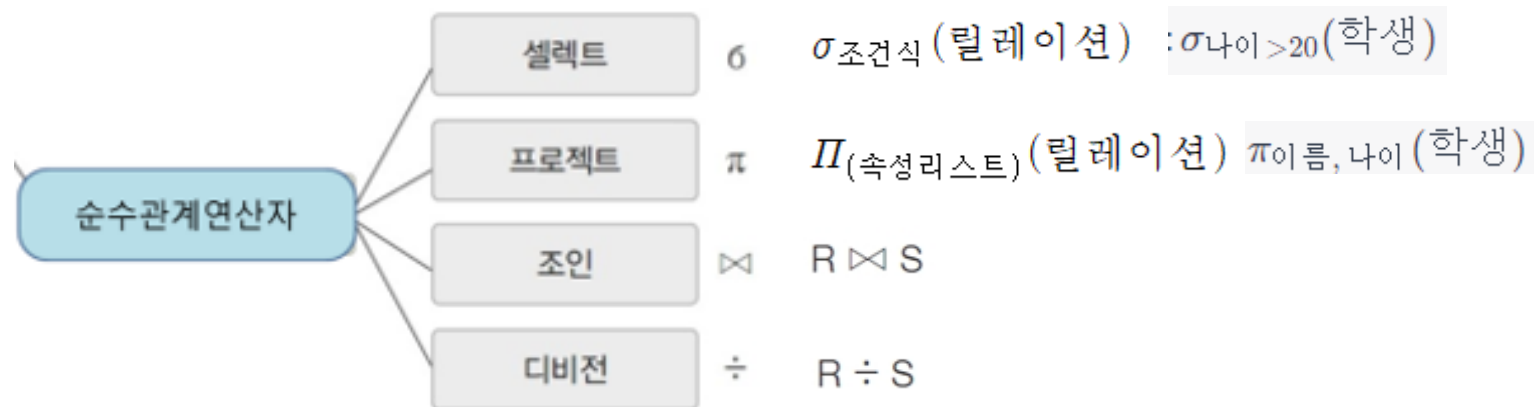


R번호	R이름	S번호	S수량
100	정수현	100	25
100	정수현	101	35
100	정수현	102	45
200	김치국	100	25
200	김치국	101	35
200	김치국	102	45
300	고아라	100	25
300	고아라	101	35
300	고아라	102	45

- $R \times S = S \times R$, $(R \times S) \times T = R \times (S \times T)$

02 관계대수(순수 관계 연산자)

- 집합관계연산은 전통적인 집합 연산을 릴레이션에 적용한 것이며, 대체로 두 릴레이션 간의 연산을 중심으로 합니다.
- 순수관계연산은 관계 대수의 기본 연산들로, 릴레이션 내부의 속성과 튜플에 특화된 연산들을 포함합니다.



02 관계대수(순수 관계 연산자)

- 선택(Select, σ)

- 릴레이션에서 조건에 만족하는 튜플을 검색한다. $\sigma_{조건식}$ (릴레이션)
- 비교 연산자(>, \geq , <, \leq , =, \neq)와 논리 연산자(AND(\wedge), OR(\vee), NOT(\neg))를 이용해 작성
- 학생 릴레이션에서 점수가 90 이상인 튜플을 검색하시오 $\sigma_{점수 \geq 90}$ (학생)

[학생]

학번	과목	교수	점수
100	정보통신	정민기	90
100	웹디자인	홍길동	80
103	정보통신	정민기	86
103	웹디자인	홍길동	95
104	사이버안보	성수현	88

\Rightarrow

$\sigma_{점수 \geq 90}$ (학생)

학번	과목	교수	점수
100	정보통신	정민기	90
103	웹디자인	홍길동	95

- 예 $\sigma_{등급 = 'gold'}$ (고객)

- 예 : 고객 릴레이션에서
등급이 gold이고, 적립금이 2000 이상인 튜플 검색

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

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

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

02 관계대수(순수 관계 연산자)

예제) $\sigma_{Avg \geq 90}(\text{성적})$: <성적> 릴레이션에서 '평균'(Avg)이 90점 이상인 튜플들을 추출하시오.

<성적>

Name	Kor	Eng	Mat	Tot	Avg
고회식	100	90	100	290	96.6
김은소	80	80	100	260	86.6
최미경	100	70	80	250	83.3
김준용	90	100	90	280	93.3
윤정희	85	95	90	270	90.0



02 관계대수(순수 관계 연산자)

- 프로젝트(Project, Π)

- 릴레이션에서 주어진 속성들의 값으로만 구성된 열을 선택한다. $\Pi_{\text{(속성리스트)}}(\text{릴레이션})$
- 학생 릴레이션에서 과목과 점수를 검색하시오. $\Pi_{\text{(과목, 점수)}}(\text{학생})$

[학생]					$\Pi_{\text{(과목, 점수)}}(\text{학생})$	
학번	과목	교수	점수		과목	점수
100	정보통신	정민기	90	\Rightarrow	정보통신	90
100	웹디자인	홍길동	80		웹디자인	80
103	정보통신	정민기	86		정보통신	86
103	웹디자인	홍길동	95		웹디자인	95
104	사이버안보	정수현	88		사이버안보	88

$\Pi_{\text{(학번)}}(\text{학생})$

학번
100
103
104

$\Pi_{\text{(학번, 과목, 점수)}}(\text{학생})$

$\Pi_{\text{(학번, 과목, 점수)}}(\sigma_{\text{교수} = \text{'정민기'}}(\text{학생}))$

02 관계대수(순수 관계 연산자)

예제) $\pi_{\text{Name, Avg}}(\text{성적})$: <성적> 릴레이션에서 'Name'과 'Avg' 속성을 추출하시오.

<성적>

Name	Kor	Eng	Mat	Tot	Avg
고회식	100	90	100	290	96.6
김은소	80	80	100	260	86.6
최미경	100	70	80	250	83.3
김준용	90	100	90	280	93.3
윤정희	85	95	90	270	90.0



02 관계대수(순수 관계 연산자)

- 조인(Join, \bowtie)
 - 공통 속성을 이용하여 2개 이상의 릴레이션을 연결하여 새로운 릴레이션을 생성
릴레이션1 \bowtie_N 릴레이션2
 - 동등조인 (equi-join) 이라고도 함
사원 $\bowtie_{(부서 = 부서)}$ 인사

[사원]			[인사]		사원 \bowtie 인사				
부서	사번	성명	부서	부서명	사원.부서	사번	성명	인사.부서	부서명
A	100	정민기	A	기획부	A	100	정민기	A	기획부
B	101	홍길동	B	총무부	B	101	홍길동	B	총무부
C	102	이주호	C	인사부	C	102	이주호	C	인사부
D	103	정수현							

02 관계대수(순수 관계 연산자)

예제) 성적 $\bowtie_{No=No}$ 주소록 : <성적> 릴레이션과 <학적부> 릴레이션을 'No' 속성을 기준으로 조인하세요.

<성적>

No	Name	py	data	Mat
9801	고회식	100	90	100
9802	김은소	80	80	100
9803	최미경	100	70	80
9804	김준용	90	100	90
9805	윤정희	85	95	90

<주소록>

No	Addr
9801	망원동
9802	서교동
9803	성산동
9804	합정동
9805	공덕동



02 관계대수(순수 관계 연산자)

- 조인의 종류

1. 결합 결과의 행에 따른 분류:

1. 내부 조인 (Inner Join)

2. 외부 조인 (Outer Join)

- 왼쪽 외부 조인 (Left Outer Join)

- 오른쪽 외부 조인 (Right Outer Join)

- 전체 외부 조인 (Full Outer Join)

2. 조인 조건의 유형에 따른 분류:

1. 세타 조인 (Theta Join)

2. 등가 조인 (Equi Join)

3. 자연 조인 (Natural Join)

02 관계대수(순수 관계 연산자)

- 조인의 종류

- 내부 조인 (Inner Join):

두 테이블이 조인 조건을 만족하는 행만 반환

- 외부 조인 (Outer Join):

조인 조건을 만족하지 않는 행도 포함

- 왼쪽 외부 조인 (Left Outer Join): 왼쪽 테이블의 모든 행과 오른쪽 테이블의 일치하는 행을 반환
- 오른쪽 외부 조인 (Right Outer Join): 오른쪽 테이블의 모든 행과 왼쪽 테이블의 일치하는 행을 반환
- 전체 외부 조인 (Full Outer Join): 두 테이블의 모든 행을 반환하며, 일치하지 않는 행은 NULL 값을 가짐

employees

emp_id	emp_name	dept_id
1	Alice	10
2	Bob	10
3	Charlie	20
4	David	30
5	Eve	NULL

departments

dept_id	dept_name
10	Sales
20	IT
30	HR
40	Finance

employees와 departments를 dept_id를 기준으로 결합

emp_id	emp_name	dept_id	dept_name
1	Alice	10	Sales
2	Bob	10	Sales
3	Charlie	20	IT
4	David	30	Finance

emp_id	emp_name	dept_id	dept_name
1	Alice	10	Sales
2	Bob	10	Sales
3	Charlie	20	IT
4	David	30	HR
5	Eve	NULL	NULL

emp_id	emp_name	dept_id	dept_name
1	Alice	10	Sales
2	Bob	10	Sales
3	Charlie	20	IT
4	David	30	HR
NULL	NULL	40	Finance

emp_id	emp_name	dept_id	dept_name
1	Alice	10	Sales
2	Bob	10	Sales
3	Charlie	20	IT
4	David	30	HR
5	Eve	NULL	NULL
NULL	NULL	40	Finance

내부 조인 (Inner Join):

Left Outer Join

Right Outer Join

Full Outer Join

02 관계대수(순수 관계 연산자)

- 조인의 종류

1. 조인 조건의 유형에 따른 분류:

1. 세타 조인 (Theta Join): 두 테이블 간에 임의의 비교 조건(예: =, <, >, <=, >= 등)을 사용하여 조인하는 방식입니다.
2. 동등 조인 (Equi Join): 두 테이블을 동등 비교 연산자(=)를 사용하여 조인합니다. 세타 조인의 특별한 경우.
3. 자연 조인 (Natural Join): 두 테이블 간에 동일한 이름을 가진 모든 열을 기준으로 동등 조인을 수행

employees			departments	
emp_id	emp_name	dept_id	dept_id	dept_name
1	Alice	10	10	Sales
2	Bob	10	20	IT
3	Charlie	20	30	HR
4	David	30	40	Finance
5	Eve	NULL		

Theta Join
조건 :
Dept_id가 10보다 큰 직원

emp_id	emp_name	dept_id	dept_id	dept_name
3	Charlie	20	20	IT
4	David	30	30	HR

Equi Join

조건 :
Employees.dept_id=departments.dept_id

emp_id	emp_name	dept_id	dept_id	dept_name
1	Alice	10	10	Sales
2	Bob	10	10	Sales
3	Charlie	20	20	IT
4	David	30	30	HR

Natural Join

두 테이블 간에 **동일한 이름**을 가진 모든 열을 기준으로 동등 조인을 수행하며 중복된 열은 한 번만 표시됨

emp_id	emp_name	dept_id	dept_name
1	Alice	10	Sales
2	Bob	10	Sales
3	Charlie	20	IT
4	David	30	HR

02 관계대수(순수 관계 연산자)

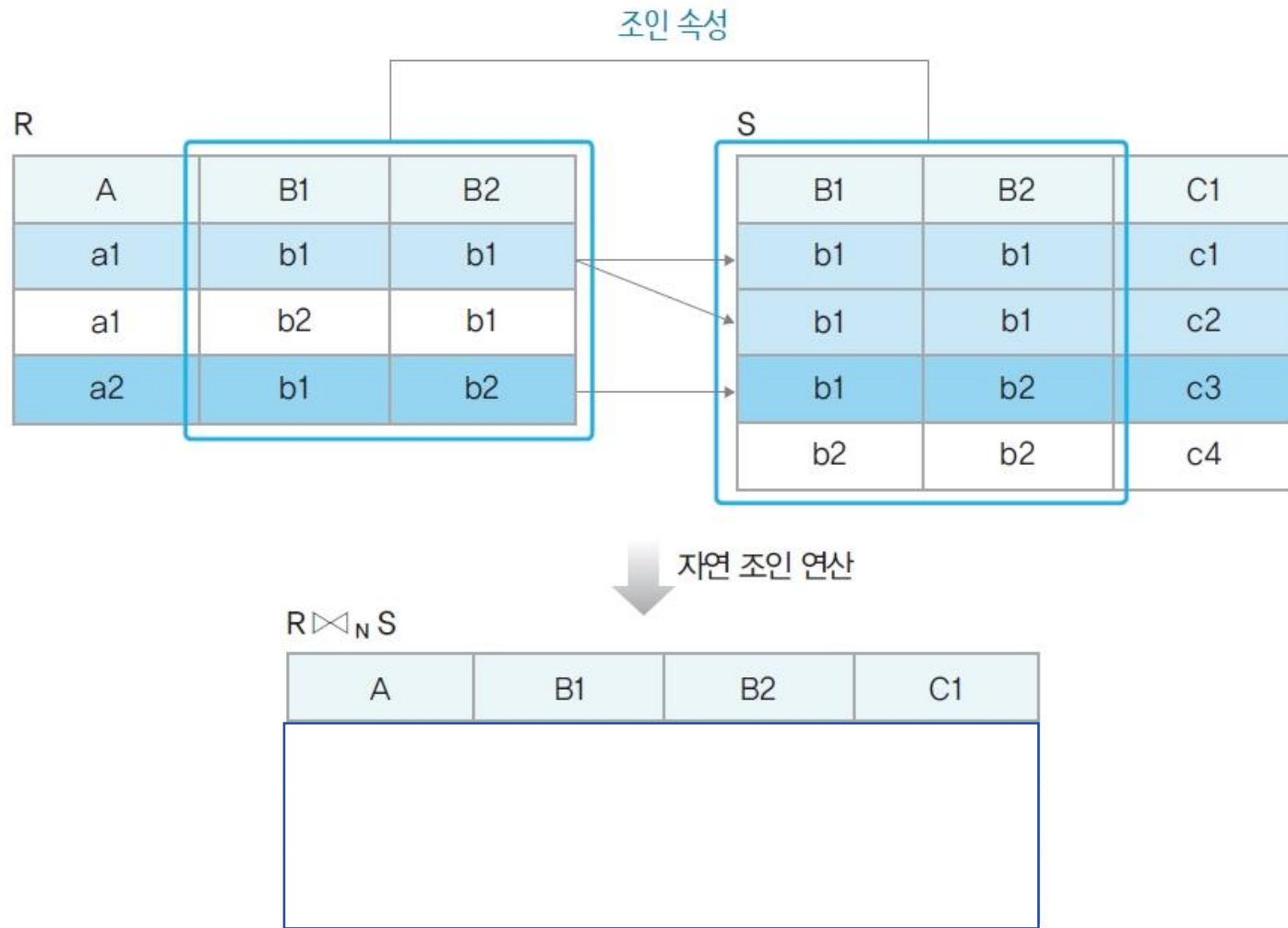
고객 릴레이션

주문 릴레이션

자연 조인 연산

고객 \bowtie_N 주문

02 관계대수(순수 관계 연산자)



02 관계대수(순수 관계 연산자)

- 디비전(Division, \div)
 - B 릴레이션의 모든 조건을 만족하는 튜플을 A 릴레이션에서 구하는 연산이다.
 - 릴레이션1 \div 릴레이션2

[고객]					
아이디	이름	나이	등급	직업	포인트
app	김현아	20	gold	학생	1000
nul	정소희	25	vip	프리랜서	2500
car	원유인	28	gold	교사	5500
klk	정지선	22	silver	학생	0

[골드]	
등급	
gold	

- 고객 \div 골드

02 관계대수(순수 관계 연산자)

- 주문내역 ÷ 제품

[주문내역]

주문고객	제품이름	제조업체
aaa	우동	한양
ccc	파이	미양
bbb	만두	한양
aaa	만두	한양
ccc	만두	한양

[제품]

제품이름
우동
만두

[제조]

제품이름	제조업체
만두	한양

주문내역 ÷ 제품

주문고객	제조업체
aaa	한양

주문내역 ÷ 제조

주문고객
bbb
aaa
ccc

02 관계대수(평가)

- 다음의 릴레이션 R1과 R2에 대한 관계대수 $R1 \div R2$ 의 결과 릴레이션으로 옳은 것은? (단, 릴레이션 R1.C2와 R2.C2는 동일한 도메인에서 정의되었다)

R1(C1, C2)	
C1	C2
1	A
2	C
1	E
1	B
3	J
4	R
3	B
2	B
5	R
3	A
4	A

R2(C2)
C2
A
B

①

C1
3
4

②

C1	C2
1	A
2	B

③

C1
1
3

④

C1
2
3

02 관계대수(평가)

- 다음 관계 대수 연산의 수행 결과로 옳은 것은?

(단, Π 는 프로젝트, σ 는 선택, \bowtie 은 자연 조인을 나타내는 연산자이다)

관계 대수: $\Pi_{\text{고객번호, 상품코드}}(\sigma_{\text{가격} \leq 40}(\text{구매} \bowtie_N \text{상품}))$

구매

고객번호	상품코드
100	P1
200	P2
100	P3
100	P2
200	P1
300	P2

상품

상품코드	비용	가격
P1	20	35
P2	50	65
P3	10	27
P4	20	45
P5	30	50
P6	40	55

①

고객번호	상품코드
100	P1
100	P3

②

고객번호	상품코드
100	P1
200	P1

③

고객번호	상품코드
100	P1
100	P3
200	P1

④

고객번호	상품코드
200	P2
100	P2
300	P2

02 관계대수(평가)

- 다음은 어느 기관의 데이터베이스 테이블을 나타낸 것이다. 다음 관계대수식을 적용한 결과의 카디널리티(cardinality)로 옳은 것은?

직원		
직원 번호	이름	부서
10	김	B20
20	이	A10
30	박	A10
40	최	C30

부서	
부서 번호	부서명
A10	기획과
B20	인사과
C30	총무과

정책		
정책 번호	정책명	제안자
100	인력양성	40
200	주택자금	20
300	친절교육	10
400	성과금	10
500	신규고용	20

$\Pi_{\text{이름, 부서명, 정책명}}(\text{부서} \bowtie_{\text{부서번호} = \text{부서}} (\Pi_{\text{정책명, 이름, 부서}} (\text{정책} \bowtie_{\text{제안자} = \text{직원번호}} \text{직원})))$

- ① 3
- ② 4
- ③ 5
- ④ 6

02 관계대수(평가)

직원		
직원 번호	이름	부서
10	김	B20
20	이	A10
30	박	A10
40	최	C30

부서	
부서 번호	부서명
A10	기획과
B20	인사과
C30	총무과

정책		
정책 번호	정책명	제안자
100	인력양성	40
200	주택자금	20
300	친절교육	10
400	성과금	10
500	신규고용	20

정책 \bowtie 제안자 = 직원번호 직원

정책 번호	정책명	제안자	이름	부서
100	인력양성	40	최	C30
200	주택자금	20	이	A10
300	친절교육	10	김	B20
400	성과금	10	김	B20
500	신규고용	20	이	A10

\bowtie 이름, 부서명, 정책명(부서 \bowtie 부서번호 = 부서(\bowtie 정책명, 이름, 부서(정책 \bowtie 제안자=직원번호 직원)))

정책명	이름	부서
인력양성	최	C30
주택자금	이	A10
친절교육	김	B20
성과금	김	B20
신규고용	이	A10

부서 \bowtie 부서번호 = 부서(\bowtie 정책명, 이름, 부서(정책 \bowtie 제안자=직원번호 직원))

정책명	이름	부서	부서명
인력양성	최	C30	총무과
주택자금	이	A10	기획과
친절교육	김	B20	인사과
성과금	김	B20	인사과
신규고용	이	A10	기획과

\bowtie 이름, 부서명, 정책명(부서 \bowtie 부서번호 = 부서(\bowtie 정책명, 이름, 부서(정책 \bowtie 제안자=직원번호 직원)))

이름	부서명	정책명
최	총무과	인력양성
이	기획과	주택자금
김	인사과	친절교육
김	인사과	성과금
이	기획과	신규고용



02 관계대수(평가)

- 릴레이션 R의 차수(Degree)가 3, 카디널리티(Cardinality)가 3, 릴레이션 S의 차수가 4, 카디널리티가 4일 때, 두 릴레이션을 카티션 프로덕트(cartesian product)한 결과 릴레이션의 차수와 카디널리티는?
 - ① 4, 4
 - ② 7, 7
 - ③ 7, 12
 - ④ 12, 12

02 관계대수(평가)

- 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 (Left Outer Join)

고객 ⋈ 주문

오른쪽 외부 조인 (Right Outer Join)

전체 외부 조인 (Full Outer Join)

02 관계대수(평가)

- 외부조인

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

왼쪽 외부 조인 (Left Outer Join)

오른쪽 외부 조인 (Right Outer Join)

전체 외부 조인 (Full Outer Join)

02 관계대수(평가)

<사원>

사번	이름
1	고회식
2	김준용

<직원>

사번	이름
2	김준용
3	윤정희

예제1) $\pi_{\text{이름}}(\text{사원}) \cup \pi_{\text{이름}}(\text{직원}) :$

예제2) $\pi_{\text{이름}}(\text{사원}) - \pi_{\text{이름}}(\text{직원}) :$

예제3) $\pi_{\text{이름}}(\text{사원}) \cap \pi_{\text{이름}}(\text{직원}) :$

예제4) $\pi_{\text{이름}}(\text{사원}) \times \pi_{\text{이름}}(\text{직원}) :$

03 관계해석

- 관계해석 (Relational Calculus) 개념
 - 관계형 데이터베이스 모델의 수학적 기반 중 하나
 - 어떤 조건을 만족하는 데이터를 찾기 위한 비절차적 방식
 - "어떻게 (how)"가 아닌 "무엇을 (what)"에 중점을 둔다
 - 예) 모든 학생들의 이름 검색.
 - 관계대수 $\pi_{\text{name}}(\text{Students})$
 - 관계해석(튜플관계해석) $\{t.\text{name} \mid t \in \text{Students}\}$
 - 의미 : "Students 릴레이션에 있는 모든 튜플 t에 대해 t의 'name' 속성 값을 가져오기"
- 튜플 관계해석(Tuple Relational Calculus, TRC)
 - 튜플에 기반한 변수를 사용하여 질의를 표현합니다.
 - 주로 "T"와 같은 변수를 사용하여 릴레이션 내의 튜플을 나타냅니다.
- 도메인 관계해석(Domain Relational Calculus, DRC)
 - 도메인(즉, 속성 값)을 기반으로 질의를 표현
 - \forall (전체에 대해) 및 \exists (존재에 대해)와 같은 수학적 표기법을 사용하여 질의 조건을 명시

- 예) Students 릴레이션에서 나이가 20세 초과인 튜플
 - 튜플 관계해석
$$\{T \mid T \in \text{Students} \wedge T.\text{age} > 20\}$$
 - 도메인 관계해석
$$\{\text{name} \mid \exists \text{age} (\text{name}, \text{age}) \in \text{Students} \wedge \text{age} > 20\}$$

수고하셨습니다 🙌