

# 제4장



## 관계 대수와 SQL

- 4.1 관계 대수
- 4.2 SQL 개요
- 4.3 데이터 정의어와 무결성 제약조건
- 4.4 SELECT문
- 4.5 INSERT, DELETE, UPDATE문
- 4.6 트리거(trigger)와 주장(assertion)
- 4.7 내포된 SQL
  - 연습문제

## 4장. 관계 대수와 SQL

### □ 관계 데이터 모델에서 지원되는 두 가지 정형적인 언어

- ✓ **관계 해석**(relational calculus)
  - 원하는 데이터만 명시하고 질의를 어떻게 수행할 것인가는 명시하지 않는 선언적인 언어
- ✓ **관계 대수**(relational algebra)
  - 어떻게 질의를 수행할 것인가를 명시하는 절차적 언어
  - 관계 대수는 상용 관계 DBMS들에서 널리 사용되는 SQL의 이론적인 기초
  - 관계 대수는 SQL을 구현하고 최적화하기 위해 DBMS의 내부 언어로서도 사용됨

### □ SQL

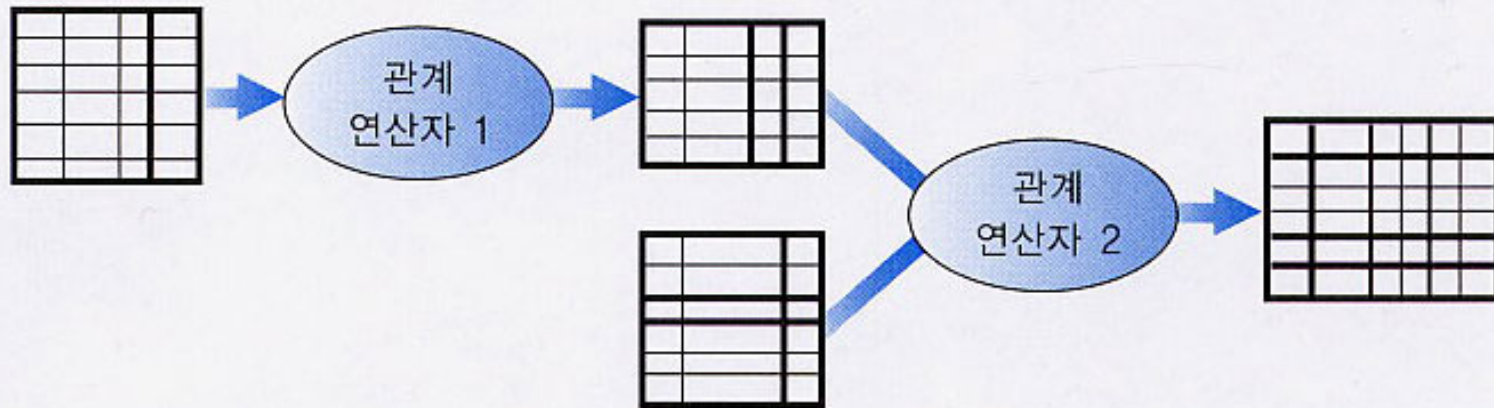
- ✓ 상용 관계 DBMS들의 사실상의 표준 질의어인 SQL을 이해하고 사용할 수 있는 능력은 매우 중요함
- ✓ 사용자는 SQL을 사용하여 관계 데이터베이스에 릴레이션을 정의하고, 관계 데이터베이스에서 정보를 검색하고, 관계 데이터베이스를 갱신하며, 여러 가지 무결성 제약조건들을 명시할 수 있음

## 4.1 관계 대수

### □ 관계 대수

- ✓ 기존의 릴레이션들로부터 새로운 릴레이션을 생성함
- ✓ 릴레이션이나 관계 대수식(이것의 결과도 릴레이션임)에 연산자들을 적용하여 보다 복잡한 관계 대수식을 점차적으로 만들 수 있음
- ✓ 기본적인 연산자들의 집합으로 이루어짐
- ✓ 산술 연산자와 유사하게 단일 릴레이션이나 두 개의 릴레이션을 입력으로 받아 하나의 결과 릴레이션을 생성함
- ✓ 결과 릴레이션은 또 다른 관계 연산자의 입력으로 사용될 수 있음

## 4.1 관계 대수(계속)



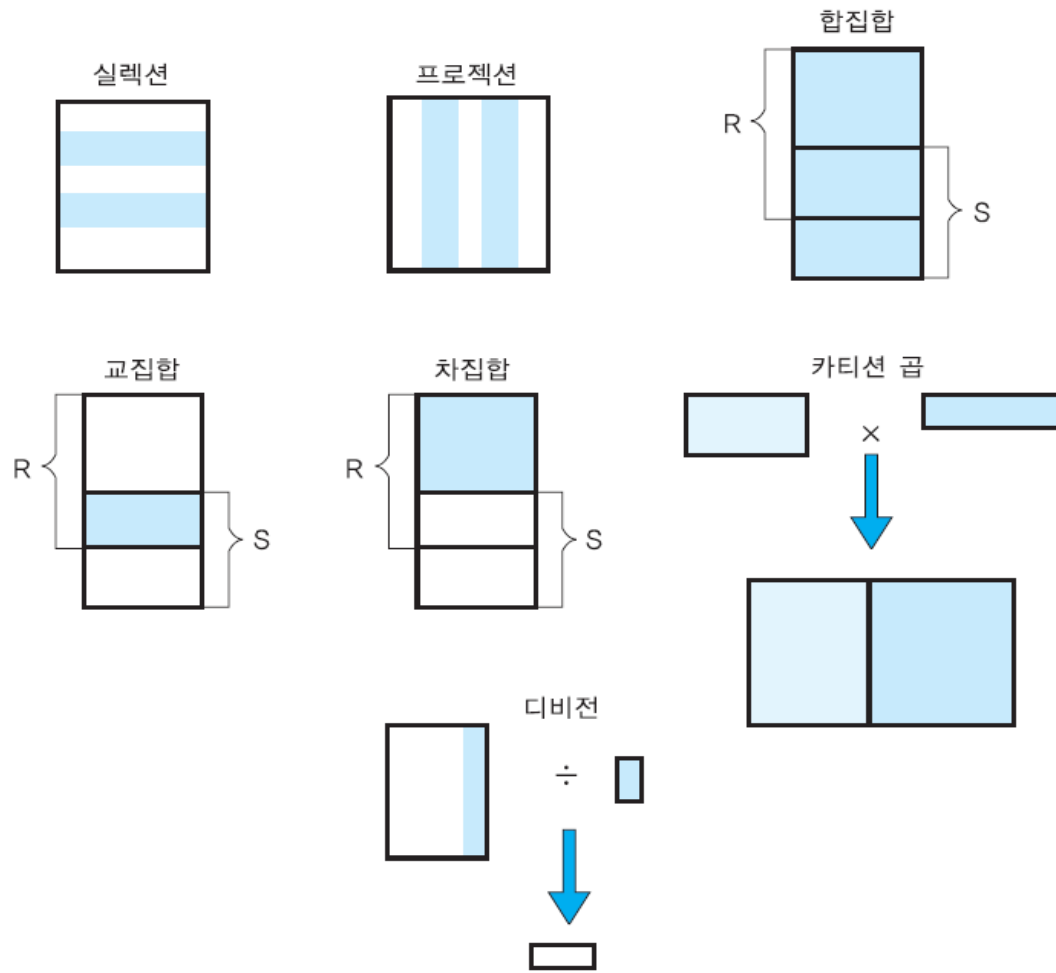
[그림 4.1] 관계 연산자

## 4.1 관계 대수(계속)

〈표 4.1〉 관계 연산자들의 종류와 표기법

분류	연산자	표기법	단항 또는 이항
필수적인 연산자	실렉션(selection)	$\sigma$	단항
	프로젝션(projection)	$\pi$	단항
	합집합(union)	$\cup$	이항
	차집합(difference)	$-$	이항
	카티션 곱(Cartesian product)	$\times$	이항
편의를 위해 유도된 연산자	교집합(intersection)	$\cap$	이항
	세타 조인(theta join)	$\bowtie$	이항
	동등 조인(equijoin)	$\bowtie$	이항
	자연 조인(natural join)	$*$	이항
	세미 조인(semijoin)	$\ltimes$	이항
	디비전(division)	$\div$	이항

## 4.1 관계 대수(계속)



[그림 4.2] 관계 연산자들의 기능

## 4.1 관계 대수(계속)

실렉션



### □ 실렉션 연산자 (형식: $\sigma_{\langle \text{실렉션 조건} \rangle}$ (릴레이션))

- ✓ 한 릴레이션에서 **실렉션 조건**(selection condition)을 만족하는 튜플들의 부분 집합을 생성함
- ✓ 단항 연산자
- ✓ 결과 릴레이션의 차수는 입력 릴레이션의 차수와 같음
- ✓ 결과 릴레이션의 카디널리티는 항상 원래 릴레이션의 카디널리티보다 작거나 같음
- ✓ 실렉션 조건을 **프레디케이트**(predicate)라고도 함
- ✓ 실렉션 조건은 일반적으로 릴레이션의 임의의 애트리뷰트와 상수, = , <>, <=, <, >=, > 등의 비교 연산자, AND, OR, NOT 등의 부울 연산자를 포함할 수 있음

## 4.1 관계 대수(계속)

### □ 실렉션 연산자(계속)

예 : 실렉션

질의: EMPLOYEE 릴레이션에서 3번 부서에 소속된 직원들을 검색하라.

EMPLOYEE

EMPNO	EMPNAME	TITLE	MANAGER	SALARY	DNO
2106	김창섭	대리	1003	2500000	2
3426	박영권	과장	4377	3000000	1
3011	이수민	부장	4377	4000000	3
1003	조민희	과장	4377	3000000	2
3427	최종철	사원	3011	1500000	3
1365	김상원	사원	3426	1500000	1
4377	이성래	사장	^	5000000	2

원하는  
튜플

$\sigma_{DNO=3} (EMPLOYEE)$

RESULT

EMPNO	EMPNAME	TITLE	MANAGER	SALARY	DNO
3011	이수민	부장	4377	4000000	3
3427	최종철	사원	3011	1500000	3





## 4.1 관계 대수(계속)

### □ 프로젝트 연산자 (형식: $\pi_{\langle \text{애트리뷰트 리스트} \rangle}(\text{릴레이션})$ )

- ✓ 한 릴레이션의 애트리뷰트들의 부분 집합을 구함
- ✓ 결과로 생성되는 릴레이션은  $\langle \text{애트리뷰트 리스트} \rangle$ 에 명시된 애트리뷰트들만 가짐
- ✓ 실렉션의 결과 릴레이션에는 중복 튜플이 존재할 수 없지만, 프로젝트 연산의 결과 릴레이션에는 중복된 튜플들이 존재할 수 있음

## 4.1 관계 대수(계속)

예 : 프로젝트

질의: 모든 사원들의 직급을 검색하라.

EMPLOYEE	EMPNO	EMPNAME	TITLE	MANAGER	SALARY	DNO
	2106	김창섭	대리	1003	2500000	2
	3426	박영권	과장	4377	3000000	1
	3011	이수민	부장	4377	4000000	3
	1003	조민희	과장	4377	3000000	2
	3427	최종철	사원	3011	1500000	3
	1365	김상원	사원	3426	1500000	1
	4377	이성래	사장	^	5000000	2

  
 $\pi_{\text{TITLE}}(\text{EMPLOYEE})$

## 4.1 관계 대수(계속)




RESULT

TITLE
대리
과장
부장
과장
사원
사원
사장

The diagram shows a table with 7 rows. The first row is the header 'TITLE'. The subsequent rows are '대리', '과장', '부장', '과장', '사원', '사원', and '사장'. The second, fourth, fifth, and sixth rows are highlighted in light blue. A blue bracket on the right side groups the second, fourth, and fifth rows together. Another blue bracket on the right side groups the fifth and sixth rows together.

중복이 존재하는 릴레이션



RESULT

TITLE
대리
과장
부장
사원
사장

The diagram shows a table with 5 rows. The first row is the header 'TITLE'. The subsequent rows are '대리', '과장', '부장', '사원', and '사장'. The first row is highlighted in light blue.

중복이 제거된 릴레이션

## 4.1 관계 대수(계속)

### □ 집합 연산자

- ✓ 릴레이션이 튜플들의 집합이기 때문에 기존의 집합 연산이 릴레이션에 적용됨
- ✓ 세 가지 집합 연산자: 합집합, 교집합, 차집합 연산자
- ✓ 집합 연산자의 입력으로 사용되는 두 개의 릴레이션은 **합집합 호환**(union compatible)이어야 함
- ✓ 이항 연산자

### □ 합집합 호환

- ✓ 두 릴레이션  $R1(A1, A2, \dots, An)$ 과  $R2(B1, B2, \dots, Bm)$ 이 합집합 호환일 필요 충분 조건은  $n=m$ 이고, 모든  $1 \leq i \leq n$ 에 대해  $\text{domain}(Ai) = \text{domain}(Bi)$

## 4.1 관계 대수(계속)

### 예 : 합집합 호환

아래의 EMPLOYEE 릴레이션 스키마와 DEPARTMENT 릴레이션 스키마는 애트리뷰트 수가 다르므로 합집합 호환이 되지 않는다.

EMPLOYEE (EMPNO, EMPNAME, TITLE, MANAGER, SALARY, DNO)

DEPARTMENT (DEPTNO, DEPTNAME, FLOOR)

그러나 EMPLOYEE 릴레이션에서 DNO를 프로젝션한 결과 릴레이션 ( $\pi_{DNO}(\text{EMPLOYEE})$ ) 과 DEPARTMENT 릴레이션에서 DEPTNO를 프로젝션한 결과 릴레이션 ( $\pi_{DEPTNO}(\text{DEPARTMENT})$ ) 은 애트리뷰트 수가 같으며 DNO와 DEPTNO의 도메인이 같으므로 합집합 호환이다.

## 4.1 관계 대수(계속)

### □ 합집합 연산자 (형식: 릴레이션1 $\cup$ 릴레이션2)

- ✓ 두 릴레이션 R과 S의 합집합  $R \cup S$ 는 R 또는 S에 있거나 R과 S 모두에 속한 튜플들로 이루어진 릴레이션
- ✓ 결과 릴레이션에서 중복된 튜플들은 제외됨
- ✓ 결과 릴레이션의 차수는 R 또는 S의 차수와 같으며, 결과 릴레이션의 애트리뷰트 이름들은 R의 애트리뷰트들의 이름과 같거나 S의 애트리뷰트들의 이름과 같음

## 4.1 관계 대수(계속)

예 : 합집합

질의: 김창섭이 속한 부서이거나 개발 부서의 부서번호를 검색하라.

EMPLOYEE	EMPNO	EMPNAME	TITLE	MANAGER	SALARY	DNO
	2106	김창섭	대리	1003	2500000	2
	3426	박영권	과장	4377	3000000	1
	3011	이수민	부장	4377	4000000	3
	1003	조민희	과장	4377	3000000	2
	3427	최종철	사원	3011	1500000	3
	1365	김상원	사원	3426	1500000	1
	4377	이성래	사장	^	5000000	2



$RESULT1 \leftarrow \pi_{DNO}(\sigma_{EMPNAME='김창섭'}(EMPLOYEE))$

RESULT1	DNO
	2

## 4.1 관계 대수(계속)

DEPARTMENT	DEPTNO	DEPTNAME	FLOOR
	1	영업	8
	2	기획	10
	3	개발	9
	4	총무	7

RESULT2  $\leftarrow \pi_{\text{DEPTNO}}(\sigma_{\text{DEPTNAME}='개발'}(\text{DEPARTMENT}))$

RESULT2	DEPTNO
	3

RESULT3  $\leftarrow \text{RESULT1} \cup \text{RESULT2}$

RESULT3	DEPTNO
	2
	3



## 4.1 관계 대수(계속)

### □ 교집합 연산자 (형식: 릴레이션1 $\cap$ 릴레이션2)

- ✓ 두 릴레이션 R과 S의 교집합  $R \cap S$ 는 R과 S 모두에 속한 튜플들로 이루어진 릴레이션
- ✓ 결과 릴레이션의 차수는 R 또는 S의 차수와 같으며, 결과 릴레이션의 애트리뷰트 이름들은 R의 애트리뷰트들의 이름과 같거나 S의 애트리뷰트들의 이름과 같음

## 4.1 관계 대수(계속)

예 : 교집합

질의: 김창섭 또는 최종철이 속한 부서이면서 기획 부서의 부서번호를 검색하라.

EMPLOYEE	EMPNO	EMPNAME	TITLE	MANAGER	SALARY	DNO
	2106	김창섭	대리	1003	2500000	2
	3426	박영권	과장	4377	3000000	1
	3011	이수민	부장	4377	4000000	3
	1003	조민희	과장	4377	3000000	2
	3427	최종철	사원	3011	1500000	3
	1365	김상원	사원	3426	1500000	1
	4377	이성래	사장	^	5000000	2

$RESULT1 \leftarrow \pi_{DNO} (\sigma_{EMPNAME='김창섭' \text{ OR } EMPNAME='최종철'} (EMPLOYEE))$

RESULT1	DNO
	2
	3

## 4.1 관계 대수(계속)

DEPARTMENT	DEPTNO	DEPTNAME	FLOOR
	1	영업	8
	2	기획	10
	3	개발	9
	4	총무	7

$\text{RESULT2} \leftarrow \pi_{\text{DEPTNO}} (\sigma_{\text{DEPTNAME}='기획'} (\text{DEPARTMENT}))$



RESULT2	DEPTNO
	2

$\text{RESULT3} \leftarrow \text{RESULT1} \cap \text{RESULT2}$

RESULT3	DEPTNO
	2

## 4.1 관계 대수(계속)

### □ 차집합 연산자 (형식: 릴레이션1 - 릴레이션2)

- ✓ 두 릴레이션 R과 S의 차집합  $R - S$ 는 R에는 속하지만 S에는 속하지 않은 튜플들로 이루어진 릴레이션
- ✓ 결과 릴레이션의 차수는 R 또는 S의 차수와 같으며, 결과 릴레이션의 애트리뷰트 이름들은 R의 애트리뷰트들의 이름과 같거나 S의 애트리뷰트들의 이름과 같음

## 4.1 관계 대수(계속)

### 예 : 차집합

질의: 소속된 직원이 한 명도 없는 부서의 부서번호를 검색하라.

DEPARTMENT	DEPTNO	DEPTNAME	FLOOR
	1	영업	8
	2	기획	10
	3	개발	9
	4	총무	7

RESULT1  $\leftarrow \pi_{\text{DEPTNO}}(\text{DEPARTMENT})$

RESULT1	DEPTNO
	1
	2
	3
	4

## 4.1 관계 대수(계속)

EMPLOYEE	EMPNO	EMPNAME	TITLE	MANAGER	SALARY	DNO
	2106	김창섭	대리	1003	2500000	2
	3426	박영권	과장	4377	3000000	1
	3011	이수민	부장	4377	4000000	3
	1003	조민희	과장	4377	3000000	2
	3427	최종철	사원	3011	1500000	3
	1365	김상원	사원	3426	1500000	1
	4377	이성래	사장	^	5000000	2

$\text{RESULT2} \leftarrow \pi_{\text{DNO}}(\text{EMPLOYEE})$

RESULT2	DNO
	2
	1
	3

$\text{RESULT3} \leftarrow \text{RESULT1} - \text{RESULT2}$

RESULT3	DNO
	4

## 4.1 관계 대수(계속)

### □ 카티션 곱 연산자 (형식: $R \times S$ )

- ✓ 카디널리티가  $i$ 인 릴레이션  $R(A_1, A_2, \dots, A_n)$ 과 카디널리티가  $j$ 인 릴레이션  $S(B_1, B_2, \dots, B_m)$ 의 카티션 곱  $R \times S$ 는 차수가  $n+m$ 이고, 카디널리티가  $i*j$ 이고, 애트리뷰트가  $(A_1, A_2, \dots, A_n, B_1, B_2, \dots, B_m)$ 이며,  $R$ 과  $S$ 의 튜플들의 모든 가능한 조합으로 이루어진 릴레이션
- ✓ 카티션 곱의 결과 릴레이션의 크기가 매우 클 수 있으며, 사용자가 실제로 원하는 것은 카티션 곱의 결과 릴레이션의 일부인 경우가 대부분이므로 카티션 곱 자체는 유용한 연산자가 아님

## 4.1 관계 대수(계속)

### 예 : 카티션 곱

질의: EMPLOYEE 릴레이션과 DEPARTMENT 릴레이션의 카티션 곱을 구하라.





## 4.1 관계 대수(계속)

### □ 관계 대수의 완전성

- ✓ 실렉션, 프로젝션, 합집합, 차집합, 카티션 곱은 관계 대수의 필수적인 연산자
- ✓ 다른 관계 연산자들은 필수적인 관계 연산자를 두 개 이상 조합하여 표현할 수 있음
- ✓ 임의의 질의어가 적어도 필수적인 관계 대수 연산자들만큼의 표현력을 갖고 있으면 **관계적으로 완전**(relationally complete)하다고 말함

## 4.1 관계 대수(계속)

### □ 조인 연산자

- ✓ 두 개의 릴레이션으로부터 연관된 튜플들을 결합하는 연산자
- ✓ 관계 데이터베이스에서 두 개 이상의 릴레이션들의 관계를 다루는데 매우 중요한 연산자
- ✓ 세타 조인(theta join), 동등 조인(equijoin), 자연 조인(natural join), 외부 조인(outer join), 세미 조인(semijoin) 등

## 4.1 관계 대수(계속)

### □ 세타( $\Theta$ ) 조인 (형식: $R \bowtie_{\langle \text{조인조건} \rangle} S$ )

- ✓ 두 릴레이션  $R(A_1, A_2, \dots, A_n)$ 과  $S(B_1, B_2, \dots, B_m)$ 의 세타 조인의 결과는 차수가  $n+m$ 이고, 애트리뷰트가  $(A_1, A_2, \dots, A_n, B_1, B_2, \dots, B_m)$ 이며, 조인 조건을 만족하는 튜플들로 이루어진 릴레이션
- ✓ 조인 조건은  $R.A_i \Theta S.B_j$ 의 형태로 주어지며  $\Theta$ 는  $\{=, <>, <=, <, >=, >\}$  중의 하나
- ✓ 세타 조인 결과는 두 릴레이션의 카티션 곱에 조인 조건을 적용한 결과와 동일

$$\checkmark R \bowtie_{R.A_i \Theta S.B_j} S \equiv \sigma_{R.A_i \Theta S.B_j} (R \times S)$$

### □ 동등 조인

- ✓ 동등 조인은 세타 조인 중에서 비교 연산자가  $=$ 인 조인

## 4.1 관계 대수(계속)

### 예 : 동등 조인

질의: EMPLOYEE 릴레이션과 DEPARTMENT 릴레이션을 동등 조인하라.

EMPLOYEE	EMPNO	EMPNAME	DNO
	2106	김창섭	2
	3426	박영권	1
	3011	이수민	3
	1003	조민희	2
	3427	최종철	3

DEPARTMENT	DEPTNO	DEPTNAME
	1	영업
	2	기획
	3	개발
	4	총무

EMPLOYEE ⋈<sub>DNO=DEPTNO</sub> DEPARTMENT

RESULT	EMPNO	EMPNAME	DNO	DEPTNO	DEPTNAME
	2106	김창섭	2	2	기획
	3426	박영권	1	1	영업
	3011	이수민	3	3	개발
	1003	조민희	2	2	기획
	3427	최종철	3	3	개발

## 4.1 관계 대수(계속)

### □ 자연 조인 (형식: $R \bowtie S$ )

- ✓ 두 릴레이션의 공통된 애트리뷰트에 대해 동등 조인을 수행하고, 동등 조인의 결과 릴레이션에 있는 두 개의 조인 애트리뷰트 중 하나를 제외한 조인
- ✓ 여러 가지 조인 연산자들 중에서 가장 자주 사용됨
- ✓ 실제로 관계 데이터베이스에서 대부분의 질의는 실렉션, 프로젝션, 자연 조인으로 표현 가능

R과 S의 자연 조인

R	A	B	C
	a1	b1	c1
	a2	b2	c2

$R * S$

S	C	D	E
	c1	d1	e1
	c3	d2	e2



RESULT

A	B	C	D	E
a1	b1	c1	d1	e1

## 4.1 관계 대수(계속)

### 예 : 자연 조인

질의: EMPLOYEE 릴레이션과 DEPARTMENT 릴레이션을 자연 조인하라.

EMPLOYEE	EMPNO	EMPNAME	DNO
	2106	김창섭	2
	3426	박영권	1
	3011	이수민	3
	1003	조민희	2
	3427	최종철	3

DEPARTMENT	DEPTNO	DEPTNAME
	1	영업
	2	기획
	3	개발
	4	총무

EMPLOYEE \* ~~DNO, DEPTNO~~ DEPARTMENT

RESULT	EMPNO	EMPNAME	DNO	DEPTNAME
	2106	김창섭	2	기획
	3426	박영권	1	영업
	3011	이수민	3	개발
	1003	조민희	2	기획
	3427	최종철	3	개발

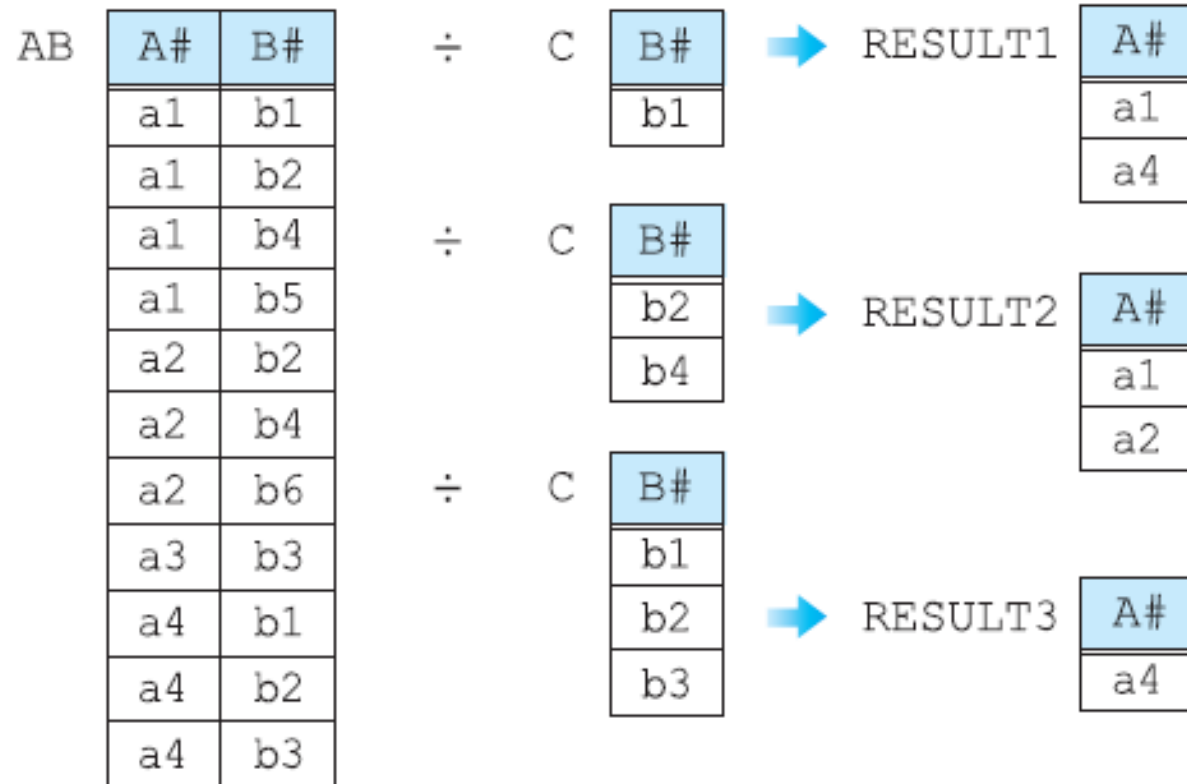
## 4.1 관계 대수(계속)

### □ 디비전 연산자 (형식: $R \div S$ )

- ✓ 차수가  $n+m$ 인 릴레이션  $R(A_1, A_2, \dots, A_n, B_1, B_2, \dots, B_m)$ 과 차수가  $m$ 인 릴레이션  $S(B_1, B_2, \dots, B_m)$ 의 디비전  $R \div S$ 는 차수가  $n$ 이고,  $S$ 에 속하는 모든 튜플  $u$ 에 대하여 튜플  $tu$ (튜플  $t$ 와 튜플  $u$ 을 결합한 것)가  $R$ 에 존재하는 튜플  $t$ 들의 집합
- ✓ 릴레이션  $S$ 의 모든(ALL) 튜플 값과 쌍을 이루는 릴레이션  $R$ 의  $A_1, A_2, \dots, A_n$  값
- ✓ “모든 ...에 대해 ~하는” 형태의 질의에 사용될 수 있음
- ✓ SQL로 표현할 때 동치를 활용: ~하지 않는 ...가 없다

## 4.1 관계 대수(계속)

예 : 디비전





## 4.1 관계 대수(계속)

### □ 관계 대수 질의의 예

#### 예 : 실렉션, 프로젝션

질의: 2번 부서나 3번 부서에 근무하는 모든 직원들의 이름과 급여를 검색하라.

EMPLOYEE (EMPNO, EMPNAME, TITLE, MANAGER, SALARY, DNO)

DEPARTMENT (DEPTNO, DEPTNAME, FLOOR)

$\pi_{EMPNAME, SALARY} (\sigma_{DNO=2 \text{ OR } DNO=3} (EMPLOYEE))$

#### 예 : 실렉션, 프로젝션, 조인

질의: 개발 부서에서 근무하는 모든 직원들의 이름을 검색하라.

$\pi_{EMPNAME} (EMPLOYEE \bowtie_{DNO=DEPTNO} (\sigma_{DEPTNAME='개발'} (DEPARTMENT)))$

## 4.1 관계 대수(계속)

### □ 관계 대수의 한계

- ✓ 관계 대수는 산술 연산을 할 수 없음
- ✓ 집단 함수(aggregate function)를 지원하지 않음
- ✓ 정렬을 나타낼 수 없음
- ✓ 데이터베이스를 수정할 수 없음
- ✓ 프로젝션 연산의 결과에 중복된 튜플을 나타내는 것이 필요할 때가 있는데 이를 명시하지 못함

## 4.1 관계 대수(계속)

### □ 추가된 관계 대수 연산자

- ✓ 집단(aggregation) 함수: AVG, SUM, MIN, MAX, COUNT

예 : 집단 함수

질의: 모든 사원들의 급여의 평균이 얼마인가?

EMPLOYEE	EMPNO	...	SALARY	...
	2106	...	2500000	...
	3426	...	3000000	...
	3011	...	4000000	...
	1003	...	3000000	...
	3427	...	1500000	...
	1365	...	1500000	...
	4377	...	5000000	...

$AVG_{SALARY}(EMPLOYEE) \rightarrow 2,928,571$

## 4.1 관계 대수(계속)

### □ 추가된 관계 대수 연산자(계속)

- ✓ 그룹화: 각 그룹에 대해 집단 함수를 적용

예 : 그룹화

질의: 각 부서별 직원들의 급여의 평균이 얼마인가?

EMPLOYEE	EMPNO	...	SALARY	DNO
	3426	...	3000000	1
	1365	...	1500000	1
	2106	...	2500000	2
	1003	...	3000000	2
	4377	...	5000000	2
	3011	...	4000000	3
	3427	...	1500000	3

RESULT	DNO	AVG (SALARY)
	1	2250000
	2	3500000
	3	2750000

$\text{DNO} \rightarrow \text{AVG (SALARY) (EMPLOYEE)}$

## 4.1 관계 대수(계속)

### □ 추가된 관계 대수 연산자(계속)

#### ✓ 외부 조인

- 상대 릴레이션에서 대응되는 튜플을 갖지 못하는 튜플이나 조인 애트리뷰트에 널값이 들어 있는 튜플들을 다루기 위해서 조인 연산을 확장한 조인
- 두 릴레이션에서 대응되는 튜플들을 결합하면서, 대응되는 튜플을 갖지 않는 튜플과 조인 애트리뷰트에 널값을 갖는 튜플도 결과에 포함시킴
- **왼쪽 외부 조인**(left outer join), **오른쪽 외부 조인**(right outer join), **완전 외부 조인**(full outer join)

## 4.1 관계 대수(계속)

### □ 왼쪽 외부 조인 (형식: $R \bowtie S$ )

- ✓ 릴레이션 R과 S의 왼쪽 외부 조인 연산은 R의 모든 튜플들을 결과에 포함시키고, 만일 릴레이션 S에 관련된 튜플이 없으면 결과 릴레이션에서 릴레이션 S의 애트리뷰트들은 널값으로 채움

R과 S의 왼쪽 외부 조인

R	<table><tr><th>A</th><th>B</th><th>C</th></tr><tr><td>a1</td><td>b1</td><td>c1</td></tr><tr><td>a2</td><td>b2</td><td>c2</td></tr></table>	A	B	C	a1	b1	c1	a2	b2	c2	$R \bowtie S$	S	<table><tr><th>C</th><th>D</th><th>E</th></tr><tr><td>c1</td><td>d1</td><td>e1</td></tr><tr><td>c3</td><td>d2</td><td>e2</td></tr></table>	C	D	E	c1	d1	e1	c3	d2	e2	➡	RESULT	<table><tr><th>A</th><th>B</th><th>C</th><th>D</th><th>E</th></tr><tr><td>a1</td><td>b1</td><td>c1</td><td>d1</td><td>e1</td></tr><tr><td>a2</td><td>b2</td><td>c2</td><td>^</td><td>^</td></tr></table>	A	B	C	D	E	a1	b1	c1	d1	e1	a2	b2	c2	^	^
A	B	C																																						
a1	b1	c1																																						
a2	b2	c2																																						
C	D	E																																						
c1	d1	e1																																						
c3	d2	e2																																						
A	B	C	D	E																																				
a1	b1	c1	d1	e1																																				
a2	b2	c2	^	^																																				

## 4.1 관계 대수(계속)

### □ 오른쪽 외부 조인 (형식: $R \bowtie S$ )

- ✓ 릴레이션 R와 S의 오른쪽 외부 조인 연산은 S의 모든 튜플들을 결과에 포함시키고, 만일 릴레이션 R에 관련된 튜플이 없으면 결과 릴레이션에서 릴레이션 R의 애트리뷰트들은 널값으로 채움

예 : R과 S의 오른쪽 외부 조인

R	<table><tr><th>A</th><th>B</th><th>C</th></tr><tr><td>a1</td><td>b1</td><td>c1</td></tr><tr><td>a2</td><td>b2</td><td>c2</td></tr></table>	A	B	C	a1	b1	c1	a2	b2	c2	$R \bowtie S$	S	<table><tr><th>C</th><th>D</th><th>E</th></tr><tr><td>c1</td><td>d1</td><td>e1</td></tr><tr><td>c3</td><td>d2</td><td>e2</td></tr></table>	C	D	E	c1	d1	e1	c3	d2	e2	→	RESULT	<table><tr><th>A</th><th>B</th><th>C</th><th>D</th><th>E</th></tr><tr><td>a1</td><td>b1</td><td>c1</td><td>d1</td><td>e1</td></tr><tr><td>∧</td><td>∧</td><td>c3</td><td>d2</td><td>e2</td></tr></table>	A	B	C	D	E	a1	b1	c1	d1	e1	∧	∧	c3	d2	e2
A	B	C																																						
a1	b1	c1																																						
a2	b2	c2																																						
C	D	E																																						
c1	d1	e1																																						
c3	d2	e2																																						
A	B	C	D	E																																				
a1	b1	c1	d1	e1																																				
∧	∧	c3	d2	e2																																				

## 4.1 관계 대수(계속)

### □ 완전 외부 조인 (형식: $R \bowtie S$ )

- ✓ 릴레이션 R와 S의 완전 외부 조인 연산은 R과 S의 모든 튜플들을 결과에 포함시키고, 만일 상대 릴레이션에 관련된 튜플이 없으면 결과 릴레이션에서 상대 릴레이션의 애트리뷰트들은 널값으로 채움

$$R \bowtie S = (R \bowtie S) \cup (R \bowtie S)$$

예 : 완전 외부 조인

R	<table><tr><th>A</th><th>B</th><th>C</th></tr><tr><td>a1</td><td>b1</td><td>c1</td></tr><tr><td>a2</td><td>b2</td><td>c2</td></tr></table>	A	B	C	a1	b1	c1	a2	b2	c2	$R \bowtie S$	S	<table><tr><th>C</th><th>D</th><th>E</th></tr><tr><td>c1</td><td>d1</td><td>e1</td></tr><tr><td>c3</td><td>d2</td><td>e2</td></tr></table>	C	D	E	c1	d1	e1	c3	d2	e2	➡	RESULT	<table><tr><th>A</th><th>B</th><th>C</th><th>D</th><th>E</th></tr><tr><td>a1</td><td>b1</td><td>c1</td><td>d1</td><td>e1</td></tr><tr><td>a2</td><td>b2</td><td>c2</td><td>∧</td><td>∧</td></tr><tr><td>∧</td><td>∧</td><td>c3</td><td>d2</td><td>e2</td></tr></table>	A	B	C	D	E	a1	b1	c1	d1	e1	a2	b2	c2	∧	∧	∧	∧	c3	d2	e2
A	B	C																																											
a1	b1	c1																																											
a2	b2	c2																																											
C	D	E																																											
c1	d1	e1																																											
c3	d2	e2																																											
A	B	C	D	E																																									
a1	b1	c1	d1	e1																																									
a2	b2	c2	∧	∧																																									
∧	∧	c3	d2	e2																																									



## Review Questions: 관계 대수

□ R을 S로 디비전한 결과는?

Q1.

R	A	B	C
	a1	b2	c3
	a3	b1	c1
	a2	b2	c2
	a1	b3	c3

S	B
	b2
	b3

Q2.

R	A	B	C	D
	a	b	c	d
	a	b	e	f
	b	c	e	f
	e	d	c	d
	e	d	e	f
	a	b	d	e

S	C	D
	c	d
	e	f

## Review Questions: 관계 대수

- ❑ 비디오 대여점에서 사용할 수 있는 아래와 같은 관계 데이터베이스 스키마를 보고 각 질의를 관계 대수식으로 표현하라

CUSTOMER(CUSTOMER\_ID, NAME, ADDRESS, PHONE)

VIDEO(VIDEO\_ID, TITLE, GENRE)

RESERVED(CUSTOMER\_ID, VIDEO\_ID, DATE)

- 제목이 ' 반지의 제왕 ' 인 비디오 테이프의 장르를 검색하라
- 예약된 비디오 테이프의 ID를 검색하라
- 예약되지 않은 비디오 테이프의 ID를 검색하라
- 예약된 모든 비디오 테이프의 제목을 검색하라
- 비디오 테이프를 예약한 고객의 이름을 검색하라
- 장르가 '액션 ' 인 비디오 테이프를 예약한 고객의 주소를 검색하라
- 장르가 '다큐멘터리'인 비디오 테이프를 모두 예약한 고객의 이름을 검색하라