

5장. 관계 대수와 관계 해석

1 관계 대수

● 관계 대수

- 원하는 데이터를 얻기 위해서 어떻게(how) 질의를 수행할 것인지 일련의 연산을 순서대로 명시해야 하는 절차적 언어

● 관계 해석

- 원하는 데이터가 무엇인지만(what)을 명시하고 질의를 어떻게 수행할 것인가는 명시하지 않는 비절차적 언어

1 관계 대수

● 대수

- 수 대신 문자를 사용해서 문제를 쉽게 하고, 수학적인 법칙을 간단하고 명확하게 표현하는 것

● 관계 대수

- 데이터베이스에 저장된 데이터를 문자와 사용한 연산을 통해 요청한 데이터를 정보화하여 얻을 수 있는데 그 원리가 산술 연산자와 유사

● 순수 관계 연산자

- SELECT, PROJECT, JOIN, DIVISION

● 일반 집합 연산자

- 합집합, 교집합, 차집합, 카티션 프로덕트(Cartesian Product)

Relational Algebra Overview

- Relational Algebra consists of several groups of operations
 - Unary Relational Operations
 - SELECT (symbol: σ (sigma))
 - PROJECT (symbol: π (pi))
 - RENAME (symbol: ρ (rho))
 - Relational Algebra Operations From Set Theory
 - UNION (\cup), INTERSECTION (\cap), DIFFERENCE (or MINUS, $-$)
 - CARTESIAN PRODUCT (\times)
 - Binary Relational Operations
 - JOIN (several variations of JOIN exist)
 - DIVISION
 - Additional Relational Operations
 - OUTER JOINS, OUTER UNION
 - AGGREGATE FUNCTIONS (These compute summary of information: for example, SUM, COUNT, AVG, MIN, MAX)

- All examples discussed below refer to the COMPANY database shown here.

Figure 5.7

Referential integrity constraints displayed on the COMPANY relational database schema.

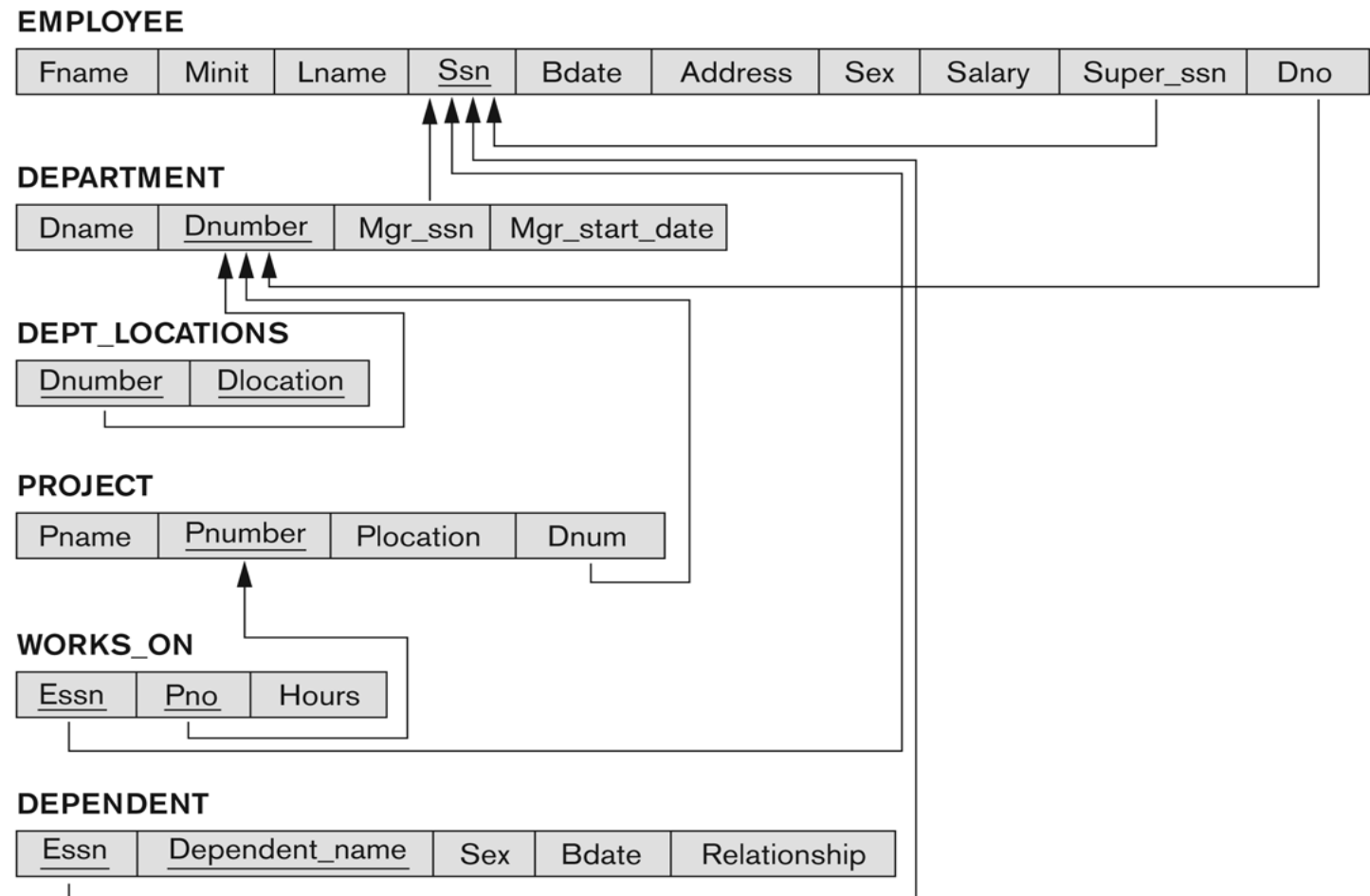


Figure 5.6

One possible database state for the COMPANY relational database schema.

EMPLOYEE

Fname	Minit	Lname	<u>Ssn</u>	Bdate	Address	Sex	Salary	Super_ssn	Dno
John	B	Smith	123456789	1965-01-09	731 Fondren, Houston, TX	M	30000	333445555	5
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5
Joyce	A	English	453453453	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5
Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E	Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000	NULL	1

DEPARTMENT

Dname	<u>Dnumber</u>	Mgr_ssn	Mgr_start_date
Research	5	333445555	1988-05-22
Administration	4	987654321	1995-01-01
Headquarters	1	888665555	1981-06-19

DEPT_LOCATIONS

<u>Dnumber</u>	<u>Dlocation</u>
1	Houston
4	Stafford
5	Bellaire
5	Sugarland
5	Houston

WORKS_ON

<u>Essn</u>	<u>Pno</u>	Hours
123456789	1	32.5
123456789	2	7.5
666884444	3	40.0
453453453	1	20.0
453453453	2	20.0
333445555	2	10.0
333445555	3	10.0
333445555	10	10.0
333445555	20	10.0
999887777	30	30.0
999887777	10	10.0
987987987	10	35.0
987987987	30	5.0
987654321	30	20.0
987654321	20	15.0
888665555	20	NULL

PROJECT

Pname	<u>Pnumber</u>	Plocation	Dnum
ProductX	1	Bellaire	5
ProductY	2	Sugarland	5
ProductZ	3	Houston	5
Computerization	10	Stafford	4
Reorganization	20	Houston	1
Newbenefits	30	Stafford	4

DEPENDENT

<u>Essn</u>	<u>Dependent_name</u>	Sex	Bdate	Relationship
333445555	Alice	F	1986-04-05	Daughter
333445555	Theodore	M	1983-10-25	Son
333445555	Joy	F	1958-05-03	Spouse
987654321	Abner	M	1942-02-28	Spouse
123456789	Michael	M	1988-01-04	Son
123456789	Alice	F	1988-12-30	Daughter
123456789	Elizabeth	F	1967-05-05	Spouse

1 관계 대수

1-1 순수 관계 연산자

● 선택(SELECT, σ), 선택연산

- 릴레이션에서 주어진 조건에 만족하는 튜플을 선택하는 연산자
- 표기 형식은 그리스 문자 시그마(sigma)를 사용한 연산자 기호인 σ 를 사용 (단, R은 릴레이션)

정의

$\sigma_{\langle \text{조건} \rangle} (R)$

- '학생' 릴레이션에서 점수 속성값이 80점 이상인 튜플을 선택하기 위한 학생 릴레이션에 대한 선택 연산을 하는 관계 대수 표현

예

$\sigma_{\text{점수} \geq 80} (\text{학생})$

1 관계 대수

1-1 순수 관계 연산자

- 학생 릴레이션에 대한 선택 연산

학생

학 번	이 름	학 년	전 공	점 수
01	오태호	3	컴퓨터	80
02	이재민	2	수학	85
03	노종문	1	수학	70
04	이영덕	1	수학	79

$\sigma_{\text{점수} \geq 80}(\text{학생})$

$\sigma_{\text{점수} \geq 80}(\text{학생})$

학 번	이 름	학 년	전 공	점 수
01	오태호	3	컴퓨터	80
02	이재민	2	수학	85

1-1 순수 관계 연산자

- SELECT σ is commutative:

$$\sigma_{\langle \text{cond1} \rangle} (\sigma_{\langle \text{cond2} \rangle} (\sigma_{\langle \text{cond3} \rangle} (R))) = \sigma_{\langle \text{cond2} \rangle} (\sigma_{\langle \text{cond3} \rangle} (\sigma_{\langle \text{cond1} \rangle} (R)))$$

- A cascade of SELECT operations may be replaced by a single selection with a conjunction of all the conditions:

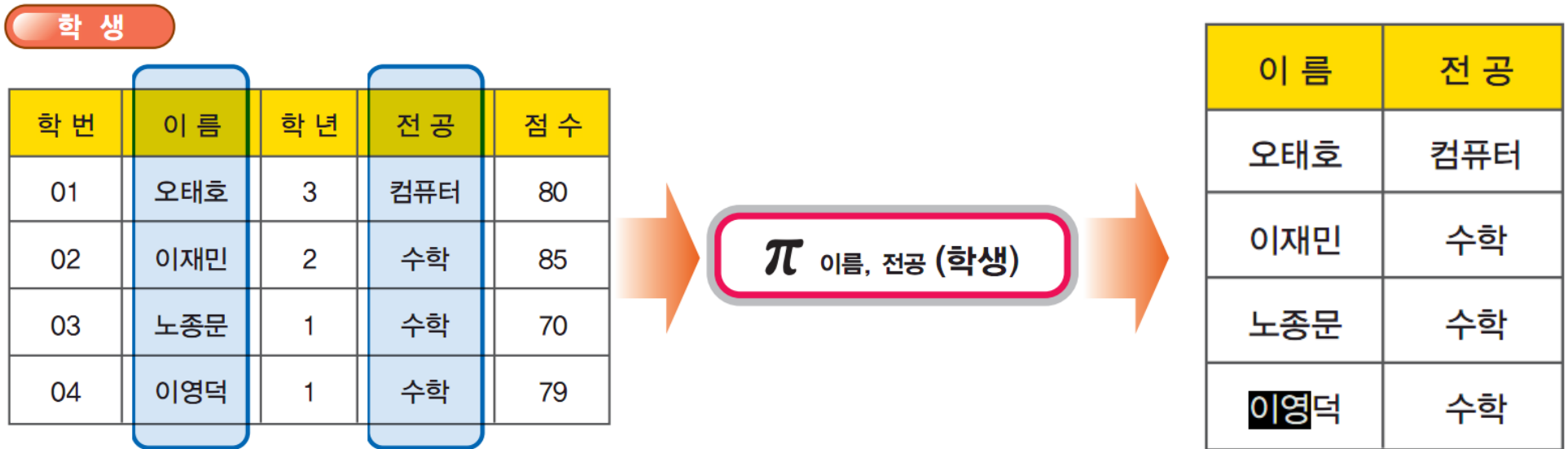
$$\sigma_{\langle \text{cond1} \rangle} (\sigma_{\langle \text{cond2} \rangle} (\sigma_{\langle \text{cond3} \rangle} (R))) = \sigma_{\langle \text{cond1} \rangle \text{ AND } \langle \text{cond2} \rangle \text{ AND } \langle \text{cond3} \rangle} (R)$$

- The number of tuples in the result of a SELECT is less than (or equal to) the number of tuples in the input relation R

1 관계 대수

1-1 순수 관계 연산자

● 학생 릴레이션에 대한 프로젝트 연산



110 page

1 관계대수

1-1 순수 관계 연산자

● 프로젝트(PROJECT, π), 프로젝트연산

Figure 6.1

Results of SELECT and PROJECT operations. (a) $\sigma_{(Dno=4 \text{ AND } Salary>25000) \text{ OR } (Dno=5 \text{ AND } Salary>30000)}(EMPLOYEE)$.
(b) $\pi_{Lname, Fname, Salary}(EMPLOYEE)$. (c) $\pi_{Sex, Salary}(EMPLOYEE)$.

(a)

Fname	Minit	Lname	<u>Ssn</u>	Bdate	Address	Sex	Salary	Super_ssn	Dno
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5

9 page

한 프로젝

(b)

Lname	Fname	Salary
Smith	John	30000
Wong	Franklin	40000
Zelaya	Alicia	25000
Wallace	Jennifer	43000
Narayan	Ramesh	38000
English	Joyce	25000
Jabbar	Ahmad	25000
Borg	James	55000

(c)

Sex	Salary
M	30000
M	40000
F	25000
F	43000
M	38000
M	25000
M	55000

10 page

- We may want to apply several relational algebra operations one after the other
- We can write a *single relational algebra expression* as follows:
 - $\pi_{\text{FNAME, LNAME, SALARY}}(\sigma_{\text{DNO}=5}(\text{EMPLOYEE}))$
- OR We can explicitly show the *sequence of operations*, giving a name to each intermediate relation:
 - $\text{DEP5_EMPS} \leftarrow \sigma_{\text{DNO}=5}(\text{EMPLOYEE})$
 - $\text{RESULT} \leftarrow \pi_{\text{FNAME, LNAME, SALARY}}(\text{DEP5_EMPS})$

Unary Relational Operations: RENAME

- The RENAME operator is denoted by ρ (rho)
- In some cases, we may want to *rename* the attributes of a relation or the relation name or both
 - Useful when a query requires multiple operations
 - Necessary in some cases (see JOIN operation later)

Unary Relational Operations: RENAME (contd.)

- The general RENAME operation ρ can be expressed by any of the following forms:
 - $\rho_{S(B_1, B_2, \dots, B_n)}(R)$ changes both:
 - the relation name to S , *and*
 - the column (attribute) names to B_1, B_1, \dots, B_n
 - $\rho_S(R)$ changes:
 - the *relation name* only to S
 - $\rho_{(B_1, B_2, \dots, B_n)}(R)$ changes:
 - the *column (attribute) names* only to B_1, B_1, \dots, B_n

Unary Relational Operations: RENAME (contd.)

- For convenience, we also use a *shorthand* for renaming attributes in an intermediate relation:
 - If we write:
 - $\text{RESULT} \leftarrow \pi_{\text{FNAME, LNAME, SALARY}} (\text{DEP5_EMPS})$
 - RESULT will have the *same attribute names* as DEP5_EMPS (same attributes as EMPLOYEE)
 - If we write:
 - $\text{RESULT (F, M, L, S, B, A, SX, SAL, SU, DNO)} \leftarrow \pi_{\text{FNAME, LNAME, SALARY}} (\text{DEP5_EMPS})$
 - The 10 attributes of DEP5_EMPS are *renamed* to F, M, L, S, B, A, SX, SAL, SU, DNO, respectively

1 관계 대수

1-1 순수 관계 연산자

● 조인(JOIN, \bowtie), 조인 연산

- 공통 속성을 중심으로 두 개의 릴레이션을 하나로 합쳐서 새로운 릴레이션
- 연산자의 종류에는 세타 조인(theta join), 동등 조인(equi join), 자연 조인(natural join), 외부 조인(outer join), 세미 조인(semi join)

● 세타 조인

- 두 릴레이션 $R(A_1, A_2, \dots, A_n)$ 과 $S(B_1, B_2, \dots, B_m)$ 의 세타 조인한 결과로 얻어지는 차수는 릴레이션 R 의 차수와 릴레이션 S 의 차수를 합한 것($n+m$)
- 어트리뷰트는 $(A_1, A_2, \dots, A_n, B_1, B_2, \dots, B_m)$ 이며, 조인 조건을 만족하는 튜플들로 이루어진 릴레이션 세타는 $\{ =, <>, <=, <, >=, > \}$ 중의 하나

1 관계 대수

1-1 순수 관계 연산자

- 동등 조인(equi join)

- 세타 조인 중에서 비교 연산자가 =인 조인이고 표기 형식은 그리스문자 \bowtie \triangleleft 를 사용

정의

$R \bowtie_{\text{키 속성 } r = \text{키 속성 } s} S$

- '학생' 릴레이션과 '성적' 릴레이션의 동등 조인을 위한 관계 대수 표현

예

학생 $\bowtie_{\text{학번} = \text{학번}}$ 성적

1 관계 대수

1-1 순수 관계 연산자

● 학생과 성적 릴레이션에 대한 동등 조인

학 생

학번 (STNO)	성명 (NAME)	학과 코드 (DNO)	학년 (YEAR)
9801	오태호	100	2
9802	이재민	200	3
9803	조남선	300	1
9804	노종문	100	3
9805	한바다	400	4

성 적

학번 (STNO)	과목 (COURSE)	성적 (SCORE)
9801	자료구조	90
9801	데이터베이스	80
9802	컴퓨터 구조	90
9803	자료 구조	80
9803	운영체제	90
9803	데이터베이스	90
9804	데이터베이스	90

학생 ⋈ 학번=학번 성적

성명 (NAME)	학과 코드 (DNO)	학년 (YEAR)	학생, 학번 (STNO)	성적, 학번 (STNO)	과목 (COURSE)	성적 (SCORE)
오태호	100	2	9801	9801	자료구조	90
오태호	100	2	9801	9801	데이터베이스	80
이재민	200	3	9802	9802	컴퓨터 구조	90
조남선	300	1	9803	9803	자료 구조	80
조남선	300	1	9803	9803	운영체제	90
조남선	300	1	9803	9803	데이터베이스	90
노종문	100	3	9804	9804	데이터베이스	90

1 관계 대수

1-1 순수 관계 연산자

● 자연 조인

- 동등 조인 결과로 얻어진 불필요한 중복되는 어트리뷰트를 한 개 제외한 조인

정의

$$R \bowtie_N S$$

·

정의

$$R * S$$

- '학생' 릴레이션과 '성적' 릴레이션의 자연 조인을 위한 관계 대수 표현

예

$$\text{학생} \bowtie_N \text{성적}$$

1 관계 대수

1-1 순수 관계 연산자

● 학생과 성적 릴레이션에 대한 자연 조인

학 생

학번 (STNO)	성명 (NAME)	학과 코드 (DNO)	학년 (YEAR)
9801	오태호	100	2
9802	이재민	200	3
9803	조남선	300	1
9804	노종문	100	3
9805	한바다	400	4

성 적

학번 (STNO)	과목 (COURSE)	성적 (SCORE)
9801	자료구조	90
9801	데이터베이스	80
9802	컴퓨터 구조	90
9803	자료 구조	80
9803	운영체제	90
9803	데이터베이스	90
9804	데이터베이스	90

학생 ⋈_N 성적

성명 (NAME)	학과 코드 (DNO)	학년 (YEAR)	학번 (STNO)	과목 (COURSE)	성적 (SCORE)
오태호	100	2	9801	자료구조	90
오태호	100	2	9801	데이터베이스	80
이재민	200	3	9802	컴퓨터 구조	90
조남선	300	1	9803	자료 구조	80
조남선	300	1	9803	운영체제	90
조남선	300	1	9803	데이터베이스	90
노종문	100	3	9804	데이터베이스	90

112 page

Example of NATURAL JOIN operation

(a)

PROJ_DEPT

Pname	<u>Pnumber</u>	Plocation	Dnum	Dname	Mgr_ssn	Mgr_start_date
ProductX	1	Bellaire	5	Research	333445555	1988-05-22
ProductY	2	Sugarland	5	Research	333445555	1988-05-22
ProductZ	3	Houston	5	Research	333445555	1988-05-22
Computerization	10	Stafford	4	Administration	987654321	1995-01-01
Reorganization	20	Houston	1	Headquarters	888665555	1981-06-19
Newbenefits	30	Stafford	4	Administration	987654321	1995-01-01

(b)

DEPT_LOCS

Dname	Dnumber	Mgr_ssn	Mgr_start_date	Location
Headquarters	1	888665555	1981-06-19	Houston
Administration	4	987654321	1995-01-01	Stafford
Research	5	333445555	1988-05-22	Bellaire
Research	5	333445555	1988-05-22	Sugarland
Research	5	333445555	1988-05-22	Houston

Figure 6.7

Results of two NATURAL JOIN operations.

(a) PROJ_DEPT \leftarrow PROJECT * DEPT.

(b) DEPT_LOCS \leftarrow DEPARTMENT * DEPT_LOCATIONS.

1 관계 대수

1-1 순수 관계 연산자

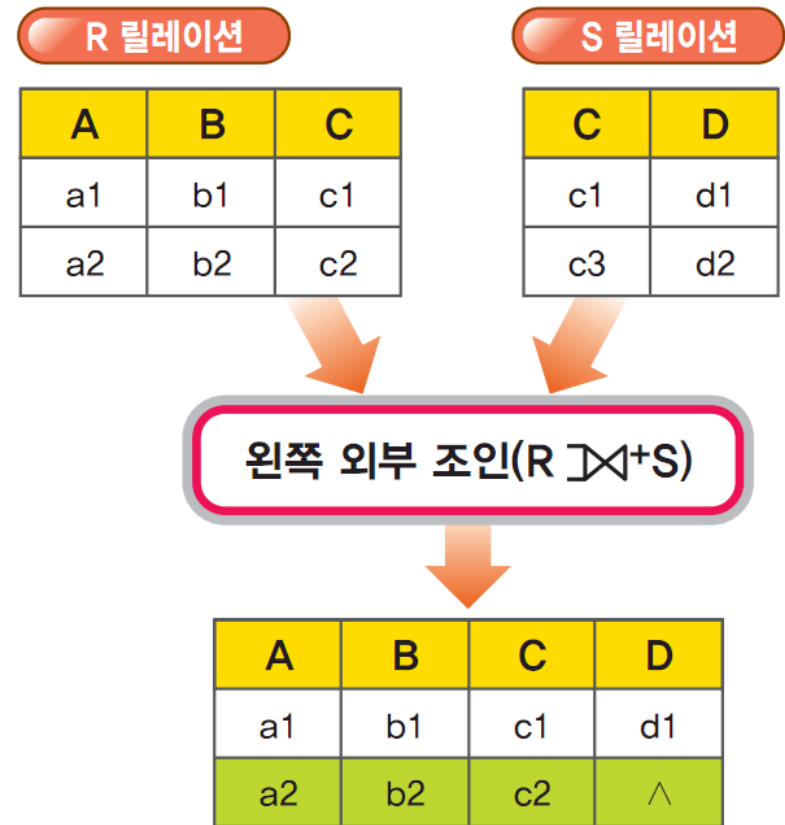
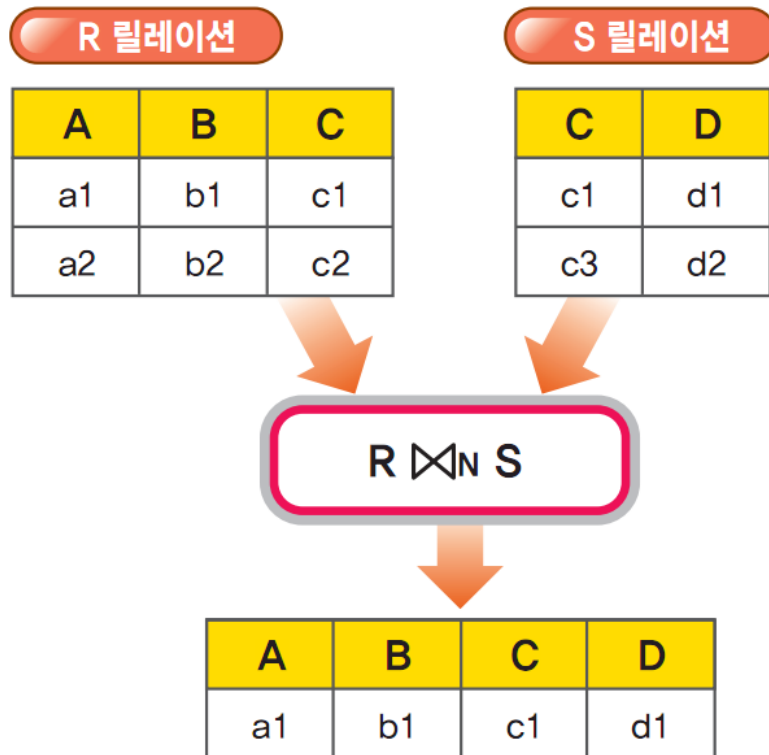
● 외부 조인

- 상대 릴레이션에서 대응되는 튜플을 갖지 못하는 튜플이나 조인 어트리뷰트에 널 값이 들어 있는 튜플들을 다루기 위해서 조인 연산을 확장한 조인
- 외부 조인의 종류로는 왼쪽 외부 조인(left outer join), 오른쪽 외부 조인(right outer join), 완전 외부 조인(full outer join)

1 관계 대수

1-1 순수 관계 연산자

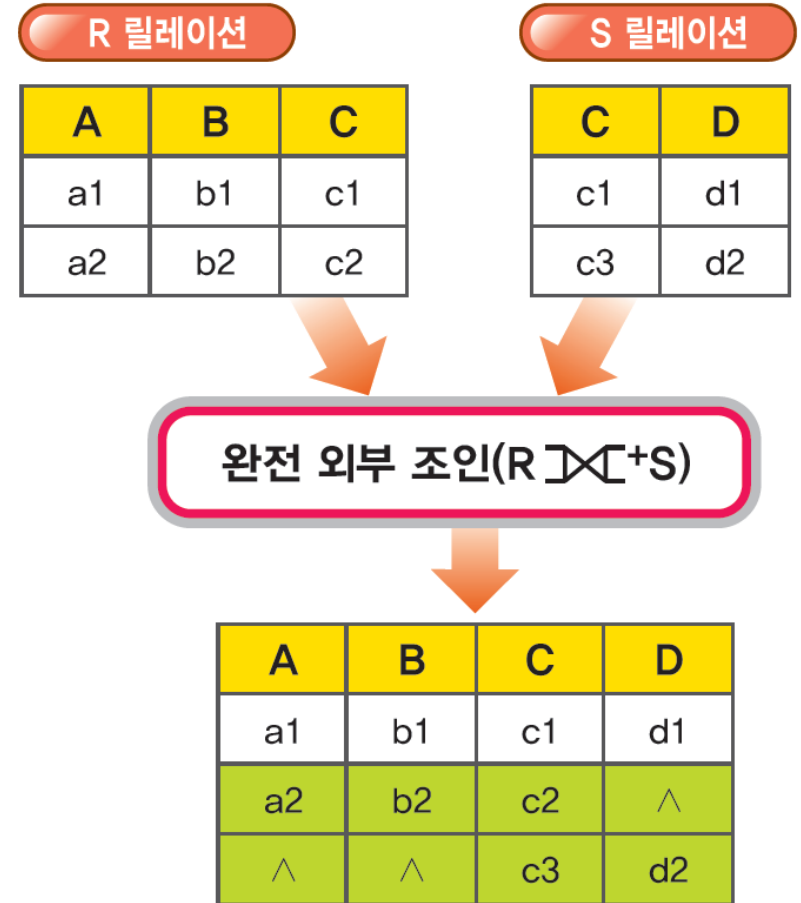
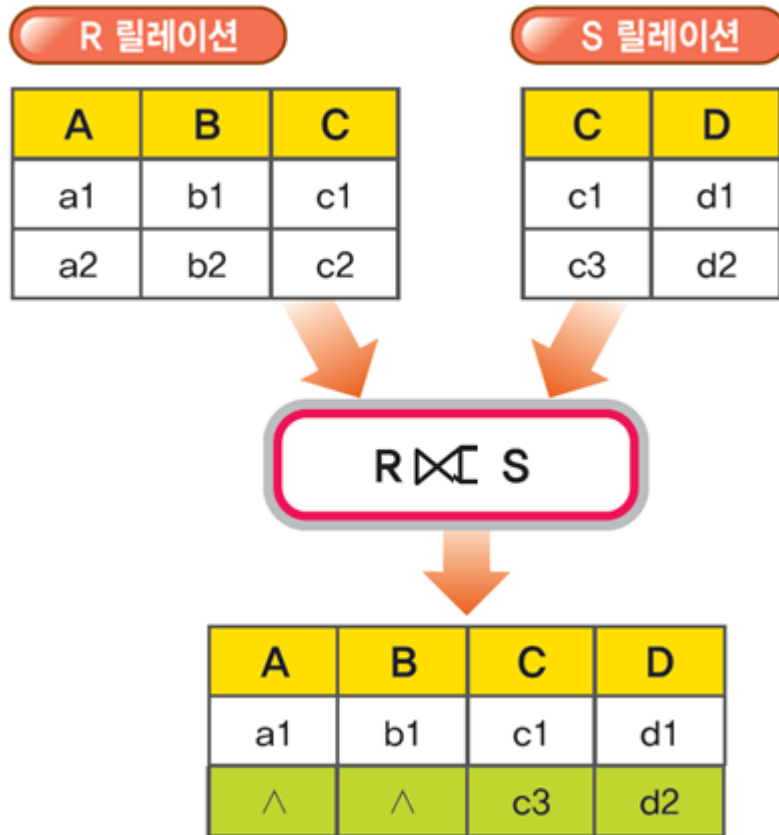
- 왼쪽 외부 조인



1 관계 대수

1-1 순수 관계 연산자

- 오른쪽 외부 조인



1 관계 대수

1-1 순수 관계 연산자

- 디비전(DIVISION, \div), 디버전연산

- $X \supset Y$ 인 2개의 릴레이션에서 $R(X)$ 와 $S(Y)$ 가 있을 때, R 의 속성이 S 의 속성값을 모두 가진 튜플에서 S 가 가진 속성을 제외한 속성만을 구하는 연산

정의

$R \text{ [속성 } r \div \text{속성 } s] S$

1 관계 대수

1-1 순수 관계 연산자

- 디비전

R	S1	S2	S3																											
<table><tr><th>A</th><th>B</th></tr><tr><td>a1</td><td>b1</td></tr><tr><td>a1</td><td>b2</td></tr><tr><td>a1</td><td>b3</td></tr><tr><td>a2</td><td>b1</td></tr><tr><td>a2</td><td>b3</td></tr><tr><td>a3</td><td>b1</td></tr><tr><td>a3</td><td>b2</td></tr><tr><td>a4</td><td>b2</td></tr></table>	A	B	a1	b1	a1	b2	a1	b3	a2	b1	a2	b3	a3	b1	a3	b2	a4	b2	<table><tr><th>B</th></tr><tr><td>b1</td></tr></table>	B	b1	<table><tr><th>B</th></tr><tr><td>b1</td></tr><tr><td>b2</td></tr></table>	B	b1	b2	<table><tr><th>B</th></tr><tr><td>b1</td></tr><tr><td>b2</td></tr><tr><td>b3</td></tr></table>	B	b1	b2	b3
A	B																													
a1	b1																													
a1	b2																													
a1	b3																													
a2	b1																													
a2	b3																													
a3	b1																													
a3	b2																													
a4	b2																													
B																														
b1																														
B																														
b1																														
b2																														
B																														
b1																														
b2																														
b3																														
	$R \div S1$	$R \div S2$	$R \div S3$																											
	<table><tr><th>A</th></tr><tr><td>a1</td></tr><tr><td>a2</td></tr><tr><td>a3</td></tr></table>	A	a1	a2	a3	<table><tr><th>A</th></tr><tr><td>a1</td></tr><tr><td>a3</td></tr></table>	A	a1	a3	<table><tr><th>A</th></tr><tr><td>a1</td></tr></table>	A	a1																		
A																														
a1																														
a2																														
a3																														
A																														
a1																														
a3																														
A																														
a1																														

Example of DIVISION

(a)				(b)	
SSN_PNOS		SMITH_PNOS		R	
Essn	Pno	Pno		A	B
123456789	1	1		a1	b1
123456789	2	2		a2	b1
666884444	3			a3	b1
453453453	1			a4	b1
453453453	2			a1	b2
333445555	2			a3	b2
333445555	3			a2	b3
333445555	10			a3	b3
333445555	20			a4	b3
999887777	30			a1	b4
999887777	10			a2	b4
987987987	10			a3	b4
987987987	30				
987654321	30				
987654321	20				
888665555	20				

SSNS		S		T	
Ssn		A		B	
123456789		a1		b1	
453453453		a2		b4	

Figure 6.8

The DIVISION operation. (a) Dividing SSN_PNOS by SMITH_PNOS. (b) $T \leftarrow R \div S$.

1 관계 대수

1-2 일반 집합 연산자

● 합집합(UNION, \cup)

- $R \cup S$ 는 R 또는 S에 있거나 R과 S 모두에 속한 튜플들로 이루어진 릴레이션
- Type compatible이 만족해야 한다.
 - R 과 S는 같은 차수의 속성과 같은 도메인을 가지고 있어야 한다.

정의

$$R \cup S = \{t | t \in R \vee t \in S\}$$

R, S에 속하는 모든 튜플의 집합

1 관계 대수

1-2 일반 집합 연산자

● 합집합

학생 1

성명 (NAME)	과목 (COURSE)	성적 (SCORE)
오태호	자료구조	90
이재민	컴퓨터 구조	90
조남선	자료 구조	80
노종문	데이터베이스	90

학생 2

성명 (NAME)	과목 (COURSE)	성적 (SCORE)
오태호	자료구조	90
이재민	컴퓨터 구조	90
조남선	자료 구조	80
조남선	운영체제	90

학생1 U 학생2

성명 (NAME)	과목 (COURSE)	성적 (SCORE)
오태호	자료구조	90
이재민	컴퓨터 구조	90
조남선	자료 구조	80
노종문	데이터베이스	90
조남선	운영체제	90

Relational Algebra Operations from Set Theory: UNION

- Example:
 - To retrieve the social security numbers of all employees who either *work in department 5* (RESULT1 below) or *directly supervise an employee who works in department 5* (RESULT2 below)
 - We can use the UNION operation as follows:
$$\begin{aligned}\text{DEP5_EMPS} &\leftarrow \sigma_{\text{DNO}=5}(\text{EMPLOYEE}) \\ \text{RESULT1} &\leftarrow \pi_{\text{SSN}}(\text{DEP5_EMPS}) \\ \text{RESULT2}(\text{SSN}) &\leftarrow \pi_{\text{SUPERSSN}}(\text{DEP5_EMPS}) \\ \text{RESULT} &\leftarrow \text{RESULT1} \cup \text{RESULT2}\end{aligned}$$
 - The union operation produces the tuples that are in either RESULT1 or RESULT2 or both

Example of the result of a UNION operation

- UNION Example

Figure 6.3

Result of the
UNION operation
 $\text{RESULT} \leftarrow \text{RESULT1} \cup \text{RESULT2}$.

RESULT1

Ssn
123456789
333445555
666884444
453453453

RESULT2

Ssn
333445555
888665555

RESULT

Ssn
123456789
333445555
666884444
453453453
888665555

1 관계 대수

1-2 일반 집합 연산자

● 교집합(INTERSECT, \cap)

- $R \cap S$ 는 R과 S 모두에 속한 튜플들로 이루어진 릴레이션
- Type compatible이 만족해야 한다.
 - R 과 S는 같은 차수의 속성과 같은 도메인을 가지고 있어야 한다.

정의

$R \cap S = \{t | t \in R \wedge t \in S\}$
R, S 양쪽 모두에 속하는 모든 튜플의 집합

117 page

1 관계 대수

1-2 일반 집합 연산자

● 교집합



1 관계 대수

1-2 일반 집합 연산자

● 차집합(DIFFERENCE, -)

- $R \setminus S$ 는 R 에는 속하지만 S 에는 속하지 않는 튜플들로 이루어진 릴레이션
- Type compatible이 만족해야 한다.
 - R 과 S 는 같은 차수의 속성과 같은 도메인을 가지고 있어야 한다.

정의

$$R \setminus S = \{t \mid t \in R \wedge t \notin S\}$$

R 에는 속하고, S 에는 속하지 않는 모든 튜플의 집합

118 page

1 관계대수

1-2 일반 집합 연산자

● 차집합

학생 1

성명 (NAME)	과목 (COURSE)	성적 (SCORE)
오태호	자료구조	90
이재민	컴퓨터 구조	90
조남선	자료 구조	80
노종문	데이터베이스	90

학생 2

성명 (NAME)	과목 (COURSE)	성적 (SCORE)
이재민	컴퓨터 구조	90
조남선	자료 구조	80

학생1 - 학생2

성명 (NAME)	과목 (COURSE)	성적 (SCORE)
오태호	자료구조	90
노종문	데이터베이스	90

118 page

Example to illustrate the result of UNION, INTERSECT, and DIFFERENCE

(a) STUDENT

Fn	Ln
Susan	Yao
Ramesh	Shah
Johnny	Kohler
Barbara	Jones
Amy	Ford
Jimmy	Wang
Ernest	Gilbert

INSTRUCTOR

Fname	Lname
John	Smith
Ricardo	Browne
Susan	Yao
Francis	Johnson
Ramesh	Shah

(b)

Fn	Ln
Susan	Yao
Ramesh	Shah
Johnny	Kohler
Barbara	Jones
Amy	Ford
Jimmy	Wang
Ernest	Gilbert
John	Smith
Ricardo	Browne
Francis	Johnson

(c)

Fn	Ln
Susan	Yao
Ramesh	Shah

(d)

Fn	Ln
Johnny	Kohler
Barbara	Jones
Amy	Ford
Jimmy	Wang
Ernest	Gilbert

(e)

Fname	Lname
John	Smith
Ricardo	Browne
Francis	Johnson

Figure 6.4

The set operations UNION, INTERSECTION, and MINUS. (a) Two union-compatible relations. (b) $\text{STUDENT} \cup \text{INSTRUCTOR}$. (c) $\text{STUDENT} \cap \text{INSTRUCTOR}$. (d) $\text{STUDENT} - \text{INSTRUCTOR}$. (e) $\text{INSTRUCTOR} - \text{STUDENT}$.

Some properties of UNION, INTERSECT, and DIFFERENCE

- Notice that both union and intersection are *commutative* operations; that is
 - $R \cup S = S \cup R$, and $R \cap S = S \cap R$
- Both union and intersection can be treated as n-ary operations applicable to any number of relations as both are *associative* operations; that is
 - $R \cup (S \cup T) = (R \cup S) \cup T$
 - $(R \cap S) \cap T = R \cap (S \cap T)$
- The minus operation is not commutative; that is, in general
 - $R - S \neq S - R$

1 관계 대수

1-2 일반 집합 연산자

● 카티션 프로덕트 연산자(CARTESIAN PRODUCT, \times)

- $R \times S$ 는 차수가 $n+m$ 이고, 카디널리티가 $i*j$ 이고, 어트리뷰트가 $(A_1, A_2, \dots, A_n, B_1, B_2, \dots, B_m)$ 이며, R 과 S 의 튜플들의 모든 가능한 조합으로 이루어진 릴레이션

정의

$$R \times S = \{ r.s \mid r \in R \wedge s \in S \}$$

R 과 S 의 접속인 모든 튜플의 집합

118 page

1 관계 대수

1-2 일반 집합 연산자

- 카티션 프로덕트 연산자



Relational Algebra Operations from Set Theory:

CARTESIAN PRODUCT (cont.)

- Generally, CROSS PRODUCT is not a meaningful operation
 - Can become meaningful when followed by other operations
- Example (not meaningful):
 - $\text{FEMALE_EMPS} \leftarrow \sigma_{\text{SEX}='F'}(\text{EMPLOYEE})$
 - $\text{EMP_NAMES} \leftarrow \pi_{\text{FNAME, LNAME, SSN}}(\text{FEMALE_EMPS})$
 - $\text{EMP_DEPENDENTS} \leftarrow \text{EMP_NAMES} \times \text{DEPENDENT}$
- EMP_DEPENDENTS will contain every combination of EMP_NAMES and DEPENDENT
 - whether or not they are actually related

Relational Algebra Operations from Set Theory:

CARTESIAN PRODUCT (cont.)

- To keep only combinations where the DEPENDENT is related to the EMPLOYEE, we add a SELECT operation as follows
- Example (meaningful):
 - $\text{FEMALE_EMPS} \leftarrow \sigma_{\text{SEX}='F'}(\text{EMPLOYEE})$
 - $\text{EMP_NAMES} \leftarrow \pi_{\text{FNAME, LNAME, SSN}}(\text{FEMALE_EMPS})$
 - $\text{EMP_DEPENDENTS} \leftarrow \text{EMP_NAMES} \times \text{DEPENDENT}$
 - $\text{ACTUAL_DEPS} \leftarrow \sigma_{\text{SSN}=\text{ESSN}}(\text{EMP_DEPENDENTS})$
 - $\text{RESULT} \leftarrow \pi_{\text{FNAME, LNAME, DEPENDENT_NAME}}(\text{ACTUAL_DEPS})$
- RESULT will now contain the name of female employees and their dependents

Figure 6.5

The CARTESIAN PRODUCT (CROSS PRODUCT) operation.

FEMALE_EMPS

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno
Alicia	J	Zelaya	999887777	1968-07-19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
Joyce	A	English	453453453	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5

EMPNAMES

Fname	Lname	Ssn
Alicia	Zelaya	999887777
Jennifer	Wallace	987654321
Joyce	English	453453453

EMP_DEPENDENTS

Fname	Lname	Ssn	Essn	Dependent_name	Sex	Bdate	...
Alicia	Zelaya	999887777	333445555	Alice	F	1986-04-05	...
Alicia	Zelaya	999887777	333445555	Theodore	M	1983-10-25	...
Alicia	Zelaya	999887777	333445555	Joy	F	1958-05-03	...
Alicia	Zelaya	999887777	987654321	Abner	M	1942-02-28	...
Alicia	Zelaya	999887777	123456789	Michael	M	1988-01-04	...
Alicia	Zelaya	999887777	123456789	Alice	F	1988-12-30	...
Alicia	Zelaya	999887777	123456789	Elizabeth	F	1967-05-05	...
Jennifer	Wallace	987654321	333445555	Alice	F	1986-04-05	...
Jennifer	Wallace	987654321	333445555	Theodore	M	1983-10-25	...
Jennifer	Wallace	987654321	333445555	Joy	F	1958-05-03	...
Jennifer	Wallace	987654321	987654321	Abner	M	1942-02-28	...
Jennifer	Wallace	987654321	123456789	Michael	M	1988-01-04	...
Jennifer	Wallace	987654321	123456789	Alice	F	1988-12-30	...
Jennifer	Wallace	987654321	123456789	Elizabeth	F	1967-05-05	...
Joyce	English	453453453	333445555	Alice	F	1986-04-05	...
Joyce	English	453453453	333445555	Theodore	M	1983-10-25	...
Joyce	English	453453453	333445555	Joy	F	1958-05-03	...
Joyce	English	453453453	987654321	Abner	M	1942-02-28	...
Joyce	English	453453453	123456789	Michael	M	1988-01-04	...
Joyce	English	453453453	123456789	Alice	F	1988-12-30	...
Joyce	English	453453453	123456789	Elizabeth	F	1967-05-05	...

ACTUAL_DEPENDENTS

Fname	Lname	Ssn	Essn	Dependent_name	Sex	Bdate	...
Jennifer	Wallace	987654321	987654321	Abner	M	1942-02-28	...

RESULT

Fname	Lname	Dependent_name
Jennifer	Wallace	Abner

1 관계 대수

1-2 일반 집합 연산자

● 외부 합집합

- U+는 완전하게 합병 가능하지 않은 두 릴레이션을 합집합으로 만드는 것



Additional Relational Operations: Aggregate Functions and Grouping

- A type of request that cannot be expressed in the basic relational algebra is to specify mathematical **aggregate functions** on collections of values from the database.
- Examples of such functions include retrieving the average or total salary of all employees or the total number of employee tuples.
 - These functions are used in simple statistical queries that summarize information from the database tuples.
- Common functions applied to collections of numeric values include
 - SUM, AVERAGE, MAXIMUM, and MINIMUM.
- The COUNT function is used for counting tuples or values.

Aggregate Function Operation

- Use of the Aggregate Functional operation G
 - $G_{\text{MAX}}(\text{Salary})$ (EMPLOYEE) retrieves the maximum salary value from the EMPLOYEE relation
 - $G_{\text{MIN}}(\text{Salary})$ (EMPLOYEE) retrieves the minimum Salary value from the EMPLOYEE relation
 - $G_{\text{SUM}}(\text{Salary})$ (EMPLOYEE) retrieves the sum of the Salary from the EMPLOYEE relation
 - $G_{\text{COUNT}}(\text{SSN}), \text{AVERAGE}(\text{Salary})$ (EMPLOYEE) computes the count (number) of employees and their average salary
 - Note: count just counts the number of rows, without removing duplicates

Using Grouping with Aggregation

- The previous examples all summarized one or more attributes for a set of tuples
 - Maximum Salary or Count (number of) Ssn
- Grouping can be combined with Aggregate Functions
- Example: For each department, retrieve the DNO, COUNT SSN, and AVERAGE SALARY
- A variation of aggregate operation G allows this:
 - Grouping attribute placed to left of symbol
 - Aggregate functions to right of symbol
 - $\text{DNO } G_{\text{COUNT SSN, AVERAGE Salary}} (\text{EMPLOYEE})$
- Above operation groups employees by DNO (department number) and computes the count of employees and average salary per department

Examples of applying aggregate functions and grouping

Figure 6.10

The aggregate function operation.

(a) $\rho_{R(Dno, No_of_employees, Average_sal)} (Dno \bowtie COUNT Ssn, AVERAGE Salary (EMPLOYEE))$.

(b) $Dno \bowtie COUNT Ssn, AVERAGE Salary (EMPLOYEE)$.

(c) $\bowtie COUNT Ssn, AVERAGE Salary (EMPLOYEE)$.

R

(a)

Dno	No_of_employees	Average_sal
5	4	33250
4	3	31000
1	1	55000

(b)

Dno	Count_ssn	Average_salary
5	4	33250
4	3	31000
1	1	55000

(c)

Count_ssn	Average_salary
8	35125

Illustrating aggregate functions and grouping

Figure 8.6

Results of GROUP BY and HAVING. (a) Q24. (b) Q26.

(a)

Fname	Minit	Lname	<u>Ssn</u>	...	Salary	Super_ssn	Dno
John	B	Smith	123456789		30000	333445555	5
Franklin	T	Wong	333445555		40000	888665555	5
Ramesh	K	Narayan	666884444		38000	333445555	5
Joyce	A	English	453453453	...	25000	333445555	5
Alicia	J	Zelaya	999887777		25000	987654321	4
Jennifer	S	Wallace	987654321		43000	888665555	4
Ahmad	V	Jabbar	987987987		25000	987654321	4
James	E	Bong	888665555		55000	NULL	1

Grouping EMPLOYEE tuples by the value of Dno

Dno	Count (*)	Avg (Salary)
5	4	33250
4	3	31000
1	1	55000

Result of Q24

Examples of Queries in Relational Algebra

- **Q1: Retrieve the name and address of all employees who work for the ‘Research’ department.**

$\text{RESEARCH_DEPT} \leftarrow \sigma_{\text{DNAME}='Research'}(\text{DEPARTMENT})$

$\text{RESEARCH_EMPS} \leftarrow (\text{RESEARCH_DEPT} \bowtie_{\text{DNUMBER}=\text{DNOEMPLOYEE}} \text{EMPLOYEE})$

$\text{RESULT} \leftarrow \pi_{\text{FNAME}, \text{LNAME}, \text{ADDRESS}}(\text{RESEARCH_EMPS})$

- **Q6: Retrieve the names of employees who have no dependents.**

$\text{ALL_EMPS} \leftarrow \pi_{\text{SSN}}(\text{EMPLOYEE})$

$\text{EMPS_WITH_DEPS}(\text{SSN}) \leftarrow \pi_{\text{ESSN}}(\text{DEPENDENT})$

$\text{EMPS_WITHOUT_DEPS} \leftarrow (\text{ALL_EMPS} - \text{EMPS_WITH_DEPS})$

$\text{RESULT} \leftarrow \pi_{\text{LNAME}, \text{FNAME}}(\text{EMPS_WITHOUT_DEPS} * \text{EMPLOYEE})$