

6장. 정규 관계형 질의 언어

이 동 호

데이터베이스 연구실
소프트웨어학부

목차

- 6.1 관계 대수
- 6.2 트플 관계 해석
- 6.3 도메일 관계 해석

6.1 관계 대수

- 절차적 언어
- 6개 기본 연산자 Six basic operators
 - 선택(select) 연산: σ
 - 추출(project) 연산: Π
 - 합집합(union) 연산: $\cup$
 - 차집합(difference) 연산: $-$
 - 카티션 곱(Cartesian product) 연산: $\times$
 - 재명명(rename) 연산: ρ
- 각 연산은 하나 혹은 두개의 릴레이션을 입력으로 받아 새로운 릴레이션을 출력함

선택 연산 - 예제

- 릴레이션 r

A	B	C	D
α	α	1	7
α	β	5	7
β	β	12	3
β	β	23	10

- $\sigma_{A=B \wedge D > 5}(r)$

A	B	C	D
α	α	1	7
β	β	23	10

① 선택 연산

- 표기 : $\sigma_p(r)$
- p : 선택 술어
- 정의 :

$$\sigma_p(r) = \{t \mid t \in r \text{ and } p(t)\}$$

선택 술어에는 여러 개의 술어들이 $\wedge$ (and), $\vee$ (or), $\neg$ (not) 등으로 연결될 수 있으며 각 술어들은 $=$, $\neq$, $>$, $\geq$, $<$, $\leq$ 등의 비교 연산자가 사용될 수 있음

- 선택 연산의 예:

$$\sigma_{dept_name="Physics"}(instructor)$$

추출 연산 - 예제

- 릴레이션 r :

A	B	C
α	10	1
α	20	1
β	30	1
β	40	2

- $\Pi_{A,C}(r)$

A	C
α	1
α	1
β	1
β	2

=

A	C
α	1
β	1
β	2

② 추출 연산

- 표기:

$$\Pi_{A_1, A_2, \dots, A_k}(r)$$

- A_1, A_2 : 속성 이름이고
- r : 릴레이션 이름

- 연산의 결과는 리스트된 k개의 속성으로 구성된 새로운 릴레이션
- 중복되는 행은 결과에서 제거됨
- 예제: instructor 릴레이션에서 *dept_name* 속성을 제거한 릴레이션

$$\Pi_{ID, name, salary}(instructor)$$

합집합 - 예제

- 릴레이션 r, s :

A	B
α	1
α	2
β	1

r

A	B
α	2
β	3

s

- $r \cup s$:

A	B
α	1
α	2
β	1
β	3

③ 합집합

- 표기: $r \cup s$
- 정의:

$$r \cup s = \{t \mid t \in r \text{ or } t \in s\}$$

- $r \cup s$ 가 유효하기 위해서,
 1. r, s 는 각 각 동일한 속성 개수를 갖아야 함
 2. 각 속성의 도메인은 서로 호환 가능해야 함 (예를 들어, r 의 두 번째 열의 속성은 s 의 두 번째 열의 속성과 같은 타입이어야 함)
- 예제 질의: Find all courses taught in the Fall 2009 semester, or in the Spring 2010 semester, or in both

$$\Pi_{course_id} (\sigma_{semester="Fall" \wedge year=2009} (section)) \cup \Pi_{course_id} (\sigma_{semester="Spring" \wedge year=2010} (section))$$

차집합

- 릴레이션 r, s :

A	B
α	1
α	2
β	1

r

A	B
α	2
β	3

s

- $r - s$:

A	B
α	1
β	1

④ 차집합

- 표기: $r - s$
- 정의:

$$r - s = \{t \mid t \in r \text{ and } t \notin s\}$$

- 차집합 연산은 호환 가능한 릴레이션에 대해서 가능
 - r 과 s 는 속성의 갯수가 동일해야 함
 - r 과 s 의 대응되는 속성들은 서로 호환 가능해야 함
- 예제: find all courses taught in the Fall 2009 semester, but not in the Spring 2010 semester

$$\Pi_{course_id} (\sigma_{semester="Fall" \wedge year=2009} (section)) - \Pi_{course_id} (\sigma_{semester="Spring" \wedge year=2010} (section))$$

카티션 곱 연산 - 예제

- 릴레이션 r, s :

A	B
α	1
β	2

r

C	D	E
α	10	a
β	10	a
β	20	b
γ	10	b

s

- $r \times s$:

A	B	C	D	E
α	1	α	10	a
α	1	β	10	a
α	1	β	20	b
α	1	γ	10	b
β	2	α	10	a
β	2	β	10	a
β	2	β	20	b
β	2	γ	10	b

⑤ 카티션 곱 연산

- 표기: $r \times s$
- 정의:

$$r \times s = \{t \mid q \mid t \in r \text{ and } q \in s\}$$

- $r(R)$ 과 $s(S)$ 의 속성이 서로 겹치지 않는다고 가정 (즉, $R \cap S = \emptyset$).
- 만약 $r(R)$ 과 $s(S)$ 가 서로 겹친다면 재명명이 사용되어야 함

연산들의 조합

- 다수의 연산들이 조합되어 사용될 수 있음
- 예제: $\sigma_{A=C}(r \times s)$

■ $r \times s$

A	B	C	D	E
α	1	α	10	a
α	1	β	10	a
α	1	β	20	b
α	1	γ	10	b
β	2	α	10	a
β	2	β	10	a
β	2	β	20	b
β	2	γ	10	b

■ $\sigma_{A=C}(r \times s)$

A	B	C	D	E
α	1	α	10	a
β	2	β	10	a
β	2	β	20	b

⑥ 재명명 연산

- 관계 대수 표현식에 이름을 붙이거나 이를 참조할 수 있음
- 하나 이상의 이름으로 릴레이션을 참조할 수 있음
- 예제:

$$\rho_x(E)$$

→ 표현식 E 를 X 라는 이름으로 재명명한다.

- 만약 관계 대수 표현식 E 가 n 개의 속성을 갖는다면

$$\rho_{x(A_1, A_2, \dots, A_n)}(E)$$

returns 표현식 E 의 결과는 X 로 재명명되고 n 개의 각 속성들은 A_1 , A_2 , ..., A_n 로 재명명 된다.

→ “표현식 E 가 n 개의 항으로 구성되었다면, E 의 결과를 x 로 재명명하고, $A_1, A_2, \dots, A_n$ 으로 재명명된 속성을 갖는다”는 의미

재명명 - 예제

Find the largest salary in the university

- Step 1: find instructor salaries that are less than some other instructor salary (i.e. not maximum)

- ▶ using a copy of *instructor* under a new name *d*

- ▶ $\Pi_{instructor.salary} (\sigma_{instructor.salary < d.salary} (instructor \times \rho_d(instructor)))$

가장 큰 salary보다
작은 것들만 return

- Step 2: Find the largest salary

- ▶ $\Pi_{salary}(instructor)$ –

- ▶ $\Pi_{instructor.salary} (\sigma_{instructor.salary < d.salary} (instructor \times \rho_d(instructor)))$

결국 최대
salary를
return 함

<i>instructor</i>	
name	salary
Lee	5000
Kim	10000
Park	7000

<i>d</i>	
name	salary
Lee	5000
Kim	10000
Park	7000

관계 대수의 6개 기본 연산 정리

- 관계 대수의 기본 표현은 데이터베이스 존재하는 릴레이션들로 구성됨
- E_1 과 E_2 를 관계 대수 표현식이라고 할때, 관계 대수는 아래의 6개 기본 연산으로 구성됨
 - $\sigma_p(E_1)$,
 - $\Pi_s(E_1)$,
 - $E_1 \cup E_2$,
 - $E_1 - E_2$,
 - $E_1 \times E_2$,
 - $\rho_x(E_1)$

추가적인 관계 대수 연산

- 6개의 기본 관계 대수 연산자를 이용하여 아래의 추가적인 관계 대수 연산을 표현할 수 있음
 - 교집합
 - 자연 조인
 - 배정(assignment)
 - 외부 조인

⑦ 교집합

- 표기: $r \cap s$
- 정의:

$$r \cap s = \{ t \mid t \in r \textbf{ and } t \in s \}$$

- 가정:
 - r, s 는 동일한 수의 속성을 갖는다.
 - r 과 s 의 속성은 서로 호환 가능하다.
- 참고: $r \cap s = r - (r - s)$

교집합 - 예제

- 릴레이션 r, s :

A	B
α	1
α	2
β	1

r

A	B
α	2
β	3

s

- $r \cap s$

A	B
α	2

⑧ 자연 조인

- 표기: $r \bowtie s$
- 정의:
 - r 과 s 를 각각 스키마 R 과 S 에서 정의된 릴레이션이라고 하자.
 - r 과 s 의 자연조인($r \bowtie s$)은 r 에 존재하는 튜플 t_r 과 s 에 존재하는 튜플 t_s 의 모든 쌍(pair)에 대해서 아래의 조건을 만족하는 튜플들의 집합이다.
 - t_r 과 t_s 가 $R \cap S$ 에 존재하는 각 속성에 대해서 동일한 값을 갖는 경우

- 예 제:

$R = (A, \underline{B}, C, \underline{D})$

$S = (E, \underline{B}, \underline{D})$

– Result schema = (A, B, C, D, E)

– $r \bowtie s$ is defined as:

$$\Pi_{r.A, r.B, r.C, r.D, s.E} (\sigma_{r.B = s.B \wedge r.D = s.D} (r \times s))$$

자연조인 - 예제

- 릴레이션 r, s :

A	B	C	D
α	1	α	a
β	2	γ	a
γ	4	β	b
α	1	γ	a
δ	2	β	b

r

B	D	E
1	a	α
3	a	β
1	a	γ
2	b	δ
3	b	ϵ

s

- $r \bowtie s$

A	B	C	D	E
α	1	α	a	α
α	1	α	a	γ
α	1	γ	a	α
α	1	γ	a	γ
δ	2	β	b	δ

자연 조인과 세타 조인

- Find the names of all instructors in the Comp. Sci. department together with the course titles of all the courses that the instructors teach
 - $\Pi_{name, title} (\sigma_{dept_name="Comp. Sci."} (instructor \bowtie teaches \bowtie course))$
- 자연 조인의 연결규칙
 - $(instructor \bowtie teaches) \bowtie course = instructor \bowtie (teaches \bowtie course)$
- 자연 조인의 교환규칙
 - $instructor \bowtie teaches = teaches \bowtie instructor$
- 세타 조인(theta join)
 - $r \bowtie_{\theta} s = \sigma_{\theta}(r \times s)$

[세타 조인]

- 세타 (θ) 는 $\{=, <>, <=, <, >=, >\}$ 중에서 하나가 될 수 있다.
- 세타조인 중에서 “=”인 경우를 동등조인이라 하고, 동등조인에서 중복된 속성을 하나 제거 한 것이 자연조인이다.
- 좀 더 일반적인 형태의 조인이라 할 수 있다.

⑨ 배정 연산

- 배정 연산($\leftarrow$)은 복잡한 질의를 간략히 표현하는 쉬운 방법을 제공함
 - 배정 연산을 활용하여 복잡한 질의의 결과를 임시 릴레이션 변수를 만들어 표현할 수 있음

⑩ 외부 조인

- 외부 조인은 정보 손실을 피하기 위한 조인의 한 방법
- 조인을 수행하고 서로 매칭되지 않아서 조인 결과에서 빠지게 되는 튜플들을 널 값을 이용하여 조인 결과에 추가함
- 참고: 널(null) 값
 - 널(null)은 값을 알 수 없다는 의미와 값이 존재하지 않는다는 두가지 의미를 갖을 수 있음
 - 널 값과 관련된 거의 모든 비교 연산은 거짓(**false**)임
 - We shall study precise meaning of comparisons with nulls later

외부 조인 - 예제

- 릴레이션 *instructor1*

<i>ID</i>	<i>name</i>	<i>dept_name</i>
10101	Srinivasan	Comp. Sci.
12121	Wu	Finance
15151	Mozart	Music

- 릴레이션 *teaches1*

<i>ID</i>	<i>course_id</i>
10101	CS-101
12121	FIN-201
76766	BIO-101

외부 조인 - 예제

- 자연 조인

instructor1 ⋈ *teaches1*

<i>ID</i>	<i>name</i>	<i>dept_name</i>	<i>course_id</i>
10101	Srinivasan	Comp. Sci.	CS-101
12121	Wu	Finance	FIN-201

- 왼쪽 외부 조인

instructor1 ⋈_L *teaches1*

<i>ID</i>	<i>name</i>	<i>dept_name</i>	<i>course_id</i>
10101	Srinivasan	Comp. Sci.	CS-101
12121	Wu	Finance	FIN-201
15151	Mozart	Music	<i>null</i>

외부 조인 - 예제(Cont.)

- 오른쪽 외부 조인

instructor ⋈_r *teaches*

<i>ID</i>	<i>name</i>	<i>dept_name</i>	<i>course_id</i>
10101	Srinivasan	Comp. Sci.	CS-101
12121	Wu	Finance	FIN-201
76766	null	null	BIO-101

- 전체 외부 조인

instructor ⋈_{all} *teaches*

<i>ID</i>	<i>name</i>	<i>dept_name</i>	<i>course_id</i>
10101	Srinivasan	Comp. Sci.	CS-101
12121	Wu	Finance	FIN-201
15151	Mozart	Music	<i>null</i>
76766	null	null	BIO-101

확장된 관계 대수 연산

- 일반화된 추출
- 집계 함수

⑪ 일반화된 추출

- 추출 연산은 추출 속성 리스트에 산술 연산자 사용 가능.

$$\Pi_{F_1, F_2, \dots, F_n}(E)$$

- 예제 :

$$\Pi_{ID, name, dept_name, salary/12}(instructor)$$

⑫ 집계 함수와 연산

- 집계 함수는 어떤 값들의 집합을 입력 받아 하나의 값을 출력

avg: 평균값

min: 최소값

max: 최대값

sum: 합계

count: 총개수

- 집계 연산의 관계 대수 표현

$$G_1, G_2, \dots, G_n \quad \mathcal{G}_{F_1(A_1), F_2(A_2), \dots, F_n(A_n)}(E)$$

E : 관계 대수 표현식

– $G_1, G_2, \dots, G_n$: 속성들의 그룹(생략 가능)

– F_i : 집계 함수

– A_i : 속성 이름

- 참고: 어떤 교재에서는 $\mathcal{G}$ (칼리그래픽 G) 대신 γ 를 사용함

집계 연산 - 예제

- 릴레이션 r

A	B	C
α	α	7
α	β	7
β	β	3
β	β	10

- $G_{\text{sum}(c)}(r)$

sum(c)
27

집계 연산 - 예제(Cont.)

- Find the average salary in each department

$\text{dept_name} \underset{\substack{\downarrow \\ \text{group by}}}{G} \text{avg}(\text{salary}) \text{ (instructor)}$
 group by Aggregate op.

ID	name	dept_name	salary
76766	Crick	Biology	72000
45565	Katz	Comp. Sci.	75000
10101	Srinivasan	Comp. Sci.	65000
83821	Brandt	Comp. Sci.	92000
98345	Kim	Elec. Eng.	80000
12121	Wu	Finance	90000
76543	Singh	Finance	80000
32343	El Said	History	60000
58583	Califieri	History	62000
15151	Mozart	Music	40000
33456	Gold	Physics	87000
22222	Einstein	Physics	95000

dept_name	avg_salary
Biology	72000
Comp. Sci.	77333
Elec. Eng.	80000
Finance	85000
History	61000
Music	40000
Physics	91000

집계 연산 - 예제(Cont.)

- 일반적으로 집계 연산의 결과는 이름이 존재하지 않음
 - 이름을 부여하기 위하여 재명명 연산 사용 가능

dept_name **G** *avg(salary) as avg_sal (instructor)*

SQL과 관계 대수

- **select** $A1, A2, \dots, An$
from $r1, r2, \dots, rm$
where P

$$\rightarrow \Pi_{A1, \dots, An} (\sigma_P (r1 \times r2 \times \dots \times rm))$$

- **select** $A1, A2, \text{sum}(A3)$
from $r1, r2, \dots, rm$
where P
group by $A1, A2$

$$\rightarrow_{A1, A2} \mathcal{G}_{\text{sum}(A3)} (\sigma_P (r1 \times r2 \times \dots \times rm))$$

6.2 튜플 관계 해석

- 비절차적 언어로 각 질의는 아래와 같이 표현됨

$$\{t \mid P(t)\}$$

→ 튜플 t 에 대해서 조건 P 를 만족하는 모든 튜플들의 집합

- t : 튜플 변수
- $t[A]$: 튜플 t 에서 속성 A 의 값
- $t \in r$
- P : 술어(조건)들의 집합

예제 질의

- Find the *ID*, *name*, *dept_name*, *salary* for instructors whose salary is greater than \$80,000

$$\{t \mid t \in instructor \wedge t[salary] > 80000\}$$

- 만약 질의는 동일한데 출력으로 속성 *ID* 만을 원할 때,

$$\{t \mid \exists s \in instructor (t[ID] = s[ID] \wedge s[salary] > 80000)\}$$

- 참고: $s(ID)$ 는 질의에서 정의된 묵시적 릴레이션

6.3 도메인 관계 해석

- 비절차적 언어로 튜플 관계 해석과 유사(질의 능력 동일)
- 각 질의는 아래와 같이 표현됨

$$\{ \langle x_1, x_2, \dots, x_n \rangle \mid P(x_1, x_2, \dots, x_n) \}$$

- $x_1, x_2, \dots, x_n$: 도메인 변수
- P : 술어 집합

예제 질의

- Find the *ID*, *name*, *dept_name*, *salary* for instructors whose salary is greater than \$80,000
 - $\{ \langle i, n, d, s \rangle \mid \langle i, n, d, s \rangle \in instructor \wedge s > 80000 \}$
- 질의를 위와 동일한데 속성 *ID* 값만 출력할 경우,
 - $\{ \langle i \rangle \mid \exists i, d, s (\langle i, n, d, s \rangle \in instructor \wedge s > 80000) \}$
- Find the names of all instructors whose department is in the Watson building
 - $\{ \langle n \rangle \mid \exists d, s (\langle i, n, d, s \rangle \in instructor \wedge \exists d, a (\langle d, b, a \rangle \in department \wedge b = \text{"Watson"})) \}$