

2장. 관계형 모델 소개

이 동 호

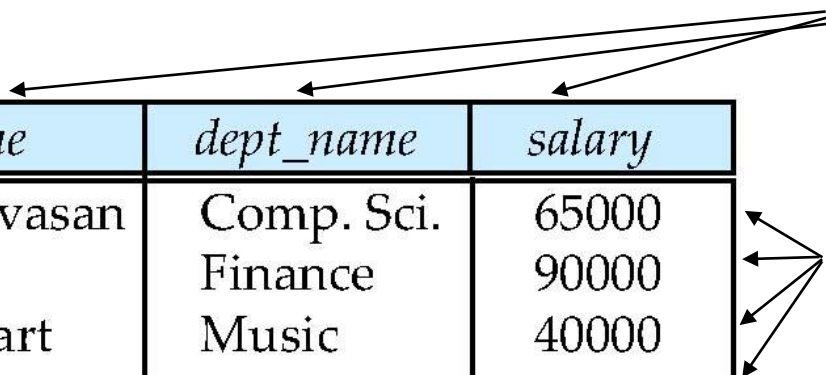
데이터베이스 연구실
소프트웨어 학부

목차

- 2.1 관계형 데이터베이스 구조
- 2.2 데이터베이스 스키마
- 2.3 키
- 2.4 스키마 다이어그램
- 2.5 관계형 질의언어
- 2.6 관계형 연산

2.1 관계형 데이터베이스 구조

- Instructor 릴레이션



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

속성(attributes)
(혹은 열)

튜플(tuples)
(혹은 행)

관계형 데이터베이스 구조 (Cont.)

- 릴레이션
 - 테이블을 의미함
- 튜플(혹은 레코드)
 - 테이블에서 하나의 행을 의미함
- 속성
 - 테이블에서 하나의 열을 의미함
 - 속성의 특징
 - 각 속성이 갖을 수 있는 값의 범위를 **도메인(domain)**이라 함
 - 속성값은 원자적(즉, 더 이상 쪼갤 수 없음)이어야 함
 - 널(null)값은 모든 도메인의 멤버
 - 널값은 많은 연산에서 복잡성을 일으킬 수 있음

2.2 데이터베이스 스키마

- 데이터베이스는 다수의 릴레이션들의 집합임
- 데이터베이스 스키마
 - 데이터베이스의 논리적 설계
- 데이터베이스 인스턴스
 - 어떤 순간 데이터베이스에 실제 데이터가 존재하는 것
- 릴레이션 스키마
 - 테이블의 논리적 설계
 - 즉, 속성들과 도메인들의 집합
- 릴레이션 인스턴스
 - 어떤 순간 테이블에 존재하는 실제 데이터가 존재하는 것

데이터베이스 스키마 (Cont.)

- 속성 : $A_1, A_2, \dots, A_n$
- 릴레이션 스키마 : $R = (A_1, A_2, \dots, A_n)$

예제:

instructor = (*ID*, *name*, *dept_name*, *salary*)

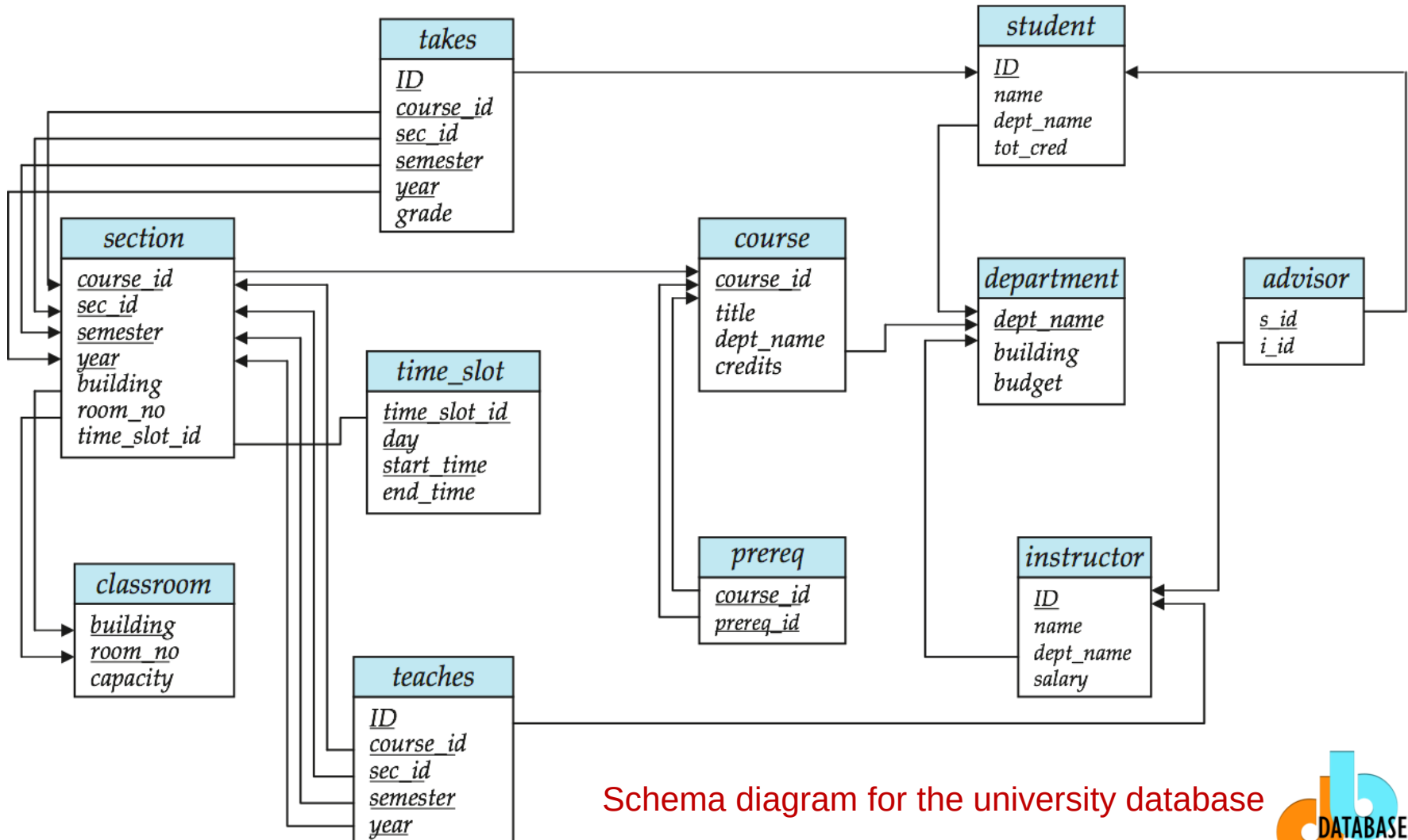
- 도메인 $D_1, D_2, \dots, D_n$ 주어졌을 경우, 어떤 릴레이션 r 은 $D_1 \times D_2 \times \dots \times D_n$ 의 부분집합이다.
 - 결국, 릴레이션은 n 개의 튜플($a_1, a_2, \dots, a_n$)들의 집합이다. (여기서 각각의 $a_i \in D_i$)
- 릴레이션의 인스턴스는 테이블로 명세됨
- 릴레이션 r 의 원소 t 는 튜플이며 테이블에서 행에 해당함

2.3 키(Keys)

- 키(K)는 속성들의 집합. 즉, $\text{키}(K) \subseteq \text{릴레이션}(R)$
- **수퍼키(superkey)**
 - 만약 K의 값들이 R의 모든 튜플들을 유일하게 구별하는데 충분하다면 K를 수퍼키라고 한다.
 - 예제: {ID}와 {ID, name}은 릴레이션 instructor의 주키
- **후보키(candidate key)**
 - 수퍼키의 부분집합이 수퍼키가 아닌 최소한의 수퍼키를 후보키라고 한다.
 - 예제: {ID}는 후보키임
 {ID, name}은 후보키 ?
 {name, dept_name}은 후보키 ?
- 후보키 중에 하나를 **주키(primary key)**로 선택할 수 있음
- **외래키(foreign key)** 제약조건: 어떤 릴레이션에 존재하는 값은 또 다른 릴레이션에 반드시 나와야 한다.

2.4 스키마 다이어그램

- 데이터베이스 스키마는 스키마 다이어그램으로 표시됨



Schema diagram for the university database

2.5 관계형 질의어

- 순수(pure) 질의어
 - 관계 대수(algebra)
 - 튜플 관계 해석(calculus)
 - 도메인 관계 해석
- 관계 연산자 (기호)
 - 선택 연산 (σ)
 - 추출 연산 (Π)
 - 합집합 ($\cup$)
 - 차집합 ($-$)
 - 교집합 ($\cap$)
 - **조인(join)** – 두 릴레이션을 합치는 연산
 - 카티션 곱(cartesian product) ($\times$)
 - 자연조인(natural join) ($\bowtie$)

2.6 관계 연산- 선택(selection) 연산

릴레이션 r

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

$A=B$ 그리고 $D > 5$ 인
조건을 만족하는 튜플

$\sigma_{A=B \text{ and } D > 5}(r)$

A	B	C	D
α	α	1	7
β	β	23	10

추출(projection) 연산

릴레이션 r

A	B	C
α	10	1
α	20	1
β	30	1
β	40	2

A 그리고 C 속성만
추출

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

A	C
α	1
α	1
β	1
β	2

=

A	C
α	1
β	1
β	2

합집합(union)

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

차집합(difference) 연산

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

교집합(intersection)

릴레이션 r, s

A	B
α	1
α	2
β	1

r

A	B
α	2
β	3

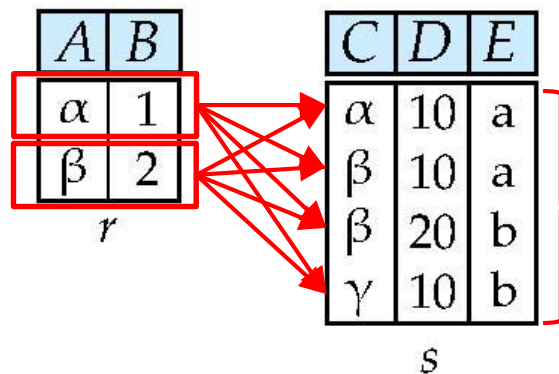
s

$r \cap s$:

A	B
α	2

카티션 곱(cartesian product)

릴레이션 r, 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

자연조인(natural join)

- 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

자연조인이란 릴레이션 R 과 S 에 공통적으로 존재하는 속성들을 이용하여 공통 속성들의 값들이 서로 같은 튜플들을 합치는 것이다.

자연조인 - 예제

릴레이션 r, s

A	B	C	D
α	<u>1</u>	α	<u>a</u>
β	2	γ	a
γ	4	β	b
α	1	γ	a
δ	2	β	b

r

B	D	E
<u>1</u>	<u>a</u>	α
<u>3</u>	<u>a</u>	β
<u>1</u>	<u>a</u>	γ
<u>2</u>	<u>b</u>	δ
<u>3</u>	<u>b</u>	ϵ

s

자연조인

$r \bowtie s$

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

관계 대수

Symbol (Name)	Example of Use
σ (Selection)	$\sigma_{\text{salary} \geq 85000}(\text{instructor})$
	Return rows of the input relation that satisfy the predicate.
Π (Projection)	$\Pi_{ID, salary}(\text{instructor})$
	Output specified attributes from all rows of the input relation. Remove duplicate tuples from the output.
$\bowtie$ (Natural Join)	$\text{instructor} \bowtie \text{department}$
	Output pairs of rows from the two input relations that have the same value on all attributes that have the same name.
$\times$ (Cartesian Product)	$\text{instructor} \times \text{department}$
	Output all pairs of rows from the two input relations (regardless of whether or not they have the same values on common attributes)
$\cup$ (Union)	$\Pi_{name}(\text{instructor}) \cup \Pi_{name}(\text{student})$
	Output the union of tuples from the two input relations.