



# Chapter 2: Intro to Relational Model

**Database System Concepts, 6<sup>th</sup> Ed.**

©Silberschatz, Korth and Sudarshan

See [www.db-book.com](http://www.db-book.com) for conditions on re-use



# Example of a Relation

속성 attributes  
(or columns)

<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

튜플  
tuples  
(or rows)



# 속성 타입 (attribute types)

- 속성 값으로 허용 되는 값의 집합을 속성의 도메인(**domain**)이라고 부른다.
- 속성 값은 일반적으로 원자값 (**atomic value**; that is, indivisible)의 성질을 만족하여야 한다.
- 널(null) 값은 모든 도메인에 포함된다고 가정한다. 그러나 널 값은 많은 연산 정의에 혼란을 야기 시킬 수 있다.



# 릴레이션 스키마와 인스턴스 (Relation Schema and Instance)

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

Example:

*instructor = (ID, name, dept\_name, salary)*

- 집합  $D_1, D_2, \dots, D_n$  이 주어지면 릴레이션  $r$  은  $D_1 \times D_2 \times \dots \times D_n$ 의 부분 집합이다.
- 따라서, 릴레이션은  $a_i \in D_i$ 인  $n$ -튜플( $a_1, a_2, \dots, a_n$ )의 집합이다.
- 릴레이션의 현재 값들 (**relation instance**) 은 테이블 형태로 표현된다.
- $r$ 의 원소  $t$ 는 테이블의 행(row)으로 표현된다.



# Relations are Unordered

- 튜플들의 순서에는 의미가 없다. 즉, 튜플은 임의의 순서로 저장될 수 있다.
- Example: *instructor* relation with unordered tuples

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



# Database

- 데이터베이스는 다수의 릴레이션으로 구성된다.
- 전체 DB 정보는 다음과 같이 부분으로 나뉘어진다

*instructor*

*student*

*advisor*

- 잘못된 설계 예:  
*univ (instructor -ID, name, dept\_name, salary, student\_id, ..)*

파급되는 결과

정보의 중복 기입 (e.g., two students have the same instructor)

널값 생성 (e.g., represent an student with no advisor)

- 정규화 이론 (Chapter 7)에서 좋은 릴레이션 스키마 설계법에 대하여 학습함.

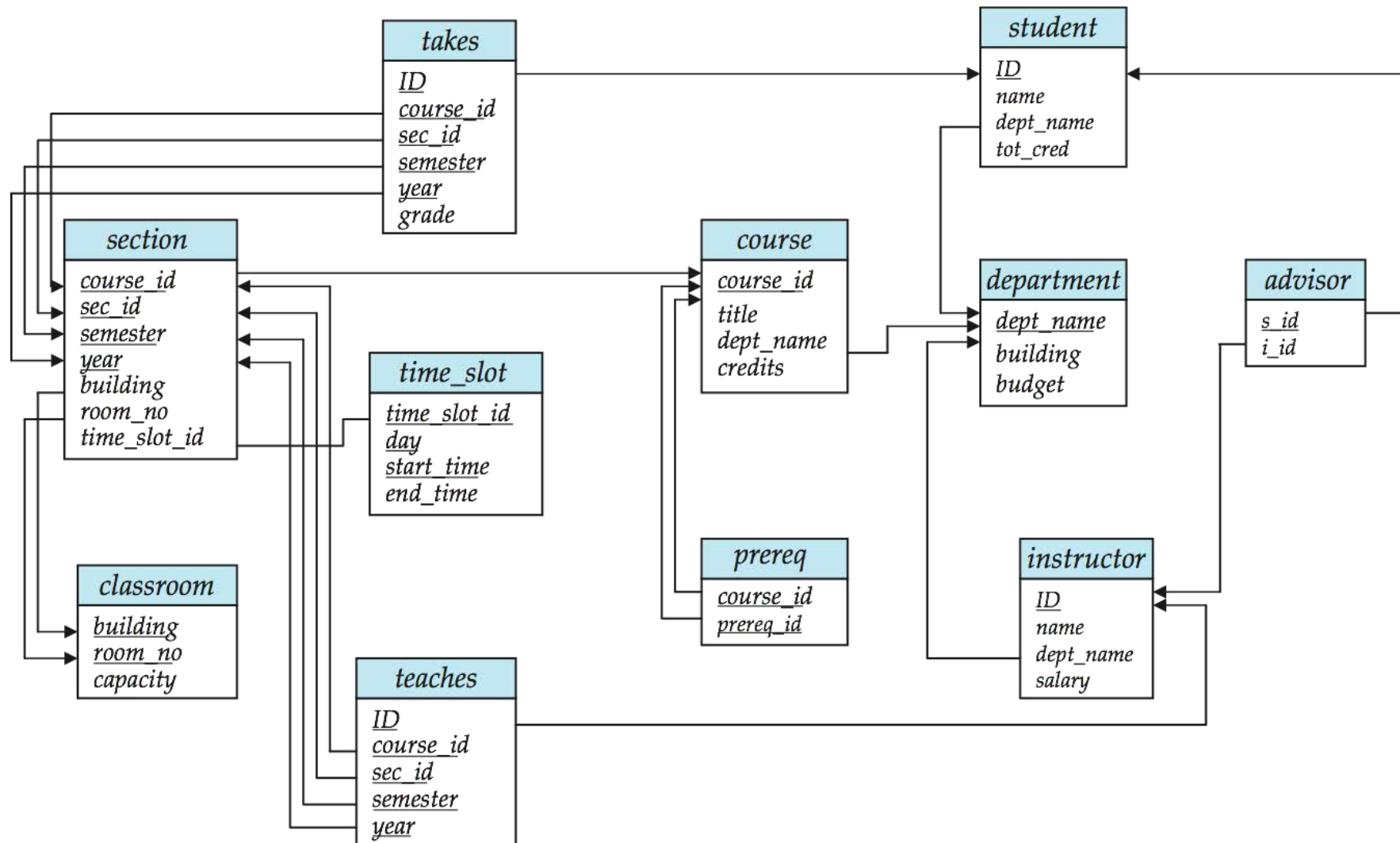


# 키 (Key)

- $K \subseteq R$  이라고 가정.
- K의 값이 각각의 가능한 릴레이션  $r(R)$ 의 고유한 튜플을 구분하는데 충분하다면 K는 R의 수퍼 키이다. “가능한 r”이란 모델링하고 있는 조직에 존재할 수 있는 릴레이션 r을 의미한다.
  - Example:  $\{ID\}$  and  $\{ID, name\}$  are both superkeys of *instructor*.
- 수퍼 키 K가 최소의 조건을 만족시키면 후보 키이다.
  - Example:  $\{ID\}$  is a candidate key for *Instructor*
- 후보 키 중 하나가 **주키 (primary key)**로 선택된다.
  - which one?
- 외래 키 제약조건 (Foreign key constraint): 하나의 릴레이션 내 속성 값이 다른 릴레이션에 존재하여야 한다.
  - Referencing relation
  - Referenced relation



# Schema Diagram for University Database







# 질의어 (Relational Query Languages)

- 사용자가 데이터베이스로부터 정보를 요청하는 언어
- 언어의 부류:
  - 절차식
  - 비절차식
- 순수언어:
  - 관계형 대수
  - 튜플 관계형 해석
  - 도메인 관계형 해석
- 순수 언어는 사람들이 사용하는 질의어의 기본을 이루고 있다.



# 관계형 대수 (Relational Algebra)

- 절차식 언어
- 6 가지 기본 연산자 (Six basic operators)
  - 선택 (select):  $\sigma$
  - 추출 (project):  $\Pi$
  - 합집합 (union):  $\cup$
  - 차집합 (set difference):  $-$
  - 카티션 곱 (Cartesian product):  $\times$
  - 재명명 (rename):  $\rho$
- 연산자는 입력으로서 하나 이상의 릴레이션을 취해 그 결과로 새로운 릴레이션을 생성한다.



# 선택연산 (Selection of tuples)

□ Relation r

A	B	C	D
$\alpha$	$\alpha$	1	7
$\alpha$	$\beta$	5	7
$\beta$	$\beta$	12	3
$\beta$	$\beta$	23	10

□ Select tuples with A=B  
and D > 5

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

A	B	C	D
$\alpha$	$\alpha$	1	7
$\beta$	$\beta$	23	10



# 추출 연산 (Selection of Columns)

□ Relation  $r$ :

A	B	C
$\alpha$	10	1
$\alpha$	20	1
$\beta$	30	1
$\beta$	40	2

□ Select A and C

□ Projection

□  $\Pi_{A, C}(r)$

A	C
$\alpha$	1
$\alpha$	1
$\beta$	1
$\beta$	2

=

A	C
$\alpha$	1
$\beta$	1
$\beta$	2



# 합집합 (Union of two relations)

□ Relations  $r, s$ :

$A$	$B$
$\alpha$	1
$\alpha$	2
$\beta$	1

$r$

$A$	$B$
$\alpha$	2
$\beta$	3

$s$

□  $r \cup s$ :

$A$	$B$
$\alpha$	1
$\alpha$	2
$\beta$	1
$\beta$	3



# 차집합 (Set difference of two relations)

□ Relations  $r, s$ :

$A$	$B$
$\alpha$	1
$\alpha$	2
$\beta$	1

$r$

$A$	$B$
$\alpha$	2
$\beta$	3

$s$

□  $r - s$ :

$A$	$B$
$\alpha$	1
$\beta$	1



# 두 릴레이션의 조인 - 카티션 곱

□ Relations  $r, s$ :

$A$	$B$
$\alpha$	1
$\beta$	2

$r$

$C$	$D$	$E$
$\alpha$	10	a
$\beta$	10	a
$\beta$	20	b
$\gamma$	10	b

$s$

□  $r \times s$ :

$A$	$B$	$C$	$D$	$E$
$\alpha$	1	$\alpha$	10	a
$\alpha$	1	$\beta$	10	a
$\alpha$	1	$\beta$	20	b
$\alpha$	1	$\gamma$	10	b
$\beta$	2	$\alpha$	10	a
$\beta$	2	$\beta$	10	a
$\beta$	2	$\beta$	20	b
$\beta$	2	$\gamma$	10	b



# Rename Operation

- 이름을 줄 수 있도록 하여 관계형 대수 표현식의 결과를 참조하도록 한다.
- 하나 이상의 이름으로 릴레이션을 참조하도록 한다.
- Example:

$$\rho_x(E)$$

이름  $x$ 라는 이름으로 표현식  $E$ 를 리턴한다.

- 관계형 대수 표현식  $E$ 가  $n$ 항이면,

$$\rho_x(A_1, A_2, \dots, A_n)(E)$$

$x$ 라는 이름으로  $A_1, A_2, \dots, A_n$ 으로 재명명된 애트리뷰트를 가진 표현식  $E$ 의 결과를 리턴한다.





# 복합 연산 (Composition of Operations)

- 여러 연산을 사용해 표현식을 만들 수 있다.
- Example:  $\sigma_{A=C}(r \times s)$

□  $r \times s$

A	B	C	D	E
$\alpha$	1	$\alpha$	10	a
$\alpha$	1	$\beta$	10	a
$\alpha$	1	$\beta$	20	b
$\alpha$	1	$\gamma$	10	b
$\beta$	2	$\alpha$	10	a
$\beta$	2	$\beta$	10	a
$\beta$	2	$\beta$	20	b
$\beta$	2	$\gamma$	10	b

□  $\sigma_{A=C}(r \times s)$

A	B	C	D	E
$\alpha$	1	$\alpha$	10	a
$\beta$	2	$\beta$	10	a
$\beta$	2	$\beta$	20	b



# 자연 조인 예제 (Natural Join Example)

□ Relations r, s:

A	B	C	D
$\alpha$	1	$\alpha$	a
$\beta$	2	$\gamma$	a
$\gamma$	4	$\beta$	b
$\alpha$	1	$\gamma$	a
$\delta$	2	$\beta$	b

r

B	D	E
1	a	$\alpha$
3	a	$\beta$
1	a	$\gamma$
2	b	$\delta$
3	b	$\epsilon$

s

□ Natural Join

□  $r \bowtie s$

A	B	C	D	E
$\alpha$	1	$\alpha$	a	$\alpha$
$\alpha$	1	$\alpha$	a	$\gamma$
$\alpha$	1	$\gamma$	a	$\alpha$
$\alpha$	1	$\gamma$	a	$\gamma$
$\delta$	2	$\beta$	b	$\delta$

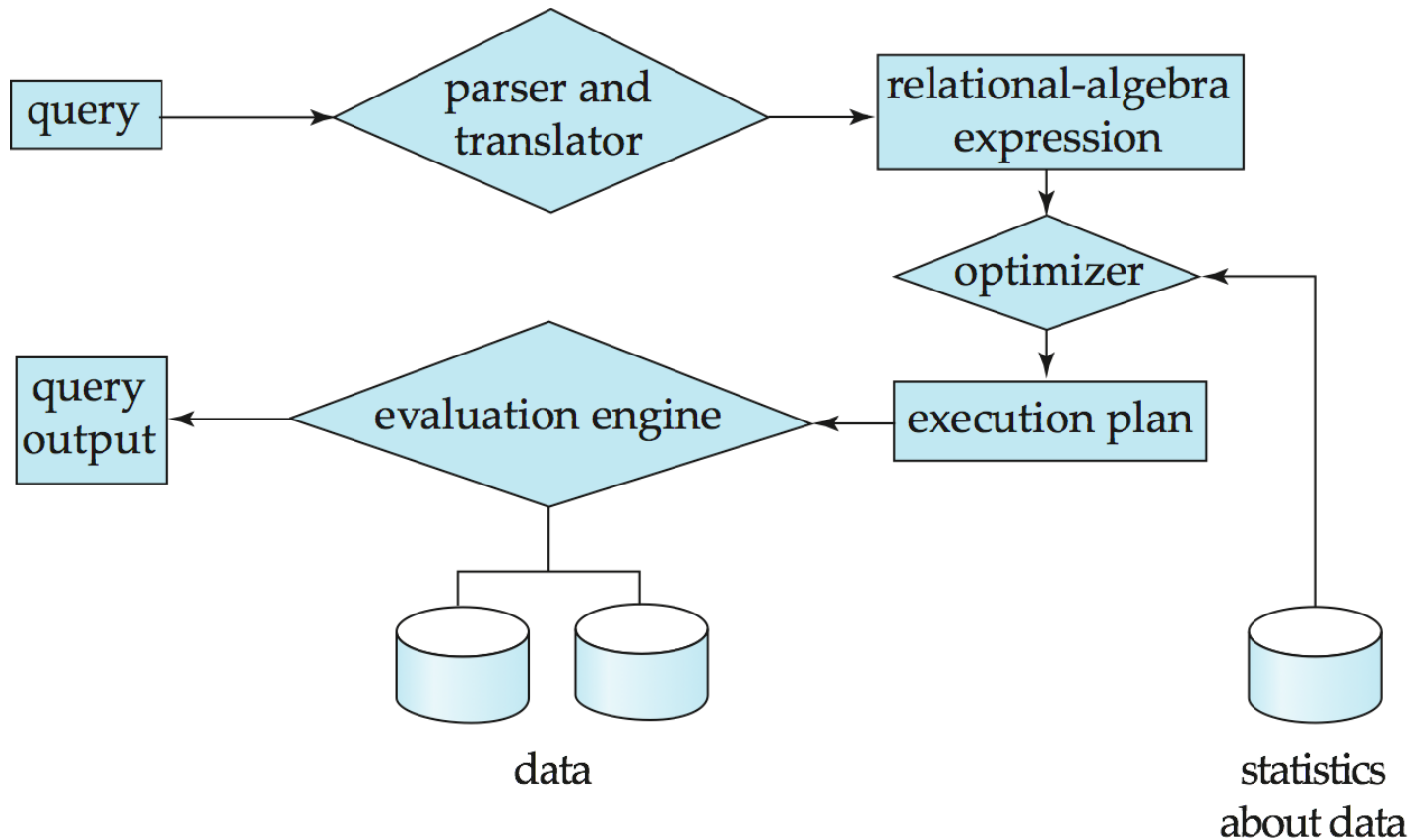
$r \bowtie s$  is defined as:

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



# 질의 처리 (Query Processing)

1. Parsing and translation
2. Optimization
3. Evaluation





# 예제 질의

- 물리학과 (Physics department)에 소속된 강사 ID와, 이 들 강사가 가르친 모든 *course\_id* 를 찾으시오.

- Query 1

$$\Pi_{instructor.ID, course\_id} (\sigma_{dept\_name="Physics"} (\sigma_{instructor.ID=teaches.ID} (instructor \times teaches)))$$

- Query 2

$$\Pi_{instructor.ID, course\_id} (\sigma_{instructor.ID=teaches.ID} (\sigma_{dept\_name="Physics"} (instructor \times teaches)))$$



# End of Chapter 2

**Database System Concepts, 6<sup>th</sup> Ed.**

©Silberschatz, Korth and Sudarshan

See [www.db-book.com](http://www.db-book.com) for conditions on re-use