CSCB20 – Week 2

Introduction to Database and Web Application Programming

Anna Bretscher

Winter 2017

This Week

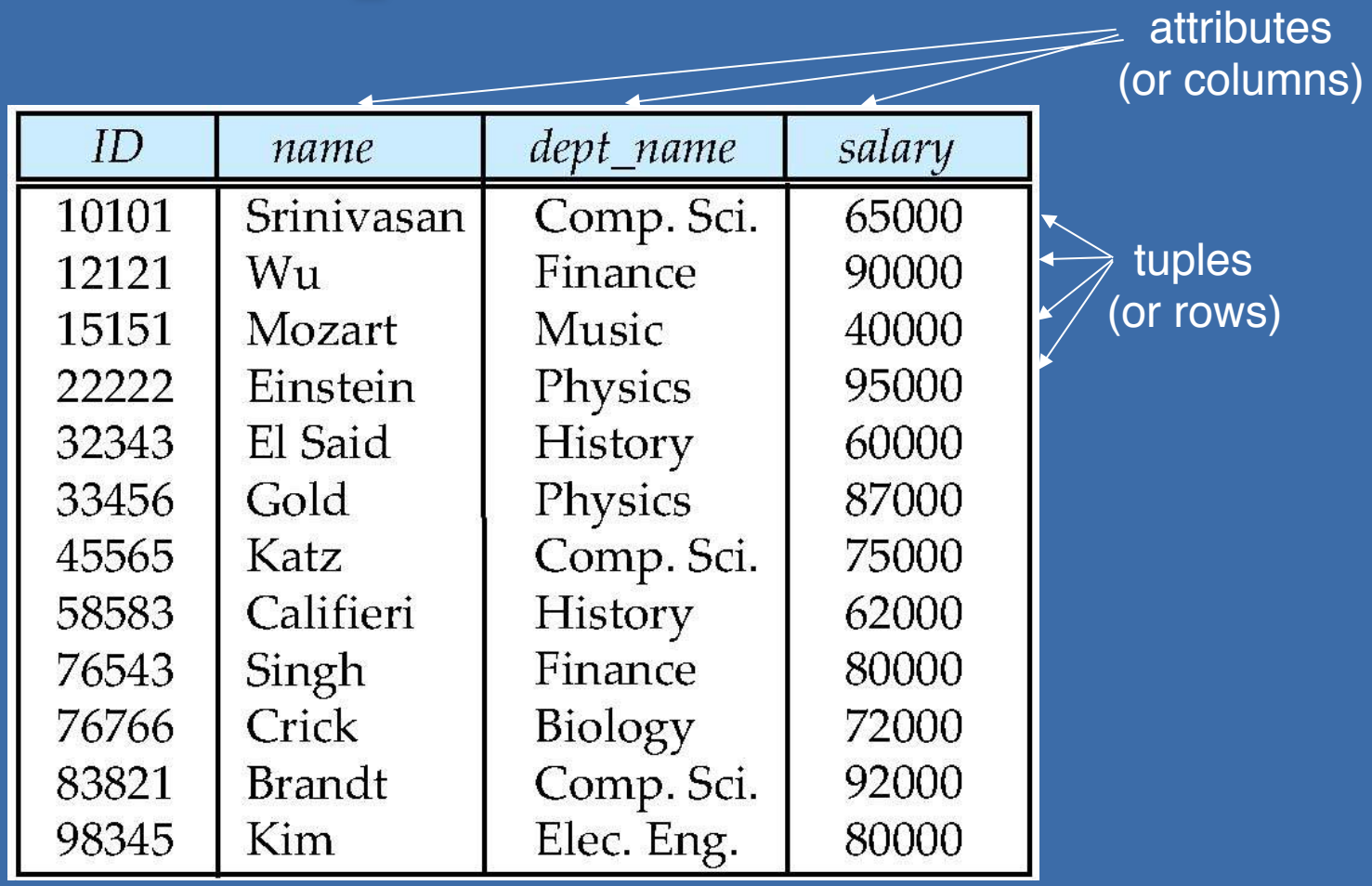
Quick Review of terminology

Relational Model Continued

- Relational diagrams
- Relational operations
- Relational algebra

Intro to SQL and MySQL (tentative)

Example of a Relation



ID	*name*	*dept_name*	*salary*
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

Relation Schema: instructor(ID, name, dept_name, salary)

Terminology

Q. What is a *superkey*?

A. A set of one or more attributes that *uniquely identify* a tuple in the relation.

Q. What is a *candidate* key?

A. A *minimal* super key.

Q. What is a *primary* key?

A. A candidate key chosen to distinguish between tuples.

Foreign Keys

A set of *attributes* in a relation (table) that is a *primary key* in *another relation*.

instructor(ID, name, dept_name, salary)

department(dept_name, building, budget)

teaches(ID, course_id, sec_id, semester, year)

The primary keys are underlined.

Q. What are the *foreign keys* for this set of relations?

A. *dept_name* in instructor

ID in teaches

Foreign Keys

A set of *attributes* in a relation (table) that is a *primary key* in *another relation*.

instructor(ID, name, dept_name, salary)
department(dept_name, building, budget)
teaches(ID, course_id, sec_id, semester, year)

The primary keys are underlined.

We say ID from teaches references instructor.

teaches is the referencing relation.
instructor is the referenced relation.

Basic Schema Constraints

Foreign Key Constraint

A foreign key value in one relation must appear in the referenced relation.

Example:

teaches(ID, course_id,sec_id, semester, year)

section(course_id, sec_id, semester, year, building, room_number, time_slot_id)

Q. What might be a foreign key constraint?

A. *course_id,sec_id, semester, year* in teaches has a foreign key constraint on section.

Basic Schema Constraints

Referential Integrity Constraint

Values appearing in specified attributes of any tuple in the referencing relation, appear in specified attributes of at least one tuple in the referenced relation.

Example:

teaches(ID, course_id,sec_id, semester, year)

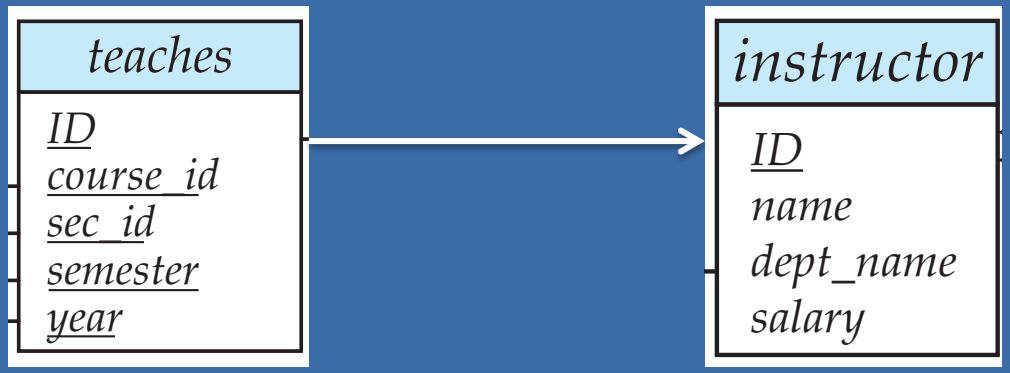
section(course_id, sec_id, semester, year, building, room_number, time_slot_id)

Q. What might be a referential integrity constraint here?

A. If a section with *(course_id, sec_id, semester, year)* exists we want there to be someone to teach it so there must be a teaches(ID, *course_id, sec_id, semester, year)*

Schema Diagrams

We can depict *foreign key constraints* and *primary keys* using a *schema diagram*.



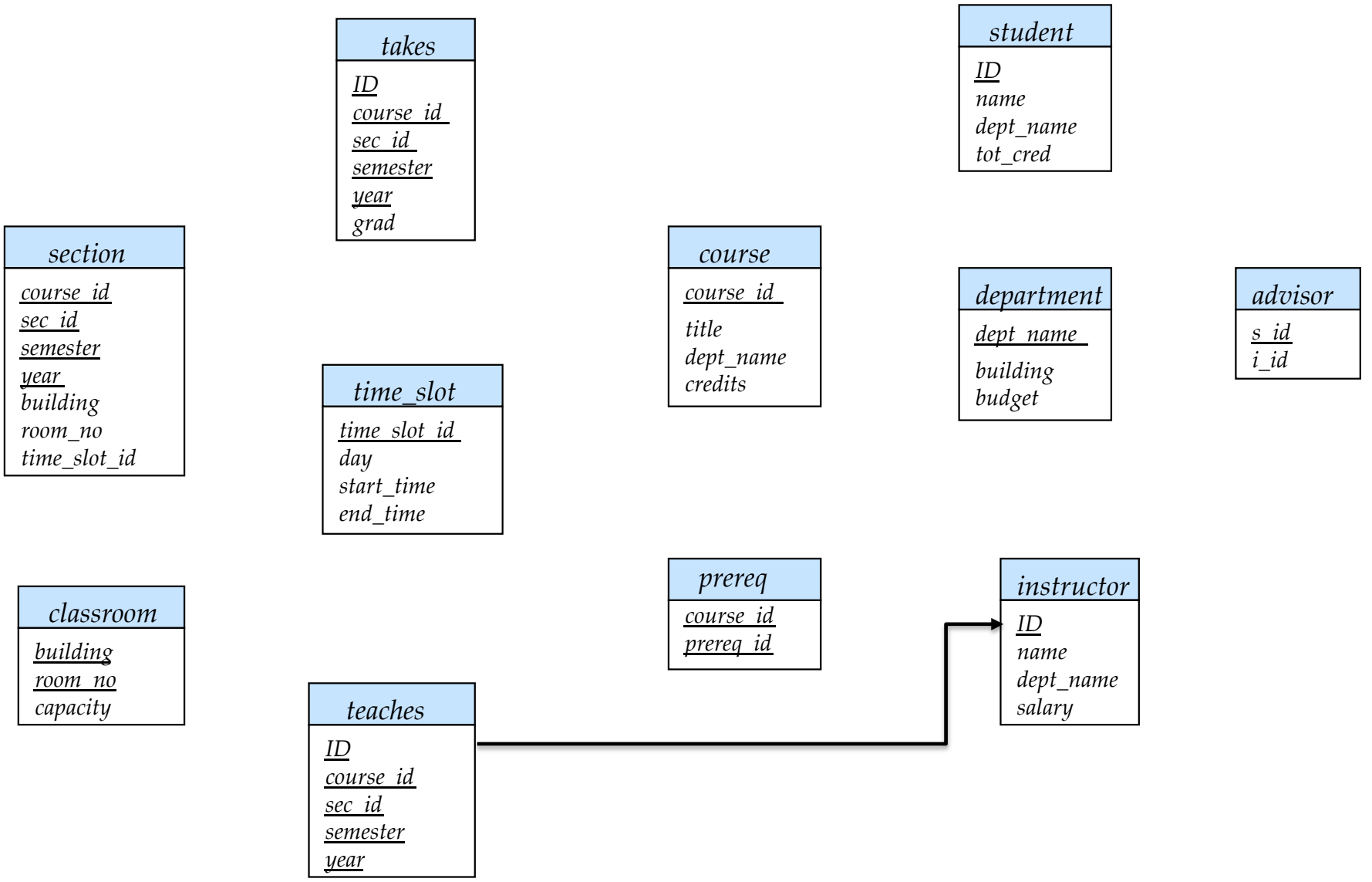
The relation is in light blue.

Primary keys are underlined.

Foreign key attributes in referencing relation

Primary key of referenced relation.

Add the Arrows…



Relational Operations

We have a set of tables or relations.

Now what? How do we get information from them?

We perform *queries*.

Simple Query:

select tuples from a relation satisfying a predicate

Results in a new relation that is a subset of the original.

Why is it useful that the result Is a relation?

Selection

Notation is $\sigma_p(x)$.

p is the *selection predicate*
x is the *relation*

p is a *boolean* formula of *terms* and *connectives.*

Connectives: $\wedge$ (and), $\vee$ (or), ~ (not)
Operators: <, >, ≤, ≥, =, ≠
Terms:

- attribute operator attribute
- attribute operator constant

Selection

Notation is $\sigma_p(x)$.

$$\sigma_{\text{salary} >= 85000}(\textit{instructor})$$

ID	*name*	*dept_name*	*salary*
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

Instructor Relation

Select the tuples with attribute salary at least 85000 from the instructor relation.

Selection

Notation is $\sigma_p(x)$.

$\sigma_{\text{salary >= 85000}}(\textit{instructor})$

ID	*name*	*dept_name*	*salary*
12121	Wu	Finance	90000
22222	Einstein	Physics	95000
33456	Gold	Physics	87000
83821	Brandt	Comp. Sci.	92000

Select the tuples with attribute salary at least 85000 from the instructor relation.

Projection

Symbol is Π

Selection of attributes.

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

ID	*name*	*dept_name*	*salary*
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

Select all tuples from the instructor relation with attributes ID and salary.

Projection

Symbol is Π

Selection of attributes.

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

ID	*salary*
10101	65000
12121	90000
15151	40000
22222	95000
32343	60000
33456	87000
45565	75000
58583	62000
76543	80000
76766	72000
83821	92000
98345	80000

Select all tuples from the instructor relation with attributes ID and salary.

Natural Join

Combine two relations into a single relation.

The tuples are joined if the attributes common to both relations are equal.

instructor ⋈ department

ID	*name*	*dept_name*	*salary*
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

⋈

dept_name	*building*	*budget*
Biology	Watson	90000
Comp. Sci.	Taylor	100000
Elec. Eng.	Taylor	85000
Finance	Painter	120000
History	Painter	50000
Music	Packard	80000
Physics	Watson	70000

Natural Join

The tuples are joined if the attributes common to both relations are equal.

instructor ⋈ department

ID	*name*	*salary*	*dept_name*	*building*	*budget*
10101	Srinivasan	65000	Comp. Sci.	Taylor	100000
12121	Wu	90000	Finance	Painter	120000
15151	Mozart	40000	Music	Packard	80000
22222	Einstein	95000	Physics	Watson	70000
32343	El Said	60000	History	Painter	50000
33456	Gold	87000	Physics	Watson	70000
45565	Katz	75000	Comp. Sci.	Taylor	100000
58583	Califieri	62000	History	Painter	50000
76543	Singh	80000	Finance	Painter	120000
76766	Crick	72000	Biology	Watson	90000
83821	Brandt	92000	Comp. Sci.	Taylor	100000
98345	Kim	80000	Elec. Eng.	Taylor	85000

Which common attribute(s) are these relations joined on?

Cartesian Product

This is the *cross product* of two relations.

Q. What is the cross product of {a, b} and {c, d}?

A. {a, b} X {c, d} produces {(a, c), (a, d), (b, c), (b, d)}

The cross product produces all possible pairs of rows of the two relations.

Q. Can you see a problem?

A. If the two relations have attributes in common, how do we tell which relation each attribute is from?

Cartesian Product Example

Relations *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 x *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

Cartesian Product Common Attributes

Relations *r, s*:

A	*B*
α	1
β	2

r

B	*D*	*E*
α	10	a
β	10	a
β	20	b
γ	10	b

s

r x *s*:

A	*r.B*	*s.B*	*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

Renaming Attributes

Allows us to refer to a relation, (say E) by more than one name.

$$\rho_x(E)$$

returns the expression *E* under the name *X*

Example.

Relations *r*

A	*B*
a	1
b	2

$r \times \rho_s(r)$

r.A	*r.B*	*s.A*	*s.B*
a	1	a	1
a	1	b	2
b	2	a	1
b	2	b	2

Union

Relations *r, s:*

For *r* ∪ *s* to be valid.

1. *r, s* must have the *same arity* (same number of attributes)

2. The attribute domains must be *compatible*

i.e, 2nd column of *r* deals with the same type of values as does the 2nd column of *s*.

Q. Did you expect there to be 4 rows?

A	B
α	1
α	2
β	1

r

A	B
α	2
β	3

s

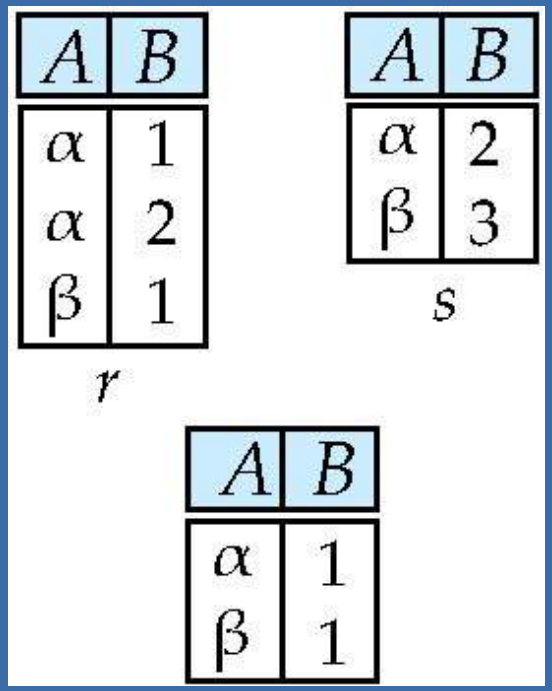
A	B
α	1
α	2
β	1
β	3

r ∪ s:

Difference

What would you expect them to be?

- Relations *r*, *s*:



A	B
α	1
α	2
β	1

r

A	B
α	2
β	3

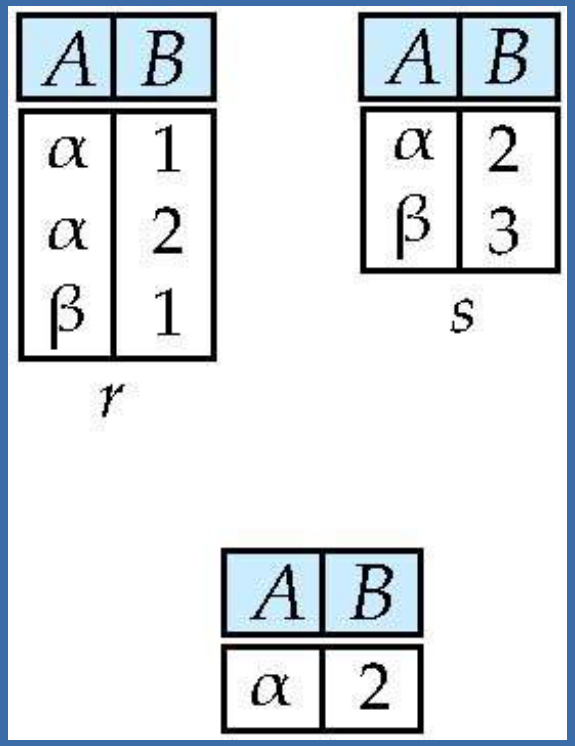
s

- $r - s$:

A	B
α	1
β	1

Intersection

- Relation *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

Note: $r \cap s = r - (r - s)$