



اصول طراحی پایگاه داده

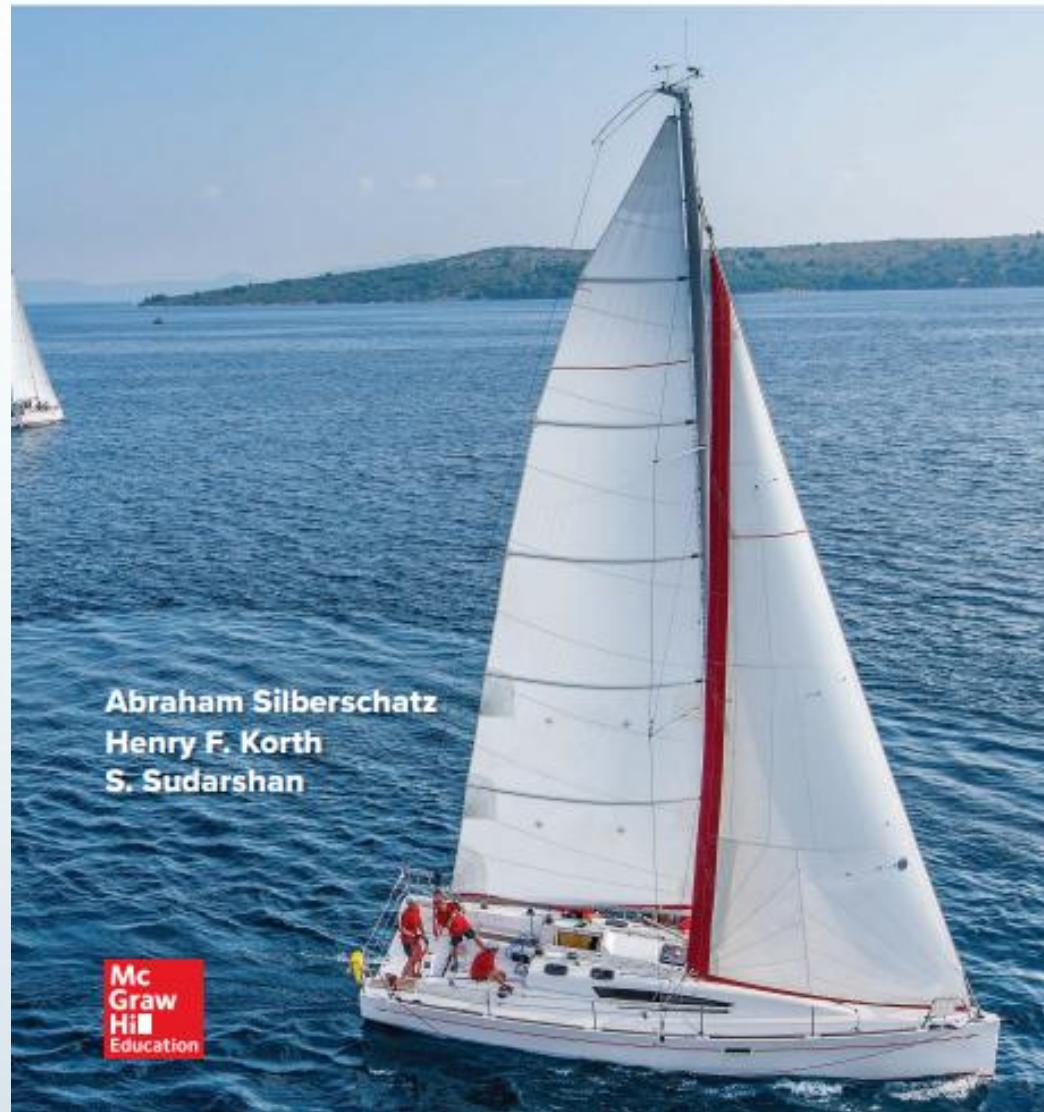
By Dr. Taghinezhad

Mail:

a0taghinezhad@gmail.com

SEVENTH EDITION

Database System Concepts





فصل ۲: مقدمه‌ای بر مدل رابطه‌ای

Database System Concepts, 7th Ed.

©Silberschatz, Korth and Sudarshan

See www.db-book.com for conditions on re-use



Outline

- ساختار پایگاه‌های داده رابطه‌ای
- طرح پایگاه داده (Schema)
- کلیدها (Keys)
- نمودارهای طرح (Schema Diagrams)
- زبان‌های پرس‌وجوی رابطه‌ای (Relational Query Languages)
- جبر رابطه‌ای (The Relational Algebra)



مثال از رابطه‌ی مربوط به اساتید

<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

ویژگی‌ها
(یا ستون‌ها)

تاپل‌ها
(یا سطر‌ها)



طرح و نمونه‌ی یک رابطه

ویژگی‌ها هستند. A_1, A_2, \dots, A_n □

طرح و الگوی رابطه را نمایش می‌دهد. $R = (A_1, A_2, \dots, A_n)$ □

مثال: □

`instructor = (ID, name, dept_name, salary)` ○

یک نمونه رابطه که بر اساس R تعریف شده است و با $r(R)$ نمایش داده

می‌شود.

مقادیر فعلی یک رابطه با استفاده از جدول مشخص می‌شوند.

هر عنصر t از رابطه r را یک تاپل(**Tuple**) می‌نامند و آن را با یک ردیف در جدول نمایش می‌دهند.



ویژگی‌ها

- ❑ مجموعه‌ی مقادیر مجاز برای هر ویژگی را **دامنه‌ی آن ویژگی (Domain)** می‌نامند.
- ❑ مقادیر ویژگی‌ها (به‌طور معمول) باید اتمی (**Atomic**) باشند؛ یعنی غیرقابل تقسیم.
- ❑ مقدار ویژه‌ای به نام **null** عضو تمامی دامنه‌ها است و بیانگر آن است که مقدار ویژگی «نامشخص» یا «نامعلوم» است.
- ❑ وجود مقدار **null** باعث پیچیدگی‌هایی در تعریف بسیاری از عملیات پایگاه داده می‌شود.



روابط بدون ترتیب هستند

- ترتیب تاپل‌ها در یک رابطه بی‌اهمیت است؛ یعنی تاپل‌ها می‌توانند به هر ترتیب دلخواهی در حافظه یا جدول ذخیره شوند

- مثال: رابطه‌ی *instructor* را می‌توان بدون هیچ ترتیب خاصی برای تاپل‌ها (ردیف‌ها) در نظر گرفت.

<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



طرح پایگاه داده (Database Schema)

- طرح پایگاه داده بیانگر ساختار منطقی پایگاه داده است.
 - نمونه‌ی پایگاه داده تصویری لحظه‌ای (**Database Instance**) از داده‌های موجود در پایگاه داده در یک زمان مشخص است.
- : مثال □
- طرح : *instructor (ID, name, dept_name, salary)*

<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

نمونه:



کلیدها

فرض کنید $K \subseteq R$

- اگر مقادیر ویژگی‌های مجموعه‌ی K برای تشخیص یکتای هر تاپل در هر رابطه‌ی ممکن (R) کافی باشند، آنگاه K را ابرکلید (Superkey) می‌نامیم.

- مثال: مجموعه‌های $\{ID\}$ و $\{ID, name\}$ هستند.
هر دو ابرکلیدهای رابطه *instructor* هستند.
- اگر ابرکلید K کمینه (Minimal) باشد – یعنی هیچ زیرمجموعه‌ای از آن نیز ابرکلید نباشد – آنگاه K را کلید کاندید (Candidate Key) می‌نامند.

- مثال: $\{ID\}$ یک کلید کاندید برای رابطه *instructor* است.
- یکی از کلیدهای کاندید به عنوان کلید اصلی (Primary Key) انتخاب می‌شود.

کدام یک؟

- حدودیت کلید خارجی (Foreign Key): مقدار موجود در یک رابطه باید در رابطه‌ی دیگری نیز ظاهر شود.
 - رابطه‌ی ارجاع‌دهنده (Referencing relation)
 - رابطه‌ی ارجاع‌شونده (Referenced relation)
- ویژگی $dept_name$ در رابطه *instructor* یک کلید خارجی است که از *dept_name* ارجاع می‌دهد.



کلیدها

- ابر کلیدها (Superkeys) کدامند?
- کلیدهای کاندید (Candidate Keys) کدامند?

<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



مثال

S

S#	Sname	City
s1	Fanavaran	Tehran
s2	Iran Segment	Ardabil
s3	Pooladin	Isfahan

P

P#	Color	Type	City
P1	Red	Iron	Tehran
P2	Green	Copper	Ardabil
P3	Blue	Brass	Isfahan
P4	Red	Iron	Tehran

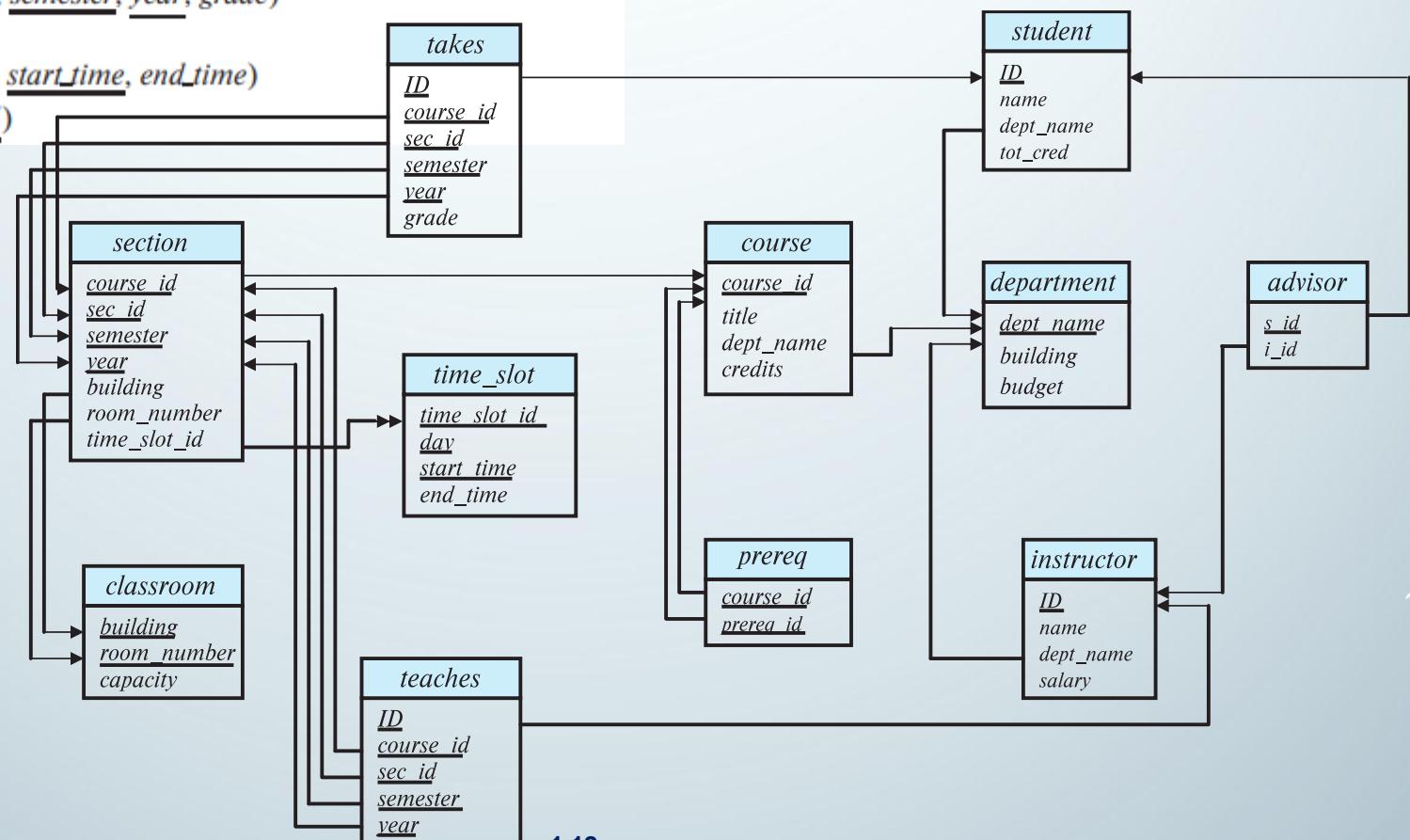
SP

S#	P#	Qty
s1	P1	300
s2	P2	200
s3	P3	400
s2	P1	300
s2	P2	400
s3	P2	200



نمودار طرح پایگاه داده دانشگاه (Schema Diagram)

classroom(building, room_number, capacity)
department(dept_name, building, budget)
course(course_id, title, dept_name, credits)
instructor(ID, name, dept_name, salary)
section(course_id, sec_id, semester, year, building, room_number, time_slot_id)
teaches(ID, course_id, sec_id, semester, year)
student(ID, name, dept_name, tot_cred)
takes(ID, course_id, sec_id, semester, year, grade)
advisor(s_ID, i_ID)
time_slot(time_slot_id, day, start_time, end_time)
prereq(course_id, prereq_id)





زبان‌های پرس‌وجوی رابطه‌ای (Relational Query Languages)

- زبان‌های پرس‌وجو رویه‌ای (Procedural) در مقابل غیررویه‌ای یا اعلامی (Declarative) هستند.
- زبان‌های خالص:
 - جبر رابطه‌ای (Relational algebra)
 - حساب رابطه‌ای تاپلی (Tuple relational calculus)
 - حساب رابطه‌ای دامنه‌ای (Domain relational calculus)
- سه زبان خالص فوق از نظر توان محاسباتی معادل یکدیگرند.
- در این فصل تمرکز ما بر جبر رابطه‌ای خواهد بود.
- جبر رابطه‌ای معادل ماشین تورینگ نیست.
- جبر رابطه‌ای شامل ۶ عمل اصلی می‌باشد.



جبر رابطه‌ای (Relational Algebra)

- یک زبان رویه‌ای (Procedural Language) که شامل مجموعه‌ای از عملیات است، به‌طوری که یک یا دو رابطه را به عنوان ورودی می‌گیرد و یک رابطه‌ی جدید به عنوان خروجی تولید می‌کند.
- شش عملگر اصلی جبر رابطه‌ای:
 - σ : (select)
 - Π : (project)
 - \cup : (union)
 - $-$: (set Difference)
 - \times : (Cartesian product)
 - ρ : (rename)



عملگر انتخاب

- عملگر انتخاب تاپل‌هایی را انتخاب می‌کند که یک شرط (Predicate) مشخص را ارضاء می‌کنند.
- نشانه‌گذاری: $\sigma_p(r)$
- در اینجا, p را شرط انتخاب (Selection Predicate) می‌نامند.
- مثال: انتخاب تاپل‌هایی از رابطه‌ی *instructor* که استاد در دپارتمان «فیزیک» قرار دارد.

Query

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

Result

<i>ID</i>	<i>name</i>	<i>dept_name</i>	<i>salary</i>
22222	Einstein	Physics	95000
33456	Gold	Physics	87000



عملگر انتخاب (ادامه)

■ در شرط انتخاب (Selection Predicate) می‌توان از عملگرهای مقایسه‌ای زیر استفاده کرد:

. =, ≠, >, ≥, <, ≤

■ همچنین می‌توان چند شرط را با استفاده از اتصال‌دهنده‌ها (Connectives) به یک شرط بزرگ‌تر ترکیب کرد:

∧ (and), ∨ (or), ¬ (not)

○ مثال: یافتن استادان دپارتمان «فیزیک» با حقوق بالاتر از ۹۰,۰۰۰ دلار

$\sigma_{dept_name="Physics"} \wedge salary > 90,000 (instructor)$

■ شرط انتخاب می‌تواند شامل مقایسه بین دو ویژگی نیز باشد.

○ مثال: یافتن تمام دپارتمان‌هایی که نام آن‌ها با نام ساختمانشان یکسان است.

$\sigma_{dept_name=building} (department)$



عملگر انتخاب - انتخاب سطرها (تاپل‌ها)

□ رابطه‌ی r : چگونه تاپل‌هایی را انتخاب کنیم که در آن‌ها

$D > 5$ و $A = B$ باشد؟

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

Select r where $A = B$ and $D > 5$ □



عملگر نمایش (Project Operation)

- عملگر پروژه یک عملگر تک جمله‌ای (Unary Operation) است که رابطه‌ی ورودی خود را بازمی‌گرداند، اما برخی از ویژگی‌ها (ستون‌ها) را حذف می‌کند.

نشانه‌گذاری:

$$\prod_{A_1, A_2, A_3, \dots, A_k} (r)$$

که در آن A_1, A_2, \dots, A_k نام ویژگی‌ها هستند و r نام رابطه است.

- نتیجه، رابطه‌ای شامل k ستون است که از حذف ستون‌هایی که در فهرست نیامده‌اند به دست می‌آید.
- ردیف‌های تکراری از نتیجه حذف می‌شوند، زیرا در مدل رابطه‌ای، رابطه یک مجموعه (Set) است و مجموعه شامل عناصر تکراری نیست.



مثال عملگر نمایش

مثال: حذف ویژگی $dept_name$ از رابطه $instructor$ ■

پرس و جو: ■

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

: نتیجه ■

ID	$name$	$salary$
10101	Srinivasan	65000
12121	Wu	90000
15151	Mozart	40000
22222	Einstein	95000
32343	El Said	60000
33456	Gold	87000
45565	Katz	75000
58583	Califieri	62000
76543	Singh	80000
76766	Crick	72000
83821	Brandt	92000
98345	Kim	80000



ترکیب عملیات رابطه‌ای (Composition of Relational Operations)

- نتیجه‌ی هر عملگر جبر رابطه‌ای خود یک رابطه است؛ بنابراین، می‌توان چندین عملگر جبر رابطه‌ای را به صورت ترکیبی در کنار هم استفاده کرد تا یک عبارت جبر رابطه‌ای تشکیل شود.
- مثال: یافتن نام تمام استادان در دپارتمان فیزیک.

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

- به جای آن که نام یک رابطه را مستقیماً به عنوان ورودی عملگر نمایش بدهیم، می‌توانیم عبارتی را به کار ببریم که خودش یک رابطه را تولید می‌کند.



عملگر ضرب دکارتی (Cartesian-Product Operation)

عملگر ضرب دکارتی (که با نماد \times نمایش داده می‌شود) به ما اجازه می‌دهد تا اطلاعات دو رابطه را با یکدیگر ترکیب کنیم.

- مثال: ضرب دکارتی دو رابطه‌ی *instructor* و *teaches* به صورت زیر نوشته می‌شود:

instructor \times *teaches*

در نتیجه، برای هر جفت ممکن از تاپل‌ها (یکی از رابطه‌ی *instructor* و دیگری از رابطه‌ی *teaches*)، یک تاپل جدید ساخته می‌شود.
(به اسلاید بعدی مراجعه کنید).

از آنجا که ویژگی *ID* در هر دو رابطه وجود دارد، برای جلوگیری از ابهام، نام رابطه‌ای که ویژگی از آن آمده را به آن اضافه می‌کنیم:

instructor.ID ○

teaches.ID ○



The *instructor* x *teaches* table

<i>instructor.ID</i>	<i>name</i>	<i>dept_name</i>	<i>salary</i>	<i>teaches.ID</i>	<i>course_id</i>	<i>sec_id</i>	<i>semester</i>	<i>year</i>
10101	Srinivasan	Comp. Sci.	65000	10101	CS-101	1	Fall	2017
10101	Srinivasan	Comp. Sci.	65000	10101	CS-315	1	Spring	2018
10101	Srinivasan	Comp. Sci.	65000	10101	CS-347	1	Fall	2017
10101	Srinivasan	Comp. Sci.	65000	12121	FIN-201	1	Spring	2018
10101	Srinivasan	Comp. Sci.	65000	15151	MU-199	1	Spring	2018
10101	Srinivasan	Comp. Sci.	65000	22222	PHY-101	1	Fall	2017
...
...
12121	Wu	Finance	90000	10101	CS-101	1	Fall	2017
12121	Wu	Finance	90000	10101	CS-315	1	Spring	2018
12121	Wu	Finance	90000	10101	CS-347	1	Fall	2017
12121	Wu	Finance	90000	12121	FIN-201	1	Spring	2018
12121	Wu	Finance	90000	15151	MU-199	1	Spring	2018
12121	Wu	Finance	90000	22222	PHY-101	1	Fall	2017
...
...
15151	Mozart	Music	40000	10101	CS-101	1	Fall	2017
15151	Mozart	Music	40000	10101	CS-315	1	Spring	2018
15151	Mozart	Music	40000	10101	CS-347	1	Fall	2017
15151	Mozart	Music	40000	12121	FIN-201	1	Spring	2018
15151	Mozart	Music	40000	15151	MU-199	1	Spring	2018
15151	Mozart	Music	40000	22222	PHY-101	1	Fall	2017
...
...
22222	Einstein	Physics	95000	10101	CS-101	1	Fall	2017
22222	Einstein	Physics	95000	10101	CS-315	1	Spring	2018
22222	Einstein	Physics	95000	10101	CS-347	1	Fall	2017
22222	Einstein	Physics	95000	12121	FIN-201	1	Spring	2018
22222	Einstein	Physics	95000	15151	MU-199	1	Spring	2018
22222	Einstein	Physics	95000	22222	PHY-101	1	Fall	2017
...
...



اتصال دو رابطه - ضرب دکارتی

A	B
α	1
β	2

r

C	D	E
α	10	a
β	10	a
β	20	b
γ	10	b

s

روابط r و 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

: r x s □



ضرب دکارتی – مسئله‌ی نام‌گذاری

A	B
α	1
β	2

r

B	D	E
α	10	a
β	10	a
β	20	b
γ	10	b

s

روابط r و 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

: r x s □



عملگر پیوند (Join Operation)

■ در عملگر ضرب کارتزینی

instructor X teaches

هر تاپل از رابطه‌ی *instructor* با هر تاپل از رابطه‌ی *teaches* ترکیب می‌شود.

○ اما بیشتر ردیف‌های حاصل، شامل اطلاعاتی درباره‌ی اساتیدی هستند که هیچ درسی را تدریس نکرده‌اند.

■ برای به‌دست آوردن فقط آن تاپل‌هایی از *instructor X teaches* که مربوط به اساتید و درس‌هایی است که واقعاً آن‌ها تدریس کرده‌اند، می‌نویسیم:

$\sigma_{instructor.id = teaches.id} (instructor \times teaches))$

○ در نتیجه، تنها تاپل‌هایی باقی می‌مانند که به اساتید و درس‌هایی که توسط آن‌ها تدریس شده است مربوط می‌شوند.

■ نتیجه‌ی این عبارت در اسلاید بعدی نمایش داده می‌شود.



عملگر پیوند (ادامه)

جدول متناظر با:

$$\sigma_{instructor.id = teaches.id} (instructor \times teaches)$$

<i>instructor.ID</i>	<i>name</i>	<i>dept_name</i>	<i>salary</i>	<i>teaches.ID</i>	<i>course_id</i>	<i>sec_id</i>	<i>semester</i>	<i>year</i>
10101	Srinivasan	Comp. Sci.	65000	10101	CS-101	1	Fall	2017
10101	Srinivasan	Comp. Sci.	65000	10101	CS-315	1	Spring	2018
10101	Srinivasan	Comp. Sci.	65000	10101	CS-347	1	Fall	2017
12121	Wu	Finance	90000	12121	FIN-201	1	Spring	2018
15151	Mozart	Music	40000	15151	MU-199	1	Spring	2018
22222	Einstein	Physics	95000	22222	PHY-101	1	Fall	2017
32343	El Said	History	60000	32343	HIS-351	1	Spring	2018
45565	Katz	Comp. Sci.	75000	45565	CS-101	1	Spring	2018
45565	Katz	Comp. Sci.	75000	45565	CS-319	1	Spring	2018
76766	Crick	Biology	72000	76766	BIO-101	1	Summer	2017
76766	Crick	Biology	72000	76766	BIO-301	1	Summer	2018
83821	Brandt	Comp. Sci.	92000	83821	CS-190	1	Spring	2017
83821	Brandt	Comp. Sci.	92000	83821	CS-190	2	Spring	2017
83821	Brandt	Comp. Sci.	92000	83821	CS-319	2	Spring	2018
98345	Kim	Elec. Eng.	80000	98345	EE-181	1	Spring	2017



عملگر پیوند (ادامه)

عملگر پیوند (Join) این امکان را فراهم می‌کند که عملگر انتخاب (Select) و عملگر ضرب دکارتی (Cartesian Product) را در قالب یک عملگر واحد ترکیب کنیم.

فرض کنید دو رابطه $r(R)$ و $s(S)$ را داریم:

اگر θ (تتا) یک شرط بر روی ویژگی‌های RUS باشد، آنگاه عملگر پیوند به صورت زیر تعریف می‌شود:

$$r \bowtie_{\theta} s = \sigma_{\theta}(r \times s)$$

بنابراین:

$$\sigma_{instructor.id = teaches.id} (instructor \times teaches))$$

را میتوان به صورت زیر نوشت:

$$instructor \bowtie Instructor.id = teaches.id teaches.$$



عملگر اجتماع (Union Operation)

- عملگر اجتماع این امکان را فراهم می‌کند که دو رابطه را با یکدیگر ترکیب کنیم.
- نشانه‌گذاری: $r \cup s$
- برای آنکه عبارت $r \cup s$ معتبر باشد، شرایط زیر باید برقرار باشند:
 1. r و s باید همارزی (Arity) داشته باشند؛ یعنی تعداد ویژگی‌ها یکسان باشد.
 2. دامنه‌های ویژگی‌ها باید سازگار باشند.
- مثال: ستون دوم رابطه‌ی r باید با نوع مقادیر ستون دوم رابطه‌ی s همخوانی داشته باشد.



عملگر اجتماع (ادامه)

- مثال: یافتن تمام درس‌هایی که در ترم پاییز ۱۴۰۷ یا در ترم بهار ۱۴۰۸ یا در هر دو ترم تدریس شده‌اند.

```
classroom(building, room_number, capacity)
department(dept_name, building, budget)
course(course_id, title, dept_name, credits)
instructor(ID, name, dept_name, salary)
section(course_id, sec_id, semester, year, building, room_number, time_slot_id)
teaches(ID, course_id, sec_id, semester, year)
student(ID, name, dept_name, tot_cred)
takes(ID, course_id, sec_id, semester, year, grade)
advisor(s_ID, i_ID)
time_slot(time_slot_id, day, start_time, end_time)
prereq(course_id, prereq_id)
```

$$\prod_{course_id} (\sigma_{semester="Fall"} \wedge year=2017(section)) \cup \prod_{course_id} (\sigma_{semester="Spring"} \wedge year=2018(section))$$



عملگر اجتماع (ادامه)

نتیجه:

$$\begin{aligned} & \prod_{course_id} (\sigma_{semester="Fall"} \wedge year=2017(section)) \cup \\ & \prod_{course_id} (\sigma_{semester="Spring"} \wedge year=2018(section)) \end{aligned}$$

course_id
CS-101
CS-315
CS-319
CS-347
FIN-201
HIS-351
MU-199
PHY-101



اجتماع دو رابطه

روابط r و s را در نظر بگیرید:

اجتماع دو رابطه رابطه ای است که تاپلهايش در یک یا هر دو رابطه وجود دارند. یعنی رکوردهایی 2 جدول با هم ترکیب شده و رکوردهای تکراری یکبار نوشته می شوند.

A	B
α	1
α	2
β	1

r

A	B
α	2
β	3

s

: $r \cup s$ □

r union s □

A	B
α	1
α	2
β	1
β	3



عملگر اشتراک مجموعه‌ای (Set-Intersection Operation)

- عملگر اشتراک مجموعه‌ای این امکان را فراهم می‌کند که تاپل‌هایی را بیابیم که در هر دو رابطه‌ی ورودی وجود دارند.

```

classroom(building, room_number, capacity)
department(dept_name, building, budget)
course(course_id, title, dept_name, credits)
instructor(ID, name, dept_name, salary)
section(course_id, sec_id, semester, year, building, room_number, time_slot_id)
teaches(ID, course_id, sec_id, semester, year)
student(ID, name, dept_name, tot_cred)
takes(ID, course_id, sec_id, semester, year, grade)
advisor(s_ID, i_ID)
time_slot(time_slot_id, day, start_time, end_time)
prereq(course_id, prereq_id)

```

نشانه‌گذاری: $r \cap s$

فرض‌ها:

r, s هم ارز باشند.

ویرگی‌های r و s با یکدیگر سازگار باشند.

مثال: یافتن مجموعه‌ی تمام درس‌هایی که هم در ترم پاییز ۲۰۱۷ و هم در ترم بهار ۲۰۱۸ تدریس شده‌اند:

نتیجه:

$$\prod_{course_id} (\sigma_{semester="Fall"} \wedge year=2017(section)) \cap \\ \prod_{course_id} (\sigma_{semester="Spring"} \wedge year=2018(section))$$

course_id

CS-101



اشتراک مجموعه‌ای دو رابطه

(Set intersection of two relations)

روابط r و s

A	B
α	1
α	2
β	1

r

A	B
α	2
β	3

s

A	B
α	2

$r \cap s$

r intersect s

نکته: $r \cap s = r - (r - s)$



عملگر تفاضل مجموعه‌ای (Set Difference Operation)

- عملگر تفاضل مجموعه‌ای این امکان را فراهم می‌کند که تاپل‌هایی را بیابیم که در یک رابطه وجود دارند ولی در رابطه‌ی دیگر نیستند.
- نشانه‌گذاری: $r - s$
- تفاضل مجموعه‌ای باید بین **روابط سازگار** گرفته شود.
- s , r هم ارز باشند.
 - دامنه‌های ویژگی‌های r و s باید سازگار باشند.
- مثال: یافتن تمام درس‌هایی که در ترم پاییز ۲۰۱۷ تدریس شده‌اند اما در ترم بهار ۲۰۱۸ تدریس نشده‌اند.

$$\prod_{course_id} (\sigma_{semester="Fall"} \wedge year=2017 (section)) - \prod_{course_id} (\sigma_{semester="Spring"} \wedge year=2018 (section))$$

course_id
CS-347
PHY-101



عملگر انتساب (The Assignment Operation)

- گاهی اوقات راحت‌تر است که یک عبارت جبر رابطه‌ای را با اختصاص دادن بخشی از آن به متغیرهای موقت رابطه‌ای بنویسیم.
- عملگر انتساب با نماد \leftarrow نشان داده می‌شود و مانند انتساب در زبان‌های برنامه‌نویسی عمل می‌کند.
- مثال: یافتن تمام استادان در دپارتمان‌های «فیزیک» و «موسیقی»:

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

- با استفاده از عملگر انتساب، یک پرس‌وجو می‌تواند به صورت یک برنامه‌ی ترتیبی نوشته شود که شامل سری از انتساب‌ها و در نهایت عبارتی است که مقدار آن به عنوان نتیجه‌ی پرس‌وجو نمایش داده می‌شود.



عملگر تغییر نام (The Rename Operation)

- نتایج عبارات جبر رابطه‌ای نامی ندارند که بتوان به آن‌ها ارجاع داد. عملگر تغییر نام با نماد ρ برای همین منظور ارائه شده است.
- نمادگذاری:

$$\rho_x(E)$$

- عبارت فوق نتیجه‌ی عبارت E را تحت نام X بازمی‌گرداند.
- شکل دیگر عملگر تغییر نام:

$$\rho_{x(A1, A2, \dots An)}(E)$$

که در آن علاوه بر نام رابطه، نام ویژگی‌ها (ستون‌ها) نیز تغییر داده می‌شوند.



پرس‌وجوهای معادل (Equivalent Queries)

- برای نوشتن یک پرس‌وجو در جبر رابطه‌ای بیش از یک روش وجود دارد.
- مثال: یافتن اطلاعات مربوط به درس‌هایی که توسط استادان دپارتمان فیزیک با حقوق بیش از ۹۰,۰۰۰ دلار تدریس شده‌اند.

پرس‌وجوی ۱

$$\sigma_{dept_name = "Physics"} \wedge salary > 90,000 \text{ (instructor)}$$

پرس‌وجوی ۲

$$\sigma_{dept_name = "Physics"} (\sigma_{salary > 90.000} \text{ (instructor)})$$

دو پرس‌وجوی مختلف ممکن است یکسان نباشند، اما معادل (Equivalent) هستند؛

یعنی روی هر پایگاه داده‌ای نتیجه‌ی یکسانی تولید می‌کنند.



پرس‌وجوهای معادل (Equivalent Queries)

- مثال: یافتن اطلاعات مربوط به درس‌هایی که توسط استادان دپارتمان فیزیک تدریس شده‌اند.

پرس‌وجوی ۱

$$\sigma_{dept_name = "Physics"}(instructor \bowtie_{instructor.ID = teaches.ID} teaches)$$

پرس‌وجوی ۲

$$(\sigma_{dept_name = "Physics"}(instructor)) \bowtie_{instructor.ID = teaches.ID} teaches$$



مثال

Name of P2 producers

اسامی تهیه کنندگان قطعه P2

S

S#	Sname	City
s1	Fanavaran	Tehran
s2	Iran Segment	Tabriz
s3	Pooladin	Tabriz

P

P#	Color	Type	City
P1	Red	Iron	Tehran
P2	Green	Copper	Tabriz
P3	Blue	Brass	Shiraz
P4	Red	Iron	Tehran

SP

S#	P#	Qty
s1	P1	300
s2	P2	200
s3	P3	400
s2	P1	300
s2	P2	400
s3	P2	200



مثال

Name of P2 producers

اسامی تهیه کنندگان قطعه P2 را تهیه میکنند ■

$$\Pi_{Sname} (\sigma_{P\#=P2} (S \bowtie SP))$$

S join SP giving Temp1

Select Temp1 where P# = 'P2' Giving Temp2

Project Temp2 [Sname]

S

S#	Sname	City
s1	Fanavaran	Tehran
s2	Iran Segment	Tabriz
s3	Pooladin	Tabriz

P

P#	Color	Type	City
P1	Red	Iron	Tehran
P2	Green	Copper	Tabriz
P3	Blue	Brass	Shiraz
P4	Red	Iron	Tehran

SP

S#	P#	Qty
s1	P1	300
s2	P2	200
s3	P3	400
s2	P1	300
s2	P2	400
s3	P2	200



مثال

اسامی تهیه کنندگان را که حداقل یک قطعه قرمز رنگ تهیه می کنند

S

S#	Sname	City
s1	Fanavaran	Tehran
s2	Iran Segment	Tabriz
s3	Pooladin	Tabriz

P

P#	Color	Type	City
P1	Red	Iron	Tehran
P2	Green	Copper	Tabriz
P3	Blue	Brass	Shiraz
P4	Red	Iron	Tehran

SP

S#	P#	Qty
s1	P1	300
s2	P2	200
s3	P3	400
s2	P1	300
s2	P2	400
s3	P2	200



مثال

اسامی تهیه کنندگان را که حداقل یک قطعه قرمز رنگ تهیه می کنند

$\text{Temp1} \leftarrow \sigma_{\text{Color}=\text{'Red'}}(\text{P})$
 $\text{Temp2} \leftarrow \Pi_{\text{p}\#}(\text{Temp1})$
 $\text{Temp3} \leftarrow (\text{temp2} \bowtie \text{SP})$
 $\text{Temp4} \leftarrow \Pi_{\text{s}\#}(\text{Temp3})$
 $\text{Temp5} \leftarrow \Pi_{\text{s}\#}(\text{Temp4} \bowtie \text{S})$
 $\text{Temp6} \leftarrow \Pi_{\text{sname}}(\text{Temp5})$

Select P where color='Red' Giving Temp1
 Project Temp1 [P#] Giving Temp2
 Temp2 Join SP Giving Temp3
 Project Temp3 [S#] Giving Temp4
 Temp4 Join S Giving Temp5
 Project Temp5 [Sname] Giving Temp6

S

S#	Sname	City
s1	Fanavararan	Tehran
s2	Iran Segment	Tabriz
s3	Pooladin	Tabriz

P

P#	Color	Type	City
P1	Red	Iron	Tehran
P2	Green	Copper	Tabriz
P3	Blue	Brass	Shiraz
P4	Red	Iron	Tehran

SP

S#	P#	Qty
s1	P1	300
s2	P2	200
s3	P3	400
s2	P1	300
s2	P2	400
s3	P2	200



مثال

اسامی تهیه کنندگانی را بباید که قطعه P2 را تهیه نمی کند ■

S

S#	Sname	City
s1	Fanavaran	Tehran
s2	Iran Segment	Tabriz
s3	Pooladin	Tabriz

P

P#	Color	Type	City
P1	Red	Iron	Tehran
P2	Green	Copper	Tabriz
P3	Blue	Brass	Shiraz
P4	Red	Iron	Tehran

SP

S#	P#	Qty
s1	P1	300
s2	P2	200
s3	P3	400
s2	P1	300
s2	P2	400
s3	P2	200



مثال

اسامی تهیه کنندگانی را بباید که قطعه P2 را تهیه نمی کند ■

$$\Pi_{Sname} \left[\left[\Pi_{s\#} (S) - \left(\Pi_{s\#} (\sigma_{P\#=P2'} (SP)) \right) \right] \circ S \right]$$

S

S#	Sname	City
s1	Fanavaran	Tehran
s2	Iran Segment	Tabriz
s3	Pooladin	Tabriz

P

P#	Color	Type	City
P1	Red	Iron	Tehran
P2	Green	Copper	Tabriz
P3	Blue	Brass	Shiraz
P4	Red	Iron	Tehran

SP

S#	P#	Qty
s1	P1	300
s2	P2	200
s3	P3	400
s2	P1	300
s2	P2	400
s3	P2	200



پیوند شرطی

عملگر پیوند شرطی (θ -join)

این عملگر، زیر مجموعه ای از ضرب دکارتی است که شرط θ روی سطرهای آن اعمال شده باشد.
ستونهای خروجی معادل ستونهای ضرب دکارتی است.

X_θ

$S \ X_{s.\text{city} >= P.\text{city}} \ P$

مثال خروجی دستور زیر

S#	Sname	City	P#	Color	Type	City
s1	Fanavaran	Tehran	P1	Red	Iron	Tehran
s1	Fanavaran	Tehran	P2	Green	Copper	Tabriz
s1	Fanavaran	Tehran	P4	Red	Iron	Tehran
s2	Iran Segment	Tabriz	P2	Green	Copper	Tabriz
s3	Pooladin	Tabriz	P3	Blue	Brass	Tabriz



(Natural Join) پیوند طبیعی

■ $R =$

A	B
X	Y
X	Z
Y	Z
Z	V

$S =$

B	C
Z	U
V	W
Z	V

■ $R | \times | S =$

A	B	C
X	Z	U
X	Z	V
Y	Z	U
Y	Z	V
Z	V	W



(Semi-join) نیم اتصال

مشابه پیوند طبیعی است با این تفاوت که فقط ستونهای جدول اول را می دهد.

$$(\sigma_{\text{city} = \text{"Tehran"}}(P)) \propto (\sigma_{\text{Qty} > 200}(SP))$$

S

S#	Sname	City
s1	Fanavaran	Tehran
s2	Iran Segment	Tabriz
s3	Pooladin	Tabriz

P

P#	Color	Type	City
P1	Red	Iron	Tehran
P2	Green	Copper	Tabriz
P3	Blue	Brass	Shiraz
P4	Red	Iron	Tehran

SP

S#	P#	Qty
s1	P1	300
s2	P2	200
s3	P3	400
s2	P1	300
s2	P2	400
s3	P2	200



اتصال خارجی یا فراپیوند (Outer Join)

۱- فراپیوند چپ L-O-JOIN : Left Outer Join

۲- فراپیوند راست R-O-JOIN : Right Outer Join

۳- فراپیوند کامل F-O-JOIN : Full Outer Join

فراپیوند چپ گونه ای از عملگر پیوند طبیعی است با این تفاوت که علاوه بر تاپلهای پیوند شدنی از دو رابطه، تاپلهای نشدنی از رابطه چپ هم ، پیوند شده با مقادیر **NULL**، در جواب وارد می شود در فراپیوند راست، تاپلهای پیوند نشدنی از رابطه سمت راستی ، پیوند شده با مقادیر **NULL** وارد می شوند.

مثال:

S L-O-JOIN SP

S

S#	Sname	Status
s1	Sn1	10
s2	Sn2	15
S3	Sn3	10
S4	Sn4	20

SP

S#	P#	Qty
S1	P1	200
S2	P3	400
S2	P4	100
S3	P1	300



Outer Join

مثال:

S L-O-JOIN SP

S#	Sname	Status	P#	Qty
s1	Sn1	10	P1	200
s2	Sn2	15	P3	400
s2	Sn2	15	P4	100
S3	Sn3	10	P1	300
S4	Sn4	20	NULL	NULL

S

S#	Sname	Status
s1	Sn1	10
s2	Sn2	15
S3	Sn3	10
S4	Sn4	20

SP

S#	P#	Qty
S1	P1	200
S2	P3	400
S2	P4	100
S3	P1	300



خلاصه‌ای از عملگرهای جبر رابطه‌ای

Symbol (Name)	Example of Use
σ (Selection)	$\sigma \text{ salary} >= 85000 \text{ (instructor)}$ Return rows of the input relation that satisfy the predicate.
Π (Projection)	$\Pi ID, \text{salary} \text{ (instructor)}$ Output specified attributes from all rows of the input relation. Remove duplicate tuples from the output.
\times (Cartesian Product)	$\text{instructor} \times \text{department}$ Output pairs of rows from the two input relations that have the same value on all attributes that have the same name.
\cup (Union)	$\Pi name \text{ (instructor)} \cup \Pi name \text{ (student)}$ Output the union of tuples from the <i>two</i> input relations.
$-$ (Set Difference)	$\Pi name \text{ (instructor)} -- \Pi name \text{ (student)}$ Output the set difference of tuples from the two input relations.
\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.



عملگر تقسیم (Divide operator)

فرض کنید یک جدول اضافی در پایگاه داده‌ی بانک داریم.

Account-types	Type
	checking
	savings

حال فرض کنید می‌خواهیم بدانیم کدام مشتریان دارای تمام انواع حساب‌ها هستند.



عملیات تقسیم

÷ or / divide

$$(\pi_{Owner, Type} \text{ Account}) \div \text{Account-types}$$

Account

Number	Owner	Balance	Type
101	J. Smith	1000.00	checking
102	W. Wei	2000.00	checking
103	J. Smith	500.00	savings
104	M. Jones	1000.00	checking
105	H. Mart	10,000.00	checking

Account-types

Type
checking
savings

این عملگر به ما کسانی را برمی‌گرداند که تمام نوع‌های موجود در جدول **Account-types** دارند.

در این مثال:

- جدول **Account-types** شامل دو نوع حساب است: **savings** و **checking**.
- بنابراین، ما می‌خواهیم صاحبان حسابی را بیابیم که هم حساب **checking** و هم حساب **savings** داشته باشند.



تعريف تقسيم

برای $R \div S$ که در آن $S(s1, s2) \text{ و } R(r1, r2, r3, r4)$ داریم:

- از آنجایی که S دارای دو ویژگی است، باید در R (مثل $r3$ و $r4$) وجود داشته باشند که دامنه آنها با $s1$ و $s2$ سازگار باشد. میتوان گفت که $(s1, s2)$ و $(r3, r4)$ سازگار برای اتحاد (union-compatible) هستند.
- نتیجه‌ی پرس‌وجو شامل ویژگی‌های باقی‌مانده $(r1, r2)$ است.
- یک تاپل $(r1, r2)$ در پاسخ قرار می‌گیرد اگر مقدار $(r1, r2)$ با هر تاپلی از S در R ظاهر شود.



کی تقسیم کنیم؟

- عملگر تقسیم به عنوان یک عملگر بنیادی پشتیبانی نمی‌شود، اما برای بیان پرس‌وجوهایی مانند:

«یافتن مشتریانی که تمام انواع حساب‌ها را دارند» مفید است.

- فرض کنید A دارای دو ویژگی x و y باشد و B فقط دارای ویژگی y باشد.

$$A/B = \{(x) \mid \exists (x,y) \in A \quad \forall (y) \in B\}$$

- یعنی $A \div B$ شامل تمام تاپل‌های (x) (مشتریان) است که برای هر تاپل y (نوع حساب) در B ، یک تاپل (x,y) در A وجود داشته باشد.

- به عبارت دیگر، اگر مجموعه‌ی مقادیر y (نوع حساب‌ها) مرتبط با یک مقدار X (مشتری) در A شامل تمام مقادیر y در B باشد، آنگاه مقدار X در $A \div B$ قرار می‌گیرد.

- به طور کلی، X و y می‌توانند هر لیستی از ویژگی‌ها باشند؛ y لیست ویژگی‌های موجود در B است، $X \cup y$ لیست ویژگی‌های A است.



مثال تقسیم

Table:A2

sno	pno
s1	p1
s1	p2
s1	p3
s1	p4
s2	p1
s2	p2
s3	p2
s4	p2
s4	p4

Table:B1

pno
p2

Table:B2

pno
p2
p4

Table:B3

pno
p1
p2
p4

$A \div B1:$

sno
s1
s2
s3
s4

$A \div B1:$

sno
s1
s4

$A \div B3:$

sno
s1

.1. ستون sno شماره‌ی تأمین‌کننده (supplier number)

.2. ستون pno شماره‌ی قطعه (part number) می‌دهد.



End of Chapter 2