

طراحی پایگاه داده رابطه ای

فاطمه منصوری

این اسلاید ها بر اساس اسلاید های درس طراحی پایگاه داده دانشگاه شریف که توسط دکتر مرتضی امینی ایجاد شده است، تهیه شده است.

طراحی پایگاه داده رابطه ای

□ در طراحی پایگاه داده های رابطه ای باید موارد زیر را مشخص نمود:

□ مجموعه ای از رابطه ها

□ کلید(های) کاندید هر رابطه

□ کلید اصلی هر رابطه

□ کلیدهای خارجی هر رابطه (در صورت وجود)

□ محدودیت های جامعیتی ناظر بر هر رابطه

طراحی با روش بالا به پایین (Top-Down)

طراحی با روش سنتز [نرم افزار ترسازی رابطه ها]

□ روش های طراحی RDB:

طراحی پایگاه داده رابطه‌ای (ادامه)

□ روشن طراحی بالا به پایین

□ ابتدا مدلسازی داده‌ها را (با روش UML [E]ER یا می‌دهیم و سپس مدلسازی را به

مجموعه‌ای از رابطه‌ها تبدیل می‌کنیم.

□ روشن طراحی سنتز رابطه‌ای (نرمال‌ترسازی)

□ ابتدا مجموعه صفات خرد جهان واقع را مشخص می‌کنیم. سپس با تحلیل قواعد و محدودیت‌های ناظر

به صفات و تشخیص وابستگی‌های بین آنها، صفات را متناسبًا با هم سنتز می‌کنیم (نوعی گروه‌بندی)

تا به مجموعه‌ای از رابطه‌های نرمال دست یابیم.

□ در عمل روش ترکیبی استفاده می‌شود، یعنی ابتدا روش بالا به پایین، سپس نرمال‌ترسازی.

ویژگی های طراحی خوب

- ❑ نمایش صحیح و واضح از خردجهان واقع باشد.
- ❑ تمام داده های کاربران قابل نمایش باشد و همه محدودیت های (قواعد) جامعیتی منظور شده باشد.
- ❑ کمترین افزونگی
- ❑ کمترین هیچ مقدار
- ❑ کمترین مشکل در عملیات ذخیره سازی
- ❑ بیشترین کارایی در بازیابی

طراحی بالا به پایین

- تبدیل نمودار ER[E] به مجموعه‌ای از رابطه‌های نرمال (و نه لزوماً در نرمال‌ترین صورت) در طراحی RDB،
نهایتاً طراح تصمیم می‌گیرد چند رابطه داشته باشد و عنوان (Heading) هر رابطه چه باشد.
- در نمودار مدلسازی معنایی داده‌ها، حالات متعدد داریم، که در ادامه به آنها می‌پردازیم.
- فرض: تا اطلاع ثانوی، همه صفات ساده‌اند و موجودیت‌ها ضعیف نیستند.

حالت ۱: طراحی ارتباط چند به چند

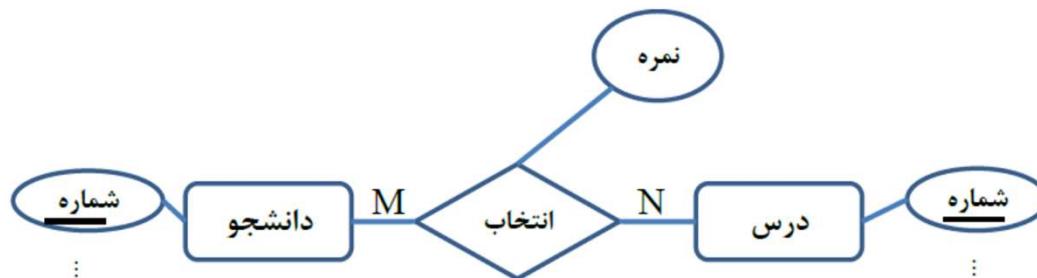
حالت ۱

درجه ارتباط: n=2

چندی ارتباط: M:N

سه رابطه لازم است.

طراحی در این حالت با کمتر از سه رابطه، افزونگی و هیچ مقداری زیادی پدید می‌آورد.



STUD (STID,)

COR (COID,)

SCR (STID, COID, GR)

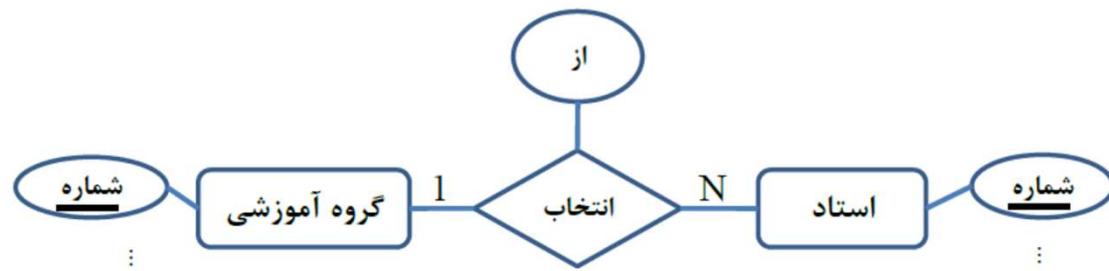
حالت ۲: طراحی ارتباط یک به چند

حالت ۲

درجه ارتباط: ۲

چندی ارتباط: ۱:N

دو رابطه لازم است. رابطه سمت ۱ به رابطه سمت N FK می‌دهد (بیرون از کلید اصلی).



DEPT (DEID, DTID,, DPHONE)

PROF (PRID, PRNAME,, PRANK, DEID, FROM)

حالت ۲: طراحی ارتباط یک به چند (ادامه)

□ در چه وضعی طراحی این حالت با سه رابطه قابل توجیه است؟

۱- وقتی که مشارکت سمت N در ارتباط غیرالزامی باشد (درصد مشارکت کمتر از ۳۰ درصد) و تعداد

استاد زیاد باشد، برای کاهش مقدار Null، رابطه نمایشگر ارتباط را جدا می کنیم.

۲- فرکانس ارجاع به خود ارتباط بالا باشد و به صفات دیگر با فرکانس پایین تری احتیاج باشد.

۳- تعداد صفات خود ارتباط زیاد باشد و باعث زیاد شدن درجه ارتباط PROF شود.

□ اگر مشارکت سمت N الزامی باشد، باید این محدودیت معنایی را از طریق هیچ مقدار ناپذیر بودن صفت کلید

خارجی (با استفاده از NOT NULL) در رابطه نمایانگر نوع موجودیت سمت N، اعلام کرد.

حالت ۳: طراحی ارتباط یک به یک

حالت ۳

□ درجه ارتباط: n=2

□ چندی ارتباط: 1:1

با دو / یا سه / یا یک رابطه طراحی می کنیم.



□ در صورت طراحی با **دو** رابطه، رابطه مربوط به نوع موجودیت با مشارکت الزامی، FK می گیرد.

COUR (COID,, BKID)

BOOK (BKID,, BKPRICE)

حالت ۳: طراحی ارتباط یک به یک (ادامه)

- وقتی با **سه** رابطه توجیه دارد که مشارکت طرفین غیرالزامی باشد، تعداد شرکت‌کنندگان (نمونه‌ها) در ارتباط زیاد باشد، درصد مشارکت در رابطه ضعیف (کمتر از ۳۰٪) باشد و نیز ملاحظاتی در مورد فرکانس ارجاع.
- وقتی با **یک** رابطه توجیه دارد که تعداد صفات موجودیت‌ها کم باشد، مشارکت طرفین الزامی باشد و فرکانس ارجاع به ارتباط کم باشد.

حالت ۴: طراحی ارتباط خود ارجاع چند به چند

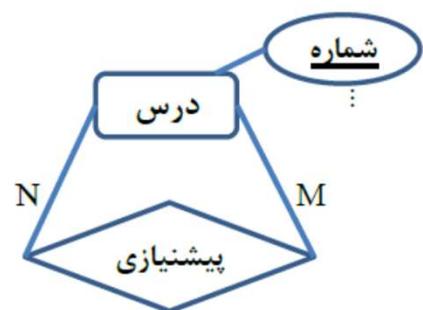
حالت ۴

حالت خاص حالت اول

درجه ارتباط: $n=1$

چندی ارتباط: $M:N$

دو رابطه لازم است.



COUR (COID,)

COPRECO (COID, PRECOD) →

بیش از یک صفت از رابطه، از یک دامنه هستند.

حالت 5: طراحی ارتباط خود ارجاع یک به چند

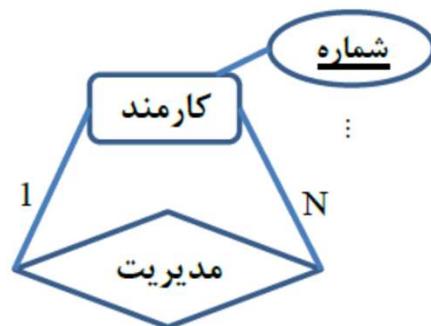
حالت 5

حالت خاص حالت دوم

درجه ارتباط: $n=1$

چندی ارتباط: $1:N$

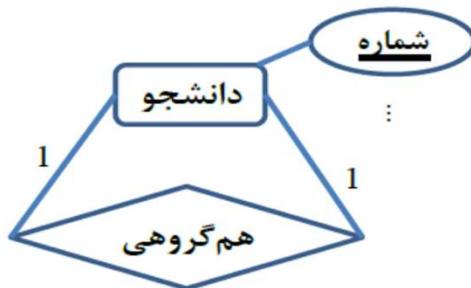
یک رابطه لازم است.



EMPL (EMID, ENAME,, EPHONE, EMGRID)

حالت 6: طراحی ارتباط خود ارجاع یک به یک

حالت 6



حالت خاص حالت سوم

درجه ارتباط: $n=1$

چندی ارتباط: 1:1

با یک یا دو رابطه طراحی می‌کنیم.

اگر مشارکت در همپروژگی زیاد نباشد، از مدل II استفاده می‌کنیم.

(I) **STPROJST** (STID, STNAME,, JSTID)
 P.K. C.K.

(II) **STUD** (STID, STNAME,)

STJST (STID, JSTID)
 C.K. C.K.

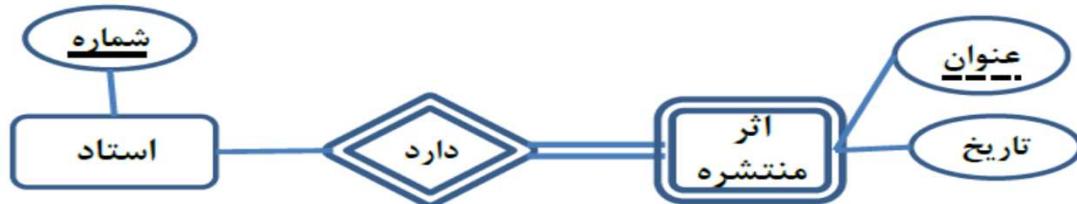
در STJST هر یک از صفات می‌توانند کلید اصلی باشند.

حالت 7: طراحی موجودیت ضعیف

حالت 7

□ موجودیت ضعیف داریم.

دو رابطه لازم است؛ یکی برای نوع موجودیت قوی، یکی برای نوع موجودیت ضعیف و ارتباط شناسا. رابطه نمایشگر موجودت ضعیف از موجودیت قوی FK می‌گیرد که در ترکیب با صفت ممیزه می‌شود PK.

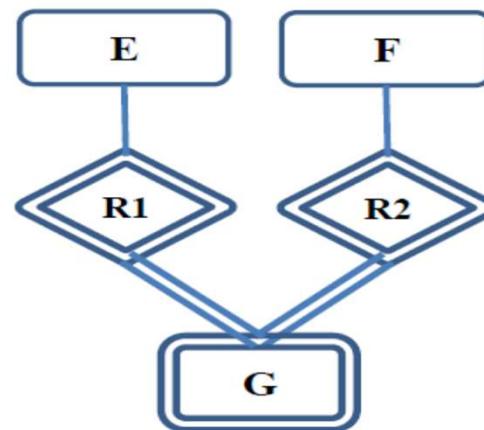
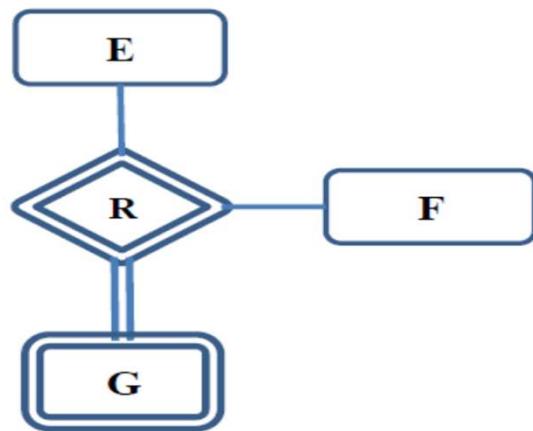


PROF (PRID, PRNAME,)

PRPUB (PRID, PTITLE, PTYPE,)

حالت 7: طراحی موجودیت ضعیف (ادامه)

□ تمرین: رابطه‌های لازم برای مدل‌های داده‌ای زیر طراحی شود.



در این حالت کلید رابطه G از ترکیب کلید رابطه‌های E و F (و در صورت وجود صفت ممیزه G) حاصل می‌گردد.

حالت 8: طراحی صفت چندمقداری

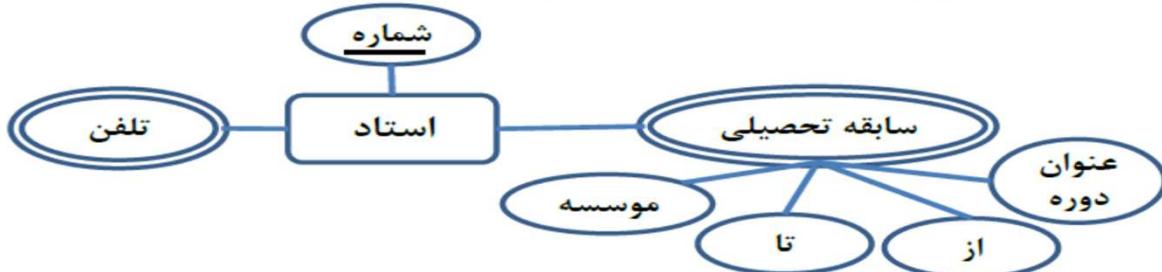
حالت 8

وجود یک صفت چندمقداری برای یک نوع موجودیت.

سه تکنیک دارد:

۱- [تکنیک عمومی] یک رابطه برای خود نوع موجودیت و یک رابطه برای هر صفت چندمقداری.

(بنابراین اگر نوع موجودیت E , m صفت چندمقداری داشته باشد، $m+1$ رابطه داریم.



PROF (PRID, PRNAME,)

PRTEL (PRID, PHONE)

رابطه نمایشگر صفت چندمقداری از نوع ✓

موجودیت اصلی FK می‌گیرد داخل کلید.

حالت 8: طراحی صفت چندمقداری (ادامه)

□ در مدلسازی، موجودیت ضعیف به صفت چندمقداری ارجحیت دارد ولی تکنیک عمومی طراحی آنها مثل هم است.

PRHIS (PRID, TTL, FROM, TO, INSTNAME,)

□ اشکال تکنیک عمومی: اگر برای نوع موجودیت اصلی اطلاعات کامل بخواهیم، باید عمل JOIN انجام دهیم که می‌تواند زمانگیر باشد.

حالت ۸: طراحی صفت چندمقداری (ادامه)

۲ - [در شرایط خاص] طراحی با یک رابطه (فرض: یک صفت چندمقداری): یک رابطه برای خود نوع موجودیت و صفت چندمقداری.

□ با فرض مشخص بودن حداکثر تعداد مقداری که صفت چندمقداری می‌گیرد، به همان تعداد صفت در رابطه در نظر می‌گیریم.

فرض: هر استاد حداکثر سه شماره تلفن دارد.



PRTELTEL (PRID, PRNAME, PRRANK, PHONE1, PHONE2, PHONE3)

□ مزیت این تکنیک: JOIN لازم ندارد.

□ عیب این تکنیک: هیچمقدار (Null) در آن زیاد است، اگر تعداد کمی از استادان، سه شماره تلفن داشته باشند.

حالت ۹: طراحی ارتباط IS-A

حالت ۹

وجود ارتباط IS-A بین دو نوع موجودیت.

چهار تکنیک دارد:

۱- فرض: نوع موجودیت E, n زیرنوع دارد.

n+1 رابطه طراحی می‌کنیم. یک رابطه برای زیرنوع و یک رابطه برای هر یک از زیرنوع‌ها.

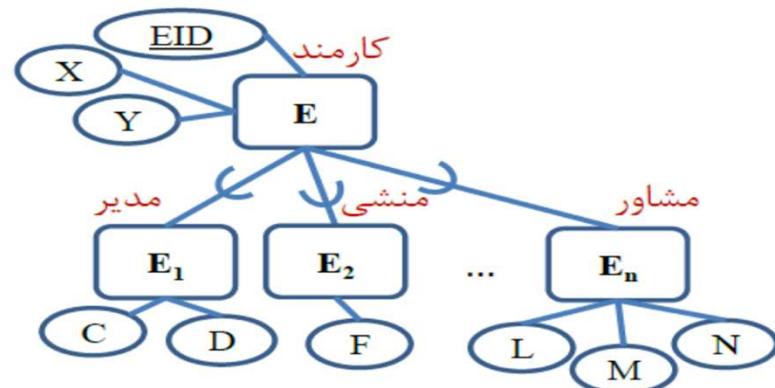
E (EID, X, Y)

E1 (EID, A, B)

E2 (EID, F)

...

En (EID, L, M, N)



حالت 9: طراحی ارتباط IS-A

- مزیت این تکنیک: شرط خاصی از نظر نوع تخصیص ندارد (تکنیک‌های دیگری که مطرح می‌شود، همگی برای شرایط خاص هستند).
- عیب این تکنیک: اگر بخواهیم در مورد یک زیرنوع، اطلاعات کامل به دست آوریم، باید JOIN کنیم.

حالت 9: طراحی ارتباط IS-A

۲- طراحی با n رابطه: برای زبرنوع، رابطه‌ای طراحی نمی‌کنیم. بنابراین صفات مشترک باید در رابطه نمایشگر هر زیرنوع وجود داشته باشد.

□ شرط لازم: باید تخصیص کامل باشد. اگر نباشد، بخشی از داده‌های محیط قابل نمایش نیستند.

E1 (EID, X, Y, A, B)

E2 (EID, X, Y, F)

...

En (EID, X, Y, L, M, N)

□ مزیت نسبت به تکنیک اول: برای به دست آوردن اطلاعات کامل زیرنوع‌ها نیازی به JOIN نیست.

□ نکته: در این تکنیک، لزوماً افزونگی پیش نمی‌آید. اگر تخصیص همپوشانی افزونگی پیش می‌آید.

حالت 9: طراحی ارتباط IS-A

۳- طراحی فقط با یک رابطه، با استفاده از صفت نمایشگر نوع زیرنوع‌ها

- شرط استفاده از این تکنیک: تخصیص مجرا باشد; یعنی یک نمونه کارمند، جزء نمونه‌های حداکثر یک زیرنوع باشد.

E (EID, X, Y, A, B, F, L, M, N, TYPE)

100	x1	y1	a1	b1	?	?	?	?	مدیر
200	x2	y2	?	?	?	12	m2	n2	مشاور

□ مزیت این تکنیک: برای به دست آوردن اطلاعات کامل زیرنوع‌ها نیازی به JOIN نیست.

□ عیب این تکنیک: هیچمقدار (Null) زیاد دارد و درجه رابطه زیاد است.

حالت ۹: طراحی ارتباط IS-A

۴- طراحی فقط با یک رابطه، با استفاده از آرایه بیتی؛ هر بیت نمایشگر نوع یک زیرنوع در واقع برای نمایش هر نمونه موجودیت، بسته به اینکه در مجموعه نمونه‌های کدام زیرنوع باشد، بیت مربوطه‌اش را ۱ می‌کنیم.

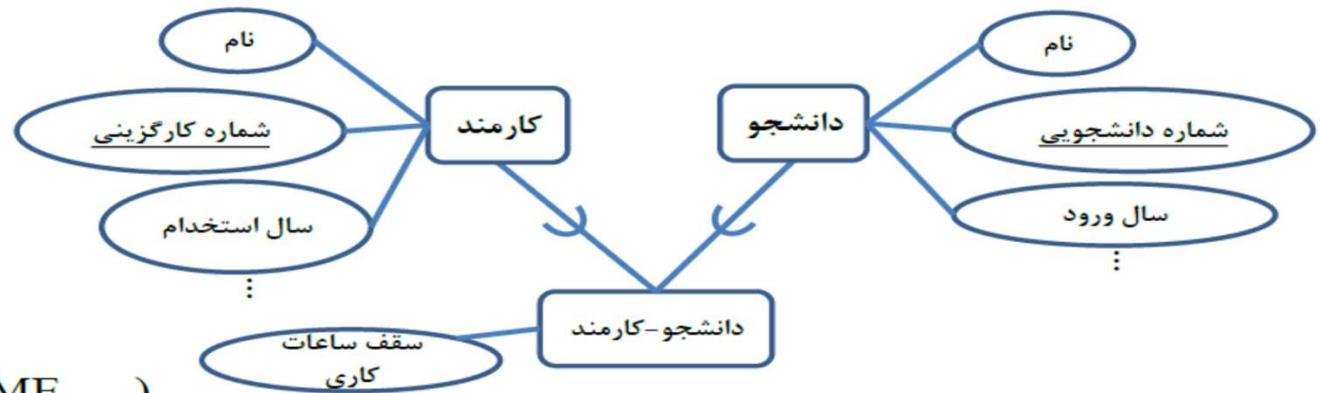
□ شرط استفاده از این تکنیک: وقتی تخصیص همپوشان باشد

آرایه بیتی			
E (EID, X, Y, A, B, F, L, M, N, TB1, TB2, ..., TBn)			
	مدیر	منشی	مشاور
100 x1 y1	1	0	0
200 x2 y2	0	1	0

حالت 10 : طراحی ارث بری چندگانه

حالت 10

- وجود ارث بری چندگانه بین یک زیرنوع با چند زیرنوع
- اگر زیرنوع، n زیرنوع داشته باشد، رابطه نمایشگر زیرنوع حداقل n کلید کاندید با ارجاع بیشتر کلید اصلی انتخاب می شود.



STUD (STID, STNAME, ...)

EMPL (EID, ENAME, ...)

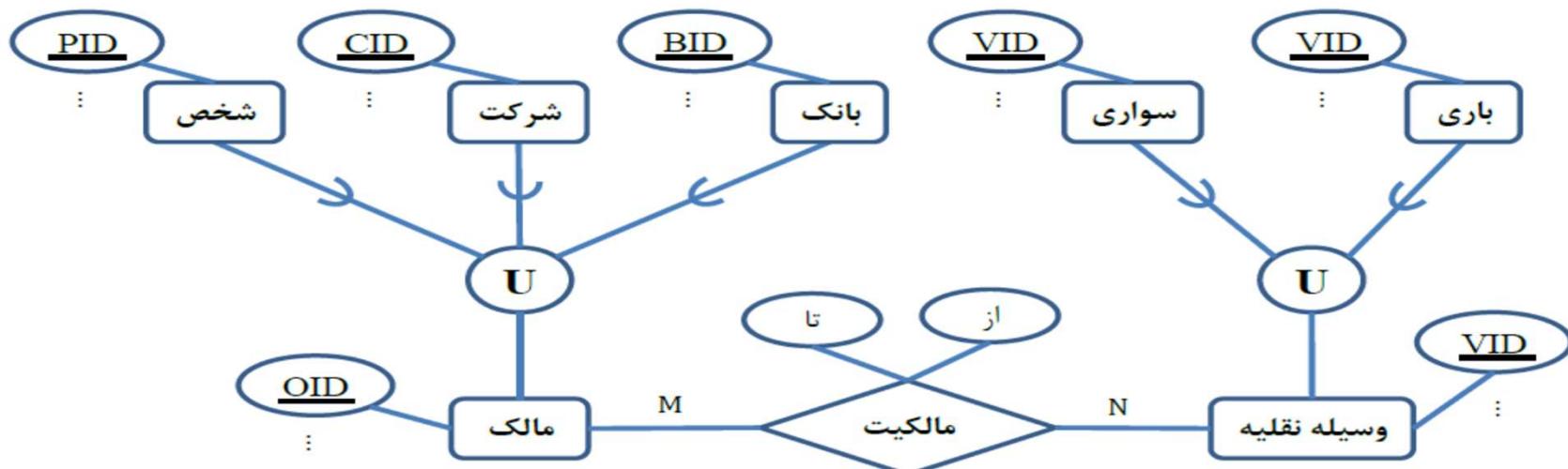
STEM (STID, EID, MAXW)

حالت ۱۱ : طراحی زیرنوع اجتماع

حالت ۱۱

نوع موجودیت E، زیرنوع U-Type (دسته یا Category) n زبرنوع است.

n+1 رابطه طراحی می کنیم.



حالت 11 : طراحی زیرنوع اجتماع (ادامه)

□ رابطه n+1

- اگر شناسه زبرنوع‌ها از دامنه‌های متفاوت باشد، رابطه نمایشگر زیرنوع، FK می‌دهد به رابطه‌های نمایشگر زبرنوع‌ها، خارج از کلید.
- اگر شناسه زبرنوع‌ها از یک دامنه باشد، کلید رابطه نمایشگر زیرنوع، همان کلید رابطه‌های نمایشگر زبرنوع‌ها است.
 - PERS** (PID,, OID)
 - COMP** (CID,, OID)
 - BANK** (BID,, OID)
 - OWNER** (OID,....) ← چون دامنه کلیدهای زبرنوع‌ها یکسان نیست، خودمان کلید ساختگی می‌گذاریم.
 - VEHIC** (VID,)
 - OWNS** (OID, VID, F, T,)
 - SAVARY** (VID, N,)
 - BARY** (VID, T,)

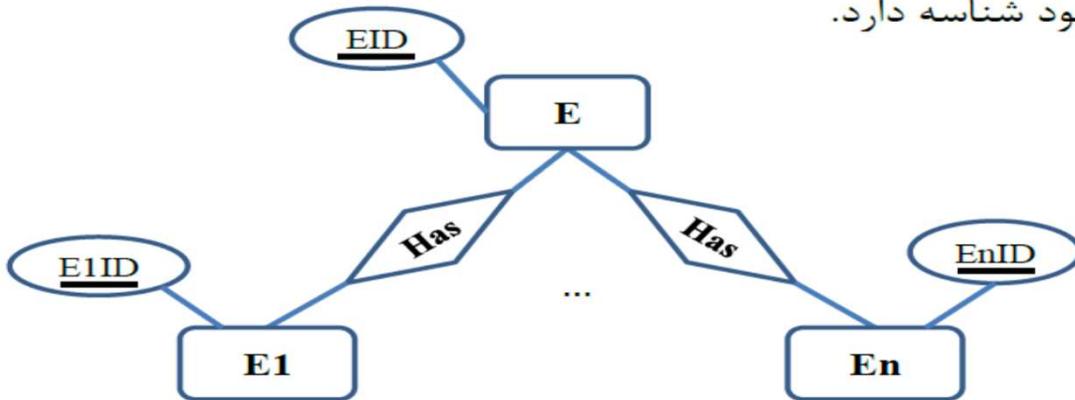
حالت ۱۲ : طراحی ارتباط IS-A-PART-OF

حالت ۱۲

وجود ارتباط IS-A-PART-OF

اگر نوع موجودیت کل، n نوع موجودیت جزء داشته باشد، تعداد $n+1$ رابطه طراحی می‌کنیم.

توجه داریم که نوع موجودیت جزء از خود شناسه دارد.



E (EID,)

E1 (E1ID, EID,)

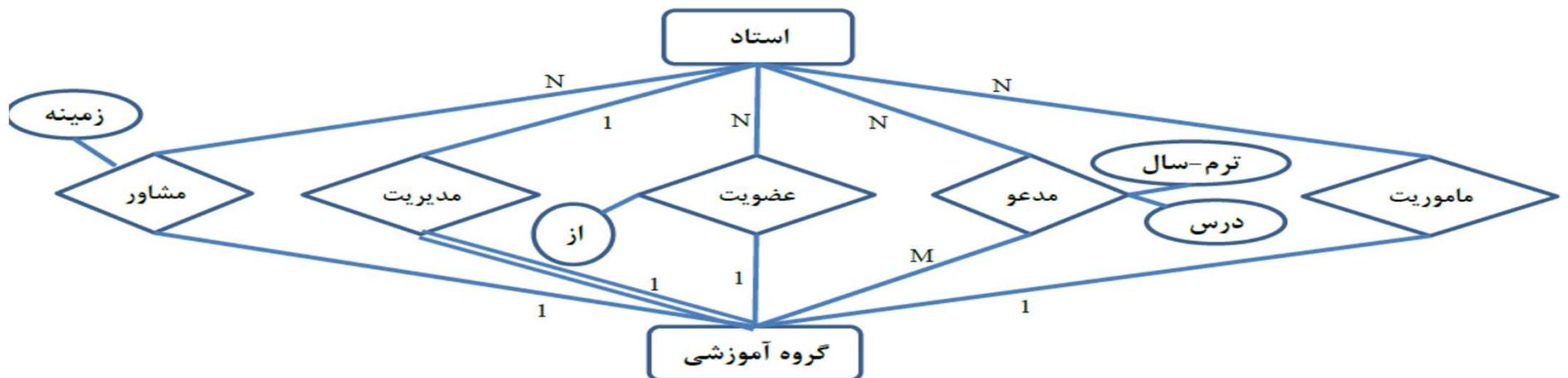
....

En (EnID, EID,)

آیا طرز طراحی دیگری وجود دارد؟ در چه شرایطی؟ 

طراحی با وجود چند ارتباط

- در صورتیکه چند ارتباط مثلاً بین دو نوع موجودیت برقرار باشد.
- هر ارتباط را با توجه به وضع آن از نظر درجه و چندی ارتباط طراحی می‌کنیم. اما برای کاهش احتمال اشتباه در طراحی توصیه می‌شود اول ارتباطهای $M:N$ ، سپس $N:1$ و در آخر $1:1$ را طراحی نماییم.



طراحی با وجود چند ارتباط

DEPT (DEID,, DPHONE, PRID)

PROF (PRID,, PRRANK, MDEID, SUB, MEMDEID, FROM, CDEID, INT)

زمینه مشاور ماموریت موضوع عضویت از سه کلید خارجی از یک دامنه



INVITED (DEID, PRID, YR, TR)

□ همین سیستم حداکثر با هفت رابطه نیز قابل طراحی است.

روش سنتز یا نرمالترسازی رابطه ها

- **ایده اصلی:** یک رابطه، هر چند نرمال (با تعریفی که قبلاً دیدیم) ممکن است آنومالی (مشکل) داشته باشد در عملیات ذخیره‌سازی (در درج، حذف یا بهنگام‌سازی).
- **آنومالی در درج:** عدم امکان درج یک فقره اطلاع که منطقاً باید قابل درج باشد.
- **آنومالی در حذف:** حذف یک اطلاع ناخواسته در پی حذف اطلاع خواسته.
- **آنومالی در بهنگام‌سازی:** بروز فزون کاری.
- پس باید رابطه را نرمال‌تر کرد.

آنومالی مثال

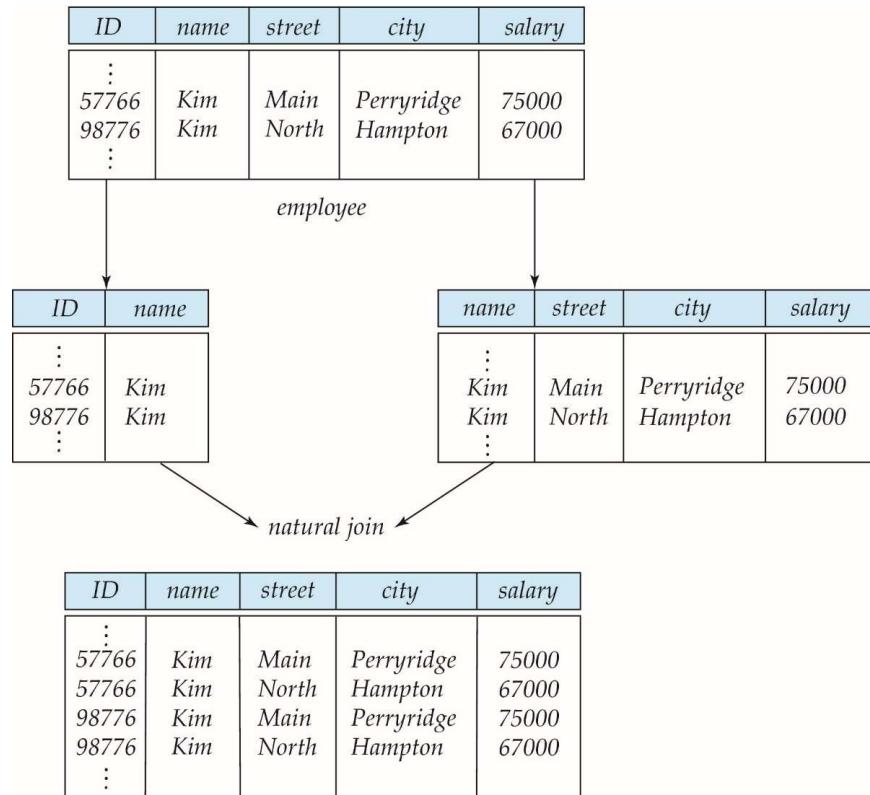
- فرض کنید دو table، دانشکده و استاد را با هم یکی کرده باشیم. و هر دو را در یک table ذخیره کرده باشیم. در این صورت، در درج، حذف و به روزرسانی آنومالی داریم.

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

decomposition

- روشی که می توانیم از آنومالی جلوگیری کنیم این است که table را به دو table دیگر تقسیم کنیم. ولی تمام تقسیم بندی ها خوب نیستند.
- *employee*(*ID*, *name*, *street*, *city*, *salary*)
- *employee1* (*ID*, *name*)
- *employee2* (*name*, *street*, *city*, *salary*)
- در حالت بالا اگر دو کارمند با نام یکسان داشته باشیم، مشکل ایجاد می شود.

نقسیم بندی lossy



در این حالت با اینکه در هنگام *join*، تاپل ها اضافه شده اند، ولی در واقع داده از دست داده ایم.
زیرا ارتباط بین *kim* و آدرس اش از دست رفته است.

Lossless decomposition

- رابطه R را در نظر بگیرید که آنرا به دو رابطه R_1 و R_2 تقسیم کرده ایم به طوریکه ویژگی های R ، اجتماع ویژگی های R_1 و R_2 باشد.
- تقسیم بندی در صورتی lossless است که اگر به جای R ، جوین دو رابطه R_1 و R_2 را قرار دهیم، داده ای از دست نرود

$$\Pi_{R_1}(r) \bowtie \Pi_{R_2}(r) = r \quad : \text{Lossless} \bullet$$

```
select *
from (select R1 from r)
      natural join
      (select R2 from r)
```

$$r \subset \Pi_{R_1}(r) \bowtie \Pi_{R_2}(r) \quad : \text{Lossy} \bullet$$

$$R = (A, B, C)$$

$$R_1 = (A, B) \quad R_2 = (B, C)$$

A	B	C
α	1	A

$$r$$

A	B
α	1

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

B	C
1	A

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

$$\Pi_A(r) \bowtie \Pi_B(r)$$

A	B	C
α	1	A

تئوری نرمالسازی

- تصمیم گیری در مورد اینکه آیا یک رابطه به حالت مناسب (حالت نرمال) است
- در حالتی که رابطه در حالت نرمال نیست آنرا به تعدادی رابطه تقسیم کنیم:
 - هر کدام از روابط در حالت مناسب (نرمال) باشند
 - تقسیم بندی lossless باشد.

فرم های نرمال

□ فرم های نرمال:

1NF
2NF
3NF

(Boyce-Codd Normal Form) BCNF

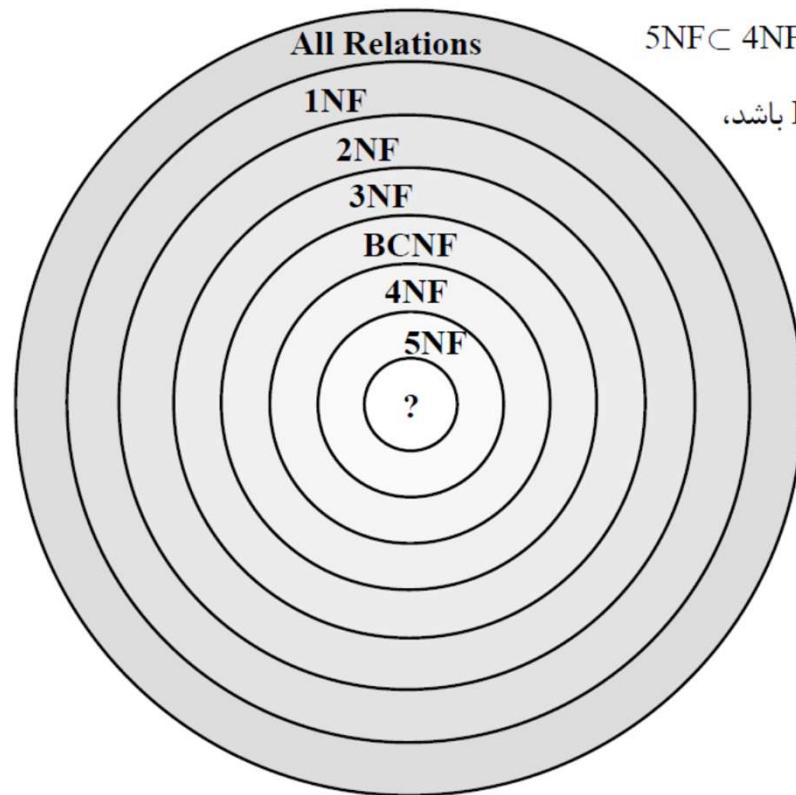
4NF
5NF
6NF

(Domain Key Normal Form) DKNF

چندان کاربرد عملی ندارند.
سطوح در تئوری؛

رابطه نرمال تر / آنومالی کمتر

رابطه بین فرم های نرمال



$5NF \subset 4NF \subset BCNF \subset 3NF \subset 2NF \subset 1NF$

یعنی به طور مثال، رابطه‌ای که BCNF باشد،

3NF هم هست.

وابستگی تابعی

- در سازمانی که برای آن پایگاه داده طراحی می‌کنیم، ممکن است قوانینی و وابستگیهایی روی داده‌ها وجود داشته باشد.

• برای مثال:

- هر دانشجو و هر استاد یک شماره مخصوص به خود دارد
- هر دانشجو و هر استاد فقط یک اسم دارد.
- هر استاد و هر دانشجو به یک دانشکده تعلق دارند.
- هر دانشکده فقط یک بودجه و یک ساختمان دارد.
- هر نمونه (instance) از پایگاه داده که وابستگی را حفظ کند، یک نمونه قانونی (legal) است.

تئوری وابستگی

□ برای بررسی فرم‌های نرمال، نیاز به مفاهیمی داریم از تئوری وابستگی (Dependency Theory).

□ مفاهیمی از تئوری وابستگی:

□ وابستگی تابعی (Functional Dependency)

□ وابستگی تابعی کامل [تام] (Fully Functional Dependency)

□ وابستگی تابعی با واسطه (Transitive Functional Dependency)

وابستگی تابعی



وابستگی تابعی (FD): صفت $R.A$ به صفت $R.B$ وابستگی تابعی دارد اگر و فقط اگر به ازای یک

مقدار از A یک مقدار از B متناظر باشد. به عبارت دیگر اگر t_1 و t_2 دو تاپل از R باشند، در این صورت:

$$\text{IF } t_1.A = t_2.A \text{ THEN } t_1.B = t_2.B$$

با فرض اینکه کل تاپلهای رابطه به صورت زیر باشد، آیا داریم:



R (A, B, C)

$a_1, b_1, c_1,$

$a_1 \rightarrow b_1$

? $A \rightarrow B$ بله

$a_1 \quad b_1 \quad c_2$

? $A \rightarrow C$ خیر

$a_2 \quad b_2 \quad c_2$

$a_1 \leq^c_1 c_2$

? $B \rightarrow A$ خیر

$a_3 \quad b_3 \quad c_3$

? $B \rightarrow C$ خیر

$a_4 \quad b_2 \quad c_3$

وابستگی تابعی(ادامه)

□ نکات:

(۱) صفات طرفین FD می‌توانند ساده یا مرکب باشند.

(۲) اگر $B \rightarrow A$, $A \rightarrow B$, لزوماً نداریم:

(۳) اگر $A \rightarrow B$, $B \subseteq A$, به FD نامهم یا بدیهی (Trivial) گوییم.

(۴) اگر $K \rightarrow G$ در رابطه R, CK یا SK باشد و $G \subseteq H_R$ آنگاه داریم:

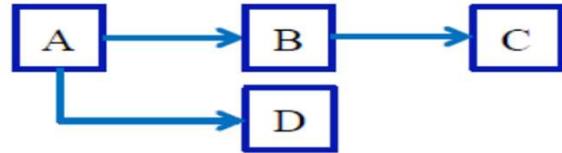
وابستگی تابعی(ادامه)

(۵) نمایش FD‌های رابطه R به روشهای مختلف:

- به صورت یک مجموعه:

$$F = \{A \rightarrow B, B \rightarrow C, A \rightarrow D\}$$

- با نمودار FD‌ها:



- روی خود عنوان رابطه با استفاده از فلش‌هایی:



وابستگی تابعی(ادامه)

(۶) **تفسیر FD:** هر FD نمایشگر یک قاعده معنایی از محیط است: نوعی قاعده جامعیتی (که باید به نحوی به سیستم داده شود.

: $STID \rightarrow STJ$: یک دانشجو فقط می‌تواند در یک رشته تحصیل کند.



: $STJ \rightarrow STD$: یک رشته فقط در یک دانشکده ارائه می‌شود.

: $STID \rightarrow STD$: یک دانشجو فقط در یک دانشکده تحصیل می‌کند.

وابستگی تابعی- قواعد آرمسترانگ

□ قواعد استنتاج آرمسترانگ

1- if $B \subseteq A$ then $A \rightarrow B \Rightarrow A \rightarrow A$

(قاعده انعکاسی)

2- if $A \rightarrow B$ and $B \rightarrow C$ then $A \rightarrow C$

(قاعده تعددی یا تراگذاری)

3- if $A \rightarrow B$ then $(A, C) \rightarrow (B, C)$

(قاعده افزایش)

4- if $A \rightarrow (B, C)$ then $A \rightarrow B$ and $A \rightarrow C$

(قاعده تجزیه)

5- if $A \rightarrow B$ and $C \rightarrow D$ then $(A, C) \rightarrow (B, D)$

(قاعده ترکیب)

6- if $A \rightarrow B$ and $A \rightarrow C$ then $A \rightarrow (B, C)$

(قاعده اجتماع)

7- if $A \rightarrow B$ and $(B, C) \rightarrow D$ then $(A, C) \rightarrow D$

(قاعده شبه تعددی)

وابستگی تابعی- قواعد آرمسترانگ

□ سه قاعده اول درست و کامل هستند، بدین معنا که با داشتن یک مجموعه از وابستگی‌های تابعی F

تمام وابستگی‌های تابعی منطقاً قابل استنتاج از F ، با همین سه قاعده به دست می‌آیند و هیچ

وابستگی تابعی دیگر (که از F قابل استنتاج نباشد) نیز به دست نمی‌آید.

□ **توجه:** سه قاعده اول به آسانی قابل اثبات هستند و قواعد دیگر از روی همانها اثبات می‌شوند.

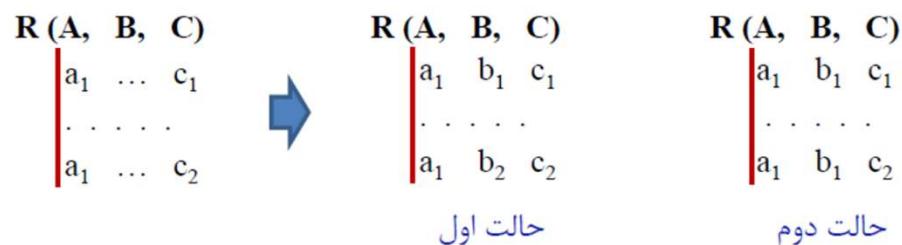
وابستگی تابعی - قواعد آرمسترانگ

تمرين: قاعده ۲ را اثبات کنید (با استفاده از برهان خلف).

□ اثبات: فرض خلف: گیریم که $C \rightarrow A$. در این صورت در رابطه R در حداقل دو تاپل، به ازای یک مقدار

A، دو مقدار متمایز از C داریم.

□ اما به ازای دو مقدار متمایز C , مقدار B ممکن است دو مقدار متمایز یا یک مقدار باشد.



□ در حالت اول، فرض $B \rightarrow A$ و در حالت دوم، فرض $C \rightarrow B$ نقض می‌شود. پس فرض خلف باطل است و حکم برقرار است.

وابستگی تابعی- قواعد آرمسترانگ

□ کاربردهای قواعد آرمسترانگ

۱- محاسبه بستار صفت A^+ :

مجموعه تمام صفاتی که با A , وابستگی تابعی دارند.

نکته: اگر $A \Leftarrow A^+ = H_R$ سوپر کلید (الگوریتم تشخیص سوپر کلید و نه کلید کاندید)

۲- محاسبه بستار مجموعه وابستگی‌های تابعی یک رابطه: F^+

مجموعه تمام FD‌هایی که از F منطقاً استنتاج می‌شوند:

$$F = \{A \rightarrow B, B \rightarrow C\} \Rightarrow F^+ = \{A \rightarrow B, B \rightarrow C, A \rightarrow C, (A,C) \rightarrow (B,C), \dots\}$$

مثالی از F^+

- $R = (A, B, C, G, H, I)$
 $F = \{ A \rightarrow B$
 $A \rightarrow C$
 $CG \rightarrow H$
 $CG \rightarrow I$
 $B \rightarrow H \}$
- Some members of F^+
 - $A \rightarrow H$
 - by transitivity from $A \rightarrow B$ and $B \rightarrow H$
 - $AG \rightarrow I$
 - by augmenting $A \rightarrow C$ with G , to get $AG \rightarrow CG$ and then transitivity with $CG \rightarrow I$
 - $CG \rightarrow HI$
 - by augmenting $CG \rightarrow I$ to infer $CG \rightarrow CGI$, and augmenting of $CG \rightarrow H$ to infer $CGI \rightarrow HI$, and then transitivity

کاربرد F^+

□ کاربردهای مهم F^+

۱- تشخیص معادل بودن دو مجموعه از FD‌های رابطه‌ای R : به طور نمونه F و G

□ شرط معادل بودن: $F^+ = G^+$

هر FD که از F به دست آید، از G هم به دست می‌آید.

۲- تشخیص FD افزونه

□ ضابطه تشخیص: وابستگی تابعی $f \in F$ را افزونه گوییم، هرگاه: $(F-f)^+ = F^+$

□ یعنی بود و نبود f در محاسبه F^+ تاثیری نداشته باشد.

کاربرد F^+ : تجزیه خوب

□ تجزیه خوب (Nonloss/Lossless Decomposition)

۱- بی‌حشو: در پیوند پرتوها، تاپل‌حشو [افزونه] بروز نکند.

۲- حافظ FD‌ها: هیچ FD‌ای در اثر تجزیه از دست نرود و همه رابطه اصلی حفظ شوند.

۳- بی‌حذف: در پیوند پرتوها هیچ تاپلی حذف نشود (صفت یا صفات پیوند هیچ‌مقدار نباشند).

۴- حافظ صفات: $\bigcup_{i \in \{1, \dots, n\}} H_{R_i} = H_R$

۱
۲
۳
۴

□ در بیشتر متون کلاسیک، بحث تجزیه خوب، تحت عنوان **تجزیه بی‌کاست یا بی‌گمشدگی**

(Nonloss/Lossless Decomposition) مطرح شده است، که منظور همان بی‌حشو و حافظ وابستگی‌های

تابعی بودن است (و دو ویژگی دیگر تجزیه خوب را پیش‌فرض تجزیه خوب بدانیم).

□ در واقع تاپلهای افزونه باعث از دست رفتن بخشی از اطلاعات می‌شوند.

تجزیه خوب

- تجزیه R به دو رابطه $R1$ و $R2$ در صورتی یک تجزیه خوب است که حداقل یکی از وابستگی های زیر در F^+ باشد.

$$R_1 \cap R_2 \rightarrow R_1$$

$$R_1 \cap R_2 \rightarrow R_2$$

- در واقع صفت مشترک $R1$ و $R2$ باید در یکی از دو رابطه superkey باشد.

الگوريتم ايجاد F^+

- To compute the closure of a set of functional dependencies F :

$$F^+ = F$$

repeat

for each functional dependency f in F^+

 apply reflexivity and augmentation rules on f

 add the resulting functional dependencies to F^+

for each pair of functional dependencies f_1 and f_2 in F^+

 if f_1 and f_2 can be combined using transitivity

then add the resulting functional dependency to F^+

until F^+ does not change any further

الگوريتم ايجاد بستار صفت

- Given a set of attributes α , define the ***closure*** of α **under** F (denoted by α^+) as the set of attributes that are functionally determined by α under F
- Algorithm to compute α^+ , the closure of α under F

```
result :=  $\alpha$ ;  
while (changes to result) do  
  for each  $\beta \rightarrow \gamma$  in  $F$  do  
    begin  
      if  $\beta \subseteq result$  then result := result  $\cup$   $\gamma$   
    end
```

مثالی از بستار صفت

- $R = (A, B, C, G, H, I)$
- $F = \{A \rightarrow B$
 $A \rightarrow C$
 $CG \rightarrow H$
 $CG \rightarrow I$
 $B \rightarrow H\}$
- $(AG)^+$
 1. $result = AG$
 2. $result = ABCG$ ($A \rightarrow C$ and $A \rightarrow B$)
 3. $result = ABCGH$ ($CG \rightarrow H$ and $CG \subseteq AGBC$)
 4. $result = ABCGHI$ ($CG \rightarrow I$ and $CG \subseteq AGBCH$)
- Is AG a candidate key?
 1. Is AG a super key?
 1. Does $AG \rightarrow R? == \text{Is } R \supseteq (AG)^+$
 2. Is any subset of AG a superkey?
 1. Does $A \rightarrow R? == \text{Is } R \supseteq (A)^+$
 2. Does $G \rightarrow R? == \text{Is } R \supseteq (G)^+$
 3. In general: check for each subset of size $n-1$

کاربردهای بستار صفت

- بررسی superkey بودن:
 - برای بررسی superkey بودن یک مجموعه صفت، بستار آن را تشکیل داده و بررسی می کنیم آیا شامل تمام صفات رابطه می شود یا خیر
- برای بررسی روابط تابعی
 - برای بررسی اینکه آیا رابطه تابعی $\beta \rightarrow \alpha$ برقرار است (یا به بیانی دیگر این رابطه در F^+ قرار می گیرد) باید بررسی کنیم که آیا $\beta \subseteq \alpha^+$ می باشد.
- برای محاسبه بستار F
 - به ازای هر $R \subseteq S$ ، $\gamma^+ \subseteq R$ را محاسبه کرده و به ازای هر $S \subseteq R$ برقرار خواهد بود.

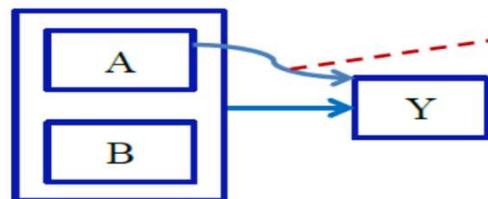
۳- محاسبه مجموعه کاهش ناپذیر FD‌های یک رابطه سه شرط دارد:

۱- هیچ FD در آن افزونه نباشد.

۲- سمت راست هر FD، صفت ساده باشد.

۳- سمت چپ هر FD، خود کاهش ناپذیر باشد: در وابستگی تابعی $X \rightarrow Y$ ، X را کاهش ناپذیر (و وابستگی $Y \rightarrow X$ را **کامل**) گوییم، هرگاه Y با هیچ زیرمجموعه از X (غیر از خود X)، FD نداشته باشد. در غیر اینصورت X را کاهش پذیر گوییم و وابستگی $X \rightarrow Y$ را **ناکامل** گوییم.

اگر وجود داشته باشد، آنگاه X کاهش پذیر و $X \rightarrow Y$ یک FD ناکامل است.



$$\begin{array}{c} X \\ \overbrace{(A, B) \rightarrow Y}^{\text{ناکامل FD}} \\ A \rightarrow Y \end{array}$$

□ **تمرین:** اگر یک FD کامل به صورت $Y \rightarrow A$ داشته باشیم، آنگاه FD ناکامل $Y \rightarrow (A, B)$ از آن قابل استنتاج است.

□ **اثبات:** با استفاده از قاعده افزایش از $Y \rightarrow A$ نتیجه می‌گیریم $(A, B) \rightarrow (Y, B)$ با استفاده از قاعده تجزیه داریم؛ $(A, B) \rightarrow B$ که یک FD بدیهی است و $Y \rightarrow B$ همان حکم است.

تمرين  **وابستگی تابعی با واسطه (TFD):** اگر $B \rightarrow A$ و $B \rightarrow C$ و $A \rightarrow B$ ، می‌گوییم C با A ، FD با واسطه از طریق B دارد.

اگر $B \rightarrow A$ هم برقرار باشد، آنگاه آن FD با واسطه، بدیهی (نامهم) است.

فرم های نرمال کلاسیک کادی

تعریف

1NF: رابطه R در 1NF است اگر و فقط اگر تمام صفات آن تک مقداری باشد.

□ این تعریف می‌گوید هر رابطه نرمال در 1NF است.

تعریف

2NF: رابطه R در 2NF است اگر و فقط اگر در 1NF باشد و هر صفت ناکلید (که خود PK یا CK نباشد و جزء PK یا CK هم نباشد) در آن، با کلید اصلی رابطه، FD کامل داشته باشد.

□ به بیان دیگر در این رابطه FD ناکامل با کلید اصلی نداشته باشیم.

□ الگوریتم تبدیل 1NF به 2NF: حذف FD های ناکامل از طریق تجزیه عمودی رابطه به طور مناسب.

تعریف

3NF: رابطه R در 3NF است اگر و فقط اگر در 2NF باشد و هر صفت ناکلید با کلید اصلی رابطه، فقط FD بی‌واسطه داشته باشد (با واسطه نداشته باشد).

□ الگوریتم تبدیل 2NF به 3NF: حذف FD های با واسطه.

فرم های نرمال کلاسیک کادی (ادامه)

مثالی قید می کنیم و در آن تا 3NF پیش می رویم.



در حالت کلی، تمام صفات دانشجو، درس و انتخاب در یک رابطه می توانند باشند.

قواعد محیط:

R (STID, COID, STJ, STD, GR)

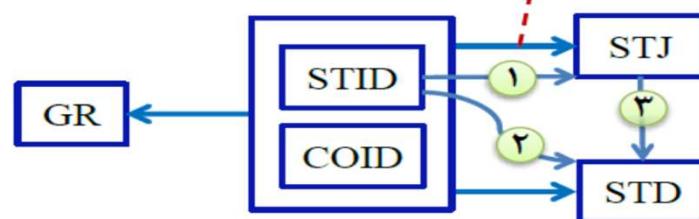
777	CO1	Phys	D11	19
777	CO2	Phys	D11	16
777	CO3	Phys	D11	11
888	CO1	Math	D12	16
888	CO2	Math	D12	18
444	CO1	Math	D12	13
555	CO1	Phys	D11	14
555	CO2	Phys	D11	12

۱- یک دانشجو در یک رشته تحصیل می کند.

۲- یک دانشجو در یک دانشکده تحصیل می کند.

۳- یک رشته در یک دانشکده ارائه می شود.

FD های ناشی از PK (PK چپ سمت)



فرم های نرمال کلاسیک کادی (ادامه)

□ رابطه **R** در 1NF است (چون همه صفات تک مقداری هستند) ولی آنومالی دارد و باید نرمال تر شود.

□ آنومالی های رابطه **R**

۱- در درج:

درج کن این فقره اطلاع درمورد یک دانشجو را: <‘666’, ‘chem’, ‘D16’>

درج ناممکن: تا ندانیم حداقل یک درسی که گرفته شده چیست.

۲- در حذف:

فرض می کنیم ‘444’ در این لحظه فقط همین تک درس را داشته باشد.

حذف کن فقط این اطلاع را: <‘444’, ‘CO1’, ‘13’>

حذف انجام می شود اما اطلاع ناخواسته هم حذف می شود.

۳- در بهنگام سازی:

تغییر رشته تحصیلی دانشجو با شماره 777 به Chem.

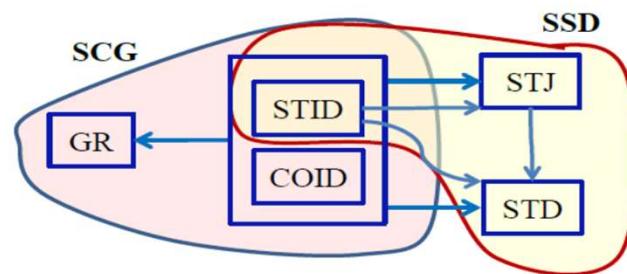
برای انجام آن فزونکاری داریم؛ بهنگام سازی منتشر شونده (Propagating Update).

فرم های نرمال کلاسیک کادی (ادامه)

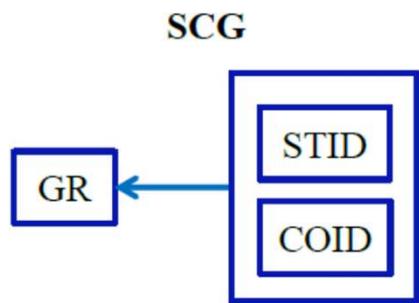
- دلیل آنومالی های رابطه R :
 - از دیدگاه عملی: پدیده اختلال اطلاعات، یعنی اطلاعات در مورد خود موجودیت دانشجو با اطلاعات در مورد انتخاب درس مخلوط شده است.
 - از دیدگاه تئوری: وجود FD های ناکامل
- $$\left[\begin{array}{l} (\text{STID}, \text{COID}) \rightarrow \text{STJ} \\ \text{STID} \rightarrow \text{STJ} \end{array} \right]$$
- $$\left[\begin{array}{l} (\text{STID}, \text{COID}) \rightarrow \text{STD} \\ \text{STID} \rightarrow \text{STD} \end{array} \right]$$
- این FD های ناکامل باید از بین بروند. برای این منظور رابطه R را باید چنان تجزیه عمودی کنیم که در رابطه های حاصل، FD ناکامل نباشد.
- برای این کار از عملگر پرتو استفاده می کنیم. پرتوی که منجر به یک تجزیه خوب شود.

فرم های نرمال کلاسیک کادی (ادامه)

$\Pi_{(STID, COID, GR)}(R)$			$\Pi_{(STID, STJ, STD)}(R)$		
SCG (STID, COID, GR)			SSD (STID, STJ, STD)		
777	CO1	19	777	Phys	D11
777	CO2	16	888	Math	D12
777	CO3	11	444	Math	D12
888	CO1	16	555	Phys	D11
888	CO2	18			
444	CO1	13			
555	CO1	14			
555	CO2	12			



فرم های نرمال کلاسیک کادی (ادامه)



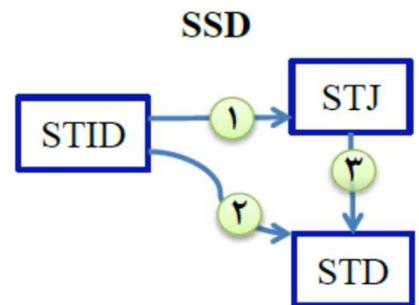
□ رابطه های جدید آنومالی های R را ندارند:

۱- درج کن: $\langle '666', 'chem', 'D16' \rangle$

بدون مشکل در SSD درج می شود.

۲- حذف کن: $\langle '444', 'CO1', 13 \rangle$

بدون مشکل از SCG حذف می شود.



۳- بهنگام سازی کن: تغییر رشته دانشجوی 777 را به Chem

بدون مشکل در SSD بروز می شود.

- در طراحی جدید، FD‌های ناکامل از بین رفته‌اند. بنابراین SSD و SCG 2NF هستند.
- تاكيد:** رابطه R، 2NF است هرگاه اولاً در 1NF باشد و ثانياً هر صفت ناکلید با کلید اصلی، FD کامل داشته باشد (رابطه، FD ناکامل نداشته باشد).
- تمرین:** بررسی شود که آیا در این تجزیه همه FD‌ها محفوظ می‌مانند؟
- نکته:** باید توجه کنیم که در تجزیه، FD‌ای از دست نرود، چون هر FD یک قاعده جامعیت در محیط است.
- توجه داشته باشید که در این تجزیه هیچ اطلاعی از دست نمی‌رود. یعنی اگر کاربر رابطه اصلی را به هر دلیلی بخواهد با پیوند دو رابطه جدید به دست می‌آید.
$$R = SCG \bowtie SSD$$

□ در حالت کلی اگر R_1, R_2, \dots, R_n پرتوهای دلخواه از R باشند، به شرط عدم وجود هیچمقدار داریم (ممکن است تاپل‌های افزونه بروز کند):

$$R \subseteq R_1 \bowtie R_2 \bowtie \dots \bowtie R_n$$

□ **تجزیه بی‌حذف:** شرطش این است که در صفات پیوند هیچمقدار (Null Value) نداشته باشیم.
□ اگر در صفات پیوند هیچمقدار داشته باشیم، چه پیش می‌آید؟

$$T(\underline{A}, B, C, D, E) \Rightarrow T_1(A, B) \quad T_2(B, C, D, E)$$

تاپل‌هایی در پیوند از دست می‌روند. به این تاپل‌ها، تاپل‌های آونگان [معلق] (Dangling) گوییم.

□ در مباحث نرمالترسازی معمولاً فرض بر این است که **صفت (صفات)** پیوند هیچمقدار ندارند.

آیا رابطه‌های جدید (SSD و SCG) آنومالی ندارند؟

آنومالی‌های SSD

۱- در درج:

اطلاع: «رشته IT در دانشکده D20 ارائه می‌شود.» به دلیل FD شماره ۳، این اطلاع منطقاً باید قابل درج باشد، اما درج ناممکن است. چون کلید ندارد، باید حداقل یک دانشجوی این رشته را بشناسیم.

۲- در حذف:

حذف کن <Chem'، '666'> و با فرض اینکه تنها یک دانشجو در رشته Chem ثبت شده است.

حذف انجام می‌شود ولی اطلاع «رشته شیمی در D16 ارائه می‌شود»، ناخواسته حذف می‌شود.

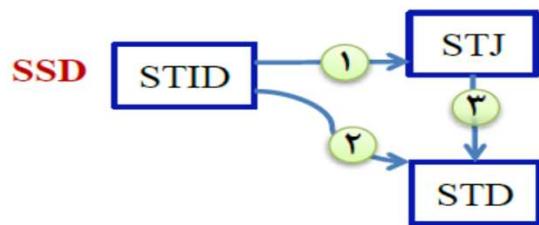
۳- در بهنگام‌سازی:

«شماره دانشکده رشته فیزیک را عوض کنید». به تعداد تمام دانشجویان این رشته باید بهنگام‌سازی شود.
SSD باید نرمال‌تر شود.



دلیل آنومالی‌های SSD

دلیل آنومالی‌های SSD، وجود FD با واسطه بین صفت ناکلید با کلید اصلی است (به دلیل FD شماره ۳).



این FD باید از بین برود.

فرض کنید SSD را به صورت زیر تجزیه کنیم:

SJ (STID, STJ) و SD (STJ, STD)

777	Phys	Phys	D11
888	Math	Math	D12
444	Math		
555	Phys		

افزونگی کم شد!

تمرین: بررسی شود که رابطه‌های جدید آنومالی‌های SSD را ندارند.

□ این رابطه‌ها در 3NF هستند.



□ اولاً در 2NF هستند.



□ ثانیاً FD با واسطه نداریم.

□ تمرین: بررسی شود که در این تجزیه هیچ اطلاعی از دست نمی‌رود و FD‌ها هم حفظ می‌شوند.

□ تاکید: رابطه R در 3NF است اگر و فقط اگر اولاً در 2NF باشد و ثانیاً هر صفت ناکلید با کلید اصلی FD

بی‌واسطه داشته باشد (تمام FD‌ها مستقیماً ناشی از PK باشد).

□ نتیجه: FD‌های ناکامل و با واسطه مزاحم هستند و باید از بین بروند.

[بحث تکمیلی] تجزیه خوب

□ تجزیه خوب (Nonloss/Lossless Decomposition)

۱- بی‌حشو: در پیوند پرتوها، تاپل‌حشو [افزونه] بروز نکند.

۲- حافظ FD‌ها: هیچ FD‌ای در اثر تجزیه از دست نرود و همه رابطه اصلی حفظ شوند.

۳- بی‌حذف: در پیوند پرتوها هیچ تاپلی حذف نشود (صفت یا صفات پیوند هیچ‌مدار نباشند).

۴- حافظ صفات: $\bigcup_{i \in \{1, \dots, n\}} H_{R_i} = H_R$

۱
۲
۳
۴

□ در بیشتر متون کلاسیک، بحث تجزیه خوب، تحت عنوان تجزیه بی‌کاست یا بی‌گمشدگی

(Nonloss/Lossless Decomposition) مطرح شده است، که منظور همان بی‌حشو و حافظ وابستگی‌های

تابعی بودن است (و دو ویژگی دیگر تجزیه خوب را پیش‌فرض تجزیه خوب بدانیم).

□ در واقع تاپلهای افزونه باعث از دست رفتن بخشی از اطلاعات می‌شوند.

قضیه ریسانن

□ قضیه ریسانن (Rissanen) :

□ رابطه R به دو پرتوش (R_1 و R_2) تجزیه خوب می‌شود، اگر R_1 و R_2 از یکدیگر مستقل باشند.

□ R_1 و R_2 مستقل از یکدیگرند اگر و فقط اگر:

- صفت مشترک، حداقل در یکی از آنها CK باشد \Leftarrow بی‌حشو بودن

- تمام FD‌های رابطه اصلی یا در مجموعه FD‌های R_1 و R_2 وجود داشته باشند یا از آنها منطقاً

استنتاج شوند \Leftarrow حافظ FD‌ها

□ نکته: بر اساس ضوابط ریسانن، اگر در رابطه $R(A, B, C)$ ، $A \rightarrow B$ و $A \rightarrow C$ برقرار

باشد، در اینصورت تجزیه خوب چنین است: $R_2(B, C)$ و $R_1(A, B)$

□ در اینجا B در رابطه دوم کلید کاندید است، چون همه صفات به آن وابستگی تابعی دارند و کاهش‌پذیر هم نیست.

□ مثال: رابطه SSD را در نظر می‌گیریم. این رابطه به سه شکل به پرتوهای دوگانی قابل تجزیه است.

- I SS (STID, STJ) SD (STJ, STD)
- II SS (STID, STJ) SD (STID, STD)
- III SS(STID, STD) SJ (STJ, STD)

□ تجزیه I خوب است، چون هر دو شرط ریسانین را دارد.
$$\left. \begin{array}{l} STID \rightarrow STJ \\ STJ \rightarrow STD \end{array} \right\} \Rightarrow STID \rightarrow STD$$

□ تجزیه II خوب نیست، چون FD از دست می‌دهد.

□ تجزیه III خوب نیست، چون FD از دست می‌دهد.

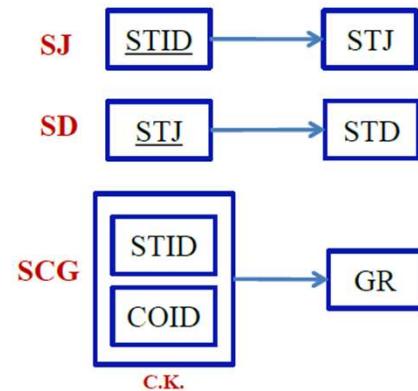
فرم نرمال BCNF

□ **اصطلاح:** در وابستگی تابعی $A \rightarrow B$ (A Determines B) به A دترمینان گویند.

□ **BCNF:** رابطه R در BCNF است اگر و فقط اگر در آن دترمینان هر FD مهم و کاهش ناپذیر، CK باشد.
در 3NF، تنها باید دترمینان رابطه PK باشد.

□ چون رابطه می‌تواند بیش از یک CK داشته باشد، BCNF از 3NF قوی‌تر است.

□ رابطه‌های زیر در BCNF هستند.



SCGJD
SCGJD $\left\{ \begin{array}{l} \text{SCG}(\underline{\text{SID}}, \underline{\text{COID}}, \text{GR}) \\ \text{SJ } (\underline{\text{STID}}, \text{STJ}) \\ \text{SD } (\underline{\text{STJ}}, \text{STD}) \end{array} \right.$

از 3NF قوی‌تر است. \Leftarrow رابطه می‌تواند در 3NF باشد، اما در BCNF نباشد. \square

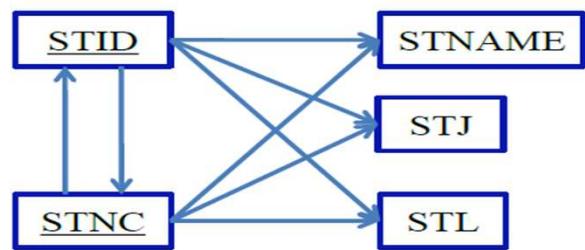
حالت I: رابطه R فقط یک CK داشته باشد. \Leftarrow اگر R در 3NF باشد، در BCNF هم هست (مثال دیده شده).

حالت II: رابطه R بیش از یک CK داشته باشد. \square

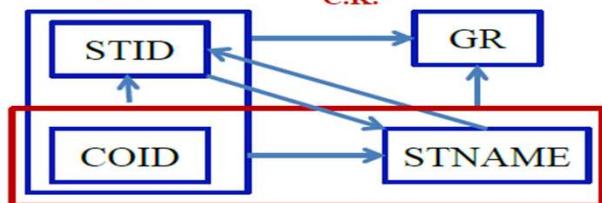
(1-II) CK‌ها مجزا باشند (صفت مشترک نداشته باشند). \Leftarrow اگر R در 3NF باشد، در BCNF هم هست.

(2-II) CK‌ها همپوشانند. \Leftarrow اگر R در 3NF باشد، لزوماً در BCNF نیست. \square

ST (STID, STNAME, STNC, STJ, STL, ...)



SCNG (STID, COID, STNAME, GR)



برای حالت 1-II
دو دترمینان، هر دو هم CK هستند.



(فرض: هیچ دو دانشجویی نام یکسان ندارند.)

برای حالت 2-II

