

فصل ۹ : Normalization

مدرس : اسماعیل نورانی

<http://www.nurani.ir/> - Info@nurani.ir

Further Normalization

نرمالترسازی رابطه (ابزار طراحی پایگاه رابطه ای ؛ آشنایی با تئوری وابستگی)

ایده اصلی : رابطه هر چند نرمال ، ممکن است بازهم آنومالیهایی (Anomalies) داشته باشد .

یادآوری : رابطه نرمال رابطه ای است که مقادیر تمام صفات خاصه اش اتمیک باشند .

- (*) عدم امکان انجام یک عمل
- (*) بروز تبعات نامطلوب روی انجام یک عمل
- (*) بروز افزونگی در سیستم در انجام یک عمل

مثال: رابطه SPC مفروض است.
 در درج: این رابطه در درج آنومالی دارد. درج کن این اطلاع را <S_۲, C_۲>

درج ناممکن است تا ندانیم چه قطعه ای تهیه کرده است زیرا کلید اصلی رابطه SPC، (S# و P#) است و درج تاپل بدون داشتن آنها ناممکن است.

SPC (S# P# QTY CITY)

S _۱	P _۱	۱۰۰	C _۲
S _۱	P _۲	۲۰۰	C _۲
S _۱	P _۳	۸۰	C _۲
S _۲	P _۱	۷۰	C _۱
S _۲	P _۲	۱۲۰	C _۱
S _۳	P _۱	۹۰	C _۱
S _۴	P _۱	۸۰	C _۳
S _۴	P _۳	۹۰	C _۳

در حذف:

حذف کن <S_۳, P_۱, ۹۰>. انجام شدنی است،
 اما اطلاع ناخواسته حذف می شود (اینکه S_۳ ساکن C_۱ است).

نرمال بودن (نرمالیتی) درجات یا سطوح یا صور دارد.

(First Normal Form)

۱NF

۲NF

۳NF

CODD در آغاز سه صورت تعریف کرده
 است که به آنها صور کلاسیک کادی می
 گوئیم

(Boyce/Codd Normal Form) BCNF

۴NF

(Projection-Join Normal Form) (PJ/NP)

۵NF

(Domain-Key Normal Form) DKNF

صور دیگر

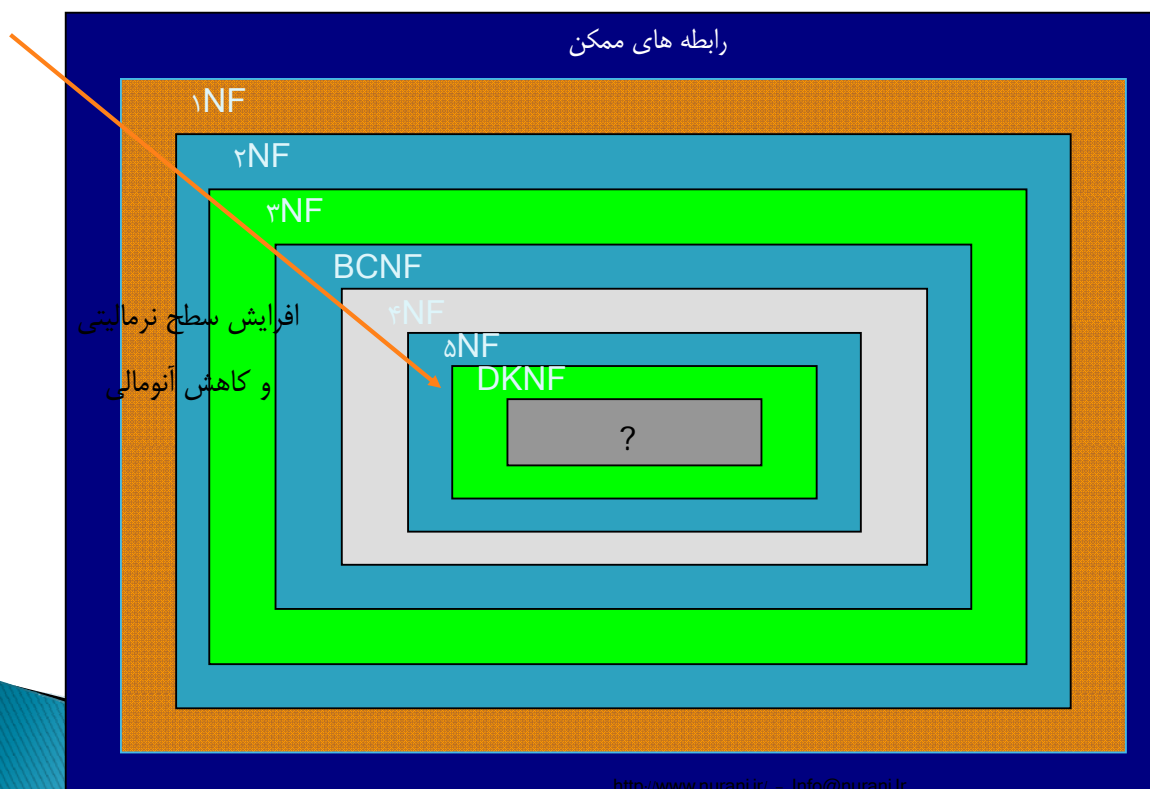
در بهنگام سازی : شهر S۱ را عوض کن .

```
UPDATE S  
SET CITY = 'C۱'  
WHERE S# = 'S۱' ;
```

عمل منطقی تاپلی تبدیل به عمل منطقی مجموعه ای و منجر به بروز فزونکاری در سیستم می شود. (چند سطر عوض میشود)

رابطه SPC آنومالی دارد . دلیل غیر تئوریک این آنومالی اینست که در رابطه SPC پدیده اختلاط اطلاعاتی وجود دارد. یعنی اطلاعات در مورد شی یا موجودیت محصول با اطلاعات در مورد تهیه کننده مخلوط شده است . شهر از صفات خاصه تهیه کننده است و با صفات خاصه محصول ترکیب شده است ، همین اختلاط اطلاعاتی سبب بروز پدیده افزونگی هم شده است .
رابطه SPC خوش طرح (well design) نیست و خواهیم دید که این طراحی باید عوض شود .

نمودار صورت مختلف نرمال بودن :



نکته : هرچه رابطه نرمالتر باشد آنومالی کمتر می شود.

شرح صور نرمالیتی :

ابتدا سطوح کلاسیک کادی را بررسی می کنیم و برای این منظور نیاز به دو مفهوم از

تئوری وابستگی داریم :

(۱) وابستگی تابعی - Functional Dependency (FD)

صفت خاصه Y به صفت خاصه X از رابطه R وابستگی تابعی دارد اگر به ازای هر مقدار از X در R فقط و فقط یک مقدار برای Y وجود داشته باشد .

$$R.X \longrightarrow R.Y$$

R	X	Y	Z
	X_1	Y_1	Z_1
	X_1	Y_1	Z_2
	X_2	Y_1	Z_2
	X_2	Y_1	Z_3
	X_1	Y_2	Z_3

مثال : رابطه R مفروض است :

آیا $X \rightarrow Y$ ؟ بله

آیا $Y \rightarrow X$ ؟ خیر

آیا $X \rightarrow Z$ ؟ خیر

مثال : رابطه S را در نظر می گیریم

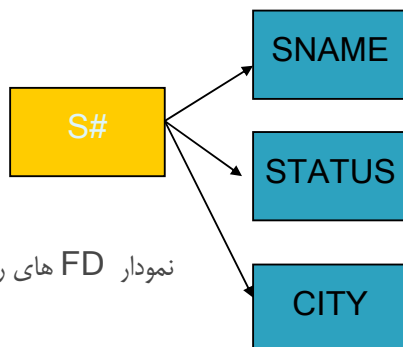
$S\#$ کلید کاندید است .

$S(S\#, SNAME, STATUS, CITY)$

$S\# \rightarrow SNAME$

$S\# \rightarrow STATUS$

$S\# \rightarrow CITY$



نمودار FD های رابطه S

چون $S\#$ کلید کاندید است این وابستگیها محرز است .

نتیجه : تمام صفات خاصه یک رابطه در طول حیات آن با کلید کاندید آن FD دارند .

نکته: FD ها در (محیط عملیاتی) **تفسیر** دارند . در واقع بیانگر قواعد معنایی محیط عملیاتی هستند (قوانین Semantic ناظر بر محیط عملیاتی) .

$S\# \rightarrow CITY$

قاعده Semantic : هر تهیه کننده در یک شهر دفتر دارد .

$S\# \rightarrow STATUS$

قاعده Semantic : هر تهیه کننده یک مقدار وضعیت دارد .

مثال : در یک محیط آموزشی :

$PR\# \rightarrow CO\#$

۱- یعنی یک استاد فقط یک درس را تدریس می کند .

۲- یک درس توسط اساتید مختلف تدریس می شود.

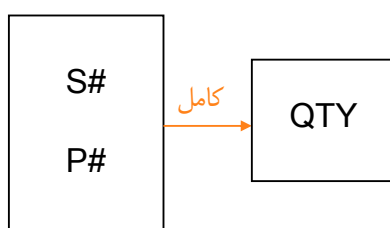
$CO\# \not\rightarrow PR\#$

نکته: DBMS باید امکان بدهد تا مجموعه FD های محیط معرفی شوند ، بعنوان مجموعه قواعد Semantic ناظر بر محیط که خود نوعی قواعد جامعیتی هستند (**Dependency Integrity Rules**) .
 عملاً این قواعد در یک طراحی خوب از طریق مفهوم کلید اصلی و کلید کاندید به یک سیستم انتقال می یابد.

full \rightarrow (۲) مفهوم "وابستگی تابعی کامل (FFD)"

صفت خاصه Y از رابطه R به صفت خاصه مرکب X از R وابستگی تابعی کامل دارد
 با نمایش $R.X \Rightarrow R.Y$ اگر و فقط اگر Y با X وابستگی تابعی داشته باشد اما با هیچ یک از اجزای تشکیل دهنده X وابستگی تابعی نداشته باشد .

مثال : در رابطه $SP(S\#, P\#, QTY)$ ، ($S\#$ و $P\#$) کلید اصلی است .

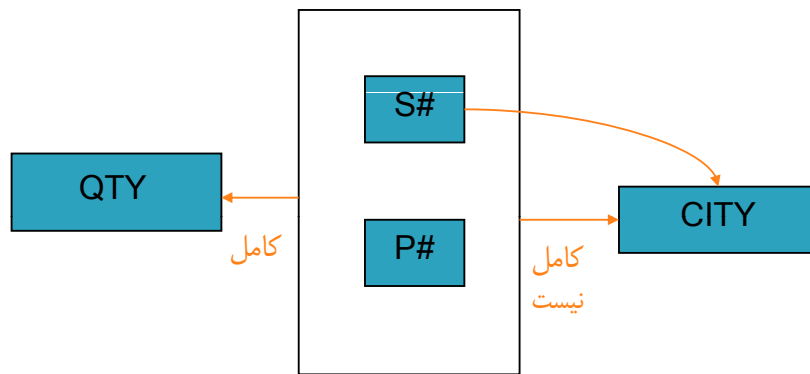


$(S\#, P\#) \rightarrow QTY$

$P\# \not\rightarrow QTY$

$S\# \not\rightarrow QTY$

SPC (S# , P# , QTY , CITY)



مشخص است که اگر سمت چپ FD کامل ، صفت خاصه ساده باشد همان FD معمولی است و کامل نبودن موضوعیت نخواهد داشت .

سطوح کلاسیک کادی :

صورت اول : 1NF

تعریف : رابطه ای 1NF است اگر تمام صفات خاصه اش اتمیک باشد این تعریف صرفاً می گوید هر رابطه نرمال 1NF است .

مثال : First (S# P# QTY CITY STATUS)

S _۱	P _۱	۱۰۰	C _۱	۱۰
S _۱	P _۲	۸۰	C _۱	۱۰
S _۲	P _۳	۹۰	C _۲	۱۵
S _۲	P _۱	۶۰	C _۲	۱۵
S _۲	P _۲	۸۰	C _۲	۱۵
S _۳	P _۱	۷۰	C _۲	۱۵
S _۴	P _۱	۱۲۰	C _۱	۱۰

در این رابطه علاوه بر FD های زیر :

$(S\#, P\#) \rightarrow QTY$
 $(S\#, P\#) \rightarrow CITY$
 $(S\#, P\#) \rightarrow STATUS$ } ناشی از محیط

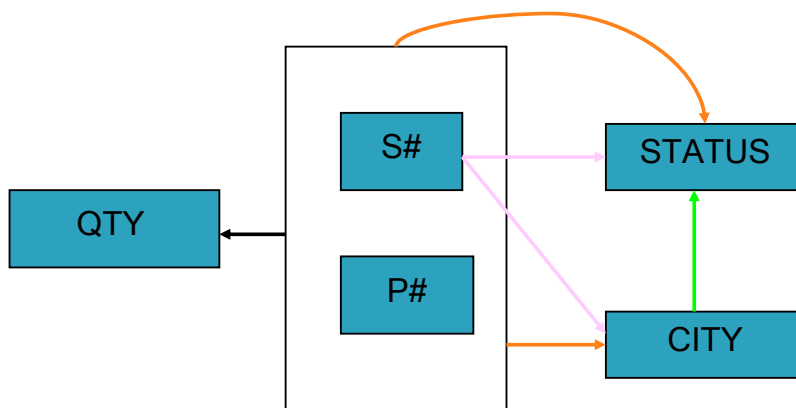
$S\# \rightarrow CITY$
 $S\# \rightarrow STATUS$ } قاعده Semantic محیط

یک FD دیگر نیز اضافه میکنیم برای ادامه مثال :

CITY \rightarrow STATUS

تمام تهیه کنندگان یک شهر یک وضعیت دارند.

نمودار FD ها چنین است .



رابطه FIRST ، NF است و مثل SPC آنومالی دارد .

۱. در درج : درج کن $\langle S_5, C_3, 15 \rangle$ * ناممکن تا P# را ندانیم
۲. در حذف : حذف کن $\langle S_3, P_1, 70 \rangle$ * اگر S₃ فقط همین یک قطعه را تهیه کرده باشد ، اطلاع ناخواسته حذف می شود
۳. در بروز رسانی : شهر S₁ را عوض کن : * سیستم با مشکل بهنگام سازی منتشر شونده مواجه است (فزونکاری در سیستم)

Decompose

پس رابطه FIRST از نظر طراحی رابطه خوبی نیست و باید تجزیه شود. فعلا می پذیریم که FIRST تبدیل یا تجزیه شود به دو رابطه زیر :

SECOND (S#, CITY , STATUS)

SP (S# , P# , QTY)

در واقع این دو رابطه حاصل دو PROJECT از رابطه First هستند ، اولی حاصل PROJECT از FIRST روی S# و CITY و STATUS است و دومی حاصل PROJECT روی S# و P# و QTY است .
(اینکجه چرا چنین باید PROJECT شد را بعدا می بینیم اما مکانیسم تجزیه PROJECT است . در هرمرحله PROJECT به دو رابطه)

یادآوری : PROJECT جبری تکراری ها را حذف می کند .

SECOND (S# , CITY , STATUS)

S#	CITY	STATUS
S ₁	C ₁	۱۰
S ₂	C ₂	۱۵
S ₃	C ₂	۱۵
S ₄	C ₁	۱۰

SP (S# , P# , QTY)

S ₁	P ₁	۱۰۰
S ₁	P ₂	۹۰
S ₂	P ₃	۹۰
S ₂	P ₁	۶۰
S ₂	P ₂	۸۰
S ₃	P ₁	۷۰
S ₄	P ₁	۱۲۰

SECOND و SP آنومیهای FIRST را ندارند . (اما ممکن است

خودشان آنومیهایی دیگر داشته باشند.)

بررسی :

۱- درج کن : $\langle S_5, C_3, 13 \rangle$ منجر می شود به درج یک تاپل مشخص در SECOND و لازم نیست بدانیم چه قطعه ای تهیه کرده است.

۲- حذف کن : $\langle S_3, P_1, 70 \rangle$

منجر می شود به یک حذف و اطلاع ناخواسته ای از بین نمی رود .

۳- شهر S₁ را عوض کن :

S# در SECOND کلید است ، پس تنها یک تاپل با S#='S₁' داریم

و بهنگام سازی منتشر شونده نداریم .

نتیجه: رفتار DBMS در پایگاه جدید برای انجام عملیات بهتر است از پایگاه قبلی زیرا آنومالیها کمتر است به بیان دیگر SECOND و SP از FIRST نرمالترند. می‌گوییم این دو رابطه **2NF** اند.

مبنای تئوریک تجزیه: (صرفنظر از دلیل عملی که جلوگیری از اختلاط اطلاعاتی است).

باید دلیل تئوریک آنومالیهای FIRST را بیابیم. نمودار FD های FIRST نشان می‌دهد که در FIRST دو FD ناکامل داریم.

$(S\#, P\#) \rightarrow CITY$

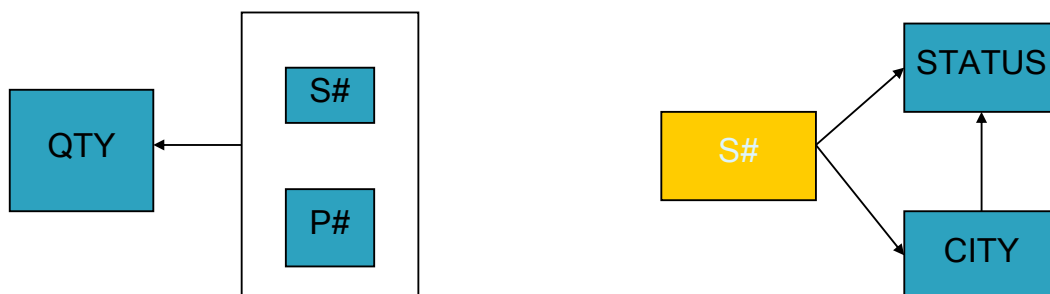
$S\# \rightarrow CITY$

$(S\#, P\#) \rightarrow STATUS$

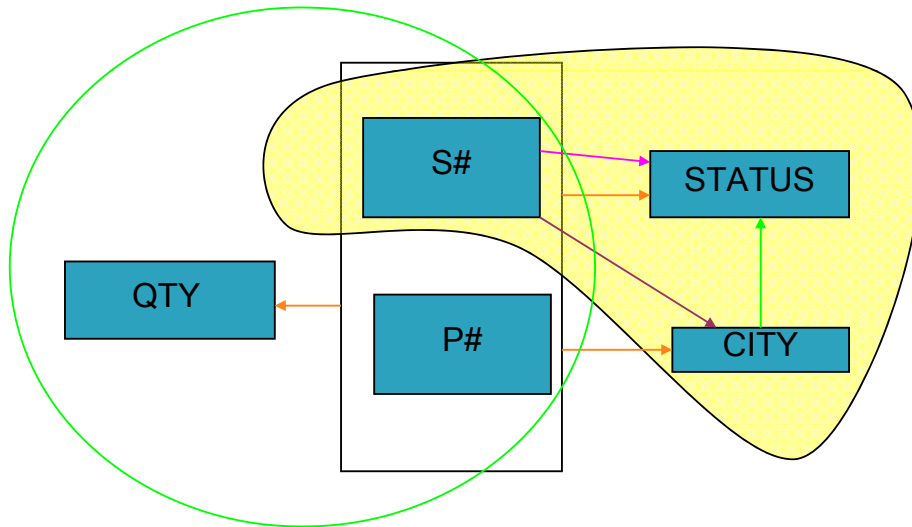
$S\# \rightarrow STATUS$

این دو FD ناکامل ریشه آنومالیهای FIRST هستند. بنابراین رابطه FIRST را با پرتوگیری به دو رابطه چنان تجزیه می‌کنیم (به این شرط که مجموعه HEADING دو رابطه همان HEADING ، FIRST باشد) که در رابطه های حاصله FD ناکامل وجود نداشته باشد.

نمودار FD های SECOND و SP به شرح زیر است:



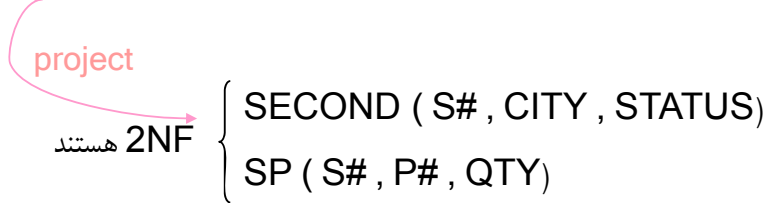
این تجزیه را می توان از روی نمودار FD های FIRST بطور گرافیکی انجام داد .



نکته : رابط FIRST را می توان با پیوند دو رابطه SP و SECOND بدست آورد . بی آنکه اطلاعاتی از دست برود و یا اطلاعات زاید(حشو) در اثر پیوند بروز کند. همانطور که SP و SECOND حاصل تجزیه FIRST هستند ، FIRST نیز حاصل پیوند SP و SECOND است . (محال است در اثر این پیوند تاپلی حذف یا اضافه شود چون S# کلید اصلی است .) در واقع این سوال مطرح می شود که اگر رابطه R را از طریق پرتوگیری (به دو رابطه R_1, R_2 تجزیه کنیم آیا همیشه $R_1 \text{ JOIN } R_2 = R$ است ؟ ؟

به دلیل اینکه صفت مشترک حداقل در یکی از دو رابطه کلید اصلی است
تاپل حذف شده و تاپل حشو نداریم .

FIRST (S#,P#,CITY,STATUS,QTY)



FIRST = SECOND JOIN SP

پک سوال : اگر رابطه R را با استفاده از PROJECT به دو رابطه R1 و R2 تجزیه کنیم
آیا همیشه $R = R1 JOIN R2$ ؟ خیر . لزوما همیشه خیر

با ذکر مثال نقض :

SPJ (S# , P# , J#)

رابطه تمام کلید

S1 P1 J2
S1 P2 J1
S2 P1 J1
S1 P1 J1

تجزیه به دو رابطه →

SP (S# , P#)

PJ (P# , J#)

S1 P1 P1 J2
S1 P2 P2 J1
S2 P1 P1 J1

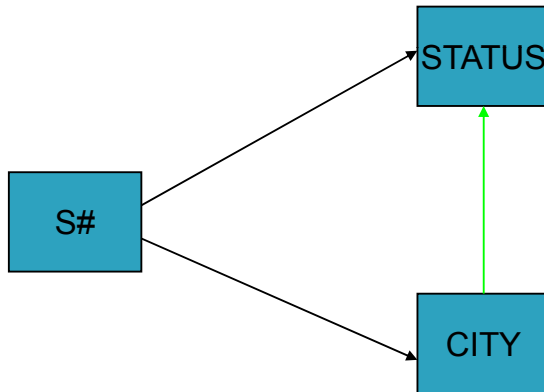
SP JOIN PJ = (S# , P# , J#)

S1 P1 J2
S1 P1 J1
S1 P2 J1
S2 P1 J2
S2 P1 J1

تاپل حشو

SECOND (S# , CITY , STATUS)

سطح نرمالیتی Second از FIRST بالاتر است و 2NF است



تعریف رابطه 2NF

رابطه ای 2NF است که اولاً 1NF باشد ثانياً هر صفت خاصه غیر کلید با کلید اصلی رابطه FD کامل داشته باشد. (در آن FD ناکامل وجود نداشته باشد.)

رابطه SECOND یک 2NF است اما آنومالیهای ظریفی دارد (البته آنومالیهای First را ندارد)

S#	City	Status
S1	C1	15
S2	C2	18
S3	C2	18
S4	C1	15
S5	C5	20

فرض کنیم تنها تهیه کنند ساکن در C5 باشد

آنرمالیهای Second

(1) در درج :

درج کن <C3,18>

این درج نا ممکن است زیرا کلید را نداریم . تا ندانیم چه تهیه کننده ای در C3 ساکن است نمیتوانیم این درج را انجام دهیم .

(2) در حذف :

حذف کن <S5,C5>

اطلاع ناخواسته "وضعیت داده شده به شهر C5، 20 است" از بین می رود.

بروز این آنومالی به دلیل یکتایی C5 است

(3) در بهنگام سازی :

Second هنوز افزونگی دارد . مثلا "وضعیت داده شده به شهر C1 را عوض کنید" سیستم باید به ازای تمام تهیه کنندگان ساکن C1 عملیات را انجام دهد .

پس بهنگام سازی منتشر شونده داریم ، یعنی عمل منطقا تاپلی تبدیل به عمل مجموعه ای میشود و موجب فزونکاری در سیستم میشود که خود نوعی آنومالی است و Second, 2NF است نه بیشتر و Second باید با پرتوگیری مناسب ، تجزیه شود .

مثلا فرض کنیم که Second به دو رابطه تجزیه شود :

SC (S# , City)

S1	C1
S2	C2
S3	C2
S4	C1
S5	C5

CS (City , Status)

C1	15
C2	18
C5	20

رابطه های CS,SC آنومالیهای Second را نداند .

1) در درج :

درج کن <C3,18>

منجر به درج یک تاپل مشخص در CS میشود (با کلید City) و مشکلی بروز نمیکند .

2) در حذف :

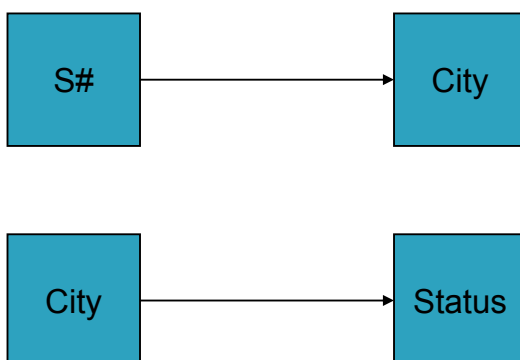
حذف کن <S5,C5>

منجر به حذف یک تاپل مشخص در SC میشود و اطلاع ناخواسته حذف نمیشود .

3) در بهنگام سازی : وضعیت شهر C1 را تغییر دهید . بهنگام سازی منتشر شونده نداریم چون City در CS کلید است ؛ عمل منطقی تاپلی کماکان تاپلی انجام میشود

- محرز است که SC , CS از Second نرمالترند ، میگوییم آنها 3NF هستند .

نموادار FD ها ی پایگاه در وضعیت جدید چنین است :



ریشه آنرمالیهای Second در چیست ؟
وجود نوع خاصی از وابستگی ، موسوم به وابستگی با واسطه (Transitive) است .

در واقع وابستگی  باعث بروز این وابستگی با واسطه شده است .

اگر $R(A,B,C,...)$ و اگر داشته باشیم $A \rightarrow B \rightarrow C$ میگوییم A به C از طریق B وابسته است ، مثلا در Second داریم :

$S\# \rightarrow City \rightarrow Status$

میبینیم که $Status$ علاوه بر اینکه بطور مستقیم با $S\#$ رابطه دارد به دلیل $City \rightarrow Status$ بطور Transitive با $S\#$ از طریق $City$ رابطه دارد و این وابستگی بدلیل وابستگی $City$ به $Status$ است . بنابراین Second را با پرتوگیری مناسب به دو رابطه چنان تجزیه میکنیم که $Transitivity$ از بین برود . حاصل دو رابطه SC و CS می باشد که $3NF$ هستند .

تعریف رابطه 3NF :

رابطه ای $3NF$ است اگر اولاً $2NF$ باشد و ثانيا هر صفت خاصه غیر کلید با کلید اصلی وابستگی بی واسطه داشته باشد ($Transitivity$ وجود نداشته باشد)
بر این اساس $3NF$ ، SC ، CS ، SP هستند .

: BCNF

تعریف :

در رابطه $R : X \rightarrow Y$ (X با y FD دارد یا X ، Y را تعیین میکند یا دترمینان میکند) .

رابطه ای $BCNF$ است اگر در آن ، هر دترمینان کلید کاندید باشد . (یعنی صفات خاصه سمت چپ باید کلید کاندید باشند)

$BCNF$ از $3NF$ نرمالتر است .

مثال : روابط SC ، CS هر دو $BCNF$ هستند :



دترمینان و تنها کلید کاندید

دترمینان و تنها کلید کاندید

مثال : BCNF ، Second ، City : دترمینان است اما کلید کاندید رابطه نیست .

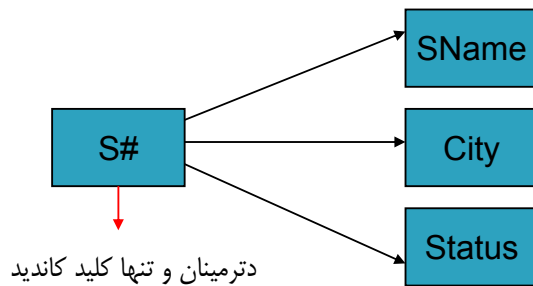
مثال : SP ، BCNF است : $(S\# , p\#) \rightarrow QTY$
دترمینان و تنها کلید کاندید

آیا هر رابطه 3NF ، BCNF است ؟

برای پاسخگویی به این سوال مهم دو حالت در نظر میگیریم :

(۱) رابطه با یک کلید کاندید :

آن کلید کاندید همان کلید اصلی است و هر رابطه 3NF در این حالت BCNF هم هست (تنها دترمینان همان کلید اصلی است)



مثال :
S (S#,.....)
P(P#,.....)

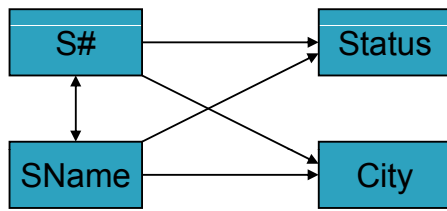
S 3NF و BCNF است . همچنین P .

(۲) رابطه با بیش از یک کلید کاندید :

خود دو حالت دارد :
کلیدهای کاندید ، صفت خاصه مشترک نداشته باشند .
کلیدهای کاندید ، صفت خاصه مشترک داشته باشند . (Overlapping)

اگر در رابطه R ، (X,Y) و (Y,Z) کلیدهای کاندید باشند در این حالت میگوییم Overlapping دارند .

2.1) فرض می کنیم در رابطه S ، Sname هم خاصیت کلید بودن را داشته باشد



$S (\underline{S\#} , \underline{Sname} , City , Status)$

دو کلید کاندید مجزا

این رابطه به وضوح BCNF است (دو ترمینان که هر دو کلید کاندیدند)

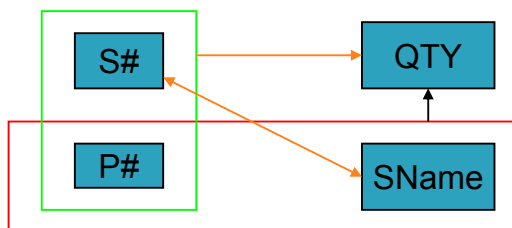
- این رابطه 3NF هم هست (چون BCNF است)
- در ضمن : 1NF هست . 2NF است (FD نا کامل نداریم)
- 3NF هم هست چون Transitivity نداریم :

$S\# \leftrightarrow SName \rightarrow City$

به دلیل وجود این پدیده Transitivity نداریم

2.2) در چنین حالتی است که اگر رابطه 3NF باشد لزوماً BCNF نمیشود

مثال : $SPS(\underline{S\#}, \underline{P\#}, Sname, QTY)$

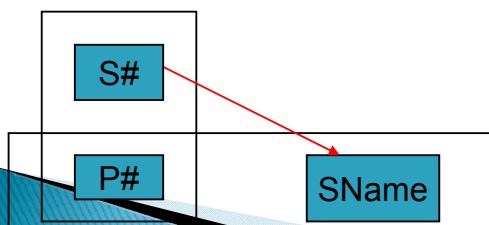


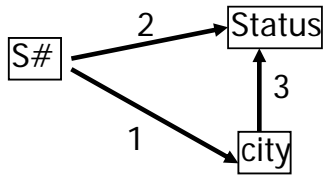
این رابطه قطعاً BCNF نیست

- S# در ترمینان است اما کلید کاندید نیست .

حال در سطوح کادی بررسی میکنیم :

1NF است (صفات خاصه اش اتومیک اند) علی رغم وجود FD ظاهراً نا کامل زیر این رابطه 2NF است زیرا این FD به علت آنکه S# خود جزئی از کلید کاندید است نا کامل به معنایی که در تعریف 2NF گفته شد نیست . زیرا در تعریف 2NF گفته می شود " هر صفت خاصه غیر کلید " و S# غیر کلید نیست و جزئی از کلید کاندید است و 2NF است (چون Transitivity نداریم)





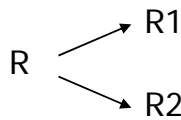
Second (S#, City, Status)

سه تجزیه متصور است :

$$A \begin{cases} SC(S\#, city) \\ CS(city, Status) \end{cases}$$

$$B \begin{cases} SC(S\#, city) \\ SS(S\#, Status) \end{cases}$$

$$C \begin{cases} SS(S\#, Status) \\ CS(city, Status) \end{cases}$$



- تجزیه ای خوب است (تجزیه فاقد گمشدگی) اگر عناصر آن از یکدیگر مستقل باشند

(Nonloss decomposition)

(lossless decomposition)

اگر از یکدیگر مستقل باشند تجزیه خوب است <http://www.nurani.ir/> Info@nurani.ir

- مستقل بودن :

توضیح غیر فرمال :

عملیات ذخیره سازی در یکی روی دیگری تاثیر نداشته باشد.

توضیح فرمال :

قضیه ریسانن : اگر رابطه R را به دو رابطه R1 و R2 تجزیه کنیم، R1 و R2 مستقل از یکدیگرند اگر شرایط زیر برقرار باشد :

1. صفت خاصه مشترک در دو رابطه حداقل در یکی از آنها کلید کاندید باشد.

2. تمام FD های موجود در R در R1 و R2 وجود داشته باشند و یا از FD های موجود در R1 و R2 منطقی قابل استنتاج باشد.

• شرط اول تضمین میکند که در اثر پیوند R1 و R2 تاپل حشو بروز نمیکند طبق خاصیت کلید که تاپل تکراری نمیپذیرد

• خاصیت دوم تضمین میکند که FD ای حذف نشود چون FD ها نشان دهنده قواعد Semantic هستند و نباید از بین بروند و اگر چنین شود تجزیه با گمشدگی خواهد بود.

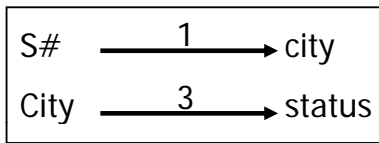
برای تجزیه های Second براساس شرایط قضیه ریسانس :

- شرط اول را دارد

چون city در رابطه دوم کلید کاندید است.

A

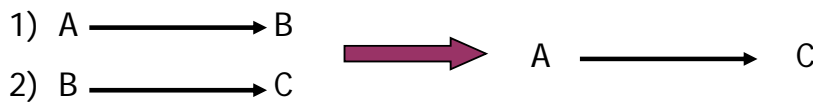
مجموعه حداقل FD های پایگاه



داریم :



بر اساس یکی قواعد آرمسترانگ موسوم به قاعده تعدی که میگوید :



منطقا نتیجه میگیریم :



اثبات قضیه تعدی :

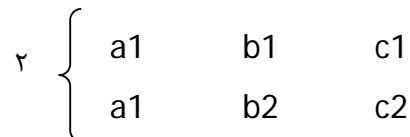
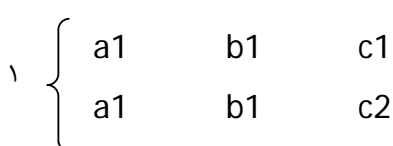
از برهان خلف استفاده میکنیم : داریم : $a \rightarrow b$, $b \rightarrow$

فرض میکنیم که $A \not\rightarrow C$ به اینصورت حداقل دوتا پل در رابطه به اینصورت داریم :

a1-----c1

a1-----c2

وضعیت B در این تاپیهها :



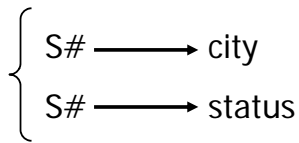
در حالت اول: فرض دوم نقض شده است. ($B \rightarrow C$)

در حالت دوم: فرض اول نقض شده است. ($A \rightarrow B$)

بنابراین $S\# \rightarrow Status$ منطقا وجود دارد و نیازی به تصریح آن نیست.

بنابراین تجزیه A خوب است.

B



- شرط اول را به خوبی دارد
- شرط دوم را ندارد یعنی FD از دست می‌دهیم زیرا از دو FD
منطقاً نمیتوان استنتاج کرد: $city \longrightarrow status$
بنابراین تجزیه B بر اساس قضیه ریسانس، تجزیه خوبی نیست.

C

- شرط اول را ندارد

مثال :

S (S#, Sname, Status, city)

BCNF است و 4NF و 5NF

SX(S#, Sname, city)

SY(S#, Status)

همچنین این دو هم BCNF و 4NF و 5NF سطح نرمالیتی افزایش نیافته است (S در بالاترین سطح نرمالیتی است).

نکته :

در عمل طراحی رابطه‌ها تا BCNF اکیدا توصیه می‌شود (با توجه به نکات قبلی) سطوح دیگر یعنی 4NF و 5NF ماهیتاً مقولات پژوهشی هستند. یعنی در عمل قریب به اتفاق رابطه‌های BCNF خودبخود 4NF اند و آنان نیز خودبخود 5NF.

مثال: $R = \{X, Y, Z, S, T, U, W\}$ با FD های زیر در نظر بگیرید.

$$F = \{S \rightarrow X, T \rightarrow Y, X \rightarrow Y, XY \rightarrow TUZ\}$$

الف) کلیدهای کاندید را بیابید.

ب) به طور کامل نرمالسازی کنید.

مثال: $R = \{X, Y, Z, S, T, U, W\}$ با FD های زیر در نظر بگیرید.

$$F = \{S \rightarrow X, T \rightarrow Y, X \rightarrow Y, XY \rightarrow TUZ\}$$

حل: الف)

$$X \rightarrow Y, XY \rightarrow TUZ \rightarrow X \rightarrow TUZ$$

$$\rightarrow X \rightarrow T, X \rightarrow U, X \rightarrow Z \quad *$$

$$S \rightarrow X, * \rightarrow S \rightarrow T, S \rightarrow U, S \rightarrow Z$$

$$S \rightarrow X, X \rightarrow Y \rightarrow S \rightarrow Y$$

$$\rightarrow S^+ = \{S, X, Y, Z, T, U\}$$

$$\rightarrow \text{کلید کاندید } (S, W)$$

مثال: $R = \{X, Y, Z, S, T, U, W\}$ با FD های زیر در نظر بگیرید.

$F = \{S \rightarrow X, T \rightarrow Y, X \rightarrow Y, XY \rightarrow TUZ\}$

حل: ب)

$F_{\text{optimal}} =$

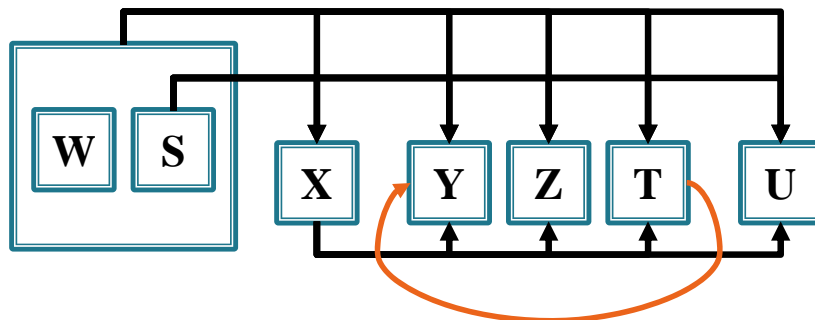
$\{ S \rightarrow X, S \rightarrow Y, S \rightarrow Z, S \rightarrow T, S \rightarrow U, \\ T \rightarrow Y, X \rightarrow Y, X \rightarrow T, X \rightarrow U, X \rightarrow Z \}$

۲۳

$F_{\text{optimal}} =$

$\{ S \rightarrow X, S \rightarrow Y, S \rightarrow Z, S \rightarrow T, S \rightarrow U, \\ T \rightarrow Y, X \rightarrow Y, X \rightarrow T, X \rightarrow U, X \rightarrow Z \}$

1 NF :

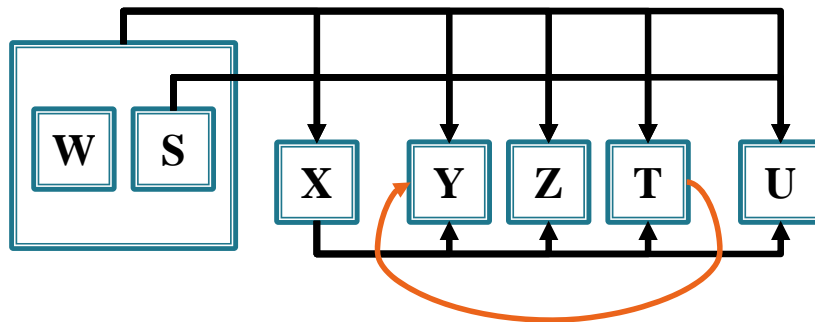


۲۴

$F_{\text{optimal}} =$

$\{ S \rightarrow X, S \rightarrow Y, S \rightarrow Z, S \rightarrow T, S \rightarrow U, \\ T \rightarrow Y, X \rightarrow Y, X \rightarrow T, X \rightarrow U, X \rightarrow Z \}$

1 NF :



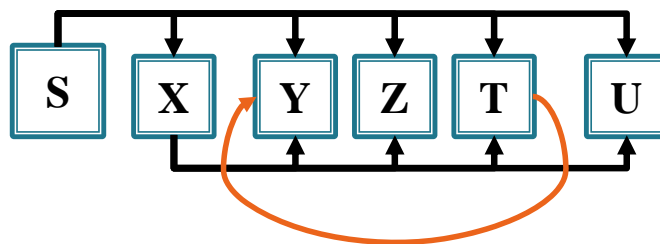
۲۵

$F_{\text{optimal}} =$

$\{ S \rightarrow X, S \rightarrow Y, S \rightarrow Z, S \rightarrow T, S \rightarrow U, \\ T \rightarrow Y, X \rightarrow Y, X \rightarrow T, X \rightarrow U, X \rightarrow Z \}$

2 NF :

$R_1 = (\underline{S}, X, Y, Z, T, U)$



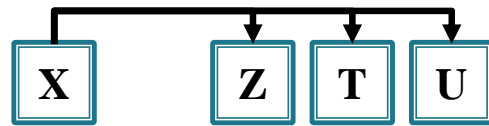
$R_2 = (\underline{S}, \underline{W})$



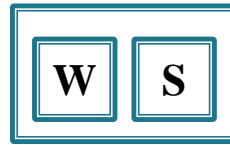
۲۶

3 NF :

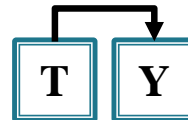
$$R_1 = (\underline{X}, Z, T, U)$$



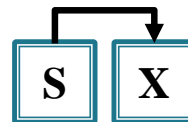
$$R_2 = (\underline{S}, W)$$



$$R_3 = (\underline{T}, Y)$$

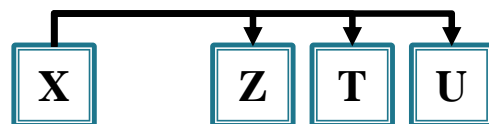


$$R_4 = (\underline{S}, X)$$



BCNF می باشد چراکه در هر رابطه فقط یک کلید کاندید وجود دارد

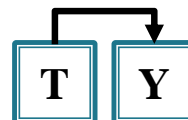
$$R_1 = (\underline{X}, Z, T, U)$$



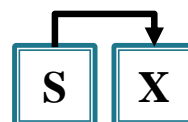
$$R_2 = (\underline{S}, W)$$



$$R_3 = (\underline{T}, Y)$$

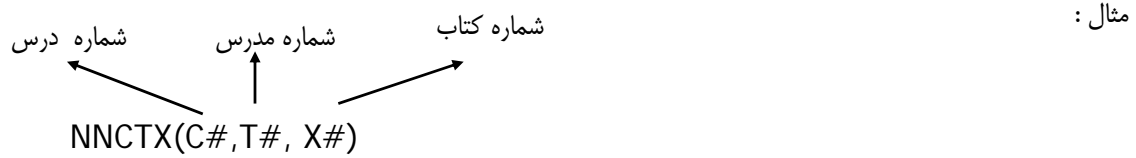


$$R_4 = (\underline{S}, X)$$



: 4NF

- بعضی از رابطه ها در شرایط بسیار خاص (بلکه نادر) علیرغم BCNF بودن باز هم آنومالی دارند یعنی حداقل از نظر تئوریک میتوان گفت BCNF بالاترین سطح نرمال بودن نیست.



صورت غیر نرمال این رابطه میتواند چنین باشد :

C1	$\left\{ \begin{array}{c} t1 \\ t2 \end{array} \right\}$	$\left\{ \begin{array}{c} x1 \\ x2 \end{array} \right\}$
C2	$\left\{ \begin{array}{c} t3 \end{array} \right\}$	$\left\{ \begin{array}{c} x1 \\ x3 \end{array} \right\}$

درس C توسط مدرس T از روی کتاب X تدریس می شود .
 در این محیط قاعده Semantic خاصی وجود دارد (که واقع بینانه هم نیست) :

یک درس توسط مجموعه ای از مدرسین تدریس میشود هرچه باشد کتاب درسی .
 و نیز یک درس از روی مجموعه ای از کتب درسی تدریس میشود هر چه باشد مدرس .

همین قاعده خاص به ما امکان می دهد که فرم نرمالتری بصورت زیر داشته باشیم .
 رابطه CTX تمام کلید است ، یعنی کلید با مجموعه عنوان برابر است .
 کلید کاندید : (C# , T# , X#)

CTX (C# , T# , X#)
C _۱ T _۱ X _۱
C _۱ T _۱ X _۲
C _۱ T _۲ X _۱
C _۱ T _۲ X _۲
C _۲ T _۳ X _۱
C _۲ T _۳ X _۳

چنین رابطه ای حداقل BCNF است ، چون یک دترمینان دارند ، که همان کلید است .

این رابطه BCNF است ولی با این همه آنرمالی دارد ناشی از همان قاعده خاص است .

-درج کن : < C_۱ , t_۴ >

برای درج این اطلاع باید دو تاپل درج شود .

C_۱ T_۴ X_۱

C_۱ T_۴ X_۲

-- هرچه مجموعه عناصر بیشتر باشد آنرمالیتی بیشتر است .

-درج کن : < C_۱, X_۳ >

باید درج شود < C_۱ , T_۱, X_۳ >

< C_۱ , T_۲ , X_۳ >

آنومالی هر چند ظریف ولی وجود دارد ، بنابراین CTX باید تجزیه شود

فعلا فرض می کنیم که این تجزیه چنین شود :

CT (C# , T#)

CX (C# , X#)

C ₁	T ₁
C ₁	T ₂
C ₂	T ₃

C ₁	X ₁
C ₁	X ₂
C ₂	X ₁
C ₂	X ₃

دو رابطه حاصله قطعا از CTX نرمالترند حداقلش این است که آنرمالیهای CTX را ندارند.

- درج کن : $\langle C_1, T_4 \rangle$

منجر به درج یک تاپل در CT و مشکلی ندارد .

- درج کن : $\langle C_5, X_3 \rangle$

منجر به درج یک تاپل در CX و مشکلی بروز نمی کند .

این تجزیه بر چه اساس است ؟

- FD ناکامل در CTX نداریم .

- Transitivity نداریم .

- دترمینان غیر کلید کاندید نداریم .

MVD

این تجزیه بر اساس نوعی وابستگی موسوم به "وابستگی چند مقداری" انجام شده است که بین C# ، T# ، X# وجود

Multi-Valued Dependency

دارند که بصورت زیر نمایش می دهیم .

$C\# \twoheadrightarrow T\#$

$C\# \twoheadrightarrow X\#$

تعریف MVD :

در رابطه R ، R(A,B,C) معمولا در رابطه های با سه صفت خاصه مرکب یا ساده (می گوییم ، B با A

وابستگی چند مقداری دارد و چنین می نویسیم $A \twoheadrightarrow B$ ، اگر به ازای یک مقدار از A مجموعه ای از مقادیر B متناظر

باشد . (اگر تعداد عناصر این مجموعه یک باشد همان FD معمولی را خواهیم داشت .) بر اساس همین تعریف و در رابطه غیر

نرمال CTX چنان داریم .

$C\# \twoheadrightarrow T\#$

$C\# \twoheadrightarrow X\#$

تعریف کاملتر :

در رابطه $R(A, B, C)$ می‌گوییم B با A وابستگی چند مقداری دارد و چنین می‌نویسیم $A \twoheadrightarrow B$. اگر به ازای یک مقدار مشخص از جفت صفت خاصه A و C مثلا A_1 و C_1 مجموعه ای خوش تعریف از B متناظر باشد.

مجموعه خوش تعریف یعنی مقادیر مجموعه B فقط به مقادیر مجموعه A بستگی داشته باشد و به C بستگی نداشته باشد.

همین خاصیت وجود مجموعه های خوش تعریف است که سبب می‌شود که MVD و رابطه های با سه صفت خاصه حتما $PAIRED$ باشد.

IF $(A \twoheadrightarrow B)$ THEN $(A \twoheadrightarrow C)$

همین پدیده MVD است که سبب شده است با دو تاپل $\langle C_1, T_2, X_2 \rangle$ ، $\langle C_1, T_1, X_1 \rangle$ باید نیز درج کنیم .

$\langle C_1, T_2, X_1 \rangle$

$\langle C_1, T_1, X_2 \rangle$

رابطه CTX را چنان تجزیه کردیم که در رابطه های حاصله پدیده MVD وجود نداشته باشد CT ، CX هر دو تمام کلیدند پس $BCNF$ اند . پدیده MVD ندارند پس $4NF$ اند .

<http://www.nurani.ir/> - Info@nurani.ir

۵۵

تعریف عملی $4NF$: رابطه ای $4NF$ است که $BCNF$ باشد و پدیده MVD در آن وجود نداشته باشد و تعریف دیگر :

این تعریف می‌گوید هرچه MVD در R باشد ، از نوع FD باشد ، از این تعریف برداشت می‌شود که اساسا رابطه MVD نداشته باشد .

$5NF$ یا PJ/NF : بعضی رابطه ها $4NF$ در شرایط بسیار خاص آنومالی ظریفی دارند (در عملیات ذخیره سازی)



(Projection-Join Normal Form)

$4NF$ حداقل از لحاظ تئوریک بالاترین سطح نرمال نیست .

<http://www.nurani.ir/> - Info@nurani.ir

۵۶

مثال :

Semantic تهیه کننده S قطعه P برای استفاده در پروژه J
 رابطه تمام کلید است ← BCNF (حداقل)
 پدیده MVD ندارد ← 4NF است .

S#	P#	J#
S ₁	P ₁	J ₂
S ₁	P ₂	J ₁
S ₂	P ₁	J ₁
S ₁	P ₁	J ₁

SP (S#	P#)	JOIN	PJ (P#	J#)
S ₁	P ₁	↔	P ₁	J ₂
S ₁	P ₂		P ₁	J ₁
S ₂	P ₁		P ₂	J ₁

فرض : رابطه باید تجزیه شود .

SP JOIN PJ

S#	P#	J#
S ₁	P ₁	J ₂
S ₁	P ₁	J ₁
S ₁	P ₂	J ₁
S ₂	P ₁	J ₂
S ₂	P ₁	J ₁

لذا SP JOIN PJ ≠ SPJ

نتیجه : رابطه SPJ ، 2-Decomposable نیست .

اما از JOIN رابطه فوق با SJ رابطه SPJ بدست می آید.

SJ (S#	J#)	S#	P#	J#
S ₁	J ₂	S ₁	P ₁	J ₂
S ₁	J ₁	S ₁	P ₁	J ₁
S ₂	J ₁	S ₁	P ₂	J ₁
		S ₂	P ₁	J ₁



* پدیده جالبی که در بعضی از 4NF ها دیده می شود .

* رابطه SPJ ، 3-Decomposable است یعنی می توان نوشت :

$$SPJ = SP \text{ JOIN } PJ \text{ JOIN } SJ$$

رابطه N-Decomposable

رابطه ای است که حاصل پیوند n پرتوش باشد .

$$R = R_1 \text{ JOIN } R_2 \dots \dots \dots R_{n-1} \text{ JOIN } R_n$$

و اگر کمتر از N پرتوش را JOIN کنیم ، رابطه بدست نیاید .

رابطه N-DECOM دارای وابستگی پیوندی N رابطه ای است و این معنا را چنین نمایش می دهیم .

$$R = JD * (R_1, R_2, \dots, R_n)$$

$$SPJ = JD * (SP, PJ, SJ) \quad \text{مثال : } n=3$$

یعنی SPJ وابستگی پیوندی به حاصل پیوند 3 پرتو دارد و نه کمتر

آنرمالی ظریفی که باعث تجزیه شده ، چه بوده است ؟ ریشه آن چیست ؟

اینکه P_1 و S_1 در SP است ، معنایش اینست که تاپلی مثلا J_2, P_1, S_1 در SPJ هست .
اطلاع

اینکه P_1 و J_1 در PJ است ، معنایش اینست که تاپلی مثلا J_1, P_1, S_2 در SPJ هست .
اطلاع

اینکه S_1 و J_1 در SJ است ، معنایش اینست که تاپلی مثلا J_1, P_2, S_1 در SPJ هست .
اطلاع

چون $SPJ = JD * (SP, PJ, SJ)$ پس اگر

J_2, P_1, S_1 در SPJ
 J_1, P_1, S_2 در SPJ
 J_1, P_2, S_1 در SPJ

آنگاه J_1, P_1, S_1 هم در SPJ است

معنای این الزام : اگر در لحظه ای از حیات رابطه SPJ داشته باشیم و بگوییم درج کن J_1, P_1, S_2 را در SPJ
 حتما باید تاپل J_1, P_1, S_1 را هم درج کرد .

S#	P#	J#
S_1	P_1	J_2
S_1	P_2	J_1
S_2	P_1	J_1
S_1	P_1	J_1

این رکوردها در ابتدا در رابطه بوده

چه محدودیتی در SPJ وجود دارد ؟

- نوعی محدودیت چرخشی در SPJ حاکم است و سبب می شود اندیس ۲ (در اینجا و در حالت کلی در این رابطه بطور
 گردش جابجا شود و این آنومالی را موجب شود

$$S = JD * (SX, SY) \left\{ \begin{array}{l} SX (S\#, SNAME, CITY) \\ SY (S\#, STATUS) \end{array} \right. \quad n=2$$

$$S = JD * (SX, SY, SZ) \left\{ \begin{array}{l} SX (S\#, SNAME) \\ SY (S\#, CITY) \\ SZ (S\#, STATUS) \end{array} \right. \quad n=3$$

$$S = JD * (SX, SZ, SW) \left\{ \begin{array}{l} SX (S\#, SNAME) \\ SY (S\#, CITY) \\ SZ (S\#, STATUS) \\ SW(S\#) \end{array} \right. \quad n=4 \text{ از نظر تئوریک وجود دارد}$$

* می بینیم که در تمام پرتوهای S ، کلیدکاندید موجود است و این تفاوت پرتوهای S و پرتوهای SPJ است. اصطلاحاً می گوئیم پرتوهای S ناشی از کلیدکاندید هستند. در پرتوهای SPJ کلیدکاندید وجود ندارد .

*مشکل SPJ اینست که یک JD با $n > 2$ دارد و این JD ناشی از وجود کلیدکاندید نیست

تعریف رابطه 5NF :

رابطه ای 5NF است که اگر تمام JD های موجود در آن ناشی از وجود کلیدکاندید می باشد . بیان عملی : اگر در یک رابطه ای حداقل دو پرتو وجود داشته باشد که پیوند آنها رابطه اولی را بدهد ، اما در پرتو کلیدکاندید حضور نداشته باشد ، آن رابطه 5NF نیست.

نکته : در عمل کفایت رابطه ها را تا سطح BCNF نرمال کنیم ، زیرا در محیطهای واقعی شرایط بروزی پدیده MVD معمولاً وجود ندارد و در نتیجه رابطه BCNF خود بخود 4NF هم هست و نیز محدودیت چرخشی موجود در SPJ یک محدودیت بسیار نادری است که معمولاً در عمل وجود ندارد بنابراین رابطه 4NF نیز در عمل معمولاً خود به خود 5NF است .

جمع بندی :

مزایای نرمالترسازی :

طراحی بهتر و واضحتر به نحوی که طراح درک روشنتری از موجودیتهای و ارتباطات محیط عملیاتی دارد (بدلیل عدم وجود اختلاط اطلاعاتی)

کاهش بعضی انواع آنومالیها (زیرا هنوز تمام وابستگی ها شناخته نشده است)

کاهش افزونگی در رابطه ها (دلیل عملی افزونگی : اختلاط اطلاعاتی دلیل تنوری : وابستگی اطلاعاتی)

امکان اعمال بعضی قواعد جامعیتی (ناشی از وابستگی ها)

معایب :

۱- ایجاد OVERHEAD در عملیاتی بازیابی --> نکته : گاه حتی لازم است برای کاهش Overhead رابطه های نرمالتر شده را DENORMALIZE کنیم ، برای آنکه به درخواست کاربران سریعتر پاسخ دهیم .

۲ - چرا برای نرمالترسازی همیشه از پرتوگیری استفاده میکنیم ، یعنی الگوریتم اصلی در نرمالترسازی ، پرتوگیری و آنگاه پیوند پرتوها ست . آیا از اپراتورهای دیگر نمی توان استفاده کرد ؟
جواب این ایراد : راه متصور دیگر استفاده از اپراتورهای SELECT و UNION است . به این معنا که یک رابطه را بر اساس ضابطه های خاصی با استفاده از SELECT بطور افقی به چند رابطه تجزیه کنیم و هرگاه به رابطه اصلی نیاز داشتیم آنها را UNION کنیم . یعنی :

مثال :

$S(S\#, Sname, Status, City)$

این رابطه را می توان با SELECT به تعدادی رابطه تقسیم کرد بدین صورت :

$SC_1(S\#, SNAME, STATUS, [CITY])$	← فقط تهیه کنندگان ساکن C_1
$SC_2(S\#, SNAME, STATUS, [CITY])$	← فقط تهیه کنندگان ساکن C_2
$SC_i(S\#, SNAME, STATUS, [CITY])$	← فقط تهیه کنندگان ساکن C_i

$S = SC_1 \text{ UNION } SC_2 \text{ UNION } \dots$