

بسمه تعالی



فصل ششم

وابستگی تابعی و نرمال سازی



اگر در یک سیستم بانک اطلاعاتی همه جداول به هم پیوند داده شود ما فقط با یک جدول سروکار میداشتیم و اینکار چه نتیجه ای در بر میداشت ؟

جواب : مزیت عمده آن ساده تر شدن درخواست ها و پرس و جوها از بانک اطلاعاتی و عدم نیاز به پیوندها (Join) و در نتیجه افزایش سرعت پاسخ گوئی

به عنوان مثال: به سه جدول معروف SP,P,S (فصل های قبل) نگاه کنید. فرض کنید جدول SP را به صورت (S#,P#,Qty,Status) تعریف و جدول S را به صورت (S#,Sname,City) تعریف می کردیم، یعنی فیلد Status را از جدول S به جدول SP می بردیم. در اینصورت جدول به صورت روبرو می شد:

همانطور که مشاهده می گردد فیلد Status برای S1 و S2 همواره ثابت و مشخص است (عدد 20 و 10) و بی جهت در جدول تکرار شده و بدین جهت افزونگی اطلاعات داریم

SP'

S#	P#	Qty	Status
S1	P1	300	20
S1	P2	200	20
S1	P3	400	20
S1	P4	200	20
S1	P5	100	20
S1	P6	100	20
S2	P1	300	10
S2	P2	400	10



معایب عمده :

۱- افزونگی داده ها (Data Redundancy)

کد دانشجویی	نام	فامیلی	ش.ش	سال تولد	نام پدر	آدرس	تلفن	نام درس	نمره
۲	علی	محمدی	۱۱	۱۳۶۰	محمد	رشت	۲۳۳۲	ریاضی	۱۳
۵	محمد	ناصری	۱۱	۱۳۶۲	رضا	زنجان	۳۲۵۱	ریاضی	۱۴
۶	علیرضا	کاظمی	۳۵۴	۱۳۵۸	حسین	انزلی	۲۱۲۱	فیزیک	۱۸
۵	محمد	ناصری	۱۱	۱۳۶۲	رضا	زنجان	۳۲۵۱	فیزیک	۱۵
۶	علیرضا	کاظمی	۳۵۴	۱۳۵۸	حسین	انزلی	۲۱۲۱	شیمی	۱۷
۲	علی	محمدی	۱۱	۱۳۶۰	محمد	رشت	۲۳۳۲	فیزیک	۱۶
۶	علیرضا	کاظمی	۳۵۴	۱۳۵۸	حسین	انزلی	۲۱۲۱	شیمی	۱۹
۶	علیرضا	کاظمی	۳۵۴	۱۳۵۸	حسین	انزلی	۲۱۲۱	فارسی	۱۶
۳	رضا	رضایی	۴۵۱	۱۳۶۰	احمد	تبریز	۲۳۲۴	فارسی	۹



۲- بی‌نظمی (Anomaly):

طراحی نامناسب یا ناقص رابطه‌های (جداول) یک سیستم بانک اطلاعاتی، باعث ایجاد بی‌نظمی یا آنومالی (Anomaly) می‌گردد.

آنومالی (Anomaly) بر مبنای ۳ حالت زیر اتفاق می‌افتد:

❑ **آنومالی در درج:** عدم امکان عمل درج (Insert) در جدول به علت عدم تعریف تمام یا بخشی از کلید اصلی که باعث نقص قانون جامعیت می‌شود.

به عنوان مثال اضافه کردن نام درس و نمره در جدول قبل غیر ممکن است، چون شماره دانشجویی (کلید اصلی) معلوم نیست

❑ **آنومالی در حذف:** بروز تبعات نامطلوب روی انجام عمل حذف (Delete) روی یک رکورد.

-حذف اطلاعات یک یا چند دانشجو میتواند باعث حذف مشخصات درس نیز گردد.

❑ **آنومالی در بروزآوری:** بروز فزونکاری در سیستم در انجام عمل بروزآوری (Update)

- اگر اسم یک درس تغییر یابد و در همه رکوردهای دانشجویانی که این درس را انتخاب کرده اند، بروزآوری نشود، باعث ناسازگاری اطلاعات و باعث نقص جامعیت می‌شود.



مقدمه نرمال سازی



5

جدول SPC

S#	P#	QTY	CITY
S1	P1	100	C2
S1	P2	200	C2
S1	P3	80	C2
S2	P1	70	C1
S2	P2	120	C1
S3	P1	90	C1
S4	P1	80	C3
S4	P3	90	C3

رابطه SPC مفروض است :

❑ آنومالی در عمل درج :

- برای درج یا اضافه کردن (Insert) اطلاع $\langle S7, C7 \rangle$ در جدول ، دچار آنومالی میشویم . چون 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 .

❑ آنومالی در عمل حذف :

- حذف اطلاع $\langle S3, P1, 90 \rangle$ انجام شدنی است ، اما اطلاع ناخواسته حذف می شود (اینکه S3 ساکن C1 است).

❑ آنومالی در عمل بروز آوری :

بهنگام سازی : شهر S1 را عوض کن .
`UPDATE SPC SET CITY = 'C1' WHERE S# = 'S1'`

این عمل ، منطقاً تاپلی (مربوط به یک رکورد) ، تبدیل به عمل منطقاً مجموعه ای (مربوط به گروهی از رکوردها) و نهایتاً منجر به بروز فزونکاری در سیستم می شود.



نتیجه اولیه :

مشخص شد که رابطه SPC دارای آنومالی است و دلیل غیر تئوریک این آنومالی اینست که در رابطه SPC پدیده اختلاط اطلاعاتی وجود دارد.

یعنی اطلاعات در مورد موجودیت محصول (P#) با اطلاعات تهیه کننده (S#) مخلوط شده است. شهر (City) از صفات خاصه تهیه کننده (S#) است و با صفات خاصه محصول (P#) ترکیب شده است ، همین اختلاط اطلاعاتی سبب بروز پدیده افزونگی هم شده است .
رابطه SPC خوش طرح (Well Design) نیست و در مراحل بعد خواهیم دید که این طراحی باید تغییر کند.

۳- مشکل مقادیر Null :

اگر فرض کنید در مثال قبل بخواهیم ثبت نام اولیه دانشجو (ثبت اطلاعات پایه ای) را انجام دهیم که هنوز انتخاب واحدی را انجام نداده باشد باید مقدار *NULL* به جای اسم درس و نمره وارد کنیم.



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 – (FD) Functional Dependency

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 رابطه

مثال : رابطه R مفروض است :

X	Y	Z
X1	Y1	Z1
X1	Y1	Z2
X2	Y1	Z2
X2	Y1	Z3
X3	Y2	Z3

آیا $X \rightarrow Y$ ؟ **بله**

آیا $Y \rightarrow X$ ؟ **خیر**

آیا $X \rightarrow Z$ ؟ **خیر**



نکات مهم :

۱- وابستگی تابعی برای همه اطلاع (رکوردها) ی ممکن (رکوردهای موجود و ممکن آینده) باید مصداق داشته باشد. یعنی وابستگی به معنی و ذات صفتها (فیلدها) مربوط است نه به موارد خاص یا صرفاً اطلاعات موجود.

۲- وابستگی تابعی میتواند برای هر بانک اطلاعاتی متفاوت باشد.

امکان یا عدم امکان تدریس یک درس در یک ترم توسط بیش از یک استاد

استاد >---- (ترم ، درس)

۳- 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.



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

$CITY \rightarrow S\#$: 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 :

$CO\# \rightarrow PR\#$: یعنی یک استاد فقط یک درس را تدریس می کند.

$PR\# \rightarrow CO\#$: یک درس توسط اساتید مختلف تدریس می شود.

مثال : در رابطه R1 وابستگی تابعی $A \rightarrow B$ و $B \rightarrow C$ را بررسی کنید؟

A	B	C
A1	B1	C1
A2	B2	C3
A1	B1	C2
A3	B4	C2
A5	B1	C1

وابستگی $A \rightarrow B$ برقرار است زیرا به ازای هیچ مقدار مساوی از A ، دو مقدار متفاوت از B وجود ندارد.

وابستگی $B \rightarrow C$ برقرار نیست زیرا به ازای B1 دو مقدار C1 و C2 در رابطه R1 وجود دارد



مثال : رابطه S را در نظر می گیریم

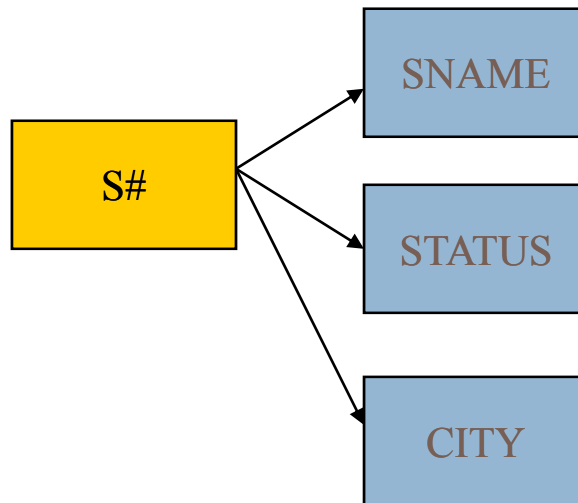
$S (S\#, SNAME, STATUS, CITY)$

$S\#$ کلید کاندید است.

$S\# \rightarrow SNAME$

$S\# \rightarrow STATUS$

$S\# \rightarrow CITY$



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

نمودار FD های رابطه S

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



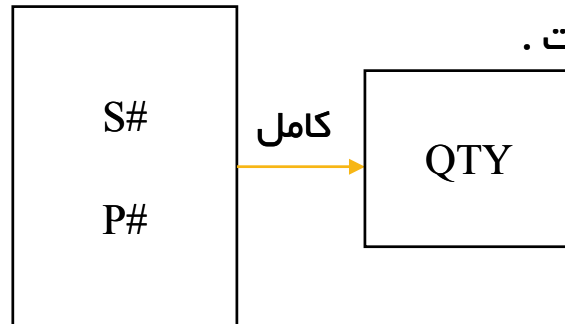
وابستگی تابعی کامل (Full Functional Dependency) (FFD)

11

۲- مفهوم وابستگی تابعی کامل (Full Functional Dependency) (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) ، (P# و S#) کلید اصلی است .



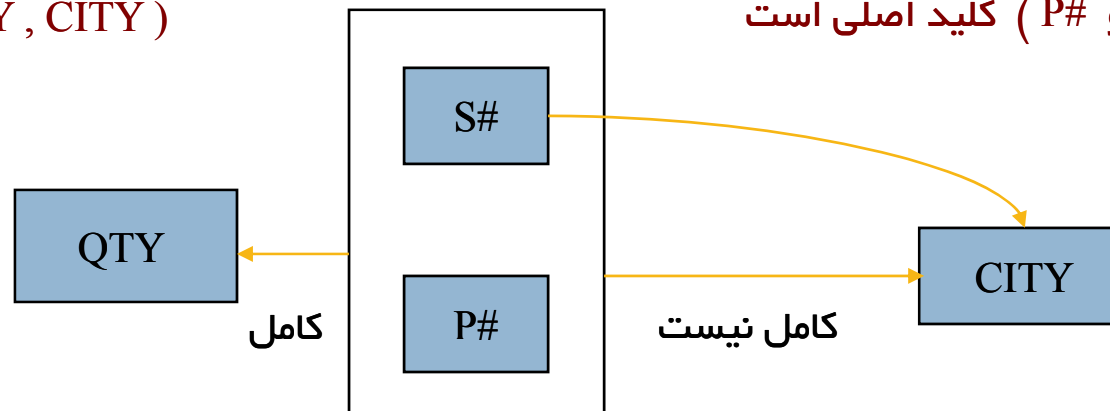
$$(S\#, P\#) \rightarrow QTY$$

$$P\# \not\rightarrow QTY$$

$$S\# \not\rightarrow QTY$$

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

در رابطه روبرو ، (P# و S#) کلید اصلی است



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



مثالی از وابستگی تابعی

12

مثال : قواعد زیر در یک سیستم بانک اطلاعاتی برقرار است نمودار FD مربوطه به آن را ، به همراه یک جدول با مقادیر دلخواه متناظر با آن را ایجاد کنید :

قاعده ۱ : هر ناشر تعدادی کتاب منتشر میکند .

قاعده ۲ : هر کتاب به تعداد خاصی منتشر میشود .

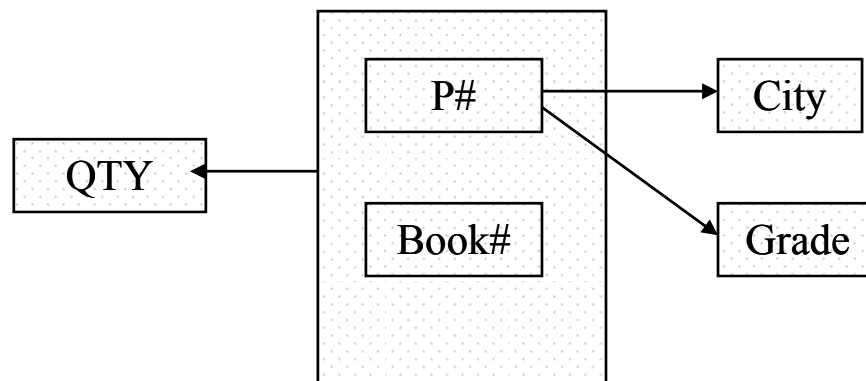
قاعده ۳ : هر ناشر در یک شهر دفتر دارد .

قاعده ۴ : هر ناشر دارای یک رتبه انتشاراتی است .

قاعده ۵ : ناشران یک شهر دارای یک رتبه انتشاراتی هستند .

جواب : P# شماره ناشر - BOOK# شماره کتاب - QTY تعداد انتشار - City شهر - Grade رتبه

- 1) $P\# \twoheadrightarrow Book\#$ 2) $(P\#, Book\#) \twoheadrightarrow QTY$ 3) $P\# \twoheadrightarrow City$
 4) $P\# \twoheadrightarrow Grade$ 5) $City \twoheadrightarrow Grade$



P#	Book#	QTY	City	Grade
P1	B1	5000	C1	40
P2	B4	3000	C2	60
P3	B3	2000	C1	40
P1	B2	5000	C1	40
P4	B3	4000	C3	50
P4	B5	3500	C3	50



مثالی از وابستگی تابعی

A	B	C
A1	B1	C1
A2	B1	C3
A1	B2	C2

مثال : در رابطه R وابستگی تابعی کامل $(A,B) \Rightarrow C$ را بررسی کنید

- رابطه $(A,B) \Rightarrow C$ برقرار است.

- رابطه $(A,B) \Rightarrow C$ برقرار است چون C به تنهایی به A و یا B وابسته نیست.

تعریف : اگر برای تمام صفهای B در رابطه R داشته باشیم $A \rightarrow B$ آنگاه A را ابر کلید (Super Key) رابطه R مینامند و بصورت $A \rightarrow R$ نمایش میدهند

اگر این وابستگی از نوع FFD باشد آنگاه A کلید کاندید رابطه R است .

تعریف : اگر B زیر مجموعه ای از A باشد آنگاه $A \rightarrow B$ این وابستگی تابعی را بدیهی (Trivial FD) گویند.

نکته : در یک بانک اطلاعاتی ما بدنبال اطلاع از همه وابستگی ها هستیم. تعدادی از وابستگی های بین صفها بشکل ساده تری قابل فهم و شناسایی هستند ولی باید سایر وابستگی ها را باید بدست آورد.

هر گاه تعدادی وابستگی داشته باشیم ممکن است بتوانیم وابستگی های دیگری نیز از آنها بدست آوریم و یا بعضی از آنها را که اطلاعات جدیدتری به ما نمی دهند ، حذف کنیم.



مثال : رابطه مفروض R را داریم :

$$R = (S, T, U, V, W)$$

$$F = (S \rightarrow T, V \rightarrow SW, T \rightarrow U)$$

آنگاه منطقاً میتوان وابستگی های زیر را نیز بدست آورد :

$$S \rightarrow U \quad (\text{زیرا } S \rightarrow T, T \rightarrow U)$$

$$V \rightarrow W$$

$$V \rightarrow S$$

$$V \rightarrow T \quad (\text{زیرا } V \rightarrow S, S \rightarrow T)$$

$$V \rightarrow U \quad (\text{زیرا } V \rightarrow S, S \rightarrow U)$$

پس نتیجه میگیریم که V کلید کاندید است .



مثالی از وابستگی تابعی

15

مثال: در بانک اطلاعاتی زیر ابتدا کلید کاندیدا را یافته و سپس آن را نرمال سازی کنید

$$R = \{A, B, C, D, E, F, G\}$$

$$F = \{AF \longrightarrow BE, FC \longrightarrow DE, F \longrightarrow CD, D \longrightarrow E, C \longrightarrow A\}$$

حل: ابتدا سمت راست وابستگی ها را به صفت تبدیل می کنیم

$$\begin{array}{lll} AF \longrightarrow B & AF \longrightarrow E & FC \longrightarrow D \\ FC \longrightarrow E & F \longrightarrow C & F \longrightarrow D \\ D \longrightarrow E & C \longrightarrow A & \end{array}$$

از شماره ۵ برای ساده کرن سمت چپ ۳ و ۴ استفاده می کنیم . در نتیجه:

$$3) F \longrightarrow D \quad 4) F \longrightarrow E$$

از شماره ۵ و ۸ می توان نتیجه گرفت $F \longrightarrow A$ این نتیجه را بر روی ۱ و ۲ اعمال می کنیم:

$$1) F \longrightarrow B \quad 2) F \longrightarrow E$$

$$F = \{F \longrightarrow A, F \longrightarrow B, F \longrightarrow C, F \longrightarrow D, F \longrightarrow E, D \longrightarrow E, C \longrightarrow A\} \quad \text{در نتیجه داریم:}$$

در نتیجه F همه صفت ها دیگر به جز G را می دهد. پس (F,G) کلید کاندیدا است.

$$1NF : F, G, A, B, C, D, E$$

$$2NF : (F, G)$$

$$F, A, B, C, D, E$$



مجموعه پوششی وابسته

16

تعریف :

اگر F یک مجموعه از وابستگی های تابعی باشد آنگاه مجموعه تمام وابستگی های تابعی که از آن منتج میشود را مجموعه پوششی F مینامیم و با F^+ نمایش میدهیم

آقای آرمسترانگ در سال ۱۹۷۴ روشی برای استخراج پوششی ارائه داد و اثبات کرد که سه قاعده زیر برای استخراج مجموع پوششی کافی است ، یعنی با اعمال مکرر این سه قاعده میتوان به تمام وابستگی های منتج دست یافت و هیچ وابستگی اضافی نیز تولید نمیشود :

۱- **بازتاب :** اگر B زیر مجموعه A باشد آنگاه $A \rightarrow B$

۲- **افزایش :** اگر $A \rightarrow B$ و C صفت خاصه باشد آنگاه $AC \rightarrow BC$

۳- **انتقال :** اگر $A \rightarrow B$ و $B \rightarrow C$ آنگاه $A \rightarrow C$

هر چند قوانین برای استخراج F^+ کافی بود ولی اعمال آنها مشکل بود و بعدها قوانین دیگری به آن اضافه که کار را تسهیل میکرد که مهمترین آنها :

۴- **اجتماع (Union) :** اگر $A \rightarrow B$ و $B \rightarrow C$ آنگاه $A \rightarrow BC$

۵- **تجزیه (Decomposition) :** اگر $A \rightarrow BC$ آنگاه $A \rightarrow B$ و $A \rightarrow C$

۶- **ترکیب (composition) :** اگر $A \rightarrow B$ و $C \rightarrow D$ آنگاه $AC \rightarrow BD$



مجموعه وابستگی بهینه

17

با اعمال قواعد فوق وابستگی زیادی بدست می آید که بعضاً "تکراری و اضافی هستند و در اینجا روشی برای حذف اینگونه وابستگی ها و رسیدن به مجموعه وابستگی بهینه ارائه میشود.

تعریف :

دومجموعه وابستگی تابعی $F1$ و $F2$ را معادل (Equivalent) مینامیم اگر مجموعه پوششی آنها برابر باشند یعنی $F1+ = F2+$

با استفاده از قواعد ۳ گانه زیر میتوانید مجموعه وابستگی را به مجموعه بهینه معادل آن تبدیل کرد :

۱- سمت راست هر وابستگی فقط یک صفت باشد.

۲- هر صفتی که $F+$ را تغییر نمی دهد از سمت چپ حذف شود.

۳- وابستگی های تکراری و اضافی حذف شود.

بطور خلاصه باید گفت : که برای یافتن وابستگی های تابعی در یک بانک اطلاعاتی میبایست ابتدا مجموعه پوششی وابستگی ها را تعیین کرد و سپس آنرا بهینه نمود.



مثال :

در بانک زیر ، مجموعه وابستگی پوششی بهینه را بیابید .

$$R=(U, V, W, X, Y, Z)$$

$$F=\{U \longrightarrow XY, X \longrightarrow Y, XY \longrightarrow ZV, U \longrightarrow ZV\}$$

حل: ابتدا F^+ را می یابیم و سپس آن را بهینه می کنیم.

$$F^+=\{U \longrightarrow XY, X \longrightarrow Y, XY \longrightarrow ZV, U \longrightarrow ZV\}$$

قاعده ۱ روی وابستگی اول:

$$F^+=\{U \longrightarrow X, U \longrightarrow Y, X \longrightarrow Y, XY \longrightarrow ZV, U \longrightarrow ZV\}$$

قاعده ۲ روی $XY \longrightarrow ZV$ با توجه به $X \longrightarrow Y$:

$$F^+=\{U \longrightarrow X, U \longrightarrow Y, X \longrightarrow Y, XY \longrightarrow ZV, U \longrightarrow ZV\}$$

حال قاعده ۱ روی سایر وابستگی ها :

$$F_{opt}=\{U \longrightarrow X, U \longrightarrow Y, U \longrightarrow Z, U \longrightarrow V, X \longrightarrow Y, X \longrightarrow Z, X \longrightarrow V\}$$



کلیدهای کاندید

19

اگر مجموعه ای از صفتها را ATTR و مجموعه وابستگی تابعی آنها و تعدادی صفت دیگر را F بنامیم آنگاه الگوریتم زیر ، مجموعه تمام صفت های وابسته به ATTR را میدهد (بهتر است ابتدا F را بهینه کنیم) .

$$ATTR^+ = ATTR$$

تکرار کن

برای هر $X \rightarrow Y$ در F

اگر X زیر مجموعه $ATTR^+$ باشد آنگاه:

$$ATTR^+ = ATTR^+ \cup V$$

تا زمانی که $ATTR^+$ دیگر تغییر نکند.

با استفاده از این الگوریتم می توان کلیدهای کاندیدا را بدست آورد. و باید به نکات زیر توجه کرد:

۱- هر کلید کاندیدا شامل مجموعه ای از صفت هایی است که در سمت چپ پیکان ها می آیند.

۲- کلید کاندیدا باید کمینه باشد ، یعنی زیر مجموعه ای از آن خاصیت کلیدی نداشته باشد.

۳- بانک اطلاعات ممکن است چند کلید کاندیدا داشته باشد.

۴- کلید کاندیدا ممکن است در یک یا چند صفت مشترک باشند.



کلیدهای کاندید

20

۵- هرگاه تعداد صفت ها در بانک زیاد و وابستگی آنها گسترده باشد، می توان با رسم نمودار وابستگی به فهم بهتر آن بانک اطلاعات کمک کرد. در این نمودار ، صفت ها در مستطیل قرار می گیرند و پیکانی از آنها به هر یک از صفت های وابسته به آن رسم می شود. برای مجموعه صفت ها و وابستگان آنها نیز مستطیل های دیگری رسم می گردد. معمولاً پیکان هایی که وابستگی به کلید اصلی را نشان می دهند در بالای صفت ها و سایر پیکان ها زیر آنها کشیده می شوند. ابتدا باید مجموعه بهینه وابستگی ها را بدست آوریم و آنگاه به رسم نمودار وابستگی مبادرت کنیم .

مثال: اگر $R = (S, TU, V, W)$ و $F = \{ S \longrightarrow T, V \longrightarrow SW, T \longrightarrow U \}$

آنگاه : $\{S, V\}^+ = \{S, V\} \quad \{S, V, T, W\} \quad \{S, V, T, W, U\}$

می توان این الگوریتم را برای مجموعه صفت های دیگر سمت چپ پیکان مرتباً بکار برد و وابستگی های مختلف و نیز کلیدها را بدست آورد.

اگر در مثال فوق V^+ را محاسبه کنیم خواهیم دید که $\{S, V\}$ کاهش پذیر است:

$$V^+ = \{V\} \Longrightarrow \{V, S, W\} \Longrightarrow \{V, S, W, T\} \Longrightarrow \{V, S, W, T, U\}$$

در نتیجه $\{S, V\}$ کلید کاندیدا نیست ولی V هست .



کلیدهای کاندید

21

مثال: در بانک اطلاعات زیر ابتدا همه کلیدهای کاندیدا را بیابید و سپس نمودار وابستگی را رسم کنید.

$$R = (U, V, W, X, Y, Z, O, P, Q)$$

$$F = \{U \rightarrow VXQ, UVP \rightarrow O, OQ \rightarrow YZ, UP \rightarrow XY\}$$

حل: ابتدا مجموعه بهینه معادل F را پیدا می‌کنیم (تمرین):

$$F = \{U \rightarrow V, U \rightarrow X, U \rightarrow Q, OQ \rightarrow Y, OQ \rightarrow Z, UP \rightarrow Y, UP \rightarrow O, UP \rightarrow Z\}$$

سپس مجموعه صفت های وابسته به تمام مجموعه صفت های چپ پیکان ها را می‌یابیم:

$$U^+ = \{U\} \rightarrow \{U, V, X, Q\}$$

$$UP^+ = \{U, P\} \rightarrow \{U, P, V, X, Q, O, Y\} \rightarrow \{U, P, V, X, O, Q, Y, Z\}$$

$$OQ^+ = \{O, Q\} \rightarrow \{O, Q, Y, Z\}$$

نتیجه می‌گیریم که بیشترین صفت ها را UP^+ می‌دهد. تنها صفتی که از UP^+ خارج است W است.

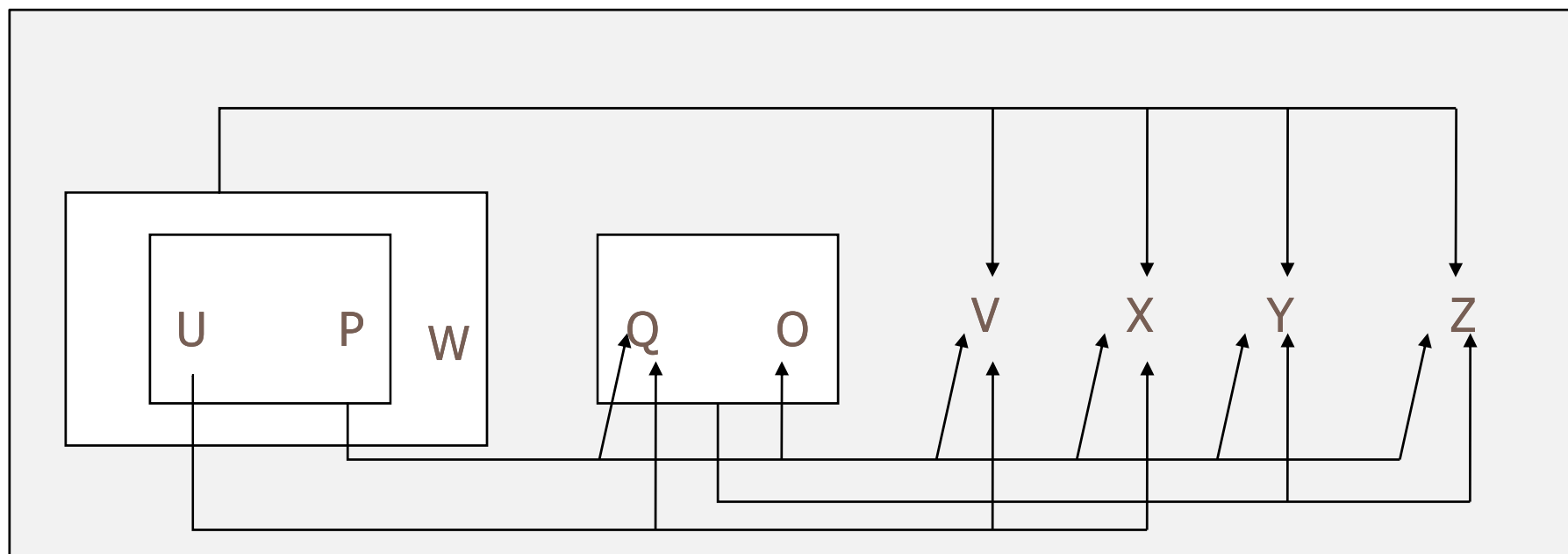
پس $UPW \rightarrow R$ کلید کاندید می‌باشد. یعنی $UPW \rightarrow R$ کلید کاندید دیگری به دست نمی‌آید زیرا از O

و Q نمی‌توان به U و P رسید (یادآوری می‌شود که کلید کاندیدا باید کمینه باشد).



نمودار وابستگی تابعی

نمودار وابستگی این بانک اطلاعات در شکل زیر مشاهده می شود.



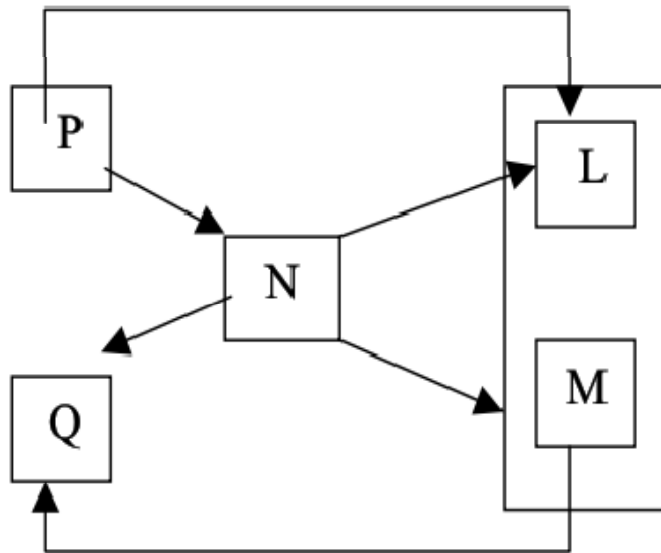
شکل - مثالی از نمودار وابستگی تابعی

تمرین: وابستگی $P \rightarrow OQ$ را به مثال فوق اضافه کنید. آیا کلید دیگری خواهیم

داشت؟



نمودار وابستگی تابعی



مجموعه حداقل FD های این رابطه را بدست آورید .

حل : با توجه به نمودار FD ها داریم :

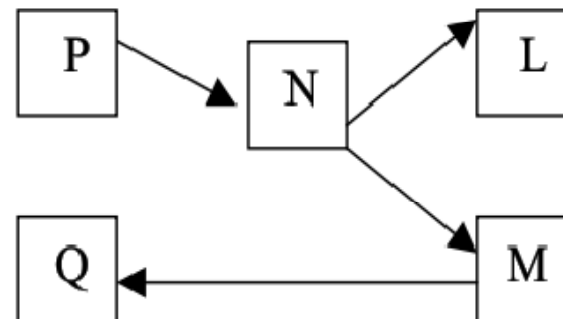
- | | |
|----------------------|---------------------------|
| 1. $P \rightarrow N$ | 4. $N \rightarrow Q$ |
| 2. $N \rightarrow L$ | 5. $N \rightarrow (L, M)$ |
| 3. $P \rightarrow L$ | 6. $M \rightarrow Q$ |

FD شماره 3 افزونه است زیرا منطقاً از FD های 1 و 2 قابل

استنتاج است . از FD شماره 2 و 5 داریم : $N \rightarrow M$

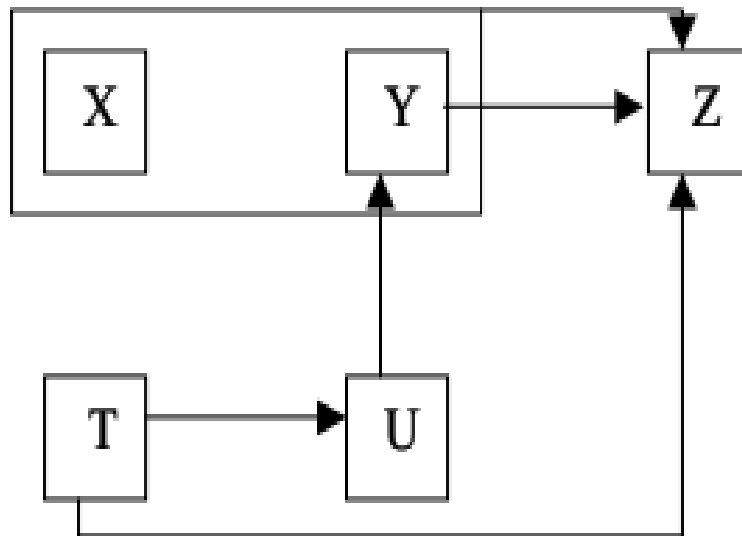
با توجه به FD های 6 و 7 ، FD شماره 4 افزونه است ، بنابراین مجموعه حداقل FD ها بصورت زیر است :

$P \rightarrow N$ و $N \rightarrow L$ و $N \rightarrow M$ و $M \rightarrow Q$





۲- در نمودار FD های زیر مجموعه حداقل FD ها را بدست آورید .



$$1. (X,Y) \rightarrow Z$$

$$2. Y \rightarrow Z$$

$$3. T \rightarrow U$$

$$4. U \rightarrow Y$$

$$5. T \rightarrow Z$$

FD شماره ۵ افزونه است .

FD شماره ۱ نیز افزونه است . چرا ؟

پس داریم : $T \rightarrow U$, $Y \rightarrow Z$

[چون از $Y \rightarrow Z$ می توان نتیجه گرفت $(X,Y) \rightarrow Z$ زیرا اگر $(X,Y) \not\rightarrow Z$ یعنی $(X1, Y1, Z1)$ و

$(X1, Y1, Z2)$ و چون $(Y1, Z1), (Y1, Z2)$ لذا $Y \not\rightarrow Z$ خلاف فرض است پس $Y \rightarrow Z$]



فرایند نرمال سازی

25

نرمال سازی (متعارف سازی)، فرایند کاهش افزونگی داده ها در یک بانک اطلاعاتی است. هنگام طراحی یک بانک اطلاعاتی، این سؤالات مهم مطرح می شود که: «با توجه به داده های عملیاتی و ارتباط بین موجودیت ها، چند جدول می بایست طراحی کرد؟ در هر جدول چه فیلدهایی باید قرار گیرد؟ رابطه جدول ها باید چگونه باشد؟» در مدل رابطه ای روشی کاملاً کلاسیک و ریاضی گونه برای پاسخگویی به سؤالات فوق وجود دارد که به روش «نرمال سازی» (Normalization) موسوم است. نرمال سازی در عمل یعنی پیروی از یک سری فرم های نرمال که منجر به تجزیه جداول می شوند.

تعریف نرمال سازی :

تجزیه یک رابطه یا یک جدول به روابط کوچکتر با رعایت دو شرط زیر را نرمال سازی می گویند:

۱ - خاصیت برگشت پذیری (اتصال جداول تجزیه شده به حالت اول)

۲ - خاصیت بدون از دست دادن اطلاعات (با اتصال مجدد جداول اطلاعاتی از دست نرفته باشد)

نکته : هرچه رابطه نرمالتر باشد آنومالی کمتر می شود.

ملاک تجزیه رابطه : تجزیه رابطه R به رابطه های R_1 و R_2 میبایست به نحوی باشد که پیوند دو رابطه R_1 و R_2 ، رابطه R را ایجاد کند و تاپلی (رکورد یا سطر) کم و یا زیاد نشود و از طرف دیگر تجزیه R میباید، وابستگی های تابعی را حفظ کند.

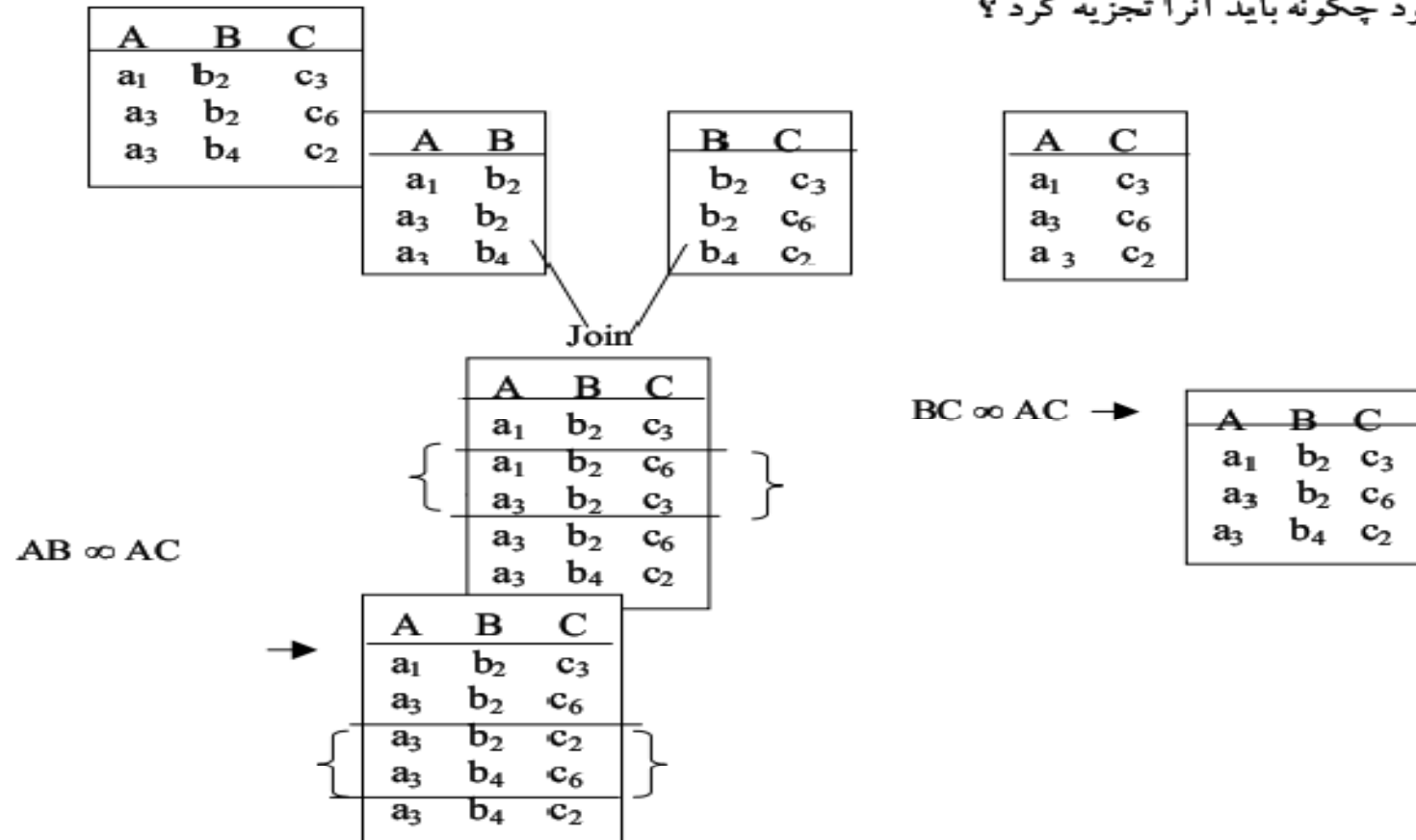


فرایند نرمال سازی

26

رابطه $R(A,B,C)$ را در نظر می گیریم . در یک لحظه از حیات رابطه ، بسط آن چنین است فرض کنیم که این رابطه

باید تجزیه شود چگونه باید آنرا تجزیه کرد ؟



می بینیم که تجزیه R بصورت $R:(AB,AC)$ یا $R:(AB,BC)$ مناسب نیست زیرا با پیوند تجزیه ها تا پیل افزونه بروز می کند اما در تجزیه $R:(BC,AC)$ این پدیده نامطلوب را در پی ندارد لذا این تجزیه مناسب است.



مزایا و معایب نرمال سازی

27

مزایای فرایند نرمال سازی :

- ❑ سازماندهی کلی بهتر برای بانک اطلاعاتی
- ❑ کاهش داده‌های تکراری (حذف افزونگی)، بالا بردن سرعت پرسجوها و بهینه‌سازی مصرف فضای دیسک
- ❑ پرهیز از درج نامطلوب، بروزرسانی و حذف امور غیرنرمال.
- ❑ سازگاری داده‌های درون بانک اطلاعاتی با اعمال آسان محدودیت‌های جامعیت
- ❑ مدیریت بهتر امنیت بانک اطلاعاتی
- ❑ طراحی انعطاف‌پذیر بانک اطلاعاتی

معایب روش نرمال‌سازی

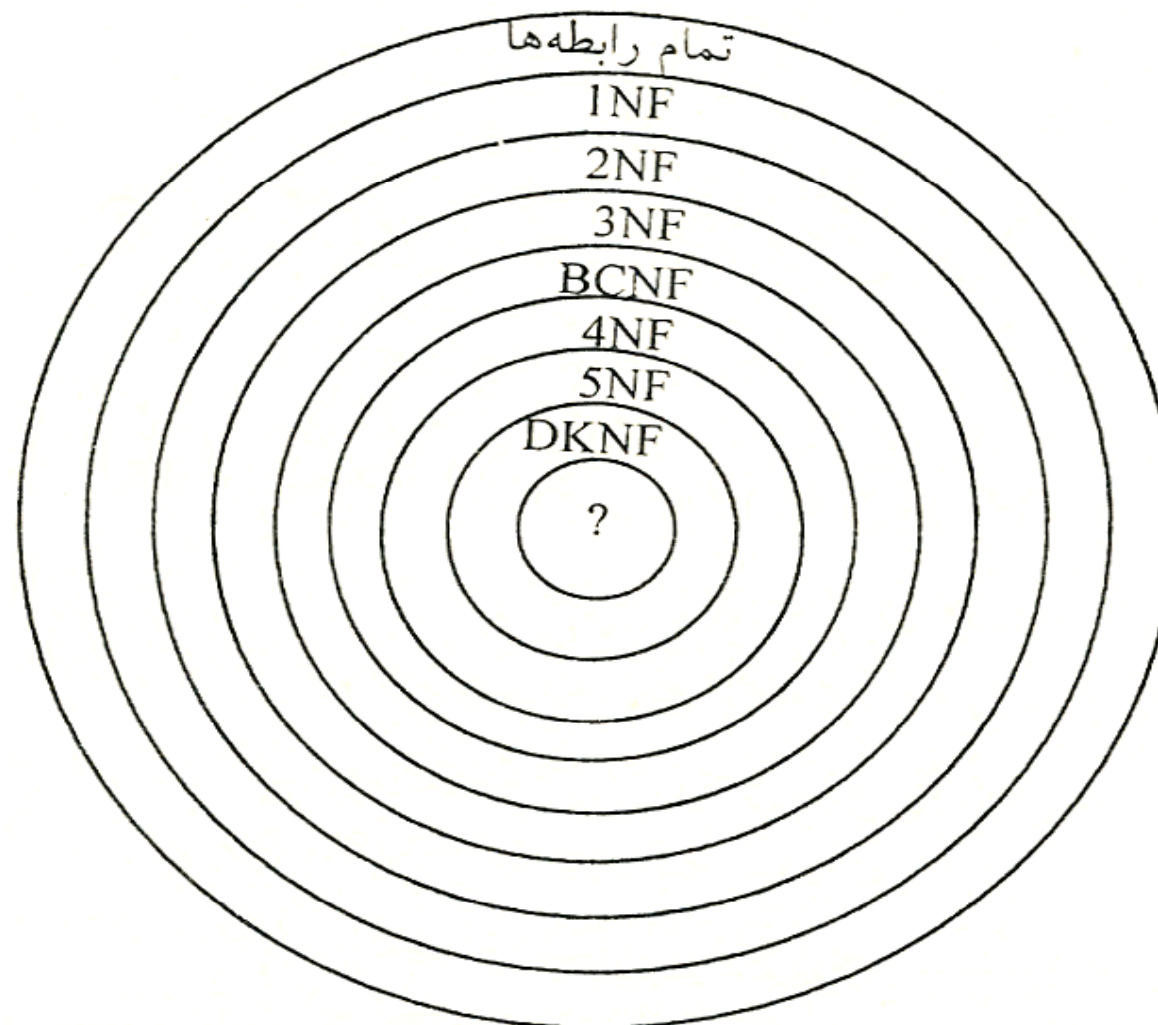
- ❑ بروز فزونکاری در سیستم در عمل بازیابی
- ❑ ایجاد نوعی افزونگی از نوع افزونگی در سطح ادراکی
- ❑ زمانگیر بودن فرآیند نرمال‌سازی به ویژه اگر محیط عملیاتی بزرگ و تعداد رابطه‌ها زیاد باشد.
- ❑ تصمیم‌گیری دشوار در اثر تعدد تجزیه‌ها
- ❑ ممکن است وابستگی بین مجموعه صفات یک خردجهان، وابستگی‌های به جز وابستگی تابعی و پیوندی باشد و بنابراین سبب مطرح شدن قواعد جامعیت دیگر و نیز ضوابط دیگری برای تجزیه رابطه‌ها شود و . . .



فرم های نرمال

درجات مختلف نرمال سازی :

فرم های نرمال، روشی برای اندازه گیری سطوح یا عمقی است که بانک اطلاعاتی تا بدان حد نرمال شده است





فرم نرمال درجه اول 1NF (First Normal Form)

29

رابطه R به فرم 1NF (First Normal Form) است اگر و فقط اگر:

- ۱ - همه کلیدهای آن تعریف شده باشند.
- ۲ - تمام صفات خاصه آن روی میدانهای اتمیک تعریف شده باشند به عبارتی دیگر صفت های آن از دامنه تودرتو (Nested Domain) نباشد. یعنی صفت ترکیبی نداشته باشیم. به عبارت ساده تر هر سلول از جدول فقط و فقط یک مقدار معین داشته باشد. مثل فیلد نام شامل (نام و نام فامیلی) نباشد یا فیلد آدرس بدون توجه به اجزاء شهر - خیابان و ... نباشد.

مثال : رابطه از درجه یک نرمال :

First (S#	P#	QTY	CITY)
S1	P1	100	C1
S1	P2	80	C1
S2	P3	90	C2
S2	P1	60	C2
S2	P2	80	C2
S3	P1	70	C2
S4	P1	120	C1



فرم نرمال درجه اول 1NF (First Normal Form)



مثال: جدول ST را در نظر بگیرید:
کلید اصلی جدول ، ST# می باشد.

St#	Name	میزان تحصیلات				
۷۸۰۱	علی	۱۴۰۰	پایگاه داده	3	20	79-2
		۱۵۰۰	ریاضی	3	10	80-1
		۱۶۰۰	تجزیه و تحلیل	3	20	80-1
۷۹۰۲	آرش	۱۴۰۰	پایگاه داده	3	7	80-1
		۱۷۰۰	تربیت بدنی	1	20	80-1

برای تبدیل این جدول به حالت نرمال ۱ ، باید آن را به فرم زیر تبدیل کنیم :
کلید اصلی جدول term + Crs# + St# می باشد .

St#	Name	Crs#	Course	unit	Grade	term
7801	علی	۱۴۰۰	پایگاه داده	۳	۲۰	79-2
7801	علی	۱۵۰۰	ریاضی ۱	۳	۱۰	80-1
7801	علی	۱۶۰۰	تجزیه و تحلیل	۳	۲۰	80-1
7902	آرش	۱۴۰۰	پایگاه داده	۳	۷	80-1
7902	آرش	۱۷۰۰	تربیت بدنی	۱	۲۰	80-1



فرم نرمال درجه دوم 2NF (Second Normal Form)

31

یک رابطه از نوع فرم نرمال درجه دوم است اگر و فقط اگر از :

۱- درجه اول نرمال باشد.

۲- تمام صفتهای خاصه ی (فیلدهای) آن به کلید اصلی وابستگی تابعی (FD) داشته باشند.

۳- کلیه صفات غیر کلیدی (غیر کلید اصلی و کلید کاندید) به زیرمجموعه های کلید اصلی وابستگی نداشته باشند.

(وابستگی جزئی Partial dependency)

جدول ۱

کد دانشجو	نام	فامیلی	ش.ش	سال تولد	نام پدر	آدرس	تلفن
۲	علی	محمدی	۱۱	۱۳۶۰	محمد	رشت	۲۳۳۲
۵	محمد	ناصری	۱۱	۱۳۶۲	رضا	زنجان	۳۲۵۱
۳	رضا	رضایی	۴۵۱	۱۳۶۰	احمد	قزوین	۲۳۲۴

در جدول بالا کد دانشجو کلید اصلی است و نام ، فامیلی ، ش.ش ، سال تولد ، نام پدر ، آدرس ، تلفن صفات غیر کلیدی و هر کدام به تنهایی به کلید اصلی وابسته اند. پس جدول بالا از درجه دوم نرمال است.

الگوریتم زیر ، جدول 1NF را به جدول 2NF تجزیه می کند :

۱- هر بخش از کلید اصلی را که ایجاد وابستگی جزئی کرده است ، با همه وابسته های آن کنار بگذارد.

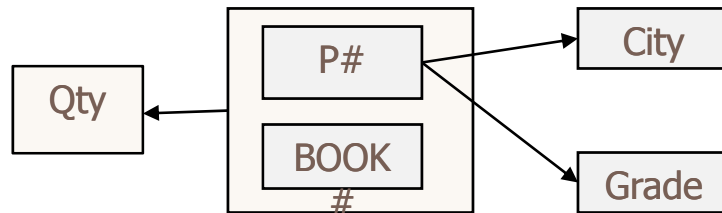
۲- کل کلید اصلی را با صفت های باقیمانده کنار هم بگذارد.

۳- صفت های کلیدی را به عنوان کلید خارجی به ۲ اضافه کن .



فرم نرمال درجه دوم 2NF (Second Normal Form)

32



مثال: رابطه وقتی در سطح 1NF باقی می ماند که وابستگی تابعی کامل بین حداقل یک فیلد با کلید اصلی نقض شود. به عنوان مثال نمودار FIRST را در نظر بگیرید:

کلید اصلی در این رابطه صفت مرکب (P#, BOOK#) است. از آنجایی که تمام فیلدها کلید FD دارند؛ داریم:

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

$BOOK# \not\rightarrow CITY, P# \not\rightarrow CITY$

$P# \rightarrow CITY$

برای برقراری وابستگی تابعی کامل باید داشته باشیم:

اما همانطور که در نمودار FD قابل مشاهده است داریم:

در نتیجه FD کامل بین CITY و (P#, BOOK#) نقض شده است.

برای رفع آنومالی و افزایش سطح نرمال باید رابطه FIRST به صورت زیر تجزیه شود :

SECOND (P# , CITY , GRADE)

PB (P# , BOOK# , QTY)

P#	CITY	GRADE
P1	C1	40
P2	C2	60
P3	C1	40
P1	C1	40
P4	C3	50
P4	C3	50

P#	BOOK#	QTY
P1	B1	1000
P2	B4	3000
P3	B3	2000
P1	B2	3000
P4	B3	4000
P4	B5	3500

رابطه SECOND در سطح دوم نرمال قرار دارد زیرا ، ۱- در 1NF است ۲- وابستگی تابعی کامل نقض نشده اما این رابطه نیز دارای آنومالی است.



فرم نرمال درجه دوم 2NF (Second Normal Form)

33

St

St#	Name	Crs#	Course	Unit	Grade	Term
7801	علی	۱۴۰۰	پایگاه داده	۳	۲۰	79-2
7801	علی	۱۵۰۰	ریاضی ۱	۳	۱۰	80-1
7801	علی	۱۶۰۰	تجزیه و تحلیل	۳	۲۰	80-1
7902	آرش	۱۴۰۰	پایگاه داده	۳	۷	80-1
7902	آرش	۱۷۰۰	تربیت بدنی	۱	۲۰	80-1

برخی از جداول در حالت نرمال ۱ هستند ولی هنوز ناهنجاری دارند. مثلاً با آن که جدول St به نرمال ۱ تبدیل شد، ولی هنوز دارای ناهنجاری های زیر است:

۱- ناهنجاری در حذف:

فرض کنید علی تنها دانشجویی باشد که درس ریاضی ۱ را گرفته است. اگر لازم باشد اطلاعات علی را حذف کنیم، به طور ناخواسته اطلاعات ریاضی ۱ را نیز از دست می دهیم.

۲- ناهنجاری در درج:

فرض کنید بخواهیم اطلاعات دانشجوی جدیدی را که هنوز هیچ درسی را نگرفته است درج کنیم. از آنجا که Course و Term جزئی از کلید اصلی هستند نمی توانند مقدار تهی داشته باشند. انجام این عمل ممکن نیست.

۳- ناهنجاری در اصلاح:

فرض کنید بخواهیم تعداد واحد درس پایگاه داده را از ۳ به ۴ تغییر دهیم. در این صورت برای حفظ سازگاری داده ها لازم است این تغییر را به دفعات مکرر و در تاپل های مختلف اعمال کنیم (اصلاح منتشر شونده).



فرم نرمال درجه دوم 2NF (Second Normal Form)

34

مثال : جدول St که در مثال قبل تبدیل به نرمال ۱ شد ، نرمال ۲ نیست چرا که کلید اصلی این جدول ، $St\# + Crs\# + Term$ است در حالی که تنها با داشتن $St\#$ می توان به Name رسید و یا تنها با داشتن $Crs\#$ می توان به Cname و Unit رسید. برای تبدیل یک جدول نرمال ۱ به نرمال ۲ مراحل زیر را دنبال کنید :

۱- کلیه ترکیبات ممکن میان اجزا کلید اصلی را به ترتیب در سطرهای مجزا بنویسید (اول ترکیبات یک جزیی ، سپس ترکیبات دو جزیی ، سپس ترکیبات سه جزیی و ...) :

$St\# , Crs\# , Term , St\# + Crs\# ,$
 $St\# + Term , Crs\# + Term , St\# + Crs\# + Term$

۲- در کنار هر یک از ردیف ها ، ویژگی هایی را که در تشکیل کلید اصلی نقشی ندارند و به ویژگی های مورد نظر وابستگی تابعی دارند و در سطرهای بالاتر نوشته نشده اند بنویسید . مثلاً با داشتن یک $St\#$ تنها به یک Name می رسیم ، پس Name به $St\#$ وابستگی تابعی دارد . Name را در ردیف اول اضافه کرده و در ردیف های بعدی ، آن را در نظر نمی گیریم :

$St\#$, Name

$Crs\#$, Cname , Unit

Term

$St\# + Crs\#$

$St\# + Term$

$Crs\# + Term$

$St\# + Crs\# + Term$, Grade



فرم نرمال درجه دوم 2NF (Second Normal Form)

35

۳- کلیه ردیف هایی را که هیچ ویژگی غیر کلیدی در آنها وجود ندارد و شامل هیچ اطلاعات مفیدی که در جداول دیگر نیز وجود ندارد حذف کرده و سایر سطرها را به جدول مجزا تبدیل کنید .

Student (St# , Name)

Course (Crs# , Cname , Unit)

SC (St# , Crs# , Term , Grade)

به این ترتیب جدول St به جداول زیر تبدیل خواهد شد :

Student

St#	Name
7801	علی
7902	آرش

Course

Crs#	Cname	Unit
۱۴۰۰	پایگاه داده	۳
۱۵۰۰	ریاضی ۱	۳
۱۶۰۰	تجزیه و تحلیل	۳
۱۷۰۰	تربیت بدنی	۱

SC

St#	Crs#	Term	Grade
۷۸۰۱	۱۴۰۰	79-2	20
۷۸۰۱	۱۵۰۰	80-1	10
۷۸۰۱	۱۶۰۰	80-1	20
۷۹۰۲	۱۴۰۰	80-1	7
۷۹۰۲	۱۷۰۰	80-1	20



فرم نرمال درجه دوم 2NF (Second Normal Form)

مثال :

نمره	نمره قبولی	تعداد واحد	نام درس	کد درس	کد دانشجو
۱۶	۱۰	۲	عربی	۲۵۱	۲
۱۵	۱۰	۳	ریاضی	۳۶۳	۵
۹	۱۰	۲	شیمی	۲۵۶	۱
۱۷	۱۰	۳	ریاضی	۳۶۳	۲
۱۶	۱۰	۲	عربی	۲۵۱	۵
۱۵	۱۰	۲	عربی	۲۵۱	۳
۱۳	۱۰	۳	ریاضی	۳۶۳	۱
۱۸	۱۰	۲	شیمی	۲۵۶	۳

در جدول روبرو کد دانشجو و کددرس (هر دو با هم) کلید اصلی هستند.
نام درس، تعداد واحد و نمره قبولی به کد درس وابسته است و نمره به کلید اصلی (کد درس و کد دانشجو) وابسته است.
پس جدول بالا از درجه دوم نرمال نیست.

جدول دوم بصورت زیر نرمال شده و به دو جدول تقسیم میشود:

2A

کد درس	نام درس	تعداد واحد	نمره قبولی
۲۵۱	عربی	۲	۱۰
۳۶۳	ریاضی	۳	۱۰
۲۵۶	شیمی	۲	۱۰

در جدول ۲A ، کددرس ، کلید اصلی است و در جدول ۲B کد دانشجو و کد درس باهم کلید اصلی هستند (کد درس در این جدول کلید خارجی ست)
بنابر این حالا هر دو جدول ما نرمال هستند .

2B

نمره	کد درس	کد دانشجو
۱۶	۲۵۱	۲
۱۵	۳۶۳	۵
۹	۲۵۶	۱
۱۷	۳۶۳	۲
۱۶	۲۵۱	۵
۱۵	۲۵۱	۳
۱۳	۳۶۳	۱
۱۸	۲۵۶	۳



فرم نرمال درجه سوم 3NF (Third Normal Form)

37

یک رابطه به شکل سوم نرمال است اگر و فقط اگر اولاً از درجه دوم باشد و ثانیاً تمامی صفات غیرکلیدی به یکدیگر وابسته نباشد (یعنی وابستگی انتقالی نداشته باشند).

مثال:

کد کالا	کد فنی	شرح کالا	قیمت	کد رنگ	نام رنگ
۱	3AB	مداد	۱۵۰	۴	قرمز
۲	5BF	خودکار	۲۵۰	۳	آبی
۴	7RF	صندلی	۳۵۰	۱	مشکی
۶	4PO	مداد	۷۵	۱	مشکی
۵	3KJ	میز	۸۵۰	۴	قرمز
۳	7NS	تراش	۴۰۰	۲	سبز

در جدول بالا کد فنی، شرح کالا، قیمت و کد رنگ همه به کد کالا وابسته هستند ولی نام رنگ هم به کد کالا وابسته است و هم به کد رنگ، بنابراین، این جدول از درجه دوم نرمال ولی از درجه سوم نرمال نیست بخاطر اینکه نام رنگ هم به کلید اصلی و هم به صفت غیرکلید یعنی همان کد رنگ وابسته است. بنابر این جدول بالا را به فرم درجه ۳ میتوان نرمال کرد.



فرم نرمال درجه سوم 3NF (Third Normal Form)

38

نام رنگ	کد رنگ	قیمت	شرح کالا	کد فنی	کد کالا
قرمز	۴	۱۵۰	مداد	3AB	۱
آبی	۳	۲۵۰	خودکار	5BF	۲
مشکی	۱	۳۵۰	صندلی	7RF	۴
مشکی	۱	۷۵	مداد	4PO	۶
قرمز	۴	۸۵۰	میز	3KJ	۵
سبز	۲	۴۰۰	تراش	7NS	۳

کد رنگ	قیمت	شرح کالا	کد فنی	کد کالا
۴	۱۵۰	مداد	3AB	۱
۳	۲۵۰	خودکار	5BF	۲
۱	۳۵۰	صندلی	7RF	۴
۱	۷۵	مداد	4PO	۶
۴	۸۵۰	میز	3KJ	۵
۲	۴۰۰	تراش	7NS	۳

نام رنگ	کد رنگ
قرمز	۴
آبی	۳
مشکی	۱
سبز	۲

تذکر: در جدول قبل کد کالا و کد فنی کلید های کاندید هستند، که می توان یکی از آنها را به دلخواه کلید اصلی انتخاب کرد، که ما در اینجا کد کالا را کلید اصلی در نظر می گیریم.

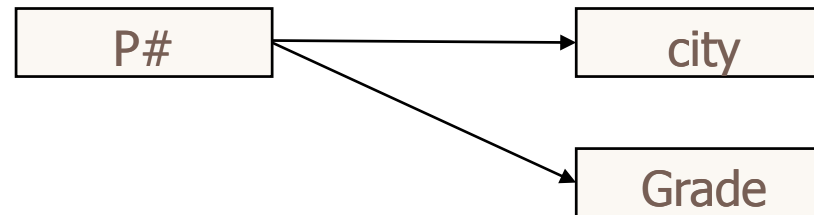


فرم نرمال درجه سوم 3NF (Third Normal Form)

39

مثال: رابطه SECOND را در نظر می گیریم. نمودار FD آن به صورت زیر است:

- 1) $P\# \longrightarrow city$
- 2) $city\# \longrightarrow Grade$
- 3) $P\# \longrightarrow Grade$



از دو وابستگی تابعی اول و دوم می توان وابستگی تابعی سوم را نتیجه گرفت، یعنی Grade وابستگی تابعی با واسطه $P\#$ از طریق city دارد.

علت بروز آنومالی Second نیز همین عامل یعنی وابستگی تابعی با واسطه است.

برای رفع آنومالی آن می باید این رابطه تجزیه شود. بنابراین Second را به رابطه های PC و CS تجزیه می کنیم:

PC ($P\#$, City)

CS (City , Grade)

PS

P#	City
P1	C1
P2	C2
P3	C1
p4	c3

CS

City	Grade
C1	40
C2	60
C3	50

با حذف آنومالی های Second، روابط جدید نه تنها در سطح 3NF قرار می گیرند؛ بلکه در سطح BCNF قرار می گیرند.



فرم نرمال درجه سوم 3NF (Third Normal Form)

40

برخی از جداول، در حالت ۲ نرمال هستند ولی هنوز مشکلاتی دارند. به عنوان مثال جدول Professor که شامل شماره، نام، کد آخرین مدرک تحصیلی و نام آخرین مدرک تحصیلی است؛ در نظر بگیرید:

Prof

Prof#	Pname	LastDegree#	LastDegreeName
100	علی راد	2	کارشناسی ارشد
101	آرش رضایی	2	کارشناسی ارشد
102	عسل شاملو	3	دکترا

این جدول، نرمال ۲ است زیرا نرمال ۱ است و آنجا که کلید اصلی تنها یک جز دارد، بدون شک هیچ وابستگی جزئی در آن وجود ندارد. با این جود این جدول هنوز دارای ناهنجاری های زیر است:

۱- ناهنجاری در حذف:

فرض کنید عسل شاملو تنها کسی باشد که مدرک دکترا دارد. در این صورت اگر بخواهیم اطلاعات عسل شاملو را حذف کنیم، اطلاعات مربوط به مدرک دکترا (کد مدرک و نام مدرک) را نیز از دست می دهیم.

۲- ناهنجاری در اصلاح:

اگر بخواهیم نام مدرک ۲ را از کارشناسی ارشد به کارشناسی تغییر دهیم، در این صورت مجبوریم این تغییر را در تاپل های مختلف تکرار کنیم (اصلاح منتشر شونده).



فرم نرمال درجه سوم 3NF (Third Normal Form)

41

جدول Professor نرمال ۳ نیست چون : $\text{LastDegree\#} \longrightarrow \text{LastDegreeName}$

برای تبدیل جداول نرمال ۲ به نرمال ۳ ، مراحل زیر را دنبال کنید :

۱- ویژگی هایی را که در وابستگی متعددی شرکت دارند را در یک جدول مجزا قرار داده و ویژگی هایی را که در طرف چپ این وابستگی قرار دارند ، به عنوان کلید اصلی معرفی کنید .

Degree (LastDegree# , LastDegreeName)

۲- بقیه ویژگی های جدول اولیه را در جدول مجزای دیگری قرار داده (Pname,prof#) ، ویژگی های تشکیل دهنده کلید

اصلی در جدول اول (lastDegree#) را به آنها اضافه کنید. Prof(Prof# , Pname , LastDegree#)

Degree

Lastdegree#	LastDegreeName
۲	کارشناسی ارشد
۳	دکترا

Prof

Prof#	Pname	LastDegree#
100	علی راد	2
101	آرش رضایی	2
102	عسل شاملو	3

الگوریتم تبدیل روابط نرمال از درجه ۲ به نرمال درجه ۳ :

۱- صفتی که وابستگی انتقالی ایجاد کرده با دیگر صفات وابسته آنرا در کنار هم بگذار.

۲-کلید اصلی را با صفت های باقیمانده کنار هم بگذار.

۳- صفت های کلیدی را به عنوان کلید خارجی به ۲ اضافه کن.



فرم نرمال BCNF (BOYCE_CODD Normal Form)

42

در دیدگاه آقای کاد، اینجا پایان راه است. آقای بویس روی یک نکته انگشت گذاشت و فرم نرمال خود BCNF را تعریف کرد.

(BOYCE_CODD NORMAL FORM) این نکته در زیر بررسی می شود.

3NF با جداولی که هر سه شرط زیر را دارند دچار مشکل می گردد :

۱- جدول حداقل دارای دو کلید کاندید باشد.

۲- این کلیدهای کاندید ترکیبی باشند.

۳- این کلیدهای ترکیبی ، صفت های مشترکی داشته باشند.

مثال: جدول دانشجو در یک موسسه کوچک که دانشجوی همنام ندارد (در این صورت هم شماره دانشجویی کلید کاندیدا است و هم نام دانشجو) :

Stud (s# , Sname , address , avg)

Candidate key (S#)

Candidate key (Sname)

وابستگی ها :

S# —————> Stud

Sname —————> Stud

این جدول در 3NF هست. از سه شرط بالا فقط شرط ۱ را داراست. بنابراین نیاز به نرمال سازی بیشتر ندارد. با در نظر گرفتن داده های مناسب می توان دید که این جدول افزونگی ندارد.



فرم نرمال BCNF (BOYCE_CODD Normal Form)

43

مثال ۱: جدول "درس - دانشجو" در همین موسسه :

Crs_stud (s# , sname , c# , score)

Candidate key (s# , c#)

Candidate key (sname , c#)

وابستگی ها:

(s# , c#) \longrightarrow crs_stud

(sname , c#) \longrightarrow crs stud

S# \longrightarrow sname

Sname \longrightarrow s#

Crs_stud

S#	Sname	C#	Score
S1	علی	C1	17
S1	علی	C2	12
S1	علی	C3	19
S1	علی	C4	20
S1	علی	C5	10

این جدول هر سه شرط بالا را داراست. بنابراین ممکن است نیاز به نرمال سازی بیشتر داشته باشد. داده های روبرو نشان می دهد که این جدول دارای افزونگی است و برای برطرف نمودن آن نیازی به نرمال سازی بیشتر دارد و می توان این جداول را به دو جدول :

st(s# , sname) و grade (s# , c# , score) شکست.

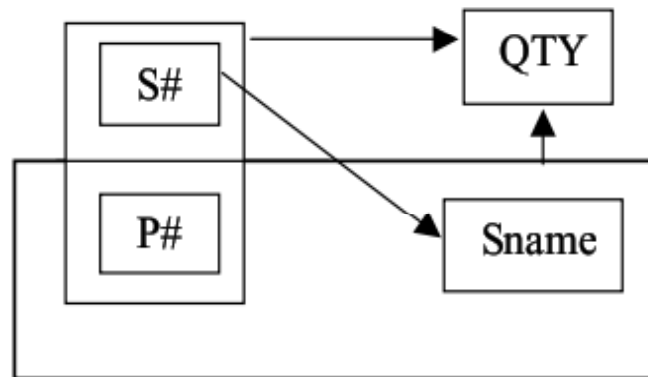
در جدول grade می توان به جای s# از sname نیز استفاده کرد.



فرم نرمال BCNF (BOYCE_CODD Normal Form)

44

مثال ۲ رابطه SPS را در نظر بگیریم : (S# , P# , SNAME , QTY)
همپوشانی دارند . نمودار وابستگی تابعی بفرم زیر است :



این رابطه BCNF نیست زیرا S# دترمینان است ولی کلید کاندید نیست . می خواهیم بررسی کنیم این رابطه چند NF است .

♦ این رابطه 1NF است زیرا صفات خاصه اش اتمیک هستند .

♦ 2NF نیز می باشد زیرا وابستگی ناکامل نداریم .

البته ظاهراً به نظر می رسد وابستگی ناکامل وجود دارد ولیکن اینطور نیست زیرا Sname خود جزئی از کلید کاندید است در حالیکه در تعریف 2NF کادی آمده است هر صفت خاصه غیر کلید و اصلاً عضویت صفت خاصه در کلید کاندید مطرح نیست . لذا 2NF می باشد .

♦ این رابطه 3NF نیز می باشد زیرا تعدی وجود ندارد .



فرم نرمال BCNF (BOYCE_CODD Normal Form)

45

می‌بینیم 3NF است ولی BCNF نیست. نکته جالبتر آنکه رابطه sps اختلاط اطلاعاتی دارد با این همه با داشتن دو کلید کاندید 3NF است در حالیکه معمولاً وجود پدیده اختلاط اطلاعاتی رابطه را در حد 1NF یا حداکثر 2NF نگه می‌دارد.

نتیجه: صرف گفتن رابطه‌ای اختلاط اطلاعاتی دارد لزوماً معنایش این نیست که سطح نرمالیتی آن پایین است. در عمل برای طراحی رابطه‌ها تا سطح BCNF نرمال می‌شوند. سطوح بالاتر بیشتر جنبه تئوریک و پژوهشی دارد و معنایش این است که تقریباً تمام رابطه‌هایی که BCNF هستند عملاً 5NF و 4NF هستند بعبارت دیگر رابطه‌هایی که BCNF باشد اما 4NF و 5NF نباشند بسیار کم‌اند.

مثال ۳: جدول امتحان شامل شماره دانشجویی، شماره درس، رتبه دانشجو در درس (غیر تکراری):

Exam (s# , subj# , rank)

Candidate key (s# , subj#)

Candidate key (subj# , rank)

وابستگی‌ها:

$\xrightarrow{\hspace{1cm}}$
 Key (s# , subj#) Exam
 $\xrightarrow{\hspace{1cm}}$
 Key (subj# , rank) Exam

این جدول در 3NF هست و هر سه شرط را نیز داراست، ولی نیاز به نرمال سازی بیشتر ندارد (با در نظر گرفتن داده‌هایی می‌توان دید که این جدول افزونگی ندارد).

چگونه باید فهمید که چنین جداولی نیاز به نرمال سازی دارند یا خیر؟ پاسخ این سوال را باید در تعریف BCNF جستجو کرد. اگر جدولی در فرم BCNF باشد، نیاز به شکستن ندارد.



فرم نرمال BCNF (BOYCE_CODD Normal Form)

46

تعریف: جدولی BCNF است که ستون های آن فقط به کلید های کاندیدش وابستگی تابعی داشته باشد. در مثال ۱ و ۳ ، وابستگی به غیر کلید های کاندیدا وجود ندارد ولی در مثال ۲ این نوع وابستگی وجود دارد

$(sname \longrightarrow s\# , s\# \longrightarrow sname)$.

توجه به نکات زیر در مورد BCNF ضروری است :

۱ - برخلاف فرم های نرمال دیگر ، BCNF بدون استفاده از 3NF و فرم های قبلی نرمال تعریف می شود. غالباً می توان بانک اطلاعات را با استفاده از تعریف BCNF در یک قدم نرمال کرد و نیازی به تعریف وابستگی هایی از قبیل وابستگی انتقالی نیست .

بنابراین ، BCNF جامع ترین تعریف نرمال سازی بر مبنای وابستگی تابعی را به طور مستقل ارائه می دهد.

۲- در مواردی ، نرمال سازی تا BCNF لازم نیست و بهتر است از تبدیل جدول فرم 3NF به BCNF خودداری کرد.



فرم نرمال BCNF (BOYCE_CODD Normal Form)

47

به عنوان مثال ، برای ثبت آدرس فعلی و آدرس خانواده دانشجویان می توان جدول زیر را تعریف کرد و برای هر دانشجو بیش از یک آدرس در نظر گرفت :

S_addr (s# , city , no , zip)

Candidate key(s# , city)

Candidate key(s# , zip)

کد پستی برای هر شهر منحصر به فرد است و می تواند به جای نام شهر قرار بگیرد. این جدول در فرم 3NF هست ولی در فرم BCNF نیست زیرا داریم : $Zip \longrightarrow city$

با این همه نمی توان با اطمینان خاطر این جدول را شکست زیرا کد پستی ، بخش جدایی ناپذیر است و جدا کردن آن باعث پیچیده شدن پرس و جو های مربوط به آدرس می شود.

چون وابستگی $city \longrightarrow zip$ نیز برقرار است ، همین مطلب در مورد 2NF هم صادق است. به طور خلاصه باید گفت که پرس و جو نیز می تواند در طراحی جداول نقش داشته باشد. اگر طراح تشخیص دهد که تجزیه یک جدول ، هرچند افزونگی هم داشته باشد ؛ باعث پایین آمدن سرعت اکثر پرس و جوها می شود ، مجاز است از نرمال تر سازی آن صرف نظر کند .



فرم نرمال BCNF (BOYCE_CODD Normal Form)

48

برخی از جداول در حالت نرمال ۳ هستند ولی هنوز ناهنجاری دارند. به عنوان مثال دانشگاهی با قوانین زیر را در نظر بگیرید :

۱- هر دانشجو می تواند در چند رشته تحصیلی ، تحصیل کند ولی در هر یک از رشته های تحصیلی ، تنها یک استاد راهنما دارد.

۲- هر استاد تنها می تواند در یک رشته تحصیلی تدریس کند ولی در هر رشته تحصیلی چندین استاد وجود دارند.

جدول Project را که شامل شماره دانشجویی ، رشته تحصیلی و استاد راهنمای دانشجو در رشته مربوطه است را در نظر بگیرید :

Project

St#	Field	Tutor
7801	مهندسی کامپیوتر	مجید رضایی
7801	ریاضی محض	آرش ریاضی دان
7801	هنر	گلناز همدوست
7902	مهندسی کامپیوتر	مجید رضایی
8001	مهندسی کامپیوتر	پروین صبا

در جدول Project ، دو کلید کاندیدا وجود دارد :

کلید کاندیدای ۱ : field + St#

کلید کاندیدای ۲ : Tutor + St#

فرض کنید کلید کاندیدای اول را به عنوان کلید اصلی انتخاب کنیم .



فرم نرمال BCNF (BOYCE_CODD Normal Form)

49

جدول Project ، نرمال ۲ است چون در آن هیچ ویژگی غیر کلیدی به قسمتی از کلید اصلی ، وابستگی ندارد (تنها ویژگی غیر کلیدی این جدول ، Tutor است که نه به St# تنها وابسته است و نه به Field تنها) .

این جدول نرمال ۳ نیز هست چون در آن هیچ ویژگی کلیدی به ویژگی غیر کلیدی دیگر ، وابستگی ندارد (چون این جدول تنها یک ویژگی غیر کلیدی دارد ، این مساله بدیهی است) .

با وجود آن که این جدول نرمال ۳ است ، ولی هنوز ناهنجاری هایی دارد :

۱- ناهنجاری در درج :

به عنوان مثال ، نمی توان استاد جدیدی را که هنوز هیچ دانشجویی با وی پروژه نگرفته است ، در جدول درج کرد چون St# جزئی از کلید اصلی است و نمی تواند تهی باشد .

۲- ناهنجاری در حذف :

فرض کنید دانشجوی شماره ۸۰۰۰۱ تنها دانشجویی باشد که با پروین صبا پروژه گرفته است ؛ در این صورت اگر این دانشجو

Project

را حذف کنیم ، اطلاعات مربوط به این استاد نیز از بین می رود .

St#	Field	Tutor
7801	مهندسی کامپیوتر	مجید رضایی
7801	ریاضی محض	آرش ریاضی دان
7801	هنر	گلناز هنردوست
7902	مهندسی کامپیوتر	مجید رضایی
8001	مهندسی کامپیوتر	پروین صبا



فرم نرمال BCNF (BOYCE_CODD Normal Form)

50

Tutor → Field

جدول Project نرمال BCNF نیست چون :

پس Tutor ، یک تعیین کننده است در حالی که کلید کاندیدا نیست.

تبدیل جداول نرمال ۲ به نرمال BCNF کاملاً مشابه روند تبدیل جداول نرمال ۲ به نرمال ۳ است.

برای تبدیل جدول Project به جدول نرمال BCNF :

۱- Tutor و Field را در یک جدول مجزا قرار داده ، Tutor را به عنوان کلید اصلی این جدول معرفی می کنیم.

TC (Tutor , Field)

۲- بقیه ویژگی ها (ST#) را در جدول دیگری قرار داده ، کلید اصلی جدول اول یعنی Tutor را به آن ها اضافه می کنیم.

TC

ST (St# , Tutor)

ST

Tutor	Field
مجید رضایی	مهندسی کامپیوتر
آرش ریاضی دان	ریاضی محض
گلناز هردوست	هنر
مجید رضایی	مهندسی کامپیوتر
پروین صبا	مهندسی کامپیوتر

St#	Tutor
7801	مجید رضایی
7801	آرش ریاضی دان
7801	گلناز هردوست
7902	مجید رضایی
8001	پروین صبا

تذکر: حالت های نرمال ۳ و نرمال BCNF معمولاً با هم معادلند. تنها در صورتی که جدول بیش از یک کلید کاندیدا داشته باشد و میان

کلید کاندیدا ، ویژگی مشترک وجود داشته باشد ، این دو حالت با هم معادل نخواهند بود. مثلاً در جدول Project ، دو کلید کاندیدا

وجود دارد و ST# میان دو کلید کاندیدا مشترک است. به همین دلیل حالت نرمال ۳ و BCNF این جدول معادل نیستند.

مثال: در بانک اطلاعاتی زیر ابتدا کلید کاندیدا را یافته و سپس آن را نرمال سازی کنید.

$R = \{A, B, C, D, E, F, G\}$

$F = \{AF \longrightarrow BE, FC \longrightarrow DE, F \longrightarrow CD, D \longrightarrow E, C \longrightarrow A\}$

حل: ابتدا سمت راست وابستگی ها را به صفت تبدیل می کنیم:

$AF \longrightarrow B$

$AF \longrightarrow E$

$FC \longrightarrow D$

$FC \longrightarrow E$

$F \longrightarrow C$

$F \longrightarrow D$

$D \longrightarrow E$

$C \longrightarrow A$

از شماره ۵ برای ساده کردن سمت چپ ۳ و ۴ استفاده می کنیم. در نتیجه خواهیم داشت :

3) $F \longrightarrow D$ 4) $F \longrightarrow E$

از شماره ۵ و ۸ می توان نتیجه گرفت $F \longrightarrow A$ و این نتیجه را بر روی ۱ و ۲ اعمال می کنیم :

1) $F \longrightarrow B$ ۲) $F \longrightarrow E$

در نتیجه داریم :

$F = \{F \longrightarrow A, F \longrightarrow B, F \longrightarrow C, F \longrightarrow D, F \longrightarrow E, D \longrightarrow E, C \longrightarrow A\}$

در نتیجه F همه صفت ها دیگر به جز G را می دهد. پس (F, G) کلید کاندیدا است.

1NF : $\underline{F, G}, A, B, C, D, E$

2NF : $(\underline{F, G})$

$\underline{E}, A, B, C, D, E$

3NF : $(\underline{F, G})$

(\underline{C}, A)

(\underline{D}, E)

(\underline{F}, B, C, A)



مثالی از فرم های نرمال

52

مثال : در بانک اطلاعاتی زیر ابتدا کلید های کاندید را بیابید و سپس آنرا بطور کامل نرمال سازی نمایید :

$$R = \{ A, B, C, D, E, F, G \}$$

$$F = \{ AF \rightarrow BE, FC \rightarrow DE, F \rightarrow CD, D \rightarrow E, C \rightarrow A \}$$

حل :

سمت راست وابستگی ها را به یک صفت تبدیل می کنیم.

- 1) $B \rightarrow AF$
- 2) $AF \rightarrow E$
- 3) $FC \rightarrow D$
- 4) $FC \rightarrow E$
- 5) $F \rightarrow C$
- 6) $F \rightarrow D$
- 7) $D \rightarrow E$
- 8) $C \rightarrow A$

از شماره ۵ می توانیم برای ساده کردن سمت چپ شماره های ۳ و ۴ استفاده کنیم . در نتیجه :

$$4) F \rightarrow E, \quad 3) F \rightarrow D$$

از شماره های ۵ و ۸ می توان نتیجه گرفت که $F \rightarrow A$



مثالی از فرم های نرمال

53

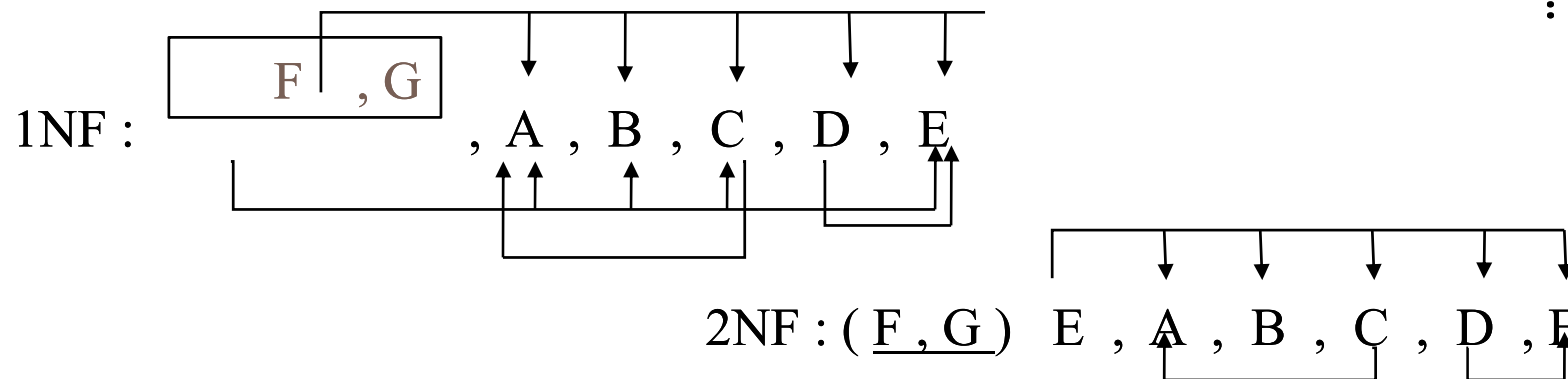
این نتیجه را روی شماره های ۱ و ۲ اعمال می کنیم :

1) $F \rightarrow B$, 2) $F \rightarrow E$

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

در نتیجه صفت F همه صفت های دیگر ، بجز G ، را می دهد ، پس (F , G) کلید کاندید است . این کلید کاندید منحصر به فرد است زیرا هیچ صفتی F و G را نمی دهد (یعنی در هر کلید کاندید ، این دو صفت لازم هستند) .

نرمال سازی :



3NF : (F, G)

(C , A)

(D , E)

(F , B , C , D)

نیازی به BCNF نیست زیرا بیش از یک کلید کاندید نداریم.



مثالی از فرم های نرمال

54

مثال : رابطه $R = \{ X, Y, Z, S, T, U, W \}$ را با FD های زیر بطور کامل نرمال سازی نمایید :

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

$$S \rightarrow X, X \rightarrow Y \Rightarrow S \rightarrow XY$$

$$S \rightarrow XY, XY \rightarrow TUZ \Rightarrow S \rightarrow T, S \rightarrow U, S \rightarrow Z \Rightarrow (S, W)$$

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

$$2\text{NF} : (\underline{S}, W)$$

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

$$3\text{NF} : (\underline{S}, W)$$

$$(T, Y)$$

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

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

نیازی به BCNF نیست زیرا بیش از یک کلید کاندید نداریم .



فرم نرمال درجه چهارم 4NF

تئوری فاگین : فرض کنید مجموعه $R\{A,B,C\}$ یک متغیر رابطه ای باشد که در آن A,B,C مجموعه هایی از صفات هستند. آنگاه R برابر با الحاق دو تصویرش بر روی $\{A,B\}$ و $\{A,C\}$ است اگر و فقط اگر R وابستگی چند مقداری $A \rightarrow \rightarrow B \mid C$ را برآورده کند.

□ متغیر رابطه ای R به **شکل چهارم نرمال** است اگر و فقط اگر، هرگاه زیر مجموعه های A و B از صفات R موجود باشند که غیر جزئی باشند، وابستگی چند مقداری $A \rightarrow \rightarrow B$ برآورده شود. در این صورت تمام صفات R نیز به طور تابعی به A وابستگی دارند.

به عبارت دیگر، تنها وابستگی های غیر جزئی در R به شکل $K \rightarrow X$ هستند (یعنی وابستگی تابعی از یک فوق کلید به صفت دیگر X).

وابستگی چند مقداری :

فرض کنید R یک متغیر رابطه ای و A, B و C زیر مجموعه ای از صفات R باشند. آنگاه وابستگی چند مقداری B بر روی A به صورت زیر نمایش می دهیم :

$$A \rightarrow \rightarrow B$$

(این نماد به این صورت خوانده می شود " A به طور چند گانه B را تعیین می کند." یا " A با دو فلش به B مراجعه میکند. این حالت برقرار است اگر و فقط اگر در هر مقدار ممکن معتبری برای R ، مجموعه ای از مقادیر B که با جفت (مقدار C و مقدار A) مطابقت دارد، فقط به مقدار A بستگی داشته باشد و مستقل از مقدار C باشد.

برخی از جداول، در حالت نرمال BCNF هستند ولی هنوز ناهنجاری هایی دارند. جدول employee را در نظر بگیرید. این جدول شامل شماره کارمندی و مهارت ها و زبان هایی است که هر کارمند بر آن مسلط است:

Employee

EMP#	Skills	LANGUAGES
100	برنامه نویسی جاوا تجزیه و تحلیل شی گرا	انگلیسی
101	برنامه نویسی دلفی تجزیه و تحلیل شی گرا طراحی وب سایت	انگلیسی آلمانی

در این جدول ، ستون های Skills و LANGUAGES دارای گروه های اطلاع تکرار شونده هستند. نمودار وابستگی این جدول به شکل زیر است :

Employee (EMP#, Skills , LANGUAGES)

فرض کنید این جدول را بصورت زیر ، تبدیل به نرمال کنیم :

Employee ۲

EMP#	SKILLS	LANGUAGES
100	برنامه نویسی جاوا	انگلیسی
100	تجزیه و تحلیل شی گرا	انگلیسی
101	برنامه نویسی دلفی	انگلیسی
101	تجزیه و تحلیل شی گرا	انگلیسی
101	طراحی وب سایت	انگلیسی
101	برنامه نویسی دلفی	آلمانی
101	تجزیه و تحلیل شی گرا	آلمانی
101	طراحی وب سایت	آلمانی

نمودار وابستگی جدول Employee 2، به شکل زیر خواهد بود :

Employee 2 (EMP#, Skills , LANGUAGES)

جدول Employee 2، نرمال BCNF است (چون یک جدول تمام کلید است و این یک مسئله بدیهی است)
ولی در درج، حذف و اصلاح، ناهنجاری دارد . مثلاً اگر کارمند ۱۰۱ ، مهارت « برنامه نویسی جاوا » را نیز کسب کند ، به جای اضافه کردن یک تاپل ، باید دو تاپل جدید به جدول اضافه شود :

انگلیسی	برنامه نویسی جاوا	۱۰۱
آلمانی	برنامه نویسی جاوا	۱۰۱

و یا اگر بخواهیم زبان آلمانی را از اطلاعات کارمند ۱۰۱ حذف کنیم، به جای حذف یک تاپل ، باید سه تاپل زیر را حذف کنیم :

آلمانی	برنامه نویسی دلفی	۱۰۱
آلمانی	تجزیه و تحلیل شی گرا	۱۰۱
آلمانی	طراحی وب سایت	۱۰۱

برای رفع این معایب، جدول را به نرمال ۴ تبدیل مکنیم.

یک جدول نرمال ۴ است اگر :

۱ – نرمال BCNF باشد.

۲ – هیچ وابستگی چند مقداری در آن وجود نداشته باشد.

تعریف وابستگی چند مقداری : اگر در جدول $T(A, B, C)$ ، به ازاء هر مقدار برای ویژگی A ، مجموعه ای از مقادیر برای ویژگی C وجود داشته باشد و این مجموعه، مستقل از مقادیر ویژگی B باشد، ویژگی C به ویژگی A وابستگی چند مقداری دارد $(A \twoheadrightarrow B)$.



مثالی از فرم نرمال درجه چهارم 4NF

مثال: جدول (EMP#, Skills, LANGUAGES) employee 2 را در نظر بگیرید:

$(Emp\#, Skills) = (101, \text{برنامه نویسی دلفی}) \rightarrow Languages = \{\text{انگلیسی و آلمانی}\}$
 $(Emp\#, Skills) = (101, \text{تجزیه و تحلیل شی گرا}) \Rightarrow Languages = \{\text{انگلیسی و آلمانی}\}$
 $(Emp\#, Skills) = (101, \text{طراحی وب سایت}) \Rightarrow Languages = \{\text{انگلیسی و آلمانی}\}$

مشاهده می کنید که تنها Emp#، مجموعه Languages را تعیین می کند و تغییر Skills هیچ نقشی در تغییر مجموعه Languages ندارد. دلیل این امر نیز آن است که مهارت های یک شخص و زبان هایی که شخص بر آن تسلط دارد هیچ ارتباطی با یکدیگر ندارند. پس Languages به Emp# وابستگی چند مقداری دارد (Emp# $\rightarrow \rightarrow$ Languages). از طرف دیگر:

$(Emp\#, Languages) = (101, \text{انگلیسی}) \Rightarrow Skills = \{\text{برنامه نویسی دلفی، تجزیه و تحلیل شی گرا، طراحی وب سایت}\}$

در واقع تنها Emp# مجموعه Skills را تعیین می کند و Languages هیچ نقشی در تغییر مجموعه Skills ندارد پس Skills نیز به Emp# وابستگی چند مقداری دارد (Emp# $\rightarrow \rightarrow$ skills). برای جلوگیری از ایجاد وابستگی چند مقداری، در همان مرحله اول و هنگام تبدیل جدول آنرمال به نرمال 1، به ازای هر گروه اطلاع تکرار شونده، باید جدول مجزایی در نظر بگیریم:



مثالی از فرم نرمال درجه چهارم 4NF

59

Emp_Skills

Emp#	Skills
100	برنامه نویسی جاوا
100	تجزیه و تحلیل شی گرا
101	برنامه نویسی دلفی
101	تجزیه و تحلیل شی گرا
101	طراحی وب سایت

Emp_Languages

Emp#	Languages
100	انگلیسی
101	انگلیسی
101	آلمانی

هر دو جدول (emp_languages(Emp#, Languages) و emp_skills (Emp# , Skills) نرمال ۴ هستند.

s#	p#	j#	Qty
S1	P1	J1	12000
S1	P2	J1	20000
S1	P3	J1	3000
S1	P4	J1	8000
S1	P1	J2	10000
S1	P1	J3	9000
S2	P1	J1	2000
S2	P1	J3	5000
S2	P3	J1	8000
S3	P1	J1	9000
S3	P2	J3	9000
S3	P1	J3	4000

مثال: جدول معروف SPJ را در نظر بگیرید:



مثالی از فرم نرمال درجه چهارم 4NF

60

هر تولید کننده ممکن است برای چندین پروژه، محصولاتی تولید کند. بنابراین، به ازای هر $S\#$ ، مجموعه ای از $J\#$ ها و مجموعه ای از $P\#$ ها وجود دارند اما:

۱) ویژگی $J\#$ به $S\#$ ، وابستگی چند مقداری ندارد چون:

$$(s\#, p\#) = (S2, P1) \Rightarrow J\# = \{ J1, J3 \}$$

$$(s\#, p\#) = (S2, P3) \Rightarrow J\# = \{ J1 \}$$

پس تنها $S\#$ ، مجموعه $J\#$ را تعیین نمی کند بلکه با تغییر مقدار $p\#$ نیز احتمال تغییر مجموعه مقادیر $J\#$ وجود دارد.

۲) ویژگی $p\#$ به $s\#$ وابستگی چند مقداری ندارد چون:

$$(s\#, j\#) = (S1, j1) \Rightarrow J\# = \{ p1, p2, J3 \}$$

$$(s\#, j\#) = (S1, j1) \Rightarrow J\# = \{ p1 \}$$

پس تنها $S\#$ ، مجموعه $p\#$ را تعیین نمی کند بلکه با تغییر مقدار $j\#$ نیز احتمال تغییر مجموعه مقادیر $p\#$ وجود دارد.

می توان نتیجه گرفت در جدول SPJ، وابستگی چند مقداری وجود ندارد و این جدول، نرمال ۴ است.



فرم نرمال درجه پنجم 5NF

61

یک متغیر رابطه ای R از درجه ۵ نرمال است، اگر فقط اگر هر وابستگی الحاق غیر جزئی که برای R برقرار است، توسط کلیدهای R ارائه شود. به طوری که :

الف: وابستگی الحاق $\{A, B, \dots, Z\}^*$ بر روی R جزئی است اگر فقط اگر حداقل یکی از A, B, \dots, Z مجموعه ای از صفات R باشند.

ب: وابستگی الحاق $\{A, B, \dots, Z\}^*$ بر روی R توسط کلید کاندید R ارائه شود اگر فقط اگر هر یک از A, B, \dots, Z فوق کلید R باشد.

نکته: $R\{A, B, C\}$ وابستگی الحاق $\{AB, AC\}^*$ را برآورده می کند اگر و فقط اگر وابستگی چند مقداری $A \twoheadrightarrow B \mid C$ را برآورده سازد.

چون این تئوری می تواند به عنوان تعریفی از وابستگی چند مقداری بیان شود، نتیجه می شود که وابستگی چند مقداری فقط حالت خاصی از وابستگی الحاق است، یا وابستگی های الحاق حالت کلی از وابستگی های

چند مقداری است. به طور رسمی، داریم :

$$A \twoheadrightarrow B \mid C = \{AB, AC\}^*$$



فرم نرمال درجه پنجم 5NF

62

یک جدول در حالت نرمال ۵ است اگر :

۱ - نرمال ۴ باشد.

۲ - نتوان آنرا به جداول های کوچک تر تقسیم کرد بطوری که حداقل یکی از جداول، شامل هیچ یک از کلید های کاندیدای جدول اولیه نباشد.

مثال : طبق قوانین یک دانشگاه :

St#	Crs#	Prof#
S1	C1	P1
S1	C2	P2
S2	C2	P1

- هر دانشجو می تواند هر درس را چند بار ولی تنها یک بار با هر استاد بگیرد.

- هر استاد می تواند درس های مختلفی را تدریس کند و هر درس توسط اساتید مختلف تدریس می شود.

جدول enroll را در نظر بگیرید :

تنها کلید کاندیدای این جدول ، $Prop\# + Crs\# + St\#$ و نمودار وابستگی این جدول به شکل زیر است :

Enroll ($St\# + Crs\# + Prop\#$)

این جدول، نرمال BCNF است و چون وابستگی چند مقداری ندارد، نرمال ۴ نیز هست. برای تشخیص نرمال ۵ بودن این جدول، باید بررسی کنیم آیا این جدول قابل تجزیه به چند جدول کوچک تر که پیوند آنها، جدول اولیه را نتیجه دهد، هست یا خیر ؟

enroll1

St#	Crs#
S1	C1
S1	C2
S2	C2

enroll2

St#	Prof#
S1	p1
S1	p2
S2	P1

این جدول را به دو جدول enroll1 , enroll2 می شکنیم :



فرم نرمال درجه پنجم 5NF

63

Enroll1 join enroll2

St#	Crs#	Prof#
S1	C1	P1
S1	C1	P2
S1	C2	P1
S1	C2	P2
S2	C2	P1

نتیجه پیوند این دو جدول، بصورت زیر خواهد بود :

همانطور که مشاهده می کنید، پیوند دو جدول، دو تاپل اضافی تولید کرد که در جدول enroll اولیه وجود نداشت. پس این تجزیه بی فایده است. به همین ترتیب، اگر این جدول را به دو جدول enroll3 و enroll4 بشکنیم، داریم :

Enroll3

St#	Crs#
S1	C1
S1	C2
S2	C2

Enroll4

St#	Prof#
S1	P1
S1	P1
S2	P2

enroll3 join enroll4

St#	Crs#	Prof#
S1	C1	P1
S1	C2	P1
S1	C2	P2
S2	C2	P1
S2	C2	P2

نتیجه دو پیوند بصورت زیر خواهد بود :



فرم نرمال درجه پنجم 5NF

64

همانطور که مشاهده می کنید، پیوند دو جدول، دو تاپل اضافی تولید کرد که در جدول enroll اولیه وجود نداشت. پس این تجزیه نیز بی فایده است .

حال روی جدول enroll، سه پرتو (از این عملگر که در جبر رابطه ای تعریف شده و بصورت \bowtie نشان داده می شود ، برای گزینش عمودی در یک جدول استفاده می ود یعنی با استفاده از این عملگر می توان ستون های مورد نظر از یک جدول را انتخاب کرد) می گیریم :

SC

St#	Crs#
S1	C1
S1	C2
S2	C2

CP

Crs#	Prof#
C1	P1
C2	P2
C2	P1

SP

St#	Prof#
S1	P1
S1	P2
S2	P1

ویژگی مشترک بین دو جدول SC و CP، CRS# است. نتیجه پیوند دو جدول ، جدول زیر خواهدبود :

CS join CP

St#	Crs#	Prof#
S1	C1	P1
S1	C2	P2
S1	C2	P1
S2	C2	P2
S2	C2	P1



فرم نرمال درجه پنجم 5NF

65

ویژگی مشترک بین دو جدول (SC join CP) و SP، St# + Prof# است. نتیجه پیوند این دو جدول، جدول مقابل خواهد بود :

(Sc join Cp) join Sp

St#	Crs#	Prof#
S1	C1	P1
S1	C2	P2
S1	C2	P1
S2	C2	P1

همانگونه که مشاهده می کنید، نتیجه (Sc join Cp) join Sp، با جدول enroll برابر نیست و شامل تاپل

اضافی (S1 , C2 , P1) است که در جدول enroll وجود ندارد. این مشکل از آنجا ناشی می شود که :

۱ - دانشجوی S1، درس C2 را گرفته است.

۲ - دانشجوی S1 با استاد P1 درس گرفته است.

۳ - درس C2 جز درس هایی است که استاد P1 تدریس می کند.

چنین نتیجه شده است که دانشجوی S1، درس C2 را با استاد P1 گرفته است در حالی که واقعیت ندارد. جدول enroll،

سه ستون دارد پس نمی توان آنرا بصورت ستونی به بیش از ۳ جدول تجزیه کرد و چون پیوند هیچ یک از تجزیه های

دوتایی و سه تایی، ما را به جدول اصلی نرساندند، می توان نتیجه گرفت جدول enroll قابل تجزیه به جداول کوچکتر که

پیوند آنها جدول اولیه را نتیجه می دهد، نیست پس جدول enroll نرمال ۵ است.

مثال : مشخصات کلی بانک اطلاعاتی ثبت نام یک دانشگاه در زیر آمده است. این بانک را طراحی و نرمال کنید:

۱- کد درسها غیر تکراری و نام و تعداد واحد دروس مشخص است.

۲- هر دانشکده درس های مشخصی را ارائه میکند و مسئول دانشکده و شهر آن مشخص است. دروس همنام در دانشکده های مختلف واحد متفاوتی دارند.

۳- هر درس در چند گروه در زمان و مکان مشخص توسط یک استاد ارائه می شود و شماره گروه در درسها و نیمسال های مختلف تکرار میگردد.

۴- دانشجویان در گروه های مختلف ثبت نام میکنند و نمره می گیرند.

جواب : ابتدا هر جداول مشخص و وابستگی تابعی آن تعیین می گردد

$Crs = \{ c\# , cname , unit , clg\# \}$

$f1 = \{ c\# \rightarrow cname , unit , clg\# , (cname , unit) \rightarrow clg\# \}$

$Clg = (clg\# , Clg\ name , city , Pname)$

$f2 = \{ clg\# \rightarrow Clg\ name , city , Pname \}$

$Sec = (s\# , sec\# , s\# , pname , term , score , time , place)$

$f3 = \{ (s\# , c\# , term) \rightarrow sec\# , score , (sec\# , c\# , term) \rightarrow pname , time , place \}$

تمرین : فرض کنید سیستم پایگاه داده ها در یک دانشگاه شامل جداول زیر باشد :

Field (field# , field Name)

در این جدول اطلاعات مربوط به رشته های تحصیلی ذخیره می شود :

field# : کد رشته تحصیلی یک کد منحصر بفرد در نظر گرفته شده است)

Field Name : نام رشته تحصیلی است

Type (type# , type Name , fee)

در این جدول اطلاعات مربوط به نوع درس ذخیره می شود :

type# : کد نوع درس (به هر نوع درس یک کد منحصر بفرد داده می شود) .

type Name : نوع درس

Fee : قیمت هر واحد

Student (st# , sname , start year , field#)

در این جدول اطلاعات مربوط به دانشجویان نگهداری می شود :

st# : شماره دانشجویی و Sname : نام دانشجو

start year : سال ورود به دانشگاه

field# : کد رشته تحصیلی دانشجو

Course (crs# , cname , unit , type#)

در این جدول اطلاعات دروس ذخیره می شود :

crs# : شماره درس و cname : نام درس

unit : تعداد واحد درس و type# : کد نوع درس (نظری ، عملی ، و ...)

CF (crs# , field# , Kind)

در این جدول مشخص می شود هر درس مربوط به کدام رشته های تحصیلی است:

crs# : شماره درس

field# : کد رشته تحصیلی

Kind : این ویژگی مشخص می کند درس مورد نظر برای رشته مورد نظر چه حالتی دارد (' P ' برای پیش نیاز ، ' A ' برای پایه ، ' T '

برای تخصصی ، ' O ' برای عمومی و ' E ' برای اختیاری)

Grades (st# , crs# , term , grade)

در این جدول اطلاعات مربوط به نمرات دانشجویان ذخیره می شود :

st# : شماره دانشجویی

crs# : شماره درس

Term : نیمسال اخذ درس توسط دانشجو (مثلاً ۸۳۱ به معنی ترم اول سال ۸۳ است)

Grade : نمره اخذ شده

Per (crs# , per#) در این درس پیش نیازهای هر درس مشخص می شوند :

crs# : شماره درس

per# : شماره درس پیش نیاز

Prof (Prof# , pname , degree)

در این جدول اطلاعات مربوط به اساتید ذخیره می شود :

Prof# : شماره استاد (به هر استاد یک شماره منحصر بفرد داده شده است)

Pname : نام استاد

Degree (برای لیسانس) ' L ' برای فوق لیسانس ، ' F ' برای دکترا ، ' D ' : آخرین مدرک تحصیلی استاد

PC (Prof# , crs# , term)

در این جدول مشخص می شود هر استاد در هر نیمسال چه دروسی را تدریس کرده است :

Prof# شماره استاد :

crs# شماره درس :

Term نیمسال تحصیلی :

Tuition (field# , start year , const Tuition)

در این جدول بر اساس سال ورود به دانشگاه، شهریه ثابت هر رشته تحصیلی مشخص می شود :

field# کد رشته تحصیلی :

start year سال ورود به دانشگاه :

const Tuition شهریه ثابت :

برای درک بهتر مساله به نمونه هایی از اطلاعات ذخیره شده در پایگاه داده توجه کنید :

Field

field#	field Name
1	مهندسی کامپیوتر
2	مهندسی الکترونیک
3	ریاضی محض

Type

type#	type Name	fee
1	نظری	5000
2	عملی	20000
3	آزمایشگاه	30000

Tuition

field#	start year	const Tuition
1	80	75000
1	81	80000
1	82	90000
2	81	80000
2	82	90000
3	82	75000



مثال هایی از نرمال سازی



Course

crs#	cname	unit	type#
1100	ادبیات	2	1
1105	تربیت بدنی	1	2
1400	برنامه سازی ۱	3	1
1402	برنامه سازی ۲	3	1
1403	ذخیره و بازیابی	3	1
1407	پایگاه داده	3	1
1500	ریاضی ۱	3	1
1600	زبان پیش نیاز	2	1

Student

st#	sname	start year	field#
8001	آرش رادمهر	80	1
8002	عسل شاملو	80	3
8003	سیاوش آزاد	80	1
8101	ساغر راد	81	1
8102	علی نیکی	81	1
8111	فرامرز نیکی	81	1
8112	علی رضایی	81	2

Prof

Prof#	pname	degree
101	فرانک شایسته	L
102	علی پیامی	F
106	آزاده نیکوکار	F
107	سیامک فرزانه	D
107	علی نادر نژاد	F

Grades

st#	crs#	term	grade
8001	1400	811	9
8001	1400	812	13
8001	1401	811	13
8101	1400	821	19
8101	1500	821	14
8111	1400	811	12
8111	1401	811	20



مثال هایی از نرمال سازی



PC

Prof#	crs#	term
101	1400	801
101	1400	802
101	1401	801
102	1400	801
108	1500	811

Per

crs#	per#
1400	1500
1402	1400
1407	1402
1407	1403

CF

crs#	field#	Kind
1100	1	O
1100	2	O
1100	3	O
1105	1	O
1105	2	O
1105	3	O
1400	1	T
1400	2	E
1400	3	E
1402	1	T
1403	1	T
1407	1	T
1500	1	P
1500	2	P
1500	3	P

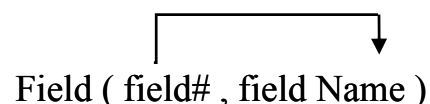
الف) صحت روابط زیر را در جدول field بررسی کرده، نمودار وابستگی این جدول را رسم کنید :

field# \longrightarrow field Name

این رابطه صحیح است چون با داشتن یک کد رشته، تنها به یک نام رشته می‌رسیم .

field# $\xrightarrow{\text{FFD}}$ field Name

این رابطه صحیح است چون با داشتن یک کد رشته، تنها به یک نام رشته می‌رسیم و field# قابل خلاصه شدن نیست .



نمودار وابستگی :

تذکر: در نمودار های وابستگی، برای مشخص کردن کلید اصلی از خط زیر استفاده می‌شود .

ب) صحت روابط زیر را در جدول type بررسی کرد، نمودار وابستگی این جدول را رسم کنید :

type# $\xrightarrow{\text{FFD}}$ type Name , fee

این رابطه صحیح است چون با داشتن یک نوع کد درس، تنها به یک نام نوع درس و تنها به یک قیمت واحد می‌رسیم.

type# + type Name $\xrightarrow{\text{FFD}}$ fee

این رابطه صحیح نیست چون type \longrightarrow fee صحت دارد. پس طرف چپ قابل خلاصه شدن است .

type# $\xrightarrow{\text{FFD}}$ type Name , fee

این رابطه صحیح است چون type# \longrightarrow type Name , fee و از طرف دیگر، type# قابل خلاصه شدن نیست.

نمودار وابستگی :

Type (type# , type Name , fee)



ج) صحت روابط زیر را در جدول Tuition بررسی کرده ، نمودار وابستگی این جدول را رسم کنید.

Field \longrightarrow const Tuition

این رابطه صحیح نیست چون با داشتن یک کد رشته، لزوماً به یک شهریه ثابت واحد نمی رسیم، چون نرخ شهریه برای ورودی های هر سال در رشته های مختلف، متفاوت است. مثلاً برای کد رشته کامپیوتر ممکن است چندین نرخ شهریه وجود داشته باشد.

start year \longrightarrow const Tuition

این رابطه صحیح نیست چون با داشتن یک سال ورود، لزوماً به یک شهریه ثابت واحد نمی رسیم، چون نرخ شهریه برای ورودی های هر سال در رشته های مختلف، متفاوت است. مثلاً برای ورودی های سال ۹۳ ممکن است چندین نرخ شهریه وجود داشته باشد .

field#+ start yea \longrightarrow const Tuition

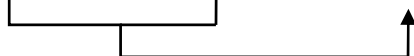
این رابطه صحیح است چون با داشتن یک کد رشته و سال ورود، لزوماً به یک شهریه ثابت می رسیم .

field#+ start year $\xrightarrow{\text{FFD}}$ const Tuition

این رابطه صحیح است چون field#+ start yea \longrightarrow const Tuition صحیح است و از طرف دیگر، field#+ start yea قابل خلاصه شدن نیست.

Tuition (field#, start year, const Tuition)

نمودار وابستگی :



د) صحت روابط زیر را در جدول st بررسی کرده و

نمودار وابستگی این جدول را رسم کنید :

sname + start year \longrightarrow field#

این رابطه صحیح است چون با داشتن نام یک دانشجو و سال ورود، لزوماً به کد رشته واحد نمی رسیم. مثلاً ممکن است دو دانشجوی ورودی ۸۰ با نام مسعود اسماعیلی در دانشگاه وجود داشته باشند که یکی در رشته عمران و دیگری در رشته کامپیوتر تحصیل می کند.



مثال هایی از نرمال سازی

74

st# → sname , start year , field#

این رابطه صحیح است چون با داشتن یک شماره دانشجویی، تنها ؛ به یک نام ، یک سال ورود و به یک کد رشته می رسیم .

st# $\xrightarrow{\text{FFD}}$ sname , start year , field#

این رابطه صحیح است چون sname , start year , field# → st# صحت دارد و از طرف دیگر، st# قابل خلاصه شدن نیست.

Student (st# , sname , start year , field#)

(صحت روابط زیر را در جدول Course بررسی کرده، نمودار وابستگی این جدول را رسم کنید:

crs# → cname , unit , type#

این رابطه صحیح است چون با داشتن یک شماره درس، تنها به یک تعداد واحد و تنها به یک کد نوع درس می رسیم.

crs# + cname → unit , type#

این رابطه صحیح است چون با داشتن یک شماره درس و نام درس، تنها به یک تعداد واحد و تنها به یک کد نوع درس می رسیم.

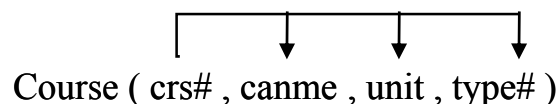
crs# $\xrightarrow{\text{FFD}}$ cname , unit , type#

این رابطه صحیح است چون crs# → cname , unit , type# برقرار است و از طرف دیگر، crs# قابل خلاصه شدن نیست.

crs# + cname $\xrightarrow{\text{FFD}}$ unit , type#

این رابطه صحیح نیست چون crs# → unit , type# صحیح است. پس سمت چپ قابل خلاصه شدن است.

نمودار وابستگی :



(ز) صحت روابط زیر را در جدول CF بررسی کرده، نمودار وابستگی این جدول را رسم کنید:

این رابطه صحیح نیست چون با داشتن یک شماره درس، لزوماً به یک کد رشته نمی رسیم. مثلاً ممکن است درس ریاضی ۱ ، هم مربوط به رشته ریاضی و هم مربوط به کامپیوتر باشد.

crs# —————> Kind

این رابطه صحیح نیست چون با داشتن یک شماره درس، لزوماً تنهابه یک حالت نمی رسیم. مثلاً ممکن است درس برنامه سازی ۲ ، برای رشته ریاضی، اختیاری و برای رشته کامپیوتر، تخصصی باشد.

field# —————> Kind

این رابطه صحیح نیست چون با داشتن یک کد رشته، لزوماً تنها به یک حالت درس نمی رسیم چون مسلماً درس های زیادی در یک رشته وجود دارند که هر کدام از آنها، حالت خاص خود را دارد. مثلاً در رشته کامپیوتر هم دروس پایه و هم دروس تخصصی و هم دروس اختیاری وجود دارند.

field# FFD —————> Kind

این رابطه صحیح نیست چون رابطه field# —————> Kind صحیح نیست.

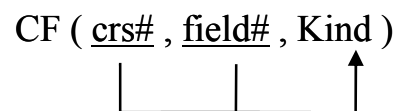
crs#+ field# —————> Kind

این رابطه صحیح است چون هر درس برای هر رشته، تنها یک حالت می تواند داشته باشد مثلاً درس برنامه سازی ۲ ، برای رشته کامپیوتر تنها تخصصی است .

crs#+ field# FFD —————> Kind

این رابطه صحیح است چون اولاً crs#+ field# —————> Kind برقرار است و ثانیاً crs#+ field# قابل خلاصه شدن نیست.

نمودار وابستگی :



س)صحت روابط زیر را در جدول Grades بررسی کرده، نمودار

وابستگی این جدول را رسم کنید :

st# \longrightarrow grade

این رابطه صحیح نیست چون یک دانشجو ممکن است نمرات زیادی داشته باشد، پس داشتن یک شماره دانشجویی لزوماً ما را به یک نمره واحد نمی رساند.

st# + crs# \longrightarrow grade

این رابطه صحیح نیست چون ممکن است یک دانشجو در یک درس چندین نمره گرفته باشد. مثلاً ممکن است دانشجوی ۷۸۰۱ بار اول درس ۱۴۰۰ نمره ۸ و بار دوم نمره ۱۱ گرفته باشد. پس با داشتن یک شماره دانشجویی و یک شماره درس، لزوماً به یک نمره واحد نمی رسیم.

st# + crs# \longrightarrow term

این رابطه درست نیست چون ممکن است یک دانشجو یک درس را در چند ترم گرفته باشد، ممکن است دانشجوی ۷۸۰۱ درس ۱۴۰۰ را هم در ترم ۸۲۱ و هم در ترم ۸۲۲ گرفته باشد. پس با داشتن یک شماره دانشجویی و یک شماره درس، لزوماً به یک ترم واحد نمی رسیم .

st# + crs# + term \longrightarrow grade

این رابطه صحیح است چون با داشتن یک شماره دانشجویی، یک شماره درس و یک ترم، تنها به یک نمره می رسیم، چون در هر درس در یک ترم برای هر دانشجو تنها یک نمره ثبت می شود.



مثال هایی از نرمال سازی

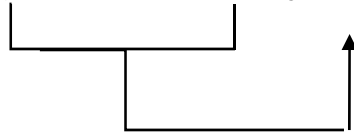
77

$st\# + Crs\# + term \xrightarrow{FFD} grade$

این رابطه صحیح است چون اولاً $st\# + crs\# + term \longrightarrow grade$ برقرار است و ثانیاً طرف چپ رابطه قابل خلاصه شدن نیست.

Grades (st# , Crs# , term , grade)

نمودار وابستگی :



ش) صحت روابط زیر را در جدول Per بررسی کرده، نمودار وابستگی این جدول را رسم کنید :

$crs\# \longrightarrow PerCrs\#$

این رابطه صحیح نیست چون با داشتن یک شماره درس، لزوماً به یک شماره درس پیش نیاز نمی رسیم. مثلاً ممکن است درس پایگاه داده بیش از یک پیش نیاز داشته باشد.

$Per\ crs\# \longrightarrow crs\#$

این رابطه صحیح نیست چون با داشتن یک شماره درس پیش نیاز، لزوماً به یک شماره درس واحد نمی رسیم. مثلاً ممکن است درس برنامه سازی ۱، پیش نیاز چند درس مختلف باشد.

$crs\# + Per\ crs\# \longrightarrow crs\# , Per\ crs\#$

صحت این رابطه بدیهی است .



مثال هایی از نرمال سازی

78

$\text{crs\#} + \text{Per crs\#} \xrightarrow{\text{FED}} \text{crs\#} , \text{Per crs\#}$

این رابطه صحیح است چون اولاً $\text{crs\#} + \text{Per crs\#} \rightarrow \text{rs\#} , \text{Per crs\#}$ برقرار است و ثانیاً طرف چپ رابطه قابل خلاصه شدن نیست.

$\text{Per} (\underline{\text{crs\#}} , \underline{\text{Per crs\#}})$

نمودار وابستگی :

و (صحت روابط زیر را در جدول Prof بررسی کرده، نمودار وابستگی این جدول را رسم کنید :

$\text{Prof\#} \longrightarrow \text{Pname} , \text{degree}$

این رابطه صحیح است چون اولاً $\text{Prof\#} \rightarrow \text{Pname} , \text{degree}$ برقرار است و ثانیاً طرف چپ رابطه قابل خلاصه شدن نیست.

$\text{Prof} (\underline{\text{Prof\#}} , \text{Pname} , \text{degree})$

نمودار وابستگی :





مثال هایی از نرمال سازی

79

$\text{crs\#} + \text{Per crs\#} \xrightarrow{\text{FFD}} \text{crs\#} , \text{Per crs\#}$

این رابطه صحیح است چون اولاً $\text{crs\#} , \text{Per crs\#} \longrightarrow \text{crs\#} + \text{Per crs\#}$ برقرار است و ثانیاً طرف چپ رابطه قابل خلاصه شدن نیست.

$\text{Per} (\underline{\text{crs\#}} , \underline{\text{Per crs\#}})$

نمودار وابستگی :

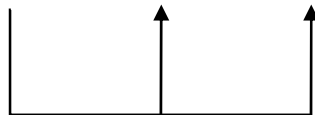
و (صحت روابط زیر را در جدول Prof بررسی کرده، نمودار وابستگی این جدول را رسم کنید :

$\text{Prof\#} \longrightarrow \text{Pname} , \text{degree}$

این رابطه صحیح است چون اولاً $\text{Prof\#} \longrightarrow \text{Pname} , \text{degree}$ برقرار است و ثانیاً طرف چپ رابطه قابل خلاصه شدن نیست.

$\text{Prof} (\underline{\text{Prof\#}} , \text{Pname} , \text{degree})$

نمودار وابستگی :





مثال هایی از نرمال سازی



ی) کلید های خارجی کلیه جداول را مشخص کنید:

نام جدول	کلید خارجی
Field	ندارد
Type	ندارد
Student	field# نسبت به جدول field
Course	type# نسبت به جدول type
CF	crs# نسبت به جدول Course و field# نسبت به field
Grades	st# نسبت به جدول Student و crs# نسبت به جدول Course
Per	crs# نسبت به جدول Course و per# نسبت به جدول Course (تشابه اسمی کلید اصلی و خارجی اهمیتی ندارد)
Prof	ندارد
PC	Prof# نسبت به جدول Prof و crs# نسبت به جدول
Tuition	field# نسبت به field

سیستم کتابخانه یک دانشگاه را در نظر بگیرید.

در این کتابخانه به هر کتاب یک شماره منحصر به فرد داده شده است مثلاً اگر در کتابخانه ، سه نسخه از یک کتاب وجود داشته باشد، هر یک از آنها به شماره مجزا خواهد داشت. در هر امانت دانشجو می تواند حداکثر سه کتاب را به امانت بگیرد (مثلاً در امانت شماره ۱۰۰۱ ممکن است دو کتاب ۱۴۰۳ و ۱۸۰۹ به امانت گرفته شوند) و باید همه کتاب هایی را که باهم از کتابخانه به امانت گرفته است ، با هم به کتابخانه برگرداند. با توجه به این قوانین ، نمودار وابستگی جدول loan را رسم کرده ، آن را به جداول نرمال ۳ تبدیل کنید.

Loan (loan# , book# , bname , author , st# , sname , field , loanDate , returnDate)

Loan#: شماره امانت

Book#: شماره کتاب

Bname: نام کتاب

Author: نام نویسنده

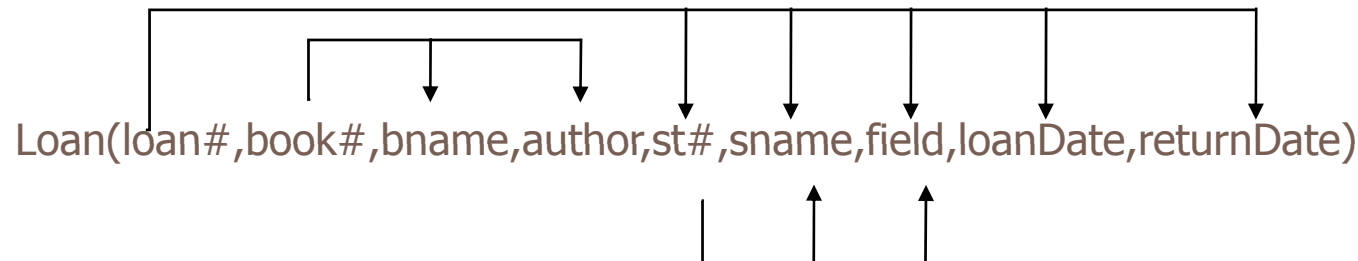
St#: شماره دانشجویی که کتاب را به امانت گرفته

Sname: نام دانشجو

Field: رشته تحصیلی دانشجو(هر دانشجو تنها در رشته تحصیل می کند)

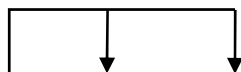
loanDate: تاریخ امانت گرفتن کتاب

ReturnDate: تاریخ برگشت کتاب به کتابخانه توسط دانشجو



تبدیل به نرمال ۲:

این جدول، نرمال ۲ نیست چون bname و author به book# که جزئی از کلید اصلی است، وابستگی تابعی دارند. همچنین، st# و sname و field و loanDate و returnDate به loan# که جزئی از کلید اصلی است وابستگی تابعی دارند. پس این جدول را به جداول نرمال ۲ می شکنیم:



L (loan# , s# , sname , field , loanDate , returnDate)

Book (book# , bname , author)

LB (loan# , book#)

نکته: توجه کنید اگر چه هیچ ویژگی غیر کلیدی که به book# و loan# وابستگی داشته باشد و در سطرها قبل نیامده باشد، وجود ندارد، ولی این ردیف را به جدول LB تبدیل کرده ایم. دلیل این امر آن است که تنها در این جدول مشخص می شود در هر امانت ، چه کتاب هایی امانت گرفته شده اند. پس این جدول اطلاعات مفیدی در اختیار می گذارد که در جداول دیگر موجود نیست.

تبدیل به نرمال ۳:

جداول B و LB نرمال ۳ هستند چون در آنها هیچ ویژگی غیر کلیدی به ویژگی غیر کلیدی دیگر وابستگی تابعی ندارد ولی جدول L نرمال ۳ نیست چون در آن sname و field به s# که غیر کلیدی است وابستگی دارد. پس جدول L را به جداول زیر که هر یک نرمال ۳ هستند می شکنیم:

L1 (s# , sname , field)

L2 (loan# , loanDate , returnDate)

پس جدول loan به چهار جدول LB و Book و L1 و L2 که همگی نرمال ۳ هستند، شکسته شد.

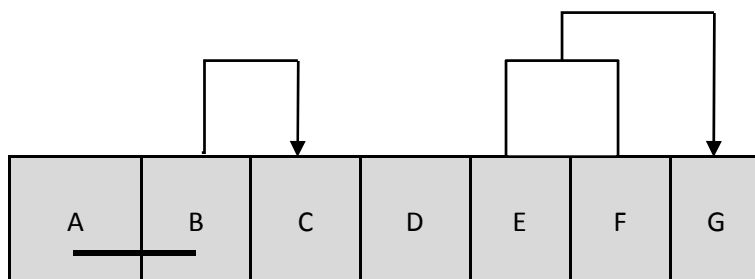


مثال هایی از نرمال سازی

83

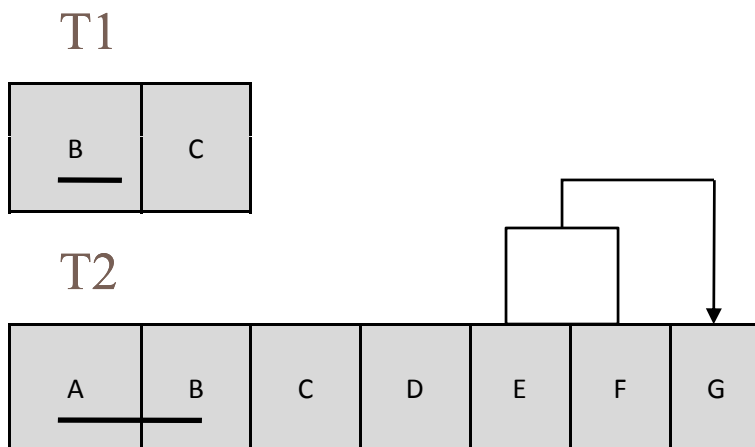
تمرین :

جدول زیر را با ذکر کلیه مراحل، به جداول نرمال ۳ تبدیل کنید.



تبدیل به نرمال ۲: از آنجا که ویژگی C به ویژگی B که بخشی از کلید اصلی است، وابستگی تابعی دارد، در این جدول ، وابستگی جزئی وجود دارد. پس نرمال ۲ نیست.

برای تبدیل به نرمال ۲:





مثال هایی از نرمال سازی

84

تبدیل به نرمال ۳: جدول T1 ، نرمال ۳ است ولی در جدول T2 ، ویژگی G که یک ویژگی غیر کلیدی است، به ویژگی های E و F که غیر کلیدی می باشند وابستگی تابعی دارد، پس جدول T2 نرمال ۳ نیست. برای تبدیل به نرمال ۳ ، این جدول را به دو جدول T21 و T22 می شکنیم:

T21

E	F	G
—	—	

T22

A	B	D	E	F
—	—			