موسسه بابان

انتشارات بابان و انتشارات راهیان ارشد

درس و کنکور ارشد

پایگاه داده‌ها

(حل سوالات کنکور ارشد ۱۴۰۱)

ویژه‌ی داوطلبان کنکور کارشناسی ارشد مهندسی کامپیوتر و IT

براساس کتب مرجع

راما کریشنان، آبراهام سیلبرشاتز و رامز المصری

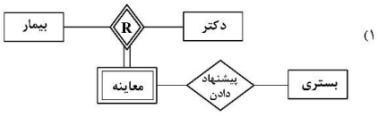
استاد خلیلی‌فر

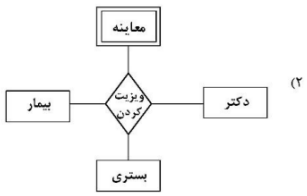
کلیه‌ی حقوق مادی و معنوی این اثر در سازمان اسناد و کتابخانه‌ی ملی ایران به ثبت رسیده است.

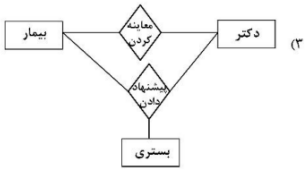
تست‌های فصل سوم: مدل رابطه‌ای

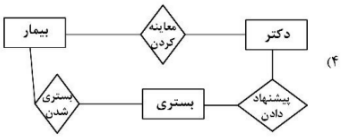
۱۱۵- در مستندات تحلیل یک سامانه پزشکی این چنین ذکر شده است: «دکتر پس از معاینه بیمار، در صورت نیاز، به وی پیشنهاد بستری شدن می‌دهد.» کدام‌یک از گزینه‌های زیر عبارت بالا را مدل می‌کند؟

(مهندسی کامپیوتر-دولتی۱۴۰۱)

۱)


۲)


۳)


۴)


عنوان کتاب: پایگاه داده‌ها

مولف: استاد خلیلی‌فر

ناشر: انتشارات راهیان ارشد و دکتری

آدرس سایت انتشارات بابان: baban.ir

پاسخ تست‌های فصل سوم: مدل رابطه‌ای

۱۱۵- گزینه (۱) صحیح است.

در سامانه پزشکی و بر اساس مفروضات صورت سوال، رابطه بین دکتر، بیمار و معاینه باید درجه 3 و رابطه بین بستری و معاینه باید درجه 2 باشد. بنابراین گزینه اول پاسخ سوال است.

۱۱۶- کدام گزینه درست است؟

(مهندسی کامپیوتر-دولتی۱۴۰۱)

۱) حصول استقلال داده‌ای منطقی و استقلال داده‌ای فیزیکی به یک اندازه مشکل است.
۲) امکان ایجاد استقلال داده‌ای فیزیکی نسبت به استقلال داده‌ای منطقی بیشتر است.
۳) حصول استقلال داده‌ای منطقی از حصول استقلال داده‌ای فیزیکی آسان‌تر است.
۴) در خصوص امکان حصول استقلال داده‌ای منطقی و فیزیکی و میزان سختی حصول آن‌ها صرفا با مشخص بودن مسئله می‌توان اظهارنظر کرد.

عنوان کتاب: پایگاه داده‌ها
مولف: استاد خلیلی‌فر
ناشر: انتشارات راهیان ارشد و دکتری
آدرس سایت گروه بابان: khalilifar.ir

پاسخ تست‌های فصل اول: مفاهیم اولیه

۱۱۶- گزینه (۲) صحیح است.

یکی از مهم‌ترین مزایای تکنولوژی پایگاه داده‌ها (مدل مفهومی پایگاه داده)، بلکه مهمترین هدف آن تأمین و افزایش استقلال داده‌ای است، به معنی وابسته نبودن برنامه‌های کاربردی به داده‌های ذخیره شده.

استقلال داده‌ای بر دو نوع می‌باشد:

۱- استقلال فیزیکی داده‌ها

به معنی مصونیت برنامه‌های کاربردی در قبال تغییراتی که در سطح فیزیکی (رسانه ذخیره‌سازی) پایگاه داده‌ها بروز می‌کند. یعنی اگر تغییری در ذخیره‌سازی داده‌ها انجام گیرد (برای مثال نوع دیسک عوض شود) برنامه‌های کاربردی هیچ تغییری نکند.

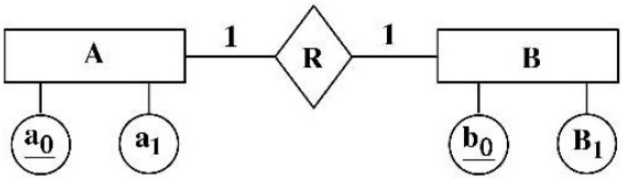
۲- استقلال منطقی داده‌ها

به معنی مصونیت برنامه‌های کاربردی در قبال تعاریف و تغییراتی که در سطح مدل طراحی (مدل رابطه‌ای) پایگاه داده بروز می‌کند. یعنی تعریف و تغییر مدل طراحی بانک (ادراکی خاص یا طراحی منطقی) از دید برنامه‌های کاربردی آنها مخفی بماند.

توجه: از آن جا که با حذف جداول، داده‌ها هم از بین می‌رود، بنابراین برنامه‌های کاربردی نسبت به حذف جداول هیچگاه «**استقلال منطقی**» نخواهند داشت. بنابراین امکان ایجاد استقلال داده‌ای فیزیکی نسبت به استقلال داده‌ای منطقی بیشتر است. پرواضح است که گزینه دوم پاسخ سوال است.

۱۱۷- کدام مورد در تبدیل نمودار موجودیت رابطه مطرح شده به جدول، درست است؟ (لازم به ذکر است تعداد نمونه موجودیت‌های A و B بسیار زیاد و نرخ شرکت کردن آنها در رابطه R بسیار اندک است.)

(مهندسی کامپیوتر-دولتی۱۴۰۱)



۱) جدول ABR

a_0	a_1	b_0	b_1

۲) جدول A

a_0	a_1	b_0

جدول B

b_0	b_1

۳) جدول A

a_0	a_1

جدول B

b_0	b_1

جدول R

a_0	b_0

۴) همه موارد درست هستند.

عنوان کتاب: پایگاه داده‌ها

مولف: استاد خلیلی‌فر

ناشر: انتشارات راهیان ارشد و دکتری

آدرس سایت گروه بابان: khalilifar.ir

۱۱۷- گزینه () صحیح است.

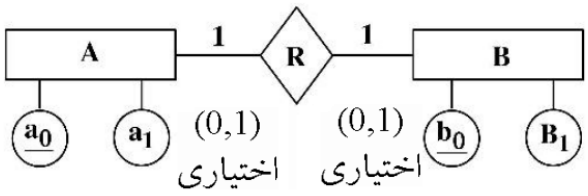
نگاشت رابطه یک به یک بین دو موجودیت به مدل رابطه‌ای

در صورت سوال ذکر شده است که «تعداد نمونه موجودیت‌های A و B بسیار زیاد و نرخ شرکت کردن آنها در رابطه R بسیار اندک است.» عبارت «**نرخ شرکت کردن آنها در رابطه بسیار اندک است**» بدین معنی است که حضور نمونه موجودیت‌های A و B در رابطه **اختیاری** است. بنابراین مطابق قوانین نگاشت رابطه یک به یک بین دو موجودیت به مدل رابطه‌ای در حالت اگر دو طرف رابطه **اختیاری** باشد دستور العمل زیر برقرار است:

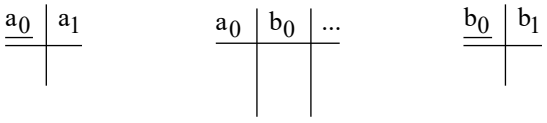
هر موجودیت به یک جدول تبدیل می‌گردد. و یک جدول ارتباط نیز به عنوان ارتباط دهنده دو جدول مورد استفاده قرار می‌گیرد. همچنین کلید کلندید جدول ارتباط یکی از کلیدهای کلندید دو جدول دیگر می‌باشد. دقت کنید که کلید کلندید جدول ارتباط یکی از کلیدهای کاندید دو جدول دیگر است و ترکیبی نخواهد بود.

روال کلی نگاشت در این حالت به صورت زیر است:

مدل تحلیل:



مدل طراحی:



جدول A　　جدول AB　　جدول B

توجه: کلید کاندید جدول AB، ستون a_0 یا ستون b_0 می‌باشد. زیرا هر یک به تنهایی قدرت تفکیک سطرها را دارند. ترکیب ستون‌های a_0 و b_0 باهم به عنوان a_0b_0، ابرکلید ایجاد می‌کند.

توجه: ستون a_0 در جدول AB به عنوان کلید خارجی تعریف می‌گردد که به جدول A ارجاع می‌کند. همچنین ستون b_0 در جدول AB به عنوان کلید خارجی تعریف می‌گردد که به جدول B ارجاع می‌کند.

توجه: سازمان سنجش آموزش کشور، در کلید اولیه خود، گزینه سوم را به عنوان پاسخ اعلام کرده بود. اما در کلید نهایی این سوال حذف گردید، که کار درستی بوده‌است. زیرا مطابق گزینه سوم به صورت زیر:

جدول A

$\underline{a_0}$	a_1

جدول B

$\underline{b_0}$	b_1

جدول R

$\underline{a_0}$	$\underline{b_0}$

ترکیب ستون‌های a_0 و b_0 باهم به عنوان a_0b_0 در جدول R، ابرکلید ایجاد می‌کند، که پاسخ درستی به صورت سوال نیست.

۱۱۸- کدام مورد، خروجی رابطه روبه‌رو است؟

(مهندسی کامپیوتر-دولتی۱۴۰۱)

$$\left(\sigma_{Avg>16}(STUD)\right)\cap\left(\sigma_{Unit=3}(CRS)\right) = ?$$

STUD (دانشجو) (S# (شماره دانشجویی) , Sname (نام و نام‌خانوادگی) , City(شهر) , Avg(معدل))
PROF (استاد) (Pname (نام استاد) , Office (شماره اتاق) , Degree(مدرک))
CRS (درس) (C# (کد درس) , Cname (نام درس) , Unit(تعداد واحد))
SEC (اخذ درس) (Sec# (کد) , C# , S# , Term (کد ترم) , Pname , Score(نمره))

۱) فقط دانشجویانی که معدل آنها در دروس 3 واحدی بالاتر از 16 است را لیست می‌کند.
۲) فقط مشخصات دانشجویانی را که دروس 3 واحدی اخذ کرده‌اند نمایش می‌دهد.
۳) دانشجویانی که معدل بالاتر از 16 هستند و دروس 3 واحدی را نیز اخذ کرده‌اند.
۴) این امکان پذیر نیست، زیرا از یک دامنه یکسان گرفته نشده است.

عنوان کتاب: پایگاه داده‌ها
مولف: استاد خلیلی‌فر
ناشر: انتشارات راهیان ارشد و دکتری
آدرس سایت گروه بابان: khalilifar.ir

پاسخ تست‌های فصل چهارم: جبر رابطه‌ای

۱۱۸- گزینه (۴) صحیح است.

عملگر σ جهت انتخاب سطر در یک جدول مورد استفاده قرار می‌گیرد.

$$\left(\sigma_{Avg>16}(STUD)\right)\cap\left(\sigma_{Unit=3}(CRS)\right)$$

عملگر $\cap$ جهت کشف سطرهای مشترک دو جدول مورد استفاده قرار می‌گیرد. در جبر رابطه‌ای اشتراک هر دو رابطه دلخواه امکان‌پذیر نیست. مگر اینکه شروط سازگاری در مورد آنها برقرار باشد. در جبر رابطه‌ای دو شرط به عنوان شروط سازگاری مطرح است:

شرط اول: تعداد ستون‌های دو جدول یکسان باشد، به عبارت دیگر دو رابطه (جدول) هم درجه باشند.

شرط دوم: نوع یا دامنه ستون‌های متناظر در دو جدول یکسان باشد.

اگر بخواهیم دو شرط فوق را در یک جمله بیان بیان کنیم، اینطور خواهد بود، شروط سازگاری یعنی تیترها در دو رابطه (جدول) یکسان باشد.

توجه: عملگر اشتراک مطرح شده در صورت سوال امکان پذیر نیست، زیرا از یک دامنه یکسان گرفته نشده است و به تبع شرط دوم سازگاری نقض شده‌است. بنابراین پرواضح است که گزینه چهارم پاسخ سوال است.

تست‌های فصل ششم: زبان SQL دستورات DML

۱۱۹- جداول روبه رو را درنظر بگیرید. کدام مورد، توصیف کوئری مطرح شده است؟ (برای راحتی، اسامی انگلیسی ستون‌ها نیز نوشته شده است.)

(مهندسی کامپیوتر-دولتی۱۴۰۱)

جدول درس (Course)

نام درس	کد درس
CName	Code

جدول دانشجو (Student)

نام و نام‌خانوادگی	شماره دانشجویی
Name	Stn

جدول درس اخذ شده (Taken)

کد درس	شماره دانشجو	نمره
CCode	SStn	Mark

```
select Name
from Student S
where not exists (select *
                  from Taken T join Student on Stn = SStn
                  where  Name = 'Mina Asadi' and not exists (select *
                                                             from Taken B
                                                             where B.SStn = S.SSn
                                                             and T.CCode = B.CCode))
```

۱) نام و نام خانوادگی دانشجویانی که همه درس‌هایی را که مینا اسدی اخذ کرده، آنها نیز اخذ کرده‌اند.

۲) نام و نام خانوادگی دانشجویانی که هیچ یک از درس‌هایی را که مینا اسدی اخذ کرده، آنها اخذ نکرده‌اند.

۳) نام و نام خانوادگی دانشجویانی که همه درس‌هایی را که مینا اسدی اخذ نکرده، آنها اخذ کرده‌اند.

۴) نام و نام خانوادگی دانشجویانی که فقط درس‌هایی را اخذ نکرده‌اند که مینا اسدی نیز آنها را اخذ نکرده است.

عنوان کتاب: پایگاه داده‌ها
مولف: استاد خلیلی‌فر
ناشر: انتشارات راهیان ارشد و دکتری
آدرس سایت گروه بابان: khalilifar.ir

پاسخ تست‌های فصل ششم: زبان SQL دستورات DML

۱۱۹- گزینه (۱) صحیح است.

عملگر تقسیم در جبررابطه‌ای به فرم زیر در SQL قابل پیاده‌سازی است.

این پرس و جو به فرم زیر توسط **الگوی اول (همه)** در SQL قابل پیاده‌سازی است.

الگوی اول (00):

```
select Name
from Student S
where not exists (select *
                  from Taken T join Student on Stn = SStn
                  where  Name = 'Mina Asadi' and not exists (select *
                                                             from Taken B
                                                             where B.SStn = S.SSn
                                                             and T.CCode = B.CCode))
```

پرس و جوی فوق را همانند سورها به صورت زیر بخوانید:

«نام و نام خانوادگی دانشجویانی که، وجود نداشته باشد (**NOT EXISTS**) دروس اخذ شده مینا اسدی را که، اخذ نکرده باشند (**NOT EXISTS**). یعنی نام و نام خانوادگی دانشجویانی که همه درس‌هایی را که مینا اسدی اخذ کرده، آنها نیز اخذ کرده‌اند.»

در پرس‌وجوی فوق به ازای حرکت در هر سطر از جدول Student، یک بار به طور کامل از ابتدا تا انتهای جدول حاصل از join دو جدول Student و Taken مطابق شرط اتصال مطرح شده یعنی Taken T join Student on Stn = SStn بررسی می‌گردد، تا مشخص گردد هیچ سطری در جدول Taken T join Student on Stn = SStn از دروس اخذ شده مینا اسدی وجود ندارد که شرط اتصال B.SStn = S.SSn and T.CCode = B.CCode موجود در select داخلی را برقرار نکرده باشد، اگر وجود نداشته باشد، سطر مورد نظر از جدول Student در خروجی نمایش داده می‌شود.

توجه: بعضی آدم‌ها در دنیا هستن که کاری نیست (NOT EXISTS) **که، انجام نداده باشن** (NOT EXISTS)**، یعنی هر کاری بگی انجام دادن. قایقی نیست که سوار نشده باشن، تفریحی نیست که انجام نداده باشن و ... این افراد به عرض زندگی بیشتر توجه داشتن تا طول زندگی!**

توجه: شرط لازم برای تبعیت از الگوی اول وجود **NOT EXISTS** اول و **NOT EXISTS** دوم است و شرط کافی برای تبعیت از الگوی اول وجود **شروط اتصال** متناسب و سازگار است.

بنابراین پرس و جوی فوق، نام و نام خانوادگی دانشجویانی که همه دروس اخذ شده مینا اسدی را اخذ کرده‌اند، نشان می‌دهد. که مطابق پرس و جوی مطرح شده در صورت سوال است. یعنی «نام و نام خانوادگی دانشجویانی که همه درس‌هایی را که مینا اسدی اخذ کرده، آنها نیز اخذ کرده‌اند.»

بنابراین پُرواضح است که گزینه اول پاسخ سوال است.

تست‌های فصل نهم: نرمال‌سازی

۱۲۰- اگر رابطه زیر تا سطح سوم، نرمال‌سازی شود پاسخ کدام است؟

(مهندسی کامپیوتر-دولتی۱۴۰۱)

$R(X,Y,Z,S,T,U,W)$

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

۱) $R_1(\underline{S,W})\ ,\ R_{21}(\underline{X},Z,T,U)\ ,\ R_{22}(\underline{T},Y)$

۲) $R_1(\underline{S,W})\ ,\ R_2(\underline{S},X,Y,Z,T,U)\ ,\ R_{21}(\underline{S},X)\ ,\ R_{221}(\underline{X},Z,T,U)$

۳) $R_1(\underline{S,W})\ ,\ R_2(\underline{S},X,Y,Z,T,U)\ ,\ R_{21}(\underline{S},X)\ ,\ R_{22}(\underline{X},Y,Z,T,U)$

۴) $R_1(\underline{S,W})\ ,\ R_{21}(\underline{S},X)\ ,\ R_{221}(\underline{X},Z,T,U)\ ,\ R_{22}(\underline{T},Y)$

عنوان کتاب: پایگاه داده‌ها
مولف: استاد خلیلی‌فر
ناشر: انتشارات راهیان ارشد و دکتری
آدرس سایت گروه بابان: khalilifar.ir

پاسخ تست‌های فصل نهم: نرمال‌سازی

۱۲۰- گزینه (۴) صحیح است.

با توجه به وابستگی‌های مطرح شده برای رابطه R(X , Y , Z , S , T , U , W) داریم:

$S \to X$
$T \to Y$
$X \to Y$
$XY \to TUZ$

$XYZSTUW - XYZTU = SW$

بنابر رابطه فوق صفات SW حتماً باید عضو کلید کاندید باشد. بستار صفات SW به صورت زیر است:

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

براساس بستار فوق، صفات SW، همه ستون‌ها را تولید می‌کند، پس صفات SW کلید کاندید می‌باشد.

قانون دوم ارسطو

هرگاه عضو کلید کاندید، حاصل از تفاضل قانون اول (روش اول یا دوم)، همه ستون‌ها را تولید کند، آن عضو کلید کاندید، تنها کلید کاندید جدول خواهد بود.

با توجه به وابستگی‌های مطرح شده برای رابطه R(X , Y , Z , S , T , U , W) داریم:

$S \to X$
$T \to Y$
$X \to Y$
$XY \to TUZ$

توجه: به طور کلی در نرمال‌سازی پس از کشف کلید کاندید، جهت کشف وابستگی بخشی و انتقالی و کنترل بهتر وابستگی‌ها می‌توان وابستگی‌ها را بهینه و کمینه کرد، اما با عدم حذف وابستگی‌های اصلی و عدم درج وابستگی‌ها فرعی. در اغلب موارد خود وابستگی‌ها گویای وابستگی بخشی و انتقالی هستند و نیاز به کمینه‌سازی وابستگی‌ها هم نیست. مهمترین عنصر نرمال‌سازی صحیح، کشف دقیق و درست کلید کاندید توسط **قوانین چهارگانه** است.

توجه: در وابستگی $XY \to TUZ$ از وابستگی $X \to Y$ ستون X می‌دهد Y را، پس درسمت چپ وابستگی $XY \to TUZ$ ستون Y زائد است، چون X می‌تواند خود Y را تولید کند، پس به جای وابستگی $XY \to TUZ$ می‌توان وابستگی کمینه $X \to TUZ$ را جایگزین کرد.

پس از حذف وابستگی‌های فرعی و قابل تولید از سایر وابستگی‌ها، وابستگی‌های کمینه به صورت زیر است:

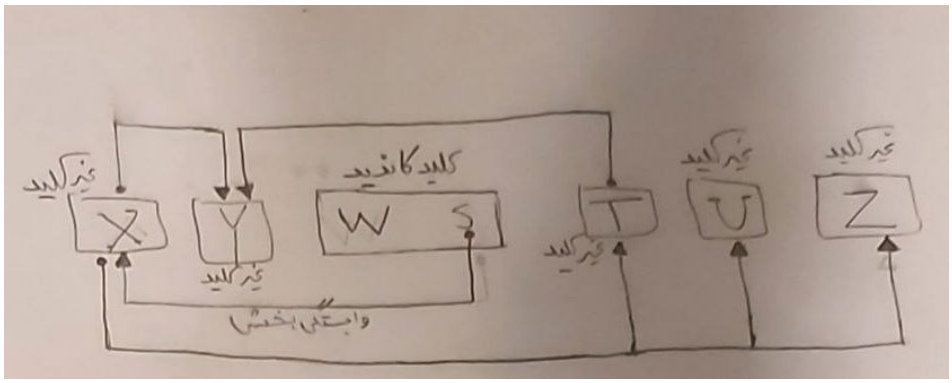
$S \to X$

$T \rightarrow Y$
$X \rightarrow Y$
$X \rightarrow TUZ$

توجه: بر اساس وابستگی‌های فوق واضح است که صفت S همه ستون‌ها را به طور مستقیم و غیرمستقیم بجز صفت W را تولید می‌کند، بستار صفت S به صورت زیر است:

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

نمودار وابستگی‌های جدول R به صورت زیر است:



به طور کلی می‌توان شروط قرار داشتن یک جدول در نرمال فرم دوم را به صورت زیر بیان کرد:

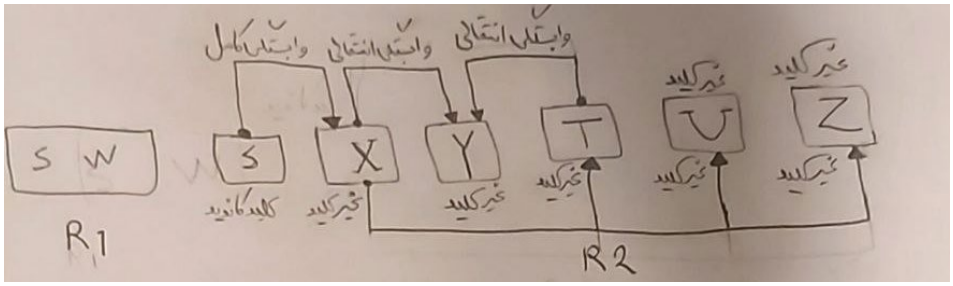
- جدول باید در نرمال فرم اول باشد.
- جدول باید فاقد وابستگی بخشی باشد.

وابستگی بخشی: وابستگی یک مؤلفه غیرکلیدی، به جزئی از کلید کاندید را وابستگی بخشی می‌نامند. با توجه به این که ترکیب صفات SW کلید کاندید رابطه R می‌باشد، وابستگی تابعی زیر:

$S \rightarrow X$

به عنوان وابستگی بخشی محسوب می‌گردند. بنابراین جدول R در نرمال فرم دوم قرار ندارد.

بنابراین برای قرار دادن جدول R، در نرمال فرم دوم، باید وابستگی بخشی از جدول R خارج گردد.



بنابراین جدول R به دو جدول کوچکتر تجزیه می‌گردد:

$R_1(\underline{S, W})$ $R_2(\underline{S}, X, Y, T, U, Z)$

با توجه به کلید کاندید SW در جدول R_1 می‌توان دریافت که این جدول فاقد وابستگی بخشی و انتقالی است، بنابراین در نرمال فرم سوم قرار دارد. همچنین از آن جا که سمت چپ تمام وابستگی‌های جدول R_1، ابرکلید است، بنابراین جدول R_1 در سطح BCNF نیز قرار دارد. از نگاه دیگر می‌توان گفت جدول R_1 تمام کلید است پس BCNF است، از منظری دیگر نیز می‌توان گفت هر جدول با 2 ستون همواره و تحت هر شرایطی قطعا BCNF است. مطابق خواسته سوال راه حل را تا سطح 3NF برای جدول R_2 ادامه می‌دهیم.

به طور کلی می‌توان شروط قرار داشتن یک جدول در نرمال فرم سوم را به صورت زیر بیان کرد:

- جدول باید در نرمال فرم دوم باشد.
- جدول باید فاقد وابستگی انتقالی باشد.

وابستگی انتقالی: وابستگی یک مؤلفه غیرکلیدی به یک مؤلفه غیرکلیدی دیگر را وابستگی انتقالی می‌نامند.

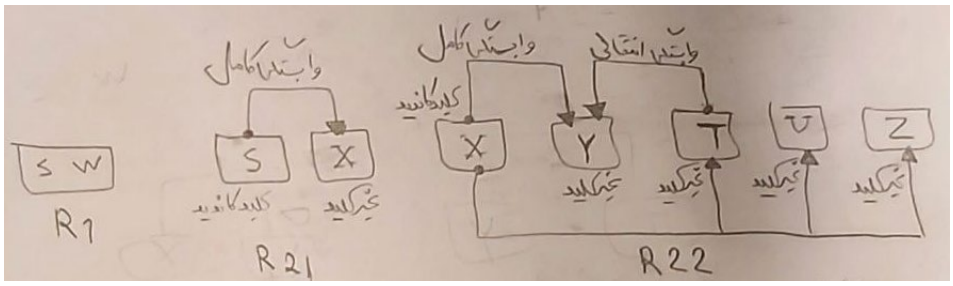
در جدول R_2 وابستگی‌های تابعی زیر:

$X \rightarrow Y$

$T \rightarrow Y$

به عنوان وابستگی انتقالی محسوب می‌گردد. بنابراین جدول R_2 در نرمال فرم سوم قرار ندارد.

بنابراین برای قرار دادن جدول R_2، در نرمال فرم سوم، باید وابستگی‌های انتقالی از جدول R_2 خارج گردند.



بنابراین جدول R_2 به دو جدول کوچکتر تجزیه می‌گردد.

$R_{21}(\underline{S}, X)$ $R_{22}(\underline{X}, Y, T, U, Z)$

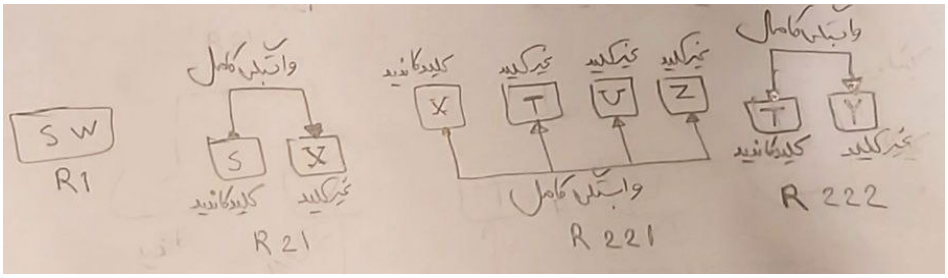
با توجه به کلید کلندید در جدول R_{21} می‌توان دریافت که این جدول فاقد وابستگی بخشی و انتقالی است و بنابراین در نرمال فرم سوم قرار دارد. همچنین از آن جا که سمت چپ تمام وابستگی‌های جدول R_{21}، ابرکلید است، بنابراین جدول R_{21} در سطح BCNF نیز قرار دارد.

توجه مهم: عمل نرمال‌سازی، قطعه و قطعه و مرحله به مرحله انجام می‌شود، و در هر مرحله و هر قطعه، در صورت وجود وابستگی بخشی و انتقالی، کل ملحقات آن به مرحله بعد منتقل می‌شود. مطابق خواسته سوال راه حل را تا سطح 3NF برای جدول R_{22} ادامه می‌دهیم.

در جدول R_{22} وابستگی تابعی زیر:

$T \rightarrow Y$

به عنوان وابستگی انتقالی محسوب می‌گردد. بنابراین جدول R_{22} در نرمال فرم سوم قرار ندارد.
بنابراین برای قرار دادن جدول R_{22}، در نرمال فرم سوم، باید وابستگی انتقالی از جدول R_{22} خارج گردد.



بنابراین جدول R_{22} به دو جدول کوچکتر تجزیه می‌گردد.

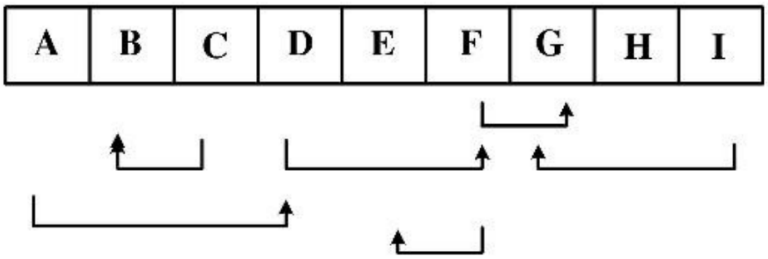
$R_{221}(\underline{X}, T, U, Z)$ $R_{222}(\underline{T}, Y)$

با توجه به کلیدهای کلندید در دو جدول R_{221} و R_{222} می‌توان دریافت که این دو جدول فاقد وابستگی بخشی و انتقالی هستند و بنابراین در نرمال فرم سوم قرار دارند. همچنین از آن جا که سمت چپ تمام وابستگی‌های دو جدول R_{221} و R_{222} ، ابرکلید است، بنابراین دو جدول R_{221} و R_{222} در سطح BCNF نیز قرار دارد. به این ترتیب با تجزیه جدول R به چهار جدول R_1، R_{21} ، R_{221} و R_{222} می‌توان جدول R را در نرمال فرم سوم و سطح BCNF قرار داد.

توجه: منظور از R_{22}در گزینه چهارم همان R_{222} است که طراح محترم در نام‌گذاری دچار خطا شده است. بنابراین پُرواضح است که گزینه چهارم پاسخ سوال است.

۸۱- جدول روبه‌رو و وابســـتگی‌های تابعی زیر مفروض اســـت. کدام یک از موارد زیر می‌تواند کاندیدای کلید اصلی رابطه باشد؟

(مهندسی *IT*-دولتی۱۴۰۱)



- $I \rightarrow G$
- $F \rightarrow G$
- $D \rightarrow F$
- $A \rightarrow D$
- $C \rightarrow\!\!\rightarrow B$ (وابستگی MVD چند مقداره)
- $F \rightarrow E$

۱) A , B , C , H , I
۲) A , C , D , F , I , H
۳) A , C , F , I
۴) A , C , D , F , H , I , B

عنوان کتاب: پایگاه داده‌ها

مولف: استاد خلیلی‌فر

ناشر: انتشارات راهیان ارشد و دکتری

آدرس سایت انتشارات بابان: baban.ir

پاسخ تست‌های فصل نهم: نرمال‌سازی

۸۱- گزینه (۱) صحیح است.

به طور کلی کلید کاندید باید دو شرط زیر را داشته باشد:
۱- ابرکلید باشد (خاصیت کلیدی داشته باشد) یعنی همه خصیصه‌ها را تولید کند.
۲- عضو زائد نداشته باشد.
به طور کلی عضو کلید کاندید از روابط زیر به دست می‌آید:

قانون اول ارسطو

روش اول:

اجتماع تمام خصیصه‌های سمت راست وابستگی‌های غیربدیهی- تمام خصیصه‌های جدول = عضو کلید کاندید

روش دوم:

$$\text{عضو کلید کاندید} = R - \bigcup_{i=1}^{n} [\,(\text{راست})\, y_i - (\text{چپ})\, x_i]$$

با توجه به وابستگی‌های مطرح شده برای رابطه R(A , B , C , D , E , F , G , H , I) داریم:

$I \to G$
$F \to G$
$D \to F$
$A \to D$
$C \to\to B$
$F \to E$

$$ABCDEFGHI - DEFG = ABCHI$$

بنابر رابطه فوق صفات ABCHI حتماً باید عضو کلید کاندید باشد. بستار صفات ABCHI به صورت زیر است:

$$\{ABCHI\}^+ = \{A,B,C,H,I,D,F,E,G\}$$

براساس بستار فوق، صفات ABCHI، همه ستون‌ها را تولید می‌کند، پس صفات ABCHI کلید کاندید می‌باشد.

توجه: دقت کنید که در روابط زیر:

روش اول:

اجتماع تمام خصیصه‌های سمت راست وابستگی‌های غیربدیهی- تمام خصیصه‌های جدول = عضو کلید کاندید

روش دوم:

$$\text{عضو کلید کاندید} = R - \bigcup_{i=1}^{n} [\,(\text{راست})\, y_i - (\text{چپ})\, x_i]$$

اجتماع تمام خصیصه‌های سمت راست وابستگی‌های غیربدیهی به معنی وابستگی‌های تابعی غیربدیهی Functional dependencies است و به معنی وابستگی‌های چندمقداری Multivalued dependencies نیست، بنابراین در تفاضل فوق، سمت راست وابستگی چندمقداری $C \rightarrow\rightarrow B$ نادیده گرفته شد.

قانون دوم ارسطو

هرگاه عضو کلید کاندید، حاصل از تفاضل قانون اول (روش اول یا دوم)، همه ستون‌ها را تولید کند، آن عضو کلید کاندید، تنها کلید کاندید جدول خواهد بود. بنابراین صفات ABCHI تنها کلید کاندید جدول R است. بنابراین پُرواضح است که گزینه اول پاسخ سوال است.

تست‌های فصل ششم: زبان SQL دستورات DML

۸۲- با فرض وجود جداول S و P و J و SPJ به صورت زیر:

-S(SID,SNAME,STATUS,CITY) جدول تولیدکنندگان شامل شماره تولیدکننده، نام تولیدکننده، وضعیت، شهر

-P(PID,PNAME,WEIGHT) جدول انواع قطعات شامل شماره قطعه، نام نوع قطعه، وزن

-J(JID,JNAME,CITY) جدول پروژه‌ها شامل شماره پروژه، نام پروژه، شهر

-SPJ(SID,PID,JID,QTY) جدول قطعات تولیدشده شامل تعداد قطعه (QTY) تولیدشده از نوع PID توسط تولیدکننده SID در پروژه JID

با توجه به اینکه هر تولیدکننده می‌تولند در یک پروژه انواع مختلفی از قطعات را با تعداد متفاوت تولید کند، کدام یک از گزینه‌های زیر، پرس‌وجوی SQL لازم برای واکشی لیست نام تولیدکنندگانی است که مجموعا بیشترین قطعه (از همه انواع قطعات) را در پروژه شماره j302 تولید کرده‌اند؟

(مهندسی *IT*-دولتی۱۴۰۱)

۱)
```
SELECT S.SNAME
FROM S NATURAL JOIN SPJ
WHERE SPJ.JID= 'j302'
GROUP BY SID
HAVING SPJ.SUM(SPJ.QTY)>=(SELECT MAX (SPJ2.QTY)
                          FROM SPJ AS SPJ2
                          WHERE SPJ2.JID = 'j302'
                          GROUP BY SPJ2.SID)
```

۲)
```
SELECT S.SNAME
FROM S NATURAL JOIN SPJ
WHERE SPJ.JID= 'j302' AND SPJ.QTY >= ALL (SELECT SUM (SPJ2.QTY)
                                          FROM SPJ AS SPJ2)
```

۳)
```
SELECT S.SNAME
FROM S NATURAL JOIN SPJ
WHERE SPJ.JID= 'j302'
GROUP BY S.SID
HAVING SUM(SPJ.QTY) >= ALL (SELECT SUM (SPJ2.QTY)
                            FROM SPJ AS SPJ2
                            WHERE SPJ2.JID = 'j302'
                            GROUP BY SPJ2.SID)
```

۴)
```
SELECT S.SNAME
FROM S
WHERE S.SID IN (SELECT SPJ.SID
                FROM SPJ
                WHERE SPJ.JID = 'j302' AND SPJ.QTY = (SELECT MAX (SPJ2.QTY)
                                                      FROM SPJ AS SPJ2)
```

عنوان کتاب: پایگاه داده‌ها

مولف: استاد خلیلی‌فر

ناشر: انتشارات راهیان ارشد و دکتری

آدرس سایت گروه بابان: khalilifar.ir

پاسخ تست‌های فصل ششم: زبان SQL دستورات DML

۸۲- گزینه (۳) صحیح است.

جداول زیر را در نظر بگیرید:

SID	SNAME	STATUS	CITY
S1	Sn1	1	C1
S2	Sn2	2	C2
S3	Sn3	3	C3
S4	Sn4	4	C4

جدول تولیدکننده S

SID	PID	JID	QTY
S1	P1	J1	10
S1	P2	J1	20
S1	P1	J302	30
S1	P2	J302	40
S2	P1	J1	50
S2	P1	J302	20
S2	P2	J302	50
S3	P1	J302	70
S3	P3	J302	80

جدول SPJ

JID	JNAME	CITY
J1	Jn1	C1
J302	Jn302	C302

جدول پروژه J

PID	PNAME	WEIGHT
P1	Pn1	100
P2	Pn2	200
P3	Pn3	300

جدول قطعه P

«کدام گزینه، پرس‌وجوی SQL لازم برای واکشی لیست نام تولیدکنندگانی است که مجموعا بیشترین قطعه (از همه انواع قطعات) را در پروژه شماره 302 تولید کرده‌اند؟»

گزینه سوم درست است، زیرا مطابق پرس‌وجوی مطرح شده، داریم:

```
SELECT S.SNAME
FROM S Natural join SPJ
WHERE SPJ.JID = 'j302'
GROUP BY S.SID
HAVING SUM(SPJ.QTY) >= ALL (SELECT SUM (SPJ2.QTY)
                            FROM SPJ AS SPJ2
                            WHERE SPJ2.JID = 'j302'
                            GROUP BY SPJ2.SID)
```

که البته پرس و جوی مطرح شده در گزینه‌ی سوم کمی خطای نحوی دارد، که فرم اصلاح شده آن به صورت زیر است:

```
SELECT S.SNAME
FROM S Natural join SPJ
WHERE SPJ.JID = 'j302'
GROUP BY S.SID , S.SNAME
HAVING SUM(SPJ.QTY) >= ALL (SELECT SUM (SPJ2.QTY)
                            FROM SPJ AS SPJ2
                            WHERE SPJ2.JID = 'j302'
                            GROUP BY SPJ2.SID)
```

توجه: در فرم اصلاح شده‌ی گزینه‌ی سوم، ستون S.SNAME جلوی دستور Group By پرس‌وجوی **خارجی** قرار گرفت. همواره به غیر از توابع آماری، همه ستون‌های جلوی دستور Select باید زیرمجموعه یا مساوی ستون‌های دستور Group By باشد.

توجه: عملگر Natural join در برخی نسخه‌های SQL پشتیبانی نمی‌شود. و به جای آن عملگرهای Jion و Inner Join البته با عملکرد تا حدی مشابه پشتیبانی می‌شود. اما در بررسی گزینه سوم این مورد را نادیده گرفتیم. البته در سوالات پایگاه داده همین کنکور مهندسی فناوری اطلاعات سال 1401 سوال دیگری از عملگر Natural join آمد که درکلید اولیه بدون خطای نحوی اعلام شد، اما در کلید نهایی نظر طراح عوض شد و عملگر Natural join را در SQL دارای خطای نحوی دانست و کلید نهایی 2 گزینه‌ای شد! یعنی هم خطا دارد و هم خطا ندارد!

توجه: زیر پرس‌وجوی مطرح شده در جلوی دستور All، یک Normal Subquery است.

بنابراین ابتدا زیر پرس‌وجوی داخلی یک بار و برای همیشه اجرا می‌گردد، سپس پرس‌وجوی خارجی به ازای حرکت در هر یک از گروه‌های خود (به دلیل اعمال گروه‌بندی) از مقادیر زیر پرس‌وجوی داخلی استفاده می‌کند. بنابراین در ادامه برای زیر پرس‌وجوی داخلی داریم:

```
SELECT SUM (SPJ2.QTY)
FROM SPJ AS SPJ2
WHERE SPJ2.JID = 'j302'
GROUP BY SPJ2.SID
```

توجه: البته نیازی هم نبود در پرس‌وجوی داخلی جدول SPJ به SPJ2 مجدد نام‌گذاری شود و همان SPJ کافی بود.

روال کار بدین صورت است که ابتدا سراغ جدول SPJ می‌رویم، سپس شرط SPJ2.JID = 'J302' بر روی جدول SPJ اعمال می‌گردد، در ادامه دستور Group by بر روی ستون SID اعمال می‌گردد، در انتها دستور SUM(SPJ2.QTY) برای هر گروه به طور مستقل محاسبه می‌گردد و در خروجی پرس‌وجو قرار می‌گیرد. بنابراین در ادامه داریم:

```
FROM SPJ AS SPJ2
WHERE SPJ2.JID = 'J302'
```

⇒

SID	PID	JID	QTY
S_1	P_1	J_{302}	30
S_1	P_2	J_{302}	40
S_2	P_1	J_{302}	20
S_2	P_2	J_{302}	50
S_3	P_1	J_{302}	70
S_3	P_3	J_{302}	80

حال در ادامه براساس زیر پرس‌وجوی مطرح شده، ابتدا دستور Group by بر روی ستون SID اعمال می‌گردد و سپس دستور SUM (SPJ2.QTY) بر روی ستون QTY در هر گروه به طور مستقل انجام می‌گردد. و خروجی‌های آن جهت استفاده پرس‌وجوی خارجی ذخیره می‌گردد، بنابراین در ادامه داریم:

(S_1)			(S_2)			(S_3)		
P_1	J_{302}	30	P_1	J_{302}	20	P_1	J_{302}	70
P_2	J_{302}	40	P_2	J_{302}	50	P_3	J_{302}	80
گروه اول			گروه دوم			گروه سوم		
SUM(QTY) = 70			SUM(QTY) = 70			SUM(QTY) = 150		

در واقع زیر پرس‌وجوی داخلی مطرح شده در جلوی دستور All، برای هر تولید کننده حاضر در پروژه J_{302}، تعداد کل قطعات تولید شده (از تمام انواع قطعات) توسط آن تولید کننده در پروژه J_{302} را می‌دهد. بنابراین داریم:

... >= All(70,70,150)

حال در ادامه، پرس‌وجوی خارجی به ازای حرکت در هر یک از گروه‌های خود، از مقادیر زیر پرس‌وجوی داخلی استفاده می‌کند.

توجه: دستور having، حرکت بر روی گروه‌ها را میسر می‌کند، همانطور که دستور where، حرکت بر روی سطرها را میسر می‌کند.

الحاق طبیعی دو جدول S و SPJ، تولیدکنندگانی که حداقل یک قطعه برای یک پروژه را تولید کرده‌اند، استخراج می‌کند، همچنین پس از اعمال شرط SPJ.JID = 'J302' داریم:

S natural join SPJ
where SPJ.JID = 'J302' ⇒

SID	SNAME	STATUS	CITY	PID	JID	QTY
S_1	Sn1	1	C1	P_1	J_{302}	30
S_1	sn1	1	C1	P_2	J_{302}	40
S_2	Sn2	2	C2	P_1	J_{302}	20
S_2	Sn2	2	C2	P_2	J_{302}	50
S_3	Sn3	3	C3	P_1	J_{302}	70
S_3	Sn3	3	C3	P_3	J_{302}	80

در ادامه برای سادگی از ستون‌های STATUS و CITY صرف‌نظر می‌کنیم، بنابراین داریم:

SID	SNAME	PID	JID	QTY
S_1	Sn1	P_1	J_{302}	30
S_1	Sn1	P_2	J_{302}	40
S_2	Sn2	P_1	J_{302}	20
S_2	Sn2	P_2	J_{302}	50
S_3	Sn3	P_1	J_{302}	70
S_3	Sn3	P_3	J_{302}	80

حال، براساس پرس‌وجوی مورد نظر، دستور S.SNAME و Group by S.SID اجرا می‌گردد، بنابراین داریم: (SID , SNAME)

$(S_1, Sn1)$			$(S_2, Sn2)$			$(S_3, Sn3)$		
P_1	J_{302}	30	P_1	J_{302}	20	P_1	J_{302}	70
P_2	J_{302}	40	P_2	J_{302}	50	P_3	J_{302}	80
گروه اول			گروه دوم			گروه سوم		

حال در ادامه، براساس پرس‌وجوی خارجی، دستور having SUM(SPJ.QTY) بر روی هر گروه به طور جداگانه اعمال می‌گردد:

توجه: دستور having بر روی گروه‌ها، اعمال می‌گردد.

$(S_1, Sn1)$	$(S_2, Sn2)$	$(S_3, Sn3)$
P_1 J_{302} 30	P_1 J_{302} 20	P_1 J_{302} 70
P_2 J_{302} 40	P_2 J_{302} 50	P_3 J_{302} 80
گروه اول	گروه دوم	گروه سوم
SUM(QTY) = 70	SUM(QTY) = 70	SUM(QTY) = 150

در واقع، پرس‌وجوی خارجی، برای هر تولید کننده‌ای که در پروژه J_{302} شرکت داشته است، اگر تعداد کل قطعات تولید شده (از تمام انواع قطعات) توسط وی در پروژه J_{302} **بیشتر** از تعداد کل قطعات تولید شده (از تمام انواع قطعات) توسط تک‌تک تولید کنندگان در پروژه J_{302} باشد، نام آن تولید کننده را در خروجی می‌آورد.

بنابراین در ادامه داریم:

گام اول: بررسی گروه اول. $70 >= \text{All}(70,70,150)$

پاسخ: SUM(QTY) با مقدار 70 در گروه اول از همه مقادیر (70,70,150) بزرگتر یا مساوی نیست.

بنابراین تولیدکننده Sn1، به عنوان تولیدکننده‌ای که بیشترین تعداد قطعه (از تمام انواع قطعات) را برای پروژه J_{302} تولید کرده باشد، در خروجی قرار نمی‌گیرد.

گام دوم: بررسی گروه دوم. $70 > = \text{All}(70,70,150)$

پاسخ: SUM(QTY) با مقدار 70 در گروه دوم از همه مقادیر (70,70,150) بزرگتر یا مساوی نیست.

بنابراین تولیدکننده Sn2، به عنوان تولیدکننده‌ای که بیشترین تعداد قطعه (از تمام انواع قطعات) را برای پروژه J_{302} تولید کرده باشد، در خروجی قرار نمی‌گیرد.

گام سوم: بررسی گروه سوم. $150 >= \text{All}(70,70,150)$

پاسخ: SUM(QTY) با مقدار 150 در گروه سوم از همه مقادیر (70,70,150) بزرگتر یا مساوی است.

بنابراین تولیدکننده Sn3، به عنوان تولیدکننده‌ای که بیشترین تعداد قطعه (از تمام قطعات) را برای پروژه J_{302} تولید کرده باشد، در خروجی قرار می‌گیرد.

بنابراین پرس‌وجوی مطرح شده در گزینه‌ی سوم، نام تولیدکنندگانی که **بیشترین** تعداد قطعه (از تمام انواع قطعات) را برای پروژه کد J_{302} تولید کرده‌اند را به صورت زیر در خروجی نمایش می‌دهد.

SNAME
Sn3

گزینه اول نادرست است، زیرا مطابق پرس‌وجوی مطرح شده، داریم:

که البته پرس و جوی مطرح شده در گزینه‌ی اول کمی خطای نحوی دارد، که فرم اصلاح شده آن به صورت زیر است:

```
SELECT S.SNAME
FROM S Natural join SPJ
WHERE SPJ.JID = 'j302'
GROUP BY S.SID , S.SNAME
HAVING SUM(SPJ.QTY) >= ALL (SELECT MAX (SPJ2.QTY)
                            FROM SPJ AS SPJ2
                            WHERE SPJ2.JID = 'j302'
                            GROUP BY SPJ2.SID)
```

توجه: در فرم اصلاح شده‌ی گزینه‌ی اول، ستون S.SNAME جلوی دستور Group By پرس‌وجوی **خارجی** قرار گرفت. همچین عملگر ALL پشت پرانتز MAX قرار گرفت چون پرانتز MAX چندین مقدار خروجی دارد و ذکر فقط « =< » خطای کامپایلری دارد.

گام اول: بررسی گروه اول. $70 >= All(40,50,80)$

پاسخ: SUM(QTY) با مقدار 70 در گروه اول از همه مقادیر (40,50,80) بزرگتر یا مساوی نیست.

گام دوم: بررسی گروه دوم. $70 >= All(40,50,80)$

پاسخ: SUM(QTY) با مقدار 70 در گروه دوم از همه مقادیر (40,50,80) بزرگتر یا مساوی نیست.

گام سوم: بررسی گروه سوم. $150 >= All(40,50,80)$

پاسخ: SUM(QTY) با مقدار 150 در گروه سوم از همه مقادیر (40,50,80) بزرگتر یا مساوی است.

بنابراین پرس‌وجوی مطرح شده در گزینه‌ی اول، نام تولیدکنندگانی که مجموع تولیدی آن در پروژه J_{302} از MAX همه تولیدکنندگان پروژه J_{302} بیشتر است را به صورت زیر در خروجی نمایش می‌دهد.

SNAME
Sn3

گزینه دوم نادرست است، زیرا مطابق پرس‌وجوی مطرح شده، داریم:

```
SELECT S.SNAME
FROM S Natural join SPJ
WHERE SPJ.JID = 'j302'
AND
SPJ.QTY>= ALL (SELECT SUM (SPJ2.QTY)
               FROM SPJ AS SPJ2)
```

نیازی به استفاده از عملگر ALL نیست چون زیرپرس‌وجوی داخلی فقط یک مقدار خروجی دارد و همان «=<» کافی است. زیرپرس‌وجوی داخلی جمع تمام قطعات تولیدی در تمام پروژه‌ها را

حساب می‌کند به صورت زیر:

No column name
370

در ادامه داریم:

```
SELECT S.SNAME
FROM S Natural join SPJ
WHERE SPJ.JID = 'j302'
AND
SPJ.QTY>= 370
```

الحاق طبیعی دو جدول S و SPJ، تولیدکنندگانی که حداقل یک قطعه برای یک پروژه را تولید کرده‌اند، استخراج می‌کند، همچنین پس از اعمال شرط SPJ.JID = 'J302' داریم:

S natural join SPJ
where SPJ.JID = 'J302' ⇒

SID	SNAME	STATUS	CITY	PID	JID	QTY
S_1	Sn1	1	C1	P_1	J_{302}	30
S_1	sn1	1	C1	P_2	J_{302}	40
S_2	Sn2	2	C2	P_1	J_{302}	20
S_2	Sn2	2	C2	P_2	J_{302}	50
S_3	Sn3	3	C3	P_1	J_{302}	70
S_3	Sn3	3	C3	P_3	J_{302}	80

و در نهایت پس از اعمال شرط SPJ.QTY>=370 خروجی به صورت زیر است:

SNAME
NULL

پرس‌وجوی گزینه دوم کار معناداری را انجام نمی‌دهد.

گزینه چهارم نادرست است، زیرا مطابق پرس‌وجوی مطرح شده، داریم:

```
SELECT S.SNAME
FROM S
WHERE S.SID IN (SELECT SPJ.SID
                FROM SPJ
                WHERE SPJ.JID = 'j302' AND  SPJ.QTY = (SELECT MAX (SPJ2.QTY)
                                                       FROM SPJ2 AS SPJ)
```

ابتدا زیرپرس‌وجوی زیر انجام می‌شود:

```
SELECT MAX (SPJ2.QTY)
FROM SPJ2 AS SPJ
```

زیرپرس‌وجوی فوق MAX قطعات تولیدی در بین تمام پروژه‌ها را حساب می‌کند به صورت زیر:

No column name
80

در ادامه داریم:

```
SELECT SPJ.SID
FROM SPJ
WHERE SPJ.JID = 'j302' AND  SPJ.QTY = 80   ⇒
```

SID	PID	JID	QTY
S_3	P_3	J_{302}	80

در ادامه از جدول S خواهیم داشت:

```
SELECT S.SNAME
FROM S
WHERE S.SID IN (S3)
```

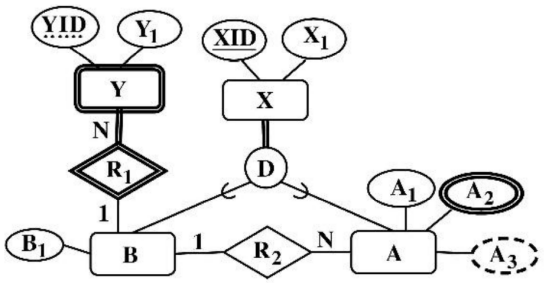
SNAME
Sn3

نام تولیدکننده‌ای استخراج می‌شود که در پروژه J_{302} قطعه تولید کرده‌است و همچنین MAX میزان قطعه تولیدی هم در کل جدول SPJ دارد. پرس‌وجوی گزینه چهارم کار معناداری را انجام نمی‌دهد.

۸۳ – با توجه به نمودار EER زیر، کدام گزینه، طراحی منطقی صحیح موجودیت‌های A و Y را نمایش می‌دهد؟

[در نمودار EER، از خط ممتد در زیر صفت برای نمایش شناسه و از خط چین برای نمایش صفت ممیزه یا کلید جزئی استفاده شده است. در طراحی جداول در گزینه‌ها، از خط ممتد در زیر صفت(ها) برای نمایش کلید اصلی و از خط چین برای نمایش کلید خارجی استفاده شده است.]

(مهندسی *IT*–دولتی۱۴۰۱)



۱) $Y\ (\underline{XID}, \underline{YID}, Y_1)$ $A\ (\underline{XID}, \underline{BXID}, A_1, A_2, A_3)$

۲) $Y\ (\underline{YID}, Y_1, \underline{XID})$ $AA2\ (\underline{XID}, \underline{A_2})$ $A\ (\underline{XID}, A_1, A_3, \underline{BXID})$

۳) $Y\ (\underline{YID}, Y_1)$ $A\ (\underline{XID}, A_1, A_2, A_3)$

۴) $Y(\underline{XID}, \underline{YID}, Y_1)$ $AA_2(\underline{XID}, \underline{A_2})$ $A\ (\underline{XID}, A_1, \underline{BXID})$

عنوان کتاب: پایگاه داده‌ها

مولف: استاد خلیلی‌فر

ناشر: انتشارات راهیان ارشد و دکتری

آدرس سایت گروه بابان: khalilifar.ir

پاسخ تست‌های فصل سوم: مدل رابطه‌ای

۸۳ - گزینه (۴) صحیح است.

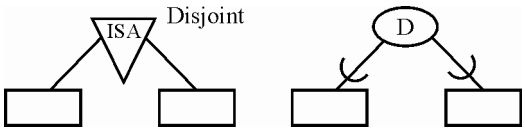
جهت تبدیل مدل ER (مدل تحلیل) به مدل مدل رابطه‌ای (مدل طراحی یا طراحی منطقی) از سمت و موجودیتی شروع کنید که ورودی صفت از موجودیت دیگری نداشته باشد.

نگاشت رابطه ISA یا وراثت به مدل رابطه‌ای

در رابطه ISA رابطه پدر با فرزندان به دو صورت رابطه اختیاری یا جزئی یا بخشی (Partial) با نماد خط عمودی و رابطه اجباری یا کلی یا کامل (Total) با نماد خط مضاعف عمودی است و رابطه فرزندان با پدر به دو صورت رابطه متصل یا پوشا یا تخصیص غیرمجزا (Overlap) و رابطه منفصل یا غیرپوشا یا تخصیص مجزا (Disjoint) می‌باشد.

توجه: در یک رابطه اجباری یا کلی (Total)، هر نمونه از موجودیت پدر حتماً می‌بایست با یکی از نمونه موجودیت‌های فرزند در ارتباط باشد.

رابطه منفصل یا غیرپوشا یا تخصیص مجزا (Disjoint) مابین فرزندان و پدر به دو شیوه زیر نشان داده می‌شود:



توجه: در رابطه منفصل یا غیرپوشا یا تخصیص مجزا (Disjoint) ارتباط پدر با فرزندان فقط و فقط یک به یک است. و همچنین اشتراک نمونه موجودیت‌ها میان موجودیت‌های فرزند با یک نمونه موجودیت از پدر همواره تهی است.

توجه: در یک رابطه منفصل یا غیرپوشا یا تخصیص مجزا (Disjoint)، نمونه موجودیت‌های فرزند نمی‌توانند به طور همزمان با نمونه‌ای از موجودیت پدر در ارتباط باشند که این یعنی همان رابطه یک به یک میان پدر و فرزندان.

بخشی از مدل EER رسم شده در صورت سوال، یک رابطه اجباری یا کلی یا کامل (Total) مابین موجودیت X و موجودیت‌های A و B و یک رابطه منفصل یا غیرپوشا یا تخصیص مجزا (Disjoint) را مابین موجودیت‌های A و B و موجودیت X نشان می‌دهد. که در ادامه فرآیند نگاشت آن به مدل رابطه‌ای را بیان می‌کنیم.

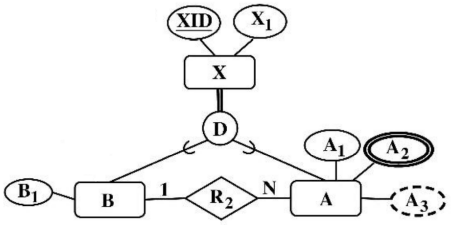
توجه: در یک رابطه اجباری یا کلی یا کامل (Total)، هر نمونه از موجودیت پدر حتماً می‌بایست با یکی از نمونه موجودیت‌های فرزند در ارتباط باشد. برای‌مثال در این سؤال، هر نمونه از

موجودیت X حتماً می‌بایست با یکی از نمونه موجودیت‌های A یا B در ارتباط باشد. به عبارت دیگر نمی‌توان نمونه‌ای از موجودیت X داشت که با هیچ یک از نمونه موجودیت‌های A یا B در ارتباط نیست.

توجه: در حالت رابطه اجباری مابین موجودیت X و موجودیت‌های A و B به ازای هر نمونه از موجودیت X، حتما نمونه موجودیتی از A یا B وجود دارد که با X رابطه برقرار کند، پس در این حالت **طراحی بهینه** این است که کل صفات موجودیت X در دو جدول موجودیت‌های A و B قرار داده شود و یک طراحی به شکل مدل **دو جدولی** ایجاد گردد، همچنین از آنجاکه رابطه مابین موجودیت‌های A و B و موجودیت X یک رابطه منفصل یا غیرپوشا یا تخصیص مجزا (Disjoint) است، پس رکوردهای تکراری در جداول A و B به ازای یک نمونه موجودیت از موجودیت X ایجاد نمی‌گردد، که افزونگی حاصل از تکرار رکوردها در جداول A و B سبب شود رکورد نمونه موجودیت‌های X در جدول X نگهداری شود و یک مدل سه جدولی ایجاد گردد. پس در این حالت طراحی بهینه این است که **کل صفات** موجودیت X در دو جدول موجودیت‌های A و B قرار داده شود و یک طراحی به شکل مدل دو جدولی ایجاد گردد. به عبارت دیگر برای موجودیت پدر، رابطه جداگانه ایجاد نمی‌کنیم و کل ویژگی‌های رابطه پدر را به موجودیت‌های فرزند اضافه می‌کنیم. **در یک عبارت ساده‌تر و واضح‌تر در شرایط پدر اجباری (Total) و فرزندان غیرپوشا (Disjoint)، کل صفات باباشو بیارید جلوی چشم بچه‌هاش، به همین سادگی.**

در ادامه فرآیند نگاشت نمودار (ISA) به مدل رابطه‌ای را شرح می‌دهیم:

مدل تحلیل (نمودار (ISA))



مدل طراحی (مدل رابطه‌ای)

همانطور که قبل‌تر گفتیم، در شرایط پدر اجباری (Total) و فرزندان غیرپوشا (Disjoint)، برای موجودیت پدر، رابطه جداگانه ایجاد نمی‌کنیم و کل ویژگی‌های رابطه پدر را به موجودیت‌های فرزند اضافه می‌کنیم.

XID	X1	A1	A2

جدول غیرنرمال A

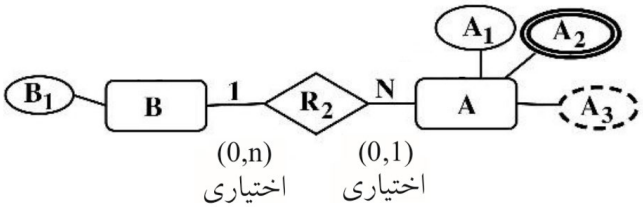
XID	X1	B1	…

جدول B

نگاشت رابطه یک به چند بین دو موجودیت به مدل رابطه‌ای

مســـتقل از اختیاری یا اجباری بودن موجودیت‌ها، هر موجودیت به یک جدول تبدیل می‌گردد. و کلید کلنلید جدول یک در جدول چند به عنوان کلید خارجی تعریف می‌گردد. همچنین صـــفات متصل به رابطه، درون چدول چند مستتر می‌شود.
بخشی از مدل EER رسم شده در صورت سوال، یک رابطه یک به چند در یک موجودیت را نشان می‌دهد. که در ادامه فرآیند نگاشت آن به مدل رابطه‌ای را شرح می‌دهیم.

مدل تحلیل



توجه: در شکل فوق و بر اساس نوع نمایش و نشانه‌گذاری‌ها صفت A_2 **چند مقداری** و صفت A_3 **مشتق (پویا)** اسـت. همانطور که قبل‌تر هم گفتیم کلید کاندید جدول موجودیت A صـفت (XID) است.

توجه: نماد خط افقی نشانه اختیاری بودن موجودیت چسبیده به آن است.
توجه: قید (0,N) نشـان می‌دهد که هر نمونه موجودیت از B حداقل با صـفر و حداکثر با N نمونه موجودیت از A ارتباط دارد و قید (0,1) نشان می‌دهد که هر نمونه موجودیت از A حداقل با صفر و حداکثر با یک نمونه موجودیت از B ارتباط دارد.

مدل طراحی (مدل رابطه‌ای)

همانطور که در مدل طراحی قبل‌تر گفتیم، مدل طراحی جدول B به صـــورت $B(\underline{XID}, X_1, B_1)$ و مدل طراحی جدول غیرنرمال A به صورت $A(\underline{XID}, X_1, A_1, A_2)$ است.

XID	X_1	B_1

جدول B

XID	X_1	A_1	A_2	BXID

جدول غیرنرمال A

توجه: کلید کاندید (XID) جدول یک یعنی **موجودیت B** در جدول چند یعنی **موجودیت A** به عنوان کلید خارجی (BXID) تعریف می‌گردد.

توجه: کلید کاندید جدول یک یعنی **موجودیت B** برابر همان **کلید کاندید سابق** در جدول موجودیت B است. یعنی کلید کاندید جدول B برابر (XID) است.

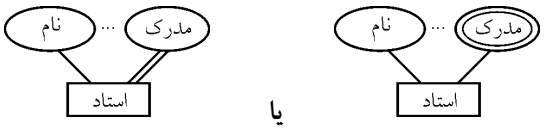
توجه: کلید کاندید جدول چند یعنی **موجودیت A** برابر همان **کلید کاندید سابق** در جدول موجودیت A است. یعنی کلید کاندید جدول A برابر (XID) است.

صفت تک مقداری و چند مقداری

بعضی از صفات چه ساده و چه مرکب فقط می‌توانند یک مقدار را بگیرند که به این صفات، صفت تک‌مقداری می‌گویند. مانند شماره دانشجویی که نمی‌تواند بیش از یک مقدار داشته باشد. این صفات در نمودار ER بصورت معمول نمایش داده می‌شوند.

صفاتی وجود دارند که می‌توانند چندین مقدار را بگیرند مانند صفت مدرک در موجودیت استاد که می‌تواند مقادیر لیسانس، فوق لیسانس و یا دکتری را در خود بگیرد. صفت چند مقداری در نمودار ER بصورت دو دایره مضاعف و یا خط مضاعف متصل شده به آن نمایش داده می‌شود.

مثال:



توجه: به مثال‌های زیر توجه کنید.

صفت ساده تک‌مقداری: مانند کدملی

صفت ساده چندمقداری: مانند مدرک تحصیلی

صفت مرکب تک‌مقداری: مانند تاریخ تولد

صفت مرکب چندمقداری: مانند آدرس

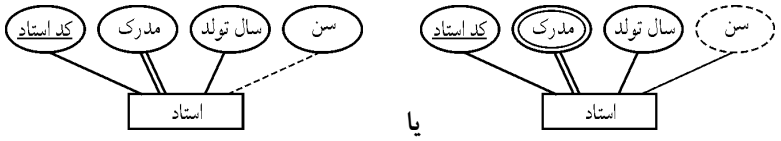
توجه: در بانک اطلاعاتی مبتنی بر مدل رابطه‌ای (جدولی) صفت چندمقداری نداریم.

صفت مشتق (پویا)

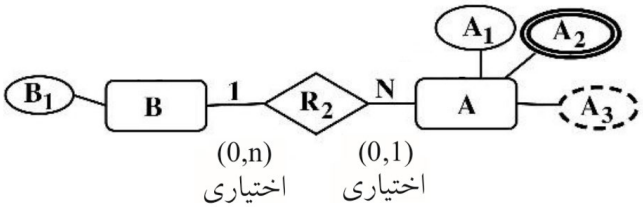
صفتی است که در موجودیت وجود خارجی ندارد ولی در صورت لزوم می‌توان آنرا بدست آورد. صفتی که مقادیر آن مدام در حال تغییر و تحول باشد، صفت پویا یا مشتق محسوب می‌گردد. بنابراین به دلیل تغییرات مداوم، توصیه می‌گردد صفت پویا در جداول بانک اطلاعات مورد استفاده قرار نگیرد و مقدار آن از طریق صفت مرتبط با آن محاسبه گردد. برای‌مثال برای محاسبه صفت سن، می‌توان صفت تاریخ تولد را در نظر گرفت و از روی این صفت، سن را محاسبه نمود.

توجه: صفت مشتق را در نمودار ER با نقطه چین به موجودیت مورد نظر متصل می‌کنند.

مثال:



همانطور که گفتیم بخشی از مدل EER رسم شده در صورت سوال به صورت زیر است:



توجه: در شکل فوق و بر اساس نوع نمایش و نشانه‌گذاری‌ها صفت A_2 **چند مقداری** و صفت A_3 **مشتق(پویا)** است.

مدل طراحی غیرنرمال:

XID	X_1	A_1	A_2	BXID

جدول غیرنرمال A

توجه: کلید کاندید جدول A برابر (XID) است.

توجه: همانطور که گفتیم در نگاشت رابطه یک به چند در دو موجودیت به مدل رابطه‌ای مستقل از اختیاری یا اجباری بودن موجودیت‌ها، هر موجودیت به یک جدول تبدیل می‌گردد. و کلید کلندید جدول یک در جدول چند به عنوان کلید خارجی تعریف می‌گردد. بنابراین در جدول غیرنرمال A صفت (BXID) به عنوان **کلید خارجی** به کلید کاندید (XID) از جدول B ارجاع می‌کند.

توجه: همانطور که گفتیم توصیه می‌گردد صفت **مشتق** (پویا) در جداول بانک اطلاعات مورد استفاده قرار نگیرد و مقدار آن از طریق صفت مرتبط با آن محاسبه گردد، بنابراین صفت مشتق A_3 در مدل EER در جدول A به عنوان ستون نیامده است.

توجه: در بانک اطلاعاتی مبتنی بر مدل رابطه‌ای (جدولی) صفت چندمقداری نداریم، صفت A_2 در مدل EER مطرح شده در صورت سوال بر اساس نوع نمایش و نشانه‌گذاری صفت چند مقداری است و همین عامل باعث شده است که بگوییم جدول A غیرنرمال است. که برای حل این مساله و حذف صفت چند مقداری A_2 از جدول A، این جدول باید در سطح اول نرمال قرار بگیرد.

نرمال فرم اول

به طور کلی می‌توان شروط قرار داشتن یک جدول در نرمال فرم اول را به صورت زیر بیان کرد:

- دارای حداقل یک کلید کاندید باشد.
- همه خصیصه‌های آن غیرقابل تجزیه باشند (جدول باید فاقد خصیصه‌های مرکب باشد)
- همه خصیصه‌های آن تک مقداری باشند (جدول باید فاقد خصیصه‌های چند مقداری باشد)

واضح است که جدول مطرح شده یعنی جدول A در فرم اول نرمال قرار **ندارد**. و باید با حذف خصیصه‌های چند مقداری آن یعنی صفت A_2 در نرمال فرم اول قرار بگیرد.

XID	X_1	A_1	A_2	BXID

جدول غیرنرمال A

توجه: یکی از روش‌های حذف خصیصه‌های چند مقداری از یک جدول غیرنرمال و تبدیل آن به جداول نرمال به اینصورت است که جدول غیرنرمال به دو جدول تجزیه شود و کلید کاندید جدول غیرنرمال به عنوان کلید خارجی در جدول دوم دیگری تعریف شود و صفت چند مقداری نیز در آن درج شود، به صورت زیر:

مدل طراحی نرمال:

XID	A_2

جدول نرمال AA_2

XID	X_1	A_1	BXID

جدول نرمال A

توجه: در حال حاضر جدول A در نرمال فرم اول قرار دارد و همچنین جدول AA_2 در نرمال فرم اول قرار دارد.

توجه: کلید کاندید جدول A صفت (XID) است.

توجه: کلید کاندید جدول AA_2 صفات (XID , A2) یعنی تمام کلید است.

توجه: تجزیه در نرمال‌سازی یعنی **دوری** و **دوستی**، دور هستن یعنی دو جدول شدن، اما دوست هم هستن یعنی توسط تعریف کلید خارجی الحاق پذیر هم هستن و توسط عمل الحاق همان جدول پایه ایجاد می‌گردد.

توجه: همانطور که واضح است، طراحی جدول A در گزینه چهارم به صورت $A(\underline{XID}, X_1, A_1, BXID)$ در نظر گرفته شده است، همچنین طراحی جدول AA_2 در گزینه چهارم به صورت $AA_2(\underline{XID, A_2})$ در نظر گرفته شده است که مطابق آنچه بیان کردیم، طراحی درستی است. بنابراین پُرواضح است که گزینه چهارم پاسخ سوال است. شاید این سوال در ذهن شما

شـکل بگیرد که چرا در گزینه‌ی چهارم، جدول A به صـورت A(XID , A_1, BXID) آمده اسـت و ستون X_1 جا افتاده است! پاسخ این است که گزینه چهارم در صورت سوال اصلی سنجش خطای حروفچینی داشته است و صحیح آن همان A(XID , X_1, A_1 , BXID) است.

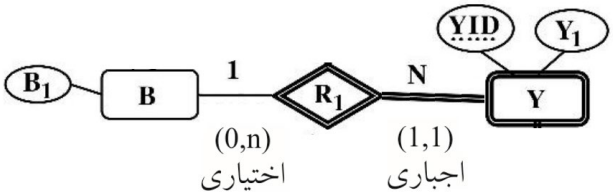
وابستگی وجودی

اگر در یک بانک اطلاعاتی، وجود یک موجودیت، وابسـته به موجودیت دیگری باشـد که در صـورت حذف و تغییر موجودیت اصـلی یعنی موجودیت قوی این موجودیت نیز تغییر کند، این نوع وابسـتگی را وابسـتگی وجودی گفته و به پدیده وابسـته، **موجودیت ضـعیف** گویند. همچنین **موجودیت ضـعیف** کلید **موجودیت قوی** را در بر دارد تا هرگونه تغییر یا حذف در موجودیت قوی به موجودیت ضعیف اعمال شود.

توجه: موجودیت ضعیف با دو مستطیل تو در تو نمایش داده می‌شود.

بخشی از مدل EER رسم شده در صورت سوال، یک رابطه یک به چند بین دو موجودیت قوی و ضعیف را نشان می‌دهد. که در ادامه فرآیند نگاشت آن به مدل رابطه‌ای را شرح می‌دهیم.

مدل تحلیل:



همانطور که در مدل طراحی قبل‌تر گفتیم، مدل طراحی جدول B به صـورت B(XID , X_1 , B_1) است.

XID	X_1	B_1

جدول B

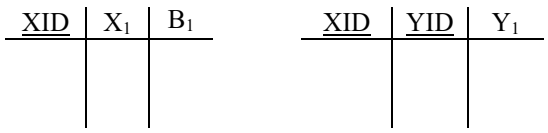
توجه: بنابراین صـفت XID **کلید موجودیت قوی** B و صـفت YID، **صـفت ممیزه موجودیت ضعیف** Y است. صفت ممیزه موجودیت ضعیف توسط نقطه چین نشان داده می‌شود.

توجه: نماد خط مضاعف افقی نشانه اجباری بودن موجودیت چسبیده به آن است.

توجه: نماد خط افقی نشانه اختیاری بودن موجودیت چسبیده به آن است.

توجه: قید (0,N) نشـان می‌دهد که هر نمونه موجودیت از B حداقل با صـفر و حداکثر با N نمونه موجودیت از Y ارتباط دارد و قید (1,1) نشـان می‌دهد که هر نمونه موجودیت از Y حداقل با یک و حداکثر با یک نمونه موجودیت از B ارتباط دارد.

مدل طراحی



جدول B *جدول Y*

توجه: کلید کاندید جدول یک یعنی **موجودیت قوی** در جدول چند یعنی **موجودیت ضـــعیف** به عنوان **کلید خارجی** تعریف می‌گردد.

توجه: کلید کاندید جدول چند یعنی **موجودیت ضعیف** برابر ترکیب **کلید خارجی** و **صفت ممیزه** در جدول موجودیت ضعیف است. یعنی کلید کاندید جدول Y برابر (XID,YID) است.

توجه: **صفت ممیزه** یا **کلید جزئی** به طور سراسری در یک موجودیت ضعیف یکتا نیست، بلکه فقط در بین نمونه‌ها یا دسته‌هایی که با موجودیت قوی ارتباط دارند، یکتا است.

توجه: یک موجودیت ضعیف **همیشه** در ارتباطش با موجودیت قوی رابطه **اجباری** دارد.

۸۴- با فرض وجود رابطه‌های S و P و J و SPJ به صورت زیر:
S(SID,SNAME,STATUS,CITY)- رابطه تولیدکنندگان شامل شماره تولیدکننده، نام تولیدکننده، وضعیت، شهر
P(PID,PNAME,WEIGHT)- رابطه انواع قطعات شامل شماره قطعه، نام نوع قطعه، وزن
J(JID,JNAME,CITY)- رابطه پروژه‌ها شامل شماره پروژه، نام پروژه، شهر
SPJ(SID,PID,JID,QTY)- رابطه قطعات تولیدشده شامل تعداد قطعه (QTY) تولیدشده از نوع PID توسط تولیدکننده SID در پروژه JID
کدام گزینه، جبر رابطه‌ای معادل شماره تولیدکنندگانی است که در همه پروژه‌ها قطعه شماره P100 را تولید کرده‌اند؟ (لازم به ذکر است * عملگر الحاق طبیعی (Natural Join) است.)
(مهندسی *IT*-دولتی۱۴۰۱)

۱) $\Pi_{<SID>}\left(\sigma_{SPJ.PID=P100}\left(S*SPJ*J\right)\right)$

۲) $\Pi_{<SID,JID>}\left(\sigma_{SPJ.PID\ =\ P100}\left(SPJ\right)\right)\div\Pi_{<JID>}\left(J\right)$

۳) $\Pi_{<SID,PID>}\left(SPJ\right)\div\Pi_{<PID>}\left(\sigma_{SPJ.PID=P100}\left(P\right)\right)$

۴) $\Pi_{<SID>}\left(S\right)-\Pi_{<SID>}\left(\sigma_{SPJ.PID\ =P100}\ \left(SPJ\right)*J\right)$

عنوان کتاب: پایگاه داده‌ها

مولف: استاد خلیلی‌فر

ناشر: انتشارات راهیان ارشد و دکتری

آدرس سایت گروه بابان: khalilifar.ir

پاسخ تست‌های فصل چهارم: جبر رابطه‌ای

۸۴- گزینه (۲) صحیح است.

جداول زیر را در نظر بگیرید:

SID	SNAME	STATUS	CITY
S1	Sn1	1	C1
S2	Sn2	2	C2
S3	Sn3	3	C3
S4	Sn4	4	C4

جدول تولیدکننده S

SID	PID	JID	QTY
S1	P100	J1	5
S1	P100	J2	10
S1	P100	J3	15
S2	P100	J1	20
S2	P100	J2	25
S2	P100	J3	30
S2	P200	J1	35
S2	P200	J2	40
S2	P200	J3	45
S3	P100	J1	50
S3	P100	J2	55
S3	P200	J1	60
S3	P200	J2	65

جدول SPJ

JID	JNAME	CITY
J1	Jn1	C1
J2	Jn2	C2
J3	Jn3	C3

جدول پروژه J

PID	PNAME	WEIGHT
P100	Pn1	1000
P200	Pn2	2000

جدول قطعه P

«کدام گزینه، جبر رابطه‌ای معادل شماره تولیدکنندگانی است که در **همه** پروژه‌ها قطعه شماره P100 را تولید کرده‌اند؟»

پرس‌وجوی فوق توسط عملگر تقسیم در جبر رابطه‌ای قابل پیاده‌سازی است.

مطابق تعریف عملگر تقسیم در جبر رابطه‌ای، تمام خصیصه‌های مقسوم علیه باید در مقسوم باشد. به عبارت دیگر مجموعه صفات مقسوم علیه، زیر مجموعه، مجموعه صفات مقسوم باشد. همچنین ستون‌های حاصل از عملگر تقسیم برابر تفاضل ستون‌های مقسوم و مقسوم علیه است. همچنین سطرهایی از جدول r در خروجی ظاهر می‌گردد که تمام اعضای جدول s را در کنار خود داشته باشد.

مثال:

a	b
5	7
5	1
4	2
4	3
5	2

جدول s

$$\div \frac{b}{1} = \frac{a}{5}$$

2

جدول r

جدول SPJ که می‌گوید چه شماره تولیدکننده‌ای، چه شماره قطعه‌ای، در چه شماره پروژه‌ای، تولید کرده‌است، مقسوم است و جدول J با شرط «همه پروژه‌ها» مقسوم علیه است.

گزینه دوم درست است، زیرا شماره تولیدکنندگانی که قطعه شماره P100 را در همه پروژه‌ها تولید کرده‌اند در خروجی قرار می‌دهد.

$$\Pi_{<SID,JID>}\left(\sigma_{SPJ.PID\,=\,P100}\left(SPJ\right)\right) \div \Pi_{<JID>}\left(J\right)$$

خروجی پرس‌وجوی داخلی در سمت چپ عملگر تقسیم به صورت زیر است:

$$\sigma_{SPJ.PID\,=\,P100}{}^{(SPJ)}$$

SID	PID	JID	QTY
S1	P100	J1	5
S1	P100	J2	10
S1	P100	J3	15
S2	P100	J1	20
S2	P100	J2	25
S2	P100	J3	30
S3	P100	J1	50
S3	P100	J2	55

خروجی پرس‌وجوی تقسیم به صورت زیر است:

$$\left[\Pi_{<SID,JID>}\left(\sigma_{SPJ.PID\,=\,P100}{}^{(SPJ)}\right) \div \Pi_{<JID>}{}^{(J)}\right]$$

SID	JID
S1	J1
S1	J2
S1	J3
S2	J1
S2	J2
S2	J3
S3	J1
S3	J2

÷

JID
J1
J2
J3

SID
S1
S2

توجه: مشاهده می‌شود که تولیدکنندگان شماره S1 و S2، قطعه شماره P100 را در **همه** پروژه‌ها تولید کرده‌اند.

گزینه اول نادرست است، زیرا شماره تولیدکنندگانی که حداقل قطعه شماره P100 را در حداقل یک پروژه تولید کرده‌اند در خروجی قرار می‌دهد.

$$\Pi_{<SID>}\left(\sigma_{SPJ.PID=P100}\left(S*SPJ*J\right)\right)$$

خروجی پرس‌وجوی داخلی به صورت زیر است:

$$\sigma_{SPJ.PID=P100}\left(S*SPJ*J\right)$$

SID	PID	JID	...
S1	P100	J1	
S1	P100	J2	
S1	P100	J3	
S2	P100	J1	
S2	P100	J2	
S2	P100	J3	
S3	P100	J1	
S3	P100	J2	

خروجی نهایی پرس‌وجو به صورت زیر است:

SID
S1
S2
S3

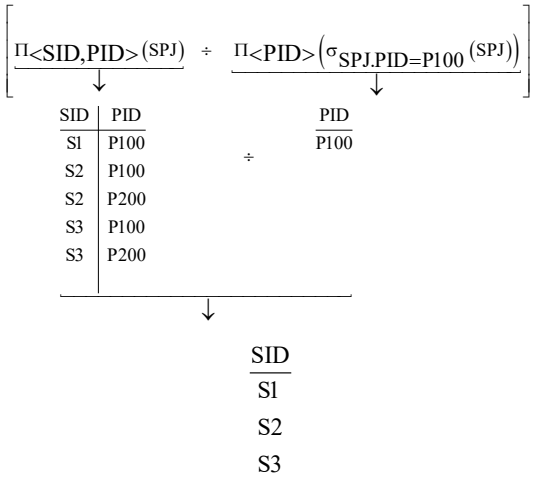
گزینه سوم نادرست است، زیرا خطای نحوی دارد.

$$\Pi_{<SID,PID>}\left(SPJ\right) \div \Pi_{<PID>}\left(\sigma_{SPJ.PID=P100}\left(P\right)\right)$$

اگر فرم اصلاح شده را به صورت زیر در نظر بگیریم، آنگاه شماره تولیدکنندگانی که حداقل قطعه شماره P100 را در حداقل یک پروژه تولید کرده‌اند در خروجی قرار می‌دهد.

$$\Pi_{<SID,PID>}\left(SPJ\right) \div \Pi_{<PID>}\left(\sigma_{SPJ.PID=P100}\left(SPJ\right)\right)$$

خروجی پرس‌وجوی تقسیم به صورت زیر است:

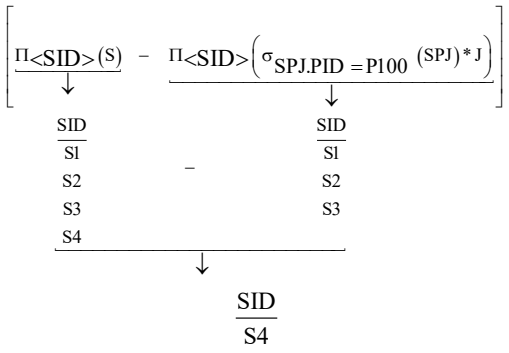


گزینه چهارم نادرست است، زیرا شماره تولیدکنندگانی که قطعه شماره P100 را در هیچ پروژه‌ای تولید نکرده‌اند در خروجی قرار می‌دهد.

$$\Pi_{<SID>}(S) - \Pi_{<SID>}\left(\sigma_{SPJ.PID=P100}\ (SPJ) * J\right)$$

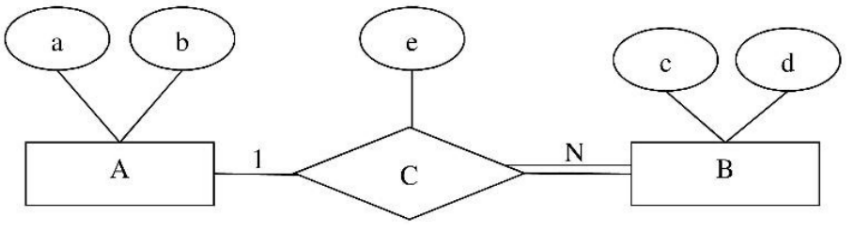
پرس‌وجوی سمت راست عملگر تفاضل، شماره تولیدکنندگانی که حداقل قطعه شماره P100 را در حداقل یک پروژه تولید کرده‌اند در خروجی قرار می‌دهد.

خروجی پرس‌وجوی تفاضل به صورت زیر است:



۸۵ – با توجه به نمودار ER زیر، کدام مورد، درست است؟

(مهندسی *IT*–دولتی۱۴۰۱)



۱) صفت e را می‌توان به موجودیت A بدون از دست دادن اطلاعات منتقل کرد.

۲) صفت e را می‌توان به هر کدام از دو موجودیت A یا B بدون از دست دادن اطلاعات منتقل کرد.

۳) صفت e را می‌توان به موجودیت B بدون از دست دادن اطلاعات منتقل کرد.

۴) صفت e را نمی‌توان بدون از دست دادن اطلاعات به هیچ کدام از موجودیت‌ها منتقل کرد.

عنوان کتاب: پایگاه داده‌ها

مولف: استاد خلیلی‌فر

ناشر: انتشارات راهیان ارشد و دکتری

آدرس سایت گروه بابان: khalilifar.ir

پاسخ تست‌های فصل سوم: مدل رابطه‌ای

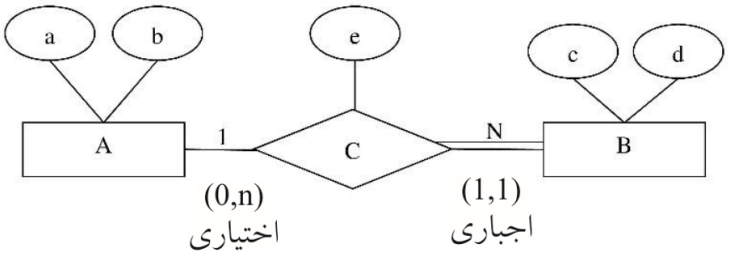
۸۵ - گزینه (۳) صحیح است.

نگاشت رابطه یک به چند بین دو موجودیت به مدل رابطه‌ای

مستقل از اختیاری یا اجباری بودن موجودیت‌ها، هر موجودیت به یک جدول تبدیل می‌گردد. و کلید کاندید جدول یک در جدول چند به عنوان کلید خارجی تعریف می‌گردد. همچنین صفات متصل به **رابطه**، درون جدول چند **مستتر** می‌شود.

روال نگاشت در حالت اجباری و اختیاری به صورت زیر است:

مدل تحلیل:



توجه: در شکل صورت سوال صفت کلید موجودیت A و B مشخص نشده است. فرض کنیم صفت a کلید موجودیت A و صفت c کلید موجودیت B است. البته این فرض در پاسخ نهایی سوال اثری ندارد.

توجه: نماد خط مضاعف افقی نشانه اجباری بودن موجودیت چسبیده به آن است.

توجه: نماد خط افقی نشانه اختیاری بودن موجودیت چسبیده به آن است.

توجه: قید (0,N) نشان می‌دهد که هر نمونه موجودیت از A حداقل با صفر و حداکثر با N نمونه موجودیت از B ارتباط دارد و قید (1,1) نشان می‌دهد که هر نمونه موجودیت از B حداقل با یک و حداکثر با یک نمونه موجودیت از A ارتباط دارد.

مدل طراحی:

a	b

جدول A

c	d	**a**	e

جدول B

توجه: کلید کاندید جدول یک یعنی **موجودیت A** در جدول چند یعنی **موجودیت B** به عنوان

کلید خارجی تعریف می‌گردد.

توجه: همچنین صفات متصل (e) به **رابطه**، درون جدول چند یعنی موجودیت B **مستتر** می‌شود.

توجه: کلید کاندید جدول یک یعنی **موجودیت A** برابر همان **کلید کاندید سابق** در جدول موجودیت A است. یعنی کلید کاندید جدول A برابر (a) است.

توجه: کلید کاندید جدول چند یعنی **موجودیت B** برابر همان **کلید کاندید سابق** در جدول موجودیت B است. یعنی کلید کاندید جدول B برابر (c) است.

توجه: همانطور که واضح است، طراحی جدول B در گزینه سوم به صورت B(c,d,a,e) در نظر گرفته شده است، که مطابق آنچه بیان کردیم، طراحی درستی است. یعنی «صفت e را می‌توان به موجودیت B بدون از دست دادن اطلاعات منتقل کرد.»، بنابراین پُرواضح است که گزینه سوم پاسخ سوال است.

تست‌های فصل ششم: زبان SQL دستورات DML

۸۶ – جداول زیر را در نظر گرفته و سپس بگویید کدام گزینه توصیف کوئری زیر است؟

(مهندسی *IT*–دولتی۱۴۰۱)

((آدرس) Address و (نام شخص) Name و (کد ملی) SSN) Person (شخص)

((مدل)Model و (سال) Year و (شماره پلاک) License) Car (ماشین)

((میزان خسارت) Damage-Amount و License , Accident-date) Accident (تصادف)

(مالک) Owner (SSN , License)

```
Select P1.Name , P2.Name
From Person P1, Person P2
Where ((Select Count (Damage-Amount)
        From Accident Natural Join Owner
        Where P1.SSN = SSn) = (Select Count (Damage-Amount)
                               From Accident Natural Join Owner
                               Where P2.SSN = SSN))
```

۱) نام زوج افرادی که تعداد تصادف‌های یکسان داشته‌اند.

۲) نام زوج افرادی که میزان خسارت تصادف‌های آنها یکسان بوده است.

۳) این پرس‌وجو از لحاظ نوشتاری یا منطقی خطا دارد.

۴) نام زوج افرادی که ماشین آنها با هم تصادف کرده است.

عنوان کتاب: پایگاه داده‌ها

مولف: استاد خلیلی‌فر

ناشر: انتشارات راهیان ارشد و دکتری

آدرس سایت گروه بابان: khalilifar.ir

پاسخ تست‌های فصل ششم: زبان SQL دستورات DML

۸۶ - گزینه (۱و۳) صحیح است.

چهار جدول Person، Car، Accident و Owner با مقادیر زیر را در نظر بگیرید:

SSN	Name	Address
Ssn1	Sn1	Add1
Ssn2	Sn2	Add2
Ssn3	Sn3	Add3

جدول Person

License	Year	Model

جدول Car

License	accident-date	damage-amount
L1	95-01-1	10
L1	95-01-2	20
L2	95-02-1	30
L2	95-02-2	40
L3	95-03-1	50

جدول Accident

SSN	License
Ssn1	L1
Ssn2	L2
Ssn3	L3

جدول Owner

توجه: دقت کنید که مطابق فرض سوال، ترکیب دو ستون License و accident-date هردو باهم به عنوان کلید کاندید جدول Accident مشخص شده است، بنابراین نباید ترکیب این دو ستون باهم سطرهای تکراری داشته باشد، پس یک خودرو (License) در یک تاریخ تصادف (accident-date) نمی‌تواند بیش از یکبار تکرار شود، یعنی نباید دوبار و در دو سطر و بیشتر تکرار شود.

پرس‌وجوی مطرح شده در صورت سوال به صورت زیر است:

```
Select P1.Name , P2.Name
From Person P1, Person P2
Where ((Select Count (Damage-Amount)
        From Accident Natural Join Owner
        Where P1.SSN = SSN) = (Select Count (Damage-Amount)
                               From Accident Natural Join Owner
                               Where P2.SSN = SSN))
```

با توجه به جداول فوق، خروجی جلوی عملگر From پس از انجام عملگر ضـــرب دکارتی به صورت زیر است:

P1.SSN	P1.Name	P1.Address	P2.SSN	P2.Name	P2.Address
Ssn1	Sn1	Add1	Ssn1	Sn1	Add1
Ssn1	Sn1	Add1	Ssn2	Sn2	Add2
Ssn1	Sn1	Add1	Ssn3	Sn3	Add3
Ssn2	Sn2	Add2	Ssn1	Sn1	Add1
Ssn2	Sn2	Add2	Ssn2	Sn2	Add2
Ssn2	Sn2	Add2	Ssn3	Sn3	Add3
Ssn3	Sn3	Add3	Ssn1	Sn1	Add1
Ssn3	Sn3	Add3	Ssn2	Sn2	Add2
Ssn3	Sn3	Add3	Ssn3	Sn3	Add3

با توجه به وجود **عملگر تســـاوی** و به **شـــرط اتصـــال زیر پرس‌وجوی داخلی به پرس‌وجوی خارجی**، زیر پرس‌وجوی داخلی فوق یک Correlated Subquery اسـت، یعنی به ازای حرکت در هریک از ســـطرهای پرس و جوی خارجی، یک بار به طور کامل از ابتدا تا انتها زیر پرس‌وجوی داخلی اجرا و بر اساس شرطی که زیر پرس‌وجوی داخلی را به پرس‌وجوی خارجی متصل می‌کند، بررســی انجام می‌شــود. مانند دو حلقه تو در تو، که به ازای هربار اجرای حلقه خارجی، یک بار به طور کامل حلقه داخلی اجرا می‌گردد.

زیر پرس‌وجوی (Select Count (Damage-Amount با شـرط اتصـال Where **P1**.SSN = SSN به پرس‌وجوی خارجی متصل شده‌است.

زیر پرس‌وجوی (Select Count (Damage-Amount با شـرط اتصـال Where **P2**.SSN = SSN به پرس‌وجوی خارجی متصل شده‌است.

توسط دستور where در فرم زیر:

```
Select P1.Name , P2.Name
From Person P1, Person P2
Where ((…) = (…))
```

برای هر ســطر از جدول حاصــل از ضــرب دکارتی Person P1, Person P2 به طور همزمان شــرط Where P1.SSN = SSN (شـــرط اتصـــال بـــه جـــدول Person P1, Person P2) و شـــرط Where P2.SSN = SSN (شـرط اتصـال به جدول Person P1, Person P2) بررســی می‌شــود. اگر به طور همزمان تســـاوی Select Count (Damage-Amount) = Select Count (Damage-Amount) برقرار

بود، آنگاه سطر مورد نظر از جدول حاصل از ضرب دکارتی Person P1, Person P2 انتخاب می‌گردد و این رویه برای تک تک سطرهای جدول حاصل از ضرب دکارتی Person P1, Person P2 ، تا به انتهای جدول ادامه پیدا می‌کند.

برای سطر اول تا آخر از جدول Person P1, Person P2 با توجه به شرط اتصال داریم:

P1.SSN	P1.Name	P1.Address	**P2.SSN**	P2.Name	P2.Address
➢ **Ssn1**	Sn1	Add1	**Ssn1**	Sn1	Add1
Ssn1	Sn1	Add1	Ssn2	Sn2	Add2
Ssn1	Sn1	Add1	Ssn3	Sn3	Add3
Ssn2	Sn2	Add2	Ssn1	Sn1	Add1
Ssn2	Sn2	Add2	Ssn2	Sn2	Add2
Ssn2	Sn2	Add2	Ssn3	Sn3	Add3
Ssn3	Sn3	Add3	Ssn1	Sn1	Add1
Ssn3	Sn3	Add3	Ssn2	Sn2	Add2
Ssn3	Sn3	Add3	Ssn3	Sn3	Add3

با توجه به جداول فوق، خروجی دستور Select Count (Damage-Amount) با شرط اتصال Where P1.~~SSN~~ = SSN Ssn1 به صورت زیر است:

Select Count (Damage-Amount)
From Accident **Natural Join** Owner
Where **P1**.SSN = SSN

License	accident-date	damage-amount	SSN
➢ L1	95-01-1	10	**Ssn1**
➢ L1	95-01-2	20	**Ssn1**
L2	95-02-1	30	Ssn2
L2	95-02-2	40	Ssn2
L3	95-03-1	50	Ssn3

با توجه به جداول فوق، خروجی دستور Select Count (Damage-Amount) با شرط اتصال Where P2.~~SSN~~ = SSN Ssn1 به صورت زیر است:

Select Count (Damage-Amount)
From Accident **Natural Join** Owner
Where **P2**.SSN = SSN

License	accident-date	damage-amount	SSN
➢ L1	95-01-1	10	**Ssn1**
➢ L1	95-01-2	20	**Ssn1**
L2	95-02-1	30	Ssn2
L2	95-02-2	40	Ssn2
L3	95-03-1	50	Ssn3

مطابق پرس‌وجوی خارجی مقدار **Ssn1** از سطر اول جدول حاصل از ضرب دکارتی Person P1, Person P2 برداشته می‌شود و داخل دو جدول زیرپرس‌وجوی **داخلی** مقایسه می‌شود. که به ازای آن 2 تصادف برای سمت P1 و 2 تصادف برای سمت P2 وجود دارد. مقدار 2 در هر دو طرف تساوی برابر است، پس سطر اول حاصل از ضرب دکارتی Person P1, Person P2 در خروجی قرار می‌گیرد. این رویه برای تک تک سطرهای جدول تا به انتهای جدول ادامه پیدا می‌کند.

توجه: عملگر Natural join در برخی نسخه‌های SQL پشتیبانی نمی‌شود. و به جای آن عملگرهای Jion و Inner Join البته با عملکرد تا حدی مشابه پشتیبانی می‌شود. بنابراین گزینه سوم هم می‌تواند پاسخ سوال باشد.

بنابراین پرس‌وجوی مطرح شده در صورت سوال ممکن است در برخی کامپایلرهای SQL خطای نحوی داشته باشد، که فرم اصلاح شده آن به صورت زیر است:

```
Select P1.Name , P2.Name
From Person P1 , Person P2
Where ((Select Count (Damage-Amount)
        From Accident Join Owner on Accident.License = Owner.License
        Where P1.SSN = SSn) = Select Count (Damage-Amount)
                              From Accident Join Owner
                              on Accident.License = Owner.License
                              Where P2.SSN = SSN))
```

خروجی نهایی یعنی «نام زوج افرادی که تعداد تصادف‌های یکسان داشته‌اند.» به صورت زیر است:

P1.Name	P2.Name
Sn1	Sn1
Sn1	Sn2
Sn2	Sn1
Sn2	Sn2
Sn3	Sn3

توجه: سازمان سنجش آموزش کشور، در کلید اولیه خود، گزینه اول را به عنوان پاسخ اعلام کرده بود. اما در کلید نهایی نظر خود را عوض کرد و گزینه اول و سوم را به عنوان پاسخ اعلام کرد، که کار درستی بوده است.