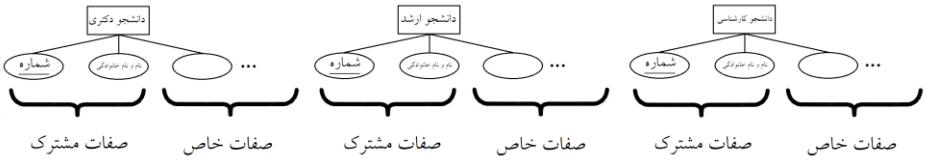
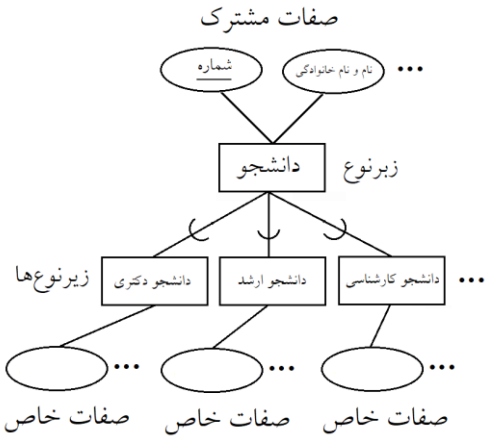
تعمیم (Generalization) و تخصیص (Specification)

به فرآیند انتقال **صفات و شناسه مشترک** بین موجودیت‌های هم خانواده به سمت بالا **تعمیم** گفته می‌شود و به فرآیند انتقال **صفات متفاوت** بین موجودیت‌های هم خانواده به سمت پایین **تخصیص** گفته می‌شود. این مدل‌سازی به این صورت تفسیر معنایی می‌شود که، اطلاعات یک رکورد اطلاعاتی به دو تکه اطلاعات عمومی و اطلاعات اختصاصی تقسیم می‌شود. اطلاعات عمومی در بخش تعمیم (پدر) نگهداری می‌شود و اطلاعات اختصاصی در بخش تخصیص (فرزند) نگهداری می‌شود. در نهایت دو سطر درج شده در دو جدول پدر و فرزند بهم مرتبط می‌شوند. تعمیم و تخصیص را با یک درخت به نام درخت تعمیم و تخصیص نمایش می‌دهند. اگر d تعداد سطوح این درخت باشد، داریم: $d \geq 2$. تعمیم و تخصیص به دو نوع کامل و ناقص به شکل رابطه پوشا (Overlap) و رابطه غیرپوشا (Disjoint) است که در ادامه مورد بررسی قرار می‌گیرد.

توجه: لازمه تعمیم و تخصیص این است که حداقل دو نوع موجودیت از پیش دیده شده داشته باشیم که حداقل در صفت شناسه، اشتراک دارند.

مثال: نوع موجودیت‌های دانشجوی کارشناسی، دانشجوی ارشد و دانشجوی دکتری را در نظر می‌گیریم. این نوع موجودیت‌ها، مجموعه‌ای از صفات مشترک دارند، مثل شماره دانشجو، نام و نام خانوادگی، سال ورود و ... و در عین حال، هریک از این انواع می‌توانند صفات خاص خود را هم داشته باشد. یک زبرنوع (ابرنوع) به نام نوع موجودیت دانشجو در نظر می‌گیریم و مجموعه صفات مشترک بین آن نوع موجودیت‌ها را به این زبرنوع می‌دهیم. به این ترتیب عمل تعمیم انجام داده‌ایم. آن نوع موجودیت‌ها اینک هریک زیرنوع این زبرنوع می‌شوند، با صفات خاص خود. به این ترتیب عمل تخصیص انجام داده‌ایم.





دو رویکرد برای ایجاد سلسله مراتب وراثت وجود دارد:

۱- رویکرد «بالا به پایین» یا تخصیص (**Specialization**) که در آن، ابتدا کلاس‌های پدر، تعریف و سپس کلاس‌های فرزند ایجاد می‌شوند.

۲- رویکرد «پایین به بالا» یا تعمیم (**Generalization**) که در آن، از طریق تعمیم کلاس‌های فرزند به کلاس‌های پدر می‌رسیم.

توجه: رابطه وراثت به عنوان رابطه «**is a**» یا «**kind of**» یا «**type of**» نیز شناخته می‌شود. زیرا در این رابطه می‌گوییم: «**A red apple is an apple**» و یا «**An apple is a fruit**».

مراحل لازم جهت مدل‌سازی یک رابطه ارث‌بری عبارتند از:

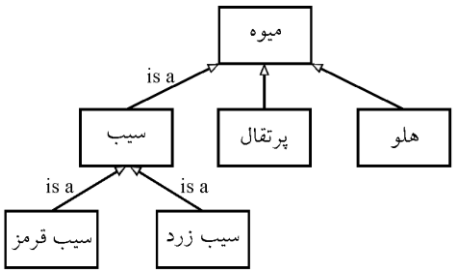
۱- ترسیم یک خط ممتد بین کلاس فرزند و کلاس پدر

۲- درج یک مثلث تو خالی در انتهایی از خط ممتد که کلاس پدر قرار دارد.

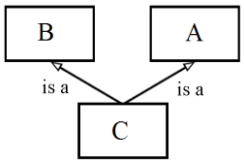
توجه: ذکر «**is a**» بر روی خط ممتد رابطه وراثت اختیاری است.

توجه: در تعریف رابطه وراثت، مشخصاتی نظیر تعدد و محدودیت وجود ندارد.

شکل مقابل نمونه‌ای از یک سلسله مراتب ارث‌بری میان «میوه»، «سیب» و «سیب قرمز»، را نشان می‌دهد.



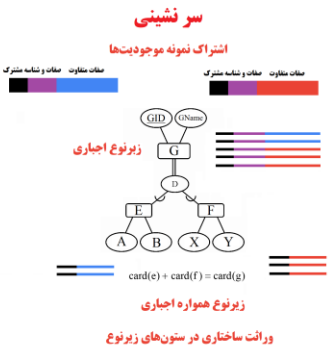
توجه: همانطورکه در فصل قبل نیز گفتیم، هرگاه یک کلاس برخی از صفات و عملیات را از یک کلاس و برخی دیگر را از یک کلاس دیگر به ارث ببرد، در این حالت وراثت چندگانه رخ داده‌است.

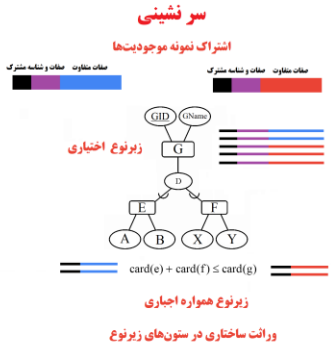


شـکل مقابـل نمونـه‌ای از مـدل‌سـازی «ارث‌بـری چندگانـه» **(Multiple Inheritance)** را نشان می‌دهد.

در شکل فوق کلاس **C** به صورت وراثت چندگانه از کلاس **A** و کلاس **B** ارث‌بری کرده است.
پس از مدل‌سازی ارتباطات ایستای میان کلاس‌های همکار نوبت به مدل‌سازی تعاملات پویای میـان اشـیاء همکار می‌رسد.

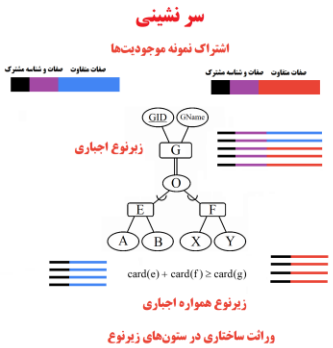
گروه بابان
BABAN.IR

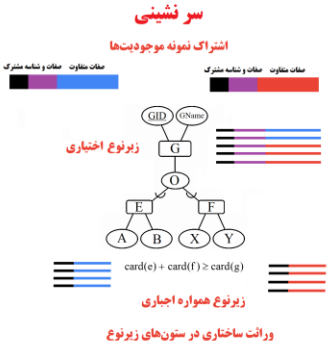




BABAN.IR

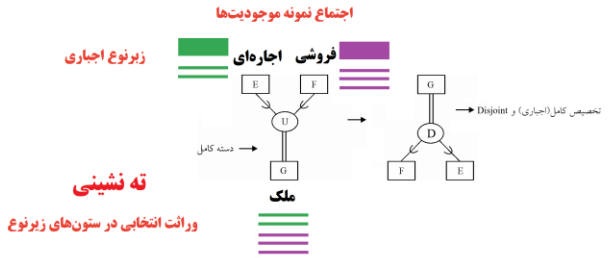
گروه بابان
BABAN.IR

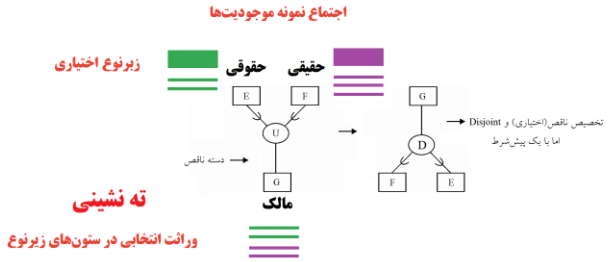




BABAN.IR

گروه بابان
BABAN.IR

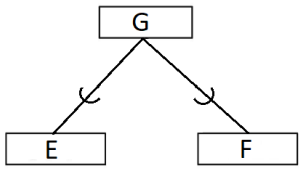




BABAN.IR

روش عمومی نگاشت رابطه ISA یا وراثت به مدل رابطه‌ای

اگر در مسائل وراثت، Disjoint و Overlap روی مدل لحاظ نشود، نگاشت به مدل رابطه‌ای از **روش عمومی** انجام می‌شود.



در این حالت به n+1 جدول نیاز داریم. اگر ابرنوع n زیرنوع داشته باشد یک جدول برای ابرنوع در نظر می‌گیریم و برای هر یک از زیرنوع‌ها هم یک جدول با صفت‌های خاص هر زیرنوع در نظر گرفته می‌شود. همچنین کلید کاندید جدول پدر(ابرنوع یا زبرنوع) در جداول فرزند (زیرنوع) به عنوان کلید خارجی تعریف می‌گردد و همزمان در جداول فرزند کلید کاندید هم می‌باشد. و یک طراحی به شکل مدل n+1 جدولی اما با تحمل سربار حاصل از عمل الحاق مابین جدول زبرنوع و زیرنوع ایجاد می‌گردد.

مثال: نوع موجودیت دانشجو و گونه‌های خاص آن: دانشجوی «کارشناسی» ، دانشجوی «ارشـد» و دانشجوی «دکتری» را در نظر می‌گیریم. چهار رابطه طراحی می‌کنیم به صورت زیر:

STUD (STID , STNAME , ...)
C.K.

صفات مشترک در همه نمونه‌های نوع موجودیت دانشجو در عنوان رابطه STUD می‌آیند.

BSSTUD (STID , A , B , ...)
C.K. , F.K.

MSSTUD (STID , M , N , ...)
C.K. , F.K.

DOCSTUD (STID , X , Y , Z , ...)
C.K. , F.K.

می‌بینیم که در این روش هرگاه زبرنوع E دارای n زیرنوع E_i باشد، n+1 جدول طراحی می‌شود.
در هریک از سه جدول آخر صفات خاص دانشجویان دوره کارشناسی، ارشد و دکتری در نظر گرفته می‌شوند و دیگر لزومی ندارد که صفات مشترک بین این گونه‌ها، که در همان جدول STUD آورده می‌شوند، در جداول دیگر تکرار شوند، چون گونه‌های دانشجو این صفات را از نوع موجودیت دانشجو به ارث می‌برند. البته کلید کاندید STUD باید در سه جدول دیگر آورده شود.
این مثال **روش عمومی** نمایش مفاهیم زبرنوع (Supertype) و زیرنوع (Subtype) در مدل رابطه‌ای را نشان می‌دهد.
توجه: دقت داشته باشیم که در جداول نشان‌دهنده هر زیرنوع، کلید خارجی جدول، کلید کاندید آن هم هست.
سوال: نـوع موجودیت‌هـای F و G مفروضـند. می‌خـواهیم بـا اسـتفاده از تکنیـک **تعمـیم**، نـوع موجودیت E را بر اساس دو نوع موجودیت F و G ، در مدل‌سازی منظور کنیم. در اینصورت نوع موجودیت‌های F و G باید:
الف) حداقل یک صفت هیچ‌مقدار ناپذیر مشترک داشته باشند.
ب)حداقل شناسه مشترک داشته باشند.
ج) در دو صفت، که یکی از آنها شناسه باشد، مشترک باشند.
د) در صفت شناسه و یک صفت چندمقداری، مشترک باشند.

سوال: اگر N تعداد صفات مشترک بین n نوع موجودیت باشد، در تعمیم این n نـوع موجودیـت، یک شرط لازم کدام است؟

الف) $N \geq 2$

ب) $N \geq 1$

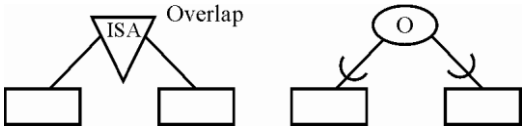
ج) $N > 1$

د) $N > 2$

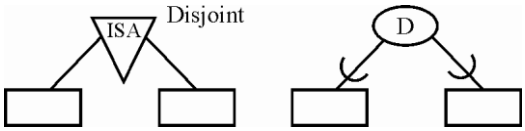
نگاشت رابطه ISA یا وراثت به مدل رابطه‌ای

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

رابطه پوشا یا تخصیص غیرمجزا (Overlap) مابین فرزندان و پدر به دو شیوه زیر نشان داده می‌شود:



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



توجه: نگاشت رابطه ISA یا وراثت به مدل رابطه‌ای به سه مدل انجام می‌شود که در ادامه بررسی می‌شود:

مدل تک جدولی

در این مدل همه ستون‌های موجودیت زبرنوع و زیرنوع در یک جدول قرار می‌گیرد.

مدل دو جدولی

در این مدل کل صفات موجودیت زبرنوع در جداول زیرنوع قرار داده می‌شود و یک طراحی به شکل مدل دو جدولی ایجاد می‌گردد. کلید کاندید هر جدول زیرنوع همان کلید کاندید زبرنوع است.

مدل سه جدولی

در این مدل به n+1 جدول نیاز داریم. اگر زبرنوع n زیرنوع داشته باشد یک جدول برای زبرنوع در نظر می‌گیریم و برای هر یک از زیرنوع‌ها هم یک جدول با صفت‌های خاص هر زیرنوع در نظر گرفته می‌شود. همچنین کلید کاندید جدول پدر(ابرنوع یا زبرنوع) در جداول فرزند (زیرنوع) به عنوان کلید خارجی تعریف می‌گردد و همزمان در جداول فرزند کلید کاندید هم می‌باشد.

توجه: در بررسی سربار جداول وجود مقادیر NULL غیرقابل تحمل‌ترین نوع سربار، سپس وجود سربار افزونگی طبیعی و نه افزونگی تکنیکی در جایگاه بعدی و در نهایت سربار الحاق طبیعی در جایگاه آخر به عنوان قابل تحمل‌ترین نوع سربار است.

توجه بسیار مهم: جهت ساده به خاطر سپردن مدل‌های مختلف نگاشت موارد زیر را در نظر بگیرید:

مورد اول: در بحث نگاشت وراثت به مدل رابطه‌ای افزایش مقادیر NULL از **فرزند نداشتن پدر** حاصل می‌شود و این حالت فقط و فقط زمانی ایجاد می‌شود که مدل تک جدولی و اختیاری باشد یا مدل دو جدولی و اختیاری باشد.

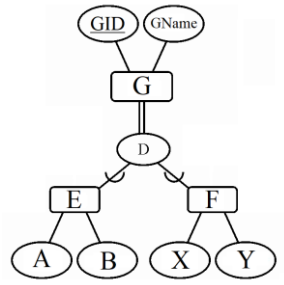
مورد دوم: در بحث نگاشت وراثت به مدل رابطه‌ای افزایش افزونگی از **تکرار رکورد پدر** حاصل می‌شود و این حالت فقط و فقط زمانی ایجاد می‌شود که مدل دو جدولی با حالت Overlap باشد.

مورد سوم: در بحث نگاشت وراثت به مدل رابطه‌ای سربار الحاق طبیعی از مدل سه جدولی حاصل می‌شود.

توجه: در یک عبارت ساده اختیاری بودن می‌تواند NULL ساز شود و Overlap بودن می‌تواند افزونگی‌ساز باشد.

الف) مدل تحلیل کامل (اجباری) و Disjoint

مدل تحلیل (نمودار ISA)



مدل طراحی (مدل رابطه‌ای)

مدل تک جدولی بهینه در شراط خاص: دقت کنید که در شرایط خاص یعنی زمانی‌که (1) تعداد صفات زبرنوع زیاد، (2) تعداد زیرنوع‌ها کم و (3) تعداد صفات زیرنوع‌ها کم باشد مدل تحلیل کامل (اجباری) و Disjoint امکان پیاده‌سازی تک جدولی دارد و بهینه هم هست.

سربار افزایش مقادیر NULL ندارد: چون ارتباط زبرنوع با زیرنوع کامل (اجباری) است، پس رکوردهای زبرنوع حداقل یک فرزند دارد و به تبع مقادیر NULL افزایشی نمی‌شود. همچنین تعداد صفات زیرنوع‌ها هم جهت NULL سازی کم است.

سربار افزایش افزونگی ندارد: چون در مدل تک جدولی یک رکورد خاص زبرنوع فقط در یک جدول درج می‌شود.

سربار الحاق طبیعی ندارد: چون در مدل تک جدولی الحاق میان جدول زبرنوع و زیرنوع وجود ندارد.

صفات مشترک		صفات خاص E		صفات خاص F	
GID	GName	A	B	X	Y
GID1	**GN1**	**A1**	**B1**	NULL	NULL
GID2	**GN2**	**A2**	**B2**	NULL	NULL
GID3	**GN3**	NULL	NULL	**X3**	**Y3**

جدول GEF

توجه: دقت کنید چون رابطه پدر با فرزندان از نوع کامل (اجباری) است، پس هر رکورد از پدر حداقل باید یک فرزند داشته باشد. برای مثال رکورد GID1,GN1 حداقل یک فرزند مثل A1,B1 از موجودیت E دارد.

توجه: دقت کنید چون رابطه فرزندان با پدر از نوع Disjoint است، پس هر رکورد از پدر فقط و فقط می‌تواند یک فرزند داشته باشد. برای مثال رکورد GID1,GN1 فقط و فقط می‌تواند یک فرزند مثل A1,B1 از موجودیت E یا از موجودیت F داشته باشد.

مدل دو جدولی بهینه: مدل تحلیل کامل (اجباری) و Disjoint امکان پیاده‌سازی دو جدولی دارد و بهینه هم هست.

سربار افزایش مقادیر NULL ندارد: چون ارتباط زبرنوع با زیرنوع کامل (اجباری) است، پس رکوردهای زبرنوع حداقل یک فرزند دارد و به تبع مقادیر NULL افزایشی نمی‌شود.

سربار افزایش افزونگی ندارد: چون ارتباط زیرنوع با زبرنوع Disjoint است، پس یک رکورد خاص زبرنوع G در دو جدول E و F تکرار نمی‌شود.

سربار الحاق طبیعی ندارد: چون در مدل دو جدولی الحاق میان جدول زبرنوع و زیرنوع وجود ندارد.

GID	GName	A	B
GID1	**GN1**	A1	B1

GID	GName	X	Y
GID3	**GN3**	X3	Y3

جدول F

GID2	GN2	A2	B2

جدول E

مدل سه جدولی غیربهینه به دلیل سربار الحاق طبیعی: مدل تحلیل کامل (اجباری) و Disjoint امکان پیاده‌سازی سه جدولی دارد اما بهینه نیست.

سربار افزایش مقادیر NULL ندارد: چون در مدل سه جدولی جداول زبرنوع و زیرنوع ازهم جدا هستن.

سربار افزایش افزونگی ندارد: چون در مدل سه جدولی جداول زبرنوع و زیرنوع ازهم جدا هستن.

سربار الحاق طبیعی دارد: چون در مدل سه جدولی الحاق میان جدول زبرنوع و زیرنوع وجود دارد.

جدول G

GID	GName
GID1	GN1
GID2	GN2
GID3	GN3

جدول E

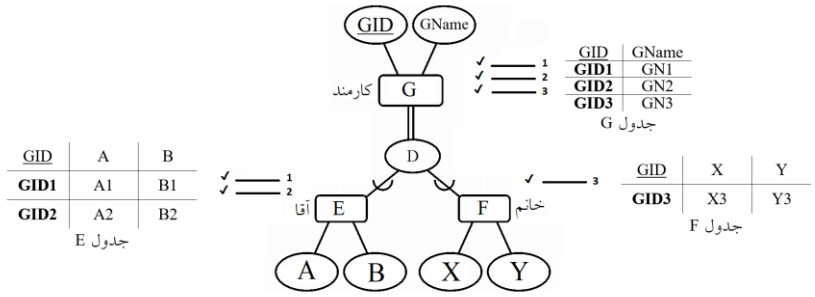
GID	A	B
GID1	A1	B1
GID2	A2	B2

جدول F

GID	X	Y
GID3	X3	Y3

توجه: دقت کنید که در رابطه‌های نشان دهنده هر زیرنوع، کلید خارجی جدول زیرنوع، کلید کاندید آن هم هست.

مدل شماتیک مدل تحلیل کامل (اجباری) و Disjoint به صورت زیر است:

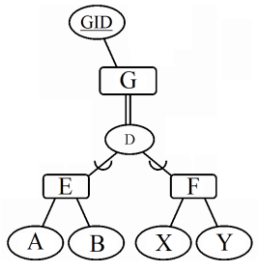


توجه: دقت کنید چون رابطه پدر با فرزندان از نوع کامل (اجباری) است، پس هر رکورد از پدر حداقل باید یک فرزند داشته باشد. برای مثال رکورد GID1,GN1 حداقل یک فرزند مثل A1,B1 از موجودیت E دارد.

توجه: دقت کنید چون رابطه فرزندان با پدر از نوع Disjoint است، پس هر رکورد از پدر فقط و فقط می‌تواند یک فرزند داشته باشد. برای مثال رکورد GID1,GN1 فقط و فقط می‌تواند یک فرزند مثل A1,B1 از موجودیت E یا از موجودیت F داشته باشد.

توجه: اگر e مجموعه نمونه‌های E ، f مجموعه نمونه‌های F و g مجموعه نمونه‌های G باشد، آنگاه $card(e)+card(f)=card(g)$ است. یعنی $2+1=3$

سوال: در نمودار زیر، نوع موجودیت G فقط یک صفت شناسه دارد. طراحی با چند رابطه (جدول) بهتر است انجام شود تا کمترین هیچ‌مقدار (NULL) و کمترین افزونگی پدید آید؟



الف) یک رابطه

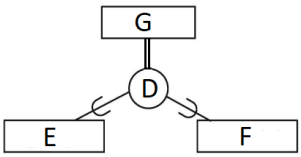
ب) دو رابطه

ج) سه رابطه

د) طراحی با یک یا دو رابطه باهم فرقی ندارند ولی از سه رابطه بهتر است.

پاسخ گزینه «ب» صحیح است.

سوال: با توجه به نمودار زیر :



اگر e مجموعه نمونه‌های E ، f مجموعه نمونه‌های F و g مجموعه نمونه‌های G باشد:

الف) $card(e)+card(f)\le card(g)$

ب) $card(e)+card(f)=card(g)$

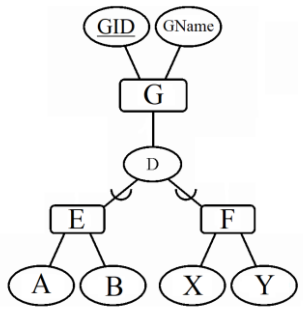
ج) $card(e)+card(f)\ne card(g)$

د) $card(e)+card(f)\ge card(g)$

پاسخ گزینه «ب» صحیح است.

ب) مدل تحلیل ناقص (اختیاری) و Disjoint

مدل تحلیل (نمودار ISA)



مدل طراحی (مدل رابطه‌ای)

مدل تک جدولی غیربهینه به دلیل افزایش مقدار NULL: دقت کنید که در شرایط خاص یعنی زمانی‌که (1) تعداد صفات زبرنوع زیاد، (2) تعداد زیرنوع‌ها کم و (3) تعداد صفات زیرنوع‌ها کم باشد مدل تحلیل ناقص (اختیاری) و Disjoint امکان پیاده‌سازی تک جدولی دارد اما بهینه نیست.

سربار افزایش مقدار NULL دارد: چون ارتباط زبرنوع با زیرنوع ناقص (اختیاری) است، پس ممکن است رکوردهای زبرنوع، فرزند نداشته باشد و به تبع مقادیر NULL افزایشی می‌شود. مانند سطر 2. هرچند تعداد صفات زیرنوع‌ها هم جهت NULL سازی کم است.

سربار افزایش افزونگی ندارد: چون در مدل تک جدولی یک رکورد خاص زبرنوع فقط در یک جدول درج می‌شود.

سربار الحاق طبیعی ندارد: چون در مدل تک جدولی الحاق میان جدول زبرنوع و زیرنوع وجود ندارد.

صفات مشترک		صفات خاص E		صفات خاص F	
GID	GName	A	B	X	Y
GID1	**GN1**	**A1**	**A1**	NULL	NULL
GID2	**GN2**	NULL	NULL	NULL	NULL
GID3	**GN3**	NULL	NULL	**X3**	**Y3**

جدول GEF

توجه: دقت کنید چون رابطه پدر با فرزندان از نوع ناقص (اختیاری) است، پس ممکن است رکوردهای پدر، فرزند نداشته باشد. برای مثال رکورد GID2,GN2 هیچ فرزندی ندارد.

توجه: دقت کنید چون رابطه فرزندان با پدر از نوع Disjoint است، پس هر رکورد از پدر فقط و فقط می‌تواند یک فرزند داشته باشد. برای مثال رکورد GID1,GN1 فقط و فقط می‌تواند یک فرزند مثل A1,B1 از موجودیت E یا از موجودیت F داشته باشد.

مدل دو جدولی غیربهینه به دلیل افزایش مقدار NULL: مدل تحلیل ناقص (اختیاری) و Disjoint امکان پیاده‌سازی دو جدولی دارد اما بهینه نیست.

سربار افزایش مقدار NULL دارد: چون ارتباط زیرنوع با زیرنوع ناقص (اختیاری) است، پس ممکن است زیرنوع، فرزند نداشته باشد و به تبع مقادیر NULL افزایشی می‌شود. مانند سطر 2 جدول E.

سربار افزایش افزونگی ندارد: چون ارتباط زیرنوع با زیرنوع Disjoint است، پس یک رکورد خاص زیرنوع G در دو جدول E و F تکرار نمی‌شود.

سربار الحاق طبیعی ندارد: چون در مدل دو جدولی الحاق میان جدول زیرنوع و زیرنوع وجود ندارد.

GID	GName	A	B
GID1	**GN1**	A1	B1
GID2	**GN2**	NULL	NULL

جدول E

GID	GName	X	Y
GID3	**GN3**	X3	Y3

جدول F

مدل سه جدولی بهینه با تحمل سربار الحاق طبیعی: مدل تحلیل ناقص (اختیاری) و Disjoint امکان پیاده‌سازی سه جدولی دارد و بهینه هم هست.

سربار افزایش مقدار NULL ندارد: چون در مدل سه جدولی جداول زیرنوع و زیرنوع ازهم جدا هستن.

سربار افزایش افزونگی ندارد: چون در مدل سه جدولی جداول زیرنوع و زیرنوع ازهم جدا هستن.

سربار الحاق طبیعی دارد: چون در مدل سه جدولی الحاق میان جدول زیرنوع و زیرنوع وجود دارد.

GID	GName
GID1	GN1
GID2	GN2
GID3	GN3

جدول G

GID	A	B

GID	X	Y

GID1	A1	B1

جدول E

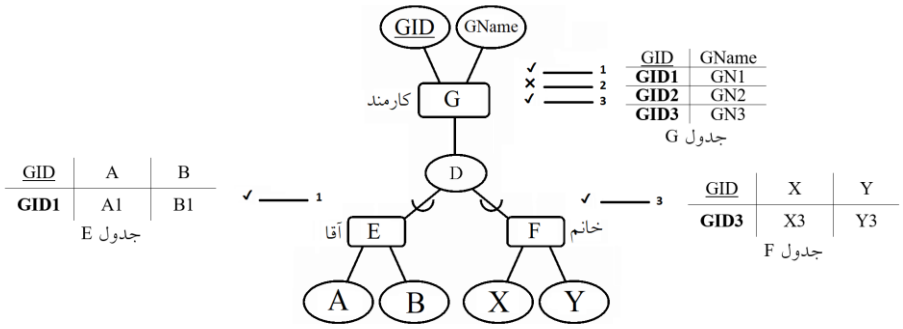
GID3	X3	Y3

جدول F

توجه: دقت کنید که مقدار GID2 در جدول E یا F درج نشده است چون فرض شده است مقادیر صفات خاص آن برابر مقدار NULL است. بنابراین مدل سه جدولی سربار افزایش مقادیر NULL را ندارد.

توجه: دقت کنید که در رابطه‌های نشان دهنده هر زیرنوع، کلید خارجی جدول زیرنوع، کلید کاندید آن هم هست.

مدل شماتیک مدل تحلیل ناقص (اختیاری) و Disjoint به صورت زیر است:



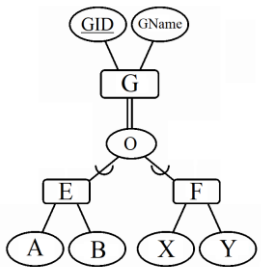
توجه: دقت کنید چون رابطه پدر با فرزندان از نوع ناقص (اختیاری) است، پس ممکن است رکوردهای پدر، فرزند نداشته باشد. برای مثال رکورد GID2,GN2 هیچ فرزندی ندارد.

توجه: دقت کنید چون رابطه فرزندان با پدر از نوع Disjoint است، پس هر رکورد از پدر فقط و فقط می‌تواند یک فرزند داشته باشد. برای مثال رکورد GID1,GN1 فقط و فقط می‌تواند یک فرزند مثل A1,B1 از موجودیت E یا از موجودیت F داشته باشد.

توجه: اگر e مجموعه نمونه‌های E ، f مجموعه نمونه‌های F و g مجموعه نمونه‌های G باشد، آنگاه $card(e)+card(f)\le card(g)$ است. یعنی $1+1\le 3$

ج) مدل تحلیل کامل (اجباری) و Overlap

مدل تحلیل (نمودار ISA)



مدل طراحی (مدل رابطه‌ای)

مدل تک جدولی بهینه در شراط خاص: دقت کنید که در شرایط خاص یعنی زمانی‌که (1) تعداد صفات زبرنوع زیاد، (2) تعداد زیرنوع‌ها کم و (3) تعداد صفات زیرنوع‌ها کم باشد مدل تحلیل کامل (اجباری) و Overlap امکان پیاده‌سازی تک جدولی دارد و بهینه هم هست.

سربار افزایش مقدار NULL ندارد: چون ارتباط زبرنوع با زیرنوع کامل (اجباری) است، پس رکوردهای زبرنوع حداقل یک فرزند دارد و به تبع مقادیر NULL افزایشی نمی‌شود. همچنین تعداد صفات زیرنوع‌ها هم جهت NULL سازی کم است.

سربار افزایش افزونگی ندارد: چون در مدل تک جدولی یک رکورد خاص زبرنوع فقط در یک جدول درج می‌شود.

سربار الحاق طبیعی ندارد: چون در مدل تک جدولی الحاق میان جدول زبرنوع و زیرنوع وجود ندارد.

صفات مشترک		صفات خاص E		صفات خاص F	
GID	GName	A	B	X	Y
GID1	**GN1**	**A1**	**B1**	**X1**	**Y1**
GID2	**GN2**	**A2**	**B2**	NULL	NULL
GID3	**GN3**	**A3**	**B3**	**X3**	**Y3**

جدول GEF

توجه: دقت کنید چون رابطه پدر با فرزندان از نوع کامل (اجباری) است، پس هر رکورد از پدر حداقل باید یک فرزند داشته باشد. برای مثال رکورد GID1,GN1 حداقل یک فرزند مثل A1,B1 از موجودیت E دارد.

توجه: دقت کنید چون رابطه فرزندان با پدر از نوع Overlap است، پس هر رکورد از پدر ممکن است بیش از یک فرزند داشته باشد. برای مثال رکورد GID1,GN1 می‌تواند دو فرزند A1,B1 از موجودیت E و X1,Y1 از موجودیت F داشته باشد.

مدل دو جدولی غیربهینه به دلیل افزایش افزونگی: مدل تحلیل کامل (اجباری) و Ovelap امکان پیاده‌سازی دو جدولی دارد اما بهینه نیست.

سربار افزایش مقدار NULL ندارد: چون ارتباط زبرنوع با زیرنوع کامل (اجباری) است، پس رکوردهای زبرنوع حداقل یک فرزند دارد و به تبع مقادیر NULL افزایشی نمی‌شود.
سربار افزایش افزونگی دارد: چون ارتباط زیرنوع با زبرنوع Overlap است، پس ممکن است یک رکورد خاص زبرنوع G در دو جدول E و F تکرار شود.
سربار الحاق طبیعی ندارد: چون در مدل دو جدولی الحاق میان جدول زبرنوع و زیرنوع وجود ندارد.

GID	GName	A	B
GID1	**GN1**	A1	B1
GID2	**GN2**	A2	B2
GID3	**GN3**	A3	B3

جدول E

GID	GName	X	Y
GID1	**GN1**	X1	Y1
GID3	**GN3**	X3	Y3

جدول F

مدل سه جدولی بهینه با تحمل سربار الحاق طبیعی: مدل تحلیل کامل (اجباری) و Overlap امکان پیاده‌سازی سه جدولی دارد و بهینه هم هست.
سربار افزایش مقدار NULL ندارد: چون در مدل سه جدولی جداول زبرنوع و زیرنوع ازهم جدا هستن.
سربار افزایش افزونگی ندارد: چون در مدل سه جدولی جداول زبرنوع و زیرنوع ازهم جدا هستن.
سربار الحاق طبیعی دارد: چون در مدل سه جدولی الحاق میان جدول زبرنوع و زیرنوع وجود دارد.

GID	GName
GID1	GN1
GID2	GN2
GID3	GN3

جدول G

GID	A	B
GID1	A1	B1
GID2	A2	B2
GID3	A3	B3

جدول E

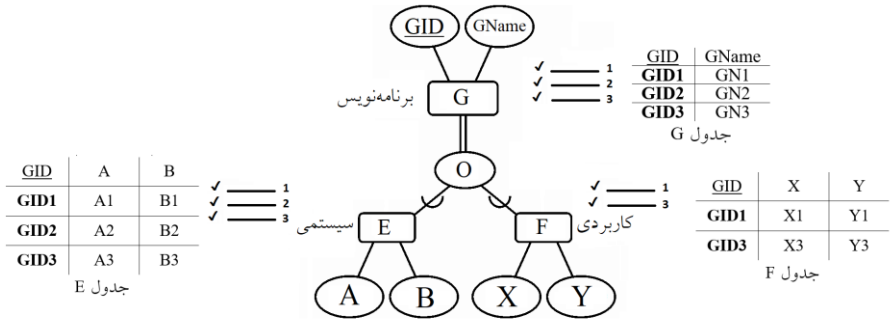
GID	X	Y
GID1	X1	Y1
GID3	X3	Y3

جدول F

توجه: دقت کنید که در رابطه‌های نشان دهنده هر زیرنوع، کلید خارجی جدول زیرنوع، کلید کاندید آن هم هست.

توجه: دقت کنید که تکرار GID1 یا GID3 در دو جدول E و F افزونگی طبیعی محسوب نمی‌شود بلکه افزونگی تکنیکی به دلیل تعریف کلید خارجی محسوب می‌شود.

مدل شماتیک مدل تحلیل کامل (اجباری) و Overlap به صورت زیر است:



GID	GName
GID1	GN1
GID2	GN2
GID3	GN3

جدول G

GID	A	B
GID1	A1	B1
GID2	A2	B2
GID3	A3	B3

جدول E

GID	X	Y
GID1	X1	Y1
GID3	X3	Y3

جدول F

توجه: دقت کنید چون رابطه پدر با فرزندان از نوع کامل (اجباری) است، پس هر رکورد از پدر حداقل باید یک فرزند داشته باشد. برای مثال رکورد GID1,GN1 حداقل یک فرزند مثل A1,B1 از موجودیت E دارد.

توجه: دقت کنید چون رابطه فرزندان با پدر از نوع Overlap است، پس هر رکورد از پدر ممکن است بیش از یک فرزند داشته باشد. برای مثال رکورد GID1,GN1 می‌تواند دو فرزند A1,B1 از موجودیت E و X1,Y1 از موجودیت F داشته باشد.

توجه: اگر e مجموعه نمونه‌های E ، f مجموعه نمونه‌های F و g مجموعه نمونه‌های G باشد، آنگاه $card(e)+card(f)\geq card(g)$ است. یعنی $3+2\geq 3$

سوال: بر اساس نمودار فوق، طراحی پایگاه داده رابطه‌ای با چند رابطه (جدول) باید انجام شود تا هیچ‌مقدار (NULL) پدید نیاید؟

الف) یک

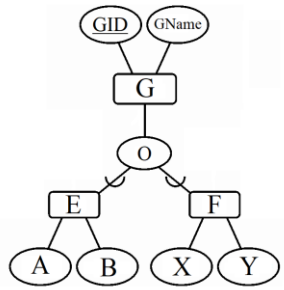
ب) دو

ج) سه

د) دو و سه

د) مدل تحلیل ناقص (اختیاری) و Overlap

مدل تحلیل (نمودار ISA)



مدل تک جدولی غیربهینه به دلیل افزایش مقدار NULL: دقت کنید که در شرایط خاص یعنی زمانی‌که (1) تعداد صفات زبرنوع زیاد، (2) تعداد زیرنوع‌ها کم و (3) تعداد صفات زیرنوع‌ها کم باشد مدل تحلیل ناقص (اختیاری) و Overlap امکان پیاده‌سازی تک جدولی دارد اما بهینه نیست.

سربار افزایش مقدار NULL دارد: چون ارتباط زبرنوع با زیرنوع ناقص (اختیاری) است، پس ممکن است رکوردهای زبرنوع، فرزند نداشته باشد و به تبع مقادیر NULL افزایشی می‌شود. مانند سطر 2. هرچند تعداد صفات زیرنوع‌ها هم جهت NULL سازی کم است.

سربار افزایش افزونگی ندارد: چون در مدل تک جدولی یک رکورد خاص زبرنوع فقط در یک جدول درج می‌شود.

سربار الحاق طبیعی ندارد: چون در مدل تک جدولی الحاق میان جدول زبرنوع و زیرنوع وجود ندارد.

صفات مشترک		صفات خاص E		صفات خاص F	
GID	GName	A	B	X	Y
GID1	**GN1**	**A1**	**B1**	**X1**	**Y1**
GID2	**Gn2**	NULL	NULL	NULL	NULL
GID3	**Gn3**	**A3**	**B3**	NULL	NULL

جدول GEF

توجه: دقت کنید چون رابطه پدر با فرزندان از نوع ناقص (اختیاری) است، پس ممکن است رکوردهای پدر، فرزند نداشته باشد. برای مثال رکورد GID2,GN2 هیچ فرزندی ندارد.

توجه: دقت کنید چون رابطه فرزندان با پدر از نوع Overlap است، پس هر رکورد از پدر ممکن است بیش از یک فرزند داشته باشد. برای مثال رکورد GID1,GN1 می‌تواند دو فرزند A1,B1 از موجودیت E و X1,Y1 از موجودیت F داشته باشد.

مدل دو جدولی غیربهینه به دلیل افزایش افزونگی و افزایش مقادیر NULL: مدل تحلیل کامل (اجباری) و Ovelap امکان پیاده‌سازی دو جدولی دارد اما بهینه نیست.

سربار افزایش مقادیر NULL دارد: چون ارتباط زبرنوع با زیرنوع ناقص (اختیاری) است، پس ممکن است زبرنوع، فرزند نداشته باشد و به تبع مقادیر NULL افزایشی می‌شود. مانند سطر 2 در جدول E و F.

سربار افزایش افزونگی دارد: چون ارتباط زیرنوع با زبرنوع Overlap است، پس ممکن است یک رکورد خاص زبرنوع G در دو جدول E و F تکرار شود.

سربار الحاق طبیعی ندارد: چون در مدل تک جدولی الحاق میان جدول زبرنوع و زیرنوع وجود ندارد.

GID	GName	A	B
GID1	**GN1**	A1	B1
GID2	**GN2**	NULL	NULL
GID3	**GN3**	A3	B3

جدول E

GID	GName	X	Y
GID1	**GN1**	X1	Y1

جدول F

مدل سه جدولی بهینه با تحمل سربار الحاق طبیعی: مدل تحلیل ناقص (اختیاری) و Overlap امکان پیاده‌سازی سه جدولی دارد و بهینه هم هست.

سربار افزایش مقدار NULL ندارد: چون در مدل سه جدولی جداول زبرنوع و زیرنوع ازهم جدا هستن.

سربار افزایش افزونگی ندارد: چون در مدل سه جدولی جداول زبرنوع و زیرنوع ازهم جدا هستن.

سربار الحاق طبیعی دارد: چون در مدل سه جدولی الحاق میان جدول زبرنوع و زیرنوع وجود دارد.

GID	GName
GID1	GN1
GID2	GN2
GID3	GN3

جدول G

GID	A	B
GID1	A1	B1
GID3	A3	B3

جدول E

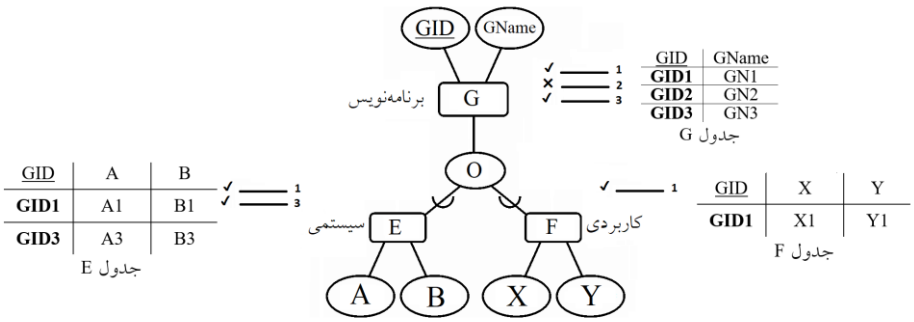
GID	X	Y
GID1	X1	Y1

جدول F

توجه: دقت کنید که مقدار GID2 در جدول E یا F درج نشده است چون فرض شده است مقادیر صفات خاص آن برابر مقدار NULL است. بنابراین مدل سه جدولی سربار افزایش مقادیر NULL را ندارد.

توجه: دقت کنید که در رابطه‌های نشان دهنده هر زیرنوع، کلید خارجی جدول زیرنوع، کلید کاندید آن هم هست.

مدل شماتیک مدل تحلیل ناقص (اختیاری) و Overlap به صورت زیر است:



توجه: دقت کنید چون رابطه پدر با فرزندان از نوع ناقص (اختیاری) است، پس ممکن است رکوردهای پدر، فرزند نداشته باشد. برای مثال رکورد GID2,GN2 هیچ فرزندی ندارد.

توجه: دقت کنید چون رابطه فرزندان با پدر از نوع Overlap است، پس هر رکورد از پدر ممکن است بیش از یک فرزند داشته باشد. برای مثال رکورد GID1,GN1 می‌تواند دو فرزند A1,B1 از موجودیت E و X1,Y1 از موجودیت F داشته باشد.

توجه: اگر e مجموعه نمونه‌های E ، f مجموعه نمونه‌های F و g مجموعه نمونه‌های G باشد، آنگاه $card(g) \geq card(e) + card(f)$ است. یعنی $3 \geq 1+2$

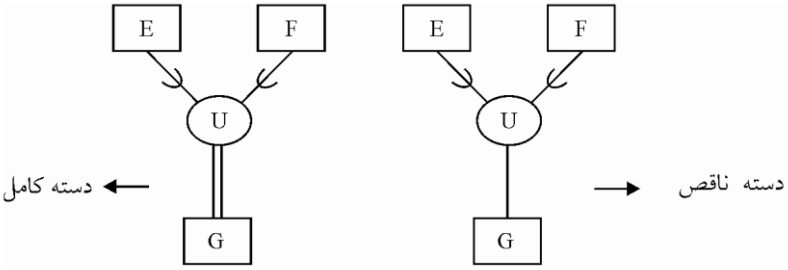
زیرنوع اجتماع (Union subtype: U-Type)

زیرنوع موجودیت G زیرنوع اجتماع نوع موجودیت‌های E ، F و... است هرگاه مجموعه نمونه‌های G ، اجتماع مجموعه نمونه‌هایی (بعضی نمونه‌ها یا همه) از E ، F و... باشد.

بنابراین زیرنوع اجتماع، گونه‌ای زیرنوع است که اولا بیش از یک زبرنوع دارد و ثانیا مجموعه نمونه‌هایش، اجتماع بعضی از (یا همه) نمونه‌های زبرنوع‌هایش است.

توجه: گاه به این گونه زیرنوع ، **دسته (Category)** یا طبقه گفته می‌شود. **زیرنوع اجتماع** را با حرف **U** (حرف اول کلمه Union) نمایش می‌دهیم.

توجه: دسته ممکن است **کامل** یا **ناقص** باشد. دسته وقتی کامل است که در مجموعه نمونه‌های زیرنوع اجتماع، **همه نمونه‌های همه زبرنوع‌ها** وجود داشته باشند، و در غیر اینصورت، دسته را ناقص گوییم.



توجه: در اینجا G یک دسته از E و F است.

توجه: یک نمونه از G یا نمونه‌ای از E و یا نمونه‌ای از F و ... است. مجموعه نمونه‌های G اجتماع نمونه‌هایی از E و نمونه‌هایی از F و ... است.

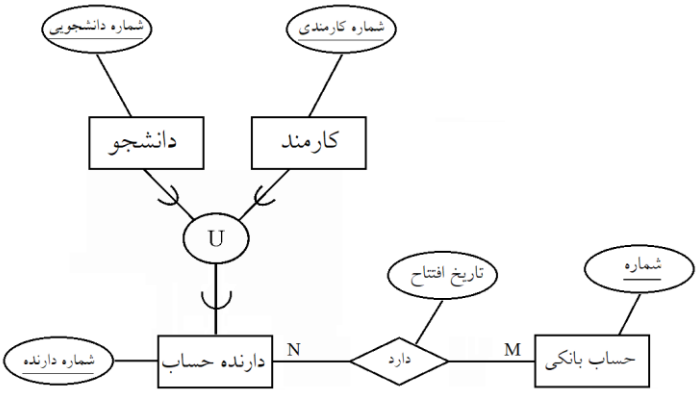
توجه: یک نمونه دسته (زیرنوع اجتماع)، بسته به اینکه از نوع کدام زیرنوع باشد، صفات همان زیرنوع را به ارث می‌برد. بر این اساس به زیرنوع اجتماع **وراثت انتخابی** نیز گفته می‌شود.

توجه: شناسه زیرنوع‌های یک **زیرنوع اجتماع** می‌تواند از دو نوع میدان (دامنه) باشد:

شناسه زیرنوع از میدان متفاوت: پس شناسه **زیرنوع** شناسه‌ای است که طراح باید در نظر بگیرد.

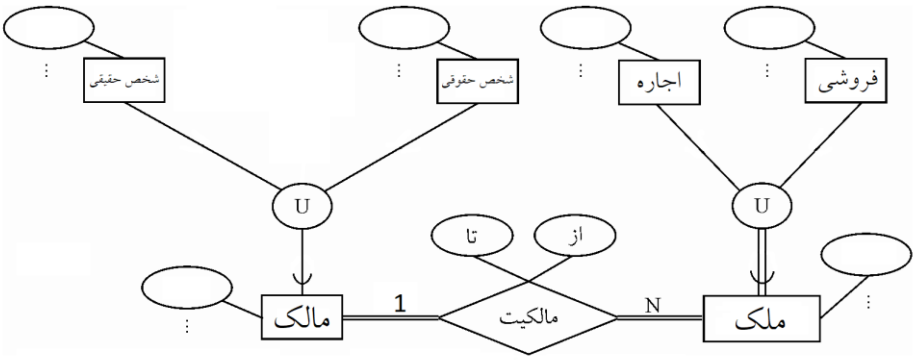
شناسه زیرنوع از میدان یکسان: پس شناسه **زیرنوع** همان شناسه زیرنوع‌ها است.

مثال: نوع موجودیت‌های دانشجو و کارمند را در نظر بگیرید. این دو نوع موجودیت می‌توانند حساب بانکی داشته باشند. در این مثال نوع موجودیت دارنده حساب می‌تواند دانشجو یا کارمند باشد (در مجموعه نمونه‌های آن، هم نمونه‌هایی از دانشجو وجود دارد و هم نمونه‌هایی از کارمند.) همانطور که گفتیم شناسه زیرنوع‌های یک زیرنوع اجتماع ممکن است از یک میدان باشند یا از میدان‌های متفاوت. در این مثال اگر به جای صفت شماره کارمندی و صفت شماره دانشجویی، یک صفت دیگر برای‌مثال کد ملی را شناسه بگیریم، شناسه دو زیرنوع دانشجو و کارمند، از یک میدان می‌شود. وقتی که شناسه زیرنوع‌ها از یک میدان نباشد، طراح باید برای زیرنوع «**دسته**»، یک شناسه در نظر بگیرد، مثل صفت شماره دارنده حساب در شکل زیر. اگر شناسه زیرنوع‌ها یکسان باشد، همان شناسه برای زیرنوع «**دسته**» منظور می‌شود.



توجه: اگر N تعداد زیرنوع‌های زیرنوع «دسته» باشد، تعداد جداول لازم N+1 است.

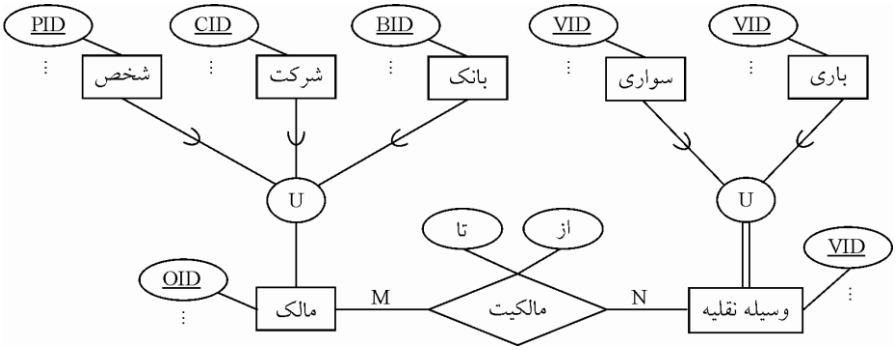
مثال: نمودار EER که دسته‌های املاک را نشان می‌دهد:



در زیرنوع اجتماع ملک، هر ملک یا فروشی است یا اجاره‌ای. پس نمونه‌های زیرنوع اجتماع جدول ملک، اجتماع نمونه املاک فروشی و نمونه املاک اجاره‌ای است. رکورد اطلاعاتی همه نمونه املاک فروشی یا اجاره‌ای الزاما به عنوان ملک در جدول ملک ثبت می‌شوند، چون یک ملک فروشی یا اجاره‌ای قطعا ملک است. پس زیرنوع اجتماع ملک یک **دسته کامل** است.

در زیرنوع اجتماع مالک، هر مالک یا شخص حقوقی است یا شخص حقیقی. پس نمونه‌های زیرنوع اجتماع جدول مالک، اجتماع نمونه اشخاص حقوقی و نمونه اشخاص حقیقی است. رکورد اطلاعاتی همه نمونه اشخاص حقوقی یا اشخاص حقیقی الزاما به عنوان مالک در جدول مالک ثبت نمی‌شوند، چون ممکن است برخی نمونه‌های شخص حقوقی یا شخص حقیقی مالک نباشند. پس زیرنوع اجتماع مالک یک **دسته ناقص** است.

مثال- مدل رابطه‌ای متناظر با نمودار **EER** زیر به چه صورتی است؟



به طور کلی در مدل رابطه‌ای، هر موجودیت شناسایی شده در نمودار ER (مدل تحلیل) هنگام نگاشت به مدل رابطه‌ای (مدل طراحی) به یک جدول تبدیل می‌شود. همچنین صفت‌های موجودیت پس از نگاشت آن در مدل رابطه‌ای به صورت ستون‌های جدول بیان می‌شوند. همچنین ارتباط بین جداول از طریق کلید خارجی برقرار می‌گردد.

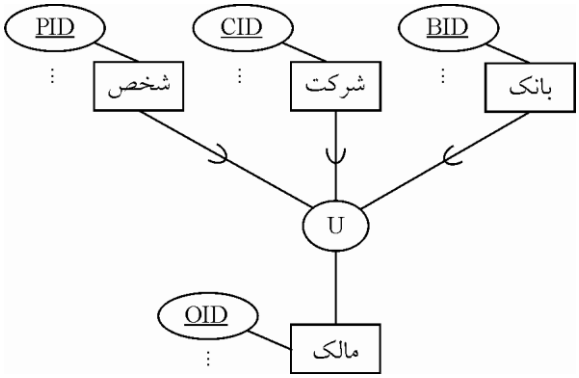
در زیرنوع اجتماع وسیله نقلیه، هر وسیله نقلیه یا باری است یا سواری. پس نمونه‌های زیرنوع اجتماع جدول وسیله نقلیه، اجتماع نمونه وسایل نقلیه باری و سواری است. رکورد اطلاعاتی همه نمونه وسایل نقلیه باری یا سواری الزاما به عنوان وسیله نقلیه در جدول وسیله نقلیه ثبت می‌شوند، چون یک وسیله نقلیه باری یا سواری قطعا وسیله نقلیه است. پس زیرنوع اجتماع وسیله نقلیه یک **دسته کامل** است.

در زیرنوع اجتماع مالک، هر مالک یا بانک است یا شرکت یا شخص. پس نمونه‌های زیرنوع اجتماع جدول مالک، اجتماع نمونه‌های بانک و شرکت و شخص است. رکورد اطلاعاتی همه نمونه‌های بانک یا شرکت یا شخص الزاما به عنوان مالک در جدول مالک ثبت نمی‌شوند، چون ممکن است برخی نمونه‌های بانک یا شرکت یا شخص مالک نباشند. پس زیرنوع اجتماع مالک یک **دسته ناقص** است.

نگاشت رابطه زیرنوع‌ها از میدان متفاوت و زیرنوع اجتماع U-Type به مدل رابطه‌ای

هر موجودیت به یک جدول تبدیل می‌گردد، در شکل زیر شناسه زیرنوع‌ها از **میدان متفاوت** (PID,CID,BID) است، پس شناسه OID توسط طراح بانک برای زیرنوع در نظر گرفته می‌شود. همچنین کلید کاندید موجودیت زیرنوع (همان شناسه انتخاب شده توسط طراح بانک) به عنوان کلیدخارجی در موجودیت زیرنوع‌ها تعریف می‌شود.

مدل تحلیل:



مدل طراحی:

PID	…	OID
PID1		OID1
PID2		OID2

جدول Pers

CID	…	OID
CID1		OID3
CID2		NULL

جدول Comp

BID	…	OID
BID1		OID4
BID2		NULL

جدول Bank

OID	…
OID1	
OID2	
OID3	
OID4	

جدول Owner

توجه: ستون OID در جدول‌های Pers، Comp و Bank به عنوان کلید خارجی تعریف می‌گردد که به جدول Owner ارجاع می‌کند.

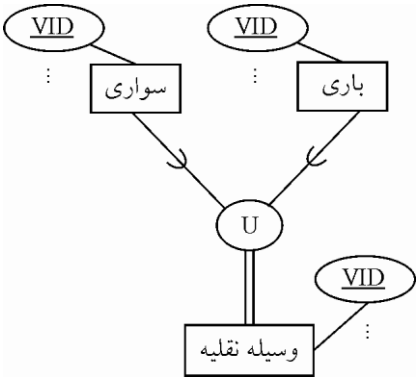
PERS (PID ,…, OID)
COMP (CID ,…, OID)
BANK (BID ,…, OID)
OWNER (OID , …)

توجه: مثال فوق از نوع دسته ناقص است و این موضوع مستقل از نوع میدان است.

نگاشت رابطه زبرنوع‌ها از میدان یکسان و زیرنوع اجتماع U-Type به مدل رابطه‌ای

هر موجودیت به یک جدول تبدیل می‌گردد، اگر شناسه زبرنوع‌ها از **میدان یکسان** باشد، کلید کاندید موجودیت زیرنوع، همان کلید کاندید موجودیت زبرنوع‌ها است.

مدل تحلیل:



مدل طراحی:

VID	N	…
VID1		
VID2		

جدول Savary

VID	T	…
VID3		
VID4		

جدول Bary

VID	…
VID1	
VID2	
VID3	
VID4	

جدول Vehic

SAVARY (VID ,…)
BARY (VID ,…)
VEHIC (VID , …)

توجه: مثال فوق از نوع دسته کامل است و این موضوع مستقل از نوع میدان است.

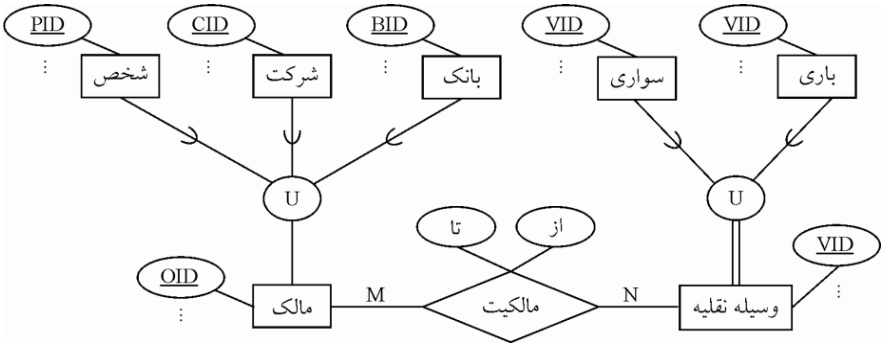
نگاشت رابطه چند به چند بین دو موجودیت به مدل رابطه‌ای

مستقل از اختیاری یا اجباری بودن موجودیت‌ها، هر موجودیت به یک جدول تبدیل می‌گردد و یک جدول پُل (Bridge) نیز به عنوان ارتباط دهنده دو جدول مورد استفاده قرار می‌گیرد. همچنین

کلید کاندید جدول پُل از ترکیب کلید کاندید دو جدول دیگر ایجاد می‌گردد. همچنین صفات متصل به **رابطه**، درون جدول پُل **مستتر** می‌شود.

روال کلی نگاشت در این حالت به صورت زیر است:

مدل تحلیل:



توجه: در شکل فوق صفت OID کلید موجودیت Owner و صفت VID کلید موجودیت Vehic است.

مدل طراحی:

OID	…

جدول Owner

OID	VID	F	T

جدول Owns

VID	…

جدول Vehic

توجه: ستون OID در جدول Owns به عنوان کلید خارجی تعریف می‌گردد که به جدول Owner ارجاع می‌کند. همچنین ستون VID در جدول Owns به عنوان کلید خارجی تعریف می‌گردد که به جدول Vehic ارجاع می‌کند.

توجه: همچنین صفات متصل (مالکیت) به **رابطه**، درون جدول پُل یعنی جدول Owns **مستتر** می‌شود.

توجه: همچنین کلید کاندید جدول پُل از ترکیب کلید کاندید دو جدول دیگر ایجاد می‌گردد. یعنی کلید کاندید جدول Owns برابر (OID,VID) است.

توجه: کلید کاندید جدول چند چپ یعنی **موجودیت Owner** برابر همان **کلید کاندید سابق** در جدول موجودیت Owner است. یعنی کلید کاندید جدول Owner برابر (OID) است.

توجه: کلید کاندید جدول چند راست یعنی **موجودیت Vehic** برابر همان **کلید کاندید سابق** در جدول موجودیت Vehic است. یعنی کلید کاندید جدول Vehic برابر (VID) است.

توجه: جدول Owner در نگاشت مرحله قبل ایجاد شده است که در اینجا دقیقاً به همان شکل و همان مشخصات، استفاده شده است.

توجه: جدول Vehic در نگاشت مرحله قبل ایجاد شده است که در اینجا دقیقاً به همان شکل و همان مشخصات، استفاده شده است.

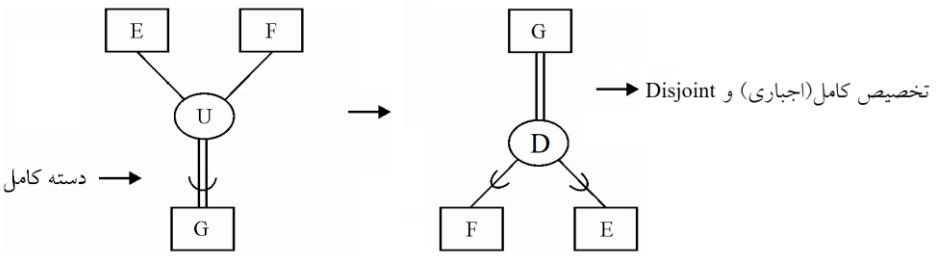
معادل‌سازی U-Type و Disjoint

زیرنوع (ابرنوع) در U-Type معادل زیرنوع در Disjoint است، البته اگر شرایط معادل بودن برقرار باشد.

توجه: در U-Type زیرنوع (ابرنوع) می‌تواند اجباری (کامل) یا اختیاری (ناقص) باشد. اما در Disjoint زیرنوع همواره بدون قید و شرط اجباری است. البته زیرنوع (ابرنوع) در Disjoint می‌تواند اجباری (کامل) یا اختیاری (ناقص) باشد. دقت کنید که زیرنوع در U-Type به خودی خود مفهوم اجباری و اختیاری را ندارد، بلکه اجباری و اختیاری بودن زیرنوع (ابرنوع) در زیرنوع نمایان می‌شود. که به آن «**دسته کامل**» یا «**دسته ناقص**» نیز گفته می‌شود.

نتیجه مهم: در U-Type اگر زیرنوع (ابرنوع) اجباری باشد که به تبع آن زیرنوع هم دسته کامل است، آنگاه در این شرایط U-Type معادل با زیرنوع Disjoint و زیرنوع (ابرنوع) اجباری (کامل) است.

توجه: هرگاه دسته **کامل** باشد، می‌توان به جای مدل «**دسته**»، از مدل تخصیص کامل(اجباری) و Disjoint استفاده کرد.

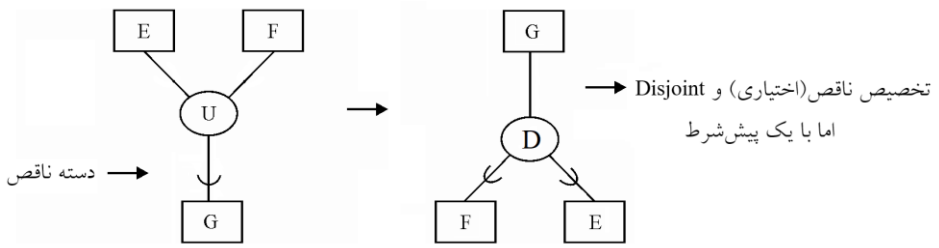


توجه: اینکه کدام مدل‌سازی انتخاب شود، بسته به نظر طراح بانک است.

توجه: هرگاه دسته **ناقص** باشد، می‌توان به جای مدل «**دسته**» ، از مدل تخصیص ناقص(اختیاری) و Disjoint اما با یک پیش‌شرط استفاده کرد.

پیش‌شرط: $C_G = C_E + C_F$

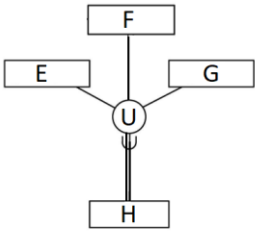
فرض کنید: C_X : کاردینالیتی مجموعه نمونه‌های نوع موجودیت X است.



توجه: اینکه کدام مدل‌سازی انتخاب شود، بسته به نظر طراح بانک است.

توجه: در U-Type، دسته کامل یا ناقص تحت هیچ شرایطی امکان مدل‌سازی توسط Overlap را ندارد. چون ماهیت Overlap همپوشانی است که با ذات زیرنوع اجتماع (U-Type) کاملا متفاوت است. همانطور که گفتیم یک نمونه دسته (زیرنوع اجتماع)، بسته به اینکه از نوع کدام زیرنوع باشد، صفات همان زیرنوع را به ارث می‌برد. و این به معنی مستقل بودن رکوردهای زیرنوع‌های مستقل ازهم در مفهوم U-Type است و تداعی‌کننده همان مفهوم Disjoint است.

سوال: با توجه به نمودار، نوع موجودیت H :



الف) همه صفات نوع موجودیت‌های E ، F و G را به ارث می‌برد.

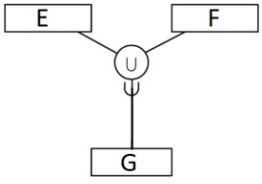
ب) همه صفات حداقل یکی از سه نوع موجودیت E ، F و G را به ارث می‌برد.

ج) حداقل شناسه سه نوع موجودیت E ، F و G را به ارث می‌برد.

د) صفات یکی از سه نوع موجودیت E یا F یا G را به ارث می‌برد.

پاسخ گزینه «د» صحیح است.

سوال: با توجه به نمودار زیر :



اگر e مجموعه نمونه‌های E ، f مجموعه نمونه‌های F و g مجموعه نمونه‌های G باشد:

الف) $g = e \cup f$

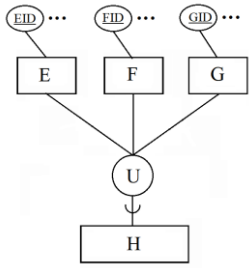
ب) $g \subset e \cup f$

ج) $g \subseteq e \cup f$

د) $g \supseteq e \cup f$

پاسخ گزینه «ج» صحیح است.

سوال: با توجه به نمودار، شناسه نوع موجودیت H چیست؟



فرض: $EID \neq FID \neq GID$

الف) $HID \in \{EID, FID, GID\}$

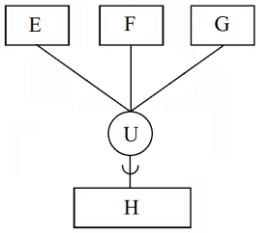
ب) $HID \in \{(\underline{EID, GID}), (\underline{EID, FID}), (\underline{GID, FID})\}$

ج) $HID \in \{\underline{EID, FID, GID}\}$

د) $HID \neq I \mid I \in \{EID, FID, GID\}$

پاسخ گزینه «د» صحیح است.

سوال: با توجه به نمودار، نوع موجودیت H چند شناسه دارد؟



الف) یک شناسه

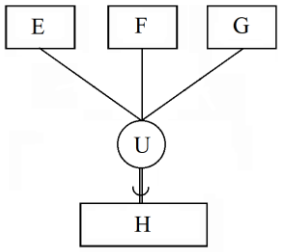
ب) یک شناسه حاصل ترکیب شناسه‌های E و F و G

ج) سه شناسه، هر یک حاصل ترکیب شناسه‌های هربار دو نوع موجودیت

د) سه شناسه

پاسخ گزینه «الف» صحیح است.

سوال: با توجه به نمودار، نوع موجودیت H چند شناسه دارد؟



الف) سه شناسه

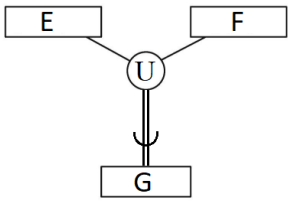
ب) حداکثر سه شناسه

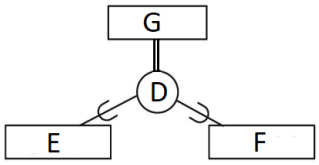
ج) یک شناسه

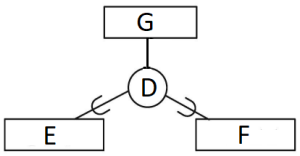
د) حداقل یک شناسه

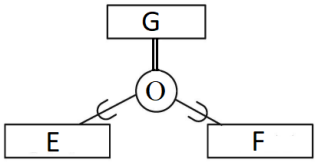
پاسخ گزینه «ج» صحیح است.

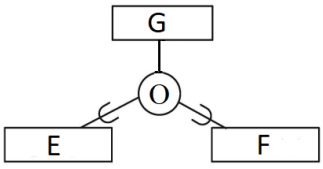
سوال: فرض کنید یک محیط را به صورت زیر مدل‌سازی کرده باشیم، به چه طرز دیگری می‌توان این محیط را مدل کرد؟



الف)


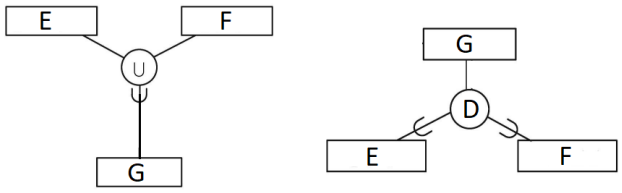
ب)


ج)


د)


پاسخ گزینه «الف» صحیح است.

سوال: مدل‌سازی I را با چه شرطی می‌توان به صورت II هم انجام داد؟



I II

فرض کنید: C_X : کاردینالیتی مجموعه نمونه‌های نوع موجودیت X است.

الف) $C_G \leq C_E + C_F$

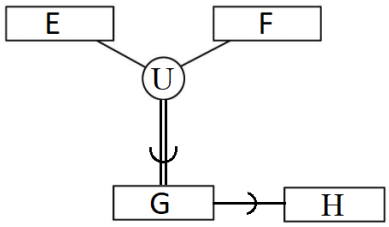
ب) $C_G > C_E + C_F$

ج) $C_G = C_E + C_F$

د) $C_G \geq C_E + C_F$

پاسخ گزینه «ج» صحیح است.

سوال: در نمودار زیر، شناسه نوع موجودیت H چه می‌تواند باشد؟



I : شناسه G ، یکسان با شناسه E و شناسه F

II : شناسه G ، یکسان با شناسه E یا شناسه F و نه هردو

III : شناسه G ، متفاوت با شناسه E و شناسه F

الف) I یا III

ب) II

ج) III

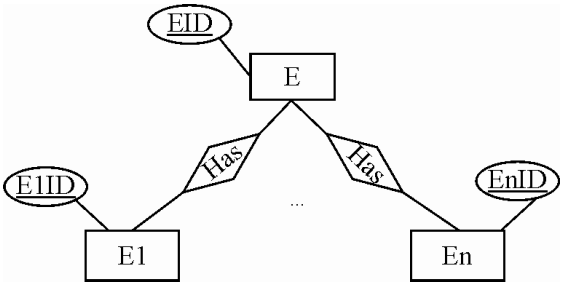
د) I یا II

پاسخ گزینه «الف» صحیح است.

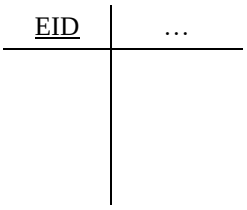
نگاشت رابطه IS-A-PART-OF به مدل رابطه‌ای

هر موجودیت به یک جدول تبدیل می‌گردد و کلید کاندید موجودیت زبرنوع در موجودیت‌های زیرنوع به عنوان کلید خارجی تعریف می‌گردد.

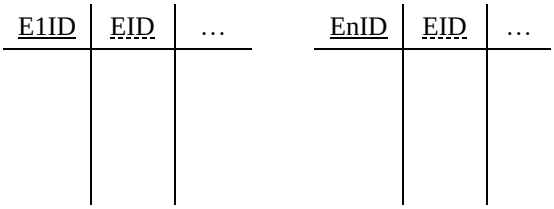
مدل تحلیل:



مدل طراحی:



جدول E



جدول E1 ***جدول En***

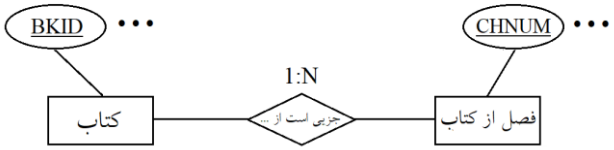
توجه: موجودیت کل (E) به طور مستقل برای خود کلید کاندید دارد و همچنین موجودیت‌های جزء (E1 و En) به طور مستقل برای خود کلید کاندید دارند.

توجه: کلید کاندید جدول E1 ترکیبی و به صورت E1ID,EID است.

توجه: ستون EID در جدول‌های E1 و Enk به عنوان کلید خارجی تعریف می‌گردد که به جدول E ارجاع می‌کند.

توجه: ستون EID در جدول E1 علاوه بر اینکه کلید خارجی است جزء کلید کاندید جدول E1 نیز است.

مثال: هر فصل کتاب جزیی از کتاب است.



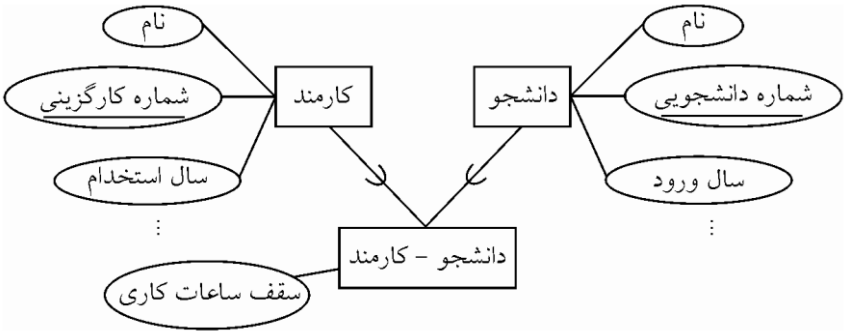
BOOK (BKID , BKTITLE , ...)
C.K.

CHAP (BKID, CHNUM , CHAPTITLE , NO – OF – PAGE , ...)
C.K.

نگاشت رابطه ارث‌بری چندگانه به مدل رابطه‌ای

هر موجودیت به یک جدول تبدیل می‌گردد و کلید کاندید موجودیت‌های زیرنوع در موجودیت زیرنوع به عنوان کلید خارجی تعریف می‌گردد. در جدول زیرنوع کلید خارجی همزمان کلید کاندید نیز می‌باشد.

مدل تحلیل:



مدل طراحی:

EID	Ename	…

جدول Empl

STID	STname	..

جدول Stud

EID	STID	MAXW

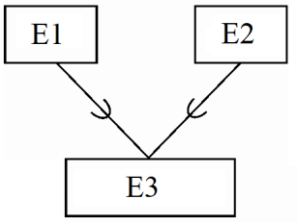
جدول Stem

EMPL (EID ,…)
STUD (STID ,…)
STEM (EID , … , STID , … , MAXW)

توجه: زیرنوع ارث‌بری چندگانه با اتصال به زبرنوع‌های خود توسط کلید خارجی، تمام صفات زبرنوع‌های خود را به ارث می‌برد. به عبارت دیگر مجموعه صفات جدول زیرنوع، **اجتماع** مجموعه صفات جداول زبرنوع‌ها است. بدین معنی که در اجتماع مجموعه صفات جداول زبرنوع‌ها، صفات تکراری یکبار در جدول زیرنوع قرار می‌گیرند و صفات تکراری حذف می‌شود.

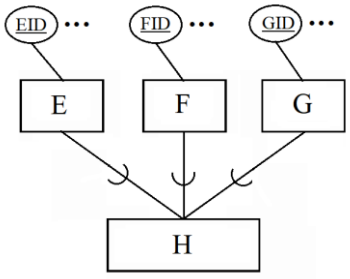
توجه: اگر زیرنوع تعداد N زبرنوع داشته باشد، جدول زیرنوع **حداقل** N کلید کاندید **مجزا** دارد. حداقل چون ممکن است هر زبرنوع خودش حداقل دو کلید کاندید مجزا داشته باشد.

سوال: در نمودار زیر، نوع موجودیت E3:



الف) دو شناسه دارد.
ب)حداقل یک شناسه مرکب دارد.
ج)حداقل دو شناسه دارد.
د) حداقل دو شناسه مرکب دارد.
پاسخ گزینه «ج» صحیح است.

سوال: با توجه به نمودار، شناسه نوع موجودیت H چیست؟



الف) $HID \in \{EID, FID, GID\}$

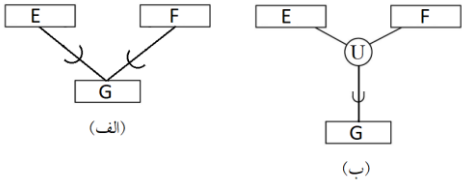
ب) $HID \in \{(EID, GID), (EID, FID), (GID, FID)\}$

ج) $HID = \{EID, FID, GID\}$

د) هیچکدام

پاسخ گزینه «الف» صحیح است.

سوال: با توجه به نمودار (الف) و (ب) و گزاره‌های I و II ، مورد درست کدام است؟



گزاره I : G یا صفات E را به ارث می‌برد یا صفات F را.

گزاره II : G هم صفات E را به ارث می‌برد و هم صفات F را.

الف) گزاره II برای (ب) و گزاره I برای (الف)

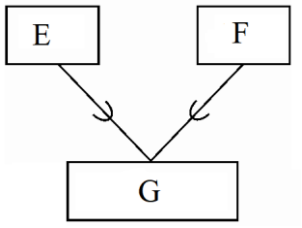
ب) گزاره II برای (الف) و گزاره I برای (ب)

ج) گزاره I برای (الف) و گزاره I برای (ب)

د) گزاره II برای (الف) و گزاره II برای (ب)

پاسخ گزینه «ب» صحیح است.

سوال: با توجه به نمودار زیر، اگر A_X مجموعه صفات نوع موجودیت X باشد، در اینصورت:



الف) $A_G = A_E \cup A_F$

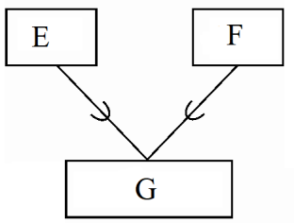
ب) $A_G \supseteq A_E \cup A_F$

ج) $A_G = A_E \cap A_F$

د) $A_G = (A_E - A_F) \cup (A_F - A_E)$

پاسخ گزینه «ب» صحیح است.

سوال: با توجه به نمودار زیر، اگر $Card[A_X]$ کاردینالیتی مجموع صفات نوع موجودیت X باشد، در این صورت :



الف) $Card[A_G] = Card[A_E] + Card[A_F]$

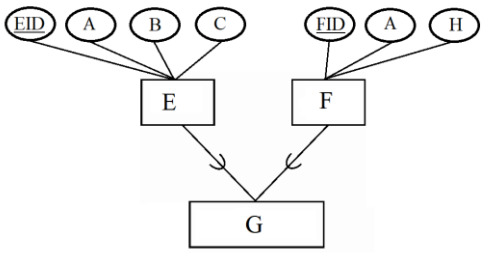
ب) $Card[A_G] \leq Card[A_E] + Card[A_F]$

ج) $Card[A_G] < Card[A_E \cup A_F]$

د) $Card[A_G] \geq Card[A_E \cup A_F]$

پاسخ گزینه «د» صحیح است. چون موجودیت G ممکن است صفات خاصه خودش را نیز اضافه کند.

سوال: با توجه به نمودار زیر، اگر در طراحی پایگاه داده رابطه‌ای برای نوع موجودیت G ، یک رابطه طراحی کنیم، درجه این رابطه چند است؟ (منظور از درجه ارتباط رابطه همان تعداد صفات G است.)



الف) پنج

ب) حداقل شش

ج) هفت

د) شش

پاسخ گزینه «د» صحیح است.