

چندی ارتباط

■ یک ارتباط از نظر چندی می تواند یکی از حالات زیر را داشته باشد:



■ یک به یک (One to One)



■ یک به چند (One to Many)



■ چند به چند (Many to Many)

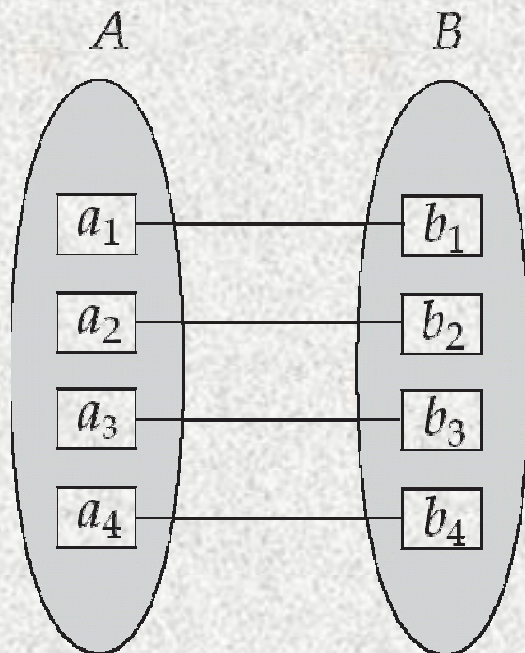
چندی ارتباط

■ اگر دو موجودیت A و B داشته باشیم ارتباط آنها از نظر چندی ارتباط می تواند به صورت های زیر باشد

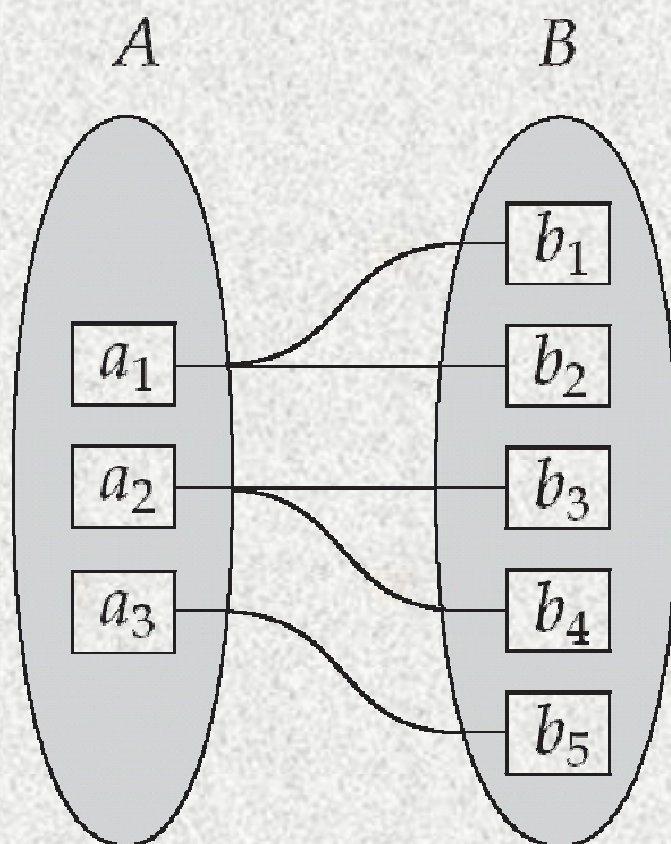
■ یک به یک (One to One):

یک نمونه از A حداکثر با یک نمونه از B

ارتباط دارد و برعکس



چندی ارتباط



■ یک به چند (One to Many):

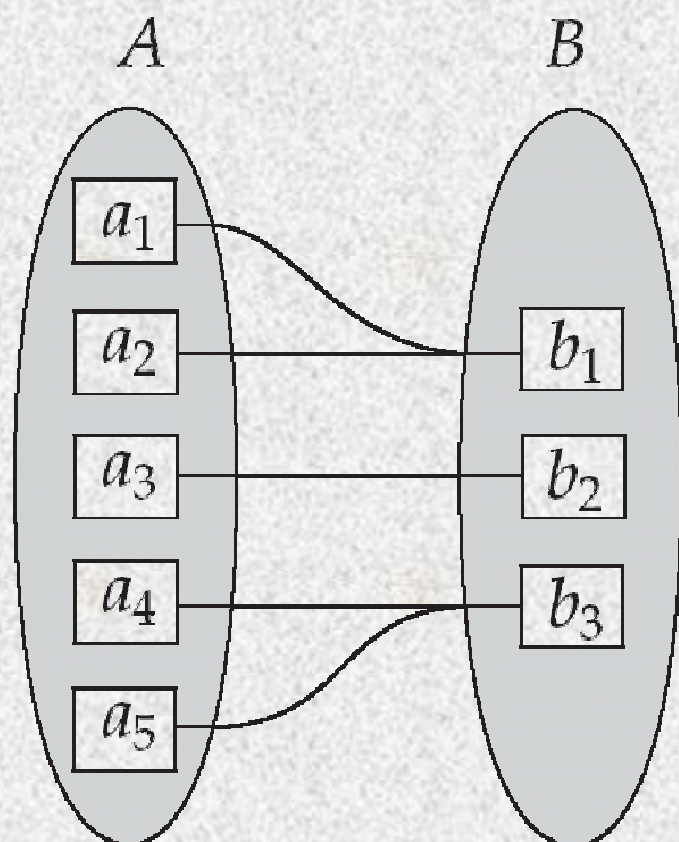
یک نمونه از A با چند نمونه از B

ارتباط دارد و هر نمونه از B فقط

با یک نمونه از A ارتباط دارد.



چندی ارتباط



■ چند به یک (Many to One):

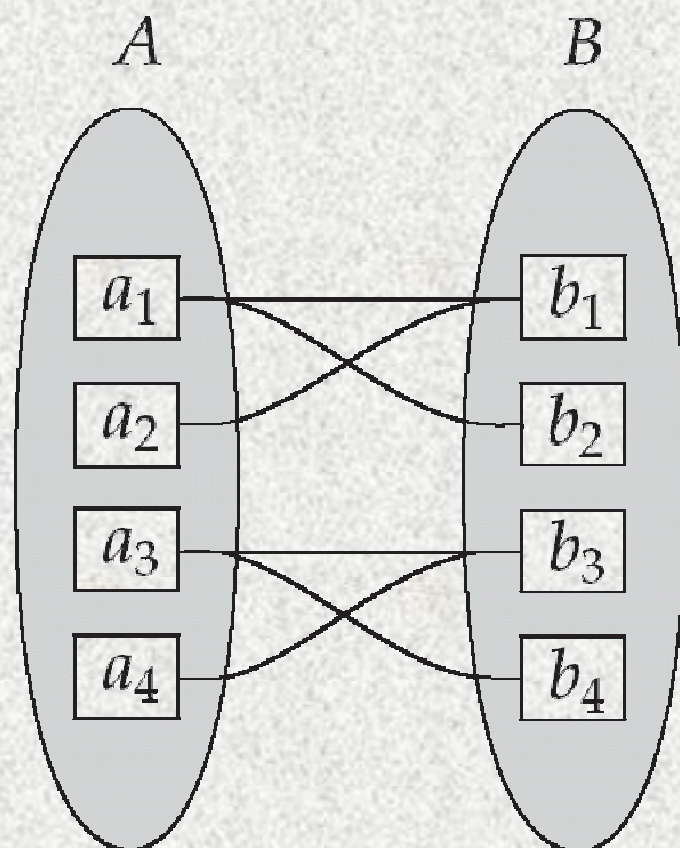
هر نمونه از A با یک نمونه از B

ارتباط دارد اما هر نمونه از B

با چند نمونه از A ارتباط دارد.



چندی ارتباط



■ چند به چند (Many to Many):

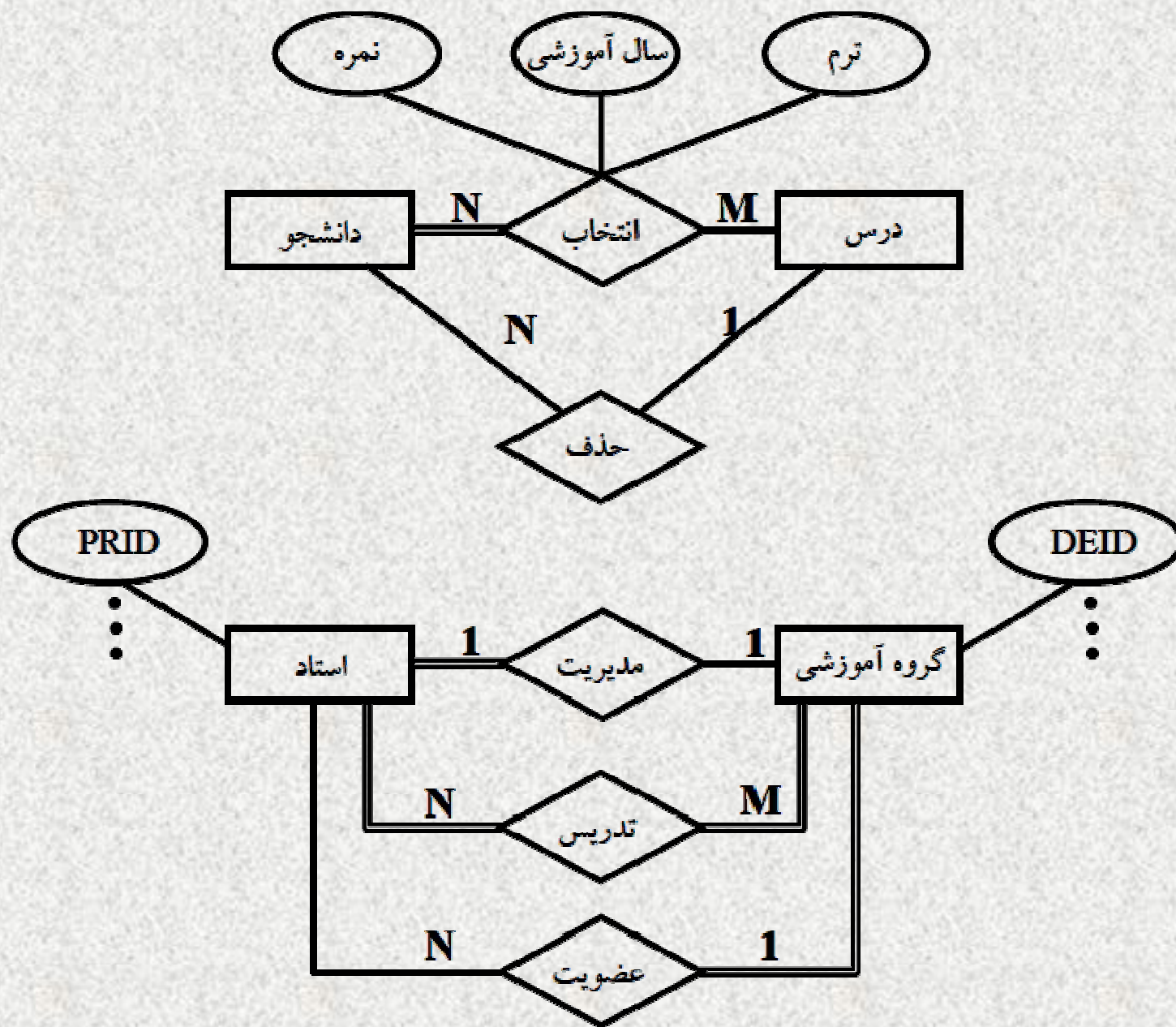
هر نمونه از A با چند نمونه از B

و هر نمونه از B با چند نمونه از A

ارتباط دارد.



مثال



حد (Cardinality) ارتباط

■ این روش روش دیگری برای نمایش چندی ارتباط می باشد حد یک ارتباط براساس جفت عدد صحیح به صورت (max,min) نشان داده می شود که در آن:

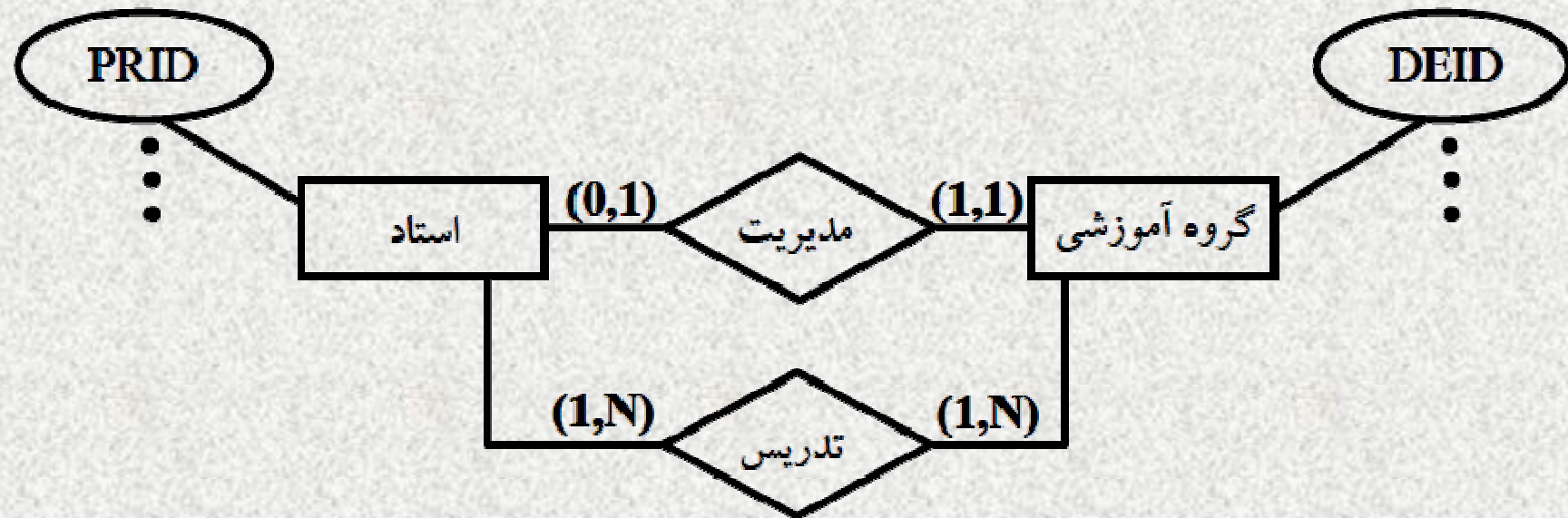
$$0 \leq \text{Min} \leq \text{Max}, 1 \leq \text{Max}$$

معنای این دو عدد این است که در هر لحظه هر نمونه ی e از موجودیت E باید حداقل در min و حداکثر در Max نمونه از ارتباط R شرکت داشته باشد

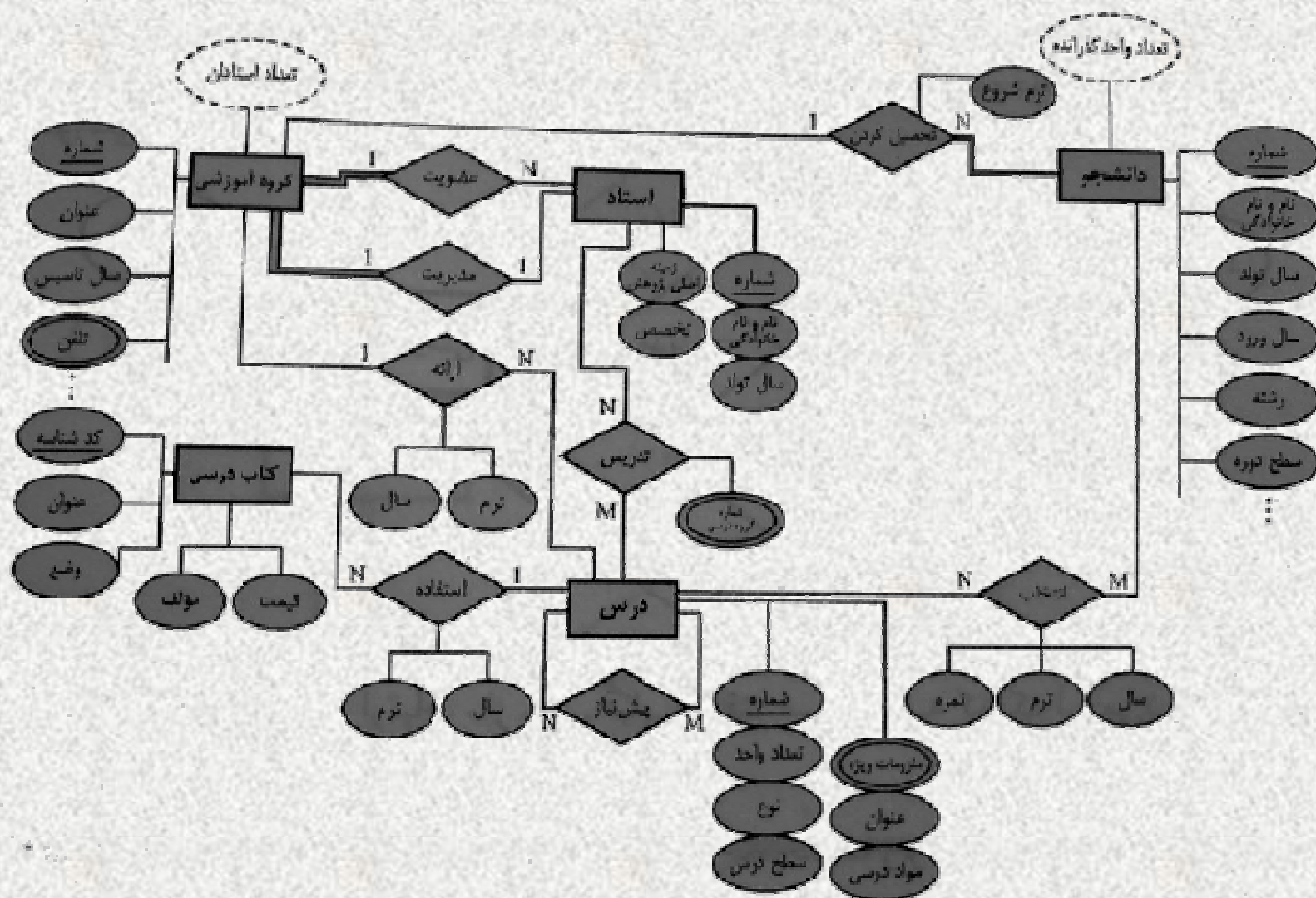
حد (Cardinality) ارتباط

- اگر min برابر صفر باشد یعنی مشارکت غیر الزامی و اگر بزرگتر از صفر باشد یعنی مشارکت الزامی است.
- از طریق این اعداد می توان حتی حالت یک به یک بودن، یک به چند و چند به چند بودن را نیز تشخیص داد.
- این روش نمایش دقیق تر از روش قبل می باشد.

حد (Cardinality) ارتباط



مدلسازی معنایی داده دانشکده (بسیار ساده شده)



ارتباط (Relationship)

■ با توجه به آنچه که گفته شد هر نوع ارتباط خصوصیات زیر را دارد:

■ نام

■ معنا

■ نمونه ها

■ حداقل یک شرکت کننده

■ درجه

■ وضعیت مشارکت شرکت کننده ها در ارتباط

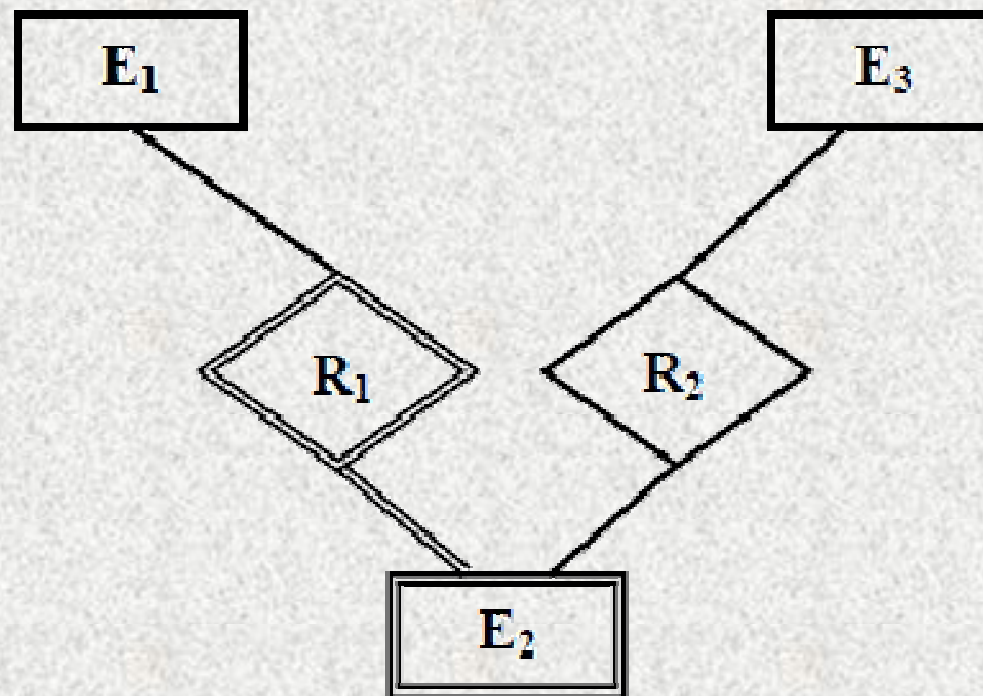
■ چندی

موجودیت ضعیف – تکمیلی

- هر چند موجودیت ضعیف شناسه ندارد اما یک صفت ممیزه دارد که به آن کلید جزئی نیز گفته می شود
- نوع موجودیت ضعیف از نوع موجودیت قوی شناسه می گیرد. در واقع شناسه ی نوع موجودیت ضعیف ترکیب شناسه ی موجودیت قوی و صفت ممیزه ی موجودیت ضعیف است
- موجودیت ضعیف خود می تواند موجودیت ضعیف دیگر نیز داشته باشد

موجودیت ضعیف – تکمیلی

■ در شکل زیر E_2 موجودیت ضعیف و E_1 موجودیت قوی آن می باشد، همچنین یک موجودیت قوی دیگر E_3 وجود دارد که ارتباط R_2 یا E_2 دارد



موجودیت ضعیف – تکمیلی

- نکات مهم در مورد نوع موجودیت ضعیف عبارتست از:
 - درجه ی ارتباط معمولاً ۲ است اما می تواند بیشتر هم باشد.
 - چندی ارتباط همیشه ۱ به N است از طرف موجودیت قوی چندی ۱ و از طرف موجودیت ضعیف چندی N .
 - موجودیت ضعیف از خود شناسه ندارد بلکه یک صفت ممیزه دارد
 - مشارکت نوع موجودیت ضعیف در ارتباط همیشه الزامی است

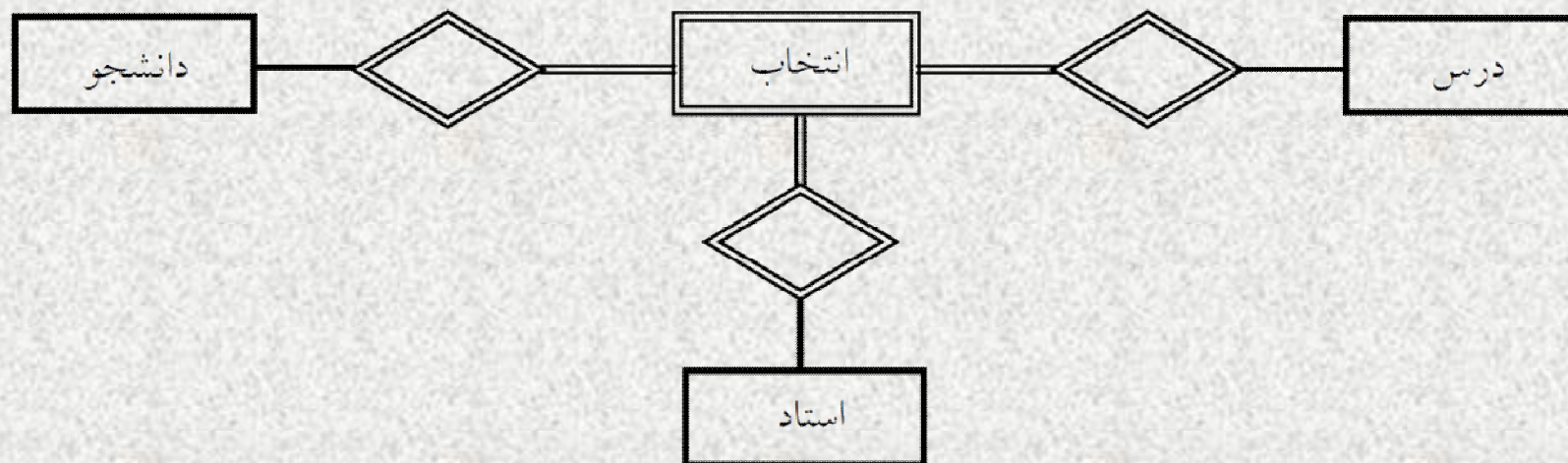
موجودیت ضعیف – تکمیلی

■ ارتباط به مثابه نوع موجودیت ضعیف: می توانیم یک ارتباط را به صورت

یک موجودیت ضعیف هم در نظر بگیریم

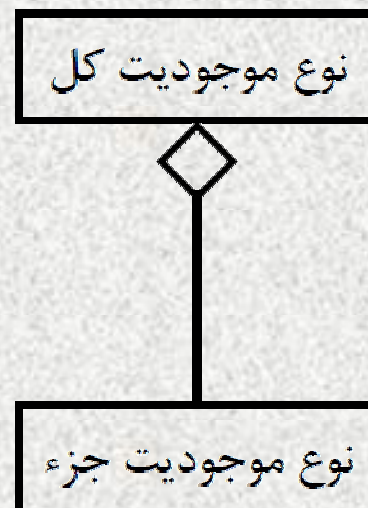
■ به عنوان مثال ارتباط درجه ی ۳ انتخاب که قبلا توضیح داده شد و با سه موجودیت دانشجو،

درس و استاد در ارتباط می باشد ، می توان با یک موجودیت ضعیف به صورت زیر نشان داد.

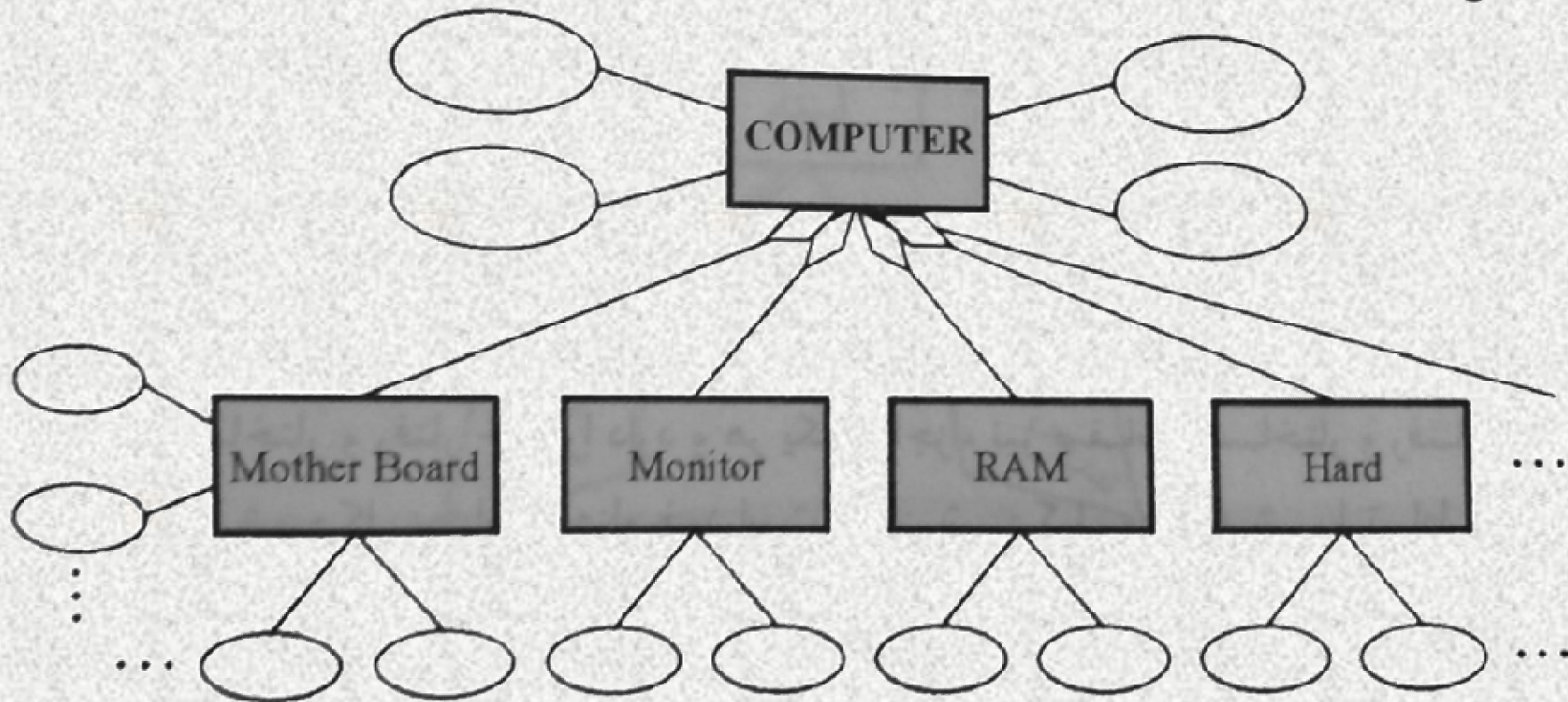


تجزیه و ترکیب

- تجزیه یا جدا سازی یعنی یک شی کل را به اجزای تشکیل دهنده ی آن تقسیم کنیم. شی کل شامل اجزای خود است و ما بین شی کل و اجزایش ارتباط شمول (Contentment) وجود دارد
- این نوع ارتباط در روش ER ارتباط “جزئی است از” (IS-A PART-OF) گفته می شود.



تجزیه و ترکیب – مثال



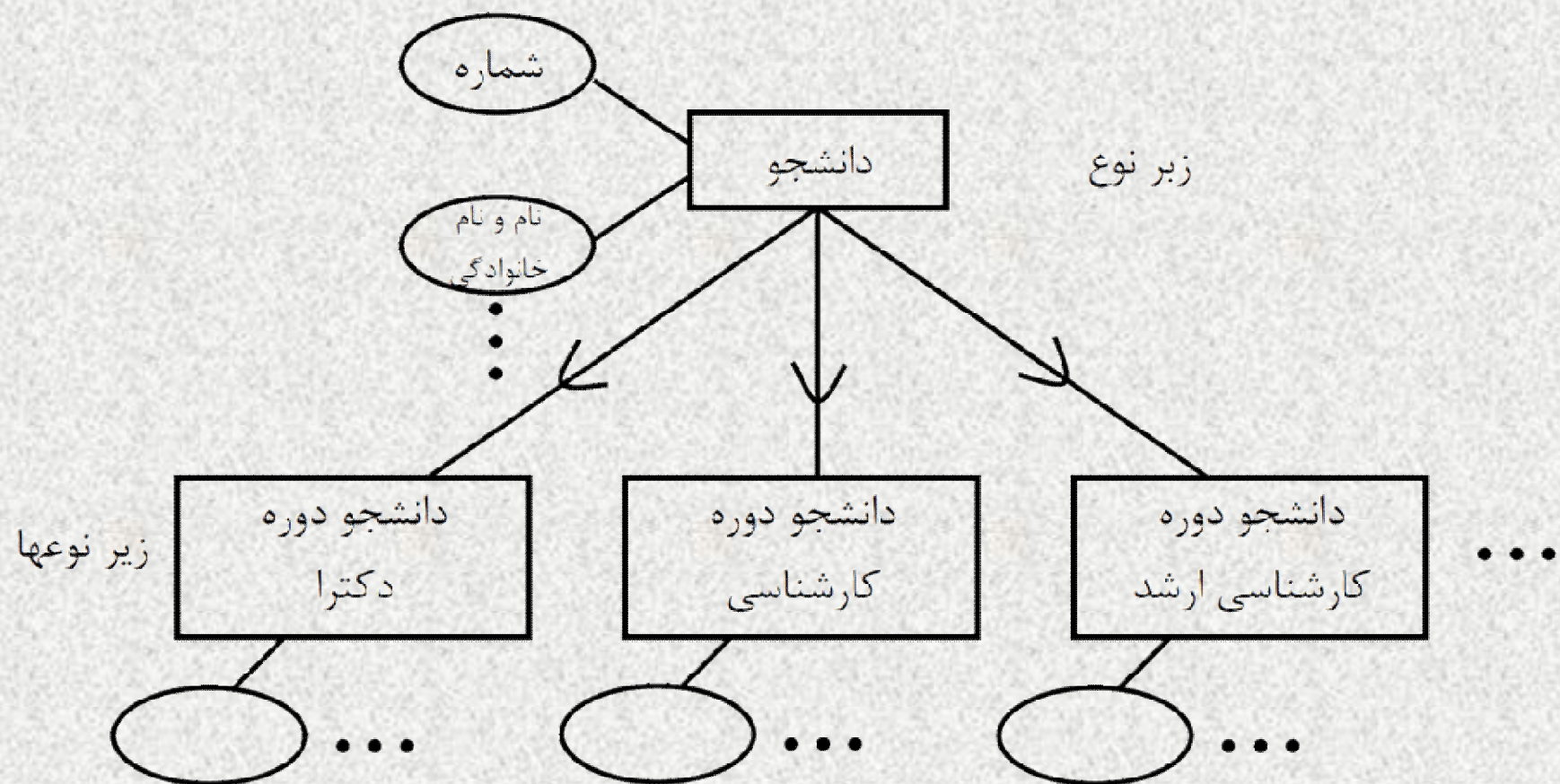
تخصیص

- تخصیص عبارتست از بازشناسی گونه های خاص یک شی براساس یک ضابطه ی مشخص.
- بین هر زیر نوع موجودیت (Sub type) و زیر نوع (Super type) آن ارتباط “هست یک... “ (IS-A) وجود دارد. به بیان دیگر هر زیر نوع گونه ی خاصی است از حداقل یک زیر نوع
- زیر نوع مجموعه ای از صفات دارد که بین تمام زیر نوع های آن مشترک است و بنابراین، هر زیر نوع، صفات زیر نوع خود را به ارث می برد. به علاوه هر زیر نوع ، صفات خاص خود را دارد.

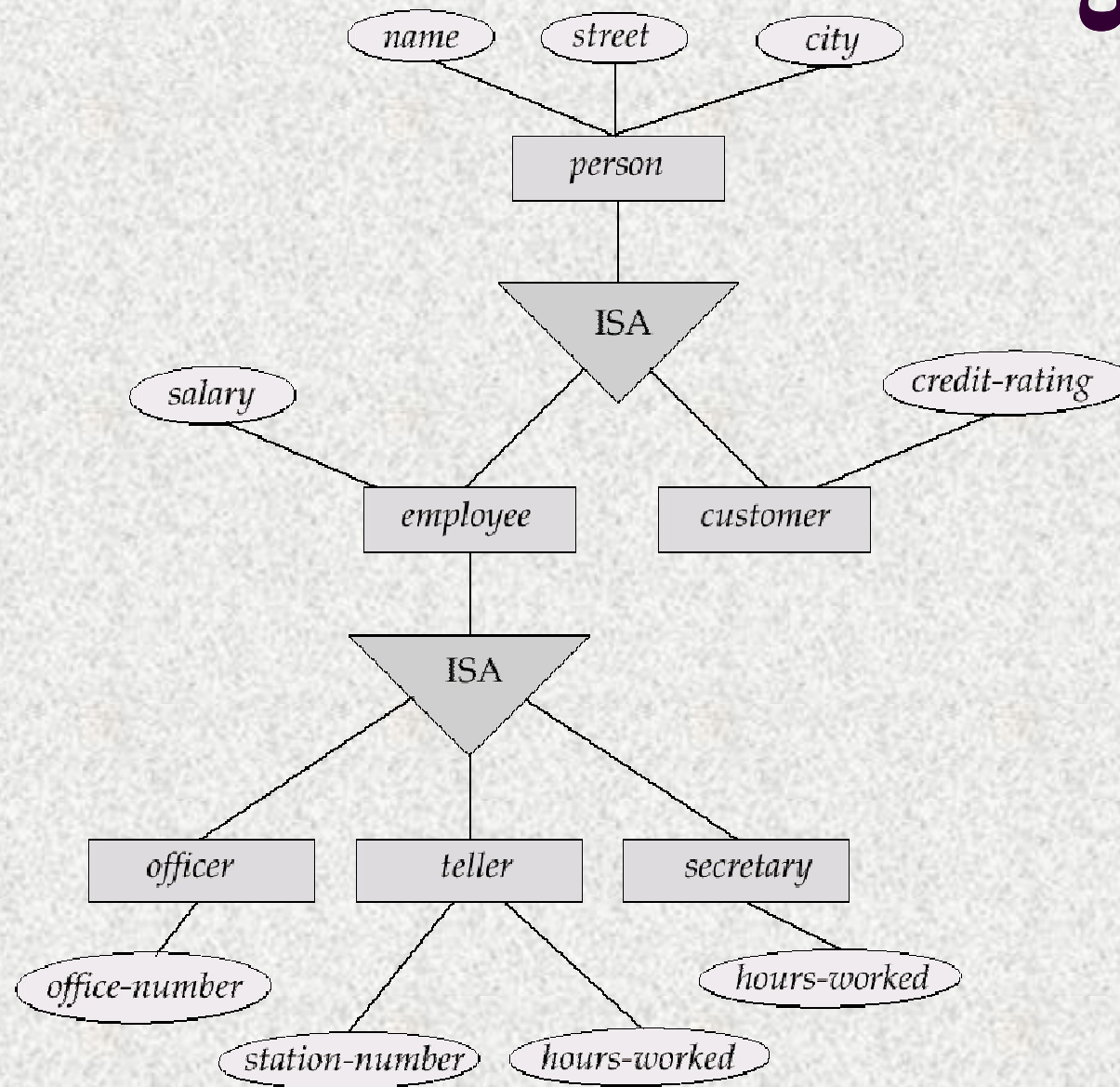
تعمیم

- **تعمیم ، عکس عمل تخصیص است ، به این معنا که با داشتن زیر نوع های خاص ، صفات مشترک بین آنها (از جمله شناسه) را در یک مجموعه صفات برای یک زیر نوع موجودیت در نظر می گیریم، طبعاً این مجموعه صفات مشترک جدا شده را ، دیگر برای هر زیر نوع تکرار نمی کنیم.**
- **از این خاصیت زمانی استفاده خواهیم کرد که در سیستم چندین موجودیت داشته باشیم که دارای صفات مشترکی باشند در این حالت یک زیر نوع ایجاد کرده و صفات مشترک را برای آن در نظر می گیریم و بقیه ی موجودیت ها را به عنوان زیر نوع آن در نظر میگیریم. در این زیر نوع ها دیگر صفات مشترک لحاظ نخواهد شد.**

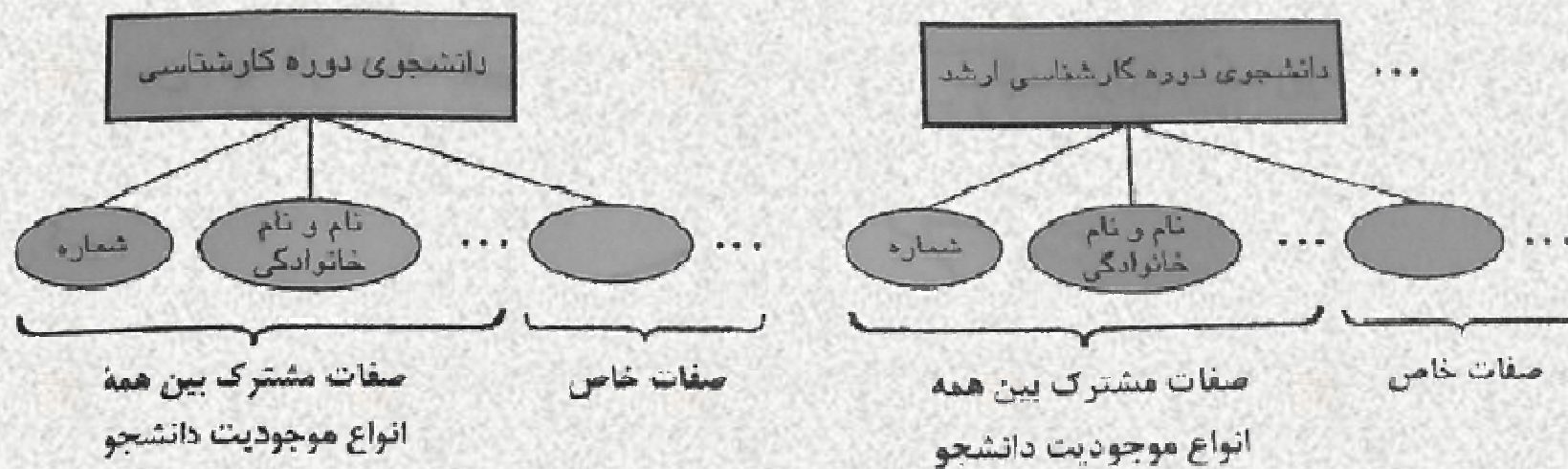
تخصیص – مثال



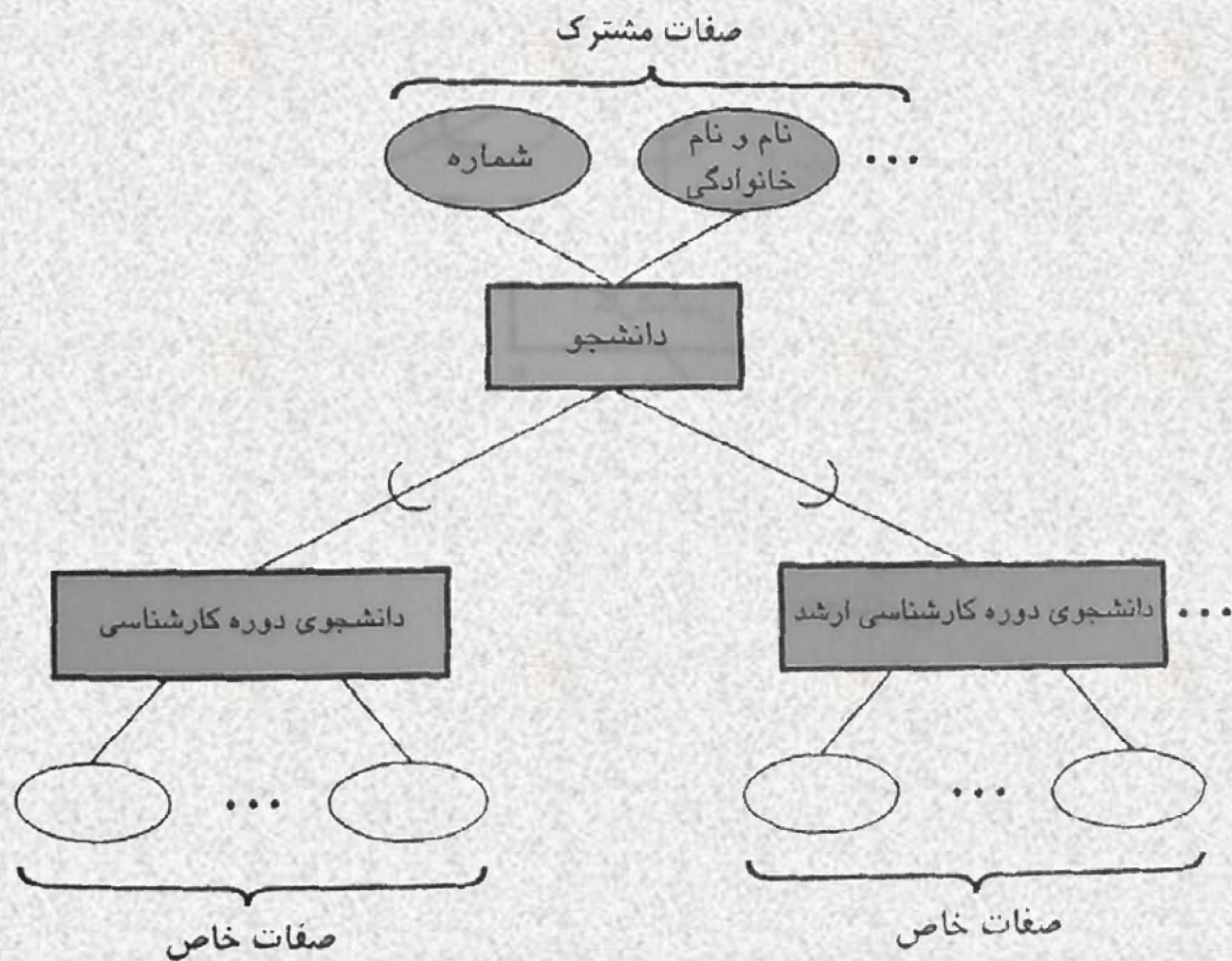
تخصيص – مثال



تعمیم - مثال

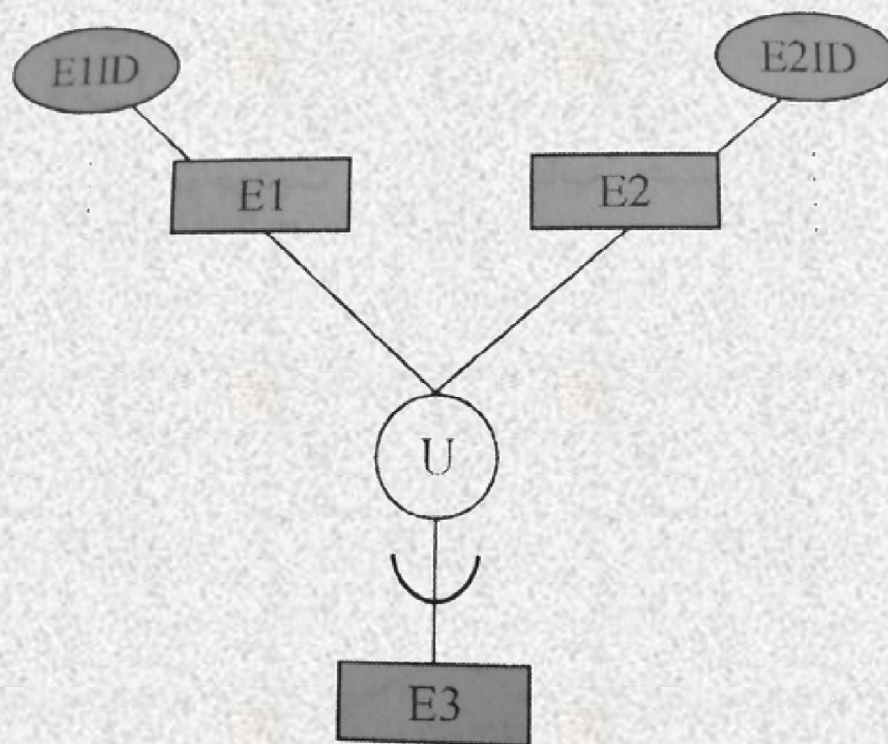


تعمیم - مثال



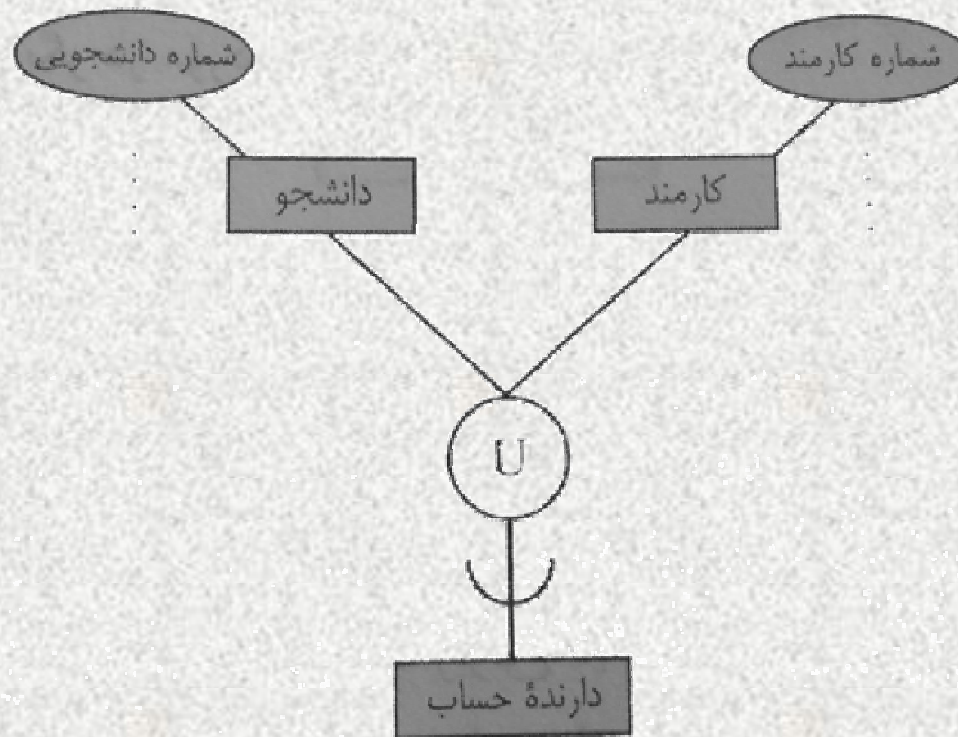
دسته بندی

- ممکن است زیر نوع های یک زیر نوع از یک نوع نباشند. به این زیر نوع ها اصطلاحاً دسته (طبقه) گفته می شود و برای نمایش آن از نماد U استفاده می شود، به صورت شکل زیر نشان داده می شود.



دسته بندی – مثال

- در این مثال دارنده ی حساب از اجتماع کارمند و دانشجو تشکیل شده است یعنی دارنده حساب یا کارمند است یا دانشجو

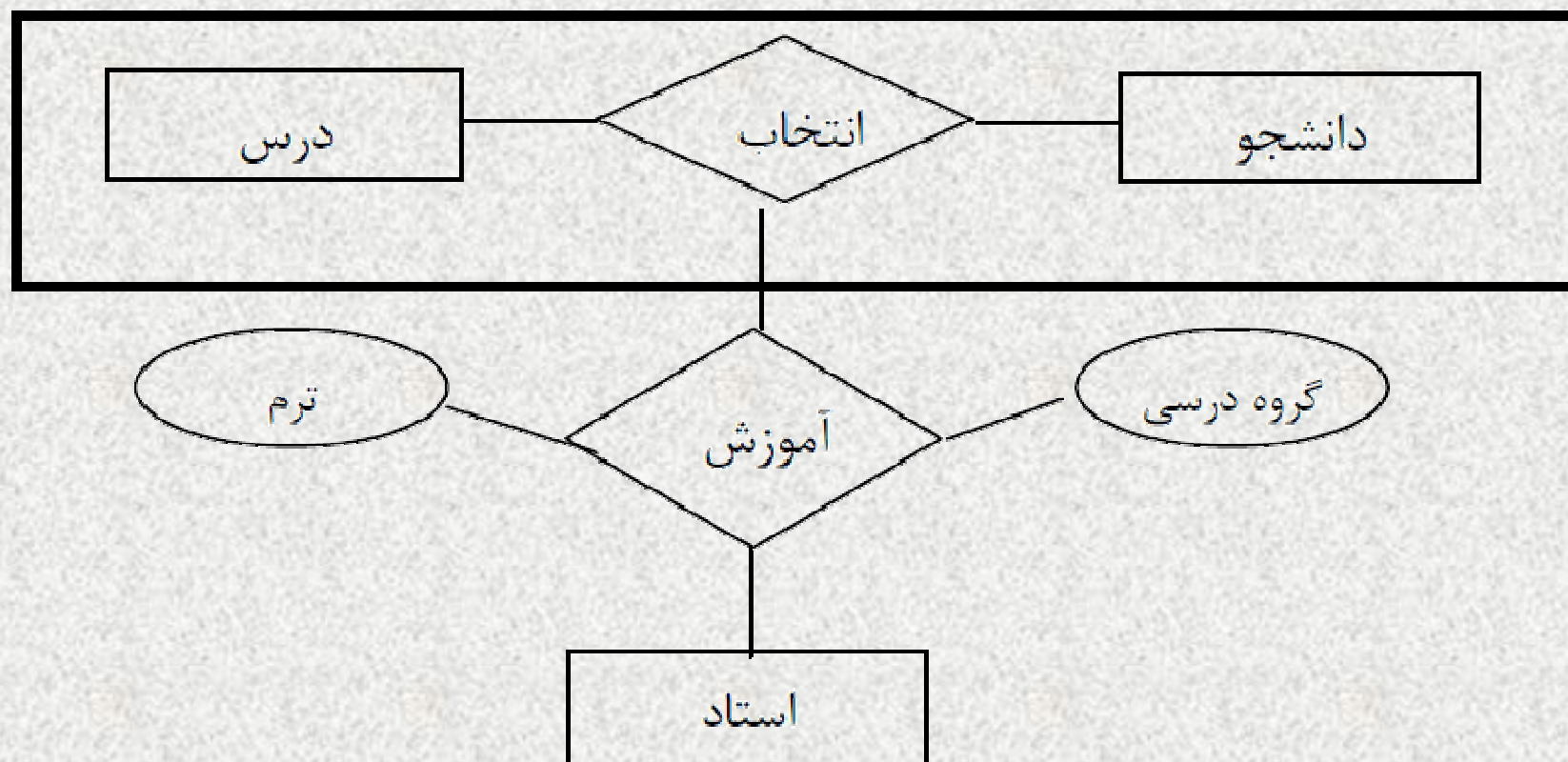


تجمع (Aggregation)

- تجمع یعنی ساختن یک نوع موجودیت جدید و واحد بر اساس دو یا بیش از دو نوع موجودیت، که خود باهم ارتباط دارند
- در واقع مجموعه ای از موجودیتها و ارتباطات را با هم مجتمع کرده و به عنوان یک نوع موجودیت واحد در نظر می گیرند و این نوع موجودیت خود می تواند با نوع موجودیت دیگری ارتباط داشته باشد
- زمانی از تجمع استفاده می شود که بخواهیم ارتباطی را بین ارتباط ها بیان کنیم و یا بخواهیم ارتباطات افزونه را کم کنیم.

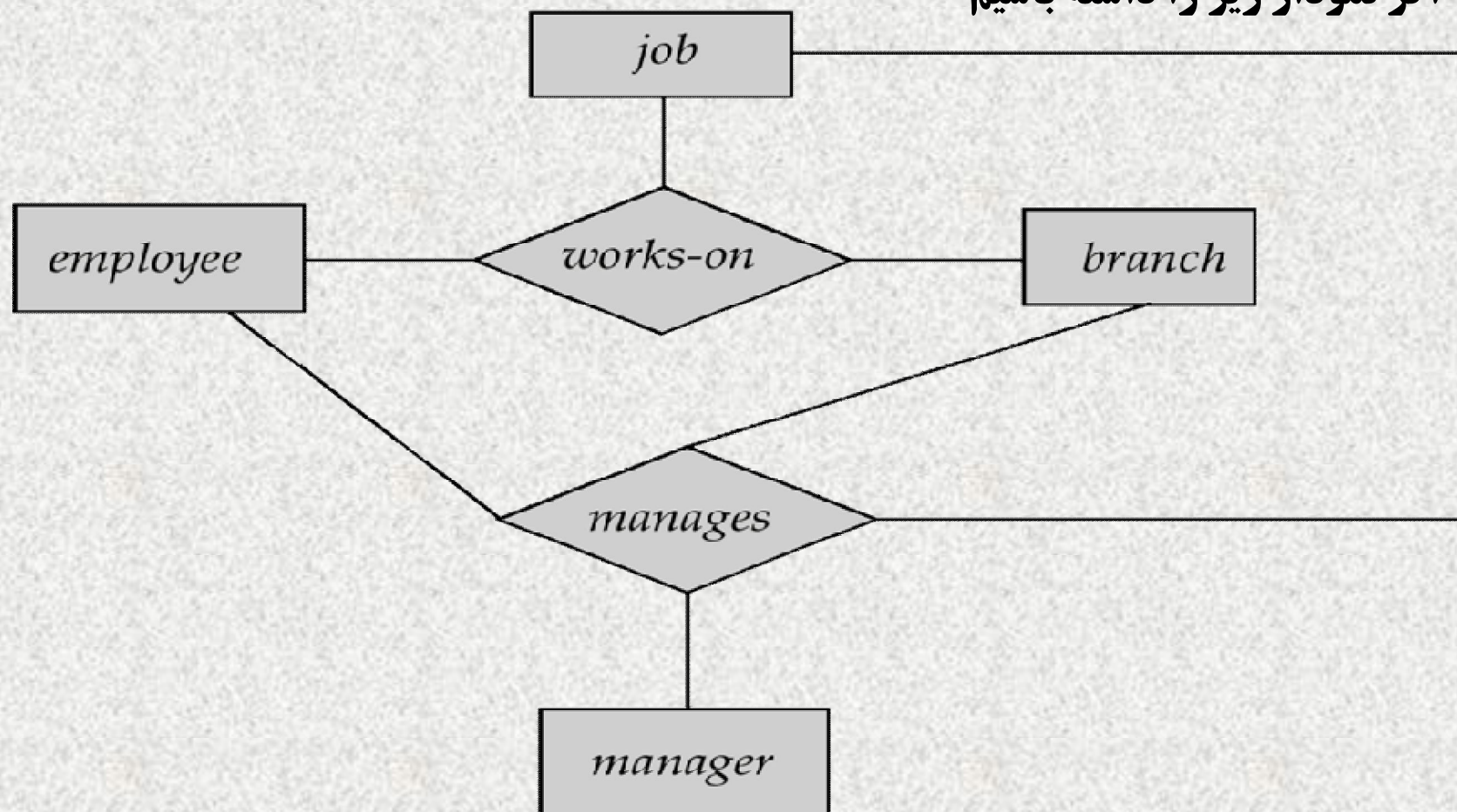
تجمع-مثال

- ارتباط بین نوع موجودیتهای دانشجو ، درس و استاد را می توان همانند شکل زیر مدلسازی نمود



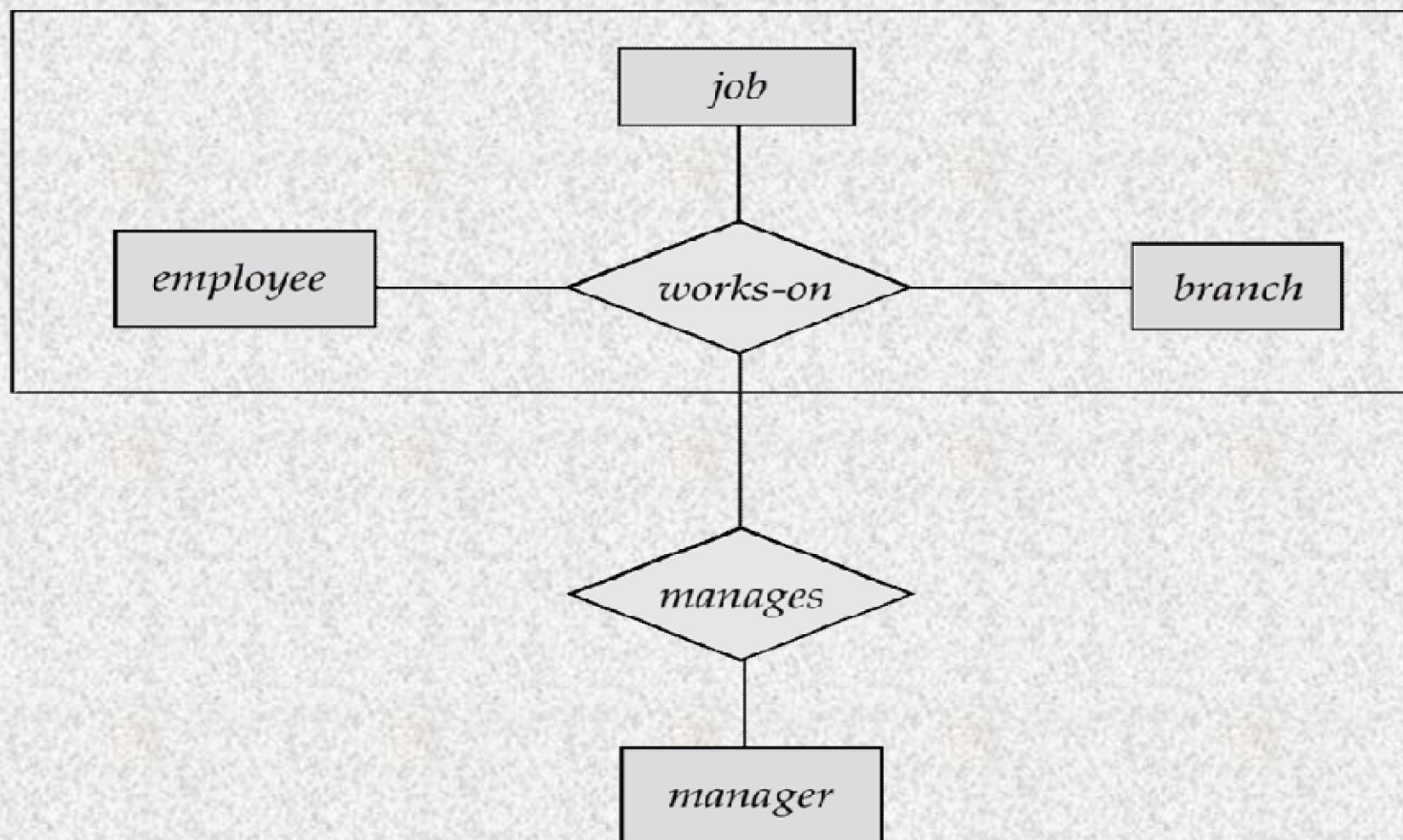
تجمع—مثال

■ اگر نمودار زیر را داشته باشیم

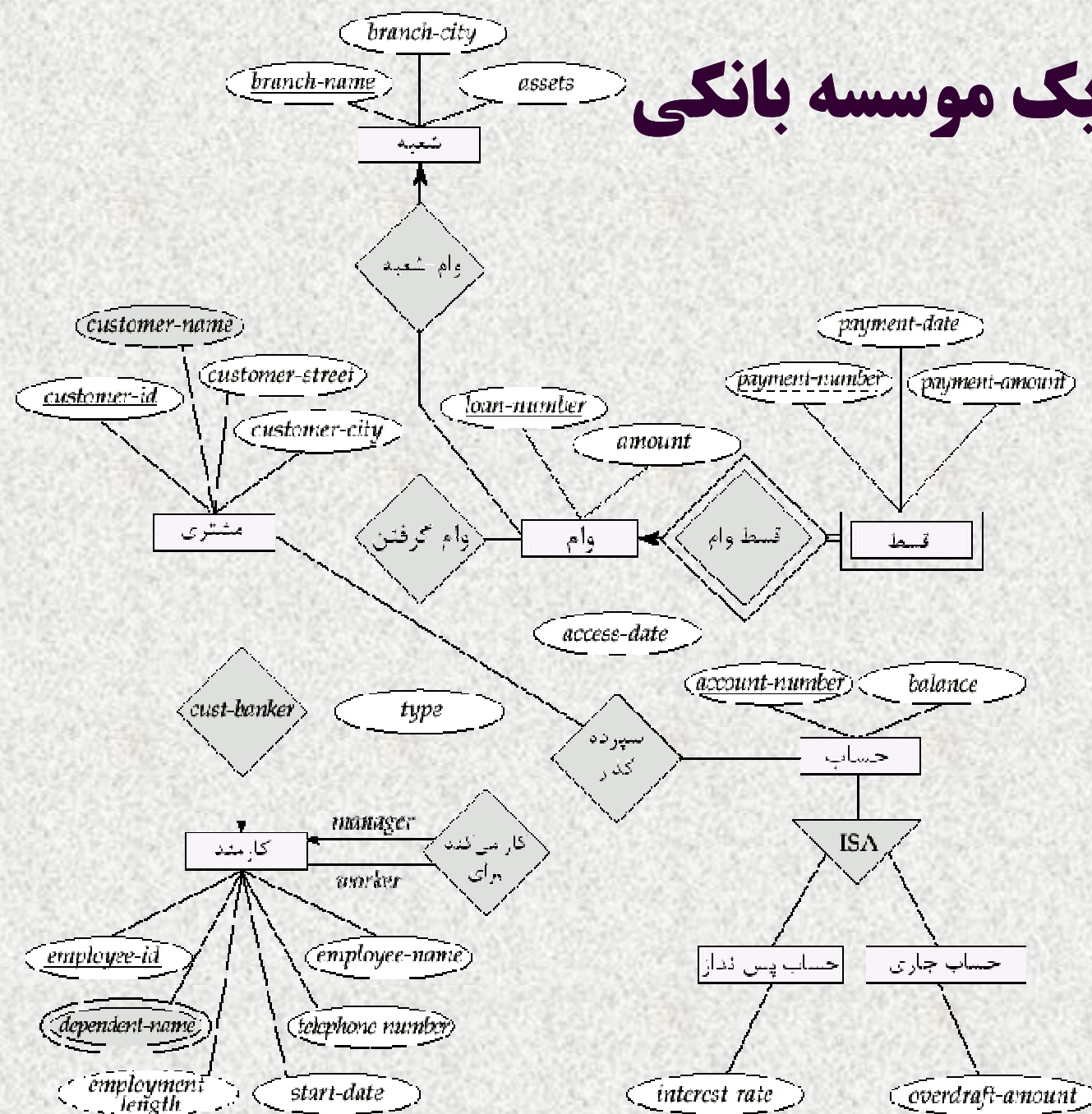


تجمع – مثال

■ با تجمع نمودار زیر را خواهیم داشت



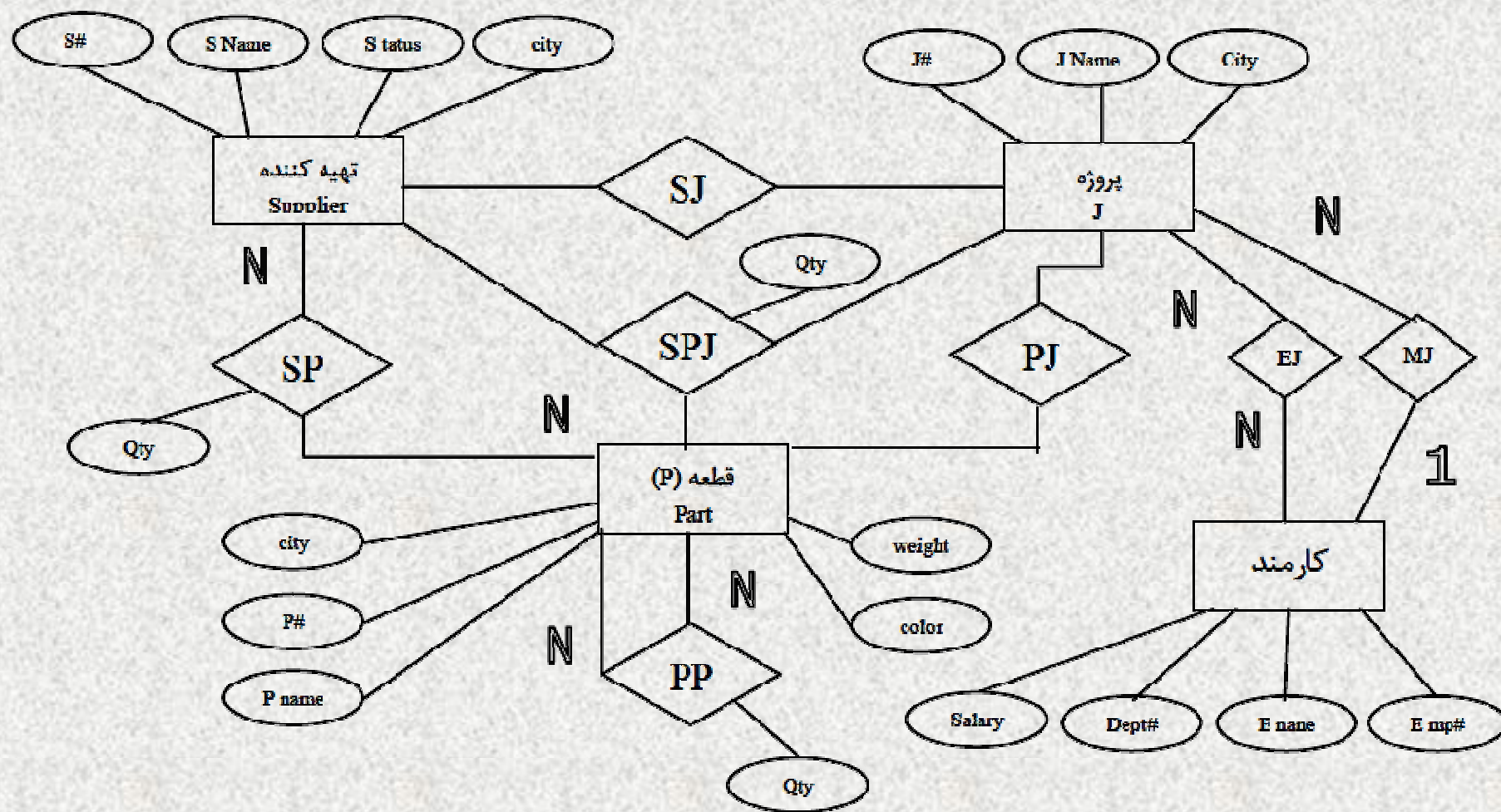
نمودار ER یک موسسه بانکی



مثال کلی – محیط عملیاتی

■ سازمان یا شرکتی را در نظر می گیریم که پروژه هایی را در دست اجرا دارد. در پروژه ها از قطعاتی در کار ساخت استفاده می شود و تهیه کنندگانی این قطعات را تأمین می کنند . قطعات در پروژه ها استفاده می شوند . هر تهیه کننده در یک شهر دفتر دارد. هر قطعه می تواند در ساخت قطعه دیگر نیز بکار رود . کارمند مدیر پروژه است و یا در پروژه کار می کند.

مثال کلی-نمودار ER محیط عملیاتی



مثال – محیط عملیاتی

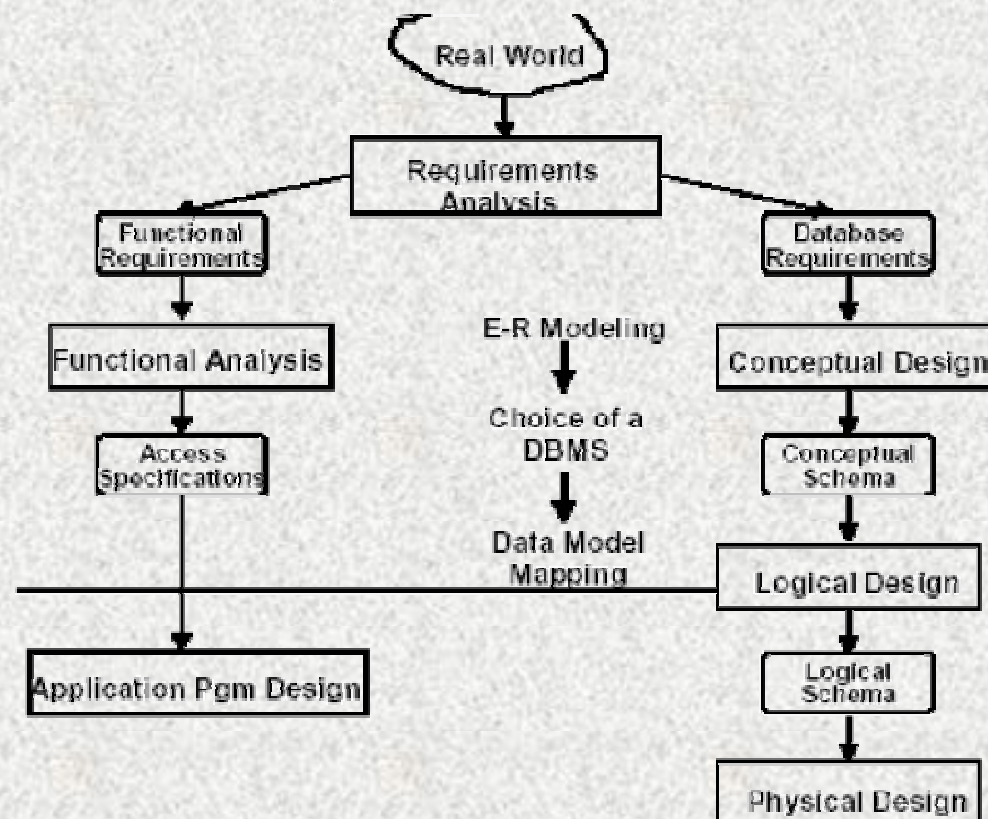
- ارتباط ممکن است مابین بیش از دو موجودیت باشد (SPJ). اطلاعاتی که از این ارتباط بین سه موجودیت به دست می آید همیشه لزوما همان اطلاعاتی نیست که از ارتباط دو به دوی موجودیت ها بدست می آید.

■ مثال

۱. تهیه کننده S1 قطعه P1 را تهیه می کند
 ۲. قطعه P1 در پروژه J1 به کار رفته است
 ۳. تهیه کننده S1 برای پروژه J1 قطعه تهیه کرده است
 ۴. تهیه کننده S1 قطعه P1 را برای استفاده در پروژه J1 تهیه کرده است
- همیشه نمی توان از اطلاع ۲ و ۳ اطلاع ۴ را نتیجه گرفت

طراحی پایگاه داده

- طراحی یک پایگاه داده مستلزمی مرحله‌ای است که در هر مرحله فعالیت‌هایی انجام میشود شکل زیر نمودار ساده شده مراحل طراحی پایگاه داده‌ها را نشان می‌دهد



طراحی پایگاه داده

- برای طراحی پایگاه داده ها ابزارهای مختلفی ارائه شده اند . این ابزارها به طراح پایگاه داده کمک می کنند تا در مراحل مختلف مدلسازی و طراحی پایگاه داده، تصمیم گیری مناسب را انجام دهد.
- مزایای استفاده از این ابزارها :
- سادگی فرایند ایجاد نمودارها
- تولید خودکار دستورات DBMS مثلا SQL
- امکان مستند سازی هر موجودیت ، صفت خاصه ، رابطه و محدودیت
- مثال: ER Studio