فصل اول

گذشته و اینک

پنج نسل تکنولوژی ذخیره و بازیابی اطلاعات

خصوصیات نسل اول –(۱۹۵۷-۱۹٤۸)

نسل فایلهای ساده ترتیبی

در این نسل رسانه خارجی معمو لا نوار بوده است .این نسل را می توان نسل بی نرم افزار واسط نیز نامید .مشخصات کلی این نسل عبارتند از:

۱. ساختار فایلها ترتیبی است.
۲. ساختار فیزیکی همان ساختار منطقی است.
۳. تنها روش پردازش فایلها، پردازش یکجا (دسته ایی) است.
۴. نسخه های متعددی از یک فایل نگاهداری می شوند.
۵. نرم افزار تنها عملیات ورودی / خروجی را انجام می دهد و نرم افزار واسطی برای مدیریت پردازش فایلها وجود ندارد.
۶. طراحی ساختار فیزیکی فایلها هم، بر عهده کاربر است (ساختار منطقی فایل عملاً همان ساختار فیزیکی است) و کاربر در برنامه اش احکام و دستورات مربوط به این کار را جای می دهد.
۷. هرگونه تغییر در ساختار داده و یا رسانه های ذخیره سازی سبب بروز تغییر در برنامه و بازنویسی و کامپایل آن می شود.
۸. داده ها برای کاربرد خاصی طراحی و سازماندهی می شوند.
۹. استفاده از یک داده در چندین کاربرد، نادر است.
۱۰. تکرار در ذخیره سازی داده ها در بالاترین حد است.

خصوصیات نسل دوم –(۱۹٦٦-۱۹۵۸)

نسل شیوه های دستیابی

این نسل را باید نسل شیو ههای دستیابی (Access Methods) نامید .مهمترین ویژگی این نسل را باید پیدایش نرم افزارهای موسوم به «شیوه های دستیابی» و همچنین ایجاد رسانه های بادستیابی مستقیم (یعنی دیسک) دانست. نرم افزار شیوه دستیابی، نرم افزاری است که به جنبه های فیزیکی محیط ذخیر ه سازی و عملیات در این محیط می پردازد .به نحوی که دیگر برنامه کاربر نیازی به پرداختن به این جنبه ها را ندارد.

مشخصات این نسل عبارتند از:

۱. نرم افزار واسط، برای ایجاد فایلها با ساختارهای گوناگون بین برنامه کاربردی و محیط ذخیره سازی وجوددارد.
۲. امکان دستیابی ترتیبی و مستقیم به رکوردها (نه فیلدها) وجود دارد.
۳. پردازش در محیطهای بلادرنگ (در زمان) بر خط و یکجا (بسته به نوع سیستم عامل) میتواند انجام شود.
۴. ساختار فیزیکی و ساختار منطقی فایلها از یکدیگر جدا هستند، ولی نه تا حدی که برنامه های کاربردی از محیط فیزیکی ذخیره سازی مستقل شوند.
۵. تغییر در رسانه های ذخیره سازی بر روی برنامه های کاربردی تاثیری چندان ندارد.
۶. هنوز امکان بازیابی بر اساس چندین کلید وجود ندارد.

۷. ایمنی و حفاظت داده ها مطرح بوده، ولی روشهای تامین ایمنی و حفاظت ابتدایی هستند.
۸. داده ها همچنان برای کاربردهای خاصی طراحی و ذخیره سازی می شوند.
۹. تکرار ذخیره سازی هنوز در حد نسبتاً بالایی وجود دارد.
۱۰. برای پیاده سازی فایل با ارتباط خاصی بین انواع رکوردها (مثلاً ارتباط سلسله مراتبی) خود برنامه ساز باید ارتباطات را در برنامه اش بسازد.
۱۱. هر چند شیوه های دستیابی بعنوان ارتباط نرم افزار واسط عمل می کنند ولی نرم افزاری برای مدیریت داده ها (خصیصه نسل سوم) وجود ندارد.

خصوصیات نسل سوم –(۱۹۷۳-۱۹٦۷)

نسل سیستم مدیریت داده ها

در این نسل نرم افزاری کاملتر از نرم افزار دستیابی به عنوان واسط بین برنام ه های کاربردی و فایلهای محیط فیزیکی طراحی و ایجاد شد . در این نسل دریافتند که می توان برنامه های کاربردی را در قبال رشد فایلها (File Growth) مث ً لا افزودن یک فیلد به یک نوع رکورد از یک فایل مصون نگاه داشت . تا قبل از این نسل برنامه های کـاربردی فقـط در قبـال تغییـرات سخت افزاری و رشد کمی فایلها (یعنی افزایش حجم داده های فایل) مصون بودند . مشخصات کلی این نسل عبارتند از:

۱. نرم افزار نسبتاً پیچیده ای بنام سیستم مدیریت داده ها، واسط است بین برنامه های کاربردی و محیط فیزیکی ذخیره سازی.
۲. فایلهای منطقی متعددی می توانند از داده های فیزیکی مشترک بهره برداری کنند، و این فایلها میتوانند بهم مرتبط باشند.
۳. میزان تکرار ذخیره سازی کاهش یافته است.
۴. داده های مشترک در کاربردهای متنوع بکار می روند.
۵. صحت داده های ذخیره شده تا حدی تامین میشود.
۶. نشانی دهی به داده ها در سطح فیلد یا گروهی از فیلدها امکان پذیر است.
۷. تسهیلاتی برای پردازش فایلها پیش بینی شده است.
۸. بازیابی بکمک چند کلید امکان پذیر است.
۹. ترکیبی از انواع ساختارهای فایل بکار گرفته میشود.

خصوصیات نسل چهارم –(۱۹۸۰-۱۹۷٤)

نسل جهش در تکنولوژی ذخیره و بازیابی اطلاعات (نسل DBMS)

این نسل از اواخر دهه ۶۰ آغاز شد و هم اکنون نیز ادامه دارد . مهمترین خصیصه این نسل مستقل شدن برنامه های کاربردی(Application Program)از جنبه ها و خصوصیات محیط فیزیکی ذخیره سازی است که اصطلاحاً به آن استقلال داده یی فیزیکی (Physical Data Independence) می گویند . در بقیه این کتاب در واقع ویژگی های DBMS را شرح می دهیم لذا در اینجا فقط بعضی از ویژگی های اصلی آن را به صورت خلاصه بیان می کنیم.

۱ – کاربران در یک محیط انتزاعی(Abstractive) و مبتنی بر یک ساختار داده یی تجریدی کار می کنند .بدین ترتیب برنامه های کاربردی از داده های محیط فیزیکی کاملاً مستقل می شوند .اصطلاح داده افزار(Data Ware) نیز ناظر به همین جنبه انتزاعی می باشد .

۲– امکان کنترل متمرکز روی تمام داده های عملیاتی وجود دارد و یکی از مزایای این کنترل متمرکز کاهش میزان افزونگی (Redundency) در ذخیره سازی داده هاست.

۳– نرم افزار پیچیده و جامع موسوم به سیستم مدیریت بانک اطلاعاتی(DBMS یعنی Data Base Management System)

واسط بین برنامه های کاربران و محیط داخلی و فیزیکی ذخیره سازی است . این نرم افزار امکان می دهد تا کاربران در یک محیط انتزاعی کار کنند و در عین حال به داده های ذخیره شده دستیابی داشته و عملیات موردنظر خود را انجام دهند.

۴– سرعت دستیابی به داده ها بالا می باشد .ایمنی داده ها زیاد است و امکان استفاده اشتراکی از داده ها وجود دارد.

۵– در این نسل مفهوم چند سطحی بودن ساختار داده یی و معماری چند سطحی ذخیره سازی مطرح و به تدریج بسط یافت این سطوح از پائین به بالا عبارتند از ساختار فیزیکی بانک ساختار داخلی بانک ساختار ادراکی بانک و ساختار خارجی بانک (این سطوح را بعداً شرح می دهیم).

۶– رشدپذیری یکی دیگر از ویژگی های این سیستم است . این رشد پذیری به خاطر وجود معماری چند سطحی و نیز استقلال برنامه های کاربردی از ساختار ذخیره سازی و استراتژی دستیابی تامین می شود. در واقع باید گفت اصلی ترین تفاوت این نسل با نسل های قبلی وجود حصاری نفوذناپذیر به نام سیستم مدیریت بانک اطلاعاتی یا DBMS است که هرگونه دستیابی به داده ها می بایست از طریق آن انجام شود.

برای راهبری این نرم افزار DBMS ، مثل اطلاعات امنیتی در باره اجازه استفاده کاربران و همچنین انجام تغییرات به دو نوع نیروی انسانی نیاز است:

الف) مدیر بانک اطلاعات (Data Base Administrator = DBA) که مسئولیت تصمیم گیری و طراحی موارد مذکور را بر عهده دارد.

ب) برنامه ساز بانک اطلاعات(Data Base Programmer = DBP) که تصمیمات مدیر را پیاده سازی می کند. تنها کسانی که می توانند دور از چشم DBMS به داده ها دسترسی داشته. باشند مدیر و برنامه سازان مجاز بانک اطلاعات هستند

DBMS در واقع انقلابی در بانکهای اطلاعاتی به شمار میآید.

در اواخر این نسل دو تحول دیگر در تکنولوژی بانک اطلاعاتی پدید آمد : یکی طراحی و ایجاد بانکهای اطلاعاتی توزیع شده (Distributed Data Base) تحت شبکه ها (که بعداً شرح می دهیم) و دیگری طراحی و ایجاد سیستم های مدیریت بانک اطلاعاتی برای کامپیوترهای شخصی.

- مستقل شدن برنامه های کاربردی از جنبه ها و خصوصیات محیط فیزیکی ذخیره سازی است.
- داده ها و ارتباطات بین آنها را در کادر یک ساختار داده یی انتزاعی می بینند و بکمک احکام زبانی خاص، عملیات مورد نظر خود را انجام می دهند و بدین ترتیب برنامه های کاربردی از داده های محیط فیزیکی کاملاً مستقل می شوند.
- امکان کنترل متمرکز روی تمام داده های عملیاتی است. از فواید آن کاهش میزان افزونگی داده هاست.
- نرم افزار پیچیده و جامع موسوم به سیستم مدیریت بانک اطلاعاتی وجود دارد.
- مفهوم چند سطحی بودن ساختار داده یی و معماری چند سطحی ذخیره سازی مطرح گشت.
- رشد پذیری بانک اطلاعاتی با توجه به معماری چند سطحی و استقلال برنامه های کاربردی
- طراحی و ایجاد بانکهای اطلاعاتی توزیع شده (برای استفاده در شبکه هایی کوچک و بزرگ)
- طراحی و ایجاد سیستم های مدیریت بانک اطلاعاتی برای استفاده از محیط کامپیوترهای شخصی

خصوصیات نسل پنجم –(۱۹۸۱-۰۰۰)

نسل بانک معرفت یا پایگاه شناخت

این نسل به نسلKnowledge Base یا پایگاه شناخت (پایگاه دانش) معروف است. در این تکنولوژی، ضمن استفاده از تکنولوژی بانکهای اطلاعاتی با بهره گیری از منطق صوری(Formal Logic) ،سیستم های خبره (Expert System)، مفاهیم هوش مصنوعی (Artificial Intelligence = AI) و پردازش زبان طبیعی(Natural Language) سیستمی طراحی و ایجاد می شود که قادر به استنتاج منطقی از داده های ذخیره شده است.

تعریف Knowledge Base : مجموعه ای از واقعیت های ساده و قواعد عام که نشاندهنده بخشی از جهان واقعی (Real World) باشد .

منظور از جهان واقعی محیطی است یعنی که در رابطه با آن می خواهیم اطلاعاتی داشته باشیم و گاه به آن محیط عملیاتی نیز گفته میشود. تفاوت اساسی بین بانکهای اطلاعاتی و بانکهای معرفت در این است که بانک معرفت حاوی مجموعه ای است از واقعیت های ساده و قواعد عام که به طور صریح بیان شده باشند، همراه با تعداد نسبتاً کمی از قواعد عام که به طور ضمنی بیان میشوند.

از خصوصیات اساسی این نسل وجود سیستمی به نام «سیستم بانک معرفت» یا KBS مخفف Knowledge Base System است که خود مجموعه ای است از امکانات نرم افزاری و سخت افزاری که مسئولیت ذخیره سازی معرفت، تامین امنیت(Security) و جامعیت (Integrity) بانک و نیز تامین نیازهای کاربران را بر عهده دارد .

اضافه کردن بانک دانش (Knowledge – Base) و قدرت استنتاج منطقی به بانک، با نامهای مختلفی مثل بانک اطلاعات پویا، بانک اطلاعات خبره، بانک اطلاعات استنتاجی (Deductive Database)، بانک اطلاعات بازگشتی (Recursive Database) بانک اطلاعات استنباطی(Referential Database) و غیره مطرح می شود .

مثلاً اگر در بانک اطلاعات دانشگاه داشته باشیم «مدرک آقای امیری دکتری است» و «تخصص آقای امیری شبکه های کامپیوتری است»

می توان خود به خود نتیجه گرفت که آقای امیری « می تواند درس شبکه های کامپیوتری را تدریس کند» بنابراین با استفاده از این روش، اطلاعات جدیدی مرتباً کشف شده و بر داده های بانک افزوده می گردد . همچنین می توان بسیاری از اطلاعات را ذخیره نکرد و در موقع لزوم آنها را استنتاج نمود.

فصل دوم

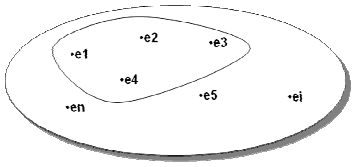
اطلاع و داده

تعریف داده

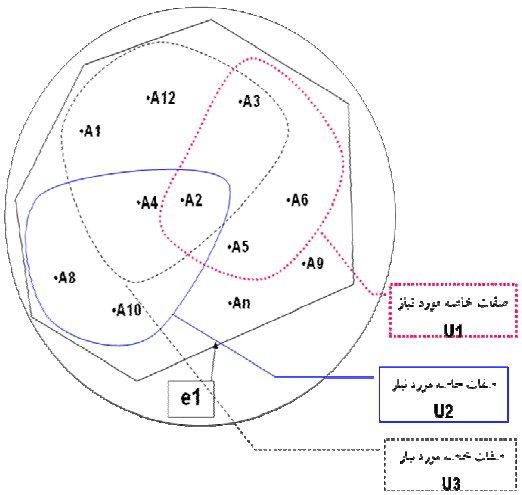
- نمایش واقعیات، پدیده ها، مفاهیم، یا معلومات، بصورتی صوری و مناسب برای برقراری ارتباط، تفسیر، یا پردازش توسط انسان یا امکانات خودکار (ANSI)
- هر نمایشی، اعم از کاراکتری یا کمیت های آنالوگ، که به آن معنایی منتسب است و یا باید منتسب شود. (ANSI)
- داده عبارت است از مقادیر صفات خاصه انواع موجودیت ها. (فرهنگ لغات انگلیسی)

آشنایی با چند اصطلاح

- نوع موجودیت: مفهوم کلی پدیده، شیئی، فرد، که درمورد آنها می خواهیم "اطلاع " داشته باشیم.
- صفت خاصه: ویژگی جداساز یک نوع موجودیت، از نوع دیگر
- مجموعهٔ انواع موجودیت ها: انتخاب عده ایی از موجودیتهای موجود یک محیط عملیاتی بر طبق نظر طراح بانک اطلاعاتی مجموعه فوق را تشکیل میدهد.



مجموع صفات خاصه موجودیت e۱ و صفات خاصهٔ مورد نیاز کاربران



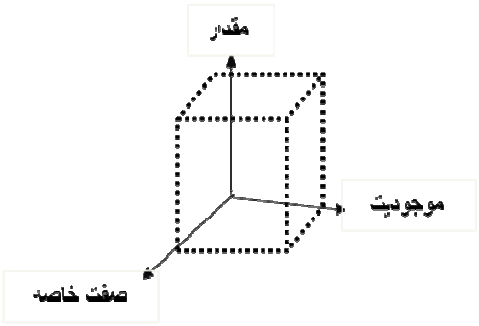
تعریف اطلاع

- معنایی که انسان به داده منتسب میکند، از طریق قرار دادی شناخته شده ای که در نمایش داده بکار رفته اند. (تعریف ANSI)
- اطلاع و داده نوعاً از یکدیگر متمایزند. اطلاع دارای خاصیت ارتباط دهندگی و انتقال دهندگی است. در حالیکه داده چنین خاصیتی ندارد. داده صرفاً نمایش رویدادها و پدیده ها است. (تعریف E.D.DEVAN)
- هنگامیکه اسم صفت خاصه و مقدار منسوب به آن در دست باشند، اطلاعی در موجودیت حاصل میشود. مقدار= اسم صفت خاصه

مثال : موجودیت دانشجو می تواند دارای صفات خاصه : نام نا م خانوادگی سال تولد معدل ترم پیش باشد و مقادیر این صفات خاصه برای یک دانشجوی خاص برابر است با: علی- اکبر- ۱۳۵۲ - ۱۶

نکته : هر صفت خاصه دارای دو مولفه می باشد الف)اسم صفت خاصه ب) مقدار صفت خاصه که در صورت وجود این جفت مولفه «اطلاع» حاصل می گردد.

بطورکلی یک فقره اطلاع را میتوان بصورت نقطه ای در یک فضا سه بعدی نمایش داد به این فضا سه بعدی، فضای اطلاعاتی می گوییم. مثلاً برای دانشجو(نام = علی) یک فقره اطلاع است.



فصل سوم

موجودیت ها و ارتباطات بین آنها

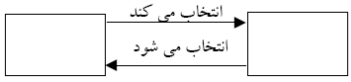
ارتباط:

هر نوع ارتباط یک معنای مشخص دارد و با یک نام بیان می شودو نیز می توان گفت که هر نوع ارتباط ،عملی است که بین موجودیتها وجود دارد.

انواع موجودیت های محیط عملیاتی با یکدیگر ارتباط دارند که معمولاً با یک عبارت فعلی همراه است . این ارتباطات که هر یک سمانتیک خاص را دارد باید شناسایی شده و در پایگاه ذخیره شوند.

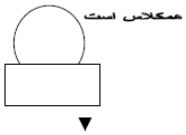
مثال: -دانشجو درس را انتخاب می کند

-درس توسط دانشجو انتخاب می شود



-بین دو موجودیت می تواند بیش از یک ارتباط متفاوت با معنای (سمانتیک) متفاوت وجود داشته باشد.

ارتباط ممکن است بین یک نوع موجودیت و خودش باشد . مثال :قطعه X در ساخت قطعه Y بکارمی رود .به این نوع ارتباط ،ارتباط بازگشتی (Recursive Relationship) نیز گفته می شود.



ماهیت ارتباط:

تناظر بین عناصر مجموعه نمونه های یک نوع موجودیت، با عناصر مجموعه نمونه های نوع موجودیت دیگر را ماهیت ارتباط گویند.

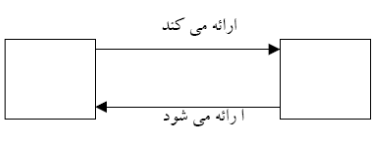
ماهیت ارتباط بین انواع موجودیت ها عبارتند از:

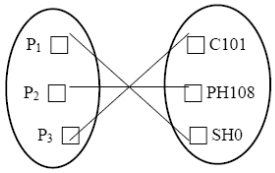
ارتباط	۱ : ۱	تناظر یک به یک
ارتباط	۱ : n	تناظر یک به چند
ارتباط	n : n	تناظر چند به چند

-ماهیت ارتباط بر مبنای قواعدمعنایی (سمانتیک) حاکم بر محیط عملیاتی تعیین می شود.

مثال :رابطه بین موجودیت های استاد و درس را در نظر بگیریم.

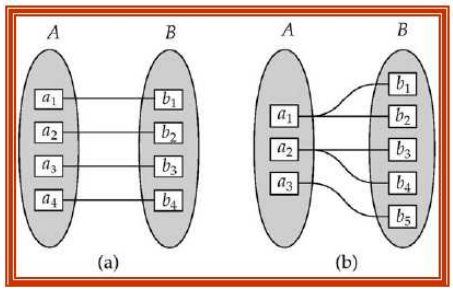
الف : یک استاد حداکثر یک درس را ارائه می کند و هر درس دقیقًا توسط یک استاد ارائه می شود.





ب: (ارتباط یک به چند) (درس - استاد)

یک درس ممکن است توسط بیش از یک استاد ارائه شود ولی یک استاد حداکثر یک درس را ارائه می کند.



ج:(ارتباط چند به چند)

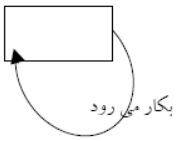
یک استاد ممکن است بیش از یک درس را ارائه کند و یک درس ممکن است توسط بیش از یک استاد ارائه شود.

درجه ارتباط:

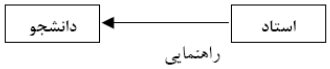
تعداد موجودیتهایی که در آن ارتباط مشارکت دارند درجه ارتباط نامیده می شود.

مثال:

ارتباط درجه ۱ یا رابطه بازگشتی:



ارتباط درجه ۲:



فصل چهارم
نمودار موجودیت- ارتباطات (نمودار ER)

مقدمه :

یک مدل داده ای مجموعه ای از ابزارهای مفهومی برای توصیف داده ها ، ارتباط بین داده ها،معانی داده ها و محدودیت های آنهاست. بعبارتی یک روش تفکر درباره داده ها که به پیاده سازی ربطی ندارد.در یک نگاه کلی میتوان انواع مدلهای داده ای را بصورت زیر نام برد :

- مدلهای مبتنی بر اشیاء Object based Logical Models

داده ها بصورت مجموعه ای از موجودیت ها که نمایش اشیاء در دنیای واقعی میباشند دیده می شوند.از جمله این مدلها مدل داده ای E/Rو مدل شیء گرایی را می توان نام برد.

- مدلهای مبتنی بر رکورد Record based Logical Models

داده ها در قالب رکوردهای ثابت و یا با طول متغییر دیده میشوند.

از انواع دیگر مدلهای داده ای می توان به مدل های داده ای شبه ساختیافته ، مدلهای شبکه ای و مدلهای سلسله مراتبی اشاره نمود.

ازجمله مدلهای منطقی مبتنی بر اشیا می توان **مدل E/R** را نام برد. در سال ۱۹۷٦ یک دانشجوی دکترای کامپیوتر دانشگاه MIT به نام چن (chen) مدلی برای طراحی بانک اطلاعاتی پیشنهاد کرد که مورد توجه عام واقع شد. وی مدل خود را E / R نامید. این مدل در طول زمان پیشرفت کرد و ساختارهای جدیدی به آن افزوده شد. در مدل E / R هر پایگاه داده دارای دو بخش پدیده یا موجودیت و ارتباط می باشد. Chen نه تنها مدل E / R را معرفی نمود بلکه نمودار موجودیت رابطه را نیز مطرح ساخت. نمودار E / R روشی برای نمایش ساختاری منطقی یک بانک اطلاعاتی به روش تصویری است. این نمودارها ابزارهایی راحت و مناسب را برای درک ارتباطات مابین موجودیها فراهم می کنند. (یک تصویر گویا تر از هزار کلمه است.)

در واقع، شهرت و محبوبیت مدل E / R به عنوان روشی برای طراحی بانک اطلاعاتی احتمالاً به روش رسم نمودارهای E / R مربوط می شد تا به جنبه های دیگر آن.

۲ – ۲ – نمایش نموداری E / R

مدل E/R بکمک نموداری تحت نام خود قابل نمایش است که در این نمودار مفاهیم مربوطه با اشکال مشخصی ترسیم میگردند.در ادامه هر یک از این اشکال نشان داده شده اند.

موجودیت:

موجودیت، چیزی است که بصورت متمایز قابل شناسایی باشد. آقای چن موجودیتها را به دو دسته منظم (قوی) و ضعیف دسته بندی کرد.

۱ – **موجودیت منظم(قوی):** موجودیتی است که وجودش وابسته به موجودیت دیگری نیست. مثل موجودیت دانشجو و موجودیت درس که هریک به تنهایی در محیط عملیاتی دانشکده مطرح میباشند.

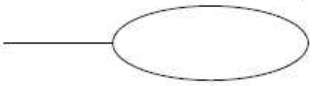
۲ – **موجودیت ضعیف:** موجودیتی است که وجودش وابسته به موجودیت دیگر است. بطور مثال موجودیت اعضاء خانواده کارمند وابسته به موجودیت کارمند می باشد. و یا موجودیت آثار منتشره استاد، موجودیت ضعیف موجودیت استاد است.

در نمودار E / R موجودیتهای قوی بصورت یک مستطیل نشان داده می شوند که محتوای آن در بر گیرنده نام نوع موجودیت مورد نظر است. در موجودیتهای ضعیف مرز مستطیل بصورت دو خطی است.

موجودیت ضعیف موجودیت قوی

-: صفات خاصه Attributes

در نمودار E/R صفات خاصه بصورت بیضی نشان داده می شوند که محتوای آن اسم صفت خاصه مورد نظر را در بر می گیرد و به وسیله خطوط تو پر به موجودیت یا رابطه مربوط متصل می شوند.



در یک تقسیم بندی صفات خاصه را می توان به شرح زیر تقسیم بندی نمود:

الف – صفت خاصه کلید(شناسه موجودیت):

یک یا چند صفت خاصه که در یک موجودیت منحصر به فرد است.

مثال: نوع موجودیت: دانشجو : صفت خاصه کلید: شماره دانشجو

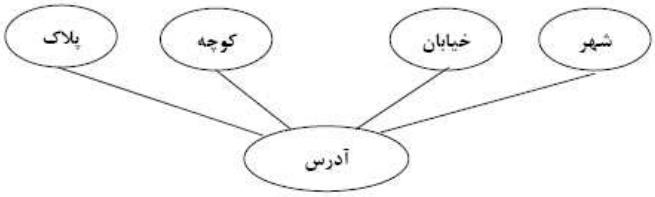
نوع موجودیت: گروه درسی : صفت خاصه کلید: شماره درس، شماره گروه و ترم

برای مشخص کردن کلید در یک موجودیت زیر صفت یا صفات خاصه کلید خط کشیده می شود.

ب – صفت خاصه ساده / مرکب:

- صفت خاصه ساده صفتی است که به اجزاء کوچکتر تجزیه پذیر نباشد.
- صفت خاصه مرکب صفتی است که به اجزاء کوچکتر تجزیه پذیر باشد.

بعنوان مثال: صفت خاصه آدرس می تواند به قسمتهایی نظیر شهر، کوچه، خیابان ... تقسیم شود.



نکته: در مدل بانک اطلاعاتی رابطه ای صفت خاصه مرکب جایی ندارد.

ج- صفت خاصه تک مقداری / چند مقداری Single Valued / Multi Valued

صفاتی که فقط یک مقدار را در هر لحظه از زمان به خود اختصاص دهند به صفات تک مقداری معروفند. به عنوان مثال شماره دانشجویی ،تاریخ تولد تک مقداری هستند.اگر برای یک صفت خاصه چندین مقدار بتواند قرار گیرد صفت خاصه چند مقداری نامیده می شود.

بطور مثال: صفت خاصه مدرک و یا تلفن برای استاد چند مقداری محسوب می شود. زیرا یک استاد می تواند دارای چند مدرک و یا تلفن مختلف باشد. در نمودار E / R برای صفت خاصه چند مقداری از بیضی دو خطی استفاده می شود.

د- صفت خاصه مشتق (استنتاجی) [دارای مقدار محاسبه شدنی]

صفتی است که در موجودیت وجود خارجی ندارد ولی در صورت لزوم می توان آن را بدست آورد.

مثال: موجودیت : استاد، صفت خاصه : تاریخ تولد، صفت خاصه مشتق: سن

در نمودار ER صفت خاصه مشتق با یک بیضی با مرز نقطه چین مشخص می شود.

نکته : تصمیم گیری در مورد صفت مشتق در یک موجودیت بعهده طراح است.

ه- صفت خاصه هیچمقدار پذیر :

هیچمقدار یعنی یک مقدار ناشناخته و یا مقدار غیر قابل اعمال .اگر مقدار یک صفت در یک یا بیش از یک نمونه از یک نوع موجودیت برابر هیچمقدار باشد آن صفت خاصه هیچمقدار پذیر است.

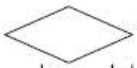
- مثال : شماره تلفن یک نمونه استاد ممکن است در دست نباشد.
- نام استاد در یک برنامه درسی ترم ممکن است هنوز اعلام نشده باشد.

توجه : در نمودار E/R این خاصیت نشان داده نمی شود.

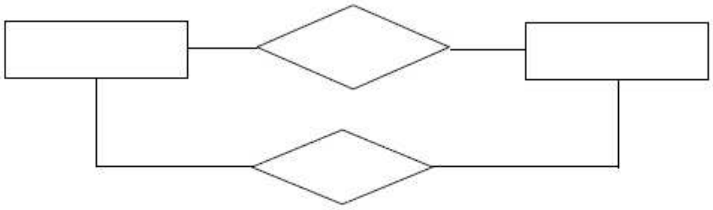
ارتباط:

هر ارتباط بصورت یک لوزی نشان داده می شود که محتوای آن در بر گیرنده نام نوع رابطه مورد نظر است.

در نمودار ER ارتباط با موجودیت ضعیف بصورت لوزی دو خطی نشان داده می شود.

ارتباط با موجودیت ضعیف ارتباط 

عنصرهای هر رابطه (شامل صفات خاصه و موجودیت ها) بوسیله خطوط پر به رابطه مربوطه وصل می شوند. هر خط ارتباطی بین موجودیت و رابطه دارای برچسب ۱ یا n می باشد که ماهیت ارتباط را مشخص می کند.

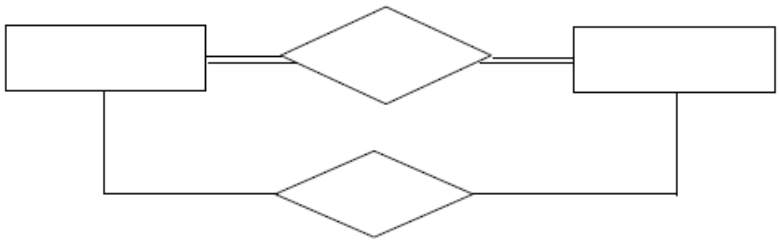


- وضع مشارکت در ارتباط

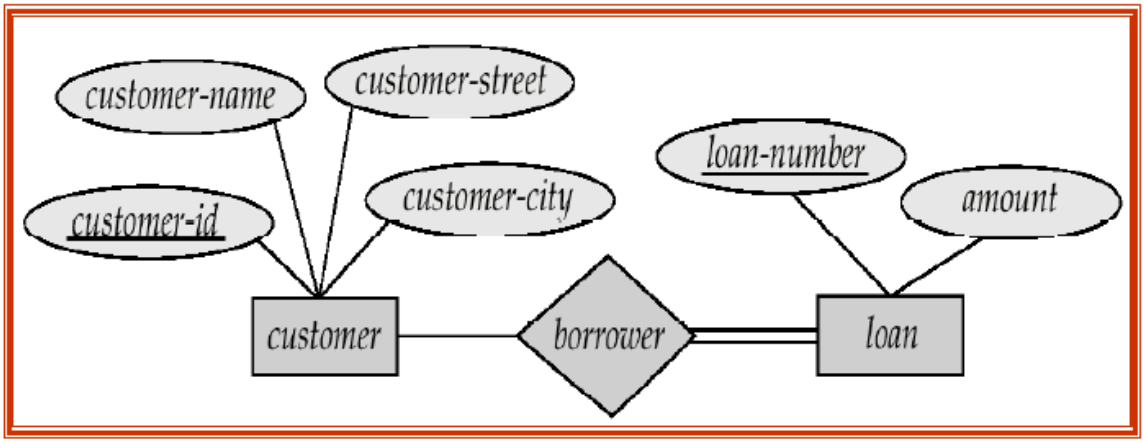
انواع موجودیتهایی که بین آنها ارتباط برقرار است شرکت کنندگان آن ارتباط نام دارند. مشارکت یک نوع موجودیت در یک نوع ارتباط ممکن است الزامی (کامل) یا غیرالزامی (ناکامل) باشد.

مشارکت یک نوع موجودیت در یک نوع ارتباط را الزامی گویند اگر تمام نمونه های آن نوع موجودیت در آن باید در ارتباط شرکت کنند.در غیر اینصورت مشارکت غیر الزامی (اختیاری) است.

مثال : مشارکت دانشجو در ارتباط انتخاب الزامی است ولی مشارکت دانشجو در ارتباط حذف درس الزامی نیست زیرا لزوما همه دانشجویان درس را حذف نمیکنند. در نمودار E/R مشارکت الزامی با دوخط نشان داده میشود.



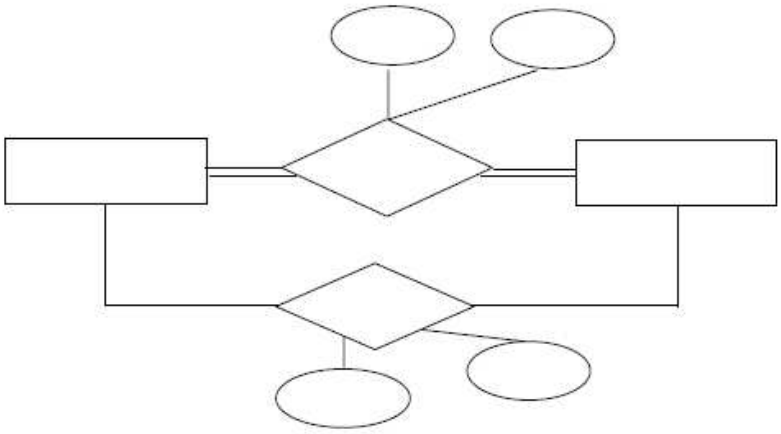
- Total participation: هر موجودیت حداقل در یک رابطه از مجموعه ارتباط مشارکت دارد.
- مثال : مشارکت وام در ارتباط وام گرفتن کامل است. هر وام باید حداقل یک مشتری متناظر با ارتباط وام گیرنده دارد.
- Partial participation : برخی موجودیت ها ممکن است در هیچ ارتباطی از مجموعه ارتباطات مشارکت نداشته باشند.
- مثلا : همه مشتریان ممکن است وام نگیرند.یعنی مشارکت مشتری ناکامل است.



نوع ارتباط به مثابه نوع موجودیت (ارتباط موجودیتی)

در یک دید کلی می توان گفت نوع ارتباط خود نوعی موجودیت است . زیرا پدیده ای است که در دنیای واقعی وجود دارد . با توجه به این تعریف می توان گفت چون نوع ارتباط خود نوعی موجودیت است لذا می تواند صفت یا صفات خاصه ای داشته باشد.اما معمولا فاقد صفت شناسه است.ارتباط یک نوع موجودیت ضعیف با موجودیت قوی معمولا صفت خاصه ندارد.

در مثال دانشجو و درس و رابطه انتخاب می توان صفت خاصه های ترم و نمره و در رابطه حذف صفات ترم و نوع حذف را در نظر گرفت.



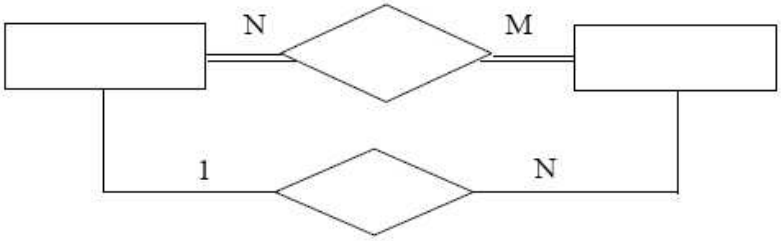
ماهیت نوع ارتباط

چگونگی تناظر بین دو مجموعه نمونه های موجودیت را ماهیت ارتباط گویند.می دانیم سه نوع تناظر وجود دارد :

تناظر یک بیک ، تناظر یک به چند و تناظر چند به چند. این سه گونه تناظر را چنین نشان میدهیم :

1 : 1 ، 1 : N ، N : M.

مثال :ماهیت ارتباط در رابطه حذف تک درس معمولا 1:Nاست (یعنی یک دانشجو یک درس را حذف میکنددولی یک درس ممکن است توسط چند دانشجو حذف شود).



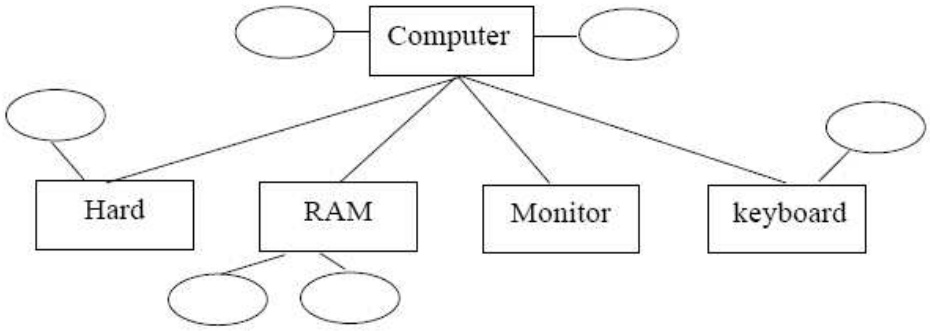
برای نمایش ماهیت ارتباط در نمودار E/R روش دیگری نیز وجود دارد . در این روش به هر مشارکت یک نوع موجودیت در یک ارتباط ، یک زوج عدد صحیح به صورت (min,max) انتساب داده می شود.به این معنی که در

هر لحظه ، هر نمونه موجودیت e از نوع E باید حداقل در min و حداکثر در max نمونه از ارتباط R شرکت داشته باشد. اگر min=0 مشارکت غیر الزامی(اختیاری) و در غیر اینصورت مشارکت الزامی است .

- **موارد اظافه شده به نمودار E/R**

تجزیه و ترکیب :

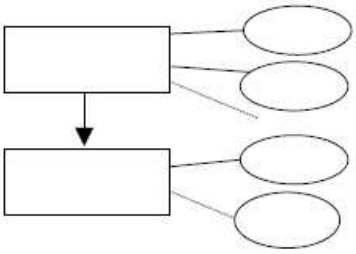
تجزیه یا جداسازی یعنی یک شیء کلی را به اجزاء تشکیل دهنده آن تقسیم کنیم . شیء کل ، صفات ،ساختار و رفتار خود را دارد و هر یک از اجزاء نیز صفات ،ساختار و رفتار خاص خودرا دارند. به این نوع ارتباط در E/ R ارتباط " جزئی است از .." و یا PART -OF – A I S گفته می شود .



زیر نوع ها و ابرنوع های موجودیت Subtype and Supertype

یک موجودیت می تواند بطور همزمان از انواع مختلفی باشد. مثلاً اگر بعضی کارمندان، برنامه نویس باشند و تمام برنامه نویسان کارمند، آنگاه می توان گفت نوع موجودیت برنامه نویس یک زیر نوع از نوع موجودیت کارمند است.

اگر نوع موجودیت y ، یک زیر نوع از نوع موجودیت x باشد آنگاه خطی جهت دار از مستطیل x به مستطیل y رسم می شود. (هر y یک x است) این ارتباط برای پرهیز از تکرار صفات خاصه بین موجودیتها در یک نمودار بکار می رود

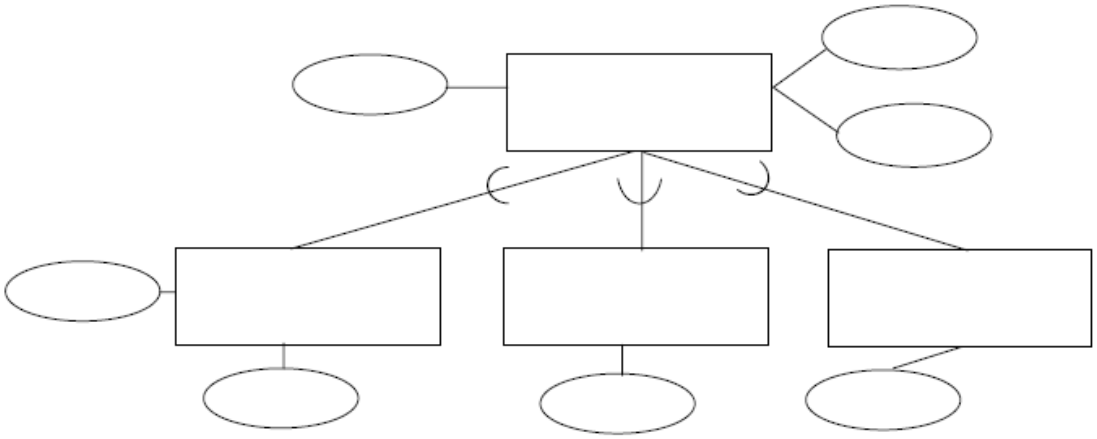


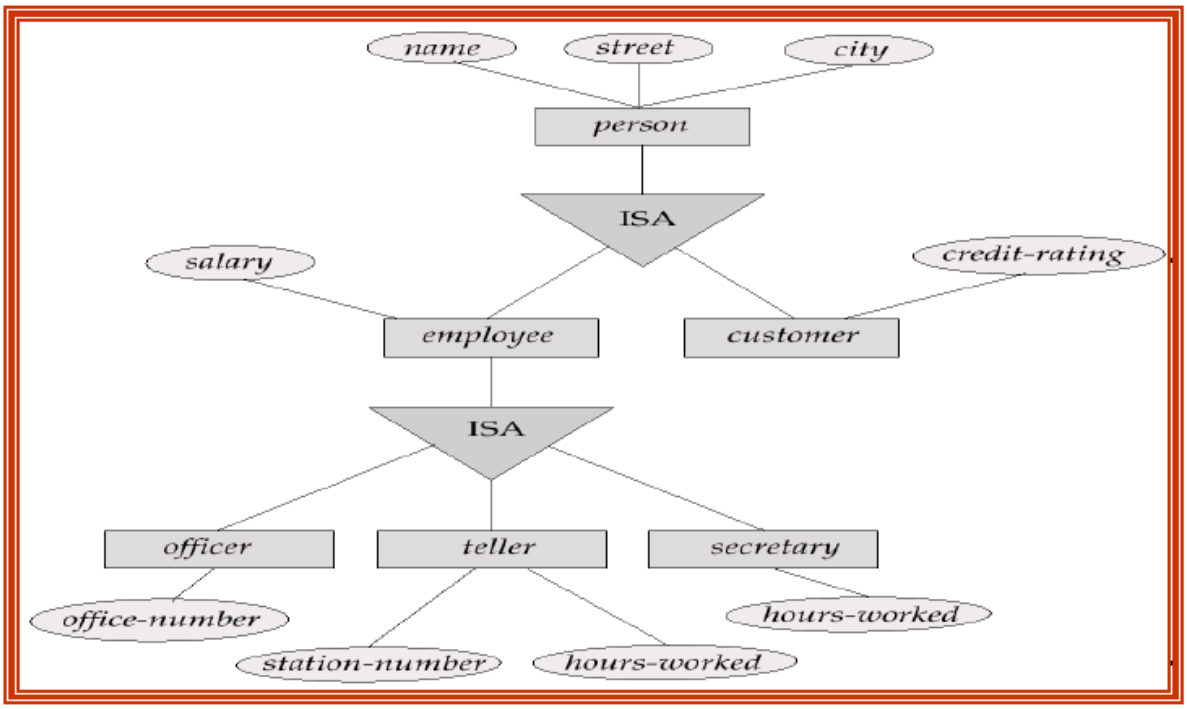
تخصیص (SPECIALIZATION):

مشخص کردن گونه های خاص یک شیء را تخصیص گویند. بطور مثال اگر شیء موجود زنده را درنظر بگیریم سه گونه خاص آن عبارتند از : انسان ، حیوان و نبات . در نمودار E/R یک موجودیت میتواند زیرنوع هایی داشته

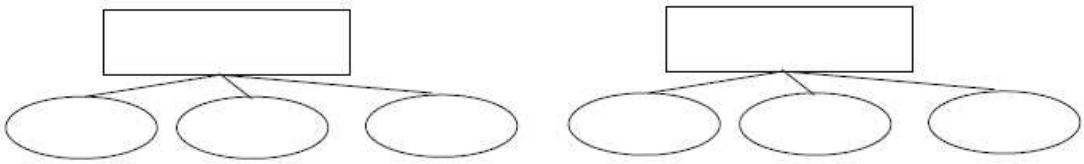
باشد . گوییم بین هر زیر نوع موجودیت و ابر نوع آن ارتباط "**گونه ای است از...**" یا **هست یک ...** I S A -وجود دارد .

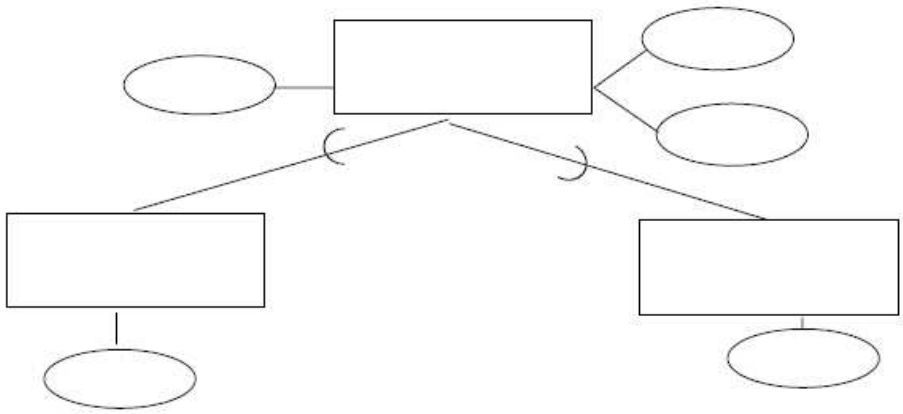
تخصیص : یک فرایند از بالا به پایین است. در تخصیص یک موجودیت به گونه های مختلف گروه بندی می شود. این گروهها به عنوان موجودیت سطح پایین تر آن موجودیت منظور می گردند. در نمودار E/R تخصیص با یک مثلث حاوی ISA نشان داده می شود. یک موجودیت سطح پایین تمام خصوصیات مجموعه موجودیت سطح بالاتر را به ارث می برد.





تعمیم عکس عمل تخصیص است به این معنا که با داشتن زیر نوع های خاص ، صفات مشترک بین آنها را در یک مجموعه صفات برای یک ابرنوع موجودیت در نظر می گیریم. تعمیم یک فرایند از پایین به بالاست که تعدادی مجموعه موجودیتی که خصوصیات مشترک دارند را باهم ترکیب می کند.





توجه :-تخصیص و تعمیم معکوس یکدیگرند که هر دو در نمودار بایک شکل نشان داده می شوند.

- **قیود تعریف شده در ارتباط سلسله مراتبی تعمیم/تخصیص**

برای مدل سازی دقیق تر یک سازمان ، طراح پایگاه داده ممکن است محدودیت ها /قیدها یی را در یک تعمیم ویژه در نظر بگیرد . یکی ازاین قید ها تعیین این است که یک موجودیت می تواندعضوی از یک مجموعه موجودیت سطح پایین در نظر گرفته شود. عضویت می تواندبر اساس شرایط و یا توسط کاربر تعیین گردد.

★ بر حسب شرط خاص تعریف شده:

دراین حالت عضویت بر اساس یک شرط صریح یا گزاره ای بیان می شود.

★condition-defined

✓E.g. all customers over 65 years are members of *senior-citizen* entity set; *senior-citizen* ISA *person*.

★ **تعریف شده توسط کاربر:**

کاربر پایگاه داده موجودیت را به یک مجموعه موجودیت مورد نظر تخصیص می دهد.

نوع دیگر محدودیت از نظر اینکه یک موجودیت سطح به یک یا بیشتر از موجودیت سطح پایین تعلق دارد تعریف می گردد.این نوع محدودیت به دو شکل مجزا (Disjoint) و همپوشانی (Overlapping)وجود دارد.

★ **Disjoint**

به معنای این است که یک موجودیت می تواند فقط به یک مجموعه موجودیت سطح پایین تعلق داشته باشد. در نمودار E/R با نوشتن عبارت disjoint در کنار مثلث ISA این قید تعریف میشود.

★**Overlapping**

به معنای این است که یک موجودیت می تواند فقط به یک یا بیشتر ازمجموعه موجودیت سطح پایین تعلق داشته باشد.

قید دیگری که در تعمیم مطرح شده ، قید کامل بودن است که بیانگر این است که یک موجودیت سطح بالاتر باید متعلق به حداقل یکی از مجموعه موجودیت های سطح پایین تر باشد یا خیر . بر این اساس دو نوع قید کامل و جزئی را داریم :

★ **total :** یک موجودیت باید متعلق به یکی از مجموعه موجودیت های سطح پایینتر باشد.

★ **partial:** یک موجودیت می تواند متعلق به یکی از مجموعه موجودیت های سطح پایینتر نباشد.

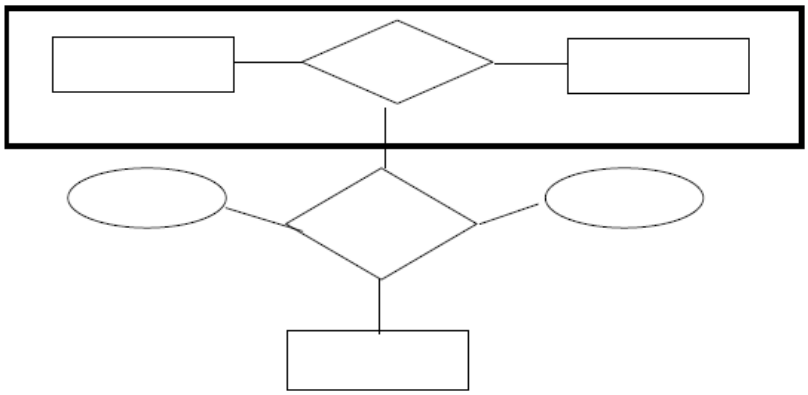
حالت پیش فرض مساله ، حالت جزئی است . برای بیان قید کامل بودن باید از دو خط استفاده گردد(همانند مشارکت الزامی/کامل در نمودار E/R)

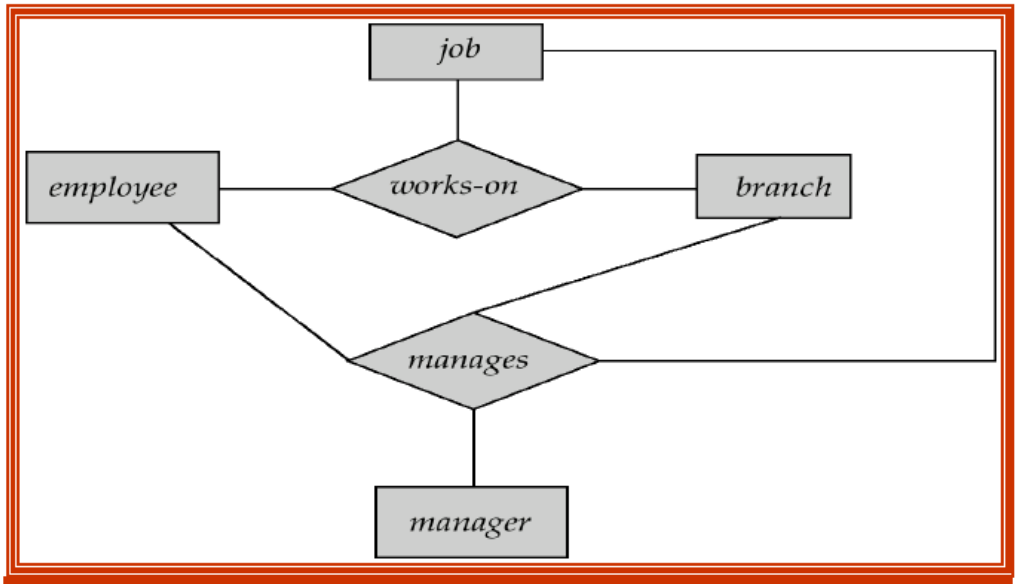
تمرین : ۵ مثال مختلف از تعمیم ذکر کنید و در هر کدام قیود مختلف را بررسی نمایید.

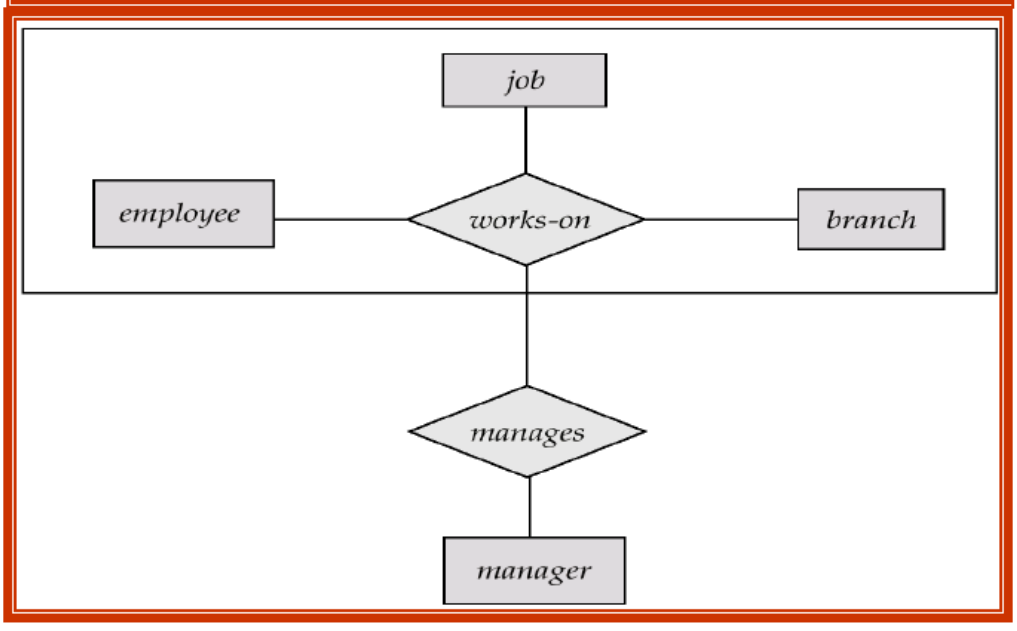
۲-۲-٦- تجمع Aggregation

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 میگیرند. و این نوع موجودیت خود می تواند با نوع موجودیت دیگری ارتباط داشته باشد .در واقع زمانی از تجمع استفاده میشود که بخواهیم ارتباطی را بین ارتباط ها بیان کنیم و یا بخواهیم ارتباطات افزونه را کم کنیم.

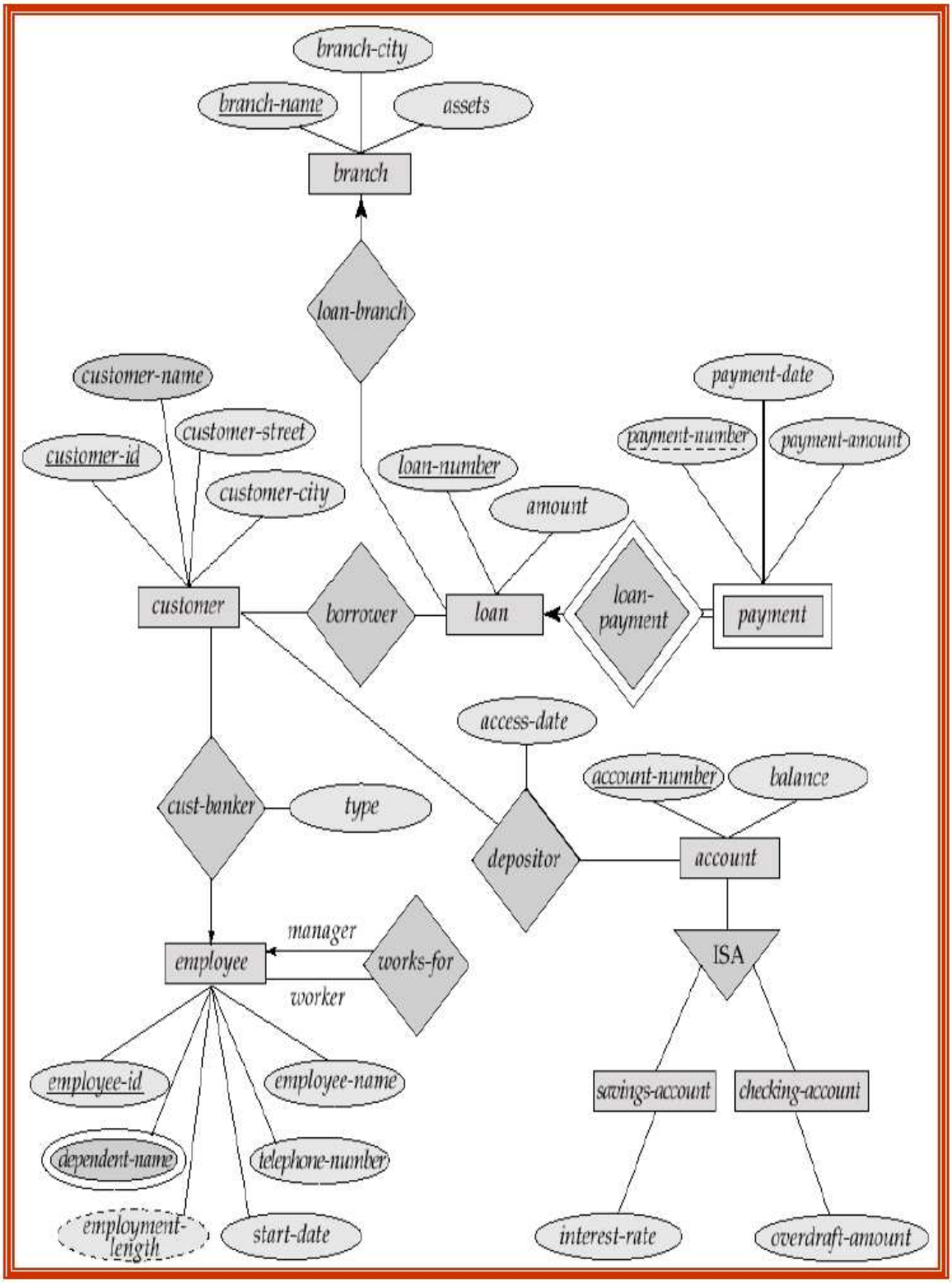
مثال : ارتباط بین نوع موجودیتهای دانشجو ، درس و استاد را می توان همانند شکل زیر مدلسازی نمود.




job
employee
works-on
branch
manages
manager


job
employee
works-on
branch
manages
manager

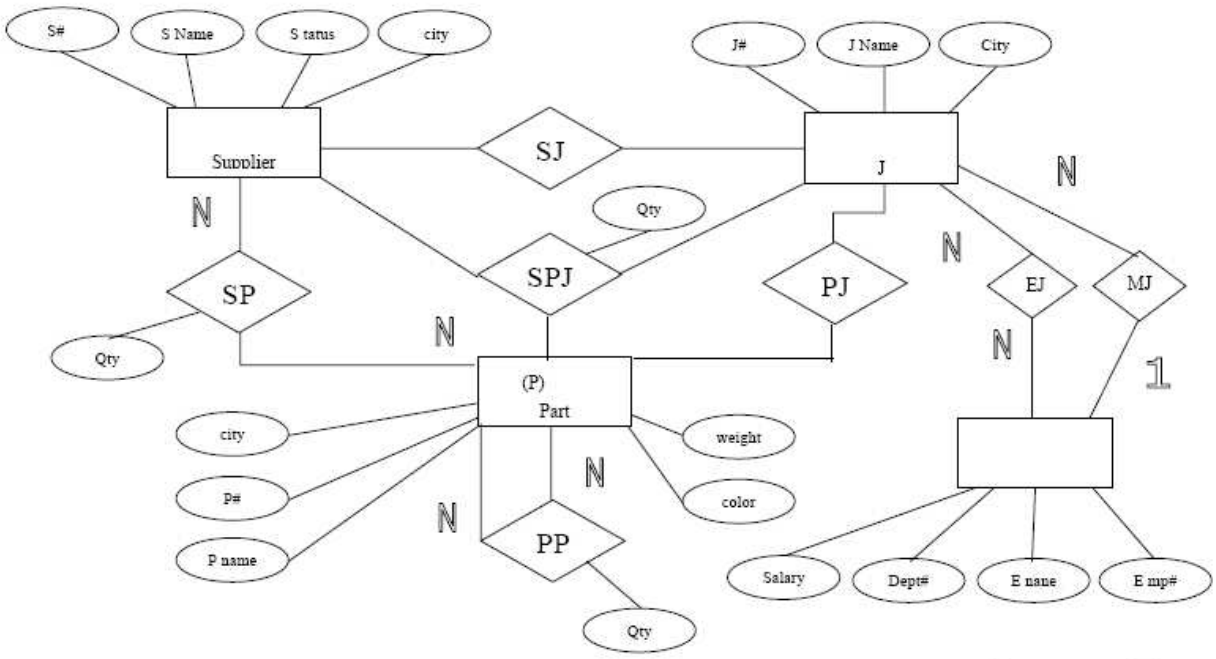
مثال : نمودار E/R یک سیستم بانکی :



مثالی دیگر :

محـیط عملیاتـی: سـازمان یـا شـرکتی را در نظـر مـی گیـریم که پروژه هایی را در دست اجرا دارد.در پروژه ها از قطعاتـی در کار ساخت استفاده می شود و تهیه کنندگانی این قطعات را تأمین می کنند. قطعات در پروژه ها استفاده می شـوند. 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.

یک نمودار ساده E/R می تواند به فرم زیر باشد:



- ارتـباط ممکـن اسـت مابین بیش از دو موجودیت باشد(SPJ) . اطلاعاتی که از این ارتباط بین سه موجودیت به دست می آید همیشه لزوماً همان اطلاعاتی نیست که از ارتباط دو به دوی موجودیت ها بدست می آید.

مثال:

۱ – تهیه کننده S_1 قطعه P_1 را تهیه می کند.

۲ – قطعه P_1 در پروژه J_1 بکار رفته است.

۳ – تهیه کننده S_1 برای پروژه J_1 قطعه تهیه کرده است.

٤ – تهیه کننده S_1 قطعه P_1 را برای استفاده در پروژه J_1 تهیه کرده است.

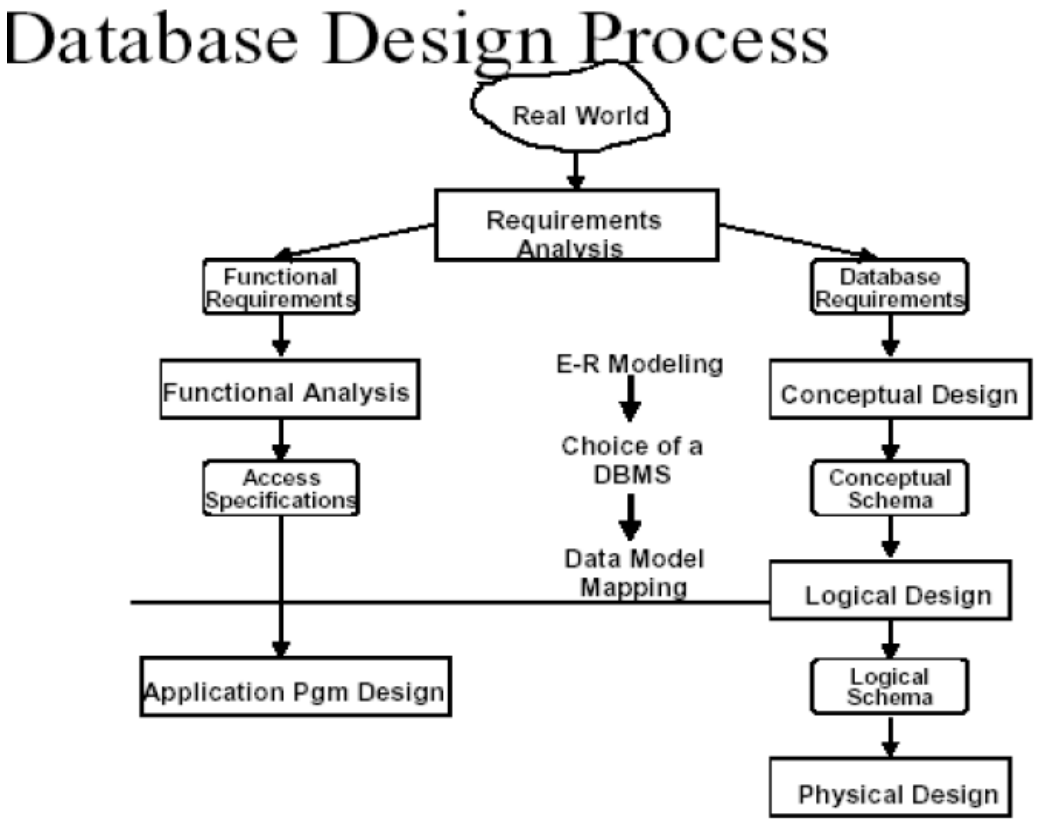
همیشه از اطلاع ۱ و ۲ و ۳ نمی توان اطلاع ٤ را نتیجه گرفت.

۲-۳- اصطلاح دام پیوندی (Connection trap)

اگر از ارتباط بین دو به دوی موجودیتها نتیجه گرفته شود که حتماً ارتباط بین سه موجودیت یا بیشتر از آن بدست آید در اینصورت طراح گرفتار دام پیوندی شده است.

۲-٤-طراحی پایگاه داده ها

طراحی یک پایگاه داده مستلزمی مراحلی است که در هر مرحله فعالیتهایی انجام میشود . شکل زیر نمودار ساده شده مراحل طراحی پایگاه داده ها را نشان می دهد.



۲-٤-طراحی پایگاه داده ها و ابزارهای Case

برای طراحی پایگاه داده ها ابزارهای مختلفی ارائه شده اند . این ابزارها به طراح پایگاه داده کمک میکنند تا درمراحل مختلف مدلسازی و طراحی پایگاه داده ،تصمیم گیری مناسب را انجام دهد.این ابزارها امکان ترسیم نمودار E/R را با استفاده از انتخاب اشیاء از یک جعبه ابزاررا بوجود می آورند.بطور کلی مزایای استفاده از این ابزارها را می توان بصورت زیر نام برد :

۱- سادگی فرایند ایجاد نمودارها

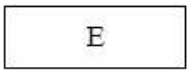
۲- تولید خودکار جملات SQL برای تعریف جدولها ، محدودیت ها ، اندیس ها و دیگر اشیاء مدل رابطه ای.
۳- امکان مستند سازی هر موجودیت ، صفت خاصه ، رابطه و محدودیت.
٤-
برخی ابزار ها یی که در حال حاضر استفاده می شوند عبارتند از::
۱- ER STUDIO برای مدلسازی E/R بانک اطلاعاتی
۲- DB Atrisan برای مدیریت پایگاه داده ها و امنیت آن
۳- Oracle Developer 2000 & Designer 2000 برای مدلسازی پایگاه داده و توسعه برنامه های کاربردی
٤- Platinum Enterprise Modeling suite : ER Win , BpWin برای مدلسازی داده ها و پردازش ها
٥- RW Metro برای تبدیل از O-O به مدل رابطه ای.
٦- Rational Rose برای مدلسازی UML و تولید برنامه های کاربردی به زبان جاوا و C++.
۷- Visio Enterprise Visual Basic برای مدلسازی داده ها و طراحی م مهندسی مجدد.
۸- X Case برای مدلسازی مفهومی .
۹- Case Studio برای مدلسازی E/R بانک اطلاعاتی.

تمرین: نمودار E/R هر یک از محیطهای عملیاتی زیر را رسم کنید:

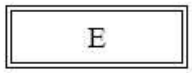
- سیستم اطلاعات یک نمایشگاه بین المللی
- سیستم اطلاعات گیاهان
- سیستم اطلاعات شخصی (P I S)
- سیستم اطلاعات آزمایشگاه طبی
- سیستم اطلاعات تعمیر و نگهداری کامپیوتر ها
- سیستم اطلاعات داروخانه یک بیمارستان
- سیستم اطلاعات فعالیتهای کلوپ فیلم
- سیستم اطلاعات مسابقات علمی
- سیستم اطلاعات یک بانک
- سیستم اطلاعات موسیقی و موسیقی دانان
- سیستم اطلاعات نقاشی و نقاشان
- سیستم اطلاعات یک مرکز تحقیقاتی

خلاصه شکلهای بکاررفته در نمودار E/R

نوع موجودیت



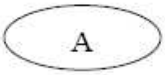
نوع موجودیت ضعیف



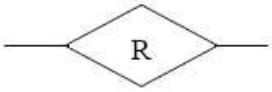
نوع رابطه

رابطه بین موجودیتها و موجودیت ضعیف

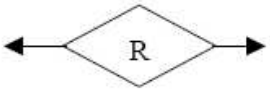
کلید اصلی



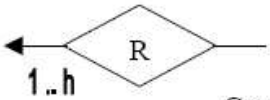
رابطه چند به چند



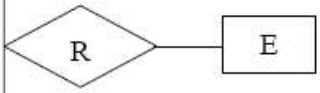
رابطه یک به یک



رابطه یک به چند



محدودیتهای Cardinality

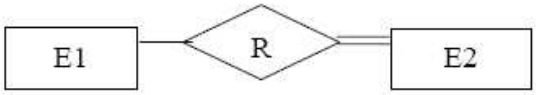


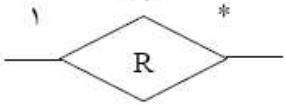
صفت خاصه

صفت خاصه چند مقداری

صفت خاصه مشتق

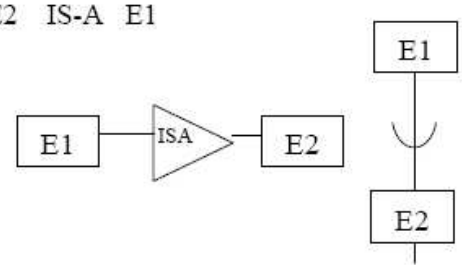
شرکت کامل (الزامی)موجودیت در رابطه

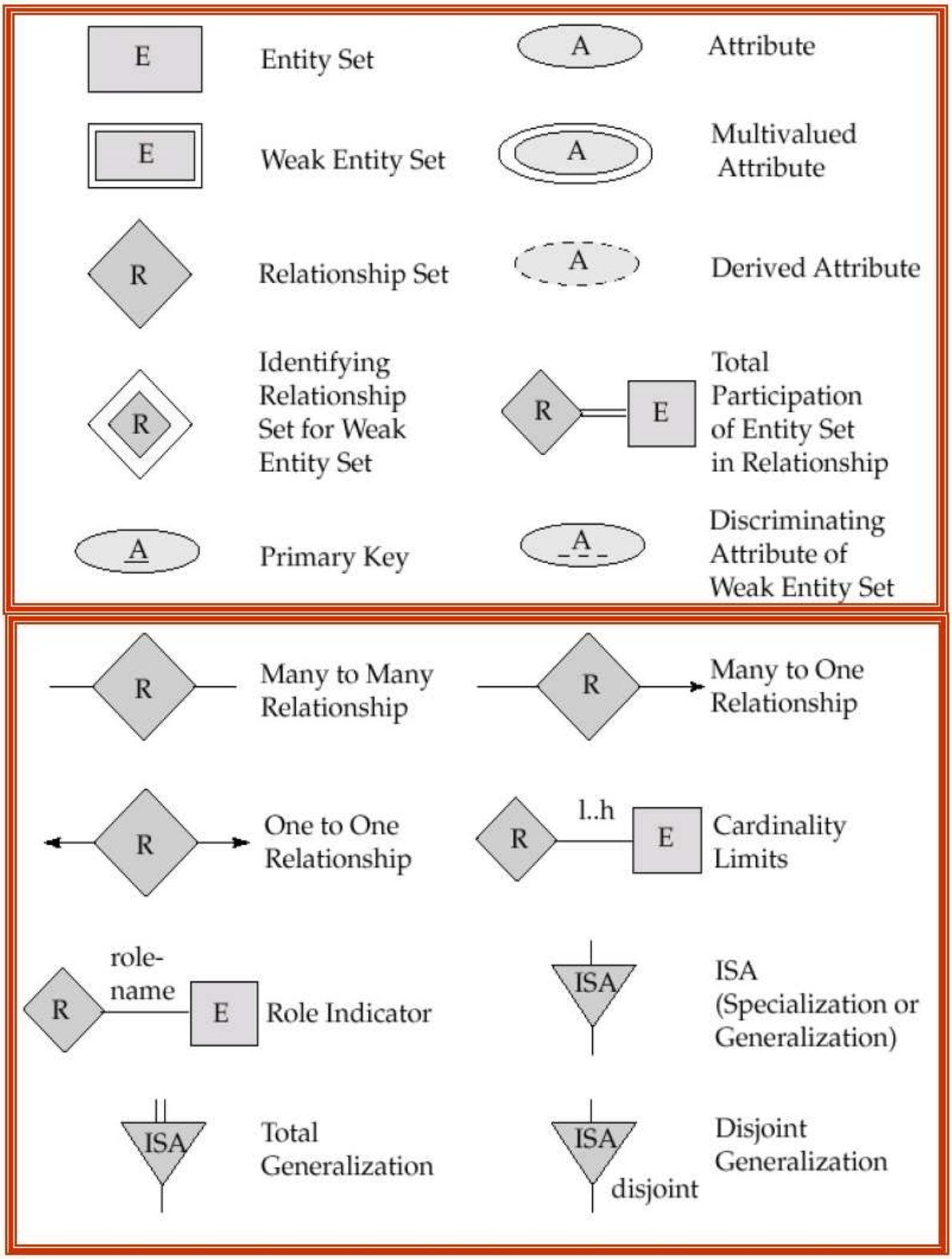


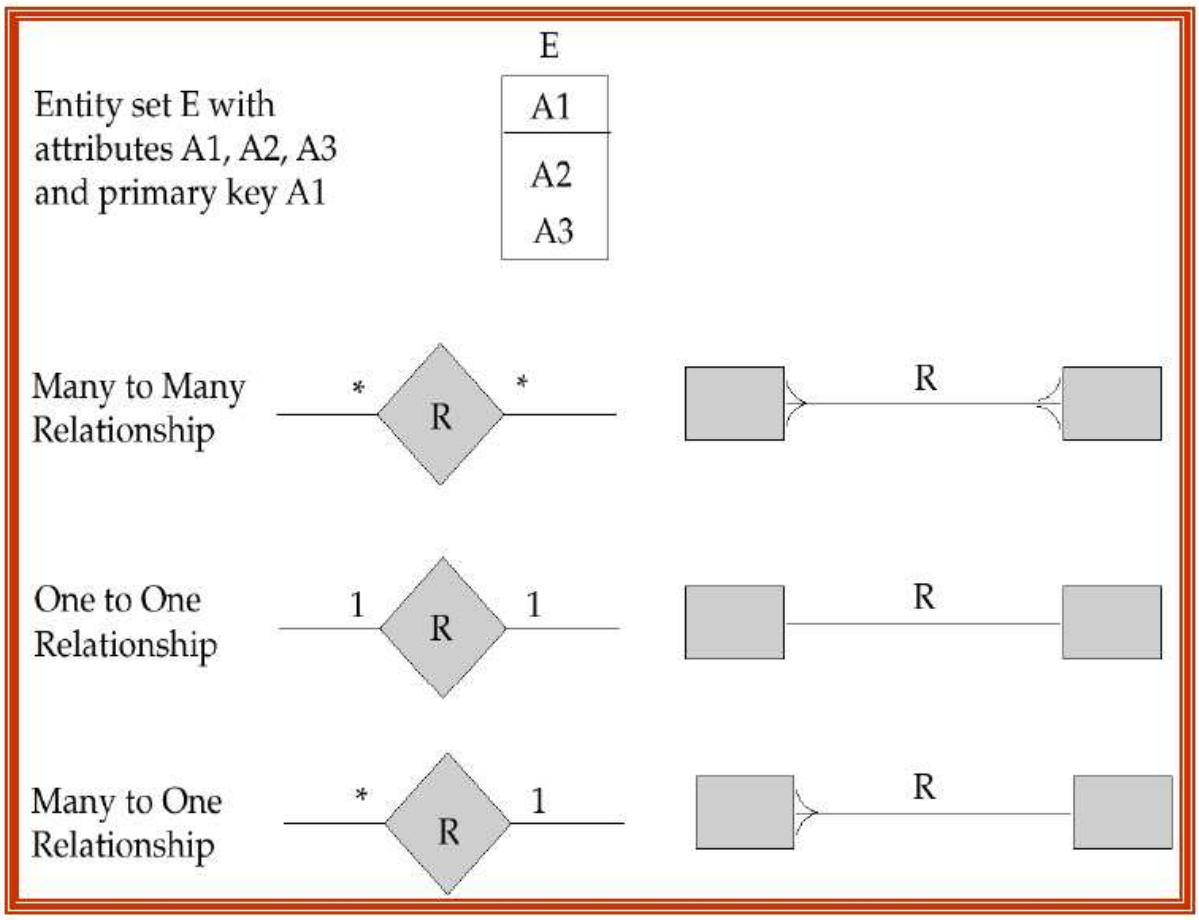


ISA (Specialization or Generalization)

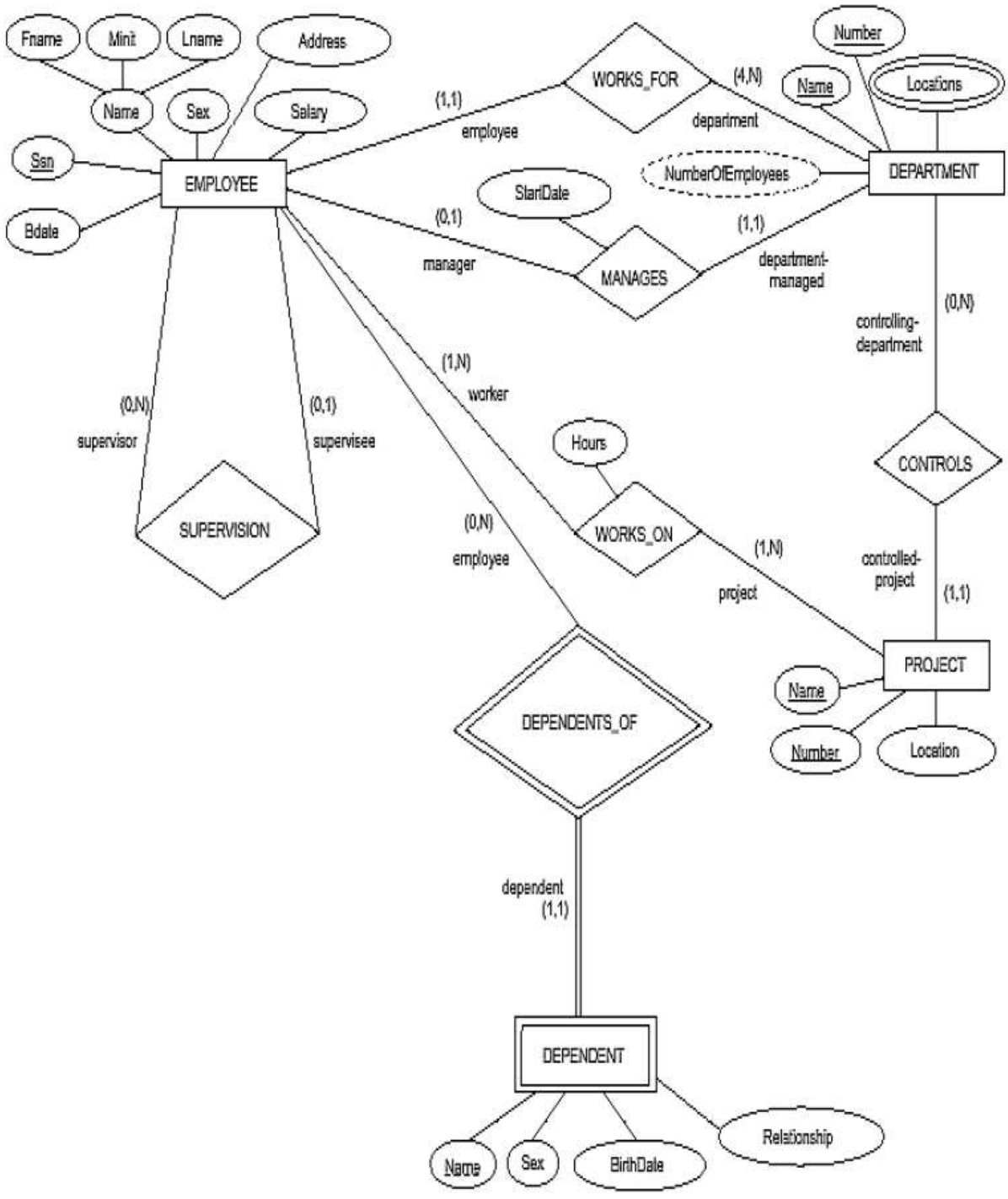
E2 IS-A E1




E
Entity Set
A
Attribute
E
Weak Entity Set
A
Multivalued Attribute
R
Relationship Set
A
Derived Attribute
R
Identifying Relationship Set for Weak Entity Set
R
E
Total Participation of Entity Set in Relationship
A
Primary Key
A
Discriminating Attribute of Weak Entity Set
R
Many to Many Relationship
R
Many to One Relationship
R
One to One Relationship
R
l..h
E
Cardinality Limits
role-name
R
E
Role Indicator
ISA
ISA (Specialization or Generalization)
ISA
Total Generalization
ISA
disjoint
Disjoint Generalization

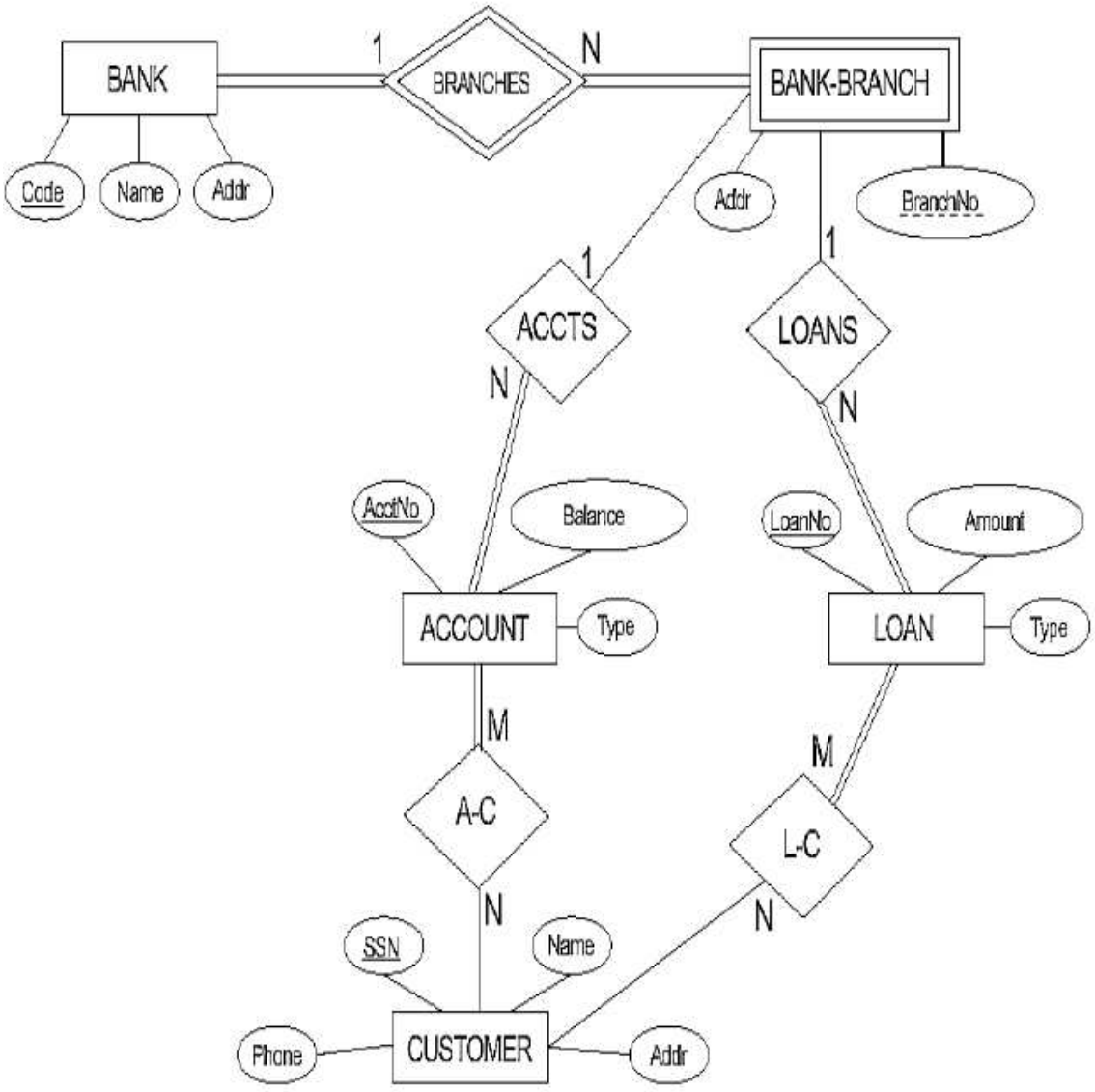

E
A1
A2
A3
Entity set E with attributes A1, A2, A3 and primary key A1
Many to Many Relationship
*
R
*
R
One to One Relationship
1
R
1
R
Many to One Relationship
*
R
1
R

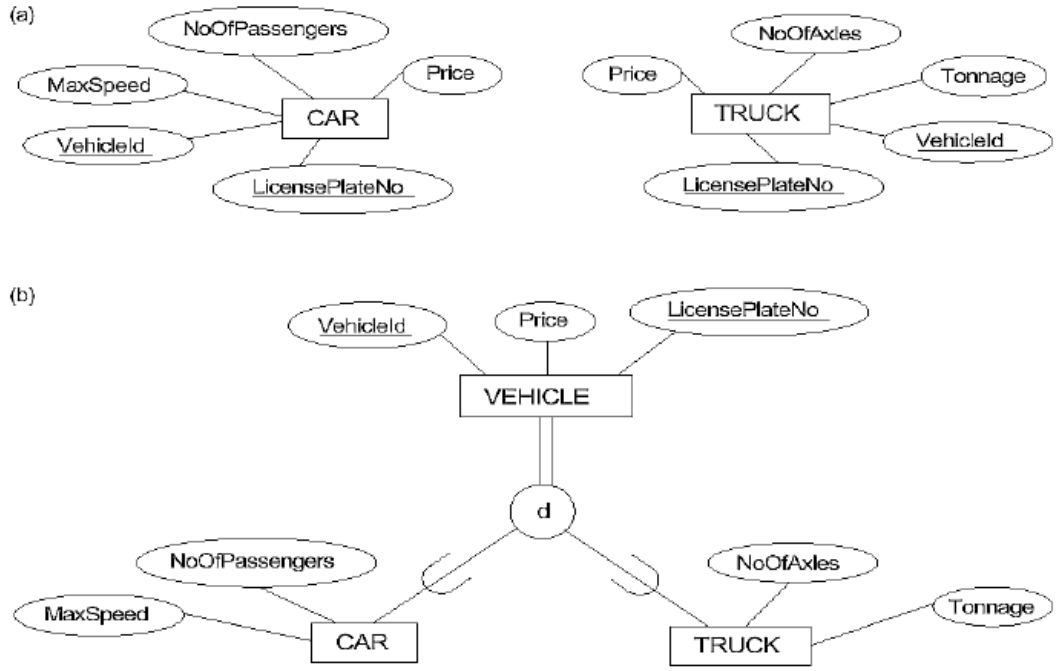
۲-۵- چند مثال از نمودار E/R



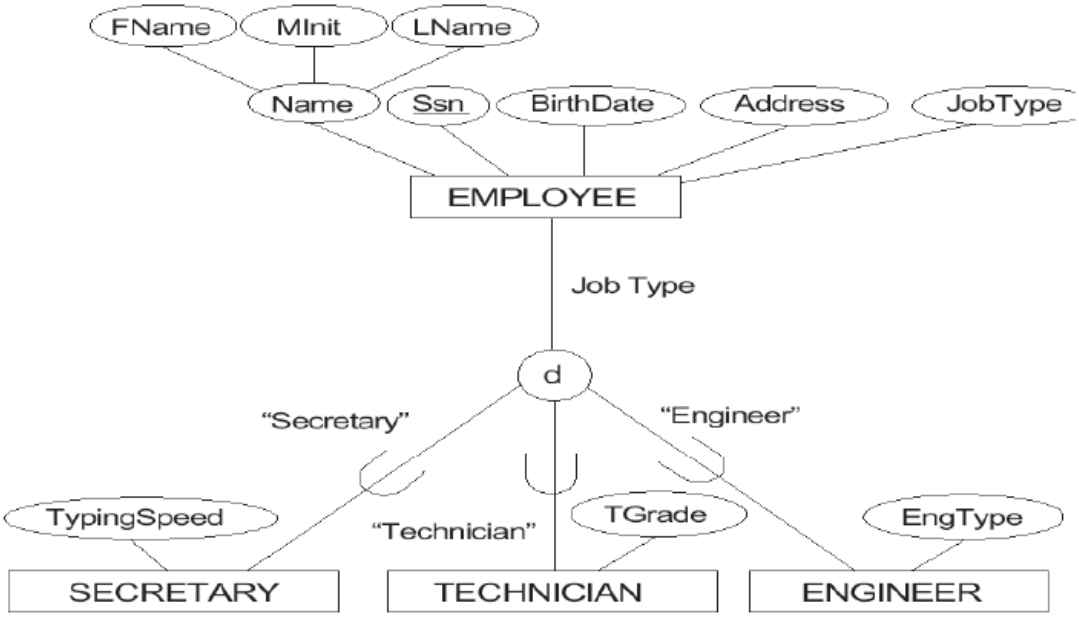
❑ مثالی از یک سیستم کارمند - پروژه

❑ نمودار E/R یک سیستم بانکی:





❑ ،مثالی از تخصیص و تعمیم:



فصل پنجم

مدل رابطه ای

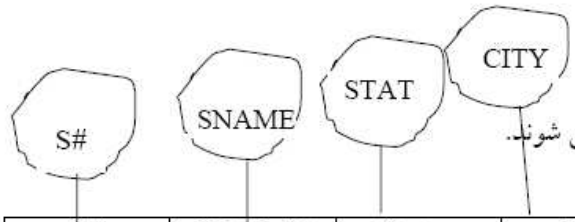
☐ تعریف رابطه :

رابطه از دو مجموعه تشکیل شده است : یکی موسوم به مجموعه عنوان یا heading رابطه و دیگری مجموعه بدنه یا body . عنوان، مجموعه ای ثابت از صفات خاصه A_1..... A_n است که این صفات خاصه هر یک مقادیرشان را از یک میدان (حوزه) می گیرند. مجموعه بدنه مجموعه ایست متغیر در زمان از چند تایی مقادیر صفات خاصه بنام تاپل(tuple). معمولاً در مدل رابطه ای از اصلاح جدول بجای رابطه نیزاستفاده می شود. یعنی جدول امکانی است برای نمایش مفهوم رابطه بخاطر تامین و وضوح کاربردی. معمولا رابطه را بصورت r (R) نمایش میدهند که به R شمای رابطه نیز گفته میشود. R = (A1,A2,A3,…)

☐ تعریف دامنه / میدان :

مجموعه ایست که مقادیریک صفت خاصه از آن برگرفته می شوند.

مثال: موجودیت تهیه کننده را در نظر می گیریم:



S#	SNAME	Status	City
۱۰۰	نام۱	۲۰	تهران
۲۰۰	نام۲	۳۰	قزوین
۳۰۰	نام ۳	۵۰	کرج
۲۵۰	نام٤	۱۵	تبریز

ارتباط بین مدل رابطه ای و نمایش جدولی:

مدل رابطه ای(تئوریسین)	ساختار جدولی(طراح)
رابطه	جدول
تاپل	سطر
صفت خاصه	ستون
میدان	مقادیر مجاز یک ستون

وقتیکه اسم و مجموعه عنوان رابطه مشخص باشد گویند ذات یا جوهر رابطه (Intension) معلوم است. به بدنه رابطه بسط رابطه (Extension) نیز گویند.

نکته: ذات رابطه ثابت در زمان است اما بسط رابطه در زمان متغیر است.

□ درجه رابطه:

تعداد صفات خاصه رابطه را درجه آن گویند. اگر درجه رابطه یک باشد رابطه یگانی ، رابطه درجه دو را دوگانی (binary) ، رابطه درجه سه را سه گانی (Ternary) و رابطه با درجه N را Nگانی (N-ray)گویند.

□ کاردینالیتی رابطه:

به تعداد تاپلهای رابطه در یک لحظه از حیات آن کاردینالیتی رابطه گویند. کاردینالیتی رابطه در طول حیات رابطه متغیر است. بعنوان مثال، رابطه از درجه رابطه ٤ میباشد.

□ خصوصیات رابطه:

۱_ به کمک یک ساختار ساده بنام جدول قابل نمایش است.

۲_ تاپل تکراری در رابطه وجود ندارد. زیرا بدنه رابطه مجموعه است و در مجموعه عناصر تکراری وجود ندارد.

۳_ تاپلهادر رابطه نظم خاصی ندارند.این خاصیت نیز از مجموعه بودن بدنه رابطه نتیجه می شود.

٤_ صفات خاصه نظم ندارند. این خاصیت نیز از مجموعه بودن عنوان رابطه نتیجه می شود

۵_ عناصر تشکیل دهنده تاپل اتمیک هستند یعنی تجزیه نشدنی می باشند. بعبارتی گوئیم یک فقره داده تجزیه نشدنی است اگر نتوان آن را به مقادیر دیگر تجزیه کرد.بعنوان مثال: مقادیر تاریخ ماهیتی غیر اتمیک دارد زیرا از سه جزء ماه ، سال و روز تشکیل شده است .در این مثال خاص اتمیک و یا غیر اتمیک بودن تاریخ بستگی به دید طراح در طراحی دارد بنابراین این دو مفهوم مطلق نیستند و به معنایی که طراح برای داده ها قائل می شود بستگی دارد.

□ مفهوم میدان و نقش آن در عملیات روی بانک:

میدان : مجموعه ای از مقادیر است که یک یا بیش از یک صفت خاصه از آن مقدار می گیرند. میدان در عملیات روی بانک مزایایی دارد که عبارتند از:

- سبب ساده تر شدن و کوتاهتر شدن شمای پایگاه داده میگردد.(از نظر تعداد احکام) . زیرا لازم نیست که در تمام رابطه ها ، هربار مشخصات صفات را بدهیم.۰

- کنترل مقداری عملیات در پایگاه.

مقادیر یک صفت خاصه در طول حیات رابطه، از مقادیر میدان برگرفته می شوند. بعبارت دیگر باید در میدان وجود داشته باشند. بنابراین به کمک مفهوم میدان می توان عملیات روی بانک را از نظر مقادیر صفات خاصه کنترل کرد.
بطور مثال اگر میدان مقادیر STAUS بصورت { 10,20,30,40,50} = Domain STATUS باشد امکان درج اطلاع (تهران و S7, Sn7, 60) در بانک میسر نمی باشد.

- امکانی است برای کنترل معنایی پرس و جوها

مثال : شماره قطعاتی را بدهید که وزن آنها برابر تعداد تهیه شده آنها باشد.

صفات وزن و تعداد به اعتبار همنوع بودن قابل مقایسه اند و سیستم می تواند با انجام مقایسه های لازم ، به پرسش کاربر پاسخ دهد. اما این دو صفت به لحاظ مفهومی غیر قابل مقایسه اند، زیرا روی دو میدان ماهیتاً متفاوت تعریف شده اند.

- پاسخگویی به بعضی از پرس و جوها را آسان میکند.

اگر امکان تعریف میدان وجود داشته باشد این تعریف وارد کاتالوگ سیستم بعنوان بخشی از شمای ادراکی پایگاه می شود و در شرایطی برخی از کاربران می توانند از آن استفاده کنند.

مثال: درچه رابطه هایی از پایگاه اطلاعاتی از تهیه کنندگان وجود دارد؟

اگر میدان ها تعریف شوند برای پاسخگویی به این پرس و جو فقط مراجعه به کاتالوگ کفایت می کند .

□ مفهوم کلید در مدل رابطه ای:

در مدل رابطه ای چند مفهوم در خصوص کلید مطرح است که عبارتند از :

- ابر کلید
- کلید کاندید
- کلید اصلی
- کلید بدیل
- کلید خارجی

□ مفهوم ابر کلید super key

مجموعه ای ازیک یا چند صفات خاصه را که دارای یکتایی مقدار باشند ابرکلید گویند. به بیان دیگر ، هر ترکیبی از صفات خاصه رابطه که در هیچ دو تاپل مقدار یکسان نداشته باشد.

□ کلید(نامزد) کاندید: Candidate Key

ابر کلیدی که خاصیتی کاهش ناپذیری داشته باشد کلید کاندید گویند. بعبارت دیگر هر زیر مجموعه از مجموعه عنوان (Ai, Aj .. Ak) که دو خاصیت زیر داشته باشد کلید کاندید گویند.

۱_ یکتایی مقدار (Uniqueness)

به این معنا که در هر لحظه از حیات رابطه مقدار (A i , Aj .. Ak) یکتا باشد.

۲_ کاهش ناپذیری minimatlity

به این معنی است که از نظر تعداد اجزاء در حداقل باشد در عین حال که یکتایی محفوظ بماند. گوئیم زیر مجموعه ای کاهش ناپذیر است یا حداقل اجزاء دارد اگر یکی از عناصر این زیرمجموعه حذف شود در زیرمجموعه باقیمانده خاصیت یکتایی مقداراز بین برود.

با توجه به تعاریف می بینیم هر ابرکلید لزوما کلید کاندید نیست اما هر کلید کاندید جزء مجموعه های ابر کلید رابطه هست.

مثال: در رابطه S صفت خاصه S# کلید کاندید است و در رابطه SP (S#, P#) کلید کاندید است.

نکته: ۱: رابطه ممکن است بیش از یک کلید کاندید داشته باشد.

نکته ۲_ وجود حداقل یک کلید کاندید در رابطه تضمین است زیرا در بدترین حالت با ترکیب تمام صفات خاصه به یکتایی مقدار می رسیم. به رابطه ای که مجموعه عنوانش کلید کاندید آن باشد اصطلاحاً رابطه تمام کلید (All key) نامیده می شود.

نکته ۳_ نقش کلید کاندید : امکانی است برای ارجاع به تاپل یعنی نوعی مکانیسم آدرس دهی در سطح تاپل است.

□ کلید اصلی: Primary Key

یکی از کلیدهای کاندید است که طراح با توجه به ملاحضات محیط عملیاتی، خود انتخاب می کند. دو ضابطه در تعیین کلید اصلی از بین کلیدهای کاندید باید در نظر گرفته شوند.

۱_ نقش و اهمیت کلید اصلی نسبت به سایر کلید های کاندید در پاسخگویی به نیاز های کاربران

۲_ کوتاهتر بودن طول کلید کاندید از نظر طول رشته بایتی حاصله از ترکیب صفات خاصه.

کلید اصلی شناسه تاپل است و بایستی به نوعی به سیستم معرفی شود که معمولا در سیستمهای رابطه ای با عبارت Primary Key (Attribute) تعریف میشود.

□ کلید نامزد(بدیل) Alternate Key

هر کلید کاندید غیر از کلید اصلی کلید بدیل نامیده می شود.اگر همه کلید های کاندید رابطه و نیز خود کلید اصلی به سیستم معرفی شوند ، دیگر نیازی به تصریح کلید دیگر با عبارت Alternate Key نیست.

□ کلید خارجی Foreign Key

هر صفت خاصه ای از رابطه Rj (ساده یا مرکب) مانند Ai که در رابطه ای دیگر مثلاً Ri کلید اصلی باشد کلید خارجی Rj نامیده می شود.

مثال : صفت خاصه S# در جدول SP کلید خارجی است و صفت خاصه P# در رابطه SP نیز کلید خارجی است.

نکته: لزومی ندارد که کلید خارجی یک رابطه جزء تشکیل دهنده کلید اصلی همان رابطه باشد هر چند در مثال بالا چنین است.

مثال: Department (Dept # , Dname , manager - Emp #, budget)

Employe (Emp # , Ename, Dept # , Salary)

در رابطه deparment صفت خاصه # dept کلید اصلی است لذا در رابطه Employe کلید خارجی است و نیز صفت خاصه # Emp در جدول Employe کلید اصلی است پس صفت خاصه Manager - Emp # در رابطه department کلید خارجی است و جزئی از کلید اصلی این رابطه هم نیست.

نکته : لزومی ندارد Rj از Ri متمایز باشد:

E mployе (Emp # , Ename, Manager -Emp # , Salay)

رابطه: کارمند مدیر است .

سوال : نقش کلید خارجی چیست؟

کلید خارجی امکانی است برای ایجاد ارتباط بین تاپلها. بعنوان مثال وجود کلید های خارجی P #, S# در رابطه Sp نمایشگر ارتباطی است که بین تاپلهای رابطه S و تاپلهای رابطه P وجود دارد.

نکته: آیا تنها عامل برقراری ارتباط کلید خارجی است؟

پاسخ منفی است هر صفت خاصه مشترک در عنوان دو رابطه امکانی است برای ایجاد و پیاده سازی نوعی ارتباط.

مثال: P (P#, ... city) , S (S#, ...,city)

وجود City در رابطه امکان ارتباط بین دو موجودیت را بوجود می آورد. در صورتیکه C ity نه کلید خارجی S و نه کلید خارجی P است.

نکته: کلید خارجی رابطه را نیز باید به سیستم معرفی نمود . برای این منظور از دستور زیر استفاده میشود :

FOREIGN KEY (Attribute) REFERENCE Relation name

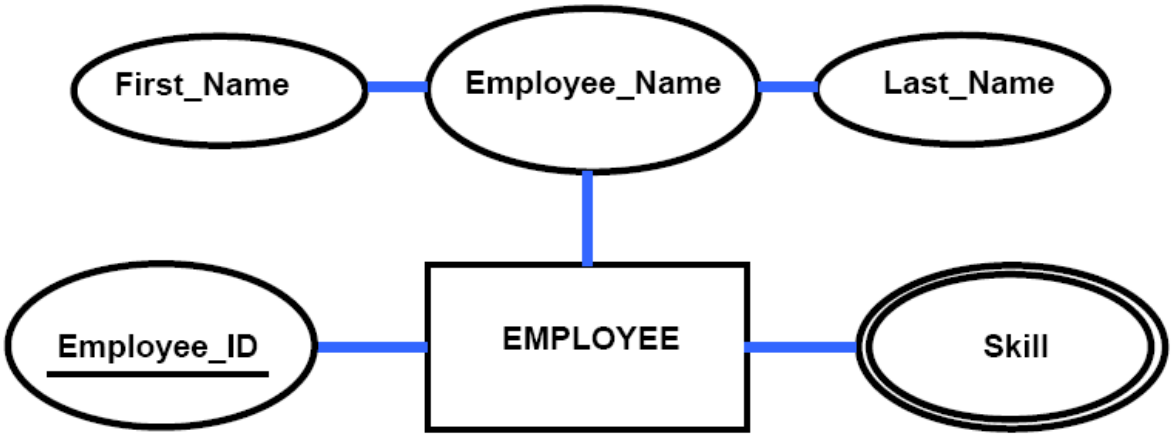
☐ تبدیل مدل E/R به مدل رابطه ای

یک پایگاه داده طراحی شده بر اساس مدل موجودیت/ارتباط می تواند توسط مجموعه ای از جدول ها نمایش داده شود.برای تبدیل مدل موجودیت/ارتباط به مدل رابطه ای از یکسری قوانین استفاده می شود که در ادامه آورده شده اند:

قاعده ۱: هر موجودیت **قوی** توسط یک جدول با همان صفات مورد نظر نمایش داده می شود.

- صفات مرکب در مدل رابطه ای وجود نداشته بلکه فقط صفات جزء آن صفات مرکب بطور مجزا در جدول قرار می گیرند.
- صفت چند مقداری M از موجودیت E با یک جدول مجزای EM نشان داده می شود. این جدول دارای ستونهای متناظر با کلید اصلی E و صفت M خواهد بود.

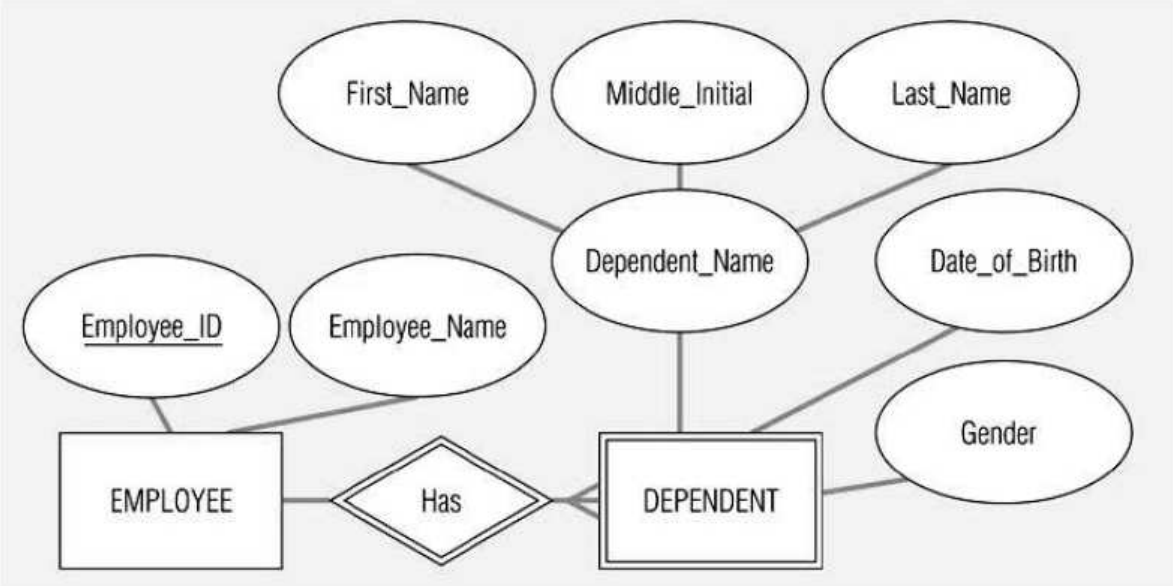
مثال : موجودیت Employee با صفت چند مقداری Skill

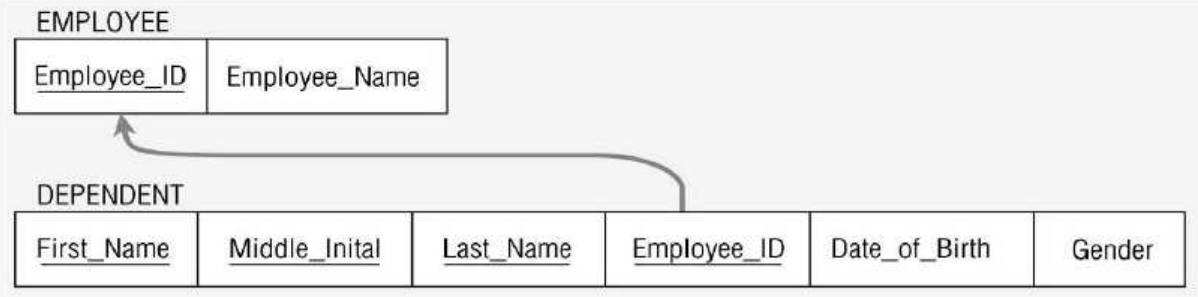


قاعـده ۲: هـر موجـودیت **ضـعیف** توسـط یک جدول با همان صفات مورد نظر به همراه کلید اصلی موجودیت قوی (کلید خارجی این جدول) نمایش داده می شود.

- کلید اصلی جدول جدید ،ترکیبی است از شناسه موجودیت ضعیف و کلید اصلی موجودیت قوی مورد نظر .

مثال : موجودیت ضعیف DEPENDENT

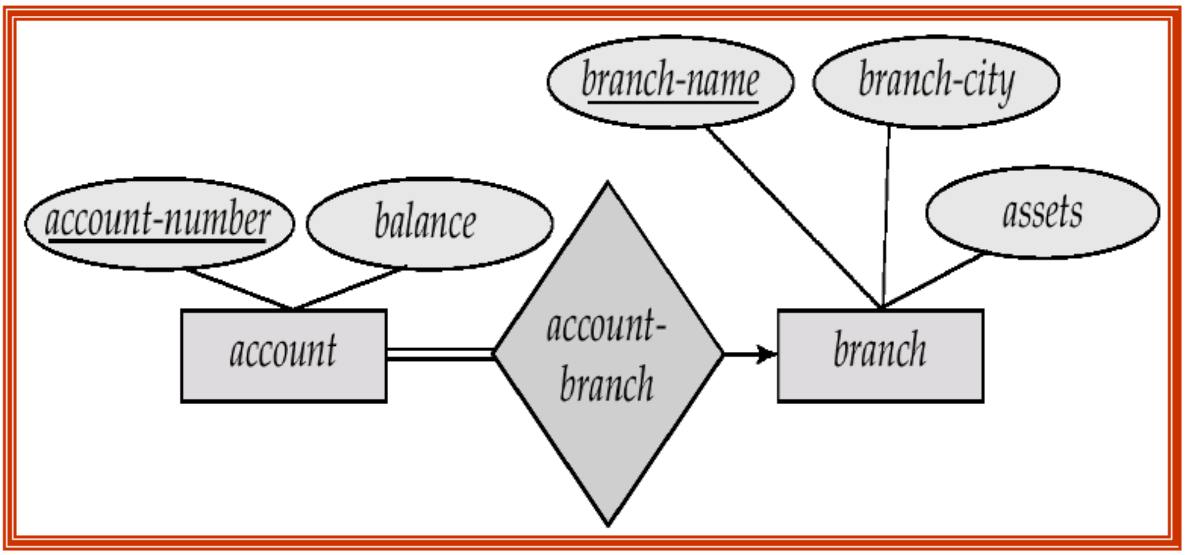




قاعده ۳ : ارتباطات درجه ۲ .

- هر ارتباط درجه دو چند به چند با یک جدول نمایش داده می شود که ستونهایش از کلیدهای اصلی دو موجودیت مشارکت یافته در ارتباط به همراه صفات دیگر مجموعه ارتباط تشکیل می گردند.
- ارتباطات چند به یک یا یک به چند در طرف چند با استفاده از صفت اضافه شده که همان کلید اصلی طرف یک می باشد نمایش داده می شوند.(جدول مجزا نخواهد بود)به عبارت دیگر کلید اصلی طرف یک به عنوان کلید خارجی طرف چند منظور می گردد.

مثال : در شکل زیر ،بجای ایجاد یک جدول برای ارتباط account-branch صفت branchname به جدول account اضافه می شود.



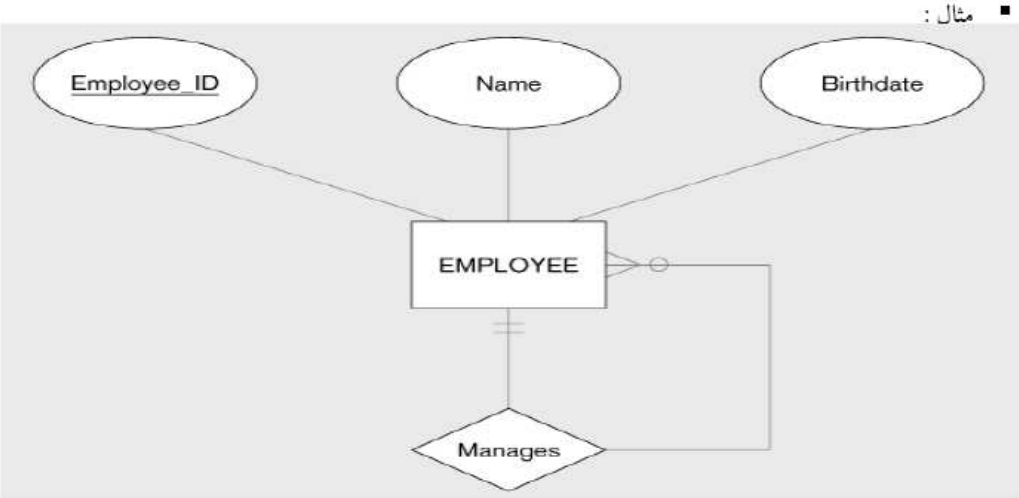
- در ارتباطات یک به یک ، هر طرفی می تواند به عنوان چند انتخاب شود.بدین معنی که کلید اصلی طرف مهمتر به عنوان کلید خارجی طرف کم اهمیت تر اضافه می شود.

قاعده ٤ : ارتباطات درجه یک(بازگشتی).

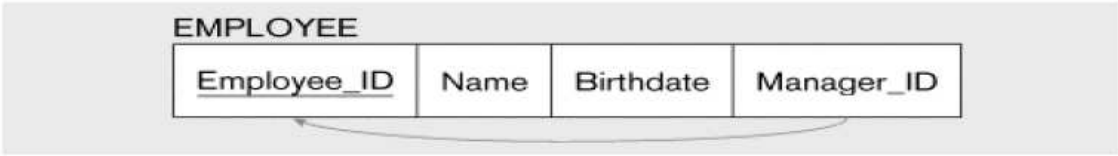
- هر ارتباط درجه یک چند به چند با یک جدول نمایش داده می شود که کلید اصلی اش دو فیلد برگرفته از کلید اصلی موجودیت است که با تغییر نام یکی از آنها ازکلید اصلی موجودیت مرتبط تشکیل می شود.

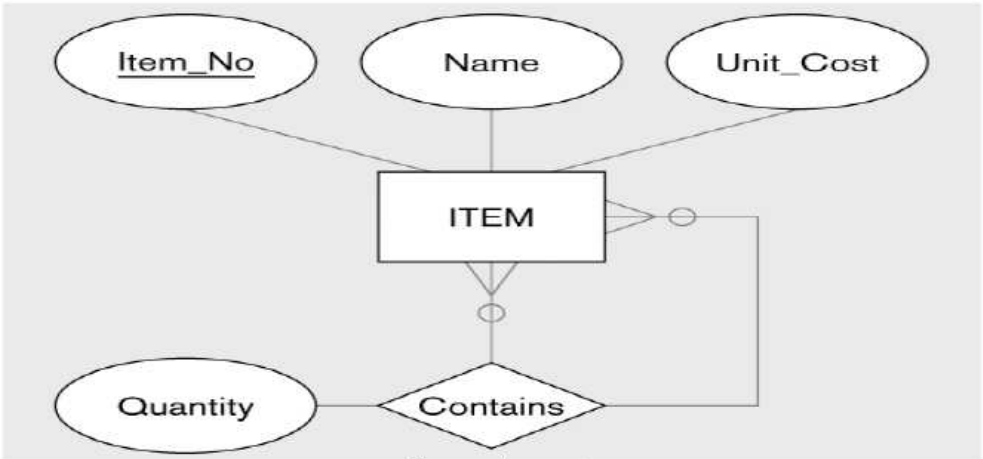
- در ارتباط بازگشتی یک به چند ، کلید خارجی بازگشتی در همان جدول مربوط به موجودیت (با تغییر نام کلید اصلی) داریم.

- مثال :

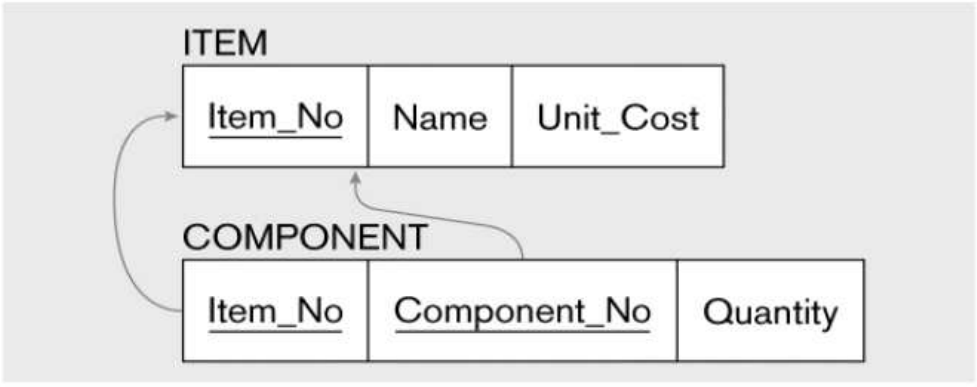


EMPLOYEE با کلید خارجی بازگشتی



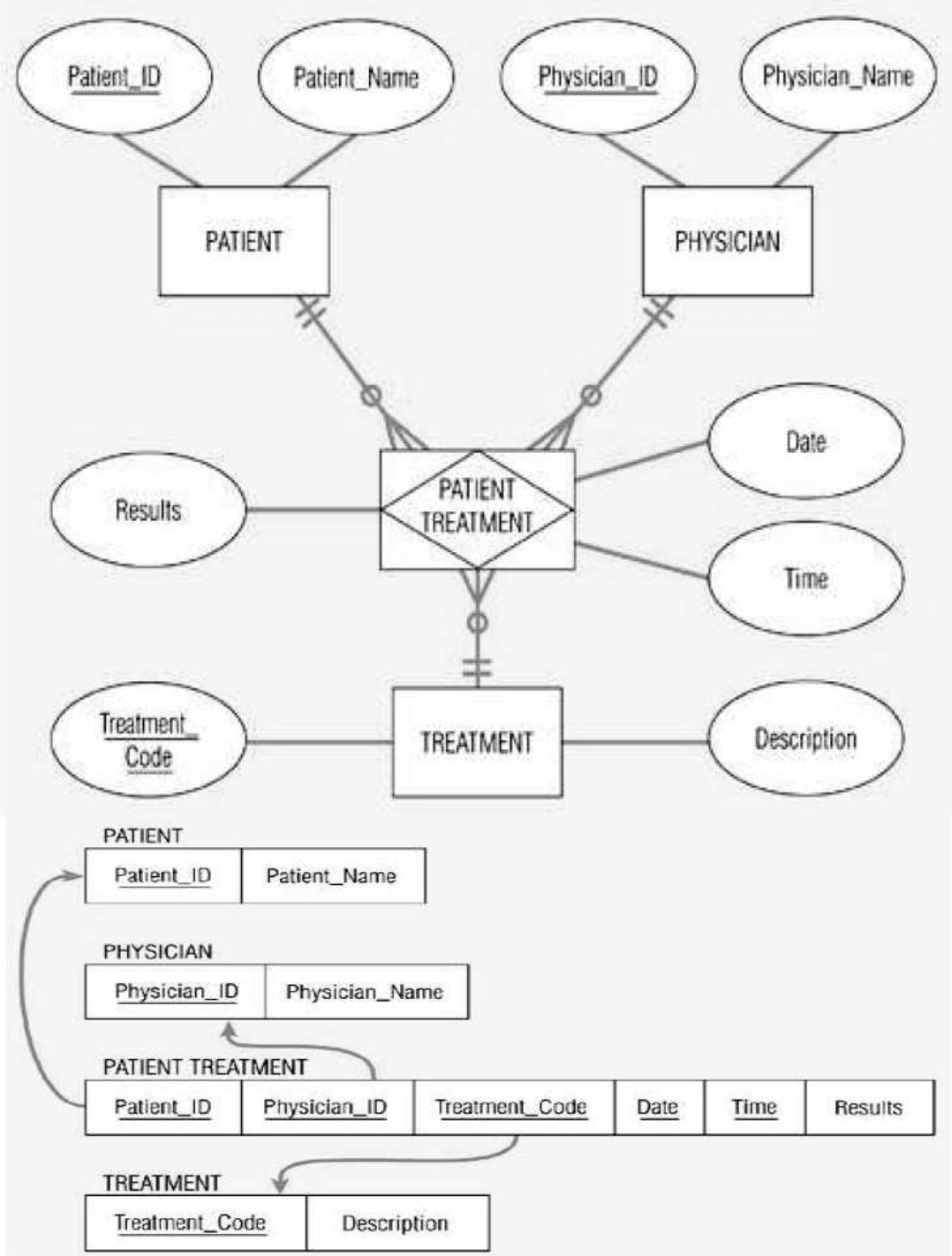


ب) دوجدول بدست آمده



قاعده ۵ : ارتباطات درجه ۳ و بیشتر .

- هر ارتباط nتایی با یک جدول نمایش داده می شود که کلید اصلی اش n فیلد برگرفته از کلید اصلی موجودیتهای مشارکت یافته در ارتباط n تایی می باشد.



قاعده ٦ : ارتباطات تعمیم/تخصیص .

- روش اول: بازای هر موجودیت سطح بالا و پایین یک جدول جداگانه در نظر گرفته شده که در جدول سطح پایین کلید اصلی موجودیت سطح بالاتر به عنوان کلید خارجی قرار می گیرد.

table table attributes
person (name, street, city)
customer (name, credit-rating)
employee(name, salary)

نکته : اشکال این روش ؟ جمع آوری اطلاعات مستلزم مراجعه به دو جدول است .

- روش دوم : بازای هر موجودیت صفات کلید اصلی و غیره موجودیت سطح بالاتر در موجودیت سطح پایینتر قرار می گیرد. در صورتی تعمیم کامل باشد نیازی به وجود جدول سطح بالاتر نخواهد بود.

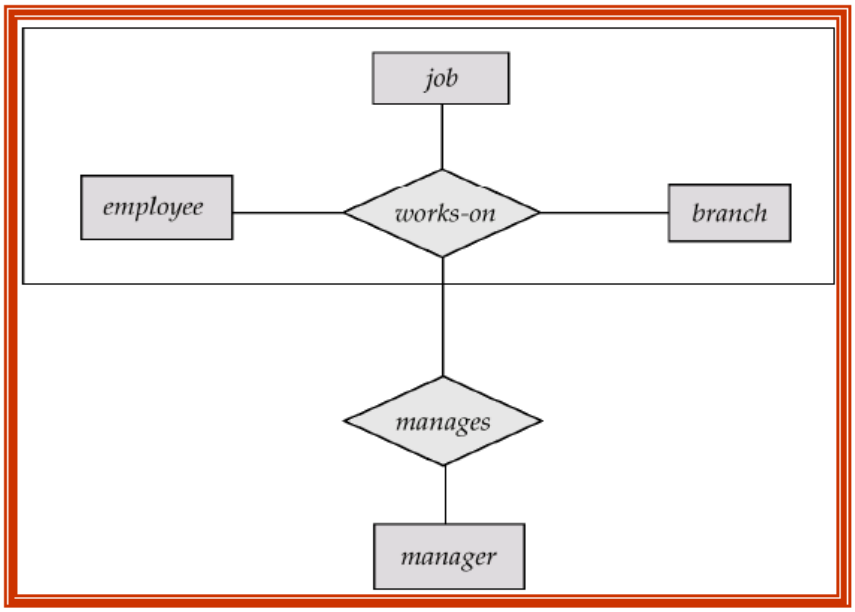
table table attributes
person (name, street, city)
customer (name, street, city, credit-rating)
employee (name, street, city, salary)

نکته : اشکال این روش ؟ افزونگی

قاعده ۷: تجمع : برای نمایش تجمع ، یک جدول شامل کلید اصلی ارتباط تجمع شده ، کلید اصلی مجموعه موجودیت مشارکت یافته ،صفات مورد نظر ایجاد می شود.

مثال : رابطه تجمع manages بین ارتباطات manager , Works-on بصورت یک جدول زیر است :

manages(employee-id, branch-name, title, manager-name)

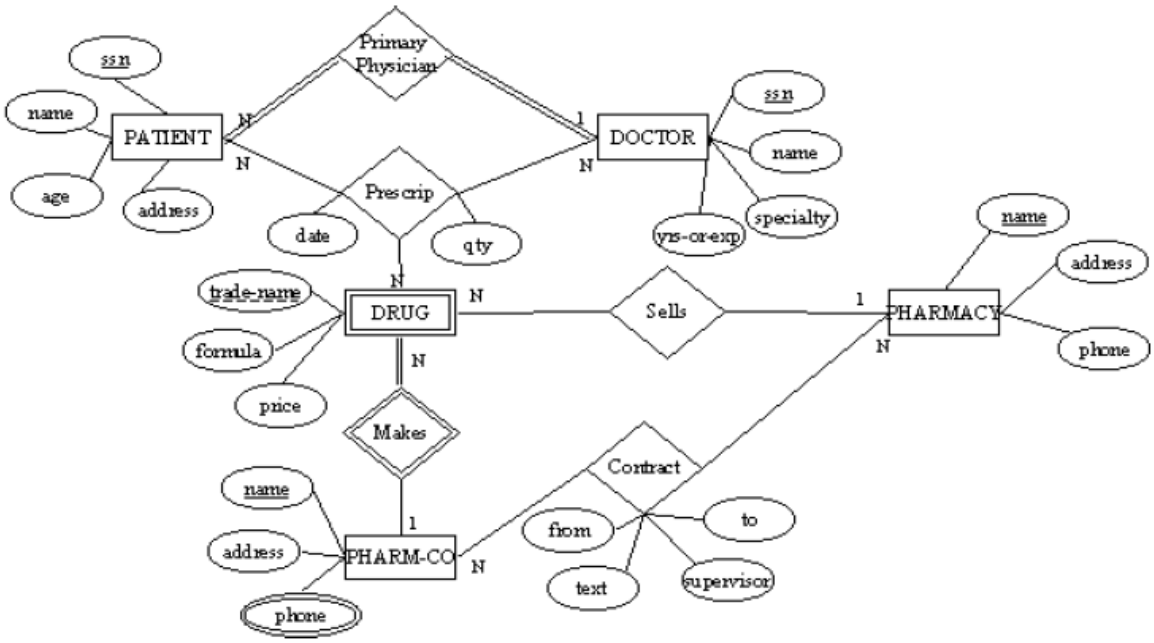


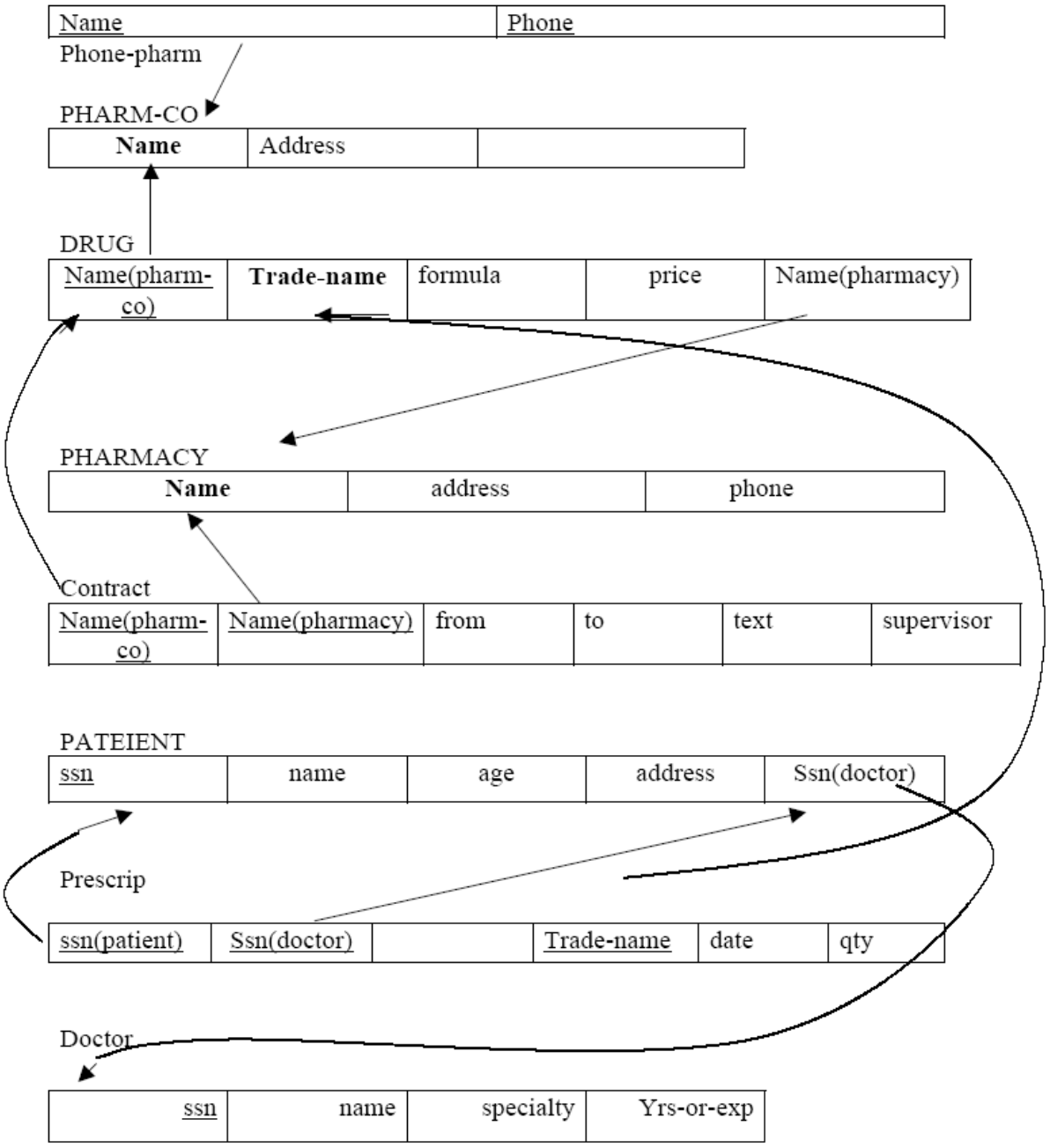
مثال: ۴- نمودار E/R زی‌ر را در نظر بگی‌ری‌د سپس جداول پای‌گاه داده را بر اساس نمودار استخراج نمای‌ی‌د. (کلی‌د های خارجی و اصلی مشخص گردند).

PATIENT : بی‌مار DOCTOR : پزشک DRUG : دارو PHARMACY

داروخانه : PHARM-COشرکت دارو سازی

Error!




Name
Phone
Phone-pharm
PHARM-CO
Name
Address
DRUG
Name(pharm-co)
Trade-name
formula
price
Name(pharmacy)
PHARMACY
Name
address
phone
Contract
Name(pharm-co)
Name(pharmacy)
from
to
text
supervisor
PATEIENT
ssn
name
age
address
Ssn(doctor)
Prescrip
ssn(patient)
Ssn(doctor)
Trade-name
date
qty
Doctor
ssn
name
specialty
Yrs-or-exp

فصل چهارم

تعریف پایگاه داده‌ها

پایگاه داده‌ها چیست؟

اصطلاحات بانک اطلاعاتی، پایگاه داده‌ها، بانک داده‌ها، پایگاه اطلاعات معادل یکدیگر می‌باشند. بانک اطلاعاتی «مجموعه‌ای است از داده‌های ذخیره شده (در مورد انواع موجودیتهای یک محیط عملیاتی و ارتباطات بین آنها) به صورت مجتمع و مبتنی بر یک ساختار، تعریف شده به طور صوری با حداقل افزونگی، تحت کنترل متمرکز، مورد استفاده یک یا چند کاربر به طور اشتراکی و همزمان»

منظور از تعریف شده به طور صوری آن است که سیستم باید به کاربران امکان دهد تا داده‌های خود را آنگونه که خود می‌بینند، به صورت انتزاعی و بدور از جنبه‌های پیاده‌سازی و نشست فیزیکی آنها روی رسانه تعریف کنند. مجتمع و مبتنی بر یک ساختار به این معناست که کل داده‌های عملیاتی محیط موردنظر در یک ساختار مشخص به صورت یکجا ذخیره شده باشند. لازمه هر تجمعی وجود یک ساختار است.

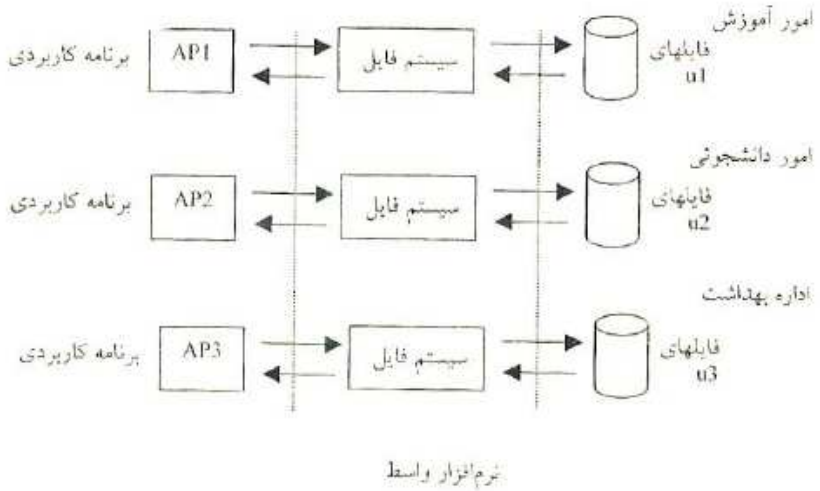
ساختار فیزیکی بانک همان ساختار فایلهایی است که آن را تشکیل می‌دهد. با مفهوم افزونگی در درس ذخیره و بازیابی آشنا شده‌اید. افزونگی عبارت است از تکرار مقادیر یک یا چند صفت خاصه در نمونه‌های مختلف یک نوع رکورد از یک فایل، به بیانی دیگر ذخیره‌سازی آن مقادیر در بیش از یک نقطه از فایل. تجمع داده‌ها و وحدت ذخیره‌سازی باعث از بین رفتن پدیده افزونگی خواهد شد یا آنرا به حداقل می‌رساند.

مثال ۱ : در محیط عملیاتی دانشگاه بخشهای امور آموزش ـ امور دانشجویی و اداره بهداشت را در نظر می‌گیریم. می‌خواهیم یک سیستم مکانیزه برای این بخشها پدید آوریم. در این مثالها تنها موجودیت دانشجو را در نظر می‌گیریم.

دانشجو دارای صفات خاصه متعددی است مثل نام ـ معدل دیپلم ـ شمارهٔ دانشجویی ـ تاریخ تولد ـ سال ورود ـ شماره دفترچه بیمه ـ وضعیت جسمانی ـ وضعیت مسکن و غیره که برای هر بخش می‌تواند قدری متفاوت باشد. برای ایجاد این سیستم دو روش کلی وجود دارد :

الف) روش کلاسیک

در این روش هر یک از بخشهای سه‌گانه به طور جداگانه سیستم خاص خود را ایجاد می‌کنند و با استفاده از امکانات سیستم فایل موجود در سیستم عامل و یک زبان سطح بالا فایلهایی را تعریف و ایجاد می‌کنند. در هر فایل رکورد موردنظر با فیلدهای موردنیاز تعریف می‌شود :



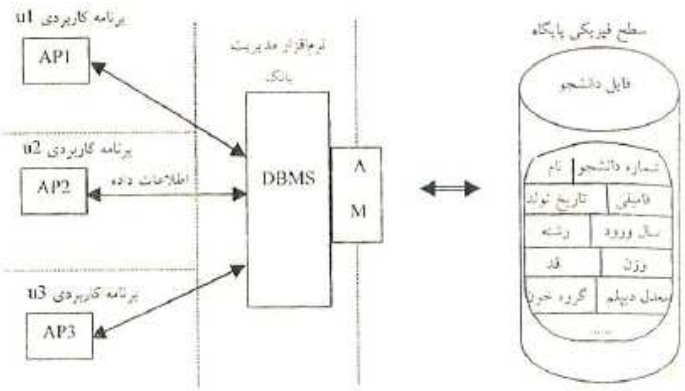
بسیاری از داده‌های مورد نیاز u1 همانهایی هستند که برای u2 یا u3 نیز مورد نیاز است. عدم تجمع داده‌های ذخیره شده و عدم وحدت ذخیره‌سازی به وضوح در این مثال مشهود است و در نتیجه پدیده افزونگی در ذخیره‌سازی داده‌ها وجود دارد.

ضمناً برنامه‌های کاربردی برای ایجاد و پردازش فایلهای خاصی نوشته شده‌اند و در صورت وجود تغییر در ساختار رکوردها و فایلها، برنامه‌ها نیز باید متناسباً تغییر کنند. در این روش امنیت و حفاظت داده‌ها مشکل است.

ب) روش بانکی

در این روش رکورد نوع دانشجو فقط یکبار در فایل ذخیره می‌شود و کاربران مختلف هر یک طبق نیاز خود از آن به طور همزمان، مشترکاً استفاده می‌کنند. در رکورد نوع دانشجو تمام صفات خاصه موردنیاز کاربران مختلف وجود دارد و طبعاً صفات خاصه مشترک، تنها یکبار در رکورد منظور می‌شوند. با آنکه در اینجا وحدت ذخیره‌سازی داریم ولی هر کاربری دید خاص خود را نسب به داده‌ها دارد. در اینجا کاربران مختلف می‌توانند به صورت همزمان با بانک کار کنند. یعنی هر کاربر بدون ایجاد محدودیت برای کاربر دیگر در هر لحظه می‌تواند بانک کار کند.

شکل زیر روش بانکی را نشان می‌دهد.



نرم‌افزار مدیریت بانک اطلاعاتی (DBMS = Data Base Management System) رابط بین برنامه‌های کاربردی و داده‌هاست و هرگونه دستیابی به داده‌ها از طریق DBMS صورت می‌گیرد و به این دلیل امنیت داده‌ها نیز در این روش زیاد است.

تذکر ۱ : همزمانی عملیات از جنبه دیگری نیز مطرح است و آن همزمانی از نظر سیستم است، یعنی فرآیندهای درون سیستمی بتوانند با همزمانی، در محیط سیستم جریان داشته باشند. اینکه تا چه حد همزمانی فرآیندها در درون سیستم امکان‌پذیر است بستگی دارد به نحوه طراحی سیستم مدیریت بانک، همچنین عملکرد سیستم عاملی که خود DBMS در محیط آن مشابه یک برنامه کاربردی اجراء می‌شود. این همزمانی به معماری سخت‌افزار کامپیوتر نیز بستگی پیدا می‌کند.

تذکر ۲ : سیستم بانک اطلاعاتی، سیستم کامپیوتری نیز بستگی پیدا می‌کند.

تذکر ۳ : به طور کلی داده‌ها در بانک اطلاعاتی، بخصوص در سیستم‌های بزرگ هم به صورت مجتمع و هم به صورت اشتراکی هستند. مفهوم مجتمع بودن این است که بانک اطلاعاتی مجموعه‌ای از فایلهاست که بخشی از اطلاعات اضافی بین آنها حذف شده است. مثلاً در بانک زیر که از دو فایل تشکیل شده است لازم نیست که در فایل وضعیت تحصیلی، محل تولد ذخیره شود چرا که در صورت نیاز با رجوع به فایل مشخصات می‌توان آن را بدست آورد.

فایل وضعیت تحصیلی	
شماره‌دانشجویی	معدل
۱۷۳	۱۸

فایل مشخصات		
شماره دانشجویی	نام	متولد
۱۷۳	علی	تهران

اگر بانک اطلاعاتی مشترک نباشد، بانک اطلاعاتی مشترک نباشد، بانک اطلاعاتی «شخصی» یا «کاربرد ویژه» نامیده می‌شود.

تذکر ۴ : بین بانک اطلاعاتی فیزیکی (مثلاً داده‌های ذخیره شده) و کاربران سیستم، لایه‌ای از نرم‌افزار وجود دارد که مدیر بانک اطلاعاتی یا کارگزار بانک اطلاعاتی (Database Server) یا سیستم مدیریت بانک اطلاعاتی (DBMS) نام دارد.

تذکر ۵ : سیستم مدیریت بانک اطلاعاتی، مهمترین جزء نرم‌افزاری در کل سیستم است. اما تنها جزء نرم‌افزاری نیست. بقیه اجزای نرم‌افزاری عبارتند از : برنامه‌های کمکی، ابزارهای تولید برنامه‌های کاربردی، ابزارهای طراحی، نویسندگان گزارش و از همه مهمتر مدیر تراکنش (Transaction Manager)

تذکر ۶ : سیستم مدیریت بانک اطلاعاتی به محصولات خاصی از فرشندگان خاص نیز اطلاق می‌شود. مثل بانک اطلاعاتی جهانی DB2 از شرکت IBM.

تذکر ۷ : داده‌ها حقایق موجود هستند که حقایق دیگری از آنها استنتاج خواهد شد. استنتاج حقایق دیگری از حقایق موجود، کاری است که سیستم مدیریت بانک اطلاعاتی هنگام پاسخ به کاربر انجام می‌دهد.

یک «واقعیت موجود» از نظر اهل منطق، گزاره درستی است، به عنوان مثال جمله «معدل دانشجو با شماره 753416 برابر ۱۷ می‌باشد» ممکن است گزاره درستی باشد. بدین ترتیب بانک اطلاعاتی «مجموعه‌ای از گزاره‌ای درست» است.

عناصر اصلی محیط بانکی

محیط بانک اطلاعاتی از عناصر اصلی زیر تشکیل شده است :

۱ـ سخت‌افزار ۲ـ نرم‌افزار ۳ـ کاربر ۴ـ داده‌ها

۱ـ سخت‌افزار محیط بانکی را می‌توان به صورت زیر تقسیم‌بندی کرد:

الف) سخت‌افزار ذخیره‌سازی داده‌ها

ب) سخت‌افزار پردازنده مرکزی

ج) سخت‌افزار ارتباطی

۲ـ نرم‌افزار محیط بانکی را می‌توان به دو دسته الف) نرم‌افزار کاربردی ب) نرم‌افزار سیستمی تقسیم‌بندی کرد.

نرم‌افزار کاربردی : نرم‌افزاری است که کاربر باید برای تماس با سیستم بانک اطلاعاتی آماده کند. این نرم‌افزار به کمک یک زبان سطح بالا و یک زبان داده‌یی (Data Language) و برخی تسهیلات نرم‌افزاری برای تماس با بانک ساخته می‌شود.

نرم‌افزار سیستمی : که از نرم‌افزار سیستمی خاص بانک (یعنی DBMS) و نرم‌افزار سیستمی عمومی (یعنی سیستم عامل) تشکیل شده است. DBMS در یک تعریف مقدماتی، سیستمی است که به کاربران امکان می‌دهد عملیات موردنظرشان را (مثل تعریف داده‌ها ـ بازیابی داده‌ها و ذخیره‌سازی داده‌ها) انجام دهند. DBMS که نرم‌افزاری پیچیده است میهمان یک سیستم عامل است و از امکانات سیستم عامل در انجام وظایفش استفاده می‌کند.

۳ـ کاربر : یکی از کاربران مهم در سیستم بانک اطلاعاتی، اداره کننده بانک اطلاعاتی Data Base Administrator (DBA) است. اداره کننده بانک فردی است که مسئولیت ایجاد، پیاده‌سازی و نگهداری بانک را در محیط عملیاتی بر عهده دارد.

کاربر دیگر برنامه‌نویس یا DBP می‌باشد. کاربر نهایی (End User) فردی است که از برنامه‌های نوشته شده استفاده می‌کند.

۴ـ داده : منظور از داده در اینجا داده‌هایی است که در مورد موجودیتهای مختلف محیط عملیاتی، می‌خواهیم ذخیره کنیم و نیز ارتباط بین انواع موجودیتها و اصطلاحاً به آن «داده‌های عملیاتی» می‌گوئیم.

تذکر : داده‌های عملیاتی با داده‌های ورودی و خروجی تفاوت دارند. داده‌های ورودی اطلاعاتی هستند که نخستین بار وارد سیستم شده می‌توانند سبب ایجاد تغییر در داده‌های عملیاتی شوند یا خود جزیی از داده‌های عملیاتی محیط گردند.

داده‌های خروجی عبارتند از پیام‌ها، پاسخ‌ها و نتایجی که سیستم پیرو درخواست کاربر به او می‌دهد. این داده‌ها می‌توانند از داده‌های عملیاتی استخراج شوند ولی خود بخشی از داده‌های عملیاتی تلقی نمی‌شوند.

بانک اطلاعاتی توزیع شده

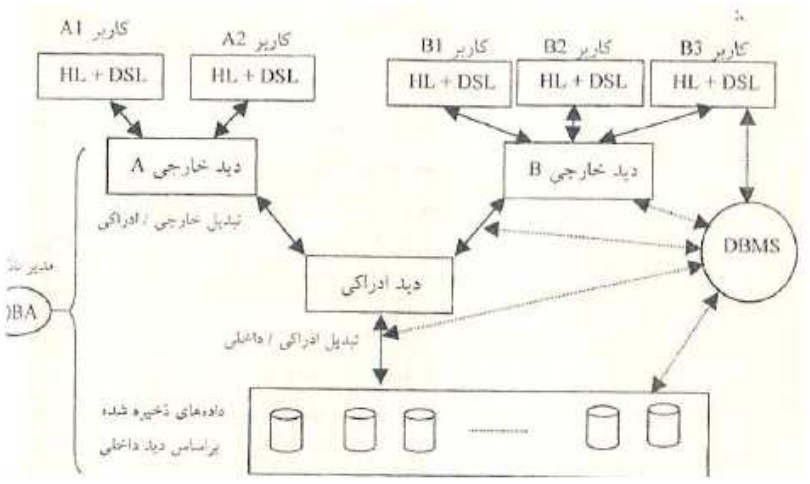
بانکهای اطلاعاتی را می‌توان در یک شبکه مورد استفاده قرار داد که خود دو نوع می‌باشند :

۱ـ بانکهای اطلاعاتی با داده‌های توزیع شده : در این شبکه یک کامپیوتر به عنوان کامپیوتر اصلی است که هدایت کل شبکه و مدیریت بانک اطلاعاتی را بر عهده دارد ولی داده‌ها در قسمتهای مختلف شبکه پخش شده‌اند.

۲ـ بانکهای اطلاعاتی با داده‌ها و سیستم توزیع شده : در این سیستم خود مدیریت بانک اطلاعاتی به صورت توزیع شده در چند کامپیوتر جای دارد یعنی پردازشها ممکن است در کامپیوترهای مختلف صورت گیرد و داده‌ها نیز در شبکه پخش می‌باشند. این نوع بانک در سیستم عاملهای توزیع شده مثل UNIX یا Windows NT قابل پیاده‌سازی است.

معماری سیستم بانک اطلاعاتی

مدل پیشنهادی ANSI (ارائه شده در سال ۱۹۷۵) برای معماری سیستم بانک اطلاعاتی به شکل زیر است :



معماری سیستم بانک اطلاعاتی از اجزاءزیر تشکیل شده است :

الف) دید ادراکی یا مفهومی (Conceptual View)

ب) دید خارجی (External View)

ج) دید داخلی یا فیزیکی (Internal View)

د) تبدیلات بین سطوح (Transformation یا Mappings)

هـ) زبان میزبان یا HL (Host Language)

و) زبان فرعی داده‌یی یا DSL (Data Sub Language)

به علاوه در چنین سیستمی سه عنصر مهم دیگر نیز وجود دارند :کاربر ـ DBA – DBMS

الف) دید ادراکی (مفهومی)

دید طراح بانک است از داده‌های ذخیره در بانک. یعنی داده‌های انواع موجودیتها و ارتباط بین آنها،آنگونه که طراح می‌بیند. دید طراح دیدی است جامع دیدهای همه کاربران و در عین حال متفاوت با هر یک از دیدهد. در این سطح ساختار داده‌ها مطرح می‌شود.

مثال ۲ : در محیط عملیاتی دانشگاه موجودیتهای درس و دانشجو را در نظر می‌گیریم. فرض می‌کنیم ساختار بانک به صورت جدولی باشد. طراح برای هر نوع موجودیت یک جدول طراحی می‌کند و برای نمایش ارتباط بین موجودیتها نیز یک جدول دیگر. مثلاً به صورت زیر :

به این ترتیب شمای ادراکی طراح بانک ادراکی به احکام کار با بانک مانند عملگرهای جدول‌یاب، جستجو، درج و حذف سطر نیاز داریم در نتیجه در شمای ادراکی کل داده‌های عملیاتی و ارتباطات بین آنها تعریف و تشریح می‌شود. شمای ادارکی به عنوان مجموعه‌ای از احکام یک برنامه است که باید کامپایل و تبدیل به برنامه مقصد شود.

جدول دانشجویی STTAB

سال ورود	نام خانوادگی	نام	شماره دانشجویی

جدول درس COTAB

ماهیت درس	تعداد واحد	نام درس	شماره درس

جدول دانشجو ـ درس STCOTAB

نمره	ترم	شماره درس	شماره دانشجو

تذکر : در سطح ادراکی ارتباط موجودیتها و صفات خاصه، امنیت و جامعیت داده‌ها و اطلاعات معنایی داده‌ها مطرح می‌گردد.

ب) دید خارجی

دید خاص کاربر است از داده‌های ذخیره شده در بانک. هر کاربر دید خاص خود را دارد. همچنین چند کاربر می‌توانند دارای دید یکسانی باشند. مثل A1 , A2 که دید خارجی A را

دارند. همانند دید ادراکی، دید خارجی نیز برای معرفی شدن نیاز به یک ساختار یا مدل داده‌یی دارد. مدل داده‌یی مورد استفاده در سطح خارجی معمولاً همان سطح ادارکی است.

مثال ۳ : کاربر A1 می‌تواند دید روبر را روی جدول STTAB داشته باشد :

STTAB1

سال ورود	شماره دانشجویی

یا کاربر B2 دید روبرو را می‌تواند روی جدول STCOTA داشته باشد :

STCOTAB1

شماره درس	شماره دانشجویی

پس در دید خارجی می‌توان شرط‌هایی قرار داد تا سطرهایی (یا ستونهایی) از جدول مبنا در دید کاربر قرار گیرد. شمای خارجی هر کاربر نیز مثل شمای ادراکی، به هر حال یک برنامه است که باید کامپیایل شود. سیستم مدیریت بانک اطلاعاتی برای انجام درخواستهای یک کاربر باید به شمای خارجی مراجعه کند.

لایه خارجی تنها لایه‌ای است که کاربران با آن سر و کار دارند لایه‌ای دیگر به مدیر و برنامه‌سازان بانک مربوط می‌شود. طبق قانون «پنهان‌سازی اطلاعات» که می‌گوید «به هر کس

به همان اندازه اطلاعات بده که نیاز دارد و نه بیشتر»؛ در لایه خارجی دیدهای مختلف کاربران مطرح است تا هر کدام بخشی از بانک را که نیاز دارند ببینند.

تذکر : سطح خارجی نزدیک‌ترین سطح به کاربران است.

ج) دید داخلی

در این سطح در واقع فایلهای محیط فیزیکی تعریف می‌شود. از نظر محتوا، ساختار و استراتژی دستیابی. این تعریف در اساس همانست که در محیط‌های غیربانکی برای ایجاد فایلها لازم است.

یک سیستم بانک اطلاعاتی، کاربران اساساً به مسائل این سطح نمی‌پردازند. در سیستم‌های موجود طراح بانک دخالت چندانی در این سطح ندارد ولی در بعضی دیگر تا حد زیادی به جنبه‌های خاص این سطح می‌پردازد. در شمای داخلی، انواع رکوردها، فایلها، صفات خاصه شاخص (استراتژی دستیابی)، نحوه نمایش و تشریح رکوردهای ذخیره شده در فایل، توالی رکوردها، تخصیص فضای ذخیره‌سازی برای داده‌ها، محل رکورد، فشردگی داده‌ای و تکنیکهای رمزگذاری داده‌ها تشریح می‌شوند. مباحث این سطح مربوط به درس ذخیره و بازیابی اطلاعات است و ما در این درس به این مسئله نمی‌پردازیم.

تذکر : سطح داخلی نزدیکترین سطح به رسانه ذخیره‌سازی فیزیکی است.

کاربران بانک اطلاعاتی

کاربران بانک اطلاعاتی را می‌توان به سه دسته تقسیم کرد : ۱ـ برنامه‌نویسان کاربردی (DBP). برنامه‌ها می‌توانند به صورت دسته‌ای (Batch) و یا به صورت پیوسته (Online) باشند که کاربران می‌توانند از یک ترمینال Online به آن دستیابی داشته باشند. ۲ـ کاربران نهایی که از طریق یکی از برنامه‌های کاربردی Online می‌توانند به بانک اطلاعاتی دستیابی داشته باشند. ۳ـ مدیر بانک (DBA)

تذکر : کاربران نهایی (End Users) همچنین می‌توانند از طریق «پردازنده زبان تقاضا» مثل SQL که بخشی از نرم‌افزار سیستم بانک اطلاعاتی است به بانک دسترسی داشته باشند.

تفاوت DA با DBA

مدیر داده‌ها یا (Data Administrator) شخصی است که کنترل مرکزی داده‌ها را در سازمان به عهده دارد. این فرد لازم است مفهوم داده‌ها را درک کند و نیاز موسسه به داده‌ها را در سطح مدیریت عالی قرار دهد.

مدیر داده‌ها تصمیم می‌گیرد که چه داده‌هایی از همان اول در بانک اطلاعاتی قرار گیرد و پس از ذخیره آنها، سیاستهایی را برای دستیابی به آنها تنظیم کند. توجه کنید که مسئول داده‌ها یک مدیر است نه یک نفر فنی.

مدیر بانک اطلاعاتی یا (DataBase Administrator) یک شخص فنی است که مسئول پیاده‌سازی تصمیمات مدیر داده‌هاست. DBA برخلاف DA، یک فرد حرفه‌ای در تکنولوژی اطلاعات (IT) می‌باشد. وظیفه DBA ایجاد بانک اطلاعاتی و پیاده‌یازی کنترلهای فنی است که سیاستگذاری مدیر داده (DA) را اعمال کند. همچنین DBA می‌بایست تضمین کند که

سیستم با کارآیی قابل قبولی کار کند. DBA مجموعه‌ای از برنامه‌نویسان و سایر افراد فنی را در اختیار دارد.

منابع:

- **کتاب پایگاه داده های آقای دکتر رانکوهی**
- **کتاب پایگاه داده های آقای مهندس جعفرنژاد قمی**
- **جزوه خلاصه درس اصول طراحی پایگاه داده های آقای مهندس بهروز معصومی**