

پایگاه داده

استاد درس: نسرین جمشیدی صالح
دکتری هوش مصنوعی و رباتیکز

فصل اول: مفاهیم اولیه

- ▶ مفاهیم امروزی در داده‌ها
- ▶ مفهوم تراکنش
- ▶ ویژگی‌های تراکنش‌ها
- ▶ ایمنی و امنیت اطلاعات و خصوصی سازی

مفاهیم امروزی در داده‌ها:

تعریف داده (data): هر موجودیت قابل مشاهده - اطلاعات خام و تغییر نیافته

تعریف کلان داده‌ها (big data): مجموعه‌ای بزرگ داده‌ها و ترکیبی از داده‌های ساختاریافته و یا نیمه ساختاریافته و یا بدون ساختار و دارای ارتباطات پیچیده میان داده است. - پردازش آنها با روش‌های سنتی غیرممکن است.

ویژگی‌های کلان داده‌ها:

۱. حجم داده: سازمانها داده‌ها را از منابع مختلف شامل تراکنش‌های کاری، ویدیو و تجهیزات و شبکه اجتماعی و ... جمع آوری می‌کنند. نگهداری این حجم اسلعات بسیار مشکل است که امروزه با پلتفرم‌هایی مانند **Hadoop , data lakes** انجام می‌شود
 ۲. سرعت: اهمیت سرعت پردازش این داده‌ها بسیار مهم است و استفاده از سنسورها و معیارهای هوشمند ، استفاده از بیگ دیتا را در کمترین زمان ممکن می‌سازد.
 ۳. تنوع: تنوع در فرمت های داده ها مانند داده های عددی، بانک های اطلاعات قدیمی، اسناد متنی ایمیل ها و ...
 ۴. پیچیدگی: پیچیدگی زیاد
- مثال: مجموعه داده‌های تعداد زیادی از بیماران مبتلا به کرونا در سنین مختلف و گروه‌های متنوع - سیستم پیشنهادات محصول با استفاده از کلان داده‌ها



► قاعده 3v:

► تعریف علم داده (data science) : تجزیه و تحلیل داده‌های کسب و کار و استخراج اطلاعات مفید و بینش‌های مخفی در داده‌ها، برای کمک در تصمیم‌گیری و برنامه‌ریزی - مهارت در مدیریت و تحلیل داده‌های کلان دارد.



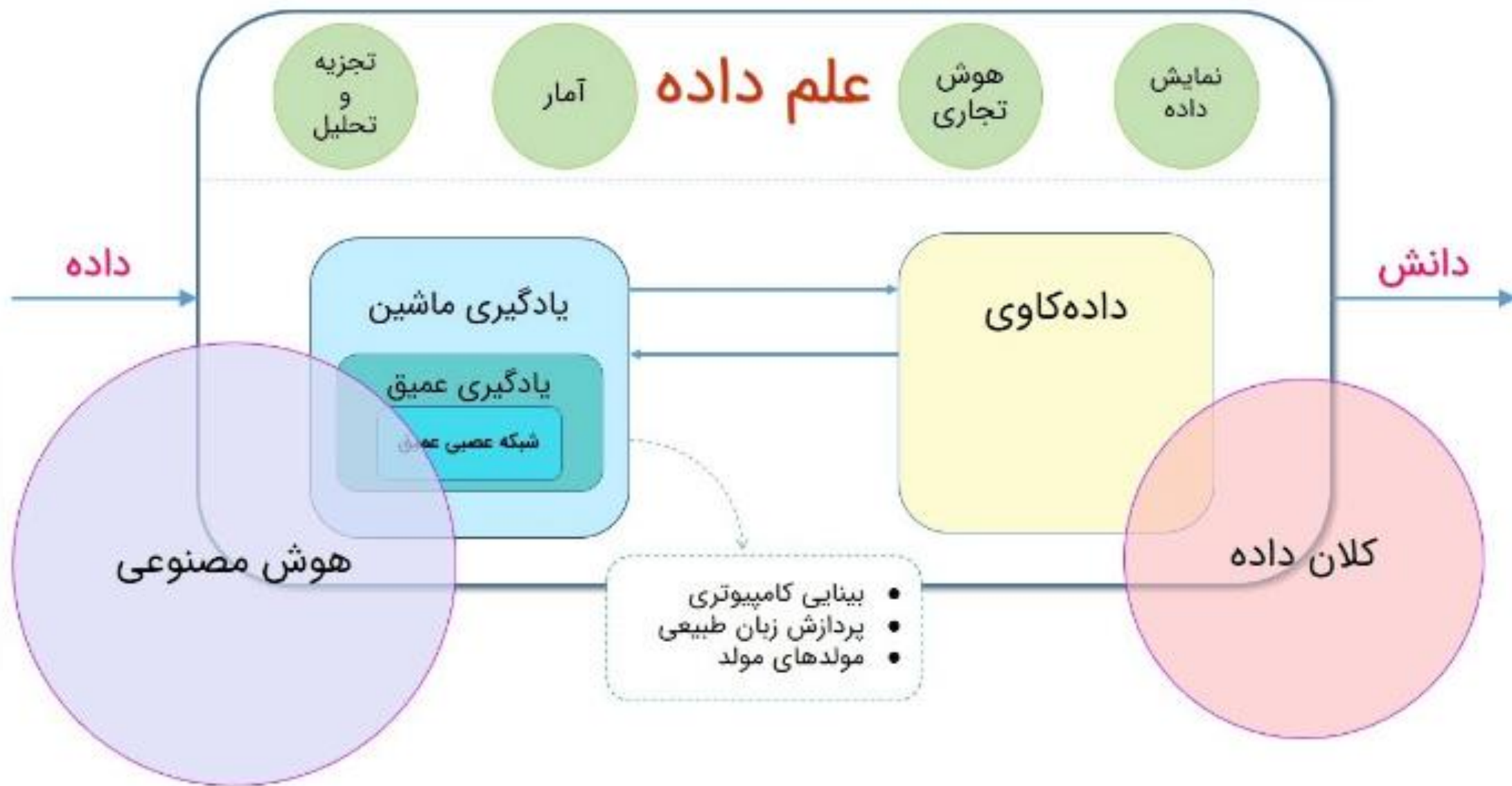
► تعریف اطلاعات (information) : داده‌های پردازش شده-داده‌های سازماندهی شده و آراسته و معنادار و تفسیر شده

► تعریف دانش: (knowledge) : نمایش نمادین بخش کوچکی از جهان واقعی

► تعریف پایگاه داده: مجموعه‌ای از داده‌های ذخیره شده و پایا به صورت مجتمع و بهم مرتبط ، تحت مدیریت یک سیستم کنترل متمرکز ، مورد استفاده یک یا چند کاربر به صورت همزمان و اشتراکی را پایگاه داده گویند.

نکته مهم : داده‌ها از روی تجزیه و تحلیل دانش بدست می‌آیند





مفهوم تراکنش: مجموعه‌ای از رخدادهای پشت سر هم روی داده‌ها به طوری که از یک نقطه آغاز و به یک نقطه پایان می‌یابد.

▶ برگرداندن تراکنش (Rollback)

▶ اجرای ناقص تراکنش

▶ تأیید تراکنش (commitment)

ویژگی‌های تراکنش:

یکپارچه بودن: (یا همه یا هیچ)

پایداری: اجرای تراکنش باید بتواند سیستم پایگاه داده را از وضعیت پایداری به یک وضعیت پایدار دیگر انتقال دهد.

ایزوله بودن: نتایج حاصل از اجرای یک تراکنش تا زمانی که اتمام تراکنش تأیید نشده نباید در سیستم پایگاه داده‌ها قابل رویت باشد بلکه باید پس از تأیید تراکنش قابل رویت باشند.

مانا بودن نتیجه اجرای تراکنشها.



ایمنی و امنیت اطلاعات و خصوصی سازی:

▶ ایمنی اطلاعات (safety information) : مکانیزمی غیرانسانی

▶ امنیت اطلاعات (security information): مکانیزمی حفاظتی برای کنترل و نظارت دسترسی کاربران مجاز به استفاده از سیستم اطلاعاتی

▶ خصوصی سازی (privacy information): قابل دسترسی بودن اطلاعات یک شخص یا یک سازمان برای سایرین با سطوح اطلاعاتی متفاوت. - خصوصی سازی بیماری بیماران برای استفاده فقط پژوهشگران (نه تاریخ تولد بیمار و اطلاعات جامعشان)

تمرین :

فلوچارت مراحل اجرا و خروجی یک تراکنش را رسم کنید.

فصل دوم: روش‌های پیاده‌سازی سیستم اطلاعاتی

- ▶ روش اول : سیستم‌های فایلینگ
- ▶ روش دوم: روش مبتنی بر پایگاه داده‌ها
- ✓ تعریف پایگاه داده
- ✓ عناصر اصلی سیستم مدیریت پایگاه داده‌ها
- ✓ افزونگی
- ✓ مزایای استفاده از سیستم‌ای مبتنی بر پایگاه داده

روش اول : سیستم‌های فایلینگ

- ▶ سیستم ناپایگاهی: در این روش نیازهای اطلاعاتی و پردازشی هر قسمت از محیط برآورده می شوند. مراحل اولیه طراحی و تولید برای هر قسمت به طور کلاسیک انجام شده و بعد از طراحی، مشخصات هر سیستم همراه با وظایف آنها مشخص می شود .
- ▶ در این روش برای ایجاد محیط ذخیره سازی اطلاعات از یک سیستم فایل (FS) و برای برنامه سازی از یک زبان سطح بالا استفاده می شود و در نهایت برای هر قسمت، یک سیستم کاربردی ایجاد می شود.



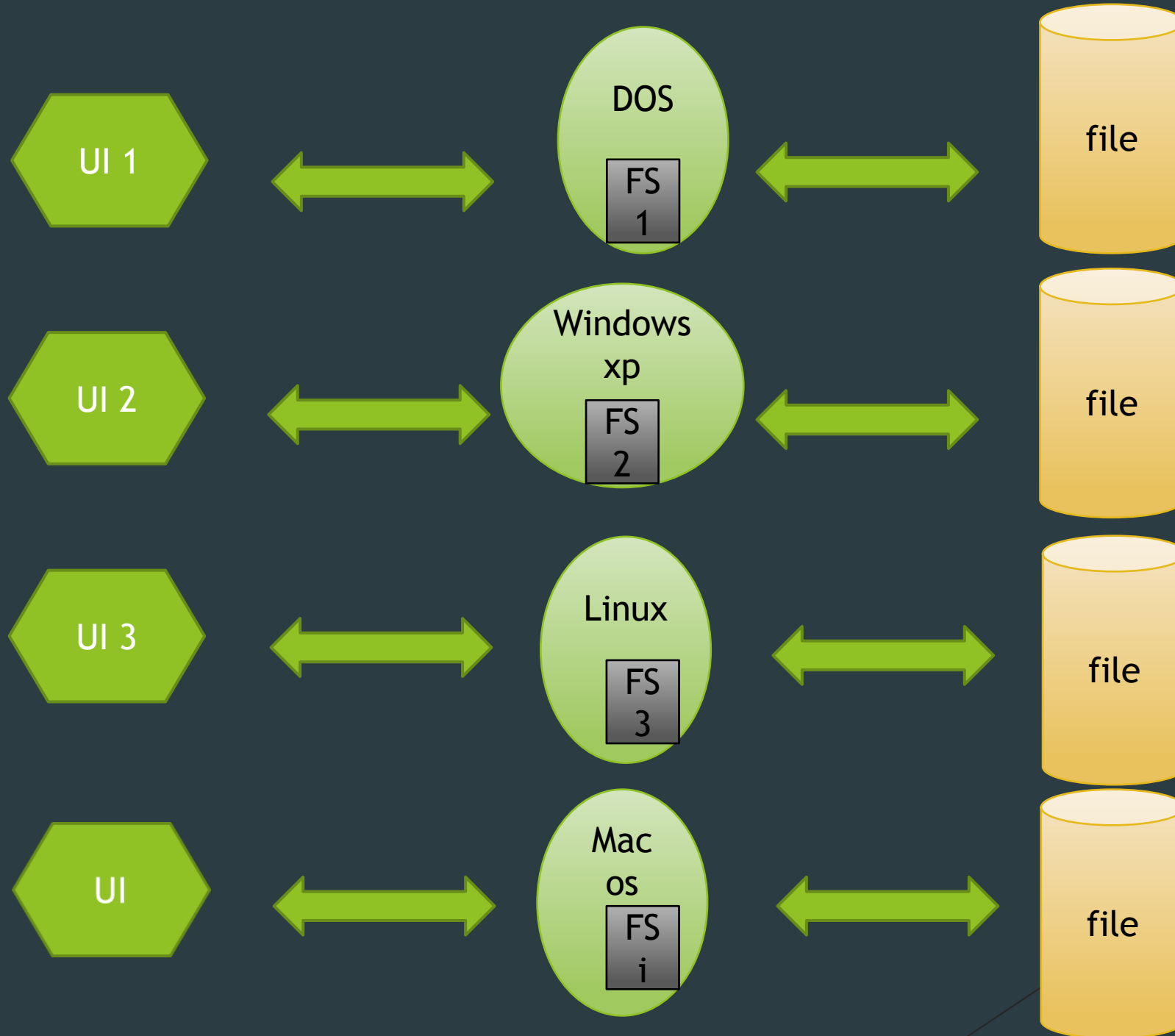
معایب روش فایلینگ:

۱- ناسازگاری داده‌ها:

مثال : شماره پرسنلی در یک سیستم درمانی : کاراکتری
اما در سیستم دیگری مانند حسابداری : عدد صحیح

۲- عدم داشتن کنترل متمرکز:

مثال: سیستم DOS، ۱۶ بیتی است در حالی سیستمی مانند ویندوز ۸ ، ۶۴ بیتی . در این
زمان عملیات درج و حذف و ویرایش اطلاعات مشکل خواهد بود.



۳- مشکل بروز نبودن اطلاعات:

مثال : استعمال موجودی کالا در دو زمان متفاوت

۴- پایین بودن ضریب ایمنی: به جهت محدود بودن مکانیستم‌های ایمنی، ضریب ایمنی آن پایین است . حتی استفاده از انواع اقسام ایمنی ها ، باز هم به علت سیستم فایلینگ مبتنی بر معماری کنترل نامتمرکز، مکانیسم حفاظتی نمی‌تواند به قدر کافی تامین کننده ضریب ایمنی بالا برای سیستم اطلاعاتی باشند.

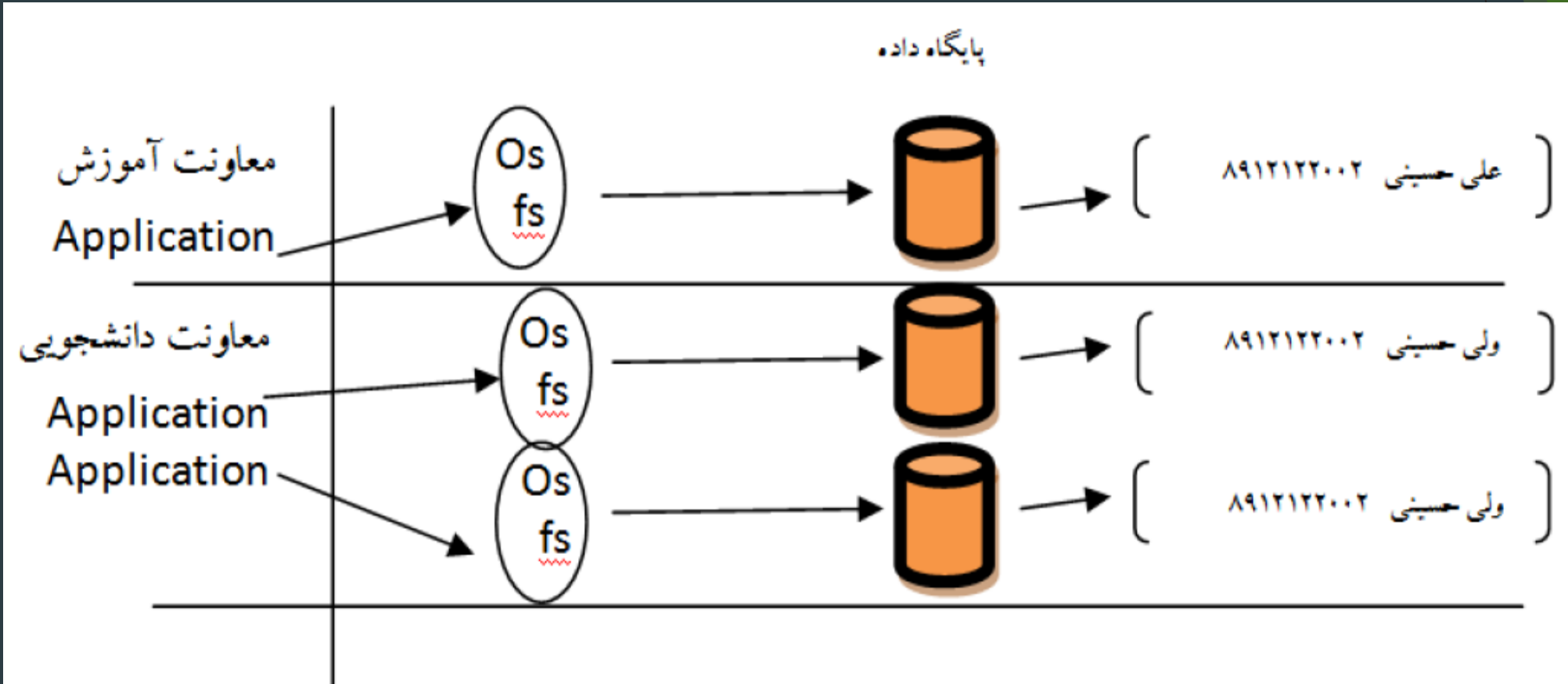
۵ - پایین بودن ضریب امنیتی : امنیت اطلاعات در سیستم فایلینگ بیشتر وابسته به نوع سیستم عامل مربوط به هر یک از زیرسیستم هاست. باز به علت نبودن سیستم کنترل متمرکز برای تعیین و مدیریت حقوق کاربری کاربران ، مدیریت امنیت اطلاعات ذخیره شده در سیستم فایلینگ کارآمد نیست.

۶- استفاده ناکارآمد از منابع

۷- سختی در توسعه سیستم و بهینه سازی سیستم

یکی از مشکلات در سیستم فایلینگ

توجه : در شکل زیر مشاهده می کنید که نام علی حسینی و در قسمت دیگر با نام ولی و با شماره دانشجویی یکسان ثبت شده است که این اشتباه هنگام ثبت نام رخ داده است ولی در هر قسمت این دانشجو با نام دیگر شناخته می شود.

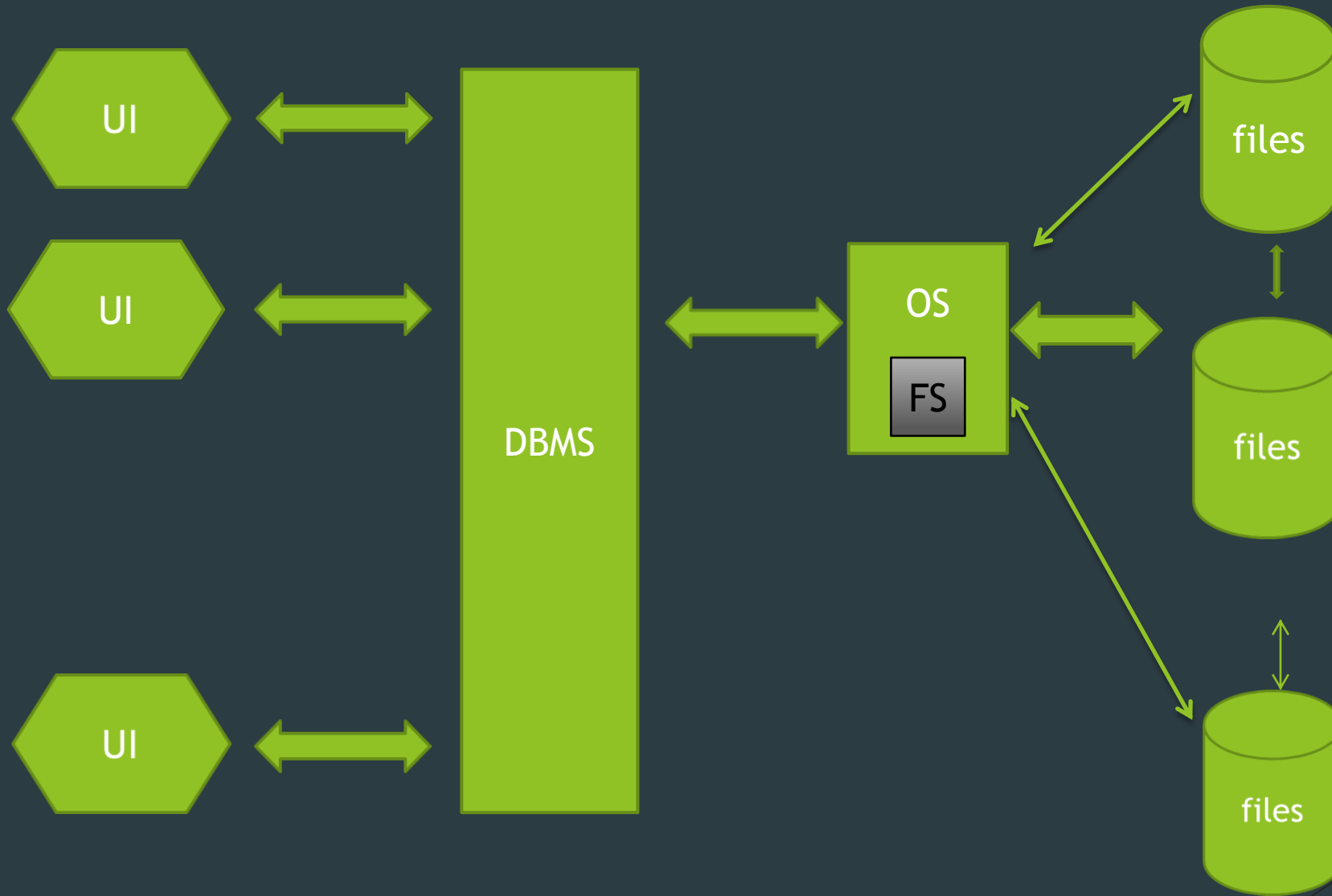


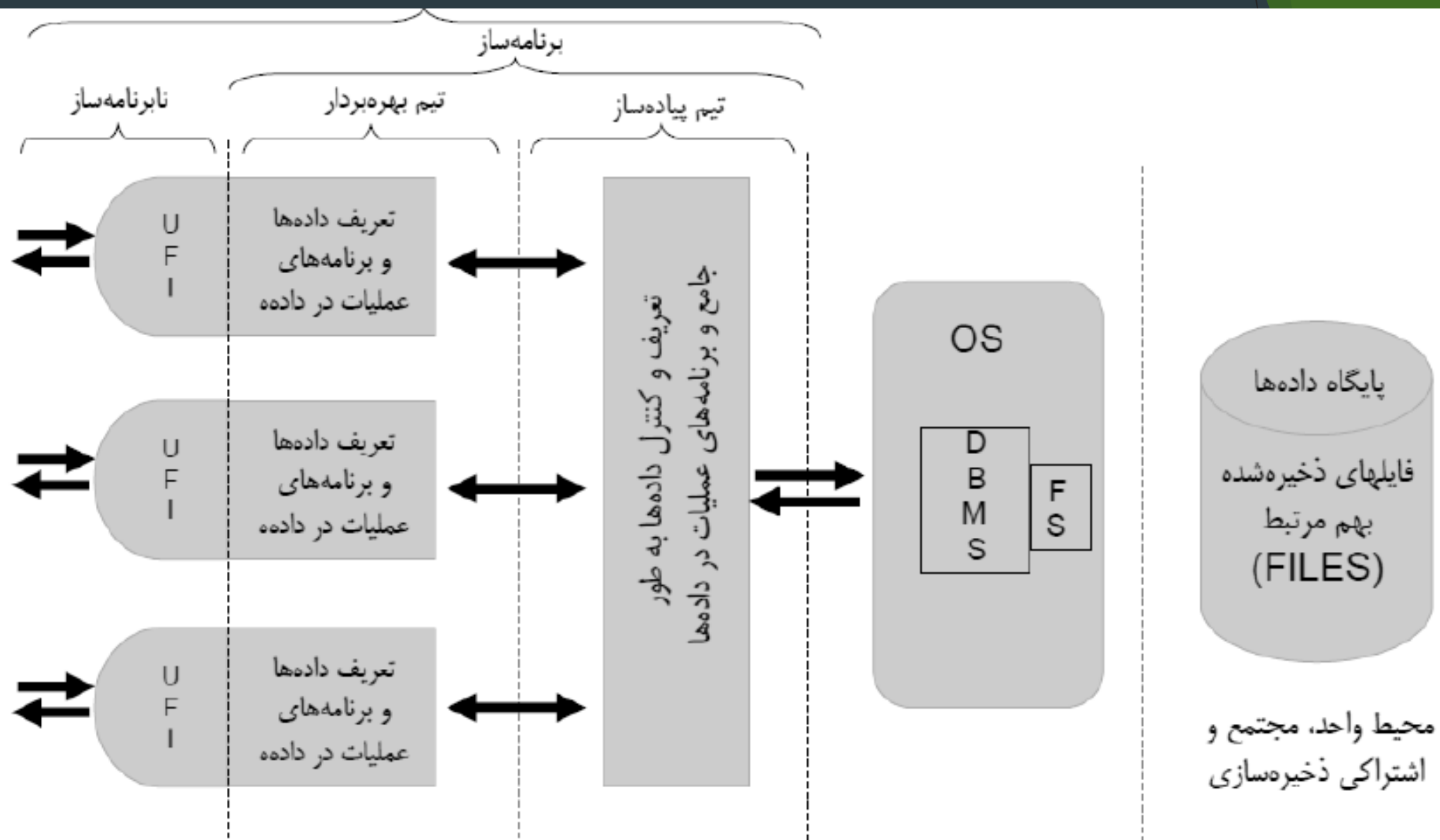
روش دوم: روش مبتنی بر پایگاه داده‌ها

- ▶ در این روش یک سیستم کنترل متمرکز وجود دارد که وظیفه مدیریت اطلاعات را بر عهده دارد.
- ▶ این سیستم مدیریت اطلاعاتی در واقع یک **نرم افزار تجاری** است که امکان طراحی، پیاده‌سازی و توسعه سیستم‌های اطلاعاتی مفید را برای متخصصان پایگاه داده فراهم می‌کند.
- ▶ متخصص ابتدا بر اساس نیاز اطلاعاتی کاربران، یک مدل اطلاعاتی را برای سیستم طراحی می‌کند و در نهایت آن را روی یک نرم افزار **DBMS** پیاده‌سازی می‌کند. در واقع این نرم افزار بر اساس طراحی انجام شده یک پایگاه داده را تولید می‌کند.

Data Base Management System : DBMS ▶

معماری عمومی سیستم مبنی بر پایگاه داده





تعریف پایگاه داده: به مجموعه ای از داده های به هم مرتبط پایگاه داده می گویند که لزوماً این داده ها از نظر فیزیکی در یک مکان قرار ندارند ولی با هم ارتباط منطقی دارند.

► مجموعه ای از اطلاعات ذخیره شده و سازماندهی شده به طوری که امکان دسترسی، تغییرات، نظارت، مدیریت و کنترل را روی این اطلاعات ذخیره شده به صورت کارا میسر سازد.

عناصر اصلی سیستم مدیریت پایگاه داده‌ها :

► مبتنی بر چهار عنصر اصلی که عبارتند از :

۱- سخت افزار ۲- نرم افزار ۳- داده‌ها ۴- کاربران

اجزای سخت افزاری :

۱- واحد ذخیره و بازیابی اطلاعات انبوه

۲- واحد پردازشگر مرکزی

۳- شبکه های ارتباطی

۴- حافظه اصلی و حافظه پنهان

► واحدهای ذخیره و بازیابی اطلاعات انبوه:

- هدف آنها : ذخیره حجم زیادی از اطلاعات و بازیابی آنها
- در گذشته متشکل از نوارهای مغناطیسی و دیسک های مغناطیسی بوده و اخیرا از SSD ها استفاده می شود.
- دیسک های سخت یا مغناطیسی قابلیت ذخیره سازی بالایی دارند
- دستگاه های ذخیره ساز انبوه در مقایسه با سایر حافظه ها از سرعت پایینی برخوردار هستند . بدین منظور از همه موارد مطرح شده استفاده می شود.
- بعدها با ارائه معماری حافظه سلسله مراتبی که در آن از همه انواع حافظه استفاده می شود، یک سیستم کارا برای ذخیره سازی و بازیابی اطلاعات ارائه شد.

► واحد پردازشگر مرکزی:

چه چیزی در پردازشگر ها مهمه؟؟؟

۱- سرعت پردازش اطلاعات

۲- کاهش زمان پاسخگویی به کاربران

در سیستم های مبتنی بر پایگاه داده ها، بنا به نیاز (تعداد کاربران همزمان، تعداد تراکنش های هم زمان و حجم اطلاعات ذخیره شده ، از سیستم های تک پردازنده و یا چند پردازنده با توانایی های متعدد استفاده می شود. - مانند دو پردازنده، هشت پردازنده، چهار پردازنده.

تعریف پردازنده ها چند هسته ای (multi-core processor): قابلیت پردازش همزمان چند واحد کاری کوچک (نخ یا thread) را دارد.

► نرم افزارها :

۱- نرم افزار سیستمی (DBMS) : یا همان سیستم مدیریت پایگاه داده‌ها است. نرم افزاری که از نظر منطقی متمرکز است و وظیفه آن فراهم کردن امکانات لازم برای تعریف، ایجاد، تغییر و نگهداری و کنترل پایگاه داده‌ها است.

۲- نرم افزار رابط کاربری (User interface): یک محیط کاربر پسند برای کاربران فراهم می کند تا آن ها بتوانند از پایگاه داده ها به صورت مطلوبی استفاده کنند. در واقع هدف ، ایجاد رابط های کاربری مناسب برای کار کاربران است.

داده‌ها: ►

۱- داده‌های سیستمی یا دادگان (Meta Date)

۲- داده‌های کاربردی یا ورودی/خروجی (input/output Data)

▶ کاربران (Users): بر اساس **کارآیی** یک سیستم اطلاعاتی و با در نظر گرفتن نیازهای کاربران آنها سنجیده و تقسیم بندی می شود.

- ۱- زمان پاسخ : مدت زمانی که کاربران در انتظار دریافت تقاضای پاسخ اش است.
- ۲- در دسترس بودن: فراهم کردن امکاناتی که منجر شود که پایگاه داده ها تحت هر شرایطی در دسترس کاربران آنها باشد. (دسترسی در هفت روز هفته و ۲۴ ساعت شبانه روز)

تقسیم بندی کاربران بر اساس ماهیت اطلاعاتی :

- ۱- مدیر داده ها یا (Data Administrator) DA
- ۲- کاربر نهایی یا (End-User) EU
- ۳- طراح پایگاه داده ها یا (Database Designer) DD
- ۴- مدیر پایگاه داده ها یا (Database Administrator) DBA

۱- مدیر داده‌ها یا **DA (Data Administrator)**: کل سیستم موجود را به خوبی می‌شناسند . در چگونگی مدیریت داده‌ها و اطلاعات موجود تبحر دارند مانند معاون مالی و اداری که سیستم را خوب می‌شناسد.

۲- کاربر نهایی یا **EU (End-User)**: کاربرانی که در نهایت می‌خواهند از سیستم تولید شده استفاده کنند. - ممکن است اطلاعاتی از اینکه سیستم اطلاعاتی که با آن کار می‌کنند مبتنی بر پایگاه داده‌ها هست یا نه، نداشته باشند. - نهایت انتظار این کاربر ، برآورده کردن عملیات مورد درخواست و درست عمل کردن و کارایی بالای سیستم و امکان دسترسی مدام در زمان کارش است.

۳- طراح پایگاه داده‌ها یا **DD (Database Designer)**: معماری یک سیستم پایگاه داده را طراحی می‌کند و منجر به این خواهد شد که داده‌های کاربردی به درستی استفاده شود به طوری که ارتباط معنای داده‌ها ، سازگاری و جامعیت آنها حفظ شود و سیستم پایگاه داده‌ها بتواند با کمترین افزونگی و در کوتاه‌ترین زمان ممکن و ایمنی بالا پاسخگوی نیازهای کاربران در همه زمان‌های مجاز باشد. - در واقع طراح منطقی پایگاه داده‌ها هستند.

۴- مدیر پایگاه داده‌ها یا **DBA (Database Administrator)**: کاربرانی که به پیاده‌سازی یا همان طراحی فیزیکی پایگاه داده‌ها می‌پردازند . مدیر پایگاه داده‌ها شخصی است که در به کارگیری یک یا چند نرم افزار پایگاه داده‌ای با سابقه و با تجربه است.

افزونگی (redundancy): ذخیره سازی داده های یکسان در چندین محل مختلف

► انواع افزونگی :

- ۱- طبیعی (ناشی از طبیعت داده ها) : هم سال بودن دانشجویان ورودی
- ۲- روش طراحی (طراحی نامناسب): ناشی از بکارگیری سیستم فایلینگ
- ۳- تکنیکی : الف (ایندکس گذاری
ب) درهم سازی

مزایای استفاده از سیستمای مبتنی بر پایگاه داده:

- ▶ دیده شدن همه داده ها به عنوان یک واحد اطلاعاتی به لحاظ منطقی
- ▶ حداقل امکان وجود ناسازگاری
- ▶ حفظ جامعیت داده ها : به معنی قابل قبول بودن اطلاعات مانند اینکه حقوق کارمندان نمی تواند از یک میزان حداقل دستمزد باشد. (حقوق کمتر از یک میزانی ، قابل قبول نیست)
- ▶ ...

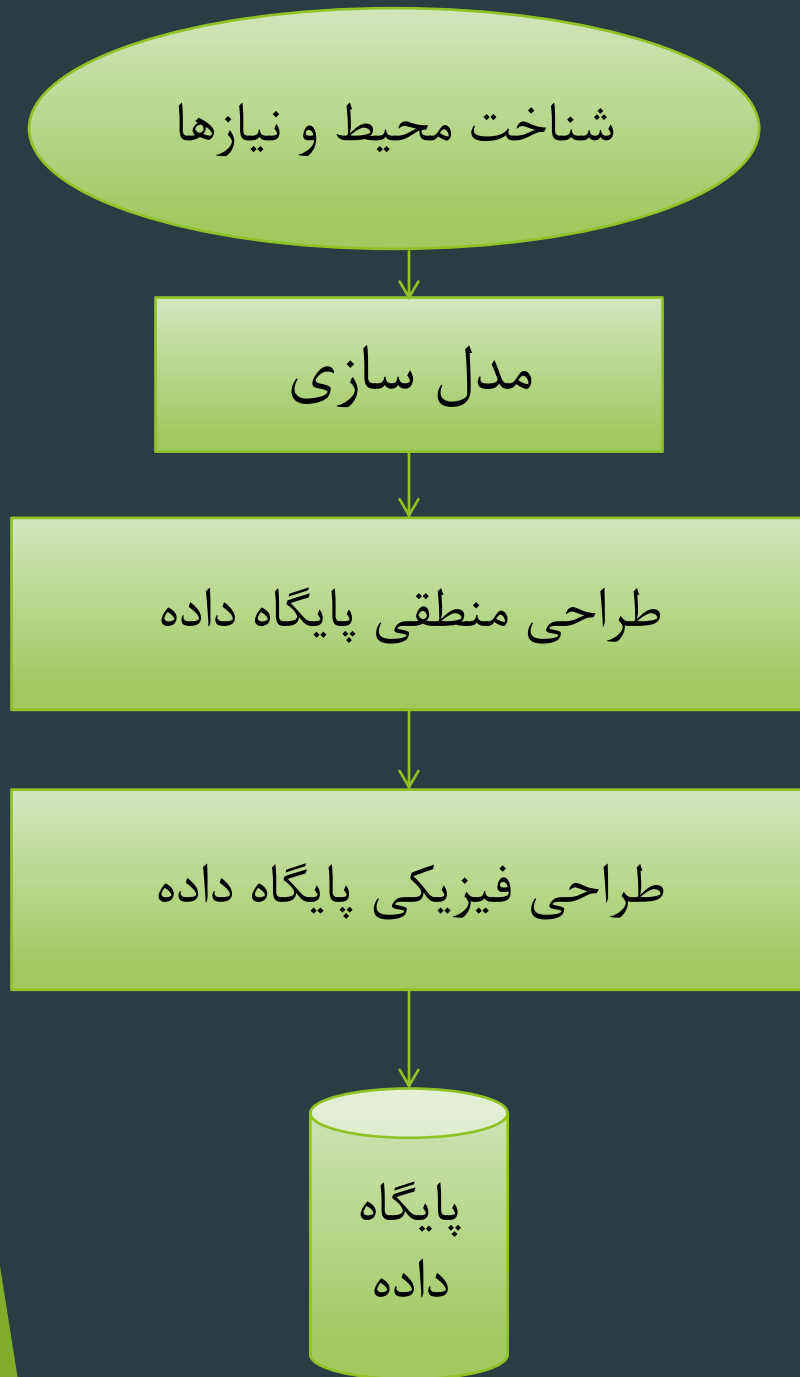
معایب سیستم های پایگاهی (سیستم بانکی)

- ▶ تحمیل هزینه های لازم برای خرید نرم افزار سیستم مدیریت پایگاه داده ها و مجوزهای مربوط به آنها و تقبل هزینه های لازم برای توسعه و بروز نگه داشتن نرم افزار
- ▶ پیدا کردن نیروهای متخصص برای نصب ، راه اندازی و سرویس نگهداری سیستم مدیریت پایگاه داده های خریداری شده و تقبل هزینه های پرداختی به آنها.

فصل سوم : مدل سازی

- ▶ شناخت نیازها
- ▶ مستندسازی
- ▶ مدل سازی
- ✓ مدل سازی فرآیندی
- ✓ مدل سازی داده ای
- ✓ مدل سازی ترکیبی

مراحل کلی ایجاد سیستم پایگاه داده



شناخت نیازها:

- ▶ نیازهای داده ای یا اطلاعاتی : نبود و نادیده گرفتن این اطلاعات ، آسیب جدی به سیستم وارد می کند . مانند کد کالا، تعداد موجودی در انبار، تاریخ موجودی کالا ، نام کالا و ...
- ▶ نیازهای عملکردی یا فرآیندی: در اثر پردازش بدست می آیند. مانند زمانی که تعداد کالایی جدید به موجودی در انبار اضافه می شود و باید طی عملیات پردازشی ، این تعداد کالا پس از وارد شدن کالا به انبار ، اضافه شود.

مستندسازی

- ▶ به منظور توسعه محصول نرم افزار و بهبود آن نیاز است
- ▶ هنگامی که مهندسی نیازها صورت میگیرد ، مجموعه ای از مستندات به صورت دفترچه و به همراه نرم افزار به کارفرما داده می شود.

مدل فرآیندی (process)

- دیاگرام جریان داده (DFD(data flow diagram))
- هدف : نمایش تصویری جریان داده در سیستم اطلاعاتی
- ابزاری برای نمایش گرافیکی فرآیندها و فعالیت های موجود و چگونگی حرکت داده ها

مدل داده‌ای (dat)

- دارای سه نوع معنایی ، منطقی و فیزیکی است.
- هدف: توصیف دقیق داده ها ، ارتباط میان آنها و محدودیت مقداری داده‌ها

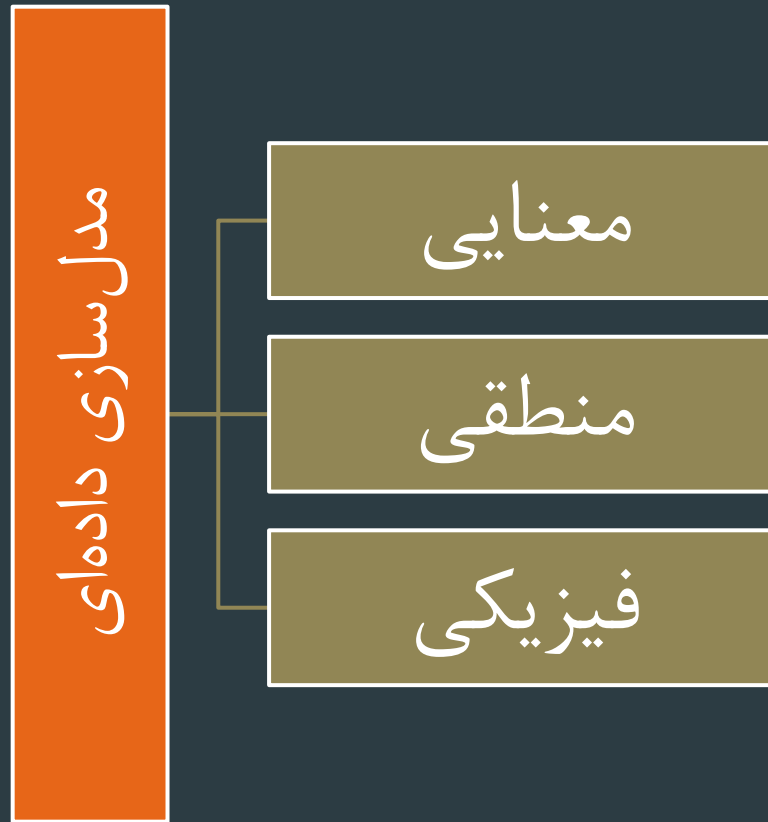
مدل ترکیبی

- کدام یک از عملکردها یا توابع روی کدام یک از موجودیت ها تاثیرگذار هستند و نوع تاثیر آنها چیست

مدل سازی فرآیندی

نام علامت	شکل علامت
فرآیند	<div><div>شماره فرآیند</div><div>نام فرآیند</div></div>
جریان داده	نام جریان داده →
محل ذخیره سازی داده	<div><div>شماره</div><div>نام محل ذخیره سازی</div></div>
موجودیت خارجی	نام موجودیت خارجی

مدل سازی داده ای



مدلسازی داده‌ای معنایی:

▶ ارائه یک مدل داده ای برای داده ها و ارتباط معنایی میان آنها در یک محیط عملیاتی به کمک مفاهیمی مستقل از جنبه های منطقی و نمایشی فیزیکی داده ها

▶ ارائه یک مدل داده ای در بالاترین سطح انتزاع برای بخشی از جهان هستی با هدف ایجاد یک پایگاه داده ها برای آن بخش.

مفاهیم کاربردی در مدل سازی داده ای معنایی:

► **موجودیت (entity):** هر چیز، هر شی در محیط عملیاتی که نیاز به اطلاعات آن داشته باشیم به طوری که آن شی قابل تمایز با چیزهای دیگر باشد.

نکته : اگر آن دو چیز قابل تفکیک از همدیگر نباشند بنابراین یک موجودیت تلقی می شوند.

۱. موجودیت قوی

۲. موجودیت ضعیف

پنج مشخصه موجودیت:

۱. منحصر به فرد: مانند اطلاعات هویتی دانشجو
 ۲. معنی دار: نامفهوم و گنگ نمی تواند باشد.
 ۳. دارای یک یا چند نمونه موجودیت : مانند اینکه یک دانشجو دارای اسم نادر ، نام خانوادگی حسینی ، رشته تحصیلی مهندسی کامپیوتر و ... است که هر کدام یک موجودیت است. **چه نوع موجودیتی؟**
 ۴. دارای یک یا چندین صفت: مانند صفت های موجودیت اطلاعات هویتی دانشجو عبارتند از نام ، نام خانوادگی ، رشته تحصیلی و مقطع تحصیلی و ...
 ۵. دارای شناسه: شناسه در واقع صفت منحصر به فرد در خصوص هر موجودیت است. مانند شماره دانشجویی در موجودیت اطلاعات هویتی دانشجو
- نکته:** برای موجودیت قوی داشتن شناسه ضروری است اما موجودیت ضعیف فاقد شناسه است

نکته : صفت می‌تواند ساده (تجزیه ناپذیر) و یا مرکب باشد . البته این مفهوم، یک مفهوم نسبی است. بدین منظور که نسبت به نوع کاربرد می‌توان در مورد مرکب بودن یا نبودن آن تصمیم‌گیری کرد. مثلاً آدرس محل سکونت دانشجو می‌تواند یک صفت ساده یا یک صفت مرکب در نظر گرفته شود. بدین ترتیب که با فرض اینکه در یک کاربرد هیچگونه عملیاتی اعم از مرتب‌سازی، جستجو و غیره روی آن صورت نگیرد، آدرس سکونت دانشجو یک صفت ساده است. اما اگر دانشجویان بر اساس شهر، منطقه و خیابان اصلی محل سکونت دسته‌بندی شوند، در چنین شرایطی صفت آدرس، صفت مرکب خواهد بود و قابل تجزیه.

نکته: هر صفت تنها می‌تواند دارای مقدارهای مخصوص به خود باشد لذا صفت‌ها نمی‌تواند پذیرنده هر گونه مقداری باشند. مانند اینکه در سیستم مدیریت پایگاه داده‌های میکروسافت اکسس همانند سایر سیستم مدیریت پایگاه داده‌ها زمانی که صفتی را تعریف می‌کنیم باید نوع آن و اندازه آن را نیز مشخص سازیم که در این صورت مقادیر اختصاص به صفت‌ها بر اساس نوع داده‌ای که در زمان طراحی تعریف شده است، می‌باشد. برای نمونه در نرم افزار مدیریت پایگاه داده میکروسافت اکسس امکان تعریف نوع داده‌ای متعدد از جمله اعداد، داده های متنی، تاریخ ، ساعت و غیره را می دهد.

نکته: هر صفت می تواند **تک مقداری** یا **چند مقداری** باشد. به طور مثال یک کاربرد صرفاً آخرین مدرک تحصیلی برای یک کارمند مدنظر باشد که در این صورت صفت مدرک تحصیلی یک صفت تک مقداری است اکا در یک کاربرد دیگر ممکن است کلیه مدارک تحصیلی یک کارمند مدنظر باشد که در این صورت صفت ، مدرک تحصیلی یک صفت چند مقداری است یعنی می تواند حاوی همه اطلاعات مربوط به همه مقاطع تحصیلی کارمند باشد شامل مدرک کاردانی، کارشناسی ، کارشناسی ارشد و دکتری باشد.

نکته: هر صفت می تواند به **صورت هیچ مقدار (NULL)** و **لزوما مقدار پذیر** باشد. به طور مثال شماره دانشجویی نمی تواند یک صفت هیچ مقدار باشد اما شماره پلاک خودرو می تواند هیچ مقدار باشد. یعنی دانشجو ممکن است ماشین نداشته باشد.

نکته: **هیچ مقدار** در پایگاه داده مطلوب نیست. هنگام مقایسه سیستم مدیریت پایگاه داده دچار مشکل ی شود زیرا امکان مقایسه یک مقدار یا یک مقدار null وجود ندارد. **راه حل** : استفاده از مقدار پیش فرض (Default value)

نکته: هر صفت می تواند **حقیقی** و **یا مشتق شده** باشد. صفت حقیقی صفتی است که قائم به ذات است یعنی به خودی خود وجود دارد و وابسته به هیچ صفت دیگری نیست. برای مثال صفت های نام دانشجو و یا نام خانوادگی دانشجو صفت حقیقی است اما صفتی مانند « حقوق خالص پرداختی » که حاصل انجام یک فرآینده محاسباتی روی حقوق ناخالص ، کسر مالیات، کسر بیمه و کسر وام های دریافتی است ، یک صفت مشتق شده است.

نکته: صفتی مرکب که یا همه یا برخی از اجزای آن چند مقداری باشد ، **صفت پیچیده** است

ارتباط (relationship): ارتباط میان دو یا چند موجودیت

► تفاوت ارتباط با رابطه؟؟ (فصل آخر)

روش های ترسیمی مدل سازی معنایی:

۱. روش دیاگرام موجودیت ارتباط ERD
(entity relationship Diagram)

۲. روش نمودار کلاسی UML
unified modeling language

نمایش اشکال هندسی در دو روش مطرح شده

UML	ERD	
داخل یک مستطیل نوشته می شود	داخل یک مستطیل نوشته می شود	موجودیت
داخل یک مستطیل که صفت مربوطه در پایین مستطیل مربوط به موجودیت نوشته می شود	داخل یک شکل دایره یا بیضی	صفت ها
زیرخط دار	زیرخط دار	صفت شناسه اول (اصلی)
زیرخط دار دابل	زیرخط دار دابل	صفت شناسه دوم
خط صاف که بالای آن نام ارتباط نوشته شده	شکل لوزی	ارتباط میان موجودیت ها

نکاتی در مورد ارتباط:



۱:۱ -

N:۱ -

۱:M -

M:N -



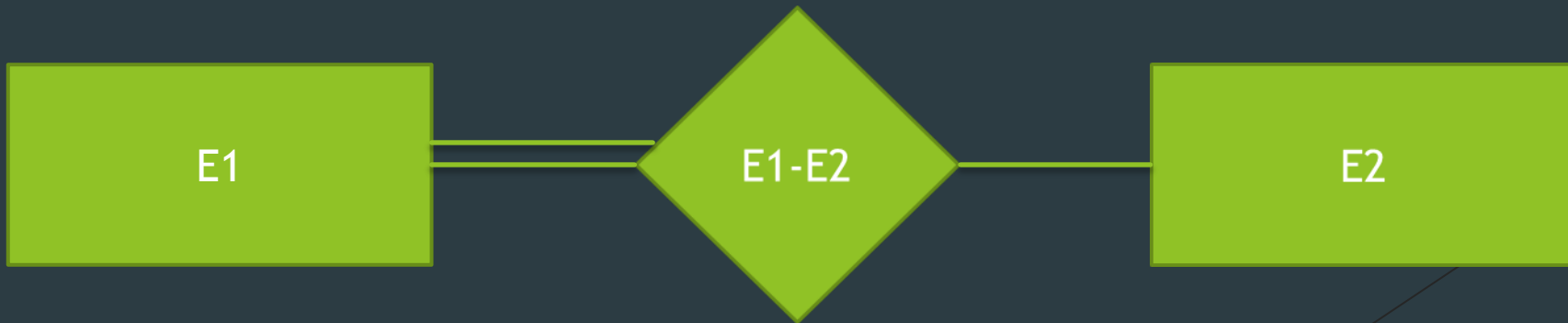
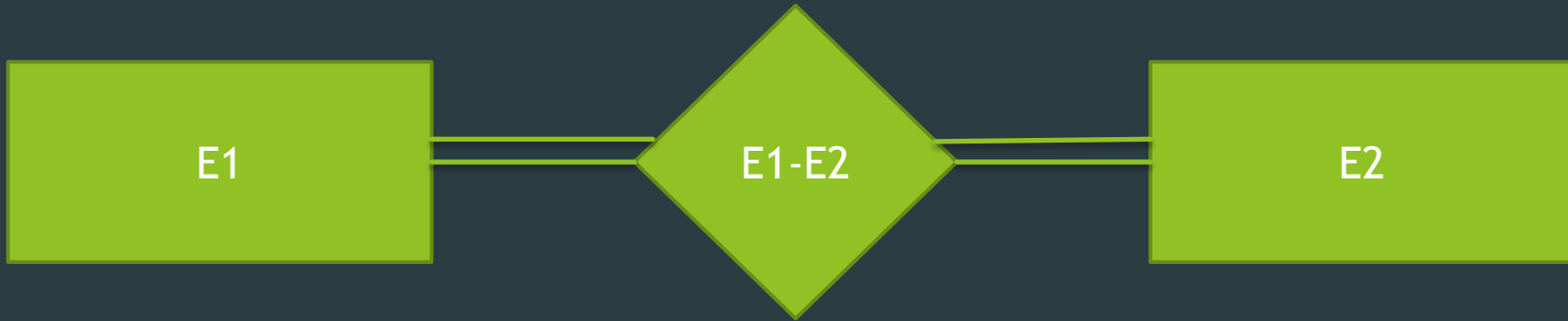
مفهوم مشارکت در ارتباط :

مشارکت الزامی

- حالتی است که در آن همه نمونه‌های موجودیت اول و همه نمونه‌های موجودیت دوم در ارتباط بین این دو موجودیت شرکت کرده باشند.

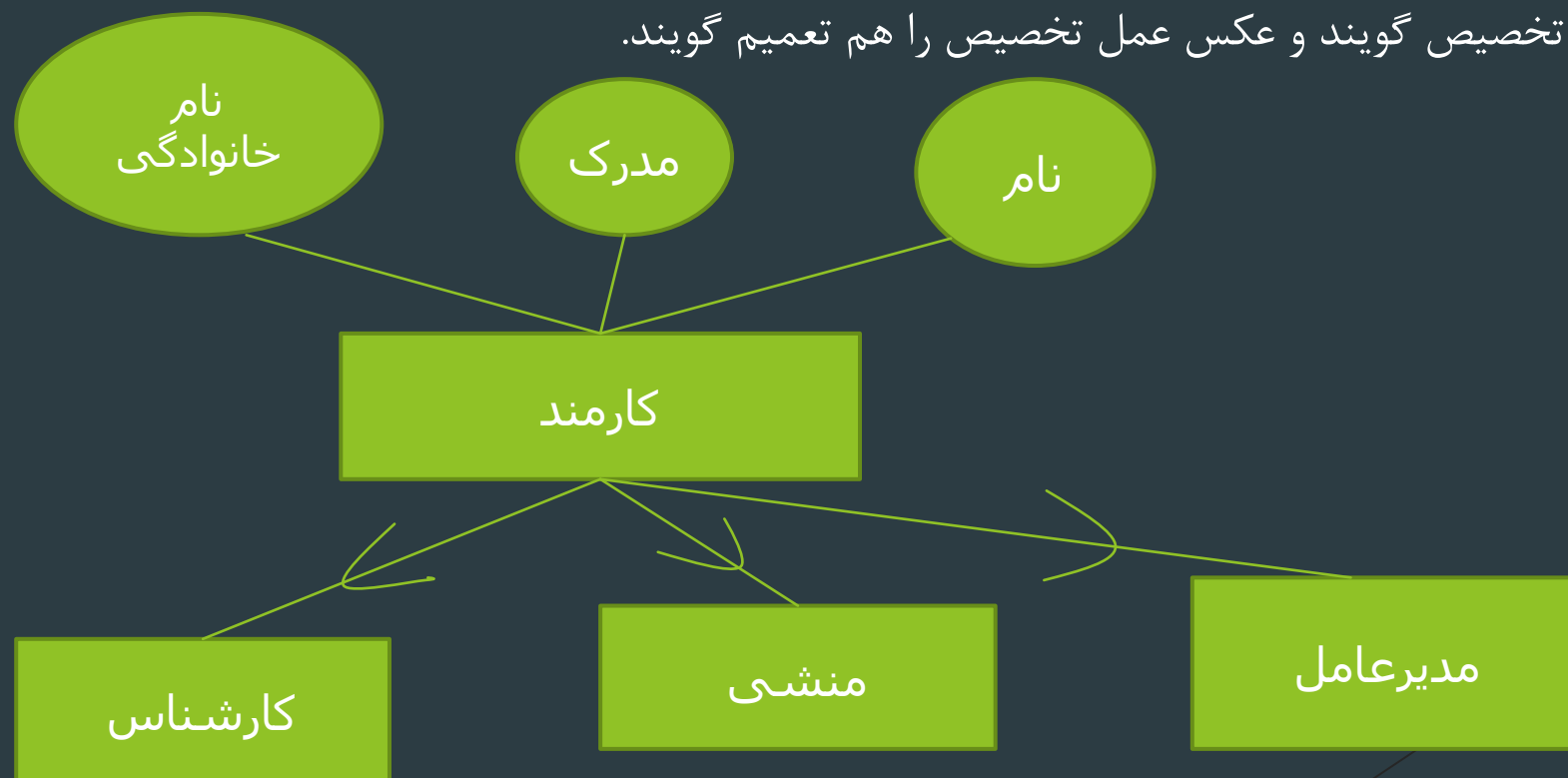
مشارکت غیرالزامی

- حالتی است که در آن ارتباط میان دو موجودیت، حداقل یک نمونه از موجودیت پیدا شد که در ارتباط شرکت نکرده باشند.



مدل سازی معنایی توسعه یافته (EER): تمامی نمادهایی را شامل می شود که در مدل سازی با روش ERD به کار می رود. (با تفسیر دقیق تر)
enhanced entity diagram

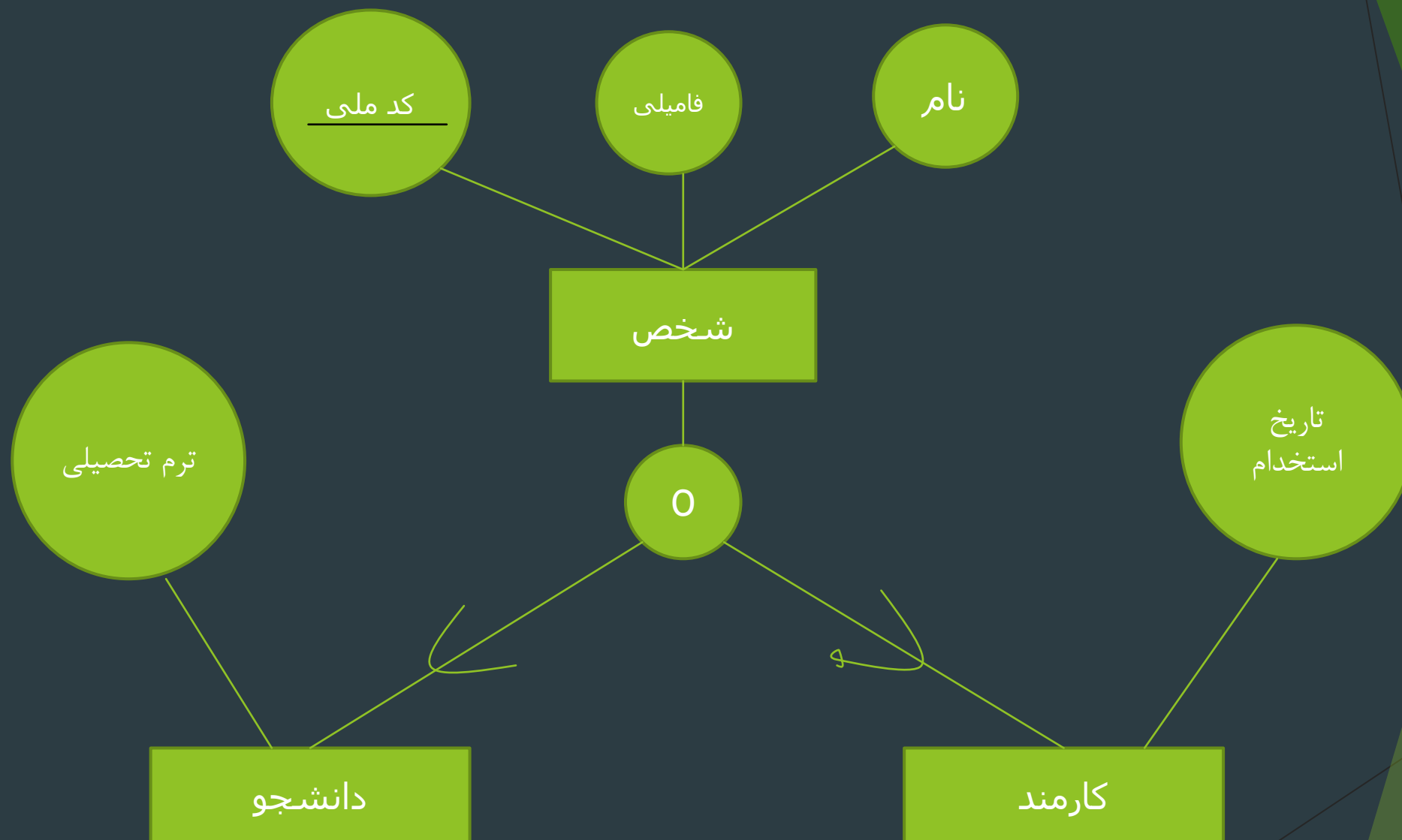
▶ ارتباط از نوع تخصیص و تعمیم یا ارتباط IS-A: اطلاعاتی که دارای عمومیت هستند و بیانگر یک موجودیت عمومی تر می شوند را موجودیت عام می گویند. به موجودیت های تخصیص داده شده نیز عمل تخصیص گویند و عکس عمل تخصیص را هم تعمیم گویند.



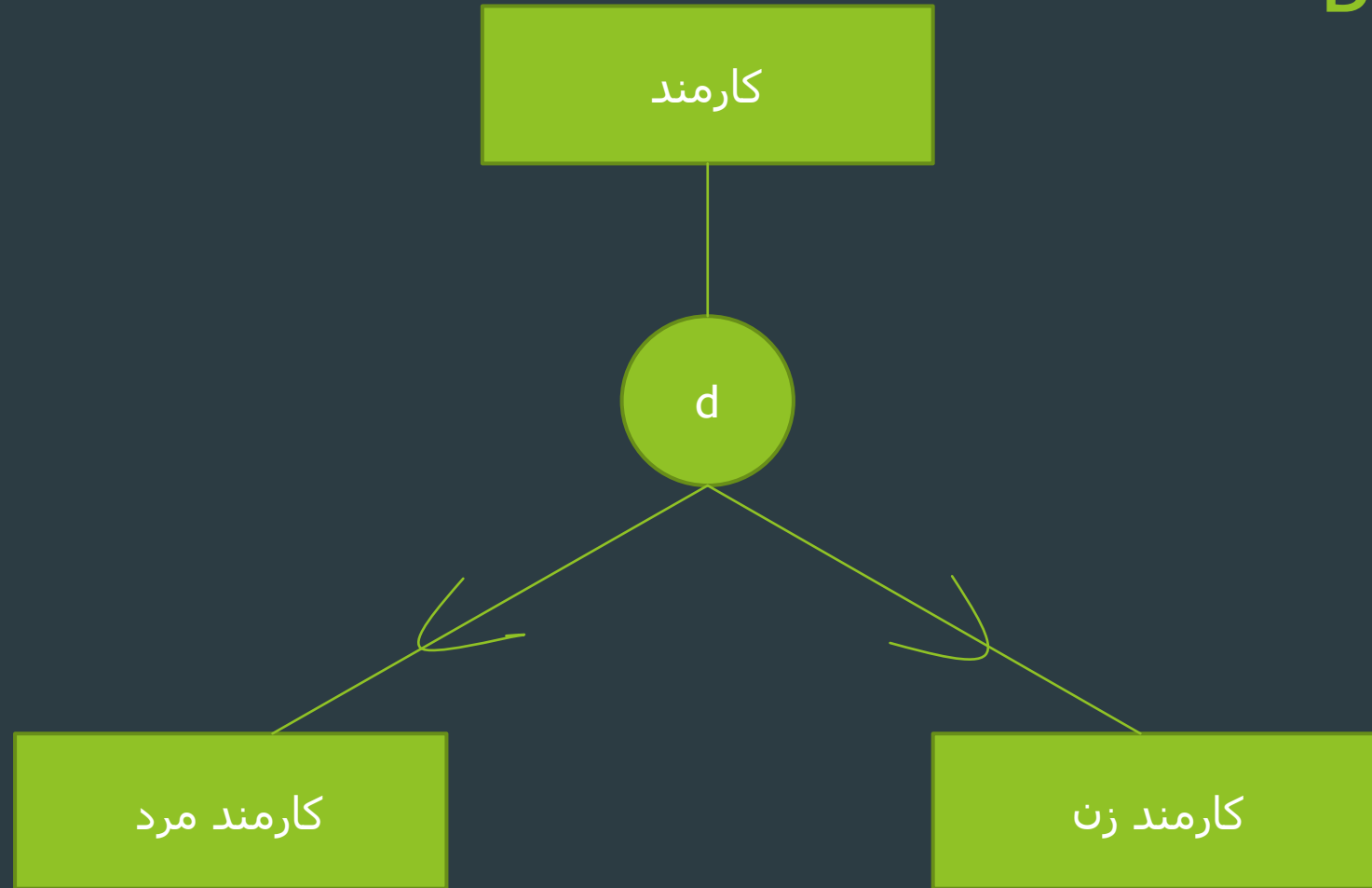
▶ مثال: اگر ابتدا موجودیت کارمند مشاهده شده باشد، سپس زیرمجموعه های مدیر عامل و کارشناس و منشی یافت شده باشد، عملیات تخصیص انجام شده و چنانچه ابتدا موجودیت های مدیرعامل، کارشناس و منشی یافت شده باشد سپس به موجودیت کلی تری به نام کارمند پی برده باشیم، عملیات تعمیم صورت گرفته است. (به توجه به اسلاید قبل)

نکاتی در مورد ارتباط :

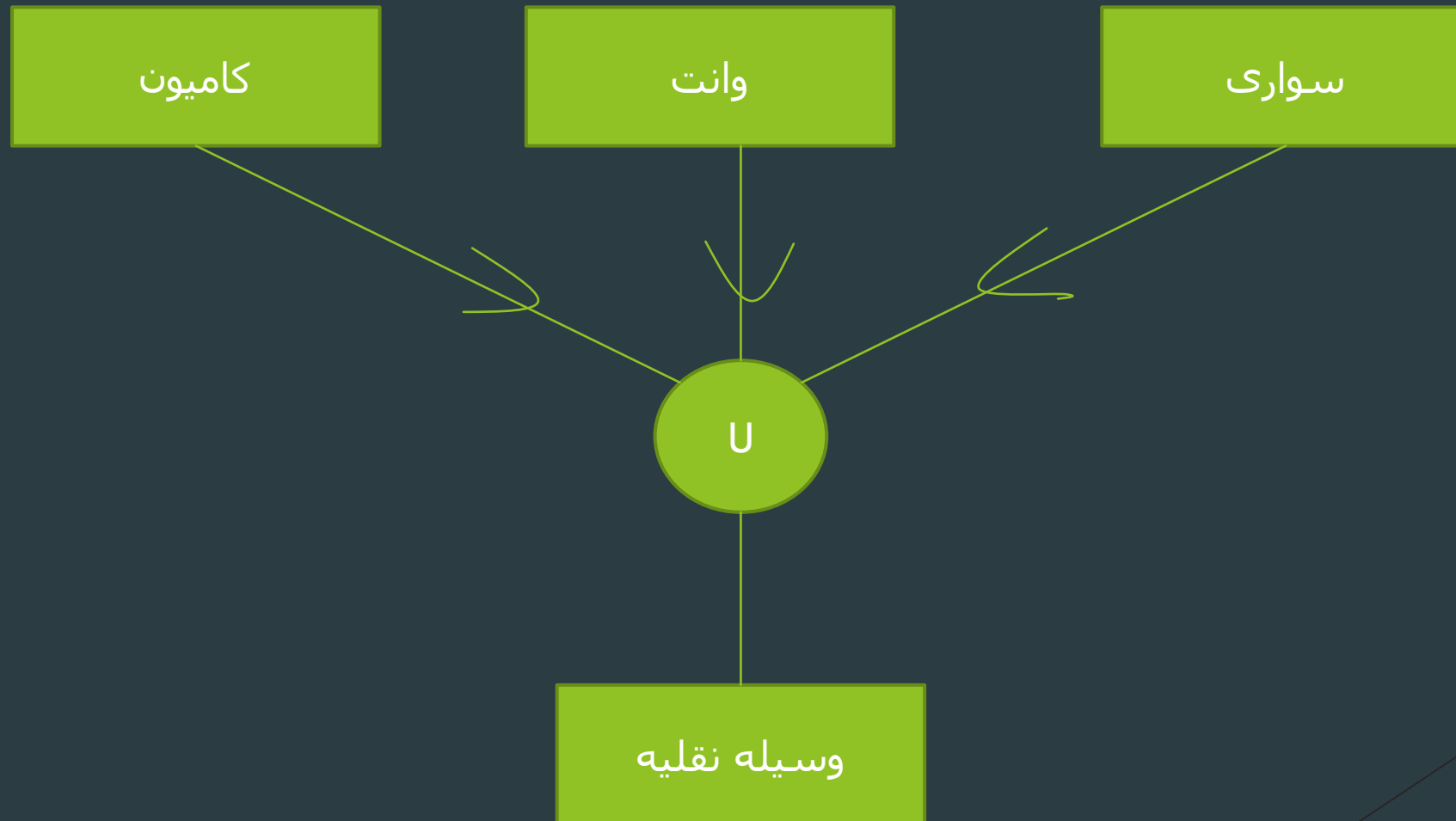
- ▶ در یک عملیات تخصیص و یا تعمیم چنانچه نمونه ای پیدا شود که عضو بیش از یک زیرمجموعه (زیرنوع) باشد به این حالت اصطلاحاً هم پوشانی (O) گویند.(شکل A)
- ▶ در یک عملیات تخصیص و یا تعمیم ، اگر هیچ نمونه ای یافت نشود که بتواند عضو بیش از یک زیرنوع باشد، به این حالت تمایز گویند که با d نشان داده می شود.(شکل B)
- ▶ ممکن است نمونه های موجودت زیرنوع ، عضو بیش از یک زیرنوع است. به بیان دیگر برخی از این نمونه ها عضو یک زیرنوع و برخی دیگر عضو یک زیرنوع دیگر باشند، که به آن زیرنوع اجتماع یا دسته می گویند. که با U نشان داده می شود.(شکل C)



شکل B



شکل c



ارتباط از نوع تجزیه و ترکیب یا ارتباط Has:

► می توان یک موجودیت را به چندین موجودیت دیگر (موجودیت جز) شکست. در واقع آن موجودیت ها ، اجزای موجودیت اولیه (موجودیت کل) هستند. به این حالت اصطلاحاً **تجزیه** می گویند و عکس عمل تجزیه را نیز **ترکیب** می گویند.

مدل سازی ترکیبی:

- ▶ تاثیر مولفه های عملکردی (توابع) و چگونگی تاثیر آنها روی مولفه های اطلاعاتی (موجودیت ها) و همچنین ارتباط میان مولفه های عملکردی یا مولفه های اطلاعاتی باشیم. در واقع مدل های ترکیبی برای بیان تاثیر مولفه های عملکردی و نوع آن روی مولفه های اطلاعاتی (موجودیت ها) به کار گرفته می شوند.
- ▶ مدل FET (function entity table) به دنبال آن هستیم که بدانیم کدام یک از عملکردها یا توابع روی کدام یک از موجودیت ها تاثیرگذار هستند و نوع تاثیر آنها چیست.

مثال FET :

نوع تاثیر هر یک از عملکرد روی موجودیت‌های مختلف با هم تفاوت دارد. برای مثال تاثیر عملکرد تابع $F!$ روی موجودیت های $E1$ با $E2$ و با $E3$ تفاوت دارد.

➤ نوع تاثیر :

➤ R (retrieve) بازیافت

➤ i (insert)

➤ U (update)

➤ D (delete)

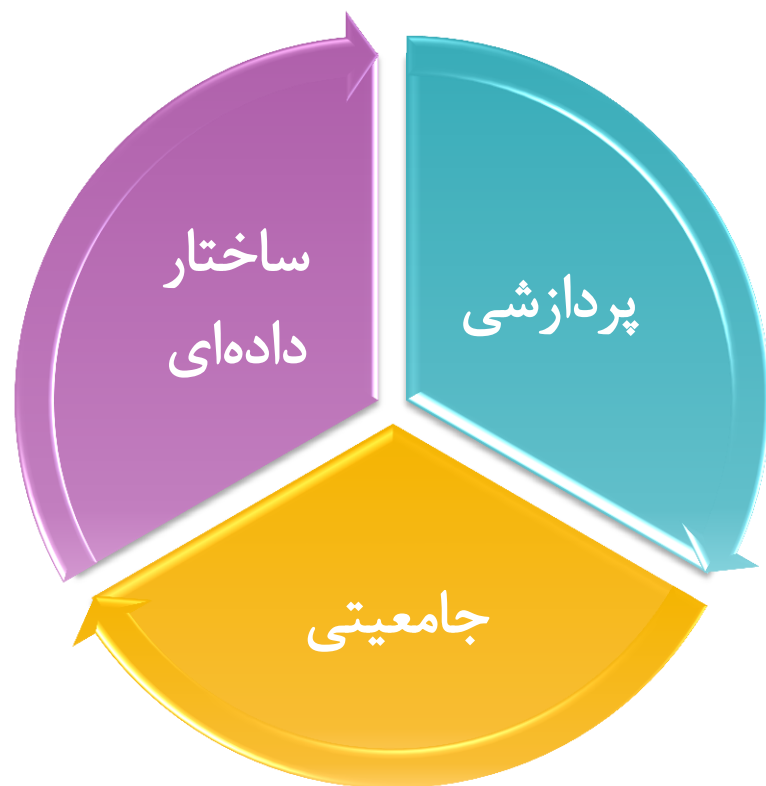
F5	F4	F3	F2	F1	F/E
	Riu			Riud	E1
	R	lu	Riud	I	E2
i				Riu	E3

فصل چهارم:
طراحی منطقی
پایگاه داده‌ها

- مدل داده‌ای
- اجزای تشکیل دهنده یک مدل داده‌ای
 - ✓ بخش ساختار داده‌ای
 - ✓ بخش عملیاتی یا پردازشی
 - ✓ بخش جامعیتی
- حالت‌های کلاسیک از مدل های داده‌ای
 - ✓ مدل سلسله مراتبی
 - ✓ مدل داده ای شبکه‌ای
 - ✓ مدل داده‌ای رابطه‌ای (جدولی)

مدل داده‌ای: روشی نمادین برای توصیف داده ها و اطلاعات


- اجزای تشکیل دهنده یک مدل داده‌ای



۱- بخش ساختار داده‌ای: منظور ساختمان داده است. مربوط به عناصر تشکیل‌دهنده پایگاه داده است. مانند چگونگی توصیف موجودیت‌ها (استفاده از یک ساختار در زبان برنامه نویسی C++)

۲- بخش عملیاتی یا پردازشی: از نظر تئوری و با استفاده از تعدادی عملگرها، بتوان به داده‌ها دسترسی داشت (عملیاتی مانند بازیافت، تغییر و حذف و ... انجام داد). - **توان با داده‌ها کار کرد.**

۳- بخش جامعیتی: شامل مجموعه‌ای از قواعد و محدودیت‌ها است تا به کمک این قواعد، بتوان **قابل قبول بودن، دقت و سازگاری** را در سیستم پایگاه داده‌ها تا جای ممکن تامین کرد.



صفت شماره دانشجویی
کاراکتری یا عددی!

فیش حقوقی
(اعشاری یا صحیح!)

شماره تلفن
(کاراکتری یا عددی!)

انواع دسته‌بندی مدل‌های
داده‌ای کلاسیک در پایگاه
داده

- ۱- سلسله مراتبی
- ۲- شبکه‌ای
- ۳- جدولی (رابطه‌ای)

مدل سلسله مراتبی:

قدیمی ترین مدل - مبتنی بر دو نوع ساختار

۱- نوع رکورد ۲- پیوند پدر فرزندی

نوع رکورد: برای نمایش **موجودیت‌ها** استفاده می‌شود.

پیوند پدر-فرزندی: برای بیان **ارتباط میان موجودیت‌ها** مورد استفاده قرار می‌گیرد که به صورت ساختار رختی قابل پیاده سازی است.

مزایای مدل سلسله مراتبی:

- ۱- بازیافت سریع اطلاعات ذخیره شده
- ۲- ساده بودن مدیریت جامعیت اطلاعات

رکورد نوع ریشه

رکورد نوع الف

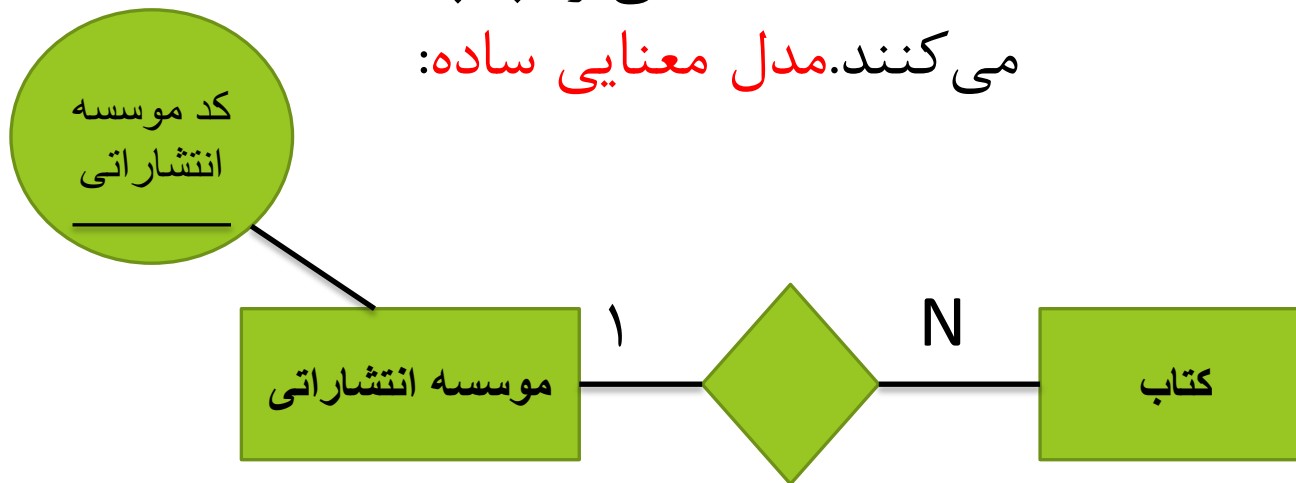
رکورد نوع ب

رکورد نوع ج

رکورد نوع...

رکورد نوع ...

مثال : موسسه انتشاراتی که
کتاب‌های مختلفی را چاپ
می‌کنند. **مدل معنایی ساده:**



نمایش به صورت **مدل سلسله مراتبی:**

رکورد نوع ریشه



رکورد نوع فرزند



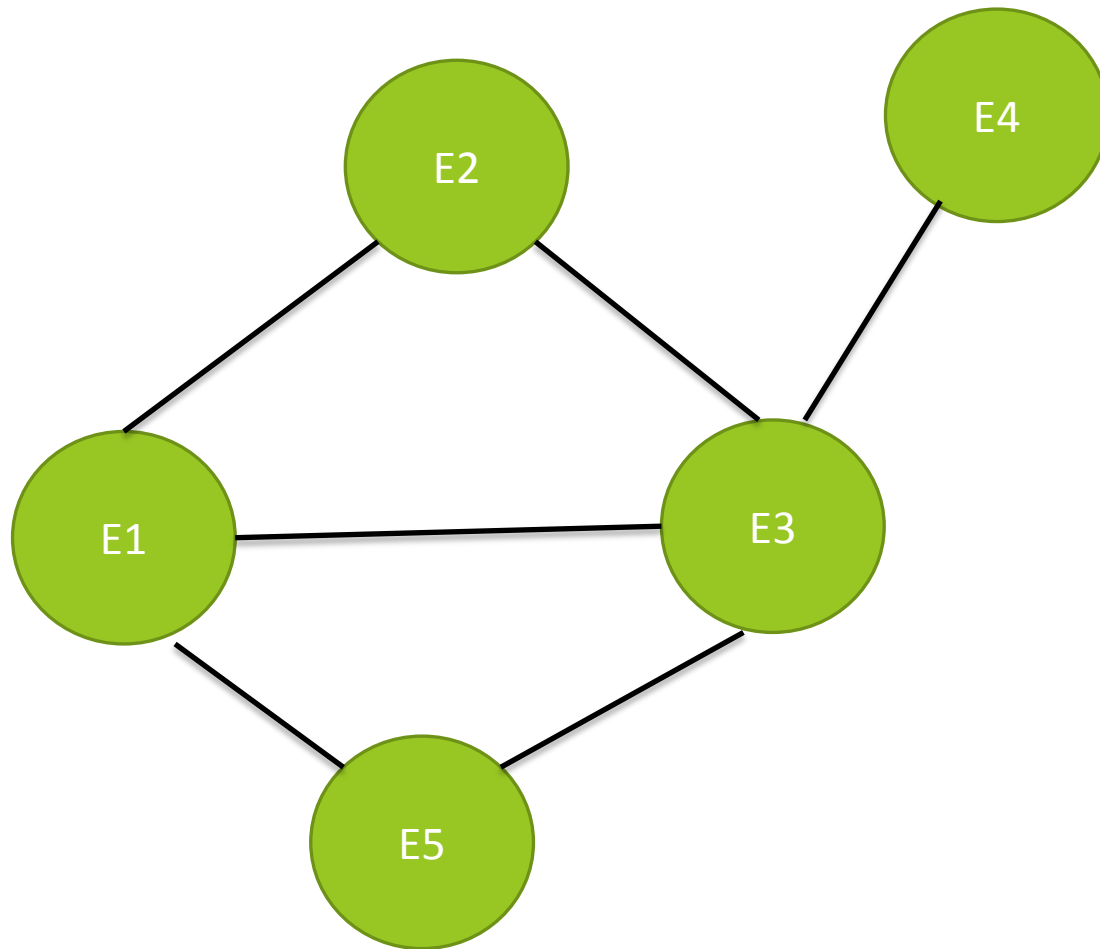
سوال : اگر به جای یک موسسه انتشاراتی با
چندین کتاب، چندین موسسه انتشاراتی با چندین
کتاب داشته باشیم، آیا می توان به راحتی مدل
سلسله مراتبی را ترسیم کرد؟

ایراد مدل سلسله مراتبی:

- نمایش ارتباطات چند به چند
دشوار است

($M:N$ چون درخت یک ریشه دارد)

- وجود دو نوع ساختار اساسی.



مدل داده‌ای شبکه‌ای
NDM: Network Data)
(Model

مبتنی بر تئوری گراف –
موجودیت‌ها به صورت گره و
ارتباط بین موجودیت‌ها به
صورت یال نمایش داده
می‌شوند- مانند مدل سلسله
مراتبی مبتنی بر دو نوع
ساختار (نوع رکورد و نوع
مجموعه)

مزایای مدل داده‌ای شبکه‌ای:

ایرادهای مدل داده‌ای شبکه‌ای:

۱- امکان داشتن بیش از یک والد

برای نوع رکورد فرزند

۲- دسترسی به داده‌ها را می‌توان از هر گره‌ای آغاز کرد.

۳- برای رسیدن به داده مورد نظر

می‌توان در هر جهتی (به سمت بالا، یا سمت پایین) در گراف حرکت

نمود.

۴- ایجاد پرسشهای پیچیده نسبتاً

ساده تر است.

۱- ساده نبودن اعمال تغییرات در ساختار پایگاه داده‌ها

۲- بروز هر گونه تغییرات در ساختار پایگاه داده‌ها، منجر به تاثیرگذاری روی برنامه‌های کاربردی می‌شود که به پایگاه داده‌ها دسترسی دارند.

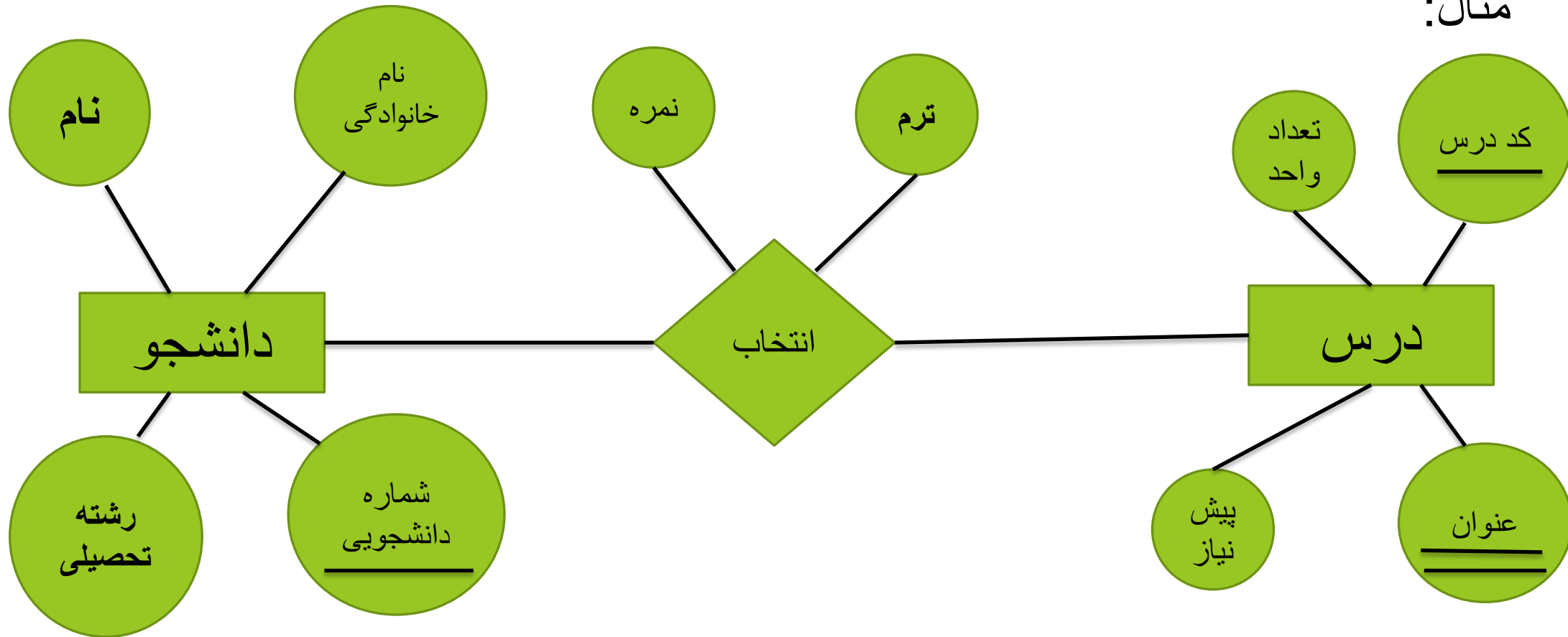
۳- نداشتن وضوح بالای کاربری

۴- وجود همچنان دو ساختار اساسی، شامل نوع رکورد و نوع مجموعه

مدل داده‌ای رابطه‌ای (جدولی):

مبتنی بر ساختار جدول است. - هر ستون در یک جدول بیانگر یک صفت و هر سطر از جدول بیانگر یک نمونه از موجودیت است.

مثال:



کد درس	عنوان	تعداد واحد	پیش نیاز

مدل رابطه‌ای (جدولی) موجودیت درس:

تمرین :

مدل رابطه‌ای (ساختار جدولی) موجودیت دانشجو و یک مدل
رابطه‌ای (ساختار جدولی) ارتباط دانشجو- درس مثال قبل را
ترسیم کنید.

مزایای مدل رابطه‌ای (جدولی):

۱- از نظر کاربر شکل نمایشی بسیار ساده‌ای دارد.

۲- از جبر رابطه‌ای و مبانی ریاضیات استفاده می‌کند.

۳- تنها یک نوع ساختار (جدولی) دارد.

۴- انواع ارتباط‌ها بین موجودیت‌ها قابل پیاده‌سازی است

۵- منطق بازیابی داده‌های ذخیره شده در این مدل ساده است.

نکته: از طریق شناسه، داده‌ها را از جدول می‌توان بازخوانی کرد

معایب مدل رابطه‌ای (جدولی):

داشتن حجم عظیمی از اطلاعات،
چالش این مدل است. (انواع محدودیت
ها در رابطه‌ها، کاربر آن را در دنیای
وب امروز و موتورهای جستجو و
شبکه‌های اجتماعی سخت می‌کند.

فصل پنجم: معماری پایگاه داده‌ها

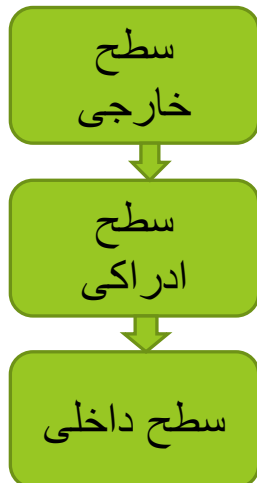
- معماری سه لایه برای پایگاه داده‌ها
- نگاشت
- استقلال داده‌ها
- ✓ استقلال فیزیکی
- ✓ استقلال منطقی
- زبان میزبان (HL)
- زبان فرعی داده‌ها (DSL)
- ✓ زبانی تعریف داده‌ها (DDL)
- ✓ زبانی برای کار با داده‌ها (DML)
- ✓ زبانی برای کنترل داده‌ها (DCL)

اجزای معماری سه لایه برای پایگاه داده‌ها:

۱- سطح خارجی : مجموعه‌ای از نگاه‌ها یا دیدهای کاربران به سیستم پایگاه داده‌ها- فراهم آورنده یک محیط انتزاعی برای کار کاربران به گونه ای که به بخشی از پایگاه داده دسترسی داشته باشند نه همه پایگاه داده
(دربرگیرنده همه دیدها اما نه یک دید کامل)
(یک کاربر می‌تواند یک دید یا چند دید نسبت به بخشی از یک پایگاه داده داشته باشد.
یک دید می‌تواند در اختیار چند کاربر قرار بگیرد.)

سوال: اطلاعات چگونه بین این لایه‌ها مبادله می‌شود؟

۲- سطح ادراکی یا شِمای ادراکی: سطح میانی- به توصیف داده‌های مورد نیاز کاربران و سازمان و ارتباط میان داده‌ها می‌پردازد.
۳- سطح داخلی: چگونگی دسترسی به اطلاعات ذخیره شده در پایگاه داده و ساختار فیزیکی پایگاه داده را شامل می‌شود- مانند مقدار فضای اختصاص داده شده به هر صفتی در حافظه (مثلاً شماره پرسنلی کارمند چه تعداد می‌تواند باشد)



نگاشت:

مکانیسم نگاشت ادراکی/داخلی کمک می کند تا سیستم مدیریت پایگاه داده بتواند به ساختارهای منطقی تعریف شده در سطح داخلی متناظر با هر یک از نمونه های موجود در سطح ادراکی دسترسی داشت.

مکانیزم نگاشت خارجی/ادراکی این امکان را فراهم می کند تا سیستم مدیریت پایگاه داده بتواند انواع دیده ها را از شمای ادراکی استخراج نماید.

استقلال داده‌ها: مصونیت لایه‌های بالاتر نسبت به تغییرات فنی اعمال شده در لایه‌های پایین‌تر.
به دو نوع تقسیم بندی می‌شود:

۱- استقلال فیزیکی: امکان تغییر در **سطح داخلی** بدون نیاز به تغییر در شمای ادراکی و خارجی. مانند تغییر در ساختار ذخیره‌سازی، تغییر ایندکس‌ها، تغییر الگوریتم‌ها و ...
در سطح داخلی رخ می‌دهد که این تغییرات نباید تاثیری در نگاه کاربران نهایی به پایگاه داده داشته باشد.

۲- استقلال منطقی: امکان تغییر در **سطح ادراکی** بدون نیاز به تغییر در شمای خارجی. گاهی به دلایل فنی و یا به دلایل نیاز سازمانی ممکن است تغییراتی مانند ایجاد موجودیت‌های جدید و ارتباط میان آنها و یا افزایش صفت‌های جدید به موجودیت‌های موجود در سطح ادراکی رخ دهد که این تغییرات نباید تاثیری در نگاه کاربران نهایی به پایگاه داده داشته باشد.

زبان میزبان **host language (HL)** : زبانی است که به وسیله آن می توان یک رابط کاربری را برای ارتباط کاربران با پایگاه داده ها فراهم آورد. مانند جاوا، C# و ...

مثال: شرکت مایکروسافت برای ارائه رابط های کاربری برای پایگاه داده Ms Access، ابزار ویژه ای به نام wizard ایجاد کرده است که می توان با کمک برای ساخت رابط کاربری به نام فرم مورد استفاده قرار گیرد.

یا شرکت اوراکل، برنامه oracle designer دارد که از آن به عنوان CASE استفاده می شود. CASE یک ابزار نرم افزاری است که برای تولید سریع پایگاه داده ها بر اساس مدل معنایی ارائه شده استفاده می شود.

زبان فرعی داده‌ها (**DSL**): برای کار با پایگاه داده‌ها، وجود یک زبان برنامه‌نویسی ضروری است. به همین علت زبانی تحت نام زبان فرعی داده‌ها توسعه یافت. زبان فرعی داده‌ها می‌تواند به صورت مستقل یا ادغام شده باشد. زبان فرعی داده‌ها وقتی مستقل محسوب می‌شود که در قالب یک محیط سطر فرمان (**CLI**) به عنوان یک رابط کاربری سطح پایین مطرح می‌شود، قابل استفاده باشد، به عبارت دیگر برای کار با آن به زبان دیگری نیاز نباشد. مانند MS SQL Query Analyzer

زبان برای تعریف داده‌ها (**DDL**): زبانی برای ایجاد و تغییر دادن کلی جدول‌ها که شامل دستوراتی از جمله create, drop, ...
زبان برای کار با داده‌ها (**DML**): زبانی برای تغییر دادن داده‌ها مثلاً اضافه نمودن و یا حذف کردن است که شامل دستوراتی از جمله insert, delete, ...
زبان برای کنترل داده‌ها (**DCL**): زبانی برای کنترل جریان داده و امنیت داده‌ها است و شامل دستوراتی از جمله rollback, commit, ...

فصل ششم : اساس مدل رابطه‌ای

- مدل رابطه‌ای
- ✓ تعریف رابطه
- ✓ سرایند (Header)
- ✓ بدنه (حذف)
- شمای رابطه
- انواع رابطه
- کلیدها و انواع آن

مدل رابطه‌ای: مبتنی بر رابطه ریاضی (توضیحات بیشتر در فصل چهارم آمده است)

میدان: از نظر ریاضی میدان مجموعه‌ای است از یک یا بیش از یک مقدار متمایز قابل اختصاص به یک صفت. هر میدان دارای خصوصیت‌های زیر است:

۱- هر میدان دارای نام است

۲- هر میدان دارای مقادیری از نوع داده‌ای مشخص است.

۳- هر میدان دارای مقادیری با فرمت مشخص است.

مثال: میدان صفت شهر محل تولد برای موجودیت دانشجو :

$D_{city} = \{\text{تهران، اصفهان، تبریز، کرمان، یزد، اهواز، ساری}\}$

سرآیند (Header): سرآیند مجموعه‌ای از n صفت متمایز به صورت $A_i:D_i$ به طوری که $i=1,2,3,...,n$ باشد و هر A_i نام یک صفت باشد و D_i میدان اختصاص داده شده به صفت A_i باشد. لذا بخش سرآیند مجموعه‌ای است که به صورت رابطه زیر تعریف می‌شود:

$$\text{header} = \{< A_1:D_1><A_2:D_2>...<A_n:D_n>\}$$

تناظر بین مفاهیم رابطه‌ای و مفاهیم جدولی

مفهوم رابطه‌ای	مفهوم جدولی
رابطه	جدول
تاپل	سطر (رکورد)
صفت	ستون (فیلد)
میدان	مقادیر قابل اختصاص به هر ستون
آریتی (Ariety)	تعداد ستون‌ها (تعداد فیلدها)
کاردینالیتی	تعداد سطر (تعداد رکوردها)

انواع رابطه:

- ۱- **رابطه نامدار:** هر رابطه در مفهوم جدولی معادل یک جدول است و هر جدول یک اسم دارد و رابطه نامدار، رابطه ای است که با یک اسم به سیستم معرفی می شود.
- ۲- **رابطه مبنا:** رابطه ای است که استقلال وجودی دارد و از رابطه دیگری مشتق نشده باشد.
- ۳- **رابطه دید:** دید نوعی رابطه نامدار است که از رابطه‌ای دیگر مشتق شده است و از نظر ماهیت یک رابطه مجازی است.
- ۴- **رابطه مشتق:** رابطه‌ای که به کمک یک عبارت رابطه‌ای برحسب یک یا چند رابطه نامدار دیگر تعریف می‌شود لذا رابطه مشتق فاقد استقلال وجودی است. **(حذف)**
- ۵- **رابطه لحظه‌ای:** یا دید ذخیره شده نوعی رابطه نامدار و مشتق است به طوری که تاپل‌های آن، اطلاعات لحظه‌ای پایگاه داده‌ها را نشان می‌دهند. **(حذف)**

انواع کلیدها:

ابرکلید (**super key=SK**): با در نظر گرفتن رابطه R, A به طوری که $r(A)$ زیر مجموعه ای از $r(R)$ باشد، هر کدام از A یک ابرکلید خواهد بود.

کلید کاندید (**candidate key=CK**): هر سوپرکلیدی که **تجزیه ناپذیر** باشد را کلید کاندید گویند. یا سوپرکلیدی که منحصر به فرد باشد و حداقل باشد و کاهش ناپذیر باشد را کلید کاندید گویند

کلید اصلی (**primary key=PK**): کلیدی اصلی یک انتخاب از بین تمام کلیدهای کاندید است و **معیار برای انتخاب هم نظر کارفرما و هم کم بودن فضای ذخیره سازی مورد نیاز است**

کلید بدیل (**Alternate key=AK**): عبارت است از اعضای مجموعه کلیدهای کاندید به جز عضوی که به عنوان کلید اصلی انتخاب شده باشد.

کلید خارجی (**Foreign Key= FK**)

فصل هفتم: جبر رابطه‌ای (حذف)

- جبر رابطه‌ای و عملگرهای آن
- حساب رابطه‌ای
- ✓ حساب رابطه‌ای تاپلی
- ✓ حساب رابطه‌ای میدانی

فصل هشتم: نرمال سازی رابطه‌ها

- آنومالی و افزونگی
 - ✓افزونگی و انواع آن
 - ✓آنومالی و انواع آن
 - ✓آنومالی‌ها در عمل درج و حذف
 - ✓آنومالی در عمل بروزرسانی
- نرمال سازی
 - ✓سطح اول نرمال
 - ✓سطح دوم نرمال
 - ✓سطح سوم نرمال
- نرمال سازی معکوس

افزونگی : در فصل‌های قبل کم و بیش در مورد آن گفته شده است که باعث افزایش بی رویه فضای مورد نیاز برای ذخیره سازی اطلاعات در سطح فایل‌ها می‌شد.

دلایل ایجاد افزونگی: طبیعت داده‌ها ، ملاحظات تکنیکی طراح پایگاه داده و ...

آنومالی:

مشکلی که به سبب آن، انجام برخی از اعمال روی پایگاه داده ناممکن یا پیچیده می‌شود و یا حتی اگر انجام یک عمل میسر باشد اما نتیجه اجرای آن عمل ممکن است منجر به بروز تغییرات نامطلوبی در اطلاعات ذخیره شده در پایگاه داده شود.

انواع آنومالی:

آنومالی در عمل درج

زمانی رخ می‌دهد که به دلایلی مانند نداشتن کلید نتوان داده جدیدی را وارد پایگاه داده‌ها نمود.

آنومالی در عمل حذف

زمانی رخ می‌دهد که با حذف نمودن اطلاعاتی، اطلاعات دیگری به طور ناخواسته حذف شود.

آنومالی در عمل بروزرسانی

زمانی رخ می‌دهد که برای بروزرسانی اطلاعاتی، لازم به انجام مکرر عمل بروزرسانی شویم که این منجر به انجام کار اضافه خواهد شد.

مثال: آنومالی در عمل حذف و در بروزرسانی را در پایگاه داده جدولی زیر ببینید:

شماره دانشجویی	نام	نام خانوادگی	کدرشته	رشته تحصیلی	دانشکده	
۹۳۱۰۰۱	علی	حسینی	۱۰۰	مهندسی کامپیوتر	فنی و مهندسی	بروز
۹۳۱۰۰۵	مهناز	مرادی	۱۰۰	مهندسی کامپیوتر	فنی و مهندسی	بروز
۹۳۳۰۰۴	شاهین	درودی	۳۰۰	پزشکی	علوم پزشکی	
۹۲۵۰۰۸	سیما	محمدی	۱۰۰	مهندسی کامپیوتر	فنی و مهندسی	بروز
۹۴۷۰۰۶	مرتضی	رستمی	۳۰۰	پزشکی	علوم پزشکی	
۹۳۹۰۰۳	همایون	شجریان	۴۰۰	موسیقی	هنر و ادبیات	حذف

نکته: اگر بخواهیم شماره ۹۳۹۰۰۳ را حذف کنیم در این صورت اطلاعات مربوط به رشته موسیقی ناخواسته حذف می‌شود.

نکته: اگر بخواهیم مهندسی کامپیوتر را با مهندسی کامپیوتر- نرم افزار بروز کنیم در این صورت سه بار باید این کار را انجام دهید که باعث اتلاف وقت می‌شود

راه حل برای کاهش آنومالی در بروزرسانی

شماره دانشجویی	نام	نام خانوادگی	کدرشته
۹۳۱۰۰۱	علی	حسینی	۱۰۰
۹۳۱۰۰۵	مهناز	مرادی	۱۰۰
۹۳۳۰۰۴	شاهین	درودی	۳۰۰
۹۲۵۰۰۸	سیما	محمدی	۱۰۰
۹۴۷۰۰۶	مرتضی	رستمی	۳۰۰
۹۳۹۰۰۳	همایون	شجریان	۴۰۰

کد رشته	رشته تحصیلی	دانشکده
۱۰۰	مهندسی کامپیوتر	فنی و مهندسی
۳۰۰	پزشکی	علوم پزشکی
۴۰۰	موسیقی	هنر و ادبیات

سطح اول نرمال:

رابطه‌ای که فاقد صفت مرکب و صفت‌های چند مقداری است. - رابطه‌ای که صفت‌های آن تجزیه‌ناپذیر است. - فاقد رابطه تو در تو. مثال برهان خلف:

در جدول زیر رابطه‌ها تو در تو است و صفت مرکب دارد، بنابراین سطح اول نرمال نیست

تهدیه کننده قطعه		صفت قطعه - تعداد	
فروشگاه عباسی	نام قطعه	تعداد قطعه	
	مولتی متر	۱۰۰	
	پتانسیومتر	۲۰۰	
	خازن	۳۰۰	
فروشگاه محمدی		مولتی متر	۱۲۰

نرمال سازی:

یکی از راه حل‌های کاهش دو مشکل افزونگی و آنومالی ، نرمال سازی است.

سطح اول (1NF)
سطح دوم (2NF)
سطح سوم (3NF)

سطح دوم نرمال:

یک رابطه در سطح دوم نرمال است اگر و فقط اگر:
اولاً: در سطح اول نرمال باشد،
ثانیاً: هر صفت غیرکلید و غیرجزء کلید با کلید اصلی وابستگی تابعی **کامل** داشته باشد.

سطح سوم نرمال:

یک رابطه در سطح سوم نرمال است اگر و فقط اگر:
اولاً: در سطح دوم نرمال باشد،
ثانیاً: هر صفت غیر کلید اصلی با کلید اصلی وابستگی تابعی **بی واسطه** داشته باشد.

نرمال سازی معکوس:

زمانی که سطح نرمال رابطه‌ها را از سطح‌های بالاتر به سطح پایین‌تری بیاوریم ،
اصطلاحاً عمل نرمال سازی معکوس گفته می‌شود.