

# پایگاه داده ها

معرفی اجمالی

# مراجع



- **A. Silberschatz, H. F. Korth, and S. Sudarshan, Database System Concepts, McGraw-Hill, 2006.**
- **C. J. Date, An Introduction to Database Systems, Addison Wesley, 2003.**
- **R. Elmasri and S. B. Navathe, Fundamentals of Database Systems, Addison Wesley, 2006.**

# نحوه ارزیابی



- امتحان میان ترم
- امتحان پایان ترم
- پروژه برنامه نویسی
- تکالیف

# سیستم‌های مدیریت پایگاه داده‌ها



- سیستم مدیریت پایگاه داده‌ها (DBMS)
  - مجموعه‌ای از داده‌های مرتبط به هم و مجموعه‌ای از برنامه‌ها برای دستیابی به آن داده‌ها
  - مجموعه داده‌های مرتبط به هم معمولاً به عنوان پایگاه داده‌ها رجوع می‌شود
- هدف اصلی از یک سیستم مدیریت پایگاه داده‌ها
  - فراهم کردن محیطی مناسب برای ذخیره و بازیابی اطلاعات پایگاه داده‌ها

## سیستم‌های مدیریت پایگاه داده‌ها



- در سالیان اولیه پیدایش کامپیوترها، برنامه‌های کاربردی پایگاه داده‌ای مستقیماً بالای سیستم‌های فایل ایجاد می‌شدند

• مثال

- در یک سیستم بانکی، اطلاعات مربوط به مشتریان، حساب‌های پس‌انداز و حساب‌های جاری در تعدادی فایل ذخیره می‌شد

- معایب استفاده از سیستم‌های فایل برای ذخیره‌سازی داده‌ها

• افزونگی داده‌ها

- اطلاعات یکسان ممکن است در چندین فایل تکرار شوند
- آدرس و شماره تلفن یک مشتری خاص ممکن است هم در فایل حاوی حساب‌های پس‌انداز و هم در فایل حاوی حساب‌های جاری ذخیره شود

# سیستم‌های مدیریت پایگاه داده‌ها



## • ناسازگاری داده‌ها

- کپی‌های متفاوت از داده‌های یکسان ممکن است با هم تطبیق نداشته باشند
- تغییر آدرس مشتری ممکن است تنها در فایل حاوی حساب‌های پس‌انداز اعمال شود

## • دشواری در دستیابی به داده‌ها

- نیاز به نوشتن یک برنامه جدید به منظور به انجام رساندن هر وظیفه جدید
- نام‌های همه مشتریانی که در یک منطقه پستی خاص زندگی می‌کنند

## • مشکلات مربوط به صحت

- مقادیر داده‌های ذخیره شده در پایگاه داده‌ها ممکن است لازم باشد انواع خاصی از قیود سازگاری را برآورده کنند
- موجودی انواع خاصی از حساب‌های بانکی هرگز نباید از یک مبلغ از قبل تعیین شده کمتر باشد

# سیستم‌های مدیریت پایگاه داده‌ها



- مشکلات مربوط به تجزیه ناپذیری
  - در بسیاری از کاربردها لازم است در صورت مشاهده خرابی در سیستم، داده‌ها به وضعیت سازگار پیش از خرابی بازگردانده شوند
  - فرض کنید هدف انتقال ۵۰ دلار از حساب A به حساب B باشد
  - وقوع خرابی در سیستم پس از برداشت ۵۰ دلار از حساب A و قبل از واریز آن به حساب B
- مشکلات ناشی از دستیابی همروند
  - در بسیاری از سیستم‌ها به منظور افزایش کارایی به چندین کاربر اجازه داده می‌شود تا داده‌ها را همزمان به روزرسانی کنند
  - فرض کنید موجودی حساب A ۵۰۰ دلار باشد
  - دو کاربر موجودی حساب A را بخوانند و همزمان ۵۰ و ۱۰۰ دلار به حساب A واریز کنند
  - موجودی حساب A ممکن است به جای ۶۵۰ دلار ۵۵۰ یا ۶۰۰ دلار باشد

# سیستم‌های مدیریت پایگاه داده‌ها



- مشکلات امنیتی
- در بسیاری از کاربردها لازم است به هر کاربر تنها اجازه دستیابی به بخشی از داده‌ها داده شود



# انتزاع داده‌ها



- یکی از اهداف اصلی از یک سیستم پایگاه داده‌ای فراهم کردن یک دید انتزاعی از داده‌ها است
- مخفی‌سازی جزئیات مربوط به ذخیره‌سازی و نگهداری داده‌ها

- سطوح انتزاع
  - سطح فیزیکی
  - سطح منطقی
  - سطح دید

# سطوح انتزاع

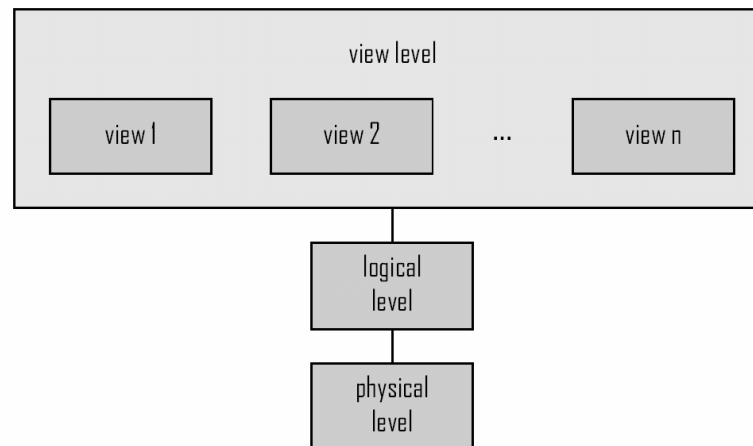


- سطح فیزیکی
  - پایین ترین سطح از انتزاع
  - چگونگی ذخیره سازی واقعی داده ها را توصیف می کند
- سطح منطقی
  - سطح بالاتری از انتزاع
  - داده های ذخیره شده در پایگاه داده ها و روابط میان آنها را توصیف می کند

# سطوح انتزاع

## • سطح دید

- بالاترین سطح از انتزاع
- تنها بخشی از پایگاه داده‌ها را توصیف می‌کند
- در بسیاری از مواقع لازم نیست کاربران به همه اطلاعات ذخیره شده در پایگاه داده‌ها دسترسی داشته باشند



# شماها و نمونه‌ها



- شماي پایگاه داده‌ها (Database Schema)
  - طراحی کلی پایگاه داده‌ها
  - انواع شماها
    - شماهای فیزیکی
      - طراحی پایگاه داده‌ها در سطح فیزیکی
    - شماهای منطقی
      - طراحی پایگاه داده‌ها در سطح منطقی
- نمونه (Instance)
  - مجموعه اطلاعات ذخیره شده در پایگاه داده‌ها در یک لحظه خاص

# پایگاه داده‌های رابطه‌ای



- هر پایگاه داده‌های رابطه‌ای مبتنی بر مدل رابطه‌ای است
- از مجموعه‌ای از جداول برای نمایش داده‌ها و روابط میان آن‌ها استفاده می‌کند

- هر جدول شامل چندین ستون است
- هر ستون یک نام منحصر به فرد دارد

<i>customer_id</i>	<i>customer_name</i>	<i>customer_street</i>	<i>customer_city</i>	<i>account_number</i>
192-83-7465	Johnson	12 Alma St.	Palo Alto	A-101
192-83-7465	Johnson	12 Alma St.	Palo Alto	A-201
677-89-9011	Hayes	3 Main St.	Harrison	A-102
182-73-6091	Turner	123 Putnam St.	Stamford	A-305
321-12-3123	Jones	100 Main St.	Harrison	A-217
336-66-9999	Lindsay	175 Park Ave.	Pittsfield	A-222
019-28-3746	Smith	72 North St.	Rye	A-201

# پایگاه داده‌های رابطه‌ای



<i>customer_id</i>	<i>customer_name</i>	<i>customer_street</i>	<i>customer_city</i>
192-83-7465	Johnson	12 Alma St.	Palo Alto
677-89-9011	Hayes	3 Main St.	Harrison
182-73-6091	Turner	123 Putnam Ave.	Stamford
321-12-3123	Jones	100 Main St.	Harrison
336-66-9999	Lindsay	175 Park Ave.	Pittsfield
019-28-3746	Smith	72 North St.	Rye

(a) The *customer* table

<i>account_number</i>	<i>balance</i>
A-101	500
A-215	700
A-102	400
A-305	350
A-201	900
A-217	750
A-222	700

(b) The *account* table

<i>customer_id</i>	<i>account_number</i>
192-83-7465	A-101
192-83-7465	A-201
019-28-3746	A-215
677-89-9011	A-102
182-73-6091	A-305
321-12-3123	A-217
336-66-9999	A-222
019-28-3746	A-201

(c) The *depositor* table

# زبان‌های پایگاه داده‌ای



- هر سیستم پایگاه داده‌ای دو زبان متفاوت را فراهم می‌کند
  - زبان تعریف داده‌ها (DDL)
  - توصیف شمای پایگاه داده‌ها
  - زبان عملیات روی داده‌ها (DML)
  - بیان پرس و جوها و به روزرسانی اطلاعات ذخیره شده در پایگاه داده‌ها
- این دو زبان در واقع دو زبان جداگانه نمی‌باشند بلکه قسمت‌هایی از یک زبان پایگاه داده‌ای واحد را تشکیل می‌دهند

## زبان عملیات روی داده‌ها (DML)



- **DML** به کاربران امکان می‌دهد تا به داده‌های موجود در یک سیستم پایگاه داده‌ای دسترسی داشته باشند و عملیات مختلفی را بر روی آن داده‌ها انجام دهند
  - بازیابی اطلاعات ذخیره شده در پایگاه داده‌ها
  - افزودن اطلاعات جدید به پایگاه داده‌ها
  - حذف اطلاعات از پایگاه داده‌ها
  - اصلاح اطلاعات ذخیره شده در پایگاه داده‌ها



# زبان عملیات روی داده‌ها (DML)



- زبان پرس و جو (Query Language)
- بخشی از DML که با بازیابی اطلاعات سروکار دارد
- زبان پرس و جوی SQL
- یکی از شناخته‌شده‌ترین زبان‌های پرس و جو
- مثال
- یافتن نام مشتری با شناسه 192-83-7465

```
select customer.customer_name  
from customer  
where customer.customer_id = '192-83-7465'
```

# زبان تعریف داده‌ها (DDL)



- از DDL برای توصیف شمای پایگاه داده‌ها استفاده می‌شود

• مثال

- تعریف جدول *account* در زبان SQL

```
create table account  
  (account_number char(10),  
   balance integer)
```

- مقادیر داده‌های ذخیره شده در پایگاه داده‌ها ممکن است لازم باشد قیود سازگاری خاصی را برآورده کنند

• مثال

- موجودی یک حساب خاص نباید کمتر از ۱۰۰ دلار شود

- DDL امکاناتی برای توصیف قیود سازگاری فراهم می‌کند

# طراحی پایگاه داده‌ها



- فرآیند طراحی در مدل رابطه‌ای شامل تصمیم‌گیری درباره موارد زیر است
  - چه خصیصه‌هایی باید در پایگاه داده‌ها ثبت شوند
  - چگونه این خصیصه‌ها باید گروه‌بندی شوند تا جداول متفاوت را شکل دهند
- استفاده از مدل موجودیت-وابستگی (Entity-Relationship Model)
- نرمال‌سازی (Normalization)

# طراحی پایگاه داده‌ها



- طراحی پایگاه داده‌ها برای سیستم بانکی
- بانک دارای تعدادی شعبه است. هر شعبه در یک شهر خاص قرار دارد و دارای یک نام منحصر به فرد است
- بانک به هر مشتری یک شناسه منحصر به فرد اختصاص می‌دهد. بانک نام، خیابان و شهر محل سکونت هر مشتری را ذخیره می‌کند
- مشتریان بانک می‌توانند حساب بانکی داشته باشند و وام دریافت کنند
- در بانک امکان افتتاح دو نوع حساب پس‌انداز و جاری وجود دارد
- هر حساب بانکی می‌تواند بین چندین مشتری مشترک باشد و هر مشتری می‌تواند چندین حساب بانکی داشته باشد
- هر حساب پس‌انداز دارای یک نرخ بهره است

## طراحی پایگاه داده‌ها



- هر مشتری می‌تواند از بانک وام دریافت کند. هر وام ممکن است بین چندین مشتری مشترک باشد
- کارمندان بانک با شماره کارمندی‌شان شناخته می‌شوند
- مدیریت بانک نام، شماره تلفن، نام‌های افراد تحت تکفل و شماره کارمندی مدیر هر کدام از کارمندان بانک را ذخیره می‌کند

# مدل موجودیت-وابستگی



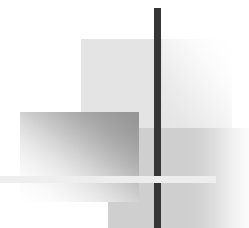
- در مدل موجودیت-وابستگی، هر سازمان به صورت مجموعه‌ای از موجودیت‌ها و وابستگی‌ها مدل می‌شود
- موجودیت (Entity)
  - یک شیء در دنیای واقعی که از سایر اشیاء قابل تمایز است
    - مشتریان
    - حساب‌های بانکی
- هر موجودیت در پایگاه داده‌ها با استفاده از یک مجموعه از خصیصه‌ها (attributes) توصیف می‌شود
  - خصیصه‌های شماره حساب (*account\_number*) و موجودی (*balance*)، یک حساب بانکی خاص را توصیف می‌کنند

# مدل موجودیت-وابستگی



- وابستگی (Relationship)
- ارتباط میان چندین موجودیت را مشخص می کند
- وابستگی سپرده گذار (*depositor*)، هر مشتری را به شماره حسابش مرتبط می کند
- مجموعه موجودیت (Entity Set)
- مجموعه همه موجودیتها از نوع یکسان
- مجموعه وابستگی (Relationship Set)
- مجموعه همه وابستگیها از نوع یکسان

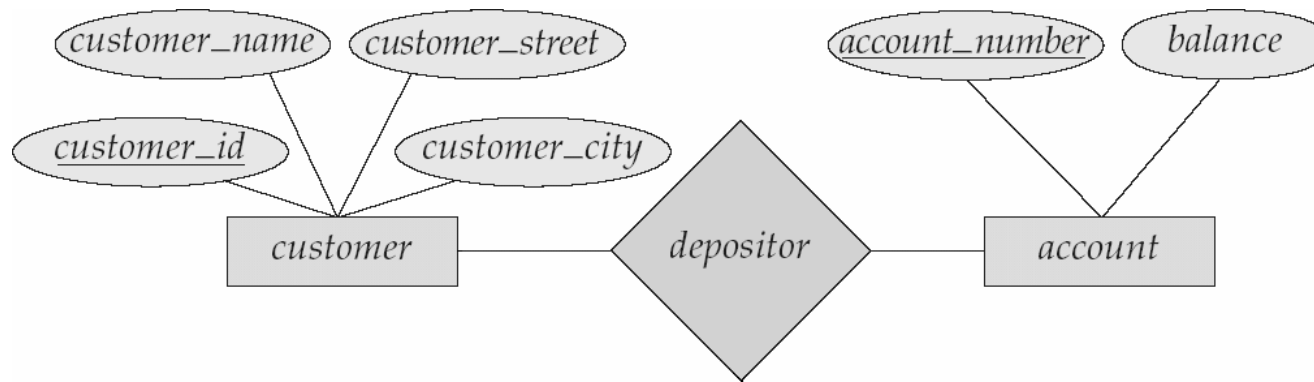
# مدل موجودیت-وابستگی



- ساختار منطقی (شمای) هر پایگاه داده‌ها را می‌توان به صورت گرافیکی توسط یک نمودار E-R نمایش داد
  - مستطیل‌ها
  - مجموعه‌های موجودیت
  - بیضی‌ها
  - خصیصه‌ها
  - لوزی‌ها
  - مجموعه‌های وابستگی
  - خطوط
- پیوند خصیصه‌ها به مجموعه‌های موجودیت و مجموعه‌های موجودیت به مجموعه‌های وابستگی



# مدل موجودیت-وابستگی



- مجموعه‌های موجودیت *customer* و *account*
- مجموعه وابستگی *depositor*

# نرمال سازی



- هدف از نرمال سازی طراحی شِمَاهایی است که در شکل نرمال مناسب باشند
- دلایل نیاز به نرمال سازی
  - تکرار اطلاعات
  - عدم توانایی برای نمایش برخی اطلاعات

# دلایل نیاز به نرمال سازی

- تکرار اطلاعات

- باعث پیچیده شدن فرآیند به روز رسانی پایگاه داده ها می شود

- مثال

- فرض کنید در سیستم بانکی به جای دو جدول *account* و *depositor* تنها یک جدول *'depositor'* وجود داشته باشد

<i>customer_id</i>	<i>account_number</i>	<i>balance</i>
192-83-7465	A-101	500
192-83-7465	A-201	900
019-28-3746	A-215	700
677-89-9011	A-102	400
182-73-6091	A-305	350
321-12-3123	A-217	750
336-66-9999	A-222	700
019-28-3746	A-201	900

- در صورتی که هدف تغییر موجودی حساب بانکی A-201 از ۹۰۰ دلار به ۹۵۰ دلار باشد، این تغییر باید در دو ردیف اعمال شود

# دلایل نیاز به نرمال سازی

- عدم توانایی برای نمایش برخی اطلاعات
- مثال

• فرض کنید به جای دو جدول *customer* و *depositor* تنها یک جدول *customer'* وجود داشته باشد

<i>customer_id</i>	<i>customer_name</i>	<i>customer_street</i>	<i>customer_city</i>	<i>account_number</i>
192-83-7465	Johnson	12 Alma St.	Palo Alto	A-101
192-83-7465	Johnson	12 Alma St.	Palo Alto	A-201
677-89-9011	Hayes	3 Main St.	Harrison	A-102
182-73-6091	Turner	123 Putnam St.	Stamford	A-305
321-12-3123	Jones	100 Main St.	Harrison	A-217
336-66-9999	Lindsay	175 Park Ave.	Pittsfield	A-222
019-28-3746	Smith	72 North St.	Rye	A-201

- امکان نمایش اطلاعات مربوط به یک مشتری خاص وجود ندارد، مگر آن که آن مشتری دارای یک حساب بانکی باشد

# مدیریت تراکنش



- تراکنش (Transaction)
- مجموعه عملیاتی که یک وظیفه منطقی واحد را در یک برنامه کاربردی پایگاه داده‌ای به انجام می‌رساند
- انتقال وجوه که در آن از موجودی یک حساب بانکی کسر و به موجودی حساب بانکی دیگر اضافه می‌شود
- لازم است که هر دو عملیات کسر شدن از یک حساب بانکی و واریز شدن به حساب بانکی دیگر یا با هم مشاهده شوند یا هیچ کدام به تنهایی مشاهده نشود
- تجزیه ناپذیری (Atomicity)
- ویژگی همه یا هیچ

# دستیابی به پایگاه داده از برنامه‌های کاربردی

- در برنامه‌های کاربردی برای دستیابی به پایگاه داده‌ها از واسط‌های برنامه کاربردی استفاده می‌شود
- از واسط‌های فوق برای ارسال دستورات DDL و DML به پایگاه داده‌ها و بازیابی نتایج استفاده می‌شود
- انواع استانداردها
  - ODBC
  - JDBC
  - ADO
  - ADO.NET