

# Data Base



سه شنبه

فروردین

28 MARCH 2017

۲۹ جمادی الثانی ۱۴۳۸



بیمه آرکان

Note

Data (types) → Structured  
→ Unstructured

Data → Storage  
→ Retrieval

- DBMS = Data Base Management System → Oracle { connection of interrelated data  
Set of programs to access data  
convenient and efficient to use
- The need for databases → Accessing various data → Banking  
→ Airlines  
→ Sales  
→ Universities  
→ human resources  
→ Manufacturing  
→ Online shopping

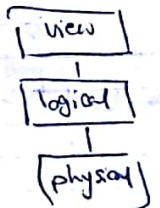
- First, databases were file systems.

- problems
- 1- Data Redundancy & inconsistency
  - 2- Difficulty in accessing data
  - 3- Data isolation
  - 4- Integrity problem
  - 5- Atomicity of updates
  - 6- Concurrent Access by multiple users
  - 7- Security problems

- Database = collection of interrelated data and a set of programs that allows users to access and modify these data

→ purposes → providing an abstract view of the data  
→ hiding details of how data are stored and maintained

- Levels of Abstraction
  - physical : how data is stored actually
  - logical : what data, relationship between data
  - view : hiding certain info from certain users



ملاحظات

Data Model = structure of dbs describing Data

- Relational → tables
- Entity-Relation ✓✓
- Object-based
- Semistructured (XML)

" relationships  
" semantics  
" constraints

ش	ی	د	س	چ	پ	ج
۱	۲	۳	۴	۵	۶	۷
۸	۹	۱۰	۱۱	۱۲	۱۳	۱۴
۱۵	۱۶	۱۷	۱۸	۱۹	۲۰	۲۱
۲۲	۲۳	۲۴	۲۵	۲۶	۲۷	۲۸
۲۹	۳۰	۳۱				



بیمه آران

- ② Schema = Overall design of db
- Instance = the actual content of db ( $\approx$  value of variable)

- physical data independence
- DDL = notation for db schema

Physical (type of variable)  
Logical

- Domain Constraints
- Referential Integrity
- Assertions
- Authorization

چهارشنبه

فروردین

MARCH 2017

۳۰ جمادی الثانی ۱۴۳۸

۹

یادداشت

Data of Data



- Data Dictionary contains metadata that describes info about db
- DML  $\approx$  query language types
  - Procedural = what data & how to get it
  - Declarative = what data (No matter how to get it)
- types of access
  - insert info
  - delete info
  - Modify info
  - Retrieve info
- Language classes
  - pure  $\xrightarrow{\text{eg}}$  Relational Algebra
  - commercial  $\xrightarrow{\text{eg}}$  SQL

- Database design
  - logical : Schema
    - Business decision : what attributes
    - CS decision : what relation
  - Physical

- Design Approaches (for "good" relation design) :
  - 1- Entity Relation Model
  - 2- Normalization Theory

- Specification of Functional Requirements : kinds of operations that can be performed on data  
 $\rightarrow$  (delete? insert? search? ...)

## ① Entity - Relationship Model

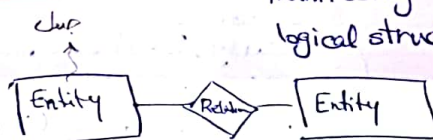
$\rightarrow$  Relation between objects (entities)

$\rightarrow$  described by a set of attributes

اول اصل به روابط کار نداریم، موجودیت ها رو مشخص می کنیم (استاد، دانشجو، دانشکده، ...)  
 بعد می بینیم به روابط بین این موجودیت ها می پردازیم

- UML = Unified Modeling Language

used for graphically manifesting the logical structures



فروردین	ش	ی	د	س	چ	پ	ج	ملاحظات
۱	۲	۳	۴	۵	۶	۷	۸	۹
۱۰	۱۱	۱۲	۱۳	۱۴	۱۵	۱۶	۱۷	۱۸
۱۹	۲۰	۲۱	۲۲	۲۳	۲۴	۲۵	۲۶	۲۷
۲۸	۲۹	۳۰	۳۱					



## • ② Normalization

اوسن چينجوي تا آ داده ها تو ميريزي رو فيز،  
بكر بويج مي كنن با توجه به دنيا و واقعات و  
فان نقض هايي كه ميديجي كنن، جدول رو مي شكوني و  
مستق مي كنن به به سنجاي سوال برسي.  
كه بازيابي داده و زخير اطلاعات توسن راحته.  
(واقعه افروني بيوه)



پنجشنبه  
فروردین  
30 MARCH 2017  
۱ رجب ۱۴۳۸

Note

problems of a bad design { Repetition of info  
Inability to represent certain info

• Storage Manager : interface between low-level data stored in db & (application & queries)

\* tasks : { Interaction with OS file manager  
Efficient { store retrieve & data update

\* Includes { Transaction manager = handling failure or concurrent transactions  
File manager = handling allocation of space / data structures  
Buffer Manager = fetch data into main memory / handle large-sized data  
Authorization / Integrity manager = authority of users / satisfaction of constraints

\* Data Structures in physical system implementation

{ Data files = stores db itself  
Data Dictionary = stores metadata  
Indices = fast access to data items.

ولادت حضرت امام محمد باقر (ع) (۵۷ هـ ق)

چهارشنبه  
فروردین  
31 MARCH 2017  
۲ رجب ۱۴۳۸

## • Query Processing

{ 1. Parsing and translation (توليد و ترجمه)  
2. Optimization  
3. Evaluation (اجرا)

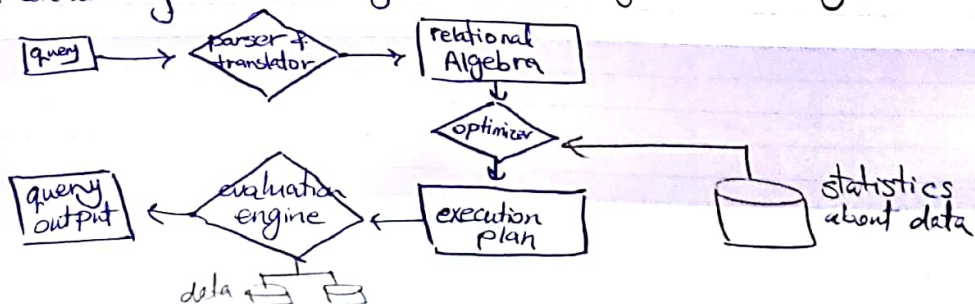
• Transaction manager = keeps db in a consistent state, despite { system failure  
transaction failure

\* Recovery manager = restore the state of db before the failure

\* Concurrency-control manager = consistency of db during concurrent transactions

فروردین

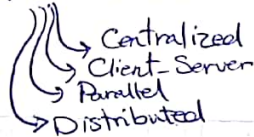
ش	ی	د	س	چ	پ	ج
۴	۳	۲	۱			
۱۱	۱۰	۹	۸	۷	۶	۵
۱۸	۱۷	۱۶	۱۵	۱۴	۱۳	۱۲
۲۵	۲۴	۲۳	۲۲	۲۱	۲۰	۱۹
۳۱	۳۰	۲۹	۲۸	۲۷	۲۶	



ملاحظات



- Data Mining : semi-auto analyzing db to find useful patterns.
- Info retrieval : querying of unstructured textual data
- Database Architecture is influenced by underlying computer systems.



شنبه

فروردین

1 APRIL 2017

۲ رجب ۱۴۳۸

یادداشت

## • History of db systems

★ 1950 , 1960 : magnetic tapes , punched cards

★ 1960 , 1970 : { Hard disk (direct access to data)  
Widespread use of Network  
high-performance transaction processing  
relational data model

★ 1980 : { Object-oriented database System  
Parallel and distributed db "  
Relational evolves to commercial (SQL)

Early  
★ 2000 : XML and XQuery

★ 1990 : { Web commerce  
data-mining apps  
multi-terabyte data warehouses



یکشنبه

فروردین

2 APRIL 2017

۲ رجب ۱۳۹۸

## Note

- Domain = allowed values for atts
- Att.s should be atomic  $\rightarrow$  e.g. "9631046" do not use for entrance year
- null  $\in$  every domain  
 $\rightarrow$   $\equiv$  unknown value  $\rightarrow$  causes complication

Instructor

(3)

ID	name	dept-name	Salary

attributes  
(column)

tuples  
(rows)  $\rightarrow$  arbitrary order



بیمه آران

	definition	example
$K \leftarrow$ superkey	sufficient to identify a unique tuple of a relation	$\{ID\}$ , $\{ID, name\}$ , ...
candidate key	if $K$ is minimal	$\{ID\}$
primary key	One of the candidate keys	
foreign key	When value in one relation appears in another (Must be primary in its own table)	dept-name

## Relational Algebra

- \* Selection of Rows  $\sigma$  / Select
- \* Selection of columns  $\pi$  / Project
- \* Union of relations  $\cup$
- \* Set difference of relations  $-$

فروردین

ش	ی	د	س	چ	پ	ج
۱	۲	۳	۴	۵	۶	۷
۸	۹	۱۰	۱۱	۱۲	۱۳	۱۴
۱۵	۱۶	۱۷	۱۸	۱۹	۲۰	۲۱
۲۲	۲۳	۲۴	۲۵	۲۶	۲۷	۲۸
۲۹	۳۰	۳۱				

Set  
\* Intersection of relations  $\cap$

\* Joining two relations (Cartesian Product)  $\times$

\* Rename  $\rho_x(E) \rightarrow$  returns the expression  $E$  under the name  $X$

\* Natural Join  $\bowtie$

\* Each query input is a table.  
Also, " " output " " " "

روز طبیعت (تعطیل)

ملاحظات





< >  $\rightarrow$  not equal to

دوشنبه

فروردین

3 APRIL 2017

۵ رجب ۱۴۳۸



یادداشت

## Summary of Relational Algebra Operators

Symbol (Name)	Example of Use
$\sigma$ (Selection)	$\sigma \text{ salary} \geq 85000 \text{ (instructor)}$ Return rows of the input relation that satisfy the predicate.
$\Pi$ (Projection)	$\Pi \text{ ID, salary (instructor)}$ Output specified attributes from all rows of the input relation. Remove duplicate tuples from the output.
$\times$ (Cartesian Product)	$\text{instructor} \times \text{department}$ Output all pairs of rows from the two input relations (regardless of whether or not they have the same values on common attributes)
$\cup$ (Union)	$\Pi \text{ name (instructor)} \cup \Pi \text{ name (student)}$ Output the union of tuples from the two input relations.
$-$ (Set Difference)	$\Pi \text{ name (instructor)} - \Pi \text{ name (student)}$ Output the set difference of tuples from the two input relations.
$\bowtie$ (Natural Join)	$\text{instructor} \bowtie \text{department}$ Output pairs of rows from the two input relations that have the same value on all attributes that have the same name.

۳	۴	۵	۶	۷	۸	۹	۱۰	۱۱	۱۲	۱۳	۱۴	۱۵	۱۶	۱۷	۱۸	۱۹	۲۰	۲۱	۲۲	۲۳	۲۴	۲۵	۲۶	۲۷	۲۸	۲۹	۳۰	۳۱
---	---	---	---	---	---	---	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----