

# Week 1

- o) Database = A collection of related data
  - = Represent some aspects of real world called the miniworld / universe of discourse
  - abstract of real world
- o) Database can be of any size & complexity
- o) Database Management System (DBMS)
  - = A collection of programs that enable users to create & maintain a database
  - = A general-purpose software system that facilitates the processes of **defining**, **constructing**, **manipulating**, and **sharing** databases among various users and applications.
- o) **Defining** database involves specifying data types, structures, and constraints of the data to be stored in the database.
- o) Definisi database/deskripsi nya juga disimpan dalam bentuk catalog/dictionary yang disebut **meta-data**.
  - ↳ Misalnya ada entity employee, trs ada attribute nama\_employee, berarti itu meta-datanya
  - ↳ Misalnya ada table, meta-datanya adalah nama table, nama kolom, data type
- o) **Constructing** database = Proses simpan data on some storage medium dicontrol oleh DBMS.
- o) **Manipulating** database includes functions such as querying the database to retrieve specific data, updating database to reflect changes in miniworld.
- o) **Sharing** database allows multiple users & programs to access database simultaneously.

o) A number of characteristics distinguish the database approach from the traditional approach of programming with files.

- ↳ Self - describing nature of database system
- ↳ Insulation between programs & data, & data abstraction
- ↳ Support of multiple views of the data
- ↳ Sharing of data and multiuser transaction processing

a) Self - describing nature of database system

↳ Database system ga hny contain database itself, tapi juga complete definition / description of database structure and constraints.

↳ Definisi ini disimpan di DBMS catalog. Informasinya stored di catalog dinamakan meta-data. ↳ bisa diakses oleh user

b) Insulation between programs & data, & data abstraction

↳ DBMS access programs don't require such changes in most cases.

Struktur data file disimpan di DBMS catalog berbeda di program akses.

↳ In traditional file processing, struktur data files is embedded in app programs. Any changes to structure of a file may require changing all programs that access that file.

↳ Program data independence

c) Support of multiple views of the data

↳ A view may be a subset of the database or it may contain virtual data that is derived from the database files but isn't explicitly stored.

↳ 1 orang nengok database buat check aja, 1 orang lain nengok database buat akses & print doang

d) Sharing of data and multiuser transaction processing

↳ DBMS must include concurrency control software to ensure that several users try to update bersamaan

↳ A transaction is an executing program/proses that include 1/more database accesses, such as reading/updating of database records.

⑨ Actors on the scene : DATA BASE

- admin
- designer
- end users

- System Analysts & App Programmers  
(Software Engineers)

⑨ Database End Users :

- Casual
- Sophisticated
- Naïve / Parametric
- Standalone users

END USERS

⑨ Keuntungan DBMS :

- Control redundancy
- Restrict unauthorized access
- Providing storage structures for efficient query processing
- Providing backup & recovery
- Enforcing integrity constraints

### a) Control redundancy

↳ Di tradisional software development pke file processing, tiap user grup jaga filenya sendiri utk handling data-processing app-nya.

↳ Duplication of effort : Need to perform a single update berkali-kali  
Storage space is wasted

Menjadi inconsistent karena kl ada update data ke 1 file tidak ke file lain

### b) Restricting unauthorized access

↳ Users ga diauthorize utk akses smlw info di database dlm skala bsr

### c) Providing storage structures for efficient query processing

↳ Index : Specialized data struktur utk speed up disk search for desired records

◦ Based on tree or hash, modified for disk search

↳ Buffering module : Maintain parts of database di main memory buffers

◦ Utk proses database direcord needed by particular query, record itu hrs copy dr disk ke memory

↳ Query processing and optimization dr DBMS utk choose efficient query execution plan utk tiap query based on existing storage structures.

#### d) Providing backup & recovery

↳ Must provide facilities dari hardware / software failures

↳ The backup and recovery subsystem dr DBMS is responsible for recovery.

#### e) Enforcing integrity constraints

↳ Data types constraints specifies data type utk tiap item

↳ Referential integrity constraints specifies record di 1 file hrs ad relasi di file lain

↳ Key constraint specifies uniqueness on data item values

#### c) Kapan ga pke DBMS?

↳ Simple database yg ga akan berubah

↳ No multiple-user yg akses database

↳ Stringent, real-time requirements for some programs that may not met because of DBMS overhead

#### c) Data Model = Collection of konsep digunakan utk describe struktur database. Kategorinya :

##### 1) High Level / Conceptual Data Models

- Entity relationship model
- Provide konsep cara user menerima data
- Entity merepresentasikan real-world object / konsep
- Attribute merepresentasikan some property of interest that further describes an entity

##### 2) Representational / Implementation Data Models

- Plg banyak pke di traditional commercial DBMS
- Relational data is widely used
- Represent data by using record structures, mly kndng dipanggil sbg record-based data models

##### 3) Low Level / Physical Data Models

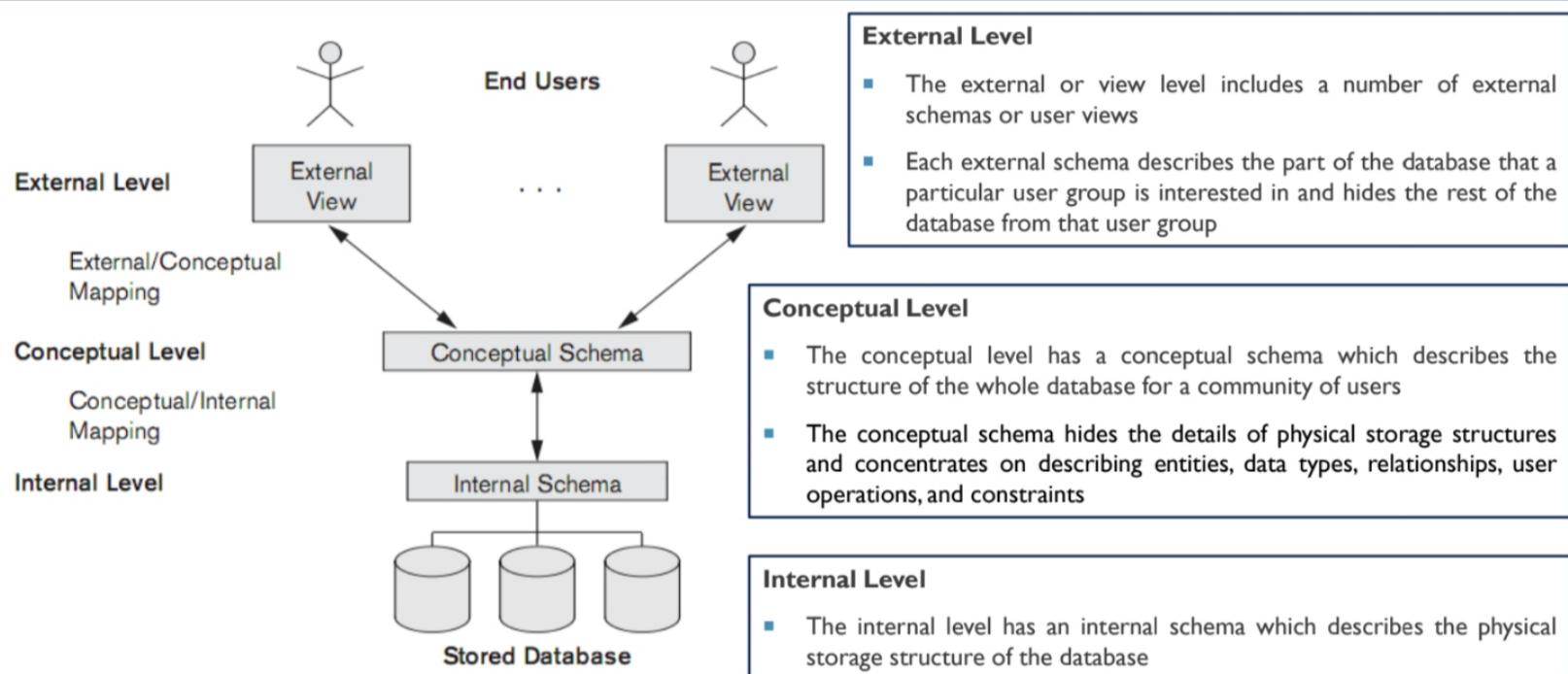
- Describe data stored as files di computer by representing information seperti record formats, record orderings, and access path.

o Access Path = Struktur yg membuat search particular database records efficient

o Deskripsi database = Database schema

o Data in database at particular time = Database state / snapshot

## o THREE-SCHEMA ARCHITECTURE



o Data Independence = Capacity to change the schema at one level of database system without having to change the schema at next higher level

### 1) Logical Data Independence

The capacity to change conceptual schema without having to change external schemas or program

### 2) Physical Data Independence

The capacity to change conceptual schema without having to change conceptual schema & external schemas need not be changed as well.

\*Mending baca PPT langsung

## Week 5

- The smallest SELECT statement di-specified consists of 1 select block dengan select clause.
- Kalau ga ada FROM clause, statementnya return hasil yang terdiri dari 1 baris.
- ```
SELECT actor_id,
            50000 AS bonus, → New column names tidak dapat dipake di SELECT clause
            salary + bonus AS total
        FROM roles;
```

yang sama.

| ◦ Comparison Operator | Meaning                  |
|-----------------------|--------------------------|
| =                     | Equal to                 |
| <= >=                 | Equal to also if null    |
| <                     | Less than                |
| >                     | Greater than             |
| <=                    | Less than or equal to    |
| >=                    | Greater than or equal to |
| <>                    | Not equal to             |
| !=                    | Not equal to             |

- Comparison operators : Put in WHERE clause.
  - Combine with AND, OR, NOT, XOR
- IN clause :  $\text{year} \in \{2010, 2012\}$ .  
 $\text{year} = 2010 \& \text{year} = 2012$ , bukan  $2010 \leq \text{year} \leq 2012$
- Kalau  $2010 \leq \text{year} \leq 2012$  pakai BETWEEN, seperti :  
 ◦ WHERE year BETWEEN 2010 AND 2012
- LIKE clause : "%....%"  
 $\hookrightarrow "0\%" \rightarrow \text{Dis}$     $\hookrightarrow "\%d\%" \rightarrow \text{Nicholas}$     $\hookrightarrow "\_d\_" \rightarrow \text{Jin}$   
 $\text{Dus}$     $\text{Ryan}$     $\text{---}$   
 $\text{Das}$     $\text{Joy}$     $\text{Kul}$   
 $\text{Tak}$

- IS NULL operator ◦  $\text{WHERE rank IS NULL}$ ,  
↳ Tampil yang NULL.  
↳ IS NOT NULL → Kebalikannya.
  - DISTINCT ◦ Biar tidak double
- Aggregation function ◦ COUNT(\*) Function  
↳ Hitung ada berapa banyak dilengkapi dengan GROUP BY.
- CCOUNT(DISTINCT year)  
↳ Hitung ada berapa macam year
- MAX & MIN function
- SUM function
- AVG function

## Week 6

- Group by clause dipakai saat ingin mengelompokkan data tiap kolom dengan masing-masing nilainya yang mungkin akan dihitung/dikelompokkan.
- GROUP\_CONCAT Function : mengelompokkan data dengan id/hasil yang sama dengan data lain dipisahkan oleh koma
- Roll up function → Kumulatif, dengan cari mana yang digroupby lalu dilanjutkan pertambahan.
- HAVING clause dipakai utk select baris setelah GROUP BY clause dieksekusi. Bs contain aggregation function.
- WHERE clause dipakai utk select baris setelah FROM clause dieksekusi. Gbs contain aggregation functions.
- ORDER BY clause → urut kolom
  - ↳ DESC ◦ Descending
  - ↳ ASC ◦ Ascending
- LIMIT clause → Batasi sampai baris ke - n.
- OFFSET clause → Skip sebesar m baris  
LIMIT 5, OFFSET 3 = LIMIT 3,5
- CASE
  - WHEN ... THEN
  - END AS...
- Use variable assign ◦  
SET @salary = 250
- > ALL => Max ◦ > ANY => Min
- < ALL =< Min ◦ < ANY =< Max ◦ = ANY = IN

## Week 4

- Database acts like container untuk set of tables.
- Table di-create di database yang sudah ada.
- Delete : DROP DATABASES
- Kalau mau specify database yang dikerjai, pakai USE database name.

| Integer Literal | Range                                                    |
|-----------------|----------------------------------------------------------|
| TINYINT         | -2 <sup>7</sup> up to and including +2 <sup>7</sup> -1   |
| SMALLINT        | -2 <sup>15</sup> up to and including +2 <sup>15</sup> -1 |
| MEDIUMINT       | -2 <sup>23</sup> up to and including +2 <sup>23</sup> -1 |
| INTEGER         | -2 <sup>31</sup> up to and including +2 <sup>31</sup> -1 |
| BIGINT          | -2 <sup>63</sup> up to and including +2 <sup>63</sup> -1 |

| Original Data Type | Synonyms        |
|--------------------|-----------------|
| TINYINT            | INT1            |
| SMALLINT           | INT2            |
| MEDIUMINT          | INT3, MIDDLEINT |
| INTEGER            | INT, INT4       |
| BIGINT             | INT8            |

- Database dibuat oleh database object / table.
- Decimal data types
  - Decimal
  - Dec
  - Numeric
  - Fixed
- Float data types
  - Float
  - Float4
  - Real
  - Double

| Alphanumeric Data Type | Maximum Length                                                                      |
|------------------------|-------------------------------------------------------------------------------------|
| CHAR                   | (2 <sup>8</sup> -1) characters                                                      |
| VARCHAR                | (2 <sup>8</sup> -1) characters<br>(2 <sup>16</sup> -1) characters from version 5.03 |
| LONGVARCHAR            | (2 <sup>24</sup> -1) characters                                                     |
| LONGTEXT               | (2 <sup>32</sup> -1) characters                                                     |

| Alphanumeric Data Type | Synonyms                                                                                                                                                |
|------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------|
| CHAR                   | CHARACTER<br>NCHAR<br>NATIONAL CHAR<br>NATIONAL CHARACTER                                                                                               |
| VARCHAR                | CHAR VARYING<br>CHARACTER VARYING<br>NATIONAL VARCHAR<br>NATIONAL CHAR VARYING<br>NATIONAL CHARACTER VARYING<br>TEXT<br>TINYTEXT (maximum length = 255) |
| LONGVARCHAR            | MEDIUMTEXT                                                                                                                                              |
| LONGTEXT               |                                                                                                                                                         |

- Temporal data types ◦ Date

- Time
- Datetime
- Timestamp
- Year

- Bit Data Types

- Blob Data Types

- Geometric Data Types

- id VARCHAR(20) NOT NULL → Gbkh ad null  
NULL → Blh ad null

id  
0001  
0002

id  
1  
2  
3

- Default → Tambah nilai awal biar tidak null

- Copy tables → CREATE TABLE movie\_copy AS  
(SELECT id, name AS title FROM  
movies)  
→ INSERT INTO movies\_copy (id, name)  
SELECT id, name  
FROM movies