7. Integrity enforcement
Data must satisfy certain constraint/ conditions.

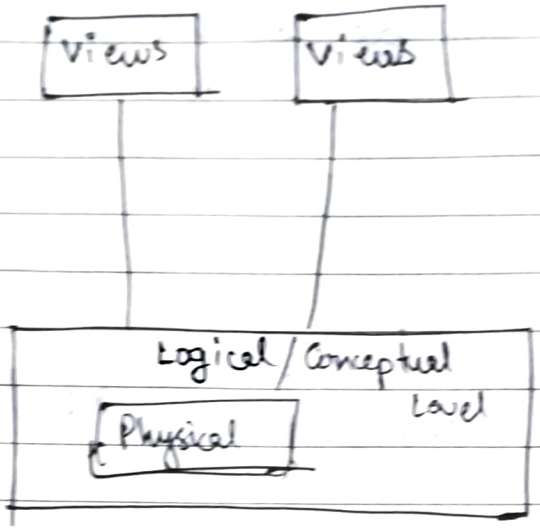
6/01/2025
Day 2

Data Abstraction

Database ⟶ No. of files in the disk.
Complex structure to keep the relations & support efficient access.

Complex internal representation to be hidden.

Physical level Defn: How data is actually stored.
↓
Conceptual / Logical Level: What Data is stored
↓
View Level: Centralised collection of Data.
All data is not avl. to every user.
Part of data is avl. to user. is the view for the user.



Data Independence

Physical Level Independence

⇓

Changes in the physical level defn. do not impact the application prog.

Conceptual level Independence (Difficult)

On the other hand, logical data independence is harder to achieve because it allows structural and constraint changes without affecting application programs—a much stricter requirement.

Can be changed but performed very rarely (infrequent)

Schema of a database

⇓

Structure of the database

Instance of a database

⇓

Content of the database at a point of time is the instance at that time (Temporal)

Data Model

There is a data model underlying the structure of a database

A set of tools to design data, their interactions semantics & constraints

Data Model

- Object based model
- Record based model

⇓

Database ⇒ Σ Files Σ ⇒ Collection

File ⇒ Σ Records

Record Based Model

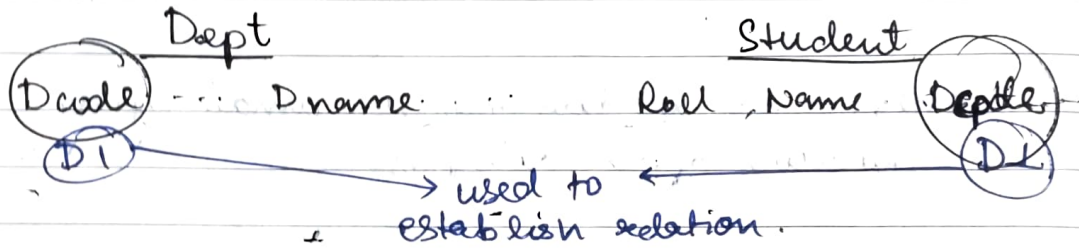
To establish rel. among records, value of common attribute is stored in the record.

Attr value → Relational model

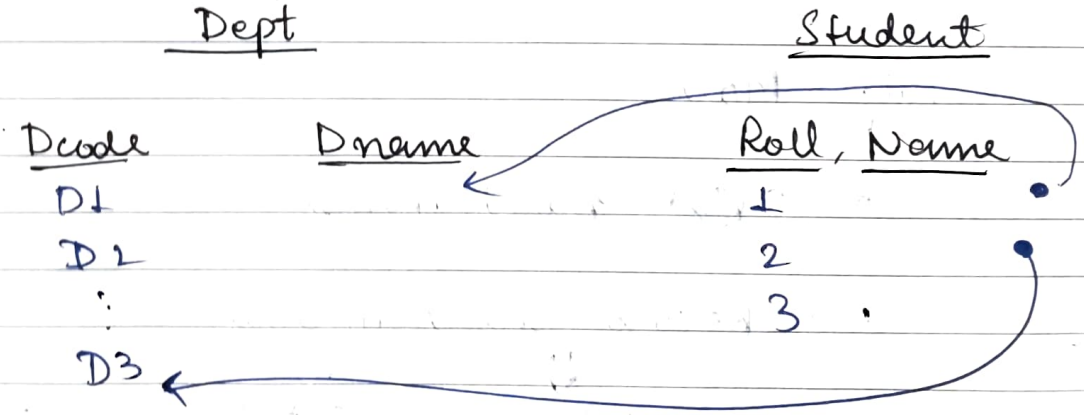
Record pointers → Network model

Record pointers → Hierarchial model

Eg of Relational Model



Network Model



To establish association b/w the records instead of keeping their attribute value in the record, a pointer to the corresponding record is stored.

Comparision b/w network & relational model

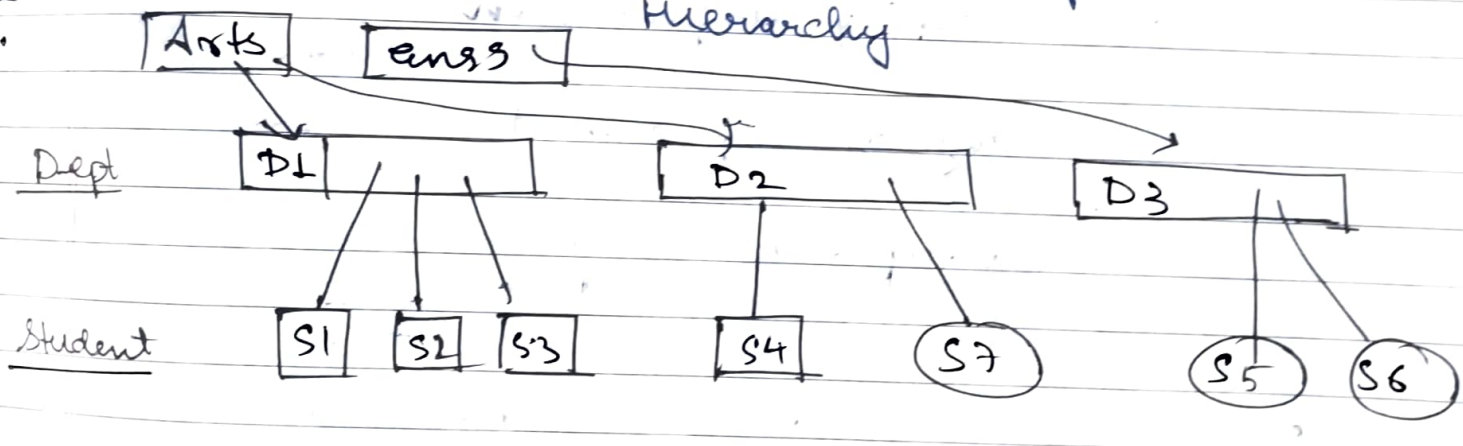
- Access to related record in network model maybe easier compared to relational model.
- If ~~a~~ related record, ~~no~~ moves to some other place in risk ⇒ no effect on relational model but overhead in network model.

In network model, lots of pointers may be there among records ⇒ Forest (Complicated structure)

⇓

Records are placed in Hierarchy.

Eg of Hierarchical model



Functional Components of DBMS

- Database manager
  S/w module that forms the core part of DBMS.
  - Interaction with file manager.
    ↓
    Part of OS.
  - To store/access data into/from database, database manager interacts with file manager
  - Handling concurrent access
  - Security enforcement
  - Integrity enforcement.
  - Backup & Recovery

- To interact with database we need some language.

  DDL → Data Defn. language.
  → specify schema of the database
  on compilation ⇒ DDL statement generates meta data (data about data) which are stored into data dictionary → Contains permission
  (Part of DBMS) → DBMS consults dict. to check for errors in command. (eg fetch ROLL but ROLL does not exist)

→ Data Manipulation Language (DML)

- To store data, to modify data
  - Retrieve Data
  - Delete Data

DML

Procedural
→ How & what needs to be defined

Non procedural
→ Only what needs to be specified.

SQL (Structured Query language)
⇓
Set of statements → DDL
→ DML

- Query Processor: User submits procedural DML statements. Preparing the efficient execution plan is done by query processor.

- DDL Compiler

DDL Statement —DDL compiler→ Metadata

- DML pre-compiler

SQL lacks procedural aspects

→ Procedural features provided by some High level language. (or PL/SQL)

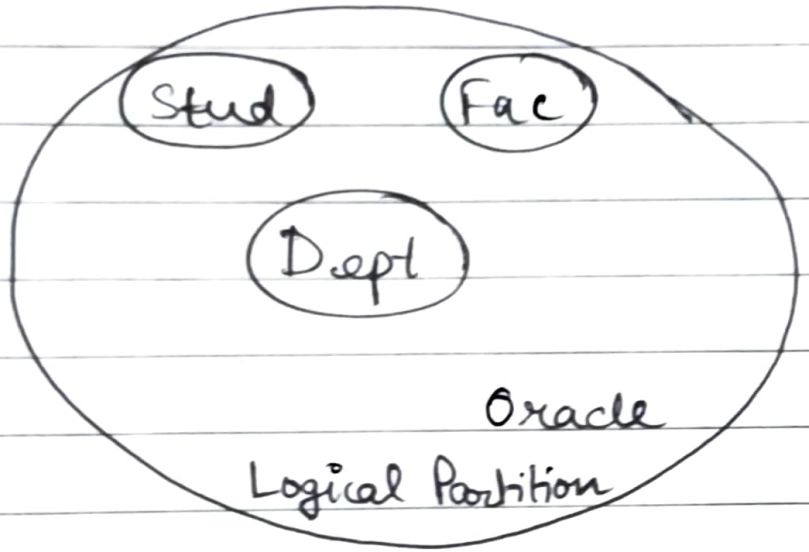
→ For efficient interaction with database, put DML statements in HLL code.
(embeded DML/SQL statements)

precompiler ⇓
DML ∧ Converts ~~embedded~~ embedded DML/SQL statements into equivalent Host lang code.

- Data Files
- ~~Data~~ Index Files
- Data Dict.

Types of users

Users
- → DBA (Database administrator)
  - To specify overall database structure & storage structure.
  - Granting/revoking system level privileges to users.
- → App. Prog: Develop app using HLL & DML Embedded SQL.
- → Sophisticated User: (DML)
- → Naive Users: Interact using UI.



Database $\Rightarrow \Sigma$ Tablespace (logical partition)

Tablespace is a collection of tables. One tablespace can be associated with many files but one file is associated with just one tablespace.

Overall Structure of DBMS (Simplified)

OVERALL STRUCTURE OF DBMS

(SIMPLIFIED)