



INTRODUCTION TO DATABASES AND TRANSACTION

INTRODUCTION

- **Database Management System**
- **DBMS Applications**
- **Purpose of Database Systems**
- **Data Abstraction**
- **View of Data**
- **Database Languages**
- **Database Architecture**
- **Database Users and Administrators**
- **Overall Structure**
- **History of Database Systems**



WHAT IS A DATABASE?

- It is a collection of related pieces of data.
- It represents and captures the information about real world enterprise or part of enterprise.
- Databases stores information to serve specific data management needs of enterprise.
- A collection of information organized in such a way that a computer program can quickly select desired pieces of data.
- **To access information from database, you need Database Management System (DBMS).**



DBMS & ITS APPLICATIONS

❖ **DBMS** contains collection of interrelated data and set of programs that allow user to access and modify these data.

- Banking: **all transactions**
- Airlines: **reservations, schedules**
- Universities: **registration, grades**
- Sales: **customers, products, purchases**
- Online retailers: **order tracking, customized recommendation**
- Manufacturing: **production, inventory, orders, supply chain**
- Human resources: **employee records, salaries, tax deductions**



PURPOSE OF DATABASE SYSTEMS

- To see why DBMS are necessary lets look at “file processing system”
- **Drawbacks** of using file systems to store data:
 - **Data redundancy and inconsistency**
 - Redundancy - duplication of information in different files
 - Inconsistency - e.g. in one file age of customer is shown 25 and in another file same customers name is shown 31.
 - **Difficulty in accessing data**
 - Need to write a new program to carry out each new task. E.g. find all customer with same postal code.
 - Difficult to write new application program.



- **Data isolation** — multiple files and formats
- **Integrity problems**
 - Data is required to satisfy the integrity constraints.
e.g. account balance should not be less than 1000.
 - Hard to add new constraints or change existing ones
- **Atomicity of updates**
 - Atomicity – transaction should happen completely or not at all.
 - Failure of updating process may leave database in an inconsistent state with partial updates carried out.
 - Example: Transfer of money from one account to another should either complete or not happen at all.



- **Concurrent access by multiple users**
 - Concurrent accessed needed for performance
 - Uncontrolled concurrent accesses can lead to inconsistencies.
e.g. Two people reading a balance and updating it at the same time
- **Security problems**
 - Hard to give access to data depending on the authority of user.
- **Theses problems lead to development of DBMS. Database Management Systems offer solutions to all the above problems.**



DATA ABSTRACTION

- ❑ Major purpose of database system are to provide users with an abstract (conceptual) view of data.
- ❑ The system hides certain details of how data is stored and maintained. Complexity should be hidden from database user.

LEVELS OF ABSTRACTION

○ Physical level:

- ❑ Describes **how** a data / records are stored.
- ❑ It is lowest level of abstraction.

○ Logical level:

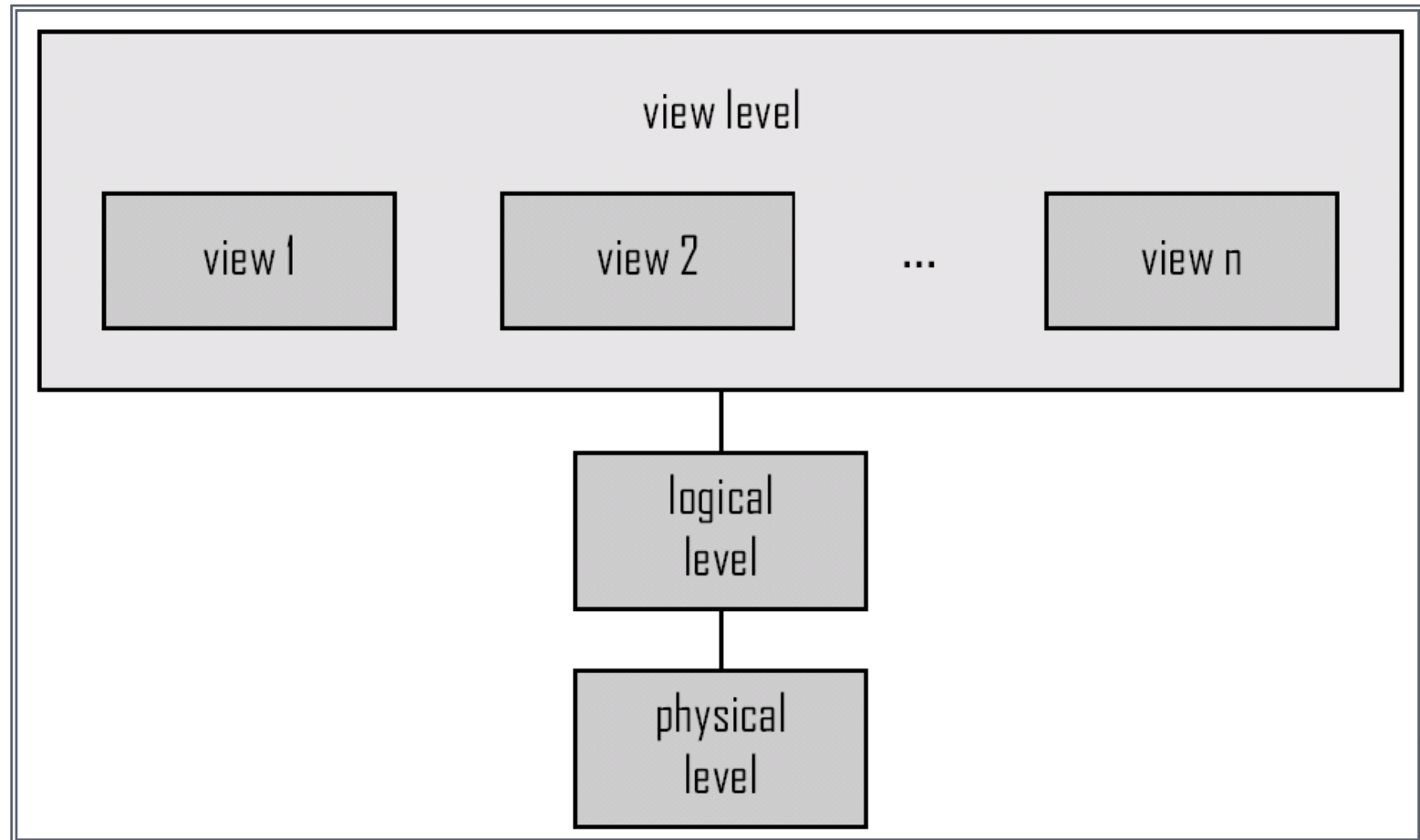
- ❑ Describes **what** data / records are stored in database.
- ❑ What relationship exist among those data
- ❑ It is next higher level of abstraction it is database administrator level.

○ View level:

- ❑ Highest level of abstraction.
- ❑ Describes part of database for particular group of users.
- ❑ There can be many different views of database depending on where in database user want to see the information.



VIEW OF DATA



DATABASE LANGUAGES

○ Data Definition Language (DDL) :

- It is a subset of SQL statements used for defining the objects in a db.
- It is used to create, modify and remove db or db objects.

○ Data Manipulation Language (DML) :

- It is a subset of SQL statements used to retrieve and manipulate data from the tables.
- It is used to insert, delete and update values in rows and columns.



DATA DEFINITION LANGUAGE (DDL) :

- Basic DDL commands are as follows :

I. **CREATE**

II. **ALTER**

III. **DROP**

CREATE

- **Create Database**

- E.g. **Create database College**
- Above statement will create database where you can store different tables.

- **Create Table**

- E.g. **Create table student(roll_no int, name char(10))**
- Above query will create table with two columns named as roll_no and name.



DATA DEFINITION LANGUAGE (DDL) :

ALTER

Alter statement allows to change structure of table without deleting or recreating it.

- ❑ Adding column to table
 - **Alter table student add percentage float**
- ❑ Removing column from table
 - **Alter table student drop column name**

DROP

Drop statement allows to delete db or table permanently.

- **Drop table student**
- **Drop database college**



DATA MANIPULATION LANGUAGE (DML) :

- ❑ Inserting records into table (INSERT)

insert into student values (11,'neha',67)

insert into student values (12,'priti',60)

- ❑ Selecting data from table (SELECT)

Select * from student

Select roll_no form student

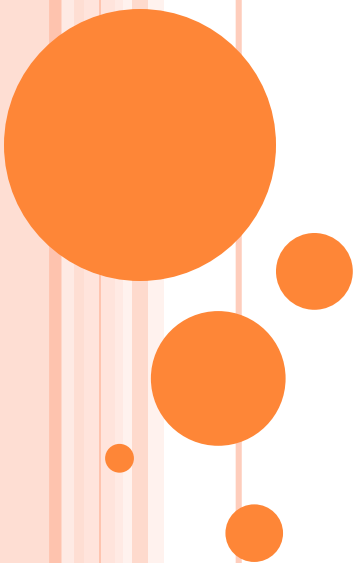
- ❑ Updating data of table (UPDATE)

Update student set percentage=78 where roll_no=11

- ❑ Deleting data from table (DELETE)

Delete from student where roll_no=11





END OF SESSION