

- In the conceptual level, internal details such as an implementation of the data structure are hidden.
 - Programmers and database administrator work at this level.

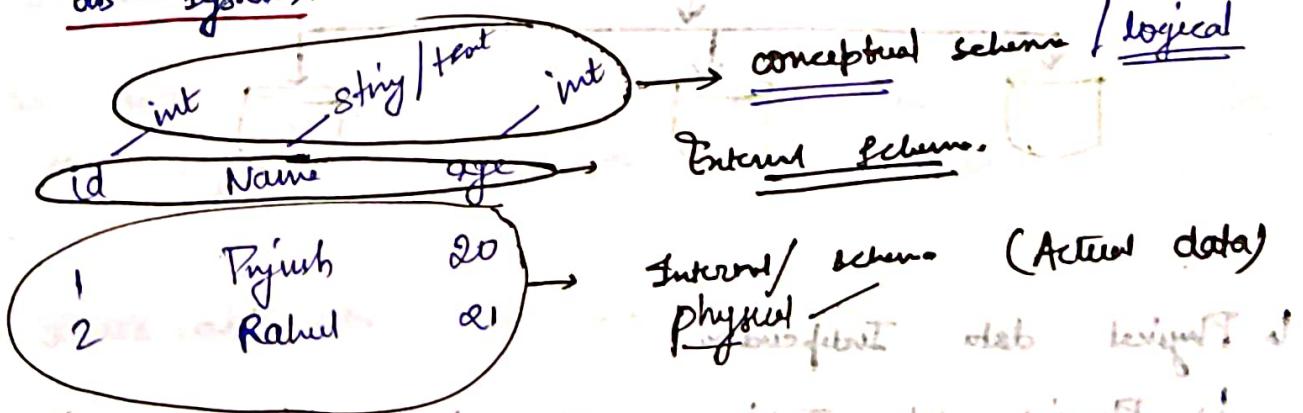
3. External level / view level

- ↳ At the external level, a DB contains several schemas that sometimes called a subschema. The subschema is used to describe diff. views of DB.

Emp No. Name Address Phone No. Dept. Design. Off.

- In each view scheme describes the DB part that a particular user group is interested and uses the remaining DB from the user group.

- L The view scheme describes the end user interaction with db systems. / logical



abs do abs abs ad abs abs abs abs abs abs abs

Data Independence

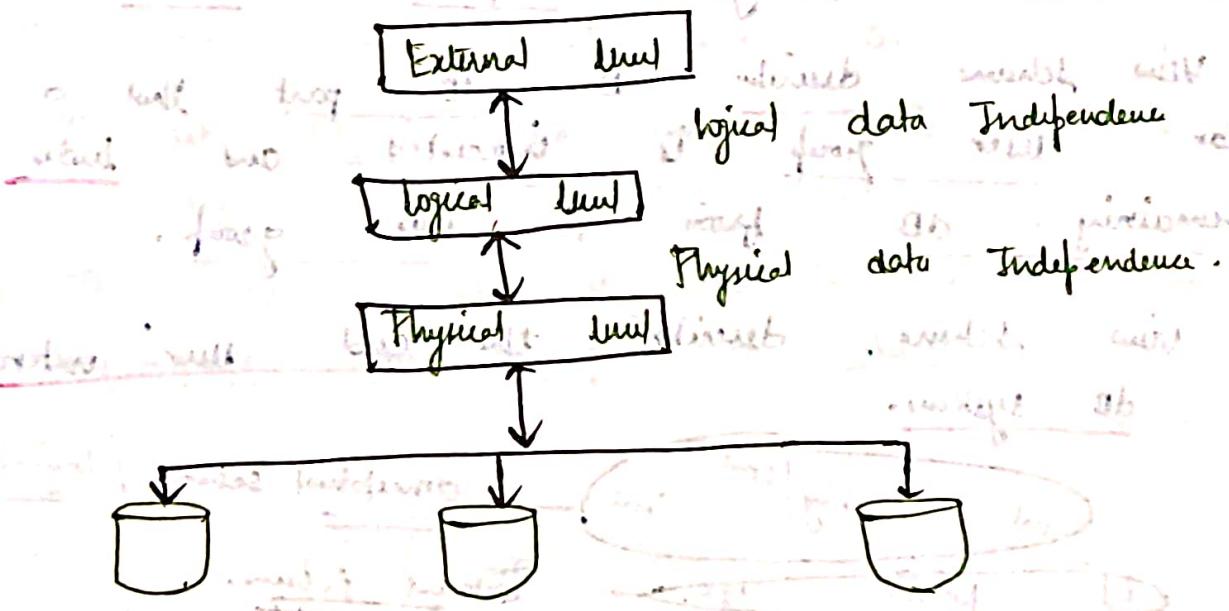
↳ The ability to modify a schema definition in one level without affecting a schema definition in the next higher level is called Data Independence.

↳ Data Independence is one of the main Advantages of DBMS.

↳ It is of two types:

(i) Physical data Independence

(ii) Logical data Independence



↳ Physical data Independence:

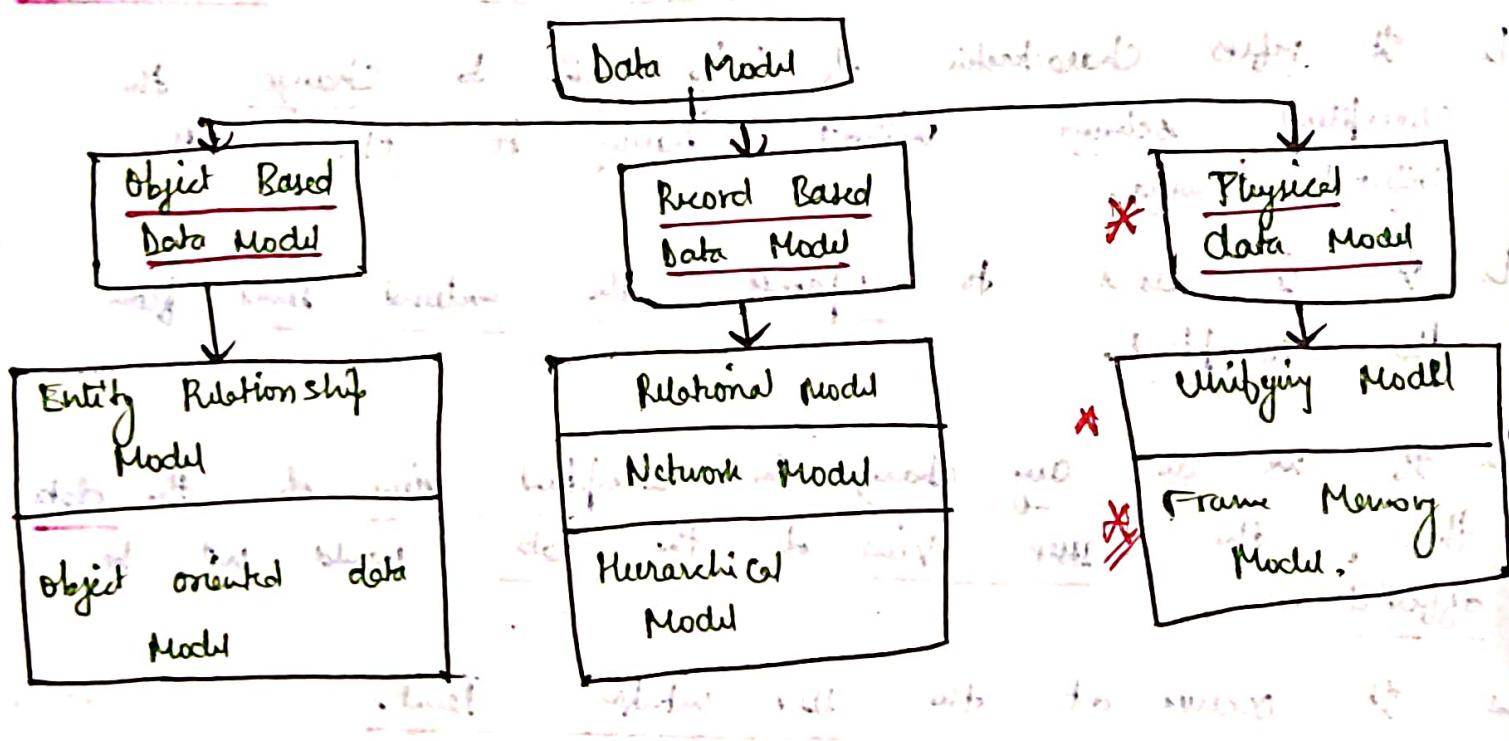
↳ Physical data Independence can be defined as the capacity to change the natural schema without having to change the conceptual schema.

↳ If we do any change in the storage size of the database system server, then the conceptual structure of the DB will not be affected.

- It is the ability to modify the physical scheme without carrying out application program to be rewritten.
 - It is used to separate conceptual level from the internal level.
 - It occurs at the logical interface level.
 - It is used with the help of odd & to represent data in a structured sufficient data level.
 - Logical data Independence
 - It is the ability to modify the conceptual schema without carrying app. program to be rewritten.
 - It refers characteristic of being able to change the conceptual schema without having to change the external schema.
 - It is used to separate the external level from the conceptual level.
 - If we do any changes in conceptual view of the data then the user view of the data would not be affected.
 - It occurs at the user interface level.

Data Models

- ↳ Data Model is the modeling of the data description, data semantics and consistency constraints of the data.
- ↳ Data Model Provide the conceptual tool for describing the design of a db at each level of db abstraction.
- ↳ It is also be defined as the collection of high level data description constructs to hide many low level storage details.
- ↳ There ~~are~~ are mainly 3 three types of Data model.



① Object Based data Model

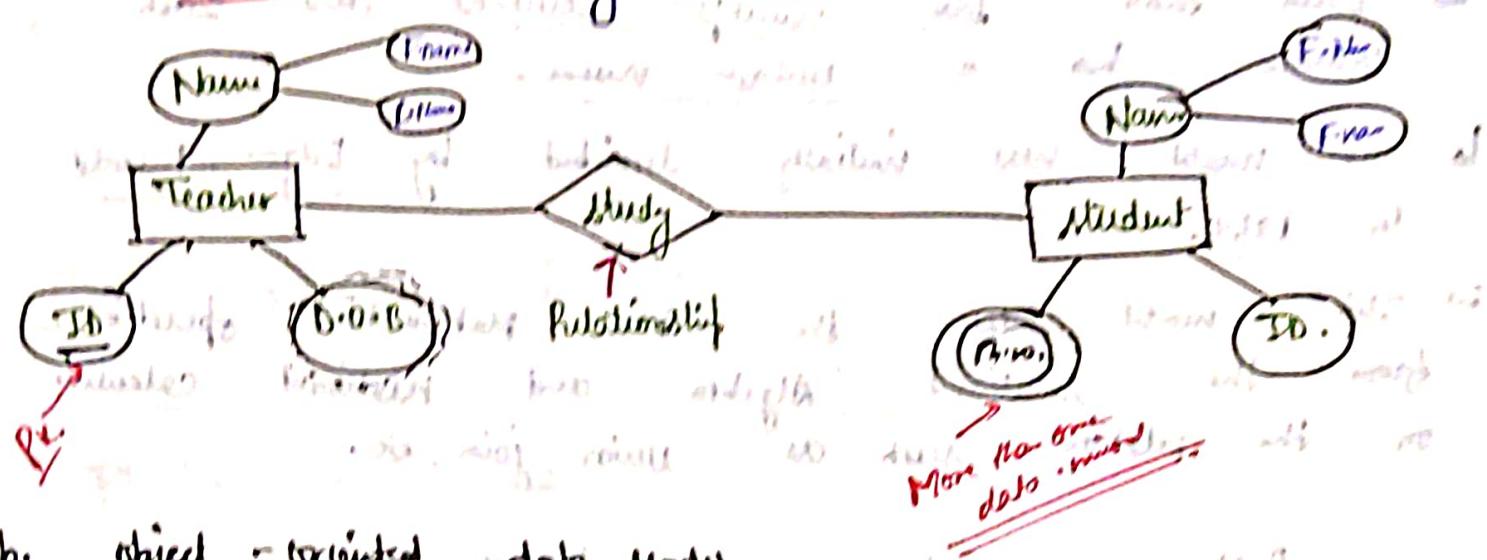
- ↳ It is used to describe the data at the logical and view level.
- ↳ It provide flexible structure and structuring capability and allow to specify data constraints.
- ↳ There are mainly two types of Object Based data model.

a. Entity Relationship Model:

↳ An ER model is the logical representation of data.

Object and relationship among them.

↳ These objects are known as entities and relationship is an association among these entities.



b. Object-oriented data Model

↳ In this, information or data is displayed as an object and these objects store the value in the instance variable.

↳ In this model object-oriented programming with messages are used.

↳ This model work with object-oriented programming language like Python, Java, VB.net and Perl, etc. It was constructed in 1980s.

v) Record Based data Model (RDBM)

↳ It is used to describe data at logical and view level.

↳ This data model is used to specify the overall logical structure and to specify the higher level structure and provide higher level description.

↳ There are three types of dataBased Model

(a) Relational data Model

↳ This type of Model designs the data in the form of rows and columns within a table.

↳ Each table has multiple columns and each column has a unique name.

↳ This model was initially described by Edgar F Codd in 1969.

↳ This model uses the certain mathematical operations from the relational Algebra and relational calculus on the relation such as union, join, etc.

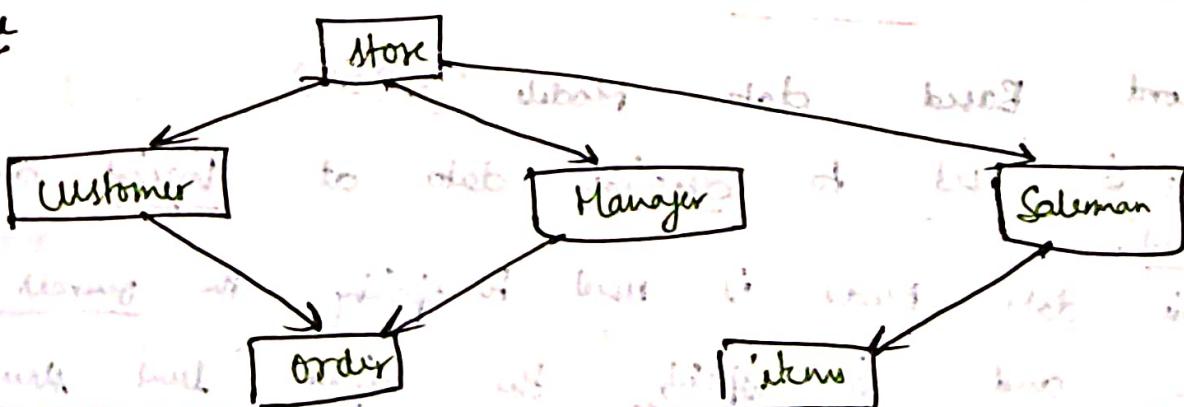
Roll Name Address

(b) Network data Model

↳ In this, data is organised into graph And it can have more than one parent node.

↳ It permits the modify of many to many Relationship in data

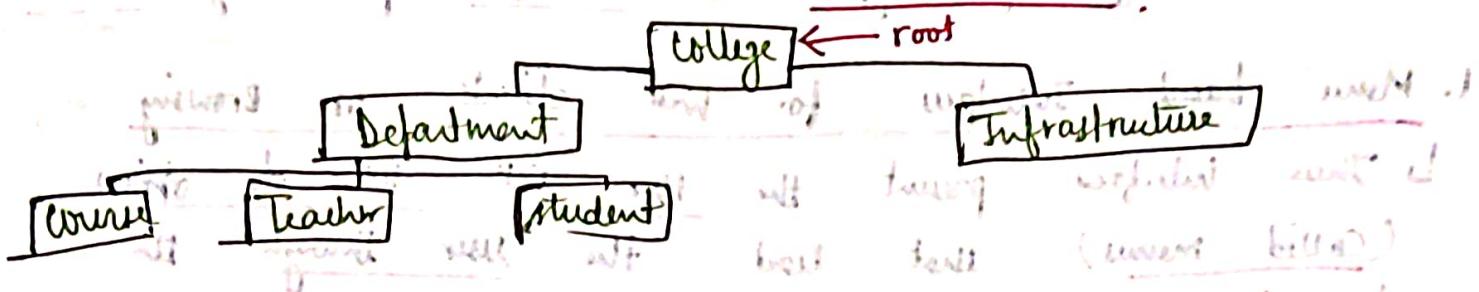
Database



(C) Hierarchical data Model

↳ The Hierarchical data model organises data in a tree structure.

In this model, each entity has only one parent and many abstract children. There is only one entity for this model that we call root.



(iii) Physical data Model

In this data Model it's used to describe the data at low level.

DBMS Interface.

- ↳ It is a user interface that allow for the ability to input query to a DB without using the query language itself.
- ↳ There are following types of Interface provided by a DBMS.
 - I. Menu based Interface for web clients or browsing
 - ↳ These interfaces present the user with list of options (Called menus) that lead the user through the formation of the request.
 - ↳ Basic Advantage of using menu is that they remove the tension of remembering specific commands and syntax of any query language.
- II. Form Based Interface
 - ↳ A Form based interfaces display a form to each user.
 - ↳ Users can fill out all of the form entries to insert new data or they fill out only certain entries in which case the DBMS will return matching data for the remaining entries.
 - ↳ Many DBMS form specialization language, special language that help programmers specify such forms.

- 3 Graphical User Interface (GUI) \leftrightarrow DBMS: all of methods
- ↳ It typically displays a schema to the user in diagrammatic form.
 - ↳ The user can specify a Query by Manipulating the diagram.
 - ↳ In many cases we utilize both menu and forms.
 - ↳ Most users like to use a pointing device such as mouse, to pick a certain part of the displayed schema diagram.

4. Natural Language Interface (NLI)

- ↳ These interface accept requests written in english or some other language and attempt to understand them.
- ↳ A natural language interface usually has its own schema which is similar to the DB conceptual schema.

5. Interface for Parametric users

- ↳ Parametric users such as bank tellers, often have a small set of operations that they must perform repeatedly system analysts and programmers design and implement a special interface for a known class of naive users.

b) Interface for the DBA \rightarrow (Database) Administrator

- ↳ Most DB system contains privileged commands that can be used only by the DBA's staff.
 - ↳ These include commands for creating accounts, altering parameters etc.
- ↳ Oracle uses SQL (Structured Query Language) which specifies how data is stored and retrieved from the database. It consists of queries and commands. A query is a statement that retrieves data from the database, while a command is a statement that manipulates the database. Oracle also supports PL/SQL which is a programming language for performing complex operations on the database.
- ↳ Oracle provides a graphical user interface called Oracle Database Control which allows users to manage their databases. It includes tools for monitoring performance, managing storage, and performing backups and restores. Oracle also provides a command-line interface called SQL*Plus which allows users to execute SQL queries and commands directly.
- ↳ Oracle also provides a web-based interface called Oracle Database Express Edition which allows users to access their databases via a web browser. This interface provides a simplified way to manage databases without needing to learn the complexities of SQL or PL/SQL.
- ↳ Oracle also provides a mobile application called Oracle Database Mobile which allows users to access their databases from their mobile devices. This application provides a simplified way to manage databases on the go.
- ↳ Oracle also provides a cloud-based interface called Oracle Database Cloud Service which allows users to manage their databases in the cloud. This interface provides a simplified way to manage databases in the cloud without needing to worry about infrastructure management.

DBMS language

- ↳ A DBMS has an appropriate language and interface to express an query and update.
- ↳ DB language can be used to read, store and update the data in the DB.
- ↳ Types of DB languages of DB:
 - DDL
 - DML
 - DCL
 - TCL
- ① DDL - data definition language
 - ↳ It is used to define db structure or pattern.
 - ↳ It is a set of SQL commands used to create, modify and delete [db structure] but not data.
 - ↳ It is used to create scheme, table, index, constraints, etc. in the db.
 - ↳ These commands are normally not used by a general user, who should be Accessing the db via an application.
 - ↳ They are normally used by the DBA to a limited extent, a db designer or Application developer.
 - ↳ DDL update a special set of tables called the data dictionary or data directory.

A list of tasks that come under DDL

CREATE - Used to create objects in the db.

ALTER - Used to Alter the structure of the db.

DROP - Used to delete the object from the db

TRUNCATE - Used to Remove all records from a table, including all space allocated for the records are removed.

COMMENT - Used to add comments to the data dictionary

RENAME - Used to rename an object

2 DML - Data Manipulation Language

↳ It is a set of SQL commands used to select, modify and delete data in db not db structure.

↳ It is used for accessing and manipulating data in a db. It handles user requests.

↳ DML statements are used to manage data within scheme objects.

List of tasks that come under DML:

SELECT : It retrieves data from a db.

INSERT : It inserts data into a table.

UPDATE : It updates existing data within a table.

DELETE : It delete all records from a table

MERGE : It performs UPSERT operation such as insert or update operation.

CALL: It is used to call a structured Query language or a Java subprogram.

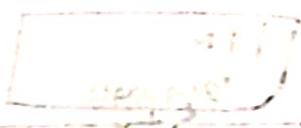
LOCK TABLE: It - controls concurrency.
It blocks multiple users from accessing the DB.

3. DCL - Data control language at not SQL language.

↳ The data control language is used to control privilege in DB.

↳ It is the component of SQL statement that control access to data and to the DB.

↳ To perform any operation in the DB, such as for creating table, sequence or view we need privileges.



* List of tasks that come under DCL.

↳ privileges are of two types.

• SYSTEM - creating a session (DDL), table etc or all privileges.

• OBJECT - Any command or Query to work on tables comes under object privilege.

* List of tasks that come under DCL.

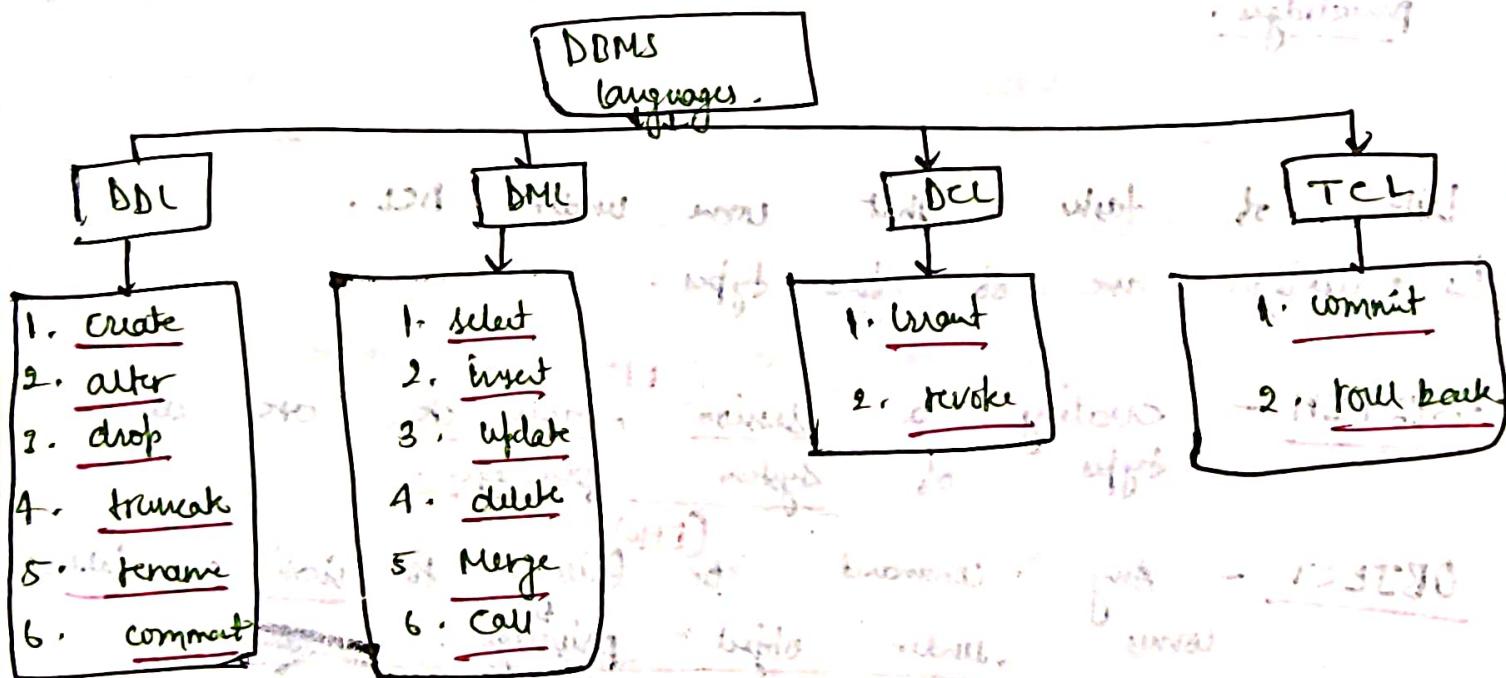
* Grant: It gives user access privilege to a DB.

* Revoke - It takes back permission from the user.
It is used to remove privilege and is of general nature.

4. TCL - Transaction control language - It is used to run the changes made by the DML statement.

↳ TCL can be grouped into a logical transaction.
↳ It is a group of tasks that comes under TCL.
Commit - It is used to save the transaction of dB.

Rollback - It is used to restore the dB to original since the last commit.



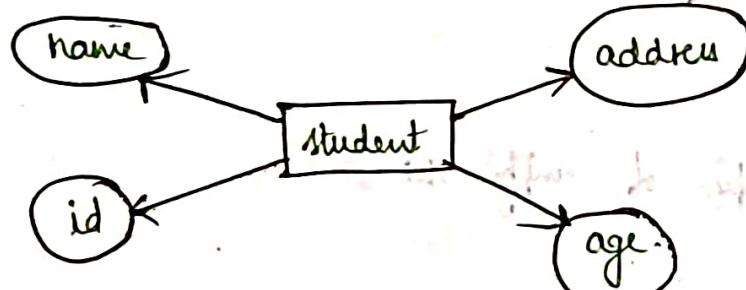
Note:-

* DQL - Data Query language tells what to tell the system.

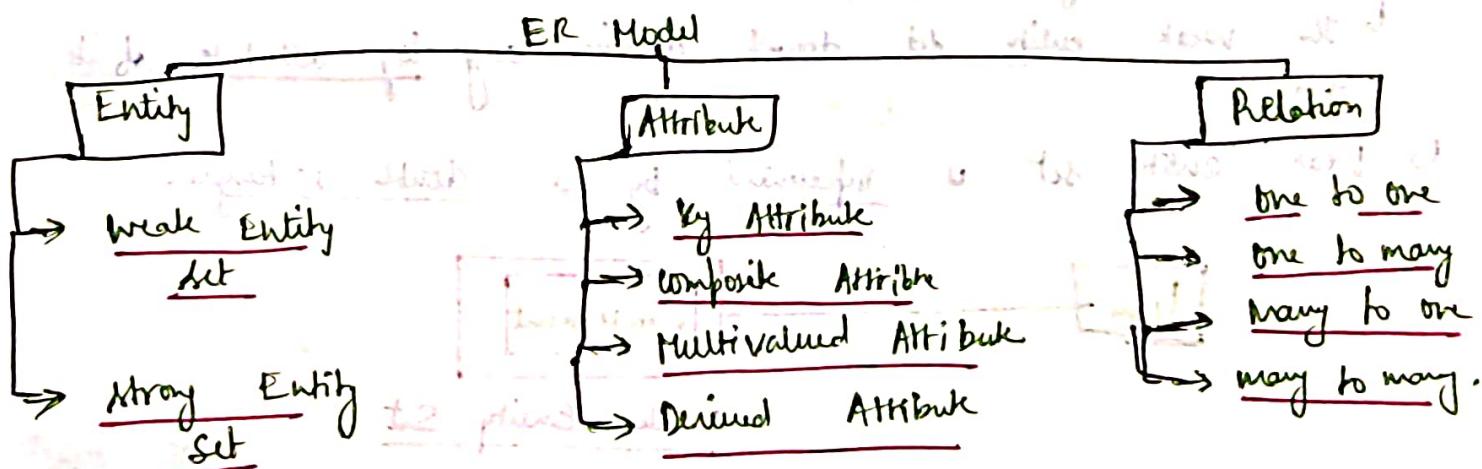
only one command.
↳ select.

ER Model concept

- ↳ Then entity relationship (ER) model is a high-level data model.
- ↳ It is based on a perception of a real world that consists of a collection of entities and of relationships among these objects.
- ↳ It depicts a conceptual design for the DB. (It also depicts a very simple and easy to design view of data.)
- For example - Suppose we design a school DB. In this DB, the student will be an entity with attribute like address, name, id, age, etc.



Concept component of ER diagram:



Entity :

↳ Entity

↳ Entity Unit 3

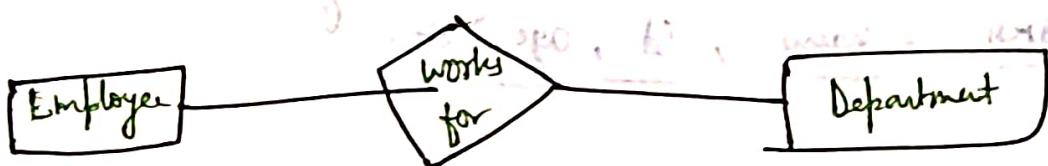
↳ Entity is anything or object in the real world that is distinguishable from all other objects.

↳ Anything about which we store information is called an entity.

Entity set :

↳ It is a set of entities of the same type that share the same properties or Attributes.

↳ An entity set can be represented as Rectangles.



Type of entity set :

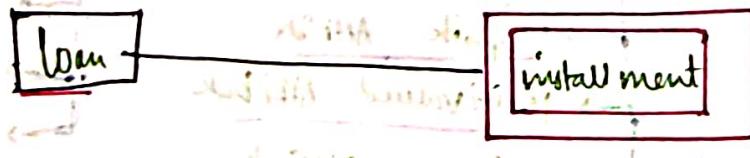
↳ There are two types of entity set.

(1) weak entity set

↳ An entity that depends on another entity called a weak entity set.

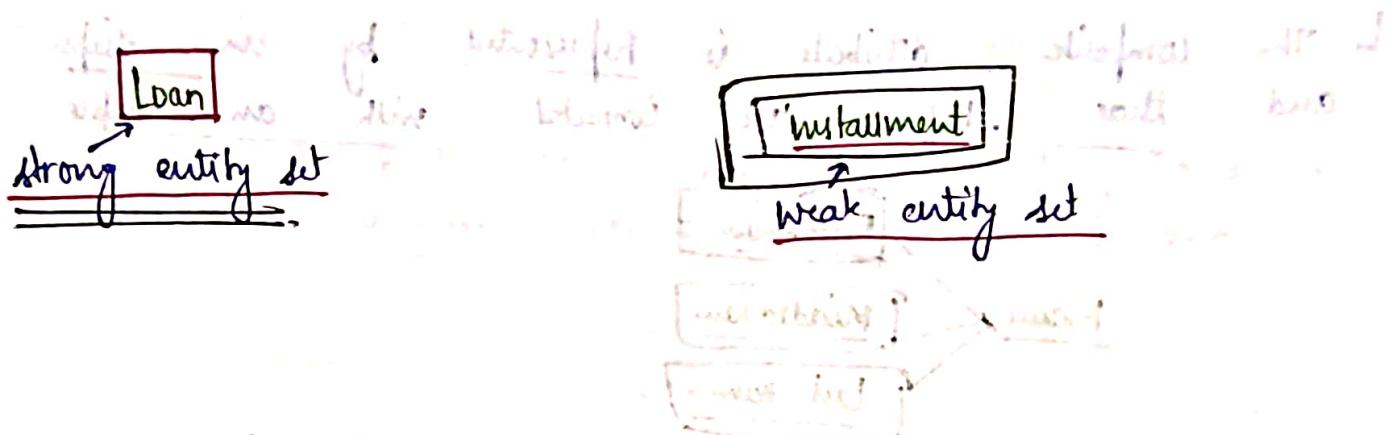
↳ The weak entity set doesn't contain any key attribute of its own.

↳ Weak entity set is represented by a double rectangle.



weak Entity Set

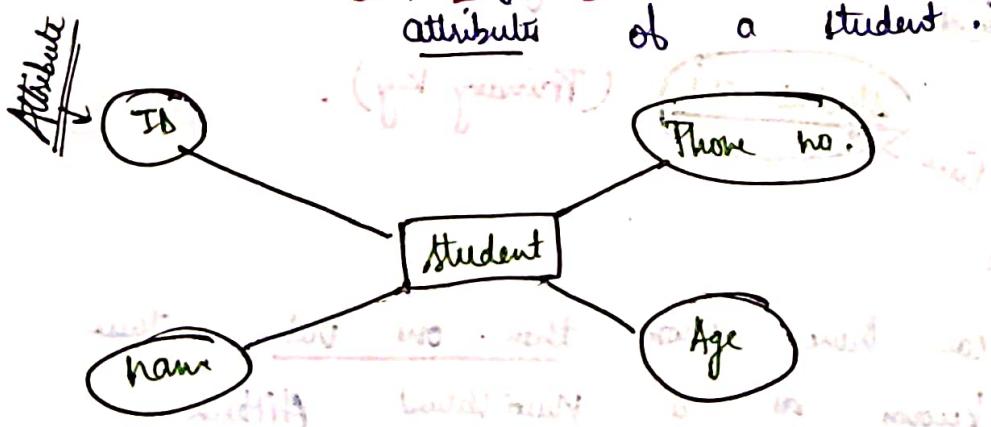
- (2) Strong entity set
- ↳ A strong entity set is an entity set that contains sufficient attributes to uniquely identify all its entities.
 - ↳ Primary key exists for a strong entity set.
 - ↳ Single rectangle is used to representing a strong entity set.



* Attributes in ER Model

- ↳ The attributes is used to describe the property of an entity.
- ↳ An entity set may contain any number of attributes.
- ↳ Attributes are represented in an Elliptical shape.

For example - ID, Age, Contact number, Name, etc. can be attributes of a student.



Type of Attribute

a) Simple Attribute

- ↳ An Attribute that cannot be further subdivided into components is a simple Attribute.

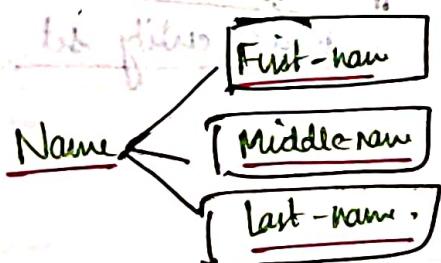
b) It is represented by ellipse.

e.g.; The Roll no. of student is, the Id no. of an employee.

b) Composite Attribute.

An Attribute that can be split into components is a Composite Attribute.

The composite attribute is represented by an ellipse and those ellipses are connected with an ellipse.

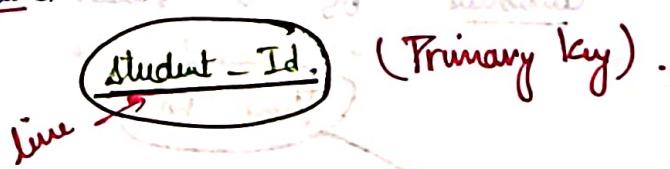


c) Key Attribute

The key Attribute is used to represent the main characteristics of an entity.

Is it represent a Primary key.

The key Attribute is represented by an ellipse with the next underlined attribute to do student.



(Primary Key).

d) MultiValued Attribute

An Attribute can have more than one value. These MultiValued Attribute.

The double ellipse is used to represent multivalued attribute.

For example: A student can have more than one phone number.

Phone no.

c) Derived Attribute

Is An Attribute that can be derived from other Attribute

is known as derived Attribute.

It can be represented by a dashed ellipse.

For example: A person's age changes over time and can be derived from another Attribute like date of birth.

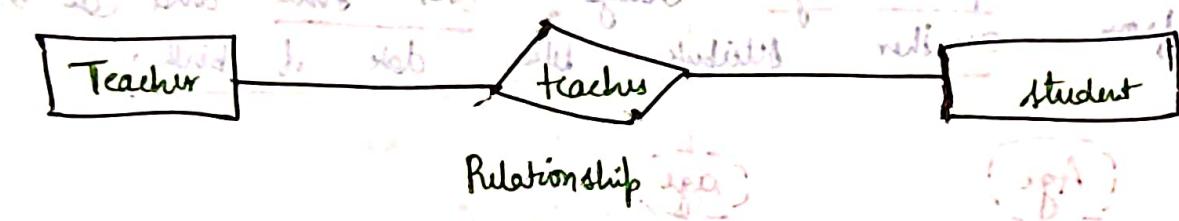
{Age}

{age}



Relationship / Mapping Constraints

- ↳ A Relationship is used to describe the Relation b/w entities.
- ↳ Diamond or Rhombus Box is used to represent the relationship.
- ↳ Mapping Cardinalities or Cardinality ratios express the no. of entities to which another entity can be associated via a relationship.



There are four types of mapping constraints or relationship.

(i) One to one Relationship

↳ When only one instance of an entity is associated with the relationship then its known as one-one Relationship.

For example - A Female can marry to one male (Vice Versa)



(ii) One to many Relationship

↳ When only one instance of the entity on the left and more than one instance of an entity on the right associates with the relationship then this is