

Q8] What is DBMS? Explain its advantages.

- A8-i) A database management system (DBMS) is a collection of programs that manages the database structure & controls access to the data stored in the database.
- ii) The DBMS serves as the intermediary between the user & the database. The database structure itself is stored as a collection of files. So, we can access the data in those files through the DBMS.
- iii) The DBMS receives all application requests and translates them into the complex operations required to fulfill those requests. The DBMS hides much of the databases internal complexity from the application programs and users.
- v) The purpose of a database is to store and retrieve information in a way that is accurate and effective and to manage the different database it contains (performance, security, availability, integrity etc).

* Advantages of DBMS -

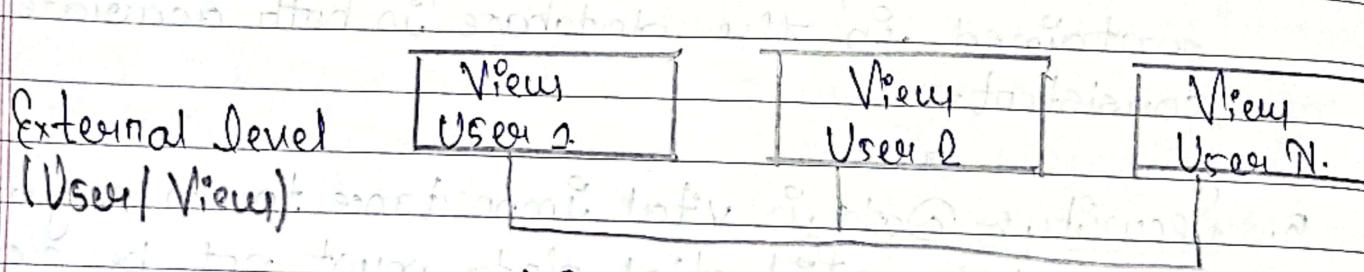
- i) Reduction of Redundancies:- Centralized data control of data by the DBA avoids unnecessary duplication of data & effectively reduces the total amount of data storage required.
- ii) Elimination of Inconsistencies:- The main advantage of avoiding duplication is the elimination of inconsistency that tends to be present in redundant data files. Any redundancy that exists in the DBMS are controlled & the system ensures that these multiple copies are consistent.

- iii. Shared data & A database allow the sharing of data under its control by any number of application programs or user. For example, the application from the public sector organizations and payroll departments can share the same data.
 - iv. Integrity & Centralized control can also ensure that adequate checks are incorporated in the dbms to provide data integrity. Data integrity means that the data contained in the database is both accurate & consistent.
 - v. Security & Data is vital importance to an organization & may be confidential data must not be accessed by unauthorized persons. Different levels of security could be implemented for various types of data & operations.
 - vi. Data Independence:- The ability to modify a scheme definition in one level without affecting a scheme definition in the next higher level is called data independence.
- Data Independence is usually considered 2 points of view.
- i. Physical data independence:- It allows changes in the physical storage devices or organization of the files to made without requiring changes in the conceptual view.
 - ii. Logical data independence:- Logical data independence indicates that, the conceptual schema can be changed without affecting the existing external schemas.

Q. What is Data Abstraction? Explain its levels.

Ans:- Data Abstraction & database systems are made-up of complex data structures. To ease the user interaction with database, the developers hide internal irrelevant details from user is called data abstraction.

* The 3 levels of abstraction are as



External Level
(User View)

View
User 1

View
User 2

View
User N.

Conceptual View.

Internal Level
(Physical)

Internal view.

Mapping supplied by DBMS.

i. **User View / External level:** This is the highest level of abstraction as seen by a user. this level of abstraction describes only the part of entire database which exists to simplify the interaction with the system.

ii. **Logical / Conceptual level:** The logical level of data abstraction defines what data are actually stored in the database & what relationships exist among those data. In relational dbms, the conceptual schema describes all relati-

ons that are stored in the database.

III. Physical / Internal Level: The physical level of abstraction is the lowest level of abstraction that describes how the data is actually stored. The physical level or internal schema, which contains the definition of the stored record, the method of representing the data fields, express the internal view and the access aids used.

Q. Who is Database Administrator? Explain the various function of DBA.

Ans-

Database Administrator: One of the main reasons of using dbms is to have a central control of both data and the programs accessing those data. A person who has such control over the system is called a Database Administrator (DBA).

- * The following are the functions of a database Administrator.
- 1. Schema Definition: The database administrator creates the database schema by executing DDL statements. Schema includes the logical structure of database table (relation) like data type of Attributes, Length of Attributes, integrity constraints etc.

2. Storage Structure & access method Definitions: Database tables or indexes are stored in flat files, heaps, BT trees etc.

3. Schema and physical Organization Modification: The DBA carries out changes to the existing schema and physical organization.

4. Giving Authorization for data modification: The DBA provides different access rights to the users according to their level. Ordinary users might have higher restricted access to data, while going up in the hierarchy to the administrator, you will get more access rights.

5. Routine Maintenance for some of the activities of a DBA are given below:

- Taking backup of database periodically.
- Ensuring enough disk space is available all the time.
- Monitoring jobs running on the database.
- Performance tuning.
- Ensure that performance is not degraded by some expensive task submitted by some user.

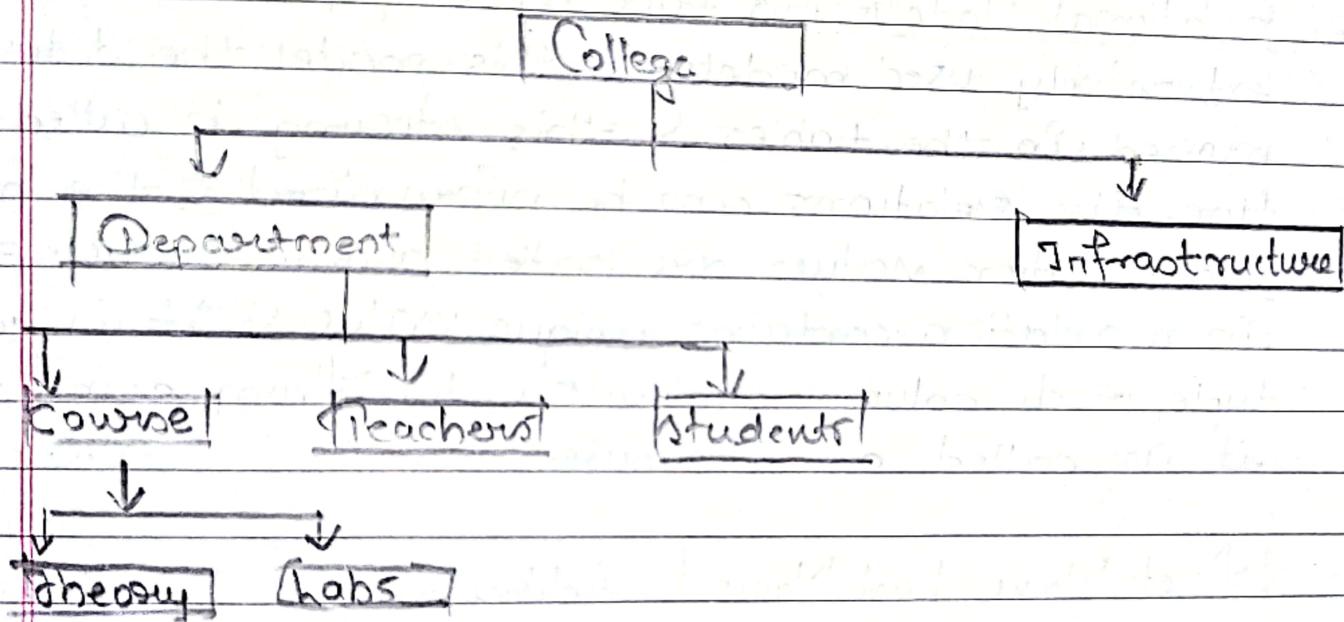
4. Why data models are used in DBMS database. Explain its components.

Ans- Data model is a logical structure of database. It describes the database to reflect entities, attributes, relationships among data, constraints etc.

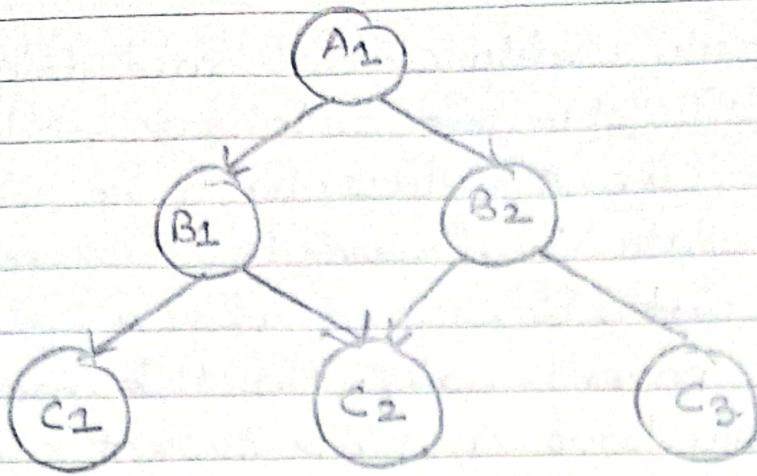
Data model can be defined as an integrated collection of concepts, for describing and manipulating data, relationships between data and constraints on the data in an organization. The purpose of a data model is to represent data & to make the data understandable.

There are 3 mostly widely record based data models are:

1. Hierarchical Model & This database model organises data into a tree-like structure with a single root to which all other data is linked. the hierarchy starts from the root data. & expands like a tree, adding child nodes to the parent nodes. In this model a child node will only have a single parent node. This data is organised into tree-like structure with one-to-many relationships between 2 different types of data.



2. Network Model & In the network data model, data model data are represented by collections of records. Relationships among data are represented by links. In this data model, graph data structure is used. It permits a record to have more than one parent.



3. Relational Model: This type model popular & the most extensively used models. In this model the data can be stored in the tables & this storing is called as relation, the relations can be normalized & the normalized relation values are called atomic values. Each row in a relation contains unique value & it is called as tuple, each column contains value from some domain it is called as attribute.

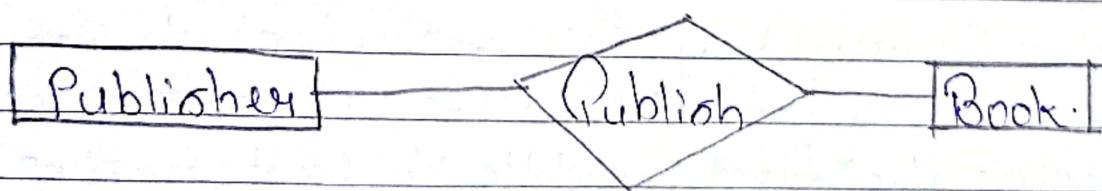
First Name	Last Name	Address	City	Age
Bhavana	Prajapati	6002, Kandivali (E)	Mumbai	21
Vanshu	Patel	Shivaji Nagar	Pune	20

Ques 7 Define - Entity, attribute, relationship, tuple, degree, cardinality.

Ans: Entity - An Entity is a person, place, thing or event about which the data are to be collected and stored. An entity is the fundamental item in any data model as it is distinguishable. i.e. each entity must be unique and distinct. For e.g. student, customer etc.

2. Attribute - An attribute is the characteristic of any entity. For e.g. customer entity can be described by attribute such as name, phone, address, gender. Each attribute is associated with a set of values called domain. Attribute are the properties which define a relation.

3. Relationship - A relationship describes an association among entities. For e.g. relationship exists between publisher and book can be described as: Many books are published by a publisher.



4. Tuple - It is nothing but a single row of a table, which contains a single record.

5. Degree - The total number of attributes which in the relation is called the degree of the relation.

6. Cardinality - Total no. of rows present in the table.

(Q 8-7) Write a note on following.

1. Primary Key :-

- It is a column or group of columns in a table that uniquely identify every row in that table.
- The primary key must contain unique values.
- A primary key column cannot have null values.
- A table can ^{have} only one primary key which may consists of single or multiple fields.
- The value in a primary key column can never be modified or updated if any foreign key refers to that primary key.

Ex :- student Id is a primary key.

2. Alternative Key :-

- Alternative keys is a column or group of columns in a table that uniquely identify every row in that table.
- A table can have multiple choices for a primary key but only ~~one~~ one can be set as the primary key.
- All the keys which are not primary key are called as Alternative key.

Ex :- the primary key for a student table may be the student id. the alternative key might combine the first, middle & last names of the student.

3. Candidate Key :-

Def :- It is a set of attributes that uniquely identify tuples in a table. Candidate key is a super key with no repeated attributes.

- The primary key should be selected from the candidate keys. Every table must have at least a single candidate key.
- A table can have multiple candidate keys but only a single primary key.

Ex:- Student Id, Email Id, Roll no are Candidate key which help us to uniquely identify.

4. Attribute & its Types:-

Ans:- Attribute:- It is define what type of data is stored in database table.

* There are 6 Types of Attributes:-

1. Simple Attributes:- simple attributes are those which can not be divided further.

2. Composite Attributes:- It means all those attributes which are composed of many other simple attribute.

Ex:- first, middle & last name.

3. Single Valued Attributes:- It means all those attributes which can take only one value for a given entity from an entity set.

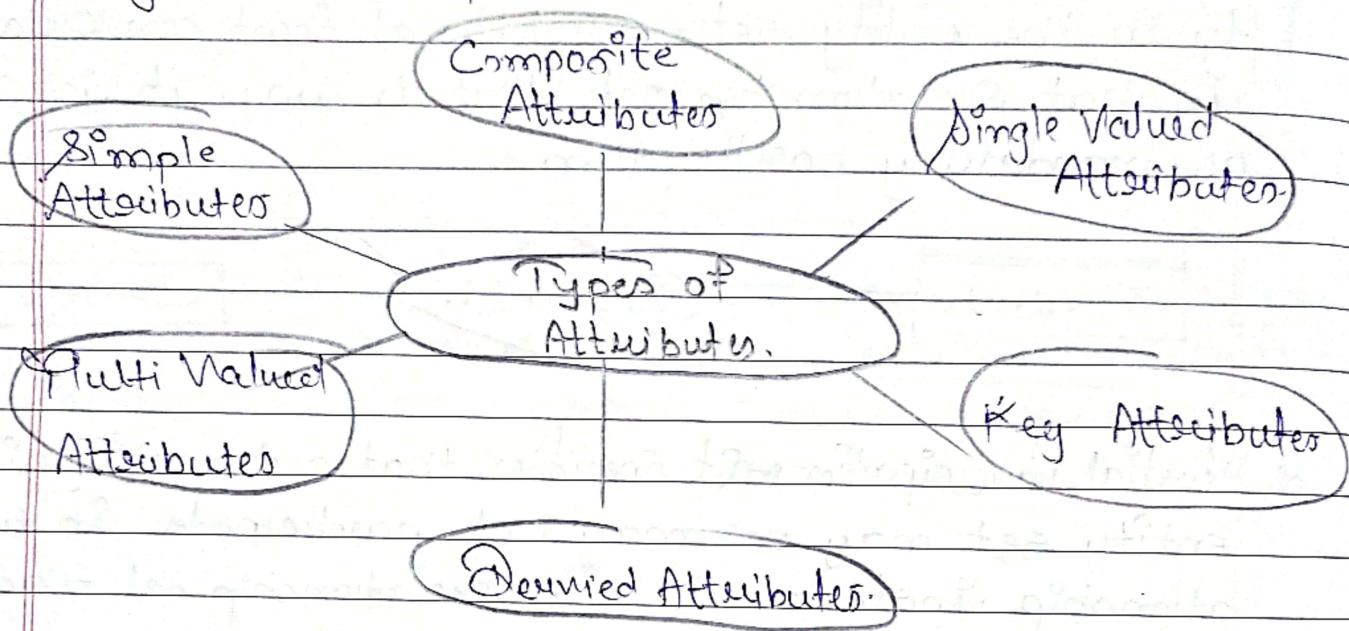
Derived.

4. Multi Valued Attributes:- It means all those attributes which can be derived from other attributes.

5. Key Attributes:- It means all those attributes which can identify an entity uniquely in an entity set.

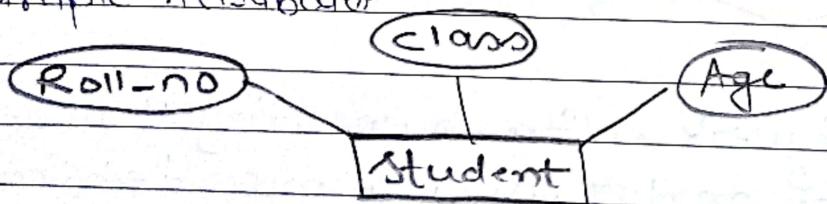
6. Multi-Derived Attributes :- It means those attributes which can take more than one value for a given entity from an entity set.

* Diagram all Types of Attributes.

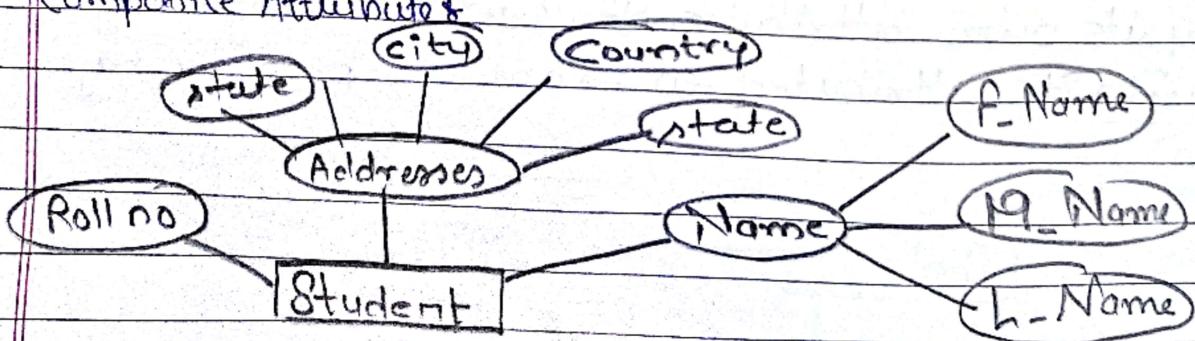


Diagrams :-

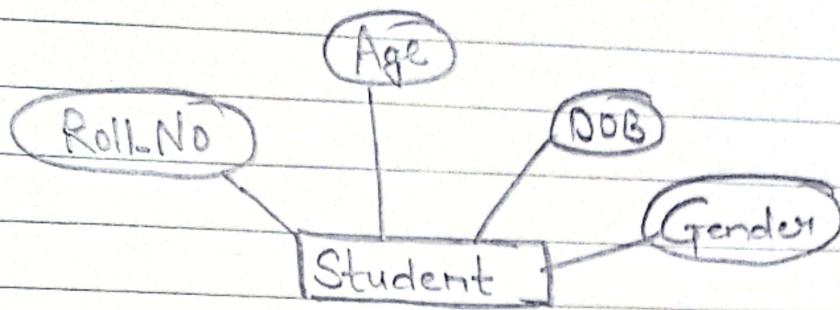
i) Simple Attributes



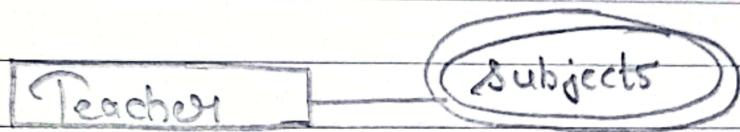
ii) Composite Attribute



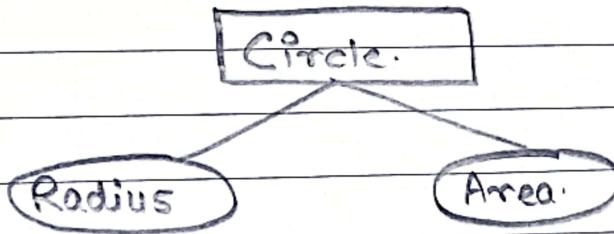
3. Single Valued Attributes:-



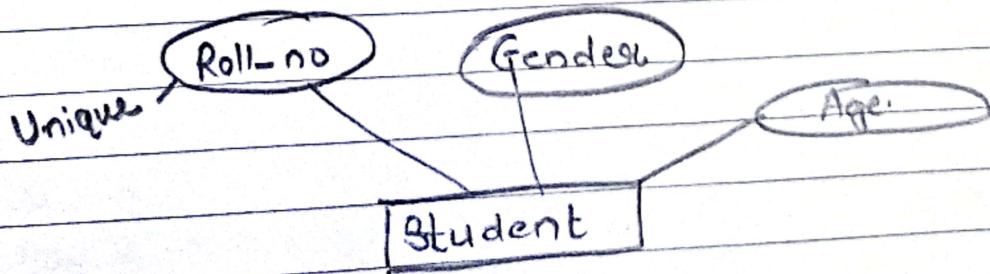
4. Multi-Valued Attributes:-



5. Derived Attributes:-



6. Key Attributes:-



5. Strong Entity.

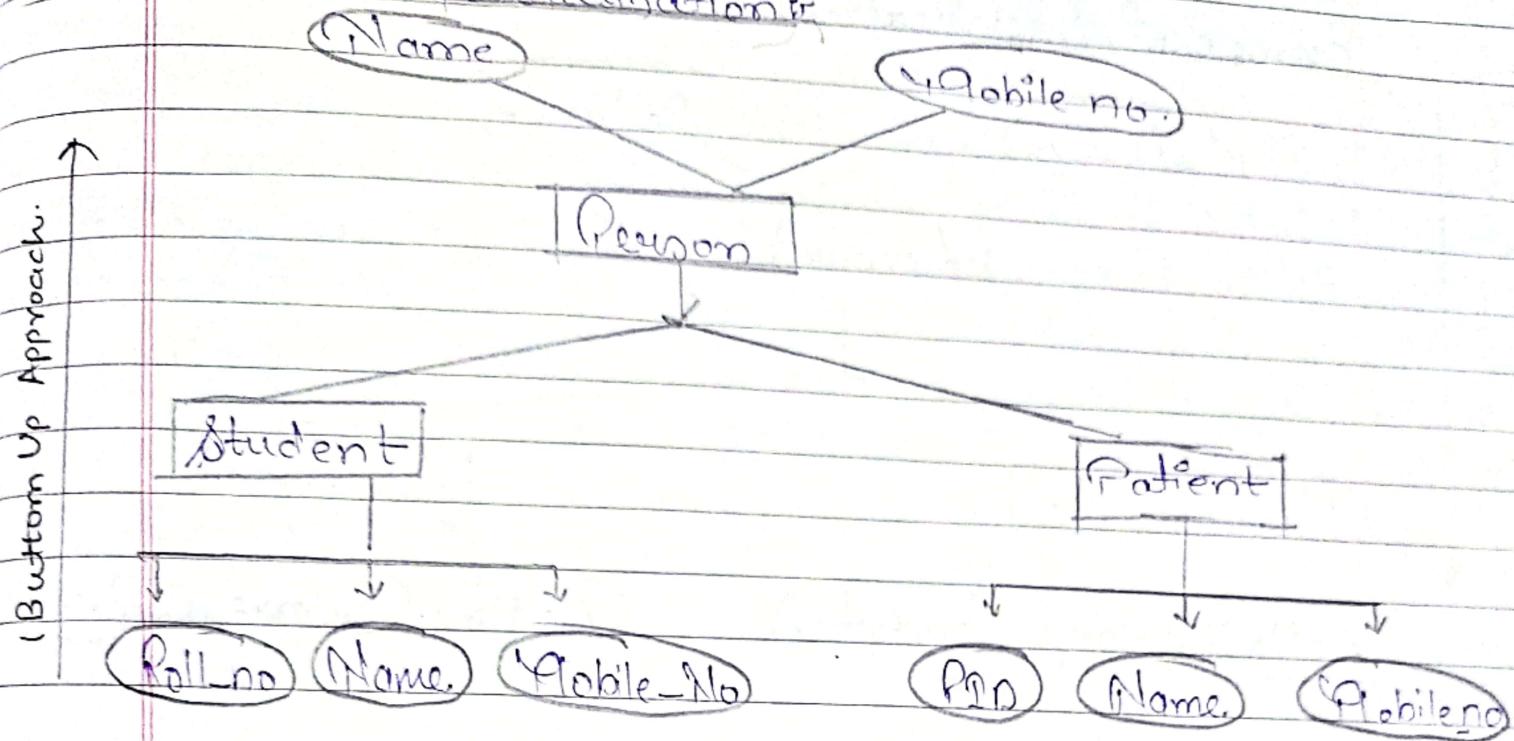
- i:-
- The strong entity has a primary key. Its existence is not dependent on any other entity. In a schema
 - strong entity is represented by rectangle. Single
 - strong entity always has the primary key in the set of attributes, that describes the strong entity. The member of a strong entity set is called dominant entity set.

Ex:- Professor is a strong entity. here, & the primary key is Professor_id.

6. Generalizations

- ii:-
- It is a bottom-up approach in which multiple lower level entities are combined to form a single higher level entity.
 - Generalization is usually used to find common attributes among entities to form a generalized entity. It can also be thought as the opposite of specialization.
 - This newly formed entity is called generalization entity.
 - A generalization hierarchy is a form of abstraction that specifies that two or more entities that share common attributes can be generalized into a higher level entity-type called as supertype.

Example of Generalization

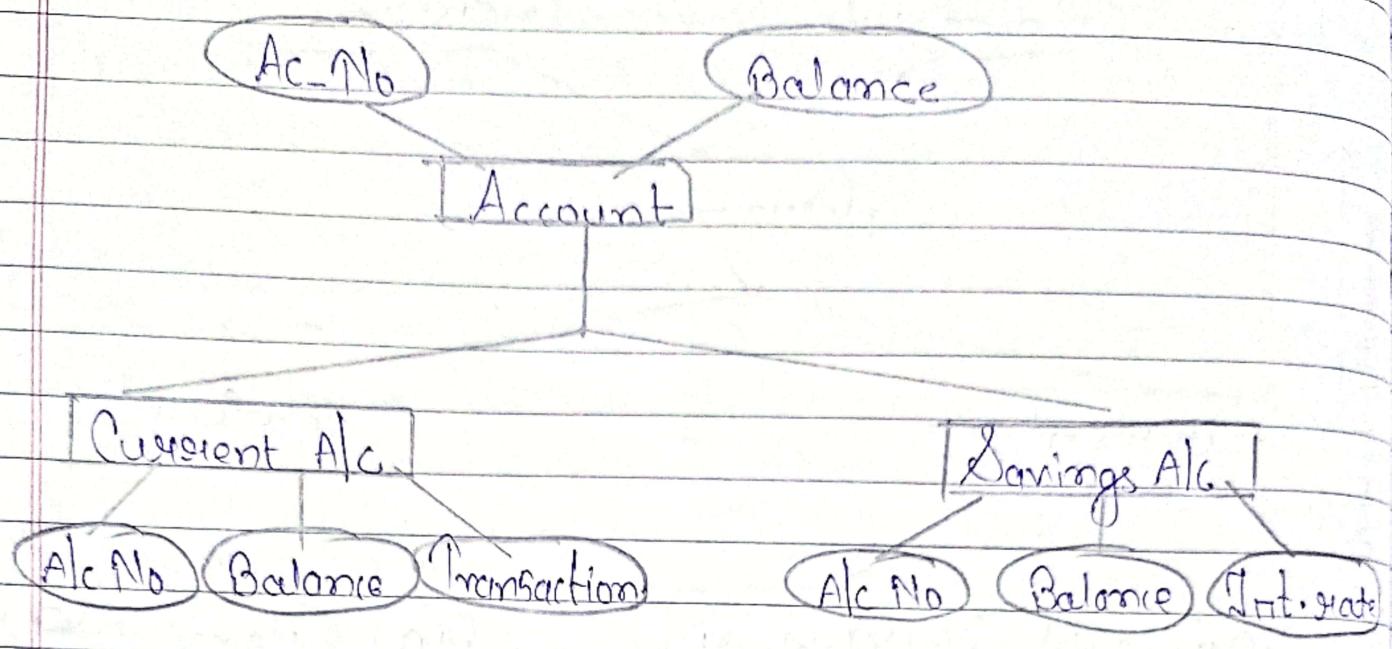


7. Specializations

- Specialization is a process in which an entity is divided into sub-entities. You can think of it as a reverse process of generalization.
- Specialization is top-down process.
- In generalization 2 entities combine together to form a new higher level entity.
- The idea behind specialization is to find the subsets of entities that have few distinguish attributes.
- Specialization is the abstracting process of introducing new characteristics to an existing class of objects to create one or more new classes of objects.

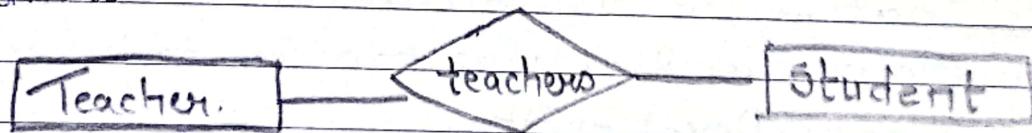
Top down Approach.

Example of Specialization



Q87] Explain Relationship with its types.

Ans:- Relationship & A Relationship describes relation between entities. Relationship is represented using diamonds or rhombus.



* There are 3 types of relationship.

1. Binary Relationship & Binary relationship means relation between 2 entities. Cardinality constraint defines the maximum number of relationship instances in which an entity can participate.

Cardinality Ratios

- Many-to-many cardinality (m:n).
- many-to-one cardinality (m:1)
- One-to-many cardinality (1:n).
- One-to-one cardinality (1:1).

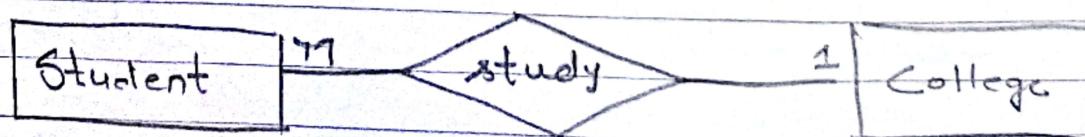
* One-to-one & when only one instance of an entity is associated with the relationship it is marked as '1:1'. only one instance of each entity should be associated with the relationship.



* One-to-many & when more than one instance of an entity is associated with a relationship. it is marked as '1:N' or '1:M'.



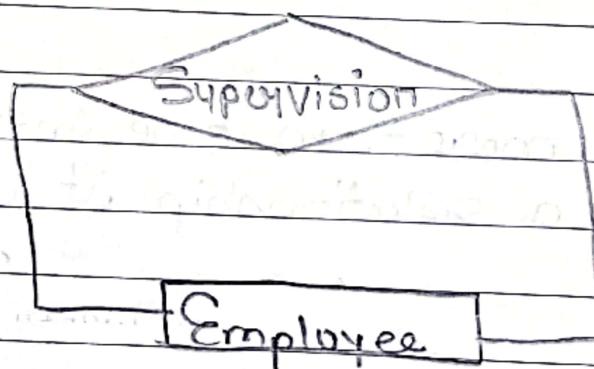
3. Many-to-one & when more than one instance of entity is associated with the relationship, it is marked as "N:1" or "M:1".



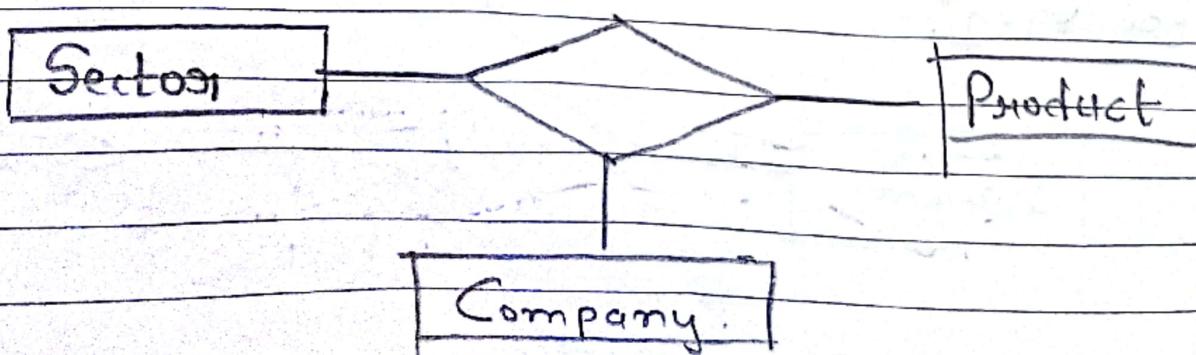
4. Many-to-Many:- The following image reflects that more than one instance of an entity on the left and more than one instance of an entity on the right can be associated with the relationship.



d. Recursive Relationship:- when an entity is related with itself is known as Recursive Relationship. In the below example an employee can be supervisor & be supervised, so there is a recursive relationship.



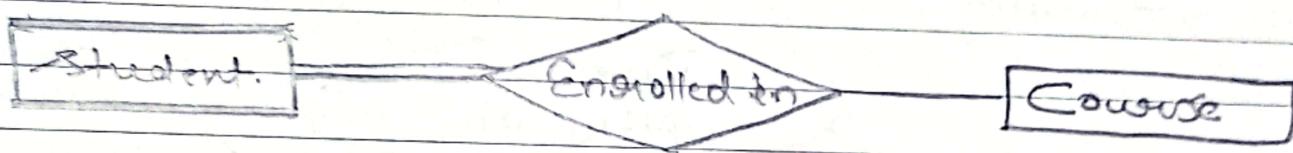
3. Ternary Relationship:- relationship of degree three is called Ternary relationship. These are 3 entities. In such relationships we always consider two entities together & then look upon the third.



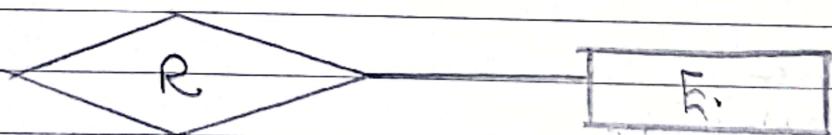
* **Participation Constraints** - It specifies whether the existence of an entity depends on its being related to another entity via the relationship type.

There are 2 types of participation constraints:-

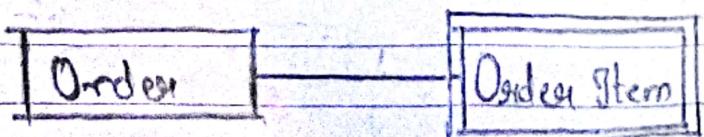
1. **Total Participation** - Total participation is when each entity in the entity set occurs in at least one relationship in that relationship set. That is why, it is called as mandatory participation.



2. **Partial Participation** - It specifies that each entity in the entity set may or may not participate in the relationship instance in that relationship set. That is why, it is called as optional participation.



* **Weak Entity** - A weak entity is an entity that depends on the existence of another entity. In more technical terms, it can be defined as an entity that cannot be identified by its own attributes. It uses a foreign key combined with its attributes to form the primary key.



Q88] Explain DDL and DML Commands.

A88] DDL stands for Data Definition Language. DDL is used for specifying the database schema. It is used for creating tables, schema, indexes, constraints etc. in database. It basically defines the column (Attributes) in table.

* Commands as DDL :-

- I] Create
- II] Alter
- III] Drop.

I] Create :- It is used to create the database or its objects. (like table).

Syntax :-

Create table table-name.

Q. Alter :- It is used to alter the structure of the database.

Syntax :-

Alter table table-name

Add column name column-definition;

3. Drop :- It is used to delete objects from the database.

Syntax :-

Drop table table-name;

* DML stands for data Manipulation Language (Data). It is a language that enables users to access or manipulate data as organized by the appropriate data model.

The types of access are :-

- Retrieval of information stored in the database.
- Insertion of new information into the database.
- Deletion of information from the database.
- Modification of information stored in the database.

* DQL commands are :-

i] Select

ii] Insert

iii] Update

iv] Delete.

ii] Select :-

Syntax of select :-

Select * from table_name;

Syntax of Insert :-

Insert into table_name values (col1, col2, col3);

Syntax of Update :-

Update table_name;

set col1 = expression1;

where conditions;

Syntax of Delete ;

Delete from table_name

where conditions;