**1.EXPLAIN IN DETAIL ABOUT THE CONCEPTS OF ER MODEL.**

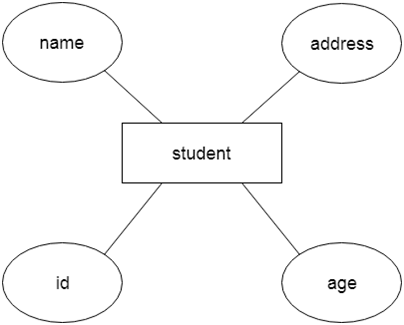
**ER MODEL:**

ER model stands for an Entity-Relationship model. It is a high-level data model. This model is used to define the data elements and relationship for a specified system.

It develops a conceptual design for the database. It also develops a very simple and easy to design view of data.

In ER modeling, the database structure is portrayed as a diagram called an entity-relationship diagram.

For example, Suppose we design a school database. In this database, the student will be an entity with attributes like address, name, id, age, etc. The address can be another entity with attributes like city, street name, pin code, etc and there will be a relationship between them.



**COMPONENTS OF ER DIAGRAM**



**1.ENTITY**

An entity may be any object, class, person or place. In the ER diagram, an entity can be represented as rectangles.

Consider an organization as an example- manager, product, employee, department etc. can be taken as an entity.



**a. Weak entity:**

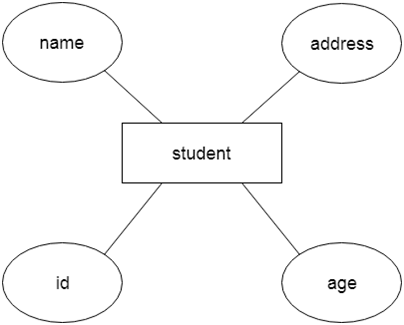
An entity that depends on another entity called a weak entity. The weak entity doesn't contain any key attribute of its own. The weak entity is represented by a double rectangle.



**2.ATTRIBUTE:**

The attribute is used to describe the property of an entity. Eclipse is used to represent an attribute.

For example, id, age, contact number, name, etc. can be attributes of a student.



**a. Key attribute:**

The key attribute is used to represent the main characteristics of an entity. It represents a primary key. The key attribute is represented by an ellipse with the text underlined.



**B. Composite attribute:**

An attribute that composed of many other attributes is known as a composite attribute. The composite attribute is represented by an ellipse, and those ellipses are connected with an ellipse.



**C. Multivalued attribute:**

An attribute can have more than one value. These attributes are known as a multivalued attribute. The double oval is used to represent multivalued attribute.

For example, a student can have more than one phone number.



**D. Derived attribute:**

An attribute that can be derived from other attribute is known as a 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.



**3.RELATIONSHIP:**

A relationship is used to describe the relation between entities. Diamond or rhombus is used to represent the relationship.



Types of relationship are as follows:

**A. One to one relationship:**

When only one instance of an entity is associated with the relationship, then it is known as one to one relationship.

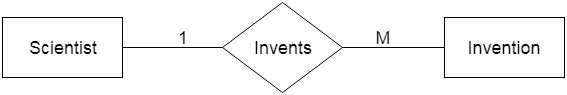
For example, A female can marry to one male, and a male can marry to one female.



**B. 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 known as a one-to-many relationship.

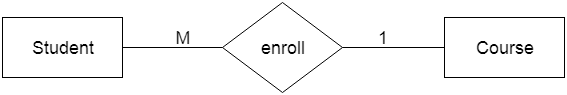
For example, Scientist can invent many inventions, but the invention is done by the only specific scientist.



**C. Many to one relationship:**

When more than one instance of the entity on the left, and only one instance of an entity on the right associates with the relationship then it is known as a many-to-one relationship.

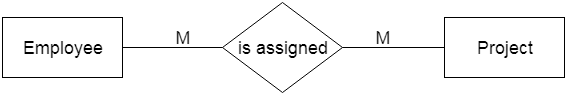
For example, Student enrolls for only one course, but a course can have many students.



**D. Many to many relationship:**

When more than one instance of the entity on the left, and more than one instance of an entity on the right associates with the relationship then it is known as a many-to-many relationship.

For example, Employee can assign by many projects and project can have many employees.



**2.what is view of data? How to specify a view? Write about view implementation techniques.**

**VIEW OF DATA:**

View of data in DBMS narrate how the data is visualized at each level of data abstraction.

Data abstraction allow developers to keep complex data structures away from the users. The developers achieve this by hiding the complex data structures through levels of abstraction.

There is one more feature that should be kept in mind i.e. the data independence. While changing the data schema at one level of the database must not modify the data schema at the next level. In this section, we will discuss the view of data in DBMS with data abstraction, data independence, data schema in detail.

View of Data in DBMS

•Data Abstraction

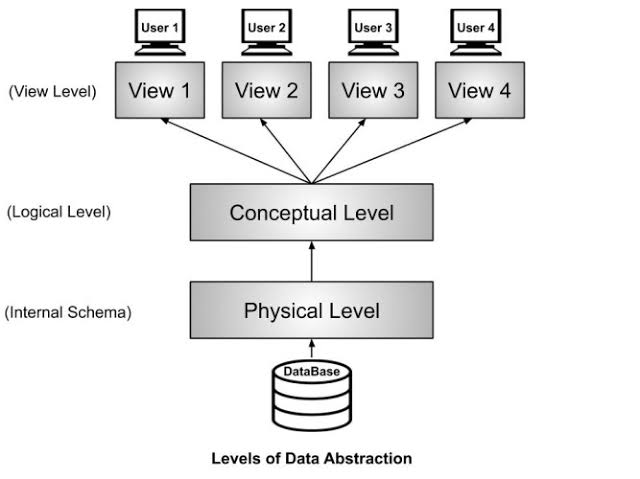
•Data Independence

•Instance and Schema

**Data abstraction:**

Data abstraction is hiding the complex data structure in order to simplify the user’s interface of the system. It is done because many of the users interacting with the database system are not that much computer trained to understand the complex data structures of the database system.

To achieve data abstraction, we will discuss a Three-Schema architecture which abstracts the database at three levels discussed below:



Three-Schema Architecture:

The main objective of this architecture is to have an effective separation between the user interface and the physical database. So, the user never has to be concerned regarding the internal storage of the database and it has a simplified interaction with the database system.

The three-schema architecture defines the view of data at three levels:

1.Physical level (internal level)

2.Logical level (conceptual level)

3.View level (external level)

**1.Physical level /internal level:**

The physical or the internal level schema describes how the data is stored in the hardware. It also describes how the data can be accessed. The physical level shows the data abstraction at the lowest level and it has complex data structures. Only the database administrator operates at this level.

**2.Logical / Conceptual level:**

It is a level above the physical level. Here, the data is stored in the form of the entity set, entities, their data types, the relationship among the entity sets, user operations performed to retrieve or modify the data and certain constraints on the data. Well adding constraints to the view of data adds the security. As users are restricted to access some particular parts of the database.

It is the developer and database administrator who operates at the logical or the conceptual level.

**3.View level/User level/External level:**

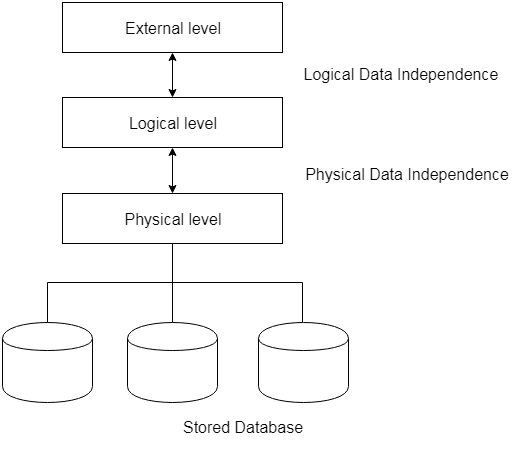
It is the highest level of data abstraction and exhibits only a part of the whole database. It

exhibits the data in which the user is interested. The view level can describe many views of the same data. Here, the user retrieves the information using different application from the database.

The figure below describes the three-schema architecture of the database:

**DATA INDEPENDENCE**

Data independence defines the extent to which the data schema can be changed at one level without modifying the data schema at the next level. Data independence can be classified as shown below:



**Logical data independence:**

Logical data independence describes the degree up to which the logical or conceptual schema can be changed without modifying the external schema. Now, a question arises what is the need to change the data schema at a logical or conceptual level?

Well, the changes to data schema at the logical level are made either to enlarge or reduce the database by adding or deleting more entities, entity sets, or changing the constraints on data.

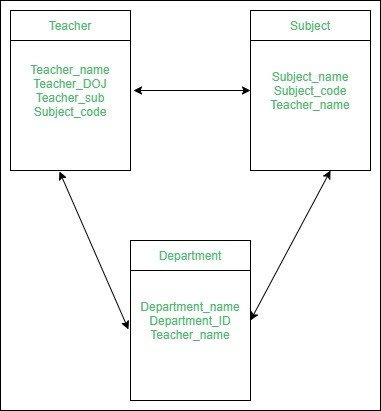
**Physical data independence:**

Physical data independence defines the extent up to which the data schema can be changed at the physical or internal level without modifying the data schema at logical and view level.

Well, the physical schema is changed if we add additional storage to the system or we reorganize some files to enhance the retrieval speed of the records.

**INSTANCES AND SCHEMAS**

**Instance:** : Instances are the collection of information stored at a particular moment. The instances can be changed by certain CRUD operations as like addition, deletion of data. It may be noted that any search query will not make any kind of changes in the instances.



**Schema:**Schema is the overall description of the database. The basic structure of how the data will be stored in the database is called schema.

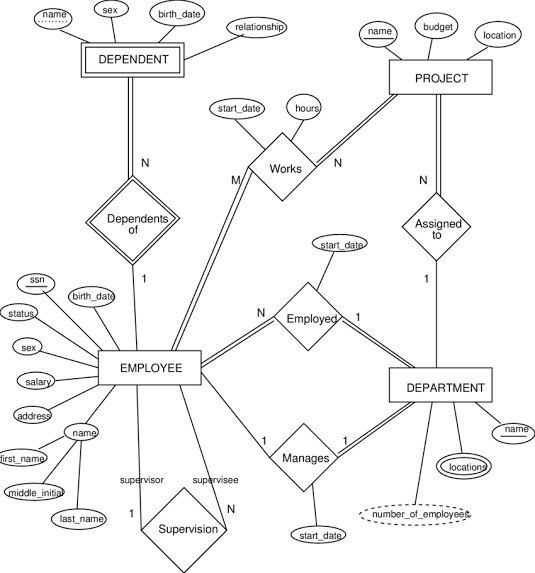
Schema is of three types: Logical Schema, Physical Schema and view Schema.

Logical Schema – It describes the database designed at logical level.

Physical Schema – It describes the database designed at physical level.

View Schema – It defines the design of the database at the view level.

**3.CONSTRUCT AN ER DIAGRAM ON EMPLOYEE DATABASE.**



**SHORT ANSWER QUESTIONS**

**1.EXPLAIN THE ADVANTAGES OF DBMS.**

**A. ADVANTAGES OF DBMS:**

**Reducing data redundancy**

The file based data management systems contained multiple files that were stored in many different locations in a system or even across multiple systems. Because of this, there were sometimes multiple copies of the same file which lead to data redundancy.

This is prevented in a database as there is a single database and any change in it is reflected immediately. Because of this, there is no chance of encountering duplicate data.

**Sharing of data**

In a database, the users of the database can share the data among themselves. There are various levels of authorisation to access the data, and consequently the data can only be shared based on the correct authorisation protocols being followed.

Many remote users can also access the database simultaneously and share the data between themselves.

**Data integrity**

Data integrity means that the data is accurate and consistent in the database. Data Integrity is very important as there are multiple databases in a DBMS. All of these databases contain data that is visible to multiple users. So it is necessary to ensure that the data is correct and consistent in all the databases and for all the users.

**Data integrity**

Data Security is vital concept in a database. Only authorised users should be allowed to access the database and their identity should be authenticated using a username and password. Unauthorised users should not be allowed to access the database under any circumstances as it violates the integrity constraints.

**Privacy**

The privacy rule in a database means only the authorized users can access a database according to its privacy constraints. There are levels of database access and a user can only view the data he is allowed to. For example - In social networking sites, access constraints are different for different accounts a user may want to access.

**Back up and recovery**

Database Management System automatically takes care of backup and recovery. The users don't need to backup data periodically because this is taken care of by the DBMS. Moreover, it also restores the database after a crash or system failure to its previous condition.

**Data consistency**

Data consistency is ensured in a database because there is no data redundancy. All data appears consistently across the database and the data is same for all the users viewing the database. Moreover, any changes made to the database are immediately reflected to all the users and there is no data inconsistency.

**2.Explian the difference among Entity, Entity type and Entity set.**

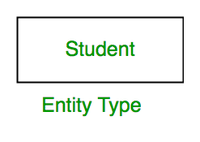
A. 1.Entity: An entity is a thing in a real-world with independent existence. An entity can exist independently and is distinguishable from other objects. It can be identified uniquely.

**Example:** A student with a particular roll number is an entity.

•An entity is represented by a set of attributes.

•In a particular relation in RDBMS, a particular record is called an entity.

2.Entity type:It refers to the category that a particular entity belongs to.

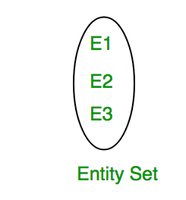


**Example:**A table named student in a university database.

•The category of a particular entity in the relation in RDBMS is called the entity type.

•It is represented by the name of the table and its schema.

3.Entity set:An entity set is a collection or set of all entities of a particular entity type at any point in time. The type of all the entities should be the same.



**Example:**The collection of all the students from the student table at a particular instant of time is an example of an entity set.

•Entity sets need not be disjoint.The collection of all the entities in the relation of RDBMS is called an entity set.

Relation With Table:

Table Name : Student

Stud\_ID Stud\_Name Stud\_Age Stud\_Gender

1 Avi 19 M

2 Ayush 23 M

3 Nikhil 21 M

4 Riya 16 F

Entity : Each row is an entity.

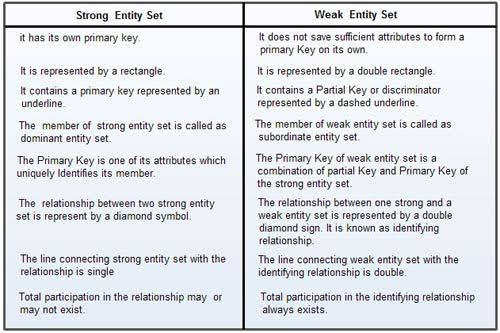
Example :

1 Avi 19 M

Entity Type : Each entity belongs to the student type. Hence, the type of entity here is a student.

Entity Set : The complete data set of all entities is called entity set. For the above table, the records with student id 1, 2, 3, 4 are the entity set.

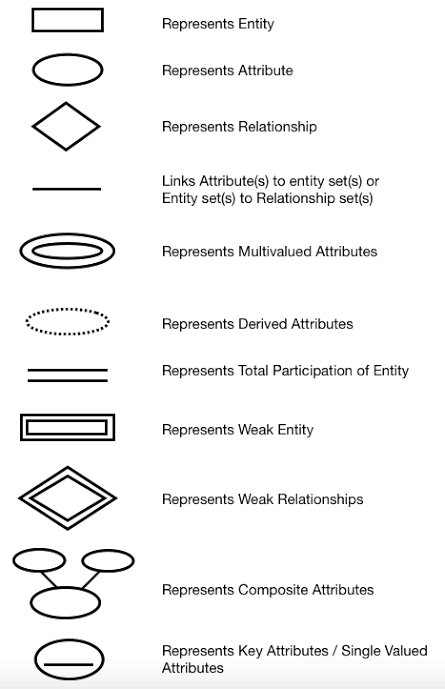
**3.Distinguish between strong entity and weak entity.**



**4.Explain the types of attributes.**

**A) please refer to essay Q1**

**5.list and explain basic symbols/notations used in ER diagram.**



**6.Define integrity constraints and it's uses.**

**A.Integrity constraints:**

Integrity Constraints are the compulsory conditions which should be satisfied by every data value present in the relational table at any instance of time to ensure that the database consists of only meaningful and relevant data.

There are four types of integrity constraints:

**Domain constraints:**Every attribute should have values within its defined domain.

**Key constraints:**There should be a primary key for every relational table.

**Entity integrity constraints:**No NULL values should be there for the Primary Keys.

**Referential integrity constraints:** In relational model, when two tables are related to each other with the help of some common attributes, the value of referencing attribute should be present in the referenced attribute else it should be NULL.

**Uses:**

1.Integrity constraints are a set of rules. It is used to maintain the quality of information.

2.Integrity constraints ensure that the data insertion, updating, and other processes have to be performed in such a way that data integrity is not affected.

3.Thus, integrity constraint is used to guard against accidental damage to the database.

**7.Define the following terms.**

**(i)Generalisation**

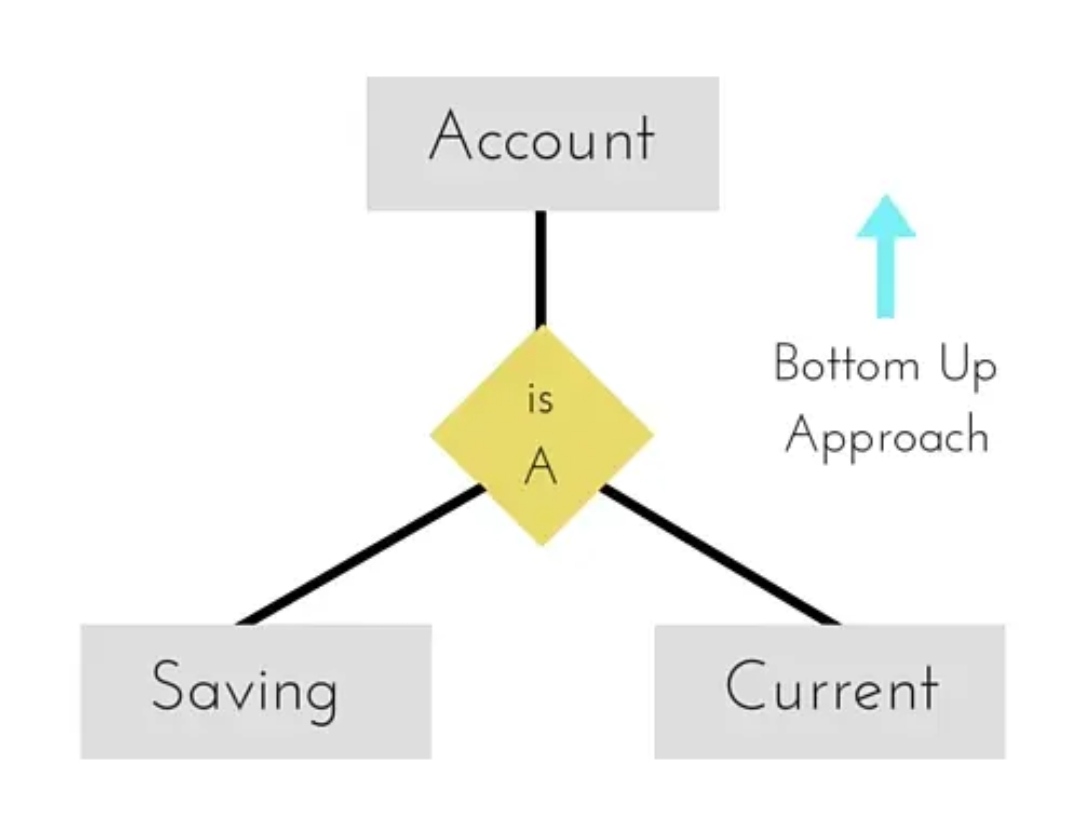
**(ii)Specialisation**

**(iii)Aggregation**

**Generalization:**

Generalization is a bottom-up approach in which two lower level entities combine to form a higher level entity. In generalization, the higher level entity can also combine with other lower level entities to make further higher level entity.

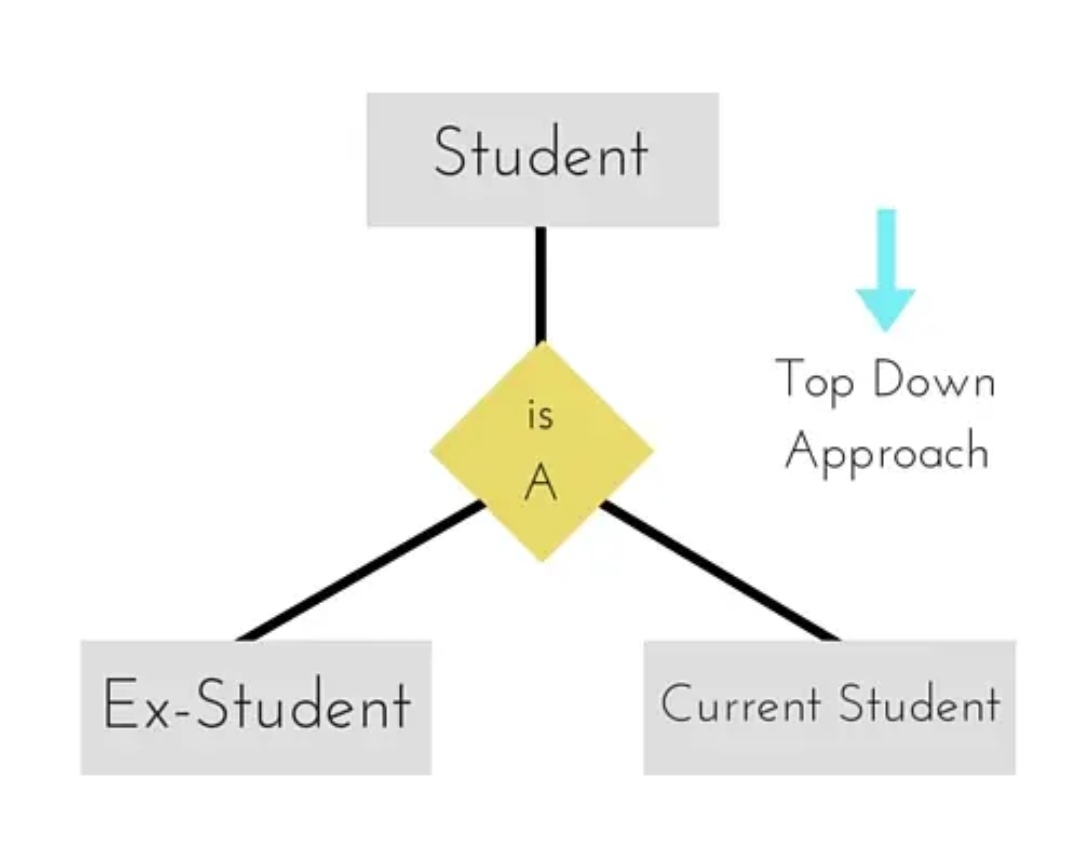
It's more like Superclass and Subclass system, but the only difference is the approach, which is bottom-up. Hence, entities are combined to form a more generalised entity, in other words, sub-classes are combined to form a super-class.



For example, Saving and Current account types entities can be generalised and an entity with name Account can be created, which covers both.

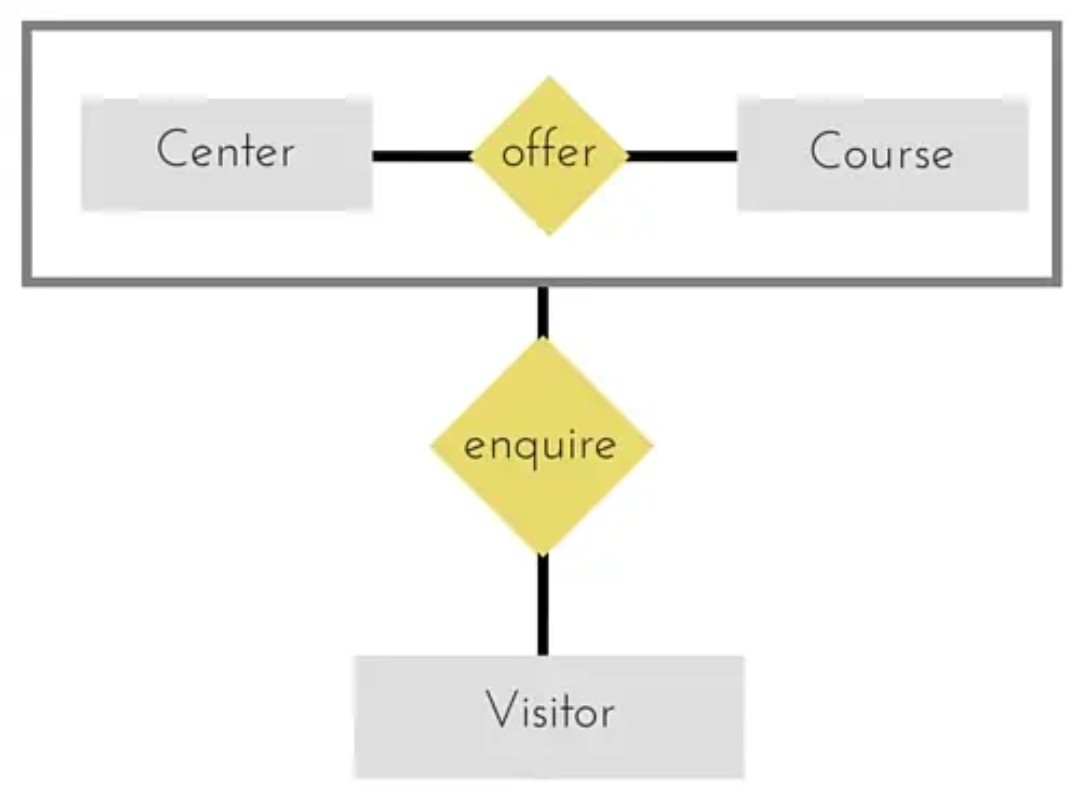
**Specialization:**

Specialization is opposite to Generalization. It is a top-down approach in which one higher level entity can be broken down into two lower level entity. In specialization, a higher level entity may not have any lower-level entity sets, it's possible.



**Aggregation:**

Aggregration is a process when relation between two entities is treated as a single entity.

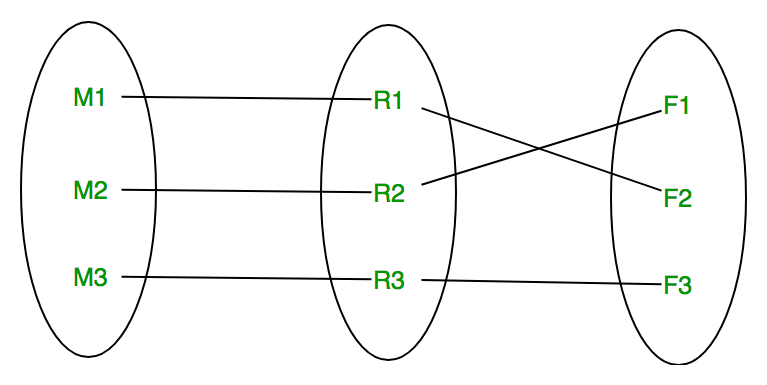


**8.Explain Cardinality using relationship sets.**

**Cardinality:**The number of times an entity of an entity set participates in a relationship set is known as cardinality. Cardinality can be of different types:

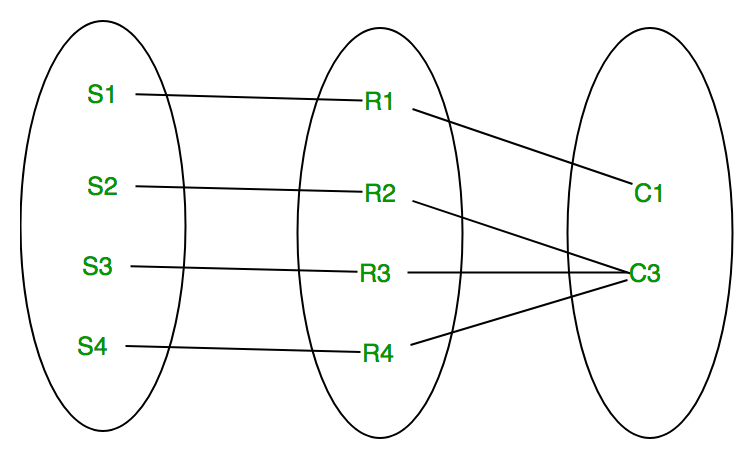
**1.One to One:**– When each entity in each entity set can take part only once in the relationship, the cardinality is one to one. Let us assume that a male can marry to one female and a female can marry to one male. So the relationship will be one to one.

Using Sets, it can be represented as:



**2.Many to one:**– When entities in one entity set can take part only once in the relationship set and entities in other entity set can take part more than once in the relationship set, cardinality is many to one.

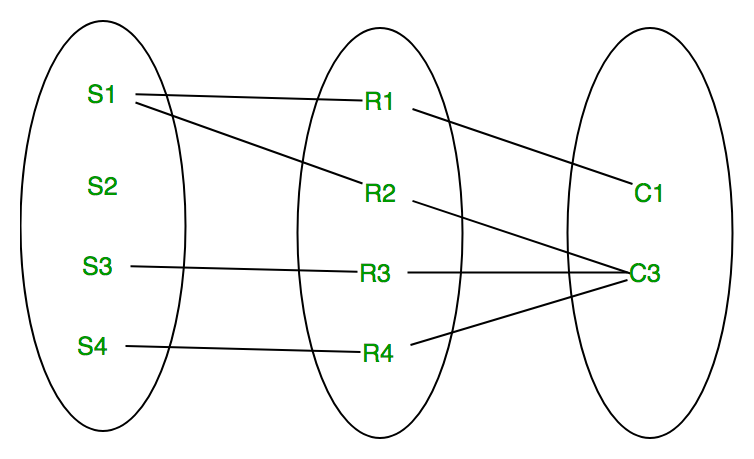
Using Sets, it can be represented as:



In this case, each student is taking only 1 course but 1 course has been taken by many students.

**3.Many to Many:**– When entities in all entity sets can take part more than once in the relationship cardinality is many to many.

Using sets, it can be represented as:



In this example, student S1 is enrolled in C1 and C3 and Course C3 is enrolled by S1, S3 and S4. So it is many to many relationships.

**9.Explain the abstraction of data in dbms.**

**A. Please refer to essay Q no. 2**

**10.Explian the participation constraints in ER model.**

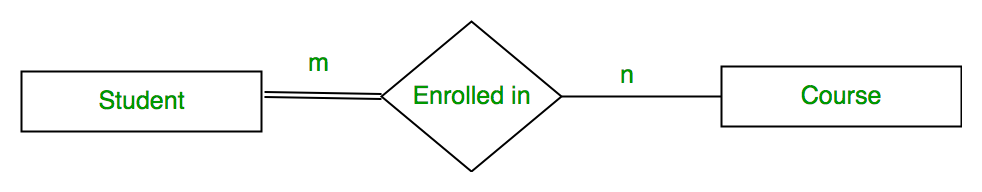
**Participation constraints:**

Participation Constraint is applied on the entity participating in the relationship set.

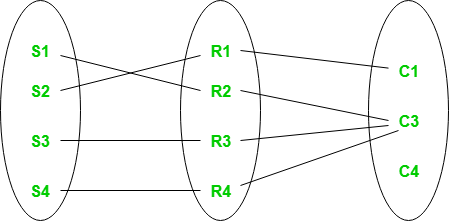
**1.Total participation:**– Each entity in the entity set must participate in the relationship. If each student must enroll in a course, the participation of student will be total. Total participation is shown by double line in ER diagram.

**2.Partial Participation:**– The entity in the entity set may or may NOT participate in the relationship. If some courses are not enrolled by any of the student, the participation of course will be partial.

The diagram depicts the ‘Enrolled in’ relationship set with Student Entity set having total participation and Course Entity set having partial participation.



Using set, it can be represented as,



Every student in Student Entity set is participating in relationship but there exists a course C4 which is not taking part in the relationship.

**MCQ's :**

1. Which of the following is used to specify whether the existence of

an entity depends on its being related to another entity via the

relationship type? [ ]

A. Entity integrity constraint

B. Cardinality ration

C. Participation constraint

D. Foreign key constraint

2. The type of attributes that can be divided into smaller parts is

classified as. [ ]

A. Multivalued attributes

B. Single valued attributes

C. Composite attributes

D. Atomic attributes

3. Consider a directed line (->) from the relationship set advisor

to both entity sets instructor and student. This indicates\_\_\_\_

cardinality. [ ]

A. One to many

B. One to one

C. Many to many

D. Many to one

4. For a weak entity set to be meaning ful, it must be associated

with another entity set called the. [ ]

A. Identity set

B. Owner set

C. Neighbour set

D. Strong entity set

5. Every weak entity set can be converted into a strong entity set

by. [ ]

A. Using generalization

B. Adding appropriate attributes

C. Using aggregation

D. None of the above

6. Relationships among entities of a single class are called

\_\_\_\_. [ ]

A. IS - A relationship

B. Recursive relationship

C. HAS - A relationship

D. None

7. The system which provides the active rules to initiate

ceratin actions after meeting specific condition is

classified as. [ ]

A. Indexed structure system

B. Triggered database system

C. Active database system

D. Graphical business structure

8. The type of constraints that specifies the uniqueness of

data stored in the database are considered as. [ ]

A. Semantics

B. Business rules

C. Controlled rules

D. Structural rules considered as.

9. The type data abstraction which allows the conceptual

representation of data in database management system

is. [ ]

A. Logical design model

B. Data model

C. Interface model

D. User friendly model

10. A main purpose of DBMS is to provide \_\_\_\_\_

view of data user. [ ]

A. Abstraction

B. None of these

C. Complete

D. Partial

**Match the following :**

1. Rectangle. [ ]. A. A distinct real world item

in a application

2. Table. [ ]. B. View of data

3. Entity. [ ] C. E-R model

4. Ellipse [ ] D. Relation

5. Logical level [ ] E. Attributes

**Key for MCQ’**

1. C

2. D

3. B

4. A

5. B

6. B

7. C

8. A

9. B

10. A

**Key for match the following :**

1. C

2. D

3. A

4. E

5. B