

1. What is the degree of a relation?

Sol: The degree of relationship is the number of occurrences in one entity which are associated to the number of occurrences in another.

There are three degrees of relation known as:

1. One-to-one (1:1)
2. One-to-many (1:M)
3. Many-to-many (M:N)

2. What is data integrity?

Sol: Data integrity refers to the accuracy, consistency and reliability of data that is stored in the database. Both database developers are responsible for implementing data integrity within one or a set of related database.

There are four types of data integrity

1. Row Integrity
2. Column Integrity
3. Referential Integrity
4. User-defined Integrity.

3. What is referential Integrity?

Sol: Referential integrity (RI) is a relational database concept, which states that table relationship must always be consistent. In other words, any foreign key field must agree with the primary key that is referenced by the foreign key. Thus any primary key field changes must be applied to all foreign key, or not at all. The same restriction also applies to foreign keys in that any update must be propagated to the primary parent key.

4. What is a composite key?

Sol: Composite key is a key which is the combination of more than one field or column of a given table. It may be a candidate key or primary key.

5. What is a foreign key?

Sol: In relational databases, a foreign key is a field or column that is used to establish a link between two tables. we ~~can~~ can also say that foreign key in one table used to point primary key in another table.

6. What is a primary key?

Sol: Primary key is the first key which is used to identify one & only one instance of an entity uniquely.
An entity can contain multiple key.

7. What is the difference between weak & strong entity set?

Strong Entity Set	Weak Entity Set
<ul style="list-style-type: none">• It has its own primary key• Represented by a rectangle• Contains a primary key represented by an underline.• member of strong entity set is called as dominant entity set.	<ul style="list-style-type: none">• It doesn't have sufficient attributes to form a primary key on its own.• Represented by a double rectangle• Contains partial key or discriminator represented by a dashed underline.• member of weak entity set is a combination of partial called as subordinate entity set.

Q. What is a strong entity & weak entity?

Sol: Strong entity has primary key. Weak entities are dependent on strong entity. Its existence is not dependent on any other entity. Strong entity is represented by a single rectangle.

Weak entity in DBMS do not have a primary key and are dependent on the parent entity. It mainly depends on other entities.

9. What is derived attribute? Give an example.

Sol: Attributes which can be derived from other information is called derived attribute. For eg: - For person entity, the value of age can be determined from the current date & the value of that person's DOB.

10. What is meant by entity set?

Sol: The collection of all entities of a particular entity type in the database at any point in time is called an entity set.

11. What are attribute of an entity?

Sol: Attributes are the particular properties that describes a entity. A particular entity will have a value for each of it's attribute. The attribute value that describes each ~~attribute~~ entity become a major part of the data stored in the database.

12) Discuss the use of foreign key with an example.

Sol: Foreign key is used to prevent actions that would destroy links between tables. Foreign key also prevent invalid data from being inserted into the foreign key column, because it has to ~~be~~ be one of the value contained in the table it points to

Example

Consider two tables

"Student" table

Sta_Id	First Name	Roll Age
101	Henson	18
102	Vinu	17

"Mark" table

Mark-Id	Mark	Sta-Id
1	10	1
2	21	2

Here Sta-id column present in "Mark" table points to the stu-id column in the "Student" table.

The Stu-id column in the Student table is the PRIMARY KEY in the Student table.

The Stu-id column in the Mark table is a foreign key in the mark table.

13) Define Entity Integrity

Sol: Entity integrity is the mechanism the system provides to maintain primary key. The primary key serves as a unique identifier for row in the table.

Entity Integrity ensures two property for primary keys:

- Primary key for a row is unique, it does not match primary key of any other row in the table.

- Primary key ~~for a~~ is not null, no component of the primary key.

Advantages of using Relational model

- **Simplicity:** A relational data model is simpler than the hierarchical and network model.
- **Structural Independence:** The relational database is only concerned with data and not with a structure. This can improve the performance of the model.
- **Easy to use:** The relational model is easy as tables consisting of rows and columns is quite natural and simple to understand
- **Query capability:** It makes possible for a high-level query language like SQL to avoid complex database navigation.
- **Data independence:** The structure of a database can be changed without having to change any application.
- **Scalable:** Regarding a number of records, or rows, and the number of fields, a database should be enlarged to enhance its usability.

Disadvantages of using Relational model

- Few relational databases have limits on field lengths which can't be exceeded.
- Relational databases can sometimes become complex as the amount of data grows, and the relations between pieces of data become more complicated.
- Complex relational database systems may lead to isolated databases where the information cannot be shared from one system to another.

What Are the Basic units of ER model?

The ER model defines the conceptual view of a database. It works around real-world entities and the associations among them. At view level, the ER model is considered a good option for designing databases.

Entity

An entity can be a real-world object, either animate or inanimate, that can be easily identifiable. For example, in a school database, students, teachers, classes, and courses offered can be considered as entities. All these entities have some attributes or properties that give them their identity.

An entity set is a collection of similar types of entities. An entity set may contain entities with attribute sharing similar values. For example, a Students set may contain all the students of a school; likewise a Teachers set may contain all the teachers of a school from all faculties. Entity sets need not be disjoint.

Attributes

Entities are represented by means of their properties, called **attributes**. All attributes have values. For example, a student entity may have name, class, and age as attributes.

There exists a domain or range of values that can be assigned to attributes. For example, a student's name cannot be a numeric value. It has to be alphabetic. A student's age cannot be negative, etc.

Types of Attributes

- **Simple attribute** – Simple attributes are atomic values, which cannot be divided further. For example, a student's phone number is an atomic value of 10 digits.
- **Composite attribute** – Composite attributes are made of more than one simple attribute. For example, a student's complete name may have first_name and last_name.
- **Derived attribute** – Derived attributes are the attributes that do not exist in the physical database, but their values are derived from other attributes present in the database. For example, average_salary in a department should not be saved directly in the database, instead it can be derived. For another example, age can be derived from data_of_birth.
- **Single-value attribute** – Single-value attributes contain single value. For example – Social_Security_Number.
- **Multi-value attribute** – Multi-value attributes may contain more than one values. For example, a person can have more than one phone number, email_address, etc.

These attribute types can come together in a way like –

- simple single-valued attributes
- simple multi-valued attributes
- composite single-valued attributes

- composite multi-valued attributes

Entity-Set and Keys

Key is an attribute or collection of attributes that uniquely identifies an entity among entity set.

For example, the roll_number of a student makes him/her identifiable among students.

- **Super Key** – A set of attributes (one or more) that collectively identifies an entity in an entity set.
- **Candidate Key** – A minimal super key is called a candidate key. An entity set may have more than one candidate key.
- **Primary Key** – A primary key is one of the candidate keys chosen by the database designer to uniquely identify the entity set.

Relationship

The association among entities is called a relationship. For example, an employee **works_at** a department, a student **enrolls** in a course. Here, Works_at and Enrolls are called relationships.

Relationship Set

A set of relationships of similar type is called a relationship set. Like entities, a relationship too can have attributes. These attributes are called **descriptive attributes**.

Degree of Relationship

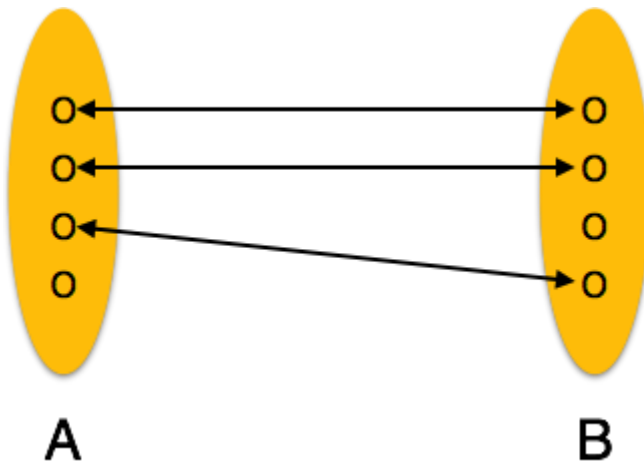
The number of participating entities in a relationship defines the degree of the relationship.

- Binary = degree 2
- Ternary = degree 3
- n-ary = degree

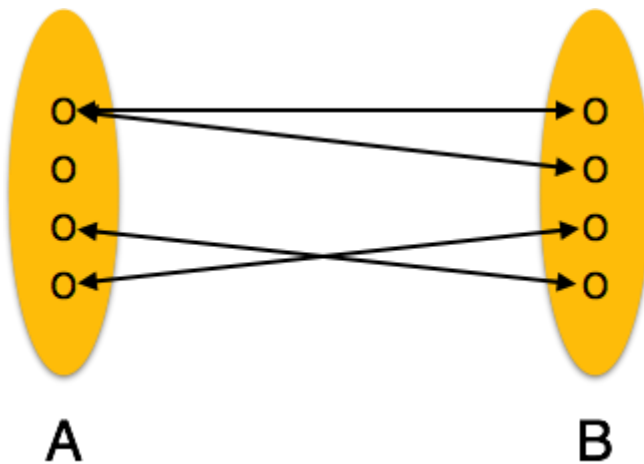
Mapping Cardinalities

Cardinality defines the number of entities in one entity set, which can be associated with the number of entities of other set via relationship set.

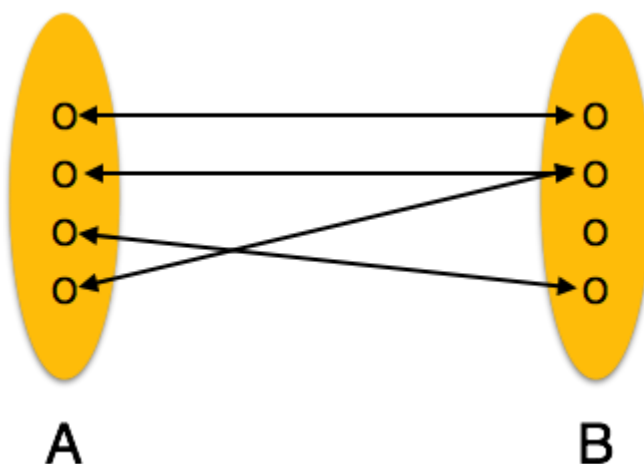
- **One-to-one** – One entity from entity set A can be associated with at most one entity of entity set B and vice versa.



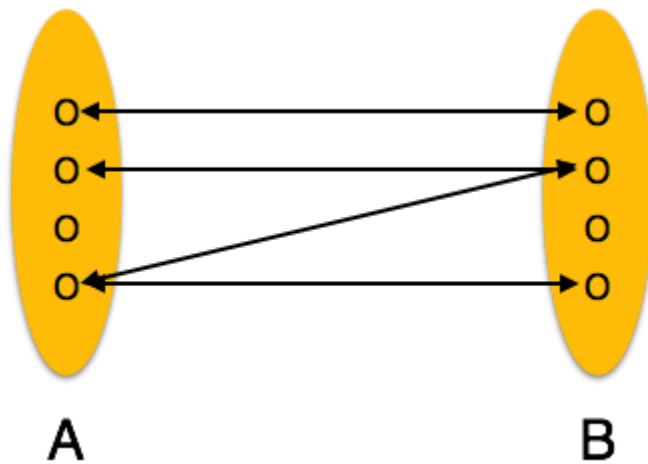
- **One-to-many** – One entity from entity set A can be associated with more than one entities of entity set B however an entity from entity set B, can be associated with at most one entity.



- **Many-to-one** – More than one entities from entity set A can be associated with at most one entity of entity set B, however an entity from entity set B can be associated with more than one entity from entity set A.



- **Many-to-many** – One entity from A can be associated with more than one entity from B and vice versa.



What is ER diagram? Explain the symbols used in it with the help of an example.

ER Model is represented by means of an ER diagram. Any object, for example, entities, attributes of an entity, relationship sets, and attributes of relationship sets, can be represented with the help of an ER diagram.

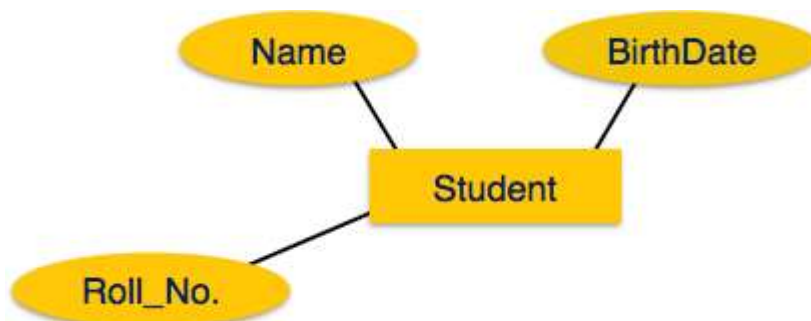
Entity

Entities are represented by means of rectangles. Rectangles are named with the entity set they represent.

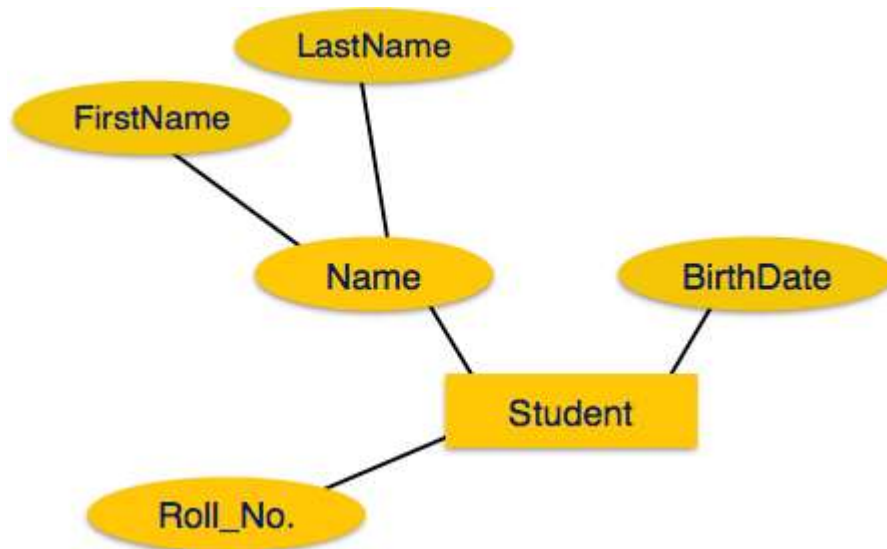


Attributes

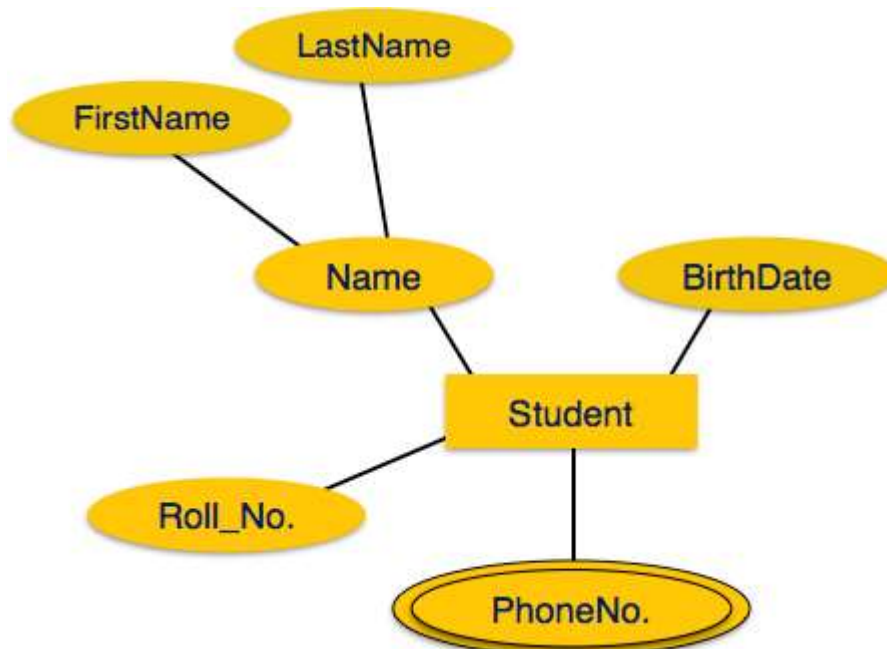
Attributes are the properties of entities. Attributes are represented by means of ellipses. Every ellipse represents one attribute and is directly connected to its entity (rectangle).



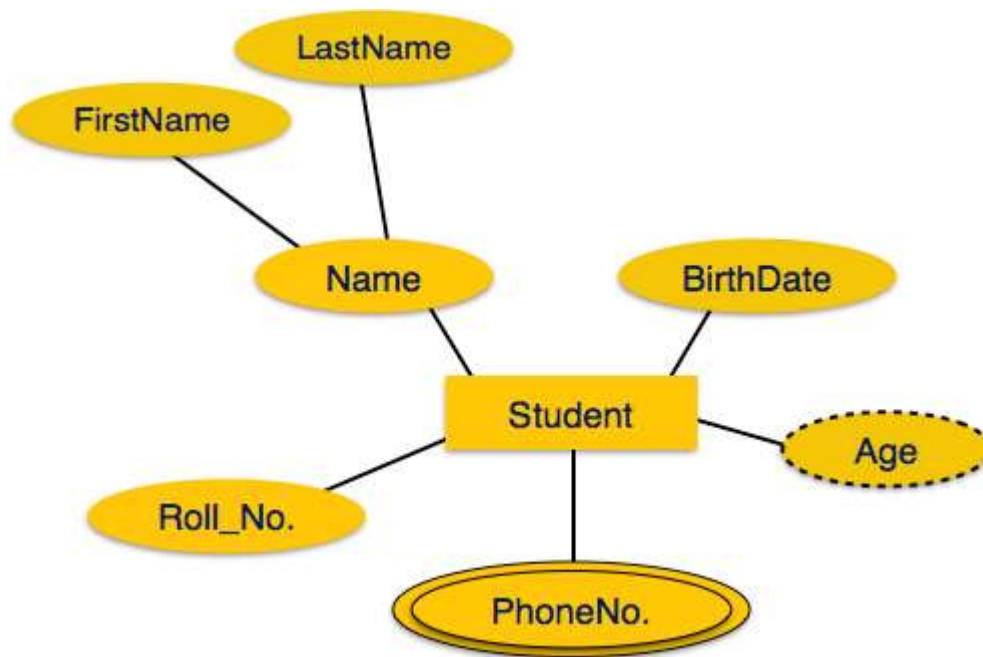
If the attributes are **composite**, they are further divided in a tree like structure. Every node is then connected to its attribute. That is, composite attributes are represented by ellipses that are connected with an ellipse.



Multivalued attributes are depicted by double ellipse.



Derived attributes are depicted by dashed ellipse.



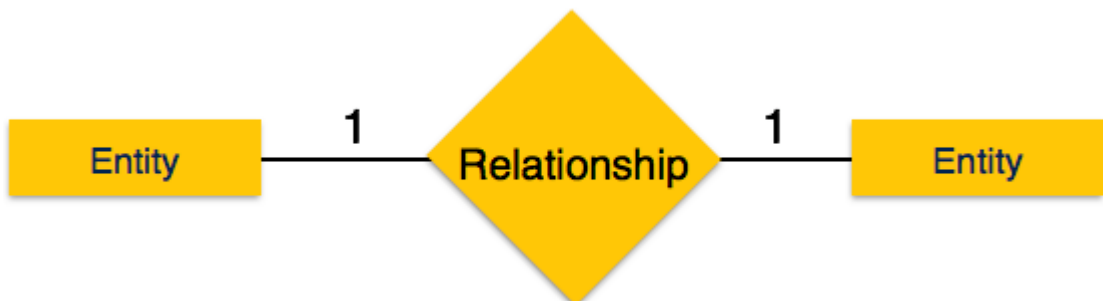
Relationship

Relationships are represented by diamond-shaped box. Name of the relationship is written inside the diamond-box. All the entities (rectangles) participating in a relationship, are connected to it by a line.

Binary Relationship and Cardinality

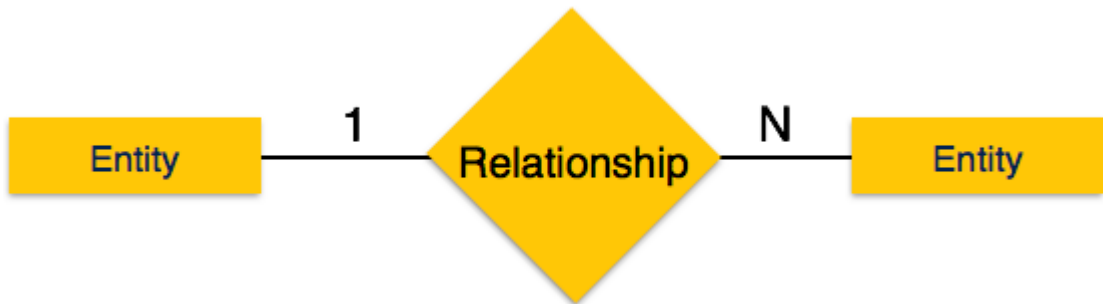
A relationship where two entities are participating is called a **binary relationship**. Cardinality is the number of instance of an entity from a relation that can be associated with the relation.

- **One-to-one** – When only one instance of an entity is associated with the relationship, it is marked as '1:1'. The following image reflects that only one instance of each entity should be associated with the relationship. It depicts one-to-one relationship.

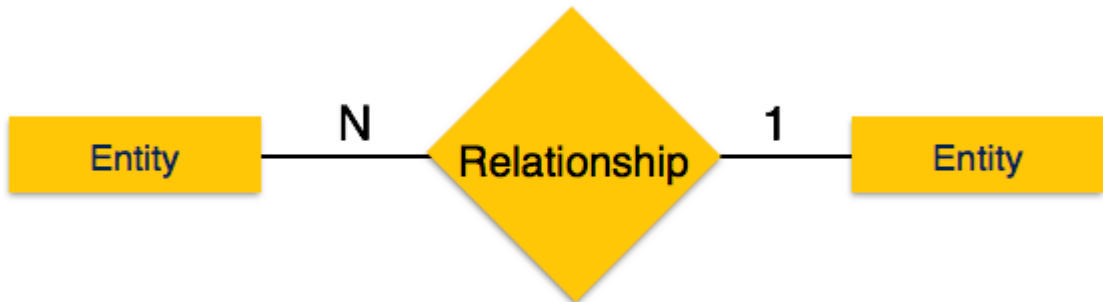


- **One-to-many** – When more than one instance of an entity is associated with a relationship, it is marked as '1:N'. The following image reflects that only one instance of entity on the left and more than one instance of an entity on the

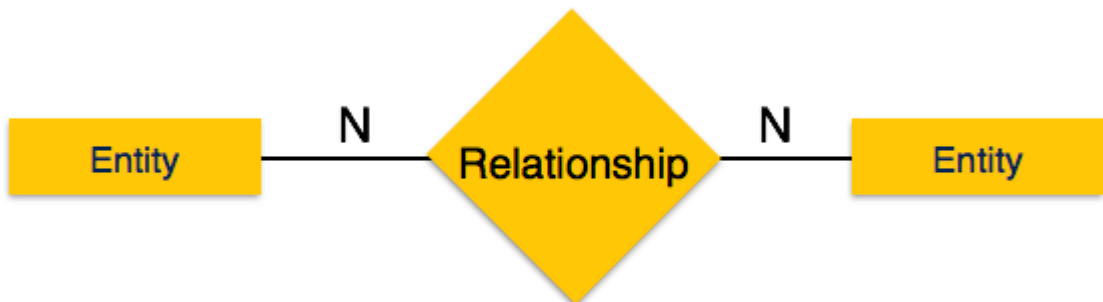
right can be associated with the relationship. It depicts one-to-many relationship.



- **Many-to-one** – When more than one instance of entity is associated with the relationship, it is marked as 'N:1'. The following image reflects that more than one instance of an entity on the left and only one instance of an entity on the right can be associated with the relationship. It depicts many-to-one relationship.



- **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. It depicts many-to-many relationship.



Participation Constraints

- **Total Participation** – Each entity is involved in the relationship. Total participation is represented by double lines.
- **Partial participation** – Not all entities are involved in the relationship. Partial participation is represented by single lines.

