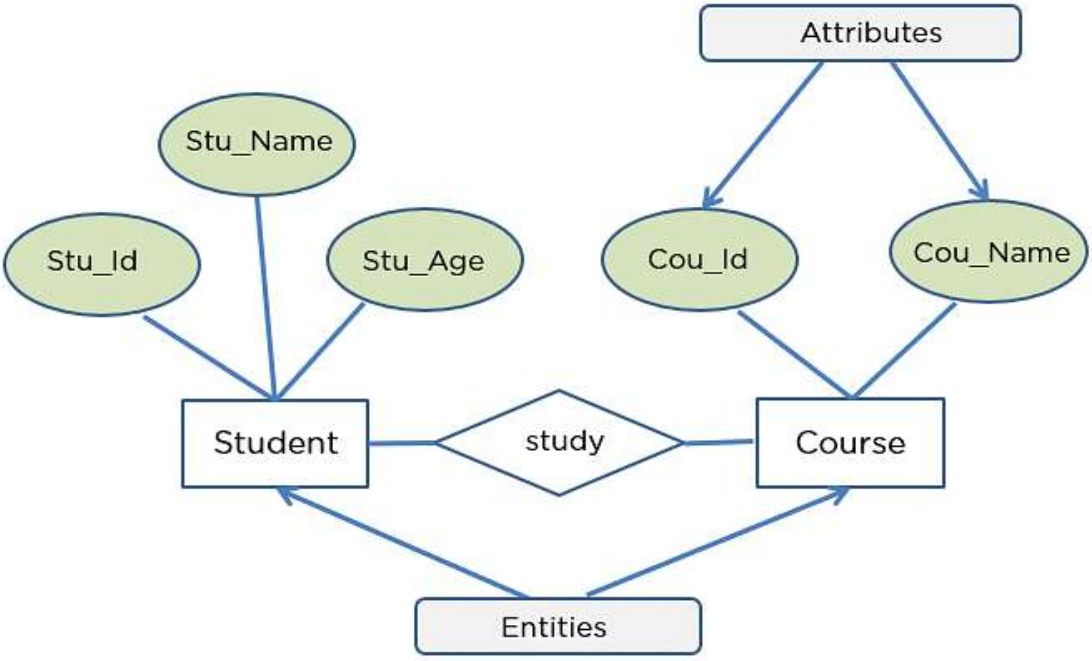
ER DIAGRAMS

An Entity Relationship Diagram is a diagram that represents relationships among entities in a database. It is commonly known as an ER Diagram. An ER Diagram in DBMS plays a crucial role in designing the database. Today's business world previews all the requirements demanded by the users in the form of an ER Diagram. Later, it's forwarded to the database administrators to design the database.



What is an ER Diagram?

An Entity Relationship Diagram (ER Diagram) pictorially explains the relationship between entities to be stored in a database. Fundamentally, the ER Diagram is a structural design of the database. It acts as a framework created with specialized symbols for the purpose of defining the relationship between the database entities. ER diagram is created based on three principal components: entities, attributes, and relationships.

The following diagram showcases two entities - Student and Course, and their relationship. The relationship described between student and course is many-to-many, as a course can be opted by several students, and a student can opt for more than one course. Student entity

possesses attributes - Stu_Id, Stu_Name & Stu_Age. The course entity has attributes such as Cou_ID & Cou_Name.

What is an ER Model?

An Entity-Relationship Model represents the structure of the database with the help of a diagram. ER Modelling is a systematic process to design a database as it would require you to analyze all data requirements before implementing your database.

History of ER models

Peter Chen proposed ER Diagrams in 1971 to create a uniform convention that can be used as a conceptual modeling tool. Many models were presented and discussed, but none were suitable. The data structure diagrams offered by Charles Bachman also inspired his model.

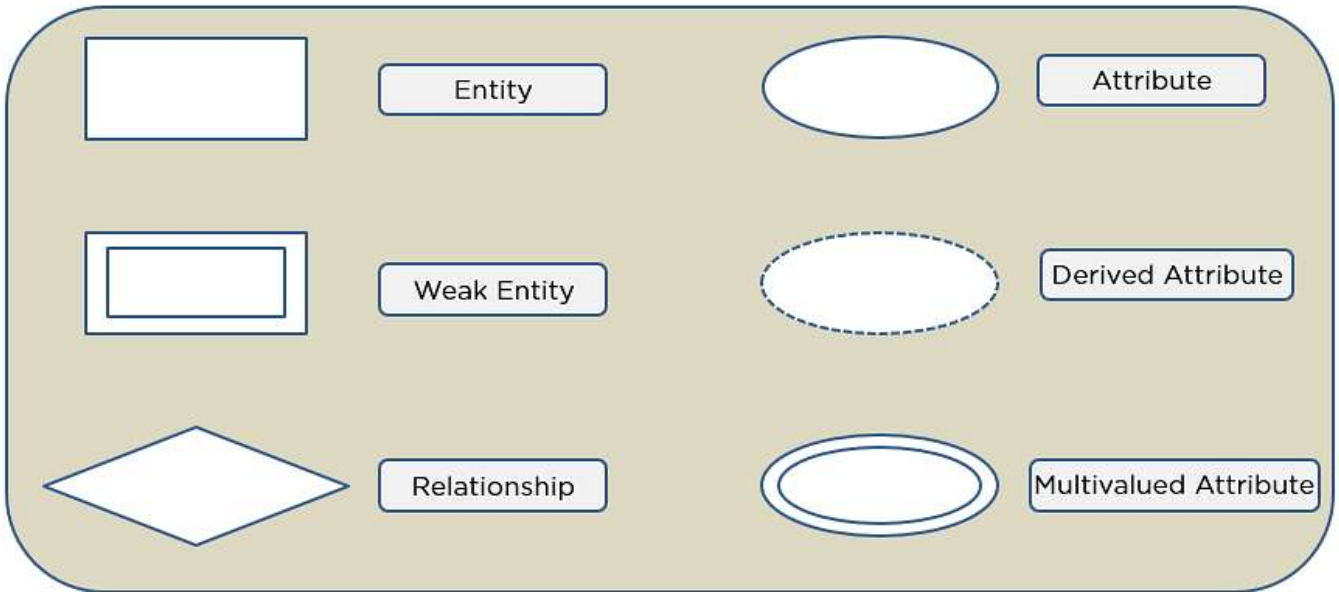
Why Use ER Diagrams in DBMS?

- ER Diagram helps you conceptualize the database and lets you know which fields need to be embedded for a particular entity
- ER Diagram gives a better understanding of the information to be stored in a database
- It reduces complexity and allows database designers to build databases quickly
- It helps to describe elements using Entity-Relationship models
- It allows users to get a preview of the logical structure of the database

Symbols Used in ER Diagrams

- Rectangles: This Entity Relationship Diagram symbol represents entity types
- Ellipses: This symbol represents attributes

- Diamonds: This symbol represents relationship types
- Lines: It links attributes to entity types and entity types with other relationship types
- Primary key: Here, it underlines the attributes
- Double Ellipses: Represents multi-valued attributes



Components of ER Diagram

You base an ER Diagram on three basic concepts:

- Entities
  - Weak Entity
- Attributes
  - Key Attribute
  - Composite Attribute
  - Multivalued Attribute
  - Derived Attribute
- Relationships
  - One-to-One Relationships
  - One-to-Many Relationships

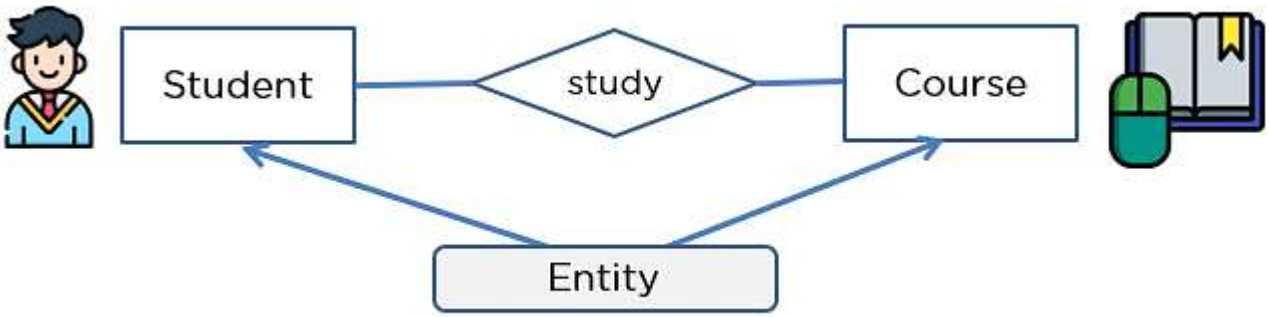
- Many-to-One Relationships
- Many-to-Many Relationships

Entities

An entity can be either a living or non-living component.

It showcases an entity as a rectangle in an ER diagram.

For example, in a student study course, both the student and the course are entities.

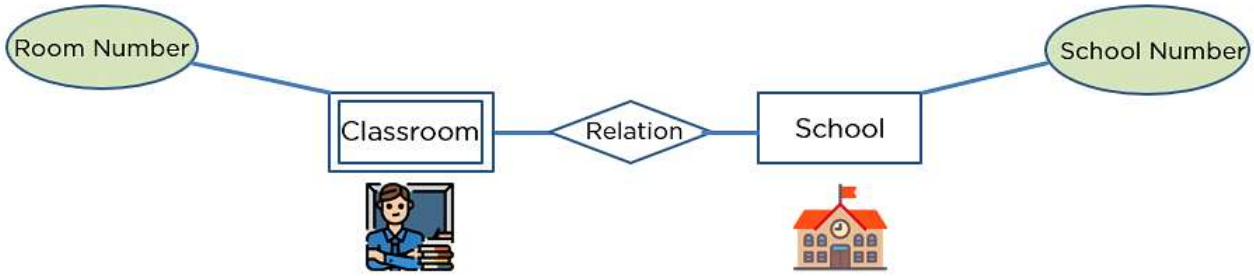


Weak Entity

An entity that makes reliance over another entity is called a weak entity

You showcase the weak entity as a double rectangle in ER Diagram.

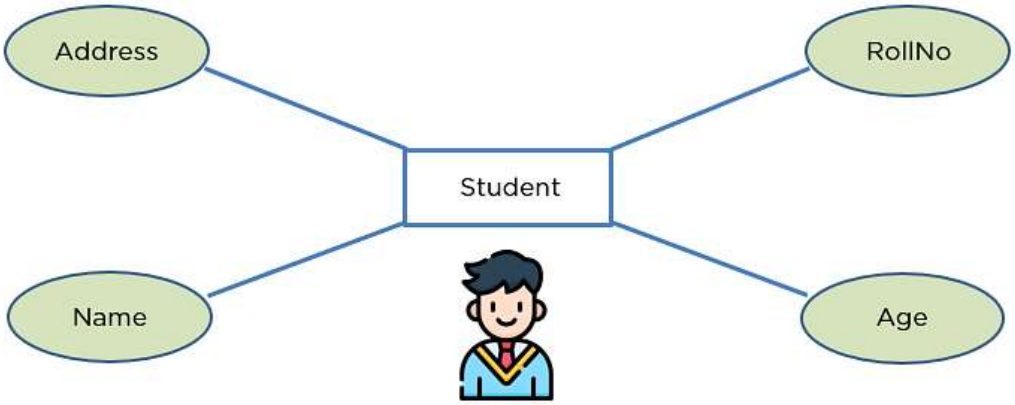
In the example below, school is a strong entity because it has a primary key attribute - school number. Unlike school, the classroom is a weak entity because it does not have any primary key and the room number here acts only as a discriminator.



Attribute

An attribute exhibits the properties of an entity.

You can illustrate an attribute with an oval shape in an ER diagram.

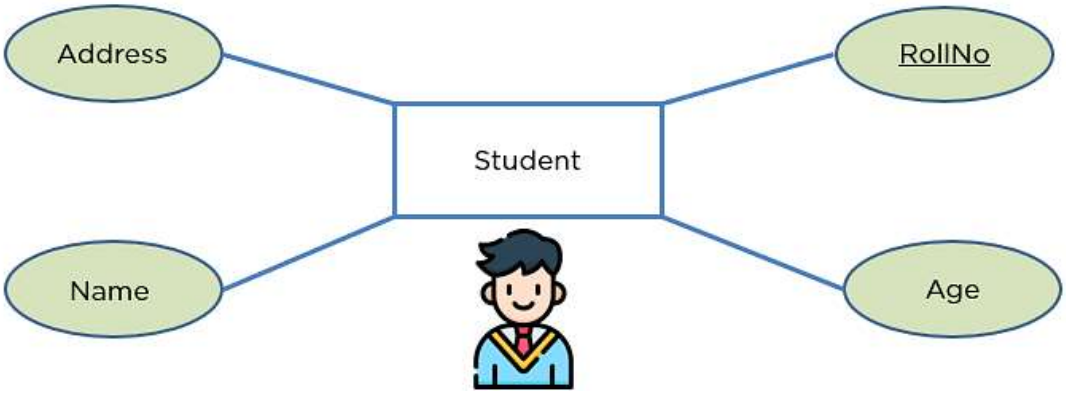


Key Attribute

Key attribute uniquely identifies an entity from an entity set.

It underlines the text of a key attribute.

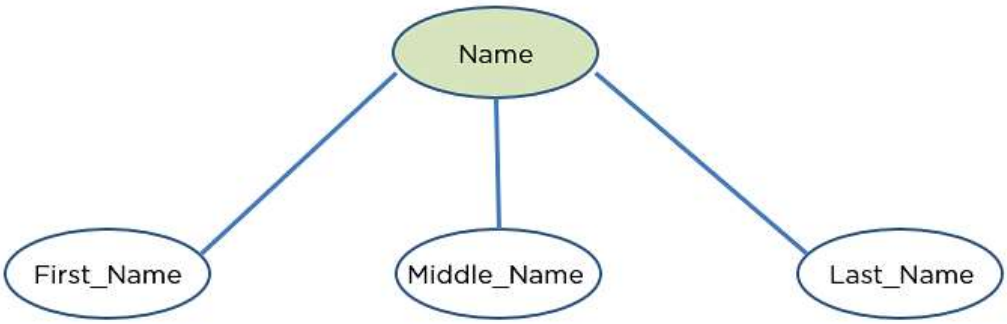
For example: For a student entity, the roll number can uniquely identify a student from a set of students.



Composite Attribute

An attribute that is composed of several other attributes is known as a composite attribute.

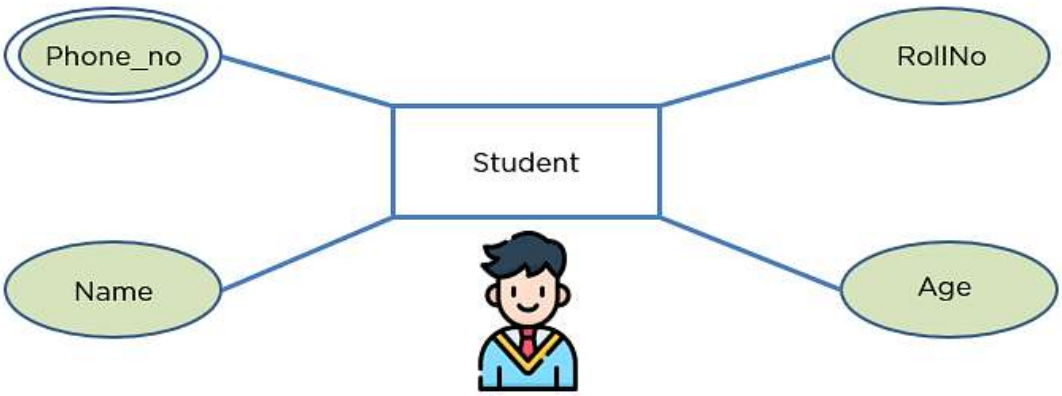
An oval showcases the composite attribute, and the composite attribute oval is further connected with other ovals.



Multivalued Attribute

Some attributes can possess over one value, those attributes are called multivalued attributes.

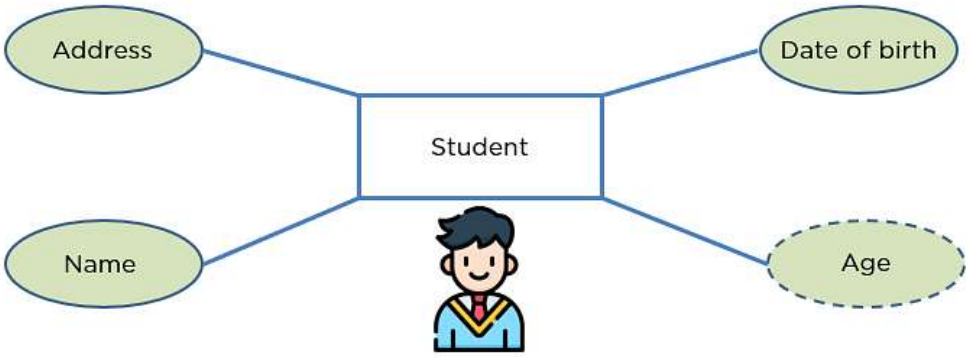
The double oval shape is used to represent a multivalued attribute.



Derived Attribute

An attribute that can be derived from other attributes of the entity is known as a derived attribute.

In the ER diagram, the dashed oval represents the derived attribute.

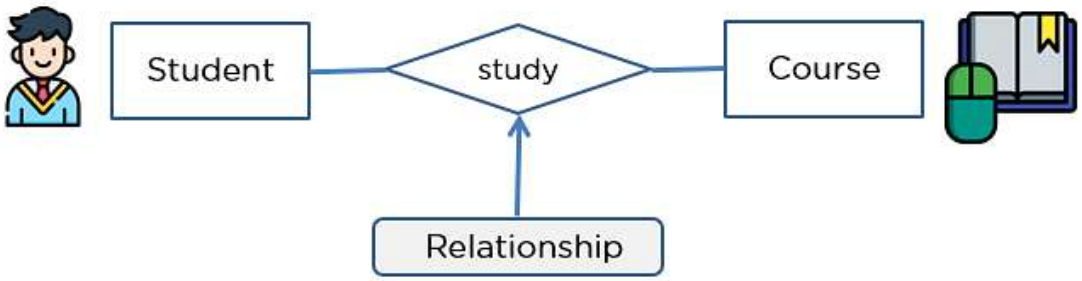


Relationship

The diamond shape showcases a relationship in the ER diagram.

It depicts the relationship between two entities.

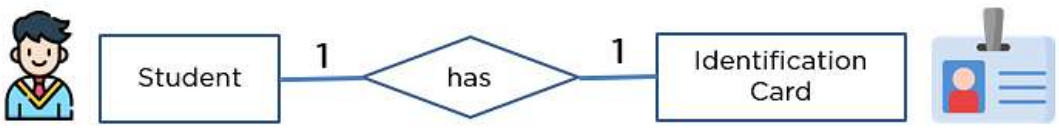
In the example below, both the student and the course are entities, and study is the relationship between them.



One-to-One Relationship

When a single element of an entity is associated with a single element of another entity, it is called a one-to-one relationship.

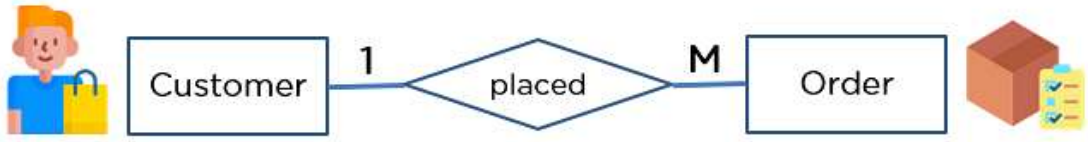
For example, a student has only one identification card and an identification card is given to one person.



One-to-Many Relationship

When a single element of an entity is associated with more than one element of another entity, it is called a one-to-many relationship

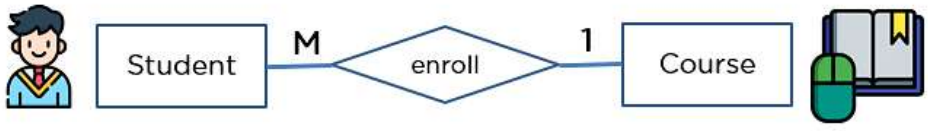
For example, a customer can place many orders, but an order cannot be placed by many customers.



Many-to-One Relationship

When more than one element of an entity is related to a single element of another entity, then it is called a many-to-one relationship.

For example, students have to opt for a single course, but a course can have many students.



Many-to-Many Relationship

When more than one element of an entity is associated with more than one element of another entity, this is called a many-to-many relationship.

For example, you can assign an employee to many projects and a project can have many employees.

How to Draw an ER Diagram?

Below are some important points to draw ER diagram:

- First, identify all the Entities. Embed all the entities in a rectangle and label them properly.
- Identify relationships between entities and connect them using a diamond in the middle, illustrating the relationship. Do not connect relationships with each other.
- Connect attributes for entities and label them properly.
- Eradicate any redundant entities or relationships.
- Make sure your ER Diagram supports all the data provided to design the database.
- Effectively use colors to highlight key areas in your diagrams.

Conclusion

ER Diagram in DBMS is widely used to describe the conceptual design of databases. It helps both users and database developers to preview the structure of the database before implementing the database.