

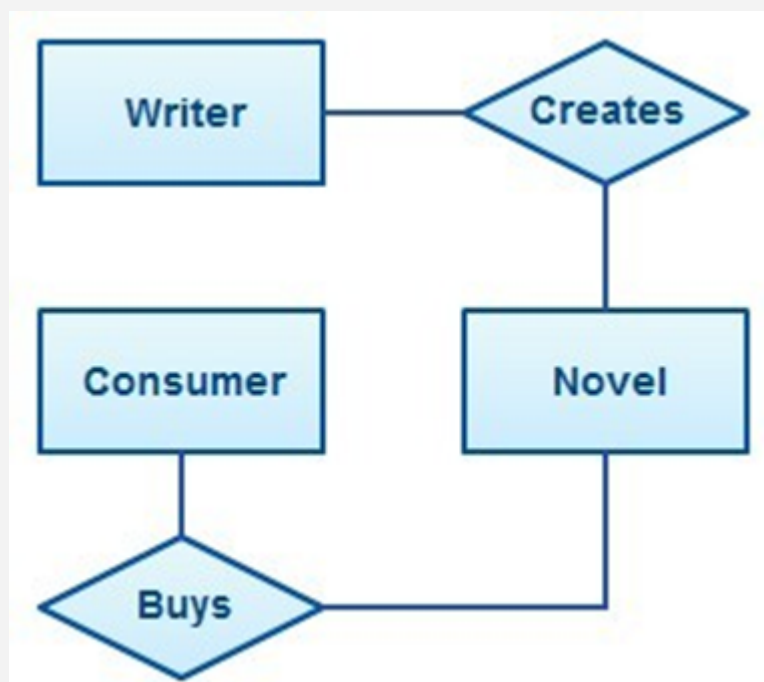
ER MODEL

Introduction to ER Models

What are ER Models

An Entity Relationship Diagram (ERD) is a visual representation of different data using conventions that describe how these data are related to each other.

For example, the elements writer, novel, and consumer may be described using ER diagrams this way:



The elements inside rectangles are called **entities** while the items inside diamonds denote the **relationships** between entities.

ER Diagram

ER-modeling is a data modeling technique used in software engineering to produce a conceptual data model of an information system.

Diagrams created using this ER-modeling technique are called Entity-Relationship Diagrams, or ER diagrams or ERDs.

Entity

It is a collection of objects. An entity is an object that is distinguishable from other objects by a set of attributes. An entity may be an 'object' with a physical existence. This is the basic object of the E-R Model, which is a 'thing' in the real world with an independent existence.

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.

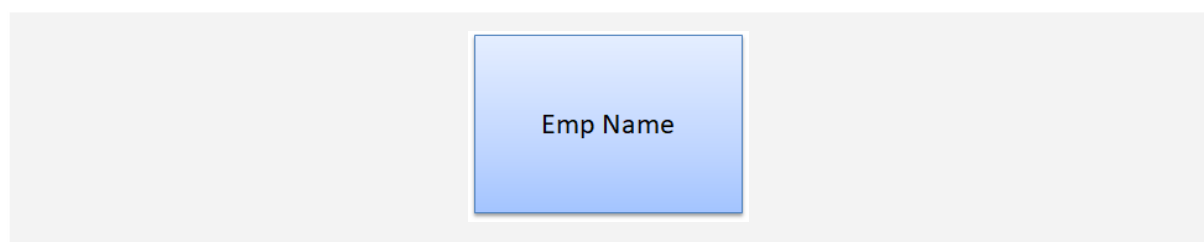
Types of entity

Entities based on their characteristics are classified as follows:

1. Strong Entities
2. Weak Entities
3. Recursive Entities
4. Composite Entities

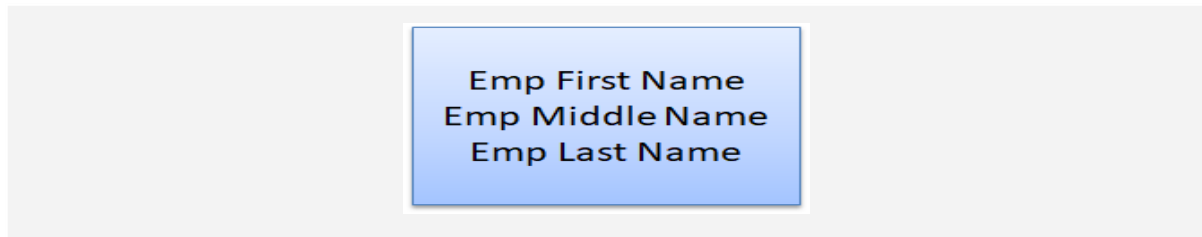
Strong Entities

An entity that exists independently of other entity type



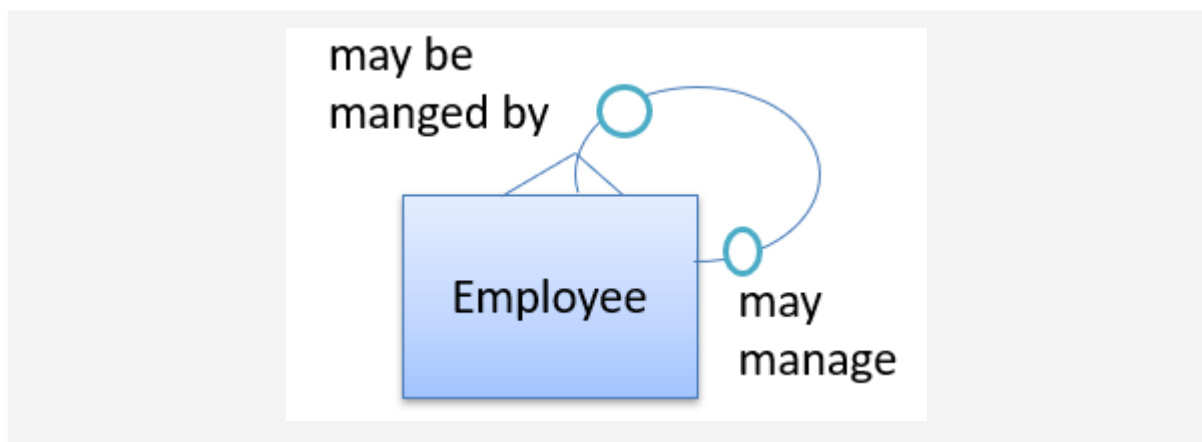
Weak Entities

An entity type whose existence depends on some other entity type.



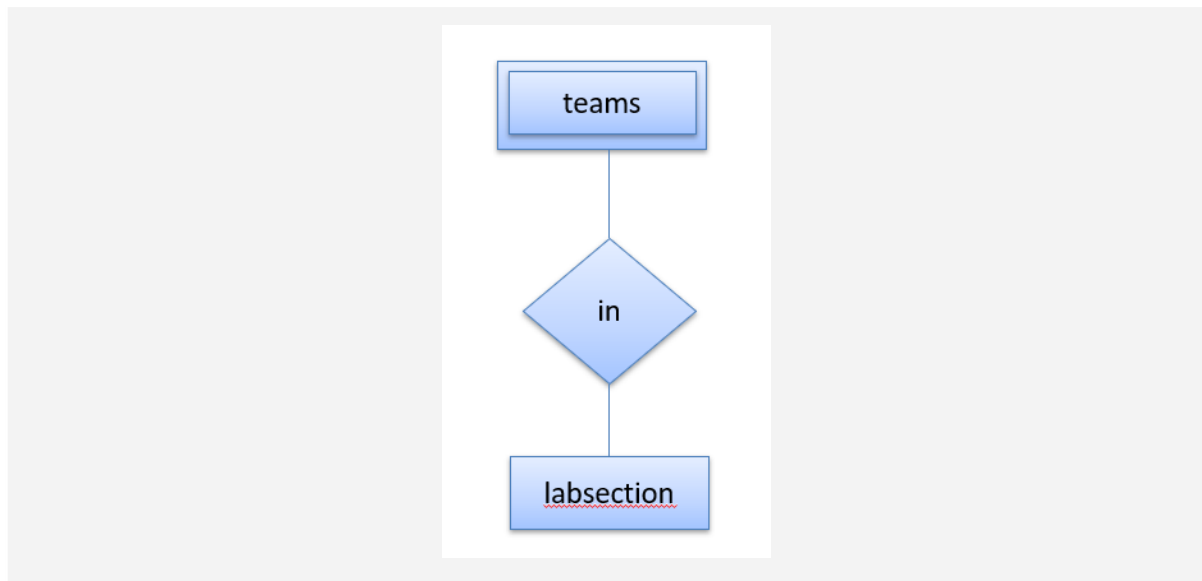
Recursive Entities

A recursive entity is one in which a relation can exist between occurrences of the same entity set. This occurs in a unary relationship. Let us take the example of an employee who is also a manager. But a manager is also an employee, whose details will be held in the employee entity.



Composite Entities

If a Many to Many relationship exists we must create a bridge entity to convert it into 1 to Many. Bridge entity composed of the primary keys of each of the entities to be connected. The bridge entity is known as a composite entity. A composite entity is represented by a diamond shape within a rectangle in an ER Diagram.



Entity Sets

An entity set is a collection of similar types of entities. An entity set may contain entities with attributes 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.

Weak Entity

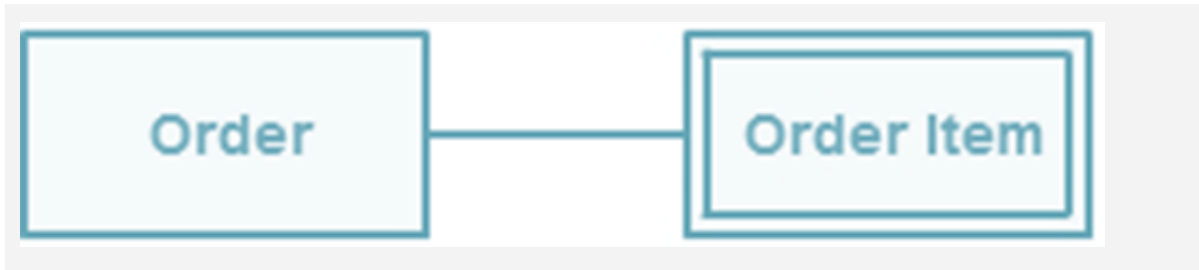
A weak entity is an entity that depends on the existence of another entity.

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.

An entity like order item is a good example for this.

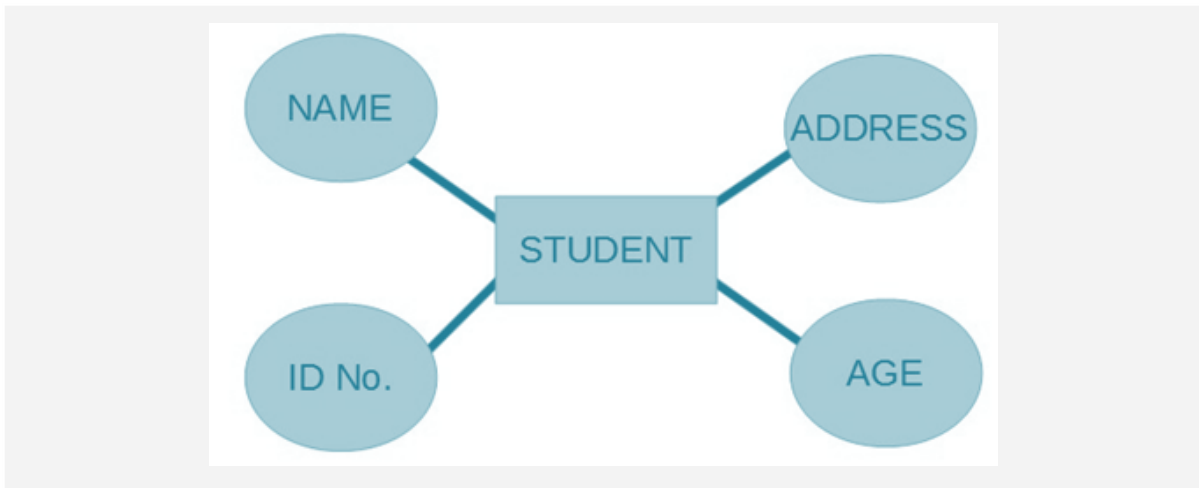
The order item will be meaningless without an order so it depends on the existence of the order.



Attribute

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.



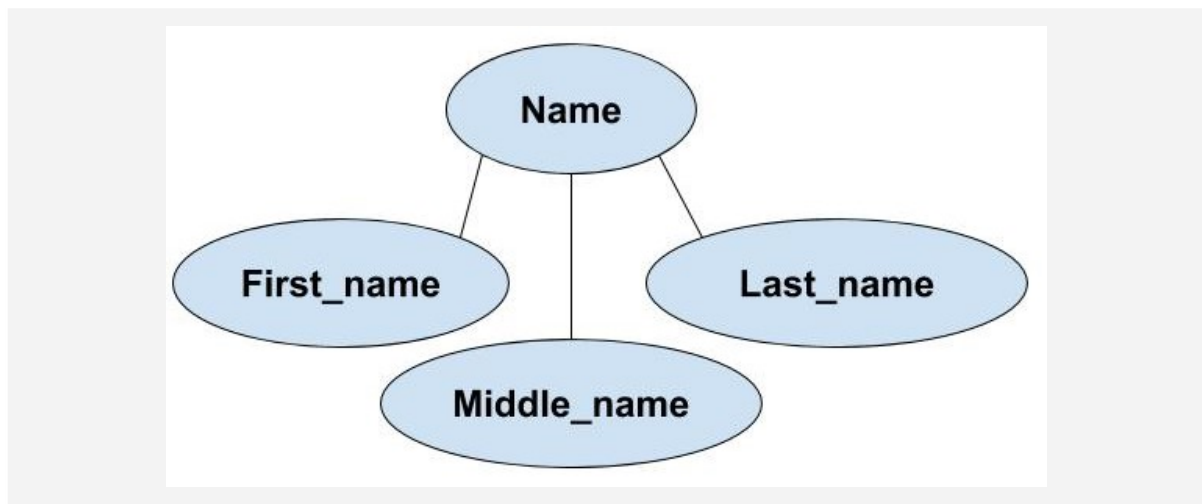
Simple attributes:

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 :

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 eclipses that are connected with an eclipse. Composite attributes are made of more than one simple attribute.

For example, a student's complete name may have first_name and last_name.

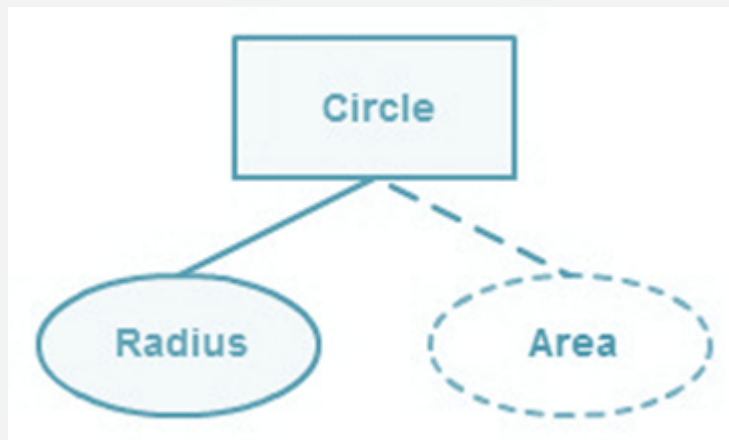
**Derived attribute :**

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.

For example for a circle the area can be derived from the radius.



Multi-valued attribute :

Multi-valued attributes may contain more than one value.

For example, a person can have more than one phone number, email_address, etc.

For example a teacher entity can have multiple subject values.

