

Data Modelling

Satyam Govila

Introduction to ER diagram

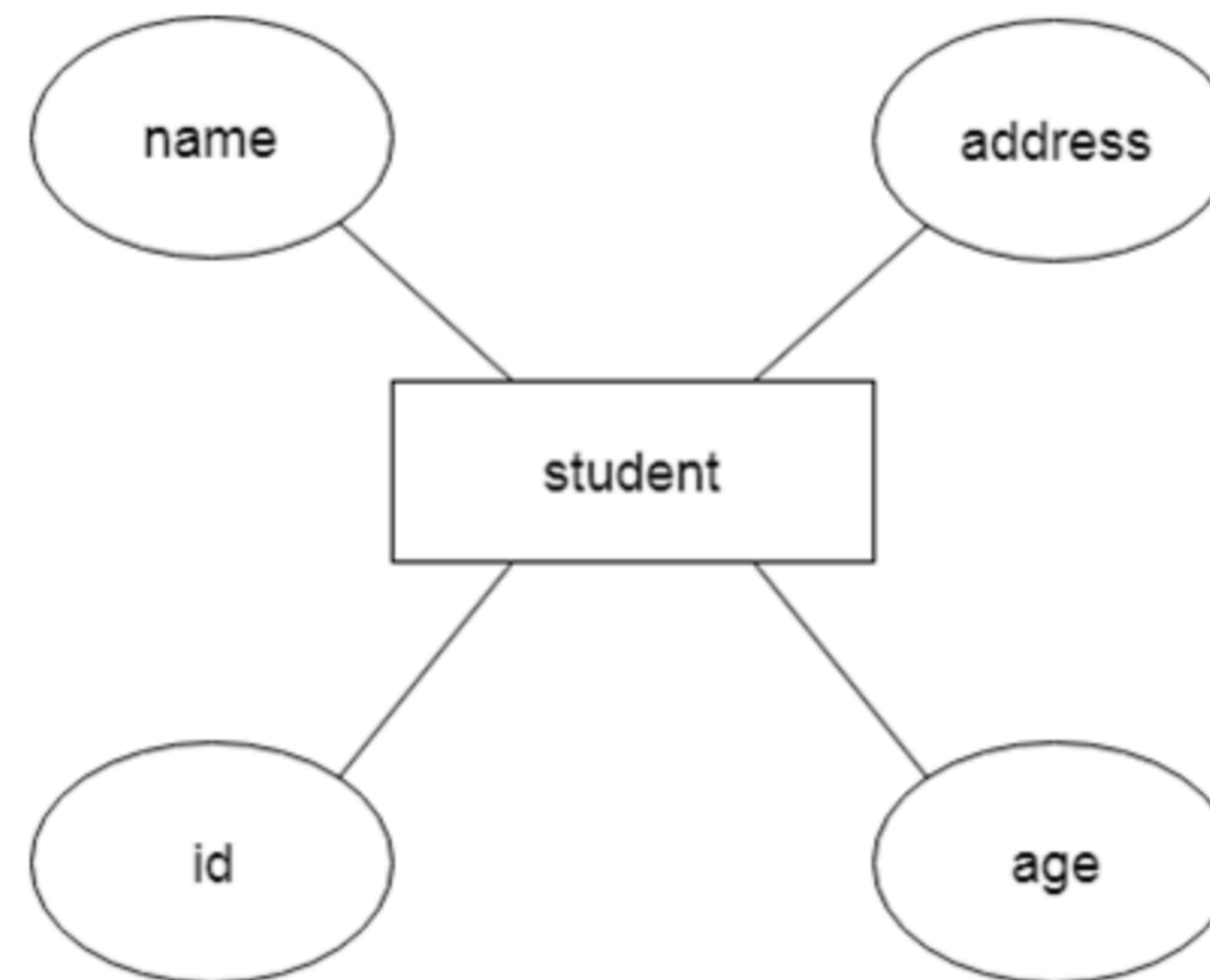
ER : Entity-Relationship

- ER diagram in DBMS stands for entity relationship, which helps us to understand the relationship between the different tables.
- It gives us the high-level understanding of the different data tables based on the attributes
- It also supports various kinds of mapping among the tables.
- It develops a conceptual design for the database. It also develops a very simple and easy to design view of data.

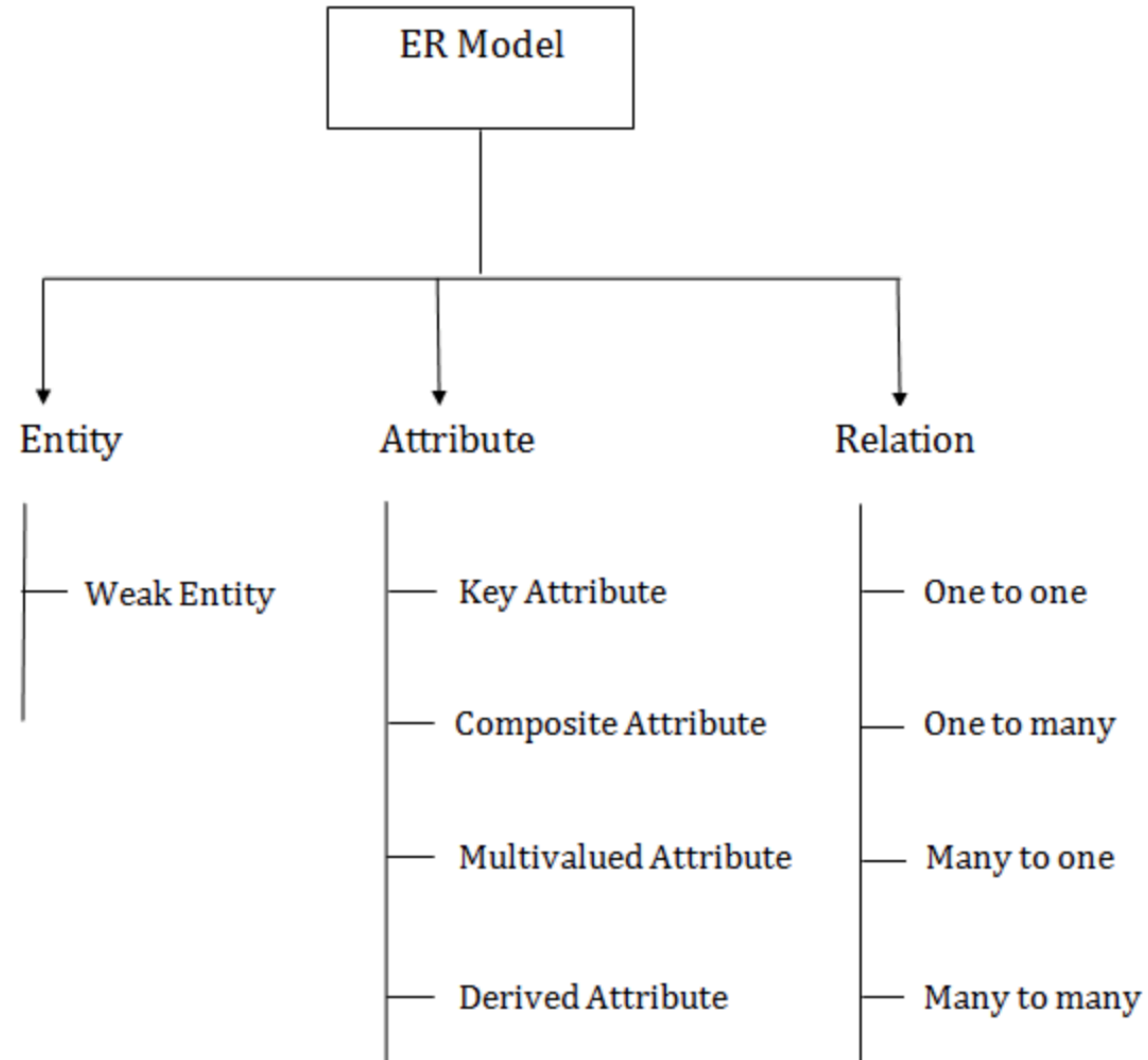
ER diagram : Entity

Entity : It is basically a table which consist of attributes called as columns.

Suppose we have to 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.



Component of ER Diagram

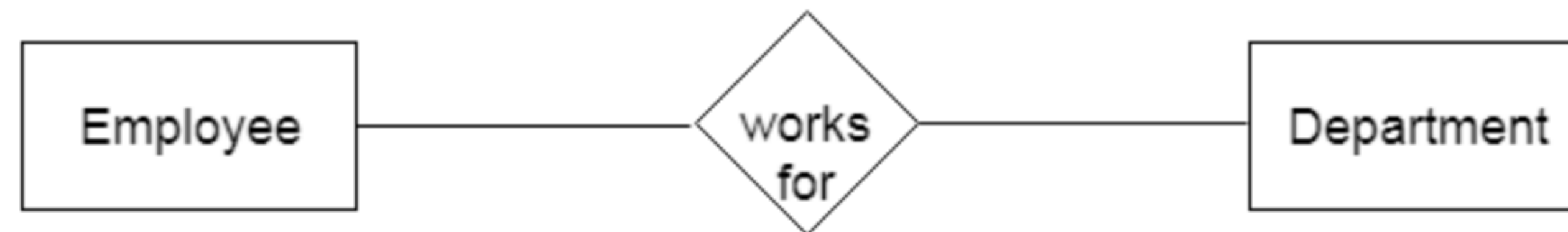


Entity

An entity may be any object, class, person or place.

In the ER diagram, an entity can be represented as rectangles.

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



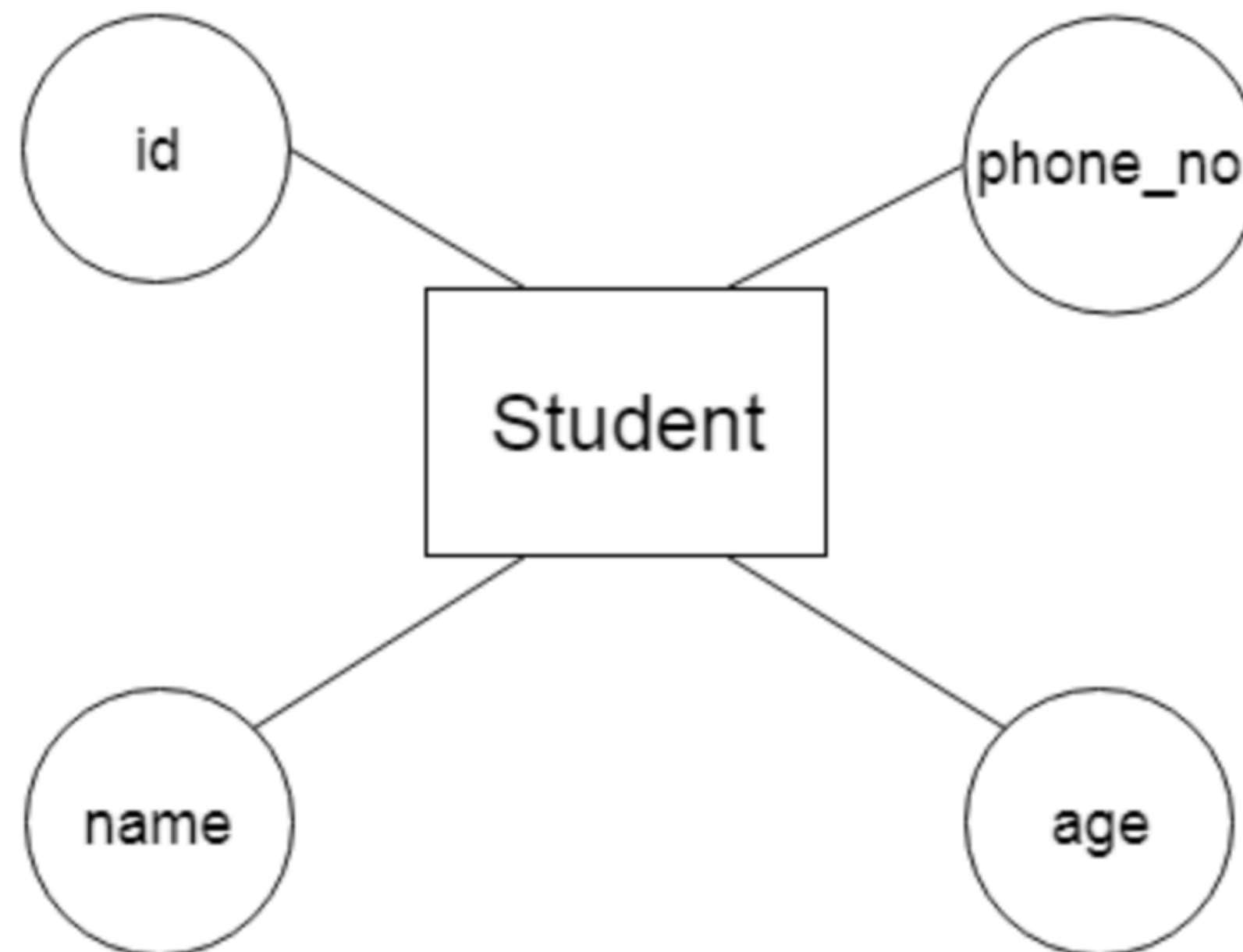
Weak Entity

- 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.



Attributes

- 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.



Types of Attributes

1. 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.

2. Composite Attribute

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

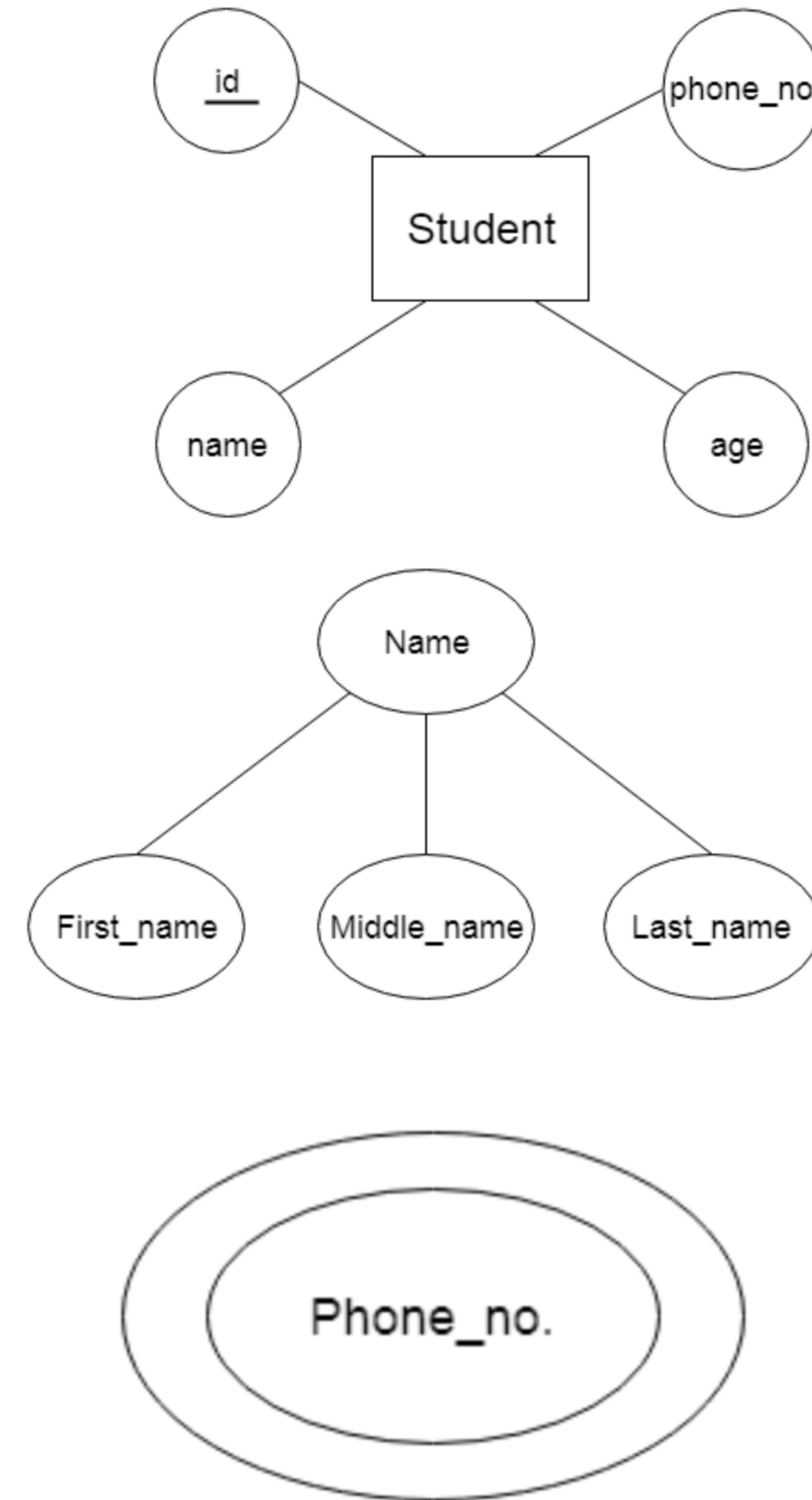
The composite attribute is represented by an ellipse, and those ellipses are connected with an ellipse.

3. Multi-Valued 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.

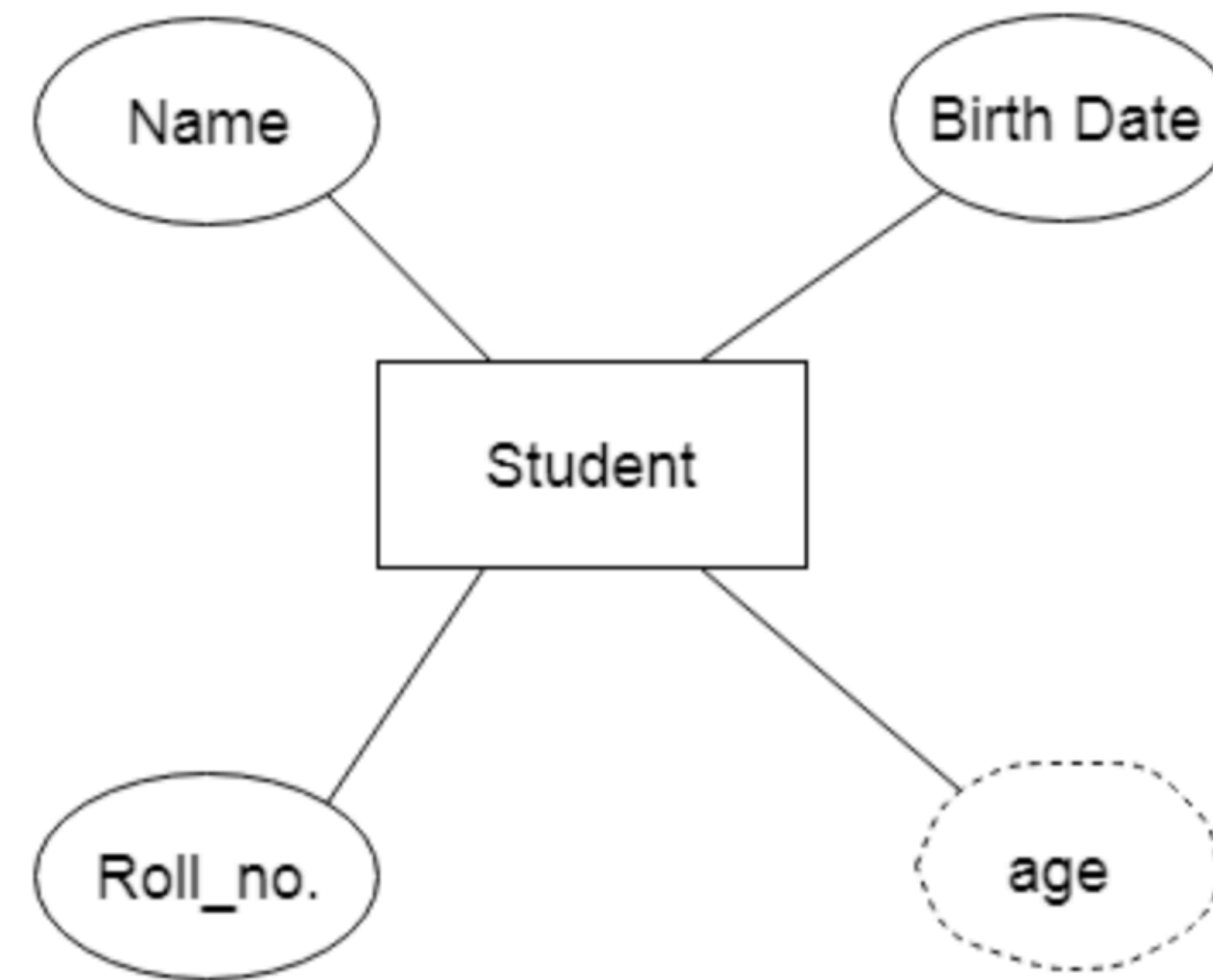


Types of Attributes

4. 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.



Relationship and its types

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



Relationship and its types

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.

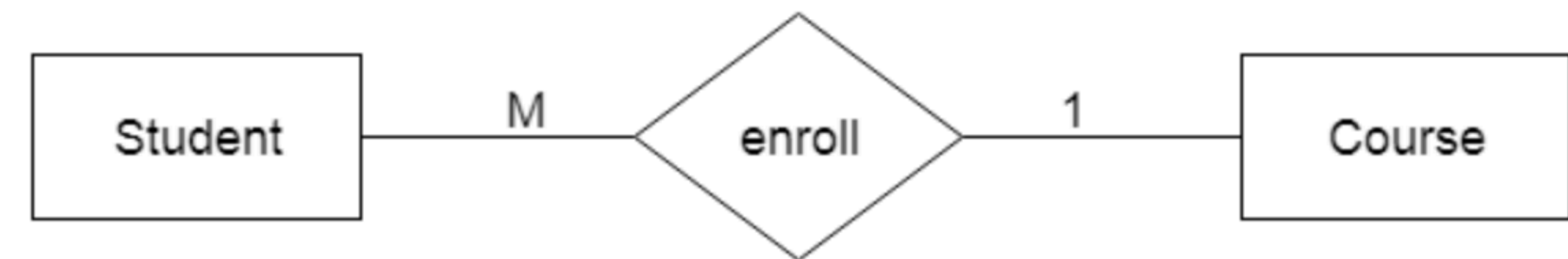


Relationship and its types

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.



Keys in Databases

- Keys play an important role in the relational database.
- It is used to uniquely identify any record or row of data from the table.
- It is also used to establish and identify relationships between tables.
- **For example**, ID is used as a key in the **Student table** because it is unique for each student. In the **PERSON table**, passport_number, license_number, SSN are keys since they are unique for each person.

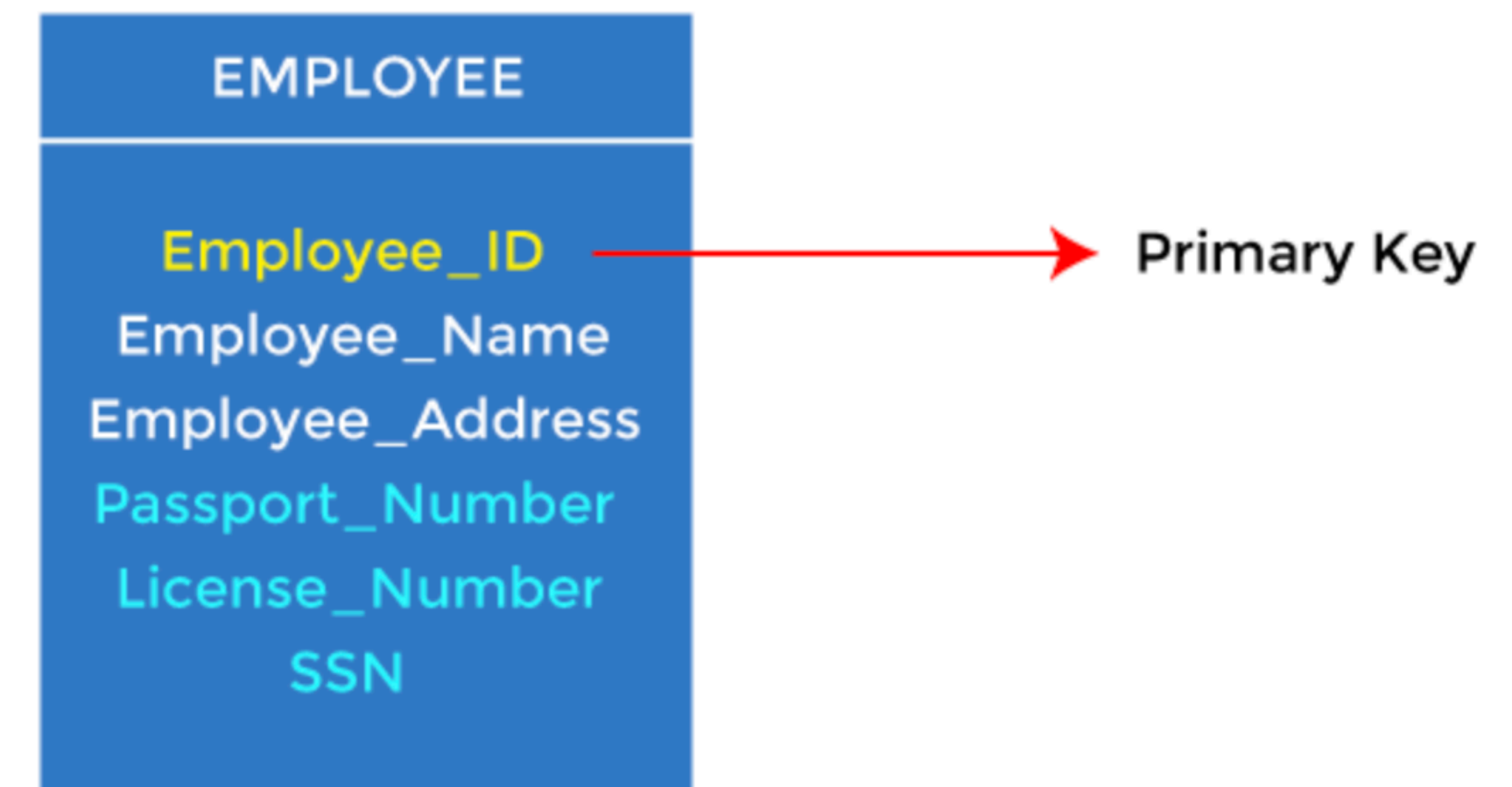
STUDENT
ID
Name
Address
Course

PERSON
Name
DOB
Passport, Number
License_Number
SSN

Types of Keys

1. Primary Key

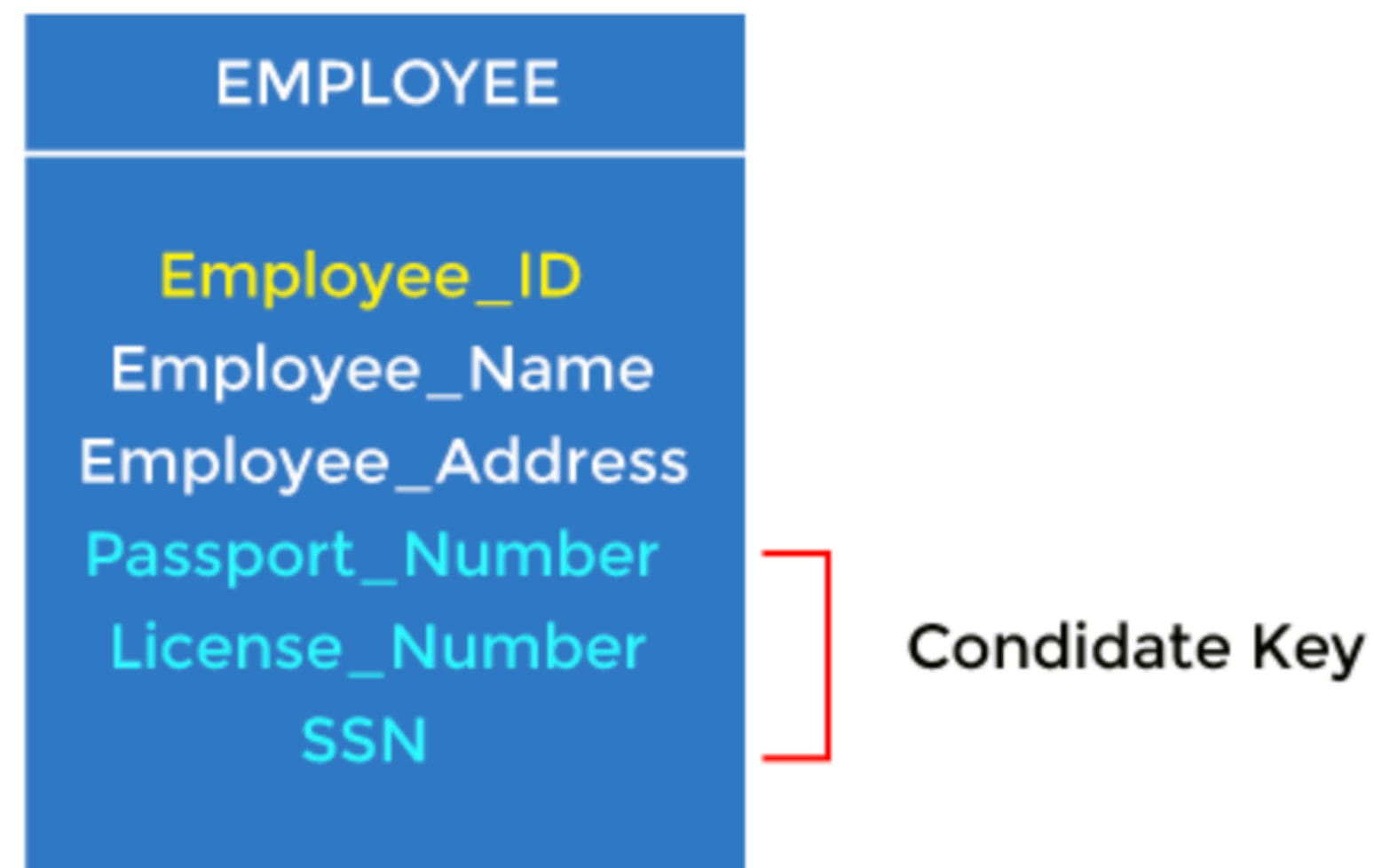
- It is the first key used to identify one and only one instance of an entity uniquely. An entity can contain multiple keys, as we saw in the PERSON table. The key which is most suitable from those lists becomes a primary key.
- In the EMPLOYEE table, ID can be the primary key since it is unique for each employee. We can even select License_Number and Passport_Number as primary keys since they are also unique.
- For each entity, the primary key selection is based on requirements and developers.



Types of Keys

2. Candidate Key

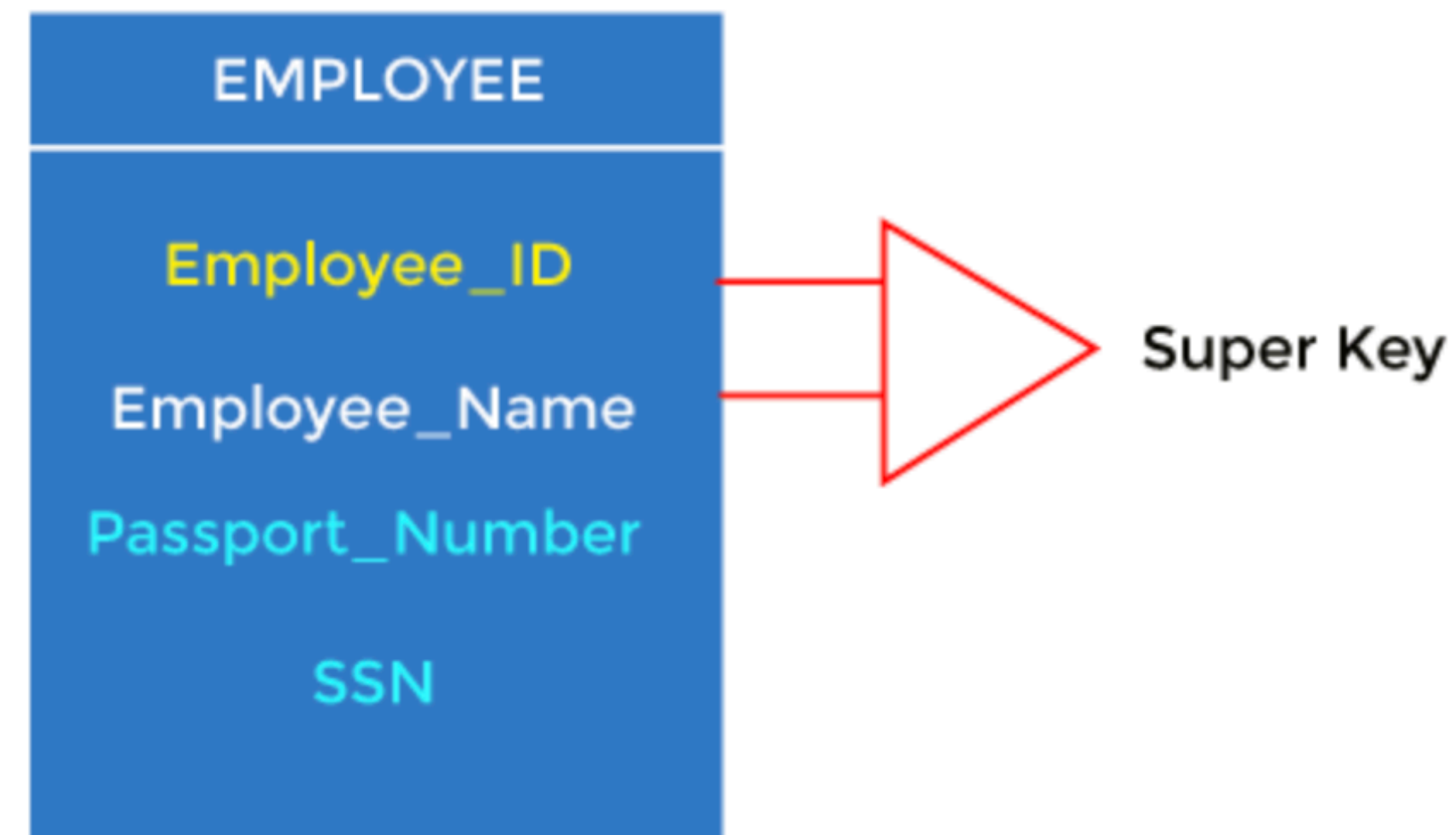
- A candidate key is an attribute or set of attributes that can uniquely identify a tuple.
- **For example:** In the EMPLOYEE table, id is best suited for the primary key. The rest of the attributes, like SSN, Passport_Number, License_Number, etc., are considered a candidate key.



Types of Keys

3. Super Key

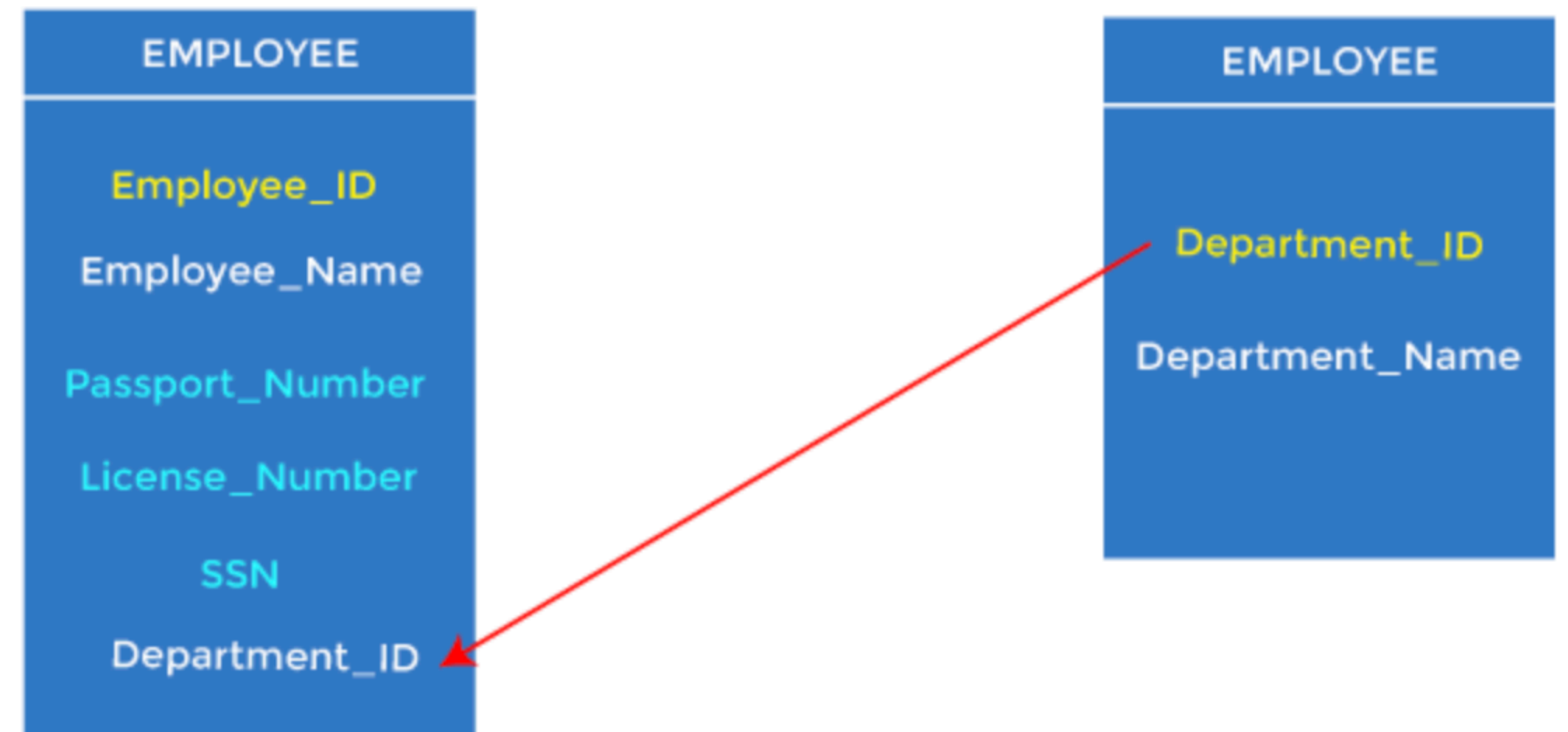
- Super key is an attribute set that can uniquely identify a tuple. A super key is a superset of a candidate key.
- For example: In the above EMPLOYEE table, for(EMPLOYEE_ID, EMPLOYEE_NAME), the name of two employees can be the same, but their EMPLOYEE_ID can't be the same. Hence, this combination can also be a key.



Types of Keys

4. Foreign Key

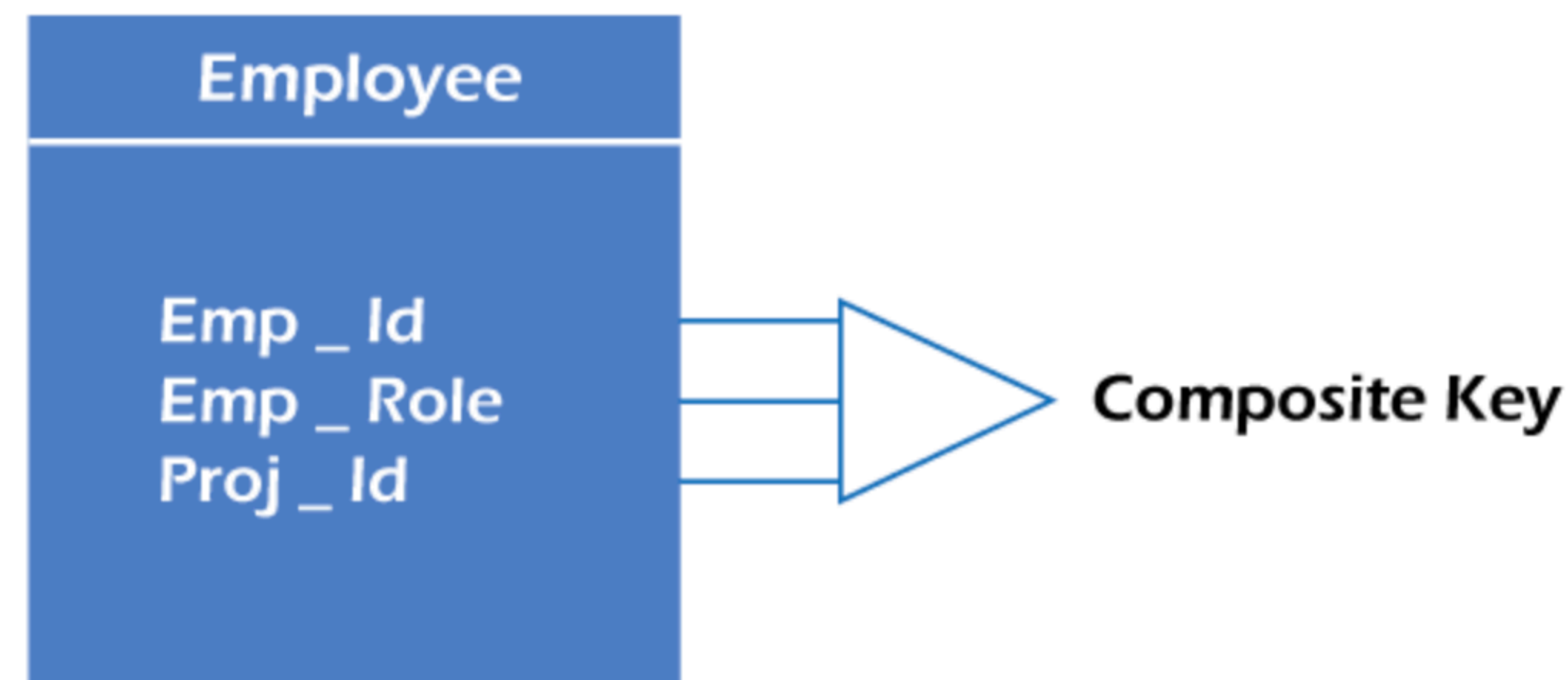
- Foreign keys are the column of the table used to point to the primary key of another table.
- Every employee works in a specific department in a company, and employee and department are two different entities. So we can't store the department's information in the employee table. That's why we link these two tables through the primary key of one table.
- We add the primary key of the DEPARTMENT table, Department_Id, as a new attribute in the EMPLOYEE table.
- In the EMPLOYEE table, Department_Id is the foreign key, and both the tables are related.



Types of Keys

5. Composite Key

- Whenever a primary key consists of more than one attribute, it is known as a composite key. This key is also known as Concatenated Key.
- **For example**, in employee relations, we assume that an employee may be assigned multiple roles, and an employee may work on multiple projects simultaneously. So the primary key will be composed of all three attributes, namely Emp_ID, Emp_role, and Proj_ID in combination. So these attributes act as a composite key since the primary key comprises more than one attribute.



- 1 A 11

- 1 B 12

Any Questions?