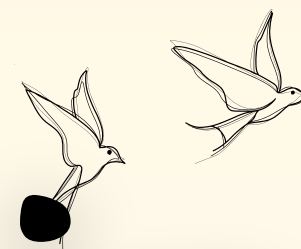




Entity Relationship Modeling





Lesson Objectives

- Students will be able to correctly use symbols to represent meanings in an Entity–Relationship (E-R) diagram.
- Students will be able to correctly define and assign relationships between entities.
- Students will be able to analyze and solve problems that arise from E-R diagrams.
- Students will be able to develop and design E-R diagrams correctly according to database design principles.

Source: Iamsiriwong, O. (2008). Database systems. Bangkok: SE-ED Education.

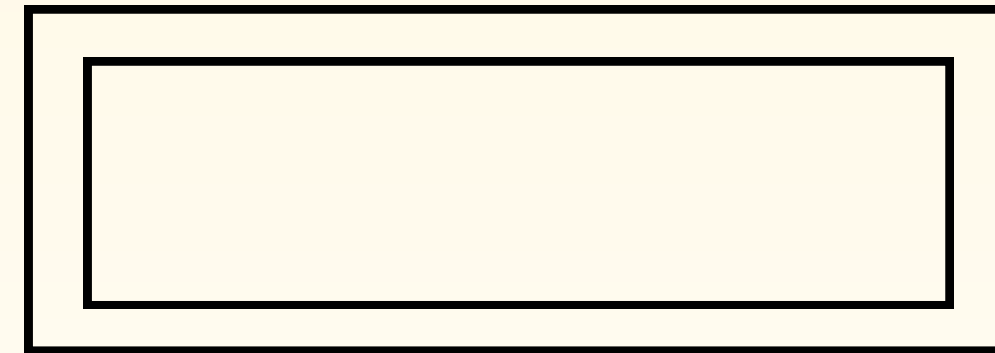


Entity

An Entity is represented by a rectangle symbol in an Entity–Relationship (E-R) diagram. The name of an entity must be a noun, which may represent a place, person, object, or event. Entities are classified into two main types



Strong Entity – A strong entity is an entity that can exist independently and does not rely on other entities. It is represented by a single rectangle. This type of entity is also called a Regular Entity.



Weak Entity – A weak entity is an entity that cannot exist independently. It must be related to or depend on another entity to exist. If the owner (strong) entity is deleted, the weak entity will also be deleted.

Source: Iamsiriwong, O. (2008). Database systems. Bangkok: SE-ED Education.



Attributes...

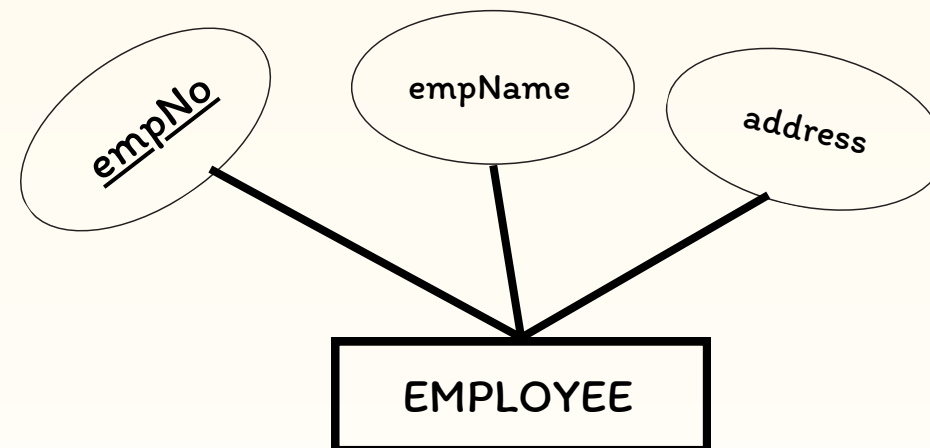
Attribute Domain

An Attribute Domain defines the set of possible values and the data type that an attribute can have. In other words, a domain specifies the range of allowable values for a particular attribute in a database.

The domain helps ensure that the data entered into the attribute is valid and consistent. For example, the attribute Sex may have a domain that allows only the values “M” (Male) or “F” (Female).

This means that no other values are permitted for this attribute.

By defining domains, database designers can control data integrity and reduce data errors in the system.

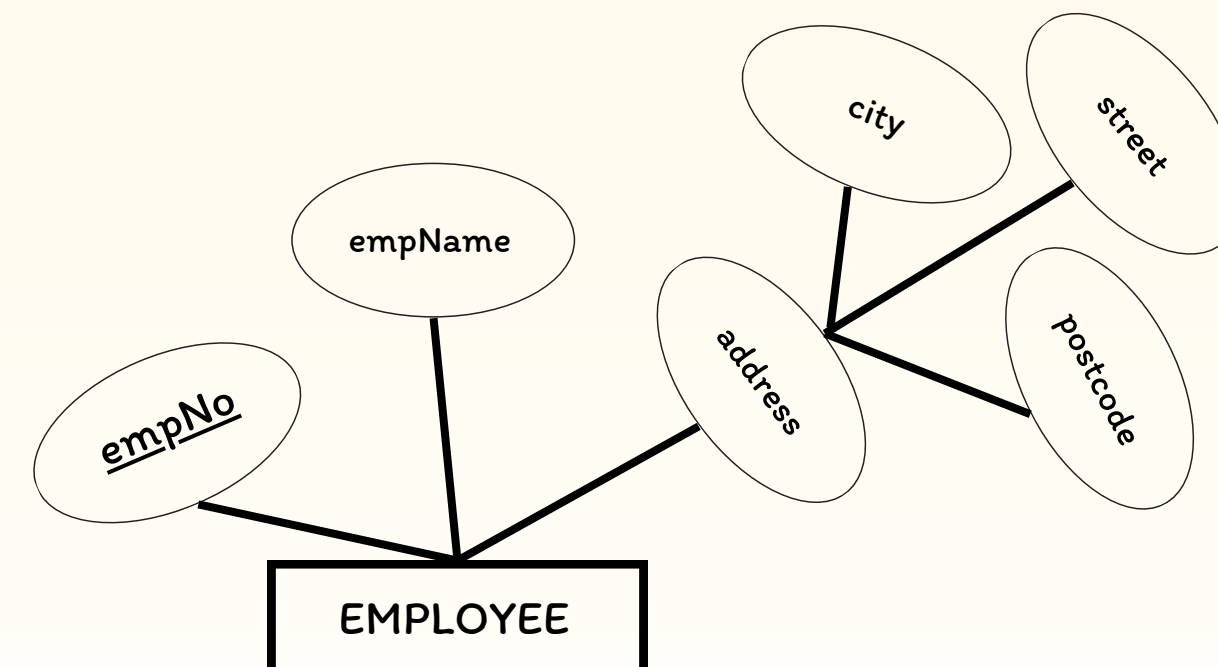


Simple Attribute and Composite Attribute

A **Simple Attribute** is an attribute that has only one component and cannot be divided into smaller parts. It represents a single, indivisible piece of information about an entity. Examples of simple attributes include Age and Sex, because these attributes contain only one value and cannot be further separated.

A **Composite Attribute**, on the other hand, is an attribute that is composed of multiple sub-attributes. It can be divided into smaller meaningful components that represent different parts of the same attribute.

For example, the attribute Address may consist of several sub-attributes such as Street and City.



Source: Iamsiriwong, O. (2008). Database systems. Bangkok: SE-ED Education.



Attributes...

Single-Valued Attribute and Multivalued Attributes

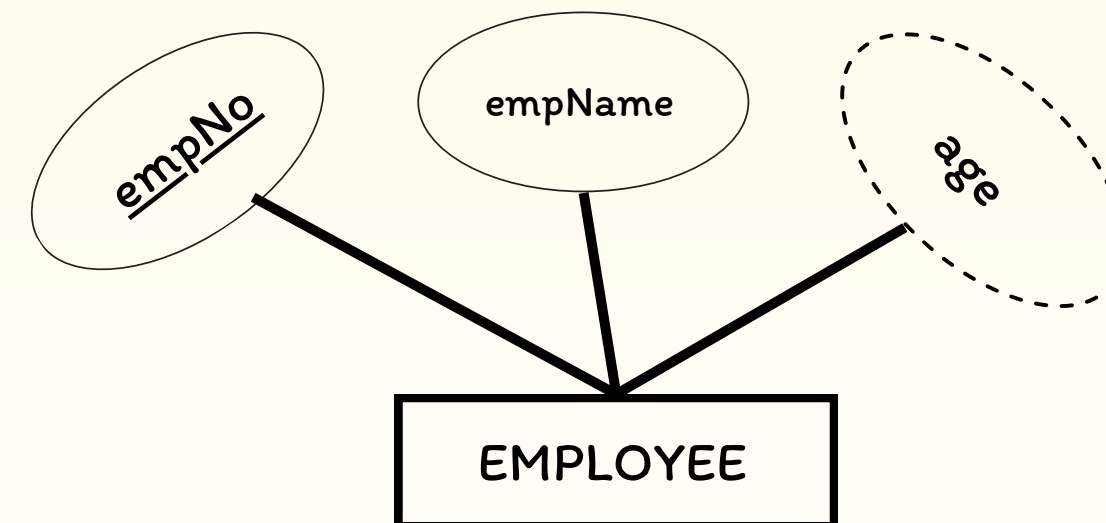
A **Simple Attribute** is an attribute that has only one component and cannot be divided into smaller parts. It represents a single, indivisible piece of information about an entity. Examples of simple attributes include Age and Sex, because these attributes contain only one value and cannot be further separated.

A **Composite Attribute**, on the other hand, is an attribute that is composed of multiple sub-attributes. It can be divided into smaller meaningful components that represent different parts of the same attribute. For example, the attribute Address may consist of several sub-attributes such as Street and City.

Derived Attribute

A **Derived Attribute** is an attribute whose value is **calculated or derived from other attributes** in the database. It does not need to be stored directly because it can be obtained by performing calculations using existing data. Derived attributes help reduce data redundancy and improve database efficiency.

For example, the attribute **Age** can be derived from the **Date of Birth**, and **Salary** may be calculated based on working hours or payment rates. In an E-R diagram, a derived attribute is represented by a dashed oval. A derived attribute is also commonly referred to as a **Computed Attribute**.



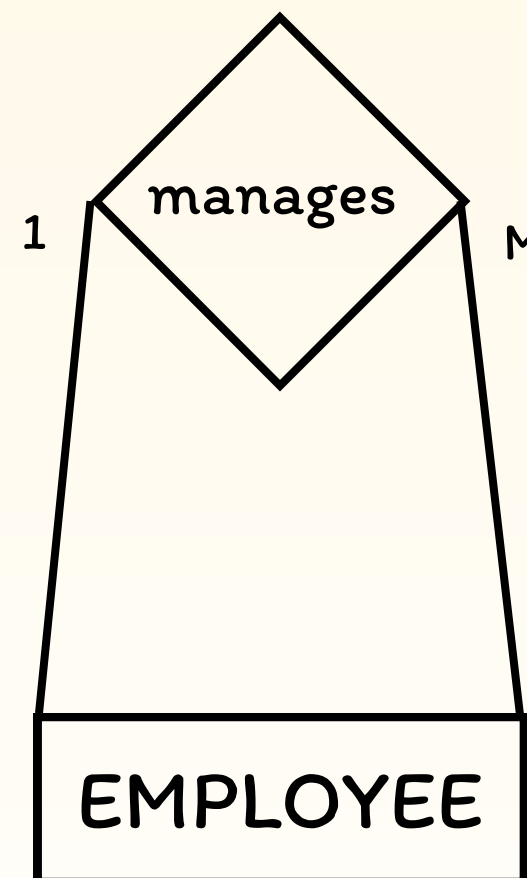
Source: Iamsiriwong, O. (2008). Database systems. Bangkok: SE-ED Education.



Relationship Degree...

Unary Relationship

A Unary Relationship is a relationship that occurs within a single entity set in an Entity–Relationship (ER) model. For example, a Manager manages Employee relationship can exist within the same entity type, Employee. This type of relationship is also known as a recursive relationship, because the entity relates to itself.



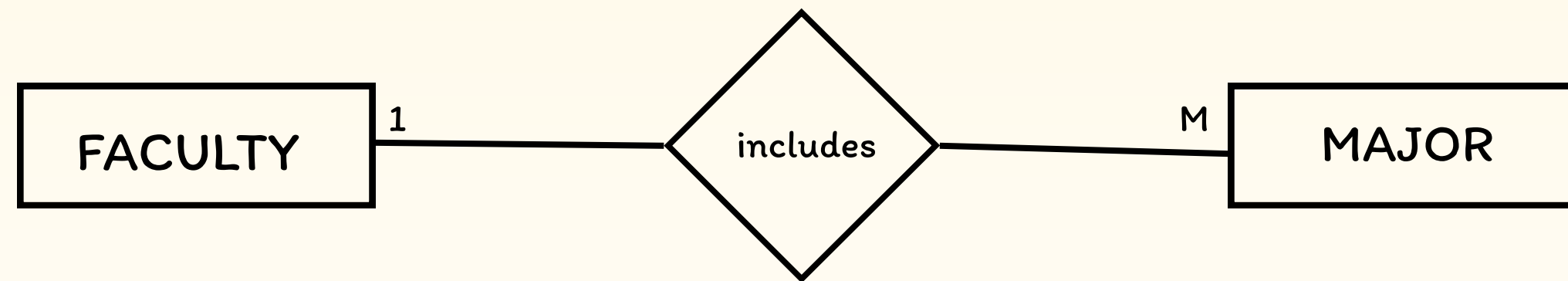
Source: Iamsiriwong, O. (2008). Database systems. Bangkok: SE-ED Education.



Relationship Degree...

Binary Relationship

A Binary Relationship is a relationship that involves two distinct entity sets in an Entity–Relationship (ER) model. It represents how two different entities are associated with each other. For example, a Faculty consists of many Departments relationship illustrates a common binary relationship.



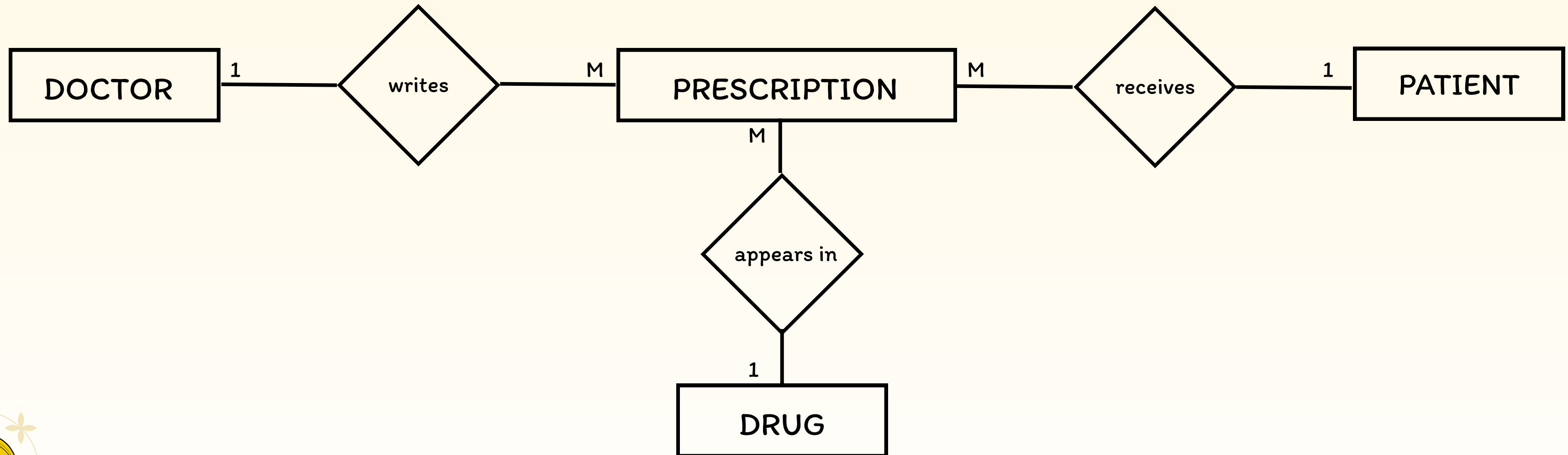
Source: Iamsiriwong, O. (2008). Database systems. Bangkok: SE-ED Education.



Relationship Degree...

Ternary Relationship

A Ternary Relationship is a relationship that involves three different entity sets in an Entity–Relationship (ER) model. It represents interactions where all three entities must participate simultaneously. For example, a Doctor writes a Prescription for a Patient, which includes one or more Drugs specified in the prescription.



Source: Iamsiriwong, O. (2008). Database systems. Bangkok: SE-ED Education.