



# Data Modeling

---

## Database Design

Department of Computer Engineering

Sharif University of Technology

Maryam Ramezani [maryam.ramezani@sharif.edu](mailto:maryam.ramezani@sharif.edu)



## The company is organized into departments:

- ❑ Each **department** has a **unique name**, a **unique number**, and a **particular employee who manages the department**. We keep track of the **start date** when that employee began **managing the department**. A department may have **several locations**.
- ❑ A department **controls** a number of **projects**, each of which has a **unique name**, a **unique number**, and a **single location**.
- ❑ The database will store each **employee's name**, **Social Security number**, **address**, **salary**, **sex** (gender), and **birth date**. An employee **is assigned to one department**, but **may work on several projects**, which are not necessarily controlled by the same department. It is required to keep track of the current number of hours per week that an employee works **on each project**, as well **as the direct supervisor of each employee** (who is another employee).
- ❑ The database will keep track of the **dependents** of each employee for insurance purposes, including each **dependent's first name**, **sex**, **birth date**, and **relationship** to the **employee**.



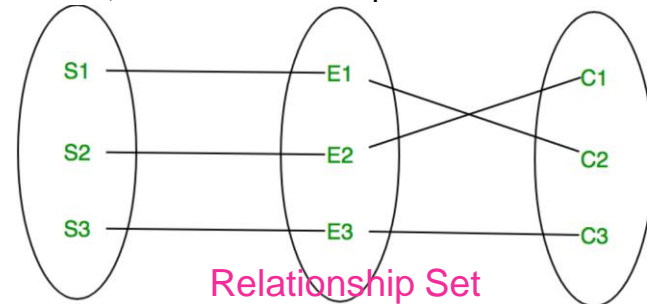
- ☐ A relationship indicated how one or more entity classes interact with one and another.
- ☐ Each entity plays a role in a relationship.

- ❑ Relationship Type: Association among two or more entities.
  - In ER diagram, the relationship type is represented by a **diamond** and connecting the entities with lines

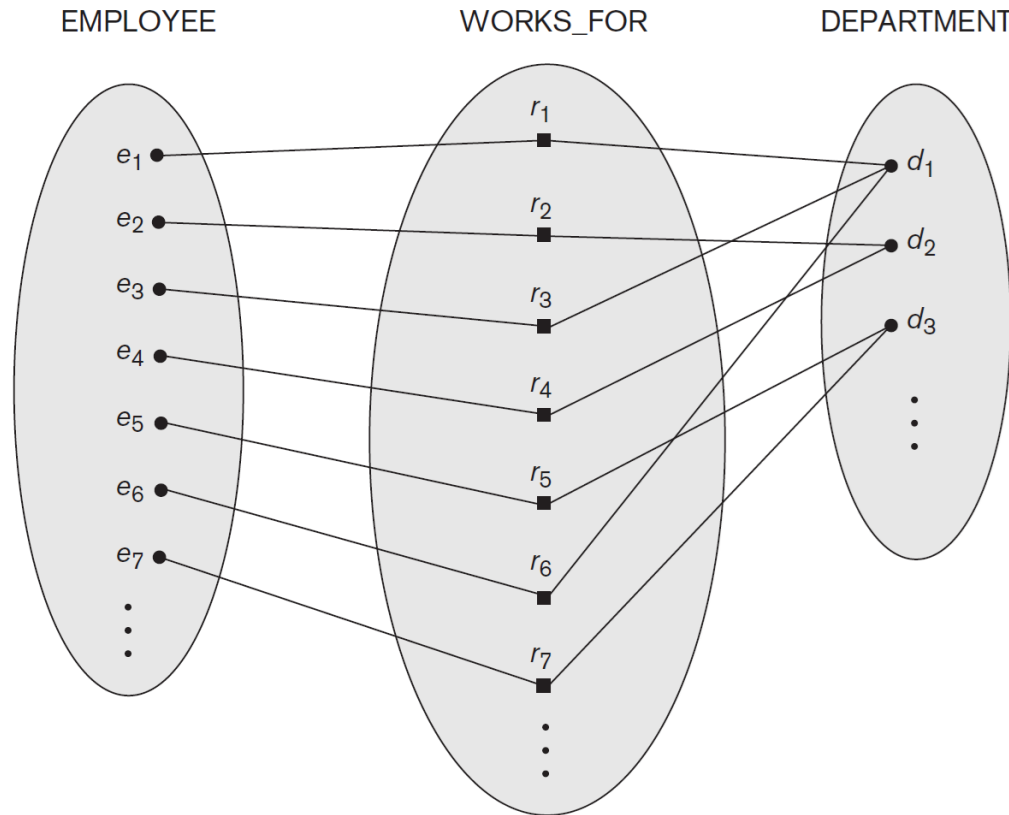


- ❑ Relationship Set: A set of relationships of the same type is known as a relationship set.

- An  $n$ -ary relationship set  $R$  relates  $n$  entity sets  $E_1 \dots E_n$ ; each relationship in  $R$  involves entities  $e_1, \dots, e_n$ .

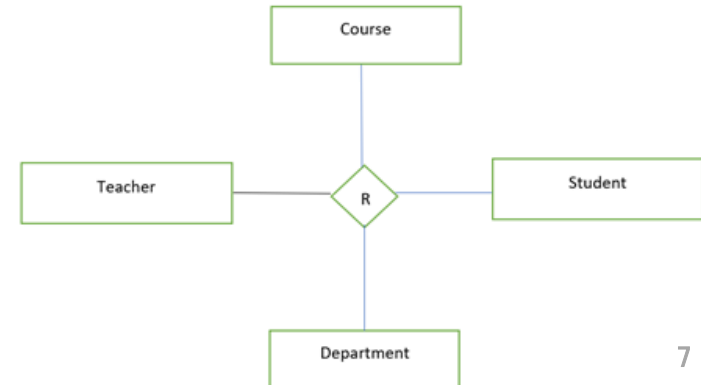
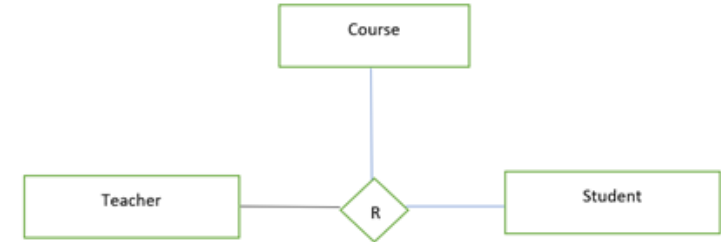
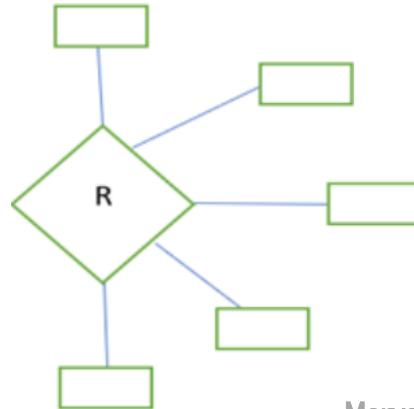
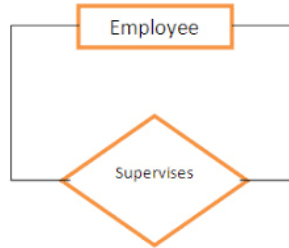


# Relationship Set



- ❑ Degree of Relationship Type: The number of participating entity types

- Unary: degree one
- Binary: degree two
- Ternary: degree three
- Quaternary: degree four
- N-ary



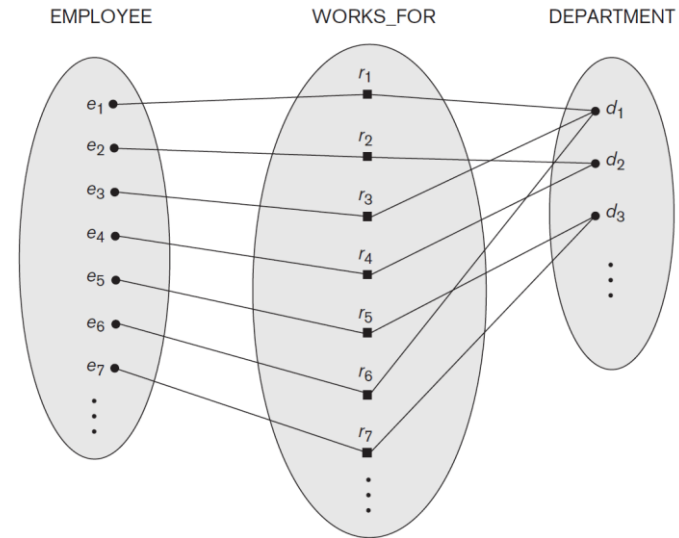
# Constraints on Binary Relationship Types



- Relationship types usually have certain constraints that limit the possible combinations of entities that may participate in the corresponding relationship set. These constraints are determined from the miniworld situation that the relationships represent.

- We can distinguish two main types of binary relationship constraints:

- Cardinality ratio
- Participation

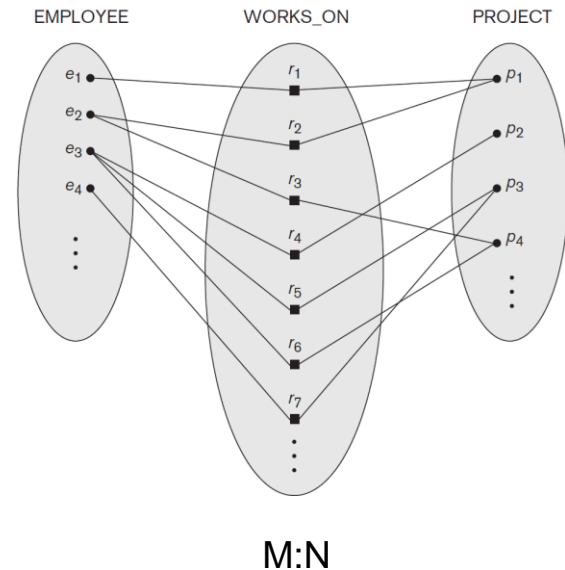
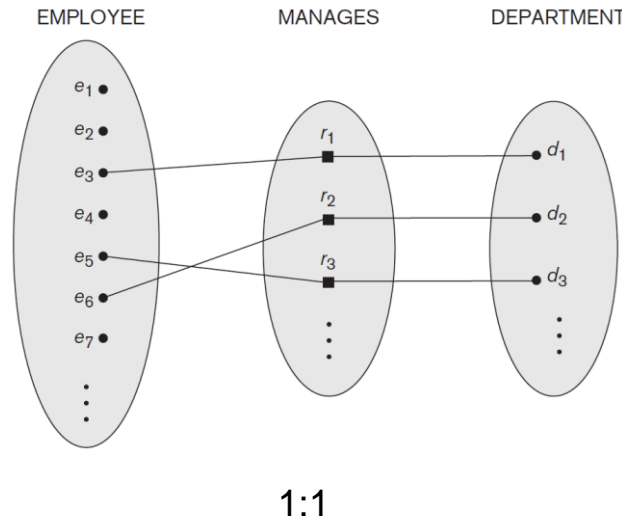


# Cardinality Ratios



- ❑ **Maximum number** of relationship instances that an entity can participate in.
- ❑ The possible cardinality ratios for binary relationship types are:

- 1:1
- M:N
- 1:N





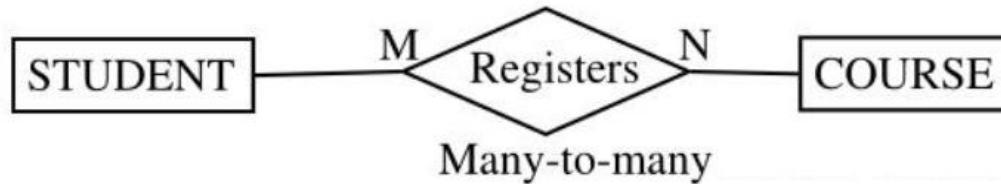
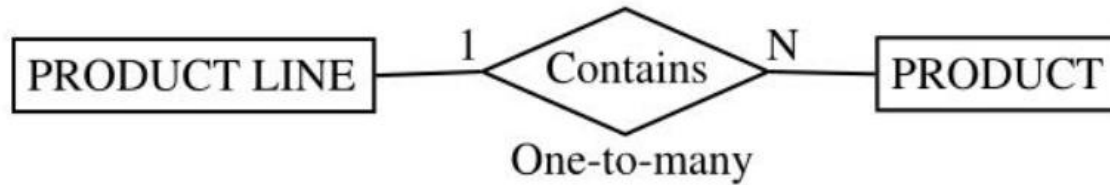
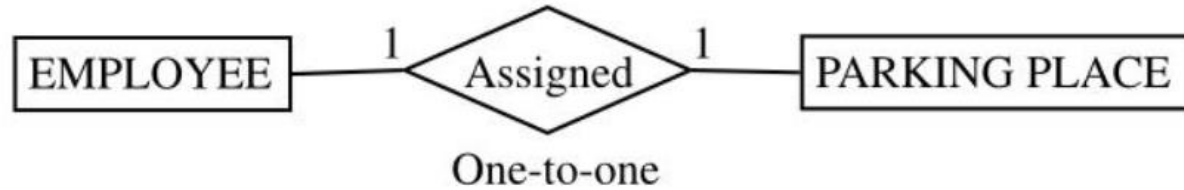


## ❑ Whether the existence of an entity depends on its being related to another entity via the relationship type.

- **Total (existence dependency):**
  - If a company policy states that every employee must work for a department, then an employee entity can exist only if it participates in at least one WORKS\_FOR relationship instance.
  - Meaning that every entity in the total set of employee entities must be related to a department entity via WORKS\_FOR.
  - ER: double line connecting the participating entity type to the relationship. A minimum of one.

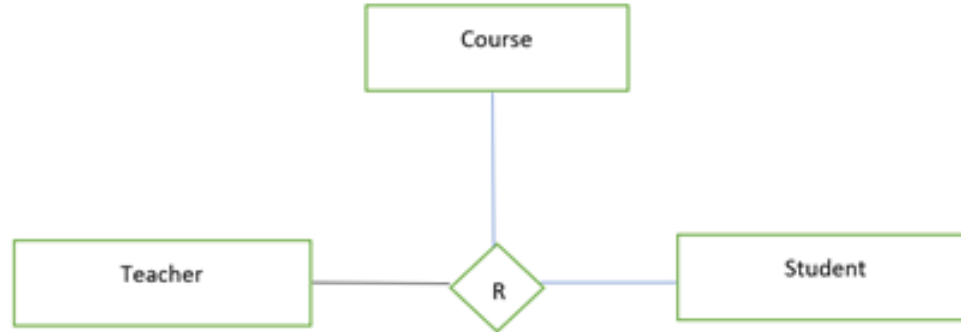


- **Partial**
  - We do not expect every employee to manage a department, so the participation of EMPLOYEE in the most one department and a department can have at most one manager.
  - Meaning that some or part of the set of employee entities are related to some department entity via MANAGES, but not necessarily all.
  - ER: single line connecting the participating entity type to the relationship. No minimum.





- Find cardinality:

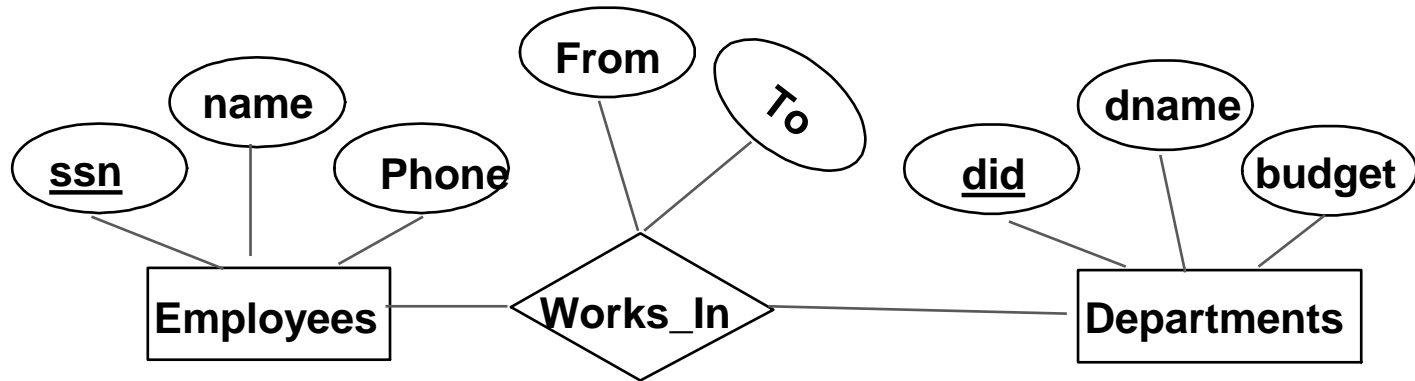


- 1:M:N

# Attributes of Relationship Types

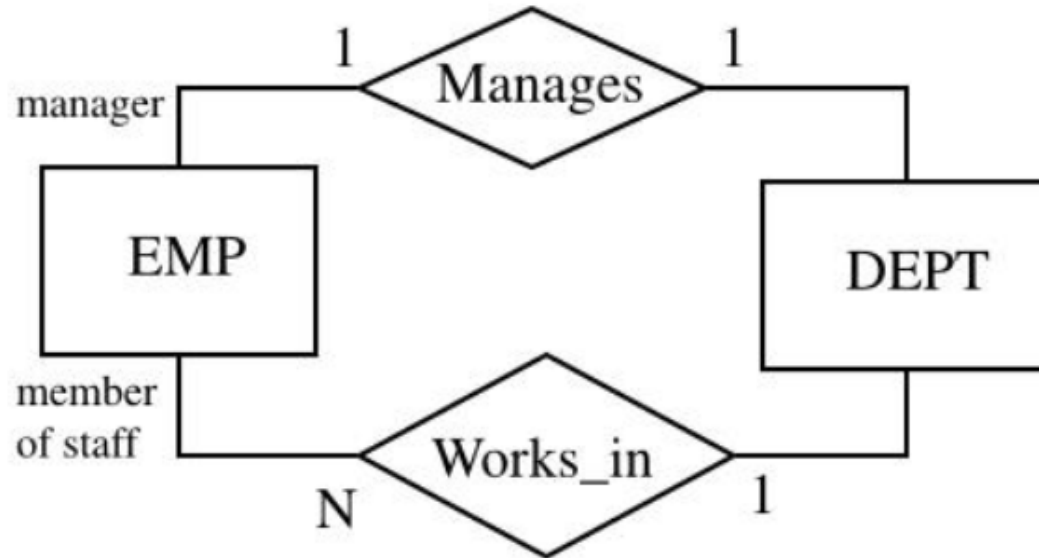


- Relationships can also have **attributes** associated to them. Generally it is not recommended to give attributes to the relationships if not required because while converting the ER model into Relational model, things may get complex and we may require to create a separate table for representing the relationship.

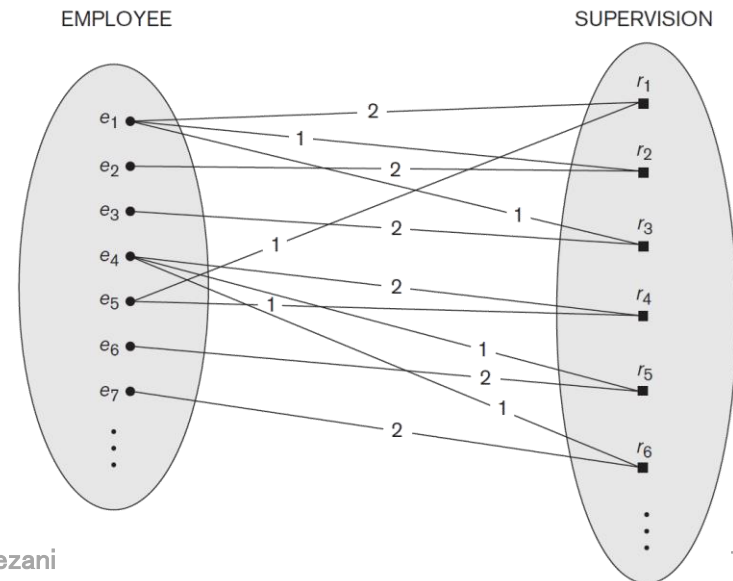
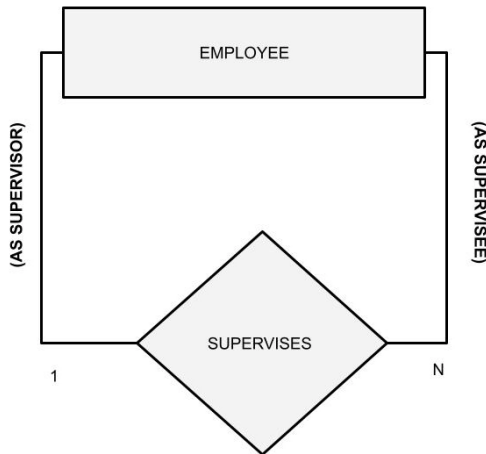




- ❑ Each entity type that participates in a relationship type plays a particular role in the relationship. Role names may also be used when two entity classes are associated through more than one relationships



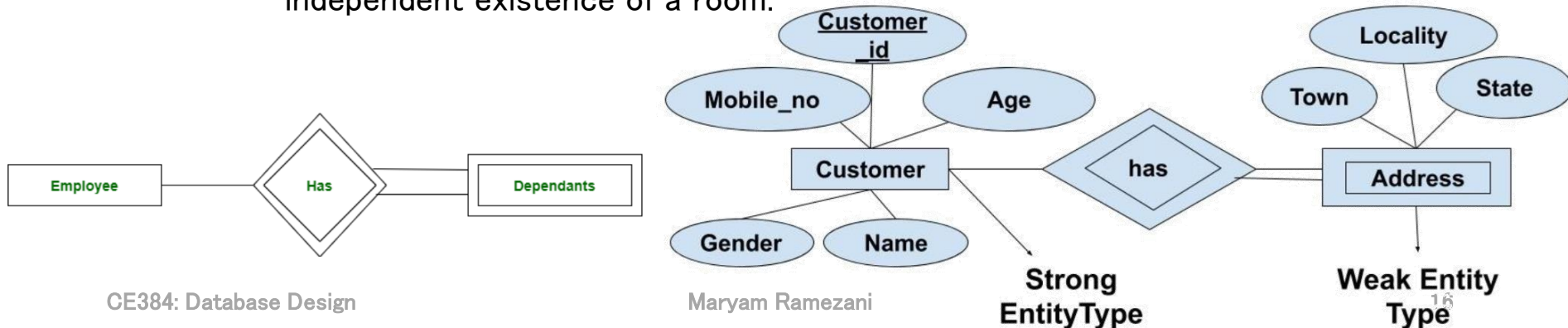
- ❑ In some cases the same entity type participates more than once in a relationship type in different roles.
- The SUPERVISION relationship type relates an employee to a supervisor, where both employee and supervisor entities are members of the same EMPLOYEE entity set.
  - A recursive relationship SUPERVISION between EMPLOYEE in
    - (1) supervisor role
    - (2) supervisee role



# Types of Entity Type



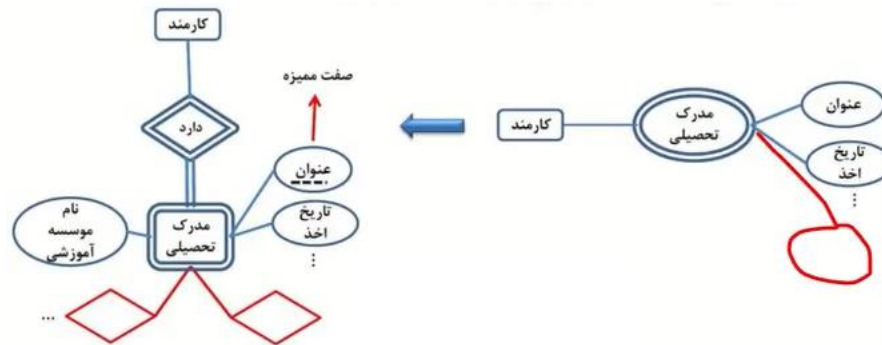
- ❑ **Strong Entity Type:** Has a **key attribute** which helps in **identifying each entity uniquely**. It is represented by a **rectangle** in ER model.
- ❑ **Weak Entity Type:** **Doesn't have a key attribute**. Weak entity type **can't be identified on its own**. It depends upon some other strong entity for its distinct identity. It is represented by a **double outlined rectangle** in ER model.
  - Relationship between the weak entity type and its identifying strong entity type is called **identifying relationship** and it is represented by a **double diamond**.
    - Example: There can be a room only if building exists. There can be no independent existence of a room.



# Types of Entity Type



- ❑ The identifying relationship is **many-to-one** from the weak entity set to the identifying entity set, and the **participation of the weak entity set in the relationship is total**.
- ❑ A weak entity set **can participate in relationships** other than the identifying relationship. Here we use weak entity instead of multivalued attribute.





# Types of Entity Type



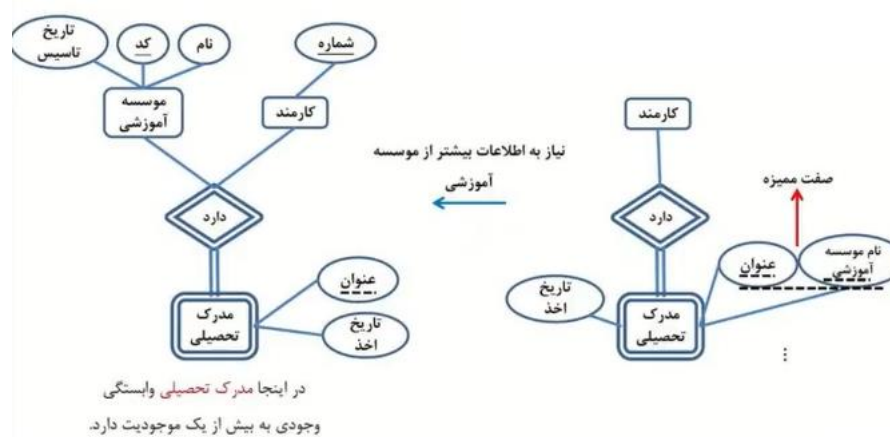
- ❑ A weak entity set may participate as owner in an identifying relationship with another weak entity set.



# Types of Entity Type



- It is also possible to have a weak entity set with more than one identifying entity set.

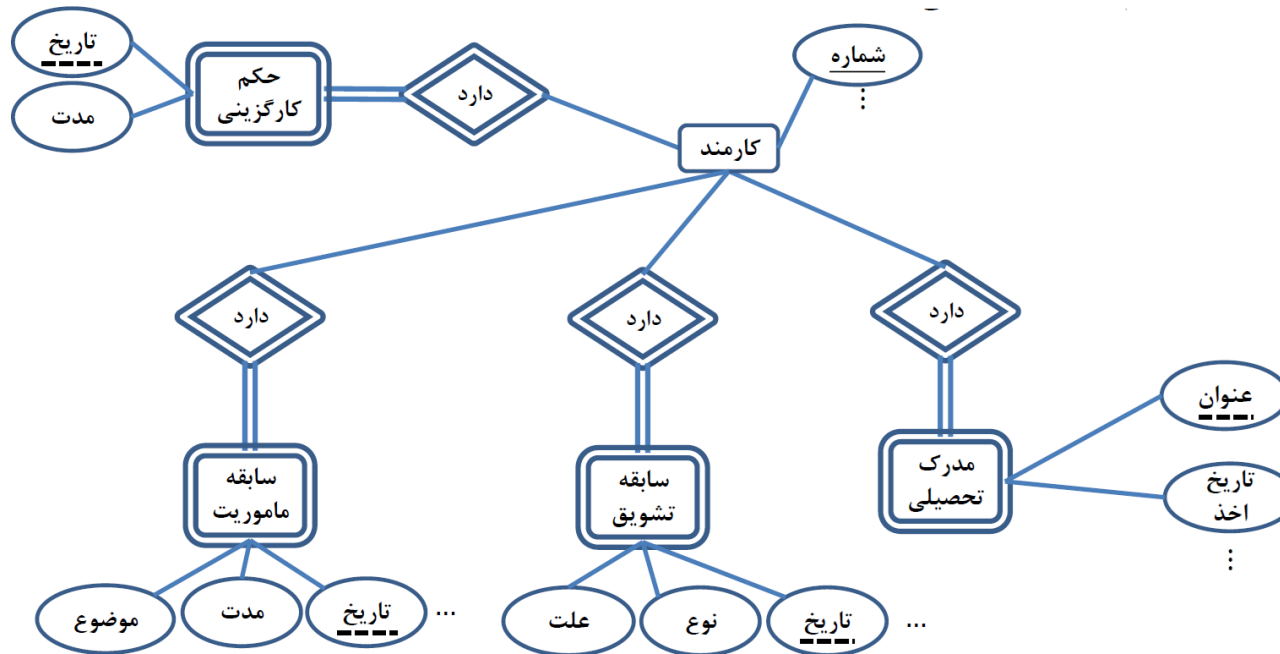


- A particular weak entity would then be identified by a combination of entities, one from each identifying entity set. The key attribute the weak entity set would consist of the union of the key attributes of the identifying entity sets, plus the discriminator of the weak entity set.

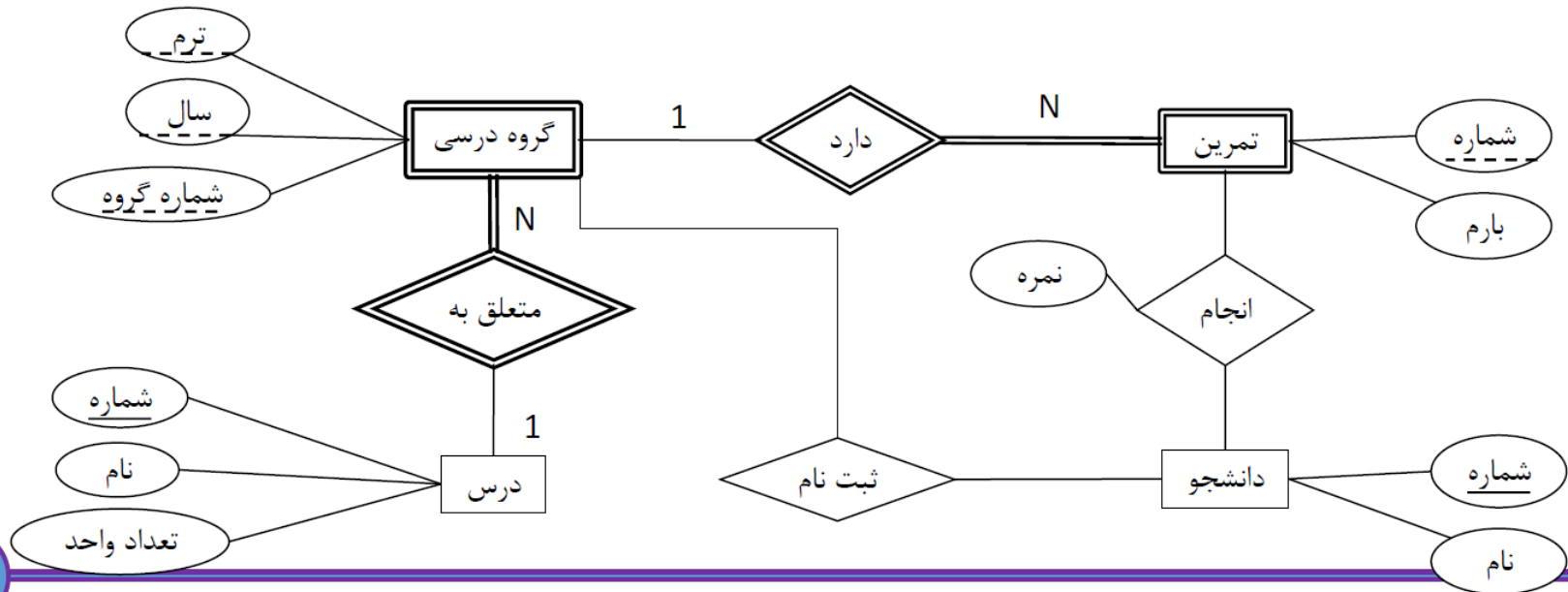
# Types of Entity Type



- Iterative in time.



## Weak Entity of a weak entity



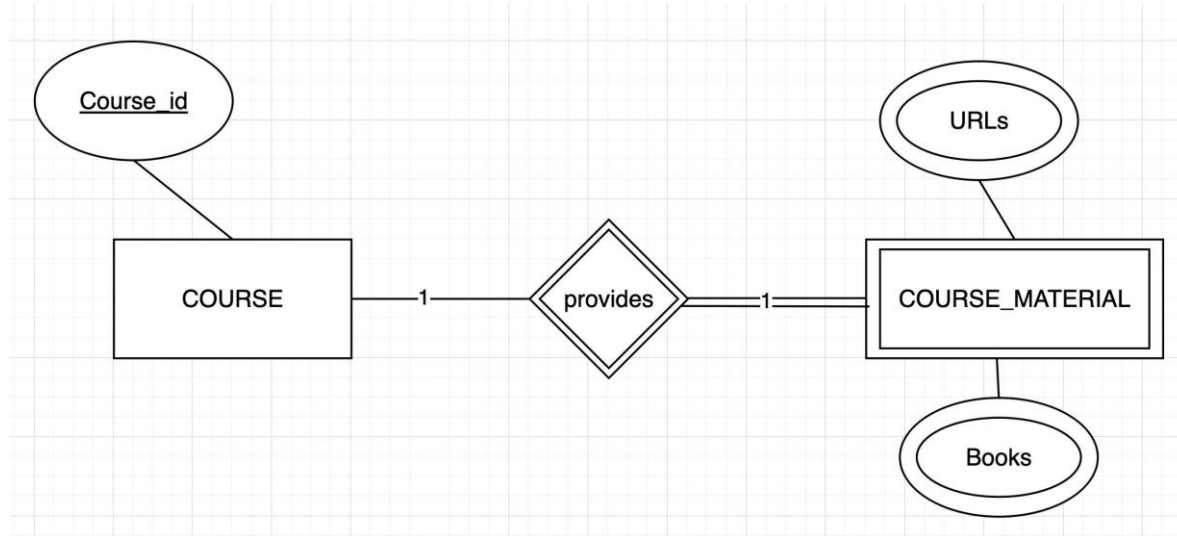


- ❑ We can convert any weak entity set to a strong entity set by simply adding appropriate attributes. Why, then, do we have weak entity sets?
  - We want to avoid the data duplication and consequent possible inconsistencies caused by duplicating the key of the strong entity.
  - Weak entities reflect the logical structure of an entity being dependent on another entity.
  - Weak entities can be deleted automatically when their strong entity is deleted.
  - Weak entities can be stored physically with their strong entities

# Strong & Weak Entity



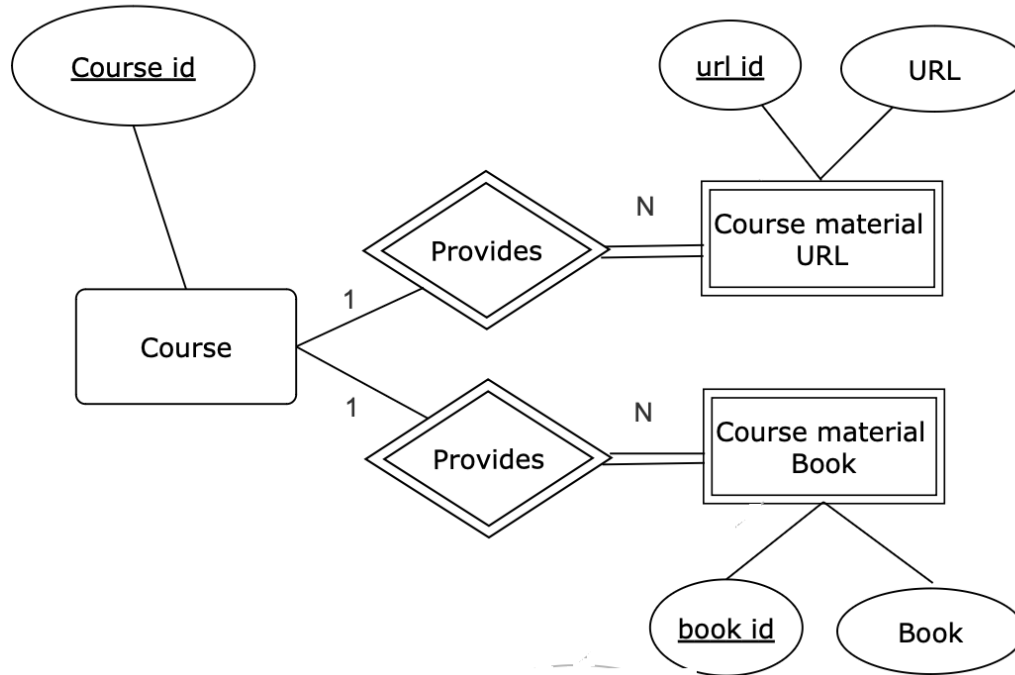
❑ Think about this design:



# Strong & Weak Entity



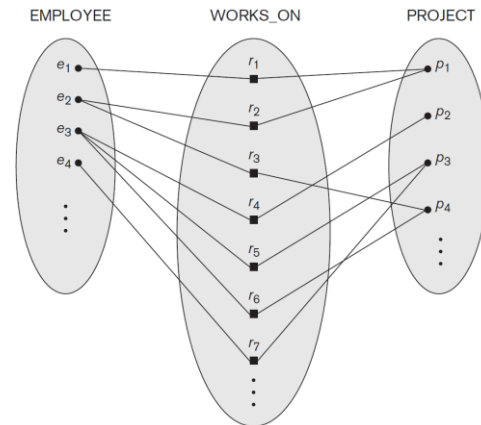
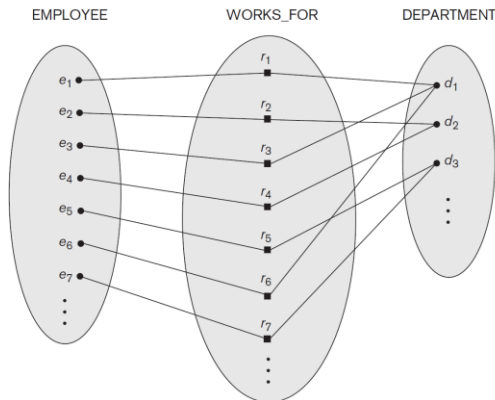
□ Solution:





## Attributes of relationship types can be migrated to one of the participating entity types:

- 1:1 = can be attribute of any entities.
- 1:N = can be migrated only to the entity type on the N-side of the relationship.
- The decision where to place a relationship attribute—as a relationship type attribute or as an attribute of a participating entity type—is determined subjectively by the schema designer.
- M:N = some attributes may be determined by the combination of participating entities in a relationship instance, not by any single entity.





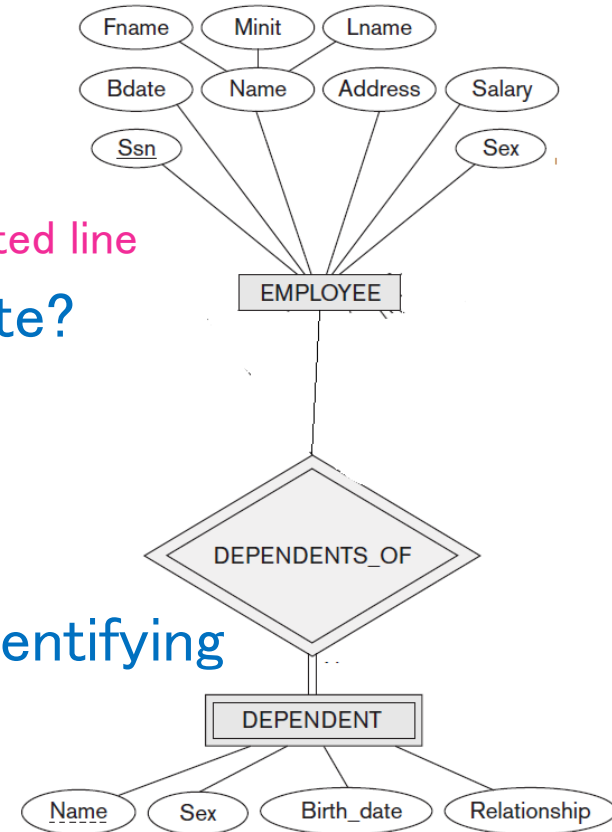
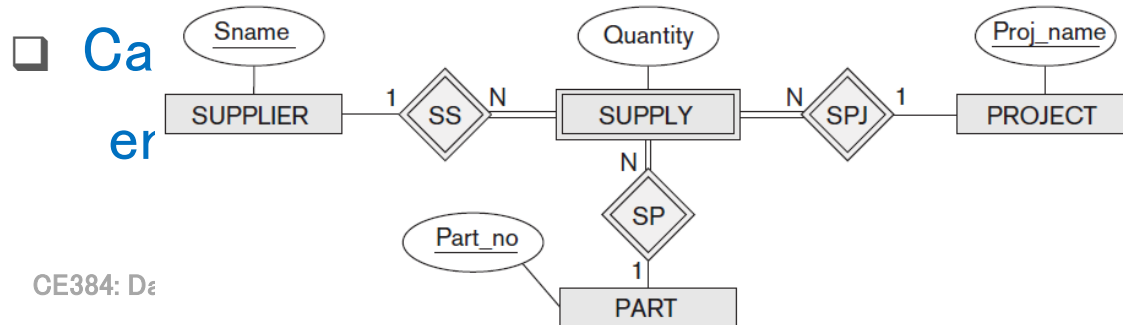


- ❑ **Not** every existence dependency results in a weak entity type.
  - Example: DRIVER\_LICENSE entity cannot exist unless it is related to a PERSON entity, even though it has its own key (License\_number) and hence is not a weak entity
- ❑ **Partial key or Discriminator:**
  - The attribute that can uniquely identify weak entities that are related to the same owner entity.
  - In the worst case, a composite attribute of all the weak entity's attributes will be the partial key.
  - In ER diagram: partial key attribute is underlined with a dashed or dotted line.

# Weak Entity Type in ER Model



- ❑ Weak entity type and its identifying relationship are represented by surrounding their boxes and diamonds with **double lines**
- ❑ Partial key attribute is underlined with a **dashed or dotted line**
- ❑ **When use weak entity or complex attribute?**
  - Data base designer choice
  - If the weak entity type participates independently in relationship types other than its identifying relationship type.

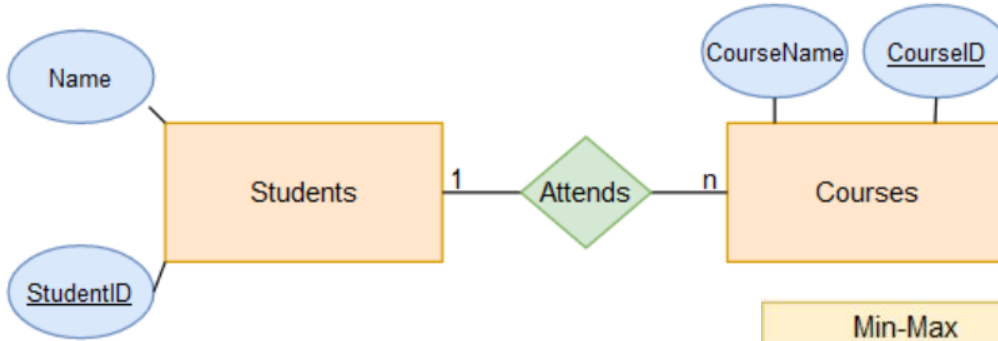




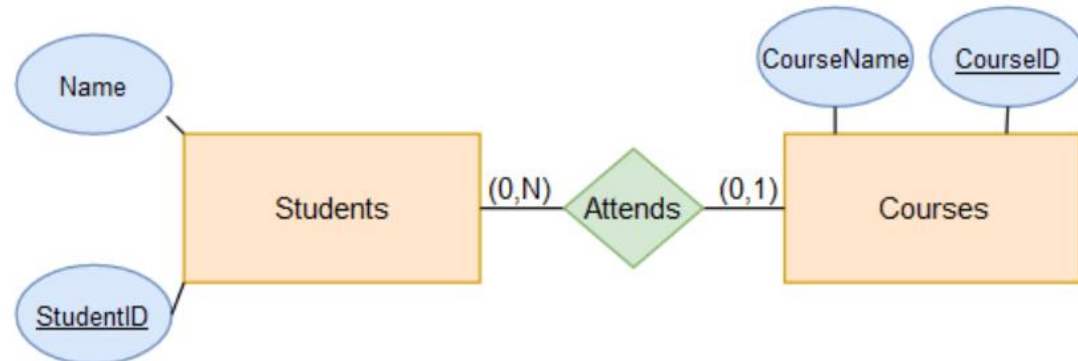
- ☐ A concept may be first modeled as an attribute and then refined into a relationship.
- ☐ An attribute that exists in several entity types may be elevated or promoted to an independent entity type.
- ☐ An inverse refinement to the previous case may be applied.

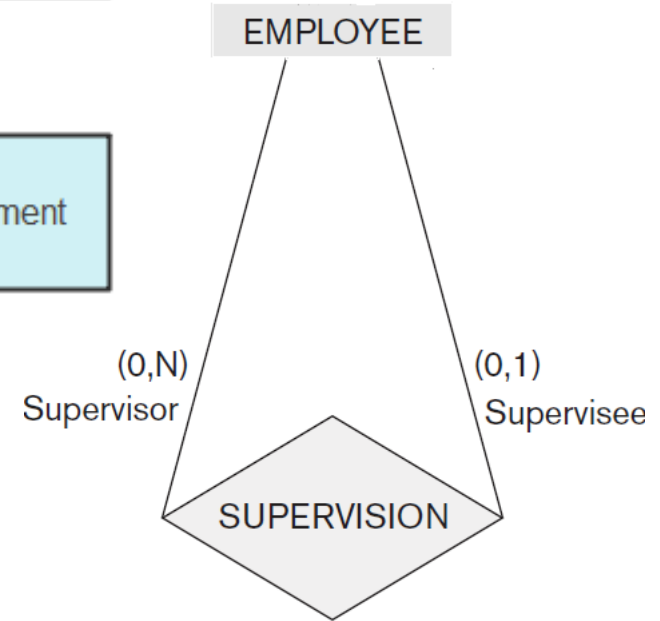


Chen

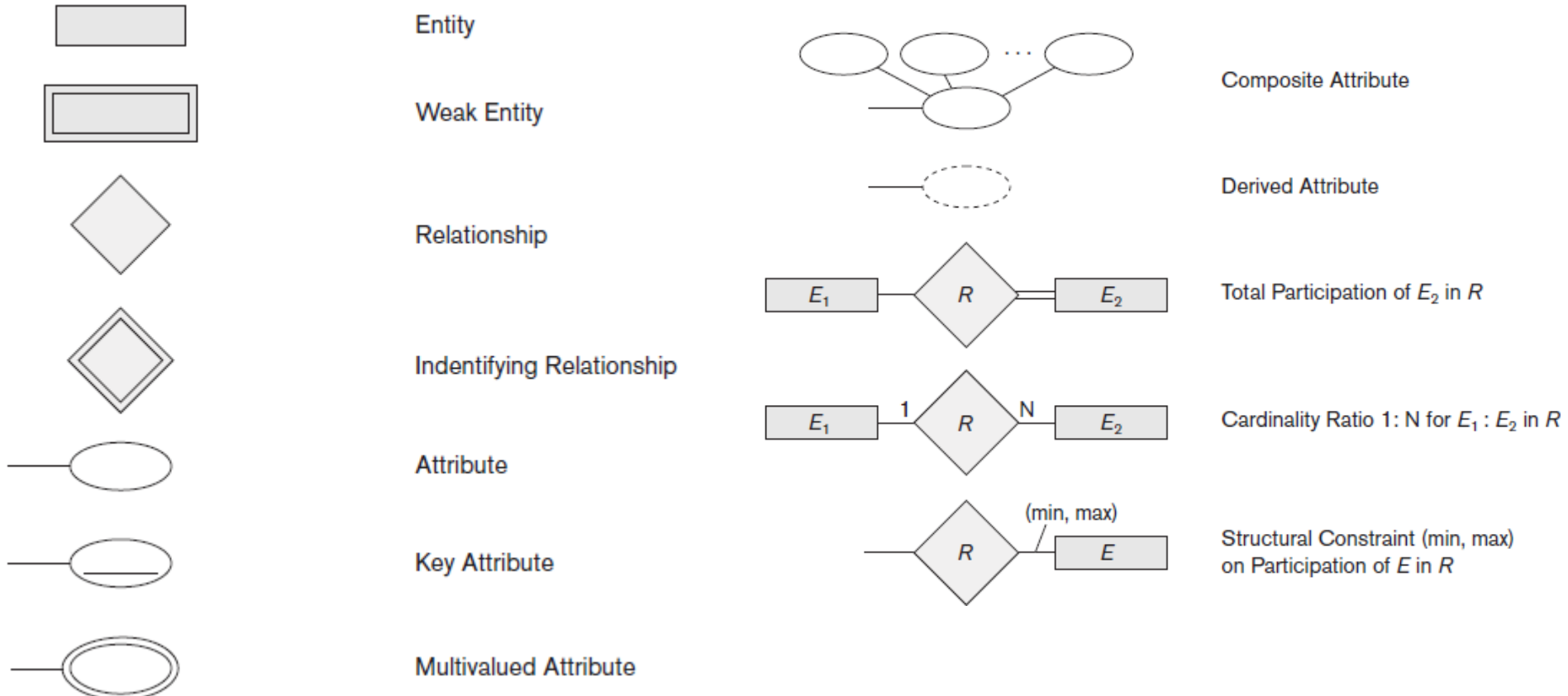


Min-Max





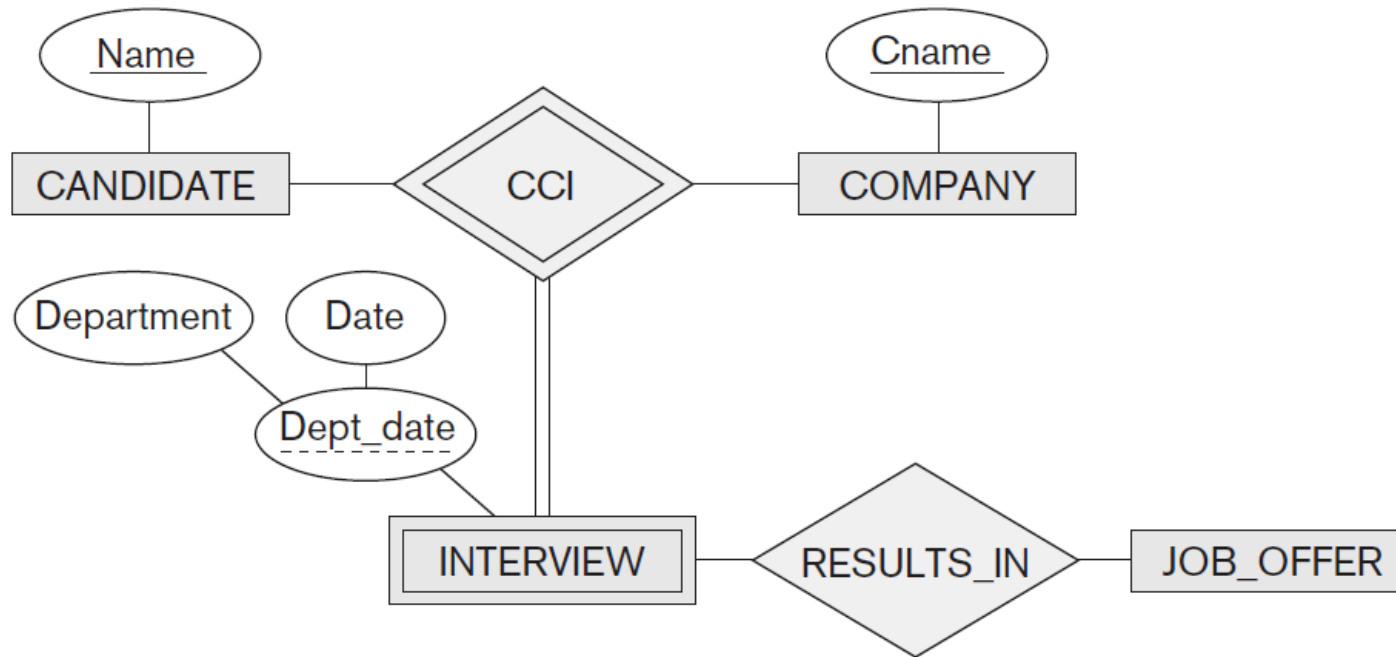
# ER Notations



# Some Examples



- ❑ A weak entity type INTERVIEW with a ternary identifying relationship type.



# Use of Entity Sets versus Attributes



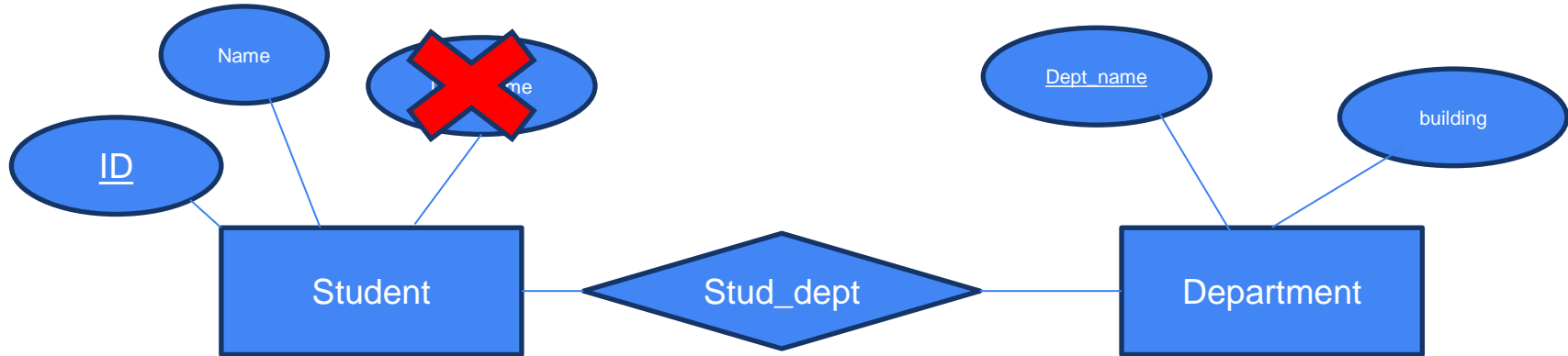
- ❑ Consider the entity set **instructor** with the additional attribute **phone number**.
- ❑ The **location** may be the office or home where the phone is located, with mobile (cell) phones perhaps represented by the value “mobile.” **Treating a phone as:**
  - an attribute phone number: instructors have precisely one phone number each.
    - **To keep extra information about a phone.**
  - a phone as an entity phone :permits instructors to have several phone numbers (including zero) associated with them.
  - phone number as a multivalued attribute: to allow multiple phones per instructor



# Common Mistakes in ER Diagrams



- ❑ Use of the key attribute of an entity set as an attribute of another entity set, instead of using a relationship:
  - Example of incorrect use of attribute:

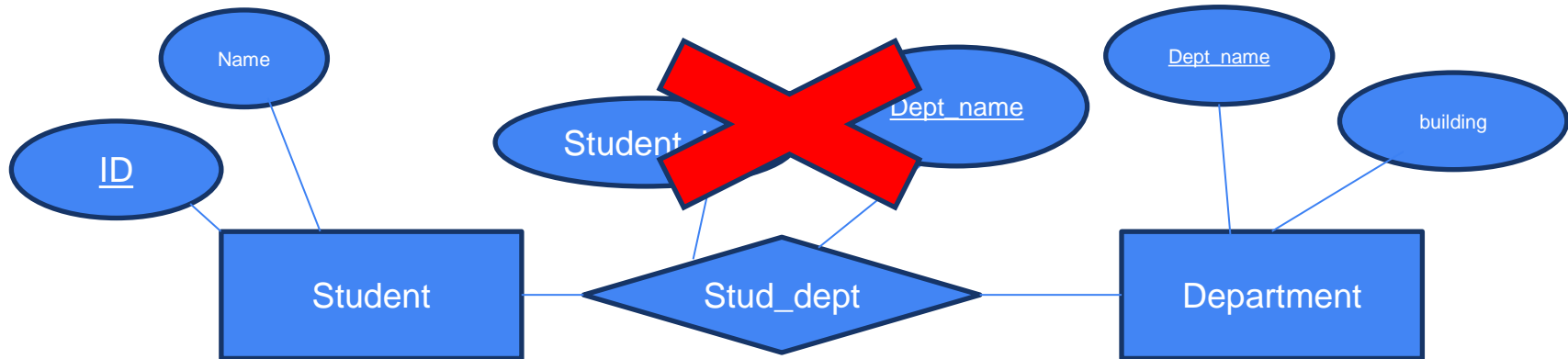


# Common Mistakes in ER Diagrams



- ❑ Designate the key attributes of the related entity sets as attributes of the relationship set

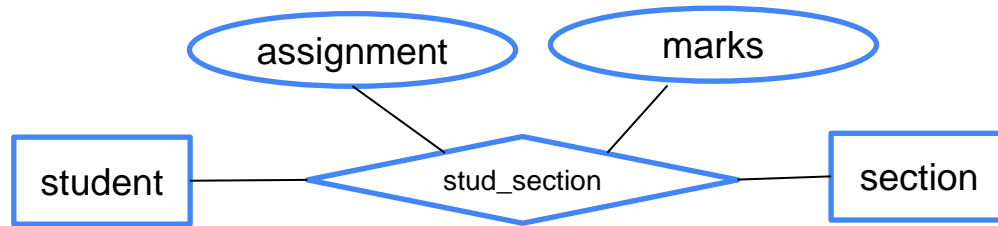
- Example of incorrect use of attribute:



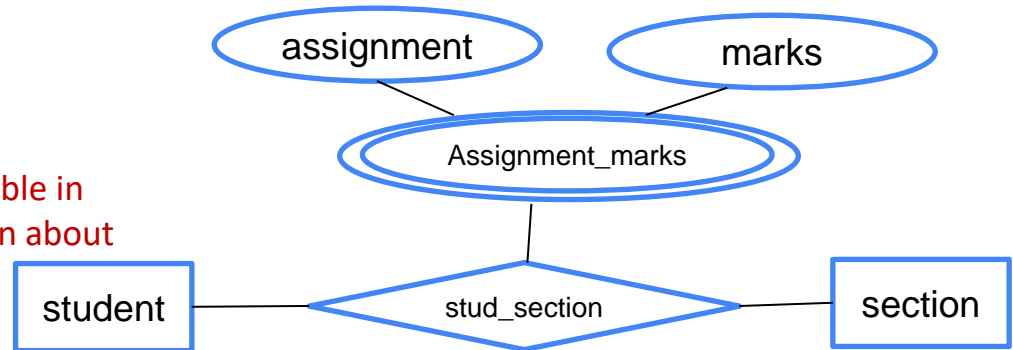
# Common Mistakes in ER Diagrams



- ❑ **NOTE:** relationship instances must be uniquely identified by the participating entities.



Modeling an assignment as a weak entity is preferable in this case, since it allows recording other information about the assignment, such as maximum marks or deadlines.





- ❑ Chapter 3 of FUNDAMENTALS OF Database Systems, SEVENTH EDITION
- ❑ Chapter 12 of Database Systems A Practical Approach to Design, Implementation, and Management, SIXth edition
- ❑ <https://www.geeksforgeeks.org/difference-between-entity-entity-set-and-entity-type/>
- ❑ <https://afteracademy.com/blog/what-is-an-entity-entity-type-and-entity-set/>