



# CSE 215: Database

Department of CSE  
Bangladesh University of Engineering & Technology



## What is it?

- The ER data model employs three basic concepts:
  - entity sets
  - relationship sets
  - attributes
- The ER model also has an associated diagrammatic representation, which can express the overall logical structure of a database graphically
  - ER diagram



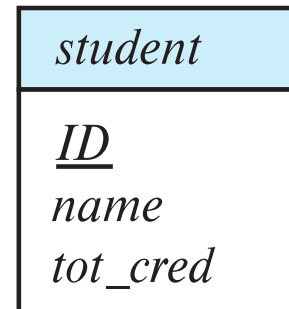
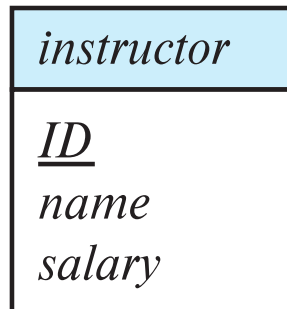
## Entity Sets

- An **entity** is an object that exists and is distinguishable from other objects
  - Example: specific person, company, plant
- An **entity set** is a set of entities of the same type that share the same properties
  - Example: set of all persons, companies, trees,
- An entity is represented by a set of attributes; i.e., descriptive properties possessed by all members of an entity set
  - Example:  
instructor = (ID, name, salary )  
course= (course\_id, title, credits)
- A subset of the attributes form a **primary key** of the entity set; i.e., uniquely identifying each member of the set



## Entity Sets – ER Diagram

- Entity sets can be represented graphically as follows:
  - Rectangles represent entity sets.
  - Attributes listed inside entity rectangle
  - Underline indicates primary key attributes





## Relationship Sets

- A **relationship** is an association among several entities

Example:

44553 (Peltier)      advisor      22222 (Einstein)  
*student* entity   relationship set   *instructor* entity

- A **relationship set** is a mathematical relation among  $n \geq 2$  entities, each taken from entity sets

$$\{(e_1, e_2, \dots, e_n) \mid e_1 \in E_1, e_2 \in E_2, \dots, e_n \in E_n\}$$

where  $(e_1, e_2, \dots, e_n)$  is a relationship

– Example:

$$(44553, 22222) \in \text{advisor}$$



## Relationship Sets

- A **relationship** is an association among several entities

Example:

44553 (Peltier)      advisor      22222 (Einstein)  
*student* entity   relationship set   *instructor* entity

- A **relationship set** is a mathematical relation among  $n \geq 2$  entities, each taken from entity sets

$$\{(e_1, e_2, \dots, e_n) \mid e_1 \in E_1, e_2 \in E_2, \dots, e_n \in E_n\}$$

where  $(e_1, e_2, \dots, e_n)$  is a relationship

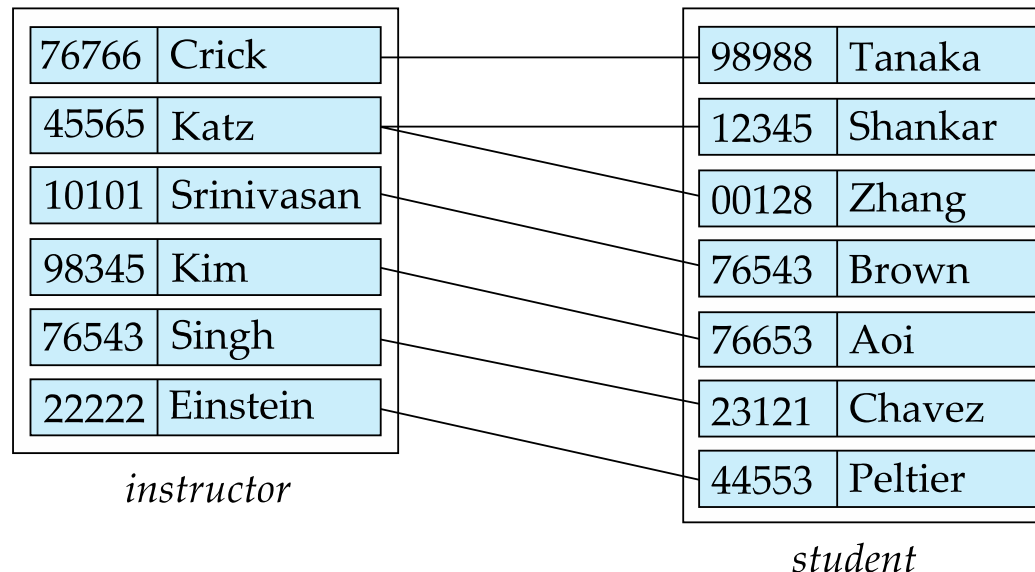
– Example:

$$(44553, 22222) \in \text{advisor}$$



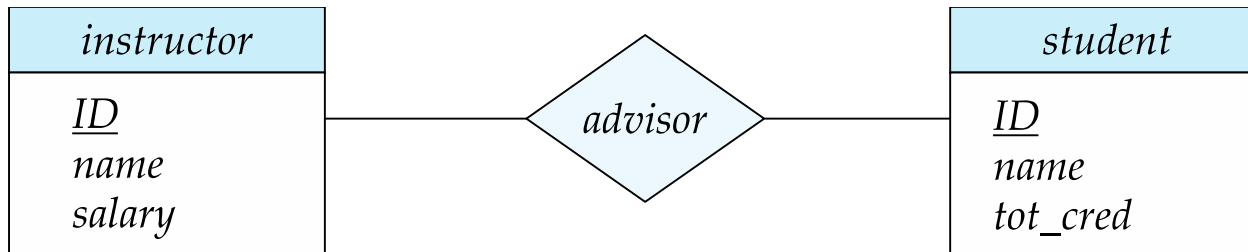
## Relationship Sets

- Example: we define the relationship set *advisor* to denote the associations between students and the instructors who act as their advisors
- Pictorially, we draw a line between related entities



## Relationship Sets-ER Diagram

- Diamonds represent relationship sets

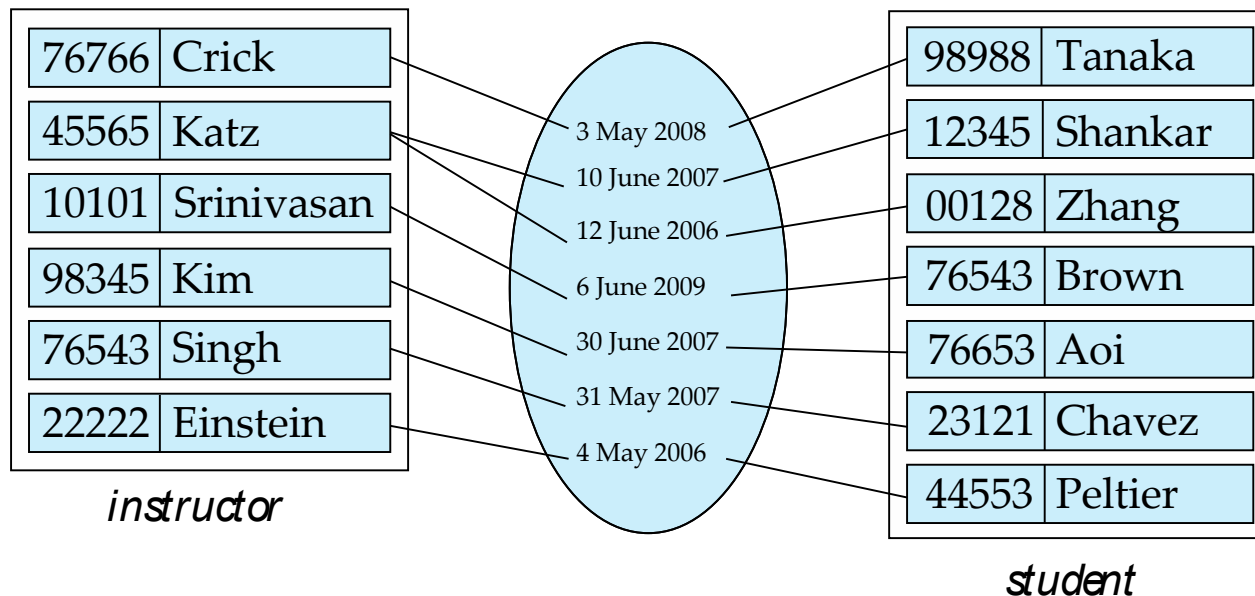






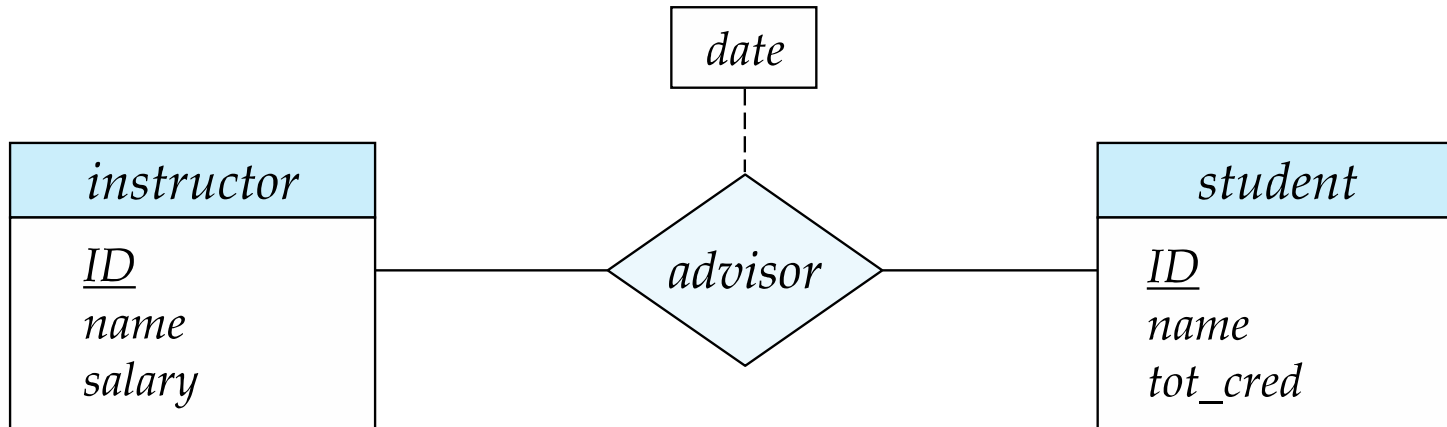
## Relationship Sets

- An attribute can also be associated with a relationship set.
- For instance, the advisor relationship set between entity sets instructor and student may have the attribute date which tracks when the student started being associated with the advisor



## Relationship Sets-ER Diagram

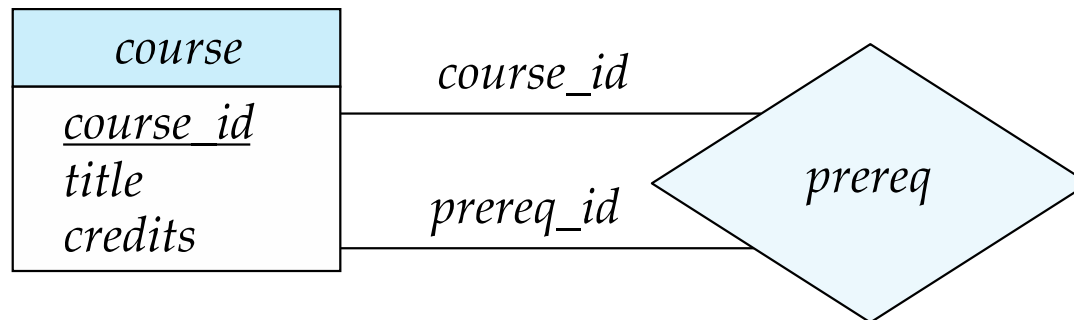
- Diamonds represent relationship sets





## Roles

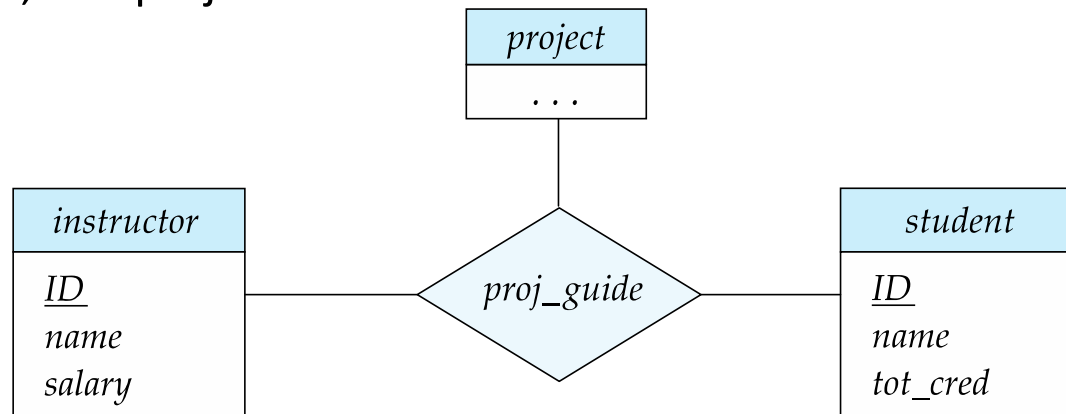
- Entity sets of a relationship need not be distinct
  - Each occurrence of an entity set plays a “**role**” in the relationship
- The labels “course\_id” and “prereq\_id” are called roles.





## Relationship Degree

- Binary relationship
  - Involve two entity sets (or degree two)
- Relationships between more than two entity sets are rare
  - Example: students work on research projects under the guidance of an instructor.
  - relationship `proj_guide` is a ternary relationship between instructor, student, and project





## Attributes

- Attribute types:
  - **Simple** and **composite** attributes
    - Composite attributes allow us to divided attributes into subparts
  - **Single-valued** and **multivalued** attributes
    - Example: multivalued attribute:  
*phone\_numbers*
  - **Derived** attributes
    - Can be computed from other attributes
    - Example: age, given *date\_of\_birth*
- **Domain** – the set of permitted values for each attribute

<i>instructor</i>
<u><i>ID</i></u>
<i>name</i>
<i>first_name</i>
<i>middle_initial</i>
<i>last_name</i>
<i>address</i>
<i>street</i>
<i>street_number</i>
<i>street_name</i>
<i>apt_number</i>
<i>city</i>
<i>state</i>
<i>zip</i>
{ <i>phone_number</i> }
<i>date_of_birth</i>
<i>age</i> ( )

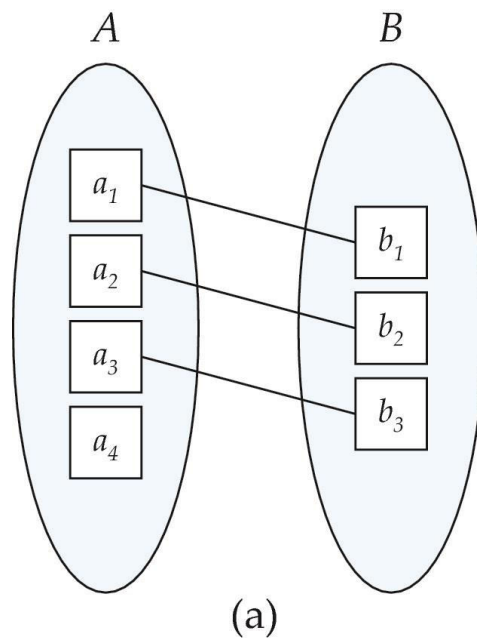


## Mapping Cardinalities

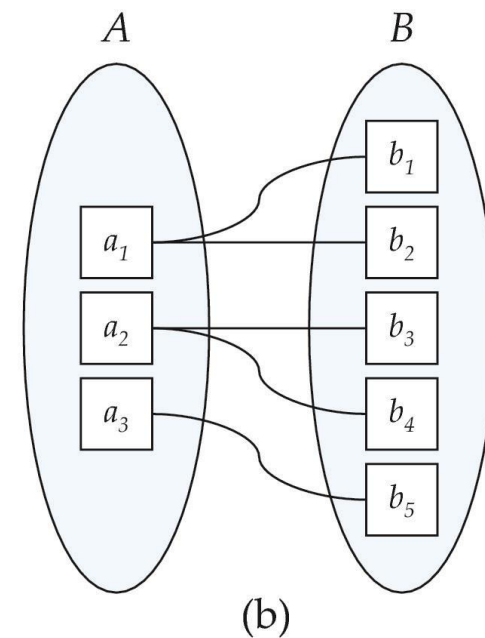
- Express the number of entities to which another entity can be associated via a relationship set
- **One to one**
  - An entity in A is associated with at most one entity in B, An entity in B is associated with at most one entity in A
- **One to many**
  - An entity in A is associated with any number (0 or more) of entities in B, An entity in B is associated with at most one entity in A
- **Many to one**
  - An entity in A is associated with at most one entity in B, An entity in B is associated with any number (0 or more) of entities in A
- **Many to many**
  - An entity in A is associated with any number (0 or more) of entities in B, An entity in B is associated with any number (0 or more) of entities in A



## Mapping Cardinalities



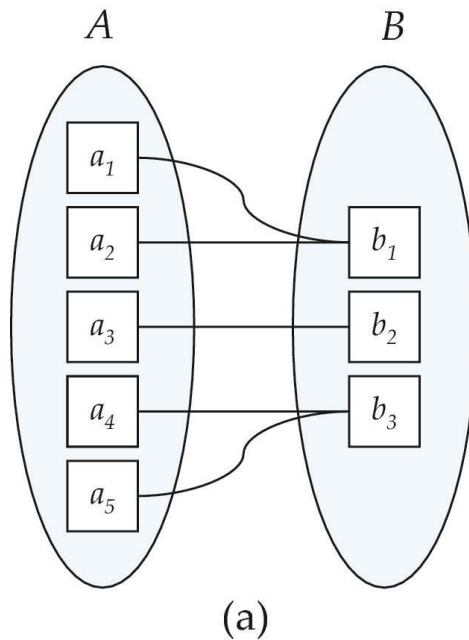
One to one



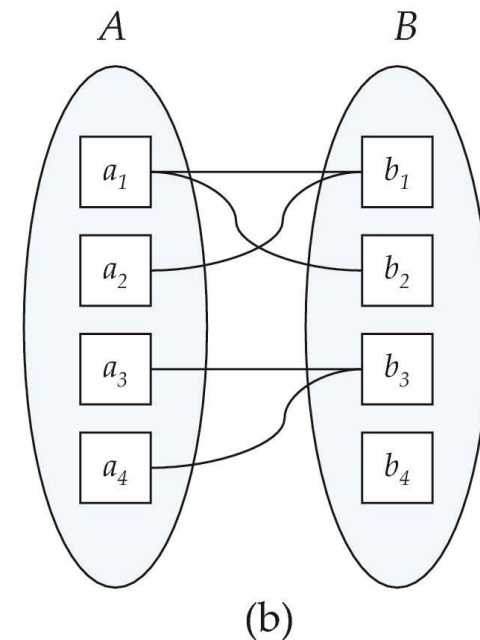
One to many



## Mapping Cardinalities



Many to one



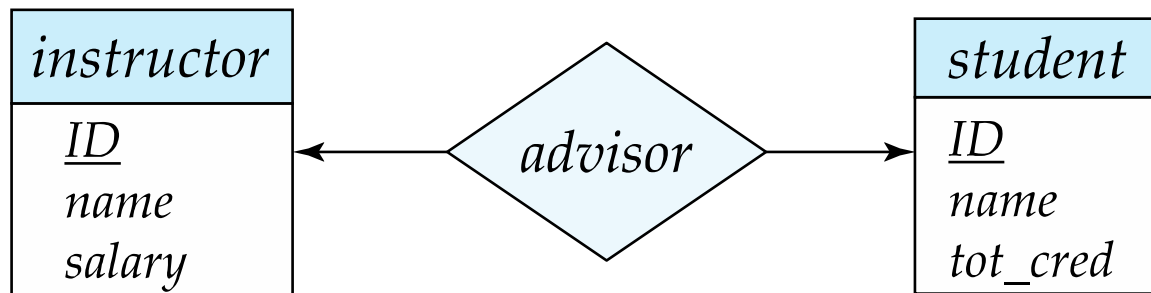
Many to many





## Cardinality Constraints-ER Diagram

- We express cardinality constraints by drawing either a directed line (⤵), signifying “one,” or an undirected line (—), signifying “many,” between the relationship set and the entity set

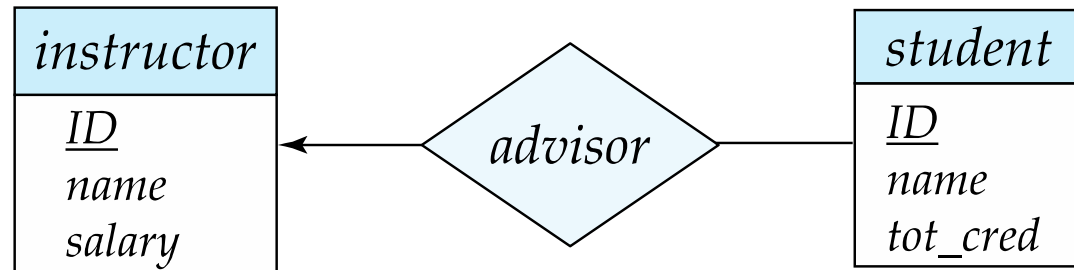


One to one



## Cardinality Constraints-ER Diagram

- We express cardinality constraints by drawing either a directed line (→), signifying “one,” or an undirected line (—), signifying “many,” between the relationship set and the entity set

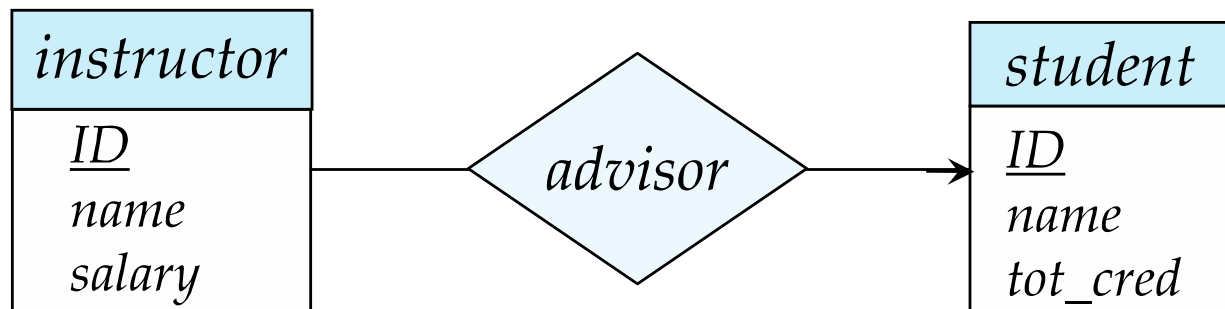


One to many



## Cardinality Constraints-ER Diagram

- We express cardinality constraints by drawing either a directed line (→), signifying “one,” or an undirected line (—), signifying “many,” between the relationship set and the entity set

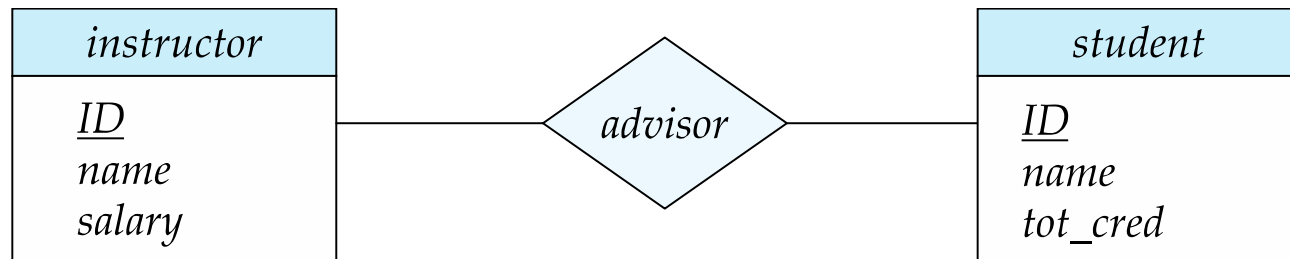


Many to one



## Cardinality Constraints-ER Diagram

- We express cardinality constraints by drawing either a directed line (→), signifying “one,” or an undirected line (—), signifying “many,” between the relationship set and the entity set

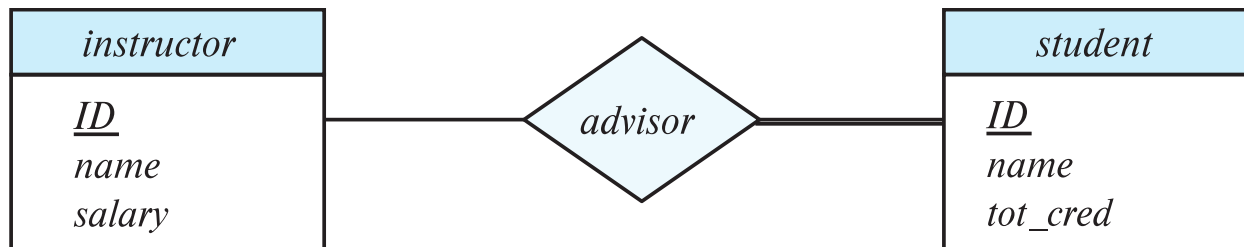


Many to many



## Cardinality Constraints-ER Diagram

- **Total participation** (indicated by double line): every entity in the entity set participates in at least one relationship in the relationship set
  - Example: participation of student in advisor relation is total, every student must have an associated instructor

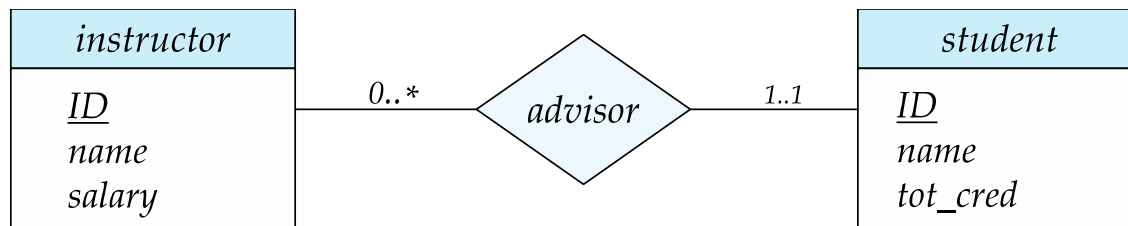


- **Partial participation**: some entities may not participate in any relationship in the relationship set
  - Example: participation of instructor in advisor is partial



## Cardinality Constraints-ER Diagram

- A line may have an associated minimum and maximum cardinality, shown in the form l..h, where l is the minimum and h the maximum cardinality
  - A minimum value of 1 indicates total participation
  - A maximum value of 1 indicates that the entity participates in at most one relationship
  - A maximum value of \* indicates no limit



**Instructor can advise 0 or more students**

**A student must have 1 advisor; cannot have multiple advisors**



# Cardinality Constraints-ER Diagram

- We allow at most one arrow out of a ternary (or greater degree) relationship to indicate a cardinality constraint
  - For example, an arrow from proj\_guide to instructor indicates each student has at most one guide for a project
- If there is more than one arrow, there are two ways of defining the meaning.
  - For example, a ternary relationship R between A, B and C with arrows to B and C could mean
    1. Each A entity is associated with a unique entity from B and C
    2. Each pair of entities from (A, B) is associated with a unique C entity, and each pair (A, C) is associated with a unique B



## Primary Key for Entity Sets

- By definition, individual entities are distinct.
- From database perspective, the differences among them must be expressed in terms of their attributes.
- The values of the attribute values of an entity must be such that they can uniquely identify the entity.
  - No two entities in an entity set are allowed to have exactly the same value for all attributes.
- A **key** for an entity is a set of attributes that suffice to distinguish entities from each other





## Keys

- Let  $K$  is a subset of attributes
- $K$  is a superkey if values for  $K$  are sufficient to identify a unique tuple of each possible relation  $r(R)$ 
  - Example:  $\{ID\}$  and  $\{ID, name\}$  are both superkeys of instructor
- Superkey  $K$  is a candidate key if  $K$  is minimal
  - Example:  $\{ID\}$  is a candidate key for Instructor
- One of the candidate keys is selected to be the primary key



# Primary Key for Relationship Sets

- To distinguish among the various relationships of a relationship set we use the individual primary keys of the entities in the relationship set
- Let  $R$  be a relationship set involving entity sets  $E_1, E_2, \dots, E_n$
- The primary key for  $R$  consists of the union of the primary keys of entity sets  $E_1, E_2, \dots, E_n$ 
  - If the relationship set  $R$  has attributes  $a_1, a_2, \dots, a_m$  associated with it, then the primary key of  $R$  also includes the attributes  $a_1, a_2, \dots, a_m$
- Example: relationship set “advisor”
  - The primary key consists of instructor.ID and student.ID



## Primary Key for Relationship Sets

- The choice of the primary key for a relationship set depends on the mapping cardinality of the relationship set
- Many-to-Many relationships
  - The preceding union of the primary keys is a minimal superkey and is chosen as the primary key
- One-to-Many relationships
  - The primary key of the “Many” side is a minimal superkey and is used as the primary key
- Many-to-one relationships
  - The primary key of the “Many” side is a minimal superkey and is used as the primary key
- One-to-one relationships
  - The primary key of either one of the participating entity sets forms a minimal superkey, and either one can be chosen as the primary key



## Weak/Strong Entity Sets

- A **weak entity set** is one whose existence is dependent on another entity, called its **identifying entity**
- Instead of associating a primary key with a weak entity, we use the primary key of identifying entity, along with extra attributes called **discriminator** to uniquely identify a weak entity
- The identifying entity set is said to **own** the weak entity set that it identifies
- The relationship associating the weak entity set with the identifying entity set is called the **identifying relationship**
- An entity set that is not a weak entity set is termed a **strong entity set**.

## Weak Entity Sets-ER Diagram

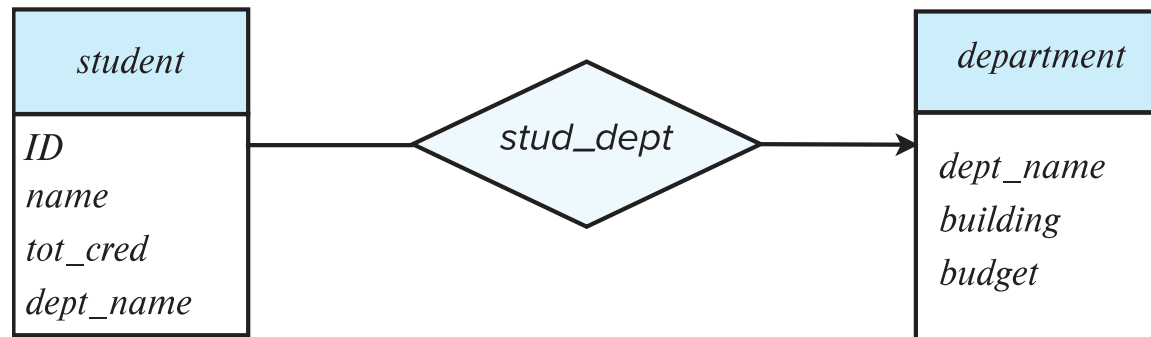
- In E-R diagrams, a weak entity set is depicted via a double rectangle.
- We underline the discriminator of a weak entity set with a dashed line.
- The relationship set connecting the weak entity set to the identifying strong entity set is depicted by a double diamond.
- Primary key for section – (course\_id, sec\_id, semester, year)





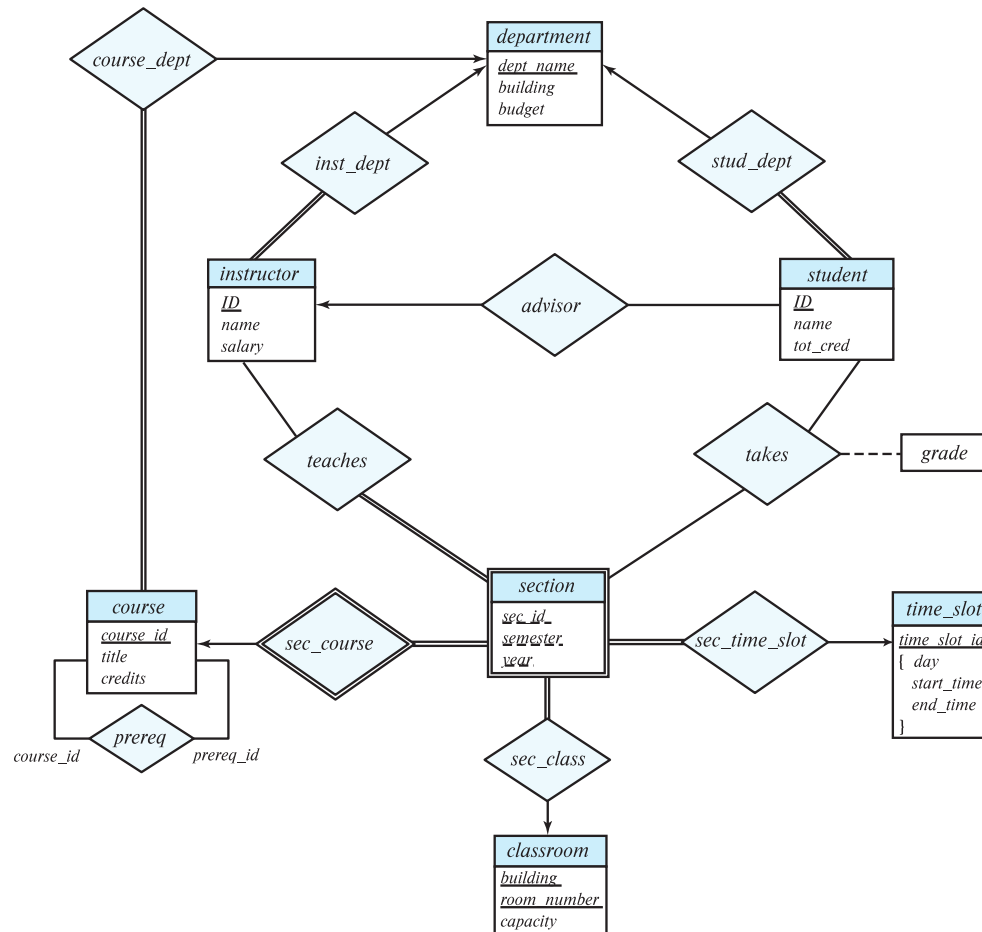
## Redundant Attributes

- We model the fact that each student has an associated department using a relationship set *stud\_dept*
- The attribute *dept\_name* in *student* below replicates information present in the relationship and is therefore redundant and needs to be removed.
- BUT: when converting back to tables, in some cases the attribute gets reintroduced, as we will see later



(a) Incorrect use of attribute

## ER Diagram of University





## Acknowledgement

- Some slides and figures are collected from
  - <https://www.db-book.com/>