# Database Management

Mr. Sandeep L Dhende

Assistant Professor, Dept. of E&TC, PICT, Pune

# Relational algebra

- Entity Relationship Model - Basic Concepts
- Entity Sets
- Relationship Sets and Weak Entity Sets
- Mapping Cardinalities
- Keys
- E-R Diagrams
- Design Issues
- Extended E-R Features- Specialization and Generalization
- Convert E-R and EER diagram into tables

# Entity-Relationship model: Basic Concepts

- Entity Relationship Model
- Models an enterprise as a collection of *entities* and *relationships*
  - Entity: a "thing" or "object" in the enterprise that is distinguishable from other objects
    - Described by a set of *attributes*
  - Relationship: an association among several entities
- Represented diagrammatically by an *entity-relationship diagram*

# Entity-Relationship model: Basic Concepts

- The ER data model was developed to facilitate database design by allowing specification of an **enterprise schema** that represents the overall logical structure of a database.
- The ER data model employs three basic concepts:
  - entity sets,
  - relationship sets,
  - attributes.
- The ER model also has an associated diagrammatic representation, the **ER diagram**, which can express the overall logical structure of a database graphically.

# Entity-Relationship model: Entity Sets

- An **entity** is an object that exists and is distinguishable from other objects.

  Example: specific person, company, event, 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, holidays

- 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-Relationship model: Entity Sets

Entity Sets -- *instructor* and *student*

| | |
|---|---|
| 76766 | Crick |
| 45565 | Katz |
| 10101 | Srinivasan |
| 98345 | Kim |
| 76543 | Singh |
| 22222 | Einstein |

*instructor*

| | |
|---|---|
| 98988 | Tanaka |
| 12345 | Shankar |
| 00128 | Zhang |
| 76543 | Brown |
| 76653 | Aoi |
| 23121 | Chavez |
| 44553 | Peltier |

*student*

Representing Entity Sets

| *instructor* |
|---|
| *ID* |
| *name* |
| *salary* |

| *student* |
|---|
| *ID* |
| *name* |
| *tot_cred* |

# Entity-Relationship model: 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

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

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

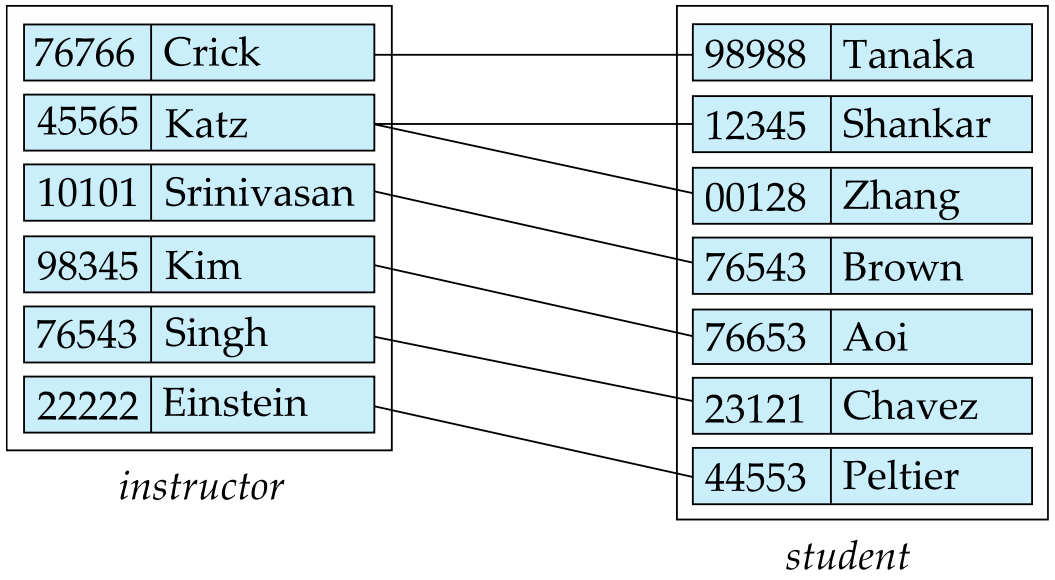
  Example:

  $$(44553,22222) \in advisor$$

- **Degree of RS- No. of participating entities in a relationship set.**

# Entity-Relationship model: Relationship Sets

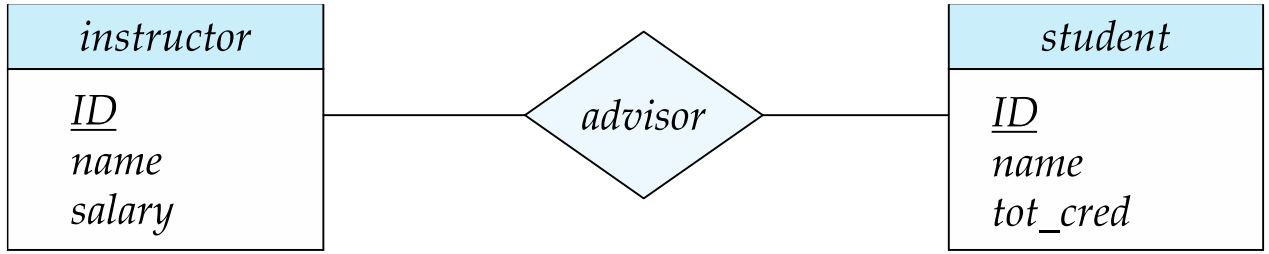
- Example: we define the relationship set *advisor* to denote the associations between students and the instructors who act as their advisors.
- Pictorially, we draw a line between related entities.



# Entity-Relationship model:

## Representing Relationship Sets via ER Diagrams

- Diamonds represent relationship sets.



# Entity-Relationship model

- Attribute types: Descriptive properties possessed by each member of an entity set.
- Attribute types:
  - **Simple** and **composite** attributes.
  - **Simple-** An attribute composed of a single component with an independent existence
  - **composite attributes-** An attribute composed of multiple components, each with an independent existence.
  - **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

# Entity-Relationship model: Weak Entity Sets

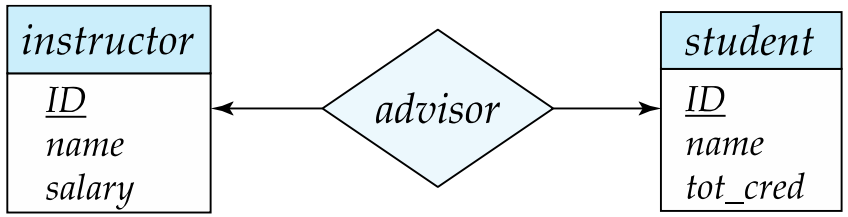
- A **weak entity set** is one whose existence is dependent on another entity, called its **identifying entity**
- Instead of associating a primary key with a weak entity, we use the identifying entity, along with extra attributes called **discriminator** to uniquely identify a weak entity.
- An entity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

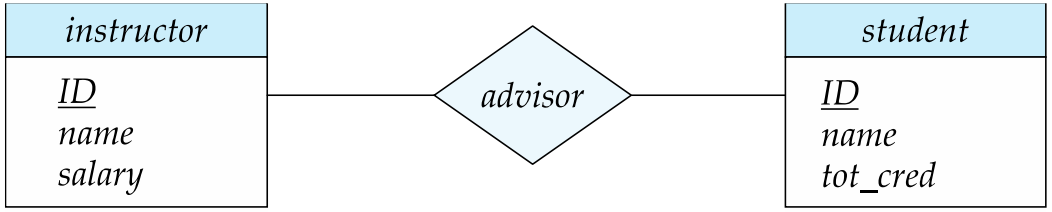
# Entity-Relationship model: Weak Entity Sets

- The relationship associating the weak entity set with the identifying entity set is called the **identifying relationship**.
- Note that the relational schema we eventually create from the entity set *section* does have the attribute *course_id*, for reasons that will become clear later, even though we have dropped the attribute *course_id* from the entity set *section*.
- Primary key for *section* – (*course_id*, *sec_id*, *semester*, *year*)

# Entity-Relationship model: Mapping Cardinality Constraints

- Express the number of entities to which another entity can be associated via a relationship set.
- Most useful in describing binary relationship sets.
- For a binary relationship set the mapping cardinality must be one of the following types:
  - **One to one**
  - One to many
  - Many to one
  - **Many to many**



# Entity-Relationship model: 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 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.

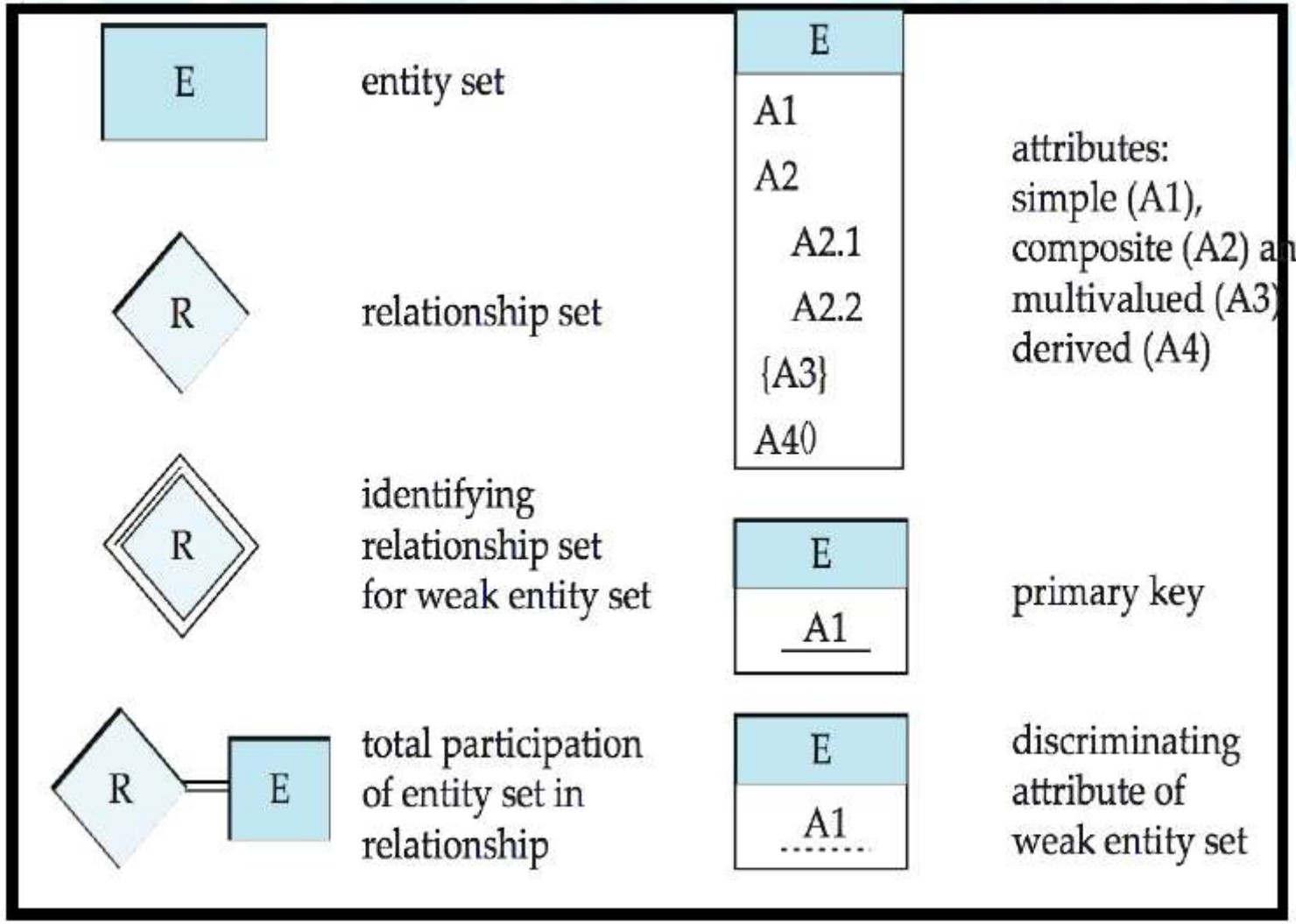
# Entity-Relationship model: ERD Development Process

- Identify the entities
- Determine the attributes for each entity.
- Select the primary key for each entity.
- Establish the relationship between the entities.
- Draw an entity model.
- Test the relationships and the keys.

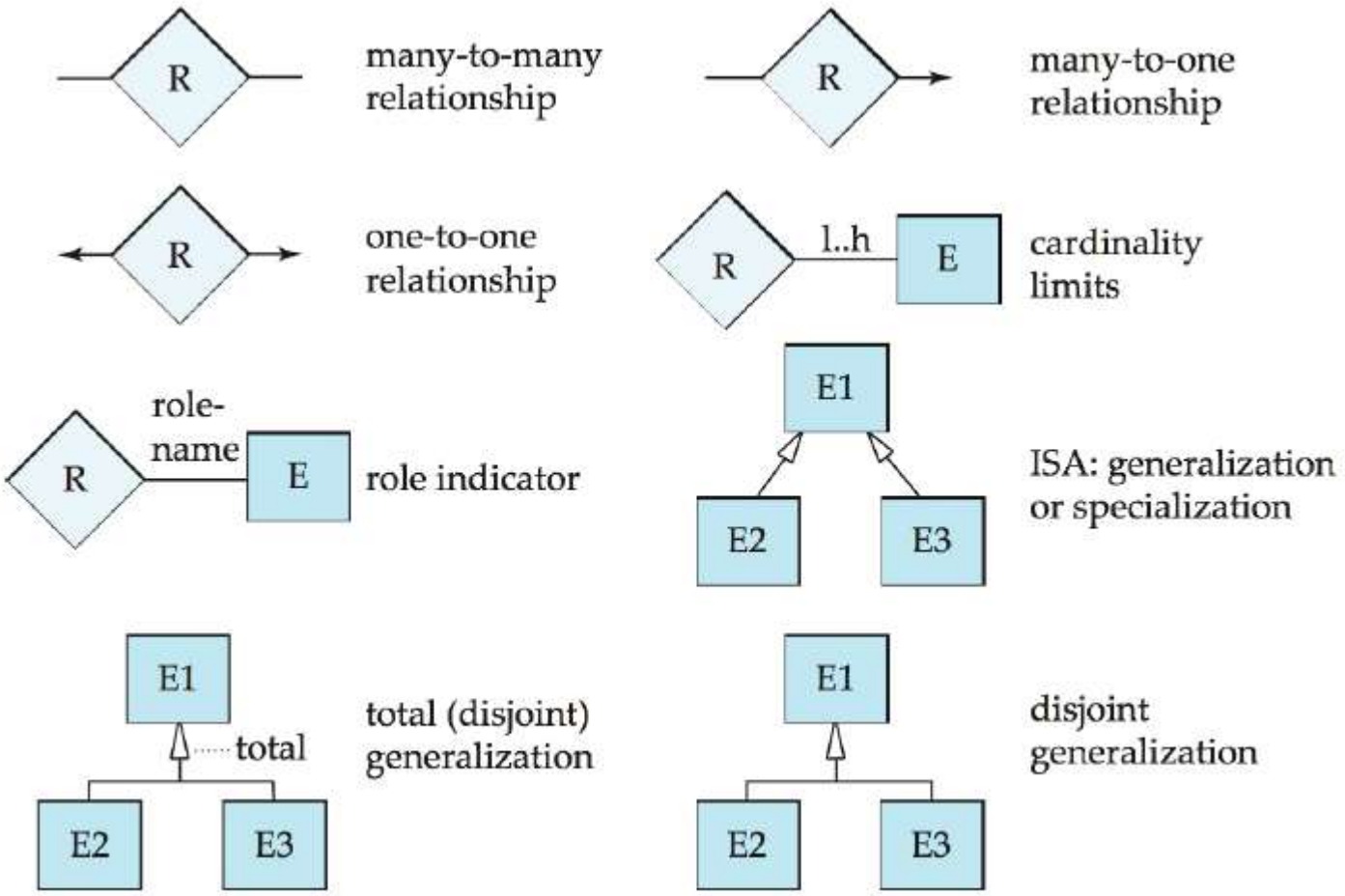
# Entity-Relationship model: ERD Development Process

- Rectangles – Entity set
- Double rectangles-weak entity set
- Diamonds - relationship sets
- Ellipses – Attributes
- Double ellipses represent multivalued attributes.
- Dashed ellipses denote derived attributes.
- Lines – link attributes to entity sets and entity sets to Relationship sets.
- Underline indicates primary key attributes

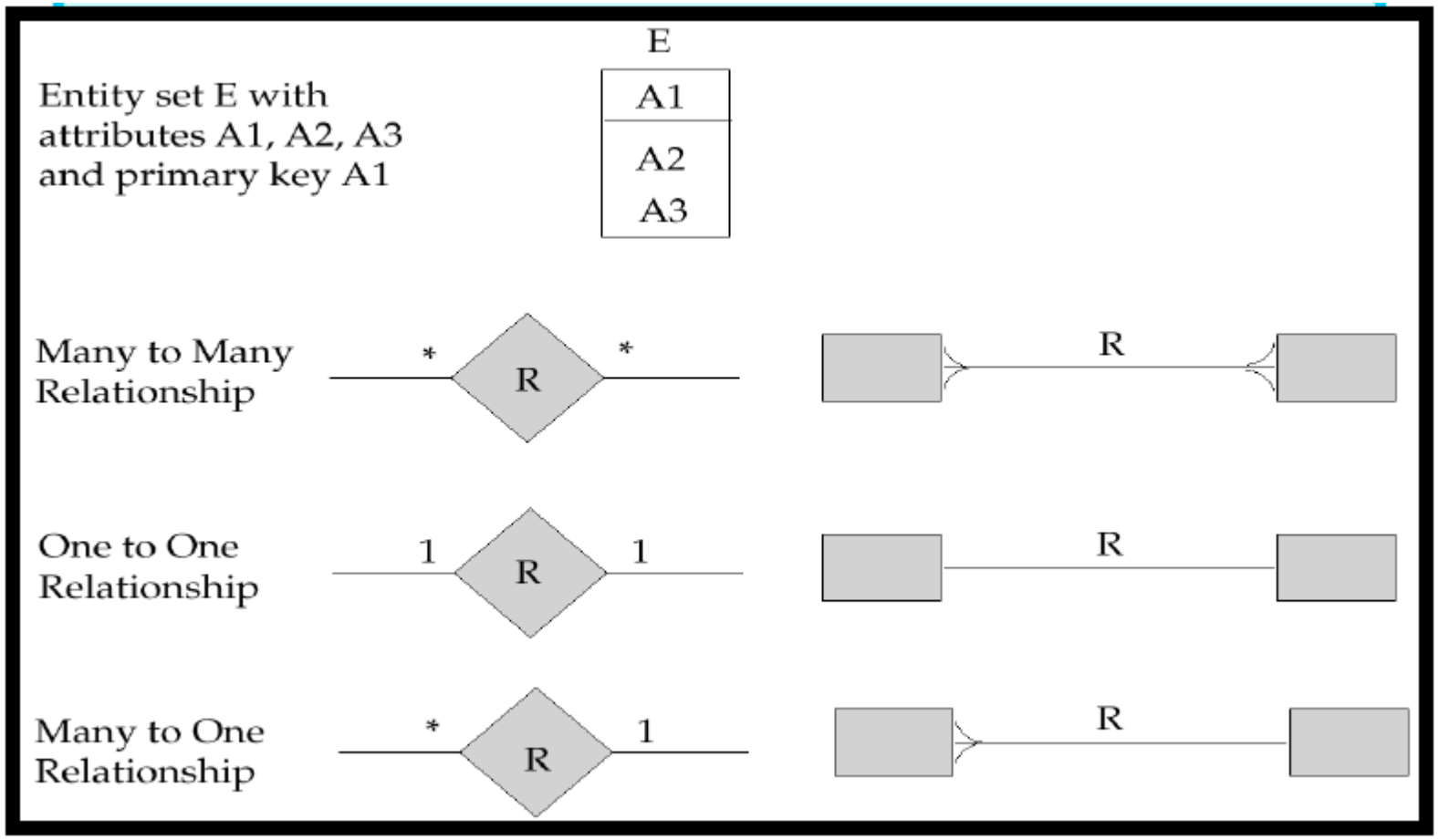
# Entity-Relationship model: ERD Development Process



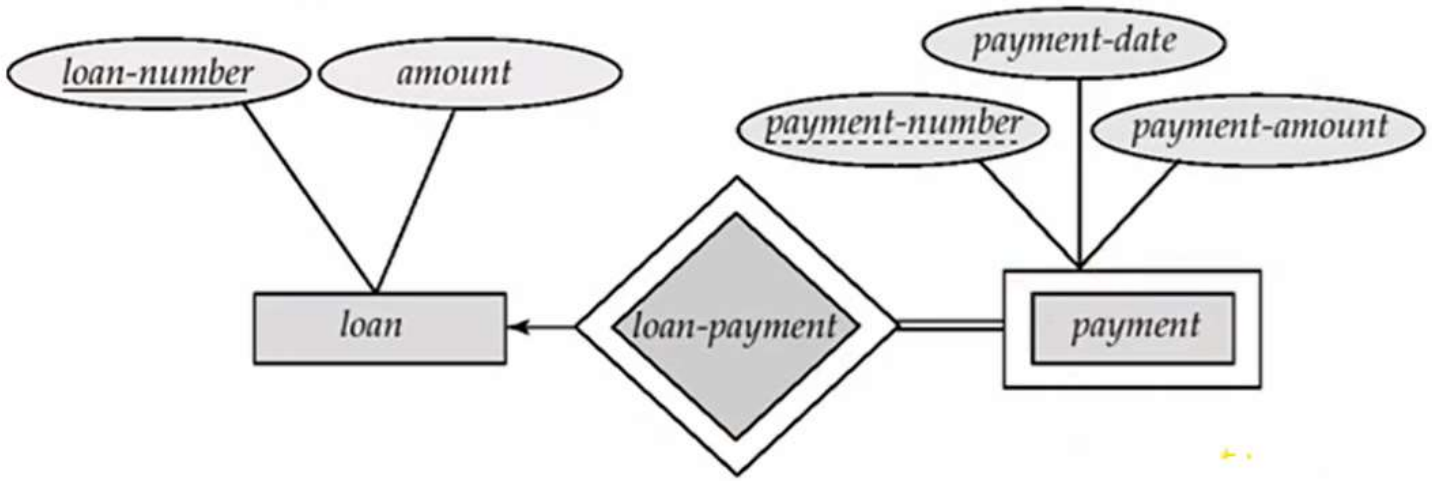
# Entity-Relationship model: ERD Development Process



# Entity-Relationship model: Alternative ER Notations



# Entity-Relationship model: Example of Weak Entity Set

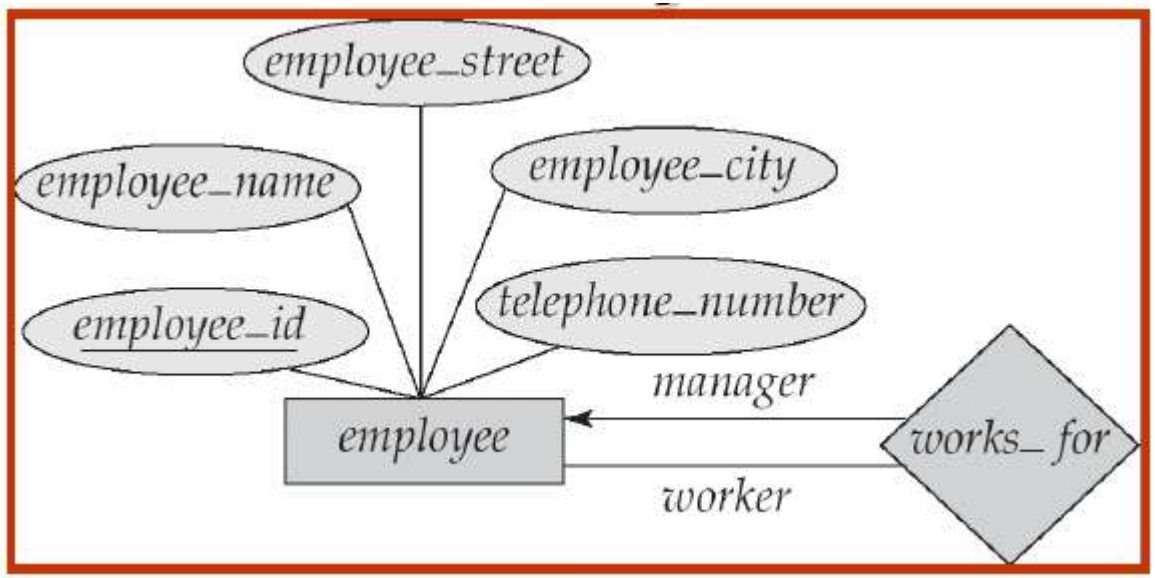



| Loan_no | Amount |
|---|---|
| L1 | 1,00,000 |
| L2 | 2,00,000 |
| L3 | 3,00,000 |

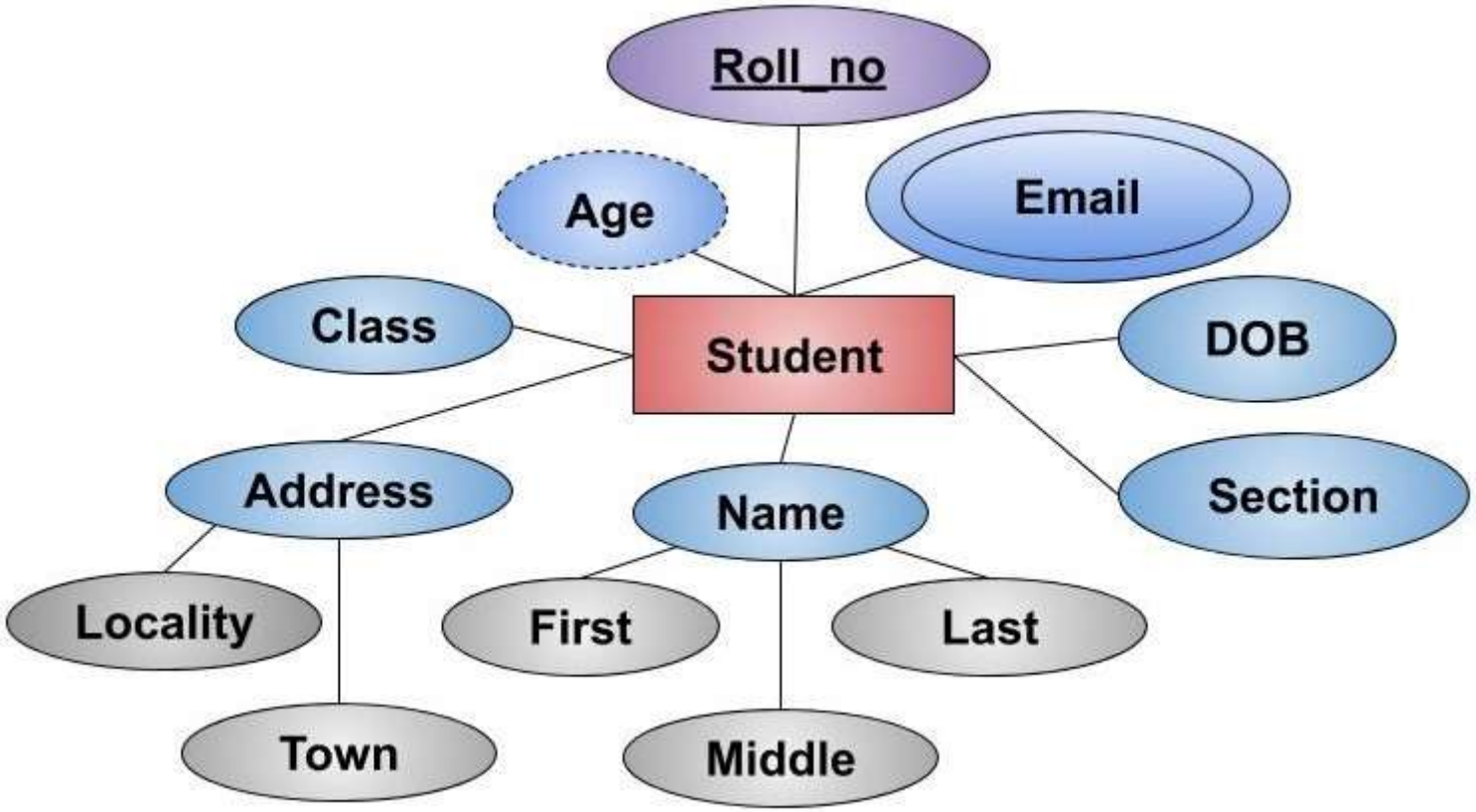
| Payment_no | Payment_date | Payment_amount |
|---|---|---|
| 1 | 05-06-2020 | 5000 |
| 1 | 08-07-2020 | 10000 |
| 1 | 10-08-2020 | 15000 |
| 2 | 05-07-2020 | 5000 |
| 2 | 08-08-2020 | 10000 |
| 2 | 10-09-2020 | 15000 |

# Entity-Relationship model: Roles

- Roles are indicated in E-R diagrams by labeling the lines that connect diamonds to rectangles

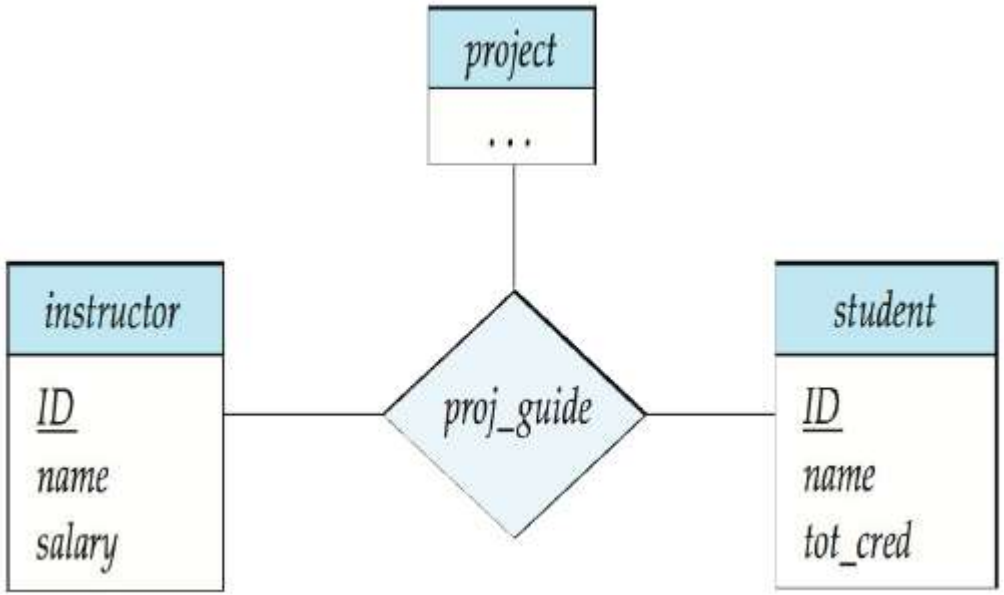


# Entity-Relationship model: Different Attributes of Student

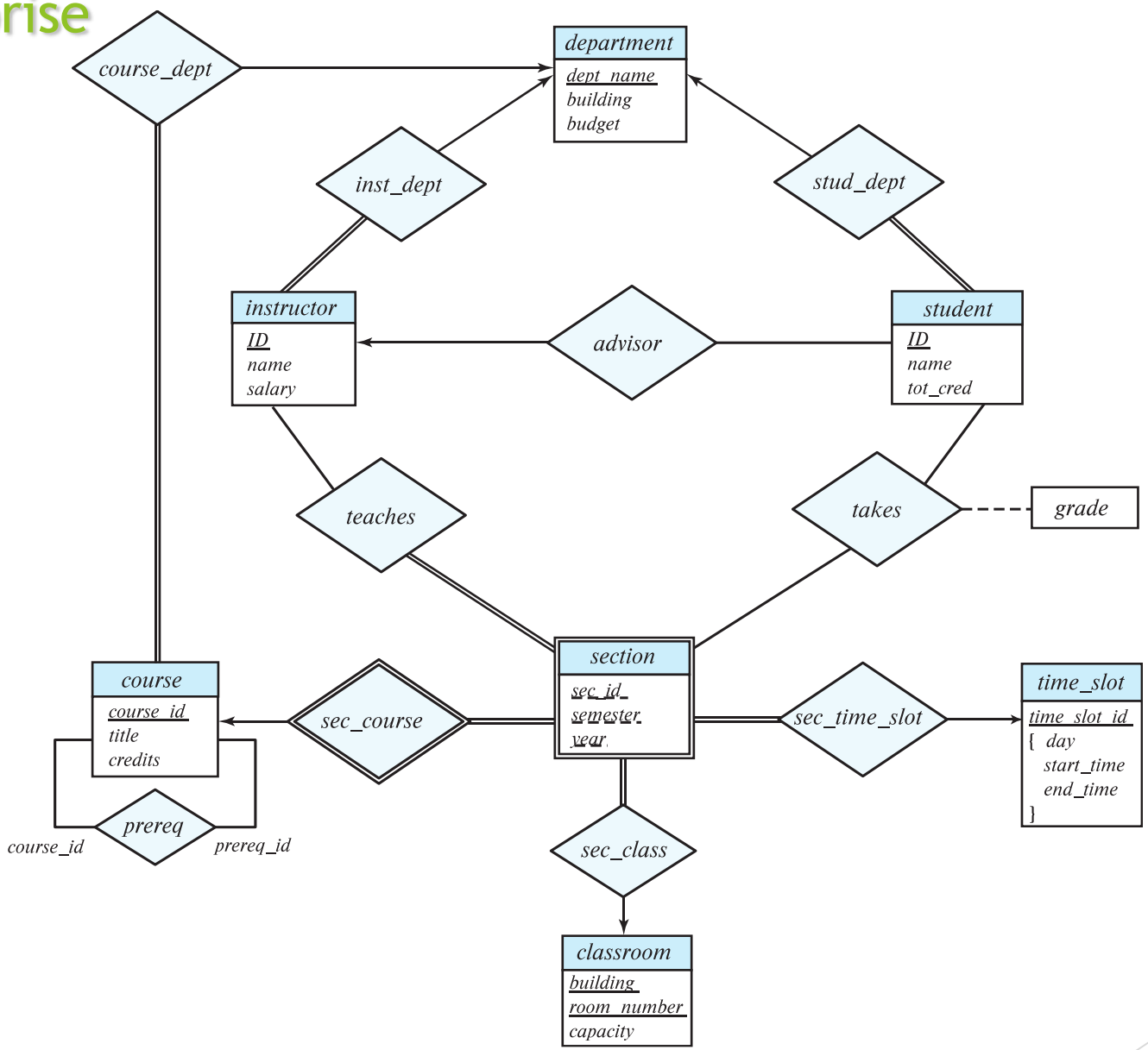


# Entity-Relationship model: Different Attributes of Student

- E-R Diagram with a Ternary Relationship



# Entity-Relationship model: E-R Diagram for a University Enterprise

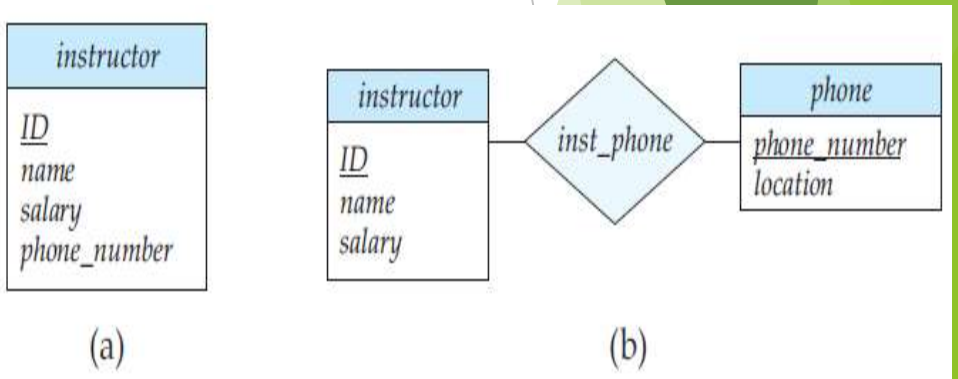


# Entity-Relationship model: Design Issues

- The notions of an entity set and a relationship set are not precise, and it is possible to define a set of entities and the relationships among them in a number of different ways. The basic designs issues are

## 1. Use of Entity Sets versus Attributes

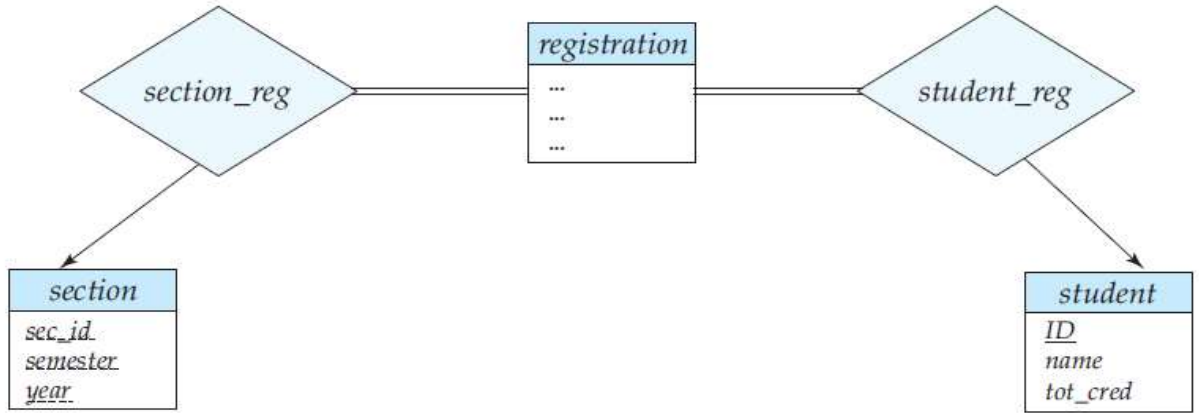
- Consider the entity set instructor with the additional attribute phone number.
- It can easily be argued that a phone is an entity in its own right with attributes phone number and location; the location may be the office or home where the phone is located, with mobile (cell) phones perhaps represented by the value “mobile.” We create:
- A *phone* entity set with attributes *phone number* and *location*.
- A relationship set *inst_phone*, denoting the association between *instructors* and the *phones* that they have.



# Entity-Relationship model: Design Issues

## 2. Use of Entity Sets versus Relationship Sets

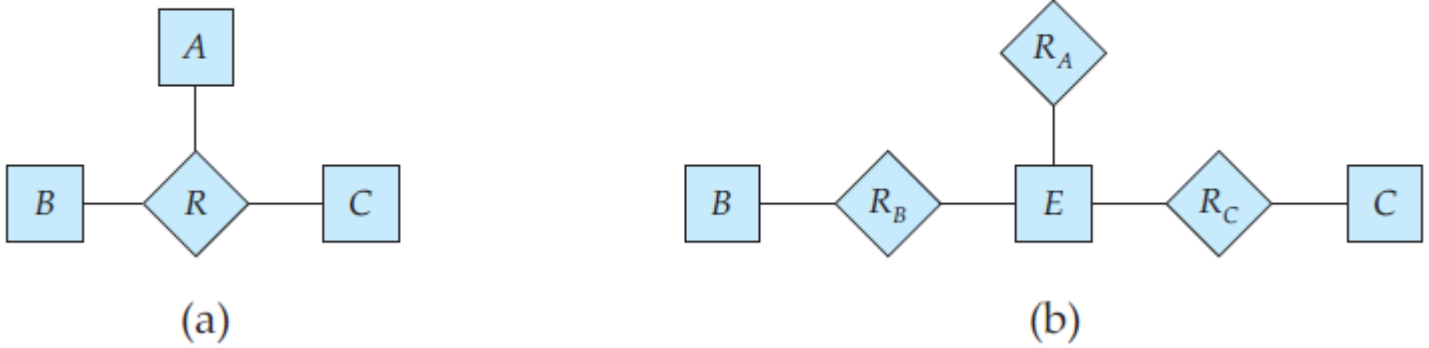
- **It is not always clear whether an object is best expressed by an entity set or a relationship set.**
- We showed the entity sets *section* and *student in earlier example* with the *takes* relationship set replaced by one entity set and two relationship sets:
- *registration*, the entity set representing course-registration records.
- *section reg*, the relationship set relating *registration* and *course*.
- *student reg*, the relationship set relating *registration* and *student*.



# Entity-Relationship model: Design Issues

## 3. Binary versus *n*-ary Relationship Sets

- Relationships in databases are often binary.
- Some relationships that appear to be non-binary could actually be better represented by several binary relationships.



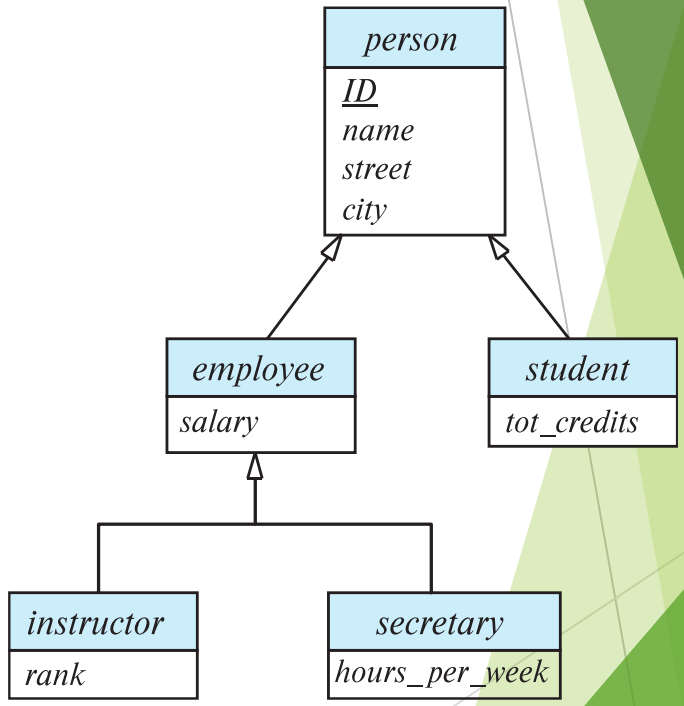
# Entity-Relationship model: Design Issues

## 4. Placement of Relationship Attributes

- The cardinality ratio of a relationship can affect the placement of relationship attributes.
- The attributes of one-to-one or one-to-many relationship sets can be associated with one of the participating entity sets, rather than with the relationship set.
- For instance, let us specify that *advisor* is a one-to-many relationship set such that one instructor may advise several students, but each student can be advised by only a single instructor.
- In this case, the attribute *date*, which specifies when the instructor became the advisor of a student, could be associated with the *student* entity set

# Entity-Relationship model: Extended E-R Features

- **Specialization**
  - Top-down design process;
    - ❖ we designate sub-groupings within an entity set that are distinctive from other entities in the set.
  - These sub-groupings become lower-level entity sets that have attributes or participate in relationships that do not apply to the higher-level entity set.
  - Depicted by a triangle component labeled ISA (e.g., instructor “is a” person).
  - Attribute inheritance – a lower-level entity set inherits all the attributes and relationship participation of the higher-level entity set to which it is linked.



# Entity-Relationship model: Extended E-R Features

- **Generalization**
  - A bottom-up design process –
    - combine a number of entity sets that share the same features into a higher-level entity set.
  - Specialization and generalization are simple inversions of each other; they are represented in an E-R diagram in the same way.
  - The terms specialization and generalization are used interchangeably.

# Entity-Relationship model: Converting ER to Tables

- Convert Entity Sets, Relationships to tables
- Convert all attributes to columns
- Assign all Primary attributes of Entity Sets to Relationship table as columns

# Thank You…!!