



# ENTITY RELATIONSHIP DIAGRAM

Prof Dr Melike Sah Direkoglu  
Software Engineering,  
[msah@ciu.edu.tr](mailto:msah@ciu.edu.tr)

Slide Credit – Benefitted from the slides of Yasemin Bay Ayzeren and Database System Concepts Book (Silberschatz, Korth, Sudarshan)

# Entity Relationship Model/Diagram

- Entity Relationship Model (ERM) or Entity Relationship Diagram (ERD) is a data model or diagram for high level descriptions of conceptual data models.
- ERDs provide a graphical notation for representing data models in terms of entities, attributes and relationships.

## Basic Concepts

The ER model employs three basic notations: entity sets, relationship sets and attributes.

# Entity Relationship Diagram (ERD)

Reminder of basic notations in ERD:

- Rectangles : Represent entity sets.
- Diamonds : Represent relationship sets.
- Ellipses : Represent attributes.
- Double ellipses: Represent multivalued attributes
- Dashed ellipses: Represent derived attributes
- Lines : Link attributes to entity sets and entity sets to relationship set.
- Double lines: Indicate total participation of an entity in a relationship set

# ERD Entities

**Entity:** An entity is a ‘thing’ or ‘object’ in the real world.

Entities are the principle data object about which information is to be collected. Entities are represented by **rectangles** in ER diagram.

An **entity set** is a set of entities of the same type that share the same properties. E.g set of all persons, companies, trees, holidays

Ex: customer → the set of all people having an account at Bank ...

student → the set of all people registered in a university.

course → the set of all courses taught in a university

customer

student

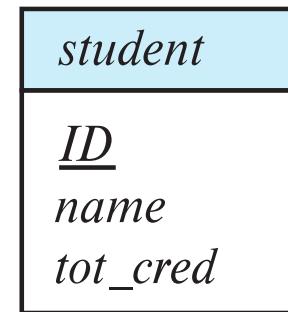
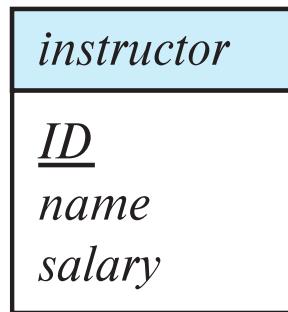
course

# Entity Sets

- An entity is represented by a **set of attributes**; i.e., descriptive properties possessed by all members of an entity set.
- Example:  
 $\textit{instructor} = (\textit{ID}, \textit{name}, \textit{salary})$   
 $\textit{course} = (\textit{course\_id}, \textit{title}, \textit{credits})$
- A subset of the attributes form a **primary key** of the entity set; i.e., uniquely identifying each member of the set.

# Representing Entity sets in ER Diagram

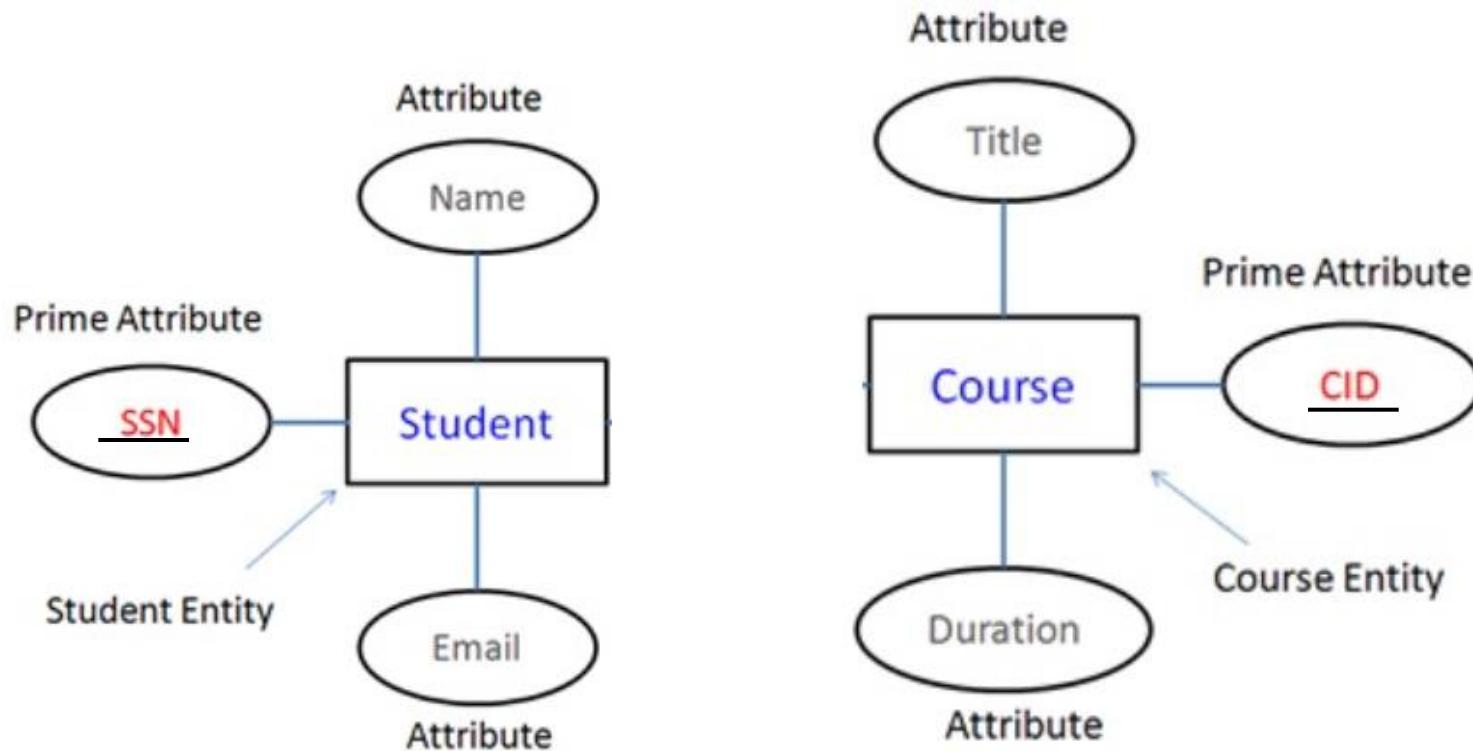
- Entity sets can be represented graphically as follows:
  - Rectangles represent entity sets.
  - Attributes listed inside entity rectangle
  - Underline indicates primary key attributes



# ERD Attributes

In ERD attributes are represented by **ellipses**.

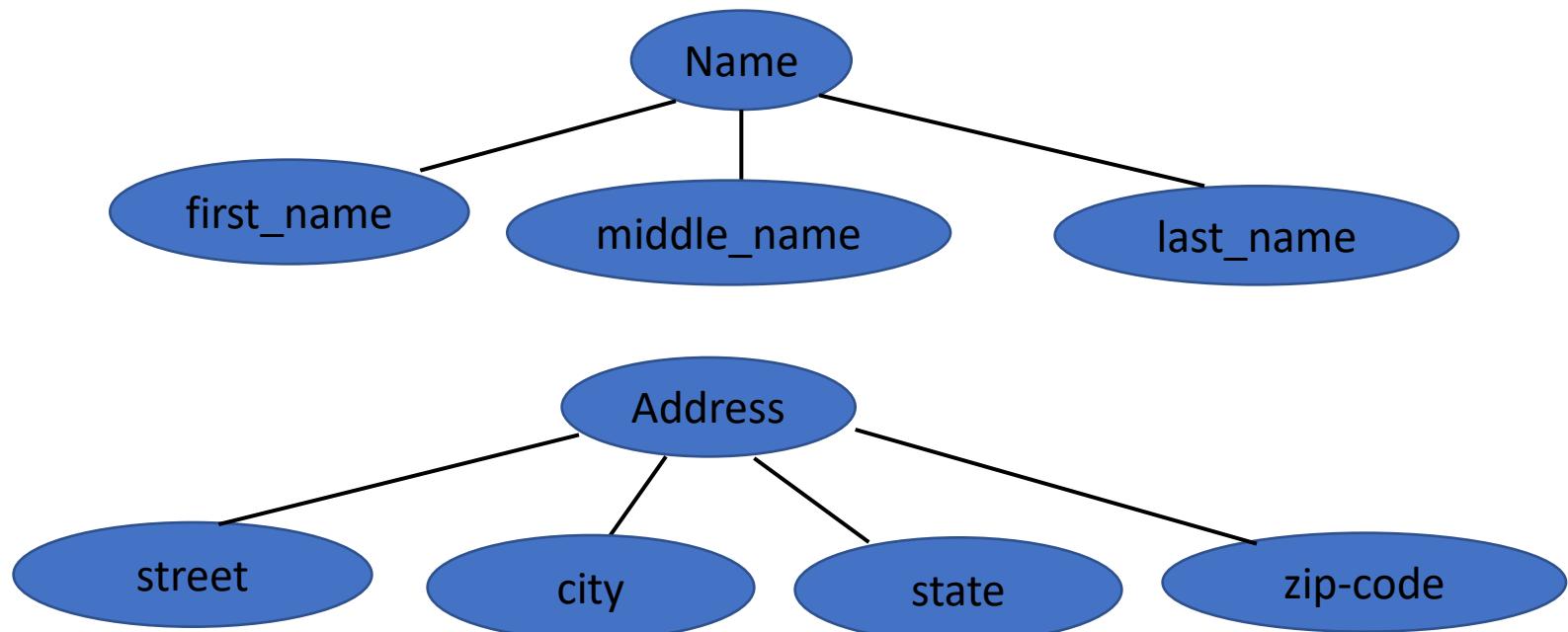
In ERD, one of the attributes is selected as the **primary key** of the entity. Primary key is indicated by **underlining the attribute**. Primary key assures the value in the specific table column is unique.



# ERD Attribute Types

## 1- Simple and Composite Attributes

- **Simple Attributes:** Not divided into subparts, for example **student\_number** is a simple attribute
- **Composite Attributes:** Can be divided into subparts.



# Complex Attributes in ER Diagram (alternative)

<i>instructor</i>
<i>ID</i>
<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>

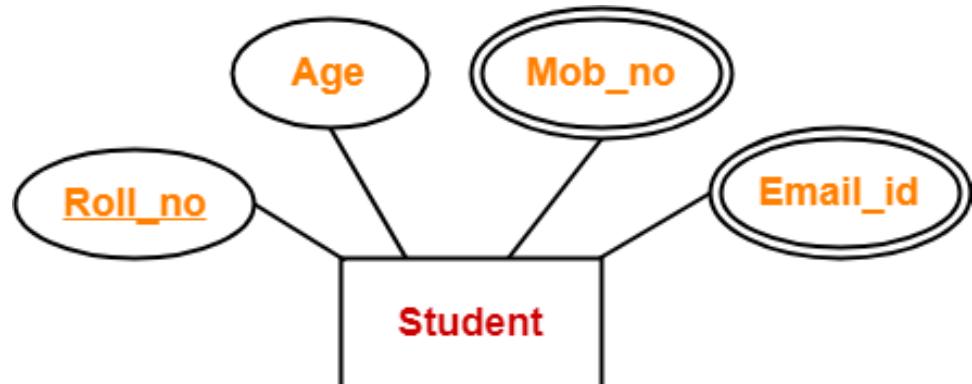
# ERD Attribute Types

## 2- Single Valued and Multivalued Attributes

student\_number  
department  
city } singlevalued

Multivalued attributes are represented by **double ellipses**

email\_id → multivalued  
telephone\_no → multivalued  
(it is possible to have multiple  
Telephone numbers or emails)



# ERD Attribute Types

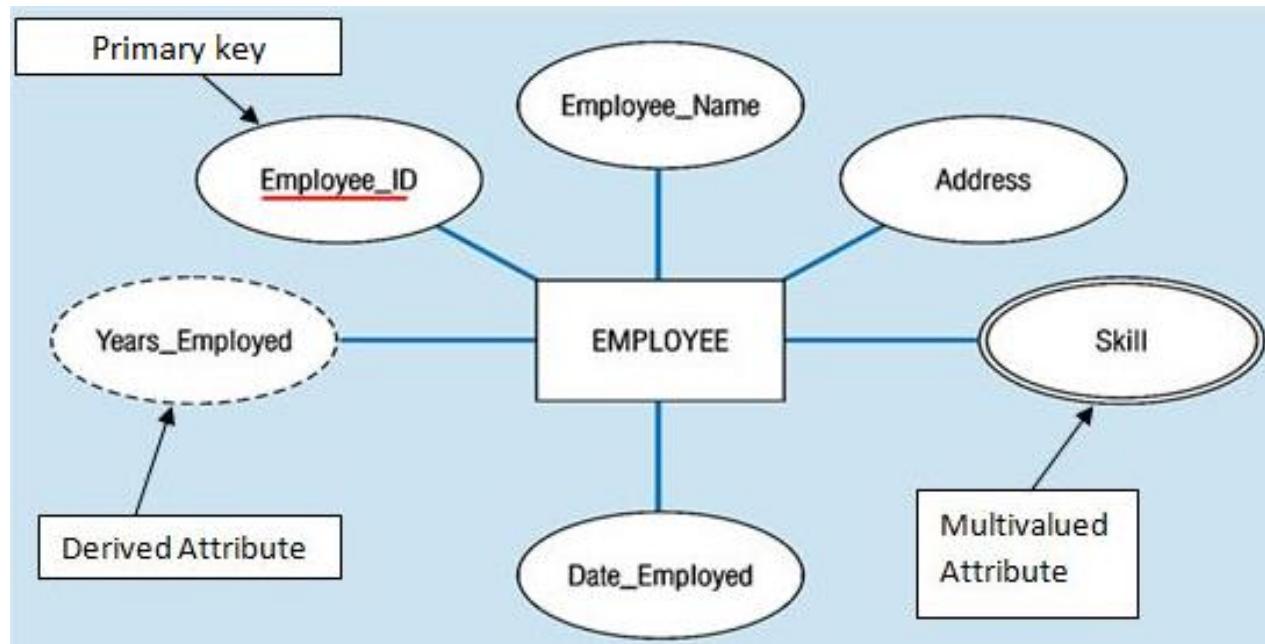
## 3- Derived Attributes

Can be derived from the values of other attributes.

years\_employed → current\_date – start\_date

net\_salary → gross\_salary – VAT

Derived attributes are represented by **dashed ellipses**



# ERD Attribute Types

## 4- Null Value Attributes

An attribute takes a null value when an entity does not have a value for it.

For example, if the participant has no comments → comments=null,

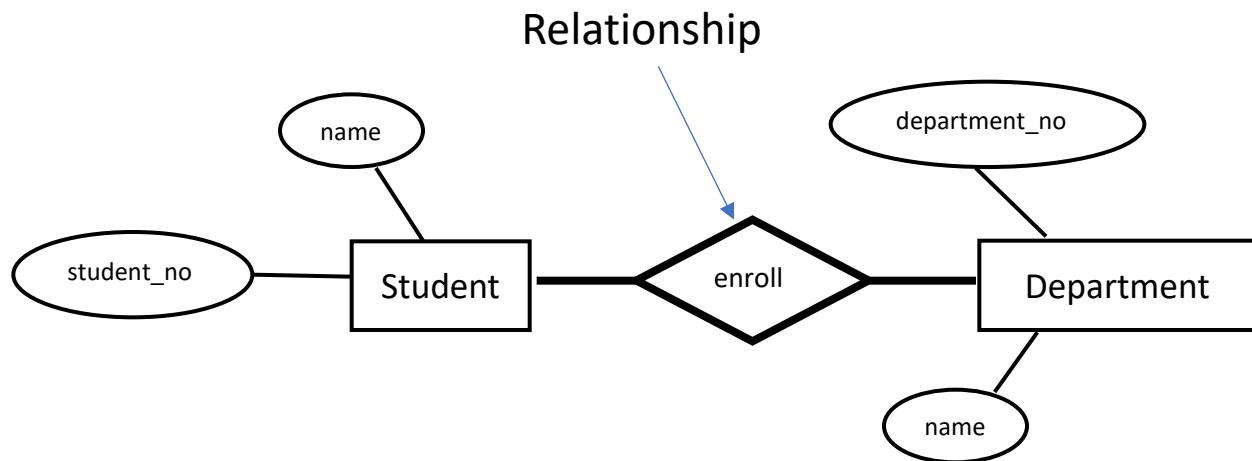
If the person lives in a house → apartment\_no=null

....

# ERD Relationships

A relationship represents an association between two or more entities.

Relationships are represented by **diamond** shape.



Student is enrolled in a department.

# 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

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

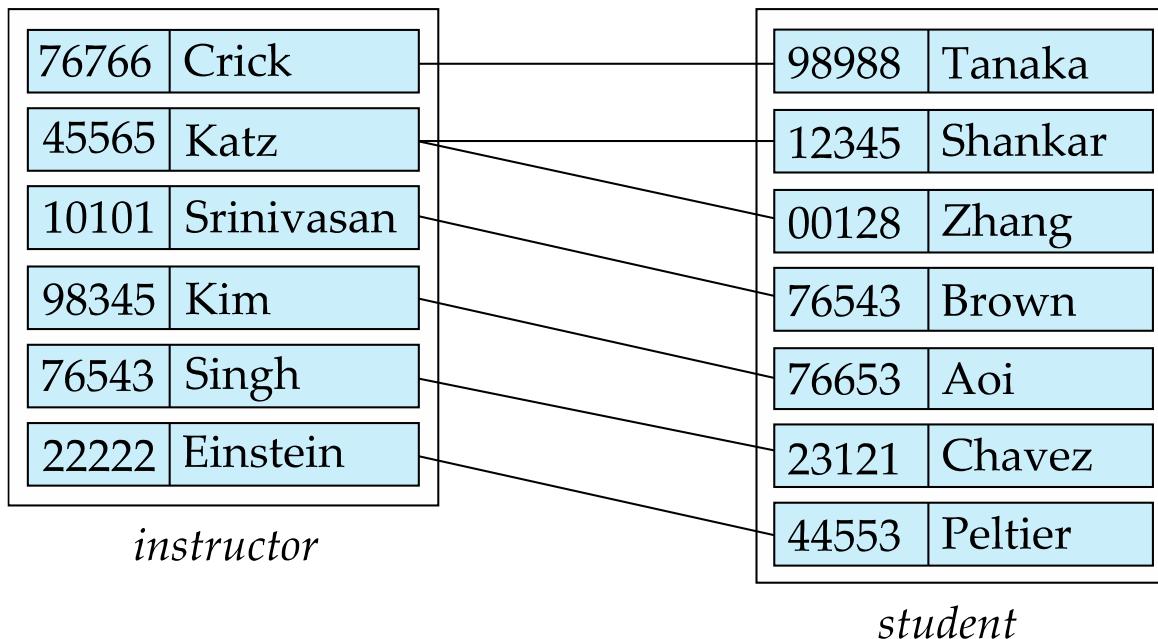
where  $(e_1, e_2, \dots, e_n)$  is a relationship

- Example:

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

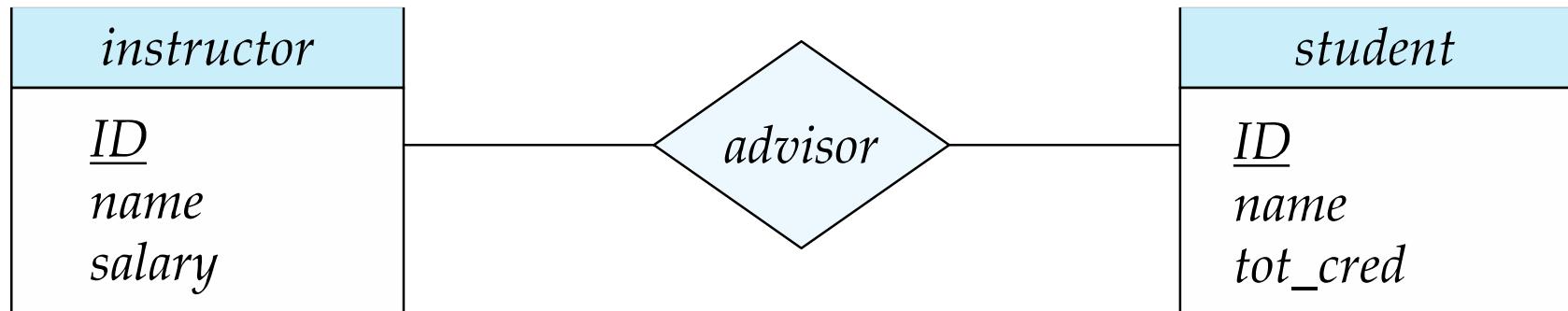
# Relationship Sets (Cont.)

- 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.



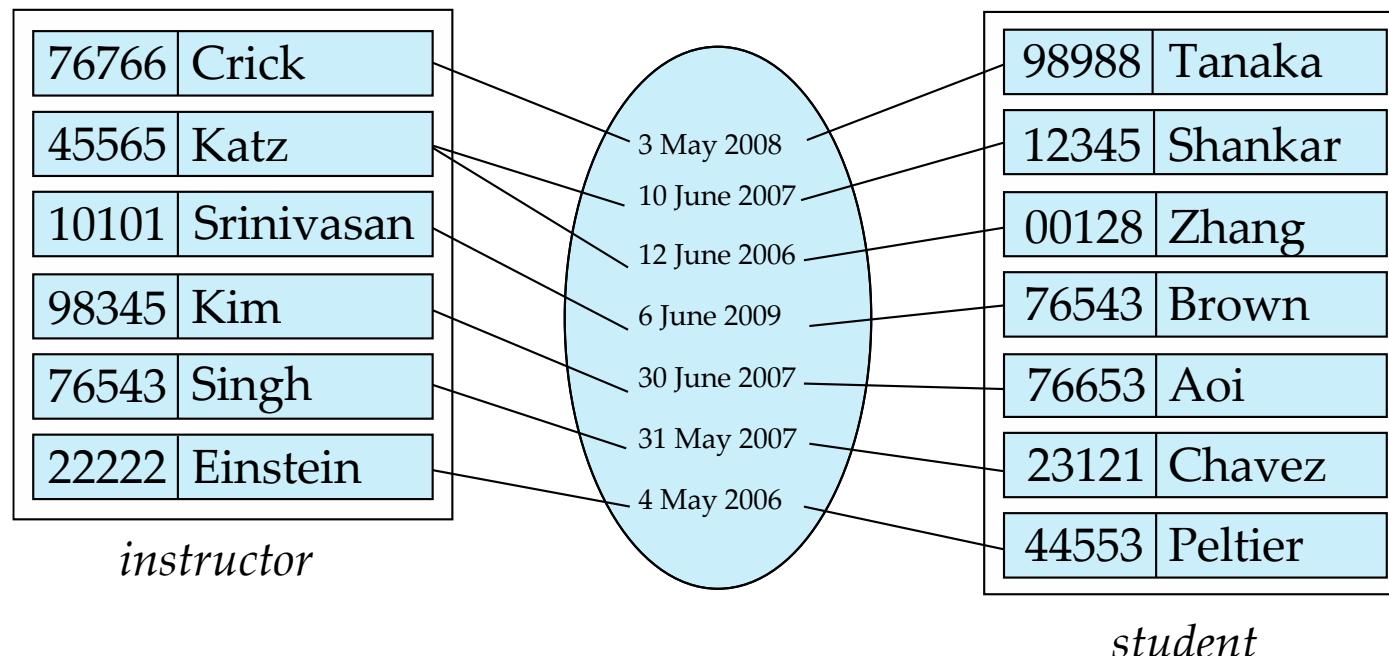
# Representing Relationship Sets via ER Diagrams

- Representation of advisor relationship in the ER diagram.

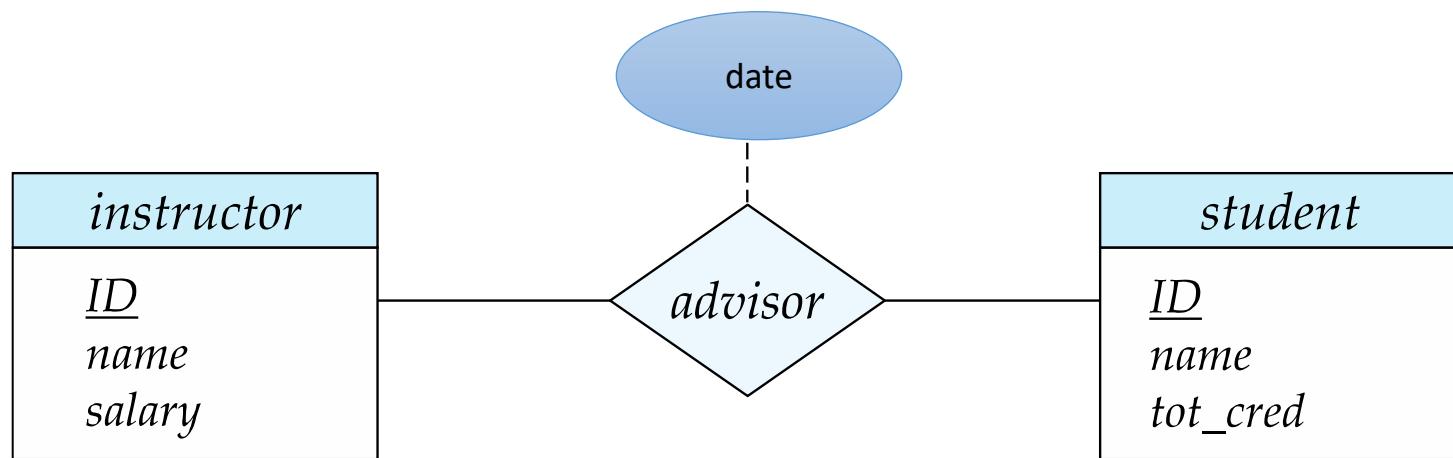


# Relationship Sets (Cont.)

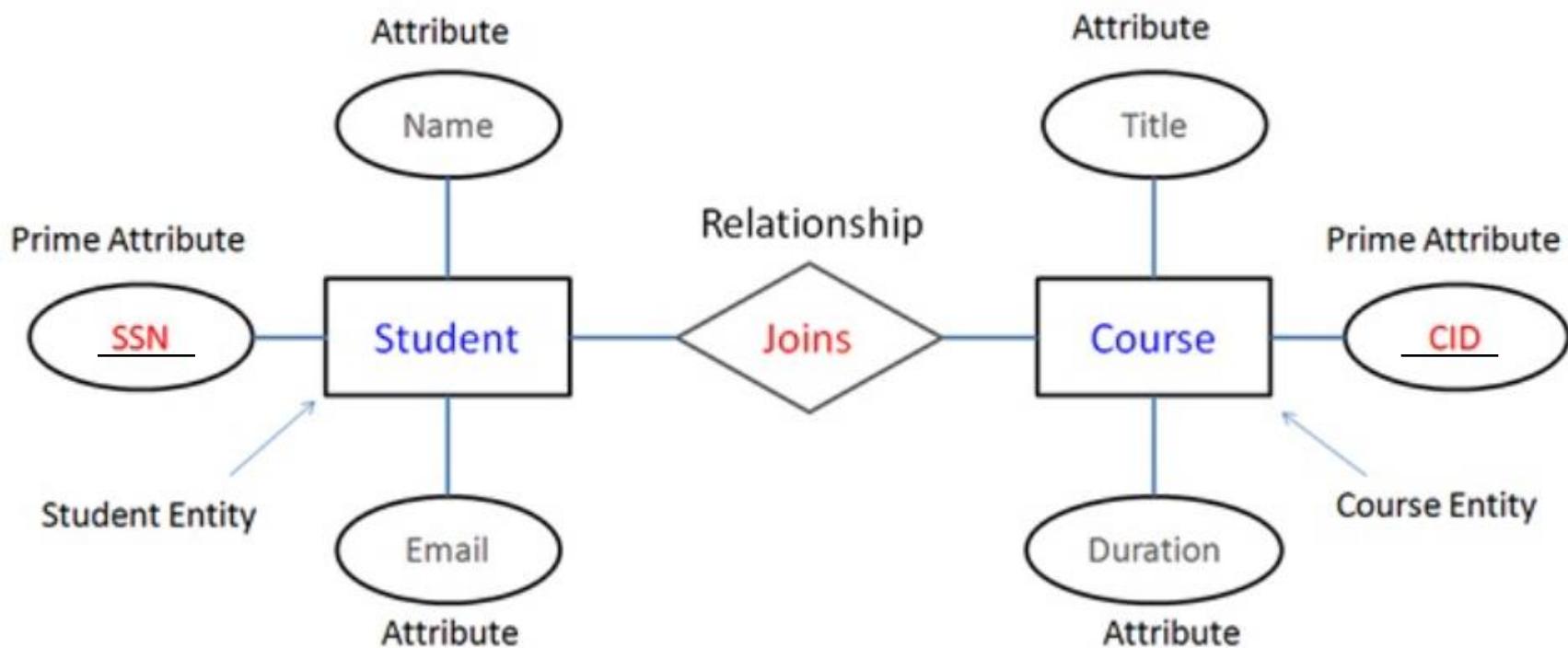
- An **attribute** can also be associated with a relationship set.
- For instance, the *advisor* relationship set between entity sets *instructor* and *student* may have the **attribute date** which tracks when the student started being associated with the advisor



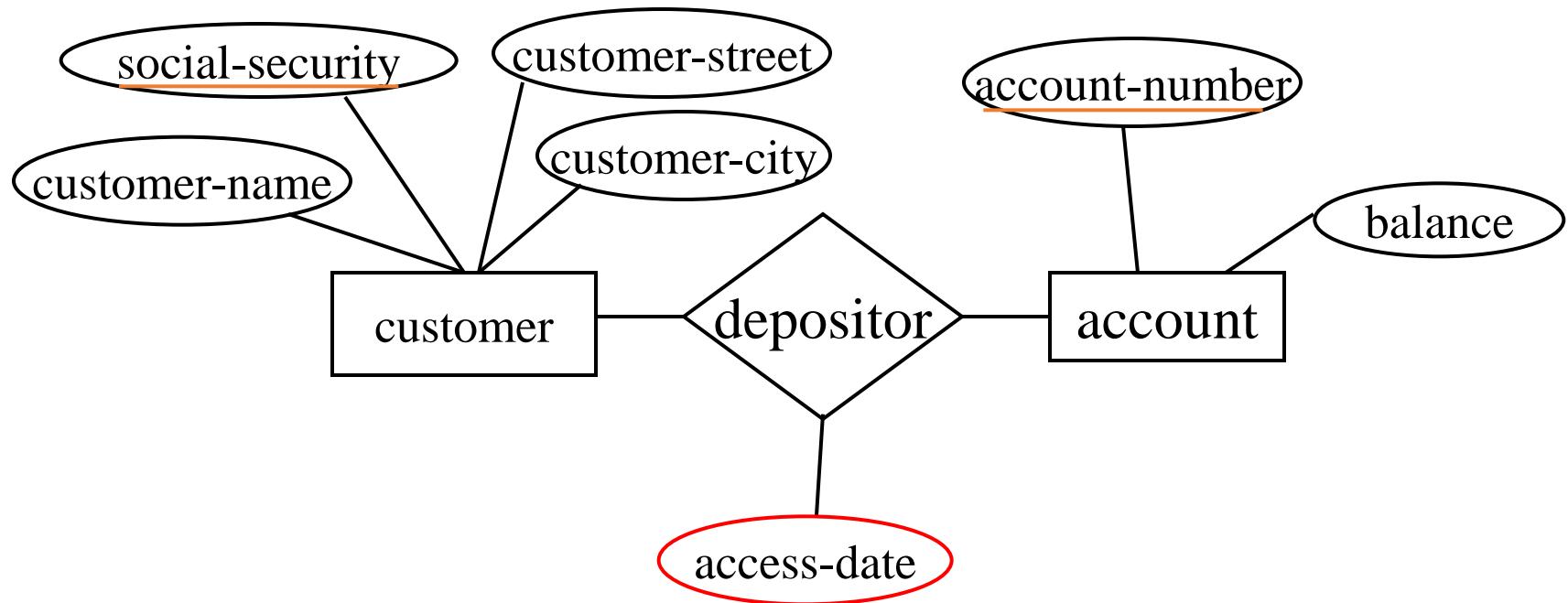
# Relationship Sets with Attributes in the ER Diagram



# More ERD Examples



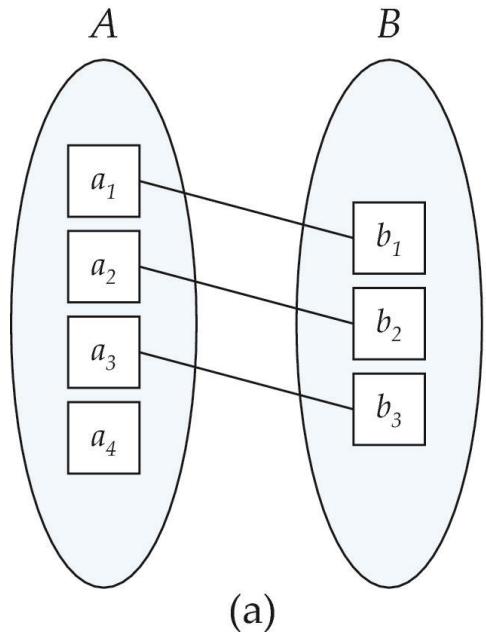
# More ERD Examples



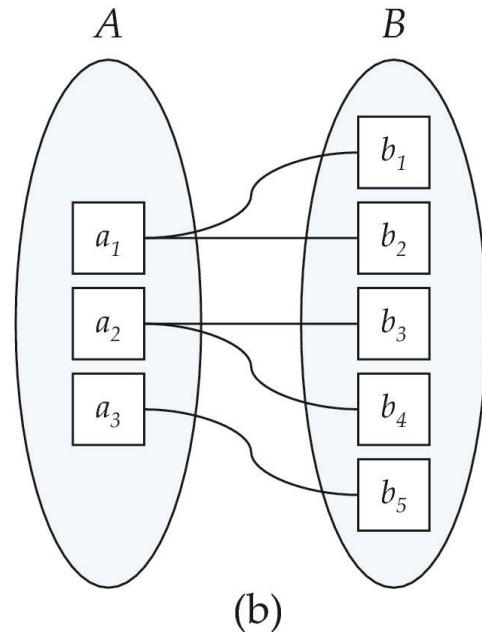
# Mapping Cardinality Constraints

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

# Mapping Cardinalities



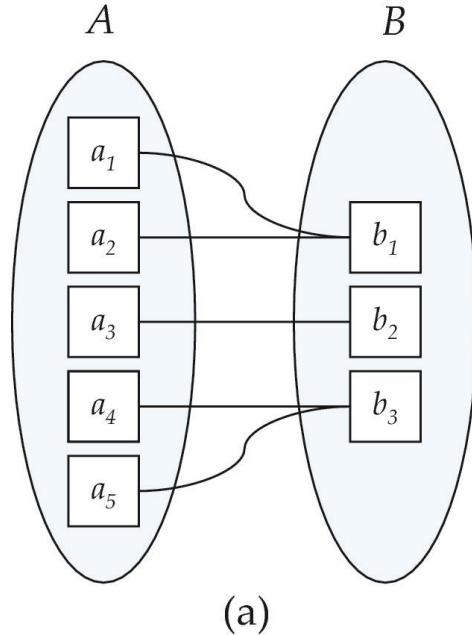
One to one



One to many

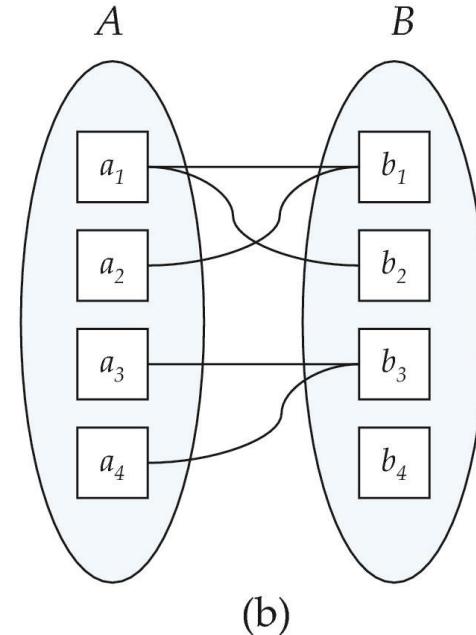
Note: Some elements in  $A$  and  $B$  may not be mapped to any elements in the other set

# Mapping Cardinalities



(a)

Many to one



(b)

Many to many

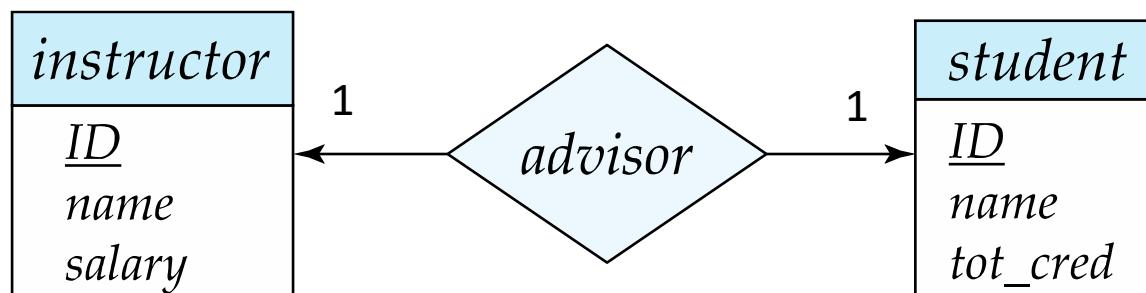
Note: Some elements in A and B may not be mapped to any elements in the other set

# Mapping Cardinalities: One to One

**One to One:** An entity in A is associated with at most one entity in B and an entity is associated with at most one entity in A.

## Example:

- A student is associated with at most one *instructor* via the relationship *advisor*
- A *student* is associated with at most one *department* via *stud\_dept*



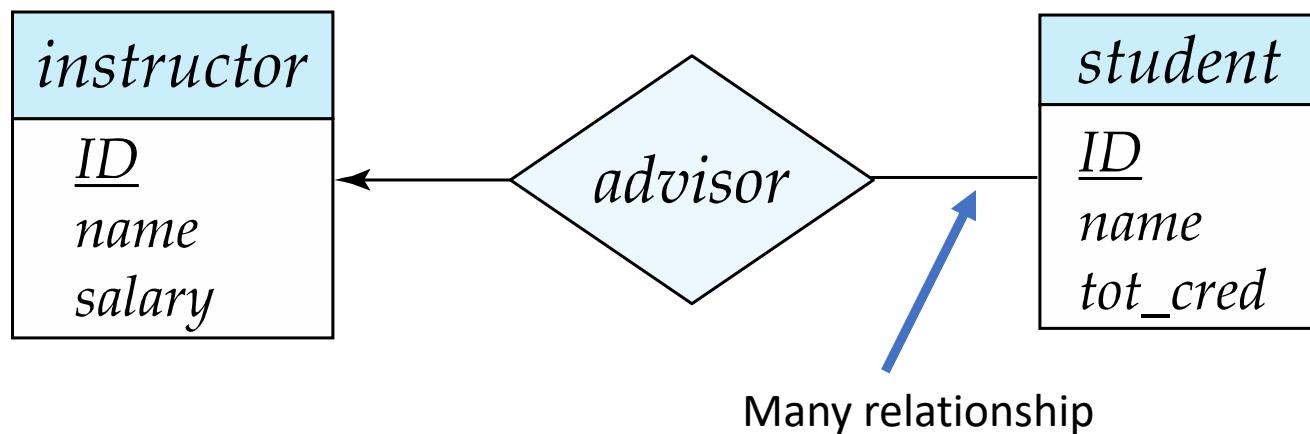
# Mapping Cardinalities: One to Many

**One to Many :** An entity in A is associated with any number of entities in B. An entity in B is associated with at most one entity in A.

**Example:**

**One to many** relationship between an *instructor* and a *student*

- an instructor is associated with several (including 0) students via *advisor*
- a student is associated with at most one instructor via advisor,



# One to Many Mapping

A doctor can write zero, one or many prescriptions.

A prescription can be written only by one doctor.



# Mapping Cardinalities: Many to One

**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 of entities in A.

## Example:

A course can be given in only one department.

In one department there can be many courses given.



# Mapping Cardinalities: Many to Many

**Many to Many :** Entities in A and B are associated with any number of entities from each other.

A student may take many courses.

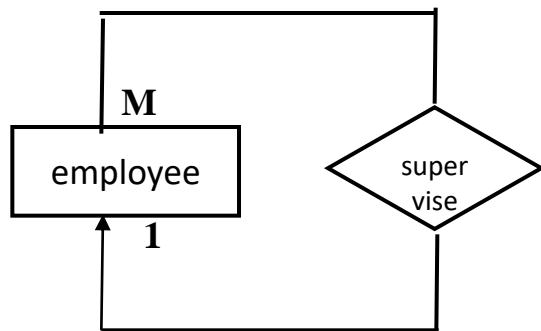
A course may be taken by many students.



# Degree of a Relationship Sets (Arity of the Relations)

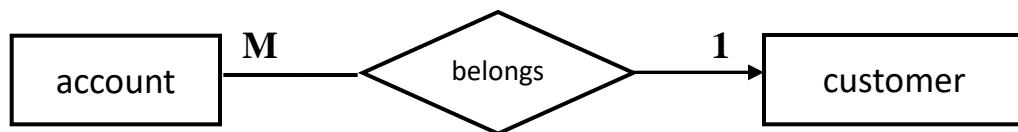
- **Unary** relationship
  - Involve one entity set (degree one)
- **Binary** relationship
  - involve two entity sets (degree two).
  - most relationship sets in a database system are binary.
- **Non-binary relationships:** Relationships between more than two entity sets are rare but possible. These relationships are called **ternary relationships**
  - Example: *students* work on research *projects* under the guidance of an *instructor*.

# Unary Relationship



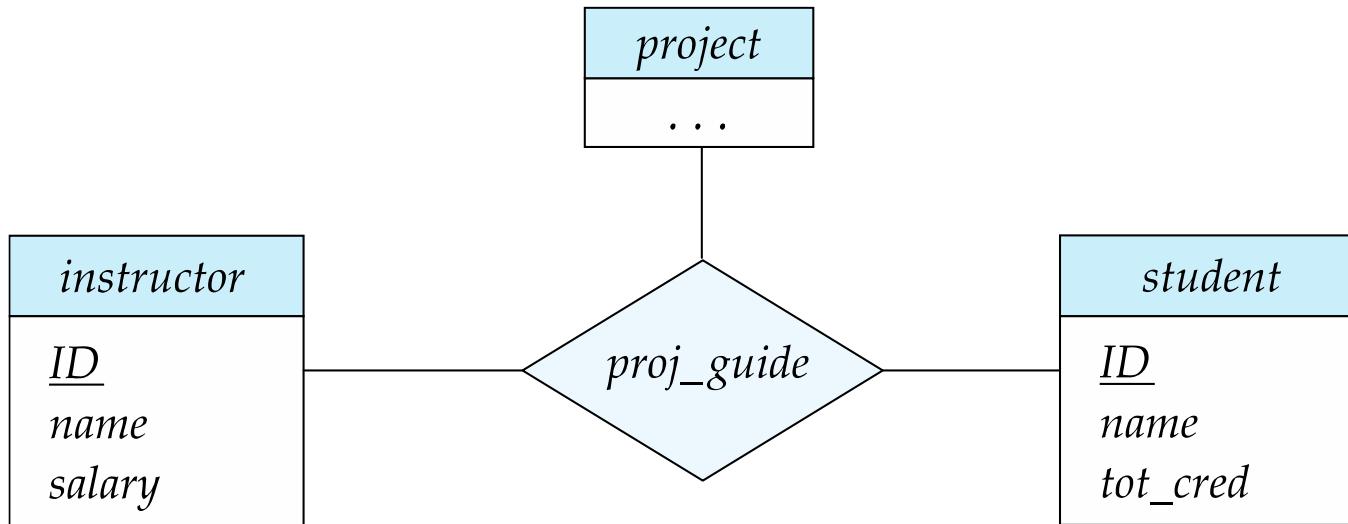
Supervise relationship **contains one** entity set; employee

# Binary Relationship



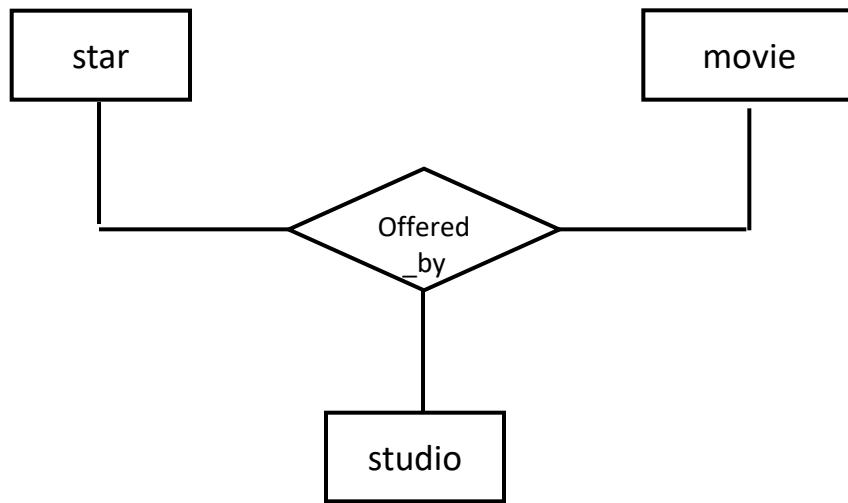
Belongs relationship **contains two** entity sets; account and customer

# Ternary Relationship



Proj\_guide relationship contains three entity sets; star, movie, studio

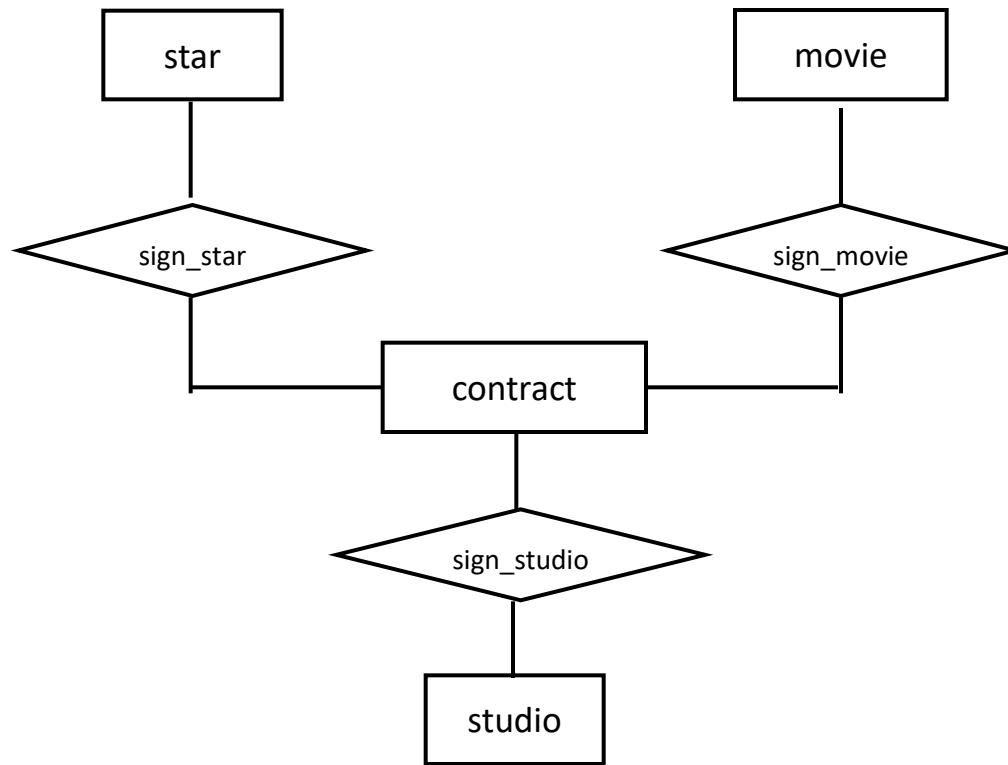
# Ternary Relationship



Offered\_by relationship **contains three** entity sets; star, movie, studio

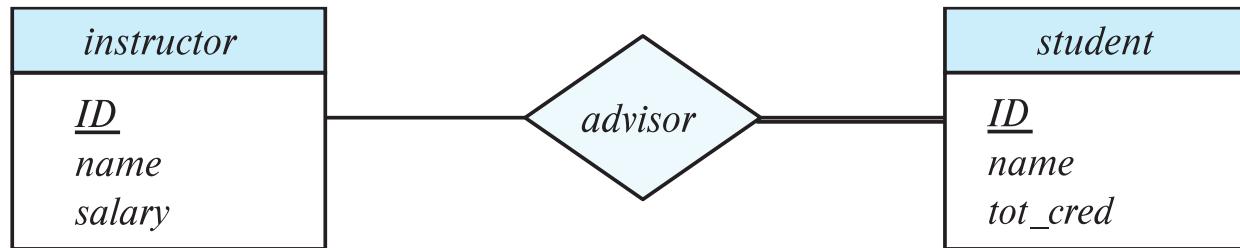
# Converting Non-Binary Relationships to Binary Form

- New binary relationships can be added to convert ternary relationships into binary relationships.



# Total and Partial Participation

- **Total participation** (indicated by double line): every entity in the entity set participates in at least one relationship in the relationship set



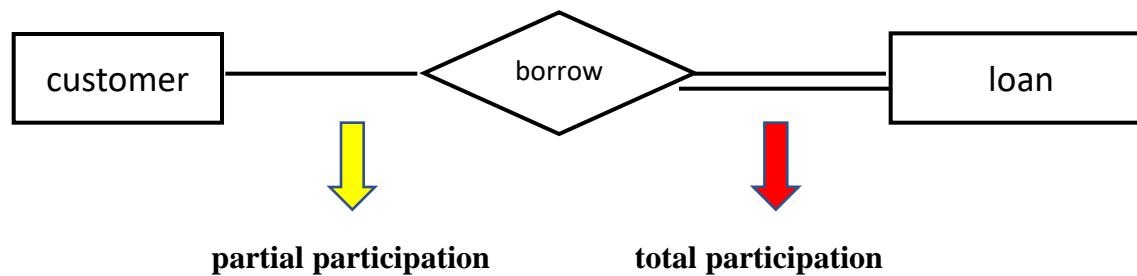
participation of *student* in *advisor* relation is total, meaning every *student* must have an associated instructor

- **Partial participation:** some entities may not participate in any relationship in the relationship set
  - Example: participation of *instructor* in *advisor* is partial, meaning that an instructor may not be an advisor of any student

# Total Participation

Ex: Participation of loan in borrower.

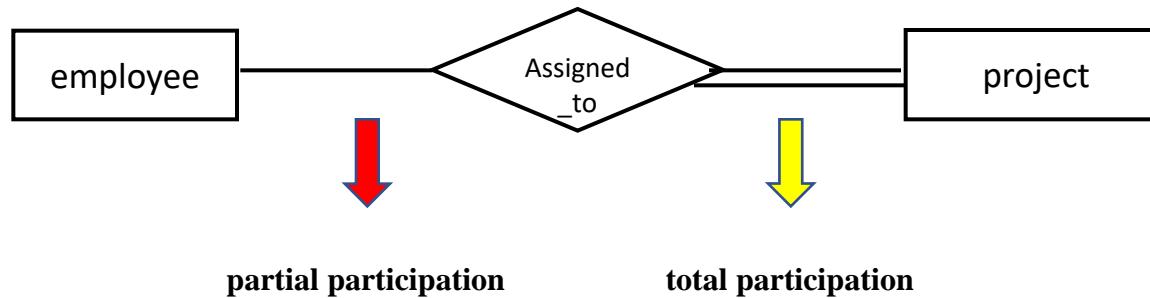
**Participation of loan in borrow is total;** every loan must have a customer associated to it via borrower.



# Entity Relationship Diagram

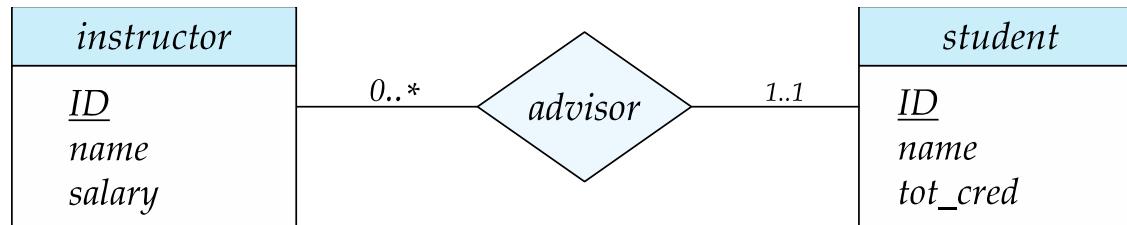
Participation of project in assigned\_to is total;

- Every project must have an employee assigned to it.
- But participation of an employee in a project is partial.
- A secretary of a company is an employee bu do not participate in any project.



# Notation for Expressing More Complex Constraints

- 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.
- Example



- Instructor can advise 0 or more students. A student must have 1 advisor; cannot have multiple advisors

# ERD Keys (Identification)

A key is an attribute or group of attributes, that assumes a unique value for each entity.

Ex:

<u>Entity</u>	<u>Key</u>
student	→ student_id
employee	→ employee_no
invoice	→ invoice_no
course	→ course_code

# Candidate Key

**Candidate Key** : is one of a number of keys that may serve as the primary key of an entity.

Employee

employee\_id → **candidate key**

name

surname

tel

address

SSN ( social security number ) → **candidate key**

# Primary Key

- **Primary Key (PK):** is a candidate key that will most commonly be used uniquely to identify a single entity. Table allows only **one** primary key.

## Employee

employee\_id → **primary key**

name

surname

tel

address

SSN ( social security number ) → candidate key

# Foreign Key

- **Foreign Key (FK)** : is a primary key of an entity that is used in other entities to identify instances of a relationship; FK is generally used to build a relationship between two tables.

## Student

student\_id ( PK )

name

surname

cgpa

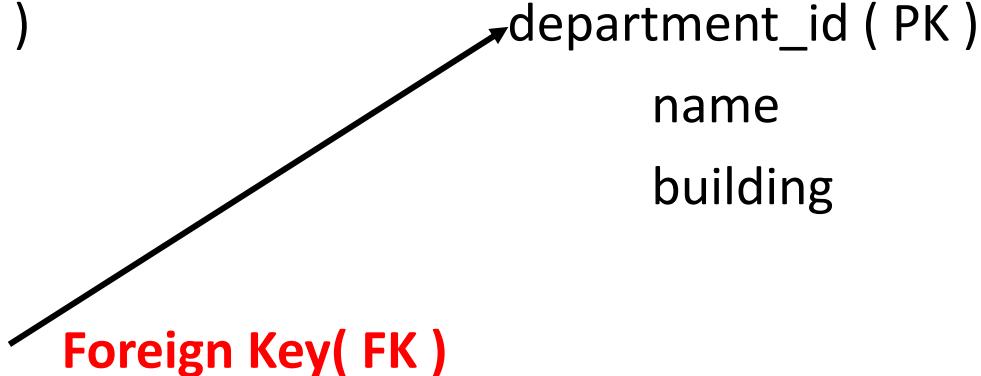
department\_id

## Department

department\_id ( PK )

name

building



# Foreign Key – Referential Integrity

When one table has a foreign key to another table, the concept of referential integrity states that **you may not add a record to the table that contains the foreign key unless there is a corresponding record in the linked table.**

std_no	name	surname	cgpa	dept_id
123	John	Smith	2,18	1
134	Jane	Hayes	3,56	1
145	Mary	Turner	1,75	2
156	Michael	Wayne	4,00	3
178	Mike	King	3,41	5

dept_id	dept_name
1	CMPE
2	ISYE
3	MISY
4	ITEC
5	COMP

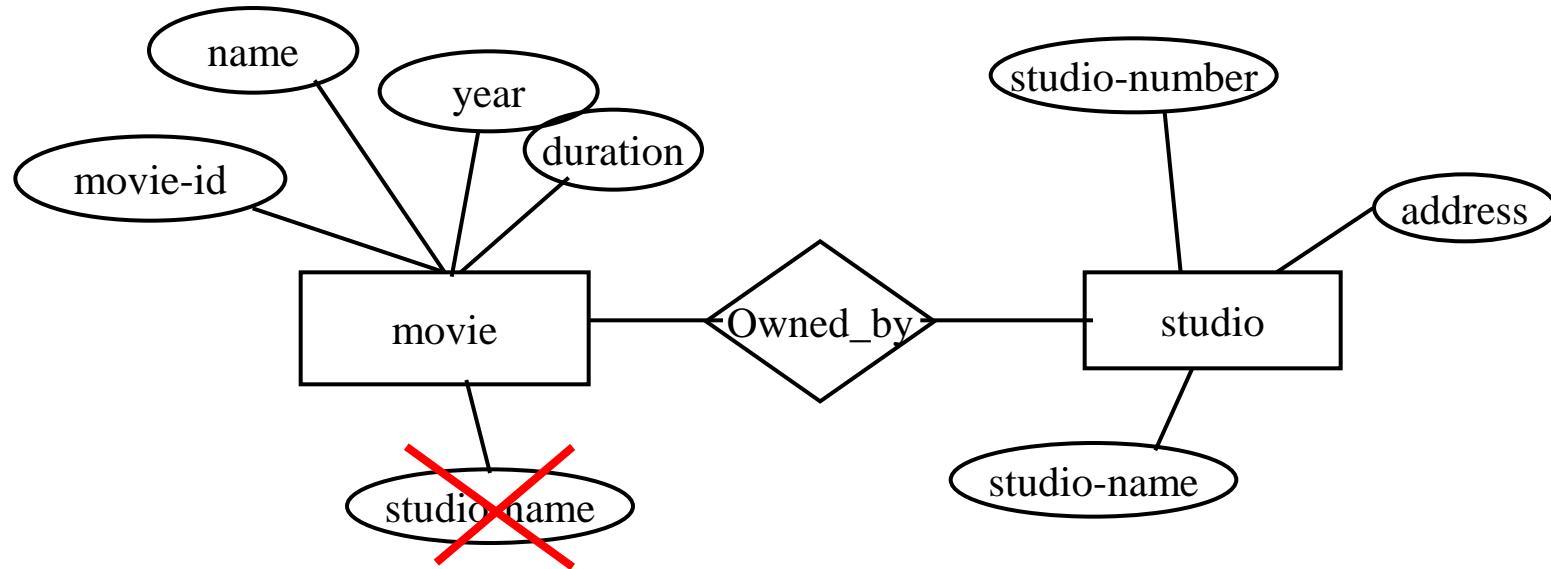


can not take any value other than 1,2,3,4,5 because of the foreign key

# Redundancy

## Avoid Uncontrolled Redundancy

Do not repeat the same value more than once unless it is absolutely necessary.

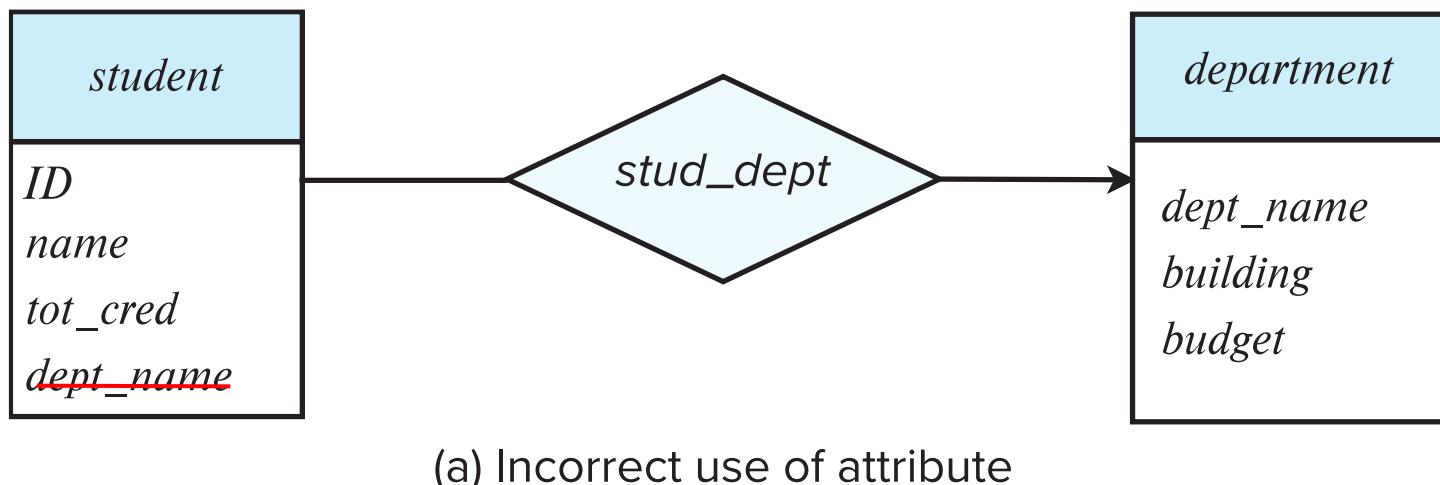


What is wrong with redundancy?

- waste of space
- inconsistencies may arise when updating /deleting information.

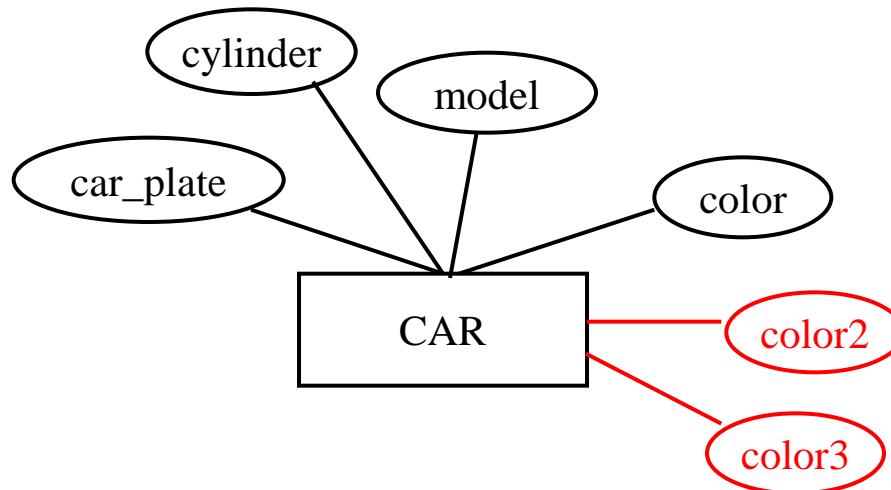
# Redundant Attributes (Cont.)

- The attribute *dept\_name* in *student* below replicates information present in the relationship and is therefore redundant and needs to be removed.



# Attribute Selection

**Pick the Right and Meaningful Attributes!**



You can do this only if the maximum number of colour is 3 and if most of the cars have exactly 3 colours.

# ERD – Exercise 1

**Draw an entity relationship diagram (ERD) with the instructions given below:**

- In a private hospital there are many doctors working
- Doctors examine many patients
- Each patient examined by only one doctor
- Each doctor writes many prescriptions
- A prescription is written by only one doctor

# ERD – Exercise 2

**Draw an ERD for XYZ company, where details are given below:**

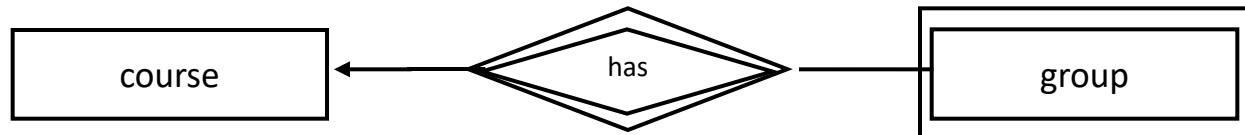
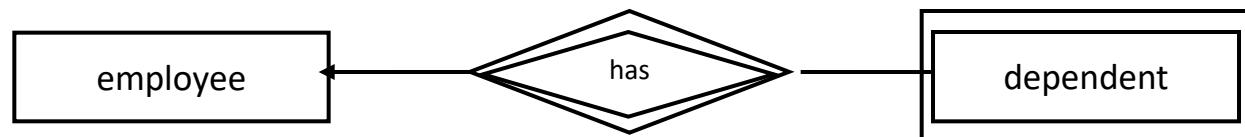
- XYZ company has several departments.
- There are many employees working in each department but an employee can work in only one department.
- Each employee working in this company has a job description( ex. programmer, secretary, project manager ... ) and many employees can have the same job descriptions.
- Some employees are assigned to projects.
- There may be many employees assigned to a project.

# Strong Entity Sets vs Weak Entity Sets

- An entity set may not have sufficient attributes to form a primary key. Such an entity set is called **weak entity** set. In other words, entity existence dependent on relationships with other entities.
- An entity set with a **primary key** is called a **strong entity** set.

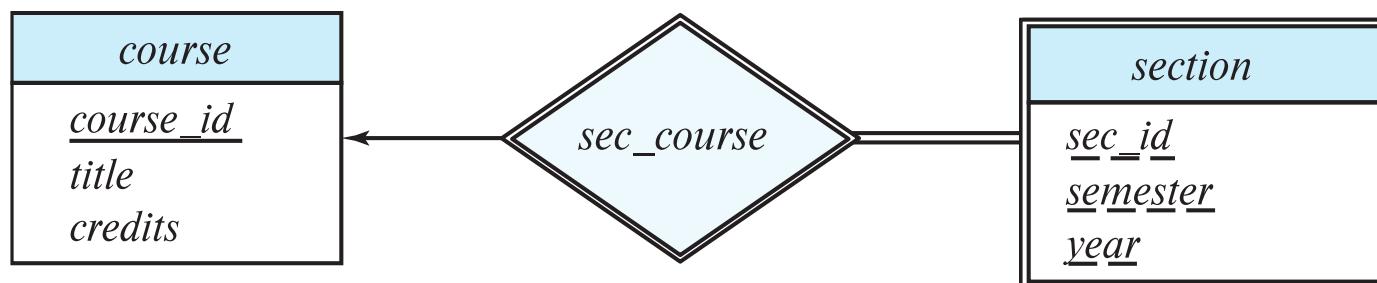
# Weak Entity

Weak entity is an entity that inherits its primary key from more than one other entity set. It is called Weak Entity because its identification is dependent on the existence of other entities.

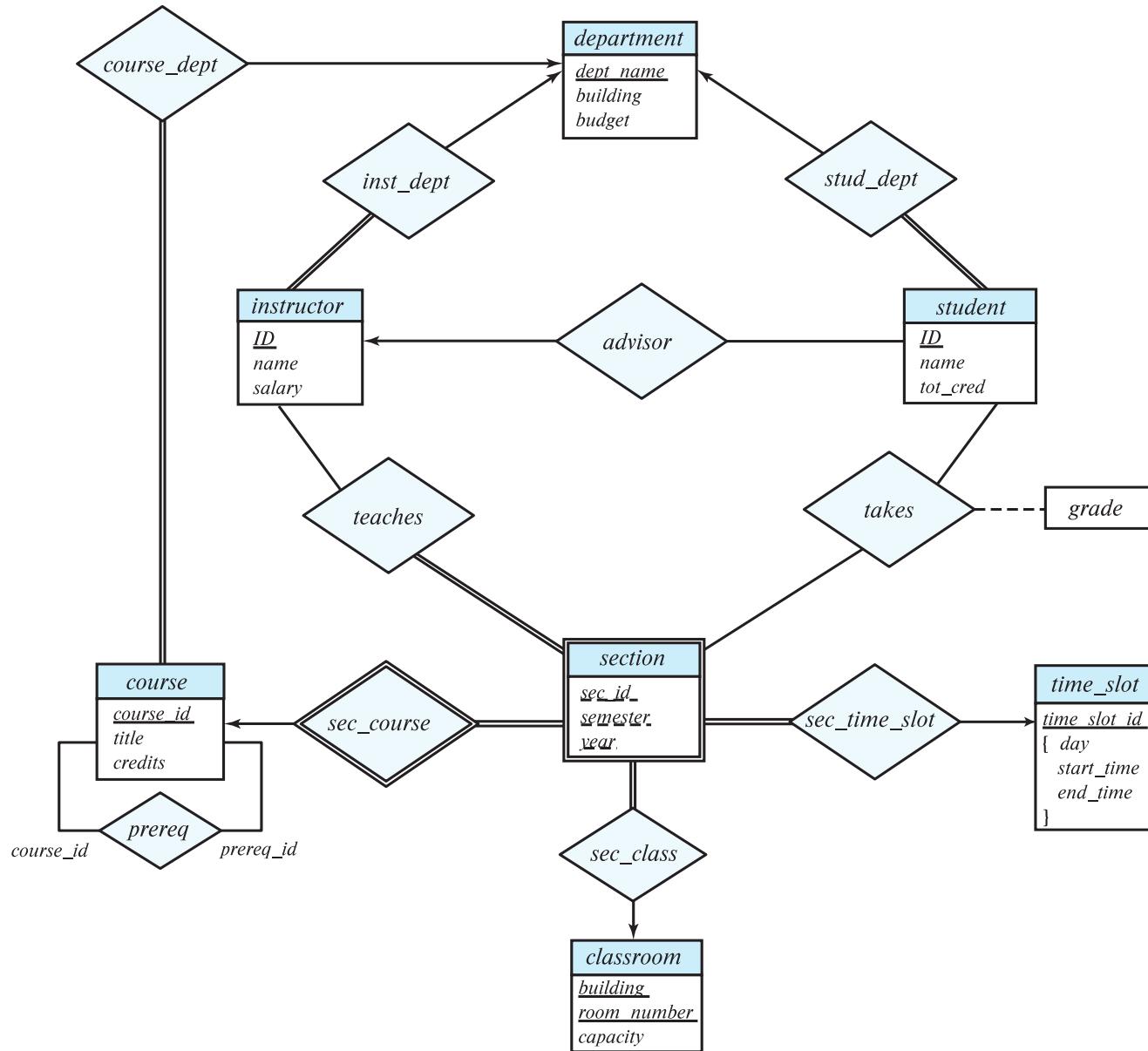


# Expressing Weak Entity Sets

- In ERDs, 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**.
- In addition, since the existence of the weak entity set depends on the relationship, total participation in the relations is required.
- Primary key for ***section – (course\_id, sec\_id, semester, year)***



# ER Diagram for a University Enterprise

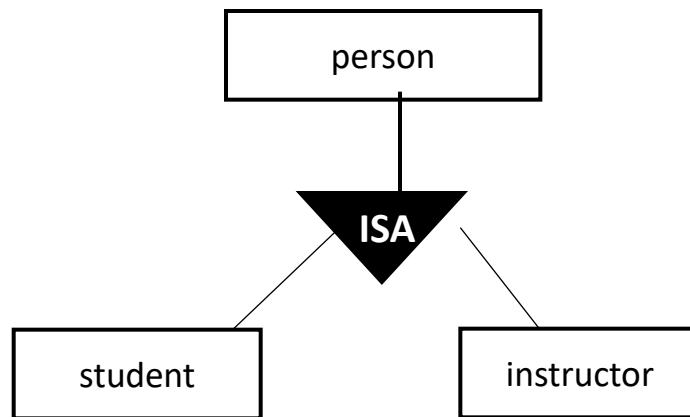


# Extended ER Features

# ERD Specialization (Generalization)

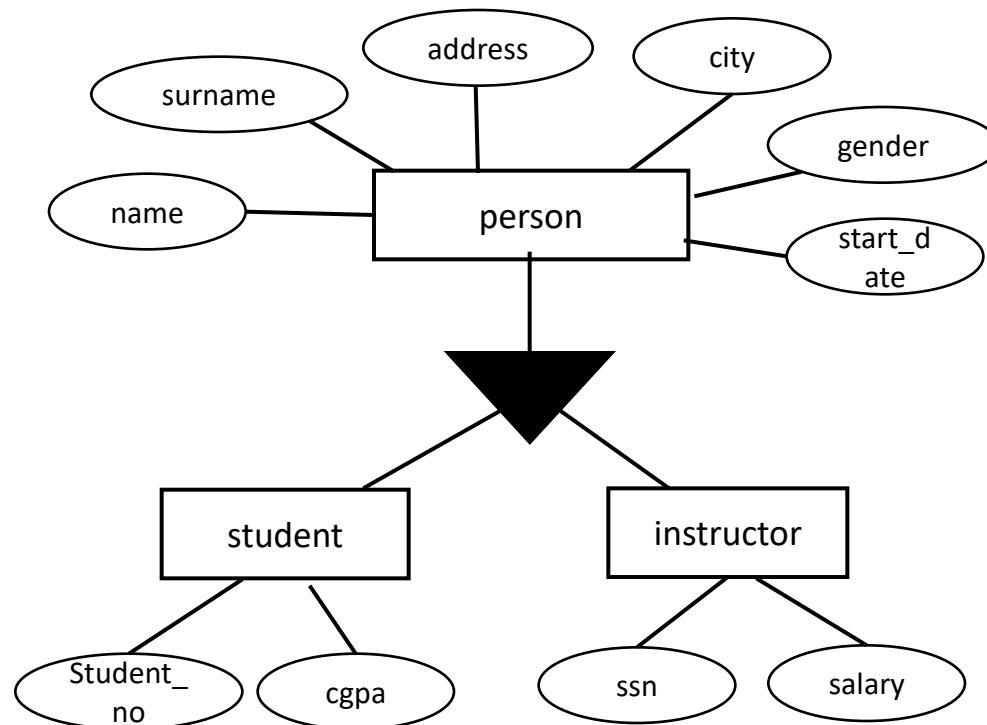
Specialization is a concept where attributes that are common to several types of an entity are grouped into their own entity.

- Student: student\_id, name, surname, address, city, gender, start date, department, cgpa
- Instructor: social\_security\_no, name, surname, address, city, gender, start date, salary



# Entity Relationship Diagram

## Specialization ( Generalization ) cont.

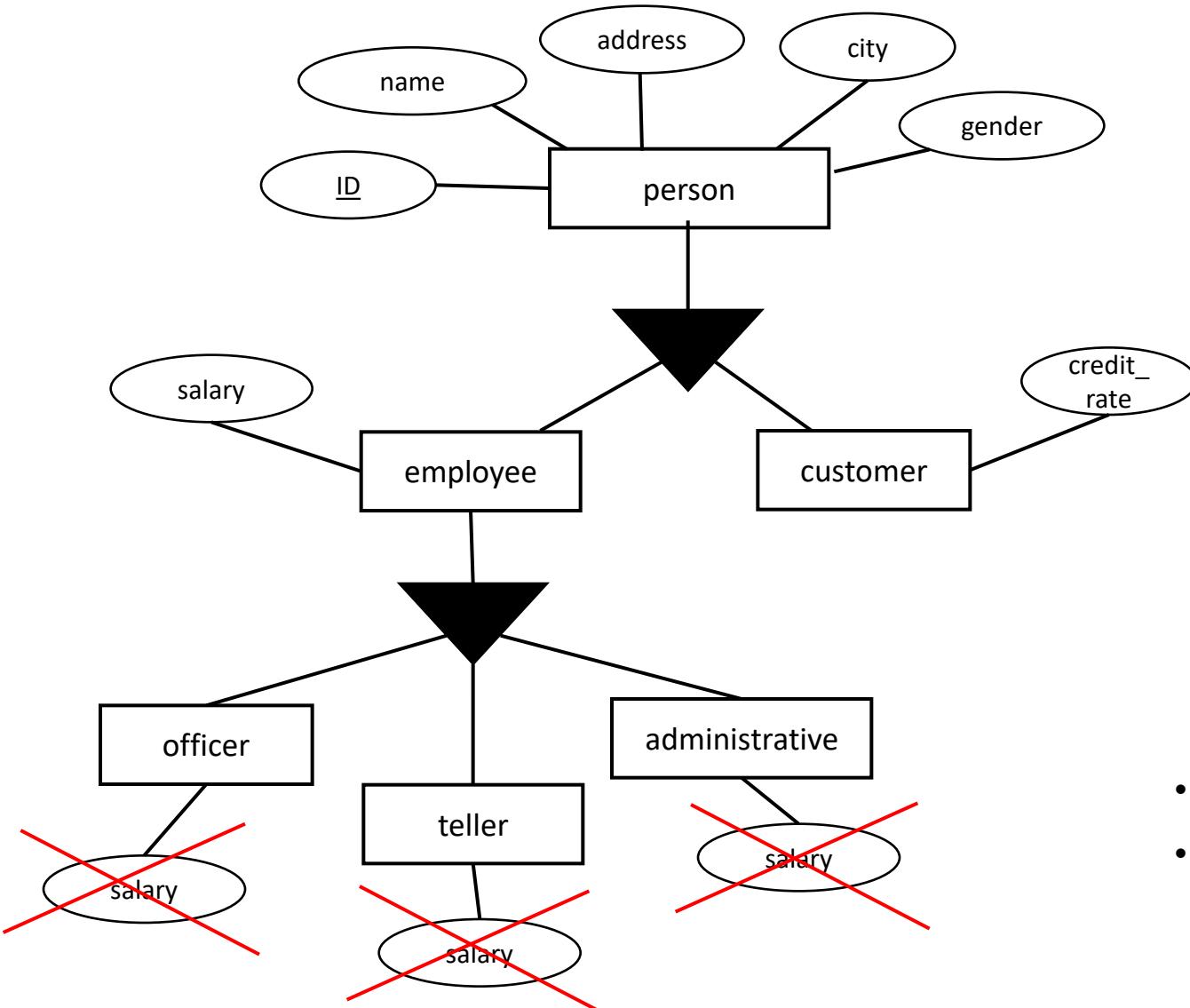


# Entity Relationship Diagram

## **Specialization ( Generalization ) cont.**

- The subgroupings become lower-level entity (child) sets that have attributes or participate in relationships that do not apply to the higher-level entity (parent)
- 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 Diagram



- Officer is-a employee.
- Employee is-a person.

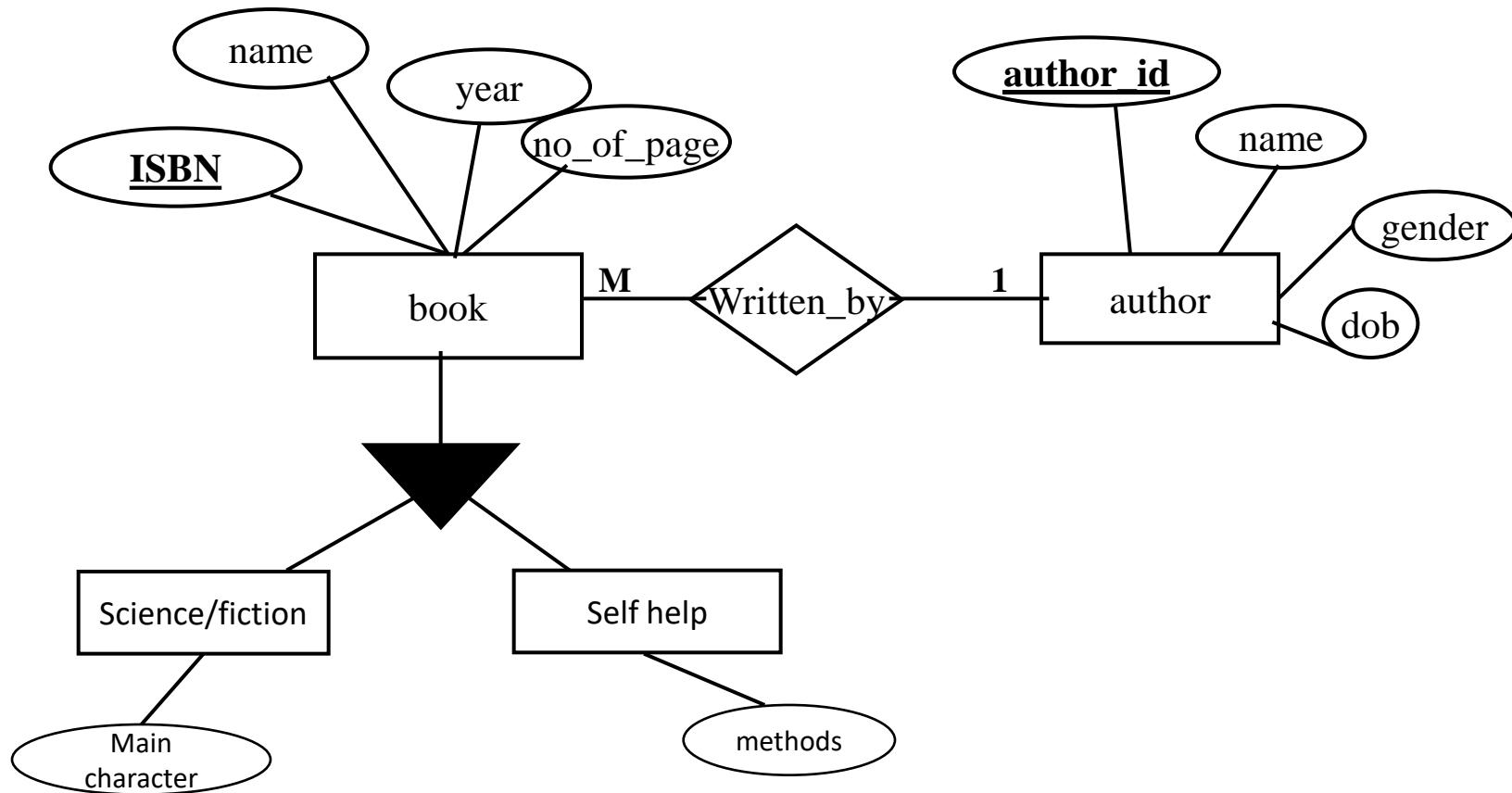
# Entity Relationship Diagram

**Ex:** Draw an ER diagram for the following situation:

There are many books (novels) in the book store. For each book, ISBN, name, year\_published and no\_of\_pages details must be stored. Each book is written by one author. For each author his/her id, name, gender and dob (date of birth) must be recorded in the database.

There are only two types of books available (science/fiction or self help). For science/fiction books, main characters must be stored and for self help, methods must be stored.

# Entity Relationship Diagram



# Reduction to Relation Schemas

# Reduction to Relation Schemas

- Entity sets and relationship sets can be expressed uniformly as *relation schemas* that represent the contents of the database.
- A database which conforms to an E-R diagram can be represented by a collection of schemas.
- For each entity set and relationship set there is a unique schema that is assigned the name of the corresponding entity set or relationship set.
- Each schema has a number of columns (generally corresponding to attributes), which have unique names.

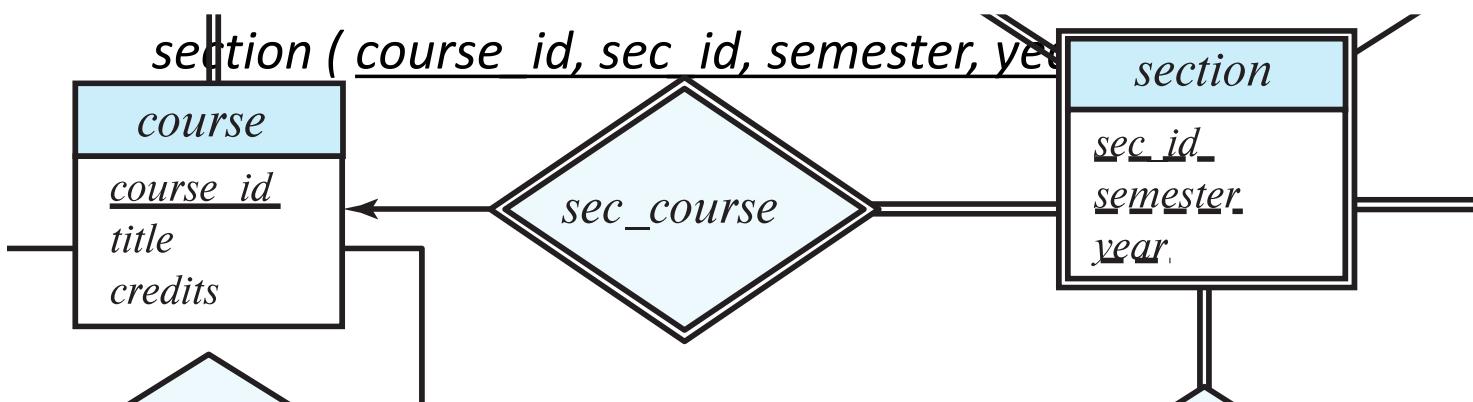
# Representing Entity Sets

- A strong entity set reduces to a schema with the same attributes plus foreign key (if exists)

*student(ID, name, tot\_cred) without foreign key*

*student(ID, name, tot\_cred, department) with foreign key*

- A weak entity set becomes a table that includes a column for the primary key of the identifying strong entity set



# Representation of Entity Sets with Composite Attributes

<i>instructor</i>
<i>ID</i>
<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>

- Composite attributes are flattened out by creating a separate attribute for each component attribute
  - Example: given entity set *instructor* with composite attribute *name* with component attributes *first\_name* and *last\_name* the schema corresponding to the entity set **has two attributes *name\_first\_name* and *name\_last\_name***
  - Prefix omitted if there is no ambiguity (*name\_first\_name* could be *first\_name*)
- Ignoring multivalued attributes, extended instructor schema is
  - *instructor(ID, first\_name, middle\_initial, last\_name, street\_number, street\_name, apt\_number, city, state, zip\_code, date\_of\_birth)*

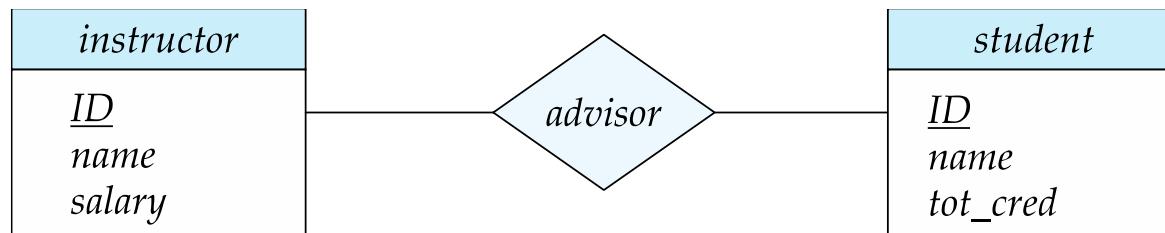
# Representation of Entity Sets with Multivalued Attributes

- A multivalued attribute  $M$  of an entity  $E$  is represented by a separate schema  $EM$
- Schema  $EM$  has attributes corresponding to the primary key of  $E$  and an attribute corresponding to multivalued attribute  $M$
- Example: Multivalued attribute  $phone\_number$  of  $instructor$  is represented by a schema:  
 $inst\_phone = (\underline{ID}, \underline{phone\_number})$
- Each value of the multivalued attribute maps to a separate tuple of the relation on schema  $EM$ 
  - For example, an  $instructor$  entity with primary key 22222 and phone numbers 456-7890 and 123-4567 **maps to two tuples:** (22222, 456-7890) and (22222, 123-4567)

# Representing Relationship Sets

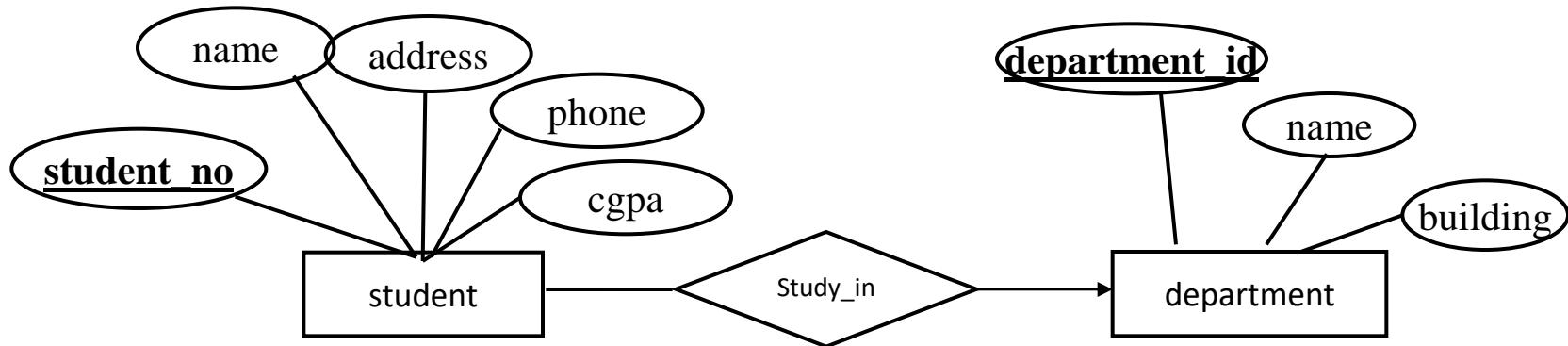
- A many-to-many relationship set is represented as a schema with attributes for the primary keys of the two participating entity sets, and any descriptive attributes of the relationship set.
- Example: schema for relationship set *advisor*

*advisor* = (s\_id, i\_id)



# Representation of ERD in Relational Data Model – More Exercises

Exercise: Show the relational data model of the given ERD



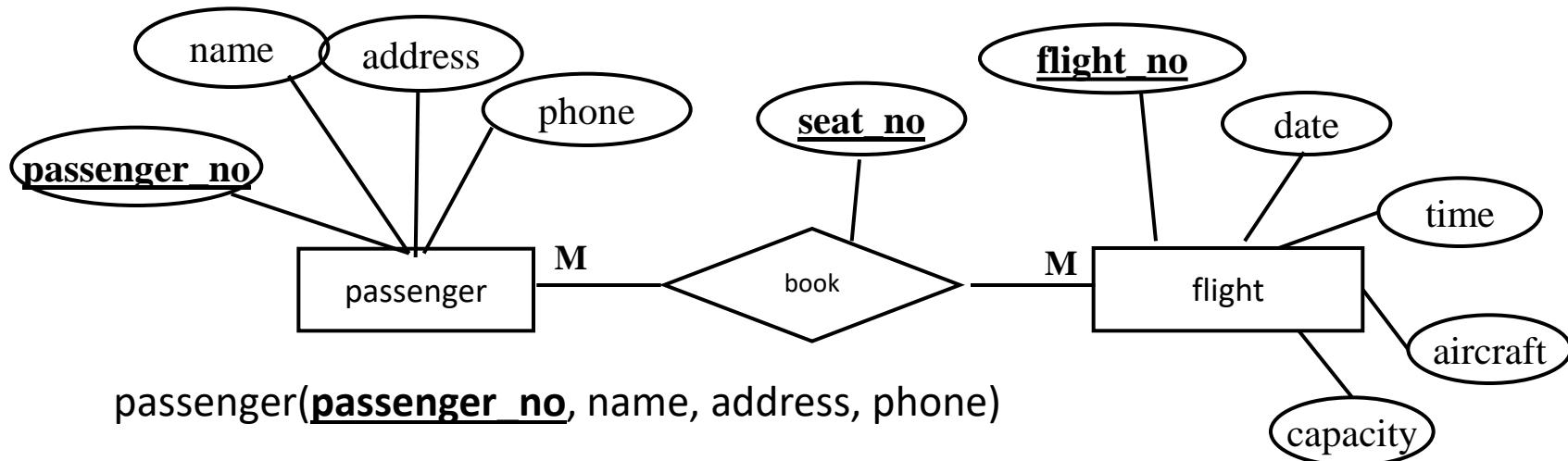
student(**student no**, name, address, phone, cgpa, **department\_id**)  
f.k

department(**department id**, name, building)

Study\_in(**student no**, **department id**)

# Entity Relationship Diagram

**Exercise: Show the relational data model of the given ERD**



passenger(**passenger no**, name, address, phone)

flight(**flight no**, date, time, aircraft, capacity)

book(**passenger no**, **flight no**, **seat no**)

# Entity Relationship Diagram

# Representation of Relational Data Model in ERD

**Exercise:** Draw the ERD of the given relational data model below:

department (dept\_no , name, budget)

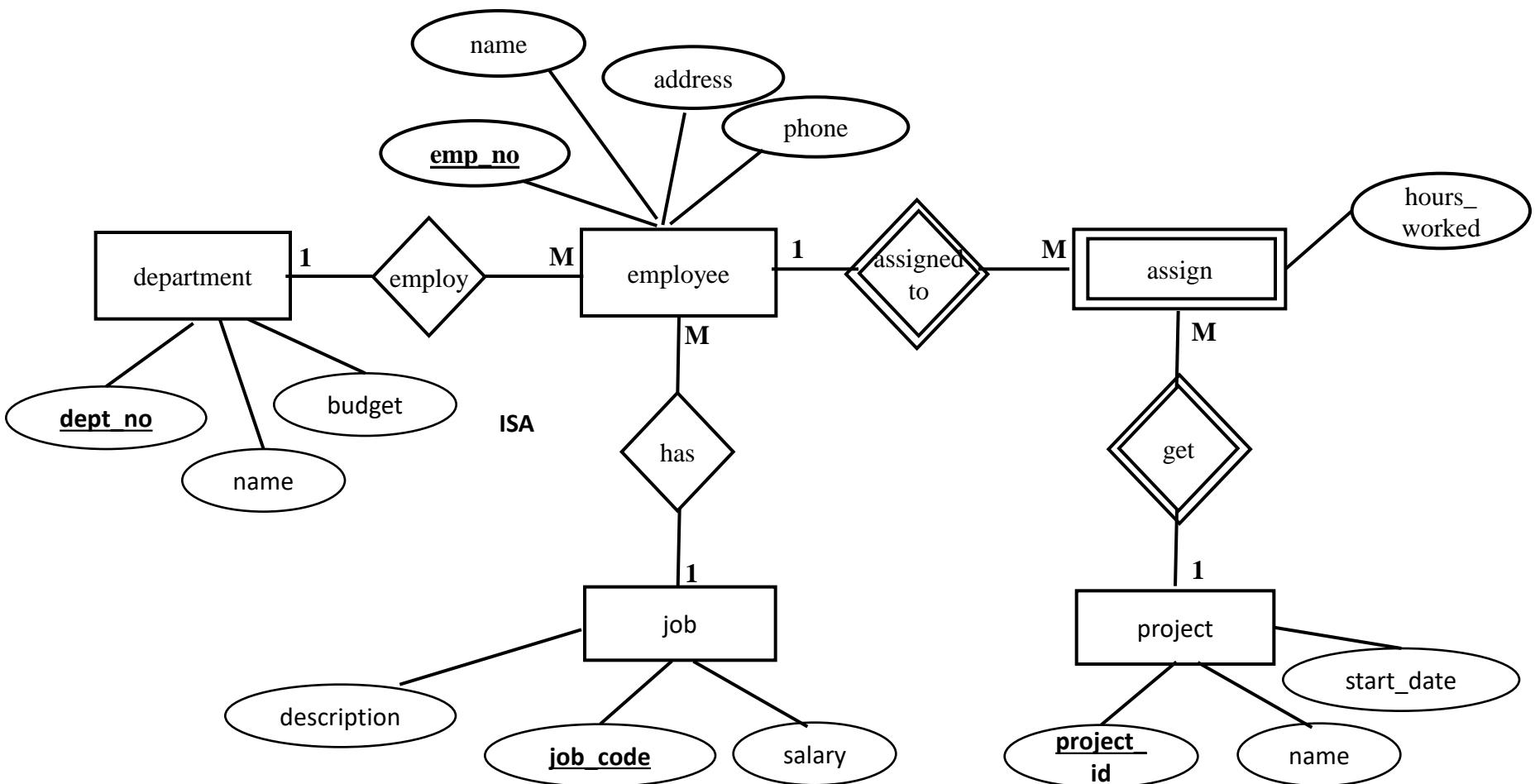
**employee (emp\_no, name, address, dept\_no, job\_code)**

jobs (job\_code, description, salary)

assign (emp\_no, project\_id, hours\_worked)  
f.k. f.k.

**projects (project\_id, name,start\_date)**

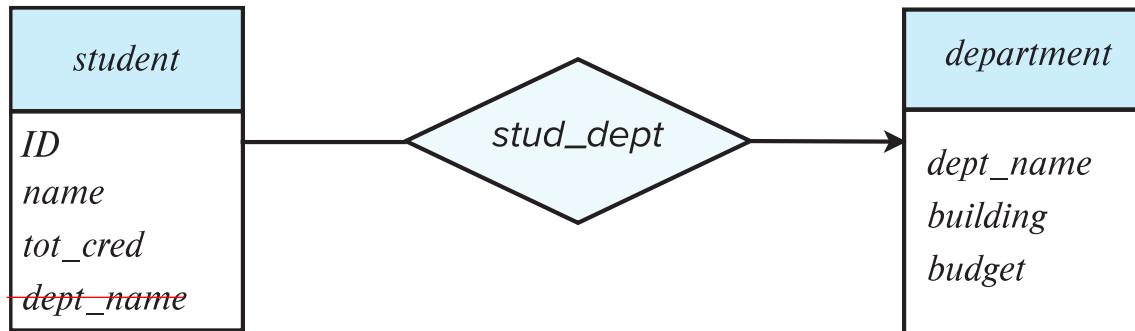
# Entity Relationship Diagram



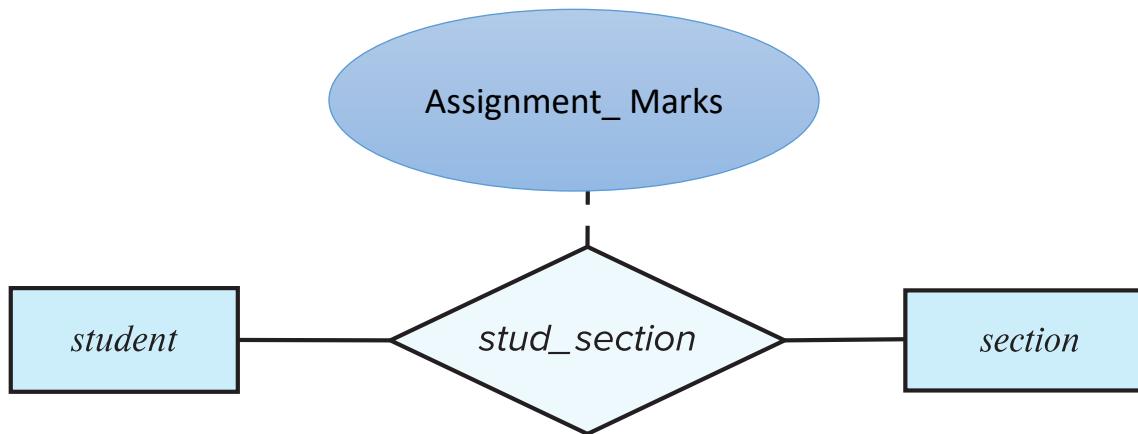
# Design Issues

# Common Mistakes in E-R Diagrams

- Example of erroneous E-R diagrams

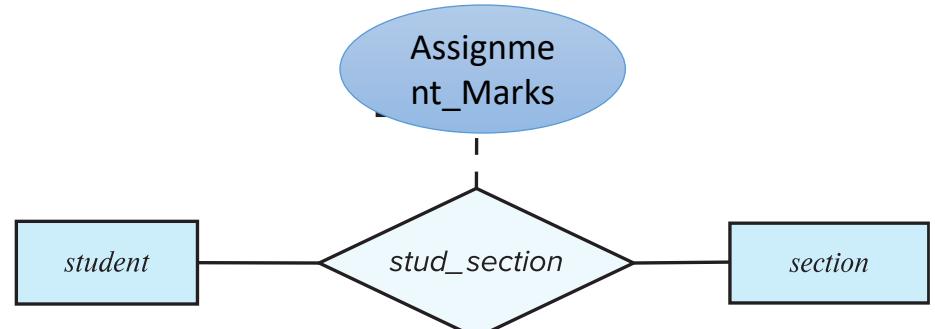


(a) Incorrect use of attribute

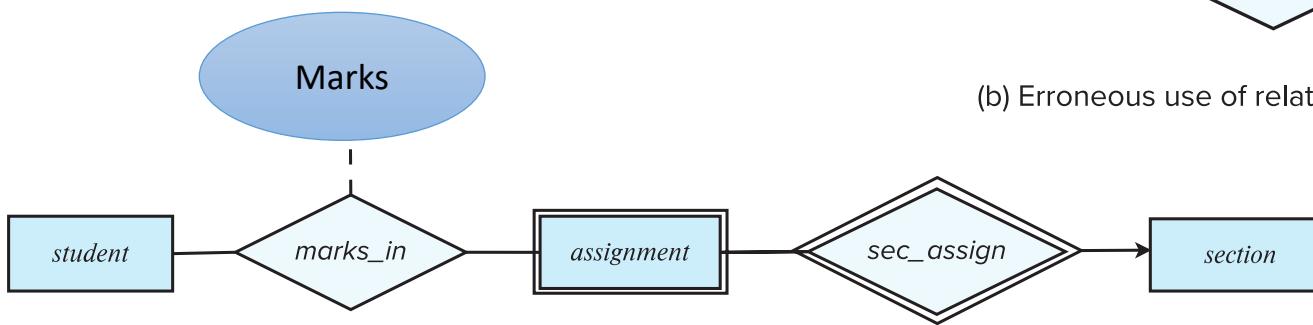


(b) Erroneous use of relationship attributes

# Common Mistakes in E-R Diagrams (Cont.)



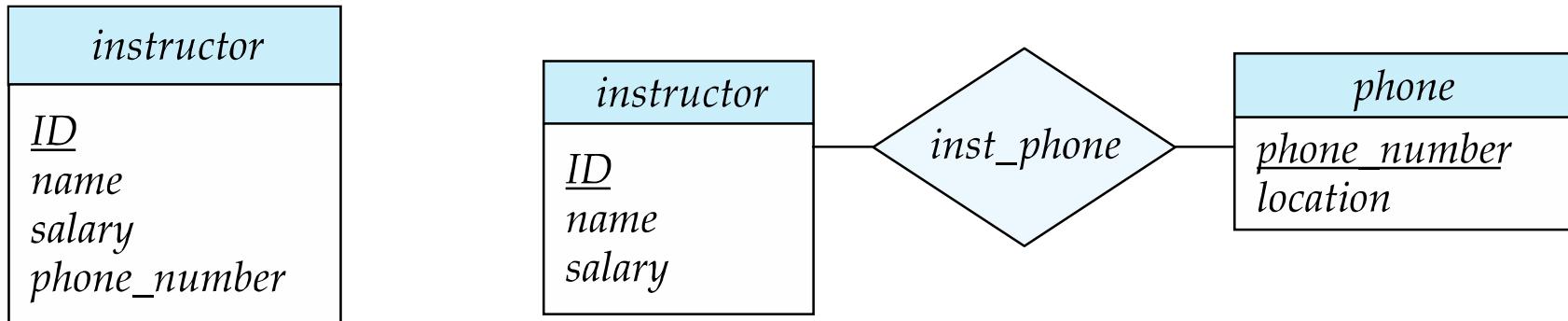
(b) Erroneous use of relationship attributes



(c) Correct alternative to erroneous E-R diagram (b)

# Entities vs. Attributes

- Use of entity sets vs. attributes

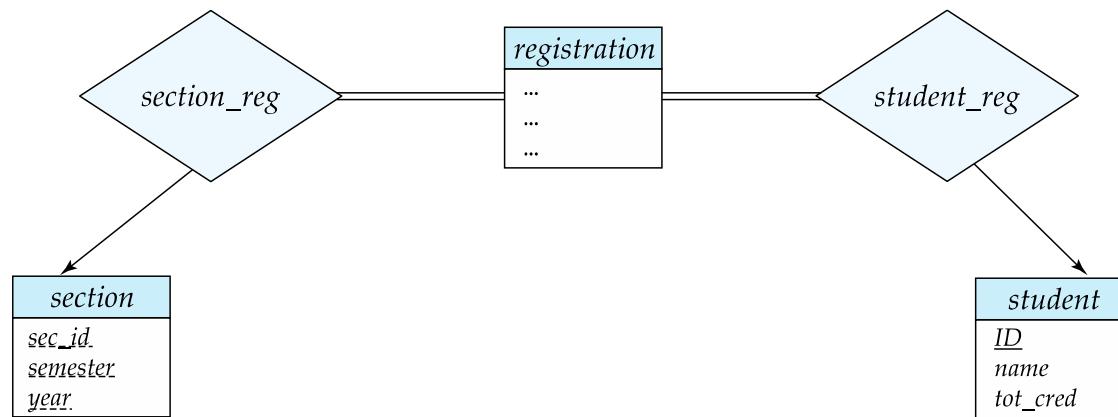


- Use of phone as an entity allows extra information about phone numbers (plus multiple phone numbers)

# Entities vs. Relationship sets

- **Use of entity sets vs. relationship sets**

Possible guideline is to designate a relationship set to describe an action that occurs between entities



- **Placement of relationship attributes**

For example, attribute date as attribute of advisor or as attribute of student