Database Management System

Unit 3: Data Modelling

Database Modelling

3.1 Entity-Relationship Model:

- Entity Sets
- Relationship Sets
- Attributes

3.2 Constraints:

- Mapping Cardinalities
- Participation Constraints
- Keys
- Redundant Attributes in Entity Sets

3.3 Entity Relationship Diagrams:

- Mapping Cardinalities
- Complex Attributes
- Strong and Weak Entity Sets
- Reduction to Relational Schemas

3.4 Extended E-R Features:

- Specialization
- Generalization
- Attribute Inheritance
- Aggregation

CE: 3.1

Entity-Relationship Model

3.1. Entity-Relationship Model

- The E-R model is very useful in mapping the meanings and interactions of real-world enterprises onto a conceptual schema.

- The E-R data model employs three basic concepts:
    - Entity sets,
    - Relationship sets, and
    - Attributes.

Entity Sets

- ***An entity*** is a "thing" or "object" in the real world that is different from all other objects.

- For example:
  - each person in a university is an entity,
  - specific person,
  - company,
  - event,
  - plant, etc.

- Entities have ***attributes***..
  - For Example: people have names and addresses.

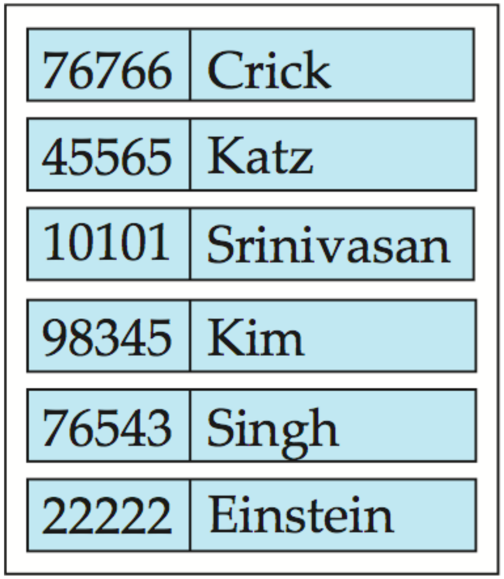
Entity Sets (Conti…)

- ***An entity set*** is a set of entities of the same type that share the same properties or attributes.

- For Example:
  - set of all persons,
  - companies,
  - trees,
  - holidays, etc.

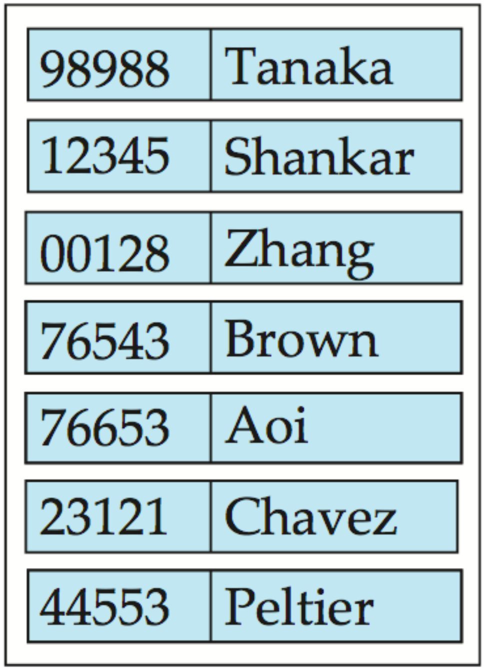
Demonstration

Entity Sets of *instructor and student*



instructor_ID	instructor_name
76766	Crick
45565	Katz
10101	Srinivasan
98345	Kim
76543	Singh
22222	Einstein

instructor



student-ID	student_name
98988	Tanaka
12345	Shankar
00128	Zhang
76543	Brown
76653	Aoi
23121	Chavez
44553	Peltier

student

Relationship Sets

- **A relationship** is an association among several entities.
- For Example:

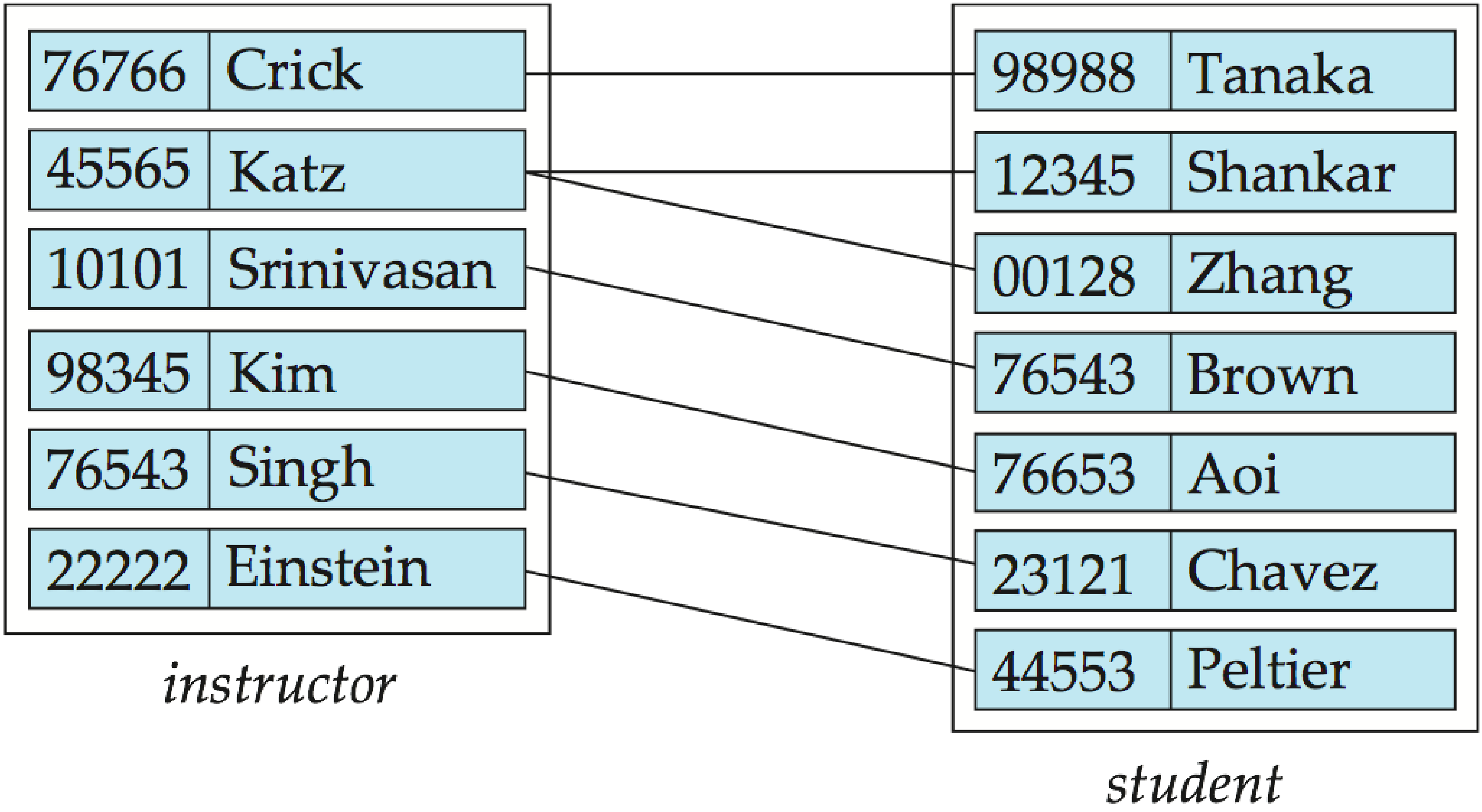
44553 (Raj)	advisor	22222 (Einstein)
student entity	**relationship set**	**instructor entity**

- A relationship set is a mathematical relation among n > 2 entities, each taken from entity sets.

- For Example:

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

Relationship Sets of *advisor* (Conti…)



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

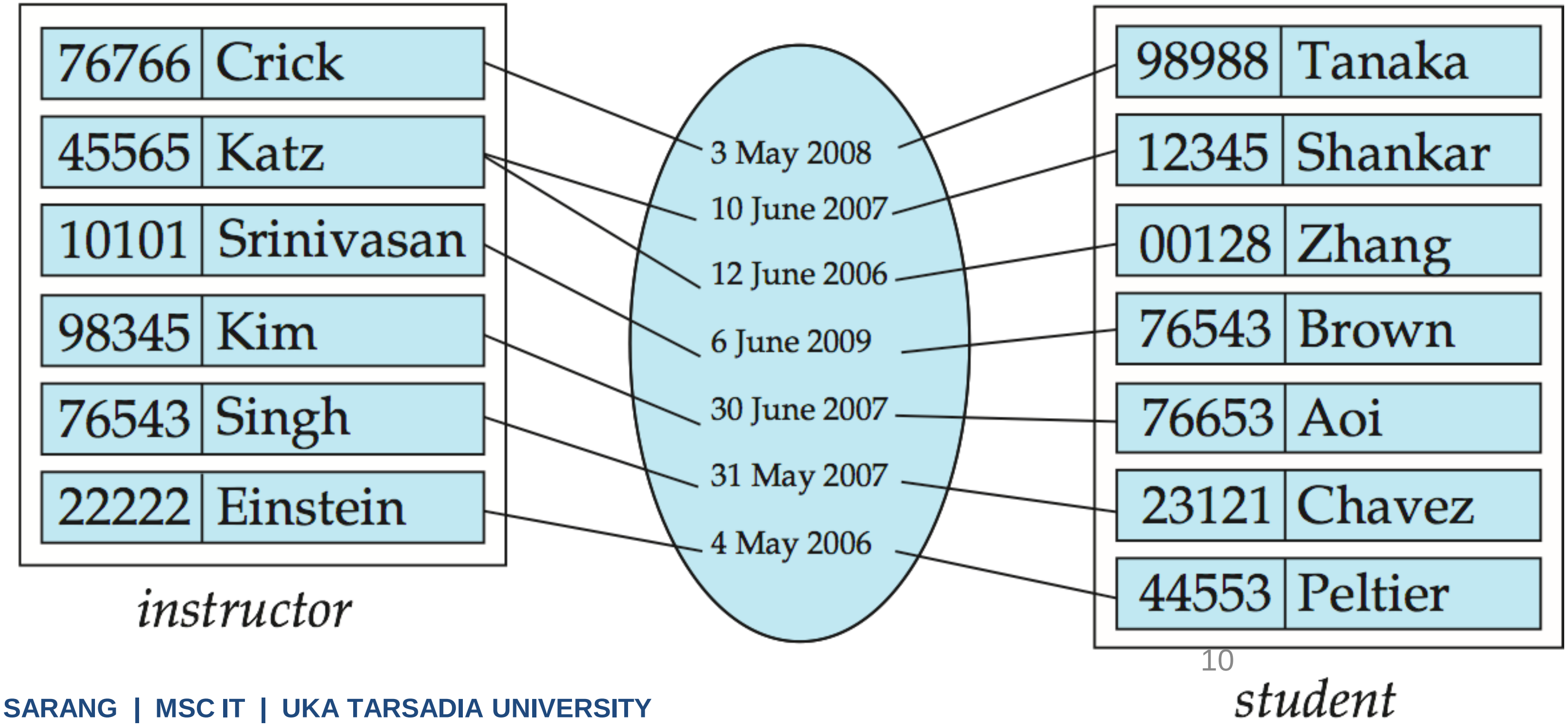
instructor

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

student

Relationship Sets (Conti…)

- ***An attribute*** can also be property of 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.*



Attributes

- An entity is represented by a set of ***attributes***, that is descriptive properties possessed by all members of an entity set.
- For Example:

  instructor = (ID, name, street, city, salary)

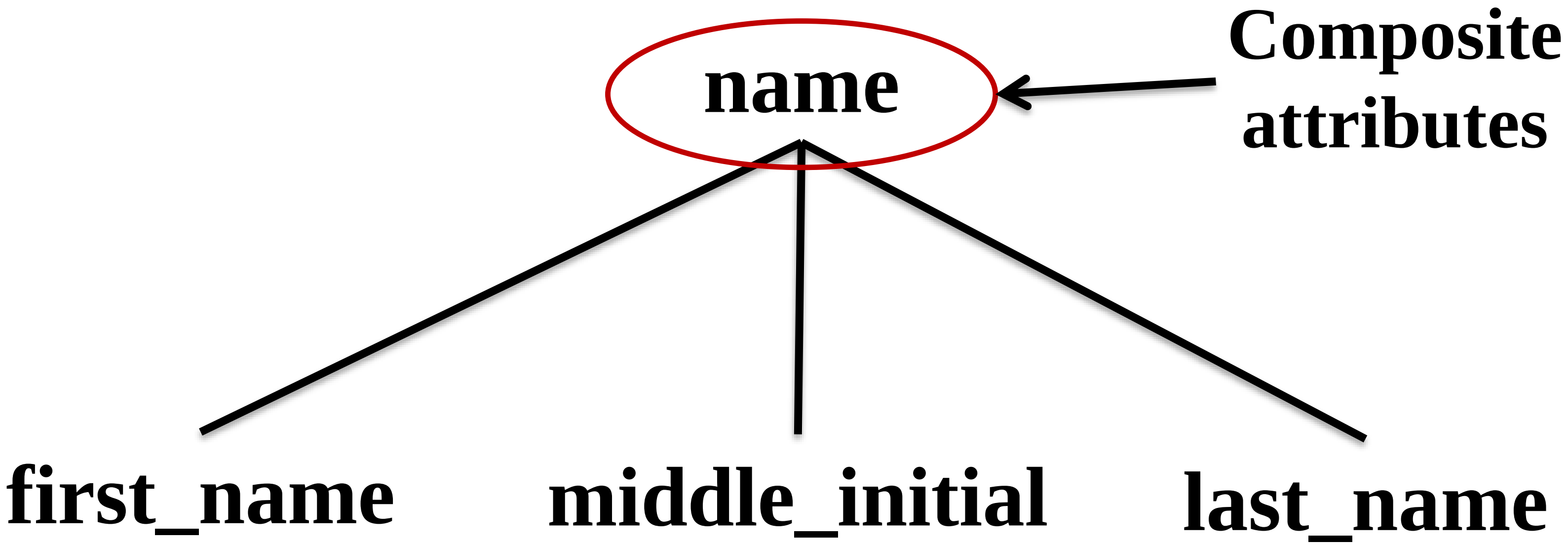
  course = (course_id, title, credits)

- **Domain** is the set of the permitted values for each attribute.
- Attributes are of different types. They are as following:
  1. Simple and composite attributes.
  2. Single-valued and multivalued attributes.
  3. Derived attributes.

1) Simple and Composite attributes

- A ***Simple attributes*** is an attribute that is not divided into subparts.

- On the other hand, a ***Composite attributes*** can be divided into subparts.



1) Simple and Composite attributes (Conti…)

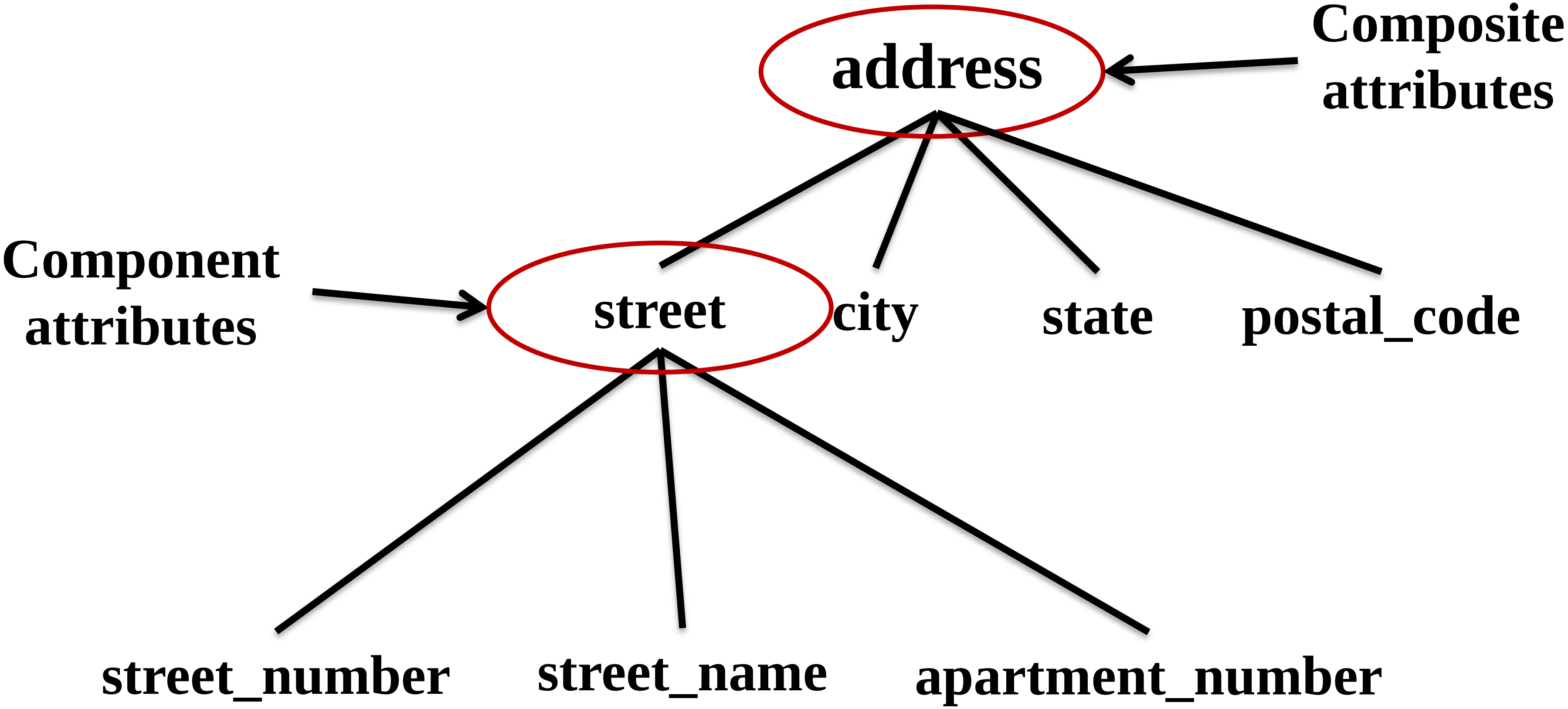
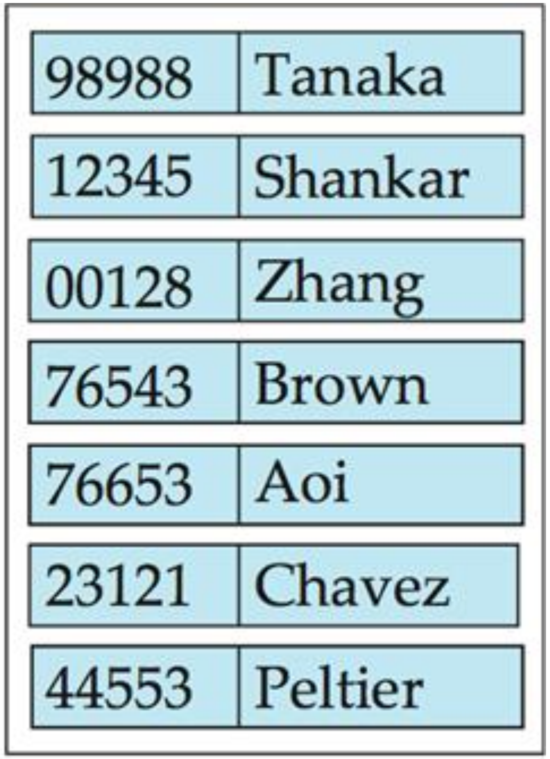


Fig.: Composite attributes instructor *name* and *address*.

2) Single-valued and Multivalued attributes



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

student

- For instance, the *student ID* attribute for a specific student entity refers to only one *student ID*. Such attributes are said to be ***single valued***.

2) Single-valued and Multivalued attributes (Conti…)

- Suppose we add a *phone number* attribute, to the *instructor* entity set.

- An *instructor* may have zero, one or several phone numbers and different instructors may have different numbers of phones. This type of attribute is said to be ***multivalued***.

- To denote a multivalued attribute, we enclose it in braces.
- For Example:

{phone number} or {dependent name}.

3) Derived attributes

- Derived attributes can be added from other attributes.

- For Example:
  - age,
  - given date_of_birth

- Suppose that the *instructor* entity set has an attribute *age*, that indicates the instructor’s age.

- If the instructor entity set also has an attribute date of birth, we can calculate age from date of birth and the current date.
- Thus, age is a derived attribute.

3) Derived attributes (Conti…)

- In this case, *date of birth* may be referred to as a **base attribute,** or **a stored attribute**.
- The value of a derived attribute is not stored but is computed when required.

International Certification Question

Q1.	A set of a “thing” or “object” in the real world that is different from all other objects is know as?
A.	Attribute
B.	Entity
C.	Relationship Sets
D.	Entity Sets

Ans: B

International Certification Question

Q2.	**Which attributes can be divided into subparts?**
A.	**Simple Attributes**
B.	**Composite Attributes**
C.	**Single-valued Attributes**
D.	**Multi-valued Attributes**

Ans: B

International Certification Question

Q3.	Which attributes can be enclose in braces?
A.	Simple Attributes
B.	Composite Attributes
C.	Single-valued Attributes
D.	Multi-valued Attributes

Ans: D

International Certification Question

Q4.	An attribute can be….
A.	a property of a entity set.
B.	a property of a relationship set.
C.	a property of a attribute set.
D.	a property of a mutli-valued relationship set.

Ans: B

International Certification Question

Q5.	The set of the permitted values for each attribute is known as?
A.	Domain
B.	Attribute
C.	Entity
D.	Relationship Sets

Ans: A

Industry Interview Questions

1. What is an entity?
2. What is an entity sets?
3. What is a relationship set?
4. What an attribute?
5. What are the different types of attributes?
6. What is the use of derived attributes?

Home Work

1. Define the following terms: [1 Mark each]
   a) Entity
   b) Entity set
   c) Relationship set
   d) Domain
   e) Attribute
2. What is the difference between Simple and Composite attributes? [2 Marks]
3. What is the difference between Single-valued and Multi-valued attributes? [2 Marks]
4. What is derived attribute? Give example. [2 Marks]
5. List and explain different types of attributes. [5 Marks]

CE: 3.2

Constraints

3.2. Constraints

- Mapping Cardinalities
- Participation Constraints

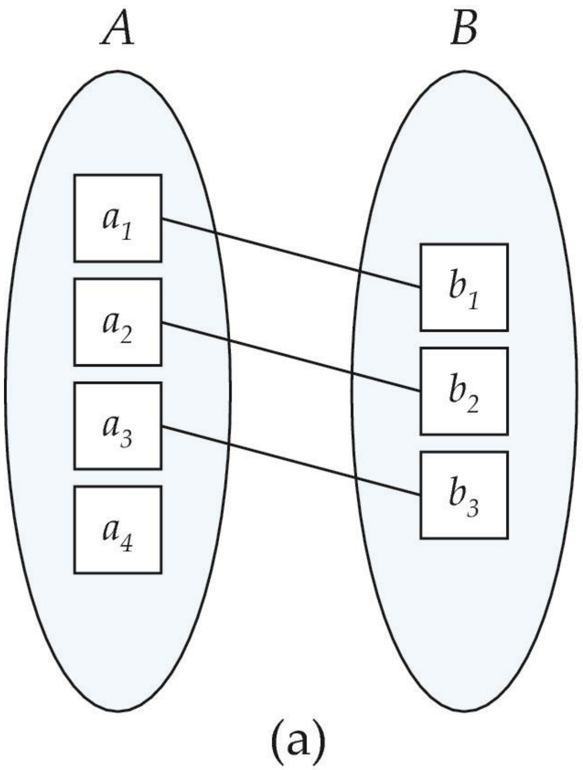
3.2.1 Mapping Cardinalities

- **Mapping cardinalities or cardinality ratios** express the number of entities to which another entity can be associated via a relationship set.

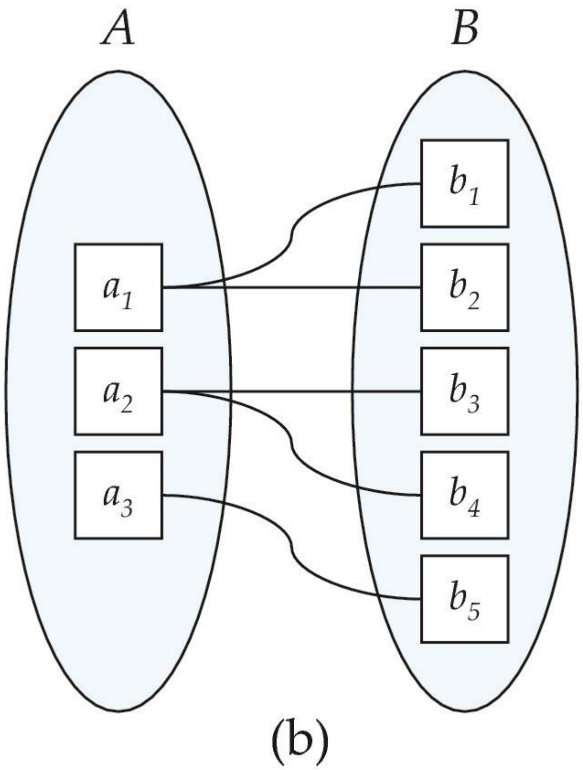
- It is most useful for describing binary relationship sets.

- For a binary relationship set R between entity sets A and B, the mapping cardinality must be one of the following:
  1) One-to-one
  2) One-to-many
  3) Many-to-one
  4) Many-to-many

3.2.1 Mapping Cardinalities (Conti…)



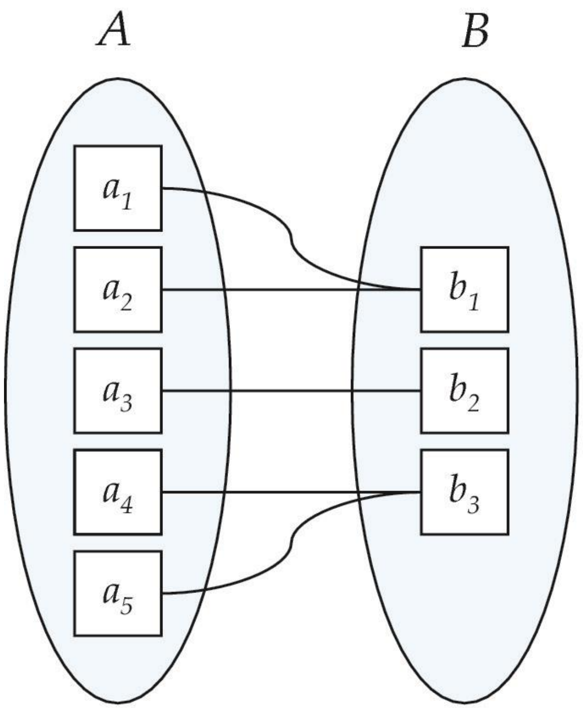
One to one



One to many

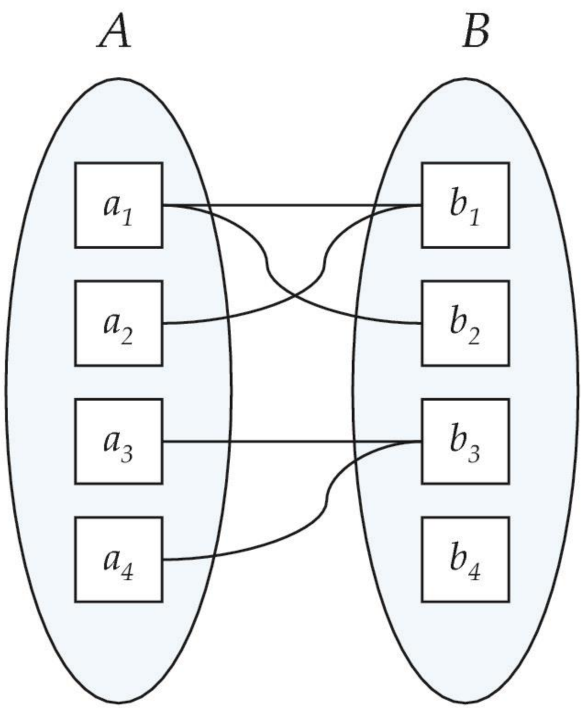
- Note: Some elements in A and B may not be mapped to any elements in the other set.

3.2.1 Mapping Cardinalities (Conti…)



(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.

3.2.2 Participation Constraints

- Participation Constraints determines whether the reality of an entity depends to another entity through the relationship.

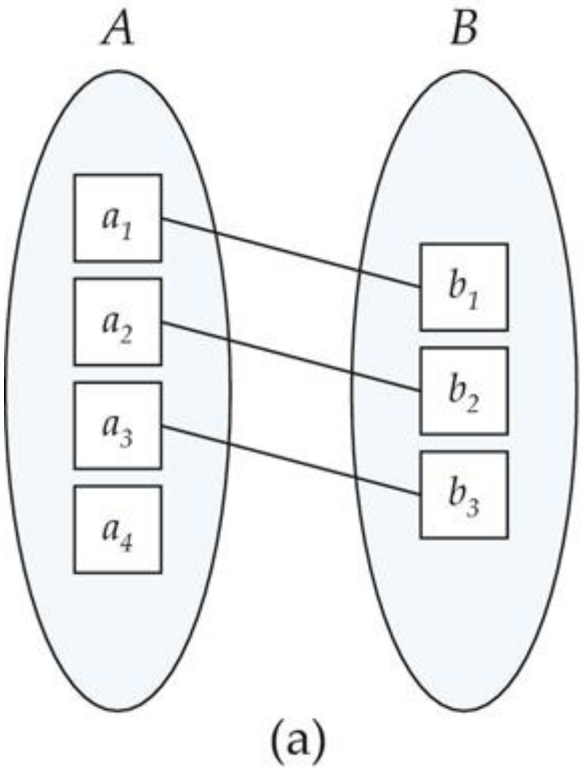
1. Total Participation
2. Partial Participation

- If every entity in E participates is in at least one relationship of R, then participation of an entity set E is in a relationship set R is said to be **total**.

- If only some entities in E participate is in relationships of R, then participation of entity set E is in a relationship R is said to be **partial**.

3.2.2 Participation Constraints (Conti…)

- For Example:

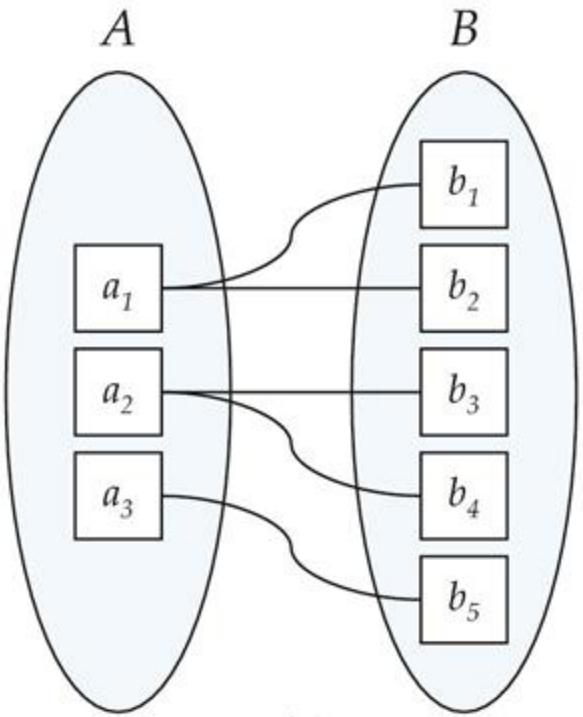


One to one

- The participation of *B* in the relationship set is **total** while the participation of A in the relationship set is **partial**.

3.2.2 Participation Constraints (Conti…)

- For Example:



(b)

One to many

- The participation of both A and B in the relationship set are **total**.

3.2.3 Keys

1) **Candidate key**
2) **Primary key**
3) **Alternate key**
4) **Super key**
5) **Foreign key**

3.2.3 Keys (Conti…)

1. **Candidate key:**

- The minimal set of attribute which can uniquely identify a tuple (record) is known as **candidate key**.

- For Example:
  - ✓ Stud_Enro is candidate key of STUDENT relation.
  - ✓ course_id is candidate key of COURSE relation.

- ➢ The value of Candidate Key is unique and non-null for every tuple.

- ➢ There can be more than one candidate key in a relation.

3.2.3 Keys (Conti…)

1. **Candidate key: (Conti…)**

- For Example:
  - ✓ Stud_Enro as well as Stud_Phone both are candidate keys for STUDENT relation.

- The candidate key can be simple (having only one attribute) or composite as well.

- For Example:

  {stud_enro, course_id} is a composite candidate key for relation STUDENT_COURSE.

3.2.3 Keys (Conti…)

2. Primary key:

- There can be more than one candidate key in a relation out of which one can be chosen as primary key.

- For Example:
  - ✓ Stud_Enro as well as Stud_Phone both are candidate keys for STUDENT relation.

  - ✓ But Stud_Enro can be chosen as primary key(only one out of many candidate keys).

3.2.3 Keys (Conti…)

3. Alternate key:

- The candidate key other than primary key is called as alternate key.

- For Example:
  - ✓ Stud_Enro as well as Stud_Phone both are candidate keys for STUDENT relation.
  - ✓ But Stud_Phone will be alternate key (only one out of many candidate keys).

3.2.3 Keys (Conti…)

4. Super key:

- The set of attributes which can uniquely identify a tuple is known as Super Key.
  (OR A combination of one or more attributes with the help of which we can uniquely identify a tuple is called super key.)

- For Example:
  - ✓ Stud_Enro, (Stud_Enro, Stud_Name), etc

- Adding zero or more attributes to candidate key generates super key.

- *A candidate key is a super key but vice versa is not true.*

3.2.3 Keys (Conti…)

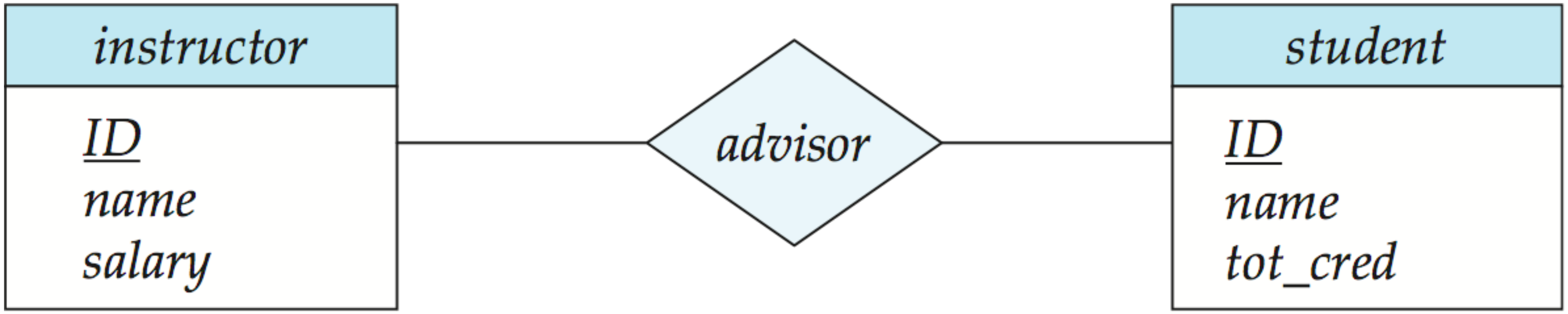
5. Foreign key:

- If an attribute that can only take the values which are present as values of some other attribute, it will be foreign key to the attribute to which it refers.

- For Example:
  - ✓ Stud_Enro in Student_course is a foreign key to Stud_Enro in STUDENT relation.

Keys for Relationship Sets

- For Example:



- Therefore, here (s_id, i_id) is the super key of advisor
  - This means that a pair of entity sets can have at most one relationship in a particular relationship set.

- For Example: if we wish to track multiple meeting dates between a *student* and his *advisor*, we cannot assume a relationship for each meeting. We can use a multivalued attribute though.

Keys for Relationship Sets (Conti…)

- When we decide what are the candidate keys, we must consider the mapping cardinality of the relationship set.

- We need to consider the relationship set for selecting the primary key, if the candidate key is more than one.

3.2.4 Redundant Attributes in Entity Sets

- Consider the entity sets *instructor* and *department*:
  - The entity set instructor includes the attributes…
    - ✓ ID, name, dept_name and salary with ID forming the primary key.
  - The entity set department includes the attributes…
    - ✓ dept_name, building and budget with dept_name forming the primary key.
- The attribute dept_name appears in both entity sets.
- But, since it is the primary key for the entity set department, it is redundant in the entity set *instructor* and needs to be removed.

International Certification Question

Q1.	Which key of an entity set is a set of one or more attributes whose values uniquely determine each entity?
A.	Primary Key
B.	Candidate Key
C.	Super Key
D.	Foreign Key

Ans: C

International Certification Question

Q2.	**Which key of an entity set is a minimal super key?**
A.	**Primary Key**
B.	**Candidate Key**
C.	**Super Key**
D.	**Foreign Key**

Ans: B

International Certification Question

Q3.	If the candidate key is more than one, Which relationship set consider for selection?
A.	Primary Key
B.	Candidate Key
C.	Super Key
D.	Foreign Key

Ans: A

International Certification Question

Q4.	If every entity in E participates is in at least one relationship of R, then participation of an entity set E is in a relationship set R is said to be?
A.	Total Participation
B.	Partial Participation

Ans: A

International Certification Question

Q5.	**With which attribute though, we cannot assume a relationship for each meeting?**
A.	**Simple**
B.	**Composite**
C.	**Single-valued**
D.	**Multi-valued**

Ans: D

Industry Interview Questions

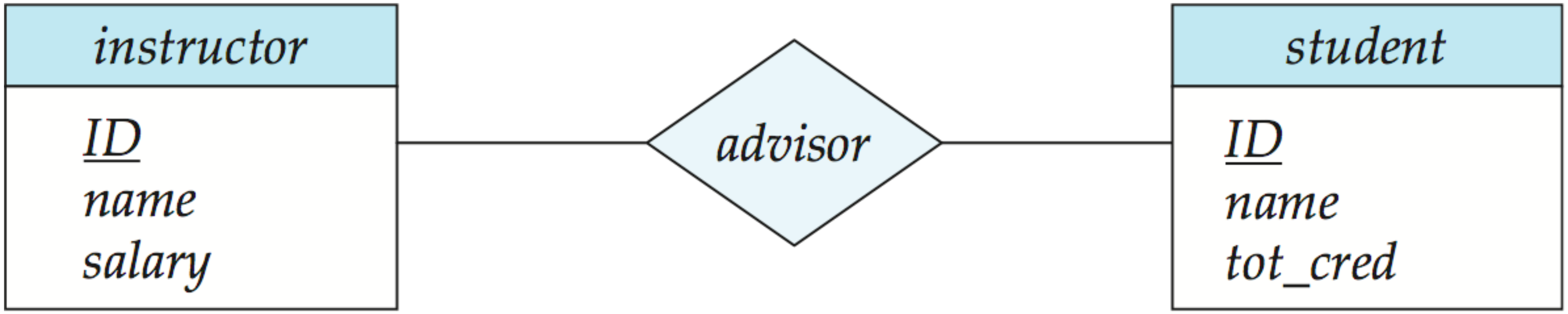
1. What is a primary key?
2. What is a candidate key?
3. When we used super key?
4. What is Mapping Cardinalities?
5. What is One-to-One mapping?
6. What is One-to-Many mapping?
7. What is Many-to-one mapping?
8. When we can use Many-to-Many relationship?
9. When we can say that the relationship set is in total participation?

Home Work

1. What is a primary key? [1 Mark]
2. What is a candidate key? [1 Mark]
3. When we used super key? [1 Mark]
4. When we can say that the relationship set is in total participation? [1 Mark]
5. Explain Participation Constraints. [2 Marks]
6. Explain Mapping Cardinalities with examples. [5 Marks]

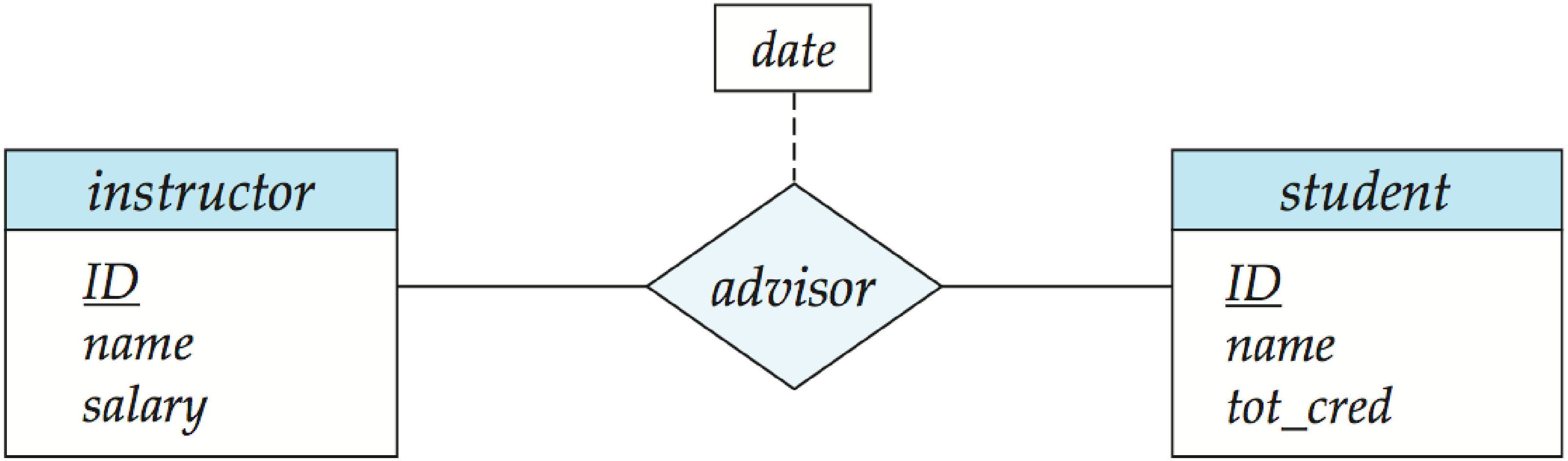
CE: 3.3
Entity Relationship Diagrams

3.3. Entity Relationship Diagrams

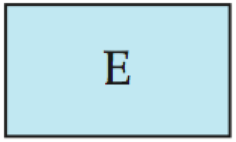


- **Rectangles** represent entity sets.
- **Diamonds** represent relationship sets.
- **Attributes** listed inside entity rectangle.
- **Underline** indicates primary key attributes.

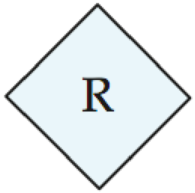
Relationship Sets with Attributes



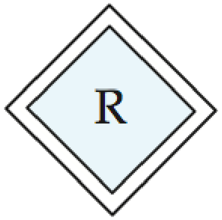
E-R Diagram Software Use Different Notations



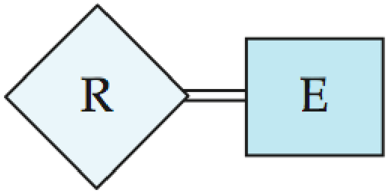
entity set



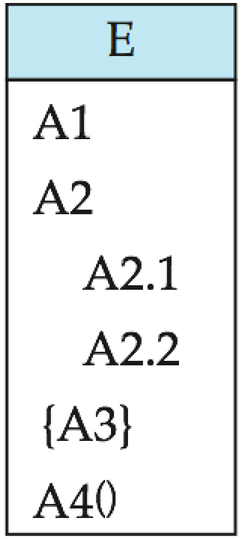
relationship set



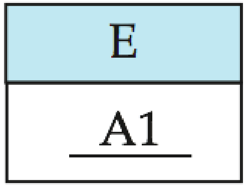
identifying
relationship set
for weak entity set



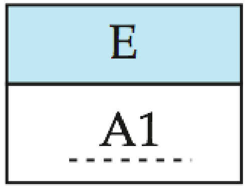
total participation
of entity set in
relationship



attributes:
simple (A1),
composite (A2) and
multivalued (A3)
derived (A4)

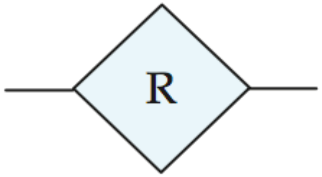


primary key

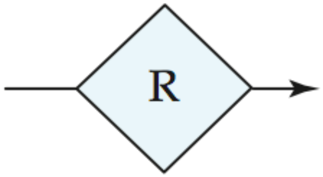


discriminating
attribute of
weak entity set

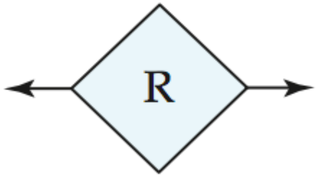
E-R Diagram Software Use Different Notations(Conti...)



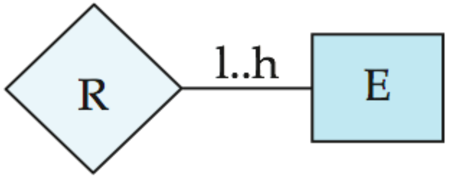
many-to-many relationship



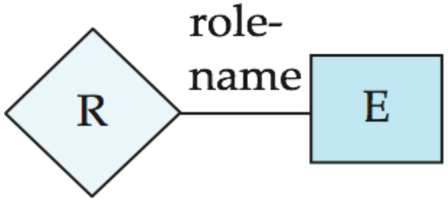
many-to-one relationship



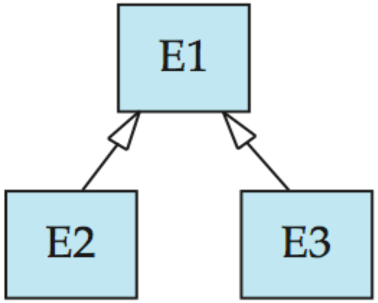
one-to-one relationship



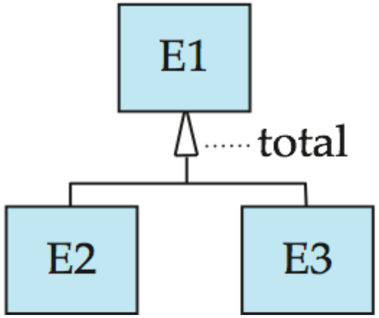
cardinality limits



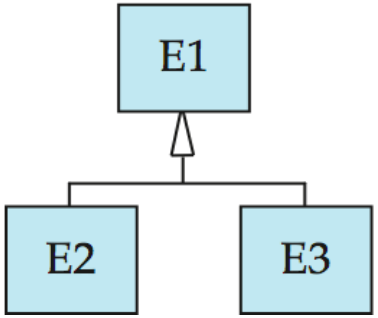
role indicator



ISA: generalization or specialization



total (disjoint) generalization



disjoint generalization

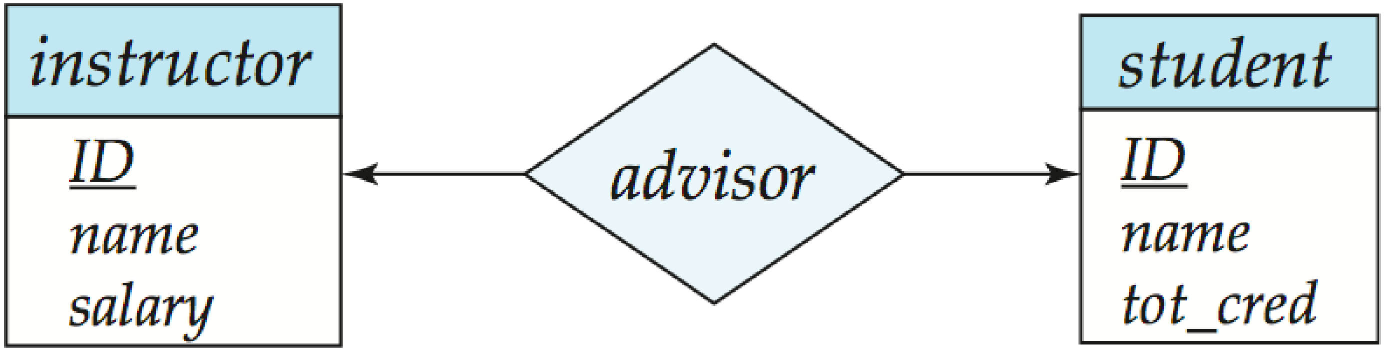
3.3.1 Mapping Cardinalities

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

- For Example:
  - One-to-one relationship.
    - ✓ 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.*

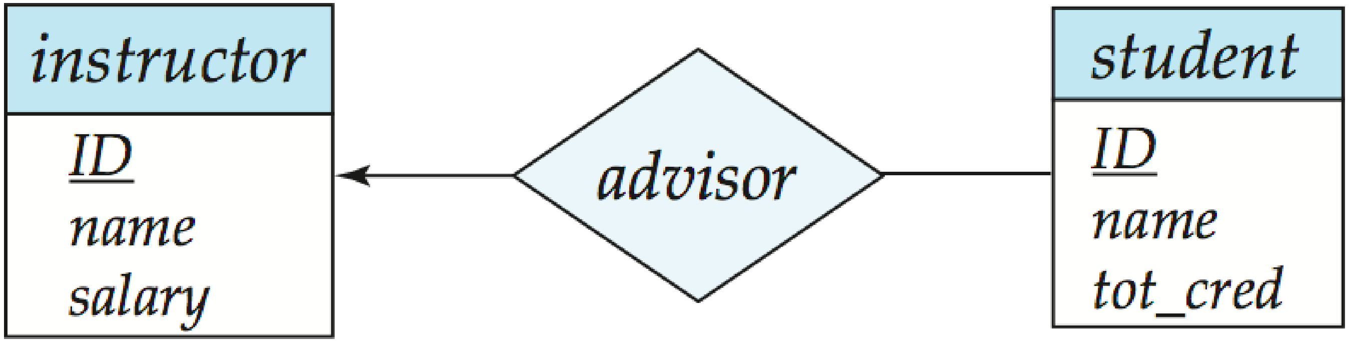
One-to-One Relationship

- One-to-One relationship between an *instructor* and a *student.*
  - an *instructor* is associated with at most one *student* via *advisor.*
  - and a *student* is associated with at most one *instructor* via *advisor*.



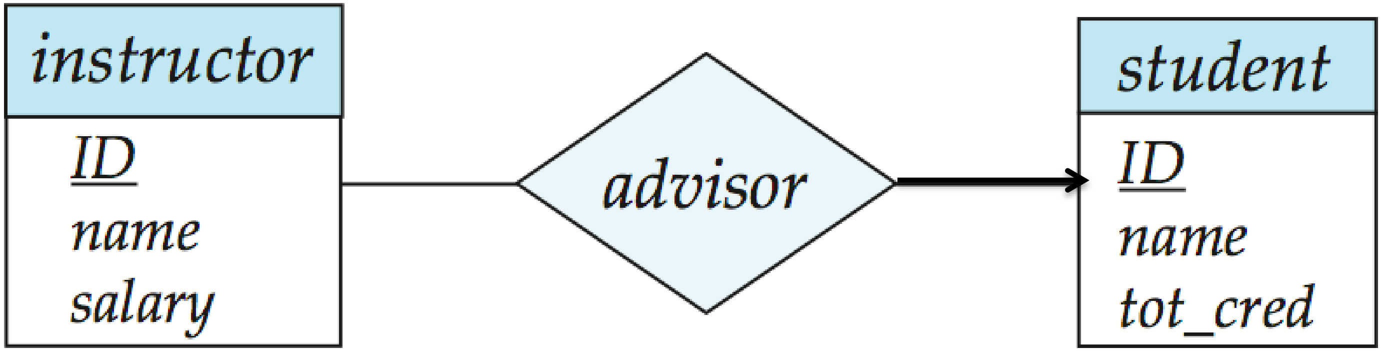
One-to-Many Relationship

- 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*.



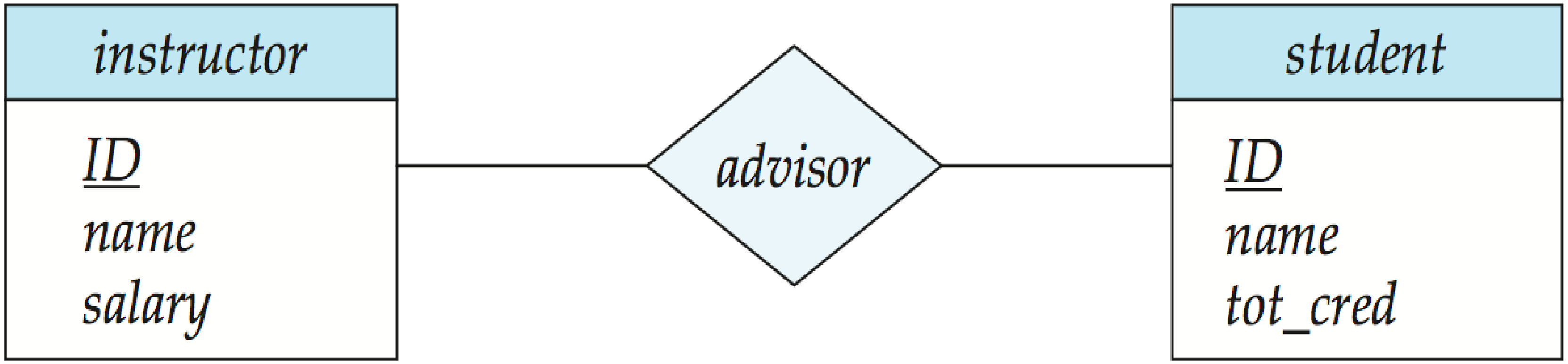
Many-to-One Relationship

- In a many-to-one relationship between an *instructor* and a *student*,
  - an *instructor* is associated with at most one *student* via *advisor*.
  - and a *student* is associated with several (including 0) *instructors* via *advisor.*



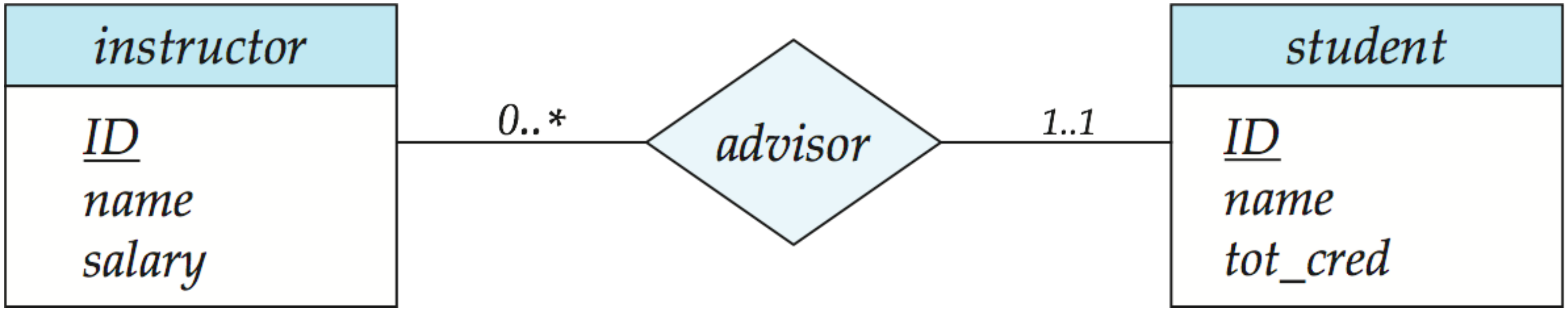
Many-to-Many Relationship

- An *instructor* is associated with several (possibly 0) *students* via *advisor*.
- A *student* is associated with several (possibly 0) *instructors* via *advisor*.



Alternative Notation for Cardinality Limits

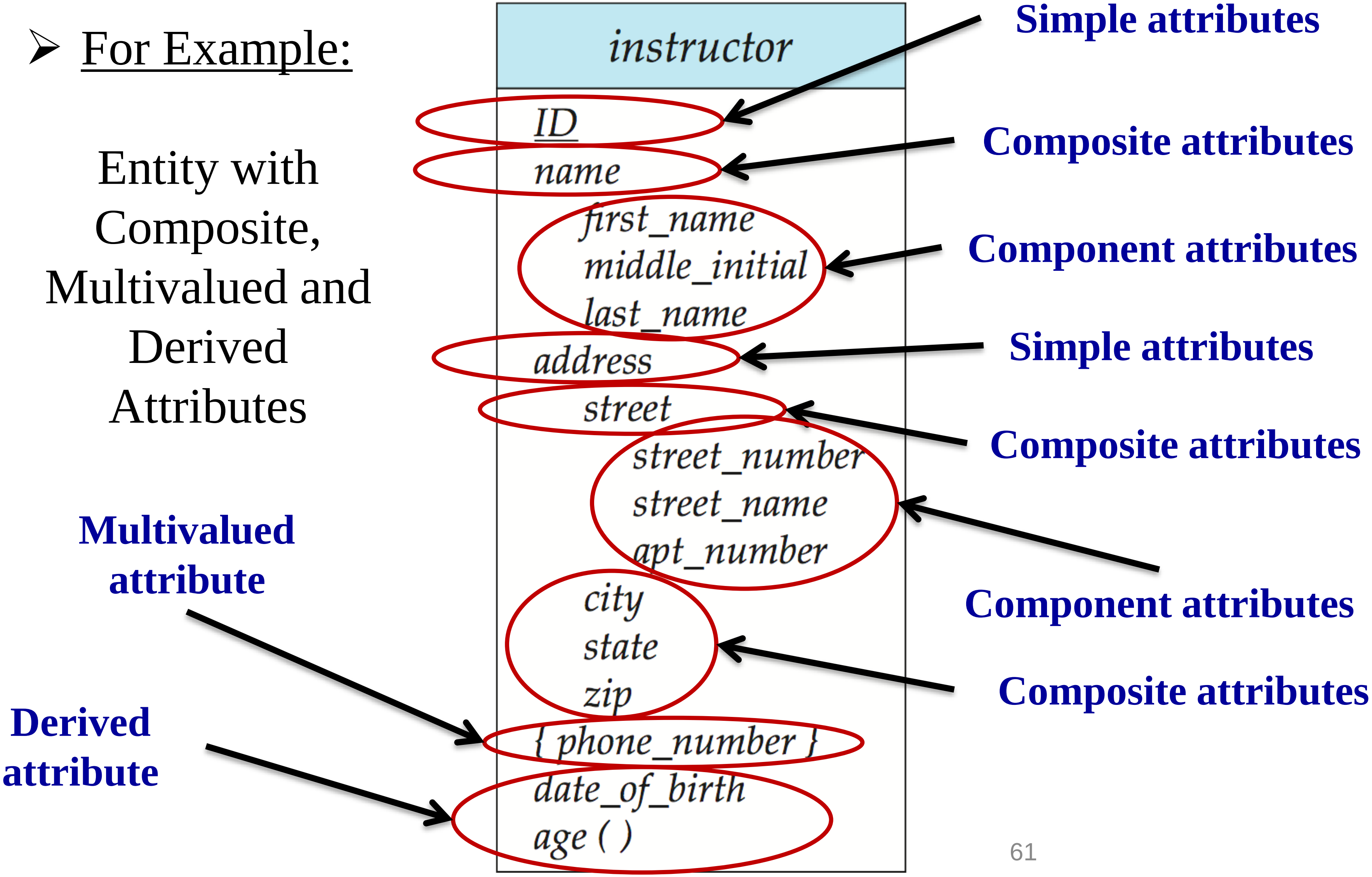
- Cardinality limits can also express participation constraints as given below.



3.3.2 Complex Attributes

- For Example:

Entity with Composite, Multivalued and Derived Attributes



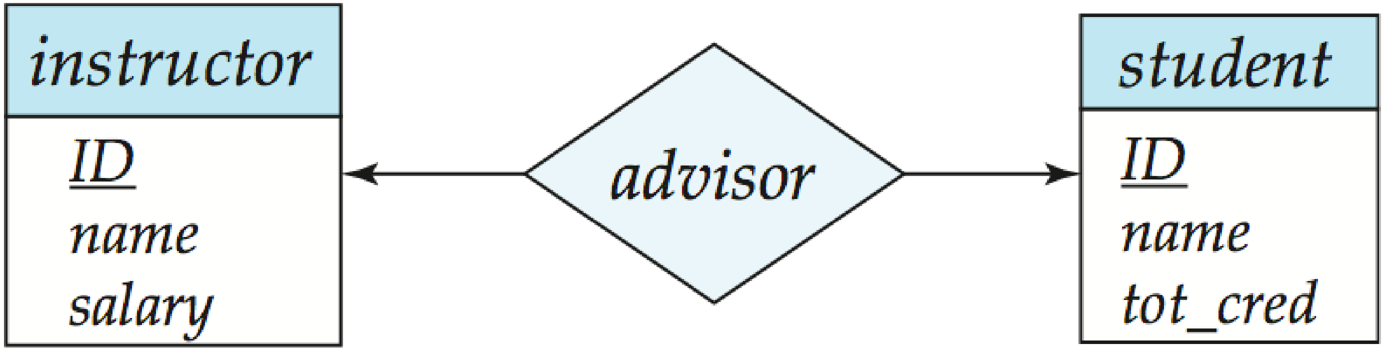
3.3.3. Strong and Weak Entity Sets

- **Strong Entity Sets**:

- An entity set that has primary key or super key is termed as ***strong entity set***.

- A member of a strong entity set is define as ***dominant(one-to-one) entity.***

- For Example:



Weak Entity Set

- An entity set that does not have a primary key is referred to as a ***weak entity set***.

or

- We can say that an entity set may not have sufficient attributes to form a primary key.

- A member of a weak entity set is define as ***subordinate entity***.

- A Weak entity is the one that depends on its **owner entity** i.e. a strong entity for its existence.

Weak Entity Set (Conti…)

❑ **Weak Entity Set (Conti…)**

➢ A weak entity is denoted by the **double rectangle**.

➢ Weak entity do not have the **primary key** instead it has a **partial key** that uniquely discriminates (separates) the weak entities.

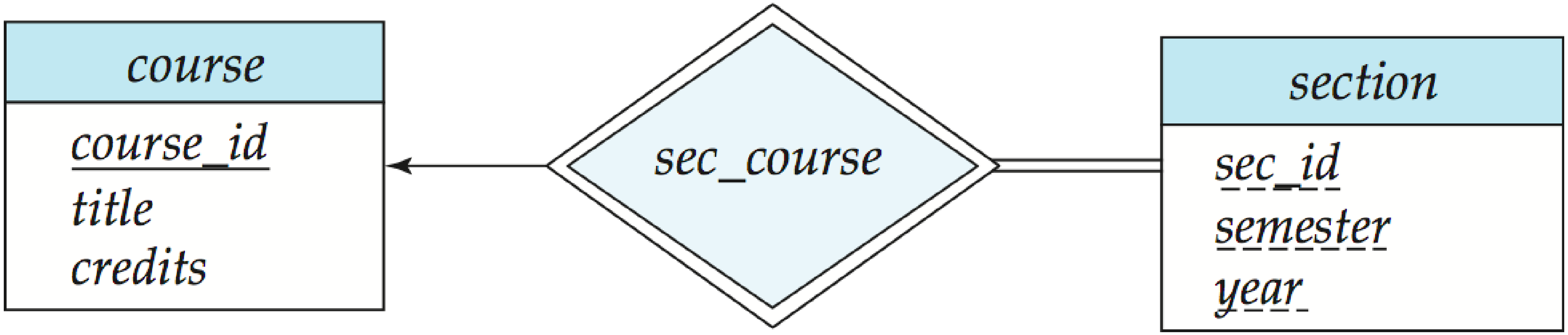
➢ The ***primary key*** *of a weak entity is a composite key* formed from…

- ***primary key*** *of the strong entity* and
- **partial key** of the weak entity.

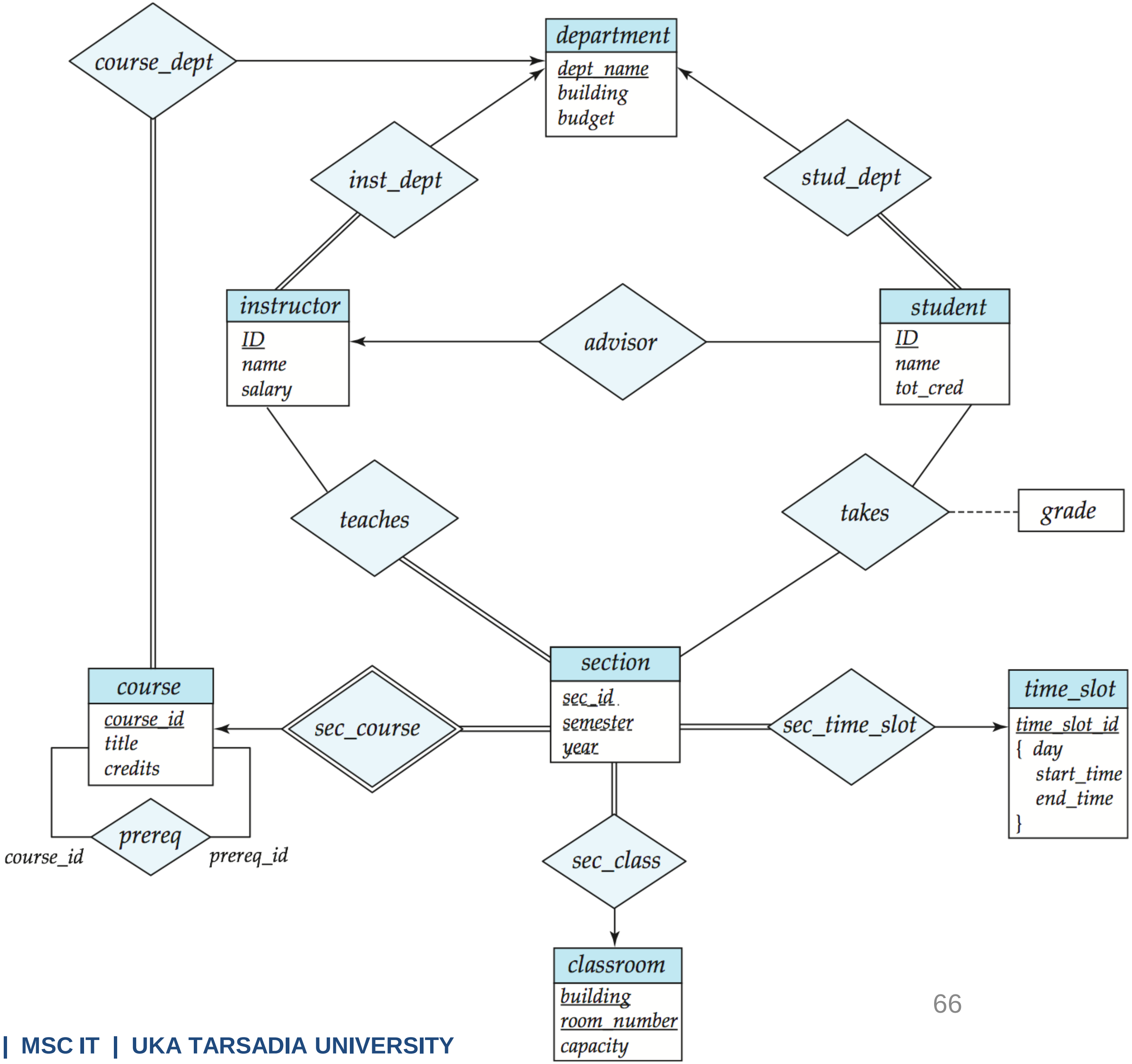
Weak Entity Set (Conti…)

❑ **Weak Entity Set (Conti…)**

➢ Primary key for ‘section’ – (course_id, sec_id, semester, year)



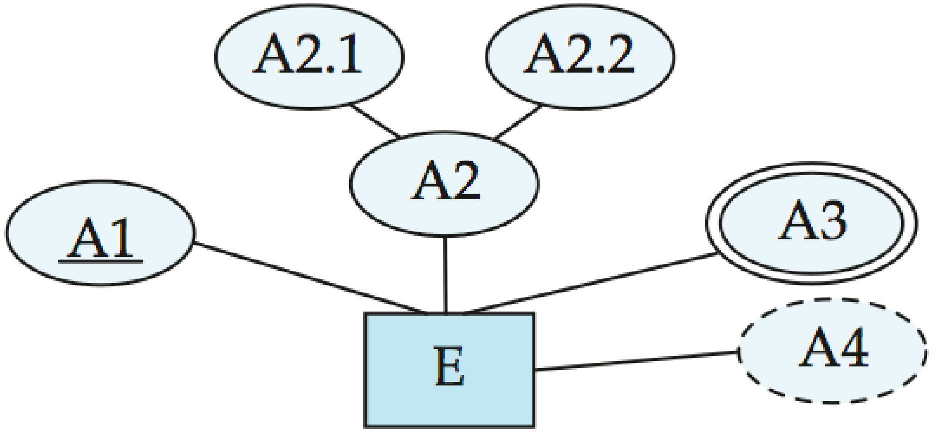
E-R Diagram for a University Enterprise



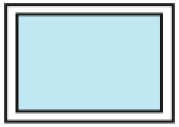
Alternative E-R notations

Chen, IDE1FX, …

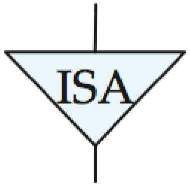
entity set E with
simple attribute A1,
composite attribute A2,
multivalued attribute A3,
derived attribute A4,
and primary key A1



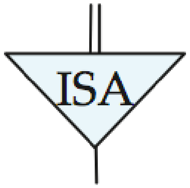
weak entity set



generalization

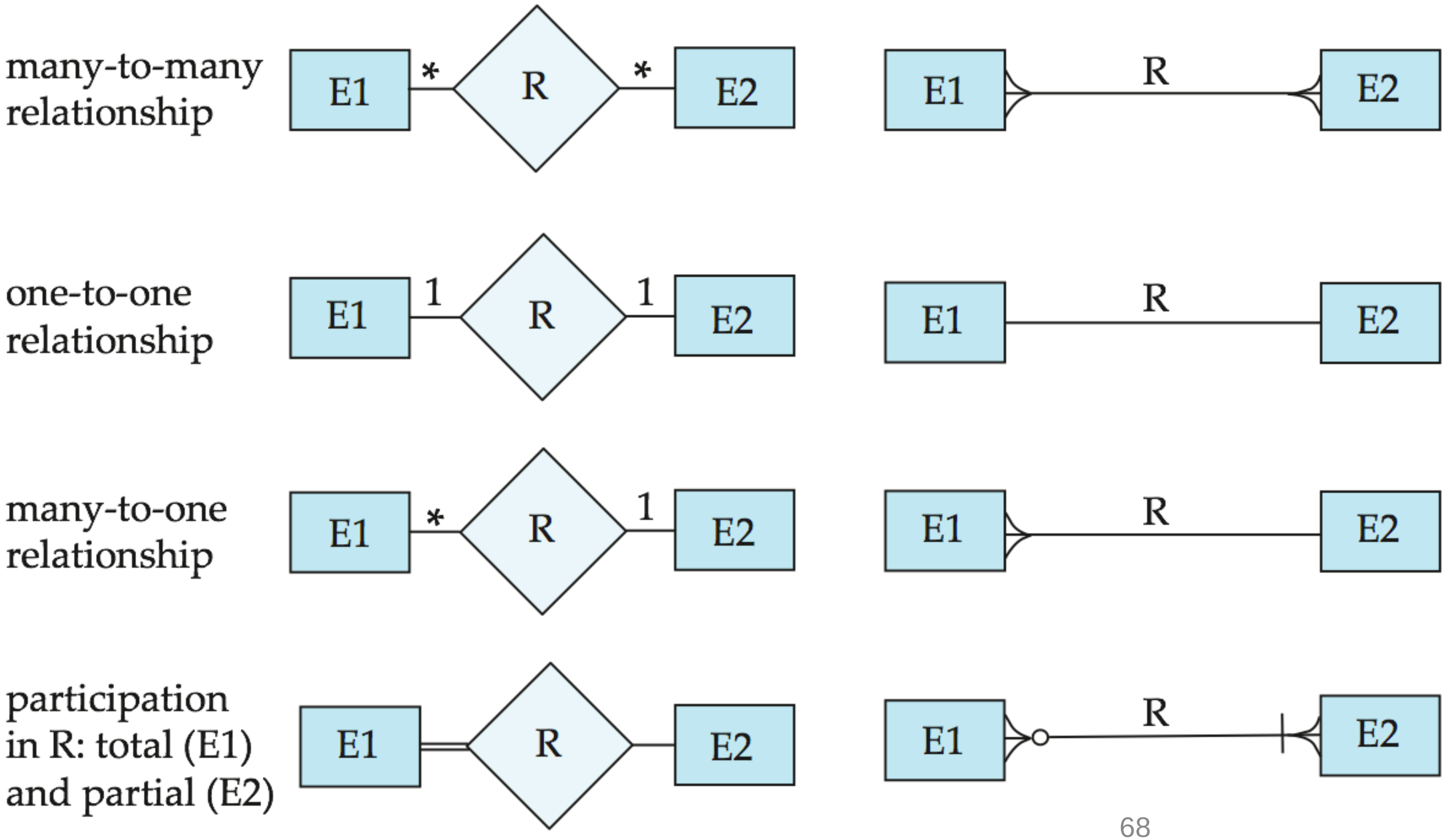


total
generalization



Alternative E-R notations (Conti...)

Chen **IDE1FX (Crows feet notation)**



Combination of Schemas:

➢ For Example:

- In a E-R Diagram of a University Enterprise...

- *inst_dept:* The schemas *instructor* and *department* correspond to the entity sets A and B, respectively.

- Thus, the schema *inst_dept* can be combined with the *instructor* schema.

- The resulting *instructor* schema consists of the attributes….

  {ID, name, dept_name, salary}

Combination of Schemas: (Conti…)

- Consider a many-to-one relationship set AB from entity set A to entity set B.

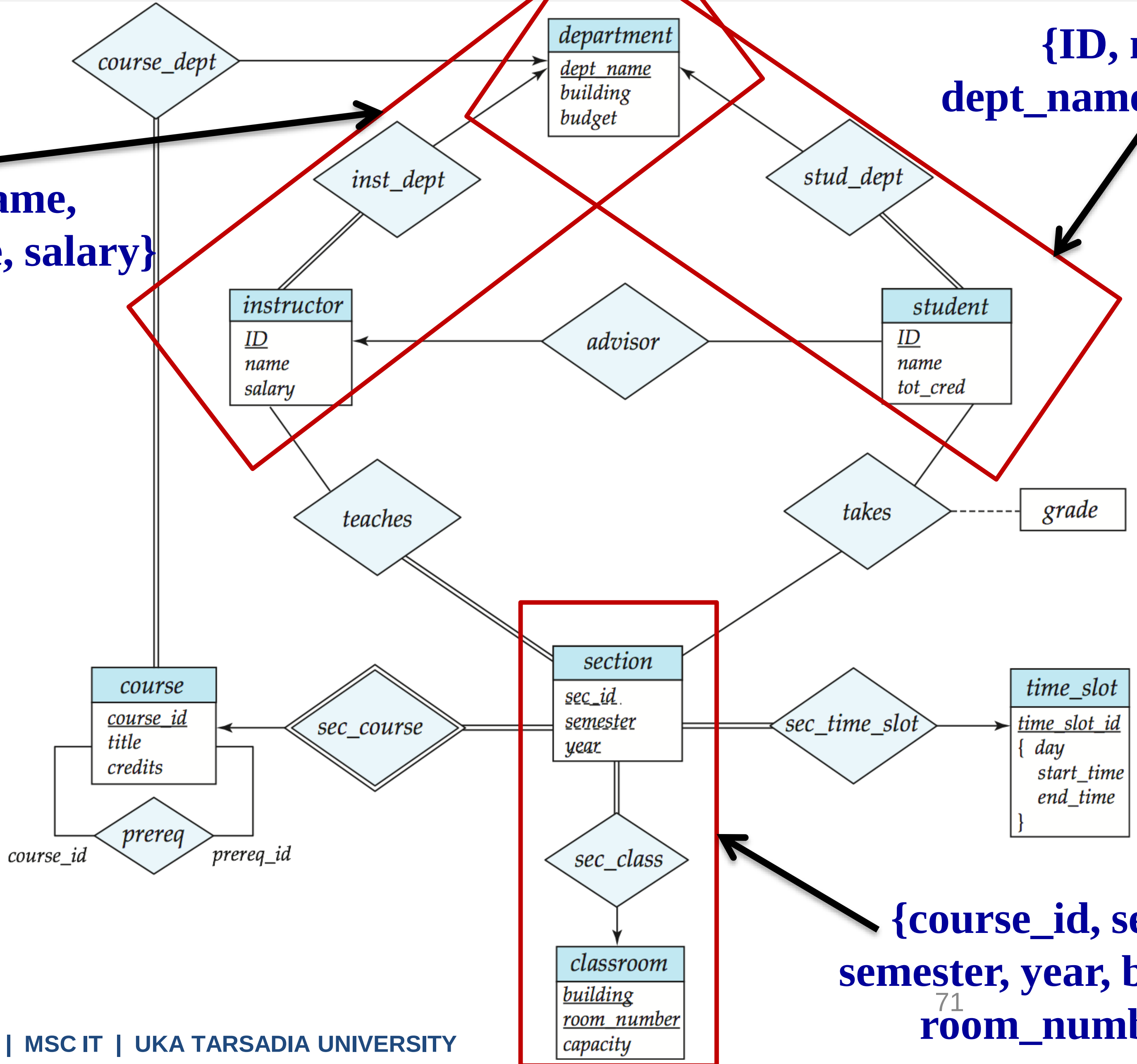
- Suppose, the participation of A in the relationship is *total*, then every entity A in the entity set B must participate in the relationship AB.

- Then we can combine the schemas A and AB to form a single schema which consisting of the union of attributes of both schemas.

- The primary key of the combined schema is the primary key of the entity set into whose schema the relationship set schema was merged.

Demonstration

Combination of Schemas: (Conti…)

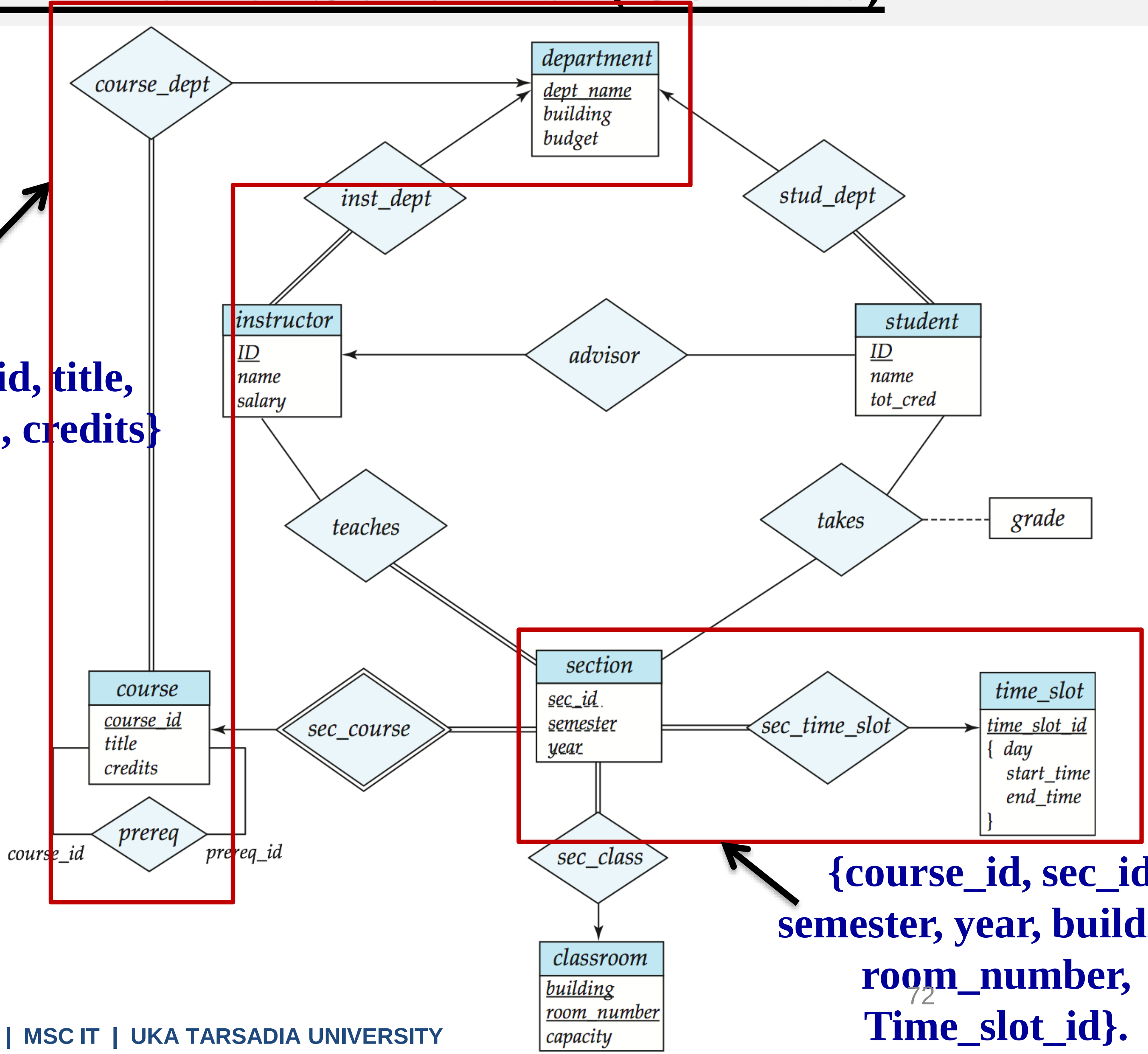


{ID, name, dept_name, salary}

{ID, name, dept_name, tot_cred}

{course_id, sec_id, semester, year, building, room_number}

Combination of Schemas: (Conti…)



{course_id, title, dept name, credits}

{course_id, sec_id, semester, year, building, room_number, Time_slot_id}.

International Certification Question

Q1.	**Diamonds represent relationship sets?**
A.	**False**
B.	**True**

Ans: B

International Certification Question

Q2.	Which attributes are removed out by creating a separate attribute for each component attribute?
A.	Composite
B.	Single-valued
C.	Multi-valued
D.	Derived

Ans: A

International Certification Question

Q3.	**Which entity set has primary key or super key ?**
A.	**Strong Entity Set**
B.	**Weak Entity Set**
C.	**Redundancy of Schemas**
D.	**Combination of Schemas**

Ans: A

International Certification Question

Q4.	**Which entity set becomes a table that includes a column for the primary key of the identifying strong entity set?**
A.	**Strong Entity Set**
B.	**Weak Entity Set**
C.	**Identifying Strong Entity Set**
D.	**Identifying Weak Entity Set**

Ans: B

International Certification Question

Q5.	Which member of a entity set is define as subordinate entity?
A.	Identifying Strong Entity Set
B.	Identifying Weak Entity Set
C.	Strong Entity Set
D.	Weak Entity Set

Ans: D

Industry Interview Questions

1. What is Complex Attributes?
2. What is Schemas?
3. What is the difference between strong entity sets and weak entity sets?
4. How we can reduce the redundancy of schemas?

Home Work

1. Define Strong and Weak Entity Set with proper example. [2 Marks]
2. Explain the representation of Weak Entity Sets. [2 Marks]
3. How we can reduce the redundancy of schemas? [2 Marks]
4. How the schemas are combined? [2 Marks]
5. Explain Mapping Cardinalities with relationship set. [5 Marks]
6. Explain E-R Diagrams with different notations. [5 Marks]
7. Explain E-R Diagrams with different alternative notations. [5 Marks]
8. Explain Strong and Weak Entity Sets with example. [5 Marks]
9. Explain how we can represent the relationship set with examples. [5 Marks]

CE: 3.4
Extended E-R Features

3.4.1. Generalization:

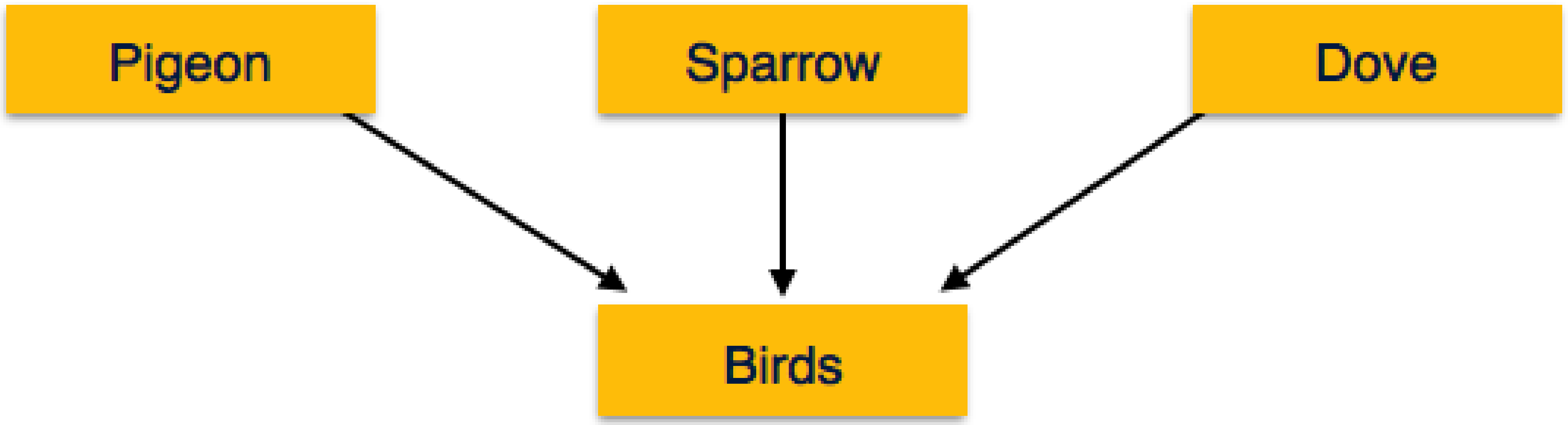
- **Generalization** is a ***bottom-up design* process**, which combines number of entity sets together, to represent more generalized view.

- For Example:
  - A particular student named "Raj" can be generalized along with all the students.

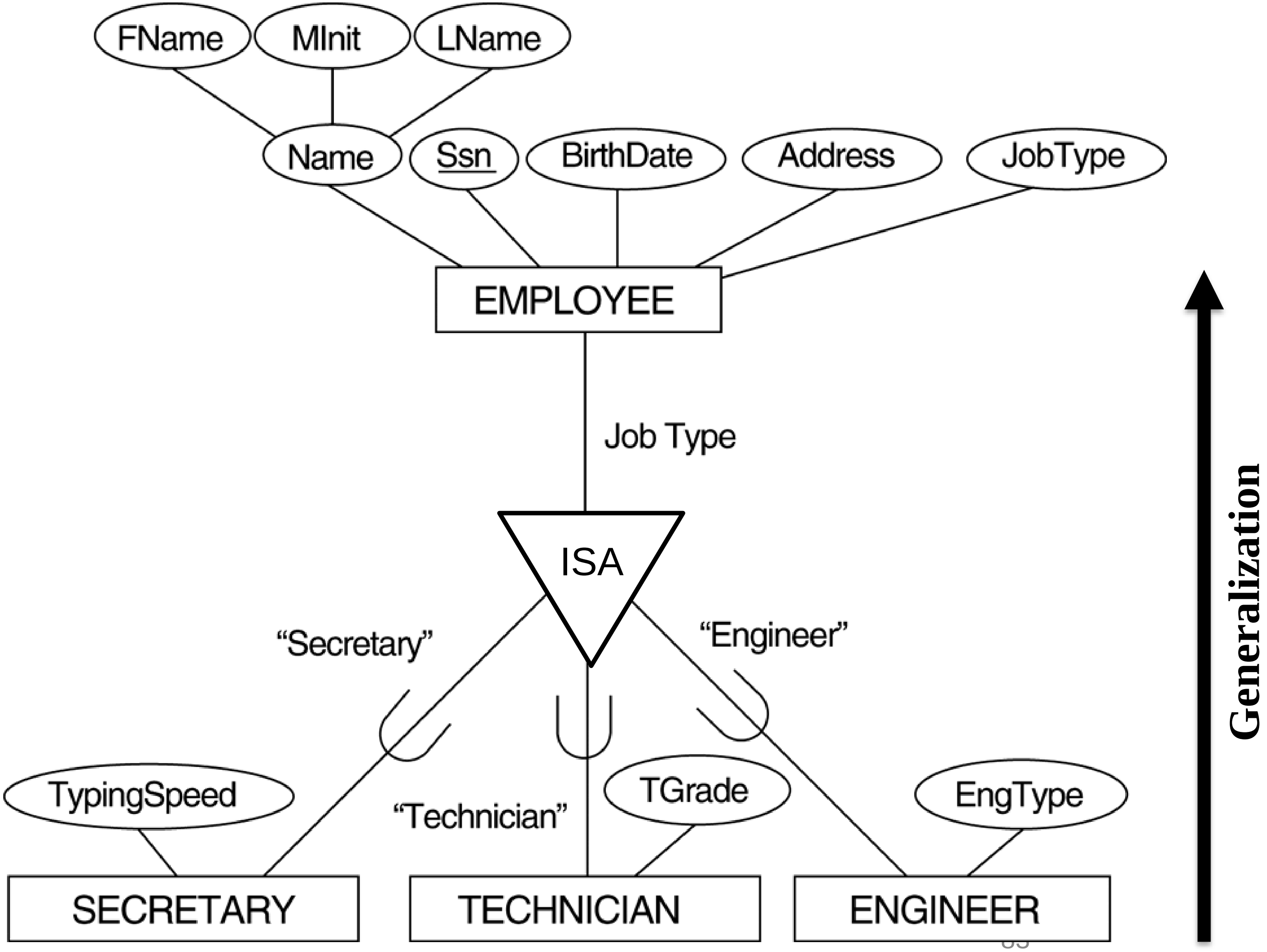
- For Example:
  - Consider the entity types **CAR** and **TRUCK**, they have common attributes like model number, manufacturer name, etc. and can be generalized into the entity type **VEHICLE**.

3.4.1. Generalization: (Conti…)

- **For Example:**
  - Pigeon, house sparrow, crow and dove can all be generalized as birds.



3.4.1. Generalization: (Conti…)



3.4.2. Specialization:

- **Specialization** is *top-down design* process.

- It is the process to split up the entities into further sub-entities on the basis of their functionalities, specialties and features.

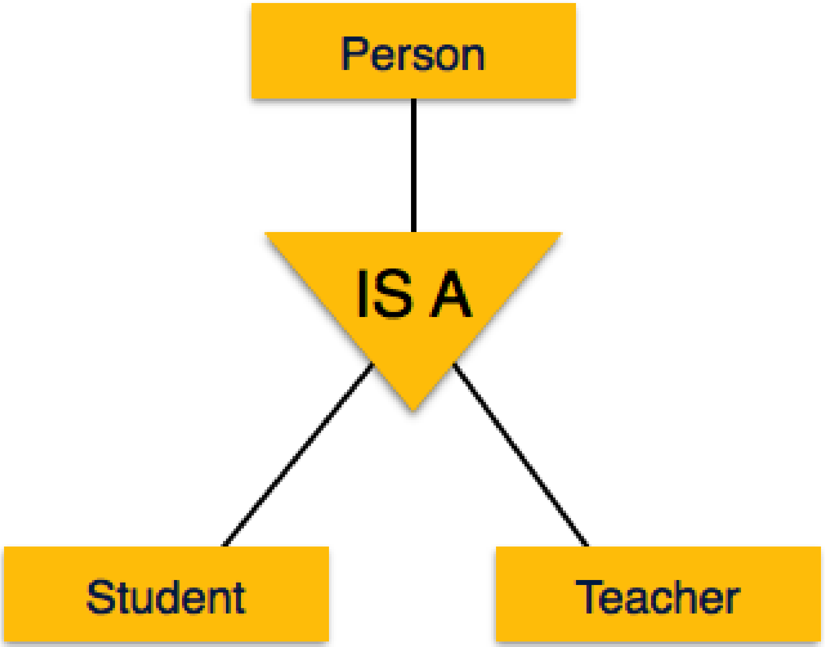
- These sub-groupings become *lower-level entity sets* that have attributes.

- It is represented by a triangle component labeled **ISA** (E.g., Person "is a" Student or Teacher).

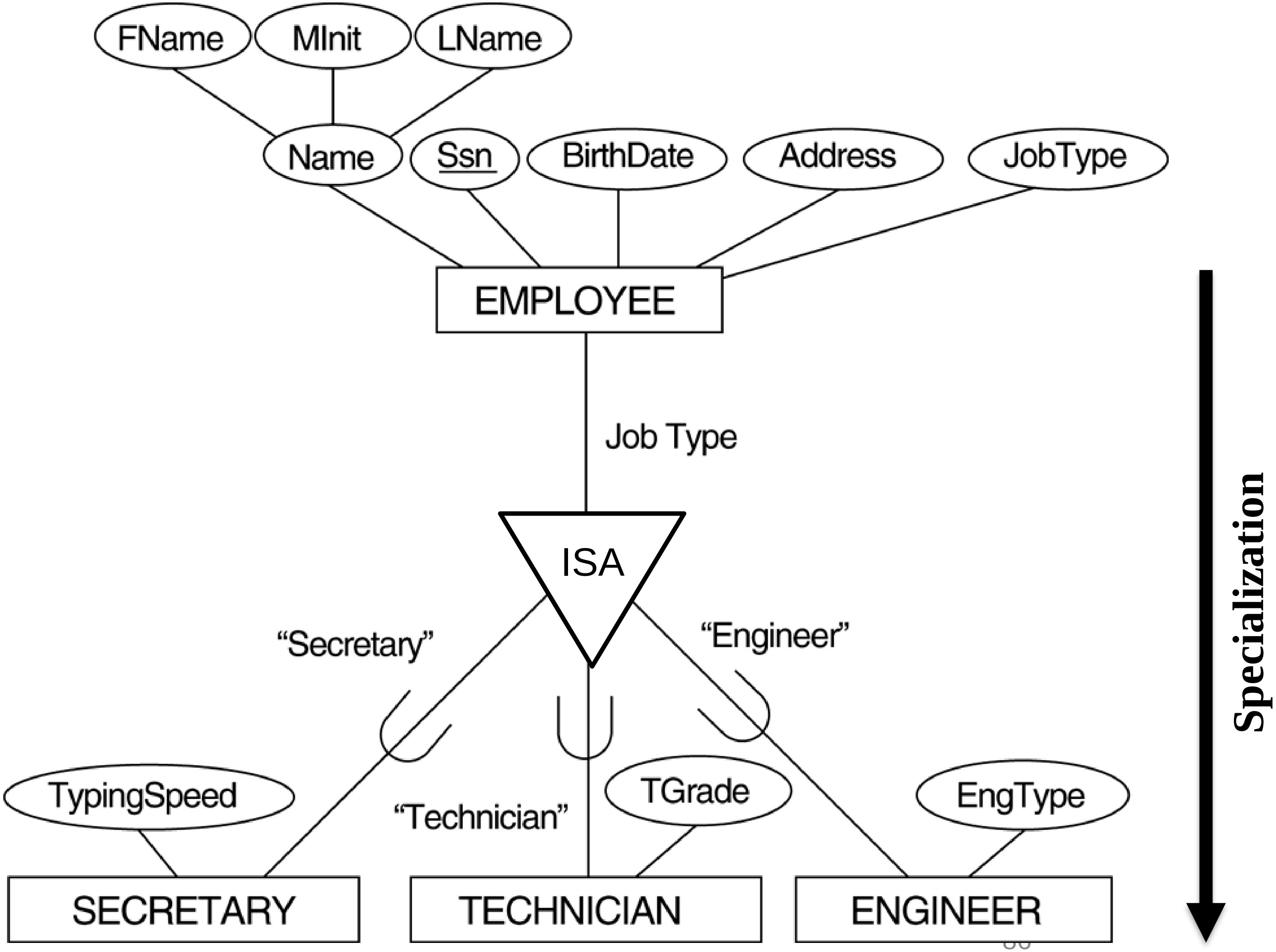
3.4.2. Specialization: (Conti…)

- For Example:

A person has name, date of birth, gender, etc. These properties are common in all persons, human beings. But in a company, persons can be identified as employee, employer, customer, or vendor, based on what role they play in the company.



3.4.2. Specialization: (Conti…)



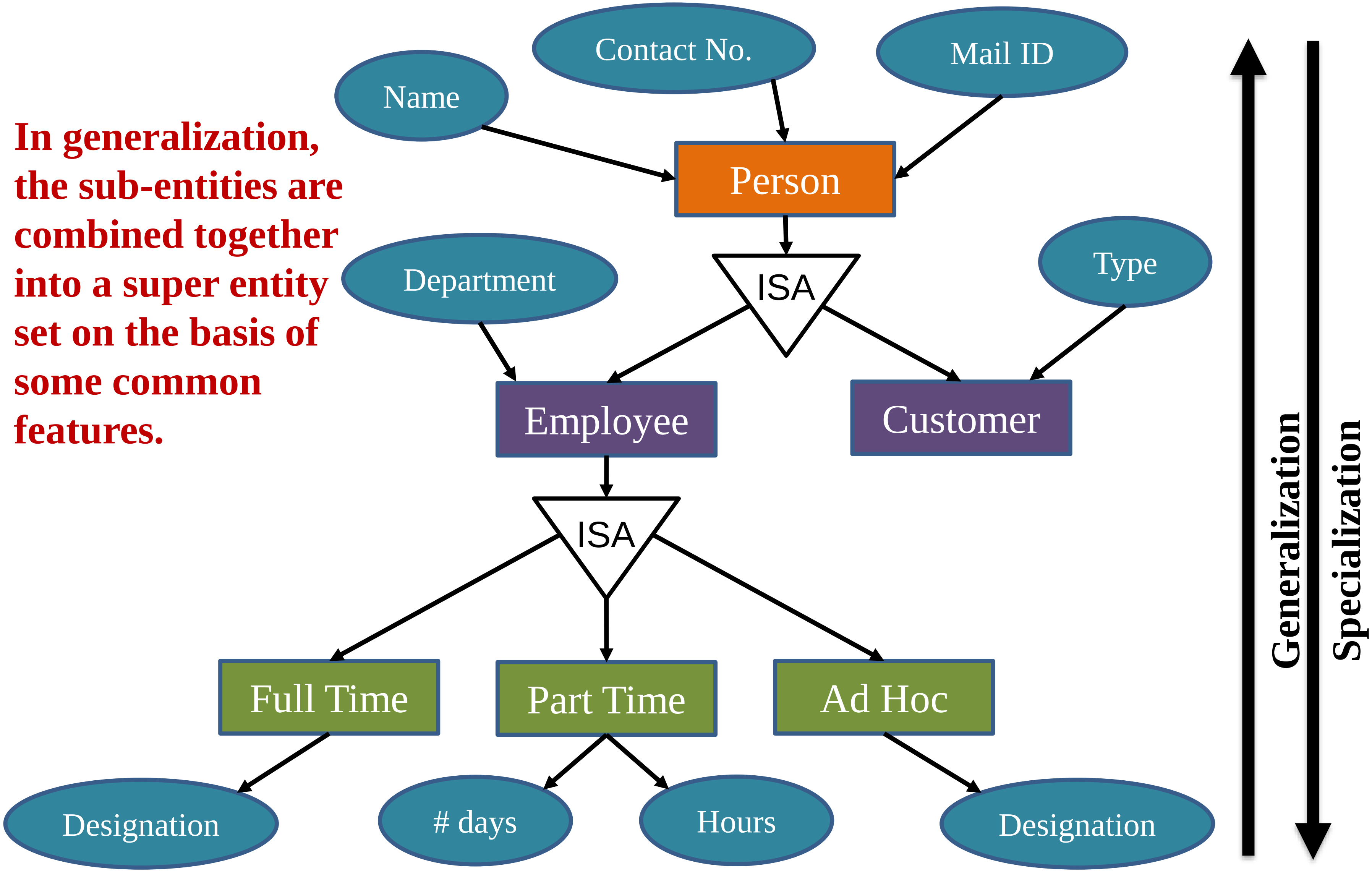
Specialization and Generalization (Conti...)

- Based on different features, it can have multiple specializations of an entity set.

- The ISA relationship also referred to as **superclass - subclass** relationship.

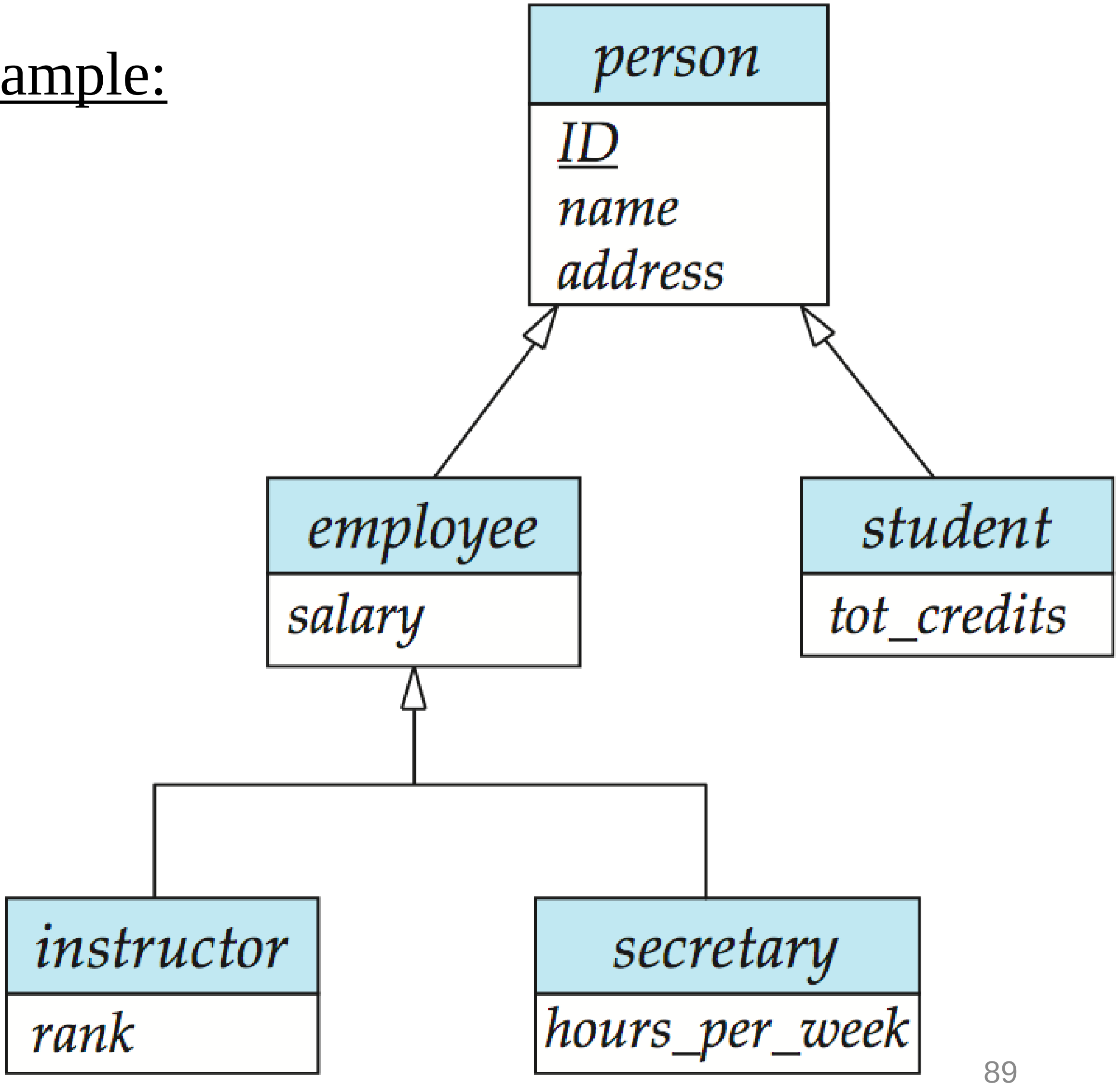
Specialization and Generalization (Conti...)

In generalization, the sub-entities are combined together into a super entity set on the basis of some common features.



Specialization and Generalization (Conti...)

- For Example:



Design Constraints on a Specialization/Generalization:

- Constraint on which entities can be members of a given lower-level entity set are:

    1. Condition-defined:
        - Example: all customers over 65 years are members of senior-citizen entity set; senior-citizen ISA person.
    2. User-defined:

- Constraint on whether or not entities may belong to more than one *lower-level entity* set within a single generalization.

    - Disjoint:
        - ✓ an entity can belong to only one lower-level entity set.
        - ✓ Note: In E-R diagram, *lower-level entity sets* have multiple link to the same triangle.

Design Constraints on a Specialization/Generalization: (Conti…)

- Overlapping:
  - ✓ an entity can belong to more than one *lower-level entity set*.

➢ The completeness constraint --- specifies whether or not an entity in the *higher-level entity set* must belong to at least one of the *lower-level entity sets* within a generalization.

a) Total:
   - An entity must belong to one of the *lower-level entity sets*.

b) Partial:
   - An entity need not belong to one of the *lower-level entity sets*.

3.4.3. **Attribute inheritance:**

- Attribute inheritance is a *lower-level entity set* which….
  - inherits all the attributes and
  - relationship participation

  of the *higher-level entity set* to which it is linked.

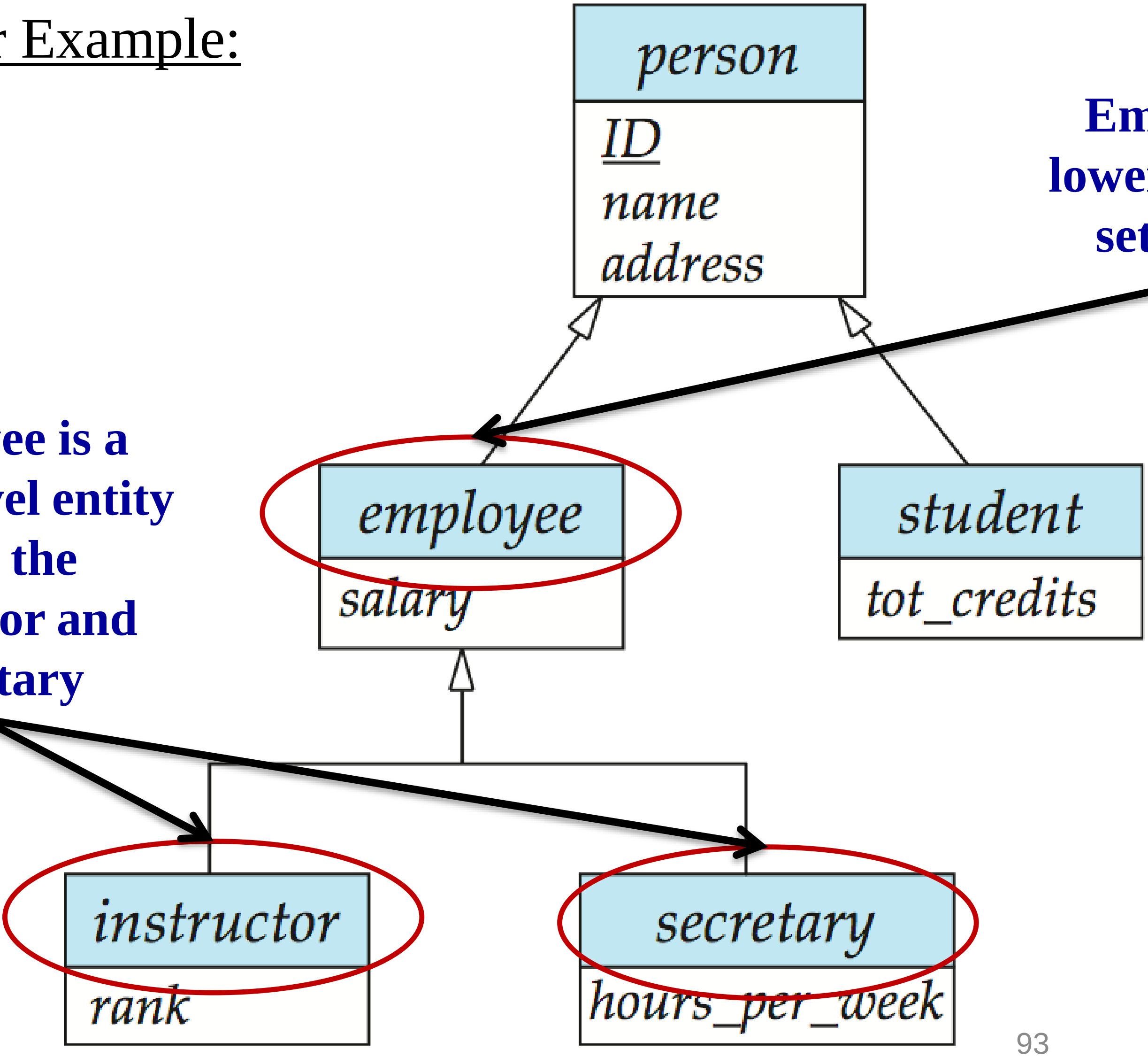
- We can also say that…

  The attributes of the *higher-level entity sets* are said to be inherited by the *lower-level entity sets*.

- An inheritance of the attribute is key property of the *higher and lower level* entities which is created by *specialization* and *generalization*.

3.4.3. Attribute inheritance: (Conti…)

- For Example:

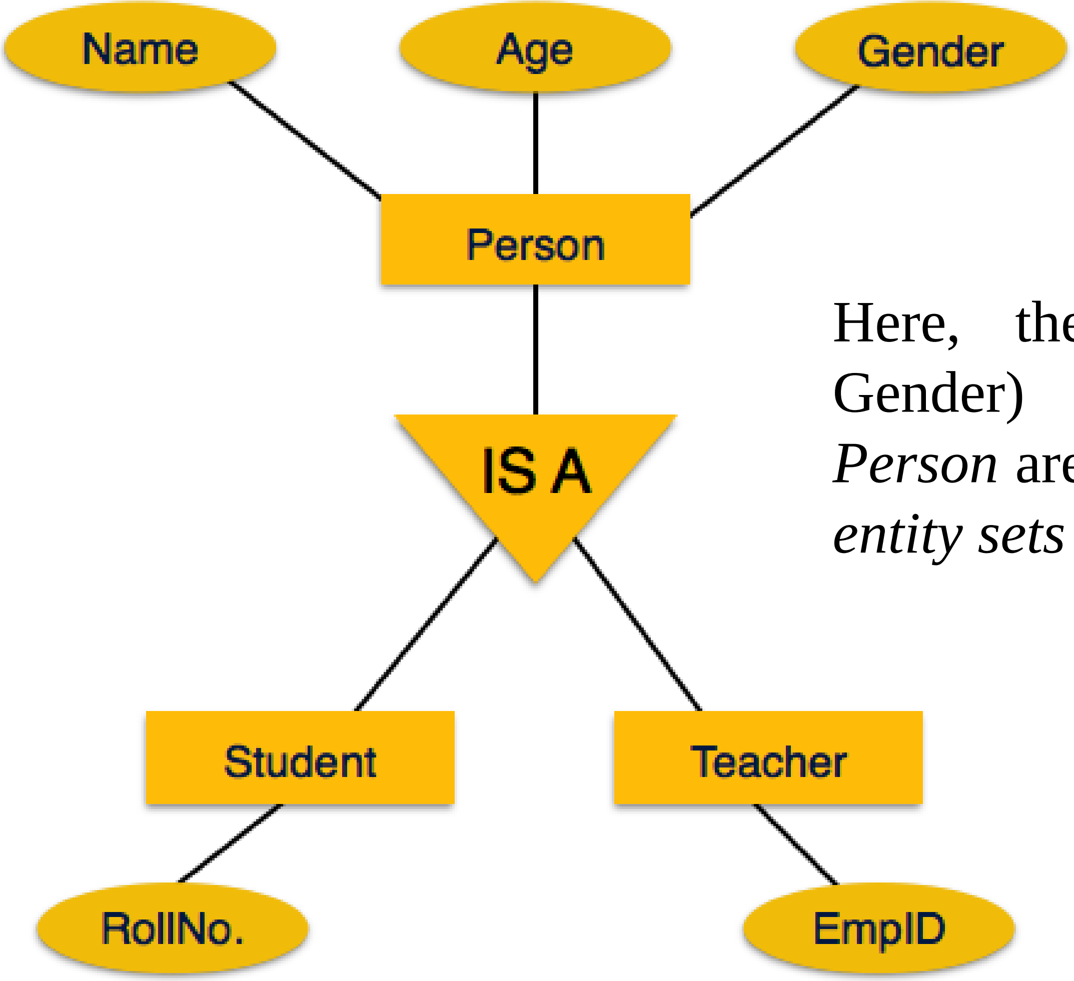


Employee is a lower-level entity set of Person

Employee is a higher-level entity set of the Instructor and Secretary

3.4.3. Attribute inheritance: (Conti…)

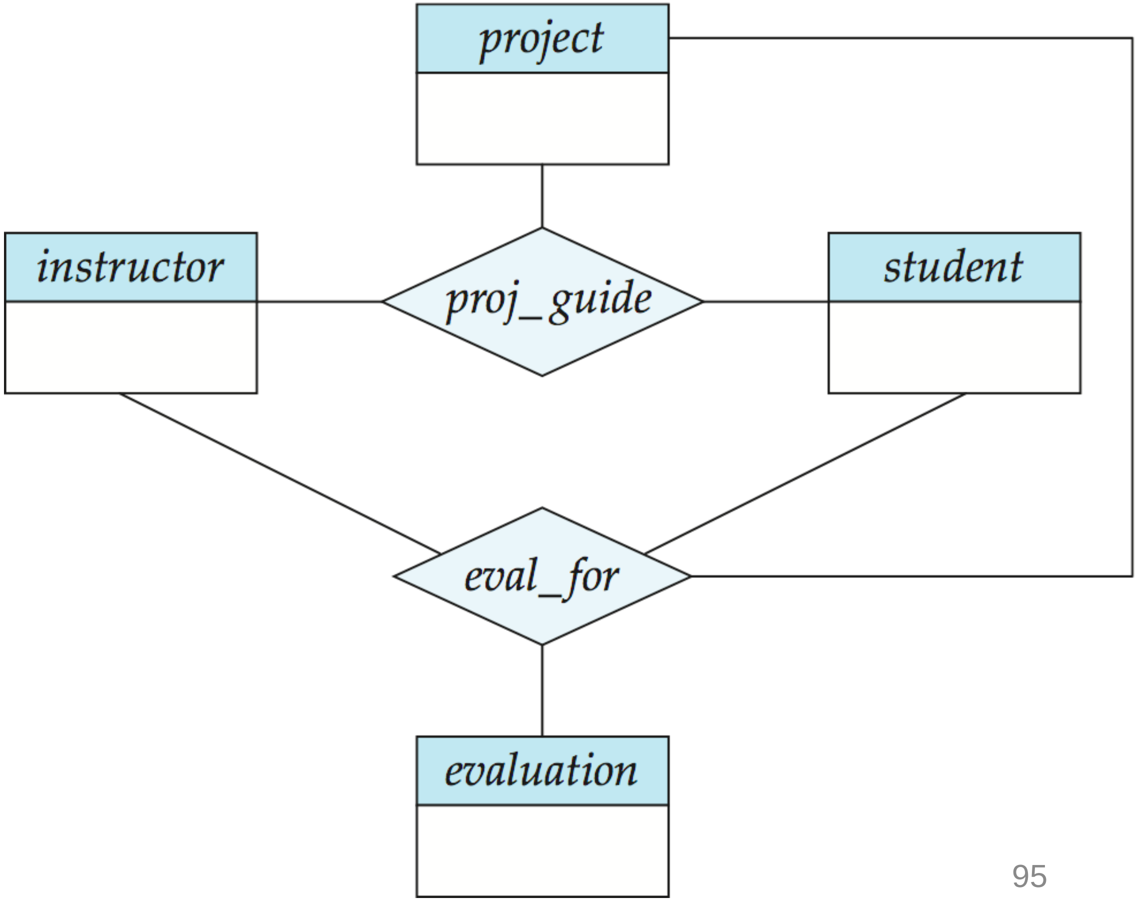
- For Example:



Here, the attributes (Name, Age, Gender) of the *higher-level entity Person* are inherited by the *lower-level entity sets* (Student and Teacher).

3.4.4. Aggregation:

- Consider the ternary relationship *proj_guide*.
- Suppose we want to record evaluations of a student by a guide on a *project*.



3.4.4. Aggregation: (Conti…)

- Relationship sets ***eval_for*** and ***proj_guide*** represent overlapping information.

  - Every ***eval_for*** relationship corresponds to a ***proj_guide*** relationship.

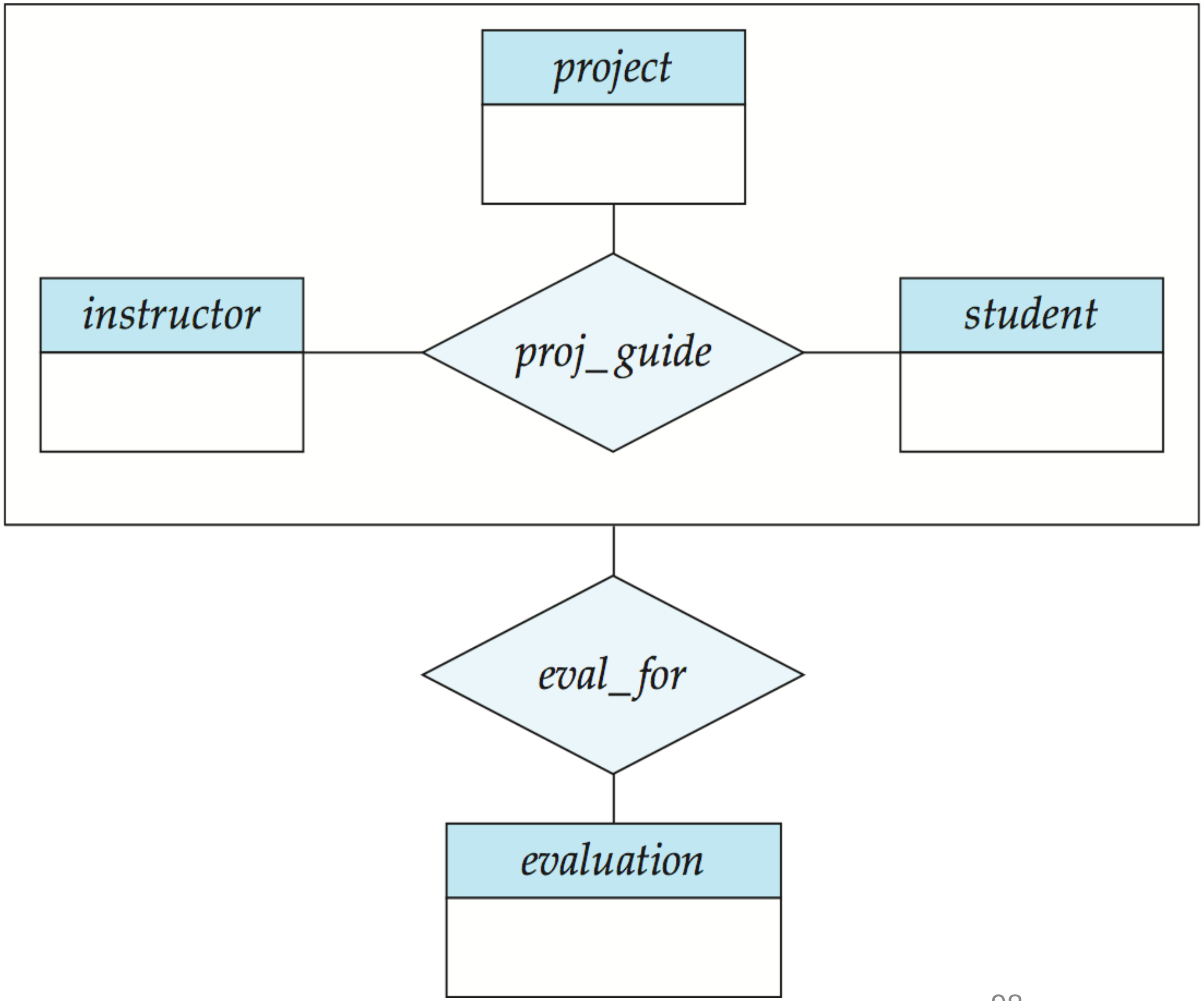
  - However, some ***proj_guide*** relationships may not correspond to any ***eval_for*** relationships.
    - ✓ So we can't discard the ***proj_guide*** relationship.

- Eliminate this redundancy via aggregation….
  - Treat relationship as an abstract entity.
  - Allows relationships between relationships.
  - Abstraction of relationship into new entity.

3.4.4. Aggregation: (Conti…)

- Without introducing redundancy, the following diagram represents:
  - A student is guided by a particular instructor on a particular project.
  - A student, instructor, project combination may have an associated evaluation.

3.4.4. Aggregation: (Conti…)



Schemas Corresponding to Aggregation

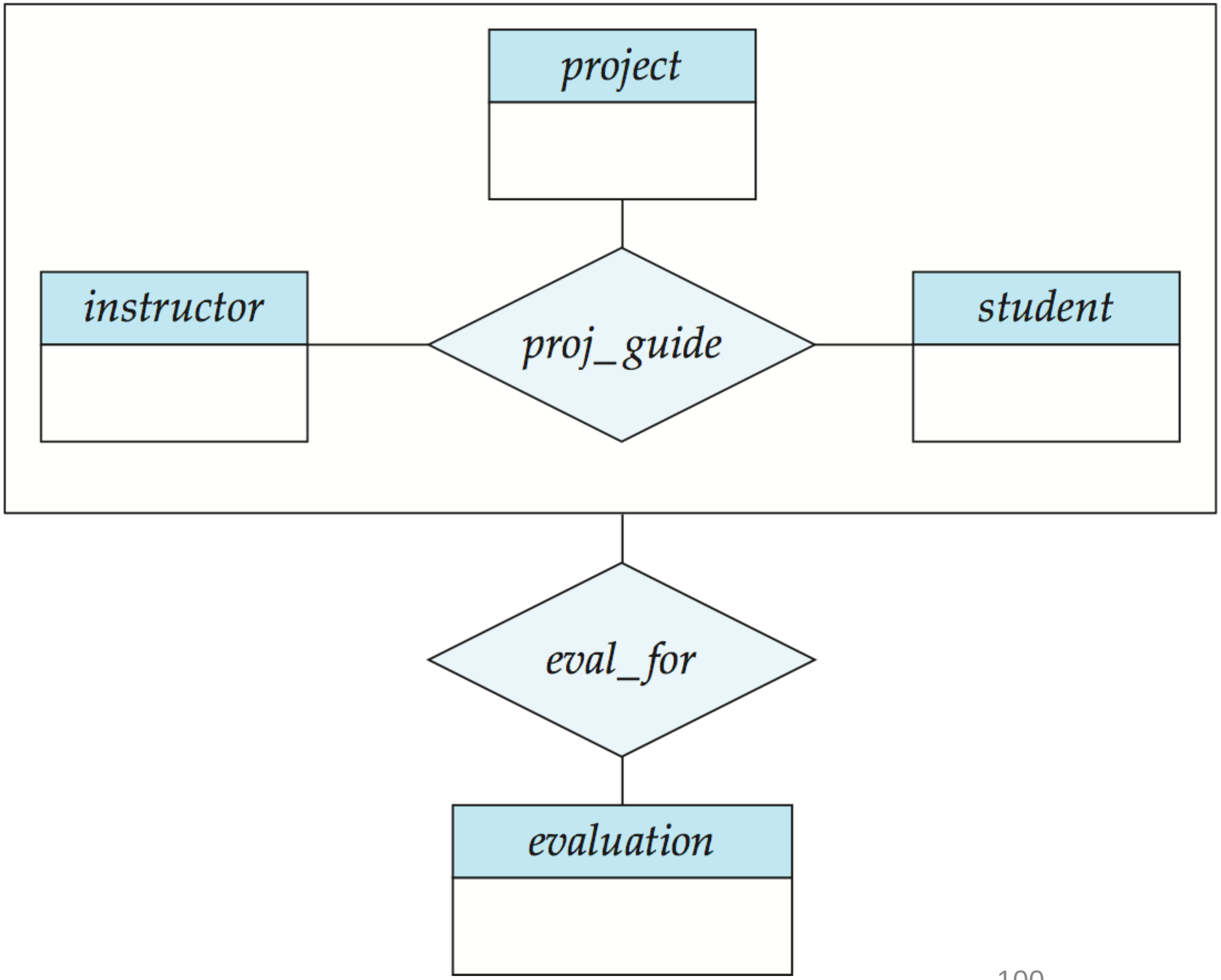
- To represent aggregation, create a schema containing
  - primary key of the aggregated relationship,
  - the primary key of the associated entity set
  - any descriptive attributes

- For Example:
  - To represent aggregation *manages* between relationship *works_on* and entity set *manager*, create a schema as…

    eval_for (s_ID, project_id, i_ID, evaluation_id)

- Schema *proj_guide* is redundant provided we are willing to store null values for attribute *manager_name* in relation on schema *manages*.

Schemas Corresponding to Aggregation(Conti...)



International Certification Question

Q1.	Which design process follow Bottom-up approach?
A.	Specialization
B.	Generalization
C.	Attribute Inheritance
D.	Aggregation

Ans: B

International Certification Question

Q2.	Which design process inherits all the attributes and relationship participation of the higher-level entity set to which it is linked?
A.	Specialization
B.	Generalization
C.	Attribute Inheritance
D.	Aggregation

Ans: C

International Certification Question

Q3.	Which design process is the combination of Specialization and Generalization?
A.	Physical Design
B.	Logical Design
C.	Attribute Inheritance
D.	Aggregation

Ans: D

International Certification Question

Q4.	**Which entity can belong to more than one lower-level entity set?**
A.	Disjoint
B.	Overlapping
C.	Total
D.	Partial

Ans: B

International Certification Question

Q5.	Which entity need not belong to one of the lower-level entity sets?
A.	Total
B.	Partial
C.	Disjoint
D.	Overlapping

Ans: B

Industry Interview Questions

1. What is Generalization?
2. What is Specialization?
3. What is Aggregation?

Home Work

1. What is disjoint? [1 Mark]
2. What is overlapping? [1 Mark]
3. Explain Attribute inheritance in short with example. [2 Marks]
4. Explain the concept of generalization and specialization with proper example. [5 Marks]
5. What are the extended features of E-R Model? Explain with example. [5 Marks]
6. Create E-R Model for Customers-Suppliers-Products. [5 Marks]
7. Create E-R diagram for a Bank Database. [5Marks]
8. Explain the concept of generalization, specialization and aggregation. Explain it with example. [7 Marks]

Thank you