

Database midterm notes

## Week 1

Two fundamental types of database implementations

- Transactional database
- Data warehouse

Relational model

- Transactional model

Multidimensional model

- Data warehouse

**Relational model** is **normalized** while **multidimensional** is **de-normalized**

### Conceptual database design

- Independent of any physical considerations
- Built using the user requirement specifications information
- Source of information for logical design phase
- Major entities and relationships (M:N permitted)

### Logical database design

- Based on a specific data model (relational)
- Independent of any DBMS or any physical considerations
- Refined conceptual data model
- All entities and relationships, full attribute specifications, unique identifiers and M:N relationships are resolved.

### Physical database design

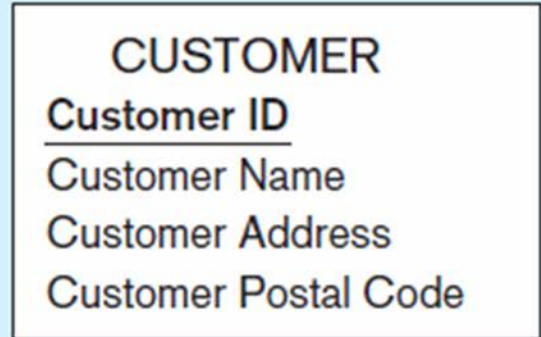
- Description of implementing the database on a secondary storage
- Base relations, file organization, indexes, integrity constraints
- Dependent on a specific DBMS

## Transforming ER diagrams into relations

### Mapping regular entities to relations

- Simple attributes: E-R attributes map directly onto the relation
- Composite attributes: use only their simple, component attributes
- Multivalued attribute: Becomes a separate relation with a foreign key taken from the superior entity

CUSTOMER entity  
type with simple  
attributes →



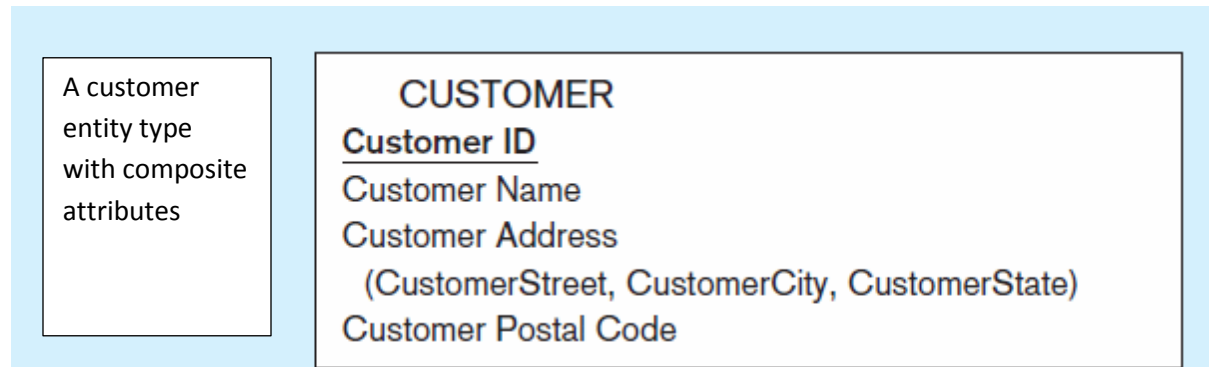
**Customer relation**

CUSTOMER

<u>CustomerID</u>	CustomerName	CustomerAddress	Cu
-------------------	--------------	-----------------	----

A blue arrow points from the text 'Customer relation' to the first column header of the table below. The table has a header row with four columns. The first column is labeled 'CustomerID', the second 'CustomerName', the third 'CustomerAddress', and the fourth 'Cu'.

## MAPPING A COMPOSITE ATTRIBUTE



Customer relation with address detail

### CUSTOMER

<u>CustomerID</u>	CustomerName	CustomerStreet	CustomerCity	CustomerState	CustomerPostalCode
-------------------	--------------	----------------	--------------	---------------	--------------------

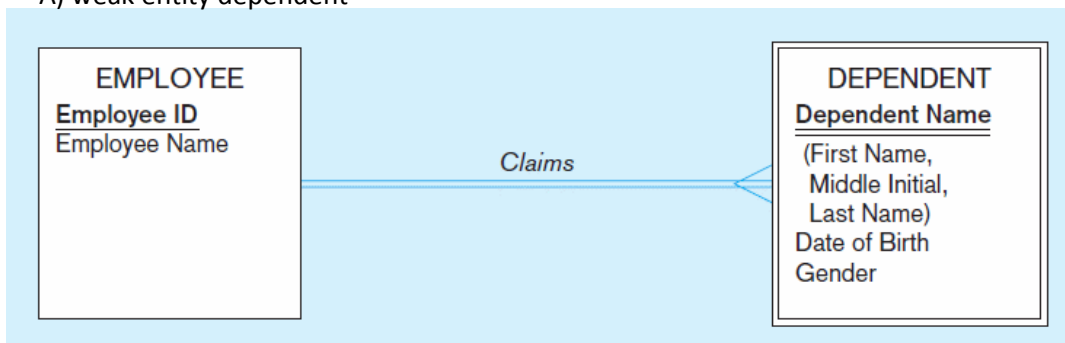
Mapping an entity with a multivalued attribute

- Look at slide 16
- Multivalued attribute becomes a separate relation with foreign key

### Transforming ER diagrams into relations (cont)

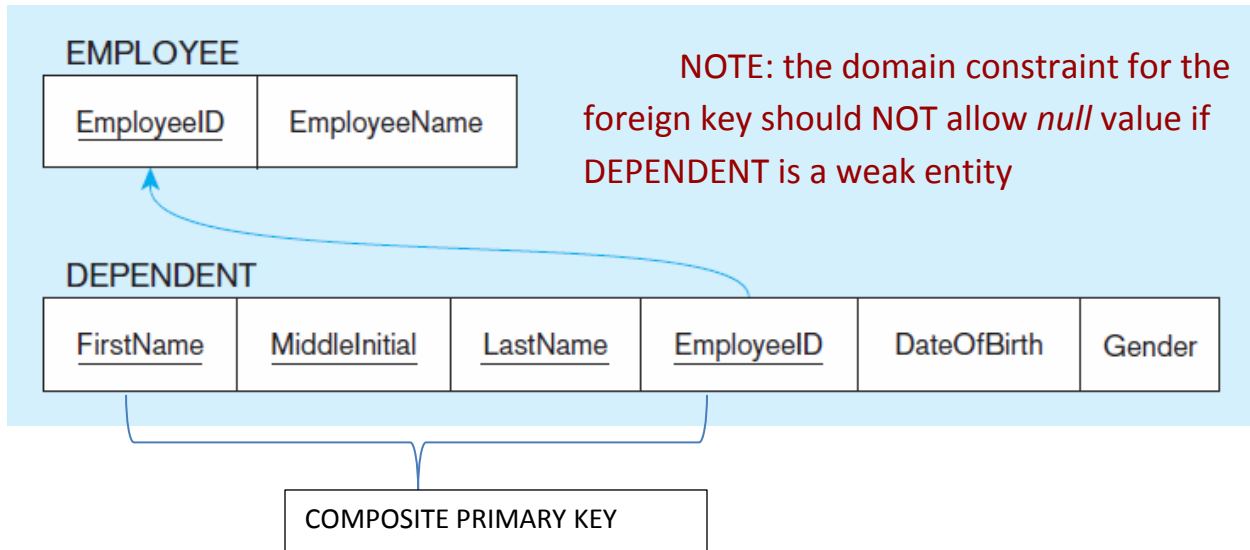
Mapping weak entities

- Becomes a separate relation with a foreign key taken from the superior entity
- Primary key composed of
  - o Partial identifier of weak entity
  - o Primary key of identifying relation (strong entity)
- An example of mapping a weak entity
- A) weak entity dependent



Example of mapping a weak entity

b) relations resulting from weak entity



### TRANSFORMING EER DIAGRAMS INTO RELATIONS (CONT)

- Mapping binary relationships
  - o One to many : primary key on the one side becomes a foreign key on the many side
  - o Many to many : create a **new relation** with the primary keys of the two entities as the primary key
  - o One to one: primary key on the mandatory side becomes a foreign key on the optional side

Look at slide 21-23

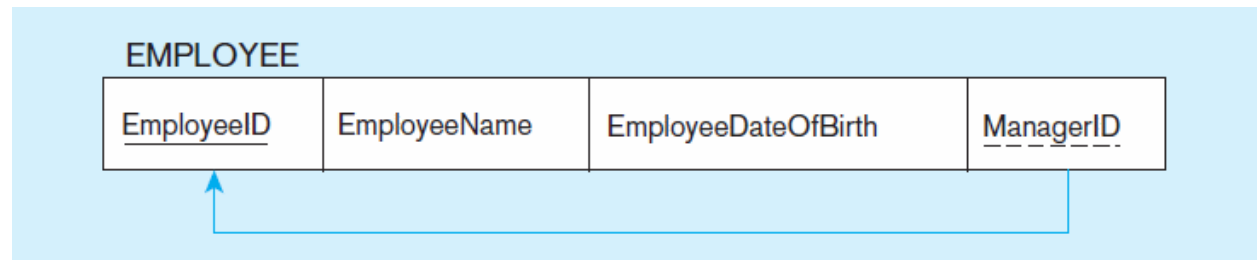
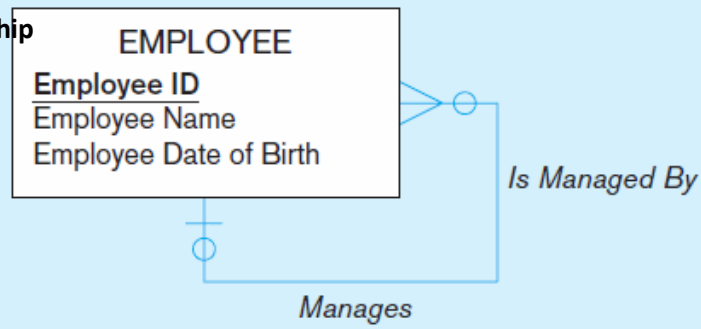
### Transforming ER Diagrams into relations

Mapping unary relationships

- One to many : recursive foreign key in the same relation
- Many to many: two relations:
  - o One for the entity type
  - o One for an associative relation in which the primary key has two attributes, both taken from the primary key of the entity

## Mapping a unary 1:n relationship

employee entity  
with unary relationship



Employee relation with recursive foreign key

## Lab 1

Ali Houmani

13W\_CST2355 Lab 1

1

Part 1:

Choose the right answer

**1. Redundancy refers to or leads to \_\_\_E\_\_\_\_\_**

- (a) The storage of the same data in more than one location
- (b) The same data are repeated in multiple tuples in the same relation or multiple relations
- (c) The wastage of storage space
- (d) Certain update anomalies, such as insertion, deletion and modification anomaly, can arise
- (e) All of the above

**2. A relation is in first normal form if \_\_\_D\_\_\_\_\_.**

- (a) All the key attributes are defined
- (b) There are no repeating groups in the relation
- (c) All attributes are dependent on the primary key
- (d) All of these

**3. A relation in which every non-key attribute is fully functionally dependent on the primary key and which has no transitive dependencies is said to be in \_\_\_A\_\_\_\_\_.**

- (a) 3NF
- (b) 2NF
- (c) 1NF
- (d) 4NF

**4. In a good database, the attributes whose values may frequently be null are avoided, because**

- (a) Can't be inserted in the primary key attributes and the attributes for which the NOT NULL constraint is specified

- (b) Waste storage space
- (c) Accounting for them while applying aggregate functions such as COUNT or AVG will lead to problems
- (d) All of the above

5. A relation can have more than one foreign key \_\_\_\_A\_\_\_\_. Give example

- (a) Yes
- (b) No

6. A foreign key must

- (a) Be numeric.
- (b) Match the field value of a primary key in a related table.
- (c) Be defined in all tables in the database.
- (d) Be unique.

7. What does referential integrity prevent?

- (a) Recursive joins
- (b) Loss of data from employee sabotage
- (c) Loss of data from any one corrupted table
- (d) Data redundancy

8. Which of the following is not true with primary key constraint?

- (a) Primary key attribute cannot have null value.
- (b) Primary key constraint enforces entity integrity.
- (c) Unique key constraint is preferable over primary key constraint.
- (d) Multiple primary key constraints can be defined on a relation.

9. The relationship between two entity types is known as ———.

- (a) Two-way relationship
- (b) Double relationship
- (c) Binary relationship

(d) Multiple relationships

**10. The strong entity type and weak entity type participates in ———.**

(a) One-to-one relationship

**(b) One-to-many relationship**

(c) Many-to-one relationship

(d) Many-to-many relationship

Part 2: SQL

**1. Which of these keywords is used to eliminate duplicate tuples from the resultant relation?**

(a) DELETE

**(b) DISTINCT**

(c) NON DUPLICATE

(d) EXISTS

**2. Which of these operators selects values that match any value in a given list of values?**

(a) BETWEEN

(b) LIKE

**(c) IN**

(d) None of these

**3. The operation that is used to retrieve those tuples present in one relation, and not present in another relation is\_\_\_\_\_.**

(a) UNION

(b) INTERSECT

**(c) MINUS**

(d) DIFFERENCE

**4. Which of these is the correct command for inserting values in the Employee relation?**

(a) INSERT VALUES ('John', 2000)INTO Employee;

(b) INSERT ('John', 2000)INTO Employee;

(c) INSERT INTO Employee VALUES ('John', 2000);

(d) None of these

**5. The command that can be used to make changes in the name of publishers in a relation PUBLISHER is \_\_\_\_\_.**

(a) UPDATE

(b) ALTER

(c) MODIFY

(d) None of these

**6. Consider the following SQL statements:**

(I) DELETE FROM Customer; (II) DROP TABLE Customer;

(a) Only statement (I) permanently removes the relation Customer

(b) Only statement (II) permanently removes the relation Customer

(c) Both statements (I) and (II) permanently remove the relation Customer

(d) None of these

**7. Which of the following will return the number of tuples in the Employee relation?**

(a) SELECT COLUMNS(\*) FROM Employee;

(b) SELECT COUNT()FROM Employee;

(c) SELECT COUNT(\*)FROM Employee;

(d) All of these

**8. Which of the following clauses is used to restrict groups returned by the GROUP BY clause?**

(a) DISTINCT

(b) WHERE

(c) EXISTS

(d) HAVING

### Part 3:

1.

```
Select avg(price) FROM publishers,books WHERE books.P_id=publishers.P_id;  
SELECT max(price) FROM publishers, books where books.P_id=publishers.P_id;  
SELECT min(price) FROM publishers, books WHERE books.P_id=publishers.P_id;
```

2. `SELECT*FROM BOOK WHERE Book_title LIKE 'D%';`

3. `SELECT*FROM books WHERE books.price =(select avg(price) from books);`

4. `SELECT * FROM authors where authors.city=(select city from authors where Aname="luis lan");`

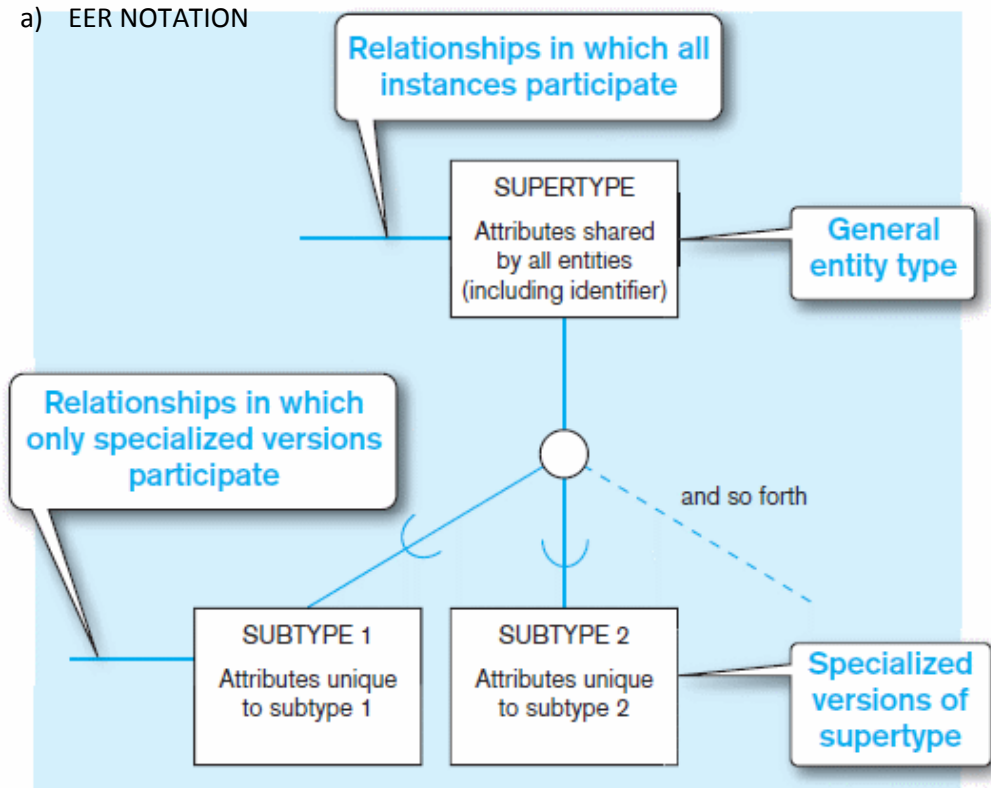
### Week 2

#### Super types and subtypes

- enhanced ER model : extends original ER model with new modeling constructs
- Subtype – a subgrouping of entities in an entity type that has attributes distinct from those in other subgroupings
- Super type: a generic entity type that has a relationship with one or more subtypes
- Attribute inheritance:
  - o Subtype entities inherit values of all attributes of the supertype
  - o An instance of a subtype is also an instance of the supertype

Basic notation for supertype/subtype notation

a) EER NOTATION



## DIAGRAMS WILL BE IN SLIDES.

### SLIDE 5

#### Relationships and subtypes

- Relationships at the **supertype level** indicate that all subtypes will participate in the relationship
- The instances of a **subtype** may participate in a relationship unique to that subtype. In this situation, the relationship is down at the subtype level.

### Slide 7

#### Generalization and specialization

- **Generalization:** the process of defining a more general entity type from a set of more specialized entity types. BOTTOM UP
- **Specialization:** the process of defining one or more subtypes of the supertype forming supertype/subtype relationships. TOP-DOWN

### SLIDE 9

### SLIDE 10

### Slide 11

### SLIDE 12

#### Constraints in supertype/SUBTYPE RELATIONSHIPS

- Completeness constraints: whether an instance of a supertype **must** also be a member of at least one subtype
  - o Total specialization rule :yes (double line) ( a patient has to be either an outpatient or a resident patient)
  - o Partial specialization rule : no (single line)( a vehicle can be a car or a truck but doesn't have to be either )

### Slide 14

### 15

#### Constraints in supertype/subtype relationships

- Disjointness constraints: whether an instance of a supertype may simultaneously be a member of two (or more) subtypes
  - o Disjoint rule: an instance of the supertype can be only one of the subtypes( a patient can be either an outpatient or a resident patient but not both at the same time)

- Overlap rule: an instance of the supertype could be more than one of the subtypes(a part may be both a manufactured part and a purchased part at the same time but must be one or the other due to total specialization)

17

18

### Constraints in supertype/subtype relationships

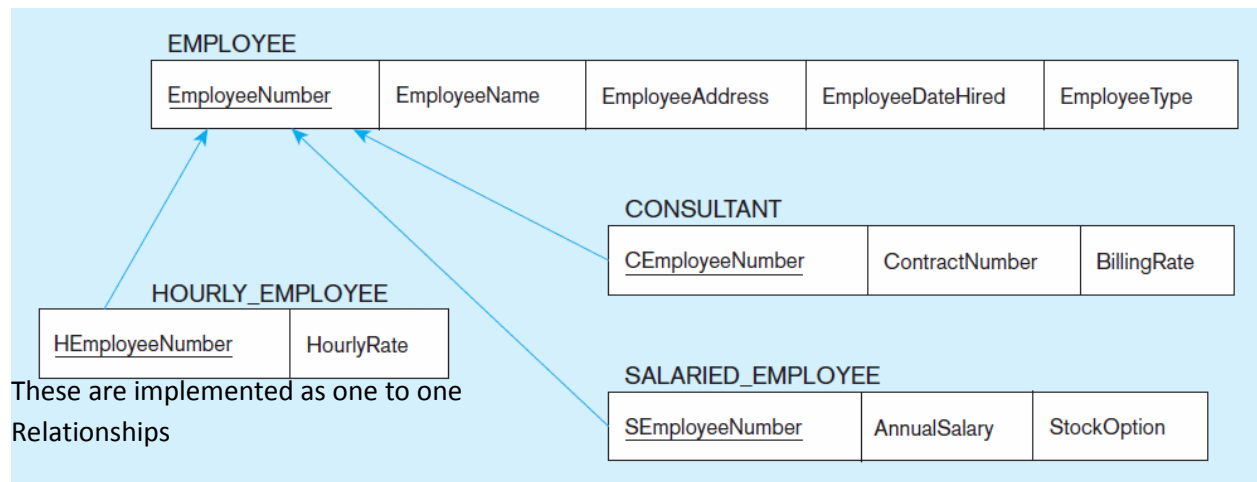
- Subtype discriminator: an attribute of the supertype whose values determine the target subtypes
  - Disjoint- a simple attribute with alternative values indicate the possible subtypes
  - Overlapping – a composite attribute whos subparts pertain to different subtype. Each subpart contains a Boolean value to indicate whether or not the instance belongs to the associated subtype

20

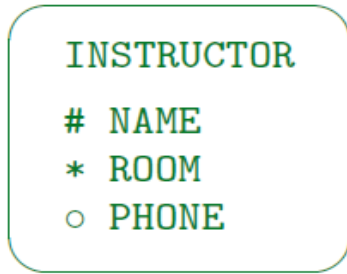
21

### Mapping supertype/subtype relationships

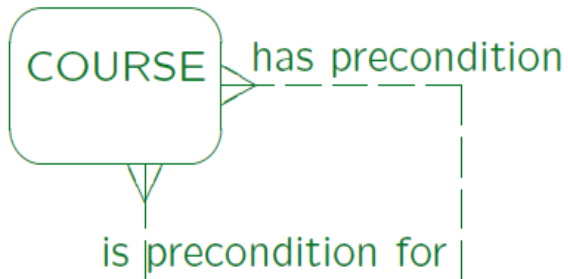
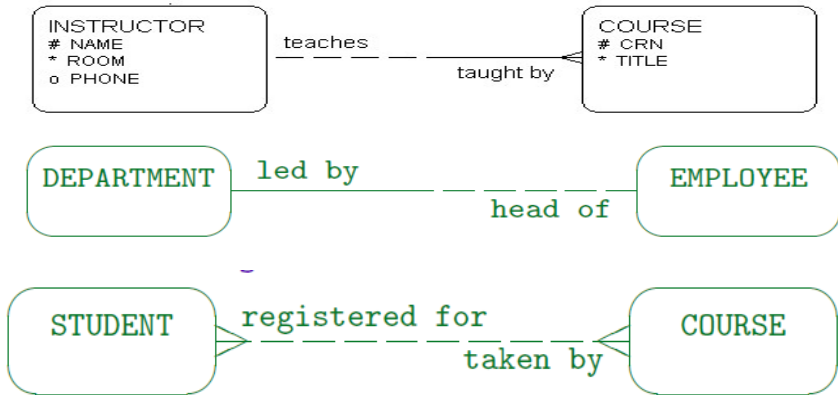
- One relation for supertype and for each subtype
- Supertype attributes (including identifier and subtype discriminator) go into supertype relation
- Subtype attributes go into each subtype;
- Primary key of a supertype relation also becomes primary key of a subtype relation
- 1:1 relationship established between supertype and each subtype, with supertype as primary table



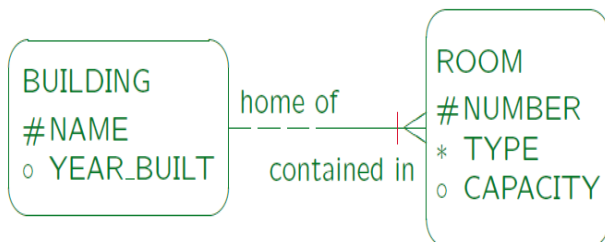
Oracle notations



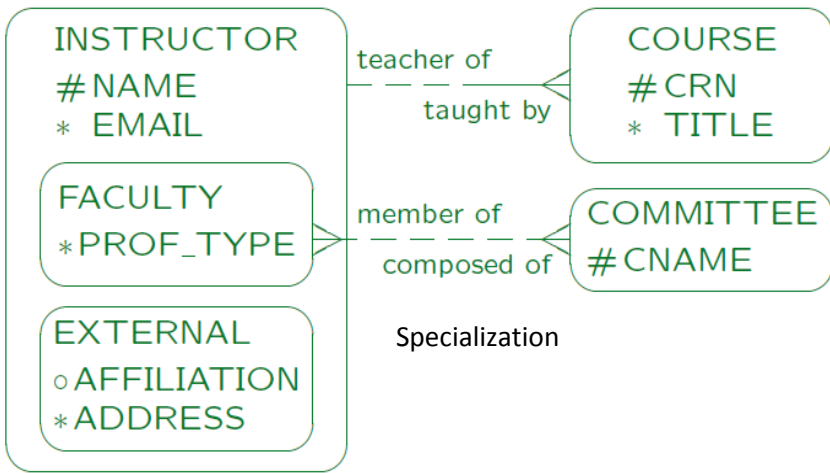
Entity and attributes



Unary/recursive relationship



Weak entity



### Week 3

#### Data integrity

- Domain integrity
- Entity integrity
- Referential integrity
- Semantic integrity

#### Entity integrity

- Each tuple in a relation is uniquely identified
- The set of attributes that participate in the primary key cannot have duplicate value or null value

#### Referential integrity

- The value of foreign key in the referencing relation must exist in the primary key attribute of the referenced relation or that the value of the foreign key is null
- The type of the foreign key in referencing relation must match the data type of the primary key of referenced relation.

#### CUSTOMER

<u>CustomerID</u>	CustomerName	CustomerAddress	CustomerCity	CustomerState	CustomerPostalCode
-------------------	--------------	-----------------	--------------	---------------	--------------------

#### ORDER

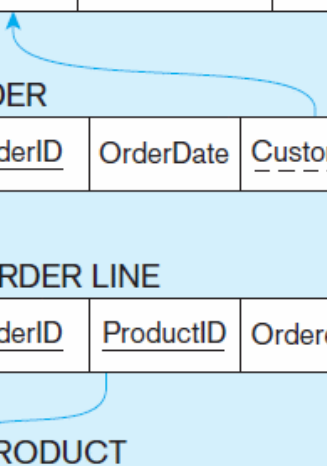
<u>OrderID</u>	OrderDate	<u>CustomerID</u>
----------------	-----------	-------------------

#### ORDER LINE

<u>OrderID</u>	<u>ProductID</u>	OrderedQuantity
----------------	------------------	-----------------

#### PRODUCT

<u>ProductID</u>	ProductDescription	ProductFinish	ProductStandardPrice	ProductLineID
------------------	--------------------	---------------	----------------------	---------------



### Semantic integrity

- Enforce the business rules and logical rules on database
- The data in the database is logically consistent and complete with respect to the real world

### Oracle 11g Database editions

#### Enterprise edition

- Premier edition
- Supports complex features (video, object database capabilities, full data warehouse cap)4
- Used by large organizations whose enterprise operations depend upon the database

#### Standard edition

- Formerly known as workgroup edition
- Multiuser edition
- Lacks sophisticated performance features
- Used by workgroup or dep level database applications

#### Standard one and express editions

- Standard one: limited to a single server environment
- Express: single user, limited feature, does not supposed manager interface

### **Slides 20-25**

### **Make relational and logical schemas to study**