Chapter 2C

Foreign Key

CSIS0278 / COMP3278

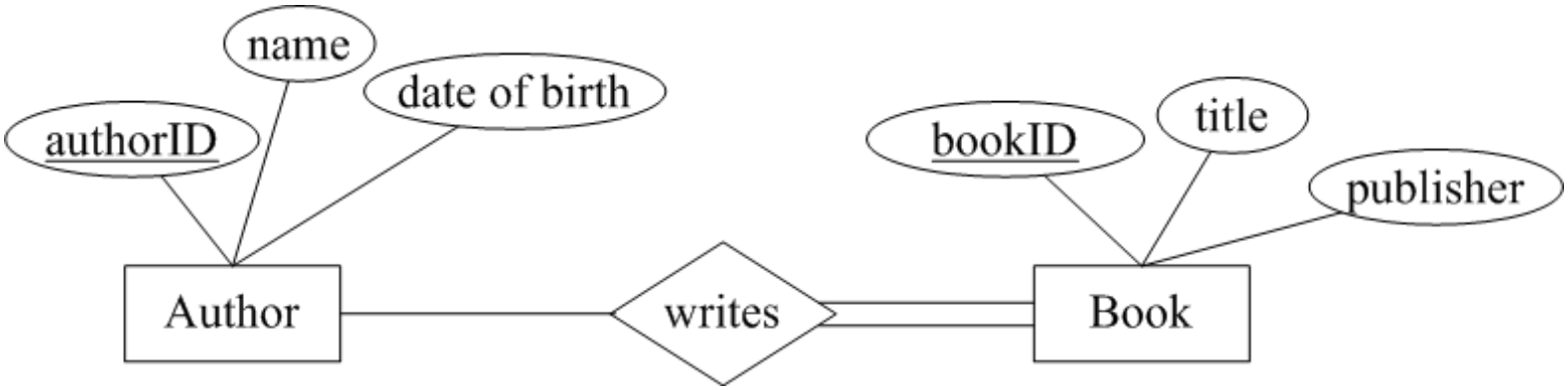
Introduction to

Database Management Systems

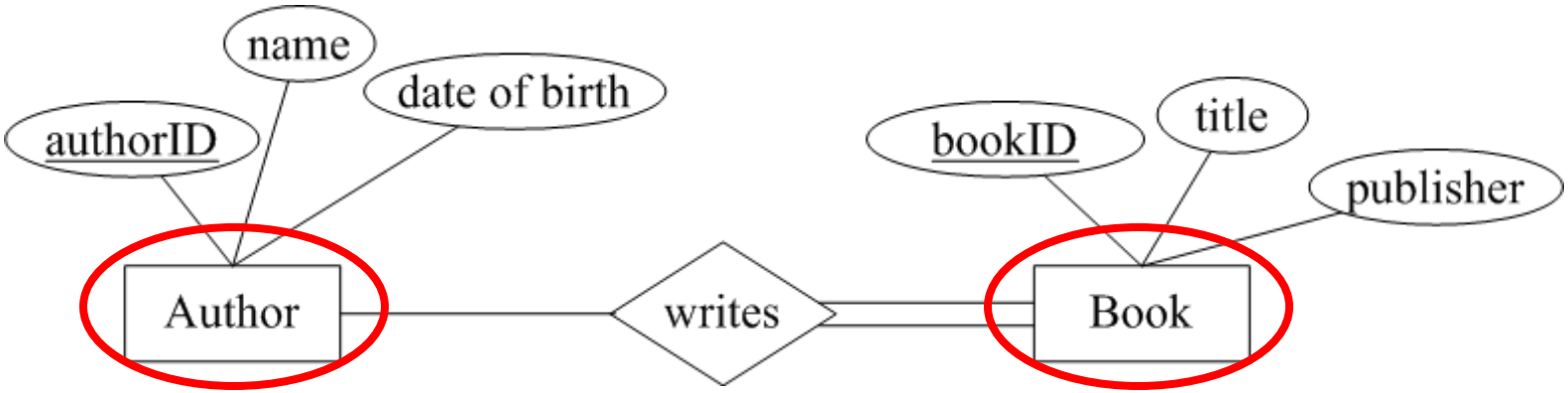
Department of Computer Science, The University of Hong Kong

Slides prepared by - **Dr. Chui Chun Kit** for students in CSIS0278/COMP3278
For other uses, please email : ckchui@cs.hku.hk

Example 1


name
date of birth
authorID
bookID
title
publisher
Author
writes
Book

Example 1



- Author (authorID, name, date of birth)
- Book (bookID, title, publisher)

To transform an ER model to relational tables…

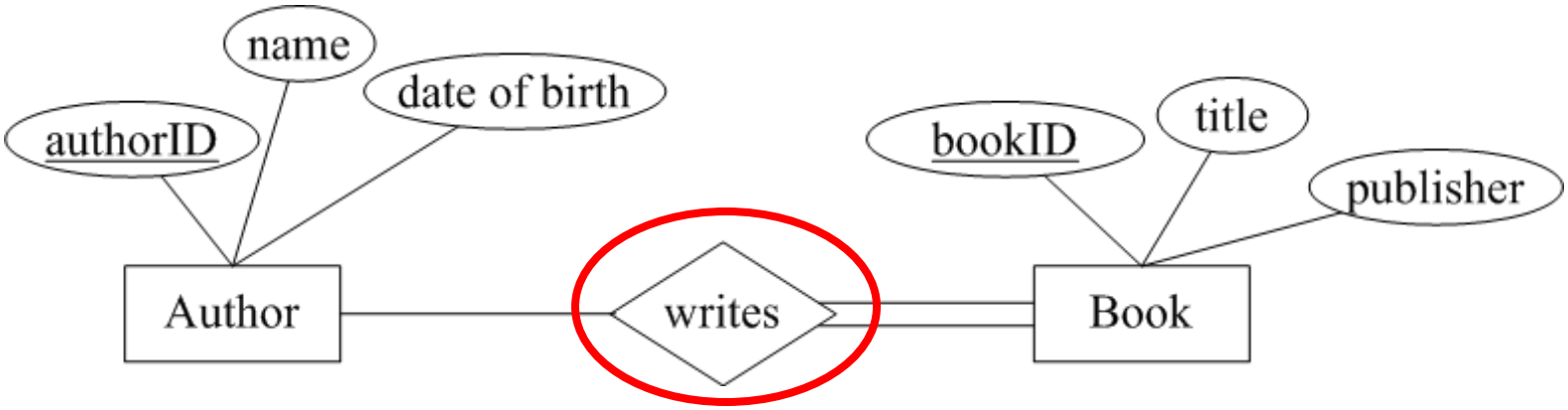
Step 1. Entity set -> table

Each **entity set** becomes a **table**.

Each **attribute** becomes a **column**.

Each **entity** is a **tuple** in the table.

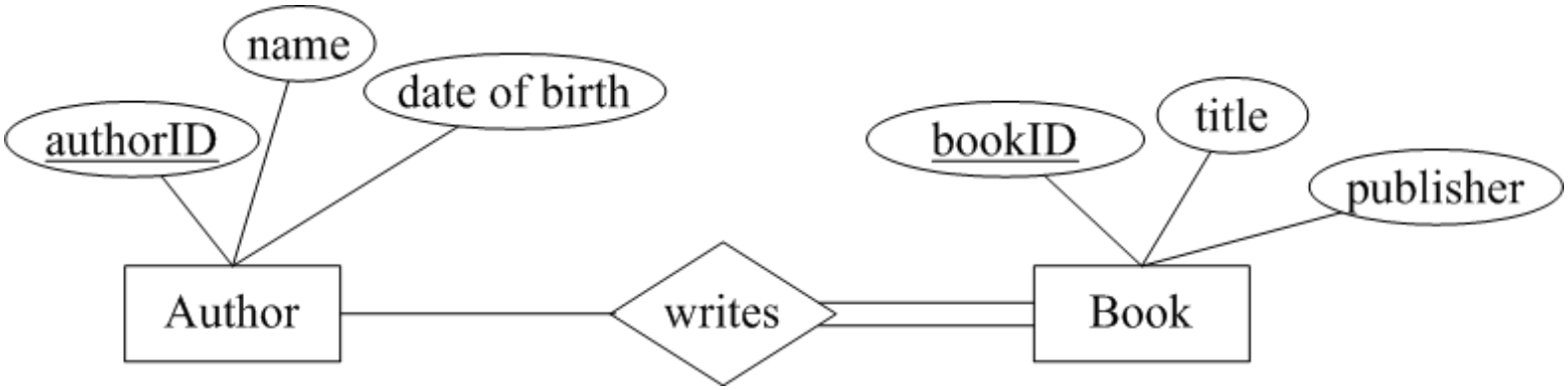
Example 1



- Author (authorID, name, date of birth)
- Book (bookID, title, publisher)
- Writes ( )

Step 2. Relationship set
Whether a relationship set becomes a table or not depends on the **mapping cardinality** of the relationship.
(many to many) , a table.

Example 1



- Author (authorID, name, date of birth)
- Book (bookID, title, publisher)
- Writes (**authorID, bookID**)

Step 3. Identify the key
What is the **primary key** of each table? Any **foreign keys**?

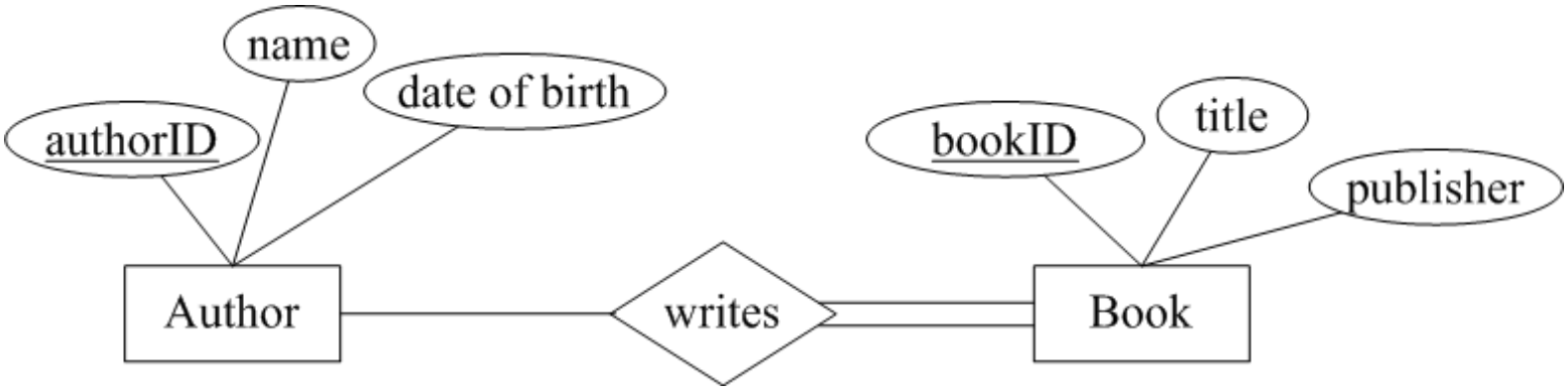
bookID is a **Foreign key** , this key is referencing the column bookID in the Book table.

authorID is another **Foreign key** , this key is referencing the column authorID in the Author table

Foreign key

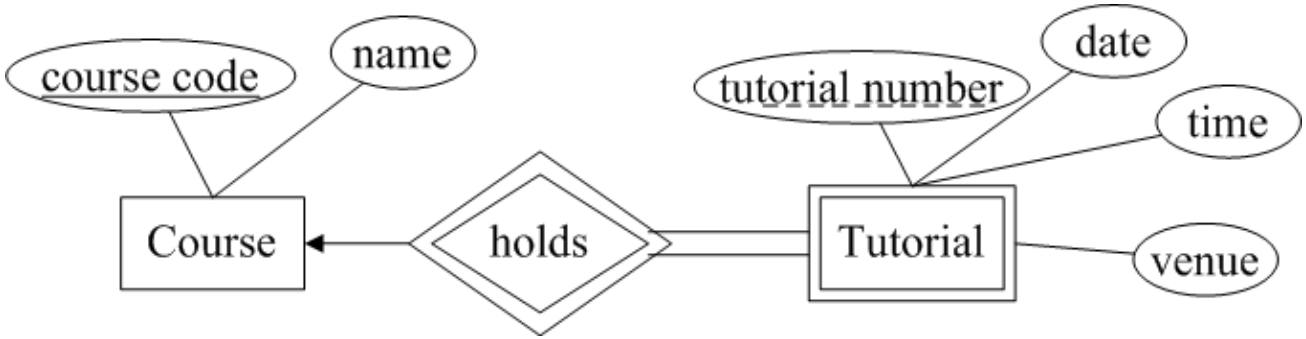
- A **foreign key** is a **referential constraint** between two tables.
- A foreign key is a field in a relational table that matches a candidate key of another table.
- **The foreign key can be used to cross-reference tables.**
  - It is used to link information together.
  - An essential part of ***database normalization*** (To be discussed in Chapter 5).

Example 1

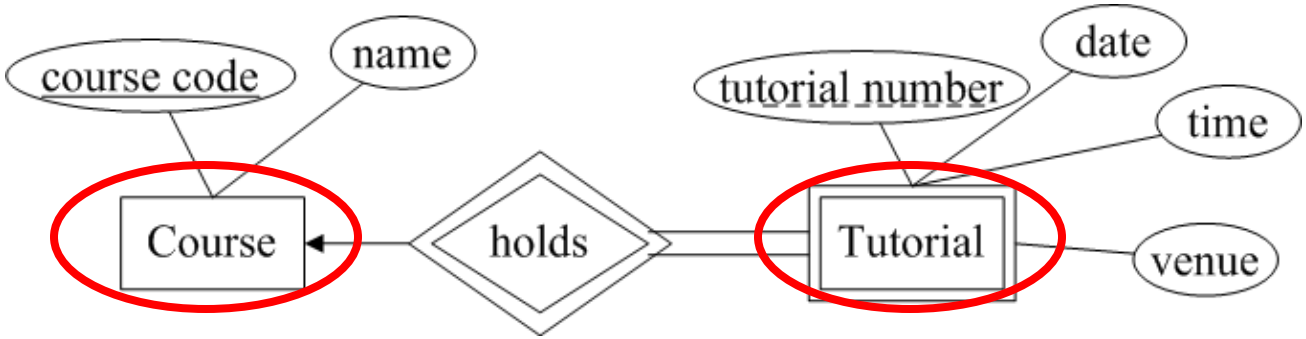


- Author (authorID, name, date of birth)
  - Foreign key: none
- Book (bookID, title, publisher)
  - Foreign key: none
- Writes (authorID, bookID)
  - Foreign keys: {authorID} referencing Author
    {bookID} referencing Book

Example 2


course code
name
tutorial number
date
time
Course
holds
Tutorial
venue

Example 2



- Course (course code, name)
- Tutorial (tutorial number, date, time, venue)

To transform an ER model to relational tables…

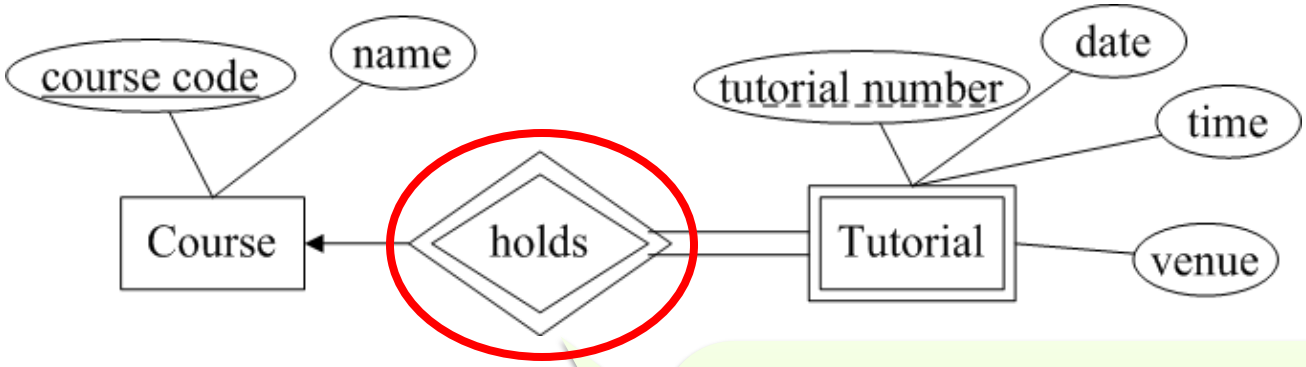
Step 1. Entity set -> table

Each **entity set** becomes a **table**.

Each **attribute** becomes a **column**.

Each **entity** is a **tuple** in the table.

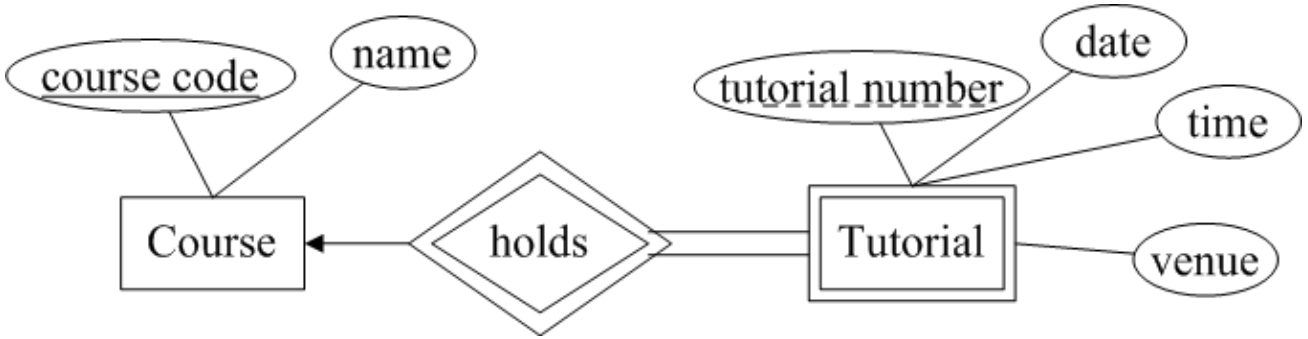
Example 2



Step 2. Relationship set
Whether a relationship set becomes a table or not depends on the mapping cardinality of the relationship.
(**one to many** or **many to one**), attributes go to **"many" side.**

- Course (course code, name)
- Tutorial (tutorial number, date, time, venue, **course_code**)

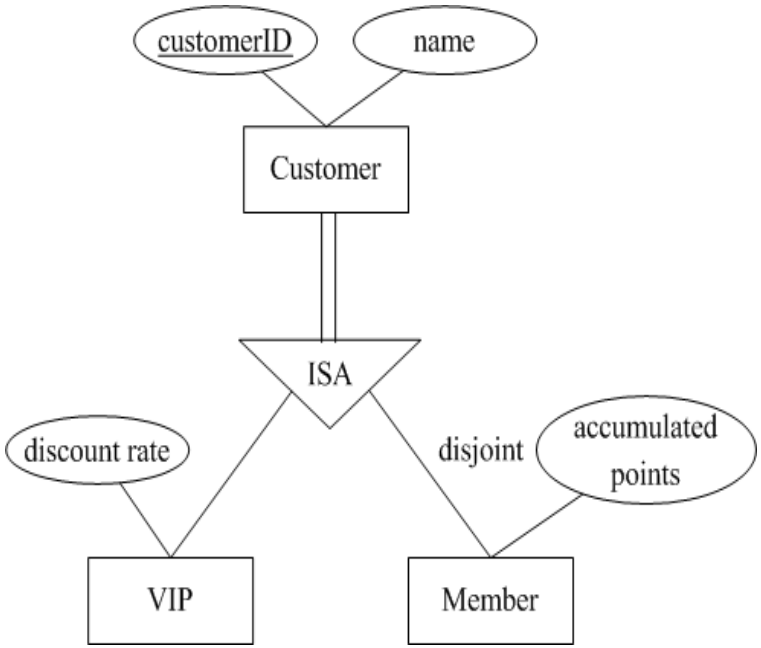
Example 2



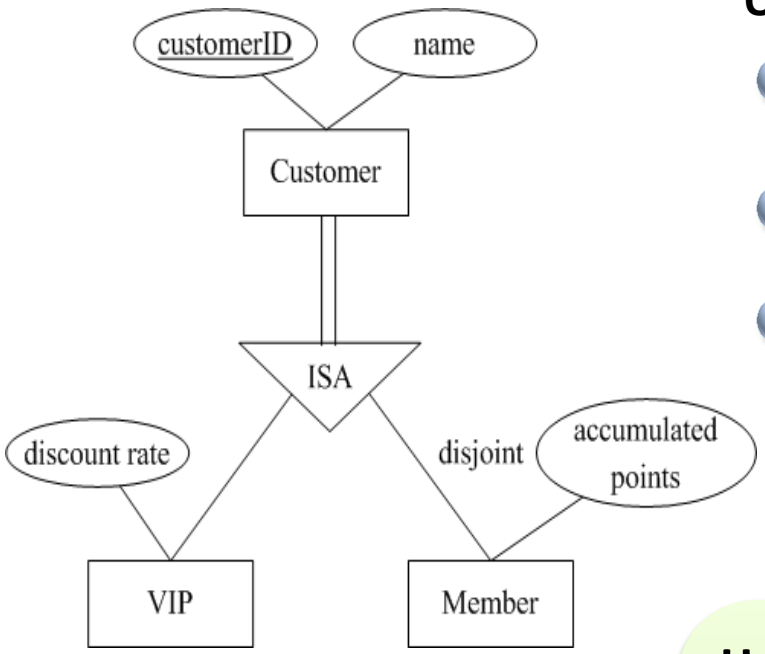
Step 3. Identify the key
What is the **primary key** of each table? Any **foreign keys**?

- Course (course code, name)
  - Foreign key: none
- Tutorial (tutorial number, date, time, venue, course code)
  - Foreign key: {course code} referencing Course

Example 3


customerID
name
Customer
ISA
discount rate
disjoint
accumulated points
VIP
Member

Example 3



Option 1

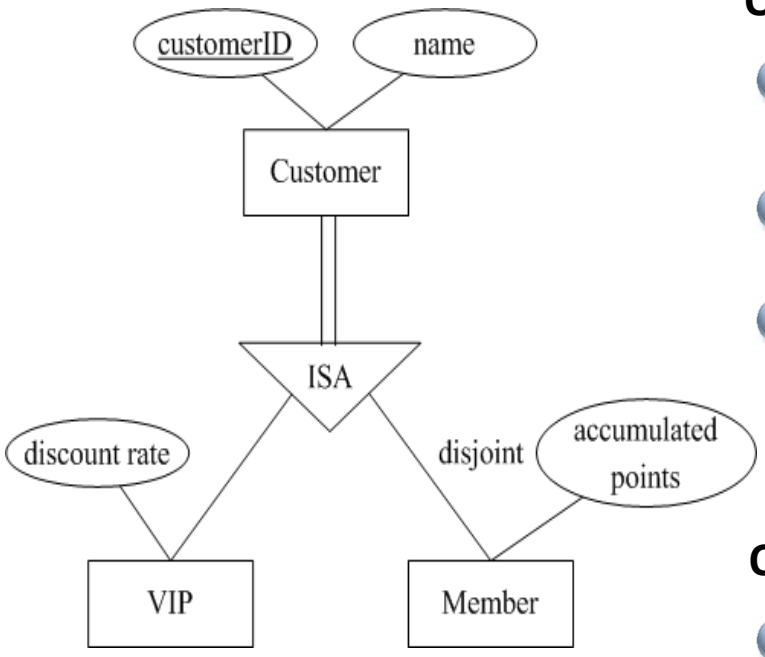
- Customer (<u>customerID</u>, name)
  Foreign key: none
- VIP (<u>customerID</u>, discount_rate)
  Foreign key: {customerID} referencing Customer
- Member (<u>customerID</u>, accumulated_points)
  Foreign key: {customerID} referencing Customer

Handling ISA relationship

Option 1 :

Form a table for higher-level entity set.
Form a table for each lower-level entity set, which contains the primary key of the higher-level entity set and local attributes.

Example 3



Option 1

- Customer (<u>customerID</u>, name)
  Foreign key: none
- VIP (<u>customerID</u>, discount_rate)
  Foreign key: {customerID} referencing Customer
- Member (<u>customerID</u>, accumulated_points)
  Foreign key: {customerID} referencing Customer

Option 2

- Customer (<u>customerID</u>, name)
  Foreign key: none
- VIP (<u>customerID</u>, name, discount_rate)
  Foreign key: {customerID} referencing Customer
- Member (<u>customerID</u>, name, accumulated_points)
  Foreign key: {customerID} referencing Customer

Handling ISA relationship

Option 2 :

Form a table for each entity set with all local and inherited attributes

Example 3

[Storage] Option 1 has less storage redundancy.
[Efficiency] Accessing data (e.g, retrieving the name and discount_rate of a VIP) in option 1 requires accessing two tables (not as efficient as option 2, which requires accessing one table only)!

What are the **advantage** and **disadvantage** of these options?

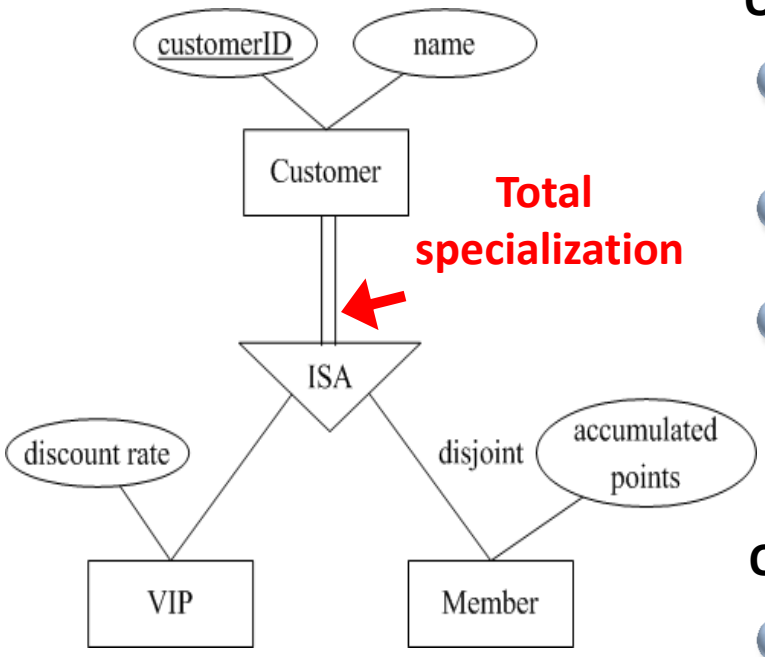
Option 1

- Customer (<u>customerID</u>, name)
  Foreign key: none
- VIP (<u>customerID</u>, discount_rate)
  Foreign key: {customerID} referencing Customer
- Member (<u>customerID</u>, accumulated_points)
  Foreign key: {customerID} referencing Customer

Option 2

- Customer (<u>customerID</u>, name)
  Foreign key: none
- VIP (<u>customerID</u>, name, discount_rate)
  Foreign key: {customerID} referencing Customer
- Member (<u>customerID</u>, name, accumulated_points)
  Foreign key: {customerID} referencing Customer

Example 3



Option 1

- Customer (customerID, name)
  Foreign key: none
- VIP (customerID, discount_rate)
  Foreign key: {customerID} referencing Customer
- Member (customerID, accumulated_points)
  Foreign key: {customerID} referencing Customer

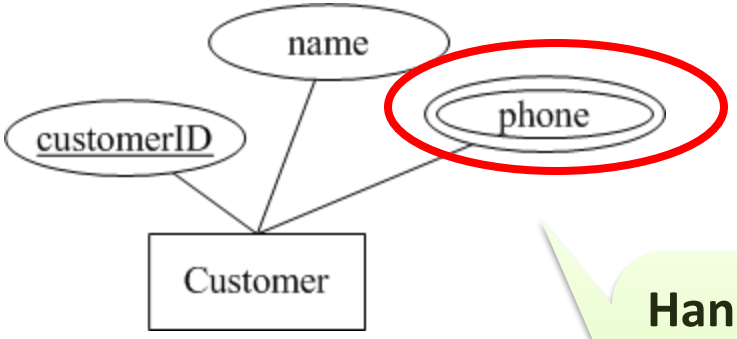
Option 2

- Customer (customerID, name)
  Foreign key: none
- VIP (customerID, name, discount_rate)
  Foreign key: {customerID} referencing Customer
- Member (customerID, name, accumulated_points)
  Foreign key: {customerID} referencing Customer

Option 3

- VIP (customerID, name, discount rate)
  Foreign key: none
- Member (customerID, name, accumulated points)
  Foreign key: none

Example 4



Handling Multivalue attributes
Multivalue attribute becomes a table.

- Customer (customerID, name)
  - Foreign key: none
- CustomerPhone (customerID, phone)
  - Foreign key: {customerID} referencing Customer

Chapter 2C

END

CSIS0278 / COMP3278

Introduction to

Database Management Systems

Slides prepared by - **Dr. Chui Chun Kit** for students in CSIS0278/COMP3278
For other uses, please email : ckchui@cs.hku.hk