



# Database

## Lesson 7. Entity Relationship Model

Viet-Trung Tran

<https://is.hust.edu.vn/~trungtv/>

# Learning Map

Sequence	Title
1	Introduction to Databases
2	Relational Databases
3	Relational Algebra
4	Structured Query Language – Part 1
5	Structured Query Language – Part 2
6	Constraints and Triggers
7	Entity Relationship Model
8	Functional Dependency
9	Normalization
10	Storage - Indexing
11	Query Processing
12	Transaction Management – Part 1
13	Transaction Management – Part 2

# Outline

- Introduction
- How to create an ERD
- Mapping from ERD to relational schema

# Objectives

- Upon completion of this lesson, students will be able to:
  - Know what the **entity relationship model** is
  - Know how to create an **ERD** from a real-world problem
  - Transform from ERD into **relational schema**

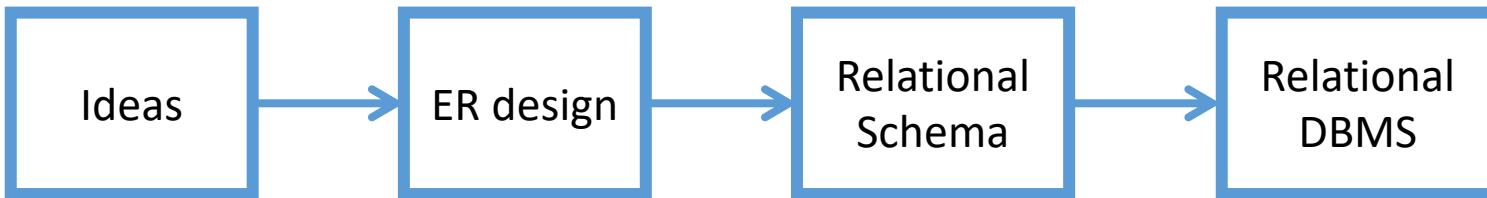
# 1. Introduction

---

- Introduction
- Entity sets
- Attributes
- Key
- Relationships

# 1.1. Introduction

- Two approaches to DB designing
  - Top - down: Entity Relationship model (ER)
  - Bottom – up: Functional Dependencies and Normalization
- ER model is used in DB design



## 1.1. Introduction

- The structure of data is represented graphically: ERD
- Three principal element types:
  - Entity sets
  - Attributes
  - Relationships

## 1.2. Entity sets

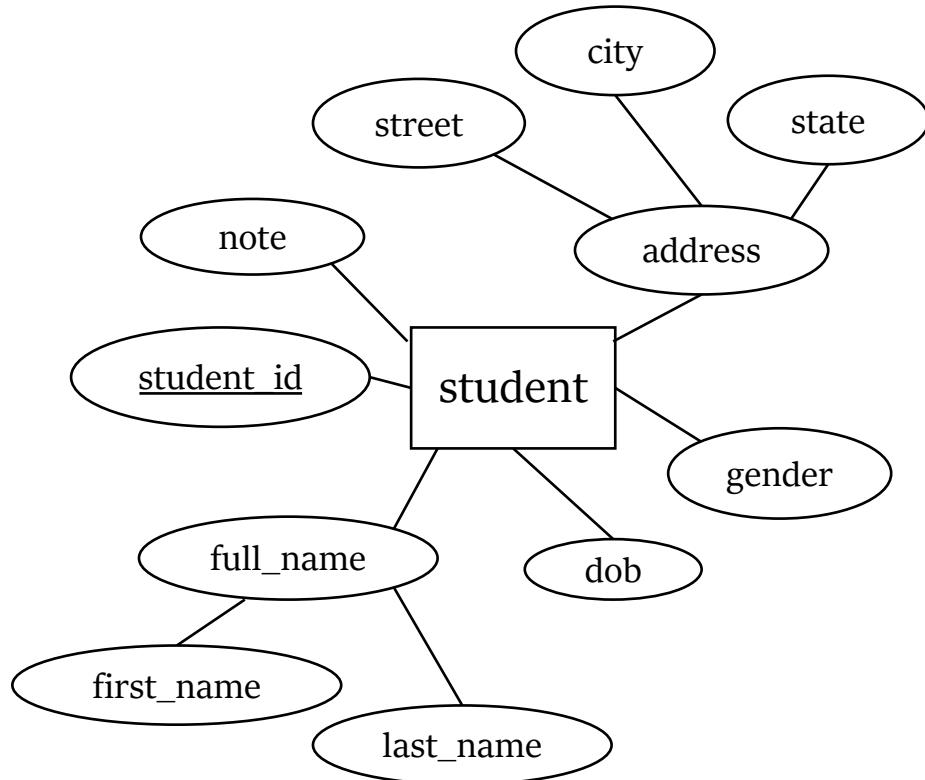
- Entity
  - is a thing in the real world with an independent existence.
  - An entity may be an object with a physical existence (a particular person, car, house, or employee) or it may be an object with a conceptual existence (a company, a job, or a university course).
- Entity sets
  - a collection of similar entities forms an entity set.
- Weak entity type vs. strong entity type
  - Weak entity type do not have key attributes of their own while strong entity type do have a key attribute
  - a weak entity can not be identified without an owner entity.
- In ERD, rectangular boxes represent for entity sets

## 1.3. Attributes

- Attributes
  - Entity sets have associated attributes, which are **properties of the entities** in that set.
  - For instance, each entity "student" has some properties such as student\_id, first\_name, last\_name, dob, gender, address, and so on.
  - In ERD, ovals represent for attributes
- Value domain of an attribute
  - Each simple attribute of an entity type is associated with a value set (or domain of values).
  - For example: domain(gender) = {male, female}; domain(dob) = {date}; domain(last\_name) = {char(30)}.

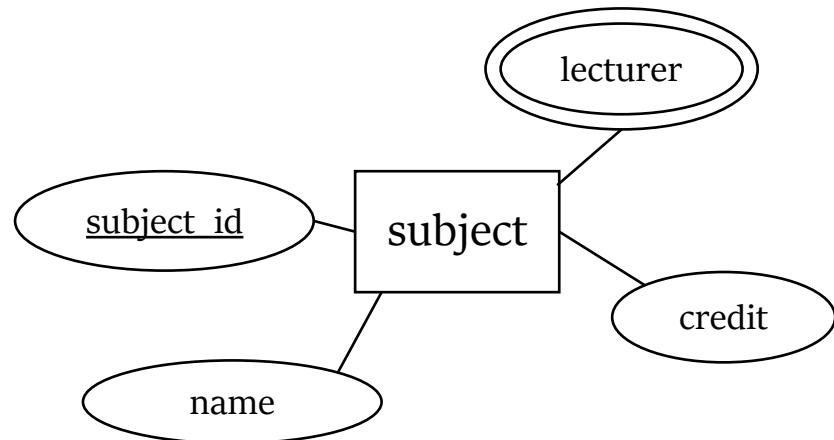
## 1.3. Attributes

- Some types of attributes
  - **Simple/atomic attributes:** Attributes that are not divisible.
  - **Composite attributes:** attributes can be divided into smaller subparts, which represent more basic attributes with independent meanings.



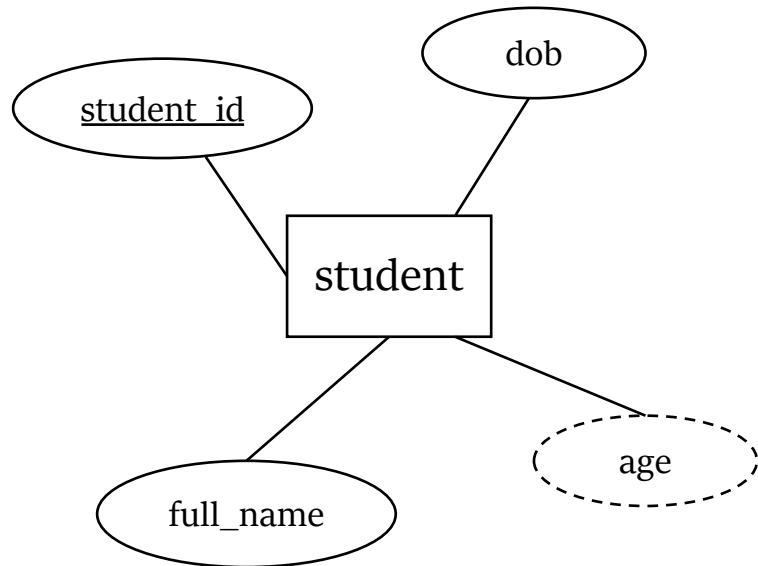
## 1.3. Attributes

- Some types of attributes
  - **Single-valued attributes**: have a single value for a particular entity
  - **Multi-valued attributes**: can have different numbers of values



## 1.3. Attributes

- Some types of attributes
  - **Stored attributes vs. Derived attributes:** age attribute is called a derived attribute and is said to be derivable from the dob attribute, which is called a stored attribute.



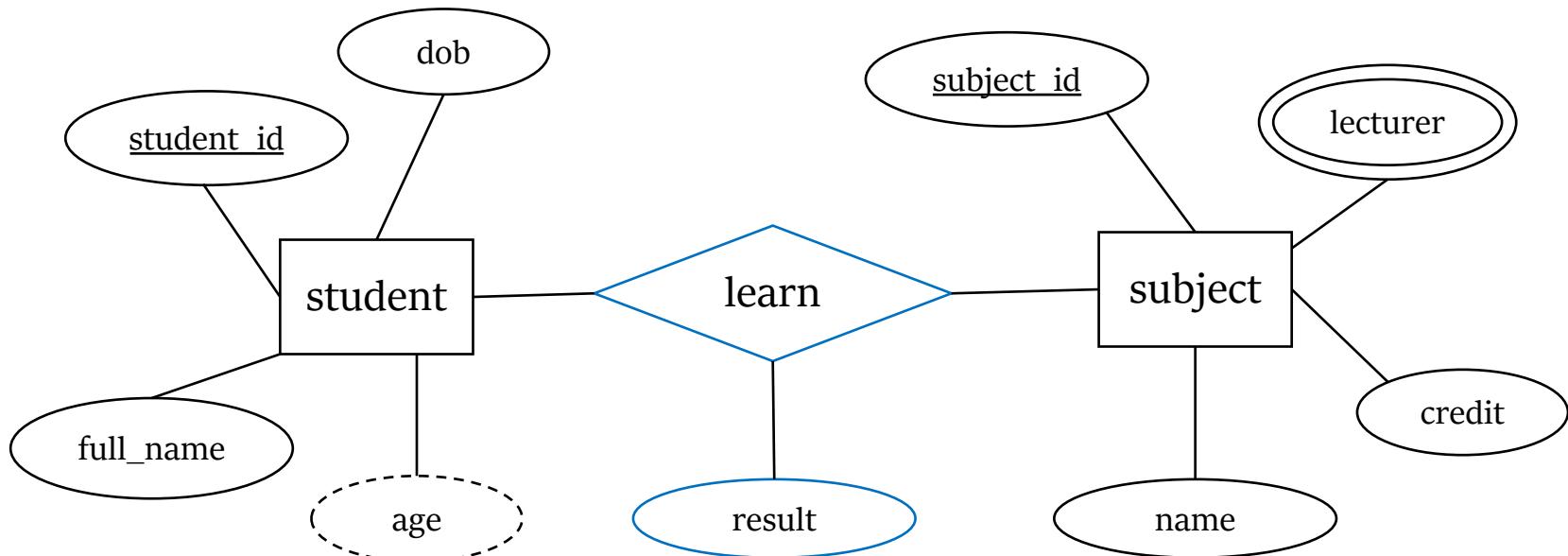
## 1.4. Key

- One or more attributes whose values are distinct for each individual entity in the entity set. Such an attribute is called **a key attribute**, and its values can be used to identify each entity uniquely.
- Each entity can have some keys. We choose one of them to be **primary key**.
- In ER diagrammatic notation, each key attribute has its name **underlined** inside the oval.

## 1.5. Relationships

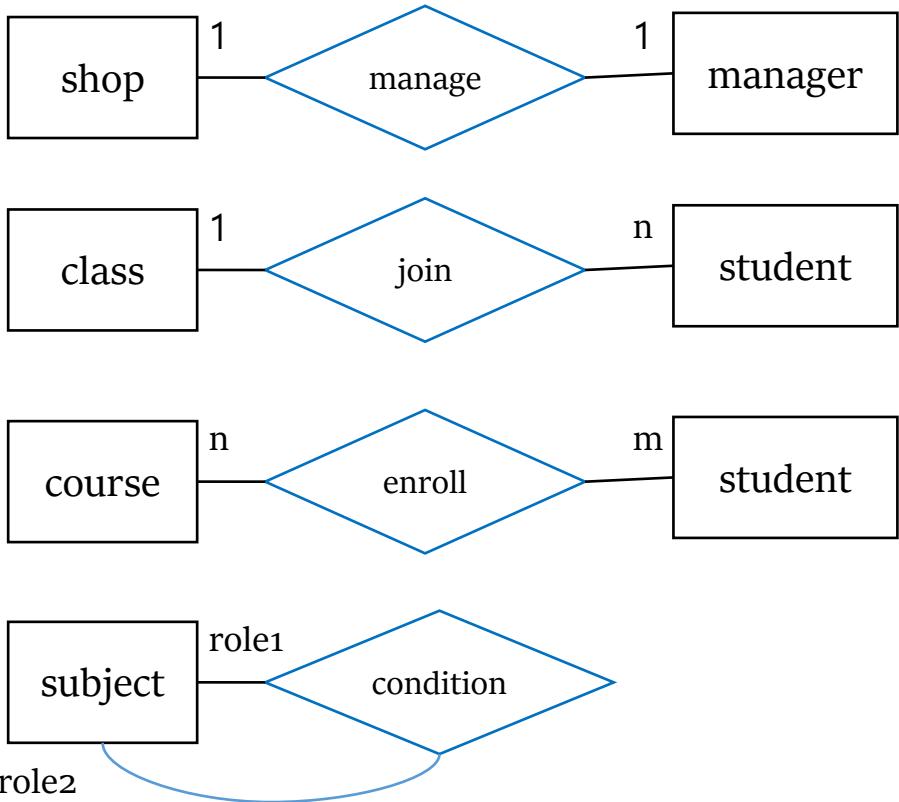
- Relationships are connections among two or more entity sets.
- In ER diagrams, relationship types are displayed as diamond-shaped boxes,
  - which are connected by straight lines to the rectangular boxes representing the participating entity types.
  - The relationship name is displayed in the diamond-shaped box.

## 1.5. Relationships



## 1.5. Relationships

- Type of relationships
  - 1 – 1
  - 1 – n
  - n – m
  - recursive



## 2. How to create an ERD

- Steps
- An example

## 2.1. Steps

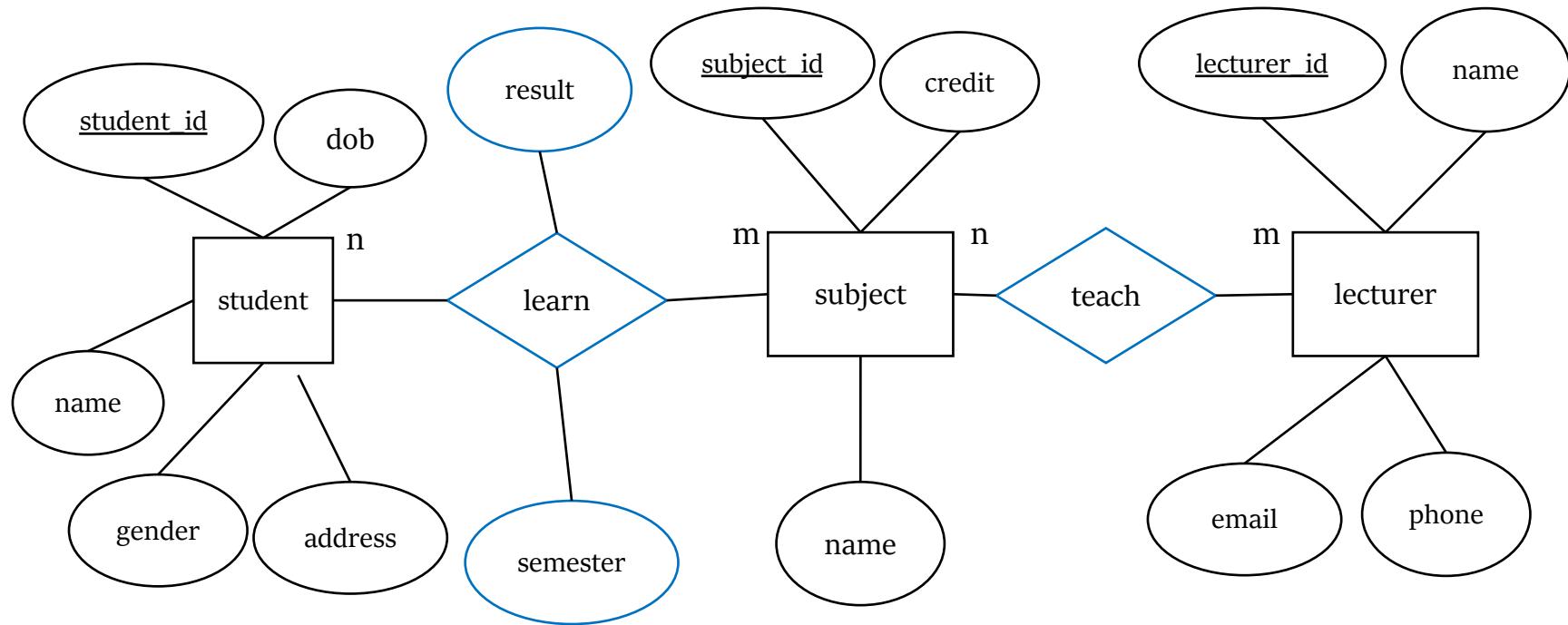
- Step 1: Identify all entity sets
  - Notice concepts, nouns
- Step 2: Identify all relationships among entity sets
  - Notice verbs
  - Type and degree of relationships

## 2.2. An example

- Read carefully the following scenario:
  - The information about **students** includes student identification (uniquely identify each student), name, gender, date of birth and address.
  - During the education time at school, students must study a lot of subjects. A subject can be learnt by students. A **subject** should be contained information such as subject identification, name and credit.
  - A lecturer can teach some subjects, and a subject can be taught by a group of lecturers. The information about **lecturers** should include lecturer identification, name, phone, email.
  - Students learn subjects at some semester, and their results should be stored.

## 2.2. An example

- We can draw this ER diagram

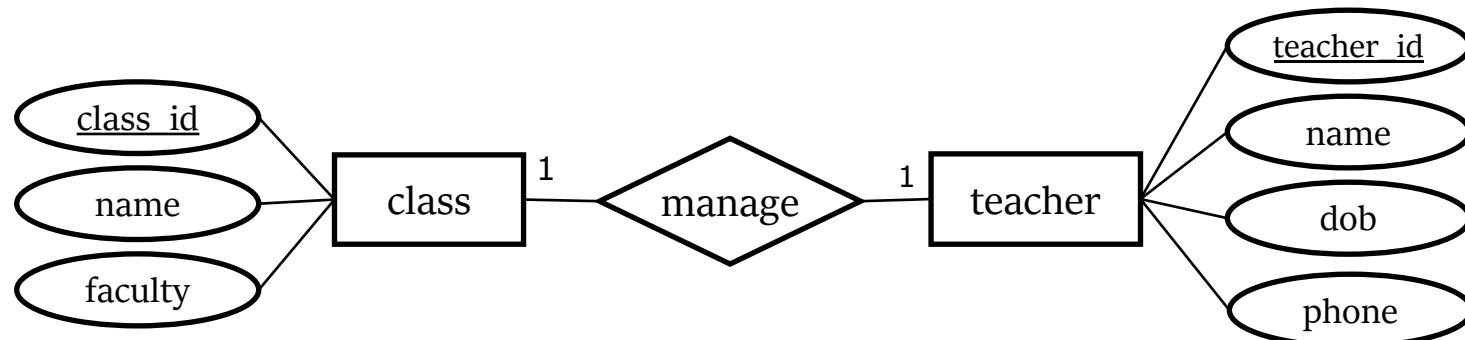


### 3. Mapping from ERD to relational schema

- Steps
- An example

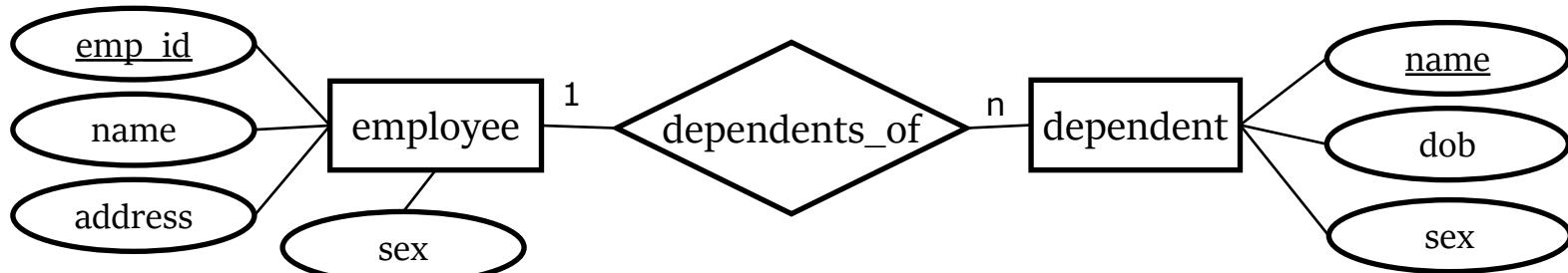
## 3.1. Steps

- Mapping of strong entity sets
  - For each entity set, create a relation that includes all the simple attributes of that entity set.
  - PK of entity set becomes PK of the relation
    - class(class\_id, name, faculty)
    - teacher(teacher\_id, name, dob, phone)



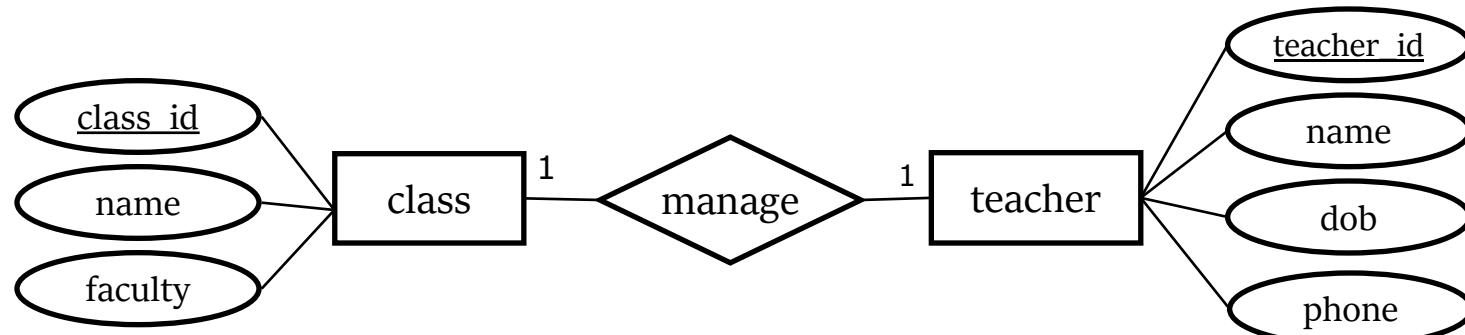
## 3.1. Steps

- Mapping of weak entity sets
  - For each entity set, create a relation that includes all the simple attributes of that entity set.
  - PK of strong entity set should be included in PK of the relation  
dependent(emp\_id, name, dob, sex)



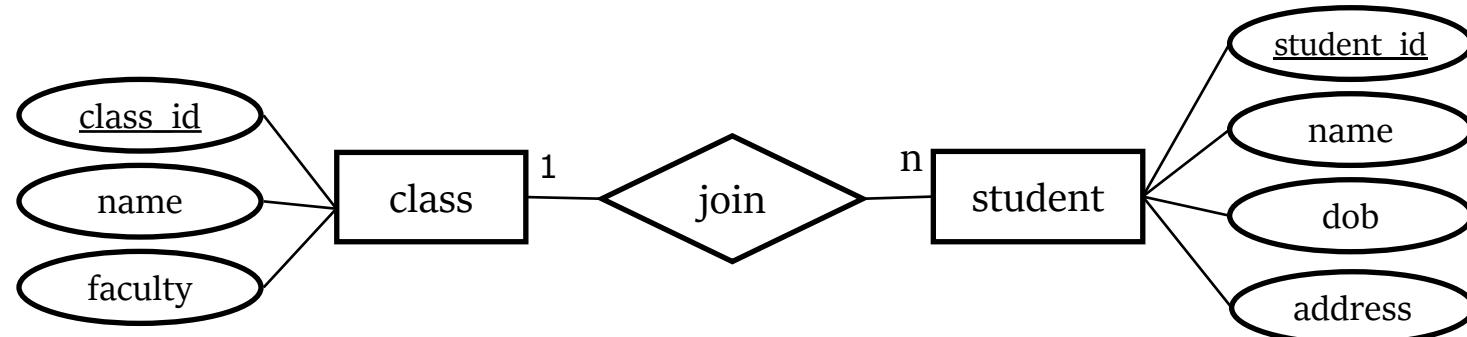
### 3.1. Steps

- Mapping of 1 - 1 relationships
  - Create a new relation which has all prime-attributes of both entity sets  
 $\text{manage}(\text{class\_id}, \text{teacher\_id})$
  - Use foreign key  
 $\text{class}(\text{class\_id}, \text{name}, \text{faculty}, \text{teacher\_id})$



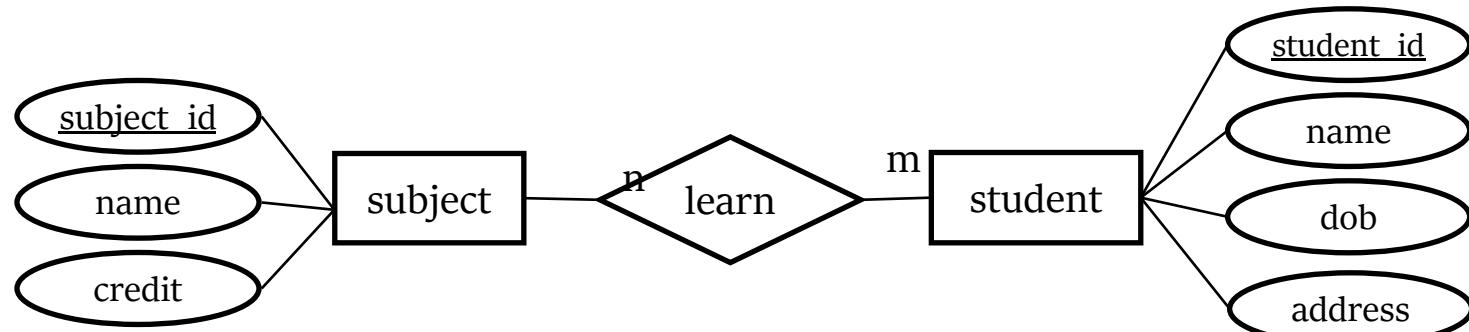
### 3.1. Steps

- Mapping of 1 - n relationships
  - Create a new relation which has all prime-attributes of both entity sets  
 $\text{join(class\_id, student\_id)}$
  - Use foreign key  
 $\text{student(student\_id, name, dob, address, class\_id)}$



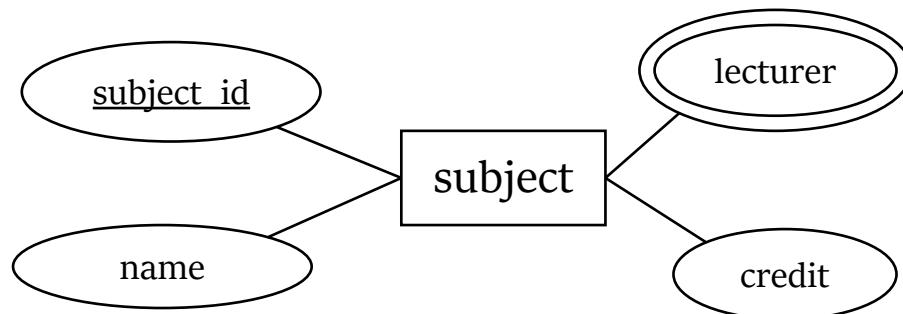
## 3.1. Steps

- Mapping of n - m relationships
  - Create a new relation which has all prime-attributes of both entity sets  
learn(subject\_id, student\_id)



## 3.1. Steps

- Mapping of multivalued attributes
  - For each multivalued attribute A, create a new relation R including an attribute corresponding to A, plus the primary key attribute K (as a foreign key in R) of the corresponding entity set
  - The primary key of R is the combination of A and K.
    - `subject_lecturer(subject_id, lecturer)`



## 3.2. An example

- student(student\_id, name, gender, dob, address)
- subject(subject\_id, name, credit)
- lecturer(lecturer\_id, name, phone, email)
- learn(student\_id, subject\_id, semester, result)
- teach(lecturer\_id, subject\_id)

# Remark

- ERD: an approach to DB designing
- Entity sets, attributes, key, relationships
- How to create an ERD
- Mapping from ERD to relational schema

# Quiz 1.

Quiz Number	1	Quiz Type	OX	Example Select
Question	How many kinds of relationship have we just studied?			
Example	<ul style="list-style-type: none"><li>A. 1</li><li>B. 2</li><li>C. 3</li><li>D. 4</li></ul>			
Answer	D			
Feedback	1-1, 1-n, n-m, recursive			

## Quiz 2.

Quiz Number	2	Quiz Type	OX	Example Select
Question	What is the type of attributes denoted by double ovals in ERD?			
Example	<ul style="list-style-type: none"><li>A. Multivalued attributes</li><li>B. Atomic attributes</li><li>C. Composite attributes</li><li>D. Derived attributes</li></ul>			
Answer	A			
Feedback				

# Summary

- Introduction
  - ERD in DB designing, and its components
- How to create an ERD
  - discover entity sets, attributes and relationships among entity sets
- Mapping from ERD to relational schema
  - transform from ERD into a set of tables



## Next lesson: Functional Dependency

- Hector Garcia-Molina, Jeffrey D. Ullman, Jennifer Widom. Database Systems: The Complete Book. Pearson Prentice Hall. the 2nd edition. 2008: Chapter 7
- Nguyen Kim Anh, Nguyên lý các hệ cơ sở dữ liệu, NXB Giáo dục. 2004: Chương 7