



HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG



BÀI GIẢNG MÔN  
CƠ SỞ DỮ LIỆU

Giảng viên:

Email:

TS. Nguyễn Thị Thu Hiền

[csdl.nth@gmail.com](mailto:csdl.nth@gmail.com)

1. Tổng quan về mô hình dữ liệu
2. Mô hình phân cấp
3. Mô hình mạng
4. Mô hình hướng đối tượng
5. Mô hình thực thể kết hợp
6. Mô hình quan hệ
7. So sánh và đánh giá

# 1. Tổng quan về mô hình dữ liệu

- Mô hình dữ liệu [Codd, 1980] gồm:
  - (1) Một tập hợp các kiểu cấu trúc dữ liệu (các khối xây dựng của bất kỳ CSDL nào phù hợp với mô hình);
  - (2) Một tập hợp các toán tử hoặc quy tắc suy luận, có thể được áp dụng cho mọi trường hợp hợp lệ của các loại dữ liệu được liệt kê (1), để truy xuất hoặc lấy dữ liệu từ bất kỳ phần nào của các cấu trúc đó theo bất kỳ kết hợp nào mong muốn;
  - (3) Một tập hợp các quy tắc toàn vẹn chung, xác định ngầm định hoặc rõ ràng tập các trạng thái CSDL nhất quán hoặc các thay đổi trạng thái hoặc cả hai - các quy tắc này đôi khi có thể được biểu thị dưới dạng quy tắc chèn-cập nhật-xóa.

DATA MODELS in DATABASE MANAGEMENT

E. F. Codd  
IBM Research Laboratory  
San Jose, California 95193

## 1 WHAT IS A DATA MODEL?

It is a combination of three components:

- 1) a collection of data structure types (the building blocks of any database that conforms to the model);
- 2) a collection of operators or inferencing rules, which can be applied to any valid instances of the data types listed in (1), to retrieve or derive data from any parts of those structures in any combinations desired;
- 3) a collection of general integrity rules, which implicitly or explicitly define the set of consistent database states or changes of state or both -- these rules may sometimes be expressed as insert-update-delete rules.

[[https://courses.washington.edu/geog482/resource/Codd\\_1981\\_Data\\_Model.pdf](https://courses.washington.edu/geog482/resource/Codd_1981_Data_Model.pdf)]

# 1. Tổng quan về mô hình dữ liệu

- Mô hình dữ liệu là mô hình hóa mô tả dữ liệu, ngữ nghĩa dữ liệu và các ràng buộc nhất quán của dữ liệu. Nó cung cấp các công cụ khái niệm để mô tả thiết kế cơ sở dữ liệu ở mỗi cấp độ trừu tượng hóa dữ liệu. [<https://www.javatpoint.com/data-models>]
- Hay mô hình dữ liệu là một tập hợp các khái niệm dùng để mô tả:
  - Dữ liệu
  - Ngữ nghĩa của dữ liệu
  - Các mối quan hệ trong dữ liệu
  - Các ràng buộc dữ liệu

# 1. Tổng quan về mô hình dữ liệu

❖ Nhận xét:

Có nhiều định nghĩa về mô hình dữ liệu, nhưng có hai quan điểm chính.

- Định nghĩa toàn diện nhất về mô hình dữ liệu đến từ Edgar Codd (1980): Mô hình dữ liệu bao gồm ba thành phần: 1) cấu trúc dữ liệu, 2) các thao tác trên cấu trúc dữ liệu và 3) các ràng buộc toàn vẹn cho các hoạt động và cấu trúc.
- Quan điểm thứ hai nảy sinh từ James Martin (1976) và những người khác đã phát triển ý tưởng về **sơ đồ cấu trúc dữ liệu** (cơ sở).
  - Phù hợp với thành phần đầu tiên được Codd (1980) trình bày rõ ràng, và là phiên bản phổ biến hơn vì tính đơn giản.
  - Tuy nhiên, thành phần thứ hai và thứ ba của Codd (1980) là cần thiết để hiểu đầy đủ về cách thức dữ liệu và nguồn gốc dữ liệu trở thành thông tin.

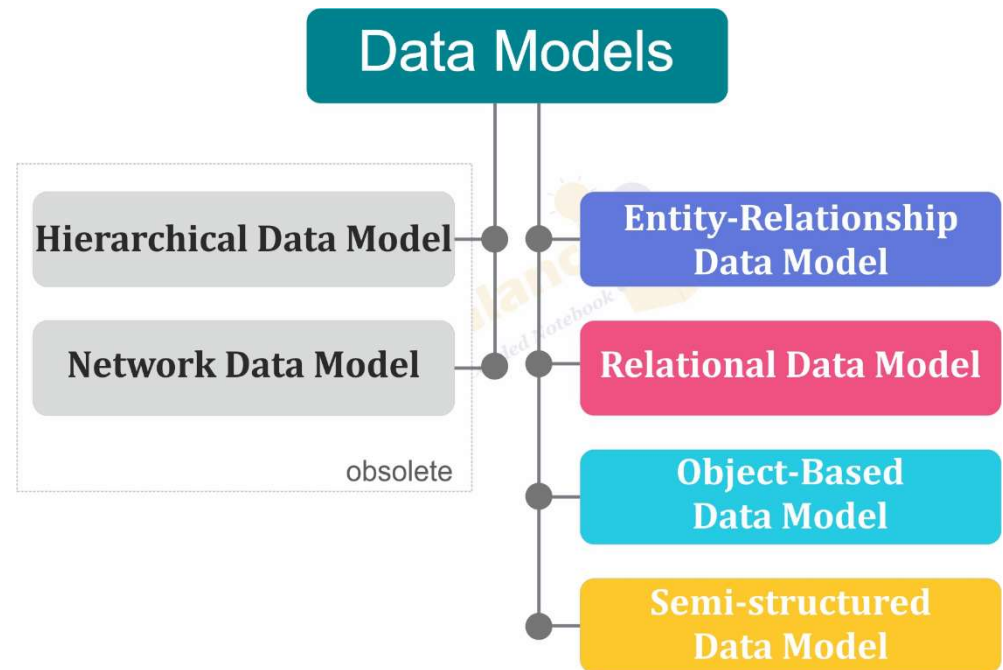
# 1. Tổng quan về mô hình dữ liệu

- ❖ Nhiều mô hình còn bao gồm cả một tập các phép toán để thao tác các dữ liệu
- ❖ **Mô hình thuộc dạng ngữ nghĩa:** tập trung về ngữ nghĩa của dữ liệu như mô hình thực thể liên kết, sử dụng để hỗ trợ người dùng có cái nhìn khái quát về dữ liệu
- ❖ **Mô hình thuộc dạng khái niệm:** tập trung vào cách thức tổ chức dữ liệu tại mức khái niệm như mô hình mạng, mô hình liên kết, mô hình quan hệ, độc lập với DBMS và hệ thống phần cứng để cài đặt cơ sở dữ liệu

# 1. Tổng quan về mô hình dữ liệu

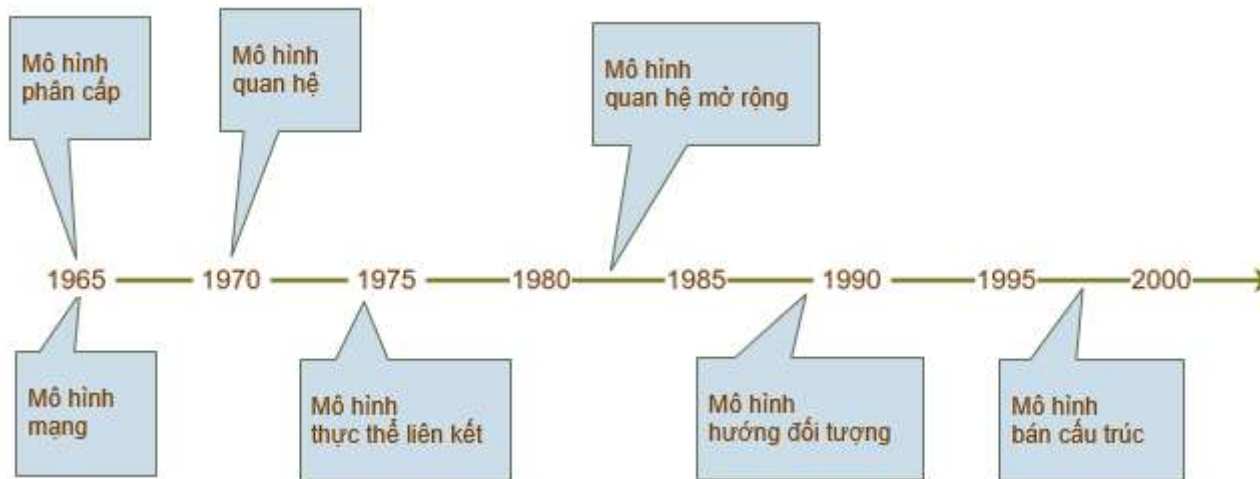
- Một số mô hình dữ liệu:

- |  |                              |
|--|------------------------------|
| 1. Hierarchical Model                        | A Hierarchical model         |
| 2. Network Model                             | Network model                |
| 3. <a href="#">Entity-relationship Model</a> | Model of Entity-Relationship |
| 4. <a href="#">Relational Model</a>          | Relational Model             |
| 5. Object-oriented Model                     | Object-Oriented Data Model   |
| 6. NoSQL Model                               | Object-Relational Data Model |
| 7. Graph Model                               | Flat Data Model              |
|  | Semi-Structured Data Model   |
|  | Associative Data Model       |
|  | Context Data Model           |



# 1. Tổng quan về mô hình dữ liệu

- Một số mô hình dữ liệu:



- A Hierarchical model
- Network model
- Model of Entity-Relationship
- Relational Model
- Object-Oriented Data Model
- Object-Relational Data Model
- Flat Data Model
- Semi-Structured Data Model
- Associative Data Model
- Context Data Model



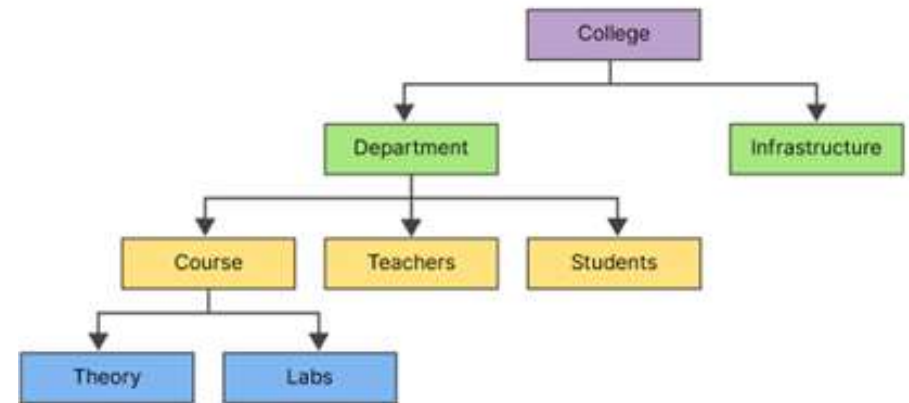
## 2. Mô hình cơ sở dữ liệu phân cấp

### ❖ Đặt vấn đề:

- Đặc điểm của các mô hình dữ liệu?
- Sự khác nhau giữa các mô hình dữ liệu?
- Các mô hình dữ liệu phổ biến ngày nay

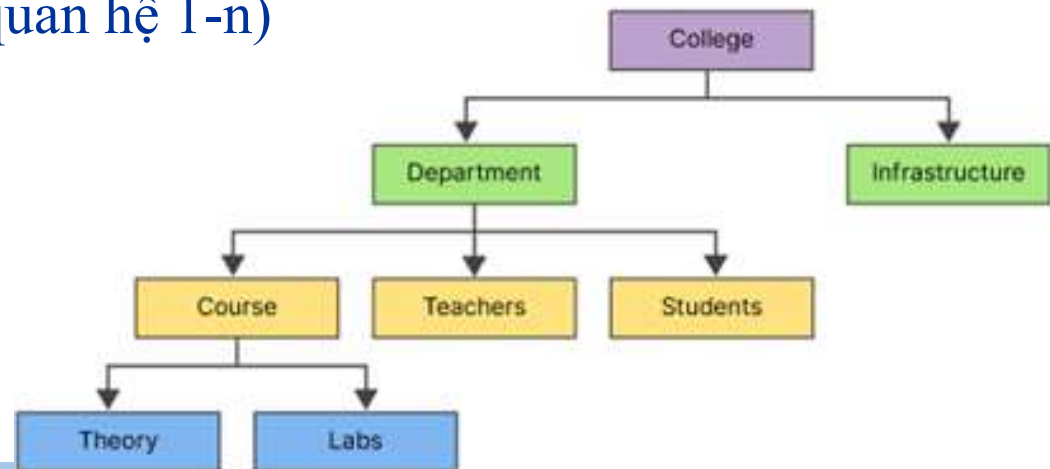
### ❖ **Mô hình cơ sở dữ liệu phân cấp**

- Ra đời những năm 60-65
- Biểu diễn theo mô hình cây
  - Quan hệ cha-con
  - Mỗi nút có 1 cha duy nhất
  - 1 CSDL = 1 tập các cây = 1 rừng
- Các khái niệm cơ bản
  - Bản ghi: chứa dữ liệu
  - Móc nối
  - Các phép toán: GET, GET UNIQUE, GET NEXT, GET NEXT WITHIN PARENT,...



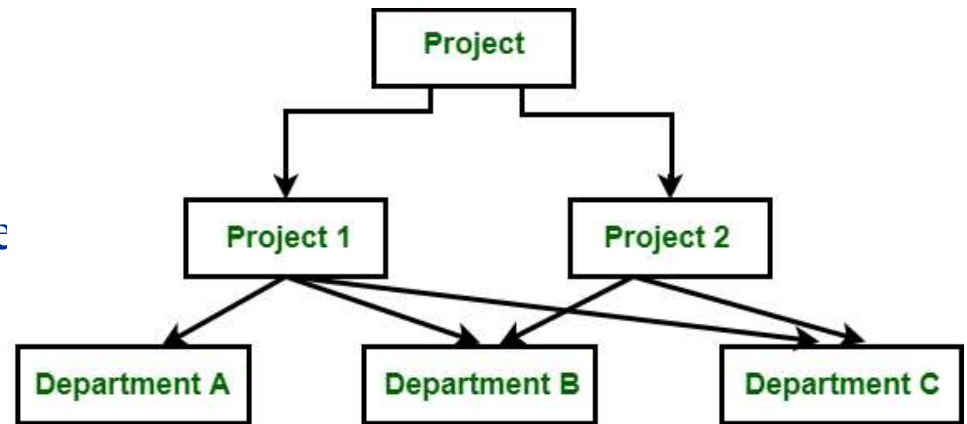
## 2. Mô hình cơ sở dữ liệu phân cấp

- Ưu điểm
  - Dễ xây dựng và thao tác
  - Tương thích với các lĩnh vực tổ chức phân cấp
  - Ngôn ngữ thao tác đơn giản: duyệt cây
- Nhược điểm
  - Sự lặp lại của các kiểu bản ghi dữ liệu dư thừa và không nhất quán.
  - Hạn chế trong biểu diễn ngữ nghĩa của các mối nối giữa các bản ghi (chỉ cho phép quan hệ 1-n)



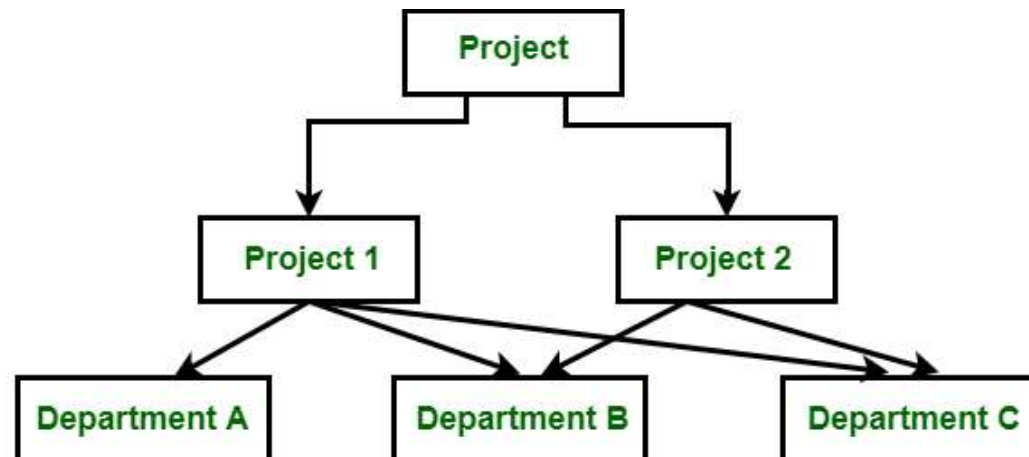
### 3. Mô hình dữ liệu mạng

- Sự ra đời
  - Sử dụng phổ biến từ những năm 60, được định nghĩa lại vào năm 71
- Biểu diễn bằng đồ thị có hướng
- Các khái niệm cơ bản
  - Tập bản ghi (record)
    - Kiểu bản ghi (record type)
    - Các trường (field)
  - Móc nối
    - Tên của móc nối
    - Chủ (ownERD) – thành viên (membERD): theo hướng của móc nối
    - Kiểu móc nối: 1-1, 1-n, đệ quy
  - Các phép toán
    - Duyệt: FIND, FIND membERD, FIND ownERD, FIND NEXT
    - Thủ tục: GET



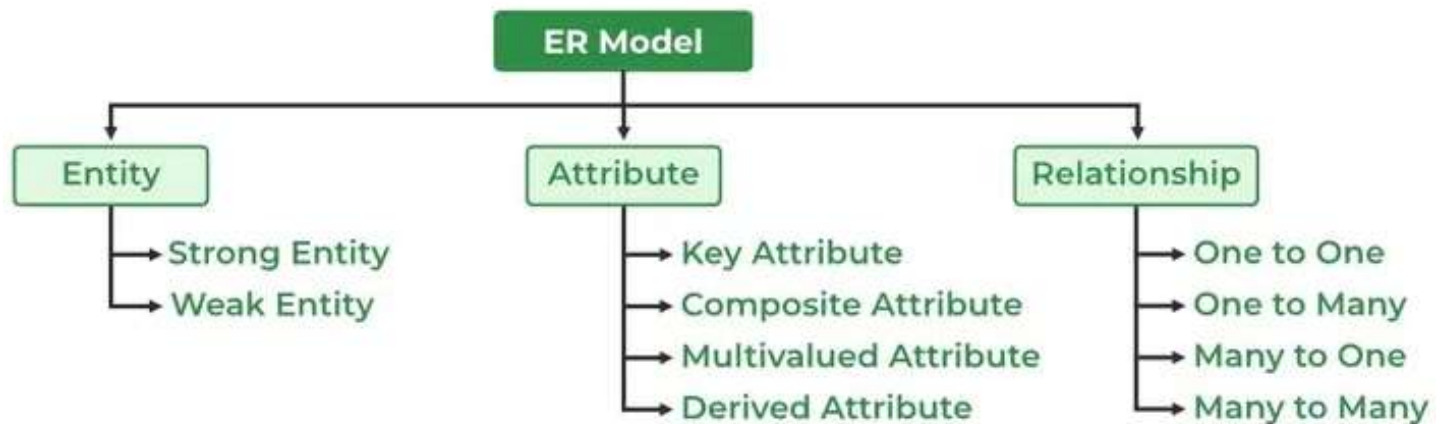
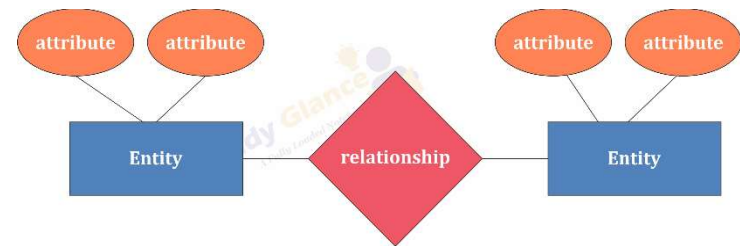
### 3. Mô hình dữ liệu mạng

- Ưu điểm
  - Đơn giản
  - Có thể biểu diễn các ngữ nghĩa đa dạng với kiểu bản ghi và kiểu móc nối
  - Truy vấn thông qua phép duyệt đồ thị
- Nhược điểm
  - Số lượng các con trỏ lớn.
  - Hạn chế trong biểu diễn ngữ nghĩa của các móc nối giữa các bản ghi



## 4. Mô hình thực thể liên kết (ERD)

- Dùng để thiết kế DB ở mức quan niệm (bước đầu thiết kế DB), làm nền tảng để ánh xạ sang một mô hình khái niệm nào đó mà Hệ quản trị CSDL sẽ sử dụng.
- Biểu diễn các đối tượng trong thế giới thực và mối quan hệ giữa chúng.
- ERD bao gồm:
  - Tập thực thể (Entity sets)/ Thực thể
  - Thuộc tính (Attributes), Khóa
  - Mối quan hệ (Relationship)



## 4. Mô hình thực thể liên kết (ERD)

### Thực thể


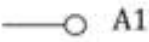
- Đặc điểm
  - Một đối tượng trong thế giới thực, tồn tại độc lập và phân biệt được với các đối tượng khác.
  - Có tên gọi riêng
  - Có danh sách thuộc tính mô tả đặc trưng của thực thể
  - Có khóa thực thể
- Tập thực thể: một tập hợp các thực thể có tính chất giống nhau
- Kí hiệu: 

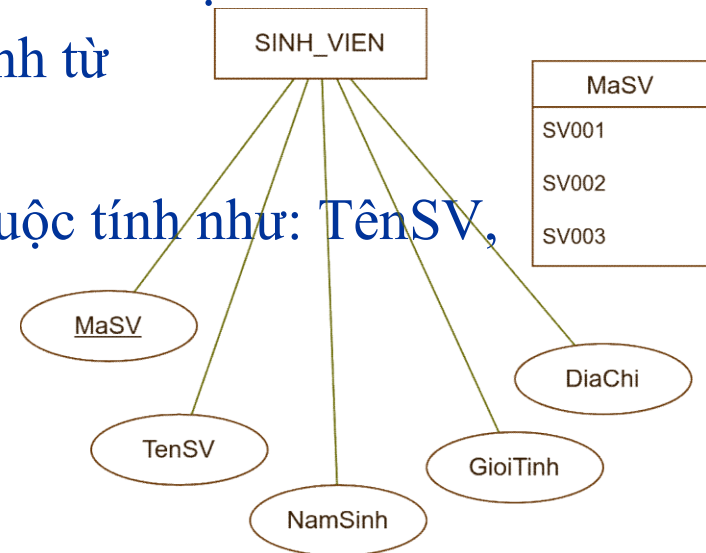
<b>E</b>
----------

, tên: danh từ hoặc cụm danh từ
- Ví dụ:
  - Thực thể: một sinh viên, một lớp
  - Tập thực thể: toàn thể sinh viên của 1 lớp, toàn thể các lớp của 1 khoa

## 4. Mô hình thực thể liên kết (ERD)

### Thuộc tính

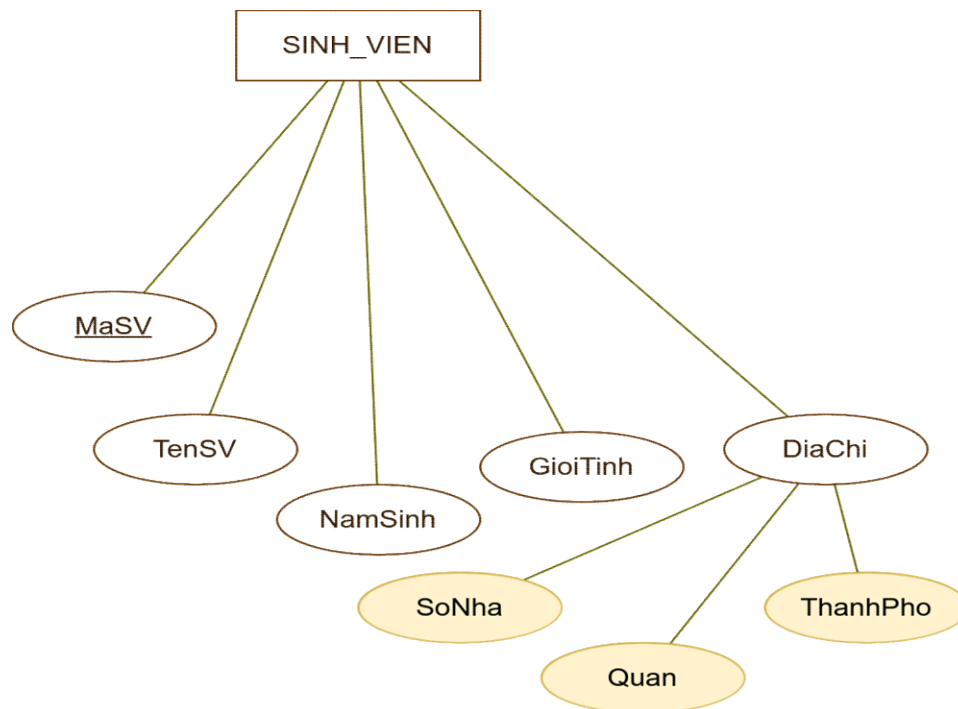
- Đặc điểm:
  - Là những đặc tính riêng biệt của một tập thực thể.
  - Diễn tả các thuộc tính thành phần của thực thể hay mối kết hợp
  - Các thông tin mở rộng
  - Là những giá trị nguyên tố (không thể chia nhỏ hơn về mặt ngữ nghĩa): kiểu chuỗi, kiểu số nguyên, kiểu số thực.
  - Tên thuộc tính: danh từ hoặc cụm danh từ
- Kí hiệu:   A1
- Ví dụ: Thực thể SINH\_VIEN có các thuộc tính như: TênSV, NamSinh,...



## 4. Mô hình thực thể liên kết (ERD)

### Phân loại thuộc tính

- Thuộc tính đơn giản (thuộc tính nguyên tố)
  - có kiểu dữ liệu nguyên tố
- Thuộc tính phức
  - có kiểu phức, định nghĩa bởi các thuộc tính khác

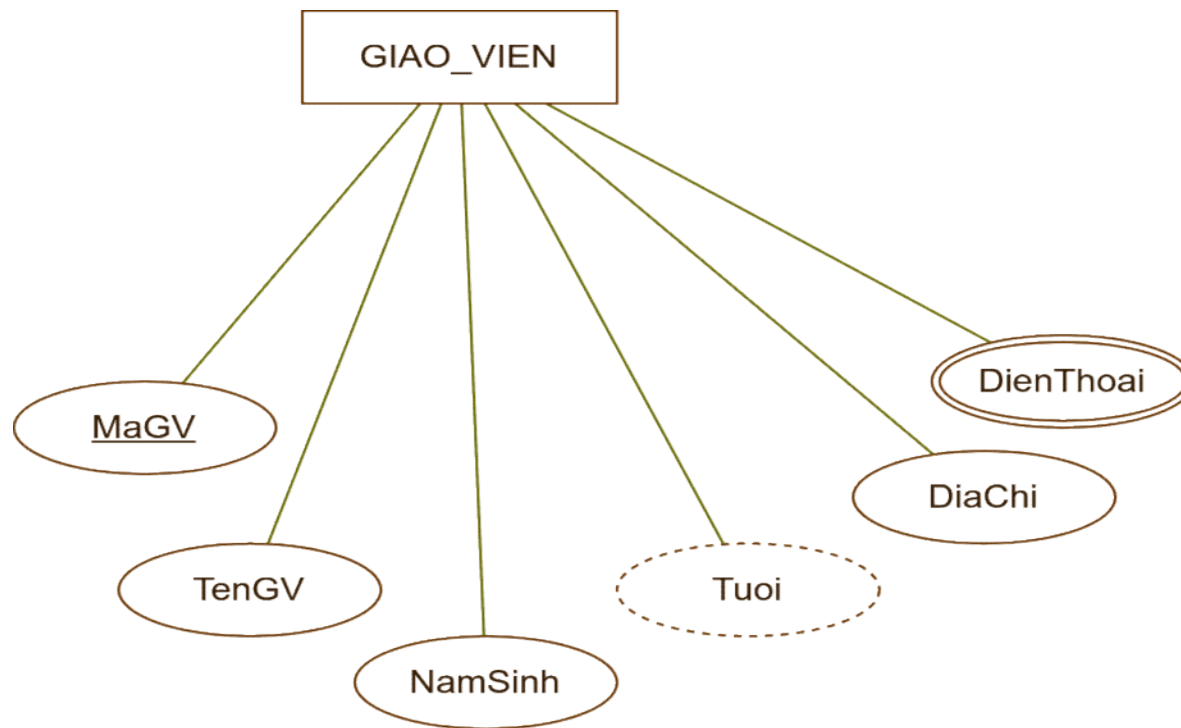




## 4. Mô hình thực thể liên kết (ERD)

### Kiểu thuộc tính

- Thuộc tính đa giá trị
  - tương ứng với mỗi thực thể, có thể nhận nhiều giá trị. VD: Số điện thoại
- Thuộc tính suy diễn
  - Giá trị được tính toán từ (các) thuộc tính khác. VD: tuổi



## 4. Mô hình thực thể liên kết (ERD)

### Kiểu thuộc tính - Thuộc tính Khoá

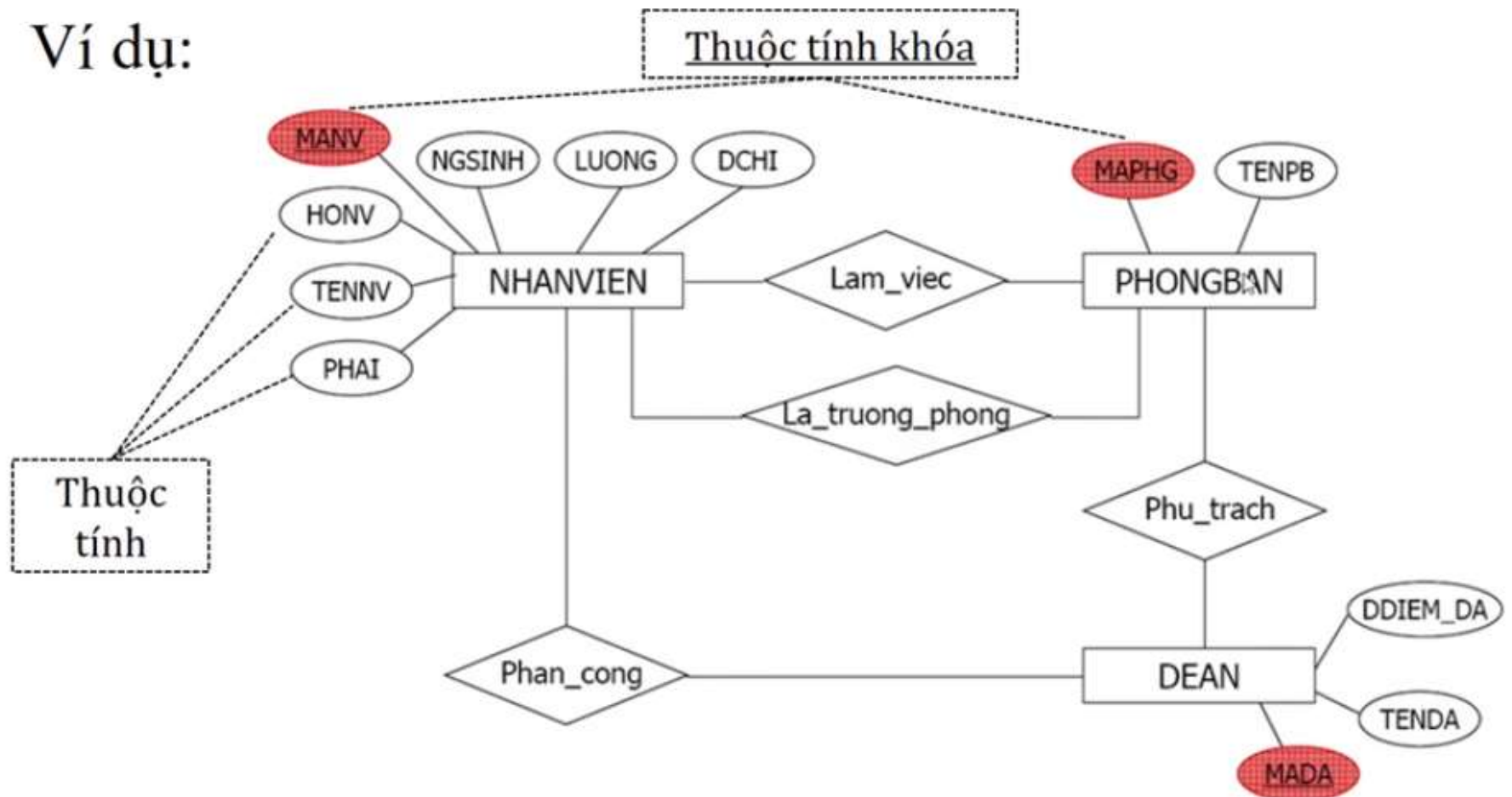
- Một hay một tập thuộc tính mà giá trị của chúng có thể xác định **duy nhất** (giá trị tại các thuộc tính trong khóa phải khác nhau) một thực thể trong tập thực thể
  - Tập thực thể SinhViên có thể dùng MaSV làm khóa
- Khóa của tập thực thể là 1 hay nhiều thuộc tính.
  - Khóa gồm nhiều thuộc tính thì gọi là khóa phức
  - Một tập thực thể có thể có nhiều khóa nhưng chỉ một trong số các khóa được chọn làm khóa chính
  - Trong sơ đồ ERD, thuộc tính nào được chọn làm **khóa chính** sẽ được **gạch chân**
- Chú ý:
  - Mỗi tập thực thể phải có 1 khóa
  - 1 khóa có thể có 1 hay nhiều thuộc tính



## 4. Mô hình thực thể liên kết (ERD)


**Kiểu thuộc tính - Thuộc tính Khoá**

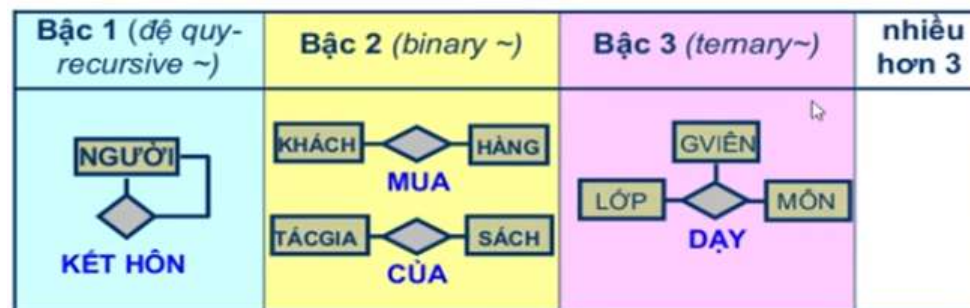
Ví dụ:



## 4. Mô hình thực thể liên kết (ERD)

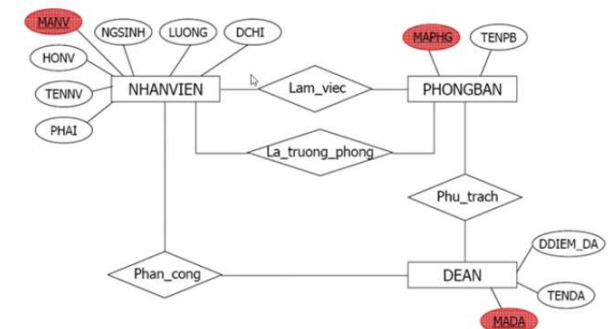
### Liên kết – tập liên kết (Kết hợp – tập kết hợp)

- Kí hiệu:
- Đặc điểm 
  - Diễn tả một môi liên hệ ít nhất giữa 2 thực thể khác nhau
    - Quan hệ giữa 2 thực thể: liên kết nhị phân
    - Quan hệ nhiều thực thể: liên kết đa phân
- Bậc của mỗi liên kết: số thực thể tham gia vào quan hệ



Ví dụ:

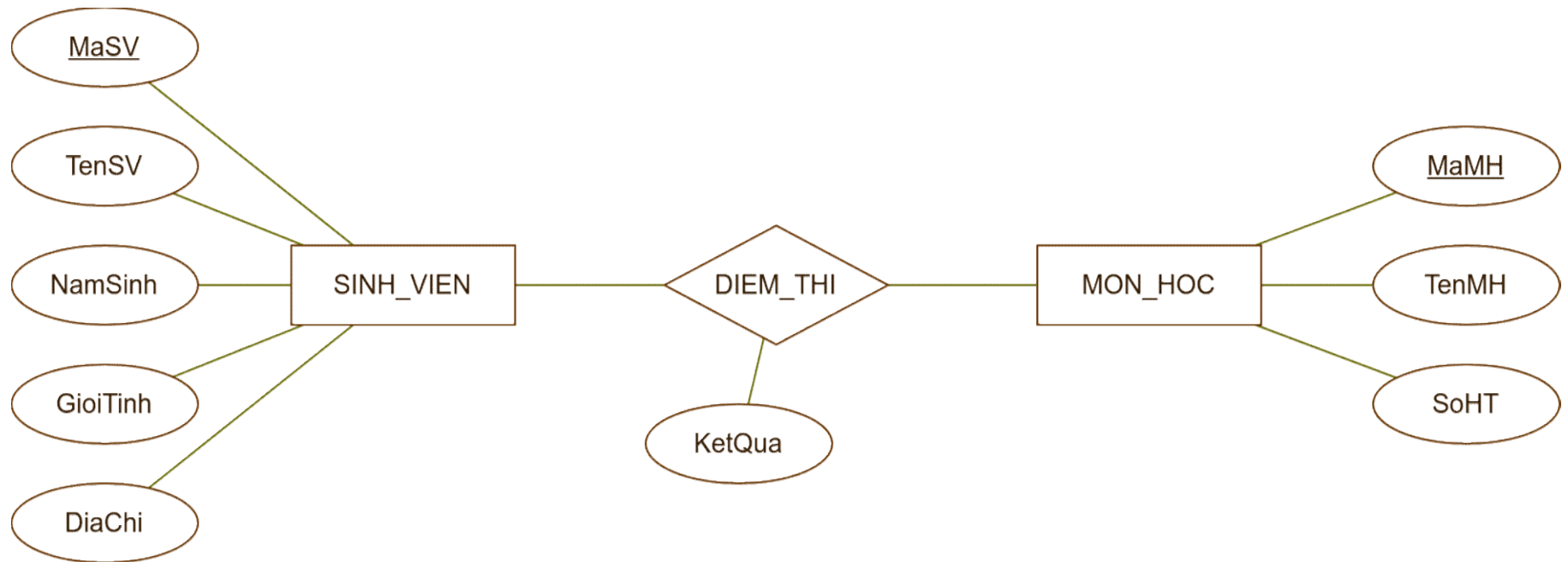
- NV (làm việc) ở phòng ban nào đó
- phòng ban có NV (là trưởng phòng)



## 4. Mô hình thực thể liên kết (ERD)

### Liên kết – tập liên kết (Kết hợp – tập kết hợp)

- Ví dụ:



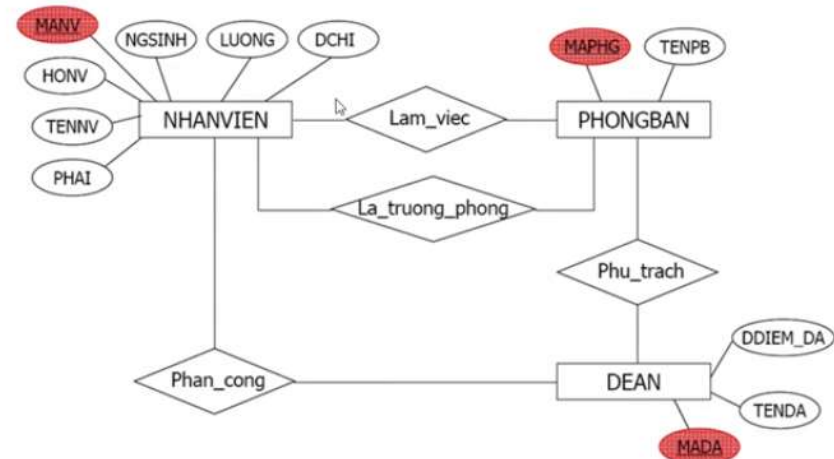
## 4. Mô hình thực thể liên kết (ERD)

### Liên kết – tập liên kết (Kết hợp – tập kết hợp)

- Tập liên kết là một tập hợp các liên kết cùng kiểu
  - Giữa tập thực thể SinhViên và Lớp có 1 tập liên kết ThànhViên, chỉ ra rằng mỗi sinh viên đều là thành viên của 1 lớp nào đó
- Một liên kết có thể có thuộc tính riêng
- Phân loại:
  - Một – một (1-1)
  - Một- nhiều (1-n).
  - Nhiều – một (n-1)
  - Nhiều – nhiều (n-n)

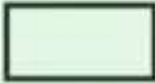




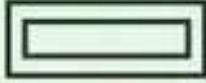
Ví dụ:

- NV (làm việc) ở phòng ban nào đó
- phòng ban có NV (là trưởng phòng)



## 4. Mô hình thực thể liên kết (ERD)

### Sơ đồ ERD

Figures	Symbols	Represents
Rectangle		Entities in ER Model
Ellipse		Attributes in ER Model
Diamond		Relationships among Entities
Line		Attributes to Entities and Entity Sets with Other Relationship Types
Double Ellipse		Multi-Valued Attributes
Double Rectangle		Weak Entity



## 4. Mô hình thực thể liên kết (ERD)

### ❖ Bản số các mối liên kết







- ❖ (min, max) chỉ định mỗi thực thể e thuộc tập các thực thể E tham gia ít nhất và nhiều nhất vào thể hiện của R.
- ❖ Chỉ quan tâm đến số nhỏ nhất và số lớn nhất
- ❖ Giải thích
  - (0,1): không hoặc một
  - (1,1): duy nhất một
  - (0,n): không hoặc nhiều
  - (1,n): một hoặc nhiều



## 4. Mô hình thực thể liên kết (ERD)

- ❖ Bản số các mối liên kết
  - Cách biểu diễn mối liên kết:
    - Biểu diễn theo số
    - Biểu diễn theo ký hiệu

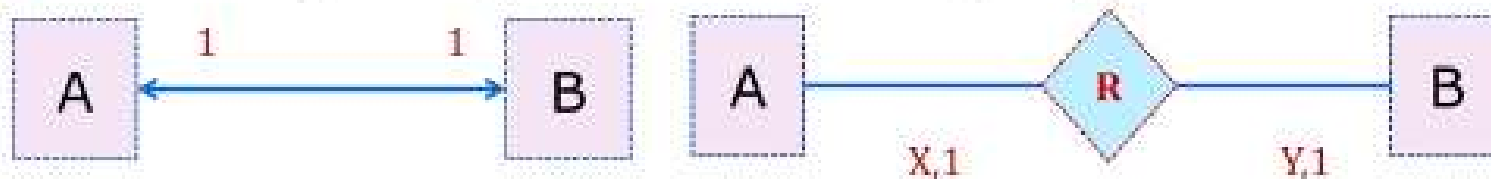
	B.số nhỏ nhất	B.số lớn nhất	Biểu diễn
1	bằng không (0)	bằng một (1)	
2	bằng không (0)	bằng nhiều (<=)	
3	bằng một (1)	bằng một (1)	
4	bằng một (1)	bằng nhiều (<=)	

## 4. Mô hình thực thể liên kết (ERD)

### ❖ Bản số các mối liên kết

#### ▪ Mối liên kết 1-1

- Mỗi cá thể của thực thể A có liên kết với 0 hay 1 cá thể trong thực thể B và ngược lại.
- R: tên của quan hệ giữa hai cá thể trong 2 thực thể A và B.

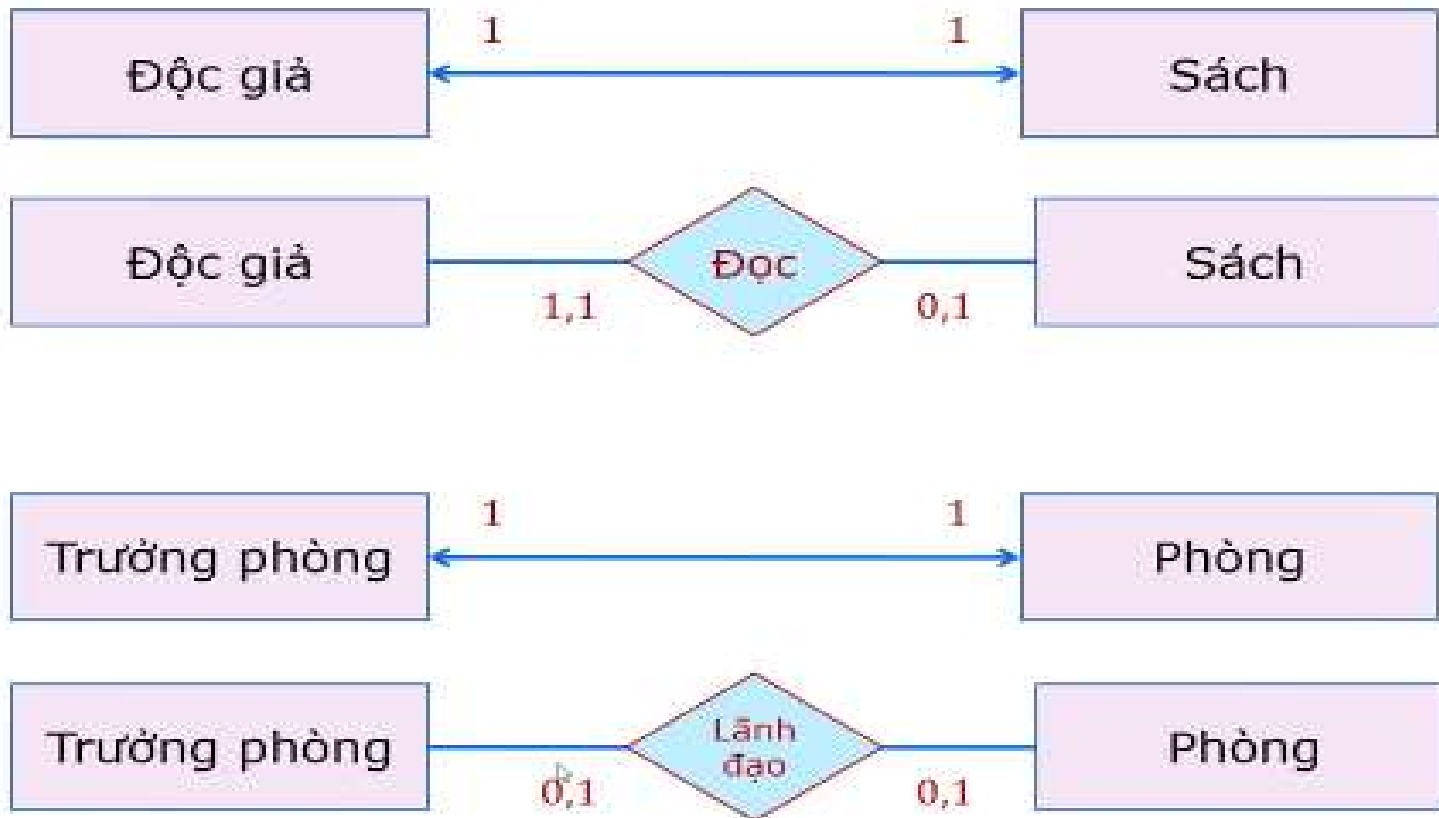


- VD: Sinh viên có thể có (0,1) tài khoản. Tài khoản thuộc về (1,1) sinh viên.



## 4. Mô hình thực thể liên kết (ERD)

- ❖ Bản số các mối liên kết
  - Mối liên kết 1-1

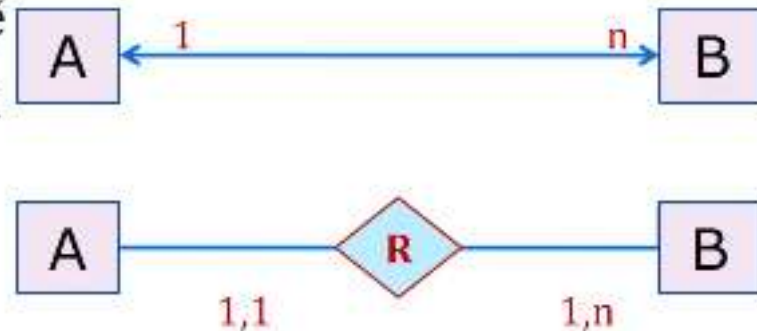


## 4. Mô hình thực thể liên kết (ERD)

### ❖ Bản số các mối liên kết

#### ▪ Mối liên kết 1-n

- Mỗi cá thể của thực thể A có liên kết với 0 hay n cá thể trong thực thể B. Tuy nhiên, 1 thực thể trong B chỉ kết hợp được với 1 thực thể trong A

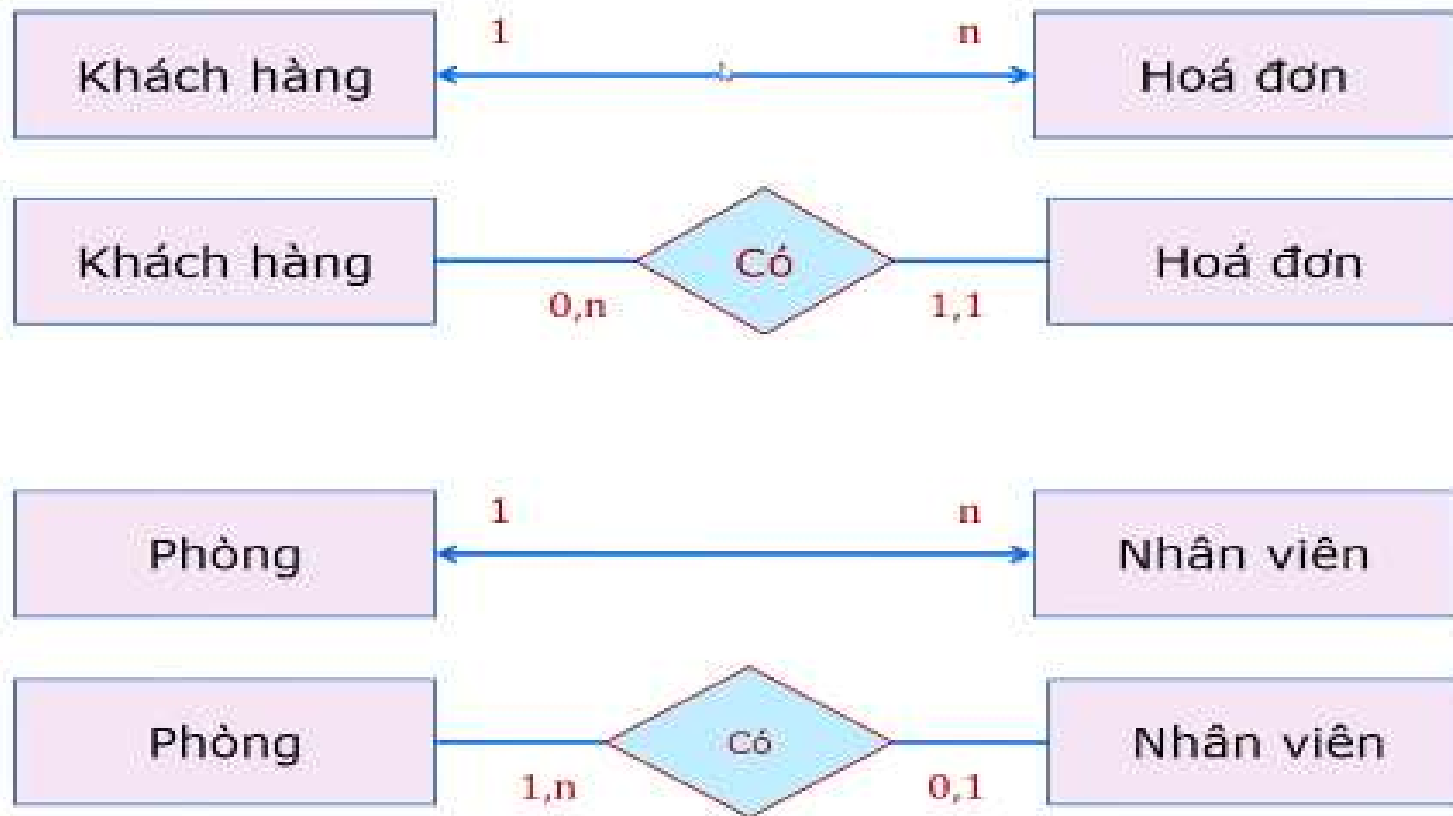


- Ví dụ: Một sinh viên có thể thuộc về (1,1) khoa. Một khoa có thể có (0,n) sinh viên.



## 4. Mô hình thực thể liên kết (ERD)

- ❖ Bản số các mối liên kết
  - Mỗi liên kết 1-n

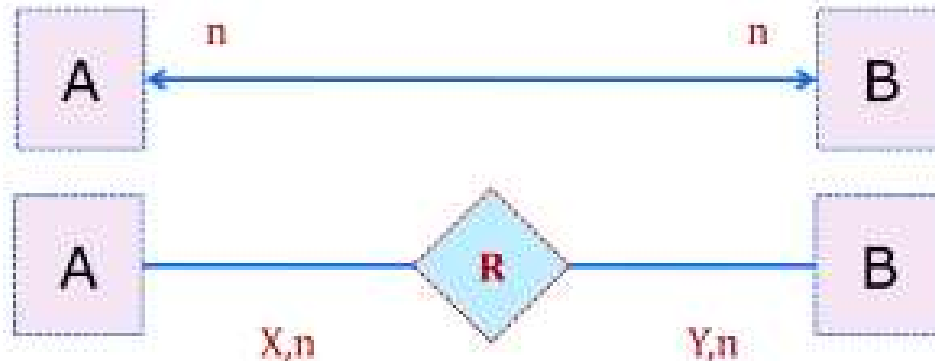


## 4. Mô hình thực thể liên kết (ERD)

### ❖ Bản số các mối liên kết

#### ▪ Mối liên kết n-n

- Mỗi cá thể của thực thể A có liên kết với 0 hay n cá thể trong thực thể B và ngược lại.

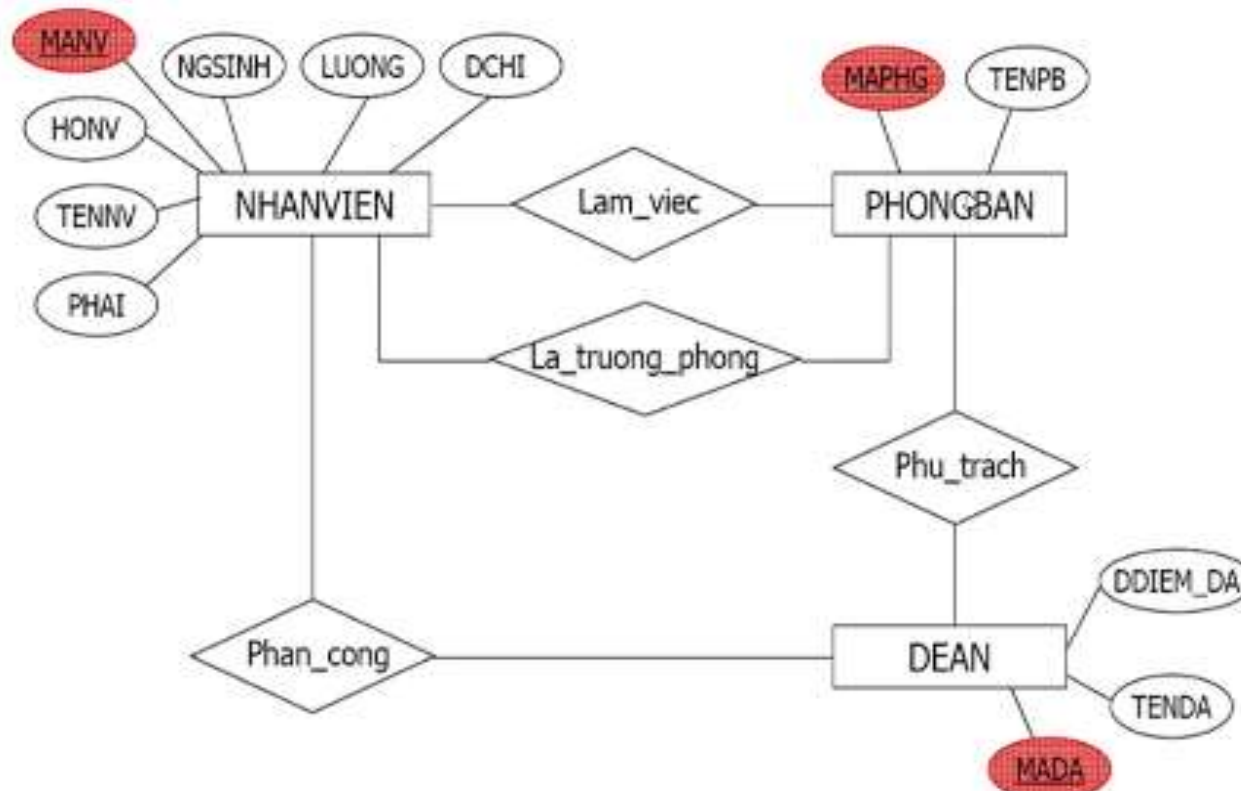


- VD: Một sinh viên có thể học (0,n) môn học. Một môn học có thể được học bởi (0,n) sinh viên.



## 4. Mô hình thực thể liên kết (ERD)

- ❖ Bản số các mối liên kết
  - Mối liên kết: ví dụ



1. Nhân viên làm việc/thuộc trong (**duy nhất**) một phòng ban nào đó.

Mỗi phòng ban có **1 hoặc nhiều** nhân viên làm việc.

2. Nhân viên có thể là **trưởng phòng hoặc không** của một phòng ban nào đó

Mỗi phòng ban có **0 hoặc 1** trưởng phòng

3. Mỗi phòng ban **phụ trách không hoặc nhiều** đề án

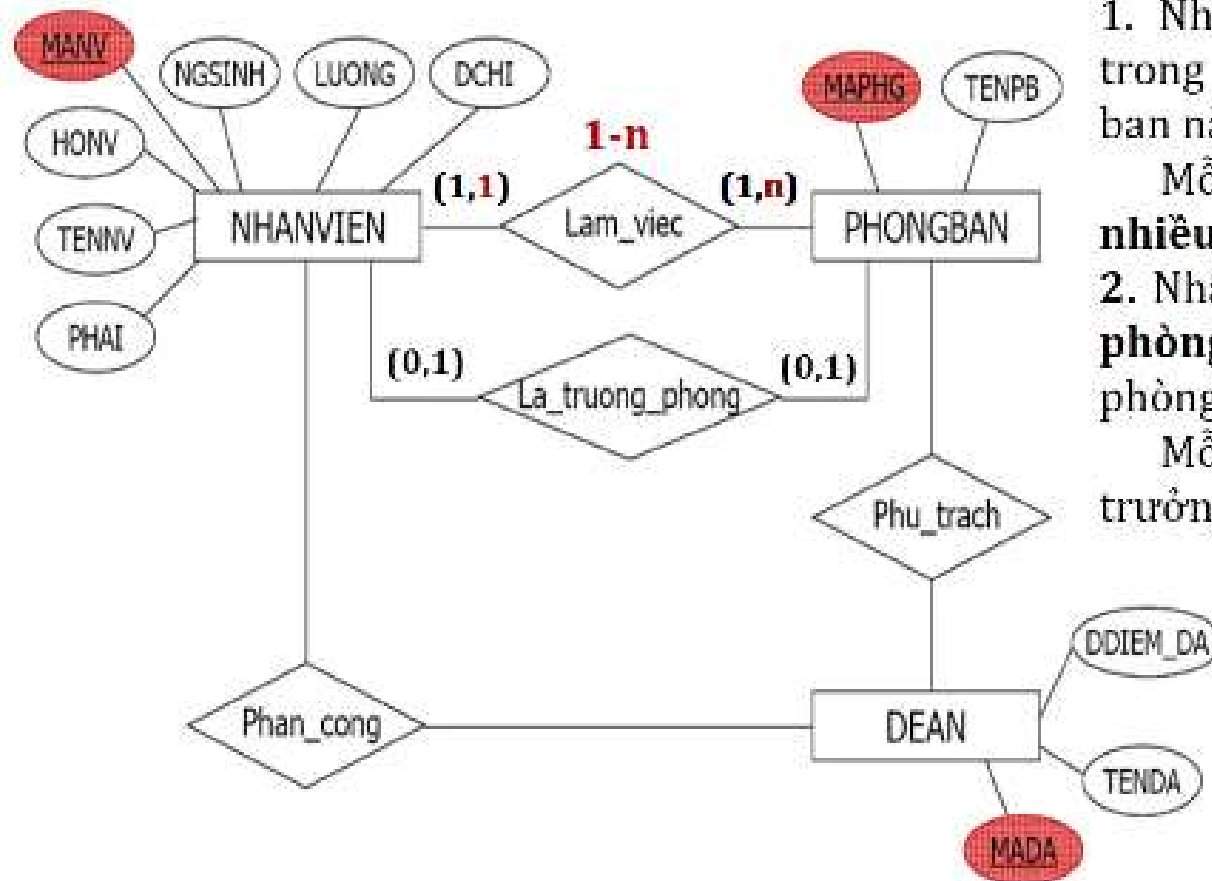
Mỗi đề án được **phụ trách bởi chỉ duy nhất 1** phòng ban

4. Mỗi nhân viên **thực hiện 0 hoặc nhiều** đề án

Mỗi đề án được **phân công cho 1 hoặc nhiều** nhân viên

## 4. Mô hình thực thể liên kết (ERD)

- ❖ Bản số các mối liên kết
- Mỗi liên kết: ví dụ



1. Nhân viên **làm việc/thuộc** trong **(duy nhất) một** phòng ban nào đó.

Mỗi phòng ban có **1 hoặc nhiều** nhân viên **làm việc**.

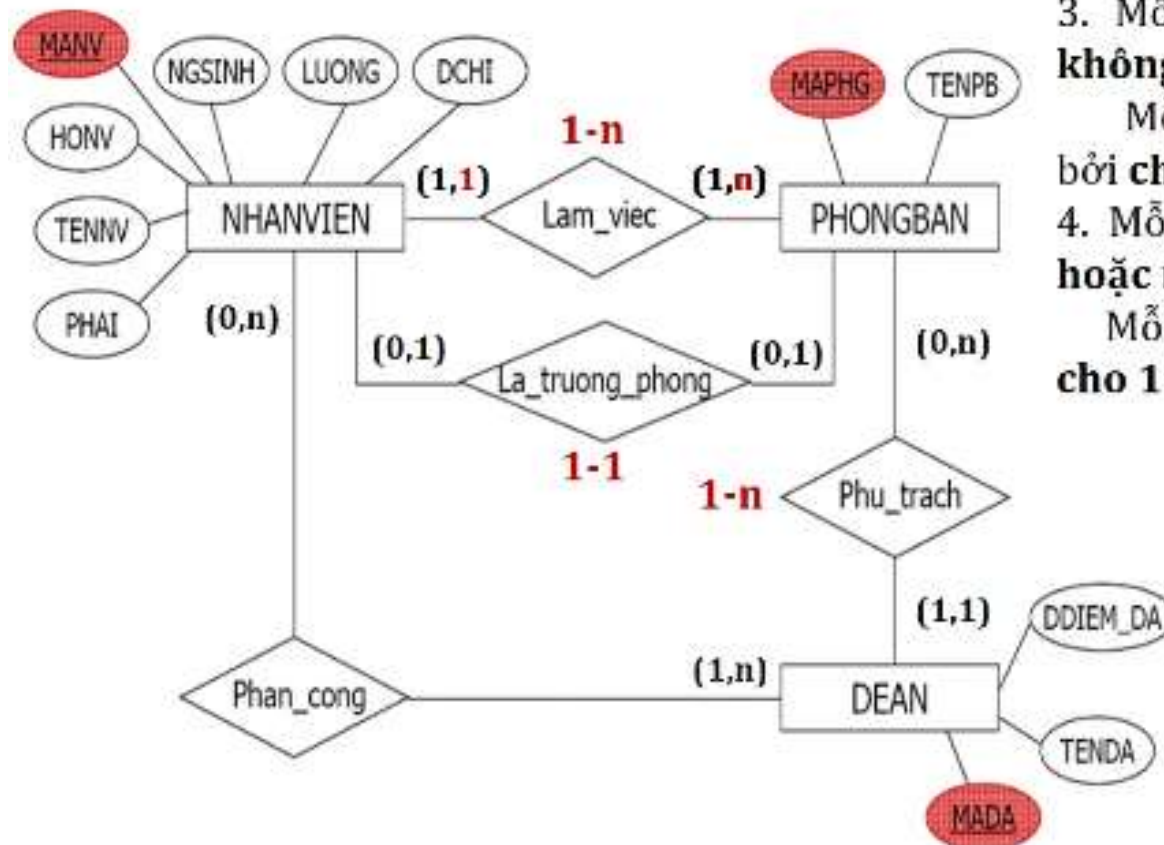
2. Nhân viên **có thể là trưởng phòng hoặc không** của một phòng ban nào đó

Mỗi phòng ban **có 0 hoặc 1** trưởng phòng



## 4. Mô hình thực thể liên kết (ERD)

- ❖ Bản số các mối liên kết
- Mỗi liên kết: ví dụ



3. Mỗi phòng ban **phụ trách không hoặc nhiều** đề án

Mỗi đề án **được phụ trách bởi chỉ duy nhất 1** phòng ban

4. Mỗi nhân viên **thực hiện 0 hoặc nhiều** đề án

Mỗi đề án **được phân công cho 1 hoặc nhiều** nhân viên

## 4. Mô hình thực thể liên kết (ERD)

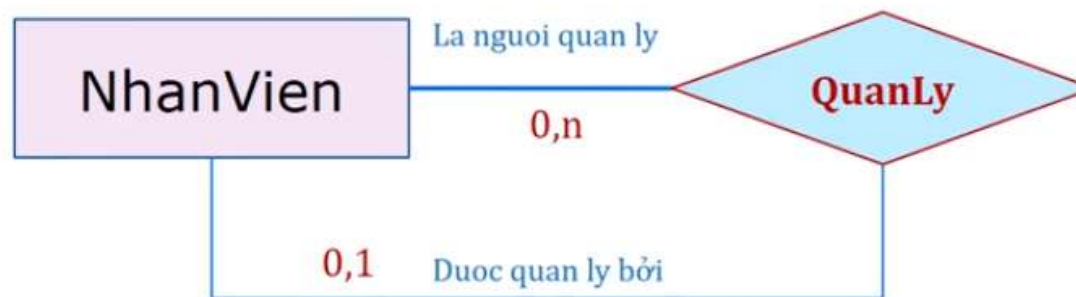
### ❖ Bản số các mối liên kết

#### ▪ Mối liên kết vòng

- Một loại thực thể có thể tham gia nhiều lần vào một quan hệ với nhiều vai trò khác nhau



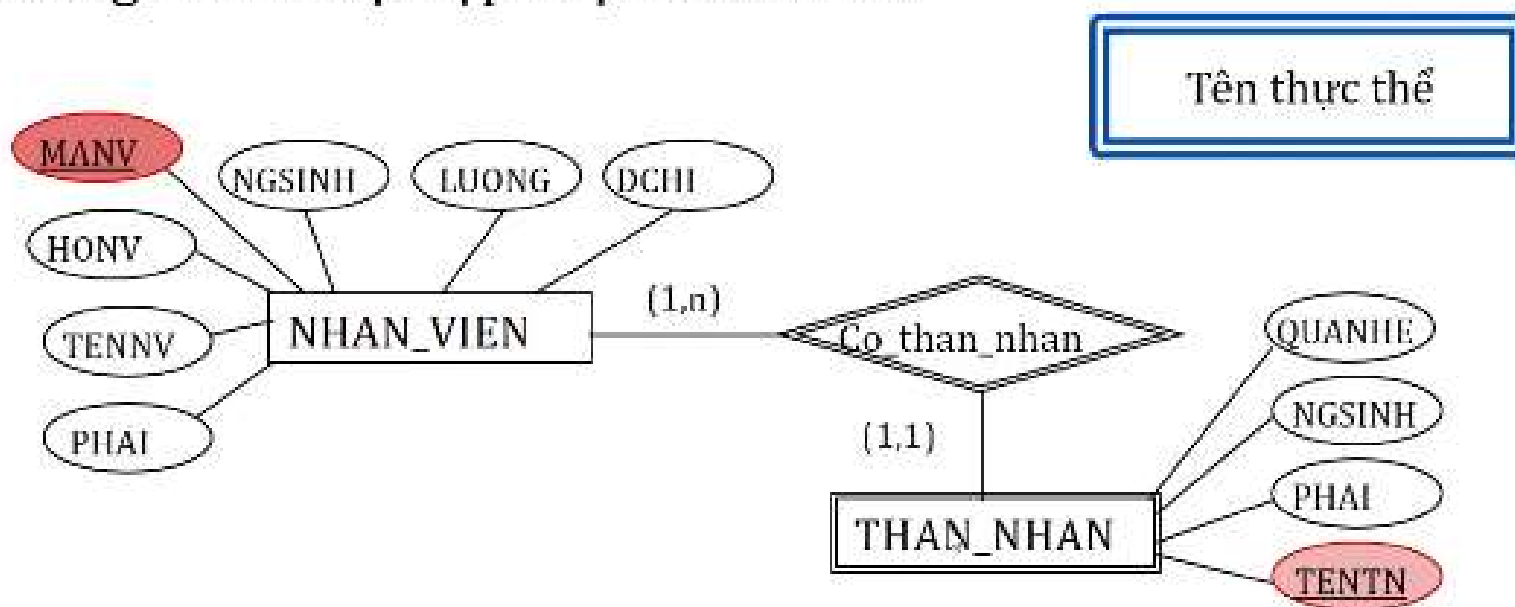
- VD:



## 4. Mô hình thực thể liên kết (ERD)

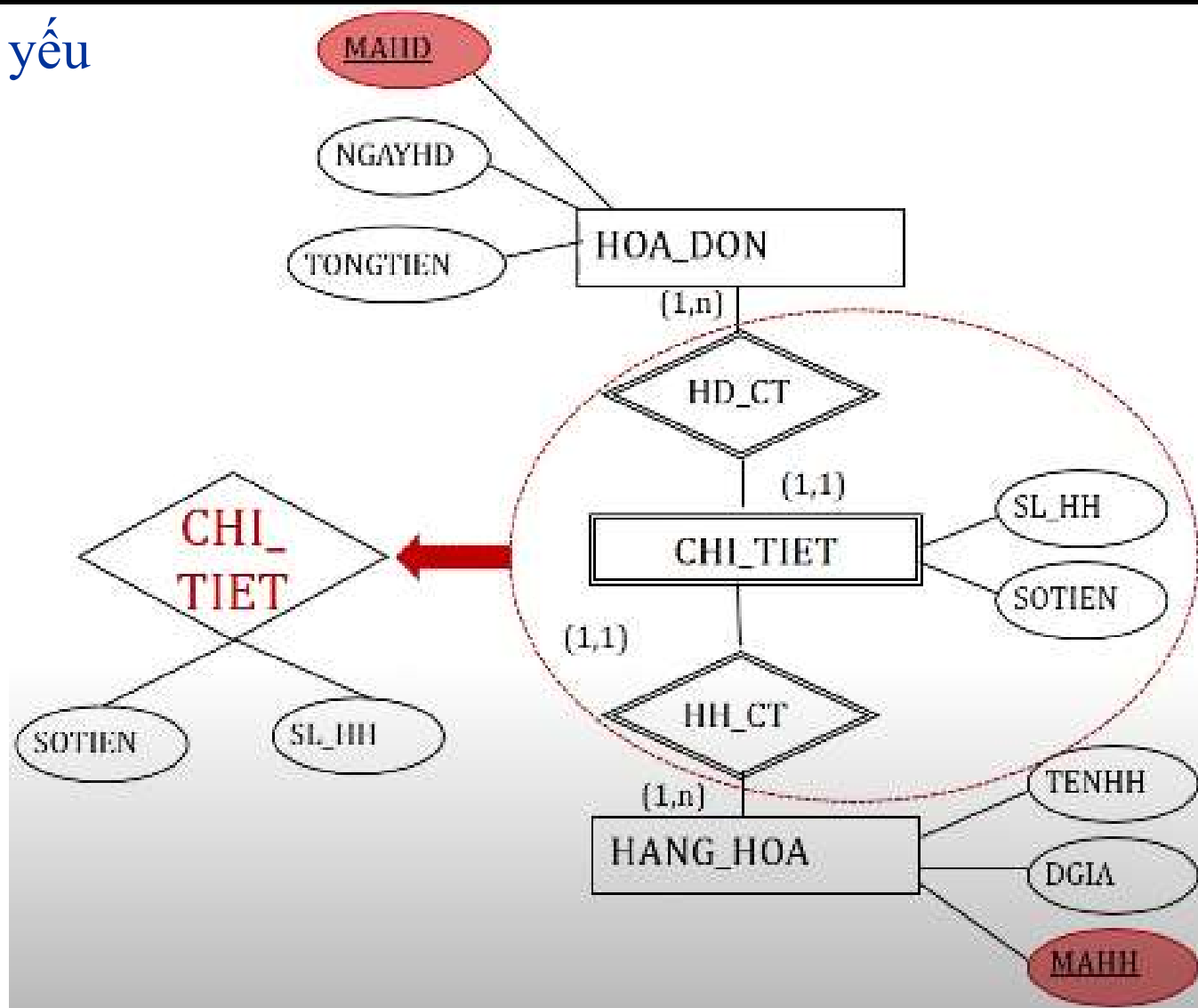
### ❖ Thực thể yếu

- ❖ Là thực thể mà khóa có được từ những thuộc tính của tập thực thể khác
- ❖ Thực thể yếu (weak entity set) phải tham gia vào mỗi quan hệ mà trong đó có một tập thực thể chính



## 4. Mô hình thực thể liên kết (ERD)

### ❖ Thực thể yếu



## 4. Mô hình thực thể liên kết (ERD)

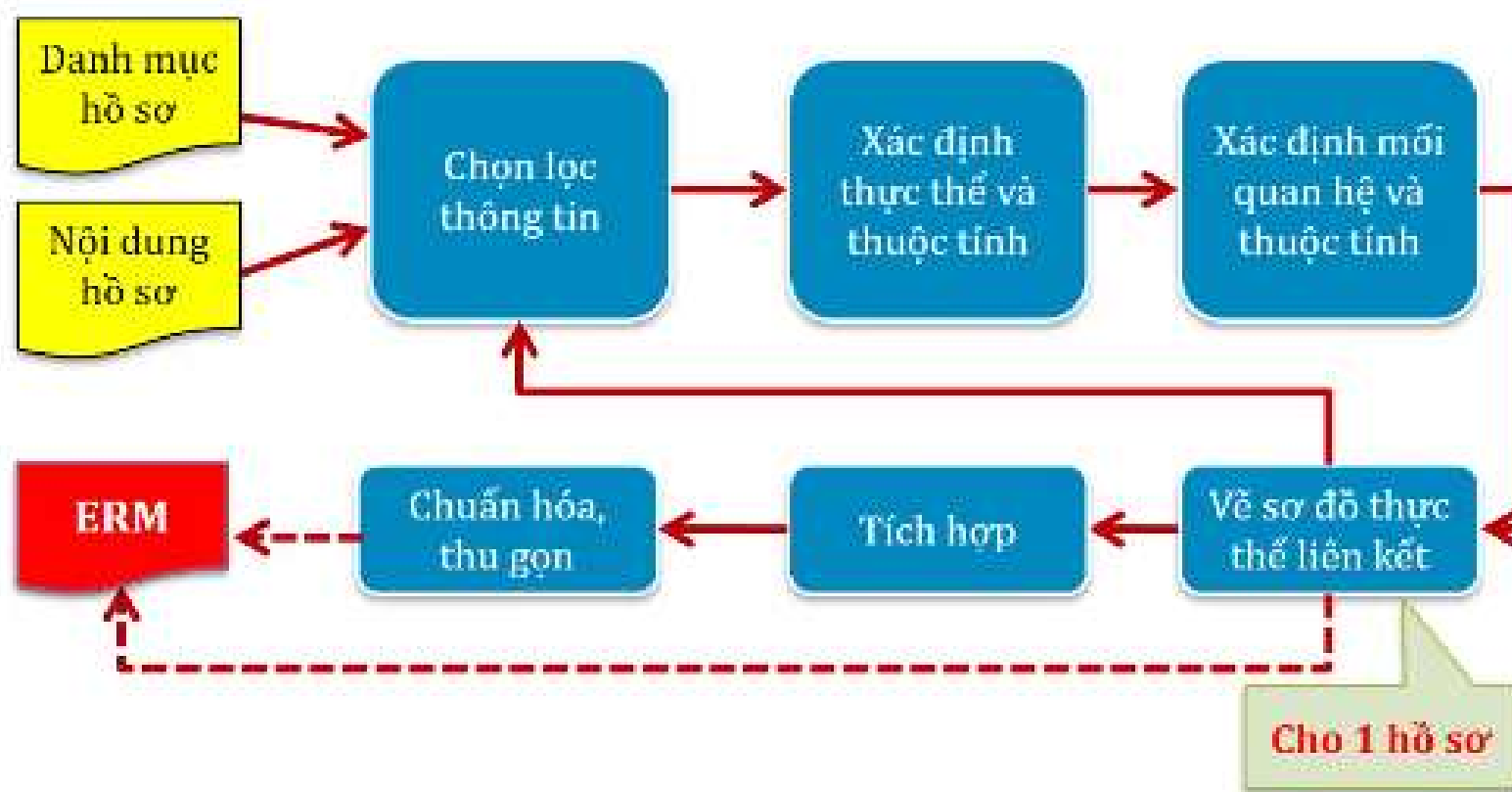
### ❖ Quy trình xây dựng ERD

#### • Cách 1: thực hiện đồng thời



## 4. Mô hình thực thể liên kết (ERD)

- ❖ Quy trình xây dựng ERD
- Cách 2: thực hiện từng hồ sơ rồi tích hợp



## 4. Mô hình thực thể liên kết (ERD)

❖ Quy trình xây dựng ERD:

### Các bước thực hiện

- **Bước 1:** Chọn lọc thông tin → Từ điển dữ liệu
  - Quy tắc:
    - Chính xác hóa
    - Chọn lọc
- **Bước 2:** Xác định thực thể
  - Quy tắc:
    - Tìm thuộc tính tên gọi → thực thể
    - Xác định thuộc tính định danh và thuộc tính còn lại
- **Bước 3:** Xác định mối quan hệ
  - Quy tắc:
    - Xác định mối quan hệ tương tác
    - Xác định mối quan hệ ràng buộc

## 4. Mô hình thực thể liên kết (ERD)

❖ Quy trình xây dựng ERD:

### Các bước thực hiện

- ❖ **Bước 4:** Vẽ sơ đồ bằng các công cụ (MS Visio, PowerDesigner, Case Studio)
  - Vẽ thực thể
  - Vẽ mối quan hệ
  - Sắp xếp lại cho cân đối, dễ nhìn
  - Bổ sung thuộc tính, gạch chân thuộc tính định danh
  - Xác định bản số
- **Bước 5:** Chuẩn hóa
  - Mục đích:
    - Loại bỏ thuộc tính lặp, nhóm lặp, ... và đảm bảo quy tắc nghiệp vụ
  - Rút gọn mô hình, nếu có thể
  - Xác định lại bản số



## 4. Mô hình thực thể liên kết (ERD)

❖ Quy trình xây dựng ERD:

**Các bước thực hiện – Ví dụ**

**Xây dựng CSDL dựa trên các loại phiếu**

**ĐƠN ĐẶT HÀNG**

Số: XXXXXX

Người đặt hàng: .....

Địa chỉ: .....

Ngày đặt: .....

Số tđ	Tên hàng	Đơn vị	Đơn giá	Số lượng	Thành tiền
XX	XXX	XXX	XX	XX	XX
...	...	...	...	...	...

**PHIẾU XUẤT KHO**

Số: XXXXXX

Họ và tên người mua hàng: .....

Địa chỉ: .....

Lý do xuất kho: .....

Xuất tại kho: .....

STT	Tên hàng	ĐVT	Số lượng	Đơn giá	Thành tiền
XXXXXX	XXXXXX	XXXXXX	XXXXXX	XXXXXX	XXXXXX
XXXXXX	XXXXXX	XXXXXX	XXXXXX	XXXXXX	XXXXXX
<b>Tổng tiền hàng</b>					XXXXXX
<b>Thuế VAT</b>					XXXXXX
<b>Tổng cộng</b>					XXXXXX

## 4. Mô hình thực thể liên kết (ERD)

❖ Quy trình xây dựng ERD:

**Các bước thực hiện – Ví dụ: Xây dựng ERD cho Đơn đặt hàng**

- Bước 1: chọn lọc thông tin

Từ trong HSDL	Từ rõ nghĩa	Viết tắt
Số	Mã đơn đặt hàng	MaDDH
Người đặt hàng	<u>Tên người đặt hàng</u>	TenNDH
Địa chỉ	Địa chỉ	DiaChi
Ngày đặt	Ngày đặt	NgayDat
Tên hàng	<u>Tên hàng</u>	TenHang
Đơn vị	Đơn vị	DonVi
Đơn giá	Đơn giá	DonGia
Số lượng	Số lượng	SoLuong
Thành tiền	Thành tiền	ThanhTien

**ĐƠN ĐẶT HÀNG**

Số: XXXXXX

Người đặt hàng: .....

Địa chỉ: .....

Ngày đặt: .....

Số tt	Tên hàng	Đơn vị	Đơn giá	Số lượng	Thành tiền
xx	xxx	xxx	xx	xx	xx
...	...	...	...	...	...

**NGƯỜI ĐẶT  
HÀNG**

**HÀNG**

## 4. Mô hình thực thể liên kết (ERD)

❖ Quy trình xây dựng ERD:

**Các bước thực hiện – Ví dụ: Xây dựng ERD cho Đơn đặt hàng**

- Bước 2: Xác định thực thể, thuộc tính
  - NGƯỜI ĐẶT HÀNG (MaNDH, TenNDH, DiaChi)
  - HÀNG HÓA(MaHH, TenHang, DonVi, DonGia)

Từ trong HSDL	Từ rõ nghĩa	Viết tắt
Số	Mã đơn đặt hàng	MaDDH
Người đặt hàng	Tên người đặt hàng	TenNDH
Địa chỉ	Địa chỉ	DiaChi
Ngày đặt	Ngày đặt	NgayDat
Tên hàng	Tên hàng	TenH
Đơn vị	Đơn vị	DonVi
Đơn giá	Đơn giá	DonGia
Số lượng	Số lượng	SoLuong
Thành tiền	Thành tiền	ThanhTien

## 4. Mô hình thực thể liên kết (ERD)

❖ Quy trình xây dựng ERD:

**Các bước thực hiện – Ví dụ: Xây dựng ERD cho Đơn đặt hàng**

- Bước 3: Xác định quan hệ

**ĐẶT HÀNG:** bao gồm các thuộc tính còn lại trong từ điển

- Bước 4: Vẽ



## 4. Mô hình thực thể liên kết (ERD)

❖ Quy trình xây dựng ERD:

**Các bước thực hiện – Ví dụ: Xây dựng ERD cho phiếu xuất kho**

Từ trong HS	Từ rõ nghĩa	Từ viết tắt
Số	Số hóa đơn xuất	SoHDX
Họ và tên người mua hàng	<b>Tên khách hàng</b>	Ten KH
Địa chỉ	Địa chỉ	Địa Chi
Lý do xuất kho	Lý do xuất kho	lyDo
Xuất tại kho	<b>Tên kho</b>	TenKho
Tên hàng	<b>Tên hàng</b>	TenHang
Đơn vị	Đơn vị	DVT
Đơn giá	Đơn giá	DonGiaBan
Thành tiền	Thành tiền	ThanhTien
Tổng tiền hàng	Tổng tiền hàng	TongTienH
Thuế VAT	Thuế VAT	VAT
Tổng cộng	Tổng cộng	TongCong

**PHIẾU XUẤT KHO**  
Số: XXXXXX

Họ và tên người mua hàng:.....

Địa chỉ:.....

Lý do xuất kho:.....

Xuất tại kho:.....

STT	Tên hàng	ĐVT	Số lượng	Đơn giá	Thành tiền

Tổng tiền hàng: .....

Thuế VAT: .....

Tổng cộng: .....

**NGƯỜI MUA HÀNG**

**KHO**

**HÀNG**



## 4. Mô hình thực thể liên kết (ERD)

❖ Quy trình xây dựng ERD:

**Các bước thực hiện – Ví dụ: Xây dựng ERD cho phiếu xuất kho**

Bước 2: Xác định thực thể

- NGƯỜI MUA HÀNG (MaNMH, TenKH, DiaChi)
- KHO (MaKho, TenKho)
- HÀNG HÓA (MaHH, TenHang, DonGiaBan, DVT)

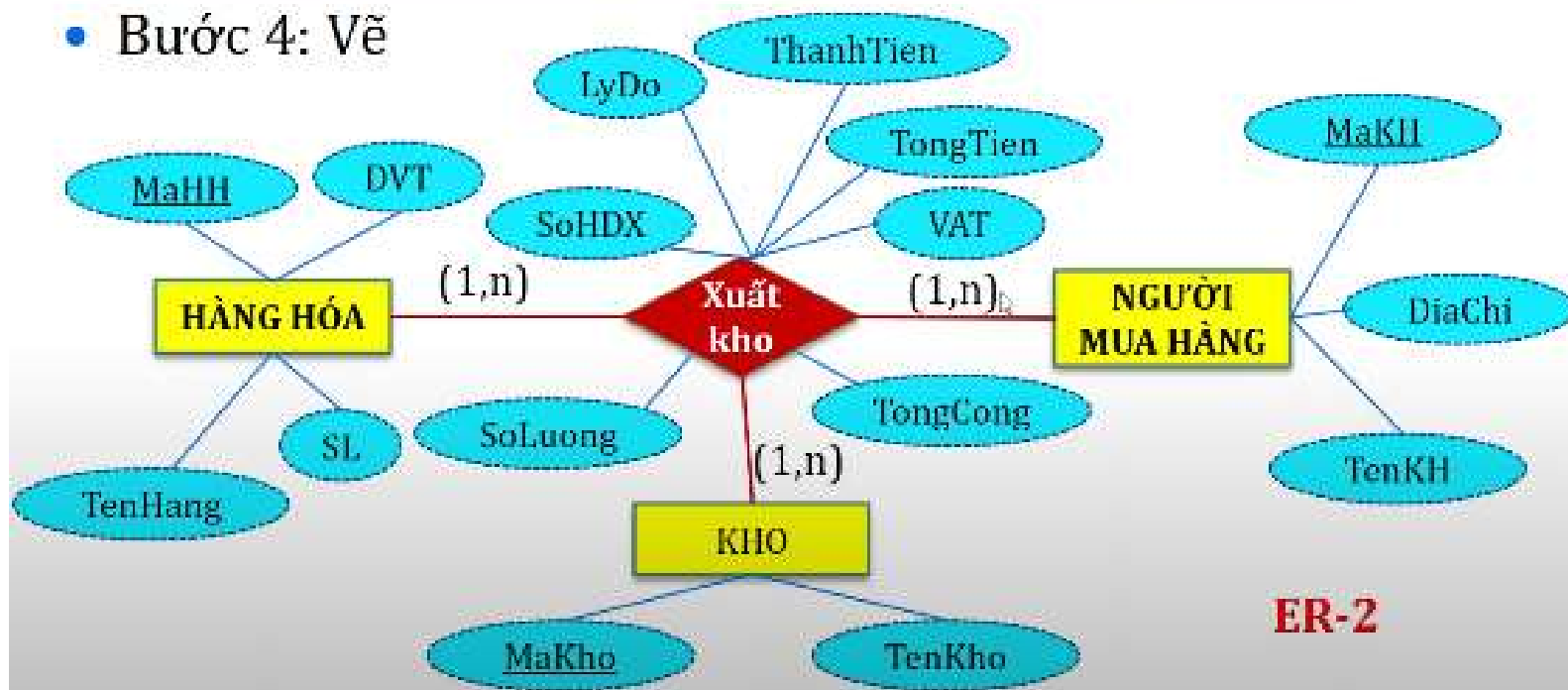
Từ trong HS	Từ rõ nghĩa	Từ viết tắt
Số	Số hóa đơn xuất	SoHDX
Họ và tên người mua hàng	Tên khách hàng	Ten KH
Địa chỉ	Địa chỉ	ĐịaChi
Lý do xuất kho	Lý do xuất kho	LyDo
Xuất tại kho	Tên kho	TenKho
Tên hàng	Tên hàng	TenHang
Đơn vị	Đơn vị	DVT
Đơn giá	Đơn giá	DonGiaBan
Thành tiền	Thành tiền	ThanhTien
Tổng tiền hàng	Tổng tiền hàng	TongTien
Thuế VAT	Thuế VAT	VAT
Tổng cộng	Tổng cộng	TongCong
Số lượng	Số lượng	SoLuong

## 4. Mô hình thực thể liên kết (ERD)

❖ Quy trình xây dựng ERD:

**Các bước thực hiện – Ví dụ: Xây dựng ERD cho phiếu xuất kho**

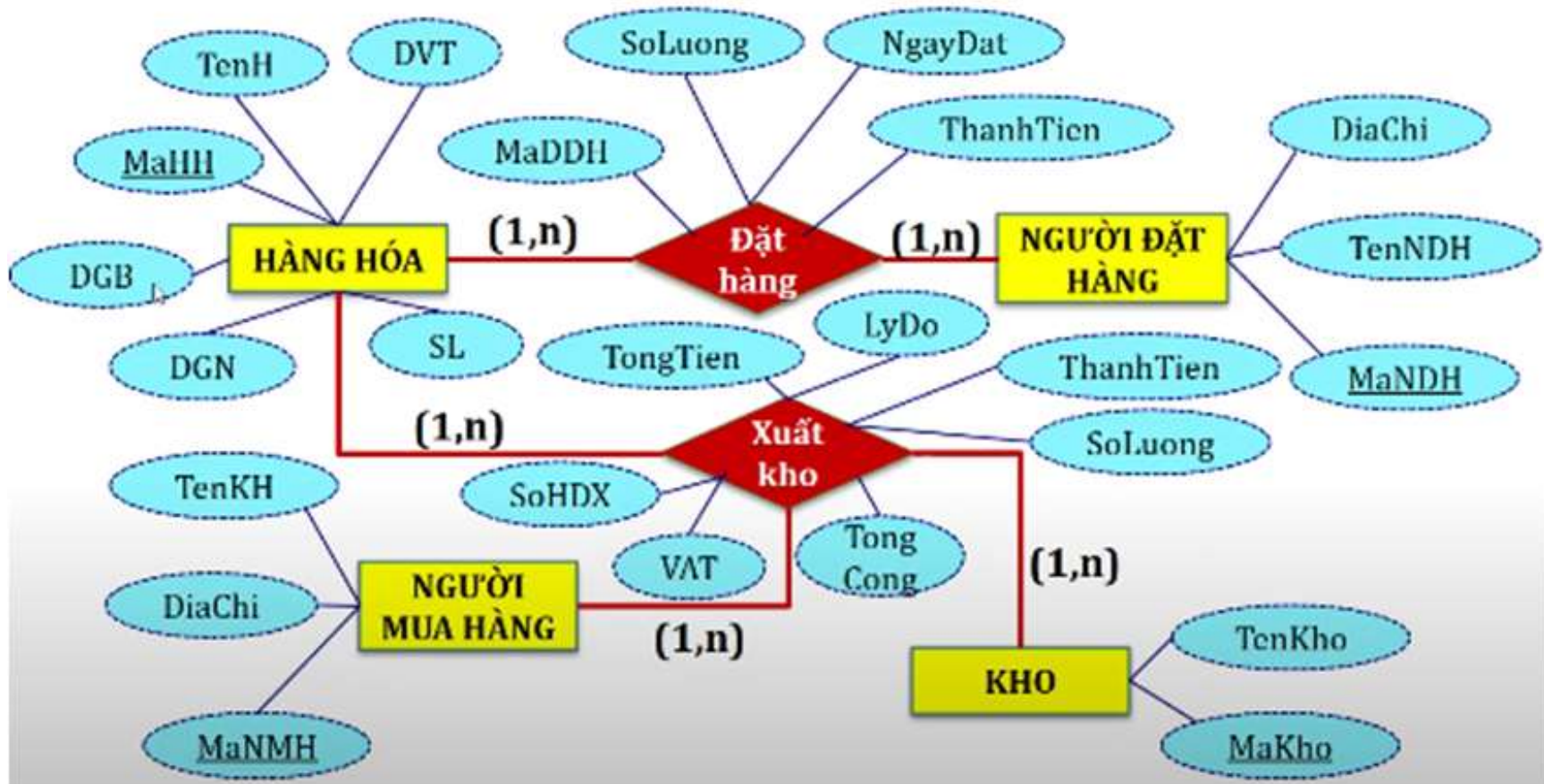
- Bước 3: Xác định quan hệ
  - XUẤT: bao gồm các thuộc tính còn lại trong từ điển
- Bước 4: Vẽ



## 4. Mô hình thực thể liên kết (ERD)

❖ Quy trình xây dựng ERD:

Các bước thực hiện – Ví dụ: Tích hợp ERD1 & ERD2





## 4. Mô hình thực thể liên kết (ERD)

### Các bước thiết kế - Tóm tắt

- Bước 1 - **Xác định các thực thể** từ CSDL đã cho.
- Bước 2 - **Tìm mối quan hệ** giữa các thực thể.
- Bước 3 - **Xác định thuộc tính khóa** cho mọi thực thể.
- Bước 4 - **Xác định thuộc tính liên quan khác**.
- Bước 5 - **Vẽ sơ đồ ERD hoàn chỉnh** với tất cả các thuộc tính bao gồm khóa chính.
- Bước 6 - **Xem lại kết quả theo yêu cầu** của CSDL.

### ❖ Các công cụ online giúp vẽ ERD

#### ● Lucidchart

- <https://www.lucidchart.com>

#### ● Drawio

- <https://www.draw.io/>

#### ● Một số mô hình ERD tham khảo

- [http://www.databaseanswERDs.org/data\\_models/index.htm](http://www.databaseanswERDs.org/data_models/index.htm)

## 4. Mô hình thực thể liên kết (ERD)

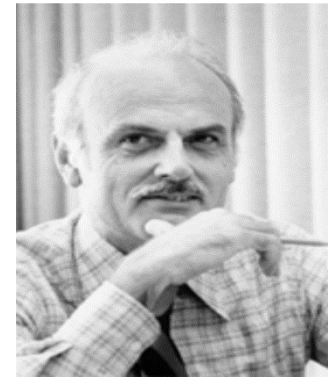
### ❖ Bài tập: Vẽ sơ đồ ERD

Phân tích và thiết kế 1 CSDL gồm các thông tin trong 1 công ty (nhân viên, phòng ban, dự án)

- Công ty được tổ chức bởi các phòng ban. Mỗi phòng ban có 1 tên duy nhất, 1 số duy nhất và 1 người quản lý (thời điểm bắt đầu công tác quản lý của người này cũng được lưu lại trong CSDL). Mỗi phòng ban có thể có nhiều trụ sở làm việc khác nhau
- Mỗi phòng điều phối một số dự án. Mỗi dự án có 1 tên và 1 mã số duy nhất, thực hiện tại một địa điểm duy nhất
- Các thông tin về nhân viên cần được quan tâm gồm: tên, số bảo hiểm, địa chỉ, lương, giới tính, ngày sinh. Mỗi nhân viên làm việc tại một phòng ban nhưng có thể tham gia nhiều dự án khác nhau. Những dự án này có thể được điều phối bởi các phòng ban khác nhau. Thông tin về số giờ làm việc trong từng dự án (theo tuần) cũng như người quản lý trực tiếp của các nhân viên cũng được lưu trữ
- Thông tin về con cái của từng nhân viên: tên, giới tính, ngày sinh

## 5. Mô hình dữ liệu quan hệ

- ❖ Ra đời: vào năm 1970 [Codd, 1970]
- ❖ Dữ liệu được biểu diễn dưới dạng bảng
- ❖ Là mô hình dữ liệu khái niệm phổ biến cho đến tận thời điểm hiện tại.
- ❖ Dựa trên lý thuyết toán học, đồng thời cũng gần với cấu trúc tệp và cấu trúc dữ liệu nên có hai loại thuật ngữ liên quan:
  - Thuật ngữ toán học: quan hệ, bộ, thuộc tính
  - Thuật ngữ hướng dữ liệu: bảng, bản ghi, trường
- ❖ Edgar Frank "Ted" Codd
  - (19/08/1923 – 18/04/2003)
  - Nhà khoa học máy tính người Anh, làm việc cho IBM.
  - Cha đẻ của mô hình dữ liệu quan hệ
  - Nhận giải thưởng Turing, năm 1981
  - Đề xuất **các quy tắc chuẩn hóa** sử dụng để thiết kế cơ sở dữ liệu, **các quy tắc** mà các nhà cung cấp phải đáp ứng để gọi sản phẩm của họ là hệ thống cơ sở dữ liệu quan hệ (1985).
  - Một số quy tắc (13 tiêu chí) xác định xem DB có phải là RDBMS chính xác hay không.
  - Một DB tuân theo quy tắc này được gọi là hệ thống quản lý DB quan hệ thực (RDBMS).



- [https://www.tutorialspoint.com/dbms/dbms\\_codds\\_rules.htm](https://www.tutorialspoint.com/dbms/dbms_codds_rules.htm)
- <https://www.geeksforgeeks.org/codds-rules-in-dbms/>
- <https://www.red-gate.com/simple-talk/databases/theory-and-design/codds-twelve-rules/>

## 5. Mô hình dữ liệu quan hệ

### ❖ 12 Quy tắc của Codd

#### ● Quy tắc 0: Quy tắc nền tảng

- DB phải được cấu trúc theo kiểu quan hệ để các khả năng quan hệ của hệ thống có thể quản lý DB.

#### ● Quy tắc 1: Quy tắc thông tin

- Một DB bao gồm nhiều loại dữ liệu, phải được ghi lại dưới dạng cột và hàng trong mỗi ô của bảng.

#### ● Quy tắc 2: Quy tắc truy cập được đảm bảo

- Giá trị khóa chính, tên cột và tên bảng của DB quan hệ có thể được sử dụng để truy xuất một cách khái niệm bất kỳ dữ liệu đơn lẻ hoặc chính xác nào (giá trị nguyên tử).

#### Codd's Rules

- 0 The Foundation Rule
- 1 Information Rule
- 2 Guaranteed Access Rule
- 3 Systematic Treatment of Null Values
- 4 Active/Dynamic Online Catalog based on the relational model
- 5 Comprehensive Data Sub-Language Rule
- 6 View Updating Rule
- 7 Relational Level Operation Rule
- 8 Physical Data Independence Rule
- 9 Logical Data Independence Rule
- 10 Integrity Independence Rule
- 11 Distribution Independence Rule
- 12 Non Subversion Rule

## 5. Mô hình dữ liệu quan hệ

### ❖ 12 Quy tắc của Codd

#### ● Quy tắc 3: Xử lý có hệ thống các giá trị rỗng (null)

- Việc xử lý các giá trị Null trong bản ghi DB được xác định theo quy tắc này. Không có giá trị trong một ô, dữ liệu bị thiếu, thông tin không phù hợp, dữ liệu không xác định, khóa chính không được rỗng, v.v., đều là ví dụ về giá trị null trong DB.

#### ● Quy tắc 4: Mục lục trực tuyến hoạt động/động dựa trên mô hình quan hệ

- Từ điển DB là sự biểu diễn logic của toàn bộ cấu trúc logic của DB mô tả cần được lưu trữ trực tuyến. Nó cấp cho người dùng quyền truy cập vào DB và sử dụng ngôn ngữ truy vấn tương đương với ngôn ngữ của DB.

#### Codd's Rules

- 0 The Foundation Rule
- 1 Information Rule
- 2 Guaranteed Access Rule
- 3 Systematic Treatment of Null Values
- 4 Active/Dynamic Online Catalog based on the relational model
- 5 Comprehensive Data Sub-Language Rule
- 6 View Updating Rule
- 7 Relational Level Operation Rule
- 8 Physical Data Independence Rule
- 9 Logical Data Independence Rule
- 10 Integrity Independence Rule
- 11 Distribution Independence Rule
- 12 Non Subversion Rule

## 5. Mô hình dữ liệu quan hệ

### ❖ 12 Quy tắc của Codd

#### ● Quy tắc 5: Quy tắc ngôn ngữ phụ dữ liệu toàn diện

- RDB hỗ trợ nhiều ngôn ngữ khác nhau và để truy cập vào DB, ngôn ngữ đó phải tuyến tính, rõ ràng hoặc có cú pháp, chuỗi ký tự được xác định rõ ràng. Nó phải hỗ trợ các hoạt động sau: định nghĩa khung nhìn, ràng buộc toàn vẹn, thao tác dữ liệu, định nghĩa dữ liệu cũng như quản lý giao dịch giới hạn.
- Sẽ bị coi là vi phạm DB nếu DB cho phép truy cập vào dữ liệu và thông tin mà không sử dụng bất kỳ ngôn ngữ nào.

#### ● Quy tắc 6: Quy tắc cập nhật chế độ xem

- Về mặt lý thuyết, một bảng xem có thể được cập nhật và hệ thống DB phải cập nhật chúng trong thực tế.

#### Codd's Rules

- 0 The Foundation Rule
- 1 Information Rule
- 2 Guaranteed Access Rule
- 3 Systematic Treatment of Null Values
- 4 Active/Dynamic Online Catalog based on the relational model
- 5 Comprehensive Data Sub-Language Rule
- 6 View Updating Rule
- 7 Relational Level Operation Rule
- 8 Physical Data Independence Rule
- 9 Logical Data Independence Rule
- 10 Integrity Independence Rule
- 11 Distribution Independence Rule
- 12 Non Subversion Rule



## 5. Mô hình dữ liệu quan hệ

### ❖ 12 Quy tắc của Codd

#### ● Quy tắc 7: Quy tắc thao tác cấp độ quan hệ (hoặc quy tắc chèn, xóa và cập nhật cấp cao)

- Ở mỗi cấp độ hoặc một hàng, hệ thống DB phải tuân thủ các hoạt động quan hệ cấp cao (ví dụ: cập nhật, chèn và xóa).
- Hệ thống DB cũng bao gồm các phép toán như giao, hợp và trừ.

#### ● Quy tắc 8: Độc lập dữ liệu vật lý

- Để truy cập DB hoặc ứng dụng, tất cả dữ liệu được lưu trữ phải độc lập về mặt vật lý. Mỗi phần dữ liệu không được phụ thuộc vào một phần dữ liệu hoặc ứng dụng khác.
- Các ứng dụng bên ngoài truy cập dữ liệu từ DB sẽ không có hiệu lực nếu dữ liệu bị thay đổi hoặc cấu trúc vật lý của DB nhất định bị sửa đổi.

#### Codd's Rules

- 0 The Foundation Rule
- 1 Information Rule
- 2 Guaranteed Access Rule
- 3 Systematic Treatment of Null Values
- 4 Active/Dynamic Online Catalog based on the relational model
- 5 Comprehensive Data Sub-Language Rule
- 6 View Updating Rule
- 7 Relational Level Operation Rule
- 8 Physical Data Independence Rule
- 9 Logical Data Independence Rule
- 10 Integrity Independence Rule
- 11 Distribution Independence Rule
- 12 Non Subversion Rule

## 5. Mô hình dữ liệu quan hệ

### ❖ 12 Quy tắc của Codd

#### ● Quy tắc 9: Độc lập dữ liệu logic

- Chỉ ra rằng mọi sửa đổi được thực hiện ở cấp độ logic (hoặc cấu trúc bảng) sẽ không có tác động đến trải nghiệm (ứng dụng) của người dùng.
- Ví dụ: nếu một bảng được chia thành hai bảng riêng biệt hoặc thành hai bảng nối để tạo ra một bảng duy nhất thì ứng dụng ở chế độ xem người dùng sẽ không bị ảnh hưởng.

#### ● Quy tắc 10: Độc lập chính trực

- Khi sử dụng SQL để đưa dữ liệu vào các ô trong bảng, DB phải đảm bảo độc lập về tính toàn vẹn.
- Tất cả các giá trị đã nhập không được thay đổi và tính toàn vẹn của dữ liệu không được phụ thuộc vào bất kỳ thành phần hoặc ứng dụng bên ngoài nào.
- Nó cũng hữu ích để làm cho mỗi ứng dụng giao diện người dùng độc lập với DB.

#### Codd's Rules

- 0 The Foundation Rule
- 1 Information Rule
- 2 Guaranteed Access Rule
- 3 Systematic Treatment of Null Values
- 4 Active/Dynamic Online Catalog based on the relational model
- 5 Comprehensive Data Sub-Language Rule
- 6 View Updating Rule
- 7 Relational Level Operation Rule
- 8 Physical Data Independence Rule
- 9 Logical Data Independence Rule
- 10 Integrity Independence Rule
- 11 Distribution Independence Rule
- 12 Non Subversion Rule



## 5. Mô hình dữ liệu quan hệ

### ❖ 12 Quy tắc của Codd

#### ● Quy tắc 11: Độc lập phân phối

- DB phải hoạt động bình thường ngay cả khi nó được lưu trữ ở nhiều vị trí và được nhiều người dùng cuối khác nhau sử dụng.
- Giả sử một người sử dụng ứng dụng để truy cập DB. Trong trường hợp như vậy, họ không được biết rằng một người dùng khác đang sử dụng cùng một dữ liệu và do đó, dữ liệu họ luôn thu được chỉ có sẵn trên một trang web.
- Người dùng cuối có thể truy cập DB và dữ liệu truy cập của mỗi người dùng phải độc lập để họ có thể chạy các truy vấn SQL.

#### ● Quy tắc 12: Quy tắc không lật đổ

- RDBMS được quy tắc này xác định là ngôn ngữ SQL để lưu trữ và thao tác dữ liệu trong DB.
- Nếu một hệ thống sử dụng ngôn ngữ cấp thấp hoặc ngôn ngữ khác để truy cập hệ thống cơ sở dữ liệu không phải là SQL thì hệ thống đó không được bỏ qua hoặc phá hoại tính toàn vẹn của dữ liệu.

#### Codd's Rules

- 0 The Foundation Rule
- 1 Information Rule
- 2 Guaranteed Access Rule
- 3 Systematic Treatment of Null Values
- 4 Active/Dynamic Online Catalog based on the relational model
- 5 Comprehensive Data Sub-Language Rule
- 6 View Updating Rule
- 7 Relational Level Operation Rule
- 8 Physical Data Independence Rule
- 9 Logical Data Independence Rule
- 10 Integrity Independence Rule
- 11 Distribution Independence Rule
- 12 Non Subversion Rule

## 5. Mô hình dữ liệu quan hệ

- ❖ Cung cấp cấu trúc dữ liệu đơn giản, đồng bộ  
Khái niệm “Quan hệ”
- ❖ Nền tảng lý thuyết vững chắc:  
Lý thuyết tập hợp
- ❖ Là cơ sở của hầu hết DBMS thương mại  
Oracle, DB2, SQL Server, ...  
Các HQT CSDL mạnh hiện nay như MS SQL, Oracle... đều được cài đặt dựa trên lý thuyết của mô hình DLQH

Các khái niệm:

- a) Quan hệ (Relation)
- b) Thuộc tính (Attribute)
- c) Lược đồ (Schema)
- d) Bộ (Tuple)
- e) Miền giá trị (Domain)

## 5. Mô hình dữ liệu quan hệ

Các khái niệm:

### ● QUAN HỆ

- Các thông tin lưu trữ trong CSDL được tổ chức thành bảng (table) gọi là quan hệ.
- Mỗi quan hệ là một bảng 2 chiều, mô tả một thực thể

1 cột là 1 thuộc tính của nhân viên

TENNV	HONV	NS	DIACHI	GT	LUONG	PHG
Tung	Nguyen	12/08/195	638 NVC Q5	Nam	40000	5
Hang	Bui	07/19/196	332 NTH Q1	Nu	25000	4
Nhu	Le	06/20/195	291 HVH QPN	Nu	43000	4
Hung	Nguyen	09/15/196	Ba Ria VT	Nam	38000	5

1 dòng là 1 nhân viên

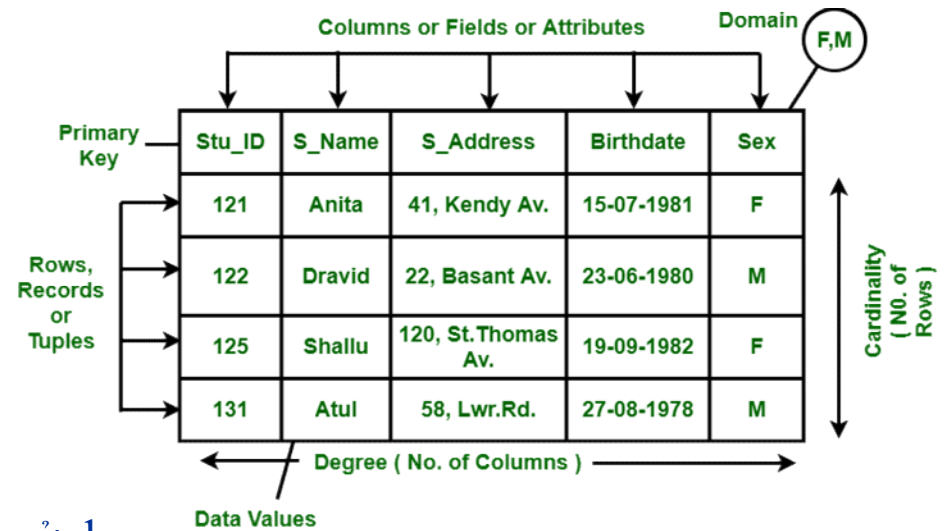
Tên quan hệ là NHANVIEN

## 5. Mô hình dữ liệu quan hệ

Các khái niệm:

QUAN HỆ: Quan hệ R gồm:

- Tập hợp các cột
  - Cố định, Được đặt tên
  - Có kiểu dữ liệu
- Tập hợp các dòng
  - Thay đổi theo thời gian.
  - Sự thay đổi phụ thuộc vào người sử dụng
- Mỗi dòng ~ Một thực thể (đối tượng ở thế giới thực)
- Quan hệ ~ Tập các thực thể



BỘ

<Tung, Nguyen, 12/08/1955, 638 NVC, Q5, Nam, 40000, 5>

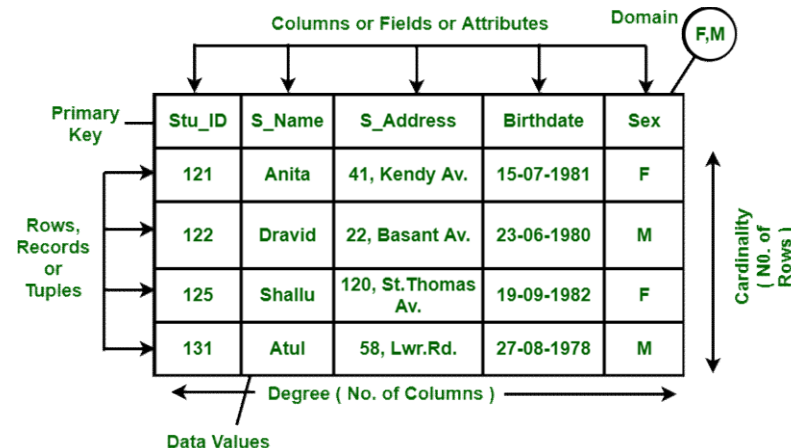
Dữ liệu cụ thể của thuộc tính

- Là các dòng của quan hệ (trừ dòng tiêu đề - tên của các thuộc tính)
- Thể hiện dữ liệu cụ thể của các thuộc tính trong quan hệ

## 5. Mô hình dữ liệu quan hệ

### Các khái niệm cơ bản

- **Quan hệ** (bảng): Cho  $n$  miền giá trị  $D_1, D_2, \dots, D_n$  không nhất thiết phân biệt,  $r$  là một quan hệ trên  $n$  miền giá trị đó nếu  $r$  là một tập các bộ  $n$  ( $d_1, d_2, \dots, d_n$ ) sao cho  $d_i \in D_i$
- Một **quan hệ** có thể được biểu diễn dưới dạng 1 **bảng** trong đó 1 **dòng** trong bảng tương đương với 1 **bộ**, một **cột** trong bảng tương đương với 1 **thuộc tính** của quan hệ
- **Bậc** của 1 quan hệ là số các thuộc tính trong quan hệ (Degree)
- **Số lượng** (lực lượng) của 1 quan hệ là số các bộ trong quan hệ (cardinality)



## 5. Mô hình dữ liệu quan hệ

### Các khái niệm cơ bản

#### THUỘC TÍNH (trường):

- Tên các cột của quan hệ, mô tả ý nghĩa cho các giá trị tại cột đó
- Là các đặc tính của một đối tượng
- Mỗi thuộc tính được xác định trên một miền giá trị nhất định gọi là **miền thuộc tính** (tất cả dữ liệu trong cùng 1 cột đều có cùng kiểu dữ liệu)
- Ví dụ:

○ STUDENT (Stu\_ID, S\_Name, S\_Address, Birthdate, Sex)

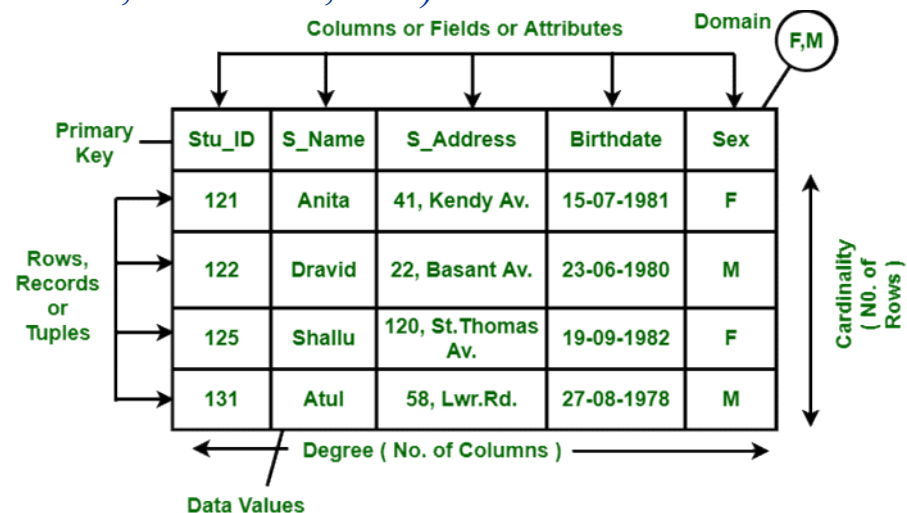
○  $\text{dom}(\text{Stu\_ID}) = \{\text{char}(5)\}$

○  $\text{dom}(\text{S\_Name}) = \{\text{char}(30)\}$

○  $\text{dom}(\text{Birthdate}) = \{\text{date}\}$

○  $\text{dom}(\text{Sex}) = \{0, 1\}$

○  $\text{dom}(\text{S\_Address}) = \{\text{char}(50)\}$





## 5. Mô hình dữ liệu quan hệ

Các khái niệm:

MIỀN THUỘC TÍNH:

- Là tập các giá trị nguyên tố gắn liền với một thuộc tính
    - Kiểu dữ liệu cơ sở
      - ✦ Chuỗi ký tự (string)
      - ✦ Số (integer)
      - ✦ Thời gian
    - Các kiểu dữ liệu phức tạp
      - ✦ Tập hợp (set)
      - ✦ Danh sách (list)
      - ✦ Mảng (array)
      - ✦ Bản ghi (record)
- } **Không được chấp nhận**
- Ví dụ
    - TENNV: string
    - LUONG: integer

## 5. Mô hình dữ liệu quan hệ

Các khái niệm:

MIỀN THUỘC TÍNH:

### ***CÁC KIỂU DỮ LIỆU***

❖ Kiểu chuỗi

CÚ PHÁP	KÍCH THƯỚC	DIỄN GIẢI
CHAR(kích_thuoc)	Tối đa 8000 kí tự.	<ul style="list-style-type: none"><li>• kích_thuoc là số kí tự lưu trữ.</li><li>• Độ dài cố định.</li><li>• Thêm dấu cách về bên phải để bù phần trống cho đủ số kí tự.</li><li>• Không chứa kí tự Unicode.</li></ul>
VARCHAR(kích_thuoc) hoặc VARCHAR(toi_da)	Tối đa 8000 kí tự hoặc theo số tối đa.	<ul style="list-style-type: none"><li>• Độ dài tùy biến.</li><li>• Nếu chỉ định là toi_da thì tối đa là 2GB.</li><li>• Không chứa kí tự Unicode.</li></ul>
NCHAR(kích_thuoc)	Tối đa 4000 kí tự.	<ul style="list-style-type: none"><li>• Giống Char nhưng có kí tự Unicode.</li></ul>
NVARCHAR(kích_thuoc) hoặc NVARCHAR(toi_da)	Tối đa 4000 kí tự hoặc theo số tối đa.	<ul style="list-style-type: none"><li>• Giống Varchar nhưng có kí tự Unicode.</li></ul>



## 5. Mô hình dữ liệu quan hệ

Các khái niệm:

MIỀN THUỘC TÍNH:

❖ Kiểu số

CÚ PHÁP	KÍCH THƯỚC	DIỄN GIẢI
BIT	số nguyên 0, 1 hoặc NULL	
TINYINT	từ 0 đến 255	
SMALLINT	từ -32768 đến 32767	
INT	<b>-2,147,483,648 đến 2,147,483,647</b>	
BIGINT	từ -9,223,372,036,854,775,808 đến 9,223,372,036,854,775,807	
DECIMAL(m,d)	<ul style="list-style-type: none"> <li>m mặc định là 18 nếu không được chỉ định cụ thể.</li> <li>d mặc định là 0 nếu không được chỉ định cụ thể.</li> </ul>	m là tổng số lượng các số còn d là số lượng các số nằm sau dấu phẩy.
FLOAT(n)	số dấu phẩy động n mặc định là 53 nếu không được chỉ định cụ thể.	n là số lượng của số bit lưu trữ trong một kí hiệu hóa học.

## 5. Mô hình dữ liệu quan hệ

Các khái niệm:

MIỀN THUỘC TÍNH:

❖ Kiểu ngày tháng, thời gian

CÚ PHÁP	KÍCH THƯỚC	DIỄN GIẢI
DATE	giá trị từ '0001-01-01' đến '9999-12-31'.	hiển thị dưới dạng 'YYYY-MM-DD'
DATETIME	<ul style="list-style-type: none"> <li>Ngày lấy từ '1753-01-01 00:00:00' to '9999-12-31 23:59:59'.</li> <li>Giờ lấy từ '00:00:00' to '23:59:59:997'</li> </ul>	hiển thị dưới dạng 'YYYY-MM-DD hh:mm:ss[.mmm]
DATETIME2(chính xác tới số thập phân của giây)	<ul style="list-style-type: none"> <li>giá trị lấy từ '0001-01-01' đến '9999-12-31'.</li> <li>Thời gian lấy từ '00:00:00' đến '23:59:59:9999999'.</li> </ul>	hiển thị dưới dạng 'YYYY-MM-DD hh:mm:ss[số giây thập phân]'
SMALLDATETIME	<ul style="list-style-type: none"> <li>giá trị lấy từ '1900-01-01' đến '2079-06-06'.</li> <li>Thời gian lấy từ '00:00:00' đến '23:59:59'.</li> </ul>	hiển thị dưới dạng 'YYYY-MM-DD hh:mm:ss'

## 5. Mô hình dữ liệu quan hệ

Các khái niệm:

MIỀN THUỘC TÍNH:

❖ Kiểu ngày tháng, thời gian

CÚ PHÁP	KÍCH THƯỚC	DIỄN GIẢI
TIME	<ul style="list-style-type: none"><li>• giá trị lấy từ '00:00:00.0000000' đến '23:59:59.9999999'.</li><li>• Ngày lấy từ '0001-01-01' đến '9999-12-31'.</li></ul>	hiển thị dưới dạng 'YYYY-MM-DD hh:mm:ss[.nnnnnnnn]
DATETIMEOFFSET (chính xác tới số thập phân của giây)	<ul style="list-style-type: none"><li>• giá trị thời gian lấy từ '00:00:00' đến '23:59:59.9999999'.</li><li>• Múi giờ lấy từ -14:00 đến +14:00.</li></ul>	hiển thị dưới dạng YYYY-MM-DD hh:mm:ss[.nnnnnnnn] [ {+ -} }hh:mm]

## 5. Mô hình dữ liệu quan hệ

Các khái niệm:

### Tính chất của quan hệ

- Các bộ trong một quan hệ là duy nhất
- Không quan tâm đến thứ tự các bộ trong quan hệ
- Thứ tự các giá trị trong bộ phải tương ứng với thứ tự liệt kê các thuộc tính của quan hệ

Bộ <Nguyễn, Tung, 12/08/1955, 638 NVC Q5, Nam, 40000, 5>

khác

Bộ <Nguyễn, Tung, 12/08/1955, 638 NVC Q5, 40000, Nam, 5>

- Một thuộc tính có thể mang giá trị rỗng (null)

### Lược đồ quan hệ

- Tên của quan hệ
- Tên của tập thuộc tính

Lược đồ quan hệ

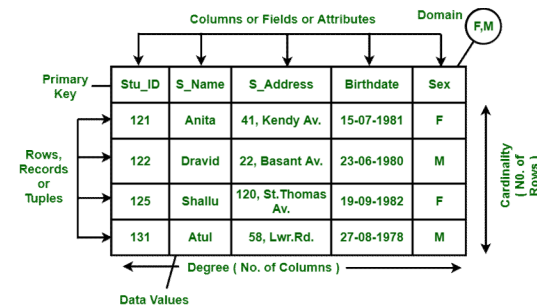
**NHANVIEN(MANV, TENNV, HONV, NS, DIACHI, GT, LUONG, PHG)**

Là tập hợp

## 5. Mô hình dữ liệu quan hệ

### Các khái niệm cơ bản

- **Định nghĩa** (tiếp): Cho  $U = \{A_1, A_2, \dots, A_n\}$  là một tập hữu hạn các thuộc tính trong đó  $\text{dom}(A_i) = D_i$ ,  $r$  là quan hệ trên tập thuộc tính  $U$  ký hiệu là  $r(U)$  nếu:
  - $r \subseteq D_1 \times D_2 \times \dots \times D_n$
 →  $U$  được gọi là sơ đồ quan hệ (lược đồ quan hệ)
- Định nghĩa **Khóa** của quan hệ  $r$  trên tập thuộc tính  $U = \{A_1, A_2, \dots, A_n\}$  là một tập  $K \subseteq U$  sao cho với bất kỳ 2 bộ  $t_1, t_2$  thuộc  $r$  đều tồn tại một thuộc tính  $A$  thuộc  $K$  mà  $t_1[A] \neq t_2[A]$ 
  - Một quan hệ có thể có nhiều khóa
  - Nếu  $K$  là khóa của  $r$  thì mọi  $K'$  sao cho  $K \subseteq K'$  đều là khóa của  $r$ .  $K'$  được gọi là **siêu khóa** của  $r$



## 5. Mô hình dữ liệu quan hệ

### Các khái niệm cơ bản

- Định nghĩa: K là **khóa tối thiểu** của r nếu K là một khoá của r và bất kỳ tập con thực sự nào của K đều không phải là khoá của r
- Định nghĩa: Một tập con  $K \subseteq U$  được gọi là **khóa ngoài** của quan hệ r(U) tham chiếu đến một quan hệ r' nếu K là khóa chính của r'

Ví dụ:

- Quan hệ: SINH\_VIEN, LOP
- Siêu khóa: {MaSV, TenSV};
- Khóa tối thiểu: {MaSV}; {TenSV, NgaySinh}
- Khóa ngoài: Tenlop nếu coi nó là khóa chính của quan hệ LOP

SINH\_VIEN

MaSV	TenSV	NgaySinh	GioiTinh	DiaChi	MaLop
SV0001	Trần T. Bình	1/4/2002	Nam	Hà Nội	VT1
SV0002	Ng. Đ. Trung	23/2/2002	Nam	Hà Nội	VT1
SV0003	Trần M. Quế	12/9/2002	Nữ	Thái Nguyên	VT3
SV0004	Ng. T. Phương	7/5/2002	Nữ	Bắc Ninh	VT4

LOP

MaLop	TenLop	Khoa	GVCN	LopTruong
VT1	Viễn Thông 1	VT	Nguyễn Văn A	Trần A
VT2	Viễn Thông 2	VT	Nguyễn Văn A	Trần B
VT3	Viễn Thông 3	VT	Nguyễn Văn A	Trần C
VT4	Viễn Thông 4	VT	Nguyễn Văn A	Nguyễn D



## 5. Mô hình dữ liệu quan hệ

### Các khái niệm cơ bản

#### KHÓA CHÍNH

- Loại khóa có ít thuộc tính nhất làm khóa chính.
- Giá trị các thuộc tính của khóa chính phải khác null.
- Ký hiệu: gạch dưới thuộc tính khóa chính
- Chọn: khóa chính = MASV (là giá trị ít thuộc tính nhất và tuân theo quy luật nào đó, không phải CMND vì thuộc tính này do 1 tổ chức khác sinh ra nên khó nắm bắt)

Xét quan hệ r:

Khóa

MASV	HOTEN	NGSINH	DCHI	CMND
10001111	Trần Thanh	01/05/1990	Q01	024555886
12345678	Thu Huyền	23/12/1989	Q10	024000555
88866611	Nguyễn Tùng	12/07/1990	Q07	024678678
10002222	Trần Thanh	05/04/1989	Q01	024123456

Khóa Chính

MASV	HOTEN	NGSINH	DCHI	CMND
10001111	Trần Thanh	01/05/1990	Q01	024555886
12345678	Thu Huyền	23/12/1989	Q10	024000555
88866611	Nguyễn Tùng	12/07/1990	Q07	024678678
10002222	Trần Thanh	05/04/1989	Q01	024123456

## 5. Mô hình dữ liệu quan hệ

### Các khái niệm cơ bản

#### THAM CHIẾU (thứ tự nhập liệu)

- ❖ Một bộ trong quan hệ R, tại thuộc tính A nếu nhận một giá trị từ một thuộc tính B của quan hệ S, ta gọi R tham chiếu S
- ❖ Bộ được tham chiếu phải tồn tại trước

S

TENPHG	MAPHG
Nghien cuu	5
Dieu hanh	4
Quan ly	1

R

TENNV	HONV	NS	DCHI	GT	LUONG	PHG
Tung	Nguyen	12/08/1955	638 NVC Q5	Nam	40000	5
Hang	Bui	07/19/1968	332 NTH Q1	Nu	25000	4
Nhu	Le	06/20/1951	291 HVH QPN	Nu	43000	4
Hung	Nguyen	09/15/1962	Ba Ria VT	Nam	38000	5



## 5. Mô hình dữ liệu quan hệ

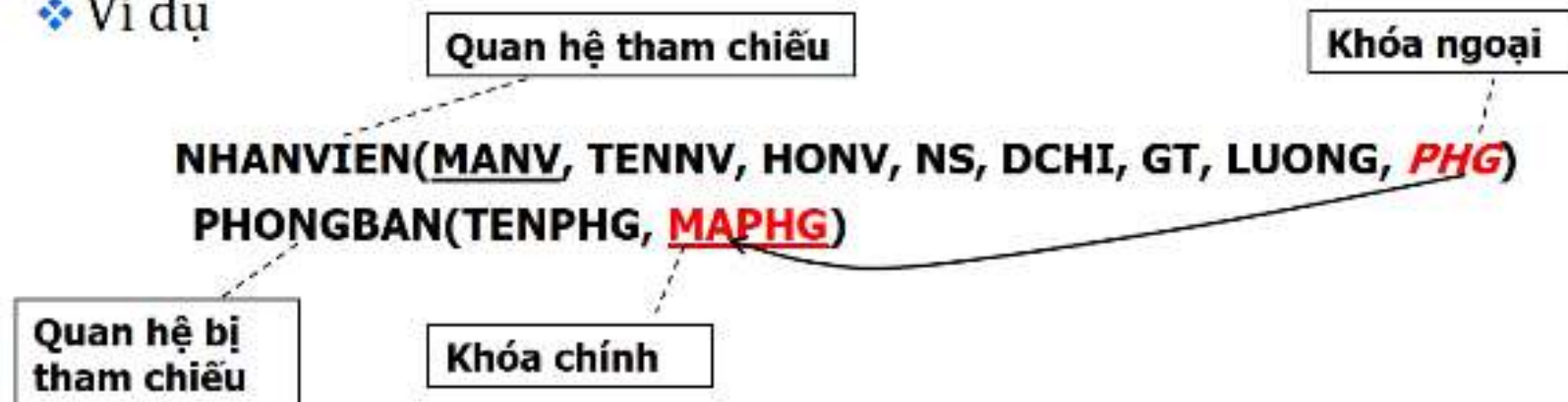
### Các khái niệm cơ bản

#### KHÓA NGOẠI

##### ❖ Xét 2 lược đồ R và S

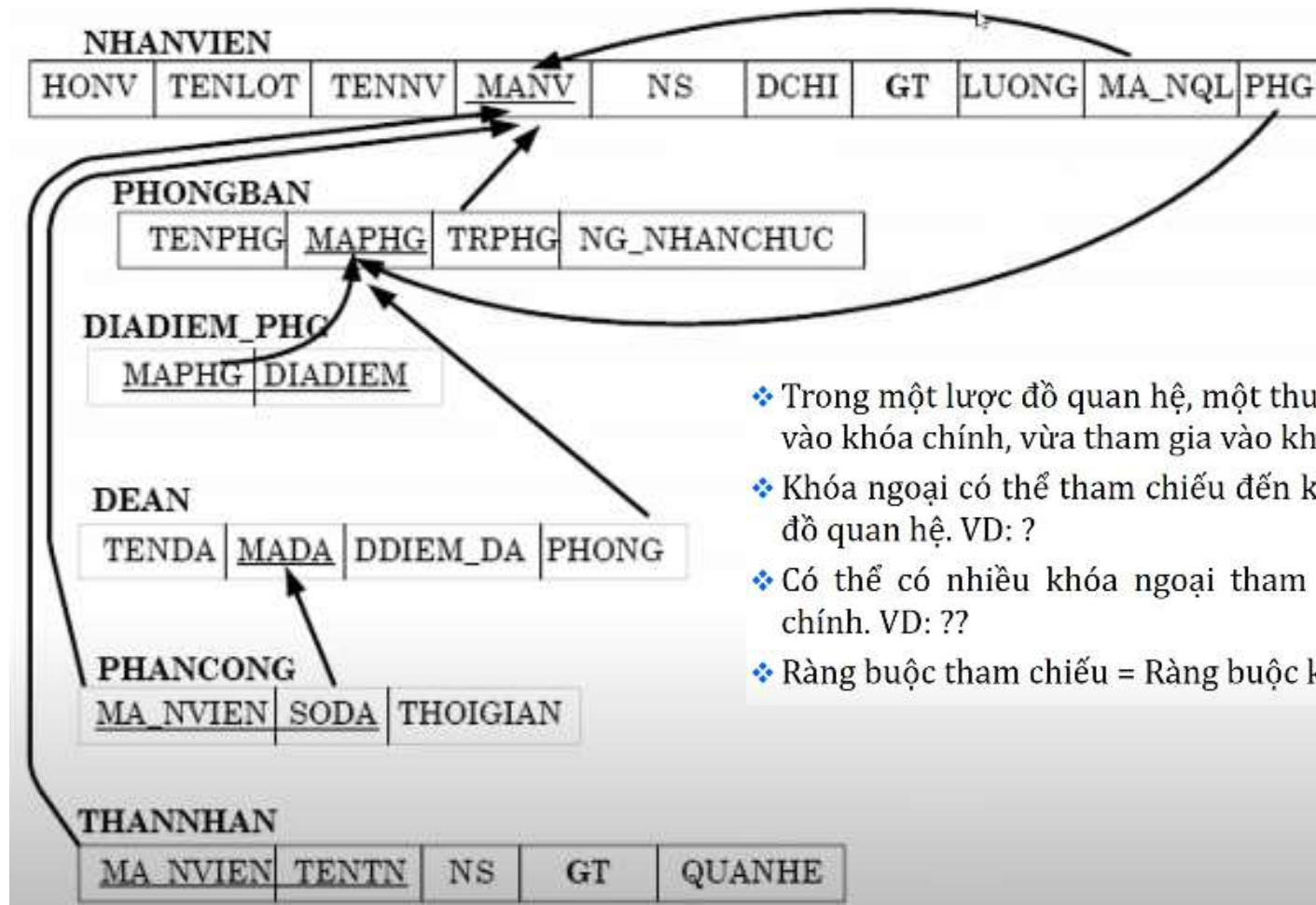
- Gọi FK là tập thuộc tính khác rỗng của R
- FK là khóa ngoại (Foreign Key) của R khi
  - ✦ Các thuộc tính trong FK phải có cùng miền giá trị với các thuộc tính khóa chính của S
  - ✦ Giá trị tại FK của một bộ  $t_1 \in R$ 
    - Hoặc bằng giá trị tại khóa chính của một bộ  $t_2 \in S$
    - Hoặc bằng giá trị rỗng

##### ❖ Ví dụ



## 5. Mô hình dữ liệu quan hệ

### Các khái niệm cơ bản KHÓA NGOẠI



- ❖ Trong một lược đồ quan hệ, một thuộc tính vừa có thể tham gia vào khóa chính, vừa tham gia vào khóa ngoại
- ❖ Khóa ngoại có thể tham chiếu đến khóa chính trên cùng 1 lược đồ quan hệ. VD: ?
- ❖ Có thể có nhiều khóa ngoại tham chiếu đến cùng một khóa chính. VD: ??
- ❖ Ràng buộc tham chiếu = Ràng buộc khóa ngoại

## 5. Mô hình dữ liệu quan hệ

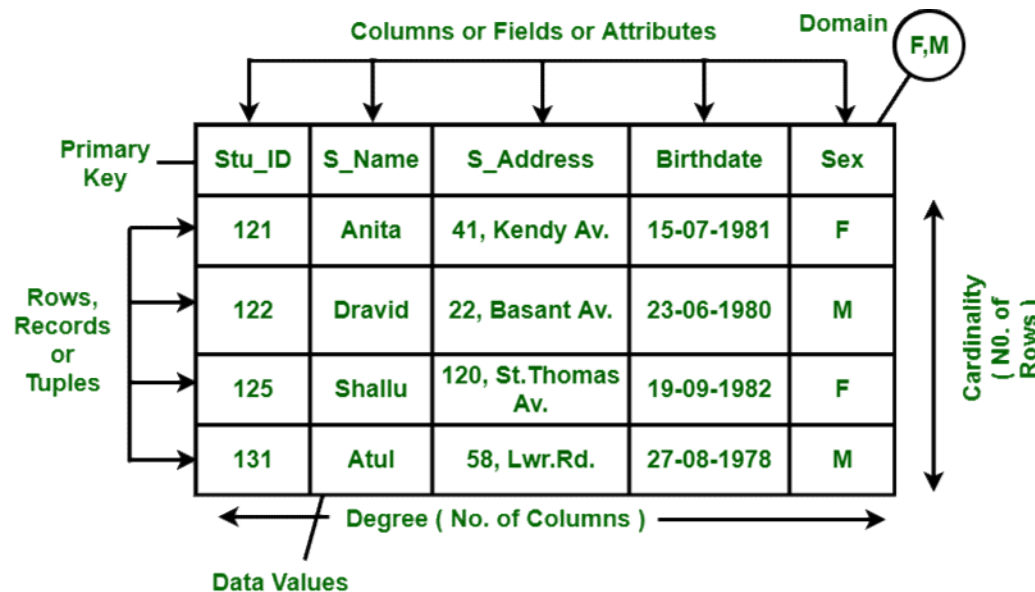
Nhận xét

- Ưu điểm

- Dựa trên lý thuyết tập hợp
- Khả năng tối ưu hoá các xử lý phong phú

- Nhược điểm

- Hạn chế trong biểu diễn ngữ nghĩa
- Cấu trúc dữ liệu không linh hoạt

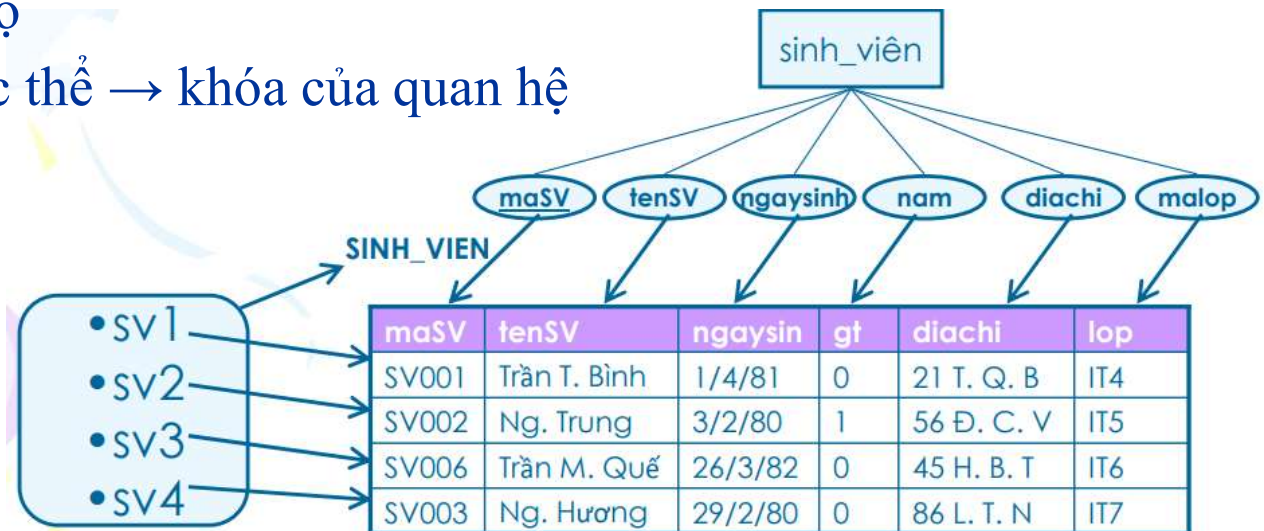


## 6. Chuyển đổi ERD sang RDM

- ❖ Biến đổi tập các thực thể
- ❖ Biến đổi các liên kết
- ❖ Các khoá của các sơ đồ quan hệ
- ❖ Các sơ đồ quan hệ với khoá chung

### CÁC BƯỚC THỰC HIỆN

- Bước 1: 1 tập thực thể 1 quan hệ
  - thuộc tính → thuộc tính (trường)
  - 1 thực thể → 1 bộ
  - khoá của tập thực thể → khóa của quan hệ



## 6. Chuyển đổi ERD sang RDM

- ❖ Biến đổi tập các thực thể
- ❖ Biến đổi các liên kết
- ❖ Các khoá của các sơ đồ quan hệ
- ❖ Các sơ đồ quan hệ với khoá chung

### CÁC BƯỚC THỰC HIỆN

- Bước 2: 1 tập thực thể xác định từ tập thực thể khác (E) qua 1 liên kết → 1 quan hệ chứa khoá của E:

- LOPTRUONG(maSV)



- Bước 3: Liên kết 1-1

- Thêm 1 quan hệ mới xác định bởi các thuộc tính nằm trong khóa của các thực thể có liên quan

- CHU\_NHIEM\_LOP(malop,maGV)

- Hoặc *Dùng khóa ngoài*

- LOP\_HOC(malop,lop,khoa,maGV)



## 6. Chuyển đổi ERD sang RDM

- ❖ Biến đổi tập các thực thể
- ❖ Biến đổi các liên kết
- ❖ Các khoá của các sơ đồ quan hệ
- ❖ Các sơ đồ quan hệ với khoá chung

### CÁC BƯỚC THỰC HIỆN

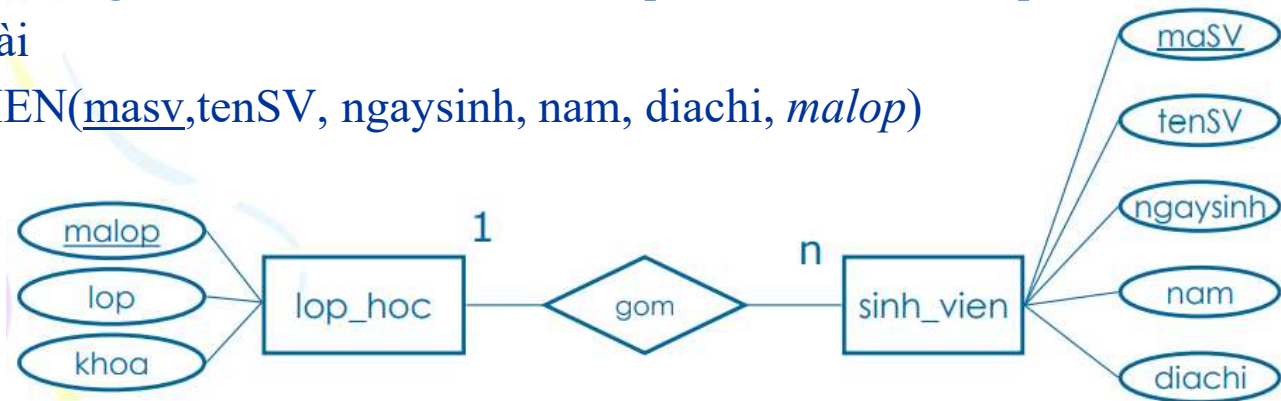
#### ● Bước 4: Liên kết 1-n

- Thêm 1 quan hệ mới xác định bởi các thuộc tính nằm trong khóa của các thực thể có liên quan

■ **SINHVIEN\_LOP**(malop,maGV)

- **Hoặc Dùng khóa ngoại:** thêm khóa chính của quan hệ bên 1 vào quan hệ bên n làm khóa ngoài

■ **SINH\_VIEN**(masv,tenSV, ngaysinh, nam, diachi, *malop*)



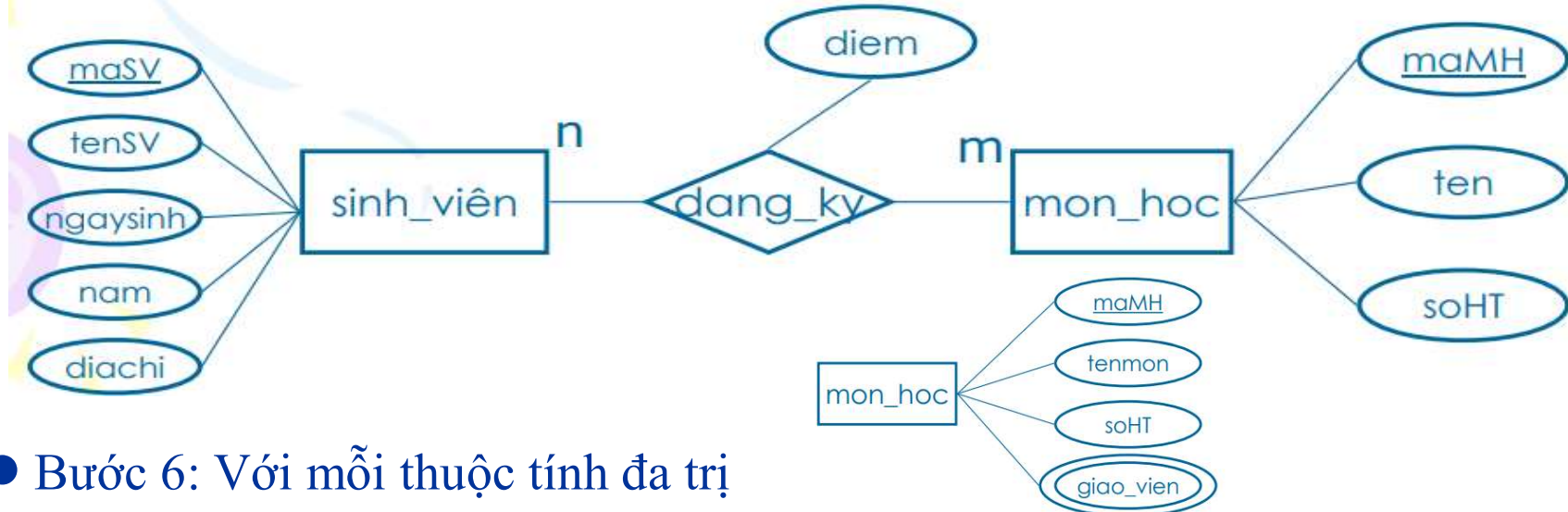


## 6. Chuyển đổi ERD sang RDM

### CÁC BƯỚC THỰC HIỆN

#### ● Bước 5: Liên kết n-n

- Thêm 1 quan hệ mới xác định bởi các thuộc tính nằm trong khóa của các thực thể có liên quan và các thuộc tính của liên kết
- DANG\_KY(maSV, maMH, diem)

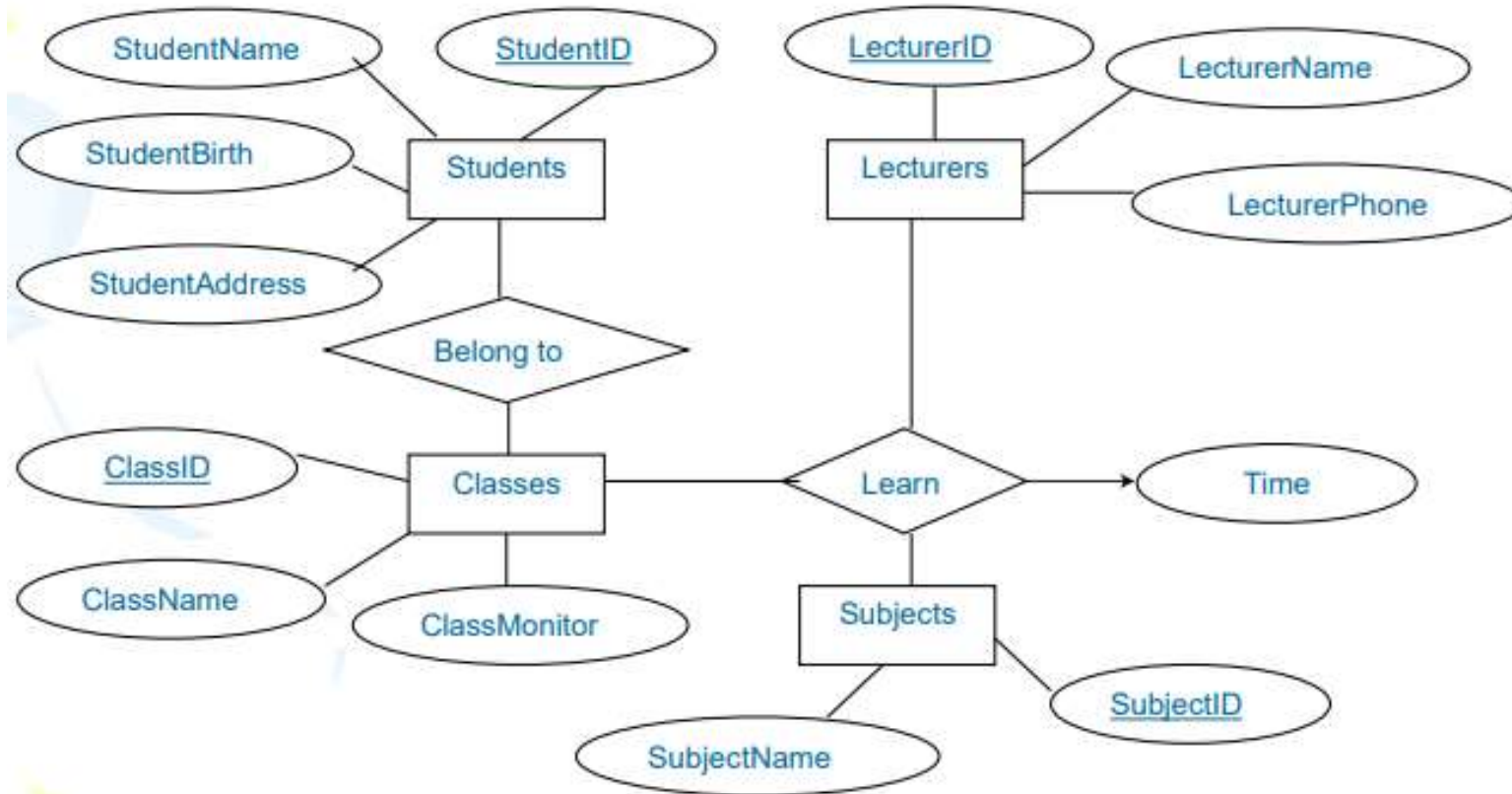


#### ● Bước 6: Với mỗi thuộc tính đa trị

- Thêm 1 quan hệ mới xác định bởi thuộc tính đa trị và khoá của tập thực thể tương ứng
- MH\_GV(maMH, giao\_vien)

## 6. Chuyển đổi ERD sang RDM

Bài tập: Cho sơ đồ thực thể liên kết bên dưới, hãy biến đổi sang mô hình quan hệ:





## 7. Mô hình dữ liệu hướng đối tượng

- Ra đời
  - Khoảng đầu những năm 90
- Biểu diễn: sơ đồ lớp
- Các khái niệm cơ bản
  - **Đối tượng**: một đối tượng trong thế giới thực, được xác định bởi một định danh duy nhất
  - **Thuộc tính**: biểu diễn một đặc tính của đối tượng,
  - **Phương thức** : thao tác được thực hiện trên đối tượng.
    - Tất cả các truy nhập vào thuộc tính của đối tượng đều phải được thực hiện thông qua các phương thức này.
  - **Lớp**: một cách thức để khai báo một tập các đối tượng có chung một tập thuộc tính và phương thức
- Ví dụ: class sinh\_vien

```
class sinh_vien {  
    string maSV;  
    string tenSV;  
    date ngaysinh;  
    boolean nam;  
    string diachi;  
    string lop;  
  
    string ten();  
    string ngay_sinh();  
    string dia_chi();  
    string lop();  
    void gan_DC(string DC_moi);  
    void gan_lop(string lop);  
}
```

## 7. Mô hình dữ liệu hướng đối tượng

Nhận xét về Mô hình dữ liệu hướng đối tượng

### ● Ưu điểm

- Cho phép định nghĩa kiểu đối tượng phức tạp
- Tính chất: bao đóng (encapsulation), kế thừa (inheritance), đa hình (polymorphism)

### ● Nhược điểm

- Cấu trúc lưu trữ phức tạp và có thể sử dụng nhiều con trỏ
- Khả năng tối ưu hoá các xử lý bị hạn chế trong nhiều trường hợp

### ● So sánh và đánh giá

● **Nhắc lại: Mô hình dữ liệu hướng đối tượng của một CSDL**

	Mô hình mạng	Mô hình phân cấp	Mô hình quan hệ	Mô hình TT-LK	Mô hình HDT
<b>biểu diễn ngữ nghĩa DL</b>	hạn chế	hạn chế	<i>tương đối đa dạng</i>	đa dạng	đa dạng
<b>lưu trữ DL</b>	s/d nhiều con trỏ	dữ liệu lặp lại	đơn giản và hiệu quả	khó lưu trữ	<i>cấu trúc phức tạp</i>
<b>khả năng truy vấn</b>	đơn giản	đơn giản	đa dạng		đa dạng
<b>hiệu quả của truy vấn</b>	ít khả năng tối ưu	ít khả năng tối ưu	tối ưu hoá tốt	không được xem xét (không hiệu quả)	<i>không h/q khi s/d nhiều con trỏ</i>

1 trúc của