



# Database & SQL

28 Tech



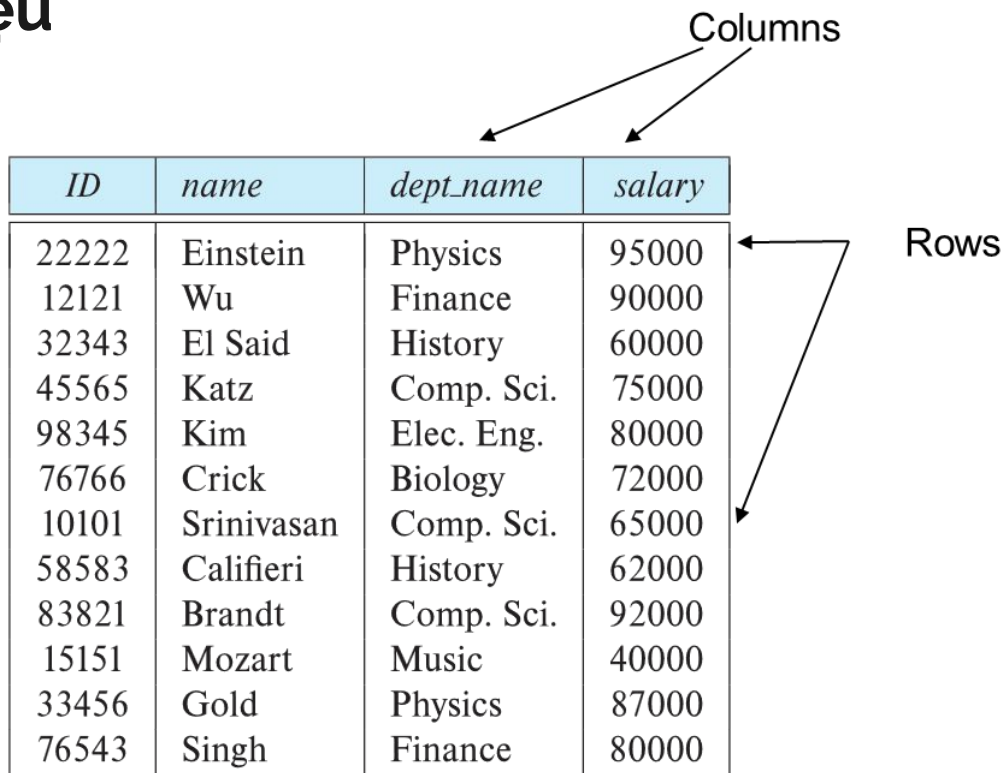
# Cơ sở dữ liệu và SQL 28Tech



1. Lý thuyết về Cơ sở dữ liệu
2. Bài tập về SQL từ cơ bản đến nâng cao
3. Cách chuẩn hóa cơ sở dữ liệu
4. Thao tác phức tạp với CSDL
5. Các project web kết nối tới Cơ sở dữ liệu

# CƠ SỞ DỮ LIỆU

# Cơ sở dữ liệu



The diagram shows a table with four columns and ten rows. Two arrows labeled 'Columns' point to the column headers 'dept\_name' and 'salary'. Two arrows labeled 'Rows' point to the first and last data rows of the table.

| <i>ID</i> | <i>name</i> | <i>dept_name</i> | <i>salary</i> |
|-----------|-------------|------------------|---------------|
| 22222     | Einstein    | Physics          | 95000         |
| 12121     | Wu          | Finance          | 90000         |
| 32343     | El Said     | History          | 60000         |
| 45565     | Katz        | Comp. Sci.       | 75000         |
| 98345     | Kim         | Elec. Eng.       | 80000         |
| 76766     | Crick       | Biology          | 72000         |
| 10101     | Srinivasan  | Comp. Sci.       | 65000         |
| 58583     | Califieri   | History          | 62000         |
| 83821     | Brandt      | Comp. Sci.       | 92000         |
| 15151     | Mozart      | Music            | 40000         |
| 33456     | Gold        | Physics          | 87000         |
| 76543     | Singh       | Finance          | 80000         |

(a) The *instructor* table

# Cơ sở dữ liệu



```
SELECT department, COUNT(*) AS total_employees
FROM employees
WHERE salary > 3000
GROUP BY department
HAVING total_employees > 5
ORDER BY total_employees DESC
LIMIT 3;
```

# Cơ sở dữ liệu



1. **Cơ sở dữ liệu** là một bộ lưu trữ dữ liệu có cấu trúc được lưu trữ trong một thiết bị điện tử. Dữ liệu có thể là văn bản, video, hình ảnh hoặc bất kỳ định dạng nào khác.
2. CSDL thể hiện các khía cạnh khác nhau của thế giới thực.
3. Một cơ sở dữ liệu quan hệ lưu trữ dữ liệu dưới dạng bảng và cơ sở dữ liệu
4. SQL (Ngôn ngữ truy vấn có cấu trúc) là ngôn ngữ tiêu chuẩn để truy cập và thao tác dữ liệu trong cơ sở dữ liệu quan hệ.

# Hệ Quản trị CSDL

Hệ Quản trị CSDL là một phần mềm cho phép tạo, cập nhật và truy xuất dữ liệu theo cách có tổ chức. Nó cũng cung cấp bảo mật cho cơ sở dữ liệu.

Cụ thể hơn, gồm một số đặc điểm chính:

1. Cho phép người dùng tạo mới CSDL
2. Cho phép người dùng truy vấn cơ sở dữ liệu
3. Hỗ trợ lưu trữ số lượng lớn dữ liệu
4. Kiểm soát truy nhập dữ liệu

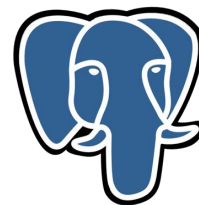


Ví dụ về DBMS quan hệ là MySQL, Oracle, Microsoft SQL Server, Postgresql. Ví dụ về DBMS NoSQL là MongoDB, Cassandra, DynamoDB và Redis.

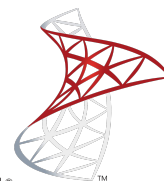
# Hệ Quản trị CSDL

## Các chức năng chính của Hệ quản trị CSDL:

1. Mô hình hóa được dữ liệu
2. Lưu trữ và truy vấn dữ liệu
3. Hỗ trợ nhiều người dùng đồng thời
4. Bảo mật cho dữ liệu
5. Backup và Recovery



PostgreSQL



Microsoft®  
SQL Server®

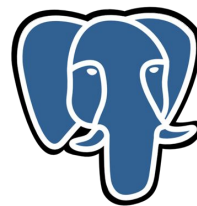




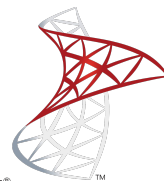
# Hệ Quản trị CSDL

**Nhược điểm** của Hệ quản trị CSDL:

1. Phức tạp
2. Kích thước lớn
3. Chi phí mua và bảo trì
4. Ảnh hưởng lớn khi có lỗi
5. Thêm giá thành cho phần cứng hỗ trợ



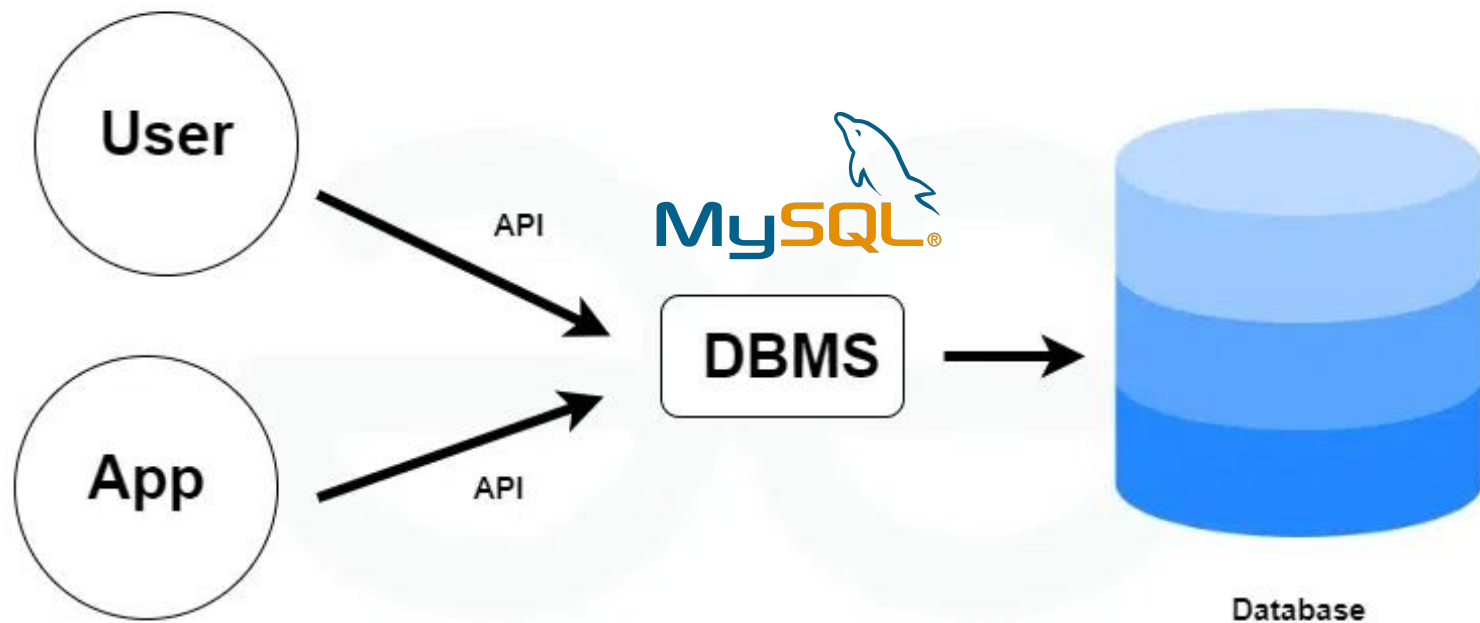
PostgreSQL



Microsoft®  
SQL Server®



# Hệ Quản trị CSDL



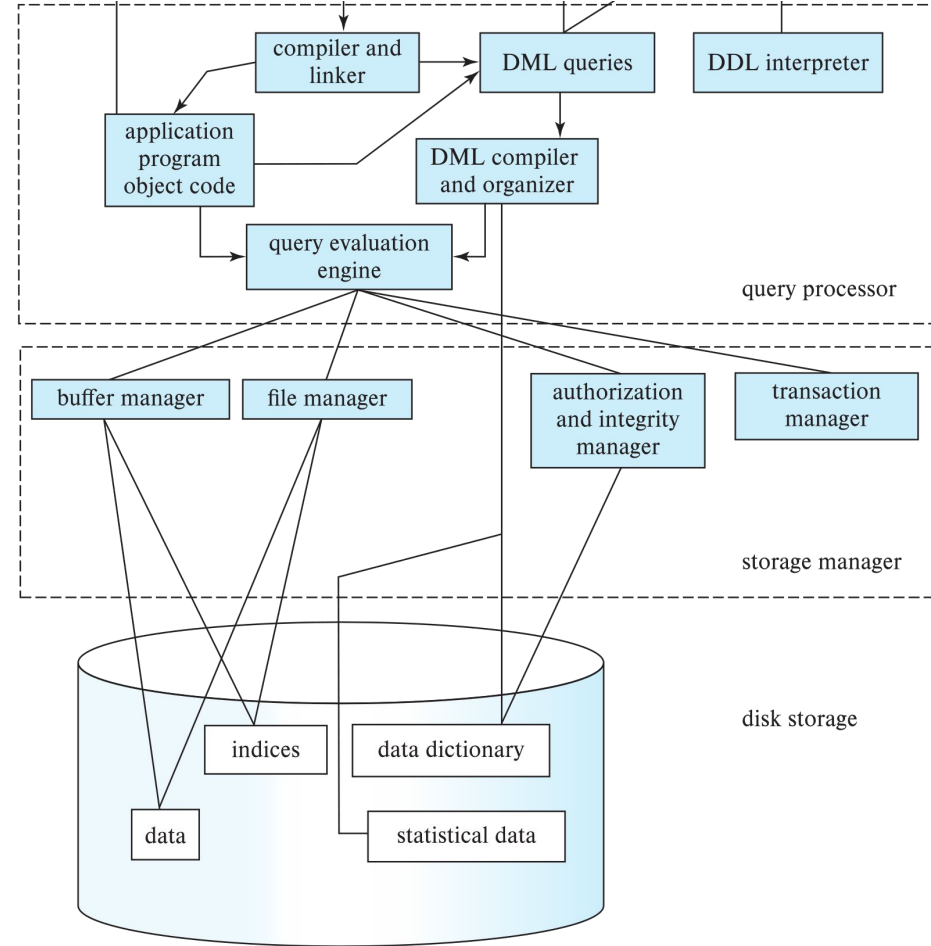
# CSDL với Hệ thống file

| Tính năng                           | Hệ thống file | Hệ quản trị CSDL |
|-------------------------------------|---------------|------------------|
| Dư thừa dữ liệu                     | Cao           | Thấp             |
| Bảo mật dữ liệu                     | Thấp          | Cao              |
| Hỗ trợ mối quan hệ giữa các dữ liệu | Không         | Đầy đủ           |
| Truy cập nhiều người dùng           | Giới hạn      | Hỗ trợ đầy đủ    |

# Các thành phần của DBMS

Thành phần chính:

1. Bộ xử lý truy vấn (Query Processor)
2. Bộ quản lý lưu trữ (Store Manager)
3. Lưu trữ đĩa



# Dữ liệu dẫn xuất và dữ liệu vật lý



## Dữ liệu vật lý

Dữ liệu có thực, được nhập vào trong CSDL

## Dữ liệu dẫn xuất

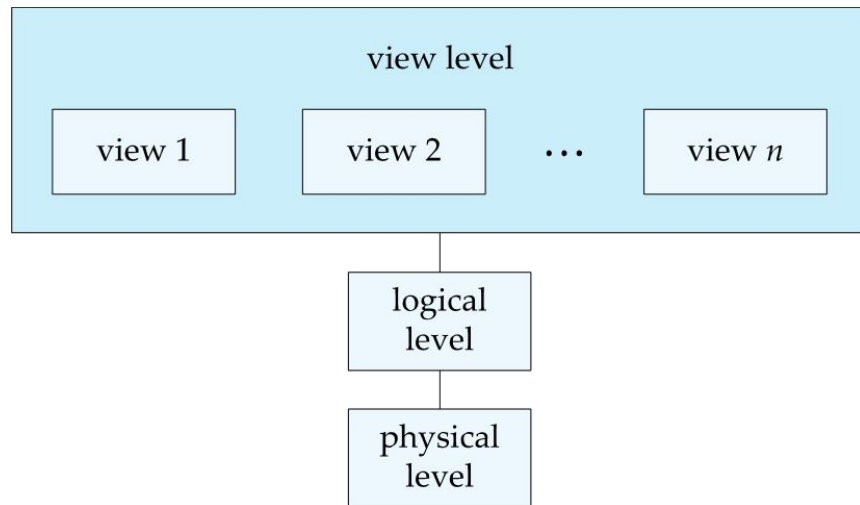
Dữ liệu được **tính toán** từ những dữ liệu nằm trong CSDL.

VD: CSDL lưu thông tin về Sinh viên, trong đó lưu điểm thi của SV.

Giá trị điểm thi trung bình của SV có cần lưu trong CSDL?

# Mô hình trừu tượng 3 lớp

Mục đích của việc trừu tượng hóa dữ liệu là **đạt được tính độc lập của dữ liệu** để tiết kiệm thời gian và chi phí cần thiết khi cơ sở dữ liệu bị sửa đổi hoặc thay đổi.



## Mô hình trừu tượng 3 lớp: Mức ngoài



1. **Mức ngoài** là khung nhìn của người sử dụng CSDL, mô tả phần CSDL tương ứng với người dùng đó.
2. Mỗi người dùng có thể chỉ thấy một phần dữ liệu mà họ được phép truy cập.
3. Có thể tồn tại **nhiều** chế độ xem của cùng 1 CSDL, các chi tiết lưu trữ và triển khai được ẩn khỏi giao diện

VD1: Giáo viên chỉ thấy danh sách sinh viên, điểm số. Sinh viên chỉ thấy điểm số của mình, không thấy dữ liệu của sinh viên khác. Admin xem tất cả thông tin

VD2: Khung nhìn sẽ không nhìn thấy Mật khẩu hoặc Lương của nhân viên, để đảm bảo an toàn bảo mật

# Mô hình trừu tượng 3 lớp: Mức logic - khái niệm



1. Là bản thiết kế logic của toàn bộ cơ sở dữ liệu.
2. Xác định cấu trúc bảng, quan hệ giữa các bảng, các ràng buộc (constraints), quy tắc nghiệp vụ
3. Ấn chi tiết vật lý của cách dữ liệu được lưu trữ

```
type instructor = record  
    ID : string;  
    name : string;  
    dept_name : string;  
    salary : integer;  
end;
```



# Mô hình trừu tượng 3 lớp: Mức trong - vật lý



1. Là mức thấp nhất, mô tả cách dữ liệu thực sự được lưu trữ trên đĩa hoặc bộ nhớ
2. Xác định cách dữ liệu được tổ chức dưới dạng file, index, cấu trúc lưu trữ

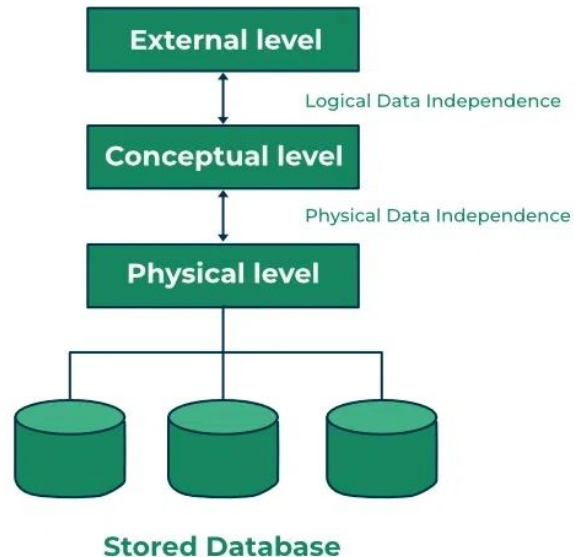
VD: Dữ liệu được lưu trong PostgreSQL/MySQL. Các bảng có thể được phân vùng để tối ưu hóa hiệu suất...

# Độc lập dữ liệu

Cụ thể, có hai cấp độ độc lập dữ liệu phát sinh từ các cấp độ trừu tượng này:

1. Độc lập dữ liệu cấp độ vật lý
2. Độc lập dữ liệu cấp độ logic

## Data Independence in DBMS



# Độc lập dữ liệu

---

## Độc lập dữ liệu cấp độ **vật lý**

- ❖ Có thể sửa đổi lược đồ vật lý mà không cần thay đổi / không ảnh hưởng tới lược đồ khái niệm.
- ❖ VD: Thay đổi kích thước lưu trữ của máy chủ hệ thống CSDL hoặc sửa cấu trúc dữ liệu dùng để lưu trữ, ... đều không ảnh hưởng tới lược đồ khái niệm

## Độc lập dữ liệu cấp độ **logic**:

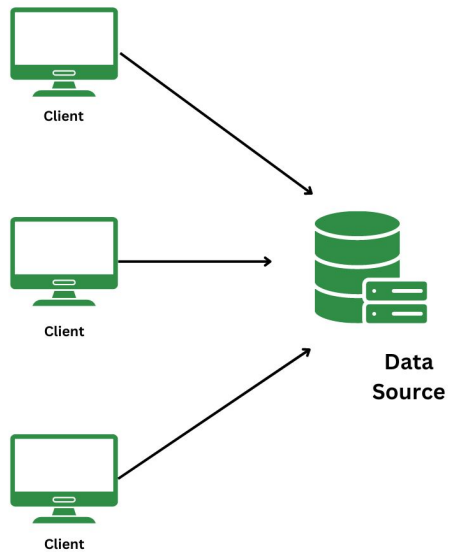
- ❖ Có thể sửa đổi lược đồ logic mà không ảnh hưởng đến lược đồ mức ngoài hoặc chương trình ứng dụng.
- ❖ VD: Chèn, thêm thuộc tính, thay đổi cấu trúc bảng, thay đổi mối quan hệ giữa các bảng,... đều không ảnh hưởng tới lược đồ mức ngoài

# MÔ HÌNH TRIỂN KHAI

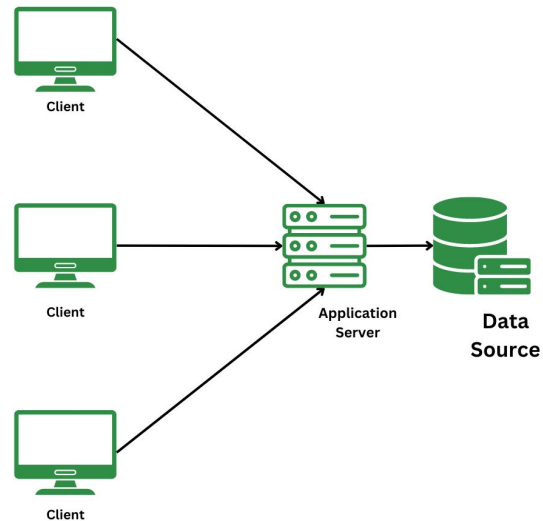


# Độc lập dữ liệu

## Two Tier Architecture



## Three Tier Architecture



## Mô hình 2 lớp (2 tier)

---

- ✓ Đơn giản, dễ triển khai và quản lý.
- ✓ Hiệu suất cao nếu số lượng người dùng nhỏ.
- ✗ Ứng dụng client cần biết trực tiếp về database (cấu trúc bảng, truy vấn SQL).
- ✗ Khó mở rộng khi số lượng client tăng cao vì tất cả client đều kết nối trực tiếp đến DB.
- ✗ Bảo mật kém hơn vì client có quyền truy cập trực tiếp vào database.

⇒ Các ứng dụng desktop kết nối trực tiếp với cơ sở dữ liệu như Microsoft Access, ứng dụng cục bộ với SQLite.

## Mô hình 3 lớp (3 tier)

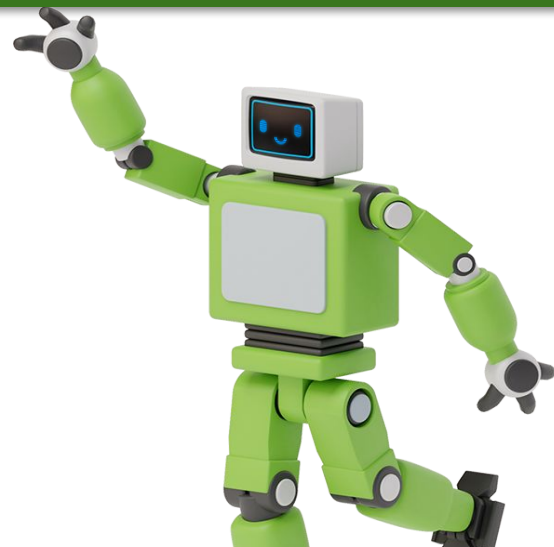


- ✓ Bảo mật cao vì client không truy cập trực tiếp database, mọi thao tác đều qua Application Server.
- ✓ Dễ mở rộng do có thể mở rộng từng tầng độc lập.
- ✓ Tái sử dụng code vì API có thể dùng chung cho nhiều client khác nhau (web, mobile).

- ✗ Phức tạp hơn so với mô hình 2 lớp.
- ✗ Hiệu suất có thể bị ảnh hưởng do phải qua một tầng trung gian.

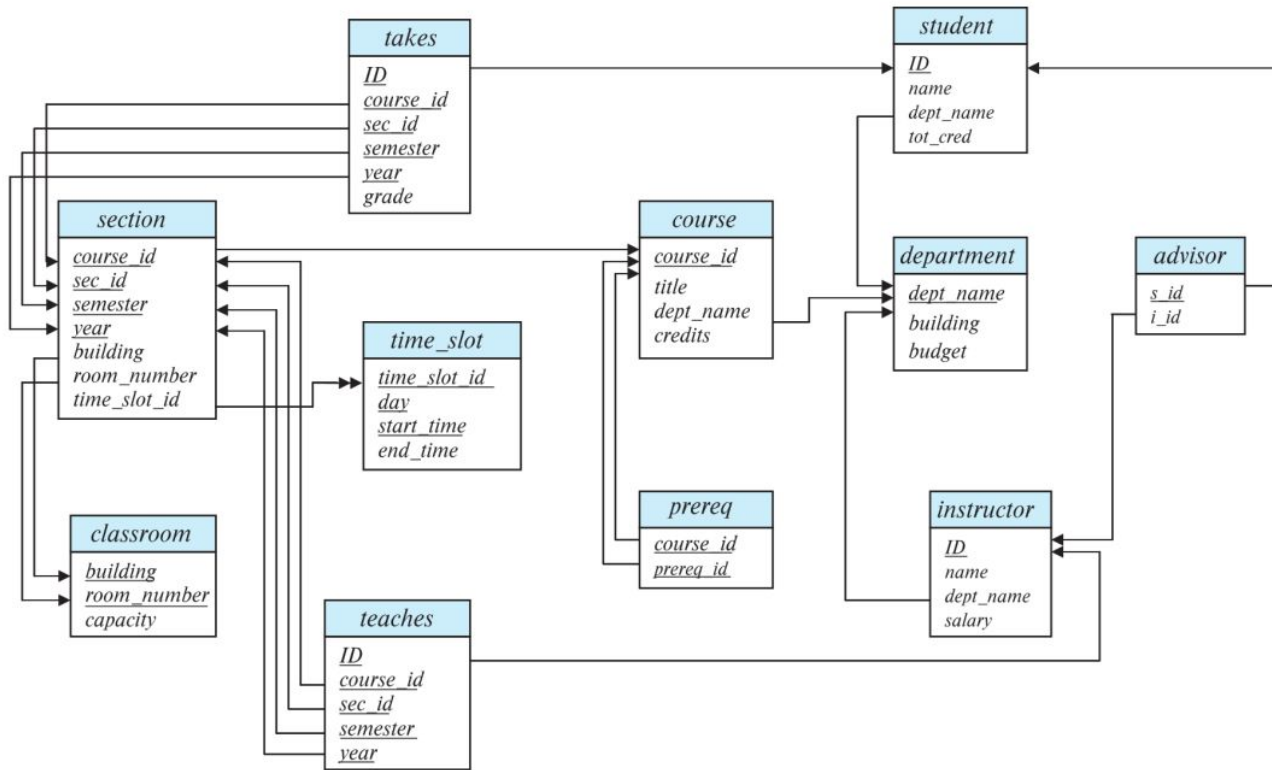
⇒ Hệ thống website, thương mại điện tử, ...

# CÁC MÔ HÌNH DỮ LIỆU





# Mô hình hóa dữ liệu



# Mô hình hóa dữ liệu



**Mô hình hóa dữ liệu** là một kỹ thuật mô hình trực quan được sử dụng để có được cái nhìn tổng quan cấp cao về cơ sở dữ liệu.

Các mô hình dữ liệu giúp hiểu các nhu cầu và yêu cầu của cơ sở dữ liệu bằng cách xác định thiết kế cơ sở dữ liệu thông qua biểu diễn **sơ đồ**.

# Quá trình thiết kế một CSDL

---

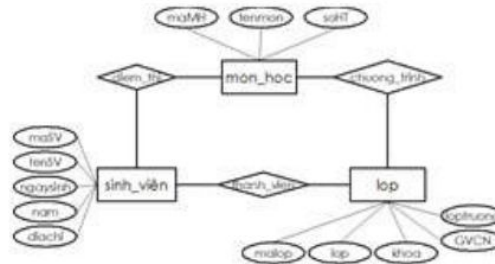
- ❖ Thiết kế cơ sở dữ liệu có thể được định nghĩa là một tập hợp các thủ tục đảm bảo hệ thống **lưu trữ dữ liệu hiệu quả, nhất quán và dễ mở rộng**
- ❖ Thiết kế cơ sở dữ liệu tốt giúp có được thông tin đúng khi cần.
- ❖ Một số điểm quan trọng cần lưu ý để đạt được thiết kế cơ sở dữ liệu tốt:
  1. Tính nhất quán và tính toàn vẹn của dữ liệu phải được duy trì.
  2. Không dư thừa
  3. Tìm kiếm nhanh hơn thông qua các chỉ số
  4. Các biện pháp bảo mật nên được thực hiện bằng cách thực thi các ràng buộc toàn vẹn khác nhau
  5. Dữ liệu nên được lưu trữ thông tin ở định dạng nguyên tử nhất có thể.

# Quá trình thiết kế một CSDL



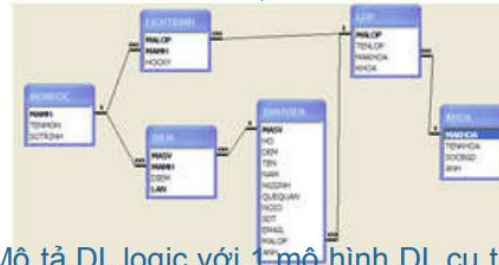
Mô tả ứng dụng

1: PHÂN TÍCH



Mô hình hoá DL (vd: Sơ đồ thực thể-liên kết)

2: THIẾT KẾ



Mô tả DL logic với 1 mô hình DL cụ thể  
(vd: Sơ đồ quan hệ)

3: CÀI ĐẶT



Cài đặt với 1 hệ quản trị CSDL  
(vd: ORACLE)

# Quá trình thiết kế một CSDL



## **Bước 1. Phân tích yêu cầu**

- ❖ Suy luận về các use case sẽ được sử dụng để ngăn chặn các vấn đề phát sinh
- ❖ Các yêu cầu về hiệu năng của hệ thống

## **Bước 2. Thiết kế CSDL mức khái niệm**

- ❖ Mô tả tổng quát về dữ liệu và các ràng buộc cần thiết trên dữ liệu này

## **Bước 3. Thiết kế CSDL mức logic**

- ❖ Thiết kế ở mức trừu tượng cao dựa vào những yêu cầu đã phân tích được, không quan tâm về thiết kế mức vật lý
- ❖ Xác định các loại dữ liệu, kiểu dữ liệu được lưu, và mối quan hệ giữa các thực thể
- ❖ Ở bước này cũng cần cải tiến lược đồ, chuẩn hóa để ngăn chặn dư thừa dữ liệu

## **Bước 4. Thiết kế CSDL mức vật lý**

- ❖ Tất cả các mối quan hệ giữa các ràng buộc dữ liệu và tính toàn vẹn được cài đặt để duy trì tính nhất quán và tạo cơ sở dữ liệu thực tế.

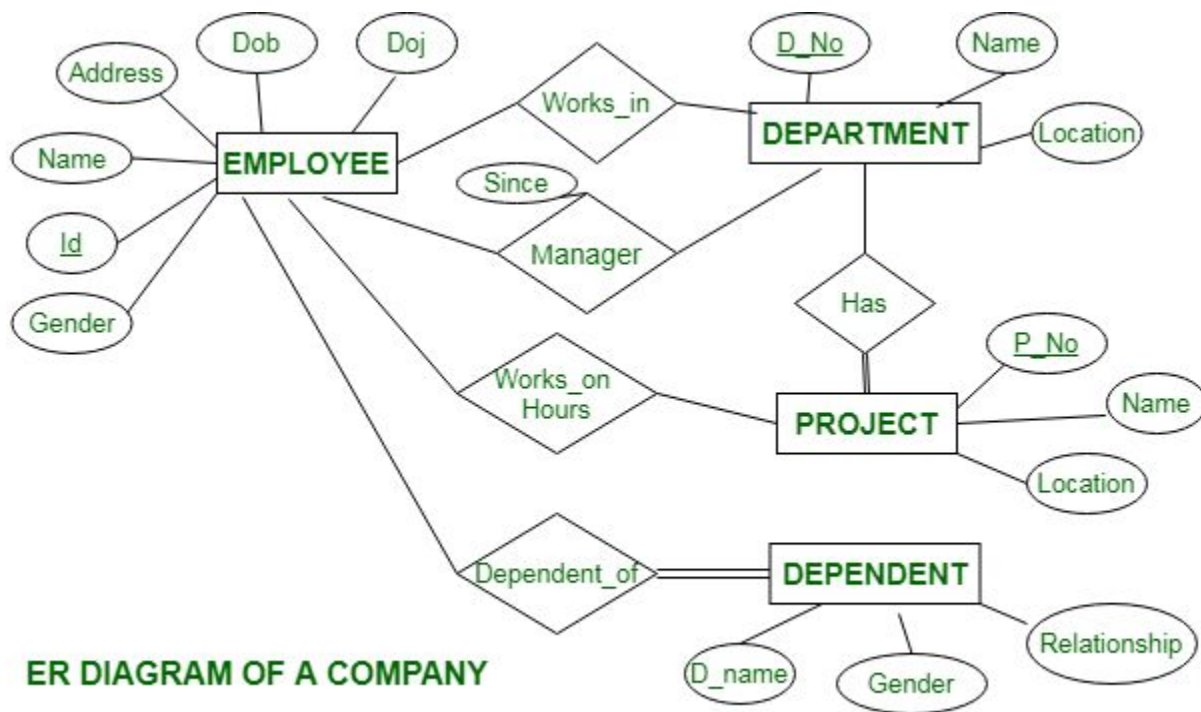
## **Bước 5. Thiết kế an toàn bảo mật**

- ❖ Phân quyền vai trò của nhóm người dùng

# MÔ HÌNH THỰC THỂ LIÊN KẾT E-R



# Lược đồ thực thể liên kết ?



# Tại sao phải sử dụng lược đồ E-R?



1. Các sơ đồ ER đại diện / ánh xạ cho lược đồ E-R trong cơ sở dữ liệu, giúp chúng dễ dàng chuyển đổi thành các **quan hệ** (bảng).
2. Các sơ đồ ER cung cấp mục đích của mô hình trong **thế giới thực** của các đối tượng.
3. Cung cấp một giải pháp **tiêu chuẩn** để trực quan hóa dữ liệu một cách hợp lý.



# Lược đồ thực thể liên kết



**Mô hình thực thể-liên kết** (Entity-Relationship, viết tắt ER) là một mô hình dữ liệu mức quan niệm nhằm mô tả **các đối tượng** trong thế giới thực và **quan hệ giữa chúng**

# Lược đồ thực thể liên kết



Lược đồ thực thể liên kết gồm 3 khái niệm cơ bản: **tập thực thể**, **tập quan hệ** và **thuộc tính**.

**Thực thể**: Là một đối tượng tồn tại trong thế giới thực (1 người, 1 chiếc xe, 1 ngôi nhà) hoặc có thể là cũng có thể là đối tượng trừu tượng (1 công ty, 1 công việc, 1 môn học, 1 khoản vay, ...)

**Tập thực thể**: Một tập hợp tất cả các thực thể được gọi là một tập hợp thực thể.

VD: Bạn SV *Nguyễn Văn Nam* là một thực thể sinh viên thì tập hợp tất cả các sinh viên được gọi là tập thực thể

# Lược đồ thực thể liên kết

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

*instructor*

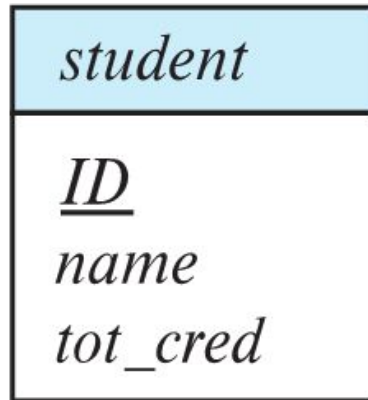
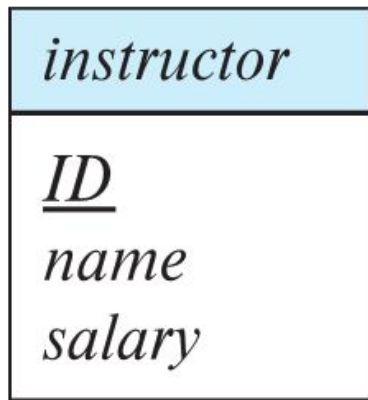
|       |         |
|-------|---------|
| 98988 | Tanaka  |
| 12345 | Shankar |
| 00128 | Zhang   |
| 76543 | Brown   |
| 76653 | Aoi     |
| 23121 | Chavez  |
| 44553 | Peltier |

*student*

Thực thể được biểu diễn bởi một tập các **thuộc tính**

# Lược đồ thực thể liên kết

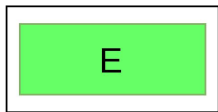
**Thuộc tính:** Là 1 đặc trưng mà trị của nó tham gia vào việc mô tả một thực thể. Mỗi thuộc tính có một tập giá trị cho phép, được gọi là **miền** (hay tập giá trị) của thuộc tính đó.



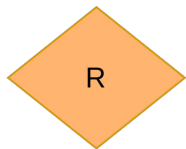
# Ký hiệu trong lược đồ e-r



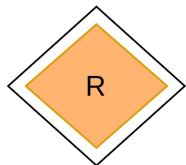
Tập thực thể



Tập thực thể yếu

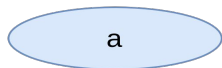


Liên kết

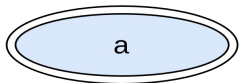


Liên kết xác định cho  
một tập thực thể yếu

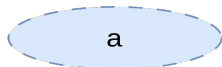
# Ký hiệu trong lược đồ E-R



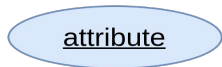
Thuộc tính (VD: first\_name, date\_of\_birth, ...)



Thuộc tính đa trị (VD: phones, hobbies, ...)



Thuộc tính dẫn xuất (VD: age - tuổi tính từ ngày sinh)



Thuộc tính khóa chính



Thuộc tính khóa phân biệt  
của 1 tập thực thể yếu

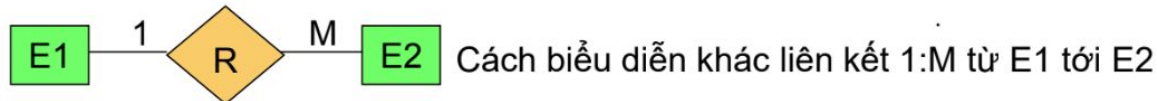
# Ký hiệu trong lược đồ E-R



Liên kết 1:1 từ E1 tới E2



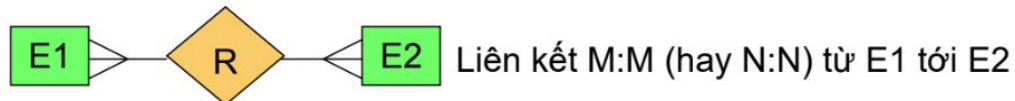
Liên kết 1:M (hay 1:N) từ E1 tới E2



Cách biểu diễn khác liên kết 1:M từ E1 tới E2



Liên kết M:1 (hay N:1) từ E1 tới E2

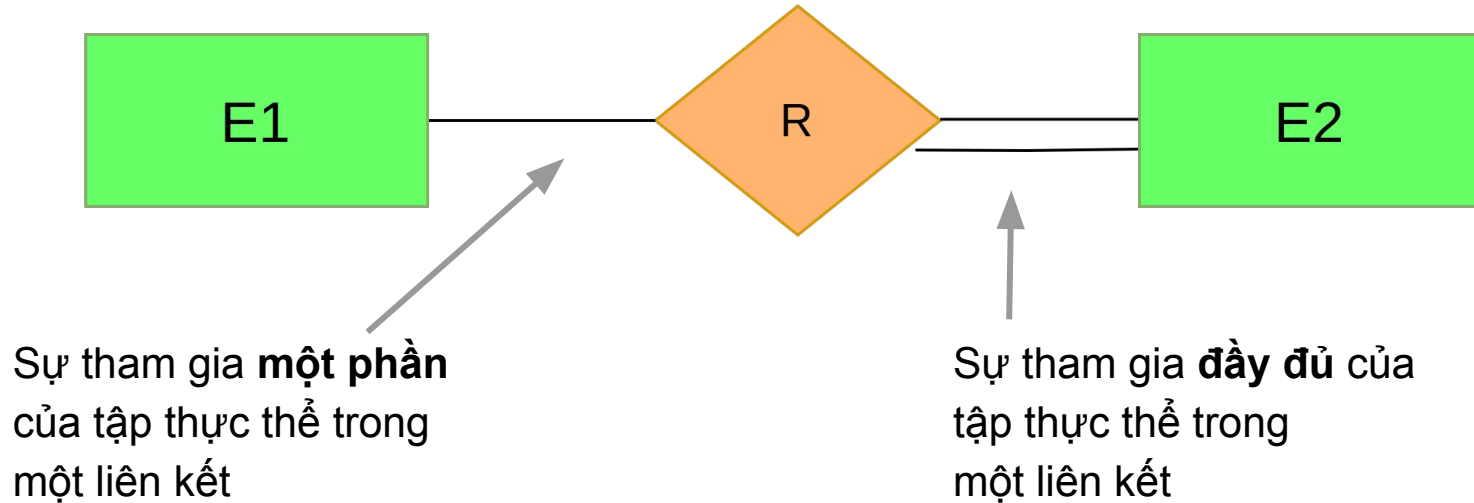


Liên kết M:M (hay N:N) từ E1 tới E2



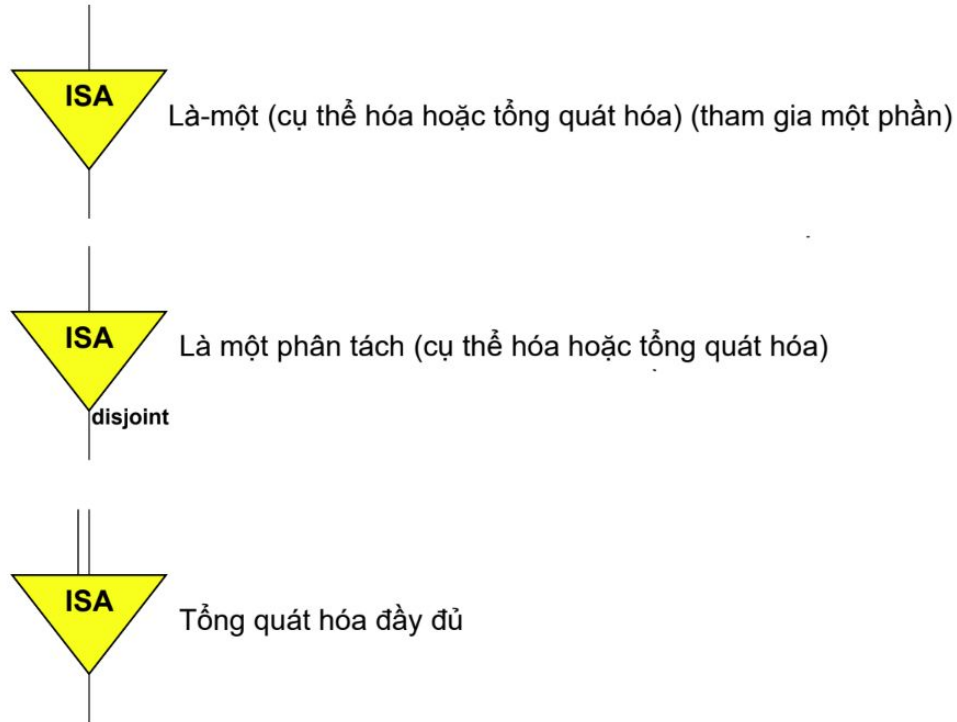
Cách biểu diễn khác liên kết M:M từ E1 tới E2

## Ký hiệu trong lược đồ E-R





# Ký hiệu trong lược đồ E-R



# Các loại Thực thể



**Thực thể mạnh:** Một thực thể mạnh là một loại thực thể có khóa chính. Thực thể mạnh không phụ thuộc vào thực thể khác trong lược đồ. Nó có một khóa chính, giúp xác định nó một cách duy nhất.

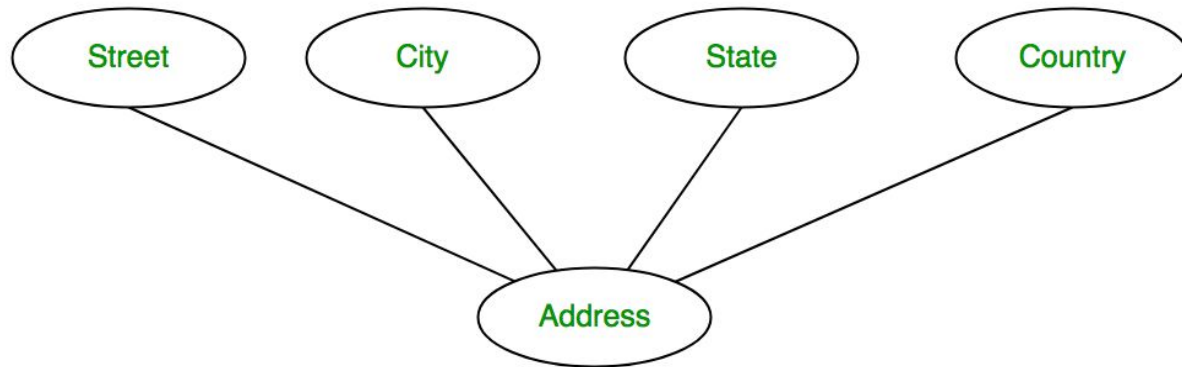
**Thực thể yếu:** Một thực thể yếu không thể xác định được khóa chính, thực thể yếu sẽ phụ thuộc vào 1 thực thể mạnh sở hữu nó.

VD: Một phòng ban có nhiều dự án, mỗi dự án chỉ được đánh số thứ tự (Dự án số 1, số 2, ...). Vậy “*dự án số 1*” **không thể** xác định được chính xác một dự án cụ thể. Nhưng “*dự án số 1*” có thể xác định được nếu nằm trong ngữ cảnh của 1 phòng ban. “phòng ban” là thực thể mạnh

# Các loại Thuộc tính

## Thuộc tính đơn hoặc thuộc tính kép

Thuộc tính đơn không bao gồm các thành phần cấu thành, trong khi thuộc tính kép bao gồm các thành phần con cấu thành.

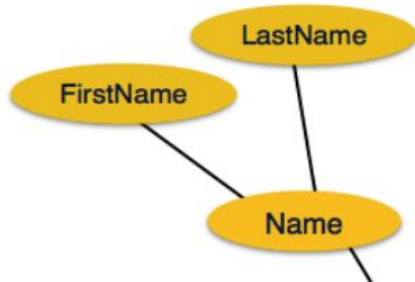


# Các loại Thuộc tính

## Thuộc tính đơn trị hoặc thuộc tính đa trị

Thuộc tính đơn trị có **nhều nhất một** giá trị tại một thời điểm cụ thể. VD: ngày sinh

Thuộc tính đa trị có thể có **nhều giá trị khác nhau** tại một thời điểm. VD: Số ĐT



Name là đơn trị hay đa trị?



Thuộc tính đa trị: Phone\_No

# Các loại Thuộc tính



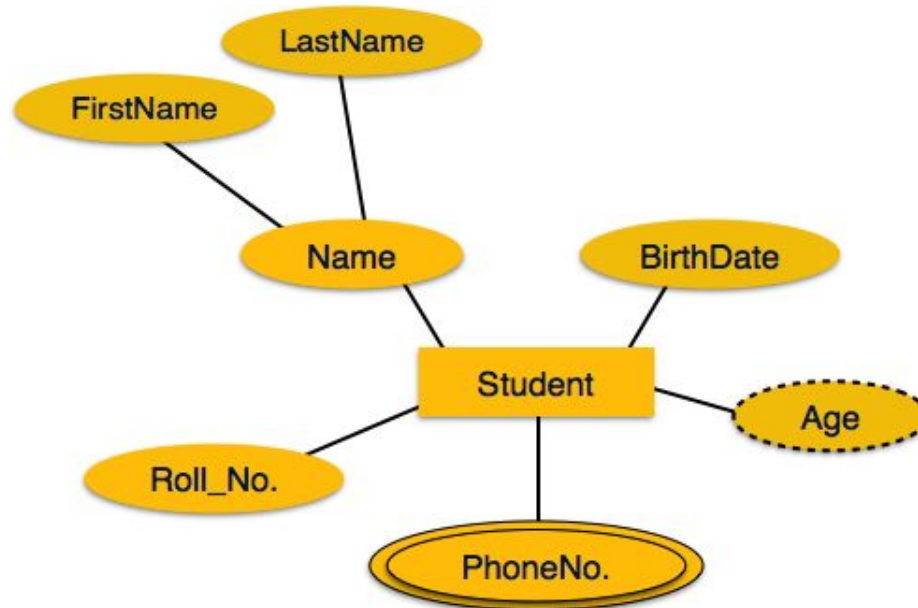
**Thuộc tính dẫn xuất:** là thuộc tính mà giá trị của nó được dẫn xuất (hoặc được tính toán) từ những giá trị của các thuộc tính hoặc các thực thể có liên quan.

VD: Tuổi là thuộc tính dẫn xuất do có thể tính được từ thuộc tính ngày sinh



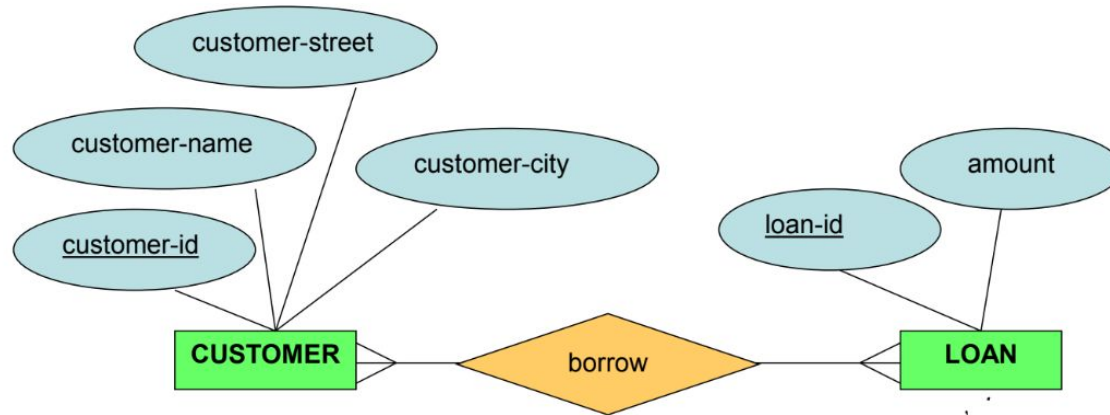
**Thuộc tính rỗng (Null):** thuộc tính nhận giá trị rỗng khi một thực thể không có giá trị cho nó.

# Các loại Thuộc tính



Tập thực thể Sinh Viên

# Ví dụ lược đồ thực thể E-R



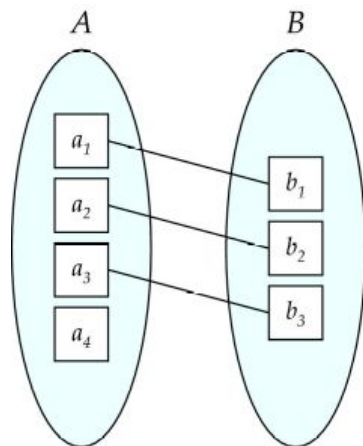
# Ánh xạ lực lượng liên kết

---

- ❖ **Ánh xạ lực lượng liên kết** thể hiện số lượng các thực thể mà một thực thể khác có thể liên hệ thông qua một tập quan hệ.
- ❖ Ràng buộc ánh xạ lực lượng liên kết có ích nhất khi mô tả các quan hệ hai ngôi.
- ❖ Với một tập quan hệ hai ngôi  $R$  giữa tập thực thể  $A$  và  $B$ , ánh xạ lực lượng liên kết gồm các loại sau:
  1. One to one
  2. One to many
  3. Many to one
  4. Many to many

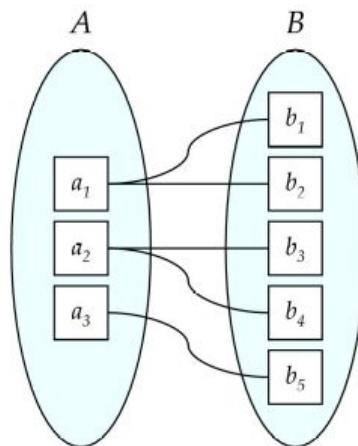


# Ánh xạ lực lượng liên kết



(a)

One to one

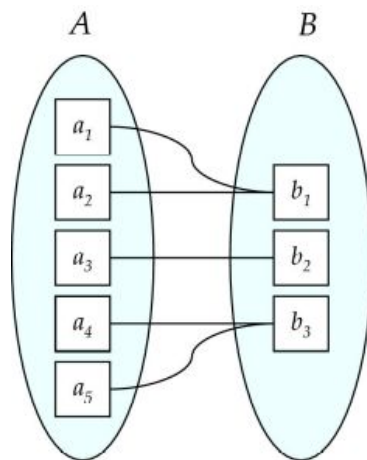


(b)

One to many

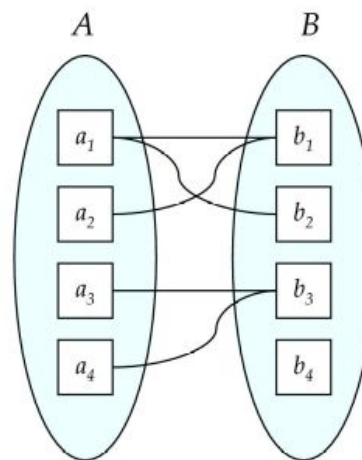
*Lưu ý: Một số phần tử trong A và B có thể không được ánh xạ tới bất kỳ phần tử nào trong tập hợp còn lại (ví dụ  $a_4$ )*

# Ánh xạ lực lượng liên kết



(a)

Many to one



(b)

Many to many

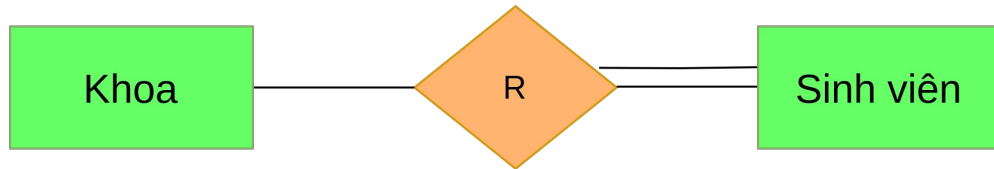
*Lưu ý: Một số phần tử trong  $A$  và  $B$  có thể không được ánh xạ tới bất kỳ phần tử nào trong tập hợp còn lại*

# Ràng buộc tham gia

**Tham gia toàn bộ:** Nếu mỗi thực thể trong tập hợp thực thể phải tham gia vào mỗi quan hệ

**Tham gia một phần:** Nếu thực thể trong tập hợp thực thể có thể hoặc không tham gia vào mỗi quan hệ

VD: Tất cả các **sinh viên** phải trực thuộc một **Khoa** nào đó. Một Khoa được phép không có sinh viên



Khách hàng và Khoản vay ?

# Khóa của một tập thực thể

Cần có một cơ chế để phân biệt các thực thể trong một tập thực thể. Cơ chế này phải thể hiện thông qua các thuộc tính của tập thực thể.

**Khóa (key)** cho phép xác định một **tập** các thuộc tính **đủ để phân biệt** các thực thể với nhau.



# Khóa của một tập thực thể

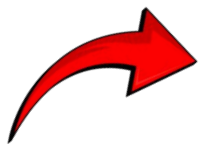
---

**Siêu khóa (super key)** là một tập gồm một hoặc nhiều thuộc tính được lựa chọn cho phép xác định duy nhất một thực thể trong tập thực thể.

STUDENTS (student\_id, name, address, age, major, minor, gpa, spring-sch)

Siêu khóa cho tập thực thể STUDENTS có thể là:

(student\_id, name, major, minor) hoặc (student\_id, name)



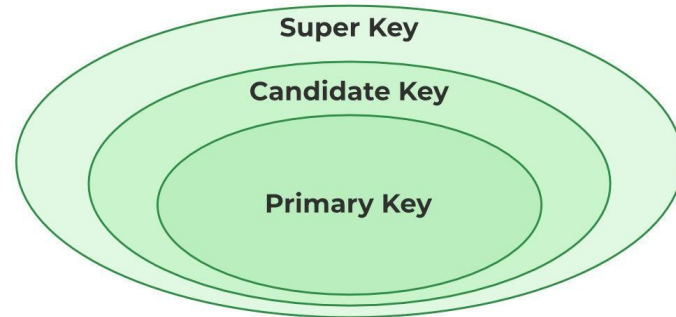
*Khái niệm siêu khóa là một định nghĩa không đầy đủ của khóa vì siêu khóa còn chứa nhiều **thuộc tính dư thừa**.*

# Khóa của một tập thực thể

**Khóa dự bị (candidate key)** là những siêu khóa mà không có tập con nào của nó là siêu khóa.

Người thiết kế CSDL chỉ chọn **một khóa dự bị** làm **khóa chính** của tập thực thể.

VD: Mã sinh viên là khóa chính của Sinh Viên. Email hoặc CCCD là khóa dự bị của Sinh viên



# Khóa của một tập thực thể

## Khóa chính:

1. Xác định duy nhất 1 hàng và không dư thừa thuộc tính.
2. Thuộc tính không được rỗng hoặc NULL.
3. Thuộc tính rất hiếm khi / gần như không thay đổi.
4. Nên ở dạng numeric

