



HỌC VIỆN NÔNG NGHIỆP VIỆT NAM
KHOA CÔNG NGHỆ THÔNG TIN

BÀI GIẢNG TIN HỌC ĐẠI CƯƠNG

Chương 5 – CƠ SỞ DỮ LIỆU

NỘI DUNG

- Cơ sở dữ liệu
- Hệ quản trị cơ sở dữ liệu
- Ngôn ngữ truy vấn SQL

NỘI DUNG

- Cơ sở dữ liệu
 - Các khái niệm cơ bản
 - Lợi ích của việc sử dụng cơ sở dữ liệu
 - Các đối tượng sử dụng cơ sở dữ liệu
 - Các mức biểu diễn của cơ sở dữ liệu
 - Mô hình dữ liệu
 - Hệ cơ sở dữ liệu
- Hệ quản trị cơ sở dữ liệu
- Ngôn ngữ truy vấn SQL

Các khái niệm cơ bản

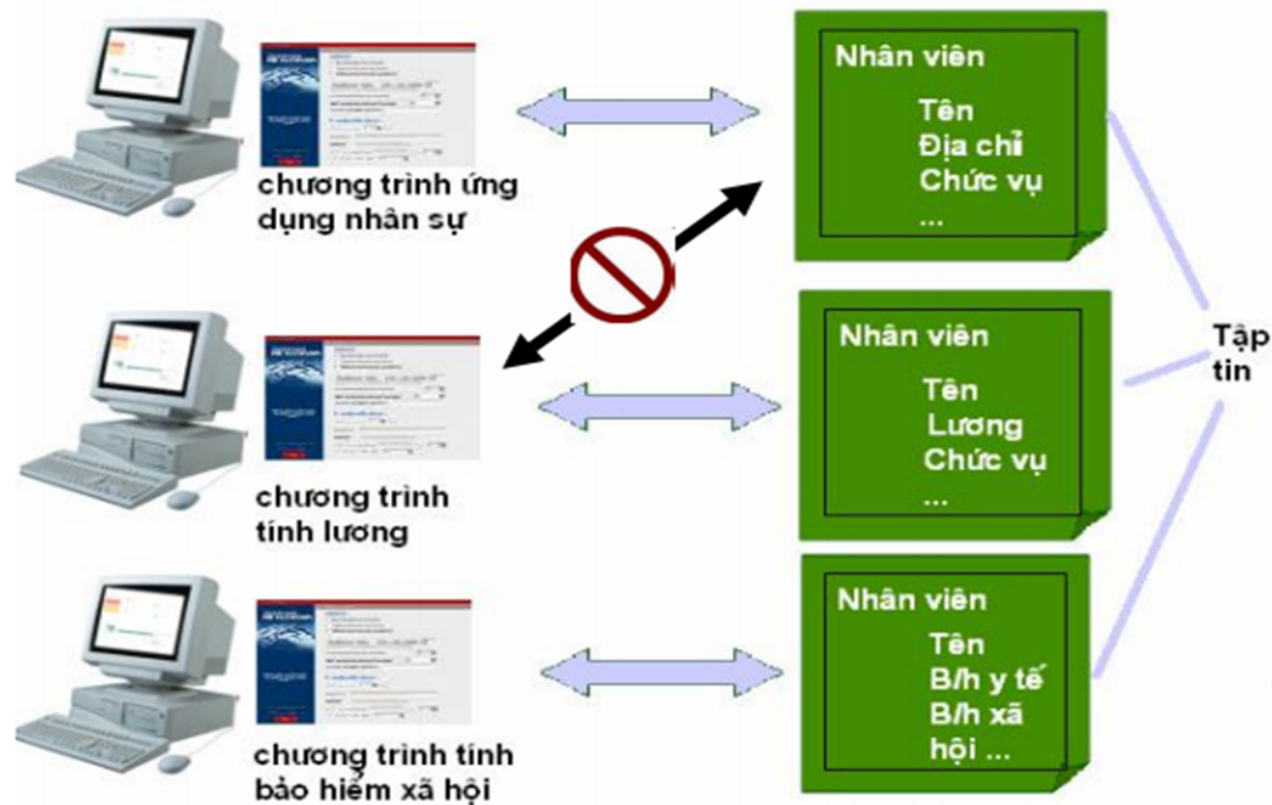
- Thông tin:
 - Là khái niệm trừu tượng giúp ta hiểu và nhận thức về đối tượng.
- Dữ liệu:
 - Là cái mang thông tin: chữ viết, hình ảnh, giọng nói...
- Cơ sở dữ liệu (database):
 - Là một hệ thống thông tin có cấu trúc được lưu trữ trên các thiết bị nhớ thứ cấp (băng từ, đĩa từ,...) để đáp ứng nhu cầu khai thác thông tin của nhiều người sử dụng hoặc nhiều chương trình ứng dụng với nhiều mục đích khác nhau.

Các khái niệm cơ bản

- Hệ cơ sở dữ liệu: Là một hệ thống gồm 4 thành phần:
 - Cơ sở dữ liệu
 - Những người sử dụng cơ sở dữ liệu
 - Hệ quản trị cơ sở dữ liệu
 - Phần cứng

Tại sao cần có CSDL?

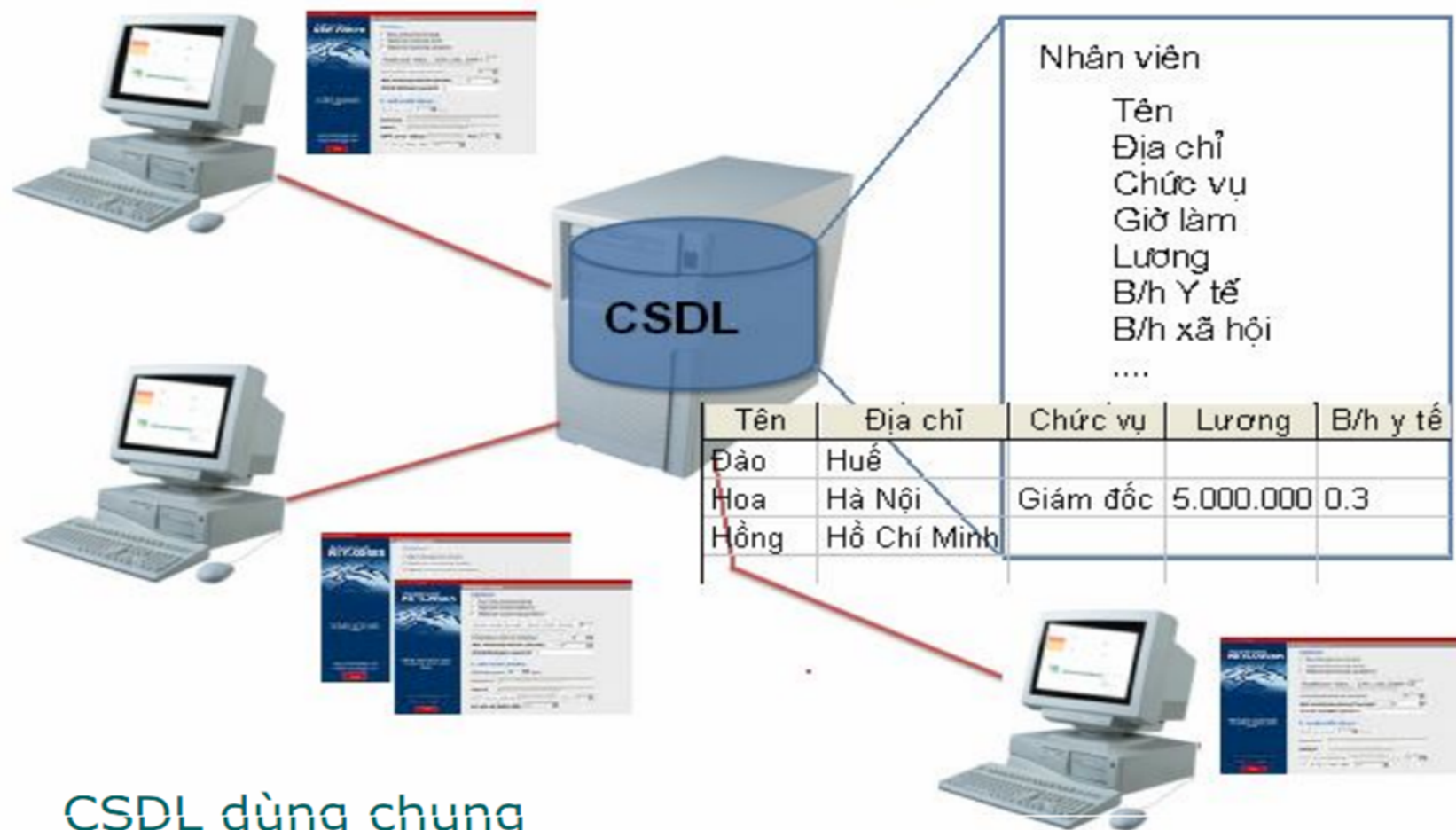
- Hướng tiếp cận hệ tập tin



Hạn chế của hướng tiếp cận hệ tập tin

- Dư thừa và không nhất quán:
 - Thông tin giống nhau có thể bị trùng lặp ở một số nơi.
 - Dữ liệu có thể không được cập nhật đúng.
- Khó khăn khi truy xuất dữ liệu
 - Nhập dữ liệu mất nhiều thời gian.
 - Dữ liệu ở những hệ thống tập tin khác nhau
- ...

CSDL dùng chung



CSDL dùng chung

Lợi ích của việc sử dụng CSDL

- Giảm bớt dư thừa dữ liệu trong lưu trữ
- Tránh được sự không nhất quán trong lưu trữ dữ liệu và bảo đảm được tính toàn vẹn của dữ liệu.
- Có thể triển khai đồng thời nhiều ứng dụng trên cùng một CSDL.
- Thống nhất các tiêu chuẩn, thủ tục và các biện pháp bảo vệ, an toàn dữ liệu.

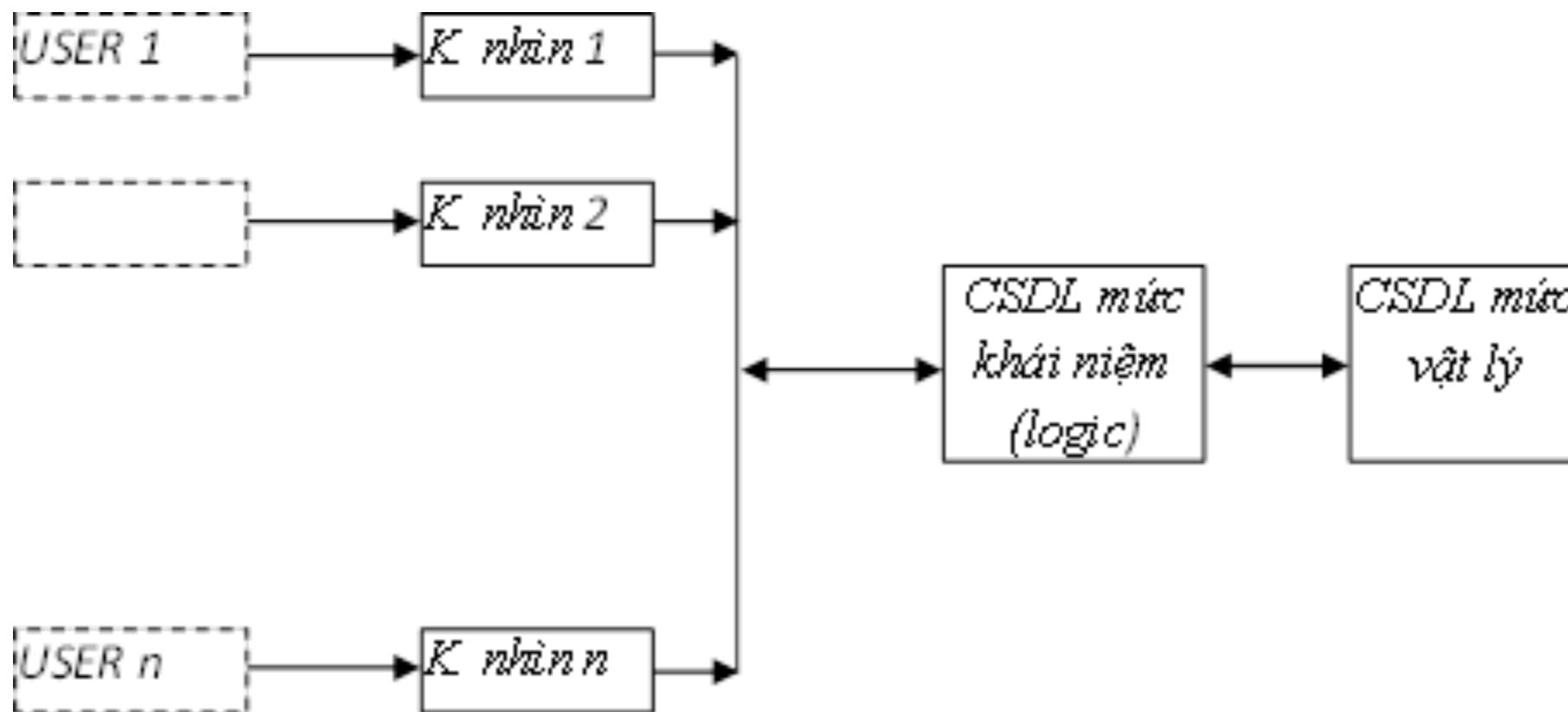
Các đối tượng sử dụng CSDL

- Người dùng cuối: Khai thác CSDL thông qua các ứng dụng hoặc dựa trên phần mềm quản trị CSDL.
- Người lập trình ứng dụng: Là người viết các chương trình ứng dụng cho phép người sử dụng cuối sử dụng CSDL.
- Người quản trị CSDL (Database Administrator): Là người thu thập dữ liệu, thiết kế và bảo trì CSDL, thiết lập các cơ chế đảm bảo an toàn cho CSDL (sao lưu, phục hồi dữ liệu).

Các mức biểu diễn một CSDL

- Mức vật lý (mức trong):
 - Nói đến cách thức lưu trữ dữ liệu như thế nào? Ở đâu? Cần các chỉ mục gì? Việc truy xuất như thế nào?
- Mức logic (mức khái niệm):
 - Trả lời câu hỏi cần phải lưu trữ những loại dữ liệu gì?
 - Mối quan hệ giữa chúng như thế nào?
- Mức khung nhìn (mức ngoài):
 - Là mức của người dùng cuối và các chương trình ứng dụng.
 - Mỗi NDC hay chương trình ứng dụng có thể được nhìn CSDL theo một góc độ (khung nhìn) khác nhau.

Các mức biểu diễn một CSDL



Mô hình dữ liệu

- *Mô hình dữ liệu* là một tập các khái niệm dùng để mô tả dữ liệu, các mối quan hệ của dữ liệu, các ràng buộc trên dữ liệu của một CSDL.
- Các mô hình dữ liệu:
 - Mô hình dữ liệu quan hệ (Relational Data Model)
 - Mô hình dữ liệu mạng (Network Data Model)
 - Mô hình dữ liệu phân cấp (Hierarchical Data Model)
 - Mô hình dữ liệu hướng đối tượng (Object Oriented Data Model)
 - ...

Mô hình dữ liệu quan hệ

- Mô hình dữ liệu quan hệ được đề xuất bởi E.F.Codd vào những năm 1970 - 1972.
- Biểu diễn mọi dữ liệu dưới dạng các bảng, bảng được định dạng gồm các hàng và cột.
- Cơ sở dữ liệu được xây dựng trên mô hình dữ liệu quan hệ được gọi là CSDL quan hệ.

Các khái niệm trong mô hình dữ liệu quan hệ

- Quan hệ
- Thuộc tính
- Bộ giá trị
- Lược đồ quan hệ
- Khóa

Quan hệ

- Một quan hệ là một bảng trong đó:
 - Mỗi hàng (bản ghi) biểu diễn một bộ giá trị của quan hệ. Số các bộ được gọi là lực lượng của quan hệ.
 - Mỗi cột (trường) biểu diễn một thuộc tính/ thành phần của các bộ. Số các thành phần được gọi là bậc của quan hệ.

Các trường ~ thuộc tính

MONHOC	MaMH	TenMH	SoTC	Hocky
	CSDL1	Cơ sở dữ liệu 1	3	4
	CSDL2	Cơ sở dữ liệu 2	3	8
	XLTT	Xử lý thông tin	3	6
	VXL	Vi xử lý	4	7
	DHMT	Đồ họa máy tính	3	7
	TKW	Thiết kế web	3	9
	TCC	Toán cao cấp	4	1

Tên bảng ~ quan hệ

Một bộ ~ bản ghi

Thuộc tính

- Thuộc tính là một tính chất riêng biệt của một đối tượng cần được lưu trữ trong CSDL để phục vụ cho việc khai thác dữ liệu về đối tượng.
- Thuộc tính được đặc trưng bởi:
 - **Tên gọi:** Thuộc tính được đặt tên một cách gợi nhớ và theo quy định.
 - **Kiểu:** Mỗi thuộc tính đều phải thuộc một kiểu dữ liệu nhất định.
 - **Miền giá trị:** Là tập tất cả các giá trị mà thuộc tính có thể có.

Bộ giá trị

- Mỗi dòng của một quan hệ, trừ dòng tiêu đề - tên của các thuộc tính, được gọi là một bộ giá trị (bản ghi)
- Đặc điểm
 - Hai bộ bất kỳ trong quan hệ là khác nhau ở ít nhất giá trị của một thuộc tính.
 - Một bộ không thể được xác định nhờ vị trí của nó trong quan hệ, mà được nhận diện nhờ khóa

Lược đồ quan hệ

- Tên của một quan hệ và tập các thuộc tính của nó được gọi là một lược đồ đối với quan hệ đó.
- Biểu diễn lược đồ cho một quan hệ bởi **tên** của quan hệ và theo sau là danh sách các thuộc tính của nó.

- Ví dụ

MONHOC	MaMH	TenMH	SoTC	Hocky
	CSDL1	Cơ sở dữ liệu 1	3	4
	CSDL2	Cơ sở dữ liệu 2	3	8
	XLTT	Xử lý thông tin	3	6
	VXL	Vi xử lý	4	7
	DHMT	Đồ họa máy tính	3	7
	TKW	Thiết kế web	3	9
	TCC	Toán cao cấp	4	1

→ lược đồ quan hệ: MONHOC(MaMH, TenMH, SoTC, Hocky)

Khóa

- Khóa (đề nghị) của một quan hệ là tập nhỏ nhất các thuộc tính mà giá trị của nó có thể xác định được duy nhất một bộ giá trị của quan hệ.
- Một quan hệ có thể có nhiều khóa đề nghị, khi đó sẽ chọn một khóa làm khóa chính
- Ví dụ: quan hệ **MONHOC(MaMH, TenMH, SoTC, Hocky)** có hai khóa đề nghị: **MaMH, TenMH**
→ chọn **MaMH** làm khóa chính

Siêu khóa

- K được gọi là siêu khóa của quan hệ R nếu $K' \subseteq K$ là một khóa của quan hệ R.
- Ví dụ : quan hệ **MONHOC(MaMH, TenMH, SoTC, Hocky)** có các siêu khóa:

$$K1 = \{MaMH, TenMH\};$$

$$K2 = \{MaMH, SoTC\}$$

$$K3 = \{MaMH, TenMH, SoTC\}$$

....

Khóa ngoại

- Một/ một tập thuộc tính của quan hệ mà giá trị của nó khớp với khóa chính của một quan hệ khác thì nó được gọi là khóa ngoại của quan hệ đó.
- Khoá ngoại dùng để biểu thị liên kết giữa quan hệ này và quan hệ khác trong mô hình quan hệ.
- Ví dụ:

KHOA(Makhoa, Tenkhoa, Diadiem, SDT)

LOP(MaLop, TenL, Siso, Makhoa)

→ Trong quan hệ **LOP**, **Makhoa** là khóa ngoại vì **Makhoa** là khóa chính của quan hệ **KHOA**

NỘI DUNG

- Cơ sở dữ liệu
- Hệ quản trị cơ sở dữ liệu
 - Khái niệm
 - Chức năng của hệ QTCSDL
 - Một số hệ QTCSDL thông dụng
- Ngôn ngữ truy vấn SQL

Khái niệm

- Hệ quản trị cơ sở dữ liệu (Database Management System - DBMS): Là một hệ thống phần mềm cho phép tạo lập cơ sở dữ liệu và điều khiển mọi truy nhập đối với cơ sở dữ liệu đó.
- Một số hệ quản trị csdl hiện nay:
 - Microsoft Access
 - Foxpro
 - DB2
 - SQL Server
 - Oracle, ...

Chức năng của hệ quản trị csdl (1)

- Cung cấp môi trường tạo lập cơ sở dữ liệu.
 - Cho phép khai báo kiểu và cấu trúc của dữ liệu, khai báo các ràng buộc trên dữ liệu được lưu trữ trong CSDL.
- Cung cấp môi trường cập nhật và khai thác dữ liệu.
 - *Cập nhật*: Thêm, xóa, sửa dữ liệu.
 - *Khai thác*: Sắp xếp, tìm kiếm, kết xuất báo cáo, ...

Chức năng của hệ quản trị csdl (2)

- Cung cấp công cụ kiểm soát, điều khiển việc truy cập vào CSDL. Hệ quản trị CSDL đảm bảo:
 - Phát hiện và ngăn chặn truy cập trái phép
 - Duy trì tính nhất quán của dữ liệu.
 - Tổ chức, điều khiển các truy cập cùng lúc.
 - Khôi phục CDSL khi gặp sự cố.
 - Quản lí các mô tả dữ liệu.

NỘI DUNG

- Cơ sở dữ liệu
- Hệ quản trị cơ sở dữ liệu
- Ngôn ngữ truy vấn SQL
 - Ngôn ngữ định nghĩa dữ liệu (Data Definition Language – DDL)
 - Ngôn ngữ thao tác dữ liệu (Data Manipulation Language – DML)

Ngôn ngữ truy vấn SQL (1)

- Là ngôn ngữ truy vấn dựa trên đại số quan hệ.
- Cho phép người dùng giao tiếp với CSDL.
- Gồm các loại:
 - **Ngôn ngữ định nghĩa dữ liệu** (Data Definition Language–DDL): Cho phép khai báo cấu trúc các bảng của CSDL, khai báo các mối liên hệ của dữ liệu và các quy tắc áp đặt lên các dữ liệu đó.

Ngôn ngữ truy vấn SQL (2)

- **Ngôn ngữ thao tác dữ liệu** (Data Manipulation Language-DML): Cho phép người sử dụng khai thác CSDL để truy vấn các thông tin cần thiết trong CSDL. Đồng thời cho phép thêm (insert), xoá (delete), sửa (update) dữ liệu trong CSDL
- **Ngôn ngữ quản lý dữ liệu** (Data Control Language-DCL): Bao gồm các câu lệnh đảm bảo tính an toàn và toàn vẹn dữ liệu, cho phép cấp phát quyền truy cập vào dữ liệu

CSDL ví dụ

Bài toán quản lý sinh viên có CSDL gồm các bảng như sau:

KHOA(Makhoa, Tenkhoa, Diadiem, SDT)

LOP(MaLop, TenL, Siso, Makhoa)

SINHVIEN(MaSV, Hodem, Ten, Ngaysinh, Gioitinh,
Tinh, MaLop)

MONHOC(MaMH, TenMH, SoTC, Dieukien)

KETQUA(MaSV, MaMH, Ketqua)

Ngôn ngữ định nghĩa dữ liệu

1. Lệnh tạo bảng
2. Lệnh sửa bảng
 1. Lệnh thêm cột
 2. Lệnh thay đổi kiểu dữ liệu cho cột
 3. Thêm ràng buộc toàn vẹn
 4. Lệnh xóa cột
3. Lệnh xóa bảng

1. Lệnh tạo bảng (1)

- Định nghĩa một bảng cần:
 - Tên bảng
 - Các thuộc tính:
 - Tên thuộc tính
 - Kiểu dữ liệu
 - Ràng buộc toàn vẹn trên thuộc tính đó

CREATE TABLE <tên bảng> (
 <tên cột 1> <kiểu dữ liệu 1> [RBTV] ,
 <tên cột 2> <kiểu dữ liệu 2> [RBTV],
 ...
 <tên cột n> <kiểu dữ liệu n> [RBTV];

1. Lệnh tạo bảng (2)

- Ràng buộc toàn vẹn:
 - NOT NULL
 - NULL
 - IDENTITY
 - UNIQUE
 - DEFAULT
 - PRIMARY KEY
 - FOREIGN KEY/ REFERENCES
 - CHECK

KHOA(MaKhoa, TenKhoa, Diadiem, SDT)**LOP**(MaLop, TenL, Siso, MaKhoa)**SINHVIEN**(MaSV, Hodem, Ten, Ngaysinh,
Gioitinh, Tinh, MaLop)**MONHOC**(MaMH, TenMH, SoTC, Dieukien)**KETQUA**(MaSV, MaMH, Ketqua)

1. Lệnh tạo bảng (3)

- Ví dụ: tạo bảng sinh viên

CREATE TABLE SINHVIEN(

MaSV char(6) not null PRIMARY KEY,

Hodem char(20) not null,

TEN char(7) not null,

Ngaysinh datetime,

gioitinh char(3),

tinh varchar(30),

MaLOP char(8) not null);

2.1 Lệnh thêm cột

ALTER TABLE <tên bảng>

ADD <tên cột> <kiểu dữ liệu> [RBTV];

- Ví dụ: thêm cột chính sách vào bảng SINHVIEN

ALTER TABLE SINHVIEN

ADD Chinh sach char(10);

2.2 Thay đổi kiểu dữ liệu cho cột

ALTER TABLE <tên bảng>

ALTER COLUMN <tên cột> <kiểu dữ liệu mới>;

- Ví dụ: sửa kiểu dữ liệu cột gioitinh trong bảng SINHVIEN từ kiểu char sang kiểu bit [0/1]

```
ALTER TABLE SINHVIEN
```

```
ALTER COLUMN gioitinh bit;
```

2.3 Thêm ràng buộc toàn vẹn (1)

- Thêm khóa chính:

`ALTER TABLE <tên bảng>`

`ADD PRIMARY KEY (DS tên cột);`

- Ví dụ: đặt khóa chính trên cột MaSV của bảng SINHVIEN

`ALTER TABLE SINHVIEN`

`add PRIMARY KEY (masv);`

2.3 Thêm ràng buộc toàn vẹn

- Tạo khóa ngoài:

ALTER TABLE <tên bảng>

ADD FOREIGN KEY (DS tên cột)

REFERENCES <tên bảng> (DS tên cột);

- Ví dụ: Đặt malop là khóa ngoài trên bảng SINHVIEN

ALTER TABLE SINHVIEN

ADD FOREIGN KEY(malop)

REFERENCES LOP (malop);

2.4 Xóa cột

- Lệnh Xóa Cột

`ALTER TABLE <tên bảng>`

`DROP COLUMN <tên cột>;`

- Ví dụ: xóa cột chính sách trong bảng SINHVIEN

`ALTER TABLE SINHVIEN`

`DROP COLUMN Chinhhsach;`

3. Xóa bảng

DROP TABLE <tên bảng>;

- Ví dụ: xóa bảng SINHVIEN

DROP TABLE SINHVIEN;

Ngôn ngữ thao tác dữ liệu

- Nhóm lệnh truy vấn dữ liệu
- Nhóm lệnh cập nhật dữ liệu

Nhóm lệnh truy vấn dữ liệu

1. Cú pháp tổng quát
2. Dạng đơn giản
3. Dấu*
4. Mệnh đề WHERE
5. Truy vấn từ nhiều Bảng
6. Mệnh đề ORDER BY
7. Mệnh đề GROUP BY
8. Mệnh đề HAVING
9. Phát biểu Select với AS
10. Phát biểu Select với DISTINCT
11. Truy vấn con

1. Cú pháp tổng quát

SELECT <Danh sách các cột>

FROM <Danh sách Bảng>

WHERE <Các điều kiện>

GROUP BY <Tên cột>

HAVING <Điều kiện dựa trên GROUP BY>

ORDER BY <Danh sách cột>

2. Dạng đơn giản

SELECT <Danh sách các cột>

FROM <Tên Bảng>

→ lấy ra một số cột trong một bảng nào đó.

- **Ví dụ:** lấy ra mã sinh viên, họ đệm và tên của các sinh viên trong bảng SINHVIEN

```
SELECT MaSV, Hodem, Ten
```

```
FROM SINHVIEN;
```

3. Dấu *

- Dấu * đại diện cho tất cả các cột

SELECT *

FROM <Tên Bảng>

- Ví dụ: Lấy ra danh sách các khoa

SELECT *

FROM KHOA;

4. Mệnh đề WHERE (1)

- Dùng để đặt điều kiện lấy dữ liệu

SELECT <Danh sách các cột>

FROM <TênBảng>

WHERE <Danh sách các điều kiện>

- Ví dụ: lấy ra mã sinh viên, họ đệm và tên của các sinh viên nữ

SELECT MaSV, Hodem, Ten

FROM SINHVIEN

WHERE Gioitinh = "nữ";

4. Mệnh đề WHERE (2)

- Các phép toán trong mệnh đề WHERE:
 - So sánh: >, <, >=, <=, =, <>
 - Logic: And, Or, Not
- Ví dụ: lấy ra mã sinh viên, họ đệm và tên của các sinh viên nữ học lớp K55CNSHA

```
SELECT MaSV, Hodem, Ten
```

```
FROM SINHVIEN
```

```
WHERE Gioitinh = "nữ" and MaLop = "K55CNSHA";
```

4. Mệnh đề WHERE (3)

- Toán tử BETWEEN <giá trị 1> AND <giá trị 2>
- Ví dụ: Lấy ra danh sách các môn học có từ 2 đến 5 tín chỉ

SELECT *

FROM MONHOC

WHERE Sotc between 2 and 5;

4. Mệnh đề WHERE (4)

- Toán tử [NOT] LIKE: tìm một mẫu ký tự
- Sử dụng với các ký tự đại diện: _ (1 ký tự), % (1 chuỗi bất kỳ)
- Ví dụ: lấy ra danh sách Sinh viên Họ 'Nguyễn'

```
SELECT *
```

```
FROM SINHVIEN
```

```
WHERE Hodem LIKE "Nguyễn*";
```

4. Mệnh đề WHERE (5)

- Toán tử [NOT] IN: phép toán so sánh trong 1 tập hợp, 1 danh sách
- Ví dụ: lấy ra thông tin của các lớp thuộc khoa công nghệ thông tin, nông học và chăn nuôi thú y.

SELECT*

FROM LOP

WHERE Makhoa IN ("CNTT", "NH", "CNTY");

5. Truy vấn từ nhiều bảng

- Khi thông tin cần lấy ra có từ nhiều bảng khác nhau, cần thực hiện truy vấn từ nhiều bảng

SELECT <Danh sách các cột>

FROM <Danh sách các Bảng>

WHERE <Cácđiềukiện>

- Lưu ý:
 - Cần liên kết các bảng lại với nhau. Với n bảng cần có n-1 điều kiện liên kết.
 - Các tên cột cùng có ở nhiều bảng cần ghi rõ theo dạng [Tên Bảng].[Tên cột]

5. Truy vấn từ nhiều bảng

- Ví dụ: Hiển thị thông tin về các sinh viên với các kết quả học tập của họ. Thông tin hiển thị cần (mã sinh viên, họ tên, ngày sinh, giới tính, tên môn học, kết quả):

```
SELECT SINHVIEN. MaSV, Hodem, Ten, Ngaysinh, Gioitinh,  
TenMH, Ketqua
```

```
FROM SINHVIEN, KETQUA, MONHOC
```

```
WHERE SINHVIEN.MaSV = KETQUA.MaSV AND  
KETQUA.MaMH = MONHOC.MaMH;
```

6. Mệnh đề ORDER BY

- Sắp xếp kết quả theo thứ tự mong muốn
- ORDER BY <DS Tên cột> [ASC | DESC]
- Ví dụ: lấy ra các sinh viên nữ học lớp K55CNSHA, được sắp xếp theo vần alphabet của họ và tên.

```
SELECT MaSV, Hodem, Ten
```

```
FROM SINHVIEN
```

```
WHERE Gioitinh = "nữ" and MaLop = "K55CNSHA"
```

```
ORDER BY Ten, Hodem;
```

7. Mệnh đề GROUP BY (1)

- Nhóm dữ liệu lại theo từng nhóm để thực hiện các phép toán thống kê

GROUP BY <DS Tên cột>

- Ví dụ: In ra danh sách các lớp và số sinh viên trong mỗi lớp từ bảng SinhVien

```
SELECT SINHVIEN.MaLop, LOP.TenL,  
COUNT(SINHVIEN.MaSV) AS 'So sinh vien'
```

```
FROM SINHVIEN, LOP
```

```
WHERE SINHVIEN.MaLop = LOP.MaLop
```

```
GROUP BY SINHVIEN.MaLop, LOP.TenL;
```

KHOA(Makhoa, Tenkhoa, Diadiem, SDT)**LOP**(MaLop, TenL, Siso, Makhoa)**SINHVIENT**(MaSV, Hodem, Ten, Ngaysinh,
Gioitinh, Tinh, MaLop)**MONHOC**(MaMH, TenMH, SoTC, Dieukien)**KETQUA**(MaSV, MaMH, Ketqua)

7. Mệnh đề GROUP BY (2)

- Một số hàm thông dụng:
 - AVG: giá trị trung bình
 - MIN: giá trị nhỏ nhất
 - MAX: giá trị lớn nhất
 - COUNT: đếm số phần tử
 - SUM: Tổng các phần tử

KHOA(Makhoa, Tenkhoa, Diadiem, SDT)**LOP**(MaLop, TenL, Siso, Makhoa)**SINHVIEN**(MaSV, Hodem, Ten, Ngaysinh,
Gioitinh, Tinh, MaLop)**MONHOC**(MaMH, TenMH, SoTC, Dieukien)**KETQUA**(MaSV, MaMH, Ketqua)

8. Mệnh đề HAVING

- Đặt điều kiện chọn sau khi đã nhóm dữ liệu bằng mệnh đề GROUP BY
- Ví dụ: In ra danh sách các lớp có số sinh viên >2 từ bảng SINHVIEN

```
SELECT SINHVIEN.MaLop, LOP.TenL, COUNT(SINHVIEN.MaSV) AS  
'So sinh vien'
```

```
FROM SINHVIEN, LOP
```

```
WHERE SINHVIEN.MaLop = LOP.MaLop
```

```
GROUP BY SINHVIEN.MaLop, LOP.TenL
```

```
HAVING COUNT(SINHVIEN.MaSV)>=2
```


KHOA(Makhoa, Tenkhoa, Diadiem, SDT)**LOP**(MaLop, TenL, Siso, Makhoa)**SINHVIEN**(MaSV, Hodem, Ten, Ngaysinh,
Gioitinh, Tinh, MaLop)**MONHOC**(MaMH, TenMH, SoTC, Dieukien)**KETQUA**(MaSV, MaMH, Ketqua)

9. Phát biểu Select với AS

- Đặt lại tên Field khi hiển thị kết quả
- Ví dụ: Hiển thị thông tin về các sinh viên với các kết quả học tập của họ. Khi hiển thị cột ketqua đổi tên thành cột DiemTB

```
SELECT SINHVIEN.MaSV, Hodem, Ten, Ngaysinh, Gioitinh, TenMH,  
Ketqua as DiemTB
```

```
FROM SINHVIEN, KETQUA, MONHOC
```

```
WHERE SINHVIEN.MaSV = KETQUA.MaSV AND KETQUA.MaMH =  
MONHOC.MaMH;
```

10. Phát biểu Select với DISTINCT

- Nếu kết quả truy vấn có nhiều bản ghi trùng nhau, để chỉ lấy 1 mẫu tin ta dùng DISTINCT
- Ví dụ: In ra danh sách các lớp trong bảng SINHVIEN

```
SELECT DISTINCT MALOP  
  
FROM SINHVIEN;
```

11. Truy vấn con (1)

- Đôi khi ta cần sử dụng kết quả của 1 câu truy vấn để làm điều kiện cho 1 câu truy vấn khác, khi đó ta gọi là truy vấn con.
- Khi thực hiện, truy vấn con sẽ được thực hiện trước, rồi lấy kết quả để thực hiện truy vấn lớn.

12. Truy vấn con (2)

- Dạng tổng quát:

SELECT <Danh sách các cột>

FROM <Danh sách Bảng>

WHERE <Các điều kiện>

...<Tên cột > IN (NOT IN, =, <>, ...)

(SELECT <Danh sách các cột>

FROM <Danh sách Bảng>

WHERE<Các điều kiện>)

12. Truy vấn con (3)

- Ví dụ: Hiển thị thông tin về những sinh viên đã đăng kí học và không phải thi lại môn học nào.

SELECT Sinhvien.MaSV, hodem, ten, ngaysinh, maMH, ketqua

FROM SINHVIEN, KETQUA

WHERE (SINHVIEN.MaSV=KETQUA.MaSV) AND SINHVIEN.MaSV
IN

(SELECT MaSV

FROM KETQUA

WHERE ketqua>=5);

Nhóm lệnh cập nhật dữ liệu

1. Lệnh thêm bản ghi
2. Lệnh sửa bản ghi
3. Lệnh xóa bản ghi

1. Lệnh thêm bản ghi (1)

INSERT INTO <tên bảng>(<danh sách các cột>)

VALUES (<danh sách các giá trị>)

- Ví dụ: thêm một dòng vào bảng SINHVIEN

INSERT INTO SINHVIEN (MaSV, Hodem, Ten, Ngaysinh,
gioitinh, tinh, MaLoP)

VALUES ("561205", "nguyễn Tiến", "Minh", "3/8/92",
"nam", "Hà Nội", "K57CNSHA");

1. Lệnh thêm bản ghi (2)

- Nếu các giá trị sau từ khoá VALUES hoàn toàn phù hợp về thứ tự với các cột trong bảng, thì danh sách các cột có thể được bỏ qua.
- Ví dụ: thêm một dòng vào bảng SINHVIEN

INSERT INTO SINHVIEN

VALUES ("563495", "Nguyễn Lan", "Phương",
"13/10/1992", "nữ", "Hà Nội", "K56CNTYA");

1. Lệnh thêm bản ghi (3)

- Ta có thể thêm các bản ghi vào bảng từ câu lệnh truy vấn (Select)

INSERT INTO <tên bảng> (<tên cột 1>, ...)

<Câu lệnh Select>

hoặc

INSERT INTO <tên bảng>

<Câu lệnh Select>

1. Lệnh thêm bản ghi (4)

- Ví dụ: thêm tất cả các dòng trong bảng DSSV vào bảng SINHVIEN:

```
INSERT INTO SINHVIEN
```

```
SELECT * FROM DSSV ;
```

- Lưu ý: hai bảng DSSV và SINHVIEN phải có cấu trúc giống nhau

2. Lệnh sửa bản ghi (1)

UPDATE <tên bảng>

SET <tên cột 1> = <giá trị 1>, <tên cột 2> = <giá trị 2> ,

...,

<tên cột n> = < giá trị n>

[WHERE <điều kiện>];

- Ý nghĩa:
 - Giá trị của các cột <tên cột 1>, <tên cột 2> ... Của những bản ghi thoả mãn điều kiện sau WHERE sẽ được sửa đổi thành <giá trị 1>, <giá trị 2>
 - Nếu không có mệnh đề WHERE thì tất cả các bản ghi của bảng sẽ được sửa đổi.

2. Lệnh sửa bản ghi (2)

- Ví dụ: sửa hộ khẩu của các sinh viên từ Hà Tây thành Hà Nội

UPDATE SINHVIEN

SET tinh = "Hà Nội"

WHERE tinh = "Hà Tây";

3. Lệnh xóa bản ghi (1)

DELETE FROM <tên bảng>

[WHERE <điều kiện>];

- ý nghĩa:
 - Các bản ghi thoả mãn điều kiện WHERE sẽ bị xoá khỏi bảng.
 - Nếu không có mệnh đề WHERE thì tất cả các bản ghi của bảng sẽ bị xoá khỏi bảng.

3. Lệnh xóa bản ghi (2)

- Ví dụ: xóa tất cả các bản ghi trong bảng DSSV

```
DELETE FROM DSSV;
```

- Ví dụ: Xóa sinh viên lớp K53MTA khỏi bảng SINHVIEN

```
DELETE FROM SINHVIEN
```

```
WHERE MaLop = "K53MTA";
```