

CHƯƠNG 3: TRUY VẤN DỮ LIỆU BẰNG NGÔN NGỮ SQL

Giảng viên: Dương Quang Huy

NỘI DUNG

1. Truy vấn dữ liệu (Select)
2. Tạo bảng ảo (View)

1. Truy vấn dữ liệu

- Cú pháp:

```
select Tên_cột_1, Tên_cột_2,...
```

```
from Tên_bảng
```

```
where <điều kiện>
```

1.1. Truy vấn đơn giản

- Ví dụ: có một bảng SinhVien

MaSV	TenSV	GioiTinh	SĐT	DiemTB
01	Nguyễn Văn Nam	Nam	0944564646	7
02	Trần Thị Hà	Nữ	09218787891	5
03	Trần Thanh Tuấn	Nam	095454545	4
04	Nguyễn Tấn Sang	Nam	0979878979	8

1.1. Truy vấn đơn giản

1) Lấy ra thông tin của các sinh viên

Cách 1:

```
select *
```

```
from SinhVien
```

Cách 2:

```
Select MaSV, TenSV, GioiTinh, SDT, DiemTB
```

```
From SinhVien
```

1.1 Truy vấn đơn giản

2) Lấy ra họ tên, điểm trung bình của các sinh viên có điểm trung bình > 6

```
Select TenSV, DiemTB
```

```
From SinhVien
```

```
Where DiemTB > 6
```

1.1. Truy vấn đơn giản

- Cú pháp truy vấn dữ liệu từ 2 bảng:

```
select Tên_cột_1, Tên_cột_2,...
```

```
from Tên_bảng_1, Tên_bảng_2
```

```
where Tên_bảng_1.Tên_cột_bảng_1 = Tên_bảng_2.Tên_cột_bảng_2
```

```
and <điều kiện>
```

1.1. Truy vấn đơn giản

- Ví dụ: có 2 bảng SinhVien và bảng Khoa

SinhVien

MaSV	TenSV	GioiTinh	ĐiểmTB	MaKH
01	Nguyễn Văn Nam	Nam	7	CNTT
02	Trần Thị Hà	Nữ	8	KT
03	Trần Thanh Tuấn	Nam	5	CNTT
04	Nguyễn Tấn Sang	Nam	6	VL

Khoa

MaKH	TenKhoa
CNTT	Công nghệ thông tin
VL	Vật lý
KT	Kinh tế
DT	Điện tử

1.1. Truy vấn đơn giản

Đề: Lấy ra mã sinh viên, họ tên, giới tính, tên khoa của các sinh viên có điểm trung bình > 6

```
Select MaSV, TenSV, GioiTinh, TenKhoa
```

```
From SinhVien s, Khoa k
```

```
Where s.MaKH = k.MaKH
```

```
and DiemTB > 6
```

1.1. Truy vấn đơn giản

❖ Mệnh đề **Select**:

□ Chứa các cột, các hàm muốn hiển thị.

□ Nếu cột tồn tại trong hai bảng trong mệnh đề From ta phải ghi rõ:

Tên_Bảng.Tên_Cột

1.1. Truy vấn đơn giản

❖ Các toán tử dùng trên mệnh đề **Select**:

□ Cú pháp: Select [Toán tử]...

Với các toán tử:

- **Distinct**: Loại bỏ các dòng dữ liệu trùng
- **Top n**: Lấy n dòng đầu tiên
- **Top n With Ties**: Lấy những dòng thuộc n cấp đầu tiên
- **Top n Percent**: Lấy N% dòng đầu tiên

1.1. Truy vấn đơn giản

❖ Mệnh đề **From**:

□ Cho biết dữ liệu trên câu truy vấn được lấy từ các bảng nào.

□ Nếu dữ liệu lấy từ nhiều bảng thì các bảng này phải có quan hệ với nhau.

Cú pháp:

```
from Tên_bảng_1, Tên_bảng_2
```

```
where Tên_bảng_1.Tên_cột_bảng_1 = Tên_bảng_2.Tên_cột_bảng_2
```

1.1. Truy vấn đơn giản

❖ Mệnh đề **Where**: Mệnh đề lọc dữ liệu trả về hoặc để gom nhóm.

□ Trong mệnh đề Where có thể sử dụng các toán tử:

➤ Like tìm chuỗi gần đúng

✓ % Thay thế 1 chuỗi.

✓ (Gạch dưới) thay thế 1 ký tự

➤ Toán tử **BETWEEN** Gt1 **AND** GT2

1.2. Sắp xếp (order by)

❖ Cú pháp:

```
select Tên_cột_1, Tên_cột_2,...
```

```
from Tên_bảng
```

```
where <điều kiện>
```

```
order by tên_cột asc/desc
```

Lưu ý: asc: sắp xếp tăng dần, desc: sắp xếp giảm dần

1.2. Sắp xếp (order by)

- Ví dụ: có một bảng SinhVien

MaSV	TenSV	GioiTinh	SĐT	DiemTB
01	Nguyễn Văn Nam	Nam	0944564646	7
02	Trần Thị Hà	Nữ	09218787891	5
03	Trần Thanh Tuấn	Nam	095454545	4
04	Nguyễn Tấn Sang	Nam	0979878979	8

1.2. Sắp xếp (order by)

1) Lấy ra thông tin các sinh viên, với điểm trung bình > 6, sắp xếp theo Mã sinh viên tăng dần

Select *

From SinhVien

Where DiemTB > 6

Order by MaSV asc

1.2. Sắp xếp (order by)

2) Lấy ra thông tin các sinh viên, với điểm trung bình > 6, sắp xếp theo Tên

Sinh viên giảm dần

Select *

From SinhVien

Where DiemTB > 6

Order by TenSV desc

1.3. Gom nhóm (Group By)

❖ Mệnh đề **Group by**: Gom nhóm dữ liệu.

❑ Mệnh đề này phải có khi trong **câu truy vấn có sử dụng hàm thống kê**

❑ Các Hàm thống kê:

SUM(TênCột): Tính tổng với cột có kiểu số.

COUNT(TênCột / *): Đếm số dòng.

AVG(TênCột): Tính giá trị trung bình.

MAX(TênCột): Giá trị lớn nhất.

MIN(TênCột): Giá trị nhỏ nhất.

1.3. Gom nhóm (Group By)

❖ Mệnh đề **Group by**: Gom nhóm dữ liệu.

□ Cú pháp:

```
select Tên_cột_1, Tên_cột_2,...
```

```
from Tên_bảng
```

```
where <điều kiện>
```

```
Group by Tên_cột_1, Tên_cột_2,...
```

1.3. Gom nhóm (Group By)

❖ Ví dụ: Có 1 bảng Nhân Viên

NhanVien

MaNV	TenNV	GioiTinh	QueQuan
01	Hải Lâm	Nam	HCM
02	Thu Thảo	Nữ	Hà Nội
03	Mai Thi	Nữ	Hà Nội
04	Quang Vinh	Nam	HCM
05	Quang Hiển	Nam	Long An

1.3. Gom nhóm (Group By)

1) Hiện ra số lượng các nhân viên ứng với từng quê

```
select QueQuan, count(*)
```

```
from NhanVien
```

```
group by QueQuan
```

2) Đếm số nam và số nữ trong công ty

```
select GioiTinh, count(*)
```

```
from NhanVien
```

```
group by GioiTinh
```

1.3. Gom nhóm (Group By)

3) Tính tổng thu nhập theo từng phòng

```
select Phong, sum(Luong)
```

```
from NhanVien
```

```
group by Phong
```

1.3. Gom nhóm (Group By)

Ví dụ khác: Cho biết thông tin khách hàng và số lần mua hàng của mỗi khách hàng

```
SELECT K.MSKH, K.TENKH, K.DIACHI, COUNT(*) AS SOLAN_MUA
FROM KHACHHANG K, HOADON H
WHERE K.MSKH=H.MSKH
GROUP BY K.MSKH, K.TENKH, K.DIACHI
```

3. Gom nhóm (Group By)

❖ Mệnh đề **Having**: Lọc nhóm dữ liệu

❑ Mệnh đề này chỉ tồn tại khi trong câu truy vấn có mệnh đề **Group By** và điều kiện lọc dữ liệu có **hàm thống kê**

❑ Cú pháp:

```
select Tên_cột_1, Tên_cột_2,...
```

```
from Tên_bảng
```

```
where <điều kiện>
```

```
Group by Tên_cột_1, Tên_cột_2,...
```

```
Having <Điều kiện lọc nhóm>
```

1.3. Gom nhóm (Group By)

Ví dụ 1: Chỉ hiện ra những phòng nào có tổng thu nhập lớn hơn 500000

```
select Phong, sum(Luong)
```

```
from NhanVien
```

```
group by Phong
```

```
having sum(Luong) > 500000
```

1.3. Gom nhóm (Group By)

Ví dụ 2: Chỉ hiện ra những tỉnh nào có số lượng người lớn hơn 10

```
select QueQuan, count(*)
```

```
from NhanVien
```

```
group by QueQuan
```

```
having count(*) > 10
```

1.4. Truy vấn lồng

- Truy vấn lồng là một truy vấn con nằm bên trong một truy vấn khác trong các hệ quản trị cơ sở dữ liệu (như Sql Server).
- Truy vấn lồng được sử dụng để giải quyết các bài toán phức tạp hơn bằng cách chia nhỏ chúng thành nhiều phần và sử dụng kết quả của truy vấn con (subquery) là điều kiện để so sánh, gán giá trị cho trong truy vấn chính (main query).



1.4. Truy vấn lồng

Ví dụ: Cho bảng nhân viên

NhanVien

MaNV	HoTen	SDT	Email	Luong
1	Phạm An	0945454	an@gmail.com	500000
2	Bằng Hữu	0456454	huu@gmail.com	400000
3	Thế An	04646464	an2@gmail.com	650000
4	Nhi	0965654	nhi@gmail.com	300000

1.4. Truy vấn lồng

Yêu cầu: Hiện tất cả những người trong bảng nhân viên có lương bằng lương lớn nhất của những người có trong công ty:

```
select Hoten, Luong
```

```
from NhanVien
```

```
where Luong = (select max(Luong) from NhanVien)
```

1.4. Truy vấn lồng

Giải thích:

Bước 1: Tìm lương lớn nhất từ bảng nhân viên

```
select max(Luong)  
from NhanVien
```

1.4. Truy vấn lồng

Giải thích:

Bước 2: Tìm thông tin các nhân viên có lương bằng lương lớn nhất tìm được ở bước 1

```
select Hoten, Luong
```

```
from NhanVien
```

```
where Luong = (select max(Luong) from NhanVien)
```

1.4. Truy vấn lồng

- **Kiểm tra:**

Kết quả khi xuất dữ liệu từ bảng NhanVien từ câu truy vấn lồng:

HoTen	Luong
Thế An	650000

1.4. Truy vấn lồng

Ví dụ 2: Có 2 bảng Employees và Departments

Employees

	id	name	email	hire_date	salary	department_id
1	101	Doe	john.doe@example.com	2020-01-15	5000.00	1
2	102	Smith	jane.smith@example.com	2021-03-12	4500.00	2
3	103	Johnson	emily.johnson@example.com	2022-05-20	4800.00	3
4	104	Brown	michael.brown@example.com	2019-11-01	5200.00	1
5	105	Davis	sarah.davis@example.com	2021-09-22	4600.00	4

Departments

Id	Name
1	Phòng Ban 1
2	Phòng Ban 2
3	Phòng Ban 3

1.4. Truy vấn lồng

Yêu cầu: tìm tên của những nhân viên làm việc ở phòng ban có mã là 1

```
SELECT name
FROM Employees
WHERE department_id = (SELECT id
                       FROM Departments
                       WHERE name = 'Phòng ban 1')
```

1.4. Truy vấn lồng

Giải thích:

Bước 1: *Tìm id của phòng ban có tên là 'Phòng ban 1'*

SELECT id

FROM Departments

WHERE name = 'Phòng ban 1'

1.4. Truy vấn lồng

Giải thích:

Bước 2: *Tìm tên của các nhân viên làm việc ở phòng ban có department_id ở Bước 1.*

```
SELECT name
FROM Employees
WHERE department_id = (SELECT id
                       FROM Departments
                       WHERE name = 'Phòng ban 1');
```

1.4. Truy vấn lồng

Kết quả:

Kết quả khi xuất dữ liệu từ bảng Employees từ câu truy vấn lồng:

	name	department_id
1	Doe	1
2	Brown	1

1.4. Truy vấn lồng

❖ Các loại truy vấn lồng

1. Truy vấn con trả về một giá trị

SELECT<danh sách các cột>

FROM<danh sách các bảng>

WHERE tên cột <so sánh tập hợp> (SELECT<danh sách các cột>
FROM<danh sách các bảng>
WHERE<điều kiện>)

So sánh tập hợp: =, >, >=, <, <=, <>

1.4. Truy vấn lồng

❖ Các loại truy vấn lồng

2. Truy vấn con trả về nhiều giá trị, sử dụng in, all, any

SELECT<danh sách các cột>

FROM<danh sách các bảng>

WHERE tên cột <so sánh tập hợp> (SELECT<danh sách các cột>
FROM<danh sách các bảng>
WHERE<điều kiện>)

So sánh tập hợp: in, all, any.

1.4. Truy vấn lồng

❖ Các loại truy vấn lồng

3. Phép trừ

SELECT<danh sách các cột>

FROM<danh sách các bảng>

WHERE tên cột <so sánh tập hợp> (SELECT<danh sách các cột>
FROM<danh sách các bảng>
WHERE<điều kiện>)

So sánh tập hợp: NOT IN, NOT EXIST

1.4. Truy vấn lồng

❖ Ví dụ 1: Có 2 bảng NhanVien và Phong

NhanVien

MaNV	HoTen	SDT	MaPhong
01	Phạm Văn Đồng	0454646	1
02	Nguyễn Tuấn Hải	7897987	2
03	Phan Văn Tuấn	41313154	1
04	Lý Kim Huệ	454612231	3

Phong

MaPhong	Ten Phong	DiaDiem
1	Bán hàng	TPHCM
2	Dịch vụ	Hà Nội
3	Lập trình	Vũng Tàu
4	Sản xuất	TPHCM

1.4. Truy vấn lồng

Yêu cầu: Tìm họ và tên, SDT của các nhân viên trực thuộc các phòng ban ở địa bàn TP HCM.

Bước 1: Tìm tất cả các phòng có địa điểm TP HCM

```
SELECT MaPhong  
FROM Phong  
WHERE DiaDiem = 'TPHCM'
```

1.4. Truy vấn lồng

Bước 2: Tìm nhân viên gồm họ tên, sdt trực thuộc các phòng trong bước 1

```
SELECT HoTen, SDT
FROM NHANVIEN
WHERE MaPhong IN (
    SELECT MaPhong
    FROM Phong
    WHERE DiaDiem = 'TPHCM'
)
```

1.4. Truy vấn lồng

Kết quả: Từ câu truy vấn con ở Bước 1

MaPhong
1
4

1.4. Truy vấn lồng

Kết quả: Từ câu truy vấn Bước 2

HoTen	SDT
Phạm Văn Đồng	0454646
Phan Văn Tuấn	41313154

1.4. Truy vấn lồng

❖ Ví dụ 2: Có 2 bảng NhanVien và PhanCong

NhanVien

MaNV	HoTen	SDT	MaPhong
01	Phạm Văn Đồng	0454646	1
02	Nguyễn Tuấn Hải	7897987	2
03	Phan Văn Tuấn	41313154	1
04	Lý Kim Huệ	454612231	3

PhanCong

SoDA	MaNV	DiaDiem
10	01	TPHCM
10	02	Hà Nội
10	04	Vũng Tàu
03	01	TPHCM

1.4. Truy vấn lồng

Yêu cầu: Tìm họ tên, sdt của các nhân viên không tham gia vào đề án 10.

Bước 1: Tìm tất cả các nhân viên tham gia vào đề án 10

```
SELECT MaNV  
FROM PHANCONG  
WHERE SODA = 10
```

1.4. Truy vấn lồng

Bước 2: Tìm nhân viên không thuộc danh sách trong bước 1

```
SELECT HoTen, SDT
FROM NHANVIEN
WHERE MaNV NOT IN (
    SELECT MaNV
    FROM PHANCONG
    WHERE SODA = 10
)
```

1.4. Truy vấn lồng

Kết quả: Từ câu truy vấn con ở Bước 1

MaNV
01
02
04

1.4. Truy vấn lồng

Kết quả: Từ câu truy vấn Bước 2

HoTen	SDT
Phạm Văn Đồng	0454646
Nguyễn Tuấn Hải	7897987
Lý Kim Huệ	454612231

1.4. Truy vấn lồng

❖ Ví dụ 3: Có 1 bảng NhanVien

NhanVien

MaNV	HoTen	Luong	MaPhong
01	Phạm Văn Đồng	600000	1
02	Nguyễn Tuấn Hải	500000	4
03	Phan Văn Tuấn	400000	2
04	Lý Kim Huệ	300000	4

1.4. Truy vấn lồng

Yêu cầu: Tìm nhân viên có lương lớn hơn lương của tất cả nhân viên phòng 4.

Bước 1: Tìm lương tất cả nhân viên phòng 4

```
SELECT LUONG  
FROM NHANVIEN  
WHERE MaPhong = 4
```

1.4. Truy vấn lồng

Bước 2: Tìm nhân viên có mức lương lớn hơn tất cả các mức lương trong bước 1

```
SELECT HoTen, Luong
FROM NHANVIEN
WHERE LUONG > ALL (
    SELECT LUONG
    FROM NHANVIEN
    WHERE MaPhong = 4
)
```

1.4. Truy vấn lồng

Kết quả: Từ câu truy vấn con ở Bước 1

Luong
500000
300000

1.4. Truy vấn lồng

Kết quả: Từ câu truy vấn Bước 2

HoTen	Luong
Phạm Văn Đồng	600000

1.4. Truy vấn lồng

Ví dụ 4: Tìm nhân viên có lương lớn hơn lương ít nhất một nhân viên phòng 4

Bước 1: Tìm lương của tất cả nhân viên phòng 4

```
SELECT LUONG  
FROM NHANVIEN  
WHERE MaPhong = 4
```

1.4. Truy vấn lồng

Bước 2: Tìm nhân viên có mức lương lớn hơn ít nhất một mức lương trong bước 1

```
SELECT HoTen, Luong
FROM NHANVIEN
WHERE LUONG > ANY (
    SELECT LUONG
    FROM NHANVIEN
    WHERE MaPhong = 4
)
```

1.4. Truy vấn lồng

Kết quả: Từ câu truy vấn con ở Bước 1

Luong
500000
300000

1.4. Truy vấn lồng

Kết quả: Từ câu truy vấn Bước 2

HoTen	Luong
Phạm Văn Đồng	600000
Nguyễn Tuấn Hải	500000
Phan Văn Tuấn	400000

2. Bảng ảo(View)

❖ Định Nghĩa:

- Bảng ảo là một đối tượng chứa câu lệnh **Select** lấy dữ liệu từ các bảng trong csdl.
- Khi truy cập vào CSLD, người sử dụng không phân biệt được bảng ảo và bảng thật.
- Có thể thêm, xóa, sửa được dữ liệu trong bảng ảo.

2. Bảng ảo(View)

❖ Cú pháp tạo bảng ảo:

```
CREATE VIEW Tên_Bang_ao [(tên_các_cột)]  
[WITH ENCRYPTION] --mã hoá nội dung câu lệnh  
AS    Câu lệnh SELECT
```

❖ Cú pháp xóa bảng ảo:

```
DROP VIEW tên_Bảng_ảo
```

2. Bảng ảo(View)

- ❖ **Ví dụ:** Danh sách những mặt hàng có đơn giá > 10000.

```
CREATE VIEW V_DonGia_10000
AS
    SELECT MSMH, TENMH, DONVITINH, DONGIA
    FROM MATHANG
    WHERE DONGIA > 10000;
```

- ❖ **Xem kết quả của View:**

```
Select * from <Ten_View>
```

- ❖ **Ví dụ:** `SELECT *FROM V_DonGia_10000;`

2. Bảng ảo(View)

- ❖ **Ví dụ:** Danh sách những mặt hàng có đơn giá > 10000.

```
CREATE VIEW V_DonGia_10000 ([Mã MH], [Tên MH])  
WITH ENCRYPTION AS  
    SELECT MSMH, TENMH  
    FROM MATHANG  
    WHERE DONGIA > 10000;
```

- ❖ Xem lại nội dung câu lệnh **SELECT** bên trong View

```
EXEC Sp_helptext V_DonGia_10000;
```



Messages

```
The text for object 'V_DonGia_10000' is encrypted.
```