



Chương 4

Ngôn ngữ SQL

Nội dung chi tiết



- Giới thiệu
- Định nghĩa dữ liệu
- Truy vấn dữ liệu
 - Truy vấn cơ bản
 - **Tập hợp, so sánh tập hợp và truy vấn lồng**
 - Hàm kết hợp và gom nhóm
 - Một số dạng truy vấn khác
- Cập nhật dữ liệu
- Khung nhìn (view)
- Chỉ mục (index)

Phép toán tập hợp trong SQL



- SQL có cài đặt các phép toán
 - Hội (UNION)
 - Giao (INTERSECT)
 - Trừ (EXCEPT)

- Kết quả trả về là tập hợp
 - Loại bỏ các bộ trùng nhau
 - Để giữ lại các bộ trùng nhau
 - UNION ALL
 - INTERSECT ALL
 - EXCEPT ALL

Phép toán tập hợp trong SQL



■ Cú pháp

```
SELECT <ds cột> FROM <ds bảng> WHERE <điều kiện>
```

UNION [ALL]

```
SELECT <ds cột> FROM <ds bảng> WHERE <điều kiện>
```

```
SELECT <ds cột> FROM <ds bảng> WHERE <điều kiện>
```

INTERSECT [ALL]

```
SELECT <ds cột> FROM <ds bảng> WHERE <điều kiện>
```

```
SELECT <ds cột> FROM <ds bảng> WHERE <điều kiện>
```

EXCEPT [ALL]

```
SELECT <ds cột> FROM <ds bảng> WHERE <điều kiện>
```

Ví dụ 5



- Cho biết các mã đề án có
 - Nhân viên với họ là 'Nguyen' tham gia hoặc,
 - Trưởng phòng chủ trì đề án đó với họ là 'Nguyen'

SELECT SODA

FROM NHANVIEN, PHANCONG

WHERE MANV=MA_NVIENT AND HONV='Nguyen'

UNION

SELECT MADA

FROM NHANVIEN, PHONGBAN, DEAN

WHERE MANV=TRPHG AND MAPHG=PHONG

AND HONV='Nguyen'

Ví dụ 6



- Tìm nhân viên có người thân cùng tên và cùng giới tính

```
SELECT TENNV, PHAI, MANV FROM NHANVIEN
```

```
INTERSECT
```

```
SELECT TENTN, PHAI, MA_NVIENTHANNHAN
```

```
SELECT NV.*
```

```
FROM NHANVIEN NV, THANNHAN TN
```

```
WHERE NV.MANV=TN.MA_NVIENTHANNHAN
```

```
AND NV.TENTN=TN.TENTN AND NV.PHAI=TN.PHAI
```

Ví dụ 7



- Tìm những nhân viên không có thân nhân nào

```
SELECT MANV FROM NHANVIEN
```

~~EXCEPT~~

```
SELECT MA_NVIENT AS MANV FROM THANNHAN
```

Truy vấn lồng



```
SELECT MANV, TENNV  
FROM NHANVIEN, PHONGBAN  
WHERE TENPHG='Nghien cuu' AND PHG=MAPHG
```

Câu truy vấn cha
(Outer query)

```
SELECT <danh sách các cột>  
FROM <danh sách các bảng>  
WHERE <so sánh tập hợp> (
```

```
SELECT <danh sách các cột>  
FROM <danh sách các bảng>  
WHERE <điều kiện>)
```

Câu truy vấn con
(Subquery)

Truy vấn lồng (tt)



- Các câu lệnh SELECT có thể lồng nhau ở nhiều mức
- Các câu truy vấn con trong cùng một mệnh đề WHERE được kết hợp bằng phép nối logic
- Câu truy vấn con thường trả về một tập các giá trị
- Mệnh đề WHERE của câu truy vấn cha
 - <biểu thức> <so sánh tập hợp> <truy vấn con>
 - So sánh tập hợp thường đi cùng với một số toán tử
 - IN, NOT IN
 - ALL
 - ANY hoặc SOME
 - Kiểm tra sự tồn tại
 - EXISTS
 - NOT EXISTS

Truy vấn lồng (tt)



■ Có 2 loại truy vấn lồng

- Lồng phân cấp

- Mệnh đề WHERE của truy vấn con không tham chiếu đến thuộc tính của các quan hệ trong mệnh đề FROM ở truy vấn cha
- Khi thực hiện, câu truy vấn con sẽ được thực hiện trước

- Lồng tương quan

- Mệnh đề WHERE của truy vấn con tham chiếu ít nhất một thuộc tính của các quan hệ trong mệnh đề FROM ở truy vấn cha
- Khi thực hiện, câu truy vấn con sẽ được thực hiện nhiều lần, mỗi lần tương ứng với một bộ của truy vấn cha

Ví dụ - Lồng phân cấp



```
SELECT MANV, TENNV  
FROM NHANVIEN, DIADIEM_PHG  
WHERE DIADIEM='TP HCM' AND PHG=MAPHG
```

```
SELECT MANV, TENNV  
FROM NHANVIEN  
WHERE PHG IN (1, 5)
```

```
SELECT MAPHG  
FROM DIADIEM_PHG  
WHERE DIADIEM='TP HCM' )
```

Ví dụ 5



SELECT DISTINCT TENDA

FROM DEAN

WHERE MADA IN (

SELECT SODA

FROM NHANVIEN, PHANCONG

WHERE MANV=MA_NVIENT AND HONV='Nguyen')

OR MADA IN (

SELECT MADA

FROM NHANVIEN, PHONGBAN, DEAN

WHERE MANV=TRPHG AND MAPHG=PHONG
AND HONV='Nguyen')

Ví dụ 7



- Tìm những nhân viên không có thân nhân nào

```
SELECT *  
FROM NHANVIEN  
WHERE MANV NOT IN (  
    SELECT MA_NVIAN  
    FROM THANNHAN )
```

```
SELECT *  
FROM NHANVIEN  
WHERE MANV <> ALL (  
    SELECT MA_NVIAN  
    FROM THANNHAN )
```

Ví dụ 8



- Tìm những nhân viên có lương lớn hơn lương của ít nhất một nhân viên phòng 4

```
SELECT *  
FROM NHANVIEN  
WHERE LUONG > ANY (  
    SELECT LUONG  
    FROM NHANVIEN  
    WHERE PHG=4 )
```

```
SELECT NV1.*  
FROM NHANVIEN NV1, NHANVIEN NV2  
WHERE NV1.LUONG > NV2.LUONG AND NV2.PHG=4
```

Ví dụ 9



- Tìm những nhân viên có lương lớn hơn lương của tất cả nhân viên phòng 4

```
SELECT *  
FROM NHANVIEN  
WHERE LUONG > ALL (  
    SELECT LUONG  
    FROM NHANVIEN  
    WHERE PHG=4 )
```

Ví dụ 10



- Tìm những trưởng phòng có tối thiểu một thân nhân

```
SELECT *  
FROM NHANVIEN  
WHERE MANV IN (SELECT MA_NVIEEN FROM THANNHAN)  
AND MANV IN (SELECT TRPHG FROM PHONGBAN)
```


Ví dụ - Lồng tương quan



```
SELECT MANV, TENNV  
FROM NHANVIEN, PHONGBAN  
WHERE TENPHG='Nghien cuu' AND PHG=MAPHG
```

```
SELECT MANV, TENNV  
FROM NHANVIEN  
WHERE EXISTS (  
    SELECT *  
    FROM PHONGBAN  
    WHERE TENPHG='Nghien cuu' AND PHG=MAPHG )
```

Ví dụ 6



- Tìm nhân viên có người thân cùng tên và cùng giới tính

```
SELECT *  
FROM NHANVIEN NV  
WHERE EXISTS (  
    SELECT *  
    FROM THANNHAN TN  
    WHERE NV.MANV=TN.MA_NVIE  
    AND NV.TENNV=TN.TENTN  
    AND NV.PHAI=TN.PHAI )
```

Ví dụ 7



- Tìm những nhân viên không có thân nhân nào

```
SELECT *  
FROM NHANVIEN  
WHERE NOT EXISTS (  
    SELECT *  
    FROM THANNHAN  
    WHERE MANV=MA_NVIEN)
```

Ví dụ 8



- Tìm những nhân viên có lương lớn hơn lương của ít nhất một nhân viên phòng 4

SELECT *

FROM NHANVIEN NV1

WHERE EXISTS (

SELECT *

FROM NHANVIEN NV2

WHERE NV2PHG=4

AND NV1.LUONG>NV2.LUONG)

Ví dụ 10



- Tìm những trưởng phòng có tối thiểu một thân nhân

```
SELECT *  
FROM NHANVIEN  
WHERE EXISTS (  
    SELECT *  
    FROM THANNHAN  
    WHERE MANV=MA_NVIENT )  
AND EXISTS (  
    SELECT *  
    FROM PHONGBAN  
    WHERE MANV=TRPHG )
```

Nhận xét IN và EXISTS



■ IN

- <tên cột> IN <câu truy vấn con>
- Thuộc tính ở mệnh đề SELECT của truy vấn con phải có cùng kiểu dữ liệu với thuộc tính ở mệnh đề WHERE của truy vấn cha

■ EXISTS

- Không cần có thuộc tính, hằng số hay biểu thức nào khác đứng trước
- Không nhất thiết liệt kê tên thuộc tính ở mệnh đề SELECT của truy vấn con
- Những câu truy vấn có = ANY hay IN đều có thể chuyển thành câu truy vấn có EXISTS

Phép chia trong SQL



R	A	B	C	D	E
	α	a	α	a	1
	α	a	γ	a	1
	α	a	γ	b	1
	β	a	γ	a	1
	β	a	γ	b	3
	γ	a	γ	a	1
	γ	a	γ	b	1
	γ	a	β	b	1

S	D	E
b_i	a	1
	b	1

$R \div S$	A	B	C
a_i	α	a	γ
	γ	a	γ

- $R \div S$ là tập các giá trị a_i trong R sao cho không có giá trị b_i nào trong S làm cho bộ (a_i, b_i) không tồn tại trong R

Phép chia trong SQL (tt)



- Sử dụng NOT EXISTS để biểu diễn

```
SELECT R1.A, R1.B, R1.C
```

```
FROM R R1
```

```
WHERE NOT EXISTS (
```

```
    SELECT *
```

```
    FROM S
```

```
    WHERE NOT EXISTS (
```

```
        SELECT *
```

```
        FROM R R2
```

```
        WHERE R2.D=S.D AND R2.E=S.E
```

```
        AND R1.A=R2.A AND R1.B=R2.B AND R1.C=R2.C ))
```


Ví dụ 11



- Tìm tên các nhân viên được phân công làm tất cả các đề án
 - Tìm tên các nhân viên mà không có đề án nào là không được phân công làm
 - Tập bị chia: PHANCONG(MA_NVIEN, SODA)
 - Tập chia: DEAN(MADA)
 - Tập kết quả: KQ(MA_NVIEN)
 - Kết KQ với NHANVIEN để lấy ra TENNV

Ví dụ 11 (tt)



```
SELECT NV.TENNV
FROM NHANVIEN NV, PHANCONG PC1
WHERE NV.MANV=PC1.MA_NVIENT
AND NOT EXISTS (
    SELECT *
    FROM DEAN DA
    WHERE NOT EXISTS (
        SELECT *
        FROM PHANCONG PC2
        WHERE PC2.SODA=DA.MADA
        AND PC1.MA_NVIENT=PC2.MA_NVIENT ))
```

