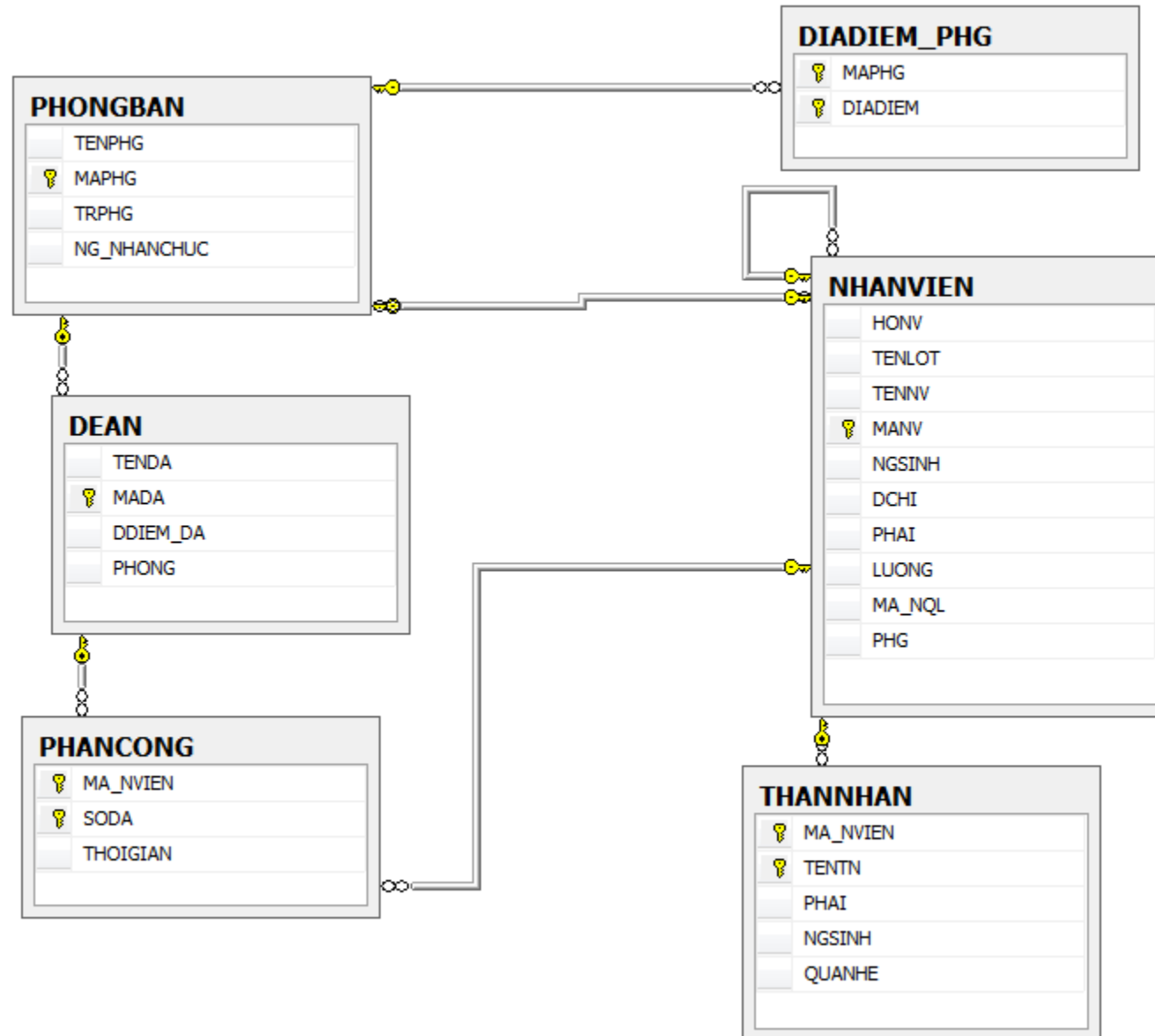


Đề án công ty Truy vấn dữ liệu



Truy vấn cơ bản

- Gồm 3 mệnh đề

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

- <danh sách các cột>
 - Tên các cột cần được hiển thị trong kết quả truy vấn
- <danh sách các bảng>
 - Tên các bảng liên quan đến câu truy vấn
- <điều kiện>
 - Biểu thức boolean xác định dòng nào sẽ được rút trích
 - Nối các biểu thức: AND, OR, và NOT
 - Phép toán: < , > , ≤ , ≥ , ≠ , =, LIKE và BETWEEN

Truy vấn cơ bản (tt)

- SQL

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

Ví dụ

Lấy tất cả các cột của
quan hệ kết quả

```
SELECT *  
FROM NHANVIEN  
WHERE PHG=5
```

MANV	HONV	TENLOT	TENNV	NGSINH	DCHI	PHAI	LUONG	MA_NQL	PHG
333445555	Nguyen	Thanh	Tung	12/08/1955	638 NVC Q5	Nam	40000	888665555	5
987987987	Nguyen	Manh	Hung	09/15/1962	Ba Ria VT	Nam	38000	333445555	5

Tìm nhân viên làm việc ở phòng số 5

Mệnh đề SELECT

```
SELECT MANV, HONV, TENLOT, TENNV  
FROM NHANVIEN  
WHERE PHG=5 AND PHAI='Nam'
```

MANV	HONV	TENLOT	TENNV
333445555	Nguyen	Thanh	Tung
987987987	Nguyen	Manh	Hung

Tìm nam nhân viên làm việc ở phòng số 5

Mệnh đề SELECT (tt)

Tên bí danh

```
SELECT MANV, HONV AS HO, TENLOT AS 'TEN LOT', TENNV AS TEN  
FROM NHANVIEN  
WHERE PHG=5 AND PHAI='Nam'
```

MANV	HO	TEN LOT	TEN
333445555	Nguyen	Thanh	Tung
987987987	Nguyen	Manh	Hung

Tìm nam nhân viên làm việc ở phòng số 5

Mệnh đề SELECT (tt)

Mở rộng

```
SELECT MANV, HONV + ' ' + TENLOT + ' ' + TENNV AS 'HO TEN'  
FROM NHANVIEN  
WHERE PHG=5 AND PHAI='Nam'
```

MANV	HO TEN
333445555	Nguyen Thanh Tung
987987987	Nguyen Manh Hung

Tìm name nhân viên làm việc ở phòng số 5

Mệnh đề SELECT (tt)

Mở rộng

```
SELECT MANV, LUONG*1.1 AS 'LUONG10%'  
FROM NHANVIEN  
WHERE PHG=5 AND PHAI='Nam'
```

MANV	LUONG10%
333445555	33000
987987987	27500

Tăng 10% lương cho nam nhân viên làm việc ở phòng số 5

Mệnh đề SELECT (tt)

Loại bỏ các dòng trùng nhau

```
SELECT DISTINCT LUONG  
FROM NHANVIEN  
WHERE PHG=5 AND PHAI='Nam'
```

LUONG

30000

25000

38000

38000

- Tồn chi phí

- Người dùng muốn thấy

Ví dụ

- Cho biết MANV và TENNV làm việc ở phòng 'Nghien cuu'

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

Mệnh đề WHERE

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

Biểu thức luận lý

↓ TRUE ↓ TRUE

Mệnh đề WHERE (tt)

Độ ưu tiên

```
SELECT MANV, TENNV  
FROM NHANVIEN, PHONGBAN  
WHERE (TENPHG='Nghien cuu' OR TENPHG='Quan ly') AND PHG=MAPHG
```

Cho biết MANV và TENNV làm việc ở phòng
'Nghien cuu' hoặc 'Quan ly'

Mệnh đề WHERE (tt)

BETWEEN

```
SELECT MANV, TENNV  
FROM NHANVIEN  
WHERE LUONG >= 20000 AND LUONG <= 30000
```

```
SELECT MANV, TENNV  
FROM NHANVIEN  
WHERE LUONG BETWEEN 20000 AND 30000
```

Cho biết MANV và TENNV có mức lương từ 20000 đến 300000

Mệnh đề WHERE (tt)

NOT BETWEEN

```
SELECT MANV, TENNV  
FROM NHANVIEN  
WHERE LUONG NOT BETWEEN 20000 AND 30000
```

Mệnh đề WHERE (tt)

LIKE

```
SELECT MANV, TENNV
```

```
FROM NHANVIEN
```

```
WHERE DCHI LIKE 'Nguyen _ _ _ _ '
```

Ký tự bất kỳ

```
SELECT MANV, TENNV
```

```
FROM NHANVIEN
```

```
WHERE DCHI LIKE 'Nguyen %'
```

Chuỗi bất kỳ

Mệnh đề WHERE (tt)

NOT LIKE

```
SELECT MANV, TENNV  
FROM NHANVIEN  
WHERE HONV LIKE 'Nguyễn'
```

```
SELECT MANV, TENNV  
FROM NHANVIEN  
WHERE HONV NOT NOT LIKE 'Nguyễn'
```

Mệnh đề WHERE (tt)

Ngày giờ

SELECT MANV, TENNV

FROM NHANVIEN

WHERE NGSINH BETWEEN '1955-12-08' AND '1966-07-19'

'1955-12-08'	YYYY-MM-DD	'17:30:00'	HH:MI:SS
--------------	------------	------------	----------

'12/08/1955'	MM/DD/YYYY	'05:30 PM'	
--------------	------------	------------	--

'December 8, 1955'

'1955-12-08 17:30:00'

Mệnh đề WHERE (tt)

NULL

- Sử dụng trong trường hợp
 - Không biết (value unknown)
 - Không thể áp dụng (value inapplicable)
 - Không tồn tại (value withheld)
- Những biểu thức tính toán có liên quan đến giá trị NULL sẽ cho ra kết quả là NULL
 - x có giá trị là NULL
 - $x + 3$ cho ra kết quả là NULL
 - $x + 3$ là một biểu thức không hợp lệ trong SQL
- Những biểu thức so sánh có liên quan đến giá trị NULL sẽ cho ra kết quả là UNKNOWN
 - $x = 3$ cho ra kết quả là UNKNOWN
 - $x = 3$ là một so sánh không hợp lệ trong SQL

Mệnh đề WHERE (tt)

NULL

```
SELECT MANV, TENNV  
FROM NHANVIEN  
WHERE MA_NQL IS NULL
```

```
SELECT MANV, TENNV  
FROM NHANVIEN  
WHERE MA_NQL IS NOT NULL
```

Cho biết MANV và TENNV không có người quản lý

Mệnh đề FROM

Không sử dụng mệnh đề WHERE

```
SELECT MANV, MAPHG  
FROM NHANVIEN, PHONGBAN  
WHERE TRUE
```

MANV	MAPHG
333445555	1
333445555	4
333445555	5
987987987	1
987987987	4
987987987	5
...	...

Mệnh đề FROM (tt)

Tên bí danh

Với mỗi phòng ban, cho biết tên phòng ban và địa điểm phòng

```
SELECT TENPHG, DIADIEM
FROM PHONGBAN, AS DIADIEM PHG AS DD
WHERE MA_PHG=MA_DIADIEM MA_PHG
```

```
SELECT TENNV, NGSINH, TEN, TEN, NGSINH
FROM NHANVIEN, NHANVIEN TN
WHERE MANV=MA_NVN
```

Ví dụ 1

- Với những đề án ở 'Ha Noi', cho biết mã đề án, mã phòng ban chủ trì đề án, họ tên trưởng phòng cùng với ngày sinh và địa chỉ của người ấy

Ví dụ 2

- Tìm họ tên của nhân viên phòng số 5 có tham gia vào đề án “Sản phẩm X” với số giờ làm việc trên 10 giờ

Ví dụ 3

- Tìm họ tên của từng nhân viên và người phụ trách trực tiếp nhân viên đó

Ví dụ 4

- Tìm họ tên của những nhân viên được “Nguyen Thanh Tung” phụ trách trực tiếp

Mệnh đề ORDER BY

- Dùng để hiển thị kết quả câu truy vấn theo một thứ tự nào đó
- Cú pháp

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

- ASC: tăng (mặc định)
- DESC: giảm

Mệnh đề ORDER BY (tt)

- Ví dụ

```
SELECT MA_NVIEN, SODA  
FROM PHANCONG  
ORDER BY MA_NVIEN DESC, SODA
```

MA_NVIEN	SODA
999887777	10
999887777	30
987987987	10
987987987	30
987654321	10
987654321	20
987654321	30

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 FROM NHANVIEN  
INTERSECT  
SELECT TENTN, PHAI FROM THANNHAN
```

```
SELECT NV.*  
FROM NHANVIEN NV, THANNHAN TN  
WHERE NV.MANV=TN.MA_NVLEN  
AND NV.TENNV=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_NVIEN 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
- Câu truy vấn con thường trả về một tập các giá trị
- Các câu truy vấn con trong cùng một mệnh đề WHERE được kết hợp bằng phép nối logic
- 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 - Lòng phân cấp

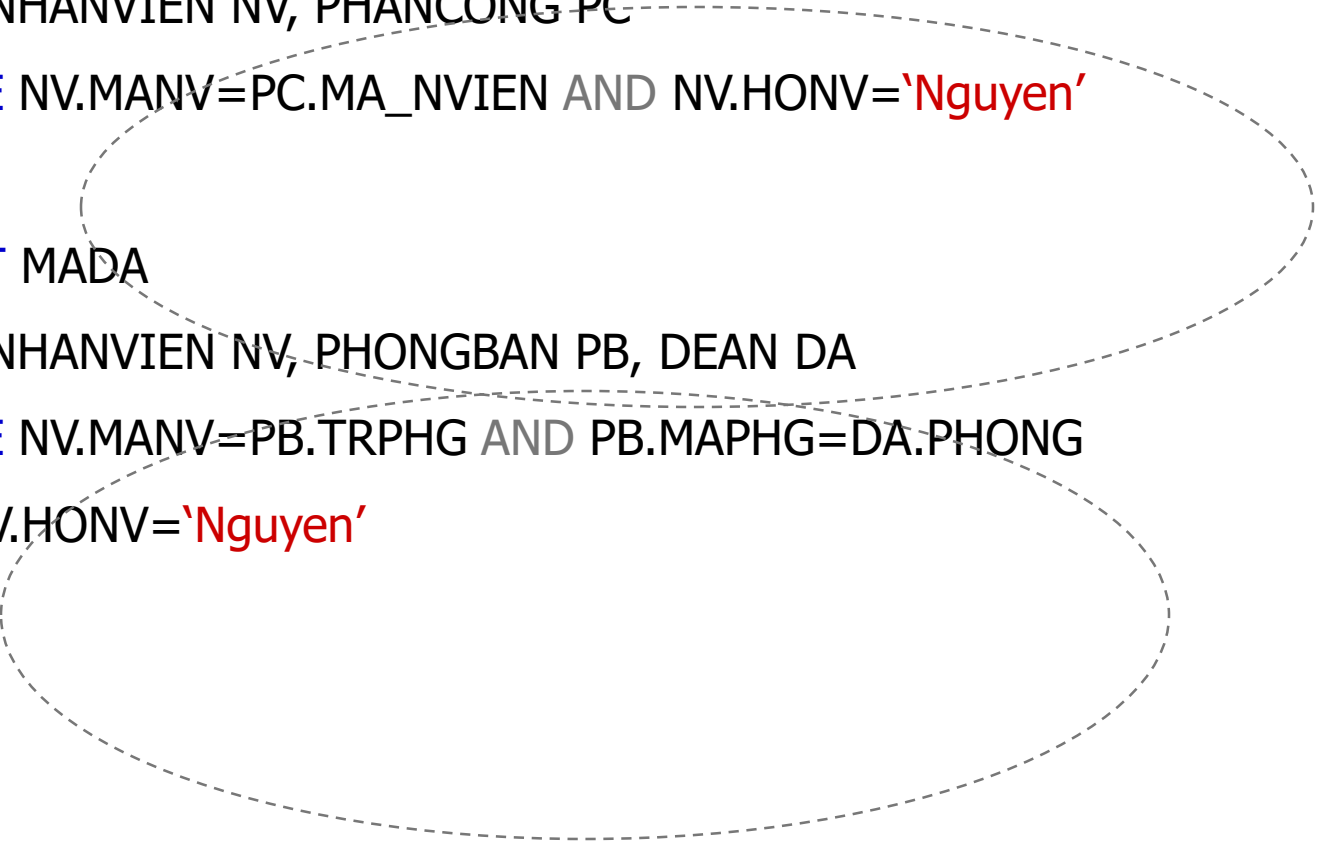
```
SELECT MANV, TENNV  
FROM NHANVIEN, PHONGBAN  
WHERE TENPHG='Nghien cuu' AND PHG=MAPHG
```

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

```
SELECT MANV, TENNV  
FROM NHANVIEN  
WHERE PHG IN ( 1, 4, 5)
```

Ví dụ 5

```
SELECT SODA
FROM NHANVIEN NV, PHANCONG PC
WHERE NV.MANV=PC.MA_NVIENT AND NV.HONV='Nguyễn'
UNION
SELECT MADA
FROM NHANVIEN NV, PHONGBAN PB, DEAN DA
WHERE NV.MANV=PB.TRPHG AND PB.MAPHG=DA.PHONG
AND NV.HONV='Nguyễn'
```



Ví dụ 7

- Tìm những nhân viên không có thân nhân nào

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

```
SELECT *  
FROM NHANVIEN  
WHERE MANV <> ALL (  
    SELECT MA_NVIENTH  
    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_NVIEN 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_NVIEN )  
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

Ví dụ 11-> chưa học count

- Tìm 3 nhân viên có lương cao nhất

```
SELECT TENNV  
FROM NHANVIEN NV1  
WHERE 2 >= (  
    SELECT COUNT(*)  
    FROM NHANVIEN NV2  
    WHERE NV2.LUONG>NV1.LUONG )
```

Hàm kết hợp

- COUNT

- COUNT(*) đếm số dòng
- COUNT(<tên thuộc tính>) đếm số giá trị khác NULL của thuộc tính
- COUNT(DISTINCT <tên thuộc tính>) đếm số giá trị khác nhau và khác NULL của thuộc tính

- MIN

- MAX

- SUM

- AVG

- Các hàm kết hợp được đặt ở mệnh đề SELECT

Ví dụ 13

- Tìm tổng lương, lương cao nhất, lương thấp nhất và lương trung bình của các nhân viên

```
SELECT SUM(LUONG), MAX(LUONG), MIN(LUONG), AVG(LUONG)  
FROM NHANVIEN
```

Ví dụ 14

- Cho biết số lượng nhân viên của phòng 'Nghien cuu'

```
SELECT COUNT(*) AS SL_NV  
FROM NHANVIEN, PHONGBAN  
WHERE PHG=MAPHG AND TENPHG='Nghien cuu'
```

Ví dụ 15

- Cho biết số lượng nhân viên của từng phòng ban

PHG	SL_NV
5	3
4	3
1	1

MANV	HONV	TENLOT	TENNV	NGSINH	DCHI	PHAI	LUONG	MA_NQL	PHG
333445555	Nguyen	Thanh	Tung	12/08/1955	638 NVC Q5	Nam	40000	888665555	5
987987987	Nguyen	Manh	Hung	09/15/1962	Ba Ria VT	Nam	38000	333445555	5
453453453	Tran	Thanh	Tam	07/31/1972	543 MTL Q1	Nu	25000	333445555	5
999887777	Bui	Ngoc	Hang	07/19/1968	33 NTH Q1	Nu	38000	987654321	4
987654321	Le	Quynh	Nhu	07/20/1951	219 TD Q3	Nu	43000	888665555	4
987987987	Tran	Hong	Quang	04/08/1969	980 LHP Q5	Nam	25000	987654321	4
888665555	Pham	Van	Vinh	11/10/1945	450 TV HN	Nam	55000	NULL	1

Gom nhóm

- Cú pháp

```
SELECT <danh sách các cột>  
FROM <danh sách các bảng>  
WHERE <điều kiện>  
GROUP BY <danh sách các cột gom nhóm>
```

- Sau khi gom nhóm

- Mỗi nhóm các bộ sẽ có cùng giá trị tại các thuộc tính gom nhóm

Ví dụ 15

- Cho biết số lượng nhân viên của từng phòng ban

```
SELECT PHG, COUNT(*) AS SL_NV  
FROM NHANVIEN  
GROUP BY PHG
```

```
SELECT TENPHG, COUNT(*) AS SL_NV  
FROM NHANVIEN, PHONGBAN  
WHERE PHG=MAPHG  
GROUP BY TENPHG
```

Ví dụ 16

- Với mỗi nhân viên cho biết mã số, họ tên, số lượng đề án và tổng thời gian mà họ tham gia

MA_NVIEN	SODA	THOIGIAN
123456789	1	32.5
123456789	2	7.5
333445555	2	10.0
333445555	3	10.0
333445555	10	10.0
888665555	20	20.0
987987987	10	35.0
987987987	30	5.0
987654321	30	20.0
987654321	20	15.0
453453453	1	20.0
453453453	2	20.0

SELECT MA_NVIEN, **COUNT**(*) **AS** SL_DA,
SUM(THOIGIAN) **AS** TONG_TG
FROM PHANCONG
GROUP BY MA_NVIEN

SELECT HONV, TENNV, **COUNT**(*) **AS** SL_DA,
SUM(THOIGIAN) **AS** TONG_TG
FROM PHANCONG, NHANVIEN
WHERE MA_NVIEN=MANV
GROUP BY MA_NVIEN, HONV, TENNV

Ví dụ 17

- Cho biết những nhân viên tham gia từ 2 đề án trở lên

MA_NVIEN	SODA	THOIGIAN
123456789	1	32.5
123456789	2	7.5
333445555	2	10.0
333445555	3	10.0
333445555	10	10.0
888665555	20	20.0
987987987	10	35.0
987987987	30	5.0
987654321	30	20.0
987654321	20	15.0
453453453	1	20.0
453453453	2	20.0

bị loại ra

Điều kiện trên nhóm

- Cú pháp

SELECT <danh sách các cột>

FROM <danh sách các bảng>

WHERE <điều kiện>

GROUP BY <danh sách các cột gom nhóm>

HAVING <điều kiện trên nhóm>

Ví dụ 17

- Cho biết những nhân viên tham gia từ 2 đề án trở lên

```
SELECT MA_NVIEN  
FROM PHANCONG  
GROUP BY MA_NVIEN  
HAVING COUNT(*) >= 2
```

Ví dụ 18

- Cho biết những phòng ban (TENPHG) có lương trung bình của các nhân viên lớn hơn 20000

```
SELECT PHG, AVG(LUONG) AS LUONG_TB  
FROM NHANVIEN  
GROUP BY PHG  
HAVING AVG(LUONG) > 20000
```

```
SELECT TENPHG, AVG(LUONG) AS LUONG_TB  
FROM NHANVIEN, PHONGBAN  
WHERE PHG=MAPHG  
GROUP BY TENPHG  
HAVING AVG(LUONG) > 20000
```

Nhận xét

- Mệnh đề GROUP BY

- Các thuộc tính trong mệnh đề SELECT (trừ những thuộc tính trong các hàm kết hợp) phải xuất hiện trong mệnh đề GROUP BY

- Mệnh đề HAVING

- Sử dụng các hàm kết hợp trong mệnh đề SELECT để kiểm tra một số điều kiện nào đó
- Chỉ kiểm tra điều kiện trên nhóm, không là điều kiện lọc trên từng bộ
- Sau khi gom nhóm điều kiện trên nhóm mới được thực hiện

Nhận xét (tt)

- Thứ tự thực hiện câu truy vấn có mệnh đề GROUP BY và HAVING
 - (1) Chọn ra những dòng thỏa điều kiện trong mệnh đề WHERE
 - (2) Những dòng này sẽ được gom thành nhiều nhóm tương ứng với mệnh đề GROUP BY
 - (3) Áp dụng các hàm kết hợp cho mỗi nhóm
 - (4) Bỏ qua những nhóm không thỏa điều kiện trong mệnh đề HAVING
 - (5) Rút trích các giá trị của các cột và hàm kết hợp trong mệnh đề SELECT

Ví dụ 19

- Tìm những phòng ban có lương trung bình cao nhất

```
SELECT PHG, AVG(LUONG) AS LUONG_TB
FROM NHANVIEN
GROUP BY PHG
HAVING MAX(LUONG) ALL (
    SELECT AVG(LUONG)
    FROM NHANVIEN
    GROUP BY PHG)
```

Ví dụ 12

- Tìm tên các nhân viên được phân công làm tất cả các đồ án

```
SELECT MANV, TENVN
FROM NHANVIEN, PHANCONG
WHERE MANV=MA_NVIENT
GROUP BY MANV, TENVN
HAVING COUNT(*) = (
    SELECT COUNT(*)
    FROM DEAN )
```

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)

Một số dạng truy vấn khác

- Truy vấn con ở mệnh đề FROM
- Điều kiện kết ở mệnh đề FROM
 - Phép kết tự nhiên
 - Phép kết ngoài
- Cấu trúc CASE

Truy vấn con ở mệnh đề FROM

- Kết quả trả về của một câu truy vấn phụ là một bảng
 - Bảng trung gian trong quá trình truy vấn
 - Không có lưu trữ thật sự
- Cú pháp

SELECT <danh sách các cột>

FROM R1, R2, (<truy vấn con>) **AS** tên_bảng

WHERE <điều kiện>

Ví dụ 18

- Cho biết những phòng ban (TENPHG) có lương trung bình của các nhân viên lớn hơn 20000

```

SELECT PHG, AVG(LUONG) AS LUONG_TB
FROM NHANVIEN
GROUP BY PHG
HAVING AVG(LUONG) > 20000

SELECT PHG, TENPHG, AVG(LUONG) AS LUONG_TB
FROM NHANVIEN, PHONGBAN
WHERE MAPHG = TEMP.PHG
GROUP BY PHG, TENPHG
HAVING AVG(LUONG) > 20000

```

Điều kiện kết ở mệnh đề FROM

- Kết bằng

```
SELECT <danh sách các cột>  
FROM R1 [INNER] JOIN R2 ON <biểu thức>  
WHERE <điều kiện>
```

- Kết ngoài

```
SELECT <danh sách các cột>  
FROM R1 LEFT | RIGHT [OUTER] JOIN R2 ON <biểu thức>  
WHERE <điều kiện>
```

Ví dụ 20

- Tìm mã và tên các nhân viên làm việc tại phòng 'Nghien cuu'

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

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

Ví dụ 21

- Tìm họ tên các nhân viên và tên các đề án nhân viên tham gia nếu có

```
SELECT NV.TENNV, NV.TENDA
FROM (PHANCONG PC JOIN DEAN DA ON SODA=MADA)
     LEFT JOIN NHANVIEN NV ON PC.MA_NVIENT=NV.MANV
```

PHANCONG join DEAN

NHANVIEN

MA_NVIENT=MANV

←
mở rộng

Cấu trúc CASE

- Cho phép kiểm tra điều kiện và xuất thông tin theo từng trường hợp
- Cú pháp

```
CASE <tên cột>  
    WHEN <giá trị> THEN <biểu thức>  
    WHEN <giá trị> THEN <biểu thức>  
    ...  
    [ELSE <biểu thức>]  
END
```

Ví dụ 22

- Cho biết họ tên các nhân viên đã đến tuổi về hưu (nam 60 tuổi, nữ 55 tuổi)

```
SELECT HONV, TENNV
FROM NHANVIEN
WHERE YEAR(GETDATE()) - YEAR(NGSINH) >= ( CASE PHAI
                                           WHEN 'Nam' THEN 60
                                           WHEN 'Nu' THEN 55
                                           END )
```

Ví dụ 23

- Cho biết họ tên các nhân viên và năm về hưu

```
SELECT HONV, TENNV  
(CASE PHAI  
    WHEN 'Nam' THEN YEAR(NGSINH) + 60  
    WHEN 'Nu' THEN YEAR(NGSINH) + 55  
END ) AS NAMVEHUU  
FROM NHANVIEN
```


Kết luận

SELECT <danh sách các cột>

FROM <danh sách các bảng>

[**WHERE** <điều kiện>]

[**GROUP BY** <các thuộc tính gom nhóm>]

[**HAVING** <điều kiện trên nhóm>]

[**ORDER BY** <các thuộc tính sắp thứ tự>]

