

TRƯỜNG ĐẠI HỌC THỦ DẦU MỘT
KHOA KỸ THUẬT CÔNG NGHỆ

CƠ SỞ DỮ LIỆU

Chương 5 NGÔN NGỮ TRUY VẤN CÓ CẤU TRÚC (SQL)

Phone: 0274. 3834930

Website: www.et.tdmu.edu.vn

NỘI DUNG

- Giới thiệu
- Định nghĩa dữ liệu
- Truy vấn dữ liệu
- Cập nhật dữ liệu
- Khung nhìn (view)
- Chỉ mục (index)



1. Giới thiệu

- Ngôn ngữ ƉSQH
 - Cách thức truy vấn dữ liệu
 - Khó khăn cho người sử dụng
- SQL (Structured Query Language)
 - Ngôn ngữ cấp cao
 - Người sử dụng chỉ cần đưa ra nội dung cần truy vấn
 - Được phát triển bởi IBM (1970s)
 - Được gọi là SEQUEL
 - Được ANSI công nhận và phát triển thành chuẩn
 - SQL-86
 - SQL-92
 - SQL-99



3

1. Giới thiệu (tt)

- SQL gồm
 - Định nghĩa dữ liệu (DDL)
 - Thao tác dữ liệu (DML)
 - Định nghĩa khung nhìn
 - Ràng buộc toàn vẹn
 - Phân quyền và bảo mật
 - Điều khiển giao tác
- SQL sử dụng thuật ngữ
 - Bảng ~ quan hệ
 - Cột ~ thuộc tính
 - Dòng ~ bộ

Lý thuyết : Chuẩn SQL-92

Ví dụ : SQL Server



4

2. Định nghĩa dữ liệu

- Kiểu dữ liệu
- Các lệnh định nghĩa dữ liệu



5

2. Định nghĩa dữ liệu

- Là ngôn ngữ mô tả
 - Lược đồ cho mỗi quan hệ
 - Miền giá trị tương ứng của từng thuộc tính
 - Ràng buộc toàn vẹn
 - Chỉ mục trên mỗi quan hệ
- Gồm
 - CREATE TABLE (tạo bảng)
 - DROP TABLE (xóa bảng)
 - ALTER TABLE (sửa bảng)
 - CREATE DOMAIN (tạo miền giá trị)
 - CREATE DATABASE
 - ...



6

2.1.Kiểu dữ liệu

- Số (numeric)
 - INTEGER
 - SMALLINT
 - NUMERIC, NUMERIC(p), NUMERIC(p,s)
 - DECIMAL, DECIMAL(p), DECIMAL(p,s)
 - REAL
 - DOUBLE PRECISION
 - FLOAT, FLOAT(p)



7

2.1.Kiểu dữ liệu (tt)

- Chuỗi ký tự (character string)
 - CHARACTER, CHARACTER(n)
 - CHARACTER VARYING(x)
- Chuỗi bit (bit string)
 - BIT, BIT(x)
 - BIT VARYING(x)
- Ngày giờ (datetime)
 - DATE gồm ngày, tháng và năm
 - TIME gồm giờ, phút và giây
 - TIMESTAMP gồm ngày và giờ



8

2.2.Lệnh tạo bảng

- Để định nghĩa một bảng
 - Tên bảng
 - Các thuộc tính
 - Tên thuộc tính
 - Kiểu dữ liệu
 - Các RBTV trên thuộc tính
- Cú pháp

```
CREATE TABLE <Tên_bảng>
(
    <Tên_cột> <Kiểu_dữ_liệu> [<RBTV>],
    <Tên_cột> <Kiểu_dữ_liệu> [<RBTV>],
    ...
    [<RBTV>]
)
```



9

Ví dụ - Tạo bảng

```
CREATE TABLE NHANVIEN
(
    MANV CHAR(9),
    HONV VARCHAR(10),
    TENLOT VARCHAR(20),
    TENNV VARCHAR(10),
    NGSINH DATETIME,
    DCHI VARCHAR(50),
    PHAI CHAR(3),
    LUONG INT,
    MA_NQL CHAR(9),
    PHG INT
)
```



10

2.2.Lệnh tạo bảng (tt)

- <RBTV>
 - NOT NULL
 - NULL
 - UNIQUE
 - DEFAULT
 - PRIMARY KEY
 - FOREIGN KEY / REFERENCES
 - CHECK
- Đặt tên cho RBTV

CONSTRAINT <Ten_RBTV> <RBTV>



11

Ví dụ - RBTV

```
CREATE TABLE NHANVIEN
(
  HONV VARCHAR(10) NOT NULL,
  TENLOT VARCHAR(20) NOT NULL,
  TENNV VARCHAR(10) NOT NULL,
  MANV CHAR(9) PRIMARY KEY,
  NGSINH DATETIME,
  DCHI VARCHAR(50),
  PHAI CHAR(3) CHECK (PHAI IN ('Nam', 'Nu')),
  LUONG INT DEFAULT (10000),
  MA_NQL CHAR(9),
  PHG INT
)
```



12

Ví dụ - RBTV

```
CREATE TABLE PHONGBAN
(
    TENPB VARCHAR(20) UNIQUE,
    MAPHG INT NOT NULL,
    TRPHG CHAR(9),
    NG_NHANCHUC DATETIME DEFAULT (GETDATE())
)
```

```
CREATE TABLE PHANCONG
(
    MA_NVIEN CHAR(9) FOREIGN KEY (MA_NVIEN)
        REFERENCES NHANVIEN(MANV),
    SODA INT REFERENCES DEAN(MADA),
    THOIGIAN DECIMAL(3,1)
)
```



13

Ví dụ - Đặt tên cho RBTV

```
CREATE TABLE NHANVIEN
(
    HONV VARCHAR(10) CONSTRAINT NV_HONV_NN NOT NULL,
    TENLOT VARCHAR(20) NOT NULL,
    TENNV VARCHAR(10) NOT NULL,
    MANV CHAR(9) CONSTRAINT NV_MANV_PK PRIMARY KEY,
    NGSINH DATETIME,
    DCHI VARCHAR(50),
    PHAI CHAR(3) CONSTRAINT NV_PHAICHK
        CHECK (PHAI IN ('Nam', 'Nu')),
    LUONG INT CONSTRAINT NV_LUONG_DF DEFAULT (10000),
    MA_NQL CHAR(9),
    PHG INT
)
```



14

Ví dụ - Đặt tên cho RBTV

```
CREATE TABLE PHANCONG
(
    MA_NVIEN CHAR(9),
    SODA INT,
    THOIGIAN DECIMAL(3,1),
    CONSTRAINT PC_MANVIEN_SODA_PK PRIMARY KEY
    (MA_NVIEN, SODA),
    CONSTRAINT PC_MANVIEN_FK FOREIGN KEY (MA_NVIEN)
    REFERENCES NHANVIEN(MANV),
    CONSTRAINT PC_SODA_FK FOREIGN KEY (SODA)
    REFERENCES DEAN(MADA)
)
```



15

2.3.Lệnh sửa bảng

- Được dùng để
 - Thay đổi cấu trúc bảng
 - Thay đổi RBTV

- Thêm cột

```
ALTER TABLE <Tên_bảng> ADD COLUMN
<Tên_cột> <Kiểu_dữ_liệu> [<RBTV>]
```

- Xóa cột

```
ALTER TABLE <Tên_bảng> DROP COLUMN <Tên_cột>
```

- Mở rộng cột

```
ALTER TABLE <Tên_bảng> ALTER COLUMN
<Tên_cột> <Kiểu_dữ_liệu_mới>
```



16

2.3.Lệnh sửa bảng (tt)

- Thêm RBTV

```
ALTER TABLE <Tên_bảng> ADD
    CONSTRAINT <Tên_RBTV> <RBTV>,
    CONSTRAINT <Tên_RBTV> <RBTV>,
    ...
```

- Xóa RBTV

```
ALTER TABLE <Tên_bảng> DROP <Tên_RBTV>
```



17

Ví dụ - Thay đổi cấu trúc bảng

```
ALTER TABLE NHANVIEN ADD NGHENGHIEP CHAR(20)
```

```
ALTER TABLE NHANVIEN DROP COLUMN NGHENGHIEP
```

```
ALTER TABLE NHANVIEN ALTER COLUMN
    NGHENGHIEP CHAR(50)
```



18

Ví dụ - Thay đổi RBTV

```
CREATE TABLE PHONGBAN
(
    TENPB VARCHAR(20),
    MAPHG INT NOT NULL,
    TRPHG CHAR(9),
    NG_NHANCHUC DATETIME
)
```

```
ALTER TABLE PHONGBAN ADD
    CONSTRAINT PB_MAPHG_PK PRIMARY KEY (MAPHG),
    CONSTRAINT PB_TRPHG FOREIGN KEY (TRPHG)
        REFERENCES NHANVIEN(MANV),
    CONSTRAINT PB_NGNHANCHUC_DF DEFAULT (GETDATE())
        FOR (NG_NHANCHUC),
    CONSTRAINT PB_TENPB_UNI UNIQUE (TENPB)
```



19

2.4.Lệnh xóa bảng

- Được dùng để xóa cấu trúc bảng
 - Tất cả dữ liệu của bảng cũng bị xóa
- Cú pháp

```
DROP TABLE <Tên_bảng>
```

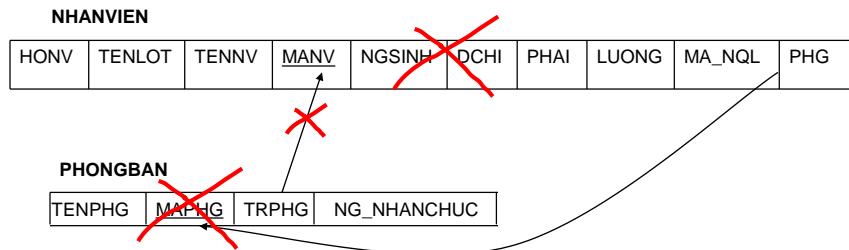
- Ví dụ

```
DROP TABLE NHANVIEN
DROP TABLE PHONGBAN
DROP TABLE PHANCONG
```



20

2.4.Lệnh xóa bảng (tt)



21

2.5.Lệnh tạo miền giá trị

- Tạo ra một kiểu dữ liệu mới kế thừa những kiểu dữ liệu có sẵn
- Cú pháp

```
CREATE DOMAIN <Tên_kdl_mới> AS <Kiểu_dữ_liệu>
```

- Ví dụ

```
CREATE DOMAIN Kieu_Ten AS VARCHAR(30)
```



22

3. Truy vấn dữ liệu

- Truy vấn cơ bản
- Tập hợp, so sánh tập hợp
- Truy vấn lồng
- Hàm kết hợp và gom nhóm
- Một số kiểu truy vấn khác



23

3. Truy vấn dữ liệu

- Là ngôn ngữ rút trích dữ liệu thỏa một số điều kiện nào đó
- Dựa trên

Phép toán ĐSQH

+

Một số bổ sung

- Cho phép 1 bảng có nhiều dòng trùng nhau
- Bảng là *bag* \neq quan hệ là *set*



24

3.1.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



25

3.1.Truy vấn cơ bản (tt)

- SQL và ĐSQH

π
~~X~~ SELECT <danh sách các cột>
~~FROM~~ <danh sách các bảng>
 σ WHERE <điều kiện>

```
SELECT L
FROM R
WHERE C
```

→



26

Ví dụ

❖ Tìm các nhân viên ở phòng 5

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

Lấy tất cả các cột của
quan hệ kết quả

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



27

3.1.1. Mệnh đề SELECT

❖ Tìm các nhân viên Nam ở phòng 5

```
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



28

3.1.1.Mệnh đề SELECT (tt)

Tên bí danh

❖ Tìm các nhân viên Nam ở phòng 5

```
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



29

3.1.1.Mệnh đề SELECT (tt)

Mở rộng

❖ Tìm các nhân viên Nam ở phòng 5

```
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



30

3.1.1.Mệnh đề SELECT (tt)

Mở rộng

- ❖ Cho biết MaNV, Luong sau khi tăng 10% của các nhân viên Nam ở phòng 5

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

MANV	LUONG10%
333445555	33000
987987987	27500



31

3.1.1.Mệnh đề SELECT (tt)

Loại bỏ các dòng trùng nhau

- ❖ Cho biết các mức lương của nhân viên Nam ở phòng 5

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

LUONG
30000
25000
28000
38000

- Tổn chi phí
- Người dùng muốn thấy



32

3.1.2.Mệnh đề WHERE

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

```

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

Biểu thức logic

↓ TRUE ↓ TRUE



33

3.1.2.Mệnh đề WHERE (tt)

Độ ưu tiên

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

```

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



34

3.1.2.Mệnh đề WHERE (tt)

BETWEEN

- Cho biết MANV và TENNV có Lương từ 20000 đến 30000

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

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



35

3.1.2.Mệnh đề WHERE (tt)

NOT BETWEEN

- Cho biết MANV và TENNV có Lương dưới 20000 hoặc trên 30000

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



36

3.1.2.Mệnh đề WHERE (tt)

LIKE

- Cho biết MANV và TENNV của những nhân viên có địa chỉ bắt đầu bằng chữ 'Nguyen'

```
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



37

3.1.2.Mệnh đề WHERE (tt)

NOT LIKE

- Cho biết MANV và TENNV của những nhân viên có họ 'Nguyen'

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

- Cho biết MANV và TENNV của những nhân viên không phải họ 'Nguyen'

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



38

3.1.2.Mệnh đề WHERE (tt)

ESCAPE

- Cho biết MANV và TENNV của những nhân viên có họ 'Nguyễn_'

```
SELECT MANV, TENNV
FROM NHANVIEN
WHERE DCHI LIKE '% Nguyens_%' ESCAPE 's'
```



'Nguyễn_'



39

3.1.2.Mệnh đề WHERE (tt)

Ngày giờ

- ❖ Cho biết các nhân viên sinh trong khoảng năm 1955 đến 1960

```
SELECT MANV, TENNV
FROM NHANVIEN
WHERE NGSINH BETWEEN '01-01-1955' AND '31-12-1960'
```

HOẶC

```
SELECT MANV, TENNV
FROM NHANVIEN
WHERE YEAR(NGSINH) BETWEEN 1955 AND 1960
```



40

3.1.2.Mệnh đề WHERE (tt)

NULL/NOT 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



41

3.1.2.Mệnh đề WHERE (tt)

NULL/NOT NULL

- Cho biết MANV và TENNV của những nhân viên không có người quản lý trực tiếp

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

- Cho biết MANV và TENNV của những nhân viên có người quản lý trực tiếp

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



42

3.1.3.Mệnh đề FROM

Không sử dụng mệnh đề WHERE

- Cho biết thông tin của tất cả nhân viên và thông tin phòng làm việc của họ.

```
SELECT *
FROM NHANVIEN, PHONGBAN
WHERE TRUE
```

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



43

3.1.3.Mệnh đề FROM (tt)

Tên bí danh

- Cho biết Tên phòng và Địa điểm của phòng ban đó.

```
SELECT TENPHG, DIADIEM
FROM PHONGBAN, DIEMPHG PHG AS DD
WHERE MAPHG=MAPHG
WHERE PB.MAPHG=DD.MAPHG
```

- Cho biết Tên và ngày sinh của tất cả nhân viên và thân nhân của họ.

```
SELECT TENNV, NGAYSINH TEN, NGAYSINH NGAYSINH
FROM NHANVIEN, THANHAN THAN TN
WHERE MANV=MA_NVN
```



44

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.

```
SELECT SODA,PHONG,HONV, TENNV, NGSINH, DCHI
FROM DEAN,PHONGBAN,NHANVIEN
WHERE PHONG=MAPHG AND TRPHG=MANV AND
DDIEM_DEAN='HA NOI'
```



45

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.

```
SELECT HONV, TENNV
FROM DEAN, PHANCONG, NHANVIEN
WHERE SODA=MADA AND MA_NVIEN=MANV
AND PHG=5 AND TENDA='San pham X'
AND THOIGIAN>10
```



46

Ví dụ 3

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

```
SELECT NV.HONV, NV.TENNV, NQL.HONV, NQL.TENNV
FROM NHANVIEN NV, NHANVIEN NQL
WHERE NV.MA_NQL=NQL.MANV
```



47

Ví dụ 4

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

```
SELECT NV2.HOVN, NV2.TENNV
FROM NHANVIEN NV1, NHANVIEN NV2
WHERE NV2.MA_NQL=NV1.MANV
      AND NV1.HONV='NGUYEN'
      AND NV1.HOLOT='THANH'
      AND NV1.TENNV='TUNG'
```



48

3.1.4. 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



49

3.1.4. Mệnh đề ORDER BY (tt)

- Ví dụ: Cho biết Mã nhân viên và mã đề án mà họ được phân công. Sắp xếp giảm dần theo Mã nhân viên, tăng dần theo mã đề án.

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

MA_NV	SODA
999887777	10
999887777	30
987987987	10
987987987	30
987654321	10
987654321	20
987654321	30



50

3.2. 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



51

3.2. Phép toán tập hợp trong SQL (tt)

- 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>
```



52

Ví dụ 5

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

```
(SELECT SODA
FROM PHANCONG, NHANVIEN
WHERE MANV=MA_NVIE AND HONV='NGUYEN')
UNION
(SELECT SODA
FROM DEAN, PHONGBAN, NHANVIEN
WHERE PHONG=MAPHG AND TRPHG=MANV AND
HONV='NGUYEN'
)
```



53

Ví dụ 6

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

```
(SELECT MANV,TENNV,PHAI
FROM NHANVIEN)
INTERSECT
(SELECT MA_NVIE, TENTN, PHAI
FROM THANNHAN)
```



54

Ví dụ 7

- Tìm những nhân viên không có thân nhân nào.

(**SELECT** MANV FROM NHANVIEN)

EXCEPT

(**SELECT** MA_NVIAN FROM THANNNHAN)



55

3.3.Truy vấn lồng

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)



56

3.3.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



57

3.3.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



58

Ví dụ - Lồng phân cấp

- ❖ Tìm những nhân viên làm việc trong các phòng ban có địa điểm ở TP HCM.

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

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



59

Ví dụ 8

- Tìm những nhân viên không có thân nhân nào.

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

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



60

Ví dụ 9

- 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
```



61

Ví dụ 10

- 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)
```



62

Ví dụ 11

- Tìm những trưởng phòng có tối thiểu một thân nhân.

```
SELECT TRPHG
FROM PHONGBAN
WHERE TRPHG IN (SELECT MA_NVIEN
                FROM THANNHAN)
```



63

Ví dụ - Lồng tương quan

- ❖ Tìm những nhân viên làm việc ở phòng Nghiên Cứu.

```
SELECT MANV, TENNV
FROM NHANVIEN, PHONGBAN
WHERE TENPHG='Nghiên cứu' AND PHG=MAPHG
```

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



64

Ví dụ 12

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

```
SELECT *
FROM NHANVIEN A
WHERE EXISTS ( SELECT *
                FROM THANNHAN B
                WHERE A.MANV=B.MA_NVLEN
                  AND A.TENNV=B.TENTN
                  AND A.PHAI=B.PHAI
              )
```



65

Ví dụ 13

- Tìm những nhân viên không có thân nhân nào.

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



66

Ví dụ 15

- Tìm những trưởng phòng có tối thiểu một thân nhân.

```
SELECT TRPHG
FROM PHONGBAN
WHERE EXISTS (SELECT *
               FROM THANNHAN
               WHERE TRPHG=MA_NVIAN
               )
```



67

Ví dụ 14

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

```
SELECT *
FROM NHANVIEN A
WHERE EXISTS (SELECT *
              FROM NHANVIEN B
              WHERE B.PHG=4
                  AND A.LUONG>B.LUONG
              )
```



68

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



69

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÷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



70

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 ))
```



71

Ví dụ 16

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



72

Ví dụ 16 (tt)

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

```
SELECT * FROM NHANVIEN A
WHERE NOT EXISTS
    (SELECT *
      FROM DEAN B
      WHERE NOT EXISTS
          (SELECT *
            FROM PHANCONG C
            WHERE C.MA_NVN=A.MANV
              AND C.SODA=B.MADA
          )
    )
```



73

3.4.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



74

Ví dụ 17

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

```
SELECT SUM(LUONG) AS TONG_LUONG,
       MAX(LUONG) AS LUONG_LON_NHAT,
       MIN(LUONG) AS LUONG_NHO_NHAT,
       AVG(LUONG) AS LUONG_TRUNG_BINH
FROM NHANVIEN
```



75

Ví dụ 18

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

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



76

Ví dụ 19

- 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	07620/1951	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



77

3.5.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



78

Ví dụ 19 (tt)

- 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
```



79

Ví dụ 20

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

MA_NVIENT	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



80

Ví dụ 20 (tt)

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

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

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



81

Ví dụ 21

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

MA_NVIENT	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



82

3.5.1.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>
```



83

Ví dụ 21 (tt)

- 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
```



84

Ví dụ 22

- 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
```



85

3.5.2. 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



86

3.5.2. 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



87

Ví dụ 23

- 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) > (SELECT AVG(LUONG)
                        FROM NHANVIEN
                        GROUP BY PHG))
```



88

Ví dụ 24

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

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



89

3.6.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



90

3.6.1. 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>
```



91

Ví dụ 25

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

Cách 1:

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

Cách 2:

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



92

3.6.2.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>
```



93

Ví dụ 26

- 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'
```



94

Ví dụ 27

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

PHANCONG join DEAN join NHANVIEN
 MA_NVIENT=MANV

```

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



95

Ví dụ 28

- Cho biết họ tên nhân viên và tên phòng ban mà họ là trưởng phòng nếu có

TENNV	HONV	TENPHG
Tung	Nguyen	Nghien cuu
Hang	Bui	null
Nhu	Le	null
Vinh	Pham	Quan ly

```

SELECT TENNV, HONV, TENPHG
FROM NHANVIEN, PHONGBAN
WHERE MANV=TRPHG
  
```



96

Ví dụ 28 (tt)

TENNV	HONV	TENPHG
Tung	Nguyen	Nghien cuu
Hang	Bui	null
Nhu	Le	null
Vinh	Pham	Quan ly

Mở rộng dữ liệu cho
bảng NHANVIEN

Cách 1:

RHONGBAN	join	NHANVIEN	
----------	------	----------	--

MANV=TRPHG

SELECT TENNV, HONV, TENPHG
FROM RHONGBAN **RIGHT JOIN** NHANVIEN ON
MANV=TRPHG



97

3.6.3.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



98

Ví dụ 29

- 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 )
```



99

Ví dụ 30

- 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
```



100

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>]
```



101

4.Cập nhật dữ liệu

- Thêm (insert)
- Xóa (delete)
- Sửa (update)



102

4.1.Lệnh INSERT

- Dùng để thêm 1 hay nhiều dòng vào bảng
- Để thêm dữ liệu
 - Tên quan hệ
 - Danh sách các thuộc tính cần thêm dữ liệu
 - Danh sách các giá trị tương ứng



103

4.1.Lệnh INSERT (tt)

- Cú pháp (thêm 1 dòng)

```
INSERT INTO <tên bảng>(<danh sách các thuộc tính>)  
VALUES (<danh sách các giá trị>)
```



104

Ví dụ 31

```
INSERT INTO NHANVIEN(HONV, TENLOT, TENNV, MANV)
VALUES ('Le', 'Van', 'Tuyen', '635635635')
```

```
INSERT INTO NHANVIEN(HONV, TENLOT, TENNV, MANV, DCHI)
VALUES ('Le', 'Van', 'Tuyen', '635635635', NULL)
```

```
INSERT INTO NHANVIEN
VALUES ('Le', 'Van', 'Tuyen', '635635635', '12/30/1952',
'98 HV', 'Nam', '37000', 4)
```



105

4.1.Lệnh INSERT (tt)

- Nhận xét
 - Thứ tự các giá trị phải trùng với thứ tự các cột
 - Có thể thêm giá trị NULL ở những thuộc tính không là khóa chính và NOT NULL
 - Câu lệnh INSERT sẽ gặp lỗi nếu vi phạm RBTV
 - Khóa chính
 - Tham chiếu
 - NOT NULL - các thuộc tính có ràng buộc NOT NULL bắt buộc phải có giá trị



106

4.1.Lệnh INSERT (tt)

- Cú pháp (thêm nhiều dòng)

```
INSERT INTO <tên bảng>(<danh sách các thuộc tính>)  
          <câu truy vấn con>
```



107

Ví dụ 32

```
CREATE TABLE THONGKE_PB  
(  
    TENPHG VARCHAR(20),  
    SL_NV INT,  
    LUONG_TC INT  
)
```

```
INSERT INTO THONGKE_PB(TENPHG, SL_NV, LUONG_TC)  
SELECT TENPHG, COUNT(MANV), SUM(LUONG)  
FROM NHANVIEN, PHONGBAN  
WHERE PHG=MAPHG  
GROUP BY TENPHG
```



108

4.2.Lệnh DELETE

- Dùng để xóa các dòng của bảng
- Cú pháp

```
DELETE FROM <tên bảng>  
[WHERE <điều kiện>]
```



109

Ví dụ 33

```
DELETE FROM NHANVIEN  
WHERE HONV='Tran'
```

```
DELETE FROM NHANVIEN  
WHERE MANV='345345345'
```

```
DELETE FROM NHANVIEN
```



110

Ví dụ 34

- Xóa đi những nhân viên ở phòng 'Nghien cuu'

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



111

4.2.Lệnh DELETE (tt)

- Nhận xét
 - Số lượng số dòng bị xóa phụ thuộc vào điều kiện ở mệnh đề WHERE
 - Nếu không chỉ định điều kiện ở mệnh đề WHERE, tất cả các dòng trong bảng sẽ bị xóa
 - Lệnh DELETE có thể gây ra vi phạm RB tham chiếu
 - Không cho xóa
 - Xóa luôn những dòng có giá trị đang tham chiếu đến
 - CASCADE
 - Đặt NULL cho những giá trị tham chiếu



112

4.2.Lệnh DELETE (tt)

MANV	HONV	TENLO	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	07620/1951	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

MA_NVIENT	SODA	THOIGIAN
333445555	10	10.0
888665555	20	20.0
987987987	10	35.0
987987987	30	5.0
987654321	30	20.0
453453453	1	20.0



113

4.2.Lệnh DELETE (tt)

TENPHG	MAPHG	MA_NVIENT	NG_NHANCHU
Nghien cuu	5	333445555	05/22/1988
Dieu hanh	4	987987987	01/01/1995
Quan ly	1	888665555	06/19/1981

MANV	HONV	TENLO	TENNV	NGSINH	DCHI	PHAI	LUONG	MA_NQL	PHG
333445555	Nguyen	Thanh	Tung	12/08/1955	638 NVC Q5	Nam	40000	888665555	NULL
987987987	Nguyen	Manh	Hung	09/15/1962	Ba Ria VT	Nam	38000	333445555	NULL
453453453	Tran	Thanh	Tam	07/31/1972	543 MTL Q1	Nu	25000	333445555	NULL
999887777	Bui	Ngoc	Hang	07/19/1968	33 NTH Q1	Nu	38000	987654321	4
987654321	Le	Quynh	Nhu	07620/1951	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



114

4.3.Lệnh UPDATE

- Dùng để thay đổi giá trị của thuộc tính cho các dòng của bảng
- Cú pháp

```
UPDATE <tên bảng>
SET <tên thuộc tính>=<giá trị mới>,
    <tên thuộc tính>=<giá trị mới>,
    ...
[WHERE <điều kiện>]
```



115

Ví dụ 35

- Cập nhật ngày sinh của nhân viên có mã số 333445555 thành 12/08/1965.

```
UPDATE NHANVIEN
SET NGSINH='08/12/1965'
WHERE MANV='333445555'
```

- Cập nhật lương của nhân viên tăng 10%.

```
UPDATE NHANVIEN
SET LUONG=LUONG*1.1
```



116

Ví dụ 36

- Với đề án có mã số 10, hãy thay đổi nơi thực hiện đề án thành 'Vung Tau' và phòng ban phụ trách là phòng 5.

```
UPDATE DEAN
SET DIADIEM_DA='Vung Tau', PHONG=5
WHERE MADA=10
```



117

4.3 Lệnh UPDATE

- Nhận xét
 - Những dòng thỏa điều kiện tại mệnh đề WHERE sẽ được cập nhật giá trị mới
 - Nếu không chỉ định điều kiện ở mệnh đề WHERE, tất cả các dòng trong bảng sẽ bị cập nhật
 - Lệnh UPDATE có thể gây ra vi phạm RB tham chiếu
 - Không cho sửa
 - Sửa luôn những dòng có giá trị đang tham chiếu đến
 - CASCADE



118

5.Khung nhìn (view)

- Định nghĩa
- Truy vấn
- Cập nhật



119

5.Khung nhìn

- Bảng là một quan hệ được tổ chức lưu trữ vật lý trong CSDL
- Khung nhìn cũng là một quan hệ
 - Không được lưu trữ vật lý (bảng ảo)
 - Không chứa dữ liệu
 - Được định nghĩa từ những bảng khác
 - Có thể truy vấn hay cập nhật thông qua khung nhìn



120

5.Khung nhìn (tt)

- Tại sao phải sử dụng khung nhìn?
 - Che giấu tính phức tạp của dữ liệu
 - Đơn giản hóa các câu truy vấn
 - Hiện thị dữ liệu dưới dạng tiện dụng nhất
 - An toàn dữ liệu



121

5.1.Định nghĩa khung nhìn

- Cú pháp

CREATE VIEW <tên khung nhìn> **AS** <câu truy vấn>

DROP VIEW <tên khung nhìn>

- Bảng ảo này có
 - Danh sách thuộc tính trùng với các thuộc tính trong mệnh đề SELECT
 - Số dòng phụ thuộc vào điều kiện ở mệnh đề WHERE
 - Dữ liệu được lấy từ các bảng ở mệnh đề FROM



122

Ví dụ 37

```
CREATE VIEW NV_P5 AS
  SELECT MANV, HONV, TENLOT, TENVN
  FROM NHANVIEN
  WHERE PHG=5
```

```
CREATE VIEW TONGLNG_SLVN_PB AS
  SELECT MAPHG, TENPB, COUNT(*) AS SLNV,
  SUM(LUONG) AS TONGLNG
  FROM NHANVIEN, PHONGBAN
  WHERE PHG=MAPHG
  GROUP BY TENPHG
```



123

5.2. Truy vấn trên khung nhìn

- Tuy không chứa dữ liệu nhưng có thể thực hiện các câu truy vấn trên khung nhìn

```
SELECT TENNV
FROM NV_P5
WHERE HONV LIKE 'Nguyễn'
```

$$NV_P5 \leftarrow \pi_{MANV, HONV, TENLOT, TENNV} (\sigma_{PHG=5} (NHANVIEN))$$

$$\pi_{TENN} (\sigma_{HONV='Nguyễn'} (NV_P5))$$


124

5.2.Truy vấn trên khung nhìn (tt)

- Có thể viết câu truy vấn dữ liệu từ khung nhìn và bảng

```
SELECT HONV, TENVN, TENDA, THOIGIAN
FROM NV_P5, PHANCONG, DEAN
WHERE MANV=MA_NVLEN AND SODA=MADA
```

$$NV_P5 \leftarrow \pi_{MANV, HONV, TENLOT, TENNV} (\sigma_{PHG=5} (NHANVIEN))$$

$$TMP \leftarrow NV_P5 \bowtie_{MANV=MA_NVLEN} PHONGBAN \bowtie_{SODA=MADA} DEAN$$

$$\pi_{TENN, TENDA, THOIGIAN} (TMP)$$


125

5.3.Cập nhật trên khung nhìn

- Có thể dùng các câu lệnh INSERT, DELETE và UPDATE cho các khung nhìn đơn giản
 - Khung nhìn được xây dựng trên 1 bảng và có khóa chính của bảng
- Không thể cập nhật dữ liệu nếu
 - Khung nhìn có dùng từ khóa DISTINCT
 - Khung nhìn có sử dụng các hàm kết hợp
 - Khung nhìn có mệnh đề SELECT mở rộng
 - Khung nhìn được xây dựng từ bảng có RB trên cột
 - Khung nhìn được xây dựng từ nhiều bảng



126

5.3.Cập nhật trên khung nhìn (tt)

- Sửa lại họ cho nhân viên mã '123456789' ở phòng 5 là 'Pham'.

```
UPDATE NV_P5  
SET HONV='Pham'  
WHERE MANV= '123456789'
```



127

6.Chỉ mục (index)

- Định nghĩa
- Chọn lựa chỉ mục



128

6.Chỉ mục

- Chỉ mục trên thuộc tính A là một cấu trúc dữ liệu làm cho việc tìm kiếm mẫu tin có chứa A hiệu quả hơn

```
SELECT *
FROM NHANVIEN
WHERE PHG=5 AND PHAI='Nu'
```

Đọc 10.000 bộ

Đọc 200 bộ

Bảng NHANVIEN có 10.000 bộ
Có 200 nhân viên làm việc cho phòng 5

Đọc 70 bộ



129

6.Chỉ mục (tt)

- Cú pháp

```
CREATE INDEX <tên chỉ mục> ON <tên bảng>(<tên cột>)
```

```
DROP INDEX <tên chỉ mục>
```

- Ví dụ

```
CREATE INDEX PHG_IND ON NHANVIEN(PHG)
```

```
CREATE INDEX PHG_PHA1_IND ON
NHANVIEN(PHG, PHAI)
```



130

6.Chỉ mục (tt)

- Nhận xét
 - Tìm kiếm nhanh trong trường hợp so sánh với hằng số và phép kết
 - Làm chậm đi các thao tác thêm, xóa và sửa
 - Tốn chi phí
 - Lưu trữ chỉ mục
 - Truy xuất đĩa nhiều
- Chọn lựa cài đặt chỉ mục hợp lý???



131

Ví dụ 38

- Xét quan hệ
 - PHANCONG(MA_NVIEN, SODA, THOIGIAN)
- Giả sử
 - PHANCONG được lưu trữ trong 10 block
 - Chi phí để đọc toàn bộ dữ liệu của PHANCONG là 10
 - Trung bình một nhân viên tham gia 3 đề án và một đề án có khoảng 3 nhân viên làm
 - Dữ liệu được trải đều trong 10 block
 - Chi phí để tìm một nhân viên hay một đề án là 3
 - Khi sử dụng chỉ mục
 - Chi phí đọc hay cập nhật chỉ mục
 - Thao tác thêm cần 2 lần truy xuất đĩa



132

Ví dụ 38(tt)

- Giả sử có 3 thao tác được thực hiện thường xuyên

– Q1

```
SELECT SODA, THOIGIAN
FROM PHANCONG
WHERE MA_NVIEN='123456789'
```

– Q2

```
SELECT MANV
FROM PHANCONG
WHERE SODA=1 AND THOIGIAN=20.5
```

– Q3

```
INSERT INTO PHANCONG
VALUES ( '123456789', 1, 20.5)
```



133

Ví dụ 38(tt)

- Bảng so sánh chi phí

Thao tác	Không có chỉ mục	Chỉ mục trên MA_NVIEN	Chỉ mục trên SODA	Chỉ mục trên cả 2 thuộc tính
Q1	10	4	10	4
Q2	10	10	4	4
Q3	2	4	4	6
Chi phí TB	$2 + 8p_1 + 8p_2$	$4 + 6p_2$	$4 + 6p_1$	$6 - 2p_1 - 2p_2$

Khoảng thời gian thực hiện Q1 là p_1

Khoảng thời gian thực hiện Q2 là p_2

Khoảng thời gian thực hiện Q3 là $1 - p_1 - p_2$



134



Phone: 0650. 3834930

Website: www.fit.tdmu.edu.vn