
Chương 4

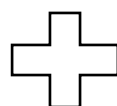
View & Stored Procedure

I. View

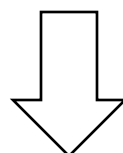
1. Định nghĩa

- ☐ View là một bảng tạm thời, **không lưu trữ dữ liệu mà nó được tạo ra để nhận dữ liệu từ các bảng khác**
- ☐ View được tạo ra từ câu lệnh truy vấn (SELECT), truy vấn từ một hoặc nhiều bảng dữ liệu
- ☐ View được sử dụng:
 - ☐ Khai thác dữ liệu từ nhiều bảng
 - ☐ Chia sẻ khai thác cho nhiều người dùng
 - ☐ An toàn trong khai thác
 - ☐ Không ảnh hưởng dữ liệu gốc

MaNV	HoTen	NgaySinh	Diachi	GT	Luong	MaPh...	TrinhDo	ChuyenMon
1	Nguyễn Thị Hà	1980-04-05 ...	Hùng Hà - Thái Bình	Nữ	2.67	1	Đại Học	Kế Toán
2	Nguyễn Văn Hải	1979-04-06 ...	Thái Thụy - Thái Bình	Nam	4	1	Thạc Sỹ	Kế Toán
3	Trần Thị Thúy	1981-08-08 ...	Thái Nguyên	Nữ	4	1	Đại Học	Kế Toán



MaPhong	TenPhong	DienThoai	Email
1	Hành Chính	0435531169	phanhchinh@fithou.edu.vn
2	Tài Vụ	0436631169	ptaivu@fithou.edu.vn
3	Kế Toán	0436677769	pketoan@fithou.edu.vn
4	Tổng hợp	0436636748	ptonghop@fithou.edu.vn
5	Nghiên cứu khoa học	0488677769	pnckh@fithou.edu.vn



mAnv	hOTEN	dlAcHI	gt	lUONG	tENpHONG	tRINHdO	cHUYENmON
1	Nguyễn Thị Hà	Hùng Hà - Thái Bình	Nữ	2.67	Hành Chính	Đại Học	Kế Toán
2	Nguyễn Văn Hải	Thái Thụy - Thái Bình	Nam	4	Hành Chính	Thạc Sỹ	Kế Toán
3	Trần Thị Thúy	Thái Nguyên	Nữ	4	Hành Chính	Đại Học	Kế Toán

2. Tạo View

- Cú pháp:

CREATE VIEW tên-khung-nhìn[(ds-tên-cột)]

as

câu-lệnh-SELECT

- Lưu ý

- Nếu không chỉ định danh sách tên cột cho khung nhìn, tên các cột chính là tiêu đề các cột trong kết quả câu SELECT
- Tên các cột trong view được chỉ định thì phải có cùng số lượng với số lượng cột trong kết quả của câu SELECT
- Phải đặt tên cho cột của View (**ds-tên-cột**) trong trường hợp sau:
 - Cột được phát sinh từ một biểu thức số học, hoặc hàm có sẵn hay hằng
 - Tồn tại hai cột trong kết quả của câu lệnh SELECT có cùng tiêu đề cột
 - Đổi tên của cột so với tên của cột trong bảng CSDL.

- Ví dụ:

```
CREATE VIEW thongtin_nv
```

```
AS
```

```
SELECT manv, hoten, tendonvi
```

```
FROM nhanvien, donvi
```

```
WHERE nhanvien.madv = donvi.madv
```

```
CREATE VIEW thongtin_nv (manv, hoten, tuoi, tendonvi)
```

```
AS
```

```
SELECT manv,hoten, datediff(year,ngaysinh,getdate()), tendonvi
```

```
FROM nhanvien,donvi
```

```
WHERE nhanvien.madv = donvi.madv
```

■ Ví dụ:

NHANVIEN(MaNV, Hoten, DiaChi, SDT, NgaySinh, GT, HSL)

HANG(MaHang, TenHang, NhaSX, TGianBaoHanh)

KHACHHANG(MaKH, TenKH, CMT, DiaChi, SoDienThoai, Email)

HOADONXUAT(MaHD, MaKH, NgayLapHD, MaNV, PhuongThucTT)

CT_HOADON(MaHD, MaHang, SoLuongMua, DonGia)

- Tạo View chứa danh sách nhân viên nữ
- Tạo View chứa danh sách nhân viên với các thông tin: mã nhân viên, họ tên, giới tính và tuổi
- Tạo View cho biết họ tên của khách hàng đã mua tổng số tiền hàng > 10 triệu
- Tạo View cho biết họ tên của nhân viên đã bán được > 20 triệu tiền hàng

*Xóa khung nhìn

- Cú pháp:

drop view *tên-khung-nhìn*

- Ví dụ:

drop view thongtin_nv

- *Lưu ý:*

- Khi một View bị xóa thì các quyền được cấp phát cho người dùng trên View đó cũng sẽ bị xóa. Nên khi tạo lại View thì phải cấp phát lại quyền cho người sử dụng

3. Thay đổi định nghĩa khung nhìn

- Dùng để định nghĩa lại View mà không làm thay đổi các quyền đã được cấp phát cho người dùng trước đó.
- Cú pháp:

```
ALTER VIEW tên_khung_nhìn [ (danh_sách_tên_cột) ]  
as  
câu_lệnh_SELECT
```


- Ví dụ:

CREATE VIEW viewlop

AS

SELECT malop, tenlop, tenkhoa

FROM lop, khoa

WHERE lop.makhoa=khoa.makhoa *AND* tenkhoa = N 'Khoa Công nghệ'

- Định nghĩa lại:

ALTER VIEW viewlop

AS

SELECT malop, tenlop, hedaotao

FROM lop, khoa

WHERE lop.makhoa=khoa.makhoa *AND* tenkhoa= N 'Khoa Công nghệ tin học'

Cho Database gồm

Khoa(MaK, TenK, SoDT)

Lop(MaLop, TenLop, Siso, MaK)

Sinhvien(MaSV,TenSV,NS,GT,MaLop)

MonHoc(MaMon,TenMon,SoDVHT)

Diem(MaSV,MaMon,Ketqua, Lanthi)

■ Tạo view

- Lấy ra danh sách sinh viên nữ học môn cơ sở dữ liệu và kết quả
- Cho biết số sinh viên đã qua môn toán rời rạc
- Lấy ra tên sinh viên và điểm trung bình của các sinh viên
- Cho biết số sinh viên học lại của từng môn

II. Thủ tục

1. *Khái niệm*

- Thủ tục lưu trữ (**Procedure**): là một đối tượng trong CSDL gồm tập nhiều lệnh SQL được nhóm lại thành một nhóm và các lệnh này sẽ được thực hiện khi thủ tục lưu trữ được thực thi.
- Thủ tục lưu trữ có thể có các thành phần:
 - ✓ *Cấu trúc điều khiển (**IF, WHILE, FOR**)*
 - ✓ *Biến để lưu các giá trị tính toán, các giá trị truy xuất từ CSDL*
 - ✓ *Các câu lệnh SQL được kết hợp thành khối lệnh trong thủ tục, một thủ tục có thể có tham số truyền vào hay giá trị trả về giống như ngôn ngữ lập trình thông thường*

2. Tạo thủ tục

```
CREATE PROC name-proc [ ( parameters ) ]  
[with RECOMPILE| ENCRYPTION | RECOMPILE, ENCRYPTION ]  
AS  
    Begin  
        các câu lệnh của thủ tục  
    End
```

Trong đó

- **Name-proc:** tên thủ tục cần tạo ra, tuân thủ nguyên tắc định danh, không quá 128 ký tự
- **Parameters:** các tham số truyền vào để thủ tục thực hiện
 - Các tham số cách nhau bởi dấu phẩy
 - Cú pháp:

@tên-tham-số *kiểu-tham-số*
 - VD:

@mamonhoc nvarchar(10)
- **RECOMPILE:** cho phép dịch lại mỗi khi được gọi
- **ENCRYPTION:** cho phép mã hóa thủ tục => không xem được nội dung của thủ tục nữa

VD: Lập DSNV của phòng ban nào đó

```
CREATE PROC DSNV_PB ( @mapb nvarchar(9) ,  
                      @gt nvarchar(3) )
```

As

Begin

```
    Select *
```

```
    From NHANVIEN
```

```
    WHERE Phg = @mapb and GT = @gt
```

End

⇒ Áp dụng

```
Execute DSNV_PB('01')
```

3. Lời gọi thủ tục

- Khi cần sử dụng thủ tục thì thực hiện lời gọi thủ tục
- Cú pháp:
name-proc [danh-sách-đối-số]
- *Số lượng đối số và thứ tự phải phù hợp với số lượng và thứ tự của tham số khi định nghĩa*

- Cách truyền tham số:

DSNV_PB (@mapb nvarchar(9) , @gt nvarchar(3))

- Gán giá trị theo thứ tự:

DSNV_PB 'PB05' , N 'Nữ'

- Gán giá trị theo tên biến

sp_DSДиem @mapb = 'PB05', @gt = N 'Nữ'

- Gán giá trị theo tên biến, không theo thứ tự

sp_DSДиem @gt = N 'Nữ', @mapb = 'PB05'

Chú ý:

- Nếu lời gọi thủ tục được thực hiện trong một thủ tục khác, trong một trigger hay kết hợp với câu lệnh SQL khác, sử dụng:

Execute *name-proc* [ds-các-đổi-số]

4. *Sử dụng biến trong thủ tục*

- Bên trong thủ tục có thể sử dụng các biến để: lưu trữ các giá trị tính toán hoặc truy xuất đến CSDL
- Các biến được khai báo với **declare** theo:
declare @tên-biến kiểu-dữ-liệu

***Thủ tục cho DS Lương cao nhất**

-- NHANVIEN(MaNV, HoTen, NgaySinh, Luong,, MaPB)

CREATE PROC DSNV_Luong_Max

As

Begin

declare @maxLuong float

set @maxLuong = Select max(Luong)
From NHANVIEN

select MaNV, HoTen, NgaySinh
from NHANVIEN

where Luong = @maxLuong

End

BÀI TẬP THỰC HÀNH

NHANVIEN(MaNV, Hoten, DiaChi, SDT, NgaySinh, GT, HSL)

HANG(MaHang, TenHang, NhaSX, TGianBaoHanh)

KHACHHANG(MaKH, TenKH, CMT, DiaChi, SoDienThoai, Email)

HOADONXUAT(MaHD, MaKH, NgayLapHD, MaNV, PhuongThucTT)

CT_HOADON(MaHD, MaHang, SoLuongMua, DonGia)

- Tạo Proc tính tổng tiền đã mua hàng của một khách hàng nào đó theo mã KH
- Tạo Proc cho biết tổng số tiền hàng đã mua của một hóa đơn nào đó
- Tạo Pro cho biết tổng số tiền hàng đã bán của một tháng nào đó
- Tạo Proc cho biết họ tên của nhân viên có tuổi cao nhất