

LẬP TRÌNH TRONG SQL SERVER

SỬ DỤNG BIẾN

❖ Khai báo biến

+ Cú pháp

`Declare @Tên_biến Kiểu [...]`

+ Ví dụ:

`Declare @Tong int, @Ngay date`

`Declare @Hoten nvarchar(45)`

❖ Gán giá trị cho biến

➤ Sử dụng lệnh Set/Select

- Để gán giá trị cụ thể hoặc biểu thức tính toán cho biến.
- Để gán giá trị được lấy ra hoặc tính toán từ DL của một cột bên trong các bảng dữ liệu.

Ví dụ 1:

```
set @Tong=0
```

```
select @Hoten='Lê Thị Hà'
```

Ví dụ 2: Tính tổng lương của các nhân viên

Declare @TL real

Select @TL=Sum(HSLuong*830000+PhuCap)

From NhanVien NV, ChucVu CV

Where NV.MaCV =CV.MaCV

❖ Xem/in giá trị của biến

Print @Tên_biến

Hoặc **Print Biểu_thức_chuỗi**

Lưu ý: Nếu **biến** trong **Biểu_thức_chuỗi** mà không phải là kiểu chuỗi, thì ta phải sử dụng hàm **STR/CAST/CONVERT** để chuyển đổi về kiểu chuỗi.

- + Hàm **STR(Biểu_thức_số)**
- + Hàm **CAST(<Biểu_Thức> AS Kiểu)**
- + Hàm **CONVERT(Kiểu,<Biểu_Thức>[,<Định_dạng>])**

Ví dụ:

Print 'Tổng lương là:'

Print @TL

Hoặc: Print 'Tổng lương là:' + Str(@TL)

Hoặc: Print 'Tổng lương là:' + Cast(@TL as char(8))

Hoặc: Print 'Tổng lương là:' + Convert(char(8),@TL)

CÁC CẤU TRÚC ĐIỀU KHIỂN

❖ Cấu trúc IF

➤ Dạng 1:

if điều_kiên1

 khối_lệnh1

[else if điều_kiên2

 khối_lệnh2

...

else

 khối_lệnh_khác]

➤ **Dạng 2:**

```
if [not] exists(Lệnh Select 1)
    khối_lệnh1
[else if [not] exists(Lệnh Select 2)
    khối_lệnh2
...
else
    khối_lệnh_khác]
```

Lưu ý:

- + **khối_lệnh** mà >1 lệnh thì phải đặt trong cặp **Begin....End**
- + từ khoá **exists** cho kết quả=true, khi câu lệnh Select trả về ít nhất 1 dòng DL, ngược lại cho kết quả =false.

Ví dụ: Tìm những nhân viên có HSLuong >6. Nếu tìm thấy thì đưa ra danh sách những nhân viên đó. Nếu không tìm thấy thì đưa ra thông báo '**không tìm thấy!**'

+ Dạng 1:

```
Declare @dem int
```

```
Select @dem= count(*) from NhanVien Where HSLuong>6
```

```
If @dem>0
```

```
begin
```

```
Print 'DS nhân viên'
```

```
Select * from NhanVien Where HSLuong>6
```

```
end
```

```
Else
```

```
Print 'không tìm thấy!'
```

+ Dạng 2:

If exists (Select * from NhanVien Where HSLuong>6)

begin

Print 'DS nhân viên'

Select * from NhanVien Where HSLuong>6

end

Else

Print 'không tìm thấy!'

CẤU TRÚC WHILE

+ Dạng 1:

While điều_kiện

khối_lệnh [break] [continue]

Ví dụ: in ra màn hình các số từ 1-100

+ Dạng 2:

While Exists(Lệnh Select)

khối_lệnh [break] [continue]

LẬP TRÌNH SỬ DỤNG HÀM - FUNCTION

❖ Function: là hàm có giá trị trả về - tức là hàm có Return.

❖ **Có 2 loại Function:**

- Hàm trả về giá trị kiểu vô hướng (Scalar Valued Function)
- Hàm trả về giá trị kiểu bảng (Table Valued Function)

HÀM TRẢ VỀ GIÁ TRỊ

❖ **Định nghĩa:** Là loại hàm sẽ trả về kết quả là một giá trị: số, chuỗi, hoặc logic.

❖ **Cú pháp hàm:**

Create Function TenHam(@ThamSo1 KieuDuLieu1, @ThamSo2 KieuDuLieu2,,,)
Returns KieuDuLieuTraVe

As

Begin --*Bắt đầu khối lệnh*

Declare @Bien KieuDuLieuTraVe --*Khai báo biến cục bộ*

-- *Xử lý thay đổi trên hàm*

Return @Bien

End --*Kết thúc khối lệnh*

❖ **Gọi hàm:** Select dbo.TenHam(DoiSo1,DoiSo2, , ,)

❖ Xóa hàm:

- Drop Function TenHam
- Hoặc: Click chuột phải trên tên hàm -> Delete

❖ Sửa lại nội dung hàm:

- Alter Function TenHam -- Các thay đổi nội dung hàm

Ví dụ 1 : Tạo hàm tính ngày trong tuần (thứ trong tuần) với TS vào là giá trị kiểu ngày.

```
Create Function fn_Thu(@ngay DATE)
Returns nvarchar(10)
As
Begin
    Declare @st nvarchar(10)
    Select @st=Case Datepart(dw,@ngay)
                When 1 Then 'Chủ nhật'
                When 2 Then 'Thứ hai'
                When 3 Then 'Thứ ba'
                When 4 Then 'Thứ tư'
                When 5 Then 'Thứ năm'
                When 6 Then 'Thứ sáu'
                Else 'Thứ bảy'
    End
    Return (@st) /* Trị trả về của hàm */
End
```

Sử dụng hàm: như là một thành phần trong biểu thức.

```
SELECT MaNV,HoTen,NgaySinh,
       NgayThu=dbo.fn_Thu(ngaysinh)
FROM NhanVien
WHERE MaDV='A01'
```

- **Cho CSDL QLBanHang:**

- SanPham(MaSP, MaHangSX, TenSP, SoLuong, MauSac, GiaBan, DonViTinh, MoTa)
- HangSX(MaHangSX, TenHang, DiaChi, SoDT, Email)
- NhanVien(MaNV, TenNV, GioiTinh, DiaChi, SoDT, Email, TenPhong)
- Nhap(SoHDN, MaSP, SoLuongN, DonGiaN)
- PNhap(SoHDN, NgayNhap, MaNV)
- Xuat(SoHDX, MaSP, SoLuongX)
- PXuat(SoHDX, NgayXuat, MaNV)

- **Ví dụ 2. Tạo hàm Đưa ra tên HangSX khi nhập vào MaSP từ bàn phím**

Create Function fn_TimHang(@MaSP nchar(10))

Returns nvarchar(20)

As

Begin

Declare @ten nvarchar(20)

Set @ten = (Select TenHang From HangSX Inner join SanPham
on HangSX.MaHangSX = SanPham.MaHangSX
Where MaSP = @MaSP)

Return @ten

End

■ **Gọi hàm:**

Select dbo.fn_TimHang('sp01')

- **Ví dụ 3.** Tạo hàm đưa ra tổng giá trị nhập từ năm nhập x đến năm nhập y, với x, y được nhập vào từ bàn phím.

Create Function fn_ThongKeNhapTheoNam(@x int,@y int)

Returns int

As

Begin

Declare @tongTien int

Select @tongTien = sum(SoLuongN*DonGiaN)

From Nhap Inner join PNhap on Nhap.SoHDN=PNhap.SoHDN

Where Year(NgayNhap) Between @x And @y

Return @tongTien

End

❖ **Gọi hàm:**

Select dbo.fn_ThongKeNhapTheoNam(2016,2020)