

FUNCTION

Nội dung

1. Tổng quan
2. Bài tập

I. TỔNG QUAN

1. Function (Hàm) là gì?

- Hàm là một đối tượng trong cơ sở dữ liệu tương tự như Thủ tục.
- Hàm phải trả về giá trị thông qua câu lệnh RETURN. Giá trị trả về có thể là vô hướng hoặc một bảng.
- Điểm khác biệt giữa hàm và thủ tục là hàm trả về một giá trị thông qua tên hàm còn thủ tục thì không.
- Ngoài các hàm được cung cấp sẵn của hệ quản trị, người sử dụng có thể tạo thêm các hàm cho riêng mình.

I. TỔNG QUAN

2. Khởi tạo Hàm

2.1. Cú pháp:

CREATE FUNCTION tên_hàm [(danh_sách_tham_số)]

RETURNS kiểu dữ liệu trả về của hàm

AS

BEGIN

Các_câu_lệnh_của_hàm

END

I. TỔNG QUAN

2.2. Ví dụ:

2.2.1. Định nghĩa hàm tính tổng hai số

--ĐỊNH NGHĨA HÀM TÍNH TỔNG HAI SỐ

```
CREATE FUNCTION ufc_tinhtong (@a int, @b int)
```

```
returns int
```

```
as
```

```
begin
```

```
    declare @tong int
```

```
    set @tong = @a + @b
```

```
    return @tong
```

```
end
```

--GỌI HÀM

```
print [dbo].[ufc_tinhtong] (10, 10)
```

```
-- [dbo]: tên của người sở hữu hàm
```

I. TỔNG QUAN

2.2. Ví dụ:

2.2.2. Định nghĩa hàm tính ngày trong tuần (thứ trong tuần) của một giá trị kiểu ngày

```
CREATE FUNCTION ufc_thu(@ngay DATETIME)
RETURNS NVARCHAR(10)
AS
BEGIN
    DECLARE @st NVARCHAR(10)
    SELECT @st=CASE DATEPART(DW,@ngay)
        WHEN 1 THEN N'Chủ nhật'
        WHEN 2 THEN N'Thứ hai'
        WHEN 3 THEN N'Thứ ba'
        WHEN 4 THEN N'Thứ tư'
        WHEN 5 THEN N'Thứ năm'
        WHEN 6 THEN N'Thứ sáu'
        ELSE N'Thứ bảy'
    END
    RETURN (@st) /* Trị trả về của hàm */
END
```

I. TỔNG QUAN

2.3. Hàm trả về giá trị kiểu bảng

Xét ví dụ:

```
create view view_thongtinnhanvien
```

```
as
```

```
select nv.honv + nv.tenlot + nv.tennv as N'Họ và tên NV', nv.ngsinh, nv.phai  
from NHANVIEN nv inner join PHONGBAN pb on nv.phg = pb.maphg  
where pb.tenphg = 'nghien cuu'
```

Câu lệnh sau dùng để làm gì?

```
select * from view_thongtinnhanvien
```

Muốn xem thông tin nhân viên ở phòng ban khác => làm như thế nào?

Cần có công cụ để trả về dữ liệu ở dạng bảng thông qua tham số truyền vào
=> hàm nội tuyến (inline function)

I. TỔNG QUAN

2.3. Hàm trả về giá trị kiểu bảng

Cú pháp:

CREATE FUNCTION tên_hàm [(danh_sách_tham_số)]

RETURNS TABLE

AS

BEGIN

Các_câu_lệnh_của_hàm

END

I. TỔNG QUAN

2.3. Hàm trả về giá trị kiểu bảng (hàm nội tuyến)

Cú pháp của hàm nội tuyến phải tuân theo các quy tắc sau:

- Kiểu trả về của hàm phải được chỉ định bởi mệnh đề **RETURNS TABLE**.
- Trong phần thân của hàm chỉ có duy nhất một câu lệnh **RETURN** xác định giá trị trả về của hàm thông qua duy nhất một câu lệnh **SELECT**. Ngoài ra, không sử dụng bất kỳ câu lệnh nào khác trong phần thân của hàm.

I. TỔNG QUAN

Ví dụ: định nghĩa hàm xem thông tin nhân viên phòng ban, với tham số truyền vào là tên phòng ban.

```
create function ufc_thongtinnhanvien (@tenphong char(10))
returns table
as
return (select nv.honv + nv.tenlot + nv.tennv as N'Họ và tên NV',
nv.ngsinh, nv.phai
from NHANVIEN nv inner join PHONGBAN pb on nv.phg =
pb.maphg
where pb.tenphg = @tenphong)
```

```
select * from [dbo].[ufc_thongtinnhanvien] ('nghien cuu')
```

I. TỔNG QUAN

2.4. Hàm trả nội tuyến với nhiều câu lệnh

Đối với hàm nội tuyến, phần thân của hàm chỉ cho phép sự xuất hiện duy nhất của câu lệnh RETURN.

Trong trường hợp cần phải sử dụng đến nhiều câu lệnh trong phần thân của hàm, ta sử dụng cú pháp như sau để định nghĩa hàm:

I. TỔNG QUAN

2.4. Hàm trả nội tuyến với nhiều câu lệnh

```
CREATE FUNCTION tên_hàm([danh_sách_tham_số])  
RETURNS @biến_bảng TABLE -- định_nghĩa_bảng  
AS  
BEGIN  
    các_câu_lệnh_trong_thân_hàm  
    RETURN  
END
```

Ví dụ: định nghĩa hàm tính tổng số nhân viên theo phòng ban

-- định nghĩa hàm tính tổng nhân viên theo phòng ban

```
create function ufc_tongNVtheoPB (@tenphong char(10))
```

```
returns @thongkenv table (maphong int, tenphong char(10), tongnv int)
```

```
as
```

```
begin
```

```
    if @tenphong = ''
```

```
        insert into @thongkenv(maphong,tenphong,tongnv)
```

```
        select pb.maphg, pb.tenphg, count(nv.manv)
```

```
        from NHANVIEN nv inner join PHONGBAN pb on nv.phg = pb.maphg
```

```
        group by pb.maphg, pb.tenphg
```

```
    else
```

```
        insert into @thongkenv(maphong,tenphong,tongnv)
```

```
        select pb.maphg, pb.tenphg, count(nv.manv)
```

```
        from NHANVIEN nv inner join PHONGBAN pb on nv.phg = pb.maphg
```

```
        where pb.tenphg = @tenphong
```

```
        group by pb.maphg, pb.tenphg
```

```
    return
```

```
end
```

-- gọi hàm

```
select * from [dbo].[ufc_tongNVtheoPB](''); select * from [dbo].[ufc_tongNVtheoPB]('dieu hanh')
```

II. MỘT SỐ VÍ DỤ

--1. Tính tổng số nhân viên theo giới tính

```
create function ufc_tongnvtheogioitinh (@phai char(3))
returns int
as
begin
return (select count(nv.manv) as N'Tổng nhân viên theo phái'
from NHANVIEN nv
where nv.phai = @phai)
end
```

-- gọi hàm cách 1:

```
print [dbo].[ufc_tongnvtheogioitinh] ('nam')
print [dbo].[ufc_tongnvtheogioitinh] ('nu')
```

--gọi hàm cách 2: sử dụng biến để nhận giá trị trả về của hàm

```
declare @sonv int
select @sonv = [dbo].[ufc_tongnvtheogioitinh]('nam')
print @sonv
```

II. MỘT SỐ VÍ DỤ

-- 2. Tính tuổi của nhân viên theo mã nhân viên

create function ufc_tinhthtuoì(@manv char(3))

returns int

as

begin

return (select DATEDIFF(YEAR,nv.ngsinh,getdate()) as 'tuoi'

from NHANVIEN nv

where nv.manv = @manv)

end

-- gọi hàm

print N'tuoi của nhan vien là: ' + cast([dbo].[ufc_tinhthtuoì] ('123') as
varchar)

II. MỘT SỐ VÍ DỤ

--3. Định nghĩa hàm trả về thông tin nhân viên làm việc ở phòng bất kỳ do người dùng nhập vào.

```
create function ufc_nhanvienphong (@maphg int)
```

```
returns table
```

```
as
```

```
return
```

```
(select nv.honv + nv.tenlot + nv.tennv as N'họ tên nhân viên', nv.ngsinh,
```

```
nv.phai
```

```
from NHANVIEN nv
```

```
where nv.phg = @maphg)
```

-- thực thi hàm

```
select * from [dbo].[ufc_nhanvienphong] (5)
```


III. BÀI TẬP

1. Tạo hàm đếm số thân nhân theo mã nhân viên.

III. BÀI TẬP

2. Tạo hàm tính số đề án mà nhân viên được phân công.

III. BÀI TẬP

3. Tạo hàm đếm số đề án do phòng ban chủ trì.

III. BÀI TẬP

4. Tạo hàm cho biết thông tin thân nhân (tên, phái, quan hệ) của nhân viên

III. BÀI TẬP

5. Tạo hàm cho biết thông tin làm việc theo đề án của nhân viên (mã nhân viên, số đề án và thời gian làm việc của nhân viên theo đề án đó). Lưu ý: nếu không nhập vào mã nhân viên thì sẽ in ra tất cả thông tin làm việc của tất cả nhân viên được phân công, ngược lại sẽ in ra thông tin làm việc của mã nhân viên được truyền vào.

III. BÀI TẬP

6. Tạo hàm nhận giá trị đầu vào là mã phòng, sau đó cho biết thông tin nhân viên (họ, tên lót, tên, phái, lương) của nhân viên có mức lương lớn hơn lương trung bình của phòng ban đó.