



Chương 7

FUNCTION



Mục tiêu

- Sau khi học xong chương này, SV có thể:
 - Hiểu khái quát về Function (Hàm)
 - Nắm vững cú pháp xây dựng Hàm
 - Biết cách gọi thực hiện Hàm
 - Xây dựng được các Hàm theo yêu cầu
 - Biết các lệnh quản lý Hàm

Nội dung chi tiết

- Khái quát về Function (Hàm)
- Phân loại Hàm
- Cú pháp xây dựng và gọi thực hiện Hàm
- Ví dụ minh họa
- Các lệnh quản lý Hàm

Khái quát về Function (Hàm)

- Function (Hàm) là một tập hợp các câu lệnh SQL được tạo ra với mục đích sử dụng lại (reuse).
- Có tham số đầu vào, nhưng không có tham số đầu ra.
- Có thể trả về 1 giá trị hoặc 1 bảng dữ liệu.
- Có thể gọi hàm trong câu truy vấn, Stored Procedure, Trigger và Constraint.
- Chia làm 2 loại:
 - Hàm có sẵn do SQL cung cấp.
 - Hàm do người dùng định nghĩa (user-defined).

Phân loại Hàm

- **Hàm có sẵn do SQL cung cấp:**

- **Hàm xác định:** trả về giá trị không thay đổi trong những lần gọi khác nhau. (VD: round, left, len, upper, reverse, ...)
- **Hàm không xác định:** trả về giá trị thay đổi trong những lần gọi khác nhau. (VD: getdate(), ...)

- **Hàm do người dùng định nghĩa (user-defined):**

- **Scalar-valued Function:** hàm trả về 1 giá trị (hàm đơn trị).
- **Table-valued Function:** hàm trả về 1 bảng.

Phân loại Hàm

- **Phân loại:**
 - **Hàm trả về 1 giá trị** (Scalar-valued Functions)
 - **Hàm trả về 1 bảng** (Table-valued Functions)
 - **Hàm đọc bảng:** sử dụng một câu lệnh Select để đọc giá trị từ các bảng, kết quả trả về là kết quả của câu Select.
 - **Hàm tạo bảng:** định nghĩa cấu trúc bảng kết quả, sử dụng lệnh Insert để đưa dữ liệu vào bảng.

Cú pháp xây dựng và gọi thực hiện

■ Hàm trả về 1 giá trị:

- Cú pháp xây dựng Hàm trả về 1 giá trị:

create function Tên_Hàm (Khai_báo_các_tham_số)

returns Kiểu_dữ_liệu_trả_về

as

begin

-- Các câu lệnh bên trong Hàm

return Giá_trị_trả_về

end

- Chú ý:

- Phải có **begin**, **end**
- Kết quả trả về là 1 giá trị đơn.

Cú pháp xây dựng và gọi thực hiện

■ Hàm trả về 1 giá trị:

- Gọi thực hiện Hàm:
- Sử dụng ở bất cứ nơi nào có thể thay bằng một giá trị đơn, trong:
 - Các mệnh đề: Select, Where, Having, Group by
 - Set của câu Update
 - Values của câu Insert
 - Mô tả Check Constraint
 - Mô tả giá trị Default
 - Mô tả công thức cột tính toán
 - Câu lệnh Case

Cú pháp xây dựng và gọi thực hiện

■ Hàm trả về 1 giá trị:

- VD1: Xây dựng và gọi Hàm trả về chuỗi ngày tháng kiểu Việt Nam với định dạng dd/MM/yyyy

```
create function fn_NgayThangVN(@ngay datetime)
returns char(10)
as
begin
    declare @day char(10)
    if @ngay is null
        set @day=CONVERT(char(10),getdate(),103)
    else
        set @day=CONVERT(char(10),@ngay,103)
    return @day
end
go

declare @ngay datetime
print dbo.fn_NgayThangVN(@ngay)
go
```

100 % <

Messages

28/07/2019

Cú pháp xây dựng và gọi thực hiện

■ Hàm trả về 1 giá trị:

- Gọi hàm trả về 1 giá trị trong mệnh đề Select:

```
select MASV,HOSV,TENSV,dbo.fn_NgayThangVN(NGAYSINH) as NgSinh
from SINHVIEN
go
```

133 %

Results Messages

	MASV	HOSV	TENSV	NgSinh
1	A01	Nguyễn Thu	Hải	23/02/1980
2	A02	Trần Văn	Chính	24/12/1982
3	A03	Lê Thu Bạch	Yến	12/12/1982
4	A04	Trần Anh	Tuấn	08/12/1984
5	A05	Trần Thanh	Triều	01/02/1980
6	B01	Trần Thanh	Mai	20/12/1981
7	B02	Trần Thị Thu	Thủy	13/02/1982
8	B03	Trần Thị	Thanh	31/12/1982

Cú pháp xây dựng và gọi thực hiện

■ Hàm trả về 1 giá trị:

- Gọi hàm trả về 1 giá trị trong mệnh đề Where:

```
select MASV,HOSV,TENSV,NGAYSINH
from SINHVIEN
where dbo.fn_NgayThangVN(NGAYSINH)='23/02/1980'
go
```

133 %

Results Messages

	MASV	HOSV	TENSV	NGAYSINH
1	A01	Nguyễn Thu	Hải	1980-02-23 00:00:00.000

Cú pháp xây dựng và gọi thực hiện

■ Hàm trả về 1 giá trị:

- Gọi hàm trả về 1 giá trị trong mệnh đề Group by:

```
select SINHVIEN.MASV, HOSV, TENSX,  
       dbo.fn_NgayThangVN(NGAYSINH) as NgSinh, ROUND(AVG(DIEM),2) as DTB  
from SINHVIEN inner join KETQUA on SINHVIEN.MASV = KETQUA.MASV  
group by SINHVIEN.MASV, HOSV, TENSX, dbo.fn_NgayThangVN(NGAYSINH)  
go
```

133 %

Results Messages

	MASV	HOSV	TENSX	NgSinh	DTB
1	A04	Trần Anh	Tuấn	08/12/1984	5
2	A03	Lê Thu Bạch	Yến	12/12/1982	8.33
3	B02	Trần Thị Thu	Thủy	13/02/1982	7
4	B01	Trần Thanh	Mai	20/12/1981	4
5	A01	Nguyễn Thu	Hải	23/02/1980	6.5
6	A02	Trần Văn	Chính	24/12/1982	5
7	B03	Trần Thị	Thanh	31/12/1982	9.5

Cú pháp xây dựng và gọi thực hiện

■ Hàm trả về 1 giá trị:

- VD2: Xây dựng Hàm trả về chuỗi đã được cắt bỏ các khoảng trắng thừa và IN HOA.

```
create function fn_ChuanHoaChuoi(@chuoi nvarchar(255))
returns nvarchar(255)
as
begin
    set @chuoi = LTRIM(RTRIM(@chuoi))
    while CHARINDEX(' ',@chuoi)>0
        set @chuoi = REPLACE(@chuoi,' ',' ')
    set @chuoi = UPPER(@chuoi)
    return @chuoi
end
go
print dbo.fn_ChuanHoaChuoi(N'  Trần  Văn  An  ')
go
```

133 % <

Messages

TRẦN VĂN AN

Cú pháp xây dựng và gọi thực hiện

■ Hàm trả về 1 giá trị:

- Gọi hàm trả về 1 giá trị trong mệnh đề Select:

```
select MASV, dbo.fn_ChuanHoaChuoi(HOSV+ ' '+TENSU) as HoTen
from SINHVIEN
go
```

133 %

Results Messages

	MASV	HoTen
1	A01	NGUYỄN THU HẢI
2	A02	TRẦN VĂN CHÍNH
3	A03	LÊ THU BẠCH YẾN
4	A04	TRẦN ANH TUẤN
5	A05	TRẦN THANH TRIỀU
6	B01	TRẦN THANH MAI
7	B02	TRẦN THỊ THU THỦY
8	B03	TRẦN THỊ THANH

Cú pháp xây dựng và gọi thực hiện

■ Hàm trả về 1 giá trị:

- Gọi hàm trả về 1 giá trị trong mệnh đề Where:

```
select *  
from SINHVIEN  
where dbo.fn_ChuanHoaChuoi(HOSV+ ' '+TENSX)= N'LÊ THU BẠCH YẾN'  
go
```

133 % <

Results Messages

	MASV	HOSV	TENSX	PHAI	NGAYSINH	NOISINH	MAKH	HOCBONG
1	A03	Lê Thu Bạch	Yến	0	1982-12-12 00:00:00.000	Hà Nội	AV	140000

Cú pháp xây dựng và gọi thực hiện

■ Hàm trả về 1 giá trị:

- Gọi hàm trả về 1 giá trị trong mệnh Set của câu lệnh Update:

```
update SINHVIEN
set HOSV = dbo.fn_ChuanHoaChuoai(HOSV), TENSVC = dbo.fn_ChuanHoaChuoai(TENSVC)
go

select *
from SINHVIEN
go
```

133 %

Results Messages

	MASV	HOSV	TENSVC	PHAI	NGAYSINH	NOISINH	MAKH	HOCBONG
1	A01	NGUYỄN THU	HẢI	0	1980-02-23 00:00:00.000	TP.HCM	AV	100000
2	A02	TRẦN VĂN	CHÍNH	1	1982-12-24 00:00:00.000	TP.HCM	TH	100000
3	A03	LÊ THU BẠCH	YẾN	0	1982-12-12 00:00:00.000	Hà Nội	AV	140000
4	A04	TRẦN ANH	TUẤN	1	1984-12-08 00:00:00.000	Long An	LS	80000
5	A05	TRẦN THANH	TRIỀU	1	1980-02-01 00:00:00.000	Hà Nội	VL	80000
6	B01	TRẦN THANH	MAI	0	1981-12-20 00:00:00.000	Bến Tre	TH	200000
7	B02	TRẦN THỊ THU	THỦY	0	1982-02-13 00:00:00.000	TP.HCM	TH	30000
8	B03	TRẦN THỊ	THANH	0	1982-12-31 00:00:00.000	TP.HCM	TH	50000

Cú pháp xây dựng và gọi thực hiện

■ Hàm trả về 1 giá trị:

- Gọi hàm trả về 1 giá trị trong mệnh đề Values của câu lệnh Insert:

```
insert into SINHVIEN values  
( 'C02',dbo.fn_ChuanHoaChuoi(N' võ trần hải '), dbo.fn_ChuanHoaChuoi(N' đăng '),  
1, '11-07-1979',N'Bình Định', 'SH',120000)  
go  
  
select *  
from SINHVIEN  
go
```

133 %

Results Messages

	MASV	HOSV	TENSV	PHAI	NGAYSINH	NOISINH	MAKH	HOCBONG
1	A01	NGUYỄN THU	HẢI	0	1980-02-23 00:00:00.000	TP.HCM	AV	100000
2	A02	TRẦN VĂN	CHÍNH	1	1982-12-24 00:00:00.000	TP.HCM	TH	100000
3	A03	LÊ THU BẠCH	YẾN	0	1982-12-12 00:00:00.000	Hà Nội	AV	140000
4	A04	TRẦN ANH	TUẤN	1	1984-12-08 00:00:00.000	Long An	LS	80000
5	A05	TRẦN THANH	TRIỀU	1	1980-02-01 00:00:00.000	Hà Nội	VL	80000
6	B01	TRẦN THANH	MAI	0	1981-12-20 00:00:00.000	Bến Tre	TH	200000
7	B02	TRẦN THỊ THU	THỦY	0	1982-02-13 00:00:00.000	TP.HCM	TH	30000
8	B03	TRẦN THỊ	THANH	0	1982-12-31 00:00:00.000	TP.HCM	TH	50000
9	C02	VÕ TRẦN HẢI	ĐĂNG	1	1979-11-07 00:00:00.000	Bình Định	SH	120000

Cú pháp xây dựng và gọi thực hiện

■ Hàm trả về 1 bảng:

- Cú pháp xây dựng hàm đọc bảng:

create function Tên_Hàm (Khai_báo_các_tham_số)

returns table

as

return câu_Select_trả_về_1_bảng_dữ_liệu

go

- Chú ý:

- Không có **begin**, **end**.
- Kết quả trả về là 1 bảng dữ liệu.

Cú pháp xây dựng và gọi thực hiện

■ Hàm trả về 1 bảng:

- VD: Xây dựng hàm đọc bảng danh sách SV theo Khoa và gọi hàm trong mệnh đề From:

```
create function fn_DSSV_TheoKhoa(@makh char(2))
returns table
as
    return select * from SINHVIEN where MAKH=@makh
go

select *
from dbo.fn_DSSV_TheoKhoa('TH')
go
```

133 %

Results Messages

	MASV	HOSV	TENSV	PHAI	NGAYSINH	NOISINH	MAKH	HOCBONG
1	A02	TRẦN VĂN	CHÍNH	1	1982-12-24 00:00:00.000	TP.HCM	TH	100000
2	B01	TRẦN THANH	MAI	0	1981-12-20 00:00:00.000	Bến Tre	TH	200000
3	B02	TRẦN THỊ THU	THỦY	0	1982-02-13 00:00:00.000	TP.HCM	TH	30000
4	B03	TRẦN THỊ	THANH	0	1982-12-31 00:00:00.000	TP.HCM	TH	50000

Cú pháp xây dựng và gọi thực hiện

■ Hàm trả về 1 bảng:

- Cú pháp xây dựng hàm tạo bảng:

```
create function Tên_Hàm (Khai_báo_các_tham_số)
returns Tên_bảng table (Các_cột_dữ_liệu_bảng_trả_về)
as
begin
    Sử dụng lệnh INSERT để đưa dữ liệu vào bảng trả về
    Cập nhật dữ liệu của các bảng liên quan (nếu có)
end
go
```

- Chú ý:

- Phải có `begin`, `end`.
- Kết quả trả về là 1 bảng mới tạo.

Cú pháp xây dựng và gọi thực hiện

■ Hàm trả về 1 bảng:

- VD: Xây dựng hàm tạo bảng danh sách SV theo Khoa có tăng phần trăm học bổng

```
create function fn_DSSV(@makh char(2), @phantram real)
returns @DSSV table(MS char(3), Ho nvarchar(15), Ten nvarchar(7), HBong int)
as
begin
    insert into @DSSV(MS, Ho, Ten, HBong)
    select MASV, HOSV, TENSX, HOCBONG
    from SINHVIEN
    where MAKH = @makh
    update @DSSV
    set HBong = HBong + (HBong * @phantram)/100
    return
end
go

select *
from fn_DSSV('TH', 5.5)
go
```

133 %

Results Messages

	MS	Ho	Ten	HBong
1	A02	TRẦN VĂN	CHÍNH	105500
2	B01	TRẦN THANH	MAI	211000
3	B02	TRẦN THỊ THU	THÙY	31650
4	B03	TRẦN THỊ	THANH	52750

Ví dụ minh họa

■ Sử dụng CSDL Quản lý bán hoa:

- VD1: Xây dựng hàm lấy đơn giá hoa

```
create function fn_DonGiaHoa(@mahoa char(4))
returns money
as
begin
    declare @dongia money
    select @dongia = DONGIA from HOA where MAHOA=@mahoa
    return @dongia
end
go
```

- Hoặc:

```
create function fn_DonGiaHoa(@mahoa char(4))
returns money
as
begin
    return (select DONGIA from HOA where MAHOA=@mahoa)
end
go
```

Ví dụ minh họa

■ Sử dụng CSDL Quản lý bán hoa:

- Áp dụng hàm lấy đơn giá hoa để định nghĩa ràng buộc kiểm tra giá bán hoa không được lớn hơn 10% đơn giá hoa

```
alter table CTDONHANG  
add constraint CK_CTDONHANG_GIABAN  
check(GIABAN <= dbo.fn_DonGiaHoa(MAHOA)*110/100)
```

Ví dụ minh họa

■ Sử dụng CSDL Quản lý sinh viên:

- Coi hàm như một bảng ảo, sử dụng trong các câu SQL độc lập hay kết hợp với các bảng khác.
- VD2: Xây dựng và gọi thực hiện hàm lọc danh sách SV theo ký tự đầu của Tên.

```
create function fn_LocDSSV(@KyTuDau char(1))
returns table
as
    return (select * from SINHVIEN where left(TENSV,1) = @KyTuDau)
go

select * from dbo.fn_LocDSSV('T')
go
```

133 %

Results Messages

	MASV	HOSV	TENSV	PHAI	NGAYSINH	NOISINH	MAKH	HOCBONG
1	A04	TRẦN ANH	TUẤN	1	1984-12-08 00:00:00.000	Long An	LS	80000
2	A05	TRẦN THANH	TRIỀU	1	1980-02-01 00:00:00.000	Hà Nội	VL	80000
3	B02	TRẦN THỊ THU	THỦY	0	1982-02-13 00:00:00.000	TP.HCM	TH	30000
4	B03	TRẦN THỊ	THANH	0	1982-12-31 00:00:00.000	TP.HCM	TH	50000

Ví dụ minh họa

■ Sử dụng CSDL Quản lý bán hoa:

- Sử dụng tương tự hàm đọc bảng, hàm tạo bảng giúp trả về những bộ giá trị phức tạp cần nhiều tính toán.
- VD3: Xây dựng và gọi hàm lọc danh sách Hoa theo Loại hoa có giảm % Đơn giá.

```
create function fn_DSHoaKhuyenMai(@malh char(4), @phantram real)
returns @HOAKHUYENMAI table(MAHOA char(4), TENHOA nvarchar(50), DONGIA money, DONGIAKM money)
begin
    insert into @HOAKHUYENMAI(MAHOA, TENHOA, DONGIA)
    select MAHOA, TENHOA, DONGIA
    from HOA where MALH=@malh
    update @HOAKHUYENMAI
    set DONGIAKM=DONGIA-(DONGIA*@phantram)/100
    return
end
go
select * from dbo.fn_DSHoaKhuyenMai('LH09', 5.5)
go
```

121 %

Results Messages

	MAHOA	TENHOA	DONGIA	DONGIAKM
1	H009	Tuổi Trẻ	670000.00	633150.00

Quản lý hàm

- Sử dụng các lệnh script tương tự như với thủ tục để quản lý các hàm:
 - Đổi tên hàm:
sp_rename 'Tên_hàm_cũ', 'Tên_hàm_mới'
 - Thay đổi nội dung hàm:
alter function ...
 - Xóa hàm:
drop function Tên_hàm

Bài tập ứng dụng

1. Xây dựng hàm `fn_TongHaiSoNguyen(@so1, @so2)` trả về tổng của hai số nguyên.
2. Xây dựng hàm `fn_TongDaySoNguyen(@n)` trả về tổng của các số nguyên từ 1 đến n.
3. Xây dựng hàm `fn_SoNT(@n)` trả về 1 nếu @n là số nguyên tố, ngược lại thì trả về 0.
4. Xây dựng hàm `fn_CacSoNT(@n)` trả về chuỗi các số nguyên tố nằm trong khoảng từ 2 đến n.

Bài tập ứng dụng

■ Sử dụng CSDL Quản lý sinh viên:

5. Xây dựng hàm fn_DTB_MH(@mamh) trả về điểm TB của môn học có mã số truyền vào.
6. Xây dựng thủ tục sp_CapNhatMH có sử dụng hàm fn_DTB_MH để cập nhật lại số tiết trong bảng MONHOC theo các qui tắc sau:
 - Tăng 10 tiết nếu ĐTB của SV học dưới 5.
 - Tăng 5 tiết nếu ĐTB của SV học từ $5 \leq \text{ĐTB} < 7$
 - Không tăng số tiết nếu ĐTB của SV học ≥ 7 hoặc không có SV học.
7. Xây dựng thủ tục sp_CapNhatMH_KyTuDau(@kytudau) có sử dụng hàm fn_DTB_MH để cập nhật lại số tiết trong bảng MONHOC cho các môn học mà tên có ký tự đầu là “T”.

Bài tập ứng dụng

■ Sử dụng CSDL Quản lý sinh viên:

8. Xây dựng hàm `fn_DanhSachSinhVien_DTB(@makh)` trả về danh sách các SV của mã khoa truyền vào, gồm các thông tin: mã SV, họ tên SV, ĐTB.
9. Xây dựng hàm `fn_DanhSachMonHoc(@masv)` trả về danh sách gồm các thông tin: mã môn học, tên môn học và điểm số tương ứng của mã SV truyền vào.
10. Xây dựng hàm `fn_DSSV_ThiMon(@mamh)` để lọc danh sách SV đã thi môn học với mã môn truyền vào, gồm các thông tin: mã SV, họ tên SV, tên khoa.

Bài tập ứng dụng

■ Sử dụng CSDL Quản lý sinh viên:

11. Xây dựng hàm `fn_DSKhoa_ThiMon(@mamh)` để lọc danh sách khoa có SV đã thi môn học với mã môn truyền vào.
12. Xây dựng hàm `fn_DSKhoa_ThiMon_Diem(@mamh)` để lọc danh sách khoa có SV đã thi môn học với mã môn truyền vào, gồm các thông tin: mã khoa, tên khoa, điểm thi cao nhất, điểm thi thấp nhất và ĐTB.
13. Xây dựng hàm `fn_LocDSSV_CapNhatHB(@makh)` để lọc danh sách SV (gồm các thông tin: mã SV, họ tên SV, học bổng mới) của mã Khoa truyền vào, có cập nhật lại học bổng của SV theo các qui tắc sau:

Bài tập ứng dụng

■ Sử dụng CSDL Quản lý sinh viên:

- Không cấp học bổng nếu ĐTB < 7
- Cấp học bổng 500.000đ nếu $7 \leq \text{ĐTB} < 8$
- Cấp học bổng 800.000đ nếu $8 \leq \text{ĐTB} < 9$
- Cấp học bổng 1.000.000đ nếu $9 \leq \text{ĐTB} \leq 10$

14. Xây dựng hàm fn_LocDSMH_CapNhatSoTiet để lọc danh sách môn học (gồm các thông tin: mã MH, tên MH, ĐTB thi của SV, số tiết cũ, số tiết mới) với số tiết mới của SV được tính theo các qui tắc sau:

- Không tăng số tiết nếu không có SV học hoặc ĐTB của SV học dưới 5.
- Tăng 5 tiết nếu ĐTB của SV học từ $5 \leq \text{ĐTB} < 7$
- Tăng 10 tiết nếu ĐTB của SV học ≥ 7

Câu hỏi trắc nghiệm

1. Chọn phát biểu SAI về Function (Hàm)?

- a. Function (Hàm) là một tập hợp các câu lệnh SQL được tạo ra với mục đích tái sử dụng.
- b. **Function có tham số đầu vào và tham số đầu ra.**
- c. Có thể trả về 1 giá trị hoặc 1 bảng dữ liệu.
- d. Có thể gọi Function trong câu truy vấn, thủ tục, trigger và các ràng buộc.

Câu hỏi trắc nghiệm

2. Hàm getdate() là:

- a. Hàm trả về 1 giá trị do người dùng định nghĩa.
- b. Hàm trả về 1 bảng do người dùng định nghĩa.
- c. Hàm xác định do SQL cung cấp.
- d. **Hàm không xác định do SQL cung cấp.**

Câu hỏi trắc nghiệm

3. Dưới đây là cú pháp xây dựng của hàm loại gì?

```
create function Tên_Hàm (Khai_báo_các_tham_số)
returns Kiểu_dữ_liệu_trả_về
as
begin
    -- Các câu lệnh bên trong Hàm
    return Giá_trị_trả_về
end
```

- a. **Hàm trả về 1 giá trị**
- b. Hàm trả về 1 bảng
- c. Hàm trả về dữ liệu
- d. Hàm do SQL cung cấp sẵn

Câu hỏi trắc nghiệm

4. Hàm xử lý sau đây dùng để làm gì?

```
create function fn_HamXuLy(@so1 int, @so2 int)
returns int
as
begin
    declare @so3 int
    if @so1 < @so2
        set @so3 = @so1
    else
        set @so3 = @so2
    return @so3
end
go
```

- a. Trả về 1 số nguyên.
- b. Trả về giá trị số lớn hơn trong 2 số nguyên nhập vào.
- c. **Trả về giá trị số nhỏ hơn trong 2 số nguyên nhập vào.**
- d. Trả về giá trị @so3.

Câu hỏi trắc nghiệm

5. Cho biết kết quả gọi thực hiện hàm xử lý sau đây:

```
create function fn_HamXuLy(@so1 int, @so2 int)
returns int
as
begin
    declare @so3 int
    if @so1 < @so2
        set @so3 = @so1
    else
        set @so3 = @so2
    return @so3
end
go
print dbo.fn_HamXuLy(3,-7)
go
```

- a. 3
- b. -7
- c. 0
- d. Hàm bị lỗi.