



CHƯƠNG 3

LẬP TRÌNH CƠ SỞ DỮ LIỆU BẰNG T-SQL



HÀM DO NGƯỜI DÙNG ĐỊNH NGHĨA

6. HÀM DO NGƯỜI DÙNG ĐỊNH NGHĨA



- Hàm là đối tượng cơ sở dữ liệu, tương tự như thủ tục.
- Có thể sử dụng hàm như một thành phần của một biểu thức (VD: Trong danh sách chọn của lệnh **SELECT**)
- Hàm có thể do HQT CSDL cung cấp sẵn và người sử dụng có thể định nghĩa các hàm cá nhân.

6. HÀM DO NGƯỜI DÙNG ĐỊNH NGHĨA



□ Giống Stored Procedure

- Là mã lệnh có thể tái sử dụng.
- Chấp nhận các tham số input.
- Dịch một lần và sau đó có thể gọi khi cần.

□ Khác Stored Procedure

- Chấp nhận **nhiều kiểu dữ liệu** giá trị trả về thông qua tên hàm.
- **Không chấp nhận tham số output.**
- Khác về cách gọi thực hiện.

6. HÀM DO NGƯỜI DÙNG ĐỊNH NGHĨA



- Hàm gồm 2 loại:
 - Giá trị trả về là kiểu dữ liệu cơ sở (int, varchar, float, datetime, ...) → **Scalar Value Function.**
 - Giá trị trả về là **Table** có được từ một câu truy vấn → **Table value Function.**

6. HÀM DO NGƯỜI DÙNG ĐỊNH NGHĨA



The screenshot displays the SQL Server Enterprise Manager interface. On the left, the 'Programmability' folder is expanded, and the 'Functions' folder is circled in red. Under 'Functions', the 'Scalar-valued Functions' folder is expanded, showing the newly created function 'dbo.XEM_TEN'. The right pane shows the SQL script used to create the function:

```
CREATE FUNCTION XEM_TEN()  
RETURNS VARCHAR(20)  
AS  
BEGIN  
    DECLARE @Hoten varchar(20)  
    SET @Hoten=(Select HOTEN  
                From SINHVIEN  
                Where MASV='3001100218')  
    RETURN @Hoten  
END
```

At the bottom, the 'Messages' pane shows the status: 'Command(s) completed successfully.'

6. HÀM DO NGƯỜI DÙNG ĐỊNH NGHĨA



ĐỊNH NGHĨA HÀM (hàm vô hướng – Scalar Value Function – Trả về duy nhất một giá trị)

CREATE FUNCTION <ten_ham> ([ds_tham_so])

RETURNS <kieu_du_lieu_tra_ve>

AS

BEGIN

<cau_lenh>

END

Dù thân function chỉ có 1 lệnh cũng phải đặt giữa Begin và End

6. HÀM DO NGƯỜI DÙNG ĐỊNH NGHĨA



Ví dụ 3.13: Tìm số lớn nhất trong 3 số a, b, c

```
Create function UF_SoLonNhat (@a int, @b int, @c int)
Returns int
As
Begin
    Declare @max int
    Set @max = @a
    If @b > max set @max = @b
    If @c > max set @max = @c
    Return @max
End
```


6. HÀM DO NGƯỜI DÙNG ĐỊNH NGHĨA



Ví dụ 3.14: Viết hàm nhận vào giá trị một ngày và trả về một chuỗi có giá trị là thứ của ngày đó trong tuần.

6. HÀM DO NGƯỜI DÙNG ĐỊNH NGHĨA



```
CREATE FUNCTION thu (@ngay DATE)
RETURNS nvarchar(10)
AS
BEGIN
    DECLARE @st nvarchar(10)
    SET @st = CASE DATEPART(dw, @ngay)
                WHEN 1 THEN 'Chu nhật'
                WHEN 2 THEN 'Thu hai'
                ...
    END
    RETURN @st
END
```

6. HÀM DO NGƯỜI DÙNG ĐỊNH NGHĨA



GỌI HÀM

```
DECLARE @st NVARCHAR(10)
```

```
SET @st = dbo.thu (GETDATE())
```

```
PRINT @st
```

6. HÀM DO NGƯỜI DÙNG ĐỊNH NGHĨA



HÀM với giá trị trả về là dữ liệu kiểu **BẢNG**

Cú pháp 1:

```
CREATE FUNCTION <ten_ham>([ds_tham_so])
```

```
RETURNS TABLE
```

```
AS
```

```
    RETURN    <cau_lenh_SELECT>
```

6. HÀM DO NGƯỜI DÙNG ĐỊNH NGHĨA



HÀM với giá trị trả về là dữ liệu kiểu **BẢNG**

Lưu ý:

- Kiểu trả về của hàm 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** (không sử dụng bất kỳ câu lệnh nào khác, không đặt trong cặp **Begin .. End**).

6. HÀM DO NGƯỜI DÙNG ĐỊNH NGHĨA



GỌI HÀM với giá trị trả về là dữ liệu kiểu **BẢNG**

```
SELECT ten_cot FROM dbo.Ten_ham(doi_so)
```

6. HÀM DO NGƯỜI DÙNG ĐỊNH NGHĨA



Ví dụ:

```
CREATE FUNCTION NOITUYEN (@TUOI INT)
```

```
RETURNS TABLE
```

```
AS
```

```
    RETURN (SELECT * FROM SINHVIEN
```

```
            WHERE TUOI > @TUOI)
```

```
-- Gọi hàm
```

```
    SELECT * FROM dbo.NOITUYEN(23)
```

6. HÀM DO NGƯỜI DÙNG ĐỊNH NGHĨA



HÀM với giá trị trả về là dữ liệu kiểu **BẢNG**

Khi cần sử dụng nhiều câu lệnh (khác câu lệnh **SELECT**) trong phần thân hàm, sử dụng cú pháp 2 như sau:

6. HÀM DO NGƯỜI DÙNG ĐỊNH NGHĨA



Cú pháp 2:

```
CREATE FUNCTION <ten_ham> ([ds_tham_so])  
RETURNS @Ten_bang_tra_ve TABLE  
        (Ten_cot Kieu_du_lieu[,...])  
AS  
BEGIN  
    <Cau_lenh>  
    Insert into @Ten_bang_tra_ve  
    RETURN  
END
```

6. HÀM DO NGƯỜI DÙNG ĐỊNH NGHĨA



Lưu ý:

- Cấu trúc bảng trả về bởi hàm được xác định dựa vào định nghĩa của bảng trong mệnh đề **RETURNS**.
- Biến **@Ten_bang_tra_ve** trong mệnh đề RETURNS có phạm vi sử dụng trong hàm và được sử dụng như một tên bảng.
- Câu lệnh **RETURN** trong thân hàm không chỉ định giá trị trả về. Giá trị trả về của hàm chính là các dòng dữ liệu trong bảng có tên là **@Ten_bang_tra_ve**

Ví dụ:



```
CREATE FUNCTION Func_TongSV (@lop SMALLINT)
RETURNS @bangthongke TABLE (
    Malop NVARCHAR(5),
    Tenlop NVARCHAR(50),
    Tongsosv INT
)

AS
BEGIN
    IF @lop = 0
        INSERT INTO @bangthongke
        SELECT lop.Malop, Tenlop, Count(Masv)
        FROM Lop, Sinhvien
        WHERE Sinhvien.Malop = Lop.Malop
        GROUP BY lop.Malop, Tenlop
```

Ví dụ:

```
CREATE FUNCTION Func_TongSV(@lop SMALLINT)
RETURNS @bangthongke TABLE (
    Malop NVARCHAR(5),
    Tenlop NVARCHAR(50),
    Tongso sv INT
)

AS
BEGIN
    ELSE
        INSERT INTO @bangthongke
        SELECT lop.Malop, Tenlop, Count(Masv)
        FROM Lop, Sinhvien
        WHERE Sinhvien.Malop = Lop.Malop and
               lop.malop = @lop
        GROUP BY lop.Malop, Tenlop

    RETURN
END
```

Ví dụ:

Câu lệnh:

```
SELECT * FROM dbo.func_TongSV(2)
```

→ Kết quả: thống kê tổng số sinh viên lớp có mã là 2.

Còn câu lệnh:

```
SELECT * FROM dbo.func_TongSV(0)
```

→ Cho biết tổng số sinh viên tất cả các lớp.

BÀI TẬP 3.4



Cho CSDL:

SinhVien(MaSV, HoTen, NgaySinh)

DiemThi(MaSV, MaMH, Diem)

1. Viết hàm trả về điểm trung bình của một sinh viên.
2. Viết hàm trả về mã SV có điểm trung bình cao nhất.
3. Viết hàm xuất danh sách các sinh viên có điểm trung bình bé hơn 5 (Thông tin gồm MASV, HOTEN)

LOP(MALOP, TENLOP)

SINHVIEN(MASV,HOTEN,TUOI, PHAI, MALOP)

MONHOC(MAMH,TENMH,SOTC)

KETQUA(MASV,MAMH, DIEM)

- 1/ Viết hàm truyền vào mã sinh viên, trả về tên lớp của sinh viên đó.
- 2/ Viết hàm truyền vào tham số mã sinh viên và mã môn học trả về điểm có sinh viên học môn đó.
- 3/ Viết hàm truyền vào mã sinh viên trả về tổng số tín chỉ tích lũy của sinh viên đó. Biết rằng chỉ tích lũy đối với môn học có điểm ≥ 5

4/ Viết hàm truyền vào tham số mã môn học, trả về bảng chứa thông tin những sinh viên học môn đó gồm (Mã sinh viên, họ tên, điểm, tên lớp)

5/ Viết hàm truyền vào mã lớp, trả về danh sách những sinh viên học lớp đó gồm: mã sinh viên, họ tên, điểm trung bình.

LOP(MALOP, TENLOP, SISO)

SINHVIEN(MASV, HOTEN, NGSINH, GIOITINH, QUEQUAN, MALOP, DIEMTB, XEPLOAI)

MONHOC(MAMH, TENMH, SOTC, BATBUOC)

KETQUA(MASV, MAMH, HOCKY, DIEMTHI)



- e) Viết hàm truyền vào tham số mã sinh viên và học kỳ, trả về tổng số tín chỉ đã đạt của sinh viên đó, biết rằng sinh viên đạt một môn nếu điểm thi ≥ 5 .
- f) Viết hàm truyền vào tham số mã lớp, trả về bảng chứa danh sách những sinh viên (mã sv, họ tên, ngày sinh) học lớp đó.
- g) Viết hàm truyền vào tham số mã môn học và học kỳ, trả về bảng chứa danh sách những sinh viên (mã sv, họ tên, ngày sinh, tên lớp) có điểm < 5 .
- h) Viết hàm truyền vào tham số mã môn học, trả về bảng chứa danh sách những sinh viên (mã sv, họ tên, ngày sinh) chưa học môn đó.
- i) Viết hàm truyền vào tham số mã sinh viên, trả về bảng chứa danh sách những môn học (mã môn, tên môn, điểm cao nhất) mà sinh viên học. Biết rằng trường hợp sinh viên học một môn học nhiều lần thì chỉ hiển thị số điểm cao nhất.

BÀI TẬP 3.4



1. Viết hàm tính điểm trung bình của một sinh viên.

```
CREATE FUNCTION FUNC_DTB (@MASV CHAR(10))  
RETURNS NUMERIC(5,1)  
AS  
    BEGIN  
        DECLARE @DTB NUMERIC(5,1)  
        SET @DTB = (SELECT AVG(DIEM)  
                    FROM DIEMTHI  
                    WHERE MASV = @MASV)  
        RETURN @DTB  
    END  
GO;
```

BÀI TẬP 3.4



1. Viết hàm tính điểm trung bình của một sinh viên.

```
--GỌI HÀM
```

```
DECLARE @DTB NUMERIC(5,1)
```

```
SET @DTB = dbo.FUNC_DT_B('SV03')
```

```
PRINT @DTB
```

```
GO;
```

BÀI TẬP 3.4



2. Viết hàm tìm mã của sinh viên có điểm trung bình cao nhất.

```
CREATE FUNCTION FUNC_TIMSV()  
RETURNS CHAR(10)  
AS  
    BEGIN  
        DECLARE @MASV CHAR(10)  
        SET @MASV = (SELECT MASV  
                     FROM DIEMTHI  
                     GROUP BY MASV  
                     HAVING AVG(DIEM) >= ALL(SELECT AVG(DIEM)  
                                             FROM DIEMTHI  
                                             GROUP BY MASV))  
        RETURN @MASV  
    END  
GO;
```

BÀI TẬP 3.4



2. Viết hàm tìm mã của sinh viên có điểm trung bình cao nhất.

```
--GỌI HÀM  
DECLARE @MASV CHAR(10)  
SET @MASV = dbo.FUNC_TIMSV()  
PRINT @MASV
```

BÀI TẬP 3.4



3. Viết hàm xuất danh sách các sinh viên có điểm trung bình bé hơn 5 (Thông tin gồm MASV, HOTEN)

```
CREATE FUNCTION FUNC_DSSVDUOITB()  
RETURNS TABLE  
AS  
    RETURN (SELECT SINHVIEN.MASV, HOTEN  
              FROM SINHVIEN, DIEMTHI  
              WHERE SINHVIEN.MASV = DIEMTHI.MASV  
              GROUP BY SINHVIEN.MASV, HOTEN  
              HAVING AVG(DIEM) <5)  
GO;
```

BÀI TẬP 3.4



3. Viết hàm xuất danh sách các sinh viên có điểm trung bình bé hơn 5 (Thông tin gồm MASV, HOTEN)

```
--GỌI HÀM  
SELECT * FROM DBO.FUNC_DSSVDUOITB()  
GO;
```

BÀI TẬP 3.4



3. Viết hàm xuất danh sách các sinh viên có điểm trung bình bé hơn 5 (Thông tin gồm MASV, HOTEN)

```
CREATE FUNCTION FUNC_DSSVDUOITB2 ()  
RETURNS @T_KQ TABLE (  
    MASV CHAR(10),  
    TENS NVARCHAR(50))  
  
AS  
  
    BEGIN  
        INSERT INTO @T_KQ  
            SELECT SINHVIEN.MASV, HOTEN  
            FROM SINHVIEN, DIEMTHI  
            WHERE SINHVIEN.MASV = DIEMTHI.MASV  
            GROUP BY SINHVIEN.MASV, HOTEN  
            HAVING AVG(DIEM) <5  
        RETURN  
    END  
  
GO;
```


