



CHƯƠNG 3

LẬP TRÌNH CƠ SỞ DỮ LIỆU BẰNG T-SQL

NỘI DUNG



1. KHAI BÁO VÀ SỬ DỤNG BIẾN

2. CÁC TOÁN TỬ

3. CÁC CẤU TRÚC ĐIỀU KHIỂN

4. CÁC HÀM THÔNG DỤNG

NỘI DUNG



5. THỦ TỤC THƯỜNG TRÚ (STORED PROCEDURE)

6. HÀM DO NGƯỜI DÙNG ĐỊNH NGHĨA

7. TRIGGER

8. KIỂU DỮ LIỆU CURSOR

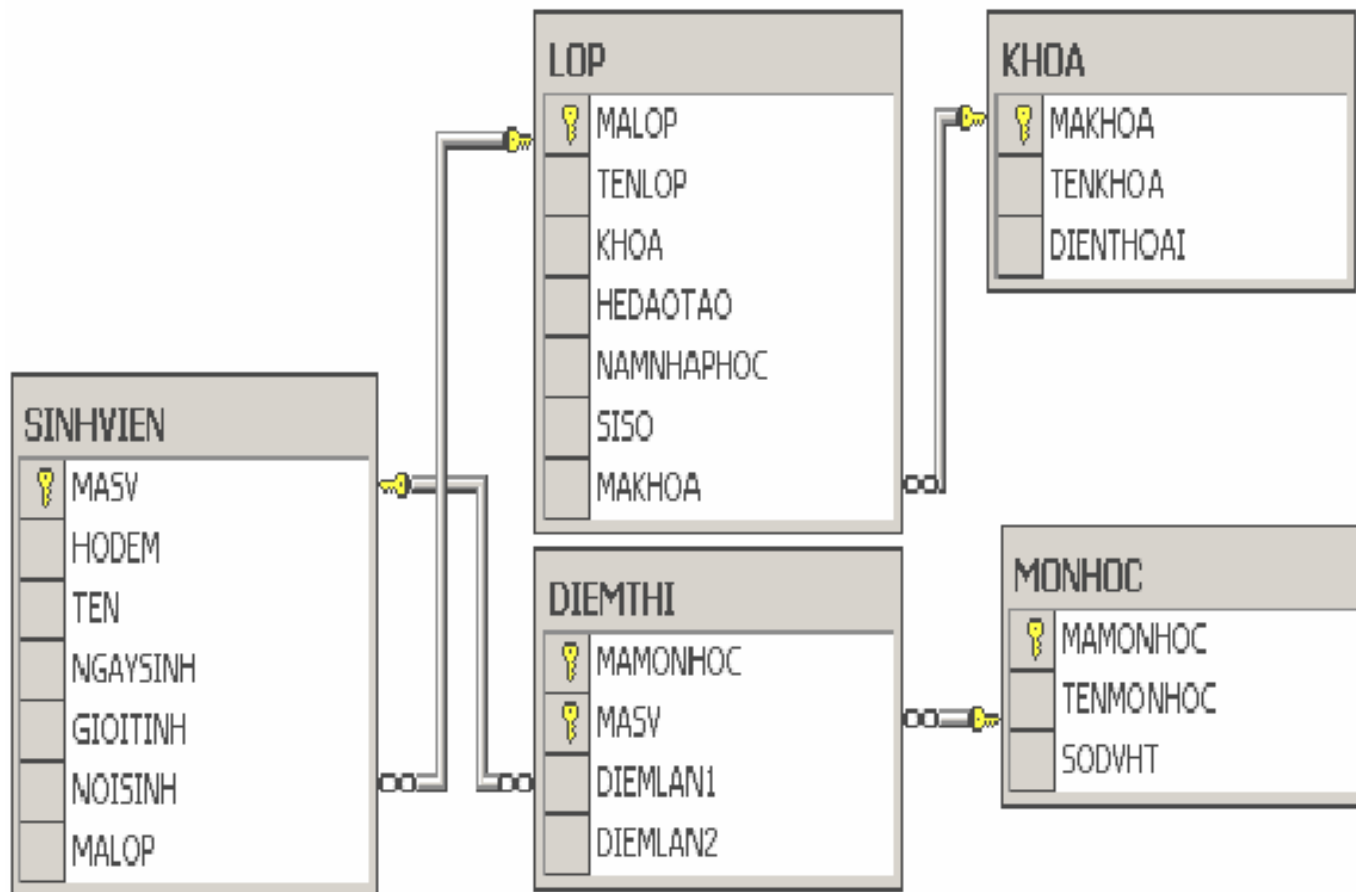


THỦ TỤC THƯỜNG TRÚ (STORE PROCEDURE – SPs)

5. STORED PROCEDURE – SPs



Cho CSDL quản lý điểm thi như sau:



5. STORED PROCEDURE – SPs



Giả sử cần thực hiện một chuỗi các thao tác trên cơ sở dữ liệu:

- 1. Bổ sung thêm môn học *cơ sở dữ liệu* có mã *CST005* và số đơn vị học trình là 5 vào bảng *MONHOC*
- 2. Lên danh sách nhập điểm thi môn *cơ sở dữ liệu* cho các sinh viên học lớp có mã *CDT002* (bổ sung thêm vào bảng *DIEMTHI* các bản ghi với cột *MAMONHOC* nhận giá trị *CST005*, cột *MASV* nhận giá trị lần lượt là mã các sinh viên học lớp có mã *CDT002* và các cột điểm là *NULL*).

5. STORED PROCEDURE – SPs

- Theo cách thông thường ta sẽ viết 2 lệnh như sau:

```
INSERT INTO monhoc
```

```
VALUES('CST005', N'Cơ sở dữ liệu',5)
```

- INSERT INTO diemthi(mamh,masv)

```
SELECT 'CST005', masv
```

```
FROM sinhvien
```

```
WHERE malop = 'CDT002'
```

Đây là mã môn học, đã cho trước nên ghi cụ thể ra luôn

5. STORED PROCEDURE – SPs



Thay vì viết 2 câu lệnh như trên, ta có thể định nghĩa 1 thủ tục với các tham số: @mamh, @tenmh, @sosvht, @malop để nhập dữ liệu cho một môn học bất kỳ do người dùng truyền vào khi gọi thủ tục.

5. STORED PROCEDURE – SPs



```
CREATE PROC sp_LenDanhSachDiem(  
    @mamh  NVARCHAR(10),  
    @tenmh  NVARCHAR(50),  
    @sodvht SMALLINT,  
    @malop  CHAR(4)  
  
AS  
BEGIN  
    INSERT INTO monhoc  
        VALUES(@mamh,@tenmh,@sodvht)  
    INSERT INTO diemthi(mamh,masv)  
        SELECT @mamh,masv  
        FROM sinhvien  
        WHERE malop=@malop  
END
```

5. STORED PROCEDURE – SPs



- Công cụ rất cần thiết cho các hệ quản trị cơ sở dữ liệu
- Là “Chương trình con” của SQL SERVER
- Chứa các lệnh T-SQL...
- Có thể gọi thủ tục nội tại ở trong SQL hay trong các ứng dụng được xây dựng bằng VB.NET, C#...
- Có thể được tạo ra từ công cụ và câu lệnh.
- Có thể chứa những câu lệnh thực hiện dữ liệu (DML) hoặc những câu lệnh truy vấn dữ liệu (SELECT).

5. STORED PROCEDURE – SPs

Đặc tính và những thuận lợi của SPs:

- Chấp nhận những tham số vào và trả về những giá trị được chứa trong các tham số ra.
- Người dùng có thể chỉ tạo SPs một lần, lưu trữ trong database một lần nhưng trong chương trình có thể gọi nó với số lần bất kỳ.
- SPs cho phép thực thi nhanh hơn: một đoạn source code khá lớn thực thi lặp đi lặp lại thì SPs thực hiện sẽ nhanh hơn.
- ...

5. STORED PROCEDURE – SPs



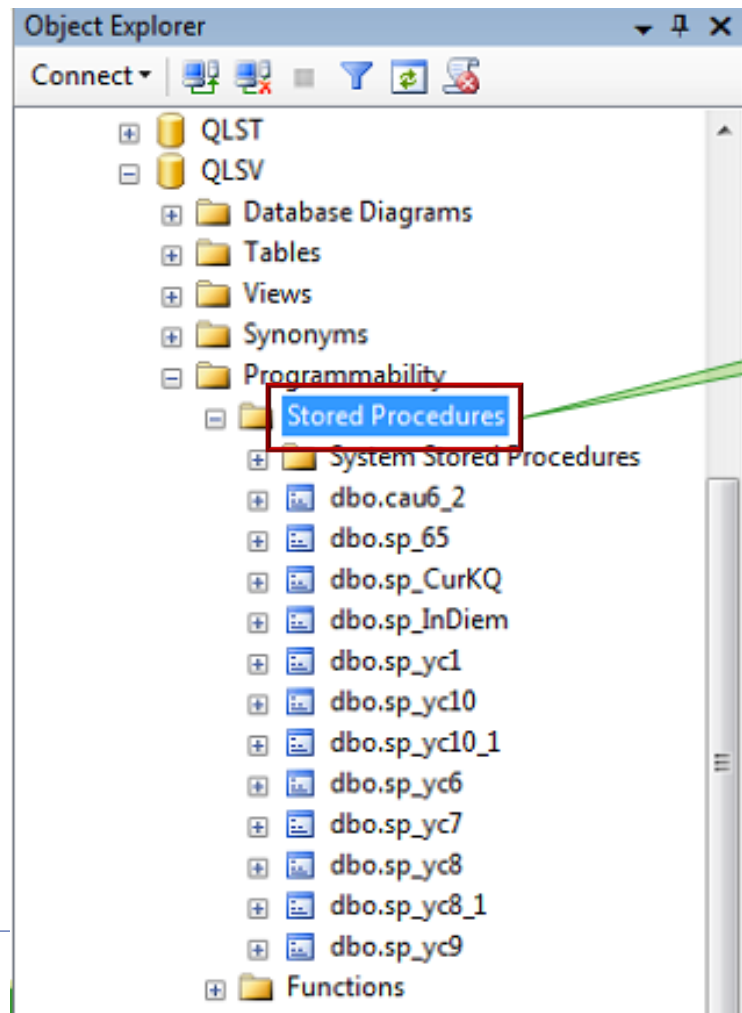
□ Có hai loại thủ tục lưu trữ:

- Thủ tục hệ thống – System Stored Procedures (thường bắt đầu bằng **sp_**).
- Thủ tục do người sử dụng tự viết gọi là **User Stored Procedures**.

5. STORED PROCEDURE – SPs



- Trong SQL Server, thủ tục được lưu trữ trong thư mục như sau:



Thư mục chứa thủ tục

5. STORED PROCEDURE – SPs



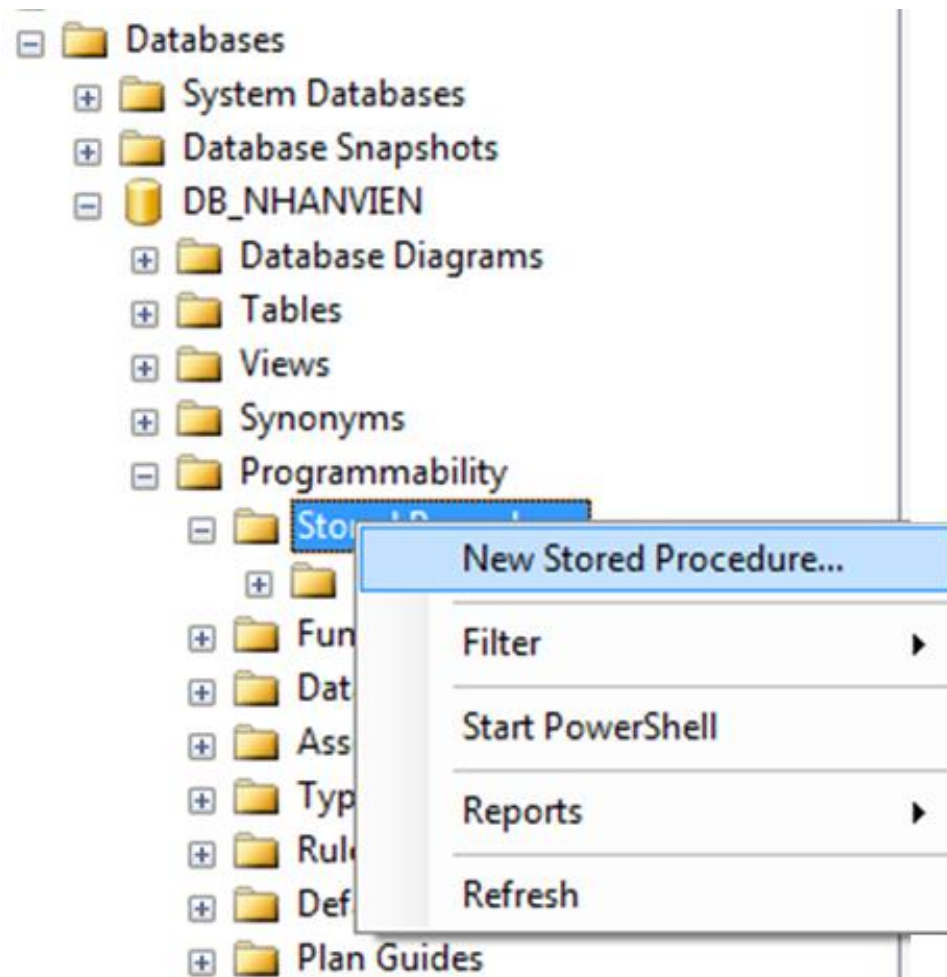
Một SPs được định nghĩa gồm những thành phần chính sau:

- ☐ Tên của SPs
- ☐ Các tham số
- ☐ Thân của SPs: bao gồm các câu lệnh T-SQL dùng để thực thi SPs

5. STORED PROCEDURE – SPs



❑ Tạo SPs bằng SQL Server Management Studio:

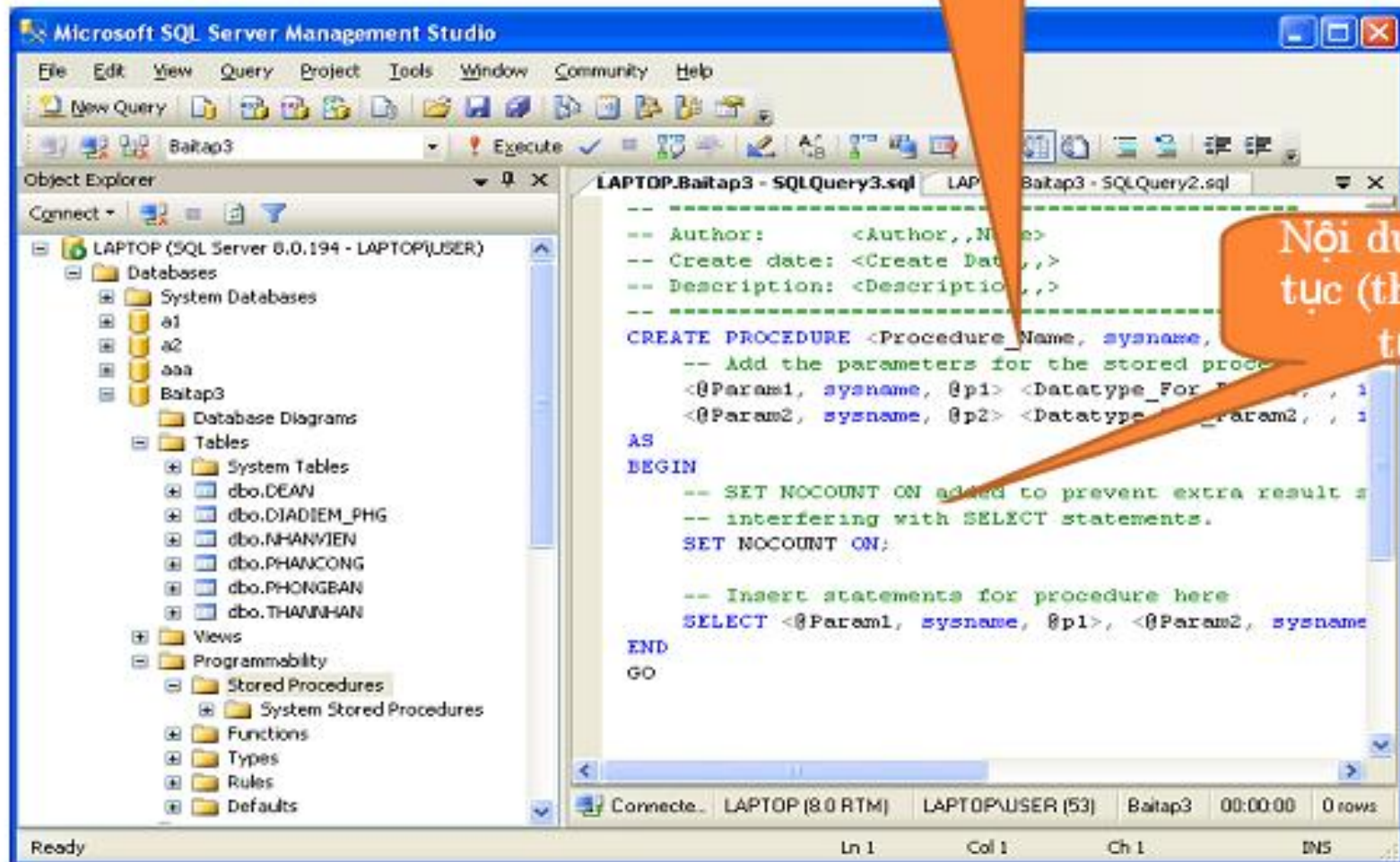


5. STORED PROCEDURE – SPs



Tên thủ tục

Nội dung thủ
tục (thân thủ
tục)



5. STORED PROCEDURE – SPs



TẠO THỦ TỤC BẰNG T-SQL

```
CREATE PROC [EDURE] Tên_thủ_tục  
[ (Danh_sách_tham_số) ]  
[ WITH RECOMPILE | ENCRYPTION | RECOMPILE, ENCRYPTION ]  
AS  
  
[Begin]  
  
    [Declare <biến cục bộ>]  
    <Các_câu_lệnh_của_thủ_tục>  
  
[End]
```

5. STORED PROCEDURE – SPs



TẠO THỦ TỤC BẰNG T-SQL

Trong đó:

- Tên_thủ_tục: Là tên thủ tục được tạo mới, phải là duy nhất trong một CSDL và đặt theo quy tắc đặt tên.
- Danh_sách_tham_số: Là các tham số dùng để nhận các giá trị bên ngoài đưa vào. Các tham số này được khai báo như sau: **@<tên_tham_số> <kiểu_dữ_liệu> [độ_dài]**.
- Số tham số tối đa trong một thủ tục là 255.

5. STORED PROCEDURE – SPs



TẠO THỦ TỤC BẰNG T-SQL

Trong đó :

- **WITH RECOMPILE:** SQL Server sẽ biên dịch lại thủ tục lưu trữ mỗi khi được gọi.
- **WITH ENCRYPTION:** SQL Server sẽ mã hóa thủ tục lưu trữ. Khi thủ tục đã được mã hóa, không thể xem được nội dung của thủ tục.

5. STORED PROCEDURE – SPs



TẠO THỦ TỤC BẰNG T-SQL

Lưu ý:

- Trong SQL Server, có thể ghi tắt **một số** từ khóa mà tên có chiều dài hơn 4 ký tự. Ví dụ: Create Proc
- Tên SPs, tên biến trong SQL Server không phân biệt chữ hoa chữ thường.

5. STORED PROCEDURE – SPs



TẠO THỦ TỤC BẰNG T-SQL

Ví dụ 1:

```
CREATE PROCEDURE XINCHAO
```

```
AS
```

```
Print N'Xin chào bạn đến với Stored Procedure'
```

5. STORED PROCEDURE – SPs



TẠO THỦ TỤC BẰNG T-SQL

Ví dụ 2: Tạo một thủ tục in ra danh sách những môn học (MAMH, TENMH) có số tiết nhiều hơn 45.

```
CREATE PROCEDURE ds_mh
```

```
AS
```

```
    SELECT  MAMH, TENMH
```

```
    FROM    MONHOC
```

```
    WHERE   SOTIET > 45
```

5. STORED PROCEDURE – SPs



TẠO THỦ TỤC BẰNG T-SQL

Lưu ý:

- CREATE PROCEDURE không chứa những câu lệnh sau: CREATE VIEW, CREATE TRIGGER, CREATE DEFAULT, CREATE PROCEDURE, CREATE RULE.
- Số biến hệ thống và số biến cục bộ chỉ bị giới hạn bởi khả năng bộ nhớ.

5. STORED PROCEDURE – SPs



TẠO THỦ TỤC BẰNG T-SQL

Lưu ý:

- Có thể có 2100 biến trong Stored Procedure.
- Kích thước tối đa cho Stored Procedure là 128 MB.

5. STORED PROCEDURE – SPs



BIÊN DỊCH VÀ GỌI THỰC THI THỦ TỤC

Biên dịch: Chọn toàn bộ mã lệnh tạo Stored Procedure

→ Nhấn **F5**

Thực thi thủ tục (gọi thủ tục):

Exec[ute] <tên_thủ_tục>

Ví dụ:

`Exec ds_mh --Gọi thực thi thủ tục tên ds_mh`

5. STORED PROCEDURE – SPs



HỦY BỎ THỦ TỤC

Cú pháp:

Drop Proc[edure] <tên thủ tục>

Ví dụ:

Drop Proc ds_mh --Hủy thủ tục tên ds_mh

5. STORED PROCEDURE – SPs



THAY ĐỔI NỘI DUNG THỦ TỤC

Cú pháp:

```
Alter Proc[edure] <tên thủ tục>
```

```
As
```

```
[Begin]
```

```
    [Declare <biến cục bộ>]
```

```
    <Các lệnh>
```

```
[End]
```

5. STORED PROCEDURE – SPs



THAM SỐ TRONG THỦ TỤC

- Một SPs có thể không có, có một hay nhiều tham số.
- Các tham số chỉ có nghĩa cục bộ trong SPs.
- Tên tham số duy nhất, nên đặt gọi nhớ.
- Một SPs cho phép tối đa 1024 tham số.

5. STORED PROCEDURE – SPs



THAM SỐ TRONG THỦ TỤC

Có **hai** loại tham số:

- Tham số đầu vào
- Tham số đầu ra

5. STORED PROCEDURE – SPs



THAM SỐ TRONG THỦ TỤC

Tham số đầu vào:

- Là tham số được dùng để nhận giá trị từ người dùng truyền vào cho thủ tục.

5. STORED PROCEDURE – SPs



THAM SỐ TRONG THỦ TỤC - Tham số đầu vào:

Cú pháp:

```
Create Procedure <tên thủ tục>  
    @<tên tham số đầu vào> <kdliệu> [độ dài] [, ...]  
As  
    [Declare <biến cục bộ>]  
    <Các lệnh>
```

5. STORED PROCEDURE – SPs



THAM SỐ TRONG THỦ TỤC - Tham số đầu vào:

Ví dụ 3.4: Tạo thủ tục **TIMSV** in ra thông tin của sinh viên có MaSV do người dùng truyền vào.

```
CREATE PROC TIMSV @MASV CHAR(10)
```

```
AS
```

```
SELECT *
```

```
FROM SINHVIEN
```

```
WHERE MASV = @MASV
```


5. STORED PROCEDURE – SPs



THAM SỐ TRONG THỦ TỤC - Tham số đầu vào:

Ví dụ 3.4: Tạo thủ tục **TIMSV** in ra thông tin của sinh viên có MaSV do người dùng truyền vào

Gọi thực hiện thủ tục:

```
Exec TIMSV 'SV01'
```

5. STORED PROCEDURE – SPs



THAM SỐ TRONG THỦ TỤC - Tham số đầu vào:

Ví dụ 3.5: Tạo thủ tục INBANGDIEM in ra bảng điểm (MAMH, TENMH, DIEM) của sinh viên có MaSV do người dùng truyền vào.

```
CREATE PROC INBANGDIEM @MASV CHAR(10)
AS
    SELECT mh.MAMH, TENMH, DIEM
    FROM KETQUA kq, MONHOC mh
    WHERE kq.MAMH = mh.MAMH AND MASV = @MASV
```

5. STORED PROCEDURE – SPs



THAM SỐ TRONG THỦ TỤC - Tham số đầu vào:

Ví dụ 3.6: Tạo thủ tục THEMMONHOC để thêm mới một môn học.

```
CREATE PROC THEMMONHOC @MAMH CHAR(10),  
    @TENMH NVARCHAR(30), @SOTIET INT  
  
AS  
  
    INSERT INTO MONHOC  
    VALUES (@MAMH, @TENMH, @SOTIET)
```

```
EXEC THEMMONHOC 'M13', 'HE QTCSDL', 30
```

5. STORED PROCEDURE – SPs



THAM SỐ TRONG THỦ TỤC

Tham số đầu ra:

- Nhận kết quả trả về từ thủ tục (giữ lại giá trị của đối số sau khi kết thúc thủ tục) và hiển thị cho người dùng.
- Được sử dụng kết hợp với từ khóa **OUTPUT** hay **OUT**.

5. STORED PROCEDURE – SPs



THAM SỐ TRONG THỦ TỤC - Tham số đầu ra:

Cú pháp:

```
Create Procedure <tên thủ tục>  
                @<tên tham số> <kdliệu> OUTPUT [...]
```

As

```
[Declare <biến cục bộ>]  
<Các lệnh>
```

5. STORED PROCEDURE – SPs



THAM SỐ TRONG THỦ TỤC - Tham số đầu ra:

Lưu ý:

- Khi gọi thủ tục có chứa tham số đầu ra phải chỉ tường minh tham số đầu ra với từ khóa **OUTPUT** hay **OUT**.

5. STORED PROCEDURE – SPs



THAM SỐ TRONG THỦ TỤC - Tham số đầu ra:

Ví dụ 3.7: Tạo thủ tục trả về **điểm một môn học** của một sinh viên.

- Tham số đầu vào: Mã sinh viên, Mã môn học.
- Tham số đầu ra: **Điểm của môn học.**

5. STORED PROCEDURE – SPs



THAM SỐ TRONG THỦ TỤC - Tham số đầu ra:

Ví dụ 3.7: Tạo thủ tục cho biết điểm một môn học của một sinh viên.

```
CREATE PROC XEMDIEM @MASV CHAR(10),  
                @MAMH CHAR(10), @DIEM INT OUTPUT  
AS  
SET @DIEM = (SELECT DIEM FROM KETQUA  
                WHERE MASV = @MASV  
                AND MAMH = @MAMH)
```


5. STORED PROCEDURE – SPs



THAM SỐ TRONG THỦ TỤC - Tham số đầu ra:

Ví dụ 3.7: Tạo thủ tục cho biết điểm một môn học của một sinh viên.

```
CREATE PROC XEMDIEM @MASV CHAR(10),  
                @MAMH CHAR(10), @DIEM INT OUTPUT  
AS  
SELECT @DIEM = DIEM FROM KETQUA  
WHERE MASV = @MASV AND MAMH = @MAMH
```

5. STORED PROCEDURE – SPs



THAM SỐ TRONG THỦ TỤC - Tham số đầu ra:

Ví dụ 3.7: Tạo thủ tục cho biết **điểm một môn học** của một sinh viên.

```
-- Gọi thực thi thủ tục
```

```
DECLARE @DIEM_SV INT
```

```
EXEC XEMDIEM 'SV01', 'M001', @DIEM_SV OUTPUT
```

```
PRINT @DIEM_SV
```

5. STORED PROCEDURE – SPs



THAM SỐ TRONG THỦ TỤC - Tham số đầu ra:

Ví dụ 3.8: Tạo thủ tục nhập vào MASV, xuất ra điểm TB của sinh viên đó và xếp loại SV theo DTB như sau:

- $DTB < 4$: Yếu
- $4 \leq DTB < 6.5$: Trung bình
- $6.5 \leq DTB < 8.5$: Khá
- $DTB \geq 8.5$: Giỏi

5. STORED PROCEDURE – SPs



```
CREATE PROC XEP_LOAI @MASV CHAR(10), @DTB FLOAT
    OUTPUT, @XEPLOAI NVARCHAR(30) OUTPUT
AS
    SELECT @DTB = AVG(DIEM)
    FROM KETQUA
    WHERE MASV = @MASV
    if @DTB < 4
        Set @XEPLOAI = N'Yếu'
    else if @DTB < 6.5
        Set @XEPLOAI = N'TB'
    else if @DTB < 8.5
        Set @XEPLOAI = N'Khá'
    else
        Set @XEPLOAI = N'Giỏi'
```

5. STORED PROCEDURE – SPs



THAM SỐ TRONG THỦ TỤC

Ví dụ 3.8: Tạo thủ tục nhập vào MASV, xuất ra điểm TB của sinh viên đó và xếp loại SV theo DTB.

```
-- Gọi thực thi thủ tục  
declare @dtbsv float, @xl nvarchar(30)  
exec xep_loai 'sv05', @dtbsv output, @xl output  
print @dtbsv  
print @xl
```

LOP(MALOP, TENLOP)

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

MONHOC(MAMH,TENMH,SOTC)

KETQUA(MASV,MAMH, DIEM)

- a/ Viết thủ tục truyền vào tham số mã sinh viên, trả về tên lớp của sinh viên đó
- b/ Viết thủ tục truyền vào mã lớp, trả về tổng số sinh viên có trong lớp đó
- c/ Viết thủ tục truyền vào mã sinh viên trả về họ tên và tổng số tín chỉ mà sinh viên đó đã học

LOP(MALOP, TENLOP)

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

MONHOC(MAMH,TENMH,SOTC)

KETQUA(MASV,MAMH, DIEM)

b/ Viết thủ tục truyền vào mã lớp, trả về tổng số sinh viên có trong lớp đó

c/ Viết thủ tục truyền vào mã sinh viên trả về họ tên và tổng số tín chỉ mà sinh viên đó đã học

LOP(MALOP, TENLOP)

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

MONHOC(MAMH,TENMH,SOTC)

KETQUA(MASV,MAMH, DIEM)

c/ Viết thủ tục truyền vào mã sinh viên trả về họ tên và tổng số tín chỉ mà sinh viên đó đã học

d/ Viết thủ tục truyền vào mã môn học, in ra danh danh sách những sinh viên có điểm ≥ 5

e/ Viết thủ tục truyền vào mã sinh viên và mã môn học, trả về *Đạt* nếu sinh viên có điểm ≥ 5 , ngược lại trả về *Không đạt*. Trường hợp sinh viên không học môn đó thì trả về *Chưa học*

5. STORED PROCEDURE – SPs

Ví dụ: (Gọi thủ tục)

--1. Truyền trị

Exec usp_ThemDangKy '001', 'HP01'

--2. Truyền trị có tên biến

Exec usp_ThemDangKy @MaHP = 'HP01', @MaSV = '001'

--3. Truyền trị có tên biến

Exec usp_ThemDangKy @MaHP, @MaSV

--4. Có output

Declare @SiSo int

Exec usp_ThemDangKy '001', 'HP01', @SiSo **output**

5. STORED PROCEDURE – SPs



MỘT SỐ VẤN ĐỀ KHÁC TRONG SPs

- Mã hóa nội dung thủ tục
- Biên dịch lại thủ tục
- Thủ tục lồng nhau
- Lệnh **return** trong SPs

5. STORED PROCEDURE – SPs

MÃ HÓA NỘI DUNG TRONG SPs

Mục đích: Không cho phép người dùng khác xem mã lệnh trong SPs.

```
Create Procedure <tên_thủ_tục>  
    @<tên_tham_số> <kdliệu> [Output] [...]
```

With Encryption

As

```
[Declare <biến_cục_bộ>]  
    <Các_lệnh>
```

5. STORED PROCEDURE – SPs

BIÊN DỊCH LẠI THỦ TỤC TRONG SPs

Mục đích: Mỗi lần có người dùng gọi thủ tục thì bản thân nó sẽ biên dịch lại.

```
Create Proc[edure] <tên_thủ_tục>  
    @<tên_tham_số> <kdliệu> [Output] [...]
```

With Recompile

As

```
[Declare <biến_cục_bộ>]  
    <Các lệnh>
```

Cách 1

5. STORED PROCEDURE – SPs



BIÊN DỊCH LẠI THỦ TỤC TRONG SPs

Mục đích: Mỗi lần có người dùng gọi thủ tục thì bản thân nó sẽ biên dịch lại.

EXEC <tên_thủ_tục> [<các tham số>] **With Recompile**

Lưu ý: Cách này dùng với việc tạo thủ tục không dùng tùy chọn With Recompile.

Cách 2

5. STORED PROCEDURE – SPs



THỦ TỤC LỒNG NHAU

- SQL cho phép các thủ tục lồng vào nhau (gọi lẫn nhau).
- SQL cho phép lồng tối đa 32 cấp.

5. STORED PROCEDURE – SPs



THỬ TỤC LÒNG NHAU

```
Create proc A  
AS  
Begin  
    -- Các lệnh  
End
```

```
Create proc B  
AS  
Begin  
    EXEC A  
    -- Các lệnh  
End
```

5. STORED PROCEDURE – SPs

LỆNH RETURN TRONG SPs

Cú pháp

RETURN [**⟨số nguyên⟩**]

Khi đó cách thức gọi thủ tục

EXEC @⟨biến⟩ = ⟨Tên_thủ_tục⟩[⟨các tham số⟩] [...]

Lưu ý: Khi gặp câu lệnh **RETURN** ngay lập tức **SPs** sẽ kết thúc.

LỆNH RETURN TRONG SPs

--Ví dụ 3.9: Tao SPs nhập vào ten mon hoc, tra ve 1 neu co sinh vien hoc mon hoc do, nguoc lai tra ve 0.

```
Create Proc Ktra_SV_MH @tenmh varchar(30) AS
    If exists (select *
               from   KetQua kq, MonHoc mh
               where  kq.MaMH = mh.MaMH And TenMH =
@tenmh)
    Begin
        return 1
    End
Else return 0
```

LỆNH RETURN TRONG SPs

--Lời gọi thủ tục

```
declare @kq int, @tenmh varchar(30)
set @tenmh = 'Triet hoc'
Exec @kq = Ktra_SV_MH @tenmh
if @kq = 0
print 'Khong ton tai sinh vien nao hoc mon ' +
@tenmh
```

LỆNH RETURN TRONG SPs

--Ví dụ 3.10: Tạo thủ tục trả về tuổi của một sinh viên – C1

```
CREATE PROCEDURE tinhhtuoi @masv char(10),  
                           @tuoi int OUTPUT  
  
AS  
  
    SET @tuoi = (SELECT datediff(yy, ngsinh, getdate())  
                 FROM sinhvien WHERE masv = @masv)
```

-----Gọi thực hiện -----

```
DECLARE @tuoisv int  
EXEC tinhhtuoi 'sv05', @tuoisv OUTPUT  
PRINT @tuoisv
```

LỆNH RETURN TRONG SPs

--Ví dụ 3.10: Tạo thủ tục trả về tuổi của một sinh viên – C2

```
CREATE PROCEDURE kttuoi @masv CHAR(10)
AS
    DECLARE @tuoi INT
    SET @tuoi = (SELECT datediff(yy, ngsinh, getdate())
                FROM sinhvien WHERE masv = @masv)
    RETURN @tuoi
```

-----Gọi thực hiện

```
DECLARE @tuoisv INT
EXEC @tuoisv = kttuoi 'sv05'
PRINT @tuoisv
```

LỆNH RETURN TRONG SPs

--Ví dụ 3.11: Tạo thủ tục KTTUOI của một SV, nếu tuổi lớn hơn 30 return 1, ngược lại return 0.

```
Create Procedure kttuoi @masv char(10)
as
    declare @tuoi int
    set @tuoi = (select datediff(yy,ngsinh,getdate())
                from sinhvien where masv = @masv)
    if @tuoi > 30
        return 1
    else
        return 0
    print 'da kiem tra xong'
```

LỆNH RETURN TRONG SPs

--Ví dụ 3.11: Tạo thủ tục KTTUOI của một SV, nếu tuổi lớn hơn 30 return 1, ngược lại return 0.

```
-----  
DECLARE    @k1    INT  
EXEC @k1 = kttuoi    'SV09'  
PRINT @k1
```

5. STORED PROCEDURE – SPs



SỬ DỤNG GIÁ TRỊ MẶC ĐỊNH CHO THAM SỐ

- Trong lệnh gọi thủ tục có tham số đầu vào, nếu không truyền giá trị cho tham số tương ứng thì hệ thống sẽ báo lỗi.
- Đôi khi gọi thủ tục, chúng ta muốn bỏ qua giá trị truyền vào cho tham số. Để thực hiện điều này ta sử dụng giá trị mặc định cho tham số theo cú pháp:

@<Tên tham số> <Kiểu dữ liệu> = <Giá trị mặc định>

5. STORED PROCEDURE – SPs



SỬ DỤNG GIÁ TRỊ MẶC ĐỊNH CHO THAM SỐ

```
CREATE PROC TT_SV @MASV CHAR(10)='SV01'  
AS  
    SELECT *  
    FROM SINHVIEN  
    WHERE MASV = @MASV  
GO
```

Khi gọi thực hiện thủ tục mà không cung cấp mã sinh viên thì giá trị mặc định sẽ được sử dụng là ‘SV01’

5. STORED PROCEDURE – SPs



SỬ DỤNG GIÁ TRỊ MẶC ĐỊNH CHO THAM SỐ

Ví dụ 3.12: Tạo thủ tục truyền vào mã lớp sẽ trả về tổng số sinh viên trong lớp đó. Trường hợp không truyền tham số thì trả về giá trị 0.

5. STORED PROCEDURE – SPs



SỬ DỤNG GIÁ TRỊ MẶC ĐỊNH CHO THAM SỐ

```
CREATE PROC TONGSV @MaLop char(10) = '0'
AS
    IF (@MaLop = '0')
        RETURN 0
    ELSE
        RETURN (SELECT COUNT(MASV)
                FROM SINHVIEN
                WHERE MALOP = @MALOP)
```

5. STORED PROCEDURE – SPs



SỬ DỤNG GIÁ TRỊ MẶC ĐỊNH CHO THAM SỐ

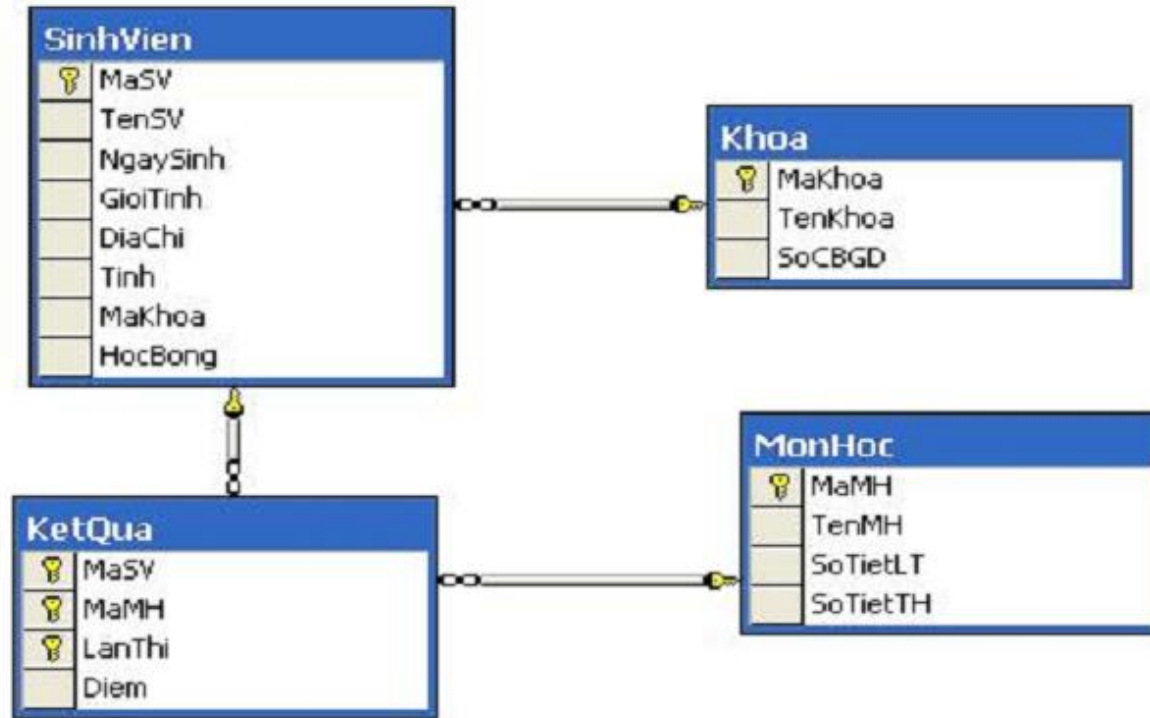
```
declare @SL int
exec @sl = TONGSV 'L001'
print @SL
```

```
declare @SL int
exec @SL = TONGSV
print @SL
```

BÀI TẬP 3.3



Xét các bảng dữ liệu có cấu trúc như sau:



Viết SPs nâng điểm của sinh viên có **MASV** là [input], **TenMH** [input], **LanThi** [input], **So_diem_duoc_nang** [input]

