
Chương 6

STORED PROCEDURE

Mục tiêu

- Sau khi học xong chương này, SV có thể:
 - Biết được Stored Procedure là gì?
 - Nắm vững cú pháp xây dựng Stored Procedure.
 - Biết cách gọi thực hiện Stored Procedure.
 - Xây dựng các loại Stored Procedure: hiển thị dữ liệu, tính toán và cập nhật dữ liệu.
 - Hiểu và tạo được các Type là kiểu dữ liệu do người dùng định nghĩa.

Nội dung chi tiết

- Stored Procedure là gì?
- Cú pháp xây dựng Stored Procedure
- Gọi thực hiện Stored Procedure
- Stored Procedure hiển thị dữ liệu
- Stored Procedure tính toán
- Stored Procedure cập nhật dữ liệu
- Kiểu dữ liệu do người dùng định nghĩa
- Bài tập ứng dụng

Stored Procedure là gì?

- Stored Procedure (thủ tục) là một tập hợp các câu lệnh SQL dùng để thực thi một nhiệm vụ nhất định. Nó hoạt động giống như một hàm trong các ngôn ngữ lập trình khác.
- Stored Procedure có thể có các tham số vào, tham số ra và có thể trả về kết quả.
- Có thể gọi Stored Procedure trực tiếp trong SQL hay trong các ứng dụng được xây dựng bằng C#, .NET...
- Các xử lý xây dựng trong Stored Procedure sẽ chạy nhanh hơn khi xây dựng ngoài ứng dụng, và tập trung tại server nên dễ quản lý.
- Theo mô hình lập trình client-server, tất cả các xử lý đều tập trung tại server => Trong SQL Server các xử lý này nằm trong Stored Procedure.

Cú pháp xây dựng Stored Procedure

**create proc Tên_Stored_Procedure [(Các_tham_số)]
as**

Khởi_lệnh_xử_ly_trong_Stored_Procedure

- Trong đó:
 - **Các_tham_số**: gồm 2 loại
 - **Tham số vào**: nhận giá trị từ người dùng truyền vào cho Stored Procedure xử lý.
 - Cú pháp:
@Tên_biến Kiểu_dữ_liệu [= Giá_trị_mặc_định]
 - **Tham số ra**: nhận kết quả trả về từ Stored Procedure và hiển thị cho người dùng.
 - Cú pháp:
@Tên_biến Kiểu_dữ_liệu output

Gọi thực hiện Stored Procedure

- Cú pháp gọi Stored Procedure không có giá trị trả về:

exec Tên_Stored_Procedure [**@Tên_tham_số_vào = Giá_trị** [,..] **@Tên_tham_số_ra = @Tên_biến output**]

- Lệnh **return** trong Stored Procedure:

- Trong Stored Procedure, có thể dùng lệnh **return** để trả về 1 giá trị hoặc để thoát khỏi Stored Procedure.
- Thoát khỏi Stored Procedure và trả về 0: **return**
- Thoát khỏi Stored Procedure và trả về giá trị: **return giá trị**

- Cú pháp gọi Stored Procedure có giá trị trả về:

exec @Tên_biến = [**@Tên_tham_số_vào = Giá_trị** [,..] **@Tên_tham_số_ra = @Tên_biến output**]

Stored Procedure hiển thị dữ liệu

- Loại Stored Procedure này có thể có tham số vào.
- Bên trong Stored Procedure chứa khối lệnh T-SQL:
 - Kiểm tra tham số vào có hợp lệ không? Nếu không thì thông báo lỗi.
 - Kiểm tra kết quả trả về có tồn tại không? Nếu có thì hiển thị kết quả, không thì thông báo không tìm thấy thông tin với tham số vào.

Stored Procedure hiển thị dữ liệu

- **VD:** Xây dựng thủ tục sp_DSSV_Thuoc_Khoa hiển thị thông tin các sinh viên với tham số vào là Tên khoa:

```
create proc sp_DSSV_Thuoc_Khoa(@tenkh nvarchar(30))
as
if not exists (select TENKH from KHOA where TENKH=@tenkh)
begin
    print N'Tên khoa không tồn tại'
    return
end
if (select count(*) from SINHVIEN inner join KHOA on SINHVIEN.MAKH=KHOA.MAKH where TENKH=@tenkh) > 0
select SINHVIEN.*,TENKH
from SINHVIEN join KHOA on SINHVIEN.MAKH=KHOA.MAKH
where TENKH=@tenkh
else
    print N'Không có SV thuộc Khoa ' + @tenkh
go

exec sp_DSSV_Thuoc_Khoa N'Tin Học'
```

100 %

Results Messages

	MASV	HOSV	TENSV	PHAI	NGAYSINH	NOISINH	MAKH	HOCBONG	TENKH
1	A02	Trần Văn	Chính	1	1982-12-24 00:00:00.000	TP.HCM	TH	100000	Tin học
2	B01	Trần Thanh	Mai	0	1981-12-20 00:00:00.000	Bến Tre	TH	200000	Tin học
3	B02	Trần Thị Thu	Thủy	0	1982-02-13 00:00:00.000	TP.HCM	TH	30000	Tin học
4	B03	Trần Thị	Thanh	0	1982-12-31 00:00:00.000	TP.HCM	TH	50000	Tin học

Stored Procedure tính toán

- Loại Stored Procedure này có thể có tham số vào, tham số ra, giá trị trả về.
- Bên trong Stored Procedure chứa câu lệnh T-SQL để kiểm tra, tính toán giá trị trả về.

■ VD:

```
--Xây dựng Stored Procedure tên sp_DemSVNu để đếm số lượng các sinh viên nữ.  
create proc sp_DemSVNu(@soSVNu int output)  
as  
select @soSVNu=count(*)  
from SINHVIEN  
where PHAI=0  
go  
  
declare @demNu int  
exec sp_DemSVNu @demNu output  
if @demNu>0  
    print N'Có ' + CAST(@demNu as varchar(10)) + N' SV nữ'  
else  
    print N'Không có SV nữ'  
go
```

100 % <

Messages

Có 5 SV nữ

Stored Procedure cập nhật dữ liệu

■ Thêm dữ liệu vào bảng:

- Loại Stored Procedure này có tham số vào là tên các cột có trong bảng cần thêm dữ liệu, trừ các cột có kiểu dữ liệu tự tăng (identity).
- Có thể có tham số ra, giá trị trả về: cho biết việc thêm dữ liệu có thành công hay không.
- Chứa khối lệnh T-SQL:
 - Kiểm tra ràng buộc dữ liệu duy nhất (Primary Key, Unique)
 - Kiểm tra ràng buộc khóa ngoại (Foreign Key)
 - Kiểm tra ràng buộc miền giá trị (Check)
 - Nếu vi phạm các ràng buộc thì không cho thêm dữ liệu mới
 - Ngược lại thì:
 - * Thêm dữ liệu mới vào bảng
 - * Cập nhật dữ liệu của các bảng liên quan (nếu có)
 - * Hiển thị thông báo đã thêm dữ liệu thành công

Stored Procedure cập nhật dữ liệu

■ Thêm dữ liệu vào bảng:

- Cú pháp kiểm tra ràng buộc dữ liệu duy nhất:

Khai báo và gán giá trị cho @biến_lỗi = 0

```
if exists (SELECT Tên_cột_duy_nhất FROM Tên_bảng WHERE  
Tên_cột_duy_nhất = @Giá_trị_cột_duy_nhất_muốn_thêm)
```

```
begin
```

```
    print N'Thông báo lỗi: vi phạm ràng buộc dữ liệu duy nhất'
```

```
    set @biến_lỗi = 1
```

```
end
```

Stored Procedure cập nhật dữ liệu

■ Thêm dữ liệu vào bảng:

- Cú pháp kiểm tra ràng buộc khóa ngoại:

```
if not exists (SELECT Tên_cột_khóa_ngoại FROM Tên_bảng WHERE  
Tên_cột_khóa_ngoại = @Giá_trị_cột_khóa_ngoại_muốn_thêm)
```

```
begin
```

```
    print N'Thông báo lỗi: vi phạm ràng buộc khóa ngoại'
```

```
    set @biến_lỗi = 1
```

```
end
```

Stored Procedure cập nhật dữ liệu

■ Thêm dữ liệu vào bảng:

- Cú pháp kiểm tra ràng buộc miền giá trị:

if @Tên_cột không thuộc miền giá trị

begin

print N'Thông báo lỗi: vi phạm ràng buộc miền giá trị'

set @biến_lỗi = 1

end

if @biến_lỗi = 1 return

Thêm dữ liệu vào bảng

Cập nhật dữ liệu của các bảng liên quan (nếu có)

print N'Thông báo đã thêm dữ liệu thành công'

Stored Procedure cập nhật dữ liệu

■ Thêm dữ liệu vào bảng:

- VD:

```
create proc sp_ThemMonHoc (@mamh char(2), @tenmh nvarchar(25), @sotiet int=30)
as
    declare @loi int = 0
    if exists (select MAMH from MONHOC where MAMH=@mamh)
    begin
        print N'Vi phạm ràng buộc dữ liệu duy nhất: Mã môn học đã có => Trùng khóa chính'
        set @loi = 1
    end
    if exists (select TENMH from MONHOC where TENMH=@tenmh)
    begin
        print N'Vi phạm ràng buộc dữ liệu duy nhất: Tên môn học đã có'
        set @loi = 1
    end
    if @sotiet < 30
    begin
        print N'Vi phạm ràng buộc miền giá trị: Số tiết phải >= 30'
        set @loi = 1
    end
    if @loi = 1 return
    insert into MONHOC values (@mamh, @tenmh, @sotiet)
    print N'Đã thêm môn học mới thành công'
go

exec sp_ThemMonHoc '07', N'Vật lý nguyên tử', 20
go
```

100 %

Messages

Vi phạm ràng buộc dữ liệu duy nhất: Mã môn học đã có => Trùng khóa chính
Vi phạm ràng buộc dữ liệu duy nhất: Tên môn học đã có
Vi phạm ràng buộc miền giá trị: Số tiết phải >= 30

Stored Procedure cập nhật dữ liệu

■ Thêm dữ liệu vào bảng:

- VD:

```
create proc sp_ThemMonHoc (@mamh char(2), @tenmh nvarchar(25), @sotiet int=30)
as
    declare @loi int = 0
    if exists (select MAMH from MONHOC where MAMH=@mamh)
    begin
        print N'Vi phạm ràng buộc dữ liệu duy nhất: Mã môn học đã có => Trùng khóa chính'
        set @loi = 1
    end
    if exists (select TENMH from MONHOC where TENMH=@tenmh)
    begin
        print N'Vi phạm ràng buộc dữ liệu duy nhất: Tên môn học đã có'
        set @loi = 1
    end
    if @sotiet < 30
    begin
        print N'Vi phạm ràng buộc miền giá trị: Số tiết phải >= 30'
        set @loi = 1
    end
    if @loi = 1 return
    insert into MONHOC values (@mamh, @tenmh, @sotiet)
    print N'Đã thêm môn học mới thành công'
go
```

```
exec sp_ThemMonHoc '08', N'Đồ họa ứng dụng'
```

```
go
```

100 %
Messages

(1 row(s) affected)

Đã thêm môn học mới thành công

```
select * from MONHOC
```

100 %

Results Messages

	MAMH	TENMH	SOTIET
1	01	Nhập môn máy tính	30
2	02	Trí tuệ nhân tạo	45
3	03	Truyền tin	45
4	04	Đồ họa	50
5	05	Văn phạm	40
6	06	Đàm thoại	30
7	07	Vật lý nguyên tử	30
8	08	Đồ Họa Ứng Dụng	30

Stored Procedure cập nhật dữ liệu

■ Xóa dữ liệu của bảng:

- Loại Stored Procedure này có tham số vào là các cột làm khóa chính trong bảng muốn xóa dữ liệu.
- Có thể có tham số ra, giá trị trả về: cho biết việc xóa dữ liệu có thành công hay không.
- Chứa câu lệnh T-SQL:
 - Kiểm tra tham số vào có hợp lệ không? Nếu không thì thông báo lỗi.
 - Kiểm tra dữ liệu xóa có tồn tại trong bảng nhiều (N) không?
 - Nếu có thì không cho xóa dữ liệu vì vi phạm dữ liệu liên quan.
 - Ngược lại thì:
 - * Xóa dữ liệu của bảng
 - * Cập nhật dữ liệu của các bảng liên quan (nếu có)
 - * Hiển thị thông báo đã xóa dữ liệu thành công

Stored Procedure cập nhật dữ liệu

■ Xóa dữ liệu của bảng:

- Nếu xóa dữ liệu bảng nhiều (N): không cần kiểm tra các ràng buộc.
- Nếu xóa dữ liệu bảng danh mục (1): phải kiểm tra dữ liệu xóa có tồn tại trong bảng nhiều (N) không? Nếu có thì không cho xóa.

- Cú pháp xóa dữ liệu của bảng:

Kiểm tra tham số vào có hợp lệ không? Nếu không thì thông báo lỗi.

```
if exists (select * from Tên_Bảng_Nhiều_N where  
Tên_Cột_Khóa_Ngoại = @Giá_trị_cột_khóa_chính_muốn_xóa)
```

```
begin
```

```
    print N'Thông báo lỗi: vi phạm dữ liệu liên quan bảng nhiều N'
```

```
    return
```

```
end
```

Xóa dữ liệu của bảng

Cập nhật dữ liệu của các bảng liên quan (nếu có)

```
print N'Thông báo đã xóa dữ liệu thành công'
```

Stored Procedure cập nhật dữ liệu

■ Xóa dữ liệu của bảng:

- VD:

```
create proc sp_XoaMonHoc(@mamh char(2))
as
if not exists (select MAMH from MONHOC where MAMH=@mamh)
begin
    print N'Mã môn học không tồn tại.'
    return
end
if exists(select MAMH from KETQUA where MAMH=@mamh)
begin
    print N'Môn học đã có sinh viên thi nên không xóa được.'
    return
end
delete MONHOC where MAMH=@mamh
print N'Đã xóa môn học thành công'
go

exec sp_XoaMonHoc '08'
go
```

100 % < Messages

(1 row(s) affected)
Đã xóa môn học thành công

```
select * from MONHOC
```

100 % <

Results Messages

	MAMH	TENMH	SOTIET
1	01	Nhập môn máy tính	30
2	02	Trí tuệ nhân tạo	45
3	03	Truyền tin	45
4	04	Đồ họa	50
5	05	Văn phạm	40
6	06	Đàm thoại	30
7	07	Vật lý nguyên tử	30

Stored Procedure cập nhật dữ liệu

■ Sửa dữ liệu của bảng:

- Loại Stored Procedure này có tham số vào là tên các cột có trong bảng cần sửa dữ liệu.
- Có thể có tham số ra, giá trị trả về: cho biết việc sửa dữ liệu có thành công hay không.
- Chứa câu lệnh T-SQL:
 - Kiểm tra dữ liệu sửa có tồn tại hay không? Nếu không thì thông báo lỗi.
 - Kiểm tra các ràng buộc: dữ liệu duy nhất, khóa ngoại và miền giá trị.
 - Sửa dữ liệu của bảng (không sửa cột làm khóa chính và cột có thuộc tính tự tăng identity).
 - Cập nhật dữ liệu của các bảng liên quan (nếu có).
 - Hiển thị thông báo đã sửa dữ liệu thành công.

Stored Procedure cập nhật dữ liệu

■ Sửa dữ liệu của bảng:

- VD:

```
create proc sp_SuaMonHoc(@mamh char(2),@tenmh nvarchar(25),@sotiet int=30)
as
if not exists(select MAMH from MONHOC where MAMH=@mamh)
begin
    print N'Mã môn học không tồn tại.'
    return
end
if exists(select TENMH from MONHOC where TENMH=@tenmh and MAMH<>@mamh)
begin
    print N'Trùng tên môn học đã có.'
    return
end
if @sotiet<30
begin
    print N'Số tiết phải >= 30.'
    return
end
update MONHOC
set TENMH=@tenmh,SOTIET=@sotiet
where MAMH=@mamh
print N'Đã sửa dữ liệu thành công'
go

exec sp_SuaMonHoc '07',N'Vật lý hạt nhân',50
go
```

100 %
Messages

(1 row(s) affected)
Đã sửa dữ liệu thành công

```
select * from MONHOC
```

121 %

Results Messages

	MAMH	TENMH	SOTIET
1	01	Nhập môn máy tính	30
2	02	Trí tuệ nhân tạo	45
3	03	Truyền tin	45
4	04	Đồ họa	50
5	05	Văn phạm	40
6	06	Đàm thoại	30
7	07	Vật lý hạt nhân	50

Kiểu dữ liệu do người dùng định nghĩa

■ Giới thiệu:

- SQL server đã định nghĩa sẵn các kiểu dữ liệu. Tuy nhiên, cũng cho phép người dùng tự định nghĩa kiểu dữ liệu riêng (**Type**).
- Khi đã tạo một **Type** thì có thể sử dụng nó trong CSDL như các kiểu dữ liệu mà SQL đã định nghĩa.

■ Phân loại:

1. **User-Defined Data Type**: kiểu dữ liệu đơn.
2. **User-Defined Tables Type**: kiểu dữ liệu bảng

Kiểu dữ liệu do người dùng định nghĩa

■ Định nghĩa và sử dụng:

- **User-Defined Data Type**: kiểu dữ liệu đơn, dùng khi có yêu cầu tạo một cột với kiểu dữ liệu tương tự nhau trong CSDL.

- **Cú pháp định nghĩa:**

CREATE TYPE Tên_Type FROM Kiểu_dữ_liệu

- **VD**: định nghĩa 4 User-Defined Data Type như sau:

```
CREATE TYPE Ten FROM nvarchar(50)
CREATE TYPE Ma FROM char(2)
CREATE TYPE Ngay FROM datetime
CREATE TYPE SoNguyen FROM int
```

và sử dụng 4 kiểu đã định nghĩa để tạo bảng hoặc khai báo biến:

```
create table KHOA (
    MAKH Ma primary key,
    TENKH Ten unique,
    NGAYTHANHLAP Ngay default getdate(),
    SOLUONGSV SoNguyen default 0 check(SOLUONGSV>=0)
)
```

```
declare @TenMon Ten = N'Hệ quản trị CSDL'
```

Kiểu dữ liệu do người dùng định nghĩa

■ Định nghĩa và sử dụng:

- **User Defined Table Types**: kiểu dữ liệu bảng, dùng khi có yêu cầu tạo một bảng với cấu trúc tương tự nhau trong CSDL.

- **Cú pháp định nghĩa:**

```
CREATE TYPE Tên_Type AS TABLE (  
    Tên_cột_thứ_1 Kiểu_dữ_liệu,  
    Tên_cột_thứ_2 Kiểu_dữ_liệu,  
    ...  
)
```

- **VD**: định nghĩa 1 User-Defined Table Type như sau:

```
CREATE TYPE Bang AS TABLE (  
    Ma char(2),  
    Ten nvarchar(50),  
    Ngay datetime,  
    SoNguyen int  
)
```

và sử dụng loại bảng trên để tạo bảng tạm thời:

```
declare @BangTam Bang
```

Bài tập ứng dụng

■ Sử dụng CSDL Quản lý sinh viên:

1. Xây dựng Stored Procedure tên `sp_KetQuaThi` với tham số vào là mã số SV (giá trị mặc định là NULL) để hiển thị thông tin: Mã SV, Họ và tên, Tên môn và Điểm. Nếu không truyền vào mã số SV thì thủ tục sẽ liệt kê kết quả thi của tất cả các sinh viên.
2. Xây dựng Stored Procedure tên `sp_TongHocBongSVTheoKhoa` với tham số vào là Tên khoa để tính tổng học bổng của các sinh viên thuộc khoa đó. Nếu Tên khoa không hợp lệ thì thông báo lỗi.
3. Xây dựng Stored Procedure tên `sp_DTB` để tính điểm trung bình với 2 tham số vào là mã môn học và mã khoa.

Bài tập ứng dụng

- **Sử dụng CSDL Quản lý sinh viên:**
 - 4. Xây dựng Stored Procedure tên `sp_HienThi_DSSV_TheoKhoa` với tham số vào là mã khoa để hiển thị thông tin sinh viên thuộc Khoa đó và có thêm cột GHI CHÚ hiển thị “Đã thi xxx môn” nếu SV có kết quả thi, ngược lại thì hiển thị “Chưa có kết quả thi” nếu SV chưa thi môn nào.
 - 5. Xây dựng Stored Procedure tên `sp_Them_SV` để thêm 1 SV mới, cần kiểm tra ràng buộc dữ liệu trước khi thực hiện lệnh thêm mới.
 - 6. Xây dựng Stored Procedure tên `sp_Xoa_SV` để xóa 1 SV với tham số vào là mã SV muốn xóa, cần kiểm tra ràng buộc dữ liệu trước khi thực hiện lệnh xóa.

Bài tập ứng dụng

- **Sử dụng CSDL Quản lý sinh viên:**
 - 7. Xây dựng Stored Procedure tên sp_Sua_SV để sửa thông tin 1 SV, cần kiểm tra ràng buộc dữ liệu trước khi thực hiện lệnh cập nhật.
 - 8. Xây dựng Stored Procedure tên sp_DTB_SV để tính điểm trung bình của sinh viên với tham số vào là mã sinh viên và tham số ra là điểm trung bình của sinh viên có mã sinh viên trùng với mã sinh viên truyền vào.
 - 9. Xây dựng Stored Procedure tên sp_SoMonDauRot để tính số môn đậu, số môn rớt của sinh viên với tham số vào là mã sinh viên, 2 tham số ra là số môn đậu và số môn rớt của sinh viên có mã sinh viên trùng với mã sinh viên truyền vào.

Câu hỏi trắc nghiệm

1. Chọn phát biểu SAI về Stored Procedure?

- a. Stored Procedure là một tập hợp các câu lệnh SQL dùng để thực thi một nhiệm vụ nhất định.
- b. Stored Procedure có thể có các tham số vào, tham số ra và có thể trả về kết quả.
- c. Cú pháp khai báo tham số ra của Stored Procedure là: @Tên_biến Kiểu_dữ_liệu output
- d. **Cú pháp tạo Stored Procedure là:**
create procedure Tên_Stored_Procedure [(Các_tham_số)]
as
Khối_lệnh_xử_lý_trong_Stored_Procedure

Câu hỏi trắc nghiệm

2. Stored Procedure bao gồm các loại nào?

- a. Hiển thị dữ liệu
- b. Tính toán
- c. Cập nhật dữ liệu
- d. **Tất cả đều đúng**

Câu hỏi trắc nghiệm

3. Chọn trình tự thực hiện ĐÚNG để thêm dữ liệu mới vào bảng:

1. Cập nhật dữ liệu của các bảng liên quan (nếu có).
2. Kiểm tra các ràng buộc dữ liệu: duy nhất, khóa ngoại, miền giá trị. Nếu vi phạm các ràng buộc thì thông báo lỗi và thoát khỏi Stored Procedure.
3. Hiển thị thông báo đã thêm dữ liệu thành công
4. Thêm dữ liệu mới vào bảng.

- a. 1, 2, 3, 4
- b. 4, 3, 2, 1
- c. 2, 4, 1, 3
- d. 2, 1, 4, 3

Câu hỏi trắc nghiệm

4. Đối với Stored Procedure cập nhật dữ liệu, không cho phép sửa thông tin cột làm khóa chính hoặc cột có thuộc tính tự tăng (identity).

- a. Đúng
- b. Sai

Câu hỏi trắc nghiệm

5. Kiểu dữ liệu do người dùng định nghĩa bao gồm mấy loại?

- a. 1
- b. 2
- c. 3
- d. 4