

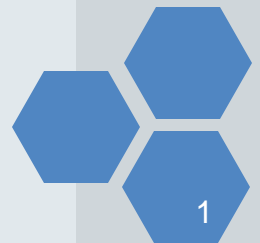
HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU – LẦN 1

STORE PROCEDURE– TRIGGER



Trình bày:

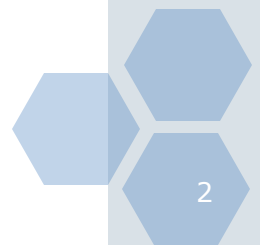
HỒ THỊ HOÀNG VY
VŨ THỊ MỸ HẰNG





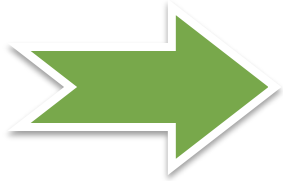
STORE PROCEDURE

TRIGGER





STORE PROCEDURE



1

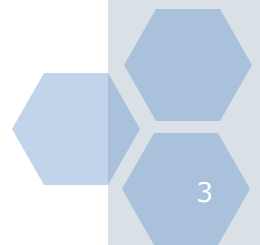
Giới thiệu sơ lược

2

Cú pháp tạo và thực thi procedure

3

Một số câu lệnh khác về procedure

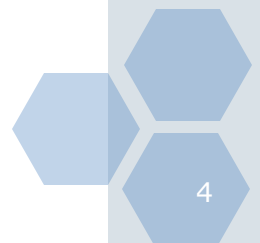




Giới thiệu sơ lược

❖ **Store procedure :**

- Thủ tục thường trú trong CSDL
- Tương tự như thủ tục trong các ngôn ngữ lập trình có thể truyền tham số, có tính tái sử dụng
- Gồm các câu lệnh T-SQL
- Được dịch và lưu trữ thành một đối tượng trong CSDL
- Được thực thi khi người dùng gọi đến





STORE PROCEDURE

1

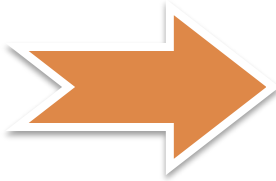
Giới thiệu sơ lược

2

Cú pháp tạo và thực thi procedure

3

Một số câu lệnh khác về procedure





Cú pháp tạo và thực thi procedure

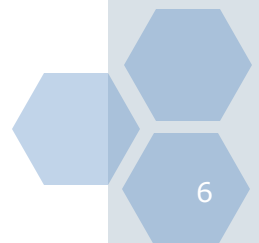
❖ Một stored procedure bao gồm:

- Tên stored proc
- Các tham số
- Các lệnh T-SQL dùng để thực thi stored proc

**Thay bằng
Create proc**

```
CREATE PROCEDURE sp_name  
    @parameters data_type input/output  
AS  
BEGIN  
    Sql_statement  
END
```

**Các tham số
vào và ra**





Cú pháp tạo và thực thi procedure

❖ Lệnh khai báo và gán biến

DECLARE @parameter_name data_type /*Gán giá trị cho biến*/

SET @parameter_name = value

SELECT @parameter_name = value

/*In thông báo ra màn hình*/

print N'Chuỗi thông báo unicode' --Ghi chú 1, một dòng

/* Ghi chú 2

Nhiều dòng

*/





Cú pháp tạo và thực thi procedure

❖ Ví dụ: stored procedure có tham số output

❖ --Tạo stored-procedure sp_tong

```
CREATE PROCEDURE sp_Tong
```

```
    @So1 int,
```

```
    @So2 int,
```

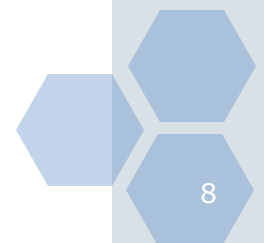
```
    @Tong int output
```

```
AS
```

```
Begin
```

```
    SET @Tong = @So1 + @So2;
```

```
End
```





Cú pháp tạo và thực thi procedure

- ❖ Ví dụ: stored procedure có giá trị trả về
- ❖ --Tạo stored-procedure sp_tong

```
CREATE PROCEDURE sp_Tong
```

```
    @So1 int,
```

```
    @So2 int
```

```
AS
```

```
Begin
```

```
    Declare @Tong int
```

```
    SET @Tong = @So1 + @So2
```

```
    return @Tong
```

```
End
```

Giá trị trả về chỉ có thể là kiểu int/float..
Return value thay cho tham số kiểu output





Cú pháp tạo và thực thi procedure

❖ Biên dịch và gọi thực thi một stored-procedure

- Biên dịch :
 - Chọn toàn bộ mã lệnh Tạo stored-procedure
 - Nhấn F5
- Gọi thực thi một store-Procedure đã được biên dịch bằng lệnh exec:

EXECUTE *proc_name* *--Stored-proc không tham số*

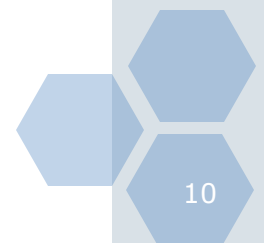
EXEC *proc_name Para1_value, Para2_value, ...* *--Stored proc có tham số*

Ví dụ:

exec sp_LoadTaiKhoan

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

Exec sp_ThemDangKy '001', 'HP01'





STORE PROCEDURE

1

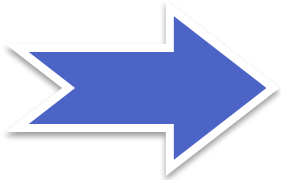
Giới thiệu sơ lược

2

Cú pháp tạo và thực thi procedure

3

Một số câu lệnh khác về procedure





Một số câu lệnh khác về procedure

❖ **Xóa procedure :**

DROP PROCEDURE

❖ **Cập nhật procedure :**

ALTER PROCEDURE sp_name

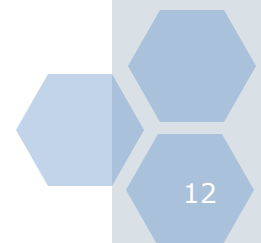
@parameters data_type input/output

AS

BEGIN

Sql_statement

END

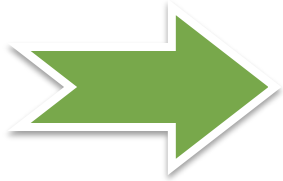




STORE PROCEDURE TRIGGER



TRIGGER



1

Giới thiệu sơ lược

2

Cú pháp tạo trigger

3

Một số câu lệnh khác về trigger



Giới thiệu sơ lược

❖ Trigger:

- Một kiểu store procedure đặc biệt
- Được dịch và lưu trữ thành một đối tượng trong CSDL
- Không có tham số input, output
- Tự động gọi khi có **biến cố** xảy ra

?



Giới thiệu sơ lược

❖ **Ràng buộc toàn vẹn :**

- Ràng buộc tham chiếu

→ *primary key, foreign key*

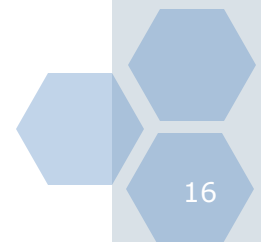
- Ràng buộc miền giá trị trên 1 cột dữ liệu

→ *check, unique, null*

- Ràng buộc liên bộ, liên thuộc tính,...

- VD1 : lương nhân viên không được cao hơn lương người quản lý của nhân viên đó
- VD2 : nhân viên chỉ được làm trưởng phòng của phòng ban mà nhân viên đó trực thuộc
- ...

→ *trigger*





Giới thiệu sơ lược

❖ Bảng tầm ảnh hưởng :

- Bảng có khả năng bị vi phạm ràng buộc
- Thao tác có thể gây ra vi phạm ràng buộc

Bảng/ Thao tác	Thêm	Xóa	Sửa
Bảng 1	-	+	+(thuộc tính)
Bảng 2	+	-	-

+ bị ảnh hưởng
- không ảnh hưởng

- Mỗi dấu + là một trigger
- Các dấu + trên cùng bảng có thể gom lại thành 1 trigger



TRIGGER

1

Giới thiệu sơ lược

2

Cú pháp tạo trigger

3

Một số câu lệnh khác về trigger



Cú pháp tạo trigger

```
CREATE TRIGGER Trigger_name
ON table | view
{ FOR | AFTER | INSTEAD OF }
{ [DELETE] [,] [INSERT] [,] [UPDATE] }
AS
BEGIN
    Sql_statement
END
```

Dùng cho table

Dùng cho view

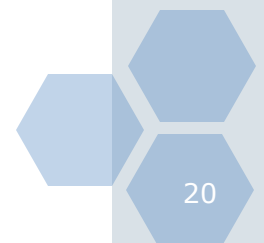


Cú pháp tạo trigger

- ❖ Ví dụ 1 : lương nhân viên không được lớn hơn lương người quản lý của nhân viên đó.

Bảng/ Thao tác	Thêm	Xóa	Sửa
NHANVIEN	+	-	+ (Luong, Ma_NQL)

- Cài trigger trên bảng NHANVIEN
- Thao tác : INSERT, UPDATE





Cú pháp tạo trigger

```
CREATE TRIGGER tr_LuongQL ON NHANVIEN
FOR INSERT, UPDATE
AS
BEGIN
    DECLARE @manql VARCHAR(10), @luong float
    SET @manql = (SELECT ma_nql FROM INSERTED)
    SET @luong = (SELECT luong FROM INSERTED)
    IF (@Luong > (SELECT luong
                    FROM nhanvien
                    WHERE manv = @manql))
    BEGIN
        RAISERROR('Luong nhan vien > luong nguoi quan li',10,0)
        ROLLBACK TRAN
    END
END
```

Trả CSDL về trạng thái trước khi trigger được gọi

Raiserror (msg | id, severity, state)



Cú pháp tạo trigger

NHANVIEN

MaNV	TenNV	Ma_NQL	Luong
001	Tâm		1000

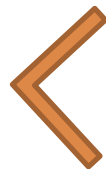
INSERT INTO NHANVIEN (MaNV, TenNV, Ma_NQL, Luong)
VALUES ('002', 'Tiến', '001', 2000)

INSERTED

MaNV	TenNV	Ma_NQL	Luong
002	Tiến	001	2000

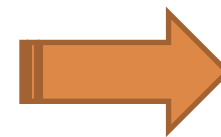
NQL

Luong
1000



INSERTED

Luong
2000



HỦY

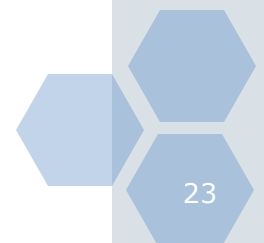


Cú pháp tạo trigger

- ❖ Ví dụ 2 : nhân viên chỉ được làm trưởng phòng của phòng ban mà nhân viên đó trực thuộc.

Bảng/ Thao tác	Thêm	Xóa	Sửa
PHONGBAN	+	-	+ (TRPHG)

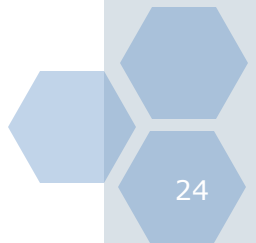
- Cài trigger trên bảng PHONGBAN
- Thao tác : INSERT, UPDATE





Cú pháp tạo trigger

```
CREATE TRIGGER tr_TrPhg ON PHONGBAN
FOR INSERT, UPDATE
AS
BEGIN
    DECLARE @trphg VARCHAR(10), @luong float
    SET @trphg = (SELECT trphg FROM INSERTED)
    SET @phg = (SELECT maphg FROM INSERTED)
    IF (@trphg not in (SELECT manv
                      FROM nhanvien
                      WHERE phg = @phg))
    BEGIN
        RAISERROR('Nhan vien chi duoc lam truong phong ban ho
                  truc thuoc',10,0)
        ROLLBACK TRAN
    END
END
```





Cú pháp tạo trigger

PHONGBAN

MaPHG	TenPHG	TRPHG	NG_NhanChuc
-------	--------	-------	-------------

```
INSERT INTO PHONGBAN (MaPHG, TenPHG, TRPHG)  
VALUES ('5', 'N'Nghien cứu', '005')
```

INSERTED

MaPHG	TenPHG	TRPHG	NG_NhanChu
5	Nghien cứu	005	

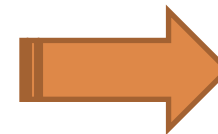
INSERTED

TRPHG
005

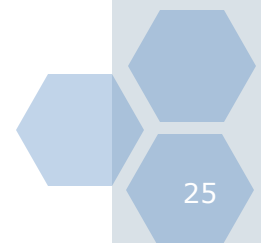
IN

NHANVIEN

MANV
005
...



OK





TRIGGER

1

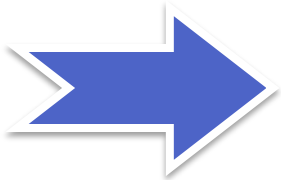
Giới thiệu sơ lược

2

Cú pháp tạo trigger

3

Một số câu lệnh khác về trigger





Một số câu lệnh khác về trigger

❖ **Xóa trigger** (một hay nhiều) :

```
DROP TRIGGER { trigger_name } [ ,...n ]
```

❖ **Cập nhật trigger** :

```
ALTER TRIGGER Trigger_name
```

```
ON table | view
```

```
{ FOR | AFTER | INSTEAD OF }
```

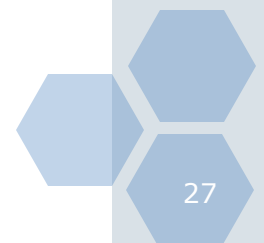
```
{ [DELETE] [,] [INSERT] [,] [UPDATE] }
```

```
AS
```

```
BEGIN
```

```
    Sql_statement
```

```
END
```





Cảm ơn các bạn đã theo dõi