



I. Tóm tắt bài thực hành

1. Yêu cầu lý thuyết

Sinh viên đã được trang bị kiến thức:

- Khai báo các RBTV có bối cảnh 1 quan hệ (Null, Not Null, Check)

2. Nội dung

❖ Ôn lại các kiến thức về RBTV có bối cảnh một quan hệ

- Ôn lại cách khai báo đã được thực hành ở các bài thực hành trước.

❖ Tìm hiểu các kiến thức về RBTV có bối cảnh trên nhiều quan hệ

- Khai báo RBTV có bối cảnh trên nhiều quan hệ.
- Tìm hiểu về kiểu dữ liệu Cursor.

❖ Thực hiện được các bài tập sau

- Sử dụng các câu lệnh khai báo RBTV có bối cảnh nhiều quan hệ trên CSDL Quản lý bán hàng và Quản lý giáo vụ.

II. Trigger

1. Khái niệm

Trigger là một thủ tục đặc biệt, dùng khai báo ràng buộc dữ liệu cho một đối tượng table hoặc view và tự động thực hiện khi một trong 3 phát biểu Insert, Update, Delete thay đổi dữ liệu trên đối tượng đó. Trigger không thể được gọi trực tiếp như các hàm, thủ tục bình thường, nó cũng không có tham số và giá trị trả về.

Trigger chỉ được thực hiện khi phát biểu cập nhật đã thỏa mãn các ràng buộc đã khai báo trên table. Lợi ích chính của trigger là chúng có thể chứa các xử lý phức tạp trên các table có dữ liệu liên quan với table đang cập nhật.

Trigger có thể chứa phát biểu `ROLLBACK TRAN` ngay cả khi không có phát biểu `BEGIN TRAN`. Trong trường hợp phát biểu `ROLLBACK TRANSACTION` bên trong một trigger được thực hiện:

- Nếu trigger này được kích hoạt bởi 1 phát biểu cập nhật từ bên trong một transaction khác, thì toàn bộ transaction đó bị hủy bỏ.
- Nếu trigger được kích hoạt bởi 1 phát biểu cập nhật từ bên trong một gói lệnh, thì sẽ hủy bỏ toàn bộ gói lệnh đó.

Dựa vào ứng dụng, có 3 loại trigger như sau: Insert trigger, Update trigger, Delete trigger.

2. Tạo trigger cho table

Cú pháp Tạo trigger:

```
CREATE TRIGGER <trigger_name> ON <table name>
    [WITH ENCRYPTION]
    FOR DELETE, INSERT, UPDATE
    AS <Các phát biểu T-sql>
```

- ✓ **trigger_name:** Tên trigger phải phân biệt.
- ✓ **ON <tablename>:** tên table mà trigger sẽ thực hiện. Không sử dụng cú pháp trigger này cho View.
- ✓ **WITH ENCRYPTION:** Mã hóa trigger, không cho xem và sửa đổi câu lệnh tạo trigger..
- ✓ **FOR DELETE, INSERT, UPDATE:** Dùng chỉ định những phát biểu cập nhật nào nào trên table sẽ kích hoạt trigger. Khi thực hiện trigger, SQL sẽ tạo các bảng tạm: INSERTED và DELETED.
 - Khi chèn mẫu tin mới vào table thì mẫu tin mới đó cũng lưu trong table INSERTED
 - Khi xóa mẫu tin trong table: Thì các mẫu tin bị xóa đó được di chuyển sang table DELETED.
 - Khi cập nhật mẫu tin trong table: thì table đó và table INSERTED đều chứa mẫu tin có nội dung mới, còn table DELETED chứa mẫu tin có nội dung cũ.

Bạn không thể thay đổi dữ liệu trên các table DELETED VÀ INSERTED. Nhưng bạn có thể dùng 2 table này để xử lý các mẫu tin trên các table liên quan. Ngoài ra, trong trigger Insert và Update, bạn có thể thay đổi nội dung của các mẫu tin mới bằng lệnh Update trên table có trigger.
- ✓ **AS:** Từ khóa bắt đầu các hành động bên trong trigger. Trigger có thể chứa hầu hết các lệnh của T-SQL ngoại trừ một số lệnh sau:
 - Các lệnh CREATE, ALTER, và DROP.
 - TRUNCATE TABLE
 - SELECT INTO (vì câu lệnh này cũng tạo ra bảng)

3. Kích hoạt/Vô hiệu một trigger

```
ALTER TABLE table ENABLE | DISABLE TRIGGER ALL | trigger_name[,...n]
```

4. Hiệu chỉnh trigger

Bạn có thể thay đổi các lệnh cần thực hiện cũng như hành động cập nhật mà Trigger sẽ được gọi thực hiện.

```
ALTER TRIGGER trigger_name ...
```

5. Xóa trigger

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

Nếu xóa một table thì tất cả trigger của nó cũng bị xóa.

III. Ví dụ

1. Ví dụ 1

Yêu cầu: (Bài tập Quản lý Bán hàng – Phần 1 – Câu 11 – Trang 3) Cài đặt trigger kiểm tra Ngày mua hàng (NGHD) của một khách hàng thành viên sẽ lớn hơn hoặc bằng ngày khách hàng đó đăng ký thành viên (NGDK).

Thực hiện:

```
1 CREATE TRIGGER trg_ins_hd ON HOADON
2 FOR INSERT
3 AS
4 BEGIN
5     DECLARE @NgayHD smalldatetime, @MaKH char(4), @NgayDK smalldatetime
6     -- Lay thong tin cua HOADON vua moi them vao
7     SELECT @NgayHD = NGHD, @MaKH = MAKH
8     FROM INSERTED
9     -- Lay thong tin KHACHHANG tuong ung
10    SELECT @NgayDK = NGDK
11    FROM KHACHHANG
12    WHERE MAKH = @MaKH
13    -- So sanh
14    IF (@NgayHD < @NgayDK) -- Neu Ngay hoa don nho hon ngay dang ky thanh vien
15    BEGIN
16        PRINT 'LOI: NGAY HOA DON KHONG HOP LE!'
17        ROLLBACK TRANSACTION
18    END
19    ELSE
20    BEGIN
21        PRINT 'THEM MOI MOT HOA DON THANH CONG!'
22    END
23 END
```

2. Ví dụ 2

Yêu cầu: Cài đặt trigger thực hiện việc cập nhật lại trị giá hóa đơn khi có một chi tiết hóa đơn mới được thêm vào.

Thực hiện:

```

1 CREATE TRIGGER trg_ins_cthd ON CTHD
2 FOR INSERT
3 AS
4 BEGIN
5     DECLARE @SoHD int, @MaSP char(4), @SoLg int, @TriGia money
6     -- Lay thông tin của CTHD vừa mới thêm vào
7     SELECT @SoHD = SOHD, @MaSP = MASP, @SoLg = SL
8     FROM INSERTED
9     -- Tính trị giá của sản phẩm mới thêm vào HOADON
10    SET @TriGia = @SoLg * SELECT GIA FROM SANPHAM WHERE MASP = @MaSP
11    -- Khai báo một CURSOR duyệt qua tất cả các CTHD đã có sản trong HOADON
12    DECLARE cur_cthd CURSOR
13    FOR
14        SELECT MASP, SL
15        FROM CTHD
16        WHERE SOHD = @SoHD
17
18    OPEN cur_cthd
19    FETCH NEXT FROM cur_cthd
20    INTO @MaSP, @SoLg
21
22    WHILE (@@FETCH_STATUS = 0)
23    BEGIN
24        -- Cộng đơn trị giá của từng sản phẩm vào biến TriGia
25        SET @TriGia = @TriGia + @SoLg * SELECT GIA FROM SANPHAM WHERE MASP = @MaSP
26        FETCH NEXT FROM cur_cthd
27        INTO @MaSP, @SoLg
28    END
29
30    CLOSE cur_cthd
31    DEALLOCATE cur_cthd
32    --Tiến hành cập nhật lại trị giá HOADON
33    UPDATE HOADON SET TRIGIA = @TriGia WHERE SOHD = @SoHD
34 END

```

IV. Bài tập yêu cầu

1. Bài tập 1 (Tại lớp)

Sinh viên hoàn thành Phần I bài tập QuanLyBanHang từ câu 11 đến 15.

2. Bài tập 2 (Về nhà)

Sinh viên hoàn thành Phần I bài tập QuanLyGiaoVu câu 9, 10 và từ câu 15 đến câu 24.

~ HẾT ~