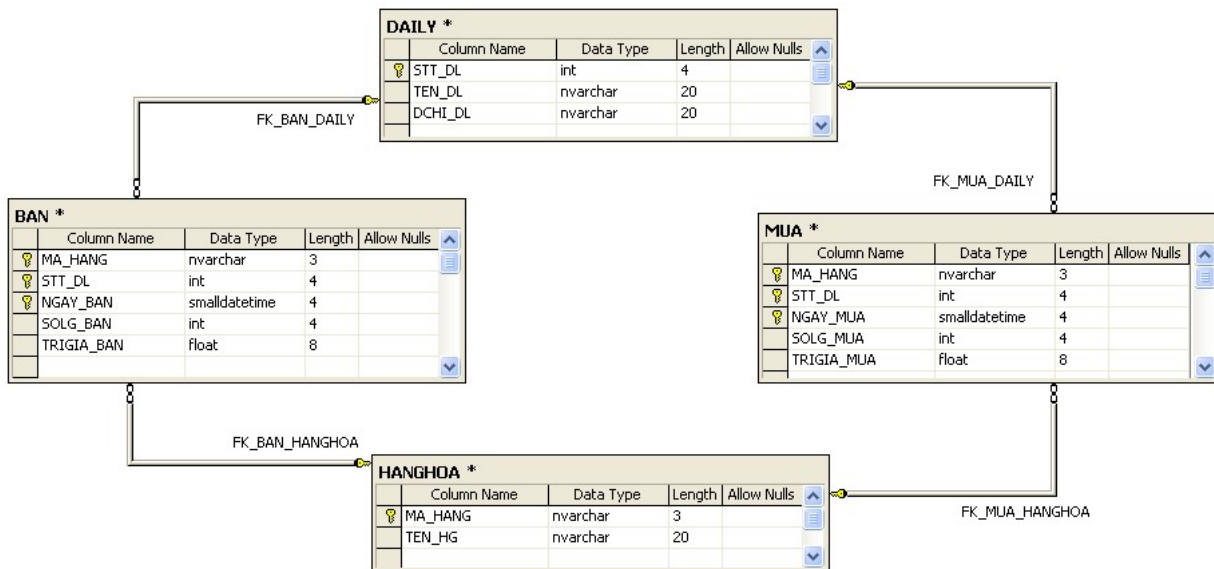


## Lab 5

# QUẢN TRỊ CƠ SỞ DỮ LIỆU VỚI SQL SERVER

**Bài 1:** Tạo CSDL QL\_HANGHOA như hình bên dưới sử dụng T-SQL trong Query Analyzer.



Hình 5.1: Cấu trúc của CSDL HangHoa

Các ràng buộc khác cần tạo:

- $NGAY\_MUA \leq NGAY\ HIỆN\ TẠI\ CỦA\ HỆ\ THỐNG$
- $NGAY\_BAN \leq NGAY\ HIỆN\ TẠI\ CỦA\ HỆ\ THỐNG$
- $SOLG\_BAN > 0, SOLG\_MUA > 0, TRIGIA\_BAN > 0, TRIGIA\_MUA > 0$
- Giá trị mặc định của  $NGAY\_MUA = NGAY\ HIỆN\ TẠI\ CỦA\ HỆ\ THỐNG$
- Giá trị mặc định của  $NGAY\_BAN = NGAY\ HIỆN\ TẠI\ CỦA\ HỆ\ THỐNG$

**Gợi ý:** Sinh viên nên làm theo các thứ tự sau:

1. Tạo CSDL QL\_HANGHOA
2. Tạo các bảng như DAILY, HANGHOA, MUA, BAN
3. Tạo các ràng buộc tham chiếu khóa ngoại với các cascade update và cascade delete.
4. Tạo các ràng buộc khác.

**Bài 2:** Sử dụng **T-SQL** để chỉnh sửa CSDL QL\_HANGHOA theo các yêu cầu sau:

- Thêm cột SODT cho phép lưu được số điện thoại gồm 11 ký tự số vào bảng DAILY.
- Thêm 2 cột DVT (kiểu CHAR(12)) và cột NCC (kiểu CHAR(30)) vào bảng HANGHOA.
- Thêm bảng LOAI\_HG(MA\_LOAI: CHAR(2), TEN\_LOAI: CHAR(50)). Khóa chính là MA\_LOAI.
- Thêm cột MA\_LOAI: CHAR(2) vào bảng HANGHOA
- Tạo khóa ngoại tham chiếu:

HANGHOA(MA\_LOAI)→ LOAI\_HG(MA\_LOAI)

với ràng buộc cascade update và cascade delete.

### **Bài 3: Ý nghĩa khóa ngoại**

#### ***Yêu cầu:***

- Về nguyên tắc khi nhập dữ liệu vào CSDL QL\_HANGHOA trên thì ta cần nhập dữ liệu cho 2 bảng DAILY và HANGHOA trước sau đó mới nhập dữ liệu cho bảng MUA và bảng BAN. Vì sao?
- Hãy thử nhập dữ liệu cho bảng MUA trước khi nhập dữ liệu cho 2 bảng DAILY và HANGHOA (Lưu ý là giá trị mã hàng hay số thứ tự đại lý nhập vào bảng MUA phải chưa có trong bảng HANGHOA và bảng DAILY). Xem thông báo lỗi của SQL SERVER và cho nhận xét ý nghĩa của cascade update.
- Tương tự thử nhập dữ liệu vào bảng DAILY và HANGHOA sau đó mới nhập dữ liệu cho bảng MUA và bảng BAN (Khi nhập lưu ý đến tính hợp lệ của dữ liệu). Sau đó thử sửa mã hàng trên bảng HANGHOA và mở bảng MUA (cả bảng BAN) để quan sát dữ liệu và nhận xét tác dụng của cascade update trong trường hợp này.
- Mở diagram của CSDL QL\_HANGHOA, click phải chuột lên quan hệ giữa bảng HANGHOA và bảng MUA (tương tự với quan hệ giữa bảng DAILY và bảng MUA), chọn properties và kiểm tra xem đã đánh dấu chọn cascade delete chưa? Hãy thử kiểm tra xét 2 trường hợp đánh dấu chọn cascade delete và bỏ chọn đánh dấu chọn cascade delete thì kết quả xóa dữ liệu ở bảng HANGHOA cho từng trường hợp sẽ như thế nào? Nhận xét ý nghĩa của cascade delete.

#### Bài 4: Lập trình T-SQL

1. Thực hiện câu lệnh tạo bảng sau:

```
create table product
(
    prod_nr int not null primary key,
    name varchar(30) not null,
    price money not null,
    type varchar(30) not null
)
```

2. Thực hiện các câu lệnh thêm dữ liệu vào bảng sau:

```
insert into product values (4,'ColorTv',700,'electronic')
insert into product values (5,'Fan',350,'electronic')
insert into product values (6,'Heater',200,'electronic')
insert into product values (7,'Webcam',60,'Computer')
```

3. Viết câu truy vấn bảng product và quan sát dữ liệu.
4. Viết một chương trình để tính trị giá trung bình của mỗi sản phẩm, ghi trị giá này ra một biến và hiển thị trị giá trung bình này thông qua biến vừa ghi.
5. Viết một chương trình để tính trị giá trung bình của mỗi sản phẩm. *Kiểm tra nếu trị giá trung bình của mỗi sản phẩm  $\geq 500$  thì hiển thị bảng giá chi tiết của mỗi sản phẩm. Ngược lại thực hiện công việc tăng đều giá mỗi sản phẩm lên 5% so với giá hiện tại của nó. Công việc kiểm tra và tăng giá được thực hiện liên tục đến khi nào trị giá trung bình  $\geq 500$  thì dừng lại và hiển thị trị giá trung bình và liệt kê bảng giá chi tiết của mỗi sản phẩm sau khi tăng.*

#### Bài 5: Store Procedure - CSDL QLHangHoa

1. Viết thủ tục tính số lượng tồn của 1 mặt hàng của 1 đại lý cụ thể.

```
Create Procedure TonKho
(
    @STT_DL int, @Ma_HAng nvarchar(3),
    @TongMua int Output, @TongBan int Output, @Ton int Output
)
As
Begin
    Select @TongMua=Sum(SoLg_Mua)
    From Mua
    Where STT_DL=@STT_DL and Ma_HAng=@Ma_Hang
    If @TongMua Is NULL Return
```

```
Select @TongBan=Sum(SoLg_Ban)
From Ban
Where STT_DL=@STT_DL and Ma_Hang=@Ma_Hang

If @TongBan Is NULL
    Set @TongBan = 0

Set @Ton = @TongMua-@TongBan
End

Gọi thực thi thủ tục:

Declare @Mua int, @Ban int, @Ton int
Execute TonKho 2, '002', @Mua Output, @Ban Output, @Ton Output
print 'Tong mua: ' + Cast(@Mua as VArchar(20)) +
    '. Tong ban ' + Cast(@Ban as varchar(20)) +
    '. Ton: ' + Cast(@Ton as varchar(20))
```

## 2. Viết thủ tục bán 1 mặt hàng của một đại lý cụ thể với số lượng và đơn giá bán cho trước, ngày bán là ngày hiện hành

```
Create Procedure BanHang
    @STT_DL int, @Ma_Hang nvarchar(3), @SoLgBan int, @DonGia int
As
Begin
    Declare @TongMua int, @TongBan int
    Select @TongMua=Sum(SoLg_Mua)
    From Mua
    Where STT_DL=@STT_DL and Ma_HAng=@Ma_Hang

    If @TongMua Is NULL
    Begin
        print 'Mat hang ' + @Ma_Hang + ' khong co' + ' o dai ly'
        Return
    End

    Select @TongBan=Sum(SoLg_Ban)
    From Ban
    Where STT_DL=@STT_DL and Ma_Hang=@Ma_Hang

    if @TongBan Is Null
        Set @TongBan=0
```

```
if @TongMua >= @TongBan + @SoLgBan
    Insert Into Ban Values(@Ma_Hang, @STT_DL, GetDate(),
        @SoLgBan, @DonGia)
else
    print 'Khong du hang de ban'
End
```

**Thực thi thủ tục:**

Execute BanHang 2, '002', 160, 12

3. Viết thủ tục cho phép tính tồn kho của mỗi mặt hàng của một đại lý cụ thể (STT\_DL là tham số).

4. Viết thủ tục cho phép tính tồn kho của mỗi mặt hàng của mỗi đại lý.

5. Giả sử các khóa ngoại của bảng BAN và MUA có thuộc loại **On Delete No Action**.

Viết thủ tục cho phép xóa một hàng hóa cùng các thông tin liên quan tới hàng hóa này ở bảng BAN và MUA với mã hàng là tham số.

**Gợi ý:**

- Thực hiện xóa ở bảng BAN, MUA
- Thực hiện xóa ở bảng HANGHOA

**Bài 6: Store Procedure - CSDL tự tạo**

Tạo bảng sau:

```
Create table nhanvien
(
    manv int not null primary key,
    hoten varchar(30) not null,
    diachi varchar(30) not null)

```

1. Viết thủ tục tên “insert\_nhanvien” cho phép thêm một mẫu tin nhân viên vào bảng nhanvien .

2. Viết đoạn lệnh thêm một mẫu tin nhân viên với nội dung sau:

1, ‘Nguyen Van Thanh’, ‘01 Ly Tu Trong, NK– TPCT’

Sau đó hiển thị bảng nhân viên để kiểm tra kết quả.

3. Viết thủ tục cập nhật địa chỉ của nhân viên

- *Input: mã nhân viên và địa chỉ đúng*

4. Viết đoạn lệnh cập nhật địa chỉ của nhân viên có manv=1 và địa chỉ đúng là

‘10 Ly Tu Trong, Q. NK – TPCT’

Sau đó hiển thị bảng nhân viên để kiểm tra kết quả.

**Bài 7: Store Procedure - CSDL Northwind****1. Viết thủ tục/hàm tính tiền hóa đơn với**

- *Input*: ORDERID

- *Output*: *Trị giá hóa đơn tương ứng*

**2. Hướng dẫn:****+ Cách 1:**

```
Create Proc CalulateMoney_Order @Order_ID char(5)
AS
select sum(UnitPrice*Quantity-
           UnitPrice*Quantity*Discount) as Total_Price
from [Order Details]
where OrderId=@Order_ID
-----
Exec CalulateMoney_Order '10249'
```

**+ Cách 2:**

```
create function dbo.CalulateMoney(@Order_ID char(5))
returns table
as
return (select sum(UnitPrice*Quantity-
                   UnitPrice*Quantity*Discount) as Total_Price
        from [Order Details] where OrderId=@Order_ID)
-----
select *
```

**+ Cách 3:**

```
create function dbo.Calulate(@Order_ID char(5))
returns money
as
Begin
    Declare @a money
    select @a= sum(UnitPrice*Quantity - UnitPrice*Quantity*Discount)
    from [Order Details]
    where OrderId=@Order_ID
    return @a
End
-----
select dbo.Calulate('10249')
```

**Bài 8: Lập trình**

1. Tạo bảng Account chứa thông tin về tài khoản của 1 người với mã số tài khoản là khóa chính (int); ngoài ra còn có tên tài khoản (varchar(30))

2. Biết rằng số tài khoản (account number) gồm 9 chữ số chẳng hạn 972428577. Viết một hàm kiểm tra số tài khoản này có hợp lệ theo nguyên tắc kiểm tra chia dư 11 ('modulo11-check').

Thí dụ:

972428577 là hợp lệ vì:

$$(9*9 + 8*7 + 7*2 + 6*4 + 5*2 + 4*8 + 3*5 + 2*7 + 1*7) \% 11 = 0$$

3. Sử dụng hàm này để định nghĩa ràng buộc toàn vẹn: Số tài khoản phải thỏa mãn nguyên tắc 'chia dư 11'

**Gợi ý:** Dùng 1 check constraint