

VIEW

1. Tạo view

```
/*
 * Tạo view chứa câu lệnh truy vấn chi tiết hóa
 * đơn đầy đủ thông tin hàng hóa bằng cách kết nối
 * các bảng lại với nhau
 */
CREATE VIEW View_ChiTietHD
AS
SELECT cthd.*, TenHH, TenLoai, TenCongTy FROM ChiTietHD cthd
      JOIN HangHoa hh ON hh.MaHH=cthd.MaHH
      JOIN Loai lo ON lo.MaLoai=hh.MaLoai
      JOIN NhaCungCap ncc ON ncc.MaNCC=hh.MaNCC
```

2. Sử dụng view

```
SELECT * FROM View_ChiTietHD WHERE MaHD=10248
```

PROCEDURE

1. Tạo procedure

```
/*
 * Thủ tục lưu cho phép truy vấn chi tiết hóa
 * đơn đầy đủ thông tin hàng hóa bằng cách kết nối
 * các bảng lại với nhau
 */
CREATE PROCEDURE sp_ChiTietHD
(
    @MaHD INT
)
AS
SELECT cthd.*, TenHH, TenLoai, TenCongTy FROM ChiTietHD cthd
      JOIN HangHoa hh ON hh.MaHH=cthd.MaHH
      JOIN Loai lo ON lo.MaLoai=hh.MaLoai
      JOIN NhaCungCap ncc ON ncc.MaNCC=hh.MaNCC
```

2. Sử dụng

```
EXECUTE sp_ChiTietHD 10248
```

3. Tạo

```
/*
 * Thủ tục lưu cho phép chèn thêm 1 loại mới
 * và lấy mã tự tăng thông qua tham số ra (output)
 */
CREATE PROCEDURE sp_InsertLoai
(
    @MaLoai INT OUTPUT,      -- tham số ra
    @TenLoai NVARCHAR(50),   -- tham số vào
    @Hinh NVARCHAR(50),      -- tham số vào
    @MoTa NVARCHAR(255)      -- tham số vào
)
AS
BEGIN
    -- thêm 1 loại vào CSDL
    INSERT INTO Loai(TenLoai, Hinh, MoTa)
        VALUES(@TenLoai, @Hinh, @MoTa)
    -- lấy mã tự tăng vừa sinh ra
    SELECT @MaLoai = @@IDENTITY
END
```

4. Sử dụng

```
DECLARE @MaHH INT
EXECUTE sp_InsertLoai [redacted], N'Tên', N'Hình', N'Mô tả'
PRINT (@MaHH)
```

global variable @@IDENTITY

Đặt tham số
@MaHH OUTPUT
vào vị trí này

5. Tạo

```
/*
 * Thủ tục lưu cho phép cập nhật loại hàng hóa
 */
CREATE PROCEDURE sp_UpdateLoai
(
    @MaLoai INT,
    @TenLoai NVARCHAR(50),
    @Hinh NVARCHAR(50),
    @MoTa NVARCHAR(255)
)
AS
BEGIN
    UPDATE Loai
        SET TenLoai=@TenLoai, Hinh=@Hinh, MoTa=@MoTa
        WHERE MaLoai=@MaLoai
END
```

6. Tạo

```
/*
 * Thủ tục lưu cho phép xóa một loại hàng hóa
 */
CREATE PROCEDURE sp_DeleteLoai
(
    @MaLoai INT
)
AS
BEGIN
    DELETE Loai WHERE MaLoai=@MaLoai
END
```

7. Tạo

```
/*
 * Thủ tục lưu cho phép lấy một loại hàng hóa
 */
CREATE PROCEDURE sp_SelectLoai
(
    @MaLoai INT
)
AS
BEGIN
    SELECT * FROM Loai WHERE MaLoai=@MaLoai
END
```

8. Tạo

```
/*
 * Thủ tục lưu cho phép lấy các loại hàng
 * hóa có tên chứa tham số
 */
CREATE PROCEDURE sp_SearchLoai
(
    @TenLoai NVARCHAR(50)
)
AS
BEGIN
    SELECT * FROM Loai
    WHERE TenLoai LIKE '%' + @TenLoai + '%';
END
```

FUNCTION

1. Tạo

```
/*  
 * Hàm tính doanh số của một mặt hàng  
 */  
CREATE FUNCTION fn_TinhDoanhSo  
(  
    @MaHH INT  
)  
RETURNS FLOAT  
AS  
BEGIN  
    DECLARE @DoanhSo FLOAT  
  
    SELECT @DoanhSo=SUM(SoLuong*DonGia*(1-GiamGia))  
        FROM ChiTietHD WHERE MaHH=@MaHH  
  
    RETURN @DoanhSo  
END
```

2. Sử dụng

```
SELECT TenHH, dbo.fn_TinhDoanhSo(MaHH) AS DoanhSo  
FROM HangHoa
```

TRIGGER

1. Tạo

```
/*
 * Trigger cập nhật tổng tiền của hóa
 * đơn khi chi tiết hóa đơn bị thay đổi
 */
CREATE TRIGGER tg_UpdateThanhTien
ON ChiTietHD
AFTER INSERT, UPDATE
AS
BEGIN
    DECLARE @MaHD INT, @ThanhTien FLOAT
    -- lấy mã hàng hóa từ bảng inserted
    SELECT @MaHD=MaHD FROM INSERTED
    -- tính tổng tiền trên bảng ChiTietHD
    SELECT @ThanhTien=SUM(SoLuong*DonGia*(1-GiamGia))
        FROM ChiTietHD WHERE MaHD=@MaHD
    -- cập nhật thành tiền của hóa đơn
    UPDATE HoaDon
        SET ThanhTien=@ThanhTien
        WHERE MaHD=@MaHD
END
```

2. Sử dụng

```
--chèn thêm một mặt hàng vào bảng chi tiết hóa đơn
INSERT INTO ChiTietHD(MaHD, MaHH, SoLuong, DonGia, GiamGia)
VALUES(10248, 1005, 10, 100, 0.1)
--kiểm tra sự thanh đổi của cột thành tiền
SELECT ThanhTien FROM HoaDon WHERE MaHD=10248
```

Problem: Paging (Phân trang)

```
/*
 * Phân trang bảng hàng hóa
 */
ALTER PROCEDURE sp_GetHangHoa
(
    @RowCount INT OUTPUT,    --tổng số bản ghi
    @Start INT = 0,         -- vị trí bản ghi bắt đầu
    @Length INT = 10        -- số lượng bản ghi cần lấy
)
AS
BEGIN
    -- bảng tạm có chứa số thứ tự các bản ghi (RowIndex)
    WITH HangHoaTmp AS
    (
        SELECT *,
            ROW_NUMBER() OVER (ORDER BY MaHH) AS RowIndex
        FROM HangHoa
    )
    -- select lại từ bảng tạm và lọc theo RowIndex
    SELECT * FROM HangHoaTmp
        WHERE RowIndex BETWEEN @Start + 1 AND @Start + @Length
    -- lấy tổng bản ghi
    SELECT @RowCount = COUNT(*) FROM HangHoa;
END
```

Bài tập Function

(sử dụng eStore2)

Câu 1: Viết 1 function tên `fn_ThongKeMatHangTheoNhaCungCap` kiểu trả về là Table dùng để lấy số mặt hàng theo từng nhà cung cấp.

Ví dụ:

MaNCC	SoLuongMatHang
1	100
2	50
3	30

Câu 2: Viết 1 function tên `fn_LayThongTinKhachHang` trả về kiểu int, nhận vào tham số là `@MaKH`: dùng để đếm khách hàng này đã mua bao nhiêu hóa đơn.

Ví dụ:

@MaKH	ANTON
SoHoaDonDaMua	100

Câu 3: Viết 1 function tên fn_LayThongTinHoaDon trả về kiểu float/money, nhận vào tham số là @MaHD: dùng để tính trị giá của hóa đơn này.

Ví dụ:

@MaHD	1
TriGia	1000

Câu 4: Viết 1 function tên fn_ThongKeHoaDon trả về kiểu int, nhận vào tham số là @Year int, @month int: dùng để đếm số lượng các hóa đơn đã đặt hàng trong tháng @month và năm @year.

Câu 5: Viết 1 function tên fn_ThongKeKhachHang trả về kiểu int, nhận vào tham số là giới tính @sex bit, @year int, @month int: dùng để đếm có bao nhiêu khách hàng ở thành phố @city đã đặt hàng trong tháng @month và năm @year

Câu 6: Viết 1 function tên fn_ThongKeMatHangTheoLoaiHang trả về kiểu Table, nhận vào tham số là @MaLoai int, @MaNCC nvarchar(50): dùng để lấy danh sách các mặt hàng thuộc loại mặt hàng có MaLoai là @MaLoai của nhà cung cấp có MaNCC là @MaNCC.

Ví dụ :

MaHH	TenHH	DonGia
1	Chai	20
2	Chang	30

Câu 7: Viết 1 function tên fn_ThongKeGiaMatHang trả về kiểu float/money, nhận vào tham số là @MaNCC nvarchar(50): dùng để tính đơn giá trung bình các mặt của nhà cung cấp có MaNCC là @MaNCC.