

```
--THUC HANH TUAN 03
--BAI TAP TUAN 04 - VIEW
-- MSSV HO PHUC LAM
use Northwind
```

--1) Tạo view vw_Products_Info hiển thị danh sách các sản phẩm từ bảng Products và bảng Categories.

--Thông tin bao gồm CategoryName, Description, ProductName, QuantityPerUnit, UnitPrice, UnitsInStock

```
create view vw_Products_Info
(CategoryName, Description, ProductName, QuantityPerUnit, UnitPrice, UnitsInStock)
as
select c.CategoryName, c.Description, p.ProductName, p.QuantityPerUnit, p.UnitPrice,
p.UnitsInStock
from Products p, Categories c
where c.CategoryID = p.CategoryID
```

--xem lai

```
select * from [dbo].[vw_Products_Info]
Go
```

--2) Tạo view List_Product_view chứa danh sách các sản phẩm dạng hộp (box) có đơn giá > 16, --thông tin gồm ProductID, ProductName, UnitPrice, QuantityPerUnit, COUNT of OrderID

```
create view List_Product_view
as
select p.ProductID, p.ProductName, p.UnitPrice, p.QuantityPerUnit, count(od.OrderID) as 'COUNT of OrderID'
from Products p, [Order Details] od
where p.ProductID = od.ProductID and p.UnitPrice > '16'
group by p.ProductID, p.ProductName, p.UnitPrice, p.QuantityPerUnit
```

--xem lai

```
select * from List_Product_view
```

Go

--3) Tạo view vw_CustomerTotals hiển thị tổng tiền bán được từ mỗi khách hàng theo tháng và theo năm.

--Thông tin gồm CustomerID, YEAR(OrderDate) AS OrderYear, MONTH(OrderDate) AS OrderMonth, SUM(UnitPrice*Quantity)

```
create view vw_CustomerTotals
```

```
as
```

```
select c.CustomerID, year(OrderDate) as OrderYear, MONTH(OrderDate) as OrderMonth ,
```

```
sum(od.UnitPrice*od.Quantity) as Total
```

```
from Customers c, Orders o, [Order Details] od
```

```
where c.CustomerID = o.CustomerID and o.OrderID = od.OrderID
```

```
group by c.CustomerID, year(OrderDate), MONTH(OrderDate)
```

--xem lại

```
select * from vw_Products_Info
```

Go

--4) Tạo view trả về tổng số lượng sản phẩm bán được của mỗi nhân viên (Employee) theo từng năm.

--Thông tin gồm EmployeeID, OrderYear, sumOfOrderQuantity

```
create view SumOfOrder
```

```
as
```

```
select EmployeeID, year(o.OrderDate) as OrderYear, Sum(Quantity) as sumOfOrderQuantity
```

```
from Orders o, [Order Details] od
```

```
where o.OrderID = od.OrderID
```

```
group by EmployeeID, year(o.OrderDate)
```

--xem lại

```
select * from [dbo].[SumOfOrder]
```

Go

--5) Tạo view ListCustomer_view chứa danh sách các khách hàng có trên 5 hóa đơn đặt hàng từ năm 1997 đến 1998,

--thông tin gồm mã khách (CustomerID) , họ tên(CompanyName), Số hóa đơn (CountOfOrders)

```
create view ListCustomer_view
as
select c.CustomerID, c.CompanyName, count(o.OrderID) as CountOfOrders
from Customers c , Orders o
where o.CustomerID = c.CustomerID and year(o.OrderDate) in (1977, 1998)
group by c.CustomerID, c.CompanyName
having count(o.OrderID) > 5
```

--xem lai

```
select * from ListCustomer_view
Go
```

--6) Tạo view ListProduct_view chứa danh sách những sản phẩm nhóm Beverages và Seafood có tổng số lượng bán trong mỗi năm trên 30 sản phẩm,

--thông tin gồm CategoryName, ProductName, Year, SumOfOrderQuantity.

```
create view ListProduct_view
as
select c.CategoryName, p.ProductName, year(o.OrderDate) as YEARR, sum(od.Quantity) as
SumOfOrderQuantity
from Categories c, Products p, [Order Details] od, Orders o
where c.CategoryID = p.CategoryID and p.ProductID = od.ProductID and od.OrderID = o.OrderID and
c.CategoryName in ('Beverages', 'Seafood')
group by c.CategoryName, p.ProductName, year(o.OrderDate)
having sum(od.Quantity) > '30'
```

```
select * from ListProduct_view
Go
```

--7) Tạo view vw_OrderSummary với từ khóa WITH ENCRYPTION gồm OrderYear (năm của ngày lập hóa đơn), OrderMonth (tháng của ngày lập hóa đơn), --OrderTotal (tổng tiền, =UnitPrice*Quantity). Sau đó xem thông tin và trợ giúp về mã lệnh của view này

```
create view VW_OrderSummary
with encryption
as
select Year(o.OrderDate) as OrderYear, month(o.OrderDate) as OrderMonth,
sum(od.UnitPrice*od.Quantity) as OrderTotal
from Orders o, [Order Details] od
where o.OrderID = od.OrderID
group by Year(o.OrderDate) , month(o.OrderDate)
```

--kiem tra ton tai

```
SELECT * FROM INFORMATION_SCHEMA.VIEWS WHERE TABLE_NAME = 'VW_OrderSummary';
```

--xem dinh nghia view

```
EXEC sp_helptext 'VW_OrderSummary';
```

--xem thong tin chi tiet

```
EXEC sp_help 'VW_OrderSummary';
```

Go

--8) Tạo view vwProducts với từ khóa WITH SCHEMABINDING gồm ProductID, ProductName, Discount.

--Xem thông tin của View. Xóa cột Discount. Có xóa được không? Vì sao?

```
create view vwProducts
WITH SCHEMABINDING
as
select p.productID, p.ProductName, od.Discount
from Products p, [Order Details] od
where p.ProductID = od.ProductID
```

```

/*
---BÀI TẬP TUẦN 5-6
*/

-- batch
--1) Viết một batch khai báo biến @tongsoHD chứa tổng số hóa đơn của sản phẩm có ProductID='31',
nếu @tongsoHD>50 thì
--in ra chuỗi “Sản phẩm 31 có trên 50 đơn hàng”, ngược lại tin ra chuỗi “Sản phẩm 31 có ít đơn
đặt hàng

-- Khai báo biến @tongsoHD
DECLARE @tongsoHD INT;

-- Tính tổng số hóa đơn của sản phẩm có ProductID = '31'
SELECT @tongsoHD = COUNT(*)
FROM [Order Details]
WHERE ProductID = 31;

-- Kiểm tra giá trị của @tongsoHD và in ra thông báo tương ứng
IF @tongsoHD > 50
    PRINT 'Sản phẩm 31 có trên 50 đơn hàng'
ELSE
    PRINT 'Sản phẩm 31 có ít đơn đặt hàng';
Go

/*2) Viết một đoạn Batch với tham số @makh và @n, chứa số hóa đơn @n của
khách hàng @makh, tham số @nam chứa năm lập hóa đơn (ví dụ
@nam=1996), nếu @n>0 thì in ra chuỗi:”Khách hàng có @n hóa đơn trong
năm 1996”, ngược lại nếu @n=0 thì in ra chuỗi “Khách hàng không có hóa
đơn nào trong năm 1996”*/
--khai báo
declare @makh nvarchar(50);

```

```

declare @n int;
declare @nam int;

--Gán giá trị
set @makh = 'ALFKI';
set @nam = '1996';

--Tính tổng số hóa đơn của khách hàng
select @n = count(*)
from Orders where CustomerID = @makh and year(OrderDate) = @nam;

--Kiểm tra điều kiện
if @n > 0      print N'Khách hàng có'+convert(nvarchar, @n) +'hóa đơn trong năm
'+convert(nvarchar, @nam);

if @n = 0      print N'Khách hàng không có hóa đơn nào trong năm '+convert(nvarchar, @nam);

Go

```

```

/*
3) Viết một batch tính số tiền giảm cho những hóa đơn (OrderID) có tổng
tiền>500, thông tin gồm OrderID, TongTien=sum(UnitPrice*Quantity),
Tiền giảm, với Tiền giảm được tính như sau:
☐ Những hóa đơn có TongTien <500 thì không giảm,
☐ TongTien từ 500 đến <5000 thì giảm 5% của TongTien
☐ TongTien từ 5000 đến <10000 thì giảm 10% của TongTien
☐ TongTien từ 10000 trở lên thì giảm 15% của TongTien
(Gợi ý: Dùng cấu trúc Case... when ...then ...)
*/

```

```

--Kiểm tra điều kiện

```

```

select od.OrderID, sum(od.UnitPrice*od.Quantity) as TongTien,
       case
         when sum(od.UnitPrice*od.Quantity) < 500 then 0
         when sum(od.UnitPrice*od.Quantity) >= 500 and sum(od.UnitPrice*od.Quantity) < 5000 then
sum(od.UnitPrice*od.Quantity) * 0.05
         when sum(od.UnitPrice*od.Quantity) >= 5000 and sum(od.UnitPrice*od.Quantity) < 10000
then sum(od.UnitPrice*od.Quantity) * 0.10
         when sum(od.UnitPrice*od.Quantity) >= 10000 then sum(od.UnitPrice*od.Quantity) * 0.15
       end as tienGiam

from [Order Details] od
group by OrderID
having sum(od.UnitPrice*od.Quantity) > 500
GO

```

```

/*
--4) Viết một Batch với 3 tham số: @MaNCC, @MaSP, @SoLuongCC, chứa giá
trị của các field SupplierID, ProductID,Quantity, với giá trị truyền cho các
biến @MaNCC, @MaSP (vd: @MaNCC =5, @MaSP =11, thì chương trình sẽ
gán giá trị tương ứng của field Quantity cho biến @SoLuongCC, nếu
@SoLuongCC trả về giá trị là null thì in ra chuỗi “Nha cung cap 5 khong
cung cap san pham 11”, ngược lại (vd: @SoLuongCC =12) thì in chuỗi “Nha
cung cap 5 cung cap san pham 11 với số lượng là 12”
*/

```

```

--Xem các bảng cần kết
select * from Suppliers
select * from Products
select * from [Order Details]

```

```

--khai bao
declare @MaNCC int;
declare @MaSP int;
declare @SoLuongCC int;

```

```

--Gán giá trị cho các tham số
set @MaNCC = 5;
set @MaSP = 11;

--truy vấn
select @SoLuongCC = od.Quantity
from Products p, Suppliers s, [Order Details] od
where p.SupplierID = s.SupplierID and od.ProductID = p.ProductID and p.SupplierID = @MaNCC and
p.ProductID = @MaSP;

--kiểm tra điều kiện
if @SoLuongCC is null
    print N'Nhà cung cấp ' + convert(nvarchar, @MaNCC) + N' không cung cấp sản phẩm ' +
convert(nvarchar, @MaSP)
else
    print N'Nhà cung cấp ' + convert(nvarchar, @MaNCC) + N' cung cấp sản phẩm ' +
convert(nvarchar, @MaSP) + N' với số lượng là ' + convert(nvarchar, @SoLuongCC);
Go

/*
5) Viết một batch thực hiện tăng đơn giá (UnitPrice) trung bình của đơn hàng(Orders) theo điều
kiện sau:
Khi trung bình của đơn giá trong đơn hàng < 50 thì cập nhật tăng đơn giá của đơn hàng lên 10%,
nếu sau khi cập nhật mà đơn giá lớn nhất của đơn hàng >300 thì dừng.
*/

--kiểm tra dữ liệu
select * from [Order Details]

--viết batch
while (select avg(UnitPrice) from [Order Details]) < 50

```



```
begin
    update [Order Details]
    set UnitPrice = UnitPrice * 1.10
    if(select max(UnitPrice) from [Order Details]) > 300
        break
    else
        continue
end
```

```
--Kiểm tra dữ liệu thực thi
select * from [Order Details]
GO
```