

## QUẢN LÝ BÁN HÀNG

### Mô Hình Cơ Sở Dữ Liệu :

#### Danh mục Vật tư (VATTU)

Tên cột	Ý nghĩa	Kiểu	Độ rộng
<u>MaVTu</u>	Mã vật tư	Char	4
TenVTu	Tên vật tư	VarChar	100
DvTinh	Đơn vị tính	VarChar	10
PhanTram	Tỷ lệ phần trăm	Real	

#### Danh mục Nhà cung cấp (NHACC)

Tên cột	Ý nghĩa	Kiểu	Độ rộng
<u>MaNhaCc</u>	Mã nhà cung cấp	Char	3
TenNhaCc	Tên nhà cung cấp	VarChar	100
Diachi	Địa chỉ	Varchar	200
DiienThoai	Điện thoại	Varchar	20

#### Đơn đặt hàng (DONDH)

Tên cột	Ý nghĩa	Kiểu	Độ rộng
<u>SoDh</u>	Số đơn đặt hàng	Char	4
NgayDh	Ngày đặt hàng, ngày	DateTime	
MaNhaCc	Mã nhà cung cấp	Char	3

#### Chi tiết Đơn đặt hàng (CTDONDH)

Tên cột	Ý nghĩa	Kiểu	Độ rộng
<u>SoDh</u>	Số đơn đặt hàng	Char	4
<u>MaVTu</u>	Mã vật tư	Char	4
SiDat	Số lượng đặt hàng	Int	

#### Phiếu nhập hàng (PNHAP)

Tên cột	Ý nghĩa	Kiểu	Độ rộng
<u>SoPn</u>	Số phiếu nhập hàng	Char	4
NgayNhap	Ngày nhập hàng	Datetime	
SoDh	Số đơn đặt hàng	Char	4

#### Chi tiết nhập hàng (CTPNHAP)

Tên cột	Ý nghĩa	Kiểu	Độ rộng
<u>SoPn</u>	Số phiếu nhập hàng	Char	4
<u>MaVTu</u>	Mã vật tư	Char	4
SiNhap	Số lượng nhập hàng	Int	
DgNhap	Đơn giá nhập hàng	Money	

#### Phiếu xuất hàng (PXUAT)

Tên cột	Ý nghĩa	Kiểu	Độ rộng
<u>SoPx</u>	Số phiếu xuất	Char	4
NgayXuat	Ngày xuất hàng	Datetime	
TenKh	Tên khách hàng	Varchar	100

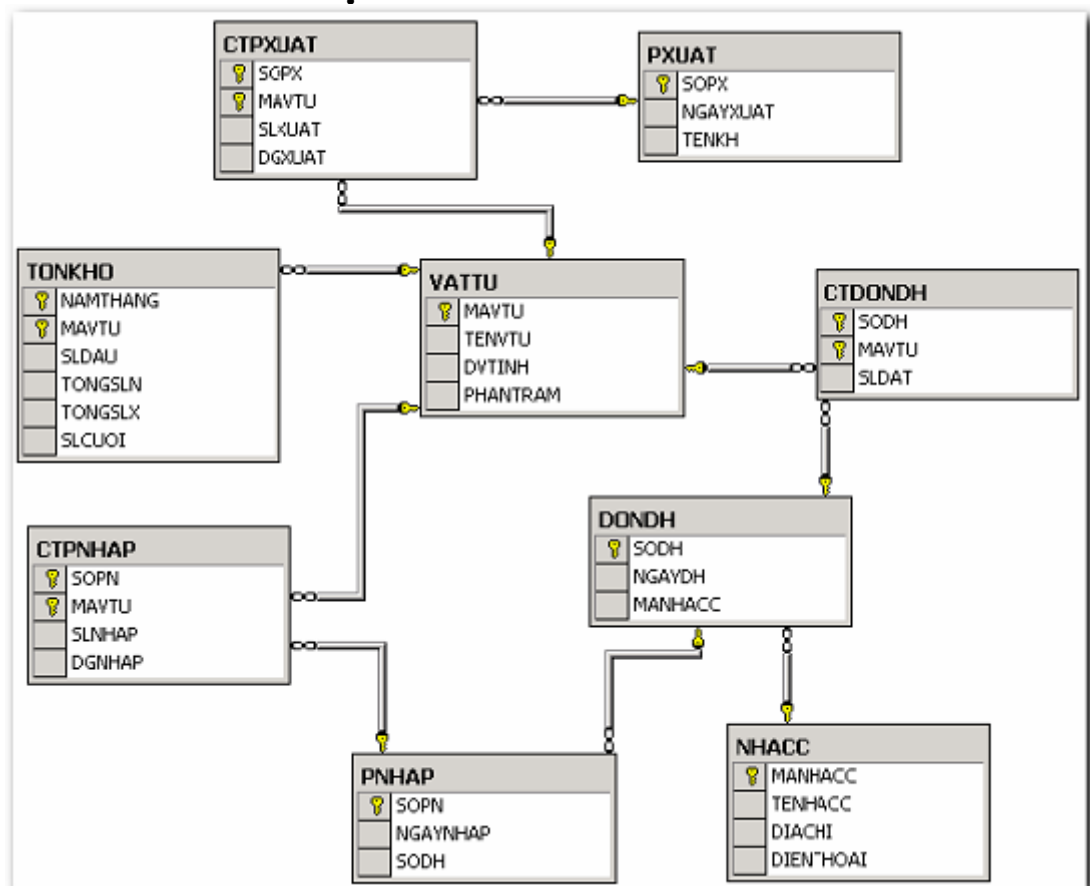
**Chi tiết xuất hàng (CTPXUAT)**

Tên cột	Ý nghĩa	Kiểu	Độ rộng
<u>SoPx</u>	Số phiếu xuất	Char	4
<u>MaVTu</u>	Mã vật tư	Char	4
SLXuat	Số lượng xuất hàng	Int	
DgXuat	Đơn giá xuất hàng	Money	

**Tồn kho (TONKHO)**

Tên cột	Ý nghĩa	Kiểu	Độ rộng
<u>NamThang</u>	Năm tháng	Char	6
<u>MaVTu</u>	Mã vật tư	Char	4
SLDau	Số lượng tồn đầu kỳ	Int	
TongSLN	Tổng số lượng nhập trong kỳ	Int	
TongSLX	Tổng số lượng xuất trong kỳ	Int	
SLCui	Số lượng tồn cuối kỳ	Int	

**Mô Hình Liên Kết Thực Thể :**



## I, Thiết Lập Cơ Sở Dữ Liệu :

### 1. Tạo CSDL Bán Hàng

*CREATE DATABASE BanHang*

*ON PRIMARY*

```
(
    NAME          =BanHang_Data,
    FILENAME       = 'C:\Program Files\Microsoft SQL Server\MSSQL\Data\BanHang_Data.mdf',
    SIZE           =10,
    MAXSIZE        =20,
    FILEGROWTH     =10%
)
LOG ON
(
    NAME          =BanHang_Log,
    FILENAME       = 'C:\Program Files\Microsoft SQL Server\MSSQL\Data\BanHang_Log.ldf',
    SIZE           =10,
    MAXSIZE        =20,
    FILEGROWTH     =10%
)
GO
USE BanHang
```

### 2. Tạo Bảng Vật Tư

*CREATE TABLE VATTU*

```
(
    MaVTu          CHAR(4),
    TenVTu         VARCHAR(100),
    DVTinh         VARCHAR(10) DEFAULT "",
    PhanTram       REAL
    CONSTRAINT PRK_VatTu_MaVTu PRIMARY KEY(MaVTu),
    CONSTRAINT UNQ_VatTu_TenVTu UNIQUE(TenVTu),
    CONSTRAINT CHK_VatTu_PhanTram CHECK(PhanTram BETWEEN 0 AND 100)
)
```

### 3. Tạo Bảng Nhà Cung Cấp

*CREATE TABLE NHACC*

```
(
    MaNhaCC        CHAR(3),
    TenNhaCC       VARCHAR(100),
    DiaChi         VARCHAR(200),
    DienThoai      VARCHAR(20) DEFAULT 'Chua co'
    CONSTRAINT PRK_NhaCC_MaNhaCC PRIMARY KEY(MaNhaCC),
    CONSTRAINT UNQ_NhaCC_MaNhaCC_TenNhaCC UNIQUE(TenNhaCC)
)
```

### 4. Tạo Bảng Đơn Đặt Hàng

*CREATE TABLE DONDH*

```
(
```

```
SoDH      CHAR(4),
MaNhaCC   CHAR(3),
NgayDH    DATETIME DEFAULT GETDATE(),
CONSTRAINT PRK_DonDH_SoDH PRIMARY KEY(SoDH)
)
```

### **5. Tạo Bảng Chi Tiết Đơn Đặt Hàng**

**CREATE TABLE CTDONDH**

```
(
    SoDH      CHAR(4),
    MaVTu     CHAR(4),
    SLDat     INT
    CONSTRAINT PRK_CTDONDH_SoDH_MaVTu PRIMARY KEY(SoDH, MaVTu),
    CONSTRAINT CHK_CTDONDH_SLDat CHECK(SLDat > 0)
)
```

### **6. Tạo Bảng Phiếu Nhập**

**CREATE TABLE PNHAP**

```
(
    SoPN      CHAR(4),
    SoDH      CHAR(4),
    NgayNhap  DATETIME
    CONSTRAINT PRK_PNHAP_SoPN PRIMARY KEY(SoPN)
)
```

### **7. Tạo Bảng Chi Tiết Phiếu Nhập**

**CREATE TABLE CTPNHAP**

```
(
    SoPN      CHAR(4),
    MaVTu     CHAR(4),
    SLNhap    INT,
    DGNhap    MONEY
    CONSTRAINT PRK_CTPNHAP_SoPN_MaVTu PRIMARY KEY(SoPN, MaVTu),
    CONSTRAINT CHK_CTPNHAP_SLNhap CHECK(SLNhap > 0 AND DGNhap > 0)
)
```

### **8. Tạo Bảng Phiếu Xuất**

**CREATE TABLE PXUAT**

```
(
    SoPX      CHAR(4),
    NgayXuat  DATETIME,
    TenKH     VARCHAR(100)
    CONSTRAINT PRK_PXUAT_SoPX PRIMARY KEY(SoPX)
)
```

### **9. Tạo Bảng Chi Tiết Phiếu Xuất**

**CREATE TABLE CTPXUAT**

```
(
    SoPX      CHAR(4),
```

```
MaVTu      CHAR(4),
SLXuat     INT,
DGXuat     MONEY
CONSTRAINT PRK_CTPXUAT_SoPX_MaVTu PRIMARY KEY(SoPX,MaVTu),
CONSTRAINT CHK_CTPXUAT_SLXuat_DGXuat CHECK(SLXuat > 0 AND DGXuat > 0)
)
```

## 10. Tạo Bảng Tồn Kho

```
CREATE TABLE TONKHO
(
    NamThang CHAR(6),
    MaVTu     CHAR(4),
    SLDau     INT DEFAULT 0,
    TongSLN   INT DEFAULT 0,
    TongSLX   INT DEFAULT 0,
    SLCuoi    AS SLDau + TongSLN - TongSLX
    CONSTRAINT PRK_TONKHO_NamThang_MaVTu PRIMARY KEY(NamThang,MaVTu),
    CONSTRAINT CHK_TONKHO_SLDau_TongSLN_TongSLX
        CHECK(SLDau >= 0 AND TongSLN >= 0 AND TongSLX >= 0)
)
```

## 11. Tạo Liên Kết Giữa Các Bảng Dữ Liệu

```
ALTER TABLE DONDH ADD CONSTRAINT FRK_DonDH_NhaCC_MaNhaCC
    FOREIGN KEY (MaNhaCC) REFERENCES NhaCC(MaNhaCC)
ALTER TABLE CTDONDH ADD CONSTRAINT FRK_CTDONDH_VATTU_MaVTu
    FOREIGN KEY (MaVTu) REFERENCES VATTU(MaVTu)
ALTER TABLE CTDONDH ADD CONSTRAINT FRK_CTDONDH_DONDH_SoDH
    FOREIGN KEY (SoDH) REFERENCES DONDH(SoDH)
ALTER TABLE PNHAP ADD CONSTRAINT FRK_PNhap_DonDH_SoDH
    FOREIGN KEY (SoDH) REFERENCES DONDH(SoDH)
ALTER TABLE CTPNHAP ADD CONSTRAINT FRK_CTPNHAP_VATTU_MaVTu
    FOREIGN KEY (MaVTu) REFERENCES VATTU(MaVTu)
ALTER TABLE CTPNHAP ADD CONSTRAINT FRK_CTPNHAP_PNHAP_SoPN
    FOREIGN KEY (SoPN) REFERENCES PNHAP(SoPN)
ALTER TABLE CTPXUAT ADD CONSTRAINT FRK_CTPXUAT_VATTU_MaVTu
    FOREIGN KEY (MaVTu) REFERENCES VATTU(MaVTu)
ALTER TABLE CTPXUAT ADD CONSTRAINT FRK_CTPXUAT_PXUAT_SoPX
    FOREIGN KEY (SoPX) REFERENCES PXUAT(SoPX)
ALTER TABLE TONKHO ADD CONSTRAINT FRK_TONKHO_VATTU_MaVTu
    FOREIGN KEY (MaVTu) REFERENCES VATTU(MaVTu)
GO
```

## II, Chèn Dữ Liệu Vào Các Bảng :

### 1. Chèn bảng Nhà Cung Cấp

```
INSERT INTO NHACC (MaNhaCC,TenNhaCC,DiaChi,DienThoai) VALUES ('C01','Bui Tien
Truong','Xuan La, Tay Ho, Ha Noi','0989995221')
INSERT INTO NHACC (MaNhaCC,TenNhaCC,DiaChi,DienThoai) VALUES ('C02','Nguyen
Thi Thu','Quan La, Tay Ho, Ha Noi','0979012300')
```

```
INSERT INTO NHACC (MaNhaCC,TenNhaCC,DiaChi,DienThoai) VALUES ('C03','Ngo
Thanh Tung','Kim Lien, Dong Da','0988098591')
INSERT INTO NHACC (MaNhaCC,TenNhaCC,DiaChi,DienThoai) VALUES ('C04','Bui Tien
Lap','Ha Noi','0904255934')
INSERT INTO NHACC (MaNhaCC,TenNhaCC,DiaChi,DienThoai) VALUES ('C05','Hong
That Cong','Ha Noi','chua co')
INSERT INTO NHACC (MaNhaCC,TenNhaCC,DiaChi,DienThoai) VALUES ('C07','Bui Duc
Kien','To 11, Cum 2, Xuan La','0437530097')
```

## **2. Chèn bảng Vật Tư**

```
INSERT INTO VATTU (MaVTu,TenVTu,DVTinh,PhanTram) VALUES ('DD01','Dau DVD Hitachi 1 dia','Bo',40)
INSERT INTO VATTU (MaVTu,TenVTu,DVTinh,PhanTram) VALUES ('DD02','Dau DVD Hitachi 3 dia','Bo',40)
INSERT INTO VATTU (MaVTu,TenVTu,DVTinh,PhanTram) VALUES ('TL15','Tu lanh Sanyo 150 lit','Cai',25)
INSERT INTO VATTU (MaVTu,TenVTu,DVTinh,PhanTram) VALUES ('TL90','Tu lanh Sanyo 90 lit','Cai',20)
INSERT INTO VATTU (MaVTu,TenVTu,DVTinh,PhanTram) VALUES ('TV14','Tivi Sony 14 inches','Cai',15)
INSERT INTO VATTU (MaVTu,TenVTu,DVTinh,PhanTram) VALUES ('TV21','Tivi Sony 21 inches','Cai',10)
INSERT INTO VATTU (MaVTu,TenVTu,DVTinh,PhanTram) VALUES ('TV29','Tivi Sony 29 inches','Cai',10)
INSERT INTO VATTU (MaVTu,TenVTu,DVTinh,PhanTram) VALUES ('VD01','Dau VCD Sony 1 dia','Bo',30)
INSERT INTO VATTU (MaVTu,TenVTu,DVTinh,PhanTram) VALUES ('VD02','Dau VCD Sony 3 dia','Bo',30)
```

## **3. Chèn bảng Đơn Đặt Hàng**

```
INSERT INTO DONDH(SoDH,MaNhaCC,NgayDH) VALUES ('D001','C03','01/15/2002')
INSERT INTO DONDH(SoDH,MaNhaCC,NgayDH) VALUES ('D002','C01','01/30/2002')
INSERT INTO DONDH(SoDH,MaNhaCC,NgayDH) VALUES ('D003','C02','02/10/2002')
INSERT INTO DONDH(SoDH,MaNhaCC,NgayDH) VALUES ('D004','C05','02/17/2002')
INSERT INTO DONDH(SoDH,MaNhaCC,NgayDH) VALUES ('D005','C02','03/01/2002')
INSERT INTO DONDH(SoDH,MaNhaCC,NgayDH) VALUES ('D006','C05','03/12/2002')
```

## **4. Chèn bảng Phiếu Nhập**

```
INSERT INTO PNHAP(SoPN,SoDH,NgayNhap) VALUES ('N001','D001','01/17/2002')
INSERT INTO PNHAP(SoPN,SoDH,NgayNhap) VALUES ('N002','D001','01/20/2002')
INSERT INTO PNHAP(SoPN,SoDH,NgayNhap) VALUES ('N003','D002','01/31/2002')
INSERT INTO PNHAP(SoPN,SoDH,NgayNhap) VALUES ('N004','D003','02/15/2002')
```

## **5. Chèn bảng Chi Tiết Đơn Đặt Hàng**

```
INSERT INTO CTDONDH(SoDH,MaVTu,SLDat) VALUES('D001','DD01',10)
INSERT INTO CTDONDH(SoDH,MaVTu,SLDat) VALUES('D001','DD02',15)
INSERT INTO CTDONDH(SoDH,MaVTu,SLDat) VALUES('D002','VD02',30)
INSERT INTO CTDONDH(SoDH,MaVTu,SLDat) VALUES('D003','TV14',10)
INSERT INTO CTDONDH(SoDH,MaVTu,SLDat) VALUES('D003','TV29',20)
INSERT INTO CTDONDH(SoDH,MaVTu,SLDat) VALUES('D004','TL90',10)
INSERT INTO CTDONDH(SoDH,MaVTu,SLDat) VALUES('D005','TV14',10)
INSERT INTO CTDONDH(SoDH,MaVTu,SLDat) VALUES('D005','TV29',20)
INSERT INTO CTDONDH(SoDH,MaVTu,SLDat) VALUES('D006','TV14',10)
INSERT INTO CTDONDH(SoDH,MaVTu,SLDat) VALUES('D006','TV29',20)
INSERT INTO CTDONDH(SoDH,MaVTu,SLDat) VALUES('D005','VD01',20)
```

## **6. Chèn bảng Chi Tiết Phiếu Nhập**

```
INSERT INTO CTPNHAP(SoPN,MaVTu,SLNhap,DGNhap) VALUES('N001','DD01',8,2500000)
INSERT INTO CTPNHAP(SoPN,MaVTu,SLNhap,DGNhap) VALUES('N001','DD02',10,3500000)
INSERT INTO CTPNHAP(SoPN,MaVTu,SLNhap,DGNhap) VALUES('N002','DD01',2,2500000)
```

```
INSERT INTO CTPNHAP(SoPN,MaVTu,SLNhap,DGNhap) VALUES('N002','DD02',5,3500000)
INSERT INTO CTPNHAP(SoPN,MaVTu,SLNhap,DGNhap) VALUES('N003','VD02',30,2500000)
INSERT INTO CTPNHAP(SoPN,MaVTu,SLNhap,DGNhap) VALUES('N004','TV14',5,2500000)
INSERT INTO CTPNHAP(SoPN,MaVTu,SLNhap,DGNhap) VALUES('N004','TV29',12,3500000)
```

### 7. Chèn bảng Phiếu Xuất

```
INSERT INTO PXUAT(SoPX,NgayXuat,TenKH) VALUES('X001','01/17/2002','Duong Minh Chau')
INSERT INTO PXUAT(SoPX,NgayXuat,TenKH) VALUES('X002','01/25/2002','Nguyen Kim Dung')
INSERT INTO PXUAT(SoPX,NgayXuat,TenKH) VALUES('X003','01/31/2002','Nguyen Tien Dung')
```

### 8. Chèn bảng Chi Tiết Phiếu Xuất

```
INSERT INTO CTPXUAT(SoPX,MaVTu,SLXuat,DGXuat) VALUES('X001','DD01',2,3500000)
INSERT INTO CTPXUAT(SoPX,MaVTu,SLXuat,DGXuat) VALUES('X002','DD01',1,3500000)
INSERT INTO CTPXUAT(SoPX,MaVTu,SLXuat,DGXuat) VALUES('X002','DD02',5,4900000)
INSERT INTO CTPXUAT(SoPX,MaVTu,SLXuat,DGXuat) VALUES('X003','DD01',3,3500000)
INSERT INTO CTPXUAT(SoPX,MaVTu,SLXuat,DGXuat) VALUES('X003','DD02',2,4900000)
INSERT INTO CTPXUAT(SoPX,MaVTu,SLXuat,DGXuat) VALUES('X003','VD02',10,3250000)
```

### 9. Chèn bảng Tồn Kho

```
INSERT INTO TONKHO(NamThang,MaVTu,SLDau,TongSLN,TongSLX) VALUES('200201','DD01',0,10,6)
INSERT INTO TONKHO(NamThang,MaVTu,SLDau,TongSLN,TongSLX) VALUES('200201','DD02',0,15,7)
INSERT INTO TONKHO(NamThang,MaVTu,SLDau,TongSLN,TongSLX) VALUES('200201','VD02',0,30,10)
INSERT INTO TONKHO(NamThang,MaVTu,SLDau,TongSLN,TongSLX) VALUES('200202','DD01',4,0,0)
INSERT INTO TONKHO(NamThang,MaVTu,SLDau,TongSLN,TongSLX) VALUES('200202','DD02',8,0,0)
INSERT INTO TONKHO(NamThang,MaVTu,SLDau,TongSLN,TongSLX) VALUES('200202','VD02',20,0,0)
INSERT INTO TONKHO(NamThang,MaVTu,SLDau,TongSLN,TongSLX) VALUES('200202','TV14',5,0,0)
INSERT INTO TONKHO(NamThang,MaVTu,SLDau,TongSLN,TongSLX) VALUES('200202','TV29',12,0,0)
```

## III, Bài Tập VIEW :

### 1. Tạo view vw\_DMVT gồm (MaVTu và TenVTu) dùng để liệt kê danh sách các vật tư hiện có trong bảng VATTU :

```
CREATE VIEW vw_DMVT
AS
    SELECT MaVTu,TenVTu
    FROM VatTu
```

```
SELECT * FROM vw_DMVT
```

### 2. Tạo view vw\_DonDH\_TongSLDatNhap gồm (SoHD, TongSLDat và TongSLNhap) dùng để thống kê những đơn đặt hàng đã được nhập hàng đầy đủ :

```
CREATE VIEW vw_DonDH_TongSLDatNhap (SoDH,TongSLDat,TongSLNhap)
AS
    SELECT DH.SoDH,SUM(SLDat),SUM(SLNhap)
    FROM DONDH DH LEFT JOIN (PNHAP PN INNER JOIN CTPNHAP CTPN ON
    PN.SoPN=CTPN.SoPN) ON DH.SoDH=PN.SoDH LEFT JOIN CTDONDH ON
    DH.SoDH=CTDONDH.SoDH
```

*GROUP BY DH.SoDH*

*SELECT \* FROM vw\_DonDH\_TongSLDatNhap*

**3. Tạo view vw\_DonDH\_DaNhapDu gồm (SoHD, DaNhapDu) có hai giá trị là ‘Da Nhap Du’ nếu đơn hàng đó đã nhập đủ hoặc ‘Chưa Nhap Du’ nếu đơn đặt hàng đó chưa nhập đủ :**

```
CREATE VIEW vw_DonDH_DaNhapDu
AS
    SELECT SoDH,DaNhapDu = CASE
        WHEN TongSLDat=TongSLNhap
        THEN 'Da nhap du'
        ELSE
        'Chua nhap du'
    END
    FROM vw_DonDH_TongSLDatNhap
```

*SELECT \* FROM vw\_DonDH\_DaNhapDu*

**4. Tạo view vw\_TongNhap gồm (NamThang, MaVTu và TongSLNhap) dùng để thống kê số lượng nhập của các vật tư trong từng năm tháng tương ứng.(không sử dụng tồn kho) :**

```
CREATE VIEW vw_TongNhap (NamThang,MaVTu,TongSLNhap)
AS
    SELECT CONVERT(CHAR(6),NgayNhap,112),MaVTu,SUM(SLNhap)
    FROM PNHAP PN INNER JOIN CTPNHAP CTPN ON PN.SoPN=CTPN.SoPN
    GROUP BY CONVERT(CHAR(6),NgayNhap,112),MaVTu
```

*SELECT \* FROM vw\_TongNhap*

**5. Tạo view vw\_TongXuat gồm (NamThang, MaVTu và TongSLXuat) dùng để thống kê SL xuất của vật tư trong từng năm tháng tương ứng.(không sử dụng TONKHO) :**

```
CREATE VIEW vw_TongXuat (NamThang,MaVatTu,TongSLXuat)
AS
    SELECT CONVERT(CHAR(6),NgayXuat,112),MaVTu,SUM(SLXuat)
    FROM PXUAT PX INNER JOIN CTPXUAT CTPX ON PX.SoPX=CTPX.SoPX
    GROUP BY CONVERT(CHAR(6),NgayXuat,112),MaVTu
```

*SELECT \* FROM vw\_TongXuat*

**6. Tạo view vw\_DonDH\_MaVTu\_TongSLNhap gồm (SoHD, NgayHD, MaVTu, TenVTu, SLDat và TongSLDaNhap) :**



```
CREATE VIEW vw_DonDH_MaVTu_TongSLNhap
(SoDH,NgayDH,MaVatTu,TenVatTu,SLDat,TongSLDaNhap)
AS
    SELECT DH.SoDH,NgayDH,CTDONDH.MaVTu,TenVTu,SUM(SLDat),SUM(SLNhap)
    FROM DONDH DH LEFT JOIN (PNHAP PN INNER JOIN CTPNHAP CTPN ON
    PN.SoPN=CTPN.SoPN)ON DH.SoDH=PN.SoDH
    LEFT JOIN CTDONDH ON DH.SoDH=CTDONDH.SoDH INNER JOIN
    VATTU ON CTDONDH.MaVTu=VATTU.MaVTu
    GROUP BY DH.SoDH,NgayDH,CTDONDH.MaVTu,VATTU.TenVTu

SELECT * FROM vw_DonDH_MaVTu_TongSLNhap
```

#### **IV, Bài Tập Thống Kê :**

##### **1. Danh sách các phiếu đặt hàng chưa được nhập hàng :**

```
SELECT      DISTINCT SoDH
FROM        vw_DonDH_MaVTu_TongSLNhap
WHERE       TongSLDaNhap IS NULL
```

Kết quả
---------

##### **2. Danh sách các mặt hàng chưa được đặt hàng bao giờ :**

```
SELECT      MaVTu,TenVTu
FROM        VATTU
WHERE       MaVTu NOT IN (SELECT DISTINCT MaVTu FROM CTDONDH)
```

##### **3. Nhà cung cấp nào có nhiều đơn đặt hàng nhất :**

```
SELECT      TOP 1 WITH TIES NCC.MaNhaCC,TenNhaCC,COUNT(DH.MaNhaCC) AS
SoLanDH
FROM        NHACC NCC INNER JOIN DONDH DH ON NCC.MaNhaCC=DH.MaNhaCC
GROUP BY   NCC.MaNhaCC,TenNhaCC
ORDER BY   SoLanDH DESC
```

##### **4. Vật tư nào có tổng số lượng xuất bán nhiều nhất :**

```
SELECT      TOP 1 WITH TIES VT.MaVTu,TenVTu,SUM(SLXuat)AS TongXuat
FROM        VATTU VT INNER JOIN CTPXUAT CTPX ON VT.MaVTu=CTPX.MaVTu
GROUP BY   VT.MaVTu,TenVTu
ORDER BY   TongXuat DESC
```

##### **5. Cho biết đơn đặt hàng nào có nhiều mặt hàng nhất :**

```
SELECT      TOP 1 WITH TIES SoDH,COUNT(MaVTu)AS SoMatHang
FROM        CTDONDH
GROUP BY   SoDH
ORDER BY   SoMatHang DESC
```

##### **6. Tạo View vw\_truc\_TG báo cáo Tình hình xuất nhập vật tư :**

```
CREATE VIEW vw_truc_TG
AS
    SELECT CONVERT(VARCHAR(10),NgayNhap,105) AS
    NgayThang,MaVTu,SUM(SLNhap)AS TongNhap, 0 AS TongXuat
    FROM PNHAP PN INNER JOIN CTPNHAP CTPN ON PN.SoPN=CTPN.SoPN
    GROUP BY NgayNhap,MaVTu

    UNION

    SELECT CONVERT(VARCHAR(10),NgayXuat,105),MaVTu, 0 AS TongXuat,
    SUM(SLXuat)AS TongNhap
    FROM PXUAT PX INNER JOIN CTPXUAT CTPX ON PX.SoPX=CTPX.SoPX
    GROUP BY NgayXuat,MaVTu

SELECT NgayThang,MaVTu,SUM(TongNhap)AS TongNhap,SUM(TongXuat)AS TongXuat
FROM vw_truc_TG GROUP BY NgayThang,MaVTu ORDER BY NgayThang
```

### **7. Cho biết tình hình nhập và đặt hàng :**

```
SELECT DH.SoDH,CTDONDH.MaVTu,TenVTu,SUM(SLDat)AS SLDat,SUM(SLNhap)AS
SLNhap
FROM DONDH DH LEFT JOIN (PNHAP PN INNER JOIN CTPNHAP CTPN ON
PN.SoPN=CTPN.SoPN)ON DH.SoDH=PN.SoDH LEFT JOIN CTDONDH ON
DH.SoDH=CTDONDH.SoDH INNER JOIN VATTU ON CTDONDH.MaVTu=VATTU.MaVTu
GROUP BY DH.SoDH,CTDONDH.MaVTu,TenVTu
```

### **8. Thống kê tình hình đặt hàng :**

```
SELECT NgayDH,CTDH.MaVTu,TenVTu,SUM(SLDat) AS TongSLDat
FROM DONDH DH INNER JOIN CTDONDH CTDH ON DH.SoDH=CTDH.SoDH INNER
JOIN VATTU VT ON CTDH.MaVTu=VT.MaVTu
GROUP BY NgayDH,CTDH.MaVTu,TenVTu
```

### **9. Thống kê tình hình nhập hàng :**

```
SELECT NgayNhap,CTPN.MaVTu,TenVTu,SUM(SLNhap) AS TongSLNhap
FROM PNHAP PN INNER JOIN CTPNHAP CTPN ON PN.SoPN=CTPN.SoPN INNER JOIN
VATTU VT ON CTPN.MaVTu=VT.MaVTu
GROUP BY NgayNhap,CTPN.MaVTu,TenVTu
```

### **10. Thống kê những đơn đặt hàng chưa nhập đủ số lượng :**

```
SELECT DISTINCT SoDH
FROM vw_DonDH_MaVTu_TongSLNhap
WHERE SLDat > TongSLDaNhap
```

## **V, Bài Tập PROCEDURE :**

**1. Xây dựng thủ tục tính SL đặt hàng với tên *Spud\_DonDH\_TinhSLDat* với 2 tham số vào là SoHD, MaVTu và 1 tham số ra là SL đặt của mỗi vật tư trong 1 số đặt hàng :**

```
CREATE PROCEDURE Spud_DonDH_TinhSLDat
    @So      CHAR(4),
    @Ma      CHAR(4),
    @SL      INT OUTPUT
AS
    SELECT      @SL = SLDat
    FROM        CTDONDH
    WHERE       SoDH = @So AND MaVTu = @Ma
GO

DECLARE      @Ra  INT
EXECUTE      Spud_DonDH_TinhSLDat 'D001','DD02',@Ra OUTPUT
PRINT        @Ra
```

*SELECT \* FROM CTDONDH*

**2. Xây dựng thủ tục tính tổng số lượng đã nhập hàng với tên *Spud\_PNHAP\_TinhTongSLNHang* với 2 tham số vào là SoHD, MaVTu và 1 tham số ra là Tổng số lượng hàng của một vật tư trong một số đặt hàng :**

```
CREATE PROCEDURE Spud_PNHAP_TinhTongSLNHang
    @PN      CHAR(4),
    @Ma      CHAR(4),
    @Tong     INT OUTPUT
AS
    SELECT      @Tong =SUM(SLNhap)
    FROM        CTPNHAP
    WHERE       SoPN = @PN AND MaVTu = @Ma
GO

DECLARE      @Ra  INT
EXECUTE      Spud_PNHAP_TinhTongSLNHang 'N001','DD01',@Ra OUTPUT
PRINT        @Ra
```

*SELECT \* FROM CTPNHAP*

**3. Xây dựng thủ tục tính số lượng tồn kho cuối kỳ của 1 vật tư với tên *Spud\_TonKho\_TinhTonCuoi* với 2 tham số vào là NamThang, MaVTu và 1 tham số ra là số lượng cuối kỳ của một vật tư trong năm tháng truyền vào :**

```
CREATE PROCEDURE Spud_TonKho_TinhTonCuoi
    @NT      CHAR(6),
    @Ma      CHAR(4),
    @SLC     INT OUTPUT
AS
```

```
SELECT      @SLC = SLCuoi
FROM        TONKHO
WHERE       NamThang = @NT AND MaVTu = @Ma
GO
DECLARE     @Ra INT
EXECUTE     Spud_TonKho_TinhTonCuoi '200201','VD01',@Ra OUTPUT
PRINT      @Ra
```

```
SELECT * FROM TONKHO
```

**4. Xây dựng thủ tục thêm dữ liệu vào bảng VATTU với tên *Spud\_VatTu\_Them* với 4 tham số vào chính là MaVTu, TenVTu, DVTinh, PhanTram (MaVTu phải duy nhất). Kiểm tra ràng buộc dữ liệu phải hợp lệ trước khi thực hiện INSERT :**

```
CREATE PROCEDURE Spud_VatTu_Them
    @Ma      CHAR(4),
    @Ten     VARCHAR(100),
    @DVT     VARCHAR(10),
    @PT      REAL
AS
    IF EXISTS(SELECT*FROM VATTU WHERE MaVTu = @Ma)
        BEGIN
            PRINT 'Ma vat tu bi trung'
            RETURN
        END
    IF @PT < 0 OR @PT > 100
        BEGIN
            PRINT 'Vuot qua phan tram quy dinh'
            RETURN
        END
    INSERT INTO VATTU VALUES(@Ma ,@Ten ,@DVT ,@PT)
GO
```

```
SELECT * FROM VATTU
```

```
EXECUTE Spud_VatTu_Them 'DD01','Gach','Vien',56
EXECUTE Spud_VatTu_Them 'GH01','Gach','Vien',120
```

**5. Xây dựng thủ tục xóa 1 vật tư có trong bảng VATTU với tên *Spud\_VatTu\_Xoa* với tham số vào chính là MaVTu cần xóa (MaVTu phải chưa có trong CTDONDH,CTPXUAT,TONKHO). Kiểm tra ràng buộc dữ liệu trước khi thực hiện lệnh DELETE :**

```
CREATE PROCEDURE Spud_VatTu_Xoa
    @MaV     CHAR(4)
AS
    IF EXISTS (SELECT * FROM CTDONDH WHERE MaVTu = @MaV)
```

```
BEGIN
    PRINT 'Khong xoa duoc vi ma vat tu nay co trong CTDONDH'
    RETURN
END
IF EXISTS (SELECT * FROM CTPNHAP WHERE MaVTu = @MaV)
    BEGIN
        PRINT 'Khong xoa duoc vi ma vat tu nay co trong CTPNHAP'
        RETURN
    END
IF EXISTS (SELECT * FROM CTPXUAT WHERE MaVTu = @MaV)
    BEGIN
        PRINT 'Khong xoa duoc vi ma vat tu nay co trong CTPXUAT'
        RETURN
    END
IF EXISTS (SELECT * FROM TONKHO WHERE MaVTu = @MaV)
    BEGIN
        PRINT 'Khong xoa duoc vi ma vat tu nay co trong TONKHO'
        RETURN
    END
DELETE FROM VATTU WHERE MaVTu = @MaV
```

```
EXECUTE Spud_VatTu_Xoa 'TV21'
```

**6. Xây dựng thủ tục sửa đổi vật tư trong bảng VATTU với tên *Spud\_VatTu\_Sua* với 4 tham số vào chính là giá trị cần thay đổi của các cột trong bảng VATTU. Thủ tục chỉ cần thực hiện lệnh UPDATE SET với các giá trị tương ứng :**

```
CREATE PROCEDURE Spud_VatTu_Sua
    @MaV      CHAR(4),
    @TenV     VARCHAR(100),
    @DVT     VARCHAR(10),
    @PT      REAL
AS
    UPDATE    VATTU
    SET       TenVTu = @TenV,DVTinh = @DVT,PhanTram = @PT
    WHERE     MaVTu = @MaV
```

```
EXECUTE Spud_VatTu_Sua 'DD01','Dau DVD ','Chiec',20
```

```
SELECT * FROM VATTU
```

**7. Xây dựng thủ tục liệt kê các cột dữ liệu trong bảng VATTU với tên *Spud\_VatTu\_BCDSach*. Thủ tục này không có tham số. Hành động duy nhất là câu lệnh truy vấn *SELECT \* FROM VATTU* sắp xếp với TenVTu tăng dần :**

```
SELECT * FROM VATTU
```

```
CREATE PROCEDURE Spud_VatTu_BCDSach
AS
    SELECT      *
    FROM        VATTU
    ORDER BY    TenVTu
GO
```

```
EXECUTE Spud_VatTu_BCDSach
```

**8. Xây dựng thủ tục liệt kê các cột bên trong bảng TONKHO có thêm cột TenVTu trong bảng VATTU với tên *Spud\_TonKho\_BCTonKho* có tham số vào là NamThang muốn lọc dữ liệu :**

```
SELECT * FROM TONKHO
```

```
CREATE PROCEDURE Spud_TonKho_BCTonKho
    @NT CHAR(6)
AS
    SELECT TONKHO.NamThang, TONKHO.MaVTu, VATTU.TenVTu, TONKHO.SLDau,
           TONKHO.TongSLN, TONKHO.TongSLX, TONKHO.SLCuoi
    FROM TONKHO INNER JOIN VATTU ON TONKHO.MaVTu = VATTU.MaVTu
    WHERE TONKHO.NamThang = @NT
GO
```

```
EXECUTE Spud_TonKho_BCTonKho '200201'
```

**9. Xây dựng thủ tục liệt kê các cột bên trong PXUAT và CTXUAT có thêm cột TenVTu trong bảng VATTU với tên *Spud\_PXuat\_BCPXuat* có tham số vào là SoPX muốn lọc dữ liệu, có giá trị mặc định là NULL. Nếu lúc gọi thực hiện thủ tục mà không truyền giá trị SoPX vào thì thủ tục sẽ liệt kê tất cả các phiếu xuất có trong bảng PXUAT :**

```
SELECT * FROM PXUAT
SELECT * FROM CTPXUAT
```

```
SELECT      PXUAT.SoPX, PXUAT.NgayXuat, PXUAT.TenKH, CTPXUAT.MaVTu,
           VATTU.TenVTu, CTPXUAT.SLXuat, CTPXUAT.DGXuat
FROM        PXUAT INNER JOIN CTPXUAT ON PXUAT.SoPX = CTPXUAT.SoPX
           INNER JOIN VATTU ON CTPXUAT.MaVTu = VATTU.MaVTu
```

```
CREATE PROCEDURE Spud_PXuat_BCPXuat
    @So CHAR(4) = NULL
AS
    IF @So IS NULL
        SELECT PXUAT.SoPX, PXUAT.NgayXuat, PXUAT.TenKH, CTPXUAT.MaVTu,
               VATTU.TenVTu, CTPXUAT.SLXuat, CTPXUAT.DGXuat
        FROM PXUAT INNER JOIN CTPXUAT ON PXUAT.SoPX = CTPXUAT.SoPX
           INNER JOIN VATTU ON CTPXUAT.MaVTu = VATTU.MaVTu
```

```
ELSE
    SELECT PXUAT.SoPX,PXUAT.NgayXuat,PXUAT.TenKH,CTPXUAT.MaVTu,
           VATTU.TenVTu,CTPXUAT.SLXuat,CTPXUAT.DGXuat
    FROM PXUAT INNER JOIN CTPXUAT ON PXUAT.SoPX = CTPXUAT.SoPX
           INNER JOIN VATTU ON CTPXUAT.MaVTu = VATTU.MaVTu
    WHERE PXUAT.SoPX = @So
GO

EXECUTE Spud_PXuat_BCPXuat 'X003'
```

**10. Xây dựng thủ tục thêm mới dữ liệu vào bảng DONDH với tên *Spud\_DONDH\_Them* có 3 tham số vào chính là SoDH, MaNhaCC, NgayDH. Kiểm tra ràng buộc trước khi thực hiện lệnh INSERT (SoHD phải duy nhất, MaNhaCC phải có bên bảng NhaCC). NgayDH có thể không truyền vào khi đó sẽ lấy giá trị mặc định là ngày hiện hành :**

```
CREATE PROCEDURE Spud_DONDH_Them
    @So          CHAR(4),
    @Ma          CHAR(4),
    @Ngay        DATETIME = NULL
AS
    IF NOT EXISTS(SELECT MaNhaCC FROM NHACC WHERE MaNhaCC = @Ma)
        BEGIN
            PRINT 'Khong co ma nha cung cap nay'
            RETURN
        END
    IF EXISTS(SELECT SoDH FROM DONDH WHERE SoDH = @So)
        BEGIN
            PRINT'Da co so dat hang nay roi'
            RETURN
        END
    IF @Ngay IS NULL
        INSERT INTO DONDH VALUES(@So,@Ma,GETDATE())
    ELSE
        INSERT INTO DONDH VALUES(@So,@Ma,@Ngay)
GO

SELECT * FROM DONDH

EXECUTE Spud_DONDH_Them'D007','C04'
EXECUTE Spud_DONDH_Them'D008','C08'
EXECUTE Spud_DONDH_Them'D008','C04'
```

**11. Xây dựng thủ tục xóa DONDH với tên *Spud\_DonDH\_Xoa* có 1 tham số vào chính là SoDH cần xóa. Nếu ràng buộc dữ liệu hợp lệ thì tự động xóa các dòng dữ liệu liên quan bên bảng CTDONDH. Kiểm tra ràng buộc dữ liệu trước khi thực hiện DELETE (SoDH phải chưa có trong PNHAP) :**

```
CREATE PROCEDURE Spud_DonDH_Xoa
```

```
        @So CHAR(4)
AS
IF EXISTS(SELECT SoDH FROM PNHAP WHERE SoDH = @So)
    BEGIN
        PRINT 'Khong xoa duoc vi SoDH nay co trong PNHAP'
        RETURN
    END
ELSE
    BEGIN
        IF EXISTS(SELECT SoDH FROM CTDONDH WHERE SoDH = @So)
            DELETE FROM CTDONDH WHERE SoDH = @So
        IF EXISTS(SELECT SoDH FROM DONDH WHERE SoDH = @So)
            DELETE FROM DONDH WHERE SoDH = @So
    END

EXECUTE Spud_DonDH_Xoa 'D001'
EXECUTE Spud_DonDH_Xoa 'D008'
```

**12. Xây dựng thủ tục sửa đổi dữ liệu trong DONDH với tên *Spud\_DonDH\_Sua* gồm 3 tham số vào chính là các giá trị cần thay đổi của các cột SoDH, MaNhaCC, NgayDH. Kiểm tra ràng buộc dữ liệu trước khi thực hiện lệnh UPDATE SET. (MaNhaCC phải có trong NHACC, NgayDH phải trước NgayNhap nếu đơn đặt hàng đã được nhập về rồi.) :**

```
CREATE PROCEDURE Spud_DonDH_Sua
    @SoDH CHAR(4),
    @MaN CHAR(4),
    @NgayDat DATETIME
AS
    IF NOT EXISTS (SELECT * FROM NHACC WHERE MaNhaCC = @MaN)
        BEGIN
            PRINT 'Ma nha cung cap khong dung '
            RETURN
        END
    IF @NgayDat > (SELECT MIN(NgayNhap) FROM PNHAP WHERE SoDH = @SoDH)
        BEGIN
            PRINT 'Ngay dat hang phai truoc ngay nhap hang '
            RETURN
        END
    UPDATE DONDH
    SET MaNhaCC = @MaN , NgayDH = @NgayDat
    WHERE SoDH = @SoDH
```

**13. Xây dựng thủ tục thêm mới dữ liệu vào CTDONDH với tên *Spud\_CTDonDH\_Them* gồm 3 tham số vào chính là giá trị cần thay đổi của các cột trong bảng DONDH là SoDH, MaVTu và SLDat. Kiểm tra các ràng buộc dữ liệu phải hợp lệ trước khi thực hiện lệnh INSERT INTO (SoDh phải**



**có bên DONDH, MaVTu phải có bên VATTU). SoDH và MaVTu phải duy nhất trong bảng CTDONDH**

**14. Xây dựng thủ tục xoá CTDONDH với tên *Spud\_CTDonDH\_Xoa* gồm 2 tham số vào chính là SoDH và MaVTu cần xoá. Kiểm tra ràng buộc dữ liệu trước khi thực hiện lệnh DELETE. SoDH và MaVTu này phải chưa được nhập vào trong các bảng liên quan PNHAP và CTPNHAP :**

**15. Xây dựng thủ tục sửa đổi dữ liệu trong CTDONDH với tên *Spud\_CTDonDH\_Sua* gồm 3 tham số vào chính là giá trị cần thay đổi của các cột trong CTDONDH : SoDH, MaVTu và SLDat. Kiểm tra các ràng buộc dữ liệu phải hợp lệ trước khi thực hiện lệnh UPDATE SET. SoDH và MaVTu phải có bên CTDONDH. SoDH mới phải  $\geq$  Tổng số lượng đã nhập hàng (nếu đã có nhập hàng) :**

```
CREATE PROCEDURE Spud_CTDonDH_Sua
    @SoDH      CHAR(4),
    @MaV       CHAR(4),
    @SL        INT
AS
    DECLARE @TongSLN INT
    IF NOT EXISTS(SELECT * FROM CTDONDH WHERE SoDH = @SoDH)
        BEGIN
            PRINT 'So dat hang khong dung'
            RETURN
        END
    IF NOT EXISTS(SELECT * FROM CTDONDH WHERE SoDH = @SoDH AND
        MaVTu = @MaV)
        BEGIN
            PRINT 'Don dat hang khong co vat tu nay '
            RETURN
        END

    SELECT      @TongSLN = SUM(SLNhap)
    FROM        CTPNHAP
    WHERE       MaVTu = @MaV AND SoPN IN (SELECT SoPN
                                           FROM PNHAP
                                           WHERE SoDH = @SoDH )

    IF @SL < @TongSLN
        BEGIN
            PRINT 'so luong nhap phai lon hon hoac bang tong so luong da nhap '
            RETURN
        END

    UPDATE      CTDONDH
    SET         SLDat = @SL
    WHERE       SoDH = @SoDH AND MaVTu = @MaV

    SELECT      SUM(SLNhap)
    FROM        CTPNHAP
    WHERE       MaVTu = 'DD01' AND SoPN IN(SELECT SoPN
                                           FROM PNHAP
                                           WHERE SoDH = @SoDH )

    SELECT * FROM CTDONDH

EXECUTE Spud_CTDonDH_Sua 'D001','DD01',9
EXECUTE Spud_CTDonDH_Sua 'DD01','DD01',15
EXECUTE Spud_CTDonDH_Sua 'D001','DD11',15
EXECUTE Spud_CTDonDH_Sua 'D001','DD01',15
```

**16. Xây dựng thủ tục cập nhật bảng VATTU với 3 chức năng là Thêm (I), Sửa (U) và Xóa (D). Kiểm tra các ràng buộc khi thực hiện chức năng :**

```
CREATE PROCEDURE Pro_VatTu
    @Loai      CHAR(1),
    @MaV       CHAR(4) = NULL,
    @TenV      VARCHAR(100) = NULL,
    @DVT       VARCHAR(10) = NULL,
    @PT        REAL = NULL
AS
    IF @Loai = 'D'
        IF NOT EXISTS(SELECT * FROM CTPNHAP WHERE MaVTu = @MaV)
            AND NOT EXISTS(SELECT * FROM CTPXUAT WHERE MaVTu = @MaV)
            AND NOT EXISTS(SELECT * FROM TONKHO WHERE MaVTu = @MaV)
            AND NOT EXISTS(SELECT * FROM CTDONDH WHERE MaVTu = @MaV)
                DELETE FROM VATTU WHERE MaVTu = @MaV
            ELSE
                PRINT ' Khong xoa duoc vi ma vat tu co lien quan den cac bang khac '

    IF @Loai = 'I'
        IF NOT EXISTS(SELECT * FROM VATTU WHERE MaVTu = @MaV)
            AND (@PT >= 0 AND @PT <= 100)
            INSERT INTO VATTU VALUES(@MaV,@TenV,@DVT,@PT)
        ELSE
            PRINT ' Khong them duoc vi trung ma vat tu '

    IF @Loai = 'U'
        IF NOT EXISTS(SELECT * FROM VATTU WHERE MaVTu = @MaV)
            AND(@PT >=0 AND @PT <= 100)
            UPDATE VATTU
            SET MaVTu = @MaV,TenVTu = @TenV,DVTinh = @DVT,
            PhanTram = @PT
            WHERE MaVTu = @MaV
        ELSE
            PRINT ' Khong cho phép sua vi co lien quan den cac bang khac '

GO

SELECT * FROM TONKHO
SELECT * FROM CTPNHAP
SELECT * FROM CTPXUAT
SELECT * FROM CTDONDH

Pro_VatTu 'I','DD01','Dau DVD TQ ','Bo',35
Pro_VatTu 'D','DD01','Dau DVD TQ ','Bo',35
Pro_VatTu 'U','DD01','Dau DVD TQ ','Bo',35

EXECUTE Pro_VatTu 'D','DD01'
```

## **VI, Bài Tập TRIGGER :**

**1. Xây dựng trigger khi thêm mới dữ liệu vào bảng PNHAP với tên TG\_PNhap\_Them. Trong đó cần kiểm tra các ràng buộc dữ liệu phải hợp lệ : Ngày nhập hàng phải sau ngày đặt hàng**

```
CREATE TRIGGER TG_PNhap_Them
ON PNHAP FOR INSERT
AS
    DECLARE    @NgayN    DATETIME ,
               @NgayDH    DATETIME ,
               @SoDH    CHAR(4)

    SELECT    @NgayN    = NgayNhap ,
              @SoDH    = SoDH
    FROM      PNHAP

    SELECT    @NgayDH    = NgayDH
    FROM      DONDH
    WHERE     SoDH = @SoDH

    IF @NgayN < @NgayDH
        BEGIN
            PRINT ' Ngày nhập phải sau ngày dat'
            ROLLBACK TRANSACTION
        END

    SELECT * FROM DONDH

    INSERT INTO PNHAP VALUES('N005','D001','2002/01/16')
```

**2. Xây dựng trigger khi thêm mới dữ liệu vào bảng CTPNHAP với tên TG\_CTPNhap\_Them. Trong đó cần kiểm tra các ràng buộc dữ liệu phải hợp lệ : số lượng nhập hàng <= (số lượng đặt – tổng số lượng đã nhập vào trước đó) :**

```
CREATE TRIGGER TG_CTPNhap_Them
ON CTPNHAP FOR INSERT
AS
    DECLARE    @SLN    INT,
               @SLD    INT,
               @TongN    INT,
               @SoDH    CHAR(4)
    DECLARE    @MaV    CHAR(4),
               @SoPN    CHAR(4)

    SELECT    @SoDH    = SoDH
    FROM      PNHAP
    WHERE     SoPN    = (SELECT SoPN FROM INSERTED)
```

```
SELECT      @MaV      = MaVTu,
            @SLN      = SLNhap,
            @SoPN     = SoPN
FROM        INSERTED
```

```
SELECT      @SLD      = SLDat
FROM        CTDONDH
WHERE       SoDH = @SoDH AND MaVTu = @MaV
```

```
SELECT      @TongN     = SUM(SLNhap)
FROM        CTPNHAP
WHERE       SoPN <> @SoPN AND MaVTu = @MaV AND SoPN IN (SELECT
SoPN FROM PNHAP WHERE SoDH = @SoDH)
```

```
IF @SLN > (@SLD - @TongN)
    BEGIN
        PRINT ' SL nhap phai nho hon SL dat tru di tong SL da nhap '
        ROLLBACK TRANSACTION
    END
```

```
SELECT * FROM CTPNHAP
SELECT * FROM CTDONDH
SELECT * FROM PNHAP
```

```
INSERT INTO PNHAP(SoPN,SoDH) VALUES('N006','D003')
INSERT INTO CTPNHAP VALUES ('N005','DD01',15)
```

```
DELETE FROM CTPNHAP WHERE SoPN = 'N005' AND MaVTu = 'DD01'
```

```
SELECT * FROM CTDONDH WHERE SODH='D003' AND MAVTU='TV14'
```

```
SELECT      SUM(SLNhap)AS TongSLN
FROM        CTPNHAP
WHERE       MaVTu = 'TV14' AND SoPN IN (SELECT SoPN FROM PNHAP WHERE
SoDH='D003')
```

```
INSERT INTO CTPNHAP VALUES ('N006','TV14',4,15)
```

**3. Xây dựng trigger khi xóa dữ liệu trong bảng PXUAT với tên TG\_PXuat\_Xoa. Trong đó cần thực hiện các hành động : thực hiện tự động xóa các dòng dữ liệu liên quan bên bảng CTPXUAT.**

```
CREATE TRIGGER TG_PXuat_Xoa
ON PXUAT FOR DELETE
AS
    DELETE CTPXUAT WHERE SoPX = (SELECT SoPX FROM DELETED)
```

**4. Xây dựng trigger khi xoá dữ liệu trong bảng PNHAP với tên TG\_PNhap\_Xoa. Trong đó cần thực hiện các hành động : thực hiện tự động xoá các dòng dữ liệu liên quan bên bảng CTPNHAP**

```
CREATE TRIGGER TG_PNhap_Xoa
ON PNHAP FOR DELETE
AS
    DELETE CTPNHAP WHERE SoPN = (SELECT SoPN FROM DELETED)
```

**5. Trong bảng PNHAP tạo thêm cột tổng trị giá có tên TONGTG với kiểu FLOAT dùng để lưu tổng trị giá của 1 phiếu nhập hàng. Trong Trigger TG\_PNhap\_Them đã xây dựng trước đó ở phần 1. Bổ sung thêm các tính toán tự động sau :**

- Tăng thêm giá trị tại cột TongTG trong bảng PNHAP khi dữ liệu trong bảng CTPNHAP được thêm vào.
- Tăng giá trị tại cột TONGSLN trong bảng TONKHO khi dữ liệu trong bảng CTPNHAP được thêm vào.

```
ALTER TABLE PNHAP ADD TongTG FLOAT
```

```
CREATE TRIGGER TG_CTPNhap_Them
ON CTPNHAP FOR INSERT
AS
    DECLARE    @TongTG    FLOAT,
               @SoPN      CHAR(4),
               @NT         CHAR(6)
    DECLARE    @SLNMoi     INT,
               @MaV        CHAR(4)

    SELECT     @SoPN      = SoPN,
               @SLNMoi    = SLNhap,
               @MaV       = MaVTu
    FROM       INSERTED

    SELECT     @TongTG    = SUM(SLNhap*DGNhap)
    FROM       CTPNHAP
    WHERE      SoPN       = @SoPN

    UPDATE     PNHAP
    SET        TongTG = @TongTG
    WHERE      SoPN      = @SoPN

    SELECT     @NT        = CONVERT(CHAR(6),NgayNhap,112)
    FROM       PNHAP
    WHERE      SoPN       = @SoPN

    UPDATE     TONKHO
```

```
SET          TongSLN = ( TongSLN + @SLNMoi)
WHERE        NamThang = @NT AND MaVTu = @MaV
```

```
SELECT * FROM PNHAP
SELECT * FROM CTPNHAP
SELECT * FROM TONKHO
```

```
--vi phạm giá trị khoá ngoài
INSERT INTO CTPNHAP (SoPN,MaVTu,SLNhap,DGNhap) VALUES ('N005','DD02',10,15)
```

```
DELETE FROM CTPNHAP WHERE SoPN = 'N005' AND MaVTu = 'DD02'
```

**6. Xây dựng trigger khi xoá dữ liệu trong bảng CTPNHAP với tên TG\_CTPNhap\_Xoa. Trong đó cần thực hiện các tính toán sau :**

- Giảm giá trị tại cột TONGTG trong bảng PNHAP
- Giảm giá trị tại cột TONGSLN trong bảng TONKHO

```
CREATE TRIGGER TG_CTPNhap_Xoa
ON CTPNHAP FOR DELETE
AS
    DECLARE      @TongTG      FLOAT,
                 @SoPN        CHAR(4),
                 @NamThang    CHAR(6)
    DECLARE      @SLNhapXoa INT,
                 @MaVTu       CHAR(4),
                 @DGXoa       FLOAT

    SELECT      @SoPN=SoPN,
                 @SLNhapXoa=SLNhap,
                 @MaVTu=MaVTu,
                 @DGXoa=DGNhap
    FROM        DELETED

    UPDATE      PNHAP
    SET          TongTG = TongTG - (@SLNhapXoa * @DGXoa)
    WHERE       SoPN=@SoPN

    SELECT      @NamThang=CONVERT(CHAR(6),NgàyNhap,112)
    FROM        PNHAP
    WHERE       SoPN=@SoPN

    UPDATE      TONKHO
    SET          TongSLN = TongSLN - @SLNhapXoa
    WHERE       NamThang=@NamThang AND MaVTu=@MaVTu
```

**7. Xây dựng trigger khi cập nhật bảng CTPNHAP, Kiểm tra tính hợp lệ của dữ liệu :**

*CREATE TRIGGER TG\_CTPNhập\_Sua  
ON CTPNHAP FOR UPDATE  
AS*

*DECLARE @TongTG Float,  
@SoPN CHAR(4),  
@MaVTu CHAR(4)  
DECLARE @NamThang CHAR(6),  
@SLNhập INT*

*IF UPDATE(SoPN)  
BEGIN  
PRINT 'Khong duoc sua doi so phieu nhap'  
ROLLBACK TRANSACTION  
END*

*IF UPDATE (MaVTu)  
BEGIN  
PRINT 'Khong duoc sua doi ma vat tu'  
ROLLBACK TRANSACTION  
END*

*SELECT @SoPN=SoPN,  
@MaVTu=MaVTu  
FROM INSERTED*

*SELECT @TongTG = SUM(SLNhap\*DGNhap)  
FROM CTPNHAP  
WHERE SoPN=@SoPN*

*SELECT @NamThang=CONVERT(CHAR(6),NgayNhap,112)  
FROM PNHAP  
WHERE SoPN=@SoPN*

*UPDATE PNHAP SET TongTG = @TongTG WHERE SoPN=@SoPN*

*UPDATE TONKHO SET TongSLN=TongSLN -(Select SLNhap FROM DELETED) +  
(SELECT SLNhap FROM INSERTED)*

*SELECT @SLNhap=SUM(SLNhap)  
FROM CTPNHAP  
WHERE MaVTu=@MaVTu AND SoPN IN (SELECT SoPN FROM PNHAP  
WHERE CONVERT(CHAR(6),NgayNhap,112) = @NamThang)*

*UPDATE TONKHO SET TongSLN=@SLNhap WHERE NamThang=@NamThang*

*SELECT \* FROM CTPNHAP  
SELECT \* FROM TONKHO*

*UPDATE CTPNHAP SET SLNhap = 6 WHERE SoPN = 'N005' AND MaVTU = 'DD01'*



