Câu 1: Tạo CSDL QLBH gồm các bảng sau :

- tblVATTU: **MaVTu char(4)**, TenVtu nvarchar(30), DVTinh nvarchar(10), Dongia int.

- tblDONDH: **SoDH Int**, NgayDH datetime, Tennhacc nvarchar(40)

- tblCTDONDH: **SoDH Int, MaVTu char(4),** SLDat int

Tạo mối quan hệ, nhập dữ liệu cho các bảng, bảng 3 ít nhất có 5 bản ghi

Câu 2: Thực hiện các chương trình sau :

1. Viết hàm (hoặc thủ tục) cho mã vật tư tự động theo quy tắc:

Mã vật tư có dạng: VT01, ‘VT’ : quy định (luôn có), 01 : là số (VD: Hiện tại Vật tư có mã cao nhất là VT04, thì sinh mã mới là VT05)

1. Sử dụng biến CURSOR cho xem SoDH, TenVtu, , Tennhacc, SLdat của những SoDH có SLDat >=A.
2. Sử dụng thủ tục CURSOR cho xem SoDH, TenVtu, , Tennhacc, SLdat của những SoDH có SLDat >=A, với A là tham số đầu vào
3. Sử dụng thủ tục cho xem tổng tiền của một sodh là A, nếu A không tồn tại có dòng thông báo (thủ tục return)
4. Viết thủ tục cập nhật trên 1 bảng nào đó: tblCTDonDH với yêu cầu : 1. SoDH và MaVtu không trùng khóa chính, 2. SoDH có trong bảng tblDonDH, 3. MaVTu có trong bảng tblVatTu, 4. A ≤ SLDat ≤ B.
5. Thủ tục hiện thị : cho xem SoDH, TenVtu, , Tennhacc, Sldat, với sodh là A ; nếu không truyền tham số thì hiện thị toàn bộ
6. Hàm: vào SoDH, ra là tổng tiền của SoDH đó
7. Hàm tạo bảng: cho ra TenVtu, Tennhacc, Sldat, của 1 sodh
8. Hàm tạo bảng: cho ra TenVtu, Tennhacc, Sldat, dongia, tongtien của 1 sodh
9. Viết thủ tục cho ra SoDH có tổng tiền thấp nhất (hoặc cao nhất)
10. Hàm tạo bảng: vào là Tên nhà cung cấp, cho ra TenVtu, tổng số lượng đặt, tổng tiền của nhà cung cấp đó.
11. Hàm tạo bảng: vào là Tên nhà cung cấp, cho ra TenVtu, tổng số lượng đặt, tổng tiền, giảm giá, tiền phải trả của nhà cung cấp đó, với yêu cầu : nếu Tên vật tư là Gạch giảm 10%, Ngói giảm 20%, Xi măng giảm 30%. Tiền phải trả bằng tổng tiền – giảm giá.
12. Hàm tạo bảng: với 1 sodh cho ra TenVtu, Tennhacc, Sldat, dongia, tongtien, giảm giá, tiền phải trả. Với cột giảm giá có yêu cầu : nếu tongtien >=A, giảm giá = 10% của tongtien, nếu A>= tongtien >=B , giảm giá =5% tongtien, nếu tongtien <B thì giảm giá =0. Cột tiền phải trả = tổng tiền – giảm giá
13. Hàm tạo bảng: với 1 sodh cho ra TenVtu, Tennhacc, Sldat, dongia, tongtien, giảm giá, tiền phải trả. Với cột giảm giá có yêu cầu : nếu Tên vật tư là Gạch giảm 10%, Ngói giảm 20%, Xi măng giảm 30%. Cột tiền phải trả = tổng tiền – giảm giá

Câu 3. Viết trigger khi thêm bản ghi cho bảng: tblCTDonDH với yêu cầu :

1. SoDH có trong bảng tblDonDH, 2. MaVTu có trong bảng tblVatTu,

3. A ≤ SLDat ≤ B.

create database vtdh

--- tblVATTU: MaVTu char(4), TenVtu nvarchar(30), DVTinh nvarchar(10), Dongia int

create table tblvattu(

Mavt char(4) primary key not null,

Tenvt nvarchar(30),

Dvtinh nvarchar(10),

Dongia int

)

--

insert into tblvattu values

('VT01',N'Sữa',N'Hộp',7000),

('VT02',N'Cocacola',N'Lon',7000),

('VT03',N'Sữa milo',N'Hộp',8000)

--

-- tblDONDH: SoDH Int, NgayDH datetime, Tennhacc nvarchar(40)

create table tbldh(

Sodh int primary key not null,

Ngaydh datetime,

Tenncc nvarchar(50)

)

--

insert into tbldh values

(1,'2017-02-09',N'Sơn Trần'),

(2,'2017-11-09',N'Cty A')

--- tblCTDONDH: SoDH Int, MaVTu char(4), SLDat int

create table tblctdh(

Sodh int foreign key references tbldh,

Mavt char(4) foreign key references tblvattu,

Sld int,

primary key (Sodh,Mavt)

)

--

insert into tblctdh values

(1,'VT03',120),

(1,'VT02',240),

(1,'VT01',40),

(2,'VT02',80),

(2,'VT03',130)

---------insert

--Tạo mối quan hệ, nhập dữ liệu cho các bảng, bảng 3 ít nhất có 5 bản ghi

Câu 2:

1. Thủ tục sinh mã tự động

create proc tt1 @mavt char(10) output

as

begin

declare @max int

Select @max=max(cast(substring(mavt,3,2) as int))+1 from tblvattu

set @mavt = '0000'+ LTRIM(cast(@max as char(10)))

set @mavt = 'VT' + SUBSTRING(@mavt,LEN(@mavt)-1,2)

end

----loi goi

declare @s\_mavt char(10)

exec tt1 @s\_mavt output

print @s\_mavt

2. Biến kiểu CURSOR

declare @s\_sodh int, @s\_tenvt nvarchar(40)

declare @s\_tenncc nvarchar(50),@s\_sld int

declare cur\_vt cursor

keyset

for

select dh.SoDH, TenVt, Tenncc, SLd

from tblvattu vt

inner join tblctdh ct on ct.mavt = vt.mavt

inner join tbldh dh on dh.sodh=ct.sodh

where sld>=10

open cur\_vt

while 0=0

begin

fetch next from cur\_vt into @s\_sodh, @s\_tenvt,@s\_tenncc,@s\_sld

if @@fetch\_status<>0 break

print str(@s\_sodh,10)+ @s\_tenvt+@s\_tenncc+str(@s\_sld,10)

end

close cur\_vt

deallocate cur\_vt

Câu 3. Thủ tục CURSOR

create proc tt\_cur1

@s\_sld int, @s\_cur cursor varying output

as

set @s\_cur=cursor

keyset

for

select dh.SoDH, TenVt, Tenncc, SLd

from tblvattu vt

inner join tblctdh ct on ct.mavt = vt.mavt

inner join tbldh dh on dh.sodh=ct.sodh

where sld>=@s\_sld

open @s\_cur

--------loi goi

declare @s\_sodh int,@s\_tenvt nvarchar(20)

declare @s\_tenncc nvarchar(20),@s\_sld int

declare @ss\_cur cursor

exec tt\_cur1 2,@ss\_cur output

while 0=0

begin

fetch next from @ss\_cur into @s\_sodh,@s\_tenvt,@s\_tenncc,@s\_sld

if @@FETCH\_STATUS <>0 break

print str(@s\_sodh,10)+@s\_tenvt+@s\_tenncc+str(@s\_sld,10)

end

close @ss\_cur

deallocate @ss\_cur

Câu 4. Sử dụng thủ tục cho xem tổng tiền của một sodh là A, (đang thiếu phần RETURN)

Create proc tt\_04 @s\_sodh int,@s\_tt int output

as

select @s\_tt = sum(SlDat\*Dongia)

from tblVATTU vt

inner join tblCTdonDH ctdh on ctdh.MaVTu=vt.MaVTu

where SoDH=@s\_sodh

----loi goi

declare @ss\_tt int

exec tt\_02 1,@ss\_tt output

print @ss\_tt

Câu 5:

alter proc tt\_05 @s\_SoDH Int, @s\_MaVTu char(4), @s\_SLDat int

as

----kiem tra khoa chính

if exists

(select \* from tblCTdonDH where sodh=@s\_SoDH and MaVTu=@s\_MaVTu)

begin

print 'trung khoa chinh'

return

end

-----kiem tra khoa ngoai 1 SoDH

if NOT exists (select \* from tblDonDH where sodh=@s\_SoDH)

begin

print 'không có khoa ngoại SoDH'

return

end

-----kiem tra khoa ngoai 2 Mavtu

if NOT exists (select \* from tblVATTU where MaVTu=@s\_MaVTu)

begin

print 'không có khoa ngoại MaVtu'

return

end

-----kiem tra mien gia trị

if NOT ((@s\_SLDat>=100) and (@s\_SLDat<=200))

begin

print 'loi 100 <= sldat <=200 '

return

end

INSERT INTO tblCTdonDH values(@s\_SoDH,@s\_MaVTu,@s\_SLDat)

----lời gọi 1

Exec tt\_05 1,'VT01',56

----lời gọi 2

Exec tt\_05 50,'VT01',56

----lời gọi 3

Exec tt\_05 5,'VT04',56

----lời gọi 4

Exec tt\_05 5,'VT02',56

----lời gọi 5

Exec tt\_05 5,'VT02',256

----lời gọi 6

Exec tt\_05 5,'VT02',156

Câu 6:

alter proc tt\_06 @s\_sodh int =NULL

as

IF @s\_sodh is NULL

select \*

from tblVATTU vt

inner join tblCTdonDH ctdh on ctdh.MaVTu=vt.MaVTu

inner join tblDonDH dh on dh.SoDH=ctdh.SoDH

ELSE

select \*

from tblVATTU vt

inner join tblCTdonDH ctdh on ctdh.MaVTu=vt.MaVTu

inner join tblDonDH dh on dh.SoDH=ctdh.SoDH

where dh.sodh= @s\_sodh

------loi goi

exec tt\_06 1

exec tt\_06 2

exec tt\_06

Câu 7:

Create function hh\_17 (@s\_sodh int)

returns int

As

Begin

declare @s\_tt int

Select @s\_tt=sum(SlDat\*Dongia)

From tblVATTU vt

Inner join tblCTDONDH ctdh on vt.mavtu = ctdh.mavtu

Where sodh = @s\_sodh

Return @s\_tt

End

-------lời gọi

print 'Tong tien = ' + str(dbo.hh\_17(2),10)

9. Hàm tạo bảng : cho ra TenVtu, Tennhacc, Sldat, dongia, tongtien của 1 sodh

create function hhh ()

returns @hh table

(TenVtu nvarchar(30),

Tennhacc nvarchar(30),

Sldat int,

dongia int,

tongtien int)

as

begin

insert into @hh (TenVtu,Tennhacc,Sldat, dongia)

select vt.tenvt,tenncc,sld, dongia

from tblvattu vt

inner join tblctdh ct on ct.mavt=vt.mavt

inner join tbldh dh on dh.sodh=ct.sodh

update @hh

set tongtien=dongia\*Sldat

return

end

--loi goi

select \* from hhh()

Câu 11.

create function th\_thu\_ty ( @s\_TenNhaCC nvarchar(40) )

Returns Table

as

Return

(

Select TenVtu,sum(SlDat) as tsld,sum(Dongia\*SlDat) as tt

From tblVATTU vt

Inner join tblCTDONDH ctdh on ctdh. MaVTu = vt. MaVTu

Inner join tblDONDH dh on dh. SoDH= ctdh. SoDH

Where TenNhaCC= @s\_TenNhaCC

group by TenVtu

)

-------Lời gọi:

Select \* from dbo.th\_thu\_ty(N'Nguyễn Văn A')

Câu 12:

alter function th\_thu\_ty1 ( @s\_TenNhaCC nvarchar(40) )

Returns @s\_table Table

(TenVtu nvarchar(30),

tsld int,

tt int,

gg int,

tpt int

)

as

BEGIN

insert into @s\_table (tenvtu,tsld,tt)

Select TenVtu,sum(SlDat) ,sum(Dongia\*SlDat)

From tblVATTU vt

Inner join tblCTDONDH ctdh on ctdh. MaVTu = vt. MaVTu

Inner join tblDONDH dh on dh. SoDH= ctdh. SoDH

Where TenNhaCC= @s\_TenNhaCC

group by TenVtu

update @s\_table

set gg=

case TenVtu

when N'Gachj' then 0.1\*tt

when N'ngói' then 0.2\*tt

else 0.3\*tt

end

update @s\_table

set tpt = tt-gg

Return

END

-------Lời gọi:

Select \* from dbo.th\_thu\_ty1(N'Nguyễn Văn A')

Câu 13:

alter function hh\_13 ( @s\_Sodh int )

RETURNS @s\_table TABLE

(sodh int,

TenVtu nvarchar(30),

Tennhacc nvarchar(40),

Sldat int,

tongtien int,

gg int,

tpt int

)

AS

BEGIN

INSERT INTO @s\_table (SoDH, TenVtu, Tennhacc,Sldat,tongtien)

Select dh.SoDH, TenVtu, Tennhacc, Sldat,Dongia\*SlDat

From tblVATTU vt

Inner join tblCTDONDH ctdh on ctdh. MaVTu = vt. MaVTu

Inner join tblDONDH dh on ctdh. SoDH = dh.sodh

Where ctdh.sodh = @s\_Sodh

update @s\_table

set gg=

CASE

when tongtien>=100000 then 0.2\*tongtien

when tongtien>=50000 then 0.1\*tongtien

else 0

end

update @s\_table

set tpt = tongtien - gg

RETURN

END

-----LOI GOI

Select \* from dbo.hh\_13 (2)

Câu 14.

alter function hh\_08 ( @s\_sodh int )

RETURNS @s\_table TABLE

(SoDH int,

TenVtu nvarchar(30),

Tennhacc nvarchar(40),

Sldat int,

tt int,

gg int,

tpt int

)

AS

BEGIN

insert into @s\_table (SoDH, TenVtu , Tennhacc, Sldat,tt)

SELECT ctdh.SoDH, TenVtu , Tennhacc, Sldat,dongia\*SlDat as tt

FROM tblVATTU VT

INNER join tblCTDONDH ctdh ON VT. MaVTu= CTDH. MaVTu

Inner join tblDONDH DH on dh. SoDH= ctdh.sodh

Where dh.sodh =@s\_sodh

update @s\_table

set gg=

CASE TenVtu

when N'Gachj' then 0.1\*tt

when N'ngói' then 0.2\*tt

else 0.3\*tt

end

update @s\_table

set tpt = tt-gg

RETURN

END

----LOI GOI

select \* from dbo.hh\_08(2)

Câu trigger:

alter trigger them\_tblctdondh

on tblctdondh

INSTEAD OF insert

as

if not exists ( select \* from tblDonDH where

sodh = (select sodh from inserted) )

begin

ROLLBACK TRANSACTION

raiserror('loi đonh', 16, 1)

return

end

if not ((select SlDat from Inserted) between 100 and 200)

begin

ROLLBACK TRANSACTION

raiserror('loi mien gia tri', 16, 1)

return

end

----- khi nào sẽ kích hoạt trigger

insert tblCTdonDH values( 7,'VT02',20)