1. Stored-procedure tính tổng của 2 số nguyên.

create PROCEDURE dbo.TinhTongHaiSoNguyen @a int , @b int

AS

Print concat('Tong hai so nguyen la ',@a + @b)

go

execute dbo.TinhTongHaiSoNguyen 2 ,3

go

1. Stored procedure liệt kê những thông tin của đầu sách, thông tin tựa sách và số lượng sách hiện chưa được mượn của một đầu sách cụ thể (ISBN).

Với **Tuasach** (ma\_tuasach, tuasach, tacgia, tomtat)

**Dausach** (isbn, ma\_tuasach, ngonngu, bia, trangthai)

**Cuonsach** (isbn, ma\_cuonsach, tinhtrang)

-- Drop the table 'TuaSach' in schema 'dbo'

IF EXISTS (

SELECT \*

FROM sys.tables

JOIN sys.schemas

ON sys.tables.schema\_id = sys.schemas.schema\_id

WHERE sys.schemas.name = N'dbo'

AND sys.tables.name = N'TuaSach'

)

DROP TABLE dbo.TuaSach

GO

CREATE TABLE dbo.TuaSach

(

ma\_tuasach char(3) NOT NULL PRIMARY KEY,

tuasach [NVARCHAR](50) NOT NULL,

tacgia [NVARCHAR](50) NOT NULL,

tomtat [NVARCHAR](50) NOT NULL,

);

GO

-- Drop the table 'Dausach' in schema 'dbo'

IF EXISTS (

SELECT \*

FROM sys.tables

JOIN sys.schemas

ON sys.tables.schema\_id = sys.schemas.schema\_id

WHERE sys.schemas.name = N'dbo'

AND sys.tables.name = N'Dausach'

)

DROP TABLE dbo.Dausach

GO

CREATE TABLE dbo.Dausach

(

isbn char(13) NOT NULL,

ma\_tuasach char(3) NOT NULL,

ngonngu [NVARCHAR](20) NOT NULL,

bia [NVARCHAR](20) NOT NULL,

trangthai [NVARCHAR](5) NOT NULL,

);

GO

-- Drop the table 'Cuonsach' in schema 'dbo'

IF EXISTS (

SELECT \*

FROM sys.tables

JOIN sys.schemas

ON sys.tables.schema\_id = sys.schemas.schema\_id

WHERE sys.schemas.name = N'dbo'

AND sys.tables.name = N'Cuonsach'

)

DROP TABLE dbo.Cuonsach

GO

CREATE TABLE dbo.Cuonsach

(

isbn char(13) NOT NULL,

ma\_cuonsach char(3) NOT NULL,

tinhtrang char(10) NOT NULL

);

GO

-- Insert rows into table 'Tuasach'

INSERT INTO Tuasach

( -- columns to insert data into

ma\_tuasach,

tuasach,

tacgia,

tomtat

)

-- ma\_tuasach char(3) NOT NULL PRIMARY KEY,

-- tuasach [NVARCHAR](50) NOT NULL,

-- tacgia [NVARCHAR](50) NOT NULL,

-- tomtat [NVARCHAR](50) NOT NULL,

VALUES

( -- first row: values for the columns in the list above

'001', 'Doraemon', 'Fujiko F. Fujio', 'Doraemon là nhân vật chính hư cấu trong loạt Manga '

),

( -- second row: values for the columns in the list above

'002', 'Spider-man', 'Stan Lee', 'Người nhện là một siêu anh hùng hư cấu '

)

-- add more rows here

GO

-- Insert rows into table 'Dausach'

INSERT INTO Dausach

( -- columns to insert data into

isbn,

ma\_tuasach,

ngonngu,

bia,

trangthai

)

VALUES

( -- first row: values for the columns in the list above

'1234567890123', '001', 'Nhat', 'Cung', 'Moi'

),

( -- second row: values for the columns in the list above

'1234567890122', '001', 'viet', 'mem', 'cu'

),

( -- second row: values for the columns in the list above

'1234567890121', '002', 'viet', 'cung', 'cu'

)

-- add more rows here

GO

-- Insert rows into table 'TuaSach'

INSERT INTO Cuonsach

( -- columns to insert data into

isbn,

ma\_cuonsach,

tinhtrang

)

VALUES

( -- first row: values for the columns in the list above

'1234567890121', '001', 'damuon'

),

( -- second row: values for the columns in the list above

'1234567890121', '002', 'chuamuon'

),

( -- second row: values for the columns in the list above

'1234567890121', '003', 'chuamuon'

),

( -- first row: values for the columns in the list above

'1234567890122', '001', 'damuon'

),

( -- second row: values for the columns in the list above

'1234567890122', '002', 'chuamuon'

),

( -- second row: values for the columns in the list above

'1234567890122', '003', 'chuamuon'

)

-- add more rows here

GO

-- Create a new stored procedure called 'getBooksInfo' in schema 'dbo'

-- Drop the stored procedure if it already exists

IF EXISTS (

SELECT \*

FROM INFORMATION\_SCHEMA.ROUTINES

WHERE SPECIFIC\_SCHEMA = N'dbo'

AND SPECIFIC\_NAME = N'getBooksInfo'

)

DROP PROCEDURE dbo.getBooksInfo

GO

-- Create the stored procedure in the specified schema

CREATE PROCEDURE dbo.getBooksInfo

AS

-- body of the stored procedure

-- Select \*

SELECT tuasach, tacgia, tomtat, ngonngu, bia, trangthai, ds.isbn,

COUNT (cs.tinhtrang) as soSachConLai

from TuaSach ts

JOIN Dausach ds

ON ts.ma\_tuasach = ds.ma\_tuasach

JOIN

Cuonsach cs

ON cs.isbn =ds.isbn

WHERE cs.tinhtrang = 'chuamuon'

GROUP BY tuasach, tacgia, tomtat, ngonngu, bia, trangthai, ds.isbn,cs.isbn

GO

-- example to execute the stored procedure we just created

EXECUTE dbo.getBooksInfo

GO

1. Viết hàm tính tuổi của người có năm sinh được nhập vào như một tham số của hàm.

create PROCEDURE dbo.TinhNamSinh @a int

AS

Print concat('Tuoi cua nguoi do la ',2022 - @a )

go

execute dbo.TinhNamSinh 2002

go

1. Cho CSDL như sau:

**DocGia** (ma\_DocGia, ho, tenlot, ten, ngaysinh)

**Nguoilon** (ma\_DocGia, sonha, duong, quan, dienthoai, han\_sd)

**Treem** (ma\_DocGia, ma\_DocGia\_nguoilon)

**Tuasach** (ma\_tuasach, tuasach, tacgia, tomtat)

**Dausach** (isbn, ma\_tuasach, ngonngu, bia, trangthai)

**Cuonsach** (isbn, ma\_cuonsach, tinhtrang)

**DangKy (**isbn, ma\_DocGia, ngay\_dk, ghichu**)**

**Muon (**isbn, ma\_cuonsach, ma\_DocGia, ngay\_muon, ngay\_hethan**)**

**QuaTrinhMuon(**isbn, ma\_cuonsach, ngay\_muon, ma\_DocGia, ngay\_hethan, ngay\_tra, tien\_muon, tien\_datra, tien\_datcoc, ghichu**)**

Việt các stored procedures thưc hiện các việc sau:

1. ***Xem thông tin độc giả***

Tên: **sp\_ThongtinDocGia**

Nội dung: Liệt kê những thông tin của độc giả tương ứng với mã độc giả. Nếu độc giả là người lớn thì hiển thị

thông tin độc giả + thông tin trong bảng người lớn. Nếu độc giả là trẻ em thì hiển thị những thông tin độc giả +

thông tin của bảng trẻ em.

Thực hiện:

[1] Kiểm tra độc giả này thuộc loại người lớn hay trẻ em.

[2] Nếu là người lớn thì: In những thông độc giả này, gồm có: thông tin độc giả + thông tin người lớn.

[3] Nếu là trẻ em thì: In những thông tin liên quan đến độc giả này, gồm có: thông tin độc giả + thông tin trẻ em.

create proc sp\_ThongTinDocGia (@maDG int)

as

begin

if exists (select \* from Nguoilon where Nguoilon.ma\_DocGia = @maDG)

begin

select \* from DocGia join Nguoilon on DocGia.ma\_DocGia = Nguoilon.ma\_DocGia where Nguoilon.ma\_DocGia = @maDG

end

else if exists (select \* from Treem where Treem.ma\_DocGia = @maDG)

begin

select \* from DocGia join Treem on DocGia.ma\_DocGia = Treem.ma\_DocGia where Treem.ma\_DocGia = @maDG

end

end

***b. Thông tin đầu sách:***

Tên: **sp\_ThongtinDausach**

Nội dung: Liệt kê những thông tin của đầu sách, thông tin tựa sách và số lượng sách hiện chưa được mượn của một đầu sách cụ thể (ISBN).

create proc sp\_ThongtinDausach(@isbn int)

as

select

Dausach.isbn, Dausach.bia, Dausach.ma\_tuasach, Dausach.ngonngu, Dausach.trangthai,

Tuasach.tuasach, Tuasach.tacgia, Tuasach.tomtat,

count(Cuonsach.tinhtrang) as slKhaDung

from Dausach join Cuonsach on Dausach.isbn = Cuonsach.isbn,

Tuasach

where Dausach.isbn = @isbn and Cuonsach.tinhtrang = 1 and Tuasach.ma\_tuasach = Dausach.ma\_tuasach

group by Dausach.isbn, Dausach.bia, Dausach.ma\_tuasach, Dausach.ngonngu, Dausach.trangthai,

Tuasach.tuasach, Tuasach.tacgia, Tuasach.tomtat

***c. Liệt kê những độc giả người lớn đang mượn sách:***

Tên: **sp\_ThongtinNguoilonDangmuon**

Nội dung: Liệt kê những thông tin của tất cả độc giả đang mượn sách của thư viện.

create proc sp\_ThongtinNguoilonDangmuon

as

begin

select \*

from Nguoilon join DocGia on Nguoilon.ma\_DocGia = DocGia.ma\_DocGia

where Nguoilon.ma\_DocGia in (select Nguoilon.ma\_DocGia from Muon join Nguoilon on Muon.ma\_DocGia = Nguoilon.ma\_DocGia)

end

***d. Liệt kê những độc giả người lớn đang mượn sách quá hạn:***

Tên: sp\_ThongtinNguoilonQuahan

Nội dung: Liệt kê những thông tin của tất cả độc giả đang mượn sách của thư viện đang trong tình trạng mượn quá hạn 14 ngày.

create proc sp\_ThongtinNguoilonQuahan

as

begin

select \*

from Nguoilon as adult join DocGia on adult.ma\_DocGia = DocGia.ma\_DocGia

where adult.ma\_DocGia in (select Muon.ma\_DocGia

from Muon

where datediff(day, Muon.ngayhethan , convert(date, getdate())) >= 14 )

end

***e. Liệt kê những độc giả người lớn đang mượn sách có trẻ em cũng đang mượn sách:***

Tên: **sp\_DocGiaCoTreEmMuon**

Nội dung: Liệt kê những những độc giả đang trong tình trạng mượn sách và những trẻ em độc giả này đang bảo lãnh cũng đang trong tình trạng mượn sách.

create proc sp\_DocGiaCoTreEmMuon

as

begin

select DocGia.ma\_DocGia, DocGia.ho, DocGia.tenlot, DocGia.ten

from Nguoilon join DocGia on Nguoilon.ma\_DocGia = DocGia.ma\_DocGia

join Treem on Nguoilon.ma\_DocGia = Treem.ma\_DocGia\_nguoilon

where Nguoilon.ma\_DocGia in (select Nguoilon.ma\_DocGia

from Muon join Nguoilon on Muon.ma\_DocGia = Nguoilon.ma\_DocGia)

and Nguoilon.ma\_DocGia in (select Treem.ma\_DocGia\_nguoilon

from Muon join Treem on Muon.ma\_DocGia = Treem.ma\_DocGia)

group by DocGia.ma\_DocGia, DocGia.ho, DocGia.tenlot, DocGia.ten

end

**Bài tập 5**: Tạo một số Trigger như sau trong CSDL **Thư viện**:

**5.1. tg\_delMuon:**

Nội dung: Cập nhật tình trạng của cuốn sách là yes.

IF OBJECT\_ID('tg\_delMuon', 'TR') IS NOT NULL

BEGIN

PRINT('trigger exists')

DROP TRIGGER dbo.tg\_delMuon

END

ELSE

PRINT('trigger not exist')

GO

CREATE TRIGGER tg\_delMuon ON Muon

AFTER DELETE

AS

DECLARE @isbn int

SELECT @isbn = de.isbn

FROM deleted de

BEGIN

UPDATE Cuonsach SET tinhtrang = 'Yes'

WHERE isbn = @isbn

END

GO

**5.2. tg\_insMuon:**

Nội dung: Cập nhật tình trạng của cuốn sách là no.

IF OBJECT\_ID ('tg\_insMuon', 'TR') IS NOT NULL

DROP TRIGGER dbo.tg\_insMuon;

GO

CREATE TRIGGER tg\_insMuon ON MUON

AFTER INSERT

AS

DECLARE @isbn int

SELECT @isbn = ne.isbn

FROM inserted ne

BEGIN

UPDATE Cuonsach SET tinhtrang = 'No'

WHERE isbn = @isbn

END

GO

**5.3. tg\_updCuonSach:**

Nội dung: Khi thuộc tính tình trạng trên bảng cuốn sách được cập nhật thì trạng thái của đầu sách cũng được cập

nhật theo. Cài đặt các thủ tục sau cho CSDL Quản lý thư viện.

IF OBJECT\_ID ('tg\_updCuonSach', 'TR') IS NOT NULL

DROP TRIGGER dbo.tg\_updCuonSach;

GO

CREATE TRIGGER tg\_updCuonSach ON CuonSach

AFTER UPDATE

AS

DECLARE @tinh\_trang nvarchar(10), @isbn int

SELECT @tinh\_trang = ne.tinhtrang, @isbn = ne.isbn

FROM inserted ne

BEGIN

UPDATE Dausach SET trangthai = @tinh\_trang

WHERE isbn = @isbn

END

GO

**5.4 tg\_InfThongBao**

Nội dung: Viết trigger khi thêm mới, sửa tên tác giả, thêm/sửa một tựa sách thì in ra câu thông báo bằng Tiếng

Việt ‘*Đã thêm mới tựa sách*’.

IF OBJECT\_ID('tg\_infThongBao', 'TR') IS NOT NULL

DROP TRIGGER dbo.tg\_InfThongBao

GO

CREATE TRIGGER tg\_infThongBao ON TuaSach

AFTER UPDATE, INSERT

AS

PRINT N'Đã thêm mới tựa sách'

*Gợi ý* :

Kiểm tra trigger đã tạo bằng khối lệnh để dữ liệu không bị thay đổi :

begin tran

--khối lệnh thêm,xóa,sửa

rollback

**Bài tập 6:**

****

Viết các Function sau trong CSDL Đề án

6.1. Viết hàm trả về tổng tiền lương trung bình của một phòng ban tùy ý (truyền vào MaPB)

CREATE FUNCTION AverageSalaryOfDept(@MAPB varchar(2)) Returns money

AS

BEGIN

DECLARE @averageSalary money

SELECT @averageSalary = avg(LUONG)

FROM NHANVIEN

WHERE PHG = @MAPB

RETURN @averageSalary

END

go

select dbo.AverageSalaryOfDept('12')

6.2. Viết hàm trả về tổng lương nhận được của nhân viên theo dự án (truyền vào MaNV và MaDA)

CREATE FUNCTION SalaryOfCustomerWithProject(@MANV varchar(9) ,@MADA varchar(2)) Returns money

AS

BEGIN

DECLARE @salary money

SELECT @salary = LUONG \*(

SELECT SUM (THOIGIAN)

FROM PHANCONG pc

WHERE MA\_NVIEN = @MANV and SODA = @MADA

)

FROM NHANVIEN

WHERE MANV = @MANV

RETURN @salary

END

go

select dbo.SalaryOfCustomerWithProject('123456789','12')

6.3. Viết hàm trả về tổng tiền lương trung bình của các phòng ban

CREATE FUNCTION CountBonus() RETURNS TABLE

AS

return select pb.TENPHG , avg(nv.LUONG)

from NHANVIEN nv

inner join PHONGBAN pb on pb.MAPHG = nv.PHG

group by (PHG)

go

select \* from SalaryOfCustomerWithProject()

go

6.4. Viết hàm trả về tổng tiền thưởng cho nhân viên dựa vào tổng số giờ tham gia dự án(Time\_Total) như sau:

- Nếu Time\_Total >=30 và <=60 thì tổng tiền thưởng = 500 ($)

- Nếu Time\_Total >60 và <100 thì tổng tiền thưởng = 1000 ($)

- Nếu Time\_Total >=100 và <150 thì tổng tiền thưởng =1200($)

- Nếu Time\_Total >=150 thì tổng tiền thưởng = 1600 ($)

CREATE FUNCTION CountBonus() RETURNS TABLE

AS

return select

CASE

WHEN sum(pcTHOIGIAN)>=30 and sum(pcTHOIGIAN) <=60 THEN 500

WHEN sum(pcTHOIGIAN)>60 and sum(pcTHOIGIAN) <100 THEN 1000

WHEN sum(pcTHOIGIAN)>=100 and sum(pcTHOIGIAN) <150 THEN 1200

WHEN sum(pcTHOIGIAN)>=150 THEN 1600

ELSE 0

END as Bonus

from PHANCONG pc

inner join NHANVIEN nv on pc.MA\_NVIEN = nv.MANV

group by (pc.MA\_NVIEN)

go

select \* from CountBonus()

go

6.5. Viết hàm trả ra tổng số dự án theo mỗi phòng ban.

CREATE FUNCTION CountProjectInDept() RETURNS TABLE

AS

return select pb.TENPHG , count(da.MADA)

from DUAN da

inner join PHONGBAN pb on pb.MAPHG = da.PHONG

group by (PHONG)

go

select \* from CountProjectInDept()

go

6.6. Viết hàm trả về kết quả là một bảng (Table), viết bằng hai cách: Inline Table-Valued Functions và Multistatement Table-Valued. Thông tin gồm: MaNV, HoTen, NgaySinh, NguoiThan, TongLuongTB.