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

CREATE PROC sum

@num1 int, @num2 int,

@result INT OUTPUT

AS

SET @result = @num1 + @num2

DECLARE @re int

EXEC sum 100, 2, @re OUTPUT

PRINT @re

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)

create proc thongtinDausach

as

begin

select \* from Dausach

end

exec thongtinDausach

go

create proc thongtinTuasach

as

begin

select \* from Tuasach

end

exec thongtinTuasach

go

create proc Available

as

begin

select Dausach.isbn, ma\_tuasach, COUNT(tinhtrang) soluong

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

where tinhtrang = 'Available'

group by Dausach.isbn, ma\_tuasach

end

exec Available

go

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

create function CalAge (@birth int)

returns int

as

begin

declare @age int

set @age = (YEAR(GETDATE()) - @birth)

return @age

end

go

select dbo.CalAge(1990) as Age;

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 @ma\_DocGia varchar(10)

AS

BEGIN

IF (YEAR(GETDATE()) – (SELECT year(ngaysinh) FROM DocGia WHERE ma\_DocGia = @ma\_DocGia) >= 18)

SELECT \*

FROM DocGia, Nguoilon WHERE DocGia.ma\_DocGia = Nguoilon.ma\_DocGia AND DocGia.ma\_DocGia = @ma\_DocGia

ELSE

SELECT \*

FROM DocGia, Treem WHERE DocGia.ma\_DocGia = Treem.ma\_DocGia AND DocGia.ma\_DocGia = @ma\_DocGia

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 varchar(10) AS

SELECT \*

FROM Tuasach, Dausach, (SELECT COUNT(ma\_counsach) as SL)

FROM Cuonsach

WHERE tinhtrang = yes and isbn = @ISBN) AS

SoLuongSachChuaMuon

WHERE Tuasach.ma\_tuasach = Dausach.ma\_tuasach and Dausach.isbn = @ISBN

***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 DocGia, Nguoilon, (SELECT ma\_DocGia FROM QuaTrinhMuon WHERE ngay\_tra = null) AS DocGiaDangMuon WHERE DocGia.ma\_DocGia = Nguoilon.ma\_DocGia AND Nguoilon.ma\_DocGia = DocGiaDangMuon.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 DocGia, Nguoilon, (SELECT ma\_DocGia FROM QuaTrinhMuon WHERE ngay\_tra = null AND DATEDIFF(DAY, ngay\_hethan, GETDATE()) > 0) AS DocGiaQuaHan WHERE DocGia.ma\_DocGia = Nguoilon.ma\_DocGia AND Nguoilon.ma\_DocGia = DocGiaQuaHan.ma\_DocGia

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

SELECT \*

FROM DocGia, Nguoilon, (SELECT Treem.ma\_DocGia\_nguoilon as Ma\_NL From Treem, QuaTrinhMuon AS TreEmMuon, QuaTrinhMuon AS NguoiLonMuon

WHERE Treem.ma\_DocGia = TreEmMuon.ma\_DocGia AND TreEmMuon.ngay\_tra = null AND Treem.ma\_DocGia\_nguoilon = NguoiLonMuon.ma\_DocGia AND NguoiLonMuon.ngay\_tra = null) AS NguoiLonDaKTra

WHERE DocGia.ma\_DocGia = Nguoilon.ma\_DocGia and Nguoilon.ma\_DocGia = NguoiLonDaKTra.Ma\_NL

**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.

CREATE TRIGGER tg\_delMuon ON Muon AFTER DELETE AS

BEGIN

DECLARE @del\_isbn char(10), @del\_ma\_cuonsach char(10)

SELECT @del\_isbn = isbn, @del\_ma\_cuonsach = ma\_cuonsach

FROM deleted UPDATE Cuonsach SET tinhtrang = yes WHERE isbn = @del\_isbn AND ma\_cuonsach = @del\_ma\_cuonsach

END

**5.2. tg\_insMuon:**

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

CREATE TRIGGER tg\_insMuon ON Muon AFTER INSERT

AS

BEGIN

DECLARE @ins\_isbn char(10), @ins\_ma\_cuonsach char(10)

SELECT @ins\_isbn = isbn, @ins\_ma\_cuonsach = ma\_cuonsach

FROM inserted UPDATE Cuonsach SET tinhtrang = no

WHERE isbn = @ins\_isbn AND ma\_cuonsach = @ins\_ma\_cuonsach

END

**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.

CREATE TRIGGER tg\_updCuonSach ON Cuonsach AFTER UPDATE

AS

BEGIN

DECLARE @upd\_tinhtrang bit, @upd\_isbn char(10)

SELECT @upd\_tinhtrang = tinhtrang, @upd\_isbn = isbn

FROM inserted UPDATE Dausach SET trangthai = @upd\_tinhtrang AND isbn =@upd\_isbn

END

**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*’.

*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

CREATE TRIGGER tg\_InfThongBao ON Tuasach AFTER INSERT, UPDATE

AS

PRINT(N‘Đã thêm mới tựa sách’)

**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 SumAvgSalaryPhg(@MaPB varchar(2)) RETURNS FLOAT

AS

BEGIN

DECLARE @Luong\_Avg\_Phg FLOAT

SELECT @Luong\_Avg\_Phg = (SUM(LUONG) / COUNT(MANV))

FROM NHANVIEN

WHERE PHG = @MaPB

RETURN @Luong\_Avg\_Phg

END

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

CREATE FUNCTION SumSalaryNVDuAn(@MaNV varchar(9), @MaDA varchar(2)) RETURNS NUMERIC AS

BEGIN

DECLARE @Luong\_NV\_DA NUMERIC

SELECT @Luong\_NV\_DA = (LUONG \* (SELECT THOIGIAN FROM PHANCONG WHERE MA\_NVIEN=@MaNV and SODA=@MaDA)/ (SELECT SUM(THOIGIAN) FROM PHANCONG

WHERE MA\_NVIEN=@MaNV))

FROM NHANVIEN

WHERE MANV = @MaNV RETURN @Luong\_NV\_DA

END

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

CREATE FUNCTION SumSalaryAvgPhg()

RETURNS @table table(PHG varchar(2), LuongTB numeric(18,0))

AS

BEGIN

INSERT @table SELECT PHG, (SUM(LUONG) / COUNT(MANV))

FROM NHANVIEN

GROUP BY PHG

RETURN

END

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 SumMoneyNV\_TimeDA(@Time\_Total numeric(18,0)) RETURNS MONEY

AS

BEGIN

IF(@Time\_Total >= 30 AND @Time\_Total <= 60)

RETURN $500

IF(@Time\_Total > 60 AND @Time\_Total < 100)

RETURN $1000

IF(@Time\_Total >= 100 AND @Time\_Total <150)

RETURN $1200

IF(@Time\_Total >= 150)

RETURN $1600

RETURN $0

END

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

CREATE FUNCTION SumDA\_PhongBan()

RETURNS @table table(PHONG varchar(2), SoLuongDA int)

AS

BEGIN

INSERT @table SELECT PHONG, COUNT(MADA)

FROM DEAN

GOUP BY PHONG

RETURN

END

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.

CREATE FUNCTION KetQuaBang()

RETURNS TABLE

AS

RETURN (SELECT NHANVIEN.MANV, NHANVIEN.HONV, NHANVIEN.TENLOT,NHANVIEN.TENNV, NHANVIEN.NGSINH, THANNHAN.TENTN, NHANVIEN.LUONG

FROM NHANVIEN, THANNHAN

CREATE FUNCTION KetQuaBang()

RETURNS @table table(MANV varchar(9), HONV nvarchar(15), TENLOT

nvarchar(30), TENNV nvarchar(30), NGSINH smalldatetime,

TENTN varchar(20), LUONGTB numeric(18,0))

AS

BEGIN

INSERT @table SELECT NHANVIEN.MANV, NHANVIEN.HONV, NHANVIEN.TENLOT, NHANVIEN.TENNV, NHANVIEN.NGSINH, THANNHAN.TENTN, NHANVIEN.LUONG

FROM NHANVIEN, THANNHAN

WHERE NHANVIEN.MANV=THANNHAN.MA\_NVIEN

RETURN

END