

Họ và tên: Đặng Thái Hòa – MSSV: 20520510

BÀI TẬP THỰC HÀNH MÔN QUẢN LÝ THÔNG TIN TUẦN 2

--- A PROCEDURE

--1A Tham số vào là MSGV, TENG V, SODT, DIACHI, MSHH, NAMHH. Trước khi insert dữ liệu cần kiểm tra MSHH đã tồn tại trong table HOCHAM chưa, nếu chưa thì trả về giá trị 0.

```
CREATE PROCEDURE THEM_GV @MSGV INT, @TENG V NVARCHAR(30),  
@SODT VARCHAR(10), @DIACHI NVARCHAR(50), @MSHH INT, @NAMHH  
SMALLDATETIME
```

```
AS
```

```
BEGIN
```

```
    IF EXISTS (SELECT * FROM HOCHAM WHERE MSHH = @MSHH)
```

```
        INSERT INTO GIAOVIEN (MSGV, TENG V, SODT, DIACHI,  
MSHH, NAMHH) VALUES (@MSGV, @TENG V, @SODT, @DIACHI, @MSHH,  
@NAMHH)
```

```
    ELSE
```

```
        PRINT '0'
```

```
END
```

--2A Tham số vào là MSGV, TENG V, SODT, DIACHI, MSHH, NAMHH. Trước khi insert dữ liệu cần kiểm tra MSGV trong table GIAOVIEN có trùng không, nếu trùng thì trả về giá trị 0.

```
CREATE PROCEDURE THEM_GV2 @MSGV INT, @TENG V NVARCHAR(30),  
@SODT VARCHAR(10), @DIACHI NVARCHAR(50), @MSHH INT, @NAMHH  
SMALLDATETIME
```

```
AS
```

```
BEGIN
```

```
    IF EXISTS (SELECT * FROM GIAOVIEN WHERE MSGV = @MSGV)
```

```

INSERT INTO GIAOVIEN (MSGV, TENGV, SODT, DIACHI,
MSHH, NAMHH) VALUES (@MSGV, @TENGV, @SODT, @DIACHI, @MSHH,
@NAMHH)
ELSE
PRINT '0'
END

```

--3A Giống (1) và (2) kiểm tra xem MSGV có trùng không? MSHH có tồn tại chưa? Nếu MSGV trùng thì trả về 0. Nếu MSHH chưa tồn tại trả về 1, ngược lại cho insert dữ liệu

```

CREATE PROCEDURE THEM_GV3 @MSGV INT, @TENGV NVARCHAR(30),
@SODT VARCHAR(10), @DIACHI NVARCHAR(50), @MSHH INT, @NAMHH
SMALLDATETIME
AS
BEGIN
IF EXISTS (SELECT * FROM GIAOVIEN WHERE MSGV = @MSGV)
PRINT '0'
ELSE
IF NOT EXISTS (SELECT * FROM HOCHAM WHERE MSHH =
@MSHH)
PRINT '1'
ELSE
INSERT INTO GIAOVIEN (MSGV, TENGV, SODT,
DIACHI, MSHH, NAMHH) VALUES (@MSGV, @TENGV, @SODT, @DIACHI,
@MSHH, @NAMHH)
END

```

--- B PROCEDURE CO THAM SO RA

--1B Đưa vào TENHV trả ra: Số GV thỏa học vị, nếu không tìm thấy trả về 0.

```

CREATE PROCEDURE SO_GV_HV @TENHV NVARCHAR(20), @SLGV INT
OUTPUT
AS
BEGIN
    SELECT @SLGV = COUNT(MSGV)
    FROM GV_HV_CN JOIN HOCVI ON GV_HV_CN.MSHV = HOCVI.MSHV
    WHERE TENHV = @TENHV
END

```

--2B Đưa vào MSDT cho biết: Điểm trung bình của đề tài, nếu không tìm thấy trả về 0.

```

CREATE PROCEDURE TINH_DTB @MSDT CHAR(6), @DIEM FLOAT OUTPUT
AS
BEGIN
    SELECT (SUM(GV_HDDT.DIEM) + SUM(GV_UVDT.DIEM) +
    SUM(GV_PBDT.DIEM)) / (COUNT(GV_HDDT.DIEM) + COUNT(GV_UVDT.DIEM)
    + COUNT(GV_PBDT.DIEM))
    FROM DETAI AS DT JOIN GV_HDDT ON DT.MSDT = GV_HDDT.MSDT
    JOIN GV_UVDT ON DT.MSDT = GV_UVDT.MSDT
    JOIN GV_PBDT ON DT.MSDT = GV_PBDT.MSDT
    WHERE DT.MSDT = @MSDT
    IF (@DIEM IS NULL)
        SET @DIEM = 0
END

```

---C TRIGGER

--1C Tạo Trigger thỏa mãn điều kiện khi xóa một đề tài sẽ xóa các thông tin liên quan.

```

CREATE TRIGGER DEL_DETAI
ON DETAI

```

INSTEAD OF DELETE

AS

BEGIN

DECLARE @MSDT CHAR(6)

SELECT @MSDT = MSDT FROM deleted

DELETE FROM SV_DETAI

WHERE MSDT = @MSDT

DELETE FROM GV_HDDT

WHERE MSDT = @MSDT

DELETE FROM GV_PBDT

WHERE MSDT = @MSDT

DELETE FROM GV_UVDT

WHERE MSDT = @MSDT

DELETE FROM HOIDONG_DT

WHERE MSDT = @MSDT

DELETE FROM DETAI

WHERE MSDT = @MSDT

END

**--2C Tạo Trigger thỏa mãn ràng buộc là một hội đồng không quá 10 đề tài. Dùng
“Group by” có được không? Giải thích.**

CREATE TRIGGER 1HDKHONGQUA10DT

ON HOIDONG_DT

FOR INSERT, UPDATE

AS

BEGIN

IF (SELECT COUNT(HOIDONG_DT.MSHD) FROM HOIDONG_DT, inserted

WHERE HOIDONG_DT.MSHD = inserted.MSHD) > 10

BEGIN

```
PRINT N'Một hội đồng không quá 10 đề tài'
ROLLBACK TRANSACTION
END
END
```

Trong trường hợp này, ta có thể sử dụng GROUP BY đối với MSHD bởi nó sẽ không ảnh hưởng đến kết quả của câu truy vấn:

```
SELECT COUNT(HOIDONG_DT.MSHD)
FROM HOIDONG_DT, inserted
WHERE HOIDONG_DT.MSHD = inserted.MSHD
GROUP BY MSHD
```

Đối với MSĐT và QUYETDINH thì ta không thể sử dụng GROUP BY bởi nó sẽ thay đổi kết quả của câu truy vấn.

---D FUNCTION

--1D Viết hàm tính điểm trung bình của một đề tài. Giá trị trả về là điểm trung bình ứng với mã số đề tài nhập vào.

```
CREATE FUNCTION F_TINHDTB (@MSDT CHAR(6))
RETURNS FLOAT
AS
BEGIN
    DECLARE @DIEM FLOAT
    SELECT @DIEM = (SUM(GV_HDDT.DIEM) + SUM(GV_UVDT.DIEM) +
SUM(GV_PBDT.DIEM)) / (COUNT(GV_HDDT.DIEM) + COUNT(GV_UVDT.DIEM)
+ COUNT(GV_PBDT.DIEM))
    FROM DETAI AS DT JOIN GV_HDDT ON DT.MSDT = GV_HDDT.MSDT
    JOIN GV_UVDT ON DT.MSDT = GV_UVDT.MSDT
    JOIN GV_PBDT ON DT.MSDT = GV_PBDT.MSDT
    WHERE DT.MSDT = @MSDT
```

```

        IF (@DIEM IS NULL)
            SET @DIEM = 0
        RETURN @DIEM
    END

```

--2D Trả về kết quả của đề tài theo MSDT nhập vào. Kết quả là DAT nếu như điểm trung bình từ 5 trở lên, và KHONGDAT nếu như điểm trung bình dưới 5.

```

CREATE FUNCTION F_KETQUADT (@MSDT CHAR(6))
RETURNS CHAR(10)
AS
BEGIN
    DECLARE @DIEM FLOAT
    DECLARE @KETQUA CHAR(10)

    SELECT @DIEM = (SUM(GV_HDDT.DIEM) + SUM(GV_UVDT.DIEM) +
SUM(GV_PBDT.DIEM)) / (COUNT(GV_HDDT.DIEM) + COUNT(GV_UVDT.DIEM)
+ COUNT(GV_PBDT.DIEM))

    FROM DETAI AS DT JOIN GV_HDDT ON DT.MSDT = GV_HDDT.MSDT
    JOIN GV_UVDT ON DT.MSDT = GV_UVDT.MSDT
    JOIN GV_PBDT ON DT.MSDT = GV_PBDT.MSDT
    WHERE DT.MSDT = @MSDT

    IF (@DIEM IS NULL)
        SET @DIEM = 0
    IF (@DIEM >= 5)
        SET @KETQUA = 'DAT'
    ELSE IF (@DIEM < 5)
        SET @KETQUA = 'KHONGDAT'
    RETURN @KETQUA
END

```

---E CURSOR

**--Viết Cursor tính điểm trung bình cho từng đề tài. Sau đó lưu kết quả vào bảng
DETAI_DIEM.**

CREATE TABLE DETAI_DIEM

(
MSDT CHAR(6) PRIMARY KEY,
DIEMTB FLOAT
)

--Khai báo con trỏ

DECLARE p CURSOR FOR SELECT MSDT, TENDT, dbo.F_TINHDTB(MSDT) AS
DIEM_TB FROM DETAI

--Thiết lập giá trị con trỏ

DECLARE @A CHAR(6), @T NVARCHAR(30), @C FLOAT

OPEN p

--Gán giá trị cho các biến

---Lấy dòng đầu tiên

FETCH p INTO @A, @T, @C

---Kiểm tra có dữ liệu không, nếu có thì duyệt tiếp các dòng tiếp theo

WHILE (@@FETCH_STATUS = 0)

BEGIN

INSERT INTO DETAI_DIEM (MSDT, DIEMTB) VALUES
(@A,ROUND(@C,2))

FETCH NEXT FROM p INTO @A, @T, @C

END

--Đóng con trỏ

CLOSE p

DEALLOCATE p