

**THỰC HÀNH QUẢN LÝ THÔNG TIN**  
**IT103 – BTTH2**

Họ và tên	Nguyễn Lê Tấn Quang
MSSV	21522512
Lớp	IE103.O11.2

**Bài tập**

**A. STORED PROCEDURES VỚI THAM SỐ VÀO**

**1. Tham số vào là MSGV, TENGV, 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.

**Trả lời:**

```
-- Tạo stored procedure
CREATE PROCEDURE InsertGiaoVien
@MSGV NVARCHAR(8),
@TENGV NVARCHAR(30),
@SODT VARCHAR(10),
@DIACHI NCHAR(50),
@MSHH INT,
@NAMHH INT
AS
BEGIN
-- Kiểm tra xem MSHH đã tồn tại trong table HOCHAM chưa
DECLARE @HochamCount INT
SELECT @HochamCount = COUNT(*) FROM HOCHAM WHERE
MSHH = @MSHH

-- Nếu MSHH không tồn tại, trả về giá trị 0
IF @HochamCount = 0
BEGIN
PRINT N'MSHH không tồn tại trong table HOCHAM.'
RETURN 0
END
```

```

-- Nếu MSHH tồn tại, thực hiện chèn dữ liệu vào bảng GIAOVIEN
INSERT INTO GIAOVIEN (MSGV, TENG V, SODT, DIACHI, MSHH,
NAMHH)
VALUES (@MSGV, @TENG V, @SODT, @DIACHI, @MSHH,
@NAMHH)

PRINT 'Dữ liệu đã được chèn thành công.'
RETURN 1
END

```

– Test

```

EXEC InsertGiaoVien ('00206', N'Nguyễn Lê Tấn Quang', '123456789',
N'Hồ Chí Minh', 3, 2000)

```

### Kết quả:



- 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.

### Trả lời:

```

CREATE PROCEDURE InsertGiaoVienWithCheck
@MSGV NVARCHAR(8),
@TENG V NVARCHAR(30),
@SODT VARCHAR(10),
@DIACHI NCHAR(50),
@MSHH INT,
@NAMHH INT
AS
BEGIN

```

```

-- Kiểm tra xem MSGV đã tồn tại trong table GIAOVIEN chưa

```

```

IF EXISTS (SELECT 1 FROM GIAOVIEN WHERE MSGV =
@MSGV)
BEGIN
    PRINT 'MSGV đã tồn tại trong table GIAOVIEN.'
    RETURN 0
END

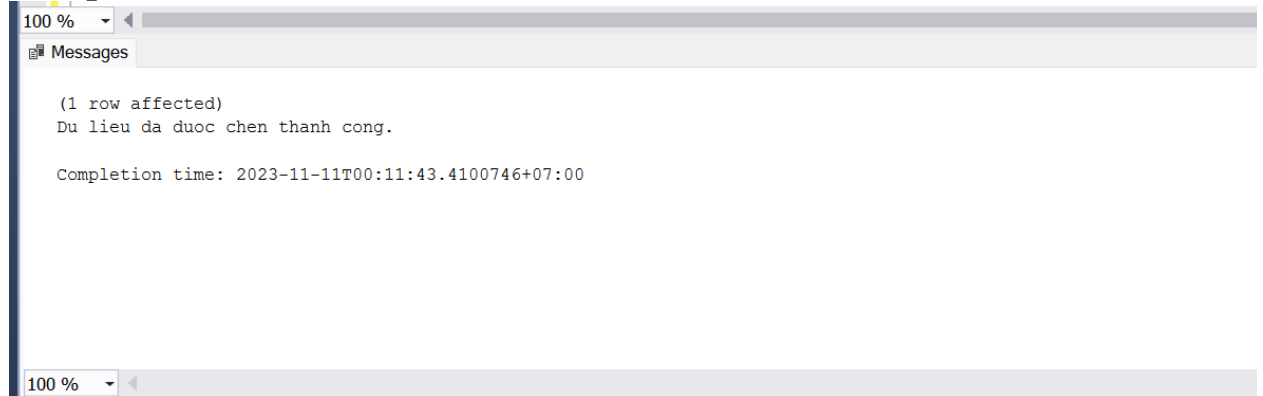
-- Neu MSGV không trùng, thực hiện chèn dữ liệu vào bảng GIAOVIEN
INSERT INTO GIAOVIEN (MSGV, TENG, SODT, DIACHI, MSHH,
NAMHH)
VALUES (@MSGV, @TENG, @SODT, @DIACHI, @MSHH,
@NAMHH)

PRINT 'Dữ liệu đã được chèn thành công.'
RETURN 1
END

--Test
EXEC InsertGiaoVienWithCheck ('00207', N'Nguyễn Văn A', '987654321',
N'Hà Nội', 2, 2001)

```

### Kết quả:



3. 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.

```

-- Tạo stored procedure
CREATE PROCEDURE InsertGiaoVienWithCheckBoth
    @MSGV NVARCHAR(8),

```

```

@TENGV NVARCHAR(30),
@SODT VARCHAR(10),
@DIACHI NCHAR(50),
@MSHH INT,
@NAMHH INT
AS
BEGIN
    -- Kiểm tra xem MSGV có trùng không
    IF EXISTS (SELECT 1 FROM GIAOVIEN WHERE MSGV =
@MSGV)
    BEGIN
        PRINT 'MSGV đã tồn tại trong table GIAOVIEN.'
        RETURN 0
    END

    -- Kiểm tra xem MSHH đã tồn tại trong table HOCHAM chưa
    IF NOT EXISTS (SELECT 1 FROM HOCHAM WHERE MSHH =
@MSHH)
    BEGIN
        PRINT 'MSHH không tồn tại trong table HOCHAM.'
        RETURN 1
    END

    -- Nếu cả MSGV không trùng và MSHH tồn tại, thực hiện chèn dữ
    liệu vào bảng GIAOVIEN
    INSERT INTO GIAOVIEN (MSGV, TENGV, SODT, DIACHI,
MSHH, NAMHH)
    VALUES (@MSGV, @TENGV, @SODT, @DIACHI, @MSHH,
@NAMHH)

    PRINT 'Dữ liệu đã được chèn thành công.'
    RETURN 2
    END

    --Test
    EXEC InsertGiaoVienWithCheckBoth '00208', N'Nguyễn Văn C',
'123123123', N'Đà Nẵng', 3, 2002

```

**Kết quả:**

#### Messages

MSHH không tồn tại trong table HOCHAM.

Completion time: 2023-11-11T00:32:41.9935453+07:00

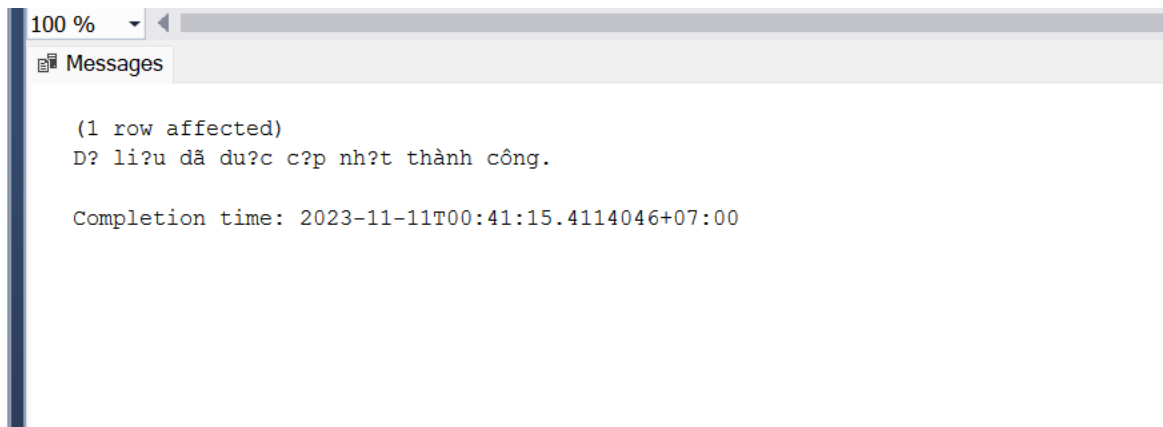
4. Đưa vào MSDT cũ, TENDT mới. Hãy cập nhật tên đề tài mới với mã đề tài cũ không đổi nếu không tìm thấy trả về 0, ngược lại cập nhật và trả về 1.

```
CREATE PROCEDURE UpdateTenDeTai
    @MSDT CHAR(6),
    @TENDT_NEW NVARCHAR(30)
AS
BEGIN
    -- Kiểm tra xem MSDT có tồn tại trong bảng DETAI không
    IF NOT EXISTS (SELECT 1 FROM DETAI WHERE MSDT
        = @MSDT)
    BEGIN
        PRINT 'MSDT không tồn tại trong bảng DETAI.'
        RETURN 0
    END

    -- Nếu MSDT tồn tại, cập nhật TENDT mới
    UPDATE DETAI
    SET TENDT = @TENDT_NEW
    WHERE MSDT = @MSDT

    PRINT 'Dữ liệu đã được cập nhật thành công.'
    RETURN 1
END
--Test
EXEC UpdateTenDeTai '97001', N'Quản Lý Thông Tin'
```

**Kết quả:**



5. Tham số đưa vào MSSV, TENSX mới, DIACHI mới thử lược dùng để cập nhật sinh viên trên, nếu không tìm thấy trả về 0, ngược lại cập nhật và trả về 1

-- Tạo stored procedure

CREATE PROCEDURE UpdateThongTinSinhVien

@MSSV CHAR(8),

@TENSX\_MOI NVARCHAR(30),

@DIACHI\_MOI NCHAR(50)

AS

BEGIN

-- Kiểm tra xem MSSV có tồn tại trong bảng SinhVien không

IF NOT EXISTS (SELECT 1 FROM SinhVien WHERE MSSV =

@MSSV)

BEGIN

PRINT 'MSSV không tồn tại trong bảng SinhVien.'

RETURN 0

END

-- Nếu MSSV tồn tại, cập nhật thông tin mới

UPDATE SinhVien

SET TENSX = @TENSX\_MOI,

DIACHI = @DIACHI\_MOI

WHERE MSSV = @MSSV

PRINT 'Thông tin sinh viên đã được cập nhật thành công.'

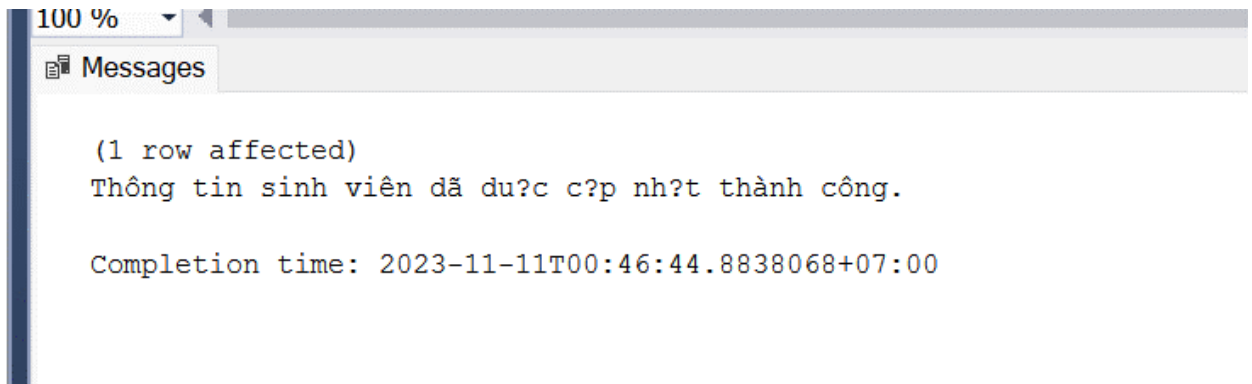
RETURN 1

END

--Test

EXEC UpdateThongTinSinhVien '13520001', N'Nguyễn Văn Anh', N'Quận 1, TP.HCM'

### Kết quả:



## B. STORED PROCEDURES VỚI THAM SỐ VÀO

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

-- Tạo stored procedure

CREATE PROCEDURE CountGiaoVienByHocVii

@TENHV NVARCHAR (20),

@SoLuongGiaoVien INT OUTPUT

AS

BEGIN

-- Khởi tạo biến đếm

SET @SoLuongGiaoVien = 0

-- Đếm số lượng giáo viên thỏa mãn học vị

SELECT @SoLuongGiaoVien = COUNT(\*)

FROM GIAOVIEN GV

INNER JOIN GV\_HV\_CN GH ON GV.MSGV = GH.MSGV

INNER JOIN HOCVI HV ON GH.MSHV = HV.MSHV

WHERE HV.TENHV = @TENHV

-- Trả về số lượng giáo viên có học vị tương ứng

SELECT @SoLuongGiaoVien AS SoLuongGiaoVien

END

--Test

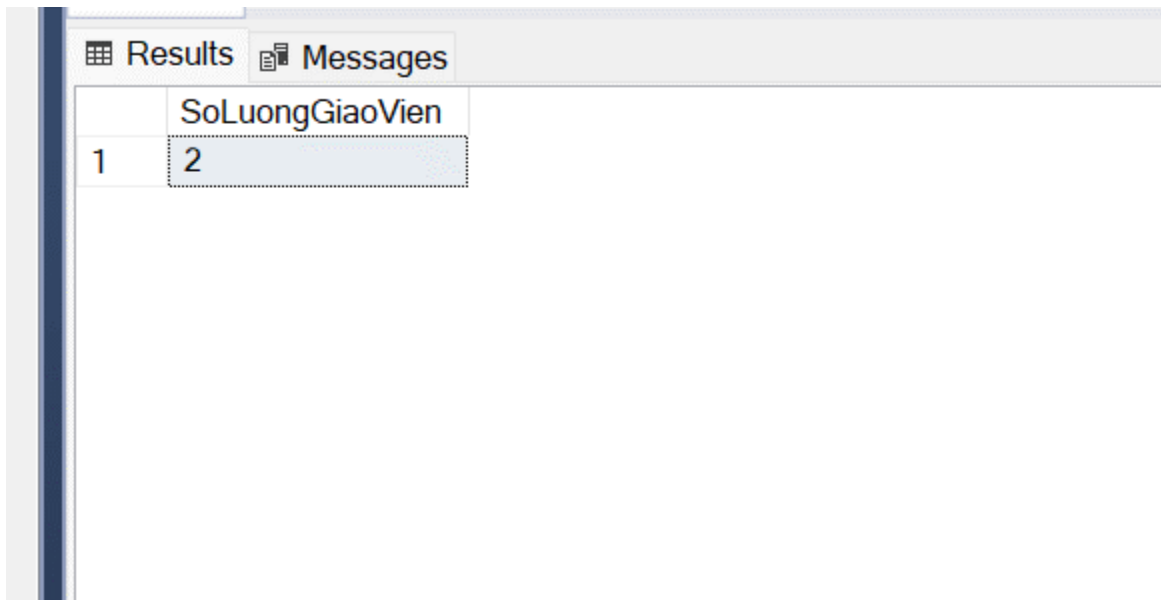
DECLARE @SoLuongGiaoVien INT

DECLARE @TenHocVi NVARCHAR(20) = N'Kỹ sư'

EXEC CountGiaoVienByHocVi @TenHocVi, @SoLuongGiaoVien

OUTPUT

**Kết quả:**



The screenshot shows the 'Results' tab in SQL Server Enterprise Manager. It displays a single row of data with the column header 'SoLuongGiaoVien' and a value of '2'.

	SoLuongGiaoVien
1	2

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

-- Tạo stored procedure

CREATE PROCEDURE AverageDiemByDeTai

@MSDT CHAR(6),

@DiemTrungBinh FLOAT OUTPUT

AS

BEGIN

-- Khởi tạo biến điểm trung bình

SET @DiemTrungBinh = 0

-- Tính điểm trung bình của đề tài

SELECT @DiemTrungBinh = AVG(DIEM)



```

FROM GV_HDDT
WHERE MSDT = @MSDT

-- Trả về điểm trung bình của đề tài
SELECT @DiemTrungBinh AS DiemTrungBinh
END

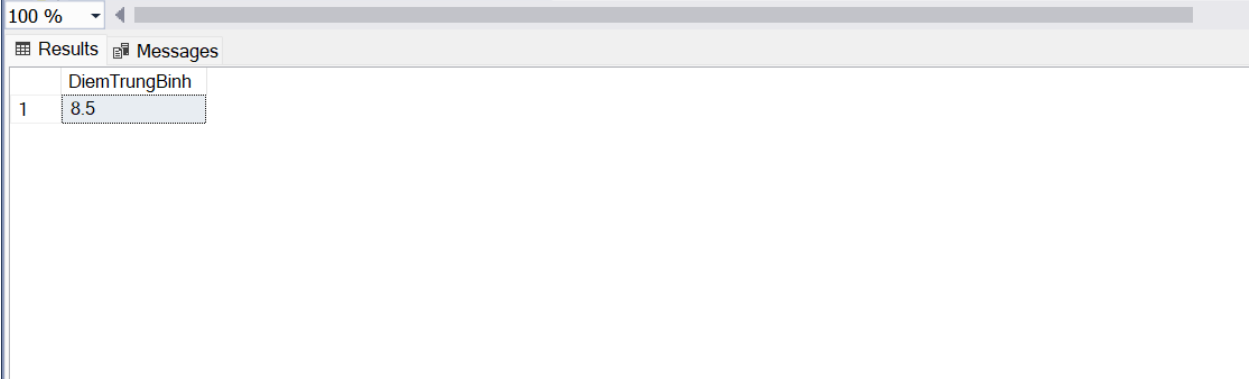
--test
-- Gọi stored procedure với tham số đầu vào và tham số ra
DECLARE @DiemTrungBinh FLOAT
DECLARE @MaDeTai CHAR(6) = '97001'

EXEC AverageDiemByDeTai @MaDeTai, @DiemTrungBinh OUTPUT

-- Hiển thị kết quả
PRINT 'Điểm trung bình của đề tài có MSDT = ' + @MaDeTai + ' là: ' +
CAST(@DiemTrungBinh AS NVARCHAR(10))

```

### Kết quả:



	DiemTrungBinh
1	8.5

3. Đưa vào TENG V trả ra: SĐT của giáo viên đó, nếu không tìm thấy trả về 0. Nếu trùng tên thì có báo lỗi không? Tại sao? Làm sao để hiện thông báo có bao nhiêu giáo viên trùng tên và trả về các SĐT.

```

-- Tạo stored procedure
CREATE PROCEDURE FindSDTByTenGV
    @TENG V NVARCHAR(30),
    @SDT NVARCHAR(10) OUTPUT,
    @SoLuongGiaoVienTrungTen INT OUTPUT
AS

```

```

BEGIN
    -- Khởi tạo biến SDT và số lượng giáo viên trùng tên
    SET @SDT = '0'
    SET @SoLuongGiaoVienTrungTen = 0

    -- Lưu kết quả tìm kiếm vào bảng tạm thời
    CREATE TABLE #TempResults
    (
        SDT NVARCHAR(10)
    )

    -- Tìm kiếm giáo viên theo tên
    INSERT INTO #TempResults (SDT)
    SELECT SODT
    FROM GIAOVIEN
    WHERE TENGV = @TENGV

    -- Đếm số lượng giáo viên trùng tên
    SELECT @SoLuongGiaoVienTrungTen = COUNT(*)
    FROM #TempResults

    -- Trả về SDT và số lượng giáo viên trùng tên
    SELECT @SDT = SDT
    FROM #TempResults

    -- Xóa bảng tạm thời sau khi sử dụng
    DROP TABLE #TempResults
END

--test
DECLARE @SDTGiaoVien NVARCHAR(10)
DECLARE @TenGiaoVien NVARCHAR(30) = N'Nguyễn Văn An' -- Đặt
giá trị TENGV muốn truy vấn
DECLARE @SoLuongGiaoVienTrungTen INT

EXEC FindSDTByTenGV @TenGiaoVien, @SDTGiaoVien OUTPUT,
@SoLuongGiaoVienTrungTen OUTPUT

-- Kiểm tra số lượng giáo viên trùng tên và hiển thị kết quả
IF @SoLuongGiaoVienTrungTen > 1

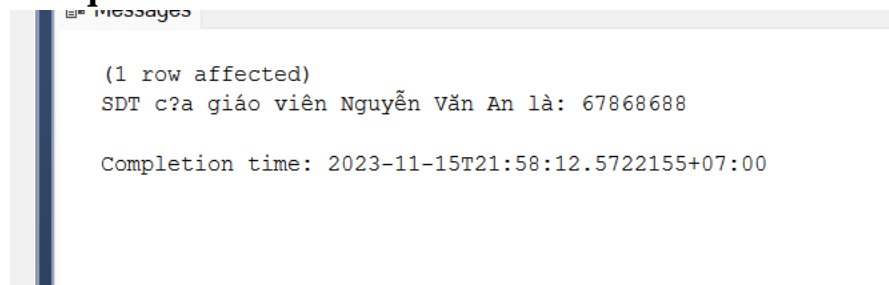
```

```

BEGIN
    PRINT 'Có ' + CAST(@SoLuongGiaoVienTrungTen AS
NVARCHAR(10)) + ' giáo viên trùng tên.'
    PRINT 'Danh sách SDT của các giáo viên:'
    PRINT @SDTGiaoVien
END
ELSE IF @SoLuongGiaoVienTrungTen = 1
BEGIN
    PRINT 'SDT của giáo viên ' + @TenGiaoVien + ' là: ' + @SDTGiaoVien
END
ELSE
BEGIN
    PRINT 'Không tìm thấy giáo viên có tên ' + @TenGiaoVien
END

```

### Kết quả:



Trong stored procedure trên thì sẽ không có báo lỗi khi có nhiều giáo viên trùng tên. Thay vào đó, nó lưu tất cả các SDT của giáo viên trùng tên vào bảng tạm thời và trả về thông tin về số lượng giáo viên trùng tên cùng với danh sách SDT.

#### 4. Đưa vào MSHD cho biết: Điểm trung bình các đề tài của hội đồng đó.

```

-- Tạo stored procedure
CREATE PROCEDURE DiemTrungBinhHoiDong
    @MSHD INT,
    @DiemTrungBinh FLOAT OUTPUT
AS
BEGIN
    -- Khởi tạo biến điểm trung bình
    SET @DiemTrungBinh = 0.0

```

```

-- Kiểm tra xem hội đồng có tồn tại không
IF NOT EXISTS (SELECT 1 FROM HOIDONG WHERE MSHD =
@MSHD)
BEGIN
    PRINT 'Hội đồng không tồn tại.'
    RETURN
END

-- Tính điểm trung bình các đề tài của hội đồng
SELECT @DiemTrungBinh = AVG(DIEM)
FROM GV_HDDT
WHERE MSĐT IN (SELECT MSĐT FROM HOIDONG_DT WHERE
MSHD = @MSHD)

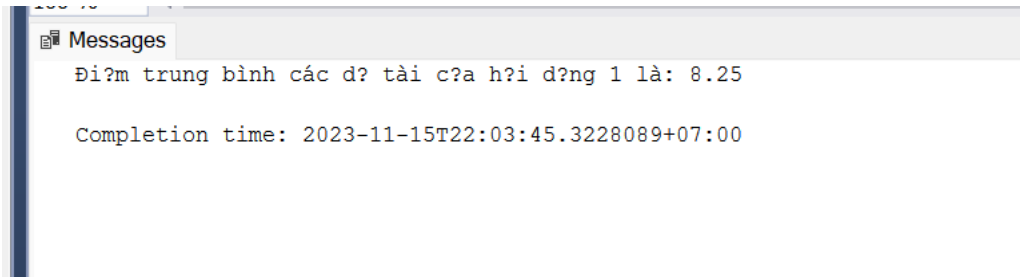
-- Kiểm tra xem có đề tài nào trong hội đồng không
IF @DiemTrungBinh IS NULL
BEGIN
    PRINT 'Hội đồng không có đề tài nào.'
END
ELSE
BEGIN
    PRINT 'Điểm trung bình các đề tài của hội đồng ' + CAST(@MSHD
AS NVARCHAR(10)) + ' là: ' + CAST(@DiemTrungBinh AS
NVARCHAR(10))
END
END

--test
-- Gọi stored procedure với tham số đầu vào và tham số ra
DECLARE @DiemTrungBinh FLOAT
DECLARE @MSHDHoiDong INT = 1 -- Đặt giá trị MSHD muốn truy vấn

EXEC DiemTrungBinhHoiDong @MSHDHoiDong, @DiemTrungBinh
OUTPUT

```

**Kết quả:**



## C. TRIGGER

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

**CREATE TRIGGER Delete\_DeTai ON DeTai FOR DELETE**

**AS**

**BEGIN**

**DELETE FROM GV\_HDDT**

**WHERE MSĐT IN (SELECT MSĐT FROM deleted)**

**DELETE FROM GV\_PBDT**

**WHERE MSĐT IN (SELECT MSĐT FROM deleted)**

**DELETE FROM GV\_UVDT**

**WHERE MSĐT IN (SELECT MSĐT FROM deleted)**

**DELETE FROM HOIDONG\_DT**

**WHERE MSĐT IN (SELECT MSĐT FROM deleted)**

**DELETE FROM SV\_DETAI**

**WHERE MSDT IN (SELECT MSDT FROM deleted)**

**END**

**2. 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 TG\_2**

**ON HOIDONG\_DT**

**AFTER INSERT**

**AS**

**BEGIN**

**DECLARE @Count int**

**SELECT @Count = COUNT(\*)**

**FROM HOIDONG\_DT**

**GROUP BY MSHD**

**HAVING COUNT(\*) > 10**

**IF @Count > 0**

**BEGIN**

**print ('Hội đồng không được quá 10 đề tài')**

**ROLLBACK TRANSACTION**

**END**

**END**

**Giải thích:** Dùng được, gom nhóm để kiểm tra số lượng đề tài cho mỗi Hội đồng theo MSHD, nếu mà count lớn hơn 0 là có hội đồng có đề tài lớn hơn 10=> rollback lại.

**Kết quả:**

```

tNU
INSERT INTO HOIDONG_DT VALUES(1, '97006', N'Được')
INSERT INTO HOIDONG_DT VALUES(1, '97007', N'Được')
INSERT INTO HOIDONG_DT VALUES(1, '97008', N'Được')
INSERT INTO HOIDONG_DT VALUES(1, '97009', N'Được')
INSERT INTO HOIDONG_DT VALUES(1, '97010', N'Được')
INSERT INTO HOIDONG_DT VALUES(1, '97011', N'Được')
INSERT INTO HOIDONG_DT VALUES(1, '97012', N'Được')
INSERT INTO HOIDONG_DT VALUES(1, '97013', N'Được')
select * from HOIDONG_DT

100 %
Messages
H71 d7ng không du?c quá 10 d? tải
Msg 3609, Level 16, State 1, Line 386
The transaction ended in the trigger. The batch has been aborted.
Completion time: 2023-11-21T14:23:39.6092349+07:00
```

**3. Tạo Trigger thỏa mãn ràng buộc là một đề tài không quá 3 sinh viên. Dùng “Group by” có được không? Giải thích.**

**CREATE TRIGGER TG\_3**

**ON SV\_DETAI**

**AFTER INSERT**

**AS**

**BEGIN**

**DECLARE @Count int**

**SELECT @Count = COUNT(\*)**

**FROM SV\_DETAI**

**GROUP BY MSDT**

**HAVING COUNT(\*) > 3**

**IF @Count > 0**

**BEGIN**

**print ('Một đề tài không quá 3 sinh viên')**

**ROLLBACK TRANSACTION**

**END**

**END**

**Giải thích:** Dùng được, gom nhóm để kiểm tra số sinh viên tài cho mỗi đề tài, nếu mà count lớn hơn 0 là có đề tài có số sinh viên > 3=> rollback lại

**Kết quả:**

```
INSERT INTO SV_DETAI VALUES('13520007','97005')
INSERT INTO SV_DETAI VALUES('13520008','97005')
delete sv_detai where mssv='13520008'

INSERT INTO STNHVTEN VALUES('13520007' N' Lê Thúy Hằng' '09766

00 %
Messages
M?t d? tài không quá 3 sinh viên
Msg 3609, Level 16, State 1, Line 426
The transaction ended in the trigger. The batch has been aborted.

Completion time: 2023-11-21T14:33:08.8208748+07:00
```

**4. Tạo Trigger thỏa mãn ràng buộc là một giáo viên muốn có học hàm PGS phải là tiến sĩ.**

**CREATE TRIGGER TG\_4**

**ON GIAOVIEN**

**AFTER UPDATE, INSERT**

**AS**

**BEGIN**

**IF EXISTS(**

**SELECT \***

**FROM inserted i**



```

JOIN GV_HV_CN gh ON i.MSGV = gh.MSGV
JOIN HOCVI hv ON gh.MSHV = hv.MSHV
JOIN HOCHAM h ON i.MSHH = h.MSHH
WHERE h.TENHH = 'PHÓ GIÁO SU' AND hv.TENHV != 'Tiến sĩ'
)

BEGIN

PRINT('Giáo viên muốn có học hàm PGS phải là TS')

ROLLBACK TRANSACTION

END

END

```

#### D. FUNCTION

1. 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 FT_1 (@MSDT char(6))
returns float
as
begin
DECLARE @DTB FLOAT, @Count FLOAT

select @DTB = (SUM(DIEM))
from GV_HDDT hd
where hd.MSDT=@MSDT

```

```
select @DTB = @DTB + (SUM(DIEM))  
from GV_PBDT pb  
where pb.MSDT=@MSDT
```

```
select @DTB = @DTB + (SUM(DIEM))  
from GV_UVDT uv  
where uv.MSDT=@MSDT
```

```
select @Count = (Count(DIEM))  
from GV_HDDT hd  
where hd.MSDT=@MSDT
```

```
select @Count = @Count + (Count(DIEM))  
from GV_UVDT uv  
where uv.MSDT=@MSDT
```

```
select @Count = @Count + (Count(DIEM))  
from GV_PBDT pb  
where pb.MSDT=@MSDT
```

```
set @DTB= @DTB/@Count;
```

```
return @DTB;
```

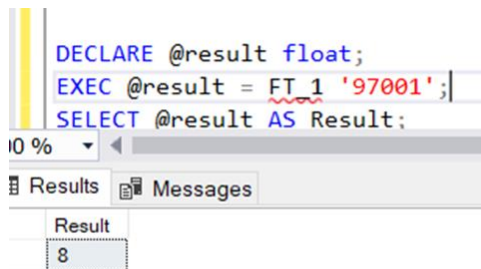
```
end
```

```
DECLARE @result float;
```

```
EXEC @result = FT_1 '97005';
```

```
SELECT @result AS Result;
```

**Kết quả:**



Result
8

**2. 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 FT_2 (@MSDT char(6))
```

```
returns varchar(10)
```

```
as
```

```
begin
```

```
declare @DiemTB float
```

```
set @DiemTB = dbo. FT_1 (@MSDT)
```

```
if (@DiemTB>5)
```

```
return 'Dat'
```

```
RETURN 'KHONG DAT'
```

```
END
```

```

DECLARE @result varchar(10);

EXEC @result = FT_2 '97001';

SELECT @result AS Result;

```

3\*. Đưa vào MSDT, trả về mã số và họ tên của các sinh viên thực hiện đề tài.

-- Tạo Function

```

CREATE FUNCTION GetSinhVienByDeTai
(
    @MSDT CHAR(6)
)
RETURNS TABLE
AS
RETURN
(
    SELECT SV.MSSV, SV.TENSV
    FROM SinhVien SV
    JOIN SV_DeTai SDT ON SV.MSSV = SDT.MSSV
    WHERE SDT.MSDT = @MSDT
);

```

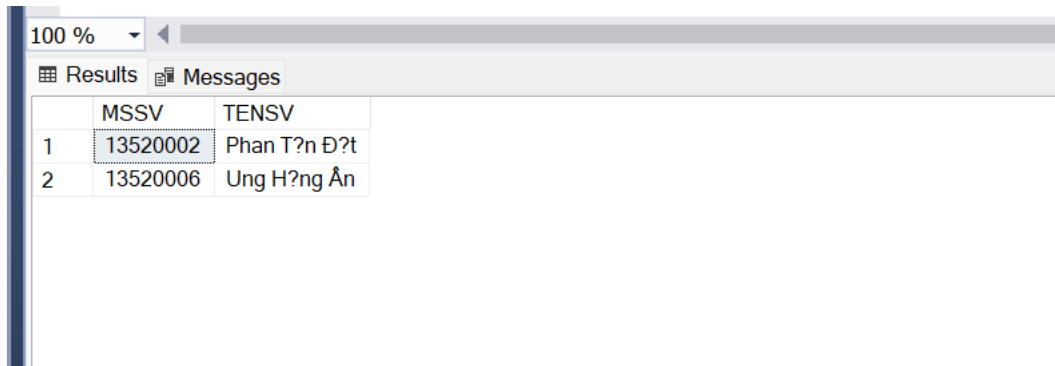
-- Sử dụng Function

```

SELECT * FROM GetSinhVienByDeTai('97005');

```

**Kết quả:**



	MSSV	TENSX
1	13520002	Phan T?n Đ?t
2	13520006	Ung H?ng Ân

## E. CURSOR

Tạo một bảng tên là DETAI\_DIEM. Cấu trúc bảng như sau:  
DETAI\_DIEM(MSx, DIEMTB)

1. 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 biến và cursor
DECLARE @MSDT CHAR(6)
DECLARE @DIEMTB FLOAT

DECLARE Cursor_Detai CURSOR FOR
SELECT MSDT
FROM Detai;

-- Mở cursor
OPEN Cursor_Detai;

-- Lặp qua từng đề tài và tính điểm trung bình
FETCH NEXT FROM Cursor_Detai INTO @MSDT;

WHILE @@FETCH_STATUS = 0
BEGIN
    -- Tính điểm trung bình cho đề tài
    SELECT @DIEMTB = AVG(DIEM)
    FROM GV_HDDT
    WHERE MSDT = @MSDT;

    -- Kiểm tra xem có điểm nào không
    IF @DIEMTB IS NOT NULL
    BEGIN
        -- Thêm dòng vào bảng DETAI_DIEM
        INSERT INTO DETAI_DIEM (MSDT, DIEMTB)
```

```

VALUES (@MSDT, @DIEMTB);

PRINT 'Đã tính điểm trung bình cho đề tài ' + @MSDT;
END
ELSE
BEGIN
    PRINT 'Không có điểm cho đề tài ' + @MSDT;
END

-- Lấy đề tài tiếp theo
FETCH NEXT FROM Cursor_Detai INTO @MSDT;
END

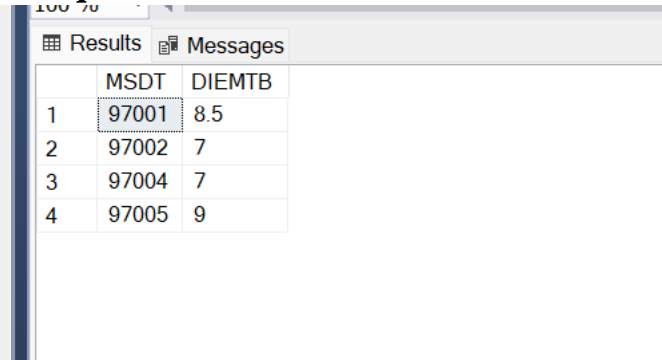
-- Đóng cursor
CLOSE Cursor_Detai;

-- Hủy cursor
DEALLOCATE Cursor_Detai;

--Test
Select * from DETAI_DIEM;

```

### Kết quả:



	MSDT	DIEMTB
1	97001	8.5
2	97002	7
3	97004	7
4	97005	9

## 2. Gộp các bước xử lý của Cursor ở câu 1 vào một Stored Procedure.

```

CREATE PROCEDURE TinhDiemTrungBinh
AS
BEGIN

```

```

DECLARE @MSDT CHAR(6)
DECLARE @DIEMTB FLOAT

DECLARE Cursor_Detai CURSOR FOR
SELECT MSDT
FROM Detai;

OPEN Cursor_Detai;

FETCH NEXT FROM Cursor_Detai INTO @MSDT;

WHILE @@FETCH_STATUS = 0
BEGIN
    SELECT @DIEMTB = AVG(DIEM)
    FROM GV_HDDT
    WHERE MSDT = @MSDT;

    -- Kiểm tra xem có điểm nào không
    IF @DIEMTB IS NOT NULL
    BEGIN
        -- Kiểm tra xem đã tồn tại trong bảng DETAI_DIEM hay chưa
        IF NOT EXISTS (SELECT 1 FROM DETAI_DIEM WHERE
MSDT = @MSDT)
        BEGIN
            INSERT INTO DETAI_DIEM (MSDT, DIEMTB)
            VALUES (@MSDT, @DIEMTB);

            PRINT 'Đã tính điểm trung bình cho đề tài ' + @MSDT;
        END
        ELSE
        BEGIN
            PRINT 'Đã tồn tại điểm cho đề tài ' + @MSDT;
        END
    END
    ELSE
    BEGIN
        PRINT 'Không có điểm cho đề tài ' + @MSDT;
    END

    FETCH NEXT FROM Cursor_Detai INTO @MSDT;

```

END

CLOSE Cursor\_Detai;

DEALLOCATE Cursor\_Detai;  
END

--Test

-- Gọi stored procedure

EXEC TinhDiemTrungBinh;

Kết quả:

```
Messages
Đã t?n t?i di?m cho d? tài 97001
Đã t?n t?i di?m cho d? tài 97002
Không có di?m cho d? tài 97003
Đã t?n t?i di?m cho d? tài 97004
Đã t?n t?i di?m cho d? tài 97005
Không có di?m cho d? tài 97006

Completion time: 2023-11-15T23:37:29.7389897+07:00
```

3. Tạo thêm cột XEPLOAI có kiểu là NVARCHAR (20) trong bảng DETAI\_DIEM, viết Cursor cập nhật kết quả xếp loại cho mỗi đề tài như sau: + "Xuất sắc": điểm trung bình từ 9 đến 10. + "Giỏi": điểm trung bình từ 8 đến 9. + "Khá": điểm trung bình từ 7 đến 8. + "Trung bình": điểm trung bình từ 5 đến 6 + "Không đạt": điểm trung bình dưới 5.

-- Thêm cột XEPLOAI vào bảng Detai\_Diem

ALTER TABLE Detai\_Diem  
ADD XEPLOAI NVARCHAR(20);

CREATE PROCEDURE CapNhatXepLoai  
AS



```

BEGIN
    DECLARE @MSDT CHAR(6)
    DECLARE @DIEMTB FLOAT
    DECLARE @XEPLOAI NVARCHAR(20)

    -- Cursor cho bảng Detai_Diem
    DECLARE Cursor_DetaiDiem CURSOR FOR
    SELECT MSDT, DIEMTB
    FROM Detai_Diem;

    OPEN Cursor_DetaiDiem;

    FETCH NEXT FROM Cursor_DetaiDiem INTO @MSDT, @DIEMTB;

    WHILE @@FETCH_STATUS = 0
    BEGIN
        -- Xác định xếp loại dựa trên điểm trung bình
        IF @DIEMTB >= 9 AND @DIEMTB <= 10
            SET @XEPLOAI = N'Xuất sắc';
        ELSE IF @DIEMTB >= 8 AND @DIEMTB < 9
            SET @XEPLOAI = N'Giỏi';
        ELSE IF @DIEMTB >= 7 AND @DIEMTB < 8
            SET @XEPLOAI = N'Khá';
        ELSE IF @DIEMTB >= 5 AND @DIEMTB < 6
            SET @XEPLOAI = N'Trung bình';
        ELSE
            SET @XEPLOAI = N'Không đạt';

        -- Cập nhật cột XEPLOAI trong bảng Detai_Diem
        UPDATE Detai_Diem
        SET XEPLOAI = @XEPLOAI
        WHERE MSDT = @MSDT;

        -- Lấy đề tài tiếp theo
        FETCH NEXT FROM Cursor_DetaiDiem INTO @MSDT,
@DIEMTB;
    END

    CLOSE Cursor_DetaiDiem;

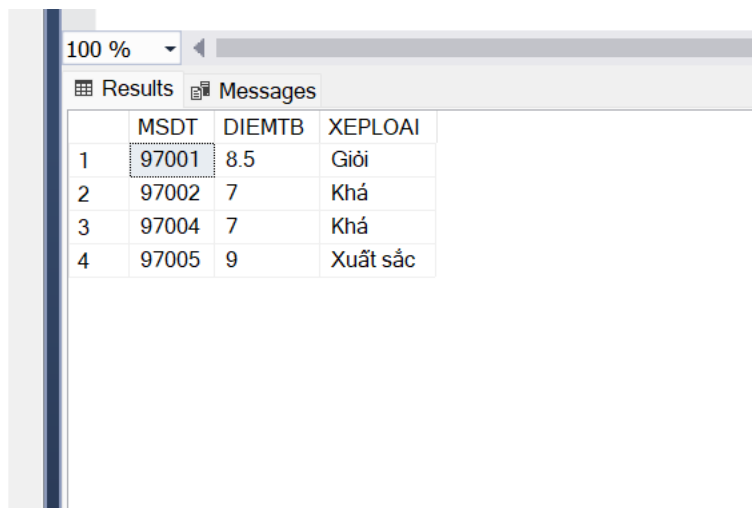
```

```
DEALLOCATE Cursor_DetaiDiem;  
END
```

```
EXEC CapNhatXepLoai;
```

```
Select * from Detai_Diem;
```

Kết quả:



The screenshot shows a SQL Server Enterprise Manager interface. At the top, there is a zoom dropdown set to '100 %' and a scroll bar. Below this are two tabs: 'Results' (active) and 'Messages'. The 'Results' tab displays a table with the following data:

	MSDT	DIEMTB	XEPLAI
1	97001	8.5	Giỏi
2	97002	7	Khá
3	97004	7	Khá
4	97005	9	Xuất sắc