

MSSV: 22521329	BÁO CÁO BÀI TẬP THỰC HÀNH MÔN QUẢN LÝ THÔNG TIN TUẦN 2
Họ và tên: Nguyễn Cao Thắng	
Lớp: IE103.O22.2	

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.

```

CREATE PROCEDURE CHECK_MSHH
(
    @MSGV INT,
    @TENGV NVARCHAR(30),
    @SODT VARCHAR(10),
    @DIACHI NVARCHAR(50),
    @MSHH INT,
    @NAMHH SMALLDATETIME
)
AS
BEGIN
    DECLARE @Exist INT
    SELECT @Exist = COUNT(*) FROM HOCHAM WHERE @MSHH = MSHH
    IF @Exist = 0
    BEGIN
        PRINT N'Không tồn tại MSHH'
        RETURN 0
    END
    ELSE
    BEGIN
        PRINT N'Tồn tại MSHH'
        INSERT INTO GIAOVIEN (MSGV, TENGV, SODT, DIACHI, MSHH,
NAMHH)
        VALUES (@MSGV, @TENGV, @SODT, @DIACHI, @MSHH, @NAMHH)
        RETURN 1
    END
END

```

```
EXEC CHECK_MSHH 00108, N'Thắng', N'Phú Yên', '35353535', 1, '2004'
EXEC CHECK_MSHH 00109, N'Thắng', N'Phú Yên', '35353535', 6, '2004'
```

%

Messages

Tồn tại MSHH

(1 row affected)

Không tồn tại MSHH

Completion time: 2024-03-27T14:32:24.5828310+07:00

2. Tham số vào là MSGV, TENGV, 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 CHECK_MSGV
(
    @MSGV INT,
    @TENGV NVARCHAR(30),
    @SODT VARCHAR(10),
    @DIACHI NVARCHAR(50),
    @MSHH INT,
    @NAMHH SMALLDATETIME
)
AS
BEGIN
    DECLARE @Exist INT
    SELECT @Exist = COUNT(*) FROM GIAOVIEN WHERE @MSGV = MSGV
    IF @Exist = 0
    BEGIN
        PRINT N'Không bị trùng MSGV'
        INSERT INTO GIAOVIEN (MSGV, TENGV, SODT, DIACHI, MSHH,
NAMHH)
VALUES (@MSGV, @TENGV, @SODT, @DIACHI, @MSHH, @NAMHH)
        RETURN 1
    END
    ELSE
    BEGIN
        PRINT N'Bị trùng MSGV'
        RETURN 0
    END
END
```

```
EXEC CHECK_MSGV 00109, N'Thắng', N'Phú Yên', '35353535', 1, '2004'
EXEC CHECK_MSGV 00201, N'Thắng', N'Phú Yên', '35353535', 1, '2004'
```

%

Messages

Không bị trùng MSGV

(1 row affected)

Bị trùng MSGV

Completion time: 2024-03-27T14:42:16.0832265+07:00

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.

```
CREATE PROCEDURE CHECK_MSHH_MSGV
(
    @MSGV INT,
    @TENGV NVARCHAR(30),
    @SODT VARCHAR(10),
    @DIACHI NVARCHAR(50),
    @MSHH INT,
    @NAMHH SMALLDATETIME
)
AS
BEGIN
    DECLARE @ExistMSHH INT, @ExistMSGV INT
    SELECT @ExistMSHH = COUNT(*) FROM HOCHAM WHERE @MSHH = MSHH
    IF @ExistMSHH = 0
    BEGIN
        PRINT N'Không tồn tại MSHH'
        RETURN 0
    END
    SELECT @ExistMSGV = COUNT(*) FROM GIAOVIEN WHERE @MSGV = MSGV
    IF @ExistMSGV > 0
    BEGIN
        PRINT N'Bị trùng MSGV'
        RETURN 0
    END
    IF @ExistMSHH > 0 AND @ExistMSGV = 0
    BEGIN
        PRINT N'Dữ liệu hợp lệ, cho phép nhập vào.'
        INSERT INTO GIAOVIEN (MSGV, TENGV, SODT, DIACHI, MSHH,
NAMHH)
VALUES (@MSGV, @TENGV, @SODT, @DIACHI, @MSHH, @NAMHH)
        RETURN 1
    END
END
```

END

```
EXEC CHECK_MSHH_MSGV 00109, N'Thắng', N'Phú Yên', '35353535', 6, '2004'  
EXEC CHECK_MSHH_MSGV 00201, N'Thắng', N'Phú Yên', '35353535', 1, '2004'  
EXEC CHECK_MSHH_MSGV 00109, N'Thắng', N'Phú Yên', '35353535', 1, '2004'
```

%

Messages

Không tồn tại MSHH
Bị trùng MSGV
Dữ liệu hợp lệ, cho phép nhập vào.

(1 row affected)

Completion time: 2024-03-27T15:16:45.5048710+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 UPDATE_DETAI  
(  
    @MSDT CHAR(6),  
    @TENDT NVARCHAR(30)  
)  
AS  
BEGIN  
    IF NOT EXISTS (SELECT * FROM DETAI WHERE @MSDT = MSDT)  
    BEGIN  
        PRINT N'Không tồn tại MSDT'  
        RETURN 0  
    END  
    ELSE  
    BEGIN  
        UPDATE DETAI SET TENDT = @TENDT WHERE MSDT = @MSDT  
        PRINT N'Đã cập nhật tên đề tài mới'  
        RETURN 1  
    END  
END
```

```
EXEC UPDATE_DETAI 97008, 'Ứng dụng dự báo thời tiết'
EXEC UPDATE_DETAI 97001, 'Ứng dụng dự báo thời tiết'
```

% ▾ ◀

Messages

Không tồn tại MSDT

(1 row affected)

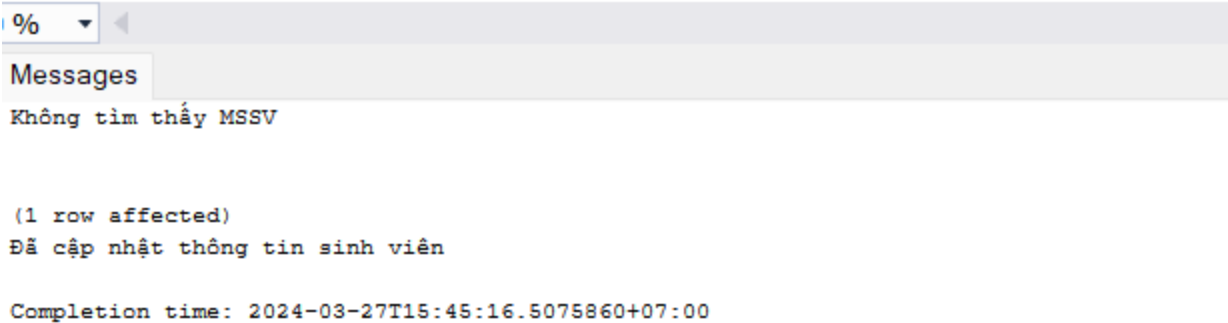
Đã cập nhật tên đề tài mới

Completion time: 2024-03-27T15:29:07.3660271+07:00

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

```
CREATE PROCEDURE UPDATE_SINHVIEN
(
    @MSSV CHAR(8),
    @TENSX NVARCHAR(30),
    @DIACHI NVARCHAR(50)
)
AS
BEGIN
    IF EXISTS (SELECT * FROM SINHVIEN WHERE @MSSV = MSSV)
    BEGIN
        UPDATE SINHVIEN SET TENSX = @TENSX WHERE MSSV = @MSSV
        UPDATE SINHVIEN SET DIACHI = @DIACHI WHERE MSSV = @MSSV
        PRINT N'Đã cập nhật thông tin sinh viên'
        RETURN 1
    END
    ELSE
    BEGIN
        PRINT N'Không tìm thấy MSSV'
        RETURN 0
    END
END
```

```
EXEC UPDATE_SINHVIEN 13520050, N'Nguyễn Cao Thắng', N'QUẬN 9'
EXEC UPDATE_SINHVIEN 13520001, N'Nguyễn Cao Thắng', N'QUẬN 9'
```

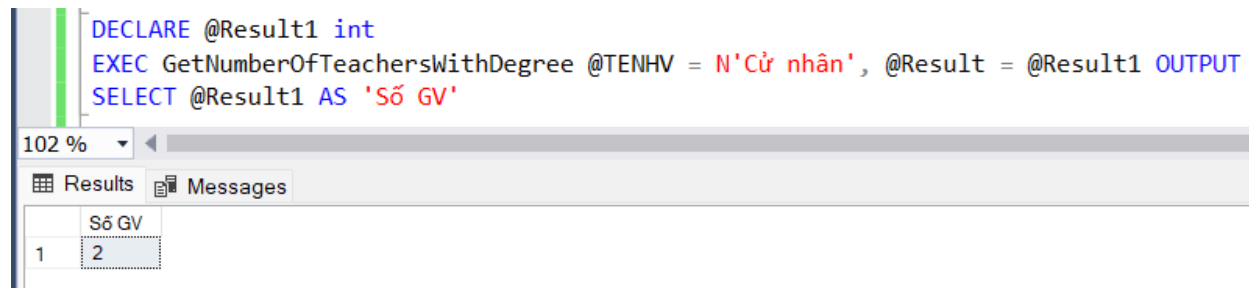


B. STORED PROCEDURES VỚI THAM SỐ VÀO VÀ RA

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

```
CREATE PROCEDURE GetNumberOfTeachersWithDegree
    @TENHV nvarchar(20),
    @Result int OUTPUT
AS
BEGIN
    SELECT @Result = COUNT(DISTINCT MSGV)
    FROM HOCVI HV
    INNER JOIN GV_HV_CN GVCN ON HV.MSHV = GVCN.MSHV
    WHERE HV.TENHV = @TENHV

    IF @Result IS NULL
        SET @Result = 0
END
GO
```



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

```
CREATE PROCEDURE DTB_DE_TAI
    @MSDT CHAR(6),
    @DIEMTB NUMERIC(5,2) OUT
AS
```

```

        IF EXISTS (SELECT * FROM DETAI WHERE @MSDT = MSDT)
BEGIN
    DECLARE @TONGDIEM FLOAT = 0, @TONGGV FLOAT = 0
    IF EXISTS (SELECT * FROM GV_HDDT WHERE @MSDT = MSDT)
    BEGIN
        DECLARE @DIEM1 FLOAT = 0
        SELECT @DIEM1 = DIEM FROM GV_HDDT WHERE @MSDT = MSDT
        SET @TONGDIEM = @TONGDIEM + @DIEM1
        SET @TONGGV = @TONGGV + 1
    END
    IF EXISTS (SELECT * FROM GV_PBDT WHERE @MSDT = MSDT)
    BEGIN
        DECLARE @DIEM2 FLOAT = 0
        SELECT @DIEM2 = DIEM FROM GV_PBDT WHERE @MSDT = MSDT
        SET @TONGDIEM = @TONGDIEM + @DIEM2
        SET @TONGGV = @TONGGV + 1
    END
    IF EXISTS (SELECT * FROM GV_UVDT WHERE @MSDT = MSDT)
    BEGIN
        DECLARE @DIEM3 FLOAT = 0, @TONGUV FLOAT = 0;
        SELECT @DIEM3 = SUM(DIEM) FROM GV_UVDT WHERE @MSDT = MSDT
        SET @TONGDIEM = @TONGDIEM + @DIEM3
        SELECT @TONGUV = COUNT(MSGV) FROM GV_UVDT WHERE @MSDT = MSDT
        SET @TONGGV = @TONGGV + @TONGUV
    END
    IF (@TONGGV = 0)
    BEGIN
        PRINT 'KHONG CO GIAO VIEN'
    END
    ELSE
    BEGIN
        SET @DIEMTB = @TONGDIEM / @TONGGV
        PRINT 'DIEM TRUNG BINH CUA DE TAI ' + @MSDT + ': ' +
CAST(@DIEMTB AS VARCHAR)
    END
    END
    ELSE
    BEGIN
        PRINT 'LOI. MSDT ' + @MSDT + 'KHONG TON TAI'
        RETURN 0
    END
GO

```

```

DECLARE @C_MSĐT CHAR(6), @C_DIEMTB NUMERIC(5,2)
SET @C_MSĐT = '97006'
EXEC P_B2 @C_MSĐT, @C_DIEMTB OUTPUT

```

2 %

Messages

DIEM TRUNG BINH CUA DE TAI 97006 : 8.33

```

DECLARE @C_MSĐT CHAR(6), @C_DIEMTB NUMERIC(5,2)
SET @C_MSĐT = '97008'
EXEC P_B2 @C_MSĐT, @C_DIEMTB OUTPUT

```

2 %

Messages

LOI. MSĐT 97008 KHONG TON TAI

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.

```

CREATE PROCEDURE TIM_SO_DIEN_THOAI
    @TENG V NVARCHAR(30),
    @SODT VARCHAR(10) OUT
AS
    IF EXISTS (SELECT * FROM GIAOVIEN WHERE @TENG = TENG)
BEGIN
    DECLARE @TRUNG TEN INT = 0
    SELECT @TRUNG TEN = COUNT(MSGV) FROM GIAOVIEN WHERE @TENG = TENG
    IF (@TRUNG TEN = 1)
    BEGIN
        SELECT @SODT = SODT FROM GIAOVIEN WHERE @TENG = TENG
        PRINT 'SĐT CUA GIAO VIEN ' + @TENG + ' LA ' + @SODT
    END
    ELSE
    BEGIN
        PRINT 'CO ' + CAST(@TRUNG TEN AS VARCHAR) + ' GIAO VIEN
TRUNG TEN'
        DECLARE @i INT = 0, @ROWNUMBER INT = 0, @TEMP INT = 0;
        WHILE (@i < @TRUNG TEN)
        BEGIN
            WITH GVBACKUP AS
            (

```



```

        SELECT SODT, ROW_NUMBER() OVER (ORDER
BY TENGV) AS ROWNUMBER
        FROM GIAOVIEN WHERE @TENGV = TENGV
    )
    SELECT @SODT = SODT FROM GVBACKUP WHERE
ROWNUMBER = @i + 1
    PRINT 'SDT CUA GIAO VIEN ' + @TENGV + ' LA
    ' + @SODT
    SET @i = @i + 1
END
END
ELSE
BEGIN
    PRINT 'LOI. KHONG TON TAI TEN GIAO VIEN'
END
GO

```

```

DECLARE @C_TENGV NVARCHAR(30), @C_SODT VARCHAR(10)
SET @C_TENGV = N'Nguyễn Văn An'
EXEC TIM_SO_DIEN_THOAI @C_TENGV, @C_SODT OUTPUT

```

12 %

Messages

SDT CUA GIAO VIEN Nguyễn Văn An LA 67868688

```

DECLARE @C_TENGV NVARCHAR(30), @C_SODT VARCHAR(10)
SET @C_TENGV = N'Nguyễn Văn A'
EXEC TIM_SO_DIEN_THOAI @C_TENGV, @C_SODT OUTPUT

```

2 %

Messages

LOI. KHONG TON TAI TEN GIAO VIEN

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

```

CREATE PROCEDURE DTB_HOIDONG
    @MSHD INT,
    @DTB FLOAT OUT
AS
    IF EXISTS (SELECT * FROM HOIDONG_GV WHERE @MSHD = MSHD)
    BEGIN
        SELECT @DTB = AVG(DIEM) FROM (SELECT DIEM FROM HOIDONG_DT A,
GV_HDDT B WHERE A.MSDT = B.MSDT AND @MSHD = A.MSHD
        UNION ALL

```

```

        SELECT DIEM FROM HOIDONG_DT A, GV_PBDT B WHERE A.MSDT = B.MSDT
AND @MSHD = A.MSHD
        UNION ALL
        SELECT DIEM FROM HOIDONG_DT A, GV_UVDT B WHERE A.MSDT = B.MSDT
AND @MSHD = A.MSHD) AS TONG_DIEM
        PRINT 'DIEM TRUNG BINH CAC DE TAI CUA HOI DONG ' + CAST (@MSHD AS
VARCHAR) + ' LA ' + CAST (@DTB AS VARCHAR)
        END
        ELSE
        BEGIN
            PRINT 'LOI. KHONG TON TAI MSHD LA ' + CAST (@MSHD AS
VARCHAR)
        END
    GO

```

```

-- DECLARE @C_MSHD INT, @C_DTB FLOAT
-- SET @C_MSHD = 3
-- EXEC DTB_HOIDONG @C_MSHD, @C_DTB OUT

```

22 %

Messages

DIEM TRUNG BINH CAC DE TAI CUA HOI DONG 3 LA 8

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
    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
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 HD_DT_RESTRICTION
ON HOIDONG_DT FOR INSERT

```

```

AS
BEGIN
    DECLARE @MSHD INT
    SELECT @MSHD = MSHD FROM INSERTED
    IF (SELECT COUNT(MSDT) FROM HOIDONG_DT WHERE MSHD = @MSHD GROUP BY
MSHD) > 4
    BEGIN
        PRINT 'LOI. SO DE TAI TRONG MOT HOI DONG KHONG QUA 10'
        ROLLBACK TRANSACTION
    END
END

```

`SELECT COUNT(MSDT) FROM HOIDONG_DT WHERE MSHD = @MSHD GROUP BY MSHD` kiểm tra xem có bao nhiêu mục nhập hiện có trong bảng HOIDONG_DT với MSHD (Hội Đồng ID) giống với mục đang được thêm vào. GROUP BY MSHD đảm bảo việc đếm được thực hiện cho mỗi giá trị MSHD.

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 SV_DT_RESTRICTION
ON SV_DETAI FOR INSERT
AS
BEGIN
    DECLARE @MSDT CHAR(6)
    SELECT @MSDT = MSDT FROM INSERTED
    IF (SELECT COUNT(MSSV) FROM SV_DETAI WHERE MSDT = @MSDT GROUP BY
MSDT) > 3
    BEGIN
        PRINT 'LOI. SO SINH VIEN THAM GIA MOT DE TAI KHONG QUA 3'
        ROLLBACK TRANSACTION
    END
END
GO

```

`SELECT COUNT(MSSV) FROM SV_DETAI WHERE MSDT = @MSDT GROUP BY MSDT` kiểm tra xem có bao nhiêu sinh viên hiện có trong bảng SV_DETAI với MSDT (Mã Đề Tài) giống với mục đang được thêm vào. GROUP BY MSDT đảm bảo việc đếm được thực hiện cho mỗi giá trị MSDT

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

```

CREATE TRIGGER CHECK_HOCHAM
ON GIAOVIEN FOR UPDATE
AS
BEGIN
    IF EXISTS (SELECT * FROM INSERTED I, HOCHAM HH, GV_HV_CN HV

```

```

WHERE I.MSHH = HH.MSHH AND I.MSGV = HV.MSGV
AND HH.TENHH = 'PHO GIAO SU' AND HV.MSHV <> 4)
BEGIN
    PRINT 'LOI. GIAO VIEN CO HOC HAM PGS PHAI LA TIEN SI'
    ROLLBACK TRANSACTION
END
END

```

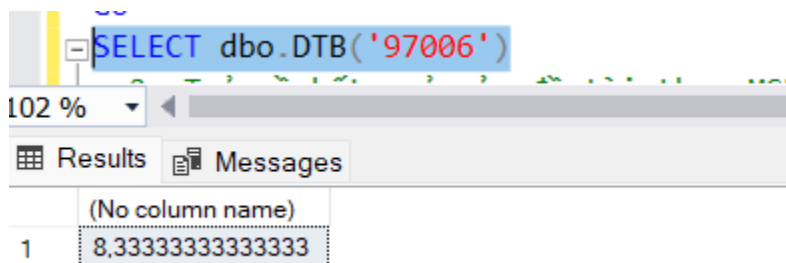
D. FUNCTION

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 DTB(@x1 CHAR(6))
    RETURNS FLOAT
AS
BEGIN
    RETURN (SELECT AVG(DIEM) FROM (SELECT DIEM FROM GV_HDDT WHERE
MSDT = @x1
    UNION ALL
    SELECT DIEM FROM GV_PBDT WHERE MSDT = @x1
    UNION ALL
    SELECT DIEM FROM GV_UVDT WHERE MSDT = @x1)AS A)
END
GO

```



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.

```

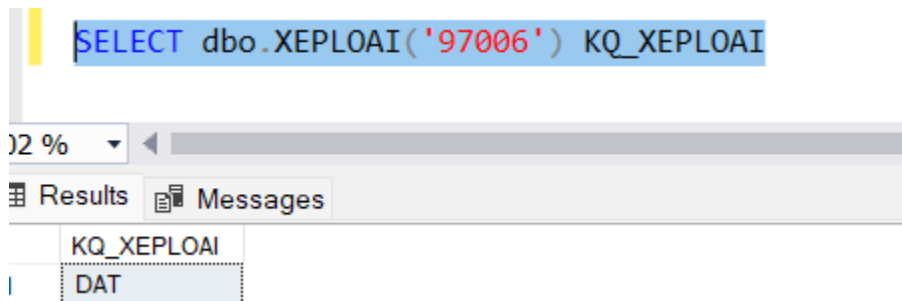
CREATE FUNCTION XEPLOAI(@x1 CHAR(6))
    RETURNS CHAR(10)
AS
BEGIN
    DECLARE @DTB FLOAT
    SELECT @DTB = AVG(DIEM) FROM (SELECT DIEM FROM GV_HDDT WHERE MSDT
= @x1
    UNION ALL
    SELECT DIEM FROM GV_PBDT WHERE MSDT = @x1
    UNION ALL

```

```

SELECT DIEM FROM GV_UVDT WHERE MSDT = (@x1) AS A
IF (@DTB >= 5)
BEGIN
    RETURN 'DAT'
END
IF (@DTB >= 0)
BEGIN
    RETURN 'KHONGDAT'
END
RETURN 'ERROR'
END

```



E. CURSOR

Tạo một bảng tên là DETAI_DIEM. Cấu trúc bảng như sau:

DETAI_DIEM(MSDT, DIEMTB)

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

```

DECLARE CUR_DETAI_DTB CURSOR FOR SELECT DETAI.MSDT FROM DETAI
OPEN CUR_DETAI_DTB
DECLARE @DIEMTB FLOAT, @DETAI_ID CHAR(6)
FETCH NEXT FROM CUR_DETAI_DTB INTO @DETAI_ID
WHILE @@FETCH_STATUS = 0
BEGIN
    SELECT @DIEMTB = (SUM(GV_HDDT.DIEM) + SUM(GV_PBDT.DIEM) +
SUM(GV_UVDT.DIEM)) / (COUNT(GV_HDDT.MSGV) + COUNT(GV_PBDT.MSGV) +
COUNT(GV_UVDT.MSGV))
    FROM DETAI AS A LEFT JOIN GV_HDDT ON A.MSDT = GV_HDDT.MSDT
    LEFT JOIN GV_PBDT ON A.MSDT = GV_PBDT.MSDT
    LEFT JOIN GV_UVDT ON A.MSDT = GV_UVDT.MSDT
    WHERE A.MSDT = @DETAI_ID
    INSERT INTO DETAI_DIEM(MSDT) VALUES (@DETAI_ID)
    UPDATE DETAI_DIEM SET DIEMTB = @DIEMTB WHERE MSDT = @DETAI_ID
    FETCH NEXT FROM CUR_DETAI_DTB INTO @DETAI_ID
END

```

```
CLOSE CUR_DETAI_DTB
DEALLOCATE CUR_DETAI_DTB
```

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

```
CREATE PROCEDURE SP_TinhDiemTB
AS
BEGIN
    DECLARE CUR_DETAI_DTB CURSOR FOR SELECT DETAI.MSDT FROM DETAI
    OPEN CUR_DETAI_DTB
    DECLARE @DIEMTB FLOAT, @DETAI_ID CHAR(6)
    FETCH NEXT FROM CUR_DETAI_DTB INTO @DETAI_ID
    WHILE @@FETCH_STATUS = 0
    BEGIN
        SELECT @DIEMTB = (SUM(GV_HDDT.DIEM) + SUM(GV_PBDT.DIEM) +
SUM(GV_UVDT.DIEM)) / (COUNT(GV_HDDT.MSGV) + COUNT(GV_PBDT.MSGV) +
COUNT(GV_UVDT.MSGV))
        FROM DETAI AS A LEFT JOIN GV_HDDT ON A.MSDT = GV_HDDT.MSDT
        LEFT JOIN GV_PBDT ON A.MSDT = GV_PBDT.MSDT
        LEFT JOIN GV_UVDT ON A.MSDT = GV_UVDT.MSDT
        WHERE A.MSDT = @DETAI_ID
        INSERT INTO DETAI_DIEM(MSDT) VALUES (@DETAI_ID)
        UPDATE DETAI_DIEM SET DIEMTB = @DIEMTB WHERE MSDT =
@DETAI_ID
        FETCH NEXT FROM CUR_DETAI_DTB INTO @DETAI_ID
    END
    CLOSE CUR_DETAI_DTB
    DEALLOCATE CUR_DETAI_DTB
END
GO
```