



BÀI TẬP 1

- SV tái tạo (restore) CSDL **QLNhanVien** từ file “**QLNhanVien.bak**” (đề cho) để làm bài.
- SV thực hiện tất cả các câu đều phải có đủ 2 phần: **tạo** stored procedure và **test** stored procedure.
- Sinh viên phải tự tạo đủ các trường hợp để kiểm thử, chứng minh là kết quả lập trình là đúng trong tất cả các trường hợp đề cho.
- SV phải lưu bài trong ổ **S:** (ổ đĩa chứa đề thi) mới có điểm.

Cho CSDL **Quản lý nhân viên** có 2 bảng như sau:

NhanVien				PhongBan			
Column Name		Data Type	Allow Nulls	Column Name		Data Type	Allow Nulls
	MaNV	int	<input type="checkbox"/>		MaPB	nvarchar(10)	<input type="checkbox"/>
	Ho	nvarchar(50)	<input checked="" type="checkbox"/>		TenPB	nvarchar(50)	<input checked="" type="checkbox"/>
	Ten	nvarchar(20)	<input checked="" type="checkbox"/>		ChiNhanh	nvarchar(50)	<input checked="" type="checkbox"/>
	MaPB	nvarchar(10)	<input checked="" type="checkbox"/>				<input type="checkbox"/>
			<input type="checkbox"/>				<input type="checkbox"/>

NhanVien(MaNV, Ho, Ten, #MaPB)

PhongBan(MaPB, TenPB, ChiNhanh)

Khóa chính: là MaNV và MaPB đã được gạch dưới

Khóa ngoại: NhanVien. MaPB là khóa ngoại tham khảo đến PhongBan.MaPB

Dữ liệu của 2 bảng:

MaNV	Ho	Ten	MaPB	MaPB	TenPB	ChiNhanh
1	Nguyễn Văn	Trung	PB01	PB01	Thiết kế	Sài gòn
2	Hồ Thanh	Tùng	PB02	PB02	Bán hàng	Sài gòn
3	Lê Ánh	Hồng	PB03	PB03	Tiếp thị	Sài gòn
4	Trần Thị	Thắm	PB01	PB04	Sản xuất	Hà nội
5	Đặng Trung	Thành	PB02	PB05	Kinh doanh	Hà nội
6	Mai Ngọc	Hòa	PB03	PB06	Kế toán	Hà nội
7	Phạm Văn	Minh	PB04			
8	Nguyễn Thanh	Thảo	PB05			
9	Trần Văn	Quang	PB06			
10	Bành Văn	Thắng	PB04			
11	Đặng Ngọc Thu	Hà	PB05			
12	Lê Văn	Mạnh	PB06			

Dữ liệu kết từ 2 bảng:

MaNV	Ho	Ten	MaPB	TenPB	ChiNhanh
1	Nguyễn Văn	Trung	PB01	Thiết kế	Sài gòn
2	Hồ Thanh	Tùng	PB02	Bán hàng	Sài gòn
3	Lê Ánh	Hồng	PB03	Tiếp thị	Sài gòn
4	Trần Thị	Thắm	PB01	Thiết kế	Sài gòn
5	Đặng Trung	Thành	PB02	Bán hàng	Sài gòn
6	Mai Ngọc	Hòa	PB03	Tiếp thị	Sài gòn
7	Phạm Văn	Minh	PB04	Sản xuất	Hà nội
8	Nguyễn Thanh	Thảo	PB05	Kinh doanh	Hà nội
9	Trần Văn	Quang	PB06	Kế toán	Hà nội
10	Bành Văn	Thắng	PB04	Sản xuất	Hà nội
11	Đặng Ngọc Thu	Hà	PB05	Kinh doanh	Hà nội
12	Lê Văn	Mạnh	PB06	Kế toán	Hà nội

Câu 1: Hãy viết 2 Stored procedure tên: **TaoPM_Ngang_PB** để tạo và lấy dữ liệu cho các phân mảnh của bảng **PhongBan**; và Stored procedure **TaoPM_Ngang_NhanVien** để tạo và lấy dữ liệu cho các phân mảnh của bảng **NhanVien**, biết:

- Bảng **PhongBan** được **phân mảnh ngang chính** dựa vào cột **ChiNhanh**. Biết cột **ChiNhanh** luôn có dữ liệu (NOT NULL) và chỉ có một trong hai giá trị là “**Sài gòn**” và “**Hà nội**”.
- Bảng **NhanVien** được **phân mảnh ngang dẫn xuất** theo các phân mảnh của bảng **PhongBan**.

Câu 2: Tạo stored procedure tên **ThemPB** để thêm 1 hàng dữ liệu vào các **phân mảnh ngang** của bảng **PhongBan**. Các tham số vào là @MaPB, @TenPB, và @ChiNhanh.

Yêu cầu: phải có báo thành công khi thêm dữ liệu thành công, hay báo lỗi chi tiết (print ra màn hình) và không thêm dữ liệu khi gặp các trường hợp ngoại lệ sau:

- Có @MaPB là NULL, hay @TenPB là NULL, hay @ChiNhanh là NULL
- @ChiNhanh chứa giá trị khác “**Sài gòn**” và “**Hà nội**”
- @MaPB đã có (nếu thêm sẽ bị trùng @MaLop)

Sinh viên phải tự tạo đủ các trường hợp để kiểm thử, chứng minh kết quả lập trình là đúng trong tất cả các trường hợp.

Câu 3: Tạo stored procedure tên **SuaPB** để sửa dữ liệu của các **phân mảnh ngang** của bảng **PhongBan** (update) tại 2 cột TenPB và ChiNhanh (không sửa cột MaPB). Các tham số vào là @MaPB, @TenPB, và @ChiNhanh, trong đó @MaPB để xác định hàng dữ liệu cần sửa.

Yêu cầu: phải có báo chi tiết việc đã thực hiện khi sửa dữ liệu thành công, hay báo lỗi chi tiết (print ra màn hình) và không sửa dữ liệu khi gặp các trường hợp ngoại lệ sau:

- Có @MaPB là NULL, hay @TenPB là NULL, hay @ChiNhanh là NULL
- Không tìm thấy @MaPB để sửa dữ liệu
- @ChiNhanh chứa giá trị khác “**Sài gòn**” và “**Hà nội**”

Sinh viên tự tạo đủ các trường hợp để kiểm thử, chứng minh kết quả lập trình là đúng trong tất cả các trường hợp.

Chú ý: nếu sửa phòng ban mà có dời dữ liệu lớp sang phân mảnh khác thì cũng phải dời nhân viên của phòng ban đó sang phân mảnh khác tương ứng. Khi dời dữ liệu thì phải thông báo dời như thế nào trong thông báo sửa dữ liệu thành công.

Câu 4: Hãy viết Stored procedure tên: **TaoPM_Doc_PB** để tạo và lấy dữ liệu cho **2 phân mảnh dọc** từ bảng **PhongBan**, biết hai phân mảnh dọc là:

PhongBan_Doc1(MaPB, TenPB)

PhongBan_Doc2(MaPB, ChiNhanh)

Câu 5: Hãy viết Stored procedure tên: **XemPB_Doc** để lập danh sách tất cả phòng ban từ **2 phân mảnh đọc** của bảng **PhongBan**, danh sách lớp gồm 3 cột: MaPB, TenPB, ChiNhanh.

Câu 6: Tạo stored procedure tên **SuaPB_Doc** để sửa các **phân mảnh đọc của bảng PhongBan** (update) tại 2 cột TenPB và ChiNhanh (không sửa cột MaPB). Các tham số vào là @MaPB, @TenPB, và @ChiNhanh, trong đó @MaPB để xác định hàng dữ liệu cần sửa.

Yêu cầu: phải có báo chi tiết việc đã thực hiện khi sửa dữ liệu thành công, hay báo lỗi chi tiết (print ra màn hình) và không sửa dữ liệu khi gặp các trường hợp ngoại lệ sau:

- Có @MaPB là NULL, hay @TenPB là NULL, hay @ChiNhanh là NULL
- Không tìm thấy @ MaPB để sửa dữ liệu
- Có @ChiNhanh chứa giá trị khác “Sài gòn” và “Hà nội”

Sinh viên phải tự tạo đủ các trường hợp để kiểm thử, chứng minh kết quả lập trình là đúng trong tất cả các trường hợp.

HẾT

-----HƯỚNG DẪN LÀM BÀI-----

Câu 1: Hãy viết 2 Stored procedure

-- Tạo PROC TaoPM_Ngang_PB:

```
CREATE PROC TaoPM_Ngang_PB
AS
BEGIN
    SELECT * INTO PhongBanHN
    FROM PhongBan
    WHERE ChiNhanh =N'Hà nội'

    SELECT * INTO PhongBanSG
    FROM PhongBan
    WHERE ChiNhanh =N'Sài gòn'
END
GO
```

-- Test PROC TaoPM_Ngang_PB:

```
EXEC dbo.TaoPM_Ngang_PB
```

-- Tạo PROC TaoPM_Ngang_NhanVien:

```
CREATE PROC TaoPM_Ngang_NhanVien
```

```

AS
BEGIN
    SELECT * INTO NhanVienHN
    FROM NhanVien
    WHERE MaPB IN (SELECT MaPB FROM PhongBanHN)

    SELECT * INTO NhanVienSG
    FROM NhanVien
    WHERE MaPB IN (SELECT MaPB FROM PhongBanSG)
END
GO

```

```
-- Test PROC TaoPM_Ngang_NhanVien:
```

```
EXEC dbo.TaoPM_Ngang_NhanVien
```

Câu 2: Tạo stored procedure tên ThemPB:

```

-- Tạo PROC ThemPB:
CREATE PROC ThemPB
@MaPB nvarchar(10),
@TenPB nvarchar(50),
@ChiNhanh nvarchar(50)
AS
BEGIN
    IF(@MaPB IS NULL)
        PRINT N'Không thêm dữ liệu được vì không có giá trị mã phòng ban'
    ELSE IF(@TenPB IS NULL)
        PRINT N'Không thêm dữ liệu được vì không có giá trị tên phòng ban'
    ELSE IF(@ChiNhanh IS NULL)
        PRINT N'Không thêm dữ liệu được vì không có giá trị chi nhánh'
    ELSE IF @ChiNhanh NOT IN (N'Hà nội', N'Sài Gòn')
        PRINT N'Không thêm dữ liệu được vì chi nhánh không hợp lệ (không phải Hà
nội hay Sài Gòn)'
    ELSE IF EXISTS (SELECT * FROM PhongBanHN where MaPB =@MaPB) OR EXISTS (SELECT *
FROM PhongBanSG where MaPB =@MaPB)
        PRINT N'Không thêm dữ liệu được vì trùng mã phòng ban'
    ELSE IF @ChiNhanh =N'Hà nội'
        BEGIN
            INSERT INTO PhongBanHN VALUES (@MaPB,@TenPB,@ChiNhanh)
            PRINT N'Thêm dữ liệu thành công vào phân mảnh PhongBanHN'
        END
    ELSE
        BEGIN
            INSERT INTO PhongBanSG VALUES (@MaPB,@TenPB,@ChiNhanh)
            PRINT N'Thêm dữ liệu thành công vào phân mảnh PhongBanSG'
        END
    END
END
GO

```

```

-- Test PROC ThemPB:
EXEC dbo.ThemPB null, N'AA', N'Sài Gòn'
EXEC dbo.ThemPB N'AA', null, N'Sài Gòn'
EXEC dbo.ThemPB N'AA', N'AA', null
EXEC dbo.ThemPB N'AA', N'AA', N'Tây Ninh'
EXEC dbo.ThemPB N'PB01', N'AA', N'Sài Gòn'
EXEC dbo.ThemPB N'PB06', N'AA', N'Sài Gòn'
EXEC dbo.ThemPB N'PB08', N'AA', N'Hà Nội'

```

```
-- Tạo PROC SuaPB:
CREATE PROC SuaPB
    @MaPB nvarchar(10),
    @TenPB nvarchar(50),
    @ChiNhanh nvarchar(50)
AS
BEGIN
    IF (@MaPB IS NULL)
        PRINT N'Không sửa dữ liệu được vì không có giá trị mã PB'
    ELSE IF (@TenPB IS NULL)
        PRINT N'Không sửa dữ liệu được vì không có giá trị tên PB'
    ELSE IF (@ChiNhanh IS NULL)
        PRINT N'Không sửa dữ liệu được vì không có giá trị chi nhánh'
    ELSE IF @ChiNhanh NOT IN (N'Hà nội', N'Sài gòn')
        PRINT N'Không sửa dữ liệu được vì chi nhánh không hợp lệ (không phải Hà nội hay Sài gòn)'
    ELSE IF EXISTS (SELECT * FROM PhongBanHN where MaPB = @MaPB)
        BEGIN
            UPDATE PhongBanHN
            SET TenPB = @TenPB, ChiNhanh = @ChiNhanh
            WHERE MaPB = @MaPB

            IF (@ChiNhanh = N'Sài gòn')
                BEGIN
                    INSERT INTO PhongBanSG
                    SELECT * FROM PhongBanHN WHERE MaPB = @MaPB
                    DELETE PhongBanHN WHERE MaPB = @MaPB

                    INSERT INTO NhanVienSG
                    SELECT * FROM NhanVienHN WHERE MaPB = @MaPB
                    DELETE NhanVienHN WHERE MaPB = @MaPB

                    PRINT N'Dữ liệu đã được cập nhật. Đã chuyển dữ liệu từ phân
mảnh PhongBanHN sang PhongBanSG. Chuyển dữ liệu từ phân mảnh NhanVienHN sang NhanVienSG'
                END
            ELSE --trường hợp (@ChiNhanh = N'Hà nội')
                PRINT N'Dữ liệu đã được cập nhật tại phân mảnh PhongBanHN'
        END
    ELSE IF EXISTS (SELECT * FROM PhongBanSG where MaPB = @MaPB)
        BEGIN
            UPDATE PhongBanSG
            SET TenPB = @TenPB, ChiNhanh = @ChiNhanh
            WHERE MaPB = @MaPB

            IF (@ChiNhanh = N'Hà Nội')
                BEGIN
                    INSERT INTO PhongBanHN
                    SELECT * FROM PhongBanSG WHERE MaPB = @MaPB
                    DELETE PhongBanSG WHERE MaPB = @MaPB

                    INSERT INTO NhanVienHN
                    SELECT * FROM NhanVienSG WHERE MaPB = @MaPB
                    DELETE NhanVienSG WHERE MaPB = @MaPB
                END
            ELSE
                PRINT N'Dữ liệu đã được cập nhật tại phân mảnh PhongBanSG'
        END
END
```

```

        PRINT N'Dữ liệu đã được cập nhật. Đã chuyển dữ liệu từ phân
        mảnh PhongBanSG sang PhongBanHN. Chuyển dữ liệu từ phân mảnh NhanVienSG sang NhanVienHN'
    END
    ELSE--trường hợp (@ChiNhanh = N'Sài Gòn')
        PRINT N'Dữ liệu đã được cập nhật tại phân mảnh PhongBanSG'
    END
    ELSE -- khi không tìm thấy @MaPB ở cả 2 phân mảnh
        PRINT N'Không sửa dữ liệu được vì không tìm thấy giá trị mã phòng ban'
END
GO
-- Test PROC SuaPB:
EXEC dbo.SuaPB null, N'AA', N'Sài Gòn'
EXEC dbo.SuaPB N'AA', null, N'Sài Gòn'
EXEC dbo.SuaPB N'AA', N'AA', null
EXEC dbo.SuaPB N'PB06', N'AA', N'Tây Ninh'
EXEC dbo.SuaPB N'AA', N'AA', N'Hà Nội'
EXEC dbo.SuaPB N'PB07', N'AA', N'Hà Nội'
EXEC dbo.SuaPB N'PB08', N'bb', N'Hà Nội'
-----

```

Câu 4: Hãy viết Stored procedure tên: TaoPM_Doc_PB:

PhongBan_Doc1(MaPB, TenPB)

PhongBan_Doc2(MaPB, ChiNhanh)

```

-- Tạo PROC TaoPM_Doc_PB:
CREATE PROC TaoPM_Doc_PB
AS
BEGIN
    SELECT MaPB, PhongBan.TenPB INTO PhongBan_Doc1
    FROM PhongBan

    SELECT MaPB, PhongBan.ChiNhanh INTO PhongBan_Doc2
    FROM PhongBan
END
GO
-- Test PROC TaoPM_Doc_PB:
EXEC dbo.TaoPM_Doc_PB

```

Câu 5: Hãy viết Stored procedure tên: XemPB_Doc:

```

-- Tạo PROC XemPB_Doc:
CREATE PROC XemPB_Doc
AS
    SELECT PhongBan_Doc1.*, PhongBan_Doc2.ChiNhanh
    FROM PhongBan_Doc1, PhongBan_Doc2
    WHERE PhongBan_Doc1.MaPB = PhongBan_Doc2.MaPB
GO
-- Test PROC XemPB_Doc:
EXEC dbo.XemPB_Doc

```

Câu 6: Tạo stored procedure tên SuaPB_Doc:

```

-- Tạo PROC SuaPB_Doc:
CREATE PROC SuaPB_Doc
@MaPB nvarchar(10),
@TenPB nvarchar(50),
@ChiNhanh nvarchar(50)
AS

```

```

BEGIN
    IF(@MaPB IS NULL)
        PRINT N'Không sửa dữ liệu được vì không có giá trị mã PB'
    ELSE IF(@TenPB IS NULL)
        PRINT N'Không sửa dữ liệu được vì không có giá trị tên PB'
    ELSE IF(@ChiNhanh IS NULL)
        PRINT N'Không sửa dữ liệu được vì không có giá trị chi nhánh'
    ELSE IF @ChiNhanh NOT IN (N'Hà nội', N'Sài Gòn')
        PRINT N'Không sửa dữ liệu được vì chi nhánh không hợp lệ'
    ELSE IF NOT EXISTS (SELECT * FROM PhongBan_Doc1 WHERE MaPB =@MaPB)
        PRINT N'Không sửa dữ liệu được vì không tìm thấy mã phòng ban'
    ELSE
        BEGIN
            UPDATE PhongBan_Doc1
            SET TenPB = @TenPB
            WHERE MaPB = @MaPB
            PRINT N'Đã sửa thành công trong bảng PhongBan_Doc1'

            UPDATE PhongBan_Doc2
            SET ChiNhanh = @ChiNhanh
            WHERE MaPB = @MaPB
            PRINT N'Đã sửa thành công trong bảng PhongBan_Doc2'
        END
    END
GO
-- Test PROC SuaPB_Doc:
EXEC dbo.SuaPB_Doc null, N'AA', N'Sài Gòn'
EXEC dbo.SuaPB_Doc N'AA', null, N'Sài Gòn'
EXEC dbo.SuaPB_Doc N'AA', N'AA', null
EXEC dbo.SuaPB_Doc N'PB06', N'AA', N'Tây Ninh'
EXEC dbo.SuaPB_Doc N'AA', N'AA', N'Hà Nội'
EXEC dbo.SuaPB_Doc N'PB01', N'AA', N'Hà Nội'
EXEC dbo.SuaPB_Doc N'PB02', N'bb', N'Sài Gòn'

```

-----HẾT-----