

**Đáp án**

**ĐỀ THI CUỐI HỌC KỲ 2 (Thi ca 5, ngày 15/05/2019)**

**NĂM HỌC 2018-2019**

**MÔN: CƠ SỞ DỮ LIỆU NÂNG CAO**

**HỆ: HC ĐẠI HỌC LỚP: C81A**

**THỜI GIAN: 90 PHÚT**

**KHÔNG ĐƯỢC SỬ DỤNG TÀI LIỆU**

**Câu 1:** (2 điểm)

Cho lược đồ toàn cục của một bảng:

**NhânViên (MãNV, HọNV,TênNV, GiớiTính, PhòngBan)**

Hãy chỉ ra rõ lỗi khi phân mảnh (vi phạm điều kiện phân mảnh đúng đắn nào và tại sao) và phân mảnh lại cho đúng trong 2 trường hợp sau:

**1.1** Bảng **NhânViên** được phân thành 2 phân mảnh ngang chính là **NhânViênNam** và **NhânViênNữ** theo cột **GiớiTính**. Biết cột **GiớiTính** có 2 giá nhập là “Nam” và “Nữ” hay có thể để trống không nhập liệu. (1 điểm)

**Đáp án đề nghị:**

- Lỗi: phân thành 2 phân mảnh ngang chính là sai, vi phạm điều kiện đầy đủ
- Lý do có lỗi: không lưu được các nhân viên không có dữ liệu trên cột **GiớiTính**.
- Sửa lỗi: phân thành 3 phân mảnh ngang chính **NhânViênNam**, **NhânViênNữ** và **NhânViên\_GiớiTínhNULL**

**1.2** Bảng **NhânViên** được phân thành 2 phân mảnh dọc là: (1 điểm)

**NhânViênDọc1(MãNV, HọNV, TênNV)**

**NhânViênDọc2(HọNV, TênNV, GiớiTính)**

**Đáp án đề nghị:**

- Lỗi: vi phạm điều kiện đầy đủ và vi phạm điều kiện tái tạo
- Lý do có lỗi: vi phạm điều kiện đầy đủ vì thiếu cột **PhòngBan**; vi phạm điều kiện tái tạo vì không lặp lại khóa chính **MãNV**
- Sửa lỗi: phân mảnh lại:

**NhânViênDọc1(MãNV, HọNV, TênNV, PhòngBan)**

**NhânViênDọc2(MãNV, HọNV, TênNV, GiớiTính)**

**Câu 2:** (3 điểm)

Hãy trình bày chi tiết về **tất cả các loại phân mảnh** của các quan hệ trong cơ sở dữ liệu phân bố. Nội dung trình bày đối với từng loại phân mảnh:

- Tên loại phân mảnh
- Mục đích (hay ý nghĩa) của việc phân mảnh theo loại này
- Cho ví dụ minh họa cụ thể việc phân mảnh theo loại này

**Đáp án đề nghị:**

Thang điểm: SV phải trình bày được **6 loại phân mảnh, mỗi loại 0,5 điểm**

1. Ngang chính
2. Ngang dẫn xuất

3. Đọc tách biệt
4. Đọc gom tụ không dư thừa
5. Đọc gom tụ có dư thừa
6. Hỗn hợp

### **Phản giả thiết cho các câu 3, 4, 5, 6:**

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

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

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

### **Các câu hỏi về phân mảnh đọc:**

Biết bảng NhanVien được phân thành 2 phân mảnh đọc:

**NhanVien \_Doc1(MaNV, Ho, Ten)**

**NhanVien \_Doc2(MaNV, MaPB)**

**Câu 3: (1 điểm)** Hãy viết code SQL để tạo Stored procedure tên **XemNV\_Doc** để lập danh sách nhân viên thuộc phòng ban có mã phòng ban là “PB1” từ **2 phân mảnh đọc** trên của bảng **NhanVien**. Danh sách gồm đúng 4 cột: MaNV, Ho, Ten, MaPB.

### **Đáp án đề nghị:**

```
SELECT NhanVien _Doc1.MaNV, Ho, Ten, MaPB
FROM NhanVien _Doc1, NhanVien _Doc2
WHERE NhanVien _Doc1.MaNV = NhanVien _Doc2.MaNV
AND MaPB = 'PB1'
```

Ghi chú: sai cơ bản: 0đ; thiếu điều kiện kết: -0,5đ; mỗi lỗi khác: -0,25đ

**Câu 4: (1,5 điểm)** Hãy viết code SQL để tạo stored procedure tên **SuaNV\_Doc** để sửa dữ liệu (update) trong các **phân mảnh đọc** của bảng **NhanVien** tại 3 cột Ho, Ten, MaPB (không sửa cột MaNV). Các tham số vào là @MaNV, @Ho, @Ten, @MaPB, trong đó @MaNV để xác định hàng dữ liệu cần sửa.

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

- Không tìm thấy @MaNV để sửa dữ liệu
- Có @MaPB chứa giá trị khác “PB1”, “PB2” và “PB3”
- Có đối số @MaNV là NULL, hay @HoNV là NULL, hay @TenNV là NULL, hay @MaPB là NULL

Sinh viên phải viết các câu lệnh thực thi stored procedure để kiểm thử đủ các trường hợp ngoại lệ, để chứng minh kết quả lập trình là đúng trong tất cả các trường hợp.

**Đáp án đề nghị:**

```
CREATE PROC SuaNV_Doc
( @MaNV int
, @HoNV nvarchar(50)
, @TenNV nvarchar(20)
, @MaPB nvarchar(10) )
AS
BEGIN

IF @MaNV IS NULL
    PRINT N'Lỗi không có MaNV, không sửa dữ liệu được!'
ELSE IF @HoNV IS NULL
    PRINT N'Lỗi không có HoNV, không sửa dữ liệu được!'
ELSE IF @TenNV IS NULL
    PRINT N'Lỗi không có TenNV, không sửa dữ liệu được!'
ELSE IF @MaPB IS NULL
    PRINT N'Lỗi không có MaPB, không sửa dữ liệu được!'
ELSE IF (@MaPB NOT IN ("PB1", "PB2", "PB3"))
    PRINT N'Lỗi: dữ liệu của MaPB không đúng, không sửa dữ liệu
được!'
ELSE IF NOT EXISTS(SELECT MaNV FROM NhanVien_Doc1 WHERE
MaNV =@ MaNV)
    PRINT N'Lỗi: không tìm thấy MaNV để sửa dữ liệu!'
ELSE
    BEGIN
        UPDATE NhanVien_Doc1
        SET HoNV = @HoNV,
            TenNV = @TenNV
        WHERE MaNV = @MaNV
        PRINT N'Dã sửa thành công một hàng dữ liệu trong phân mảng
NhanVien_Doc1'

        UPDATE NhanVien_Doc2
        SET MaPB = @MaPB
        WHERE MaNV = @MaNV
        PRINT N'Dã sửa thành công một hàng dữ liệu trong phân mảng
NhanVien_Doc2'
    END
END
```

Ghi chú: sai cơ bản: 0đ; mỗi lỗi khác: -0,25đ

**Các câu hỏi về phân mảng ngang:**

**Câu 5: (1 điểm)** Hãy viết các biểu thức đại số quan hệ để tạo các phân mảng theo miêu tả sau:

- 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 ba giá trị là “**Sài gòn**”, “**Đà Nẵng**” và “**Hà nội**”.
- Bảng **NhanVien** được **phân mảnh ngang dãy xuất** theo các phân mảnh của bảng **PhongBan**.

#### Đáp án đề nghị:

3 phân mảnh ngang chính: (0,5 điểm)

$$\text{PhongBan\_SG} = \sigma_{\text{ChiNhanh}=\text{"Sài gòn"}(\text{PhongBan})}$$

$$\text{PhongBan\_ĐN} = \sigma_{\text{ChiNhanh}=\text{"Đà Nẵng"}(\text{PhongBan})}$$

$$\text{PhongBan\_HN} = \sigma_{\text{ChiNhanh}=\text{"Hà nội"}(\text{PhongBan})}$$

3 phân mảnh ngang dãy xuất: (0,5 điểm)

$$\text{NhanVien\_SG} = \text{NhanVien} | ><_{MaPB} \text{PhongBan\_SG}$$

$$\text{NhanVien\_ĐN} = \text{NhanVien} | ><_{MaPB} \text{PhongBan\_ĐN}$$

$$\text{NhanVien\_HN} = \text{NhanVien} | ><_{MaPB} \text{PhongBan\_HN}$$

**Câu 6: (1,5 điểm)** 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 thông báo chi tiết việc đã thực hiện khi thêm dữ liệu thành công, hay thông 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
  - Bị trùng **@MaPB** đã có
  - **@ChiNhanh** chứa giá trị khác “**Sài gòn**”, “**Đà Nẵng**” và “**Hà nội**”
- Sinh viên phải viết các câu lệnh để kiểm thử đủ các trường hợp ngoại lệ, để chứng minh kết quả lập trình là đúng trong tất cả các trường hợp.

#### Đáp án đề nghị:

CREATE PROC **ThemPB**

```
( @ MaPB nvarchar(10)
  , @ TenPB nvarchar(50)
  , @ ChiNhanh nvarchar(50))
```

AS

BEGIN

IF @ MaPB IS NULL

PRINT N'Lỗi không có MaPB, không thêm phòng ban được!'

ELSE IF @ TenPB IS NULL

PRINT N'Lỗi không có TenPB, không thêm phòng ban được!'

ELSE IF @ ChiNhanh IS NULL

PRINT N'Lỗi không có ChiNhanh, không thêm phòng ban được!'

ELSE IF @ChiNhanh NOT IN ('Sài gòn', 'Đà Nẵng', 'Hà nội')

PRINT N'Lỗi: dữ liệu của ChiNhanh không đúng, không thêm phòng ban được!'

ELSE IF(EXISTS(SELECT MaPB FROM PhongBan\_SG WHERE MaPB = @MaPB) OR EXISTS(SELECT MaPB FROM PhongBan\_ĐN WHERE MaPB = @MaPB) OR EXISTS(SELECT MaPB FROM PhongBan\_HN WHERE MaPB = @MaPB))

```
PRINT N'Lỗi: Bị trùng mã MaPB, không thêm phòng ban được!'
ELSE IF(@ChiNhanh = N'Sài gòn')
    BEGIN
        INSERT INTO PhongBan_SG
            VALUES(@MaPB, @TenPB, @ChiNhanh)
        PRINT N'Thêm dữ liệu thành công vào PhongBan_SG'
    END
ELSE IF(@ChiNhanh = N'Dà nẵng')
    BEGIN
        INSERT INTO PhongBan_DN
            VALUES(@MaPB, @TenPB, @ChiNhanh)
        PRINT N'Thêm dữ liệu thành công vào PhongBan_DN'
    END
ELSE
    BEGIN
        INSERT INTO PhongBan_HN
            VALUES(@MaPB, @TenPB, @ChiNhanh)
        PRINT N'Thêm dữ liệu thành công vào PhongBan_HN'
    END
END
```

Ghi chú: sai cơ bản: 0đ; mỗi lỗi khác: -0,25đ

**HẾT**

**Xác nhận của Trưởng Khoa**