**Bài 01**

1. **Tạo cơ sở dữ liệu DATBAO gồm 3 bảng**

**BAO** (**MaToBao**, TenBao, DiaChi)

**DOCGIA**(**MaDG**, HoTen, NgaySinh, GioiTinh, DiaChi)

**DATMUA** (**MaDG**, **MaToBao**, Quy, Nam, SoLuong, DonGia)

1. **Thêm thông tin vào các bảng**

* Tạo Stored Procedure (SP) với các tham số đầu vào phù hợp.

SP thứ nhất thực hiện chèn dữ liệu vào bảng **BAO**.

SP thứ hai thực hiện chèn dữ liệu vào bảng **DOCGIA**.

SP thứ ba thực hiện chèn dữ liệu vào bảng **DATMUA**.

* Yêu cầu mỗi SP phải kiểm tra tham số đầu vào. Với các cột không chấp nhận thuộc tính Null.
* Với mỗi SP viết 3 lời gọi thành công.

1. **Viết Hàm**

Viết hàm các tham số đầu vào tương ứng với các cột của bảng **Docgia**. Hàm này trả về mã độc giả (giá trị của cột khóa chính của bảng **Docgia**) thỏa mãn các giá trị được truyền tham số.

1. **Tạo View**

Tạo View lưu thông tin của TOP 2 độc giả có số lượng báo lớn nhất gồm: Madg, Hoten, Tenbao, soluong.

1. **Viết thủ tục**

Viết một SP nhận hai tham số đầu vào là khoảng thời gian đặt mua báo. SP này thực hiện thao tác xóa thông tin những tờ báo được đăng trong khoảng thời gian được truyền vào qua các tham số.

**Yêu cầu:** Sử dụng giao dịch trong thân SP, để đảm bảo tính toàn vẹn dữ liệu khi một thao tác xóa thực hiện không thành công.

**Bài 02**

1. **Tạo cơ sở dữ liệu QLHH gồm 4 bảng.**

**Khachhang** (**Makh**, tenkh, diachi, dienthoai, gioitinh)

**Mathang** (**Mamh**, tenmh, dongia)

**DonDH** (**Madh**, **makh** ,ngaydh, ngaygh)

**ChitietDH** (**Madh**, **Mamh**, soluong)

1. **Thêm thông tin vào các bảng**

* Tạo Stored Procedure (SP) với các tham số đầu vào phù hợp.

SP thứ nhất thực hiện chèn dữ liệu vào bảng **Khachhang**.

SP thứ hai thực hiện chèn dữ liệu vào bảng **Mathang**.

SP thứ ba thực hiện chèn dữ liệu vào bảng **Dondh**.

SP thứ tư thực hiện chèn dữ liệu vào bảng **Chitietdh**.

* Yêu cầu mỗi SP phải kiểm tra tham số đầu vào. Với các cột không chấp nhận thuộc tính Null.
* Với mỗi SP viết 3 lời gọi thành công.

1. **Viết Hàm**

Viết hàm các tham số đầu vào tương ứng với các cột: của bảng khách hàng. Hàm này trả về makh (giá trị của cột khóa chính của bảng **khachhang**) thỏa mãn các giá trị được truyền tham số.

1. **Tạo View**

Tạo View lưu thông tin của TOP 2 khách hàng có số tiền lớn nhất gồm các thông tin sau: Tenkh, Tenmh, ngaydh, ngaygh, soluong, dongia, thanhtien.

@MaMH VARCHAR(10),

@TenMH NVARCHAR(50) ,

@DonGia MONEY

@MaKH VARCHAR(5) =NULL,

@TenKH NVARCHAR(50),

@DiaChi NVARCHAR(200),

@SDT VARCHAR(15),

@GioiTinh VARCHAR(5)

@MaDH VARCHAR(10) ,

@MAKH VARCHAR(5),

@NgayDat DATE ,

@NgayGh DATE

@MADH VARCHAR(10),

@MAMH VARCHAR(10),

@SLUONG INT

1. **Viết thủ tục**

Viết một SP nhận một tham số đầu vào kiểu int là số soluong. SP này thực hiện thao tác xóa thông tin của các sinh viên và thông tin điểm thi của chúng, nếu diem tương ứng của sinh viên này lớn hơn điểm được truyền vào.

**Yêu cầu**: Sử dụng giao dịch trong thân SP, để đảm bảo tính toàn vẹn dữ

liệu khi một thao tác xóa thực hiện không thành công.

**Bài 03**

1. **Tạo cơ sở dữ liệu QLHH gồm 4 bảng.**

**Khachhang** (**Makh**, tenkh, diachi, dienthoai, gioitinh)

**Mathang** (**Mamh**, tenmh, dongia)

**DonDH** (**Madh**, ngaydh, ngaygh, **makh**)

**ChitietDH** (**Madh**, **Mamh**, soluong)

1. **Thêm thông tin vào các bảng**

* Tạo Stored Procedure (SP) với các tham số đầu vào phù hợp.

SP thứ nhất thực hiện chèn dữ liệu vào bảng **Khachhang**.

SP thứ hai thực hiện chèn dữ liệu vào bảng **Mathang**.

SP thứ ba thực hiện chèn dữ liệu vào bảng **Dondh**.

SP thứ tư thực hiện chèn dữ liệu vào bảng **Chitietdh**.

* Yêu cầu mỗi SP phải kiểm tra tham số đầu vào. Với các cột không chấp nhận thuộc tính Null.
* Với mỗi SP viết 3 lời gọi thành công.

1. **Viết Hàm**

Viết hàm các tham số đầu vào tương ứng với các cột: của bảng khách hàng. Hàm này trả về makh (giá trị của cột khóa chính của bảng **khachhang**) thỏa mãn các giá trị được truyền tham số.

1. **Tạo View**

Tạo View lưu thông tin của TOP 2 khách hàng có số tiền lớn nhất gồm các thông tin sau: Tenkh, Tenmh, ngaydh, ngaygh, soluong, dongia, thanhtien.

**Yêu cầu**: Sử dụng giao dịch trong thân SP, để đảm bảo tính toàn vẹn dữ

liệu khi một thao tác xóa thực hiện không thành công.

**Bài 04**

1. **Tạo cơ sở dữ liệu QLSV gồm 3 bảng.**

Bảng **SINHVIEN** (**MaSV,** HoTen, NgaySinh, GioiTinh, Lop**)**

Bảng **MONHOC** (**MaMonHoc,** TenMonHoc, SoTinChi**)**

Bảng **DIEM**: (**MaSV, MaMonHoc,** DiemLan1, DiemLan2**)**

1. **Thêm thông tin vào các bảng**

* Tạo Stored Procedure (SP) với các tham số đầu vào phù hợp.

SP thứ nhất thực hiện chèn dữ liệu vào bảng **SINHVIEN**.

SP thứ hai thực hiện chèn dữ liệu vào bảng **MONHOC**.

SP thứ ba thực hiện chèn dữ liệu vào bảng **DIEM**.

* Yêu cầu mỗi SP phải kiểm tra tham số đầu vào. Với các cột không chấp nhận thuộc tính Null.
* Với mỗi SP viết 3 lời gọi thành công.

1. **Viết Hàm**

Viết hàm các tham số đầu vào tương ứng với các cột của bảng **Sinhvien**. Hàm này trả về mã sinh viên (giá trị của cột khóa chính của bảng **Sinhvien**) thỏa mãn các giá trị được truyền tham số.

1. **Tạo View**

Tạo View lưu thông tin của TOP 2 sinh viên có điểm thi lớn nhất gồm: Masv, Hotensv, tenmh, lanthi, diem.

1. **Viết thủ tục**

Viết một SP nhận một tham số đầu vào kiểu int là số DIEM. SP này thực hiện thao tác xóa thông tin của các sinh viên và thông tin điểm thi của chúng, nếu điểm tương ứng của sinh viên này lớn hơn điểm được truyền vào.

**Yêu cầu**: Sử dụng giao dịch trong thân SP, để đảm bảo tính toàn vẹn dữ

liệu khi một thao tác xóa thực hiện không thành công.

**Bài 05**

1. **Tạo cơ sở dữ liệu GIAOHANG gồm 3 bảng.**

**KHACHHANG** (**MaKH,** HoTenKH, DChiKH, GTinh, DThoaiKH**)**

**NHANVIEN** (**MaNV,** HoTenNV, DChiNV, GTinh, DThoaiNV**)**

**DONHANG**: (**MaKH, MaNV,** NgayDH, NgayGH**)**

1. **Thêm thông tin vào các bảng**

* Tạo Stored Procedure (SP) với các tham số đầu vào phù hợp.

SP thứ nhất thực hiện chèn dữ liệu vào bảng **KHACHHANG**.

SP thứ hai thực hiện chèn dữ liệu vào bảng **NHANVIEN**.

SP thứ ba thực hiện chèn dữ liệu vào bảng **DONHANG**.

* Yêu cầu mỗi SP phải kiểm tra tham số đầu vào. Với các cột không chấp nhận thuộc tính Null.
* Với mỗi SP viết 3 lời gọi thành công.

1. **Viết Hàm**

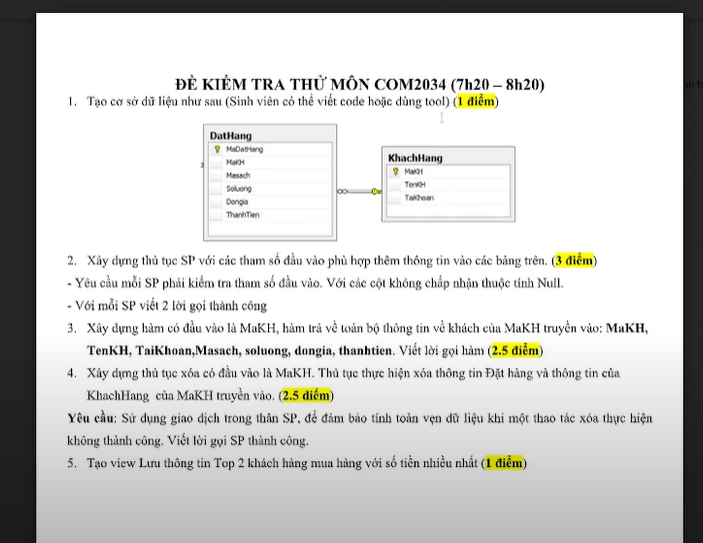
Viết hàm các tham số đầu vào tương ứng với các cột của bảng **KHACHHANG**. Hàm này trả về **MaKH** thỏa mãn các giá trị được truyền tham số.

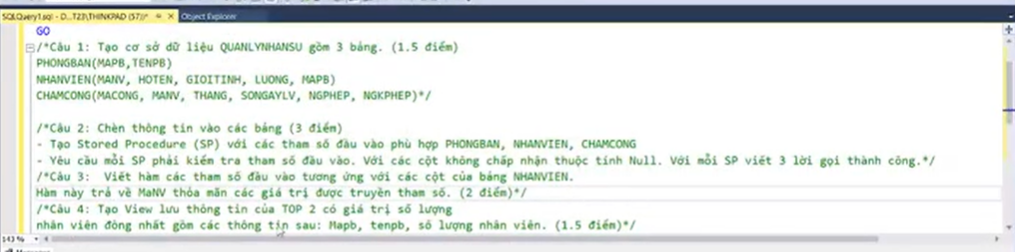
1. **Tạo View**

Tạo View lưu thông tin của TOP 2 đơn hàng có ngày đặt gần nhất: MaKH, HoTenKH, DChiKH, DThoaiKH, MaNV, HoTenNV, DThoaiNV.

1. **Viết thủ tục**

Viết một SP nhận một tham số đầu vào là NgayGH. SP này thực hiện thao tác xóa thông tin của khách hàng và nhân viên giao hàng tương ứng.





**Đề 1:**

Câu 1: Tạo cơ sở dữ liệu **BENHNHAN** gồm 3 bảng. (1.5 điểm)

BENHNHAN (MaBN, HoTenBN, GioiTinh, NgaySinh, DiaChi)

BACSI (MaBS, HoTenBS, ChuyenKhoa, SoDienThoai)

DIEUTRI (MaBN, MaBS, NgayDieuTri, NgayRaVien)

Câu 2: Chèn thông tin vào các bảng (3 điểm)

- Tạo Stored Procedure (SP) với các tham số đầu vào phù hợp BENHNHAN, BACSI, DIEUTRI.

- Yêu cầu mỗi SP phải kiểm tra tham số đầu vào. Với các cột không chấp nhận thuộc tính Null.

- Với mỗi SP viết 3 lời gọi thành công.

Câu 3: Viết Hàm tham số đầu vào tương ứng với các cột của bảng BENHNHAN. Hàm này trả về MaBN thỏa mãn các giá trị được truyền tham số. (2 điểm)

Câu 4: Tạo View lưu thông tin của TOP 2 bệnh nhân có NgayRaVien gần nhất, gồm các thông tin sau: MaBN, HoTenBN, NgayDieuTri, DiaChi, HoTenBS, ChuyenKhoa, SoDienThoai. (1.5 điểm)

Câu 5: Viết một SP nhận một tham số đầu vào là NgayRaVien. SP này thực hiện thao tác xóa thông bệnh nhân và bác sĩ tương ứng. (2 điểm)

Yêu cầu: Sử dụng giao dịch trong thân SP, để đảm bảo tính toàn vẹn dữ liệu khi một thao tác xóa thực hiện không thành công.

Đề 2:

Câu 1: Tạo cơ sở dữ liệu **DANGKYHOC** gồm 3 bảng. (1.5 điểm)

SINHVIEN (MaSV, HoTenSV, GioiTinh, NgSinh, DiaChi, SDThoai)

MONHOC (MaMH, TenMH, STChi)

DANGKY (MaSV, MaMH, NgayDK, HocKy)

Câu 2: Chèn thông tin vào các bảng (3 điểm)

- Tạo Stored Procedure (SP) với các tham số đầu vào phù hợp vào bảng SINHVIEN, MONHOC, DANGKY.

- Yêu cầu mỗi SP phải kiểm tra tham số đầu vào. Với các cột không chấp nhận thuộc tính Null.

- Với mỗi SP viết 3 lời gọi thành công.

Câu 3: Viết hàm các tham số đầu vào tương ứng với các cột của bảng SINHVIEN . Hàm này trả về MaSV thỏa mãn các giá trị được truyền tham số. (2 điểm)

Câu 4: Tạo View lưu thông tin của TOP 2 ngày đăng ký gần nhất, gồm các thông tin sau: MaSV, HoTenSV, DiaChi, SDThoai, MaMH, TenMH, NgayDK, HocKy. (1.5 điểm)

Câu 5: Viết một SP nhận một tham số đầu vào là NgayDK. SP này thực hiện thao tác xóa thông tin sinh viên và môn học tương ứng. (2 điểm)

Yêu cầu: Sử dụng giao dịch trong thân SP, để đảm bảo tính toàn vẹn dữ liệu khi một thao tác xóa thực hiện không thành công.

Đề 3:

Câu 1: Tạo cơ sở dữ liệu **QUANLYNHANSU** gồm 3 bảng. (1.5 điểm)

PHONGBAN(**MAPB**,TENPB)

NHANVIEN(**MANV**, HOTEN, GIOITINH, LUONG, MAPB)

CHAMCONG(**MACONG**, **MANV**, THANG, SONGAYLV, NGPHEP, NGKPHEP)

Câu 2: Chèn thông tin vào các bảng (3 điểm)

- Tạo Stored Procedure (SP) với các tham số đầu vào phù hợp PHONGBAN, NHANVIEN, CHAMCONG

- Yêu cầu mỗi SP phải kiểm tra tham số đầu vào. Với các cột không chấp nhận thuộc tính Null.

Với mỗi SP viết 3 lời gọi thành công.

Câu 3: Viết hàm các tham số đầu vào tương ứng với các cột của bảng NHANVIEN.

Hàm này trả về MaNV thỏa mãn các giá trị được truyền tham số. (2 điểm)

Câu 4: Tạo View lưu thông tin của TOP 2 có giá trị số lượng nhân viên đông nhất gồm các thông tin sau:

Mapb, tenpb, số lượng nhân viên. (1.5 điểm)

Câu 5: Viết một SP nhận một tham số đầu vào là số ngày phép

SP này thực hiện thao tác xóa thông tin chấm công và nhân viên tương ứng nếu tổng số ngày phép lớn hơn giá trị tham số đầu vào. (2 điểm)

--de 9,10,11

**Đề 4:**

Tạo bảng lớp học: mã lớp, tên lớp, sĩ số

Tạo bảng sinh viên : mã sinh viên, tên sinh viên, ngày sinh, địa chỉ, mã lớp

Tạo bảng môn học: mã môn , tên môn

Tạo bảng điểm: id, điểm, mã môn, mã sinh viên

Viết proc insert bảng sinh viên. kiểm tra tuổi sinh viên phải >17

Viết hàm nhận vào mã sinh viên trả ra điểm các môn của sinh viên đó

Viết trigger insert điểm cho sinh viên. Nếu sinh viên đã có điểm môn đó rồi -> thông báo sinh viên đã có điểm

Viết view trả sinh viên có điểm trung bình cao nhất (chia cho tổng số môn)

Viết trigger insert bảng sinh viên -> update bảng lớp học tương ứng

Viết trigger delete bảng sinh viên -> update bảng lớp học tương ứng

Viết proc insert lớp học. Nếu mã lớp tồn tại thì update lại tên lớp

Đề 5:

Tạo bảng sản phẩm : mã sản phẩm, tên sản phẩm, số lượng, giá bán

Tạo bảng giỏ hàng : mã giỏ hàng, người mua, ngày mua

Tạo bảng chi tiết giỏ hàng : mã giỏ hàng, mã sản phầm, số lượng mua

1.viết proc insert cho bảng sản phẩm. nếu sản phẩm đã tồn tại -> sản phẩm đã tồn tại (1đ)

2.Viết proc insert bảng giỏ hàng (1đ)

3.Viết proc insert bảng chi tiết giỏ hàng. nếu số lượng mua > số lượng của sản phẩm ->nhập lại số lượng. Nếu tạo chi tiết

giỏ hàng thành công thì update lại số lượng sản phẩm (3đ)

Mỗi proc được gọi 3 lần

4.Viết view hiển thị thông tin của sản phẩm có nhiều lượt mua nhất (1đ)

5.Viết hàm nhận vào mã sản phẩm trả ra số lượng sản phầm đã bán của sản phẩm đó (1đ)

6.Tạo view hiển thị thông tin tên sản phẩm, số lượng , giá bán, thành tiền của sản phẩm (1đ)

7.Tạo view lấy ra top 2 sản phẩm có số lượng nhiều nhất (1đ)

8.Viết trigger kiểm tra khi insert cho bảng giỏ hàng , nếu mã giỏ hàng tồn tại -> thông báo mã đã tồn tại (1đ)

**Đề 6:**

Tạo bảng loại sách : mã loại, mã sách, số lượng

Tạo bảng sách : mã sách, tên sách, tác giả

Tạo bảng phiếu mượn: mã phiếu, mã sách, tên sinh viên, trạng thái, ngày mượn, số lượng mượn

Viết proc insert bảng sách nếu mã sách đã tồn tại -> hiển thị sách đã tồn tại

Viết proc insert bảng kho sách

Viết proc insert bảng phiếu mượn, nếu mã sách không tồn tại -> thông báo sách không tồn tại, nếu số lượng mượn > số lượng sách trong kho

-> thông báo không đủ sách cho mượn, trường hợp insert thành công -> update lại số lượng sách trong kho = số lượng - số lượng mượn

Viết proc trả sách : update lại trạng thái của phiếu mượn, đồng thời update lại số lượng sách trong kho

Viết view hiển thị top 2 sách có lượt mượn nhiều nhất

Viết view hiển thị tên sinh viên và sách mượn

Viết hàm nhận vào mã sách -> hiển thị số lượt mượn của sách

Viết hàm nhận vào mã sách -> hiển thị sinh viên chưa trả sách

Đề 7:

Tạo bảng phòng ban: mã phòng ban, tên phòng ban, số lượng nhân viên

Tạo bảng nhân viên : mã nhân viên, tên nhân viên, mã phòng ban, lương, ngày sinh

Viết proc insert bảng nhân viên. kiểm tra mã nhân viên đã tồn tai -> thông báo mã đã tồn tại, ngược lại cho insert.

Khi insert bảng nhân viên đồng thời tăng số lượng nhân viên của phòng ban đó +1

Viết hàm nhận vào mã nhân viên trả ra lương và thuế của nhân viên đó . thuế = 10% \*lương

viết proc insert dữ liệu bảng phòng ban -> kiểm tra mã phòng ban đã tồn tại -> thông báo đã tồn tại

viết trigger check insert bảng nhân viên . Nếu lương <100 thông báo -> nhập lại lương

Tạo view hiển thị top 2 phòng ban có nhiều nhân viên nhất

Tạo view hiển thị top 3 nhân viên có mức lương cao nhất của phòng KT

Viết proc delete bảng nhân viên. Xoá nhân viên thì số lương nhân viên giảm 1.

**Đề 8:**

Tạo bảng lớp học : mã lớp, tên lớp

Tạo bảng hoc sinh có các trường. mã học sinh, tên học sinh, năm sinh, mã lớp , điểm

1. Viết SP với các tham số thích hợp insert cho bảng lớp học và sinh viên , mỗi sp phải kiểm tra đầu vào, check null các trường

Viết 3 lời gọi SP

2.Viết hàm nhận vào mã lớp -> lấy ra thông tin các nhân viên có mã lớp đó

3.Viết SP nhận vào số sinh viên. SP thực hiện xóa sinh viên của lớp đó nếu tổng số sinh viên của lớp < giá trị tham số được truyền vào

4. Viết trigger kiểm tra dữ liệu. khi insert sinh viên điểm phải nằm trong khoảng 0<=điểm<=10

5. Viết hàm nhận vào mã lớp, đếm số sinh viên của lớp đó