**Người thực hiện: Tôn Hoàng Cầm Mã SV: 23110186**

Em cam đoạn tất cả các câu dưới đây đều do em viết code và đều đã chạy ra kết quả

**Bài 1 :**

**--1 Hãy cho biết tên các dự án mà nhân viên có mã ‘NV01’ tham gia**

select distinct Duan.TenDa

from PhanCong PC

join Duan on PC.MaDa = Duan.MaDa where PC.MaDa = 'NV01';

**--2 Tính tổng thời gian tham gia các dự án của mỗi nhân viên**

select NV.MaNV, NV.HonV, NV.Tenlot, NV.TenNV, sum( PC.ThoiGian) as TongThoiGian

from NhanVien NV

left join PhanCong PC on NV.MaNV = PC.MaNV

group by NV.MaNV, NV.HonV, NV.Tenlot, NV.TenNV;

**--3 Cho biết họ tên các nhân viên chưa tham gia dự án nào**

select NV.HonV, NV.Tenlot, NV.TenNV

from NhanVien NV

left join PhanCong PC on NV.MaNV = PC.MaNV

where PC.MaNV is null;

**--a Tìm ngày sinh và địa chỉ của nhân viên “Nguyễn Bảo Hùng”**

select NgSinh, Dchi from NhanVien

where HonV = 'Nguyen' and Tenlot = 'Bao' and TenNV = 'Hung';

**--b Tìm tên và địa chỉ của các nhân viên làm việc cho phòng “Nghiên cứu”**

select NV.HonV, NV.Tenlot, NV.TenNV, NV.Dchi

from NhanVien NV

join PhongBan PB on NV.Phong = PB.MaPB

where PB.TenPB = 'Nghien cuu';

**--c Với mỗi dự án triển khai ở Gò Vấp, cho biết mã dự án, mã phòng quản lý và họ tên, ngày sinh trưởng phòng**

select Da.MaDa, Da.Phong, NV.HonV, NV.Tenlot, NV.TenNV, NV.NgSinh

from Duan Da

join PhongBan PB on Da.Phong = PB.MaPB

join NhanVien NV on PB.TrPhong = NV.MaNV

where Da.DiaDiem = 'Go Vap'

**--d Với mỗi nhân viên, cho biết họ tên nhân viên và họ tên của người quản lý nhân viên đó**

select NV.HonV, NV.Tenlot, NV.TenNV,

NQL.HonV as HonQL, NQL.Tenlot as TenlotNQL, NQL.TenNV as TenNQL

from NhanVien NV

left join NhanVien NQL on NV.MaNQL = NQL.MaNV;

**--e Cho biết mã nhân viên, họ và tên của các nhân viên của phòng “Nghiên cứu” có mức lương từ 30000 đến 50000**

select NV.MaNV, NV.HonV, NV.Tenlot, NV.TenNV

from NhanVien NV

join PhongBan PB on NV.Phong = PB.MaPB

where PB.TenPB = 'Nghien cuu' and NV.Luong between 30000 and 50000;

**--f Cho biết mã nhân viên, họ tên nhân viên và mã dự án, tên dự án của các dự án mà họ tham gia**

select NV.MaNV, NV.HonV, NV.Tenlot, NV.TenNV, Da.MaDa, Da.TenDa

from NhanVien NV

join PhanCong PC on NV.MaNV = PC.MaNV

join Duan Da on PC.MaDa = Da.MaDa;

**--g Cho biết mã nhân viên, họ tên của những người không có người quản lý**

select NV.MaNV, NV.HonV, NV.Tenlot, NV.TenNV

from NhanVien NV

where NV.MaNQL is null;

**--h Cho biết họ tên của các trưởng phòng có thân nhân**

select DISTINCT NV.HonV, NV.Tenlot, NV.TenNV

from NhanVien NV

join PhongBan PB on NV.MaNV = PB.TrPhong

join ThanNhan TN on NV.MaNV = TN.MaNV;

**--i Tính tổng lương nhân viên, lương cao nhất, lương thấp nhất và mức lương trung bình**

select sum(Luong) as TongLuong, max(Luong) as LuongCaonhat,

min(Luong) as LuongThapNhat, avg(Luong) as LuongTrungBinh

from NhanVien;

**--j Cho biết tổng số nhân viên và mức lương trung bình của phòng “Nghiên cứu”**

select count(NV.MaNV) as TongSonhanVien, avg(NV.Luong) as LuongTrungBinh

from NhanVien NV

join PhongBan PB on NV.Phong = PB.MaPB

where PB.TenPB = 'Nghien cuu';

**--k Với mỗi phòng, cho biết mã phòng, số lượng nhân viên và mức lương trung bình**

select PB.MaPB, count(NV.MaNV) as SoLuongNhanVien, avg(NV.Luong) as LuongTrungBinh

from PhongBan PB

LEFT join NhanVien NV on PB.MaPB = NV.Phong

group by PB.MaPB;

**--l Với mỗi dự án, cho biết mã dự án, tên dự án và tổng số nhân viên tham gia**

select Da.MaDa, Da.TenDa, count(PC.MaNV) as SoLuongNhanVien

from Duan Da

LEFT join PhanCong PC on Da.MaDa = PC.MaDa

group by Da.MaDa, Da.TenDa;

**--m Với mỗi dự án có nhiều hơn 2 nhân viên tham gia, cho biết mã dự án, tên dự án và số lượng nhân viên tham gia**

select Da.MaDa, Da.TenDa, count(PC.MaNV) as SoLuongNhanVien

from Duan Da

join PhanCong PC on Da.MaDa = PC.MaDa

group by Da.MaDa, Da.TenDa

having count(PC.MaNV) > 2;

**--n Với mỗi dự án, cho biết mã số dự án, tên dự án và số lượng nhân viên phòng số 5 tham gia**

select Da.MaDa, Da.TenDa, count(PC.MaNV) as SoLuongNhanVien

from Duan Da

join PhanCong PC on Da.MaDa = PC.MaDa

join NhanVien NV on PC.MaNV = NV.MaNV

where NV.Phong = 5

group by Da.MaDa, Da.TenDa;

**--o Với mỗi phòng có nhiều hơn 2 nhân viên, cho biết mã phòng và số lượng nhân viên có lương lớn hơn 25000**

select NV.Phong, count(NV.MaNV) as SoLuongNhanVien

from NhanVien NV

where NV.Luong > 25000

group by NV.Phong

having count(NV.MaNV) > 2;

**--p Với mỗi phòng có mức lương trung bình lớn hơn 30000, cho biết mã phòng, tên phòng, số lượng nhân viên của phòng đó**

select PB.MaPB, PB.TenPB, count(NV.MaNV) as SoLuongNhanVien

from PhongBan PB

join NhanVien NV on PB.MaPB = NV.Phong

group by PB.MaPB, PB.TenPB

having avg(NV.Luong) > 30000;

**--q Với mỗi phòng có mức lương trung bình lớn hơn 30000, cho biết mã phòng, tên phòng, số lượng nhân viên nam của phòng đó**

select PB.MaPB, PB.TenPB, count(NV.MaNV) as SoLuongNhanVienNam

from PhongBan PB

join NhanVien NV on PB.MaPB = NV.Phong

where NV.Gioitinh = 'Nam'

group by PB.MaPB, PB.TenPB

having avg(NV.Luong) > 30000;

**Bài 2 :**

**--a Địa chỉ và số điện thoại của Nhà xuất bản “Addison Wesley”**

select DiaChi, Sdt

from NXB

where TenNXB = 'Addison Wesley';

**--b Mã sách và tựa sách của những cuốn sách xuất bản bởi “Addison Wesley”**

select Masach, Tua

from DauSach

join NXB on DauSach.MaNXB = NXB.MaNXB

where NXB.TenNXB = 'Addison Wesley';

**--c Mã sách và tựa sách của những cuốn sách có tác giả là “Hemingway”**

select DauSach.Masach, Tua

from DauSach

join TacGia on DauSach.Masach = TacGia.Masach

where TacGia.TenTacGia = 'Hemingway';

**--d Với mỗi đầu sách, cho biết tựa và số lượng cuốn sách thư viện đang sở hữu**

select DauSach.Tua, count(CuonSach.MaCuon) as SoLuongCuon

from DauSach

left join CuonSach on DauSach.Masach = CuonSach.Masach

group by DauSach.Tua;

**--e Với mỗi độc giả, cho biết tên, địa chỉ và số lượng cuốn sách đã mượn**

select DocGia.TenDG, DocGia.DiaChi, count(Muon.MaCuon) as SoLuongMuon

from DocGia

left join Muon on DocGia.MaDG = Muon.MaDG

group by DocGia.TenDG, DocGia.DiaChi;

**--f Mã cuốn, tựa sách và vị trí của những cuốn sách do “Addison Wesley” xuất bản**

select CuonSach.MaCuon, DauSach.Tua, CuonSach.ViTri

from CuonSach

join DauSach on CuonSach.Masach = DauSach.Masach

join NXB on DauSach.MaNXB = NXB.MaNXB

where NXB.TenNXB = 'Addison Wesley';

**--g Với mỗi đầu sách, cho biết tên nhà xuất bản và số lượng tác giả**

select NXB.TenNXB, DauSach.Tua, count(TacGia.TenTacGia) as SoLuongTacGia

from DauSach

join NXB on DauSach.MaNXB = NXB.MaNXB

left join TacGia on DauSach.Masach = TacGia.Masach

group by NXB.TenNXB, DauSach.Tua;

**--h Thông tin độc giả đã mượn từ 5 cuốn sách trở lên**

select DocGia.TenDG, DocGia.DiaChi, DocGia.Sdt

from DocGia

join Muon on DocGia.MaDG = Muon.MaDG

group by DocGia.TenDG, DocGia.DiaChi, DocGia.Sdt

having count(Muon.MaCuon) >= 5;

**--i Mã NXB, tên NXB và số lượng đầu sách của NXB trong CSDL**

select NXB.MaNXB, NXB.TenNXB, count(DauSach.Masach) as SoLuongDauSach

from NXB

left join DauSach on NXB.MaNXB = DauSach.MaNXB

group by NXB.MaNXB, NXB.TenNXB;

**--j Mã NXB, tên NXB và địa chỉ của những NXB có từ 100 đầu sách trở lên**

select NXB.MaNXB, NXB.TenNXB, NXB.DiaChi

from NXB

join DauSach on NXB.MaNXB = DauSach.MaNXB

group by NXB.MaNXB, NXB.TenNXB, NXB.DiaChi

having count(DauSach.Masach) >= 100;

**--k Mã NXB, tên NXB và số lượng tác giả đã hợp tác với NXB đó**

select NXB.MaNXB, NXB.TenNXB, count(distinct TacGia.TenTacGia) as SoLuongTacGia

from NXB

join DauSach on NXB.MaNXB = DauSach.MaNXB

join TacGia on DauSach.Masach = TacGia.Masach

group by NXB.MaNXB, NXB.TenNXB;

**--l l. Tựa sách và số lượng tác giả của những cuốn sách có tác giả là “Hemingway” mà độc giả “Nguyễn Văn A” đã từng mượn**

select DauSach.Tua, count(distinct TacGia.TenTacGia) as SoLuongTacGia

from DauSach

join TacGia on DauSach.Masach = TacGia.Masach

join CuonSach on DauSach.Masach = CuonSach.Masach

join Muon on CuonSach.MaCuon = Muon.MaCuon

join DocGia on Muon.MaDG = DocGia.MaDG

where TacGia.TenTacGia = 'Hemingway' and DocGia.TenDG = 'Nguyễn Văn A'

group by DauSach.Tua;

**Bài 3 :**

**--a Tìm tên những nhân viên ở cơ quan có mã số là 50**

select ten

from nv

where mscoquan = 50;

**--b Tìm mã số tất cả các cơ quan từ quan hệ NV**

select distinct mscoquan

from nv;

**--c Tìm tên các nhân viên ở cơ quan có mã số là 15, 20, 25**

select ten

from nv

where mscoquan IN (15, 20, 25);

**--d Tìm tên những người làm việc ở Đồ Sơn**

select nv.ten

from nv

join coquan on nv.mscoquan = coquan.mscoquan

where coquan.diachi = 'Do Son';

**Bài 4 :**

**--1 Danh sách các người thợ không tham gia vào hợp đồng sửa chữa nào**

select tho.matho, tho.tentho

from tho

left join chitiet\_hd on tho.matho = chitiet\_hd.matho

where chitiet\_hd.matho is null;

**--2 Danh sách hợp đồng đã thanh lý nhưng chưa thanh toán đầy đủ**

select hopdong.sohd, khachhang.tenkh, hopdong.trigiahd,

coalesce(sum(phieuthu.sotienthu), 0) as tongdathanhtoan

from hopdong

left join phieuthu on hopdong.sohd = phieuthu.sohd

join khachhang on hopdong.makh = khachhang.makh

where hopdong.ngayngthu is not null -- đã thanh lý

group by hopdong.sohd, khachhang.tenkh, hopdong.trigiahd

having coalesce(sum(phieuthu.sotienthu), 0) < hopdong.trigiahd;

**--3 Danh sách hợp đồng cần hoàn tất trước ngày 31/12/2002**

select sohd, ngayhd, makh, soxe, trigiahd, ngaygiaodk

from hopdong

where ngaygiaodk < '2002-12-31' and ngayngthu is null;

**--4 Người thợ thực hiện nhiều công việc nhất**

select top 1 tho.matho, tho.tentho, count(chitiet\_hd.macv) as soluongcongviec

from tho

left join chitiet\_hd on tho.matho = chitiet\_hd.matho

group by tho.matho, tho.tentho

order by soluongcongviec desc;

**--5 Người thợ có tổng trị giá công việc cao nhất**

select top 1 tho.matho, tho.tentho, coalesce(sum(chitiet\_hd.trigiacv), 0) as tongtrigiacv

from tho

left join chitiet\_hd on tho.matho = chitiet\_hd.matho

group by tho.matho, tho.tentho

order by tongtrigiacv desc;

**Bài 5 :**

**--a Danh sách giáo viên dạy môn học có số tiết từ 45 trở lên**

select gv.magv, gv.tengv, mhoc.tenmh, mhoc.sotiet

from gv

join mhoc on gv.mamh = mhoc.mamh

where mhoc.sotiet >= 45;

**--b Danh sách giáo viên được phân công gác thi trong học kỳ 1**

select gv.magv, gv.tengv

from gv

join pc\_coi\_thi on gv.magv = pc\_coi\_thi.magv

where pc\_coi\_thi.hky = 1;

**--c Danh sách giáo viên không được phân công gác thi trong học kỳ 1**

select gv.magv, gv.tengv

from gv

left join pc\_coi\_thi on gv.magv = pc\_coi\_thi.magv and pc\_coi\_thi.hky = 1

where pc\_coi\_thi.magv is null;

**--d Cho biết lịch thi môn Văn (TENMH = ‘VĂN HỌC’)**

select buoithi.hky, buoithi.ngay, buoithi.gio, buoithi.phg, buoithi.tgthi

from buoithi

join mhoc on buoithi.mamh = mhoc.mamh

where mhoc.tenmh = 'van hoc';

**--e Buổi gác thi của giáo viên chủ nhiệm môn Văn**

select pc\_coi\_thi.magv, gv.tengv, pc\_coi\_thi.hky, pc\_coi\_thi.ngay, pc\_coi\_thi.gio, pc\_coi\_thi.phg

from pc\_coi\_thi

join gv on pc\_coi\_thi.magv = gv.magv

join mhoc on gv.mamh = mhoc.mamh

where mhoc.tenmh = 'van hoc';

Câu hỏi ôn tập chương 4

1. Ngôn ngữ SQL là gì?

là ngôn ngữ dùng để thao tác và quản lý dữ liệu trong hệ quản trị cơ sở dữ liệu quan hệ

1. Thứ tự viết các mệnh đề trong câu SQL?

* SELECT: Chọn các cột cần lấy dữ liệu.
* FROM: Chỉ định bảng hoặc các bảng cần lấy dữ liệu.
* JOIN: Kết hợp các bảng với nhau (nếu cần).
* WHERE: Điều kiện lọc các hàng dữ liệu.
* GROUP BY: Nhóm các hàng dữ liệu theo một hoặc nhiều cột.
* HAVING: Điều kiện lọc các nhóm dữ liệu (sau khi đã nhóm).
* ORDER BY: Sắp xếp các hàng dữ liệu theo một hoặc nhiều cột.
* LIMIT: Giới hạn số lượng hàng dữ liệu trả về

1. Thứ tự thực hiện các mệnh đề trong câu SQL?

* FROM: Mệnh đề FROM: Xác định bảng hoặc tập hợp dữ liệu từ nơi bạn muốn truy vấn dữ liệu. Trong trường hợp này, bạn đang truy vấn dữ liệu từ bảng Users u và Posts p.
* INNER JOIN: Mệnh đề INNER JOIN: Kết hợp các bảng Users và Posts dựa trên điều kiện u.Id = p.OwnerUserId. Điều này xảy ra sau khi mệnh đề FROM đã được thực hiện.
* WHERE: Mệnh đề WHERE: Áp dụng điều kiện lọc dữ liệu. Trong trường hợp này, chỉ lấy các bài đăng (Posts) có PostTypeId = 2.
* GROUP BY: Mệnh đề GROUP BY: Nhóm các dòng dữ liệu dựa trên cột location. Sau đó, bạn có thể áp dụng các hàm tổng hợp như COUNT(\*) để đếm số lượng dòng dữ liệu trong mỗi nhóm.
* HAVING: Mệnh đề HAVING: Áp dụng điều kiện lọc dữ liệu cho các nhóm đã được tạo bởi GROUP BY. Trong trường hợp này, chỉ lấy các nhóm có location chứa từ 'vietnam' và có số lượng dòng dữ liệu (COUNT(\*)) lớn hơn 10.
* SELECT: Mệnh đề SELECT: Chọn cột location và COUNT(\*) (được đặt tên là cnt) từ kết quả truy vấn. Đây là mệnh đề cuối cùng trong câu truy vấn SELECT.
* DISTINCT: Mệnh đề DISTINCT: Loại bỏ các giá trị trùng lặp từ kết quả truy vấn. Trong trường hợp này, chỉ lấy 10 bản ghi đầu tiên (top 10) từ kết quả sau khi áp dụng DISTINCT.
* ORDER BY: Mệnh đề ORDER BY: Sắp xếp kết quả theo cột cnt (số lượng dòng dữ liệu) theo thứ tự tăng dần.

1. Cho Q(A, B, C). Viết câu truy vấn tương đương với biểu thức đại số quan hệ πA,B(Q)

select a, b from q;

1. Cho Q(A, B, C, D). Dấu \* ở mệnh đề Select trong câu truy vấn SELECT \* FROM Q có nghĩa là gì?

\* đại diện cho tất cả các cột trong bảng Q.

1. Cho Q(A: int, B: int, C: int). Viết 2 câu truy vấn tương đương với biểu thức đại số quan hệ πA,B,C(σA>B(Q))

SELECT A, B, C FROM Q WHERE A > B;

1. Cho Q(A: int, B: int, C: int). Tìm các chỗ sai trong câu truy vấn

SELECT A.Q FROM Q WHERE A.Q > ‘5’

A.Q sai cú pháp, đúng phải là Q.A hoặc chỉ A.

'5' là kiểu chuỗi, nhưng A.Q có thể là số nguyên → cần A.Q > 5.

SELECT A FROM Q WHERE A > 5;

1. Cho SV(MaSV: string, Hoten: string, MaKh:string, DiemTB:real). Câu truy vấn nào dưới đây thực hiện lập danh sách SV thuộc Khoa có MaKh là ‘50’ và có điểm tb > 8:
2. SELECT MaSV, Hoten FROM SV WHERE MaKh = 50, DiemTB >8
3. SELECT MaSV, Hoten FROM SV WHERE MaKh = ‘50’, DiemTB >8
4. SELECT MaSV, Hoten FROM SV WHERE MaKh = ‘50’∧ DiemTB >8
5. SELECT MaSV, Hoten FROM SV WHERE MaKh = ‘50’ and DiemTB >8

SELECT MaSV, Hoten FROM SV WHERE MaKh = '50' AND DiemTB > 8;

1. Cho SV(MaSV: string, Hoten: string, MaKh:string, DiemTB:real). Câu truy vấn nào dưới đây sai và tại sao?
2. SELECT MaSV, Hoten FROM SV WHERE MaKh = 50, DiemTB >8
3. SELECT MaSV.SV, Hoten.SV FROM SV WHERE DiemTB >8
4. SELECT SV.MaSV, SV.Hoten FROM SV WHERE DiemTB >8

Sai : SELECT MaSV, Hoten FROM SV WHERE MaKh = 50, DiemTB > 8; Lỗi: Dùng dấu phẩy , thay vì AND.

Sai : SELECT MaSV.SV, Hoten.SV FROM SV WHERE DiemTB > 8; Lỗi: Cấu trúc MaSV.SV sai.

1. Cho NV(MaNV: string, Hoten: string, Luong: Real, MaPB:string). Câu SQL nào dưới đây đúng:
2. SELECT MaNV, Hoten, Luong \* 2 From NV WHERE MAPB = ‘50’
3. SELECT MaNV, Hoten, Luong \* 2 Thuong From NV WHERE MAPB = ‘50’
4. SELECT MaNV, Hoten, Luong \* 2 as Thuong Form NV WHERE MAPB = ‘50’

Câu a đúng

1. Cho NV(MaNV: string, Hoten: string, Luong: Real, MaPB:string). Hãy cho biết kết quả của câu truy vấn: SELECT MaNV, Hoten, Luong \* 2 as Thuong FROM NV

Tạo thêm cột Thuong (lương nhân đôi) trong kết quả.

1. Từ khóa AS trong câu truy vấn SELECT MaNV, Hoten, Luong \* 2 as Thuong FROM NV dùng để làm gì?

AS tạo bí danh cho cột hoặc bảng

1. Cho NhanVien(MaNV: string, Hoten: string, Luong: Real). câu truy vấn dưới thực hiện việc gì:

SELECT NV.MaNV, NV.Hoten, NV.Luong FROM NhanVien as NV WHERE Luong >1000

Giúp đặt tên tắt NV cho bảng NhanVien

1. Từ khóa AS trong câu truy vấn SELECT MaNV, Hoten, Luong FROM NhanVien as NV dùng để làm gì?

As dùng đặt bí danh bảng

1. bí danh của thuộc tính được đặt ở mệnh đề SELECT có được dùng ở các mệnh đề khác không? Tại sao?

* Không được phép.
* Vì: SQL thực thi SELECT sau WHERE.

1. Phát biểu nào dưới đây đúng:
2. Các biểu thức điều kiện ở mệnh đề WHERE phải được liên kết bằng AND hoặc OR
3. Các biểu thức điều kiện ở mệnh đề WHERE phải được phân cách bằng dấu phẩy

Câu a đúng

1. LIKE hoặc NOT LIKE dùng để làm gì?

Tìm kiếm chuỗi có dạng nhất định.

1. Ý nghĩa của ký tự đại diện ‘%’ hoặc ‘\_’ trong biểu thức điều kiện so sánh chuỗi.

% : Bất kỳ số ký tự nào

\_ : Một ký tự bất kỳ

1. Cho Q(A: nvarchar(4), B: int). câu SQL SELECT \* FROM Q WHERE A like ‘%a% thực  
    hiện việc gì?

Tìm các giá trị A có chứa chữ "a" ở bất kỳ vị trí nào

1. Cho Q(A: varchar(4), B: int). Câu SQL SELECT \* FROM Q WHERE A like ‘\_a% thực  
    hiện việc gì?

Tìm giá trị A có chữ "a" ở vị trí thứ hai.

1. Cho Q(A: varchar(4), B: int). Viết câu SQL tương đương với câu   
    SELECT \* FROM Q WHERE B Between 10 and 20

SELECT \* FROM Q WHERE B >= 10 AND B <= 20;

1. Hàm gộp là gì và gồm những hàm nào?

* COUNT()
* SUM()
* AVG()
* MAX()
* MIN()

1. Cho NV(MaNV: varchar(10), Luong: int, MaPB: varchar(4)).   
    Câu SQL SELECT count(\*), avg(luong) FROM NV thực hiện việc gì?

Đếm số nhân viên

1. Cho NV(MaNV: varchar(10), Luong: int, MaPB: varchar(4)).   
    Trong SQL Server, kết quả của câu SQL: SELECT count(\*), avg(luong) FROM NV là gì?

Tính trung bình lương.

1. Cho NV(MaNV: varchar(10), Luong: int, MaPB: varchar(4)).   
    Trong SQL Server, kết quả của câu SQL: SELECT count(\*) Sonv, avg(luong) as Luongtb FROM NV là gì?

Sonv: Tổng số nhân viên trong bảng NV

Luongtb: Trung bình lương của nhân viên

1. Cho NV(MaNV: varchar(10), Luong: int, MaPB: varchar(4)).   
    Câu SQL SELECT count(MaNV), avg(luong) FROM NV GROUP BY MaPB thực hiện việc gì?

Tính số lượng nhân viên (COUNT(MaNV)) và trung bình lương (AVG(luong)) theo từng MaPB (mã phòng ban).

1. Cho NV(MaNV: varchar(10), Luong: int, MaPB: varchar(4)).   
    Trong SQL Server, kết quả của câu SQL: SELECT MaPB, count(MaNV), avg(luong) FROM NV GROUP BY MaPB là gì?

Hiển thị mã phòng (MaPB), số lượng nhân viên (COUNT(MaNV)), và trung bình lương (AVG(luong)) của từng phòng.

1. Phát biểu nào dưới đây đúng khi câu SQL có sử dụng mệnh đề GROUP BY
2. Các thuộc tính có trong mệnh đề SELECT phải có trong mệnh đề GROUP BY hoặc trong hàm gộp
3. Thuộc tính có trong mệnh đề GROUP BY có thể không có trong mệnh đề SELECT

Câu a đúng

1. Phát biểu nào dưới đây đúng:
2. Trong câu SQL, mệnh đề HAVING được sử dụng khi điều kiện chọn liên quan tới hàm gộp
3. Trong câu SQL, mệnh đề HAVING (nếu có) phải nằm sau mệnh đề GROUP BY

Cả hai câu a và b đúng

1. Phát biểu nào dưới đây đúng:
2. Trong câu SQL, khi sử dụng mệnh đề HAVING thì phải có mệnh đề GROUP BY
3. Trong câu SQL, mệnh đề WHERE sử dụng với điều kiện không dùng hàm gộp
4. Trong câu SQL, có thể có cả mệnh đề WHERE và mệnh đề HAVING

Cả ba câu đều đúng

1. Cho NV(MaNV: varchar(10), Luong: int, MaPB: varchar(4)).   
   câu SQL: SELECT MaPB, count(MaNV) FROM NV GROUP BY MaPB HAVING count(MaNV) > 2; thực hiện việc gì?

Lấy danh sách phòng ban có hơn 2 nhân viên.

1. Cho Q(A: varchar(10), B: int, C: varchar(4)). Chỉ những chỗ sai của câu SQL:   
    SELECT A, count(b) as v FROM Q.A GROUP BY C HAVING v > 2;

Sai ở:

* SELECT A, COUNT(b) AS v FROM Q.A GROUP BY C HAVING v > 2;
* Q.A không đúng cú pháp.
* Cột A không có trong GROUP BY

1. Cho NV(MaNV: varchar(10), Luong: int, MaPB: varchar(4)). Chỉ những chỗ sai của câu SQL: SELECT MaPB, count(MaNV) as sonv FROM NV n WHERE sonv > 2;

sonv không thể dùng trong WHERE

1. Mệnh đề ORDER BY dùng để làm gì?

Dùng để sắp xếp kết quả theo thứ tự tăng (ASC) hoặc giảm (DESC).

1. Cho NV (MaNV, Hoten, NgSinh). Câu SQL:   
    SELECT Hoten, NgSinh FROM NV ORDER BY Hoten, NgSinh DESC; thực hiện việc gì?

Sắp xếp theo Hoten tăng dần

Nếu trùng Hoten, sắp xếp NgSinh giảm dần

1. Cho Q(A, B, C). Viết câu SQL cho kết quả tương đương với câu   
    SELECT \* FROM Q ORDER BY A DESC, C

SELECT A, B, C FROM Q ORDER BY A DESC, C ASC;

1. Trình bày phép kết nội và phép kết ngoài

Kết nội (INNER JOIN): Chỉ lấy bản ghi có dữ liệu khớp.

Kết ngoài (OUTER JOIN): Lấy cả dữ liệu không khớp

1. Cho Q1(A: int, B: int) và Q2(C: int, D: int) và câu SQL   
   SELECT A, C FROM Q1 INNER JOIN Q2 ON A = C; thực hiện việc gì

Lấy cột A từ Q1 và C từ Q2 khi giá trị A = C

1. Cho Q1(A: int, B: int) và Q2(C: int, D: int). Câu SQL nào dưới đây đúng
2. SELECT A, C FROM Q1 JOIN Q2 ON A = C;
3. SELECT A, C FROM Q1 INNER JOIN Q2 ON A = C;

Cả hai câu đúng

1. Cho Q1(A: int, B: int) và Q2(C: int, D: int). Câu SQL nào dưới đây đúng
2. SELECT A, C FROM Q1 LEFT JOIN Q2 ON A = C;
3. SELECT A, C FROM Q1 LEFT OUTER JOIN Q2 ON A = C;

Cả hai câu đúng

1. Cho Q1(A: int, B: int) và Q2(C: int, D: int). Câu SQL   
   SELECT \* FROM Q1 FULL JOIN Q2 ON A = C; thực hiện việc gì?

Lấy tất cả bản ghi từ Q1 và Q2, bao gồm giá trị không khớp.

1. Cho Q1(A: int, B: int) và Q2(C: int, D: int). Câu SQL   
   SELECT \* FROM Q1 FULL OUTER JOIN Q2 ON A = C; tương đương với
2. SELECT \* FROM Q1 JOIN Q2 ON A = C;
3. SELECT \* FROM Q1 FULL JOIN Q2 ON A = C

Câu b đúng

1. Cho Q1(A: int, B: int) và Q2(C: int, D: char(5)). Câu SQL   
   SELECT \* FROM Q1 OUTER JOIN Q2 A = C WHERE D = null; sai ở những chỗ nào?

OUTER JOIN thiếu LEFT hoặc RIGHT.

WHERE D = NULL sai cú pháp (IS NULL mới đúng)

1. Cho SV(MaSV, Hoten) và Dangky(MaSV, MaMH). Câu SQL  
   (SELECT MaSV FROM SV) EXCEPT (SELECT MaSV FROM Dangky); thực hiện việc gì?

Lấy sinh viên chưa đăng ký môn học.

1. Cho SV(MaSV, Hoten) và Dangky(MaSV, MaMH). Câu SQL nào ở dưới tương đương với câu truy vấn (SELECT MaSV FROM SV) EXCEPT (SELECT MaSV FROM Dangky)
2. SELECT SV.MaSV FROM SV LEFT JOIN Dangky ON SV.MaSV = Dangky.MaSV
3. SELECT SV.MaSV FROM SV LEFT JOIN Dangky ON SV.MaSV = Dangky.MaSV WHERE Dangky.MaMH IS NULL
4. SELECT SV.MaSV FROM SV LEFT JOIN Dangky ON SV.MaSV = Dangky.MaSV WHERE Dangky.MaMH = NULL

Câu b đúng

1. Cho SV(MaSV, Hoten) và Dangky(MaSV, MaMH). Câu SQL nào ở dưới tương đương với câu truy vấn (SELECT MaSV FROM SV) EXCEPT (SELECT MaSV FROM Dangky)
2. SELECT SV.MaSV FROM SV LEFT JOIN Dangky ON SV.MaSV = Dangky.MaSV WHERE Dangky.MaMH IS NULL
3. SELECT s.MaSV FROM SV s LEFT JOIN Dangky d ON s.MaSV = d.MaSV WHERE Dangky.MaMH IS NULL

Cả hai câu đúng

1. Cho SV(MaSV, Hoten) và Dangky(MaSV, MaMH). Chỉ những chỗ sai trong câu SQL  
   SELECT MaSV FROM SV OUTER JOIN Dangky ON SV.MaSV = Dangky.MaSV WHERE Dangky.MaMH = NULL;

Sai : OUTER JOIN thiếu LEFT hoặc RIGHT

WHERE Dangky.MaMH = NULL sai cú pháp

1. Cho SV(MaSV, Hoten) và Dangky(MaSV, MaMH). Câu SQL nào ở dưới cho kết quả tương đương với câu truy vấn (SELECT MaSV FROM SV) INTERSECT (SELECT MaSV FROM Dangky)
2. SELECT MaSV FROM Dangky
3. (SELECT MaSV FROM SV) EXCEPT (SELECT MaSV FROM Dangky)
4. SELECT MaSV FROM SV

Câu a đúng

1. Cho Q1(A, B) và Q2(A, B). Câu SQL  
   (SELECT A, B FROM Q1) INTERSECT (SELECT A, B FROM Q2); thực hiện việc gì?

Lấy các bản ghi có mặt trong cả Q1 và Q2.

1. Cho SV(MaSV, Hoten) và Dangky(MaSV, MaMH). Câu SQL dưới thực hiện việc gì (SELECT MaSV FROM SV) UNION (SELECT MaSV FROM Dangky)

Lấy danh sách tất cả mã sinh viên, kể cả sinh viên chưa đăng ký.

1. Cho Q1(A, B) và Q2(A, B). Câu SQL  
   (SELECT \* FROM Q1) UNION (SELECT \* FROM Q2); thực hiện việc gì?

Gộp hai bảng Q1 và Q2, không có bản ghi trùng lặp.

1. Cho SV(MaSV, Hoten) và Dangky(MaSV, MaMH). Câu truy vấn  
    SELECT Hoten FROM SV WHERE MaSV NOT IN (SELECT MaSV FROM Dangky); thực hiện việc gì?

Lấy tên sinh viên chưa đăng ký môn học.

1. Cho SV(MaSV, Hoten) và Dangky(MaSV, MaMH). Câu SQL tương đương với câu:  
    SELECT MaSV FROM SV WHERE MaSV NOT IN (SELECT MaSV FROM Dangky)

SELECT MaSV FROM SV LEFT JOIN Dangky ON SV.MaSV = Dangky.MaSV

WHERE Dangky.MaMH IS NULL;

1. Cho NV(MaNV, Hoten, Luong). Câu SQL   
    SELECT \* FROM NV WHERE Luong > (SELECT AVG(Luong) FROM NV); thực hiện việc gì?
2. Liệt kê danh sách NV có lương cao nhất
3. Liệt kê danh sách NV có lương lớn hơn mức lương trung bình

Câu b đúng

1. Cho NV(MaNV, Hoten, Luong). Câu SQL nào dưới đây tương đương với  
   SELECT \* FROM NV WHERE Luong > (SELECT AVG(Luong) FROM NV);
2. SELECT \* FROM NV, (SELECT AVG(Luong) as ltb FROM NV) Q WHERE Luong > LTB
3. SELECT \* FROM NV HAVING Luong > (SELECT AVG(Luong) FROM NV)

Câu a đúng.

1. Cho NV(MaNV, Hoten, Luong). Câu SQL   
    SELECT \* FROM NV WHERE Luong >= all (SELECT Luong FROM NV); thực hiện việc gì?

Lấy nhân viên có mức lương cao nhất

1. Cho NV(MaNV, Hoten, Luong). Câu SQL   
   SELECT \* FROM NV WHERE Luong = (SELECT max(Luong) FROM NV); thực hiện việc gì?

Lấy nhân viên có lương cao nhất.

1. Cho NV(MaNV, Hoten, Luong). Câu SQL nào dưới đây tương đương với  
   SELECT \* FROM NV WHERE Luong >= all (SELECT Luong FROM NV)
2. SELECT \* FROM NV WHERE Luong >= any (SELECT Luong FROM NV);
3. SELECT \* FROM NV WHERE Luong = (SELECT max(Luong) FROM NV)

Câu b đúng

1. Truy vấn con tương quan là gì?

* Là truy vấn con phụ thuộc vào từng dòng của truy vấn cha.
* Truy vấn con sẽ chạy lại từng dòng một của bảng bên ngoài

1. Công dụng của EXISTS

* Kiểm tra sự tồn tại của bản ghi trong một truy vấn con.
* Nếu truy vấn con có kết quả, EXISTS trả về TRUE, ngược lại FALSE.

1. Cho NV(MaNV, Hoten) và ThanNhan(MaNV, TenThannhan). Câu SQL nào dưới đây đúng
2. SELECT Hoten FROM NV WHERE EXISTS (SELECT \* FROM THANNHAN tn WHERE NV.MaNV = tn.MaNV);
3. SELECT Hoten FROM NV WHERE MaNV EXISTS (SELECT \* FROM THANNHAN tn WHERE NV.MaNV = tn.MaNV);

Câu a đúng