Câu hỏi ôn tập chương 4

1. Ngôn ngữ SQL là gì?

SQL (Structure Query Language) là ngôn ngữ truy vấn có cấu trúc.

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

select 🡪 from 🡪 where 🡪 group by 🡪 having 🡪 orrder by

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

from 🡪 where 🡪 group by 🡪 having 🡪 select 🡪 order by 🡪 kết quả truy vấn

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

***hay ?***

select distinct 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?

Dấu \* ở mệnh đề select có tác dụng lấy tất cả các cột mà không cần ghi cụ thể tên cột trong quan hệ là gì.

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

***hay ?***

select distinct 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’

Chỗ sai: A.Q

Sửa lại: Q.A

Giải thích: do sai cú pháp

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

- Câu a sai do ràng buộc về miền giá trị của MaKh là string, 50 là kiểu dữ liệu int, và giữa hai thuộc tính là toán tự logic **and** không phải dấu ,

1. SELECT MaSV.SV, Hoten.SV FROM SV WHERE DiemTB >8

Sai cú pháp do phải là <ten\_quan\_he>.<ten\_thuoc\_tinh> mới đúng, phải là SV.MaSV mới đúng

1. SELECT SV.MaSV, SV.Hoten FROM SV WHERE DiemTB >8

Đúng

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’
5. 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

Kết quả của câu truy vấn là 1 bảng mới có 3 cột: cột 1 là MaNV cột 2 là Hoten cột 3 là cột Thuong với giá trị của cột là giá trị của cột Luong trong quan hệ NV tương ứng \* 2

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

Từ khóa as trong câu truy vấn để tạo 1 bí danh cho cột có giá trị là giá trị từ cột Luong\*2, bí danh này chỉ tồn tại trong truy vấn đó

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

Chọn các cột MaNV, Hoten, Luong từ bảng nhân viên với điều kiện Luong nhân viên đó lớn hơn 1000

(Khi so sánh nvarchar phải có chữ N trước)

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

Từ khóa as trong câu truy vấn trên là đặt bí danh cho quan hệ NhanVien thành NV

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

Không vì theo thứ tự thực hiện select ở phía sau from, where, group by và having nên không thể sử dụng các bí danh đặt ở select cho các mệnh đề khác được

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
4. LIKE hoặc NOT LIKE dùng để làm gì?

- Dùng để so sánh chuỗi.

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

- Các ký tự đại diện: ‘%’ (đại diện một nhóm ký tự bất kỳ), ‘\_’ (đại diện 1 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 tất cà các bộ có giá trị a ở vị trí bất kì trong cột A

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

- Tìm tất cả các bộ có giá trị ‘\_a….’ tức là trước a là 1 kí tự bất kì sau có có thể có có thể không có bất kì chuỗi kỉ tự nào cột A

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 between 10 and 20

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

Hàm gộp là các hàm lấy vảo 1 tập giá trị và trả về 1 giá trị: min, max, sum, avg, count

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

Câu lệnh SQL trên thực hiện đếm số lượng bộ có trong quan hệ NV và tính lương trung bình của toàn bộ NV

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?

Là 1 bảng gồm 2 cột không có tên: cột 1 chứa tổng số lượng các bộ có trong NV, cột 2 tính lương trung bình của toàn bộ NV

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?

Là 1 bảng gồm 2 cột có tên: cột 1 tên Sonv chứa tổng số lượng các bộ có trong NV, cột 2 tên Luongtb tính lương trung bình của toàn bộ NV

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?

Kết quả trả về 1 quan hệ mới đếm số mã nhân viên và tính lương trung bình dựa trên MaPB cụ thể: với cột 1 là số lương nhân viên của mỗi phòng, cột 2 là tính lương trung bình của phòng ban đó

A screenshot of a computer

Description automatically generated

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?

Kết quả trả về 1 quan hệ mới đếm số mã nhân viên và tính lương trung bình dựa trên MaPB cụ thể: cột 1 là cột gom nhóm gồm các mã phòng ban (khác nhau), với cột 2 là số lương nhân viên của từng phòng ban, cột 3 là tính lương trung bình của phòng ban đó

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

Thứ tự mệnh đề: select 🡪 from 🡪 where 🡪 group by 🡪 having 🡪 order by

(SFWGHO - Smart Friends Will Gather Here Often)

Thứ tự thực hiện mệnh đề: from 🡪 where 🡪 group by 🡪 having 🡪 select 🡪 order by 🡪 kết quả

(FWGHSO - Friends Will Gather Here Seeking Onion)

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

Kiễm tra chưa chắc

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?

Thực hiện đếm số nhân viên của từng phòng ban, và trả về 1 quan hệ có 2 cột: cột 1 là MaPB, cột 2 là cột số lượng với điều kiện có số lượng nhân viên lớn hơn 2

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;

Những chỗ sai :

FROM Q.A

HAVING v > 2

GROUP BY C

Lí do: sau FROM phải là tên của 1 quan hệ không phải là tên của 1 thuộc tính

Having thì câu điều kiện phải liên quan đến hàm gộp và do select có thứ tự thực hiện sau having nên v không hợp lệ

Vì thuộc tính A có trong select nhưng không ở trong hàm gộp thì phải trong group by

- Có thể sửa như vầy:

select a, count(b) as v

from q

group by a, c

having count(b) > 0

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;

Sai ở select MaPB do không có trong hàm gộp thì phải có trong mệnh đề group by nhưng không có trong cả hai, va where thuc hien truoc select nen sonv khong xac dinh

Nhung loi sai:

WHERE sonv > 2

SELECT MaPB, count(MaNV)

Co the sua:

select mapb, count(manv) as sonv from NV n

group by mapb having count(manv) > 2

1. Mệnh đề ORDER BY dùng để làm gì?

Để sắp xếp thuộc tính theo thứ tự tăng dần hoặc giảm dần

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

Thực hiện lấy ra cột Hoten và NgSinh trong quan hệ NV với họ tên được sắp xếp tăng dần, nếu họ tên trùng thì ngày sinh giảm dần

A screenshot of a computer

Description automatically generated

1. Cho Q(A, B, C). Viết câu SQL cho kết quả tương đương với câu   
    SELECT \* FROM Q ORDER BY A DESC, C
2. Trình bày phép kết nội và phép kết ngoài

***Phép kết nội:***

Select <ten\_cot\_1>, <ten\_cot\_2>…..

From <quan\_he\_1>, <quan\_he\_2>

Where <dieu\_kien\_ket>

Select <ten\_cot\_1>, <ten\_cot\_2>…..

From <quan\_he\_1> inner join <quan\_he\_2> on <dieu\_kien\_ket>

***Phép kết ngoại:***

Select <ten\_cot\_1>, <ten\_cot\_2>,….

From <quan\_he\_1> left/right/full outer join <quan\_he\_2> on <dieu\_kien\_ket>

(tat ca cac join dieu giu nguyen so cot bang (cot cua qh1 + cot cua qh2)

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

Kết nội bảng Q1 và Q2 những cột thỏa điều kiện Q1.A = Q2.C, giá trị trả về là 1 bảng gồm 2 cột A và C và thỏa mãn điều kiện cho trước

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;**
4. Cho Q1(A: int, B: int) và Q2(C: int, D: int). Câu SQL nào dưới đây đúng
5. **SELECT A, C FROM Q1 LEFT JOIN Q2 ON A = C;**
6. **SELECT A, C FROM Q1 LEFT OUTER JOIN Q2 ON A = C;**
7. 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?

Thực hiện kết toàn phần 2 bảng: kết quả trả về gồm 4 cột NEW(A,B,C,B) với giá trị của bảng là kết quả kết nếu giá trị cột A bảng Q1 = cột C bảng Q2, và các hàng còn lại ở bảng Q1 không thỏa điều kiện sẽ set giá trị cột C và D là NULL, và các cột ở Q2 không thỏa điểu kiện sẽ set giá trị cột A và B là NULL

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**
4. 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?

Câu lệnh sai: thứ nhất là từ khóa outer join (thiếu left, right hay full) thứ hai trước điều kiện A = C thiếu từ khóa ‘on’

Đúng là: SELECT \* FROM Q1 **LEFT/RIGHT/FULL** OUTER JOIN Q2 **ON** A = C WHERE D = null

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

🡪 Thực hiện việc chọn ra những sinh viên chưa đăng kí 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 truy vấn (SELECT MaSV FROM SV) EXCEPT (SELECT MaSV FROM Dangky):

Phân tích từng cụm nhỏ trong cậu truy vấn trên: (SELECT MaSV FROM SV) tức là chiếu tất cả MaSV riêng biệt từ bảng SV, (SELECT MaSV FROM Dangky) tức là chiếu tất cả MaSV riêng biệt từ bảng Dangky, EXCEPT tức là ngoại trừ là phép Trừ trong các phép toán đại số quan hệ 🡪 lấy những cột có trong bảng bên trái nhưng không có trong bảng bên phải

Vậy kết quả sẽ trả về 1 bảng gồm 1 cột MaSv có trong SV nhưng không có trong Dangki, nôm na là những sinh viên không đăng kí

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
4. 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;

Những chỗ sai trong câu truy vấn: OUTER JOIN thiếu lựa chọn LEFT, RIGHT, FULL

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) intersect là phép giao
2. **SELECT MaSV FROM Dangky**
3. (SELECT MaSV FROM SV) EXCEPT (SELECT MaSV FROM Dangky)
4. SELECT MaSV FROM SV
5. 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?

Thực hiện chọn các bộ chung giữa 2 bảng 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)

Trả về những cột mà là hợp những MaSV có ở SV và cũng có ở Dangky, tương đương với trả về cột MaSV trong SV

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

Trả về những cột mà là hợp những MaSV có ở SV và cũng có ở Dangky, tương đương với trả về cột MaSV trong SV

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?

Thực hiện trả về tên những sinh viên không có trong bảng Dangky

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)

Thực hiện trả về cột MaSV có trong bảng SV nhưng không có trong bảng Dangky

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**
4. Cho NV(MaNV, Hoten, Luong). Câu SQL nào dưới đây tương đương với  
   SELECT \* FROM NV WHERE Luong > (SELECT AVG(Luong) FROM NV);
5. **SELECT \* FROM NV, (SELECT AVG(Luong) as ltb FROM NV) Q WHERE Luong > LTB**
6. SELECT \* FROM NV HAVING Luong > (SELECT AVG(Luong) FROM NV)
7. Cho NV(MaNV, Hoten, Luong). Câu SQL   
    SELECT \* FROM NV WHERE Luong >= all (SELECT Luong FROM NV); thực hiện việc gì?

Thực hiện trả về nhân viên trong bảng NV có Luong cao nhất

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

Thực hiện trả về nhân viên trong bảng NV có Luong 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)
4. Truy vấn con tương quan là gì?

Cho phép so sánh giữa thuộc tính của quan hệ bên trong truy vấn con với thuộc tính của quan hệ trong truy vấn bao bên ngoài

1. Công dụng của EXISTS

EXISTS được dùng để kiểm tra kết quả của câu truy vấn con tương quan là rỗng hay không rỗng. Nếu kết quả của câu truy vấn là rỗng thì EXISTS trả về False, ngược lại nó trả về True.

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);

create table dbo.NV(

MaNV nchar(10) primary key,

Hoten nvarchar(50) not null,

Luong real not null,

MaPB nchar(10) not null

)

create table dbo.Q(

A nvarchar(4) not null,

B int not null

)

======================================

SELECT TOP (1000) [A]

,[B]

FROM [chuong4].[dbo].[Q]

SELECT \*

FROM Q

WHERE A like N'a%'

SELECT \*

FROM Q

WHERE A like N'\_a%'

create table SV (

MaSV nvarchar(20),

Hoten nvarchar(50)

)

create table Dangky (

MaSV nvarchar(20),

MaMH nvarchar(50)

)

\* Chú ý câu 28 (hay)