

**KHOA CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG
ĐẠI HỌC CẦN THƠ**

Giáo trình:

**THỰC HÀNH
HỆ CƠ SỞ DỮ LIỆU**

**Biên soạn: Trần Ngân Bình – Bùi Đăng Hà Phương
Tháng 9/2012**

PHẦN 1. TỔNG QUAN

I. MỤC ĐÍCH YÊU CẦU

Giáo trình Thực hành Hệ Cơ Sở Dữ Liệu (CSDL) là một phần của môn Hệ CSDL nhằm giúp sinh viên nắm bắt được nội dung lý thuyết về ngôn ngữ SQL, và thực hiện một cách thành thạo các câu lệnh truy vấn trên cơ sở dữ liệu.

Các hệ quản trị CSDL được sử dụng phổ biến hiện nay đều hỗ trợ ngôn ngữ SQL theo chuẩn ít nhất là SQL-92. Ngoài những tính năng mà chuẩn đã nêu, mỗi phần mềm hệ quản trị CSDL có thể có những tính năng riêng thêm vào.

Phần mềm được chọn để thực hành là hệ quản trị CSDL SQL Server 2005. Trong 30 tiết (6 buổi) thực hành, sinh viên sẽ được rèn luyện từng dạng câu hỏi truy vấn thông qua các bài tập cụ thể trên các cơ sở dữ liệu cụ thể.

Sau khi học xong môn này, sinh viên phải:

- Nhận biết được sự thông thương giữa các bảng trong một cơ sở dữ liệu. Để từ đó biết cách kết nối các bảng để truy xuất dữ liệu theo yêu cầu.
- Hiểu được câu hỏi truy vấn để từ đó có thể chọn lựa dạng câu hỏi truy vấn phù hợp.
- Vận dụng được các dạng câu truy vấn khác nhau để trả lời câu hỏi như: câu truy vấn lồng nhau, câu truy vấn cần gom nhóm các dòng, câu truy vấn có sử dụng các hàm hàm xử lý chuỗi, hàm xử lý ngày tháng, hàm chuyển kiểu, ...

II. NỘI DUNG CỐT LÕI

Tài liệu hướng dẫn gồm 3 phần:

- *Phần 1:* giới thiệu tổng quan về môn học, trong đó trình bày cụ thể các kiến thức, kỹ năng mà sinh viên cần đạt được sau khi học môn này. Ngoài ra, còn trình bày sơ lược về nội dung của tài liệu, kiến thức tiên quyết và phương pháp học tập.
- *Phần 2:* giới thiệu các nét cơ bản về hệ quản trị SQL Server 2005. Ngoài ra, trong phần này, giáo trình cũng trình bày một cách chi tiết các thao tác cần tiến hành để có thể thực hành.
- *Phần 3:* gồm 5 bài thực hành cụ thể. Mỗi bài sẽ có bố cục gồm 4 phần như sau:
 1. Mục tiêu: nêu rõ mục tiêu rèn luyện của bài.
 2. Lý thuyết: Nhắc lại một cách ngắn gọn câu lệnh SELECT cần dùng trong bài.
 3. Bài tập có hướng dẫn: Gồm một số câu hỏi truy vấn trên một CSDL cho trước. Mỗi câu hỏi sẽ được phân tích để đi đến câu lệnh Select. Những điểm cần lưu ý sẽ được nhấn mạnh. Ngoài ra, một số câu hỏi nhỏ sẽ được đưa ra trong tình huống của câu hỏi giúp sinh viên hiểu rõ hơn vấn đề.
 4. Bài tập tự làm: gồm một số câu hỏi truy vấn trên một CSDL khác với phần bài tập có hướng dẫn.

Trong 4 bài đầu tiên, mỗi bài sẽ tập trung rèn luyện một dạng câu hỏi truy vấn. Bài thực hành thứ 5 là bài tập tổng hợp gồm tất cả các dạng câu hỏi đã học ở các bài trước, vì vậy sinh viên sẽ tự thực hiện mà không có phần hướng dẫn. Mỗi buổi thực hành sẽ tương ứng với một bài trong tài liệu hướng dẫn này.

III. KIẾN THỨC TIỀN QUYẾT

Là một phần của môn học Hệ CSDL, phần thực hành này được đưa vào giảng dạy cho sinh viên với yêu cầu sinh viên đã học xong chương SQL của môn Hệ CSDL.

IV. PHƯƠNG PHÁP HỌC TẬP

Với mục tiêu nâng cao khả năng tự học tập và tự nghiên cứu của sinh viên, người soạn đã cố gắng lồng ghép vào nội dung các ví dụ minh họa đơn giản, cụ thể; đồng thời bố trí bố cục với mong muốn tạo sự dễ hiểu cho sinh viên và người đọc.

Để học tốt môn học này, sinh viên cần phải:

- ✓ Trước mỗi buổi thực hành, sinh viên cần dành khoảng 3 giờ tự học để:
 - Xem lại phần lý thuyết của câu lệnh sẽ thực hành (phần 1 và 2 của mỗi bài + tài liệu tham khảo nếu cần thiết)
 - Tự làm bài tập có hướng dẫn tại nhà. Sinh viên nên tự suy nghĩ tìm ra đáp án cho các câu hỏi bài tập trước khi nhìn vào đáp án của giáo viên cung cấp.
- ✓ Trong mỗi buổi thực hành, sinh viên thực hiện phần bài tập tự làm. Nếu không kịp, sinh viên tự sắp xếp thời gian để hoàn tất bài tập của buổi trước trước khi tiến hành bài thực hành kế tiếp.
- ✓ Khi thực thi câu lệnh, nếu hệ thống báo lỗi, SV cần phải đọc thông báo lỗi, tìm hiểu ý nghĩa và cách giải quyết lỗi đó. Một số lỗi thông dụng được giải thích tại trang 13.

MỤC LỤC

I. MỤC ĐÍCH YÊU CẦU	1
II. NỘI DUNG CỐT LÕI	1
III. KIẾN THỨC TIỀN QUYẾT	1
IV. PHƯƠNG PHÁP HỌC TẬP	2
PHẦN 2. CƠ BẢN VỀ SQL Server 2005	5
I. GIỚI THIỆU SƠ LƯỢC VỀ SQL SERVER 2005	5
1. Kết nối vào Server:	5
2. Cách thức lưu trữ Cơ sở dữ liệu trong SQL Server:	6
3. Chọn hay mở CSDL làm việc:	6
4. Các kiểu dữ liệu thông dụng trong SQL:	6
5. Quan sát CSDL bằng Object Browser:	7
6. Soạn thảo và thực thi câu truy vấn:	9
7. Một số hàm thông dụng trong SQL Server:	10
II. CÁC LỖI THƯỜNG GẶP TRONG SQL	13
1. Tên cột không tìm thấy trong bảng:	13
2. Tên bảng không tìm thấy:	13
3. Lỗi cú pháp:	13
4. Tên cột có mặt ở nhiều bảng:	13
5. Phép so sánh không tương thích kiểu:	13
III. HƯỚNG DẪN THỰC HÀNH	14
1. Mở Query Analyzer đăng nhập vào SQL Server trên máy chủ.	14
2. Quan sát CSDL:	14
3. Mở CSDL muốn truy vấn:	14
4. Soạn thảo và thực hiện câu truy vấn.	14
5. Quan sát kết quả và kiểm chứng.	14
6. Lưu lại các câu truy vấn vào tập tin *.sql	14
IV. CÁCH LƯU và XÓA BẢNG TẠM:	14
V. SAO LƯU/ PHỤC HỒI CSDL:	14
1. Sử dụng lệnh:	14
2. Sử dụng giao diện chương trình:	15
PHẦN 3. BÀI TẬP THỰC HÀNH	20
BÀI 1:	
I. MỤC ĐÍCH	21
II. LÝ THUYẾT	21
III. BÀI TẬP CÓ HƯỚNG DẪN:	21
IV. BÀI TẬP TỰ LÀM:	25
BÀI 2:	
I. MỤC ĐÍCH	27
II. LÝ THUYẾT	27
III. BÀI TẬP CÓ HƯỚNG DẪN:	28
IV. BÀI TẬP TỰ LÀM:	32
BÀI 3:	
I. MỤC ĐÍCH:	34
II. LÝ THUYẾT:	34
III. BÀI TẬP CÓ HƯỚNG DẪN	34
IV. BÀI TẬP TỰ LÀM	37

BÀI 4:

I. MỤC ĐÍCH:	38
II. BÀI TẬP CÓ HƯỚNG DẪN:	38
III. BÀI TẬP TỰ LÀM	41

BÀI 5:

I. MỤC ĐÍCH:	43
II. LÝ THUYẾT:	43
III. BÀI TẬP CÓ HƯỚNG DẪN:	44
IV. BÀI TẬP TỰ LÀM:	46

BÀI 6:

I. MỤC ĐÍCH:	47
II. CSDL: QUẢN LÝ DỰ ÁN (CSDL6)	47
III. CÂU HỎI:	47



TÀI LIỆU THAM KHẢO

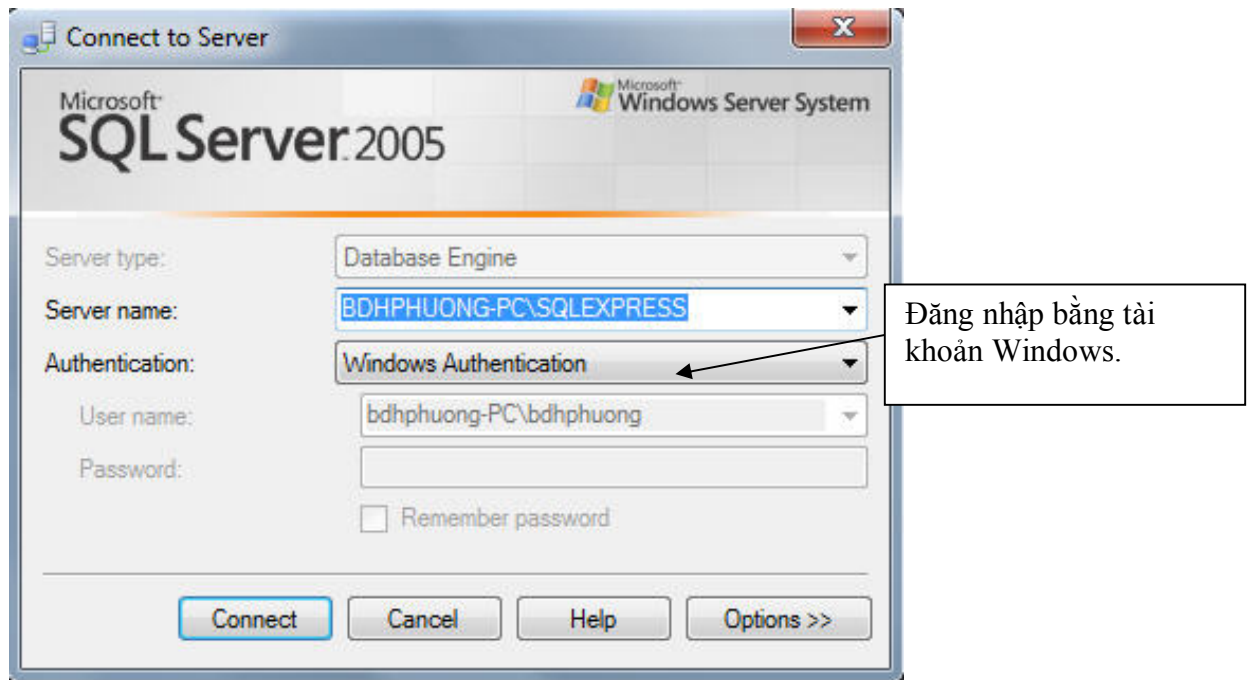
- [1] **Giáo Trình CSQL** - Đinh Khắc Quyền. Khoa Công Nghệ Thông Tin, 2005
- [2] **SQL Tutorial:** <http://www.w3schools.com/sql/>
- [3] **SQL Server 2000 Book Online**

PHẦN 2. CƠ BẢN VỀ SQL SERVER 2005

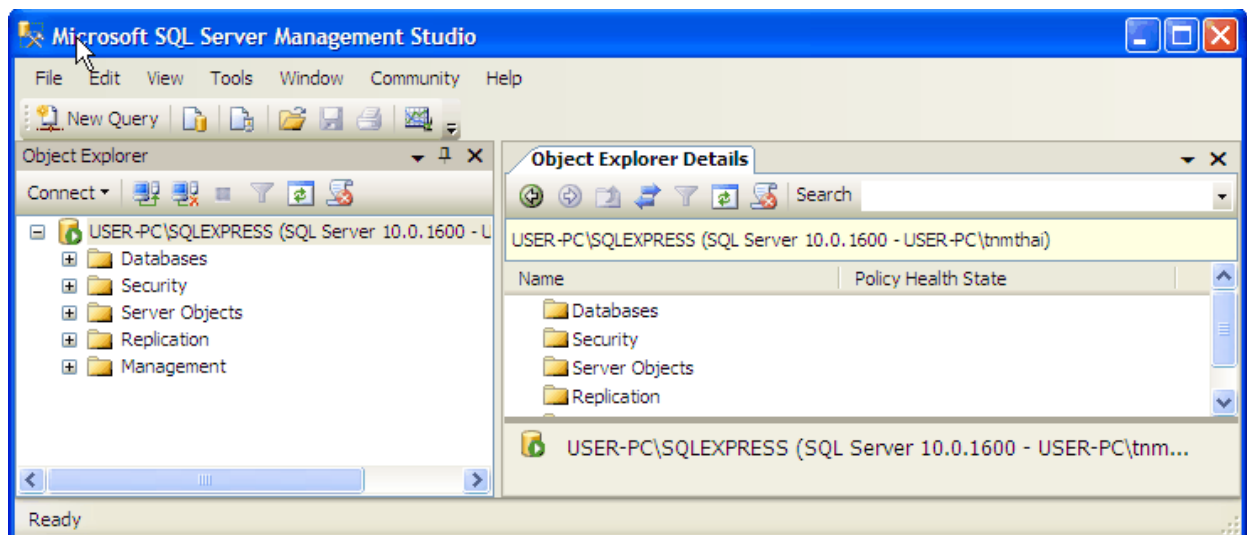
I. GIỚI THIỆU SO LƯỢC VỀ SQL SERVER 2005.

1. Kết nối vào Server:

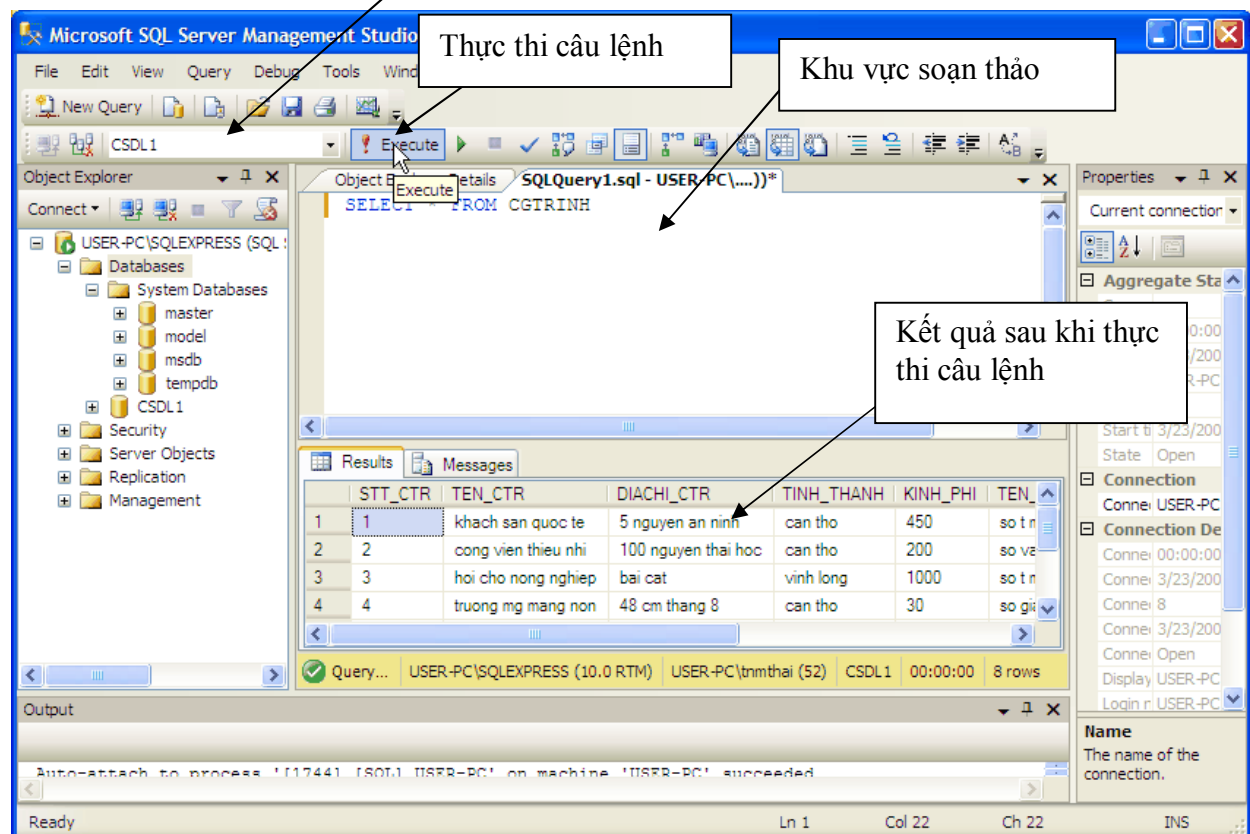
Chọn **Start/All Programs/Microsoft SQL Server 2005/ SQL Server Management Studio Express**. Cửa sổ đăng nhập sẽ xuất hiện như hình 1:



Sau khi chọn **Connect**, cửa sổ **SQL Server Management Studio** sẽ xuất hiện như hình 2:



Để soạn thảo và thực thi câu lệnh, click vào **New Query** như hình 3.



2. Cách thức lưu trữ Cơ sở dữ liệu trong SQL Server:

Một CSDL SQL Server bao gồm:

- 1 hoặc nhiều data file: 1 file dữ liệu chính (*.mdf) và không hoặc nhiều file dữ liệu phụ (*.ndf).
- 1 hoặc nhiều log file (*.ldf): đây chính là nhật ký giao tác trên CSDL.

SQL Server quản lý cùng lúc nhiều CSDL, chia làm hai loại:

- CSDL hệ thống (system databases): dùng để lưu trữ thông tin quản lý toàn bộ hệ thống SQL Server. Có bốn CSDL hệ thống, đó là: master, model, tempdb, msdb
- CSDL người dùng (user databases): là CSDL do người dùng tạo ra. Có 2 CSDL mẫu thông dụng trong SQL Server là: Northwind và pubs

3. Chọn hay mở CSDL làm việc:

Vì hệ thống quản lý nhiều CSDL nên khi làm việc với SQL Server, điều đầu tiên là phải chọn CSDL nào để làm việc trên đó. Có 2 cách:

Cách 1: Nhấp chọn CSDL từ danh sách xổ xuống của combo box trên thanh công cụ (xem hình 2)

Cách 2: Thực thi lệnh: `USE <Tên CSDL>`

4. Các kiểu dữ liệu thông dụng trong SQL:

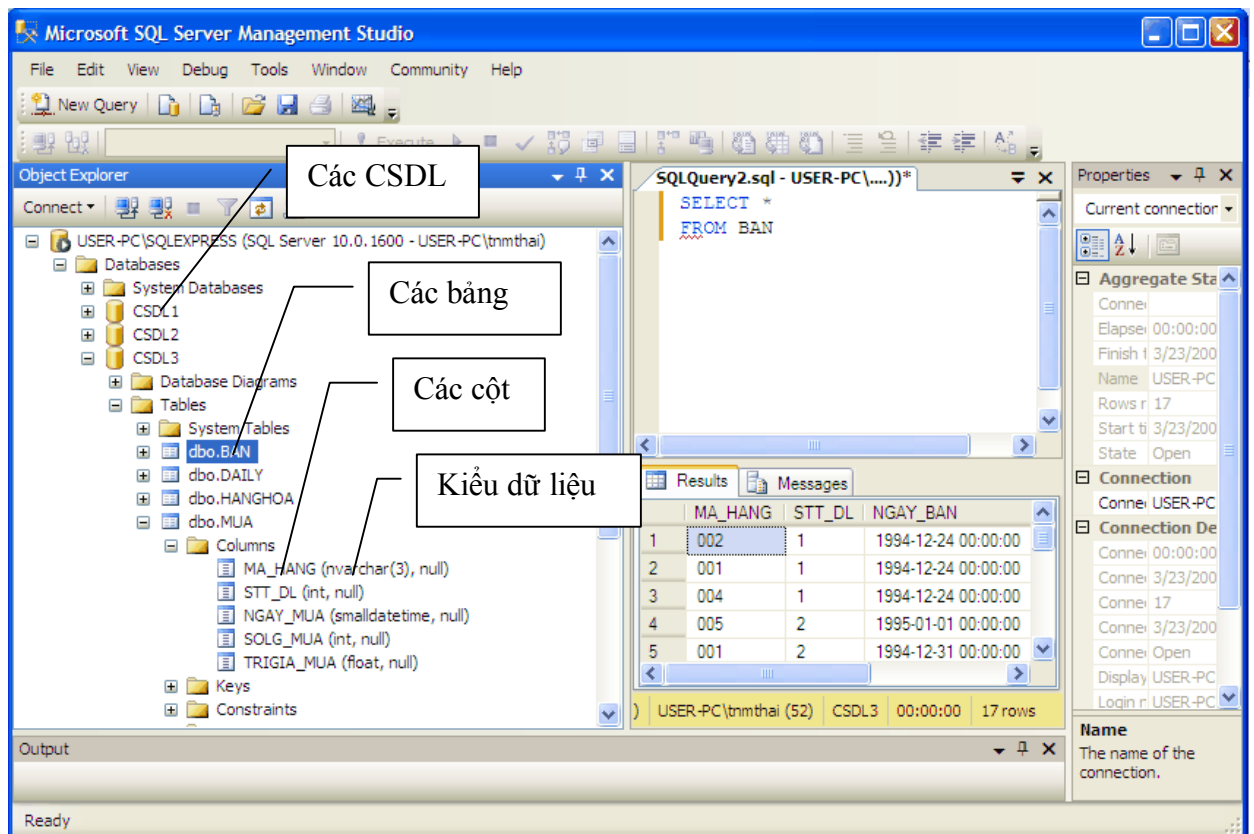
Tên kiểu	Giải thích	Biểu diễn hằng
Bit	Số nguyên 0 hoặc 1	0, 1
Int	Số nguyên từ -2^{31} (-2.147.483.648) đến $2^{31} - 1$ (2.147.483.647)	-1000, 23455
SmallInt	Số nguyên từ -2^{15} (-32.768) đến $2^{15} - 1$	32124, -764

	(32.767)	
TinyInt	Số nguyên từ 0 đến 255	31, 45
Decimal/ Numeric	Các số thập phân từ $-10^{38} - 1$ đến $10^{38} - 1$	1894.1204
Money	Tiền tệ từ -2^{63} đến $2^{63} - 1$	234, -87.65 \$12, \$542023.14
SmallMoney	Tiền tệ từ -214.748,3648 đến 214.748,3647	-\$45.56 -34.54
Float	Số thực từ $-1,79 \text{ E} +308$ đến $1,79 \text{ E} +308$	101.5E5 0.5E-2
Real	Số thực từ $-3.40 \text{ E} + 38$ đến $3.40 \text{ E} + 38$	245.21E-10
DateTime	Ngày giờ từ 1/1/1753 đến 31/10/9999	'April 15, 1998' , '15 April, 1998' , '980415' , '04/15/98'
SmallDateTime	Ngày giờ từ 1/1/1900 đến 06/6/2079	nghư DateTime
Char	Ký tự có độ dài ô nhớ cố định và tối đa là 8000 ký tự (không hỗ trợ Unicode)	'50% complete.' 'O'Brien' "O'Brien"
nchar	Ký tự có độ dài ô nhớ cố định và tối đa là 4000 ký tự (hỗ trợ Unicode)	'Nguyễn Văn Minh' 'Lương Tâm'
varchar	Ký tự có độ dài ô nhớ không cố định và tối đa là 8000 ký tự (không hỗ trợ Unicode)	'50% complete.' 'O'Brien' "O'Brien"
nvarchar	Ký tự có độ dài ô nhớ không cố định và tối đa là 4000 ký tự (hỗ trợ Unicode)	'Nguyễn Văn Minh' 'Lương Tâm'

5. Quan sát CSDL bằng Object Browser:

a. Quan sát lược đồ CSDL:

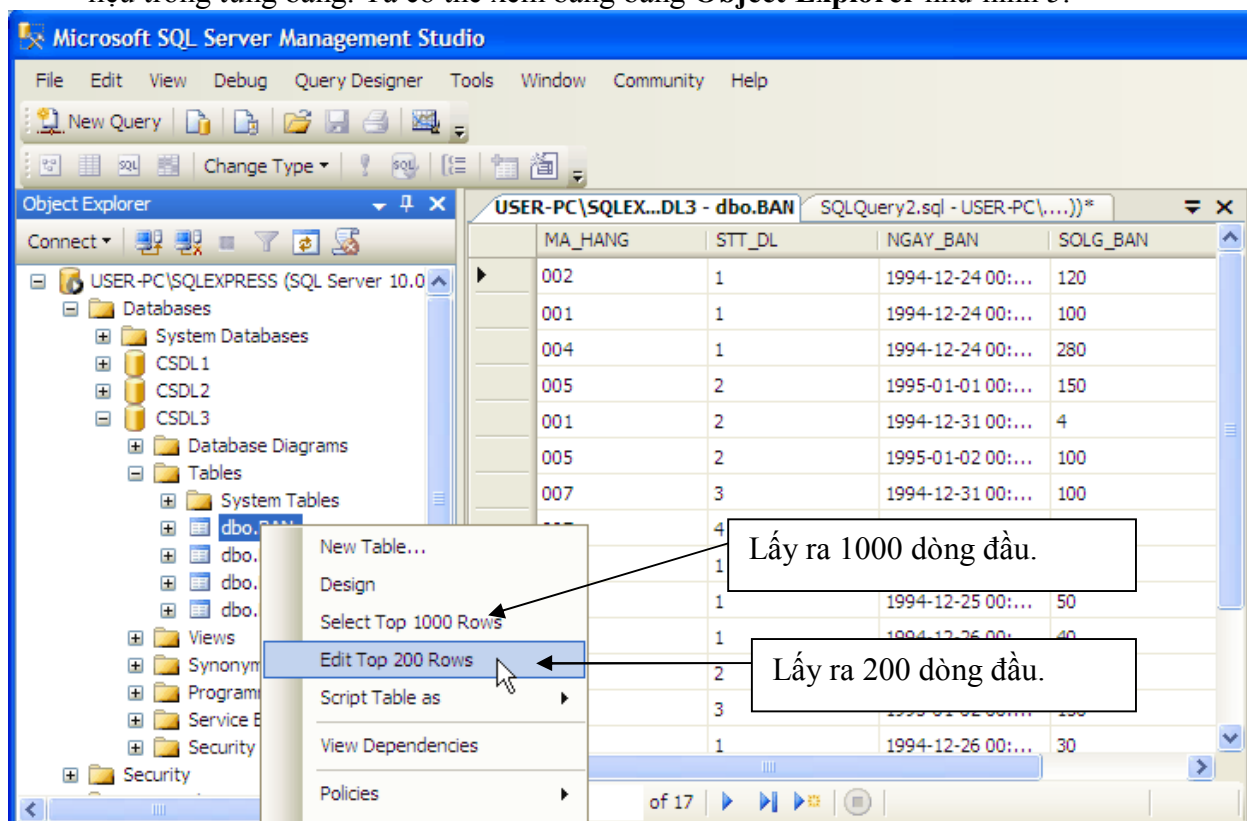
Để có thể truy vấn trên một CSDL nào đó, điều quan trọng là ta phải biết lược đồ của CSDL đó. Ta có thể quan sát lược đồ của các CSDL mà SQL Server đang quản lý bằng cách nhấp vào **View -> Object Explorer**. Khi đó sẽ có một cửa sổ Object Explorer xuất hiện ở bên trái như hình 3 cho phép ta quan sát cấu trúc của các bảng trong CSDL như sau:



Hình 4: Cửa sổ quan sát CSDL.

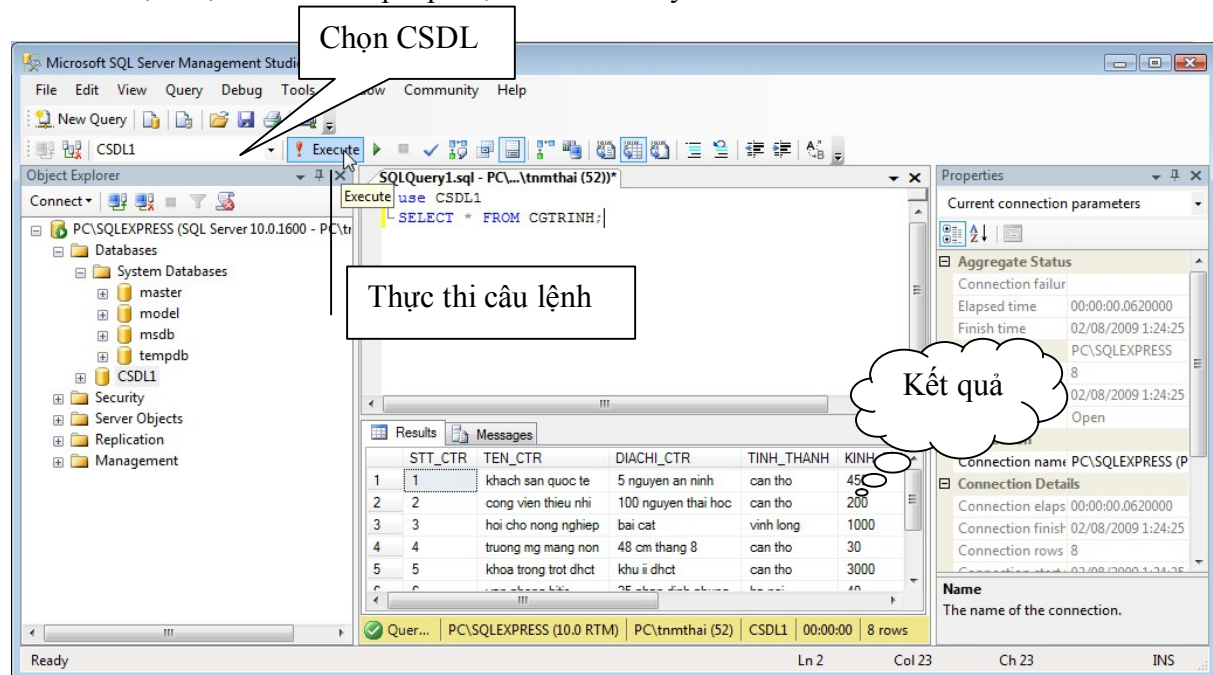
b. Mở bảng dữ liệu:

Để có thể truy vấn được chính xác, ngoài việc quan sát lược đồ CSDL, ta còn phải quan sát dữ liệu trong từng bảng. Ta có thể xem bảng bằng **Object Explorer** như hình 5.



6. Soạn thảo và thực thi câu truy vấn:

Nếu chưa kết nối với SQL Server thì ta không thể thực hiện truy vấn. Để kết nối, ta vào menu **File -> Connect Object Explorer**. Khi đó cửa sổ kết nối sẽ xuất hiện như hình 1 ở trang 4. Sau khi kết nối thì xuất hiện một cửa sổ cho phép soạn thảo câu truy vấn như hình 6.



Trong quá trình thao tác, có thể mở thêm nhiều tập tin truy vấn bằng cách vào menu **File -> New** hoặc click vào biểu tượng trên thanh công cụ.

a. Kiểm tra cú pháp câu lệnh:

Sau khi gõ câu lệnh vào phần nhập câu lệnh, có thể kiểm tra cú pháp câu lệnh bằng cách nhấp vào biểu tượng trên thanh công cụ (xem hình 5) hoặc nhấn **Ctrl + F5**. Nếu câu lệnh không có lỗi cú pháp thì cửa sổ nhỏ bên phải dưới sẽ hiển thị câu:

The command(s) completed successfully.

Nếu câu lệnh có lỗi cú pháp thì lỗi sẽ được hiển thị.

b. Thực thi câu lệnh:

Thực thi tất cả các câu lệnh trong tập tin script hiện hành bằng cách nhấp vào nút trên thanh công cụ (xem hình 5) hoặc nhấn **F5**.

Nếu chỉ muốn thực thi một hoặc một số câu lệnh nào thôi thì bôi đen các lệnh đó và nhấn **F5**.

c. Ghi chú trong SQL Query:

Khi cần ghi chú thích hoặc tạm đóng không thực thi một đoạn lệnh nào đó, ta có thể sử dụng kí hiệu ghi chú như sau:

Ghi chú một dòng: Đặt dấu `--` trước dòng ghi chú

Ghi chú nhiều dòng: Đặt ghi chú trong cặp dấu `/* ... */`

d. Lưu lại truy vấn trong tập tin lệnh (script): (*.sql)

Sau khi nhập vào các câu truy vấn (như hình 5), các câu lệnh truy vấn có thể được lưu lại dưới dạng file text (*.sql) bằng cách chọn **File > Save** hoặc nhấn **Ctrl + S**.

Tập tin này sau khi lưu vào đĩa có thể được mở lên lại để thực thi bằng cách vào **File > Open**.

7. Một số hàm thông dụng trong SQL Server:

a. Xử lý chuỗi:

Lưu ý: SQL Server không phân biệt chữ hoa và chữ thường. Vì vậy:
 điều kiện: `lower(ten_ctr)='khach san QUOC TE'` => vẫn cho ra kết quả

Một số hàm xử lý chuỗi thông dụng:

- **LEN(<chuỗi>):** Trả về chiều dài chuỗi
- **LOWER(<chuỗi>):** đổi <chuỗi> sang dạng chữ thường
 Ví Dụ: `Select LOWER("Tran PHI phap")` => "tran phi phap"
- **UPPER(<chuỗi>):** đổi <chuỗi> sang dạng chữ hoa
 Ví Dụ: `Select UPPER("Tran PHI phap")` => "TRAN PHI PHAP"
- **LEFT(<chuỗi>, <số n>):** trả về chuỗi mới gồm n ký tự bên trái của <chuỗi>
 Ví Dụ: `Select LEFT('Tin Hoc', 3)` => 'Tin'
- **RIGHT(<chuỗi>, <số n>):** trả về chuỗi mới gồm n ký tự bên phải của <chuỗi>
- **SUBSTRING(<chuỗi>, <số n>, <số m>):** trả về chuỗi con bằng cách lấy từ <chuỗi> m ký tự từ vị trí n.
 Ví Dụ: `Select SUBSTRING('Tin Mới Học', 5, 3)` => 'Mới'
- **STUFF(<chuỗi 1>, <số n>, <số m>, <chuỗi 2>):** thay m ký tự trong <chuỗi 1> từ vị trí n bằng <chuỗi 2>.
 Ví Dụ: `Select STUFF('Tin Học', 5, 0, 'Mới')` => 'Tin Mới Học' -- xen vào
`Select STUFF('Tin Hầy Học', 5, 3, 'Mới')` => 'Tin Mới Học' -- thay thế
`Select STUFF('Tin Hầy Học', 5, 4, '')` => 'Tin Học' -- xóa
- **CHARINDEX (<chuỗi 1>, <chuỗi 2> [, <số n>]):** trả về vị trí xuất hiện lần đầu tiên của chuỗi 1 trong chuỗi 2, bắt đầu tìm từ vị trí thứ n của chuỗi 2.
 Ví Dụ: `Select CHARINDEX('qua', 'noi qua ma hong qua')` => 5
`Select CHARINDEX('qua', 'noi qua ma hong qua', 6)` => 17
- **PATINDEX (<'%mẫu tìm kiếm%'>, <chuỗi>):** trả về vị trí bắt đầu xuất hiện mẫu lần đầu tiên trong chuỗi, nếu không tìm thấy mẫu thì trả về 0.
 Ví Dụ: `Select PATINDEX('%om%', 'lom khom duoi nui tieu vai chu')` => 2
`Select PATINDEX('%__om %', 'lom khom duoi nui tieu vai chu')` => 5

b. Xử lý ngày

Lưu ý: - Kiểu *DATETIME* của SQL Server bao gồm nhiều thành phần: ngày / tháng / năm/ giờ / phút / giây / phần triệu của giây.
 - Để tránh nhầm lẫn: nên mô tả năm bằng 4 chữ số, mô tả tháng bằng tên tắt của tháng (Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec).

- **GETDATE():** Cho kết quả là ngày hiện hành.
- **DATEPART(<thành phần>, <ngày>):** Trả về giá trị của <thành phần> trong <ngày>.

Các thành phần thông dụng của ngày gồm có:

Thành phần	Viết tắt	Ý nghĩa
Year	yy, yyyy	Năm
Quarter	qq, q	Quý
Month	mm, m	Tháng
Dayofyear	dy, y	Ngày của năm (1-> 366)
Day	dd, d	Ngày của tháng (1 -> 31)
Week	wk, ww	Tuần trong năm (1 -> 52)
WeekDay	Dw	Ngày trong tuần (1 -> 7)
Hour	hh	Giờ (1 -> 24)

Ví Dụ: `Select DATEPART (year, GETDATE()) => 2006`

`Select DATEPART(week, '20 Aug, 2006') => 34`

- **DATEADD** (<thành phần>, <số n>, <ngày>): cộng vào <thành phần> của <ngày> một số n.

Ví Dụ: `Select DATEADD (week, 5, '20 Aug, 2006') => 2006-09-24`

Cho biết 10 ngày sau kể từ ngày 05/08/2006 là ngày thứ mấy:

`Select DATEPART (weekday, DATEADD (day, 10, '5 Aug, 2006')) => 3(thứ ba)`

- **DATEDIFF** (<thành phần> , <ngày bắt đầu>, <ngày kết thúc>): trả về số khác biệt giữa 2 thành phần của ngày bắt đầu và ngày kết thúc.

Ví Dụ: Cho biết từ ngày 05/08/2006 đến ngày 11/9/2006 có bao nhiêu ngày:

`Select DATEDIFF (day, '5 Aug, 2006', '11 Sep, 2006') => 37 ngày`

`Select DATEDIFF (hour, '5 Aug, 2006', '11 Sep, 2006') => 888 giờ`

- **DATENAME**(<thành phần>, <ngày>): Trả về tên của <thành phần> trong <ngày>.

Ví Dụ: `Select DATENAME (month, '5 Aug, 2006') => August`

`Select DATENAME (weekday, '5 Aug, 2006') => Saturday`

c. Các hàm xử lý số

- **FLOOR**(<số>): Cho kết quả là số nguyên gần nhất nhỏ hơn hoặc bằng <số>

Ví Dụ: `SELECT FLOOR(123.45), FLOOR(-123.45), FLOOR($123.45)`

`=> 123 -124 123.0000`

- **CEILING**(<số>): Cho kết quả là số nguyên gần nhất lớn hơn hoặc bằng <số>

Ví Dụ: `SELECT CEILING(123.45), CEILING(-123.45), CEILING($123.45)`

`=> 124 -123 124.0000`

d. Hàm chuyển đổi kiểu

- **CAST** (<dữ liệu> AS <kiểu>): Trả về <dữ liệu> với <kiểu> mới.

Ví Dụ: `SELECT 10 + cast ('34.5' as float) => 44.5`

`SELECT right(CAST(124 AS char(4)) , 2) => 4`

e. Cấu Trúc Phân Nhánh CASE:

Đánh giá danh sách các điều kiện và trả về biểu thức phù hợp.

CASE có hai dạng:

- Dạng đơn giản: So sánh hai biểu thức để xác định kết quả trả về.

```
CASE <tên cột>
  WHEN <giá trị 1> THEN <kết quả 1>
  [WHEN <giá trị 2> THEN <kết quả 2> ...]
  [
    ELSE <kết quả ngoại lệ>
  ]
END
```

- Dạng tìm kiếm: Đánh giá các điều kiện để xác định kết quả trả về.

```
CASE
  WHEN <biểu thức điều kiện 1> THEN <kết quả trả về 1>
  [WHEN <biểu thức điều kiện 2> THEN <kết quả trả về 2> ...]
  [
    ELSE <kết quả trả về ngoại lệ>
  ]
END
```

```
Ví Dụ:      Select CASE
              When 10 > 2 THEN 'DUNG'
              When 10 = 3 THEN 'SAI'
            END
=> DUNG
```

II. CÁC LỖI THƯỜNG GẶP TRONG SQL

1. Tên cột không tìm thấy trong bảng:

Thông báo lỗi: Invalid column name <tên cột>.

Khi gặp lỗi này hãy xem lại tên cột (còn gọi là tên thuộc tính/ tên trường) trong câu lệnh đã đúng chưa.

2. Tên bảng không tìm thấy:

Thông báo lỗi: Invalid object name <tên bảng>.

Khi gặp lỗi này hãy xem lại tên bảng trong câu lệnh đã đúng chưa.

3. Lỗi cú pháp:

Thông báo lỗi: Incorrect Syntax near <từ gần lỗi nhất>.

Khi gặp lỗi này hãy xem lại **cú pháp** trong câu lệnh đã đúng chưa, có thể sai từ khóa (gõ sai các từ SELECT, FROM, WHERE, ..), hoặc thiếu dấu ngoặc của hàm hoặc của select con, ...

4. Tên cột có mặt ở nhiều bảng

Thông báo lỗi: Ambiguous column name <tên cột>

Khi gặp lỗi này hãy xem lại <tên cột> gây lỗi trong câu lệnh đã có đặt tên bảng hoặc bí danh đứng trước chưa. Cú pháp: <tên bảng hoặc bí danh>.**MA_HANG**

5. Phép so sánh không tương thích kiểu

Thông báo lỗi: Error converting data type ...

hoặc Syntax error converting ...

Khi gặp lỗi này hãy xem lại trong câu lệnh đã có phép toán so sánh nào đó không tương thích kiểu dữ liệu.

III. HƯỚNG DẪN THỰC HÀNH

1. Mở Query Analyzer đăng nhập vào SQL Server trên máy chủ.

2. Quan sát CSDL:

Sử dụng Object Browser để xem cấu trúc của các bảng cũng như dữ liệu của từng trường và mở từng bảng dữ liệu để quan sát như hướng dẫn ở mục 5 ở trang 7 .

Khi quan sát CSDL cần lưu ý:

- Sự liên thông giữa các bảng dữ liệu. Tốt nhất là vẽ sơ đồ liên thông lên giấy nháp để có thể tham khảo trong quá trình truy vấn thông tin từ nhiều bảng.
- Kiểu dữ liệu của từng trường, để đặt điều kiện trên các cột cho đúng kiểu hoặc áp dụng các hàm xử lý dữ liệu phù hợp.

3. Mở CSDL muốn truy vấn:

Như đã hướng dẫn ở mục 3 ở trang 6

4. Soạn thảo và thực hiện câu truy vấn.

Như đã hướng dẫn ở mục **Error! Reference source not found.** ở trang **Error! Bookmark not defined.**

5. Quan sát kết quả và kiểm chứng.

Cần phải kiểm tra lại kết quả truy vấn xem có đúng hay không bằng cách mở lại các bảng để kiểm tra xem kết quả đạt được có đúng như mong muốn.

6. Lưu lại các câu truy vấn vào tập tin *.sql.

IV. CÁCH LƯU và XÓA BẢNG TẠM:

Trong khi thực hiện một số truy vấn, thỉnh thoảng ta cần lưu lại những kết quả truy vấn vào một bảng nào đó, thì ta dùng mệnh đề INTO sau mệnh đề SELECT và trước FROM.

Ví dụ: Select * **Into NV** From NhanVien;

Sau khi thực thi câu lệnh này, trong CSDL sẽ xuất hiện thêm một bảng NV lưu kết quả của câu truy vấn.

Trong trường hợp ta chỉ muốn bảng này tồn tại cho đến hết phiên làm việc (session) hiện tại thôi, thì ta có thể lưu vào bảng tạm (là những bảng có tên bắt đầu bằng dấu #), các bảng này sẽ được tạo trong CSDL tempdb của hệ thống. Khi client ngưng kết nối với Server thì các bảng này sẽ tự động bị xóa.

Ví Dụ: Select STT_CTR, count(*) As TongSoCN **Into #CAU3E**
 From THAMGIA
 Group by STT_CTR

Trong khi client vẫn còn đang kết nối thì những bảng tạm này vẫn còn trong CSDL tempdb. Vì vậy, sau khi chạy xong câu ví dụ trên, nếu ta chạy lại câu lệnh này thì SQL sẽ báo lỗi là bảng tạm này đã tồn tại. Nên ta phải xóa bảng tạm này trước khi muốn chạy lại câu lệnh trên bằng lệnh xóa bảng DROP TABLE <tên bảng>.

VD: DROP TABLE #CAU3E

V. SAO LƯU/ PHỤC HỒI CSDL:

Có thể chép CSDL từ máy này sang máy khác bằng cách sử dụng lệnh sao lưu / phục hồi CSDL của SQL Server. Có thể thực hiện bằng lệnh hoặc giao diện.

1. Sử dụng lệnh:

a.Sao lưu CSDL:

Để sao lưu một CSDL ta thực hiện các bước sau:

Bước 1. Mở CSDL muốn sao lưu bằng lệnh:

USE <Tên CSDL>

Bước 2. Tạo thiết bị sao lưu gắn với một tập tin sao lưu bằng lệnh:

EXEC sp_addumpdevice 'disk' , '<tên thiết bị>' , '<đường dẫn và tên tập tin>'

Bước 3. Ra lệnh sao lưu CSDL vào thiết bị sao lưu:

BACKUP DATABASE <tên CSDL muốn chép> TO <Tên thiết bị>

VD: Sao lưu CSDL có tên là CSDL1 vào ổ đĩa D và đặt tên là CSDL1_backup.bak:

Mở CSDL cần chép:

USE CSDL1

Tạo thiết bị sao lưu tên BAK1 gắn với tập tin D:\CSDL1_backup.bak

EXEC sp_addumpdevice 'disk', 'BAK1', 'D:\CSDL1_backup.bak'

Ra lệnh sao lưu CSDL1 vào thiết bị sao lưu vừa tạo BAK1

BACKUP DATABASE CSDL1 TO BAK1

Lưu ý: Không thể tạo hai thiết bị sao lưu cùng tên. Nghĩa là nếu đã tạo thiết bị BAK1 rồi thì những lần sao chép sau không cần phải thực hiện bước 2 nữa.

b.Phục hồi CSDL từ tập tin đã sao lưu:

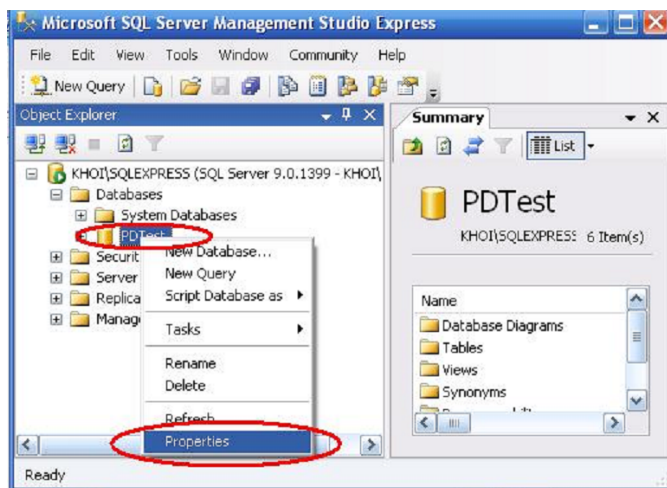
Giả sử ta muốn chép CSDL1 về máy nhà (đã có cài SQL Server). Thì ta sẽ thực hiện các bước sau:

- Thực hiện sao lưu CSDL1 như đã hướng dẫn ở mục trên.
- Chép tập tin sao lưu về máy nhà.
- Mở Query Analyzer
- Ra lệnh phục hồi CSDL1 từ tập tin sao lưu:

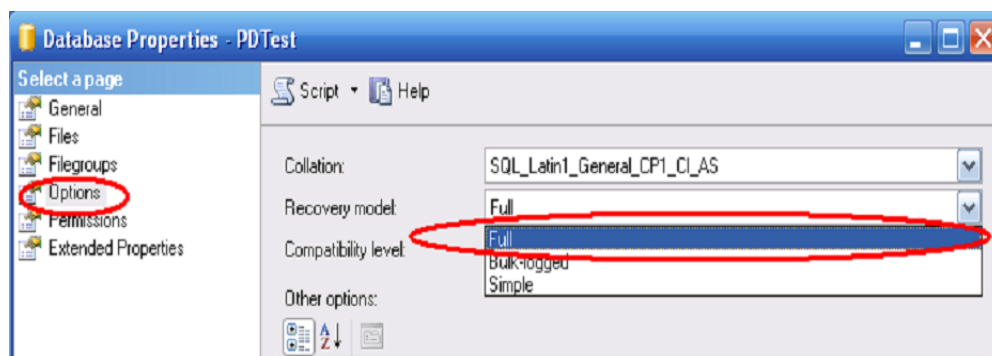
RESTORE DATABASE <Tên CSDL> FROM DISK = '<đường dẫn và tên file sao lưu>'

2. Sử dụng giao diện chương trình:**a.Sao lưu CSDL:**

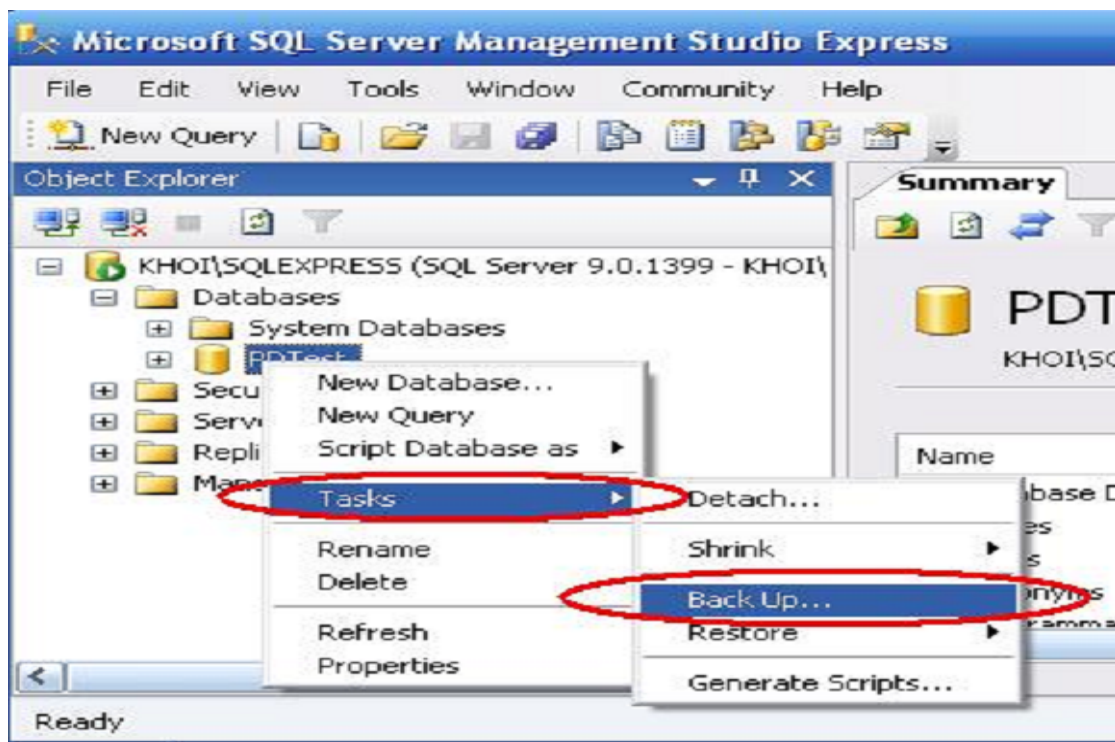
Bước 1: Từ nút “Databases”, click phải lên CSDL muốn sao lưu sau đó chọn “Properties”



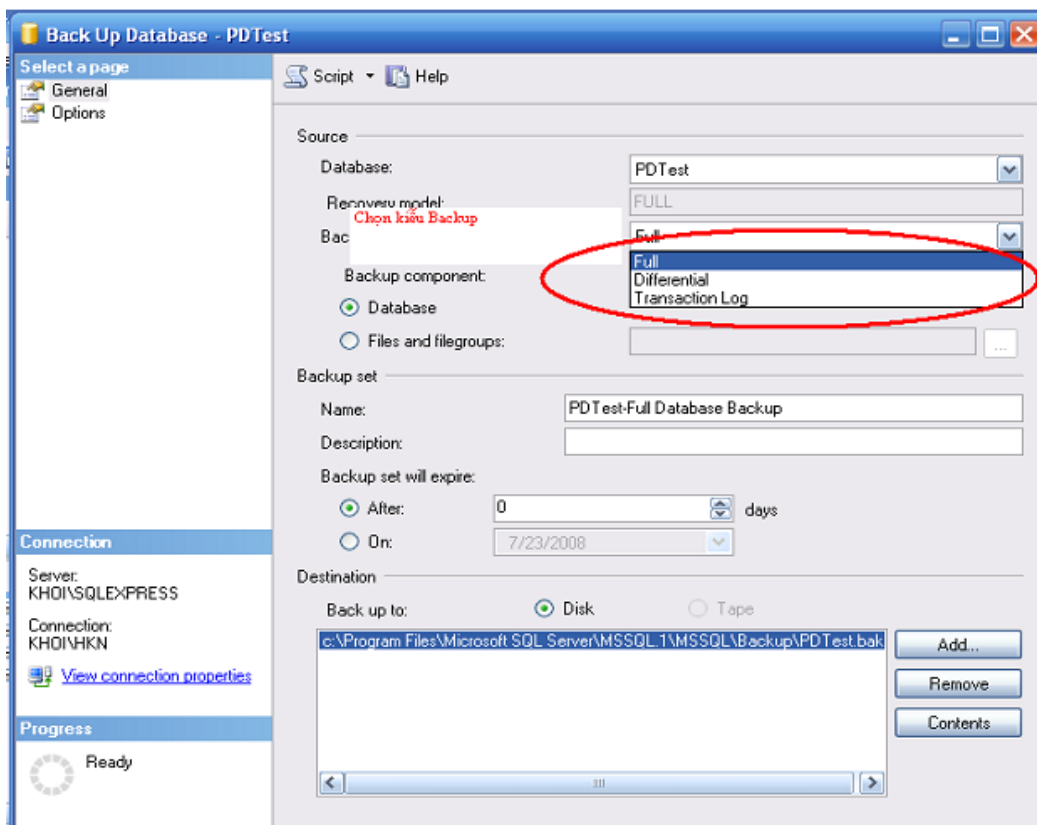
Chọn OK



Bước 2: Từ nút “Databases”, click phải lên CSDL muốn sao lưu sau đó chọn “Tasks” và chọn “Backup” .

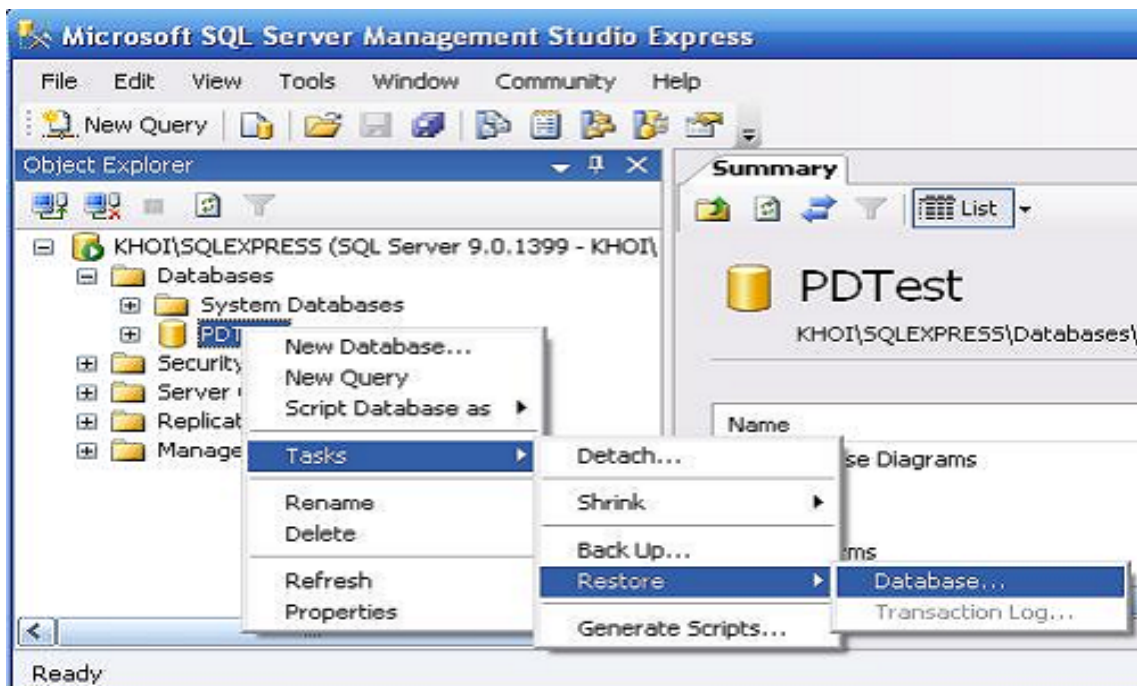


Bước 3: Nhấn chọn “Add..” và nhập vào đường dẫn và tên file sao lưu, nhấn OK để quá trình sao lưu hoàn tất

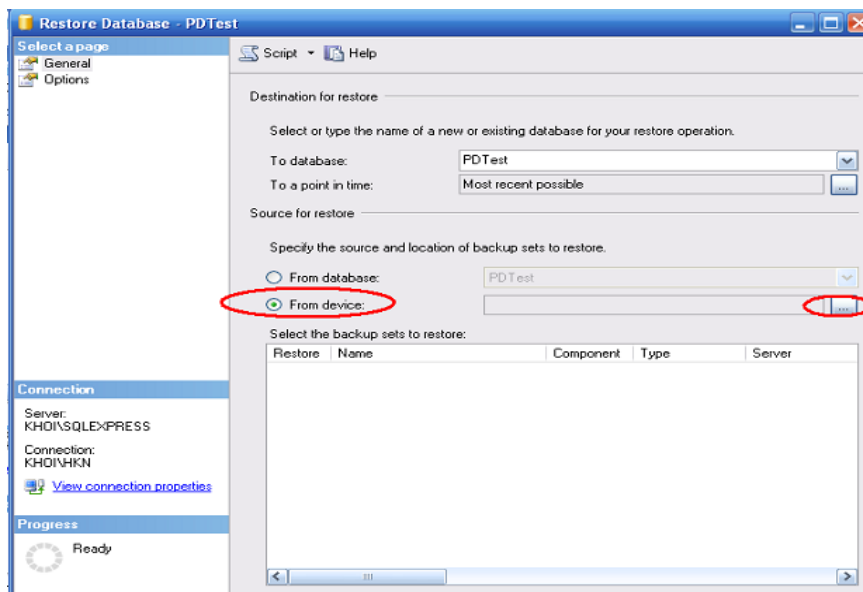


b. Phục hồi CSDL:

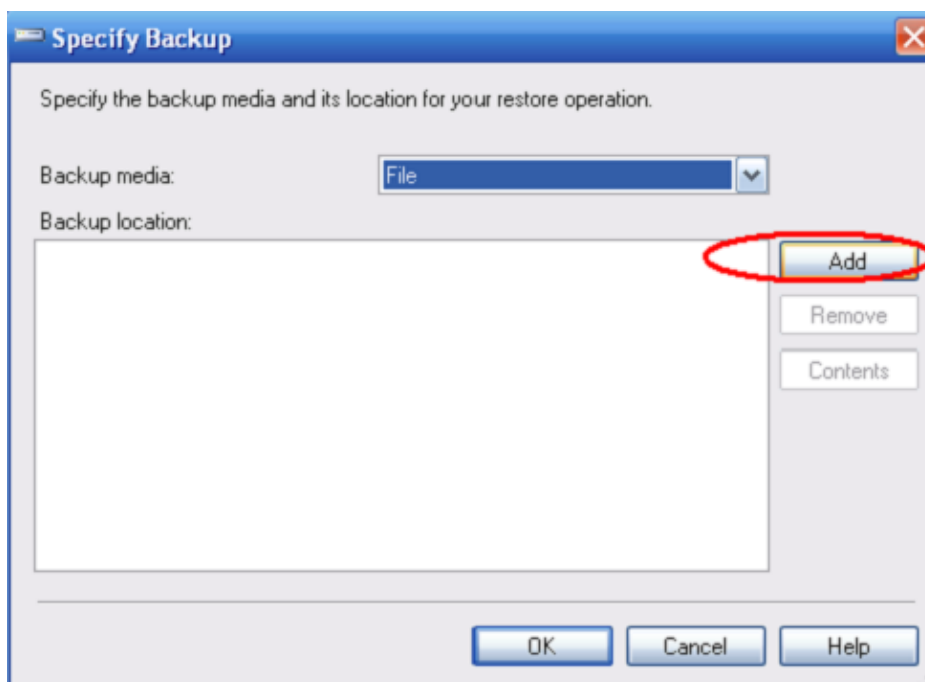
Bước 1: Click chuột phải lên “Databases” sau đó chọn “Tasks” và chọn “Restore”, chọn “Database”



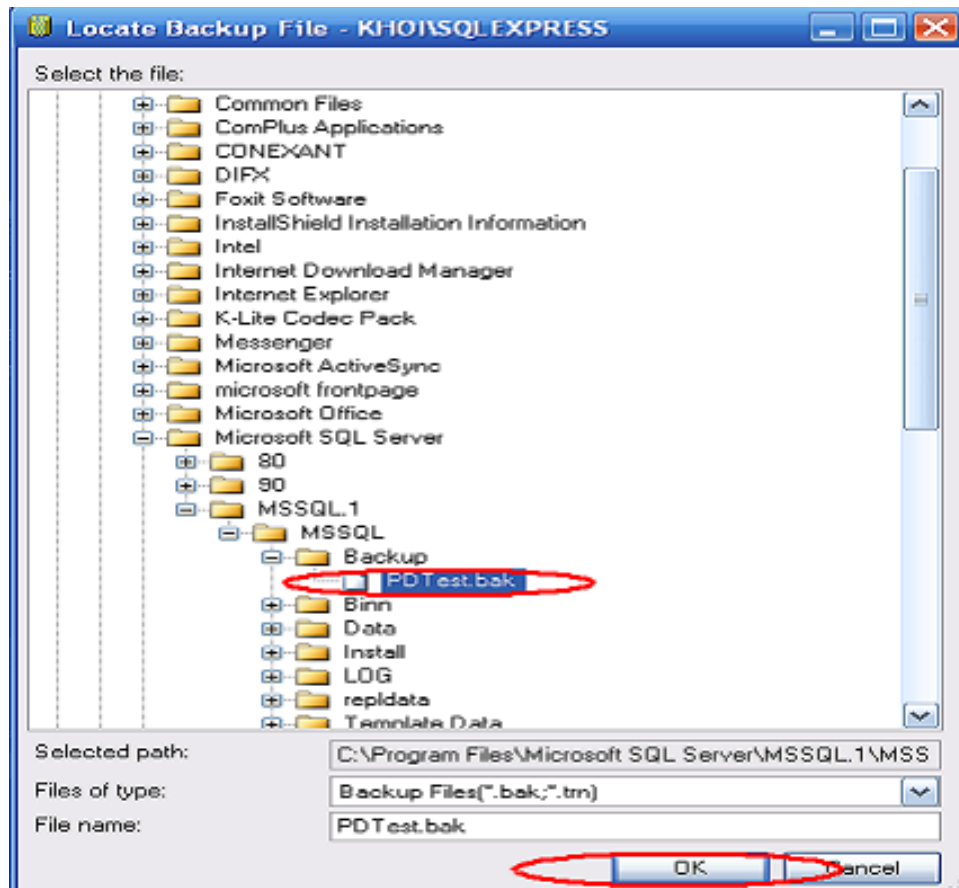
Bước 2: Chọn “From device” sau đó chọn đường dẫn đến file đã sao lưu CSDL trước đó



Nhấn Add



Nhấn OK



PHẦN 3. BÀI TẬP THỰC HÀNH

Phần thực hành được chia thành 6 bài. Trong 4 bài đầu tiên, mỗi bài sẽ tập trung rèn luyện một dạng câu hỏi truy vấn, bài 5 dành cho các câu lệnh tạo, sửa bảng, xen, sửa, xóa dòng dữ liệu. Mỗi bài sẽ có bố cục như sau:

1. Mục tiêu
2. Lý thuyết: Nhắc lại một cách ngắn gọn câu lệnh cần dùng trong bài.
3. Bài tập có hướng dẫn: Gồm một số câu hỏi truy vấn trên một CSDL cho trước. Mỗi câu hỏi sẽ được phân tích để đi đến câu lệnh. Những điểm cần lưu ý sẽ được nhấn mạnh. Ngoài ra, một số câu hỏi nhỏ sẽ được đưa ra trong tình huống của câu hỏi giúp sinh viên hiểu rõ hơn vấn đề.
4. Bài tập tự làm: gồm một số câu tương tự như trong phần bài tập có hướng dẫn nhưng không kèm đáp án.

Bài thực hành 6 dành cho bài tập tổng hợp gồm tất cả các dạng câu hỏi, và sinh viên sẽ tự thực hiện mà không có phần hướng dẫn.

BÀI 1

I. MỤC ĐÍCH

Luyện tập khả năng nhận biết sự thông thương trong một cơ sở dữ liệu, từ đó kết nối dữ liệu để truy xuất thông tin cần thiết.

II. LÝ THUYẾT

Câu lệnh SQL sử dụng trong bài này:

SELECT <DS tên cột>

FROM <Tên bảng> [<bí danh>] [, <Tên bảng> [<bí danh>] [...]]

[WHERE <DS các ĐK AND hoặc OR với nhau>]

III. BÀI TẬP CÓ HƯỚNG DẪN:

1. CSDL: Quản lý công trình (CSDL1)

KTRUCSU(HOTEN_KTS, NAMS_KTS, PHAI, NOI_TN, DCHI_LL_KTS)

Mỗi kiến trúc sư có một họ tên, một năm sinh, phái, nơi tốt nghiệp, địa chỉ liên lạc. Giả sử không có hai kiến trúc sư trùng tên.

CHUTHAU(TEN_THAU, TEL, DCHI_THAU)

Mỗi chủ thầu xây dựng công trình đều có một tên thầu, một số điện thoại và một địa chỉ. Giả sử không có hai chủ thầu trùng tên.

CHUNHAN(TEN_CHU, DCHI_CHU)

Mỗi chủ nhân của một công trình đều có một tên chủ nhân và một địa chỉ. Giả sử không có hai chủ nhân trùng tên.

CONGNHAN(HOTEN_CN, NAMS_CN, NAM_VAO_N, CH_MON)

Mỗi công nhân tham gia xây dựng các công trình đều có một họ tên, một năm sinh, một năm vào nghề, và một chuyên môn nào đó. Giả sử không có hai công nhân trùng họ tên.

CGTRINH(STT_CTR, TEN_CTR, DCHI_CTR, TINH_THANH, KINH_PHI, TEN_CHU, TEN_THAU, NGÀY_BD)

Mỗi công trình có một số thứ tự, một tên công trình, một địa chỉ, thuộc về một tỉnh thành nào đó, được xây dựng với một kinh phí (đơn vị tính là triệu đồng), thuộc về sở hữu của một chủ nhân, do một chủ thầu xây dựng, và được khởi công xây từ một ngày nào đó.

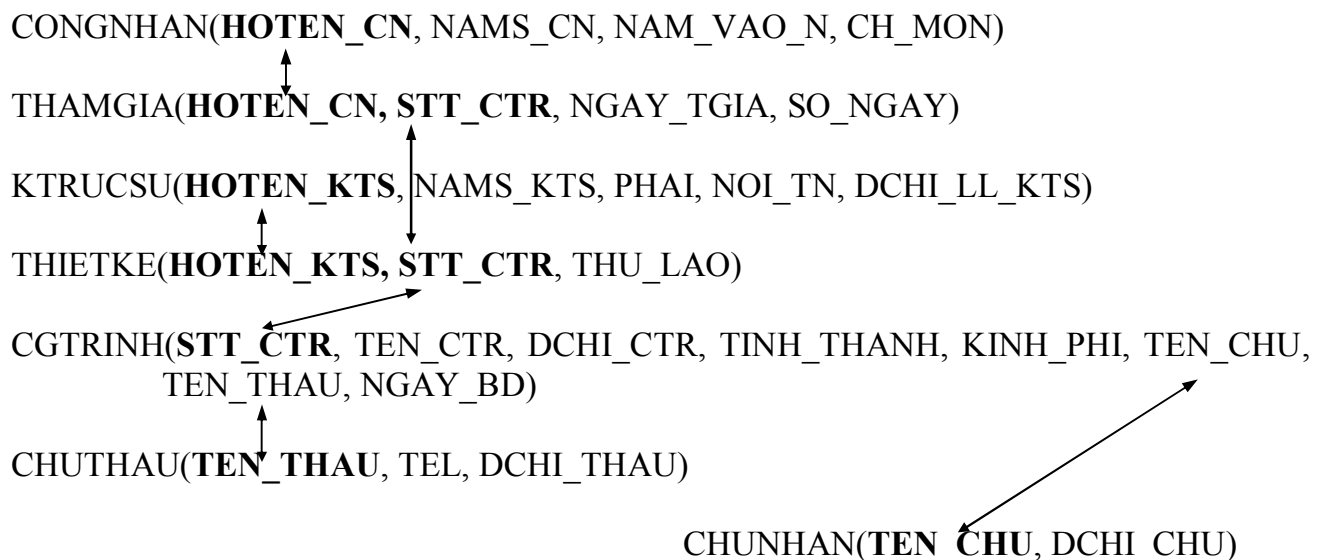
THAMGIA(HOTEN_CN, STT_CTR, NGÀY_TGIA, SO_NGAY)

Một công nhân có thể tham gia xây dựng nhiều công trình, và một công trình cũng có nhiều công nhân tham gia. Khi một công nhân tham gia vào một công trình nào đó sẽ được ghi nhận lại ngày bắt đầu tham gia và số ngày tham gia.

THIETKE(HOTEN_KTS, STT_CTR, THU_LAO)

Một kiến trúc sư có thể thiết kế nhiều công trình, mỗi công trình cũng có thể do nhiều kiến trúc sư cùng thiết kế. Khi một kiến trúc sư thiết kế một công trình sẽ có một thù lao tương ứng.

2. Yêu cầu 1: Tìm sự thông thương giữa các bảng trong CSDL:



3. Yêu cầu 2: Mở các bảng dữ liệu để xem kiểu dữ liệu của từng trường và quan sát dữ liệu của từng bảng.

4. Yêu cầu 3: Hãy viết câu lệnh SQL để trả lời các câu hỏi sau:

a. Hãy cho biết tên và địa chỉ các công trình do chủ thầu Công ty xây dựng số 6 thi công.

Phân tích câu hỏi:

Yêu cầu cho biết: TEN_CTR, DCHI_CTR

=> lấy từ bảng: CGTRINH

Điều kiện: TEN_THAU = 'Cty xd so 6'

=> Trong CSDL, tên thầu có mặt ở cả

hai bảng CGTRINH và CHUTHAU, nên ta chỉ cần đặt ĐK trên bảng CGTRINH đã chọn ở bước trên.

Lưu ý: Sinh viên cần phải mở bảng CGTRINH để xem dữ liệu thực tế của bảng; như trong trường hợp này tên thầu Công Ty xây dựng số 6 được lưu trữ thành chuỗi 'Cty xd so 6'.

Câu truy vấn:

```

Select      TEN_CTR, DCHI_CTR
From        CGTRINH
Where       TEN_THAU = 'Cty xd so 6'
  
```

b. Tìm tên và địa chỉ liên lạc của các chủ thầu thi công công trình ở Cần Thơ do kiến trúc sư Lê Kim Dung thiết kế.

Phân tích câu hỏi:

Yêu cầu cho biết: TEN_THAU, DCHI_THAU => Lấy từ bảng: CHUTHAU

Điều kiện: TINH_THANH = 'Can tho' => Đặt ĐK trên bảng: CGTRINH

Và HOTEN_KTS = 'Le Kim Dung'

=> Trong CSDL, họ tên KTS có mặt

ở cả hai bảng KTRUCSU và THIETKE, nhưng ta phải chọn bảng THIETKE, vì bảng này mới cho biết KTS nào thiết kế công trình nào, vì vậy ta đặt ĐK trên bảng: THIETKE

Như vậy, ta sẽ truy xuất 3 bảng: CHUTHAU, CGTRINH, THIETKE. Ta thấy các bảng này đều liên thông với nhau, nghĩa là chúng có cột chung, có thể nối kết tự nhiên được.

Lưu ý: Khi câu truy vấn truy xuất thông tin từ nhiều bảng thì nhớ phải:

- Mô tả đầy đủ các điều kiện kết nối các bảng.
- Các cột chung phải đặt tên bảng (hoặc bí danh nếu bảng đã có bí danh) phía trước tên cột, phân cách bằng dấu chấm.
- Khi đặt bí danh cho bảng có thể có hoặc không có từ khóa AS

Câu truy vấn:

```

Select      a.TEN_THAU, DCHI_THAU
From  CHUTHAU as a, CGTRINH b, THIETKE as c
Where      a.TEN_THAU = b.TEN_THAU
          And  b.STT_CTR = c.STT_CTR;
          And  TINH_THANH = 'Can tho'
          And  HOTEN_KTS = 'Le Kim Dung'

```

c. Hãy cho biết nơi tốt nghiệp của các kiến trúc sư đã thiết kế công trình Khách Sạn Quốc Tế ở Cần Thơ.

Phân tích câu hỏi:

Yêu cầu cho biết: NOI_TN => Lấy từ bảng: KTRUCSU
 Điều kiện: TEN_CTR = 'KS Quoc Te' } => Đặt ĐK trên bảng: CGTRINH
 Và TINH_THANH = 'Can Tho' }

Như vậy, ta sẽ truy xuất 2 bảng: KTRUCSU và CGTRINH. Tuy nhiên, ta thấy hai bảng này không liên thông nhau hay không có cột chung, không thể nối kết tự nhiên được. Hay nói cách khác, hai bảng này không cho biết KTS nào đã thiết kế công trình Khách sạn quốc tế. Vậy, ta phải thêm vào bảng THIETKE trong truy vấn.

Câu truy vấn:

```

Select      NOI_TN
From  KTRUCSU as a, CGTRINH as b, THIETKE as c
Where      a.HOTEN_KTS = c.HOTEN_KTS
          And  b.STT_CTR = c.STT_CTR;
          And  TEN_CTR = 'KS Quoc Te'
          And  TINH_THANH = 'Can Tho'

```

d. Cho biết họ tên, năm sinh, năm vào nghề của các công nhân có chuyên môn hàn hoặc điện đã tham gia các công trình mà chủ thầu Lê Văn Sơn đã trúng thầu.

Phân tích câu hỏi:

Yêu cầu cho biết: HOTEN_CN, NAMS_CN, NAM_VAO_N => Lấy từ bảng: CONGNHAN

Điều kiện: CH_MON = 'Han' } => Đặt ĐK trên bảng: CONGNHAN
 Hoặc CH_MON = 'Dien' }
 Và TEN_THAU = 'Le Van Son' => Đặt ĐK trên bảng: CGTRINH

Như vậy, ta sẽ truy xuất 2 bảng: CONGNHAN và CGTRINH. Tương tự như câu c, hai bảng này không cho biết công nhân nào đã tham gia công trình nào. Vậy, ta phải thêm vào bảng THAMGIA trong truy vấn.

Lưu ý: Trong danh sách các điều kiện vừa có AND vừa có OR, ta phải dùng dấu ngoặc để chỉ rõ thứ tự kiểm tra điều kiện.

Câu truy vấn:

```

Select      HOTEN_CN, NAMS_CN, NAM_VAO_N
From  CONGNHAN a, CGTRINH b, THAMGIA c
Where      a.HOTEN_CN = c.HOTEN_CN And  b.STT_CTR = c.STT_CTR
          And  ( CH_MON = 'Han' or CH_MON = 'Dien' )
          And  TEN_THAU = 'Le Van Son'

```

e. Những công nhân nào đã bắt đầu tham gia công trình Khách sạn Quốc Tế ở Cần Thơ trong giai đoạn từ ngày 15/12/94 đến ngày 31/12/94, số ngày tương ứng là bao nhiêu.

Phân tích câu hỏi:

Yêu cầu cho biết: HOTEN_CN, NGAY_TGIA, SO_NGAY => Ta lấy từ bảng: THAMGIA

(Ta không cần lấy từ bảng CONGNHAN vì không cần các thông tin khác của công nhân)

Điều kiện: NGÀY_TGIA thuộc '12/15/94' -> '12/31/94' => Đặt ĐK trên THAMGIA
 TEN_CTR = 'KS Quoc Te' } => Đặt ĐK trên bảng: CGTRINH
 Và TINH_THANH = 'Can Tho'

Như vậy, ta sẽ truy xuất 2 bảng: THAMGIA và CGTRINH.

Lưu ý: Cần lưu ý về cách biểu diễn ngày tháng mà hệ thống đang sử dụng. Chẳng hạn 'mm/dd/yy'. An toàn nhất là biểu diễn tháng bằng tên; vd: '15 Dec, 1994' hoặc 'Dec 15, 1994'

Câu truy vấn:

```
Select      HOTEN_CN, NGÀY_TGIA, SO_NGAY
From        CGTRINH as b, THAMGIA as c
Where       b.STT_CTR = c.STT_CTR
And         NGÀY_TGIA between 'Dec 15, 1994' And 'Dec 31, 1994'
And         TEN_CTR = 'KS Quoc Te'
And         TINH_THANH = 'Can Tho'
```

f. Cho biết tên và địa chỉ của công trình mà công nhân Nguyễn Hồng Vân đang tham gia vào ngày 18/12/94.

Phân tích câu hỏi:

Yêu cầu cho biết: TEN_CTR, DCHI_CTR => Ta lấy từ bảng: CGTRINH
 Điều kiện: Đang tham gia vào ngày 18/12/94 có nghĩa là: ngày 18/12/94 phải nằm trong khoảng từ ngày bắt đầu và ngày kết thúc tham gia (ngày kết thúc tức là ngày bắt đầu tham gia + số ngày tham gia).

=> NGÀY_TGIA <= '12/18/94'
 NGÀY_TGIA + SO_NGAY >= '12/18/94'
 HOTEN_CN = 'Nguyen Hong Van' } => Đặt ĐK trên THAMGIA

Lưu ý: Để cộng ngày tháng trong SQL Server, ta dùng hàm DATEADD. Tham khảo tại phần giới thiệu các hàm xử lý ngày tháng ở trang 10

Câu truy vấn:

```
Select TEN_CTR, DCHI_CTR      From CGTRINH as b, THAMGIA as c
Where b.STT_CTR = c.STT_CTR    And NGÀY_TGIA <= 'Dec 18, 1994'
And DATEADD(dd, SO_NGAY, NGÀY_TGIA) >= 'Dec 18, 1994'
And HOTEN_CN = 'Nguyen Hong Van'
```

g. Cho biết họ tên và năm sinh của các kiến trúc sư đã tốt nghiệp ở thành phố Hồ Chí Minh và đã thiết kế ít nhất một công trình có kinh phí đầu tư trên 400 triệu đồng.

Phân tích câu hỏi:

Yêu cầu cho biết: HOTEN_KTS, NAMS_KTS => Lấy từ bảng: KTRUCSU
 Điều kiện: NOI_TN = 'TP HCM' => Đặt ĐK trên bảng: KTRUCSU
 Và KINH_PHI > 400 => Đặt ĐK trên bảng: CGTRINH

Như vậy, ta sẽ truy xuất 2 bảng: KTRUCSU và CGTRINH. Tương tự câu c, vì hai bảng này không liên thông nhau nên ta phải thêm vào bảng THIETKE trong truy vấn.

Lưu ý: - Vì đơn vị tính của kinh phí trong CSDL là triệu đồng nên 400 triệu sẽ là 400.
 - Từ 'ít nhất một' trong câu truy vấn có thể bỏ qua vì nếu một KTS không có công trình nào thỏa điều kiện thì khi đặt ĐK chọn và kết nối, KTS đó sẽ tự động bị loại bỏ. Quy luật này áp dụng đúng cho mọi câu truy vấn.

Câu truy vấn:

```
Select      HOTEN_KTS, NAMS_KTS
From        KTRUCSU a, CGTRINH b, THIETKE c
Where       a.HOTEN_KTS = c.HOTEN_KTS
And         b.STT_CTR = c.STT_CTR;
And         NOI_TN = 'TP HCM'
And         KINH_PHI > 400
```

IV. BÀI TẬP TỰ LÀM:

1. CSDL: Quản lý báo chí (CSDL2)

BAIBAO(STT_BAI, TUA, MA_THLOAI)

Mỗi bài báo được đánh một số thứ tự phân biệt, có một tựa bài và cho biết nó thuộc thể loại nào.

BAO_TCHI(MA_BAO_TC, TEN, DCHI_TS, TEL, DINH_KY)

Mỗi tờ báo hoặc tạp chí có một mã riêng biệt, có một tên, một địa chỉ toà soạn, một số điện thoại, và được xuất bản theo một định kỳ nào đó.

THE_LOAI(MA, DGAIAI)

Mỗi thể loại bài báo có một mã thể loại và một diễn giải.

DANG_BAI(STT_BAI, MA_BAO_TC, SO_BAO_TC, TRANG)

Mỗi bài báo khi được đăng trong một báo hoặc tạp chí nào đó, thì được ghi nhận là đã đăng trong số báo hoặc số tạp chí nào và đăng tại trang số mấy.

PHATHANH(MA_BAO_TC, SO_BAO_TC, NGÀY)

Mỗi số báo hoặc số tạp chí khi được phát hành sẽ được ghi nhận ngày phát hành.

VIETBAI(STT_BAI, BUT_HIEU)

Mỗi bài báo được viết bởi một tác giả mà ta ghi nhận bằng bút hiệu.

2. Yêu cầu 1: Tìm sự thông thương giữa các bảng trong CSDL2

3. Yêu cầu 2: Mở các bảng dữ liệu để xem kiểu dữ liệu của từng trường và quan sát dữ liệu của từng bảng.

4. Yêu cầu 3: Hãy viết câu lệnh SQL để trả lời các câu hỏi sau:

a. **Tìm tên các báo / tạp chí có đăng bài của tác giả Tô Phan. Cho biết đăng trên số nào? Được phát hành vào những ngày nào?**

b. **Nội dung của tạp chí Kiến thức ngày nay số 156 gồm những bài tựa gì? Đăng từ trang nào?**

c. **Cho biết tựa và thể loại tất cả các bài báo do tác giả Nguyễn Trọng Vinh sáng tác và đã được đăng trên các báo / tạp chí xuất bản hàng tuần. Cho biết đồng thời tên và số của tờ báo / tạp chí tương ứng mỗi bài được đăng.**

d. **Cho biết các số báo / tạp chí và các ngày phát hành tương ứng trong năm 1994 của tờ báo Lao động.**

e. **Tờ Tuổi trẻ chủ nhật số 23 có những bài bút ký nào? (Cho biết tựa của chúng)**

f. **Cho biết ngày đăng, tên và số của tờ báo / tạp chí đã đăng bài “Chúc vui năm Lợn”.**

g. **Cho biết tên, địa chỉ và số điện thoại của tờ báo tạp chí đã đăng bài “Tìm hiểu chất độc sarin” trong tuần lễ từ 23/04/95 đến 30/04/1995.**

h. **Tác giả Hàn Ngọc Cẩm có những truyện nào đã được đăng trên báo / tạp chí? Cho biết tựa của chúng.**

i. **Cho biết số lượng bài báo mà tạp chí Kiến thức ngày nay đã đăng trên tất cả các số báo.**

j. **Cho biết thể loại của các bài báo do các tác giả có họ Lê sáng tác.**

k. **Tìm tựa các bài thơ đã được đăng trên các báo/tạp chí trong tuần lễ từ 23/4/95 đến 30/4/95.**

l. Tìm tên các tác giả được đăng bài trên báo Cựu Chiến Binh.

m. Tìm tên những báo/tạp chí có đang xã luận. In ra tên các bài xã luận đó và các số báo/tạp chí tương ứng.

BÀI 2

I. MỤC ĐÍCH

Luyện tập câu lệnh SELECT lồng nhau để trả lời cho các câu hỏi cần sử dụng phép giao hoặc phép trừ. Ngoài ra, còn luyện tập khả năng vận dụng các hàm kết tập max, min, sum, count, avg.

II. LÝ THUYẾT

Câu lệnh SQL sử dụng trong bài này:

SELECT <Tên các cột>

FROM <Tên bảng>

WHERE <điều kiện chứa select con>

(**SELECT** <Tên cột>

FROM <Tên bảng> [**WHERE** <Điều kiện>])

[**AND** | **OR** <điều kiện chứa select con>

(**SELECT** <Tên cột>

FROM <Tên bảng> [**WHERE** <Điều kiện>])

[...]]

*** Điều kiện chứa select con có các dạng sau đây:**

<Tên cột> <so sánh> (<select con>):

ĐK đúng khi giá trị của cột so sánh đúng với giá trị trả về từ select con.

<Tên cột> <so sánh> **ALL** (<select con>):

ĐK đúng khi giá trị của cột so sánh đúng với tất cả các giá trị trả về từ select con.

<Tên cột> <so sánh> **ANY|SOME** (<select con>):

ĐK đúng khi giá trị của cột so sánh đúng với bất kỳ một giá trị nào trả về từ select con.

<Tên cột> [**NOT**] **IN** (<select con>):

ĐK đúng khi giá trị của cột nằm trong tập hợp các giá trị trả về của select con.

[**NOT**] **EXISTS** (<select con>):

ĐK đúng khi kết quả trả về của select con khác rỗng.

Khi sử dụng các dạng này, câu lệnh select con cần phải tuân theo quy tắc sau:

Select con trả về:	Một cột	Nhiều cột
Một dòng	Dạng 1	Dạng 5
Nhiều dòng	Dạng 2, 3, 4	

Lưu ý: - Câu lệnh *SELECT* trong *SQL Server 2000* cho phép lồng đến tối đa là 32 mức.
- Mệnh đề *HAVING* cũng có thể chứa *SELECT* con như mệnh đề *WHERE*.

III. BÀI TẬP CÓ HƯỚNG DẪN:

1. CSDL: Quản lý công trình (CSDL1)

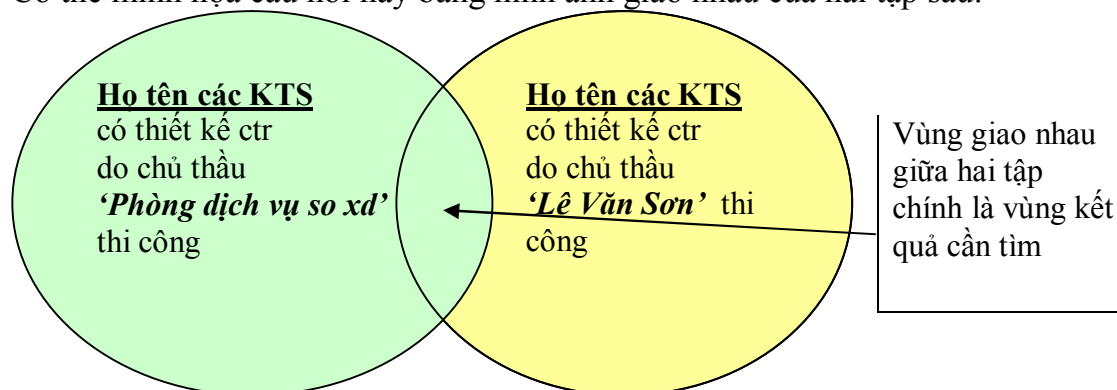
KTRUCSU(HOTEN_KTS, NAMS_KTS, PHAI, NOI_TN, DCHI_LL_KT)
 CHUTHAU(TEN_THAU, TEL, DCHI_THAU)
 CHUNHAN(TEN_CHU, DCHI_CHU)
 CONGNHAN(HOTEN_CN, NAMS_CN, NAM_VAO_N, CH_MON)
 CGTRINH(STT_CTR, TEN_CTR, DCHI_CTR, TINH_THANH, KINH_PHI,
 TEN_CHU, TEN_THAU, NGAY_BD)
 THAMGIA(HOTEN_CN, STT_CTR, NGAY_TGIA, SO_NGAY)
 THIETKE(HOTEN_KTS, STT_CTR, THU_LAO)
 Tham khảo trang 21 để xem diễn giải cho từng bảng.

2. Yêu cầu 1: Hãy viết câu lệnh SQL để trả lời các câu hỏi sau:

- a. Cho biết họ tên các kiến trúc sư *vừa* thiết kế các công trình do Phòng Dịch vụ Sở Xây dựng thi công, *vừa* thiết kế các công trình do chủ thầu Lê Văn Sơn thi công.

Phân tích câu hỏi:

Có thể minh họa câu hỏi này bằng hình ảnh giao nhau của hai tập sau:



Vậy ta phải tìm hai tập hợp này bằng hai câu lệnh SELECT và giao chúng lại với nhau. Tuy nhiên, vì phép giao không được hỗ trợ trực tiếp nên ta sử dụng dạng câu SELECT lồng nhau trong đó sử dụng từ khóa IN để kết nối select cha và select con với nhau (tức là dạng điều kiện chứa select con số 4 đã đề cập ở trên).

Lưu ý: Cách nhận biết dạng câu hỏi sử dụng phép GIAO để trả lời:

*Yêu cầu tìm một <đối tượng nào đó> **vừa** thực hiện <một hành động> này hay sở hữu một <tính chất> này, lại **vừa** thực hiện <một hành động> khác hay sở hữu một <tính chất> khác*

Câu truy vấn:

```
Select distinct HOTEN_KTS From THIETKE a, CGTRINH b
Where a.STT_CTR = b.STT_CTR
and TEN_THAU = 'phong dịch vụ so xd'
and HOTEN_KTS IN
(Select HOTEN_KTS From THIETKE a, CGTRINH b
Where a.STT_CTR = b.STT_CTR
and TEN_THAU = 'le van son')
```

Lưu ý: Trong câu SELECT con dạng này (dùng từ khóa IN) ta chỉ được select một cột duy nhất (không được select nhiều cột), và cột select phải là cột ta đặt trong điều kiện chứa select con. Trong ví dụ trên đó là cột hoten_kts.

Thắc mắc: Select cha và select con của câu truy vấn trên có thể đổi chỗ cho nhau được không?

b. Cho biết tên công trình có kinh phí cao nhất.

Phân tích câu hỏi:

Vì ta chưa biết kinh phí cao nhất là bao nhiêu nên ta phải dùng một select con kết hợp với hàm max để tìm ra con số này. Vì chỉ có duy nhất một con số lớn nhất, nên ta có thể sử dụng dạng điều kiện chứa select con số 1.

Câu truy vấn:

```
Select TEN_CTR From CGTRINH
Where KINH_PHI =
  (Select max(KINH_PHI) From CGTRINH)
```

Lưu ý: Đối với các câu hỏi dạng câu hỏi:

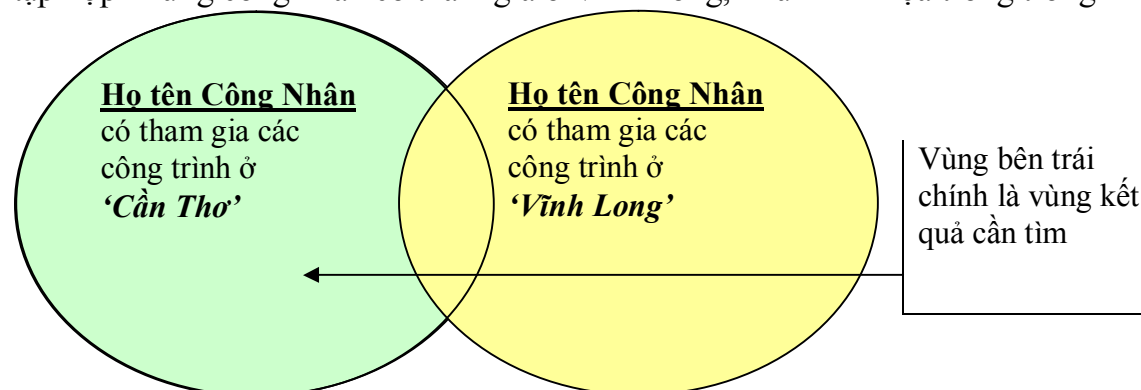
Tìm <đối tượng> sở hữu một <thông số> < lớn / nhỏ / nhiều / ...>. nhất.

Ta đều phải sử dụng select con trong câu truy vấn.

c. Cho biết họ tên các công nhân có tham gia các công trình ở Cần Thơ, nhưng không tham gia công trình ở Vĩnh Long.

Phân tích câu hỏi:

Như ta biết CSDL của ta chỉ lưu trữ những thông tin có hay đã xảy ra. Vì vậy để trả lời được câu hỏi này, ta chỉ có thể lấy tập hợp những công nhân có tham gia ở Cần Thơ trừ đi tập hợp những công nhân có tham gia ở Vĩnh Long, như minh họa trong hình sau:



Vậy trước hết ta phải tìm hai tập hợp này bằng hai câu lệnh SELECT và sau đó trừ tập thứ nhất cho tập thứ hai. Tuy nhiên, vì phép trừ không được hỗ trợ trực tiếp nên ta sử dụng dạng câu SELECT lồng nhau trong đó sử dụng từ khóa NOT IN để kết nối select cha và select con với nhau (tức là dạng điều kiện chứa select con số 4 đã đề cập ở trên).

Lưu ý: Cách nhận biết dạng câu hỏi sử dụng phép TRỪ để trả lời:

*Yêu cầu tìm một <đối tượng nào đó> có thực hiện <một hành động> này hay sở hữu một <tính chất> này, **nhưng không** (hoặc **chưa**) thực hiện <một hành động> khác hay sở hữu một <tính chất> khác*

Câu truy vấn:

```
Select distinct      HOTEN_CN      From THAMGIA a, CGTRINH b
Where a.STT_CTR = b.STT_CTR
and TINH_THANH = 'Can Tho'
and HOTEN_CN NOT IN
  (Select HOTEN_CN      From THAMGIA a, CGTRINH b
   Where a.STT_CTR = b.STT_CTR
    and TINH_THANH = 'Vinh Long')
```

Thắc mắc: *Select cha và select con của câu truy vấn trên có thể đổi chỗ cho nhau được không?*

d. Cho biết tên của các chủ thầu đã thi công các công trình có kinh phí lớn hơn tất cả các công trình do chủ thầu Phòng Dịch vụ Sở Xây dựng thi công.

Phân tích câu hỏi:

Trước hết ta phải dùng một select để tìm tất cả các kinh phí của các công trình do chủ thầu Phòng dịch vụ sở xây dựng thi công. Sau đó một select thứ 2 để tìm các chủ thầu thi công các công trình lớn hơn tất cả các kinh phí này. Ta thấy hai câu select này có thể lồng nhau select thứ nhất sẽ là select con vì nó được thực hiện trước, còn select thứ 2 sẽ là select cha, và select cha sẽ dùng điều kiện chứa select con dạng thứ 2 (có từ khóa ALL).

Câu truy vấn:

```
Select TEN_THAU From CGTRINH
Where KINH_PHI > ALL
      (Select KINH_PHI From CGTRINH
       Where TEN_THAU = 'Phòng dịch vụ so xd')
```

Thắc mắc: *Câu này ta có thể viết lại bằng cách sử dụng dạng điều kiện chứa select con số 1 được không?*

e. Cho biết họ tên các kiến trúc sư có thù lao thiết kế cho một kiến trúc sư của một công trình nào đó dưới giá trị trung bình thù lao thiết kế của một kiến trúc sư cho một công trình.

Phân tích câu hỏi:

Trước hết ta phải dùng một select để tìm giá trị trung bình thù lao thiết kế của một công trình. Sau đó dùng một select thứ 2 để tìm các kiến trúc sư có thù lao thiết kế một công trình nào đó nhỏ hơn con số tìm được. Ta thấy hai câu select này có thể lồng nhau: select thứ nhất sẽ là select con vì nó được thực hiện trước, còn select thứ 2 sẽ là select cha, và select cha sẽ dùng điều kiện chứa select con dạng thứ 1.

Câu truy vấn:

```
Select HOTEN_KTS From THIETKE
Where THU_LAO <
      (Select Avg(THU_LAO) From THIETKE)
```

f. Tìm tên và địa chỉ những thầu đã trúng thầu công trình có kinh phí thấp nhất.

Phân tích câu hỏi:

Vì ta chưa biết kinh phí thấp nhất là bao nhiêu nên ta phải dùng một select con kết hợp với hàm min để tìm ra con số này. Vì chỉ có duy nhất một con số nhỏ nhất, nên ta có thể sử dụng dạng điều kiện chứa select con số 1.

Câu truy vấn:

```
Select TEN_THAU, DCHI_THAU From CHU_THAU a, CGTRINH b
Where a.TEN_THAU = b.TEN_THAU And KINH_PHI =
      (Select min(KINH_PHI) From CGTRINH)
```

Thắc mắc: - Dạng câu hỏi này giống câu hỏi nào trong bài thực hành này?
- Câu này có thể sử dụng dạng điều kiện chứa select con số 3 hay không?

g. Tìm họ tên và chuyên môn của các công nhân tham gia các công trình do kiến trúc sư 'Le Thanh Tung' thiết kế.

Phân tích câu hỏi:

Trước hết ta phải dùng một select để tìm số thứ tự các công trình do kiến trúc sư Lê Thanh Tùng thiết kế. Sau đó dùng một select thứ 2 để tìm các công nhân đã tham gia vào các công trình tìm được. Ta thấy hai câu select này có thể lồng nhau: select thứ nhất sẽ là select con vì nó được thực hiện trước, còn select thứ 2 sẽ là select cha, và select cha sẽ dùng điều kiện chứa select con dạng thứ 4.

Câu truy vấn:

```
Select      a.HOTEN_CN, CH_MON
From        CONGNHAN a, THAMGIA b
Where       a.HOTEN_CN = b.HOTEN_CN
            and b.STT_CTR IN
            (Select STT_CTR From THIETKE
             Where HOTEN_KTS = 'Le Thanh Tung')
```

h. Tìm các cặp tên của chủ thầu có trúng thầu các công trình tại cùng một thành phố.

Phân tích câu hỏi:

Vì mỗi dòng trong bảng CGTRINH chỉ cho biết chủ thầu nào thi công công trình nào, nên để tìm được những cặp chủ thầu trúng thầu các công trình tại cùng thành phố, thì ta phải mở bảng CGTRINH hai lần với hai bí danh khác nhau, kết nối chúng bằng các điều kiện khác tên thầu nhưng cùng tỉnh thành:

```
a.TEN_THAU <> b.TEN_THAU      (1)
a.TINH_THANH = b.TINH_THANH
```

Tuy nhiên, với điều kiện kết nối như vậy, ta sẽ nhận được hai dòng cho mỗi cặp với thứ tự tên thầu ngược nhau; chẳng hạn: (Le Van Son, Cty xd so 6) và (Cty xd so 6, Le Van Son). Để khắc phục tình trạng này, ta sửa lại điều kiện kết nối (1) bằng:

```
a.TEN_THAU < b.TEN_THAU
```

Câu truy vấn:

```
Select      Distinct c1.TEN_THAU, c2.TEN_THAU, c1.TINH_THANH
From        CGTRINH c1, CGTRINH c2
Where       c1.TEN_THAU < c2.TEN_THAU
            and c1.TINH_THANH = c2.TINH_THANH
```

i. Tìm các cặp tên của các công nhân có làm việc chung với nhau trong ít nhất là hai công trình.

Phân tích câu hỏi:

Trước hết ta phải dùng một select để tìm các cặp họ tên công nhân có tham gia cùng công trình. Sau đó dùng một select thứ hai nhóm từng cặp lại và đếm số công trình họ tham gia chung với nhau, chọn ra những cặp có số công trình tham gia chung lớn hơn hoặc bằng 2. Hai select này không thể lồng nhau, vì select thứ hai nhóm dữ liệu trên kết quả của select thứ nhất. Vì vậy bắt buộc ta phải dùng 2 select. Select thứ nhất sẽ đưa kết quả vào bảng #TAM bằng mệnh đề INTO <Tên Bảng>

Lưu ý: Các bảng có tên bắt đầu bằng dấu # sẽ được lưu vào CSDL tempdb. Đây là CSDL nháp, vì vậy, sau khi ta đóng kết nối với SQL Server thì các bảng này sẽ bị xóa.

Câu select thứ nhất hoàn toàn tương tự như câu h ở trên. Vì mỗi dòng trong bảng THAMGIA chỉ cho biết công nhân nào tham gia công trình nào, nên để tìm được những cặp công nhân tham gia cùng công trình, thì ta phải mở bảng THAMGIA hai lần với hai bí danh khác nhau, kết nối chúng bằng các điều kiện:

```
a.HOTEN_CN <> b.HOTEN_CN      (1)
a.STT_CTR = b.STT_CTR
```


Tuy nhiên, với điều kiện kết nối như vậy, ta sẽ nhận được hai dòng cho mỗi cặp với thứ tự họ tên ngược nhau; chẳng hạn: (Nguyễn Văn A, Trần Thị B) và (Trần Thị B, Nguyễn Văn A). Để khắc phục tình trạng này, ta sửa lại điều kiện kết nối (1) bằng:

a.HOTEN_CN < b.HOTEN_CN

Câu truy vấn:

```
Select t1.HOTEN_CN As cn1, t2.HOTEN_CN As cn2, t1.STT_CTR
Into      #CAU_2i
From THAMGIA t1, THAMGIA t2
Where t1.HOTEN_CN < t2.HOTEN_CN
      and t1.STT_CTR = t2.STT_CTR
Order by cn1, cn2
```

Lưu ý: Mặc dù hai cột lấy từ hai bảng với bí danh khác nhau nhưng có tên giống nhau, nên ta phải đặt lại tên cột bằng từ khóa AS thì mới có thể lưu lại vào bảng mới.

```
Select cn1, cn2      From #CAU_2i
Group by cn1, cn2
Having count(STT_CTR) >=2
```

IV. BÀI TẬP TỰ LÀM:

1. CSDL: Quản lý mua bán hàng hóa (CSDL3)

HANGHOA(MA_HANG, TEN_HG)

Mỗi hàng hóa được gán một mã hàng riêng biệt để dễ phân biệt, và có một tên hàng.

DAILY(STT_DL, TEN_DL, DCHI_DL)

Mỗi đại lý được gán một số thứ tự phân, có một tên đại lý và một địa chỉ liên lạc.

MUA(MA_HANG, STT_DL, NGÀY_MUA, SOLG_MUA, TRIGIA_MUA)

Khi một đại lý mua một mặt hàng nào đó (MA_HANG), vào một ngày nào đó (NGÀY_MUA), sẽ được ghi nhận lại số lượng mua và trị giá mua tổng cộng.

BAN(MA_HANG, STT_DL, NGÀY_BAN, SOLG_BAN, TRIGIA_BAN)

Khi một đại lý bán một mặt hàng nào đó (MA_HANG), vào một ngày nào đó (NGÀY_BAN), sẽ được ghi nhận lại số lượng bán và trị giá bán tổng cộng.

2. Yêu cầu 1: Tìm sự thông thương giữa các bảng trong CSDL3

3. Yêu cầu 2: Mở các bảng dữ liệu để xem kiểu dữ liệu của từng trường và quan sát dữ liệu của từng bảng.

4. Yêu cầu 3: Hãy viết câu lệnh SQL để trả lời các câu hỏi sau:

- a. Tìm tên những đại lý vừa có bán coca cola vừa có bán pepsi cola.**
- b. Tìm tên những mặt hàng được mua nhưng chưa được bán.**
- c. Tìm tên và địa chỉ những đại lý có mua ít nhất cùng một mặt hàng với Vạn Lợi mua.**
- d. Cho biết tên các mặt hàng được ít nhất là hai đại lý mua vào với số lượng trên 50.**
- e. Tìm tổng trị giá mua Coca Cola ở đại lý Tân Hiệp Hưng.**
- f. Tìm tổng trị giá bán Coca Cola ở đại lý Tân Hiệp Hưng.**
- g. Tìm tên những mặt hàng được bán ở đại lý Tân Hiệp Hưng vào cả hai ngày 15/12/94 và ngày 31/12/94.**
- h. Có bao nhiêu mặt hàng được mua vào nhưng chưa được bán ra ở đại lý Tân Hiệp Hưng trong tháng 12 năm 1994.**
- i. Tìm tên các mặt hàng có mua và bán trong cùng một ngày ở cùng một đại lý.**
- j. Tìm tên và địa chỉ của các đại lý và những mặt hàng có số lượng mua và bán bằng nhau trong cùng một ngày.**
- k. Tìm tên đại lý đã mua coca cola với số lượng nhiều hơn tất cả các đại lý khác.**

BÀI 3**I. MỤC ĐÍCH:**

Luyện tập câu lệnh SELECT có GROUP BY, HAVING có kết hợp với các dạng SELECT đã thực tập ở các bài trước.

II. LÝ THUYẾT:

Câu lệnh SQL sử dụng trong bài này:

```
SELECT <các cột phân nhóm>, <hàm-kết-tập(<biểu thức>)>
FROM <bảng>
[ WHERE <Điều kiện chọn dòng> ]
[ GROUP BY <Cột để phân nhóm> [, <Cột để phân nhóm> [...]] ]
[ HAVING <điều kiện chọn nhóm> ]
```

III. BÀI TẬP CÓ HƯỚNG DẪN**1. CSDL: Quản lý Công Trình (CSDL 1)**

1. KTRUCSU(HOTEN_KTS, NAMS_KTS, PHAI, NOI_TN, DCHI_LL_KT)
 2. CHUTHAU(TEN_THAU, TEL, DCHI_THAU)
 3. CHUNHAN(TEN_CHU, DCHI_CHU)
 4. CONGNHAN(HOTEN_CN, NAMS_CN, NAM_VAO_N, CH_MON)
 5. CGTRINH(STT_CTR, TEN_CTR, DCHI_CTR, TINH_THANH, KINH_PHI, TEN_CHU, TEN_THAU, NGÀY_BD)
 6. THAMGIA(HOTEN_CN, STT_CTR, NGÀY_TGIA, SO_NGAY)
 7. THIETKE(HOTEN_KTS, STT_CTR, THU_LAO)
- Tham khảo trang 21 để xem diễn giải cho từng bảng.

2. Yêu cầu 1: Hãy viết câu lệnh SQL để trả lời các câu hỏi sau:

- a. Tìm tổng kinh phí của tất cả các công trình theo từng chủ thầu.

Phân tích câu hỏi:

Vì mỗi dòng dữ liệu trong bảng CGTRINH chỉ cho biết kinh phí của từng công trình, nên muốn biết tổng kinh phí theo từng chủ thầu thì ta phải nhóm các dòng dữ liệu theo cột TEN_THAU, sau đó dùng hàm sum để cộng kinh phí cho mỗi nhóm.

Câu truy vấn:

```
Select    TEN_THAU, sum(KINH_PHI) As TongKP
From      CGTRINH
Group by  TEN_THAU
```

Thắc mắc: Nếu trong mệnh đề Select ta thêm cột TEN_CTR thì có được không? Tại sao?

- b. Cho biết họ tên các kiến trúc sư có tổng thù lao thiết kế các công trình lớn hơn 25 triệu.

Phân tích câu hỏi:

Vì mỗi dòng dữ liệu trong bảng THIETKE chỉ cho biết thù lao thiết kế của một kiến trúc sư cho một công trình nào đó, nên muốn biết tổng thù lao thiết kế các công trình của từng kiến trúc sư, thì ta phải nhóm các dòng dữ liệu theo cột HOTEN_KTS, sau đó dùng hàm sum để cộng thù lao cho mỗi nhóm. Sau cùng, dùng mệnh đề HAVING để chọn lại các nhóm (hay các kiến trúc sư) có tổng thù lao > 25.

Câu truy vấn:

```
Select    HOTEN_KTS, sum(THU_LAO) As TongTL
```

```

From      THIETKE
Group by  HOTEN_KTS
Having    sum(THU_LAO) > 25

```

c. Cho biết số lượng các kiến trúc sư có tổng thù lao thiết kế các công trình lớn hơn 25 triệu.

Phân tích câu hỏi:

Câu hỏi này tương tự như câu b, nhưng thay vì liệt kê họ tên các kiến trúc sư thỏa điều kiện, thì ta chỉ cần cho biết có tổng cộng bao nhiêu kiến trúc sư thỏa điều kiện như vậy. Vì vậy, trước hết ta phải tìm danh sách các kiến trúc sư thỏa điều kiện trước rồi mới đếm trên danh sách tìm được. Hay nói khác hơn, ta phải thực hiện 2 bước bằng 2 lệnh select khác nhau:

Câu 1. giống hệt câu b, nhưng kết quả xuất ra sẽ được lưu vào bảng #CAU_3C

Câu 2. Đếm số dòng trên tập tin #CAU_3C

Câu Truy Vấn

```

Select    HOTEN_KTS, sum(THU_LAO) As TongTL    Into    #CAU_3C
From      THIETKE
Group by  HOTEN_KTS
Having    sum(THU_LAO) > 25

```

```

Select    count(*) As Solg_KTS
From      #CAU_3C

```

Thắc mắc: Ở bước 2 ta có cần phải đếm phân biệt không?

d. Tìm tổng số công nhân đã tham gia của mỗi công trình.

Phân tích câu hỏi:

Vì mỗi dòng dữ liệu trong bảng THAMGIA chỉ cho công nhân nào tham gia công trình nào, nên muốn biết tổng số công nhân đã tham gia theo từng công trình, thì ta phải nhóm các dòng dữ liệu theo cột STT_CTR, sau đó dùng hàm count để đếm số dòng (mỗi dòng là một công nhân) trong mỗi nhóm để có được tổng số công nhân tham gia.

Câu truy vấn:

```

Select    STT_CTR, count(*) As TongSoCN
From      THAMGIA
Group by  STT_CTR

```

Thắc mắc: Trong mệnh đề Select của câu này có thể thêm cột HOTEN_CN không?

e. Tìm tên và địa chỉ công trình có tổng số công nhân tham gia nhiều nhất.

Phân tích câu hỏi:

Câu hỏi này tương tự như câu d, nhưng thay vì cho biết tổng số công nhân tham gia của tất cả các công trình, thì ở đây ta chỉ yêu cầu kết xuất ra công trình nào có tổng số công nhân tham gia nhiều nhất.

Vì vậy, trước hết ta phải thực hiện câu d trước đưa kết quả vào bảng #CAU_3E, từ bảng #CAU_3E ta mới có thể chọn ra công trình có tổng số công nhân tham gia nhiều nhất được. Vậy, ta phải thực hiện 2 bước bằng 2 lệnh select khác nhau:

Câu 1. giống hệt câu d, nhưng kết quả xuất ra sẽ được lưu vào bảng #CAU_3E

Câu 2. Chọn ra dòng có TongSoCN cao nhất từ tập tin #CAU_3E, đồng thời kết nối với bảng CGTRINH để lấy thông tin về tên và địa chỉ công trình.

Câu Truy Vấn:

```

Select    STT_CTR, count(*) As TongSoCN    Into    #CAU_3E
From      THAMGIA

```

```

Group by STT_CTR

Select  TEN_CTR, DIACHI_CTR, TongSoCN
From    #CAU_3E a, CGTRINH b
Where   a.STT_CTR = b.STT_CTR
        And    TongSoCN =
              (Select  Max(TongSoCN)
               From    #CAU_3E)

```

f. Cho biết tên các thành phố và kinh phí trung bình cho mỗi công trình của từng thành phố tương ứng.

Phân tích câu hỏi:

Vì mỗi dòng dữ liệu trong bảng CGTRINH chỉ cho biết kinh phí của mỗi công trình và cho biết công trình đó nào thuộc tỉnh thành nào, nên muốn biết kinh phí trung bình cho mỗi công trình của từng thành phố, thì ta phải nhóm các dòng dữ liệu theo cột TINH_THANH, sau đó dùng hàm avg để tính kinh phí trung bình cho mỗi nhóm (tức là mỗi thành phố).

Câu truy vấn:

```

Select    TINH_THANH, Avg(KINH_PHI) As KinhPhiTB
From      CGTRINH      Group By    TINH_THANH

```

g. Cho biết họ tên các công nhân có tổng số ngày tham gia vào các công trình lớn hơn tổng số ngày tham gia của công nhân Nguyen Hong Van.

Phân tích câu hỏi:

Để trả lời được câu hỏi này trước hết ta phải biết tổng số ngày tham gia các công trình của mỗi công nhân. Sau đó mới chọn ra những công nhân có tổng số ngày này lớn hơn tổng số ngày của công nhân Nguyễn Hồng Vân. Vì vậy, ta phải thực hiện hai bước này bằng 2 lệnh Select khác nhau:

Câu 1. Nhóm dữ liệu trong bảng THAMGIA theo HOTEN_CN, và tính tổng số ngày tham gia. Lưu kết quả vào bảng #CAU_3G

Câu 2. Chọn các dòng trên bảng #CAU_3G thỏa điều kiện.

Câu truy vấn:

```

Select    HOTEN_CN, Sum(SO_NGAY) As TongSoNgay      Into    #CAU_3G
From      THAMGIA
Group By  HOTEN_CN

Select    HOTEN_CN, TongSoNgay      From    #CAU_3G
Where     TongSoNgay >
          (Select TongSoNgay      From    #CAU_3G
           Where   HOTEN_CN = 'Nguyen Hong Van')

```

h. Cho biết tổng số công trình mà mỗi chủ thầu đã thi công tại mỗi thành phố.

Phân tích câu hỏi:

Vì mỗi dòng dữ liệu trong bảng CGTRINH chỉ cho công trình nào thuộc tỉnh thành nào và do chủ thầu nào thi công, nên muốn biết tổng số công trình mà mỗi chủ thầu đã thi công tại mỗi tỉnh thành, thì ta phải nhóm các dòng dữ liệu theo 2 cột TEN_THAU và TINH_THANH, sau đó dùng hàm count để đếm số dòng chính là số công trình của mỗi nhóm.

Câu Truy Vấn:

```

Select    TEN_THAU,    TINH_THANH, Count (STT_CTR) As    TongSoCTR
From      CGTRINH
Group By  TEN_THAU,    TINH_THANH

```

i. Cho biết họ tên công nhân có tham gia ở tất cả các công trình.

Phân tích câu hỏi:

Vì mỗi dòng dữ liệu trong bảng THAMGIA chỉ cho công nhân nào tham gia công trình nào, nên muốn biết công nhân nào có tham gia hết tất cả các công trình trước hết ta đếm xem mỗi công nhân đã tham gia tổng cộng bao nhiêu công trình, đưa vào bảng #CAU_3i. Sau đó, chọn từ bảng #CAU_3i những công nhân có tổng số công trình tham gia bằng với tổng số công trình ta có, bằng cách đếm số dòng trong bảng CGTRINH.

Câu Truy Vấn:

```
Select  HOTEN_CN, Count (STT_CTR) As TongSoCtrTG
Into    #CAU_3i
From    THAMGIA
Group by HOTEN_CN
```

```
Select  HOTEN_CN      From  #CAU_3i
Where   TongSoCtrTG =
        (Select Count(*) From CGTRINH)
```

IV. BÀI TẬP TỰ LÀM

1. CSDL: Quản lý mua bán hàng hóa (CSDL3)

1. HANGHOA(MA_HANG, TEN_HG)
2. DAILY(STT_DL, TEN_DL, DCHI_DL)
3. MUA(MA_HANG, STT_DL, NGÀY_MUA, SOLG_MUA, TRIGIA_MUA)
4. BAN(MA_HANG, STT_DL, NGÀY_BAN, SOLG_BAN, TRIGIA_BAN)

Tham khảo trang 32 để xem diễn giải cho từng bảng.

2. Yêu cầu 1: Hãy viết câu lệnh SQL để trả lời các câu hỏi sau:

- a. Tìm đơn giá mua trung bình của bia Sài Gòn tính trên tất cả các đại lý.
- b. Tìm đơn giá mua trung bình của bia Sài Gòn tính trên từng đại lý
- c. Hiển thị số lượng tồn kho của từng mặt hàng theo từng đại lý.
- d. Tìm tên, địa chỉ của đại lý có tổng giá trị mua trong một ngày lớn hơn 700000.
- e. Với mỗi đại lý, hãy cho biết ngày nào có số lượng mặt hàng bán ra là ít nhất.
- f. Tìm tên và địa chỉ của những đại lý có tổng trị giá bán cao nhất.
- g. Tìm tổng số tiền mà mỗi đại lý đã chi để mua hàng theo từng tháng.
- h. Tìm tên và tổng số lượng bán ra của các mặt hàng đã bán được với số lượng nhiều nhất của từng đại lý.
- i. Cho biết tổng số món hàng mà mỗi đại lý kinh doanh (mua và bán).
- j. Tìm tên đại lý đã mua vào mặt hàng bia Heineken với đơn giá mua là cao nhất.

BÀI 4**I. MỤC ĐÍCH:**

Luyện tập câu lệnh SELECT có sử dụng các hàm xử lý chuỗi, hàm xử lý ngày tháng, hàm chuyển kiểu, ...; kết hợp với các dạng câu SELECT của các bài trước.

II. BÀI TẬP CÓ HƯỚNG DẪN:**1. CSDL: Quản lý giảng dạy (CSDL5)**

CHUCDANH(MACD, TENCĐ, GIOCHUAN, TIEN_1TIET)

Mỗi chức danh có một mã chức danh phân biệt, một tên chức danh, có một số lượng giờ chuẩn tương ứng, và số tiền một tiết trả theo chức danh đó.

GIAOVIEN(MAGV, HOTEN, PHAI, MACD)

Mỗi giáo viên có một mã giáo viên phân biệt, một họ tên, phái và một mã chức danh tương ứng.

GD_0506(MAGV, MONDAY, LOPDAY, SOTIET, SISO)

Thông tin về việc giảng dạy của giáo viên trong năm học 05-06 bao gồm mã giáo viên, tên môn dạy, tên lớp dạy, số tiết dạy, sĩ số lớp.

TAMUNG(MAGV, NGÀY_TA, SOTIEN)

Mỗi giáo viên có thể tạm ứng tiền. Khi tạm ứng, thông tin về ngày tạm ứng và số tiền tạm ứng sẽ được ghi nhận lại.

2. Yêu cầu 1: Tìm sự thông thương giữa các bảng trong CSDL**3. Yêu cầu 2: Mở các bảng dữ liệu để xem kiểu dữ liệu của từng trường và quan sát dữ liệu của từng bảng****4. Yêu cầu 3: Hãy viết câu lệnh SQL để trả lời các câu hỏi sau:**

- a. In ra danh sách mã giáo viên, họ tên các cán bộ cùng với danh xưng là ‘Thầy’ nếu cán bộ là nam, và ‘Cô’ nếu cán bộ là nữ.

Phân tích câu hỏi:

Ta thấy danh xưng Thầy hay Cô sẽ phụ thuộc vào cột PHAI trong bảng GIAOVIEN.

Ta sẽ dùng cấu trúc CASE đơn giản để kiểm tra cột PHAI và trả về thông tin phù hợp.

Tham khảo cú pháp của CASE ở trang 11.

Câu Truy Vấn:

```

Select MAGV, Case PHAI
              When 0      Then 'Thầy ' + HOTEN
              When 1      Then 'Cô ' + HOTEN
End          As HT
From        GIAOVIEN

```

- b. In ra danh sách gồm 2 cột: Cột ‘Họ giáo viên’ và cột ‘Chữ Lót+Tên giáo viên’

Phân tích câu hỏi:

Cột HOTEN trong bảng GIAOVIEN bao gồm cả họ và tên. Để tách họ và tên ra thành 2 cột thì ta dùng các hàm left để lấy họ, hàm right để lấy ra tên. Để tách họ ra, ta cần biết chiều dài của họ, hay nói khác hơn là vị trí khoảng trắng đầu tiên trong chuỗi HOTEN. Ta sẽ sử dụng hàm CharIndex để tìm vị trí này. Còn tên ta sẽ lấy từ bên phải sang một số lượng kí tự bằng chiều dài họ tên trừ đi chiều dài họ. Tham khảo cú pháp của các hàm này ở trang 10.

Câu Truy Vấn:


```

Select Left(HOTEN, CharIndex(' ', HOTEN)) As Ho_Gv,
       Right (HOTEN, Len(HOTEN) - CharIndex(' ', HOTEN)) As Ten_GV
From GIAOVIEN

```

c. Cho biết danh sách họ tên các giáo viên chỉ dạy cho các lớp điện tử.

Phân tích câu hỏi:

Từ **chỉ** trong câu hỏi cho ta biết đây là dạng câu hỏi sử dụng phép Trừ hai tập hợp. Tập hợp thứ nhất (T1) là tập hợp các giáo viên có dạy cho lớp Điện tử, tập hợp thứ hai (T2) là tập hợp các giáo viên có dạy các lớp ghép Điện tử và Tin Học hoặc các lớp khác (như Lý, Chế biến,...). Lấy T1 trừ cho T2. Có nghĩa là T1 là select cha, còn T2 là select con. Điều kiện chứa select con sử dụng từ khóa NOT IN.

Câu Truy Vấn:

```

Select distinct a.MAGV, HOTEN From GD_0506 a, GIAOVIEN b
Where a.MAGV = b.MAGV
      And LOPDAY Like '%DT%'
      And a.MAGV Not In
          (Select MAGV From GD_0506
           Where LOPDAY Like '%TH%'
            or LOPDAY not like '%DT%')

```

d. Tìm các giáo viên có chức danh là Giảng Viên đã dạy cho lớp DT20. (lưu ý: lớp DT-TH20 là lớp ghép của DT20 và TH20).

Câu Truy Vấn:

```

Select HOTEN, LOPDAY From GIAOVIEN a, GD_0506 b, CHUCDANH c
Where a.MAGV = b.MAGV      And a.MACD = c.MACD
      And TENCĐ = 'Giang Vien'      And LOPDAY Like '%DT%20%'

```

e. Tìm họ tên các giáo viên có tên lót là Văn, và có tạm ứng tiền ít nhất là 2 lần.

Phân tích câu hỏi:

Vì mỗi dòng của bảng tạm ứng chỉ cho biết giáo viên nào tạm ứng ngày nào, với số tiền bao nhiêu thôi, nên muốn biết giáo viên nào tạm ứng ít nhất hai lần thì ta phải nhóm dữ liệu theo cột MAGV, sau đó đếm số dòng của mỗi nhóm và chỉ chọn nhóm có số dòng lớn hơn 1.

Câu Truy Vấn:

```

Select HOTEN, Count(NGAY_TA) As SoLanTA
From GIAOVIEN a, TAMUNG b
Where a.MAGV = b.MAGV      And HOTEN Like '% Van %'
Group By b.MA_GV, HOTEN
Having count(NGAY_TA) > 1

```

f. Cho biết họ tên các giáo viên đã tạm ứng trong tháng 1 năm 2006?

Câu Truy Vấn:

```

Select HOTEN, NGAY_TA
From GIAOVIEN a, TAMUNG b
Where a.MAGV = b.MAGV
      And DatePart(month, NGAY_TA) = 1
      And DatePart(year, NGAY_TA) = 2006

```

g. Cho biết họ tên các giáo viên đã tạm ứng trong 3 tháng gần đây?

Câu Truy Vấn:

```

Select HOTEN, NGAY_TA
From GIAOVIEN a, TAMUNG b
Where a.MAGV = b.MAGV
      And DateDiff(month, NGAY_TA, GETDATE()) <= 3

```

h. Cho biết tổng số tiết (đã qui chuẩn) mà giáo viên có tên Nam đã giảng dạy.

Biết rằng:

+ Số tiết qui chuẩn = SOTIET * HESO;

trong đó HESO được tính như sau:

Nếu số lớp ≤ 80 thì HESO = 1; Nếu 80 < số lớp ≤ 120 thì HESO = 1.2; Nếu 120 < số lớp ≤ 150 thì HESO = 1.4; Ngược lại, số lớp > 150 thì HESO = 1.5

Phân tích câu hỏi:

Ở đây ta phải dùng cấu trúc CASE dạng tìm kiếm để tính số tiết qui chuẩn. Tham khảo trang 11

Câu truy vấn:

```
Select Sum( Case
              When SISO <= 80 Then SOTIET
              When SISO <= 120 Then SOTIET * 1.2
              When SISO <= 150 Then SOTIET * 1.4
              Else SOTIET * 1.5
            END)
From GIAOVIEN a, GD_0506 b
Where a.MAGV = b.MAGV
      And HOTEN Like '% Nam'
```

i. Tìm họ tên các giáo viên có dạy vượt giờ chuẩn

Phân tích câu hỏi:

Tương tự như câu h, ta cũng phải tính số tiết qui chuẩn của từng giáo viên sau đó chọn ra những giáo viên có tổng số tiết qui chuẩn lớn hơn giờ chuẩn theo chức danh qui định.

Câu truy vấn.

-- Tính tổng số số tiết qui chuẩn cho từng giáo viên, đưa vào bảng tạm #CAU_4i

```
Select MAGV, Sum( Case
                  When SISO <= 80 Then SOTIET
                  When SISO <= 120 Then SOTIET * 1.2
                  When SISO <= 150 Then SOTIET * 1.4
                  Else SOTIET * 1.5
                END) As TongSTQC
Into #CAU_4i
From GD_0506 b
Group By MAGV
```

-- Hiển thị họ tên các giáo viên có tổng số tiết dạy vượt giờ chuẩn.

```
Select HOTEN, TongSTQC, GIOCHUAN
From #CAU_4i a, GIAOVIEN b, CHUCDANH c
Where a.MAGV = b.MAGV And b.MACD = c.MACD
      And TongSTQC > GIOCHUAN
```

j. Hãy tính số tiền thực lĩnh của tất cả các cán bộ trong trường.

Biết rằng:

+ Cán bộ chỉ được lãnh tiền giảng dạy khi đã dạy vượt giờ chuẩn (nghĩa là số tiết trong giờ chuẩn không được tính tiền).

+ Số tiền giảng dạy thực tế cho một giáo viên là:

Số tiết vượt giờ qui chuẩn * TIEN_1TIET.

+ Thực lĩnh của cán bộ được tính theo công thức như sau:

Nếu cán bộ có dạy vượt giờ và có tạm ứng thì: Thực lĩnh = số tiền giảng dạy - tổng số tiền tạm ứng. Nếu cán bộ có dạy vượt giờ mà không có tạm ứng thì:

Thực lĩnh = số tiền giảng dạy. Nếu cán bộ không có dạy vượt giờ mà có tạm ứng thì: Thực lĩnh = - tổng số tiền tạm ứng. Nếu cán bộ không có dạy vượt giờ và không có tạm ứng thì: Thực lĩnh = 0.

Phân tích câu hỏi :

Tương tự như câu i, ta cũng tính tổng số giờ dạy chuẩn của mỗi giáo viên, đưa vào bảng tạm #CAU_4j_1.

Tính tổng số tiền tạm ứng của mỗi cán bộ, đưa vào bảng tạm #CAU_4j_2.

Kết nối hai bảng này để tính số tiền thực lĩnh. Tuy nhiên, nếu kết nối tự nhiên bình thường, thì những người có mặt ở một trong hai bảng sẽ biến mất (nghĩa là những người chỉ có dạy hoặc chỉ có tạm ứng), đây là điều ta không muốn, hoặc có những người không có dạy cũng không hề tạm ứng cũng phải xuất ra kết quả. Vì vậy, ở đây ta phải sử dụng kết nối mở rộng.

Lưu ý: SQL Server 2000 hỗ trợ kết nối mở rộng về một phía bằng cách dùng từ khóa left / right outer join.

Câu hỏi truy vấn :

```

Select MAGV, Sum( Case
    When SISO <= 80 Then SOTIET
    When SISO <= 120 Then SOTIET * 1.2
    When SISO <= 150 Then SOTIET * 1.4
    Else SOTIET * 1.5
END) As TongSTQC Into #CAU_4j_1
From GD_0506 b Group By MAGV

Select MAGV, Sum(SOTIEN) As TongTA Into #CAU_4j_2
From TAMUNG Group By MAGV

Select a.MAGV, CASE
    When (TongSTQC > GIOCHUAN) And TongTA is null
        Then TIEN_1TIET * (TongSTQC - GIOCHUAN)
    When (TongSTQC > GIOCHUAN) And TongTA is not null
        Then (TIEN_1TIET * (TongSTQC - GIOCHUAN) ) - TongTA
    When (TongSTQC <= GIOCHUAN) And TongTA is null
        Then 0
    Else -TongTA End
    As ThucLinh
From #CAU_4j_1 a right outer join GIAOVIEN b on
    a.MAGV = b.MAGV join CHUCDANH c on b.MACD = c.MACD
left outer join #CAU_4j_2 d on d.MAGV = a.MAGV

```

III. BÀI TẬP TỰ LÀM

1. CSDL: Quản lý câu lạc bộ (CSDL4)

CLB(MA_CLB, TEN_CLB, DCHI_CLB)

Mỗi câu lạc bộ có một mã câu lạc bộ phân biệt, một tên và một địa chỉ câu lạc bộ.

DOI(MA_DOI, MA_CLB, LT, PHAI)

Trong cùng một câu lạc bộ, mỗi đội có một mã đội phân biệt. Nhưng hai câu lạc bộ có thể có hai đội trùng mã. Mỗi đội thuộc một lứa tuổi, một phái nào đó và thuộc 1 câu lạc bộ.

NHA_TD(MA_NHATD, DCHI_NHATD, TG_SOSAN)

Mỗi nhà thi đấu có một mã nhà thi đấu phân biệt, một địa chỉ và tổng số sân nó có.

TGIAN_TD(LUATUOI, TGIAN_TD)

Thời gian thi đấu trong một trận đấu được qui định theo lứa tuổi của đội thi đấu.

TRANDAU(MA_TD, NGAY_TD, GIO_TD, MA_NHATD)

Mỗi trận đấu có một mã trận đấu phân biệt, thi đấu vào ngày, giờ nào đó và diễn ra tại một nhà thi.

THAMGIA(MA_TD, MA_DOI, MA_CLB)

Khi một đội của một câu lạc bộ tham gia thi đấu trong một trận đấu nào đó. Thi thông tin sẽ được lưu lại trong bảng này.

VDV(MA_VDV, TEN_VDV, DCHI_VDV, PHAI, LUATUOI, MA_DOI, MA_CLB)

Mỗi vận động viên có một mã vận động viên phân biệt, một tên, một địa chỉ, thuộc một phái, một lứa tuổi nào đó, đồng thời phải cho biết vận động viên đó chơi cho đội nào của câu lạc bộ nào.

2. Yêu cầu 1:

- a. Tìm tên tất cả các vận động viên có tham gia thi đấu trong tháng 1 năm 1990.
- b. Cho biết thông tin về các trận đấu gồm: mã trận đấu, địa chỉ nhà thi đấu nơi diễn ra trận đấu, ngày thi đấu, thời điểm bắt đầu và kết thúc trận đấu.
- c. Tìm các cặp mã trận đấu kết thúc vào cùng ngày giờ.
- d. Tìm tên các vận động viên có thi đấu tại nhà thi đấu số 27 trong ngày 900616, từ 10h sáng đến 4h30 chiều.
- e. Tìm địa chỉ nhà thi đấu, nơi mà vận động viên Tuấn đang thi đấu vào 3h chiều ngày 901206.
- f. Tìm danh sách các trận đấu bắt đầu hoặc kết thúc trong khoảng thời gian từ 12h đến 16h diễn ra trên một sân của nhà thi đấu số 49 trong ngày 900805.
- g. Tìm tên các vận động viên có tham gia thi đấu trong các tháng ít ra là như vận động viên Tuấn.
- h. Tìm số lượng trận đấu được tổ chức theo từng năm của từng nhà thi đấu.
- i. Tìm danh sách các câu lạc bộ có đội thi đấu tại nhà thi đấu số 01 vào các ngày trong tháng 1 năm 90.
- j. Tìm danh sách các đội có vận động viên có tên bắt đầu bằng ký tự H, và có thi đấu trong tháng 8 năm 90.

BÀI 5**I. MỤC ĐÍCH:**

Luyện tập câu lệnh tạo CSDL, tạo bảng với một số ràng buộc cơ bản, lệnh xen, sửa, xóa dữ liệu.

II. LÝ THUYẾT:

Câu lệnh SQL sử dụng trong bài này:

1. Tạo CSDL:

```
CREATE DATABASE <tên CSDL mới>
```

2. Chọn CSDL hiện hành:

```
USE <tên CSDL>
```

3. Tạo bảng:

Lệnh tạo bảng với ràng buộc trên cột:

```
CREATE TABLE <tên_bảng_mới> (
    { <tên_cột> {<kiểu> | <tên_miền_trị>}
      [ NOT NULL ]
      [ PRIMARY KEY | UNIQUE ]
      [ DEFAULT <trị_mặc_định>]
      [ CHECK (<điều_kiện>)]
      [ REFERENCES <tên_bảng_cha>
        [(<danh_sách_tên_cột_của_khóa_chính_trong_bảng_cha>)]
        [ON UPDATE <hành_động>]
        [ON DELETE <hành_động>] ]
    [...] } )
```

Lệnh tạo bảng với ràng buộc trên bảng:

```
CREATE TABLE <tên_bảng> ( { <tên_cột> {<kiểu> | <tên_miền_trị>} [...] }
  [, PRIMARY KEY (<danh_sách_cột> )
  { [, UNIQUE (<danh_sách_cột>)] [, ...] }
  { [, FOREIGN KEY (<ds_cột_khóa_ngoại>
    REFERENCES <tên_bảng_cha> [(danh_sách_thuộc_tính_khóa)],
    [ON UPDATE <hành_động>]
    [ON DELETE <hành_động>] ]
    [...] }
  { [, CHECK (<điều_kiện>) [...] }
  )
```

4. Sửa bảng:

```
ALTER TABLE <tên_bảng>
  [ ADD <tên_cột> {<kiểu> | <miền_trị>} <ràng_buộc_trên_cột> ]
  | [ DROP <tên_cột>
  | [ ALTER COLUMN <tên_cột> <kiểu_mới>
  | [ ADD [CONSTRAINT [tên_RB] ] <định_nghĩa_RB>]
  | [ DROP CONSTRAINT <tên_RB>]
```

5. Xóa bảng:

```
DROP TABLE <tên_bảng>
```

6. Xen, sửa, xóa dữ liệu:

Xen một dòng vào bảng đã tồn tại:

```
INSERT INTO <tên bảng>
[( <tên cột 1> [, <tên cột 2> [, ...]])]
VALUES (<biểu thức 1> [, <biểu thức 2> [, ...]])
```

Xen kết quả truy vấn vào bảng đã tồn tại:

```
INSERT INTO <Tên bảng>
[( <tên cột 1> [, <tên cột 2> [, ...]])]
<lệnh SELECT>
```

Cập nhật dữ liệu:

```
UPDATE <Tên bảng>
SET <Tên Cột> = <biểu thức>
[, <Tên cột> = <biểu thức> [, ...]]
[WHERE <Điều kiện chọn các dòng cập nhật>]
```

Xóa dữ liệu:

```
DELETE <Tên bảng>
[WHERE <Điều kiện chọn dòng muốn xóa>]
```

III. BÀI TẬP CÓ HƯỚNG DẪN:

1. Cài đặt cơ sở dữ liệu quản lý bán hàng cùng với các ràng buộc khóa chính, khóa ngoại, ràng buộc miền trị,...

Khách (makh, tenkh, dchi_kh, dthoai_kh, congno)

Mỗi khách hàng có một mã phân biệt, một tên, địa chỉ, điện thoại và công nợ.

Hoadon (so_hd, ngay_hd, ngayxuat, trigia, makh)

Mỗi hóa đơn có một số hóa đơn phân biệt, có một ngày ghi hóa đơn, ngày xuất hàng, trị giá hóa đơn và xuất cho một khách hàng duy nhất. Hàng hóa chỉ xuất sau khi đã lập hóa đơn.

Ct_hd (so_hd, mahh, gia_ban, sl, thanhtien)

Một hóa đơn bán nhiều mặt hàng, mỗi mặt hàng có một số lượng bán, giá bán và thành tiền. Thành tiền chính là giá bán x số lượng.

Phieuthu (so_pt, makh, ngaythu, sotien)

Mỗi khi thu tiền của một khách hàng, một phiếu thu được xuất với số phân biệt, có ghi nhận ngày thu và số tiền thu.

Tạo CSDL và chọn CSDL hiện hành:

```
CREATE DATABASE QLBNH;

Go -- Kết thúc gói (batch), thực hiện xong trước khi thực hiện lệnh kế
USE QLBNH;
```

Các câu lệnh tạo bảng:

```
CREATE TABLE Khách (makh INT PRIMARY KEY IDENTITY,
tenkh char(40) not null, dchi_kh char(50),
dthoai_kh char(15) CHECK (dthoai_kh LIKE
```

```
'([0-9][0-9][0-9])-[0-9][0-9][0-9][0-9][0-9][0-9][0-9])', congno money)
CREATE TABLE Hoadon (so_hd INT PRIMARY KEY IDENTITY,
    ngay_hd DATETIME CHECK (ngay_hd <= getdate()),
    ngayxuat DATETIME CHECK (ngayxuat <= getdate()),
    trigia MONEY NOT NULL, makh INT NOT NULL REFERENCES Khach,
    CHECK(ngay_hd <= ngayxuat) )
CREATE TABLE Ct_hd (so_hd INT REFERENCES Hoadon NOT NULL,
    mahh CHAR(5) NOT NULL, gia_ban MONEY,
    sl INT CHECK (sl>0), thanh_tien AS gia_ban * sl PERSISTED,
    PRIMARY KEY (so_hd, mahh))
CREATE TABLE PhieuThu (So_pt INT PRIMARY KEY IDENTITY,
    makh INT REFERENCES Khach NOT NULL,
    ngay_thu DATETIME NOT NULL CHECK (ngay_thu <=getdate()),
    sotien MONEY)
```

Lưu ý:

- Kiểu dữ liệu của khóa chính và khóa ngoại phải giống nhau.
- Từ khóa IDENTITY qui định giá trị của cột là do hệ thống tự tăng.
- Cột thanh_tien của bảng ct_hd là cột có giá trị được tính toán được (computed column) từ biểu thức đặc tả sau từ khóa AS, PERSISTED qui định cột này sẽ được lưu vật lý.

2. Sửa bảng thêm ràng buộc duy nhất cho cột số điện thoại của bảng Khach, đặt trị mặc định cho các cột ngày ghi hóa đơn và ngày xuất hàng là ngày hiện tại.

```
ALTER TABLE Khach ADD CONSTRAINT uq_sdt UNIQUE (dthoai_kh);
ALTER TABLE Hoadon ADD CONSTRAINT df_ngay_hd DEFAULT getdate() FOR ngay_hd;
ALTER TABLE hoadon ADD CONSTRAINT df_ngay_xuat DEFAULT getdate() FOR ngayxuat;
```

3. Xóa đi ràng buộc vừa tạo

```
ALTER TABLE Hoadon DROP CONSTRAINT df_ngay_hd
```

4. Thêm dữ liệu vào các bảng bằng lệnh

```
insert into Khach (tenkh, dthoai_kh) values ('Nguyen van an','0123456789');

select * from Khach;

insert into Hoadon (ngay_hd, ngayxuat, trigia, makh)
    values ('1-feb-2011', '2-feb-2011', 1000, 1);
insert into Hoadon (trigia, makh)
    values (2000, 1);
select * from Hoadon;

insert into Ct_hd values (1, 'BGOMO', 50,10);
insert into Ct_hd values (1, 'BGTID', 100,5);
```

Lưu ý: Trong lệnh insert, cột tính toán 'thành tiền' sẽ không được đưa giá trị vào.

```
select * from Ct_hd; --xem cột thành tiền đã có giá trị chưa?
```

5. Cập nhật lại ngày xuất hàng của hóa đơn 1 là ngày 3-feb-2011

```
Update Hoadon set ngayxuat = '3-feb-2011' where so_hd = 1
```

6. Xóa toàn bộ dữ liệu vừa nhập

```
delete ct_hd;
delete Hoadon;
delete Khach;
```

Hỏi: Có thể xóa dữ liệu trong các bảng theo trình tự các lệnh delete khác đi không? Tại sao?

IV. BÀI TẬP TỰ LÀM:Sử dụng CSDL1:

1. Hãy cài đặt tất cả các ràng buộc khóa chính, khóa ngoại, khóa duy nhất, miền giá trị cho từng bảng.
2. Trong bảng CongNhan, hãy cập nhật giá trị cho các cột năm sinh và năm vào nghề thành 4 con số (VD: 60 -> 1960).

Trong bảng ThamGia:

3. Hãy thêm cột ThamNien (số năm làm việc tính đến thời điểm tham gia công trình, >0), DonGia (đơn giá ngày công, >0) và cột TienCong (tiền công) ràng buộc >0 và là cột tính toán từ đơn giá ngày công và số ngày tham gia.
4. Cập nhật dữ liệu cho cột ThamNien theo ngày tham gia và năm vào nghề.
5. Cập nhật dữ liệu cho cột DonGia theo thâm niên như sau:
Thâm niên >20 năm => đơn giá = 100
Thâm niên >15 năm => đơn giá = 80
Thâm niên >10 năm => đơn giá = 50
6. Quan sát cột TienCong trong bảng xem có giá trị chưa.

BÀI 6**I. MỤC ĐÍCH:**

Bài tập tổng hợp

II. CSDL: QUẢN LÝ DỰ ÁN (CSDL6)

NHA_CCAP(MA_NCC, TEN_NCC, VON, TH_PHO)

Mỗi nhà cung cấp có một mã nhà cung cấp, một tên, một số vốn điều lệ (đơn vị: triệu đồng), phải có vốn điều lệ tối thiểu là 10 triệu, và đóng tại một thành phố nào đó, mặc định có giá trị là 'Can Tho'.

HANG_HOA(MA_HANG, TEN_HANG, MAU, TR_LUONG)

Mỗi hàng hóa hay mặt hàng có một mã hàng, một tên hàng, có một màu và một trọng lượng duy nhất (phải >0).

DU_AN(MA_DU_AN, TEN_DU_AN, TH_PHO)

Mỗi dự án có một mã, một tên và thuộc về một thành phố nào đó.

CUNG_UNG(MA_NCC, MA_HANG, MA_DU_AN, SO_LUONG)

Một nhà cung cấp sẽ cung ứng nhiều mặt hàng cho một dự án với số lượng cung cấp khác nhau (phải >0).

III. CÂU HỎI:

1. Sau khi phục hồi CSDL6, hãy thêm tất cả các ràng buộc cho từng bảng.
2. Trong bảng Cung Ứng, hãy thêm cột đơn giá (>0) và cột thành tiền trong đó cột thành tiền là cột có giá trị được tính toán từ đơn giá và số lượng.
3. Trong bảng Cung Ứng, hãy cập nhật đơn giá cho mặt hàng p1 là 10, quan sát cột thành tiền.
4. Tìm thông tin đầy đủ gồm mã, tên và thành phố của mỗi dự án ở TP HCM.
5. Tìm mã và tên các mặt hàng được cung ứng cho dự án Tân an.
6. Tìm tên và vốn của các nhà cung cấp đã cung ứng mặt hàng Sơn Bạch Tuyết cho dự án Tân Phú. Cột vốn cần ghi thêm đơn vị tiền tệ đi kèm.
7. Tìm tên các dự án do nhà cung cấp S1 cung ứng.
8. Tìm tên nhà cung cấp đã cung ứng các mặt hàng màu đỏ cho dự án J1.
9. Tìm tên của các mặt hàng mà không có mặt hàng nào có trọng lượng nhỏ hơn.
10. Tìm tên nhà cung cấp đã cung ứng cho cả hai dự án J1 và J2.
11. Hiện thị số mặt hàng mà mỗi nhà cung cấp đã cung ứng cho các dự án.
12. Tìm số lượng trung bình các mặt hàng mà mỗi nhà cung cấp có thể cung ứng.
13. Với mỗi mặt hàng, hãy hiển thị số lượng cung ứng mà mỗi nhà cung cấp đã cung ứng cho các dự án.
14. Đối với mỗi mặt hàng cung ứng cho một dự án, tìm mã hàng, tên dự án và tổng số lượng hàng tương ứng.
15. Đối với mỗi mặt hàng cung ứng cho một dự án, tìm tên hàng, tên dự án và tổng số lượng hàng tương ứng.
16. Tìm tên nhà cung cấp có vốn đầu tư cao nhất.
17. Tìm tên các nhà cung cấp đã cung ứng cho một dự án bất kỳ ở thành phố Hồ Chí Minh.
18. Tìm mã của các nhà cung cấp đã cung ứng cho ít nhất hai dự án có cùng thành phố với nhà cung cấp đó.
19. Tìm tên các dự án không được cung ứng bất cứ mặt hàng màu đỏ nào bởi một hãng ở thành phố Hồ Chí Minh.

20. Tìm mã của các nhà cung cấp cung ứng ít nhất là một mặt hàng, các mặt hàng này được cung ứng ít nhất bởi một nhà cung cấp, nhà cung cấp này cung cấp ít nhất là một mặt hàng màu đỏ.
21. Tìm mã của các dự án đang sử dụng ít nhất là một mặt hàng mà S1 có khả năng cung ứng.
22. Tìm tên dự án đã được cung ứng mặt hàng Xi Măng Hoàng Thạch với số lượng lớn nhất.
23. Tìm mã các dự án được cung ứng hoàn toàn bởi hãng S2.
24. Tìm mã các dự án chỉ sử dụng những mặt hàng mà hãng S1 có khả năng cung ứng.
25. Tìm tổng số lượng của mỗi mặt hàng mà mỗi dự án đã được cung cấp.
26. Tìm tổng số dự án được hãng S3 cung ứng.
27. Tìm tổng số dự án mà mỗi nhà cung cấp đã cung ứng.
28. Tìm tổng số lượng mặt hàng P1 do hãng S1 cung ứng.
29. Tìm trung bình số lượng cho mặt hàng P1 được cung ứng cho một dự án.
30. Tìm trung bình số lượng cho mặt hàng P1 được cung ứng bởi một hãng.
31. Tìm trung bình số lượng cho mặt hàng P1 được cung ứng cho một dự án bởi một hãng.
32. Tìm tên nhà cung cấp đã cung ứng cho ít nhất là 3 dự án.
33. Tìm số lượng các nhà cung cấp đã cung ứng cho các dự án tại ít nhất là 3 thành phố.
34. Tìm số lượng trung bình các bao xi măng (bất kể hiệu gì, màu gì) mà mỗi dự án đã tiêu thụ.
35. Tìm tên dự án đã được cung ứng mặt hàng sơn (bất kể hiệu gì, màu gì) với tổng số lượng cao nhất.
36. Tìm mã của những hãng nào cung ứng cùng một mặt hàng cho tất cả các dự án.
37. Tìm mã các dự án được hãng S1 cung ứng cho tất cả các mặt hàng mà hãng S1 có khả năng cung ứng.
38. Tìm mã các dự án được cung ứng bởi tất cả những hãng nào cung ứng mặt hàng màu đỏ.
39. Tìm mã các mặt hàng được cung ứng cho tất cả các dự án ở TP HCM.