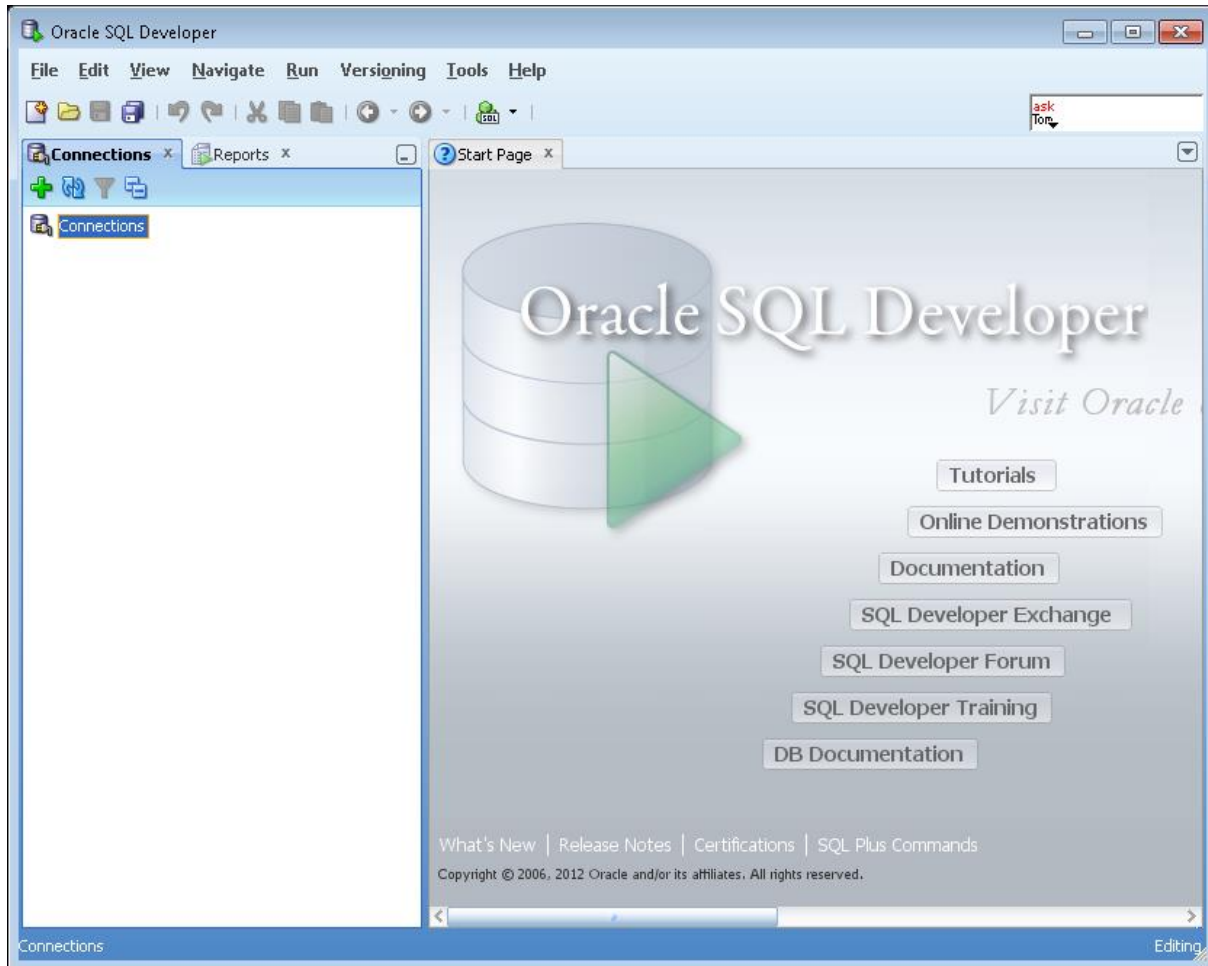



BÀI TẬP SQL TRONG CSDL ORACLE

1.1. Cách sử dụng SQL Developer để làm việc với CSDL.

Khởi động phần mềm Oracle SQL Developer



Hình 8: Giao diện Oracle SQL Developer

- Bấm biểu tượng  (dấu + màu xanh) để tạo một kết nối mới đến database, nhập các thông tin như hình dưới, nếu test thành công, thì bấm connect.

Các thông số:

Connection Name: Tên kết nối đến 1 Schema trong 1 database. (Có thể đặt tùy ý).

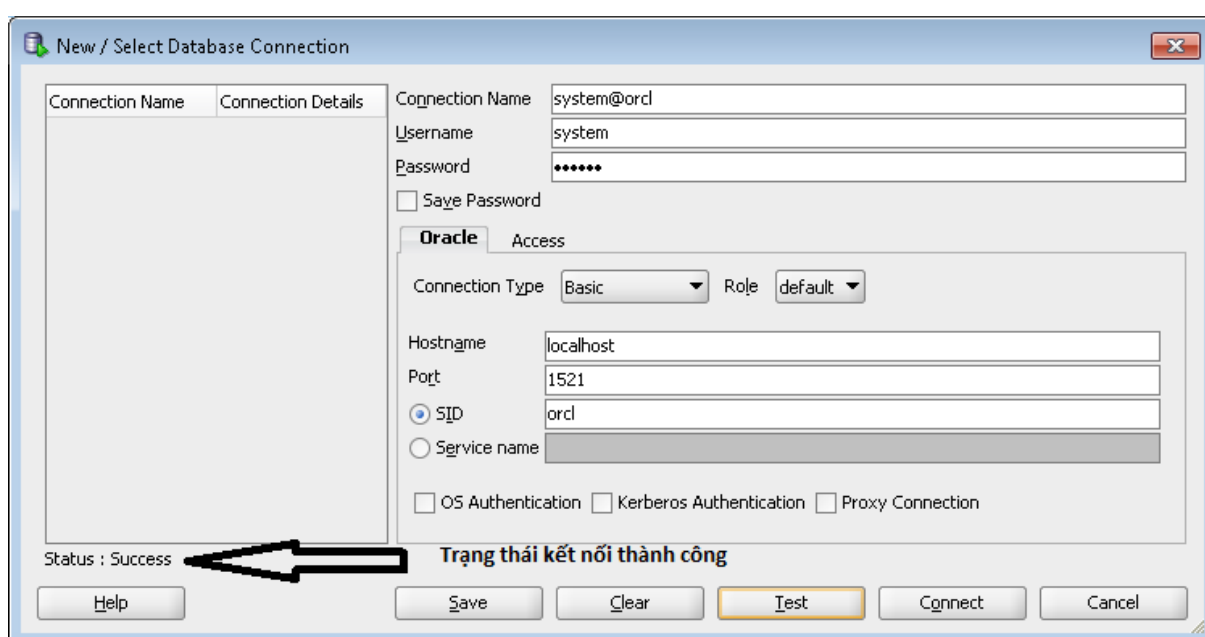
Username: Tên user

Password: Mật khẩu của với user

Hostname: Tên host, có thể sử dụng tên của máy tính, localhost, hoặc địa chỉ IP của máy tính.

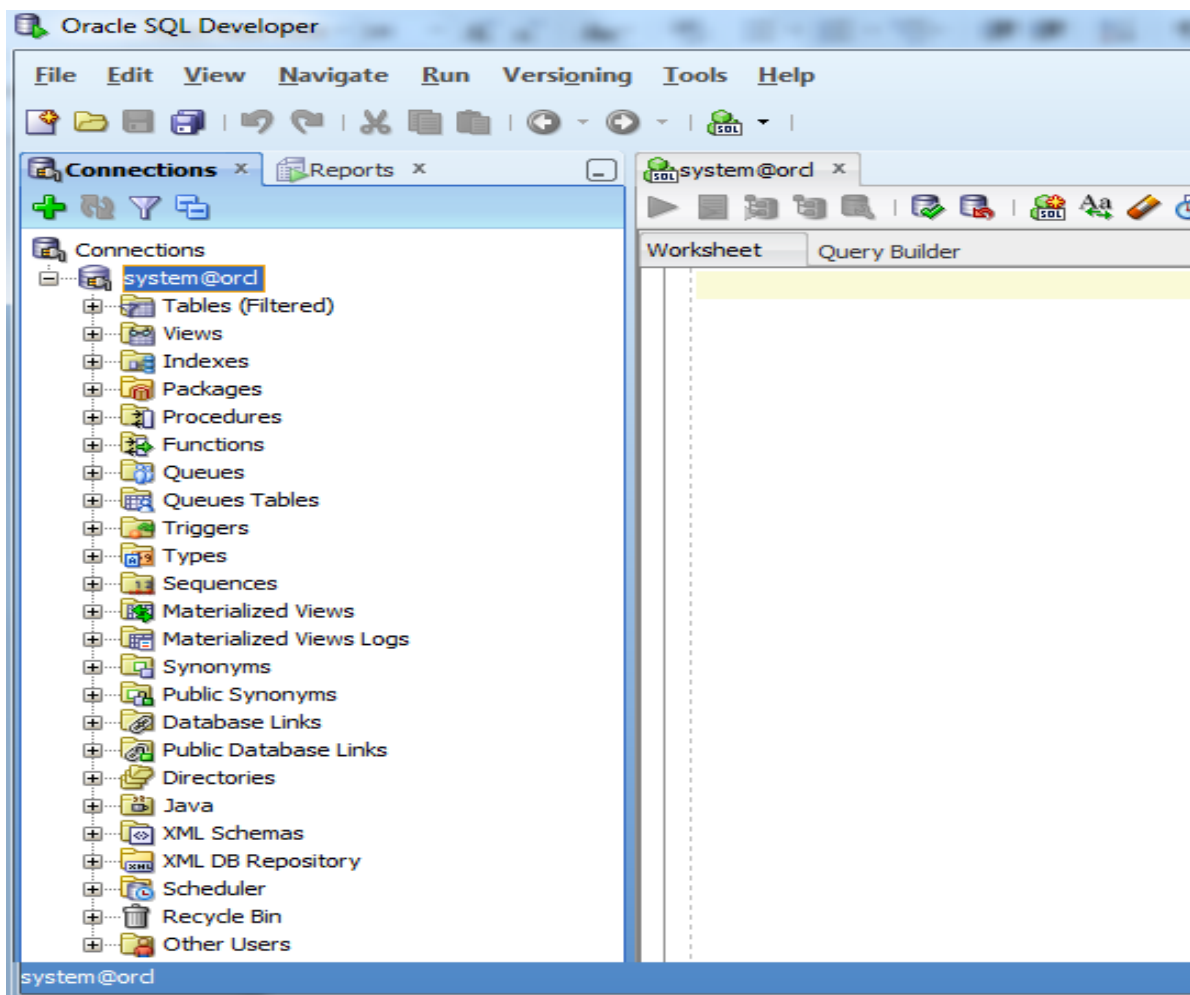
Port: cổng mà listener lắng nghe những yêu cầu từ client để gửi đến Sever. Listener được hiểu như là một cầu kết nối giữa client và các server. Thường sử dụng cổng mặc định 1521.

SID (Oracle System Identifier): là một tên duy nhất cho một database instance trên 1 máy chủ cụ thể. Ở đây chính là tên database lúc khởi tạo.



Hình 9: Giao diện thông tin kết nối đến database

Kết nối thành công, ta được giao diện làm việc như hình dưới:



Hình 10: Giao diện kết nối thành công đến database, bao gồm các đối tượng trong 1 schema.

- Để viết câu lệnh SQL, ta nhấn tổ hợp phím **Alt + F10**.
- Để thực thi câu lệnh sql, ta nhấn tổ hợp phím **Ctrl + Enter**.

CHƯƠNG 3. NGÔN NGỮ SQL VÀ NGÔN NGỮ THỦ TỤC PL/SQL

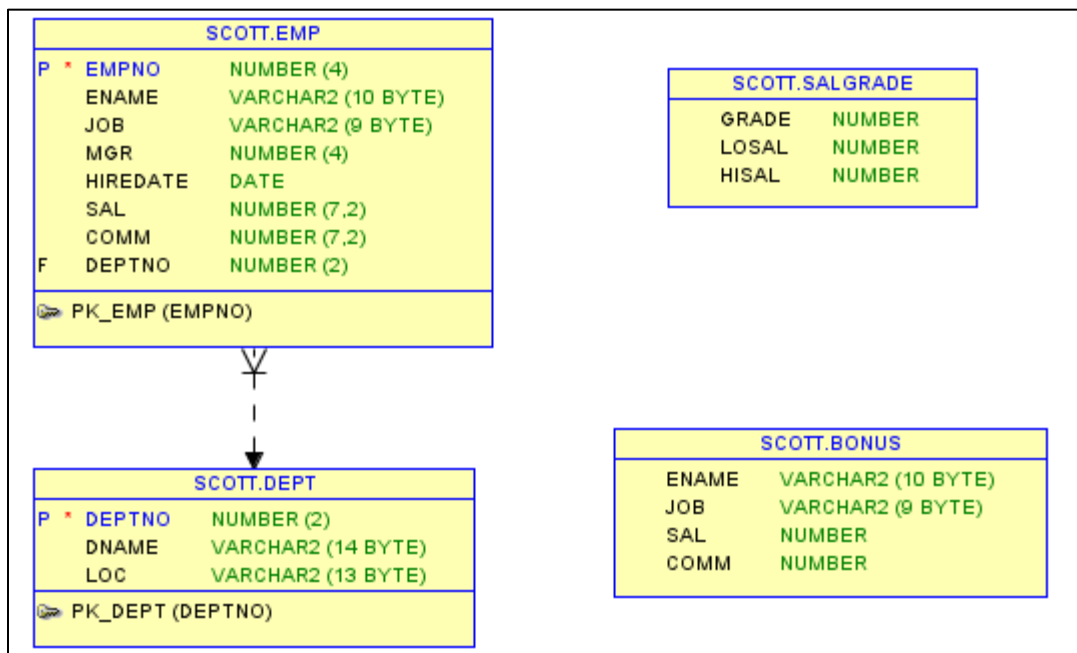
3.1. Ngôn ngữ SQL

3.1.1. Các nhóm lệnh SQL cơ bản

- **Ngôn ngữ định nghĩa dữ liệu (DDL – Data Definition Language):** tạo table (bảng), view (khung nhìn), sửa cấu trúc table và thêm, xóa RBTV, xóa table, xóa view, đổi tên table.
- **Ngôn ngữ thao tác dữ liệu (DML – Data Manipulation Language):** thêm, xóa, sửa dữ liệu, và truy vấn dữ liệu (insert, delete, update, select).
- **Ngôn ngữ điều khiển dữ liệu (DCL – Data Control Language):** tạo quyền hạn, xóa quyền, làm cho hiệu lực/mất hiệu lực quyền, tạo người dùng, đổi mật khẩu, xóa người dùng, cấp quyền và thu hồi quyền sử dụng trên cơ sở dữ liệu.
- Ngoài ra còn có các lệnh điều khiển: lệnh COMMIT, lệnh ROLLBACK, lệnh SAVEPOINT, lệnh AUTOCOMMIT.
- Lệnh thao tác trên các thành phần CSDL khác: Synonym, Index và Sequence.

3.1.2. SQL*Plus và truy vấn dữ liệu cơ bản

- **Sơ đồ quan hệ cơ sở dữ liệu thực hành** (Đã được tạo trong user scott)

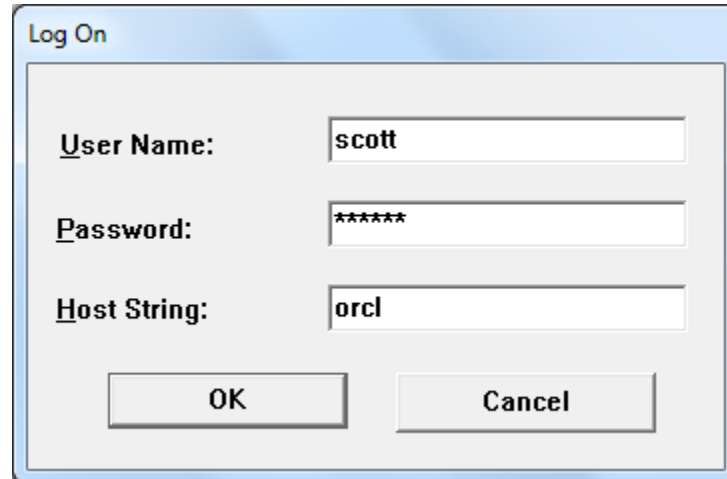


- **SQL*Plus:**

Để sử dụng SQL plus, ta vào: *Start/All Programs/Oracle - DevSuiteHome1/Application Development/SQL Plus*

User Name: Tên user **Password:** Mật khẩu của user

Host String: Tên Database



Hình 11: Giao diện log On SQL Plus

Ghi chú: Ta có thể sử dụng SQL Plus thông qua cmd bằng cách gõ lệnh:

sqlplus [<username/password>]

- Lệnh kết nối tới CSDL

Cú pháp:

Conn[ect] <user_name>/<password>[@<database>];

Với:

user_name: Tên truy nhập

password: Mật khẩu truy nhập

database: Tên database truy nhập

Ví dụ:

Conn scott/abc123@orcl;

Đặc biệt : Kết nối tới user sys : con[nect] sys[/password] as sysdba

- Hiện thị cấu trúc bảng dữ liệu

Cú pháp:

Desc[ribe] <table_name>;

Với:

table_name: Tên bảng cần hiển thị cấu trúc

Ví dụ:

Desc Dept;

- Ghi nhận việc thay đổi dữ liệu: sau khi thực hiện câu lệnh insert, update, delete, để dữ liệu thực sự thay đổi trong bảng, cần phải thực hiện câu lệnh sau: **commit;**

- **Câu lệnh truy vấn cơ bản**

+ Cú pháp:

SELECT [DISTINCT] {*, column [alias],} FROM table;

Với:

SELECT Hiển thị nội dung của một hay nhiều cột

DISTINCT Phân biệt nội dung giữa các dòng dữ liệu trả về

* Lấy tất cả các cột trong bảng

column Tên cột dữ liệu cần trả về

alias phần tiêu đề của cột dữ liệu trả về

FROM table Tên bảng chứa dữ liệu truy vấn

+ Các thành phần khác có thể đưa vào mệnh đề SELECT trong câu lệnh truy vấn

- Biểu thức toán học
- Column alias
- Các column được ghép chuỗi
- Literal (Các chuỗi ký tự)

Biểu thức toán học:

Trong mệnh đề SELECT biểu thức toán học có thể các giá trị (column hoặc hằng số), các toán tử, các hàm.

Các toán tử được dùng là (+), (-), (*), (/). Độ ưu tiên của các toán tử giống trong phần số học.

Ví dụ:

SELECT* *ename, sal *12, comm FROM emp;

Tiêu đề của cột (column alias):

Trong mệnh đề SELECT, column alias là phần nhãn hiển thị của column khi lấy số liệu ra. Trong column alias không được có dấu cách và viết cách sau tên column một dấu cách. Column alias được chấp nhận có dấu cách khi được đặt trong dấu nháy kép (“ ”).

Ví dụ: (ANUAL chính là column alias)

SELECT* *ename, SAL*12 ANUAL, comm FROM emp;

Ghép tiếp các cột dữ liệu:

Toán tử ghép tiếp chuỗi (||) cho phép ghép tiếp dữ liệu trong các cột khác nhau của cùng một dòng dữ liệu với nhau thành một chuỗi.

Ta có thể có nhiều toán tử ghép chuỗi trong cùng một column alias.

Ví dụ:

select empno||ename employee from emp;

Ghép tiếp chuỗi ký tự

Trong mệnh đề SELECT, ta có thể thực hiện ghép tiếp bất kỳ ký tự nào, biểu thức hay số nào mà không phải là column hoặc column alias.

Ví dụ:

select empno || ename || ‘work in department’ || deptno ‘employee detail’ from emp;

- **Truy vấn có điều kiện**

- **Mệnh đề WHERE**

Cú pháp:

SELECT [DISTINCT] {*, column [alias],...} FROM table [WHERE condition (s)];

Mệnh đề WHERE dùng để đặt điều kiện cho toàn bộ câu lệnh truy vấn. Trong mệnh đề WHERE có thể có các thành phần:

- Tên column
- Toán tử so sánh
- Tên column, hằng số hoặc danh sách các giá trị.

Ví dụ:

select deptno, job, ename, sal from emp where sal between 1000 and 2000 ;

➤ **Mệnh đề ORDER BY**

Cú pháp:

SELECT [DISTINCT] {*, column [alias],...} FROM table [WHERE condition]

[ORDER BY expr/position [DESC/ASC]];

Mệnh đề ORDER BY dùng để sắp xếp số liệu được hiển thị và phải đặt ở vị trí sau cùng của câu lệnh truy vấn.

Ví dụ:

select ename, job, sal*12, deptno from emp order by ename;

Mệnh đề ORDER BY mặc định sắp xếp theo thứ tự tăng dần ASC[ENDING]: số thấp trước, ngày nhỏ trước , ký tự theo bảng chữ cái.

Mệnh đề Order còn có thể sắp xếp nhiều column. Các column cần sắp xếp được viết thứ tự sau mệnh đề ORDER BY và cách bởi dấu phẩy (.). Column nào gần mệnh đề ORDER BY hơn có mức độ ưu tiên khi sắp xếp cao hơn. Chỉ định cách thức sắp xếp ASC/DESC được viết sau column cách bởi một dấu cách.

Ví dụ:

SELECT DEPTNO, JOB, ENAME, SAL FROM EMP ORDER BY DEPTNO, SAL DESC;

➤ **Mệnh đề GROUP BY [HAVING]**

Cú pháp:

SELECT [DISTINCT] {*, column [alias],...} FROM table [WHERE condition]
[GROUP BY expr] [HAVING condition] [ORDER BY expr/position
[DESC/ASC]];

Mệnh đề GROUP BY sẽ nhóm các dòng dữ liệu có cùng giá trị của expr. Ví dụ:
GROUP BY JOB nghĩa là sẽ nhóm các nghề giống nhau.

Mệnh đề HAVING là đặt điều kiện của nhóm dữ liệu. Có thể đặt ngay trước hoặc ngay sau mệnh đề GROUP BY. Mệnh đề này khác mệnh đề WHERE ở chỗ mệnh đề WHERE đặt điều kiện cho toàn bộ câu lệnh SELECT.

VD: *select job, max(sal) from emp group by job having max(sal)>=3000;*

3.1.3.Truy vấn dữ liệu mở rộng

- **Kết hợp dữ liệu từ nhiều bảng**

- + **Mối liên kết tương đương**

Mối liên kết tương đương được thể hiện trong mệnh đề WHERE.

Để liên kết trong mệnh đề WHERE phải chỉ rõ tên của các column và mệnh đề được đặt tương đương. Ví dụ: emp.deptno = dept.deptno

Các column trùng tên phải được chỉ rõ column đó nằm ở bảng nào thông qua tên hoặc qua alias. Tên trùng này có thể đặt trong các mệnh đề khác như SELECT, ORDER BY..

VD: *select a.deptno, a.ename, a.job, b.dname from emp a, dept b where a.deptno = b.deptno order by a.deptno;*

- + **Mối liên kết cộng**

Mối liên kết cộng trả về cả các giá trị NULL trong biểu thức điều kiện. Dấu (+) để ở vế nào tính thêm các giá trị NULL ở vế đó.

Một câu lệnh select chỉ đặt được 1 mối liên kết cộng, dấu (+) đặt ở bên phải column liên kết.

Trong mệnh đề WHERE của mỗi liên kết cộng không được dùng toán tử IN hoặc OR để nối các điều kiện liên kết khác.

VD:

select e.ename, d.deptno, d.dname from emp e, dept d where e.deptno (+) = d.deptno and d.deptno in (30, 40);

+ **Liên kết của bảng với chính nó (tự thân)**

Có thể liên kết bảng với chính nó bằng cách đặt alias.

VD: Hiển thị thông tin bao gồm tên nhân viên, lương nhân viên, tên người quản lý của nhân viên đó, lương người quản lý đó với điều kiện lương của nhân viên lớn hơn lương người quản lý nhân viên đó.

Select e.ename ten_nhan_vien, e.sal luong, m.ename ten_quan_ly, m.sal luong_quan_ly from emp e, emp m where e.mgr = m.empno and e.sal > m.sal;

R	TEN_NHAN_VIEN	R	LUONG	R	TEN_QUAN_LY	R	LUONG_QUAN_LY
1	SCOTT		3000	JONES		2975	
2	FORD		3000	JONES		2975	

• **Lệnh truy vấn lồng nhau**

+ **Câu lệnh SELECT lồng nhau**

Trong mệnh đề WHERE.

VD: Tìm những nhân viên làm cùng nghề với BLAKE

Select ename, job from emp where job = (select job from emp where ename = 'BLAKE');

Trong mệnh đề HAVING.

VD: Tìm những phòng có mức lương trung bình lớn hơn phòng 30.

Select deptno, avg(sal) from emp having avg(sal) > (select avg(sal) from emp where deptno=30) group by deptno;

+**Toán tử SOME/ANY/ALL/NOT IN/EXISTS**

TÊN TOÁN TỬ	DIỄN GIẢI
NOT IN / IN	Không thuộc / Thuộc

ANY và SOME	So sánh một giá trị với mỗi giá trị trong một danh sách hay trong kết quả trả về của câu hỏi con, phải sau toán tử =
ALL	So sánh một giá trị với mọi giá trị trong danh sách hay trong kết quả trả về của câu hỏi con.
EXISTS	Trả về TRUE nếu có tồn tại.

Ví dụ:

*Select * from emp where sal = any (select sal from emp where deptno=30);*

*Select * from emp where sal >= all (select distinct sal from emp where deptno =30) order by sal desc;*

Select ename, sal, job, deptno from emp where sal > some (select distinct sal from emp where deptno =30);

BÀI TẬP

1. Hiển thị tên nhân viên và thu nhập trong một năm của các nhân viên.
2. Hiển thị thông tin nhân viên theo nội dung: Who, what and when, dữ liệu hiển thị như ví dụ dưới đây:
KING HAS HELD THE POSITION OF PRESIDENT IN DEPT 10 SINCE 17-11-1981
3. Hiển thị cấu trúc bảng emp.
4. Chọn nhân viên trong bảng EMP có mức lương từ 1000 đến 2000 (chọn các trường ENAME, DEPTNO, SAL).
5. Hiển thị thông tin những nhân viên làm công việc thư ký (clerk) tại phòng 20.
6. Hiển thị tất cả những nhân viên mà tên có các ký tự TH và LL.
7. Hiển thị tên nhân viên, mã phòng ban, ngày gia nhập công ty sao cho gia nhập công ty trong năm 1983.
8. Tìm lương thấp nhất, lớn nhất và lương trung bình của tất cả các nhân viên

9. Tìm lương nhỏ nhất và lớn nhất của mỗi loại nghề nghiệp.
10. Đếm xem có bao nhiêu quản lý(manager) trong danh sách nhân viên.
11. Tìm tất cả các phòng ban mà số nhân viên trong phòng > 3
12. Hiển thị tên nhân viên, vị trí địa lý(LOC), tên phòng với điều kiện lương >1500.
13. Hiển thị tên nhân viên, nghề nghiệp, lương, mức lương, tên phòng làm việc trừ nhân viên có nghề là cleck và sắp xếp theo mức lương tăng dần.
14. Hiển thị chi tiết về những nhân viên kiếm được 36000\$ 1 năm hoặc nghề là cleck. (gồm các trường tên, nghề, thu nhập, mã phòng, tên phòng, mức lương)
15. Hiển thị những phòng không có nhân viên nào làm việc
16. Tìm những nhân viên kiếm được lương cao nhất trong mỗi loại nghề nghiệp.
17. Tìm mức lương cao nhất trong mỗi phòng ban, sắp xếp theo thứ tự phòng ban.
18. Tìm nhân viên gia nhập vào phòng ban sớm nhất, sắp xếp theo mã phòng ban tăng dần.
19. **Hiển thị những nhân viên có mức lương lớn hơn lương TB của phòng ban mà họ làm việc.