

Exercice SQL pour Oracle

Deux tables sont utilisées :

- La table EMP des employés
- La table DEPT des départements

EMP (EMPNO, ENAME, JOB, #MGR, HIREDATE, SAL, COMM, #DEPTNO)

DEPT (DEPTNO, DNAME, LOC)

Donnez les requêtes SQL permettant de trouver :

1 : Informations sur les employés dont la fonction est "MANAGER" dans les départements 20 et 30

Select * from emp where job = 'MANAGER' and deptno in (20,30);

2 : Liste des employés qui n'ont pas la fonction "MANAGER" et qui ont été embauchés en 81

a : Select ename, job, hiredate from emp where job != 'MANAGER' and TO_CHAR(HIREDATE,'YY') = '81'

B : Select ename, job, hiredate from emp where job <> 'MANAGER' and HIREDATE between = '01/01/81' and '31/12/81'

3 : Liste des employés ayant un "M" et un "A" dans leur nom

select ename from emp where ename like '%M%' and ename like '%A%'

4 : Liste des employés ayant deux "A" dans leur nom

select ename from emp where ename like '%A%A%'

5 : Liste des employés ayant une commission

select * from emp where comm IS NOT NULL

6 : Liste des noms, numéros de département, jobs et dates d'embauches, triés par :

- numéro de département croissant,

- ordre alphabétique des jobs,

- ancienneté croissante (les derniers embauchés d'abord)

select ename, deptno, job, hiredate from emp order by deptno, job, hiredate DESC

7 : Liste des employés travaillant à "DALLAS"

select emp.* from emp, dept where emp.deptno = dept.deptno and loc='DALLAS'

8 : Noms et dates d'embauche des employés embauchés avant leur manager, avec le nom et la date d'embauche du manager

select e.ename, e.hiredate, m.ename, m.hiredate from emp e, emp m where e.mgr=m.empno and e.hiredate < m.hiredate

9 : Noms et dates d'embauche des employés embauchés avant 'BLAKE'

a: select e1.ename, e1.hiredate from emp e1, emp bl where bl.name='black' and e1.hiredate<bl.hiredate

b: select ename, hiredate from emp where hiredate < (select hiredate from emp where ename = 'BLACK')

10 : Lister les noms et numéros des employés n'ayant pas de subordonnés

select ename, empno from emp X where NOT EXISTS (select mgr from emp where X.empno = mgr)

pour afficher l'employé non énuméré dans la liste des managers avec des subordonnées vide.

ont fait un test sur le mgr vide pour chercher les subordonnées vide.

select ename, empno from emp where empno NOT IN(select mgr from emp where mgr IS NOT NULL)

11 : Employés embauchés le même jour que 'FORD'

a : select * from emp where hiredate=(select hiredate from emp where ename='FORD')

b : select e1.* from emp e1, emp e2 where e1.hiredate=e2.hiredate and e2.ename='FORD'

12 : Employés ayant le même manager que 'CLARK'

select * from emp where mgr=(select mgr from emp where ename='CLARK')

Pour ne pas afficher 'CLARK' dans le résultat, ajouter **and ename != 'CLARK'**

13 : Employés embauchés avant tous les employés du département 10

select * from emp where hiredate < (select min(hiredate) from emp where deptno = 10)

14 : Employés ayant le même job et même manager que 'TURNER'

select * from emp where (job,mgr) = (select job,mgr from emp where ename='TURNER');

select e1.* from emp e1,emp2 where e1.job=e2.job and e1.mgr=e2.mgr and e2.ename='TURNER';

15 : Employés de département 'RESEARCH' embauchés le même jour que quelqu'un du département 'SALES'

select emp.* from emp, dept where emp.deptno=dept.deptno and dname = 'RESEARCH' and hiredate IN

(select hiredate from emp, dept where emp.deptno=dept.deptno and dname='SALES');

16 : Employés gagnant plus que leur manager (on ne prend pas comm en compte)

select e1.* from emp e1, emp manager where e1.sal >manager.sal and manager.empno=e1.mgr

17 : Liste des noms des employés avec les salaires tronqués au millier

select ename, TRUNC(sal,-3) "Salaire au millier" from emp order by sal;

18 : Liste des employés en remplaçant les noms par "---" dans le département 10

Select DECODE(deptno,10,'---',ename), deptno from emp;

19 : Faire un histogramme des salaires

taper "set arraysize 1" sous SQL*Plus afin d'éviter l'erreur "Débordement tampon. Réduire ARRAYSIZE ou augmenter MAXDATA avec SET" col Histogramme format a50

select ename, LPAD('>',sal/5000*20,'#') HISTOGRAMME from emp order by sal desc;

20 : Noms des employés avec date de début du mois d'embauche

col mois format a35 heading "Début du mois d'embauche"

select ename, trunc(hiredate,'mm') mois from emp;

21 : Nom et nombre de mois d'ancienneté des employés le 1^{er} janvier 2000

select ename, month_between(TO_DATE('01-01-2000', 'DD-MM-YYYY'), hiredate) "Ancienneté en mois" from emp order by 2 desc

22 : Listes des salaires par job et par département

select job, decode(deptno,10,sal,0) "Département 10", decode(deptno,20,sal,0) "Département 20"

decode(deptno,30,sal,0) "Département 30" from emp;

23 : Salaire moyen en tenant compte des commission

select AVG(sal+comm) as "Salaire moyen" from emp;

24 : Nombre d'employés pour chaque job

select job, count(*) as "Nombre d'employés" from emp GROUP BY job order by 2 desc

25 : Nombre d'employés dans chaque tranche de salaire (tranche en millier)

select trunc(sal,-3) Tranche, count(*) as "Nombre d'employés" from emp GROUP BY trunc(sal,-3);

26 : Employés ayant le salaire le plus élevé dans chaque département

select deptno, MAX(sal) from emp GROUPE BY deptno

27 : Job ayant le salaire moyen le plus bas

select job, AVG(sal) as "Salaire moyen" from emp GROUPE BY job HAVING AVG(sal)=(select

MIN(AVG(sal)) from emp GROUPE BY job)

28 : Somme des salaires par job et par département

select job, sum(decode(deptno,10,sal,0)) "Département 10", sum(decode(deptno,20,sal,0)) "Département 20",sum(decode(deptno,30,sal,0)) "Département 30" from emp group by job;

29 : Totaliser l'état précédent par job

select job, sum(decode(deptno,10,sal,0)) "Département 10", sum(decode(deptno,20,sal,0)) "Département 20",sum(decode(deptno,30,sal,0)) "Département 30",sum(sal) "Total par job" from emp group by job;

30 : Arbre hiérarchique de la société

select lpad(' ',LEVEL*3)||ename Hiérarchie from emp CONNECT BY mgr = PRIOR empno START WITH mgr is null;

31 : Ajouter au salaire des managers 5% du salaire de KING, vérifier par un select

UPDATE emp E SET sal=(select E.sal+0.05*emp.sal from emp where ename='KING') where job='MANAGER';

select * from emp;

33 : Supprimer les employés dont le salaire est inférieur au salaire moyen de leur département

DELETE FROM emp where sal<(select avg(E.sal) from emp E where E.deptno=emp.deptno);

34 : Augmenter les salaires des managers de 10%

UPDATE emp SET sal=sal+0.1*sal from emp where job='MANAGER';

35 : Création d'une table dont la structure est identique à la table DEPT

Create table departement as select * from dept ;

36 : Modification de cette table afin de rendre le "deptno" obligatoire

alter table departement modify(deptno number(2) not null);