

La base de données de démonstration d'Oracle a les schémas de tables suivantes :

DEPT (Deptno , Dname , City)

BONUS (Ename , Job , Sal , Comm)

SALGRADE (Grade , LoSal , HiSal)

EMP (Empno , Ename , Job , MgrNo , Hiredate , Sal , Comm , Deptno)

Exercice 1

Ecrire en SQL d'Oracle les requêtes suivantes :

1. Chercher les villes de départements ayant au moins un employé qui a une commission supérieure à son salaire.

```
> SELECT City
FROM DEPT,EMP
WHERE DEPT.Deptno=EMP.Deptno
AND Comm>Sal;
```

2. Liste des employés dont les noms ne commencent pas par 'A' et ne se terminent pas par 'S' et qui sont embauchés après le 31-DEC-81.

```
> SELECT Ename
FROM EMP
WHERE TO_CHAR(Hiredate, 'YY-MM-DD')>'81-12-31'
AND Ename NOT LIKE 'A%S';
```

3. Liste des employés, avec leurs départements de travail, dont les commissions sont inconnues.

```
> SELECT Ename, Dname
FROM EMP, DEPT
WHERE DEPT.Deptno=EMP.Deptno
AND Comm IS NULL;
```

4. Liste des employés qui travaillent dans le département numéro 10 ou 30, et qui ont les salaires entre 1000 et 3000 euros. Afficher la liste avec les colonnes EMP-NAME, DEPTNO, JOB, SALARY.

```
> SELECT Ename AS EMP-NAME, Deptno AS DEPTNO, Job AS JOB, Sal AS SALARY
FROM EMP
WHERE Deptno IN (10,30)
AND Sal BETWEEN 1000 AND 3000;
```

5. Liste des employés qui travaillent comme 'SALESMAN' ou 'CLERK', et qui ont les salaires entre 1000 et 3000 euros. Afficher la liste avec les colonnes EMP-NAME, JOB, SALARY, COMM.

```
> SELECT Ename AS ENAME, Job AS JOB, Sal AS SALARY, Comm AS COMM
FROM EMP
WHERE Job IN ('SALESMAN', 'CLERK')
AND Sal BETWEEN 1000 AND 3000;
```

6. Liste des commissions qui apparaissent à la fois dans la table **BONUS** et dans la table **EMP**.

```
> SELECT Comm
FROM BONUS

INTERSECT

SELECT Comm
FROM EMP;
```

7. Liste des commissions qui apparaissent dans la table **EMP** et qui n'apparaissent pas dans la table **BONUS**.

```
> SELECT Comm
FROM EMP

MINUS

SELECT Comm
FROM BONUS;
```

8. Chercher les directeurs et les départements des employés qui ont le même salaire que *WARD* et qui ont la commission supérieure à la commission de *WARD*.

```
> SELECT MgrNo,Dname
FROM DEPT,EMP
WHERE DEPT.Deptno=EMP.Deptno
AND Sal=(SELECT SAL
FROM EMP
WHERE Ename='WARD' )
AND Comm>(SELECT Comm
FROM EMP
WHERE Ename='WARD' );
```

9. Liste des employés qui ne sont ni '*PRESIDENT*' ni '*MANAGER*'.

```
> SELECT Ename
FROM EMP

MINUS

SELECT *
FROM EMP
WHERE Job IN ('PRESIDENT','MANAGER');
```

10. Liste des employés qui ont le même grade de salaire que *JONES*.

```
> SELECT Ename
FROM EMP E
WHERE (SELECT Grade
FROM SALGRADE
WHERE E.Sal>=LoSal
AND E.Sal<=HiSal)
=
(SELECT Grade
FROM SALGRADE,EMP
WHERE EMP.Sal>=LoSal
AND EMP.Sal<=HiSal
AND EMP.Ename='JONES');
```

11. Liste des départements qui n'ont pas d'employés figurés dans la table **BONUS**.

```
> SELECT *
FROM DEPT

MINUS

SELECT DEPT.Deptno,DEPT.Dname,DEPT.City
FROM DEPT,EMP,BONUS
WHERE EMP.Deptno=DEPT.Deptno
AND EMP.Ename=BONUS.Ename;
```

12. Liste d'employés dans le département '*RESEARCH*' ou '*ACCOUNTING*' qui ne sont pas figurés dans la table **BONUS**.

```
> SELECT EMP.Ename
FROM DEPT,EMP
WHERE DEPT.Deptno=EMP.Deptno
AND Dname IN ('RESEARCH','ACCOUNTING')

MINUS

SELECT EMP.Ename
FROM DEPT,EMP,BONUS
WHERE EMP.Deptno=DEPT.Deptno
AND EMP.Ename=BONUS.Ename;
```

13. Liste de directeurs et leurs employés. Trier la liste dans l'ordre croissant de noms de directeurs, et noms d'employés.

```
> SELECT D.Ename, E.Ename
FROM EMP E, EMP D
WHERE D.Empno=E.MgrNo
ORDER BY D.Ename ASC, E.Ename ASC;
```

14. Liste des employés qui a un salaire supérieur au salaire de son directeur.

```
> SELECT E.Empno, E.Ename, E.Job, E.MgrNo, E.Hiredate, E.Sal, E.Comm, E.Deptno
FROM EMP E, EMP D
WHERE D.Empno=E.MgrNo
AND D.Sal<E.Sal;
```

15. Liste des noms de départements tels que le président n'est pas un employé dans ces départements. Quels sont les grades de salaires des employés dans ces départements.

```
> SELECT Dname, Grade
FROM DEPT, SALGRADE, EMP
WHERE DEPT.Deptno=EMP.Deptno
AND EMP.Sal>=SALGRADE.LoSal
AND EMP.Sal<=SALGRADE.HiSal

MINUS

SELECT Dname, Grade
FROM DEPT, EMP, SALGRADE
WHERE DEPT.Deptno=EMP.Deptno
AND EMP.Job LIKE 'PRESIDENT';
```

Exercice 2

Ecrire en SQL d'Oracle les requêtes suivantes :

1. Supposons que la charge sociale est de l'ordre de 15% du salaire. Pour les employés de la table **EMP**, dont la charge est supérieure à 700 euros ou le salaire est inférieur à 1500 euros, afficher leurs noms, leur salaire et la charge avec deux chiffres décimaux.

```
> SELECT Ename AS Employe, Sal AS Salaire, ROUND(Sal*0.15,2) AS Charges
FROM EMP
WHERE Sal*0.15>700
OR Sal<1500;
```

2. Supposons que la charge sociale est de l'ordre de 15% du salaire, et que la cotisation à la sécurité sociale dépend des grades de salaires : 5% pour grade 1, 5.5% pour grade 2, ..., 7% pour grade 5. Afficher les noms des employés de la table **EMP**, leurs salaires, les charges et les cotisations à la sécurité sociale, avec deux chiffres décimaux.

```
> SELECT EMP.Ename AS Employe, EMP.Sal AS Salaire, ROUND(Sal*0.15,2) AS
Charges, ROUND(Sal*(4.5+Grade*0.5)/100,2) AS Cotisation_SS
FROM EMP, SALGRADE
WHERE EMP.Sal>=LoSal
AND EMP.Sal<=HiSal;
```

3. Supposons que chaque employé dans la table **BONUS** a une prime de 5% de son salaire plus la commission, si celle-ci n'est pas nulle, sinon la prime est égal à 5% de son salaire plus 100 euros. Afficher les noms de ces employés, leurs salaires, commissions et primes.

```
> SELECT EMP.Ename AS Employe, EMP.Sal AS Salaire, EMP.Comm AS
Commission, ROUND(EMP.Sal*5/100+EMP.Comm,2) AS Prime
FROM EMP, BONUS
WHERE EMP.Ename=BONUS.Ename
AND (BONUS.Comm!=0 AND BONUS.Comm IS NOT NULL)

UNION
```

```

SELECT EMP.Ename,EMP.Sal,EMP.Comm,ROUND(EMP.Sal*5/100+100,2)
FROM EMP,BONUS
WHERE EMP.Ename=BONUS.Ename
AND (BONUS.Comm=0 OR BONUS.Comm IS NULL);

```

4. Supposons que chaque employé figuré dans la table **BONUS** a une prime de 6% de son salaire si son salaire est de grade 1, 7% pour le grade 2, ..., 10% pour le grade 5. Afficher les noms de ces employés, leurs salaires, commissions et primes, si leurs primes ne dépassent pas 1000 euros.

```

> SELECT EMP.Ename AS Employe,EMP.Sal AS Salaire,EMP.Comm AS
Commission,ROUND(EMP.Sal*(5+Grade)/100,2) AS Prime
FROM EMP,BONUS,SALGRADE
WHERE EMP.Ename=BONUS.Ename
AND EMP.Sal>=LoSal
AND EMP.Sal<=HiSal;

```

5. Calculer la somme des salaires et le nombre d'employés de la table **EMP**.

```

> SELECT SUM(Sal) AS Somme_Salaires,COUNT(*) AS NB_Employes
FROM EMP;

```

6. Calculer la somme des salaires et le nombre des employés des deux départements n°30 ou n°20 dont le salaire est inférieur à 1000 euros ou qui n'ont pas de commission.

```

> SELECT SUM(Sal) AS Somme_Salaires,COUNT(*) AS NB_Employes
FROM EMP
WHERE Deptno IN (20,30)
AND (Sal<1000 OR Comm IS NULL);

```

7. Avec la même hypothèse que la question 1, calculer les sommes des salaires et de charges sociales des employés dont le numéro de directeur (*MgrNo*) est 7698.

```

> SELECT SUM(Sal) AS Somme_Salaires,ROUND(SUM(Sal*0.15),2) AS
Somme_Charges
FROM EMP
WHERE MgrNo=7698;

```

8. Avec la même hypothèse que la question 2, calculer les sommes des salaires, de charges sociales, et de cotisations à la sécurité sociale des employés ayant un salaire supérieur ou égal à 1500 euros, ou ayant une commission connue et un salaire inférieur à 1500 euros.

```

> SELECT SUM(Sal) AS Somme_Salaires,ROUND(SUM(Sal*0.15),2) AS
Somme_Charges,ROUND(SUM((Sal*4.5+Grade*0.5)/100),2) AS
Somme_Cotisation_SS
FROM EMP,SALGRADE
WHERE Sal>=1500
OR (Sal<1500 AND Comm IS NOT NULL)
AND EMP.Sal>=LoSal
AND EMP.Sal<=HiSal;

```

9. Liste du salaire minimal, du salaire maximal, du salaire moyen, de l'écart-type et de la variance des salaires dans la table **EMP**.

```

> SELECT MIN(Sal) AS Salaire_MIN,MAX(Sal) AS Salaire_MAX,ROUND(AVG(Sal),
2) AS Salaire_MOYEN,ROUND(STDDEV(Sal),2) AS ECART_TYPE,
ROUND(VARIANCE(Sal),2) AS VARIANCE
FROM EMP;

```

10. Liste des départements avec le nombre d'employés dans chaque département, ainsi que la somme des salaires annuels à payer aux employés.

```

> SELECT Dname AS Departement,COUNT(Ename) AS Nb_Employes,SUM(Sal*12) AS
Somme_Salaires_Annuels
FROM EMP,DEPT
WHERE DEPT.Deptno=EMP.Deptno
GROUP BY Dname;

```

11. Liste des sommes de salaires d'employés d'après les grades de salaire.

```
> SELECT Grade, SUM(Sal) AS Somme_Salaires_Employes
FROM EMP, SALGRADE
WHERE EMP.Sal >= LoSal
AND EMP.Sal <= HiSal
GROUP BY Grade;
```

12. Liste des grades de salaires : pour chaque grade de salaire, donner les numéros de départements et pour chaque département, donner le nombre d'employés du département ayant ce grade de salaire, la somme de leurs salaires, si la moyenne de ces salaires dépasse 2000 euros.

```
> SELECT S1.Grade, E1.Deptno, COUNT(E1.Ename), SUM(E1.Sal)
FROM EMP E1, SALGRADE S1
WHERE E1.Sal >= S1.LoSal
AND E1.Sal <= S1.HiSal
AND (SELECT AVG(E2.Sal)
FROM EMP E2, SALGRADE S2
WHERE E2.Sal >= S2.LoSal
AND E2.Sal <= S2.HiSal
AND S2.Grade = S1.Grade
AND E2.Deptno = E1.Deptno
GROUP BY S2.Grade, E2.Deptno)
> 2000
GROUP BY S1.Grade, E1.Deptno
ORDER BY S1.Grade ASC, E1.Deptno ASC;
```

13. Liste des départements tels que la somme des salaires de leurs employés est la plus grande parmi les sommes de salaires des employés de chaque département.

```
> SELECT Dname
FROM DEPT D1
WHERE (SELECT SUM(E2.Sal)
FROM EMP E2, DEPT D2
WHERE E2.Deptno = D2.Deptno
AND D2.Deptno = D1.Deptno
GROUP BY E2.Deptno)
=
(SELECT MAX(SUM(Sal))
FROM EMP, DEPT
WHERE EMP.Deptno = DEPT.Deptno
GROUP BY DEPT.Deptno);
```

14. Liste des employés ayant les salaires inférieurs au salaire moyen dans leur département.

```
> SELECT Ename
FROM EMP E
WHERE E.Sal < (SELECT AVG(Sal)
FROM EMP, DEPT
WHERE EMP.Deptno = DEPT.Deptno
AND E.Deptno = EMP.Deptno
GROUP BY DEPT.Deptno);
```

15. Liste des employés les plus anciens dans chaque département.

```
> SELECT E.Ename
FROM EMP E, DEPT
WHERE DEPT.Deptno = E.Deptno
AND TO_CHAR(E.Hiredate, 'YY-MM-DD') <=
(SELECT MIN(TO_CHAR(EMP.Hiredate, 'YY-MM-DD'))
FROM EMP
WHERE E.Deptno = EMP.Deptno
GROUP BY EMP.Deptno);
```