

1) La clause GROUP BY.



L'objectif du groupement de données (clause GROUP BY) est, comme le nom l'indique, de "regrouper" les données, chaque groupe étant l'**ensemble des lignes ayant une valeur commune**. C'est la clause GROUP BY de la phrase SQL qui nous permet de donner le regroupement à réaliser :

SELECT <liste d'attributs>

FROM < liste de tables >

GROUP BY ...

La clause GROUP BY est obligatoire dès lors qu'il y a à la fois des attributs et des fonctions de calcul derrière la clause SELECT.

Exemple n°1 :

```
SELECT to_char(hiredate,'YYYY') as "Annee", count(*) AS "Nombre d'embauches"
FROM emp
GROUP BY to_char(hiredate,'YYYY')
```

--> Affiche, pour chaque année où une embauche a eu lieu, le nombre d'embauches effectuées.

Résultat affiché :

Annee	Nombre d'embauches
1980	4
1982	2
1981	8

Exemple n°2 : Afficher, pour chaque département embauchant au moins un salarié, l'effectif total du département.

Résultat affiché :

Nom du departement	Effectif
ACCOUNTING	3
RESEARCH	5
SALES	6

Compléments sur les requêtes de sélection.

Remarques :

- Si on a une seule expression dans la clause (ex : GROUP BY expr1), ceci définit les groupes comme les ensembles de lignes pour lesquelles l'expression «expr1» prend la même valeur.
- Si plusieurs expressions sont présentes dans la clause GROUP BY (ex : GROUP BY expr1, expr2...), les groupes sont définis de la façon suivante : parmi toutes les lignes pour lesquelles expr_1 prend la même valeur, on regroupe celles ayant expr_2 identique, ...

Remarque : Dans la liste des colonnes résultat d'un SELECT comportant une fonction de groupe, ne peuvent figurer que des caractéristiques de groupe, c'est-à-dire :

- soit des fonctions de groupe ;
- soit des expressions figurant dans le GROUP BY.

Exemple n°3 : Afficher, pour chaque département, sa localisation ainsi que le nombre de salariés qui y travaillent.

DNAME	LOC	EFFECTIF
RESEARCH	DALLAS	5
SALES	CHICAGO	6
ACCOUNTING	NEW YORK	3

2) Sélection (ou élimination) de certains groupes.



Lorsque l'on obtient le résultat d'une requête issue d'un regroupement, on peut souhaiter ne sélectionner que certains groupes parmi d'autres. Pour cela, il faut ajouter à GROUP BY la clause HAVING qui permet la sélection de groupes parmi d'autres.

Attention à ne pas confondre HAVING et WHERE : **HAVING permet la sélection de groupes à la suite d'une requête avec regroupement, alors que WHERE permet de sélectionner des lignes pour construire la requête.**

Exemple n°4 :

```
SELECT v.nom as "Village", count(g.gauno) as "Effectif", v.nbhutte as "Nombre de huttes"
FROM Village v, Gaulois g
WHERE v.vilno = g.vilno
GROUP BY v.vilno, v.nom, v.nbhutte
HAVING count(g.gauno) > 50
ORDER BY v.nom
```

Compléments sur les requêtes de sélection.

Résultat affiché :

Village	Effectif	Nombre de huttes
Alesia	2	568
Lutece	6	734
Serum	4	356

Exercices d'application.

1. Total des salaires versés pour chaque département
2. Salaire le plus élevé de chaque département.
3. Donner le nombre d'employés de tous les départements existants.
4. Donner la moyenne des salaires des analystes groupés par service
5. Donner le nom des services dans lesquels la moyenne des salaires est supérieure à 1800€
6. Afficher les lettres de l'alphabet qui sont l'initiale d'au moins trois employés.
7. Quel est le salaire moyen, les salaires min et max et le nombre d'employé par profession ?
8. Pour chaque fonction (job), donner le nombre d'employés qui l'exercent et le salaire moyen.
9. Quelles sont les fonctions pour lesquelles travaillent plus de trois personnes ?
10. Combien y a-t-il d'employés par département et par fonction et quelle est la moyenne de leurs salaires ?

2) Requêtes corrélées.



Principe général : Une **sous-requête corrélée** est une sous-requête qui s'exécute pour chaque ligne de la requête principale et non une fois pour toute. Pour arriver à un tel résultat, il suffit de faire varier une condition (en général un prédicat) **en rappelant dans la sous-requête la valeur d'une colonne de la requête principale.**

Exemple 1 : Afficher les employés dont le salaire est supérieur à la moyenne des salaires de leurs départements.

--> La condition de la clause WHERE doit être spécifique à chaque employé, dans la mesure où chaque employé a un département, et chaque département a une moyenne de salaires différente.

```
SELECT *  
FROM emp e  
WHERE sal > (SELECT AVG(sal)  
             FROM emp  
             WHERE e.deptno = deptno) ;
```

Compléments sur les requêtes de sélection.

L'exécution de cette requête se produit ainsi :

- 1- La requête principale fixe une ligne de la table EMP.
- 2- La valeur de e.deptno est sauvegardée dans une variable intermédiaire, la sous-requête est alors évaluée.
- 3- La condition de la clause WHERE de la requête principale est évaluée, la ligne est retournée ou non suivant la valeur de la condition.
- 4- Itération des étapes 1, 2 et 3 pour les lignes restantes de la table EMP.

Exemple 2 : Pour chacun des départements, afficher son nom, sa localisation, et le nombre de salariés qui y travaillent.

DNAME	LOC	Nombre de salaries
ACCOUNTING	NEW YORK	3
RESEARCH	DALLAS	5
SALES	CHICAGO	6
OPERATIONS	BOSTON	0

(Autre solution à tester : passer par un GROUP BY.)

Exemple 3 : Pour chaque manager, afficher son nom, sa fonction, ainsi que le nombre de ses subordonnés.

NOM	FONCTION	Nombre de subordonnes
BLAKE	PLEADER	5
CLARK	PLEADER	1
FORD	ANALYST	1
JONES	PLEADER	2
KING	PRESIDENT	3
SCOTT	ANALYST	1

Exemple 4 : Pour tous les départements comptant plus de 3 salariés, afficher le nom du département, sa localisation ainsi que la moyenne des salaires versés.