

BD01 et WGBD : Travaux pratiques 6

La clause HAVING

Attention, pour des raisons d'optimisation, MySQL ne génère pas d'erreur pour certaines instructions (fausses) de ce TP, et pire encore fournit un résultat dénudé de sens. Tapez l'instruction ci-dessous afin de générer les messages d'erreur :

```
SET GLOBAL sql_mode = "ONLY_FULL_GROUP_BY,STRICT_ALL_TABLES";
```

Avant de commencer ce laboratoire, lisez l'introduction tout en essayant chaque requête SQL.

La clause having

Il est possible d'éliminer des groupes obtenus par une requête avec regroupement.

La clause HAVING détermine une condition qui s'applique au groupe.

La condition est évaluée pour chaque groupe. Les groupes qui ne satisfont pas à la condition sont éliminés.

Exemples.

- 1) On veut connaître les identifiants de produits dont le total des quantités livrées (tous fournisseurs confondus) est strictement supérieur à 1250.

```
select id_p from spj group by id_p having sum(qty) > 1250
```

```
ID_P
P3
P4
P6
```

- 2) On veut connaître le nombre total des pièces fournies (toutes pièces confondues) par les fournisseurs dont l'identifiant est plus grand que S3.

```
select id_s, sum(qty) from spj group by id_s having id_s > 'S3'
```

```
ID_S    SUM
S4       600
S5      3100
```

La norme SQL indique que les conditions de la clause HAVING doivent porter soit sur des champs de la clause GROUP BY, soit sur une fonction d'agrégation.

```
select id_s, count(*) from spj group by id_s having count(*) > 2
```

ID_S	COUNT
S2	8
S5	10

Instruction correcte :

La clause having porte sur une fonction d'agrégation.

```
select id_s,id_p,count(*) from spj group by id_s,id_p having id_p > 'P3'
```

ID_S	ID_P	COUNT
S2	P5	1
S3	P4	1
S4	P6	2
S5	P4	1
S5	P5	3
S5	P6	2

Instruction correcte :

La clause having porte sur un champ présent dans le GROUP BY.

```
select id_s, count(*) from spj group by id_s having sum(qty) > 800
```

ID_S	COUNT
S1	2
S2	8
S5	10

Instruction correcte :

La clause having porte sur une fonction d'agrégation.

```
select id_s, count(*) from spj group by id_s having qty > 200
```

ID_S	COUNT
S1	2
S3	2
S4	2
S5	10

Instruction incorrecte

La clause having ne porte pas sur une fonction d'agrégation ou un champ de la clause GROUP BY. (plusieurs valeurs de qty existent dans un groupe)

Remarque : lorsque la condition de sélection ne porte pas sur une fonction d'agrégation mais **porte sur un des champs de la clause GROUP BY**, on peut alors indifféremment l'exprimer soit dans la clause where, ou dans la clause having.

```
select id_s, sum(qty) from spj group by id_s having id_s > 'S3'
est identique à
select id_s, sum(qty) from spj where id_s > 'S3' group by id_s
```

Exercices

1) Obtenir les projets d'Athens qui ont utilisé au total plus de 1000 pièces

```
ID_J    SUM
J4      3300
```

2) On souhaite obtenir l'identifiant des pièces et l'identifiant des projets lorsque la moyenne des quantités livrées (de cette pièce à ce projet) est supérieure à 320

```
ID_P  ID_J
P1     J4
P3     J4
P3     J5
P3     J6
P3     J7
P4     J2
P4     J4
P5     J4
P5     J5
P6     J4
```

3) On souhaite obtenir l'identifiant des fournisseurs qui ont livré la même pièce à au moins trois projets différents.

```
ID_S
S2
S5
```

4) On souhaite obtenir les pièces qui ont été livrées par au moins deux fournisseurs différents.

```
ID_P
P1
P3
P4
P5
P6
```

5) On souhaite obtenir les pièces qui ont été livrées à un même projet par au moins deux fournisseurs différents.

```
ID_P
P1
P3
```

6) On veut, pour chaque pièce, la somme totale livrée à condition qu'il y ait eu plus de 3 livraisons pour cette pièce.

```
ID_P    SUM
P3      3500
P5      1100
P6      1300
```