

## BD01 et WSGBD : Travaux pratiques 5

### Agrégation et partitionnement

**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.**

### **Les calculs et les fonctions d'agrégation.**

SQL nous permet de faire des calculs sur des tables (rappel : un produit cartésien ou une jointure de tables est aussi une table).

- Il est possible de faire une opération qui affecte l'ensemble des contenus d'une colonne. Les opérateurs mis à notre disposition sont +, -, \*, /.  
Exemple : on souhaite afficher la table SPJ avec 500 unités de plus pour chaque commande effectuée.

```
SELECT id_s, id_p, id_j, qty+500 FROM spj
```

- Il est également possible de compter le nombre de lignes d'une table (pas le nombre de colonnes), la moyenne d'une colonne, la somme d'une colonne, de fournir la valeur minimale ou maximale parmi les contenus d'une colonne.

Pour ce faire, les fonctions mises à notre disposition sont :

AVG(nom\_de\_colonne) retourne la moyenne de la colonne

AVG(DISTINCT nom\_de\_colonne) retourne la moyenne des lignes qui ont des contenus différents pour nom de colonne.

COUNT(\*) ou COUNT(nom\_de\_colonne) retourne le nombre de lignes

COUNT(DISTINCT nom\_de\_colonne) retourne le nombre de lignes qui ont des contenus différents pour nom de colonne.

MIN(nom\_de\_colonne) retourne la valeur minimale d'une colonne

MAX(nom\_de\_colonne) retourne la valeur maximale d'une colonne

SUM(nom\_de\_colonne) retourne la somme d'une colonne.

SUM(DISTINCT nom\_de\_colonne) retourne la somme des lignes de contenus différents pour nom de colonne.

Ces fonctions s'appellent des **fonctions d'agrégation**, car elles réduisent un **ensemble de valeurs à une valeur unique**.

Exemple : soit la table SPJ suivante :

ID_S	ID_P	ID_J	QTY	DATE_DERNIERE_LIVRAISON
=====	=====	=====	=====	=====
S1	P1	J1	200	5/10/2001
S1	P1	J4	700	10/05/2001
S2	P3	J1	400	20/05/2001
S2	P3	J2	200	30/07/2000
S2	P3	J3	200	10/05/2001
S2	P3	J4	500	3/10/2001
S2	P3	J5	600	20/09/2001
S2	P3	J6	400	12/05/2000
S2	P3	J7	800	23/08/2001
S2	P5	J2	100	23/06/2000
S3	P3	J1	200	7/07/2001
S3	P4	J2	500	18/05/2001
S4	P6	J3	300	10/05/2001
S4	P6	J7	300	16/09/2001
S5	P2	J2	200	10/11/2001
S5	P2	J4	100	17/04/2001
S5	P5	J5	500	8/02/2001
S5	P5	J7	100	25/06/2001
S5	P6	J2	200	9/02/2001
S5	P1	J4	100	18/03/2000
S5	P3	J4	200	19/05/2001
S5	P4	J4	800	10/05/2001
S5	P5	J4	400	16/12/2001
S5	P6	J4	500	10/10/2001

1) Obtenir le nombre de lignes de la table SPJ

```
select count(*)
from spj
```

2) Obtenir la moyenne des quantités vendues

```
select avg(qty)
from spj
```

3) Obtenir la plus petite et la plus grande quantité livrée

```
select min(qty), max(qty)
from spj
```

4) Obtenir la somme de toutes les quantités vendues

```
select sum(qty)
from spj
```

5) Obtenir le nombre de fournisseurs dans SPJ.

```
select count(distinct id_s)
from spj
```

## La clause where et les fonctions d'agrégation.

- L'évaluation d'une fonction d'agrégation s'effectue **après** avoir appliqué la clause where si elle est présente.

La clause where permet donc d'éliminer des lignes de la table AVANT d'effectuer une fonction d'agrégation

Exemples :

1) Obtenir le nombre de lignes de la table SPJ qui concerne la pièce 'P2'

```
select count(*) from spj where id_p='P2'
```

COUNT  
2

2) Obtenir la moyenne des quantités vendues par le fournisseur 'S1'

```
select avg(qty) from spj where id_s='S1'
```

AVG  
450

3) Obtenir la plus petite et la plus grande quantité livrée au projet 'P5'

```
select min(qty), max(qty) from spj where id_p='P5'
```

MIN                  MAX  
100                  500

4) Obtenir la somme de toutes les quantités vendues par le fournisseur S5

```
select sum(qty) from spj where id_s='S5'
```

SUM  
3100

## Les fonctions d'agrégation sur des regroupements de lignes.

- Les fonctions d'agrégation peuvent s'appliquer sur des groupes de lignes de la table et non sur la table entière.

Il faut alors spécifier les conditions de regroupement dans une clause GROUP BY.

```
select id_s, sum(qty) from spj group by id_s
```

Les champs de la clause group by sont appelés facteur de regroupement.

Exemples :

1) Obtenir les sommes totale des articles tous confondu vendus par fournisseur

☛ `select id_s, sum(qty) from spj group by id_s`

ID_S	SUM
S1	900
S2	3200
S3	700
S4	600
S5	3100

2) Obtenir les sommes des articles vendus par fournisseur et par pièce

☛ `select id_s,id_p, sum(qty) from spj group by id_s,id_p`

ID_S	ID_P	SUM
S1	P1	900
S2	P3	3100
S2	P5	100
S3	P3	200
S3	P4	500
S4	P6	600
S5	P1	100
S5	P2	300
S5	P3	200
S5	P4	800
S5	P5	1000
S5	P6	700

## Lien entre la clause select et les facteurs de regroupement.

Le ou les facteurs de regroupement doivent former des groupes dans lesquels il n'y a **qu'une seule valeur** possible pour les champs du select.

Exemple : Soit un regroupement se faisant sur la quantité (en gras figure le groupe correspondant à la quantité 200)

ID_S	ID_P	ID_J	QTY	DATE_DERNIERE_LIVRAISON
<b>S1</b>	<b>P1</b>	<b>J1</b>	<b>200</b>	<b>5/10/2001</b>
S1	P1	J4	700	10/05/2001
S2	P3	J1	400	20/05/2001
<b>S2</b>	<b>P3</b>	<b>J2</b>	<b>200</b>	<b>30/07/2000</b>
<b>S2</b>	<b>P3</b>	<b>J3</b>	<b>200</b>	<b>10/05/2001</b>
S2	P3	J4	500	3/10/2001
S2	P3	J5	600	20/09/2001
S2	P3	J6	400	12/05/2000
S2	P3	J7	800	23/08/2001
S2	P5	J2	100	23/06/2000
<b>S3</b>	<b>P3</b>	<b>J1</b>	<b>200</b>	<b>7/07/2001</b>
S3	P4	J2	500	18/05/2001
S4	P6	J3	300	10/05/2001
S4	P6	J7	300	16/09/2001
<b>S5</b>	<b>P2</b>	<b>J2</b>	<b>200</b>	<b>10/11/2001</b>
S5	P2	J4	100	17/04/2001
S5	P5	J5	500	8/02/2001
S5	P5	J7	100	25/06/2001
<b>S5</b>	<b>P6</b>	<b>J2</b>	<b>200</b>	<b>9/02/2001</b>
S5	P1	J4	100	18/03/2000
<b>S5</b>	<b>P3</b>	<b>J4</b>	<b>200</b>	<b>19/05/2001</b>
S5	P4	J4	800	10/05/2001
S5	P5	J4	400	16/12/2001
S5	P6	J4	500	10/10/2001

`select count(id_s), qty from spj group by qty`  
est correct car chaque regroupement ne donnera qu'une seule ligne de résultat.

Pour le regroupement 200 on aura : 7 200



Par contre

`select id_s, qty from spj group by qty`  
est incorrect car il y a plusieurs id\_s pour un regroupement.

Pour le regroupement 200 on aura : S1,S2,S3,S5

`id_s` n'est ni un facteur de regroupement, ni une fonction d'agrégation

Afin de garantir cette unicité de valeur, tous les champs se trouvant dans la clause select sont soit des fonctions d'agrégation, soit des facteurs de regroupement se trouvant dans la clause group by.

Remarque, on peut avoir un facteur de regroupement non présent dans la clause select.

Exemple: `select count(id_s) from s group by status`



## EXERCICE 1.

Expliquez pourquoi **les instructions suivantes sont incorrectes.**

- a) `select id_p, sum(weight) from p group by color`
- b) `select id_s, sum(qty) from spj`
- c) `select id_p, weight from p group by id_p`



## EXERCICE 2

**1) Déterminez par quantité le nombre de livraisons effectuées**

Exemple : il y a eu 4 livraisons de 100 unités, 7 livraisons de 200 unités, etc...

**2) On souhaite connaître le nombre de livraisons de chaque fournisseur.**

Exemple : le fournisseur S2 a effectué 8 livraisons. Le fournisseur S3 a effectué 2 livraisons.

**3) On veut connaître le nombre de pièces fournies par projet. On souhaite le nom du projet et non pas son identifiant.**

Exemple : on a fourni 800 pièces au projet Sorter, 1200 pièces au projet Display.

**4) On veut connaître le poids de chaque livraison.**

ID_S	ID_P	ID_J	
S1	P1	J1	2400
S1	P1	J4	8400
S5	P1	J4	1200
Etc...	...	...	...

**5) On veut connaître le poids total des pièces livrées par projet.**

Exemple : on a livré pour 12600 livres au projet j1 (cela provient de  $200 \times 12 + 400 \times 17 + 200 \times 17$ )  
 18800 livres au projet j2  
 etc...

**6) On veut connaître le poids de « la livraison la plus lourde » ?**

Réponse : 13600

**7) Pour chaque pièce fournie à un projet, obtenir le numéro de pièce, le numéro de projet et la quantité totale fournie (de cette pièce à ce projet) dont l'intitulé sera « total ».**

ID_J	ID_P	TOTAL
J1	P1	200
J4	P1	800
J2	P2	200
J4	P2	100
J1	P3	600
J2	P3	200
J3	P3	200
J4	P3	700
J5	P3	600
J6	P3	400
J7	P3	800
J2	P4	500
J4	P4	800

