

Bases de Données Relationnelles

Partie 3 : SQL – Jointures externes, agrégats et groupement

Master 2 CGSI

CY Cergy Paris Université

Tianxiao.Liu@cyu.fr

Plan : séance 3

- Jointures externes
 - Jointure externe gauche
 - Jointure externe droite
- Expression des agrégats
- Regroupement du résultat d'agrégats

Jointures externes 1/5

- Principe
 - Les lignes dans une des deux tables (ou bien toutes les deux) qui ne satisfont pas au prédicat seront également comprises dans le résultat
- Deux types de jointures externes
 - Jointure externe gauche (LEFT OUTER JOIN)
 - Jointure externe droite (RIGHT OUTER JOIN)
- Possible de combiner avec les expressions
 - **IS NULL** et **IS NOT NULL**
 - → Voir l'exercice en TD

Jointures externes 2/5

- Jointure externe gauche
 - Principe : toutes les lignes de la **première** table seront comprises dans le résultat
 - Syntaxe

```
SELECT ...
```

```
FROM nom_table1 AS alias1 LEFT OUTER JOIN  
     nom_table2 AS alias2
```

```
ON (alias1.colonne1 = alias2.colonne2);
```

Jointures externes 3/5

- Jointure externe gauche (suite)

- Exemples

```
SELECT *  
FROM client AS c LEFT OUTER JOIN  
    commande AS co  
    ON (c.id_client = co.id_client);
```

Cette requête liste tous les clients avec les informations de leurs commandes, y compris aussi les clients n'ayant effectué aucune commande.

Jointures externes 4/5

- Jointure externe droite
 - Principe : toutes les lignes de la **deuxième** table seront comprises dans le résultat
 - Syntaxe

```
SELECT ...
```

```
FROM nom_table1 AS alias1 RIGHT OUTER JOIN  
    nom_table2 AS alias2  
    ON (alias1.colonne1 = alias2.colonne2);
```

Jointures externes 5/5

- Jointure externe droite (suite)

- Exemples

```
SELECT *  
FROM commande AS co RIGHT OUTER JOIN  
    voiture AS v  
    ON (co.id_voiture = v.id_voiture);
```

Cette requête liste toutes les commandes et les informations des voitures commandées, mais aussi les voitures jamais commandées.

Expression des agrégats 1/2

- Objectif
 - Effectuer un calcul sur les valeurs d'une colonne d'ensemble de lignes
- Calculs possibles
 - La somme : **SUM**
 - La moyenne : **AVG**
 - La valeur minimale : **MIN**
 - La valeur maximale : **MAX**
 - Le comptage de lignes : **COUNT**
- Syntaxe
 - `SELECT expression d'agrégat`
 - ...

Expression des agrégats 2/2

- Exemples

```
SELECT SUM(co.quantite) FROM commande AS co;
```

```
SELECT MIN(co.quantite) FROM commande AS co;
```

```
SELECT AVG(co.quantite)
FROM commande AS co INNER JOIN voiture AS v
      ON (co.id_voiture = v.id_voiture)
WHERE v.marque = 'Peugeot';
```

Cette requête calcule la moyenne des quantités par commande de voitures "Peugeot".

Groupement des résultats 1/3

- Objectif
 - Calculer les **agrégats** pour chaque groupe de données
 - Le groupement se fait sur la valeur d'une colonne

- Syntaxe

```
SELECT expression d'agrégat  
FROM nom_table AS alias  
WHERE ...  
GROUP BY alias.colonne  
HAVING condition (qui contient l'expression d'agrégat)
```

La clause HAVING est facultative.

Groupement des résultats 2/3

- Exemple

```
SELECT MIN(co.quantite)
FROM commande AS co
GROUP BY co.id_client;
```

Cette requête retrouve la minimale quantité de voitures commandées pour chaque client. On peut ajouter la colonne utilisée par le groupement dans la projection, ceci rend le résultat plus compréhensible :

```
SELECT co.id_client, MIN(co.quantite)
FROM commande AS co
GROUP BY co.id_client;
```

Groupement des résultats 3/3

- Exemple

```
SELECT v.marque, SUM(co.quantite)
FROM commande AS co INNER JOIN voiture AS v
      ON (co.id_voiture = v.id_voiture)
GROUP BY v.marque
HAVING SUM(co.quantite) >= 20;
```

Cette requête affiche la somme des voitures commandées, groupées par marque. Seulement les marques avec une somme supérieure ou égale à 20 sont affichées dans le résultat.