



M114 – CM6

SQL – Agrégation & Vues

Denis PALLEZ
<http://denispallez.i3s.unice.fr>

Select – Agrégation

- Effectuer des calculs (fonctions d'agrégation) sur un groupe de tuples
 - Agrégation → Agréger ≈ Regrouper = Partitionner
- Exemple
 - Affichage des moyennes des compteurs par marque de voiture
- Étapes
 - Définir les groupes
 - Comment partitionner la relation?
 - Identifier les fonctions d'agrégation sur ces groupes
 - Ajouter éventuellement des conditions sur ces groupes
 - Que les moyennes > 10000 km par exemple
- Exception
 - Il existe déjà 1 groupe par défaut : la table toute entière
⇒ Possible d'utiliser des fonctions d'agrégation sans définir de groupe

2015-2016 M114 – Intro. BD – <http://denispallez.i3s.unice.fr> 96

Select – Fonctions d'agrégation

- Opèrent sur un attribut (souvent numérique)
 - COUNT nombre de valeurs
 - SUM somme des valeurs
 - AVG moyenne des valeurs
 - MAX plus grande valeur
 - MIN plus petite valeur

2015-2016 M114 – Intro. BD – <http://denispallez.i3s.unice.fr> 97

Select – Fonctions d'agrégation

```

Client (ID, Nom, Ville)
Achat(ID#, Ref#, Immatriculation, Date)
Voiture (Ref, Marque, Modele, Serie, Prix)
  
```

- Exemples
 - Donner le nombre de clients


```

Select COUNT (id)
From Client ;
          
```
 - Donner le nombre de clients qui ont acheté une voiture


```

Select Count (*)
From Client ;
          
```
 - Donner le chiffre d'affaires des ventes de 2015


```

Select SUM(Prix)
From Achat A Inner Join Voiture V On (A.Ref = V.Ref)
Where Year(Date) = 2015 ;
          
```
 - Marque et Prix des voitures moins cher que la moyenne des voitures


```

Select Marque, Prix
From Voiture
Where Prix < (
          
```

2015-2016 M114 – Intro. BD – <http://denispallez.i3s.unice.fr> 98

Select – Group by

- Sert à définir les groupes
 - Partitionne la relation
- Fonctions d'agrégation appliquée à chaque groupe
- Des filtres de sélection peuvent être appliqués AVANT et / ou APRES le groupement

Select – Group by

SELECT Concession,
SUM(CA) AS Somme
FROM ChiffreA
GROUP BY Concession ;

DateCA	Concession	CA
15/11/2006	Nord	900
15/11/2006	Paris	1500
15/11/2006	Sud	850
16/11/2006	Nord	1200
16/11/2006	Paris	900
16/11/2006	Sud	800
17/11/2006	Nord	800
17/11/2006	Paris	1000
17/11/2006	Sud	550

Concession	Somme
Nord	2900
Paris	3400
Sud	2200

Select – Group by

SELECT DateCA,
SUM(CA) AS Somme
FROM ChiffreA
GROUP BY DateCA ;

DateCA	Concession	CA
15/11/2006	Nord	900
15/11/2006	Paris	1500
15/11/2006	Sud	850
16/11/2006	Nord	1200
16/11/2006	Paris	900
16/11/2006	Sud	800
17/11/2006	Nord	800
17/11/2006	Paris	1000
17/11/2006	Sud	550

DateCA	Somme
15/11/2006	3250
16/11/2006	2900
17/11/2006	2350

Select – Group by

- 1 observation ?
 - SELECT Concession, SUM(CA) AS Somme
FROM ChiffreA
GROUP BY Concession;
 - SELECT DateCA, SUM(CA) AS Somme
FROM ChiffreA
GROUP BY DateCA;

Select – Group by

- 1 observation ?
 - `SELECT Concession, SUM(CA) AS Somme`
`FROM ChiffreA`
`GROUP BY Concession;`
 - `SELECT DateCA, SUM(CA) AS Somme`
`FROM ChiffreA`
`GROUP BY DateCA;`
- Contrainte
 - Dans `GROUP BY`, on retrouve tous les attributs du `SELECT` sauf les fonctions d'agrégations

Select – Groupement & Filtrage

- Possible de « filtrer » les données AVANT et/ou APRES un regroupement
 - AVANT
 - Grâce à la clause `WHERE`
 - Restrictions sur les enregistrements
 - APRES
 - Grâce à la clause `HAVING`
 - Restrictions sur les groupes créés par `group by`
 - Syntaxe


```
SELECT      champ1, opération(champ2)
FROM        nomTable1
WHERE       condition de filtrage "avant groupement"
GROUP BY    champ1
HAVING      condition de filtrage "après groupement"
```

Select – Groupement & Filtrage

- Filtrage AVANT


```
SELECT  Concession, SUM(CA) AS Somme
FROM    ChiffreA
WHERE   CA >= 900
GROUP BY Concession ;
```

ChiffreA

DateCA	Concession	CA
15/11/2006	Nord	900
15/11/2006	Paris	1500
15/11/2006	Sud	850
16/11/2006	Nord	1200
16/11/2006	Paris	900
16/11/2006	Sud	800
17/11/2006	Nord	800
17/11/2006	Paris	1000
17/11/2006	Sud	550

DateCA	Concession	CA
15/11/2006	Nord	900
15/11/2006	Paris	1500
15/11/2006	Sud	850
16/11/2006	Nord	1200
16/11/2006	Paris	900
16/11/2006	Sud	800
17/11/2006	Nord	800
17/11/2006	Paris	1000
17/11/2006	Sud	550

Concession	Somme
Nord	2100
Paris	3400

DateCA	Concession	CA
15/11/2006	Nord	900
16/11/2006	Nord	1200
15/11/2006	Paris	1500
16/11/2006	Paris	900
17/11/2006	Paris	1000

Select – Groupement & Filtrage

- Filtrage APRÈS


```
SELECT  Concession, SUM(CA) AS Somme
FROM    ChiffreA
GROUP BY Concession
HAVING  SUM(CA) >= 2500 ;
```

ChiffreA

DateCA	Concession	CA
15/11/2006	Nord	900
15/11/2006	Paris	1500
15/11/2006	Sud	850
16/11/2006	Nord	1200
16/11/2006	Paris	900
16/11/2006	Sud	800
17/11/2006	Nord	800
17/11/2006	Paris	1000
17/11/2006	Sud	550

DateCA	Concession	CA
15/11/2006	Nord	900
16/11/2006	Nord	1200
17/11/2006	Nord	800
15/11/2006	Paris	1500
16/11/2006	Paris	900
17/11/2006	Paris	1000
15/11/2006	Sud	850
16/11/2006	Sud	800
17/11/2006	Sud	550

Concession	Somme
Nord	2900
Paris	3400
Sud	2200

Concession	Somme
Nord	2900
Paris	3400
Sud	2200

Select – Groupement & Filtrage

- Filtrage Avant ET Après

```
SELECT Concession, SUM(CA) AS Somme
FROM ChiffreA
WHERE CA >= 900
GROUP BY Concession
HAVING SUM(CA) >= 3000 ;
```

ChiffreA

DateCA	Concession	CA
15/11/2006	Nord	900
15/11/2006	Paris	1500
15/11/2006	Sud	850
16/11/2006	Nord	1200
16/11/2006	Paris	900
16/11/2006	Sud	800
17/11/2006	Nord	800
17/11/2006	Paris	1000
17/11/2006	Sud	550

DateCA	Concession	CA
15/11/2006	Nord	900
15/11/2006	Paris	1500
15/11/2006	Nord	1200
16/11/2006	Paris	1500
16/11/2006	Nord	1200
16/11/2006	Paris	900
17/11/2006	Paris	1000

Concession	Somme
Paris	3400

Concession	Somme
Nord	2400
Paris	3400

15/11/2006	Nord	900
16/11/2006	Nord	1200
15/11/2006	Paris	1500
16/11/2006	Paris	900
17/11/2006	Paris	1000

Select – Groupement

- Noms des clients qui ont acheté TOUTES les Ferraris?
 - Clients qui ont acheté un nombre distinct de Ferraris égales au nb total de Ferraris (version sans groupement)

```
Select Nom, Id
From Achat A Inner Join Client C ON (A.ID=C.ID)
Where Marque = 'Ferrari'
Group by Nom, Id
Having Count(Distinct Ref) = (
    Select count(*)
    From Voiture
    Where Marque= 'Ferrari'
);
```

Vues

- Vue = Table Virtuelle = Requête SQL stockée
- Syntaxe


```
Create View <nom_vue> AS
Select ...
```
- Exemple


```
Voiture (Ref, Marque, Modele, Serie, Prix)
Create View Ferraris As
Select Ref, Marque, Modele from Voiture
Where Marque = 'Ferrari' ;
```
- Intérêts
 - Vue = comme une autre table


```
Select * From Ferraris Where Prix < 50000 ;
```

 - Ré-évaluée avant toute utilisation
 - Simplifier l'accès à certaines informations
 - Cacher des données sensibles

MAJ Vues

- Objectif
 - Mettre à jour la table support à une vue
- Exemple


```
Insert into Ferraris (17235, 'Ferrari', 'F250',
300000) ;
```
- Restrictions
 - Vue basée sur une seule table
 - Toute colonne non référencée dans la vue doit pouvoir être mise à NULL ou disposer d'une valeur par défaut
 - Attribut Serie dans Ferraris
 - On ne peut mettre à jour un attribut résultat d'un calcul
 - L'option WITH CHECK OPTION permet de garantir que toute ligne insérée dans la vue satisfait la requête définissant la vue