

Base de données – Agrégation

On utilise la base de données Notes

I. Besoins

On cherche à utiliser les données de la base pour faire des statistiques.

Cela nécessite d'extraire les données (exemple : étudiants, notes)
de regrouper les données à traiter dans un ensemble (exemple : notes d'un étudiant)
de leur appliquer des fonctions (exemple : moyenne)

Extraction des données : vu, requêtes de sélection et jointures

Regrouper les données : en langage SQL : agrégation de données

Fonctions : fonctions incluses dans le langage MIN, MAX, SUM, AVG, COUNT

II. Table de démonstration

Table Colles...

IdColl	NomEt	Note
1	Martine	11
2	Cynthia	11
3	Alphonse	8
4	Myrtille	15
5	Nour	14
6	Aisha	10
7	Tim	12
8	Martine	10
9	Nour	15
10	Myrtille	14
11	Alphonse	10
12	Martine	13
13	Cynthia	13

III. Principe de l'agrégation

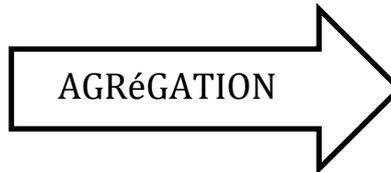
Les fonctions doivent être appliquées à un ensemble de données regroupées d'après une référence commune. Dans l'exemple : on cherche à traiter toutes les notes de colle de chaque étudiant.

On va donc *agrég* toutes les notes (données) d'après les noms des étudiants (référence : nom de l'étudiant). On définit un *agrégat* auquel on applique la fonction.

Agréger les valeurs de l'attribut *Note* sur l'attribut *NomEt* consiste en la définition de sous-ensembles dans la table pour leur appliquer un calcul.

La table intermédiaire avec les sous-ensembles n'apparaît pas on code le calcul et on affiche directement le résultat

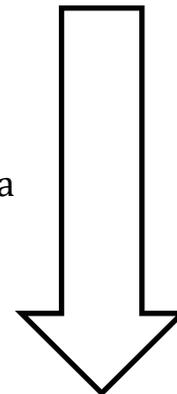
IdColl	NomEt	Note
1	Martine	11
2	Cynthia	11
3	Alphonse	8
4	Myrtille	15
5	Nour	14
6	Aisha	10
7	Tim	12
8	Martine	10
9	Nour	15
10	Myrtille	14
11	Alphonse	10
12	Martine	13
13	Cynthia	13



NomEt	Note
Aisha	10
Alphonse	8
Alphonse	10
Cynthia	11
Cynthia	13
Martine	11
Martine	10
Martine	13
Myrtille	15
Myrtille	14
Nour	14
Nour	15
Tim	12

Ce tableau n'apparaît pas : c'est un intermédiaire du calcul

Calcul sur les agrégats (formule avec la fonction, ici : moyenne)



NomEt	Moyenne
Aisha	10
Alphonse	9
Cynthia	12
Martine	11,3333
Myrtille	14,5
Nour	14,5
Tim	12

IV. Syntaxe

Dans une requête, on définit :

- la sélection (**SELECT**) avec la forme de la table souhaitée et la fonction choisie
- la référence pour l'agrégation.

Exemple :

```
SELECT NomEt, AVG(Colles.Note)
FROM Colles
GROUP BY NomEt;
```

Renvoie

NomEt	AVG(Colles.Note)
Aisha	10
Alphonse	9
Cynthia	12
Martine	11,333333
Myrtille	14,5
Nour	14,5
Tim	12

On note :

Structure

- la nature des agrégats est défini en dernier grâce à l’instruction **GROUP BY**
- le calcul effectué sur chaque agrégat est défini dans la sélection, ici **AVG()**

Attributs

- les agrégats sont déterminés par les valeurs d’un attribut, ici **NomEt**
- le calcul est effectué sur un autre attribut, ici **Note** de la table **Colles**

V. Fonctions

MIN() : renvoie la plus petite valeur de l’agrégat

MAX() : renvoie la plus grande valeur de l’agrégat

AVG() : renvoie la moyenne des valeurs (numériques) de l’agrégat

SUM() : renvoie la somme des valeurs (numériques)

VI. Applications

Appeler la liste des noms des étudiants et de leurs notes minimales, leurs notes maximales et leurs moyennes.

```
SELECT NomEt, MIN(Colles.Note), , MAX(Colles.Note) , AVG(Colles.Note)
FROM Colles
GROUP BY NomEt;
```

Variante : appeler les mêmes informations mais en renommant les attributs en « Prénom », « +basse », « +haute », « Moyenne ».

```
SELECT NomEt AS "Prénom", MIN(Colles.Note) AS "+basse", , MAX(Colles.Note) ,AS "+haute"
AVG(Colles.Note) AS "Moyenne"
FROM Colles
GROUP BY NomEt;
```

Avec des jointures...

a. Appeler une table donnant les nom de classe, prénom des étudiants, notes, pour toutes les colles

```
SELECT NomClass, NomEt, Note
FROM Colles JOIN Etudiants ON NomEt=Nom JOIN Classes ON IdClass=Classe
```

b. Appeler une table avec nom de classe, prénoms, note la plus haute, note la plus basse pour chaque étudiant

```
SELECT NomClass, NomEt, MAX(Note), MIN(Note)
FROM Colles JOIN Etudiants ON NomEt=Nom JOIN Classes ON IdClass=Classe
GROUP BY NomEt
```

c. Appeler la liste donnant la moyenne de chacune des classes sous forme d'un tableau avec le nom de classe, la moyenne de cette classe.

```
SELECT NomClass, AVG(Note)
FROM Colles JOIN Etudiants ON NomEt=Nom JOIN Classes ON IdClass=Classe
GROUP BY NomClass
```

V. COUNT

La fonction **COUNT()** renvoie le nombre de lignes dans une table. Plus précisément :

COUNT(*) renvoie le nombre de lignes de la table associée à la requête

Exemple :

```
SELECT COUNT(*)
FROM Etudiants
```

Si att est un attribut d'une table : **COUNT(att)**° renvoie le nombre de cases non vides associées. (Dans beaucoup de tables, les cases sont non vides et cela donne le même résultat que COUNT(*))

Exemple :

```
SELECT COUNT(Etudiants.nom)
FROM Etudiants
```

Si att est un attribut d'une table : **COUNT(DISTINCT att)** renvoie le nombre de valeurs non nulles et différentes entre elles associées à l'attribut att.

Exemple

```
SELECT COUNT(DISTINCT Etudiants.Classe)
FROM Etudiants
```

En combinant avec l'instruction WHERE, on peut compter des enregistrements vérifiant une condition...

Application :

Appeler le nombre de notes au-dessus de la moyenne en colle.

Exemple :

```
SELECT COUNT(*)
FROM Colles
WHERE Note >=10
```

Appeler le nombre de notes au-dessus de la moyenne en colle, en TB1.

```
SELECT COUNT(*)
FROM Colles JOIN Etudiants ON NomEt=Nom
WHERE Note >=10 AND Classe =1
```

FIN