



**Département d'informatique**

**DICJ**

**Les bases de données  
DML – Data  
Manipulation Language**

420-JQA-JQ

Fonctions d'aggrégation

Regroupement de données

GROUP BY

# Modifier les noms de colonnes

Il y a trois méthodes:

```
SELECT NoClient 'Numéro de Client'  
FROM   tblClient;
```

```
SELECT NoClient AS " Numéro de Client"  
FROM   tblClient;
```

```
SELECT "NoClient" = Numéro de Client  
FROM   tblClient;
```

Numéro de client

B062  
B112  
B332  
B512  
C003  
C123  
C400  
D063  
F010  
F011  
F400  
K111  
K729  
L422  
S127  
S712

# Extraction simple IN

- IN: représente un ensemble de valeur.
- Not in: exclut un ensemble de valeur.

```
SELECT NoClient
FROM   tblClient
WHERE  CatClient IN ('C1','C2','C3');
```

```
SELECT NoClient
FROM   tblClient
WHERE  LocaliteClient NOT IN('Turin','Breda');
```

# Les fonctions d'agrégation

- Vous pouvez synthétiser toutes les données d'une table en créant une requête d'agrégation impliquant une fonction.
- Lorsque vous exécutez la requête, le jeu de résultats contient une seule ligne, ou une ligne par groupe, avec les informations de synthèse.

Fonction	Description
AVG (expr)	Moyenne des valeurs d'une colonne ou d'un groupe de lignes. La colonne ne peut contenir que des valeurs numériques.
COUNT(expr) Ou COUNT(*)	Nombre de valeurs dans une colonne ou nombre de lignes dans une table ou un groupe (si vous spécifiez *). COUNT(expr) ignore les valeurs NULL, mais COUNT(*) les inclut.
MAX(expr)	La plus grande valeur d'une colonne ou d'un groupe de lignes.
MIN(expr)	La plus petite valeur d'une colonne ou d'un groupe de lignes.
SUM(expr)	Total des valeurs d'une colonne ou d'un groupe de ligne. La colonne ne peut contenir que des valeurs numériques.

# EXAMPLES

```
SELECT *  
FROM tblGens
```

TblGens		
Nom	Prenom	Age
Dupond	Pierre	24
Martin	Marc	48
Dupont	Jean	51
Martin	Paul	36
Dupond	Lionel	68
Chirac	Jacques	70

```
SELECT AVG(Age) AS 'Moyenne'  
FROM tblGens
```

Moyenne
49

```
SELECT MAX(Age) 'MAX'  
FROM tblGens
```

MAX
70

```
SELECT AVG(Age) AS 'Moyenne'  
FROM tblGens  
WHERE Nom='Dupond'
```

Moyenne
46

# GROUP BY

- Permet le regroupement des données ayant la même valeur.
- S'emploie avec l'instruction SELECT pour classer des données identiques par groupe.
- La clause Group by vient après la clause WHERE et précède la clause ORDER BY.

```
SELECT  
FROM  
JOIN  
WHERE  
GROUP BY  
ORDER BY
```

- Ce qui est trié doit être une colonne nommée dans le GROUP BY.
- Si une colonne ne se trouve pas dans l'instruction SELECT, elle ne peut pas être utilisée dans la clause GROUP BY

# Les données groupées (Group By)

LOCALITES			
NCLI	NOM	LOCALITE	COMPTE
F400	JACOB	Bruxelles	0
B332	MONTI	Genève	0
K111	VANBIST	Lille	720
S127	VANDERKA	Namur	-4580
L422	FRANCK	Namur	0
C123	MERCIER	Namur	-2300
B062	GOFFIN	Namur	-3200
S712	GUILLAUME	Paris	0
F010	TOUSSAINT	Poitiers	0
B112	HANSENNE	Poitiers	1250
C400	FERARD	Poitiers	350
C003	AVRON	Toulouse	-1700
B512	GILLET	Toulouse	-8700
F011	PONCELET	Toulouse	0
K729	NEUMAN	Toulouse	0
D063	MERCIER	Toulouse	-2250

→ le groupe des clients de Genève

→ le groupe des clients de Namur

→ le groupe des clients de Poitiers

# Un exemple avec fonctions

```
SELECT Localite,  
       COUNT(*) AS 'Nombre Clients',  
       AVG(Compte) AS 'Moyenne Compte'  
FROM   tblClient  
GROUP BY Localite;
```

Localite	Nombre Clients	Moyenne Compte
Bruxelles	1	0.00
Geneve	1	0.00
Lille	1	720.00
Namur	4	-2520.00
Paris	1	0.00
Poitiers	3	533.33
Toulouse	5	-2530.00

le groupe des clients de Genève

le groupe des clients de Namur

le groupe des clients de Poitiers

On s'intéresse aux **localités** et non plus aux **clients**



# HAVING

- La clause HAVING agit comme le filtre WHERE.
- Il permet de filtrer non plus les données, mais les opérations résultant des regroupements.

```
SELECT Localite, COUNT(*), AVG(Compte)
FROM   tblClient
GROUP BY Localite
HAVING COUNT(*) >= 3;
```

Localite	COUNT(*)	AVG(Compte)
Namur	4	-2520.00
Poitiers	3	533.33
Toulouse	5	-2530.00

# Avec et sans HAVING

- Afficher le prix moyen par type de livres.

```
SELECT TypeLivre, AVG(PrixLivre)
FROM   tblLivre
GROUP BY TypeLivre
```

	TypeLivre	(Aucun nom de c
1	cui_modeme	78,50
2	cui_traditio	108,6666
3	gestion	95,50
4	informatique	146,00
5	psychologie	92,40

- Même chose pour les types dont le prix moyen est supérieur à 95\$.

```
SELECT TypeLivre, AVG(PrixLivre)
FROM   tblLivre
GROUP BY TypeLivre
HAVING AVG(PrixLivre)>95
```

	TypeLivre	(Aucun nom de cc
1	cui_traditio	108,6666
2	gestion	95,50
3	informatique	146,00