

## Les bases de la consultation d'une source de données via un SGBD & R

Pour BTS AG PME-PMI (SGI & A5.3 + A7.1 du référentiel)

Nous avons appris la commande SELECT dans un précédent cours. Ci-dessous, vous trouverez un rappel dans ce document. Merci d'y jeter un œil et d'essayer de comprendre en vous imprégnant des principes. Cela peut vous paraître très abstrait, cependant, plus vous serez à l'aise avec cette partie de votre examen, et meilleure seront vos chances de gagner du temps sur l'épreuve, donc des points !

Source (Developpez.com)

**SELECT** : Permet de sélectionner une ou plusieurs lignes d'une table dans une base de données ;

```
SELECT PRP_ID, PRP_CODE_TITRE,  
PRP_NOM, PRP_PRENOM  
FROM T_PROSPECT
```

	PRP_ID	PRP_CODE_TITRE	PRP_NOM	PRP_PRENOM
	1	M.	Dupont	Alain
	2	Mme.	Durand	Aline
	3	M.	Dubois	NULL

Seconde étape, Il faut maintenant insérer ces données dans les colonnes CLI\_ID, TIT\_CODE, CLI\_NOM, CLI\_PRENOM correspondantes de la table T\_CLIENT. Celle-ci contient :

```
SELECT CLI_ID, TIT_CODE, CLI_NOM, CLI_PRENOM  
FROM T_CLIENT
```

	CLI_ID	TIT_CODE	CLI_NOM	CLI_PRENOM
	1	M.	DUPONT	Alain
	2	M.	MARTIN	Marc
	3	M.	BOUVIER	Alain
	4	M.	DUBOIS	Paul
	5	M.	DREYFUS	Jean
	6	M.	FAURE	Alain
	7	M.	LACOMBE	Paul
	8	Melle.	DUHAMEL	Evelyne
	9	Mme.	BOYER	Martine
	10	M.	MARTIN	Martin

```
SELECT CLI_ID, TIT_CODE, CLI_NOM, CLI_PRENOM  
FROM T_CLIENT WHERE CLI_NOM='DUPONT'
```

Sélectionnez

	CLI_ID	TIT_CODE	CLI_NOM	CLI_PRENOM
	1	M.	DUPONT	Alain
	2	M.	MARTIN	Marc
	3	M.	BOUVIER	Alain
	4	M.	DUBOIS	Paul
	5	M.	DREYFUS	Jean
	6	M.	FAURE	Alain
	7	M.	LACOMBE	Paul
	8	Melle.	DUHAMEL	Evelyne
	9	Mme.	BOYER	Martine
	10	M.	MARTIN	Martin
	...			

La clause **Where** permet d'indiquer Où l'on veut chercher et Quoi « Le critère » de recherche !  
L'opérateur « = » est facile à retenir, mais il existe aussi d'autres opérateurs :  
< : Inférieur à

> : Supérieur à  
<> : Différent de  
>= : Supérieur ou égale à  
<= : Inférieur ou égale à

Avec la clause Where, on peut rajouter l'élément « **Like** » qui équivaut à « Comme » sous MS Excel et permet une recherche égale, ou générique souvent employée dans des champs de type TEXTE (VARCHAR, CHAR, BLOB...)

Vous disposez toujours de l'accès à vos comptes sur les exercices réalisés avec la création d'un site web de type WordPress.

Nous verrons dans un prochain TD les autres fonctions SQL et requêtes, mais en attendons, ayez la maîtrise de la clause Select.

<b>SELECT</b>	Spécification des colonnes du résultat
<b>FROM</b>	Spécification des tables sur lesquelles porte l'ordre
<b>WHERE</b>	Filtre portant sur les données (conditions à remplir pour que les lignes soient présentes dans le résultat)
<b>GROUP BY</b>	Définition d'un groupe (sous ensemble)
<b>HAVING</b>	Filtre portant sur les résultats (conditions de regroupement des lignes)
<b>ORDER BY</b>	Tri des données du résultat

Regroupement d'information:

Si je souhaite regrouper tous les DUPONT :

- **Select \* From T\_CLIENT Where Nom='DUPONT' Group By Cli\_Nom**

Pour ne sélectionner que certains champs et pas d'autres :

- **Select CLI\_NOM, CLI\_PRENOM From T\_CLIENT Where Nom='DUPONT' Group By Cli\_Nom**

Pour classer (ou regrouper) par nom de famille :

- **Select CLI\_NOM, CLI\_PRENOM From T\_CLIENT Group By CLI\_NOM**

Et si je souhaitais classer par Ville ?

- **Select CLI\_VILLE, CLI\_NOM, CLI\_PRENOM From T\_CLIENT Group By CLI\_VILLE**

Remarquer que pour réaliser un bon regroupement il est nécessaire d'avoir le champ qui doit être regroupé dans la commande select, à défaut, cela serait / provoquerait une erreur.

Trier :

Pour trier j'utilise la clause « Order By » !

En admettant que je souhaite trier par ordre alphabétique mes clients :

- **Select CLI\_NOM, CLI\_PRENOM From T\_CLIENT Order By CLI\_NOM ASC**

« ASC » dans la clause « Order By » indique un tri croissant

« DESC » dans la clause « Order By » indique un tri décroissant

Dans un champ de type « Date », il est important de savoir que si l'on veut obtenir la période la plus récente, on trierait par ordre décroissant ! Puis inversement.

On peut combiner « Group By » et Order By » :

Je souhaite regrouper les clients par nom et les classer dans l'ordre décroissant :

- **Select CLI\_NOM, CLI\_PRENOM From T\_CLIENT Group By Cli\_Nom Order By Desc**

L'usage de la fonction « Group By » entraine parfois l'usage de la fonction « Having », et souvent on la trouve accompagné de la fonction « SUM ».

**Exemple :** (source <http://sql.sh>)

Pour utiliser un exemple concret, imaginons une table « achat » qui contient les achats de différents clients avec le coût du panier pour chaque achat.

id	client	tarif	date_achat
1	Pierre	102	2012-10-23
2	Simon	47	2012-10-27
3	Marie	18	2012-11-05
4	Marie	20	2012-11-14
5	Pierre	160	2012-12-03

Si dans cette table on souhaite récupérer la liste des clients qui ont commandé plus de 40€, toutes commandes confondues alors il est possible d'utiliser la requête suivante :

```
SELECT client, SUM(tarif)
FROM achat
GROUP BY client
HAVING SUM(tarif) > 40
```

**Résultat :**

client	SUM(tarif)
Pierre	262
Simon	47

La cliente « Marie » a cumulée 38€ d'achat (un achat de 18€ et un autre de 20€) ce qui est inférieur à la limite de 40€ imposée par HAVING. En conséquent cette ligne n'est pas affichée dans le résultat.