

Commandes SQL pour base SQLite3

Sortons un peu de l'exemple SQL trouvé sur tous les programmes de base de données SQLite3 sous Gambas2:

```
sql = "SELECT * FROM machin WHERE id LIKE '%" & "" & "%'"
```

1) Première commande: **Ouvrir la totalité de la table**

```
sql = "SELECT * FROM machin"
```

machin étant le nom de notre table et * sélectionne l'ensemble des champs. Mais il n'est pas toujours nécessaire de sélectionner tous les champs.

Prenons notre exemple de base:

Nom de la base: **mabase**

2 tables que nous appelons: **PRIMAIRE** et **SECONDAIRE**

la table PRIMAIRE

CHAMPS de la table I

Id (notre compteur incrémenté)	INTEGER PRIMARY KEY
sexe	TEXT
nom	TEXT
taille	NUMERIC
ville	TEXT

La table SECONDAIRE

CHAMPS de la table II

Idsecond (compteur de la table II)	INTEGER PRIMARY KEY
Idprimaire (liaison à l'id de la table I)	NUMERIC
poids	NUMERIC
notes	TEXT

Pour ouvrir la totalité de la première table:

```
sql = "SELECT * FROM PRIMAIRE"
```

Mais nous pouvons ouvrir seulement les 2 champs nécessaires à notre programme par exemple:

```
sql = "SELECT nom, id FROM PRIMAIRE"
```

2) Ouvrir la base par **ordre alphabétique**:

```
sql = "SELECT nom, id FROM PRIMAIRE ORDER BY nom"
```

de manière **ascendante**:

```
sql = "SELECT nom, id FROM PRIMAIRE ORDER BY nom ASC"
```

ou de manière **descendante**:

```
sql = "SELECT nom, id FROM PRIMAIRE ORDER BY nom DESC"
```

3) Sélectionner une **ligne** dans une table:

par exemple

```
lindex = ListView1.key
```

```
sql = "SELECT * FROM PRIMAIRE WHERE id=" & lindex
```

4) Nous voulons uniquement sélectionner des champs **distincts** d'une liste: voir dans notre exemple la comboBox des villes:

```
sql = "SELECT DISTINCT ville FROM PRIMAIRE ORDER BY  
ville"
```

5) Sélectionner dans un champ les données **contenant** une lettre ou une partie de texte que nous écrivons dans une TextBox:

```
sql = "SELECT * FROM PRIMAIRE Where nom LIKE '%" &  
TextBox2.Text & "%' ORDER BY nom"
```

6) Sélectionner dans un champ les données **commençant par**:

```
sql = "SELECT * FROM PRIMAIRE Where nom LIKE '" &  
TextBox1.Text & "%' ORDER BY nom"
```

7) Sélectionner dans un champ les données **se terminant par**:

```
sql = "SELECT * FROM PRIMAIRE Where nom LIKE '%" &  
TextBox3.Text & "' ORDER BY nom"
```

8) Il est également possible de sélectionner **une liste d'une première table en faisant une requête sur la seconde table.**

Par exemple:

```
sql = "Select * FROM PRIMAIRE, SECONDAIRE Where  
SECONDAIRE.poids >= " & ComboBox3.Text & " AND  
SECONDAIRE.idprimaire = PRIMAIRE.id GROUP BY PRIMAIRE.nom  
ORDER By nom"
```

Nous listons ici les noms de la première table des personnes ayant un poids supérieur ou égal à une valeur rentrée dans la seconde table.

Aide SQL pour base SQLite3 avec deux tables liées

Henri
Pascal
Antoine
Loic
Marine
Loana
Louis
Marie
Océane
Martin

Nom: Henri
id: 1
Sexe: M
Ville: LILLE
Taille: 177

id	id primaire	Poids	Notes
1	1	56	Je fais un essai
5	1	64	

Liste de tous les noms par ordre alphabétique

Liste de tous les noms par ordre inverse

Liste des noms commençant par: A

Liste des noms contenant: A

Liste des noms terminés par: A

Liste des noms résidant à: BREST

Liste des noms d'une taille = ou > à: 150

Liste des noms ayant un poids > ou = à: 50

Liste des noms de sexe: M

Quelques **commandes SQL** à utiliser sur base [SQLite3](#) à deux tables interconnectées.

Cet exemple utilise:
une table du nom de **PRIMAIRE** (sexe TEXT, id INTEGER PRIMARY KEY, nom TEXT, taille NUMERIC, ville TEXT)
une autre table **SECONDAIRE** (poids NUMERIC, idprimaire NUMERIC, idsecond INTEGER PRIMARY KEY, notes TEXT)

Exemple réalisé par taoteblues.

Quitter