

# [REBOLZONE]

(c) 2005 RebolZone - <http://rebolzone.free.fr>

## MySql

---

Très souvent les applications font appel à des bases de données pour stocker de l'information. Les versions gratuites de Rebol ne disposent pas d'un accès à ces bases de données. Il existe un protocole d'accès à **MySql** pour Rebol, développé en Rebol !, par DocKimbel.

On peut télécharger gratuitement <http://rebol.softinnov.org/mysql> (Merci !!!) ce protocole d'accès à MySql et l'utiliser pour développer tout type d'applications utilisant MySql. Son installation et son utilisation sont très simples. Bien sûr, on peut l'utiliser en environnement Windows, Unix, Linux ... Je l'ai utilisé avec MySql sous Windows et sous Unix.

Nous allons à partir de quelques exemples voir l'utilisation de l'accès à MySql. Pour plus d'informations, merci de vous reporter à la documentation officielle.

### Chargement du protocole MySql

Pour utiliser les fonctions d'accès à MySql, il faut en premier lieu, charger le protocole dans le script Rebol. Il faut faire référence au fichier `mysql-protocol.r` en précisant le chemin d'accès.

```
rebol[
Version: 1.1.2
auteur: "rebolzone"
date: 01-30-2003
Commentaires: "Utilisation de MySql"
]

; Chargement du protocole MySql
do %/C/Rebol/mysql-r099/mysql-protocol.r
```

### Ouverture d'une connexion MySql

On peut alors ouvrir un port sur le serveur MySql, pour une base de données :

```
db: open mysql://root@localhost/comdb
print type? db ; renvoi port
```

Dans cet exemple, un port est ouvert sur la base `comdb`, sur le serveur local, pour l'utilisateur `root`, sans mot de passe. On peut utiliser un utilisateur avec un mot de passe :

```
db: open mysql://team1:ADt22Xk@localhost/comdb ; ouverture avec le user team1 et son mot de pa
```

On peut également mentionner l'adresse IP du serveur, dans le cas d'un serveur distant. On peut aussi préciser le port d'écoute de MySql (serveur:NoPort), par défaut le port est 3306.

La variable `db` est le port MySql, elle sera utilisée au cours de nos exemples, mais rien ne nous empêche d'utiliser une variable nommée `database`.

### Exécution de requêtes SQL

L'exécution de requêtes SQL ( Select, Insert, Delete, Update, Create, Drop ...) se fait par l'instruction Rebol **insert** :

```
; Insertion dans la table client, d'un nouvel enregistrement
insert db ["insert into client (nom_cli) values ('Martin')"]
```

```
; Insertion avec une variable
client: "Dubois"
insert db ["insert into client (nom_cli) values (?)" client]

; Sélection de tout les clients
insert db ["select * from client"]
```

## Récupération des données d'un Select

Pour récupérer les données issues d'un sélect, il faut utiliser l'instruction **copy**. Le résultat du sélect est un tableau de blocks ou none si pas de résultat.

```
; Parcours du Recordset et affichage des enregistrements
foreach client copy db [probe client]
```

## Exécution de commandes d'administration

Le protocole permet d'envoyer au serveur MySql, certaines commandes d'administration telles que create-db, process-kill ...

```
; Récupération des statistiques du serveur MySql
print insert db [statistics]
```

## Fermeture de la connexion à MySql

La fermeture se fait par close :

```
close db
```

En conclusion, l'utilisation de ce protocole est très simple. Cela permet de faire de multiples applications utilisant MySql avec ou sans interface graphique.