

Gestion des Utilisateurs

Date : 01/12/2023 **XAMPP Version :** 8.1.10 **DOCKER:** 4.21.1 **Control Panel Version :** 3.3.0 **Type de serveur :** MariaDB **Version du serveur :**

Xampp: 10.4.25-MariaDB docker: 11.1.2-MariaDB

Se connecter à MySQL/MariaDB

Une fois mysql installé sur votre serveur (dans notre cas via Xampp ou Docker) on pourra se rendre dans une invite de commande (terminal) et taper :

Attention, si vous utiliser docker, il faudra d'abord se connecter au conteneur du serveur MySQL/MariaDB

```
mysql -V  
# ou  
mariadb -V
```

Si la commande est reconnue, cela affichera la version du gestionnaire de BDD. Sinon il faudra se rendre dans le dossier du gestionnaire pour lancer les commandes. (pour Xampp : `c:/xampp/mysql/bin`) ou alors ajouter ce même dossier à nos "**path**" dans les "**variables d'environnement**" de l'ordinateur (ou du conteneur).

Cela fait on pourra lancer la commande suivante :

```
mysql -u nomUtilisateur -p  
# ou  
mariadb -u nomUtilisateur -p
```

*par défaut le nom d'utilisateur est souvent **root**.*

Elle indique d'utiliser "**mysql**" puis de se connecter avec un utilisateur ("**-u**") nommé "**root**" et qu'il faudra entrer un mot de passe ("**-p**"). (Avec Xampp, le mot de passe par défaut de l'utilisateur root est "" rien du tout)

On verra notre **input** remplacé par :

```
MariaDB [(none)]>
```

GESTION DES UTILISATEURS

C'est donc ici que seront rentré nos premières commandes.

Créer un utilisateur

On va commencer par créer un nouvel utilisateur plus sécurisé :

```
CREATE USER "Nolwenn"@"localhost" IDENTIFIED BY "chaussette";
```

Convention : Les mots clef utilisé en sql sont écrit en majuscule. **Règle** : Chaque ligne de code est terminé par un ";".

Ici on trouve les mots-clefs :

- **"CREATE"** qui indique que l'on va créer quelque chose
- **"USER"** qui indique que l'on crée un utilisateur.
- Entre guillemet le nom de l'utilisateur, suivi de "@" puis de **"l'hôte"** de l'utilisateur. Nous travaillons en local, donc **"localhost"** sera suffisant. (dans un cas contraire cela pourrait être une adresse IP ou un URL).
- **"IDENTIFIED BY"** indique que l'on va donner un mot de passe puis entre guillemet le mot de passe.

Gestions des droits

Actuellement notre utilisateur n'a aucun droit, si il se connecte, il ne pourra rien faire. On va alors lui donner des droits :

```
GRANT ALL PRIVILEGES ON * . * TO "Nolwenn"@"localhost";
```

Les mots-clefs sont :

- **"GRANT"** indique que l'on va donner des droits.
- **"ALL PRIVILEGES"** On donne tous les droits.
- **"ON"** Précise sur quel élément il a ces droits.
- **"* . *** on indique les BDD, puis un "." suivi des table sur lesquels on a les droits. ("*" signifie **"tout"**)
- **"TO"** précède l'utilisateur auquel on donne les droits.

Ici on a donné *tous les droits*, sur *toute les BDD* et *toute les tables* à notre utilisateur.

Une fois tout les privileges paramétrés, il est bon d'envoyer le tout avec :

```
FLUSH PRIVILEGES;
```

(cela peut marcher sans, mais il vaut mieux faire les choses bien)

Ici on est dans un cas très précis où on veut pouvoir accéder à tout, mais dans le cas d'une entreprise on donnera sûrement accès à nos utilisateurs qu'à certaines BDD, certaines tables ou encore certains droits. Par exemple :

```
GRANT CREATE, DROP, DELETE ON truc . * TO "Maurice"@"localhost";
```

(On donne le droit de créer, de jeter ou de supprimer sur toute les tables de la bdd "truc")

Pareillement on peut retirer des droits avec :

```
REVOKE DROP, DELETE ON truc . * FROM "Maurice"@"localhost";
```

(On retire les droits de jeter et supprimer sur toute les tables de la bdd "truc")

Selon la version de mysql, cela peut entrainer des erreurs de droit d'accès si ce n'est pas root qui lance cela.

On peut vérifier les droits d'un utilisateur avec :

```
SHOW GRANTS FOR "Maurice"@"localhost";
```

- Le mot clef "**SHOW**" indique que l'on veut voir quelque chose.
- On indique ensuite ce qu'on veut voir, ici "**GRANTS**".
- Puis pour qui avec "**FOR**"

Supprimer et Modifier un Utilisateur

Supprimer un utilisateur est possible avec :

```
DROP USER "Maurice"@"localhost";
```

On souhaite jeter "**DROP**" un utilisateur.

Pour modifier un mot de passe existant :

```
ALTER USER "Maurice"@"localhost" IDENTIFIED BY "machin";
```

- le mot clef "**ALTER**" signifie que l'on va modifier quelque chose.

Enfin, si on souhaite changer d'utilisateur on va devoir quitter MySQL et se reconnecter, pour quitter on utilisera :

```
quit;
```