

## Fonctionnement api

### 1. Installation mariadb:

```
sudo apt-get install libmariadbclient-dev
```

Pour l'api on utilisera une connexion à la base de données de mariadb  
faire attention à ne pas installer la bibliothèque de MySQL la connexion  
avec la bdd ne fonctionnera pas.

1.1 Vérifier la bonne installation de la bibliothèque : effectuant la  
commande suivante : `dpkg -l | grep mariadb`

1.2 Mettre à jour son système en utilisant la commande :  
`sudo apt update -y && sudo apt full-upgrade -y`

1.3 installer Apache 2 :  
`sudo apt-get install apache2`

1.4 vérifier que apache2 marche  
`systemctl status apache2`  
vous pouvez vérifier en mettant votre adresse ip dans votre navigateur

### 1.5 installation du serveur mariadb

Entre dans

```
Cd /var/www/html
```

Puis

```
Cd /var/www/html# apt-get install mariadb-server
```

en cas première installation effectuer commande suivante pour être sûr du  
bon fonctionnement :

```
/var/www/html# mysql_secure_installation
```

```
set root password ? [Y/n] : Y (c'est mieux d'en configurer un)
```

```
new password : ...
```

```
re-enter new password :
```

```
Remove anonymous users ? [Y/n] : Y
```

Disallow root login remotely ? [Y/n] : Y

remove test database and access to it ? [Y/n] : Y

reload privilege table now [Y/n]: Y

Une fois terminer vous revenez ici  
/var/www/html#

1.6 entre dans mariadb

/var/www/html# mysql -u root -p

Enter password : (entrer le mdp que vous avez configurer plus tôt)

MariaDB [(none)]>

2. Configuration des fichiers de l'api

2.1 installer un IDE de préférence Clion ou qt créator

2.2 créé un nouveau projet en c++ 17

2.3 accéder au fichier du projet

2.4 importer les fichiers qui se trouve dans ressources projet

[mariadbpp-master, main.cpp, CMakeLists.txt] les fichiers main et cmake sont des fichiers de base lors de la création d'un projet il faudra donc appuyer sur remplacer lorsque vous les importez

2.5 modifier les fichier Cmake, revenez sur l'IDE aller dans votre fichiers cmake ensuite il faudra modifier la ligne 5 vous devrez mettre le chemin vers le fichier include de mariadbpp-master

2.6 modifier le fichier main.cpp à la ligne 14 si besoin (par défaut il n'y a pas de mot de passe) et n'oublier pas de mettre le nom de la base de donner que vous voulez utiliser

2.7 compiler le main.cpp, ouvrir un terminal, accéder au dossier ou son stocké vos projets

```

(navexd@Navepc)-[~] save main main.cpp CMakeLists.txt
$ ls
CLionProjects Desktop Documents Downloads Music Pictures Public Templates Videos soft

(navexd@Navepc)-[~]
$ cd CLionProjects

(navexd@Navepc)-[~/CLionProjects]
$ ls
test untitled untitled1 untitled2

(navexd@Navepc)-[~/CLionProjects]
$ cd untitled1

(navexd@Navepc)-[~/CLionProjects/untitled1]
$ ls
CMakeLists.txt build cmake-build-debug main main.cpp main.cpp.save mariadbpp-master

(navexd@Navepc)-[~/CLionProjects/untitled1]
$

```

Entrez cette commande dans votre terminal `g++ -o test main.cpp -lmariadb`

Fait un `ls` pour vérifier que le fichier est bien créé

2.8 vous pourrez ensuite faire la commande `./main`

```

(navexd@Navepc)-[~/CLionProjects/untitled1]
$ ./main
Connection object OK, let's connect.
Connection success

(navexd@Navepc)-[~/CLionProjects/untitled1]
$

```