

Petit Guide PSQL

Ce guide est minimaliste. Il est fait pour le CREMI avec les machines Linux.

Initialisation de postgresql

Il faut définir deux variables d'environnement pour préciser le **host** et le **port** de notre serveur de base de données.

```
# Définir le host
export PGHOST=/tmp/$LOGNAME
# Définir le port
export PGPORT=$UID
```

L'initialisation du serveur nécessite les deux commandes suivantes (à ne faire qu'une fois sur la machine) :

```
# Initialiser les fichiers nécessaires à PSQL
/usr/lib/postgresql/11/bin/pg_ctl -D /tmp/$LOGNAME/ -l /tmp/$LOGNAME/startup.log
initdb

# Changer la configuration de base pour que cela marche au CREMI
pg_conftool -v /tmp/$LOGNAME/postgresql.conf set unix_socket_directories
/tmp/$LOGNAME/
```

Démarrage et arrêt du serveur

Dès que la base est initialisée, vous pouvez démarrer le serveur avec la commande suivante :

```
# Start database
/usr/lib/postgresql/11/bin/pg_ctl -D /tmp/$LOGNAME/ -l /tmp/$LOGNAME/startup.log
start
```

Vous pouvez alors vérifier que votre base est bien lancée avec la commande suivante :

```
#list all databases
psql -l
```

Quand vous avez terminé, n'oubliez pas d'éteindre votre base avec la commande suivante :

```
# Stop database
/usr/lib/postgresql/11/bin/pg_ctl -D /tmp/$LOGNAME/ -l /tmp/$LOGNAME/startup.log
```

```
stop
```

Exécution des commandes

La manipulation de la base de données nécessite l'utilisation de **psql**. C'est un simple client (en ligne de commande) de la base de données.

Pour démarrer le client **psql**, il faut cibler une base de données existante. Postgres démarre avec la base nommée **postgres**. On va donc commencer avec cette base.

La commande suivante permet de démarrer le client **psql** sur la base **postgres** :

```
psql -d postgres
```

Le client **psql** permet de lancer des commandes suivantes:

- \l : liste les bases de données
- \c <data_base_name> : connecte à une base de données
- \d : liste les tables de la base de données sur laquelle on est connecté
- \d <table_name> : affiche le schéma de la table tition SQL
- SQL : exécute une instruction SQL
- ? : liste toutes les commandes de **psql**
- \h <sql_name>: affiche l'aide d'une instruc
- \q : quitte le client

Quelques instructions SQL:

- Création d'un base : CREATE DATABASE <nom_de_la_base>;
- Création d'une table : CREATE TABLE <nom_de_la_table> (<nom_de_la_colonne> <type_de_la_colonne>, ...);
- Ajouter une ligne : INSERT INTO <nom_de_la_table> (<nom_de_la_colonne>, ...) VALUES (, ...);
- Voir toutes les lignes : SELECT * FROM <nom_de_la_table>;

Ainsi les commandes suivantes exécutées par **psql** permettent :

1. de créer la base **test**
2. de s'y connecter
3. de créer une table **ma_table** qui a deux colonnes (id et nom) de type **int** et **varchar**
4. d'ajouter une ligne dans la table **ma_table**
5. de voir toutes les lignes de la table **ma_table**

```
CREATE DATABASE test;
\c test
CREATE TABLE ma_table ( id INT PRIMARY KEY NOT NULL, nom VARCHAR(30) );
\d ma_table
INSERT INTO images (id, nom_fichier, hauteur, largeur) VALUES (1, 'image1.jpg',
```

```
100, 200);  
SELECT * FROM ma_table;
```