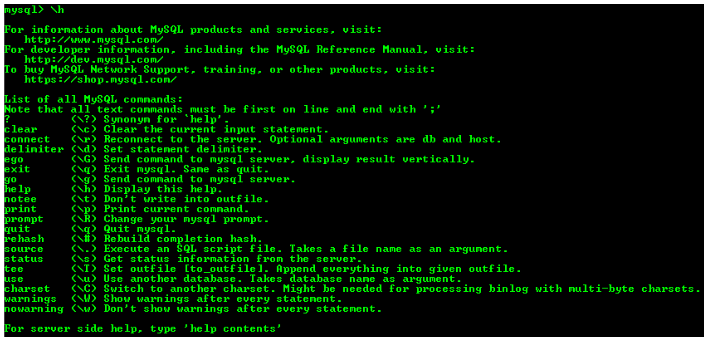
Quelques commandes utiles pour MySQL et SQLite en shell



```
mysql> \h

For information about MySQL products and services, visit:
   http://www.mysql.com/
For developer information, including the MySQL Reference Manual, visit:
   http://dev.mysql.com/
To buy MySQL Network Support, training, or other products, visit:
   https://shop.mysql.com/

List of all MySQL commands:
Note that all text commands must be first on line and end with ';'
?         (\?) Synonym for `help'.
clear     (\c) Clear the current input statement.
connect   (\r) Reconnect to the server. Optional arguments are db and host.
delimiter (\d) Set statement delimiter.
ego       (\G) Send command to mysql server, display result vertically.
exit      (\q) Exit mysql. Same as quit.
go        (\g) Send command to mysql server.
help      (\h) Display this help.
notee     (\t) Don't write into outfile.
print     (\p) Print current command.
prompt    (\R) Change your mysql prompt.
quit      (\q) Quit mysql.
rehash    (\#) Rebuild completion hash.
source    (\.) Execute an SQL script file. Takes a file name as an argument.
status    (\s) Get status information from the server.
tee       (\T) Set outfile [to_outfile]. Append everything into given outfile.
use       (\u) Use another database. Takes database name as argument.
charset   (\C) Switch to another charset. Might be needed for processing binlog with multi-byte charsets.
warnings  (\W) Show warnings after every statement.
nowarning (\w) Don't show warnings after every statement.

For server side help, type 'help contents'
```

MySQL (MariaDB)

Pour installer le serveur de BDD MySQL :

```
sudo apt update
sudo apt install mysql-server
```

Il y a de nombreuses solutions de connexion à l'environnement MySQL/MariaDB par le biais d'applications (GUI) ou en ligne de commande.

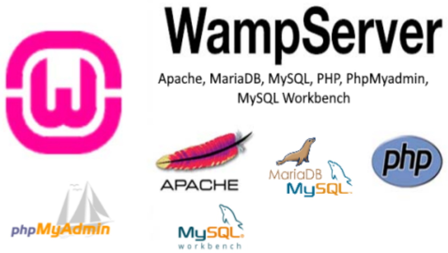


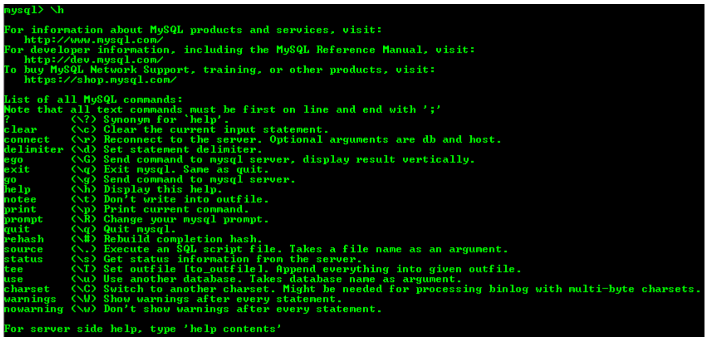
Figure 1: "Interfaces Utilisateurs Graphiques"

Explorons une connexion dans **bash**.

Avant d'exécuter des commandes, connectez-vous à MySQL :

```
mysql -u root -p
```

(si vous utilisez un autre utilisateur, remplacez `root` par votre nom d'utilisateur)



```
mysql> \h

For information about MySQL products and services, visit:
   http://www.mysql.com/
For developer information, including the MySQL Reference Manual, visit:
   http://dev.mysql.com/
To buy MySQL Network Support, training, or other products, visit:
   https://shop.mysql.com/

List of all MySQL commands:
Note that all text commands must be first on line and end with ';'
?         (\?) Synonym for `help'.
clear     (\c) Clear the current input statement.
connect   (\r) Reconnect to the server. Optional arguments are db and host.
delimiter (\d) Set statement delimiter.
ego       (\G) Send command to mysql server, display result vertically.
exit      (\q) Exit mysql. Same as quit.
go        (\g) Send command to mysql server.
help      (\h) Display this help.
notee     (\t) Don't write into outfile.
print     (\p) Print current command.
prompt    (\R) Change your mysql prompt.
quit      (\q) Quit mysql.
rehash    (\#) Rebuild completion hash.
source    (\.) Execute an SQL script file. Takes a file name as an argument.
status    (\s) Get status information from the server.
tee       (\T) Set outfile [to_outfile]. Append everything into given outfile.
use       (\u) Use another database. Takes database name as argument.
charset   (\C) Switch to another charset. Might be needed for processing binlog with multi-byte charsets.
warnings  (\W) Show warnings after every statement.
nowarning (\w) Don't show warnings after every statement.

For server side help, type 'help contents'
```

Figure 2: "Pour une connexion avec bash"

Commandes de base

```sql
SHOW DATABASES;            -- Affiche la liste des bases de données
USE ma_base;               -- Sélectionne la base de données "ma_base"
SHOW TABLES;               -- Liste les tables de la base sélectionnée
DESCRIBE ma_table;         -- Affiche la structure de la table "ma_table"
```

Gestion des bases de données

```sql
CREATE DATABASE ma_base; -- Crée une base de données "ma_base"
DROP DATABASE ma_base;   -- Supprime la base de données "ma_base"
```

Gestion des tables

```sql
CREATE TABLE utilisateurs (
    id INT PRIMARY KEY AUTO_INCREMENT,
    nom VARCHAR(100),
    email VARCHAR(100) UNIQUE
); -- Crée une table "utilisateurs"

DROP TABLE utilisateurs; -- Supprime la table "utilisateurs"
```

Manipulation des données

```sql
INSERT INTO utilisateurs (nom, email) VALUES ('Alice', 'alice@example.com');
SELECT * FROM utilisateurs;
```

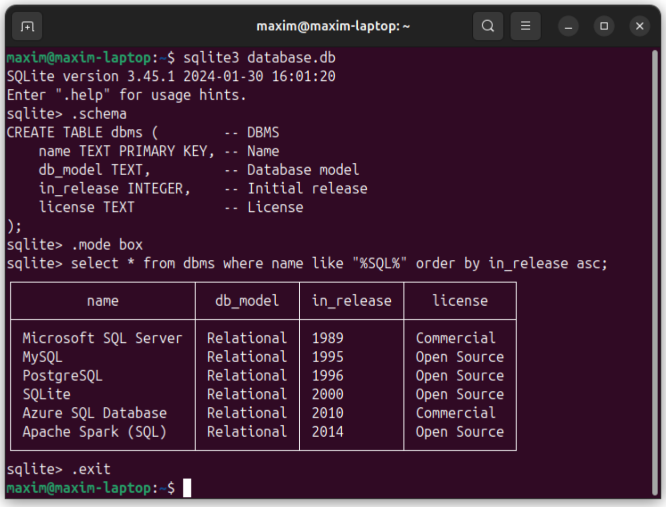
```sql
UPDATE utilisateurs SET nom = 'Bob' WHERE id = 1;
DELETE FROM utilisateurs WHERE id = 1;
```

Export / Import

```
mysqldump -u root -p ma_base > backup.sql  # Exporter la base de données
mysql -u root -p ma_base < backup.sql      # Importer une base de données
```

SQLite

SQLite est un moteur de base de données léger qui utilise un fichier `.sqlite` ou `.db`.



```
maxim@maxim-laptop:~$ sqlite3 database.db
SQLite version 3.45.1 2024-01-30 16:01:20
Enter ".help" for usage hints.
sqlite> .schema
CREATE TABLE dbms (          -- DBMS
    name TEXT PRIMARY KEY, -- Name
    db_model TEXT,         -- Database model
    in_release INTEGER,    -- Initial release
    license TEXT           -- License
);
sqlite> .mode box
sqlite> select * from dbms where name like "%SQL%" order by in_release asc;
┌───────────────────────┬────────────┬────────────┬─────────────┐
│         name          │  db_model  │ in_release │   license   │
├───────────────────────┼────────────┼────────────┼─────────────┤
│ Microsoft SQL Server  │ Relational │ 1989       │ Commercial  │
│ MySQL                 │ Relational │ 1995       │ Open Source │
│ PostgreSQL            │ Relational │ 1996       │ Open Source │
│ SQLite                │ Relational │ 2000       │ Open Source │
│ Azure SQL Database    │ Relational │ 2010       │ Commercial  │
│ Apache Spark (SQL)    │ Relational │ 2014       │ Open Source │
└───────────────────────┴────────────┴────────────┴─────────────┘
sqlite> .exit
maxim@maxim-laptop:~$
```

Connexion à SQLite

```
sqlite3 ma_base.sqlite
```

Ou si vous avez un fichier existant :

```
sqlite3 ma_base.db
```

Commandes de base

```
.databases          -- Affiche les bases disponibles
.tables             -- Liste les tables dans la base sélectionnée
```

```
.schema ma_table   -- Affiche la structure de la table "ma_table"
.quit              -- Quitter SQLite
```

Gestion des tables

```
CREATE TABLE utilisateurs (
    id INTEGER PRIMARY KEY AUTOINCREMENT,
    nom TEXT,
    email TEXT UNIQUE
);

DROP TABLE utilisateurs;
```

Manipulation des données

```
INSERT INTO utilisateurs (nom, email) VALUES ('Alice', 'alice@example.com');
SELECT * FROM utilisateurs;
UPDATE utilisateurs SET nom = 'Bob' WHERE id = 1;
DELETE FROM utilisateurs WHERE id = 1;
```

Export / Import

```
sqlite3 ma_base.sqlite .dump > backup.sql  # Exporter la base de données
sqlite3 ma_base.sqlite < backup.sql        # Importer la base de données
```