

### Bases

#### Afficher la version du serveur PostgreSQL

```
postgres=# SELECT version();
```

#### Afficher la version du client PostgreSQL

```
psql -V
```

#### Vérifier si Postgres tourne

```
/etc/init.d/postgresql status
```

#### Démarrer / arrêter / redémarrer Postgres (SysVinit)

```
/etc/init.d/postgresql start | stop | restart
```

#### Démarrer / arrêter / redémarrer Postgres (Systemd)

```
systemctl start | stop | restart postgresql.service
```

#### Saisir nos requêtes dans notre éditeur de prédilection

```
postgres=# \e
```

#### Changer le mot de passe root de PostgreSQL

```
psql postgres postgres
```

```
postgres=# ALTER ROLE postgres WITH PASSWORD 'toto';
```

```
postgres=# \q
```

#### Listez les fonctions disponibles

```
postgres=# \df
```

#### Afficher l'aide

```
postgres=# \?
```

```
postgres=# \h CREATE
```

```
postgres=# \h CREATE INDEX
```

#### Listez les DB

```
postgres=# \l
```

#### Créer une DB depuis Postgres

```
postgres=# CREATE DATABASE maDB WITH OWNER akiko;
```

#### Créer une DB depuis le Shell

```
postgres@debianVM:~$ /usr/bin/createdb maDB -O midori
```

### Bases (cont)

#### Supprimer une base depuis Postgres

```
postgres=# DROP DATABASE maDB;
```

#### Supprimer une base depuis le Shell

```
postgres@debianVM:~$ dropdb maDB
```

#### Listez les tables d'une DB

```
postgres=# \c postgres
```

```
postgres=# \d
```

#### Obtenir une liste des types de données disponibles dans PostgreSQL

```
postgres=# SELECT typename,typlen from pg_type where typtype='b';
```

#### Rediriger les résultats d'une requête vers un fichier

```
postgres=# \o nom_fichier (active la redirection)
```

```
postgres=# \o (annule la redirection)
```

### Sauvegardes

#### Sauvegarder une seule base

```
pg_dump -U utilisateur nom_DB -f madb.sql
```

#### Sauvegarder toutes les bases

```
pg_dumpall > tout.sql
```

#### Sauvegarder les objets globaux (users + tablespaces)

```
pg_dumpall -g > tout.sql
```

#### Sauvegarder une table

```
pg_dump --table articles -U midori nom_DB -f unetable.sql
```

#### Restaurer une base

```
psql -U midori -d nom_DB -f madb.sql
```

#### Restaurer toutes les bases

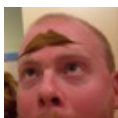
```
psql -f tout.sql
```

#### Restaurer une table

```
psql -f unetable.sql nom_DB
```

#### Sauvegarder une base locale et la restaurer sur un serveur distant en une ligne

```
pg_dump nom_DB_source | psql -h serveur nom_DB_cible
```



### Index

#### Voir les index d'une table

```
postgres=# \d nom_table
```

#### Création d'un index sur une table

```
CREATE INDEX nom ON table USING type_index (colonne);
```

Postgres supporte 5 types d'index : Balanced-Tree (btree ; utilisé par défaut), hash (transactions non-journalisées, déconseillé), Generalized Search Tree (gist), Generalized Inverted Indexes (gin) et Space-Partitioned GIST (spgist).

### Investiguer, analyser

#### Consulter le fichier d'historique

```
cat ~/.psql_history
```

#### Activer / désactiver le timing

```
postgres=# \timing
```

#### Obtenir les détails d'exécution d'une requête sans la faire tourner

```
postgres=# EXPLAIN SELECT typename,typlen from pg_type
where typtype='b';
```

#### Obtenir les statistiques d'une requête qui vient de tourner

```
postgres=# EXPLAIN ANALYSE SELECT typename,typlen from
pg_type where typtype='b';
```

Un peu à la manière de Linux avec son *bash\_history*, PostgreSQL conserve une liste des commandes exécutées dans le fichier *\$HOME/.psql\_history*.

Le timing chronomètre le temps qu'une requête prend pour s'exécuter.

### Utilisateurs et rôles

#### Créer un utilisateur depuis le Shell

```
/usr/bin/createuser midori
```

#### Créer un utilisateur depuis Postgres

```
postgres=# CREATE USER akiko WITH password 'yamete';
```

#### Accorder des privilèges à un utilisateur

```
postgres=# GRANT ALL PRIVILEGES ON DATABASE
CityHunter TO akiko;
```

#### Supprimer un utilisateur depuis le Shell

```
/usr/bin/dropuser midori
```

#### Supprimer un utilisateur depuis Postgres

```
postgres=# DROP USER akiko;
```

#### Lister les rôles

```
postgres=# \du
```

#### Créer un rôle

```
postgres=# CREATE ROLE formateur WITH LOGIN
ENCRYPTED PASSWORD 'apprendre' CREATEDB;
```

#### Créer un rôle avec plusieurs privilèges

```
postgres=# CREATE ROLE demidieu WITH LOGIN
ENCRYPTED PASSWORD 'toto' CREATEDB CREATEROLE
REPLICATION SUPERUSER;
```

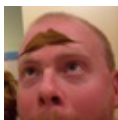
#### Modifier un rôle

```
postgres=# ALTER ROLE demidieu CREATEROLE CREATEDB
REPLICATION SUPERUSER;
```

#### Supprimer un rôle

```
postgres=# DROP ROLE formateur;
```

Ces deux variantes nécessitent d'être en utilisateur *postgres* (su - postgres).



By **TME520** (TME520)  
[cheatography.com/tme520/](https://cheatography.com/tme520/)  
[tme520.com](https://tme520.com)

Published 14th April, 2015.  
 Last updated 12th May, 2016.  
 Page 2 of 3.

Sponsored by **Readable.com**  
 Measure your website readability!  
<https://readable.com>

### Espace disque

#### Calculer l'espace disque occupé par une base

```
postgres=# SELECT pg_database_size('foodb');
```

```
postgres=# SELECT pg_size_pretty(pg_database_size('foodb'));
```

#### Calculer l'espace disque occupé par une table (index inclus)

```
postgres=# SELECT pg_size_pretty(pg_total_relation_size('grosse_table'));
```

#### Calculer l'espace disque occupé par une table (sans l'index)

```
postgres=# SELECT pg_size_pretty(pg_relation_size('grosse_table'));
```

#### Trouver la table la plus volumineuse (variante 1)

```
postgres=# SELECT relname, relpages FROM pg_class ORDER BY relpages DESC;
```

#### Trouver la table la plus volumineuse (variante 2)

```
postgres=# SELECT nspname || '.' || relname AS tablename,
pg_size_pretty(pg_table_size((nspname || '.' || relname)::regclass))
AS size FROM pg_class c JOIN pg_namespace n ON (c.relnamespace = n.oid) WHERE relkind = 'r' ORDER BY pg_table_size((-nspname || '.' || relname)::regclass) DESC LIMIT 10;
```

### Transactions

#### Démarrer une transaction

```
postgres=# BEGIN
```

#### Faire un rollback

```
postgres=# ROLLBACK
```

#### Faire un commit (fin de transaction)

```
postgres=# COMMIT
```

### SQL

#### Compter les lignes d'une table

```
postgres=# select count(*) from table;
```

#### Générer une série de nombres et l'insérer dans une table

```
postgres=# INSERT INTO nombres (num) VALUES ( generate_series(1,100));
```

#### Récupérer la seconde valeur la plus petite d'une colonne

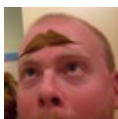
```
postgres=# SELECT MIN(num) from number_table WHERE num
> ( SELECT MIN(num) FROM number_table );
```

#### Récupérer la seconde valeur la plus grande d'une colonne

```
postgres=# SELECT MAX(num) FROM number_table WHERE
num < ( SELECT MAX(num) FROM number_table );
```

#### Crypter, puis enregistrer un mot de passe

```
postgres=# SELECT crypt ( 'midori', gen_salt('md5') );
```



By **TME520** (TME520)  
[cheatography.com/tme520/](https://cheatography.com/tme520/)  
[tme520.com](https://tme520.com)

Published 14th April, 2015.  
 Last updated 12th May, 2016.  
 Page 3 of 3.

Sponsored by **Readable.com**  
 Measure your website readability!  
<https://readable.com>