

Lexique simplifié récapitulant les principales commandes SQL (version 0.8)

(lexique établi d'après la documentation de PostgreSQL 8) - A. FRANÇOIS

Gestion des bases de données

CREATE DATABASE nom

DROP DATABASE nom

Gestion des utilisateurs

CREATE USER nom [[WITH] option [...]]

où option peut être :

```
SUPERUSER | NOSUPERUSER
| CREATEDB | NOCREATEDB
| CREATEUSER | NOCREATEUSER
| LOGIN | NOLOGIN
| CONNECTION LIMIT limite_connexion
| [ ENCRYPTED | UNENCRYPTED ] PASSWORD 'motdepasse'
| VALID UNTIL 'timestamp'
```

DROP USER [IF EXISTS] nom [, ...]

Gestion des privilèges

GRANT { { SELECT | INSERT | UPDATE | DELETE | TRIGGER } [, ...] | ALL [PRIVILEGES] }
ON [TABLE] tablename [, ...]
TO { username | GROUP groupname | PUBLIC } [, ...] [WITH GRANT OPTION]

GRANT { { CREATE | TEMPORARY | TEMP } [, ...] | ALL [PRIVILEGES] }
ON DATABASE dbname [, ...]
TO { username | GROUP groupname | PUBLIC } [, ...] [WITH GRANT OPTION]

REVOKE [GRANT OPTION FOR]
{ { SELECT | INSERT | UPDATE | DELETE } [, ...] | ALL [PRIVILEGES] }
ON [TABLE] nom_table [, ...]
FROM { nom_utilisateur } [, ...]
[CASCADE | RESTRICT]

REVOKE [GRANT OPTION FOR]
{ { CREATE | TEMPORARY | TEMP } [, ...] | ALL [PRIVILEGES] }
ON DATABASE nom_base [, ...]
FROM { nom_utilisateur | GROUP nom_groupe | PUBLIC } [, ...]
[CASCADE | RESTRICT]

Gestion des tables

CREATE TABLE nom_table (
 nom_colonne type_données [contrainte_colonne [...]]
 | contrainte_table
 | [, ...]
)

où *contrainte_colonne* est :

```
{ NOT NULL |
  NULL |
  UNIQUE |
```

```
PRIMARY KEY |
CHECK (expression) |
REFERENCES table_reference [ ( colonne_reference ) ]
[ ON DELETE CASCADE ] [ ON UPDATE CASCADE ] }
```

et *contrainte_table* est :

```
{ UNIQUE ( nom_colonne [, ... ] ) |
  PRIMARY KEY ( nom_colonne [, ... ] ) |
  CHECK ( expression ) |
  FOREIGN KEY ( nom_colonne [, ... ] ) REFERENCES
table_reference [ ( colonne_reference [, ... ] ) ]
[ ON DELETE CASCADE ] [ ON UPDATE CASCADE ] }
```

DROP TABLE nom [, ...] [CASCADE]

ALTER TABLE nom
action [, ...]

ALTER TABLE nom
RENAME [COLUMN] colonne TO nouvelle_colonne

ALTER TABLE nom
RENAME TO nouveau_nom

où *action* est parmi :

```
ADD [ COLUMN ] colonne type [ column_constraint [ ... ] ]
DROP [ COLUMN ] colonne RESTRICT | CASCADE ]
ALTER [ COLUMN ] colonne TYPE type [ USING expression ]
ALTER [ COLUMN ] colonne SET DEFAULT expression
ALTER [ COLUMN ] colonne DROP DEFAULT
ALTER [ COLUMN ] colonne { SET | DROP } NOT NULL
ALTER [ COLUMN ] colonne SET STATISTICS integer
ALTER [ COLUMN ] colonne SET STORAGE { PLAIN | EXTERNAL |
EXTENDED | MAIN }
ADD contrainte_table
DROP CONSTRAINT nom_contrainte [ RESTRICT | CASCADE ]
OWNER TO nouveau_propriétaire
```

DROP TABLE nom [, ...] [CASCADE | RESTRICT]

Manipulation des données

INSERT INTO table [(colonne [, ...])]
VALUES (expression [, ...])

UPDATE table SET colonne = expression [, ...]
[FROM depuis_liste]
[WHERE condition]

DELETE FROM [ONLY] table [WHERE condition]

SELECT [DISTINCT]
* | expression [AS nom_sortie] [, ...]
[FROM élément_from [, ...]]
[WHERE condition]
[GROUP BY expression [, ...]]
[HAVING condition [, ...]]
[{ UNION | INTERSECT | EXCEPT } select]
[ORDER BY expression [ASC | DESC] [, ...]]
[LIMIT nombre]
[OFFSET début]

où *élément_from* peut être :

```
[ ONLY ] nom_table [ * ] [ [ AS ] alias [ ( alias_colonne
[, ...] ) ] ]
( select ) [ AS ] alias [ ( alias_colonne [, ...] ) ]
nom_fonction ( [ argument [, ...] ] ) [ AS ] alias
[ ( alias_colonne [, ...] | définition_colonne [, ...] ) ]
nom_fonction ( [ argument [, ...] ] ) AS
( définition_colonne [, ...] )
élément_from [ NATURAL ] type_jointure élément_from [ ON
condition_jointure | USING ( colonne_jointure [, ...] ) ]
```

Gestion des vues

CREATE VIEW nom [(nom_colonne [, ...])] AS requête

DROP VIEW nom [, ...] [CASCADE | RESTRICT]

Gestion des index

CREATE [UNIQUE] INDEX nom ON table [USING méthode]
({ colonne | (expression) } [classe_opérateur]
[, ...])
[TABLESPACE espace_logique]
[WHERE prédicat]

DROP INDEX nom [, ...] [CASCADE | RESTRICT]

Gestion des transactions

BEGIN [WORK | TRANSACTION]

END [WORK | TRANSACTION]

COMMIT [WORK | TRANSACTION]

ROLLBACK [WORK | TRANSACTION]

Fonctions agrégats

sum(), count(), min(), max(), avg()

Principaux types

boolean : booléen

varchar(n) : suite de caractères de longueur variable mais inférieure ou égale à n

char(n) : suite de caractères de longueur fixe de taille n

text : chaîne de caractères de longueur variable

date : date du calendrier au format ISO ('aaaa-mm-jj')

int, int4 : entier

float4 : nombre à virgule flottante simple précision

float8 : nombre à virgule flottante double précision

serial : entier de 4 octets à incrémentation automatique (engendre la création d'une séquence)

timestamp : date et heure

Méta-caractères servant à décrire la syntaxe de SQL

| : ou
[] : partie facultative
... : répétition
{ } : partie obligatoire

Langage de définition

```
CREATE TABLE nom_table (  
    nom_col1    type        PRIMARY KEY  
    nom_col2    type        REFERENCES  nom_table (nom_col)  
    nom_col3    type        CHECK (expression),  
    nom_col4    type        UNIQUE,  
    nom_col5    type        NOT NULL,  
  
    PRIMARY KEY (nom_col1, nom_col2, ...),  
    FOREIGN KEY (nom_col1, nom_col2, ...)  
        REFERENCES nom_table(nom_col1, nom_col2, ...)  
)
```

```
CREATE TABLE nom_table (nom_col1, ...)  
AS SELECT ...
```

```
ALTER TABLE nom_table  
    ADD(nom_col    type        NOT NULL, ...)  
ou DROP(nom_col, ...)  
ou MODIFY(nom_col    type        NOT NULL, ...)
```

```
ALTER TABLE nom_table  
    ADD CONSTRAINT FOREIGN KEY (nom_col,...)  
    REFERENCES nom_table (nom_col,...)  
    ON UPDATE CASCADE ou SET NULL  
    ou NO ACTION ou SET DEFAULT  
    ON DELETE CASCADE ....;
```

```
DROP TABLE nom_table  
    CASCADE ;
```

```
CREATE VIEW nom_vue  
AS SELECT ...
```

```
CREATE UNIQUE INDEX nom_index  
    ON nom_table (nom_col ASC ou DESC, ...);
```

```
DROP INDEX nom_index ON nom_table  
    ou TABLE nom_table;
```

Langage de manipulation

```
INSERT INTO nom_table (nom_col1, nom_col2, ...)  
VALUES(valeur1, valeur2, ...);
```

```
UPDATE nom_table  
    SET nom_col = expression  
    WHERE condition;
```

```
DELETE FROM nom_table  
    WHERE condition ;
```

Langage de transaction

```
START TRANSACTION ou BEGIN WORK  
COMMIT  
ROLLBACK
```

Langage de contrôle

```
GRANT SELECT ou INSERT ou UPDATE ou DELETE ou ALL  
    ON nom_objet  
    TO users,... ou PUBLIC  
    WITH GRANT OPTION
```

```
REVOKE SELECT ou ...  
    ON nom_objet  
    FROM users,...  
    RESTRICT ou CASCADE
```

Langage d'interrogation

```
SELECT ALL ou DISTINCT nom_col1, nom_col2...  
FROM nom_table1  
    INNER JOIN nom_table2 ON (condition)  
WHERE condition  
ORDER BY ...;
```

```
SELECT ALL ou DISTINCT nom_table1.nom_col1, ...  
FROM nom_table1  
    LEFT ou RIGHT OUTER JOIN nom_table2
```

```
SELECT nom_col1, nom_col2, fonction_regroupement(...), ...  
FROM ...  
WHERE condition  
GROUP BY ...  
HAVING condition
```

```
SELECT ...  
    UNION ou EXCEPT ou INTERSECT ...  
SELECT ....
```

Opérateurs de comparaison

```
= != <> > >=  
IN NOT IN ANY ALL  
NOT BETWEEN x AND y NOT EXISTS
```

Caractères génériques

```
% _
```

Lexique simplifié récapitulant les principales commandes SQL

(lexique établi d'après la documentation de PostgreSQL 8)