

NAME

DROP OWNED – remove database objects owned by a database role

SYNOPSIS

DROP OWNED BY { *name* | CURRENT_ROLE | CURRENT_USER | SESSION_USER } [, ...] [CASCADE | RESTRICT]

DESCRIPTION

DROP OWNED drops all the objects within the current database that are owned by one of the specified roles. Any privileges granted to the given roles on objects in the current database or on shared objects (databases, tablespaces, configuration parameters) will also be revoked.

PARAMETERS

name

The name of a role whose objects will be dropped, and whose privileges will be revoked.

CASCADE

Automatically drop objects that depend on the affected objects, and in turn all objects that depend on those objects (see Section 5.15).

RESTRICT

Refuse to drop the objects owned by a role if any other database objects depend on one of the affected objects. This is the default.

NOTES

DROP OWNED is often used to prepare for the removal of one or more roles. Because **DROP OWNED** only affects the objects in the current database, it is usually necessary to execute this command in each database that contains objects owned by a role that is to be removed.

Using the CASCADE option might make the command recurse to objects owned by other users.

The **REASSIGN OWNED** command is an alternative that reassigns the ownership of all the database objects owned by one or more roles. However, **REASSIGN OWNED** does not deal with privileges for other objects.

Databases and tablespaces owned by the role(s) will not be removed.

See Section 21.4 for more discussion.

COMPATIBILITY

The **DROP OWNED** command is a PostgreSQL extension.

SEE ALSO

REASSIGN OWNED (**REASSIGN OWNED**(7)), DROP ROLE (**DROP ROLE**(7))