NAME

createdb - create a new PostgreSQL database

SYNOPSIS

createdb [*connection-option*...] [*option*...] [*dbname* [*description*]]

DESCRIPTION

createdb creates a new PostgreSQL database.

Normally, the database user who executes this command becomes the owner of the new database. However, a different owner can be specified via the **-O** option, if the executing user has appropriate privileges.

createdb is a wrapper around the SQL command **CREATE DATABASE**. There is no effective difference between creating databases via this utility and via other methods for accessing the server.

OPTIONS

createdb accepts the following command-line arguments:

dbname

Specifies the name of the database to be created. The name must be unique among all PostgreSQL databases in this cluster. The default is to create a database with the same name as the current system user.

description

Specifies a comment to be associated with the newly created database.

-D *tablespace*
--tablespace=*tablespace*

Specifies the default tablespace for the database. (This name is processed as a double-quoted identifier.)

-e
--echo

Echo the commands that createdb generates and sends to the server.

-E *encoding*
--encoding=*encoding*

Specifies the character encoding scheme to be used in this database. The character sets supported by the PostgreSQL server are described in Section 24.3.1.

-l *locale*
--locale=*locale*

Specifies the locale to be used in this database. This is equivalent to specifying both **--lc-collate** and **--lc-ctype**.

--lc-collate=*locale*

Specifies the LC_COLLATE setting to be used in this database.

--lc-ctype=*locale*

Specifies the LC_CTYPE setting to be used in this database.

--icu-locale=*locale*

Specifies the ICU locale ID to be used in this database, if the ICU locale provider is selected.

--locale-provider={libc|icu}

Specifies the locale provider for the database's default collation.

-O *owner*
--owner=*owner*

Specifies the database user who will own the new database. (This name is processed as a double-quoted identifier.)

-S *template*
--strategy=*strategy*

Specifies the database creation strategy. See CREATE DATABASE STRATEGY for more details.

-T *template*
--template=*template*

Specifies the template database from which to build this database. (This name is processed as a double-quoted identifier.)

-V
--version

Print the createdb version and exit.

-?
--help

Show help about createdb command line arguments, and exit.

The options **-D**, **-l**, **-E**, **-O**, and **-T** correspond to options of the underlying SQL command **CREATE DATABASE**; see there for more information about them.

createdb also accepts the following command-line arguments for connection parameters:

-h *host*
--host=*host*

Specifies the host name of the machine on which the server is running. If the value begins with a slash, it is used as the directory for the Unix domain socket.

-p *port*
--port=*port*

Specifies the TCP port or the local Unix domain socket file extension on which the server is listening for connections.

-U *username*
--username=*username*

User name to connect as.

-w
--no-password

Never issue a password prompt. If the server requires password authentication and a password is not available by other means such as a .pgpass file, the connection attempt will fail. This option can be useful in batch jobs and scripts where no user is present to enter a password.

-W
--password

Force createdb to prompt for a password before connecting to a database.

This option is never essential, since createdb will automatically prompt for a password if the server demands password authentication. However, createdb will waste a connection attempt finding out that the server wants a password. In some cases it is worth typing **-W** to avoid the extra connection attempt.

--maintenance-db=*dbname*

Specifies the name of the database to connect to when creating the new database. If not specified, the postgres database will be used; if that does not exist (or if it is the name of the new database being created), template1 will be used. This can be a connection string. If so, connection string parameters will override any conflicting command line options.

ENVIRONMENT

PGDATABASE

If set, the name of the database to create, unless overridden on the command line.

PGHOST
PGPORT
PGUSER

Default connection parameters. **PGUSER** also determines the name of the database to create, if it is not specified on the command line or by **PGDATABASE**.

PG_COLOR

Specifies whether to use color in diagnostic messages. Possible values are always, auto and never.

This utility, like most other PostgreSQL utilities, also uses the environment variables supported by libpq (see Section 34.15).

DIAGNOSTICS

In case of difficulty, see CREATE DATABASE (**CREATE_DATABASE**(7)) and **psql**(1) for discussions of potential problems and error messages. The database server must be running at the targeted host. Also, any default connection settings and environment variables used by the libpq front-end library will apply.

EXAMPLES

To create the database demo using the default database server:

```
$ createdb demo
```

To create the database demo using the server on host eden, port 5000, using the template0 template database, here is the command-line command and the underlying SQL command:

```
$ createdb -p 5000 -h eden -T template0 -e demo
CREATE DATABASE demo TEMPLATE template0;
```

SEE ALSO

dropdb(1), CREATE DATABASE (**CREATE_DATABASE**(7))