

NAME

CALL – invoke a procedure

SYNOPSIS

CALL *name* ([*argument*] [, ...])

DESCRIPTION

CALL executes a procedure.

If the procedure has any output parameters, then a result row will be returned, containing the values of those parameters.

PARAMETERS

name

The name (optionally schema-qualified) of the procedure.

argument

An argument expression for the procedure call.

Arguments can include parameter names, using the syntax *name* => *value*. This works the same as in ordinary function calls; see Section 4.3 for details.

Arguments must be supplied for all procedure parameters that lack defaults, including OUT parameters. However, arguments matching OUT parameters are not evaluated, so it's customary to just write NULL for them. (Writing something else for an OUT parameter might cause compatibility problems with future PostgreSQL versions.)

NOTES

The user must have EXECUTE privilege on the procedure in order to be allowed to invoke it.

To call a function (not a procedure), use **SELECT** instead.

If **CALL** is executed in a transaction block, then the called procedure cannot execute transaction control statements. Transaction control statements are only allowed if **CALL** is executed in its own transaction.

PL/pgSQL handles output parameters in **CALL** commands differently; see Section 41.6.3.

EXAMPLES

CALL do_db_maintenance();

COMPATIBILITY

CALL conforms to the SQL standard, except for the handling of output parameters. The standard says that users should write variables to receive the values of output parameters.

SEE ALSO

CREATE PROCEDURE (**CREATE_PROCEDURE**(7))