LISTEN

LISTEN — listen for a notification

Synopsis

```
LISTEN channel
```

Description

`LISTEN` registers the current session as a listener on the notification channel named *`channel`*. If the current session is already registered as a listener for this notification channel, nothing is done.

Whenever the command `NOTIFY channel` is invoked, either by this session or another one connected to the same database, all the sessions currently listening on that notification channel are notified, and each will in turn notify its connected client application.

A session can be unregistered for a given notification channel with the `UNLISTEN` command. A session's listen registrations are automatically cleared when the session ends.

The method a client application must use to detect notification events depends on which PostgreSQL application programming interface it uses. With the libpq library, the application issues `LISTEN` as an ordinary SQL command, and then must periodically call the function `PQnotifies` to find out whether any notification events have been received. Other interfaces such as libpgtcl provide higher-level methods for handling notify events; indeed, with libpgtcl the application programmer should not even issue `LISTEN` or `UNLISTEN` directly. See the documentation for the interface you are using for more details.

NOTIFY contains a more extensive discussion of the use of `LISTEN` and `NOTIFY`.

Parameters

channel
Name of a notification channel (any identifier).

Notes

`LISTEN` takes effect at transaction commit. If `LISTEN` or `UNLISTEN` is executed within a transaction that later rolls back, the set of notification channels being listened to is unchanged.

A transaction that has executed `LISTEN` cannot be prepared for two-phase commit.

Examples

Configure and execute a listen/notify sequence from psql:

```
LISTEN virtual;
NOTIFY virtual;
Asynchronous notification "virtual" received from server process with PID 8448.
```

Compatibility

There is no `LISTEN` statement in the SQL standard.

See Also

NOTIFY, UNLISTEN

Submit correction

If you see anything in the documentation that is not correct, does not match your experience with the particular feature or requires further clarification, please use this form to report a documentation issue.

Copyright © 1996-2018 The PostgreSQL Global Development Group