

NAME

`pg_isready` - check the connection status of a PostgreSQL server

SYNOPSIS

pg_isready [*connection-option...*] [*option...*]

DESCRIPTION

`pg_isready` is a utility for checking the connection status of a PostgreSQL database server. The exit status specifies the result of the connection check.

OPTIONS

-d *dbname*

--dbname=*dbname*

Specifies the name of the database to connect to. The *dbname* can be a connection string. If so, connection string parameters will override any conflicting command line options.

-h *hostname*

--host=*hostname*

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. Defaults to the value of the **PGPORT** environment variable or, if not set, to the port specified at compile time, usually 5432.

-q

--quiet

Do not display status message. This is useful when scripting.

-t *seconds*

--timeout=*seconds*

The maximum number of seconds to wait when attempting connection before returning that the server is not responding. Setting to 0 disables. The default is 3 seconds.

-U *username*

--username=*username*

Connect to the database as the user *username* instead of the default.

-V

--version

Print the pg_isready version and exit.

-?

--help

Show help about pg_isready command line arguments, and exit.

EXIT STATUS

pg_isready returns 0 to the shell if the server is accepting connections normally, 1 if the server is rejecting connections (for example during startup), 2 if there was no response to the connection attempt, and 3 if no attempt was made (for example due to invalid parameters).

ENVIRONMENT

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

The environment variable **PG_COLOR** specifies whether to use color in diagnostic messages. Possible values are always, auto and never.

NOTES

It is not necessary to supply correct user name, password, or database name values to obtain the server status; however, if incorrect values are provided, the server will log a failed connection attempt.

EXAMPLES

Standard Usage:

```
$ pg_isready
/tmp:5432 - accepting connections
$ echo $?
0
```

Running with connection parameters to a PostgreSQL cluster in startup:

```
$ pg_isready -h localhost -p 5433
localhost:5433 - rejecting connections
$ echo $?
1
```

Running with connection parameters to a non-responsive PostgreSQL cluster:

```
$ pg_isready -h someremotehost  
someremotehost:5432 - no response  
$ echo $?  
2
```