

Getting started with PostgreSQL

Jens Lechtenbörger

Winter Term 2023/2024

1 Installation

PostgreSQL is a free/libre and open source object-relational database system. Lots of [alternatives](#) exist. If you wonder whether it is still a good idea to learn about PostgreSQL, please note that major developments based on that system are ongoing, e.g., [Neon](#) and [CockroachDB](#) as [serverless](#) databases or [Greenplum](#) with a focus on analytics.

On GNU/Linux, some version of PostgreSQL is probably available as package in your distribution (e.g., `sudo apt install postgresql`). [Other versions can be installed as well](#).

For other operating systems, please check out the [download page](#).

If you are familiar with [Docker](#), images exist as well (`docker pull postgres`).

1.1 Notes on Windows

I do not recommend Windows. I recommend [GNU/Linux](#).

I use PostgreSQL via its command line tool `psql`. On Windows, this needs to be activated manually. (In Windows terminals, executable programs such as `psql` are located by searching through various directories, in particular those listed in the system environment variable `Path`.)

Install PostgreSQL as instructed on the download page mentioned above. This asks you for (a) an installation directory and (b) a password for the database user `postgres`. Remember both.

Add the `bin` sub-directory with executable files to your `Path` environment variable so that you can execute them from a terminal: Open the installation directory in an explorer window, change to the `bin` sub-directory. Copy its location, add it to the `Path` system environment variable: Open System settings (e.g., right-click on Windows symbol), search for environment (umgebung), change system environment variables, locate system variable “Path”, add a new entry with the copied location.

Open a terminal (WindowsKey+R, then type `cmd`). Type `psql`. If that asks you for a password, everything is fine. By default, user `postgres` exists, with the password you remembered above.

2 First Commands

- Connect to database as user `postgres`: `psql -U postgres`

- Type `help` for help, e.g.:
 - `\?` to see built-in commands (such as `\dt`)
- Type SQL commands, ending with “;”, e.g., `create table ...; or select * from ... ;`
- Create database user and new database
 - PostgreSQL maintains its own user accounts, largely independently of OS level user accounts.
 - Create new database user: `createuser -U postgres -drS newUserName`
 - * You probably want to use your OS user name in place of `newUserName` as that is used by default when not supplying a username with option `-U`
 - Create new database owned by previously created user: `createdb -U postgres -O newUserName newDatabase`
 - (On GNU/Linux maybe use `sudo -u postgres bash` first, then invoke above commands without the option `-U postgres`)
- To execute SQL instructions stored in a file, e.g., `backup.sql`, use `psql` with option `-f backup.sql`. E.g., to execute SQL commands inside the previously created database (where the OS user name is also a database user):

```
psql newDatabase -f backup.sql
```

License Information

Source files are available on [GitLab](#) (check out embedded submodules) under free licenses. Icons of custom controls are by [@fontawesome](#), released under [CC BY 4.0](#).

Except where otherwise noted, the work “Getting started with PostgreSQL”, © 2020, 2022, 2024-2025 [Jens Lechtenbörger](#), is published under the [Creative Commons license CC BY-SA 4.0](#).