

# Базовый инструментарий Использование psql



## **Авторские права**

© Postgres Professional, 2017 год.

Авторы: Егор Рогов, Павел Лузанов

## **Использование материалов курса**

Некоммерческое использование материалов курса (презентации, демонстрации) разрешается без ограничений. Коммерческое использование возможно только с письменного разрешения компании Postgres Professional. Запрещается внесение изменений в материалы курса.

## **Обратная связь**

Отзывы, замечания и предложения направляйте по адресу:

[edu@postgrespro.ru](mailto:edu@postgrespro.ru)

## **Отказ от ответственности**

Компания Postgres Professional не несет никакой ответственности за любые повреждения и убытки, включая потерю дохода, нанесенные прямым или непрямым, специальным или случайным использованием материалов курса. Компания Postgres Professional не предоставляет каких-либо гарантий на материалы курса. Материалы курса предоставляются на основе принципа «как есть» и компания Postgres Professional не обязана предоставлять сопровождение, поддержку, обновления, расширения и изменения.

Запуск psql и подключение к БД

Получение справочной информации

Работа в psql

Настройка

Терминальный клиент для работы с PostgreSQL

Поставляется вместе с СУБД

Используется администраторами и разработчиками для интерактивной работы и выполнения скриптов

Для работы с СУБД PostgreSQL существуют различные сторонние инструменты, рассмотрение которых не входит в рамки курса.

В курсе мы будем использовать терминальный клиент psql:

1. psql — это единственный клиент, поставляемый вместе с СУБД.
2. Навыки работы с psql пригодятся разработчикам и администраторам БД вне зависимости от того, с каким инструментом они будут работать дальше.

<https://postgrespro.ru/docs/postgresql/10/app-psql.html>

## Запуск

```
$ psql -d база -U роль -h узел -p порт
```

## Новое подключение в psql

```
=> \c[onnect] база роль узел порт
```

## Информация о текущем подключении

```
=> \conninfo
```

При запуске psql нужно указать параметры подключения.

К обязательным параметрам подключения относятся: имя базы данных, имя пользователя, имя сервера, номер порта. Если эти параметры не указаны, psql попытается подключиться, используя значения по умолчанию:

- *база* — совпадает с именем пользователя;
- *роль* — совпадает с именем пользователя ОС;
- *узел* — локальное соединение;
- *порт* — обычно 5432.

Значения по умолчанию позволяют пользователю postgres подключаться к PostgreSQL без указания параметров.

Если требуется выполнить новое подключение не выходя из psql, то нужно выполнить команду `\connect`.

Команда `\conninfo` выдает информацию о текущем подключении.

Дополнительная информация о возможностях настройки подключения:

<https://postgrespro.ru/docs/postgresql/10/libpq-envvars.html>

<https://postgrespro.ru/docs/postgresql/10/libpq-pgservice.html>

<https://postgrespro.ru/docs/postgresql/10/libpq-pgpass.html>

## В командной строке ОС

```
$ psql --help  
$ man psql
```

## В psql

=> \?	список команд psql
=> \? variables	переменные psql
=> \h[elp]	список команд SQL
=> \h <i>команда</i>	синтаксис команды SQL
=> \q	выход

Справочную информацию по psql можно получить не только в документации, но и прямо в системе.

psql с ключом `--help` выдает справку по запуску. А если в системе была установлена документация, то справочное руководство можно получить командой `man psql`.

psql умеет выполнять команды SQL и свои собственные команды.

Внутри psql есть возможность получить список и краткое описание команд psql. Все команды psql начинаются с обратной косой черты (`\`).

Команда `\help` выдает список команд SQL, которые поддерживает сервер, а также синтаксис конкретной команды SQL.

И еще одна команда, которую полезно знать, хоть она и не имеет отношения к справке, это `\q` — выход из psql.

## Скрипты при запуске psql

- общий системный файл `psqlrc`
- пользовательские файлы `~/.psqlrc`

## История команд

- пользовательские файлы `~/.psql_history`
- поиск по истории с помощью `readline`
- количество хранимых команд можно изменить

При запуске `psql` выполняются команды, записанные в двух файлах — общесистемном и пользовательском (если они есть, конечно).

Общий системный файл называется `psqlrc` и располагается в каталоге `/usr/local/pgsql/etc` при обычной сборке из исходных кодов.

Расположение этого каталога можно узнать командой `pg_config --sysconfdir`.

Пользовательский файл находится в домашнем каталоге пользователя ОС и называется `.psqlrc`. Его расположение можно изменить, задав переменную окружения `PSQLRC`.

В эти файлы можно записать команды, настраивающие `psql` — например, изменить приглашение, включить вывод времени выполнения команд и т. п.

История команд сохраняется в файле `.psql_history` в домашнем каталоге пользователя. Расположение этого файла можно изменить, задав переменную окружения `PSQL_HISTORY` или переменную `psql HISTFILE`. По умолчанию хранится 500 последних команд; это число можно изменить переменной `psql HISTSIZE`.

Пролистывать историю команд можно стрелками вверх и вниз, искать с помощью `Ctrl+R` — доступен весь набор команд, предлагаемых `readline`,



Psql — терминальный клиент для работы с СУБД

При запуске требуются параметры подключения

Выполняет команды SQL и psql

Средства для интерактивной работы, а также подготовки и выполнения скриптов



1. Запустите `psql` и проверьте информацию о текущем подключении.
2. Выведите строки таблицы `pg_tables`.
3. Установите команду «`less -XS`» для постраничного просмотра и еще раз выведите все строки `pg_tables`.
4. Настройте `psql` так, чтобы для каждой команды печаталось время ее выполнения. Убедитесь, что при повторном запуске эта настройка сохраняется.
5. Приглашение по умолчанию показывает имя базы данных. Настройте приглашение так, чтобы дополнительно выводилась информация о пользователе: `роль@база=#`

1. Здесь и далее до конца курса мы работаем под пользователем `postgres`:

```
$ sudo su - postgres
```

При запуске `psql` можно не указывать параметры подключения, будут действовать значения по умолчанию.

Если PostgreSQL был остановлен, то его нужно вновь запустить.

3. Настройку переменной окружения `PAGER` лучше сделать в файле `.psqlrc`. Используйте команду `\setenv`, чтобы установить переменную окружения ОС. Это позволит использовать значение «`less -XS`» только при работе в `psql`. Для всех остальных команд ОС будут использоваться настройки в ОС (например, из `.profile`).

4. Команду `psql` для вывода времени выполнения запросов можно найти в документации или командой `\?`.

5. Описание настройки приглашения в документации:

<https://postgrespro.ru/docs/postgresql/10/app-psql.html#app-psql-prompting>