*The following guide is for getting PostgreSQL installed and up and running without admin rights (running locally). You will need to download a ZIP file and extract the contents to your local disk.*

*Download link and sample install path are given below.*

*Binaries from installer* ***Version 9.6.14*** *were used (there could be some problems on Windows with version 10).*

# Installation and Setup

Download the ZIP file from <https://www.enterprisedb.com/products-services-training/pgbindownload>

Unzip the archive into a directory of your choice

PostgreSLQ located: \Local\PostgreSQL

Open cmd prompt

Set path using temporarily

> **set PATH=%PATH%;Local\PostgreSQL\pgsql\bin**

Run initdb (this can be found in the subdirectory pgsql\bin)

initdb creates a new PostgreSQL database cluster. A database cluster is a collection of databases that are managed by a single server instance.

> **initdb -D Local\PostgreSQL \pgdata -U postgres -W -E UTF8**

\pgdata should be an empty directory you created

Enter your password

If this ran correctly, \pgdata will now have a bunch of files and folders in it

The initdb step only needs to be done once (provided you don’t delete pgdata). Skip to next step to start the server if you’ve already created a database cluster for yourself.

To start Postgres, run:

> **"pg\_ctl" -D "Local\PostgreSQL\pgdata" -l logfile start**

This has (!) to be done as the user who ran initdb to avoid any problems with the access to the data directory.

To shutdown Postgres:

> **"pg\_ctl" -D "Local\PostgreSQL\pgdata" -l logfile stop**

# Creating a database from scratch

To create a database, run

> **created mydb**

If this produces no response then this step was successful and you can skip over the remainder of this section.

If you get the ERROR:

createdb: could not connect to database template1: FATAL: role "USER" does not exist

**OPTION 1:**

> **psql -U postgres**

The prompt should turn to postgres=#

# **CREATE USER usr.name;**

# **ALTER USER usr.name SUPERUSER CREATEDB;**

# **\du**

# **\q**

\q should return you to the original prompt >

Modify createdb command

> **createdb -U usr.name mydb**

**OPTION 2:**

Skip the step of creating your username with Create DB privilege and just use postgres

> **createdb –U postgres mydb**

You can remove your database with

> **dropdb –U usr.name mydb**

Or

> **dropdb –U postgres mydb**

To check the list of your databases, return to the postgres=# prompt

> **psql -U postgres**

# **\l**

You should see mydb in the list

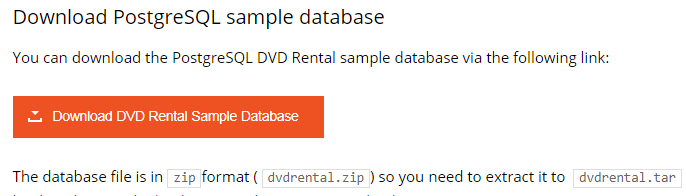
To connect to the database

# **\c mybd**

# Load sample database into the PostgreSQL database server

Downloaded dvd sample data here: <http://www.postgresqltutorial.com/postgresql-sample-database/>

* Scroll down towards bottom of page and look for



Saved sample data here: Local\PostgreSQL\sample

You need to create a new database in the PostgreSQL database server before loading database schema and data into the database. See previous steps above.

Navigate to the folder with the dvd sample data

> **cd Local\PostgreSQL\sample**

> **dir**

Verify dvdrental.tar exists

Load data into mydb database

> **pg\_restore -U usr.name -d mydb Local\PostgreSQL\sample\dvdrental.tar**

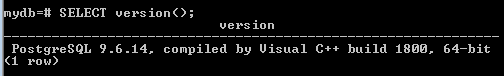
# Access your database

> **psql –U usr.name mydb**

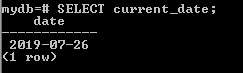
Prompt should change to mydb=#

Try out these commands:

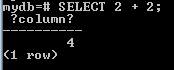
# **SELECT version();**



# **SELECT current\_date;**



# **SELECT 2 + 2;**



To get help

# **\h**

# SQL Language (Using psql)

Create a tutorial folder and cd to it

> **psql -U usr.name -s mydb**

The -s option puts you in single step mode which pauses before sending each statement to the server.

Connect to mydb

# **\c mydb**

List tables in mydb using psql

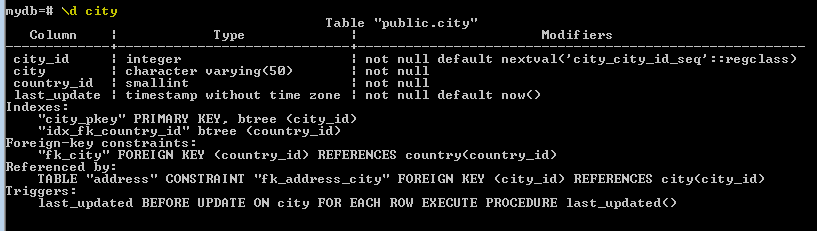
# **\d**

# **\dt**

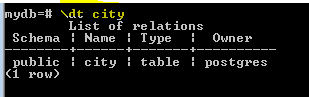
# **\dt \*.\***

Find the information on columns of a table (e.g., city) using \d

# **\d city**



# **\d+ city**



Find the information on columns of a table (e.g., city) using information\_schema

**SELECT**

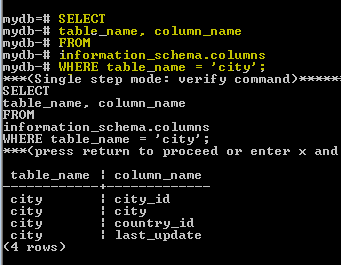
**table\_name, column\_name**

**FROM**

**information\_schema.columns**

**WHERE**

**table\_name = 'city';**

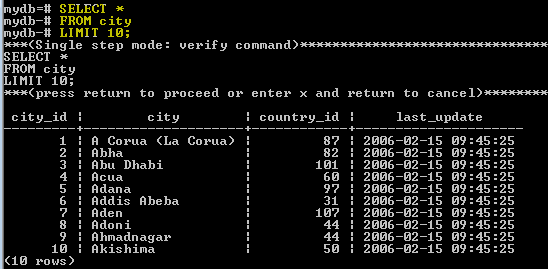


Show top 10 rows in a table

SELECT \*

FROM city

LIMIT 10;



PostgreSQL DESCRIBE TABLE using psql

Additional commands of interest:

<https://www.w3resource.com/PostgreSQL/connect-to-postgresql-database.php>

Here are some common psql commands

* To view help for psql commands, type \?.
* To view help for SQL commands, type \h.
* To view information about the current database connection, type \conninfo.
* To list the database's tables and their respective owners, type \dt.
* To list all of the tables, views, and sequences in the database, type \z.
* To exit the psql program, type \q.

# Connect to Tableau Server

Use computer name in server block:

Use username:

Change

Local\PostgreSQL\pgdata\pg\_hba.conf

* Change trust to md5
* Add line at EOF: host all all 0.0.0.0/0 md5

Local\PostgreSQL\pgdata\postgresql.conf

* Change #listen\_addresses = ‘localhost’ to listen\_address = ‘\*’

Restart server if needed:

> **"pg\_ctl" -D "Local\PostgreSQL\pgdata" -l logfile restart**

<https://stackoverflow.com/questions/5598517/find-the-host-name-and-port-using-psql-commands>

<https://stackoverflow.com/questions/34627661/postgresql-9-3-query-to-get-hostname-port-number-and-username>

restart=# SELECT CURRENT\_USER usr

restart-# ,inet\_server\_addr() host

restart-# ,inet\_server\_port() port;

usr | host | port

-----+------+------

usr.name | ::1 | 5432

(1 row)

OR

restart=# SELECT \*

restart-# FROM pg\_settings

restart-# WHERE name = 'port';

# Connect to Tableau Desktop

Use ***localhost***

Otherwise, connecting to Postgres should be the same as connecting to the Server