CSCI 5800 – Big Data Systems

Prelab #7 - Time Series Analysis

Table of Contents

[Installing PostgreSQL 12.1 2](#_Toc511802439)

[Installing TimescaleDB 3](#_Toc511802440)

[Create Databases and Import Data 4](#_Toc511802441)

[Connect to PostgreSQL 5](#_Toc511802442)

***Disclaimer***

*We build this guide for you to install the necessary tools for the assignment on time series. This guide assumes you are working on the Ubuntu VM you used for the Lab06 and Assignment #6. If you are not using that VM, please refer to the software manuals and this guide to install the necessary systems on your computer.*

# Installing PostgreSQL 12.1

<https://www.postgresql.org/>

1. TimeScaleDB needs PostgreSQL 12.1. Open a terminal (Make sure you are root):

sudo apt-get install wget ca-certificates

wget --quiet -O - https://www.postgresql.org/media/keys/ACCC4CF8.asc | sudo apt-key add –

sudo sh -c echo "deb http://apt.postgresql.org/pub/repos/apt/ 'lsb\_release -cs'-pgdg main" >> /etc/apt/sources.list.d/pgdg.list

sudo apt-get update

1. Check PostgresSQL version

apt show postgresql

1. Install PostgreSQL package

sudo apt-get install postgresql postgresql-contrib

1. Start the PostgreSQL service using:

sudo service postgresql start

5.Check PostgreSQL service status using:

sudo service postgresql status

6. Change user to Postgres:

sudo su – postgres

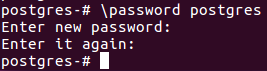
7. Connect to Postgres

psql

8. Set a password for your instance (use an easy password like 123123123 so you do not forget it):

\password postgres

* 1. Enter a password and press enter. Then re-enter the new password and press enter again (password will not be displayed on the screen for security purposes).



1. Quit PostgreSQL using:

\q

# Installing TimescaleDB

<https://www.timescale.com/>

<https://docs.timescale.com/timescaledb/latest/how-to-guides/install-timescaledb/self-hosted/ubuntu/installation-apt-ubuntu/>

1. Install TimeSclaeDB using the apt package:

*sudo sh -c "echo 'deb [signed-by=/usr/share/keyrings/timescale.keyring] https://packagecloud.io/timescale/timescaledb/ubuntu/ $(lsb\_release -c -s) main' > /etc/apt/sources.list.d/timescaledb.list"*

*wget --quiet -O - https://packagecloud.io/timescale/timescaledb/gpgkey | sudo gpg --dearmor -o /usr/share/keyrings/timescale.keyring*

*sudo apt-get update*

*# Now install appropriate package for PG version*

*sudo apt install timescaledb-2-postgresql-12*

1. Now we need to modify PostgreSQL settings:

*cd /etc/postgresql/12/main*

*sudo nano postgresql.conf*

* 1. Add this line *in postgresql.conf (Line number doesn’t matter in this case)*:

Add s*hared\_preload\_libraries = 'timescaledb'*

Text

Description automatically generated

2.2. Save the file and close it.

1. Restart PostgreSQL instance

*sudo service postgresql restart*

# Create Databases and Import Data

1. Connect to PostgreSQL:

*psql -U postgres -h localhost*

Text

Description automatically generated

1. Create a database named *nyc\_data*:

CREATE DATABASE nyc\_data;

\c nyc\_data

CREATE EXTENSION IF NOT EXISTS timescaledb CASCADE;

1. Now Quit PostgreSQL using:

*\q*

1. Download the provided csv and sql files from Canvas. Then unzip them and copy them into your home directory. We have reduced the size of dataset for you and it now contains data for 5 days (more than 1M records).
2. Create tables using the sql file

*psql -U postgres -d nyc\_data -h localhost < /home/<user>/Downloads/nyc\_data.sql*

Text

Description automatically generated

Import data from the csv file (it may take several minutes):

*psql -U postgres -d nyc\_data -h localhost -c "\COPY rides FROM /home/<user>/Downloads/nyc\_data\_rides.csv CSV”*

# 

# Connect to PostgreSQL

Open PostgreSQL and open the nyc\_data database:

*psql -U postgres -h localhost -d nyc\_data*

Then use the following command to see the list of tables:

*\dt*

Text

Description automatically generated

Now use the following command to see the columns of *rides* table (you can press *q* for going back to the terminal):

*\d rides*

Text

Description automatically generated with low confidence