Manjaro Linux

How to install pgAdmin4 on Manjaro Linux

11 months ago • by Adnan Shabbir

The **pgAdmin4** is a client for PostgreSQL databases whereas PostgreSQL is an advanced Object-Relational database management system. The **pgAdmin4** eases the management of PostgreSQL by providing a visual interface. The notable use of the **pgAdmin4** tool is to execute queries, read results, update data, and create databases.

The **pgAdmin4** tool can be obtained on Linux, macOS, and Windo website. However, the available setup support cannot be used to i Manjaro Linux. In this article, we have compiled an installation anc on Manjaro.

As pgAdmin4 is meant to serve as a graphical interface for managing Postgres databases. That means an installation and configuration of PostgreSQL is necessary to carry out. For this, follow the steps provided below.

Step 1 : Install the PostgreSQL server on Manjaro, as it is necessary to run pgAdmin4 on your system.

```
$ sudo pacman -S postgresql
```

Step 2 : Login as a postgres user with the help of command provided below.

```
$ sudo -u postgres -i
```

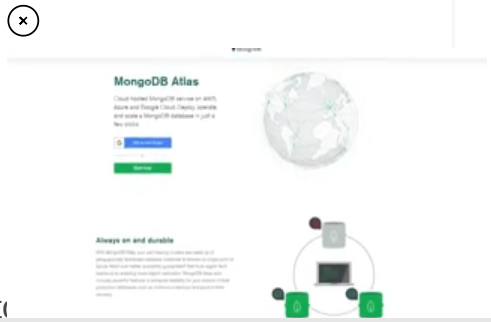
After logging into postgres, you have to create a PostgreSQL clust
stated command

```
$ initdb --locale $LANG -E UTF8 -D '/var/lib/postgres/data/'
```


Step 3 : Restart your terminal and then use the following comman

After enabling, check the status of the service to verify its running or not.

```
$ sudo systemctl status postgresql
```



Step 4 : This step comprises several commands that are related t

After accessing the postgres shell, type "**\password**" to set a password. Choose the password and enter again to confirm:

```
\password
```

Now, come out of postgres shell by typing "**\q**". It is recommended to note down the username and password as it would be required in connecting with pgAdmin4

How to install and use pgAdmin4 on Manjaro Linux

The installation and usage of pgAdmin4 are linked with creating the virtual environment of pgAdmin4. So, in this section, a step-by-step procedural guide will Python-based virtual environment of pgAdmin4 and then use it to related operations.

Step 1 : Make the essential directories, **/var/lib** and **/var/log** for **p**

```
$ sudo mkdir /var/lib/pgadmin
$ sudo mkdir /var/log/pgadmin
```

Step 2 : Change the owner of the files using the commands provided below.

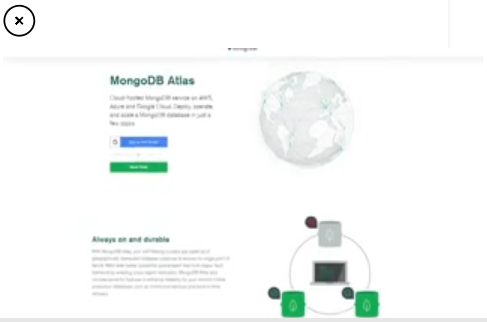
```
$ sudo chown $USER /var/lib/pgadmin
$ sudo chown $USER /var/log/pgadmin
```

Step 3 : Create the Python-based virtual environment by issuing the command written below.

```
$ python3 -m venv pgadmin4
```

Activate the environment with the help of the command written below.

```
$ source pgadmin4/bin/activate
```



Step 4 : Now run the following command to install pgAdmin4

```
$ pip install pgadmin4
```

Step 5 : Navigate to the pgAdmin4 and start the pgAdmin4 service by using the command provided below.

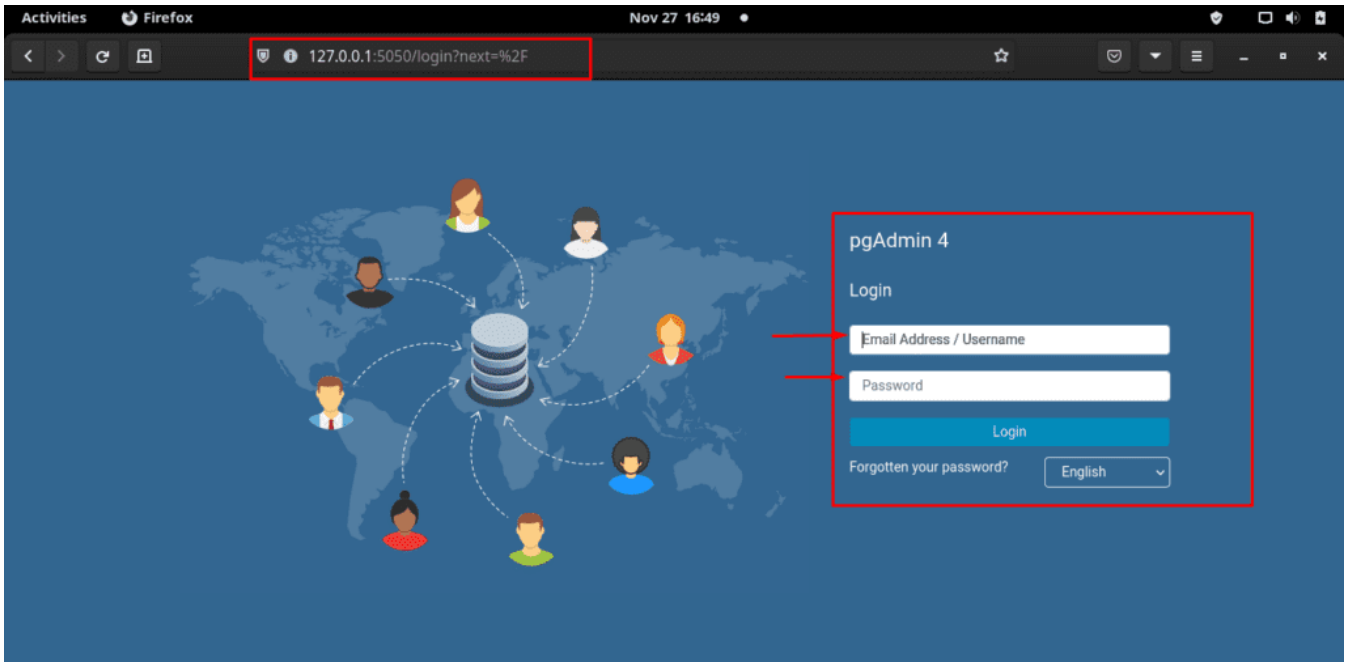
```
~ cd pgadmin4
~/pgadmin4
```

After executing the below mentioned command, you will be asked to enter the email address and password that will be used to log in to the web interface.

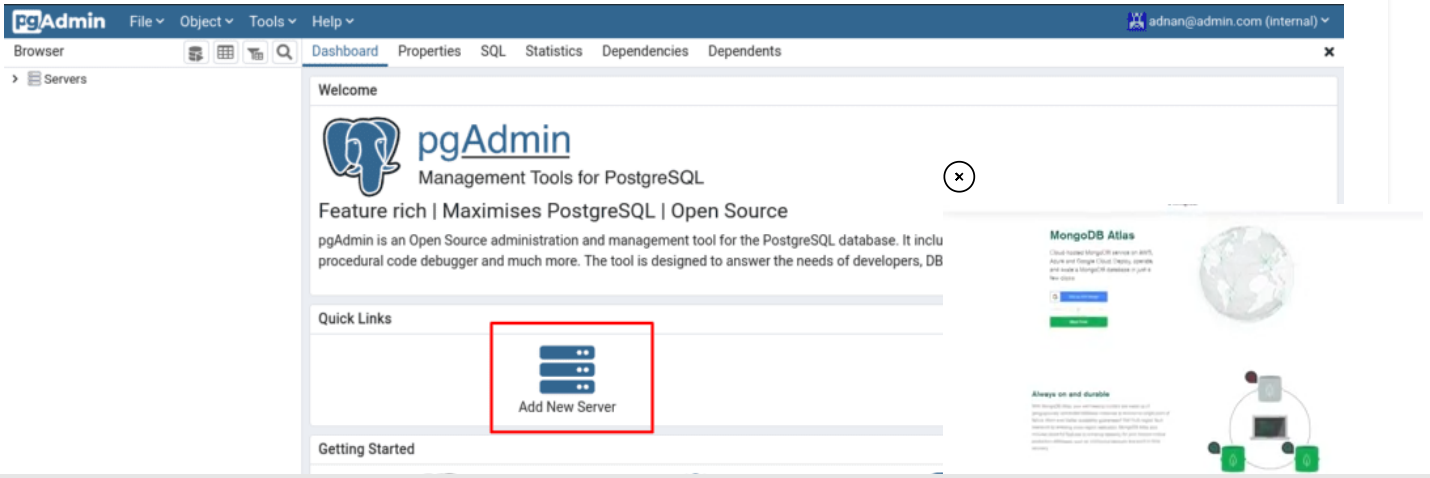
```
$ pgadmin4
```

```
~/pgadmin4 pgadmin4                                    pgadmin4
NOTE: Configuring authentication for SERVER mode.

Enter the email address and password to use for the initial pgAdmin user account
:

Email address: adnan@admin.com
Password:
Retype password:

Starting pgAdmin 4. Please navigate to http://127.0.0.1:5050 in your browser.
2021-11-27 16:44:06,726: WARNING        werkzeug:       WebSocket transport not
available. Install simple-websocket for improved performance.
 * Serving Flask app "pgadmin" (lazy loading)
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deployme
nt.
   Use a production WSGI server instead.
 * Debug mode: off
```

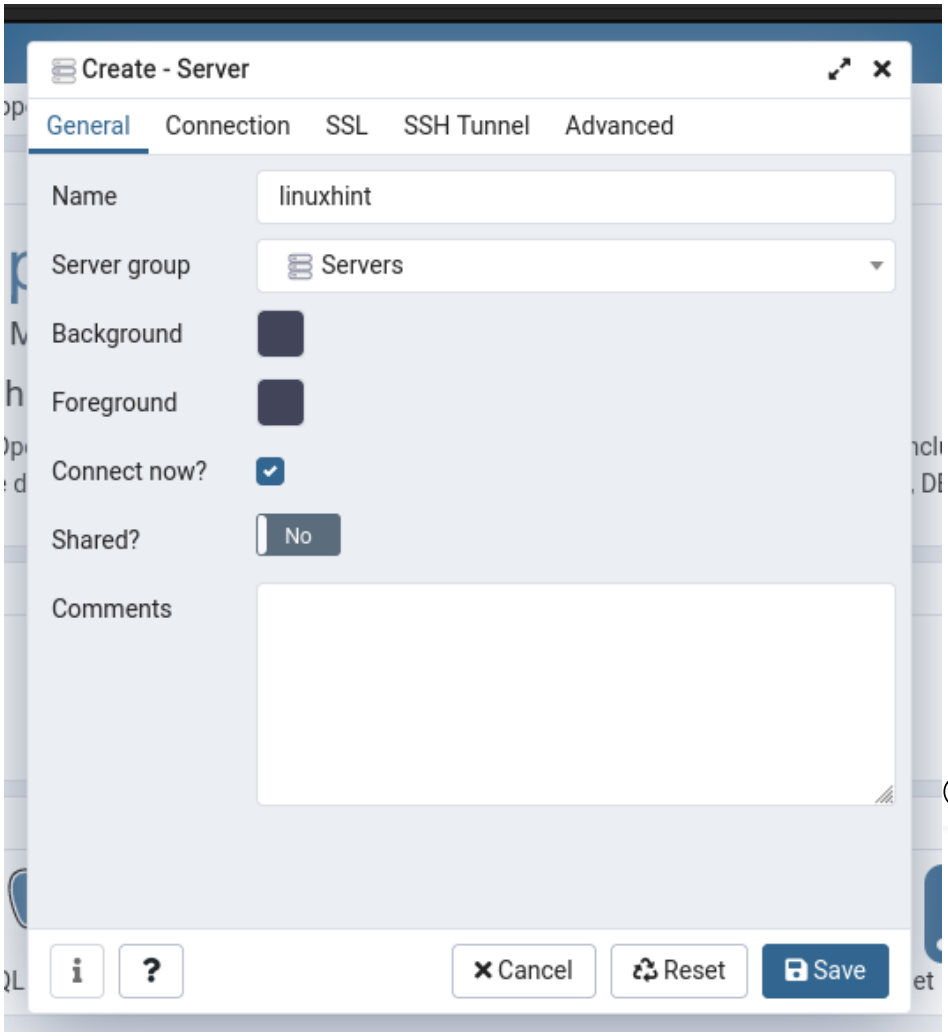
You must observe that the output is notifying you to navigate to ad **https://127.0.0.1:5050.** Enter the address in any browser and you following interface. Where a login prompt will appear. Use the ema chose in the above steps.

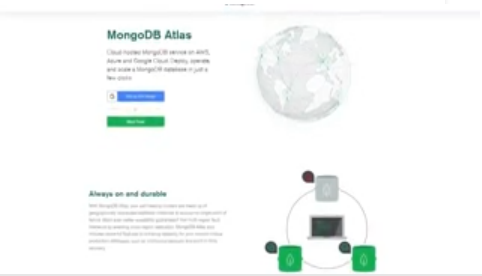


After successful login, the interface is provided below. Click on “**Add New Server**” option to add a new server for your postgres databases.



In the “**General**” tab, enter the server’s name, for instance we have set it to “**linuxhint**“.

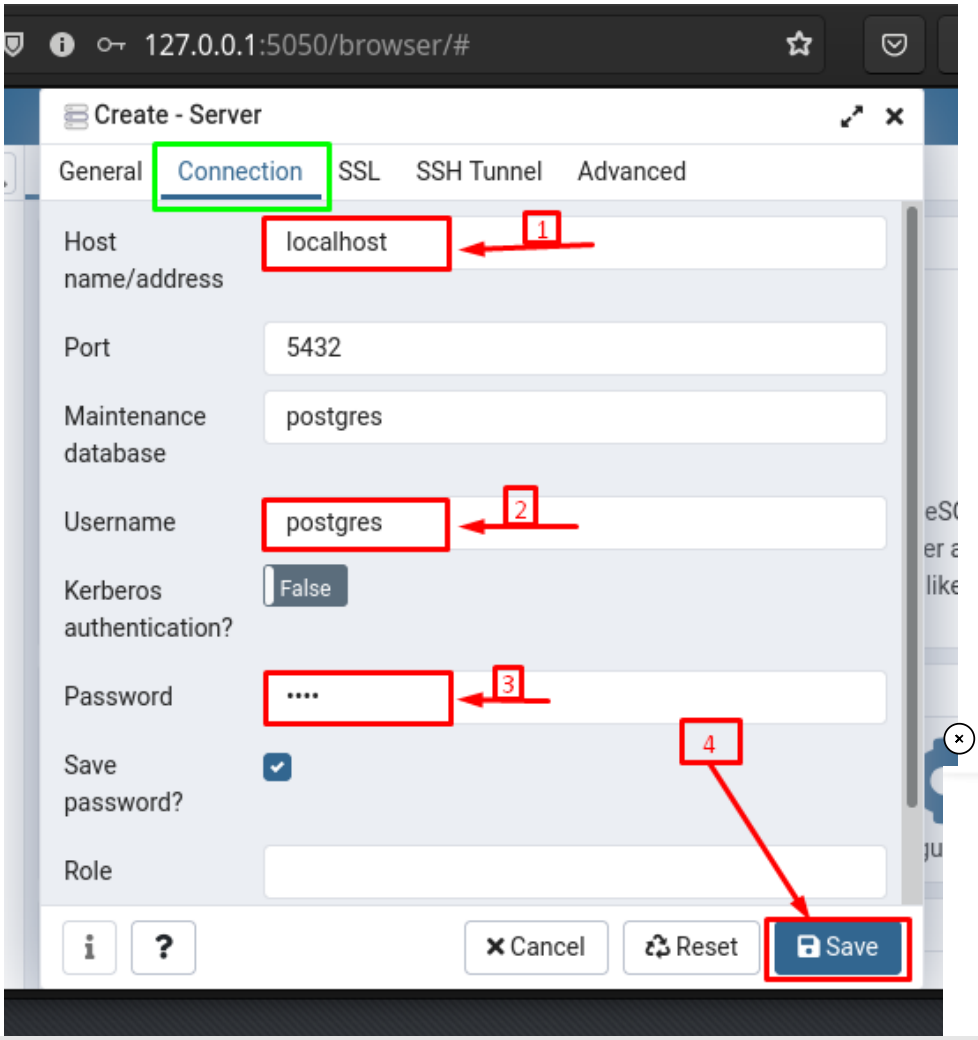




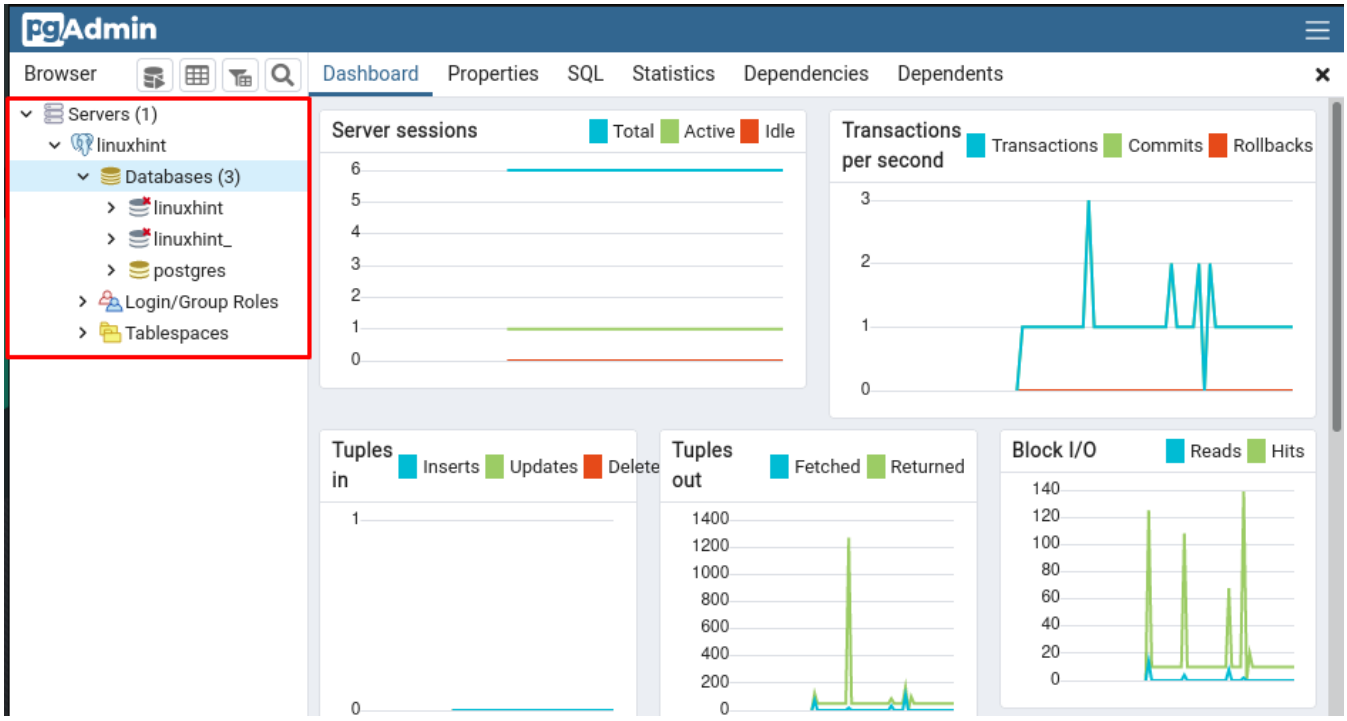
For further configurations, navigate to the next tab named “**Connection**“, and here you have to enter the details in the following options:

- Enter the Hostname or address that would be “localhost”
- Provide the Username and Password of the postgres database user. For this, refer to **step 4** of the **Prerequisites** section.

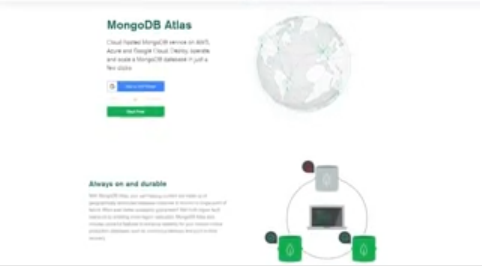
After that, navigate to “**Save**” option;

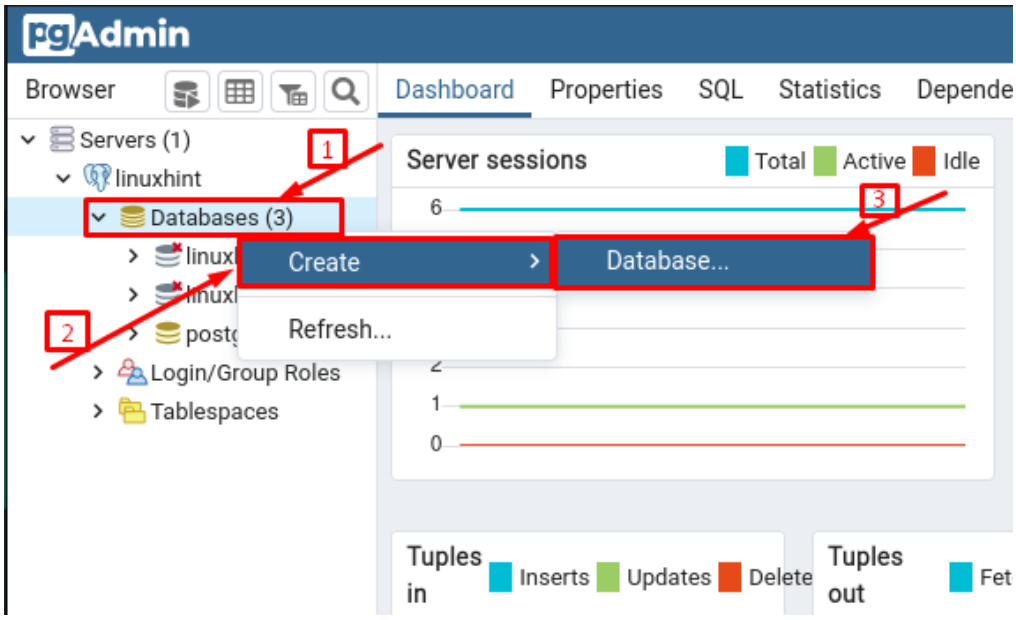


Once it is saved, the “**linuxhint**” server can be seen on the “**Dashboard**” of pgAdmin4. Under the “**linuxhint**” tab you would observe the default databases, the login roles for the users.

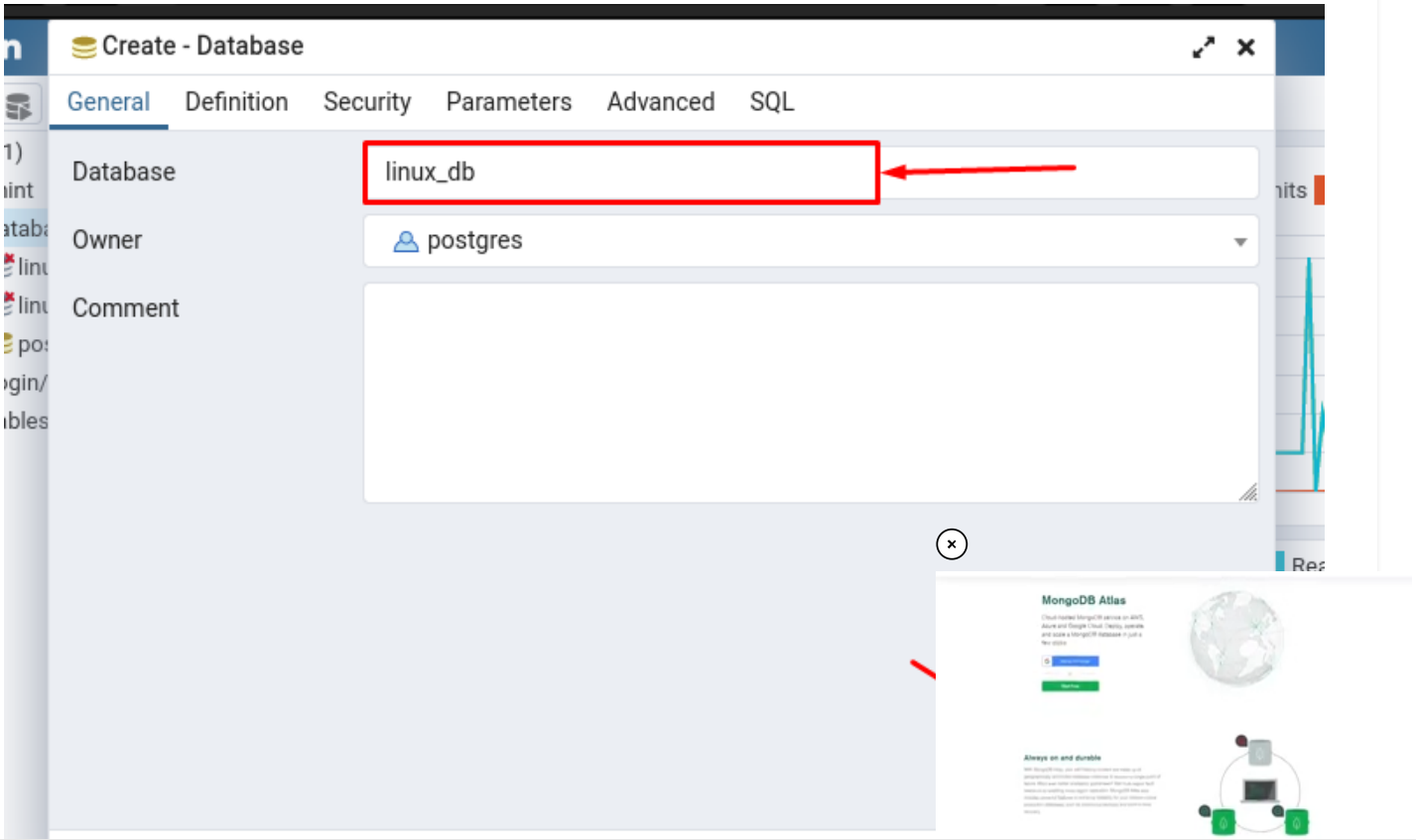


Create a database using pgAdmin4 : To create a database, righ and then navigate to “**Create**” followed by the “**Database…**” optio

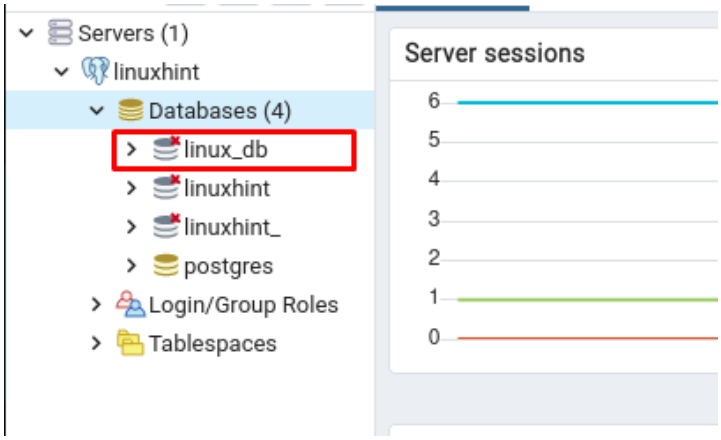




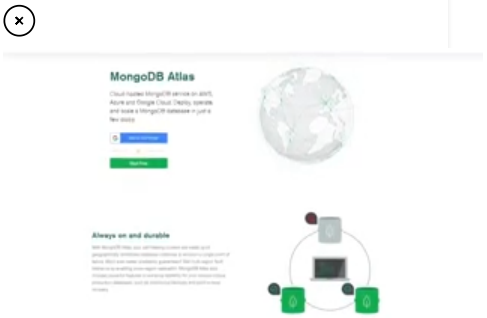
Choose the name and click on “**Save**” to create that database.

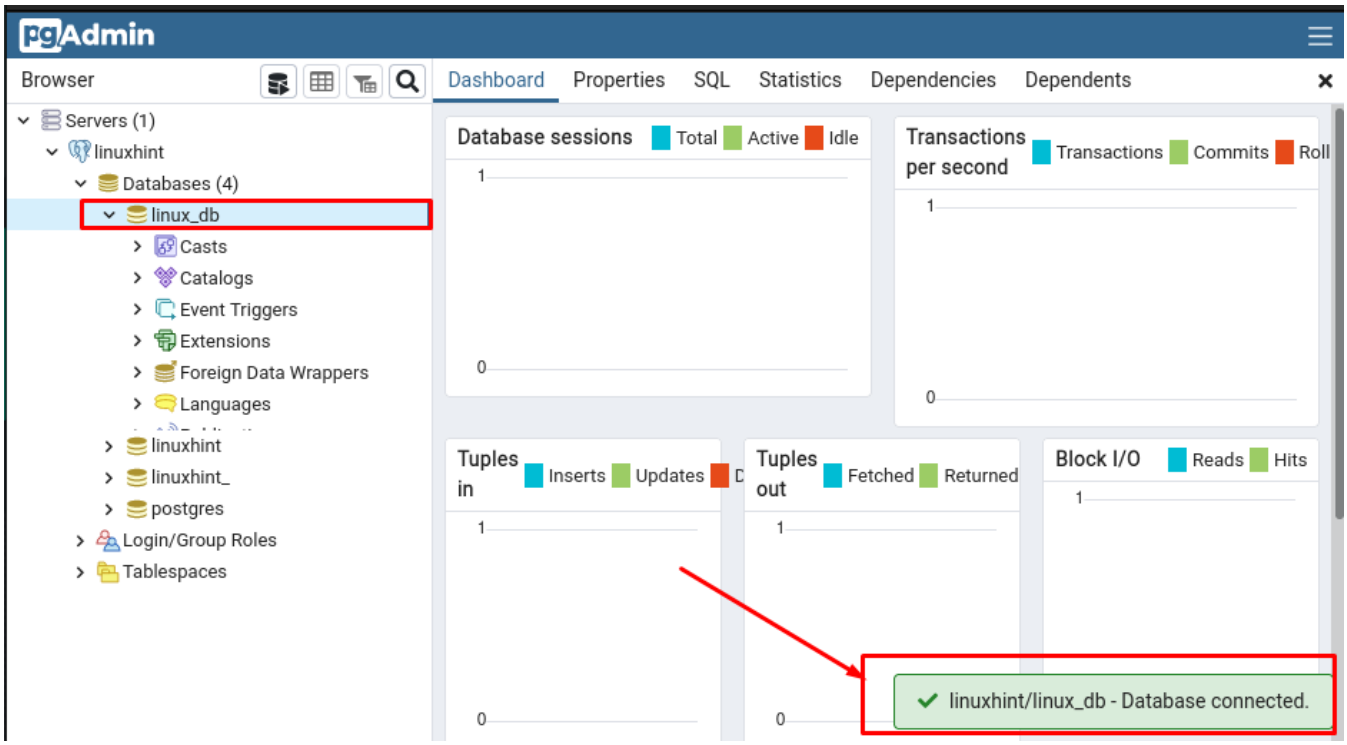


You would observe “**linux_db**” is created but not connected yet:



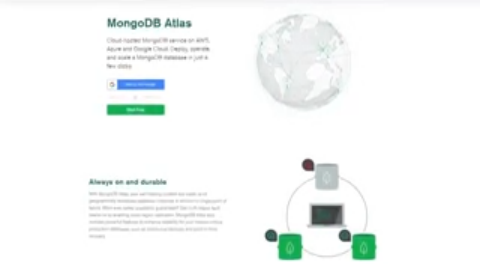
To connect the database with the server, click on the database name and you will receive a database connected prompt that shows your database has been connected.

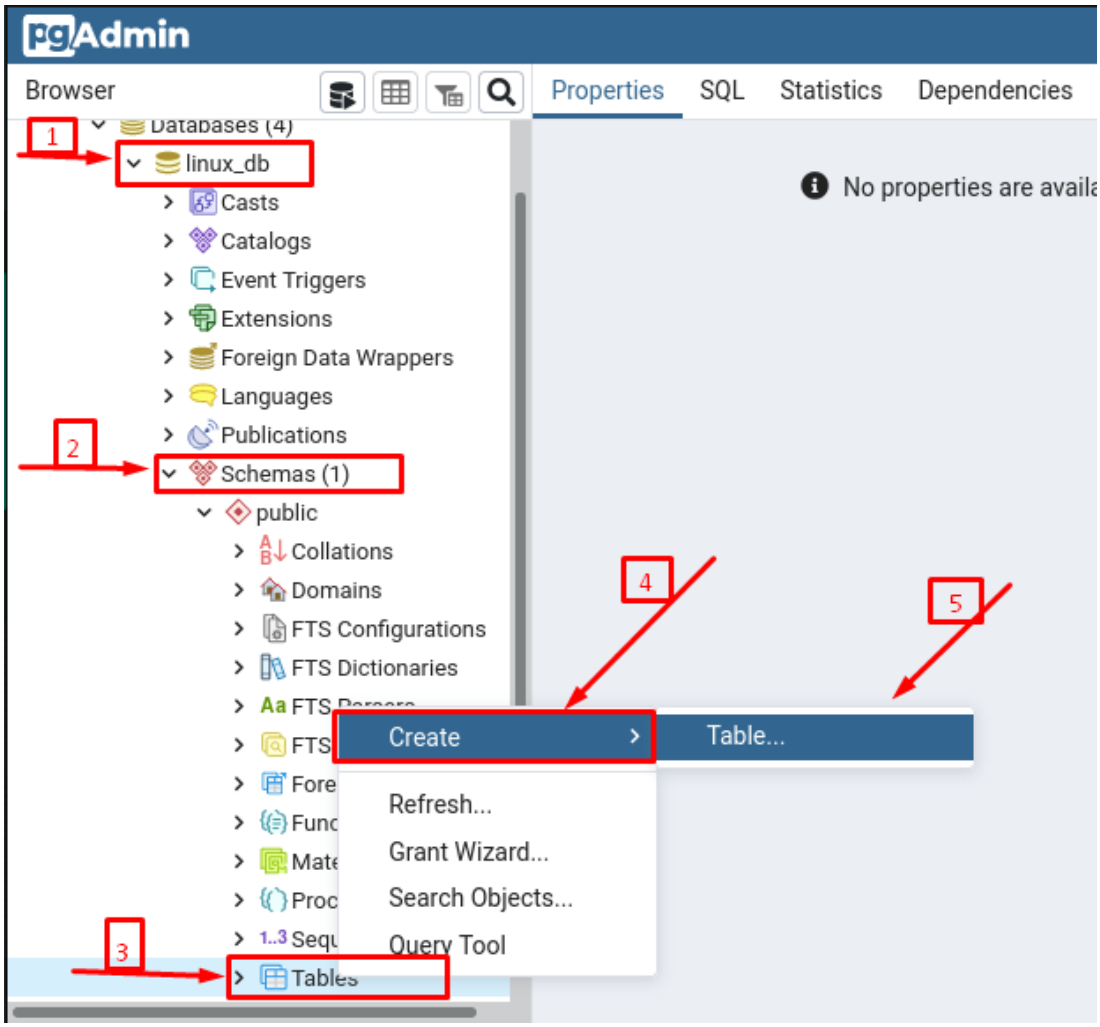




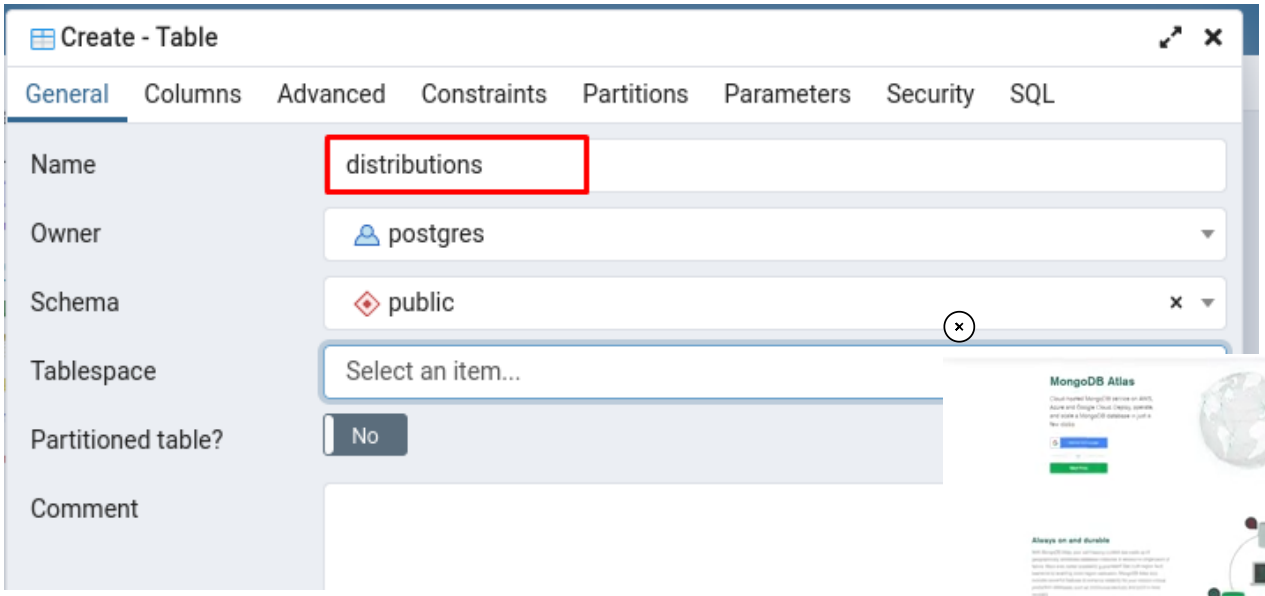
Create a table in the database using pgAdmin : For the creation of tables, you must locate the tables option in that database. To do so, perform the following actions:

- Click on the database and then look for “**Schemas**” in that drop down menu.
- In the Schemas, open the “**public**” schema and scroll down, you would see the “**Tables**” option and right click on it to create new table.



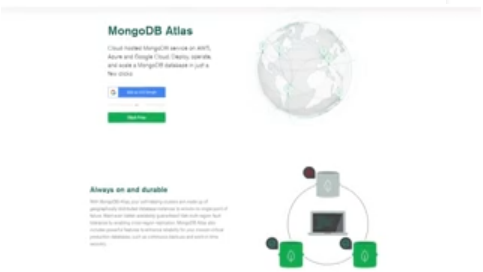


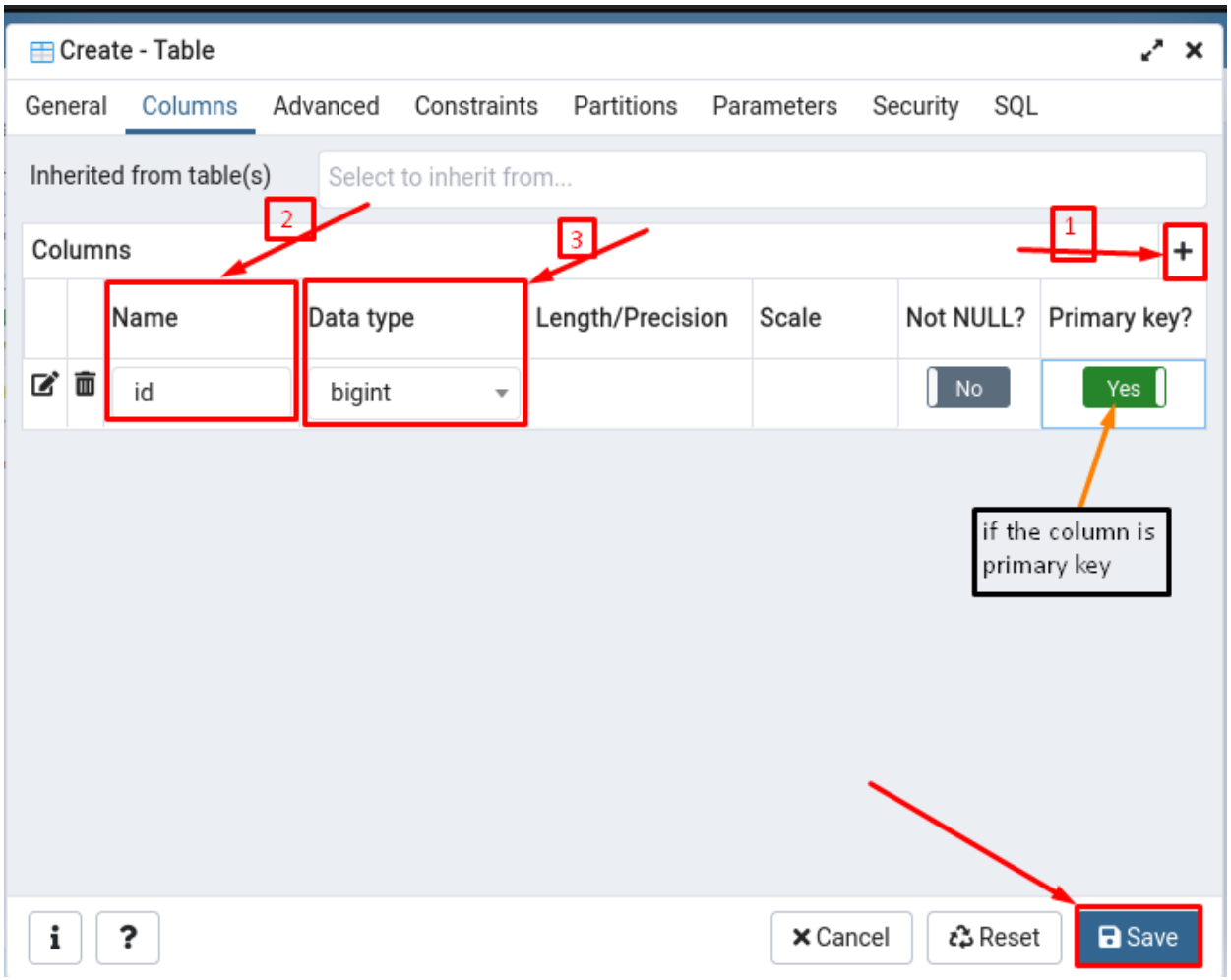
After that, you have to choose the name for that table (**distributions** in our case):



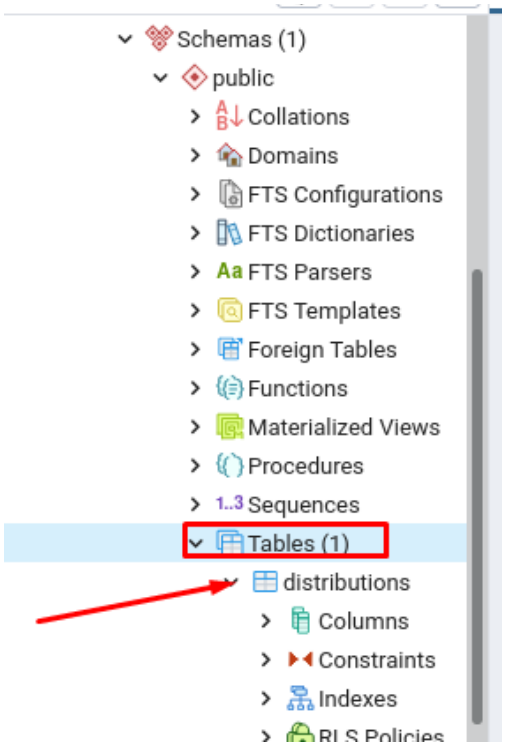
To add columns, you must navigate to the “**Columns**” tab and then perform the following steps.

- Click on the “**+**” option and then type the name of the column with the data type you want to assign
- (**Optional**) Turn the toggle button to ON if the column is a primary key
- After that, click on **Save** to create columns and tables successfully





Once the tables and columns are created, you can see them by navigating to “**Schemas**” and then under “**public**” schema, you can observe the “**distributions**”



Conclusion

In this technologically rich era, the graphical interface of the database server must be efficient enough to perform rigorous database operations. Therefore, several database servers are supported by some graphical tools to perform their duties. This post provides a brief guide on pgAdmin4, which is a database management tool for PostgreSQL databases. You have learned to install pgAdmin4 and set up an environment to use it on Manjaro Linux. Moreover, the basic working of this tool is also explained with images. This guide would be beneficial for the users that are using postgres database management systems in their organizations.

ABOUT THE AUTHOR

Adnan Shabbir

View all posts

RELATED LINUX HINT POSTS

- Linux Mint vs. Manjaro: Which One is Better?
- How to Install Jekyll SSG on Manjaro Linux 21
- How to Install Android Studio on Manjaro Linux 21
- Manjaro Vs. Ubuntu
- How to install and configure TFTP server on Manjaro
- How do I wipe my hard drive in Manjaro
- Manjaro Xfce vs KDE