Installing PostgreSQL on Windows

Topics to be covered:

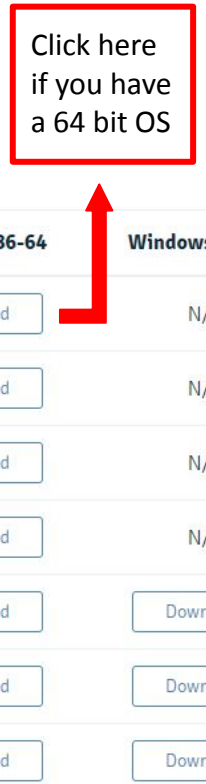
- Installing PostgreSQL on windows 10 devices
- Basic command line options

What is PostgreSQL?

PostgreSQL, also known as Postgres, is a free and open-source relational database management system emphasizing extensibility and SQL compliance.

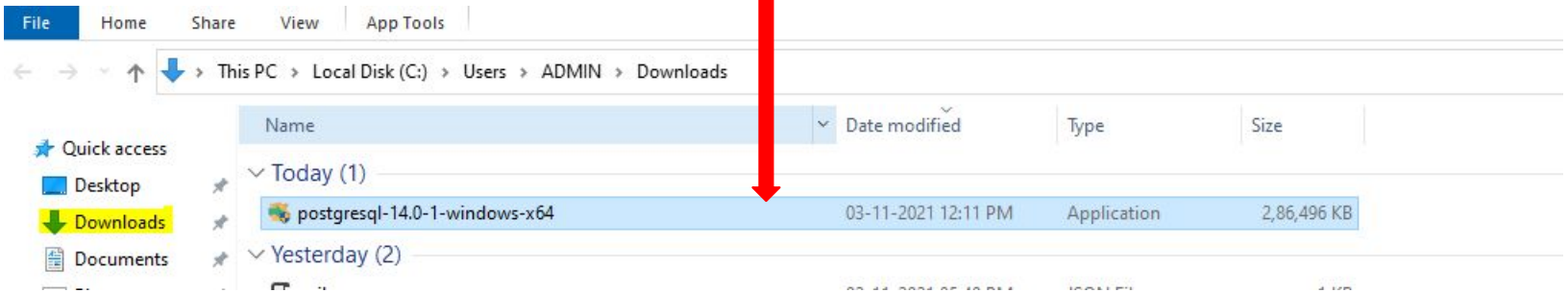
PostgreSQL Installation Link

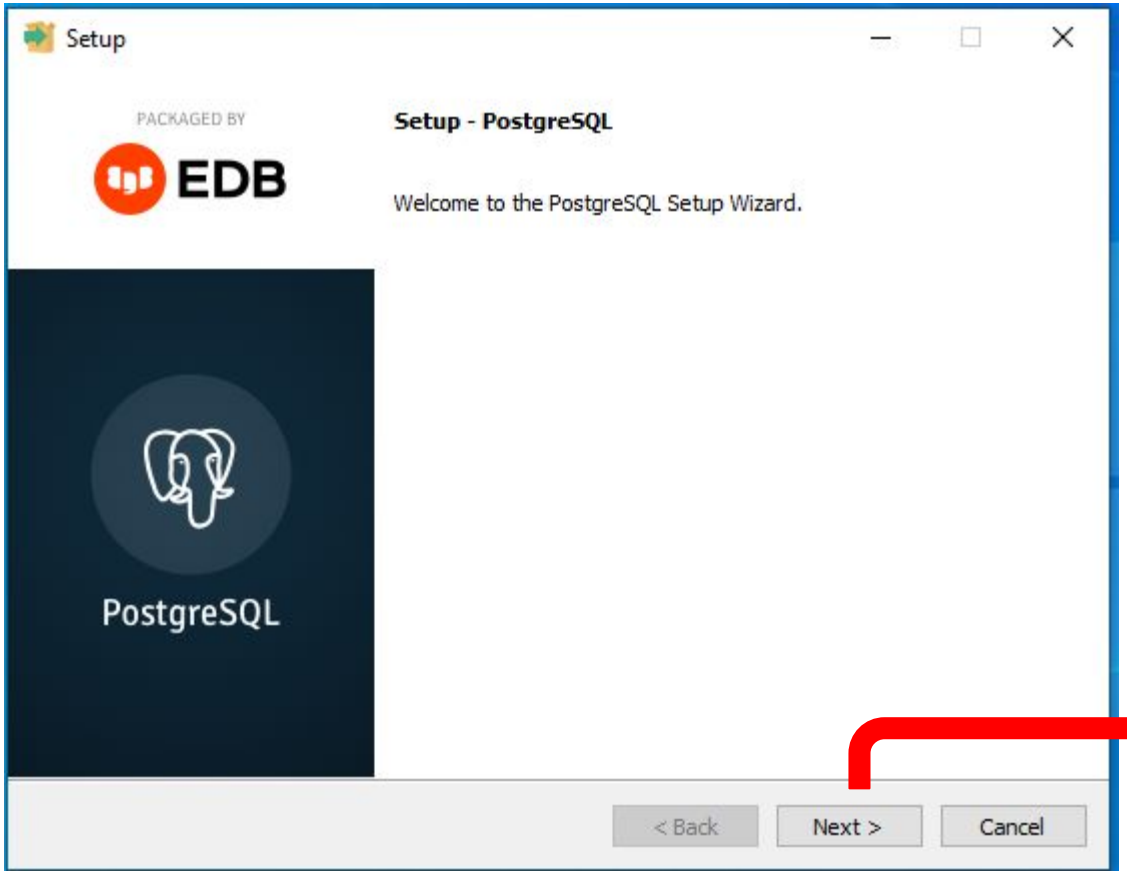
PostgreSQL Database Download



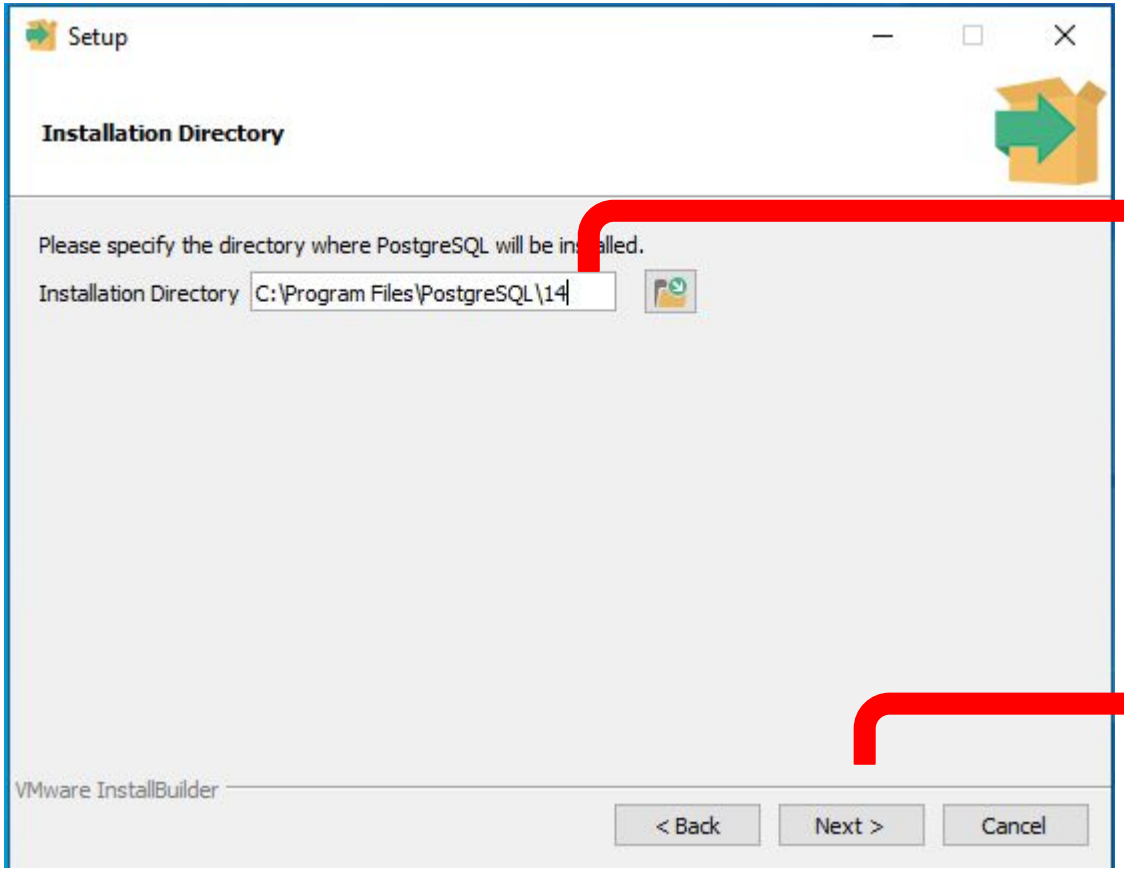
Version	Linux x86-64	Linux x86-32	Mac OS X	Windows x86-64	Windows x86-32
14	N/A	N/A	Download	Download	N/A
13.4	N/A	N/A	Download	Download	N/A
12.8	N/A	N/A	Download	Download	N/A
11.13	N/A	N/A	Download	Download	N/A
10.18	Download	Download	Download	Download	Download
9.6.23	Download	Download	Download	Download	Download
9.5.25 (Not Supported)	Download	Download	Download	Download	Download

In your Downloads folder, right click on the installer and click on Run as Administrator



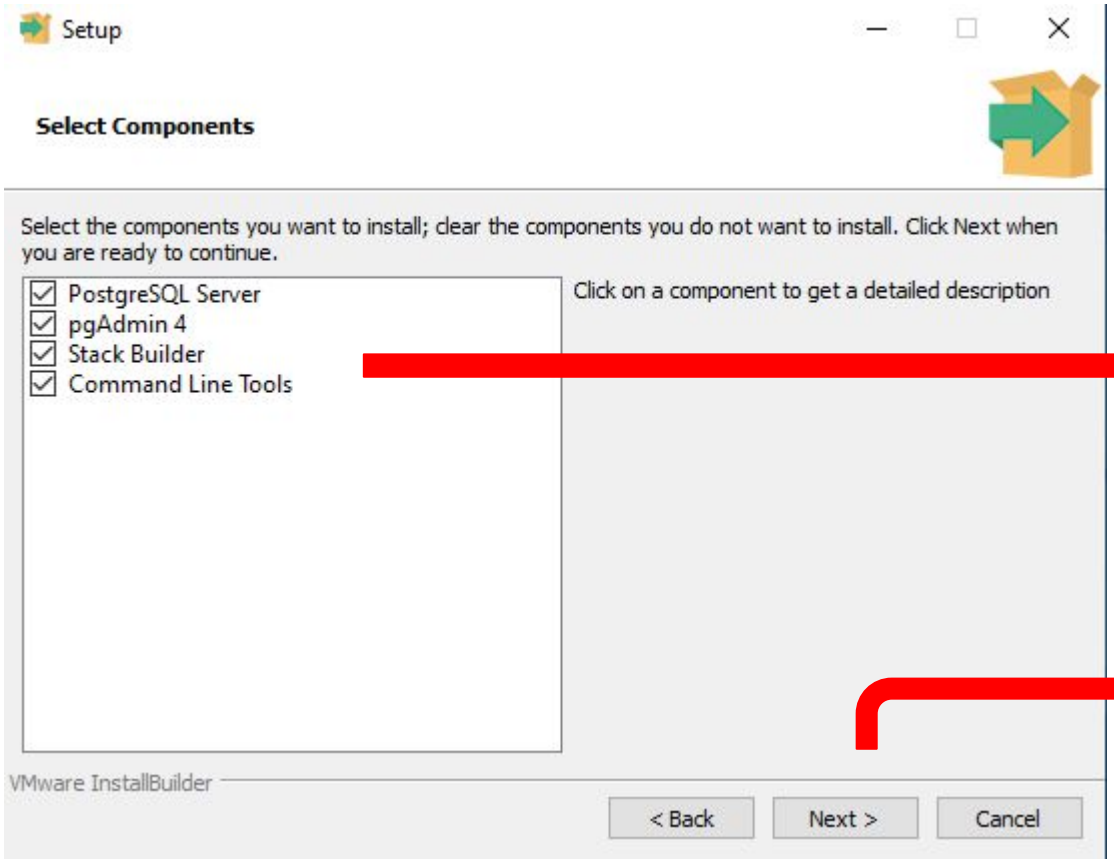


We will be using all the Default settings.
Click on Next



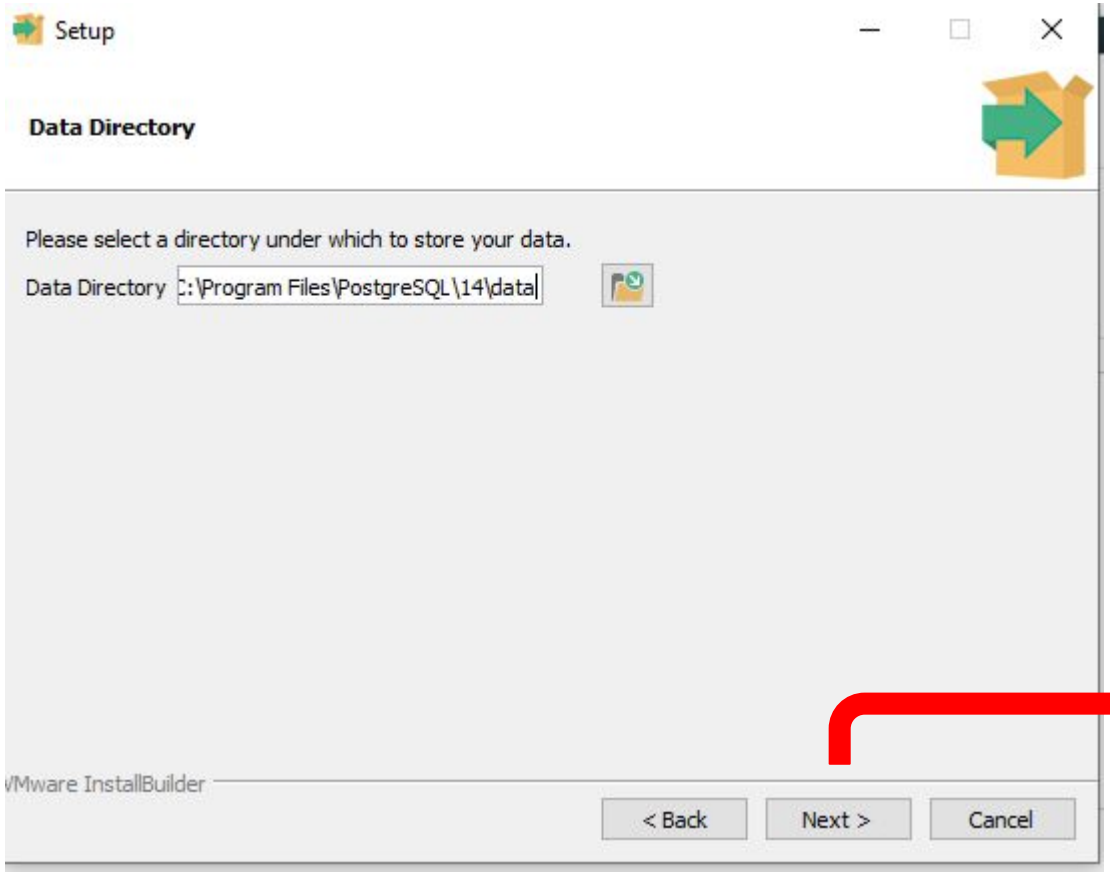
Your application will be installed in this path

We will be using all the Default settings.
Click on Next

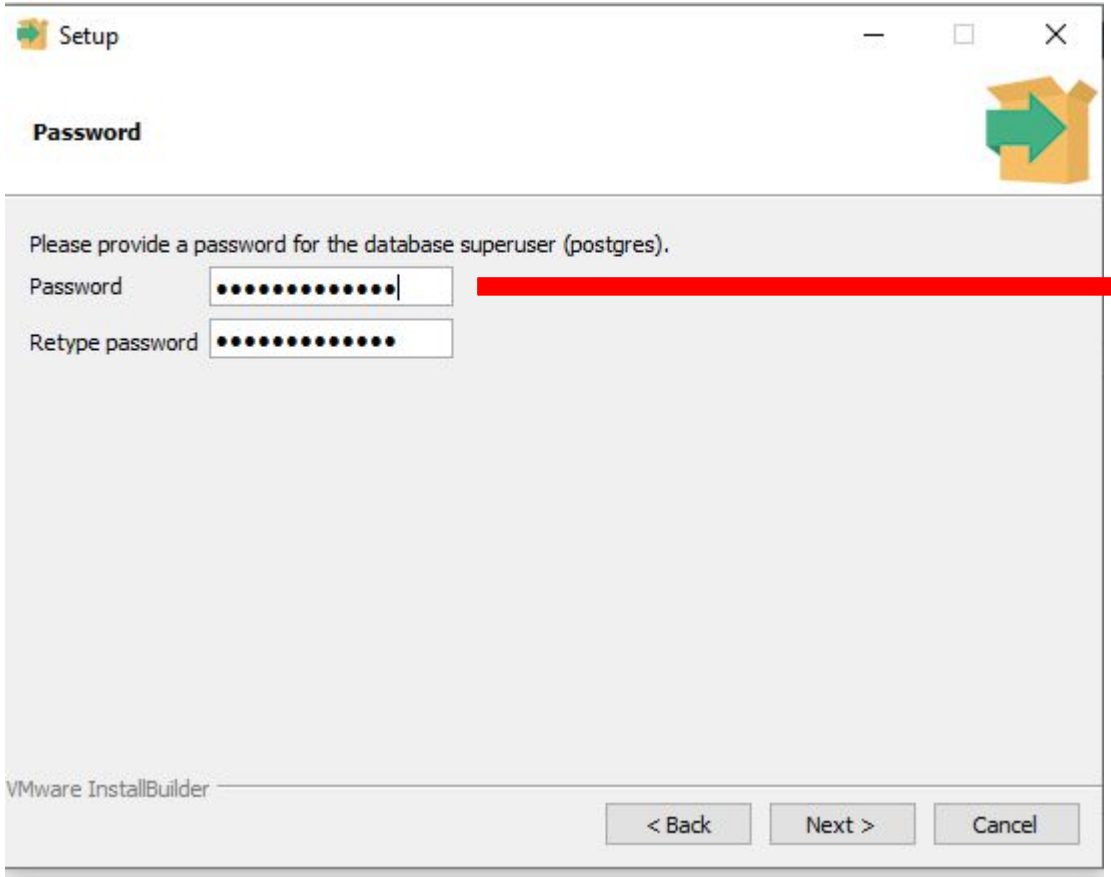


We will install all the components at this time

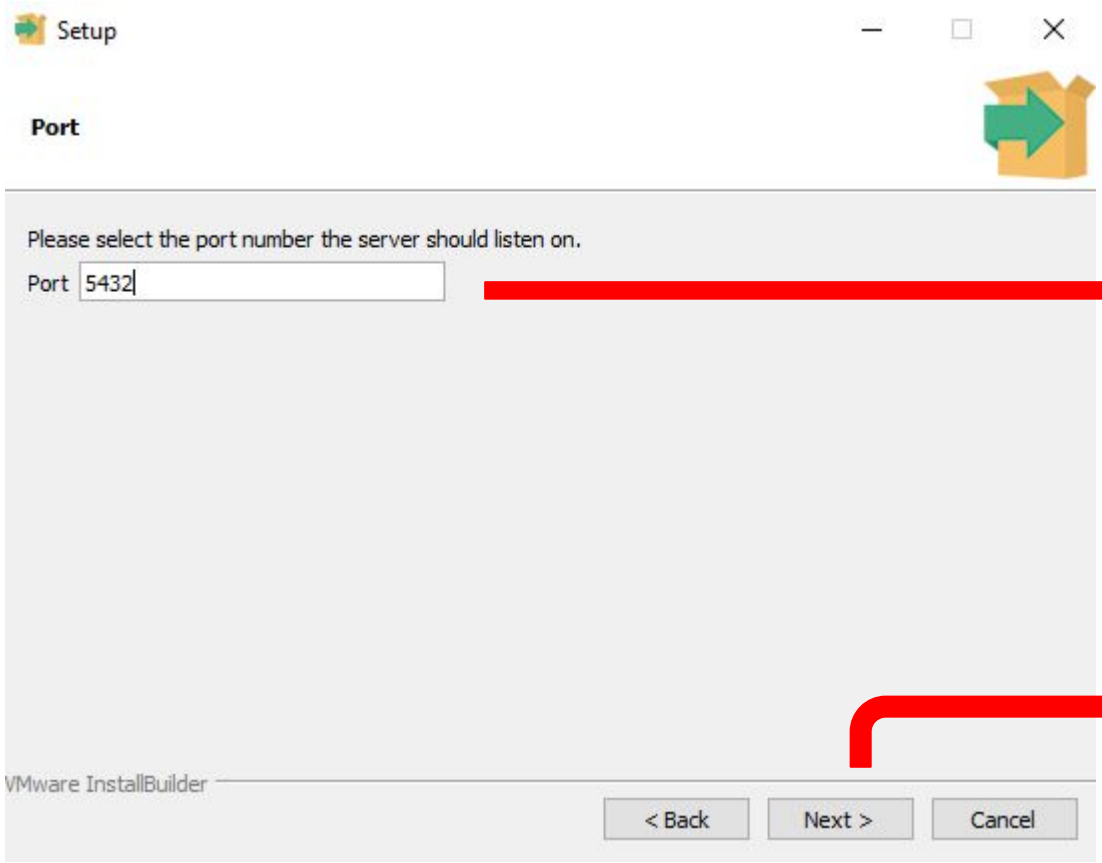
We will be using all the Default settings.
Click on Next



We will be using all the Default settings.
Click on Next

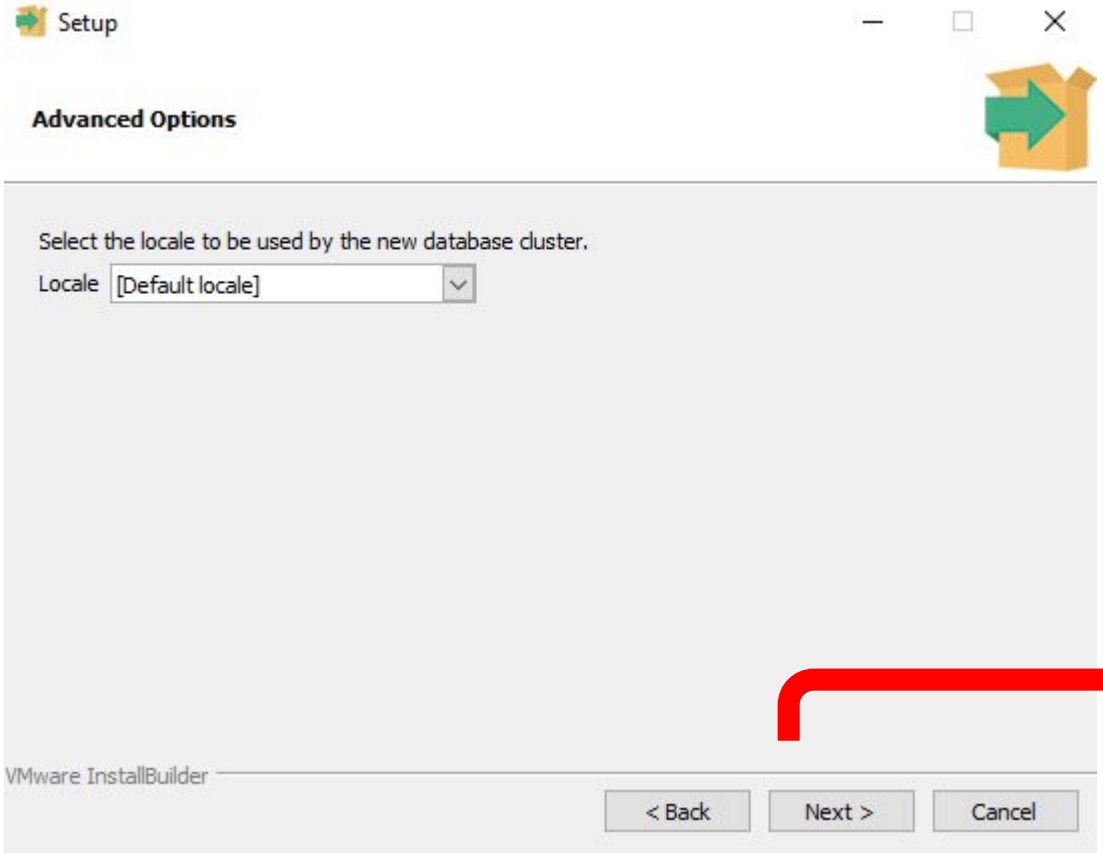


Enter a suitable password here
You will need this password to login to your database
Make a note of this password

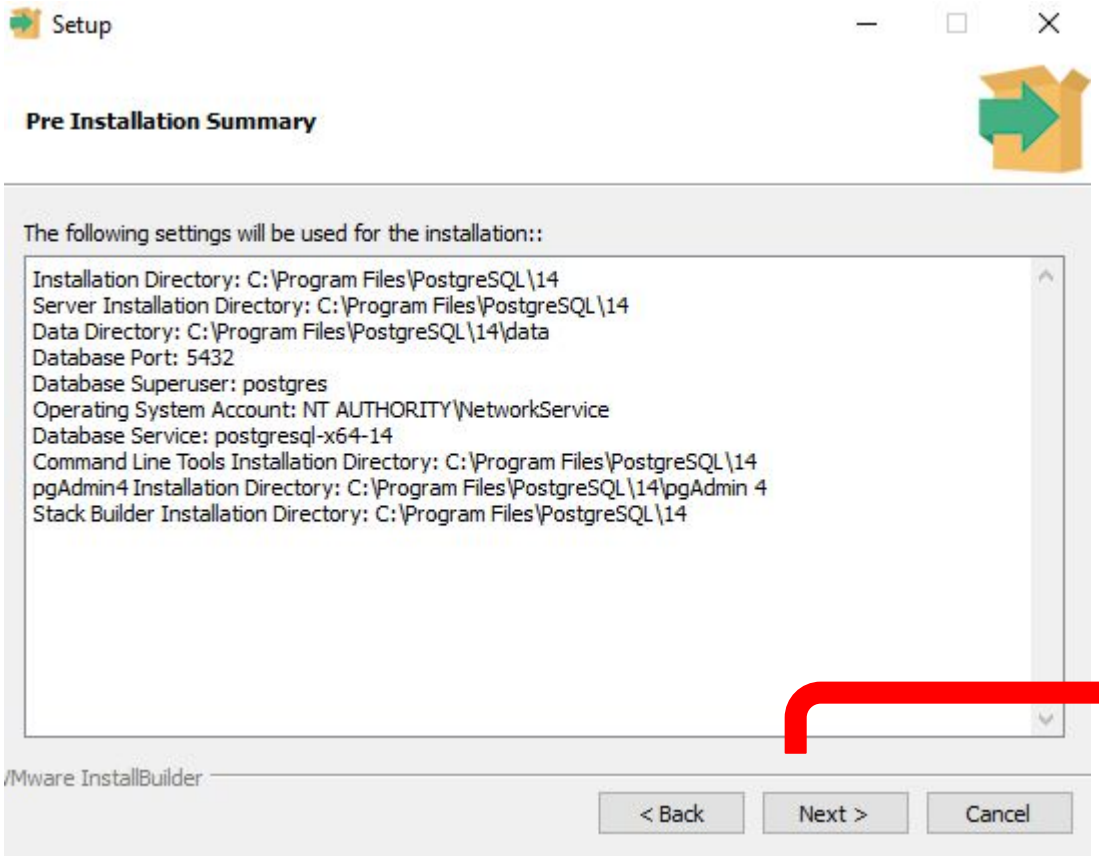


This is the default port.
Make sure that no other application is using this port
The database server will listen on this port

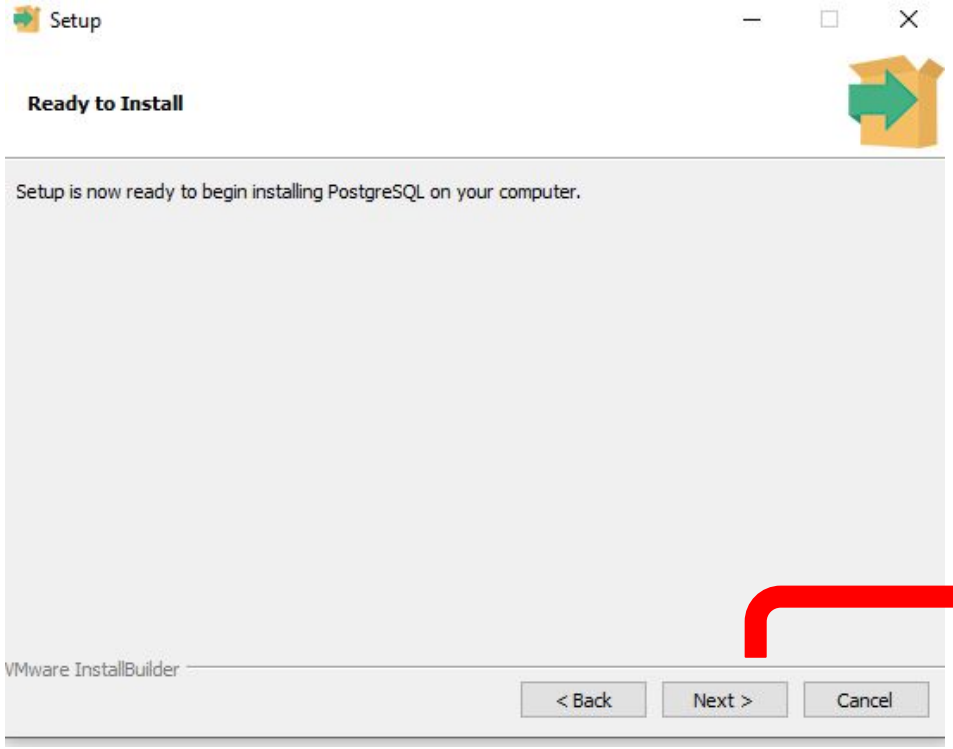
We will be using all the Default settings.
Click on Next



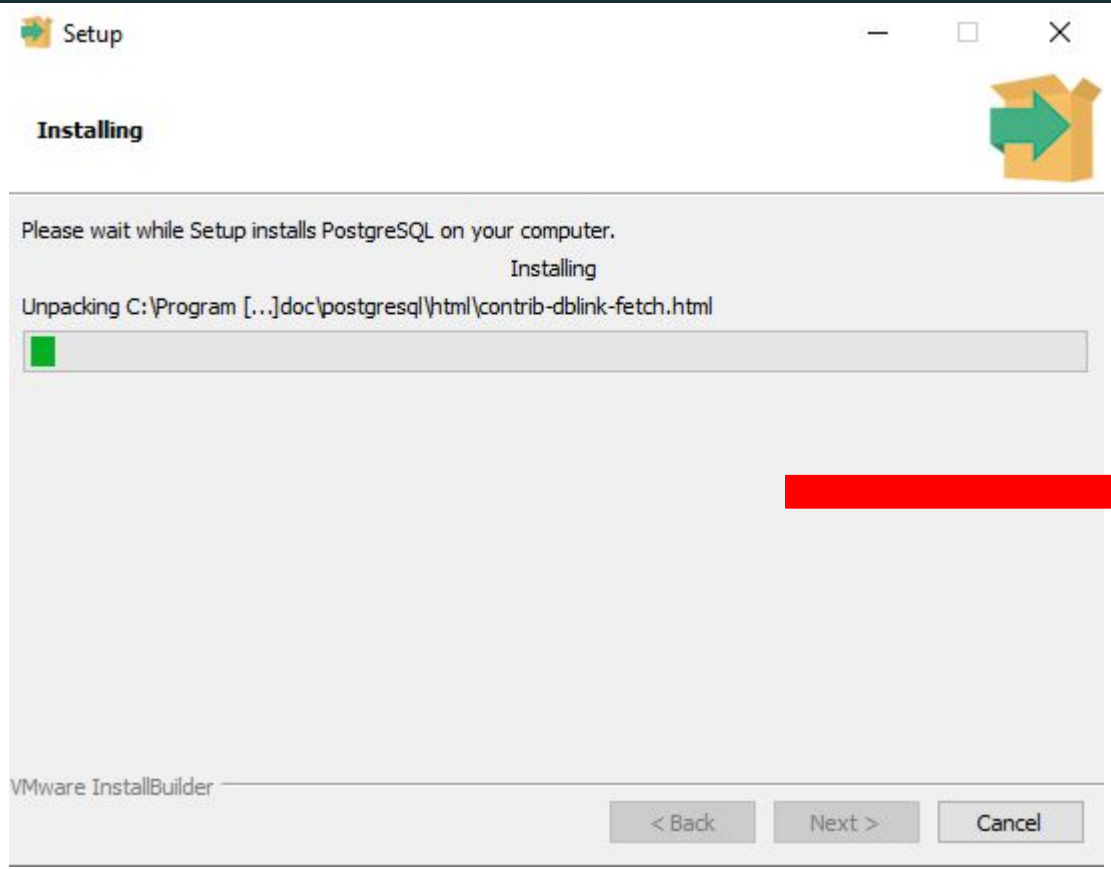
Click on Next



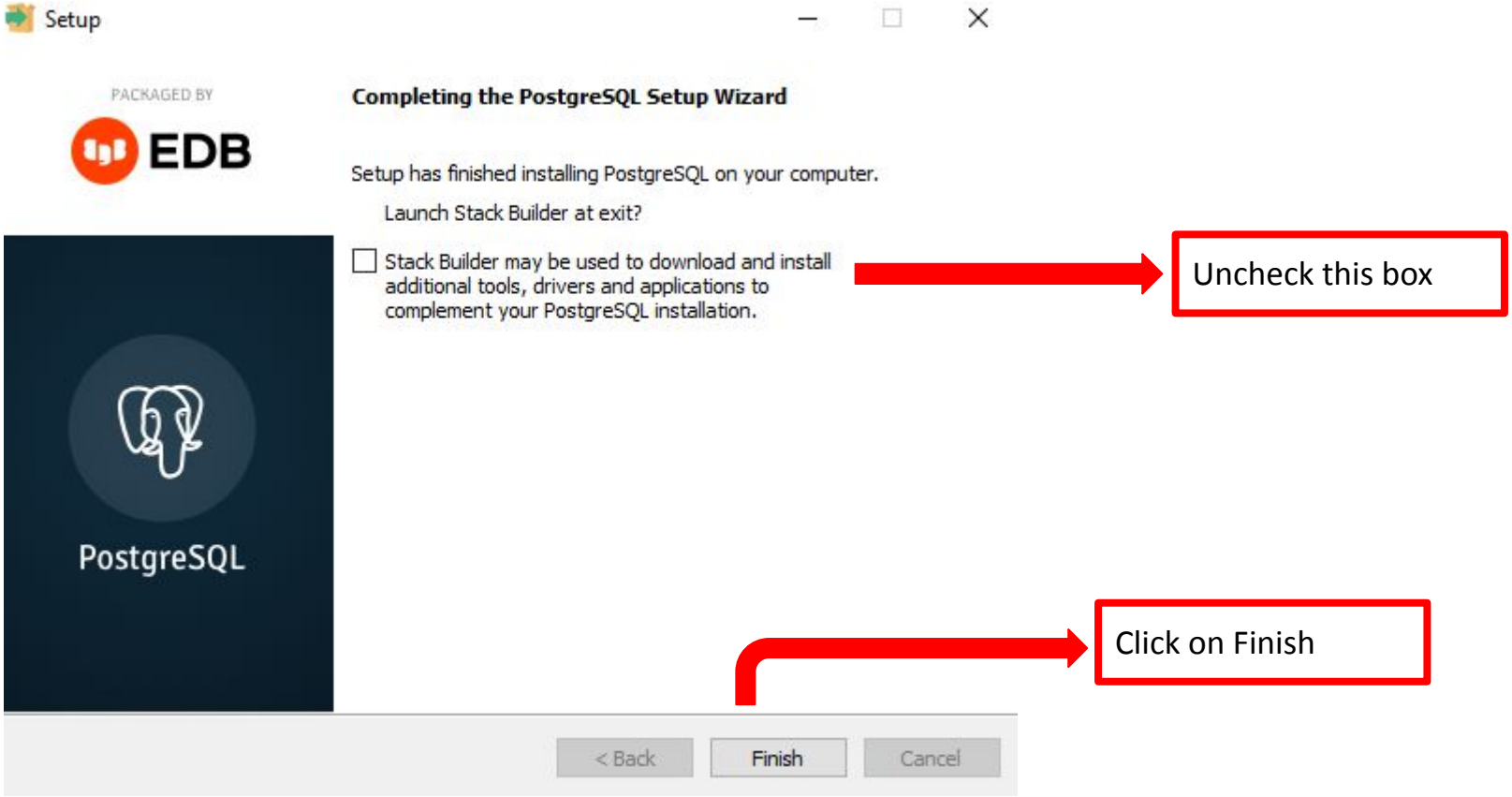
Click on Next



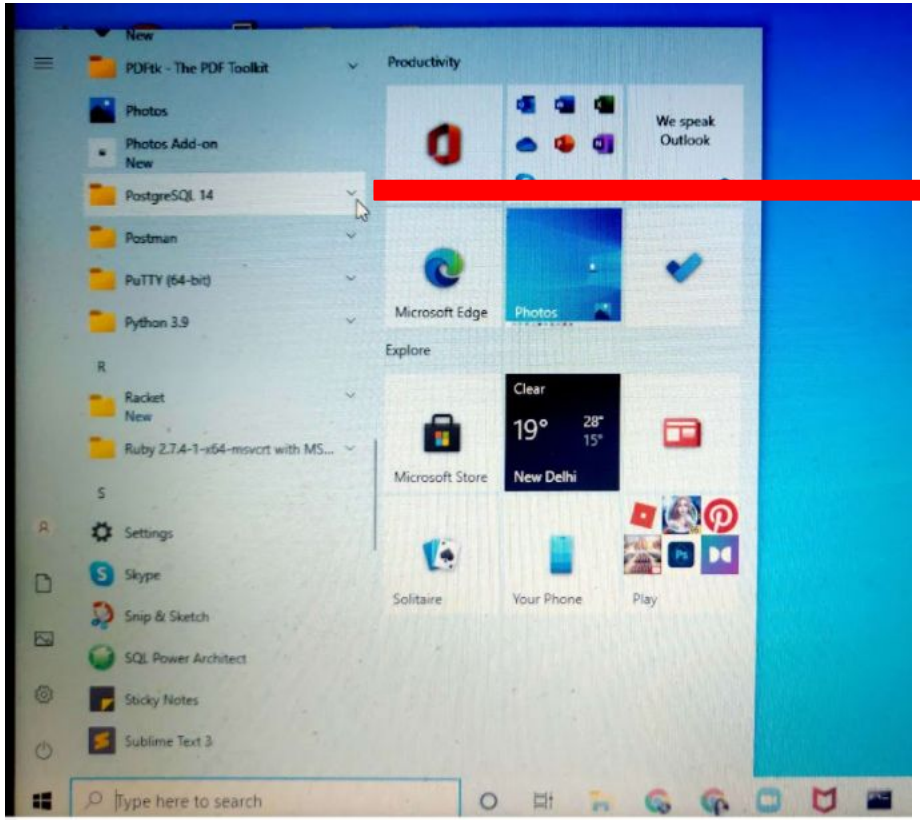
Click on Next



Installation could take around 5-10 minutes based on your machine performance

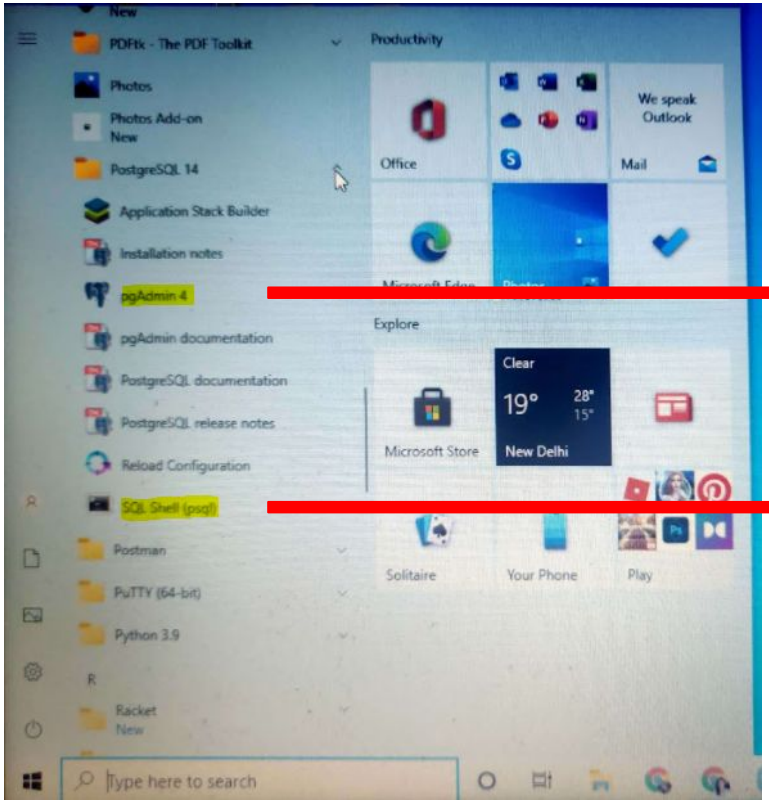

Setup
PACKAGED BY
EDB
Completing the PostgreSQL Setup Wizard
Setup has finished installing PostgreSQL on your computer.
Launch Stack Builder at exit?
Stack Builder may be used to download and install additional tools, drivers and applications to complement your PostgreSQL installation.
PostgreSQL
Uncheck this box
Click on Finish
< Back
Finish
Cancel

In your Start Menu (Click on the windows icon on your taskbar), search for PostgreSQL 14



For 64 bit OS: PostgreSQL 14
For 32 bit OS: PostgreSQL 10.8

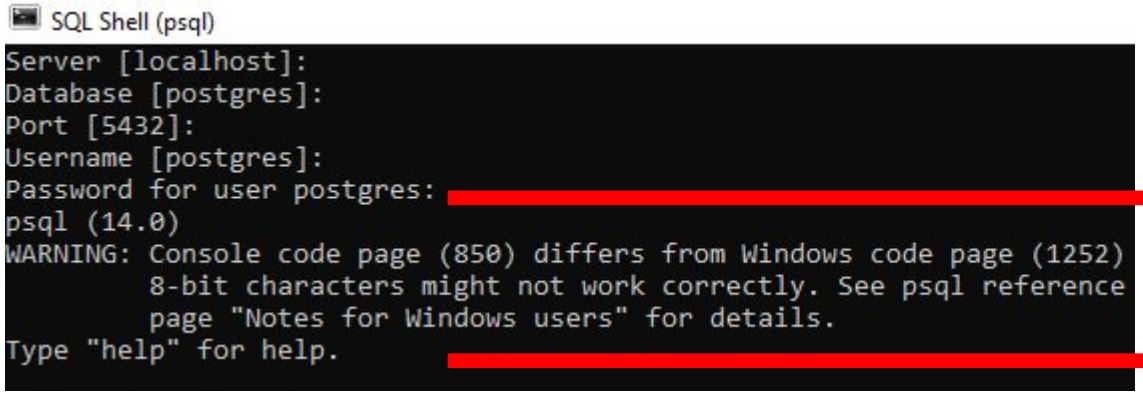
Click on the arrow beside PostgreSQL 14 to expand the options



pgAdmin 4:
pgAdmin is a web-based GUI tool used to interact with the Postgres database sessions, both locally and remote servers as well.

SQL Shell (psql):
psql is a terminal-based front-end to PostgreSQL. It enables you to type in queries interactively, issue them to PostgreSQL, and see the query results. Alternatively, input can be from a file or from command line arguments.

SQL Shell (psql)



Enter the password that was set during installation (See slide11)

This warning can be ignored!

\l command lists all the databases that exist on your machine

```
postgres=# \l
                                  List of databases
   Name    |  Owner   | Encoding |      Collate       |       Ctype        |   Access privileges
-----------+----------+----------+--------------------+--------------------+-----------------------
 postgres  | postgres | UTF8     | English_India.1252 | English_India.1252 |
 template0 | postgres | UTF8     | English_India.1252 | English_India.1252 | =c/postgres          +
           |          |          |                    |                    | postgres=CTc/postgres
 template1 | postgres | UTF8     | English_India.1252 | English_India.1252 | =c/postgres          +
           |          |          |                    |                    | postgres=CTc/postgres
(3 rows)
```

Default Database created at the time of installation

Command to create a new database named ‘mydb’
Note: Do not forget the semi colon at the end!

```
postgres=# create database mydb;
CREATE DATABASE
postgres=# \l
                                  List of databases
   Name    |  Owner   | Encoding |      Collate       |       Ctype        |   Access privileges
-----------+----------+----------+--------------------+--------------------+-----------------------
 mydb      | postgres | UTF8     | English_India.1252 | English_India.1252 |
 postgres  | postgres | UTF8     | English_India.1252 | English_India.1252 |
 template0 | postgres | UTF8     | English_India.1252 | English_India.1252 | =c/postgres          +
           |          |          |                    |                    | postgres=CTc/postgres
 template1 | postgres | UTF8     | English_India.1252 | English_India.1252 | =c/postgres          +
           |          |          |                    |                    | postgres=CTc/postgres
(4 rows)
```

\l command lists the new database ‘mydb’ that was created using the above command

```
postgres=# create user testuser with password 'pass';
CREATE ROLE
postgres=# \c mydb
You are now connected to database "mydb" as user "postgres".
mydb=# \c postgres
You are now connected to database "postgres" as user "postgres".
postgres=# \c mydb
You are now connected to database "mydb" as user "postgres".
mydb=# create schema demo;
CREATE SCHEMA
mydb=# create table demo.test(firstname CHAR(10), lastname CHAR(20));
CREATE TABLE
mydb=# \d demo.test;
                Table "demo.test"
  Column   |     Type      | Collation | Nullable | Default
-----------+---------------+-----------+----------+---------
 firstname | character(10) |           |          |
 lastname  | character(20) |           |          |


mydb=# select * from demo.test;
 firstname | lastname
-----------+----------
(0 rows)


mydb=# insert into demo.test values('John','Doe');
INSERT 0 1
mydb=# select * from demo.test;
 firstname  |       lastname
------------+----------------------
 John       | Doe
(1 row)
```

Command to create a new user ‘testuser’ with password ‘pass’

Command to connect to a DB: **\c <DB name>**

Schema to manage your tables and views. Default schema **public**

Command to create table:
create <schema_name.table_name> (column1 type,column2 type, … constraints)

Command to describe table: **\d <schema_name.table_name>**

Command to display specific columns in a table:
select column_name from <schema_name.table_name>

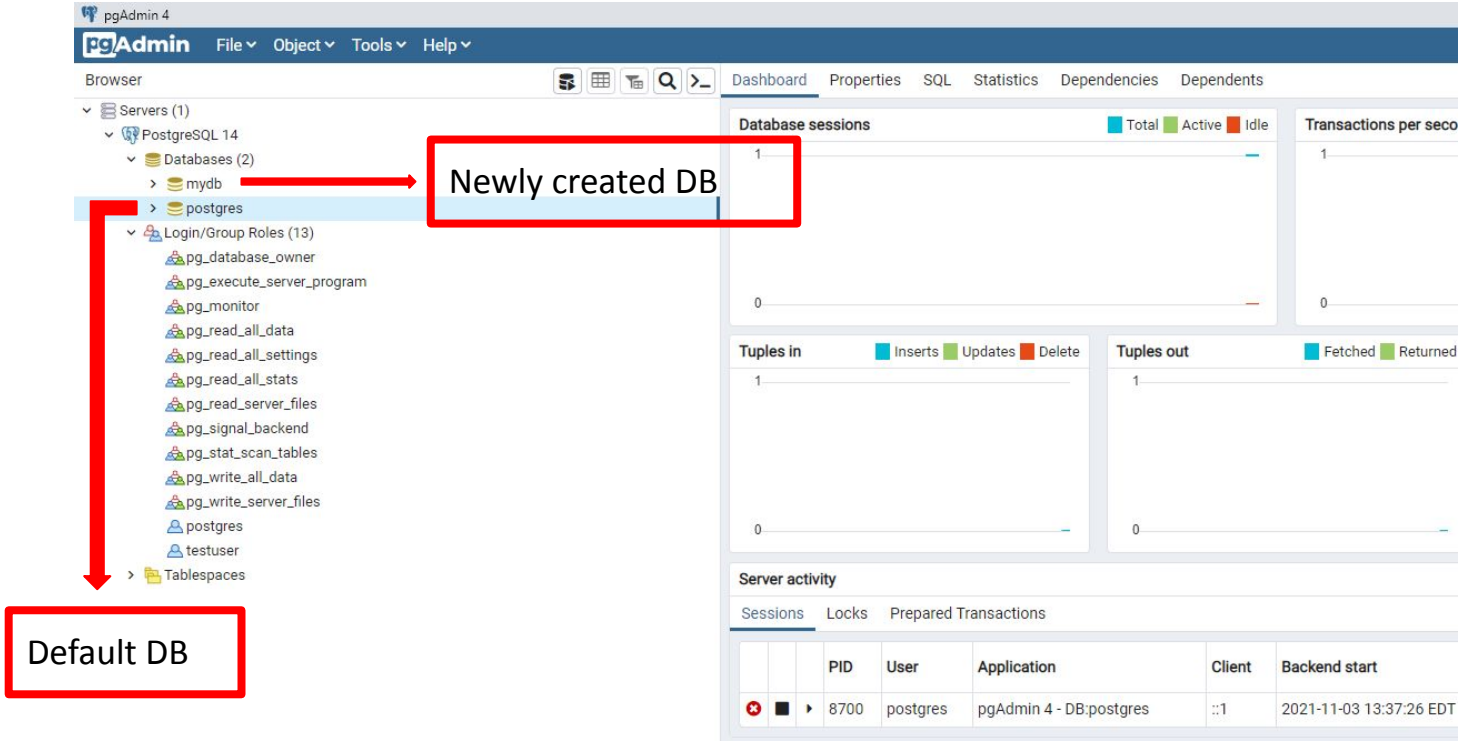
Command to display all columns in a table:
select * from <schema_name.table_name>

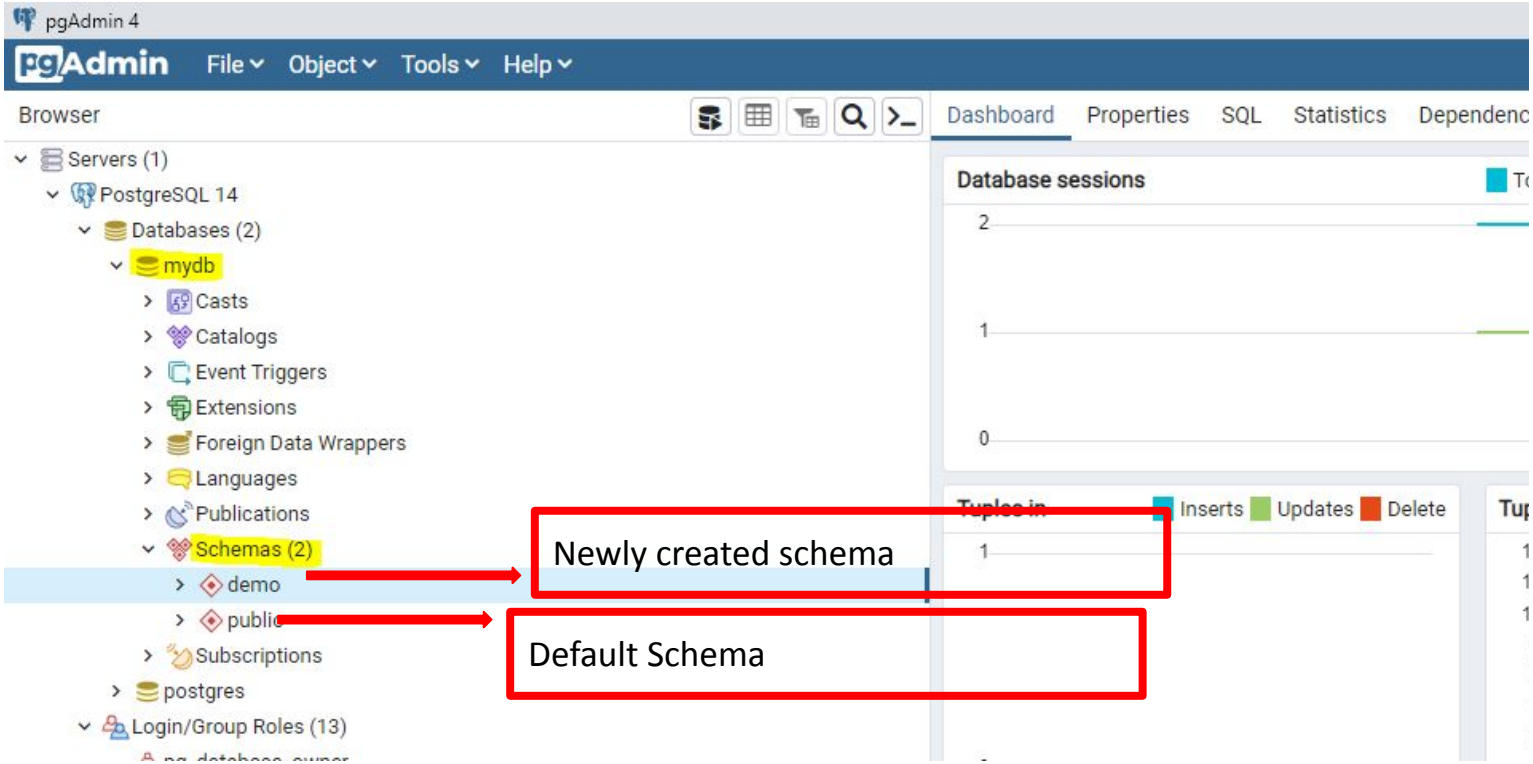
Command to insert data in a table:
Insert into <schema_name.table_name> values(value1, value2..)

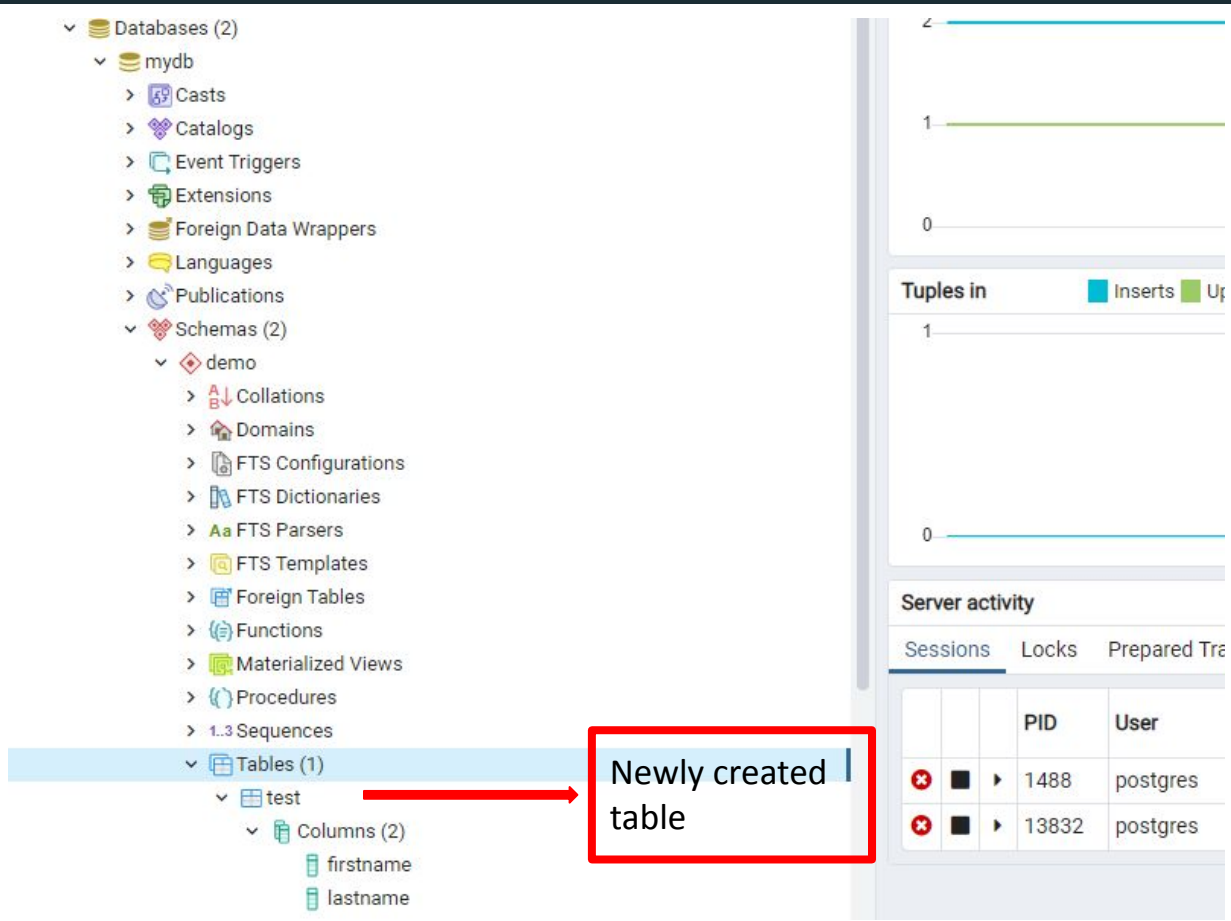
Adding constraints like Primary Key after the table has been created

```
mydb=# alter table demo.test
mydb-# ADD constraint test_pk PRIMARY KEY (firstname);
ALTER TABLE
mydb=# \d demo.test
                    Table "demo.test"
  Column   |     Type      | Collation | Nullable | Default
-----------+---------------+-----------+----------+---------
 firstname | character(10) |           | not null |
 lastname  | character(20) |           |          |
Indexes:
    "test_pk" PRIMARY KEY, btree (firstname)
```

PgAdmin 4 screenshots




pgAdmin 4
pgAdmin
File
Object
Tools
Help
Browser
Dashboard
Properties
SQL
Statistics
Servers (1)
PostgreSQL 14
Databases (2)
mydb
Casts
Catalogs
Event Triggers
Extensions
Foreign Data Wrappers
Languages
Publications
Schemas (2)
demo
public
Subscriptions
postgres
Login/Group Roles (13)
Database sessions
Inserts
Updates
Delete
Newly created schema
Default Schema


Databases (2)
mydb
Casts
Catalogs
Event Triggers
Extensions
Foreign Data Wrappers
Languages
Publications
Schemas (2)
demo
Collations
Domains
FTS Configurations
FTS Dictionaries
FTS Parsers
FTS Templates
Foreign Tables
Functions
Materialized Views
Procedures
Sequences
Tables (1)
test
Columns (2)
firstname
lastname
Newly created table
2
1
0
Tuples in
Inserts
1
0
Server activity
Sessions
Locks
Prepared Tra
PID
User
1488
postgres
13832
postgres

In case of any questions/doubts, you can:

- Post your questions on edstem
- Email at tfb9946@nyu.edu