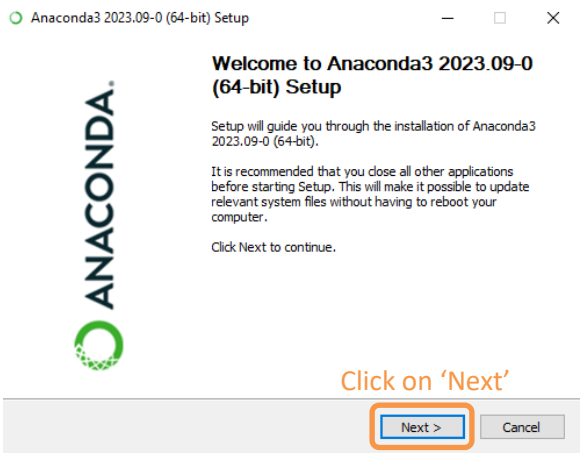
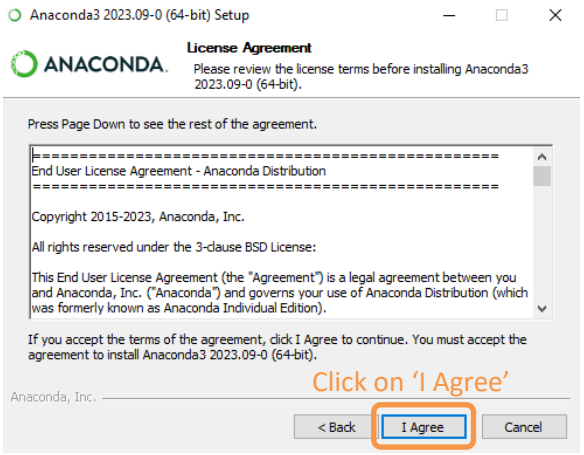
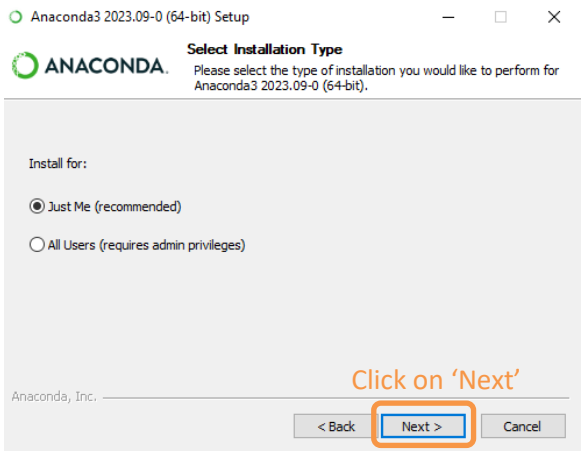
Step 1: Install Anaconda on your computer

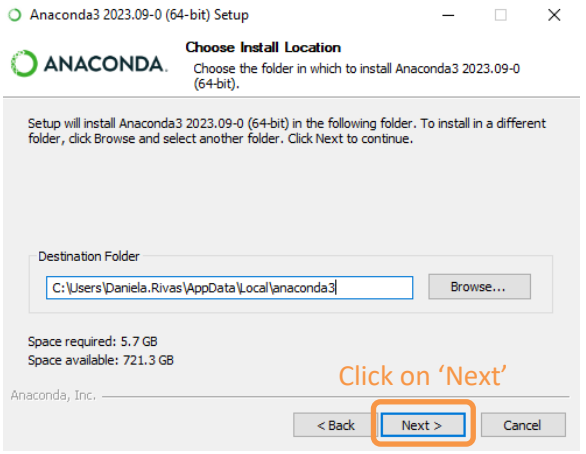
1. Go to https://www.anaconda.com/products/distribution
2. Scroll down to the bottom of the page where you will find the different Anaconda Installers:

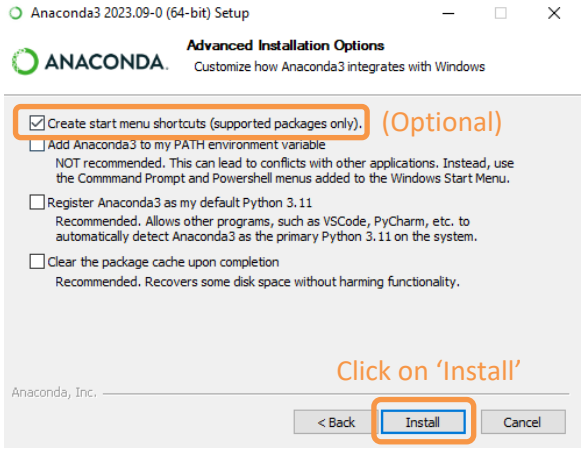
3. Click on the correct installer for your operating system, and the installer will start to download.
4. Go to the folder where your downloads are stored and double click on the installer.
5. Follow the installation steps as follow:

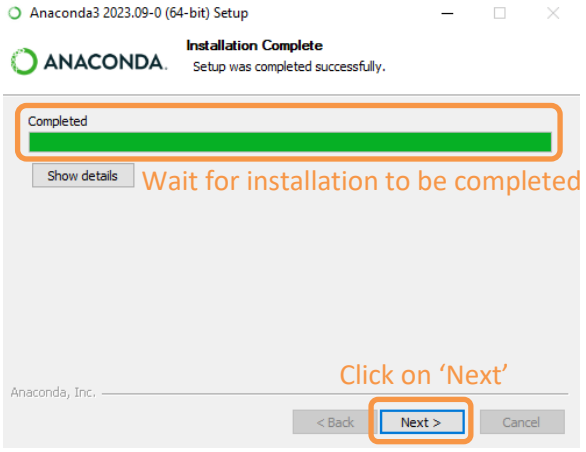


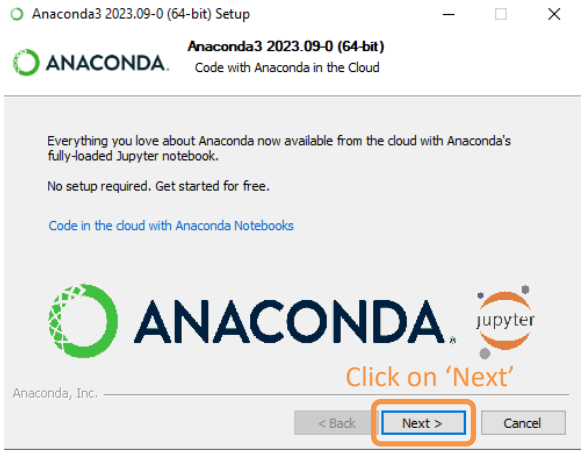


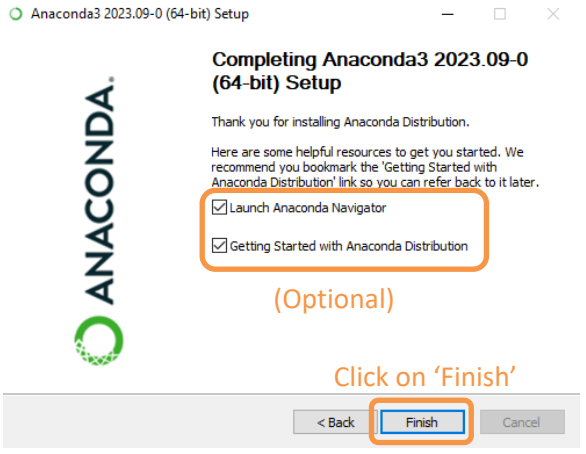

Anaconda3 2023.09-0 (64-bit) Setup
ANACONDA.
Select Installation Type
Please select the type of installation you would like to perform for Anaconda3 2023.09-0 (64-bit).
Install for:
Just Me (recommended)
All Users (requires admin privileges)
Click on 'Next'
Anaconda, Inc.
< Back
Next >
Cancel


Anaconda3 2023.09-0 (64-bit) Setup
ANACONDA.
Choose Install Location
Choose the folder in which to install Anaconda3 2023.09-0 (64-bit).
Setup will install Anaconda3 2023.09-0 (64-bit) in the following folder. To install in a different folder, click Browse and select another folder. Click Next to continue.
Destination Folder
C:\Users\Daniela.Rivas\AppData\Local\anaconda3
Browse...
Space required: 5.7 GB
Space available: 721.3 GB
Click on 'Next'
Anaconda, Inc.
< Back
Next >
Cancel


Anaconda3 2023.09-0 (64-bit) Setup
ANACONDA.
Advanced Installation Options
Customize how Anaconda3 integrates with Windows
Create start menu shortcuts (supported packages only).
(Optional)
Add Anaconda3 to my PATH environment variable
NOT recommended. This can lead to conflicts with other applications. Instead, use the Commmand Prompt and Powershell menus added to the Windows Start Menu.
Register Anaconda3 as my default Python 3.11
Recommended. Allows other programs, such as VSCode, PyCharm, etc. to automatically detect Anaconda3 as the primary Python 3.11 on the system.
Clear the package cache upon completion
Recommended. Recovers some disk space without harming functionality.
Click on 'Install'
Anaconda, Inc.
< Back
Install
Cancel


Anaconda3 2023.09-0 (64-bit) Setup
ANACONDA.
Installation Complete
Setup was completed successfully.
Completed
Show details
Wait for installation to be completed
Click on 'Next'
Anaconda, Inc.
< Back
Next >
Cancel

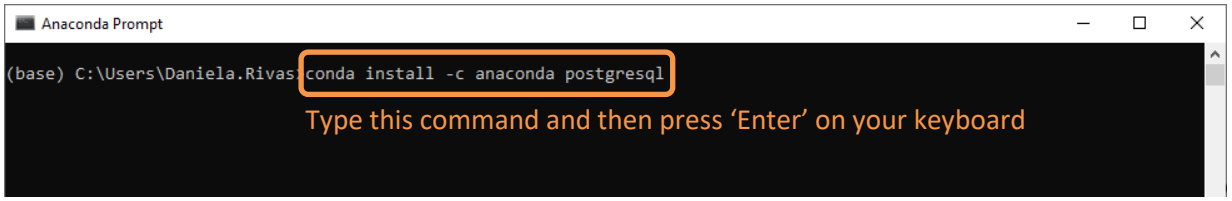

Anaconda3 2023.09-0 (64-bit) Setup
ANACONDA.
Anaconda3 2023.09-0 (64-bit)
Code with Anaconda in the Cloud
Everything you love about Anaconda now available from the cloud with Anaconda's fully-loaded Jupyter notebook.
No setup required. Get started for free.
Code in the cloud with Anaconda Notebooks
ANACONDA.
jupyter
Click on 'Next'
Anaconda, Inc.
< Back
Next >
Cancel


Anaconda3 2023.09-0 (64-bit) Setup
ANACONDA.
Completing Anaconda3 2023.09-0 (64-bit) Setup
Thank you for installing Anaconda Distribution.
Here are some helpful resources to get you started. We recommend you bookmark the 'Getting Started with Anaconda Distribution' link so you can refer back to it later.
Launch Anaconda Navigator
Getting Started with Anaconda Distribution
(Optional)
Click on 'Finish'
< Back
Finish
Cancel

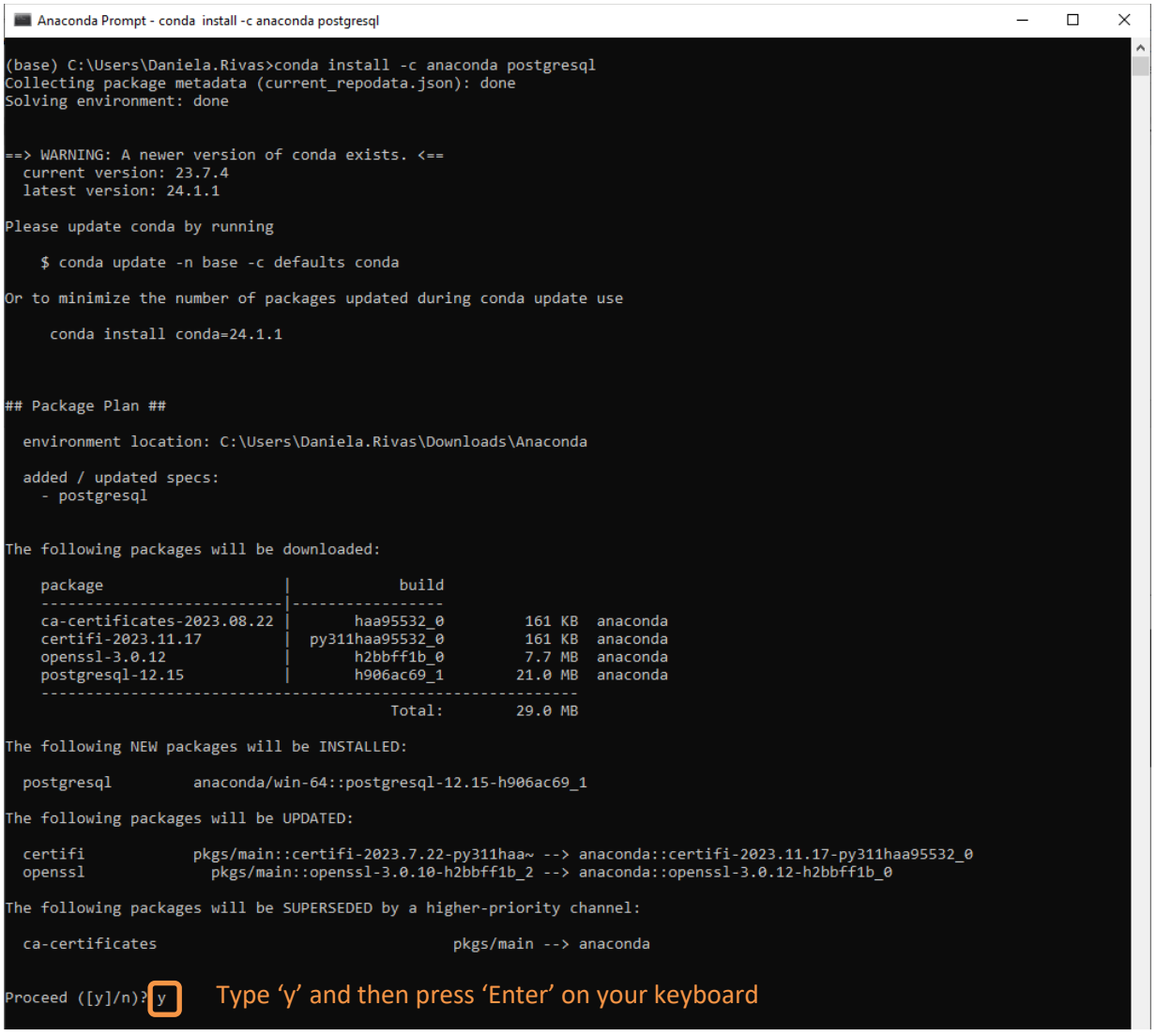
Step 2: Install and start PostgreSQL locally

You are going to use PostgreSQL later in this course. PostgreSQL is a database engine implementing SQL standards. To install the PostgreSQL package we will use Anaconda:

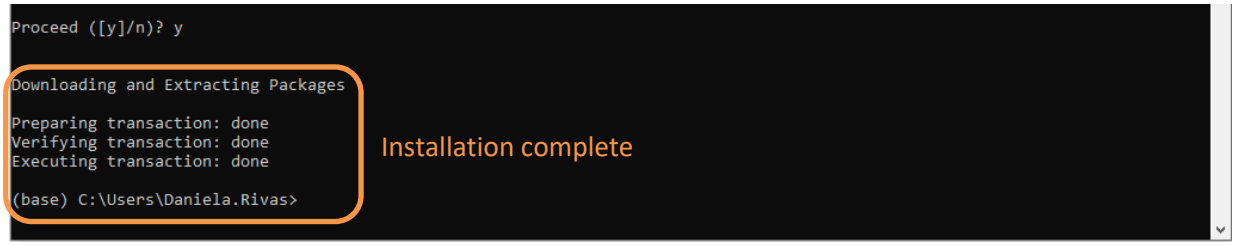
1. If your operating system is macOS, open the Terminal. If your operating system is Windows, open the Anaconda Prompt as Administrator, and type the following command: "`conda install -c anaconda postgresql`", and then press Enter on your keyboard.



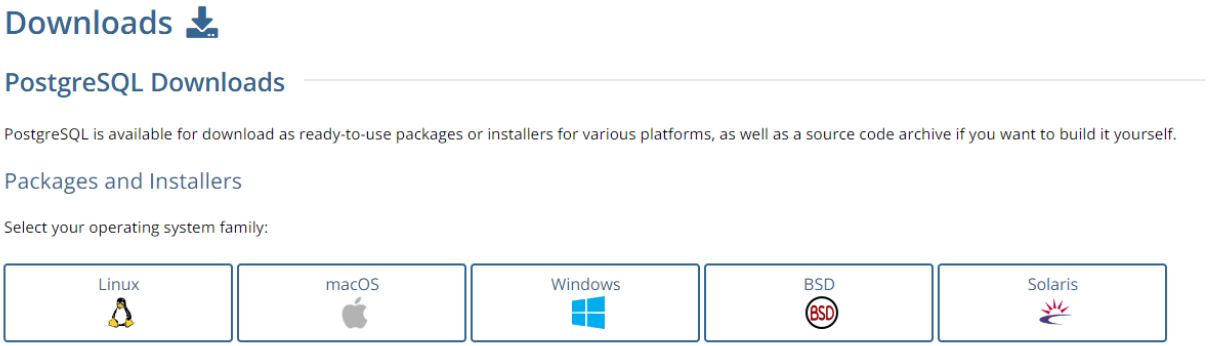
2. Wait for the installation to proceed and then type "y" when asked to proceed:



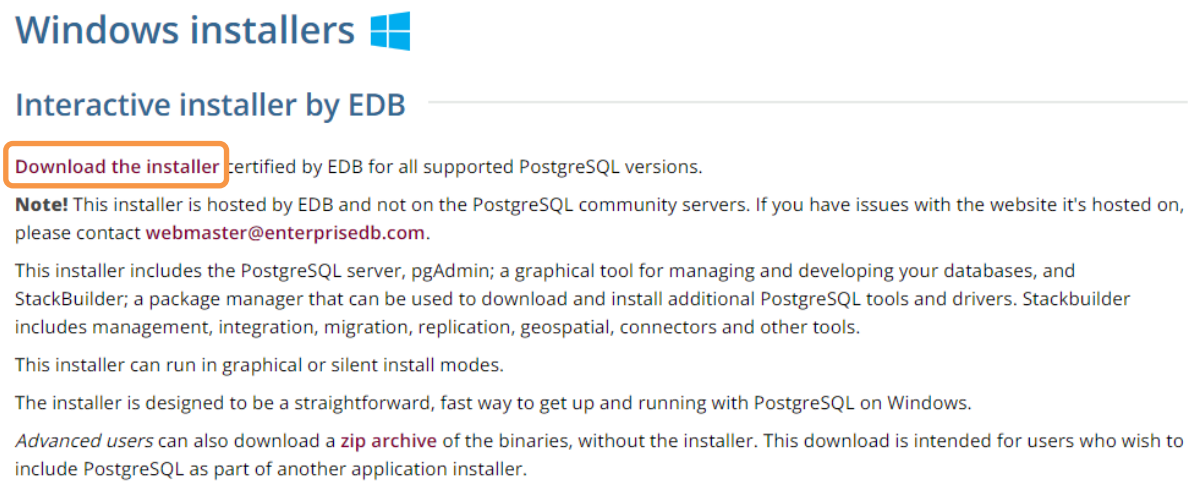
3. Wait for the installation to complete and close the terminal:



4. Go to https://www.postgresql.org/download/
5. Click on the correct installer for your operating system:



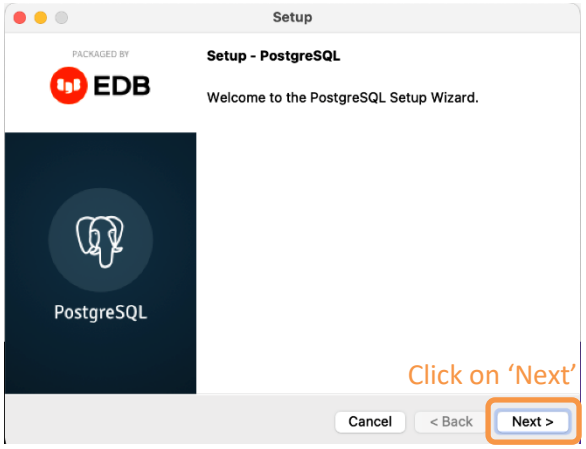
6. Click on “Download the installer”:



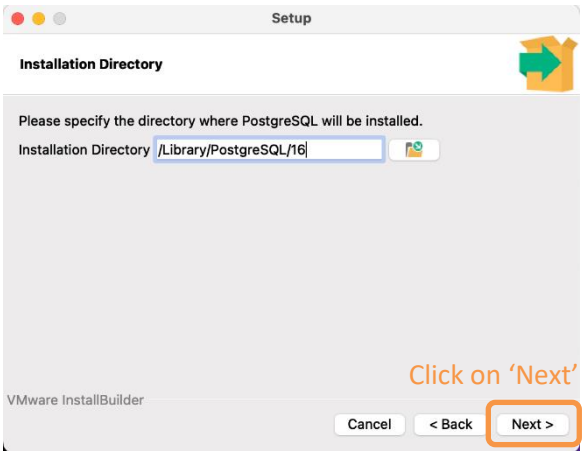
7. Click on the latest PostgreSQL version compatible with your operating system, and the installer will start to download.

Click on the correct installer for your operating system

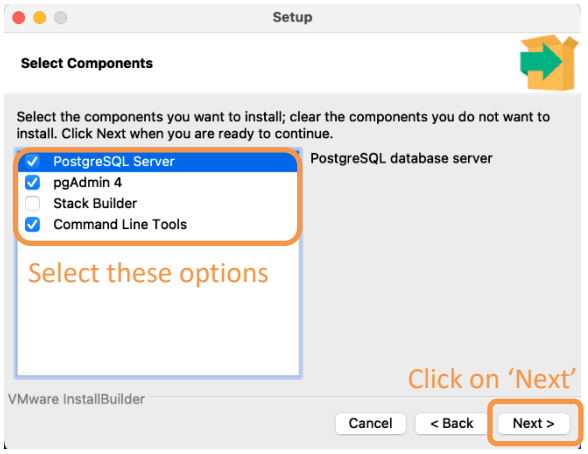
8. Go to the folder where your downloads are stored and double click on the installer.
9. Follow the installation steps as follow:

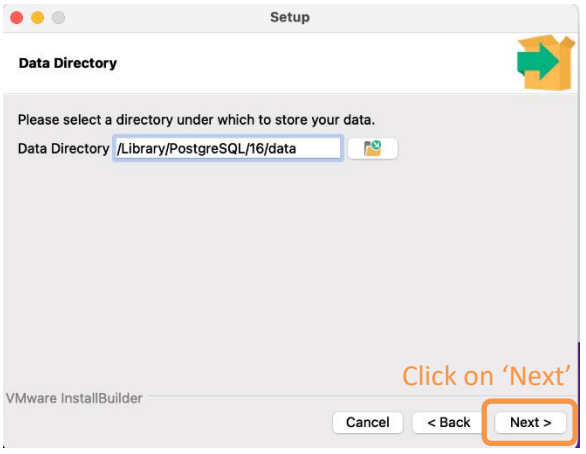


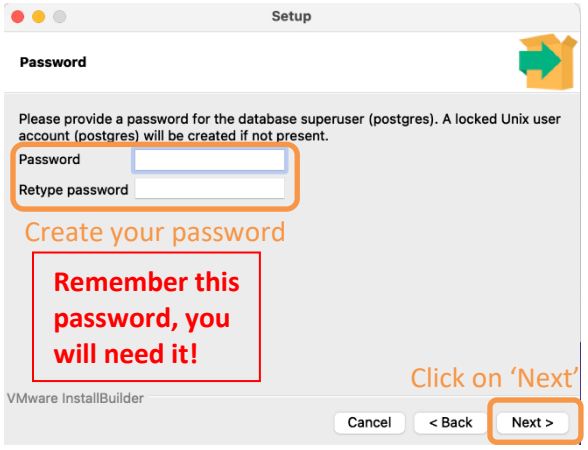
Click on 'Next'

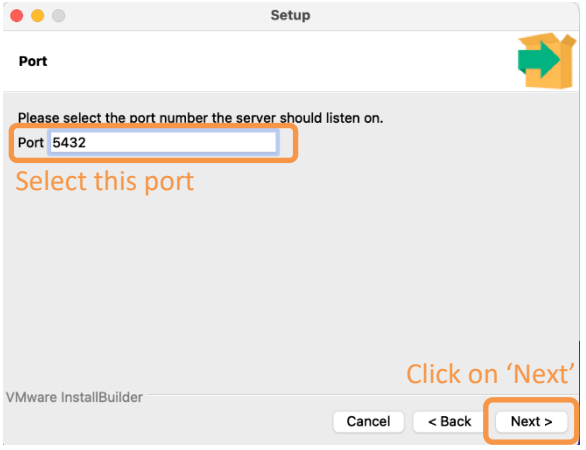


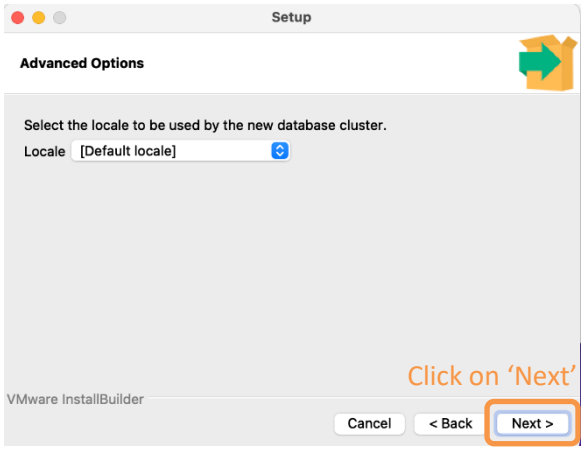
Click on 'Next'

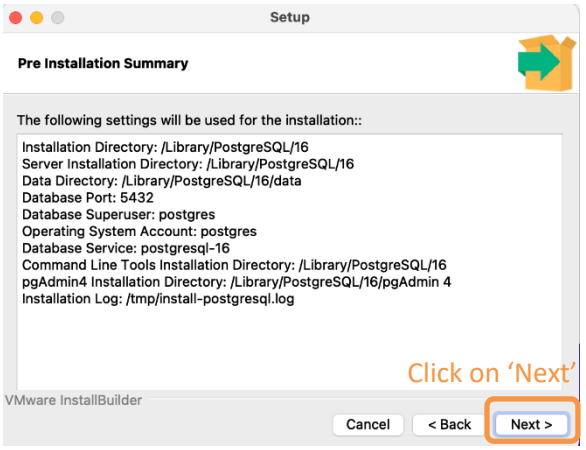

Setup
Select Components
Select the components you want to install; clear the components you do not want to install. Click Next when you are ready to continue.
PostgreSQL Server
pgAdmin 4
Stack Builder
Command Line Tools
PostgreSQL database server
Select these options
Click on 'Next'
VMware InstallBuilder
Cancel
< Back
Next >

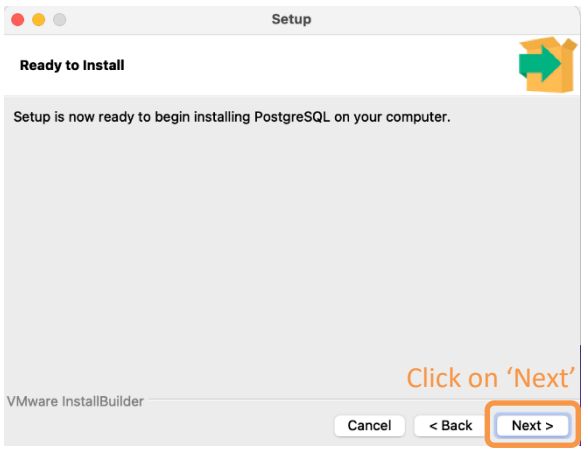

Setup
Data Directory
Please select a directory under which to store your data.
Data Directory /Library/PostgreSQL/16/data
Click on 'Next'
VMware InstallBuilder
Cancel
< Back
Next >

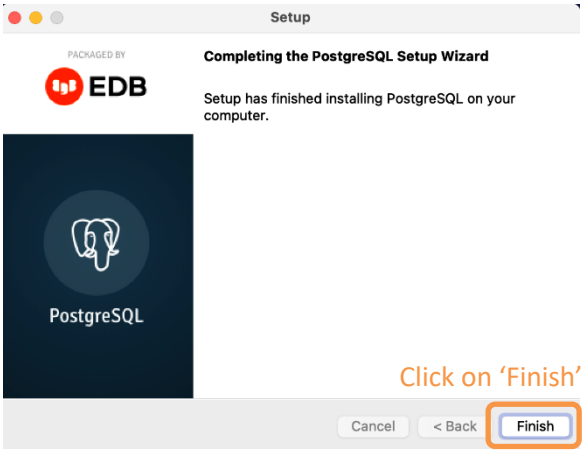

Setup
Password
Please provide a password for the database superuser (postgres). A locked Unix user account (postgres) will be created if not present.
Password
Retype password
Create your password
Remember this password, you will need it!
Click on 'Next'
VMware InstallBuilder
Cancel
< Back
Next >


Setup
Port
Please select the port number the server should listen on.
Port 5432
Select this port
Click on 'Next'
VMware InstallBuilder
Cancel
< Back
Next >

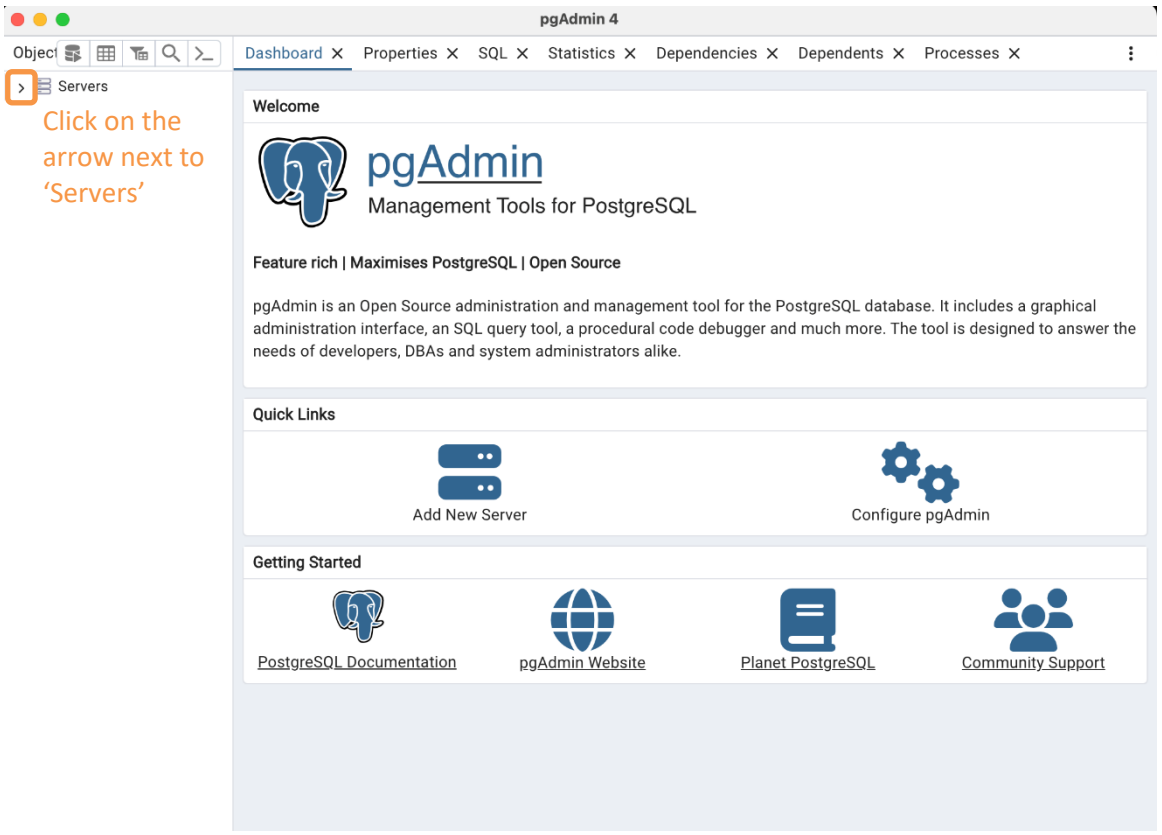

Setup
Advanced Options
Select the locale to be used by the new database cluster.
Locale [Default locale]
Click on 'Next'
VMware InstallBuilder
Cancel
< Back
Next >


Setup
Pre Installation Summary
The following settings will be used for the installation::
Installation Directory: /Library/PostgreSQL/16
Server Installation Directory: /Library/PostgreSQL/16
Data Directory: /Library/PostgreSQL/16/data
Database Port: 5432
Database Superuser: postgres
Operating System Account: postgres
Database Service: postgresql-16
Command Line Tools Installation Directory: /Library/PostgreSQL/16
pgAdmin4 Installation Directory: /Library/PostgreSQL/16/pgAdmin 4
Installation Log: /tmp/install-postgresql.log
Click on 'Next'
VMware InstallBuilder
Cancel
< Back
Next >

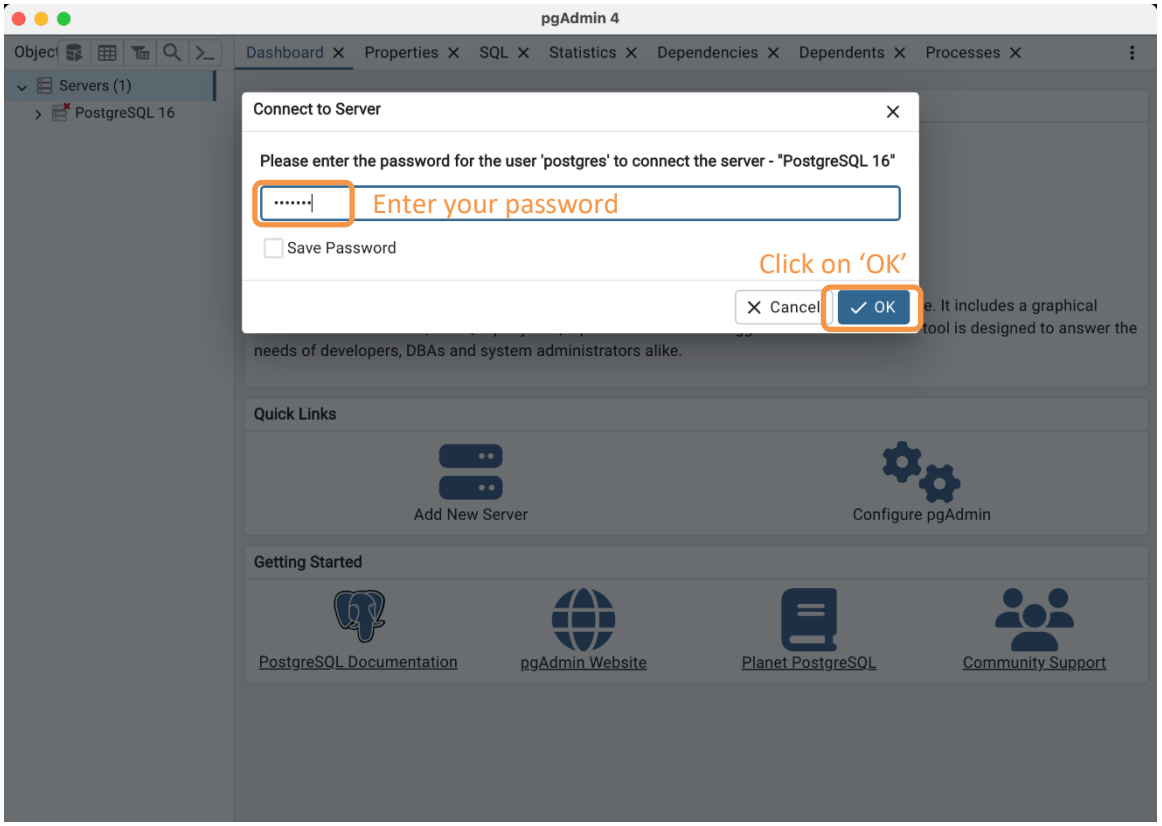




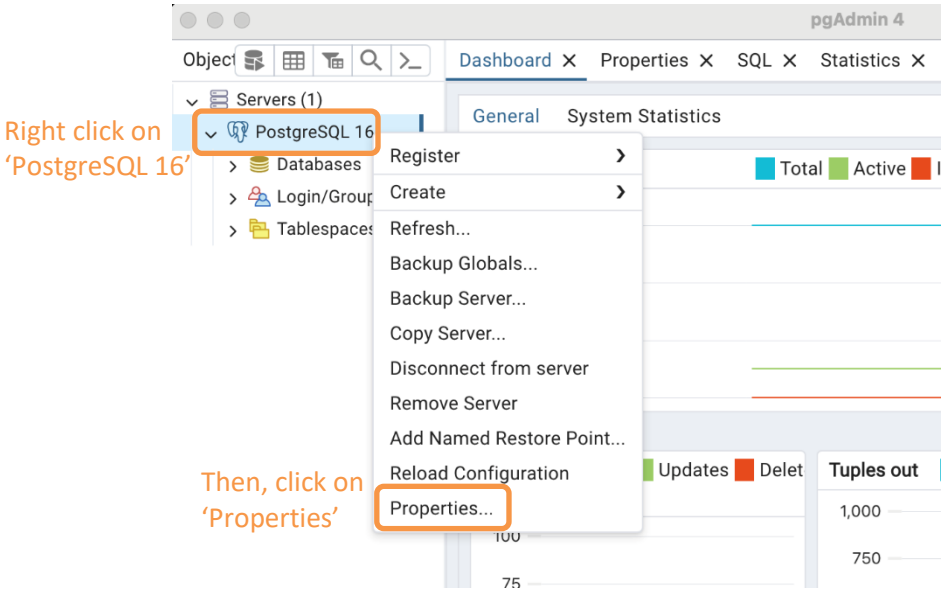
10. Open pgAdmin4 and from the menu on the left, click on the arrow on the left of "Servers".



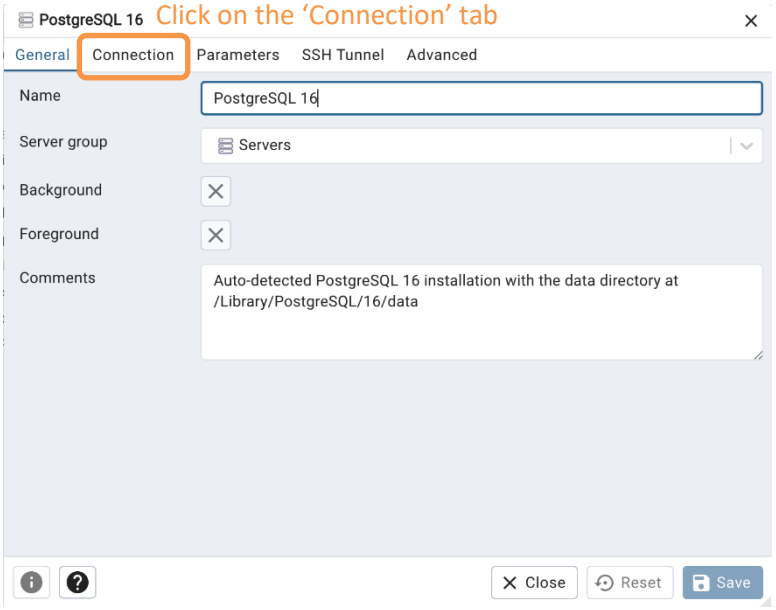
11. You will be asked to enter the password you created during the installation process. Type in your password and then click on ‘OK’.



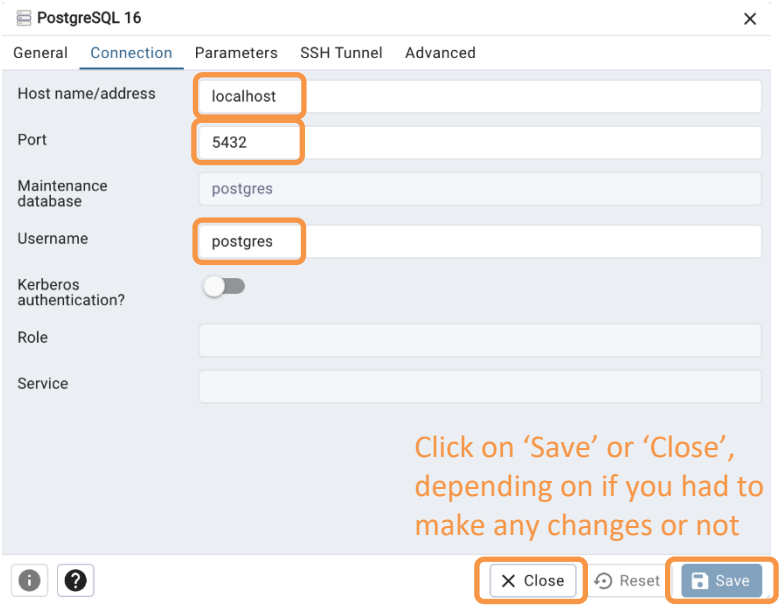
12. After you have entered your password you will be able to see the PostgreSQL Server on the menu on the left. Right click on ‘PostgreSQL 16’ (Note that this will vary depending on the version you installed), and then click on ‘Properties…’:



13. The properties window will open in a pop-up window. Click on the “Connection” tab:

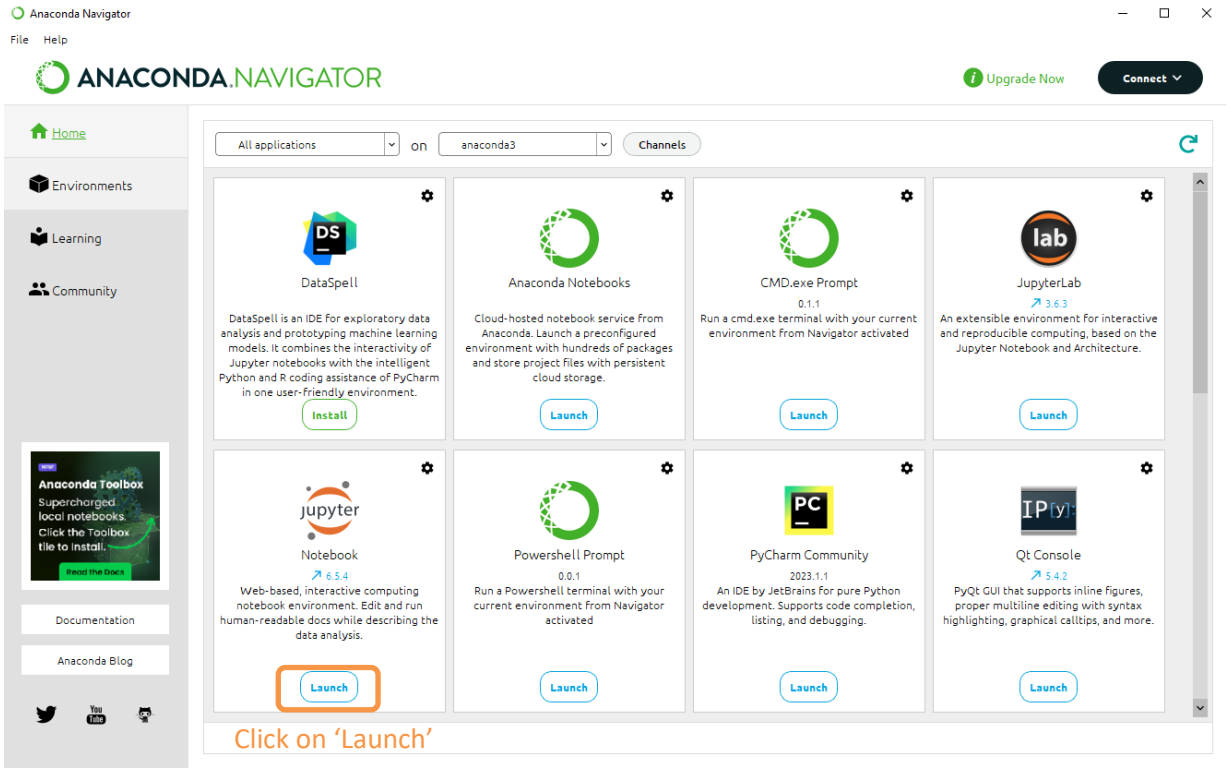


14. Make sure that the following options are selected and then click on ‘Save’ or ‘Close’, depending on if you had to make any changes or not.



Step 3: Connect and work with the server

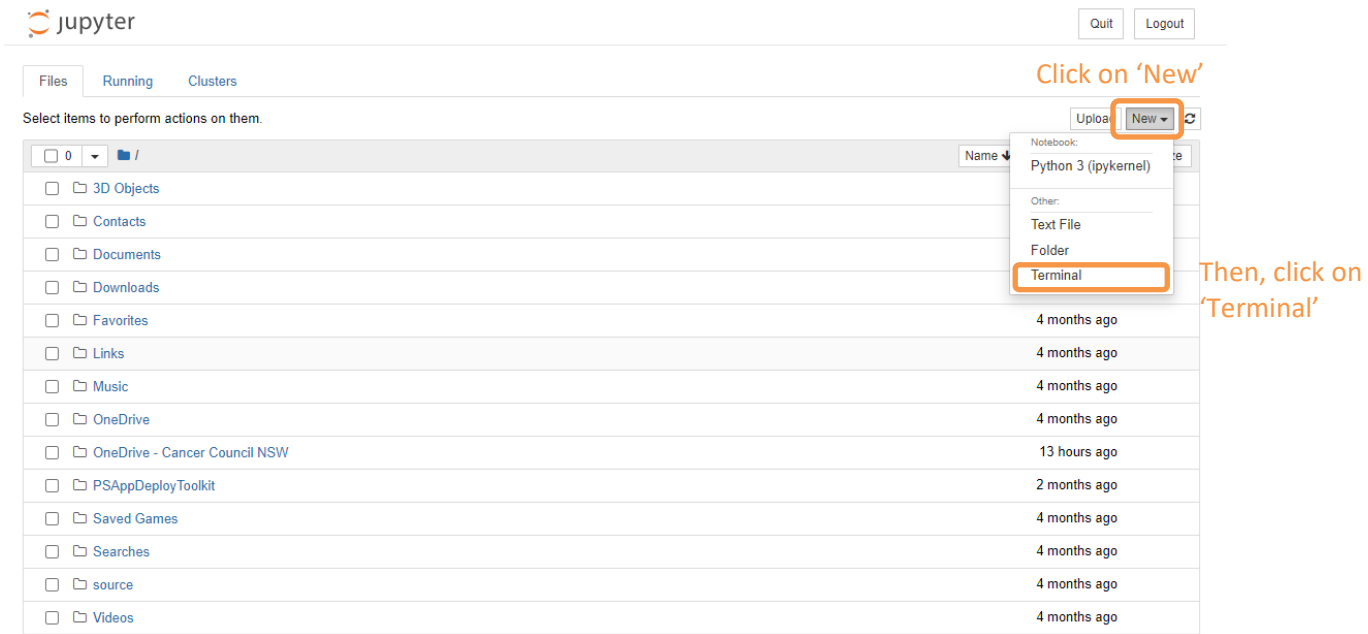
1. Open the Anaconda Navigator, and look for the Jupyter Notebook on the list of applications, then click on 'Launch':



2. A Jupyter file browser will open as a web browser tab.

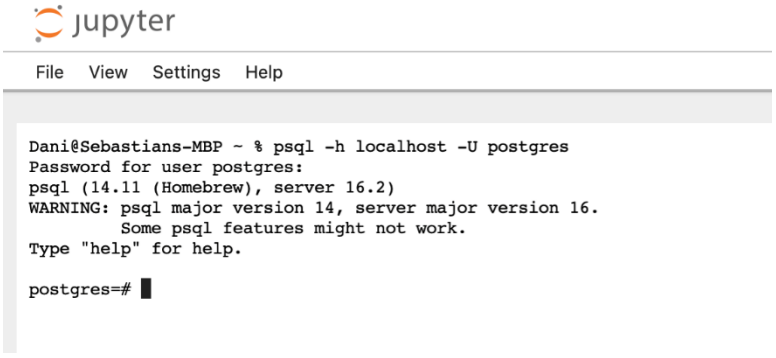
Option 1:

1. In the upper right select New → Terminal to open the terminal.



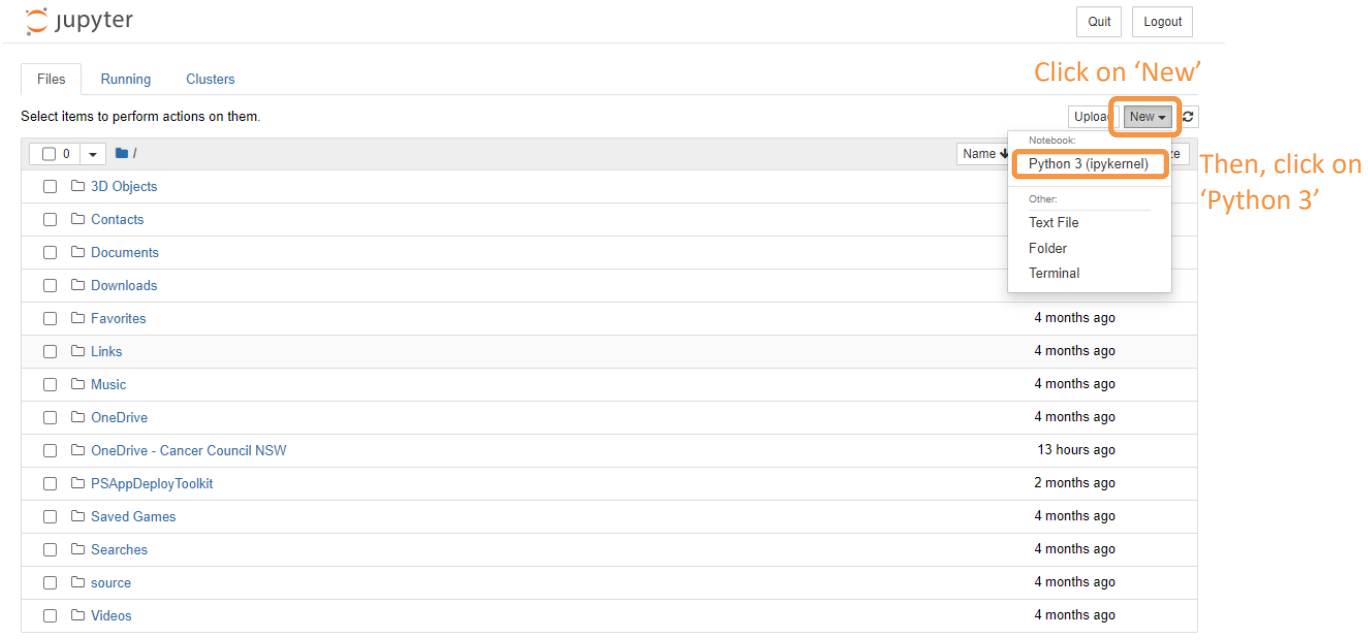
2. Once a terminal is opened, type the command "`psql -h localhost -U postgres`" and press 'Enter' on your keyboard. You will be asked to enter your password, which is the same you created when installing pgAdmin. Enter your password and press 'Enter' on your keyboard. Note that you will not see anything on the screen when typing your password (for security reasons), but it will work.

3. Then wait for the connection to be established and you will see "`postgres=#`" on the terminal:

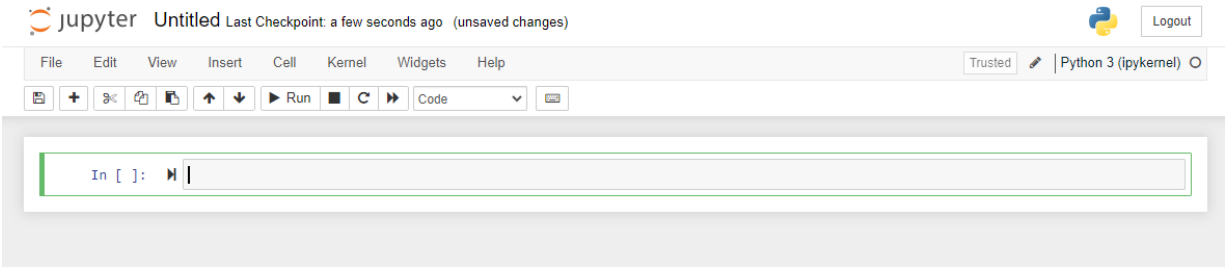


Option 2:

1. In the upper right select New → Python 3 to open a new Jupyter Notebook:



2. A new window will open for the Jupyter Notebook:



3. Copy the following code in the Jupyter Notebook:
   a. Install psycopg2 package. **Only need to run it once**:

```
pip install psycopg2
```

```
In [1]: pip install psycopg2
Collecting psycopg2
  Downloading psycopg2-2.9.9-cp311-cp311-win_amd64.whl.metadata (4.5 kB)
Downloading psycopg2-2.9.9-cp311-cp311-win_amd64.whl (1.2 MB)
   ---------------------------------------- 0.0/1.2 MB ? eta -:--:--
   - -------------------------------------- 0.0/1.2 MB 660.6 kB/s eta 0:00:02
   -------- ------------------------------- 0.2/1.2 MB 3.6 MB/s eta 0:00:01
   ----------------- ---------------------- 0.5/1.2 MB 4.9 MB/s eta 0:00:01
   ------------------------- ------------ 0.8/1.2 MB 5.6 MB/s eta 0:00:01
   ---------------------------------- -- 1.1/1.2 MB 5.7 MB/s eta 0:00:01
   ---------------------------------------- 1.2/1.2 MB 5.3 MB/s eta 0:00:00
Installing collected packages: psycopg2
Successfully installed psycopg2-2.9.9
Note: you may need to restart the kernel to use updated packages.
```

   b. Connect to the database server: **(make sure you maintain the correct indentations on the code, otherwise it won't work)**

```
import psycopg2
def pgconnect():
  """ Connect to the PostgreSQL database server """
  conn = None
  try:
    # connect to the PostgreSQL server
    print('Connecting to the PostgreSQL database...')
    conn = psycopg2.connect(host = 'localhost',
            database = 'postgres',
            user = 'postgres',
            password = 'abcd1234')
  print('Connected')

  except Exception as e:
    print("Unable to connect to the database")
    print(e)
  return conn

conn = pgconnect()
```

Insert the password you created when installing pgAdmin

```
import psycopg2
def pgconnect():
    """ Connect to the PostgreSQL database server """
    conn = None
    try:
        # connect to the PostgreSQL server
        print('Connecting to the PostgreSQL database...')
        conn = psycopg2.connect(host = 'localhost',
                                database = 'postgres',
                                user = 'postgres',
                                password = 'dani123')
        print('Connected')

    except Exception as e:
        print('Unable to connect to the database')
        print(e)

    return conn

conn = pgconnect()
```

```
Connecting to the PostgreSQL database...
Connected
```

You will see this message if successfully connected to the database