



postgreSQL,  
pgAdmin,  
and creating the  
`data_analyst_jobs` table

PostgreSQ**L**

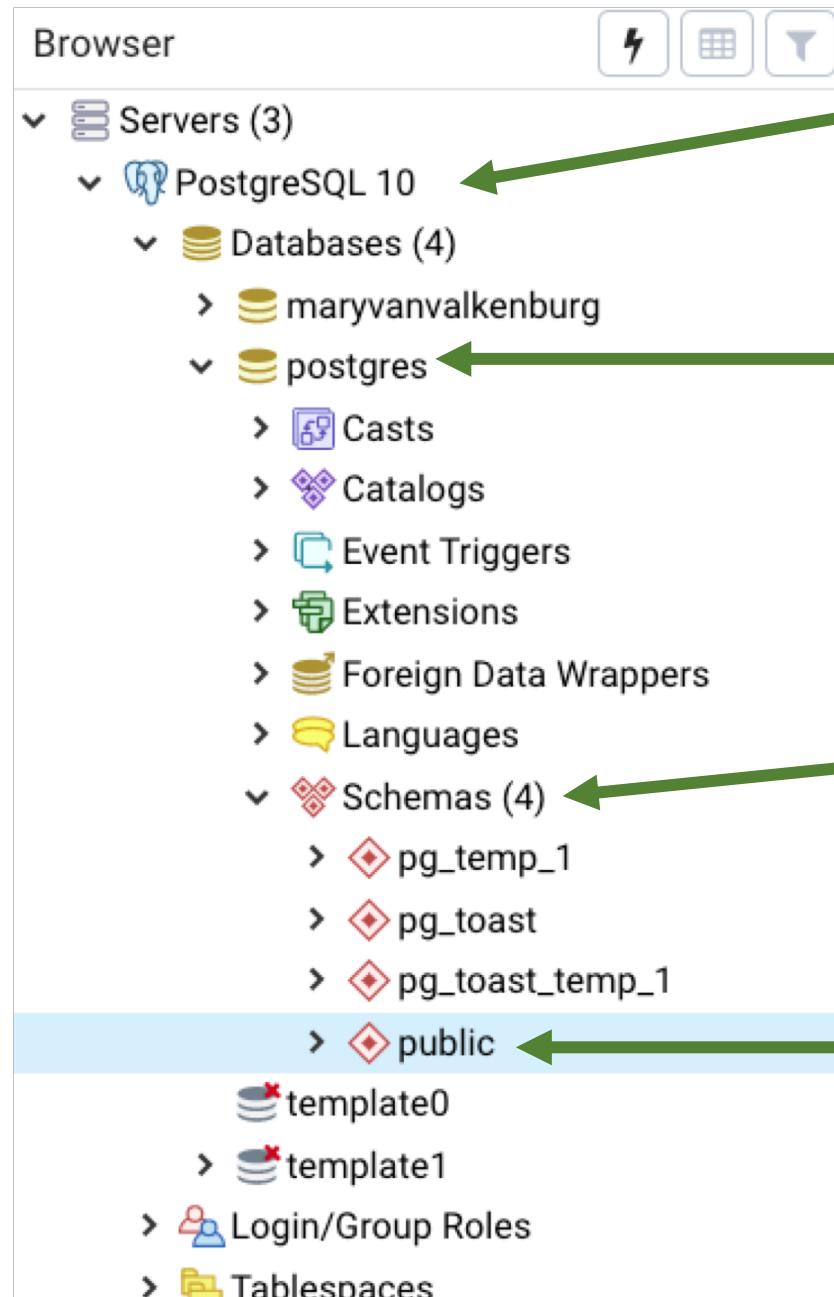
## 1. Install PostgreSQL

1. Mac users install the postgres.app - (<https://postgresapp.com/>)
2. Windows Users install version 12.0 - (<https://www.enterprisedb.com/downloads/postgres-postgresql-downloads>)
  - accept the defaults (PgAdmin will be installed with PostgreSQL)

## 2. Mac Users Install PgAdmin

1. Mac users install the dmg file here - <https://www.postgresql.org/ftp/pgadmin/pgadmin4/v4.13/macos/>
- 

Since this is not secure data and it is for learning use, I ***highly recommend*** setting the postgres and pgAdmin administrative passwords to **postgres**. This is not something you would do with secure data in a production environment!

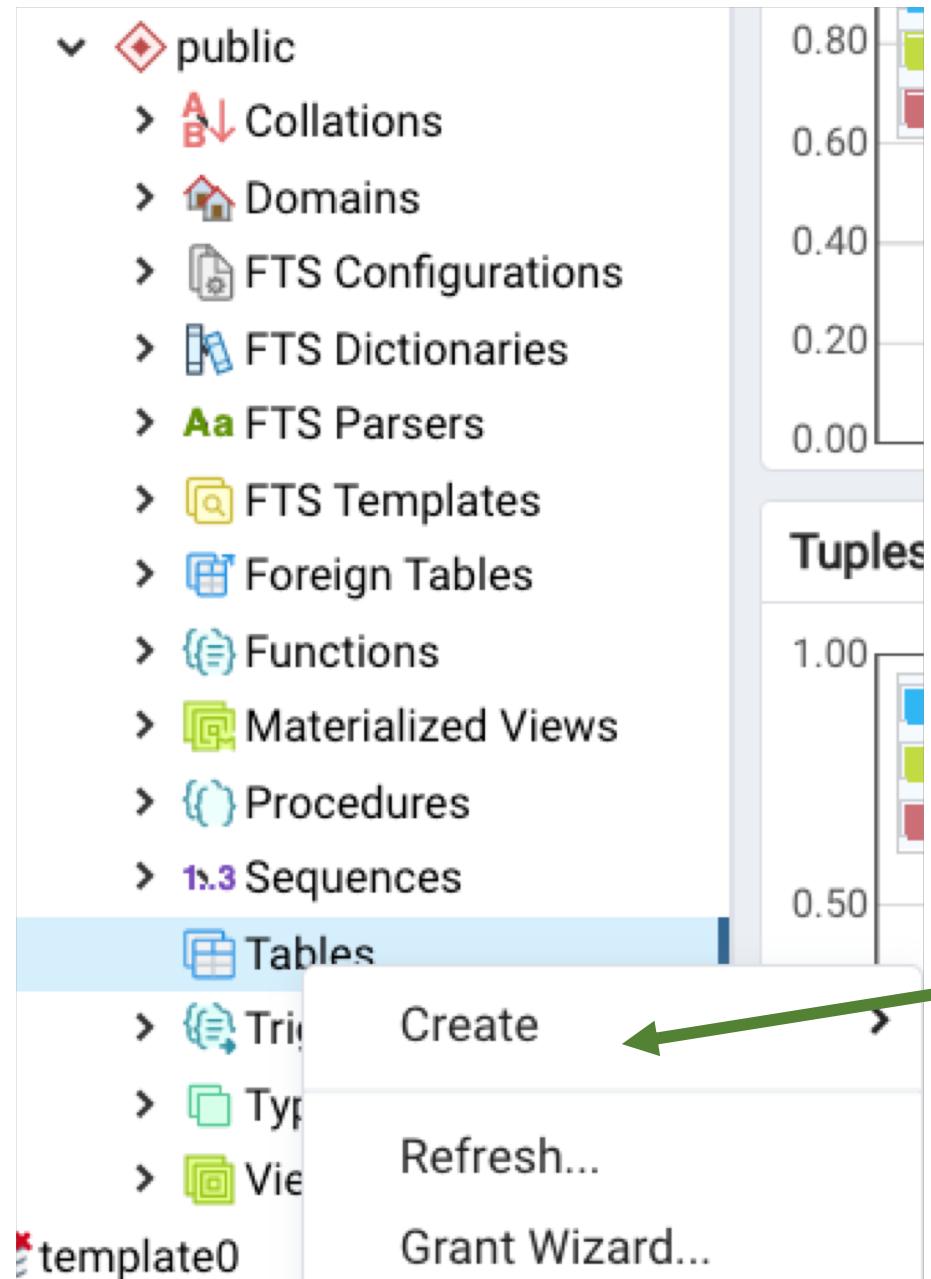


On the PostgreSQL 10 Server

In the postgres database

Expand the Schemas

Find the public schema



Right-click on Tables → Create → Table...

Under the **General** tab

- give your table the name  
**data\_analyst\_jobs**

Create - Table

General Columns Constraints Advanced Partition Parameters Security SQL

Name: data\_analyst\_jobs

Owner: postgres

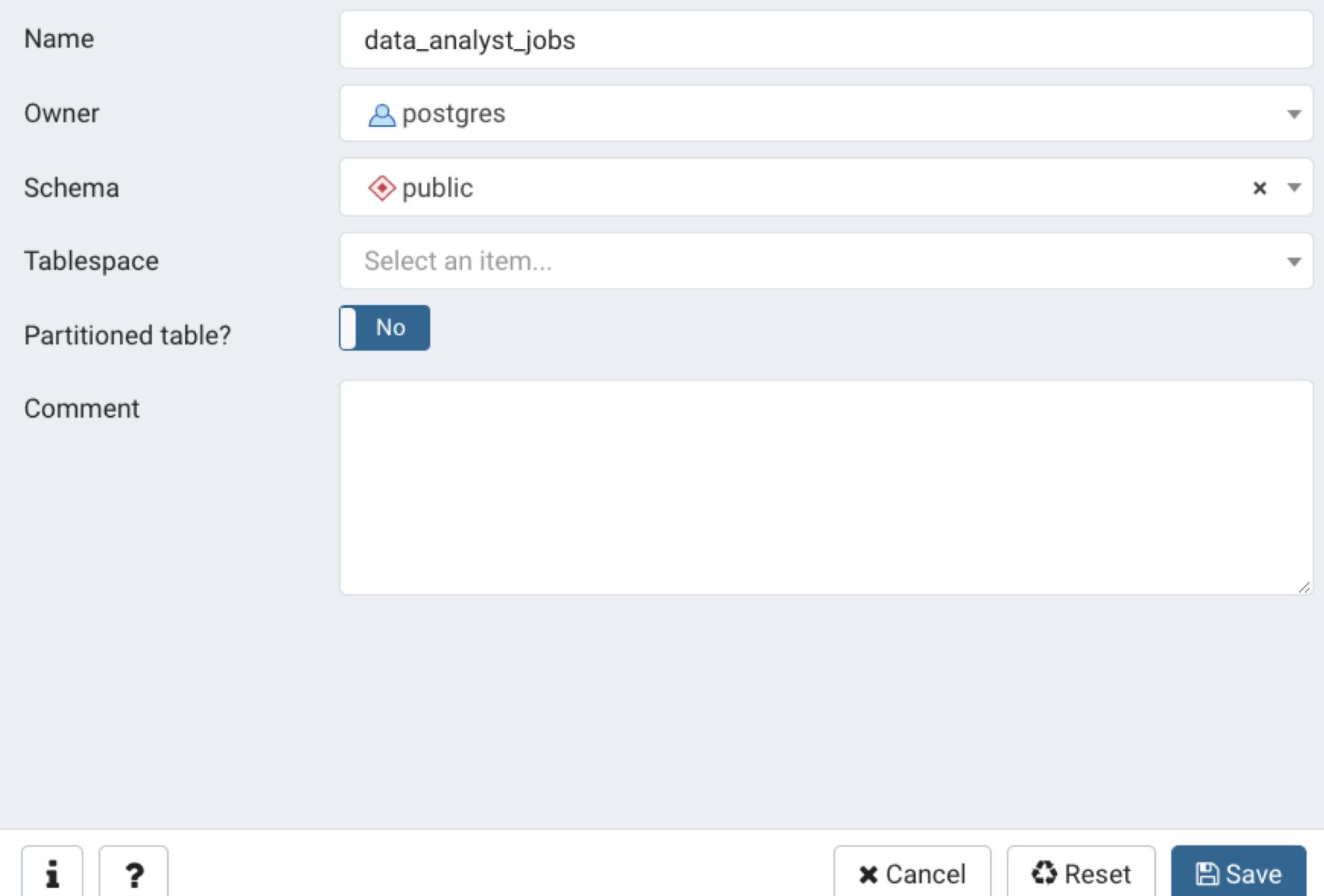
Schema: public

Tablespace: Select an item...

Partitioned table?: No

Comment:

i ? Cancel Reset Save



## Under the **Columns** tab

- define your columns with the names and data types pictured

data\_analyst\_jobs

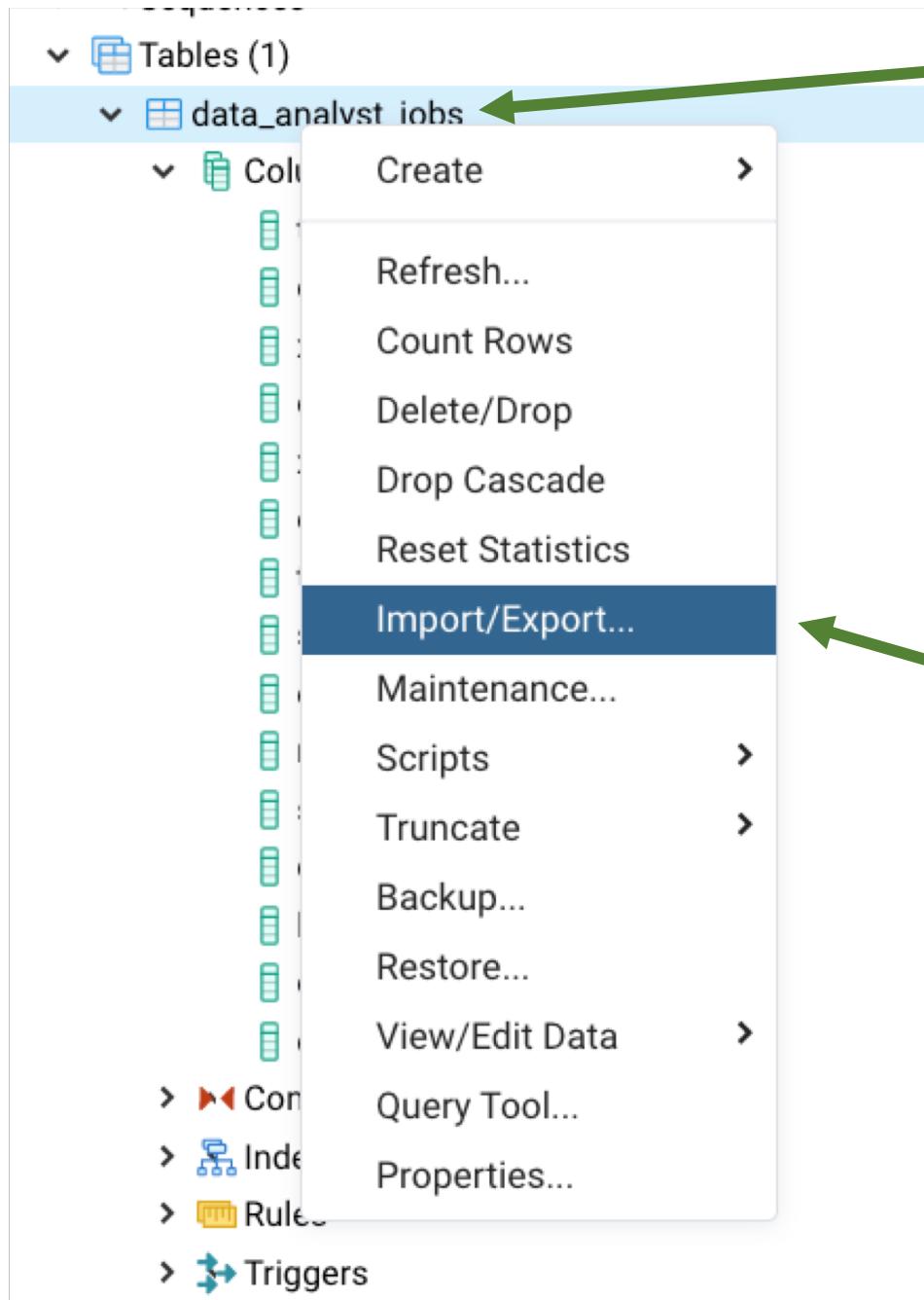
General Columns Constraints Advanced Parameters Security SQL

Inherited from table(s) Select to inherit from...

Columns

	Name	Data type	Length	Precision	Not NULL?	Primary key?
	title	text			<input type="checkbox"/> No	<input type="checkbox"/> No
	skill	text			<input type="checkbox"/> No	<input type="checkbox"/> No
	company	text			<input type="checkbox"/> No	<input type="checkbox"/> No
	review_count	integer			<input type="checkbox"/> No	<input type="checkbox"/> No
	star_rating	numeric			<input type="checkbox"/> No	<input type="checkbox"/> No
	days_since_posting	integer			<input type="checkbox"/> No	<input type="checkbox"/> No
	location	text			<input type="checkbox"/> No	<input type="checkbox"/> No
	company_size	text			<input type="checkbox"/> No	<input type="checkbox"/> No
	domain	text			<input type="checkbox"/> No	<input type="checkbox"/> No

Cancel Reset Save



Right-click on the **data\_analyst\_jobs** table

Select **Import/Export**

Import/Export data - table 'data\_analyst\_jobs'

Options Columns

Import/Export Import ← Be sure to toggle to Import

**File Info**

Filename: /Users/maryvanvalkenburg/NSS/random\_data/job\_titles/indeed\_data\_analysts.csv ... → Click the ellipsis (...) and browse to the csv file

Format: csv

Encoding: Select an item...

**Miscellaneous**

OID: No This file **does** have a header row

Header: Yes ← It's comma delimited

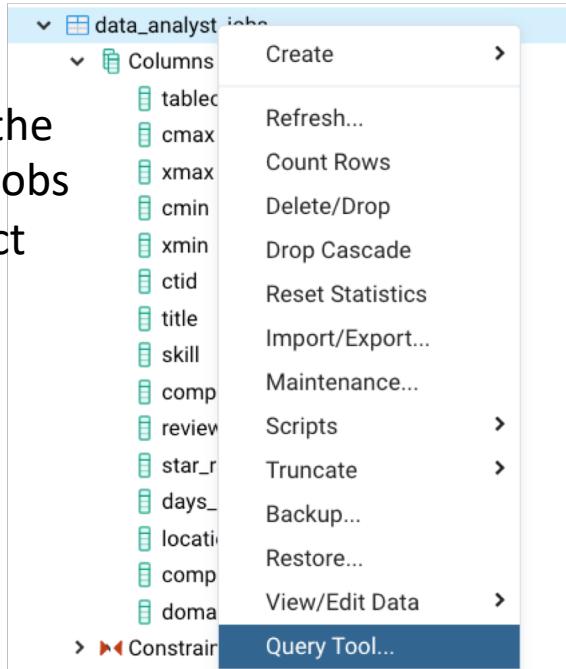
Delimiter: ,  
Specifies the character that separates columns within each row (line) of the file. The default is a tab character in text format, a comma in CSV format. This must be a single one-byte character. This option is not allowed when using binary format.

Quote: "  
Specifies the quoting character to be used when a data value is quoted. The default is double-quote. This must be a single one-byte character. This option is allowed only when using CSV format.

# Once you get a **success** message:

1.

Right-click on the `data_analyst_jobs` table and select 'Query Tool'



2.

Run a query to count the rows in your new table.

A screenshot of a 'Query Editor' window. The tab bar shows 'Query Editor' and 'Query History'. The main area contains a single line of SQL code: '1 Select count (\*) from data\_analyst\_jobs;'. Below the editor is a 'Data Output' tab. A table is displayed with one row: 'count' (datatype bigint) has a value of 1793. There are also tabs for 'Explain', 'Messages', and 'Notifications'.

	count	bigint
1	1793	

Use comments to keep your SQL commands and answers:

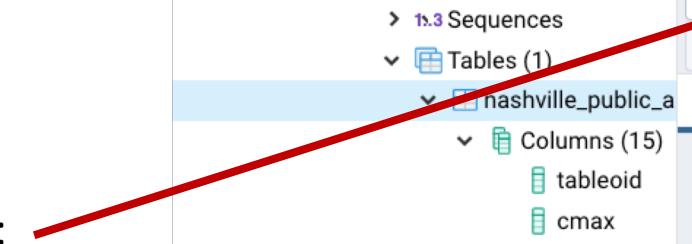
`/* multiline comment`

Uses these

Symbols \*/

-- single line comments like this

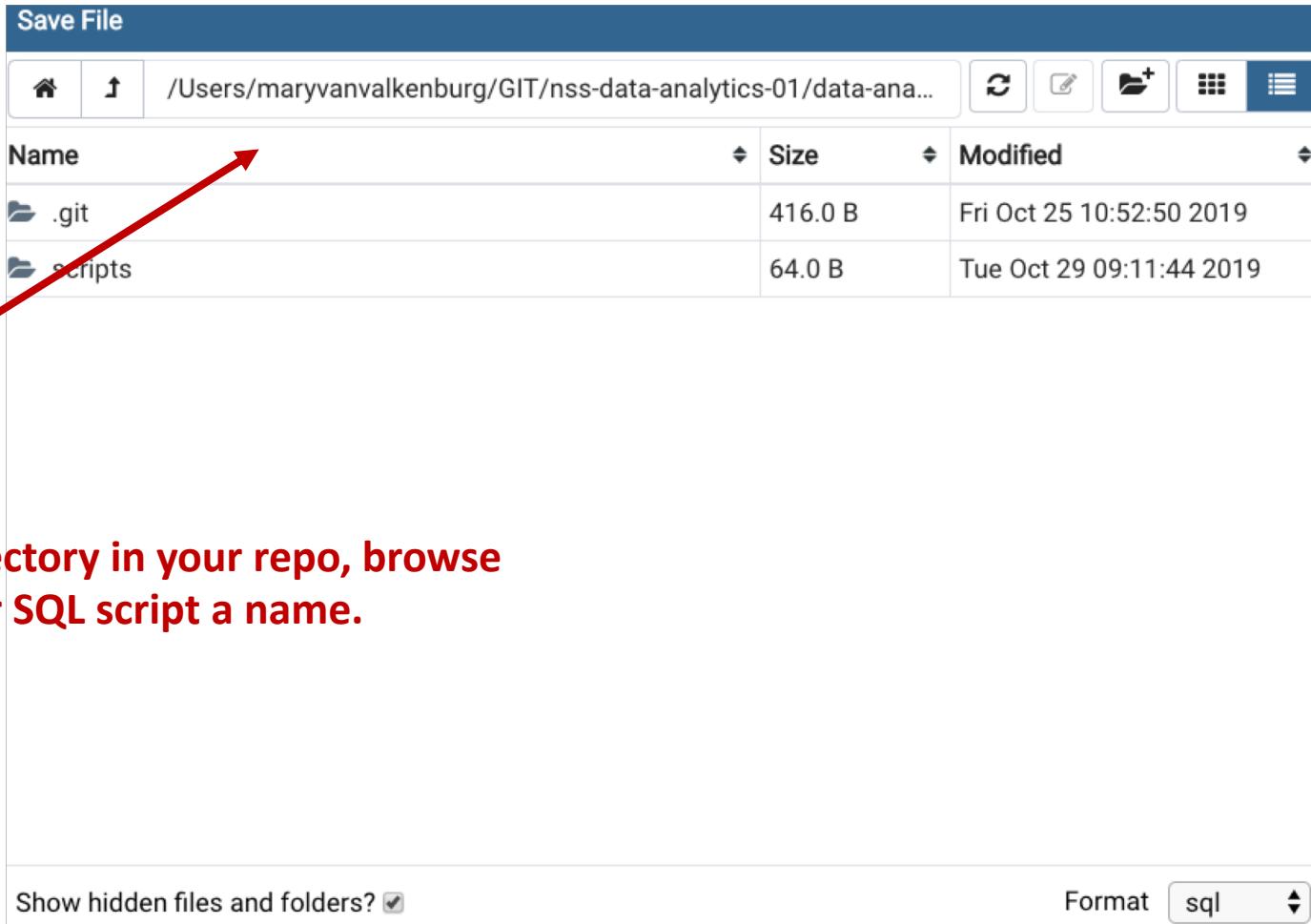
And save your script after working:



The screenshot shows the pgAdmin interface. On the left, the 'Browser' panel displays database objects: Materialized Views, Sequences, and Tables (1). The 'nashville\_public\_art' table is selected, showing 15 columns: tableoid, cmax, xmax, cmin, xmin, ctid, title, last\_name, first\_name, address, and medium. On the right, the 'Query Ed' (Query Editor) panel contains the following SQL script:

```
1 /* Select distinct(medium)
2  from nashville_public_art; */
3
4 Select *
5 from nashville_public_art
6 where medium = 'Latex Paint';
7
```

The 'Save As' button in the toolbar is highlighted with a red arrow. The toolbar also includes other buttons for Save, Undo, Redo, Copy, Paste, Find, and Delete.



Create a script directory in your repo, browse to it, and give your SQL script a name.

Be sure SQL format is selected.

**When you return to  
continue working, use  
the folder icon to  
open your SQL script.**

