


Postgres and pgAdmin

DATA ANALYTICS BOOTCAMP

NASHVILLE  SOFTWARE SCHOOL

Install PostgreSQL and pgAdmin

- Download and install the latest version of PostgreSQL for your OS:
<https://www.enterprisedb.com/downloads/postgres-postgresql-downloads>
 - For Mac and Windows 64-bit, this will be Version 15.1
 - For Windows 32-bit, this will be Version 10.23
 - Linux users will need to use the following link:
<https://www.postgresql.org/download/>. You will also need to install pgAdmin. Installation instructions can be found here:
<https://www.pgadmin.org/download/>.
- During installation you can accept all the defaults **except** you can uncheck **Stack Builder** on the **Select Components** menu
- When asked for a database password, use **postgres**
- Confirm the port number is **5432**



Very important to use this password!

*Note: Since this is not secure data, we can set our passwords to **postgres**. This is not something you would do with secure data in a production environment*

Launch pgAdmin

- Either search for **pgAdmin** in the search bar or navigate to it in your file browser
- Once launched, it will prompt you for a password, which you should have set to **postgres** during the install.

Unlock Saved Passwords

✕

Please enter your master password.

This is required to unlock saved passwords and reconnect to the database server(s).



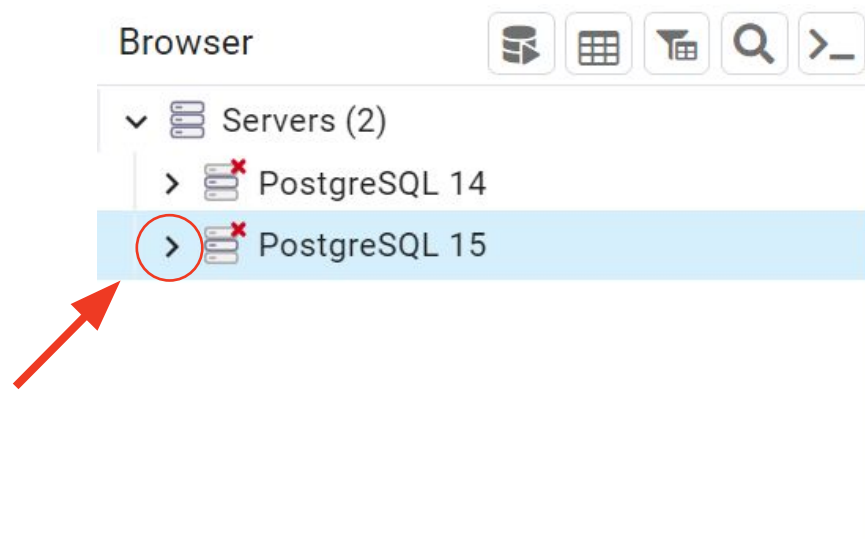
Reset Master Password

✕ Cancel

✓ OK

Access Server

- Click the arrow next to **PostgreSQL 15** in the left hand menu



Access Server

- You will be prompted to enter a password for the server, which is also **postgres**
- Check the **Save Password** box so you won't be asked for a password at this stage again
- Click **OK**

Connect to Server

✕

Please enter the password for the user 'postgres' to connect the server - "PostgreSQL 15"

.....

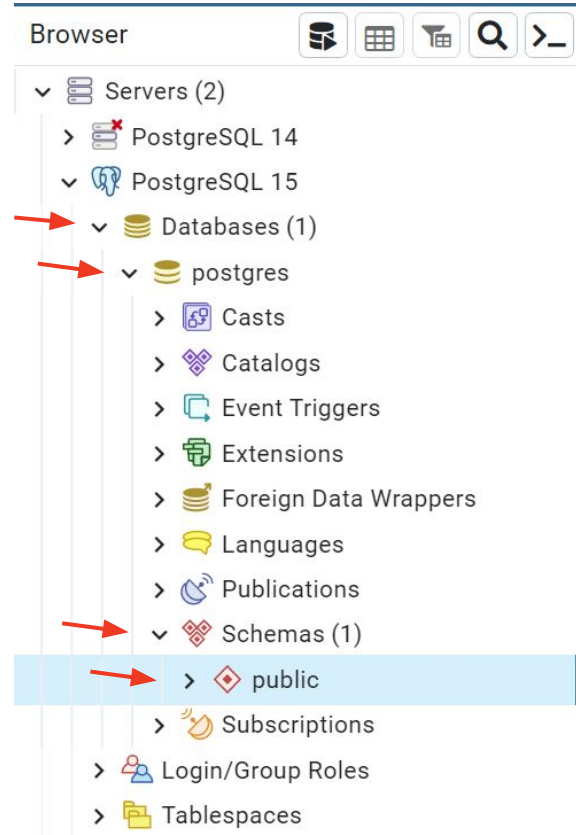
☒ Save Password

✕ Cancel

✓ OK

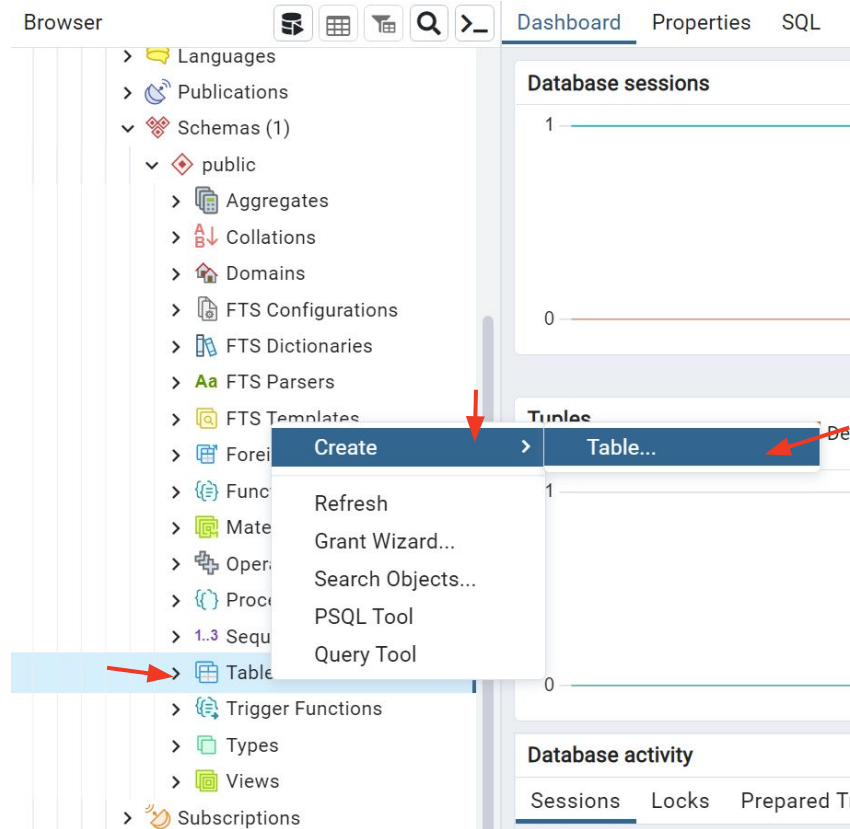
Create a Table

- Click the arrow next to **Databases**
- Click the arrow next to **postgres**
- Click the arrow next to **Schemas**
- Click the arrow next to **public**



Create a Table

- Right-click on **Tables**
- Hover over **Create**
- Select **Table**



Create a Table

- In the **General** tab, name the table `data_analyst_jobs`

Create - Table

General Columns Advanced Constraints Partitions Parameters Security SQL

Name data_analyst_jobs

Owner postgres

Schema public

Tablespace Select an item...

Partitioned table? ☐

Comment

i ? *X Close* *Reset* *Save*

Create a Table

- Under the **Columns** tab define your columns with the **names** and **data types** shown on the next slide
 - The names need to match what you see **EXACTLY**; be careful not to put in any extra spaces at the beginning or end
 - **Click the + symbol** in the upper right hand corner of the window to create a new column
 - The only fields you should have to edit are **Name** and **Data type**
 - These column names and types correspond to the columns in the csv we will be using
- Click **Save** when you are done

Create a Table

Create - Table

GeneralColumnsAdvancedConstraintsPartitionsParametersSecuritySQL

Inherited from table(s)

Select to inherit from...

Columns

+

		Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?	Default
⋮	✎	title	text			<input type="checkbox"/>	<input type="checkbox"/>	
⋮	✎	skill	text			<input type="checkbox"/>	<input type="checkbox"/>	
⋮	✎	company	text			<input type="checkbox"/>	<input type="checkbox"/>	
⋮	✎	review_count	integer			<input type="checkbox"/>	<input type="checkbox"/>	
⋮	✎	star_rating	numeric			<input type="checkbox"/>	<input type="checkbox"/>	
⋮	✎	days_since_posting	integer			<input type="checkbox"/>	<input type="checkbox"/>	
⋮	✎	location	text			<input type="checkbox"/>	<input type="checkbox"/>	
⋮	✎	company_size	text			<input type="checkbox"/>	<input type="checkbox"/>	
⋮	✎	domain	text			<input type="checkbox"/>	<input type="checkbox"/>	

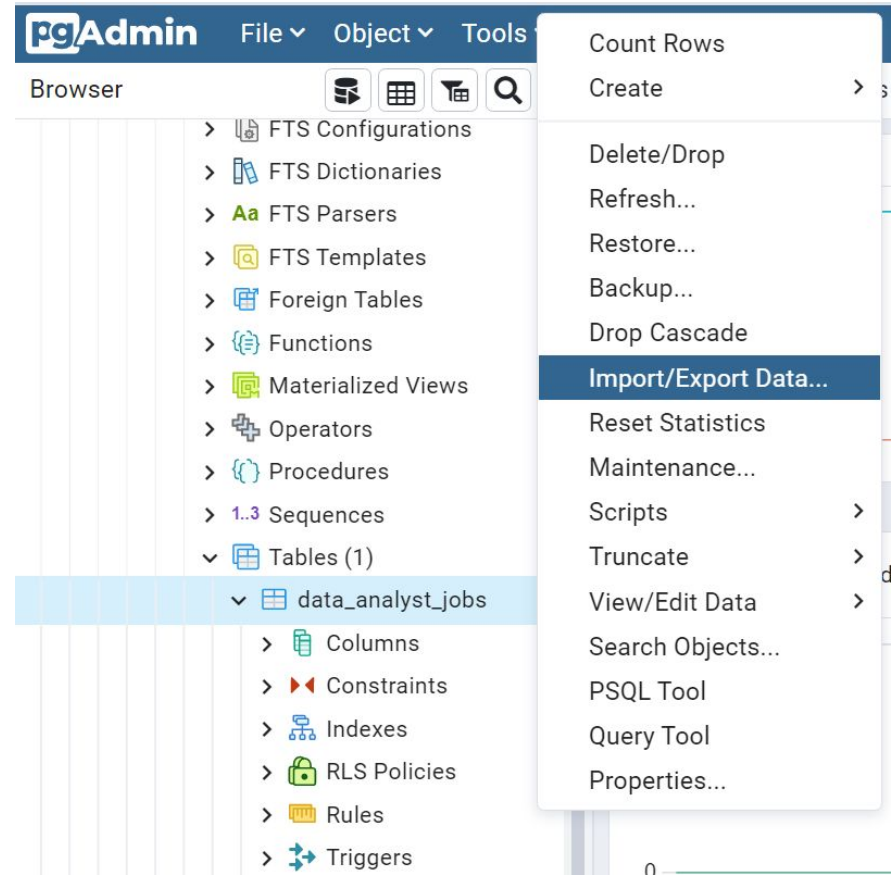
i?

CloseResetSave

Create a Table

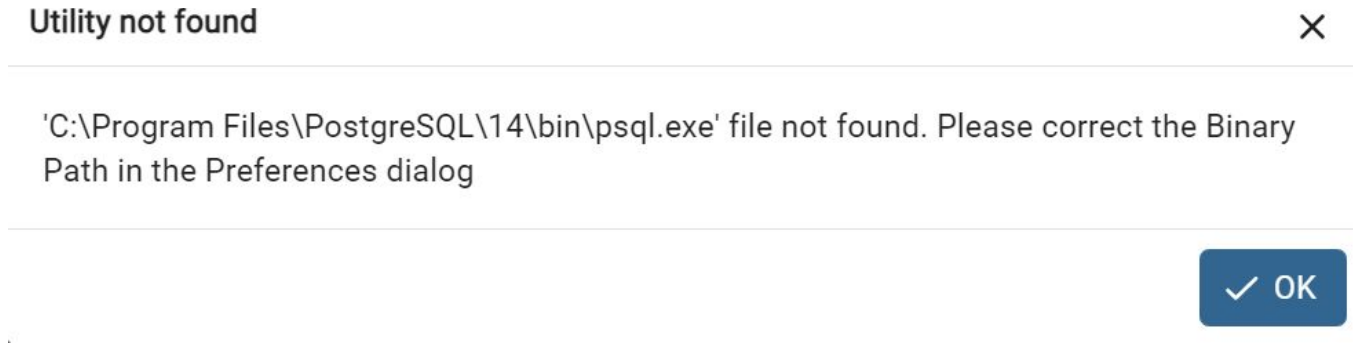
After clicking Save...

- Right-click on the **data_analyst_jobs** table
- Select **Import/Export Data...**



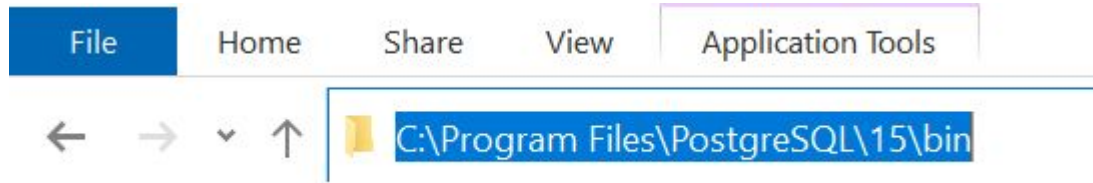
Utility not found

- If you get this error message (or anything similar that mentions the Binary Path), you must take extra steps to set the binary path.



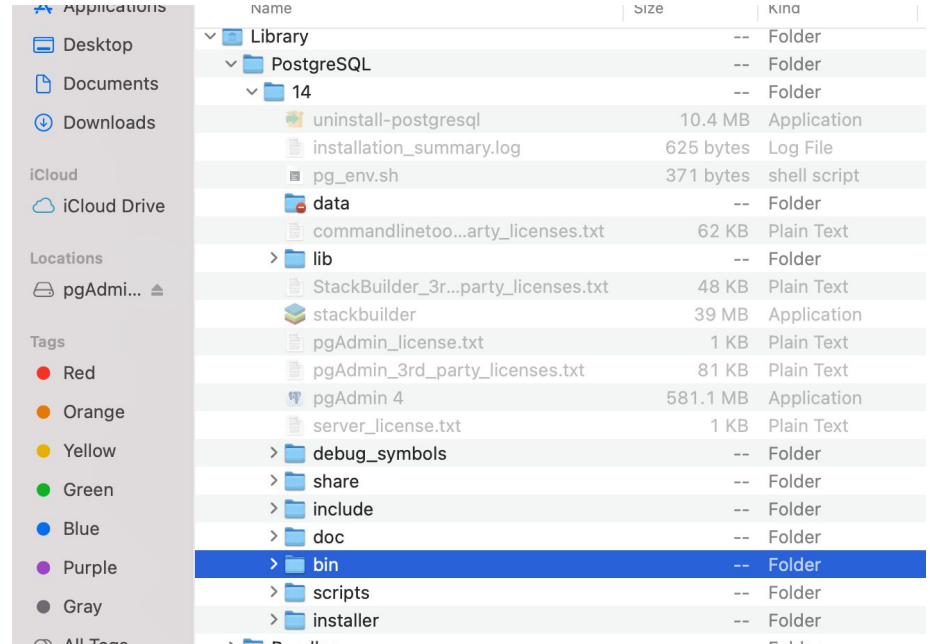
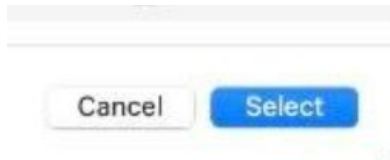
Utility not found

- On Windows, navigate to the **Program Files** folder
- Find the **PostgreSQL** folder and inside. There should be a **15** folder with a **bin** folder inside of that (14 for Macs)
- Click in the address bar and copy the **entire path** once you are inside the **bin** folder
- It should look something like this:



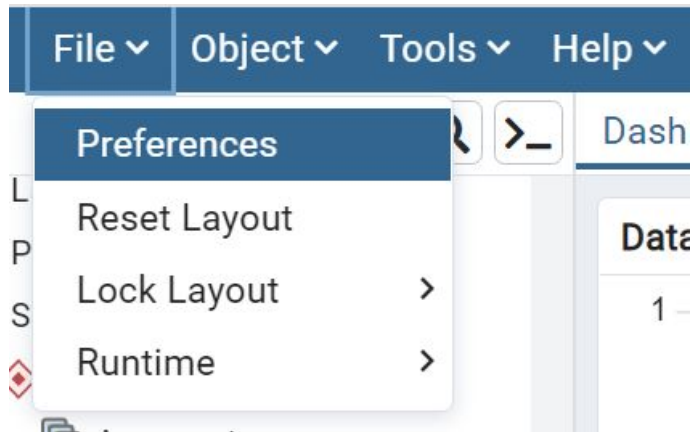
Utility not found

- On Mac, navigate to the **Macintosh HD** folder
- From there go to **Library** → **PostgreSQL** → **15** and select **bin** (14 for Macs)
- Click **Select** in the bottom right corner



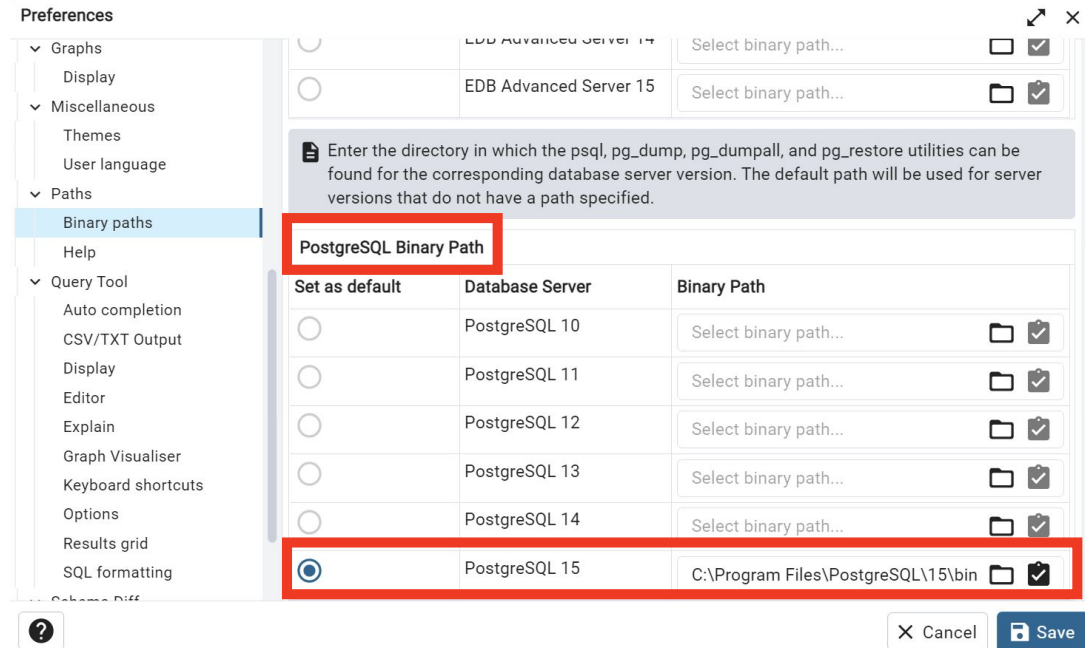
Utility not found

- In pgAdmin, Go to **File** → **Preferences**



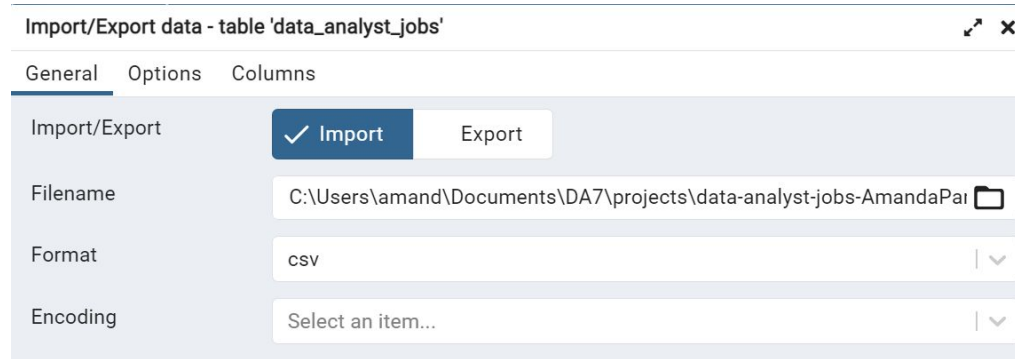
Utility not found

- In the left hand menu, scroll down to **Paths** and click on **Binary Paths**
- Scroll down to the second set of servers called **PostgreSQL Binary Path**
- Paste the path into the **Binary Path** space next to **PostgreSQL 15 (14 for Macs)**
- Check the button next to it under **Set as default**
- Click **Save**



Create a Table


- In the **General** tab, be sure the **Import/Export** option is toggled to **Import**
Selecting Export will clear out the contents of the CSV and you will have to clone down the data again
- Click on the **folder** icon and navigate to the file **indeed_data_analysts.csv** in the data folder of the assignment repo
- Double click on the file





Import/Export data - table 'data_analyst_jobs'

General Options Columns

Import/Export ☒ Import ☐ Export

Filename C:\Users\amand\Documents\DA7\projects\data-analyst-jobs-AmandaPa... 

Format csv 

Encoding Select an item... 

Create a Table

- In the **Options** tab, change the **Header** button to **blue** to indicate our file has headers for the columns
- Be sure the **Delimiter** is set to “,”
- Click **OK**

Import/Export data - table 'data_analyst_jobs'

General Options Columns

OID ☐

Header ☒

Delimiter | v

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 | v

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.

Escape | v

Specifies the character that should appear before a data character that matches the QUOTE value. The default is the same as the QUOTE value (so that the quoting character is doubled if it appears in the data). This must

Success!

- Your data will import and show the following success message

Process started ✕

Copying table data 'public.data_analyst_jobs' on database 'postgres' and server 'PostgreSQL 15 (localhost:5432)'

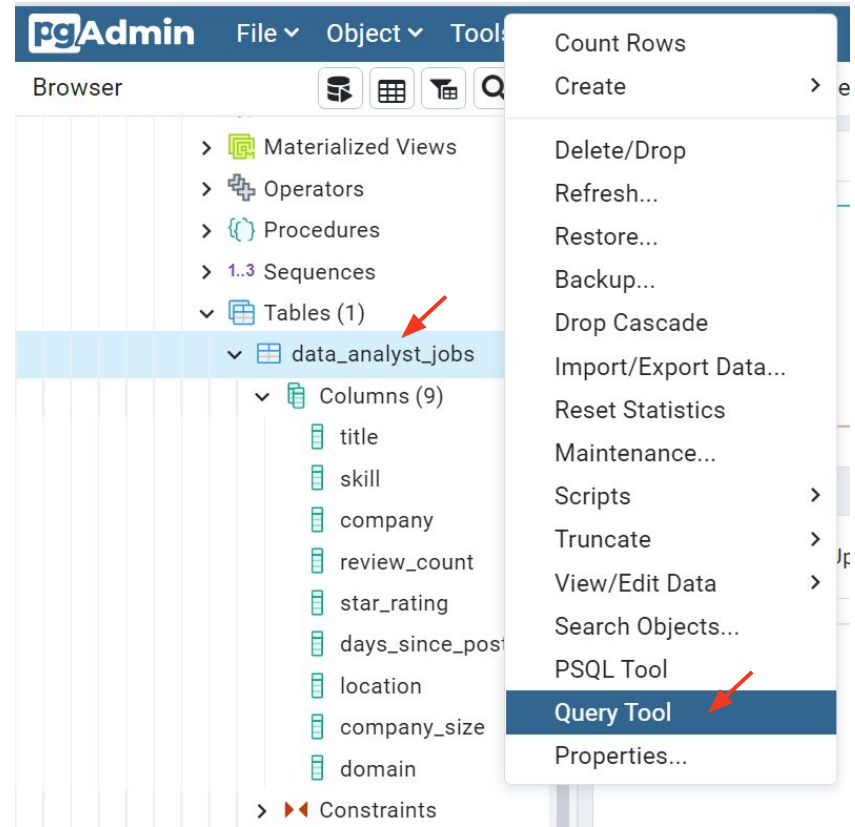
 [View Processes](#)



pgAdmin Key Features & Tips

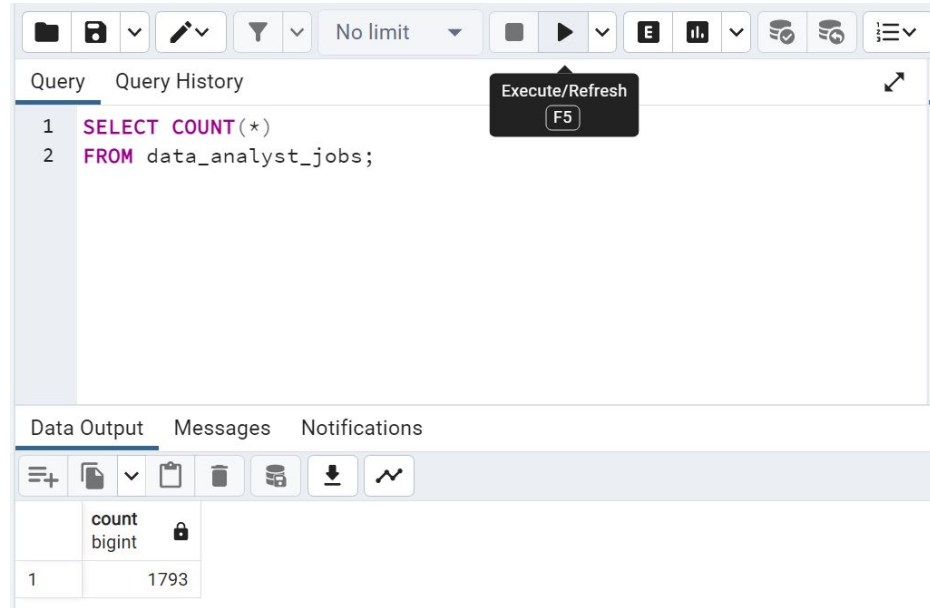
Starting a Script

- To start a new script where you will write your queries, **right-click** on **data_analyst_jobs** and select **Query Tool**



Starting a Script

- To test that you have imported successfully, run the query shown to make sure you have the right number of rows
- You're now ready to write your own queries



The screenshot displays a SQL query editor interface. At the top, there is a toolbar with various icons for file operations, editing, and execution. Below the toolbar, the 'Query' tab is active, showing a SQL query:

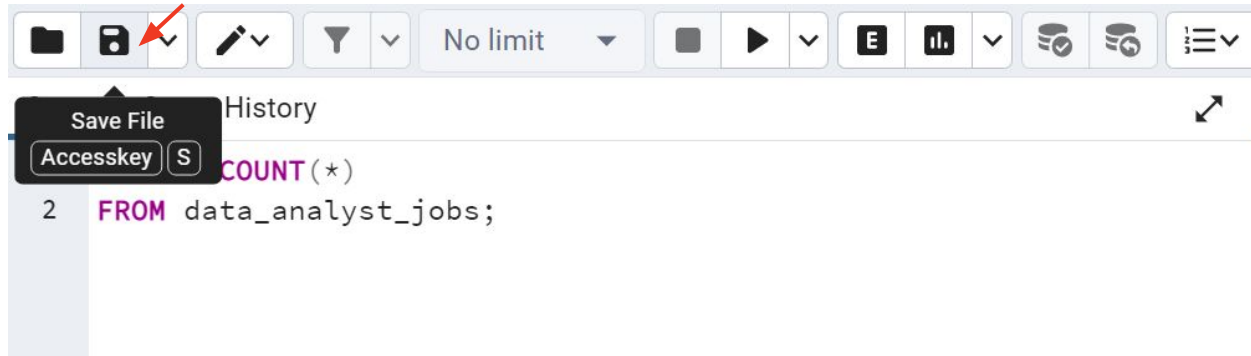
```
1 SELECT COUNT(*)
2 FROM data_analyst_jobs;
```

A tooltip labeled 'Execute/Refresh' with the keyboard shortcut 'F5' is visible over the execution button in the toolbar. Below the query editor, the 'Data Output' tab is active, displaying the results of the query in a table format:

	count bigint	
1	1793	

Saving Your Script

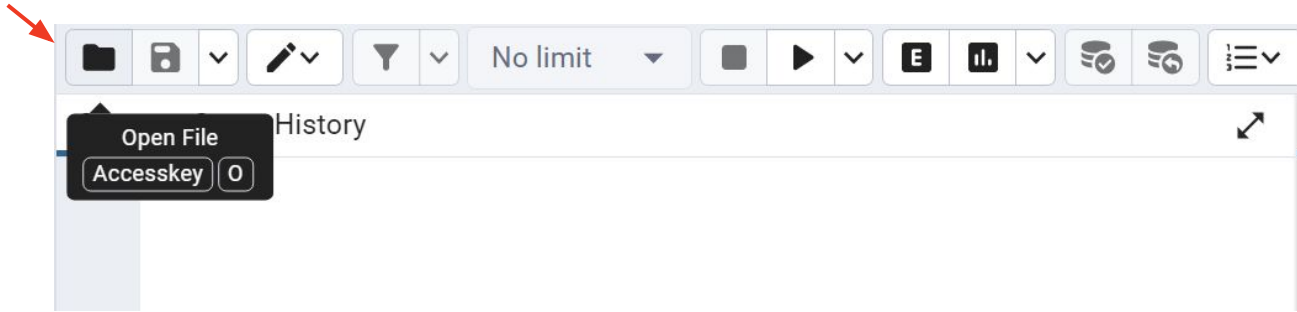
- To save, first click the floppy disk icon at the top left of the query tool



- If not already created, create a folder called **scripts** within your SQL folder
- Navigate into the **scripts** folder, name the file something meaningful, ensure the file type is *.sql and click **Save**

Opening Your Script

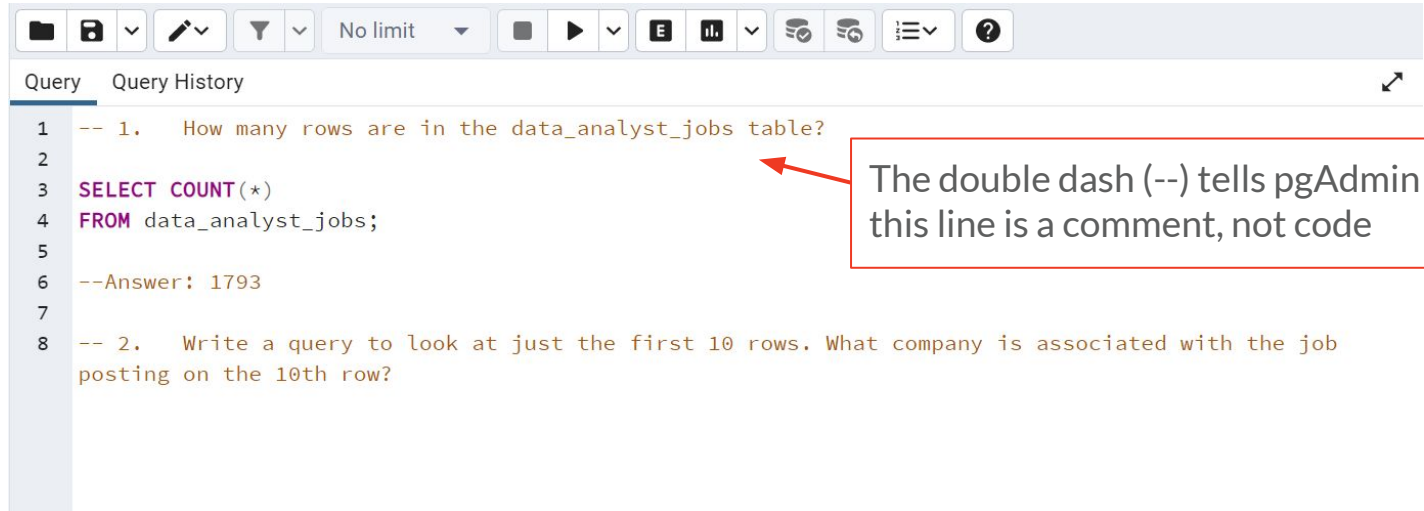
- When you return to work on the script you saved, click on the folder icon



- Navigate back to the **scripts** folder and select your saved script

Formatting Questions and Answers

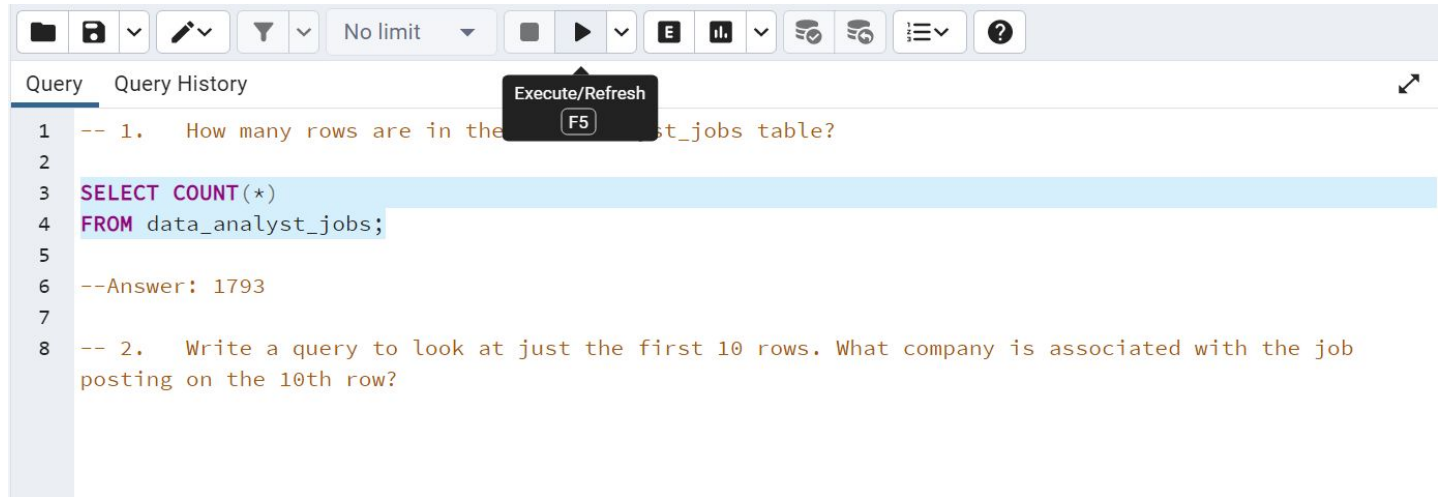
In order for your answers to be clearly labeled and easy to read through, put the question first as a comment, then the answer query, followed by the answer as a comment



```
1  -- 1.   How many rows are in the data_analyst_jobs table?
2
3  SELECT COUNT(*)
4  FROM data_analyst_jobs;
5
6  --Answer: 1793
7
8  -- 2.   Write a query to look at just the first 10 rows. What company is associated with the job
        posting on the 10th row?
```

Formatting Questions and Answers

When you have multiple queries in a script, it is good practice to highlight the query you are working on before running it to ensure queries before it won't be run first.

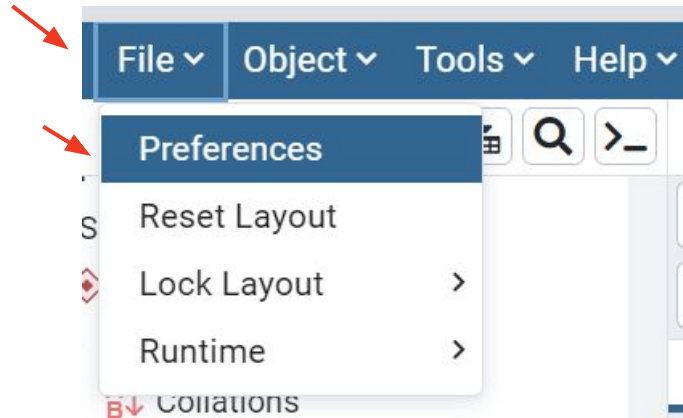


The screenshot shows a SQL IDE interface. The toolbar at the top includes icons for file operations, editing, filtering, and execution. The 'Execute/Refresh' button (a play icon) is highlighted with a tooltip that says 'Execute/Refresh' and 'F5'. Below the toolbar, there are two tabs: 'Query' and 'Query History'. The 'Query' tab is active, showing a script with two queries. The first query is highlighted in blue. The second query is not highlighted.

```
1 -- 1. How many rows are in the st_jobs table?
2
3 SELECT COUNT(*)
4 FROM data_analyst_jobs;
5
6 --Answer: 1793
7
8 -- 2. Write a query to look at just the first 10 rows. What company is associated with the job
   posting on the 10th row?
```

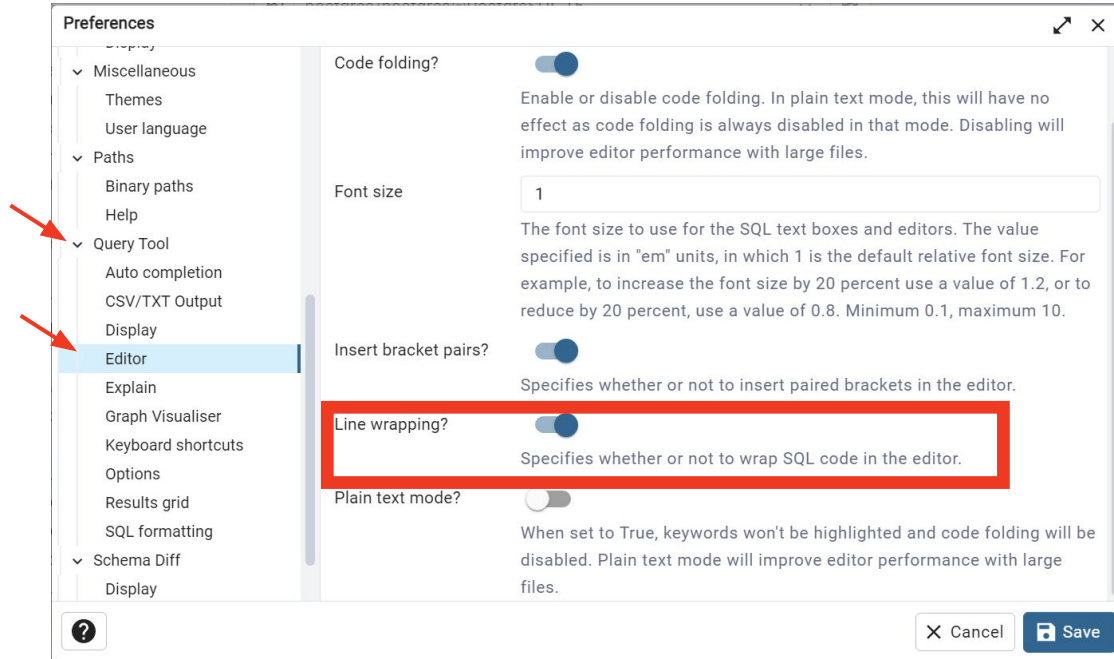
Text Wrapping

- Automatically making your comments and queries wrap text to the next line instead of creating a scroll bar to have to scroll through will make working in pgAdmin a lot easier.
- To turn this setting on, go to **File** → **Preferences**



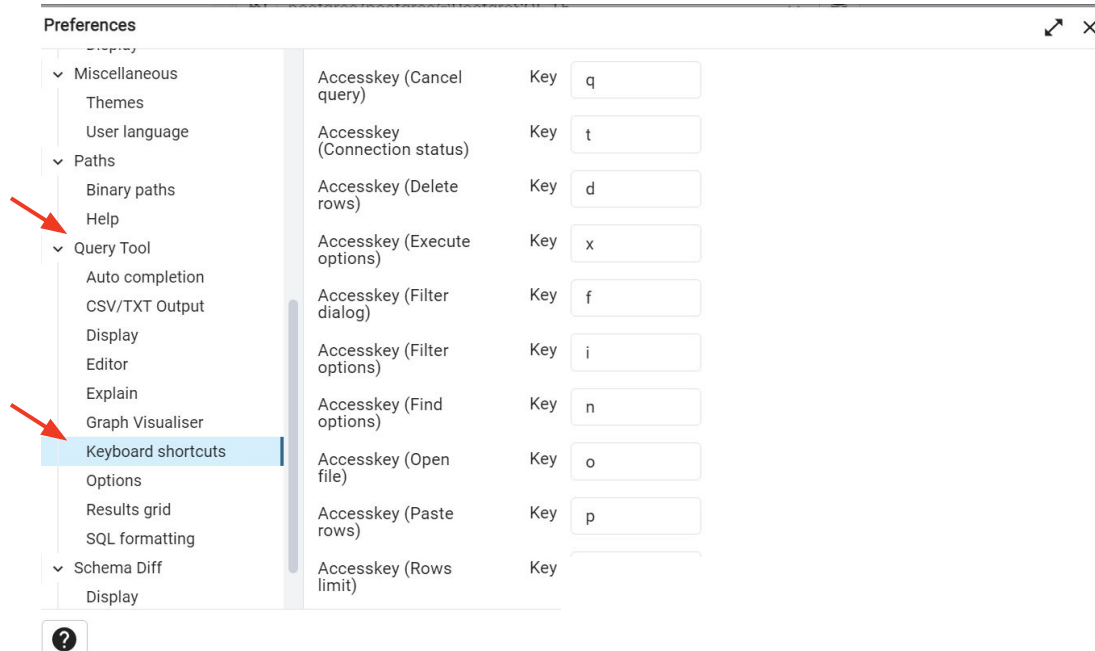
Text Wrapping

- Then select **Editor** under **Query Tool** on the left hand menu and turn on the **Line wrapping?** option.



Shortcuts

From the **File** → **Preferences** menu, select **Keyboard shortcuts** on the left hand menu under **Query Tool**



Shortcuts

- From this page, you can see the shortcut keys available and change any to make use easier for you.
- One example that many students choose to change is the **Execute query** shortcut. **F5** is located in an inconvenient spot on the keyboard for frequent use. Changing this shortcut to **Shift + Enter** will make SQL query execution more in line with Jupyter Notebook execution when we get to Python.

Execute query

Key

Enter



Shift



Ctrl



Alt/Option

Shortcuts

Another helpful shortcut that is not listed on the shortcuts page is **ctrl + /** which allows you to comment out everything in a selection at once.

```
2. Write a query to look at just the first 10 rows. What company is associated with the job posting on the 10th row?
```

```
-- 2. Write a query to look at just the first 10 rows. What company is associated with the job posting on the 10th row?
```