

Last edited by  [Isabel F Freitas](#) 1 year ago

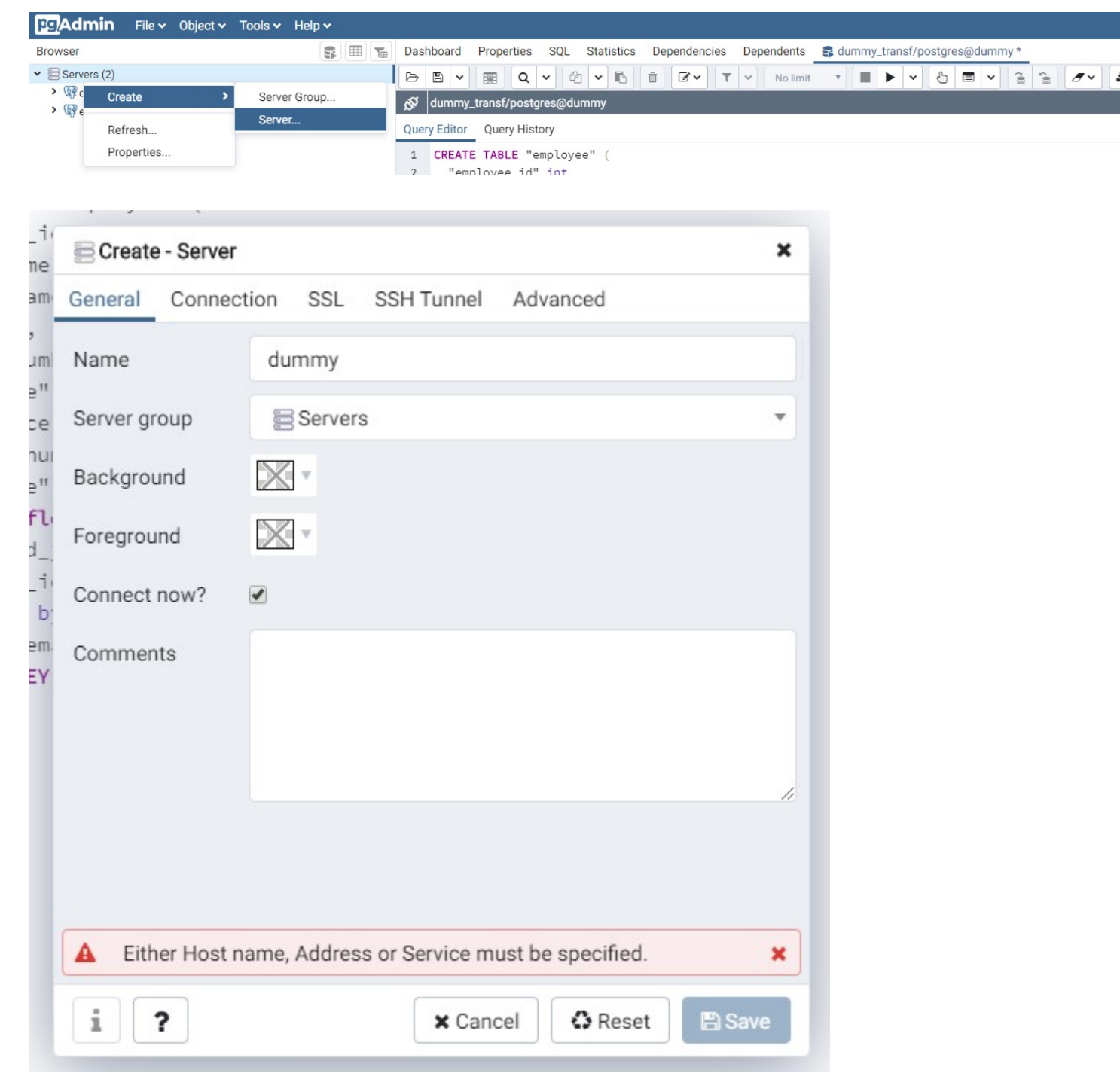
pgAdmin4 and postgres

Useful commands

- Find out port: `netstat -nlp|grep postgres` or `netstat -nlp|grep postgresql`
- Find out status: `service postgresql status`
- Check for multiple instances of postgresql: `pg_lsclusters`

How to create a new server instance (e.g. dummy)

First Open pgAdmin4: <http://localhost/pgadmin4/browser/>



Create - Server

General

Connection

SSL

SSH Tunnel

Advanced

Host name/address

localhost

Port

5433

Maintenance database

postgres

Username

postgres

Password

.....

Save password?

☐

Role

Service

i

?

Cancel

Reset

Save

Create a database within that server

Servers (2)

> dummy

> el

Create

Refresh...

Disconnect Server

Remove Server

Reload Configuration

Add Named Restore Point...

Backup Globals...

Backup Server...

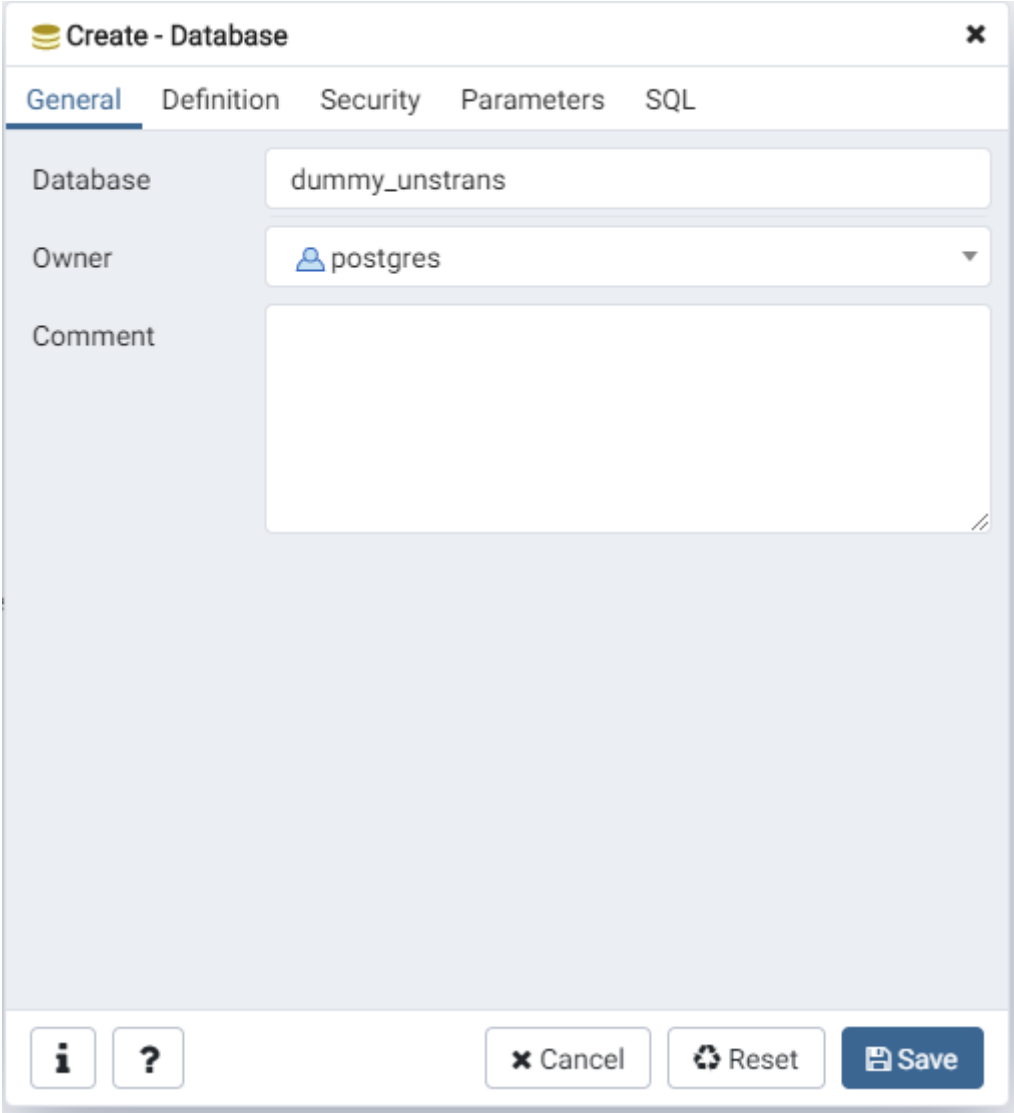
Properties...

Server...

Database...

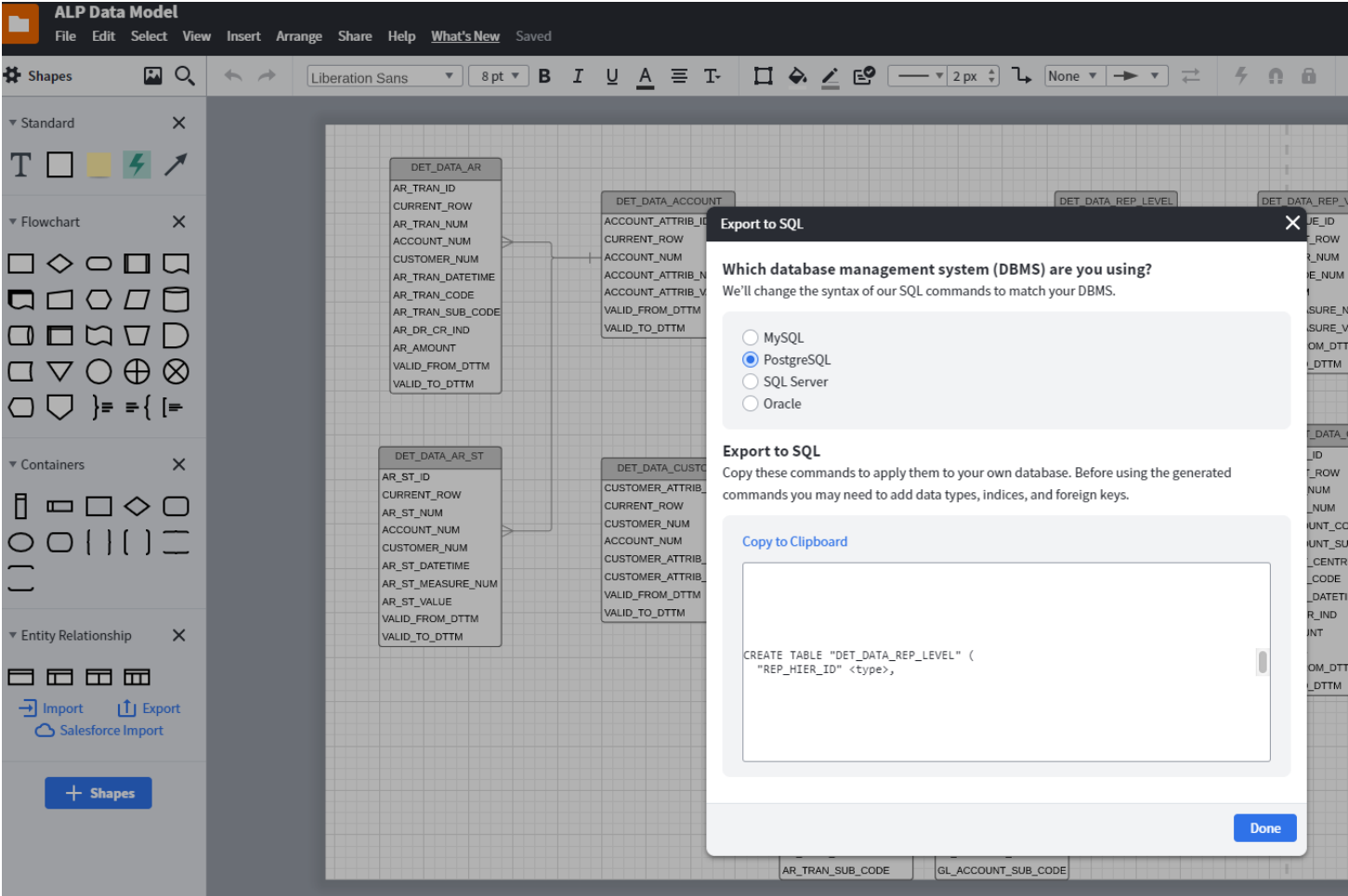
Login/Group Role...

Tablespace...



Create a schema using the LucidChart SQL from the Entity Relationship Diagram (ERD)

Lucidchart:



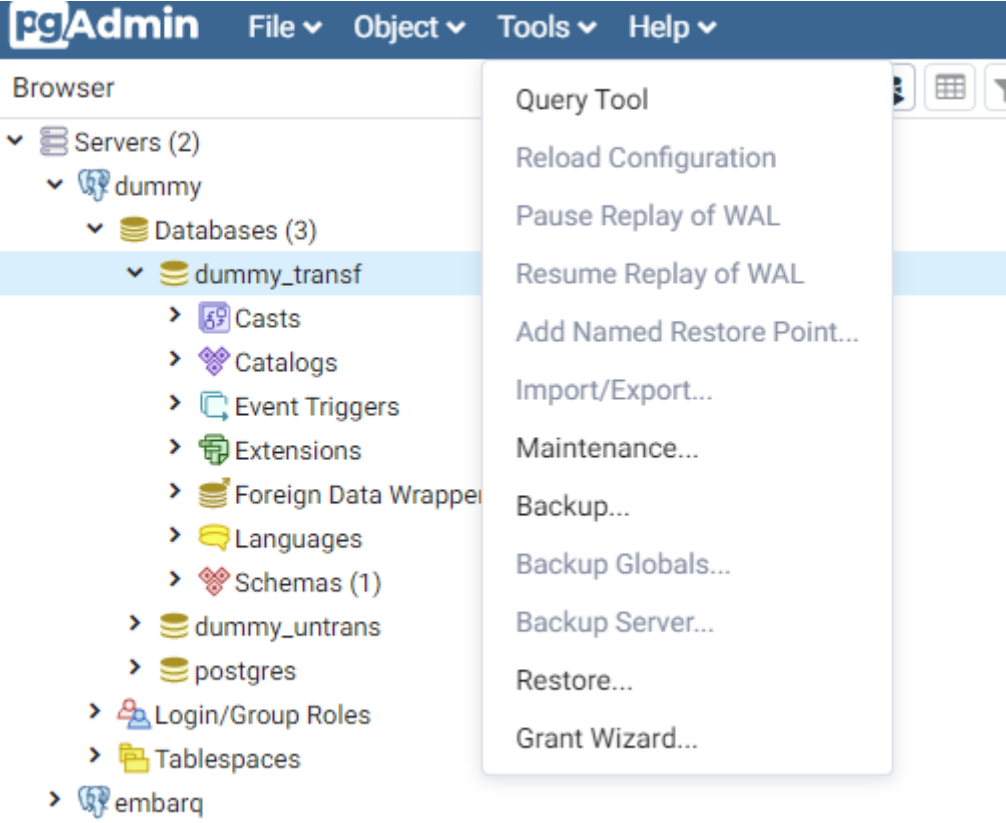
command: Copy the SQL

pgAdmin4:

select the server of interest

select the database of interest

open the query tool



Paste the SQL code from LucidChart

And execute

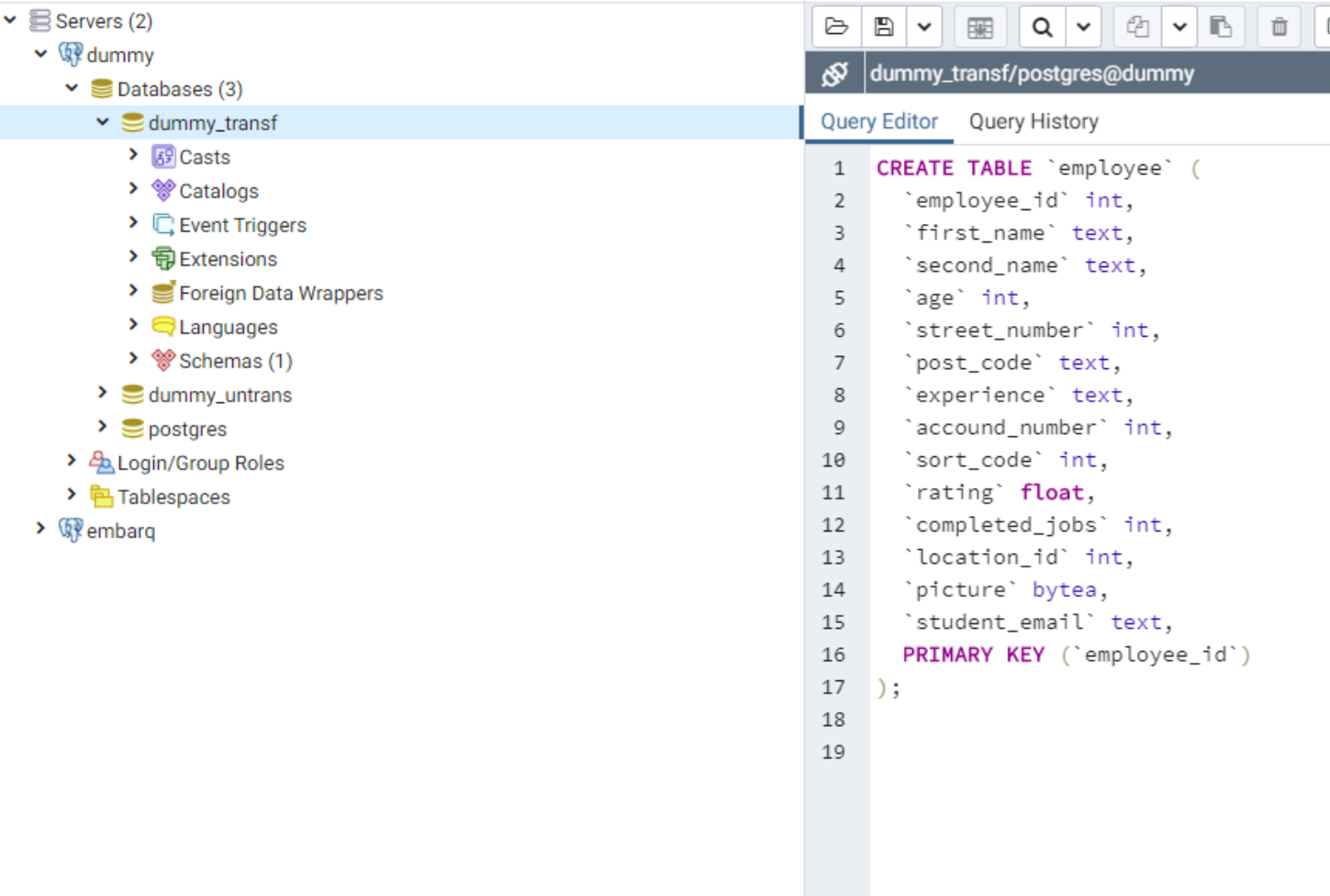


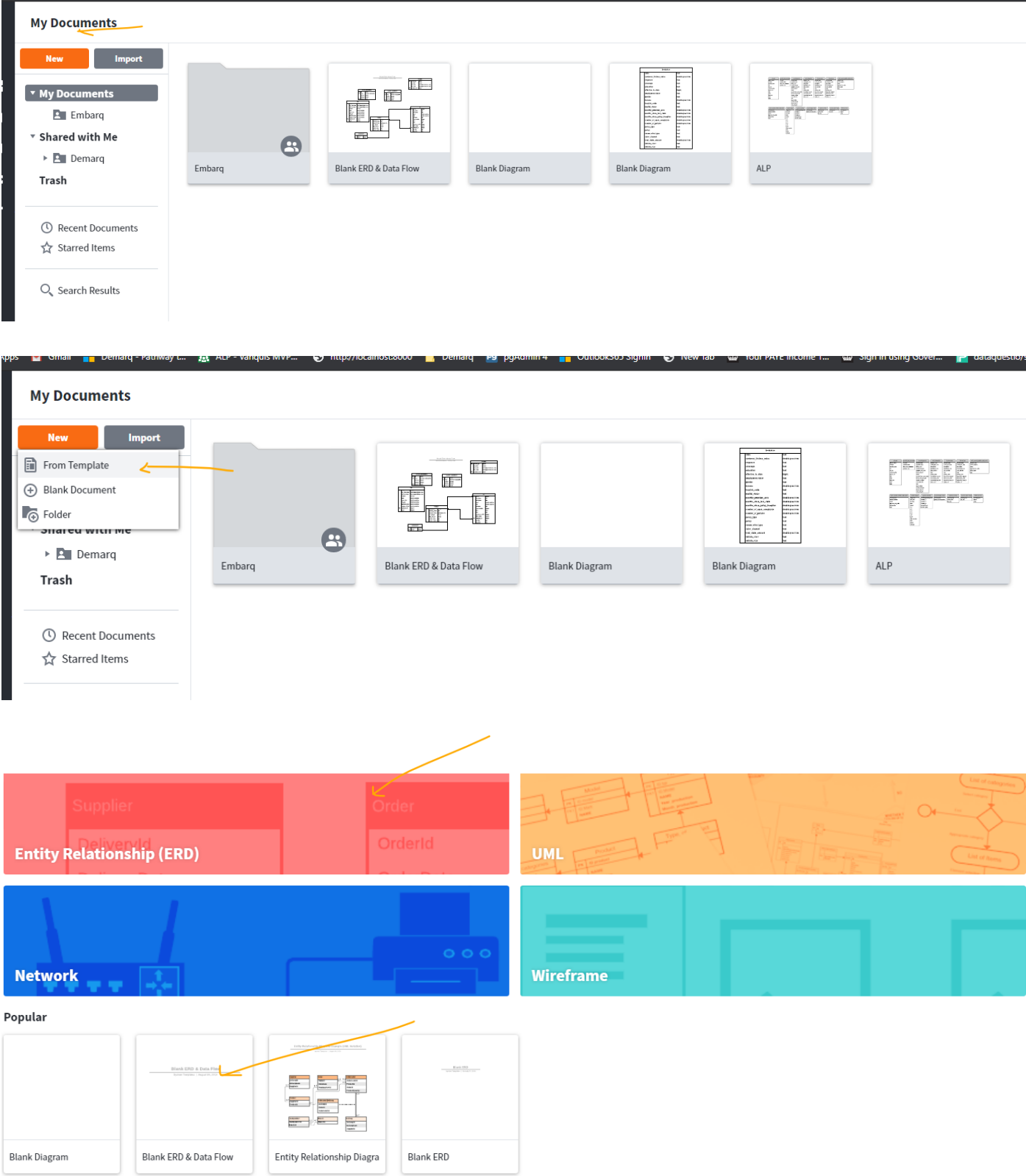
Table will appear



Create a ERD schema using the LucidChart from the Database SQL:

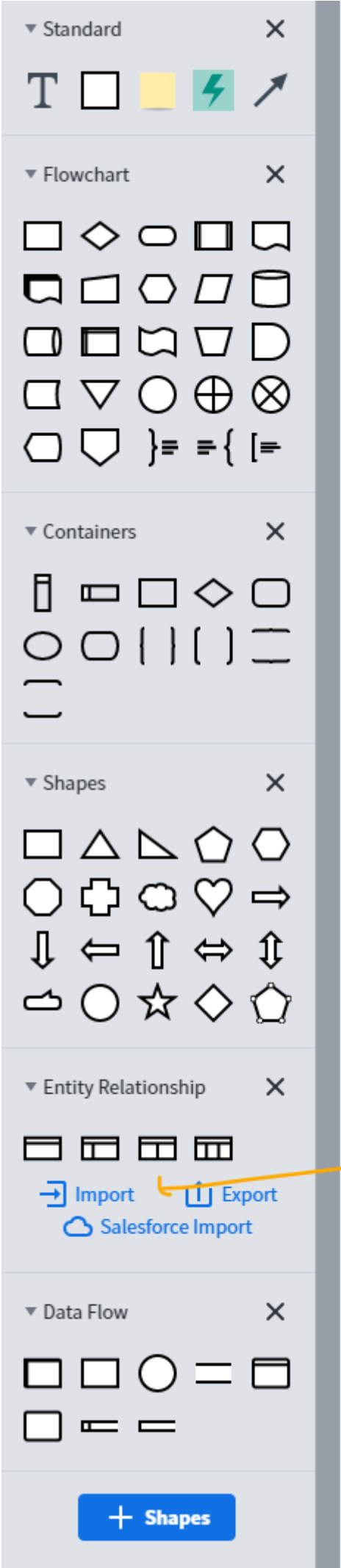
<https://www.youtube.com/watch?v=yFjeJnV42Lg>

1. Go to LucidCharts and create a new ERD blank document



2. In the ERD documnet:

- Under Entity Relationship select **import**:



- Select the **PostgreSQL** option

Import from SQL

Query

Upload

Which database management system (DBMS) are you importing?

☐ MySQL

☐ Oracle

☒ PostgreSQL

☐ SQL Server

Run this query on your database.

This query will create the structure we need to generate diagram shapes from your database schema. In the next step, you'll upload the results this query creates.

Copy to Clipboard

```
SET enable_nestloop=0;SELECT 'postgresql' AS
dbms,t.table_catalog,t.table_schema,t.table_name,c.column_name,c.ordinal_
position,c.data_type,c.character_maximum_length,n.constraint_type,k2.tabl
e_schema,k2.table_name,k2.column_name FROM information_schema.tables t
NATURAL LEFT JOIN information_schema.columns c LEFT
JOIN(information_schema.key_column_usage k NATURAL JOIN
information_schema.table_constraints n NATURAL LEFT JOIN
information_schema.referential_constraints r)ON
c.table_catalog=k.table_catalog AND c.table_schema=k.table_schema AND
c.table_name=k.table_name AND c.column_name=k.column_name LEFT JOIN
information_schema.key_column_usage k2 ON
k.position_in_unique_constraint=k2.ordinal_position AND
r.unique_constraint_catalog=k2.constraint_catalog AND
r.unique_constraint_schema=k2.constraint_schema AND
r.unique_constraint_name=k2.constraint_name WHERE t.TABLE_TYPE='BASE
TABLE'
```

?

Need help? Try our [ERD Import and Export Tutorial](#) or [contact us](#).

Cancel

Next

- Copy the code under the **Run this query on your database**

import from SQL

Query

Upload

Which database management system (DBMS) are you importing?

☐ MySQL

☐ Oracle

☒ PostgreSQL

☐ SQL Server

Run this query on your database.

This query will create the structure we need to generate diagram shapes from your database schema. In the next step, you'll upload the results this query creates.

Copy to Clipboard

SET enable_nestloop=0;SELECT 'postgresql' AS dbms,t.table_catalog,t.table_schema,t.table_name,c.column_name,c.ordinal_position,c.data_type,c.character_maximum_length,n.constraint_type,k2.table_schema,k2.table_name,k2.column_name FROM information_schema.tables t NATURAL LEFT JOIN information_schema.columns c LEFT JOIN(information_schema.key_column_usage k NATURAL JOIN information_schema.table_constraints n NATURAL LEFT JOIN information_schema.referential_constraints r)ON c.table_catalog=k.table_catalog AND c.table_schema=k.table_schema AND c.table_name=k.table_name AND c.column_name=k.column_name LEFT JOIN information_schema.key_column_usage k2 ON k.position_in_unique_constraint=k2.ordinal_position AND r.unique_constraint_catalog=k2.constraint_catalog AND r.unique_constraint_schema=k2.constraint_schema AND r.unique_constraint_name=k2.constraint_name WHERE t.TABLE_TYPE='BASE

Need help? Try our [ERD Import and Export Tutorial](#) or [contact us.](#)

Cancel

Next

3. On pgAdmin4:

- select the server of interest
- select the database of interest
- open the query tool
- Paste the code from Lucidcharts into the query tool of the database

A screenshot of the pgAdmin4 web interface. On the left, a tree view shows the database structure, with 'embarq' selected under 'Databases (3)'. The main panel shows the 'Query Editor' with a SQL query pasted in. An orange arrow points from the query text in the previous block to the query editor area.

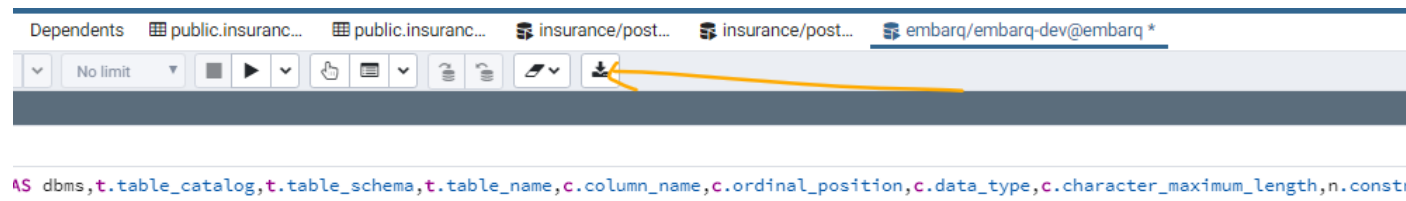
- Run the query

A screenshot of the pgAdmin4 web interface, focusing on the query tool's toolbar. An orange arrow points to the 'Execute' button (a green play icon). Below the toolbar, the beginning of the SQL query is visible: 'ostgresql' AS dbms,t.table_catalog,t.table_schema,t.table_name,c.column_name,c.ordinal_position,c.'

- Download the output (.csv)

https://gitlab.demarq.com/iff11/python/-/wikis/pgAdmin4-and-postgres

8/13



4. On Lucidcharts:

- Click Next

Query Upload

Which database management system (DBMS) are you importing?

☐ MySQL
☐ Oracle
☒ PostgreSQL
☐ SQL Server

Run this query on your database.

This query will create the structure we need to generate diagram shapes from your database schema. In the next step, you'll upload the results this query creates.

[Copy to Clipboard](#)

```
SET enable_nestloop=0;SELECT 'postgresql' AS
dbms,t.table_catalog,t.table_schema,t.table_name,c.column_name,c.ordinal_
position,c.data_type,c.character_maximum_length,n.constraint_type,k2.tabl
e_schema,k2.table_name,k2.column_name FROM information_schema.tables t
NATURAL LEFT JOIN information_schema.columns c LEFT
JOIN(information_schema.key_column_usage k NATURAL JOIN
information_schema.table_constraints n NATURAL LEFT JOIN
information_schema.referential_constraints r)ON
c.table_catalog=k.table_catalog AND c.table_schema=k.table_schema AND
c.table_name=k.table_name AND c.column_name=k.column_name LEFT JOIN
information_schema.key_column_usage k2 ON
k.position_in_unique_constraint=k2.ordinal_position AND
r.unique_constraint_catalog=k2.constraint_catalog AND
r.unique_constraint_schema=k2.constraint_schema AND
r.unique_constraint_name=k2.constraint_name WHERE t.TABLE_TYPE='BASE
```

Need help? Try our [ERD Import and Export Tutorial](#) or [contact us](#).

Cancel Next

- Chose the **Upload a file from your computer** option and **choose the file**

Import from SQL

[Query](#)[Upload](#)

Upload the results generated by your database query.
We support tab-, semicolon-, and comma-separated results as a .CSV, .TSV, or .TXT file.

☒ **Upload a file from your computer**

Choose File

No file chosen

☐ **Paste the output as plain text**

?

Need help? Try our [ERD Import and Export Tutorial](#) or [contact us](#).

Back

Import

Import from SQL

[Query](#)[Upload](#)

Upload the results generated by your database query.
We support tab-, semicolon-, and comma-separated results as a .CSV, .TSV, or .TXT file.

☒ **Upload a file from your computer**

Choose File

No file chosen

☐ **Paste the output as plain text**

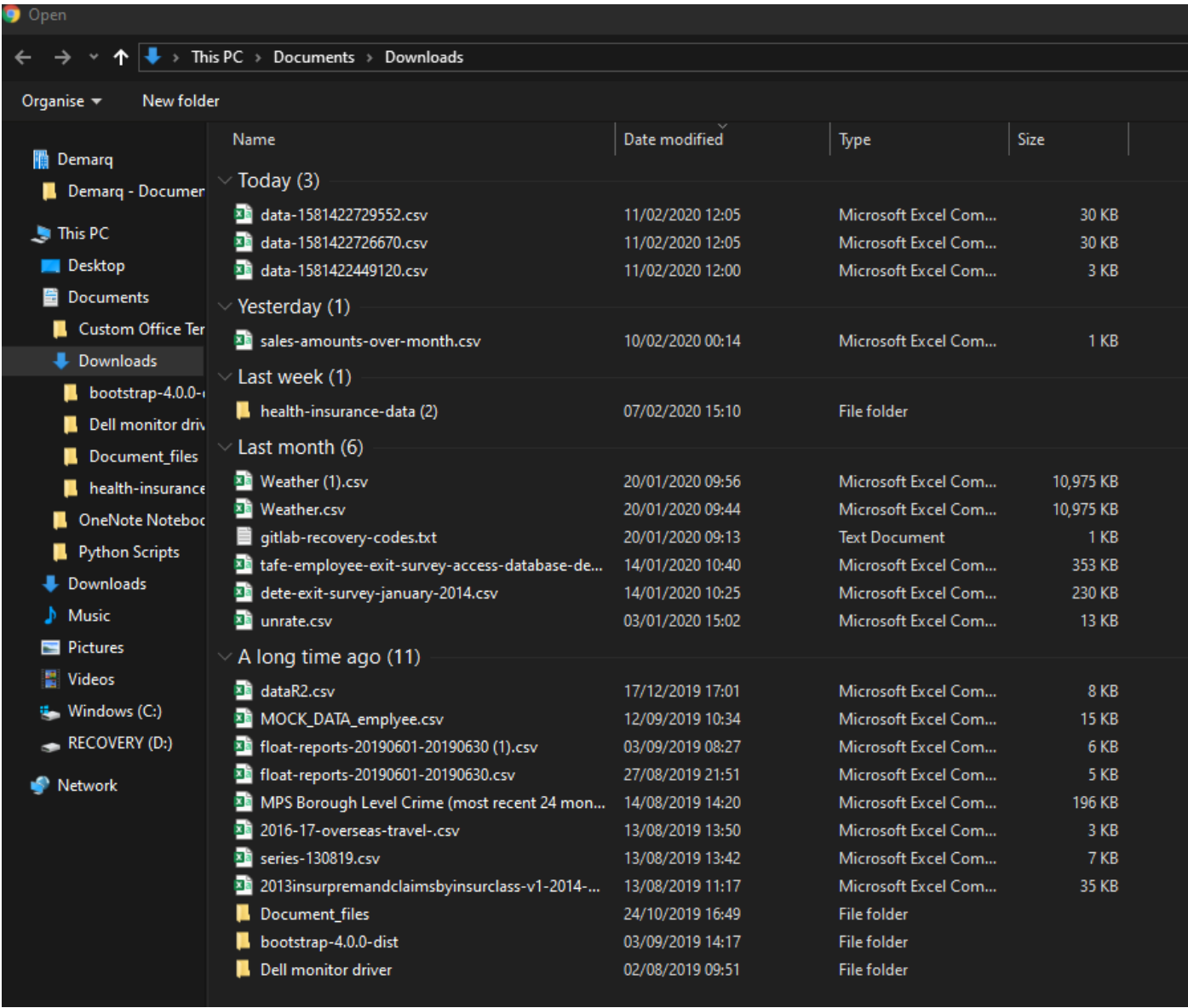
?

Need help? Try our [ERD Import and Export Tutorial](#) or [contact us](#).

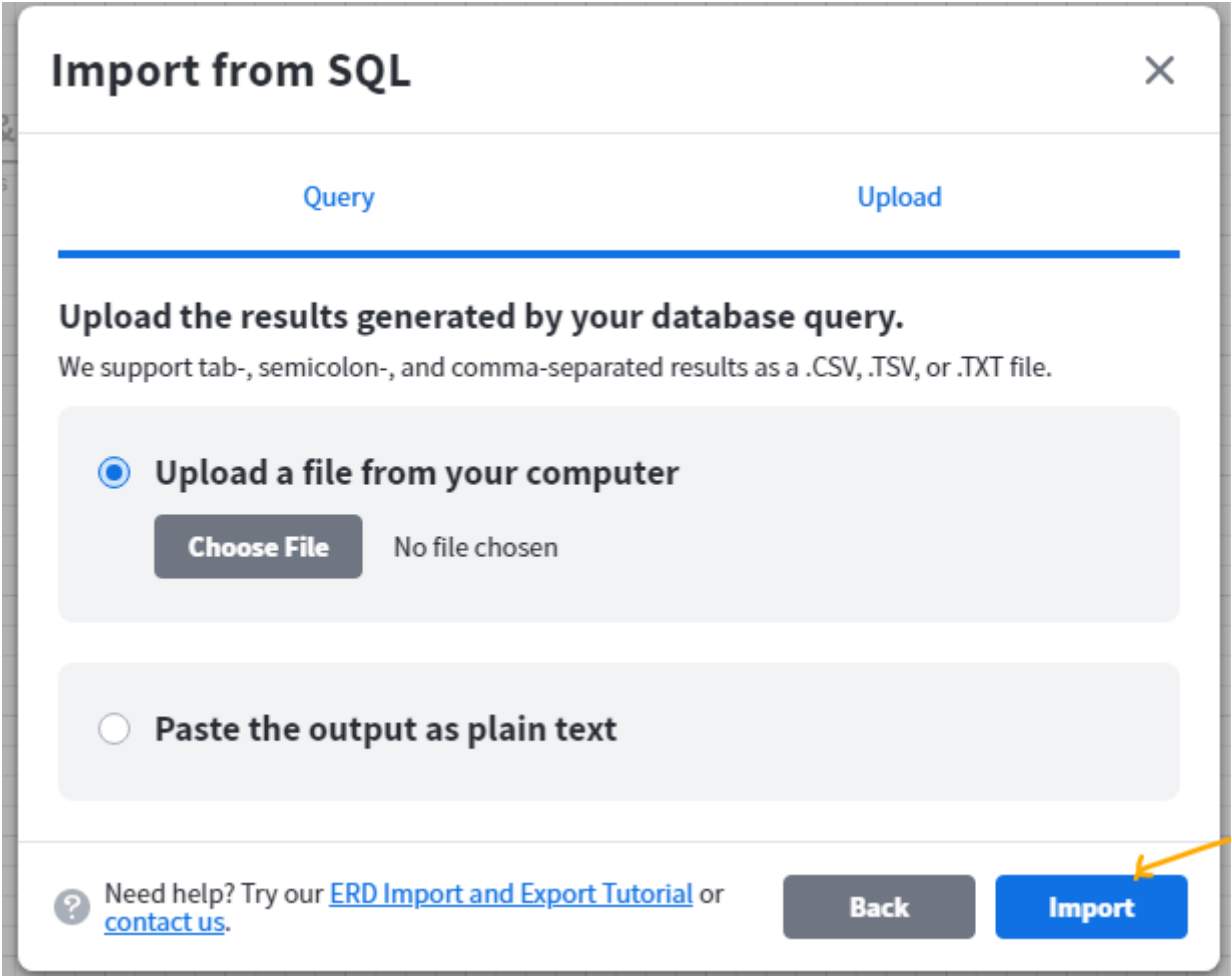
Back

Import

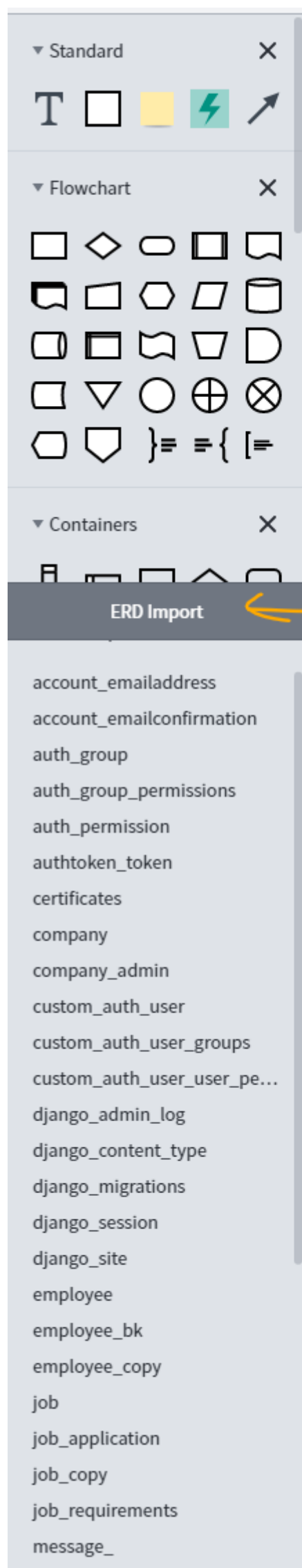
- Chose the file



- Finally import it



- Under **ERD Import** all the columns should be visible



- Drag the entities out and the entity relationships should be kept

