

PSQL Shell

Let's open up the **PSQL shell** and enter in these commands that we just created here in our schema.sql file. This **PSQL shell** should have been installed automatically when you installed **PSQL**. If not simply go to the **PSQL** website to download and install it again. If you are first logging in to the **PSQL shell** you will be prompted to set the server, database name, port, username and password. Leave the port to the **default 5432** and setup the rest of the credentials to anything you want.

So now you should just be seeing postgres# on the terminal or whatever you set the database name as. This means we are ready to start entering in **SQL** commands. Instead of using the default database let's create a new one with the command CREATE DATABASE database1 and then connect to it with \c database1. If done correctly you should see the database#.

If you want a list of all the commands you can type help or \? in the **PSQL shell**. Always remember to end your SQL queries with a ; which is one of the most common errors when working with SQL.

From here we can just copy and paste our commands from the schema.sql file.

To see a list of our tables we use the \dt command and you should be seeing this in the terminal.

And we have successfully set up the database!

Now we need to actually connect this **database** to our **server**. Doing this is extremely simple. We can do this by making use of

the pg library. Install the pg library if you have not already done so and make sure you are in the Server directory we do not want to install this library in our React app.

Create a separate file called db.js in the **main directory** and set it up as follows:

```
const { Pool } = require('pg')
```

```
const pool = new Pool({  
  user: 'postgres',  
  host: 'localhost',  
  database: 'postgres',  
  password: '',  
  port: 5432  
})
```

```
module.exports = pool
```

These are going to be the same credentials you set when setting up the **PSQL shell**.

And that's it we have setup our database with use with our server. We can now begin making queries to it from our express server.