

## PostgreSQL

Dicas e macetes:  
Instalação no Linux:

```
sudo apt-get install -y postgresql-12 postgresql-contrib-12
```

Criando o banco

```
sudo su - postgres
psql -U postgres -c "CREATE ROLE myuser ENCRYPTED PASSWORD
'mypass' LOGIN;"
psql -U postgres -c "CREATE DATABASE mydb OWNER myuser;"
# Você pode se conectar no novo banco pra não precisar sair.
\c mydb
```

```
CREATE SCHEMA myschema;
```

```
\q # sair
```

```
psql mydb
```

### Script para criar o banco direto

```
createdb -U postgres mydb
# ou
sudo -u postgres -i createdb mydb
```

Assim você pode criar um script, em Shell ou Python para rodar automaticamente.

Exemplo:

```
# create_database.py
import subprocess

subprocess.call("createdb -U postgres mydb", shell=True)
```

Veja um exemplo mais completo em [python\\_commands.py](#).

Se quiser você também pode criar as tabelas diretamente pelo terminal.

*Não se esqueça do mydb no final do comando.*

```
psql -U postgres -c "CREATE TABLE produtos (id SERIAL PRIMARY KEY,
produto VARCHAR(100));" mydb
```

### **Cria usuário**

sudo su - postgres

### **Cria o banco de dados**

createdb clbit

### **Acessar o banco**

psql clbit

**O comando a seguir define direito de acesso ao novo usuário.**

GRANT ALL PRIVILEGES ON DATABASE clbit TO postgres;

### **Sair do psql**

\q

createdb mydb

dropdb mydb

### **Acesso ao banco**

psql mydb

mydb=> SELECT version();

ajuda

\h

sair

\q

\l

\c database\_name

\dt

\?

## Lendo os comandos de um arquivo específico

```
\i basics.sql
```

### Consultas

```
SELECT * FROM clima INNER JOIN cidades ON (clima.cidade =  
cidades.nome);
```

### Funções de agregação

Encontrar a maior temperatura mínima ocorrida em qualquer lugar.

```
SELECT max(temp_min) FROM clima;
```

ERRADO

```
SELECT cidade FROM clima WHERE temp_min = max(temp_min);
```

Para saber a cidade (ou cidades) onde esta temperatura ocorreu usamos uma **subconsulta**.

```
SELECT cidade FROM clima WHERE temp_min = (SELECT max(temp_min)  
FROM clima);
```