

# NappAcademy

## Sprint 4

prof. Orlando Saraiva Jr

- Instalação SGBD – DBeaver
- Hospedagem do SGBD em ambiente remoto (Heroku)
- Criação de um banco de dados no SGBD com inserção de registros
- Backup e Restauração do SGBD
- Conexão entre o SGBD e uma linguagem de programação (Python)

# Participantes

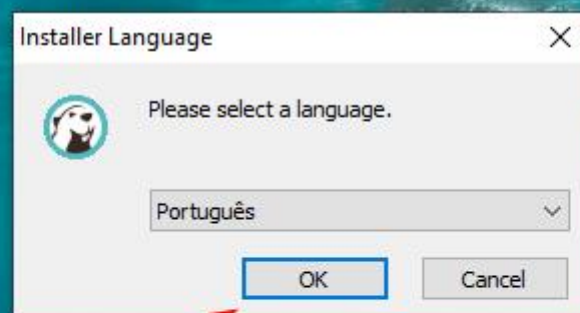
- Cauê Viel
- Rian Scavazza
- Matheus Santos
- André Conforti
- Ronaldo Montan

# **Instalação SGBD – DBeaver**



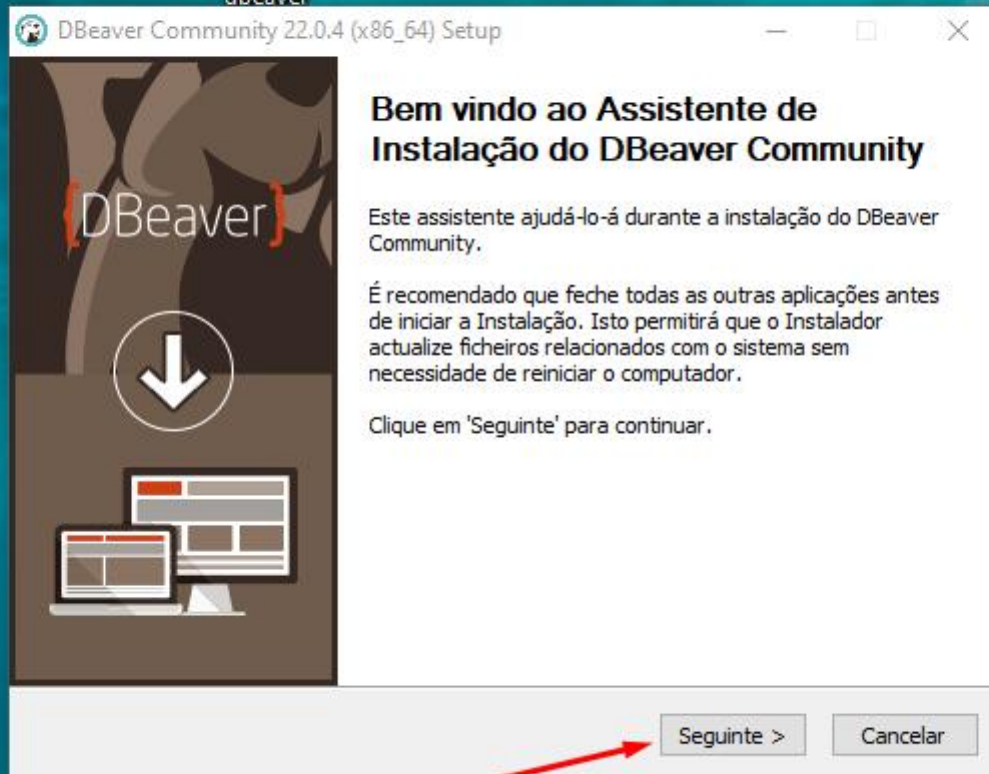
**Abrir**

- Executar como administrador
- Compartilhar com o Skype
- Solucionar problemas de compatibilidade
- Fixar em Iniciar
- Compartilhar
- Conceder acesso a >
- Adicionar para o arquivo...
- Adicionar para "dbeaver.rar"
- Comprimir e enviar por e-mail...
- Comprimir para "dbeaver.rar" e enviar por e-mail
- Fixar na barra de tarefas
- Restaurar versões anteriores
- Enviar para >
- Recortar
- Copiar
- Colar
- Criar atalho
- Excluir
- Renomear
- Propriedades



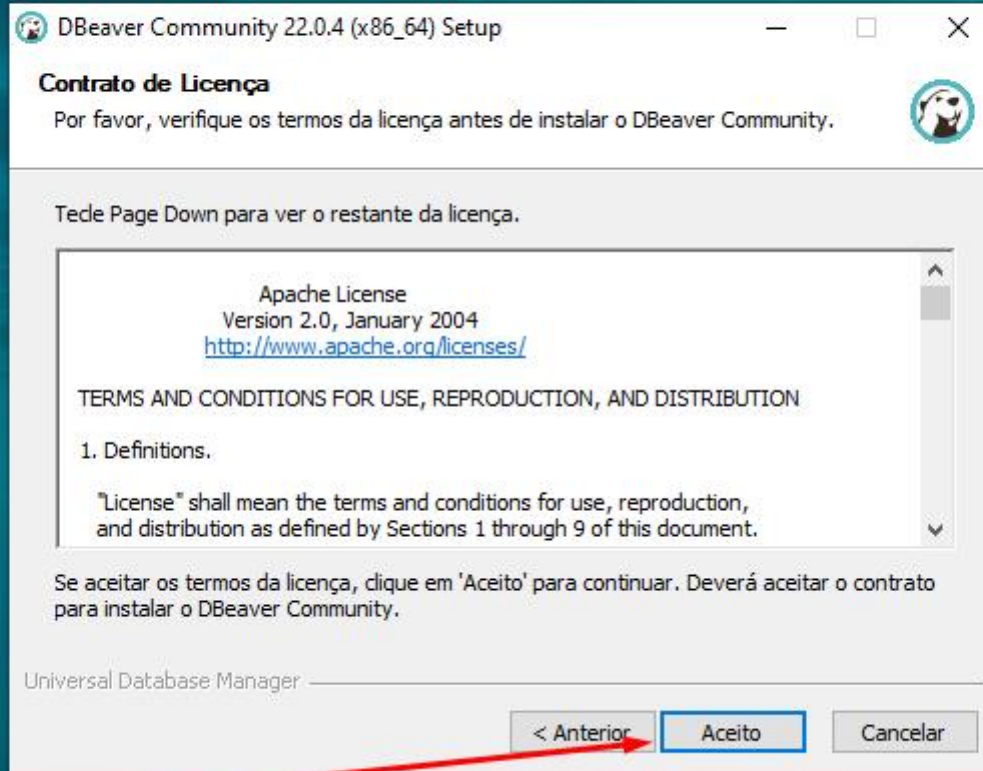


dbeaver





dbeaver







dbeaver

DBEaver Community 22.0.4 (x86\_64) Setup

**Choose Users**

Choose for which users to install DBEaver Community.

Select whether to install DBEaver Community for all users or for current user.

☐ For anyone who uses this computer (all users)

☒ For me (Napp)

Fresh install for current user.

Universal Database Manager

< Anterior Seguinte > Cancelar





dbeaver

DBEaver Community 22.0.4 (x86\_64) Setup

### Escolha de Componentes

Escolha quais as características do DBEaver Community que deseja instalar.

Marque os componentes que deseja instalar e desmarque os componentes que não deseja instalar. Clique em 'Seguinte' para continuar.

Escolha os componentes para instalar:

- ☒ **DBEaver Community**
- ☒ Include Java
- ☐ Reset Settings
- ☐ Associate .SQL files

Espaço necessário: 124.8MB

Descrição  
Posicione o rato sobre um componente para ver a sua descrição.

Universal Database Manager

< Anterior   Seguinte >   Cancelar



dbeaver

**DBEaver Community 22.0.4 (x86\_64) Setup**

**Escolha do Local da Instalação**  
Escolha a pasta na qual deseja instalar o DBEaver Community.

O DBEaver Community será instalado na seguinte pasta. Para instalar numa pasta diferente, clique em 'Procurar...' e escolha outra pasta. Clique em 'Seguinte' para continuar.

Pasta de Destino

Espaço necessário: 124.8MB  
Espaço disponível: 155.5GB

Universal Database Manager



dbeaver

**DBEaver Community 22.0.4 (x86\_64) Setup**

**Escolha uma Pasta do Menu Iniciar**

Escolha uma pasta do Menu Iniciar para os atalhos do programa.

Selecione uma pasta do Menu Iniciar em que deseja criar os atalhos do programa. Você pode também digitar um nome para criar uma nova pasta.

DBEaver Community

- Accessibility
- Accessories
- Administrative Tools
- Discord Inc
- FileZilla FTP Client
- K-Lite Codec Pack
- LibreOffice 7.2
- Lightshot
- Maintenance
- NVIDIA Corporation
- Oracle VM VirtualBox

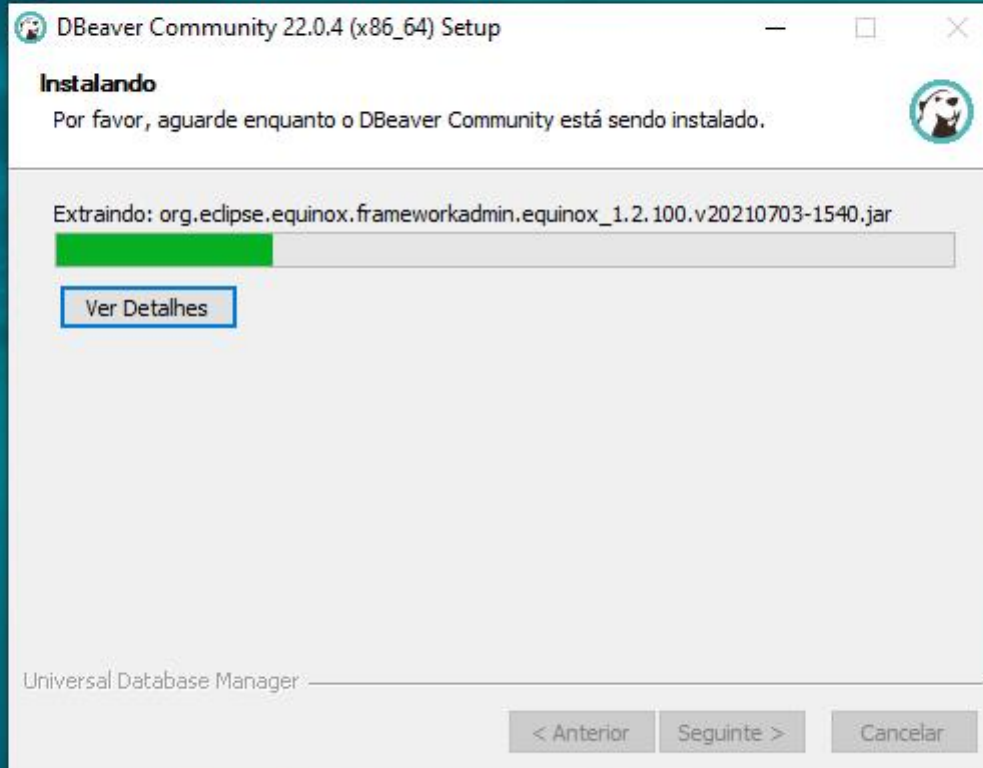
☐ Não criar atalhos

Universal Database Manager

< Anterior   **Instalar**   Cancelar

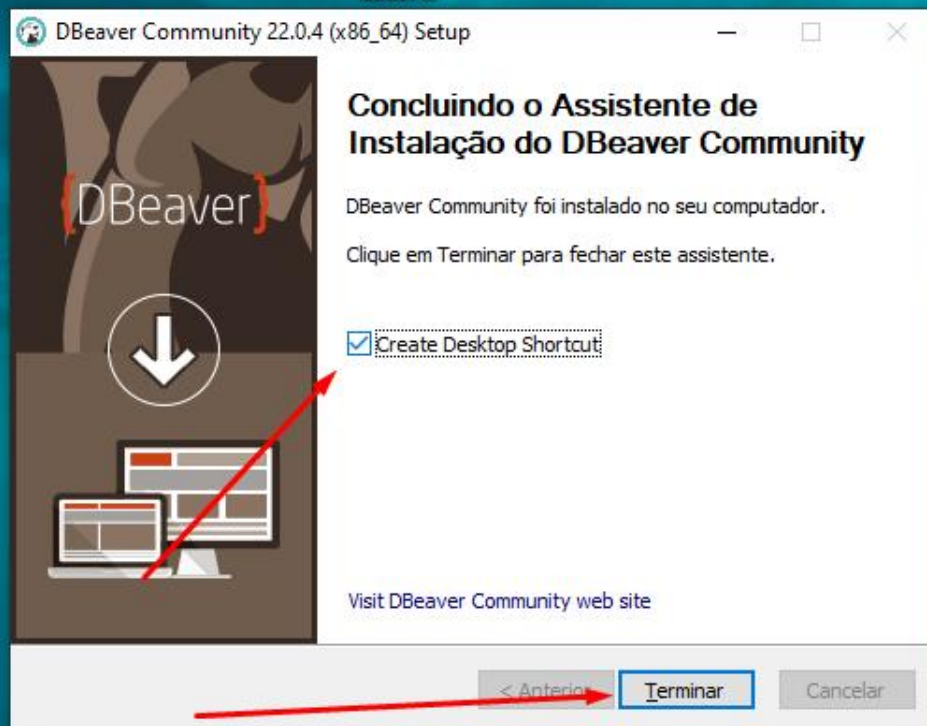


dbeaver





dbeaver

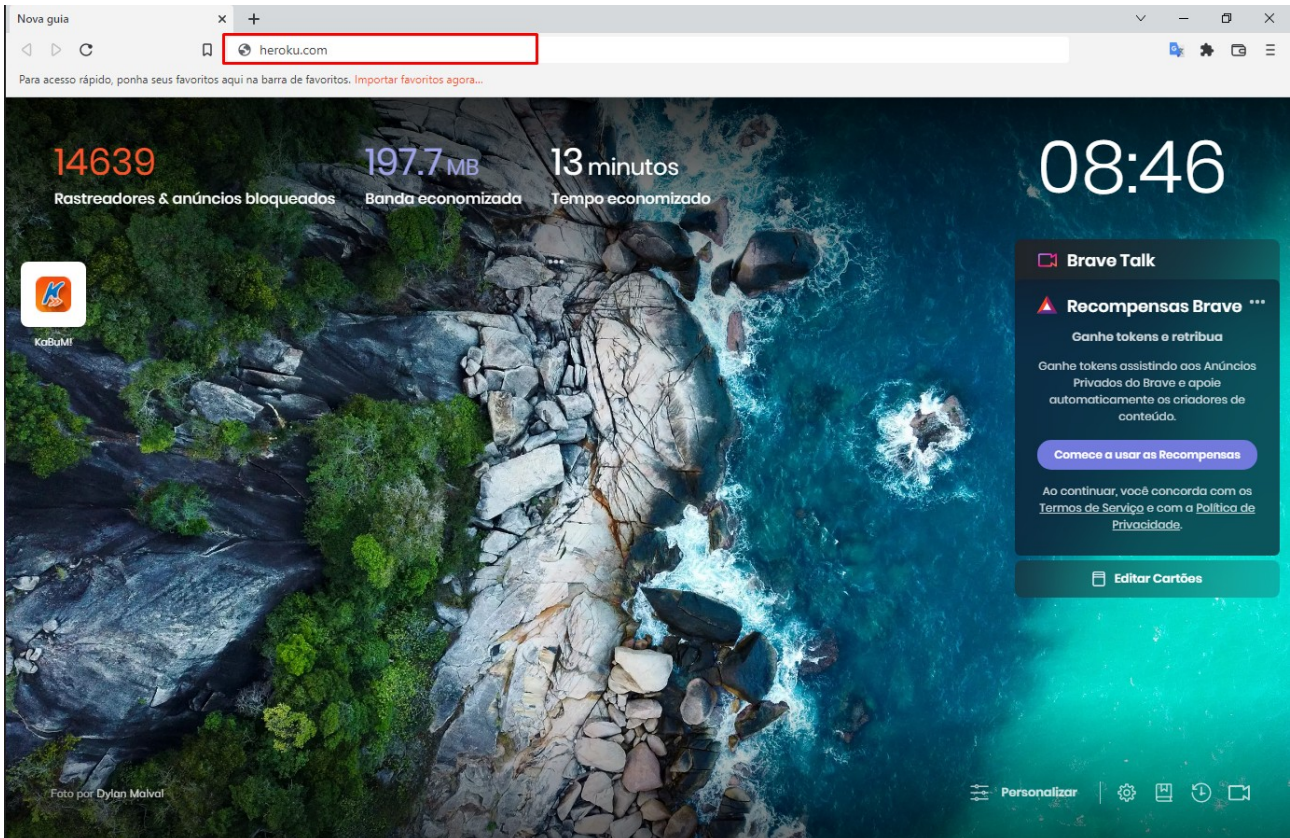


# **Hospedagem do SGBD em ambiente remoto (Heroku)**

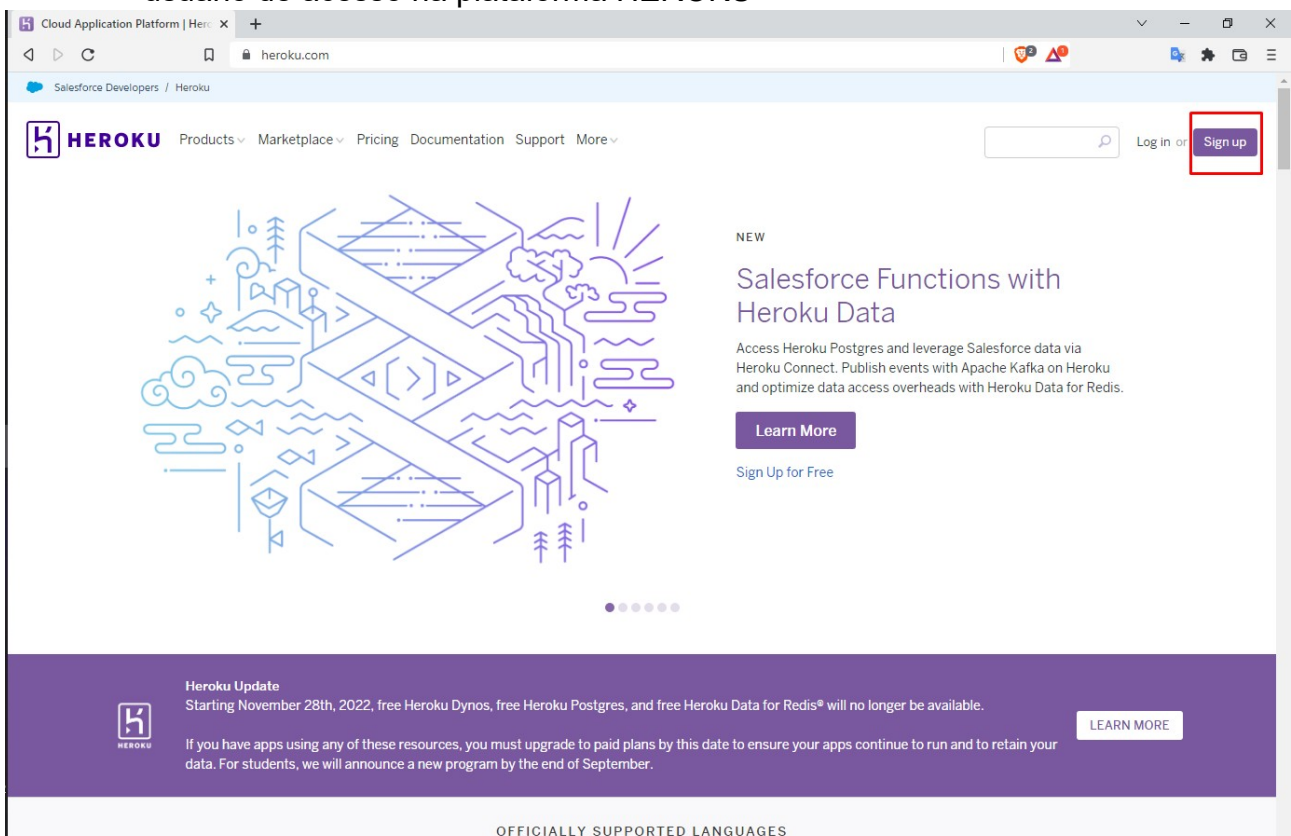


# Tutorial Hospedagem do Banco de Dados

- 1º Passo:
  - Acessar o navegador de sua preferência.
- 2º Passo:
  - Pesquisar a URL ("heroku.com") do site na qual hospedaremos nosso banco de dados:



- 3º Passo:
  - Para que seja possível hospedar o banco de dados, será necessário criar um usuário de acesso na plataforma HEROKU





- 4º Passo:
  - Definir as informações de usuário, conforme a preferência sua:

Sign up for free and experience Heroku today

**Free account**  
Create apps, connect databases and add-on services, and collaborate on your apps, for free.

**Your app platform**  
A platform for apps, with app management & instant scaling, for development and production.

**Deploy now**  
Go from code to running app in minutes. Deploy, scale, and deliver your app to the world.

First name \*  
Aluno

Last name \*  
Napp

Email address \*  
alunonapp@academy.com

Company name  
Napp

Role \*  
Student

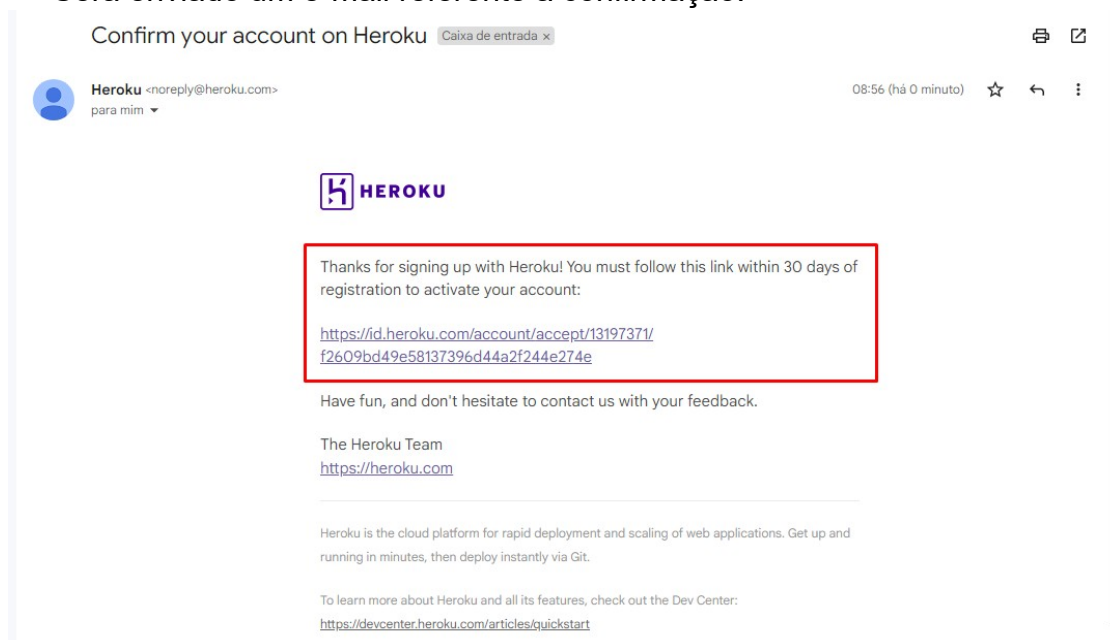
Country/Region \*  
Brazil

Primary development language \*  
I'm not a developer

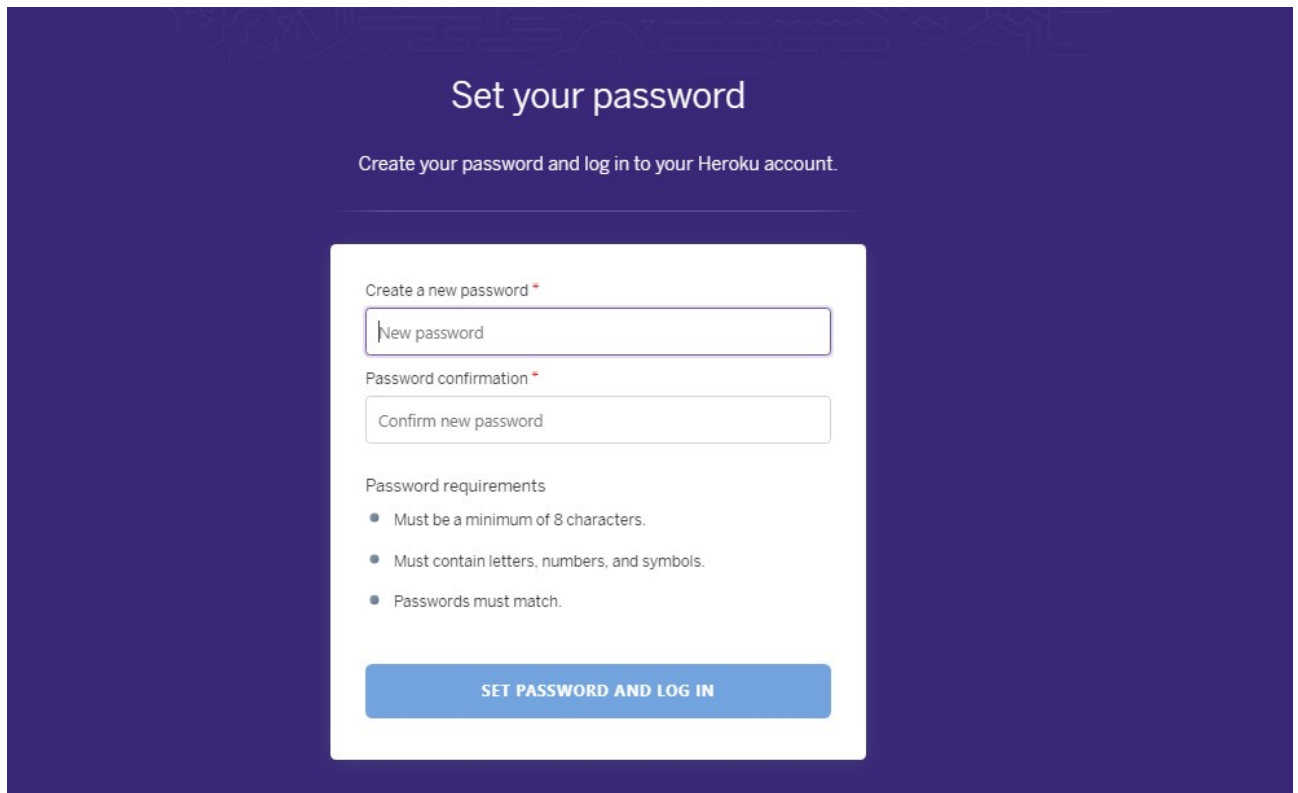
☒ I'm not a robot reCAPTCHA Privacy - Terms

**CREATE FREE ACCOUNT**

- 5º Passo:
  - Logo após a criação de usuário ser feita, será necessário confirmar o mesmo. Será enviado um e-mail referente a confirmação.



- 6º Passo:
  - Agora será necessário definir uma senha para a conta. E aceitar os termos de serviços do site.



Set your password

Create your password and log in to your Heroku account.

Create a new password \*

New password

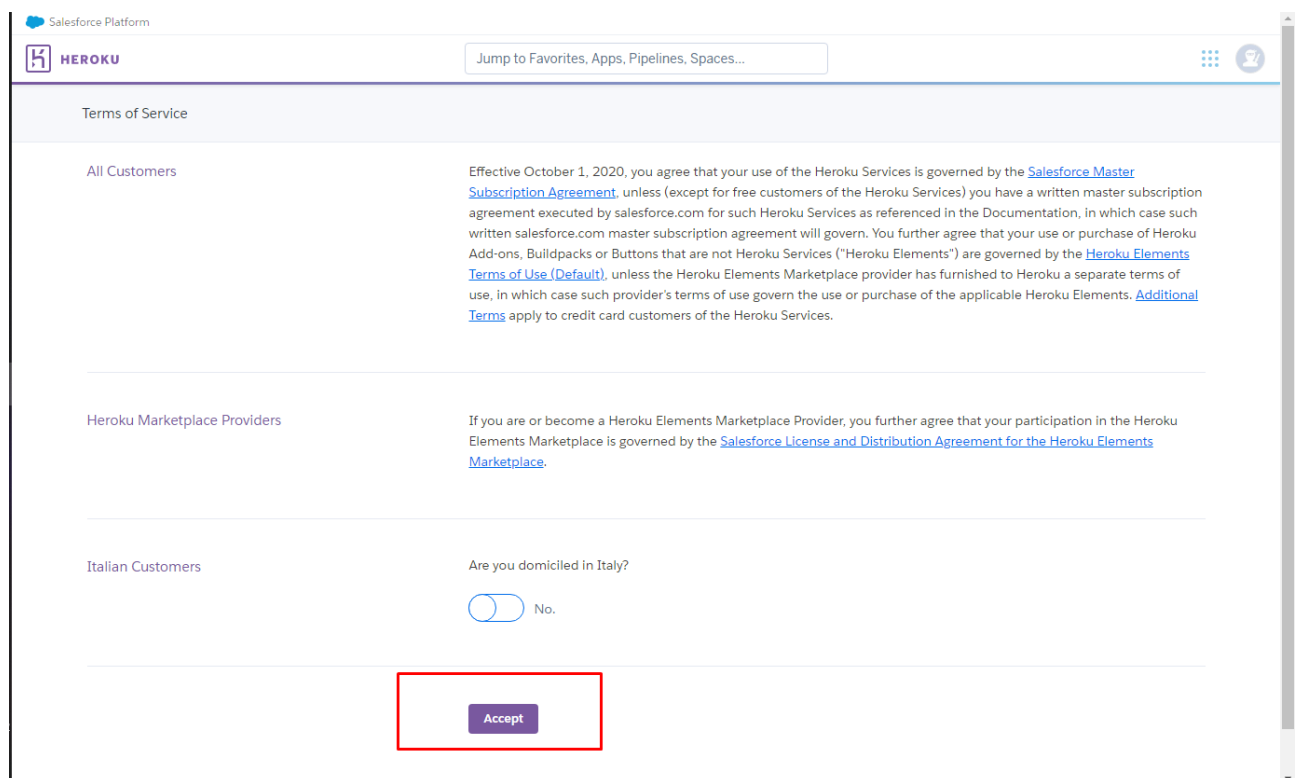
Password confirmation \*

Confirm new password

Password requirements

- Must be a minimum of 8 characters.
- Must contain letters, numbers, and symbols.
- Passwords must match.

SET PASSWORD AND LOG IN



Salesforce Platform

HEROKU

Jump to Favorites, Apps, Pipelines, Spaces...

Terms of Service

All Customers

Effective October 1, 2020, you agree that your use of the Heroku Services is governed by the [Salesforce Master Subscription Agreement](#), unless (except for free customers of the Heroku Services) you have a written master subscription agreement executed by salesforce.com for such Heroku Services as referenced in the Documentation, in which case such written salesforce.com master subscription agreement will govern. You further agree that your use or purchase of Heroku Add-ons, Buildpacks or Buttons that are not Heroku Services ("Heroku Elements") are governed by the [Heroku Elements Terms of Use \(Default\)](#), unless the Heroku Elements Marketplace provider has furnished to Heroku a separate terms of use, in which case such provider's terms of use govern the use or purchase of the applicable Heroku Elements. [Additional Terms](#) apply to credit card customers of the Heroku Services.

Heroku Marketplace Providers

If you are or become a Heroku Elements Marketplace Provider, you further agree that your participation in the Heroku Elements Marketplace is governed by the [Salesforce License and Distribution Agreement for the Heroku Elements Marketplace](#).

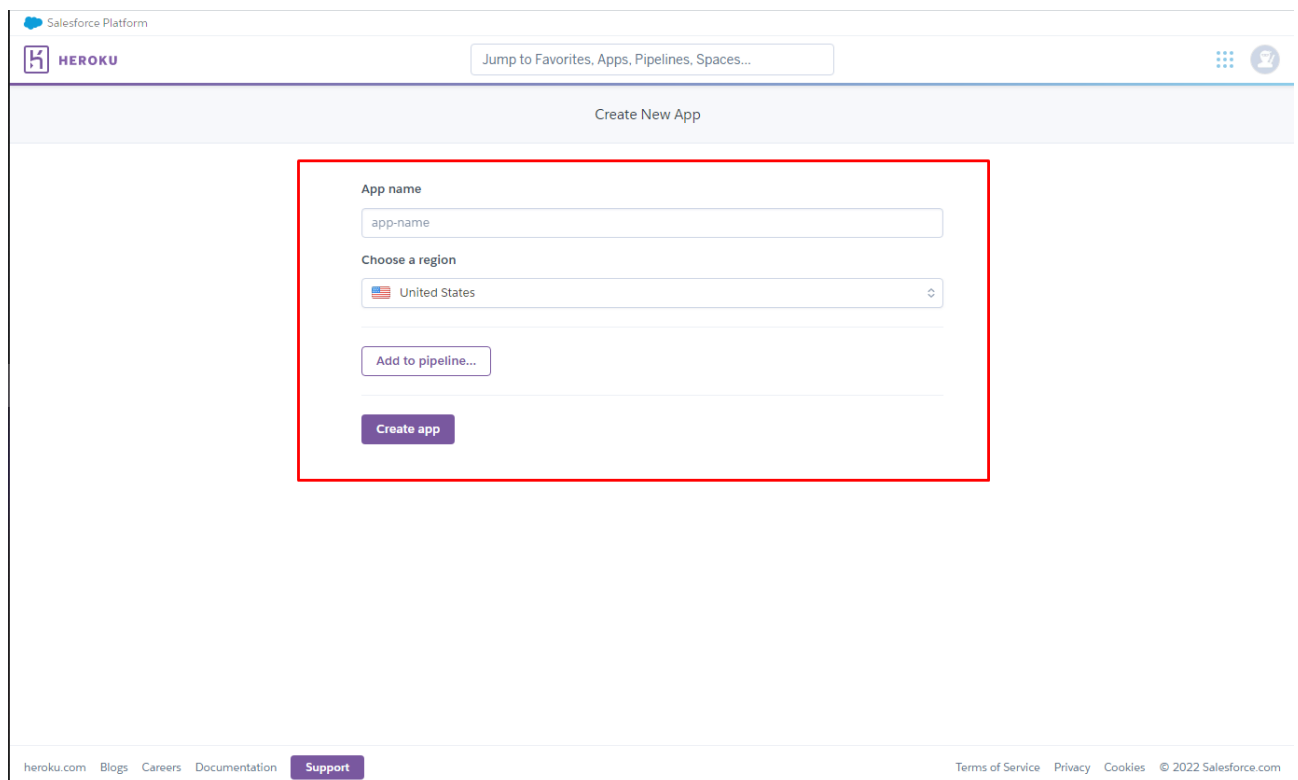
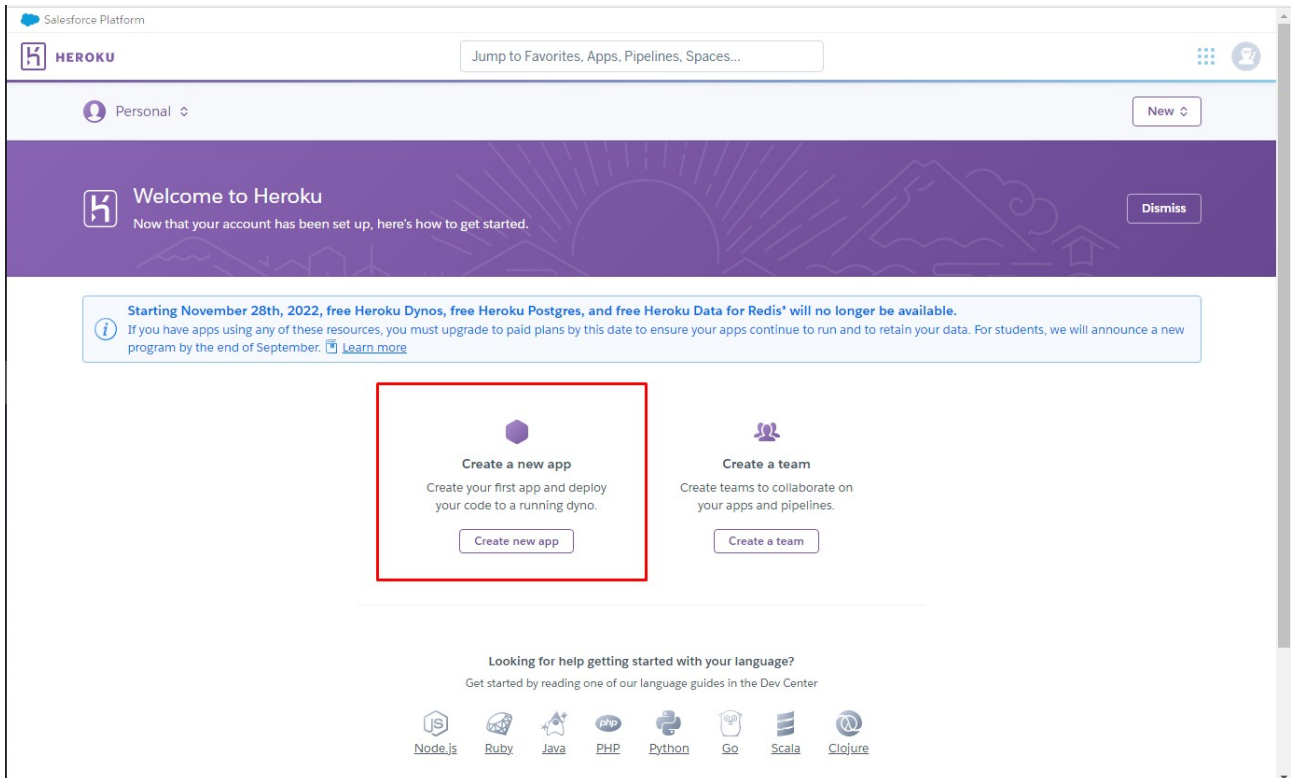
Italian Customers

Are you domiciled in Italy?

☐ No.

Accept

- 7º Passo:
  - Agora iremos em “Create new app”. Definiremos o nome do “app” e sua região que por padrão é Estado Unidos da América



- 8º Passo:
  - Agora iremos até o campo “Resources”:

The screenshot shows the Heroku dashboard for the application 'nappacademypostgresql'. The 'Resources' tab is selected and highlighted with a red box. The dashboard includes sections for adding the app to a pipeline, deployment methods (Heroku Git, GitHub, Container Registry), and instructions for deploying using Heroku Git. The 'Add this app to a pipeline' section includes a dropdown to 'Choose a pipeline'. The 'Deployment method' section shows options for Heroku Git, GitHub, and Container Registry. The 'Deploy using Heroku Git' section provides instructions on how to install the Heroku CLI and create a new Git repository.

- 9º Passo:
  - Ir até o campo “Add-ons” e pesquisar por “Postgresql” e selecionar a opção Heroku Postgresql, sendo a opção “Free” da ferramenta

The screenshot shows the Heroku dashboard for the application 'nappacademypostgresql'. The 'Add-ons' section is highlighted with a red box. It shows that the 'heroku-postgresql' add-on has been installed. Below this, there is a search bar and a list of add-ons. The 'Heroku Postgres' add-on is listed with the plan 'Hobby Dev' and 'Free' selected. The estimated monthly cost is shown as '\$0.00'. The footer of the page includes links for 'heroku.com', 'Blogs', 'Careers', 'Documentation', 'Support', 'Terms of Service', 'Privacy', 'Cookies', and '© 2022 Salesforce.com'.

- 10º Passo:
  - Clicar no “Heroku Postgresql”, iremos ser redirecionados a página de configuração de hospedagem.

Salesforce Platform

DATA

Datastores > postgresql-metric-73686

SERVICE heroku-postgresql PLAN hobby-dev BILLING APP nappacademypostgresql

Overview Durability Settings Dataclips

HEALTH

Available

PRIMARY Yes VERSION 14.5 CREATED 3 minutes ago MAINTENANCE Unsupported ROLLBACK Unsupported

UTILIZATION

0 of 20	0 of 10,000	8.5 MB	0
CONNECTIONS	ROWS IN COMPLIANCE	DATA SIZE	TABLES

heroku.com Blogs Careers Documentation Support Terms of Service Privacy Cookies © 2022 Salesforce.com

- 11º Passo:
  - A partir deste momento nosso Banco de Dados já está hospedado no Heroku, agora iremos procurar suas credencias de acesso, iremos clicar em “Settings” e após isto vamos expandir o campo “View Credentials”

Salesforce Platform

DATA

Datastores > postgresql-metric-73686

SERVICE heroku-postgresql PLAN hobby-dev BILLING APP nappacademypostgresql

Overview Durability **Settings** Dataclips

ADMINISTRATION

Database Credentials

Get credentials for manual connections to this database.

View Credentials...

Reset Database

Reset the database to its originally-provisioned state, deleting all data inside it.

Reset Database...

Destroy Database

Destroys the database and all of the data inside it.

Destroy Database...

heroku.com Blogs Careers Documentation Support Terms of Service Privacy Cookies © 2022 Salesforce.com

- 12º Passo:
  - Agora temos as credencias para acessar o nosso Banco de Dados:

ADMINISTRATION

Database Credentials

Get credentials for manual connections to this database.

Cancel

Please note that **these credentials are not permanent**.

Heroku rotates credentials periodically and updates applications where this database is attached.

Host	ec2-44-205-112-253.compute-1.amazonaws.com
Database	dcpf93c3k9k5lf
User	hlstnjnykzzmd
Port	5432
Password	ec7eba8f698c2e5802c85069a9edd63747c31da530b2c94e25d76fc954d4d88e
URI	postgres://hlstnjnykzzmd:ec7eba8f698c2e5802c85069a9edd63747c31da530b2c94e25d76fc954d4d88e@ec2-44-205-112-253.compute-1.amazonaws.com:5432/dcpf93c3k9k5lf
Heroku CLI	heroku pg:psql postgresql-metric-73686 --app nappacademypostgresql

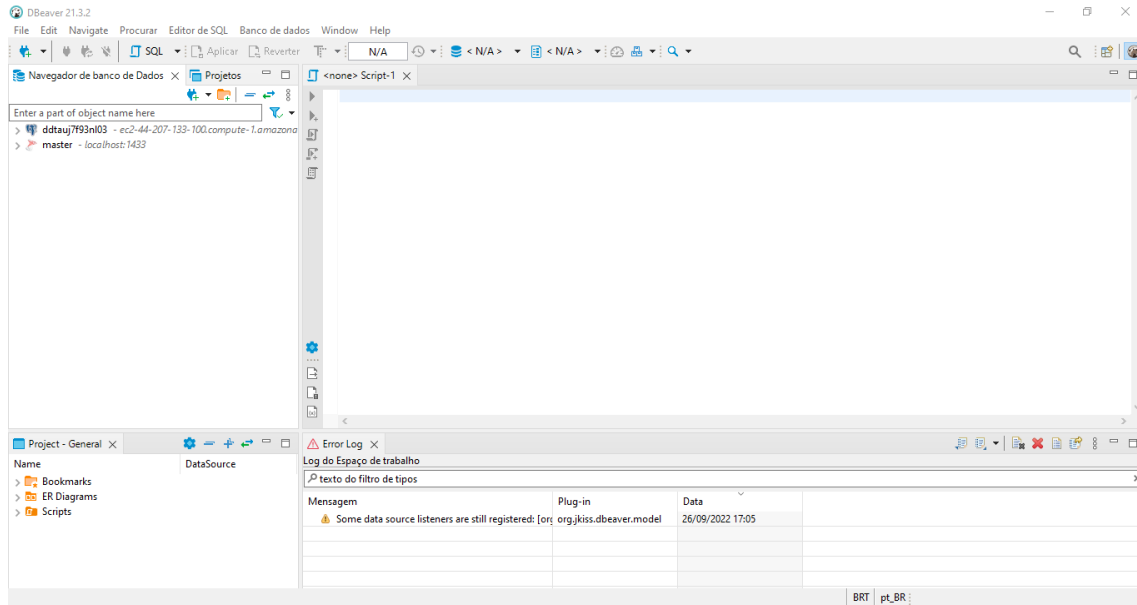
**Criação de um banco de  
dados no SGBD com  
inserção de registros**



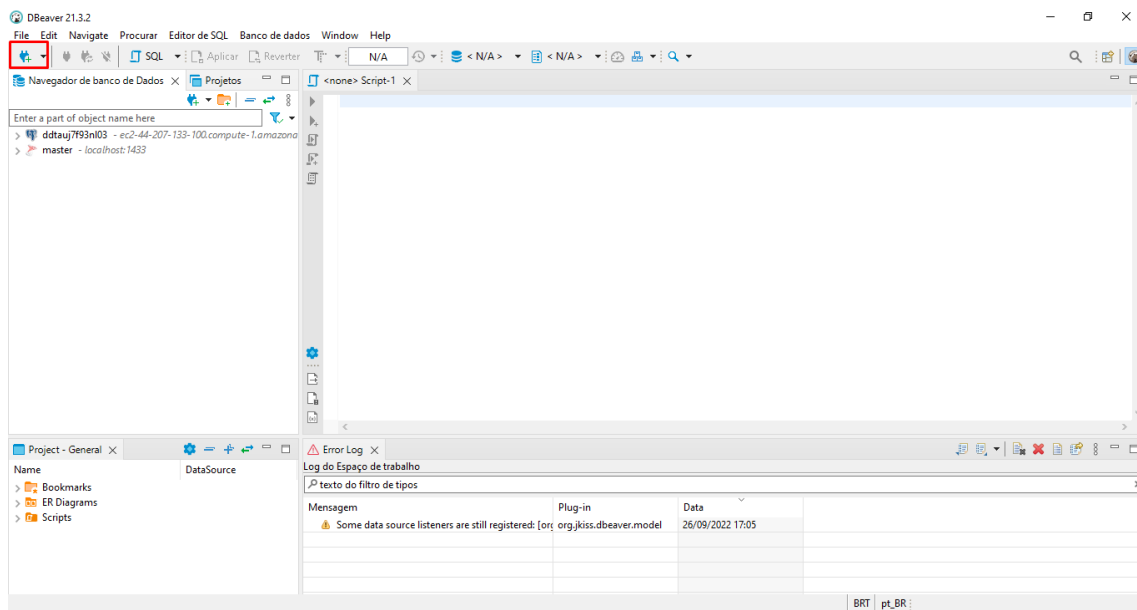
# CONEXÃO COM O BANCO, CRIAÇÃO DAS TABELAS E INSERÇÃO DE REGISTROS

- CONFIGURAÇÃO DA FERRAMENTA

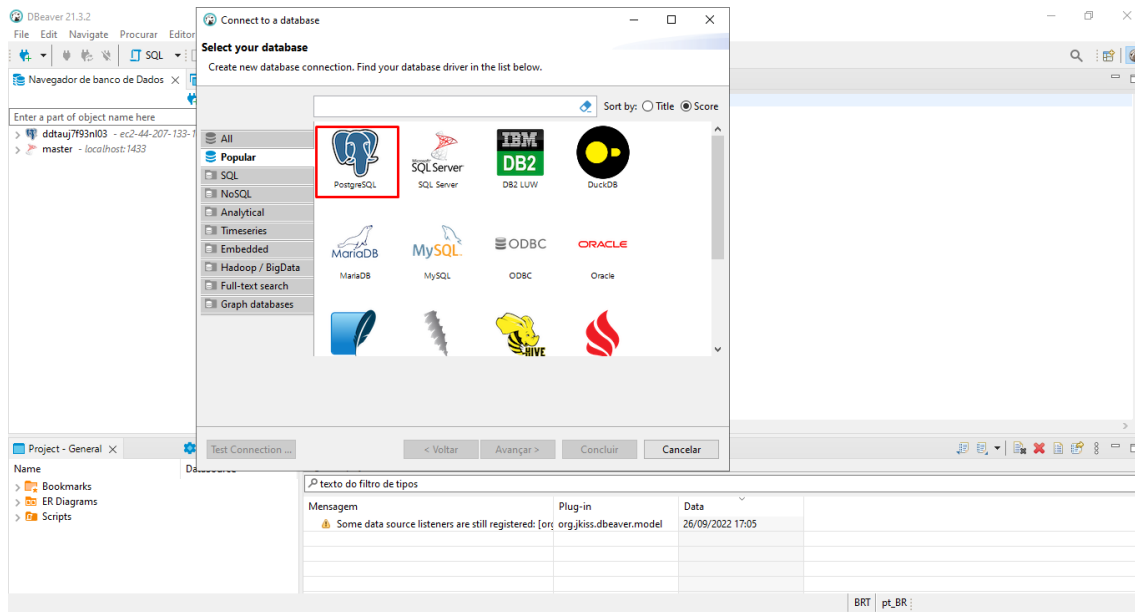
Tela inicial do nosso aplicativo:



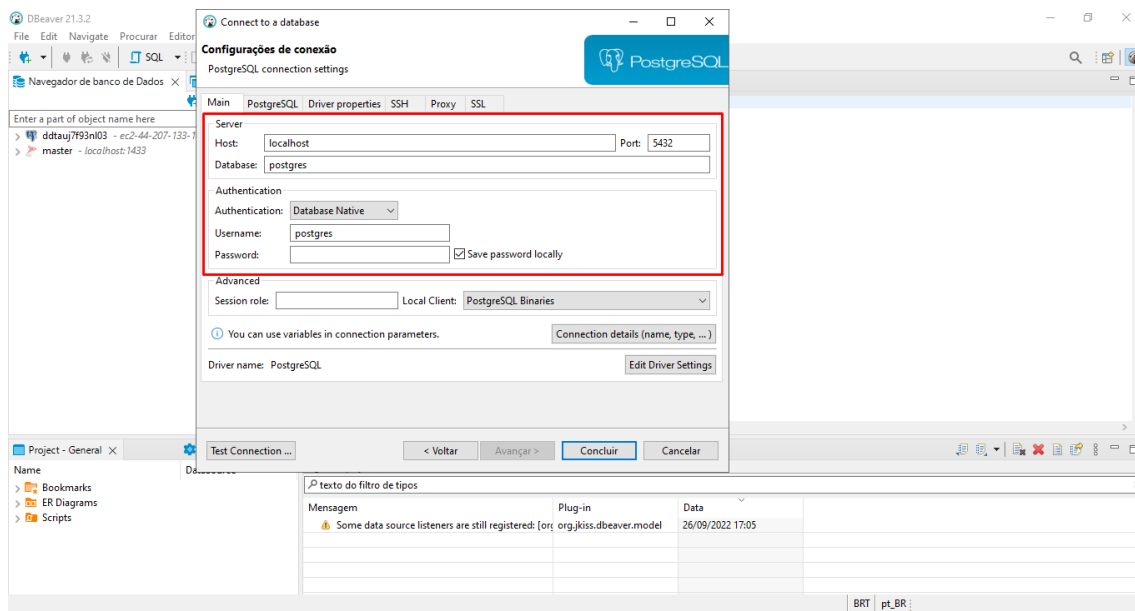
Ao entrar no aplicativo nos deparamos com essa tela inicial, onde no canto superior esquerdo é visível um botão parecido com uma tomada através dele que iremos começar a fazer a conexão com o banco.



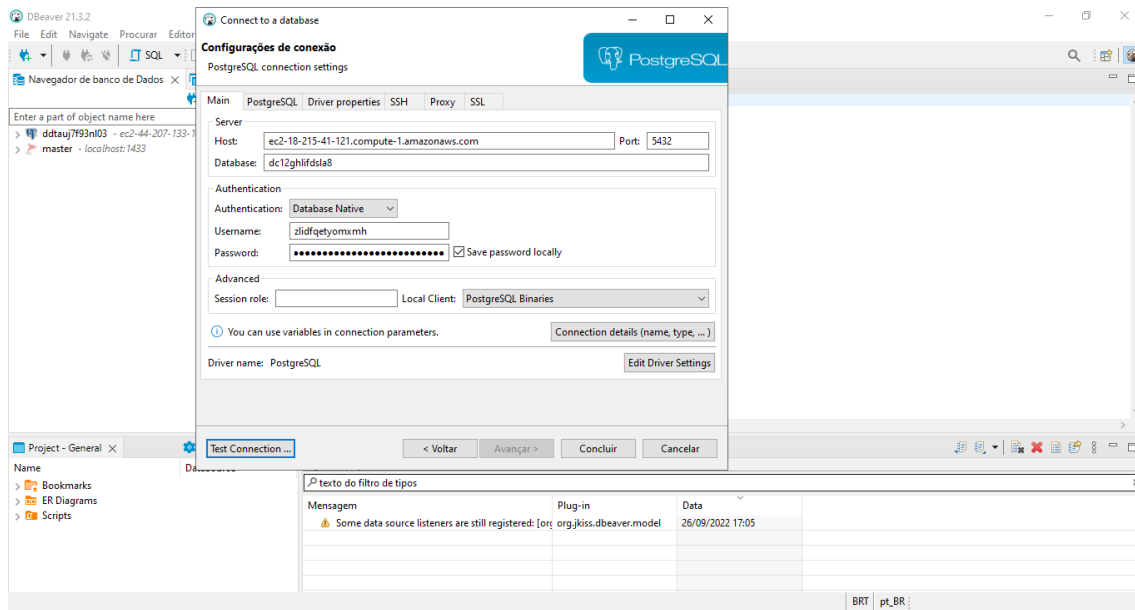
Após clicar nesse botão, é aberto uma janela com várias opções de bancos para se escolher em qual conectar, o nosso é banco foi feito em PostgreSQL, esse que está selecionado abaixo:



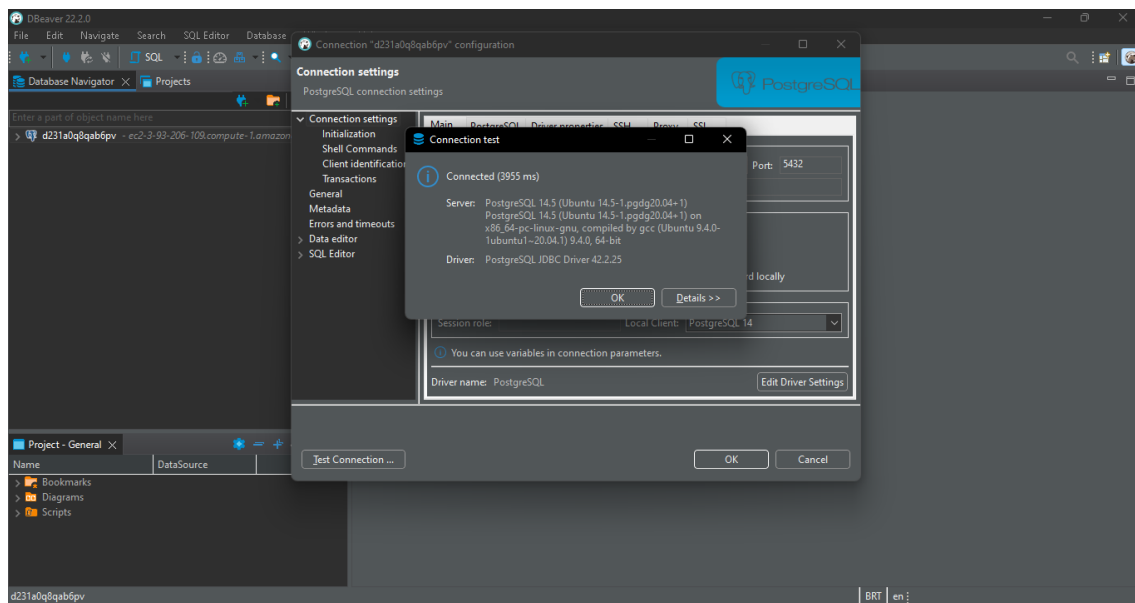
Depois que escolhermos nosso banco, é aberto essa janela para fazer a conexão, pedindo informações de nossa conexão, como o host, port, database, user e nossa password.



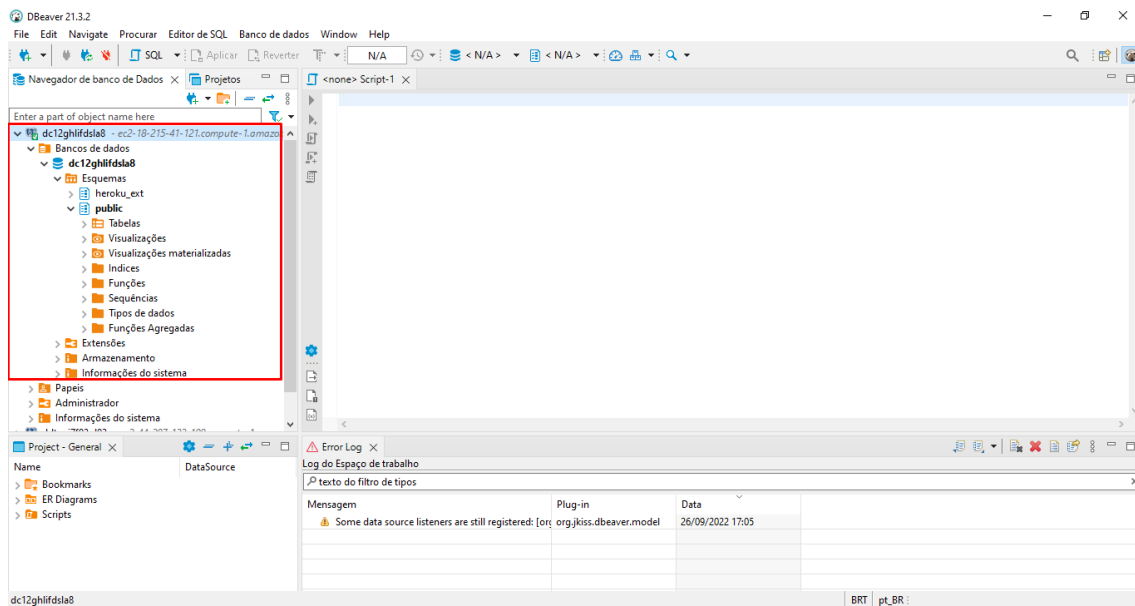
Após preenchido todas as informações, usamos o test connection, para verificar se tudo está certo e se a conexão foi realizada ou não:



Ufa, conexão realizada com sucesso, podemos prosseguir.



Após todos esses passos conseguimos a conexão e esse é nosso banco, sem tabelas por enquanto, apenas com dois Esquemas Heroku e o Public onde ficará as nossas tabelas.

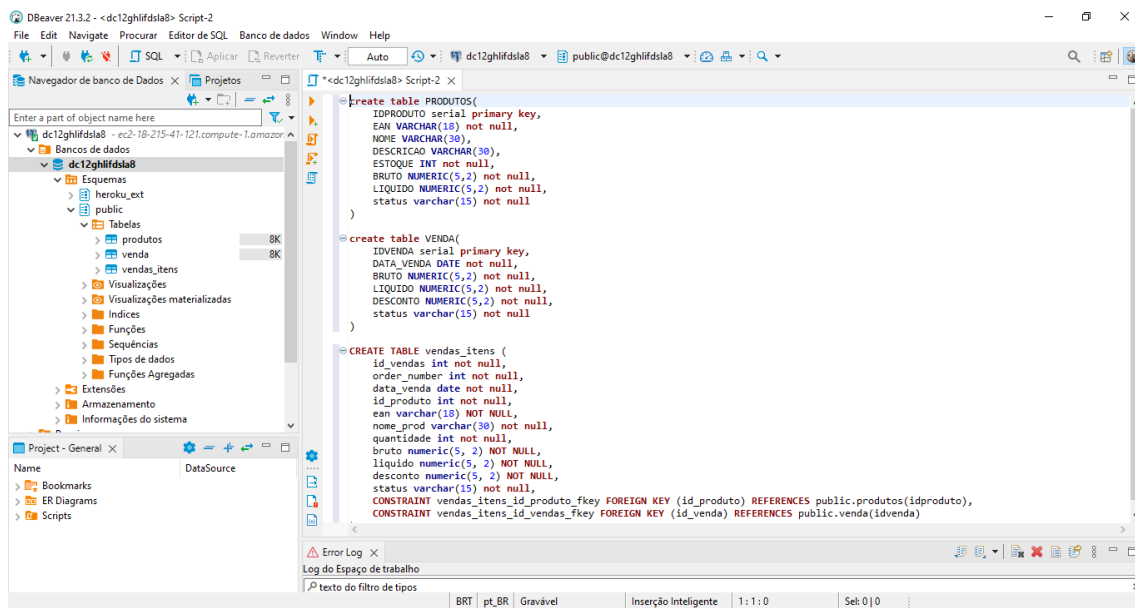


## • CRIANDO TABELAS E INSERINDO DADOS

**OBS:** Para a criação do banco foi tomado como base as queries que são usadas na equipe do sellout score.

No nosso banco possuímos três tabelas:

Vendas, Vendas por item e produtos, abaixo está a criação delas:



Na tabela de Vendas temos:

**IDVENDA** – Chave primária e auto\_incrementavel da nossa tabela.

**DATA\_VENDA** – Campo do tipo date onde será especificado a data de nossa venda.

**BRUTO** – Preço Bruto da nossa venda

**LÍQUIDO** – Preço Líquido da nossa venda, onde pode ser aplicado um desconto

**DESCONTO** – O desconto da nossa venda

**STATUS** – O status da venda define se a venda é **faturada** ou **cancelada**.

Produtos:

**IDPRODUTO** – Id de registro do produto, chave primária e auto\_incrementavel da nossa tabela.

**EAN** – Um código universal composto por 13 dígitos para a identificar um produto que pode ser precificado e vendido no varejo.

**NOME** – Nome do nosso produto.

**DESCRICAO** – Descrição do produto.

**ESTOQUE** – Quantidade do produto em estoque.

**BRUTO** – Preço bruto do produto.

**LÍQUIDO** – Preço líquido do produto.

**STATUS** – Status do produto que define se ele está **ativo** ou **inativo**.

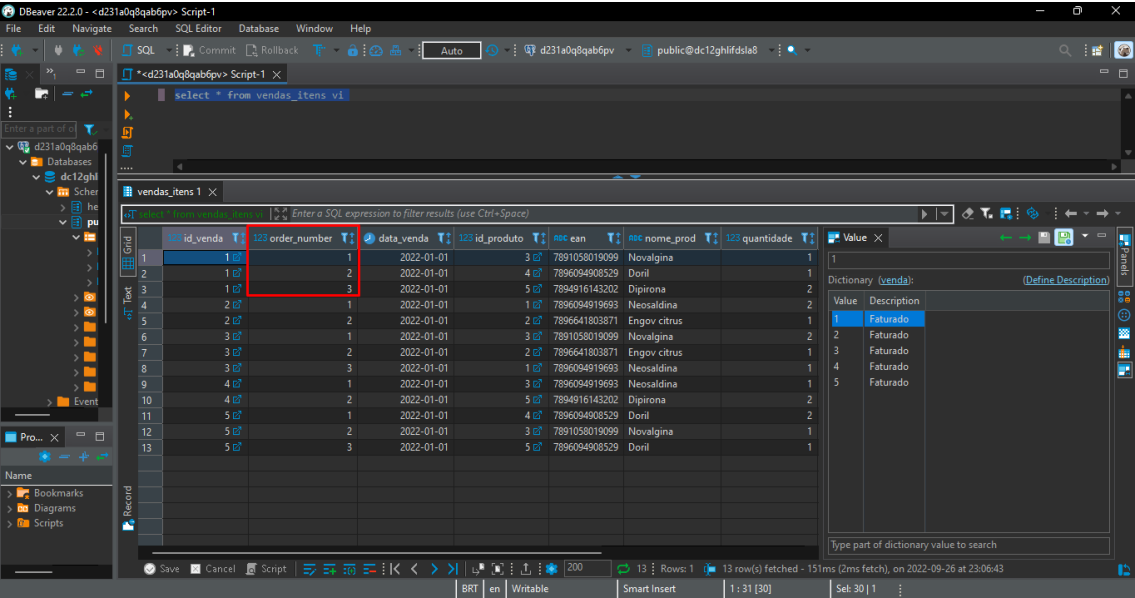
Vendas por item:

Diferente das outras tabelas essa aqui não possui chave primária, possuí apenas duas chaves estrangeiras, **ID\_VENDA** que faz referencia a tabela de Vendas e **ID\_PRODUTO** que faz referência a tabela de Produtos.

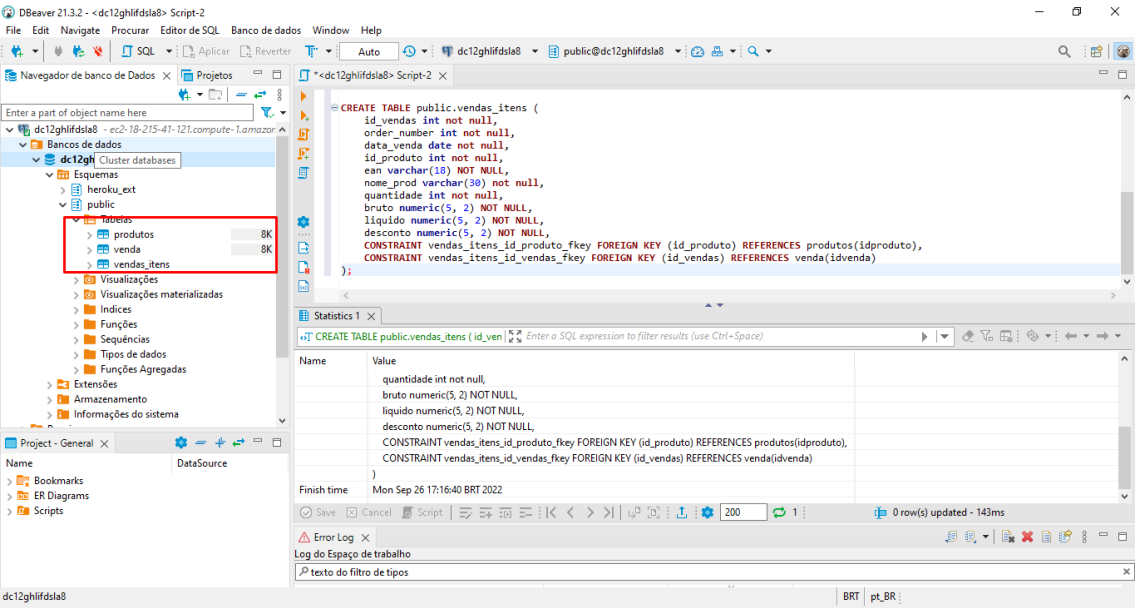
Essa tabela contém: **EAN, ID\_PRODUTO, BRUTO, LÍQUIDO, DESCONTO, STATUS, DATA\_VENDA, NOME\_PRODUTO** que já foram explicados.

**ORDER\_NUMBER** – O order\_number é o id sequencial de produtos presentes em uma venda, vamos imaginar uma venda que contenha 3 produtos, com esse campo identificamos quantos produtos teremos nessa venda, como no exemplo abaixo:

Ele enumera os itens presentes na venda.

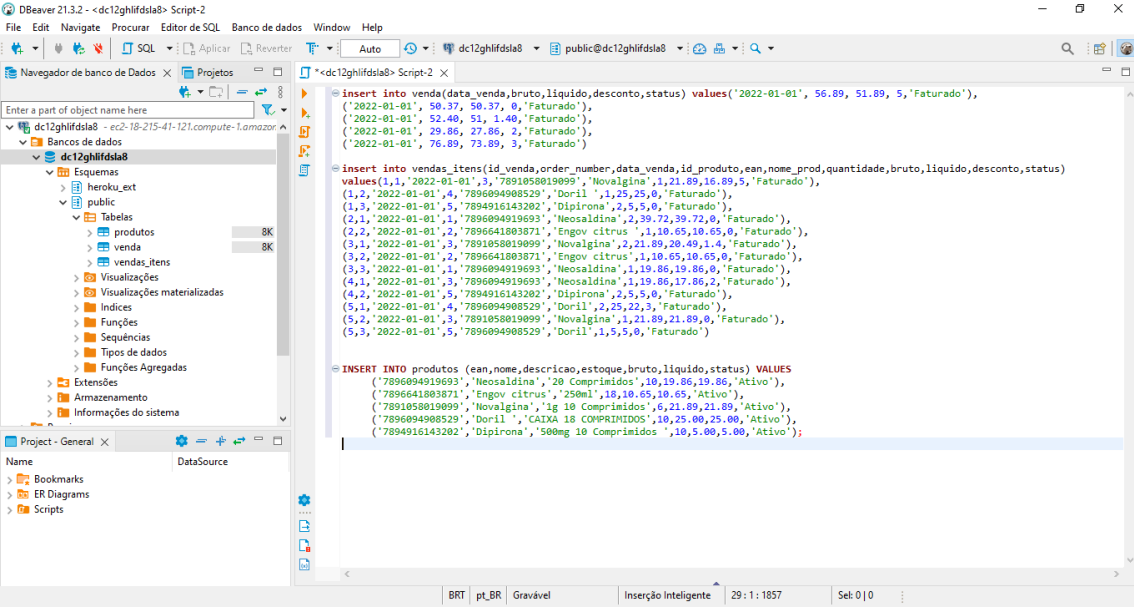


Após a explicação e criação das nossas tabelas, elas iram aparecer no lado esquerdo de nossa tela, dentro de public.



Vamos povoar um pouco com alguns registros, aqui está a print de um comando insert padrão, mas lembrando que nas tabelas de vendas e produtos, os ids não precisam ser especificados porque são auto\_incrementaveis.

Fora isso um comando padrão.



The screenshot shows the DBeaver 21.3.2 interface with a script editor containing three SQL insert statements. The first statement inserts into the 'venda' table, the second into 'vendas\_itens', and the third into 'produtos'. The database schema on the left shows tables like 'produtos', 'venda', and 'vendas\_itens'.

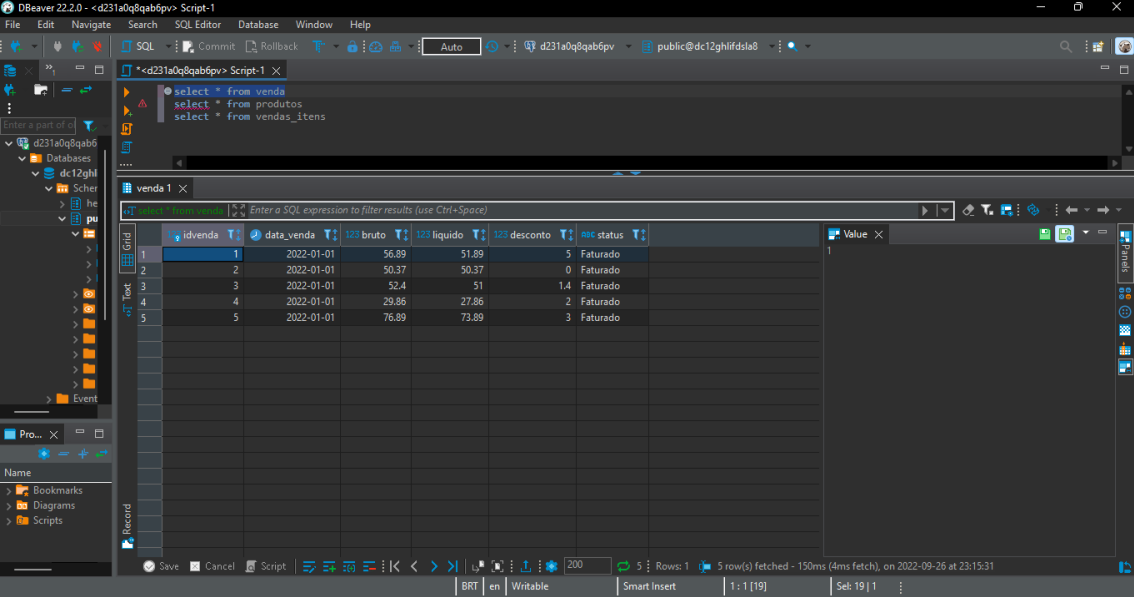
```
insert into venda(data_venda,bruto,liquido,desconto,status) values('2022-01-01', 56.89, 51.89, 5, 'Faturado'),
('2022-01-01', 50.37, 50.37, 0, 'Faturado'),
('2022-01-01', 52.40, 51.14, 0, 'Faturado'),
('2022-01-01', 29.86, 27.86, 2, 'Faturado'),
('2022-01-01', 76.89, 73.89, 3, 'Faturado')

insert into vendas_itens(id_venda,order_number,data_venda,id_produto,ean,nome_prod,quantidade,bruto,liquido,desconto,status)
values(1,1,'2022-01-01',3,'7891050019099','Novalgina',1,21.89,16.89,5,'Faturado'),
(1,2,'2022-01-01',4,'7896094908529','Doril',1,25.25,0,'Faturado'),
(1,3,'2022-01-01',5,'7894916143202','Dipirona',2,5,5,0,'Faturado'),
(2,1,'2022-01-01',1,'7896094919693','Neosaldina',2,39.72,39.72,0,'Faturado'),
(2,2,'2022-01-01',2,'7896641803871','Engov citrus',1,10.65,10.65,0,'Faturado'),
(3,1,'2022-01-01',3,'7891050019099','Novalgina',2,21.89,20.49,1.4,'Faturado'),
(3,2,'2022-01-01',2,'7896641803871','Engov citrus',1,10.65,10.65,0,'Faturado'),
(3,3,'2022-01-01',1,'7896094919693','Neosaldina',1,19.86,19.86,0,'Faturado'),
(4,1,'2022-01-01',3,'7896094919693','Neosaldina',1,19.86,17.86,2,'Faturado'),
(4,2,'2022-01-01',5,'7894916143202','Dipirona',2,5,5,0,'Faturado'),
(5,1,'2022-01-01',4,'7896094908529','Doril',2,25.25,0,'Faturado'),
(5,2,'2022-01-01',3,'7891050019099','Novalgina',1,21.89,21.89,0,'Faturado'),
(5,3,'2022-01-01',5,'7896094908529','Doril',1,5,5,0,'Faturado')

INSERT INTO produtos (ean,nome_descricao,estoque,bruto,liquido,status) VALUES
('7896094919693','Neosaldina',20,39.72,39.72,'Ativo'),
('7896641803871','Engov citrus',250ml,10.65,10.65,'Ativo'),
('7891050019099','Novalgina',1g,21.89,21.89,'Ativo'),
('7896094908529','Doril',1,25.25,0,'Ativo'),
('7894916143202','Dipirona',500mg,10,5.00,5.00,'Ativo');
```

E para finalizar um select \* em todas as tabelas para vermos os dados inseridos:

Venda:



The screenshot shows the DBeaver 22.0 interface with a SQL query executed in the script editor. The results are displayed in a table grid with columns: id\_venda, data\_venda, bruto, liquido, desconto, and status. The status column is highlighted in blue.

id_venda	data_venda	bruto	liquido	desconto	status
1	2022-01-01	56.89	51.89	5	Faturado
2	2022-01-01	50.37	50.37	0	Faturado
3	2022-01-01	52.4	51	1.4	Faturado
4	2022-01-01	29.86	27.86	2	Faturado
5	2022-01-01	76.89	73.89	3	Faturado



Produto:

The screenshot shows the DBeaver SQL Editor with a query window titled "Script-1". The query is: `select * from vendas;  
select * from produtos;  
select * from vendas_itens;`. The "produtos" table is selected in the left sidebar. The main window displays the results of the query in a grid format. The status bar at the bottom indicates "5 row(s) fetched - 150ms, on 2022-09-26 at 23:15:47".

idproduto	ean	nome	descricao	estoque	bruto	liquido	status
1	7896094919693	Neosaldina	20 Comprimidos	10	19.86	19.86	Ativo
2	7896641803871	Engov citrus	250ml	18	10.65	10.65	Ativo
3	7891058019099	Novalgina	1g 10 Comprimidos	6	21.89	21.89	Ativo
4	7896094908529	Doril	CAIXA 18 COMPRIMIDOS	10	25	25	Ativo
5	7894916143202	Dipirona	500mg 10 Comprimidos	10	5	5	Ativo

Vendas por item:

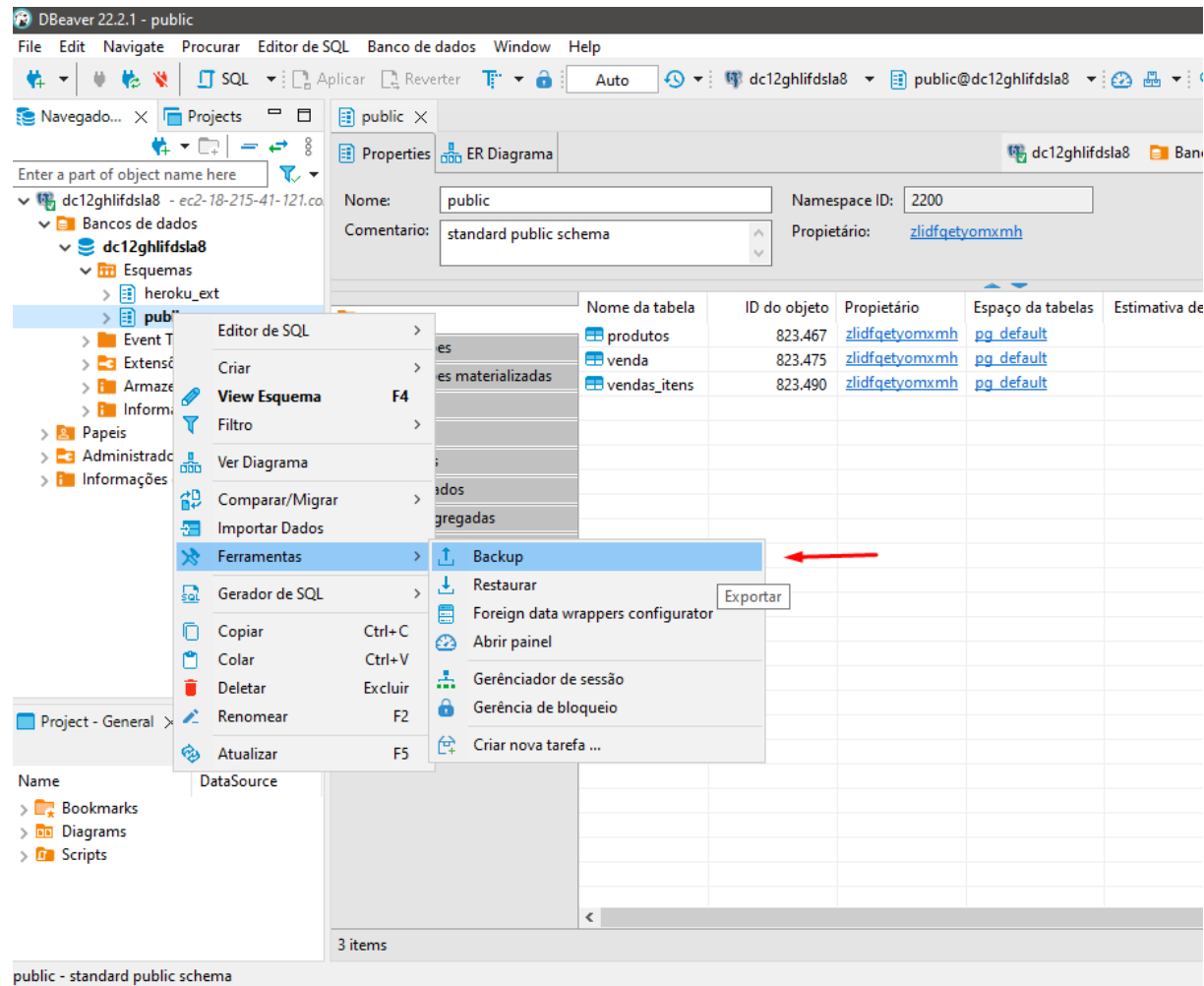
The screenshot shows the DBeaver SQL Editor with a query window titled "Script-1". The query is: `select * from vendas;  
select * from produtos;  
select * from vendas_itens;`. The "vendas\_itens" table is selected in the left sidebar. The main window displays the results of the query in a grid format. The status bar at the bottom indicates "13 row(s) fetched - 165ms (1ms fetch), on 2022-09-26 at 23:16:11".

id_venda	order_number	data_venda	id_produto	ean	nome_prod	quantidade
1	1	2022-01-01	3	7891058019099	Novalgina	1
2	2	2022-01-01	4	7896094908529	Doril	1
3	3	2022-01-01	5	7894916143202	Dipirona	2
4	2	2022-01-01	1	7896094919693	Neosaldina	2
5	2	2022-01-01	2	7896641803871	Engov citrus	1
6	3	2022-01-01	3	7891058019099	Novalgina	2
7	3	2022-01-01	2	7896641803871	Engov citrus	1
8	3	2022-01-01	1	7896094919693	Neosaldina	1
9	4	2022-01-01	3	7896094919693	Neosaldina	1
10	4	2022-01-01	5	7894916143202	Dipirona	2
11	5	2022-01-01	4	7896094908529	Doril	2
12	5	2022-01-01	3	7891058019099	Novalgina	1
13	5	2022-01-01	5	7896094908529	Doril	1

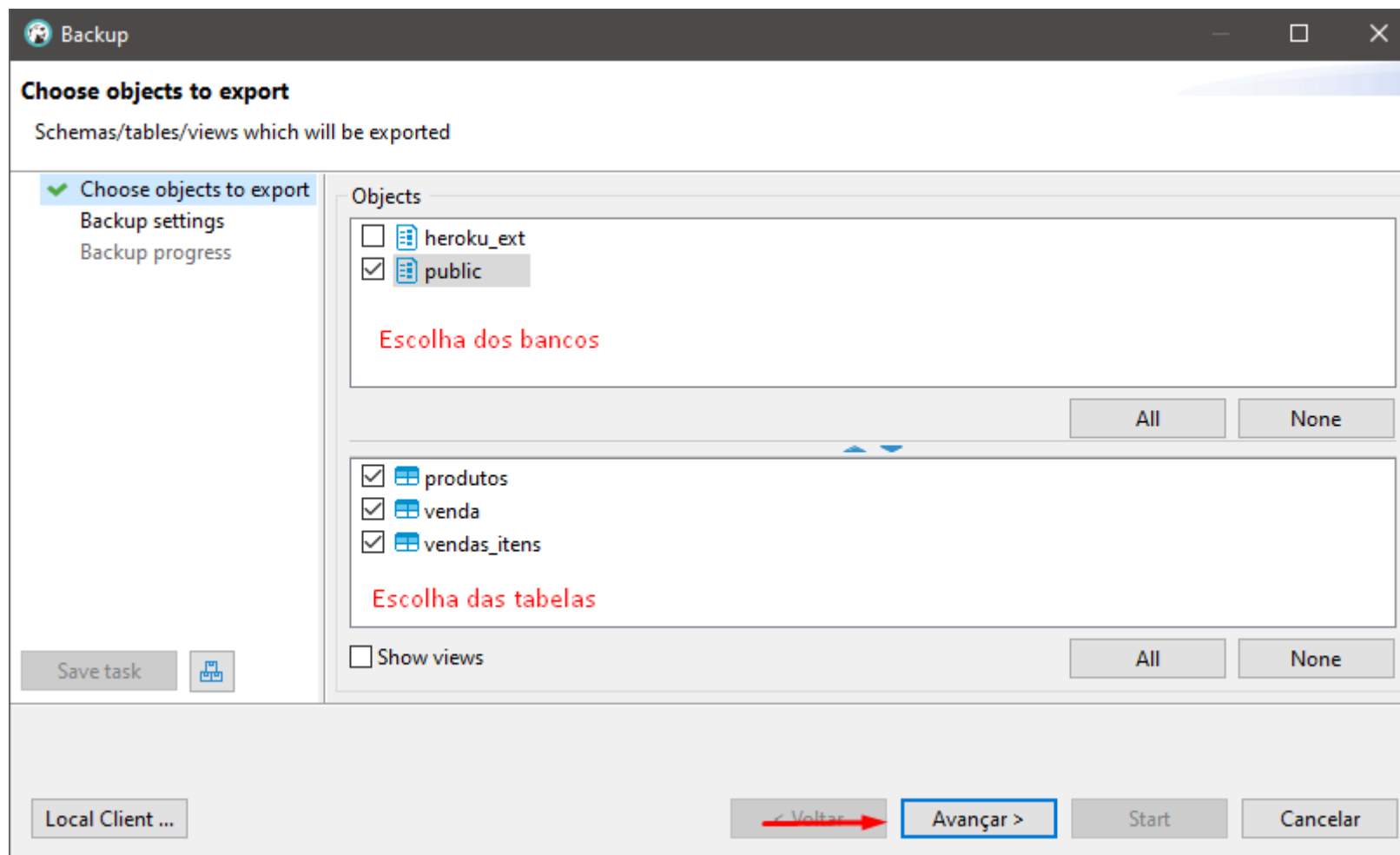
# **Backup e Restauração do Banco de Dados**

Para criar um backup de um banco de dados no DBeaver é bem simples.

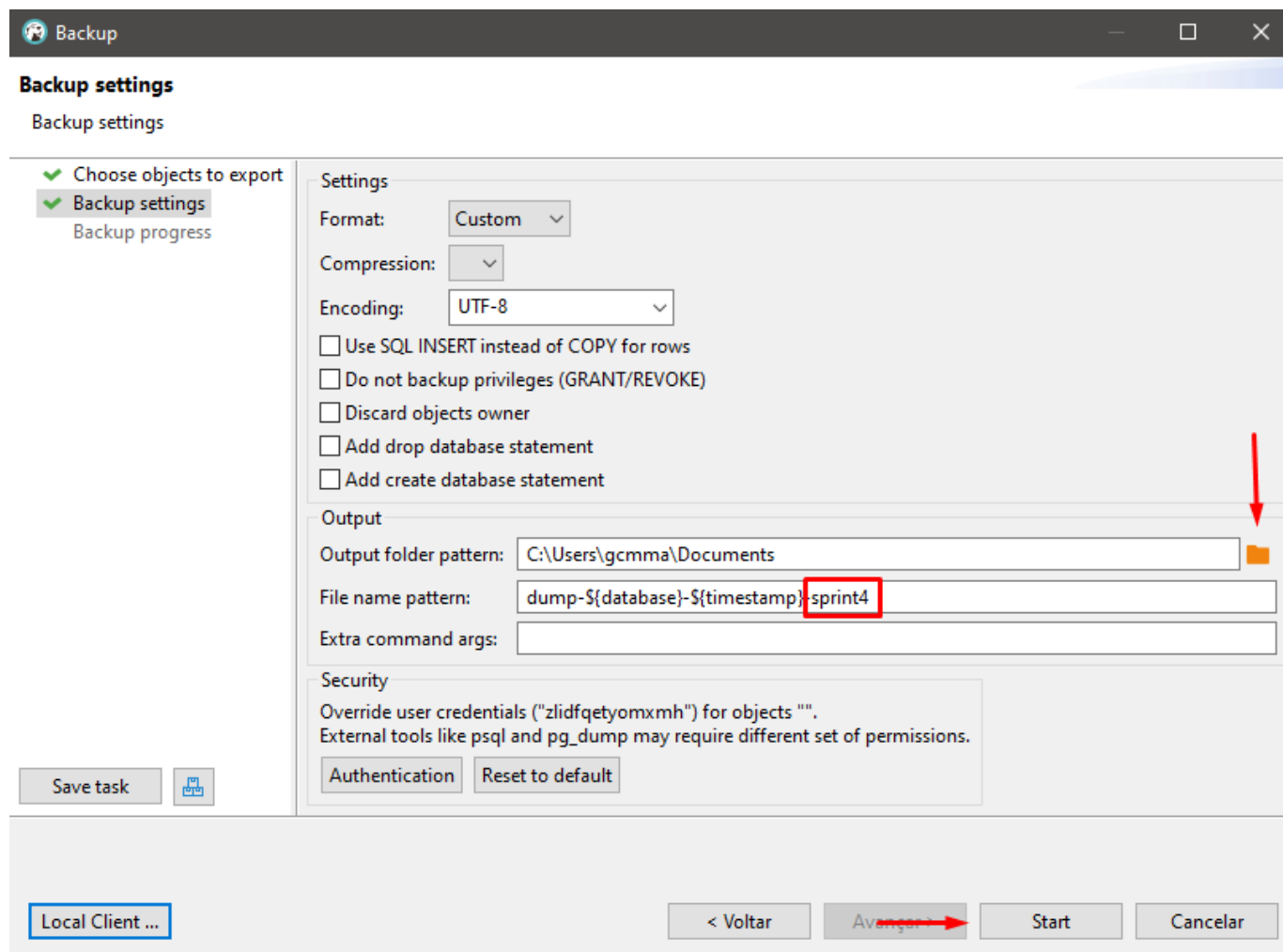
1º Passo: abra o DBeaver, selecione o banco de dados e com o botão direito do mouse, clique em **Tools – Backup**



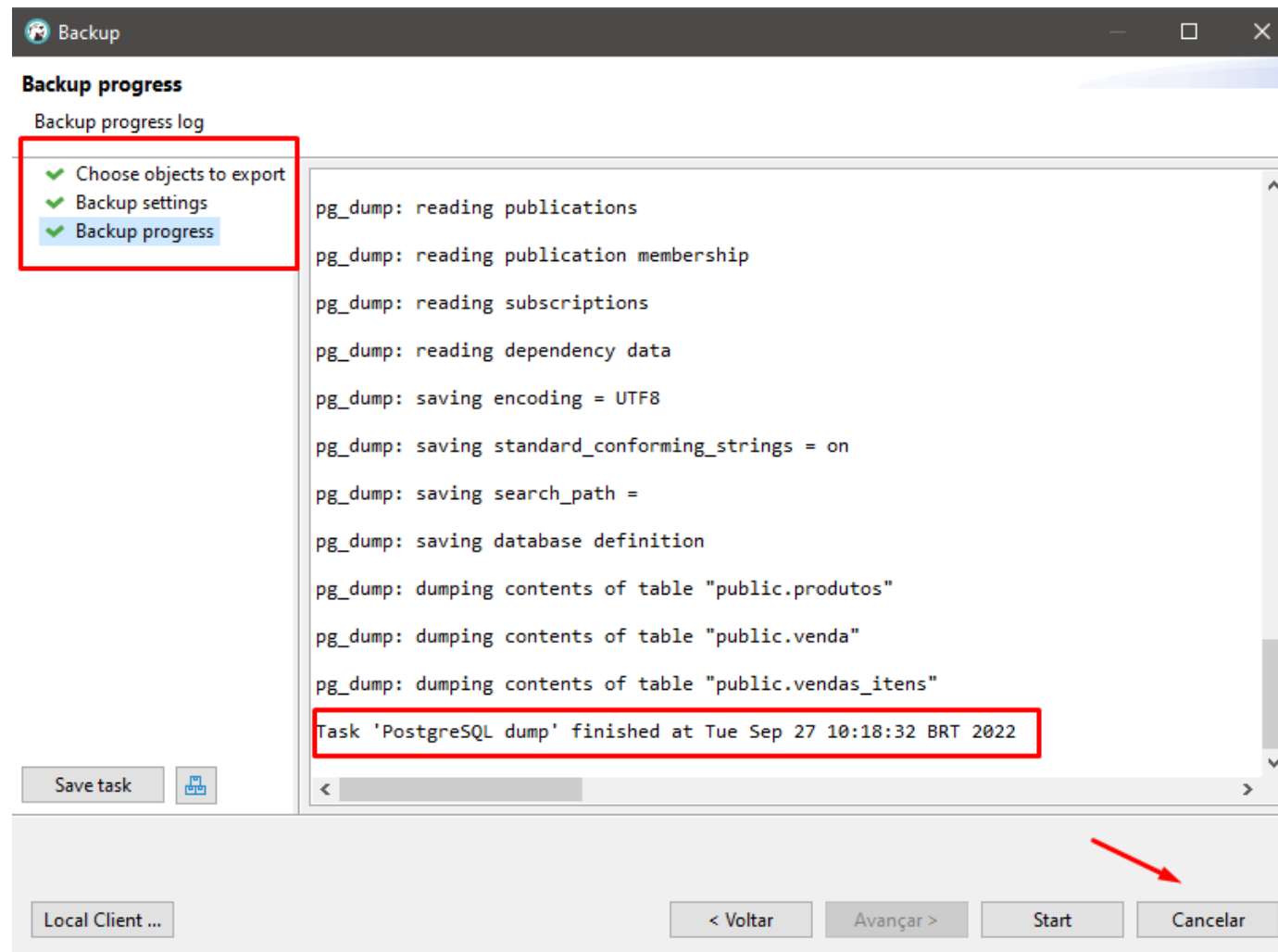
2º Passo: será aberta uma janela, onde você deverá escolher quais bancos e quais tabelas estarão inclusos no backup. Após selecionados, clique em **Avançar**.



3º passo: Será aberta uma janela para configuração do backup, onde você deve escolher o caminho e o nome do arquivo de backup por exemplo. Após feitas suas escolhas, clique em **Start (Iniciar)**

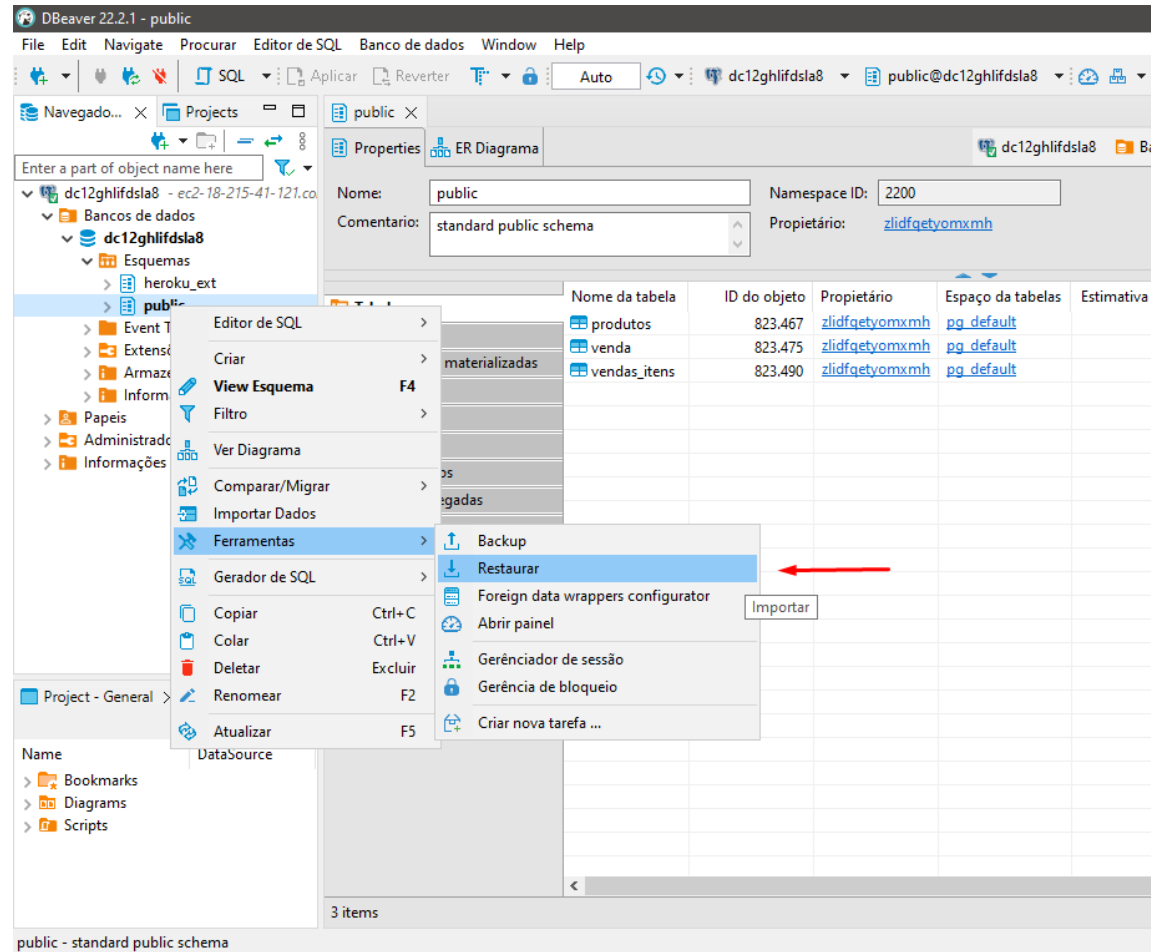


4º Passo: na próxima janela, será mostrado a execução da tarefa. Caso tudo esteja ok, seu backup está pronto. Basta clicar em **Cancelar** e verificar no caminho escolhido o arquivo de backup.



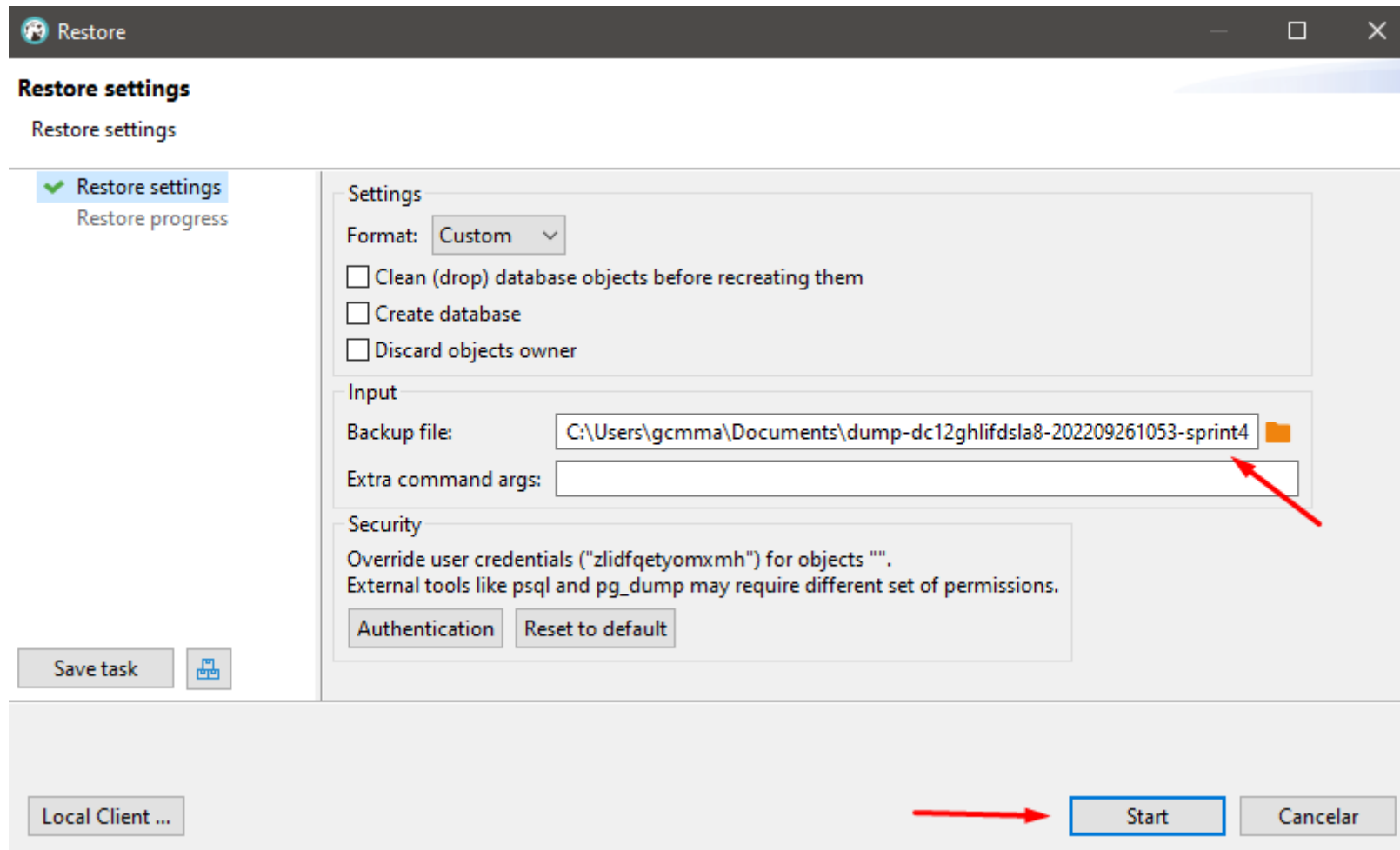
Para fazer uma restauração de um backup já feito anteriormente, basta seguir os passos:

1º Passo: abra o DBeaver, selecione o banco de dados e com o botão direito do mouse, clique em **Tools – Restore**



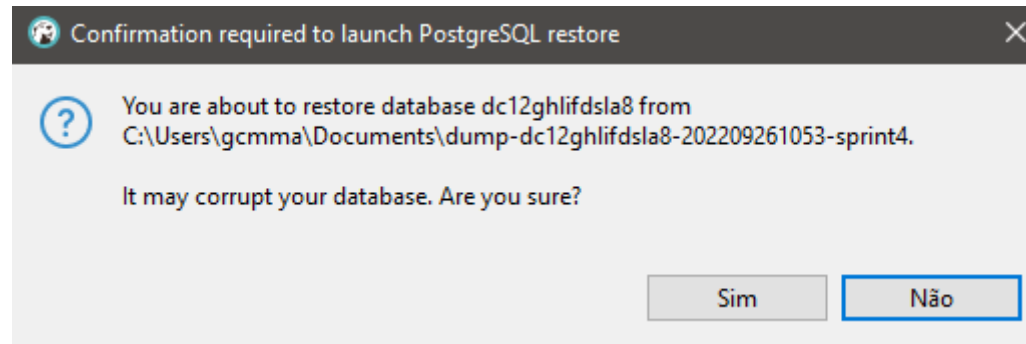


2º Passo: na próxima janela será necessário escolher o caminho onde se encontra o arquivo de backup. Após selecionado, clique em **Start** para começar a restauração.



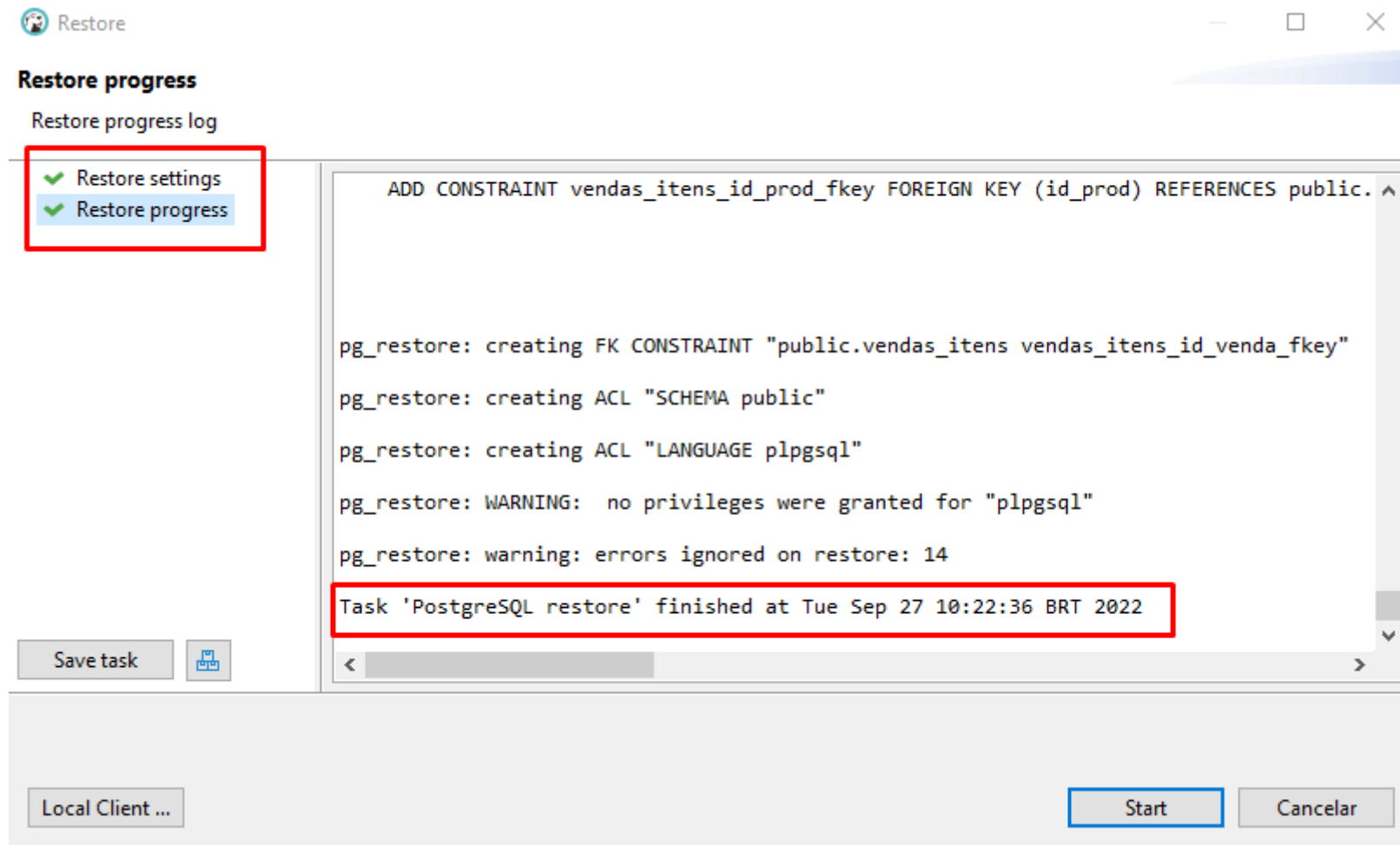
3º Passo: uma janela de aviso será mostrada para informar que se for feita uma restauração, seu arquivo poderá ser corrompido. Mas uma restauração deve ser feita justamente para corrigir um problema que possa ter ocorrido em seu banco. Então se você tiver certeza que precisa de uma restauração, clique em

**Sim**



4º Passo: Será mostrada uma janela sobre o andamento da restauração. Clique em **Start** e após concluído o processo, clique em **Cancelar**.

E pronto, seu banco estará recuperado!



# **Conexão entre o SGBD e uma linguagem de programação (Python)**

Conexão no banco de dados postgres utilizando uma linguagem de programação, Python;

Primeiro é preciso ter a biblioteca psycopg2 e pandas instalado, abra o cmd e verifique utilizando o comando [ pip list ]; caso não se encontre pode seguir com esse comando para instalação [ pip install psycopg2 ] e [ pip install pandas ]

1- Após isso vamos desenrolar o código, crie um nome para o arquivo e realize o import deles conforme imagem abaixo;

```
conexao_sql_s4.py > ...
1  import psycopg2 #conexão e comandos no banco de dados postgres
2  import pandas as pd #manipulação e tratamento de dados
3
```

2- Vamos criar uma variável para conter as credenciais de acesso, basta preencher em seu devido local. Nota, caso a porta seja padrão pode omiti-la.

```
4
5  #=====Conexão com o banco de dados Postgres=====
6  conexao = psycopg2.connect(host = '',
7                             database = '',
8                             user = '',
9                             password = '')
```

3-Declare um cursor para para navegar e realizar procedimentos no banco de dados;

```
11  #=====Declarando o cursor=====
12  cursor = conexao.cursor()
13
```

4-Faremos uma consulta na tabela de vendas\_itens, passe para o cursor o que você precisa executar (o select). Vamos armazenar o resultado em uma variável que para este caso chama-se linhas.

Após isso faremos um tratamento usando a biblioteca pandas para exportar esse arquivo formatado. Crie uma nova variável para receber o DataFrame formatando o nome de cada coluna (vale lembrar que precisa saber a quantidade de coluna existente em seu select para atribuir).

Salve sua consulta em um arquivo.csv criando um separador de colunas.

```
13
14  #=====realizando consulta order_item=====
15  cursor.execute(f"""select * from vendas_itens""")
16  linhas = cursor.fetchall()#armazena o select executado
17  order_itens = pd.DataFrame(linhas, columns=['Id_venda', 'Order_number', 'Data_venda',
18  order_itens.to_csv('order_itens.csv', index = False, sep=';')
19
```

5-Após isso, encerre o cursor e a conexão;

```
34  
35 #=====Encerradno consulta e conexão  
36 cursor.close()  
37 conexao.close()  
38  
39
```