

Sistemas de Apoio a Decisão

(Inteligência nos Negócios - *Business Inteligente*)

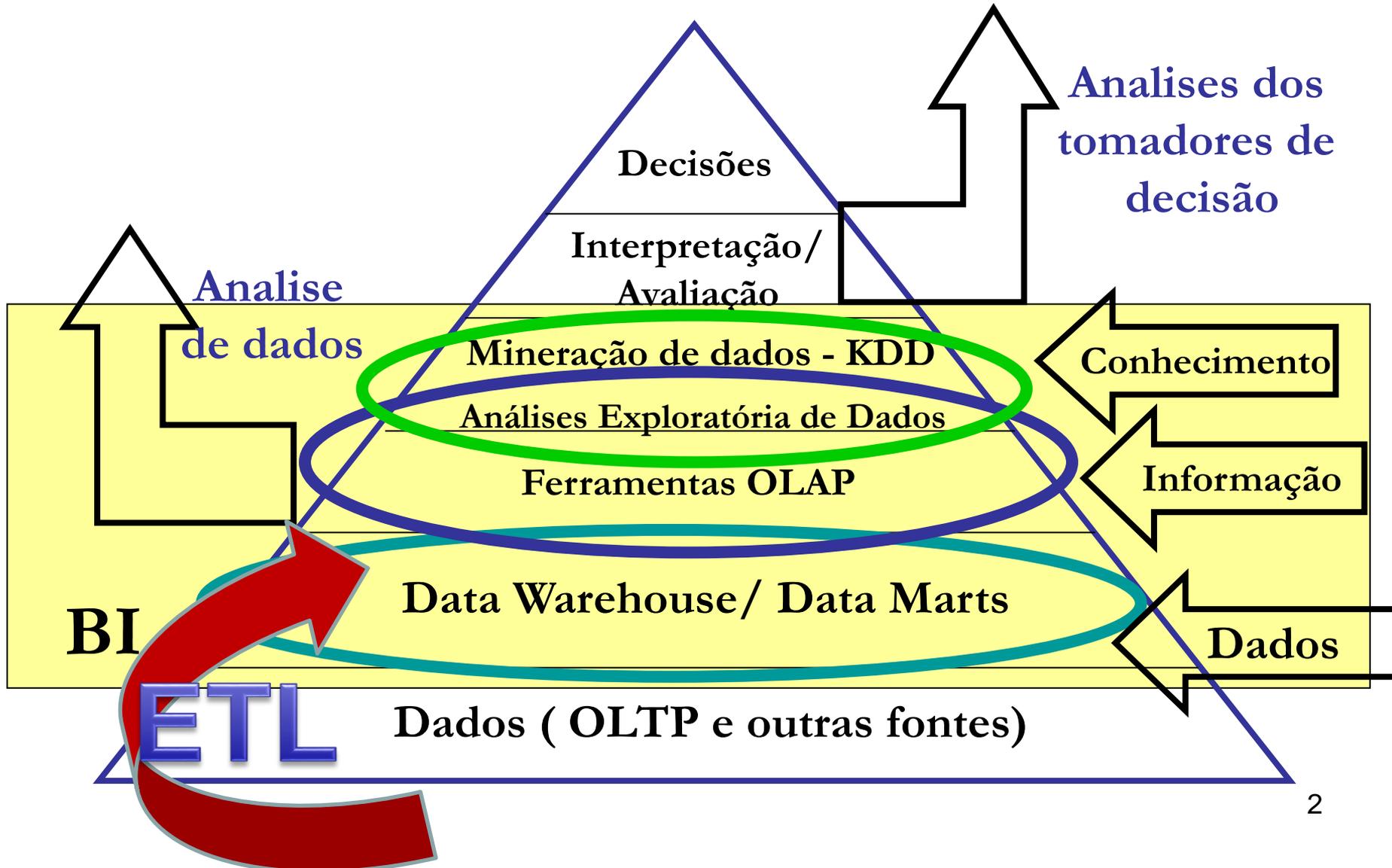


Sistemas de Informação

Aran Bey Tcholakian Morales, Dr. Eng.

(Apostila 3: PostgreSQL)

Fundamentação da disciplina



PostgreSQL



PostgreSQL

Baixar e instalar o aplicativo do endereço (verifique a versão de seu SO): <http://www.postgresql.org/download/> (ver figura);

11th February 2021: PostgreSQL 13.2, 12.6, 11.11, 10.16, 9.6.21, & 9.5.25 Released!

Quick Links

- Downloads
 - Packages
 - Source
- Software Catalogue
- File Browser

Downloads

PostgreSQL Downloads

PostgreSQL is available for download as ready-to-use packages or installers for various platforms, as well as a source code archive if you want to build it yourself.

Packages and Installers

Select your operating system family:



Source code

The source code can be found in the main [file browser](#) or you can access the source control repository directly at git.postgresql.org. Instructions for building from source can be found in the [documentation](#).

Beta/RC Releases and development snapshots (unstable)

There are source code and binary [packages](#) of beta and release candidates, and of the current development code available for testing and evaluation of new

PostgreSQL

11th February 2021: PostgreSQL 13.2, 12.6, 11.11, 10.16, 9.6.21, & 9.5.25 Released!

Windows installers

Interactive installer by EDB

Download the installer certified by EDB for all supported PostgreSQL versions.

This installer includes the PostgreSQL server, pgAdmin, a graphical tool for managing and developing your databases, and StackBuilder; a package manager used to download and install additional PostgreSQL tools and drivers. Stackbuilder includes management, integration, migration, replication, geospatial connectors and other tools.

This installer can run in graphical or silent install modes.

The installer is designed to be a straightforward, fast way to get up and running with PostgreSQL on Windows.

Advanced users can also download a **zip archive** of the binaries, without the installer. This download is intended for users who wish to include PostgreSQL



Why EDB? ▾

Products ▾

Services ▾

Support ▾

Resources ▾

Plans

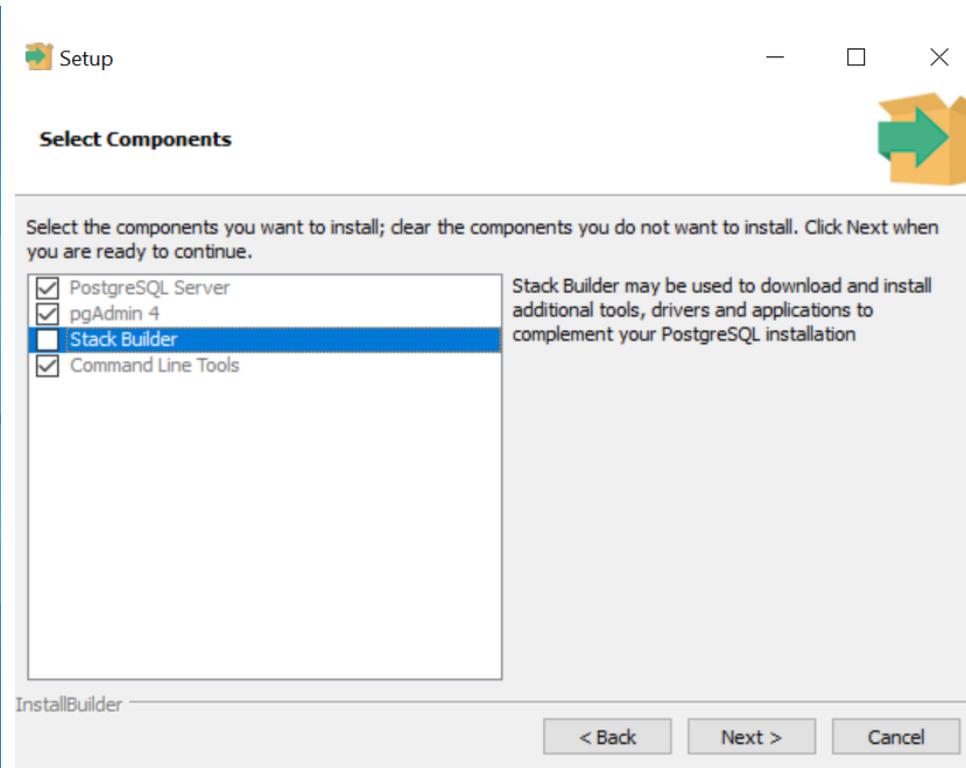
Contact

Sign In

[Downloads](#)

Version	Linux x86-64	Linux x86-32	Mac OS X	Windows x86-64	Windows x86-32
13.2	N/A	N/A	Download	Download	N/A
12.6	N/A	N/A	Download	Download	N/A
11.11	N/A	N/A	Download	Download	N/A
10.16	Download				
9.6.21	Download				
9.5.25 (Not Supported)	Download				
9.4.26 (Not Supported)	Download				
9.3.25 (Not Supported)	Download				

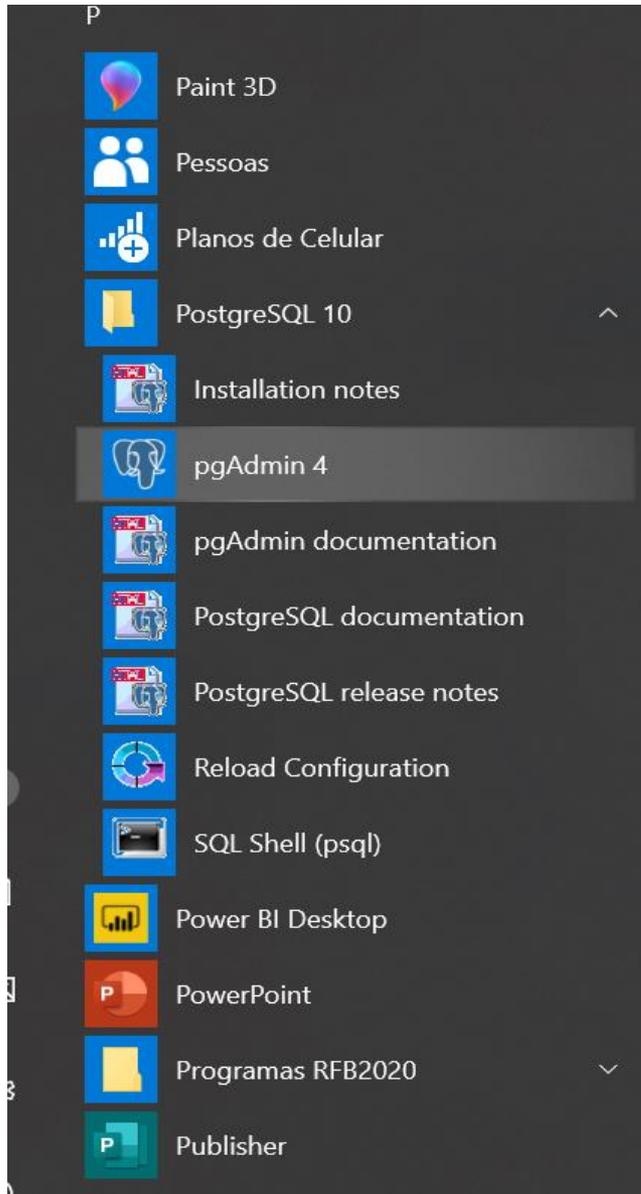
PostgreSQL



O “*Stack Builder*” não tem necessidade de ser instalado.

Atenção, lembre da **senha do banco**. A senha será solicitada sempre que levantar o servidor.

PostgreSQL



Executar o *pgAdmin*, que é uma interface de administração do Postgres

PostgreSQL

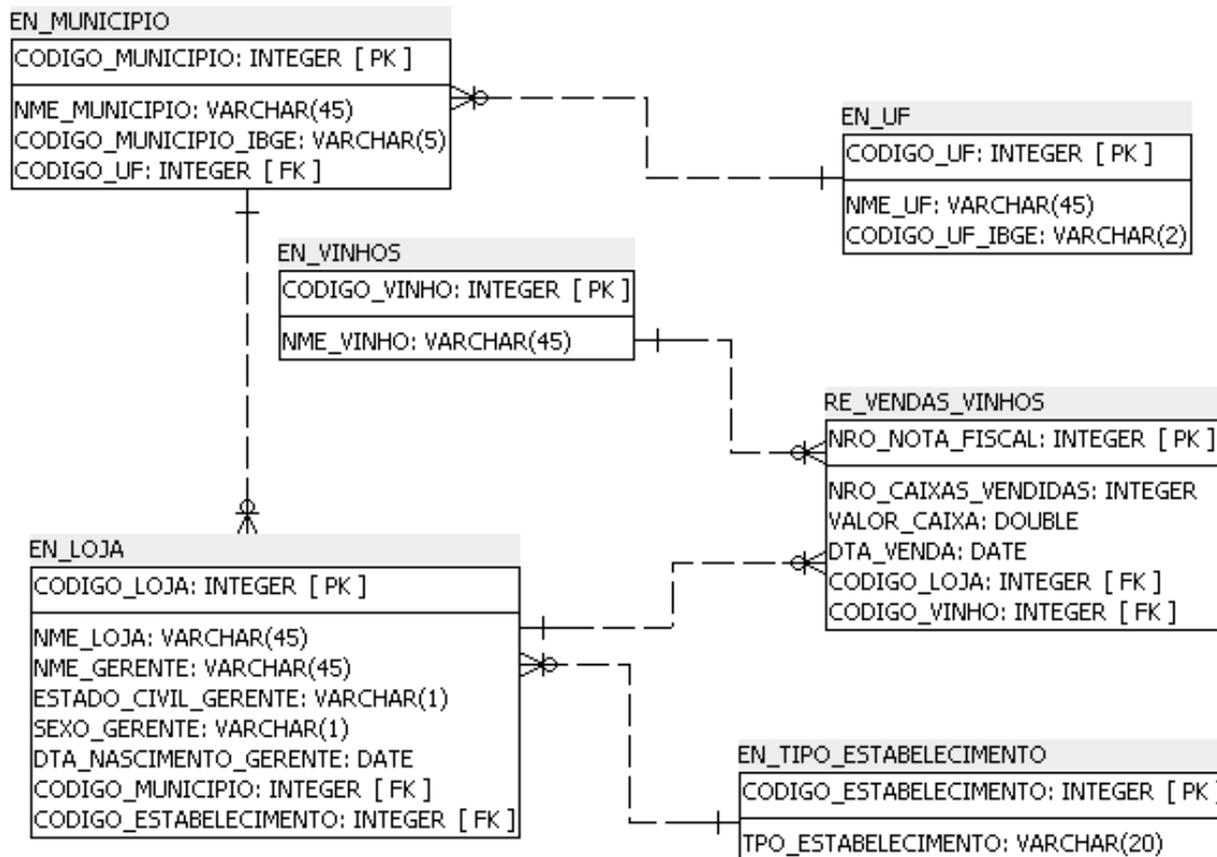
The image shows a browser window displaying the pgAdmin web interface. The browser's address bar shows the URL `127.0.0.1:54091/browser/`. The pgAdmin interface includes a navigation menu with options like 'File', 'Object', 'Tools', and 'Help'. The main content area features a 'Welcome' message with the pgAdmin logo and the text 'Management Tools for PostgreSQL'. A modal dialog box titled 'Unlock Saved Passwords' is overlaid on the interface, containing the following text: 'Please enter your master password. This is required to unlock saved passwords and reconnect to the database server(s)'. Below the text is a text input field labeled 'Password'. At the bottom of the dialog, there are four buttons: a help icon (?), a 'Reset Master Password' button, a 'Cancel' button, and an 'OK' button. The background interface also shows buttons for 'Add New Server' and 'Configure pgAdmin', and a 'Getting Started' section with links to PostgreSQL, pgAdmin Website, Planet PostgreSQL, and Community Support.

Depois de abrir o *pgAdmin*, o banco nos solicita a senha, criada na instalação.

PostgreSQL

Exercício da folha 5.

A empresa de vendas de vinhos BOAVIDA, tem um sistema de acompanhamento das vendas a seus clientes, mostrado no arquivo “*Modelo_Relacional_Vinhos.architect*”.



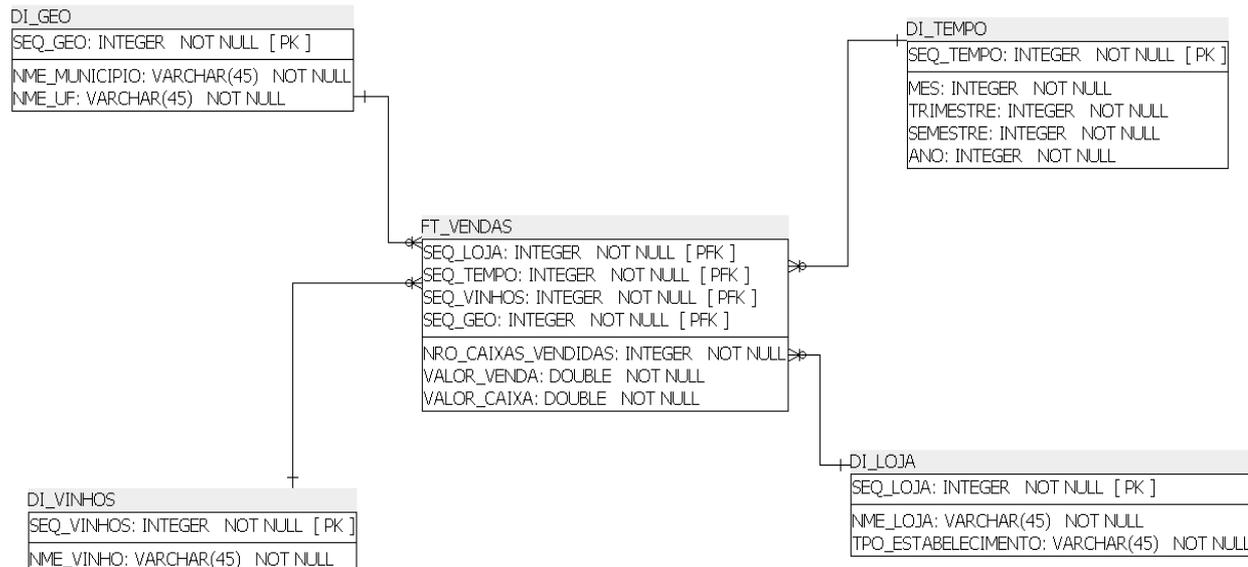
PostgreSQL

Exercício da folha 5.

A empresa BOAVIDA, tem também um modelo dimensional para:

- Acompanhar a evolução mensal das vendas (número de caixas vendidas, valor total das vendas e preço médio da caixa) para cada marca de vinho, para cada loja, tipo de estabelecimento e por município.

O modelo é mostrado no arquivo “*Modelo_Dimesional_Vinhos.architect*”.



PostgreSQL

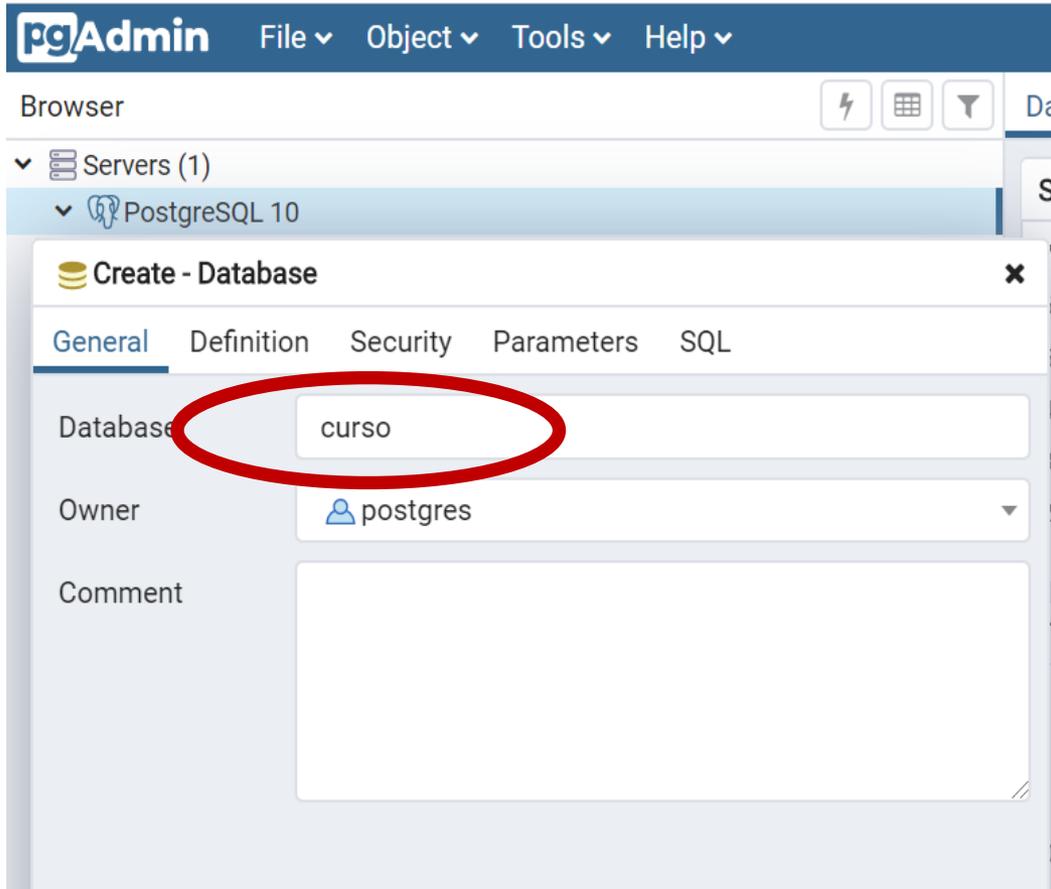
Exercício da folha 5.

The screenshot shows the pgAdmin web interface. In the left-hand 'Browser' pane, the 'Servers' tree is expanded to show a PostgreSQL 10 server. A context menu is open over this server, with the 'Create' option selected. A sub-menu is displayed, showing 'Database...' as the selected option. Other options in the sub-menu include 'Server...', 'Login/Group Role...', and 'Tablespace...'. The main dashboard area contains several performance charts: 'Server sessions' (Total, Active, Idle), 'Transactions per second' (Transactions, Commits, Rollbacks), 'Tuples in' (Inserts, Updates, Deletes), 'Tuples out' (Fetched, Returned), and 'Block I/O' (Reads, Hits). The browser address bar shows '127.0.0.1:54091/browser/'.

1. Criamos uma nova base de dados sobre o servidor local (localhost) do owner “PostgreSQL10”.

PostgreSQL

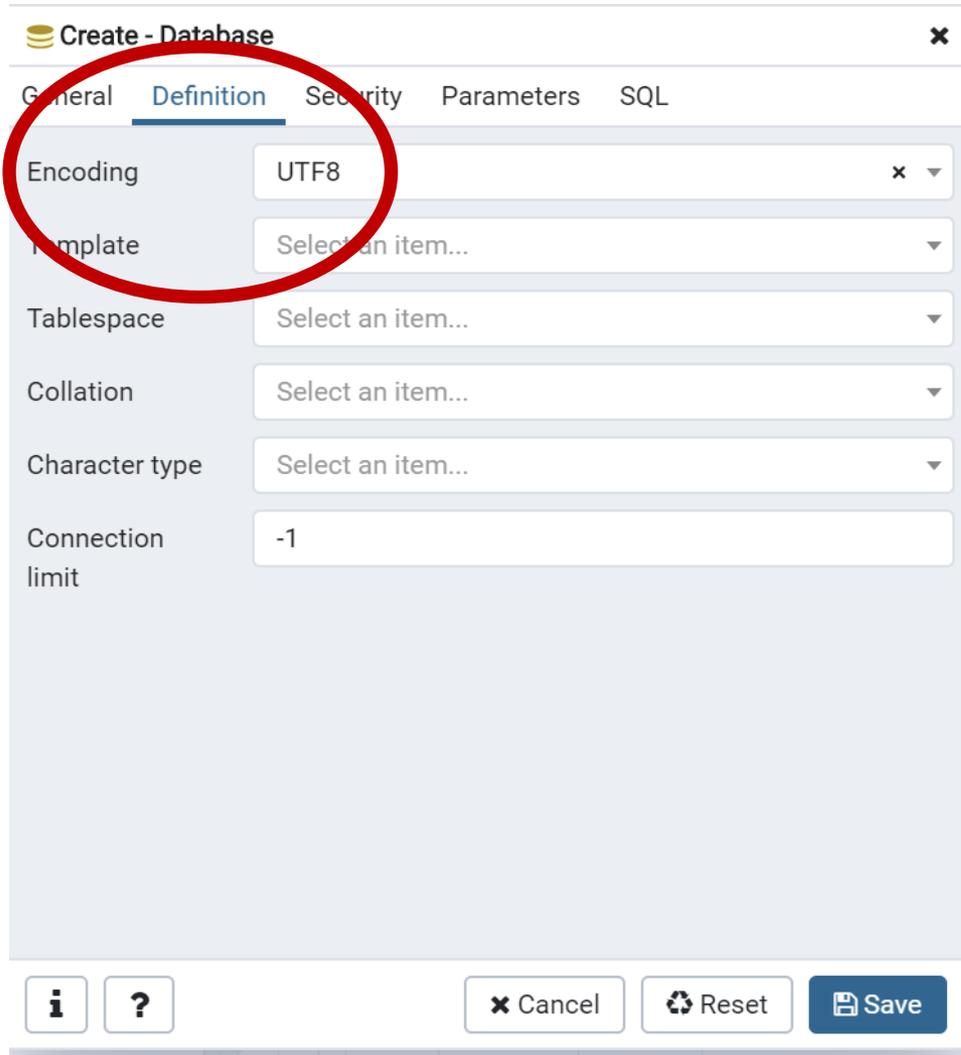
Exercício da folha 5.



2. Escolhemos um nome para a base de dados, por exemplo “curso”.

PostgreSQL

Exercício da folha 5.

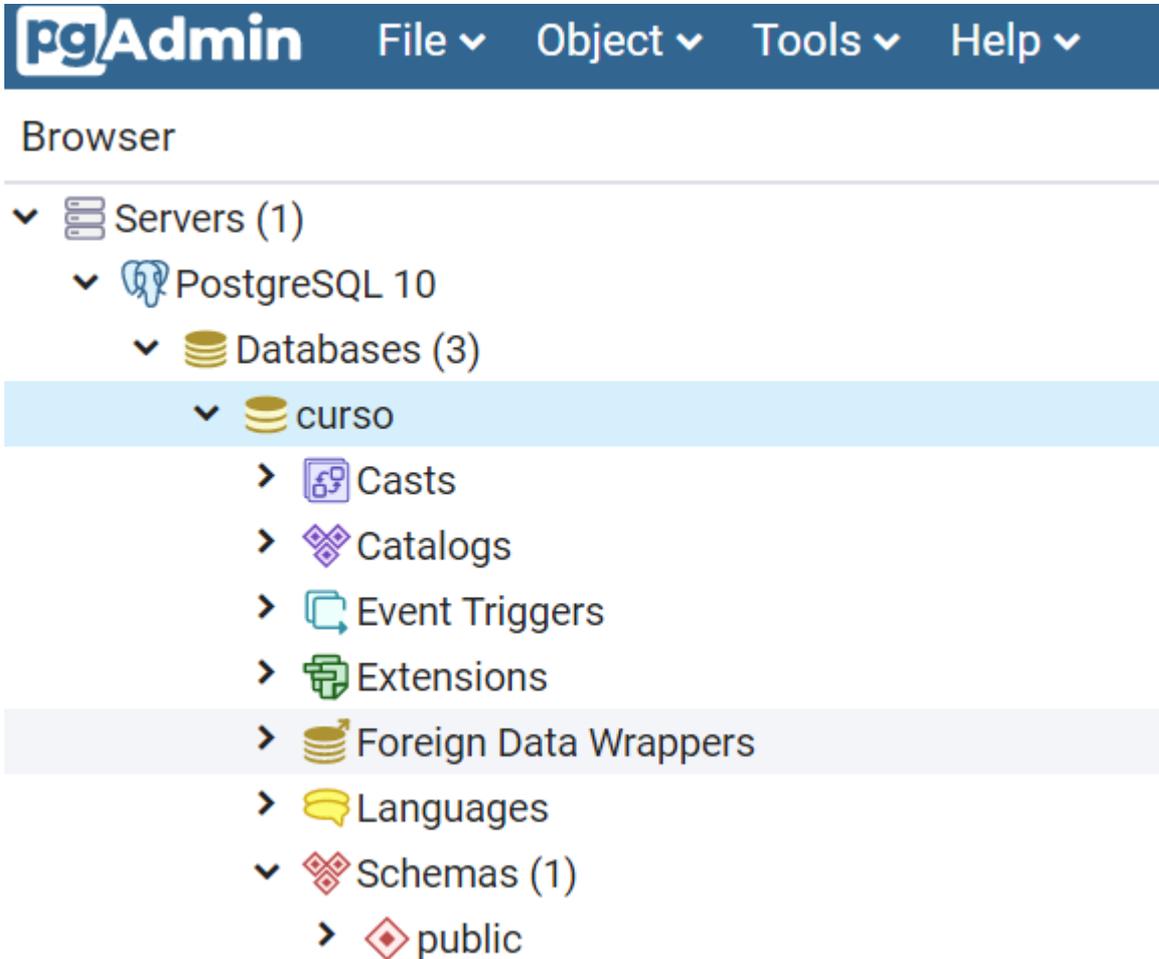


The image shows a screenshot of the 'Create - Database' dialog box in a PostgreSQL client. The dialog has a title bar with a close button (x) and a hamburger menu icon. Below the title bar are five tabs: 'General', 'Definition', 'Security', 'Parameters', and 'SQL'. The 'Definition' tab is selected and highlighted with a blue underline. A red circle is drawn around the 'Definition' tab and the 'Encoding' field. The 'Encoding' field is a dropdown menu with 'UTF8' selected. Below it are four more dropdown menus: 'Template' (with 'Select an item...' text), 'Tablespace' (with 'Select an item...' text), 'Collation' (with 'Select an item...' text), and 'Character type' (with 'Select an item...' text). At the bottom of the dialog is a 'Connection limit' field with the value '-1'. At the very bottom of the dialog are four buttons: an information icon (i), a help icon (?), a 'Cancel' button with a close icon (x), a 'Reset' button with a refresh icon, and a 'Save' button with a save icon.

3. Escolhemos o tipo de codificação, UTF8.

PostgreSQL

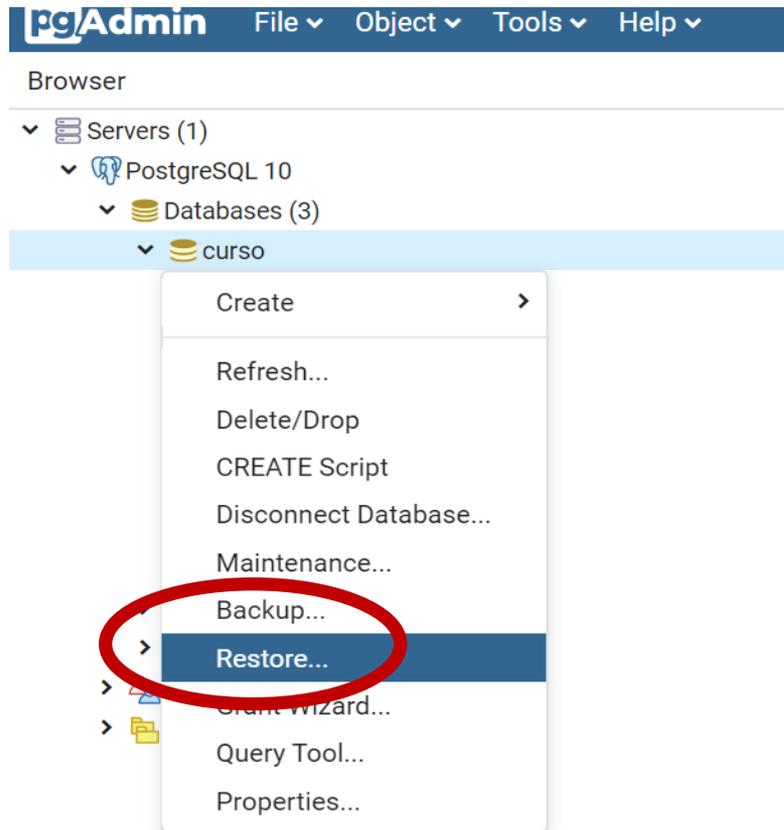
Exercício da folha 5.



4. A base de dados foi criada, com o Schema “public”.

PostgreSQL

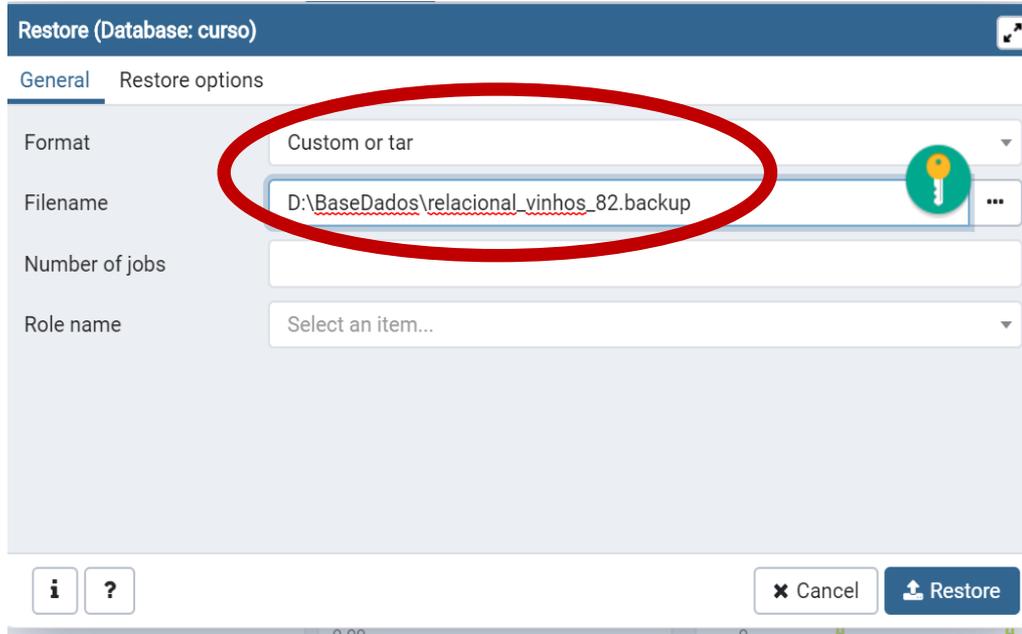
Exercício da folha 5.



5. No Database “curso”, escolhemos a opção “Restore”, para restaurar o banco de dados *relacional_vinhos_82.backup*.

PostgreSQL

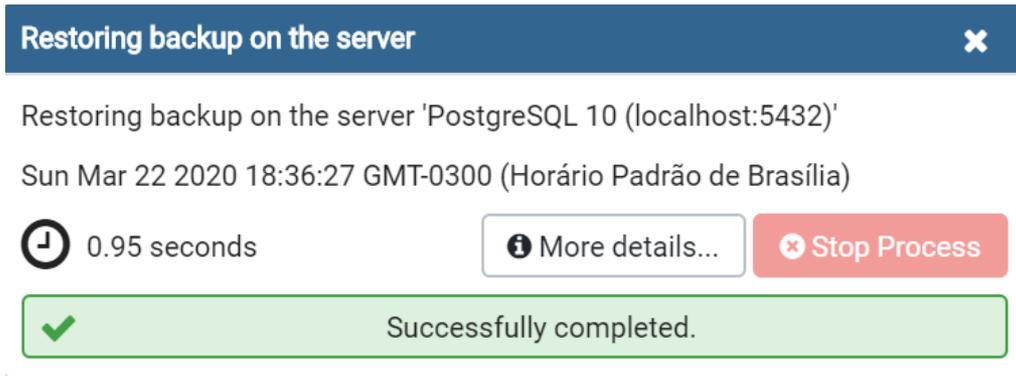
Exercício da folha 5.



6. Escolhemos o arquivo de backup correspondente a base restaurada,

PostgreSQL

Exercício da folha 5.

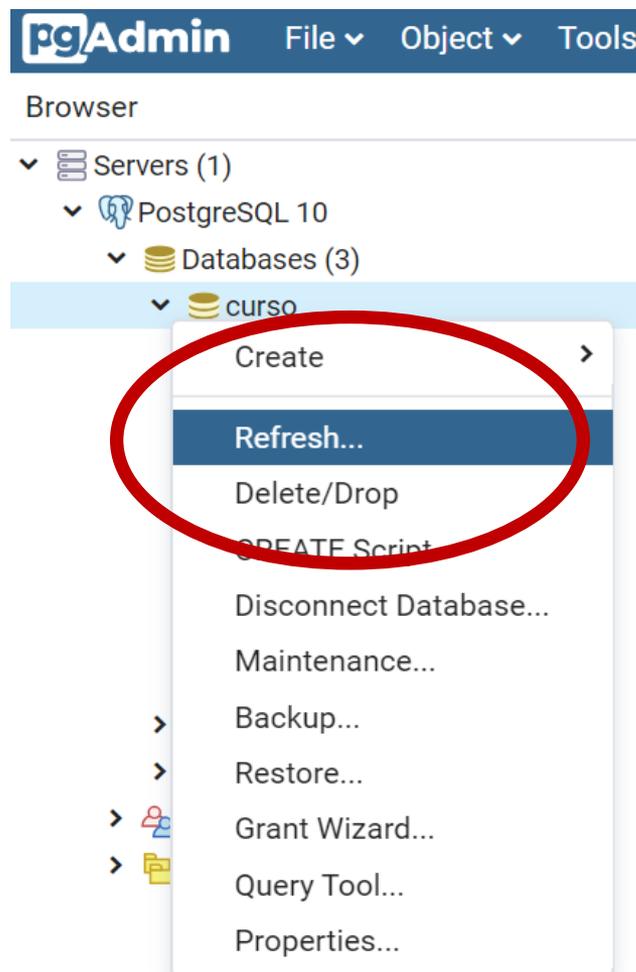


The screenshot shows a notification window titled "Restoring backup on the server" with a close button (X) in the top right corner. The main text reads "Restoring backup on the server 'PostgreSQL 10 (localhost:5432)'" followed by the timestamp "Sun Mar 22 2020 18:36:27 GMT-0300 (Horário Padrão de Brasília)". Below this, a clock icon is followed by "0.95 seconds". To the right are two buttons: "More details..." with an information icon (i) and "Stop Process" with a close icon (X). At the bottom, a green bar with a checkmark icon contains the text "Successfully completed."

7. Verificamos se a restauração foi bem sucedida.

PostgreSQL- Visualizando os dados

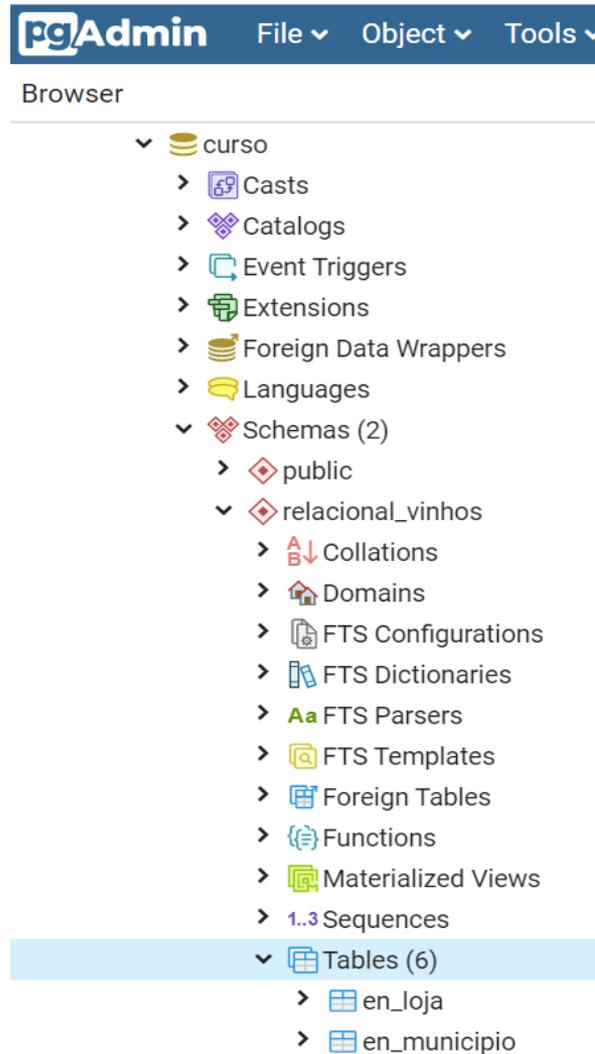
Exercício da folha 5.



8. Se o esquema “relacional_vinhos” não aparece, damos um “*refresh*” no esquema.

PostgreSQL- Visualizando os dados

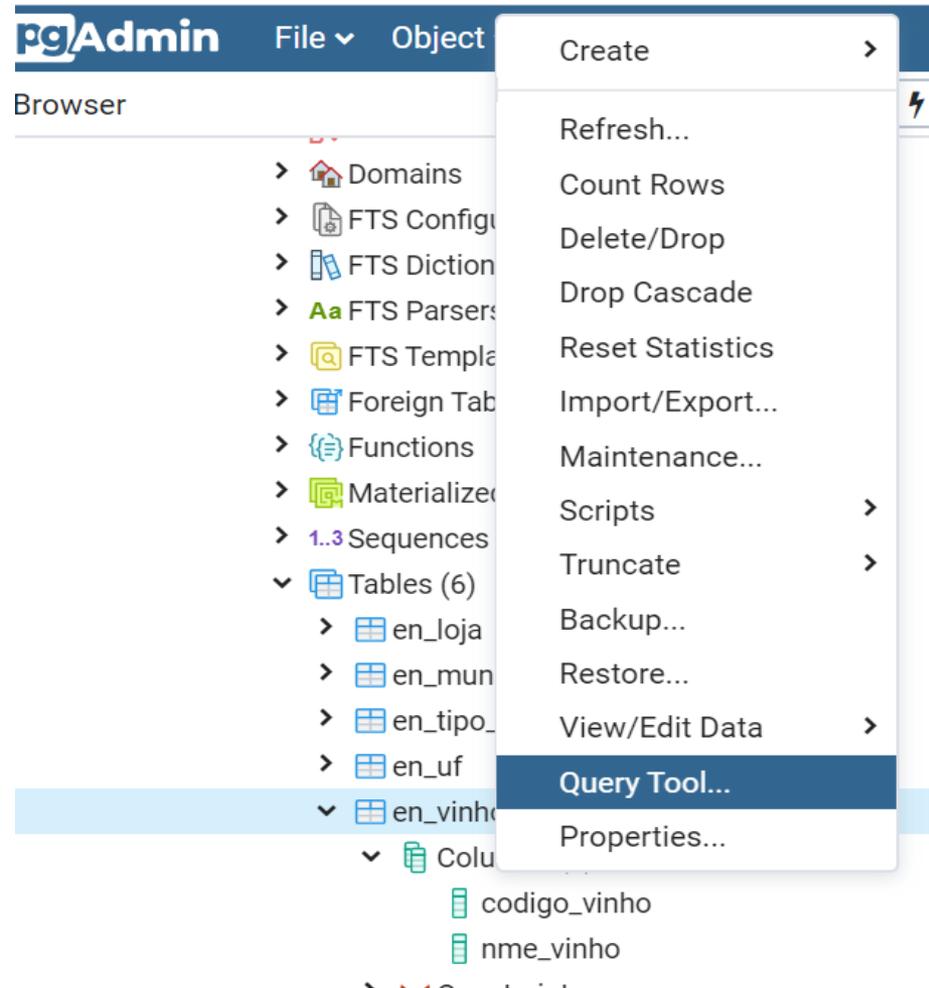
Exercício da folha 5.



9. Agora podemos visualizar o schema e as tabelas do modelo.

PostgreSQL- Visualizando os dados

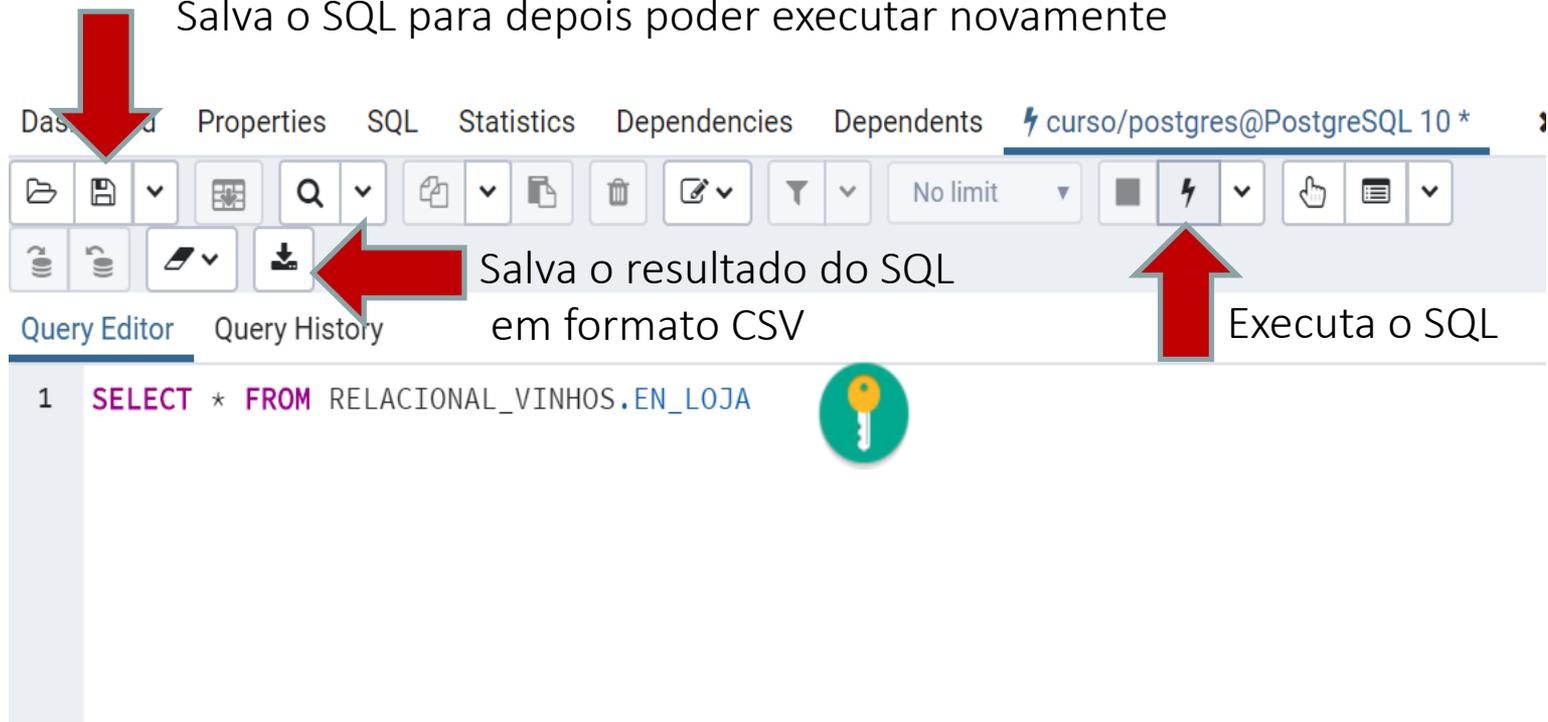
Exercício da folha 5.



10. Podemos fazer consultas SQLs nas tabelas do esquema.

PostgreSQL- Visualizando os dados

Salva o SQL para depois poder executar novamente



Query Editor Query History

Salva o resultado do SQL em formato CSV

Executa o SQL

```
1 SELECT * FROM RELACIONAL_VINHOS.EN_LOJA
```

Data Output Explain Messages Notifications

	codigo_loja [PK] integer	codigo_estabelecimento integer	codigo_municipio integer	nme_loja character varying (45)	nme_gerente character varying (45)
1	1	2	4551	Bigfoot Breweries	João
2	84	1	4551	Hotel Cinco Estrelas	Maria
3	66	3	4549	Escargots Nouveaux	Eduardo
4	263	1	4551	Hotel e Spa Bom Appetite	Ana Paula
5	240	1	4551	Hotel Le Plata	Ernesto

11. Podemos fazer consultas SQLs nas tabelas do esquema.

PostgreSQL- Visualizando os dados

Exercício da folha 5.

- ▼  curso
 - >  Casts
 - >  Catalogs
 - >  Event Triggers
 - >  Extensions
 - >  Foreign Data Wrappers
 - >  Languages
 - ▼  Schemas (3)
 - >  dwvinhos
 - >  public
 - >  relacional_vinhos

12. Restauramos a base `dw_vinhos_82.backup`, usando os mesmos passos realizados para a base `relacional_vinhos.backup`.

PostgreSQL- Visualizando os dados

Exercício da folha 5.

Utilizando os dados do esquema “*relacional_vinhos*” e do esquema “*dwvinhos*”, responder com SQLs:

- Quantas caixas de vinhos foram vendidas por ano, para cada marca (nome) de vinho?
- Quantas caixas de vinho foram vendidas e qual foi o valor obtido com as vendas para cada marca de vinho, por tipo de estabelecimento e por ano? Ordenar por tipo de estabelecimento e por ano.
- Qual foi o valor das vendas por trimestre? Considerar os meses 1,2 e 3 como o primeiro trimestre, 4,5 e 6 o segundo, e assim sucessivamente.
- Qual foi o valor obtido com as vendas por tipo de estabelecimento no município de Florianópolis, no primeiro trimestre do ano de 2009?

PostgreSQL- Visualizando os dados

Exercício da folha 5.

```
-- Relacional SQL A
SELECT
    V.NME_VINHO AS VINHO,  EXTRACT(YEAR FROM R.DTA_VENDA) AS ANO,
    SUM(R.NRO_CAIXAS_VENDIDAS) AS TOTAL_VENDAS
FROM
    RELACIONAL_VINHOS.EN_VINHOS V,  RELACIONAL_VINHOS.RE_VENDAS_VINHOS R
WHERE
    V.CODIGO_VINHO = R.CODIGO_VINHO
GROUP BY
    VINHO, ANO
ORDER BY
    TOTAL_VENDAS DESC
```

PostgreSQL- Visualizando os dados

Exercício da folha 5.

```
-- Dimensional SQL A
SELECT
    DV.NME_VINHO AS VINHO, DT.ANO AS ANO,
    SUM(FT.NRO_CAIXAS_VENDIDAS) AS TOTAL_CAIXAS_VENDIDAS
FROM
    DWVINHOS.FT_VENDAS FT, DWVINHOS.DI_TEMPO DT, DWVINHOS.DI_VINHOS DV
WHERE
    FT.SEQ_VINHOS = DV.SEQ_VINHOS AND
    FT.SEQ_TEMPO = DT.SEQ_TEMPO
GROUP BY
    VINHO, ANO
ORDER BY
    TOTAL_CAIXAS_VENDIDAS DESC
```

PostgreSQL- Visualizando os dados

Exercício da folha 5.

```
-- Relacional SQL B
```

```
SELECT  
    V.NME_VINHO AS VINHO, EXTRACT(YEAR FROM R.DTA_VENDA) AS ANO,  
    E.TPO_ESTABELECIMENTO AS ESTABELECIMENTO, SUM(R.NRO_CAIXAS_VENDIDAS) AS TOTAL_VENDAS,  
    SUM(R.NRO_CAIXAS_VENDIDAS * R.VALOR_CAIXA) AS VALOR_VENDAS  
FROM  
    RELACIONAL_VINHOS.EN_VINHOS V, RELACIONAL_VINHOS.RE_VENDAS_VINHOS R,  
    RELACIONAL_VINHOS.EN_LOJA L, RELACIONAL_VINHOS.EN_TIPO_ESTABELECIMENTO E  
WHERE  
    V.CODIGO_VINHO = R.CODIGO_VINHO AND R.CODIGO_LOJA = L.CODIGO_LOJA AND  
    L.CODIGO_ESTABELECIMENTO = E.CODIGO_ESTABELECIMENTO  
GROUP BY  
    VINHO, ANO, ESTABELECIMENTO  
ORDER BY  
    ESTABELECIMENTO, ANO
```

PostgreSQL- Visualizando os dados

Exercício da folha 5.

```
-- Dimensional SQL B
SELECT
  DV.NME_VINHO AS VINHO, DT.ANO AS ANO, DL.TPO_ESTABELECIMENTO AS ESTABELECIMENTO,
  SUM(FT.NRO_CAIXAS_VENDIDAS) AS TOTAL_CAIXAS_VENDIDAS,
  SUM(FT.VALOR_VENDA) AS TOTAL_VALOR_VENDA
FROM
  DWVINHOS.FT_VENDAS FT, DWVINHOS.DI_TEMPO DT, DWVINHOS.DI_LOJA DL, DWVINHOS.DI_VINHOS DV
WHERE
  FT.SEQ_VINHOS = DV.SEQ_VINHOS AND
  FT.SEQ_TEMPO = DT.SEQ_TEMPO AND
  FT.SEQ_LOJA = DL.SEQ_LOJA
GROUP BY
  VINHO, ANO, ESTABELECIMENTO
ORDER BY
  ESTABELECIMENTO, ANO
```

PostgreSQL- Visualizando os dados

Exercício da folha 5.

-- Relacional SQL C – Opção 1

SELECT

EXTRACT (QUARTER FROM R.DTA_VENDA) AS TRIMESTRE,

SUM(R.NRO_CAIXAS_VENDIDAS*R.VALOR_CAIXA) AS TOTAL_VALOR_VENDA

FROM

RELACIONAL_VINHOS.RE_VENDAS_VINHOS R

GROUP BY

TRIMESTRE

PostgreSQL- Visualizando os dados

Exercício da folha 5.

-- Relacional SQL C – Opção 2

SELECT

CASE

WHEN EXTRACT (MONTH FROM R.DTA_VENDA) IN (1,2,3) THEN 'TRIMESTRE 1'

WHEN EXTRACT (MONTH FROM R.DTA_VENDA) IN (4,5,6) THEN 'TRIMESTRE 2'

WHEN EXTRACT (MONTH FROM R.DTA_VENDA) IN (7,8,9) THEN 'TRIMESTRE 3'

WHEN EXTRACT (MONTH FROM R.DTA_VENDA) IN (10,11,12) THEN 'TRIMESTRE 4'

END AS TRIMESTRE,

SUM(R.NRO_CAIXAS_VENDIDAS*R.VALOR_CAIXA) AS TOTAL_VALOR_VENDA

FROM

RELACIONAL_VINHOS.RE_VENDAS_VINHOS R

GROUP BY

TRIMESTRE

-- Relacional SQL C – Opção 3

```
SELECT 'TRIMESTRE 1' AS TRIMESTRE,  
       SUM(R.NRO_CAIXAS_VENDIDAS*R.VALOR_CAIXA) AS TOTAL_VALOR_VENDA  
FROM RELACIONAL_VINHOS.RE_VENDAS_VINHOS R  
WHERE EXTRACT (MONTH FROM R.DTA_VENDA) IN (1,2,3)  
  
UNION  
  
SELECT 'TRIMESTRE 2' AS TRIMESTRE,  
       SUM(R.NRO_CAIXAS_VENDIDAS*R.VALOR_CAIXA) AS TOTAL_VALOR_VENDA  
FROM RELACIONAL_VINHOS.RE_VENDAS_VINHOS R  
WHERE EXTRACT (MONTH FROM R.DTA_VENDA) IN (4,5,6)  
  
UNION  
  
SELECT 'TRIMESTRE 3' AS TRIMESTRE,  
       SUM(R.NRO_CAIXAS_VENDIDAS*R.VALOR_CAIXA) AS TOTAL_VALOR_VENDA  
FROM RELACIONAL_VINHOS.RE_VENDAS_VINHOS R  
WHERE EXTRACT (MONTH FROM R.DTA_VENDA) IN (7,8,9)  
  
UNION  
  
... /** idêntico aos anteriores para 'TRIMESTRE 4' **/
```

PostgreSQL- Visualizando os dados

Exercício da folha 5.

```
-- Dimensional SQL C
```

```
SELECT
```

```
    DT.TRIMESTRE AS TRIMESTRE,    SUM(FT.VALOR_VENDA) AS TOTAL_VALOR_VENDA
```

```
FROM
```

```
    DWVINHOS.FT_VENDAS FT, DWVINHOS.DI_TEMPO DT
```

```
WHERE
```

```
    FT.SEQ_TEMPO = DT.SEQ_TEMPO
```

```
GROUP BY
```

```
    TRIMESTRE
```

```
ORDER BY
```

```
    TRIMESTRE
```

PostgreSQL- Visualizando os dados

Exercício da folha 5.

```
-- Relacional SQL D
```

```
SELECT
    E.TPO_ESTABELECIMENTO, SUM(R.NRO_CAIXAS_VENDIDAS*R.VALOR_CAIXA) AS TOTAL_VALOR_VENDA
FROM
    RELACIONAL_VINHOS.RE_VENDAS_VINHOS R, RELACIONAL_VINHOS.EN_LOJA L,
    RELACIONAL_VINHOS.EN_TIPO_ESTABELECIMENTO E, RELACIONAL_VINHOS.EN_MUNICIPIO M
WHERE
    R.CODIGO_LOJA = L.CODIGO_LOJA AND
    L.CODIGO_ESTABELECIMENTO = E.CODIGO_ESTABELECIMENTO AND
    EXTRACT (MONTH FROM R.DTA_VENDA) IN (1,2,3) AND
        EXTRACT (YEAR FROM R.DTA_VENDA) = '2009' AND
    L.CODIGO_MUNICIPIO = M.CODIGO_MUNICIPIO AND
    M.NME_MUNICIPIO = 'Florianópolis'
GROUP BY
    E.TPO_ESTABELECIMENTO
```

PostgreSQL- Visualizando os dados

Exercício da folha 5.

```
-- Dimensional SQL D
```

```
SELECT
    DL.TPO_ESTABELECIMENTO AS ESTABELECIMENTO, SUM(FT.VALOR_VENDA) AS TOTAL_VALOR_VENDA
FROM
    DWVINHOS.FT_VENDAS FT, DWVINHOS.DI_TEMPO DT, DWVINHOS.DI_LOJA DL, DWVINHOS.DI_GEO DG
WHERE
    FT.SEQ_LOJA = DL.SEQ_LOJA AND
    FT.SEQ_TEMPO = DT.SEQ_TEMPO AND
    DT.TRIMESTRE = 1 AND DT.ANO = 2009 AND
    FT.SEQ_GEO = DG.SEQ_GEO AND
    DG.NME_MUNICIPIO = 'Florianópolis'
GROUP BY
    ESTABELECIMENTO
ORDER BY
    ESTABELECIMENTO
```

Fundamentação da disciplina

