

PostgreSQL

Readme.rmd

Sergio Pedro R Oliveira

2022-08-23

Contents

1	Objetivo	3
2	Referência	3
3	Aula 117 - Instalação do PostgreSQL, conectando servidor ao pgAdmin 4 e acessando psql	4
3.1	Instalação do PostgreSQL	4
3.2	Conectando pgAdmin 4 ao Servidor	4
3.3	Acessando PostgreSQL pelo terminal - psql	5
3.4	Alterando senha do usuario postgres	5
4	Aula 119 - Primeiros passos pgAdmin4	6
4.1	Acessando um banco de dados	6
4.2	Criando um novo banco de dados	6
4.3	Conectando num banco de dados	6
4.4	Abrindo aba para escrever consulta SQL (Query Tool)	7
5	Aula 120 - datestyle	8
5.1	Padrão de data de sistema	8
5.2	Função datestyle	8
5.3	Configurando um outro padrão de data	8
6	Aula 121 - Abrir arquivo “.sql” no pgAdmin4	10
7	Aula 122 - Introdução a funções de agregação	11
7.1	Teoria	11
7.2	Funções de agregação	11
7.3	<i>Alias</i>	12
7.4	GROUP BY	13
8	Aula 123 - Estatística básica média e soma	14
8.1	Limite de linhas mostradas numa consulta - LIMIT	14
8.2	ORDER BY	14
8.3	Média - AVG	15
8.4	Soma total - SUM	15
9	Aula 124 - Estatística básica médias	16
9.1	Preparação dos dados para aplicação de estatística básica	16
9.2	Média	16
9.3	Mediana	16

9.4	Moda	16
9.5	Amplitude	16
9.6	Variancia	16
9.7	Desvio padrão	16
9.8	Coefficiente de variancia	16
10	Observações	17
10.1	Exportação de dados	17
10.2	Breve explicação de Business Intelligence e Data Science	17
11	Andamento dos Estudos	18
11.1	Assunto em andamento	18

1 Objetivo

Estudo dirigido de **PostgreSQL**.

2 Referência

Vídeo aulas “O curso completo de Banco de Dados e SQL, sem mistérios” - Udemey.

3 Aula 117 - Instalação do PostgreSQL, conectando servidor ao pgAdmin 4 e acessando psql

3.1 Instalação do PostgreSQL

3.1.1 Principais programas

- **PostgreSQL**

É um sistema gerenciador de banco de dados objeto relacional (SGBD), desenvolvido como projeto de código aberto, que pode ser baixado pelo site:

<https://www.postgresql.org/download/>

- **pgAdmin 4**

É uma interface web com o banco de dados. Pode ser baixado pelo site:

<https://www.pgadmin.org/download/>

- **psql**

O psql é um front-end baseado em terminal para o PostgreSQL.

- **Sublime Text**

- Sublime Text é um editor de código-fonte multi-plataforma.
- Ele suporta nativamente muitas linguagens de programação e linguagens de marcação.
- Serve para escrever os script's “.sql”, antes de lançar no banco de dados.

3.2 Conectando pgAdmin 4 ao Servidor

- Primeiro após fazer as instalações, ao abrir o **pgAdmin 4**, o programa vai pedir para registrar uma senha para proteção do sistema.
- Antes de adicionar o novo servidor no **pgAdmin 4**, é necessário mudar a senha do PostgreSQL, acessando ele pelo terminal, pelo **psql**.

- Assim se torna necessário abrir o terminal e acessar o psql:

```
sudo -u postgres psql  
senha_ sudo
```

- Para mudar a senha do usuario postgres, basta digitar o comando:

```
ALTER USER postgres PASSWORD 'novo_password'
```

- Após a mudança da senha, podemos registrar o novo servidor no **pgAdmin 4**.
 - Clickar com o botão esquerdo em “servers” > “Register” > “server”.
 - Na aba “General”, basta adicionar um nome para o server.
“localhost” [nome mais comum]

- Na aba “Connection” é necessário preencher:
 - * Hostname: “localhost”
 - * Port: 5432
 - * Maintenance database: postgres
 - * Username: postgres
 - * Password: [repetir a senha cadastrada anteriormente no psql]
- Ao clicar em “**Salvar**” o novo servidor estará conectado.

3.3 Acessando PostgreSQL pelo terminal - psql

- Para acessar o **PostgreSQL** pelo terminal do **UBUNTU** o comando é:
`sudo -u postgres psql`
senha_sudo

3.4 Alterando senha do usuario postgres

- O comando para alterar usuário e senha no Postgres pelo terminal é:
ALTER USER postgres **PASSWORD** ‘*novo_password*’
- Este comando é útil para conectar o servidor a interface *pgAdmin4*, pois necessita criar uma senha para o usuário *postgres*.

4 Aula 119 - Primeiros passos pgAdmin4

4.1 Acessando um banco de dados

- Para acessar um dos bancos de dados, basta abrir o programa **pgAdmin 4**.
- Inserir a senha de proteção do programa.
- Clickar dentro aba lateral “**Browser**” na opção **Servers** para se conectar ao servidor.
- Inserir a senha do **servidor**.
- Assim, será mostrado o nome do servidor, expandindo ele, será mostrado os bancos de dados que nele estão contidos.
- Entre os bancos de dados disponiveis o “*postgres*” é o bando de dados reservado do sistema.
 - o *postgres* é o nome do root do sistema **PostgreSQL**.

4.2 Criando um novo banco de dados

- Na aba lateral “**Browser**”, nas opções **Servers > localhost > Databases**.
- Para criar um novo banco de dados:
 - Clickar na opção **Databases** com o botão direito.
 - Seguir as opções: **Create > Database**.
 - Preencher as opções na aba “**General**”:
 - * **Database:** [Nome do banco de dados]
 - * **Owner:** [Responsavel pelo banco de dados]
 - * **Comment:** [Comentario/resumo sobre o banco de dados, um texto]
 - * **Save** para criar o banco de dados.
- O novo banco de dados e suas pastas estara disponivel na aba lateral **Browser**, dentro de **Databases**.

4.3 Conectando num banco de dados

- Para se conectar a um banco de dados, basta clicar nele na aba lateral “**Browser**”.
- Para verificar em qual banco de dados esta conectado:

- Dentro da aba superior **Dashboard** > na parte inferior da janela, nas opções:
 - * **User** informa o usuário logado, no momento.
 - * **Application** informa o banco de dados que esta conectado, no momento.

4.4 Abrindo aba para escrever consulta SQL (Query Tool)

- **Query Tool** é a aba na qual se escreve as instruções SQL.
- Na aba superior, na opção **TOOLS** > **Query Tool**, abre a aba para escrever as instruções **SQL**.

5 Aula 120 - datestyle

5.1 Padrão de data de sistema

- O padrão de data do sistema é:
'DD/MM/YYYY', **DMY**.

5.2 Função datestyle

- É uma função que mostrar o padrão de data (**DATE**) em que o sistema esta configurado.
- Sintaxe:
SHOW DATESTYLE;

5.3 Configurando um outro padrão de data

- No ubuntu:
 - Na pasta:
/etc/postgresql/14/main/
 - No arquivo “/postgresql.conf”, onde ficam guardadas as configurações do PostgreSQL.
 - Basta abrir com editor de texto (Sublime text, Notepad++, ...) e procurar por “datestyle”.
 - Para alterar o padrão basta mudar a arrumação das letras e salvar o arquivo.
 - Dado que **dmy** é:
 - * **d** é day
 - * **m** é month
 - * **y** é year
 - Lembrar de salvar comentado em baixo a configuração original antes salvar uma alteração.
 - Reiniciar o servidor (computador), para implementar as mudanças.
- No windows:
 - Na pasta:
C:/Arquivos de Programas/PostgreSQL/14[*Numero da versão do PostgreSQL*]/data/
 - No arquivo “/postgresql.conf”, onde ficam guardadas as configurações do PostgreSQL.
 - Basta abrir com editor de texto (Sublime text, Notepad++, ...) e procurar por “datestyle”.
 - Para alterar o padrão basta mudar a arrumação das letras e salvar o arquivo.
 - Dado que **dmy** é:
 - * **d** é day

- * **m** é month
- * **y** é year
- Lembrar de salvar comentado em baixo a configuração original antes salvar uma alteração.
- Reiniciar o servidor, para implementar as mudanças.
 - * Para reiniciar o servidor, no “executar”, digitar “serviços” e clicar na opção de programa “SERVIÇOS”.
 - * Dentro de “SERVIÇOS”, o programa vai mostrar todos os serviços do **WINDOWS**, procurar pelo “PostgreSQL”.
 - * Selecionar o “PostgreSQL” e clicar em “reiniciar o serviço”.
 - * Voltar no **pgAdmin 4** dar “refresh” na tabela, ou servers.
 - * Caso a conexão não esteja estabelecida, basta clicar em “**Query Tool**” para restabeler nova conexão.

6 Aula 121 - Abrir arquivo “.sql” no pgAdmin4

- Ao iniciar o programa **pgAdmin4**, abrir a aba **Query Tools** de programação **SQL**.
- Com a aba “**Query Tools**” aberta, clicar na opção “**Open File**”, navegar pelas pastas e selecionar o arquivo com extensão “.sql” para abrir.
- O arquivo será aberto na aba “**Query Tools**”.

7 Aula 122 - Introdução a funções de agregação

7.1 Teoria

- O que são funções de agregação?
 - Funções de agregação são funções SQL que permitem executar uma operação aritmética nos valores de uma coluna em todos os registros de uma tabela.
 - Uma função de agregação executa um cálculo em um conjunto de valores e retorna um único valor.
 - As funções de agregação frequentemente são usadas com a cláusula **GROUP BY** da instrução **SELECT**.
 - As funções de agregação agregam, somam e resumem registros, o que é apreciado em *data science*.

7.2 Funções de agregação

- **AVG()**
 - Calcula a média aritmética sobre o conjunto de linhas fornecido.
 - Retorna a média aritmética dos valores dos registros.
 - Sintaxe:
SELECT
setor,
AVG(salario) AS “MEDIA DE SALARIO”
FROM *tabela*
GROUP BY *setor*;
- **COUNT()**
 - Essa função retorna o número de itens encontrados em um grupo.
 - Com exceção da função **COUNT(*)**, as funções de agregação ignoram valores nulos.
 - Sintaxe:
SELECT
setor,
COUNT(nome) AS “NUMERO FUNCIONARIOS”
FROM *tabela*
GROUP BY *setor*;
ou
SELECT
COUNT(*) AS “NUMERO DE REGISTROS”
FROM *tabela*;
- **MIN()**
 - Retorna o valor Mínimo de um conjunto de valores.
 - Sintaxe:
SELECT

```

    setor,
    MIN(salario) AS "MENOR SALARIO DO SETOR"
FROM tabela
GROUP BY setor;

```

- **MAX()**

- Retorna o Valor máximo de um conjunto de valores.

- Sintaxe:

```

SELECT
    setor,
    MAX(salario) AS "MAIOR SALARIO DO SETOR"
FROM tabela
GROUP BY setor;

```

- **SUM()**

- Total (Soma) de um conjunto de valores.

- Sintaxe:

```

SELECT
    setor,
    SUM(salario) AS "TOTAL DE SALARIOS DO SETOR"
FROM tabela
GROUP BY setor;

```

7.3 *Alias*

- Um *alias* de coluna permite atribuir um nome temporário a uma coluna ou expressão na lista de projeção de uma instrução **SELECT**.
 - O *alias* da coluna existe temporariamente durante a execução da consulta.
 - É principalmente importante colocar *alias* em colunas que levam formulas, para facilitar o entendimento de quem vai ler a consulta.
 - Sintaxe:
- ```

SELECT
 AVG(coluna1) AS "ALIAS"
...

```

## 7.4 GROUP BY

- A cláusula **GROUP BY** divide as linhas retornadas da instrução **SELECT** em grupos.
- Para cada grupo, você pode aplicar uma função agregada, por exemplo, **SUM()** para calcular a soma dos itens ou **COUNT()** para obter o número de itens nos grupos.
- A cláusula de instrução divide as linhas pelos valores das colunas especificadas na cláusula **GROUP BY** e calcula um valor para cada grupo.
- O **PostgreSQL** avalia a cláusula **GROUP BY** após as cláusulas **FROM** e **WHERE** e antes das cláusulas **HAVING SELECT**, **DISTINCT**, **ORDER BY** e **LIMIT**.



- Sintaxe:  
**SELECT** *Country, Region, SUM(sales) AS "Total Sales"*  
**FROM** *Sales*  
**GROUP BY** *Country, Region;*

## 8 Aula 123 - Estatística básica média e soma

### 8.1 Limite de linhas mostradas numa consulta - LIMIT

- O comando **LIMIT** determina a quantidade máxima de linhas/registros que serão mostrados de uma determinada consulta.
- O comando vem acompanhado do número de linhas da visualização da consulta.
- Sintaxe:  
**SELECT \* FROM *tabela***  
**LIMIT 10;**

### 8.2 ORDER BY

- A palavra-chave **ORDER BY** é usada para classificar o conjunto de resultados em ordem crescente ou decrescente.
- A ordem na qual as linhas são retornadas em um conjunto de resultados não é garantida, a menos que uma cláusula **ORDER BY** seja especificada.
- **ORDER BY** organiza os resultados de acordo com uma ou mais colunas da tabela, podendo definir a ordem dos resultados como crescente ou decrescente.
  - **ASC**  
Classifica os registros em ordem crescente.
  - **DESC**  
Classifica os registros em ordem decrescente.
- A palavra-chave **ORDER BY** classifica os registros em ordem crescente por padrão. Para classificar os registros em ordem decrescente, use a palavra-chave **DESC**.
- Várias colunas de classificação podem ser especificadas. Os nomes de coluna devem ser exclusivos. A sequência das colunas de classificação na cláusula **ORDER BY** define a organização do conjunto de resultados classificado. Ou seja, o conjunto de resultados é classificado pela primeira coluna e então essa lista ordenada é classificada pela segunda coluna e assim por diante.
- É possível ao invés de especificar o nome do campo/coluna no **ORDER BY**, substituir pela posição em que a coluna aparece na cláusula **SELECT**. Porém não é entendida por outros bancos de dados e usuários com tanta facilidade quanto com a especificação do nome de coluna real. Além disso, as alterações na lista de seleção, como a alteração da ordem das colunas ou a adição de novas colunas, exigirão a modificação da cláusula **ORDER BY** para evitar resultados inesperados.
- Sintaxe com exemplo:  
**SELECT \* FROM *Customers***  
**ORDER BY *Country* ASC, *CustomerName* DESC;**

### 8.3 Média - AVG

- A função **AVG()** retorna a média dos valores em um grupo.
- Ignora valores nulos.
- Sintaxe:  
**SELECT**  
**AVG**(*preco*) **AS** "PRECO\_MEDIO"  
**FROM** *produto*;

### 8.4 Soma total - SUM

- A função **SUM()** retorna a soma de todos os valores ou somente os valores **DISTINCT** na expressão.
- **SUM()** pode ser usado exclusivamente com colunas numéricas.
- Valores nulos são ignorados.
- Sintaxe:  
**SELECT**  
*nome*,  
**SUM**(*valor*) **AS** "TOTAL\_RECEBIDO"  
**FROM** *produto*  
**GROUP BY** *id*;

## **9 Aula 124 - Estatística básica médias**

**9.1 Preparação dos dados para aplicação de estatística básica**

**9.2 Média**

**9.3 Mediana**

**9.4 Moda**

**9.5 Amplitude**

**9.6 Variância**

**9.7 Desvio padrão**

**9.8 Coeficiente de variância**



## 10 Observações

### 10.1 Exportação de dados

- Uma das maneiras mais fácil de exportar dados é através da extensão “.csv”.
- O **PostgreSQL** oferece opções para facilmente exportar dados em “.csv”.
- Passo a passo:
  - Basta fazer a consulta que deseja exportar, pela aba “**Query Tools**”.
  - Lembrando de colocar *alias* nas colunas/campos que levam funções, para melhor entendimento de quem for fazer a leitura do arquivo exportado.
  - Na janela em que aparece o resultado da consulta, tem a aba “Data Output” (na qual, por default, já é a aba em que aparecem os resultados das consultas), tem o ícone “*Save results to file*”.
  - Ao clicar no ícone “*Save results to file*”, é oferecido a opção de salvar a consulta como “.csv”.

### 10.2 Breve explicação de Business Intelligence e Data Science

- Business Intelligence (BI):
  - Esta preocupado com entender o que aconteceu no passado.
- Data Science:
  - Através dos dados, tentar prever tendências futuras.

## **11 Andamento dos Estudos**

### **11.1 Assunto em andamento**

Atualmente estou estudando Módulo 30 - AULA 124.