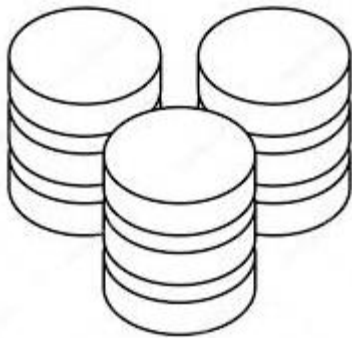


Banco de Dados II

Aspectos de Segurança em SQL



Profa. Damires Souza
damires@ifpb.edu.br



Aspectos de Segurança de Dados

- **Área ampla:**

- Questões legais, éticas, políticas
- Questões relacionadas **ao software**
 - **Nível da aplicação**
 - **Nível do BD**
- Necessidade de identificar níveis de segurança e de **categorizar os dados, usuários e privilégios**
- Auditorias



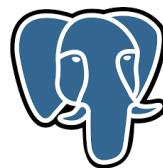
O que pode ser feito com SQL?

- i. **Criação de contas de usuários**
 - Papéis e privilégios
- ii. **Controle de acesso a objetos**
 - Controle de **níveis de privilégios/permissions** sobre objetos
 - Uso de **views** e **triggers**

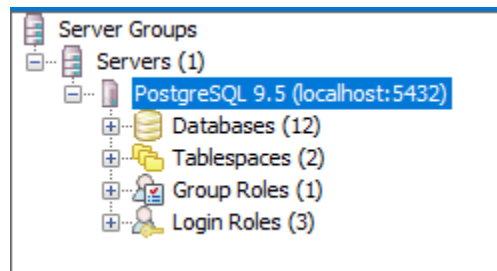
SQL não é só manipulação de dados



Usuários



- Superusuário **postgres**
 - Conexão inicial



CREATE ROLE name [[WITH] option [...]]

Onde *option* pode ser:

SUPERUSER | NOSUPERUSER |

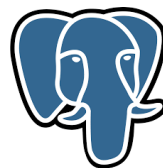
CREATEDB | NOCREATEDB |

CREATEROLE | NOCREATEROLE |

INHERIT | NOINHERIT |

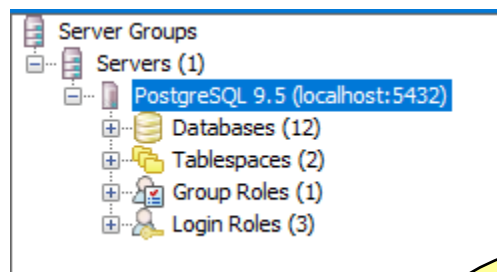
LOGIN | NOLOGIN

Usuários



- Superusuário **postgres**

- Conexão inicial



Crie um usuário com suas iniciais.

** Criação de role/usuário

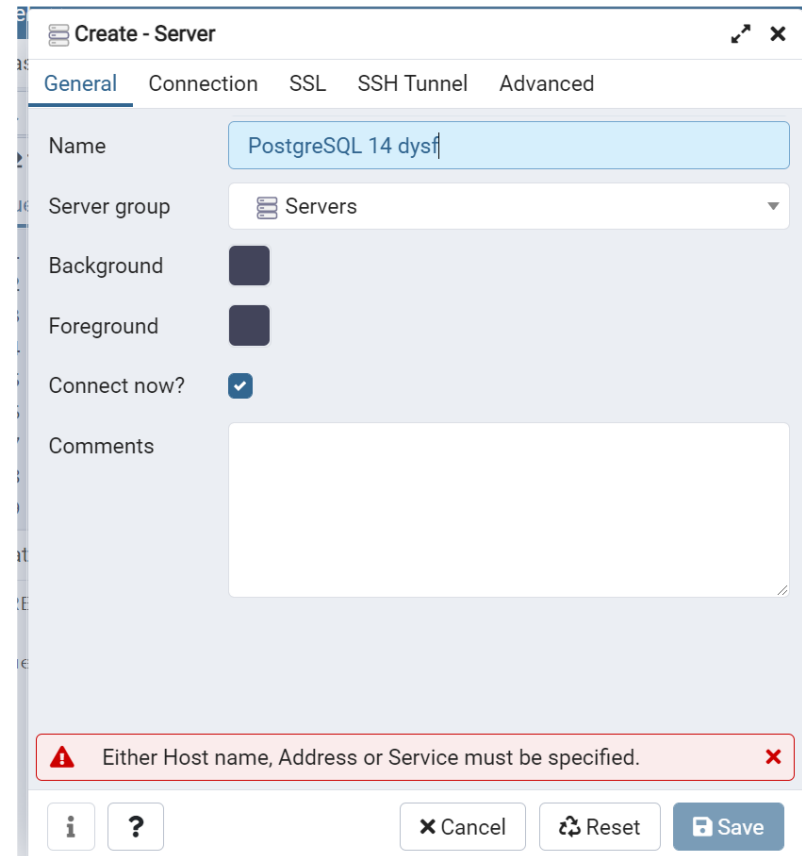
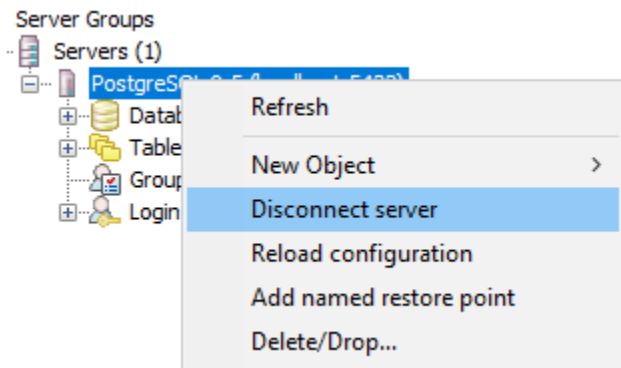
```
CREATE ROLE dysf LOGIN  
PASSWORD 'bd2'  
SUPERUSER CREATEDB CREATEROLE ;
```

```
** ALTER ROLE  
** DROP ROLE
```



Login com usuário novo

- Desconectar do **postgres**
- Logar com **dysf**



Login com usuário novo

Create - Server

General Connection SSL SSH Tunnel Advanced

Host name/addresslocalhost

Port5432

Maintenance databasepostgres

Usernamedysf

Kerberos authentication?False

Password...

Save password?☒

Roledysf

Service

?

Cancel

Reset

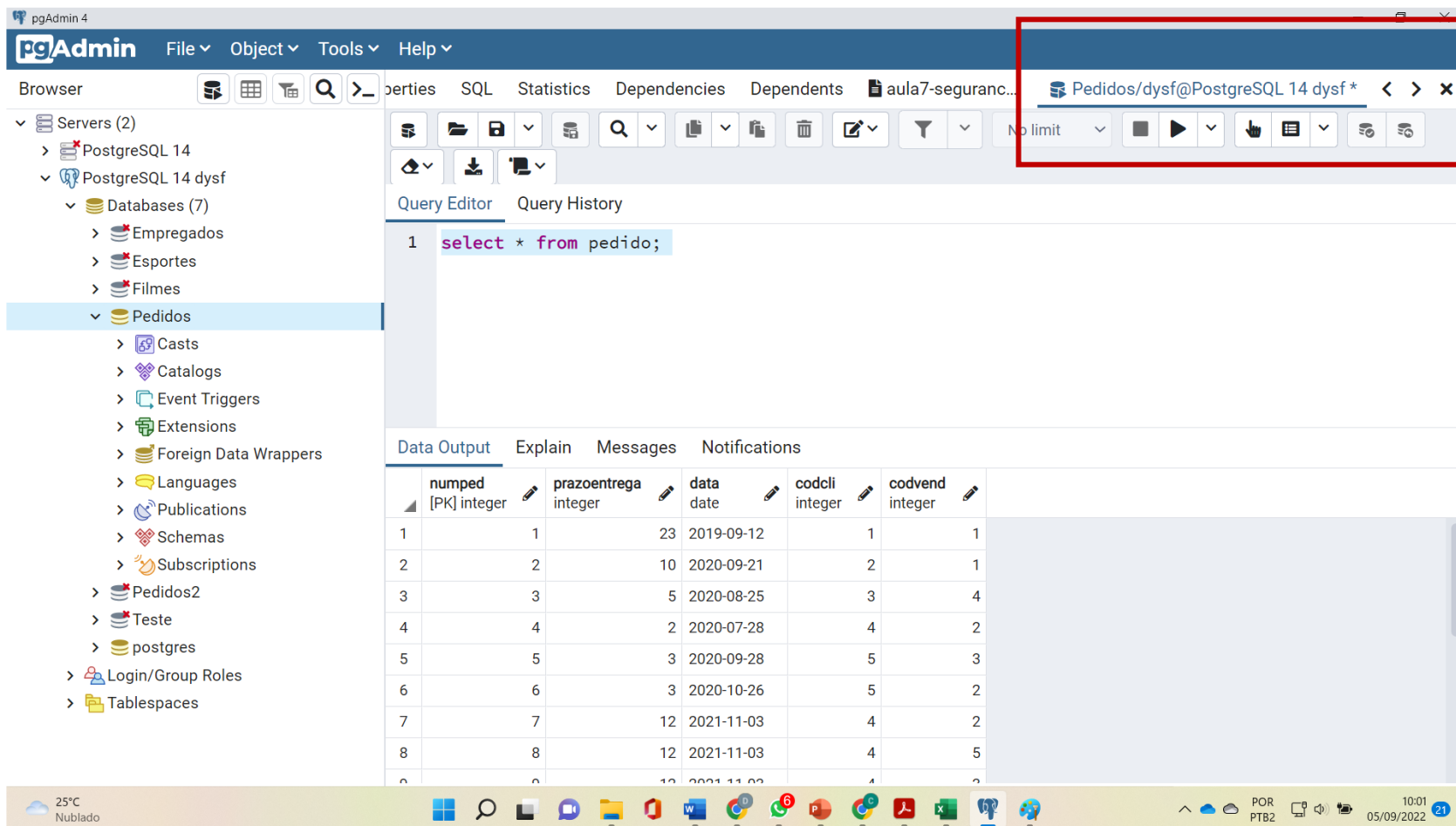
Save

Servers (2)

> PostgreSQL 14

> PostgreSQL 14 dysf

Login com usuário novo



The screenshot shows the pgAdmin 4 interface. The left sidebar displays the database structure, with the 'Pedidos' database selected. The main window shows the Query Editor with the SQL query: `select * from pedido;`. The Data Output tab displays the results of the query, showing 8 rows of data. A red box highlights the top right corner of the interface, which contains the tab bar and the toolbar.

	numped [PK] integer	prazoentrega integer	data date	codcli integer	codvend integer
1	1	23	2019-09-12	1	1
2	2	10	2020-09-21	2	1
3	3	5	2020-08-25	3	4
4	4	2	2020-07-28	4	2
5	5	3	2020-09-28	5	3
6	6	3	2020-10-26	5	2
7	7	12	2021-11-03	4	2
8	8	12	2021-11-03	4	5

Esquema público



Novo BD



Usuários

- Quando um novo BD é criado, por padrão, o Postgres cria um esquema público para ele

Vamos pensar em **privilégios** agora...



SQL: Comando GRANT

```
GRANT <privilégio(s)> [coluna(s)] | ALL  
ON <tabela | view | function>  
TO <role | public> [with Grant option] ;
```

Public: todos os papéis;

With grant option: permite que o usuário que recebeu os privilégios conceda-os a outros; se forem retirados os privilégios concedidos com esta opção, eles também serão removidos dos usuários que os receberam

```
GRANT Select ON Produto TO dysf;
```

```
GRANT All privileges ON Cliente TO public;
```

```
GRANT all on pedido to dysf WITH GRANT OPTION;
```

```
GRANT update(valor) on produto to dysf;
```

SQL: Comando REVOKE

```
REVOKE <privilégios> ON <tabela/view/function>  
FROM <usuário(s)>;
```

Revoke select on produto from dysf;

Revoke select on cliente from public;

Revoke insert, update on pedido from dysf;

Exemplo

➤ [dysf] Criar tabela **CIDADE**

cidade	
General	Columns
Name	cidade
Owner	dysf2
Schema	public

➤ [postgres] Criar novo usuário **bd2**;

Create role **bd2**

Login Password 'bd2';

- ▼ Login/Group Roles (16)
- bd2
 - dysf

Exemplo - teste

dysf:

Grant select on cidade to bd2;

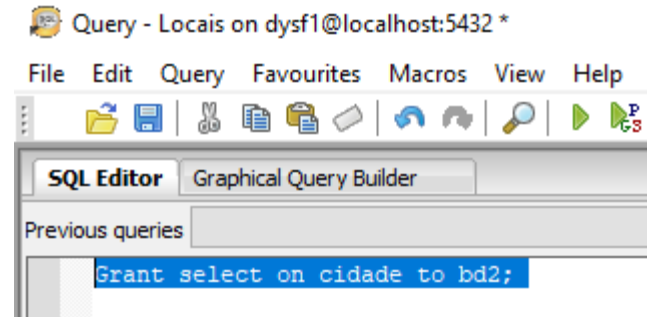
**Bd2: select * from cidade;
insert into cidade
values (2,'Campina Grande');**

dysf:

Grant insert on cidade to bd2;

dysf:

[Revoke select on cidade from bd2;]



Segurança com View

BD Pedidos/usuário postgres

- Útil para simplificar o uso do sistema e para **melhorar a segurança**

Exemplo: Criação e uso da view **CientesVIP**

ATENÇÃO

****** antes de criar a view ***insira 5 pedidos*** feitos pela cliente **Maria Portela**

Segurança com View

Criação e uso da view **ClientesVIP**

postgres:

create or replace view clientesVIP as

SELECT c.nome as "VIP"

FROM cliente c join pedido p on c.codcli = p.codcli

Group by c.nome

Having count(*) > 2;

postgres: select * from ClientesVIP;

postgres: Grant select on clientesVIP to dysf,bd2;