



SQL

DEVinHouse

Parcerias para desenvolver a sua carreira

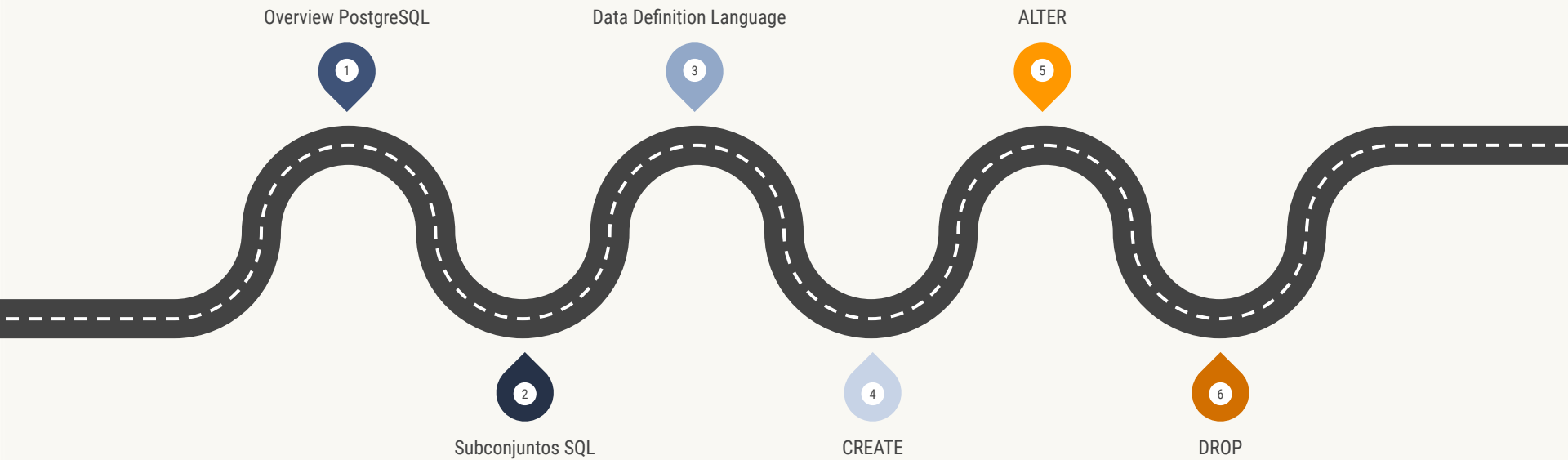
SENAI

<LAB365>

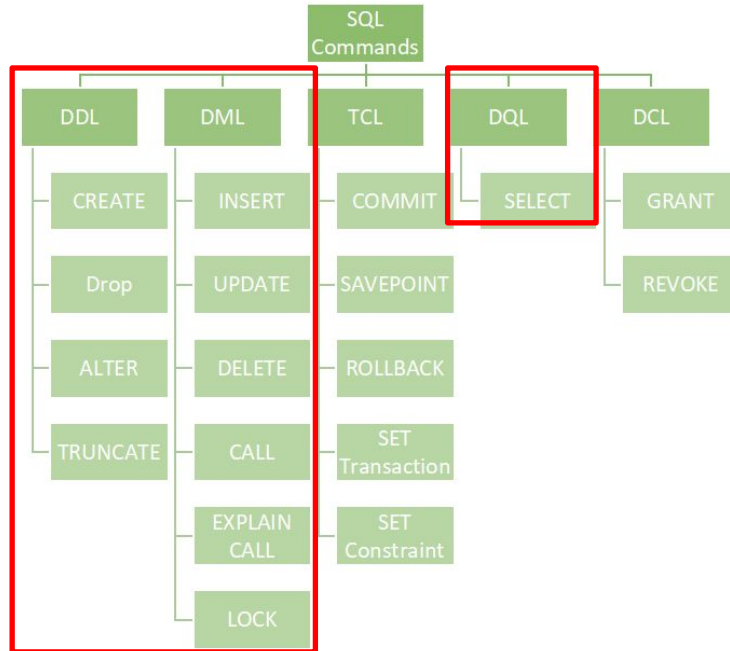
OBSERVAÇÕES

- A interação é crucial para um melhor entendimento;
- Qualquer dúvida, favor levantar a mão ou enviar no chat;
- Caso queira, pode enviar sua dúvida via Slack;
- Utilize os materiais complementares;

AGENDA



SUBCONJUNTO DO SQL



DATA DEFINITION LANGUAGE



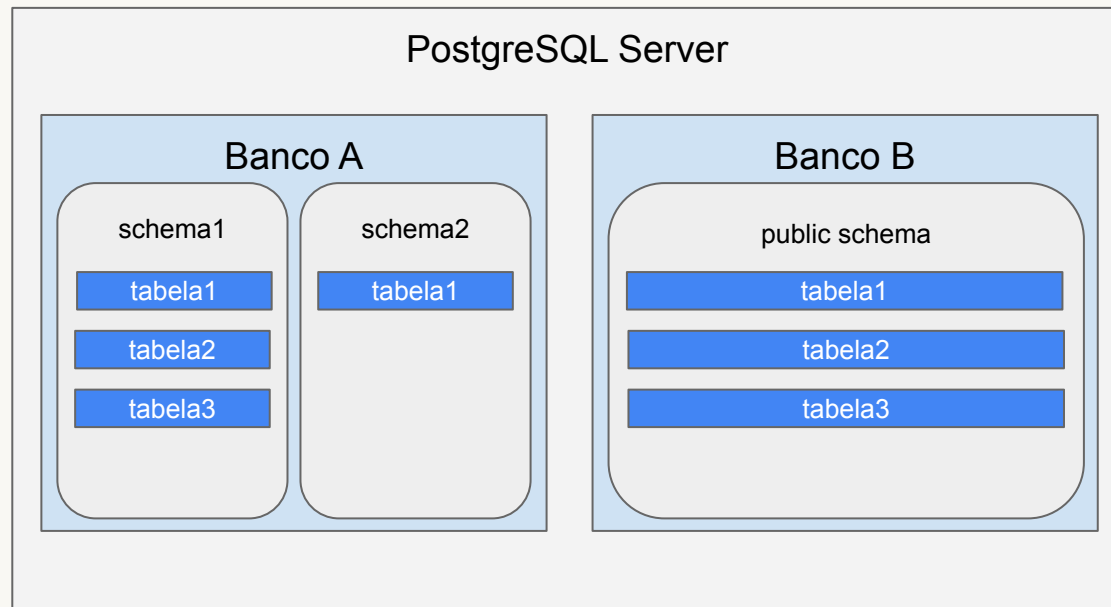
DEVinHouse

Parcerias para desenvolver a sua carreira

SENAI

<LAB365>

SCHEMA



CREATE

- Para criarmos tabelas no banco de dados será necessário realizar a declaração CREATE TABLE, como podemos observar abaixo:

```
CREATE TABLE schema_name.table_name (  
    column_name TYPE column_constraint,  
    table_constraint table_constraint  
);
```

- No exemplo acima nós temos:
 - **schema_name:** Nome do schema que a tabela será criada
 - **table_name:** Nome da tabela a ser criada;
 - **column_name:** Nome da coluna a ser criada;
 - **TYPE:** O tipo de dados da coluna;
 - **column_constraint** (restrições aplicadas à coluna): define os papéis da coluna (PRIMARY KEY, NOT NULL, etc.)

CREATE

- Podemos utilizar o create para databases, schemas e tabelas, como nos exemplos abaixo:

```
CREATE SCHEMA schema_name;
```

```
CREATE DATABASE database_name;
```


TIPOS DE DADOS

TEXTO

CHAR (N)
VARCHAR (N)
TEXT

NUMÉRICOS

INT / INTEGER

(DECIMAL,
NUMERIC) (P,E)

FLOAT, REAL

DATAS

DATE

DATETIME
TIMESTAMP

OUTROS

BOOLEAN

ENUM

Observações:

- **N**: Número limite de caracteres utilizados na declaração do tipo de dado;
- **P,E**: Devido ao fato do **DECIMAL/NUMERIC** serem números precisos, utilizamos o **P**, ou precisão, para orientarmos a quantidade de caracteres e o **E**, ou escala, para orientarmos os número após a vírgula;

RESTRIÇÕES

Podemos colocar algumas restrições em nossas Colunas:

- **NOT NULL** - o valor da coluna não pode ser nulo (em branco).
- **UNIQUE** - o valor da coluna deve ser exclusivo em toda a tabela.
- **PRIMARY KEY** - essa restrição é a combinação de restrições NOT NULL e UNIQUE.
 - Você pode definir uma coluna como PRIMARY KEY usando a restrição no nível da coluna. Caso a chave primária contenha várias colunas, você deve usar a restrição em nível de tabela.
- **CHECK** - permite verificar uma condição ao inserir ou atualizar dados. Por exemplo, os valores na coluna preço da tabela de produtos devem ser valores positivos.
- **REFERENCES** - restringe o valor da coluna que existe em uma coluna em outra tabela. Você usa REFERENCES para definir a restrição de chave estrangeira.

CAMPOS AUTO GERADOS

- Para criarmos tabelas com campos auto gerados utilizamos o comando **GENERATED ALWAYS AS [...] STORED**:

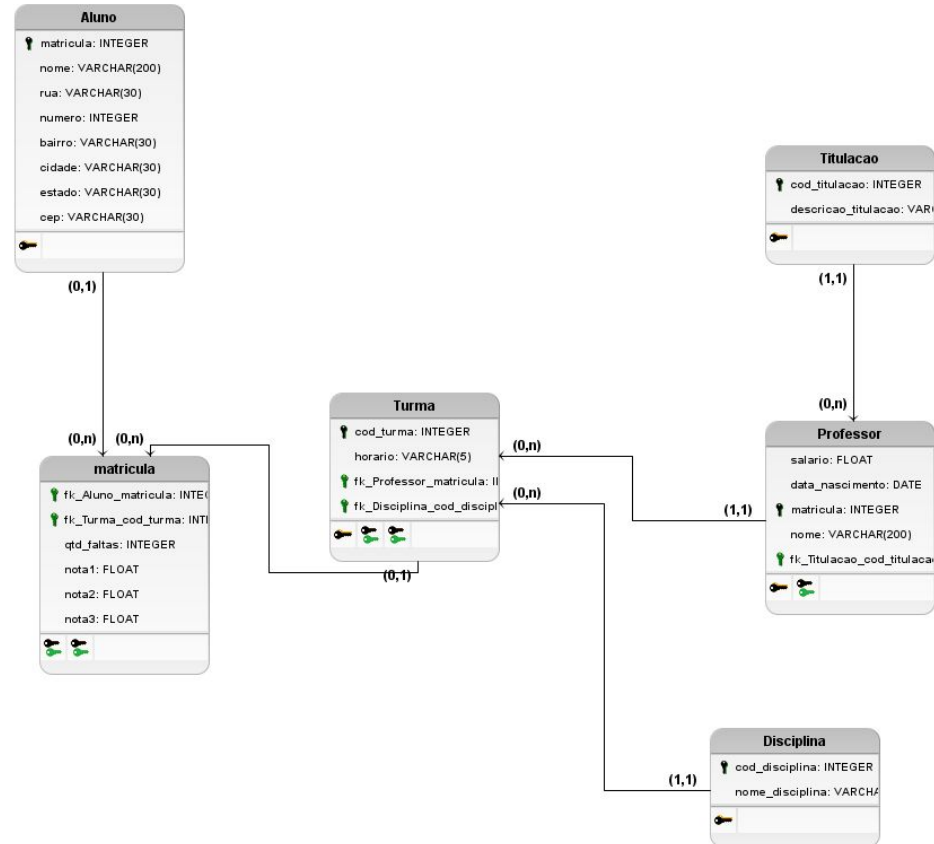
```
create table pessoa (  
    id serial primary key,  
    primeiro_nome varchar(40) not null,  
    ultimo_nome varchar(40) not null,  
    nome_completo varchar(80) generated always as (primeiro_nome || ' ' || ultimo_nome)  
stored  
);
```

- O exemplo acima é basicamente uma forma de realizarmos a criação de um campo auto gerado para a construção do nome completo;

O tipo de valor de uma coluna auto gerada pode ser:

- **STORED:** o valor é gerado no momento da criação do registro e é salvo na tabela;
- **VIRTUAL:** o valor não é salvo na tabela e é computado no momento em que o registro é consultado. **O PostgreSQL não implementa este tipo de coluna autogerada.**

EXERCÍCIO 1



EXERCÍCIO 1

- A partir do modelo lógico apresentado anteriormente, realize:
 - Criação de schema;
 - Criação das tabelas contendo os tipos de dados informados no modelo.
 - Obs:.. Não existe a necessidade de adicionar FK nesse primeiro momento.

Alterando uma tabela no nosso banco:

Adicionar Coluna na tabela:

```
ALTER TABLE table_name ADD column_name datatype;
```

Remover uma Coluna da Tabela:

```
ALTER TABLE table_name DROP COLUMN column_name;
```

Alterando ou Modificando uma Coluna:

```
ALTER TABLE table_name ALTER COLUMN column_name datatype;
```

Renomear uma Coluna:

```
ALTER TABLE table_name RENAME COLUMN column_name TO new_column;
```

EXERCÍCIO 2

- A partir das tabelas criadas anteriormente realize:
 - Alteração do tipo de dados de pelo menos oito campos.
 - Alteração do nome de pelo menos três colunas.
 - Identificar um caso de uso e adicionar uma coluna contendo um campo autogerado.
 - Remover uma coluna.

Alterando uma tabela no nosso banco:

Remover uma tabela:

- `DROP TABLE schema_name.table_name RESTRICT;`
- `DROP TABLE schema_name.table_name;`
- ***RESTRICT*** é o valor padrão;

Remover uma tabela retirando os objetos dependentes da mesma:

- `DROP TABLE schema_name.table_name CASCADE;`
- ***Seja cauteloso na hora de utilizar este comando;***

Remover um schema :

- `DROP SCHEMA schema_name;`

Remover um banco de dados :

- `DROP DATABASE database_name;`

EXERCÍCIO 3

- A partir das tabelas criadas anteriormente realize:
 - Remova as tabelas que você queira.
 - Remova os schemas que foram criados.



DEVinHouse

Parcerias para desenvolver a sua carreira

OBRIGADO!



<LAB365>