Cheatsheet do Comando `CHECK`

A restrição **`CHECK`** é uma poderosa ferramenta de validação de dados em PostgreSQL (e em outros SGBDs). Ela permite que você defina uma expressão booleana (que retorna `verdadeiro` ou `falso`) que deve ser satisfeita para que uma linha seja inserida ou atualizada na tabela. Se a condição for falsa, a operação será rejeitada.

Sintaxe Básica

A restrição `CHECK` pode ser adicionada ao criar uma tabela ou em uma tabela já existente.

1. Adicionando ao criar a tabela:

```
SQL
CREATE TABLE nome_da_tabela (
    coluna1 tipo,
    coluna2 tipo,
    CONSTRAINT nome_da_restricao CHECK (condicao)
);
```

2. Adicionando a uma tabela existente (`ALTER TABLE`):

```
SQL
ALTER TABLE nome_da_tabela
ADD CONSTRAINT nome_da_restricao CHECK (condicao);
```

O que você pode validar com `CHECK`?

Você pode usar qualquer expressão que resulte em um valor booleano (`true` ou `false`). Isso inclui:

- **Valores numéricos**: Checar se um número está em um determinado intervalo.
- **Valores de texto**: Checar o formato do texto, se ele contém apenas dígitos, se é um email válido, etc.
- **Valores de data**: Garantir que uma data seja maior ou menor que outra.
- **Condições entre colunas**: Comparar valores de duas ou mais colunas na mesma linha.

Exemplos Práticos

Exemplo 1: Validação de Valores Numéricos

- **Missão**: Garantir que a idade de um cliente seja sempre maior que 18.

```
SQL
CREATE TABLE clientes (
   id SERIAL PRIMARY KEY,
   nome VARCHAR(255),
   idade INT,
   CONSTRAINT chk_idade CHECK (idade >= 18)
);

-- Isso funcionará
INSERT INTO clientes (nome, idade) VALUES ('Maria', 25);

-- Isso retornará um erro
INSERT INTO clientes (nome, idade) VALUES ('Pedro', 16);
-- ERRO: new row for relation "clientes" violates check constraint "chk_idade"
```

Exemplo 2: Validação de Valores de Texto (com `LIKE` e `SIMILAR TO`)

- **Missão**: Garantir que o nome de um produto comece com "A" ou "B".

```
SQL
CREATE TABLE produtos (
   id SERIAL PRIMARY KEY,
   nome VARCHAR(255),
   CONSTRAINT chk_nome_produto CHECK (nome LIKE 'A%' OR nome LIKE 'B%')
);

-- Isso funcionará
INSERT INTO produtos (nome) VALUES ('Abacate');

-- Isso retornará um erro
INSERT INTO produtos (nome) VALUES ('Cenoura');
```

Exemplo 3: Checagem de CPF ser apenas dígitos (usando expressões regulares)

A validação de um CPF pode ser feita de forma elegante usando expressões regulares, que são suportadas pelo PostgreSQL. A função `SIMILAR TO` ou o operador `~` (para regex) são ideais para essa tarefa.

- **Missão**: Garantir que a coluna `cpf` contenha apenas 11 dígitos.

SQL
```
CREATE TABLE funcionarios (
   id SERIAL PRIMARY KEY,
   nome VARCHAR(255),
   cpf VARCHAR(11) NOT NULL,
   CONSTRAINT chk_cpf_apenas_digitos CHECK (cpf SIMILAR TO '[0-9]{11}')
);

-- Isso funcionará (11 dígitos)
INSERT INTO funcionarios (nome, cpf) VALUES ('Carlos', '12345678901');

-- Isso retornará um erro (contém letra)
INSERT INTO funcionarios (nome, cpf) VALUES ('Ana', '1234567890a');

-- Isso retornará um erro (contém menos que 11 dígitos)
INSERT INTO funcionarios (nome, cpf) VALUES ('João', '123');
```

- **`SIMILAR TO`**: É o operador padrão para correspondência de padrões no SQL.
- **`[0-9]{11}`**: Essa é a expressão regular que faz a validação:
    - `[0-9]`: Indica que o caractere deve ser um dígito de 0 a 9.
    - `{11}`: Indica que o padrão anterior (`[0-9]`) deve ser repetido exatamente 11 vezes.

Exemplo 4: Comparando colunas

- **Missão**: Garantir que a `data_devolucao` de um empréstimo seja sempre depois da `data_emprestimo`.

SQL
```
ALTER TABLE emprestimos
ADD CONSTRAINT chk_datas CHECK (data_devolucao IS NULL OR data_devolucao >
data_emprestimo);

-- Isso funcionará (data de devolução é posterior)
UPDATE emprestimos SET data_devolucao = '2025-08-20' WHERE id = 1;

-- Isso retornará um erro
UPDATE emprestimos SET data_devolucao = '2025-08-10' WHERE id = 1;
-- ERRO: new row for relation "emprestimos" violates check constraint "chk_datas"
```

- `data_devolucao IS NULL`: Essa parte é crucial para permitir que os empréstimos possam ser criados sem uma data de devolução, já que o livro ainda não foi devolvido. A condição só será ativada se uma data for fornecida.