



# BANCO DE DADOS

DDL: ALTER



INSTITUTO FEDERAL DE  
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA  
RIO GRANDE DO NORTE

Msc. Eliezio Soares  
eliezio.soares@ifrn.edu.br | <https://docente.ifrn.edu.br/elieziosoares>

# MODIFICAÇÃO DE TABELAS

- Caso precise realizar uma alteração em uma tabela, seja por um erro na criação da mesma, seja por alterações nos requisitos, você pode:
  - Apagar a tabela e cri-la novamente.
    - Mas a tabela pode já conter dados.
    - A tabela pode ser utilizada por outros objetos.
  - Há comandos que permitem realizar alterações em tabelas existentes.
    - Isso é diferente de alterar os dados.
    - O foco aqui é alterar a definição dos dados, a estrutura da tabela.
- **ALTER TABLE**

# MODIFICAÇÃO DE TABELAS

- ALTER TABLE
  - Adicionar Colunas
  - Remover Colunas
  - Adicionar Restrições (constraints)
  - Remover Restrições (constraints)
  - Modificar valor padrão
  - Modificar tipo de dado de um campo
  - Renomear colunas
  - Renomear tabela

# ADICIONAR COLUNA

```
ALTER TABLE Produto ADD COLUMN descricao text;
```

- A nova coluna será criada;
- A nova coluna será preenchida com o valor padrão
- É possível definir constraints:

```
ALTER TABLE Produto ADD COLUMN descricao text CHECK (descricao <> "");
```

# REMOVER COLUNA

```
ALTER TABLE Produto DROP COLUMN descricao;
```

- Todos os dados dessa coluna desaparecem.
- As constraints relacionadas a coluna também são apagadas.
- Caso a coluna seja referenciada por uma chave estrangeira, não será possível apagar a coluna sem problemas.
  - É possível autorizar o PostgreSQL a apagar tudo o que dependa da coluna:

```
ALTER TABLE Produto DROP COLUMN descricao CASCADE;
```

# ADICIONAR RESTRIÇÃO

```
ALTER TABLE Produto ADD CHECK (nome <> "");
```

```
ALTER TABLE Produto ADD CONSTRAINT nome_restricao UNIQUE (produto_num);
```

```
ALTER TABLE Produto ADD FOREIGN KEY (produto_grupo_id) REFERENCES produto_grupo;
```

# ADICIONAR RESTRIÇÃO

- Para adicionar uma restrição **NOT NULL**, use a sintaxe a baixo:

```
ALTER TABLE Produto ALTER COLUMN produto_num SET NOT NULL;
```

# REMOVER RESTRIÇÃO

- É preciso saber o nome da constraint:

```
ALTER TABLE Produto DROP CONSTRAINT nome_restricao;
```

- As restrições NOT NULL funcionam diferente:

```
ALTER TABLE Produto ALTER COLUMN produto_num DROP NOT NULL;
```

# MODIFICAR O VALOR PADRÃO DE UMA COLUNA

- **Adicionar um novo padrão para uma coluna:**

```
ALTER TABLE Produto ALTER COLUMN preco SET DEFAULT 7.77;
```

- Esse comando não altera valores existentes nas linhas da tabela.
- O comando afeta as inserções futuras.

- **Para remover algum DEFAULT:**

```
ALTER TABLE Produto ALTER COLUMN preco DROP DEFAULT;
```

- Isso equivale a "...SET DEFAULT NULL;"

# MODIFICAR O TIPO DE DADO DE UMA COLUNA

- Para converter uma coluna para um tipo de dado diferente:

```
ALTER TABLE Produto ALTER COLUMN preco TYPE numeric;
```

- O PostgreSQL tentará converter cada valor existente na tabela para o novo tipo de dado.

# RENOMEAR COLUNA

- Para renomear uma coluna:

```
ALTER TABLE Produto RENAME COLUMN produto_num TO produto_numero;
```

# RENOMEAR TABELA

- Para renomear uma tabela:

```
ALTER TABLE Produto RENAME TO itens;
```

# DÚVIDAS?



# REFERÊNCIAS BIBLIOGRÁFICAS

PostgreSQL 9.0.22 Documentation. Disponível em:  
<<https://www.postgresql.org/files/documentation/pdf/9.0/postgresql-9.0-US.pdf>>. Acesso em 27 Set. 2016.