

# ALTER



# 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 cria-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



## o ALTER TABLE

- o Adicionar Colunas
- o Remover Colunas
- o Adicionar Restrições (constraints)
- o Remover Restrições (constraints)
- o Modificar valor padrão
- o Modificar tipo de dado de um campo
- o Renomear colunas
- o 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 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 :

```
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;
```

# 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;
```

# Criação da Tabela **T\_CLASSE\_SOCIAL**.



```
1 CREATE TABLE T_CLASSE_SOCIAL(  
2     COD_CLASSE_SOCIAL CHAR(1) PRIMARY KEY,  
3     DSC_FAIXA_RENDA VARCHAR2(20)  
4 )
```

# ADD



- ALTER TABLE com o objetivo de adicionar a coluna TESTE, do tipo NUMBER(3), à tabela T\_CLASSE\_SOCIAL.

```
ALTER TABLE T_CLASSE_SOCIAL  
ADD (TESTE NUMBER(3))
```

# Modify



- A cláusula MODIFY é utilizada em conjunto com a instrução ALTER TABLE para modificar uma coluna.
- É possível mudar tanto o tamanho da coluna, como o tipo de dados. Na prática, a modificação de uma coluna costuma ser necessária nos seguintes casos:
- aumentar a largura de um campo NUMBER , CHAR ou VARCHAR ou
- aumentar a precisão de um campo NUMBER que possui casas decimais.

```
ALTER TABLE T_CLASSE SOCIAL MODIFY  
(TESTE VARCHAR2(50))
```

# RENAME COLUMN



- Para renomear uma coluna, aplica-se a cláusula RENAME COLUMN.

```
ALTER TABLE T_CLASSE_SOCIAL RENAME  
COLUMN TESTE TO TESTE_NOVO;
```





- Renomeando uma Tabela

```
ALTER TABLE T_CLASSE_SOCIAL  
RENAME TO T_CLASSE;
```

# DROP COLUMN



- Uma coluna pode ser eliminada de uma tabela com o uso da cláusula DROP COLUMN. Quando a coluna é excluída, todos os dados que nela estavam armazenados também são apagados. Estes dados não poderão ser recuperados (a não que se recupere um backup).

```
ALTER TABLE T_CLASSE DROP COLUMN TESTE_NOVO
```