

03.5

SQL SELECT

Select

SELECT simples

O comando **SELECT** permite recuperar os dados de um objeto do banco de dados, como uma tabela, view e, em alguns casos, uma stored procedure (alguns bancos de dados permitem a criação de procedimentos que retornam valor). A sintaxe mais básica do comando é:

SELECT <lista_de_campos> FROM <nome_da_tabela></nome_da_tabela></lista_de_campos>

Exemplo:

SELECT CODIGO, NOME FROM CLIENTES SELECT * FROM CLIENTES

O caractere * representa todos os campos. Apesar de prático, este caractere não é muito utilizado, pois, para o SGBD é mais rápido receber o comando com todos os campos explicitados. O uso do * obriga o servidor a consultar quais são os campos antes de efetuar a busca dos dados, criando mais um passo no processo.

Where

A cláusula Where permite ao comando SQL passar condições de filtragem.

```
SELECT CODIGO, NOME FROM CLIENTES WHERE CODIGO = 10
```

```
SELECT CODIGO, NOME FROM CLIENTES WHERE UF = 'RJ'
```

```
SELECT CODIGO, NOME FROM CLIENTES WHERE CODIGO >= 100 AND CODIGO <= 500
```

```
SELECT CODIGO, NOME FROM CLIENTES WHERE UF = 'MG' OR UF = 'SP'
```

Com ela podemos escolher apenas algumas informações dentro de uma lista de possibilidades.

Lembra o comando IF de algoritmo.

LIKE

Para busca parcial de string, o SELECT fornece o operador LIKE. Veja o exemplo abaixo:

```
SELECT CODIGO, NOME FROM CLIENTES
```

```
WHERE NOME LIKE 'MARIA%'
```

Neste comando, todos os clientes cujos nomes iniciam com Maria serão retornados. Se quisermos retornar os nomes que contenham 'MARIA' também no meio, podemos alterar para o exemplo a seguir:

```
SELECT CODIGO, NOME FROM CLIENTES
```

```
WHERE NOME LIKE '%MARIA%'
```

O uso de máscara no início e no fim da string fornece maior poder de busca, mas causa considerável perda de performance. Este recurso deve ser utilizado com critério.

SQL SELECT DISTINCT Statement

SQL SELECT Statement

SQL SELECT: Um Guia para Iniciantes em
SQL