

CONSULTA DE REGISTROS

- Para extrair listagens de registros de uma tabela, a linguagem SQL disponibiliza o comando **SELECT** que anexado a alguns parâmetros, fornece uma instrução de consulta:

```
SELECT [tipo] <campos> FROM <tabela> [condição];
```

CONSULTA DE REGISTROS

- Apresentar a listagem de todos os funcionários cadastrados:

```
SELECT * FROM funcionario;
```

CONSULTA DE REGISTROS

- Observe o uso do valor * (asterisco), indicando que devem ser mostrados todos os campos da tabela **funcionario**.
- O comando **SELECT** apresenta os dados e o comando **FROM** indica de que tabela a apresentação será feita.

listar os nomes e a função
de todos os funcionários

CONSULTA DE REGISTROS

- O exemplo seguinte apresenta uma listagem dos nomes e das funções de todos os registros armazenados na tabela **funcionario**.

```
SELECT nome, funcao FROM funcionario;
```

listar os nomes
de todos os funcionários
que trabalham no departamento 3

CONSULTA DE REGISTROS | WHERE

- O exemplo seguinte exibe uma listagem dos registros da tabela funcionario que estejam relacionados com o departamento 3.

```
SELECT nome FROM funcionario WHERE departamento = '3';
```

listar os nomes e o salário
de todos os funcionários
ordenados pelo nome

CONSULTA DE REGISTROS | ORDER BY

- O exemplo seguinte mostra a listagem dos nomes e dos salários dos registros da tabela funcionario por ordem alfabética do campo nome.

```
SELECT nome, salario FROM funcionario ORDER BY nome;
```

Formato descendente:

```
SELECT nome, salario FROM funcionario ORDER BY nome DESC;
```

listar o departamento e o nome
de todos os funcionários
ordenados pelo departamento

CONSULTA DE REGISTROS | ORDER BY

- O exemplo seguinte mostra uma listagem dos departamentos e nomes dos registros da tabela funcionario por ordem do campo departamento em formato ascendente e do campo nome em formato descendente.

```
SELECT departamento, nome
```

```
FROM funcionario
```

```
ORDER BY departamento, nome DESC;
```

listar o nome
de todos os funcionários
que trabalham no departamento 3
ordenados pelo nome

CONSULTA DE REGISTROS | WHERE & ORDER BY

- O exemplo mostra os nomes da tabela funcionario ordenados pelo campo nome em formato ascendente, dos funcionários do departamento 3.

```
SELECT nome
```

```
FROM funcionario
```

```
WHERE departamento = '3'
```

```
ORDER BY nome;
```

ALTERAÇÃO DE REGISTROS

- No processo de manutenção de registros em bancos de dados é necessário periodicamente atualizar dados e informações existentes.
- O processo de alteração de registros é considerado no programa MySQL pelo comando da linguagem SQL denominado **UPDATE** e o comando **SET** que possui a seguinte sintaxe:

UPDATE <tabela> **SET** <campo> = <expressão> [condição];

ALTERAÇÃO DE REGISTROS

- O funcionário de código de registro 7 teve um aumento de salário de 1200.50 para 2300.56 e essa condição precisa ser considerada na tabela **funcionario**.
- Mas antes de proceder à alteração do salário do funcionário, execute a listagem de seus dados atuais com o comando:

```
SELECT * FROM funcionario WHERE id = 7;
```

ALTERAÇÃO DE REGISTROS

- Em seguida execute a instrução de atualização de dados de um determinado registro para atualizar o campo salário como indicado:

```
UPDATE funcionário
```

```
SET salario = 2300.56
```

```
WHERE id = 7;
```


ALTERAÇÃO DE REGISTROS

- A funcionária Ana Bastos do departamento 5, fez uma solicitação de transferência de departamento, a qual foi aceita pela chefia. Ela pediu para ser transferida para o departamento 3:

```
SELECT * FROM funcionario WHERE nome = 'ana bastos';
```

```
UPDATE funcionario SET departamento = '3' WHERE nome = 'ana bastos';
```

```
SELECT * FROM funcionario WHERE nome = 'ana bastos';
```

ALTERAÇÃO DE REGISTROS

- O dono da empresa Virtual Ltda. Concedeu um aumento de 10% a todos os seus funcionários após uma reinvidicação.

```
SELECT * FROM funcionario;
```

ALTERAÇÃO DE REGISTROS

- Para efetivar o aumento de 10% concedido pela empresa, é necessário usar a instrução de atualização com a seguinte sintaxe:

```
UPDATE funcionario SET salario = salario * 1.10;
```

```
SELECT * FROM funcionario;
```

REMOÇÃO DE REGISTROS

- Dentre as várias atividades da manutenção de registros, existe a possibilidade de remover aqueles que não sejam mais necessários.
- É preciso usar a instrução **DELETE FROM** que possui a seguinte sintaxe:

DELETE FROM <tabela> [condição];

REMOÇÃO DE REGISTROS

- De todos os comandos existentes na linguagem estruturada de consulta SQL, este é o mais perigoso, pois se acionado de forma inadequada, pode ocasionar prejuízos:

REMOÇÃO DE REGISTROS

DELETE FROM funcionario;

Ocorre a remoção de todos os registros da tabela funcionario. Por isso é conveniente sempre usar o parâmetro **WHERE** nas operações de remoção de registros, a menos que se tenha a certeza de que é preciso realmente apagar todos os registros.

REMOÇÃO DE REGISTROS

- A empresa Virtual Ltda. está passando por uma reengenharia; o departamento 2 foi extinto e seus funcionários foram demitidos.
- Desta forma, torna-se necessário remover os registros que atendam a estes critérios de condição:

```
DELETE FROM funcionario WHERE departamento = '2';
```

REMOÇÃO DE REGISTROS

- O funcionário Marcelo Souza, insatisfeito com as mudanças da empresa, conseguiu um emprego no concorrente, então solicitou sua demissão, portanto esse registro precisa ser eliminado:

```
DELETE FROM funcionario WHERE nome = 'marcelo souza';
```