

BANCO DE DADOS

Linguagem de Manipulação de Dados (DML)

A inserção dos registros nas entidades é realizada com o comando **INSERT**, que possui a seguinte sintaxe básica:

Sintaxe: INSERT INTO [tabela] (lista de colunas) VALUES (lista de valores)

A lista de colunas logo após o nome da tabela é opcional, se forem informadas as colunas chamamos de insert referencial e se não forem informadas as colunas de insert posicional.

A vantagem do modelo referencial sobre o modelo posicional, é que neste último todas as colunas da tabela precisam constar no values para encaixar com as colunas, além de correremos o risco de alterarem a lista de colunas e os dados ficarem invertidos.

Os seguintes códigos SQL adicionam um novo hospede:

```
INSERT INTO hospede (nome, genero, biotipo, altura)
VALUES ('MIGUEL', 1, 1, 1.67)
```

Adicione, conforme as tabelas 1 e 2 vistas no tópico anterior os registros de HOSPEDE e QUARTO. A modificação dos registros nas entidades é realizada com o comando **UPDATE**, que possui a seguinte sintaxe básica:

```
UPDATE [tabela] SET [campo1 = valor1] [, campoN = valorN] WHERE [condição]
```

Apesar da cláusula **WHERE** constar no **UPDATE**, é opcional. Entretanto, sem essa opção o comando modificará todos os registros.

O **exemplo** abaixo é o recomendado, pois inclui a definição para um filtro de modificação:

```
UPDATE hospede SET biotipo = 2 WHERE nome = 'MIGUEL'
```

A exclusão dos registros nas entidades é realizada com o comando **DELETE**, que possui a seguinte sintaxe básica:

```
DELETE FROM [tabela] WHERE [condição]
```

Apesar da cláusula **WHERE** constar no comando **DELETE**, é opcional. Entretanto, sem essa opção o comando eliminará todos os registros.

O exemplo abaixo é o recomendado, pois inclui a definição para um filtro de exclusão:

```
DELETE FROM hospede WHERE nome = 'MIGUEL'
```

A consulta dos registros nas entidades é realizada com o comando **SELECT**, que possui a seguinte sintaxe básica:

Sintaxe: SELECT <lista_de_colunas> FROM <nome_tabela>

Por exemplo, podemos utilizar o seguinte comando para mostrar todos os hóspedes cadastrados:

```
SELECT * FROM hospede
```

O exemplo utiliza o coringa * para selecionar as colunas na ordem em que foram criadas. A instrução **SELECT**, como pudemos observar seleciona um grupo de registros de uma (ou mais) tabela(s).

No caso a instrução **FROM** nos indica a necessidade de pesquisarmos tais dados apenas na tabela HOSPEDE.

A cláusula **DISTINCT** elimina duplicidades, significando que somente relações distintas serão apresentadas como resultado de uma pesquisa.

```
SELECT DISTINCT nome FROM hospede
```

A cláusula **ORDER BY** modificará a ordem de apresentação do resultado da pesquisa ascendente usando o termo "**ASC**" (por padrão) ou descendente usando o termo "**DESC**".

```
SELECT nome, genero FROM hospede ORDER BY nome
```

Operadores Aritméticos

São quatro os operadores aritméticos que podem formar expressões:

- + adição
- subtração
- * multiplicação
- / divisão

Na consulta a seguir, usamos o operador de adição para adicionar 0.05 ao valor corrente de DESCONTO a todos os hóspedes que já possuem descontos.

A cláusula **SELECT** contém duas expressões: NOME e a expressão formada pelo operador aritmético, DESCONTO + 0.05 mostrando um novo nome para esta coluna (novo_desconto).

A tabela resultado deverá conter duas colunas, uma para cada expressão:

```
SELECT nome, desconto + 0.05 novo desconto FROM quarto WHERE desconto IS NOT NULL
```

A consulta abaixo pode ser usada para determinar por quantos dias os hóspedes ficaram no **Spa**. Este número é obtido subtraindo-se a data de chegada de saída.

```
SELECT nome, saida - chegada tempo FROM quarto
```