



INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
PARAÍBA



Linguagem SQL

DML - Linguagem de Manipulação de Dados

Sintaxe:

```
INSERT [INTO] <tabela> [(lista de colunas)]  
VALUES <(lista de valores)>
```

Exemplo 1: Cadastrar a cidade de Muriaé, com código 91, do estado de Minas Gerais (sigla "MG"), no Brasil (código "BRA").

```
INSERT INTO Cidade (idcidade, nome, uf, idpais)  
VALUES (91, 'Muriaé', 'MG', 'BRA');
```



- Se uma coluna aceitar valores nulos ou possuir um valor default, é possível omiti-la em uma instrução INSERT.
- Sempre deverá ser inserido um valor para uma coluna que seja NOT NULL.

Exemplo 2: Cadastrar o cliente de código "ANBAL", nome "Antonio Barbosa Leite", tipo "F", endereço "Av. Rui Carneiro, 723", cidade João Pessoa (código 7).

```
INSERT Cliente(idcliente,nome,tip,endereco,idcidade)
VALUES ('ANBAL','Antonio Barbosa Leite','F','Av. Rui
Carneiro,723',7);
```



- Na inserção de dados parciais:
 - é inserido um valor nulo para as colunas que aceitam valores nulos e não possuem valores default;
 - é inserido o valor default para as colunas que contém uma constraint DEFAULT;
 - um valor nulo poderá ser especificado explicitamente, digitando NULL sem aspas simples;
 - Deverá ser especificado um valor nulo para uma coluna que possua a propriedade AUTO INCREMENT.



Criação da tabela Teste1

CREATE TABLE Teste1

(

coluna1 INT NOT NULL AUTO_INCREMENT PRIMARY KEY,

coluna2 VARCHAR(40) NULL DEFAULT 'XXXXXX',

coluna3 INT NULL,

coluna4 VARCHAR(30) NOT NULL

);



```
INSERT INTO Teste1(coluna4)
VALUES ('valor 1-4');

INSERT INTO Teste1(coluna2,coluna4)
VALUES (NULL,'valor 2-4');

INSERT INTO Teste1(coluna2,coluna3,coluna4)
VALUES (NULL,NULL,'valor 3-4');

INSERT INTO Teste1(coluna2,coluna3,coluna4)
VALUES (DEFAULT,DEFAULT,'valor 4-4');

INSERT INTO Teste1(coluna2,coluna3,coluna4)
VALUES (NULL,DEFAULT,'valor 5-4');
```



Sintaxe:

```
INSERT [INTO] <tabela> [<lista de colunas>]  
VALUES  
SELECT ... FROM ...
```

Exemplo 3: Cadastrar a cidade de Olinda, do estado de Pernambuco ("PE"), no Brasil ("BRA").

```
INSERT INTO Cidade  
SELECT MAX(idcidade)+1, 'Olinda', 'PE', 'BRA'  
FROM Cidade;
```



Criação da tabela TESTE2

CREATE TABLE Teste2

(

func_id INT NOT NULL PRIMARY KEY AUTO_INCREMENT,

nome VARCHAR(40) NULL,

sexo CHAR(1) NULL,

endereço VARCHAR(40) NULL

);



Inclusão de dados na tabela TESTE2

```
INSERT INTO Teste2  
SELECT NULL, nome, sexo, endereco  
FROM funcionario  
WHERE estcivil = 'C';  
  
SELECT * FROM Teste2;
```

- As colunas que recebem os dados devem ser do mesmo tipo e possuir tamanho maior ou igual ao das colunas da tabela de origem.
- A primeira coluna, por ser AUTO_INCREMENT, não deve ter valor informado no INSERT, por isso foi colocado NULL nela.



Criação da tabela TESTE3

CREATE TABLE Teste3

(

pedido INT NOT NULL PRIMARY KEY,

data DATETIME NOT NULL,

cliente VARCHAR(40) NOT NULL,

vendedor VARCHAR(40) NOT NULL

);



Inclusão de dados na tabela TESTE3

```
INSERT INTO Teste3
SELECT P.idpedido, P.datapedid, C.nome, F.nome
FROM pedido P
JOIN cliente C
ON P.idcliente = C.idcliente
JOIN Funcionario F
ON P.idvendedor = F.idfuncionario;

SELECT * FROM Teste3;
```



Sintaxe:

```
UPDATE tabela  
SET coluna = expressão | DEFAULT | NULL  
WHERE condição
```

Exemplo 4: Dar um aumento de 12% no preço de venda dos produtos dos tipos 2 e 3.

```
UPDATE produto  
SET venda = venda * 1.12  
WHERE idtipo IN (2,3);
```



Exemplo 5: Dar um aumento de 25% no salário dos funcionários casados e manter o salário antigo para os demais funcionários. Armazenar os novos valores em uma nova coluna da tabela.

```
ALTER TABLE funcionario  
ADD novosalario DECIMAL(10,2) NULL;
```

```
UPDATE funcionario  
SET novosalario = salario * 1.25  
WHERE estcivil = 'C';
```

```
UPDATE funcionario  
SET novosalario = salario  
WHERE estcivil <> 'C';
```



Sintaxe:

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

Exemplo 6: Excluir todos os clientes do tipo pessoa física que não possuam telefone

```
DELETE FROM cliente  
WHERE tipo = 'F' and fone IS NULL;
```



Exemplo 7: Excluir todos os produtos com quantidade em estoque maior que 100 ou preço de custo menor ou igual a R\$ 10,00.

```
DELETE FROM produto  
WHERE quantest > 100 OR custo <= 10.00;
```

Exemplo 8: Excluir todos os registros da tabela de tipos.

```
DELETE FROM tipo;
```