

# BANCOS DE DADOS

Cap.4 - SQL (Pt.2)



Prof. MSc. Renzo P. Mesquita  
renzo@inatel.br

# Objetivos

- Discutirmos detalhes importantes sobre a Inserção, Atualização e Exclusão de registros em BDs Relacionais;
- Destacarmos comandos importantes da subfamília DML (Data Manipulation Language) da linguagem SQL;



# Capítulo 4

## SQL (Pt.2)

*4.6. A Subfamília DML;*

*4.7. Inserção de Dados com INSERT;*

*4.8. Atualização de Dados com UPDATE;*

*4.9. Exclusão de Dados com DELETE;*





## 4.6. A Subfamília DML

*Os comandos SQL que lidam com a manipulação de dados presentes dentro das tabelas dos Bancos de Dados pertencem à subfamília DML (Data Manipulation Language).*

*Seus principais comandos são:*

- **INSERT:** usado para inserir dados em uma tabela;
  - **UPDATE:** usado para atualizar registros em uma tabela;
  - **DELETE:** usado para excluir registros das tabelas;
- Em algumas obras, o comando SELECT também é tratado como um comando DML. Porém, nesta disciplina, estudaremos ele de forma separada como parte da subfamília DQL (Data Query Language) na próxima parte deste capítulo;
  - Nesta parte do capítulo, veremos aplicações fundamentais dos comandos DML, porém, quando avançarmos para comandos DQL, vamos aprender recursos avançados que inclusive podem ser aplicados para criarem comandos DML ainda mais poderosos;



## 4.7. Inserção de Dados com INSERT

*O comando INSERT INTO é utilizado para se inserir novas linhas (registros) em uma tabela.*

Existem duas formas de se fazer uso deste comando:

### FORMA 1 - APENAS VALORES

Basta que TODAS as colunas sejam preenchidas na ordem que aparecem na Tabela.

*Ex:* **INSERT INTO** Empregado  
**VALUES** ("Jason","L","Scott",653298653,"1990-12-30",  
"Alameda dos Anjos, 55, Porto Caribe, Nova York",  
1,4000,NULL,3);

- Para Tabelas que possuem chaves auto incrementáveis, basta adicionar NULL na posição da mesma para passar a responsabilidade do preenchimento do valor para o Banco.





## 4.7. Inserção de Dados com INSERT

### FORMA 2 - ESPECIFICANDO ATRIBUTOS

*Ex:* **INSERT INTO** Empregado (SSN, PNome, UNome, DNumero\_Departamento)  
**VALUES** (993295651, "Kimberly", "Hart", 3);

- Os atributos que não forem especificados serão registrados com seus valores DEFAULT ou NULL. Os atributos preenchidos deverão respeitar seus domínios (tipo e perfil) de dados;
- Atributos (ou Colunas) NOT NULL deverão ser obrigatoriamente preenchidos, independente se estamos usando a FORMA 1 ou 2;
- Chaves Primárias auto incrementáveis não precisam aparecer na relação de atributos, pois já serão preenchidas automaticamente;
- Lembre-se que se uma Inserção não obedecer às restrições do Modelo Relacional (Chave, Entidade e Integridade Referencial) a mesma não vai dar certo;
- Restrições Semânticas (Ex: usando CHECK ou TRIGGERS) também poderão barrar a correta Inserção de dados, porém, precisam ser previamente programadas no Banco;
- Valores UNIQUE, caso não obedecidos, também podem barrar uma Inserção.

## 4.8. Atualização de Dados com UPDATE

*O comando UPDATE é utilizado para se atualizar dados que já se encontram cadastrados em uma determinada tabela do Banco.*

São 3 as principais formas de se fazer uso deste comando:

### FORMA 1 - ATUALIZANDO UMA ÚNICA COLUNA

*Ex:* `UPDATE` Projeto  
      `SET` PLocalização = "Central City"  
      `WHERE` PNumero = 3;

- Observe que a cláusula SET, dentro do comando UPDATE, especifica o(s) atributo(s) que será (ão) modificado(s) e seu(s) novo(s) valor(es);
- A cláusula WHERE seleciona a(s) linha(s) de uma única Tabela que será (ão) modificada(s);
- Vale ressaltar que, uma atualização no valor da chave primária pode propagar-se para os valores das chaves estrangeiras (desde que estas chaves estrangeiras estejam devidamente configuradas como ON UPDATE CASCADE);



## 4.8. Atualização de Dados com UPDATE

### FORMA 2 - ATUALIZANDO MÚLTIPLAS COLUNAS

*Ex:* **UPDATE** Empregado  
      **SET** Salario = 1.1\*Salario, SSN\_Supervisor = 3  
      **WHERE** DNumero\_Departamento = 2;

- No exemplo, observe dois campos sendo atualizados na cláusula SET;
- Observe também que múltiplas linhas podem ser impactadas com essa atualização (todos que possuem DNumero\_Departamento = 2);
- Veja que operações matemáticas também podem ser feitas caso as colunas sejam números;

### FORMA 3 - OMITINDO A CLÁUSULA WHERE

*Ex:* **UPDATE** DEPTO\_LOCALIZACOES  
      **SET** DLocalizacao = "Cidade de Pallet";

- No exemplo, apesar de não usar a Cláusula WHERE parecer algo prático, esta prática precisa ser usada com cautela para não impactar registros indesejados;
- Alguns Bancos, por padrão, não deixam essa operação ser realizada, buscando proteger os usuários. Configurações extras podem ser necessárias para usá-la.





## 4.9. Exclusão de Dados com DELETE

*O comando DELETE FROM é utilizado para se excluir registros de uma determinada tabela do Banco.*

São 2 as principais formas de se fazer uso deste comando:

### FORMA 1 - EXCLUINDO UM ÚNICO REGISTRO

```
DELETE FROM Departamento  
WHERE DNumero = 3;
```

### FORMA 2 - EXCLUINDO MÚLTIPLOS REGISTROS

Ex: 

```
DELETE FROM Empregado  
WHERE Salario > 10000 AND DNumero_Departamento = 2;
```

- Observe que dentro da Cláusula WHERE podemos usar operadores lógicos (AND, OR e NOT) e de comparação (=, <, <=, >, >=, <> (diferente)) para realizarmos filtros mais precisos. Estes operadores também podem ser utilizados no WHERE de atualizações (UPDATE) e consultas (SELECT);
- É importante ressaltar que, se uma chave estrangeira estiver configurada como ON DELETE CASCADE, se a chave primária pelo qual ela faz referência for excluída, todos os registros que fazem referência a esta chave também serão;



## 4.9. Exclusão de Dados com DELETE

### Exercício Proposto

Crie os seguintes comandos DML propostos sobre o Esquema abaixo:

#### EMPREGADO

PNome	MInicial	UNome	<u>SSN</u>	DataNasc
Endereco	Sexo	Salario	SSN_Supervisor	DNumero_Departamento

#### DEPARTAMENTO

DNome	<u>DNumero</u>	SSN_Empregado	DataInicio
-------	----------------	---------------	------------

#### DEPTO\_LOCALIZACOES

<u>LNumero</u>	DLocalizacao	DNumero_Departamento
----------------	--------------	----------------------

#### PROJETO

PNome	<u>PNumero</u>	PLocalizacao	DNumero_Departamento
-------	----------------	--------------	----------------------

#### TRABALHA\_EM

<u>SSN_Empregado</u>	<u>PNumero_Projeto</u>	Horas
----------------------	------------------------	-------

#### DEPENDENTE

<u>SSN_Empregado</u>	<u>Nome_Dependente</u>	Sexo	DataNasc	Parentesco
----------------------	------------------------	------	----------	------------

- 1) Insira dois novos registros para Departamento preenchendo apenas 3 campos possíveis;
- 2) Insira um terceiro registro para Departamento apenas com nome e número;
- 3) Insira dois 2 registros preenchendo apenas 5 campos e 2 registros completos de Empregado;
- 4) Insira 3 novos registros para Projeto preenchendo apenas nome e número;
- 5) Atualize quem é o supervisor de 3 empregados que foram cadastrados;
- 6) Atualize quem é o gerente de cada um dos Departamentos que foram cadastrados;
- 7) Atualize a localização e o departamento de cada projeto cadastrado;
- 8) Exclua todos os projetos que pertençam a um determinado departamento específico.

**FIM  
DO  
CAPÍTULO 4  
(Parte 2)**

