



# Agenda

## **1. Comandos DML**

**1. Insert**

**2. Delete**

**3. Update**

**4. Commit**

**5. Rollback**

**6. Select**



# INSERT

- Adiciona uma linha na tabela existente.

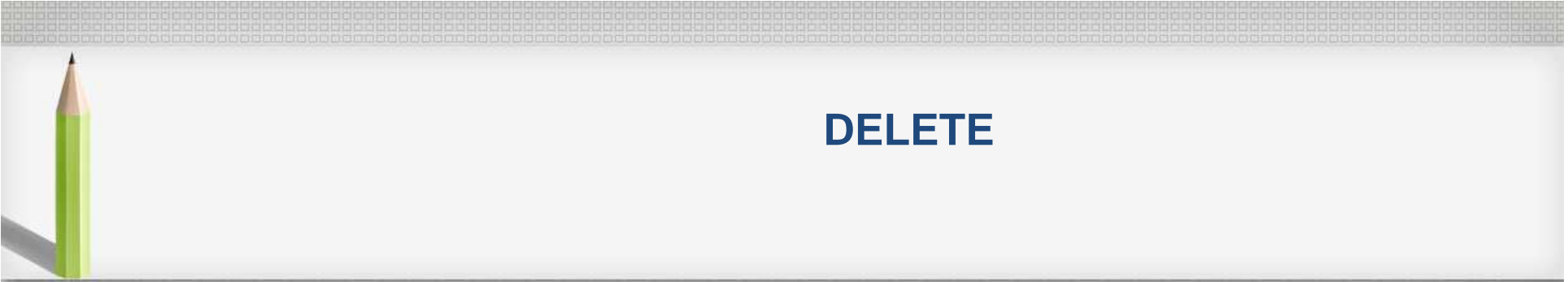
- Sintaxe:

- ◆ INSERT INTO tabela (atributo1, atributo2, ..., atributon)  
VALUES valor1, valor2, ..., valorn) ou comando select

- Exemplo:

- ◆ INSERT INTO Empregado (matricula, nomeemp, depart,  
salario) VALUES (2, 'Jose Lopes', 1, 2500);

- ◆ INSERT INTO Empregado VALUES (1, 'Maria dos Santos',  
'Rua Carlos Gomes, n.20', 1, 3000);



- **Remove linhas na tabela existente.**
- **Sintaxe:**
  - ◆ **DELETE FROM tabela WHERE <condição>;**  
**<condição> = <atributo> <operador> <constante>**  
**ou**  
**<atributo> <operador> <atributo>**  
**ou**  
**<atributo> <operador> <comando select>**
- **Exemplo:**
  - ◆ **DELETE FROM Empregado;**
  - ◆ **DELETE FROM Empregado WHERE nomeemp ='Anildo Mattos';**

## ■ Sintaxe:

◆ **DELETE FROM tabela WHERE <condição>;**  
**<condição> = <atributo> <operador> <constante>**  
**ou**  
**<atributo> <operador> <atributo>**  
**ou**  
**<atributo> <operador> <comando select>**

## ■ Exemplo:

❖ **DELETE FROM Empregado;**

```
◆ DELETE FROM Empregado WHERE nomeemp ='Anildo Mattos';
```





## UPDATE

■ Modifica valores de um ou mais atributos de uma ou mais linhas selecionadas.

■ Sintaxe:

◆ UPDATE tabela SET <atribuições> WHERE <condição>;  
    <atribuições> = <atributo=constantes>  
                    ou  
                    <atributo=atributo>  
                    ou  
                    <atributo=operações de atributos e  
                    constantes>

■ Exemplo:

◆ UPDATE Empregado SET salario = 2000 WHERE depart = 1;  
◆ UPDATE Empregado SET salario = salario\*1.1;  
◆ UPDATE TrabalhaEm SET horas=20 WHERE codproj=01 and matemp=11111;



# COMMIT e ROLLBACK

## ■ COMMIT

- ◆ Utilizado para confirmar uma transação.

## ■ ROLLBACK

- ◆ Utilizado para desfazer uma transação.



## SELECT

- Instrução básica para recuperar informações no banco de dados. Retorna um result set. Navegação é decidida pelo sistema.

- Sintaxe:

**SELECT** <lista de atributos e funções>

**FROM** <lista de tabelas>

**[WHERE** <condições>

**[GROUP BY** <atributos de agrupamento>

**[HAVING** <condição de agrupamento>

**[ORDER BY** <list de atributos>];





## SELECT

- Exemplo:

- SELECT nome, endereco FROM empregado; → Projeção

- SELECT \* FROM empregado;

- SELECT DISTINCT salario FROM empregado;

- Condições do comando WHERE

- Operações de comparação (=, >, <)

- Faixas (BETWEEN e NOT BETWEEN)

BETWEEN – usado para especificar uma faixa inclusiva; os valores extremos da faixa também são incluídos na busca.

NOT BETWEEN – Negação do BETWEEN. Exclui os valores especificados na faixa, inclusive os extremos.



## SELECT

- Correspondência de caracteres (LIKE e Not LIKE)
- Valores desconhecidos (IS NULL e IS NOT NULL)
- Listas (IN e NOT IN)
- Combinações com AND e OR
- NOT pode negar qualquer expressão booleana e chaves como LIKE, NULL, BETWEEN e IN





## SELECT

- Selecione a matricula e nome dos empregados com codigo de departamento entre 1 e 4.

- ◆ **Select** matricula, nome **from** empregado **where** coddepart **between** 1 and 4;

- Selecione a matricula e nome dos empregados que ganham salário fora da faixa de 1.000,00 e 3.000,00 reais.

- **Select** matricula, nomeemp, salario **from** empregado **where** salario **not between** 1000 and 3000;

- Selecione o nome e bairro de empregados que começam com as letras “MAR”

- ◆ **Select** empregado.nomeemp **from** empregado **where** empregado.nomeemp **like** “MAR%”



## SELECT

- Selecione o nome dos empregados dos departamento que começam com as letras “MAR” e de 5 caracteres
  - ◆ Select nome from empregado where nome like “MAR\_ \_”
- Selecione o nome e bairro de empregados, cujos nomes começam com qualquer letra entre M e P, inclusive
  - Select nome, bairro from empregado where nome like “[M-P]”



## SELECT

- Selecione a matricula, nome e bairro dos empregados com matricula na lista (1, 2, 3)

- ◆ **Select** matricula, nome, **from** empregado **where** matricula **in** (1, 2, 3)

- Selecione a matricula, nome e bairro dos empregados que a matricula não esteja na lista (1, 2, 3)

- ◆ **Select** matricula, nomeemp **from** empregado **where** matricula **not in** (1,2, 3)





## SELECT

- Selecione o código e nome de departamentos, cujo código seja 1 e o nome Informática
  - **Select** código, nome **from** departamento **where** código = 1 **and** nome = 'Informática';
- Selecione o código e nome de departamentos, cujo código seja 1 ou o nome Administração
  - **Select** código, nome **from** departamento **where** código = 1 **or** nome = 'Administração'



# SELECT

- Renomeando colunas

- Pode-se estabelecer um outro nome para ser utilizado na saída da declaração select, ao invés do nome da coluna.

- Sintaxe

- **Select** <cabecalho\_da\_coluna as nome\_da\_coluna>

- Exemplo

- **Select** empregado.nome as 'Nome do Empregado' **from** empregado
- **Select** 'Nome do Empregado', nome, 'Bairro', bairro **from** empregado