

FACEC

F a c u l d a d e

Cont. Linguagem SQL

Professor: Yuri Ferreira

➤ Revisão aula anterior:

- Visões;
- Índices;
- Triggers;

- Conteúdo:
 - Funções e Procedures;
 - Exercícios;

➤ Funções e Procedimentos:

- Desde a SQL:1999 as estruturas de ***functions*** e ***procedures*** são aceitas, com isso oferecem o propósito geral de uma linguagem de programação;
- Objetivo é **armazenar** e **executar** a lógica empresarial em construções procedurais dentro do banco de dados;
- Uma das vantagens dessa abordagem é que possibilita um **ponto único** de mudança, caso as regras empresariais mudem, e também várias aplicações podem **acessar** este recurso;

➤ **Funções e Procedimentos:**

- A instrução tem a forma: **begin ... end** e pode conter várias instruções SQL dentro desta declaração;
- **Variáveis** locais podem ser declaradas;
- É possível utilizar estruturas de **loop** como **while**, **repeat** e **for**;
- A instrução **leave** pode ser utilizada para sair do loop;
- Instruções condicionais também são aceitas, como a estrutura **If-then-else**;

➤ Funções e Procedimentos:

- No entanto cada vendedor de SGDB descreve sua sintaxe para criar funções e procedimentos;
- Ex: Definições no PostgreSQL:

```
CREATE [OR REPLACE] FUNCTION nome_funcao(p1 tipo, p2 tipo)
  RETURNS type AS $$
BEGIN
  -- logica
END;
$$
LANGUAGE plpgsql;
```

- **Funções e Procedimentos:**
- **Tipos de Parâmetros:**
- Existem 4 tipos no PostgreSQL: **IN**, **OUT**, **INOUT** e **VARIADIC**;
- Por padrão os parâmetros que não são especificados são implicitamente de entrada (**IN**);
- A Função pode retornar mais de um parâmetro de saída (**OUT**);
- Quando especificado parâmetro(s) de saída (**OUT**), não é necessário a declaração “**returns**” depois do cabeçario da função;

➤ **Funções e Procedimentos:**

➤ **Ex: uma função simples de incremento:**

```
create function inc(val integer) returns integer as $$  
begin  
    return val + 1;  
end; $$  
language plpgsql;  
  
-- executando a função  
select inc(1);
```

- **Funções e Procedimentos:**
- **Estrutura de Condição IF-THEN-ELSE:**

```
IF condicao1 THEN  
    declaracao1;  
ELSIF condicao2 THEN  
    declaracao2;  
ELSE  
    declaracao3;  
END IF;
```

➤ Funções e Procedimentos:

➤ Exemplo utilizando bloco de código anônimo do PostgreSQL:

```
DO $$  
DECLARE  
    a integer := 10;  
    b integer := 10;  
BEGIN  
    IF a > b THEN  
        RAISE NOTICE 'a é maior que b';  
    ELSIF a < b THEN  
        RAISE NOTICE 'a é menor que b';  
    ELSE  
        RAISE NOTICE 'a é igual a b';  
    END IF;  
END $$;
```

- **Funções e Procedimentos:**
- Estruturas de Loop:
- **Loop simples, While, For;**
- **Cursor** serve como um ponteiro para buscar registros em ordem crescente ou decrescente;
- Os Loops geralmente são utilizados em procedures, para atualizações de informações em massa e que precisam de cálculos entre outras tabelas;

- **Funções e Procedimentos:**
- Estrutura instrução **Procedure**: disponível no PostgreSQL a partir da versão 11;

```
create procedure transfer_sal(int, int, numeric)
language plpgsql
as $$
begin
update ...;

update ...;
commit;
end;
$$;
-- Executando a Procedure
call transfer_sal(5,4,200);
```

➤ **Funções e Procedimentos:**

- Por definição as **funções** retornam um valor e são utilizadas para fazerem cálculos;
- Os **procedimentos** por sua vez, são mais utilizados para fazerem **atualizações** (insert, update, delete) contendo controles de transações com as tags: **Commit** e **Rollback**;

- **Funções e Procedimentos:**
- **Mensagens e geração de excessões:**
- **Raise Notice:** pode ser utilizado visualizar a depuração de uma função;
- **Raise Exception:** Dispara uma exceção e para a execução da função;

➤ Exercícios:

- 1) Criar uma função que recebe o num_depto como parâmetro e retorna o tamanho do departamento (Pequeno, Médio ou Grande), seguindo os critérios:
 - Pequeno menor que 3 funcionários;
 - Médio de 3 a 5 funcionários;
 - Grande acima de 5 funcionários;

➤ Referências:

- SILBERSCHATZ, A.; KORTH, F.; SUDARSHA, S. Database System Concepts. 6. ed. Nova York: MC Graw Hill, 2011.
- ELMASRI, R.; NAVATHE B. Sistemas de banco de dados. 6. Ed. São Paulo, SP: Pearson Addison-Wesley, 2011.