



Stored Procedures: Funções em Linguagem SQL



Professor

Fernando Martins Medeiros

Estruturas condicionais

FOR

A instrução *WHILE* repete uma seqüência de instruções enquanto a expressão de condição for avaliada como verdade. A condição é verificada logo antes de cada entrada no corpo do laço.

Sintaxe:

```
[<<rótulo>>]  
FOR nome IN [ REVERSE ] expressão .. expressão LOOP  
    instruções  
END LOOP;
```

```
for counter in 1..6 by 2 loop  
    raise notice 'counter: %', counter;  
end loop;
```

Exemplo:

```
FOR i IN 1..10 LOOP  
    -- algum processamento  
    RAISE NOTICE 'i é %', i;  
END LOOP;
```

```
FOR i IN REVERSE 10..1 LOOP  
    -- algum processamento  
END LOOP;
```

Estruturas condicionais

RECORD

Código:

```
CREATE OR REPLACE FUNCTION teste_record()
RETURNS void AS $$
DECLARE
    modelo RECORD;
BEGIN
    FOR modelo IN SELECT * FROM modelo ORDER BY id LIMIT 5 LOOP
        RAISE INFO '|MODELO| ID...: %', modelo.id;
        RAISE INFO '|MODELO| NOME.: %', modelo.nome;
        RAISE INFO '|-----';
    END LOOP;
END;
$$ LANGUAGE 'plpgsql';

SELECT teste_record();
```

Saída:

```
INFO: |MODELO| ID...: 1
INFO: |MODELO| NOME.: Integra GS 1.8
INFO: |-----|
INFO: |MODELO| ID...: 2
INFO: |MODELO| NOME.: Legend 3.2/3.5
INFO: |-----|
INFO: |MODELO| ID...: 3
INFO: |MODELO| NOME.: NSX 3.0
INFO: |-----|
INFO: |MODELO| ID...: 4
INFO: |MODELO| NOME.: MARRUÁ 2.8 12V 132cv TDI Diesel
INFO: |-----|
INFO: |MODELO| ID...: 5
INFO: |MODELO| NOME.: 145 Elegant 1.7/1.8 16V
INFO: |-----|
```

Estruturas condicionais

RECORD

Código:

```
CREATE OR REPLACE FUNCTION teste_record2()  
RETURNS void AS $$  
DECLARE  
    modelo RECORD;  
BEGIN  
  
    SELECT * INTO modelo FROM modelo WHERE id = 20;  
  
    RAISE INFO ' |-----';  
    RAISE INFO ' |MODELO| ID...: %', modelo.id;  
    RAISE INFO ' |MODELO| NOME.: %', modelo.nome;  
    RAISE INFO ' |-----';  
  
END;  
$$ LANGUAGE 'plpgsql';  
  
SELECT teste_record2();
```

Saída:

```
INFO:  |-----  
INFO:  |MODELO| ID...: 20  
INFO:  |MODELO| NOME.: Spider 2.0/3.0  
INFO:  |-----
```

Estruturas condicionais

CURSOR

```
CREATE OR REPLACE FUNCTION teste_cursor()
RETURNS void AS $$
DECLARE
    cur_marca CURSOR FOR SELECT * FROM marca ORDER BY id LIMIT 3;
BEGIN
    FOR marca IN cur_marca LOOP
        RAISE INFO ' |-----';
        RAISE INFO ' |MARCA1x| ID...: %', marca.id;
        RAISE INFO ' |MARCA1x| NOME.: %', marca.nome;
        RAISE INFO ' |-----';
    END LOOP;

    FOR marca IN cur_marca LOOP
        RAISE INFO ' |-----';
        RAISE INFO ' |MARCA2x| ID...: %', marca.id;
        RAISE INFO ' |MARCA2x| NOME.: %', marca.nome;
        RAISE INFO ' |-----';
    END LOOP;
END;
$$ LANGUAGE 'plpgsql';

SELECT teste_cursor();
```

Saída:

```
INFO: |-----
INFO: |MARCA1x| ID...: 1
INFO: |MARCA1x| NOME.: Acura
INFO: |-----
INFO: |-----
INFO: |MARCA1x| ID...: 2
INFO: |MARCA1x| NOME.: Agrale
INFO: |-----
INFO: |-----
INFO: |MARCA1x| ID...: 3
INFO: |MARCA1x| NOME.: Alfa Romeo
INFO: |-----
INFO: |-----
INFO: |MARCA2x| ID...: 1
INFO: |MARCA2x| NOME.: Acura
INFO: |-----
INFO: |-----
INFO: |MARCA2x| ID...: 2
INFO: |MARCA2x| NOME.: Agrale
INFO: |-----
INFO: |-----
INFO: |MARCA2x| ID...: 3
INFO: |MARCA2x| NOME.: Alfa Romeo
INFO: |-----
```

Estruturas condicionais

Erros e mensagens

Código:

```
CREATE OR REPLACE FUNCTION erros_mensagens()  
RETURNS void AS $$  
DECLARE  
BEGIN  
  
    RAISE INFO 'Essa é uma mensagem do tipo informação gerada em %', now() ;  
    RAISE LOG 'PEssa é uma mensagem do tipo log gerada em %', now();  
    RAISE DEBUG 'Essa é uma mensagem do tipo debug gerada em %', now();  
    RAISE WARNING 'Essa é uma mensagem do tipo atenção gerada em %', now();  
    RAISE NOTICE 'Essa é uma mensagem do tipo notícia gerada em %', now();  
    RAISE EXCEPTION 'Essa é uma mensagem do tipo exceção gerada em %', now();  
  
END;  
$$ LANGUAGE 'plpgsql';  
  
SELECT erros_mensagens();
```

Saída:

```
INFO: Essa é uma mensagem do tipo informação gerada em 2021-10-28 17:14:05.913534-03  
WARNING: Essa é uma mensagem do tipo atenção gerada em 2021-10-28 17:14:05.913534-03  
NOTICE: Essa é uma mensagem do tipo notícia gerada em 2021-10-28 17:14:05.913534-03
```

```
ERROR: Essa é uma mensagem do tipo exceção gerada em 2021-10-28 17:14:05.913534-03
```

```
***** Error *****
```

```
ERROR: Essa é uma mensagem do tipo exceção gerada em 2021-10-28 17:14:05.913534-03  
SQL state: P0001
```

Estruturas condicionais

Tratamento de excessões

Código:

```
CREATE OR REPLACE FUNCTION lanca_multa_excessao(var_id_multa int, var_id_ano int,  
var_id_condutor int, var_data_autuacao DATE)  
RETURNS boolean AS $$  
DECLARE  
BEGIN  
  
    INSERT INTO multa (id, id_ano, id_condutor, data_autuacao)  
    VALUES (var_id_multa, var_id_ano, var_id_condutor, var_data_autuacao);  
    RAISE INFO 'Multa %: Lançando multa com data de autuação em %', var_id_multa, var_data_autuacao;  
  
    RETURN true;  
  
    EXCEPTION  
        WHEN unique_violation THEN  
            RAISE INFO 'Multa % com data de autuação % já lançada no banco de dados', var_id_multa, var_data_autuacao;  
            RETURN false;  
END;  
$$ LANGUAGE 'plpgsql';  
  
SELECT lanca_multa_excessao(5151, 200, 1, '2021-09-10');
```

1ª execução

	lanca_multa_excessao boolean
1	t

INFO: Multa 5252: Lançando multa com data de autuação em 2021-09-10

2ª execução

	lanca_multa_excessao boolean
1	f

INFO: Multa 5252 com data de autuação 2021-09-10 já lançada no banco de dados