



Banco de Dados II

Prof. Vinícius Alves Hax



Na aula anterior

- Funções
- Variável found



Na aula de hoje

- Seleção com CASE
- Outros comandos de repetição




Seleção com CASE




Case

- A construção case é equivalente ao switch de várias linguagens de programação
- Muito útil quando se precisa comparar uma variável com várias possibilidades
- Ela pode ser usada de duas formas:
 - Para comparar um valor com outros
 - Para testar várias expressões



Comparando uma variável com
um conjunto de valores



```
CASE var
    WHEN valor1 THEN
        Instrução1;
    WHEN valor2 THEN
        Instrução2;
    WHEN valor3 THEN
        Instrução3;
    ELSE
        Instrução padronizada;
END CASE;
```

Exemplo 1 de case

```
do $$  
declare  
numero integer;  
begin  
select level from teacher into numero where id = 1;  
case numero  
when 100 then  
    raise notice 'Nível é 100';  
when 90 then  
    raise notice 'Nível é 90';  
else  
    raise notice 'Nível não é 100 e nem 90';  
end case;  
end; $$
```




CASE

WHEN expressão_1 THEN

Instrução1;

WHEN expressão_2 THEN

Instrução2;

WHEN expressão_3 THEN

Instrução3;


ELSE

Instrução padronizada;

END CASE;

Exemplo 2 de case

```
do $$  
declare  
numero integer;  
begin  
select level from teacher into numero where id = 2;  
case  
when numero <= 100 and numero > 90 then  
    raise notice 'Nível está entre 100 e 90';  
when numero <= 90 and numero > 80 then  
    raise notice 'Nível está entre 90 e 80';  
else  
    raise notice 'Nível é menor ou igual que 80';  
end case;  
end; $$
```



Repetição com while

Repetição com while

- Vimos que quando queremos percorrer os resultados de uma query usamos a estrutura for
- E quando quisermos fazer uma repetição independente de uma query?
 - Podemos usar a estrutura while



Sintaxe básica do while

```
while TESTE loop  
    COMANDOS;  
end loop;
```

Exemplo while

```
do $$  
declare  
    numero integer;  
begin  
    numero := 0;  
    while numero < 10 loop  
        raise notice 'Número vale %', numero;  
        numero := numero + 1;  
    end loop;  
end; $$
```

Exercício

- Faça um programa em PL/pgSQL que (usando as estruturas case e while) percorra todos os números no intervalo entre 1 e 10 e para cada número e para cada número faça o teste se ele é divisível por 2, por 3, ou por ambos. Mostre uma mensagem correspondente. Ou seja, se o número for divisível por 2 deverá aparecer “Número divisível por 2” e o mesmo para 3.
 - Dica 1: o operador de resto da divisão é o %
 - Dica 2: lembre-se que alguns números são divisíveis por 2 E por 3 também



Referências

- Case
<https://www.postgresqltutorial.com/postgresql-plpgsql/plpgsql-case-statement/>
- Criando funções
<https://www.postgresqltutorial.com/postgresql-plpgsql/postgresql-create-function/>