

```

1  /* Aktv 3 - 5-  Elabore uma função que retorne a quantidade e a soma do valor total dos pedidos feitos por um cliente
2  passando como parâmetro o nome ou parte do nome, em um intervalo de tempo.  */
3  CREATE OR REPLACE TYPE resultado_cli AS VARRAY(5) OF NUMBER ;
4  CREATE OR REPLACE FUNCTION qtde_pedidos ( vcli IN cliente.nome_fantasia%TYPE ,
5  vini IN pedido.dt_hora_ped%TYPE, vfim IN pedido.dt_hora_ped%TYPE)
6  RETURN resultado_cli
7  IS vqtde resultado_cli := resultado_cli() ;
8  vargumento cliente.nome_fantasia%TYPE := '%' || UPPER(vcli) || '%' ;
9  vsetem SMALLINT := 0 ;
10 BEGIN
11 vqtde.EXTEND(2) ;
12 SELECT COUNT(*) INTO vsetem FROM cliente
13 WHERE upper(nome_fantasia) LIKE vargumento ;
14 IF vsetem = 1 THEN
15 SELECT COUNT(*), SUM(p.vl_total_ped) INTO vqtde(1), vqtde(2)
16 FROM pedido p, cliente c
17 WHERE p.cod_cli = c.cod_cli
18 AND upper(nome_fantasia) LIKE vargumento
19 AND p.dt_hora_ped BETWEEN vini AND vfim ;
20 ELSIF vsetem > 1 THEN
21 RAISE_APPLICATION_ERROR ( -20011, 'Existe mais de um cliente com este nome. Seja mais específico!');
22 END IF ;
23 RETURN vqtde ;
24 EXCEPTION
25 WHEN NO_DATA_FOUND THEN
26 RAISE_APPLICATION_ERROR ( -20029, 'Cliente não encontrado');
27 END ;
28
29 SELECT qtde_pedidos('gun', current_date - 1000, current_date) FROM DUAL;
30
31 SELECT * FROM cliente ;
32
33 --Aktv 3 - 6 - Elabore uma função que retorne o valor total vendido, calculado em reais,
34 --para um determinado Gerente de uma equipe de vendas em um período de tempo,
35 --tendo como parâmetros de entrada o código do gerente e as datas de início e término do período, ou seja,
36 --quanto foi vendido pelos vendedores que são gerenciados por este gerente.
37 --Considere que na tabela de cargo existe o cargo "Gerente Vendas". Faça as validações necessárias.
38
39 ALTER TRIGGER valida_admissao DISABLE ;
40 SELECT * FROM cargo ; -- gerente vendas = 9
41 SELECT * FROM funcionario WHERE REG_FUNC GERENTE = 9 ;
42 UPDATE funcionario SET cod_cargo = 9 WHERE cod_cargo = 10 ;
43
44 SELECT f.*
45 FROM funcionario f, cargo c
46 WHERE f.cod_cargo = c.cod_cargo
47 AND UPPER( c.nome_cargo) LIKE '%GERENTE DE VENDAS%' ;
48
49 -- função
50 CREATE OR REPLACE FUNCTION total_gerente_vendas ( vgerente IN funcionario.reg_func%TYPE,
51 vini IN pedido.dt_hora_ped%TYPE, vfim IN pedido.dt_hora_ped%TYPE)
52 RETURN NUMBER IS
53 vtotal pedido.vl_total_ped%TYPE := 0 ;
54 vsegerente cargo.nome_cargo%TYPE ;
55 BEGIN
56 SELECT c.nome_cargo INTO vsegerente
57 FROM funcionario f, cargo c
58 WHERE f.cod_cargo = c.cod_cargo
59 AND f.reg_func = vgerente ;
60
61 IF UPPER (vsegerente) NOT LIKE '%GERENTE DE VENDAS%' THEN
62 RAISE_APPLICATION_ERROR ( -20001, ' Funcionario não é gerente de Vendas !!' );
63 ELSE
64 SELECT SUM(p.vl_total_ped) INTO vtotal
65 FROM pedido p, funcionario f
66 WHERE p.reg_func_vendedor = f.reg_func
67 AND f.reg_func_gerente = vsegerente
68 AND p.dt_hora_ped BETWEEN vini AND vfim ;
69 END IF;
70 RETURN vtotal ;
71 END ;
72
73 -- testando
74 SELECT total_gerente_vendas (2, current_timestamp - 500, current_timestamp) FROM dual ;

```