|  |  |  |  |
| --- | --- | --- | --- |
| RM: |  | Nome: |  |

**INSTRUÇÕES:**

**Lista de Exercícios Práticos**

**ATENÇÃO:**

* Entrega no portal:
  + **Para todas as questões deverá ser incluso o bloco PL/SQL e o print de execução dele.**

1. Faça uma procedure chamada PRC\_INSERE\_PEDIDO\_RMNNNNN para inserir dados na tabela de pedidos, valide os campos para que caso eles estejam incompletos lance um RAISE\_APPLICATION\_ERROR mostrando os dados que não estejam de acordo com as regras:

DAT\_PEDIDO não pode ser maior que a DAT\_ENTREGA

DAT\_PEDIDO não pode ser superior a SYSDATE

VALOR PEDIDO não pode ser 0 ou inferior

Inclua tratamento de exceções necessárias, mas nestas use RAISE\_APPLICATION\_ERROR para mandar uma EXCEPTION para as rotinas que chamem esta procedure. (nota: cuidado com OTHERS pois caso um RAISE\_APPLICATION\_ERROR seja feito na seção de BEGIN irá provocá-lo)

1. Faça uma função chamada FUN\_CONSULTA\_PED\_RMNNNNN, para consultar PEDIDO usando como parâmetro o código do pedido. Use para fazer a consulta SELECT INTO, e use os EXCEPTIONS necessários.
2. Faça uma Função chamada FUN\_TOTAL\_PEDIDO\_RMNNNNN, para que faça uma totalização dos pedidos a partir do parâmetro do CODIGO DO CLIENTE. Crie os tratamentos de exceptions necessários. Cuidado não use RAISE\_APPLICATION\_ERROR.
3. Altere a função do exercício 2 para que tenha um exception chamado INCONSISTENCIA\_DADOS para que em caso de dados inconsistentes realize um RAISE usando esse EXCEPTION. A função deverá tratar os problemas decorrentes de exception. Mude o nome da função para FUN\_INSERE\_NF2\_RMNNNNN.
4. Faça uma função chamada FUN\_VALIDA\_PRODUTO\_RMNNNNN que receberá um CÓDIGO DE PRODUTO e deverá retornar TRUE se estiver cadastrada e FALSE caso não esteja. Use SELECT ...INTO e realize tratamento de exceções necessárias.

**Instruções:**

Baixe o arquivo do modelo pedido que está na área de apostilas

Descompacte em um diretório (procure não usar a área de trabalho)