|  |  |  |  |
| --- | --- | --- | --- |
| RM: | 553315 | Nome: | Murillo Ferreira Ramos |
| RM: | 553874 | Nome: | Larissa lopes oliveira |
| RM: | 553874 | Nome: | Pedro Luiz Prado |

**INSTRUÇÕES:**

**Avaliação Prática**

**ATENÇÃO:**

* Utilize as tabelas já criadas e preenchidas em sala de aula do arquivo SCRIPT\_MODELO\_PEDIDO.
* O trabalho pode ser feito em grupos de até 3 pessoas.
* A entrega deverá ser realizada na área de submissão de trabalhos no portal do aluno até o dia 06/11/2024, no horário da aula.
* A lista contém 10 questões, cada uma valendo 1 ponto. Todas as questões têm o mesmo peso na avaliação.
* Para cada questão, envie os scripts completos utilizados para a resolução.
* Inclua um print de compilação e teste de todos os exercícios. A ausência dos prints resultará em uma dedução de 0,25 ponto por exercício sem o respectivo print.

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

##### Questão 1: Procedure para Listar Pedidos de um Cliente

**Descrição**: Crie uma procedure chamada listar\_pedidos\_cliente que recebe o código de um cliente e exibe seus pedidos e valores. Se o cliente não existir, lance uma exceção personalizada. Use um **cursor** para listar os pedidos e um **IF** para verificar se há registros antes de exibir os pedidos.

**Requisitos**:

* **Exceção**: Lance uma exceção personalizada se o cliente não for encontrado.
* **JOIN**: Use INNER JOIN para combinar pedido com historico\_pedido.
* **IF**: Verifique se o cliente possui pedidos.
* **Cursor**: Utilize um cursor para iterar sobre os pedidos.

##### Questão 2: Função para Calcular Total em Estoque de Produtos Ativos

**Descrição**: Desenvolva uma função chamada calcular\_estoque\_total que retorna o estoque total dos produtos ativos. Lance uma exceção se não houver produtos ativos. Use **variáveis** para somar o estoque e um **LOOP** para iterar pelos produtos ativos.

**Requisitos**:

* **Exceção**: Lance uma exceção se não houver produtos ativos.
* **JOIN**: Utilize LEFT JOIN entre produto e estoque\_produto.
* **LOOP**: Itere pelos produtos ativos.
* **Variável**: Armazene o total de estoque.

##### Questão 3: Procedure para Atualizar Preço de Produtos para um Estado com Controle de Erros

**Descrição**: Crie uma procedure chamada atualizar\_preco\_estado que aumenta o preço dos produtos comprados por clientes de um estado específico. Se o estado não tiver clientes, lance uma exceção. Utilize um **WHILE** para iterar sobre os produtos e um **IF** para verificar se o produto está ativo.

**Requisitos**:

* **Exceção**: Lance uma exceção caso o estado não tenha clientes.
* **JOIN**: Use INNER JOIN entre cliente e produto.
* **WHILE**: Utilize WHILE para iterar sobre os produtos.
* **IF**: Verifique se o produto está ativo antes de aplicar o ajuste.

##### Questão 4: Função para Contar Pedidos Cancelados

**Descrição**: Desenvolva uma função chamada contar\_pedidos\_cancelados que recebe o código de um cliente e conta seus pedidos cancelados. Lance uma exceção se o cliente não tiver pedidos. Utilize um **cursor** para iterar pelos pedidos e um **IF** para verificar o status de cancelamento.

**Requisitos**:

* **Exceção**: Lance uma exceção caso o cliente não tenha pedidos.
* **JOIN**: Use INNER JOIN entre cliente e pedido.
* **Cursor**: Utilize um cursor para iterar pelos pedidos.
* **IF**: Verifique se o pedido está cancelado.

##### Questão 5: Procedure para Listar Produtos em Estoque Acima de um Valor com Tratamento de Erros

**Descrição**: Crie uma procedure chamada listar\_produtos\_em\_estoque\_minimo que exibe produtos com estoque acima de um valor mínimo. Se não houver produtos suficientes, lance uma exceção. Utilize um **LOOP** para iterar sobre os produtos e um **IF** para verificar a quantidade.

**Requisitos**:

* **Exceção**: Lance uma exceção se não houver produtos em estoque suficiente.
* **JOIN**: Use LEFT JOIN entre produto e estoque\_produto.
* **LOOP**: Utilize um loop para iterar sobre os produtos.
* **IF**: Verifique se a quantidade em estoque é superior ao valor mínimo.

**R**: Interface gráfica do usuário, Texto, Aplicativo, Email

Descrição gerada automaticamente

##### Questão 6: Função para Calcular Média de Desconto por Pedido com Exceções

**Descrição**: Desenvolva uma função chamada media\_desconto\_pedido que calcula a média dos descontos aplicados aos pedidos de um cliente. Lance uma exceção caso o cliente não tenha descontos aplicados. Utilize um **cursor** para iterar sobre os pedidos e uma **variável** para armazenar a soma dos descontos.

**Requisitos**:

* **Exceção**: Lance uma exceção se o cliente não tiver descontos.
* **JOIN**: Use INNER JOIN entre pedido e historico\_pedido.
* **Cursor**: Utilize um cursor para iterar sobre os descontos.
* **Variável**: Armazene a soma dos descontos.

**R**: Interface gráfica do usuário, Texto, Aplicativo, Email

Descrição gerada automaticamente

##### Questão 7: Procedure para Atualizar Estoque de Produtos com Exceções

**Descrição**: Crie uma procedure chamada atualizar\_estoque\_produto que recebe o código de um produto e a quantidade a ser adicionada. Caso a quantidade final seja negativa, ajuste para zero e lance uma exceção. Utilize um **LOOP** para simular a atualização e um **IF** para verificar o estoque final.

**Requisitos**:

* **Exceção**: Lance uma exceção se o estoque final for negativo.
* **JOIN**: Use INNER JOIN entre produto e estoque\_produto.
* **LOOP**: Simule a atualização repetida para cada unidade.
* **IF**: Verifique se o estoque final é negativo.

**R**: Interface gráfica do usuário, Texto, Aplicativo, Email

Descrição gerada automaticamente

##### Questão 8: Função para Calcular Quantidade Média de Itens por Pedido

**Descrição**: Desenvolva uma função chamada media\_itens\_por\_pedido que calcula a média de itens nos pedidos de um cliente. Lance uma exceção caso o cliente não tenha pedidos. Use um **WHILE** para contar os itens e uma **variável** para armazenar a soma das quantidades.

**Requisitos**:

* **Exceção**: Lance uma exceção se o cliente não tiver pedidos.
* **JOIN**: Use INNER JOIN entre cliente e pedido.
* **WHILE**: Utilize um laço WHILE para contar os itens.
* **Variável**: Armazene a soma das quantidades.

**R**: Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente

##### Questão 9: Procedure para Listar Clientes com Pedidos Acima de um Valor

**Descrição**: Crie uma procedure chamada listar\_clientes\_pedidos\_acima\_valor que exibe os clientes com pedidos acima de um valor especificado. Lance uma exceção se nenhum cliente for encontrado. Utilize um **cursor** para iterar sobre os pedidos e um **IF** para verificar se o valor do pedido excede o limite.

**Requisitos**:

* **Exceção**: Lance uma exceção caso nenhum cliente tenha pedidos acima do valor.
* **JOIN**: Utilize INNER JOIN entre cliente e historico\_pedido.
* **Cursor**: Utilize um cursor para iterar pelos pedidos.
* **IF**: Verifique se o valor do pedido excede o limite.

**R**: Interface gráfica do usuário, Texto, Aplicativo, Email

Descrição gerada automaticamente

##### Questão 10: Função para Calcular Total de Estoque por Tipo de Produto

**Descrição**: Desenvolva uma função chamada calcular\_estoque\_tipo\_produto que recebe o tipo de produto e retorna o total em estoque. Lance uma exceção se não houver produtos desse tipo em estoque. Use um **cursor** para iterar pelos produtos e um **LOOP** para somar a quantidade total.

**Requisitos**:

* **Exceção**: Lance uma exceção se não houver estoque para o tipo de produto.
* **JOIN**: Use INNER JOIN entre produto e estoque\_produto.
* **Cursor**: Utilize um cursor para iterar sobre os produtos.
* **LOOP**: Utilize um loop para somar a quantidade em estoque.

**R**: Interface gráfica do usuário, Texto, Aplicativo, chat ou mensagem de texto, Email

Descrição gerada automaticamente