

AVC – AVALIAÇÃO CONTINUA 2024/2

CURSO: GESTÃO DE TECNOLOGIA DA INFORMAÇÃO (2A)

Professora: Ingrid Batista

Data de Entrega: 06/11/2024

Nota: _____

Nº	RA	Nome Completo
1	5446511	Amanda de Abreu da Silva
2	5476275	Arthur Ygor Vieira dos Santos
3	5528186	Beatriz da Silva Santos
4	5539528	Geovana Silva Rodrigues
5	5449961	John Altemy Silva de Oliveira
6	5364043	Marcos de Souza Ferreira
7	5502861	Thomaz Gabriel Barbosa da Silva
8	5387981	Victor Gabriel Jesus Rodrigues

ORIENTAÇÕES:

1. Esta atividade deve ser composta por no **mínimo 5 e no máximo 8 alunos**.
2. Não deve ter cópias de internet ou IA, a entrega será submetida a um software de plágio.
3. Não serão aceitas entregas após a data estipulada.
4. O documento final deverá ser entregue via teams por TODOS os componentes e uma cópia impressa por grupo.
5. Deverá ser entregue com os seguintes links:
 - a. Código fonte através do GitHub.
 - b. Vídeo explicando o funcionamento e execução do projeto, todos os componentes devem ter participação.

ENTREGA:

Link GitHub	https://github.com/Amanda-Abreu/AVC-Ingrid
Link Vídeo	https://youtu.be/O4LjnP5qTm8?si=g_pguQMBxjErvTWL

ENUNCIADO:

Sistema de Controle de Encomendas de uma Loja Virtual:

Objetivo: Desenvolver um sistema para gerenciar encomendas em uma loja virtual. O sistema deve permitir adicionar produtos a um pedido, calcular o valor total, aplicar descontos.

Regras do Sistema:

- **Produtos:** Cada produto tem um nome, categoria (ex: eletrônicos, vestuário), preço e quantidade em estoque. Os produtos serão armazenados em uma lista de tuplas, onde cada tupla terá o formato (nome, categoria, preço, quantidade). O estoque inicial de cada produto deve ser definido pelo grupo e atualizado conforme as encomendas são feitas.
- **Encomendas:** Cada pedido é feito por um cliente e contém uma lista de produtos selecionados, armazenada em um dicionário. O dicionário tem o nome do cliente como chave e a lista de produtos (tuplas) como valor. O sistema deve permitir que o cliente adicione ou remova produtos de sua encomenda e calcular o total da compra considerando a quantidade de cada item.

Ações do Sistema:

- **Adicionar produtos à encomenda:** O cliente pode escolher produtos da loja. O sistema verifica se há estoque suficiente antes de adicionar ao pedido.
- **Remover produtos da encomenda:** O cliente pode remover itens do pedido antes da finalização.
- **Finalizar compra:** Ao finalizar a compra, o sistema deve reduzir a quantidade de itens no estoque e iniciar uma nova venda.
- **Desconto:** Um desconto deve ser aplicado se o valor total da compra for maior que um determinado valor (por exemplo, 10% de desconto em compras acima de R\$ 500).

Requisitos do Sistema:

- **Condicionais:** Verificar se o produto está disponível em estoque, aplicar descontos conforme o valor total.
- **Laços While e For:** Usar para verificar a quantidade de produtos em estoque e para iniciar uma nova venda após o encerramento da mesma.
 - **Listas e Tuplas:** Os produtos serão armazenados como tuplas, e as encomendas serão gerenciadas como listas de produtos associadas a clientes.
- **Dicionários:** Cada cliente terá um dicionário associando seu nome aos seus pedidos.

Requisitos Técnicos:

- **Gestão de Produtos:** O sistema deve permitir a adição de novos produtos, verificar a disponibilidade e atualizar a quantidade no estoque conforme as compras são feitas.
- **Controle de Encomendas:** Implementar funções para adicionar/remover produtos de um pedido, calcular o valor total e aplicar descontos.

QUESTIONÁRIO:

1. **O que acontece se o estoque não for suficiente? Como o sistema responde? Anexe um print.**

Quando a quantidade solicitada pelo usuário excede o estoque disponível, o sistema informa que a quantidade solicitada não pode ser atendida. Ele exibe uma mensagem que indica o número de itens disponíveis.

Comportamento do Sistema

Por exemplo, se um usuário tentar adicionar 5 unidades de um produto que só tem 3 disponíveis, o sistema responderá com algo como:

```
Estoque insuficiente! Apenas 3 produtos disponíveis.
```

```
Estoque atual:
```

```
Notebook (Categoria: Eletrônicos) -- Preço: R$3500.00 -- Quantidade: 10 -- Reservado: 0  
Camiseta (Categoria: Vestuário) -- Preço: R$50.00 -- Quantidade: 100 -- Reservado: 0  
Smartphone (Categoria: Eletrônicos) -- Preço: R$2000.00 -- Quantidade: 15 -- Reservado: 0  
Calça Jeans (Categoria: Vestuário) -- Preço: R$120.00 -- Quantidade: 50 -- Reservado: 0
```

```
Escolha o número do produto que deseja comprar: 1
```

```
Digite a quantidade que deseja comprar: 15
```

```
Estoque insuficiente! Apenas 10 produtos disponíveis.
```

Campus Guarulhos: Rua João Gonçalves, 471 - Centro Guarulhos - SP

Campus Interlagos: Rua Prof. Enéas de Siqueira Neto, 340 - Jardim das Imbuías, São Paulo - SP

Campus Metrô Adolfo Pinheiro: Rua Isabel Schmidt, 349 - Santo Amaro, São Paulo - SP

Campus Metrô Marechal Deodoro: Rua Dr. Gabriel dos Santos, 30 - Santa Cecília, São Paulo - SP

2. Como o sistema atualiza o estoque quando um produto é removido do pedido? Anexe um print.

Atualmente, o sistema não possui uma funcionalidade explícita para remover itens do carrinho, mas se isso fosse implementado, o ideal seria que, ao remover um item do carrinho, a quantidade correspondente fosse subtraída da quantidade reservada no estoque.

Comportamento do Sistema

Se um produto for removido, o sistema poderia executar as seguintes etapas:

Identificar o Produto: Encontrar o produto que está sendo removido do carrinho.

Atualizar o Estoque: Subtrair a quantidade do produto reservado para devolver ao estoque disponível.

Suponha que um usuário tenha adicionado 5 camisetas ao carrinho, mas decide remover 2. O sistema poderia responder assim:

```
Carrinho de compras:
1. Camiseta - Quantidade: 5 - Preço Total: R$250.00

Removendo 2 camisetas do carrinho...

Estoque atualizado:
1. Camiseta (Categoria: Vestuário) -- Preço: R$50.00 -- Quantidade: 100 -- Reservado: 3
```

3. O que acontece se o cliente tentar finalizar a compra com um produto que não está mais disponível em estoque? Anexe um print.

Se o cliente tentar finalizar a compra com um produto que não está mais disponível em estoque, a função finalizar_compra não permitirá a finalização, já que ela verifica se o carrinho contém produtos.

No entanto, a verificação específica para produtos que já estão esgotados não está explicitamente no código da função finalizar_compra.

Se o carrinho estiver vazio, o sistema exibe uma mensagem informando que o carrinho está vazio e não finaliza a compra.

Caso o produto tenha sido adicionado ao carrinho, mas seu estoque foi esgotado antes da finalização, isso não é tratado diretamente na função.

Portanto, a função permite a finalização, mas não atualiza corretamente o estoque.

```
Carrinho de compras:  
1. Notebook - Quantidade: 1 - Preço Total: R$3500.00  
Total do carrinho: R$3500.00  
Compra finalizada com sucesso! Estoque atualizado.
```

4. **Como o sistema calcula e aplica o desconto? Mostre um exemplo de cálculo e um print do sistema funcionando e do código.**

Função para calcular o desconto

```
def calcular_desconto(preco_compra):
```

```
    if preco_compra >= 1000:
```

```
        desconto = preco_compra * 0.10 # 10% de desconto
```

```
        preco_final = preco_compra - desconto
```

```
        return preco_final, desconto
```

```
    else:
```

```
        return preco_compra, 0 # Sem desconto
```

Entrada de dados

```
preco_compra = float(input("Digite o valor da compra: R$ "))
```

Calculo

Cálculo


```
preco_final, desconto = calcular_desconto(preco_compra)
```

```
# Saída de dados
```

```
print(f"\nValor da compra: R$ {preco_compra:.2f}")
```

```
print(f"Desconto aplicado: R$ {desconto:.2f}")
```

```
print(f"Preço final após desconto: R$ {preco_final:.2f}")
```

5. Como o sistema lida com a adição de um novo produto que já existe no estoque? Anexe um print.

```
# Função para remover item do carrinho
```

```
def remover_item(carrinho, indice):
```

```
    if 0 <= indice < len(carrinho):
```

```
        item_removido = carrinho.pop(indice)
```

```
        print(f'{item_removido}' removido do carrinho.)
```

```
    else:
```

```
        print("Índice inválido. Tente novamente.")
```

6. Como o sistema recalcula o total do pedido após a remoção de um produto?

```
# Função para calcular o total do carrinho e aplicar desconto
```

```
def finalizar_compra():
```

```
    if not carrinho:
```

```
        print("Seu carrinho está vazio. Adicione algum produto.")
```

```
    return
```

```
preco_total_carrinho = 0.00
```

```
for item in carrinho:
```

```
    nome_item, quantidade_comprada, preco_total_item = item
```

```
preco_total_carrinho += preco_total_item
for i, produtos in enumerate(armazem):
    nome, categoria, preco, quantidade_estoque, _ = produtos
    if nome == nome_item:
        armazen[i] = (nome, categoria, preco, quantidade_estoque -
quantidade_comprada, 0)

desconto = calcula_desconto(preco_total_carrinho)

if desconto > 0:
    print(f"Você teve um desconto de R${desconto:.2f}")

print(f"O valor total da sua compra foi de R${preco_total_carrinho -
desconto:.2f}")
print("Compra finalizada com sucesso! Estoque atualizado.")
carrinho.clear()
```