

1. TÍTULO:

Desenvolvendo um Sistema de Controle de Estoque com Python: Soluções para Gerenciamento em uma Loja de Eletrônicos

2. CENÁRIO:

Você foi contratado como desenvolvedor(a) para criar um sistema de controle de estoque para uma loja de eletrônicos. O sistema deve ser capaz de adicionar novos produtos, atualizar produtos existentes e visualizar o estoque, além de fornecer funcionalidades adicionais

3. DESAFIO:

Na primeira etapa do projeto, você deve se concentrar em desenvolver a estrutura inicial do sistema, utilizando os conceitos de lógica computacional e controle de fluxo em Python.

Criação da base do sistema:

a) Implemente um menu de opções para o usuário, permitindo que ele selecione diferentes funcionalidades do sistema.

b) Inclua as opções de *adicionar produto*, *atualizar produto*, *excluir produto*, *visualizar estoque* e *sair do sistema*.

Adicionar produto:

a) Ao selecionar a opção de adicionar produto, o sistema deve solicitar as seguintes informações:

- Nome do produto
- Preço do produto
- Quantidade em estoque

Atualizar produto:

a) Ao selecionar a opção de atualizar produto, o sistema deve pedir o nome do produto para atualizar e solicitar as seguintes informações para atualizar

- Preço do produto
- Quantidade em estoque

Excluir produto:

a) Ao selecionar a opção de excluir produto, o sistema deve pedir o nome para excluir o produto

Visualizar estoque:

a) Ao selecionar a opção de visualizar estoque, o sistema deverá mostrar a lista de produtos, com as seguintes informações:

- Nome do produto
- Preço do produto
- Quantidade em estoque

Você deverá subir no AVA:

- arquivo .py contendo toda a programação desenvolvida

4. TEORIA NA PRÁTICA:

Na programação, o verdadeiro aprendizado acontece quando colocamos a teoria em prática. Ao desenvolver um sistema, como um sistema de controle de estoque para uma loja de eletrônicos, você terá a chance de aplicar os conceitos fundamentais de lógica computacional, controle de fluxo e estrutura de dados, transformando conhecimento teórico em soluções práticas e funcionais.

Por exemplo, ao implementar a funcionalidade de adição de novos produtos ao sistema, você poderá utilizar uma estrutura condicional para verificar se o produto já existe no estoque e, caso contrário, adicionar a nova entrada. Da mesma forma, para garantir que as informações sobre o estoque sejam exibidas corretamente ao usuário, você pode usar loops como o "FOR" ou "WHILE" para percorrer a lista de produtos, apresentando seus detalhes de maneira organizada.

Além disso, é fundamental utilizar os conceitos de controle de fluxo, como estruturas condicionais e loops, para garantir a execução correta das funcionalidades do sistema. Se, por exemplo, o usuário quiser atualizar a quantidade de um produto existente, o sistema deve ser capaz de realizar essa verificação e proceder à alteração adequada. Utilizar loops pode ajudar a repetir essas verificações de maneira contínua e eficiente.

As estruturas de dados apropriadas, como listas, dicionários ou até mesmo classes de objetos, serão necessárias para armazenar e organizar as informações dos clientes, produtos e vendas. Escolher a estrutura correta ajudará a manter os dados de maneira acessível e eficiente, garantindo que o sistema funcione de acordo com as necessidades da loja.

Por fim, lembre-se de aplicar as boas práticas de programação, como organização clara do código, comentários explicativos e modularização, para tornar o sistema mais legível e fácil de manter. Além disso, desenvolver uma interface amigável para o usuário, mesmo que em formato simples, como um menu de opções, irá melhorar a experiência de quem utilizar o sistema.

Siga os princípios da lógica computacional e utilize a linguagem de programação Python para implementar o sistema de gerenciamento de vendas. Através desse projeto, você terá a oportunidade de aplicar seus conhecimentos, aprimorar suas habilidades e criar um sistema funcional e eficiente!

Boa sorte no desenvolvimento do sistema e aproveite a oportunidade para explorar ao máximo as ferramentas e técnicas aprendidas!

5. LEVANTAMENTO DE SOLUÇÕES

Antes de iniciar o desenvolvimento de qualquer sistema, é essencial realizar um **levantamento de soluções**, ou seja, identificar quais estratégias e tecnologias serão utilizadas para resolver os problemas propostos. No caso do sistema de controle de estoque para uma loja de eletrônicos, o levantamento de soluções envolve entender as necessidades do sistema e planejar como os conceitos de lógica computacional e Python serão aplicados para atender essas demandas.

Funcionalidades Necessárias O primeiro passo é listar as principais funcionalidades que o sistema deve oferecer:

- Adicionar novos produtos ao estoque.
- Atualizar informações de produtos existentes.
- Visualizar o estoque atual de forma clara e organizada.

- Registrar e controlar vendas, vinculando cada venda aos produtos vendidos e ao cliente.

Estruturas Condicionais e Controle de Fluxo Para garantir que essas funcionalidades sejam implementadas corretamente, o sistema deve contar com mecanismos de controle de fluxo que gerenciem a execução das diferentes tarefas. Estruturas condicionais como "IF" e "ELSE" são essenciais para verificar situações específicas, como garantir que o produto a ser adicionado não esteja duplicado, ou que os dados de vendas sejam válidos antes de serem registrados.

Uso de Loops e Estruturas de Dados Loops, como "FOR" ou "WHILE", desempenham um papel fundamental ao percorrer listas de produtos e vendas, permitindo que o sistema organize e exiba essas informações de forma eficiente. Estruturas de dados adequadas, como listas para armazenar produtos ou dicionários para relacionar vendas com clientes, são essenciais para garantir o bom funcionamento do sistema.

Interface do Usuário e Experiência Outro ponto importante no levantamento de soluções é a criação de uma interface amigável para o usuário. Mesmo em um sistema simples baseado em linha de comando, é crucial oferecer uma experiência clara e intuitiva. Isso pode ser feito com menus que guiem o usuário pelas opções de adicionar produtos, consultar estoque ou finalizar uma venda, por exemplo.

Boas Práticas de Programação Além disso, o levantamento de soluções deve incluir a aplicação das boas práticas de programação, como a modularização do código para facilitar a manutenção, uso de variáveis nomeadas de forma clara, e inclusão de comentários explicativos. Isso garante que o sistema seja não apenas funcional, mas também eficiente e de fácil compreensão para futuras atualizações.

Conclusão Com o levantamento de soluções bem definido, o desenvolvimento do sistema pode seguir um plano organizado, onde cada componente é pensado para atender a uma necessidade específica. Esse processo garante que o resultado final será um sistema



robusto, eficiente e funcional, capaz de gerenciar o estoque e as vendas da loja de eletrônicos de maneira eficiente.

Utilize essa etapa para planejar e estruturar as melhores abordagens e ferramentas para solucionar o problema de forma lógica e eficiente!



Roteiro do estudante

1. Leia o cenário:

Sua primeira tarefa é se tornar um detetive. Use os recursos que fornecemos, mas também não tenha medo de sair em busca de suas próprias informações adicionais. Queremos que você compreenda o tópico e explore o problema a fundo.

2. Encare o desafio:

Muita atenção para não perder o foco durante o estudo. Você precisa compreender qual é o desafio para não perder isso de vista durante todo o processo.

3. Teoria na prática:

Este é o momento de pesquisar o que já existe no mercado e ler todas as indicações que o professor fizer. Afinal, antes de pensar em resolver o desafio, é preciso reunir as ferramentas necessárias e reconhecer o que já existe no mercado de trabalho da sua profissão para lidar com esse tipo de situação. Não esqueça de trazer um exemplo concreto de uma solução já existente.

4. Levante hipóteses para resolver o problema:

Depois de reunir todas as suas informações, é hora de se tornar um(a) verdadeiro(a) analista. **Busque identificar padrões, relações, causas e efeitos.** Procure contradições ou conflitos de informação. **Seja crítico** e **questione** o que você encontrou durante sua investigação.

5. Aponte a melhor solução:

Agora é o momento de se tornar um(a) solucionador(a) de problemas. Com base no que você descobriu e analisou, dentre as hipóteses, qual você escolheu como uma solução para o problema? Chegou o momento de explicar em detalhes: por que essa é a melhor solução neste cenário? Use seus conhecimentos para argumentar.

6. Crie um plano de ação:



Depois de desenvolver sua solução, é hora de mostrar a viabilidade e a aplicabilidade dela. Explique em detalhes como o problema será resolvido, o passo a passo, em quanto tempo e quais as implicações disso. Lembre-se de que sua comunicação deve ser clara e eficaz para que todos entendam sua proposta. Nesta etapa, você terá que apresentar sua solução. Use a sua criatividade para apresentá-la da melhor maneira possível.

7. Autoavaliação:

Por fim, tornando-se seu próprio crítico. Avalie o que você fez bem e o que poderia ter feito melhor. Pense sobre como você pode aplicar o que aprendeu em situações futuras. Lembre-se, não estamos apenas procurando a solução correta, mas também queremos ver como você chegou até ela. A jornada é tão importante quanto o destino!

**Como seu trabalho será avaliado?**

Sua avaliação será baseada em uma série de critérios, com uma pontuação total de até 4,0 pontos.

Lembre-se, o objetivo aqui não é apenas obter a pontuação máxima, mas aprender durante o processo. Aproveite essa jornada!