

Trabalho de Paradigmas de Programação em Python

Objetivo

O objetivo deste trabalho é exercitar o uso de **paradigmas de programação** em Python, com foco em **Programação Estruturada** e **Orientação a Objetos (POO)**. Você irá implementar uma classe Produto e criar um menu interativo para manipular os objetos dessa classe.

Requisitos do Trabalho

1. Classe Produto

Implemente uma classe chamada Produto com as seguintes características:

- **Atributos:**
 - nome: nome do produto.
 - preco: preço do produto.
 - quantidade: quantidade disponível do produto.
- **Métodos:**
 - `__init__(self, nome, preco, quantidade)`: inicializa o produto com o nome, preço e quantidade fornecidos.
 - `exibir_informacoes(self)`: exibe as informações do produto (nome, preço e quantidade).
 - `atualizar_preco(self, novo_preco)`: atualiza o preço do produto.
 - `atualizar_quantidade(self, nova_quantidade)`: atualiza a quantidade do produto.
- **Exemplo:**

```
class Pessoas:
```

```
    def __init__(self, nome: str, idade: int) -> None:
```

```
        """Construtor da classe Pessoas.
```

```
        Args:
```

```
            nome (str): Nome da pessoa.
```

```
            idade (int): Idade da pessoa.
```

```
        Returns:
```

```
            None
```

```
        """
```

```
self.nome = nome
```

```
self.idade = idade
```

```
def exibir_informacoes(self) -> None:
```

```
    """Método para exibir informacoes da pessoa.
```

```
    Args:
```

```
        None
```

```
    Returns:
```

```
        None
```

```
    """
```

```
    print("Informacoes da pessoa:")
```

```
    print(f"Nome: {self.nome}")
```

```
    print(f"Idade: {self.idade}")
```

```
def atualizar_idade(self, nova_idade: int) -> None:
```

```
    """Método para atualizar a idade da pessoa.
```

```
    Args:
```

```
        nova_idade (int): Nova idade da pessoa.
```

```
    Returns:
```

```
        None
```

```
    """
```

```
    self.idade = nova_idade
```

```
    print(f"Idade atualizada para {self.idade}")
```

```
def calcular_idade_em_dias(self) -> None:
```

```
    """Método para calcular a idade da pessoa em dias.
```

Args:

None

Returns:

None

"""

```
print(f"Idade em dias: {self.idade * 365}")
```

2. Menu Interativo

Crie um arquivo separado que funcione como o **menu principal** para manipular os objetos da classe Produto. O menu deve oferecer as seguintes opções:

1. **Adicionar Produto:** Cria uma nova instância da classe Produto com os dados inseridos pelo usuário.
2. **Exibir Produtos:** Mostra uma lista de todos os produtos cadastrados com suas informações detalhadas.
3. **Atualizar Preço:** Permite ao usuário escolher um produto pelo nome e atualizar seu preço.
4. **Atualizar Quantidade:** Permite ao usuário escolher um produto pelo nome e atualizar sua quantidade.
5. **Sair:** Encerra o programa.

3. Estrutura do Programa

- Use **Programação Estruturada** para o menu, criando funções separadas para cada operação (adicionar produto, exibir produtos, atualizar preço, atualizar quantidade).
- Utilize **Orientação a Objetos** para manipular os produtos, garantindo a encapsulação dos dados e a reutilização de métodos da classe Produto.

4. Regras e Validações

- O programa deve tratar entradas inválidas (por exemplo, tentar atualizar um produto inexistente ou inserir valores inválidos para preço ou quantidade).
- Use laços e condicionais para criar o fluxo do menu, garantindo que o usuário possa realizar operações repetidamente até escolher a opção "Sair".

Entrega

- Envie o código-fonte da sua implementação em Python, dividindo o código entre a definição da classe Produto e o menu interativo em arquivos separados.
- O código deve estar **bem documentado**, com comentários explicando as funções e os principais trechos de código.

- Entrega no SAVA por link do github, PÚBLICO, até 01/11/2024.

Avaliação

A avaliação considerará:

- Corretude da implementação.
- Uso adequado dos paradigmas estruturado e orientado a objetos.
- Tratamento de erros e validações.
- Clareza e organização do código.

Boa sorte e mãos à obra!