

UNIP - Universidade Paulista
Curso Analise desenvolvimento de Sistema

**Osmar Teodoro dos Santos, Alan Mamani Ramos, Gabriel Paulo
Mariano dos Santos, Victor Augusto Silva Alberto**

Documentação Código New_divine

São Paulo
2025

Osmar Teodoro dos Santos - RA:R669173

Alan Mamani Ramos - RA:R652AB-8

Gabriel Paulo Mariano dos Santos - RA:R867ij8

Victor Augusto Silva Alberto - RA:H762BH1

Documentação Código New_divine

Trabalho de Matemática Estatística apresentado à
UNIP - Universidade Paulista, como requisito
para a aprovação na disciplina.

Orientador: Prof(a). Alex

São Paulo

2025

Documentação Código New_divine

Introdução

Este projeto tem como função principal auxiliar uma empresa a entender ou manejar o código solicitado feito em python possuindo as características de poder Inserir, Alterar e Deletar dados do estoque. Realizar compras e mostrar o valor final ao comprador além de Aplicar o desconto de 5% que deve ser mostrado ao comprador. O sistema precisa mostrar a quantidade de peças em estoque além de necessitar a realização de calcular o total de vendas e retorna esse valor em porcentagem. E como último requisito foi exigido que o sistema calcule quantas vendas foram realizadas no dia, na semana, no mês e no ano.

Desenvolvimento

Possíveis Bugs

- Erro de falta de documento no diretório: Ao iniciar o código certifique-se todos os arquivos baixados compactados no zip estão juntos onde deseja rodar o código exemplo o VS Code
- Erro de Rodar o código novamente com ele já em execução: Caso execute o código novamente sendo que já esteja em execução poderá aparecer erros variados conduto é necessário apenas rodar o código uma terceira vez para que ele retorne a normalidade
- Erro de Adição ao código: Ao rodar o código apertando no botão do Vs Code é recomendado que clique novamente ao terminal pois digitar qualquer coisa sem clicar no termina resultará em uma adição do que foi digitado ao código podendo gerar falhas e impedir o funcionamento

Funções Utilizada no código

- | | |
|--|---|
| 1. print()
Descrição: Exibe informações no terminal. | Descrição: Abre um arquivo para leitura ou escrita e garante que ele será fechado automaticamente após o uso. |
| 2. if, elif, else
Descrição: Estruturas condicionais que permitem executar diferentes blocos de código com base em condições. | 4. readline() e readlines()
Descrição: Métodos para ler arquivos. readline(): Lê uma única linha do arquivo. |
| 3. with open() | |

`readlines()`: Lê todas as linhas do arquivo e retorna uma lista.

5. `write()`

Descrição: Escreve dados em um arquivo.

6. `import`

Descrição: Importa módulos ou bibliotecas para usar suas funcionalidades.

7. `os.system()`

Descrição: Executa comandos do sistema operacional.

8. `time.sleep()`

Descrição: Faz o programa "pausar" por um tempo especificado (em segundos).

9. `input()`

Descrição: Recebe entrada do usuário.

10. `int()` e `float()`

Descrição: Convertem valores para inteiros (`int`) ou números de ponto flutuante (`float`).

11. `split()`

Descrição: Divide uma string em partes com base em um delimitador.

12. `strip()`

Descrição: Remove espaços em branco ou caracteres específicos do início e do fim de uma string.

13. `datetime`

Descrição: Biblioteca para manipular datas e horas.

14. `try` e `except`

Descrição: Tratam erros que podem ocorrer durante a execução do código.

15. `for`

Descrição: Estrutura de repetição que itera sobre elementos de uma sequência (como listas ou strings).

16. `range()`

Descrição: Gera uma sequência de números.

17. `append()`

Descrição: Adiciona um elemento ao final de uma lista.

18. `len()`

Descrição: Retorna o número de elementos em uma sequência (como listas ou strings).

19. `sum()`

Descrição: Retorna a soma de todos os elementos de uma lista ou sequência numérica.

20. `float()`

Descrição: Converte um valor para um número de ponto flutuante.

Explicação Código

A explicação neste momento atual da documentação será feito a partir de blocos pré-estruturados por meios dos comentários dentro do próprio código nomeado por meios de números que serão utilizados aqui e possui a finalidade de auxiliar na compreensão, A ordem dos blocos no código pode não ser explicitamente apresentada durante este documento já que o intuito é torná-lo intuitivo e completo para quem o utilizar.

Bloco 1: o bloco se localiza os import e a função tempo existente no código explicados nos blocos posteriores

Bloco 2: O código do bloco 2 tem como objetivo criar um menu introdutório para quem está rodando o código e oferece duas opções para iniciar o processo , pós-escolhas ele mostra uma tela de verificando e segue para o bloco 3

```

# Bloco 2 -----
os.system("cls")
print(p)
print('Seja muito bem vindo ao serviços de auto peça')
print('qual seu objetivo por hj ?')
print(p)
time.sleep(4)
os.system("cls")
print(p)
escolhas = int(input(
    "Escolha uma das opções abaixo:\n"
    "1 - Abrir a parte de funcionários\n"
    "2 - Realizar uma compra de peças\n"
    "Digite sua escolha: "
))
print(p)
time.sleep(1)
os.system('cls')
print(p)
print('Verificando .....')
print(p)
time.sleep(1)
os.system("cls")
# Bloco 2 -----
```

anexo 4 - Bloco 2 do codigo new_divine (autores)

Bloco 3: O bloco 3 possui duas divisões devido a um if e um else, por isso sua explicação é dividida em sub blocos que serão explicados posteriormente com o decorrer do trabalho mas em principio o codigo executando o if ele segue para setor de compras onde mostrar inicialmente todos os dados armazenados nos arquivos (.txt) de nomes Banco_de_Dados, EstMax e Estoques, e depois permite as funcionalidades por medios de if, elif, else que serão explicadas nos blocos futuros.

```

if escolhas == 1:
    print(p)
    print('Seja muito bem vindo a parte de funcionários')
    print(p)
    time.sleep(2)
    with open('Banco_de_Dados.txt', 'r') as arquivo:
        conteudo = arquivo.readlines()

    print(p)
    print('Dados do arquivo:')
    for linha in conteudo:
        print(linha.strip())
    print(p)

    with open('EstMax.txt', 'r') as arquivo:
        estvol = int(arquivo.readline().strip())

    with open('Estoques.txt', 'r') as arquivo:
        estoque_atual = int(arquivo.readline().strip())

    CE= estoque_atual

    estvoll = (estvol - CE)

    print(p)
    print(f'Estoque Atuais:{estvoll}')
    print(p)

    print('O que deseja fazer ?')
    print(p)
    time.sleep(2)
    escolhas2 = int(input(
        "Escolha uma das opções abaixo:\n"
        "1 - alterar dados 1\n"
        "2 - inserir dados 2 \n"
        "3 - deletar dados 3\n"
        "4 - mostrar vendas do total no estoque 4\n"
        "5 - relatório de vendas dia-semana-mes-ano 5\n"
        "Digite sua escolha: "
    ))

```

anexo 5 - Bloco de parte dos funcionários do código new_divine (autores)

A parte dedicada ao else se mostra um setor de compras onde a pessoa coloca quantas pessoas deseja comprar e recebe as informações de volta da compra e os descontos aplicados. Os dados da compra são salvos nos arquivos (.txt) correspondentes para a utilização desses dados posteriormente

```

else:
    print(p)
    print('Seja muito bem vindo a parte de compras')
    print(p)
    time.sleep(2)
    os.system('cls')
    print(p)
    print('Quantas peças deseja comprar ?')
    print(p)
    compra = int(input('Digite a quantidade de peças que deseja comprar: '))
    print(p)
    time.sleep(2)

    import Givining_UP
    Givining_UP.compras += compra

    print(p)
    time.sleep(2)
    os.system('cls')
    print(p)
    print('Calculando .....')
    print(p)
    time.sleep(2)
    os.system('cls')

    calculos = Givining_UP.compras * precos
    desconto = calculos * 0.05
    valordescontos = calculos - desconto

    print(p)
    print(f'A sua compra foi de {calculos} reais')
    print(p)
    time.sleep(1)
    print(p)
    print(f'Porém com o desconto de primeira compra de 5% fica no valor de {valordescontos} reais')
    print(p)

    with open('Estoques.txt', 'r') as arquivo:
        estoque_atual = int(arquivo.readline().strip())

    CE= estoque_atual + compra

```

anexo 6 - Bloco 3 da parte de compras do código new_divine (autores)

```

with open('Estoques.txt', 'w') as arquivo:
    arquivo.write(f'{CE}')

data_atual = datetime.now().strftime('%d/%m/%Y %H:%M:%S')
with open('Banco_de_Dados.txt', 'a') as arquivo:
    arquivo.write(f'Compra: {Givining_UP.compras} peças, Total: {calculos} reais, Data: {data_atual}\n')

print('Os dados da compra foram salvos no arquivo "Banco_de_Dados.txt".')
print(p)
time.sleep(4)
os.system('cls')

```

anexo 7 - Bloco 3 parte de compras (continuação do anexo 6)(autores)

Bloco 3.1: Este bloco serve para alteração de valor no estoque, caso venha der a necessidade de reduzir o estoque máximo.

```

# Bloco 3.1 -----
if escolhas2 == 1:
    time.sleep(2)
    os.system('cls')
    print(p)
    print('Alterando dados .....')
    print(p)
    time.sleep(2)
    os.system('cls')
    with open('EstMax.txt', 'w') as arquivo:
        arquivo.write('')

    alter = int(input('Digite a quantidade de peças que deseja que tenha no estoque maximo agr: '))

    with open('EstMax.txt', 'w') as arquivo:
        arquivo.write(f'{alter}')

    time.sleep(2)
    os.system('cls')
    print(p)
    print('Alterando dados .....')
    print(p)

    print(p)
    print('Dados alterados com sucesso')
    print(p)
    time.sleep(2)
    os.system('cls')

```

anexo 8 - Bloco 3.1 alteração de Dados(autores)

Bloco 3.2: Esta parte possui como função principal executar o código para inserir valores no estoque máximo caso deseje aumentar o valor do estoque, este trecho serve unicamente para aumentar o estoque.

```

# Bloco 3.2 -----
elif escolhas2 == 2:

    with open('EstMax.txt', 'r') as arquivo:
        estins = int(arquivo.readline().strip())

    print(p)
    ins= int(input('Digite a quantidade de peças que deseja inserir no estoque: '))
    print(p)
    time.sleep(2)
    os.system('cls')

    with open('EstMax.txt', 'w') as arquivo:
        arquivo.write(f'{estins + ins}')

    print(p)
    print('Inserindo dados .....')
    print(p)
    time.sleep(2)
    os.system('cls')

    print(p)
    print('Dados inseridos com sucesso')
    print(p)

    time.sleep(2)
    os.system('cls')

```

anexo 9 - Bloco 3.2 inserindo Dados(autores)

Bloco 3.3:Esta função apenas mostra o total de vendas em porcentagem a partir do total do estoque, caso o estoque sofra alteração essa etapa também sofrerá alteração

```

# Bloco 3.3 -----
elif escolhas2 == 4:

    print(p)
    time.sleep(2)
    with open('EstMax.txt', 'r') as arquivo:
        estomax = int(arquivo.readline().strip())

    with open('Estoques.txt', 'r') as arquivo:
        estoque_atual = int(arquivo.readline().strip())

    Por = (estoque_atual /estomax ) * 100

    print(p)
    print(f'Vendas a partir do estoque total:{Por} %')
    print(p)
    time.sleep(5)
    os.system('cls')

```

anexo 8 - Bloco 3.3 Porcentagem de vendas(autores)

Bloco 3.4: A etapa a seguir serve para mostrar as compras realizadas durante os período proposto dentro das opções de escolhas


```
# Bloco 3.4 -----
elif escolhas2 == 5:
    print(p)
    print("Escolha o período para o relatório:")
    print("1. Dia")
    print("2. Semana")
    print("3. Mês")
    print("4. Ano")
    periodo_opcao = input("Digite sua escolha: ")
    print(p)
    time.sleep(3)
    os.system('cls')

    if periodo_opcao == "1":
        relatorio_compras("dia")
    elif periodo_opcao == "2":
        relatorio_compras("semana")
    elif periodo_opcao == "3":
        relatorio_compras("mes")
    elif periodo_opcao == "4":
        relatorio_compras("ano")
    else:
        print("Opção inválida.")
```

anexo 8 - Bloco 3.4 Mostrar compras realizadas em um período determinado(autores)

Bloco 3.5: Nesse trecho é deletado os valores das compras e do estoque onde foi armazenada a informação de compras retomando tudo ao estado inicial.

```
# Bloco 3.5 -----
else:
    print(p)
    print('Deletando dados .....')
    print(p)

    with open('Banco_de_Dados.txt', 'w') as arquivo:
        arquivo.write('')

    with open('Estoques.txt', 'w') as arquivo:
        arquivo.write('0')

    time.sleep(2)
    os.system('cls')
    print(p)
    print('Dados deletados com sucesso')
    print(p)
    time.sleep(2)
    os.system('cls')
```

anexo 8 - Bloco 3.5 Deleta as compras realizadas (autores)

Conclusão

O código em execução em execução inicia com uma introdução e depois se segue para o primeiro parâmetro de escolhas

```
=====
Seja muito bem vindo ao serviços de auto peça
qual seu objetivo por hj ?
=====
█
```

anexo 9 - Introdução (autores)

```
=====
Escolha uma das opções abaixo:
1 - Abrir a parte de funcionários
2 - Realizar uma compra de peças
Digite sua escolha: 
```

anexo 10 - parâmetro de escolhas 1 (autores)

Na segunda etapa a partir da escolha de ir pela primeira opção o código irá carregar e mostrar mais informações referente ao que deseja gerenciar no setor de estoques e informações adicionadas para auxiliar no gerenciamento

```
=====
Seja muito bem vindo a parte de funcionários
=====
Dados do arquivo:
Compra: 3 peças, Total: 30 reais, Data: 23/03/2025 17:12:48
Compra: 4 peças, Total: 40 reais, Data: 23/03/2025 17:13:17
Compra: 5 peças, Total: 50 reais, Data: 23/03/2025 18:08:49
Compra: 10 peças, Total: 100 reais, Data: 24/03/2025 18:54:08
Compra: 25 peças, Total: 250 reais, Data: 24/03/2025 19:23:23
Compra: 3 peças, Total: 30 reais, Data: 25/03/2025 13:00:32
Compra: 4 peças, Total: 40 reais, Data: 02/04/2025 12:38:05
Compra: 3 peças, Total: 30 reais, Data: 02/04/2025 12:42:11
=====
Estoque Atuais:343
=====
O que deseja fazer ?
=====
Escolha uma das opções abaixo:
1 - alterar dados 1
2 - inserir dados 2
3 - deletar dados 3
4 - mostrar vendas do total no estoque 4
5 - relatório de vendas dia-semana-mes-ano 5
Digite sua escolha: 
```

anexo 11 - setor de funcionário (autores)

Entre a escolhas realizadas as opções alterar, inserir e relatório de vendas irão pedir que coloque valores e por isso vai retornar aonde os dados foram gravados ou uma mensagem de sucesso.

Seguindo agora pelo caminho de opção 2 o código irá abrir o setor de compras e pedir a quantidade de peças que irá comprar

```
=====
Quantas peças deseja comprar ?
=====
Digite a quantidade de peças que deseja comprar: 
```

anexo 12 - setor de compras (autores)

Após essa etapa irá mostrar o valor da compra e o desconto e gravar os resultados nos arquivos (.txt)

```
=====
a sua compra foi de 80 reais
=====
Porem com o desconto de primeira compra de 5% fica no valor de 76.0 reais
=====
Os dados da compra foram salvos no arquivo "Banco_de_Dados.txt".
=====
█
```

anexo 13 - resultado da compra - (autores)

Referências

A Biblioteca Padrão do Python. Disponível em:
<<https://docs.python.org/pt-br/3.13/library/index.html>>.

W3 SCHOOLS. Python Functions. Disponível em:
<https://www.w3schools.com/python/python_functions.asp>.