

Missão prática - Nível 3 – Mundo 1

“ Estruturando os dados (Missão) ”

//Disciplina: Nivel3: Estruturando dados

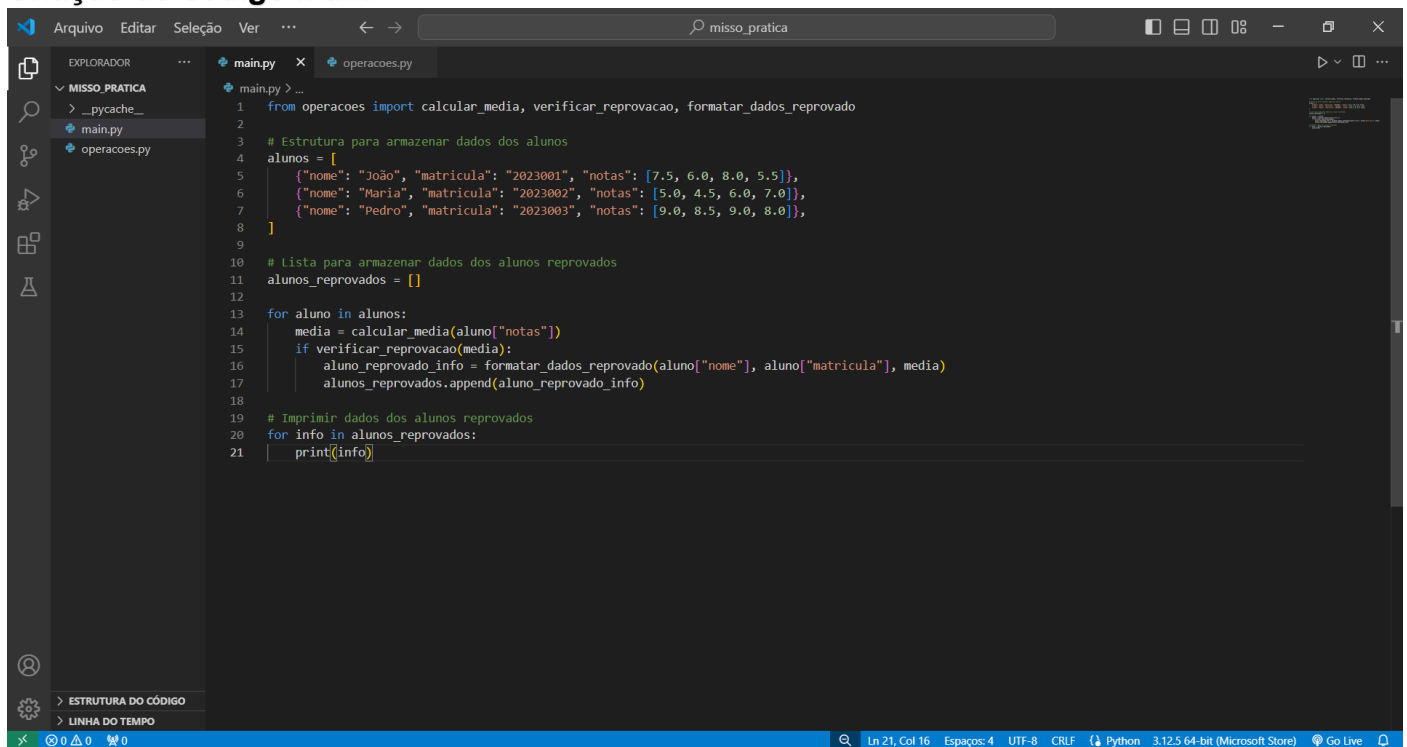
//Professor: Raul Carlos Costa Queiros

//Aluno: Rafael Lima de Medeiros

//Turma: 2024.2

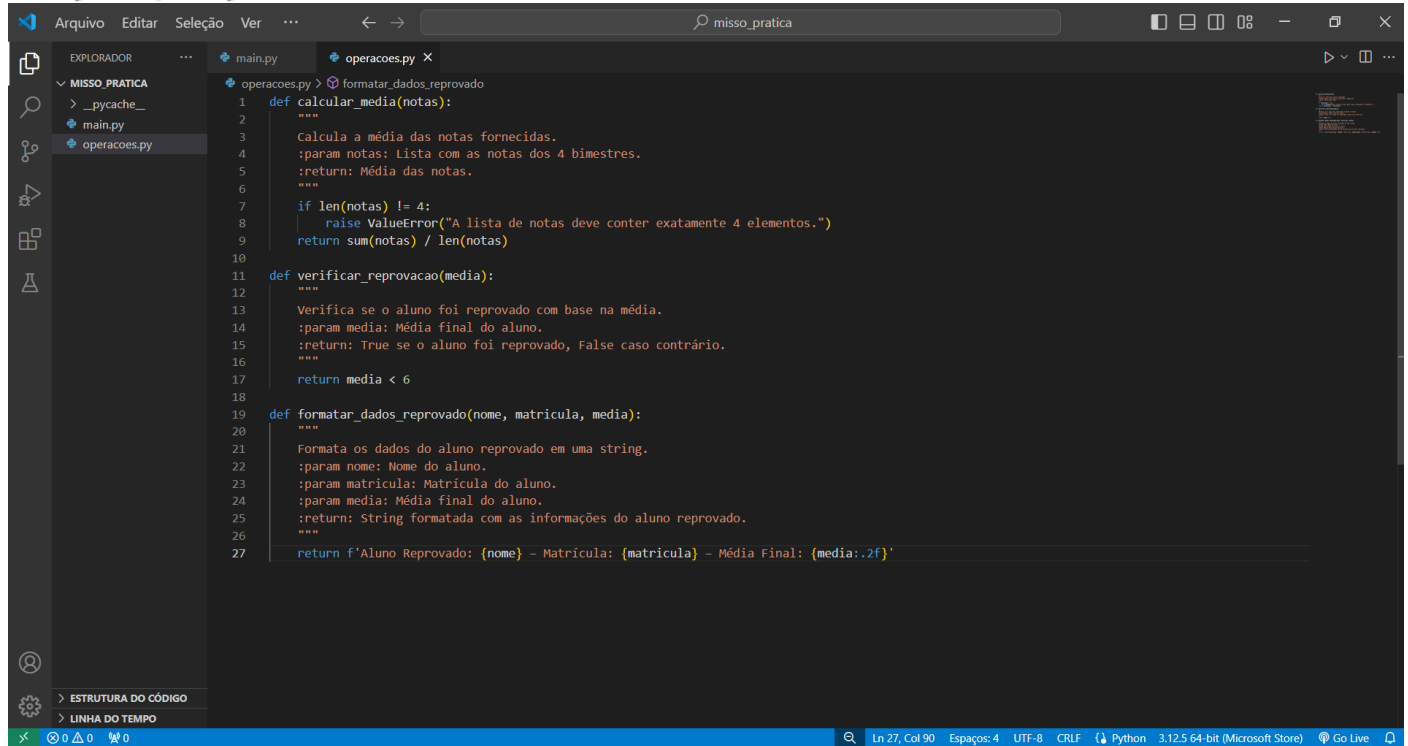
//Data atual: 13/08/2024

Criação do código main:



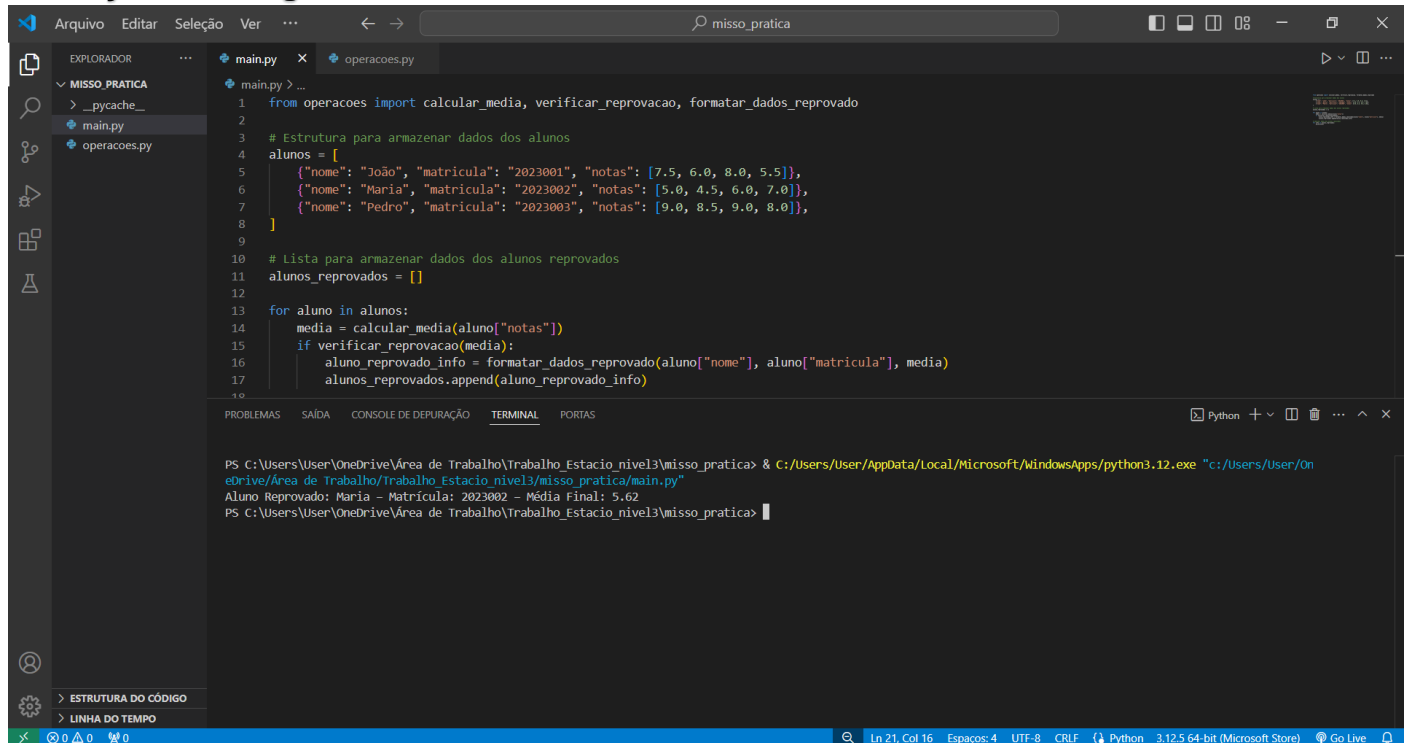
```
1 from operacoes import calcular_media, verificar_reprovacao, formatar_dados_reprovado
2
3 # Estrutura para armazenar dados dos alunos
4 alunos = [
5     {"nome": "João", "matricula": "2023001", "notas": [7.5, 6.0, 8.0, 5.5]},
6     {"nome": "Maria", "matricula": "2023002", "notas": [5.0, 4.5, 6.0, 7.0]},
7     {"nome": "Pedro", "matricula": "2023003", "notas": [9.0, 8.5, 9.0, 8.0]},
8 ]
9
10 # Lista para armazenar dados dos alunos reprovados
11 alunos_reprovados = []
12
13 for aluno in alunos:
14     media = calcular_media(aluno["notas"])
15     if verificar_reprovacao(media):
16         aluno_reprovado_info = formatar_dados_reprovado(aluno["nome"], aluno["matricula"], media)
17         alunos_reprovados.append(aluno_reprovado_info)
18
19 # Imprimir dados dos alunos reprovados
20 for info in alunos_reprovados:
21     print(info)
```

Criação operações:



```
1 def calcular_media(notas):
2     """
3     Calcula a média das notas fornecidas.
4     :param notas: Lista com as notas dos 4 bimestres.
5     :return: Média das notas.
6     """
7     if len(notas) != 4:
8         raise ValueError("A lista de notas deve conter exatamente 4 elementos.")
9     return sum(notas) / len(notas)
10
11 def verificar_reprovacao(media):
12     """
13     Verifica se o aluno foi reprovado com base na média.
14     :param media: Média final do aluno.
15     :return: True se o aluno foi reprovado, False caso contrário.
16     """
17     return media < 6
18
19 def formatar_dados_reprovado(nome, matricula, media):
20     """
21     Formata os dados do aluno reprovado em uma string.
22     :param nome: Nome do aluno.
23     :param matricula: Matricula do aluno.
24     :param media: Média final do aluno.
25     :return: String formatada com as informações do aluno reprovado.
26     """
27     return f'Aluno Reprovado: {nome} - Matricula: {matricula} - Média Final: {media:.2f}'
```

Execução do código:



```
1 from operacoes import calcular_media, verificar_reprovacao, formatar_dados_reprovado
2
3 # Estrutura para armazenar dados dos alunos
4 alunos = [
5     {"nome": "João", "matricula": "2023001", "notas": [7.5, 6.0, 8.0, 5.5]},
6     {"nome": "Maria", "matricula": "2023002", "notas": [5.0, 4.5, 6.0, 7.0]},
7     {"nome": "Pedro", "matricula": "2023003", "notas": [9.0, 8.5, 9.0, 8.0]},
8 ]
9
10 # Lista para armazenar dados dos alunos reprovados
11 alunos_reprovados = []
12
13 for aluno in alunos:
14     media = calcular_media(aluno["notas"])
15     if verificar_reprovacao(media):
16         aluno_reprovado_info = formatar_dados_reprovado(aluno["nome"], aluno["matricula"], media)
17         alunos_reprovados.append(aluno_reprovado_info)
```

PROBLEMAS SAÍDA CONSOLE DE DEPURACÃO **TERMINAL** PORTAS

```
PS C:\Users\User\OneDrive\Área de Trabalho\Trabalho_Estacio_nivel3\misso_pratica> & C:\Users\User\AppData\Local\Microsoft\WindowsApps\python3.12.exe "c:/Users/User/OneDrive/Área de Trabalho/Trabalho_Estacio_nivel3/misso_pratica/main.py"
Aluno Reprovado: Maria - Matricula: 2023002 - Média Final: 5.62
PS C:\Users\User\OneDrive\Área de Trabalho\Trabalho_Estacio_nivel3\misso_pratica>
```

Código MAIN:

```
from operacoes import calcular_media, verificar_reprovacao, formatar_dados_reprovado

# Estrutura para armazenar dados dos alunos
alunos = [
    {"nome": "João", "matricula": "2023001", "notas": [7.5, 6.0, 8.0, 5.5]},
    {"nome": "Maria", "matricula": "2023002", "notas": [5.0, 4.5, 6.0, 7.0]},
    {"nome": "Pedro", "matricula": "2023003", "notas": [9.0, 8.5, 9.0, 8.0]},
]

# Lista para armazenar dados dos alunos reprovados
alunos_reprovados = []

for aluno in alunos:
    media = calcular_media(aluno["notas"])
    if verificar_reprovacao(media):
        aluno_reprovado_info = formatar_dados_reprovado(aluno["nome"], aluno["matricula"], media)
        alunos_reprovados.append(aluno_reprovado_info)

# Imprimir dados dos alunos reprovados
for info in alunos_reprovados:
    print(info)
```

Código OPERAÇÕES:

```
def calcular_media(notas):
    """
    Calcula a média das notas fornecidas.
    :param notas: Lista com as notas dos 4 bimestres.
    :return: Média das notas.
    """
    if len(notas) != 4:
        raise ValueError("A lista de notas deve conter exatamente 4 elementos.")
    return sum(notas) / len(notas)

def verificar_reprovacao(media):
    """
    Verifica se o aluno foi reprovado com base na média.
    :param media: Média final do aluno.
    :return: True se o aluno foi reprovado, False caso contrário.
    """
    return media < 6

def formatar_dados_reprovado(nome, matricula, media):
    """
    Formata os dados do aluno reprovado em uma string.
    :param nome: Nome do aluno.
    :param matricula: Matrícula do aluno.
    :param media: Média final do aluno.
    :return: String formatada com as informações do aluno reprovado.
    """
    return f'Aluno Reprovado: {nome} – Matrícula: {matricula} – Média Final: {media:.2f}'
```