

Nome: João Geiger Piza

Repositório no Github: <https://github.com/joaogpizza/Lipai>

Códigos das Videoaulas:

aula01.py:

```
#open("caminho", "r")

# Mode
# r - leitura
# a - Append / incrementar
# w - Escrita
# x - Criar Arquivo
# rw - Leitura + Escrita

# arquivo = open("src/06-arquivos/test3.txt", "x")

# # print(arquivo.readable())
# # print(arquivo.read())
# # print(arquivo.readline())
# # print(arquivo.readline())
# # print(arquivo.readline())
# # print(arquivo.readline())

# lista = arquivo.readlines()

# print(lista)

# print(lista[3])

# arquivo.write("Python\n")
# arquivo.write("C++\n")
# arquivo.write("Terraform\n")


# arquivo.close()

import os

# if os.path.exists("src/06-arquivos/test2.txt"):
#     os.remove("src/06-arquivos/test2.txt")
# else:
#     print("Arquivo não existe")

os.rmdir("src/06-arquivos/nova_pasta")
```

```
aula02.py:
```

```
with open('src/06-arquivos/arquivo.txt', 'r') as arq:  
    print(next(arq))  
    print(next(arq))
```

Exercícios:

```
ex01.py:
```

```
""" Exercicio 01 S3.A2 """  
  
def carregar_dados_alunos(nome_arquivo):  
    """ Retorna uma tupla de dicionários com dados de alunos. """  
    resultado = ()  
  
    with open(nome_arquivo, "r", encoding="utf8") as arq:  
        linhas = arq.readlines()  
        dicionarios = []  
        for linha in linhas:  
            linha.strip('\n')  
            infos = linha.split(',')  
            dionario = {  
                'prontuario': infos[0],  
                'nome': infos[1],  
                'email': infos[2]  
            }  
            dicionarios.append(dionario)  
  
    resultado = tuple(dicionarios)  
    return resultado  
  
  
tupla = carregar_dados_alunos("src/06-arquivos/exercicios/txtex01.txt")  
  
for aluno in tupla:  
    print(f'Prontuário: {aluno['prontuario']} \n' +  
          f'Nome: {aluno['nome']} \n' +  
          f'Email: {aluno['email']} \n' +  
          '-----\n')
```

```
ex02.py:
```

```
""" Ex 02 S3.A2 """
```

```

def carregar_dados_projetos(nome_arquivo):
    """ Retorna uma tupla de dicionários com dados de projetos. """
    dicionarios = []

    with open(nome_arquivo, 'r', encoding='utf8') as arq:
        linhas = arq.readlines()
        for linha in linhas:
            linha.strip('\n')
            infos = linha.split(',')
            dionario = {
                'codigo': int(infos[0]),
                'titulo': infos[1],
                'responsavel': infos[2]
            }
            dicionarios.append(dionario)

    resultado = tuple(dicionarios)
    return resultado

```

```

tupla =
carregar_dados_projetos("src/06-arquivos/exercicios/txtex02.txt")

for projeto in tupla:
    print(f'Código: {projeto['codigo']}\n' +
          f'Título: {projeto['titulo']}\n' +
          f'Responsável: {projeto['responsavel']} \n' +
          '-----\n')

```

ex03.py:

```

""" Exercício 03 S3.A2"""

def linha_para_dict(linha, chaves):
    """ Recebe uma linha e uma lista de chaves e retorna um dicionário.
    """

    dionario = {}

    linha.strip('\n')
    infos = linha.split(',')
    if len(infos) != len(chaves):
        print('ERRO: menos informações do que necessário')

```

```
        return dicionario
    for i in range(len(infos)):
        dicionario[chaves[i]] = infos[i]
    return dicionario

print(linha_para_dict('SP00001,Maria da Silva,maria@email.com',
['prontuario','nome','email']))
print(linha_para_dict('banana,3', ['item', 'quantidade']))
```

Reflexão sobre arquivos:

- 1) Permite seccionar melhor as partes da string, tratando cada uma por uma chave identificadora.
- 2) Quando se quer um acesso mais fácil às informações do arquivo, ao invés de ter que caçar string por string pela informação.
- 3) Ler um arquivo em python é tornado trivial pela função open, enquanto transformar as linhas em estruturas de dados requer maior esforço mental (apesar de mesmo assim ser bem tranquilo).