

# FILE JEDI

**DOMINE A FORÇA DO PYTHON**



Aprenda a usar python para trabalhar com arquivos

**MURILO OLIVEIRA**

# Trabalhando com Arquivos em Python

## Introdução

Neste ebook, vamos explorar como utilizar Python para trabalhar com arquivos, com foco especial em arquivos de texto (txt) e arquivos de valores separados por vírgulas (csv). Também abordaremos o uso de data frames com a biblioteca Pandas. Vamos manter tudo simples e direto ao ponto, sempre com exemplos de código para facilitar a compreensão.



01

# Manipulando TXT

---

# Abrindo e Lendo Arquivos de Texto



Para ler um arquivo de texto em Python, usamos a função `open` combinada com métodos de leitura como `read`, `readline`, ou `readlines`.

```
# Abrindo e lendo um arquivo inteiro
with open('arquivo.txt', 'r') as file:
    conteudo = file.read()
    print(conteudo)

# Lendo linha por linha
with open('arquivo.txt', 'r') as file:
    for linha in file:
        # strip remove espaços extras e quebras de linha
        print(linha.strip())
```

snappify.com



# Escrevendo em Arquivos de Texto



Para escrever em arquivos de texto, utilizamos os modos w (escrita) ou a (anexar).

```
# Escrevendo um novo conteúdo (sobrescreve o arquivo se já existir)
with open('arquivo.txt', 'w') as file:
    file.write('Olá, Mundo!\n')

# Anexando conteúdo ao arquivo existente
with open('arquivo.txt', 'a') as file:
    file.write('Adicionando uma nova linha.\n')
```

snappify.com



02

# **Manipulando Arquivos CSV**

---

# Lendo Arquivos CSV



Para ler arquivos CSV, podemos usar a biblioteca padrão csv.

```
import csv

# Lendo um arquivo CSV
with open('dados.csv', 'r') as file:
    leitor_csv = csv.reader(file)
    for linha in leitor_csv:
        print(linha)
```

snappify.com



# Escrevendo em Arquivos CSV



Para escrever em arquivos CSV, utilizamos o método writer.

```
import csv

# Escrevendo em um arquivo CSV
with open('dados.csv', 'w', newline='') as file:
    escritor_csv = csv.writer(file)
    escritor_csv.writerow(['Nome', 'Idade', 'Cidade'])
    escritor_csv.writerow(['João', '25', 'São Paulo'])
    escritor_csv.writerow(['Maria', '30', 'Rio de Janeiro'])
```

snappify.com





# 03

## Data Frames no Pandas

---

Os seletores de descendente e filho permitem que você selecione elementos com base em sua hierarquia dentro do HTML. Eles são úteis para estilizar elementos específicos dentro de um contexto mais amplo. Confira os exemplos a seguir:

# Introdução ao Pandas



A biblioteca Pandas é extremamente útil para manipulação de dados, especialmente quando estamos lidando com grandes volumes de informações organizadas em tabelas.

```
import pandas as pd
```



snappify.com

# Lendo Arquivos CSV com Pandas



Usar Pandas para ler arquivos CSV é muito simples com o método `read_csv`.

```
# Lendo um arquivo CSV
df = pd.read_csv('dados.csv')
print(df)
```



snappify.com

# Escrevendo Data Frames em Arquivos CSV



Para salvar data frames em arquivos CSV, utilizamos o método `to_csv`.

```
# Salvando um data frame em um arquivo CSV  
df.to_csv('dados_salvos.csv', index=False)
```

snappify.com



# AGRADECIMENTOS

---

# OBRIGADO POR LER ATÉ AQUI



Esse Ebook foi gerado por IA, e diagramado por humano.  
O passo a passo se encontra no meu Github

•

Esse conteúdo foi gerado com fins didáticos de construção,  
não foi realizado uma validação cuidadosa humana no  
conteúdo e pode conter erros gerados por uma IA.



<https://github.com/Murilo47/prompts-recipe-to-create-a-ebook/tree/main>

Baseado no trabalho de:

<https://github.com/felipeAguiarCode/prompts-recipe-to-create-a-ebook>

