



Integrar Python com arquivos de texto (.txt) é uma tarefa comum na programação. Aqui estão algumas coisas que você deve saber ao trabalhar com integração entre Python e arquivos de texto:



#### 1. Abrindo e Fechando Arquivos:

- Para abrir um arquivo, use a função `open()`, que retorna um objeto do tipo arquivo.
- Certifique-se de fechar o arquivo usando o método `close()` quando você terminar de usá-lo.

#### 2. Modos de Abertura:

- Ao abrir um arquivo, você especifica um modo (leitura, escrita, anexação, etc.).
- Os modos comuns incluem `'r'` (leitura), `'w'` (escrita), `'a'` (anexação) e `'b'` (modo binário).

#### 3. Leitura de Arquivos:

- Use o método `read()` para ler todo o conteúdo do arquivo de uma vez, ou `readline()` para ler linha por linha.
- O método `readlines()` lê todas as linhas em uma lista.

#### 4. Escrita em Arquivos:

- Use o método `write()` para escrever strings em um arquivo.
- Lembre-se de adicionar caracteres de quebra de linha `'\n'` quando desejar criar novas linhas.

#### 5. Anexação a Arquivos:

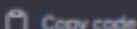
- O modo `'a'` permite que você adicione conteúdo ao final de um arquivo existente.

#### 6. Gerenciamento Contextual de Arquivos:

- Use a declaração `with` (context manager) para abrir arquivos. Isso garantirá que o arquivo seja fechado automaticamente quando você sair do bloco `with`.

Exemplo de leitura de um arquivo:

python

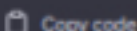


Copy code

```
with open('arquivo.txt', 'r') as arquivo:
    conteudo = arquivo.read()
    print(conteudo)
```

Exemplo de escrita em um arquivo:

python



Copy code

```
with open('saida.txt', 'w') as arquivo:
    arquivo.write('Olá, mundo!\nNova linha.')
```