



escola
britânica de
artes criativas
& tecnologia

Módulo | Python: Arquivos & Funções

Caderno de **Exercícios**

Professor [André Perez](#)

▼ Tópicos

1. Leitura;
 2. Escrita;
 3. Funções;
 4. Escopo.
-

▼ Exercícios

▼ 0. Preparação do ambiente

Neste exercício vamos trabalhar com os arquivos de csv e texto definidos abaixo. Execute cada uma das células de código para escrever os arquivos na sua máquina virtual.

- **carros.csv**: arquivo csv com informações sobre carros (venda, manutenção, portas, etc.).

```
%%writefile carros.csv
id,valor_venda,valor_manutencao,portas,pessoas,porta_malas
1,vhigh,med,2,2,small
2,med,vhigh,2,2,small
3,low,vlow,4,1,small
```

```
5,low,high,2,2,small
6,low,high,4,4,big
7,low,high,4,4,big
8,low,med,2,2,small
9,low,med,2,2,small
10,low,med,2,2,small
11,low,med,4,4,big
12,low,low,2,2,small
13,low,low,4,4,small
14,low,low,4,4,med
```

Overwriting carros.csv

- **musica.txt**: arquivo texto com a letra da música Roda Viva do Chico Buarque.

```
%%writefile musica.txt
```

```
Roda Viva
Chico Buarque
```

```
Tem dias que a gente se sente
Como quem partiu ou morreu
A gente estancou de repente
Ou foi o mundo então que cresceu
A gente quer ter voz ativa
No nosso destino mandar
Mas eis que chega a roda viva
E carrega o destino pra lá
```

```
Roda mundo, roda-gigante
Roda moinho, roda pião
```

```
O tempo rodou num instante
Nas voltas do meu coração
A gente vai contra a corrente
Até não poder resistir
Na volta do barco é que sente
O quanto deixou de cumprir
Faz tempo que a gente cultiva
A mais linda roseira que há
Mas eis que chega a roda viva
E carrega a roseira pra lá
```

```
Roda mundo, roda-gigante
Roda moinho, roda pião
```

Overwriting musica.txt

1. Extração de coluna de arquivo csv

1.1. Extraia os valores `valor_venda` e armazena em uma lista.

```
valor_venda = []

with open(file='./carros.csv', mode='r', encoding='utf8') as arquivo:
    # lê o cabeçalho
    # lê a primeira linha
    while linha:
        # quebra a string nas virgulas e salva os resultados em uma lista
        # seleciona o segundo elemento da lista
        # salva o valor na lista de valor_venda
        # lê uma nova linha, se a linha não existir, salva o valor None

print(valor_venda)
```

1.2 Complete a função abaixo para extrair uma coluna, do arquivo csv em uma lista.

```
def extrai_coluna_csv(nome_arquivo: str, indice_coluna: int):

    coluna = []

    # leia o arquivo com o comando 'with' utilizando o parametro 'nome_arquivo'
    # extraia a coluna do arquivo utilizando o parametro 'indice_coluna'

    return coluna
```

Você pode testar a função com os códigos abaixo.

```
# extrair a coluna valor_manutencao
valor_manutencao = extrai_coluna_csv(nome_arquivo='./carros.csv', indice_coluna=2)
print(valor_manutencao) # deve retornar ['med', 'vhigh', 'vhigh', ...]

# extrair a coluna porta_malas
porta_malas = extrai_coluna_csv(nome_arquivo='./carros.csv', indice_coluna=5)
print(porta_malas) # deve retornar ['small', 'small', 'small', ...]
```

Exercício bônus

1. Funções para arquivo csv

Complete a função abaixo para extrair uma coluna do arquivo csv em uma lista. Os elementos devem ter o tipo de dado correto.

```
def extrai_coluna_csv(nome_arquivo: str, indice_coluna: int, tipo_dado: str):

    coluna = []

    # leia o arquivo com o comando 'with' utilizando o parametro 'nome_arquivo'
    # extraia a coluna do arquivo utilizando o parametro 'indice_coluna'
    # use a estrutura de decisão if/elif/else para fazer a conversão do tipo de dados uti

    return coluna
```

Você pode testar a função com os códigos abaixo.

```
# extrair a coluna valor_venda
valor_venda = extrai_coluna_csv(nome_arquivo='./carros.csv', indice_coluna=1, tipo_dado='str')
print(valor_venda) # deve retornar ['vhigh', 'med', 'low', ...]

# extrair a coluna pessoas
pessoas = extrai_coluna_csv(nome_arquivo='./carros.csv', indice_coluna=4, tipo_dado='int')
print(pessoas) # deve retornar [2, 2, 2, ...]
```

2. Funções para arquivo txt

Complete a função abaixo para extrair uma as palavras de uma linha do arquivo txt em uma lista.

```
def extrai_linha_txt(nome_arquivo: str, numero_linha: int):

    palavras_linha = []

    # leia o arquivo com o comando 'with' utilizando o parametro 'nome_arquivo'
    # extraia a linha do arquivo utilizando o parametro 'numero_linha'
    # quebre a linha em palavras com o comando split, note que o separador é um espaço ' '

    return palavras_linha
```

Você pode testar a função com os códigos abaixo.

```
linha10 = extrai_linha_txt(nome_arquivo='./musica.txt', numero_linha=10)
print(linha10) # deve retornar ['Mas', 'eis', 'aue', 'chega', 'a', 'roda', 'viva']
```

