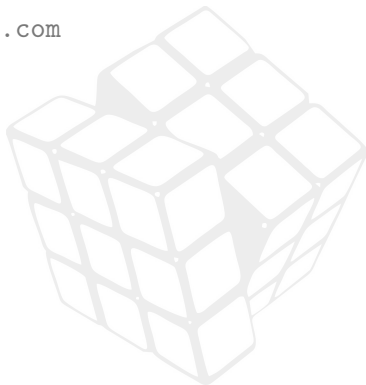


Algoritmos e Programação

Arquivos

Gustavo Sávio
gsoprofessor@gmail.com

2017.2



É uma representação digital de um conjunto de dados, utilizado para armazenar informações.

- ▶ Normalmente utilizam armazenamento não volátil
 - ▶ Exemplo: Disco Rígido

```
1 arquivo = open(nome_arquivo, modo_leitura, encoding)
```

Caractere	Significado	Padrão
'w'	Escrita	
'r'	Leitura	Sim
'a'	Adição	
'x'	Apenas para criação, falha se o arquivo existir	
'b'	Binário	
't'	Modo texto	Sim
+	Abre um arquivo para atualização (leitura e escrita)	

open - Criando um arquivo texto

```
1 # Cria um arquivo texto chamado dados.txt
2 arquivo = open('dados.txt', 'w', encoding='UTF-8')
3
4 # Fecha o arquivo
5 arquivo.close()
```

write - Criando e escrevendo em arquivo texto

```
1 # Cria um arquivo texto chamado dados.txt
2 arquivo = open('dados.txt', 'w', encoding='UTF-8')
3
4 # Escreve no arquivo
5 arquivo.write('Escrevendo uma linha no arquivo texto\n')
6 arquivo.write('Escrevendo uma nova linha no arquivo texto\n')
7
8 # Fecha o arquivo
9 arquivo.close()
```

writelines - Criando e adicionando listas em arquivo texto

```
1  # Lista
2  lista_nomes = ['Gustavo', 'João', 'Maria', 'Rita']
3  lista_carros = ['Ferrari', 'Porsche', 'Fiat', 'BMW']
4
5  # Cria um arquivo texto chamado dados.txt
6  arquivo = open('dados.txt', 'w', encoding='UTF-8', newline='\n')
7
8  # Escreve no arquivo
9  arquivo.writelines(lista_nomes)
10 arquivo.writelines(lista_carros)
11
12 # Fecha o arquivo
13 arquivo.close()
```

read - Lendo informações

```
1 # Abre o arquivo em modo leitura
2 arquivo = open('dados.txt', 'r', encoding='UTF-8')
3
4 # Ler todo o arquivo e coloca em uma string
5 print(arquivo.read())
6
7 # Fecha o arquivo
8 arquivo.close()
```

readline - Lendo as linhas do arquivo

```
1 # Abre o arquivo em modo leitura
2 arquivo = open('dados.txt', 'r', encoding='UTF-8')
3
4 # Ler todo o arquivo e coloca em uma string
5 print(arquivo.readline())
6
7 # Fecha o arquivo
8 arquivo.close()
```

readlines - Lendo as linhas do arquivo e retornando uma lista

```
1  # Abre o arquivo em modo leitura
2  arquivo = open('dados.txt', 'r', encoding='UTF-8')
3
4  # Ler todo o arquivo e coloca em uma string
5  lista = arquivo.readlines()
6
7  print(lista)
8
9  for elemento in lista:
10     print(elemento)
11
12 # Fecha o arquivo
13 arquivo.close()
```

Adicionando informações em arquivos existentes

```
1 # Abre o arquivo em modo de adição
2 arquivo = open('dados.txt', 'a', encoding='UTF-8')
3
4 arquivo.write('Adicionando uma linha')
5
6 # Fecha o arquivo
7 arquivo.close()
```

Lendo uma linha em específico

```
1 # Abre o arquivo em modo de adição
2 arquivo = open('dados.txt', 'r', encoding='UTF-8')
3
4 # Ler uma linha em específico
5 linha = arquivo.readlines()[10]
6
7 # Fecha o arquivo
8 arquivo.close()
```



Criando um arquivo binário

```
1 lista =[b'Escrevendo no arquivo', b'Escrevendo a segunda linha']
2
3 # Criando um arquivo binário
4 binario = open('binario.bin', 'wb')
5
6 # Escrevendo um valor
7 binario.write(b'escrevendo no arquivo\n')
8
9 # Escrevendo vários valores
10 binario.writelines(lista)
11
12 # Fechando o arquivo
13 binario.close()
```

Lendo um arquivo binário

```
1 # Lendo um arquivo binário
2 binario = open('binario.bin', 'rb')
3
4 # Iterando sobre os valores
5 for b in binario:
6     print(b)
7
8 # Obtendo os valores
9 print(binario.readlines())
10
11 # Fechando o arquivo
12 binario.close()
```

```
1 # Abrindo o arquivo
2 binario = open('binario.bin', 'rb')
3
4 # Seek move o cursor do arquivo para a posição do byte informado
5 binario.seek(10)
6
7 # tell() indica a posição atual
8 print(binario.tell())
9
10 # Ler a posição atual
11 print(binario.readline())
12
13 # Fechando o arquivo
14 binario.close()
```

-  Allen B. Downey; Think Python How to Think Like a Computer Scientist.
-  Docs Python 3. <https://docs.python.org/3/>