



UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE
DEPARTAMENTO DE INFORMÁTICA E MATEMÁTICA APLICADA

DIM0320 Algoritmos e Programação de Computadores



#MANIPULANDO ARQUIVOS

ELIEZIO SOARES

ELIEZIOSOARES@DIMAP.UFRN.BR

Dados Temporários

- Todos os dados trabalhados em nossos programas são armazenados na memória principal.
- A memória principal é volátil.
- Logo, ao terminar a execução do programa, os dados somem.

Arquivos

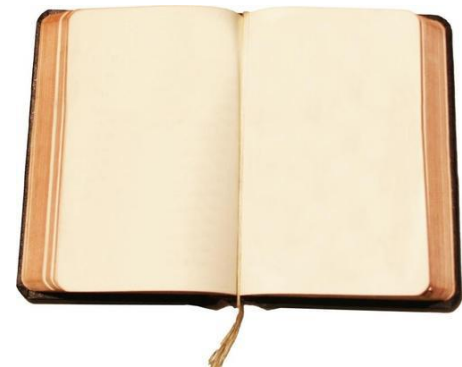
- Um arquivo é um registro armazenado em dispositivos de memória secundária.
- Armazenamento persistente.
- Disco rígido (HD), SSD, pen-drive, cd, dvd...
- O nome de um arquivo é composto por:
 - O diretório +
 - Seu nome;

Abrindo Arquivos

- Para ler ou escrever um arquivo em um programa Python é preciso primeiro abri-lo:

```
arquivo = open(nomeArquivo, 'r')
```

- A função open recebe dois parâmetros:
 - Nome do arquivo (diretório + nome do arquivo + extensão)
 - Modo de acesso: **w** (write / escrita), ou **r** (read / leitura)
- A função open retorna uma instância do arquivo a ser manipulado.
- Caso não exista o arquivo, o mesmo será criado.



Lendo Arquivos

- Uma vez que um arquivo foi aberto, podemos lê-lo:

```
arquivo = open('idsOrdenados.txt', 'r')
conteudo = arquivo.read()
print(conteudo)
```

- A função `read` lê o conteúdo do arquivo:
 - A função retornará todo o conteúdo do arquivo.
 - A função pode receber um parâmetro inteiro correspondente a quantidade de caracteres a ser lido.

Lendo Arquivos de Texto

- O método `readline` lê todos os caracteres da próxima linha.

```
print("\n\t\t Abrindo arquivo desordenado.")

arquivo = open('idsOrdenados.txt', 'r')
linha = "."
while linha != "":
    linha = arquivo.readline()
    print(linha)

print("\n\t\t Fim do arquivo.")
arquivo.close()
```

Lendo Arquivos de Texto

- O método `readlines` retorna todas as linhas do arquivo em uma lista.

```
arquivo = open('idsOrdenados.txt', 'r')
linhas = arquivo.readlines()
for linha in linhas:
    print(linha)

print("\n\t\t Fim do arquivo.")
arquivo.close()
```

Escrevendo Arquivos

- O método `write` permite escrever dados dentro do arquivo:
 - A função recebe um parâmetro do tipo string.
 - O arquivo foi aberto no modo “w” (escrita).

```
def gerarListaOrdenada():  
    quantidade = int(input('Informe a quantidade de elementos: '))  
    arquivo = open('idsOrdenados.txt', 'w')  
  
    for elemento in range(quantidade):  
        arquivo.write(str(elemento)+"\n")  
  
    arquivo.close()
```

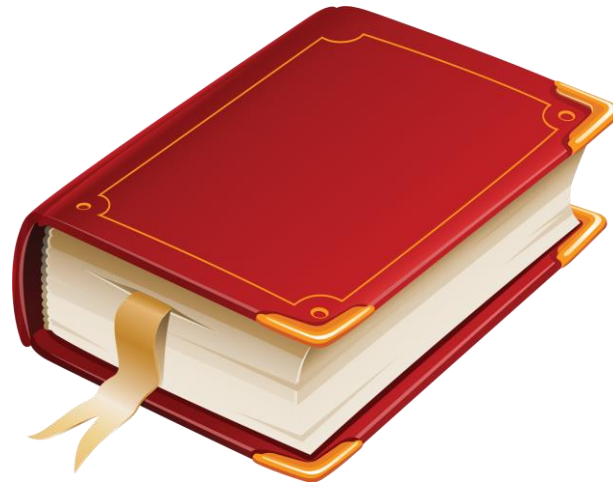

Escrevendo Arquivos

- O método `writelines` permite escrever dados dentro do arquivo de uma só vez.
 - A função recebe uma lista de strings como parâmetro.
 - O arquivo foi aberto no modo “w” (escrita).

```
def gerarListaOrdenada():  
    lista = []  
    quantidade = int(input('Informe a quantidade de elementos: '))  
    arquivo = open('idsOrdenados.txt', 'w')  
  
    for elemento in range(quantidade):  
        lista.append(str(elemento)+"\n")  
  
    arquivo.writelines(lista)  
    arquivo.close()
```

Fechando Arquivos

- A função `close()` informa que o programa terminou de usar o arquivo.



Exemplo 1

```
def copiaArquivo(velhoArquivo, novoArquivo):  
    f1 = open(velhoArquivo, "r")  
    f2 = open(novoArquivo, "w")  
    while 1:  
        texto = f1.read(50)  
        if texto == "":  
            break  
        f2.write(texto)  
    f1.close()  
    f2.close()  
    return
```

Exemplo 2

```
def gerarListaInversamenteOrdenada():
    lista = []
    quantidade = int(input('Informe a quantidade de elementos: '))
    arquivo = open('idsOrdenadosInv.txt', 'w')

    quantidade -= 1
    while quantidade >= 0:
        lista.append(str(quantidade) + "\n")
        quantidade -= 1

    arquivo.writelines(lista)
    arquivo.close()
```

Dúvidas



Exercício 1

- 1) Desenvolva um algoritmo que escreve em disco um arquivo com números ordenados crescentemente de 1 a 100. Cada número deve ser separado por “;”. O arquivo deve se chamar “crescente.txt”.

- 2) Desenvolva um algoritmo que escreve em disco um arquivo com números ordenados decrescentemente de 100 a 1. Cada número deve estar em uma linha. O arquivo deve se chamar “decrescente.txt”.

Exercício 2

1) Desenvolva um programa que cadastra alunos:

- O programa deve registrar nome, email e curso do aluno.
- Cada novo registro deve ser armazenado em arquivo.
- O usuário deve ter as seguintes opções:
 - Cadastrar novo aluno;
 - Listar alunos cadastrados;
 - Buscar aluno;
 - Remover aluno