

Disciplina:

Linguagem de Programação

Arquivo em Python

Prof^a. Dr^a. Giovana Angélica Ros Miola
giovana.miola@fatec.sp.gov.br



Como armazenar dados permanentemente?

- Arquivos são uma excelente forma de entrada e saída de dados para programas. Com eles, podemos ler dados de outros programas e ou até mesmo da internet.
- O que é um arquivo? Um arquivo é uma área em disco onde podemos ler e gravar informações. Um arquivo é acessado por nome e é onde podemos ler e escrever linhas de texto ou dados em geral.
- A maioria das linguagens de programação possuem meios para essa manipulação.
- Para acessar um arquivo, precisamos abri-lo.

Trabalhando com arquivo

- Na abertura de um arquivo, informamos o nome do arquivo, com o nome da pasta onde ele se encontra (se necessário) e que **operações** queremos realizar: **leitura** e/ou **escrita**.
- A função **open** (abertura de arquivo), utiliza os parâmetros **nome** e **modo**.
- O **nome** é o nome do arquivo em si, por exemplo, leiname.txt
- O **modo** indica as operações que podem ser realizadas.
- **Modos** de abertura dos arquivos:

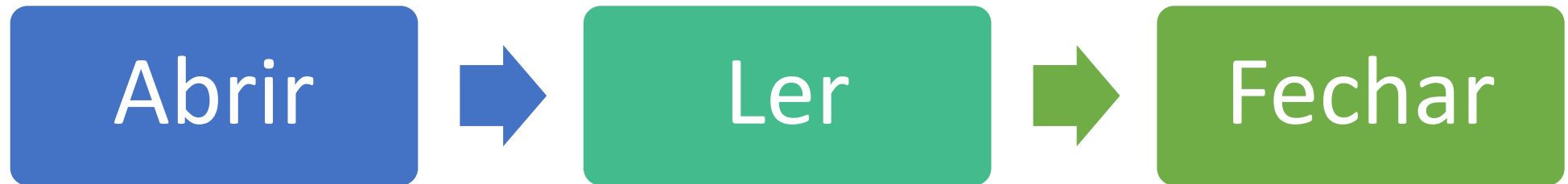
Modo	Operações
r	leitura
w	escrita, apaga o conteúdo se já existir
a	escrita, mas preserva o conteúdo se já existir
b	modo binário
+	atualização (leitura e escrita)

Trabalhando com arquivo

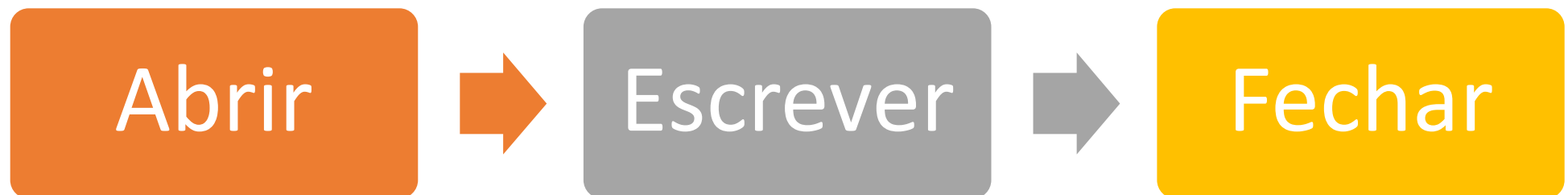
- Os modos podem ser combinados (“r+”, “w+”, “a+”, “r+b”, “w+b”, “a+b”). Verifique quando for necessário e olhe a documentação da linguagem.
- A função **open** retorna um objeto do tipo file (arquivo). É esse objeto que vamos utilizar para ler os dados no arquivo
- A função **write** escreve ou gravar dados no arquivo
- A função **close** fecha um arquivo.

Trabalhando com arquivo

Ao trabalhar com arquivos, **devemos sempre realizar o seguinte ciclo e exatamente nesta ORDEM**



..... ou



Cuidado com arquivo aberto

- A abertura realiza a ligação entre o programa e o espaço em disco, gerenciado pelo sistema operacional.
- As etapas de leitura e/ou escrita são as operações que desejamos realizar no programa e o **fechamento** informa ao sistema operacional que **não vamos mais trabalhar com o arquivo**.
- O **fechamento** do arquivo é muito importante, pois cada arquivo aberto consome recursos do computador.
- Só o fechamento do arquivo garante a liberação desses recursos e preserva a integridade dos dados do arquivo.

Abrindo, escrevendo e fechando um arquivo

```
arquivo = open('numeros.txt', 'w') # apagando e criando
for linha in range(1,21):
    arquivo.write('%d\n' % linha)
arquivo.close()
```

- Quando executar este código, nada aparecerá na tela. O programa criará um novo arquivo na pasta atual, ou seja, no mesmo local em que você gravou seu programa.
- O modo "w" cria o arquivo se ele não existir. Caso já exista, seu conteúdo é apagado.

Abrindo, escrevendo e fechando um arquivo

```
arquivo = open('numeros.txt', 'a') # append
for linha in range(1,11):
    arquivo.write('%d\n' % linha)
arquivo.close()
```

- O modo "a" cria o arquivo para escrever, mas preserva o que já existe nele

Abrindo, escrevendo e fechando um arquivo

```
arquivo = open('numeros.txt', 'r')  
for x in arquivo.readlines():  
    print(x)  
arquivo.close()
```

- O modo "r" abre o arquivo apenas para leitura
- O método/função `readlines`, gera uma lista em que cada elemento é uma linha do arquivo.

Trabalhando com dois arquivos

- Gravação de números pares e ímpares em arquivos diferentes:

```
impares = open("impares.txt", "a")
pares = open("pares.txt", "a")
for n in range(1,51):
    if n % 2 == 0:
        pares.write("%d\n" % n)
    else:
        impares.write("%d\n" % n)
impares.close()
pares.close()
```

Arquivo com estrutura

Fórmula referente ao exercício 4 do Colab

$$\text{IMC} = \frac{\text{PESO}}{(\text{ALTURA})^2}$$

```
class TipoFuncionario:
    matricula = 0
    nome = ""
    salario = 0.0

def Cadastrar(vetFuncionario):
    arquivo = open('arq_Funcionario.txt', 'a')
    print('Cadastro de Funcionários.....')
    for i in range(3):
        f = TipoFuncionario()
        f.matricula = int(input('Digite a matrícula: '))
        f.nome = input('Digite o nome: ')
        f.salario = float(input('Digite o salário: '))
        vetFuncionario.append(f)
        arquivo.write(f'{f.matricula} {f.nome} {f.salario:.2f}\n')
    #ou
    #arquivo.write('%d %s %.2f\n' % (f.matricula, f.nome, f.salario))
    arquivo.close()
    return vetFuncionario
```

```
def Mostrar():
    arquivo = open('arq_Funcionario.txt', 'r')
    print('Apresentação dos dados dos funcionários .....')
    print('Matrícula\tNome\tSalário')
    for linha in arquivo.readlines():
        mat, nome, sal = linha.strip().split(" ")
        print(mat, '\t\t', nome, '\t', sal)
    arquivo.close()

def main():
    op = 1
    while op >= 1 and op <= 2:
        print('*** Gerenciamento de Funcionários COM ARQUIVO ***')
        print('1- Cadastrar')
        print('2- Mostrar')
        print('3- Sair')
        op = int(input('Digite a opção: '))
        if op == 1:
            vetF=[]
            vetF = Cadastrar(vetF)
        elif op == 2:
            Mostrar()

main()
```