



Algoritmos e Programação de Computadores

Aula 08 - Arquivos

Prof. Dr. Rodolfo Carneiro Cavalcante
rodolfo.cavalcante@arapiraca.ufal.br
Universidade Federal de Alagoas
Campus de Arapiraca



Introdução

- Entrada e saída são operações de comunicação de um programa com o mundo externo
- Essa comunicação pode ser de várias formas
 - Impressão na tela
 - Envio de dados pela rede (via porta)
 - Leitura e escrita de arquivos
- Em Python, um arquivo pode ser lido/escrito por meio de um objeto da classe file

Introdução



- As funções que realizamos com arquivos são basicamente:
 - Abrir
 - Fechar
 - Ler
 - Escrever
- Ao abrir o arquivo, devemos informar qual o propósito (leitura ou escrita)

Arquivos



Abrindo arquivo

```
arquivo = open('nome_arquivo',modo)
```

- Nome: nome do arquivo que será aberto (com diretório)
- Modo: mode de abertura (opcional)
 - r: leitura (default)
 - w: escrita (sobrescreve o arquivo, se já existente)
 - a: adiciona ao arquivo se já existente (append)
 - r+: leitura e escrita
 - b: binário (rb,wb,ab)

Arquivos



Abrindo arquivo para escrita

```
arq = open('teste.txt', 'w')  
arq.write('Olá, mundo!')  
arq.close()
```

- Escreve a string no arquivo
- Uso de buffers para escrita
 - Por isso precisa do `close()` para garantir a escrita física
- Use `'\n'` para pular linha no arquivo

Arquivos



Abrindo arquivo para leitura

```
arq = open('teste.txt','r')  
texto = arq.read( )  
print(texto)  
arq.close( )
```

Arquivos



Abrindo arquivo para ler várias linhas

```
arquivo.readlines( )
```

- Retorna uma lista com as linhas do texto
- O caractere ‘\n’ é exibido junto com a string na linha

Arquivos



Abrindo arquivo para leitura

```
arq = open('teste.txt','r')
for line in arq.readlines():
    print(line)
arq.close( )
```


Arquivos



Abrindo arquivo para adicionar dados

```
arq = open('teste.txt','a')  
arq.write('Olá, novamente!')  
arq.close( )
```

```
arq = open('teste.txt')  
texto = arq.read( )  
print(texto)  
arq.close( )
```

Arquivos



Abrindo arquivo para escrever dados

```
nomes = ['ana','bob','pedro','joao']  
for nome in nomes:  
    arquivo.write(nome+'\n')  
arquivo.write(string)
```

Arquivos e SO



- Operações de entrada e saída são na verdade realizadas pelo sistema operacional
- O módulo os possui variáveis e funções que ajudam um programa Python a se adequar ao sistema operacional
- Processamento de arquivos tipicamente demandam por interações com o SO

Arquivos e SO



Funções importantes da biblioteca os

```
import os
# retorna o diretório corrente
os.getcwd( )
# muda o diretório corrente para dir
os.chdir(dir)
#verifica se um arquivo existente
os.path.exists(path)
#retorna a lista dos arquivos naquele diretório
os.listdir(path)
```

Exercícios



1. Escreva um programa que lê um arquivo contendo a identidade e o nome de várias pessoas, no seguinte formato:

5384423 Manoel

4345566 Alberto

3235574 Mariana

o programa deve gerar um dicionário no qual as chaves são as identidades e os valores os nomes. Ao final o programa deve exibir o dicionário

Exercícios



2. Escreva um programa que lê um arquivo contendo endereços IPs, da seguinte forma:

200.135.80.9

192.168.1.1

8.35.67.74

257.32.4.5

O programa deve mostrar os IPs indicando os que são válidos e inválidos (um endereço ip válido não pode ter uma de suas partes maior que 255).

Exercícios



3. Faça um programa que leia um número qualquer de notas em um arquivo. Após a leitura dos dados, faça o seguinte:

- Mostre a quantidade de notas que foram lidas.
- Exiba todas as notas na ordem em que foram informadas.
- Calcule e mostre a soma das notas.
- Calcule e mostre a média das notas.
- Calcule e mostre a quantidade de notas acima da média calculada.

Exercícios



4. Faça um programa que leia a temperatura média de cada mês do ano em um arquivo e armazene-as em uma lista. Em seguida, calcule a média anual das temperaturas e mostre a média calculada juntamente com todas as temperaturas acima da média anual, e em que mês elas ocorreram (mostrar o mês por extenso: 1 - Janeiro, 2 - Fevereiro, . . .).

Exercícios



5. Implemente uma agenda telefônica onde o usuário pode inserir, remover, editar e buscar por telefones de contatos. Cada contato tem um nome e apenas um telefone. Utilize um arquivo para servir como banco de dados da aplicação.

Pandas



- Python for Data Analysis
- Manipulação de dados estruturados
- Tabelas em vários formatos
 - csv, excel, sql
- Dados semi-estruturados
 - json, xml
- E vários outros



Pandas



- Instalação

```
pip install pandas
```

- Uso

```
import pandas as pd
```

DataFrame



- Estrutura de dados do pandas
- Estrutura de dados de 2 dimensões (tabela) com colunas rotuladas
- Cada coluna é um vetor rotulado



Leitura de Dados

- Arquivo csv (comma separated vector)

```
df = pd.read_csv('filename.csv')
```

```
print(df.head( ))  
print(df.head(20))  
print(df.columns)  
print(df.tail( ))  
print(df.tail(10))
```

Leitura de Dados



- Arquivo csv (comma separated vector)

```
print(df.name)
print(df['name'])
```

```
print(df.sort_values(by= 'created_at'))
```

```
print(df.quantity.sort_values)
```

```
print(df.loc[0])
```

```
venda = df.loc[0]
print(venda.email)
```

Leitura de Dados



- Arquivo csv (comma separated vector)

```
df[df.email == 'jaida.schaden@example.com']  
df[df.quantity > 5]
```

Escrita de Dados



- Dataframe é um dicionário
- O nome da coluna é uma chave
- Valores relacionados a uma chave são uma lista ou vetor
- Podemos escrever um dicionário nesse formato em arquivo

Escrita de Dados



```
alunos = ['ana', 'carlos', 'gilberto', 'marta']  
trabalhos = [9,8,7,6]  
provas = [9,8,7,6]  
seminarios = [9,8,7,6]  
artigos = [9,8,7,6]
```

```
df = pd.DataFrame({'aluno': alunos, 'trabalho': trabalhos, 'prova': provas,  
                  'seminário': seminarios, 'artigo': artigos})  
df.to_csv('notas.csv', index=False)
```

Exercícios



1. Faça um programa que lê um arquivo csv contendo dados de temperatura e umidade em vários dias em uma cidade e diga qual foi o dia mais quente o mais frio da cidade.
2. Faça um programa que recebe do usuário duas notas de vários alunos e escreve um arquivo csv contendo as duas notas e a média do aluno.

Exercícios



3. Faça um programa que recebe dados de vários contatos (nome, telefone, email) e escreve os dados em um arquivo csv. O programa tem uma função que lê o arquivo e faz busca pelo nome do contato.



Algoritmos e Programação de Computadores

Aula 08 - Arquivos

Prof. Dr. Rodolfo Carneiro Cavalcante
rodolfo.cavalcante@arapiraca.ufal.br