

Arquivos de Dados

CIÊNCIA DA
COMPUTAÇÃO

ATITUS
EDUCAÇÃO



ATITUS
EDUCAÇÃO

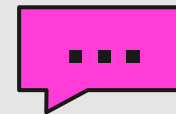


- Aula 19 -
Pensamento Computacional

Prof. Me. Lucas R. C. Pessutto



Na aula de hoje...



01

Arquivos JSON



02

Arquivos CSV

Arquivos...

- * Permitem que programas façam armazenamento **permanente** (ou **persistente**) de informações
- * Pode-se utilizar os mesmos dados entre execuções diferentes de um programa ou de programas diferentes
- * Arquivos são armazenados em dispositivos de **memória secundária** (disco rígido, CD, DVD, pendrives)
- * Podem armazenar mais dados que a memória principal comporta



Arquivos em Python

Um arquivo é um conjunto de *bytes*, colocados um após o outro (*stream*), armazenados em um dispositivo de memória secundária

- * A linguagem não impõe estrutura alguma aos arquivos
- * Arquivos podem ser usados de duas formas:
 - Como **fonte** de dados para o programa
 - Arquivo de entrada (**input**)
 - Como **destino** de dados de um programa
 - Arquivo de saída (**output**)

Arquivos de Dados

JSON (JavaScript Object Notation)

É um formato leve de troca de dados inspirado pela sintaxe de objetos JavaScript

```
{  
  "items": ["i1", "i2", "i3"],  
  "numero": 1234,  
  "ativo": true  
}
```

Padronizado pela [RFC 7159](#)

O módulo `json` oferece duas funções principais para carregar (`load`) e escrever (`dump`) arquivos nesse formato

Documentação completa em docs.python.org/pt-br/3/library/json.html

CSV (Comma-Separated Values)

Formato mais comum de importação e exportação de planilhas e bancos de dados

```
header-1, header-2, header-N  
dado-1.1, dado-1.2, dado-1.N  
dado-2.1, dado-2.2, dado-2.N  
...  
dado-M.1, dado-M.2, dado-M.N
```

Padronizado pela [RFC 4180](#)

O módulo `csv` em Python oferece objetos `reader` e `writer` para manipular arquivos nesse formato

Documentação completa em docs.python.org/pt-br/3/library/csv.html

Arquivos JSON

Chaves

{

```
"id": 1,  
"Modelo": "Celta",  
"Placa": "AAA1234"
```

}

Valores

Os dados no arquivo json são organizados da mesma forma que em um dicionário!

Arquivos JSON

* Primeiramente, o módulo `json` não é “nativo” ele precisa ser importado em programas que o utilizem com `import json`

`json.load(fp, ...)`

* Desserializa um arquivo (`fp`) com suporte a `.read()` contendo um documento JSON, carregando seu conteúdo e retornando um objeto Python

`json.dump(obj, fp, ...)`

* Serializa um objeto Python (`obj`) como um stream JSON formatado descarregando em um arquivo (`fp`) com suporte a `.write()`

Problema 1: Hashmon®

Enunciado de um problema:

Você resolveu criar um jogo de cartas chamado **Hashmon®**. No seu jogo, cada carta representa um personagem com as seguintes características: **nome** (str), **tipo** (grama, água ou fogo) (str), **HP** (quantidade de vida) (int) e **fraqueza** (os mesmos valores do tipo) (str). Para auxiliar na criação do baralho você resolve fazer um programa em Python onde você vai digitando iterativamente as características das cartas e o programa gera um aviso em duas situações:

1. caso você tente inserir uma carta com nome repetido
2. caso você tente inserir uma carta com as outras características repetidas

Seu programa deve permitir salvar e recuperar de um arquivo seu baralho de Hashmons

Problema 3: Hashmon®



Chaves: nome (str)



Valores: características (dict)

Descrição dos Dados
e Algoritmos



```
baralho = {  
    "Hashzard": {  
        "hp": 20,  
        "tipo": "fogo",  
        "fraqueza": "grama"  
    },  
    "Squirrel":  
    {...}  
}
```

Carregando o Baralho

Vamos usar `json.load` para carregar o conteúdo de um arquivo JSON e definir o conteúdo inicial do baralho

```
import json
```

```
def carregar_baralho():
```

```
    try:
```

```
        with open("baralho.json", "r") as arquivo:
```

```
            return json.load(arquivo)
```

```
    except FileNotFoundError:
```

```
        print("O arquivo não foi encontrado!")
```

Se existir, o arquivo "baralho.json" será aberto para leitura ("r")

`json.load` vai tratar de desserializar todo o conteúdo do arquivo sem necessidade de um laço de iteração

Se o arquivo não existe (`FileNotFoundError`) não é um problema, pode ser a primeira execução do programa

Salvando o Baralho

Vamos usar `json.dump` para escrever a lista completa no arquivo texto `"baralho.json"` aberto para escrita (`"w"`)

```
import json
```

```
def salvar_baralho(dicionario):
```

```
    try:
```

```
        with open("baralho.json", "w") as arquivo:
```

```
            json.dump(dicionario, arquivo)
```

```
    except OSError:
```

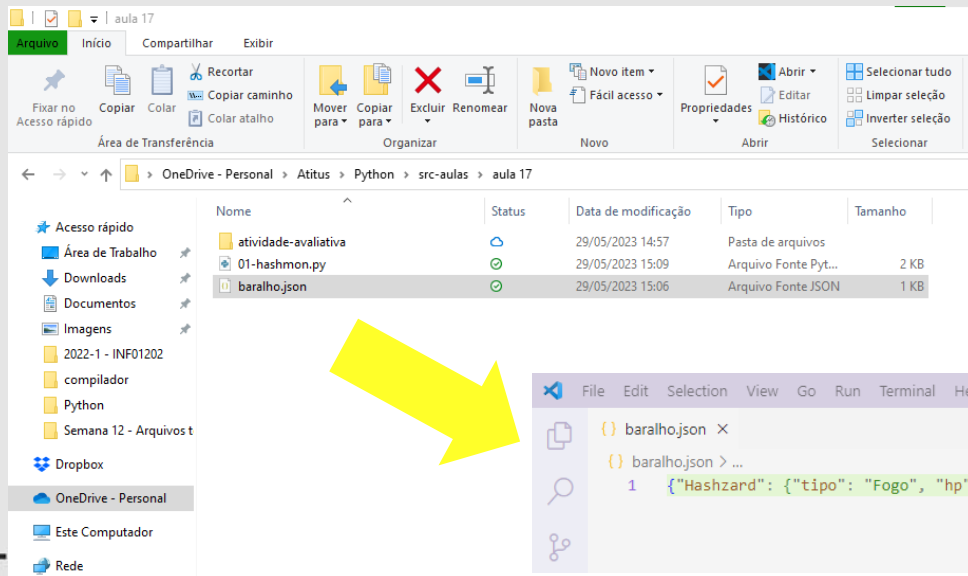
```
        print("Erro ao salvar")
```

Um arquivo chamado "baralho.json" será criado (ou sobrescrito) no mesmo diretório onde estiver rodando o programa em Python

Aqui, novamente, não há necessidade de iterar sobre a lista, todos os itens serão serializados pelo `json.dump` e salvos no arquivo

Arquivos JSON

* Após a execução do código você deve conseguir acessar o arquivo criado com o conteúdo da sua lista de compras em formato JSON



Exercício: Países/Cidades/Estados

- * Utilizando o arquivo `paises-estados-cidades.json`, que contém informações sobre todos os países, estados e cidades do planeta.
- * Crie um arquivo chamado `cidades_A.json`, que contém as informações das cidades cujo nome inicia com a letra “A”.

```
{  
  "Alegrete": {  
    "País": "Brasil",  
    "Estado": "Rio Grande do Sul"  
  },  
  "Altamira": {  
    "País": "Brasil",  
    "Estado": "Pará"  
  }  
}
```

Arquivos CSV

quantitativo-de-alunos.csv - Notepad

File Edit Format View Help

Cabeçalhos

CodCurso	NomeCurso	Ano	Periodo	Vinculados	Matriculados	Ingressantes	Diplomados	Evadidos
298	ADMINISTRAÇÃO	2010	1	1501	1337	166	91	24
583	ADMINISTRAÇÃO - EAD	2010	1	351	345	0	0	6
299	AGRONOMIA	2010	1	495	441	47	31	20
300	ARQUITETURA E URBANISMO	2010	1	625	557	53	41	12
301	ARQUIVOLOGIA	2010	1	157	134	30	4	6
303	ARTES VISUAIS	2010	1	593	498	81	37	19
801	ARTES VISUAIS - ENSINO A DISTÂNCIA - EAD	2010	1	90	90	0	0	0

Caractere delimitador

Nem sempre os arquivos CSV vão incluir cabeçalhos. O ideal é se orientar pelo dicionário de dados.

Um caractere específico é escolhido para delimitar os campos. Os mais comuns são vírgula, ponto e vírgula e tabulação (\t).

É preciso também verificar a codificação do arquivo CSV. Normalmente a informação é fornecida junto com o arquivo. No caso desse conjunto de dados específico a codificação é **utf-8**.

Arquivos CSV

* O módulo csv também não é “nativo” e precisa ser importado em programas que o utilizem com `import csv`

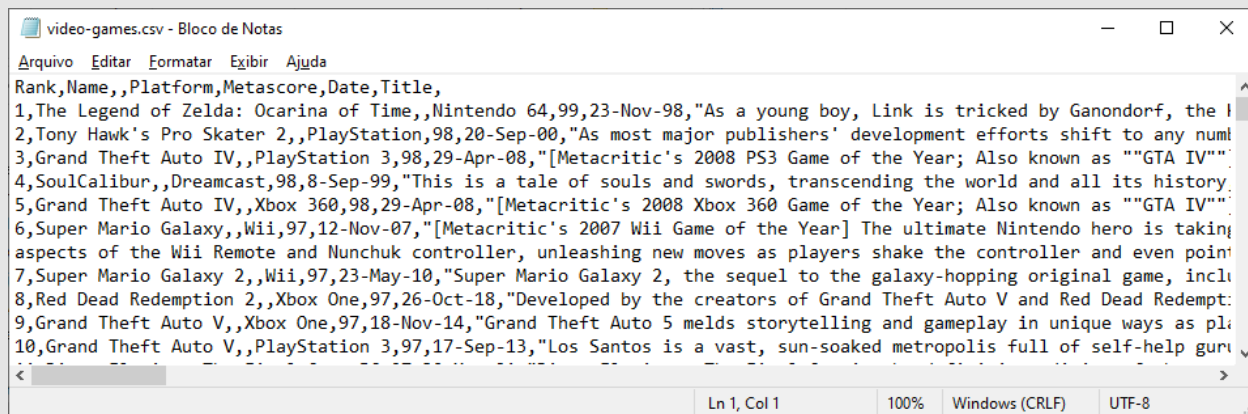
`csv.reader(csvfile, ...)`

* Retorna um objeto de leitura para iterar sobre as linhas do csvfile fornecido. No geral, csvfile deve ser um arquivo texto aberto para leitura, mas pode ser outro tipo de iterável.

`csv.writer(csvfile, ...)`

* Retorna um objeto de escrita capaz de converter os dados em strings delimitadas para gravação em arquivos. No geral, csvfile será um arquivo aberto para escrita.

Exemplo: Top-100 Videogames



```
Rank,Name,,Platform,Metascore,Date,Title,
1,The Legend of Zelda: Ocarina of Time,,Nintendo 64,99,23-Nov-98,"As a young boy, Link is tricked by Ganondorf, the
2,Tony Hawk's Pro Skater 2,,PlayStation,98,20-Sep-00,"As most major publishers' development efforts shift to any num
3,Grand Theft Auto IV,,PlayStation 3,98,29-Apr-08,"[Metacritic's 2008 PS3 Game of the Year; Also known as ""GTA IV""
4,SoulCalibur,,Dreamcast,98,8-Sep-99,"This is a tale of souls and swords, transcending the world and all its history
5,Grand Theft Auto IV,,Xbox 360,98,29-Apr-08,"[Metacritic's 2008 Xbox 360 Game of the Year; Also known as ""GTA IV""
6,Super Mario Galaxy,,Wii,97,12-Nov-07,"[Metacritic's 2007 Wii Game of the Year] The ultimate Nintendo hero is taking
aspects of the Wii Remote and Nunchuk controller, unleashing new moves as players shake the controller and even poin
7,Super Mario Galaxy 2,,Wii,97,23-May-10,"Super Mario Galaxy 2, the sequel to the galaxy-hopping original game, inclu
8,Red Dead Redemption 2,,Xbox One,97,26-Oct-18,"Developed by the creators of Grand Theft Auto V and Red Dead Redempt
9,Grand Theft Auto V,,Xbox One,97,18-Nov-14,"Grand Theft Auto 5 melds storytelling and gameplay in unique ways as pla
10,Grand Theft Auto V,,PlayStation 3,97,17-Sep-13,"Los Santos is a vast, sun-soaked metropolis full of self-help guri
```

Rank	Name		Platform	Metascore	Date	Title
1	The Legend of Zelda: Ocarina of Time		Nintendo 64	99	23-Nov-98	As a young boy, Link is tricked by Ganondorf, t
2	Tony Hawk's Pro Skater 2		PlayStation	98	20-Sep-00	As most major publishers' development efforts
3	Grand Theft Auto IV		PlayStation 3	98	29-Apr-08	[Metacritic's 2008 PS3 Game of the Year; Also k
4	SoulCalibur		Dreamcast	98	8-Sep-99	This is a tale of souls and swords, transcending
5	Grand Theft Auto IV		Xbox 360	98	29-Apr-08	[Metacritic's 2008 Xbox 360 Game of the Year; ,
6	Super Mario Galaxy		Wii	97	12-Nov-07	[Metacritic's 2007 Wii Game of the Year] The ul
7	Super Mario Galaxy 2		Wii	97	23-May-10	Super Mario Galaxy 2, the sequel to the galaxy-
8	Red Dead Redemption 2		Xbox One	97	26-Oct-18	Developed by the creators of Grand Theft Auto
9	Grand Theft Auto V		Xbox One	97	18-Nov-14	Grand Theft Auto 5 melds storytelling and gam
10	Grand Theft Auto V		PlayStation 3	97	17-Sep-13	Los Santos is a vast, sun-soaked metropolis full

Exemplo: Top-100 Videogames

* Lendo o arquivo e escrevendo cada linha na tela:

```
import csv
```

```
with open("video-games.csv", "r", encoding="utf-8") as arquivo:
    csv_reader = csv.reader(arquivo, delimiter=",")
    titulos = next(csv_reader) # "pula" os cabeçalhos
    for linha in csv_reader:
        print(linha)
```

O método `csv.reader` retorna um objeto leitor para iterar sobre as linhas do arquivo `_csv`. O delimitador de campos do arquivo é `,`.

A variável `linha` vai representar cada linha do arquivo CSV como uma lista

Primeira Linha do arquivo:

```
['1', 'The Legend of Zelda: Ocarina of Time', '', 'Nintendo 64', '99',  
'23-Nov-98', 'As a young boy, Link is ... of the Seven Sages.', '']
```

Exemplo: Top-100 Videogames

- * Lendo o arquivo e escrevendo cada linha na tela:

```
import csv
```

```
with open("video-games.csv", "r", encoding="utf-8") as arquivo:
    csv_reader = csv.reader(arquivo, delimiter=",")
    titulos = next(csv_reader) # "pula" os cabeçalhos
    for linha in csv_reader:
        print(f"| Rank      : {linha[0]}")
        print(f"| Título    : {linha[1]}")
        print(f"| Plataforma: {linha[3]}")
        print(f"| Lançamento : {linha[5]}")
        print('-' * 30)
```

Exemplo: Top-100 Videogames

- * Lendo os dados do arquivo para dentro de um dicionário

```
import csv

with open("video-games.csv", "r", encoding="utf-8") as
arquivo:
    csv_reader = csv.DictReader(arquivo, delimiter=",")
    for linha in csv_reader:
        print(linha)
```

Primeira Linha do arquivo:

```
{'Rank': '1', 'Name': 'The Legend of Zelda: Ocarina of Time', '': '',
'Platform': 'Nintendo 64', 'Metascore': '99', 'Date': '23-Nov-98',
'Title': 'As a young boy, Link is ... of the Seven Sages.'}
```

Exemplo: Top-100 Videogames

- * Lendo os dados do arquivo para dentro de um dicionário

```
import csv
```

```
with open("video-games.csv", "r", encoding="utf-8") as arquivo:
    csv_reader = csv.DictReader(arquivo, delimiter=",")
    for linha in csv_reader:
        print(f"| Rank      : {linha['Rank']}")
        print(f"| Título    : {linha['Name']}")
        print(f"| Plataforma : {linha['Platform']}")
        print(f"| Lançamento : {linha['Date']}")
        print('-' * 30)
```

Exemplo de Gravação de Dados

* Salvando dados linha a linha no arquivo:

```
import csv
```

Esse parâmetro faz com que o arquivo não escreva o \n no arquivo duas vezes

```
with open("alunos.csv", "w", newline='', encoding="utf-8") as arquivo:
```

```
    csv_writer = csv.writer(arquivo, delimiter=";")
```

```
    csv_writer.writerow(["Nro", "Nome", "Nota"])
```

```
    csv_writer.writerow([1, "Pedro Carlos", 9.5])
```

```
    csv_writer.writerow([2, "Maria Clara", 9.9])
```

```
    csv_writer.writerow([3, "Enzo Gabriel", 9.2])
```

"Linha de Cabeçalho"

Delimitador pode ser personalizado

```
alunos.csv
1  Nro;Nome;Nota
2  1;Pedro Carlos;9.5
3  2;Maria Clara;9.9
4  3;Enzo Gabriel;9.2
```

Exemplo de Gravação de Dados

- * Salvando dados de um dicionário no arquivo:

```
import csv
```

```
aluno1 = {"Nro": 1, "Nome": "Pedro Carlos", "Nota": 9.5}
```

```
aluno2 = {"Nro": 2, "Nome": "Maria Clara", "Nota": 9.9}
```

```
aluno3 = {"Nro": 3, "Nome": "Enzo Gabriel", "Nota": 9.2}
```

```
alunos = [aluno1, aluno2, aluno3]
```

```
cabecalho = ['Nro', 'Nome', 'Nota']
```

```
with open("alunos_dict.csv", "w", newline='', encoding="utf-8") as arquivo:  
    csv_writer = csv.DictWriter(arquivo, delimiter="$", fieldnames=cabecalho)  
    csv_writer.writeheader()  
    csv_writer.writerows(alunos)
```

Grava a lista de cabeçalhos

Grava o dicionário

Exercício 1

A partir do arquivo vídeo-games.csv, crie outro arquivo csv com três colunas: ranking, nome do jogo e data de lançamento dos jogos para PC contidos no arquivo de entrada

Inclua os cabeçalhos no arquivo.

Utilize a tabulação (\t) como separador dos dados

Exercício 2

A partir do arquivo vídeo-games.csv, crie outro arquivo csv com três colunas: ranking, nome do jogo e metaspore dos **jogos que foram lançados em maio**.

Inclua os cabeçalhos no arquivo.

Utilize a o caractere @ como separador dos dados

E agora?

Agora é encontrar dados de interesse e usar as suas novas habilidades para criar aplicações com eles.
Alguns exemplos:

Portal Brasileiro de Dados Abertos

10.000+ conjuntos de dados de diversas fontes do governo brasileiro.



Dados RS

1.000+ conjuntos de dados do órgãos do governo do RS (segurança, trânsito, educação, saneamento, etc.)



Kaggle

Plataforma para cientistas de dados com milhares de datasets em muitas categorias (jogos, saúde, educação)



FiveThirtyEight

Dezenas de datasets em temas como esportes, eleições, filmes, entre outros.



Para análises mais complexas e geração de gráficos talvez você queira estudar os módulos

matplotlib

matplotlib.org

pandas

pandas.pydata.org