

# Persistência de Arquivos: CSV, TSV, Planilhas

QXD0099 - Desenvolvimento de Software para Persistência

**Universidade Federal do Ceará - *Campus* Quixadá**

Prof. Francisco Victor da Silva Pinheiro  
victorpinheiro@ufc.br



# Agenda

- Lendo Strings do Teclado e salvando em um arquivo
- CSV - Comma-separated values
- TSV - Tab-separated values
- Planilhas

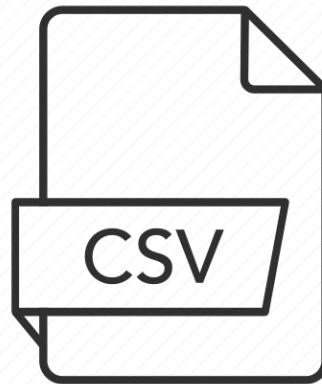
# Lendo Strings do Teclado e salvando em um arquivo

```
# Abrindo o arquivo para escrita
with open('arquivo.txt', 'w', encoding='utf-8') as file:
    # Lendo strings do teclado
    while True:
        try:
            # Lê uma linha do teclado
            line = input()
            # Escreve a linha no arquivo
            print(line, file=file)
        except EOFError:
            # Termina o loop quando não houver mais
            # entrada (Ctrl+D no Linux/macOS ou Ctrl+Z no Windows)
            break
```

- **with open('arquivo.txt', 'w', encoding='utf-8')**: Abre o arquivo `arquivo.txt` em modo de escrita, usando a codificação UTF-8.
- **input()**: Lê uma linha de entrada do teclado.
- **print(line, file=file)**: Escreve a linha no arquivo. Isso é equivalente ao `PrintStream.println()` do Java.
- **EOFError**: Interrompe o loop quando o usuário envia um sinal de fim de entrada (Ctrl+D em Linux/macOS ou Ctrl+Z no Windows).

# CSV - Comma-separated values

- O formato CSV é bastante simples e suportado por quase todas as planilhas eletrônicas e SGDB disponíveis no mercado.
- Cada linha do arquivo representa um registro, e os valores dentro desse registro são separados por vírgulas (ou outro delimitador, como ponto e vírgula, em algumas regiões).
- O formato é amplamente utilizado devido à sua simplicidade e compatibilidade com diversos sistemas e programas, como planilhas e bancos de dados.



# CSV - Comma-separated values

Year	Make	Model	Description	Price
1997	Ford	E350	ac, abs, moon	3000.00
1999	Chevy	Venture "Extended Edition"		4900.00
1999	Chevy	Venture "Extended Edition, Very Large"		5000.00
1996	Jeep	Grand Cherokee	MUST SELL! air, moon roof, loaded	4799.00

Year,Make,Model,Description,Price  
 1997,Ford,E350,"ac, abs, moon",3000.00  
 1999,Chevy,"Venture ""Extended Edition""",,4900.00  
 1999,Chevy,"Venture ""Extended Edition, Very Large""",,5000.00  
 1996,Jeep,Grand Cherokee,"MUST SELL!  
air, moon roof, loaded",4799.00

# Abrindo arquivos CSV

```
import pandas as pd

df = pd.read_csv('veiculos.csv')

df
```

	Year	Make	Model	Description	Price
0	1997	Ford	E350	ac, abs, moon	3000.0
1	1999	Chevy	Venture "Extended Edition"		NaN 4900.0
2	1999	Chevy	Venture "Extended Edition, Very Large"		NaN 5000.0
3	1996	Jeep	Grand Cherokee	MUST SELL! air, moon roof, loaded	4799.0

- **import pandas as pd**: Importa a biblioteca **pandas** e a associa ao alias **pd**, facilitando seu uso.
- **df = pd.read\_csv('veiculos.csv')**: Utiliza a função **pd.read\_csv()** para ler o arquivo **veiculos.csv** e armazena os dados em um DataFrame chamado **df**. Esse DataFrame permite manipular, filtrar, agrupar e processar os dados tabulares de forma eficiente.
- **df**: Essa última linha é usada para visualizar o conteúdo do DataFrame **df**. Em um ambiente de desenvolvimento como Jupyter Notebook, colocar apenas **df** ao final do código exibirá o conteúdo da tabela carregada.

# TSV - Tab-separated values

Sepal length	Sepal width	Petal length	Petal width	Species
5.1	3.5	1.4	0.2	I. setosa
4.9	3.0	1.4	0.2	I. setosa
4.7	3.2	1.3	0.2	I. setosa
4.6	3.1	1.5	0.2	I. setosa
5.0	3.6	1.4	0.2	I. setosa

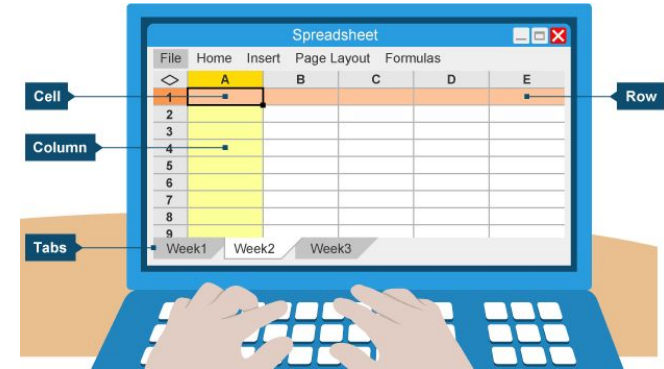
```

Sepal length Sepal width Petal length Petal width Species
5.1 3.5 1.4 0.2 I. setosa
4.9 3.0 1.4 0.2 I. setosa
4.7 3.2 1.3 0.2 I. setosa
4.6 3.1 1.5 0.2 I. setosa
5.0 3.6 1.4 0.2 I. setosa

```

# Planilhas

- Excel
- Google Planilha
- Desenvolvimento de Software Low-Code / No-Code usando Planilhas:
  - AppSheet do Google - <https://www.appsheet.com/>
  - Bubble - <https://bubble.io/>
  - Glide - <https://www.glideapps.com/>





# Abrindo arquivos de planilhas

```
import pandas as pd

# Carrega o arquivo Excel
df = pd.read_excel('nome_do_arquivo.xlsx',
sheet_name='Nome_da_Planilha')

# Exibe os dados carregados
print(df)
```

- **'nome\_do\_arquivo.xlsx'**: O nome ou caminho do arquivo Excel que deseja abrir.
- **sheet\_name**: O nome ou índice da aba (planilha) que deseja carregar. Pode ser uma string com o nome da aba, um número (começando do 0), ou **None** (para carregar todas as planilhas como um dicionário de DataFrames).

	Year	Make	Model	Description	Price
3	1996	Jeep	Grand Cherokee	MUST SELL! air, moon roof, loaded	4799
4	2024	Toyota	Corolla	Red	100000

# Carregar todas as planilhas

```
# Carrega todas as abas como um dicionário de DataFrames
dfs = pd.read_excel('nome_do_arquivo.xlsx', sheet_name=None)

# Exibe as chaves do dicionário (nomes das planilhas)
print(dfs.keys())

# Exibe o DataFrame da primeira aba
print(dfs['Nome_da_Planilha'])
```

- Esse método é especialmente útil para arquivos com múltiplas planilhas que precisam ser manipuladas em conjunto.

	Year	Make	Model	Description	Price
3	1996	Jeep	Grand Cherokee	MUST SELL! air, moon roof, loaded	4799
4	2024	Toyota	Corolla	Red	100000

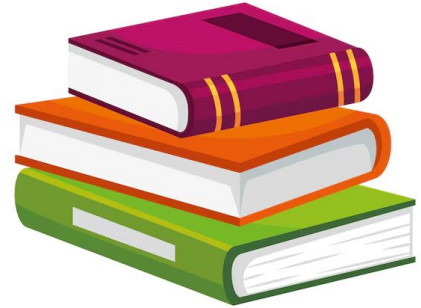
# Bibliografia Básica

- SADALAGE, P. J. E FOWLER, M. NoSQL Essencial. Editora Novatec, São Paulo, 2013.
- REDMOND, E.; WILSON, J. R. Seven Databases in Seven Weeks: A Guide to Modern Databases and the NoSQL Movement. 1ª edição, 2012. The Pragmatic Programmers.
- ULLMAN, J.D.; WIDOW, J. First Course in Database Systems. 3a edição, 2007. Prentice Hall.
- HAMBRICK, G. et al. Persistence in the Enterprise: A Guide to Persistence Technologies; 1ª edição, 2008. IBM Press.
- ELMASRI, R.; NAVATHE, S. B. Sistemas de banco de dados. 4ª edição, 2009. Pearson/Addison-Wesley.



# Bibliografia Complementar

- WHITE, Tom. Hadoop: the definitive guide. California: O'Reilly, 2009. xix, 501 p. ISBN 9780596521974 (broch.).
- AMBLER, S.W., SADALAGE, P.J. Refactoring Databases: Evolutionary Database Design. 1a edição, 2011. Addison Wesley.
- SILBERSCHATZ, A.; SUDARSHAN, S. Sistema de banco de dados. 2006. Campus.
- LYNN, B. Use a cabeça! SQL. 1ª edição, 2008. ALTA BOOKS.
- SMITH, Ben. JSON básico: conheça o formato de dados preferido da web. São Paulo: Novatec, 2015. 400 p. ISBN 9788575224366 (broch.).
- HITZLER, P., KRÖTZSCH, M., and RUDOLPH, S. (2009). Foundations of Semantic Web Technologies. Chapman & Hall/CRC.
- ANTONIOU, G. and HARMELEN, F. (2008). A Semantic Web Primer. Second Edition, Cambridge, MIT Press, Massachusetts.
- HEATH, T. and BIZER, C. (2011). Linked Data: Evolving the Web into a Global Data Space. Morgan & Claypool, 1st edition.



# Obrigado!

## Dúvidas?



**Universidade Federal do Ceará - *Campus* Quixadá**

**Prof. Francisco Victor da Silva Pinheiro**  
victorpinheiro@ufc.br

