



Campus: POLO SAGUAÇU - JOINVILLE - SC

Curso: DESENVOLVIMENTO FULL STACK

Disciplina: Tecnologias Para Desenv. De Solucoes de Big Data

Turma: 9001

Período: 2025.2

Aluno: Jederson Borges de Oliveira

Link: <https://github.com/JedersonBorges/missao-pratica-DGT2823.git>

Microatividades

Microatividade 1: Descrever como ler um arquivo CSV usando a biblioteca Pandas (Python)

```

# Procedimento 1: Salvar o conjunto de dados em formato CSV
csv_conteudo = ""ID;Duration;Date;Pulse;Maxpulse;Calories
0;60;'2020/12/01';110;130;4091
1;60;'2020/12/02';117;145;4790
2;60;'2020/12/03';103;135;3400
3;45;'2020/12/04';109;175;2824
4;45;'2020/12/05';117;148;4060
5;60;'2020/12/06';102;127;3000
6;60;'2020/12/07';110;136;3740
7;450;'2020/12/08';104;134;2533
8;30;'2020/12/09';109;133;1951
9;60;'2020/12/10';98;124;2690
10;60;'2020/12/11';103;147;3293
11;60;'2020/12/12';100;120;2507
12;60;'2020/12/12';100;120;2507
13;60;'2020/12/13';106;128;3453
14;60;'2020/12/14';104;132;3793
15;60;'2020/12/15';98;123;2750
16;60;'2020/12/16';98;120;2152
17;60;'2020/12/17';100;120;3000
18;45;'2020/12/18';90;112;NaN
19;60;'2020/12/19';103;123;3230
20;45;'2020/12/20';97;125;2430
21;60;'2020/12/21';108;131;3642
22;45;NaN;100;119;2820
23;60;'2020/12/23';130;101;3000
24;45;'2020/12/24';105;132;2460
25;60;'2020/12/25';102;126;3345
26;60;'2020/12/26';100;120;2500
27;60;'2020/12/27';92;118;2410
28;60;'2020/12/28';103;132;NaN
29;60;'2020/12/29';100;132;2800
30;60;'2020/12/30';102;129;3803
31;60;'2020/12/31';92;115;2430
"""

with open("pico_web.csv", "w") as arquivo:
    arquivo.write(csv_conteudo)

print("Arquivo 'pico_web.csv' criado com sucesso!\n")

# Procedimento 2:
# - Importar a biblioteca pandas
# - Criar uma variável
# - Ler o conteúdo do arquivo CSV
# - Atribuir os dados lidos do CSV à variável criada anteriormente
# - Salvar as alterações
# - Imprimir/exibir em tela os dados da variável.

import pandas as pd

tabela_exercicios = pd.read_csv("pico_web.csv", sep=';', engine='python', encoding='utf-8')
print("📄 Tabela de exercícios carregada:")
print(tabela_exercicios)

```

Arquivo 'pico_web.csv' criado com sucesso!

Tabela de exercícios carregada:

	ID	Duration	Date	Pulse	Maxpulse	Calories
0	0	60	'2020/12/01'	110	130	4091.0
1	1	60	'2020/12/02'	117	145	4790.0
2	2	60	'2020/12/03'	103	135	3400.0
3	3	45	'2020/12/04'	109	175	2824.0
4	4	45	'2020/12/05'	117	148	4060.0
5	5	60	'2020/12/06'	102	127	3000.0
6	6	60	'2020/12/07'	110	136	3740.0
7	7	450	'2020/12/08'	104	134	2533.0
8	8	30	'2020/12/09'	109	133	1951.0
9	9	60	'2020/12/10'	98	124	2690.0
10	10	60	'2020/12/11'	103	147	3293.0
11	11	60	'2020/12/12'	100	120	2507.0
12	12	60	'2020/12/12'	100	120	2507.0
13	13	60	'2020/12/13'	106	128	3453.0
14	14	60	'2020/12/14'	104	132	3793.0
15	15	60	'2020/12/15'	98	123	2750.0
16	16	60	'2020/12/16'	98	120	2152.0
17	17	60	'2020/12/17'	100	120	3000.0
18	18	45	'2020/12/18'	90	112	NaN
19	19	60	'2020/12/19'	103	123	3230.0
20	20	45	'2020/12/20'	97	125	2430.0
21	21	60	'2020/12/21'	108	131	3642.0
22	22	45	NaN	100	119	2820.0
23	23	60	'2020/12/23'	130	101	3000.0
24	24	45	'2020/12/24'	105	132	2460.0
25	25	60	'2020/12/25'	102	126	3345.0
26	26	60	20201226	100	120	2500.0
27	27	60	'2020/12/27'	92	118	2410.0
28	28	60	'2020/12/28'	103	132	NaN
29	29	60	'2020/12/29'	100	132	2800.0
30	30	60	'2020/12/30'	102	129	3803.0
31	31	60	'2020/12/31'	92	115	2430.0

Microatividade 2: Descrever como ler um arquivo CSV usando a biblioteca Pandas (Python)



```
#####  
# Microatividade 2: Criar subconjunto de dados com 3 colunas  
  
colunas_selecionadas = ['ID', 'Duration', 'Calories']  
subconjunto_exercicios = tabela_exercicios[colunas_selecionadas].copy()  
print("\nSubconjunto de colunas selecionadas:", list(subconjunto_exercicios.columns))  
  
# Salvar o subconjunto em novo CSV  
subconjunto_exercicios.to_csv("pico_web_subset.csv", index=False, sep=';')  
print("\nSubconjunto salvo como 'pico_web_subset.csv'")  
  
# Exibir o subconjunto  
print("\nSubconjunto de dados:")  
print(subconjunto_exercicios)  
print("\nPrimeiras 5 linhas do subconjunto:")  
print(subconjunto_exercicios.head())
```



Arquivo 'pico_web.csv' criado com sucesso!



Arquivo 'pico_web.csv' criado com sucesso!

Subconjunto de colunas selecionadas: ['ID', 'Duration', 'Calories']

Subconjunto salvo como 'pico_web_subset.csv'

Subconjunto de dados:

	ID	Duration	Calories
0	0	60	4091.0
1	1	60	4790.0
2	2	60	3400.0
3	3	45	2824.0
4	4	45	4060.0
5	5	60	3000.0
6	6	60	3740.0
7	7	450	2533.0
8	8	30	1951.0
9	9	60	2690.0
10	10	60	3293.0
11	11	60	2507.0
12	12	60	2507.0
13	13	60	3453.0
14	14	60	3793.0
15	15	60	2750.0
16	16	60	2152.0
17	17	60	3000.0
18	18	45	NaN
19	19	60	3230.0
20	20	45	2430.0
21	21	60	3642.0
22	22	45	2820.0
23	23	60	3000.0
24	24	45	2460.0
25	25	60	3345.0
26	26	60	2500.0
27	27	60	2410.0
28	28	60	NaN
29	29	60	2800.0
30	30	60	3803.0
31	31	60	2430.0

Primeiras 5 linhas do subconjunto:

	ID	Duration	Calories
0	0	60	4091.0
1	1	60	4790.0
2	2	60	3400.0
3	3	45	2824.0
4	4	45	4060.0

Microatividade 3: Descrever como configurar o número máximo de linhas a serem exibidas na visualização de um conjunto de dados usando a biblioteca Pandas (Python)

```
#####  
# Microatividade 3: Descrever como configurar o número máximo de linhas a serem exibidas na visualização de um conjunto  
# de dados usando a biblioteca Pandas (Python)  
  
pd.set_option('display.max_rows', 9999)  
  
# Imprimir na tela o conjunto de dados original  
print("\n📄 Exibindo todas as linhas da tabela de exercícios (Microatividade 1):")  
print(tabela_exercicios.to_string())
```

Arquivo 'pico_web.csv' criado com sucesso!

📄 Exibindo todas as linhas da tabela de exercícios (Microatividade 1):

	ID	Duration	Date	Pulse	Maxpulse	Calories
0	0	60	'2020/12/01'	110	130	4091.0
1	1	60	'2020/12/02'	117	145	4790.0
2	2	60	'2020/12/03'	103	135	3400.0
3	3	45	'2020/12/04'	109	175	2824.0
4	4	45	'2020/12/05'	117	148	4060.0
5	5	60	'2020/12/06'	102	127	3000.0
6	6	60	'2020/12/07'	110	136	3740.0
7	7	450	'2020/12/08'	104	134	2533.0
8	8	30	'2020/12/09'	109	133	1951.0
9	9	60	'2020/12/10'	98	124	2690.0
10	10	60	'2020/12/11'	103	147	3293.0
11	11	60	'2020/12/12'	100	120	2507.0
12	12	60	'2020/12/12'	100	120	2507.0
13	13	60	'2020/12/13'	106	128	3453.0
14	14	60	'2020/12/14'	104	132	3793.0
15	15	60	'2020/12/15'	98	123	2750.0
16	16	60	'2020/12/16'	98	120	2152.0
17	17	60	'2020/12/17'	100	120	3000.0
18	18	45	'2020/12/18'	90	112	NaN
19	19	60	'2020/12/19'	103	123	3230.0
20	20	45	'2020/12/20'	97	125	2430.0
21	21	60	'2020/12/21'	108	131	3642.0
22	22	45	NaN	100	119	2820.0
23	23	60	'2020/12/23'	130	101	3000.0
24	24	45	'2020/12/24'	105	132	2460.0
25	25	60	'2020/12/25'	102	126	3345.0
26	26	60	20201226	100	120	2500.0
27	27	60	'2020/12/27'	92	118	2410.0
28	28	60	'2020/12/28'	103	132	NaN
29	29	60	'2020/12/29'	100	132	2800.0
30	30	60	'2020/12/30'	102	129	3803.0
31	31	60	'2020/12/31'	92	115	2430.0

Microatividade 4: Descrever como exibir as primeiras e últimas “N” linhas de um conjunto de dados usando a biblioteca Pandas (Python)

```
#####
# Microatividade 4: Exibir as primeiras e últimas 10 linhas

# Mostrar as primeiras 10 linhas da tabela original
print("\n📄 Primeiras 10 linhas da tabela de exercícios:")
print(tabela_exercicios.head(10))

# Mostrar as últimas 10 linhas da tabela original
print("\n📄 Últimas 10 linhas da tabela de exercícios:")
print(tabela_exercicios.tail(10))
```

Arquivo 'pico_web.csv' criado com sucesso!

📄 Primeiras 10 linhas da tabela de exercícios:

	ID	Duration	Date	Pulse	Maxpulse	Calories
0	0	60	'2020/12/01'	110	130	4091.0
1	1	60	'2020/12/02'	117	145	4790.0
2	2	60	'2020/12/03'	103	135	3400.0
3	3	45	'2020/12/04'	109	175	2824.0
4	4	45	'2020/12/05'	117	148	4060.0
5	5	60	'2020/12/06'	102	127	3000.0
6	6	60	'2020/12/07'	110	136	3740.0
7	7	450	'2020/12/08'	104	134	2533.0
8	8	30	'2020/12/09'	109	133	1951.0
9	9	60	'2020/12/10'	98	124	2690.0

📄 Últimas 10 linhas da tabela de exercícios:

	ID	Duration	Date	Pulse	Maxpulse	Calories
22	22	45	NaN	100	119	2820.0
23	23	60	'2020/12/23'	130	101	3000.0
24	24	45	'2020/12/24'	105	132	2460.0
25	25	60	'2020/12/25'	102	126	3345.0
26	26	60	20201226	100	120	2500.0
27	27	60	'2020/12/27'	92	118	2410.0
28	28	60	'2020/12/28'	103	132	NaN
29	29	60	'2020/12/29'	100	132	2800.0
30	30	60	'2020/12/30'	102	129	3803.0
31	31	60	'2020/12/31'	92	115	2430.0

Microatividade 5: Descrever como exibir informações gerais sobre as colunas, linhas e dados de um conjunto de dados usando a biblioteca Pandas (Python)

```
#####  
# Microatividade 5: Informações gerais sobre colunas, linhas e dados  
  
# Informações gerais do conjunto de dados  
print("Informações gerais da tabela de exercicios:")  
tabela_exercicios.info()  
  
# Total de linhas  
total_linhas = len(tabela_exercicios)  
print("\n Total de linhas:", total_linhas)  
  
# Total de colunas  
total_colunas = len(tabela_exercicios.columns)  
print("\n Total de colunas:", total_colunas)  
  
# Quantidade de dados nulos por coluna  
print("\n Quantidade de valores nulos por coluna:")  
print(tabela_exercicios.isnull().sum())  
  
# Tipos de dados de cada coluna  
print("\n Tipos de dados das colunas:")  
print(tabela_exercicios.dtypes)  
  
# Memória utilizada pelo conjunto de dados  
print("\n Memória utilizada pelo conjunto de dados:")  
print(tabela_exercicios.memory_usage(deep=True))
```

Arquivo 'pico_web.csv' criado com sucesso!

Informações gerais da tabela de exercícios:

<class 'pandas.core.frame.DataFrame'>

RangeIndex: 32 entries, 0 to 31

Data columns (total 6 columns):

#	Column	Non-Null Count	Dtype
0	ID	32 non-null	int64
1	Duration	32 non-null	int64
2	Date	31 non-null	object
3	Pulse	32 non-null	int64
4	Maxpulse	32 non-null	int64
5	Calories	30 non-null	float64

dtypes: float64(1), int64(4), object(1)

memory usage: 1.6+ KB

Total de linhas: 32

Total de colunas: 6

Quantidade de valores nulos por coluna:

ID	0
Duration	0
Date	1
Pulse	0
Maxpulse	0
Calories	2

dtype: int64

Tipos de dados das colunas:

ID	int64
Duration	int64
Date	object
Pulse	int64
Maxpulse	int64
Calories	float64

dtype: object

Memória utilizada pelo conjunto de dados:

Index	132
ID	256
Duration	256
Date	1919
Pulse	256
Maxpulse	256
Calories	256

dtype: int64