

# UNIVERSIDADE ESTÁCIO CURSO DE DESENVOLVIMENTO FULL STACK

**RELATÓRIO MISSÃO PRATICA 3 - MUNDO 5** 

TURMA 23.1 MOISÉS EDUARDO GOMES DA COSTA CAMPUS:VIA CORPVS - FORTALEZA (CE)

## Missão Prática 3 – Mundo 5

#### Introdução

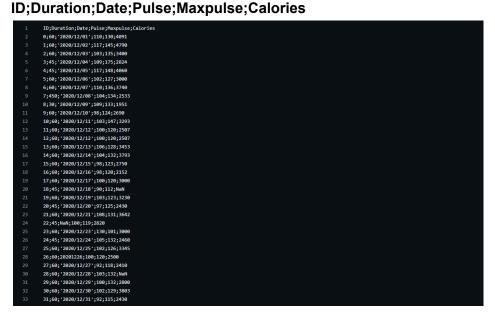
A Missão Prática 3 do Mundo 5 tem como objetivo aplicar conhecimentos de análise de dados usando Python e bibliotecas como Pandas e Matplotlib. O exercício propõe um estudo exploratório de dados fornecidos em um arquivo .csv, com foco em realizar limpeza, agregações, visualizações e interpretações estatísticas dos dados.

### Objetivo

- 1. Descrever como ler um arquivo CSV usando a biblioteca Pandas (Python);
- 2. Descrever como criar um subconjunto de dados a partir de um conjunto existente usando a biblioteca Pandas (Python);
- 3. Descrever como configurar o número máximo de linhas a serem exibidas na visualização de um conjunto de dados usando a biblioteca Pandas (Python);
- Descrever como exibir as primeiras e últimas "N" linhas de um conjunto de dados usando a biblioteca Pandas (Python); Descrever como exibir informações gerais
- 5. sobre as colunas, linhas e dados de um conjunto de dados usando a biblioteca Pandas (Python);

#### Relação com os Requisitos

1.Para essa atividade você deverá, obrigatoriamente, utilizar o conjunto de dados (fornecido anteriormente, na seção "Contextualização") composto pelas colunas



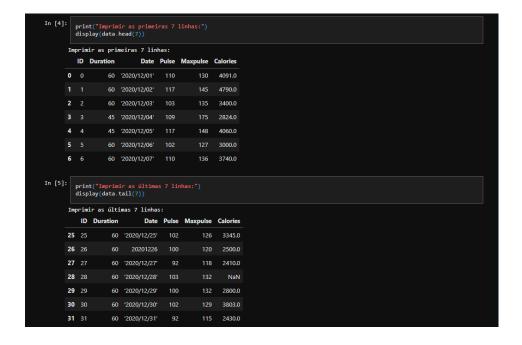
- 2. Crie um novo arquivo/script;
- 3.Leia o conteúdo do CSV fornecido, atentando-se para a necessidade ou não de incluir parâmetros adicionais como os relativos ao separador dos dados, a engine e o enconding;

```
In [1]: import pandas as pd

In [2]: data = pd.read_csv('online_pico.csv', sep=';', engine='python', encoding='utf-8')
```

- 4. Atribua os dados lidos a uma variável;
- 5. Verifique se os dados foram importados adequadamente:
  - a. Imprima as informações gerais sobre o conjunto de dados;

b. Imprima as primeiras e últimas N linhas do arquivo.



6. Crie uma nova variável e atribua a ela uma cópia do conjunto de dados original (variável criada no passo 4);

```
In [6]: dados_copy = data.copy()
```

- 7. Nessa nova variável, contendo uma cópia dos dados:
  - a. Substitua todos os valores nulos da coluna 'Calories' por 0;
  - b. Imprima o conjunto de dados para verificar se a mudança acima foi aplicada com sucesso;

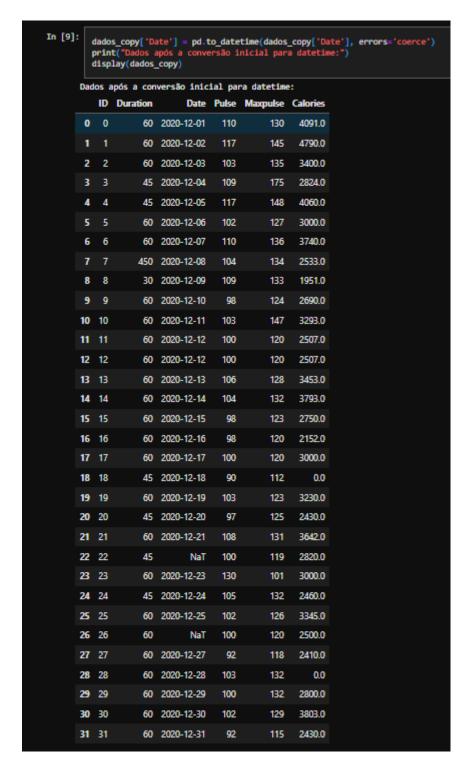
	print		lories'] = 0 de dados ap copy)			
Amo	stra	de dados	após substi	tuir n	ulos em 'C	alories':
	ID	Duration	Date	Pulse	Maxpulse	Calories
0	0	60	'2020/12/01'	110	130	4091.0
1	1	60	'2020/12/02'	117	145	4790.0
2	2	60	"2020/12/03"	103	135	3400.0
3	3	45	"2020/12/04"	109	175	2824.0
4	4	45	"2020/12/05"	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	0.0
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	0.0
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

### 8. Ainda na nova variável:

- a. Substitua os valores nulos da coluna 'Date' por '1900/01/01';
- b. Imprima o conjunto de dados e confira se a mudança foi aplicada com sucesso;

	print		ete'] = dados epós substitu _copy)			
Dad	los a	pós subst	ituir nulos	em 'Dat	te':	
	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	0.0
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	1900/01/01	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	0.0
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

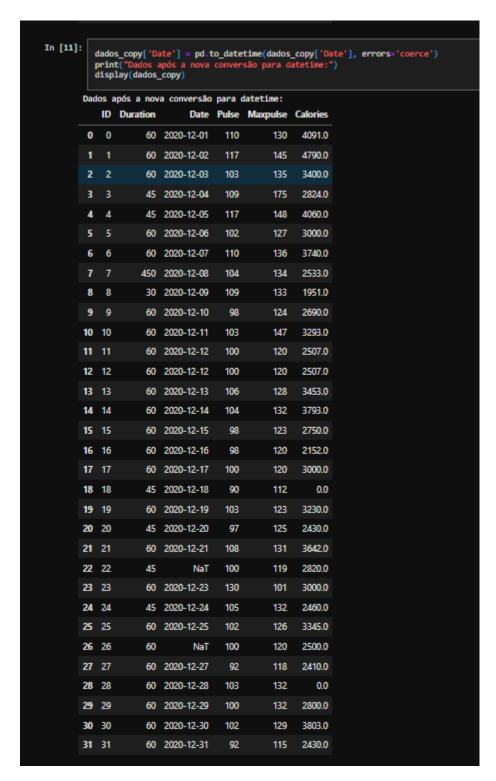
 Transforme os dados da coluna 'Date' em datetime usando o método 'to\_datetime';



- 9. Tendo seguido todas as instruções anteriores, ao executar o passo anterior você deverá ter encontrado um erro informando que o valor '1900/01/01' não corresponde ao formato '%Y/%m/%d'. Para resolver esse problema:
  - a. Substitua, na coluna 'Date', o valor '1900/01/01' por 'NaN' (usei o NaT invés do NaN por se tratarem de dados);

dados_copy['Date'] = dados_copy['Date'].replace('1 print("Dados após substituir '1900/01/01' por NaT: display(dados_copy)  Dados após substituir '1900/01/01' por NaT: ID Duration Date Pulse Maxpulse Calories  0 0 60 2020-12-01 110 130 4091.0
print("Dados após substituir '1900/01/01' por NaT: display(dados_copy)  Dados após substituir '1900/01/01' por NaT:  ID Duration Date Pulse Maxpulse Calories
Dados após substituir '1900/01/01' por NaT:  ID Duration Date Pulse Maxpulse Calories
ID Duration Date Pulse Maxpulse Calories
<u>.</u>
1 1 60 2020-12-02 117 145 4790.0
<b>2</b> 2 60 2020-12-03 103 135 3400.0
3 3 45 2020-12-04 109 175 2824.0
<b>4</b> 4 45 2020-12-05 117 148 4060.0
<b>5</b> 5 60 2020-12-06 102 127 3000.0
6 6 60 2020-12-07 110 136 3740.0
<b>7</b> 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
<b>10</b> 10 60 2020-12-11 103 147 3293.0
<b>11</b> 11 60 2020-12-12 100 120 2507.0
<b>12</b> 12 60 2020-12-12 100 120 2507.0
<b>13</b> 13 60 2020-12-13 106 128 3453.0
<b>14</b> 14 60 2020-12-14 104 132 3793.0
<b>15</b> 15 60 2020-12-15 98 123 2750.0
<b>16</b> 16 60 2020-12-16 98 120 2152.0
<b>17</b> 17 60 2020-12-17 100 120 3000.0
<b>18</b> 18 45 2020-12-18 90 112 0.0
<b>19</b> 19 60 2020-12-19 103 123 3230.0
<b>20</b> 20 45 2020-12-20 97 125 2430.0
<b>21</b> 21 60 2020-12-21 108 131 3642.0
22 22 45 NaT 100 119 2820.0
<b>23</b> 23 60 2020-12-23 130 101 3000.0
<b>24</b> 24 45 2020-12-24 105 132 2460.0
<b>25</b> 25 60 2020-12-25 102 126 3345.0
<b>26</b> 26 60 NaT 100 120 2500.0
<b>27</b> 27 60 2020-12-27 92 118 2410.0
<b>28</b> 28 60 2020-12-28 103 132 0.0
29 29 60 2020-12-29 100 132 2800.0
<b>30</b> 30 60 2020-12-30 102 129 3803.0
<b>31</b> 31 60 2020-12-31 92 115 2430.0

- b. Utilizando o método 'to\_datetime', repita o passo de transformação dos dados da coluna 'Date' para datetime;
- c. Imprima o conjunto de dados para verificar se as mudanças acima foram aplicadas com sucesso;



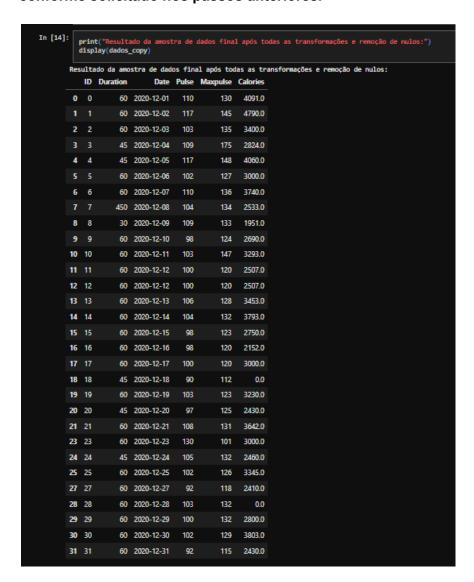
10. Nesse ponto, você deverá ter esbarrado em outro erro, informando agora que o valor "20201226" não corresponde ao formato ""%Y/%m/%d"". Você precisará, agora, na coluna 'Date", transformar especificamente esse valor, atualmente uma string, para o formato datetime. Para isso você deverá combinar os métodos 'replace' e 'to\_datetime';

	dados_ print( displa	copy[ *Da *Dados a y(dados_	ite'] = <b>pd.t</b> ipós corrigi	to_date ir '202	time(dados 01226' e c	_copy[*Da	20201226', '2020/12/26') ste'], errors='coerce') para datetime:")
Dau		ouration	_		Maxpulse 1		cine:
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	0.0	
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	NaT	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			2020-12-25	102	126	3345.0	
	26	60	NaT	100	120	2500.0	
	27		2020-12-27	92	118	2410.0	
	28		2020-12-28	103	132	0.0	
	29		2020-12-29	100	132	2800.0	
	30		2020-12-30	102	129	3803.0	
31	31	60	2020-12-31	92	115	2430.0	

11. Após o passo anterior, execute novamente a transformação de todos os dados da coluna 'Date' para o formato datetime (usando o to\_datetime). Imprima o conjunto de dados atual para verificar se todas as transformações foram executadas com sucesso;

```
In [13]: dados_copy.dropna(subset=['Date'], inplace=True)
```

- 12. Por fim, remova os registros contendo valores nulos. Nesse ponto, apenas a coluna 'Date' possui um registro que atende a essa premissa (linha 22). Logo, utilize-a como base para realizar a transformação solicitada;
- 13. Imprima o dataframe e verifique se todas as transformações foram executadas conforme solicitado nos passos anteriores.



Link do Github: <a href="https://github.com/MoisesE2/Missao3Mundo5">https://github.com/MoisesE2/Missao3Mundo5</a>