



# **Estácio**

**UNIVERSIDADE ESTÁCIO**  
**CURSO DE DESENVOLVIMENTO FULL STACK**

**RELATÓRIO MISSÃO PRÁTICA 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);
4. 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

ID;Duration;Date;Pulse;Maxpulse;Calories

```
1 ID;Duration;Date;Pulse;Maxpulse;Calories
2 0;60;'2020/12/01';110;130;4891
3 1;60;'2020/12/02';117;145;4790
4 2;60;'2020/12/03';103;135;3400
5 3;45;'2020/12/04';109;175;2824
6 4;45;'2020/12/05';117;148;4060
7 5;60;'2020/12/06';102;127;3000
8 6;60;'2020/12/07';110;136;3740
9 7;450;'2020/12/08';104;134;2533
10 8;30;'2020/12/09';109;133;1951
11 9;60;'2020/12/10';98;124;2690
12 10;60;'2020/12/11';103;147;3293
13 11;60;'2020/12/12';100;120;2507
14 12;60;'2020/12/12';100;120;2507
15 13;60;'2020/12/13';106;128;3453
16 14;60;'2020/12/14';104;132;3793
17 15;60;'2020/12/15';98;123;2750
18 16;60;'2020/12/16';98;120;2152
19 17;60;'2020/12/17';100;120;3000
20 18;45;'2020/12/18';90;112;NaN
21 19;60;'2020/12/19';103;123;3230
22 20;45;'2020/12/20';97;125;2430
23 21;60;'2020/12/21';108;131;3642
24 22;45;NaN;100;119;2820
25 23;60;'2020/12/23';130;101;3000
26 24;45;'2020/12/24';105;132;2460
27 25;60;'2020/12/25';102;126;3345
28 26;60;20201226;100;120;2500
29 27;60;'2020/12/27';92;118;2410
30 28;60;'2020/12/28';103;132;NaN
31 29;60;'2020/12/29';100;132;2800
32 30;60;'2020/12/30';102;129;3803
33 31;60;'2020/12/31';92;115;2430
```

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 encoding;

```
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;

```
In [3]: print("Informações gerais sobre o conjunto de dados:")
display(data.info())
```

```
Informações gerais sobre o conjunto de dados:
<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
None
```

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

```
In [4]: print("Imprimir as primeiras 7 linhas:")
display(data.head(7))
```

Imprimir as primeiras 7 linhas:

	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

```
In [5]: print("Imprimir as últimas 7 linhas:")
display(data.tail(7))
```

Imprimir as últimas 7 linhas:

	ID	Duration	Date	Pulse	Maxpulse	Calories
25	25	60	'2020/12/25'	102	126	3345.0
26	26	60	'2020/12/26'	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

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:

- Substitua todos os valores nulos da coluna 'Calories' por 0;
- Imprima o conjunto de dados para verificar se a mudança acima foi aplicada com sucesso;

```
In [7]: dados_copy['Calories'] = dados_copy['Calories'].fillna(0)
print("Amostra de dados após substituir nulos em 'Calories':")
display(dados_copy)
```

Amostra de dados após substituir nulos em 'Calories':

	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:

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

```
In [8]: dados_copy['Date'] = dados_copy['Date'].fillna('1900/01/01')
        print("Dados após substituir nulos em 'Date':")
        display(dados_copy)
```

Dados após substituir nulos em 'Date':

	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	'2020/12/26'	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

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

```
In [9]: dados_copy['Date'] = pd.to_datetime(dados_copy['Date'], errors='coerce')
        print("Dados após a conversão inicial para datetime:")
        display(dados_copy)
```

Dados após a conversão inicial para datetime:

	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	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	25	60	2020-12-25	102	126	3345.0
26	26	60	NaT	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

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:

- Substitua, na coluna 'Date', o valor '1900/01/01' por 'NaN' (usei o NaT invés do NaN por se tratarem de dados);

```
In [18]: dados_copy['Date'] = dados_copy['Date'].replace('1900-01-01', pd.NaT)
         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
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	25	60	2020-12-25	102	126	3345.0
26	26	60	NaT	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

- 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;

```
In [11]: dados_copy['Date'] = pd.to_datetime(dados_copy['Date'], errors='coerce')
         print("Dados após a nova conversão para datetime:")
         display(dados_copy)
```

Dados após a nova conversão para datetime:

	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	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	25	60	2020-12-25	102	126	3345.0
26	26	60	NaT	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



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';

```
In [12]: dados_copy['Date'] = dados_copy['Date'].replace('20201226', '2020/12/26')
dados_copy['Date'] = pd.to_datetime(dados_copy['Date'], errors='coerce')
print("Dados após corrigir '20201226' e converter para datetime:")
display(dados_copy)
```

Dados após corrigir '20201226' e converter para datetime:

	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	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	25	60	2020-12-25	102	126	3345.0
26	26	60	NaT	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

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.

```
In [14]: print("Resultado da amostra de dados final após todas as transformações e remoção de nulos:")  
display(dados_copy)
```

Resultado da amostra de dados final após todas as transformações e remoção de nulos:

	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
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
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

Link do Github: <https://github.com/MoisesE2/Missao3Mundo5>