Universidade Estácio de Sá

- DESENVOLVIMENTO FULL STACK-
- Disciplina: RPG0033 - TRATANDO A IMENSIDÃO DOS DADOS
- Semestre Letivo: 2024.2
- Repositorio Git:https://github.com/Gregdev22/Missao-3-Mundo-5

- EMERSON GREGORIO ALVES - MATRICULA: 2022.0908.4986

Missão Prática | Nível 3 | Mundo 5

TRATANDO A IMENSIDÃO DOS DADOS

📋 Objetivos da Prática

- Descrever como ler um arquivo CSV usando a biblioteca Pandas (Python);
- Descrever como criar um subconjunto de dados a partir de um conjunto existente usando a biblioteca Pandas (Python);
- 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 sobre as colunas, linhas e dados de um conjunto de dados usando a biblioteca Pandas (Python);

Códigos

As atividades foram desenvolvidas no Google Colab.

https://github.com/Gregdev22/Missao-3-Mundo-5/blob/main/MP_NV3.ipynb

```
[1] pip install pandas

[2] import pandas as pd

[3] df = pd.read_csv('/content/drive/MyDrive/MP N3.csv', sep=';')
    display(df)
    df.head(4)
    df.tail(4)
```

```
Out:
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

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

ID      Duration       Date   Pulse   Maxpulse       Calories
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
```

```
[4] df_copia = df.copy()
    df_copia['Calories'] = df_copia['Calories'].fillna(0)
    display(df_copia)
```

Out:

```
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
```

```
[5] df_copia['Date'] = df_copia['Date'].fillna('1900/01/01')
    display(df_copia)
```

Out:

```
        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
```

```
[6] df_copia['Date'] = pd.to_datetime(df_copia['Date'])
    display(df_copia)
```

```
Out:

-----------------------------------------------------------------------
-----
ValueError                                Traceback (most recent call
last)
<ipython-input-27-2d20429ac12b> in <cell line: 1>()
----> 1 df_copia['Date'] = pd.to_datetime(df_copia['Date'])
      2 display(df_copia)

4 frames
/usr/local/lib/python3.10/dist-
packages/pandas/_libs/tslibs/strptime.pyx in
pandas._libs.tslibs.strptime.array_strptime()

ValueError: time data "1900/01/01" doesn't match format "'%Y/%m/%d'",
at position 22. You might want to try:
    - passing `format` if your strings have a consistent format;
    - passing `format='ISO8601'` if your strings are all ISO8601 but
not necessarily in exactly the same format;
    - passing `format='mixed'`, and the format will be inferred for
each element individually. You might want to use `dayfirst` alongside
this.
```

```
[7] df_copia['Date'] = df_copia['Date'].replace('1900/01/01','NaN')
    display(df_copia)
```

```
Out:

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] df_copia['Date'] = pd.to_datetime(df_copia['Date'])
    display(df_copia)
```

Out:

```
-----------------------------------------------------------------------
-----
ValueError                                Traceback (most recent call
last)
<ipython-input-29-2d20429ac12b> in <cell line: 1>()
----> 1 df_copia['Date'] = pd.to_datetime(df_copia['Date'])
      2 display(df_copia)

4 frames
/usr/local/lib/python3.10/dist-
packages/pandas/_libs/tslibs/strptime.pyx in
pandas._libs.tslibs.strptime.array_strptime()

ValueError: time data "20201226" doesn't match format "'%Y/%m/%d'", at
position 26. You might want to try:
    - passing `format` if your strings have a consistent format;
    - passing `format='ISO8601'` if your strings are all ISO8601 but
not necessarily in exactly the same format;
    - passing `format='mixed'`, and the format will be inferred for
each element individually. You might want to use `dayfirst` alongside
this.
```

```
[9] df_copia['Date'] =
df_copia['Date'].replace('20201226',"'2020/12/26'")
    df_copia['Date'] = pd.to_datetime(df_copia['Date'])
    display(df_copia)
```

Out:

```
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      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
```

```
[10] df_copia.dropna(inplace=True)
```

```
[11] display(df_copia)
```

Out:

```
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
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
```

Bade de Dados:

```
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;20201226;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
```