Estácio - Mundo 5 - Missão Nível 3

Faculdade Estácio - Polo Itaipava - Petrópolis/RJ.
Curso: Desenvolvimento Full Stack.
Disciplina: Nível 3: Tratando a imensidão dos dados
Semestre Letivo: 5.

Integrante: Jeferson Jones Smith da Rocha.
Repositório: https://github.com/JefersonSmith/estacio-mundo5-nivel3

Introdução

Este documento detalha o processo de desenvolvimento da missão prática "Tratando a Imensidão dos Dados", parte do curso RPG0033. O objetivo principal foi aplicar técnicas de tratamento e limpeza de dados utilizando a biblioteca Pandas da linguagem Python, preparando um conjunto de dados brutos para análises futuras.

O trabalho foi dividido em duas partes principais: um conjunto de microatividades para familiarização com funções básicas do Pandas e a missão prática, focada na limpeza e transformação de um conjunto de dados específico.

Objetivos

Os objetivos desta prática, conforme definidos no material de orientação, foram:

- Ler arquivos CSV utilizando a biblioteca Pandas.
- Criar subconjuntos de dados a partir de um conjunto existente.
- Configurar opções de visualização de DataFrames no Pandas.
- Exibir linhas específicas (primeiras e últimas) de um conjunto de dados.
- Obter informações gerais sobre um DataFrame (colunas, tipos de dados, valores nulos, uso de memória).
- Realizar a limpeza de dados, incluindo:
  - Tratamento de valores ausentes (NaN).
  - Conversão de tipos de dados (especificamente, strings para datetime).
  - Correção de formatos inconsistentes.
  - Remoção de registros inválidos.

Desenvolvimento:

O desenvolvimento foi realizado em duas etapas principais, correspondentes aos scripts microatividades.py e missao_pratica.py

Microatividades (microatividades.py)

- Leitura do CSV: O arquivo dados_atividade.csv foi lido para um DataFrame Pandas (df_original) utilizando pd.read_csv() , especificando o separador (;), a engine (python) e o caractere de citação (\\') para lidar com as aspas simples nas datas
- Criação de Subconjunto: Um novo DataFrame (df_subconjunto) foi criado contendo apenas as colunas Duration , Pulse e Calories do DataFrame original.
- Configuração de Exibição: A opção pd.options.display.max_rows foi temporariamente configurada para 9999 para permitir a exibição completa do DataFrame original usando o método to_string()

- Exibição de Linhas: Os métodos head(10) e tail(10) foram usados para exibir as 10 primeiras e as 10 últimas linhas do DataFrame original, respectivamente.
- Informações Gerais: O método info() foi chamado no DataFrame original para exibir um resumo, incluindo o número de linhas, colunas, tipos de dados, contagem de valores não nulos e uso de memória

Missão Prática (missao_pratica.py)

Passo 1: Leitura do CSV:

```
[Running] python -u "c:\Users\Smith\Desktop\Estácio\Mundo 5\Atividade 3 - Oficial\missao_pratica.py"
--- Miss�o Pr�tica: Tratamento de Dados ---

Passo 1: Lendo o arquivo CSV
DataFrame Original Lido:
    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
```

Passo 2: Verificação da Importação

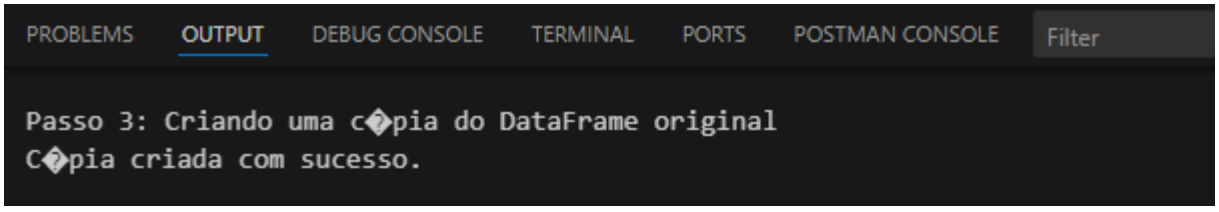
```
Passo 2: Verificando a importa��o dos dados

Informa��es 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

Primeiras 5 linhas do arquivo:
   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

�ltimas 5 linhas do arquivo:
    ID  Duration        Date  Pulse  Maxpulse  Calories
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
```

Passo 3: Criação de Cópia:



Passo 4: Tratamento de Nulos em Calories

```
Passo 4: Substituindo valores nulos da coluna 'Calories' por 0
DataFrame ap�s substitui��o dos valores 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

Passo 5: Tratamento Inicial de Nulos em Date:

```
Passo 5: Substituindo valores nulos da coluna 'Date' por '1900/01/01'
DataFrame ap◆s substitui◆◆o dos valores 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	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

Passo 6: Primeira Tentativa de Conversão de Date:

```
Passo 6: Tentando transformar a coluna 'Date' em datetime
Erro na convers◆o: 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.
```

Passo 7: Reversão e Novo Tratamento de Nulos em Date:

```
Passo 7: Substituindo '1900/01/01' por NaN na coluna 'Date'
DataFrame ap�s substitui��o de '1900/01/01' por NaN:
```

	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

Passo 8: Segunda Tentativa de Conversão de Date:

```
Passo 8: Tentando novamente transformar a coluna 'Date' em datetime
Erro na convers�o: 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.
```

Passo 9: Tratamento de Formato Inconsistente em Date:

```
Passo 9: Transformando o valor '20201226' para o formato datetime
Valor '20201226' transformado com sucesso.
```

Passo 10: Conversão Final de Date:

```
Passo 10: Executando a transforma��o final da coluna 'Date' para datetime
Convers�o final realizada com sucesso.
DataFrame ap�s convers�o final da coluna '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	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

Passo 11: Remoção de Registros com Date Nulo:

```
Passo 11: Removendo registros com valores nulos na coluna 'Date'
DataFrame ap�s remo��o de registros com valores 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
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
```

Passo 12: Verificação final

```
Passo 12: Verifica��o final do DataFrame

Informa��es gerais sobre o DataFrame tratado:
<class 'pandas.core.frame.DataFrame'>
Index: 31 entries, 0 to 31
Data columns (total 6 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   ID        31 non-null     int64
 1   Duration  31 non-null     int64
 2   Date      31 non-null     datetime64[ns]
 3   Pulse     31 non-null     int64
 4   Maxpulse  31 non-null     int64
 5   Calories  31 non-null     float64
dtypes: datetime64[ns](1), float64(1), int64(4)
memory usage: 1.7 KB
```

```
DataFrame tratado final:
    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

Miss�o Pr�tica conclu�da com sucesso!
```

Resultados

Ao final do processo, as microatividades demonstraram o conhecimento básico das operações de leitura, seleção e visualização de dados com **Pandas**. A missão prática resultou em um **DataFrame** (df_tratado no script missao_pratica.py) limpo e pronto para análise:

- Os valores ausentes na coluna **Calories** foram substituídos por **0**.
- A coluna **Date** foi convertida para o tipo **datetime**.
- O formato de data inconsistente ('20201226') foi corrigido.
- O registro que originalmente possuía a data ausente foi removido.
- O DataFrame final contém **31 linhas e 6 colunas**, sem valores nulos nas colunas **ID**, **Pulse**, **Duration**, **Maxpulse**, e com a coluna **Date** e **Calories** tendo os valores nulos originais substituídos por **0**.

Conclusão

A prática permitiu aplicar conceitos fundamentais de tratamento de dados com **Pandas**, abordando problemas comuns como valores ausentes e formatos inconsistentes. O processo seguiu as etapas definidas, resultando em um conjunto de dados tratado e adequado para análises posteriores.