



# Estácio

## **Missão Prática | Nível 3 | Mundo 5**

**TRATANDO A IMENSIDÃO DOS DADOS.**

Aluna: Simone Ramos de Jesus.

Matricula: 202208290965.

Curso EAD: Desenvolvimento Full Stack.

Campus Virtual EAD: Polo Prado – Belo Horizonte – MG.

Ano: 2024.

# Microatividade 1

## Descrever como ler um arquivo CSV usando a biblioteca Pandas (Python)

### - Procedimentos:

1 - Salve o conjunto de dados em formato CSV que utilizará num local acessível pela ferramenta de escrita de código que utilizará;

2 - Crie um novo arquivo e:

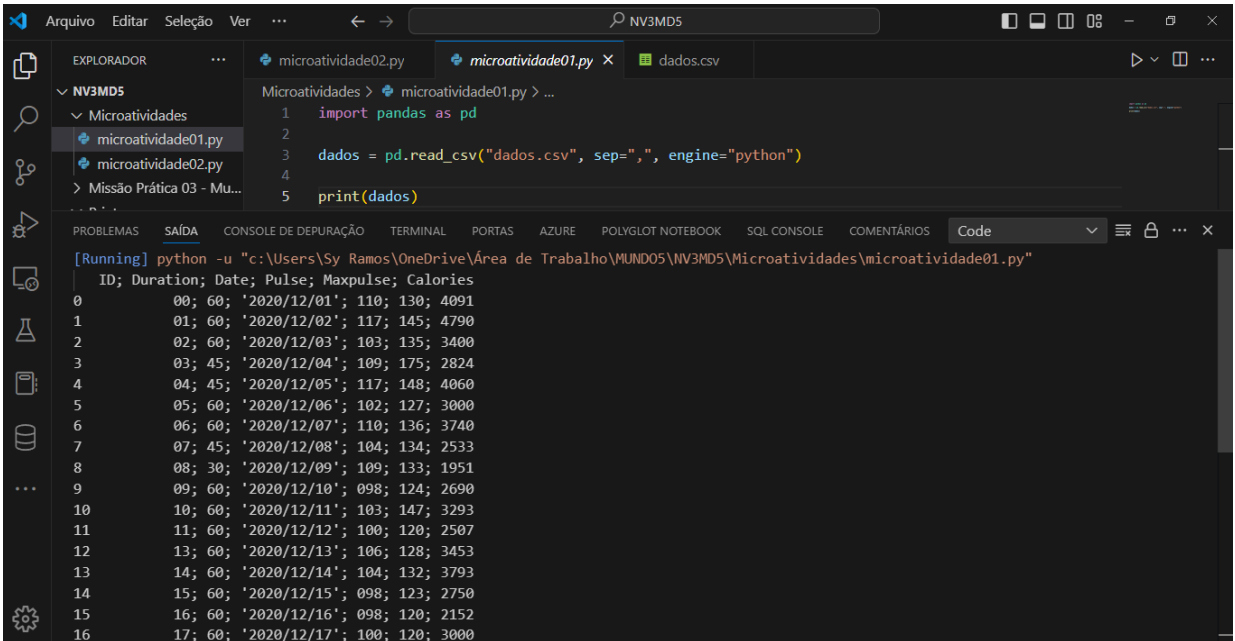
-Importe a biblioteca pandas;

-Cria uma variável;

-Leia o conteúdo do arquivo CSV, passando como parâmetros o separador de colunas, a engine com o valor 'python' e o encoding relativo aos dados constantes no arquivo lido (esse último parâmetro pode ser opcional, dependendo do encoding existente);

-Atribua os dados lidos do CSV à variável criada anteriormente; Salve as alterações;

-Imprima/exiba em tela os dados da variável.



The screenshot shows a code editor with a file explorer on the left and a code editor on the right. The file explorer shows a folder named 'NV3MD5' containing a subfolder 'Microatividades' with files 'microatividade01.py' and 'microatividade02.py'. The code editor shows the content of 'microatividade01.py' with the following code:

```
1 import pandas as pd
2
3 dados = pd.read_csv("dados.csv", sep=";", engine="python")
4
5 print(dados)
```

The output of the code is displayed in the console, showing a table with 18 rows and 6 columns: ID, Duration, Date, Pulse, Maxpulse, and Calories.

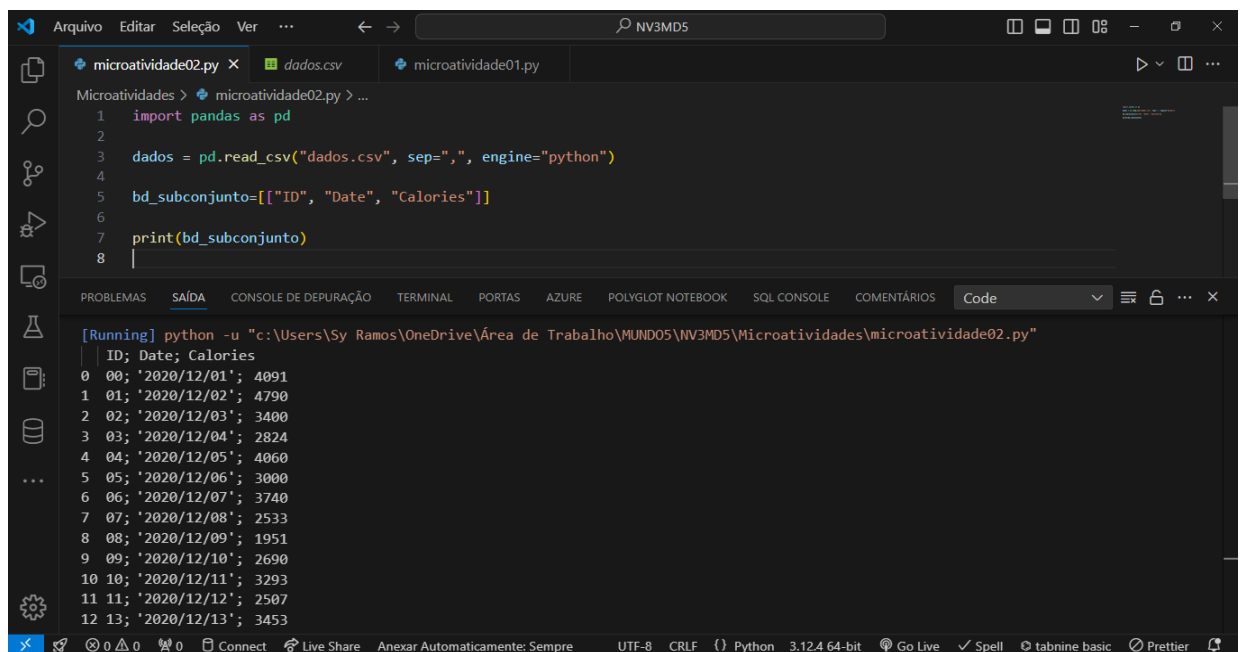
ID	Duration	Date	Pulse	Maxpulse	Calories
00	60	'2020/12/01'	110	130	4091
01	60	'2020/12/02'	117	145	4790
02	60	'2020/12/03'	103	135	3400
03	45	'2020/12/04'	109	175	2824
04	45	'2020/12/05'	117	148	4060
05	60	'2020/12/06'	102	127	3000
06	60	'2020/12/07'	110	136	3740
07	45	'2020/12/08'	104	134	2533
08	30	'2020/12/09'	109	133	1951
09	60	'2020/12/10'	098	124	2690
10	60	'2020/12/11'	103	147	3293
11	60	'2020/12/12'	100	120	2507
12	13	'2020/12/13'	106	128	3453
13	60	'2020/12/14'	104	132	3793
14	15	'2020/12/15'	098	123	2750
15	60	'2020/12/16'	098	120	2152
16	17	'2020/12/17'	100	120	3000

## Microatividade 2

Descrever como criar um subconjunto de dados a partir de um conjunto existente usando a biblioteca Pandas (Python)

### - Procedimentos:

- 1 - No mesmo arquivo/script utilizado na microatividade 1, crie uma nova variável;
  - 2 - Atribua, a essa nova variável, um subconjunto de dados contendo apenas parte das colunas (recomenda-se a utilização de 3 colunas) disponíveis no conjunto de dados original;
  - 3 - Salve as alterações realizadas;
- Imprima/exiba em tela os dados da nova variável (que contém o subconjunto de dados).



The screenshot shows a Visual Studio Code editor window with a Python script named `microatividade02.py`. The script uses the `pandas` library to read a CSV file and create a subset of data. The output of the script is displayed in the terminal window below the editor.

```
1 import pandas as pd
2
3 dados = pd.read_csv("dados.csv", sep=";", engine="python")
4
5 bd_subconjunto=[["ID", "Date", "Calories"]]
6
7 print(bd_subconjunto)
8
```

The terminal output shows the execution of the script, displaying the subset of data as a list of lists:

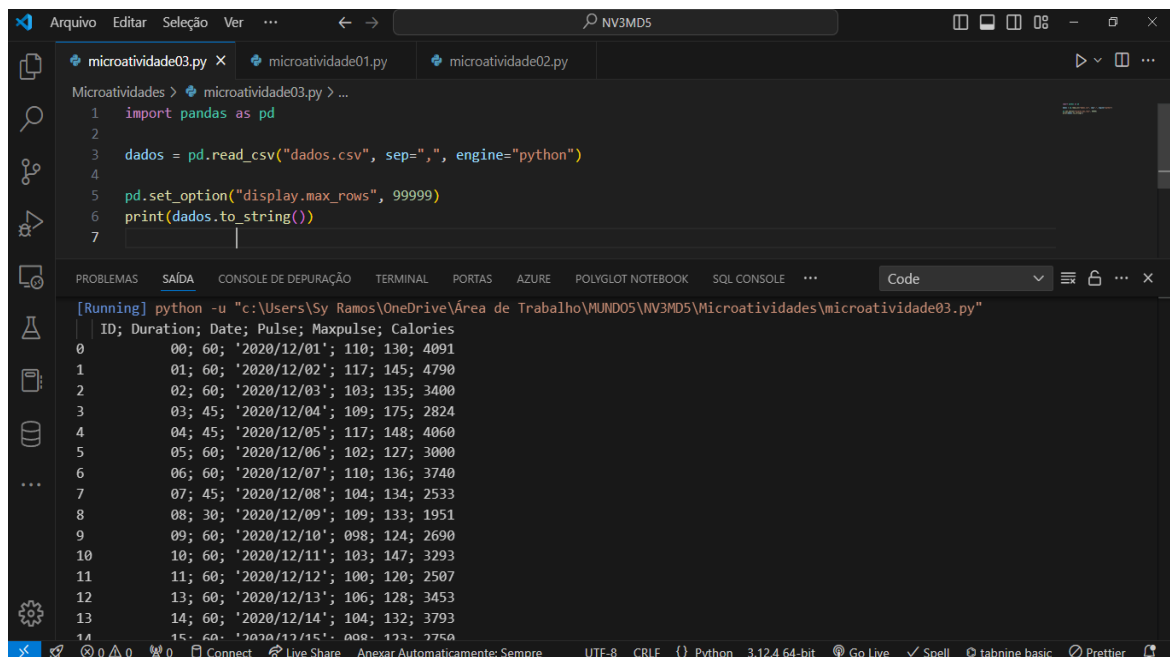
```
[Running] python -u "c:\Users\Sy Ramos\OneDrive\Área de Trabalho\MUNDOS\NV3MD5\Microatividades\microatividade02.py"
ID; Date; Calories
0 00; '2020/12/01'; 4091
1 01; '2020/12/02'; 4790
2 02; '2020/12/03'; 3400
3 03; '2020/12/04'; 2824
4 04; '2020/12/05'; 4060
5 05; '2020/12/06'; 3000
6 06; '2020/12/07'; 3740
7 07; '2020/12/08'; 2533
8 08; '2020/12/09'; 1951
9 09; '2020/12/10'; 2690
10 10; '2020/12/11'; 3293
11 11; '2020/12/12'; 2507
12 13; '2020/12/13'; 3453
```

## 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)

### - Procedimentos:

- 1 - Abra o arquivo/script utilizado nas microatividades anteriores;
- 2 - Usando as opções de configuração da biblioteca pandas, defina um novo valor para a propriedade “max\_rows”, definindo o novo valor para 9999;
- 3 - Salve as alterações;
- 4 - Imprima na tela o conjunto de dados original (criado na microatividade 1) usando o método “to\_string()”.



The screenshot shows a Visual Studio Code editor window with a Python script named `microatividade03.py` open. The script reads a CSV file, sets the display maximum rows to 9999, and prints the data using `to_string()`. The terminal output shows the first 15 rows of the CSV data.

```
1 import pandas as pd
2
3 dados = pd.read_csv("dados.csv", sep=";", engine="python")
4
5 pd.set_option("display.max_rows", 99999)
6 print(dados.to_string())
7
```

Terminal Output:

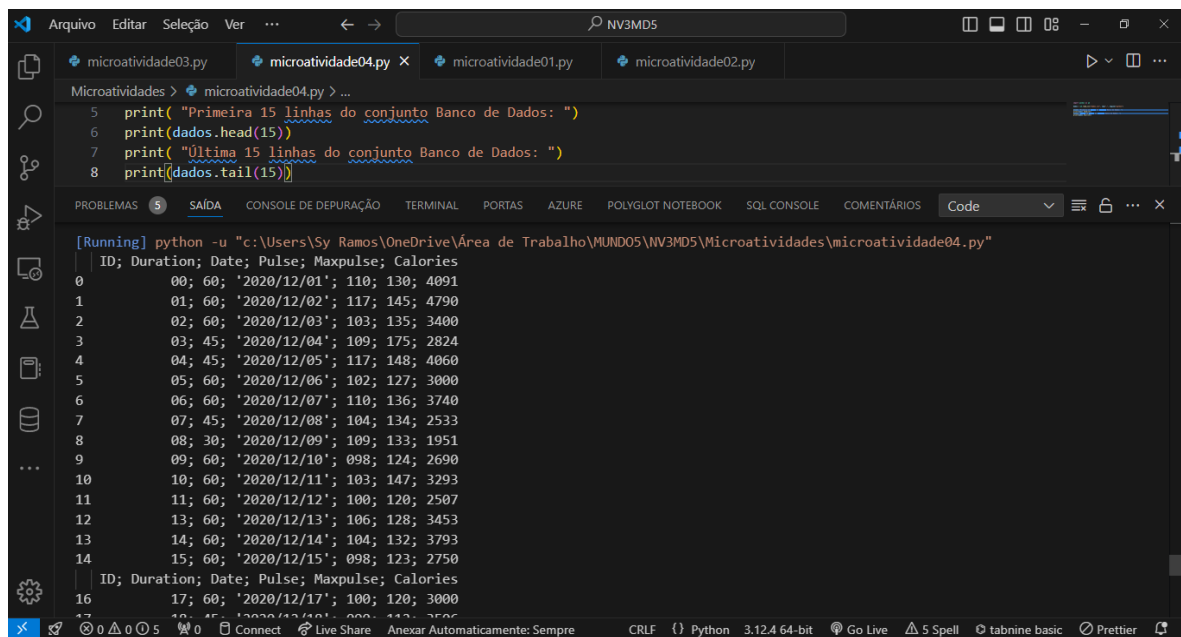
```
[Running] python -u "c:\Users\Sy Ramos\OneDrive\Área de Trabalho\MUNDOS\NV3MD5\Microatividades\microatividade03.py"
ID; Duration; Date; Pulse; Maxpulse; Calories
0 00; 60; '2020/12/01'; 110; 130; 4091
1 01; 60; '2020/12/02'; 117; 145; 4790
2 02; 60; '2020/12/03'; 103; 135; 3400
3 03; 45; '2020/12/04'; 109; 175; 2824
4 04; 45; '2020/12/05'; 117; 148; 4060
5 05; 60; '2020/12/06'; 102; 127; 3000
6 06; 60; '2020/12/07'; 110; 136; 3740
7 07; 45; '2020/12/08'; 104; 134; 2533
8 08; 30; '2020/12/09'; 109; 133; 1951
9 09; 60; '2020/12/10'; 098; 124; 2690
10 10; 60; '2020/12/11'; 103; 147; 3293
11 11; 60; '2020/12/12'; 100; 120; 2507
12 13; 60; '2020/12/13'; 106; 128; 3453
13 14; 60; '2020/12/14'; 104; 132; 3793
14 15; 60; '2020/12/15'; 008; 122; 2750
```

## Microatividade 4

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

### - Procedimentos:

- 1 - Abra o arquivo/script utilizado nas microatividades anteriores;
- 2- Imprima na tela as apenas as primeiras 10 linhas do conjunto de dados original (criado na microatividade 1);
- 3- Imprima na tela as apenas as últimas 10 linhas do conjunto de dados original (criado na microatividade 1).



The screenshot shows a Visual Studio Code editor with a file named `microatividade04.py` open. The script contains the following code:

```
5 print("Primeira 15 linhas do conjunto Banco de Dados: ")
6 print(dados.head(15))
7 print("Última 15 linhas do conjunto Banco de Dados: ")
8 print(dados.tail(15))
```

The terminal output shows the execution of the script, displaying the first 15 rows of a dataset and the last 15 rows. The dataset has columns: ID, Duration, Date, Pulse, Maxpulse, and Calories.

```
[Running] python -u "c:\Users\Sy Ramos\OneDrive\Área de Trabalho\MUNDO5\NV3MD5\Microatividades\microatividade04.py"
ID; Duration; Date; Pulse; Maxpulse; Calories
0      00; 60; '2020/12/01'; 110; 130; 4091
1      01; 60; '2020/12/02'; 117; 145; 4790
2      02; 60; '2020/12/03'; 103; 135; 3400
3      03; 45; '2020/12/04'; 109; 175; 2824
4      04; 45; '2020/12/05'; 117; 148; 4060
5      05; 60; '2020/12/06'; 102; 127; 3000
6      06; 60; '2020/12/07'; 110; 136; 3740
7      07; 45; '2020/12/08'; 104; 134; 2533
8      08; 30; '2020/12/09'; 109; 133; 1951
9      09; 60; '2020/12/10'; 098; 124; 2690
10     10; 60; '2020/12/11'; 103; 147; 3293
11     11; 60; '2020/12/12'; 100; 120; 2507
12     13; 60; '2020/12/13'; 106; 128; 3453
13     14; 60; '2020/12/14'; 104; 132; 3793
14     15; 60; '2020/12/15'; 098; 123; 2750
ID; Duration; Date; Pulse; Maxpulse; Calories
16     17; 60; '2020/12/17'; 100; 120; 3000
17     18; 45; '2020/12/18'; 100; 110; 3000
18     19; 45; '2020/12/19'; 100; 110; 3000
```

## Microatividade 5

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

### - Procedimentos

1 - Abra o arquivo/script utilizado nas microatividades anteriores;

2 - Tendo como base o conjunto de dados original:

a. Imprima as informações gerais sobre o conjunto – suas colunas, linhas e dados;

b. Descubra a partir do comando acima:

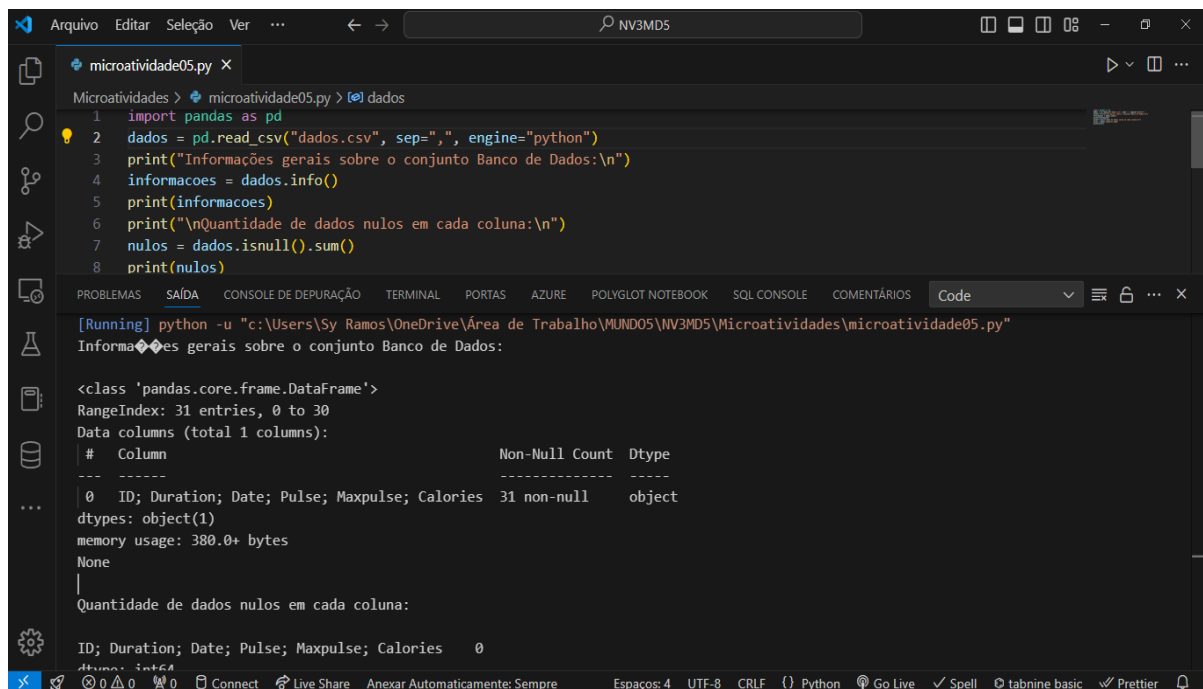
i. total de linhas;

ii. O total de colunas;

iii. A quantidade de dados nulos, caso existam;

iv. O tipo de dado de cada coluna;

v. A quantidade de memória utilizada pelo conjunto de dados.



The screenshot shows a VS Code editor window with a file named `microatividade05.py`. The script uses the `pandas` library to read a CSV file and print general information about the dataset. The terminal output shows the following:

```
[Running] python -u "c:\Users\Sy Ramos\OneDrive\Área de Trabalho\MUND05\NV3MD5\Microatividades\microatividade05.py"
Informações gerais sobre o conjunto Banco de Dados:

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 31 entries, 0 to 30
Data columns (total 1 columns):
#   Column                                     Non-Null Count  Dtype
---  ---                                     -
0   ID; Duration; Date; Pulse; Maxpulse; Calories  31 non-null     object
dtypes: object(1)
memory usage: 380.0+ bytes
None
|
Quantidade de dados nulos em cada coluna:

ID; Duration; Date; Pulse; Maxpulse; Calories    0
dtypes: int64(1)
```

# Missão Prática

## Tratando a imensidão dos dados

### Contextualização

Como Analista de Dados, você recebeu, em um novo projeto, um conjunto de dados. Sua principal tarefa é tratar os dados desse conjunto a fim de que possam ser utilizados para a descoberta de conhecimento através de sua posterior análise e interpretação. Para tal tarefa, você deverá utilizar a linguagem Python e a biblioteca Pandas. O passo-a-passo de todo o processo de tratamento dos dados é apresentado a seguir, no roteiro de prática.

#### - Procedimentos:

- 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
- 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;
- 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);
- 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;

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;
- c.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';
- 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';

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;

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.



```
Arquivo Editar Seleção Ver ... NV3MD5

nv3md5.py X
Missão Prática 03 - Mundo 05 > nv3md5.py > ...
1 import pandas as pd
2 import numpy as np
3
4
5 dados = pd.read_csv("dados.csv", sep=";", engine="python", encoding="utf-8")
6
7 print("Informações detalhada do conjunto Banco de Dados:")
8 print(dados.info())
9 print("\n Primeira 8 linhas, da tabela conjunto Banco de Dados:\n ")
10 print(dados.head())
11 print("\n Última 8 linhas, da tabela conjunto Banco de Dados:\n ")
12 print(dados.tail())
13

[Running] python -u "c:\Users\Sy Ramos\OneDrive\Área de Trabalho\MUNDO5\NV3MD5\Missão Prática 03 - Mundo 05\nv3md5.py"
Informações detalhada do conjunto Banco de Dados:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 31 entries, 0 to 30
Data columns (total 1 columns):
# Column Non-Null Count Dtype
---
0 ID; Duration; Date; Pulse; Maxpulse; Calories 31 non-null object
dtypes: object(1)
memory usage: 380.0+ bytes
None
```

```
Arquivo Editar Seleção Ver ... NV3MD5

nv3md5.py X
Missão Prática 03 - Mundo 05 > nv3md5.py > ...
12 print(dados.tail())
13
14 dados_cod_table = dados.copy()
15
16 dados_cod_table["Calories"].fillna(0, inplace = True)
17 print("\nDados após a substituição dos valores nulos na coluna 'Calories': ")
18 print(dados_cod_table)
19
20 dados_cod_table.fillna("1900/01/01", inplace = True)
21 print("\nDados após a substituição dos valores nulos na coluna 'Date':")
22 print(dados_cod_table)
23
24 dados_cod_table['Date'] = pd.to_datetime(dados_cod_table["Date"], errors='coerce')
25
None
Primeira 8 linhas, da tabela conjunto Banco de Dados:
ID; Duration; Date; Pulse; Maxpulse; Calories
0 00; 60; '2020/12/01'; 110; 130; 4091
1 01; 60; '2020/12/02'; 117; 145; 4790
2 02; 60; '2020/12/03'; 103; 135; 3400
3 03; 45; '2020/12/04'; 109; 175; 2824
4 04; 45; '2020/12/05'; 117; 148; 4060

Última 8 linhas, da tabela conjunto Banco de Dados:
```

```
Arquivo Editar Seleção Ver ... NV3MD5

nv3md5.py X
Missão Prática 03 - Mundo 05 > nv3md5.py > ...
14 dados_cod_table = dados.copy()
15
16 dados_cod_table["Calories"].fillna(0, inplace = True)
17 print("\nDados após a substituição dos valores nulos na coluna 'Calories': ")
18 print(dados_cod_table)
19
20 dados_cod_table.fillna("1900/01/01", inplace = True)
21 print("\nDados após a substituição dos valores nulos na coluna 'Date':")
22 print(dados_cod_table)
23
24 dados_cod_table['Date'] = pd.to_datetime(dados_cod_table["Date"], errors='coerce')
25 dados_cod_table['Date'] = replace(pd.Timestamp('1900-01-01'), np.nan, inplace = True) # type: ignore
26 dados_cod_table['Date'] = pd.Timestamp(dados_cod_table['Date'], errors='coerce')
27
28 print("\nDados após a tentativa de conversão da coluna Date para datetime (com NaNs):\n")
29 print(dados_cod_table)
30
31 dados_cod_table["Date"] = dados_cod_table["Date"].replace("20201226", "2020-12-26")
32 dados_cod_table["Date"] = pd.to_datetime(dados_cod_table["date"], errors='coerce')
33
34 print("\nDados após a conversão final da coluna 'Date'")
35 print(dados_cod_table)
36
37 dados_cod_table.dropna(inplace=True)
38 print("\nDataFrame final após a remoção dos registros com valores nulos")
39 print(dados_cod_table)
```