Missão Prática – Mundo 05 – Nível 03

Gilvan Pereira de Oliveira – 2023.01.53256-61197

Polo Centro – São Lourenço Da Mata – PE

RPG0033 – TRATANDO A IMENSIDÃO DOS DADOS – 9001 – 2025.1

https://github.com/GilvanPOliveira/FullStack/tree/main/Mundo05/tratandoDados

Contextualização

Para resolução das micro atividades será necessário ter em mãos um conjunto de dados no formato CSV. Tais dados podem ser obtidos a partir de fontes gratuitas, disponíveis na Web (como, por exemplo, o dataset disponível em https://archive.ics.uci.edu/dataset/352/online+retail), assim como um ambiente contendo o interpretador da linguagem python.

Ainda em relação aos conjuntos de dados, a atividade “pico web” terá como base o seguinte conjunto de dados (que deverá ser copiado e salvo num arquivo “csv”, usando como separados de colunas o“;”):

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	‘2020/12/26’	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

dados.csv

```
data
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
```

O uso do dataframe acima é imprescindível, uma vez que ele contém dados não válidos que deverão ser tratados posteriormente. Vide as linhas 18 e 28 (coluna Calories); 22 e 26 (coluna Date).

Micro atividade 1: Descrever como ler um arquivo CSV usando a biblioteca Pandas (Python)

Criação do Script de leitura local

leitura_csv.py

```python
import pandas as pd

dadosInformados = './dados.csv'

dados = pd.read_csv(dadosInformados, sep=';', engine='python', encoding='utf-8')

print(dados)
```

- Importar a biblioteca pandas, necessária para a manipulação de dados;
- Definir o arquivo CSV que conterá os dados informados e a sua localização;
- Ler o arquivo CSV utilizando os seguintes parâmetros:
  - sep: define o separador de colunas, que é o ponto e vírgula;
  - engine: define a linguagem utilizada, que é o Python;
  - encoding: define a codificação do arquivo, como ‘uft-8’, que é a mais comum.
- Exibe o conteúdo lido do arquivo CSV, no terminal.

```
[Running] python -u "[redacted]\microAtividade01\leitura_csv.py"
    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

[Done] exited with code=0 in 8.576 seconds
```

Criação do Script de leitura remoto

csv_externo.py

```python
from ucimlrepo import fetch_ucirepo

# fetch dataset
online_retail = fetch_ucirepo(id=352)

# data (as pandas dataframes)
X = online_retail.data.features
y = online_retail.data.targets

# metadata
print(online_retail.metadata)

# variable information
print(online_retail.variables)
```

```
          name     role  ...     units missing_values
0    InvoiceNo       ID  ...      None             no
1    StockCode       ID  ...      None             no
2  Description  Feature  ...      None             no
3     Quantity  Feature  ...      None             no
4  InvoiceDate  Feature  ...      None             no
5    UnitPrice  Feature  ...  sterling             no
6   CustomerID  Feature  ...      None             no
7      Country  Feature  ...      None             no

[8 rows x 7 columns]

[Done] exited with code=0 in 5.484 seconds
```

- Necessário efetuar a instalação do pacote: ucimlrepo;
- Criar um novo arquivo: csv_externo.py;
- O código do exemplo foi fornecido pelo link, fornecido no sway da missao;
    - https://archive.ics.uci.edu/dataset/352/online+retail

No arquivo criado:

- Importar o pacote instalado para localizar os dados desejados;
- Selecionar o id dos dados a serem exibidos;
- Separar os dados de forma ampla e de forma específica;
- Exibir ambas as formas no terminal.

Em cima como foi exibido no VsCode, abaixo como consta no site:

Variables Table

Variable Name	Role	Type	Description	Units	Missing Values
InvoiceNo	ID	Categorical	a 6-digit integral number uniquely assigned to each transaction. If this code starts with letter 'c', it indicates a cancellation		no
StockCode	ID	Categorical	a 5-digit integral number uniquely assigned to each distinct product		no
Description	Feature	Categorical	product name		no
Quantity	Feature	Integer	the quantities of each product (item) per transaction		no
InvoiceDate	Feature	Date	the day and time when each transaction was generated		no
UnitPrice	Feature	Continuous	product price per unit	sterling	no
CustomerID	Feature	Categorical	a 5-digit integral number uniquely assigned to each customer		no
Country	Feature	Categorical	the name of the country where each customer resides		no

Micro atividade 2: Descrever como criar um subconjunto de dados a partir de um conjunto existente usando a biblioteca Pandas (Python)

leitura_csv.py

```python
import pandas as pd

dadosInformados = './dados.csv'

dados = pd.read_csv(dadosInformados, sep=';', engine='python', encoding='utf-8')

subConjunto = dados[['ID', 'Date', 'Calories']]

print(subConjunto)
```

```
[Running] python -u "                                ;\microAtividade02\leitura_csv.py"
    ID         Date  Calories
0    0  '2020/12/01'    4091.0
1    1  '2020/12/02'    4790.0
2    2  '2020/12/03'    3400.0
3    3  '2020/12/04'    2824.0
4    4  '2020/12/05'    4060.0
5    5  '2020/12/06'    3000.0
6    6  '2020/12/07'    3740.0
7    7  '2020/12/08'    2533.0
8    8  '2020/12/09'    1951.0
9    9  '2020/12/10'    2690.0
10  10  '2020/12/11'    3293.0
11  11  '2020/12/12'    2507.0
12  12  '2020/12/12'    2507.0
13  13  '2020/12/13'    3453.0
14  14  '2020/12/14'    3793.0
15  15  '2020/12/15'    2750.0
16  16  '2020/12/16'    2152.0
17  17  '2020/12/17'    3000.0
18  18  '2020/12/18'       NaN
19  19  '2020/12/19'    3230.0
20  20  '2020/12/20'    2430.0
21  21  '2020/12/21'    3642.0
22  22           NaN    2820.0
23  23  '2020/12/23'    3000.0
24  24  '2020/12/24'    2460.0
25  25  '2020/12/25'    3345.0
26  26      20201226    2500.0
27  27  '2020/12/27'    2410.0
28  28  '2020/12/28'       NaN
29  29  '2020/12/29'    2800.0
30  30  '2020/12/30'    3803.0
31  31  '2020/12/31'    2430.0

[Done] exited with code=0 in 1.287 seconds
```

- Utilizando-se o mesmo arquivo da micro atividade anterior;
- Foi criado um subconjunto contendo apenas as colunas 'ID', 'Date' e 'Calories';
- Foi exibido esse subconjunto.

Micro atividade 3: Descrever como configurar o número máximo de linhas a serem exibidas na visualização de um conjunto de dados usando a biblioteca Pandas (Python)

leitura_csv.py

```python
import pandas as pd

dadosInformados = './dados.csv'

dados = pd.read_csv(dadosInformados, sep=';', engine='python', encoding='utf-8')

pd.set_option('display.max_rows', 9999)

print(dados.to_string())
```

```
[Running] python -u "[redacted]\microAtividade03\leitura_csv.py"
    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

[Done] exited with code=0 in 0.592 seconds
```

- Utilizando-se o mesmo arquivo da micro atividade 01;
- Foi solicitado o uso de uma propriedade de exibição do Pandas:
  - display.max_rows definindo seu valor para 9999;
- Porém, para exibir os valores de ‘dados’, foi necessário utilizar o método to_string() para converter o DataFrame em uma string completa.

OBS: Foram exibidos apenas 31 linhas devido a quantidade de linhas que o arquivo ‘dados’ possui, porém o código foi ajustado para exibir até 9999 linhas.

Micro atividade 4: Descrever como exibir as primeiras e últimas “N” linhas de um conjunto de dados usando a biblioteca Pandas (Python)

leitura_csv.py

```python
import pandas as pd

dadosInformados = './dados.csv'

dados = pd.read_csv(dadosInformados, sep=';', engine='python', encoding='utf-8')

print("Primeiras 10 linhas do DataFrame:")
print(dados.head(10))

print("\nUltimas 10 linhas do DataFrame:")
print(dados.tail(10))
```

```
[Running] python -u "[redacted]\microAtividade04\leitura_csv.py"
Primeiras 10 linhas do DataFrame:
   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

Ultimas 10 linhas do DataFrame:
    ID  Duration          Date  Pulse  Maxpulse  Calories
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

[Done] exited with code=0 in 0.684 seconds
```

- Utilizando-se o mesmo arquivo da micro atividade 01;
- Foi solicitado exibir as 10 primeiras e as 10 últimas linhas do arquivo DataFrame: ‘dados’;

Micro atividade 5: Descrever como exibir informações gerais sobre as colunas, linhas e dados de um conjunto de dados usando a biblioteca Pandas (Python)

Utilizando-se o mesmo arquivo da micro atividade 01, foi solicitado:

leitura_csv.py

```python
import pandas as pd

dadosInformados = './dados.csv'

dados = pd.read_csv(dadosInformados, sep=';', engine='python', encoding='utf-8')

print("Informacoes gerais sobre o conjunto de dados:")
dados.info()

total_linhas, total_colunas = dados.shape
print(f"\nTotal de linhas: {total_linhas}")
print(f"Total de colunas: {total_colunas}")

print("\nQuantidade de dados nulos por coluna:")
print(dados.isnull().sum())

print("\nTipo de dado de cada coluna:")
print(dados.dtypes)

print("\nMemoria utilizada pelo conjunto de dados:")
print(dados.memory_usage(deep=True))

```

Exibir informações gerais sobre o conjunto de dados: suas colunas, linhas e dados

```
[Running] python -u "c                                  ;\microAtividade05\leitura_csv.py"
Informacoes 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
```

Total de Linhas e Colunas

```
Total de linhas: 32
Total de colunas: 6
```

Verificar a quantidade de dados nulos, caso existam

```
Quantidade de dados nulos por coluna:
ID          0
Duration    0
Date        1
Pulse       0
Maxpulse    0
Calories    2
dtype: int64
```

O tipo de dado de cada coluna

```
Tipo de dado de cada coluna:
ID            int64
Duration      int64
Date         object
Pulse         int64
Maxpulse      int64
Calories    float64
dtype: object
```

A quantidade de memória utilizada pelo conjunto de dados

```
Memoria utilizada pelo conjunto de dados:
Index        132
ID           256
Duration     256
Date        1919
Pulse        256
Maxpulse     256
Calories     256
dtype: int64

[Done] exited with code=0 in 0.701 seconds
```