



UNIVERSIDADE
Estácio de Sá

Universidade	Estácio de Sá
Campus	Polo de Cobilândia / Vila – Velha/ES
Nome do Curso	Desenvolvimento Full Stack
Nome da Disciplina	DGT2823 – Tecnologias para Desenvolvimento de Soluções de Big Data
Turma	9001
Semestre	Segundo Semestre de 2025
Integrantes do Grupo	Tiago de Jesus Pereira Furtado
Matrícula	202306189045

**VILA VELHA
2025**

Microatividade 1:

1º Descrever como ler um arquivo CSV usando a biblioteca Pandas (Python)

1- Objetivo da Prática:

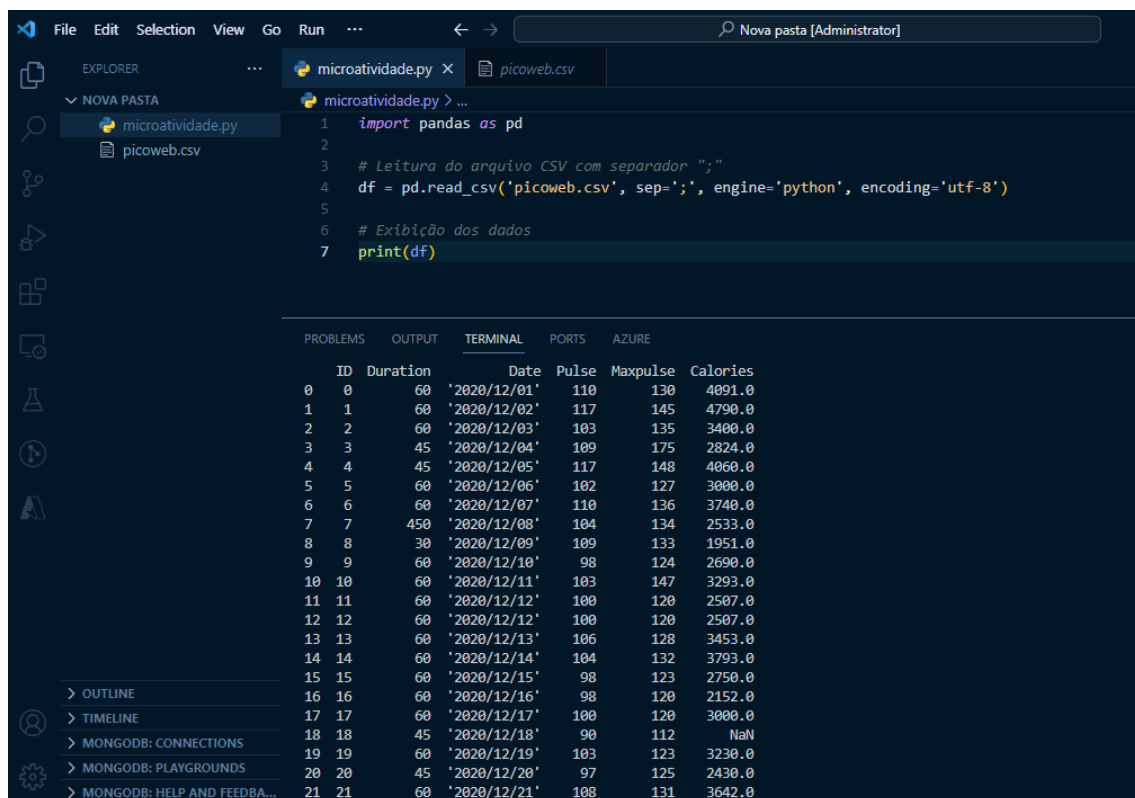
- Descrever como ler um arquivo CSV usando a biblioteca Pandas (Python);
- Descrever como criar um subconjunto de dados a partir de um conjunto existente usando a biblioteca Pandas (Python);
- Descrever como configurar o número máximo de linhas a serem exibidas na visualização de um conjunto de dados 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);

Material necessário para a prática

- Interpretador Python ou ambiente de codificação (JupyterLab / Jupyter Notebooks / Google Colab);
- Biblioteca pandas;
- Editor ou IDE (caso vá utilizar os interpretados python para execução dos scripts criados).

- Procedimentos

1. Salve o conjunto de dados em formato CSV que utilizará num local acessível
2. pela ferramenta de escrita de código que utilizará;
3. Crie um novo arquivo e:
 - a) Importe a biblioteca pandas;
 - b) Cria uma variável;
 - c) 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);
 - d) Atribua os dados lidos do CSV à variável criada anteriormente; salve as alterações;
 - e) Imprima/exiba em tela os dados da variável.



```
File Edit Selection View Go Run ... Nova pasta [Administrator]
EXPLORER
NOVA PASTA
microatividade.py
picoweb.csv
microatividade.py
1 import pandas as pd
2
3 # Leitura do arquivo CSV com separador ";"
4 df = pd.read_csv('picoweb.csv', sep=';', engine='python', encoding='utf-8')
5
6 # Exibição dos dados
7 print(df)
```

	ID	Duration	Date	Pulse	Maxpulse	Calories
0	0	60	'2020/12/01'	110	130	4891.0
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

- Resultados Esperados ✨

O resultado esperado dessa microatividade é verificar se o aluno possui o conhecimento necessário para configurar um ambiente local de desenvolvimento ou utilizar ambientes remotos, além de ser capaz de manusear bibliotecas, como a pandas, e realizar a leitura de dados de uma fonte externa e exibir seu conteúdo.

Microatividade 2:

👉 **2º Descrever como criar um subconjunto de dados a partir de um conjunto existente usando a biblioteca Pandas (Python)**

Objetivo da Prática:

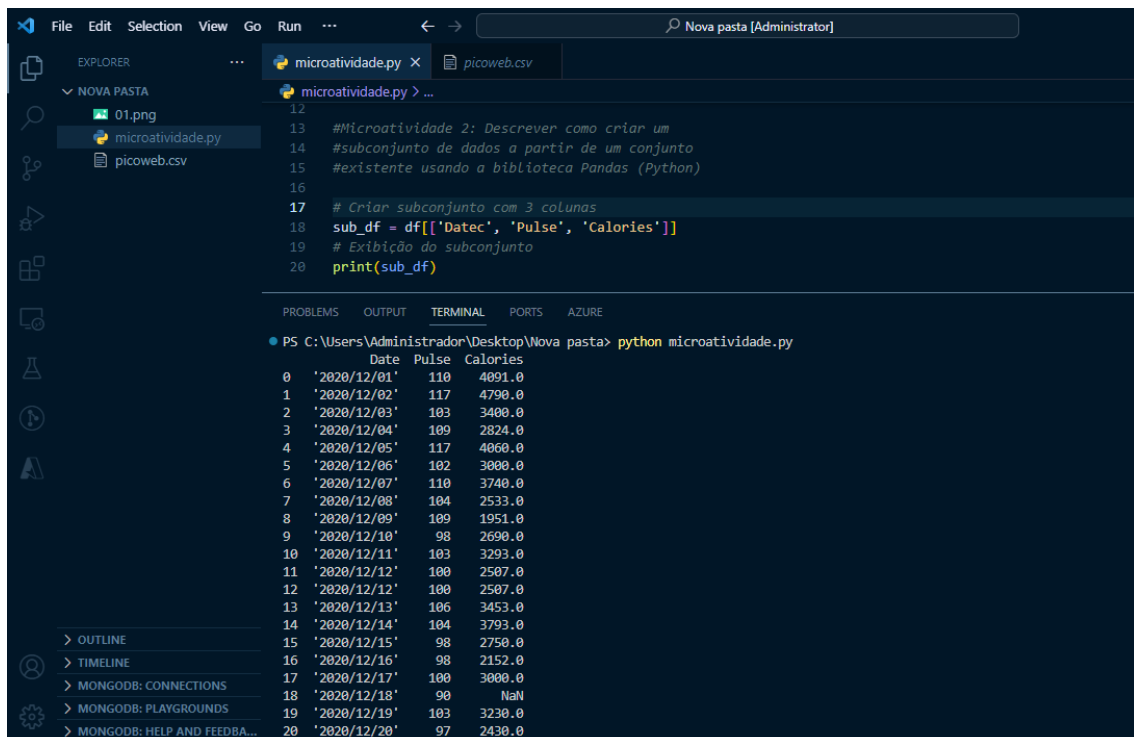
- Descrever como ler um arquivo CSV usando a biblioteca Pandas (Python);
- Descrever como criar um subconjunto de dados a partir de um conjunto existente usando a biblioteca Pandas (Python);
- Descrever como configurar o número máximo de linhas a serem exibidas na visualização de um conjunto de dados 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);

Material necessário para a prática

- Interpretador Python ou ambiente de codificação (JupyterLab / Jupyter Notebooks / Google Colab);
- Biblioteca pandas;
- Editor ou IDE (caso vá utilizar os interpretados python para execução dos scripts criados).

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;
4. Imprima/exiba em tela os dados da nova variável (que contém o subconjunto de dados).



The screenshot shows a VS Code editor window with a file explorer on the left and a terminal at the bottom. The file explorer shows a folder named 'NOVA PASTA' containing files '01.png', 'microatividade.py', and 'picoweb.csv'. The 'microatividade.py' file is open in the editor, showing the following code:

```
12
13 #Microatividade 2: Descrever como criar um
14 #subconjunto de dados a partir de um conjunto
15 #existente usando a biblioteca Pandas (Python)
16
17 # Criar subconjunto com 3 colunas
18 sub_df = df[['Date', 'Pulse', 'Calories']]
19 # Exibição do subconjunto
20 print(sub_df)
```

The terminal at the bottom shows the command `python microatividade.py` being executed, resulting in the following output:

```
PS C:\Users\Administrador\Desktop\Nova pasta> python microatividade.py
```

	Date	Pulse	Calories
0	'2020/12/01'	110	4891.0
1	'2020/12/02'	117	4790.0
2	'2020/12/03'	103	3400.0
3	'2020/12/04'	109	2824.0
4	'2020/12/05'	117	4060.0
5	'2020/12/06'	102	3000.0
6	'2020/12/07'	110	3740.0
7	'2020/12/08'	104	2533.0
8	'2020/12/09'	109	1951.0
9	'2020/12/10'	98	2690.0
10	'2020/12/11'	103	3293.0
11	'2020/12/12'	100	2507.0
12	'2020/12/12'	100	2507.0
13	'2020/12/13'	106	3453.0
14	'2020/12/14'	104	3793.0
15	'2020/12/15'	98	2750.0
16	'2020/12/16'	98	2152.0
17	'2020/12/17'	100	3000.0
18	'2020/12/18'	90	NaN
19	'2020/12/19'	103	3230.0
20	'2020/12/20'	97	2430.0

- Resultados Esperados 🌟

O resultado esperado dessa microatividade é verificar se o aluno possui o conhecimento relativo à manipulação de conjuntos de dados – mais precisamente sobre a criação de subconjuntos a partir de conjuntos pré-existent.

Microatividade 3:

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

Objetivo da Prática:

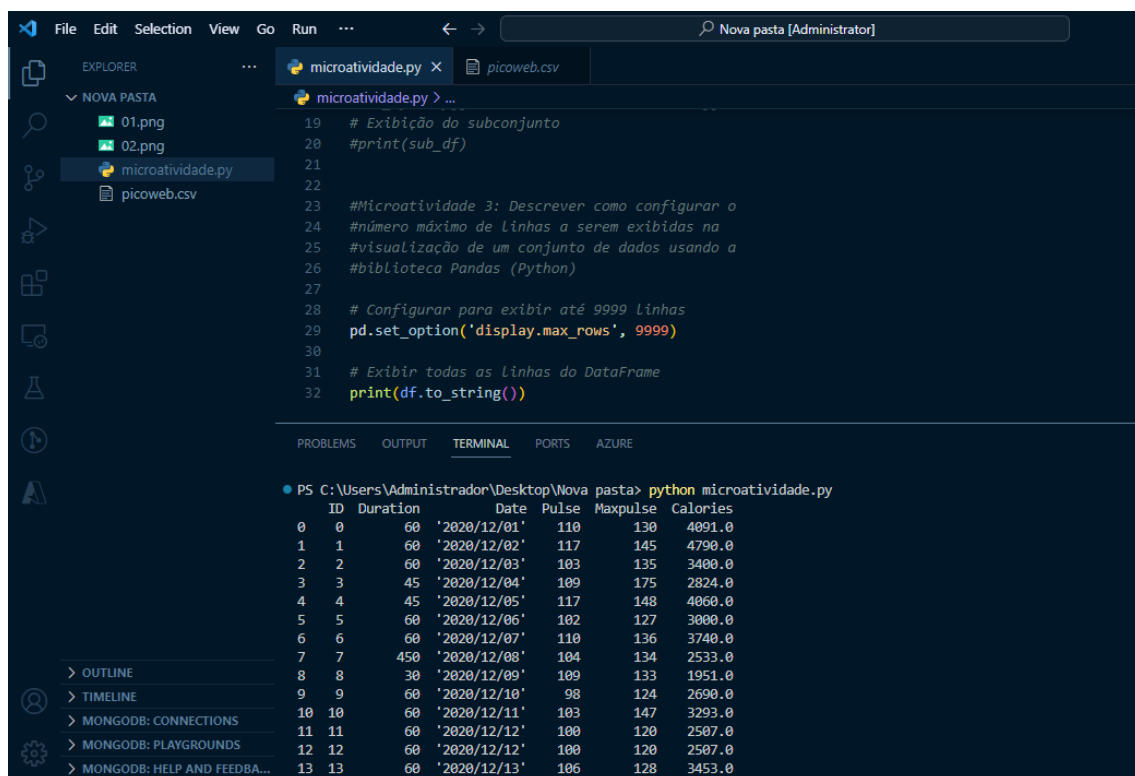
- Descrever como ler um arquivo CSV usando a biblioteca Pandas (Python);
- Descrever como criar um subconjunto de dados a partir de um conjunto existente usando a biblioteca Pandas (Python);
- Descrever como configurar o número máximo de linhas a serem exibidas na visualização de um conjunto de dados 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);

Material necessário para a prática

- Interpretador Python ou ambiente de codificação (JupyterLab / Jupyter Notebooks / Google Colab);
- Biblioteca pandas;
- Editor ou IDE (caso vá utilizar os interpretados python para execução dos scripts criados).

- 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 code editor with a file explorer on the left and a terminal at the bottom. The file explorer shows a folder named 'NOVA PASTA' containing '01.png', '02.png', 'microatividade.py', and 'picoweb.csv'. The code editor displays the contents of 'microatividade.py', which includes comments in Portuguese and Python code using the pandas library. The terminal shows the command 'python microatividade.py' being executed, resulting in a table of data with columns: ID, Duration, Date, Pulse, Maxpulse, and Calories. The data is displayed as a table with 14 rows.

```
19 # Exibição do subconjunto
20 #print(sub_df)
21
22
23 #Microatividade 3: Descrever como configurar o
24 #número máximo de linhas a serem exibidas na
25 #visualização de um conjunto de dados usando a
26 #biblioteca Pandas (Python)
27
28 # Configurar para exibir até 9999 linhas
29 pd.set_option('display.max_rows', 9999)
30
31 # Exibir todas as linhas do DataFrame
32 print(df.to_string())
```

PS C:\Users\Administrador\Desktop\Nova pasta> python microatividade.py

ID	Duration	Date	Pulse	Maxpulse	Calories
0	0	'2020/12/01'	110	130	4091.0
1	1	'2020/12/02'	117	145	4790.0
2	2	'2020/12/03'	103	135	3400.0
3	3	'2020/12/04'	109	175	2824.0
4	4	'2020/12/05'	117	148	4060.0
5	5	'2020/12/06'	102	127	3000.0
6	6	'2020/12/07'	110	136	3740.0
7	7	'2020/12/08'	104	134	2533.0
8	8	'2020/12/09'	109	133	1951.0
9	9	'2020/12/10'	98	124	2690.0
10	10	'2020/12/11'	103	147	3293.0
11	11	'2020/12/12'	100	120	2507.0
12	12	'2020/12/12'	100	120	2507.0
13	13	'2020/12/13'	106	128	3453.0

- Resultados Esperados 🌟

O resultado esperado dessa microatividade é verificar se o aluno possui o conhecimento relativo às opções de configuração da biblioteca Pandas, sendo capaz de manipulá-las.

Microatividade 4:

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

Objetivo da Prática:

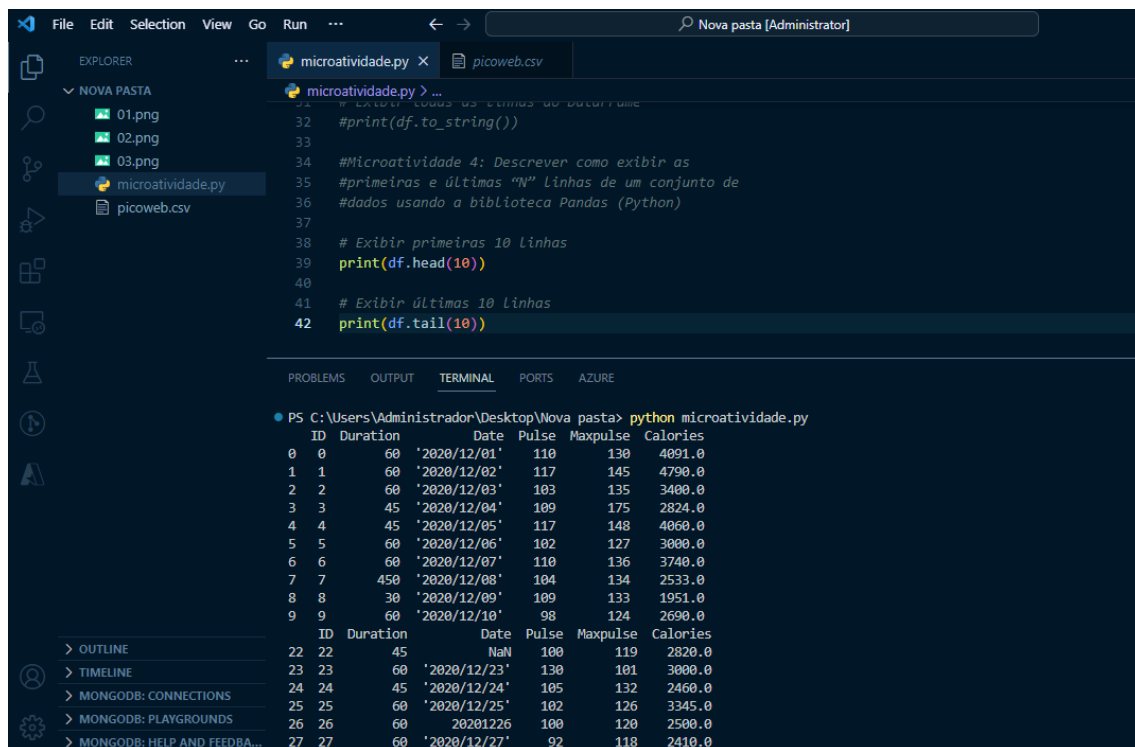
- Descrever como ler um arquivo CSV usando a biblioteca Pandas (Python);
- Descrever como criar um subconjunto de dados a partir de um conjunto existente usando a biblioteca Pandas (Python);
- Descrever como configurar o número máximo de linhas a serem exibidas na visualização de um conjunto de dados 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);

Material necessário para a prática

- Interpretador Python ou ambiente de codificação (JupyterLab / Jupyter Notebooks / Google Colab);
- Biblioteca pandas;
- Editor ou IDE (caso vá utilizar os interpretados python para execução dos scripts criados).

- 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 VS Code editor window titled 'Nova pasta [Administrador]'. The Explorer sidebar on the left shows a folder named 'NOVA PASTA' containing files '01.png', '02.png', '03.png', 'microatividade.py', and 'picoweb.csv'. The main editor area shows the 'microatividade.py' file with the following code:

```
32 #print(df.to_string())
33
34 #Microatividade 4: Descrever como exibir as
35 #primeiras e últimas "N" linhas de um conjunto de
36 #dados usando a biblioteca Pandas (Python)
37
38 # Exibir primeiras 10 Linhas
39 print(df.head(10))
40
41 # Exibir últimas 10 Linhas
42 print(df.tail(10))
```

The TERMINAL panel at the bottom shows the command prompt output after running 'python microatividade.py' in the directory 'C:\Users\Administrador\Desktop\Nova pasta'. The output displays two tables of data. The first table shows the first 10 rows of the dataset, and the second table shows the last 10 rows.


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

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

- Resultados Esperados 🌟

O resultado esperado dessa microatividade é verificar se o aluno possui o conhecimento relativo a alguns dos métodos de visualização de dados disponíveis na biblioteca Pandas.

Microatividade 5:

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

Objetivo da Prática:

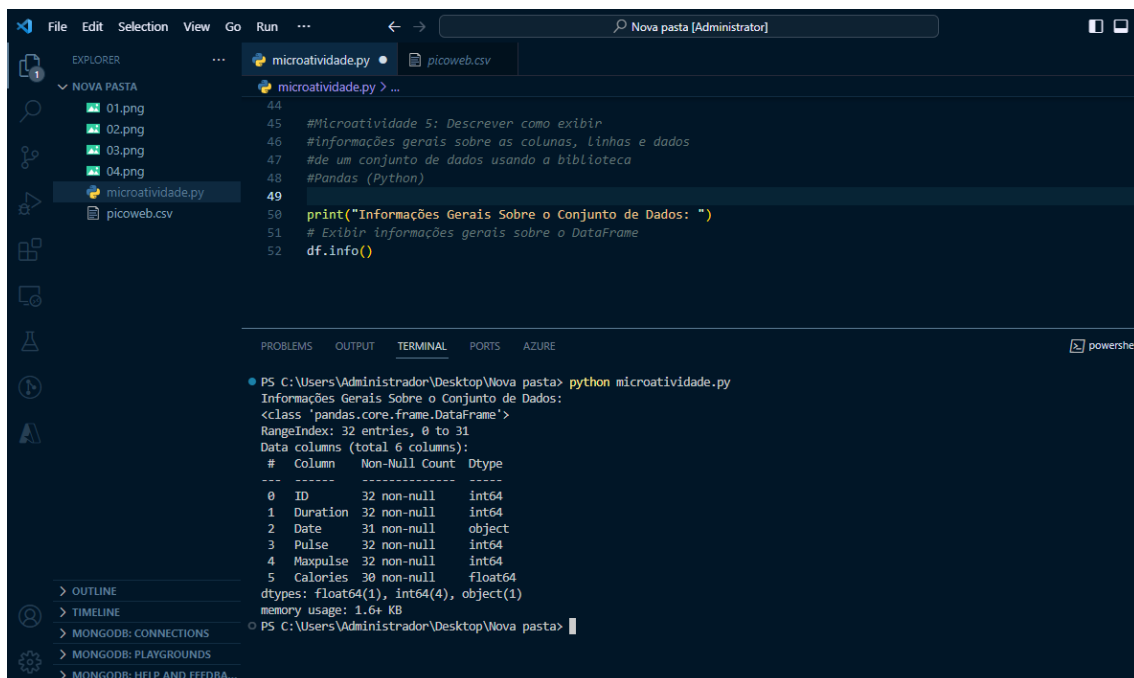
- Descrever como ler um arquivo CSV usando a biblioteca Pandas (Python);
- Descrever como criar um subconjunto de dados a partir de um conjunto existente usando a biblioteca Pandas (Python);
- Descrever como configurar o número máximo de linhas a serem exibidas na visualização de um conjunto de dados 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);

Material necessário para a prática

- Interpretador Python ou ambiente de codificação (JupyterLab / Jupyter Notebooks / Google Colab);
- Biblioteca pandas;
- Editor ou IDE (caso vá utilizar os interpretados python para execução dos scripts criados).

- 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. O 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 Visual Studio Code editor window with a file explorer on the left and a terminal at the bottom. The file explorer shows a folder named 'NOVA PASTA' containing files '01.png', '02.png', '03.png', '04.png', 'microatividade.py', and 'picoweb.csv'. The 'microatividade.py' file is open in the editor, showing the following code:

```
44
45 #Microatividade 5: Descrever como exibir
46 #informações gerais sobre as colunas, Linhas e dados
47 #de um conjunto de dados usando a biblioteca
48 #Pandas (Python)
49
50 print("Informações Gerais Sobre o Conjunto de Dados: ")
51 # Exibir informações gerais sobre o DataFrame
52 df.info()
```

The terminal at the bottom shows the output of running the script:

```
PS C:\Users\Administrador\Desktop\Nova pasta> python microatividade.py
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
PS C:\Users\Administrador\Desktop\Nova pasta>
```

- Resultados esperados 🌟

O resultado esperado dessa microatividade é verificar se o aluno é capaz de extrair informações gerais sobre um conjunto de dados utilizando a biblioteca Pandas

Trabalho Prático

Através dessa atividade o aluno realizará a limpeza de um conjunto de dados, tornando-o apto a ser usado em tarefas de mineração/análise de 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.

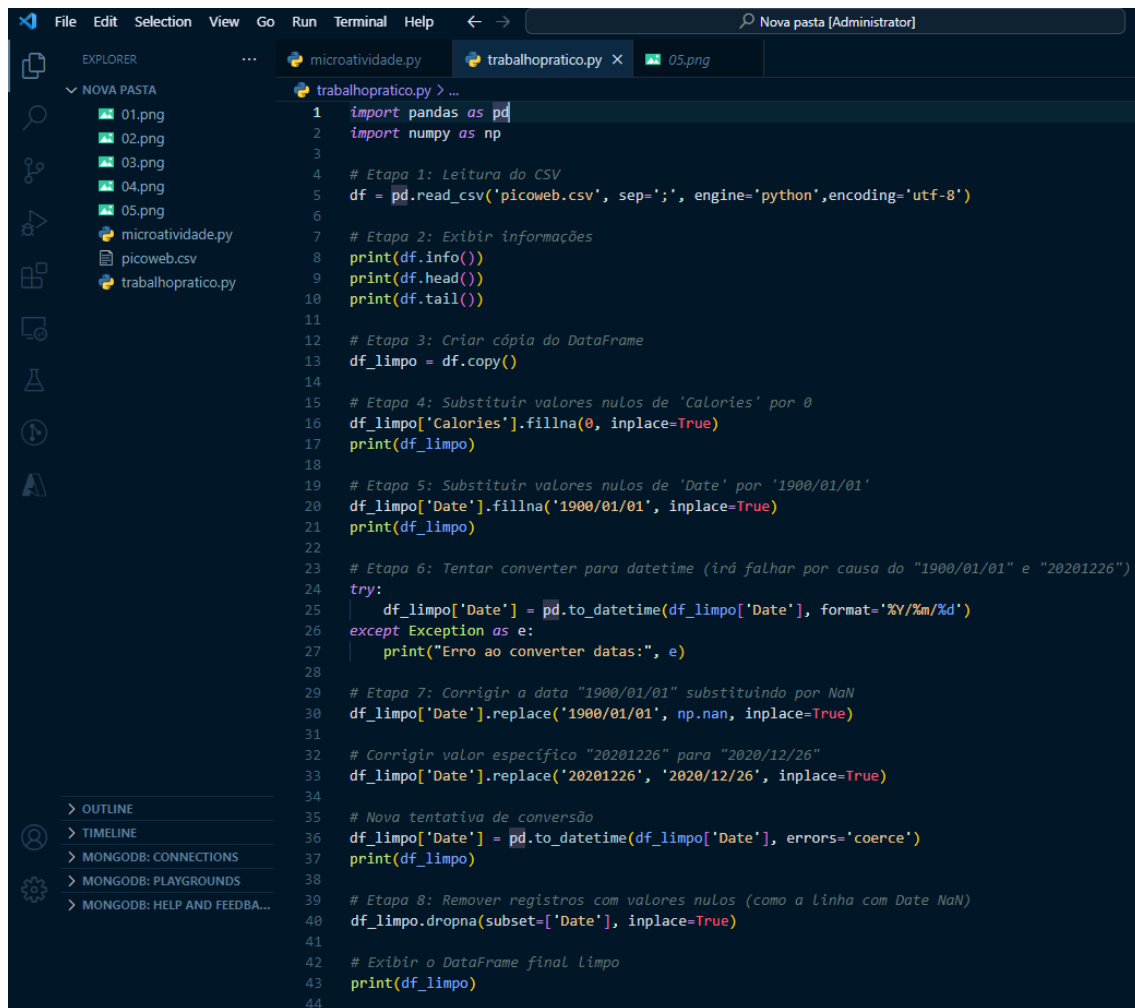
Roteiro de prática

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.

Codigo do Trabalho Pratico:



The image shows a screenshot of a Visual Studio Code editor window. The interface includes a top menu bar with 'File', 'Edit', 'Selection', 'View', 'Go', 'Run', 'Terminal', and 'Help'. Below the menu is a toolbar with icons for Explorer, Search, Source Control, Run and Debug, and Testing. The Explorer sidebar on the left shows a file tree for a 'NOVA PASTA' (New Folder) containing several image files (01.png to 05.png), a 'microatividade.py' file, a 'picoweb.csv' file, and a 'trabalhopratico.py' file. The main editor area displays the 'trabalhopratico.py' file, which contains a Python script for data cleaning using pandas and numpy. The script is divided into eight numbered steps, each with a comment in Portuguese. The script imports pandas as 'pd' and numpy as 'np'. It reads a CSV file 'picoweb.csv' using 'pd.read_csv'. It then performs several data cleaning operations: displaying information, creating a copy of the DataFrame, filling missing values in the 'Calories' column with 0, filling missing values in the 'Date' column with '1900/01/01', attempting to convert the 'Date' column to datetime (with error handling), replacing the placeholder date with NaN, correcting a specific date, and finally dropping rows with missing values in the 'Date' column. The script concludes by printing the final cleaned DataFrame.

```
1 import pandas as pd
2 import numpy as np
3
4 # Etapa 1: Leitura do CSV
5 df = pd.read_csv('picoweb.csv', sep=';', engine='python', encoding='utf-8')
6
7 # Etapa 2: Exibir informações
8 print(df.info())
9 print(df.head())
10 print(df.tail())
11
12 # Etapa 3: Criar cópia do DataFrame
13 df_limpo = df.copy()
14
15 # Etapa 4: Substituir valores nulos de 'Calories' por 0
16 df_limpo['Calories'].fillna(0, inplace=True)
17 print(df_limpo)
18
19 # Etapa 5: Substituir valores nulos de 'Date' por '1900/01/01'
20 df_limpo['Date'].fillna('1900/01/01', inplace=True)
21 print(df_limpo)
22
23 # Etapa 6: Tentar converter para datetime (irá falhar por causa do "1900/01/01" e "20201226")
24 try:
25     df_limpo['Date'] = pd.to_datetime(df_limpo['Date'], format='%Y/%m/%d')
26 except Exception as e:
27     print("Erro ao converter datas:", e)
28
29 # Etapa 7: Corrigir a data "1900/01/01" substituindo por NaN
30 df_limpo['Date'].replace('1900/01/01', np.nan, inplace=True)
31
32 # Corrigir valor específico "20201226" para "2020/12/26"
33 df_limpo['Date'].replace('20201226', '2020/12/26', inplace=True)
34
35 # Nova tentativa de conversão
36 df_limpo['Date'] = pd.to_datetime(df_limpo['Date'], errors='coerce')
37 print(df_limpo)
38
39 # Etapa 8: Remover registros com valores nulos (como a linha com Date NaN)
40 df_limpo.dropna(subset=['Date'], inplace=True)
41
42 # Exibir o DataFrame final limpo
43 print(df_limpo)
44
```

Saida do Codigo do Trabalho Pratico no Terminal:

```
PROBLEMA OUTPUT TERMINAL PORTS ALIAS
PS C:\Users\Administrador\Desktop\Woa pasta> python trabalho pratico.py
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         32 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: 1.5+ KB

None
# Duration      Date      Pulse      Maxpulse      Calories
0  0      00/2000/12/01      110      130      4091.0
1  1      00/2000/12/02      117      145      4790.0
2  2      00/2000/12/03      103      135      3600.0
3  3      00/2000/12/04      109      175      2824.0
4  4      00/2000/12/05      117      148      4860.0
5  5      00/2000/12/06      102      127      3000.0
6  6      00/2000/12/07      110      136      3740.0
7  7      00/2000/12/08      124      154      2531.0
8  8      00/2000/12/09      109      133      1951.0
9  9      00/2000/12/10      90      124      2000.0
10 10     00/2000/12/11      103      147      3201.0
11 11     00/2000/12/12      100      120      2507.0
12 12     00/2000/12/13      106      128      3453.0
13 13     00/2000/12/14      104      132      2793.0
14 14     00/2000/12/15      90      123      2770.0
15 15     00/2000/12/16      90      120      2152.0
16 16     00/2000/12/17      100      130      3000.0
17 17     00/2000/12/18      90      112      0.0
18 18     00/2000/12/19      103      123      3230.0
19 19     00/2000/12/20      97      125      2430.0
20 20     00/2000/12/21      100      131      3642.0
21 21     00/2000/12/22      100      119      2030.0
22 22     00/2000/12/23      130      101      3000.0
23 23     00/2000/12/24      105      132      2400.0
24 24     00/2000/12/25      103      100      1345.0
25 25     00/2000/12/26      100      120      2500.0
26 26     00/2000/12/27      92      118      2410.0
27 27     00/2000/12/28      103      132      0.0
28 28     00/2000/12/29      100      122      2000.0
29 29     00/2000/12/30      102      120      1801.0
30 30     00/2000/12/31      92      115      2430.0

C:\Users\Administrador\Desktop\Woa pasta>trabalho pratico.py:20: FutureWarning: A value is trying to be set on a copy of
a Pandas object that has been modified in place using inplace. This behavior will change in pandas 3.0. This inplace method will never work because the intermediate object on which we s
et the value using an inplace method.
df["Calories"] = df["Calories"].fillna(0, inplace=True)
# Duration      Date      Pulse      Maxpulse      Calories
0  0      00/2000/12/01      110      130      4091.0
1  1      00/2000/12/02      117      145      4790.0
2  2      00/2000/12/03      103      135      3600.0
3  3      00/2000/12/04      109      175      2824.0
4  4      00/2000/12/05      117      148      4860.0
5  5      00/2000/12/06      102      127      3000.0
6  6      00/2000/12/07      110      136      3740.0
7  7      00/2000/12/08      124      154      2531.0
8  8      00/2000/12/09      109      133      1951.0
9  9      00/2000/12/10      90      124      2000.0
10 10     00/2000/12/11      103      147      3201.0
11 11     00/2000/12/12      100      120      2507.0
12 12     00/2000/12/13      106      128      3453.0
13 13     00/2000/12/14      104      132      2793.0
14 14     00/2000/12/15      90      123      2770.0
15 15     00/2000/12/16      90      120      2152.0
16 16     00/2000/12/17      100      130      3000.0
17 17     00/2000/12/18      90      112      0.0
18 18     00/2000/12/19      103      123      3230.0
19 19     00/2000/12/20      97      125      2430.0
20 20     00/2000/12/21      100      131      3642.0
21 21     00/2000/12/22      100      119      2030.0
22 22     00/2000/12/23      130      101      3000.0
23 23     00/2000/12/24      105      132      2400.0
24 24     00/2000/12/25      103      100      1345.0
25 25     00/2000/12/26      100      120      2500.0
26 26     00/2000/12/27      92      118      2410.0
27 27     00/2000/12/28      103      132      0.0
28 28     00/2000/12/29      100      122      2000.0
29 29     00/2000/12/30      102      120      1801.0
30 30     00/2000/12/31      92      115      2430.0

C:\Users\Administrador\Desktop\Woa pasta>trabalho pratico.py:20: FutureWarning: A value is trying to be set on a copy of a DataFrame
object that has been modified in place using inplace. This behavior will change in pandas 3.0. This inplace method will never work because the intermediate object on which we are setting v
alues using an inplace method.
df["Calories"] = df["Calories"].fillna(0, inplace=True)
# Duration      Date      Pulse      Maxpulse      Calories
0  0      00/2000/12/01      110      130      4091.0
1  1      00/2000/12/02      117      145      4790.0
2  2      00/2000/12/03      103      135      3600.0
3  3      00/2000/12/04      109      175      2824.0
4  4      00/2000/12/05      117      148      4860.0
5  5      00/2000/12/06      102      127      3000.0
6  6      00/2000/12/07      110      136      3740.0
7  7      00/2000/12/08      124      154      2531.0
8  8      00/2000/12/09      109      133      1951.0
9  9      00/2000/12/10      90      124      2000.0
10 10     00/2000/12/11      103      147      3201.0
11 11     00/2000/12/12      100      120      2507.0
12 12     00/2000/12/13      106      128      3453.0
13 13     00/2000/12/14      104      132      2793.0
14 14     00/2000/12/15      90      123      2770.0
15 15     00/2000/12/16      90      120      2152.0
16 16     00/2000/12/17      100      130      3000.0
17 17     00/2000/12/18      90      112      0.0
18 18     00/2000/12/19      103      123      3230.0
19 19     00/2000/12/20      97      125      2430.0
20 20     00/2000/12/21      100      131      3642.0
21 21     00/2000/12/22      100      119      2030.0
22 22     00/2000/12/23      130      101      3000.0
23 23     00/2000/12/24      105      132      2400.0
24 24     00/2000/12/25      103      100      1345.0
25 25     00/2000/12/26      100      120      2500.0
26 26     00/2000/12/27      92      118      2410.0
27 27     00/2000/12/28      103      132      0.0
28 28     00/2000/12/29      100      122      2000.0
29 29     00/2000/12/30      102      120      1801.0
30 30     00/2000/12/31      92      115      2430.0
```

- Resultados esperados 🌟

O resultado esperado dessa microatividade é verificar se o aluno possui conhecimentos básicos sobre python – mais precisamente sobre a biblioteca Pandas, sendo capaz de utilizá-la na leitura e manipulação de dados, realizando tarefas como a leitura de arquivos externos, a utilização de dataframes em memória, a exibição de informações e dados, assim como o tratamento/ transformação dos mesmos.