Trabalho Prático | DGT2823 Tecnologias para desenv. de soluções de big data

Material de **orientações** para desenvolvimento do **trabalho prático** da disciplina DGT2823 Tecnologias para desenv. de soluções de big data

📍 **O trabalho prático deve ser feito individualmente.**

DGT2823 - Tecnologias para desenv. de soluções de big data

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);

Entrega

- As microatividades irão dar suporte para o desenvolvimento do Trabalho Prático. Elas têm apoio/gabarito para resolução no próprio documento;
- A entrega esperada é o Trabalho Prático, descrito neste documento após as Microatividades;

👉 Trabalho prático

- Contextualização

Para resolução das microatividades 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. Sobre esse último requisito, podem ser utilizados tanto o interpretados instalado localmente em um computador, quanto soluções como o Jupyter Lab ou o Jupyter Notebooks (que podem ser instaladas localmente) ou o Google Colab, que pode ser usado remotamente. Entre tais opções, e visando agregar ainda mais conhecimento, sugere-se a utilização do JupyterLab instalado localmente (há várias opções, como imagens Docker, entre outras).

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 2

1;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).

Microatividade 1: Descrever como ler um arquivo CSV 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 o interpretados python para execução dos scripts criados).

- 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:
    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 enconding relativo aos dados constantes no arquivo lido (esse último parâmetro pode ser opcional, dependendo do enconding existente);
    d. Atribua os dados lidos do CSV à variável criada anteriormente; Salve as alterações;
    e. Imprima/exiba em tela os dados da variável.

- 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: Descrever como criar um subconjunto de dados a partir de um conjunto existente 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 o 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).

- 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-existentes.

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)

- Material necessário para a prática

- Interpretador Python ou ambiente de codificação (JupyterLab / Jupyter Notebooks / Google Colab);
- Biblioteca pandas;
- Editor ou IDE (caso vá utilizar o 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()”.

- 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: Descrever como exibir as primeiras e últimas “N” linhas 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 o 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).

- 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: 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 o 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.

- 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

- Material necessário para a prática

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

- 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 enconding;
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.

- 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.

📌*Referências*

Não foram utilizadas referências bibliográficas para a elaboração das atividades.

Entrega do trabalho prático

Chegou a hora, gamer!

✍Armazene o projeto em um repositório no GIT (DEIXAR PÚBLICO PARA O TUTOR CONSEGUIR ACESSAR) .

✍ Compartilhe o link do repositório do GIT com o seu tutor para correção da prática, por meio da **Sala de Aula Virtual**, na aba "**Trabalhos**" do respectivo nível de conhecimento.

✍ **Ei, verifique o prazo de entrega deste trabalho pois não recebemos trabalhos fora do prazo!**