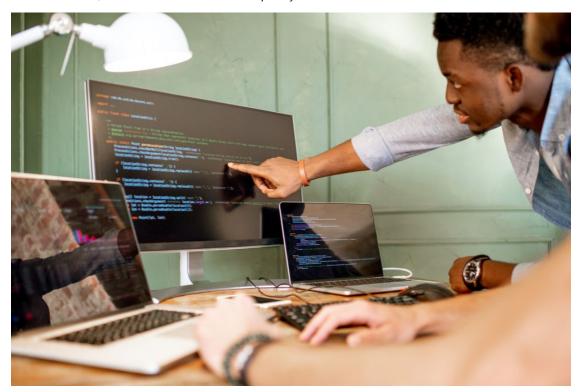
FOCO NO MERCADO DE TRABALHO

INTRODUÇÃO A MANIPULAÇÃO DE DADOS EM PANDAS

Vanessa Cadan Scheffer

TRANSFORMAÇÃO DOS DADOS E EXTRAÇÃO DE INFORMAÇÕES

A biblioteca pandas possui métodos capazes de fazer a leitura dos dados e o carregamento em um DataFrame, além de recursos como a aplicação de filtros.





Fonte: Shutterstock.

Deseja ouvir este material?

Áudio disponível no material digital.

DESAFIO

Como desenvolvedor em uma empresa de consultoria de software, você foi alocado em um projeto para uma empresa de geração de energia. Essa empresa tem interesse em criar uma solução que acompanhe as exportações de etanol no Brasil. Esse tipo de informação está disponível no site do governo brasileiro http://www.dados.gov.br/dataset, em formatos CSV, JSON, dentre outros.

No endereço http://www.dados.gov.br/dataset/importacoes-e-exportacoes-de-etan-ol é possível encontrar várias bases de dados (datasets), contendo informações de importação e exportação de etanol. O cliente está interessado em obter

informações sobre a Exportação Etano Hidratado (barris equivalentes de petróleo) 2012-2020, cujo endereço é http://www.dados.gov.br/dataset/importacoes-e-exportacoes-de-etanol/resource/ca6a2afe-def5-4986-babc-b5e9875d39a5. Para a análise será necessário fazer o download do arquivo.

O cliente deseja uma solução que extraia as seguintes informações:

- Em cada ano, qual o menor e o maior valor arrecadado da exportação?
- Considerando o período de 2012 a 2019, qual a média mensal de arrecadamento com a exportação.
- Considerando o período de 2012 a 2019, qual ano teve o menor arrecadamento? E o menor?

Como parte das informações técnicas sobre o arquivo, foi lhe informado que se trata de um arquivo delimitado CSV, cujo separador de campos é ponto-e-vírgula e a codificação do arquivo está em ISO-8859-1. Como podemos obter o arquivo? Como podemos extrair essas informações usando a linguagem Python? Serão necessários transformações nos dados para obtermos as informações solicitadas?

RESOLUÇÃO

Para começar a resolver o desafio, precisamos fazer o download do arquivo com os dados. Podemos acessar o endereço http://www.dados.gov.br/dataset/importac oes-e-exportacoes-de-etanol/resource/ca6a2afe-def5-4986-babc-b5e9875d39a5 e clicar no botão "ir para recurso" ou então digitar o endereço http://www.anp.gov.br/arquivos/dadosabertos/iee/exportacao-etanol-hidratado-2012-2020-bep.csv que fará o download do arquivo de modo automático. Após obter o arquivo, basta copiá-lo para a pasta do projeto.

Conforme orientações, o arquivo é delimitado, mas seu separador padrão é o ";" e a codificação do arquivo foi feita em ISO-8859-1. Portanto, teremos que passar esses dois parâmetros para a leitura do arquivo usando a biblioteca pandas, uma vez que o delimitar padrão da biblioteca é o ",". No código a seguir, estamos fazendo a importação dos dados. Veja que temos 9 linhas e 8 colunas.

```
In [28]: import pandas as pd

df_etanol = pd.read_csv('exportacao-etanol-hidratado-
2012-2020-bep.csv', sep=';', encoding="ISO-8859-1")

print(df_etanol.info())
df_etanol.head(2)
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9 entries, 0 to 8
Data columns (total 17 columns):
ANO
                       9 non-null int64
PRODUTO
                       9 non-null object
MOVIMENTO COMERCIAL
                       9 non-null object
                       9 non-null object
UNIDADE
JAN
                       9 non-null object
                       9 non-null object
FEV
                        9 non-null object
MAR
ABR
                       9 non-null object
MAI
                        8 non-null object
JUN
                       8 non-null object
JUL
                       8 non-null object
AG0
                       8 non-null object
                        8 non-null object
SET
                       8 non-null object
OUT
                       8 non-null object
NOV
DEZ
                        8 non-null object
TOTAL
                       9 non-null object
dtypes: int64(1), object(16)
memory usage: 1.3+ KB
None
```

Out[28]:

	ANO	PRODUTO	MOVIMENTO COMERCIAL	UNIDADE	JAN	FEV	MAR	ABR	MAI	JUN
0	2012	ETANOL HIDRATADO (bep)	EXPORTACAO	bep	87231,41132	141513,5186	122157,3385	98004,42926	153286,6078	144373,6894
1	2013	ETANOL HIDRATADO (bep)	EXPORTAÇÃO	bep	673419,9767	387331,6487	96929,59201	54390,05046	115092,482	387498,3792

Agora que temos os dados, vamos dividir nossa solução em duas etapas: a de transformação dos dados e a de extração de informações.

ETAPA DE TRANSFORMAÇÕES

Vamos começar removendo as colunas que sabemos que não serão utilizadas, afinal, quanto menos dados na memória RAM, melhor. Veja no código a seguir a remoção de três colunas, com o parâmetro inplace=True, fazendo com que a transformação seja salva no próprio objeto.

339162.21 354343.2858 434799.8585

Agora vamos redefinir os índices do DF, usando a coluna ANO. Esse passo será importante para a fase de extração de informações. Veja que também optamos em remover a coluna do DF (drop=True).

Como os dados são de origem brasileira, a vírgula é usada como separador decimal, o que não condiz com o padrão da biblioteca pandas. Precisamos converter todas as vírgulas em ponto. Para isso vamos utilizar uma estrutura de repetição que filtra cada coluna, criando uma Series, o que nos habilita a utilizar a funcionalidade str.replace(',', '.') para a substituição.

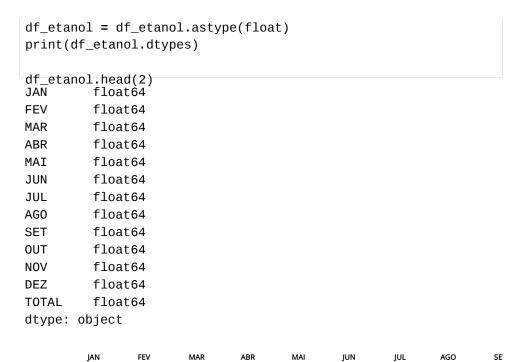
```
for mes in 'JAN FEV MAR ABR MAI JUN JUL AGO SET OUT NOV DEZ
In [31]:
              TOTAL'.split():
                   df_etanol[mes] = df_etanol[mes].str.replace(',', '.')
              print(df_etanol.dtypes)
              df_etanol.head(2)
              JAN
                         object
                         object
              FEV
                         object
              MAR
              ABR
                         object
              MAI
                         object
              JUN
                         object
              JUL
                         object
              AG0
                         object
                         object
              SET
              OUT
                         object
              NOV
                         object
                         object
              DEZ
              TOTAL
                         object
              dtype: object
Out[31]:
                                                                              AGO
               ANO
               2012 87231.41132 141513.5186 122157.3385 98004.42926 153286.6078 144373.6894 384743.6142 244861.0289 702267.5798
```

Mesmo trocando a vírgula por ponto, a biblioteca ainda não conseguiu identificar como ponto flutuante. Portanto, vamos fazer a conversão usando o método astype(float).

2013 673419.9767 387331.6487 96929.59201 54390.05046 115092.482 387498.3792

```
In [32]:
```

Ver anotações



Out[32]:

ANO									
2012	87231.41132	141513.5186	122157.33850	98004.42926	153286.6078	144373.6894	384743.6142	244861.0289	702267.579
2013	673419.97670	387331.6487	96929.59201	54390.05046	115092.4820	387498.3792	339162.2100	354343.2858	434799.858

PESQUISE MAIS

Poderíamos ter usado a biblioteca locale para fazer parte desse trabalho, que tal se aprofundar e pesquisar mais?!

ETAPA DE EXTRAÇÃO DE INFORMAÇÕES

Agora que preparamos os dados, podemos começar a etapa de extração das informações solicitadas. Vamos começar extraindo o menor e maior valor arrecadado em cada ano. Como nosso índice é o próprio ano, podemos usar a função loc para filtrar e então os métodos min() e max(). Para que a extração seja feita para todos os anos, usamos uma estrutura de repetição.

Nas linhas print(f"Menor valor = {minimo:,.0f}".replace(',', '.')) print(f"Maior valor = {maximo:,.0f}".replace(',', '.')) do código a seguir, estamos fazendo a impressão dos valores solicitados. Para que fique mais claro a leitura, formatamos a exibição. O código minimo:,.0f faz com que seja exibida somente a parte inteira e o separador de milhar seja feito por vírgula. Em seguida substituimos a vírgula por ponto que é o padrão brasileiro.

```
Ano = 2012
Menor valor = 87.231
Maior valor = 4.078.157
-----
Ano = 2013
Menor valor = 54.390
Maior valor = 4.168.543
-----
Ano = 2014
Menor valor = 74.303
Maior valor = 2.406.110
-----
Ano = 2015
Menor valor = 31.641
Maior valor = 3.140.140
-----
Ano = 2016
Menor valor = 75.274
Maior valor = 3.394.362
```

Agora, vamos implementar o código para extrair a média mensal, considerando o período de 2012 a 2019. Novamente, podemos usar o loc para filtrar os anos requisitados e, para cada coluna, extrair a média. Na linha 5 fazemos a extração, mas veja que está dentro de uma estrutura de repetição, mês a mês. Na linha 6 fazemos a impressão do resultado, também formatando a saída. Veja que o mês de abril apresenta um rendimento bem inferior aos demais!

```
# Considerando o período de 2012 a 2019, qual a média mensal
In [34]:
            de arrecadamento com a exportação
            print("Média mensal de rendimentos:")
            for mes in 'JAN FEV MAR ABR MAI JUN JUL AGO SET OUT NOV
            DEZ'.split():
                media = df_etanol.loc[2012:2019, mes].mean()
                print(f"{mes} = {media:,.0f}".replace(',', '.'))
            Média mensal de rendimentos:
            JAN = 248.380
            FEV = 210.858
            MAR = 135.155
            ABR = 58.929
DESAFIO DA INTERNETO 6.013
            JUN = 244.645
            JUL = 295.802
Ganhar habilidade en programação exige estudo e treino (muito treino). Acesse o
            SET = 354.454
endereço https://www.kaggle.com/datasets, faço seu cadastro e escolha uma base de
```

Agora precisamos descobrir qual ano teve a menor e a maior quantia em exportação, considerando o período de 2012 a 2019. Para isso vamos usar o

dados para treinar e desenvolver seu conhecimento com a biblioteca pandas.

método idxmin() para descobrir o mínimo e idxmax() para o máximo.

```
In [35]: # Considerando o período de 2012 a 2019, qual ano teve o menor
arrecadamento? E o menor?

ano_menor_arrecadacao = df_etanol.loc[2012:2019,
    'TOTAL'].idxmin()
ano_maior_arrecadacao = df_etanol.loc[2012:2019,
    'TOTAL'].idxmax()

print(f"Ano com menor arrecadação = {ano_menor_arrecadacao}")
print(f"Ano com maior arrecadação = {ano_maior_arrecadacao}")
Ano com menor arrecadação = 2017
Ano com maior arrecadação = 2013
```

Agora é com você, que tal agora organizar as códigos em funções e deixar a solução pronta para ser usada pela equipe?!