

UNIVERSIDADE DO OESTE DE SANTA CATARINA – UNOESC
Engenharia de Computação

JACKSON MICHEL HILLESHEIM FLORENCIO

Trabalho TR1

Valor final da gasolina dos meses de 2020

Chapecó/SC

2022

JACKSON MICHEL HILLESHEIM FLORENCIO

Trabalho TR1: Valor final da gasolina dos meses de 2020

Trabalho de Conclusão de curso apresentado ao
Curso de Estatística e Algoritmos, Área da
Engenharia de Computação, da Universidade do
Oeste de Santa Catarina, Campus de Chapecó.

Orientador: Prof. Marcel Tomas Grando

Chapecó/SC

2022

RESUMO

Neste artigo elaborarei um gráfico na linguagem “Python”, sobre o valor final da gasolina em cada mês de 2020. Explicando para que serve o gráfico, e explicando cada comando aplicado no programa para melhor entendimento.

SUMÁRIO

1 INTRODUÇÃO	5
2 DESENVOLVIMENTO.....	6
3 REFERÊNCIAS.....	9

INTRODUÇÃO

Neste trabalho elaborei um programa em Python utilizando a biblioteca “Matplotlib”, para fazer a geração do gráfico necessário na aplicação desenvolvida.

O tema escolhido foi o valor final da gasolina dos meses de 2020, para termos uma ideia da variação de preços no início da pandemia.

DESENVOLVIMENTO

Primeiramente precisamos fazer a importação da biblioteca no nosso programa utilizando o *import* e nome da biblioteca:

```
# importando biblioteca
import matplotlib.pyplot as grafico
```

Utilizando a o comando “as” para abreviar o nome da biblioteca e facilitar a utilização, ou seja criando um aliás, para toda vez que for utilizar a biblioteca não necessitará ficar digitando o nome da biblioteca, e sim apenas “grafico”.

O próximo passo foi criar a minha lista manualmente:

```
mes = ['Jan', 'Fev', 'Mar', 'Abr', 'Mai', 'Jun', 'Jul', 'Ago', 'Out',
       'Nov', 'Dev']
preco = [4.38, 4.38, 4.19, 4.85, 3.78, 4.09, 4.10, 4.14, 4.21, 4.29, 4.42]
```

Para criar uma lista em python precisamos apenas definir um nome um sinal de atribuição e utilizar colchetes [], a lista mês como não são números preciso defini-los como uma string usando aspas.

Poderia fazer a utilização da biblioteca “pandas” mas como era uma lista pequena de apenas 11 dígitos preferi inclui-los manualmente.

Agora defini o título do gráfico:

```
# definindo titulo do grafico
grafico.title('Valor final da gasolina dos meses de 2020', color='blue')
```

Utilizando o comando *grafico.title* para definir o título que aparecerá assim que o gráfico for gerado, e também definindo uma cor para melhor apresentação, usando o comando *color* um sinal de atribuição e digitando a cor desejável, no meu caso eu utilizei a cor azul, e como é uma string precisa estar em aspas.

O próximo passo é definir o tamanho do eixo X, como o valor mais alto é de 4.85, defino um valor máximo ao eixo X:

```
# definindo o tamanho maximo do eixo X
grafico.axis(xmax=6)
```

Essa função *axis* é utilizada para definir algumas prioridades para os eixos, no meu caso eu defini que o valor máximo do eixo X é 6.

No passo seguinte adicionarei títulos em cada eixo usando o seguinte comando:

```
# adicionando titulos nos eixos X e Y e, definindo a cor
grafico.ylabel('Mês', color='blue')
grafico.xlabel('Preço', color='red')
```

A função `label` é utilizada para colocar textos não editáveis no gráfico, `ylabel` para definir o título no eixo Y como ‘Mês’ e aplicando a cor azul, e `xlabel` para definir o título no eixo X como ‘Preço’, aplicando a cor vermelha.

O passo a seguir será a geração do gráfico:

```
# Gerando Grafico e adicionando legenda
grafico.barh(mes, preco, label='Valor')
grafico.legend()
```

A biblioteca `matplotlib` existem vários tipos de gráficos, porém o escolhido por mim foi o gráfico de barras ou histogramas.

Histograma é um gráfico usado para análise de dados. Esse tipo de apresentação facilita a observação dos valores. É basicamente formado por colunas que indicam a frequência de dados obtidos em uma pesquisa ou estudo.

Para criar o gráfico utilizei o comando `grafico.barh`, o H significa que o gráfico será gerado na posição horizontal em paralelo ao eixo X.

Após isso defini as listas necessárias para a geração do gráfico, (mês e preço, que foram as listas criadas no começo do programa), e usando a função ‘`label`’ para criar uma legenda ‘valor’, e adicionar a legenda ao gráfico utilizando a função `grafico.legend()`.

Agora irei adicionar o valor de cada mês no final do gráfico para ter uma melhor visualização usando o comando de repetição ‘`for`’:

```
# adicionando os valores de cada grafico ao final
for index, value in enumerate(preco):
    grafico.text(value, index,
                 str(value))
```

A função `index` serve para retornar o `index` de um determinado elemento.

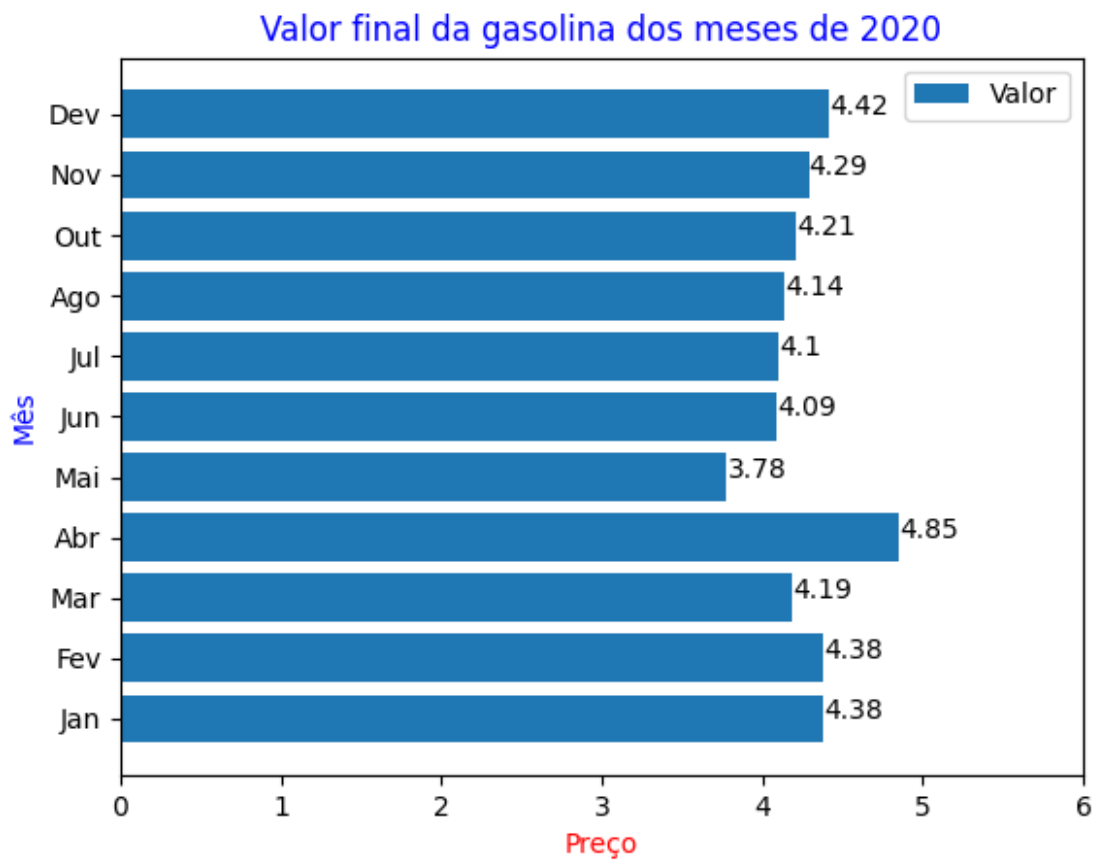
Já a função `enumerate()` pode receber como entrada uma lista e irá retornar um objeto do tipo `enumerate`, que poderá ser percorrido pelo `for`, e selecionando a lista que ele utilizará, no caso a lista ‘preço’.

O próximo passo é colocar o valor retornado pela função acima, usando o `grafico.text`, que pegará os valores retornados e anexará ao final de cada gráfico gerado.

O último passo é garantir que o gráfico será gerado na tela do usuário:

```
grafico.show()
```

Ficando o gráfico da seguinte forma:



REFERÊNCIAS

<https://www.kaggle.com/datasets/matheusfreitag/gas-prices-in-brazil>

