

Lista_04

October 9, 2020

1 Lista de Exercícios 4

*Alguns exercícios foram retirados do livro: [The Python Workbook - Ben Stephenson](#).

```
[1]: import random as rd
```

1.1 QUESTÃO 1:

Crie uma função que inverta um dicionário. A função deve receber como parâmetro de entrada um dicionário e retornar um novo dicionário em que as chaves são valores do dicionário de entrada e os valores são listas de chaves do dicionário de entrada. Por exemplo:

```
dic_entrada={"chave1" : "valor1", "chave2" : "valor2", "chave3" : "valor1"}
```

```
dic_saida={"valor1" : ["chave1", "chave3"], "valor2" : ["chave2"]}
```

```
[ ]:
```

1.2 QUESTÃO 2:

Crie uma função que simule o lançamento de dois dados (de seis lados) N vezes. Conforme seu programa é executado, ele deve contar o número de vezes que ocorre cada total, ou seja, a cada lançamento você deve armazenar a soma dos resultados de ambas as faces obtidas e ir contando quantas vezes esses valores se repetem até o final da simulação. Sua função deverá retornar um dicionário com a frequência de cada resultado e a porcentagem esperada pela teoria da probabilidade para cada total. Ou seja, um dicionário em que as chaves são os resultados e o valores são listas de dois elementos, em que o primeiro elemento é a frequência (em porcentagem) e o segundo, o valor esperado (em porcentagem).

Teste para $N = 1000$.

```
[ ]:
```

1.3 QUESTÃO 3:

Provavelmente você não utilizou um celular básico. Na época em que as mensagens de texto eram enviadas usando o próprio teclado numérico. Vamos fingir que não faz tanto tempo assim, está bem? Até porque eles ainda são comercializados.

Em telefones assim, cada tecla possui várias letras associadas a ela, vários toques de tecla são necessários para a maioria das letras. Pressionar o número uma vez gera a primeira letra da tecla. Pressionar o número 2 ou 3 vezes gera a segunda ou terceira letra da tecla.

Sua tarefa é criar uma função que recebe como parâmetro de entrada uma mensagem (o dicionário para a conversão também pode ser uma entrada) e retorne a sequência de cliques em cada tecla caso a mensagem fosse escrita em um celular básico.

Para isso, você deve contruir um dicionário que mapeie de cada letra ou símbolo para as teclas pressionadas.

CHAVE	SÍMBOLO
1	. , ? ! :
2	A B C
3	D E F
4	G H I
5	J K L
6	M N O
7	P Q R S
8	T U V
9	W X Y Z
0	espaço

Por exemplo, se você passar como entrada “Hello, World!” então, seu programa deve retornar 4433555555666110966677755531111.

OBS: Certifique-se de que seu programa lida com letras maiúsculas e minúsculas. Ignore quaisquer caracteres que não estejam listados na tabela acima (também não estamos levando em consideração a acentuação das letras).

[]:

1.4 QUESTÃO 4:

O arquivo `dados_eleicao.txt` contém os dados de uma eleição, com as informações sobre número do eleitor (**NumEleitor**), bairro (**Bairro**) e código do candidato (**CodCandidato**).

Sua tarefa é criar uma função que leia esse arquivo e:

- calcule o total de votos e a porcentagem do total por **CodCandidato**;
- calcule o total de votos e a porcentagem do total por **Bairro**;
- calcule o total de votos e a porcentagem do total por **CodCandidato+Bairro**.

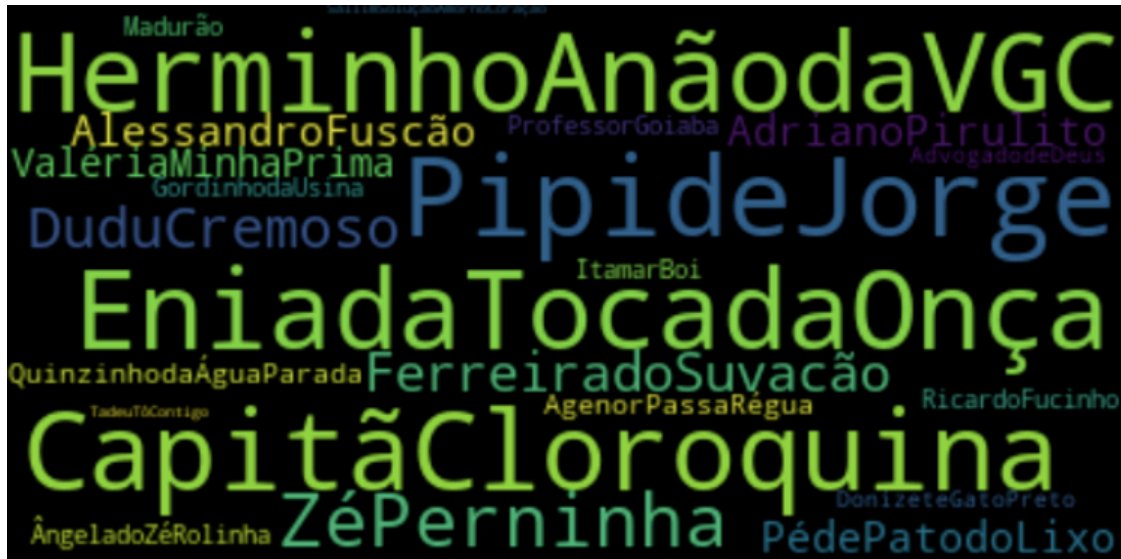
Armazene esses resultados em três dicionários distintos para serem a saída da sua função.

DICA: É interessante trabalhar com a ideia de criar funções auxiliares, quando uma determinada rotina se repete ao longo da função principal.

[]:

1.5 QUESTÃO 5:

Ainda usando o arquivo `dados_eleicao.txt`, crie uma função que retorne um dicionário em ordem decrescente com os 10 candidatos que mais receberam votos. Veja a imagem abaixo, uma *word cloud*, para ter uma ideia da resposta (quanto maior a palavra mais vezes ela se repete).



```
[2]: #Vamos usar os resultados obtidos na Questão 4!  
#candidatos,bairros,cand_bairros=eleicao('dados_eleicao.txt')
```

```
[3]: #Curiosidade! Código para obter a word cloud.  
#!pip3 install WordCloud  
  
# import matplotlib.pyplot as plt  
# from wordcloud import WordCloud  
  
# #Os espaços entre os nomes foram retirados para não serem tratados como  
#   ↳ pessoas diferentes!  
# candidatosWC=[i.replace(' ','') for i in list(candidatos.keys())]  
# wc = WordCloud(background_color="black", repeat=False).generate(" ".  
#   ↳ join(candidatosWC))  
  
# plt.figure(figsize=(12,6))  
# plt.axis("off")  
# plt.imshow(wc, interpolation="bilinear")  
# plt.show()
```

```
[ ]:
```