

# Lista de Exercícios em Python

## 1. Verificar Elementos em uma Lista

- **Descrição:** Utilize o `in` not para verificar se um elemento não está presente na lista.

- **Lista:**

```
lista = [1, 3, 4, 5, 6]
```

- **Exercício:**

- Verifique se o valor `2` está presente na lista e imprima uma mensagem indicando o resultado.

## 2. Converter Lista para Dicionário

- **Descrição:** Converta a lista acima para um dicionário onde os elementos da lista são as chaves e os valores são o dobro das chaves.

- **Lista:**

```
lista = [1, 3, 4, 5, 6]
```

- **Exercício:**

- Desenvolva a lógica de conversão e imprima o dicionário resultante.

## 3. Ajustar um Endereço IP

- **Tópicos Estudados:** Funções, `for`, operadores condicionais.
- **Descrição:** Crie uma função que ajusta um endereço IP para o formato especificado.
- **Entrada:** `1.1.1.1`
- **Saída Esperada:** `1[.]1[.]1[.]1`

- a) Descreva textualmente o passo a passo da sua solução.
- b) Desenvolva a função usando `if` e `for`.
- c) Pesquise e explique o funcionamento da função built-in `replace()` e como ela poderia ser aplicada nesta solução.
- d) Demonstre a solução usando o `replace()`.

## 4. Contar Números com Dígito Par

- **Tópicos Estudados:** Funções, `for`, `len()`, operadores condicionais, e operadores aritméticos.
- **Entrada:**

```
numeros = [12, 342, 345, 2, 8, 7896]
```

- **Exercício:** a) Descreva textualmente o passo a passo da sua solução.
- b) Crie a função que conta quantos números têm um número par de dígitos.
- c) Explique o funcionamento da função built-in `len()`.

## 5. Somar Elementos Únicos de um Array

- **Entrada:**

```
meu_array = [1, 2, 3, 2, 1, 4, 10]
```

- **Exercício:** a) Descreva textualmente o passo a passo da sua solução.
- b) Crie um dicionário que armazene as chaves e valores com a quantidade de cada elemento do array.
- c) Crie a função que retorna a soma dos elementos únicos.

## 6. Retornar o Elemento Que Aparece Uma Vez

- **Entrada:**

```
array = [1, 2, 2, 1, 3, 3, 4, 5, 5, 6, 6]
```

- **Exercício:** a) Descreva textualmente o passo a passo da sua solução.
- b) Crie a função que retorna o único elemento que aparece uma vez.

## 7. Funções de Análise de um Array de Números Inteiros

- **Entrada:**

```
lista = [1, 5, 33, 8, 77, 43, 124, 6, 8, 99]
```

- **Exercício:**
  - a) Crie uma função que retorne o menor número.
  - b) Crie uma função que retorne o maior número.
  - c) Crie uma função que retorne a soma do menor número com o maior número, utilizando `sort()`.
  - d) Crie uma função que retorne a soma de todos os números.
  - e) Crie uma função que retorne a média dos valores do array.

## 8. Manipulação de uma Lista

- **Descrição:** Crie uma lista vazia e realize as seguintes operações:
  - a) Adicione o valor de `10` nela.
  - b) Adicione o valor `20`.
  - c) Adicione o valor `30`.
  - d) Remova o valor `20`.
  - e) Remova o valor `10`.
  - f) Explique o funcionamento do método `append()`.
  - g) Explique o funcionamento do método `pop()`.
  - h) Explique o funcionamento do método `insert()`.
  - i) Explique o funcionamento do método `sort()`.
  - j) Explique o funcionamento do método `count()`.

- **k)** Explique o funcionamento do método `sum()` em uma lista de inteiros.

## 9. Manipulação de um Dicionário

- **Descrição:** Crie um dicionário vazio e realize as seguintes operações:

- **a)** Adicione uma entrada com a chave `a` e o valor `410`.
- **b)** Adicione uma entrada com a chave `b` e o valor `3`.
- **c)** Adicione uma entrada com a chave `b` e o valor `4`.
- **d)** Adicione uma entrada com a chave `c` e o valor `56`.
- **e)** Remova a entrada com a chave `b` que possui o valor `3`.
- **f)** Remova a entrada com a chave `b` que possui o valor `4`.
- **g)** Explique o funcionamento do método `update()`.
- **h)** Explique o funcionamento do método `pop()`.
- **i)** Explique o funcionamento do método `get()`.
- **j)** Explique o funcionamento do método `keys()`.
- **k)** Explique o funcionamento do método `values()`.
- **l)** Explique o funcionamento do método `items()`.

## 10. Verificação de Sequência em uma String

- **Entrada:** Uma string contendo apenas os caracteres `'a'` e `'b'`.
- **Exemplo de entrada:** `"abababab"`
- **Exemplo de entrada:** `"babababa"`
- **Exemplo de entrada:** `"aaaaabbbbbba"`
- **Exercício:** Crie uma função que retorna `True` se cada `'a'` aparece antes de cada `'b'` na string, caso contrário, retorna `False`.

## 11. Encontrar o Único Número que Faltou em um Array

- **Entrada:**

```
array = [0, 2, 3, 4, 5, 6, 7, 8]
```

- **Exercício:**

- Retorne o único número no intervalo  $[0, n]$  que falta no array, onde  $n$  é o tamanho do array.
- Dica: Use o `len()` para obter o tamanho e calcule a soma total com a fórmula da soma dos primeiros  $n$  números.

## 12. Encontrar o Elemento com Mais Frequência em um Array

- **Entrada:**

```
array = [1, 2, 3, 3, 3, 4, 5]
```

- **Exercício:** Crie uma função que encontre o elemento que aparece com mais frequência na lista.

## 13. Converter Uma Lista em Dicionário

- **Entrada:**

```
meu_array = [1, 2, 2, 3, 3, 4, 4, 4, 5, 7, 1]
```

- **Exercício:** **a)** Explique o passo a passo da solução com suas palavras.  
**b)** Desenvolva a solução que converte a lista em um dicionário e retorne o valor do elemento cuja chave é igual ao seu valor. ``

### Dicas para Gerenciamento do Exercício

- Execute os exercícios em uma ordem progressiva, começando com os mais simples e utilizando o conhecimento adquirido para resolver os mais complexos.
- Cada exercício pode ser feito em um ambiente Python interativo, como Jupyter Notebook, Google Colab, ou um editor local.

- Encoraje a escrita de comentários explicativos no código para reforçar a compreensão de cada parte da solução.

Se precisar de ajustes ou de mais exercícios, estou aqui para ajudar!