

# Mundo 1 - Manipulação de Strings

## 1) Crie um programa que leia o nome completo de uma pessoa e mostre:

- O nome com todas as letras maiúsculas
- O nome com todas as letras minúsculas
- Quantas letras ao todo, sem considerar os espaços
- Quantas letras tem o primeiro nome

a )

Esta primeira forma nós iniciamos dando uma certa manipulação no input sendo ele do tipo string e já limpando com o strip os espaços indesejado, após isso são 2 comando simples de colocar em maiúsculo e minúsculo e na 4 linha já fica um pouco mais interessante.

Calculamos a quantidade de letras sem os espaços substituindo os mesmos com literalmente nada representado por "", na próxima linha criamos uma lista, pois todo split cria uma lista.

Na ultima linha é um pouco interessante o uso, pois mostramos o primeiro nome mostrando o primeiro elemento da lista com lista [0] e após isso para sabermos a quantidade de letras, basta darmos um find no primeiro espaço e como a contagem do python começa em 0, sempre vai dar a quantidade certa de letras.

```
nome = str(input('DIGITE SEU NOME: ')).strip()
print('NOME EM LETRAS MAIÚSCULAS --> {}'.format(nome.upper()))
print('NOME EM LETRAS MINÚSCULAS --> {}'.format(nome.lower()))
print('QUANTIDADE DE LETRAS -----> {}'.format(len(nome.replace(' ', ''))))
lista = nome.split()
print('SEU PRIMEIRO NOME É {} E TEM {} LETRAS'.format(lista[0], nome.find(' ')))
```

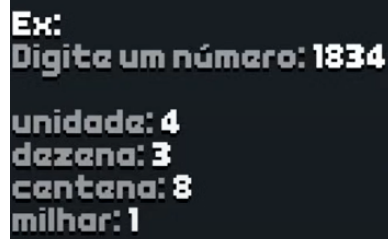
b )

Outra forma de resolver os dois últimos comando seria para saber a quantidade de letras desconsiderando os espaços era diminuir o tamanho da variável em len () com a contagem de quantos espaços existem e sobraria apenas as letras.

No ultimo comando é mais simples, mostramos o primeiro nome apresentando o primeiro elemento da lista e depois o len do primeiro elemento da lista.

```
nome = str(input('DIGITE SEU NOME: ')).strip()
print('NOME EM LETRAS MAIÚSCULAS --> {}'.format(nome.upper()))
print('NOME EM LETRAS MINÚSCULAS --> {}'.format(nome.lower()))
print('QUANTIDADE DE LETRAS -----> {}'.format(len(nome)-nome.count(' ')))
lista = nome.split()
print('SEU PRIMEIRO NOME É {} E TEM {} LETRAS'.format(lista[0], len(lista[0])))
```

2 ) Faça um programa que leia um número de 0 a 9999 e mostre na tela cada um dos dígitos separados.



```
Ex:
Digite um número: 1834

unidade: 4
dezena: 3
centena: 8
milhar: 1
```

a )

Nesta primeira tentativa eu havia colocado o número como int, mas o python não deixa criar listas, ou vetores de números, sendo assim precisa ser uma cadeia de caracteres

Este script possui um problema, pois se o usuário não digitar a centena e milhar o programa dá erro, sendo assim precisamos resolver de forma matemática.

```
num = (input('DIGITE UM NÚMERO INTEIRO DE 0 A 9999: '))
print('UNIDADE: {}'.format(num[3]))
print('DEZENA: {}'.format(num[2]))
print('CENTENA: {}'.format(num[1]))
print('MILHAR: {}'.format(num[0]))
```

b )

Para resgatarmos a unidade, dezena, centena e milhar, podemos seguir uma fórmula sendo uma divisão inteira para pegar o trecho onde está o valor e dividirmos pelo resto de divisão de 10 para pegar o valor específico.

O que não tiver valor retornará zero, isto é uma forma de se resolver sem usar estrutura condicional.

```
num = int(input('DIGITE UM NÚMERO INTEIRO DE 0 A 9999: '))
u = num // 1 % 10
d = num // 10 % 10
c = num // 100 % 10
m = num // 1000 % 10
print('Analisando o número {}'.format(num))
print('UNIDADE: {}'.format(u))
print('DEZENA: {}'.format(d))
```

```
print('CENTENA: {}'.format(c))
print('MILHAR: {}'.format(m))
```

$$\begin{array}{r}
 1234 // 1 \rightarrow 1234 \begin{array}{r} 10 \\ 4 \end{array} \\
 1234 // 10 \rightarrow 123 \begin{array}{r} 10 \\ 3 \end{array} \\
 1234 // 100 \rightarrow 12 \begin{array}{r} 10 \\ 2 \end{array} \\
 1234 // 1000 \rightarrow 1 \begin{array}{r} 10 \\ 1 \end{array}
 \end{array}
 \qquad
 \begin{array}{r}
 23 // 1 \rightarrow 23 \begin{array}{r} 10 \\ 3 \end{array} \\
 23 // 10 \rightarrow 2 \begin{array}{r} 10 \\ 2 \end{array} \\
 23 // 100 \rightarrow 0 \\
 23 // 1000 \rightarrow 0
 \end{array}$$

**3 ) Crie um programa que leia o nome de uma cidade e diga se ela começa ou não com o nome Santo.**

a )

Fluxo de pensamento seria, pegar o input, separar numa lista os nomes e depois pesquisar com o comando IN se no primeiro nome é Santo ou não, pois se pesquisasse no nome inteiro sem sem em lista e fosse Santo no final, daria True, exemplo "Paulo Santo"

```
cidade = input('DIGITE A SUA CIDADE: ').strip().title()
lista = cidade.split()
print('CIDADE {} COMEÇA COM SANTO ? {}'.format(cidade, 'Santo' in lista[0]))
```

b )

Na resolução do guanabara, não precisou separar para listas, ele apenas verificou se nos 5 primeiros caracteres da variável está escrito Santo e ficou mais resumido que o script acima que eu fiz.

```
cid = str(input('Em que cidade você nasceu?: ')).strip()
print(cid[:5].upper() == 'SANTO')
```

**4 ) Crie um programa que leia o nome de uma pessoa e diga se ela tem Silva no nome**

a )

```
nome = input('DIGITE SEU NOME: ').title()
print('O nome {} possui Silva ? --> {}'.format(nome, 'Silva' in nome))
```

## 5 ) Faça um programa que leia uma frase pelo teclado e mostre:

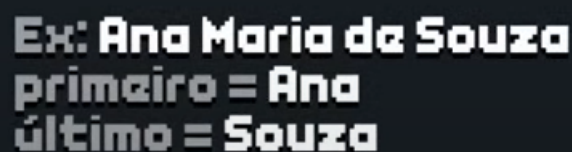
- Quantas vezes aparece a letra "A"
- Em que posição ela aparece a primeira vez
- Em que posição ela aparece a última vez

a )

Ao utilizar o find, podemos somar 1 que mostrara o local exato da letra como comumente as pessoas conta, além disso existe um método de rfind onde ele localiza a partir da direita como também existe o lfind que procura a partir da esquerda.

```
frase = str(input('DIGITE UMA FRASE: ')).strip().upper()
print('QUANTIDADE DE LETRA \'A\': {}'.format(frase.count('A')))
print('PRIMEIRA POSIÇÃO LOCALIZADA: [{}]'.format(frase.find('A')+1))
print('ÚLTIMA POSIÇÃO LOCALIZADA: [{}]'.format(frase.rfind('A')+1))
```

6 ) Faça um programa que leia o nome completo de uma pessoa, mostrando em seguida o primeiro e o último nome separadamente.



Ex: Ana Maria de Souza  
primeiro = Ana  
último = Souza

a )

```
nome = input('DIGITE SEU NOME: ').strip()
lista = nome.split()
```

```
print('PRIMEIRO NOME --> {}'.format(lista[0]))  
print('ULTIMO NOME ----> {}'.format(lista[len(lista)-1]))
```