



- Definição de *Strings*
- Principais Métodos e Operações
- Exemplos de Uso
- Formatar *String* com *f-string*

Python

Lógica de Programação

Strings

01 Definição de Strings

- Strings são basicamente uma lista/array de caracteres em Unicode.
 - Por exemplo a String 'Daniel', apresentada na imagem a seguir, é uma lista os 6 caracteres do nome
 - E, como uma lista, o primeiro índice começa sempre com zero (0) e termina com o tamanho - 1 (neste caso, 6-1 = 5).
 - Complementarmente, veja que, embora incomum, existe a possibilidade de usar índices negativos.

Índices Positivos

0	1	2	3	4	5
---	---	---	---	---	---

D	A	N	I	E	L
---	---	---	---	---	---

-6	-5	-4	-3	-2	-1
----	----	----	----	----	----

Índices Negativos

- Vamos para o nosso primeiro exemplo a seguir.
 - Na linha 1, declaramos a variável Daniel, que pode estar entre aspas simples ' ou duplas ", desde que seja consistente
 - Na linha 3, apresentamos a variável
 - Na linha 4, apresentamos o primeiro caractere da String ('D'), que está na posição 0
 - Opcionalmente, como vimos anteriormente, podemos usar índices negativos, a exemplo da linha 5.

```

1 nome = 'Daniel'
2
3 print(nome)
4 print(nome[0])
5 print(nome[-6])
    
```

02 Principais Métodos e Operações

- A seguir, relacionamos os principais métodos utilizados em String.

Operação	Descrição	Exemplo
len(nome)	Tamanho da String	len(nome)
for	Lista caracteres de uma String	nome = 'Daniel Abella' for letra in nome: print(letra)
s2 in s1	Verifica se substring s2 é parte da String s1	nome = 'Daniel Abella' print('Da' in nome) if 'Da' in nome: print('meu nome tem da')
nome[i]	Obtém caractere de uma data posição	nome[0]
nome[i:j]	Obtém caracteres entre as posições	nome[0:2]
nome.lower()	Converte para minúsculo	nome.lower()
nome.upper()	Converte para maiúsculo	nome.upper()
nome.count('a')	Verifica quantas vezes 'A' aparece em nome	nome.count('a')
nome.isnumeric()	Verifica se a String é composta por números	nome.isnumeric()
nome.isalpha()	Verifica se a String é composta por letras	nome.isalpha()
nome.split()	Quebra a String por espaço	nome = 'Daniel Abella' nome.split(' ') #Retornará 2 strings (uma 'Daniel' e outra 'Abella')
nome.split('-')	Quebra a String em várias utilizando '-' como delimitador	nome = 'Daniel-Abella' nome.split('-') #Retornará 2 strings (uma 'Daniel' e outra 'Abella')
nome.replace(antigo, novo)	Substitui um texto por outro	nome = 'Daniel' nome.replace('da', 'vo') #Retornará 'voniel'
nome.join(sobre nome)	Retorna uma string com o nome e sobrenome concatenados	nome.join('Abella') #Retornará 'DanielAbella'
nome.startswith('Da')	Verifica se o nome inicia com 'Da'	if nome.startswith('Da'): print('inicia com 'Da')
s1 + s2	Junta as 2 strings	s1 + s2
s1 * 5	Repete a s1 por 5 vezes	print(s1 * 5)
capitalize()	Primeira letra maiúscula e o resto minúscula	nome.capitalize()
find('ni')	Verifica em que posição tá 'ni'	nome.find('ni')

03 Exemplos de Uso

- A seguir, apresentaremos exemplos de cada um dos métodos ou operações com códigos objetivos. Experimente fazer você mesmo 😊

For e Len

```
1 nome = 'Daniel'
2 print(len(nome)) #Saída: 13
3 ▼ for letra in nome:
4     print(letra) #Letra a letra
```

Operador in

```
1 nome = 'Daniel'
2 print('Da' in nome) #Saída: true
3 ▼ if 'Da' in nome:
4     print('Da está dentro do nome')
```

Obter substring (pedaço da String)

```
1 nome = 'Daniel Abella'
2 primeiraLetra = nome[0] #Saída 'D'
3 primeiroNome = nome[0:6] #Saída 'Daniel'
4 print(primeiraLetra, primeiroNome)
```

Converte para Maiúsculo e Minúsculo

```
1 nome = 'Daniel'
2 nomeMaiusculo = nome.upper()
3 nomeMinusculo = nome.lower()
4 print(nomeMaiusculo) #Saída: DANIEL
5 print(nomeMinusculo) #Saída: daniel
```

É maiusculo? É minúsculo?

```
1 nome = 'DANIEL'
2
3 ▼ if nome.isupper():
4     print('maiusculo')
5 ▼ elif nome.islower():
6     print('minusculo')
```

Quantos 'a' tem? A string é número ou letra?

```
1 nome = '1'
2 ▼ if nome.isnumeric():
3     print('é número') #Entra aqui
4 ▼ elif nome.isalpha():
5     print('é letra')
6
7 nomeCompleto = 'Daniel Abella'
8 quantidadeA = nomeCompleto.count('a')
9 print(quantidadeA) #Saída: 2
```

Juntar Strings com join e +

```
1 nome = 'Daniel'
2 sobrenome = 'Abella'
3 sobrenomes = ['Abella', 'Souza']
4
5 nomeCompleto1 = nome + sobrenome
6 nomeCompleto2 = nome.join(sobrenomes)
7
8 print(nomeCompleto1) #Saída DanielAbella
9 print(nomeCompleto2) #Saída AbellaDanielSouza
```

Fatiar Strings com Split

```
1 nome1 = 'Daniel Abella'
2 nome2 = 'Daniel-Abella'
3
4 nomesFatiadosEspaco = nome1.split()
5 nomesFatiadosHifen = nome2.split('-')
6
7 ▼ for nome in nomesFatiadosEspaco:
8     print(nome)
9
10 ▼ for nome in nomesFatiadosHifen:
11     print(nome)
```

Saída
Daniel
Abella
Daniel
Abella

Substituir Caractere com Replace

```
1 nome = 'Daniel'
2 nomeModificado = nome.replace('Da', 'vo')
3 print(nomeModificado) #Retornará 'voniel'
```

Começa com 'Da'

```
1 nome = 'Daniel'
2
3 print(nome.startswith('Da'))
4 ▼ if nome.startswith('Da'):
5     print('Nome inicia com Da')
```

Em que posição tá a String 'ni'

```
1 nome = 'Daniel'
2 print(nome.find('ni')) #saída: 2
```

04 Formatar String com f-string

- Existem várias maneiras de formatar as Strings com Python
- Antes do Python 3.6, era usado `$(=formatting)` ou `.format()`
- Agora no Python 3.6, temos o f-string que deixa tudo mais fácil
- Exemplo a seguir.

```
1 nome = 'Daniel'
2 idade = 36
3 print(f"meu nome é {nome} e eu tenho {idade} anos")
```