

# Python - Aula 3

String e Listas

# Python - Strings

## 1) Declaração de Strings

```
x = 'Isto é uma string'
```

```
y = "Isto é outra string"
```

```
z = """Ainda outra string"""
```

# Python - Strings

## 1) Declaração de Strings

```
>>> x = 'Uma string'
>>> x
'Uma string'
>>> 
```

# Python - Strings

## 1) Declaração de Strings

```
>>> x = 'Uma string'
>>> x
'Uma string'
>>> █
```

```
>>>
>>> y = "<a href='#'>Um link</a>"
>>> y
"<a href='#'>Um link</a>"
>>> █
```

# Python - Strings

## 1) Declaração de Strings

```
>>> x = 'Uma string'
>>> x
'Uma string'
>>> █
```

```
>>>
>>> y = "<a href='#'>Um link</a>"
>>> y
"<a href='#'>Um link</a>"
>>> █
```

```
>>>
>>> z = """Uma string com
... multiplas linhas"""
>>> z
'Uma string com\nmultiplas linhas'
>>> █
```

# Python - Strings

## 2) Formatação de Strings

```
local = "feira"
```

```
frase = "Vovó foi a %s" % local
```

```
frase = "Vovó foi a {}".format(local)
```

# Python - Strings

## 2) Formatação de Strings

```
local = "feira"
```

```
pessoa = "Vovó"
```

```
frase = "%s foi a %s" % (pessoa, local)
```

```
frase = "{} foi a {}".format(pessoa, local)
```

# Python - Strings

## 3) Manipulando Strings

```
>>> texto = "Python"  
>>> texto[0]  
'p'  
>>> texto[2]  
't'  
>>> 
```



# Python - Strings

## 3) Manipulando Strings

```
>>> texto[-1]
'n'
>>> texto[:2]
'Py'
>>> texto[3:]
'hon'
>>> 
```

# Python - Strings

## 3) Manipulando Strings

**Obs.:** Os comando utilizados  
podem ser agrupados

```
>>> texto[-1]
'n'
>>> texto[:2]
'Py'
>>> texto[3:]
'hon'
>>> █
```

# Python - Strings

## 4) Métodos de Strings

### Replace:

```
>>> texto = "Vovó foi a feira"
>>> texto.replace("feira", "praça")
'Vovó foi a praça'
>>> texto
'Vovó foi a feira'
>>> █
```

# Python - Strings

## 4) Métodos de Strings

### Split:

```
>>>  
>>> texto = "Vovó foi a feira"  
>>> texto.split(" ")  
['Vovó', 'foi', 'a', 'feira']  
>>> █
```

# Python - Strings

## 4) Métodos de Strings

### Split:

```
>>>  
>>> texto = "Vovó foi a feira"  
>>> texto.split(" ")  
['Vovó', 'foi', 'a', 'feira']  
>>> █
```

```
>>>  
>>> frutas = "abacate,abacaxi,maçã,pera"  
>>> frutas.split(",")  
['abacate', 'abacaxi', 'maçã', 'pera']  
>>> █
```

# Python - Strings

## 4) Métodos de Strings

**Join:**

```
>>>  
>>> frases  
['Vovó foi', 'feira']  
>>> " a ".join(frases)  
'Vovó foi a feira'  
>>> 
```

# Python - Strings

## 4) Métodos de Strings

### Strip:

```
>>>
>>> nome = "    Fulano de Tal    "
>>> nome.strip()
'Fulano de Tal'
>>> nome = "Fulano    de    Tal"
>>> nome.strip()
'Fulano    de    Tal'
>>> █
```

# Python - Strings

## 4) Métodos de Strings

### Upper:

```
>>> texto  
'Vovó foi a feira'  
>>> texto.upper()  
'VOVÓ FOI A FEIRA'  
>>> 
```



# Python - Strings

## 4) Métodos de Strings

### Upper:

```
>>> texto  
'Vovó foi a feira'  
>>> texto.upper()  
'VOVÓ FOI A FEIRA'  
>>> █
```

### Lower:

```
>>> texto  
'Vovó foi a feira'  
>>> texto.lower()  
'vovó foi a feira'  
>>> █
```

# Python - Strings

## 4) Métodos de Strings

### Upper:

```
>>> texto  
'Vovó foi a feira'  
>>> texto.upper()  
'VOVÓ FOI A FEIRA'  
>>> █
```

### Lower:

```
>>> texto  
'Vovó foi a feira'  
>>> texto.lower()  
'vovó foi a feira'  
>>> █
```

### Capitalize:

```
>>> texto  
'Vovó foi a feira'  
>>> texto.capitalize()  
'Vovó foi a feira'  
>>> █
```

# Python - Strings

## 4) Métodos de Strings

### Upper:

```
>>> texto  
'Vovó foi a feira'  
>>> texto.upper()  
'VOVÓ FOI A FEIRA'  
>>> █
```

### Lower:

```
>>> texto  
'Vovó foi a feira'  
>>> texto.lower()  
'vovó foi a feira'  
>>> █
```

### Capitalize:

```
>>> texto  
'Vovó foi a feira'  
>>> texto.capitalize()  
'Vovó foi a feira'  
>>> █
```

### Title:

```
>>> texto  
'Vovó foi a feira'  
>>> texto.title()  
'Vovó Foi A Feira'  
>>> █
```

# Python - Listas

Lista é uma **estrutura de dados** composta por itens de forma linear, na qual cada um pode ser acesso através de um índice que representa sua posição na coleção

# Python - Listas

Lista é uma **estrutura de dados** composta por itens de forma linear, na qual cada um pode ser acesso através de um índice que representa sua posição na coleção.

Os índices de uma lista são **inteiros**, iniciando por **zero**.

# Python - Listas

## 5) Declarando Listas

```
x = []
```

```
x = ["abacate", "pera", "laranja"]
```

```
x = list()
```

```
x = [True, "feira", 12, [57, "Vovó"], 3.14]
```

# Python - Listas

## 6) Operadores de Lista

**Índice:**

```
>>> lista = ["Vovó foi a feira", 12, True, 3.14]
>>> lista[0]
'Vovó foi a feira'
>>> lista[2]
True
>>> █
```

# Python - Listas

## 6) Operadores de Lista

### Existência de item em lista (IN):

```
>>> lista  
['Vovó foi a feira', 12, True, 3.14]  
>>> 12 in lista  
True  
>>> 70 in lista  
False  
>>> █
```



# Python - Listas

## 7) Métodos de Lista

### Append:

```
>>>  
>>> lista  
['Vovó foi a feira', 12, True, 3.14]  
>>> lista.append("Jacaré")  
>>> lista  
['Vovó foi a feira', 12, True, 3.14, 'Jacaré']  
>>> █
```

# Python - Listas

## 7) Métodos de Lista

### Insert:

```
>>> lista
['Vovó foi a feira', 12, True, 3.14, 'Jacaré']
>>> lista.insert(2, "Maçã")
>>> lista
['Vovó foi a feira', 12, 'Maçã', True, 3.14, 'Jacaré']
>>> █
```

# Python - Listas

## 7) Métodos de Lista

### Pop:

```
>>> lista
['Vovó foi a feira', 12, 'Maçã', True, 3.14, 'Jacaré']
>>> lista.pop()
'Jacaré'
>>> lista
['Vovó foi a feira', 12, 'Maçã', True, 3.14]
>>> █
```

# Python - Listas

## 7) Métodos de Lista

### Pop:

**Obs.:** O método **pop()** **DEVOLVE** o valor removido da lista

```
>>> lista
['Vovó foi a feira', 12, 'Maçã', True, 3.14, 'Jacaré']
>>> lista.pop()
'Jacaré'
>>> lista
['Vovó foi a feira', 12, 'Maçã', True, 3.14]
>>> █
```

# Python - Listas

## 7) Métodos de Lista

### Remove:

```
>>> lista
['Vovó foi a feira', 12, 'Maçã', True, 3.14]
>>> lista.remove(True)
>>> lista
['Vovó foi a feira', 12, 'Maçã', 3.14]
>>> 
```

# Python - Listas

## 7) Métodos de Lista

### Remove:

**Obs.:** O método `remove()` **NÃO DEVOLVE** o valor removido da lista

```
>>> lista
['Vovó foi a feira', 12, 'Maçã', True, 3.14]
>>> lista.remove(True)
>>> lista
['Vovó foi a feira', 12, 'Maçã', 3.14]
>>> 
```