

Aula 08

Revisando

Na aula anterior vimos um novo tipo do Python, o tipo lista.

Vimos como acessar e alterar pelo índice, adicionar no final com `append`, remover com `pop` e com `remove`, repartir ela com os colchetes e dois pontos (e aplicando também nas strings esse conceito).

```
>>> misto = ['Mercúrio', 42, 'Terra', 3.14, True, [1,2,3]]
>>> misto
['Mercúrio', 42, 'Terra', 3.14, True, [1, 2, 3]]
>>> misto[0] = 'Vênus'
>>> misto
['Vênus', 42, 'Terra', 3.14, True, [1, 2, 3]]
>>> misto.append(5)
>>> misto
['Vênus', 42, 'Terra', 3.14, True, [1, 2, 3], 5]
>>> misto.pop(3)
3.14
>>> misto
['Vênus', 42, 'Terra', True, [1, 2, 3], 5]
>>> misto.remove(True)
>>> misto
['Vênus', 42, 'Terra', [1, 2, 3], 5]
>>> misto[2:]
['Terra', [1, 2, 3], 5]
```

E vimos com ordenar uma lista com `sort` e com `reverse`.

```
>>> numeros = [1,5,9,5,1,12,1461,0,-4,-564]
>>> numeros
[1, 5, 9, 5, 1, 12, 1461, 0, -4, -564]
>>> numeros.sort()
>>> numeros
[-564, -4, 0, 1, 1, 5, 5, 9, 12, 1461]
>>> numeros.sort(reverse=True)
>>> numeros
[1461, 12, 9, 5, 5, 1, 1, 0, -4, -564]
>>> numeros.sort()
>>> numeros
[-564, -4, 0, 1, 1, 5, 5, 9, 12, 1461]
>>> numeros.reverse()
>>> numeros
[1461, 12, 9, 5, 5, 1, 1, 0, -4, -564]
```

Lembre-se que não é possível ordenar uma lista de tipos diferentes com esses métodos.

Também vimos a função built-in, que retorna o tamanho das listas e strings.

```
>>> planetas = ['Mercúrio', 'Vênus', 'Terra', 'Marte']
>>> len(planetas)
4
>>> nome = 'Gandalf, o Cinzento'
>>> len(nome)
19
```

E vimos como dividir as strings em listas com split e depois como recriar uma string a partir de uma lista.

```
>>> nome = 'Gandalf, o Cinzento'
>>> nome.split()
['Gandalf,', 'o', 'Cinzento']
>>> nome.split(',')
['Gandalf', ' o Cinzento']
>>> lista = nome.split()
>>> lista
['Gandalf,', 'o', 'Cinzento']
>>> '+'.join(lista)
'Gandalf,+o+Cinzento'
```

Comando for

O loop for (laço de repetição usando o for) é usado para iterar sobre uma sequência (como listas, tuplas, conjuntos) ou um objeto iterável (como a string).

```
for <variável> in <sequência>:
    # código a ser executado para cada item na sequência
```

A cada iteração, o valor do próximo item na sequência é atribuído à variável especificada e o código dentro do loop é executado. Quando todos os itens na sequência tiverem sido iterados (tiverem sido exauridos), o loop é finalizado.

No exemplo a seguir, a cada passagem pela lista é pego o próximo valor dela e associado à variável i.

```
>>> for i in [1,2,3,4,5]:
...     print('i =',i)
...
i = 1
i = 2
i = 3
i = 4
i = 5
```

Também podemos usar uma variável iterável no for :

```
>>> planetas = ['Mercúrio', 'Vênus', 'Terra', 'Marte']
>>> for planeta in planetas:
...     print('O planeta é {}'.format(planeta))
...
O planeta é Mercúrio
O planeta é Vênus
O planeta é Terra
O planeta é Marte
```

Neste exemplo, usamos o loop for para passar por cada letra na palavra :

```
>>> palavra = 'superman'
>>> for letra in palavra:
...     print('\t',letra)
...
        s
        u
        p
        e
        r
        m
        a
        n
```

Podemos ter um for dentro do outro.

```
>>> elfos = 'Três Anéis para os Reis-Elfos sob este céu;'
>>> lista = elfos.split()
>>> lista
['Três', 'Anéis', 'para', 'os', 'Reis-Elfos', 'sob', 'este', 'céu']
>>> for palavra in lista:
...     for letra in palavra:
...         print(letra, end='-')
...     print()
...
T-r-ê-s-
A-n-é-i-s-
p-a-r-a-
o-s-
R-e-i-s---E-l-f-o-s-
s-o-b-
e-s-t-e-
c-é-u;-
```

Função range

A função `range()` em Python é usada para criar uma sequência numérica específica. Ela é muito útil para percorrer sequências numéricas e para criar loops `for` que precisam executar um número específico de vezes. A sintaxe geral da função é:

```
range(<início>, [<fim>], [<passo>])
```

Onde

- início (obrigatório) : é o número inicial da sequência;
- fim (opcional) : é o número final da sequência (não incluso na sequência);
- passo (opcional) : é o tamanho do incremento (por padrão é 1);

Exemplos :

```
>>> for i in range(5):  
...     print(i)  
...  
0  
1  
2  
3  
4
```

No exemplo acima, o `range()` foi chamado com um único argumento, 5. Isso cria uma sequência de números inteiros de 0 a 4. O valor inicial é 0 e o valor final é 4 (repare que o 5 não está incluído na sequência). O tamanho do incremento é 1, que é o valor padrão.

```
>>> for i in range(2,21,2):  
...     print(i)  
...  
2  
4  
6  
8  
10  
12  
14  
16  
18  
20
```

No exemplo acima, o `range()` foi chamado com três argumentos, 2, 21 e 2. Isso cria uma sequência de números inteiros de 2 a 20, com incremento de 2. O valor inicial é 2, o valor final é 21 (que não está incluído na sequência) e o tamanho do incremento é 2.

A função `range()` também suporta valores negativos para os argumentos início, fim e passo.

Valores negativos para início e fim podem ser usados para criar sequências de números decrescentes, enquanto valores negativos para passo podem ser usados para criar sequências de números decrescentes com incrementos específicos.

No exemplo abaixo, o `range()` foi chamado com três argumentos, 9, -1 e -1. Isso cria uma sequência de números inteiros de 9 a 0, com incremento de -1. O valor inicial é 9, o valor final é -1 (que não está incluído na sequência) e o tamanho do incremento é -1.

```
>>> for i in range(9,-1,-1):
...     print(i)
...
9
8
7
6
5
4
3
2
1
0
```

É importante notar que, quando se usa valores negativos, é importante garantir que o valor inicial seja menor que o valor final, caso contrário, a sequência será vazia.

```
>>> for i in range(5,1):
...     print(i)
...
>>> _
```

Podemos usar o `range` com o `for` para preencher uma lista de strings.

```
>>> lista = []
>>> for i in range(3):
...     texto = input('digite algo : ')
...     lista.append(str(i) + ' : ' + texto)
...
digite algo : eu vi algo incrível
digite algo : python é muito divertido
digite algo : não sei o que escrever
>>>
>>> lista
['0 : eu vi algo incrível', '1 : python é muito divertido', '2 : não sei o que escrever']
>>> for item in lista:
...     print(item)
...
0 : eu vi algo incrível
1 : python é muito divertido
2 : não sei o que escrever
```

Casting para list do range

Além disso, podemos usar a função `range()` para criar listas. Assim como convertemos uma string em um inteiro anteriormente, nós convertemos o range em uma lista. Para esse cast, usamos a função `list()`.

```
>>> # cria um iterador com o range
>>> numeros = range(10)
>>> type(numeros)
<class 'range'>
>>> print(numeros)
range(0, 10)
>>>
>>> # converte para uma lista
>>> lista_int = list(numeros)
>>> type(lista_int)
<class 'list'>
>>> lista_int
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
```

Isso cria uma lista de números inteiros de 0 a 9.

for e range

É possível usar a função `range()` para criar uma sequência de números e iterar sobre ela. Por exemplo, este loop imprimirá os números de 1 a 5:

```
>>> for num in range(1,6):
...     print(num)
...
1
2
3
4
5
```

Usando o teste de par e ímpar, podemos testar todos os números gerados pelo range e mostrar apenas os números pares:

```
>>> for num in range(10):
...     if num % 2 == 0:
...         print('é par :',num)
...     else:
...         print('é ímpar :',num)
...
é par : 0
é ímpar : 1
é par : 2
é ímpar : 3
é par : 4
é ímpar : 5
é par : 6
é ímpar : 7
é par : 8
é ímpar : 9
```

Dicas VS Code

Atalhos:

Legenda -> [comando] : [descrição]

- alt + seta cima -> move a linha onde está o cursor para cima
- alt + seta baixo -> move a linha onde está o cursor para baixo
- alt + clique esquerdo do mouse -> insere múltiplos cursores onde clicar

Sugestões de Extensões:

- indent-rainbow – ajuda na indentação do código;
- IntelliCode – inteligência artificial para ajudar nas sugestões do código;

Exercícios para Entregar

1. Faça um programa que leia um número inteiro e exiba a soma dos seus dígitos. Exemplo: 486 vai gerar $4+8+6 = 18$.
2. Utilizando apenas 2 comandos for, exiba a tabuada dos números 1 a 10.
3. Crie um algoritmo que peça uma frase a um usuário. Converta para uma lista. Verifique qual é a maior palavra da lista. Ao final mostre a maior palavra, quantas letras ela tem e em qual posição ela foi encontrada.
4. Crie um programa que peça para o usuário digitar um texto. Pegue a string e separe e 3 listas distintas o que é caractere, número e espaços em branco. Depois exiba os conteúdos das listas.

Exercícios para Praticar

for e strings

1. Crie um programa que exiba cada caractere de uma string utilizando o laço de repetição for.
2. Faça um programa que leia uma string e conte o seu tamanho utilizando o laço de repetição for (não utilize a função len).
3. Crie um programa que leia uma frase longa e exiba cada palavra em uma linha utilizando o laço de repetição for.
4. Faça um programa que leia uma string e exiba apenas as vogais utilizando o laço de repetição for.
5. Crie um programa que leia uma string e exiba apenas as consoantes utilizando o laço de repetição for.
6. Faça um programa que leia uma string e exiba apenas as letras maiúsculas utilizando o laço de repetição for.
7. Crie um programa que leia uma string e exiba apenas as letras minúsculas utilizando o laço de repetição for.
8. Faça um programa que leia uma string e exiba cada caractere em uma linha utilizando o laço de repetição for.
9. Crie um programa que leia uma string e substitua todas as letras "a" por "*" utilizando o laço de repetição for (não utilize o método replace das strings).
10. Faça um programa que leia uma string e exiba a string invertida utilizando o laço de repetição for (não utilize listas para isso).
11. Crie um programa que leia uma frase e exiba apenas as palavras que começam com a letra "a" utilizando o laço de repetição for. Se não tiver nenhuma palavra que comece com 'a', avise ao usuário.
12. Faça um programa que leia uma string e verifique se ela é um palíndromo utilizando o laço de repetição for.
13. Crie um programa que leia uma string e exiba apenas as palavras que possuem mais de 5 caracteres utilizando o laço de repetição for.
14. Faça um programa que leia uma frase e exiba apenas as palavras que possuem a letra "e" utilizando o laço de repetição for.
15. Crie um programa que leia uma string e conte quantas vezes uma determinada letra aparece na string utilizando o laço de repetição for (não utilize o método count das strings).
16. Faça um programa que leia uma string e exiba apenas as palavras que terminam com a letra "o" utilizando o laço de repetição for.
17. Crie um programa que leia uma string e exiba apenas as palavras que possuem a letra "u" em qualquer posição utilizando o laço de repetição for.
18. Faça um programa que leia uma string e substitua todas as letras minúsculas por maiúsculas utilizando o laço de repetição for.
19. Crie um programa que leia uma string e exiba apenas as palavras que possuem a letra "i" em qualquer posição utilizando o laço de repetição for.
20. Crie um programa que leia uma lista de nomes e exiba-os na tela utilizando o laço de repetição for.

for e números

1. Crie um programa que exiba os números de 0 a 10 utilizando o laço de repetição for.
2. Faça um programa que exiba os números ímpares de 1 a 20 utilizando o laço de repetição for.

3. Crie um programa que exiba os números pares entre 0 e 50 utilizando o laço de repetição for.
4. Faça um programa que leia um número e exiba a tabuada de multiplicação desse número utilizando o laço de repetição for.
5. Crie um programa que leia um número e exiba todos os números pares de 0 até esse número utilizando o laço de repetição for.
6. Faça um programa que leia um número e exiba todos os números ímpares de 0 até esse número utilizando o laço de repetição for.
7. Crie um programa que leia um número e exiba a soma de todos os números pares de 0 até esse número utilizando o laço de repetição for.
8. Faça um programa que leia um número e exiba a soma de todos os números ímpares de 0 até esse número utilizando o laço de repetição for.
9. Crie um programa que leia um número e exiba se ele é primo ou não utilizando o laço de repetição for.
10. Crie um programa que leia um número e exiba todos os números primos de 0 até esse número utilizando o laço de repetição for.
11. Crie um programa que leia um número e exiba todos os números divisíveis por 3 de 0 até esse número utilizando o laço de repetição for.
12. Crie um programa que leia um número e exiba a sequência de Fibonacci até esse número utilizando o laço de repetição for.
13. Faça um programa que leia um número e uma string e exiba a string a quantidade de vezes do número utilizando o laço de repetição for.
14. Crie um programa que leia um número e exiba a sua tabuada de adição utilizando o laço de repetição for.
15. Faça um programa que leia um número e exiba a sua tabuada de subtração utilizando o laço de repetição for.
16. Crie um programa que leia um número e exiba a sua tabuada de multiplicação utilizando o laço de repetição for.
17. Faça um programa que leia um número e exiba a sua tabuada de divisão utilizando o laço de repetição for.
18. Crie um programa que leia um número e exiba o seu fatorial utilizando o laço de repetição for (não utilize a função factorial do módulo math).

for e listas

1. Faça um programa que leia uma lista de 10 números e calcule a média aritmética utilizando o laço de repetição for.
2. Faça um programa que leia uma lista de números e calcule o seu produto utilizando o laço de repetição for.
3. Crie um programa que leia uma lista de nomes e exiba o número de caracteres de cada nome utilizando o laço de repetição for.
4. Faça um programa que leia uma lista de números e exiba o maior e o menor valor utilizando o laço de repetição for.
5. Faça um programa que leia uma lista de números e exiba apenas os números maiores que 10 utilizando o laço de repetição for.
6. Crie um programa que leia uma lista de nomes e exiba apenas os nomes que começam com a letra "A" utilizando o laço de repetição for.
7. Faça um programa que leia uma lista de números e exiba apenas os números pares utilizando o laço de repetição for.
8. Crie um programa que leia uma lista de nomes e exiba apenas os nomes que contém a letra "e" utilizando o laço de repetição for.
9. Faça um programa que leia uma lista de números e exiba a soma dos números pares utilizando o laço de repetição for.
10. Crie um programa que leia uma lista de nomes e exiba apenas os nomes que possuem mais de 5 caracteres utilizando o laço de repetição for.
11. Faça um programa que leia uma lista de números e exiba a média aritmética dos números pares utilizando o laço de repetição for.

12. Faça um programa que leia uma lista de números e exiba a média aritmética dos números ímpares utilizando o laço de repetição for.
13. Crie um programa que leia uma lista de nomes e exiba apenas os nomes que possuem a letra "a" na segunda posição utilizando o laço de repetição for.
14. Faça um programa que leia uma lista de números e exiba a soma dos números ímpares utilizando o laço de repetição for.
15. Crie um programa que leia uma lista de nomes e exiba apenas os nomes que terminam com a letra "o" utilizando o laço de repetição for.
16. Crie um programa que leia uma lista de nomes e exiba apenas os nomes que possuem mais de uma vogal utilizando o laço de repetição for.
17. Faça um programa que leia uma lista de números e exiba apenas os números que são divisíveis por 3 utilizando o laço de repetição for.
18. Crie um programa que leia uma lista de nomes e exiba apenas os nomes que possuem mais de uma palavra utilizando o laço de repetição for.
19. Faça um programa que leia uma lista de números e exiba a soma dos números que estão nas posições ímpares da lista utilizando o laço de repetição for.
20. Crie um programa que leia uma lista de nomes e exiba apenas os nomes que possuem a letra "u" em qualquer posição utilizando o laço de repetição for.