



Linguagem de Programação

Professor Anderson I. S. Abreu

Estruturas de dados em Python – Parte I



Tópicos da nossa aula:

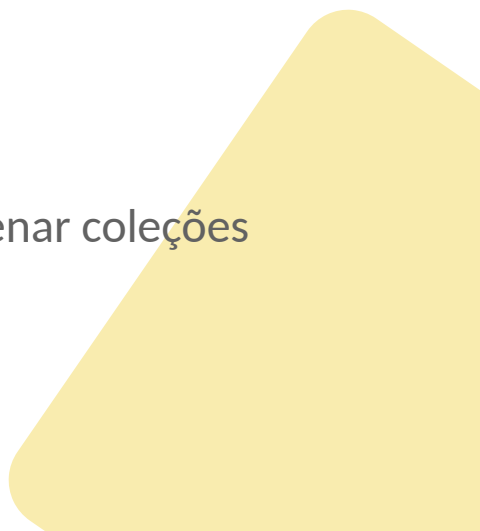
1. Objetos do tipo sequência;
2. Lista;
3. Tuplas.



1

Objetos do tipo sequência

Sequências são estruturas de dados que nos permitem armazenar coleções ordenadas de informações.

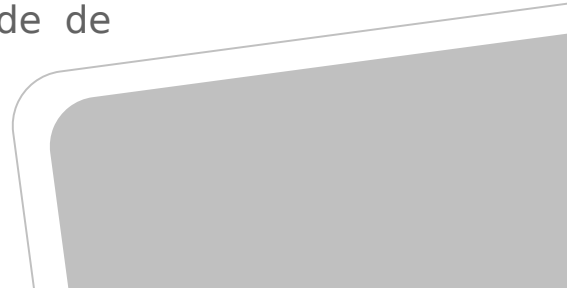


Vai começar a sequência!

Coleções versáteis que podem armazenar vários valores;

Servem para organizar dados em uma ordem específica e são indexados por números inteiros não negativos;

O primeiro elemento da sequência é acessado pelo índice 0, o segundo pelo índice 1, e assim por diante, até o último elemento, que está na posição $n - 1$, onde n representa a capacidade de armazenamento da sequência;



Operações em comum de sequências.

Operação	Resultado
<code>x in s</code>	True caso um item de <code>s</code> seja igual a <code>x</code> , senão false
<code>s + t</code>	Concatenação de <code>s</code> e <code>t</code>
<code>n * s</code>	Adiciona <code>s</code> a si mesmo <code>n</code> vezes
<code>s[i]</code>	Acessa o valor guardado na posição <code>i</code> da sequência
<code>s[i:j]</code>	Acessa os valores da posição <code>i</code> até <code>j</code>
<code>s[i:j:k]</code>	Acessa os valores da posição <code>i</code> até <code>j</code> , com passo <code>k</code>
<code>len(s)</code>	Comprimento de <code>s</code>
<code>min(s)</code>	Menor valor de <code>s</code>
<code>max(s)</code>	Maior valor de <code>s</code>
<code>s.count(x)</code>	Número total de ocorrência de <code>x</code> em <code>s</code>

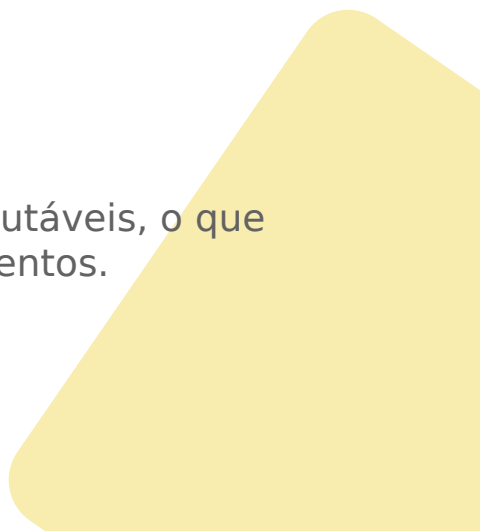
Vamos para um exemplo!



2

Listas

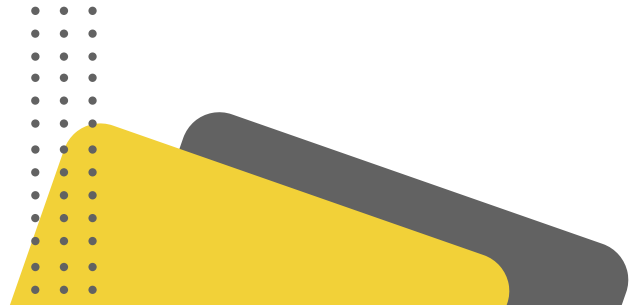
São uma forma fundamental de objetos do tipo sequência, são mutáveis, o que significa que podemos adicionar, remover e alterar elementos.



Mutabilidade!

As listas são estruturas de dados em Python conhecidas por sua mutabilidade, o que significa que você pode adicionar ou remover elementos conforme necessário.

Vamos de exemplo!!!!



List comprehensions, Map e Filter

As list comprehensions, ou listcomps, são uma abordagem pythônica para criar listas com base em objetos iteráveis;

Especialmente útil quando você deseja transformar ou filtrar as informações de uma sequência existente para criar uma nova sequência com as informações desejadas;

Vamos de exemplo!!!!

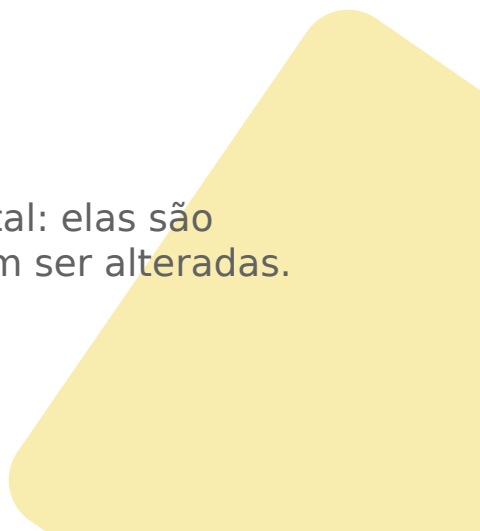




3

Tuplas

São semelhantes às listas, mas com uma diferença fundamental: elas são imutáveis. Isso significa que, uma vez criadas, as tuplas não podem ser alteradas.

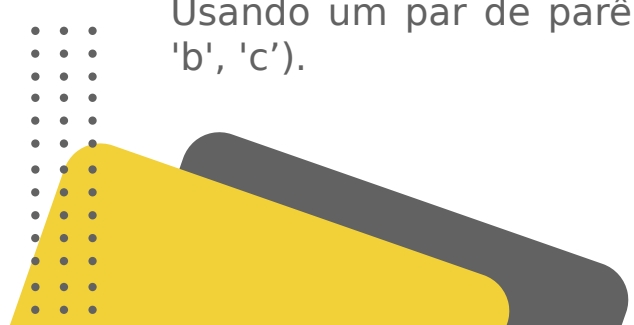


Imutável

A distinção principal entre listas e tuplas é que as listas são mutáveis, permitindo a atribuição de valores a posições específicas, enquanto as tuplas são objetos imutáveis;

Tupla vazia: `tupla1 = ()`;

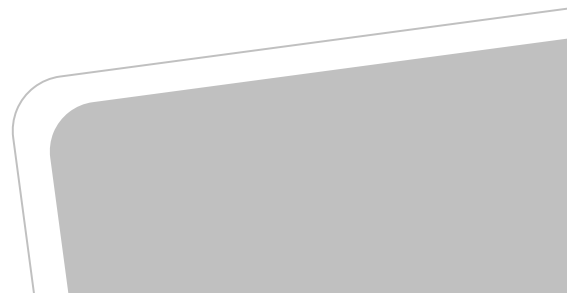
Usando um par de parênteses e elementos separados por vírgulas: `tupla2 = ('a', 'b', 'c')`.



Vogais

```
vogais = ('a', 'e', 'i', 'o', 'u')  
print(f"Tipo do objeto vogais = {type(vogais)}")  
for p, x in enumerate(vogais):  
    print(f"Posição = {p}, valor = {x}")
```

Vamos ao código!





Aplicando a aula!

Você está gerenciando uma lista de convidados para uma festa e uma lista de pessoas que confirmaram sua presença. Você deseja identificar as pessoas que ainda não confirmaram e convidá-las novamente.

Obrigado!

