

 04

Conclusão

Transcrição

[00:00] Nosso curso chegou ao final. Quero agradecer muito por terem assistido esse curso comigo e espero que tenham gostado, que tenham feito todos os exercícios e que tenham tirado as dúvidas.

[00:13] Vamos fazer uma revisão rápida de tudo que aprendemos. Primeiro, começamos construindo nossa classe, extraíndo argumentos, usando só o método `find`, só para pegar argumentos de dentro da nossa url.

[00:30] Depois disso, aprendemos o método `len`, que tivemos que criar um método diferente dentro da nossa classe para conseguir usar esse `len` de forma mais limpa, para não ter uma lógica muito grande dentro de outra classe, que ficaria muito difícil de fazer a manutenção depois.

[00:50] Depois de aprender o método `len`, aprendemos o método `lower`, para deixar nossa classe mais forte e pronta para qualquer tipo de problema que possa vir do sistema. A nossa classe vem de uma forma, com maiúsculas, minúsculas, misturado, não importa. Ela vai cair com o atributo em tudo minúsculo para nossa instância. Usamos o `lower` também para não ter risco de procurar uma coisa que não exista na nossa url.

[01:22] Depois disso, vimos o método `replace`, para caso tenha algum outro tipo de bug da nossa classe, em que o usuário digitasse errado a moeda origem. Deixamos nossa classe bem robusta.

[01:38] Depois de aprender alguns métodos de string, aprendemos `regex`, que descobrimos ser formas de retirar padrões de texto bem definidos, como um número de celular, e-mail, ou qualquer coisa desse tipo de dentro de um texto humano, de um post de Facebook, um comentário no Instagram, uma mensagem no WhatsApp, qualquer coisa parecida.

[02:02] Nós olhamos muito a documentação e vimos que a documentação do Python é onde está todo o material que podemos saber da linguagem.

[02:16] Depois, voltamos para a nossa classe e começamos a construir métodos especiais. Vocês viram que métodos especiais são formas de deixarmos nossa classe mais forte e fazer com que ela tenha funções mais adequadas para ela e para coisas que você quer retornar usando funções do Python. Vimos que quando criamos uma variável do tipo `int` ou `str` ou `float`, que não usamos, estamos acessando uma classe built-in do Python, que é a classe responsável por dar esses métodos aos nossos tipos de variáveis, ao `string`, por exemplo. Vimos que a classe `str` tem o método `replace`, `lower`, `len`.

[03:03] Vimos que a classe `int` não tem o método especial `len`, e por isso quando tentamos usar a função `len` no tipo `int`, ele não funciona. Ele retorna para nós que o tipo `int` não possui `len`. E vimos que esse tipo de funcionalidade pode ser aplicada dentro das nossas classes.

[03:23] Depois, fizemos o principal. Demos uma representação `string` para a nossa classe, e a nossa classe começou a não dar vários números aleatórios quando printamos. Ela passou a ser uma coisa visível, uma coisa que faz sentido para quem está vendo.

[03:48] Vimos o método `format` também, que é um método de uma classe `string`. Com ele conseguimos fazer de forma mais bonita essa representação. E por fim, criamos outro método especial, o `eq`, que posso comparar instâncias diferentes da mesma classe.

[04:14] Lembram que antes de ter isso, ao invés de comparar pela url, o Python comparava pelo `id`? Porque ele não sabia por onde comparar e os `ids` eram diferentes uns dos outros.

[04:29] Busquem mais conteúdo de Python, é uma linguagem muito bacana e que está crescendo muito. Vai ser muito importante para a carreira de vocês. Espero que tudo que vimos aqui seja aplicável e que vocês coloquem na caixa de ferramentas de desenvolvedores de vocês.

[04:43] Muito obrigado por assistir e até logo.