

Iniciando com Strings

Transcrição

[00:00] Fala, pessoal. Bem-vindos ao nosso curso de Python.

[00:01] Quero começar a introduzir a vocês o que é uma string. Dar um conceito bem básico para nas próximas aulas começarmos a criar nosso projeto aplicando alguns métodos mais avançados das strings.

[00:15] Primeiramente, uma string é uma sequência de caracteres um do lado do outro. Isso significa que se eu tenho várias letras uma do lado da outra, tenho uma palavra. Se tenho várias palavras juntas, tenho uma frase. Se tenho várias frases juntas, tenho um texto, uma música, um livro.

[00:37] Uma string, em essência, serve para trabalharmos com textos. Como criamos isso dentro do Python? Vou começar criando uma variável que vai ser uma string e vou mostrá-la na tela.

[00:51] Quero uma variável bem simples, o meu nome, por exemplo: `meuNome = Rodrigo`

[01:01] E vou dar um print: `print (meuNome)`

[01:05] Não se preocupe com a estrutura do PyCharm que na aula que vem vamos criar um projeto juntos. Veja só, deu erro, disse que a o nome não está definido, porque o Python buscou uma variável com o nome de Rodrigo. Ele não mostrou uma string para nós. Ele tentou mostrar uma variável.

[01:26] Toda string precisa estar entre aspas para poder ser uma string: `meuNome = "Rodrigo"`

[01:36] Printando de novo, funciona normalmente. Podem ser aspas duplas ou aspas simples, mas tem que ser o mesmo tipo de aspa no início e no fim: `meuNome = 'Rodrigo'`

[01:53] Percebe que não precisei dizer para o Python o que é essa minha variável? Não precisei colocar antes de tudo um string, ou um str, por exemplo. Ele inclusive acusa um erro se eu tentar fazer isso, porque o Python é uma linguagem dinamicamente tipada. Ele sozinho identifica qual o tipo de variável está sendo trabalhado naquele momento.

[02:18] Falando em tipo, o Python também possui uma função bem bacana, que é a função `type`. Com ela, consigo mostrar ao usuário ou para mim mesmo que aquela variável é do tipo que quero que ela seja: `print (type(meuNome))`

[02:39] Rodando, ele me diz que a classe é `str`. A classe `str` é a classe do Python responsável por lidar com strings em geral.

[02:50] Agora quero criar outra variável qualquer e ver o tipo dela: `minhaIdade = 26`

[02:59] Vou printar essa variável: `print (minhaIdade)`. E vou printar também o tipo: `print (type(minhaIdade))`

[03:12] Nós sabemos que 26 é um inteiro, e é isso que o Python nos diz. É da classe `INT`, que no Python é responsável por lidar com números inteiros. Se eu adicionar um `.3`, ele me diz que é do tipo `FLOAT`, porque não é mais um inteiro.

[03:38] Vou apagar esses prints antigos e vou criar a variável: `sobreMim = 'Meu nome é Rodrigo e minha idade é 26'`. Vou printar isso: `print(sobreMim)`, e também vou ver o tipo: `print (type(sobreMim))`.

[04:10] É um texto com um número dentro. Será que ela é inteira ou uma string? O Python acusa como uma string, porque está entre aspas. É um texto, não é mais uma palavra qualquer. É uma coisa maior, é um conjunto de palavras.

[04:29] Na aula que vem vamos ver os métodos que veremos ao longo do curso, e também vamos começar a fazer o fatiamento, a retirar sub-strings de dentro de uma string maior. Por exemplo, se eu quisesse ao invés de pegar a string completa pegar somente minha idade, na aula que vem vamos começar a descobrir como fazer isso.