

Para saber mais: Alterações com strings

No último vídeo vimos uma série de funções que fazem parte do tipo `str` (*string*). Algumas funções foram úteis para verificar se há alguma `substring` dentro da palavra, como por exemplo `find`, `startswith` ou `endswith`.

Outras funções tem a função de alterar a palavra como `upper`, `lower`, `capitalize` ou `strip`. Quando falamos sobre esse tipo de funções, é importante lembrar que elas na verdade não alteram a string original. Por exemplo, veja esse código abaixo e tente descobrir a saída:

```
>>> palavra = "alura"  
>>> palavra.upper()  
>>> print(palavra) #qual é o resultado?
```

A resposta é "alura" com letras minúsculas! Isto é porque o `upper` sempre devolve uma nova string e não altera a string original. Essa regra também aplica para todas as outras funções do tipo `str` !

O tipo `str` foi criado de tal maneira que é impossível alterá-lo, qualquer função de alteração devolve uma nova string, que representa a alteração. Esse principio também é chamado de **imutabilidade**. Strings são imutáveis, são **sequências imutáveis**!

O mais legal ainda é que strings não são a única sequência imutável e veremos ainda nesse curso uma outra. Aguarde :)