**Para saber mais: cópias de arrays**

Ao trabalharmos com arrays ou outras variáveis em Python, eventualmente fazemos cópias das variáveis que estamos lidando. Quando fazemos a cópia de uma variável, temos que ter em mente o que acontece com ela para evitar que usemos determinados valores pensando que são outros valores. Para lidar com cópias dos arrays, a biblioteca Numpy possui uma função específica.

Neste exemplo, possuo um array com o nome preco\_imoveis que desejo fazer uma cópia.

preco\_imoveis = np.array([10000,120000,11000,200000])

O objetivo é colocar os valores deste array em um de nome específico para o preço de imóveis em São Paulo. Vamos fazer a cópia usando o sinal de igual (=).

preco\_imoveis\_sao\_paulo = preco\_imoveis

Se você verificar os valores na nova variável preco\_imoveis\_sao\_paulo, verá que eles são os mesmos da variável preco\_imoveis. Mas, já que "guardamos" os preços na variável preco\_imoveis\_sao\_paulo, que tal substituir o valor do primeiro imóvel do array preco\_imoveis com um novo que acabou de entrar para o nosso cadastro?

preco\_imoveis[0] = 120000

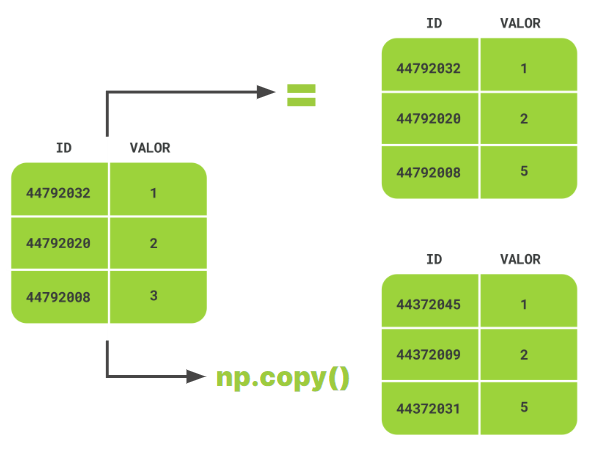
Se você tentar reproduzir o processo feito até agora, verá que essa alteração no array preco\_imoveis também alterou a primeira posição do array preco\_imoveis\_sao\_paulo para o valor de 120000.

Para criar um array desvinculado do array original existe uma função da numpy, a np.copy(). Para criar a cópia utilizando ela bastaria seguir esse código:

preco\_imoveis\_sao\_paulo = np.copy(preco\_imoveis)

Assim, mesmo que alterarmos o array preco\_imoveis, isso não afetará o array preco\_imoveis\_sao\_paulo.

Isso ocorre porque, quando usamos o sinal de igual para copiar o array, temos um novo array que aponta para os mesmos locais na memória. Já quando usamos o np.copy(), os locais na memória são diferentes.



**Para saber mais: regressão**

A regressão é muito utilizada em ciência de dados para a previsão e também a interpolação de valores. Seu uso tem diversos aspectos e eu separei alguns cursos da Alura que você pode fazer para se aprofundar no assunto.

* [Curso Online Data Analysis: previsões com Google Sheets | Alura](https://cursos.alura.com.br/course/data-analysis-previsoes-google-sheets)
* [Curso Online Regressão linear: testando relações e prevendo resultados | Alura](https://cursos.alura.com.br/course/data-science-modelo-regressao-linear)
* [Curso Online Regressão Linear: técnicas avançadas de modelagem | Alura](https://cursos.alura.com.br/course/data-science-modelo-regressao-linear-assimetria-statsmodel)
* [Curso Online Estatística com Python: Correlação e Regressão | Alura](https://cursos.alura.com.br/course/estatistica-correlacao-regressao)