**Para saber mais: removendo uma coluna**

Para removermos dados de um DataFrame, podemos utilizar o método [drop()](https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.drop.html" \t "_blank). Ele permite a remoção tanto de linhas quanto de colunas, com a utilização dos respectivos parâmetros index e columns ou do parâmetro axis que indicam os índices (linhas) pelo valor 0 e colunas pelo valor 1.

Esse método pode remover mais de um elemento de uma só vez se for passado como uma lista e gera uma cópia do DataFrame como resultado com as colunas removidas. O DataFrame original não é modificado e caso seja o desejo manter as alterações, é necessário armazenar a cópia na mesma variável ou usar o parâmetro inplace=True. É um método muito completo e útil.

Mas existe uma outra forma de remover dados com a biblioteca pandas: o método [pop()](https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.pop.html). Ele não é tão completo como o drop() e só é possível remover colunas do DataFrame e apenas uma por vez.

Além disso, diferente do método drop(), não é necessário fazer a atribuição do DataFrame na mesma variável, porque o pop() remove a coluna no DataFrame original.

O código para utilizar o método pop() é o seguinte:

dataframe.pop(nome\_coluna)

O método pop() também retorna a coluna que está sendo removida, portanto, caso a coluna seja utilizada posteriormente em alguma análise, é possível armazená-la em uma variável na mesma linha de código:

coluna\_removida = dataframe.pop(nome\_coluna)