**07 - Métodos de dicionários**

Nós aprendemos a realizar algumas operações simples com dicionários, e agora conheceremos alguns métodos que podem nos auxiliar bastante no nosso trabalho com essas estruturas de dados: update(), copy(), pop() e clear().

O método update() permite a realização de atualizações em um dicionário, como inclusões ou alterações nos dados inclusos nele. Terminamos o vídeo anterior eliminando a chave "Passat" do dicionário dados, resultando no seguinte conjunto:

dadosCOPIAR CÓDIGO

{'Crossfox': 72832.16, 'DS5': 124549.07, 'Jetta Variant': 88078.64}

Queremos que a entrada "Passat" retorne ao nosso dicionário, e faremos isso usando o update(), que receberá como parâmetro a informação {'Passat': 106161.94}.

dados.update({'Passat': 106161.94})

dadosCOPIAR CÓDIGO

Ao visualizarmos, a entrada terá sido incluída novamente.

{'Crossfox': 72832.16, 'DS5': 124549.07, 'Jetta Variant': 88078.64, 'Passat': 106161.94}

O método update() também nos permite passar mais de uma informação ou atualizarmos um dado que já está presente no dicionário. Por exemplo, iremos, ao mesmo tempo, alterar o valor de "Passat" para 106161.95 (uma mudança de um centavo) e incluir um novo veículo, "Fusca", com o valor 150000.

dados.update({'Passat': 106161.95, 'Fusca': 150000})

dadosCOPIAR CÓDIGO

{'Crossfox': 72832.16, 'DS5': 124549.07, 'Fusca': 150000, 'Jetta Variant': 88078.64, 'Passat': 106161.95}

Note que a informação foi atualizada com sucesso e o novo carro também foi incluído. O método copy() segue a mesma explicação de quando trabalhamos com listas. Se criarmos uma variável dadosCopy e atribuirmos a ela o dicionário dados, não criaremos efetivamente uma cópia, mas sim um novo rótulo para o mesmo espaço na memória.

dadosCopy = dadosCOPIAR CÓDIGO

De modo a realmente gerarmos uma cópia na qual conseguiremos fazer alterações sem afetar o dicionário original, podemos utilizar o copy().

dadosCopy = dados.copy()

dadosCopyCOPIAR CÓDIGO

{'Crossfox': 72832.16, 'DS5': 124549.07, 'Fusca': 150000, 'Jetta Variant': 88078.64, 'Passat': 106161.95}

Para testarmos, eliminaremos a entrada "Fusca" de dadosCopy.

del dadosCopy['Fusca']

dadosCopyCOPIAR CÓDIGO

{'Crossfox': 72832.16, 'DS5': 124549.07, 'Jetta Variant': 88078.64, 'Passat': 106161.95}

Comparando-o com o dicionário dados, perceberemos que o original continou intacto.

dadosCOPIAR CÓDIGO

{'Crossfox': 72832.16, 'DS5': 124549.07, 'Fusca': 150000, 'Jetta Variant': 88078.64, 'Passat': 106161.95}

O método pop() elimina uma chave do dicionário e retorna o seu valor. Existem algumas especificações resumidas no texto deixado em nosso notebook, mas trabalharemos cada ponto com calma.

Se a chave for encontrada no dicionário, o item é removido e seu valor é retornado. Caso contrário, o valor especificado como default é retornado. Se o valor default não for fornecido e a chave não for encontrada no dicionário um erro será gerado.

Começaremos nossos testes usando o dicionário dadosCopy, a partir do qual chamaremos o método pop() passando como parâmetro a entrada "Passat".

dadosCopy.pop('Passat')COPIAR CÓDIGO

106161.95

Repare que obtivemos como retorno o valor atribuído à chave que foi eliminada. Se visualizarmos o dicionário, perceberemos que o item foi removido com sucesso.

dadosCopyCOPIAR CÓDIGO

{'Crossfox': 72832.16, 'DS5': 124549.07, 'Jetta Variant': 88078.64}

Se executarmos dadosCopy.pop('Passat') novamente, receberemos um erro, afinal a chave não será encontrada no dicionário.

dadosCopy.pop('Passat')COPIAR CÓDIGO

KeyError Traceback (most recent call last) ipython-input-24-b12fd0f355e0 in module() ----> 1 dadosCopy.pop('Passat') KeyError: 'Passat'

O método copy() pode receber um parâmetro "*default*" que permite uma espécie de tratamento desse erro. Ou seja, se a chave não for encontrada no dicionário, nosso retorno será o valor passado como *default*. Em nosso exemplo, além de "Passat", passaremos a mensagem "Chave não encontrada".

dadosCopy.pop('Passat', 'Chave não encontrada')COPIAR CÓDIGO

Ao executarmos, como a chave não mais existe no dicionário, receberemos justamente a mensagem que preparamos.

Chave não encontrada

Já se passarmos um veículo que ainda consta no dicionário, é o seu valor que nos será devolvido.

dadosCopy.pop('DS5', 'Chave não encontrada')COPIAR CÓDIGO

124549.07

Como resultado, teremos um dicionário com somente duas informações.

dadosCopyCOPIAR CÓDIGO

{'Crossfox': 72832.16, 'Jetta Variant': 88078.64}

Para finalizar, conheceremos o método clear(), que simplesmente limpa todo o dicionário. Faremos nosso teste com o dicionário dadosCopy.

dadosCopy.clear()

dadosCopyCOPIAR CÓDIGO

Como retorno, teremos um dicionário vazio.

{}

Existem outros métodos que podem ser encontrados na [documentação a respeito de dicionários em Python](https://docs.python.org/3.6/library/stdtypes.html#typesmapping), e que poderão lhe auxiliar em diversas situações durante seu trabalho como cientista de dados. No próximo vídeo conversaremos sobre iteração em dicionários.