**02 - Criando listas**

Já aprendemos a criar variáveis e com quais tipos de dados o Python trabalha, como os numéricos *int* e *float* (inteiros e pontos flutuantes, que representam números reais), o *str* (abreviação de string e que lida com textos) e *bool* (abreviação de booleano, representando *true* ou *false*). Nas aulas anteriores, criamos variáveis atribuindo a elas valores únicos, como nos exemplos abaixo:

ano atual = 2019

km = 4000COPIAR CÓDIGO

Em data science, costumamos trabalhar com grandes conjuntos de informações. Nesse curso, por exemplo, temos um *dataset* com diversos carros, cada um com seu ano de fabricação, quilometragem, valor e assim por diante. Se criássemos uma variável para cada um desses dados, teríamos um programa muito extenso e inviável, sem contar com a possibilidade de erros.

A solução para esse problema é a utilização das **listas** do Python, sequências mutáveis que servem para armazenar coleções de itens normalmente homogêneos, o que faz mais sentido, mas não necessariamente homogêneos. Na seção **3. TRABALHANDO COM LISTAS** do nosso notebook, temos um breve texto mostrando as formas de criação de uma lista, incluindo algumas células já preenchidas para não tomarmos muito tempo digitando esses dados.

Nosso primeiro exemplo é uma lista com diversos acessórios de um veículo. Para criá-la, abrimos e fechamos colchetes ([]) e passamos os itens separados por vírgulas.

Acessorios = ['Rodas de liga', 'Travas elétricas', 'Piloto automático', 'Bancos de couro', 'Ar condicionado', 'Sensor de estacionamento', 'Sensor crepuscular', 'Sensor de chuva']

AcessoriosCOPIAR CÓDIGO

Ao executarmos, recebemos a lista, também indicada por colchetes, contendo os itens dentro dela.

['Rodas de liga', 'Travas elétricas', 'Piloto automático', 'Bancos de couro', 'Ar condicionado', 'Sensor de estacionamento', 'Sensor crepuscular', 'Sensor de chuva']

Ao usarmos a função type() para verificar o tipo da variável Acessorios, receberemos como retorno list, que é o tipo de lista do Python. Como citado anteriormente, as listas são preferencialmente homogêneas de modo a termos uma organização, mas podem conter valores de tipos variados, como strings, numéricos ou booleanos. Observe o exemplo abaixo:

Carro\_1 = ['Jetta Variant', 'Motor 4.0 Turbo', 2003, 44410.0, False, ['Rodas de liga', 'Travas elétricas', 'Piloto automático'], 88078.64]

Carro\_2 = ['Passat', 'Motor Diesel', 1991, 5712.0, False, ['Central multimídia', 'Teto panorâmico', 'Freios ABS'], 106161.94]COPIAR CÓDIGO

Aqui temos uma lista com todas as informações do Carro\_1, como nome, motorização, ano de fabricação, quilometragem e assim por diante. Repare que essa lista contém também outra lista com os acessórios do veículo, demonstrando quão versátil é essa funcionalidade. NO futuro aprenderemos a acessar os itens de uma lista, inclusive aqueles contidos em uma sublista.

Ao executarmos esse código, poderemos visualizar as informações de cada um dos carros:

Carro\_1COPIAR CÓDIGO

['Jetta Variant', 'Motor 4.0 Turbo', 2003, 44410.0, False, ['Rodas de liga', 'Travas elétricas', 'Piloto automático'], 88078.64]

Carro\_2COPIAR CÓDIGO

['Passat', 'Motor Diesel', 1991, 5712.0, False, ['Central multimídia', 'Teto panorâmico', 'Freios ABS'], 106161.94]

Assim como é possível termos uma lista dentro de outra lista, também podemos criar uma lista contendo variáveis que armazenam outras listas, como no exemplo:

Carros = [Carro\_1, Carro\_2]

CarrosCOPIAR CÓDIGO

[['Jetta Variant', 'Motor 4.0 Turbo', 2003, 44410.0, False, ['Rodas de liga', 'Travas elétricas', 'Piloto automático'], 88078.64], ['Passat', 'Motor Diesel', 1991, 5712.0, False, ['Central multimídia', 'Teto panorâmico', 'Freios ABS'], 106161.94]]

Veja que logo de cara abrimos dois colchetes, já que a informação de cada carro é também uma lista em si. Ou seja, estamos trabalhando com uma lista de listas (listas estas que também possuem outras listas!). No próximo vídeo conheceremos algumas operações e métodos que nos permitem modificar as listas.