

## Para saber mais: inicializando uma lista de números pares

O recurso *List Comprehension* também permite utilizar condições para o preenchimento da lista. Considere o objetivo de inicializar uma lista com os números pares a partir de uma lista de números inteiros qualquer, por exemplo, os números 1,3,4,5,7,8,9. Para descobrir se um número é par, usamos a condição `numero%2 == 0`, que verifica se o resto de uma divisão por 2 é zero. O código abaixo usa um loop para inicializar a lista de pares.

```
inteiros = [1,3,4,5,7,8,9]
pares = []
for numero in inteiros:
    if numero % 2 == 0:
        pares.append(numero)
```

[COPIAR CÓDIGO](#)

Pesquise como o podemos usar o *List Comprehension* para fazer o mesmo que o código acima.

### Opinião do instrutor

Dado o código de geração da lista de pares abaixo:

```
inteiros = [1,3,4,5,7,8,9]
pares = []
for numero in inteiros:
```



```
if numero % 2 == 0:  
    pares.append(numero)
```

[COPIAR CÓDIGO](#)

O código usando *List Comprehension* relativo ficaria muito mais enxuto:

```
inteiros = [1,3,4,5,7,8,9]  
pares = [x for x in inteiros if x % 2 == 0]
```

[COPIAR CÓDIGO](#)

Repare o `if` depois do `for` que define a condição! Muito melhor não?

*List Comprehension* é um dos recursos mais legais da linguagem Python :)