

ASIMOV

range()

Nesta palestra curta estaremos discutindo a função `range()`. Ainda não desenvolvemos um nível muito profundo de conhecimento de funções, mas podemos entender o básico desta função simples (mas extremamente útil!).

`range()` nos permite criar uma lista de números que variam de um ponto de partida *até* um ponto final. Também podemos especificar o tamanho do passo. Vamos percorrer alguns exemplos:

```
In [1]: range(0,10)
```

```
Out[1]: range(0, 10)
```

```
In [2]: x = range(0,10)
        type(x)
```

```
Out[2]: range
```

```
In [3]: start = 0 # Por padrão
        stop = 20
        x = range(start,stop)
```

```
In [5]: list(x)
```

```
Out[5]: [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19]
```

Ótimo! Observe como foi *até* 20, mas na verdade não produz 20. Assim como na indexação. E o tamanho do passo? Podemos especificar isso como um terceiro argumento:

```
In [3]: x = range(start,stop,2)
        # Mostrar
        x
```

```
Out[3]: range(0, 20, 2)
```

Impressionante! Bem, é isso ... ou não?

Você pode estar se perguntando, o que acontece se eu quiser usar uma grande variedade de números? O meu computador pode armazenar tudo na memória?

Grande pensamento! Este é um dilema que pode ser resolvido com o uso de um gerador. Para uma explicação simplificada: um gerador permite a geração de objetos gerados que são fornecidos naquela instância, mas não armazena cada instância gerada na memória.

Isso significa que um gerador não criaria uma lista ao gerar um `range()`, mas, em vez disso, fornece uma geração única dos números nesse intervalo. A boa notícia é que `range()` se comporta como um gerador e você não precisa se preocupar com isso.