

for e *range* em Python

Prof. Antonio Carlos Mariani

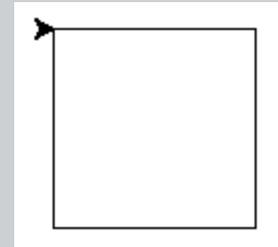
INE | CTC



UNIVERSIDADE FEDERAL
DE SANTA CATARINA

Desenhando um quadrado

Código Python para desenhar um quadrado:

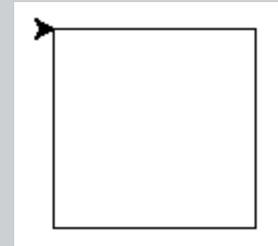


```
quadrado.py x
1 # Importar o módulo "turtle"
2 import turtle
3
4 # Criar uma tartaruga e referenciá-la pelo nome "zezinho"
5 zezinho = turtle.Turtle()
6
7 # Desenhar um quadrado
8 zezinho.forward(100)
9 zezinho.right(90)
10 zezinho.forward(100)
11 zezinho.right(90)
12 zezinho.forward(100)
13 zezinho.right(90)
14 zezinho.forward(100)
15 zezinho.right(90)
16
17 # Indicar que a tarefa foi concluída
18 turtle.done()
```

Note que as linhas 8 e 9 aparecem repetidas 4 vezes. Em vez de copiar e colar estas duas linhas quatro vezes, podemos reescrever o código de forma mais sintética conforme mostrado a seguir.

Desenhando um quadrado

Código alternativo com estrutura de repetição “for”:



```
quadrado.py x
1 # Importar o módulo "turtle"
2 import turtle
3
4 # Criar uma tartaruga e referenciá-la pelo nome "zezinho"
5 zezinho = turtle.Turtle()
6
7 # Desenhar um quadrado
8 for i in range(4):
9     zezinho.forward(100)
10    zezinho.right(90)
11
12 # Indicar que a tarefa foi concluída
13 turtle.done()
```

O comando “for” em conjunto com a função “range” (linha 8) faz com que as linhas 9 e 10 sejam repetidas 4 vezes (4 iterações).

Note que as linhas 9 e 10 devem ser deslocadas para a direita (“tab”) de forma a indicar que estão sujeitas à repetição.

A função “range”

A função “range” em Python gera uma sequência de número inteiros na forma de uma progressão aritmética (PA). Ela pode ser utilizada de duas formas:

1. **range(fim)**
2. **range(início, fim [, passo])**

Na primeira forma o “início” é sempre 0 (zero). Já na segunda o “passo” é opcional (por isso estar entre colchetes), sendo utilizado o valor 1 (um) em caso de omissão (ausência).

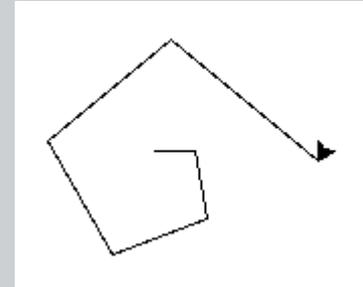
Exemplos:

Expressão	Sequência correspondente	Equivalência
range(5)	[0, 1, 2, 3, 4]	range(0, 5, 1)
range(0, 5)	[0, 1, 2, 3, 4]	range(0, 5, 1)
range(10, 14)	[10, 11, 12, 13]	range(10, 14, 1)
range(2, 20, 3)	[2, 5, 8, 11, 14, 17]	
range(10, 0, -2)	[10, 8, 6, 4, 2]	

Importante: note que o “fim” é exclusivo, ou seja, a sequência vai até o valor imediatamente anterior a ele (nunca o inclui).

O comando “for”

O código abaixo produz a imagem ao lado.



```
7 for distancia in range(20, 100, 15):  
8     zezinho.forward(distancia)  
9     zezinho.right(80)  
10
```

A função “range(20, 100, 15)” implica a sequência: [20, 35, 50, 65, 80, 95]

O comando “for”, por sua vez, faz com que a variável “distancia” assuma cada um desses valores (nesta ordem, um a um) e para cada valor execute as linhas 8 e 9. Este código é equivalente, portanto, a:

```
7 zezinho.forward(20)  
8 zezinho.right(80)  
9 zezinho.forward(35)  
10 zezinho.right(80)  
11 zezinho.forward(50)  
12 zezinho.right(80)  
13 zezinho.forward(65)  
14 zezinho.right(80)  
15 zezinho.forward(90)  
16 zezinho.right(80)  
17 zezinho.forward(95)  
18 zezinho.right(80)
```

Neste caso a execução do comando “for” implica a realização de 6 iterações.

Contato

antonio.c.mariani@ufsc.br

CTC/INE



UNIVERSIDADE FEDERAL
DE SANTA CATARINA