PYTHON ACADEMY

A FUNÇÃO RANGE DO PYTHON

Vamos tratar de um assunto muito importante nesse ebook: como construir loops utilizando a função range do Python!

PYTHONACADEMY.COM.BR

Olá, olá Pythonista!

Vamos tratar de um assunto muito importante nesse post: como construir *loops* utilizando a função `range` do Python!

Ainda não está familiarizado com `for`, `while` e *loops* do Python?

Então já acessa nosso post completo sobre Estruturas de Repetição no Python clicando aqui!

Depois, **vamos nessa**!

A função `range()`

Essa função é de grande ajuda quando o tema é repetição, laços, `for` etc.

Ela permite especificar o inicio de uma sequência, o passo (ou pulo) e valor final.

Com isso, o Python nos retorna uma sequência de números para que possamos iterar!

Sua sintaxe pode ter as seguintes três formas, sendo que seu único parâmetro obrigatório é o `fim`:

```
range(fim)                  # range(stop)
range(inicio, fim)          # range(start, stop)
range(inicio, fim, passo)   # range(start, stop, step)
```

É importante ressaltar que na versão 3.x do Python, a função `range()` retorna um objeto iterável e não mais uma lista com elementos, por essa razão devemos converter o retorno para listas com a função `list()`.

Veja alguns exemplos de como criar listas com range:

```
# Gerar lista com (fim)
print(list(range(5)))

# Gerar com (inicio, fim)
print(list(range(5, 10)))

# Gerar com (inicio, fim, passo)
print(list(range(0, 10, 2)))
```

Resultando do código acima:

```
[0, 1, 2, 3, 4]
[5, 6, 7, 8, 9]
[0, 2, 4, 6, 8]
```

Utilizando `range`

Sabendo que a estrutura de repetição `for` executa um ciclo para cada elemento de um iterável, e a função `range` é um iterável, podemos criar uma harmonia perfeita entre esses dois:

```
for num in range(4):
    print(num)
```

O código acima irá imprimir a sequência de 4 itens especificado no range:

```
0
1
2
3
```

O `range` não se aplica diretamente ao `while`, porém um exemplo um pouco mais complexo pode demonstrar eles juntos:

```
lista = list(range(6))
comprimento = 6

while len(lista) == comprimento:
    print(lista)
    lista.append('item')
else:
    print(f'O comprimento da Lista é {len(lista)} e ultrapassou {com-
        primento}!')
    print(lista)
```

Saída:

```
[0, 1, 2, 3, 4, 5]
O comprimento da Lista é 7 e ultrapassou 6!
[0, 1, 2, 3, 4, 5, 'item']
```

Explicando: basicamente ele verifica se o tamanho da lista é igual a 6 itens. Após a primeira verificação ele adiciona mais um item (`item`), assim modificando seu comprimento e encerrando o laço!

> 💡 *Estou desenvolvendo o **DevBook**, uma plataforma que usa IA para gerar ebooks técnicos profissionais. Te convido a conhecer!*

Conclusão

Nesse post vimos uma função muito importante para geração de sequência de números em Python: a famosa `range()`!

Aprendê-los bem é básico para todos desenvolvedor Python.

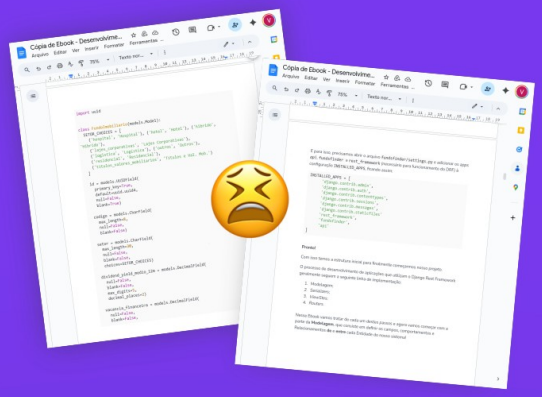
Se ficou com alguma dúvida, fique à vontade para deixar um comentário no box aqui embaixo! Será um prazer te responder! 😉

Não se esqueça de conferir!

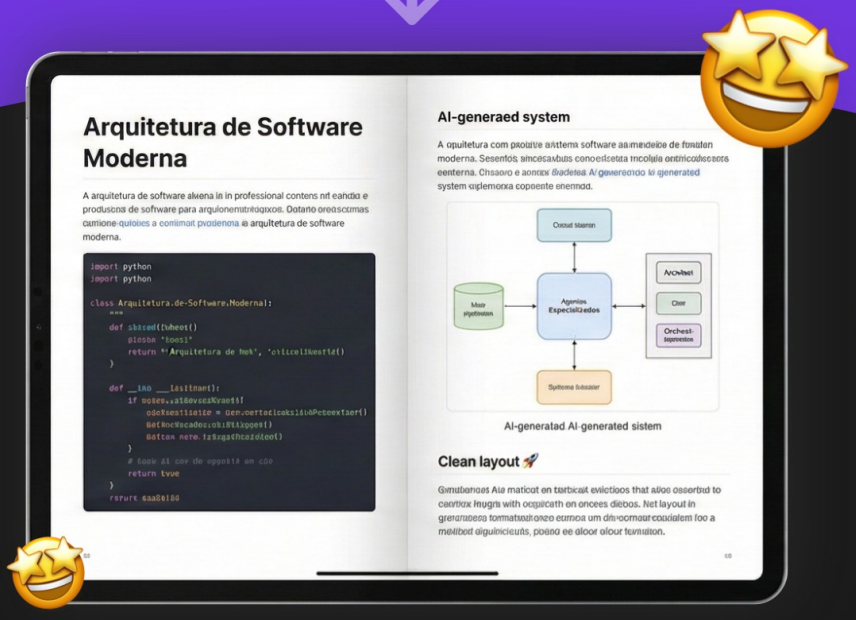
DevBook

Crie Ebooks técnicos em minutos com IA

Conheça a 1ª IA Especializada na criação de Ebooks **com código**!



Chega de formatar código no Google Docs



Syntax Highlight

Infográficos feitos por IA

Adicione Banners Promocionais

Deixe que nossa IA faça o trabalho pesado

Edite em Markdown em Tempo Real

TESTE AGORA

PRIMEIRO CAPÍTULO 100% GRÁTIS