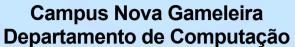
CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA DE MINAS GERAIS



Laboratório de Programação de Computadores I

Aula 4 - Estrutura de Repetição - for-in

Prof. Me. Júnio Soares Dias E-mail: juniosdias@cefetmq.br

Engenharia (2025).

Curiosidade: Por que Python? python





Sabemos que: "A programação é uma atividade complexa, que envolve conhecimento, prática e proficiência."

Mas a palavra-chave de Python é: SIMPLICIDADE

(...) Dentre todas as características dessa linguagem de programação, sem sombra de dúvidas, a simplicidade é a que mais chama a atenção. Basta você tê-la instalado em seu sistema e chamá-la no console que já pode sair usando!

Python nasceu com esse objetivo: ser simples, acessível e fácil de usar. Entretanto, não podemos atribuir o sucesso da linguagem a um único fator. A tecnologia é muito mais que simplicidade. Estamos falando de uma linguagem incrível e que veio para ficar! (...)

Fonte: Post da Bycode (2025), disponível em: https://becode.com.br/porque-aprender-python/#google_vignette, acesso em: 20 abr 2025

Laboratório de Programação de Computadores I - Engenharia (2025), Prof. Júnio S. Dias.

Slide 2

Conteúdo da aula

Esta aula vai praticar:

Estrutura de Repetição - for-in

Slide 3

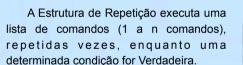
Codificação em Python: range, for-in, else.

Laboratório de Programação de Computadores I - Engenharia (2025), Prof. Júnio S. Dias.

Fontes: Downey (2016), Python

Docs (2025) e Sebesta (2018).

Revisão pela prática



O comando "for-in" é utilizado para testar a condição antes da execução dos comandos.

· Pode ser útil quando o número de repetições necessárias é fixo, como uma lista (i.e., número de iterações é conhecido).

🖢 puthon



CEFET-MG - LPC1 - Código Python

Exemplo da Estrutura de Repetição: for-in

lists = range(10) soma=0

print("n=")

for n in lists:

Saída: n: 0.1, 2, 3, 4, 5, 6, 7, 8, 9,

print(n,", ") soma+=n

print("Soma=", soma) #Saída; Soma=45

Laboratório de Programação de Computadores I - Engenharia (2025), Prof. Júnio S. Dias.

Slide 4

Range

🔑 python™

- · A função range tem como objetivo produzir um dado iterável a partir de 1, 2 ou 3 argumentos que recebe.
 - Exemplo: "range(x, y, z)".
- · Na função range os argumentos são os números inteiros que determinarão os elementos do iterável que será gerado, esses são: valor inicial(x), valor de parada(y) e incremento(z).

Gera: 0 1 2 3 4 5 6 7 8 9

CEFET-MG - Aula 4 LPC1 - Código Python

Exemplos da função range = número iterável

range(2,6)

range(10)

Gera: 2345

range(3,15,2)

Gera: 3, 5, 7, 9, 11, 13

Fonte: Python Docs (2025).

CEFET-MG - LPC1 - Código Python # Exemplo da Estrutura de Repetição for-in - else

print("Encontrei o número 4!")

print("O número 4 não foi encontrado na lista.")

print("O else garante execução de instruções do bloco. i=0.")

break # Interrompe o loop

for-in com uso de dado iterável

print("Imprime impares menores que 10=")

lista = [1, 2, 3, 5, 5]

for numero in lista

if numero == 4

Laboratório de Programação de Computadores I - Engenharia (2025), Prof. Júnio S. Dias.

Slide 5

Estrutura de Repetição - for-in



Essa estrutura é utilizada quando se

Os comandos serão executados.

utilizando, por exemplos, um dado

iterável uma lista, e seu conteúdo vai

variar do "valor inicial" até o "valor final"

Dado em formato de lista:

list(range(10))=[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]

sabe o número de vezes em que um

trecho do programa deve ser repetido.



CEFET-MG - LPC1 - Código Python # Exemplo da Estrutura de Repetição for-in

Uso de range para gerar uma lista lists = list(range(10)) print(type(lists)) soma=0

for-in com uso de lista gerada por range. for i in range(1000): if i%2==0: soma+=i

print("Soma pares menores que 1000=",soma) # for-in com uso de dado iterável

print("Imprime impares menores que 10=")

for i in range(10): if i%2!=0: print(i)

Fontes: Downey (2016) e Python Docs (2025).

Laboratório de Programação de Computadores I - Engenharia (2025), Prof. Júnio S. Dias.

Slide 6

Estrutura de Repetição for-in - else





Slide 7

O comando "for-in" associado ao "else" permite a execução de instruções deste bloco mesmo se não houver um ciclo no loop.

Já o comando "for-in" associado ao "break" e "else" apresentam mais uma opção de filtro para buscas no loop.

Fonte: Python Docs (2025).

for i in range(0)

if i%2!=0

Laboratório de Programação de Computadores I - Engenharia (2025), Prof. Júnio S. Dias.

Estrutura de Repetição for-in - black e pass





CEFET-MG - LPC1 - Código Python # Exemplo da Estrutura de Renetição for-in - break - pass

Uso de range para gerar uma lista lists = list(range(10)) print(type(lists))

for-in com uso de lista gerada por range for i in range(1000): if i<110.

soma+=i break #continue

print("Flse, I=".i. " soma=".soma)

print("Fora do bloco, i=",i, " soma=",soma)

for-in com uso de dado iterável print("Executa for-in sem instrução - pass") for i in range(10): pass

print("O else garante execução de uma iteração. i=0.")

Fonte: Python Docs (2025).

else.

Curiosidades:

desse dado.

O "break" inserido dentro de um else no loop encerra a execução do loop, inclusive a do "else" fora do loop, e passa para o próximo bloco de instrução.

Já o comando "continue" interrompe o passo do loop, mas continua o loop e as instruções do else são executadas.

O pass permite o "for-in" sem instrução, mas as instruções do "else" são executadas.

Laboratório de Programação de Computadores I - Engenharia (2025), Prof. Júnio S. Dias.

Slide 8

Prática 4



• Abra o arquivo com as instruções da prática 4 e execute as instruções apresentadas. O arquivo encontra-se no SIGAA.

Entrega da Prática

• Faça a entrega da prática 4 até a próxima aula no SIGAA, verifique o período de submissão no questionário.

Laboratório de Programação de Computadores I - Engenharia (2025), Prof. Júnio S. Dias.

Slide 9

Conteúdo Complementar



• Assista ao vídeo sobre o uso da Estrutura de Repetição - "for-in" no Python do "Curso do Instituto de Computação IC/UFRJ", analise a sua aplicação no contexto do seu curso:

https://www.youtube.com/watch?v=gi6G4xyXx0Q



Laboratório de Programação de Computadores I - Engenharia (2025), Prof. Júnio S. Dias.

Slide 10

Dúvidas



Slide 11

Encaminhe suas dúvidas para o e-mail:

juniosdias@cefetmg.br



Laboratório de Programação de Computadores I - Engenharia (2025), Prof. Júnio S. Dias.

Referências



DOWNEY, Allen B. **Pense em Python**. 2. Ed. Editora Novatec, 2016. Disponível em: https://penseallen.github.io/PensePython2e/. Acesso em: 15 abr 2025.

PYTHON DOCS. **Python Software Fundation**. Disponível em: https://www.python.org/doc/, acesso em: 29 mar 2025.

SEBESTA, R. W. Conceitos de Linguagens de Programação. 11. ed. Bookman Editora, 2018.

Laboratório de Programação de Computadores I - Engenharia (2025), Prof. Júnio S. Dias.

Slide 12