




Paradigmas de linguagens de programação em python

Professores:

Sebastião Rogério feat. Kayo Monteiro



00

For

Estrutura de Repetição – FOR

O laço for nos permite percorrer os itens de uma coleção e, para cada um deles, executar o bloco de código declarado no loop. Sua sintaxe é a seguinte:



Estrutura de Repetição – FOR

O laço for nos permite percorrer os itens de uma coleção e, para cada um deles, executar o bloco de código declarado no loop. Sua sintaxe é a seguinte:



```
for variavel in lista  
comandos
```

Estrutura de Repetição – FOR

Enquanto percorremos a lista de valores, a variável indicada no for receberá, a cada iteração, um item da coleção. Assim, podemos executar algum processamento com esse elemento. No código abaixo percorremos a lista nomes e imprimimos cada elemento.



```
nomes = ['Pedro', 'João', 'Leticia']  
for n in nomes:  
    print(n)
```

Estrutura de Repetição – FOR



```
nomes = ['Pedro', 'João', 'Leticia']  
for n in nomes:  
    print(n)
```

Estrutura de Repetição – FOR ELSE



```
nomes = ['Pedro', 'João', 'Leticia']  
for n in nomes:  
    print(n)  
else:  
    print("Todos os nomes foram listados com sucesso")
```

Estrutura de Repetição – FOR ELSE

Na primeira linha definimos uma variável que armazenará uma lista de nomes. Após isso, a instrução for percorre todos esses elementos e atribui um a um à variável n, que será impressa, como pode ser visto na linha 3. Após o loop se encerrar, o bloco de código contido na instrução else é acionado, imprimindo a mensagem na tela.

```
nomes = ['Pedro', 'João', 'Leticia']  
for n in nomes:  
    print(n)  
else:  
    print("Todos os nomes foram listados com sucesso")
```


Exercicio

Crie um programa que solicita ao usuário uma palavra e utiliza um loop for para verificar se essa palavra é um palíndromo (ou seja, se pode ser lida da mesma forma de trás para frente). Utilize a estrutura for-else para verificar se todos os caracteres correspondentes são iguais.

Exercicio – Solução



```
# Solicitando uma palavra ao usuário
palavra = input("Digite uma palavra: ")

# Percorrendo a palavra até a metade
for i in range(len(palavra) // 2):
    # Comparando a letra com a correspondente no final da
    # palavra
    if palavra[i] != palavra[-(i + 1)]:
        print(f"A palavra {palavra} não é um palíndromo.")
        break
else:
    print(f"A palavra {palavra} é um palíndromo.")
```

Explicação Detalhada:

Iteração até a metade da palavra: O loop for percorre até a metade da palavra. Isso é feito com `range(len(palavra) // 2)`, já que não é necessário comparar além da metade (por exemplo, se a primeira metade for igual à segunda, a palavra é palíndroma).

Comparação de letras: Para cada índice `i`, a letra `palavra[i]` (da parte inicial) é comparada com `palavra[-(i + 1)]` (da parte final).

Exemplo: se `i = 0`, a primeira letra será comparada com a última (`-1`); se `i = 1`, a segunda será comparada com a penúltima (`-2`), e assim por diante.

Interrupção com break: Se qualquer letra não coincidir, o `break` é acionado, interrompendo o loop, e o programa imprime que a palavra não é um palíndromo.

Uso do else: O bloco `else` será executado apenas se o `for` concluir todas as comparações sem acionar o `break`, ou seja, quando todas as letras coincidirem e a palavra for um palíndromo.