

Facy Wyden
Campus Batista Campos

Professor/Coordenador: Alexandre Abreu de Freitas

Aluno: Lucas Vinicius Silveira (202402390562).

Tema: Pesquisa científica sobre escopo estático e dinâmico.

Belém / Pará

2024

1. Introdução

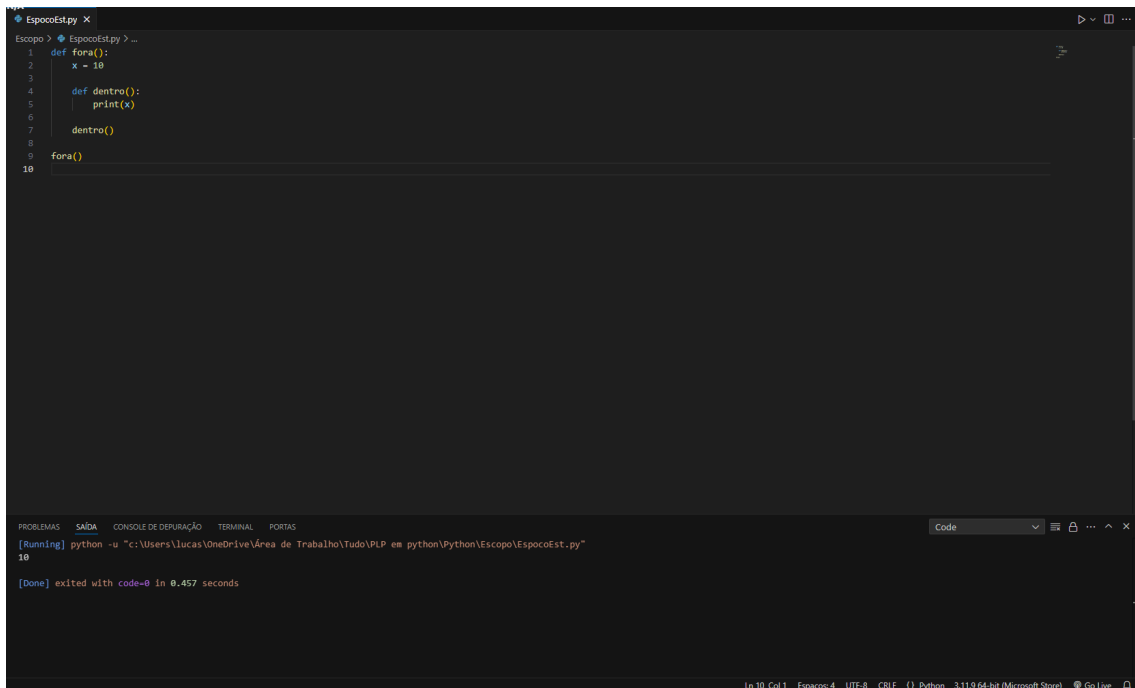
O escopo de uma variável em um programa define onde ela pode ser acessada. Existem dois principais tipos de escopo que determinam a forma como as variáveis são resolvidas dentro das funções: **escopo estático** (ou léxico) e **escopo dinâmico**. Este relatório tem como objetivo explicar a diferença entre esses dois tipos de escopo, com exemplos práticos em Python.

2. Escopo estático

2.1 Definição

O **escopo estático** resolve variáveis com base na organização estrutural do código, ou seja, onde as variáveis são declaradas em relação ao ponto em que a função é **definida**, e não no ponto em que ela é chamada. Esse tipo de escopo é usado pela maioria das linguagens modernas, incluindo Python.

2.2 Exemplo



```
1 def fora():
2     x = 10
3
4     def dentro():
5         print(x)
6
7     dentro()
8
9 fora()
10
```

PROBLEMAS SAÍDA CONSOLE DE DEBURAÇÃO TERMINAL PORTAS

[Running] python -u "c:\Users\lucas\OneDrive\Área de Trabalho\Tudo\PLP em python\Python\Escopo\EspocoEst.py"

10

[Done] exited with code=0 in 0.457 seconds

Ln 10, Col 1 Espaços: 4 UTF-8 CRLF () Python 3.11.9 64-bit (Microsoft Store) Go Live

Explicação:

1. Quando a função `dentro()` é chamada ela procura pela variável `x`.
2. Como `x` não está dentro de `dentro()`, ela busca no escopo em que ela foi definida, que é o escopo da função `fora()`.
3. A variável `x` é encontrada e seu valor é impresso.

Este comportamento ocorre porque o Python usa o escopo estático, então a função interna (`dentro`) têm acesso às variáveis do escopo em que foi criada, não do escopo em que é chamada.

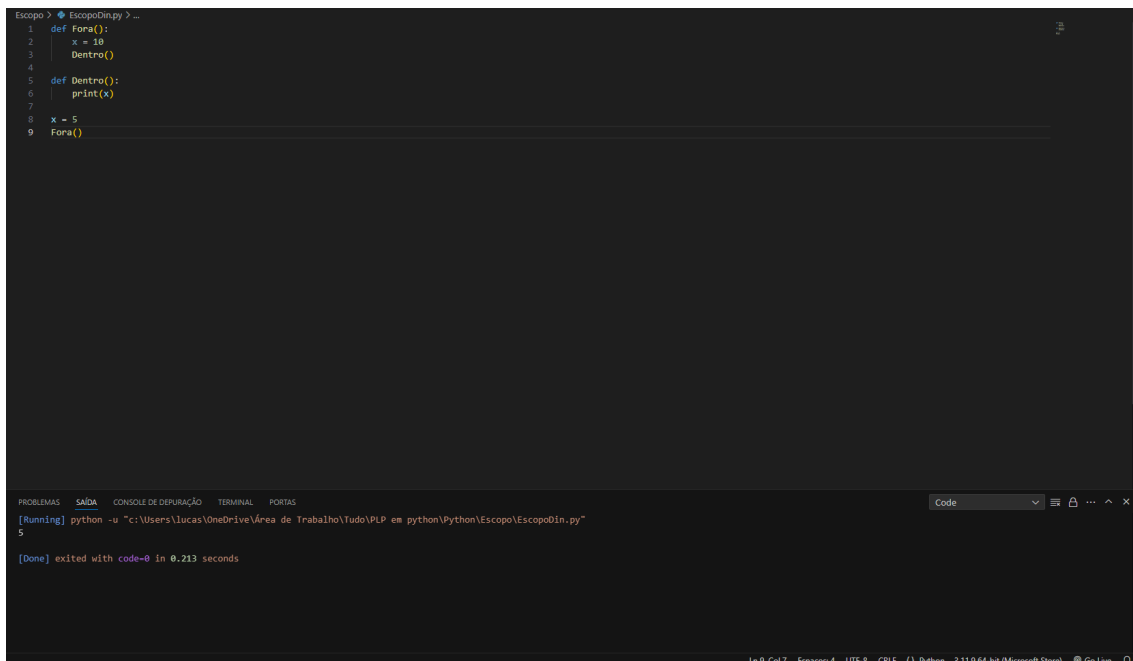
3. Escopo dinâmico

3.1 Definição

O **escopo dinâmico** funciona de forma diferente do escopo estático. Nele, as variáveis são encontradas com base no momento em que a função é **chamada**, e não onde ela foi definida. Isso significa que a função vai procurar as variáveis nas funções que estão **ativas** no momento da execução, ou seja, na pilha de chamadas.

O **escopo dinâmico** resolve variáveis com base no momento da **chamada** da função, não em seu ponto de definição. Neste caso, quando uma função é chamada, o interpretador verifica as variáveis no escopo das funções que estão na pilha de chamadas no momento da execução.

3.1 Exemplo



```
Escopo > EscopoDin.py > ...
1 def Fora():
2     x = 10
3     Dentro()
4
5 def Dentro():
6     print(x)
7
8 x = 5
9 Fora()
```

PROBLEMAS SAÍDA CONSOLE DE DEPURACÃO TERMINAL PORTAS

[Running] python -u "c:\Users\lucas\OneDrive\Área de Trabalho\Tudo\PLP em python\Python\Escopo\EscopoDin.py"

5

[Done] exited with code=0 in 0.213 seconds

Ln 9, Col 7 Espaço: 4 UTF-8 CRLF Python 3.11.9 64-bit (Microsoft Store) Go Live

Explicação:

Caso python usasse o escopo dinâmico:

1. Quando **Fora()** chama **Dentro()**, o Python tentaria encontrar a variável **x**.
2. Como **Dentro()** foi chamada por **Fora()**, o Python procuraria **x** em **Fora()**, encontrando **x = 10**.
3. **Saída: 10**.