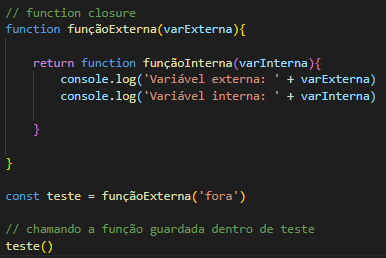
**Function Enclosures**

Variáveis globais duram até a página ser recerregada ou sair da página, mas toda variável dentro de uma função é destruída toda vez que a função terminar de ser executada.

Enclosures é quando temos, dentro de uma função, uma **variável privada**, que está dentro de outra função

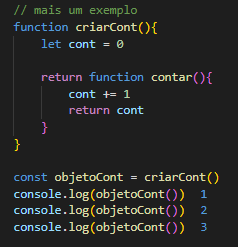
**Ex. 1:** no caso abaixo, o valor do parÂmetro da funçãoExterna foi guardado dentro da funçãoIterna, que por sua vez foi guardada dentro da variável ‘teste’.

Quando a variável ‘teste’ foi chamada, perceba que o parâmetro anteriormente passado para funçãoExterna – a string ‘fora’ – foi mantido na funçãoInterna, mesmo que a funçãoExterna já foi executada.





**Ex. 2:** criando uma função contadora que, cada vez que é chamada, a variável cont é acrescida de 1 unidade. Perceba que a função guardada em objetoCont – função contar() – tem a variável cont, que veio da função criarCont(), como se fosse um atributo



**Ex. 2:** perceba que **x** vai estar definido dentro de teste() e nas funções abaixo dela, assim como **y** vai estar defninido em teste2() e em todas as funções abaixo dela e, por fim, **z** está definido apenas em teste3().

Isso porque usamos **let**: a variável terá escopo local.

Y não está definido em teste(), assim como z não está defnido em teste3() por causa da declaração let.

