

Closure

Relaciona-se ao conceito de **Escopo**. Escopo refere-se ao "mundo" em que uma função "vive" e, portanto, consegue enxergar seus elementos.

```
1  // Este arquivo representa o Estado
2  const x = 3
3
4  function cidade() {
5      const x = 97
6      function bairro(n) {
7          return x + n
8      }
9      return bairro
10 }
11
12 // Não funciona: bairro está fora do escopo Estado
13 //console.log(bairro(5))
14
15 // Funciona: cidade está dentro do escopo Estado
16 console.log(cidade()(5))
17
18 module.exports = cidade()
```



Closure é quando uma função **"lembra"** de seu **escopo léxico**, ou seja, o mundo em que ela foi definida/escrita. Logo, mesmo que a função seja executada fora desse escopo léxico, ela irá priorizar suas "origens"

```
1  // Arquivo que representa um novo Estado
2
3  const bairro = require('./principios10_closureEscopo')
4
5  const x = 1000
6
7  // O valor de x acima não influencia
8  // porque o bairro "lembra" de suas origens
9  console.log(bairro(10))
```

