

ESCOPO DE FUNÇÃO

Variáveis declaradas dentro de funções não são acessadas fora das mesmas.

Escopo evita conflitos

```
1 function mostrarCarro() {
2   var carro = 'Fusca';
3   console.log(carro);
4 }
5
6 mostrarCarro(); // Fusca no console
7 console.log(carro); // Erro, carro is not defined
```

VARIÁVEL GLOBAL (ERRO)

Declarar variáveis sem a palavra chave `var`, `const` ou `let`, cria uma variável que pode ser acessar em qualquer escopo (global). Isso é um erro. `'use strict'` impede isso.

Erro

```
1 function mostrarCarro() {
2   carro = 'Fusca';
3   console.log(carro);
4 }
5
6 mostrarCarro(); // Fusca
7 console.log(carro); // Fusca
```

ESCOPO DE FUNÇÃO (PAI)

Variáveis declaradas no escopo pai da função, conseguem ser acessadas pelas funções.

Pai

```
1 var carro = 'Fusca';
2
3 function mostrarCarro() {
4   var frase = `Meu carro é um ${carro}`;
5   console.log(frase);
6 }
```

```
7
8  mostrarCarro(); // Meu carro é um Fusca
9  console.log(carro); // Fusca
```

ESCOPO DE BLOCO

Variáveis criadas com `var`, vazam o bloco. Por isso com a introdução do ES6 a melhor forma de declarmos uma variável é utilizando `const` e `let`, pois estas respeitam o escopo de bloco.

escopo

```
1  if(true) {
2    var carro = 'Fusca';
3    console.log(carro);
4  }
5  console.log(carro);
```

VAR VAZA O BLOCO

Mesmo com a condição falsa, a variável ainda será declarada utilizando hoisting e o valor ficará como undefined.

var

```
1  if(false) {
2    var carro = 'Fusca';
3    console.log(carro);
4  }
5  console.log(carro); // undefined
```

CONST E LET NO LUGAR DE VAR

A partir de agora vamos utilizar apenas `const` e `let` para declarmos variáveis.

let

```
1  if(true) {
2    const carro = 'Fusca';
3    console.log(carro);
4  }
5  console.log(carro); // erro, carro is not defined
```

{ } CRIA UM BLOCO

Chaves `{}` criam um escopo de bloco, não confundir com a criação de objetos `= {}`

```
{
```

```
1 {
2   var carro = 'Fusca';
3   const ano = 2018;
4 }
5 console.log(carro); // Carro
6 console.log(ano); // erro ano is not defined
```

Exercício

- Por qual motivo o código abaixo retorna com erros?

ex1

```
1  {
2    var cor = 'preto';
3    const marca = 'Fiat';
4    let portas = 4;
5  }
6  console.log(var, marca, portas);
```

- Como corrigir o erro abaixo?

ex2

```
1  function somarDois(x) {
2    const dois = 2;
3    return x + dois;
4  }
5  function dividirDois(x) {
6    return x + dois;
7  }
8  somarDois(4);
9  dividirDois(6);
```

- O que fazer para total retornar 500?

ex3

```
1  var numero = 50;
2
3  for(var numero = 0; numero < 10; numero++) {
4    console.log(numero);
5  }
6
7  const total = 10 * numero;
8  console.log(total);
```