

# Aula 06

## FRONT END: JAVASCRIPT



Professor  
Lucas Fernandes

# Trabalhando com funções

- Funções
  - declaração
  - parâmetros
  - retorno
- Escopo de variáveis
  - var x let
- Objetos e métodos

# Funções

- Código reutilizável
- Abstrai a implementação
- Entradas, saída
- Efeitos colaterais



## Funções



```
>> sum(5, 10)  
← 15
```



```
>> fullName('John', 'Doe')  
← "John Doe"
```

# Funções

- Efeitos colaterais
  - altera o ambiente

**'Hello, world!'**



console.log

```
>> console.log('Hello, world!')
```

```
Hello, world!
```

```
← undefined
```

\* Exibe texto no console \*

## Declaração de funções

```
function fullName(firstName, lastName) {  
  let firstAndLastName = `${firstName} ${lastName}`;  
  return firstAndLastName;  
}
```

## Declaração de funções

```
function sum(a, b) {  
    return a + b;  
}
```

```
let sum = function (a, b) {  
    return a + b;  
}
```

## Atribuição de funções

- Variáveis também podem apontar para funções





# Escopo de variáveis

- A qual variável um nome se refere?
- Cada escopo é uma nova área com nomes disponíveis
- Se um nome não existe em um escopo ele vai ser procurado no escopo acima

## Escopo de variáveis

```
let a = 3;
let b = 4;
let c = 5;

function myFunc(a) {
  let b = 14;

  console.log(
    `a: ${a}, b: ${b}, c: ${c}`
  );
}
```

```
console.log('Externo');
console.log(
  `a: ${a}, b: ${b}, c: ${c}`
);
// Externo
// a: 3, b: 4, c: 5

console.log('Interno');
myFunc(13);
// Interno
// a: 13, b: 14, c: 5
```

# Escopo de variáveis

## **escopo global**

a = 3  
b = 4  
c = 5

*Externo*  
a: 3, b: 4, c: 5

## **escopo myFunc**

a = 13 (argumento)  
b = 14

*Interno*  
a: 13, b: 14, c: 5

## var e let

- var
  - escopo de função
- let
  - escopo de bloco\*

*\* blocos são as estruturas onde abrimos e fechamos chaves  
ex: for, while, if*

## var e let

```
for(var a = 0; a < 1; a++) {  
    console.log('interno');  
    console.log(a);  
}  
console.log('externo');  
console.log(a);  
  
// interno  
// 0  
// externo  
// 1
```

```
for(let b = 0; b < 1; b++) {  
    console.log('interno');  
    console.log(b);  
}  
console.log('externo');  
console.log(b);  
  
// interno  
// 0  
// externo  
// erro -- b não existe
```

# Objetos e métodos

- Propriedades podem apontar para funções

```
function sayHello() {  
    console.log('Hello!!')  
}  
  
let myObj = {  
    hi: sayHello  
}
```

```
myObj.hi()  
// Hello!
```

# MATONDO!

(OBRIGADO)



Diversidade para Inovação.

Acesse:  
<https://toti.site/>