

Aonde você quer chegar?  
**Vai com a**



**UniCesumar**

EDUCAÇÃO PRESENCIAL E A DISTÂNCIA





# Programação Front-end

Professor Felipe Lemos Haertel

- **Aula 1: Princípios de desenvolvimento e Introdução ao HTML 5**
  - Unidade I e Unidade II
- **Complementação do HTML5 com aplicação de CSS**
  - Unidade II e III
- **Introdução ao JavaScript**
  - **Unidade IV**
- **HTML, CSS e JavaScript com jQuery**
  - Unidade IV

 <https://haertel.com.br>



# JS

- JavaScript
  - Também chamada de JS, é a linguagem de criação de scripts para a Web;
- É utilizado por bilhões de páginas para:
  - Adicionar funcionalidades;
  - Verificar formulários;
  - Comunicar com servidores;
  - E muitos mais.





## ■ Introdução

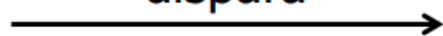
- Originalmente desenvolvido por Brendan Eich da Netscape, sob o nome de Mocha, posteriormente teve seu nome mudado para LiveScript, e por fim JavaScript.
- Linguagem interpretada no navegador
- Modelo de execução controlado por eventos
- Tipagem dinâmica: tipos são associados com valores e não com variáveis

## ■ Funcionalidades

- Manipular conteúdo e apresentação
- Inserir marcação HTML
- Controlar de forma dinâmica a folha de estilo
- Manipular o navegador
- Criar janelas (alertas, confirmações)
- Interagir com formulários
- Acessar campos e valores digitados em formulários
- Validar dados



dispara



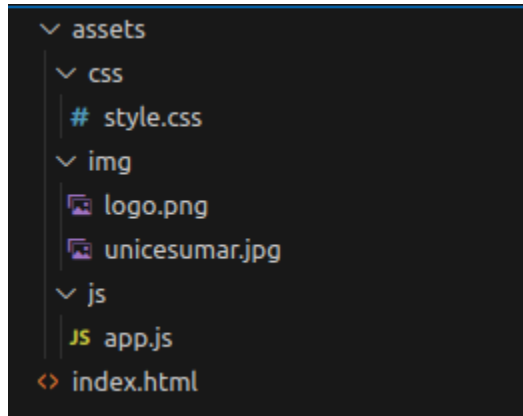
- **Ações em elementos HTML**

- Carregamento da página
- Clique em um botão
- Campo de formulário modificado

- **Bloco de código executado quando é "chamado"**

- Abre uma janela de alerta
- Modifica o árvore do documento HTML
- Modifica o estilo de um elemento HTML





- Estrutura

`<script language="javascript">`

...

`</script>`

- São os delimitadores, respectivamente, de início e fim da codificação JavaScript. Todos os comandos e instruções JavaScript devem estar internamente entre esses delimitadores.

`<script src="app.js" ></script>`





## ■ Variáveis e constantes

- `var js_nome;`
  - É a definição de uma variável identificada por `js_nome`, para reservar espaço na memória, onde uma informação será armazenada.
- `let js_nome;`
  - variáveis com escopo de bloco.
- `const js_nome`
  - declaração de constantes

## ■ Var

```
var mensagemForaDoIf = 'Aula';  
if(true) {  
    var mensagemDentroDoIf = 'JavaScript';  
    console.log(mensagemDentroDoIf) // JavaScprit;  
}  
console.log(mensagemForaDoIf); // Aula  
  
console.log(mensagemDentroDoIf); // JavaScript
```

```
JavaScript  
Aula  
JavaScript  
> |
```

- `var`

```
var mensagemForaDoIf = 'Aula';  
if(true) {  
    var mensagemDentroDoIf = 'JavaScript';  
    console.log(mensagemDentroDoIf) // JavaScript;  
}  
console.log(mensagemForaDoIf); // Aula  
  
console.log(mensagemDentroDoIf); // JavaScript
```





```
JavaScript
Aula
✖ Uncaught ReferenceError: escopoBloco is not
  defined
    at index.html:46:21
```

- let

```
if(true) {
    var escopoFuncao = 'Aula';
    let escopoBloco = 'JavaScript';
    console.log(escopoBloco);
    // JavaScript
}
console.log(escopoFuncao); // Aula
console.log(escopoBloco);
// Apresenta erro
```



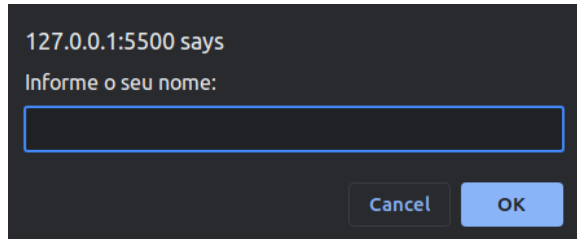
```
Aula
❌ ▶ Uncaught TypeError: Assignment to constant variable.
    at index.html:41:10
> |
```

- const

```
const mensagem = 'Aula';
console.log(mensagem); // Aula
mensagem = 'JavaScript';
```

*// imprime Aula e da erro*





- `window.prompt`

```
let js_name = window.prompt('Informe o seu nome:');
```





Olá Felipe. Seja Bem Vindo

- document.write

```
document.write('Olá ' + js_name + '. Seja Bem Vindo');
```



- ; (ponto-e-vírgula)
  - É utilizado para indicar o final da linha de comando;
  - No JavaScript ele é OPCIONAL.

## ■ IF e ELSE

```
if (12 < 5)
{
    //Instruções a serem executadas se o desvio for verdadeiro
}
else if ("palavra" == "texto")
{
    //Instruções a serem executadas se o desvio anterior for
    //falso e este desvio for verdadeiro
}
else
{
    //Instruções a serem executadas se o desvio anterior for
    //falso
}
```



- While

```
parar = 'N'  
while (parar != 'S'){  
    parar = window.prompt('deseja parar o laço? (S/N)');  
}
```

- Do while

```
do{
```

```
    parar = window.prompt('deseja parar o laço? (S/N)');
```

```
} while (parar != 'S')
```

## ■ for

```
for (i = 0; i < 8; i++) {  
    //Codigo a ser executado enquanto a condição for satisfeita.  
}
```

## ■ Tipos de Dados

```
<script>  
  let a = "20"  
  let b = a * 2   // b = 40  
  let c = a / 2   // c = 10  
  let d = a - 2   // d = 18  
  let e = a + 2   // e = '202'  
  
  console.log("a=" + a + "/b=" + b + "/c=" + c + "/d=" + d + "/e=" + e)  
</script>
```



- Conversão
  - Number()
  - parseInt()
  - parseFloat()

## ■ typeof

```
<script>
```

```
let fruta = "bergamota"
```

```
let valor = 25.40
```

```
let verdade = true
```

```
let novoValor
```

```
console.log(typeof fruta) //string
```

```
console.log(typeof preco) // number
```

```
console.log(typeof levar) // boolean
```

```
console.log(typeof novoValor //undefined
```

```
</script>
```

- `querySelector()` e `querySelectorAll()`
  - `document.querySelector("form")`
  - `document.querySelector("#nome")`
  - `document.querySelector(".lista")`
  - `document.querySelectorAll(selectors);`

## ■ addEventListener

### ■ Mouse Events:

- click, dblclick, mousedown, mouseup, contextmenu, mouseout, mousewheel, mouseover

### ■ Touch Events:

- touchstart, touchend, touchmove, touchcancel

### ■ Keyboard Events:

- keydown, keyup, keypress

### ■ Form Events:

- focus, blur, change, submit

### ■ Window Events:

- resize, scroll, load, unload, hashchange



# Qual é o seu nome?

Nome:

```
<h1>Qual é o seu nome?</h1>
```

```
<form>
```

```
  <label for="nome">Nome:</label>
```

```
  <input type="text" id="nome">
```

```
  <input type="submit" value="Salvar">
```

```
</form>
```

```
<h2></h2>
```



# Qual é o seu nome?

Nome:

**Olá Felipe**

```
frm.addEventListener("submit", (e) => {  
  if(!frm.nome.value){  
    resposta.innerText = "Preencha o nome"  
  } else {  
    var nome = frm.nome.value  
    resposta.innerText = `Olá ${nome}`  
  }  
  e.preventDefault()  
})
```



## ■ Array

- `let nomes = []`
- `let nomes = new array()`
- `nomes[0] = "Bruna"`
- `nomes[1] = "Felipe"`
- `nomes[2] = "Renata"`

- **Inclusão e Exclusão**
  - **push()**
    - Adiciona elemento ao final do vetor
  - **unshift()**
    - Adiciona elemento no início do vetor
  - **pop()**
    - Remove elemento do final do vetor
  - **shift()**
    - Remove elemento do início do vetor



- For .. of

```
for (const nome of nomes){  
  console.log(nome)  
}
```

- A cada interação a variável **nome** recebe um elemento do array **nomes**

## ■ forEach

```
nomes.forEach((nome, i) => {  
  console.log(` ${i+1}: ${nome}`)  
})
```

- No forEach podemos obter o conteúdo do array e o índice de cada elemento do array
- O índice i é opcional

## ■ Vetores de Objetos

```
alunos.push({ nome: "Bruna", idade: 33 })  
alunos.push({ nome: "Felipe", idade: 33 })  
alunos.push({ nome: "Renata", idade: 33 })
```

```
for (const aluno of alunos){  
  const {nome, idade} = aluno  
  console.log(`${nome} - ${idade} anos`)  
}
```

## ■ Function

```
function ola() {  
  alert("Olá")  
}  
ola()
```

```
const ola2 = () => {  
  alert("Ola2")  
}  
ola2()
```



## ■ Função com passagem de parâmetro

```
const situacao = (nota, media) => {  
  if (nota >= media) {  
    alert("aprovado")  
  } else {  
    alert("reprovado")  
  }  
}  
  
const prova = Number(prompt("Nota: "))  
  
situacao(prova, 6)
```

- Função com retorno de valor

```
const situacao=(nota,media)=> {  
  const resultado=nota>=media?"Aprovado":"Reprovado"  
  return resultado  
}
```

```
alert(`O aluno está ${situacao(prova,6)}`)
```

## ■ Funções com parâmetros Rest

- Utilizado para receber um conjunto de valores, que são convertidos para um vetor;

```
const somarValores = (...valores) => {  
  let soma = 0;  
  for(const valor of valores){  
    soma += valor  
  }  
}  
  
somarValores(4,3,9)  
somarValores(10,8)
```





# UniCesumar

EDUCAÇÃO PRESENCIAL E A DISTÂNCIA