



DESENVOLVIMENTO WEB

FULL STACK

JAVASCRIPT

- Javascript é o responsável por deixar nossa interface mais dinâmica;
- Desenvolver um bom código JS, requer dedicação e conhecimento;
- Devido a baixa curva de aprendizado e sua baixa tipagem, é muito fácil escrever códigos de baixa performance e confusos;

JAVASCRIPT INLINE

```
<html>
  <body>

    <script language="javascript" type="text/javascript">
      <!--
        document.write("Hello World!")
      //-->
    </script>

  </body>
</html>
```

VARIÁVEIS

1. `var inteiro = 5;`
2. `var flutuante = 10.5;`
3. `var string = "String";`
4. `var bool = false;`

FUNÇÕES

```
1. function hello () {  
2.     console.info("Hello world");  
3. }  
4.  
5. hello();
```

ESCOPO

```
> var global = "Variável Global";  
  
function ola () {  
    var local = "Variável Local";  
    console.info(local);  
}  
  
ola();  
  
console.info(global);  
console.info(local);
```

❏ Variável Local

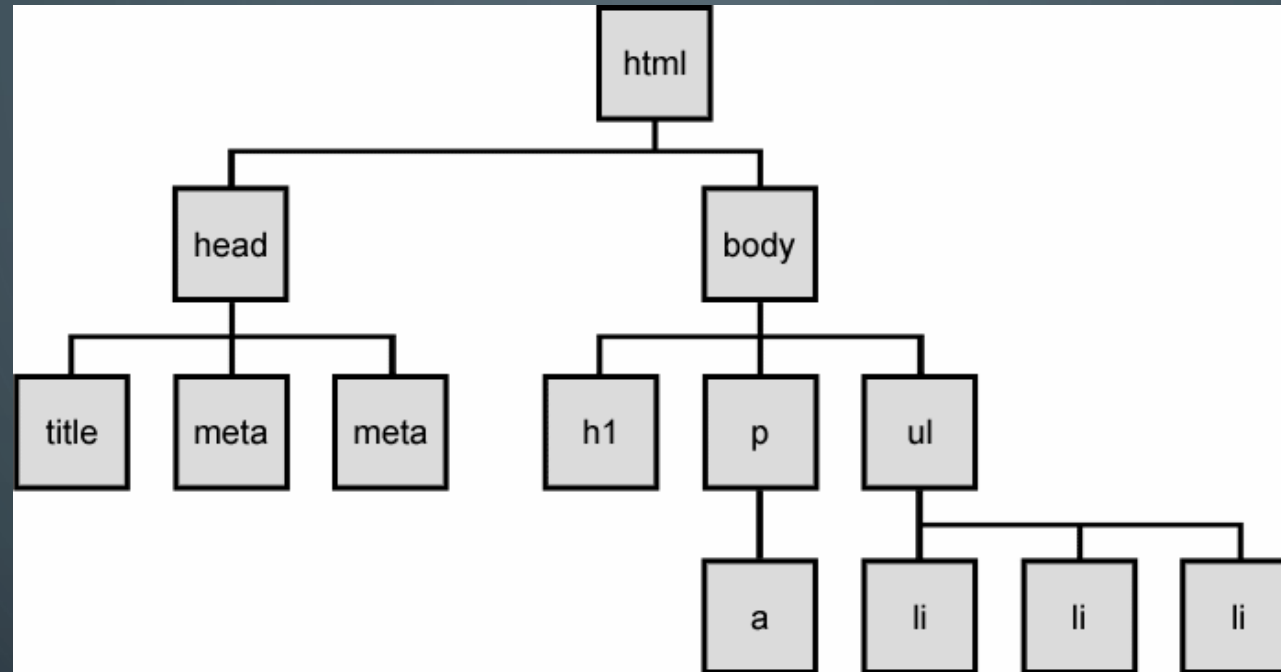
❏ Variável Global

❌ ▶ Uncaught ReferenceError: local is not defined(...)

> |

DOM – DOCUMENT OBJECT MODEL

- O DOM é uma multi-plataforma que representa como as marcações em HTML, XHTML e XML são organizadas e lidas pelo navegador que você usa



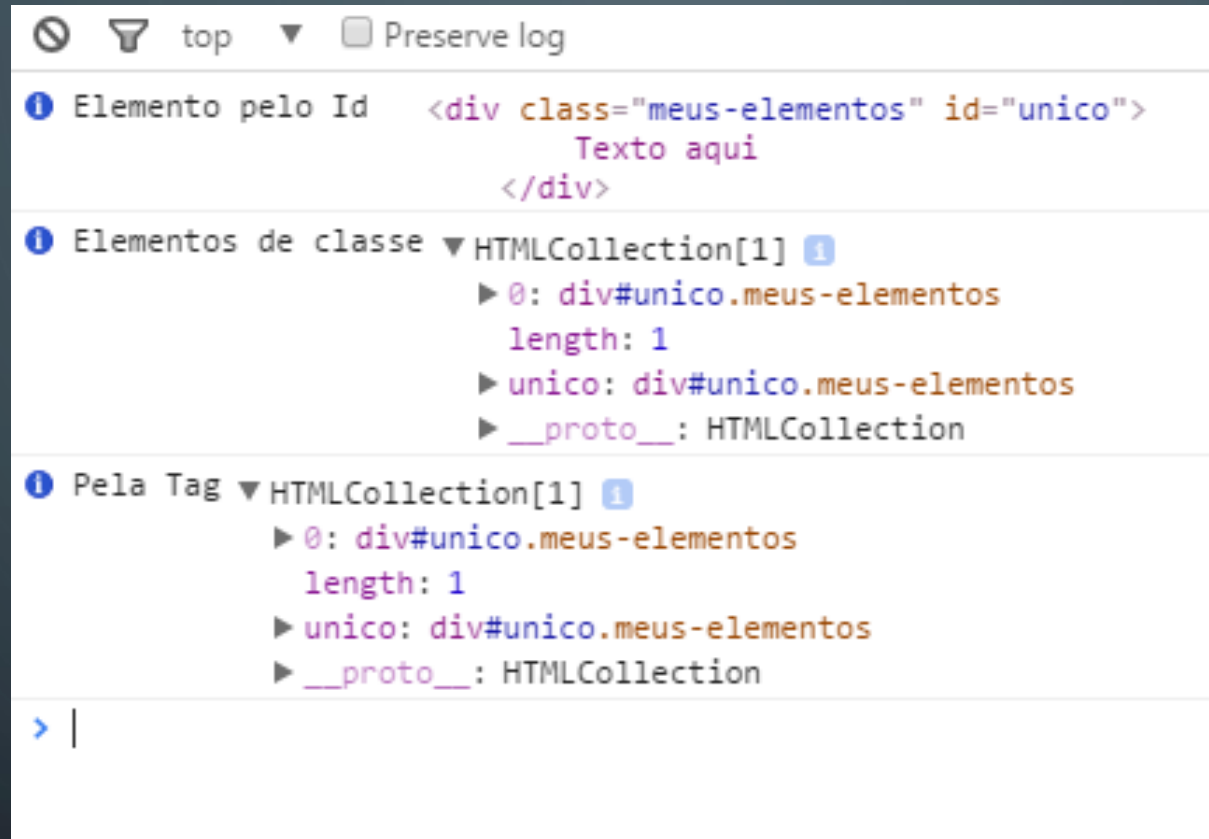
- <http://tableless.com.br/tenha-o-dom/>

ACESSANDO UM ELEMENTO NO DOM

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <meta charset="utf-8" />
    <title>Javascript Ex 1</title>
    <link type="text/css" href="https://cdnjs.cloudflare.com/ajax/libs/t
  </head>
  <body>
    <div class="meus-elementos" id="unico">
      Texto aqui
    </div>

    <script src="scripts/main.js"></script>
  </body>
</html>
```

```
1  var byId = document.getElementById("unico");
2
3  console.info("Elemento pelo Id", byId);
4
5  var byClassName = document.getElementsByClassName("meus-elementos");
6
7  console.info("Elementos de classe", byClassName);
8
9  var byTagName = document.getElementsByTagName("div");
10
11 console.info("Pela Tag", byTagName);
12
```



EVENTOS

- Para interagir com os elementos do DOM utilizamos eventos, e são através deles que capturamos as ações feitas pelo usuário;
- Para isso precisamos adicionar um “ouvinte” que irá ser o responsável por “escutar” uma determinada ação e executar os comandos necessários.

EXEMPLO

```
var btn_alert = document.getElementById("button-alert");

btn_alert.addEventListener("click", function () {
    /** Mensagem no console */
    console.info("Clicou");
    /** Alerta no navegador */
    alert("Alert Clicou");
});

var btn_end = document.getElementById("button-end");

btn_end.addEventListener("click", function () {
    // Cria um Elemento P
    var p = document.createElement("p");
    // Cria um elemento de texto
    var textNode = document.createTextNode("Ok");
    p.appendChild(textNode);
    // Pega o Body
    var body = document.getElementsByTagName("body")[0];
    // Adiciona o elemento P ao body
    body.appendChild(p);
});
```

EXERCÍCIO 1

- Criar um formulário de cadastro de pessoas, onde um usuário poderá ter vários endereços, mas ele irá adicionando conforme a necessidade.
- Utilizar somente Javascript Puro e Bootstrap (CSS)
- Plus 1: Inserir um contador para os campos: Endereço 1, Endereço 2, Endereço 3

TIPOS DE EVENTO

- Alguns tipos mais comuns de eventos dos objetos do DOM
 - Click
 - Mouse Over
 - Mouse Out
 - Load (ao carregar a página)
 - Key Down, Key Up, Key Press

EXERCÍCIO 2

- Fazer uma animação coma logo da URI que está no repositório do GIT. As animações deverão partir do teclado, para cada seta pressionada a logo deverá se dirigir para o canto correspondente.
- * Utilizar as transições para fazer a animação
- Somente Javascript puro e Bootstrap