

JavaScript

Quem eu sou?

Certificado

O SENAI - Dr. Celso Charuri
certifica que

Gabriel Pucci Bizio

natural de Franca/SP, nacionalidade Brasileira, RG [REDACTED] CPF [REDACTED] nascido(a) em
27/03/2003, concluiu no período de 04/11/2021 a 18/11/2021 o curso de

Programação WEB: HTML, CSS e JavaScript

JavaScript

|

O que é?

- Usado majoritariamente na programação web.
- Permite e facilita a implementação de várias funcionalidades importantes em websites.
- Roda nas engines dos navegadores e no node.js.

JavaScript

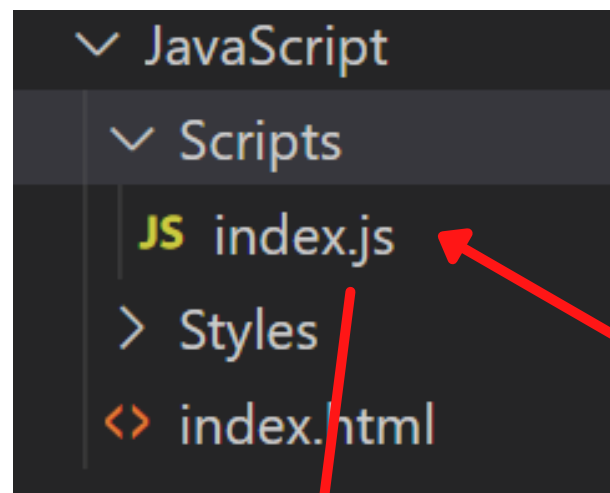


O que é?

- É declarado no arquivo HTML (Boa prática : o código em si fica em um arquivo separado, que é mencionado no final da tag '<body>').
- Veremos sua sintaxe, declaração de variáveis e operadores.
- Possui struct e ferramentas orientadas para objeto.

JavaScript

Declaração



```
JavaScript > Scripts > JS index.js > ...
1  console.log("Hello world");
2
```

```
JavaScript > <> index.html > html
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta http-equiv="X-UA-Compatible" content="IE=edge">
6      <meta name="viewport" content="width=device-width, initial-scale=1.0">
7      <title>JavaScript</title>
8  </head>
9  <body>
10     <button onclick="alertaSoma(2, 4)"> Clique aqui... </button>
11     <p id="demo">... E esse parágrafo ira desaparecer.</p>
12
13
14     <script src="Scripts/index.js"></script>
15
16 </body>
17 </html>
```

JavaScript

Variáveis

Variáveis no JavaScript são dinâmicas
Var vs Let vs Const.

```
// How to create variables:
```

```
var x;  
let y;
```

```
// How to use variables:
```

```
x = 5;  
y = 6;  
let z = x + y;
```

```
const z = 10;  
const z = 7; //Vai dar erro.  
z = 10; //vai dar erro.
```

```
let x = 5;  
x = 1; //Não daria erro.  
let x = 2; //Vai dar erro.
```

```
var y = 5;  
y = 1;  
var y = 2; //Não vai dar erro.
```

JavaScript



Variáveis

A maior diferença entre os tipos de declaração é
no escopo da variável:

```
function varB(){  
  if (true){  
    var b = 10;  
  }  
  console.log(b); //Certo!  
}
```

```
function letB(){  
  if (true){  
    let b = 10;  
  }  
  console.log(b); //Errado  
}
```

JavaScript



Variáveis

- Number: Armazena numeros (inteiros e decimais).
- String e boolean.
- Undefined: valor de variáveis declaradas mas sem valores atribuídos.
- NULL: Valor vazio ou inexistente. Precisa ser atribuído.
- Objeto: Em JS, arrays são objetos. Outros tipos de objetos lembram o dicionário em python.

JavaScript



Funções

O JavaScript aceita o uso arbitrário de chaves no código. Mas é recomendado que elas sejam sempre usadas.

```
function alertaSoma(x, y){  
  z = x + y;  
  alert(z);  
}
```

- Argumentos são passados por valor
- Objetos são passados por referência

```
<button onclick="alertaSoma(2, 4)"> Clique aqui... </button>
```


JavaScript



Métodos

- 'console.log()' => O print. Mostra uma mensagem no console.
- 'document.getElementById()' => Seleciona um elemento HTML.
- '.innerHTML' => Vem depois do método anterior. Escreve dentro do elemento.
- 'alert()' => Mostra uma caixa de alerta que exibe o parâmetro.

JavaScript



Métodos

- Operadores:
https://www.w3schools.com/js/js_operators.asp
- Operadores aritméticos :
https://www.w3schools.com/js/js_arithmetic.asp

JavaScript

Eventos

Eventos são conceitos importantes. Quando as condições que caracterizem certo evento são concluídas, o script executa um código fornecido (para isso se usa função).

```
<button onclick="getDate('demo')">The time is?</button>

<p id="demo"></p>

<script>
//Seleciona o elemento pelo id (parâmetro) e escreve a data atual
function getDate(date){
    document.getElementById(date).innerHTML=Date()
}
</script>
```

JavaScript



Eventos

- 'onchange' => O elemento HTML é alterado.
- 'onclick' => Usuário clica em um elemento.
- 'onmouseover' => Usuário passa o cursor sobre o elemento.
- 'onmouseout' => Usuário remove o cursor do elemento.
- 'onkeydown' => Usuário pressiona uma tecla.
- 'onload' => Página termina de carregar.

JavaScript



Arrays

Arrays no JavaScript lembram listas no python, mas com algumas diferenças:

- Os valores são acessados através de seu índice.
- Como as variáveis são de tipo dinâmico, arrays aceitam variáveis de diferentes tipos.
- São variáveis especiais do tipo 'objeto'.
- Podem armazenar outros arrays, ou objetos 'normais'.

JavaScript

Arrays

```
const cars = ["Ford", "Volvo", "BMW"];  
  
cars[0] = "Opel";  
  
console.log(cars);
```

▼ Array(3) **i**

- 0: "Opel"
- 1: "Volvo"
- 2: "BMW"
- length: 3

JavaScript > Scripts > JS index.js > ...

```
1  const x = [2003, "Gabriel", [27, 03, 2003]];  
2  console.log(x);  
3  console.log(typeof(x));
```

▼ Array(3) [2003, "Gabriel", (3) [...]]

- 0: 2003
- 1: "Gabriel"
- ▶ 2: Array(3) [27, 3, 2003]
 - length: 3
 - ▶ <prototype>: Array []

object

JavaScript



Objetos

Lembram dicionários em python:

```
const carro = {  
  nome: "Gol", marca: "VW", ano: 2005}  
  
console.log(carro['marca']); //Notação de colchetes.  
  
console.log(carro.marca); //Notação de ponto.
```

VW

VW

JavaScript

|

Extras

Jogo que ensina JavaScript:
<https://00-joe-js.github.io/vast/>

Fontes:

- <https://www.javascriptprogressivo.net/>
- <https://fonts.google.com/specimen/Montserrat>
- <https://www.w3schools.com/js/default.asp>

JavaScript




Exercício

1 - Crie um botão que delete o conteúdo de um parágrafo da página e um botão que o coloca de volta



jQuery

Quem eu sou?

 **SERVIÇO NACIONAL DE APRENDIZAGEM INDUSTRIAL**
DEPARTAMENTO REGIONAL DO PARANÁ 

Certificado

O SENAI - Dr. Celso Charuri
certifica que

MELISSA
M D W

natural de Franca/SP, nacionalidade Brasileira, RG CPF nascido(a) em
27/03/2003, concluiu no período de 04/11/2021 a 18/11/2021 o curso de

Programação WEB: HTML, CSS e JavaScript



- Biblioteca que facilita o uso de JavaScript
- Faz com que o documento HTML seja facilmente manipulado, com animações e eventos.



jQuery

|

O que é?

JavaScript:

```
document.getElementById("demo").innerHTML =  
"Hello, World!";
```

jQuery:

```
$("#demo").html("Hello, World!");
```

jQuery

Como usar?

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta http-equiv="X-UA-Compatible" content="IE=edge">
6      <meta name="viewport" content="width=device-width,
7          initial-scale=1.0">
8      <title>Document</title>
9  </head>
10 <body>
11
12  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.
13  5.1/jquery.min.js"></script> <!-- link para a biblioteca
14  jquery -->
15  <script src="index.js"> </script>
16 </body>
17 </html>
```



`$(selector).ação()`

- \$ é o método de acesso à biblioteca
- O (selector) que se conectará ao elemento do HTML (ID, Class, etc.)
- .ação() é que será aplicado no elemento selecionado

```
$(selector).evento(function(){  
    $(selector).evento();  
});
```

```
$("#demo").click(function(){  
    $(this).hide();  
});
```

`$(this).hide()` - Esconde o elemento atual

`$("p").hide()` - Esconde todos os elementos "p"

`$(".test").hide()` - Esconde todos os elementos
dentro da `class="test"`

`$("#test").hide()` - Esconde todos os elementos
com `id="test"`



São baseados nos selectors do CSS

Selector de elemento:

```
$("#h3").css({color: "blue"});
```

Selector de classe:

```
$(".menu").css({margin: "0 auto"});
```

Selector de ID:

```
$("#clientes").css({border: "3px solid yellow"});
```

Refinar a busca dos elementos no HTML

First e Last

```
$("#header nav li:first).css({border: "2px solid red"})
```

```
$("#header nav li:last).css({border: "2px solid red"})
```

Even e Odd

```
$("#header nav li:even").css({border: "2px solid red"})  
$("#header nav li:odd").css({border: "2px solid red"})
```

Less than e Greater than

```
$("#social-nav li:lt(3)").css({border: "2px solid red"})  
$("#social-nav li:gt(3)").css({border: "2px solid red"})
```

Filtro por atributo - Qualquer div com o atributo

```
$("div[class]").css({color: "blue"});
```

```
$("img[alt]").css({color: "blue"});
```

```
$("img[alt=valor]").css({color: "blue"});
```

- `click()`: executa com um único clique do mouse;
- `hover()`: executa quando o mouse passa sobre um elemento;
- `mouseenter()` and `mouseleave()`: aplicam-se apenas ao mouse que entra ou sai de um elemento, respectivamente;
- `submit()`: executa quando um formulário é enviado;
- `scroll()`: executa uma rolagem na tela;
- `keydown()`: executa quando você pressiona uma tecla no teclado.

Append - Adiciona o conteúdo no final do selector

```
var reginaRouca = "se liga hein";  
$("#p1").append(reginaRouca);
```

Prepend - Adiciona o conteúdo no começo do selector

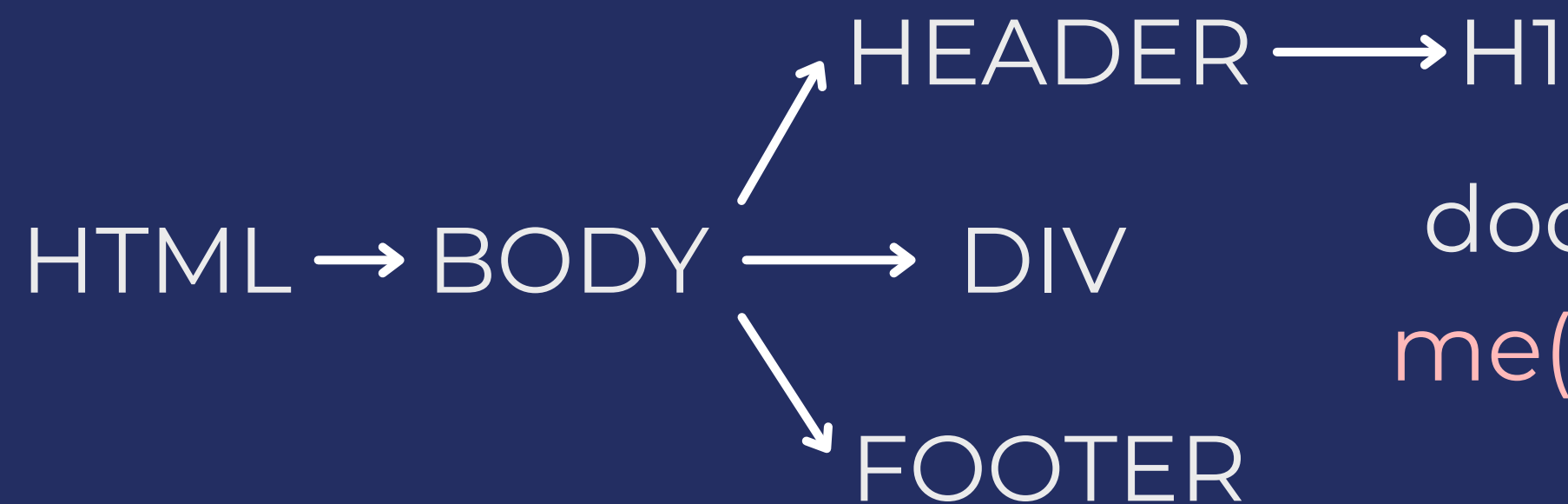
```
var reginaRouca = "se liga hein";  
$("#p1").prepend(reginaRouca);
```

jQuery

|

DOM

Document Object Model, é uma maneira de descrever os relacionamentos entre os elementos de uma página



```
document.getElementsByTagName("header").getElementsByTagName("h1");
```




Permite rodar vários comandos, um depois do outros, no mesmo elemento

```
$("#p1").css("color", "red")  
        .slideUp(2000)  
        .slideDown(2000);
```



jQuery



Exemplo

```
<body>

  <button id="btn">CLICA AQUI!!</button>

  <p> numero de vagas: <span id="demo">5</span></p>
```

```
1  let x = 4;
2
3  $(document).ready(function(){
4      $("#btn").click(function(){
5          $("#demo").html(x);
6          if(x == 0){
7              $("#btn").prop("disabled", true);
8          }
9          x--;
10     });
11 });
```



jQuery Progressivo

Lista de Selectors

Lista de Eventos

