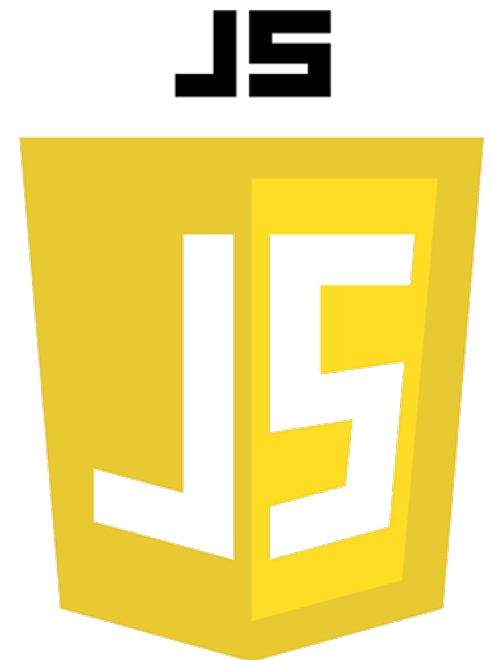




Introdução ao JavaScript

Prof. Dr. Anderson Rodrigues



Tecnologias em uma página Web?

HTML



Base
estrutural da
página WEB

JS



Oferece
interatividade
com os
elementos da
página WEB

CSS



Define a
aparência da
página



O que é JavaScript ?

- É uma linguagem de programação **interpretada** e leve.
- Pode ser considerada uma **linguagem orientada a objeto** embora não suporte totalmente todos os princípios da POO.
- Atualmente o JavaScript é amplamente utilizado para oferecer **interatividade, dinamicidade e flexibilidade** para as páginas web.
- É a única linguagem que pode ser executada **diretamente no navegador**.
- Também é utilizado para desenvolver apps móveis, desktops e jogos.

Tipos de variáveis no JavaScript

TIPO	DESCRIÇÃO	EXEMPLO
Number	número inteiros e de ponto flutuante.	30 12.6 1000 230.30
String	Textos. Obs.: Os textos são delimitados por aspas simples ('), aspas duplas (") ou crases.	'Casa', "643.344.876-44", 'Aposentado'
Boolean	Verdadeiro ou Falso	true ou false
Null	representa ausência de valor	null
Object	estruturas de dados complexas como arrays, datas ou classes.	let data=new Date(1920/12/20)
Undefined	representa valor não definido	Undefined

Declaração de variáveis no JavaScript

Declaração	Exemplo	Escopo	Mutabilidade
var	var n = 10;	De função ou global. Não tem escopo de bloco. Variáveis declaradas com var são elevadas (hoisted), o que significa que podem ser usadas antes de serem declaradas no código.	Podem ter seu valor modificado.
let	let n = 10;	Escopo de bloco. Não podem ser usadas antes da declaração no código.	Podem ter seu valor modificado.
const	const n = 10;	Escopo de bloco. Não podem ser usadas antes da declaração no código.	Não pode ser modificada.

Operadores aritméticos no JavaScript

São utilizados para a realização de cálculos aritméticos:

Operador	Descrição	Exemplo	Resultado
+	Adição	let soma = 5 + 3	soma será 8
-	Subtração	let dif = 5 - 3	a subtração será 2
*	Multiplicação	let prod = 5 * 3;	o produto será 15
/	Divisão	let quoc = 10/2;	a divisão será 5
%	Módulo	let resto = 10 % 3;	o resto será 1
++	Incremento	let a = 5; a++;	a será 6
--	Decremento	let a = 5; a--;	a será 4

Operadores de atribuição no JavaScript

São utilizados para atribuir valores a variáveis:

Operador	Descrição	Exemplo	Resultado
=	Atribuição	let a = 5;	a será 5
+=	Atribuição de adição	let a = 5; a += 3;	a será 8
-=	Atribuição de subtração	let a = 5; a -= 3	a será 2
*=	Atribuição de multiplicação	let a = 5; a *= 3;	a será 15
/=	Atribuição de divisão	let a = 10; a /= 2;	a será 5

Operadores de comparação no JavaScript

São utilizados para realizar comparação entre valores:

Operador	Descrição	Exemplo	Resultado
<code>==</code>	igualdade	<code>5 == "5"</code>	retornará true
<code>===</code>	igualdade em valor e tipo	<code>5 === "5"</code>	retornará false
<code>!=</code>	diferença	<code>5 != 6</code>	retornará true
<code>></code>	maior que	<code>5 > 4</code>	retornará true
<code>>=</code>	maior ou igual	<code>5 >= 4</code>	retornará false
<code><</code>	menor que	<code>4 < 3</code>	retornará false
<code><=</code>	menor ou igual á	<code>4 <= 4</code>	retornará true

São utilizados para relacionar duas ou mais condições:

Operadores lógicos no JavaScript

Operador	Descrição	Exemplo	Resultado
&&	E lógico	$(5 > 3) \ \&\& \ (3 < 1)$	false
 	OU lógico	$(5 > 3) \ \ (3 < 1)$	true
!	NÃO lógico	$! (5 > 3)$	false

São utilizados para controlar o fluxo de execução dos comandos.
Permitindo a tomada de decisões:

Estruturas de controle condicionais no JavaScript

```
let idade = 18;  
if (idade >= 18) {  
  console.log("Você pode dirigir")  
}
```

if (condição) { }

```
let idade = 18;  
if (idade >= 18) {  
  console.log("Você pode dirigir");  
} else {  
  console.log("Você não pode dirigir");  
}
```

if (condição) { } else { }

```
let idade = 18;  
if (idade >= 18) {  
  console.log("Você pode dirigir");  
} else if (idade == 17) {  
  console.log("Você não pode dirigir no próximo ano");  
} else {  
  console.log("Você não pode dirigir");  
}
```

if (condição) { }
else if (condição) { }
else { }

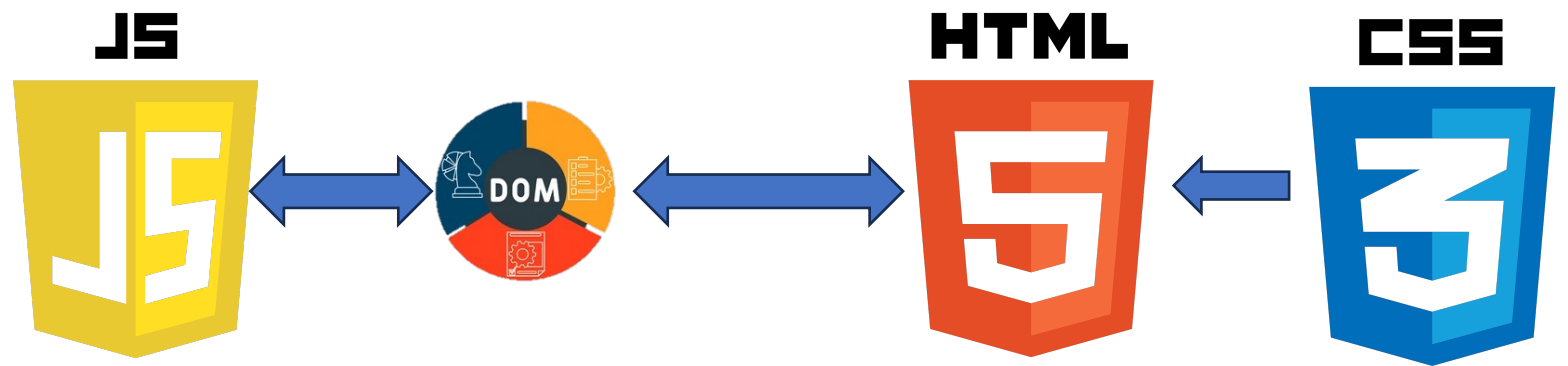
JavaScript: Funções

São utilizadas para executar uma tarefa específica dentro da aplicação JavaScript.

```
function <nome_da_funcao> (argumentos)
{
    <corpo da função>
}
```

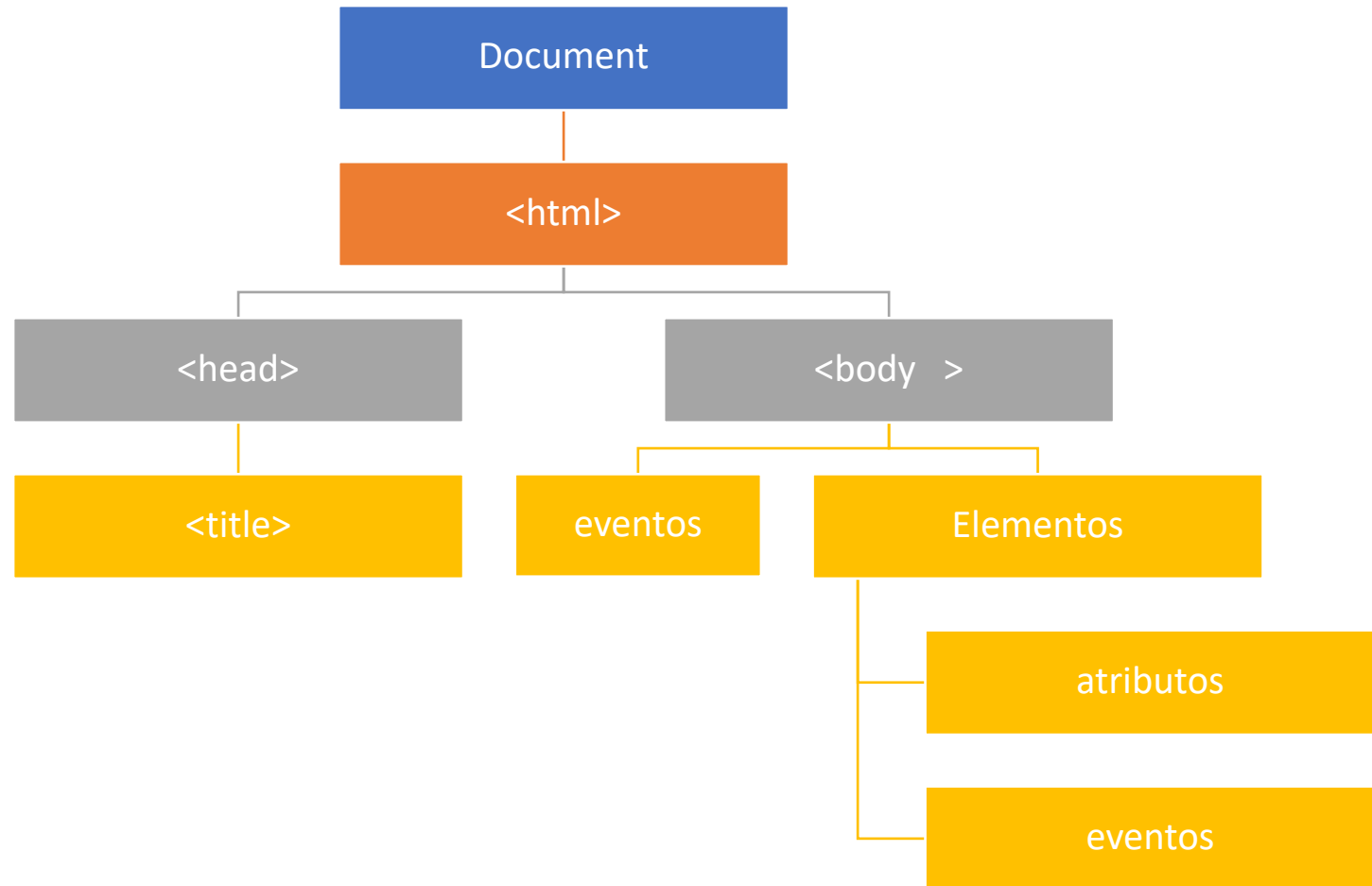
```
//Função para calcular o fatorial de um número
function fatorial(numero) {
    // fatorial de 0 é 1
    if (numero === 0) {
        return 1;
    } else {
        // fatorial(n) = n * fatorial(n-1)
        return numero * calcularFatorial(numero - 1);
    }
}
```

Elementos da construção de páginas web dinâmicas



JavaScript: Document Object Model (DOM)

O Document Object Model (DOM) controla e representa todos os elementos presentes em um documento HTML ou XML. Os principais elementos controlados pelo DOM incluem:



JavaScript:

Principais funções de acesso ao DOM.

- **document.getElementById(id):**
Descrição: Retorna uma referência para o elemento que possui o atributo id especificado.
- **document.getElementsByClassName(className):**
Descrição: Retorna uma lista de elementos que possuem a classe CSS especificada.
- **document.getElementsByTagName(tagName)**
Descrição: Retorna uma lista de elementos com o nome da tag HTML especificada.
- **document.querySelector(selector):**
Descrição: Retorna o primeiro elemento que corresponde ao seletor CSS especificado.

JavaScript:
Principais
funções de
acesso ao
DOM.

ATIVIDADES PRÁTICAS

https://drive.google.com/drive/folders/1TKG_7kWpPFoUzhwSsniYzIYR_1I2AFwi?usp=drive_link



JavaScript: Principais funções de acesso ao DOM.

- **document.querySelectorAll(selector):**
Descrição: Retorna todos os elementos que correspondem ao seletor CSS especificado como uma NodeList.
- **parentNode.childNodes:**
Descrição: Retorna uma NodeList de nós filhos de um nó pai.
- **parentNode.firstChild e parentNode.lastChild:**
Descrição: Retorna o primeiro e o último filho de um nó pai, respectivamente.
- **element.parentNode:**
Descrição: Retorna o nó pai do elemento.

JavaScript:

Principais funções de acesso ao DOM.

- **element.nextSibling e element.previousSibling:**
Descrição: Retorna o nó irmão seguinte e anterior, respectivamente.
- **element.classList:**
Descrição: Retorna uma coleção de classes do elemento como um objeto DOMTokenList.



JavaScript:
Principais
funções de
acesso ao
DOM.

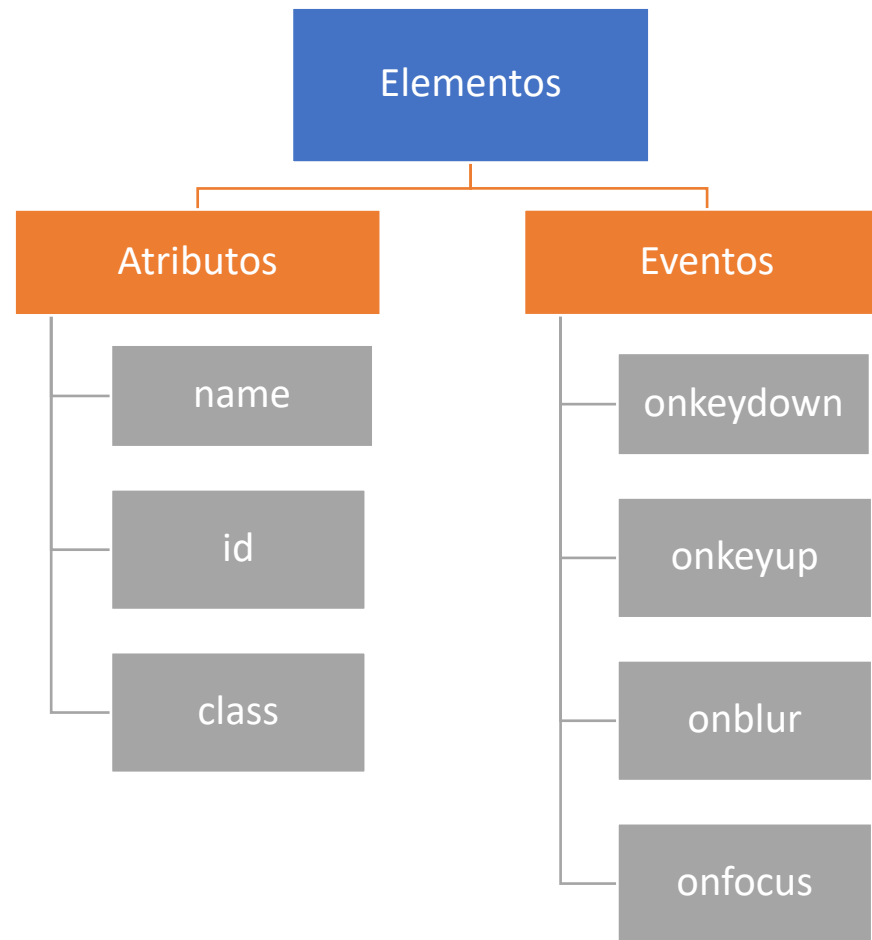
ATIVIDADES PRÁTICAS

https://drive.google.com/drive/folders/1TKG_7kWpPFoUzhwSsniYzIYR_1I2AFwi?usp=drive_link



JavaScript: Document Object Model (DOM)

Todos os elementos que fazem parte do DOM estão estruturados em **atributos** e **eventos**.



JavaScript:

Principais atributos do DOM

id:

Função: Identifica exclusivamente um elemento na página. Usado para referenciar o elemento no JavaScript ou em folhas de estilo CSS.

class:

Função: Define uma ou mais classes para um elemento. Utilizado para aplicar estilos CSS ou identificar grupos de elementos no JavaScript.

style:

Função: Permite a definição de estilos diretamente no elemento. Pode conter propriedades CSS como cor, tamanho, margens, etc.

src:

Função: Especifica o caminho do arquivo para elementos como , <script>, <audio>, etc.

JavaScript:

Principais atributos do DOM

alt:

Função: Fornece um texto alternativo para elementos como ``. Exibido se a imagem não puder ser carregada.

width e height:

Função: Define as dimensões de elementos como ``, `<canvas>`, etc.

type:

Função: Especifica o tipo de um elemento, comum em elementos `<input>`, `<button>`, `<script>`, etc.

value:

Função: Define o valor de um elemento, frequentemente usado em elementos de formulário como `<input>` e `<button>`.

name:

Função: Especifica um nome para elementos como `<input>`, usado para identificação ao enviar dados através de formulários.

JavaScript:

Principais atributos do DOM

href:

Função: Indica o URL de destino para elementos <a> (âncora) e outros que referenciam recursos externos.

placeholder:

Função: Exibe um texto temporário em campos de entrada, fornecendo uma dica sobre o que deve ser inserido.

checked:

Função: Em elementos <input> do tipo checkbox ou radio, indica se o elemento está marcado.

disabled:

Função: Desabilita a interação com um elemento, como em <input>, <button>, etc.

JavaScript:

Principais atributos do DOM

readonly:

Função: Torna um elemento apenas leitura, aplicável a campos de entrada como `<input>` e `<textarea>`.

maxlength:

Função: Define o número máximo de caracteres permitidos em campos de entrada, como `<input>` e `<textarea>`.

colspan e rowspan:

Função: Controla a união de células em tabelas HTML, especificando o número de colunas ou linhas que uma célula deve ocupar.

action:

Função: Especifica o URL para onde os dados de um formulário devem ser enviados quando submetidos.

method:

Função: Indica o método HTTP a ser usado ao enviar dados de formulário. Geralmente "GET" ou "POST".

JavaScript: Principais atributos do DOM

target:

Função: Especifica onde abrir o link quando um elemento <a> é clicado. Pode ser _blank, _self, _parent, _top, etc.

for:

Função: Associado a elementos <label>, especifica qual campo de entrada o rótulo está associado, referenciando o id desse campo.

JavaScript: Principais elementos e seus atributos

Elemento <html>:

Atributos:
lang, class, id.

Elemento <head>:

Atributos:
title, meta, link, style.

Elemento <button> (botão):

Atributos:
type, name, value, id, class, style.

Elemento <form> (formulário):

Atributos:
action, method, id, class, style.

Elemento :

Atributos:
id, class, style.



JavaScript: Principais elementos e seus atributos

Elemento <h1>, <h2>, <h3>, ... <h6> (cabeçalhos):

Atributos:

id, class, style.

Elemento (lista não ordenada):

Atributos:

id, class, style.

Elemento (item de lista):

Atributos:

id, class, style.

Elemento <div>:

Atributos:

id, class, style.

Elemento <p> (parágrafo):

Atributos:

id, class, style.



JavaScript: Principais elementos e seus atributos

Elemento <a> (âncora):

Atributos:

href, target, id, class, style.

Elemento (imagem):

Atributos:

src, alt, width, height, id, class, style.

Elemento <input> (campo de entrada):

Atributos:

type, name, value, placeholder, id, class, style.

Elemento <select> (lista de seleção):

Atributos:

name, id, class, style.

JavaScript: Principais elementos e seus eventos

Elemento <html>:

1. onload: Dispara quando o documento é totalmente carregado.
2. onunload: Dispara quando o usuário sai da página.

Elemento <body>:

1. onload: Dispara quando o corpo do documento é totalmente carregado.
2. onunload: Dispara quando o usuário sai da página.
3. onresize: Dispara quando o tamanho da janela é alterado.
4. onscroll: Dispara quando a barra de rolagem é movida.

Elemento <div>:

1. onclick: Dispara quando o elemento é clicado.
2. onmouseover: Dispara quando o ponteiro do mouse está sobre o elemento.
3. onmouseout: Dispara quando o ponteiro do mouse deixa o elemento.

JavaScript: Principais elementos e seus eventos

Elemento **<button>**:

1. onclick: Dispara quando o botão é clicado.

Elemento **<input>**:

1. onchange: Dispara quando o valor do campo de entrada é alterado.
2. onfocus: Dispara quando o campo de entrada obtém foco.
3. onblur: Dispara quando o campo de entrada perde o foco.

Elemento **<a> (âncora)**:

1. onclick: Dispara quando o link é clicado.
2. onmouseover: Dispara quando o ponteiro do mouse está sobre o link.
3. onmouseout: Dispara quando o ponteiro do mouse deixa o link.

JavaScript: Principais elementos e seus eventos

Elemento <form> (formulário):

1. onsubmit: Dispara quando o formulário é enviado.
2. onreset: Dispara quando o formulário é resetado.

Elemento <select> (lista de seleção):

1. onchange: Dispara quando o valor da lista de seleção é alterado.

Elemento (imagem):

1. onload: Dispara quando a imagem é carregada.
2. onerror: Dispara se houver um erro ao carregar a imagem.