

JavaScript

- × O que é e Porque estudar;**
- × Introdução;**
- × Onde colocar o código?;**
- × Saídas;**
- × Statements;**
- × Sintaxe e comentários;**
- × Variáveis;**
- × Let e Const;**
- × Operadores;**

JavaScript

× Tipos de dados;

× Funções;

× Objetos;

× Eventos;

× Strings;

× Arrays;

× Condicional;

JavaScript

- × **Switch;**
- × **Loop For;**
- × **Loop for in;**
- × **Loop for of;**
- × **Loop while;**
- × **Dom.**



JavaScript

***O que é e Porque estudar:**

- É a linguagem de programação mais popular do mundo;
- É a linguagem de programação da WEB;
- Fácil de aprender;
- Como JavaScript é possível programar o comportamento das páginas.

JavaScript

× Introdução:

- Com JavaScript é possível mudar o conteúdo de um elemento HTML;
- É possível mudar o valor de um atributo HTML;
- É possível alterar um estilo CSS;
- Esconder ou mostrar um elemento HTML.



JavaScript

× Onde colocar o código JavaScript?

- Pode ser inserido entre a tag `<script>` e `</script>` dentro do `<body>` e/ou da `<head>`
- Dica: colocar os scripts no final da tag `<body>` melhora o desempenho da página;
- Pode ser colocar dentro de um arquivo externo (myScript.js, por exemplo). Nesse caso é necessário fazer referencia ao arquivo usando a tag `<script src="arquivo.js"></script>`

JavaScript

× Onde colocar o código JavaScript?

- Vantagens de usar arquivo externo:
 - O código HTML, fica separado;
 - Mais fácil a leitura e manutenção;
 - Arquivos .js podem ficar em cache aumentando a velocidade de carregamento da página

JavaScript

× Saídas:

- JavaScript permite algumas possibilidades para mostrar os dados;
 - Escrever em um elemento HTML usando `innerHTML`;
 - Escrever em uma saída HTML usando `document.write()`;
 - Usar um alert box, usando `window.alert()`;
 - Escrever no console do browser, usando `console.log()`.

JavaScript

×Declarações:

- Um programa de computador é uma lista de instruções para serem executadas pelo computador;
- Essas instruções são chamadas de declarações;
- Um programa JavaScript é uma lista de declarações;
- As declarações do JavaScript são compostas de: valores, operadores, expressões, palavras reservadas e comentários;
- As declarações são executadas uma por uma, na ordem que foram escritas.

JavaScript

× Declarações:

- O ponto e vírgula separa as declarações em JavaScript;
- Algumas palavras reservadas não podem ser usadas como nomes de variáveis. (https://www.w3schools.com/js/js_reserved.asp);



JavaScript

× Sintaxe:

- A sintaxe do JavaScript é um conjunto de regras, que definem como os programas são construídos;
- Os valores podem ser fixos (literais) ou variáveis;
- Literais, são números (com ou sem o ponto decimal) e strings escritas entre aspas simples (' ') ou duplas (" ");
- Para declarar uma variável use a palavra reservada var e um sinal de igual para atribuir valor a variável

JavaScript

× Comentários:

- Comentários de uma única linha podem ser feitos usando `//`;
- Comentários de mais de uma linha, começam com `/*` e terminam com `*/`;

JavaScript

×Variáveis:

- Armazenam o valor dos dados;
- Em 2015 a versão ES6 permite usar a palavra reservada `const` para variáveis que não mudam de valor e `let` para definir uma variável em um escopo;
- Algumas versões de browsers não suportam ES6;

JavaScript

× let:

- Let e const fornecem variáveis (e constantes) de escopo de bloco;
- Antes de ES2015 existiam apenas dois tipos de escopo: global e escopo de função;
- Variáveis declaradas como var não possuem escopo de bloco;
- Variáveis declaradas como let possuem escopo de bloco, ou seja só existem dentro do bloco que foram criadas.

JavaScript

***const:**

- Const tem o mesmo principio da variável let, porem o valor é constante, ou seja, não pode ser alterado;
- Outro ponto importante é que variáveis const devem receber um valor no momento que são declaradas.

JavaScript

× Operadores:

- O operador igual (=) é usado para atribuir valores a uma variável;

Operator	Description
+	Addition
-	Subtraction
*	Multiplication
**	Exponentiation (ES2016)
/	Division
%	Modulus (Remainder)
++	Increment
--	Decrement

JavaScript

*Operadores:

Operator	Example	Same As
=	<code>x = y</code>	<code>x = y</code>
+=	<code>x += y</code>	<code>x = x + y</code>
-=	<code>x -= y</code>	<code>x = x - y</code>
*=	<code>x *= y</code>	<code>x = x * y</code>
/=	<code>x /= y</code>	<code>x = x / y</code>
%=	<code>x %= y</code>	<code>x = x % y</code>
**=	<code>x **= y</code>	<code>x = x ** y</code>

JavaScript

*Operadores:

Operator	Description
&&	logical and
	logical or
!	logical not

Operator	Description
==	equal to
===	equal value and equal type
!=	not equal
!==	not equal value or not equal type
>	greater than
<	less than
>=	greater than or equal to
<=	less than or equal to
?	ternary operator

JavaScript

× Operadores:

Operator	Description
<code>typeof</code>	Returns the type of a variable
<code>instanceof</code>	Returns true if an object is an instance of an object type

JavaScript

*Operadores:

Operator	Description	Example	Same as	Result	Decimal
&	AND	5 & 1	0101 & 0001	0001	1
	OR	5 1	0101 0001	0101	5
~	NOT	~ 5	~0101	1010	10
^	XOR	5 ^ 1	0101 ^ 0001	0100	4
<<	Zero fill left shift	5 << 1	0101 << 1	1010	10
>>	Signed right shift	5 >> 1	0101 >> 1	0010	2
>>>	Zero fill right shift	5 >>> 1	0101 >>> 1	0010	2

JavaScript

×Tipos de dados:

- Podem ser: numbers, string, objects, boolean, Arrays, e outros;
- Quando um numero é adicionado a uma string o numero será convertido para uma string.

JavaScript

× Funções:

- São um bloco de código específico para determinada tarefa;
- Sintaxe: palavra reservada `function` + nome da função seguido de `()`;
- Os parênteses podem incluir parâmetros, separados por vírgula;
- Argumentos são valores que recebidos quando a função é chamada.

JavaScript

×Objetos:

- Um objeto propriedades e métodos;
- Objetos são variáveis. Suas propriedades são colocados entre { }
- O valor é separado do nome por dois pontos, e as propriedades são separadas por vírgula
- Para acessar uma propriedade podemos escrever: `objectName.propertyName` ou `objectName["properttName"]`.

JavaScript

×Eventos:

- São “coisas” que acontecem a elementos HTML;
- JavaScript pode “reagir” a esses eventos;
- Pode ser algo que o browser fez ou o usuário;
- Botão clicado, campo field preenchido, são exemplos de eventos;
- O código JavaScript costuma ser longo, por isso, é comum ver os atributos eventos chamando funções.

JavaScript

*Eventos:

Event	Description
onchange	An HTML element has been changed
onclick	The user clicks an HTML element
onmouseover	The user moves the mouse over an HTML element
onmouseout	The user moves the mouse away from an HTML element
onkeydown	The user pushes a keyboard key
onload	The browser has finished loading the page

- https://www.w3schools.com/jsref/dom_obj_event.asp

JavaScript

× Strings:

- São usados para armazenar e manipular textos;
- Strings são zero ou mais caracteres escritos entre aspas (duplas ou não);
- Strings podem ser objetos.

JavaScript

× Arrays:

- São usados para armazenar múltiplos valores em uma variável;
- Podem armazenar variáveis de tipos diferentes;
- Os itens são separados por vírgulas, e todos eles estão entre colchetes;
- O primeiro elemento é o índice zero.

JavaScript

× Condicionais:

- São usados para tomar decisões baseado em diferente condições;
- JavaScript tem 4 declarações condicionais:
 - if
 - else
 - else if
 - switch

JavaScript

×Loops:

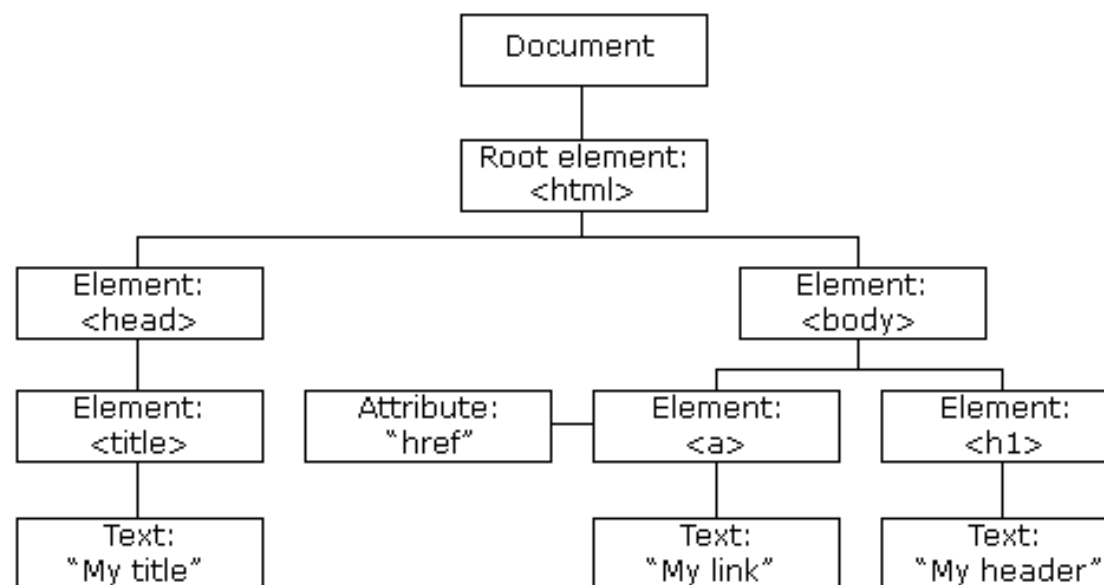
- Loops permitem executar um bloco por um numero de vezes;
- Suporta diferentes tipos de loops:
 - for;
 - for in;
 - for of;
 - while;
 - do/while.

JavaScript

× DOM:

- Document Object Model;

The HTML DOM Tree of Objects





JavaScript

×DOM:

- JavaScript pode mudar ou remover todos os elementos e atributos HTML de uma página;
- JavaScript pode alterar os elementos CSS;
- JavaScript pode criar e adicionar novos elementos HTML em uma página.
- Para um assunto mais profundo sobre DOM acesse:
https://www.w3schools.com/js/js_htmlDOM.asp

JavaScript

× REFERENCIA:

- <https://www.w3schools.com/js/default.asp>