HTML, CSS e Javascript

Estrutura básica de um documento HTML.

Utilização de tags semânticas: header, nav, section,

article, etc. Formulários e elementos interativos.

Seletores CSS e regras de estilo.

Estilização de texto, cores e imagens.

JAVASCRIPT

Posicionamento de elementos e layout responsivo.

Variáveis, tipos de dados, operadores, estruturas de controle e manipulação do DOM.

Programação assíncrona em JavaScript.

Trabalho com Promises e tratamento de operações

assíncronas. Uso do Async/await para lidar com

chamadas assíncronas de forma mais concisa.

HTML CSS JAVASCRIPT

HyperText Markup Language
(Linguagem de Marcação de Hipertexto)

Linguagem de marcação utilizada para estruturar o conteúdo e a apresentação de documentos na web.

Texto
Imagens
Links
Navegação

Estrutura semântica do conteúdo

Cascading Style Sheets
(Folhas de Estilo em Cascata)

Linguagem de estilo utilizada para controlar a apresentação e o layout dos elementos HTML

Cores
Fontes
Bordas
Tamanhos

Trata da aparência visual do conteúdo

Uma linguagem de programação completa e versátil, usada principalmente para criar interatividade e dinamismo em páginas da web

Manipulação do dom
Lida com eventos
Realizar validações
Criar animações
Comunicação com servidor
Criação de aplicações complexas

HTML

<head >

<title>

Tag

Conteúdo

As tag's servem para marcarmos o conteúdo no HTML. .

Semantica

Interação com CSS e JavaScript

Tag

Conteúdo

As tag's servem para marcarmos o conteúdo no HTML. .

Semantica

Interação com CSS e JavaScript

Meu gato é muito gordo.

Meu gato é muito gordo.

Tag

As tag's servem para marcarmos o conteúdo no HTML. .

Semantica

Interação com CSS e JavaScript

Conteúdo

Meu gato é muito gordo.

Meu gato é muito gordo.

Tags

<h1> <h2> <h3> <h4> <h5> <h6>

<a>

Estrutura do HTML

Esta declaração define o tipo de documento

<!DOCTYPE html>

Estrutura do HTML

<html> <head>

como metadados, scripts, links para estilos (CSS) e outros elementos que não são visíveis na página. É onde você define configurações importantes para a página.

Esta tag envolve todo o conteúdo da página e representa a raiz do documento HTML.

Contém informações gerais sobre o documento,

<title>

Esta tag é usada para definir o título da página.

<meta>

É usada para fornecer metadados adicionais sobre a página. Isso inclui informações como a codificação de caracteres, descrições para mecanismos de busca e outras informações relevantes.

</head>

Contém todo o conteúdo visível da página.

<body>

```
<meta charset="UTF-8">
```

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

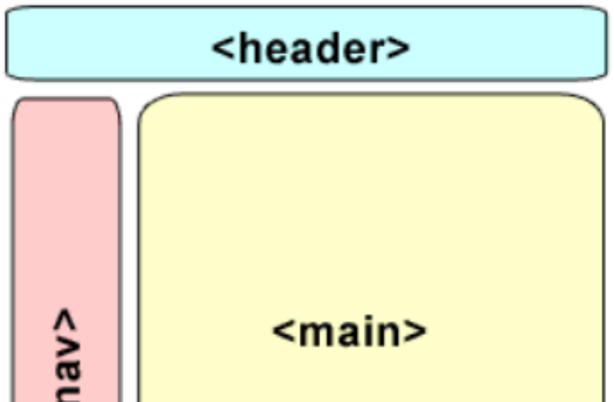
Semântica

Acessibilidade

[redacted]

<main>

<nav>



<article>

<footer>

<header>

<section>

Lista


```
<ul>
  <li> Fox</li>
  <li> Camaro</li>
  <li> Lancer</li>
</ul>
```

```
<ol>
  <li>Caio</li>
  <li>Julia</li>
  <li>Ronaldo</li>
  <li>Felipe</li>
</ol>
```


<dl>

<dt> <dd>

<dt>
<dd>

O que é CSS
    Cascading Style Sheets
O que é HTML?
    HyperText Markup Language

(São Paulo)
Pink Floyd- 20:00 - Hoje Iron
Maiden- 21:00 - Amanhã Lulu
Santos- 15:00 - Amanhã

(Rio de Janeiro)
Iron Maiden - 20:00 - Hoje
Lulu Santos - 21:00 - Hoje
Jota Quest - 15:00 - Amanhã

Formulário

<form>

<input>

<label>

for

id

```html
<form method="POST" class="formCreate">
  <h1>Cadastrar usuário</h1>
  <p>Digite os seus dados de cadastro nos campos abaixo.</p>
  <label for="email">E-mail</label>
  <input type="email" placeholder="Digite seu e-mail" autofocus="true" id="email" required />
  <label for="name">Nome</label>
  <input type="text" placeholder="Digite seu nome" id="name" required />
  <label for="password">Senha</label>
  <input type="password" placeholder="Digite sua senha" id="password" required />
  <input type="submit" value="Enviar" class="btn" />
</form>
```

EXPERIÊNCIA

Desenvolvo projetos utilizando **JavaScript**, **Java** e **Python**. Além disso, também dou aulas sobre tecnologia nos cursos de graduação.

2024 | **Faculdade Impacta**

Professor universitário

Atuação como docente em cursos de graduação, com foco em disciplinas de tecnologia e programação, abrangendo diversas linguagens e frameworks.

JS · Python · Java · React · Docker

2024 | **Media Portal**

Líder de desenvolvimento

Líder de desenvolvimento de aplicações Java utilizando arquitetura REST com Spring Boot, aplicando metodologias ágeis e práticas de TDD para garantir qualidade e eficiência no código.

Java · SpringBoot · TDD

2023 | **Facens**

Professor

Atuação como docente nos programas de Extensão e Pós-Graduação, ministrando a disciplina de Fundamentos de Desenvolvimento Web e Mobile, com foco em tecnologias emergentes e práticas avançadas.

JS · HTML · CSS · POC

Desenvolvedor Full Stack & Professor

Localizado em São Paulo

EXP

Desenvolvo projetos utilizando **JavaScript**, **Java** e **Python**. Além disso, também dou aulas sobre tecnologia nos cursos de graduação.

28/02

21/03

Cria uma relação entre um documento HTML e um arquivo de estilo CSS.

Seletor

a

red

none

Bloco css Propriedade valor

Seletores

TAGS

ID

CLASS

Color

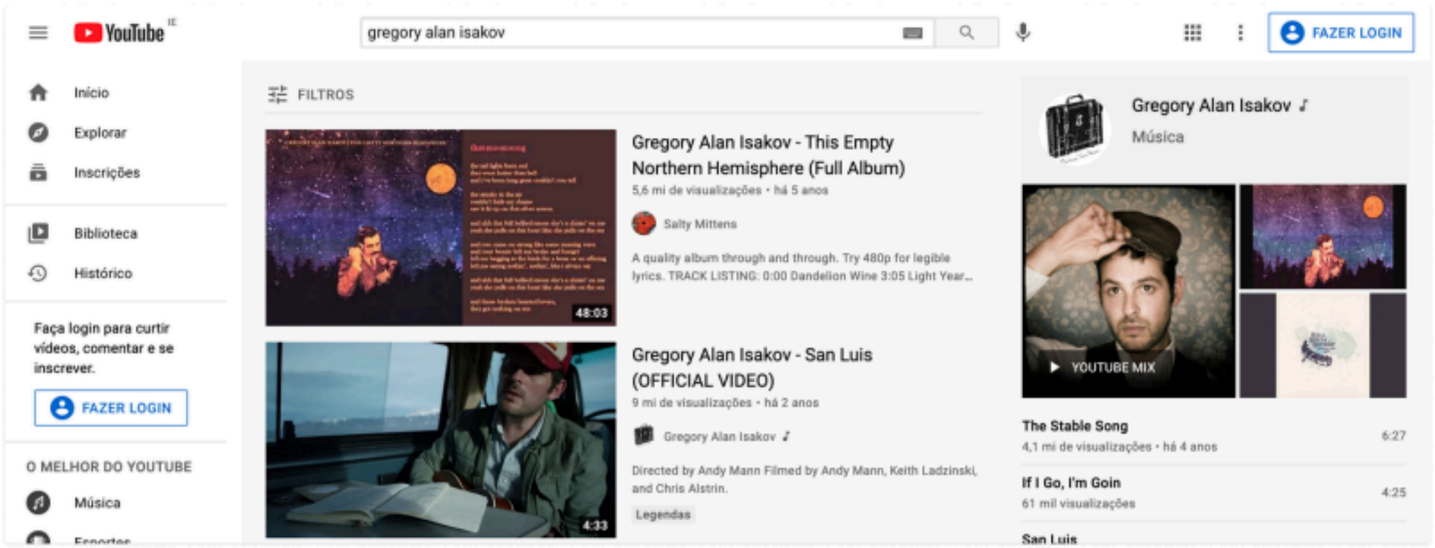
hexadecimal:

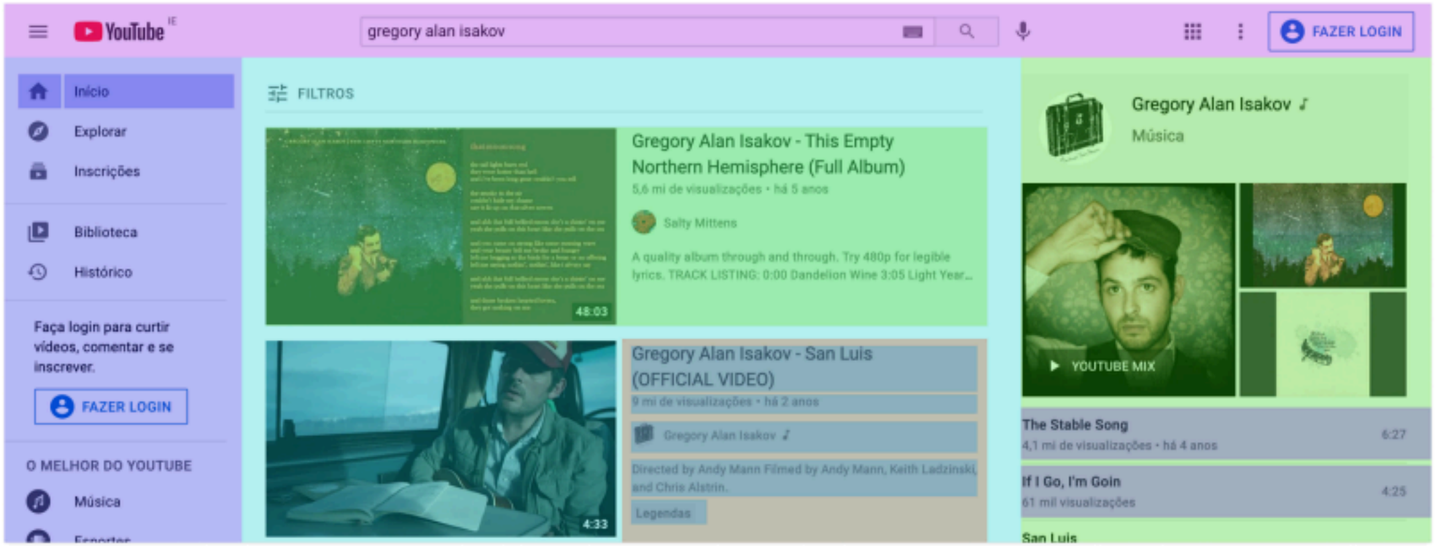
rgba(0, 0, 0, 1):

background / background-color:

color:

Box model


YouTube IE
gregory alan isakov
FAZER LOGIN
Início
Explorar
Inscrições
Biblioteca
Histórico
Faça login para curtir vídeos, comentar e se inscrever.
FAZER LOGIN
O MELHOR DO YOUTUBE
Música
FILTROS
Gregory Alan Isakov - This Empty Northern Hemisphere (Full Album)
5,6 mi de visualizações • há 5 anos
Salty Mittens
A quality album through and through. Try 480p for legible lyrics. TRACK LISTING: 0:00 Dandelion Wine 3:05 Light Year...
48:03
Gregory Alan Isakov - San Luis (OFFICIAL VIDEO)
9 mi de visualizações • há 2 anos
Gregory Alan Isakov
Directed by Andy Mann Filmed by Andy Mann, Keith Ladzinski, and Chris Alstrin.
Legendas
4:33
Gregory Alan Isakov
Música
YOUTUBE MIX
The Stable Song
4,1 mi de visualizações • há 4 anos
6:27
If I Go, I'm Goin
61 mil visualizações
4:25
San Luis


YouTube IE
gregory alan isakov
FAZER LOGIN
Início
Explorar
Inscrições
Biblioteca
Histórico
Faça login para curtir vídeos, comentar e se inscrever.
FAZER LOGIN
O MELHOR DO YOUTUBE
Música
FILTROS
Gregory Alan Isakov - This Empty Northern Hemisphere (Full Album)
5,6 mi de visualizações • há 5 anos
Salty Mittens
A quality album through and through. Try 480p for legible lyrics. TRACK LISTING: 0:00 Dandelion Wine 3:05 Light Year...
48:03
Gregory Alan Isakov - San Luis (OFFICIAL VIDEO)
9 mi de visualizações • há 2 anos
Gregory Alan Isakov
Directed by Andy Mann Filmed by Andy Mann, Keith Ladzinski, and Chris Alstrin.
Legendas
4:33
Gregory Alan Isakov
Música
YOUTUBE MIX
The Stable Song
4,1 mi de visualizações • há 4 anos
6:27
If I Go, I'm Goin
61 mil visualizações
4:25
San Luis

Box model

Content (conteúdo):

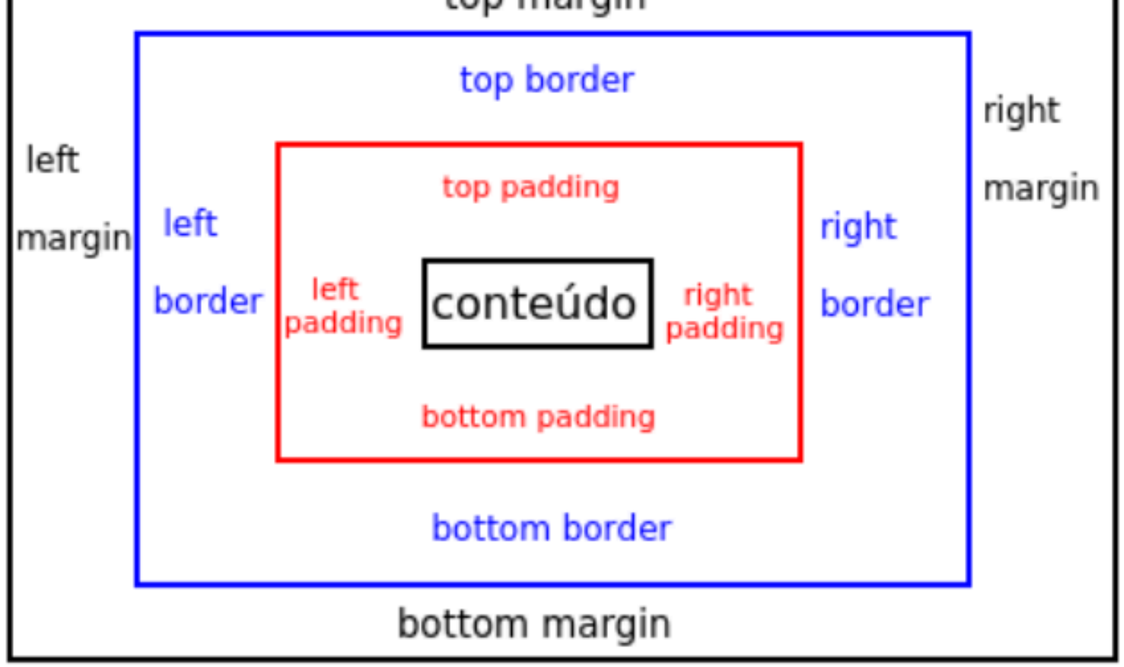
Padding (preenchimento):

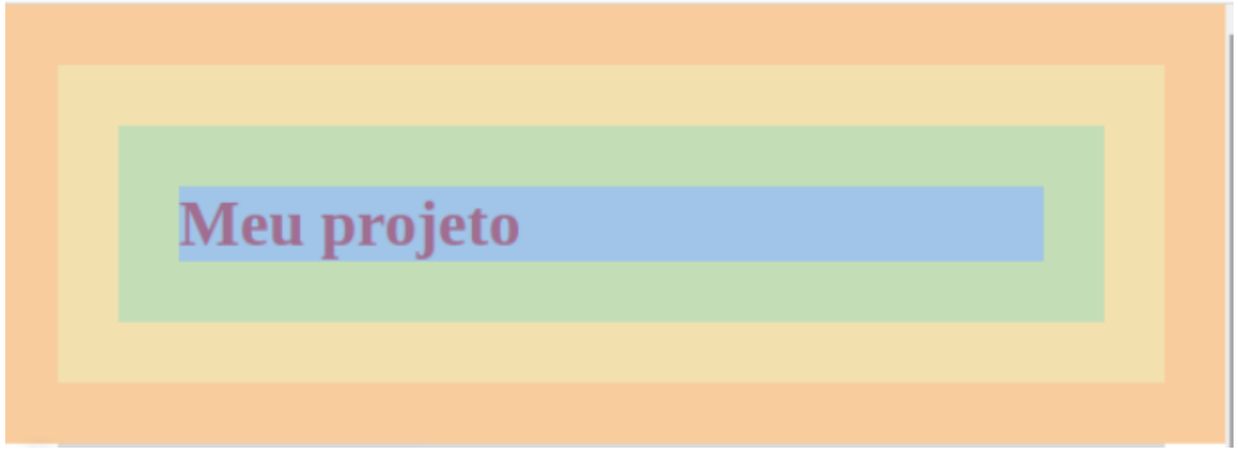
Border (borda):

Margin (margem):

Width (largura):

Height (altura):



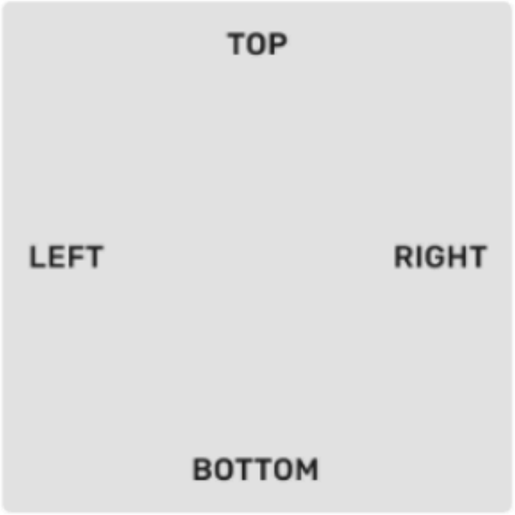


Posicionamento

```
  padding: 20px 40px;
  margin: 30px 100px 60x 200px;
}
```

Top, Right, Bottom, Left

```
div {
  padding: top/bottom right/left;
  margin: top right bottom left;
}
```



Posicionamento

block

inline

width, height, margin (top/bottom)

Posicionamento

display: grid;

grid-template-columns:

fr:

Posicionamento

align-content:

justify-content:

place-content:

align-items:

justify-items:

place-items:

Posicionamento

display: flex:

flex-wrap:

gap:

Posicionamento

position: fixed:

position: relative:

top, right, bottom e left

position: absolute:

z-index:

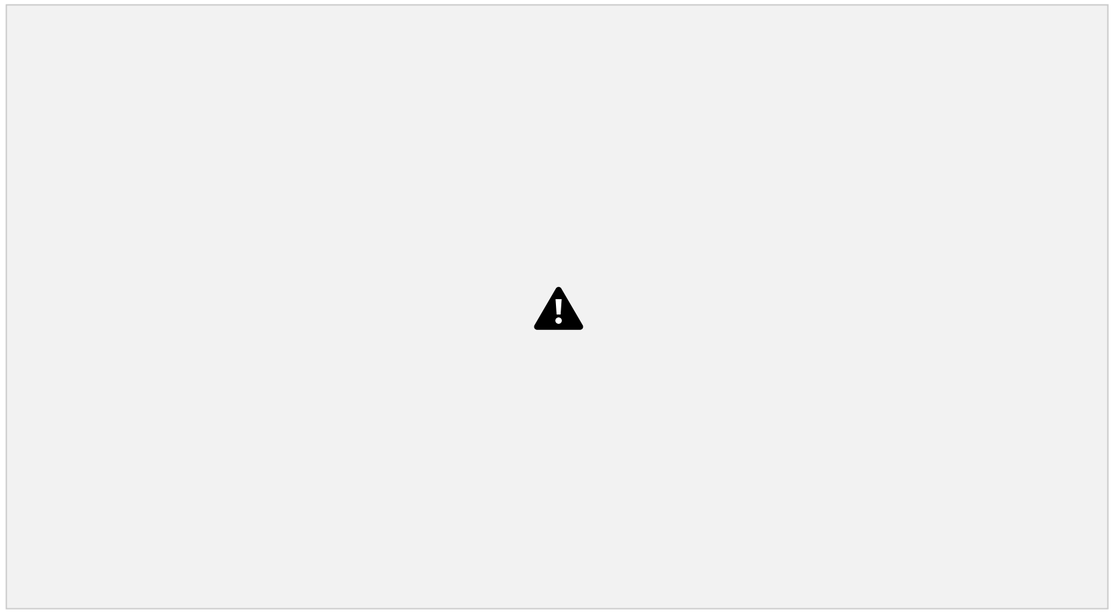
Responsivo

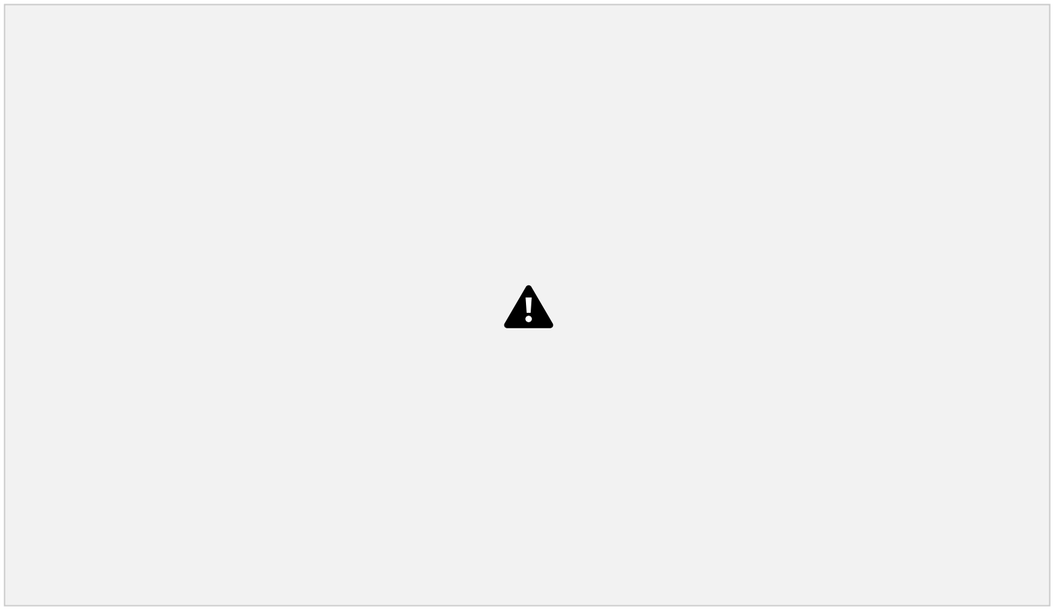
@media: @media

Documentação

Mozilla

W3C





28/02

21/03

CODIFICAR ONLINE

CODIFICAR COMPUTADOR

A LINGUAGEM JS

Linguagem de programação interpretada.

Front end

Manipulação do DOM, comunicação assíncrona com o back end e mais

Back end

Comunicação com banco de dados, manipulação de arquivos e mais (Node)

Jogos

Geralmente se aproveitando de benefícios do HTML5

SINTAXE BÁSICA

Variáveis

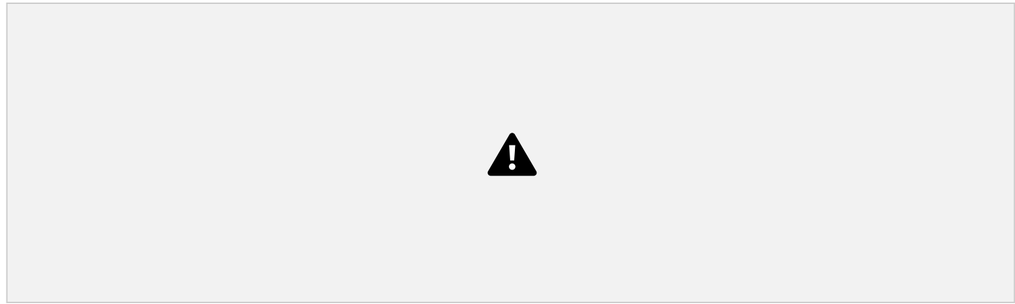
Responsáveis por guardar dados na memória. Inicia com a palavra var, let ou const

SINTAXE: Palavra chave var seguida do nome, sinal de igual e o valor.

Os nomes podem iniciar com $, _, ou letras.

Podem conter números mas não iniciar com eles

Case sensitive: nome é diferente de Nome

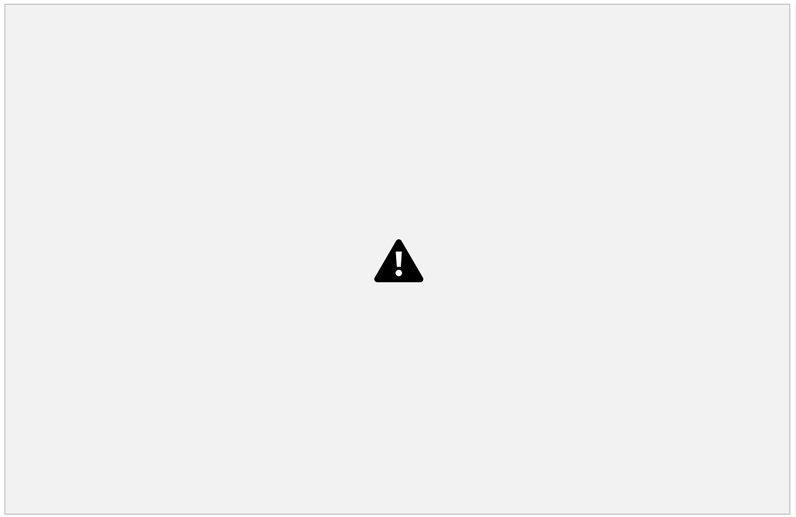


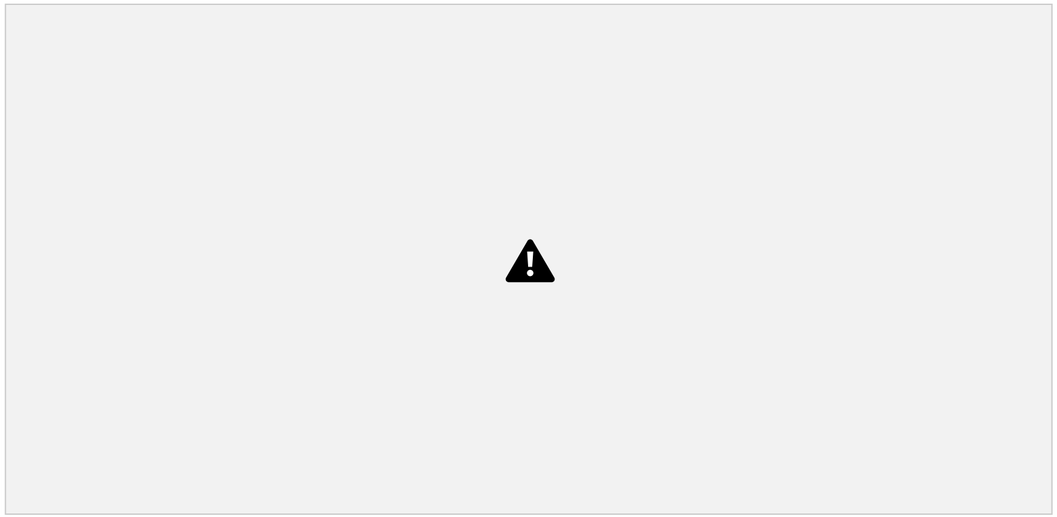
Não utilizar palavras reservadas: https://www.w3schools.com/js/js_reserved.asp

Camel case: É comum nomearmos assim: abrirModal

Variáveis

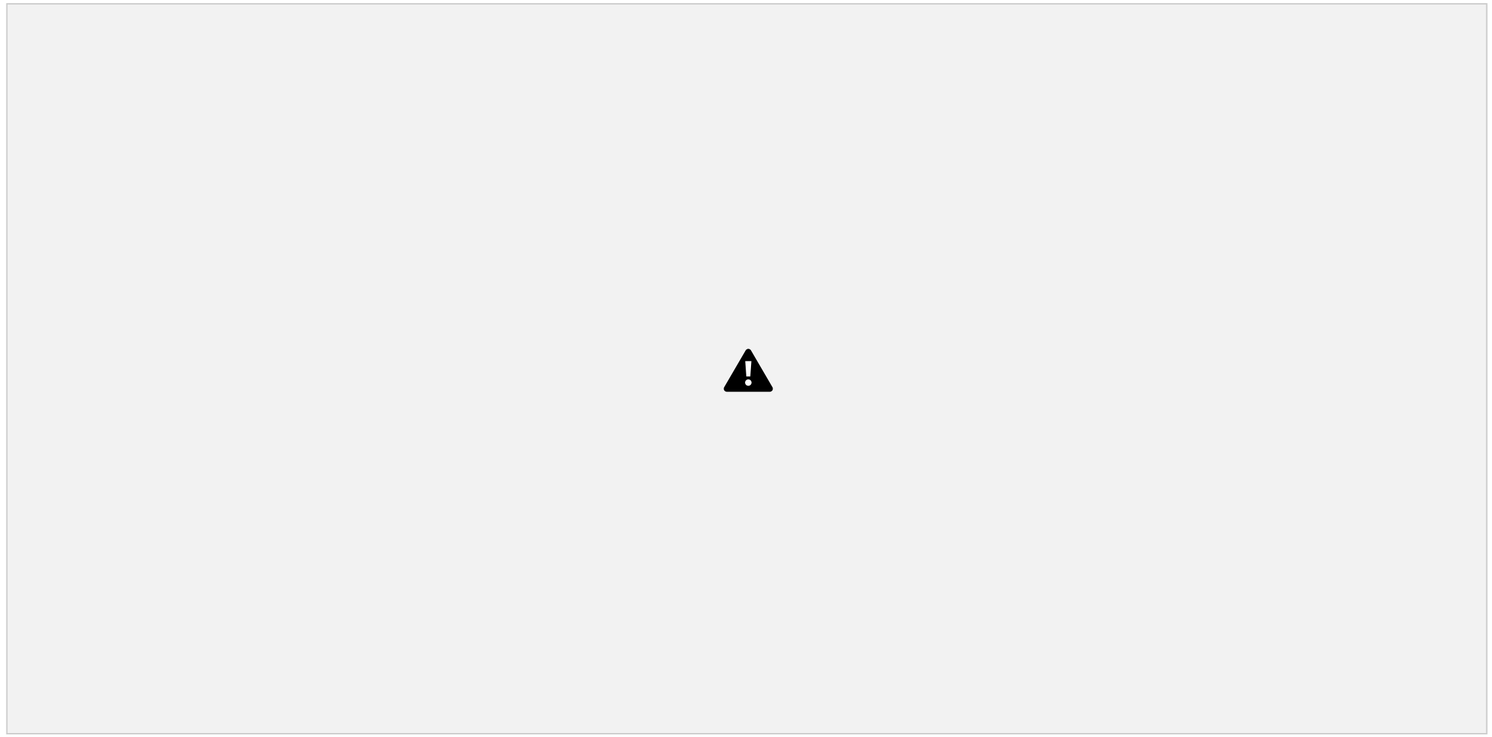
HOISTING: São movidas para cima do código, porém o valor atribuído não é movido.





MUDAR O VALOR ATRIBUÍDO: É possível mudar os valores atribuídos a variáveis declaradas com var e let. Porém não é possível modificar valores das declaradas com const

Tipos de dados

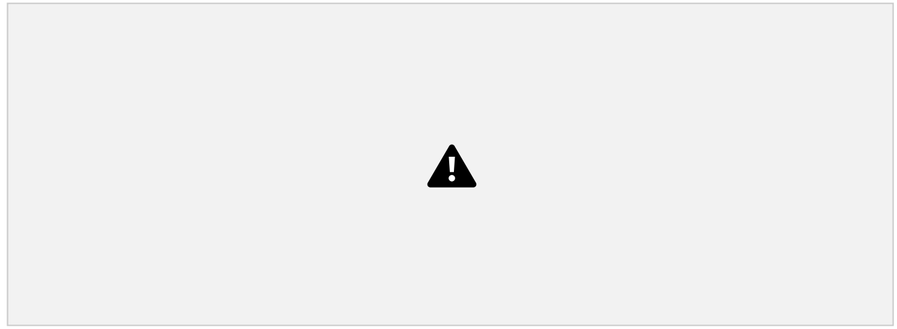


undefined é usado para indicar que uma variável foi declarada, mas ainda não foi atribuído um valor.

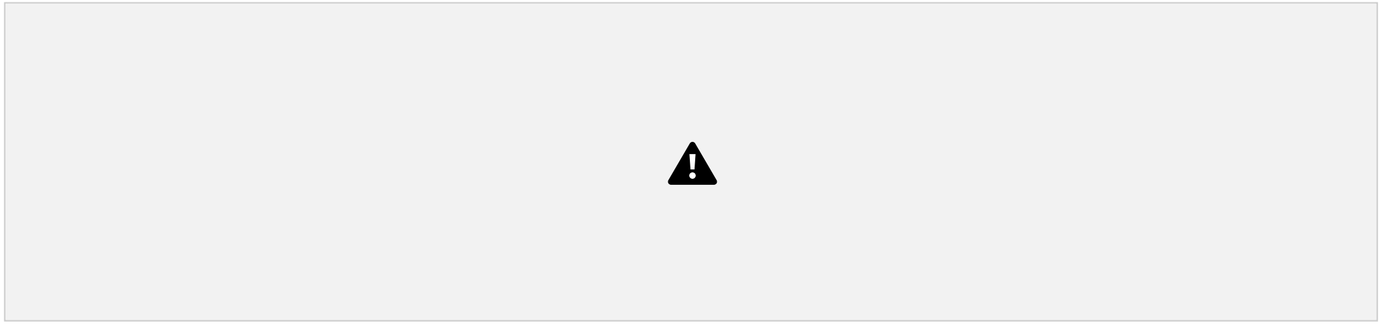
null é usado para indicar explicitamente a ausência de valor ou para definir uma propriedade como não possuindo um valor válido.

Tipos de dados

VERIFICAR TIPO DE DADO

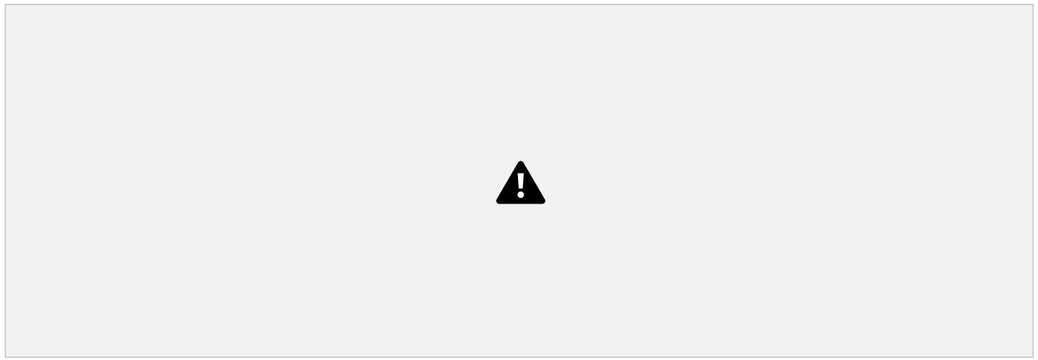


STRING

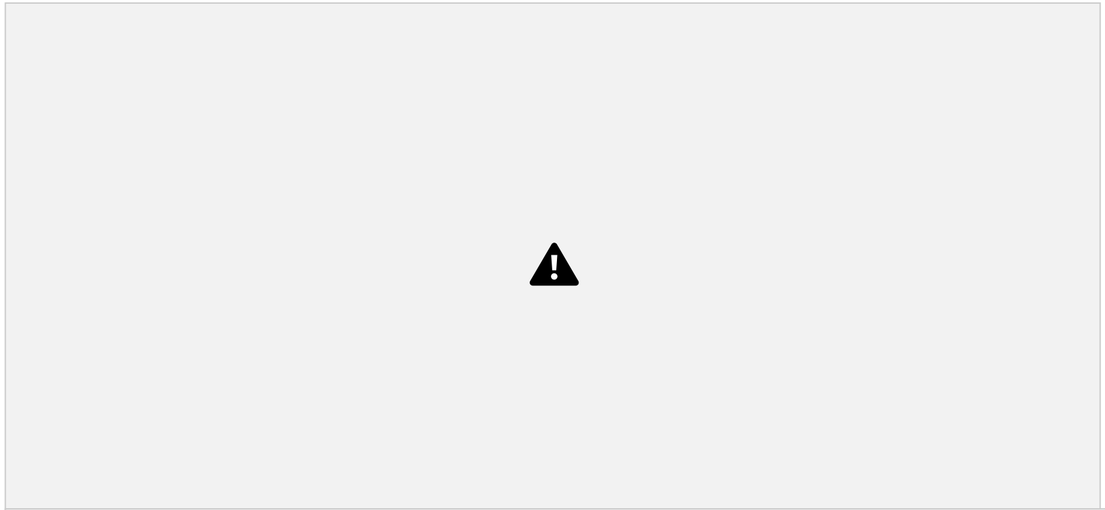


Números e Operadores

Em JavaScript, números são representados como valores numéricos de ponto flutuante de 64 bits, seguindo o padrão IEEE 754.



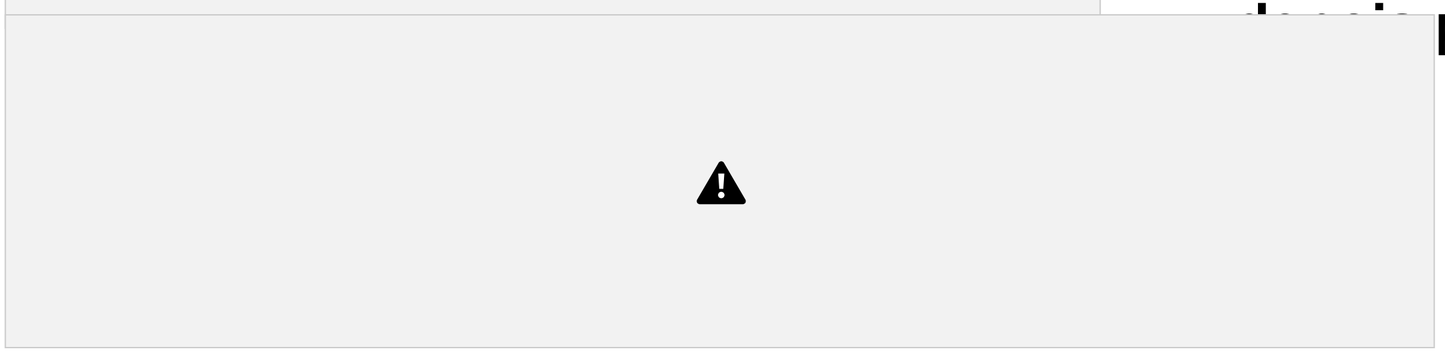
Números e Operadores



Lembrando que soma + em Strings serve para concatenar

A ordem importa

Começa por multiplicação e divisão, depois por soma e subtração.

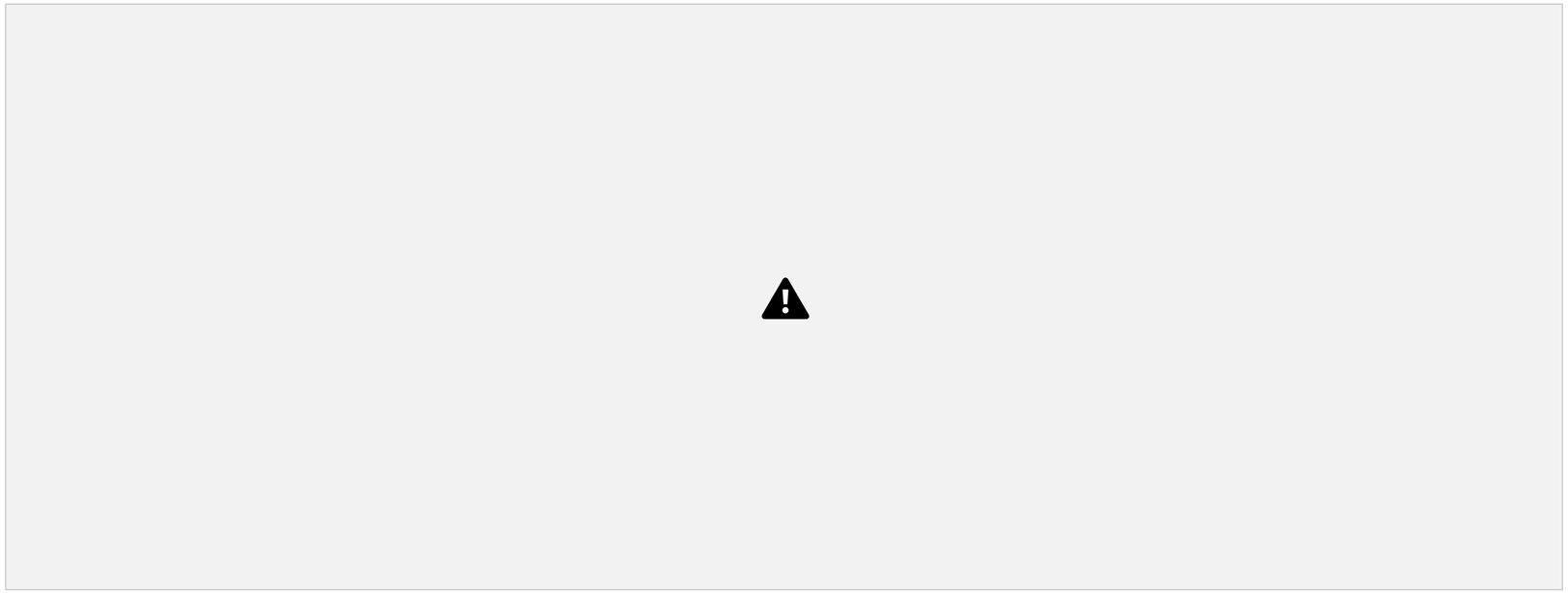


É possível verificar se uma variável é NaN ou não com a função isNaN()

Precisão

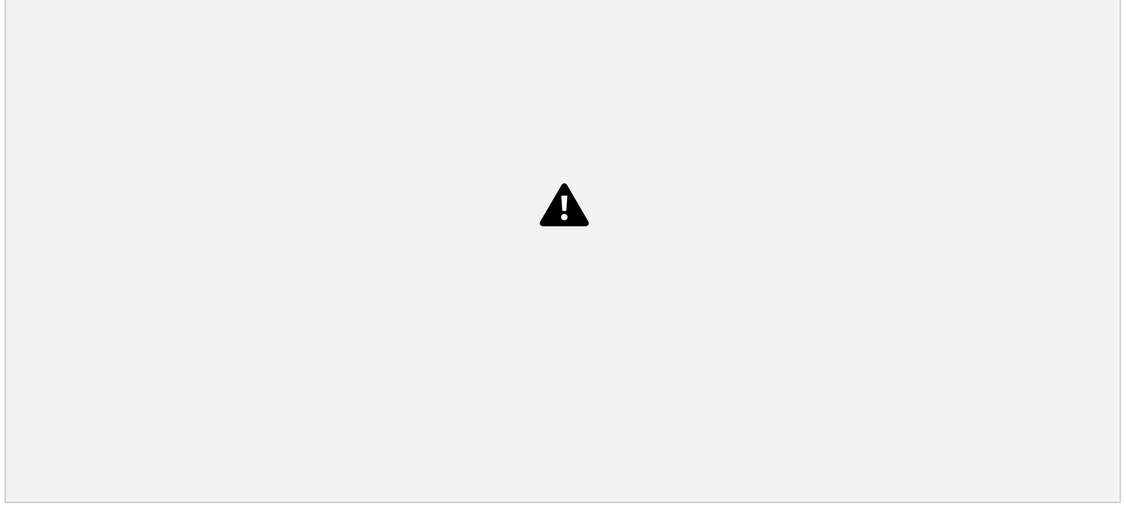
Devido à representação de números de ponto flutuante, operações matemáticas em JavaScript podem resultar em erros

de



arredondamento. Isso significa que nem todos os números podem ser representados com precisão perfeita. Por exemplo, 0.1 + 0.2 não é exatamente igual a 0.3 em JavaScript.

Operadores Aritméticos Unários



Em programação, unário é um termo que se refere a operadores ou expressões que envolvem apenas um único operando. Isso significa que eles operam em apenas um valor ou variável.

Mesma coisa para o decremento --x

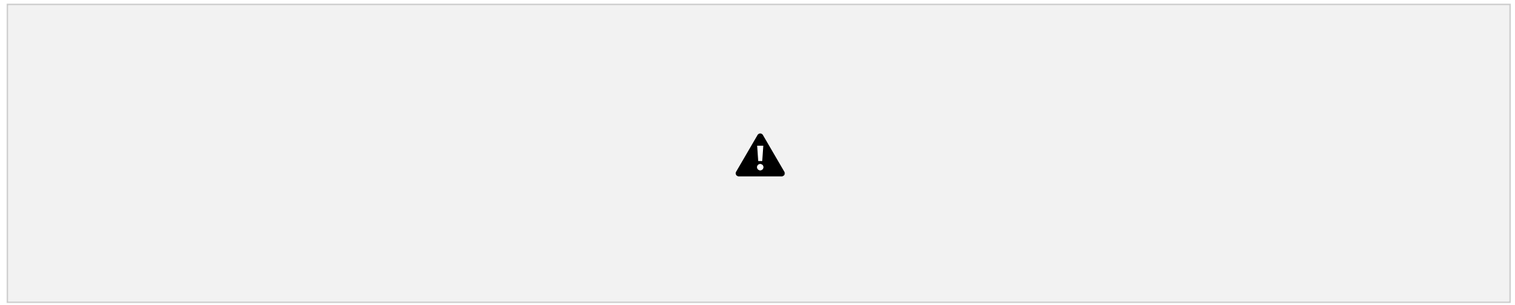
Unários

O + e - tenta transformar o valor seguinte em númer

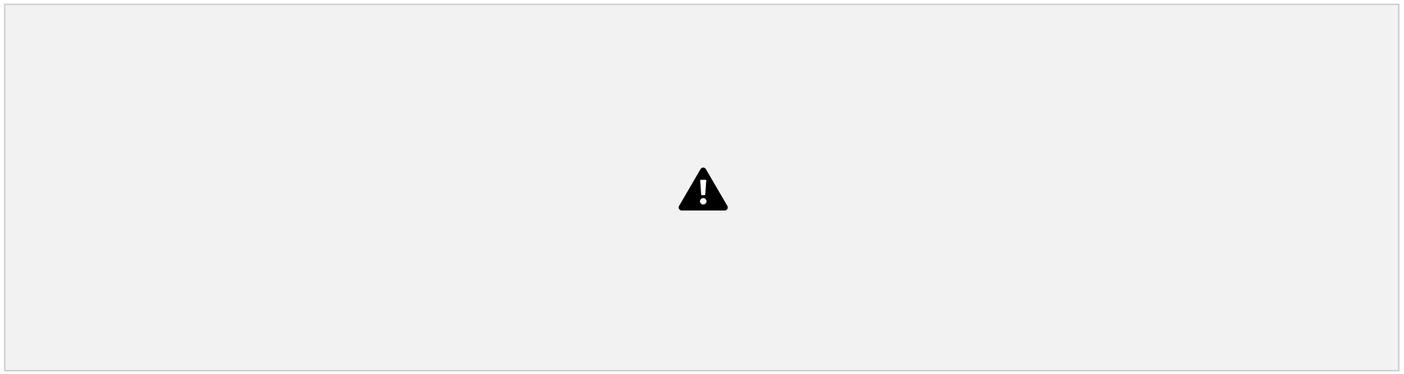
O - antes de um número torna ele negativo

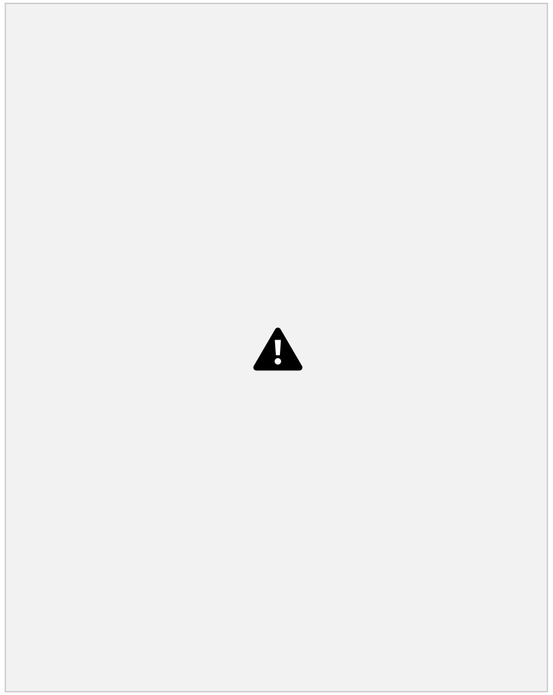
Exercícios

1 - Como dividir o peso por 2



NAN = NOT A NUMBER

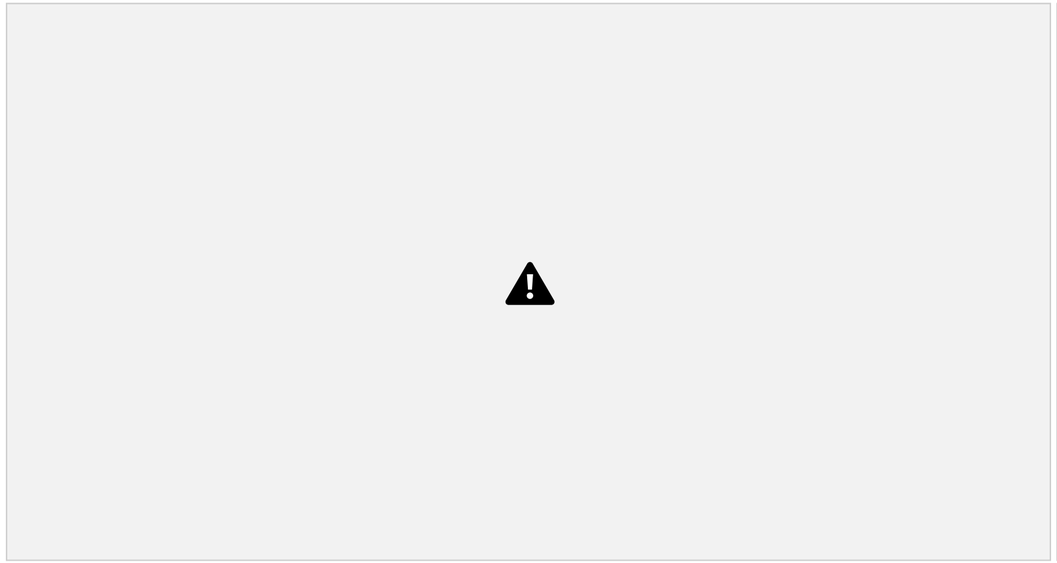


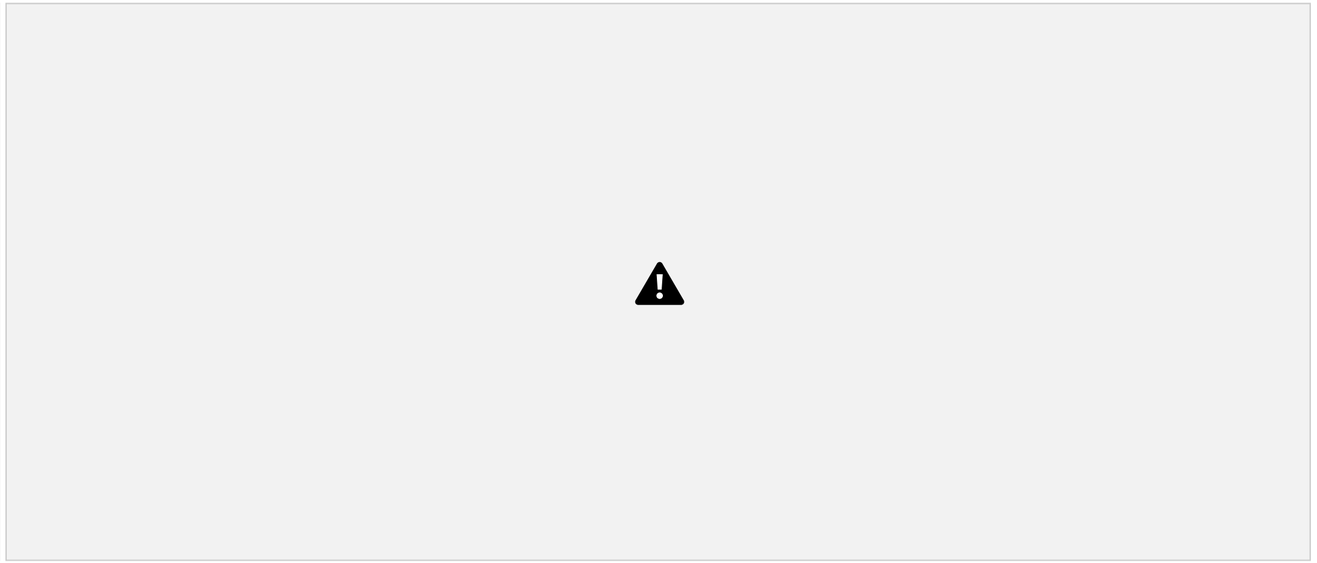


A expressão isNaN(true) e IsNaN(false) retorna false em JavaScript. Isso ocorre porque o valor booleano true e false são implicitamente convertidos para o número 1 e 0 respectivamente, antes de ser avaliado pela função isNaN().

Função

Bloco de código que pode ser executado e reutilizado. Valores podem ser passados por uma função e a mesma retorna outro valor.





criar

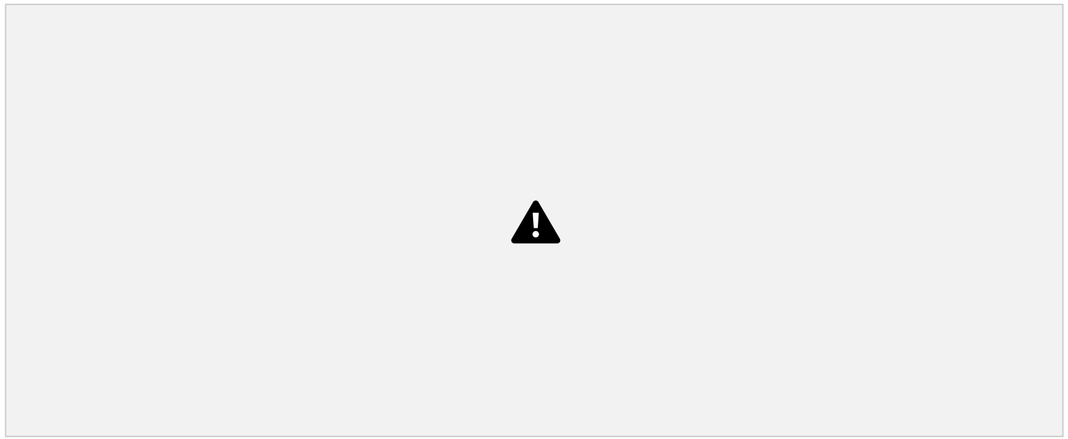
parâmetros

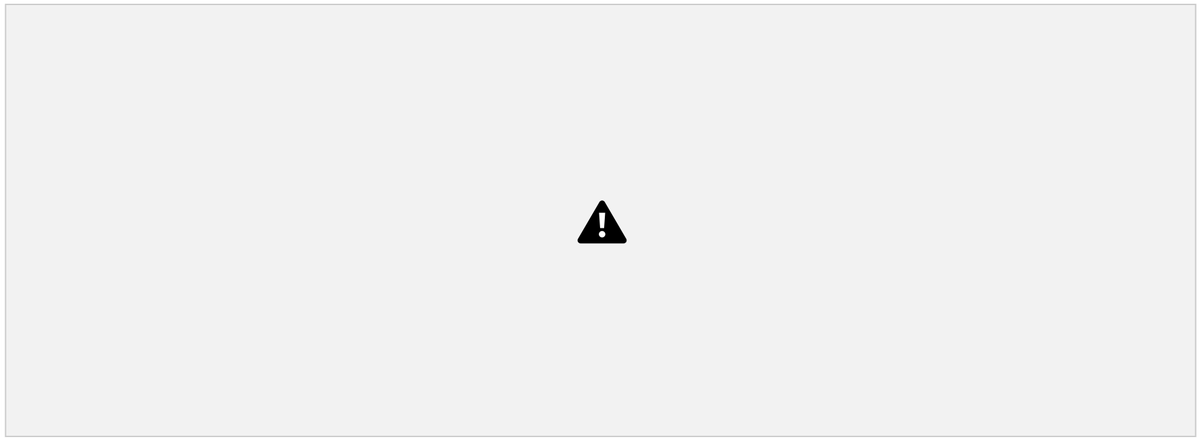
executar

argumentos

Função

Callback, geralmente são funções que ocorrem após algum evento.





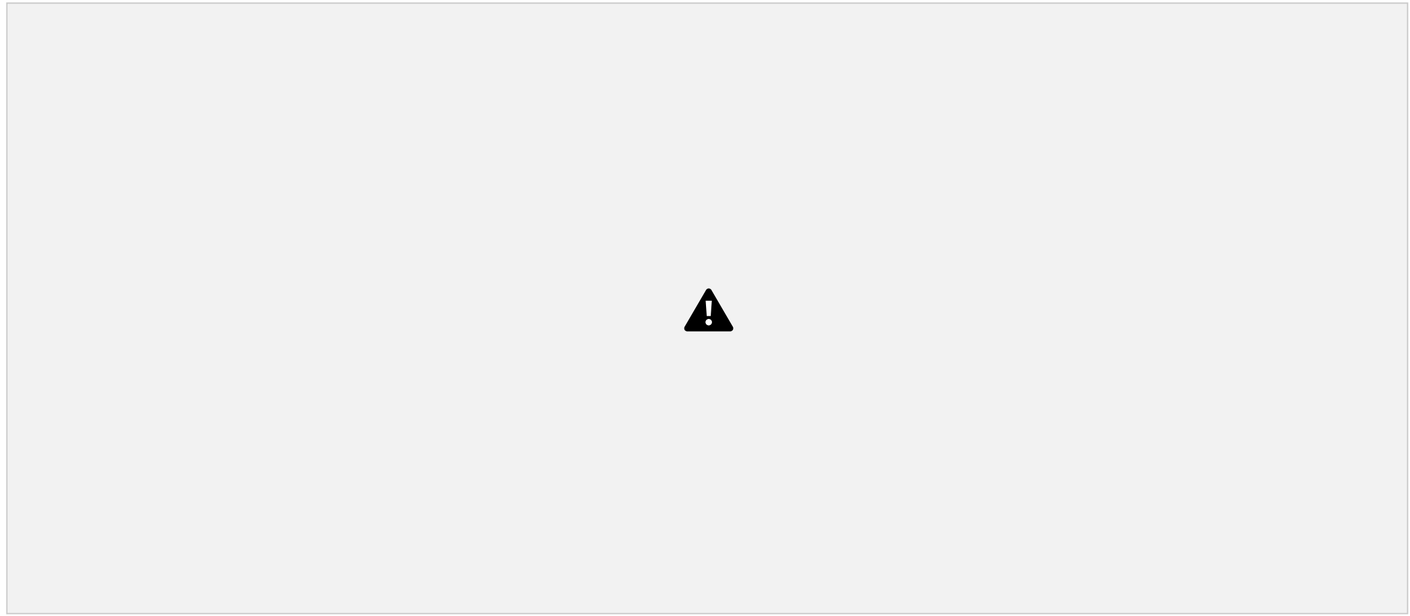
function() {} () => {}

ESCOPO: Variáveis e funções definidas dentro de um bloco { }, não são visíveis fora dele.

var, let e const

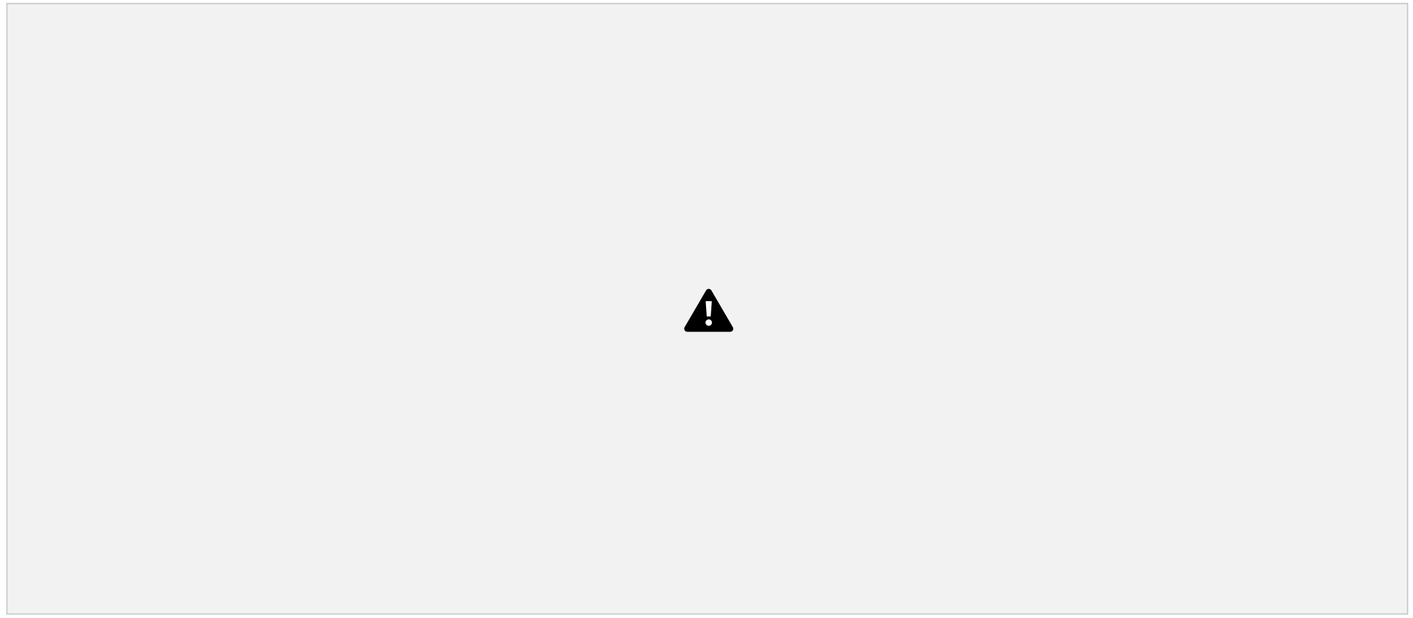
As variáveis declaradas com var têm escopo de função, o que significa que elas

são visíveis em toda a função em que foram declaradas. Se você declarar uma variável var dentro de um bloco (como um loop for), ela ainda será visível fora desse bloco.



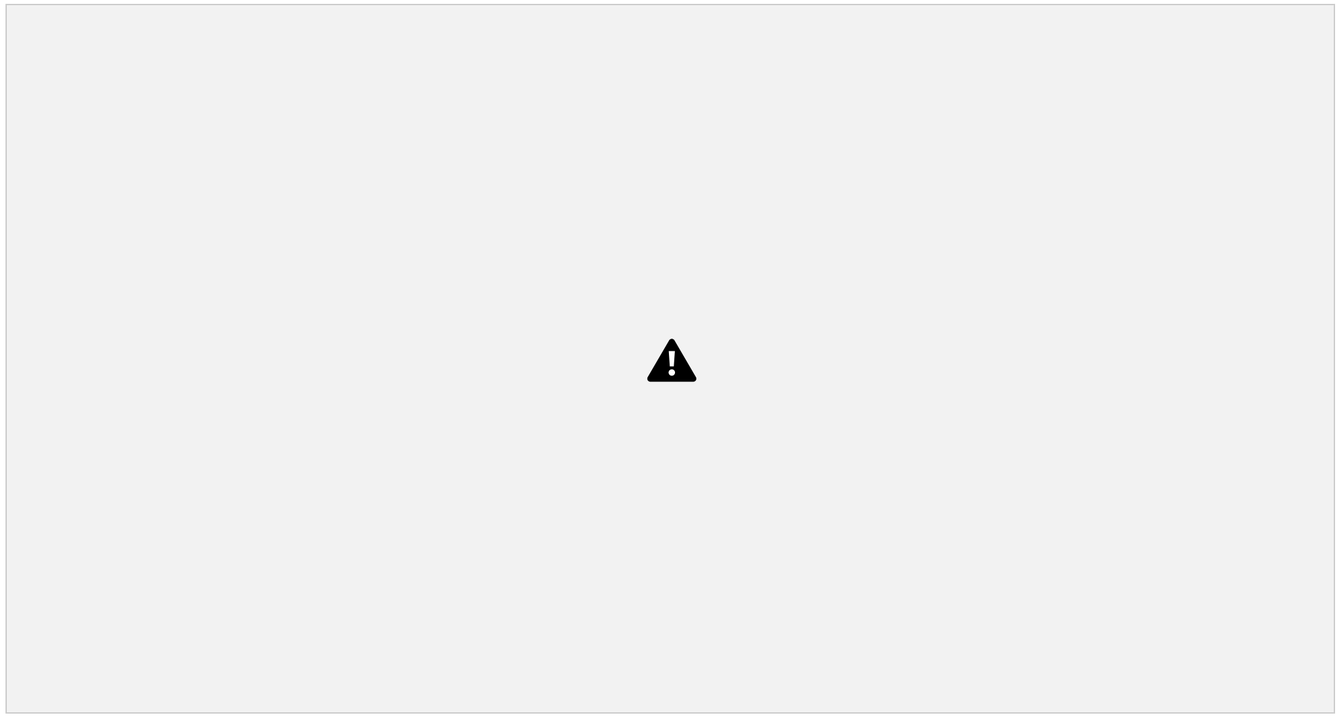
var, let e const

As variáveis declaradas com let têm escopo de bloco, o que significa que elas são visíveis apenas dentro do bloco em que foram declaradas.

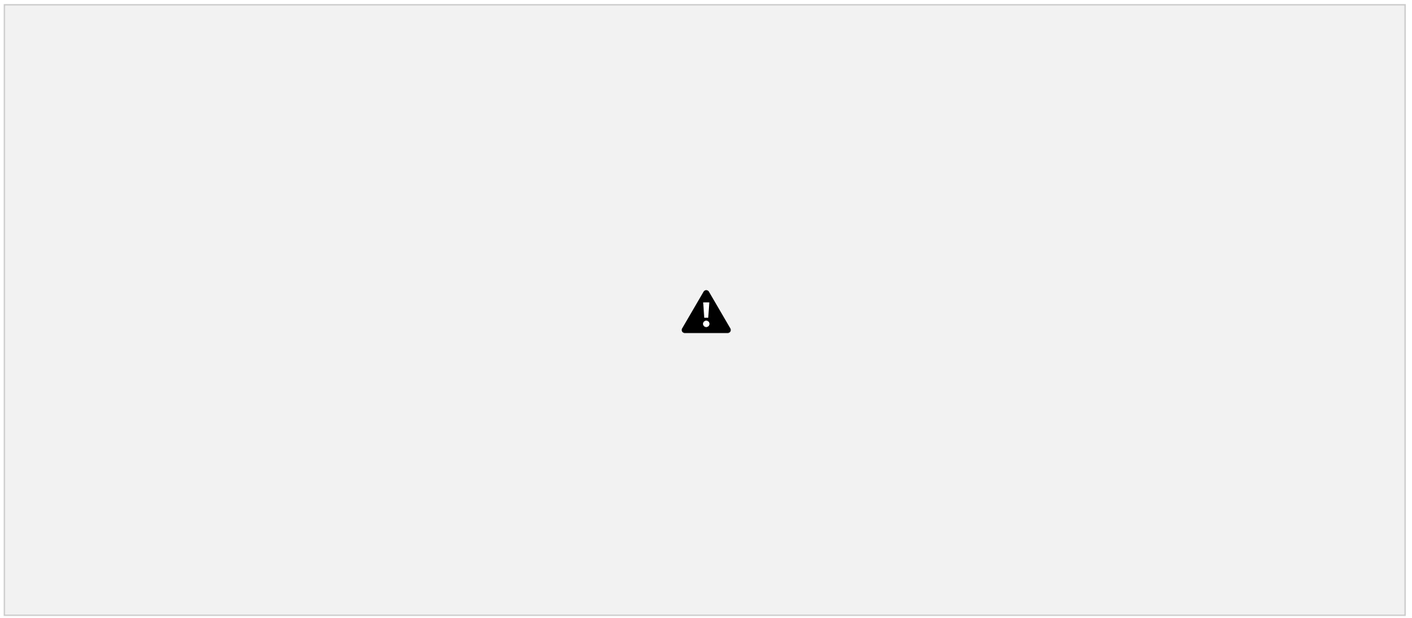


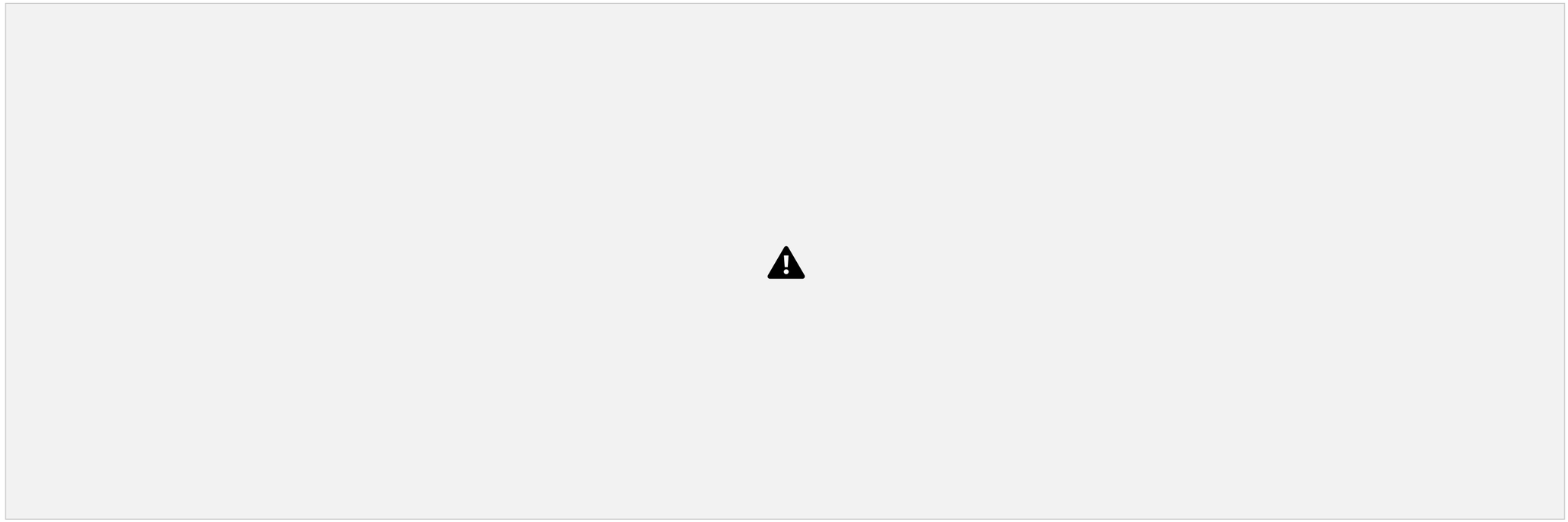
var, let e const

As variáveis declaradas com const são constantes, ou seja, uma vez que você atribui um valor a elas, esse valor não pode ser alterado posteriormente. Além disso, assim como let, as variáveis declaradas com const também têm escopo de bloco.

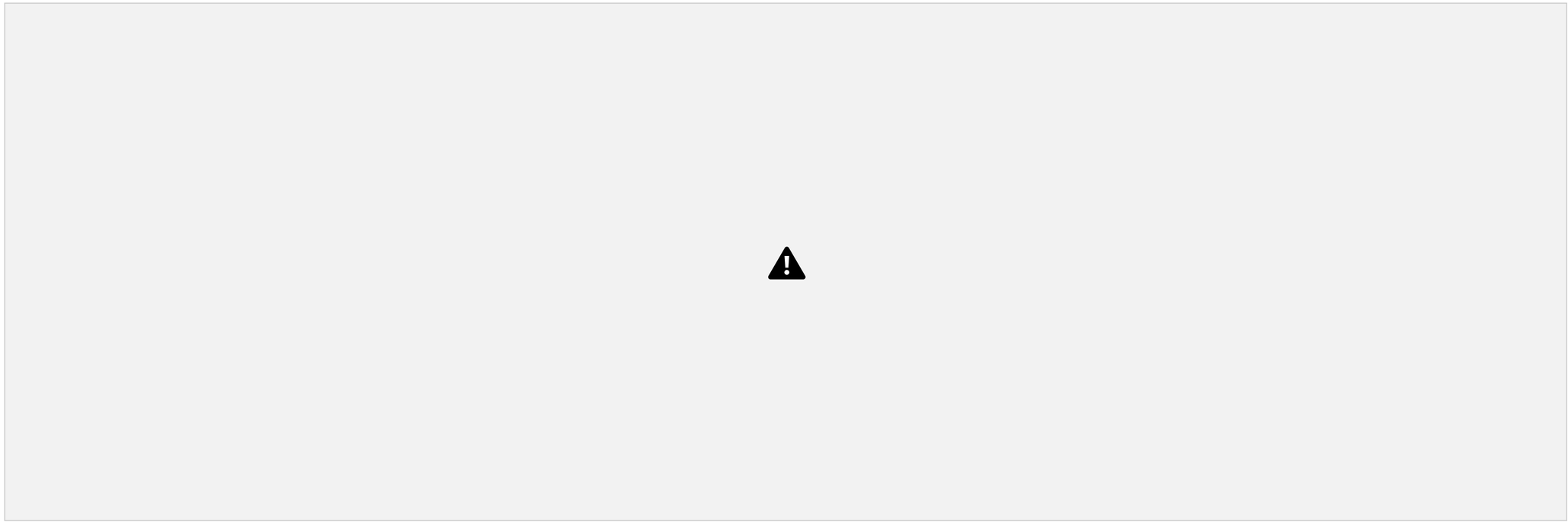


Exercício





Exercício



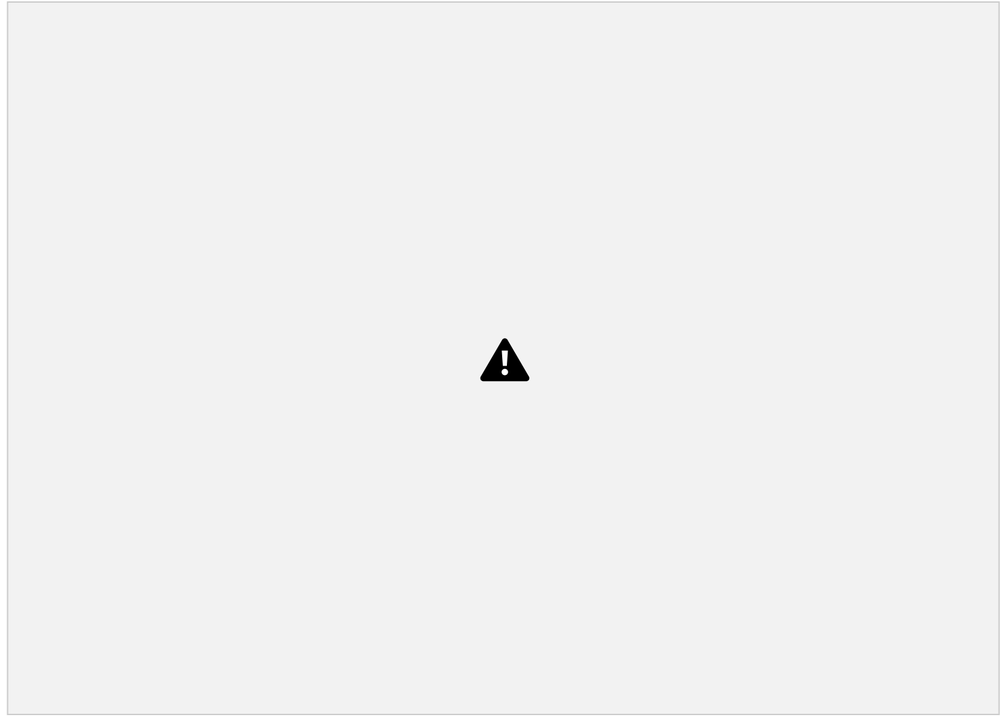
Objeto

Conjunto de variáveis e funções, que são chamadas de propriedades e métodos.

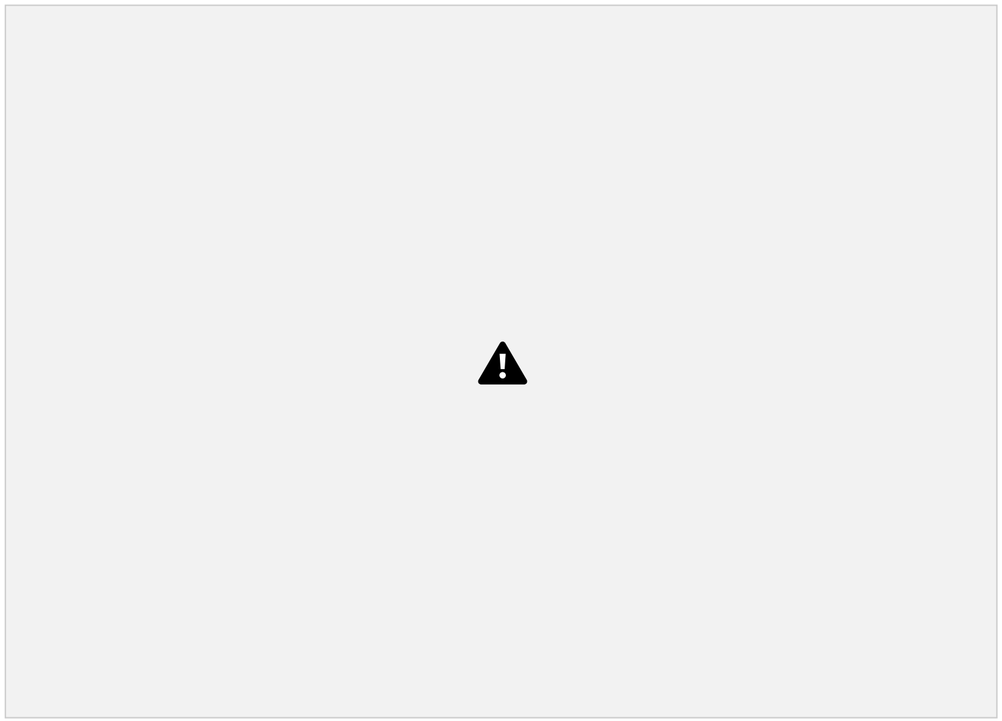
Propriedades: As propriedades em um objeto são os valores associados a chaves específicas. No objeto

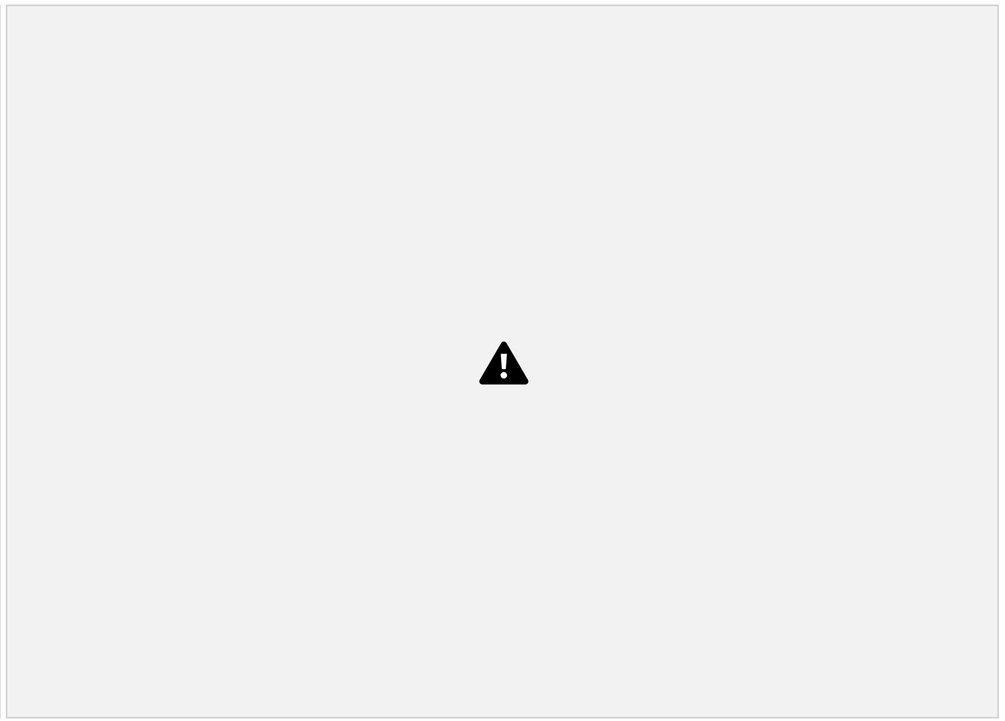
quadrado, por exemplo, temos as propriedades lados. Cada propriedade tem um nome (chave) e um valor. No caso de quadrado, lados é uma propriedade com o valor 4.

Métodos: Métodos são funções que estão associadas a um objeto. No exemplo quadrado, temos os métodos area e perimetro. Esses métodos são funções que podem ser chamadas usando a sintaxe nomeDoObjeto.nomeDoMetodo(). Os métodos podem receber parâmetros e executar ações relacionadas ao objeto em que estão definidos.



Objeto

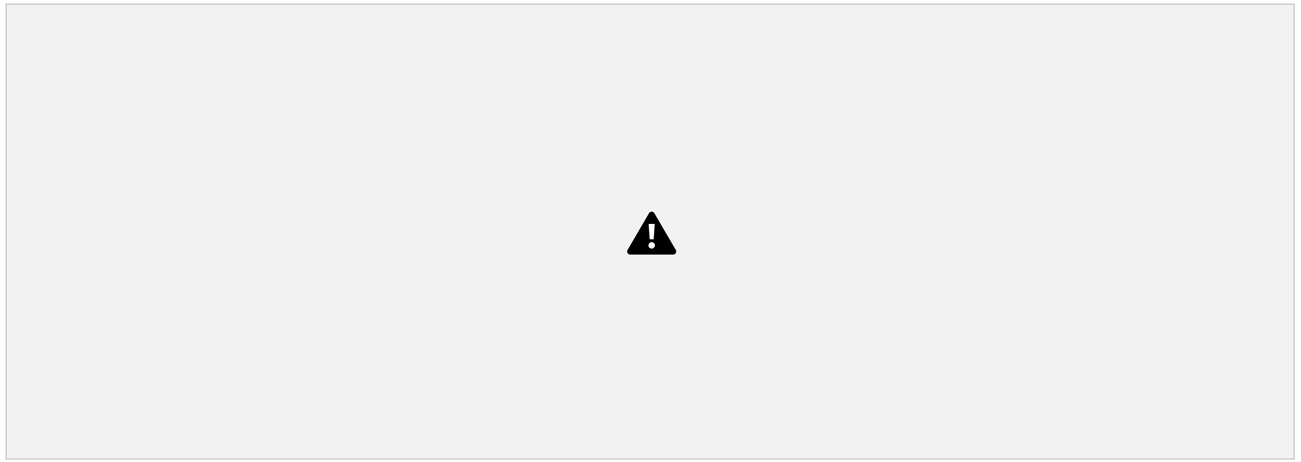




Array

É um grupo de valores geralmente relacionados. Servem para guardarmos diferentes valores em uma única variável.

MÉTODOS E PROPRIEDADES DE UMA ARRAY



E o DOM?

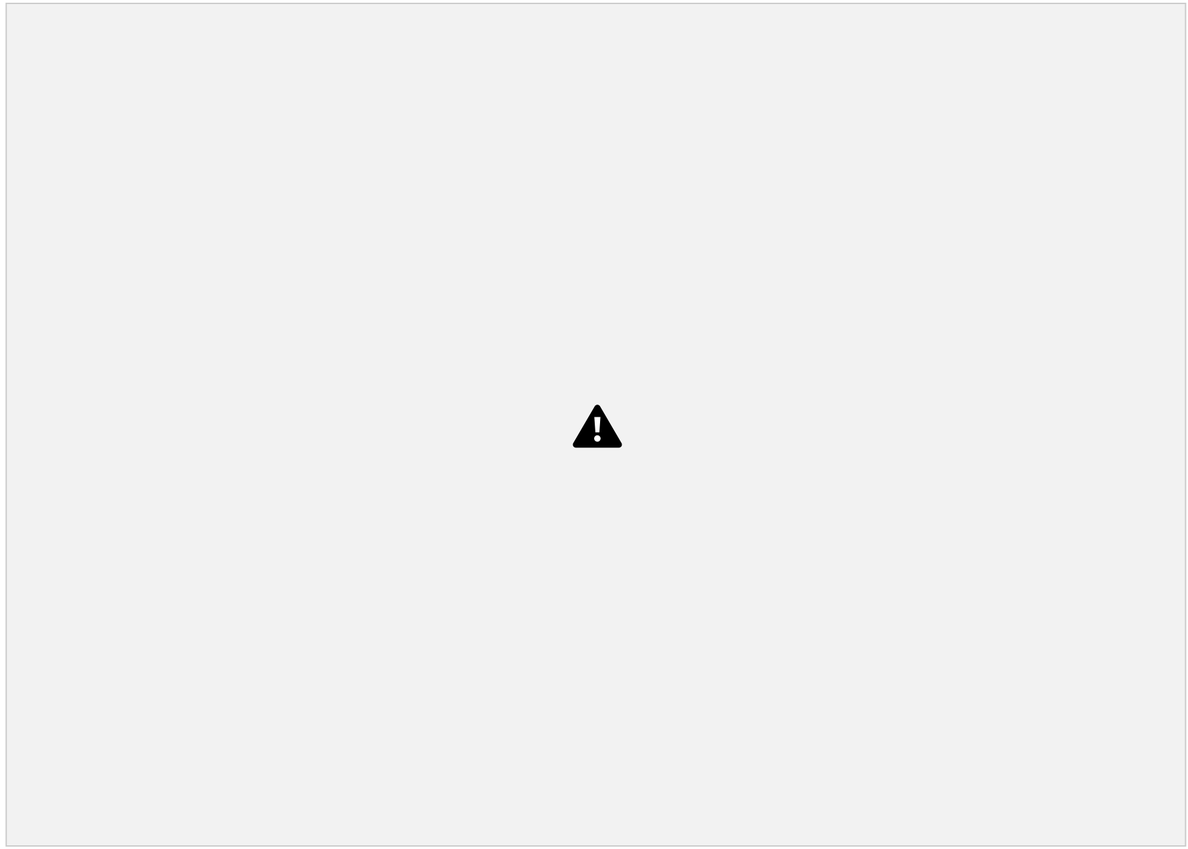
DOCUMENT OBJECT MODEL (DOM)

É uma interface que representa documentos HTML através de objetos. Com ela é possível manipular a estrutura, estilo e conteúdo destes documentos.

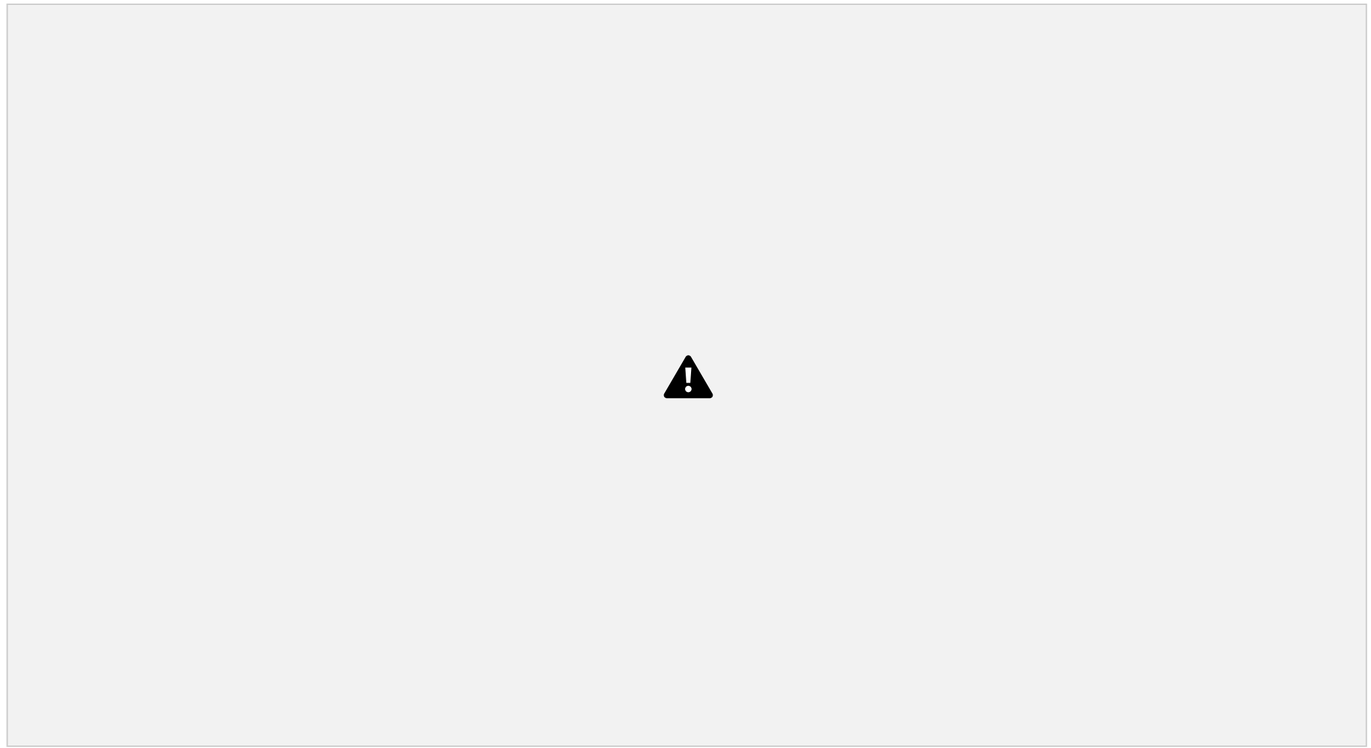
Quando uma página é carregada o navegador cria um modelo de objetos que representa a página (DOM).

O DOM é construído como uma árvore de objetos

Lembre-se que o HTML é basicamente um conjunto de marcadores aninhados.
Um marcador pode ter marcadores, que pode ter marcadores...



DOCUMENT OBJECT MODEL (DOM)



Porque estudar o DOM?

DOCUMENT OBJECT MODEL (DOM)

JS consegue recuperar a estrutura da página em um objeto.

–Ou seja, temos toda a estrutura da página em objetos dentro do JS.

- **Assim, podemos manipular nossas páginas utilizando JS:** –Mudar o conteúdo de elementos HTML.

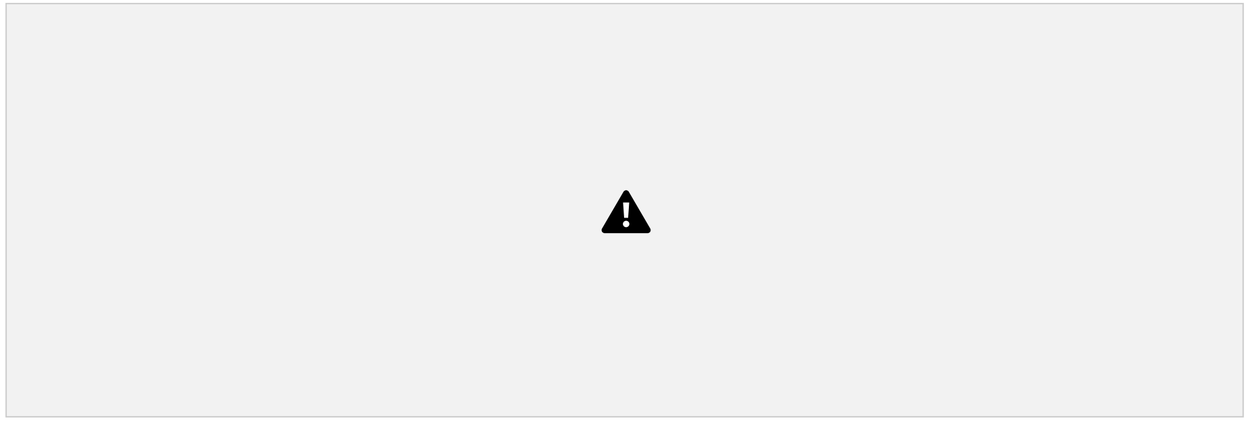
–Mudar o estilo de elementos HTML.

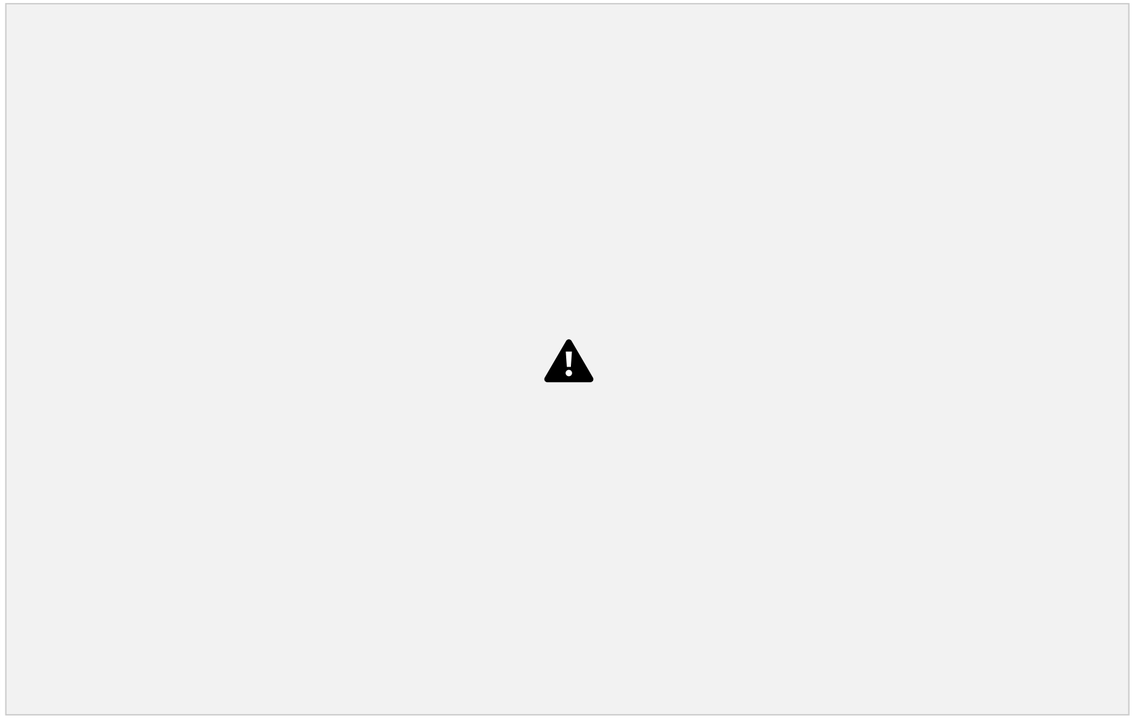
–Trabalhar com eventos.

–Adicionar e remover elementos HTML.

WINDOW E DOCUMENT

São os objetos principais do DOM, boa parte da manipulação é feita através dos seus métodos e propriedades.





Acessando o HTML no JS

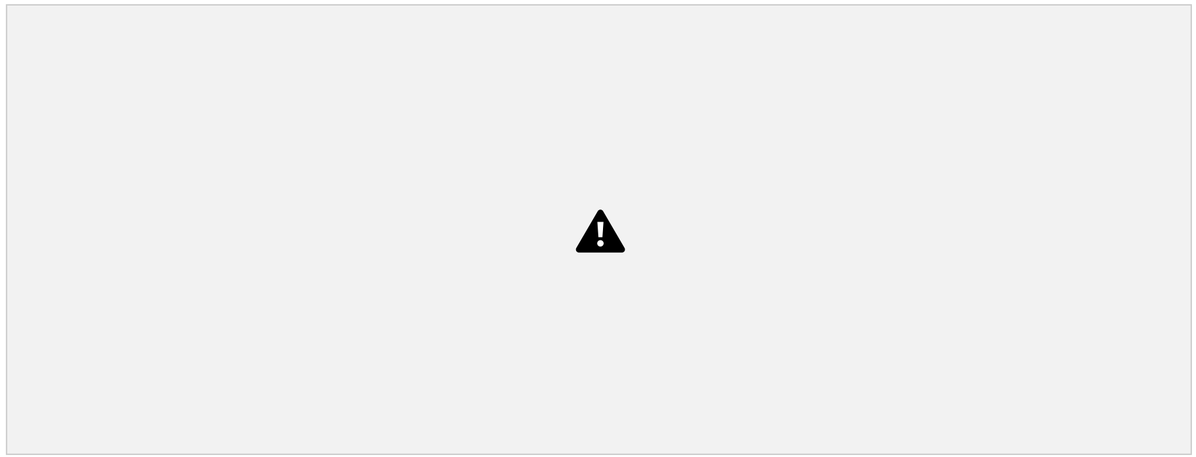
Uma das grandes funcionalidades de JS é manipular a página HTML,

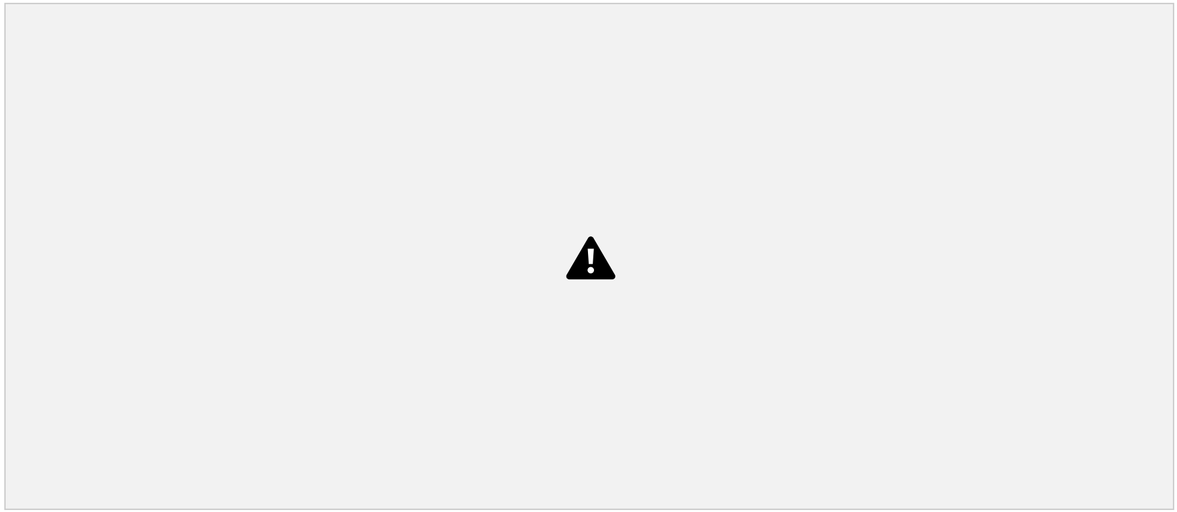
para deixar as páginas mais dinâmicas.

A primeira coisa que precisamos fazer é encontrar um elemento do HTML

ID

getElementById seleciona e retorna elementos do DOM



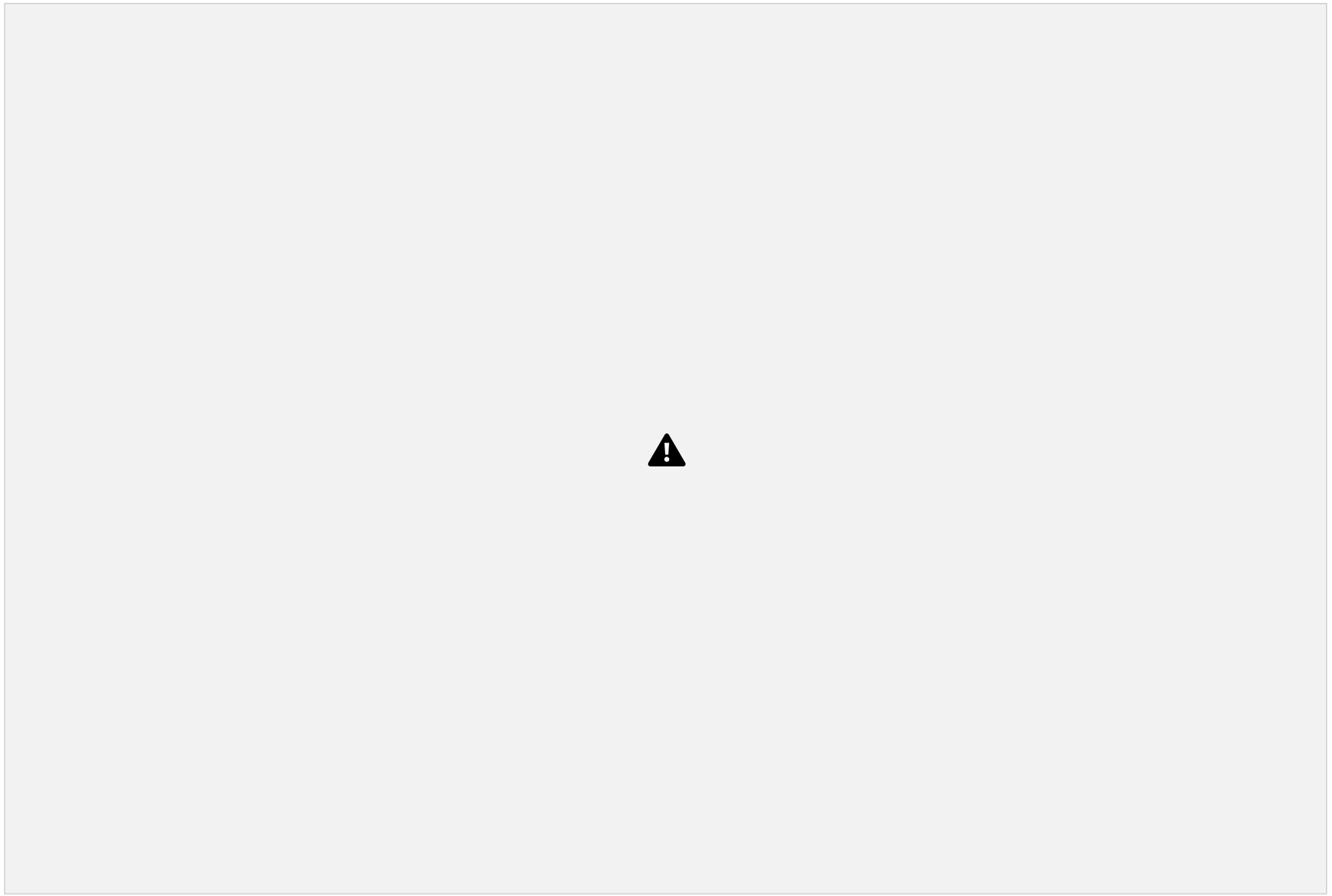


O document.getElementById(idname) permite referenciar dinamicamente qualquer elemento do documento HTML através de um ID.

Lembre-se: o ID é um atributo definido nos marcadores HTML. Ex:

Parágrafo com ID

Hands-on



Hands-on

- **No exemplo anterior o alert() não mostrou muita informação relevante:**
  – [object HTMLParagraphElement]

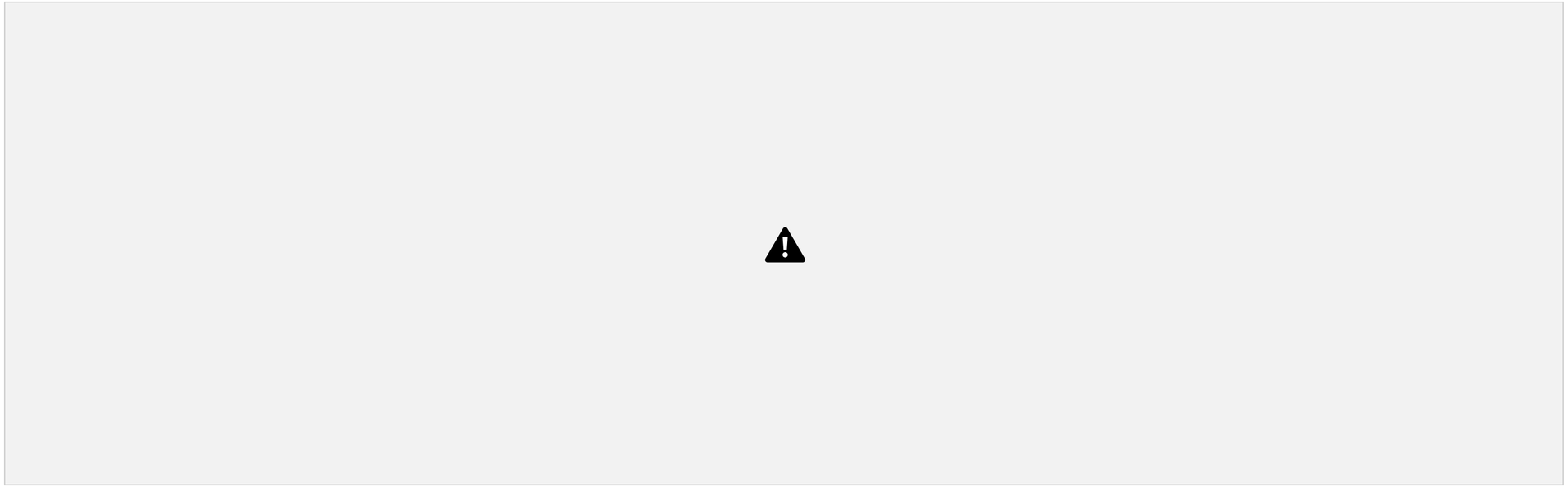
- **Conseguimos entender que é um objeto do tipo parágrafo em HTML. Mas e o que tem dentro disso?**

- **JS oferece um método para saber o que há dentro dos objetos JS**
  – console.log(obj);

- **Onde visualizamos?**
  – Nas ferramentas de desenvolvedor do Chrome ou do Firefox, apertando na tecla F12.

CLASSE E TAG

getElementsByClassName e getElementsByTagName selecionam e retornam uma lista de elementos do DOM. A lista retornada está ao vivo, significa que se elementos forem adicionados, ela será automaticamente atualizada.



getElementsByTagName

- Existem outras formas de recuperar elementos HTML, não só pelo ID. Por exemplo, pelo

nome do marcador (tag):

```
const parag = document.getElementsByTagName("p");
```

•Repare que o argumento não é o valor do atributo name, mas sim o nome do marcador (p, div, span, h1, table).

•Portanto, forma este método retorna TODOS os elementos com o nome do argumento.

•Cada elemento é acessado na forma de um array:

parag[0]

parag[1]

parag.length

getElementsByClassName

Outra forma de recuperar elementos HTML é pelo nome da classe:

```
const x = document.getElementsByClassName("abc");
```

Este método retorna TODOS os elementos com o nome do argumento. No exemplo, retorna todos os elementos que têm o atributo class="abc".

Cada elemento é acessado na forma de um array:

x[0]
x[1]
x.length

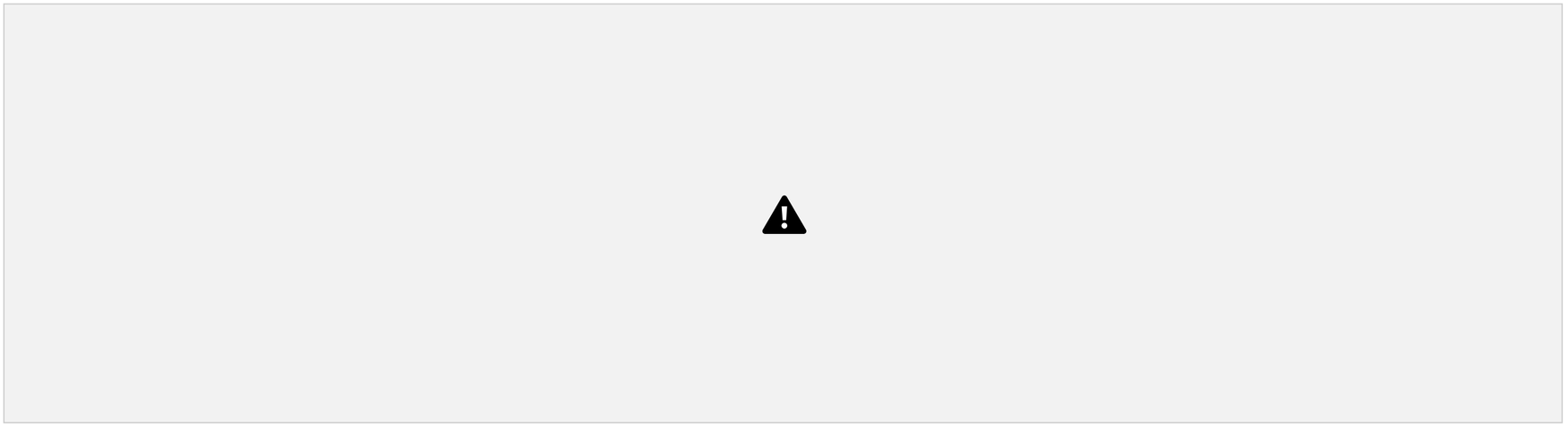
CLASSE E TAG

NodeList: Usado para coleções de elementos do DOM, geralmente obtidos por métodos como querySelectorAll. Não possui métodos avançados de manipulação de dados como os arrays.

Array: Uma estrutura de dados mais geral que pode conter qualquer tipo de valor. Possui uma variedade de métodos poderosos para manipulação e transformação de dados.

SELETOR GERAL LISTA

QuerySelectorAll retorna todos os elementos compatíveis com o seletor CSS em uma NodeList.



SELETOR GERAL LISTA

```
const a = document.querySelectorAll("p.especial");
```

Este método retorna TODOS os elementos que satisfaçam ao seletor escolhido e retorna um array com todos eles.

Se quiser apenas o primeiro elemento ao invés de um array (especialmente se souber que só haverá um elemento onde o seletor fará sentido) use **querySelector** ao invés de querySelectorAll:

```
const b = document.querySelector("div.titulo");
```

SELETOR GERAL LISTA

Além de localizar os elementos da página (do DOM), podemos também alterá-los. A manipulação de objetos HTML do JavaScript pode:

Escrever no documento HTML (document.write):

```
document.write("<p>Parágrafo</p>");
```

Mudar conteúdo HTML de um elemento (.innerHTML):

```
document.getElementById("id").innerHTML = "texto";
```

Mudar o valor de um atributo (.nomeAtributo):

```
document.getElementById("id").src = "imagem.jpg";
```

SELETOR GERAL LISTA

A manipulação do HTML também pode ser feito para os elementos que foram recuperados em um vetor:

SELETOR GERAL LISTA

A manipulação do HTML também pode modificar o CSS usando a

sintaxe “elemento.style.propriedade”. Exemplos:

Eventos

Eventos

Eventos são ações que podem ser detectadas pelo JS.

Todo elemento HTML possui eventos que podem ser disparados por funções JavaScript.

Exemplos de eventos aplicados nos elementos:

–Clique com o mouse – **onclick**

–Passar o mouse em cima – **onmouseover**

–Tirar o mouse de cima – **onmouseout**

–E ainda muitos outros para combinar com formulários, como **onfocus**, **onblur**, **onchange**, **onsubmit** e etc.