

`console.log()` - Mostra no console

`alert()` - Mostra uma janela de alerta na pagina

`confirm()` - Mostra uma janela de confirmação na pagina

`prompt()` - Mostra uma janela de pergunta na pagina

`document.write()` - Escreve no documento

`var` - variavel

`let` - variavel atualizada

`const` - constante (valor fixo)

`typeof()` - Mostra o tipo primitivo

`parseInt()` - Converte para numero inteiro

`parseFloat()` - Converte para numero flutuante

`Number()` - Converte para sua categoria numerica

`toString()` - Converte para string

`String()` - Converte para string

`.length` - Mostra a quantidade de elementos/caracteres

`.toUpperCase()` - Transforma em maiusculas

`.toLowerCase()` - Transforma em minusculas

`.toFixed()` - Define a quantidade de casas decimais

`.replace(' ', ' ')` - Troca uma coisa por outra

`.innerText = ' '` - Adiciona um texto

`.innerHTML = ' '` - Adiciona um texto que permite formatação em html

`.text = ' '` - Adiciona conteudo em uma opção (tag <option>)

`${ }` - troca pelo valor do elemento que esta dentro

`.getDay()` - Pega o dia da semana

`.getDate()` - Pega o dia do mês

`.getMonth()` - Pega o mês

`.getFullYear()` - Pega o ano

`.toLocaleString(idioma, { })` - Configura como o numero será mostrado

`.style.cssText = ``` - Possibilita a estilização em formato css

`.style.color = ''` - Muda a cor de algum elemento

`.style.background = ''` - Muda a cor de fundo de algum elemento

`on` - Prefixo de ligação de ação

`onload = ''` - Faz a função carregar junto do site

`click` - Clicar

`mouseover` - Cursor entrar

`mouseout` - Cursor sair

`mousemove` - Cursor se mover

`mousedown` - Clicar e segurar

`mouseup` - Soltar o clique

`.addEventListener(' ',)` - Presta atenção em um determinado evento

`document.getElementsByTagName(' ')` - Seleciona todas as tags de algum tipo

`document.getElementById(' ')` - Seleciona o elemento pelo Id

`document.getElementsByName(' ')` - Seleciona todos os elementos que tenham o mesmo nome

`document.getElementsByClassName(' ')` - Seleciona todos os elementos que tenham a mesma classe

`document.querySelectorAll(' ')` - Seleciona o elemento pelo seletor

`document.querySelector(' ')` - Seleciona todos os elementos que tenham o mesmo seletor

`document.createElement(' ')` - Cria uma tag

`.setAttribute(' ', ' ')` - Cria ou substitui um atributo e define seu valor

`.appendChild()` - Adiciona um elemento como filho

`.insertAdjacentElement(' ')` - Configura e adiciona em que posição algum elemento será adicionado

`.checked` - Checado

`if` - Se

`else` - Se não

`else if` - Se não, se

`switch()` - Testar

`case` - Caso resulte

`default` - Padrão

`?` - Pergunta

`:` - Se true : se false

`!` - Não

`&&` - E

`||` - Ou

`>` - Maior

`<` - Menor

`=` - Recebe

`==` - Igual

`!=` - Diferente

`while()` - Enquanto

`do... while()` - Faça... Enquanto

`for()` - Repete se uma condição é verdadeira

`in` - percorre um objeto automaticamente

`of` - Percorre um array automaticamente

`break` - Encerra

`[]` - Array

`{ }` - Objeto

`...` - spread / rest

`.indexOf()` - Busca algum elemento dentro de um array e retorna sua posição, se o elemento não existir retorna -1

`.push()` - Adiciona dentro de um array na ultima posição

`.sort()` - Coloca os elementos em ordem crescente

`.map()` - Percorre um array transformando seus elementos e no final gera outro array com os elementos ja transformados

`.filter()` - Percorre um array filtrando seus elementos e no final gera outro array com os elementos que passaram no filtro

`.reduce()` - Percorre um array e no final gera um unico valor

`delete` - Deleta propriedades de objetos

`Object.assign()` - Junta dois ou mais objetos

`{ }` - Extrai dados de objetos, pegando o nome da propriedade e transformando em variavel

`get` - É usado quando um metodo retorna um valor

`set` - É usado quando um metodo altera / define um valor

`Object.keys()` - Retorna um array com todas as propriedades de um objeto

`Object.values()` - Retorna um array com todos os valores de um objeto

`this.` - Faz referencia ao elemento em si

`setInterval()` - Executa alguma função em intervalos de milisegundos

`setTimeout()` - Executa alguma função assim que o tempo em milisegundos acaba

`function` - Cria uma função

`=>` - Cria uma arrow function

`init` - Isola uma função

`return` - Retorna algum valor de uma função

`import` - Importa o conteudo

`export` - Exporta o conteudo

`export default` - Exporta o conteudo como principal

`from " "` - Especifica o caminho do arquivo em que o conteudo esta

`as` - altera o nome colocando um apelido

`*` - Seleciona todos ↑

`//` - Regex literal

`new RegExp()` - Regex não literal

`.test()` - Testa uma string usando regex e retorna true ou false

`.exec()` - Testa uma string usando regex e retorna informações do que foi identificado

`.match()` - Testa um regex em uma string e retorna informações do que foi identificado ou uma lista do que foi encontrado caso tenha a flag 'g'

`.matchAll()` - Retorna uma lista com todos os padrões encontrados

`.search()` - Testa um regex em uma string e retorna a posição da primeira ocorrência ou -1 se nada for encontrado

`.replaceAll()` - Retorna uma nova string com as ocorrências de um padrão substituídas por outro padrão

`new` - Cria uma instância de um tipo de objeto definido pelo usuário

`.dotall` - Verifica se a flag 's' tá ativa e retorna true ou false

`class` - Define uma classe não literal

`static` - Define que o método é estático

`extends` - Cria uma classe filha de outra classe

`#` - Torna uma propriedade ou método privado

`prototype` - Cria ou substitui propriedades e métodos após a declaração da classe

`.call()` - Acessa ou altera o valor de um atributo através de uma função

`.apply()` - Acessa ou altera o valor de um atributo através de uma função e os valores precisam estar em um array

`.bind()` - Cria uma nova função a partir de um objeto e de outra função

`.hasOwnProperty()` - Identifica uma propriedade através de um objeto

`.propertyIsEnumerable()` - Verifica se o valor de uma propriedade é iterável e se é do próprio objeto retornando true ou false

`Object.defineProperty()` - Define e/ou configura uma propriedade em um objeto

`Object.defineProperties()` - Define e/ou configura várias propriedades em um objeto

`.getOwnPropertyDescriptor()` - Acessa as configurações de uma propriedade

`Object.preventExtensions()` - Bloqueia adições ou remoções externas no objeto

`Object.isExtensible()` - Verifica se o objeto é extensível e retorna true ou false

`Object.seal()` - Sela um objeto

`Object.isSealed()` - Verifica se o objeto está selado e retorna true ou false

`Object.freeze()` - Congela um objeto

`Object.isFrozen()` - Verifica se o objeto está congelado e retorna true ou false

`Object.getPrototypeOf()` - Retorna a função construtora da qual o objeto foi criado

`Object.create()` - Cria um objeto a partir de outro

`instanceof` - Verifica se o objeto foi criado a partir da função construtora e retorna true ou false

`\n` - Quebra a linha

`try { }` - Testa o código ↑

`catch() { }` - Captura o erro do try

`finally { }` - Final da ação try : catch

`throw new` - Gera um novo erro

`document.forms` - Captura todos os forms da página

`document.forms.item()` - Captura um formulário pela posição

`document.forms.namedItem(" ")` - Captura um formulário pelo name=" "

`document.forms[].elements[]` - Captura um input de um formulário e o formulário pela posição

`new Date()` - Cria um novo objeto do tipo Date

`.getDate()` - Retorna o dia do mês atual (.set para alterar)

`.getMonth()` - Retorna o mês atual (.set para alterar)

`.getFullYear()` - Retorna o ano atual (.set para alterar)

`.getDay()` - Retorna o dia da semana atual (`.set` para alterar)

`.getHours()` - Retorna a hora atual (`.set` para alterar)

`.getMinutes()` - Retorna o minuto atual (`.set` para alterar)

`.getSeconds()` - Retorna o segundo atual (`.set` para alterar)

`.toDatestring()` - Retorna o dia do mês atual por estenço

`.toLocaleDateString()` - Retorna a data atual no formato padrão

`.toLocaleString()` - Retorna as horas e a data atual no formato padrão

`Math.E` - Retorna o numero Euler

`Math.PI` - Retorna o numero PI

`Math.SQRT2` - Retorna a raiz quadrada de 2

`Math.SQRT1_2` - Retorna a raiz quadrada de $\frac{1}{2}$

`Math.LN2` - Retorna o logaritmo natural de 2

`Math.LN10` - Retorna o logaritmo natural de 10

`Math.LOG2E` - Retorna a base 2 do logaritmo de E

`Math.LOG10E` - Retorna a base decimal do logaritmo de E

`Math.abs(x)` - Retorna o valor absoluto de X

`Math.acos(x)` - Retorna o arco cosseno de X

`Math.asin(x)` - Retorna o arco seno de X

`Math.atan(x)` - Retorna o arco tangente de X como um valor numérico entre $\pi/2$ e $\pi/2$ radiano

`Math.atan2(y,x)` - Retorna o arco tangente do quociente dos argumentos Y e X

`Math.ceil(x)` - Retorna o valor de X arredondado para cima

`Math.cos(x)` - Retorna o cosseno de X em radianos

`Math.exp(x)` - Retorna o valor de Math.E elevado a X

`Math.floor(x)` - Retorna o valor de X arredondado para baixo

`Math.log(x)` - Retorna o logaritmo natural (base Math.E) de X

`Math.max(x, y, z, ...)` - Retorna o maior valor dos argumentos

`Math.min(x, y, z, ...)` - Retorna o menor valor dos argumentos

`Math.pow(x, y)` - Retorna o valor de X elevado a Y

`Math.random()` - Retorna um número aleatório entre 0 e 1

`Math.random()*num` - Retorna um numero aleatorio entre 0 e o numero especificado

`Math.floor(Math.random()*num)` - Retorna um numero arredondado aleatorio entre 0 e o numero especificado

`Math.round(x)` - Retorna o inteiro mais próximo, arredonda para cima ou para baixo

`Math.sin(x)` - Retorna o seno de X em radianos

`Math.sqrt(x)` - Retorna a raiz quadrada de X

`Math.tan(x)` - Retorna a tangente do ângulo X

`keydown` - Evento quando uma tecla é pressionada

`.key` - Retorna a tecla pressionada

`requestAnimationFrame()` - Executa uma animação com frames automaticos, ou seja, executa uma função em loop com tempo definido automaticamente

`cancelAnimationFrame` - Interrompe uma animação

`.hasAttribute(" ")` - Verifica se um elemento possui um atributo especifico

`.removeAttribute(" ")` - Remove um atributo especifico de um elemento

`.charAt()` - Retorna o caractere que esta na posição especificada

`.charCodeAt()` - Retorna o codigo do caractere que esta na posição especificada

`fromCharCode()` - Retorna o caractere apartir do codigo especificado

`.concat()` - Concatena strings ou arrays

`.lastIndexOf(" ")` - Retorna a posição do ultimo caractere encontrado ou -1

`.localeCompare()` - Compara duas strings e verifica se são iguais, retornando 0 ou -1

`.slice()` - Corta e captura apartir de uma posição inicial e final

`.split(" ")` - Divide uma string apartir de um caractere especificado e retorna as divisões em um array

`.substring()` - Semelhante ao slice vai cortar e capturar uma string da posição inicial até a posição final

`.trim()` - Remove espaços antes e depois de uma string

`.startsWith(" ")` - Verifica se uma string começa com os caracteres especificados e retorna true ou false

`.endsWith(" ")` - Verifica se uma string termina com os caracteres especificados e retorna true ou false

`.includes(" ")` - Determina se um array contém determinado item, ou se uma string possui os caracteres especificados

`.repeat()` - Concatena uma mesma string determinado numero de vezes

`.search(" ")` - Busca uma ocorrencia e retorna a posição do primeiro caractere encontrado

`.valueOf()` - Retornar o valor primitivo do objeto especificado ou o valor original

`.fromCodePoint()` - Retorna / transforma o codigo ascii em caractere

`window.open()` - Abre e configura a abertura de uma nova janela a partir de uma url

`.close()` - Fecha uma janela aberta a partir de uma variavel

`window.innerWidth` - Retorna a largura interna da janela

`window.innerHeight` - Retorna a altura interna da janela

`window.outerWidth` - Retorna a largura total da janela

`window.outerHeight` - Retorna a altura total da janela

`window.screenX` - Retorna o posicionamento X (horizontal) da janela

`window.screenY` - Retorna o posicionamento Y (vertical) da janela

`window.scrollBy(x, y)` - Posiciona a pagina em uma determinada posição X e Y

`window.length` - Informa a quantidade de iframes que tem dentro da pagina

`window.frames` - Retorna um array com todos os iframes da pagina

`navigator.cookieEnabled` - Verifica se tem cookies habilitados e retorna true ou false

`navigator.geolocation` - Verifica se a geolocalização esta habilitada e retorna o objeto ou undefined

`navigator.geolocation.getCurrentPosition()` - Retorna a localização geografica do browser

`.coords` - Retorna a latitude e a longitude

`.coords.latitude` - Captura a latitude

`.coords.longitude` - Captura a longitude

`navigator.language` - Retorna o idioma do browser

`navigator.onLine` - Verifica se o browser esta online e retorna true ou false

`navigator.userAgent` - Retorna qual agente foi enviado para o browser pelo servidor

`screen.width` - Retorna a largura disponivel da tela

`screen.height` - Retorna a altura disponivel da tela

`screen.availWidth` - Retorna a largura da tela ao todo

`screen.availHeight` - Retorna a altura da tela ao todo

`screen.colorDepth` - Retorna a profundidade em bits da paleta de cor pra exibir imagens

`screen.pixelDepth` - Retorna a resolução da cor em bits por cada pixel da tela

`history.length` - Retorna o tamanho do historico

`history.back()` - Volta pra pagina anterior

`history.forward()` - Avança pra próxima pagina

`history.go()` - Permite avançar ou retroceder determinado numero de elementos do historico

`location` - Retorna informações sobre a URL

`location = " "` - Redireciona para outra URL (site/link)

`location.hash` - Retorna a hashtag e tudo o que estiver depois dela em uma URL

`location.href` - Retorna a URL

`location.href = " "` - Redireciona para outra URL (site/link)

`location.origin` - Retorna o protocolo, hostname e o numero da porta

`location.pathname` - Retorna o caminho, tudo o que estiver depois do protocolo e do hostname

`location.port` - Retorna o numero da porta

`location.protocol` - Retorna o protocolo da URL

`location.search` - Retorna a interrogação e tudo o que estiver depois dela em uma URL

`location.assign(" ")` - Carrega uma nova pagina ou site

`location.replace(" ")` - Carregar uma nova pagina ou site e remove a anterior do historico

`location.reload()` - Recarrega a pagina

`localStorage.setItem(" ", " ")` - Define ou altera um conjunto chave : valor (vale para o sessionStorage)

`localStorage.getItem(" ")` - Retorna o valor de uma chave (vale para o sessionStorage)

`localStorage.removeItem(" ")` - Remove um conjunto chave : valor pela chave (vale para o sessionStorage)

`localStorage.clear()` - Limpa todo o Local Storage (vale para o sessionStorage)

`localStorage.key()` - Pesquisa e retorna uma chave ou retorna null (vale para o sessionStorage)

`document.createAttribute(" ")` - Cria um novo atributo

`.value = " "` - Define um valor para o novo atributo

`.setAttributeNode()` - Adiciona ou substitui o novo atributo no elemento especificado

`.getAttribute(" ")` - Retorna o valor de um determinado atributo em um determinado elemento

`.insertBefore()` - Adiciona um novo elemento antes de algo

`.childNodes.length` - Conta os elementos filhos de determinado elemento

`.childNodes` - Retorna uma lista com os filhos de um elemento

`.replaceChild()` - Substitui um elemento filho por outro elemento

`.createTextNode(" ")` - Cria um texto que é armazenado em uma variavel, essa variavel pode ser inserida como filho de um elemento

`.parentNode` - Retorna o elemento pai

`.parentNode.nodeName` - Retorna o nome do elemento pai

`.remove()` - Remove um elemento

`.removeChild()` - Remove um determinado elemento filho

`.hasFocus()` - Retorna true ou false se a pagina ou algum elemento da pagina estiver em foco

`document.activeElement` - Retorna o elemento selecionado ou ativo dentro da pagina

`document.activeElement.tagName` - Retorna o nome da tag do elemento selecionado ou ativo dentro da pagina

`document.activeElement.id` - Retorna o id do elemento selecionado ou ativo dentro da pagina

`document.activeElement.value` - Retorna o value do elemento selecionado ou ativo dentro da pagina

`.dataset.` - Altera ou retorna o dataset de determinado elemento

`.target` - Retorna o elemento que foi clicado

`.scrollIntoView()` - Rola a pagina até um determinado elemento

`.hidden = " "` - Oculta ou não um elemento a partir de um valor booleano

`function*` - Cria uma função geradora

`.next()` - Vai pra próxima iteração

`yield` - Permite a iteração do item

`new Map()` - Cria um objeto do tipo map (coleção de chave : valor)

`.set(" ", " ")` - Altera ou define um conjunto chave : valor em um objeto map

`.get(" ")` - Retorna o valor de uma chave de um objeto map

`.size` - Retorna o tamanho do map

`.keys()` - Retorna um novo objeto Array que contem o indice de cada item no array

`.has(" ")` - Verifica se uma chave existe no objeto map retornando true ou false

`.clear()` - Limpa todo o map

`.delete(" ")` - Deleta um determinado conjunto a partir de uma chave no objeto map

`new Set()` - Cria ou converte em um objeto do tipo Set (coleção de valores)

`.add()` - Adiciona valores em um objeto do tipo Set

`JSON.parse()` - Converte o json em um array de objetos

`JSON.stringify()` - Converte um array de objetos ou objeto em um json

`"use strict"` - Ativa o modo estrito

`.stopPropagation()` - Para a propagação de um evento em elementos filhos

`clearTimeout()` - Encerra a execução do setTimeout

`clearInterval()` - Encerra a execução do setInterval

`.children` - Retorna todos os elementos filho

`.firstChild` - Retorna o primeiro filho (sendo qualquer um)

`.firstElementChild` - Retorna o primeiro elemento filho

`.lastElementChild` - Retorna o ultimo elemento filho

`.lastChild` - Retorna o ultimo filho (sendo qualquer um)

`document.getRootNode()` - Retorna o nó raiz do documento

`.ownerDocument` - Retorna o proprietario do documento

`.hasChildNodes` - Verifica se o elemento possui filhos e retorna true ou false

`.parentNode` - Retorna o pai

`.parentElement` - Retorna o elemento pai

`.nextSibling` - Retorna o próximo irmão ou null se for o ultimo

`.previousSibling` - Retorna o irmão anterior ou null se for o primeiro

`.classList` - Retorna uma lista contendo o valor do atributo class de um elemento

`.classList.add(" ")` - Adiciona um novo valor na lista, ou seja, adiciona uma nova classe em um elemento

`.classList.contains(" ")` - Verifica se a lista tem uma classe específica e retorna true ou false

`.classList.item()` - Retorna a classe que está em determinada posição na lista

`.classList.length` - Retorna o tamanho da lista

`.classList.remove(" ")` - Remove uma ou mais classes com determinado nome

`.classList.toggle(" ")` - Vai adicionar uma classe caso não exista ou remover caso exista

`.classList.value` - Retorna os valores de class de um elemento como uma única string

`.className` - Retorna o valor do atributo class de um determinado elemento

`.className = " "` - Define o valor do atributo class de um determinado elemento

`.forEach(() => { })` - Percorre um array enquanto executa uma função que recebe os itens do array como parametro

`.find(() => { })` - Percorre um array enquanto executa uma função, se essa função retornar true ela vai parar e resultar no item atual, se retornar false vai ter como resultado undefined

`.every(() => { })` - Verifica se todos os itens do array passam pelo teste implementado pela função, se todos passarem retornará true, se não false

`.some(() => { })` - É uma variação do every, vai testar se ao menos um dos itens do array passa no teste implementado pela função, retornando true ou false

`navigator.clipboard.read()` - Vai colar da área de transferência

`navigator.clipboard.readText()` - Vai colar um texto da área de transferência

`navigator.clipboard.write()` - Vai copiar algo e mandar pra área de transferência

`navigator.clipboard.writeText()` - Vai copiar um texto e mandar pra area de transferencia

`window.print()` - Imprime a tela da pagina

`Symbol()` - Retorna um objeto do tipo symbol que possui um identificador unico

`Symbol(" ")` - Define uma descrição para o symbol

`Symbol.for(" ")` - Cria um identificador no registro global para o Symbol ou retorna se já existir

`Symbol.keyFor()` - Retorna o identificador do Symbol que executou o for, se não executou retorna undefined

`simbolo.description` - Retorna a descrição do simbolo

`.createCaption()` - Cria um elemento do tipo caption

`.deleteCaption()` - Deleta um elemento do tipo caption

`.createTHead()` - Cria um elemento do tipo thead

`.deleteTHead()` - Deleta um elemento do tipo thead

`.createTBody()` - Cria um elemento do tipo tbody

`.tBodies` - Retorna uma coleção com todos os elementos do tipo tbody

`.createTFoot()` - Cria um elemento do tipo tfoot

`.deleteTFoot()` - Deleta um elemento do tipo tfoot

`.rows` - Retorna uma coleção com todos os elementos tr

`.insertRow()` - Adiciona um novo elemento tr em uma posição especifica

`.deleteRow()` - Deleta um elemento tr em uma posição especifica

`.cells` - Retorna uma coleção com todos os elementos td e th

`.insertCell()` - Adiciona um novo elemento td em uma posição especifica

`.deleteCell()` - Deleta um elemento td em uma posição especifica

`fetch()` - Busca recursos da rede (dados de sites ou APIs) e retorna uma promessa que é cumprida assim que a resposta estiver disponível, ou seja, busca dados json e retorna uma promessa desses dados

`.json()` - Pega uma promessa json (fluxo response) e lê até a conclusão, ou seja, recebe uma promessa com dados json e retorna esses dados como um array de objetos

`Array.from()` - Converte, cria ou transforma qualquer objeto iterável em um array

`.padStart()` - Preenche o início de uma string com outra até determinado comprimento

`.padEnd()` - Preenche o fim de uma string com outra até determinado comprimento

`.trimStart()` - Remove os espaços iniciais de uma string

`.trimEnd()` - Remove os espaços finais de uma string

`.codePointAt()` - Retorna o valor do ponto de código unicode na posição especificada

`.at()` - Retorna o item do array na posição especificada

`.entries()` - Retorna um array chave / valor onde a chave é o índice e o valor são os itens do array alvo

`.fill()` - Altera os itens de um array da posição inicial até a posição final com um valor estático

`.findIndex(() => { })` - Retorna a posição do primeiro item que cumpre o teste passado pela função ou -1

`.findLast(() => { })` - Retorna o primeiro item na ordem inversa que cumpre o teste passado pela função ou undefined

`.findLastIndex(() => { })` - Retorna a posição do primeiro item na ordem inversa que cumpre o teste passado pela função ou -1

`.flat()` - Retorna um array com os elementos dos sub-arrays concatenados até a profundidade especificada

`.flatMap(() => { })` - Percorre um array e retorna outro com cada um dos valores transformados a partir de uma função e achatados em um nível, é a junção do map com o flat de profundidade 1

`Array.isArray()` - Determina se o valor passado é um array retornando true ou false

`.join()` - Concatena todos os itens de um array a partir de um separador, retornando no final uma string

`Array.of()` - Cria um array com os valores do parametro

`.pop()` - Remove e retorna o ultimo item do array

`.reduceRight(() => { },)` - Funciona da mesma forma que o reduce mas na direção oposta

`.reverse()` - Inverte e retorna o array invertido

`.shift()` - Remove e retorna o primeiro item de um array

`.splice()` - Remove, substitui ou adiciona itens de um array modificando o conteudo original e retorna os itens removidos como um novo array

`.unshift()` - Adiciona um ou mais itens no inicio de um array e retorna o novo tamanho dele

`.copyWithin()` - Copia e cola os itens de um array de uma posição para outra

`.values()` - Imprime todos os elementos do array

`Object.fromEntries()` - Transforma uma lista de pares chave-valor em um objeto

`Object.getOwnPropertyDescriptors()` - Retorna todas as configurações de todas as propriedades de um objeto

`Object.getOwnPropertyNames()` - Retorna um array de todas as propriedades encontradas diretamente em um determinado objeto

`Object.getOwnPropertySymbols()` - Retorna um array de todas as propriedades de simbolo encontradas diretamente em um determinado objeto

`Object.hasOwn()` - Retorna true se o objeto especificado tiver a propriedade indicada como sua própria propriedade ou false se a propriedade for herdada ou não existir

`Object.is()` - Determina se dois valores são o mesmo valor e retorna true ou false

`.prototype.isPrototypeOf()` - Verifica se um objeto existe na cadeia de prototipos de outro objeto

`delete` - Deleta propriedades de objetos

`in` - Deleta propriedades de objetos

`.after()` - Adiciona um elemento depois de outro

`.append()` - Adiciona um elemento ou texto dentro de outro elemento (como filho)

`.attributes` - Retorna uma coleção com todos os atributos de um determinado elemento

`.autocapitalize` - Verifica se o elemento possui o atributo autocapitalize

`.autofocus` - Verifica se o elemento possui o atributo autofocus e retorna true ou false

`.before()` - Adiciona um elemento ou texto antes de outro elemento

`.childElementCount` - Retorna o numero de elementos filho de um determinado elemento

`.clientHeight` - Retorna a altura interna de um elemento em pixels ou 0

`.clientLeft` - Retorna a largura da borda esquerda de um elemento em pixels

`.clientTop` - Retorna a largura da borda superior de um elemento em pixels

`.clientWidth` - Retorna a largura interna de um elemento em pixels ou 0

`.cloneNode()` - Duplica e retorna um determinado elemento

`.contains()` - Retorna um valor booleano indicando se um determinado elemento for descendente (filho, neto, ...) de outro

`.contentEditable` - Verifica se um determinado elemento é ou não editável e retorna true, false ou inherit

`.dir` - Retorna a direção de escrita de um texto

`.dir = " "` - Define a direção de escrita de um texto

`.style` - Permite estilizar um elemento

(NÃO COMPLETO! (PAROU EM "J - ESTILIZAÇÃO"))

Link Importante: https://www.youtube.com/watch?v=ckfVA_ikFh4&ab_channel=Mango