**JavaScript**

Variável pode ser representado por var ou let

= significa recebe

Null - retira da variável

**Identificadores (variável)**

* Podem começar com letra, $ ou \_
* Não podem começar com números
* É possível usar letras ou números
* É possível usar acentos e símbolos
* Não podem conter espaço
* Não podem ser palavras reservas (ex: var, let)

Tipos de variáveis:

* Number (representa int e float)
  + Infinty
  + NaN
* String (cadeias de caracteres, conjuntos de números)
* Boolean
* Null
* Undefined
* Object
  + Array
* Function

Sinal de + concatena, ou seja junta, ou para adição.

Conversão número para string:

* Number.parseInt(n) – converte string em número inteiro
* Number.parseFloat(n) – converte string em número real
* Se colocar só Number(n) fica no automático.

Conversão string para número:

* String(n)
* N.toString()

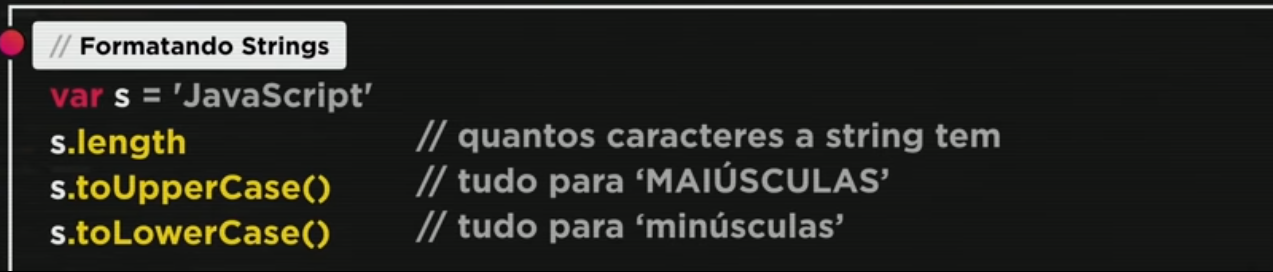
Template Strings:

* `Eu estou aprendendo ${n}`

Place rolder:

* ${n}

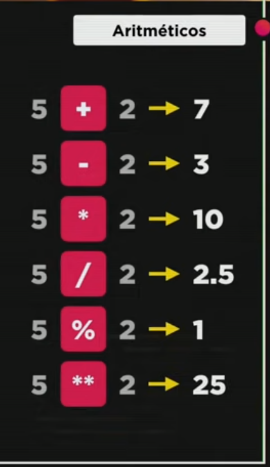
Formatação de strings:



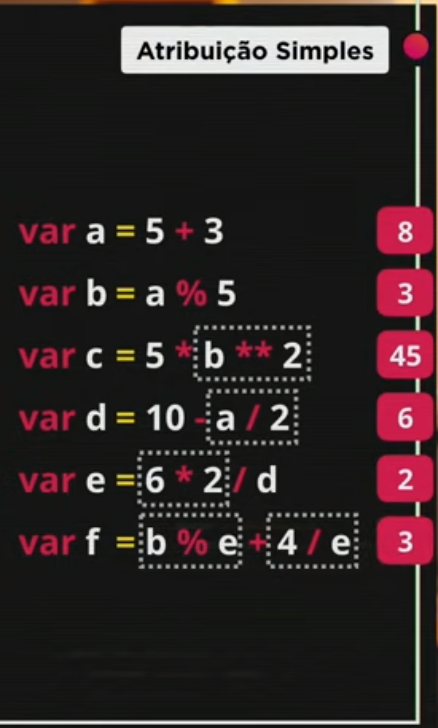
Formatação de números:

* N.toFixed(2) (fixa pra 2 casas decimais após a virgula)
* n.toFixed(2).replace('.', ',') (replace troca um item por outro)
* n1.toLocaleString('pt-BR', {style: 'currency', currency: 'BRL'}) (converte para moeda que quiser)

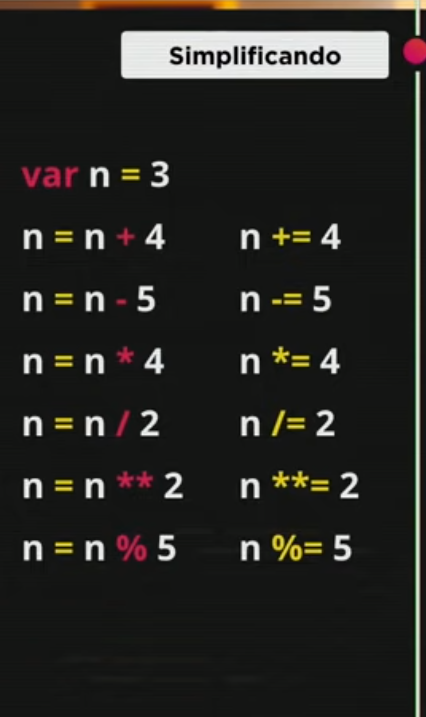
Operadores aritméticos:



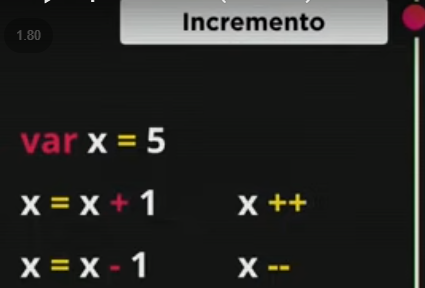
Atribuição simples:



N = n + 4 pode ser simplificado para n += 4

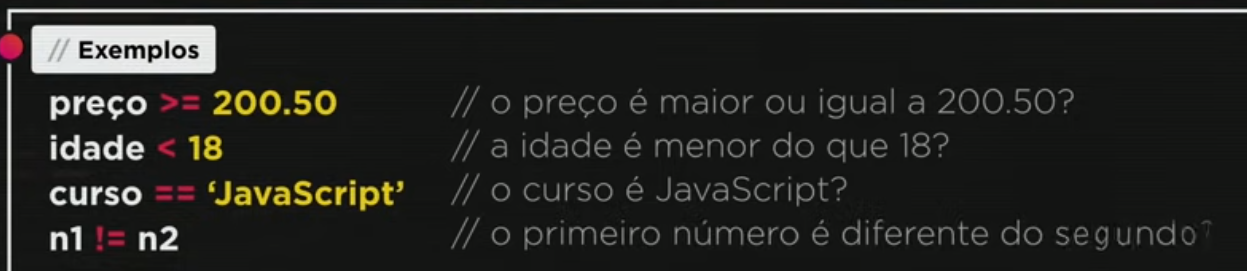


Posso substituir x = x +1 por x ++



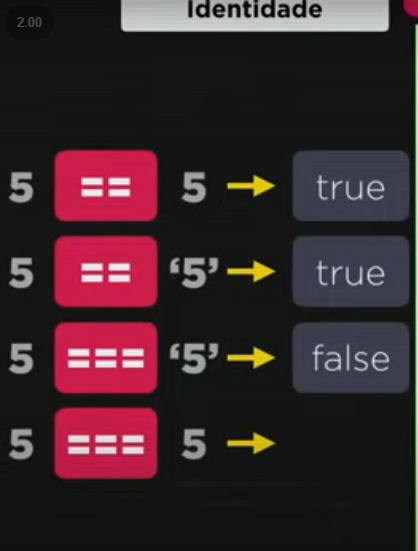
Operadores relacionais: (resultado sempre boolean: true ou false)



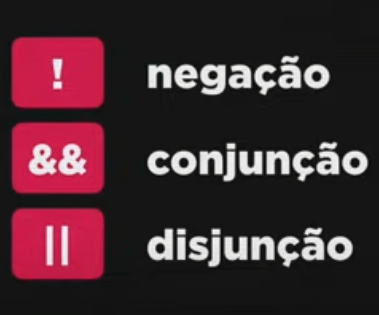


Operadores relacionais de identidade (JavaScript não testa o tipo):

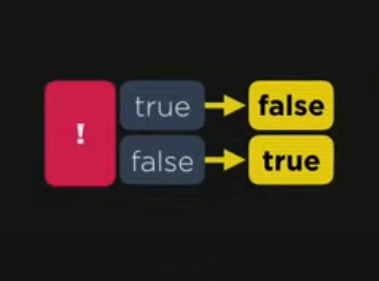
= = = - verifica até o tipo. / != =



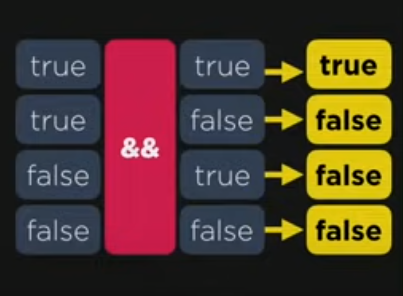
Operadores lógicos:



!:



&&:

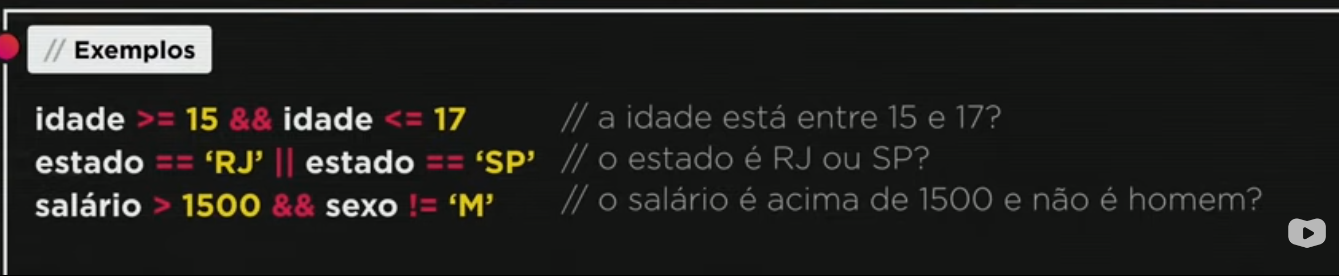


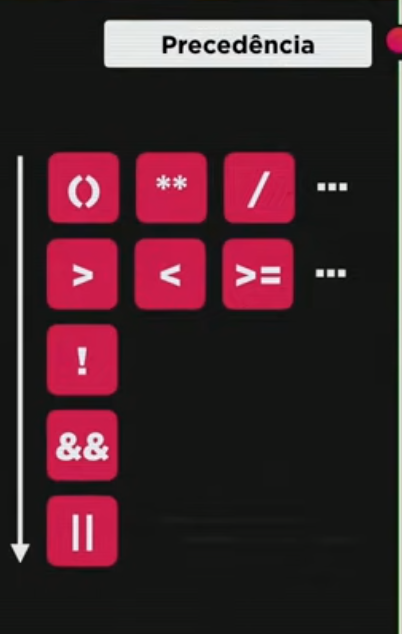
|| -



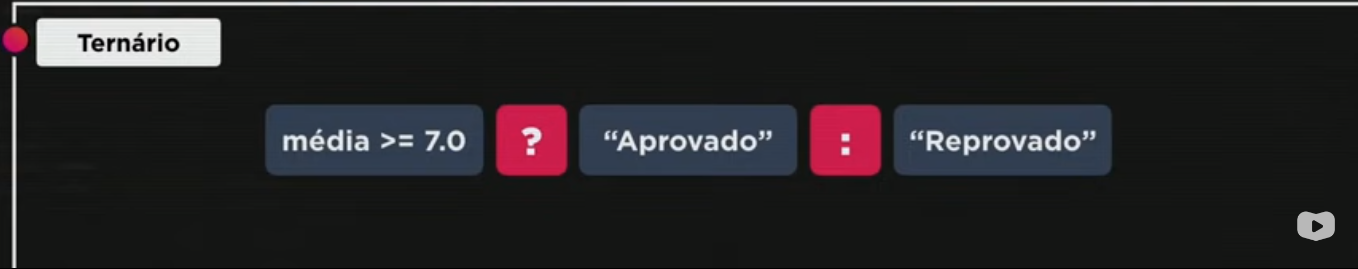
Ordem operadores: 1 aritméticos, 2 relacionais e 3 lógicos

Ordem lógicos: ! - && - ||



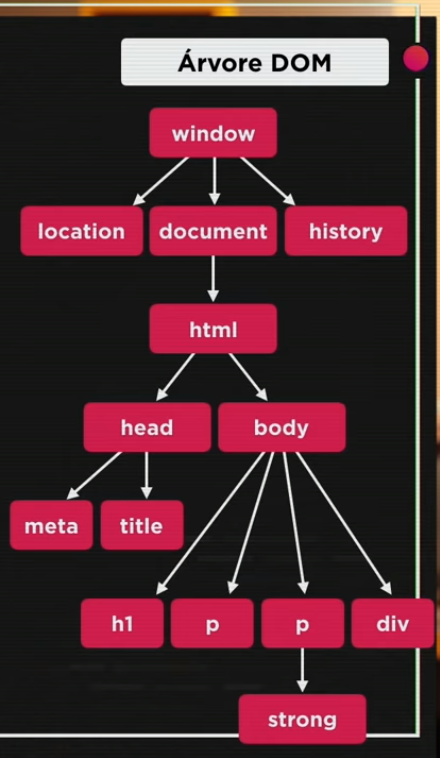


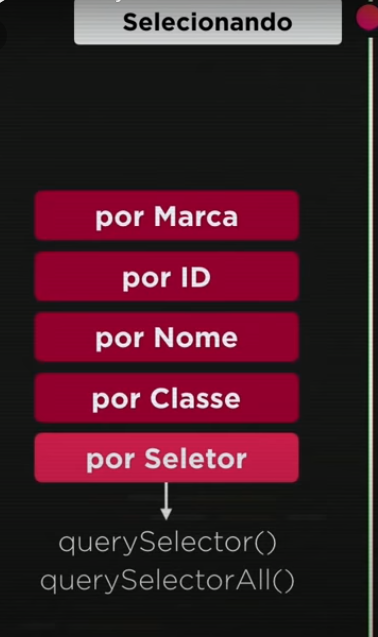
Operador ternário:



DOM – Document Object Model (conjunto de objetos dentro do navegador que dá acesso aos componentes internos do site)

Árvore DOM: tudo no javascript está dentro de window

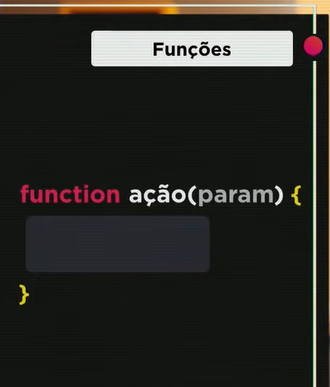


 seletores

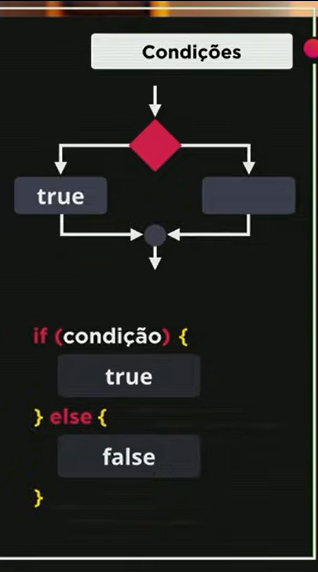
Eventos DOM –

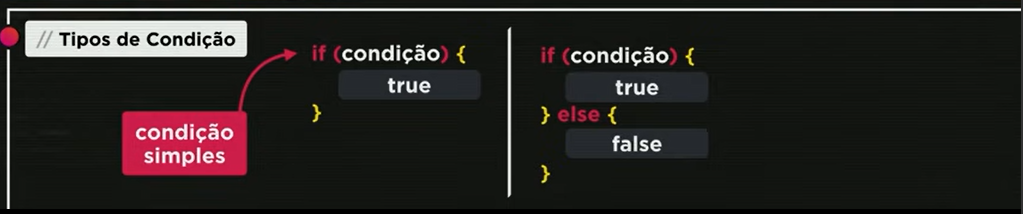


Funções: conjunto de linhas que executam quando o evento ocorre.

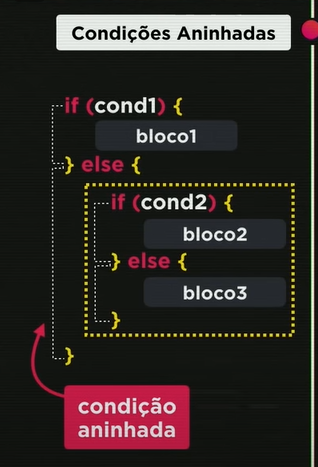


Condições:

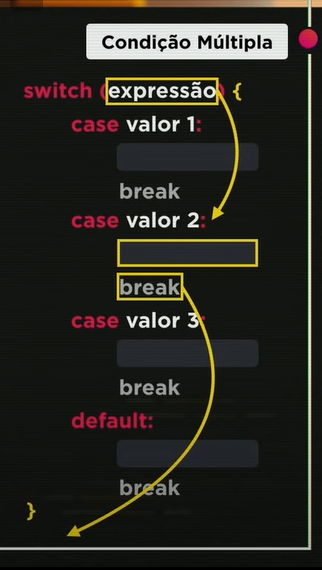
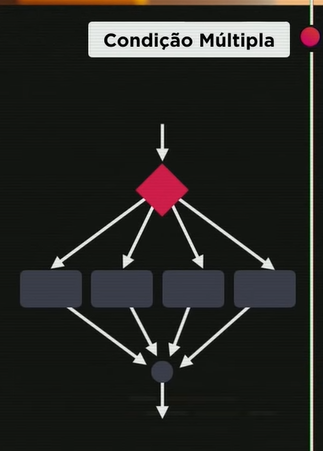




Condições aninhadas;

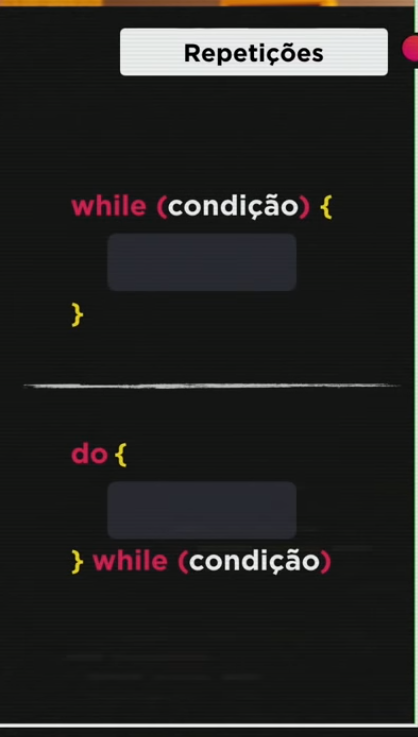


Condições múltiplas:



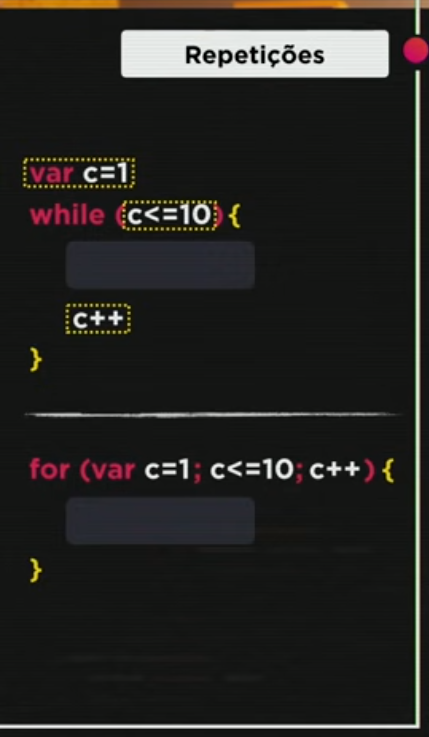
Obrigatório o uso de break, exceto no default.

Repetições:



While = teste lógico no início

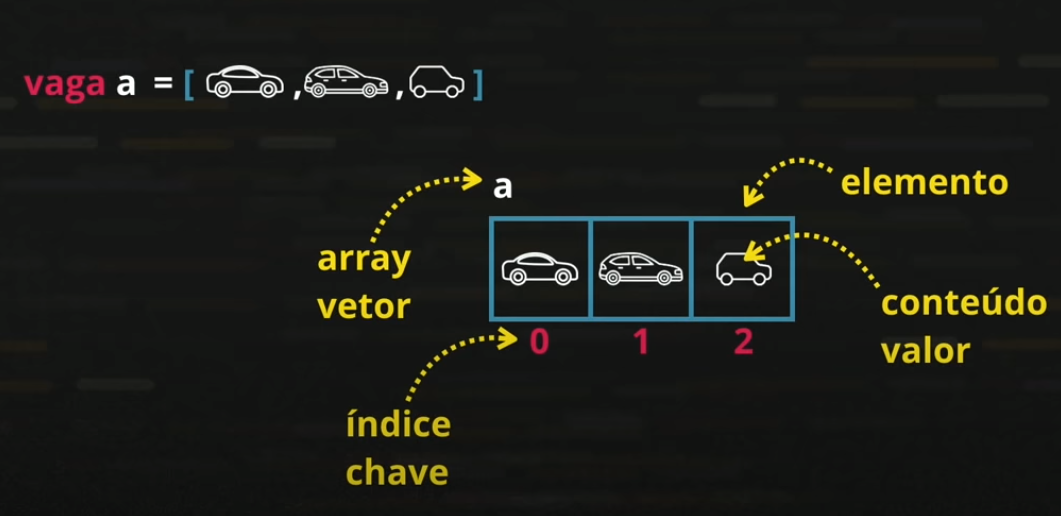
Do while = teste lógico no final.



apprendChild = adicionar elemento filho a tal item

ex: X(onde queroadicionar).appendChild(o que quero adicionar)

Variáveis Compostas (array)



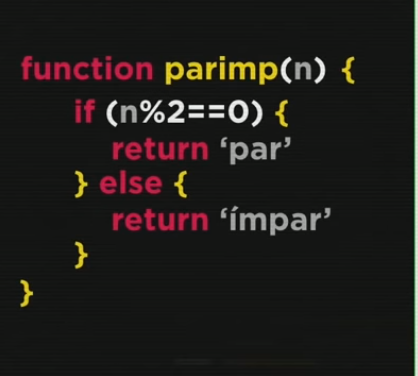
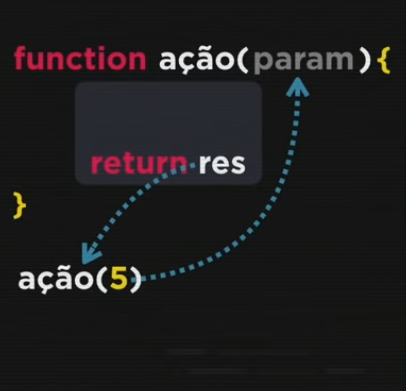
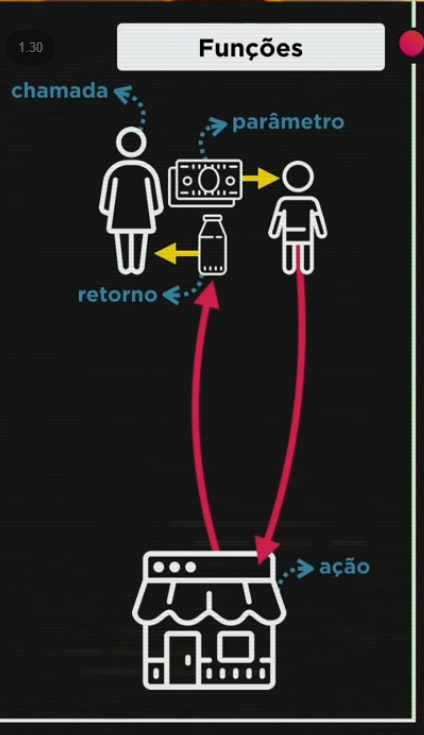
Add no array:

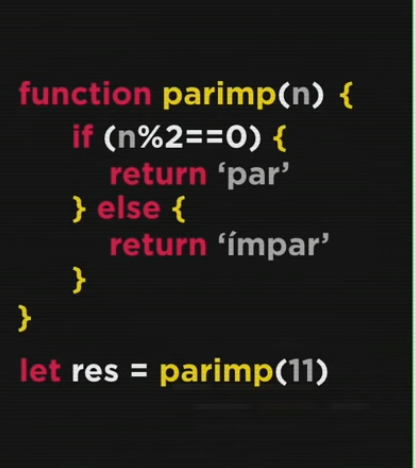
* var[posição no array] = 6(valor que quer inserir)
* var.push(valor que quer inserir)
* var.length - mostra quantos elementos tem no array
* num.sort() – coloca todos elementos em ordem crescente



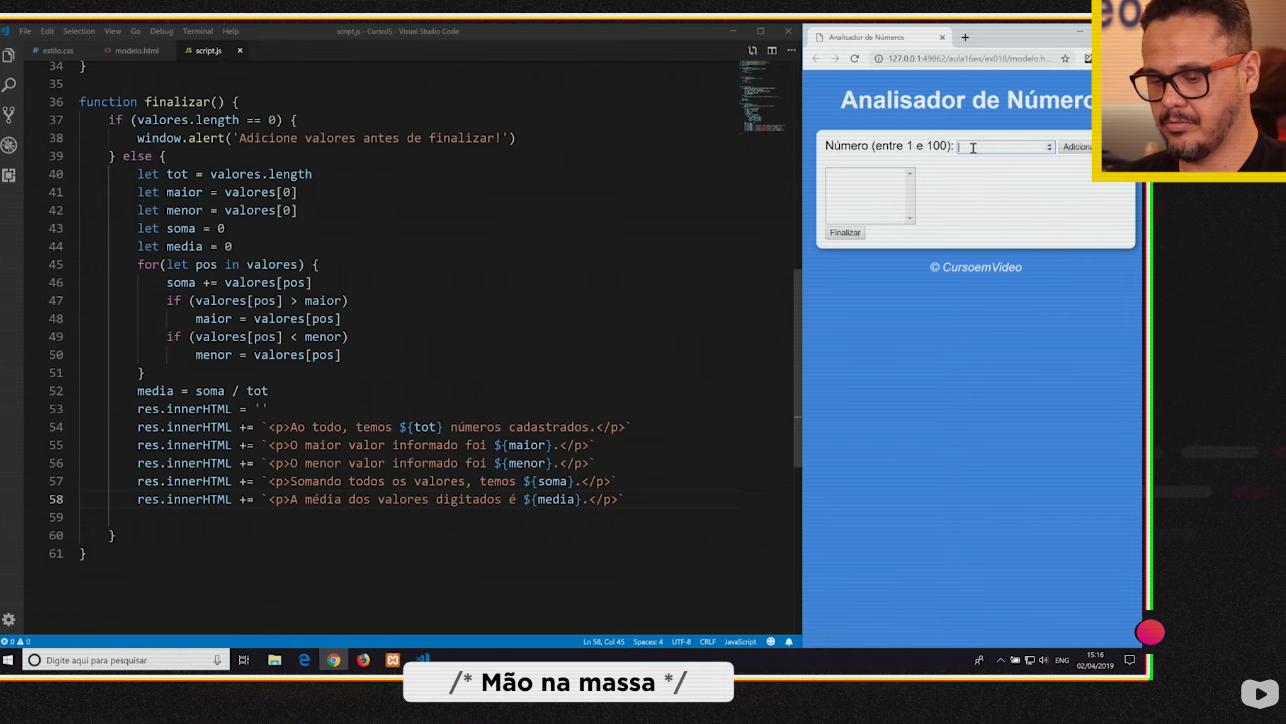
Var.indexOf(7) – procura o valor 7 no vetor e fala posição

Função: são ações executadas assim que são chamadas ou em decorrência de algum evento. Pode receber parâmetros e retornar um resultado.





Outra forma de resolver o ex16:



Estudar funções, objetos, modularização, RegEx, JSON, AJAX, NodeJS.

Arrays em JavaScript é heterogêneo (string, número, valor lógico)

Chaves declara objeto. Objetos podem guardar funções

