

HISTÓRICO E VERSÕES

Do Mocha aos dias atuais

01

O CENÁRIO ATUAL

Problemas resolvidos e novos problemas

PRINCIPAIS CARACTERÍSTICA

Paradigmas, tipagem, virtudes fraqueza

03

CONTEÚD

04



var, let, cosnt e hoisting

05

ESCOPO

Global e loca

06

ESTRUTURAS DE CONTROLE

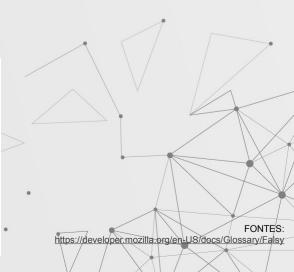
Diferentes estruturas condicionais e de repetição



6. MAIS DO MESMO, MAS...

- if... else... else if...
- switch... case...
- ?
- O (condicao)? <se true>: <se false>;
- O let teste = "" ? "verdadeiro" : "falso"; console.log(teste); // falso.
- mas ... (o que é falso?) falsy

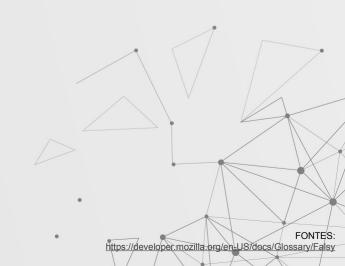
| false | The keyword false |
|-----------|---|
| 0 | The number zero |
| - 0 | The number negative zero |
| 0n | BigInt, when used as a boolean, follows the same rule as a Number. On is falsy. |
| ш | Empty string value |
| null | null - the absence of any value |
| undefined | undefined - the primitive value |
| NaN | NaN - not a number |



ATRIBUINDO VALORES PADRÕES E EXECUTANDO CÓDIGO const var2 = outraVar ?? 0; // coalescencia nilla

const var1 = outraVar || 0;

- const var3 = true;
- if(var3) {
- console.log('Var3 é true');
- var3 && console.log('Var3 é true');



ESTRUTURAS DE REPETIÇÃO

- for;
- do... while
- while... do

- for... in (chaves de objetos)
- for... of (iterables)
- .forEach



ARRAYS

- São objetos em Javascript;
- Declaração:

var alunos = ["Maria", "Joao",,"Jose"]; // empty-slot na posição 2 (undefined)
console.log(alunos);
var alunos1 = new Array("Maria", "Joao","Jose"); // ", não funciona
console.log(alunos1);

- Principais propriedades, métodos e características:
 - O sort;
 - O push & pop;
 - O unshift & shift (remove o primeiro elemento);
 - O splice(inicio, quantosARemover, aAdicionar1, aAdicionar2, ...);
 - O reverse;
 - O slice(inicio, fim) (cria nova array com pedaço da primeira); // ver -1, -2
 - ... (spread ver Math.min e Math.max)
 - filter
 - O map
 - O reduce



USAR PONTO E VÍRGULA?

• Efeitos indesejados de não usar:

```
function getObjeto() {
function getObjeto() {
                                                                                   return
       return
                                             Ele interpretou
                                             assim
                                                                                      atributo: "valor"
          atributo: "valor"
                                                                           const a = 1
const a = 1
                                             Ele interpretou
                                                                           const b = 2
const b = 2
                                             assim
                                                                           const c = a + b(a + b).toString(); // TypeError: b is not a function
const c = a + b
(a + b).toString(); // TypeError: b is not a function
```