

Curso Básico de JavaScript - Apostila

Atribuição de variáveis

In [4]:

```
%%js  
  
var nomeDaVariavel = "camelCase padrão";  
const saoConstantes = "Diferente de variavel";  
  
nomeDaVariavel = "Posso atualizar uma variavel mas não posso atualizar uma const";
```

Tipos de dados em JavaScript

In [5]:

```
%%js  
  
var string = "Strings = conjunto de caracteres";  
var number = 1;  
var booleanoVerdadeiro = true;  
var booleanoFalso = false;
```

Operadores aritméticos em JavaScript

In [6]:

```
%%js  
  
var adicao = " + -> Adição";  
var subtracao = " - -> Subtração";  
var multiplicacao = " * -> Multiplicação";  
var divisao = " / -> Divisão";  
var exponenciacao = " ** -> Exponenciação";  
var modulo = " % -> Módulo da divisão ou Resto da divisão";
```

Operadores de comparação em JavaScript

In [7]:

```
%%js

/* Os operadores de comparação retornam valores
   booleanos Verdadeiro (true) ou Falso (false).

   EX: 1 < 2 -> true (1 é maior que 2?)
        2 > 3 -> false (2 é maior que 3?)
        1 <= 1 -> true (1 é menor ou igual a 1?)
        "A" === "A" -> true ("A" é igual a "A"?)
        "A" !== "a" -> true ("A" é diferente de "a" ?)

*/

var maiorQue = " > -> Maior que";
var menorQue = " < -> Menor que";
var maiorOuIgual = " >= -> Maior ou igual";
var menorOuIgual = " <= -> Menor ou igual";
var igualA = " === -> Igual a";
var diferenteDe = " !== -> Diferente de";

var igualA2 = " == -> Igual a, porém, tem algum detalhe específico";
var diferenteDe = " != -> Diferente de, porém, tem algum detalhe específico";
```

Operadores Lógicos em JavaScript

In [29]:

```
%%js

/* Operadores Lógicos:

   AND("E") -> &&
   OR("OU") -> ||

   Utilizando o "E" -> && -> quando necessitamos que
   TODAS condicionais sejam atendidas.
   EX: "Só vou à praia se fizer Sol e(&&) eu passar protetor"

   Utilizando o "OU" -> || -> quando necessitamos que
   ALGUMA das condicionais seja atendida.
   EX: "Vou a paraia se eu tiver protetor ou(||) se eu tiver dinheiro pra comprar prot
   etor "

*/

var tenhoProtetor = true; // Tenho protetor
var hojeFazSol = false; // Hoje não faz sol

var vouPraPraia = tenhoProtetor && hojeFazSol; // Como não faz sol não vou pra praia
console.log(vouPraPraia) // Não atendemos a todos os requisitos.

var tenhoProtetor = false;
var tenhoDinheiro = true;

var vouPraPraia = tenhoProtetor || tenhoDinheiro // Como tenho dinheiro vou pra praia
console.log(vouPraPraia) // Atendemos a pelo menos 1 dos requisitos.
```

Estruturas Condicionais em JavaScript

In [22]:

```
%%js

/* Nas estruturas condicionais temos IF, ELSE IF e ELSE
a estrutura básica de utilização das condicionais é EX:

    if (condição) {
        "o que será realizado caso condição for aceita"
    } else if (condição2) {
        "o que será realizado caso condição 2 for aceita"
    } else {
        "o que será realizado caso nenhuma das condições for aceita"
    }; */

var se = "if"
var casoContrario = "else"
var seIssoTambem = "else if"

if (se === "if") {
    console.log('Se é igual a IF faça o que está entre os parenteses')
} else if (casoContrario === "if") {
    console.log('Algo está errado aí, casoContrario teria que ser ELSE')
} else {
    console.log('Tudo deu errado, só sobrou nós aqui kkkk')
}
```

Arrays em JavaScript

In [17]:

```
%%js

var listaDeNomes = ["Esta", "é", "uma", "lista", "de", "nomes", "mas", "pode", "conter",
, "numeros", 1, 3, true];

listaDeNomes[0]; // Desta forma posso acessar um elemento da lista. Elemento de indice
0

listaDeNomes[listaDeNomes.length - 1]; // Também é possível acessar um elemento desta f
orma ultimo elemento da lista.

listaDeNomes.length; // Desta forma posso ver o nº de itens de uma lista

listaDeNomes.push("Adicionando um Elemento"); // O comando push adiciona um elemento na
lista

var tamanhoDoArray = listaDeNomes.length; // Posso atribuir o tamanho do array a uma va
riavel
```

Propriedades das Strings em JavaScript

In [21]:

```
%%js

/* As strings tem propriedades bem parecidas com os arrays
desta forma podemos acessar elentos pelo índice, concatenar
e também podemos medir o tamanho de uma string */

"Este é um teste".length; // Desta forma temos o comprimento de uma string

var nomeDaString = "Lucas";
var nomeDaString2 = "Santiago";

console.log(nomeDaString + nomeDaString2) // Esta é uma forma de realizar concatenação
console.log(nomeDaString + " " + nomeDaString2) // Esta é uma forma de acrescentar espa
ço entre strings

var iniciaisDoNome = nomeDaString[0] + nomeDaString2[0] // Concatenando elementos a par
tir do índice na string
console.log(iniciaisDoNome) // Imprimindo na tela LS iniciais concatenadas
```

Estrutura de Repetição -> For em JavaScript

In [28]:

```
%%js

/* Estruturas de repetição com for tem como caratecrística
segur um fluxo recorrente de trabalho. EX:

    1           2           4
    for(var declaraVar = 0; declaraVar <= 10; declaraVar++){
        console.log("O que será realizado")
    }           3

    Ou seja, 1 -> Definir Variavel, 2 -> Checar condição (t/f)
              3 -> Realiza a ação      4 -> Incrementa contador. */

var numero = 7;
var listaDeAlunos = ["Ana", "Lucas", "Paulo"];

for(var contador = 0; contador <= 10; contador++){
    var mensagem = numero + "x" + contador + "=" + numero * contador;
    console.log(mensagem)
}

for(var contador2 = 0; contador2 < listaDeAlunos.length; contador2++){
    var mensagem2 = "Bem vindo(a), " + listaDeAlunos[contador2] + "!";
    console.log(mensagem2)
}
```

Funções em JavaScript

In []:

```
%%js

/* -> Funções: São blocos de código que encapsulam instruções
    que executam uma tarefa específica.

    -> Objetivo: Modularizar o programa, ou seja, dividir nosso
    programa em partes menores em que cada parte
    tenha uma unica responsabilidade.
*/

// Estrutura de uma função:

function nomeDaFuncao(parametro){
    // o que a funcao faz
}

// Executando uma função:

nomeDaFuncao(valorDoParametro)

// Exemplo:

var statusCarro = true; // Se status true -> Liga o carro

function ligarOuDesligar(statusCarro){
    if(statusCarro === true){
        var mensagem = "Vrum... Carro ligado " + " statusCarro= " + statusCarro;
        return(mensagem);
    } else {
        var mensagemOff = "Puff... Carro desligado" + " statusCarro= " + statusCarro;
        return(mensagemOff);
    }
}

ligarOuDesligar(statusCarro) // Execução da função.
```