Módulo B

Aula 06

Manipulação de dados

Vamos aprnder sobre as diferença sobre o let, var e const

Valores primitivops:

Number

- Infinity

- NaN

String

Boolean

Null

Underfined

Object

- array

Function

Os dois principais

Number

Strring

E boolean

Para onde vai o valor digitado e vamos começar a manipular os dados

Temos que sempre fazer conversão de tipos

Sendo

window.Number

Number.parseInt(n)

Number.parseFloat(n)

Utilizando o Number.parseInt ou Number.parseFloat eu tenho que usar quando eu quero que realmente apenas fça isso sabe ? tipo onde preciso realmente converter o valor sempre para inteiro ou sempre para real (real é o numero com pontos “floats” como por exemplo 5.7 ou 5.5 ou 5.0 ta vendo que estou colocando ele sempre com uma cada decimal ? isso é um float

mAs p JS tem a função de spó aceitar que a operação é matemática utilizando o ‘Number”

se eu coloco apenas p Number na frente do comando window

por exemplo

var n1 = Number(window.prompt(“Digite um número”))  
com isso o Js já entende que é um valor “numérico” e não uma string

Agora para conveter um Number para String você pode suar

String(variável)

Ou

Variável.toString()

Templatestrings

Usar crases `` para conseguir usar ${variavel}

Sempre use as crases te ajuda demais

Outras variações

s.length = quantos caracteres a string tem

s.toUpperCase() = deixa as letras MAISCULAS

s.toLowerCase() = deixa as letras minúsculas

para a transformação de números podemos utilizar o “toFixed”

seria isso aqui:

var n1 = 1545.5555

se eu usar o n1.toFixed(2) ele fica

1545.55

Se eu quiser trocar o “.” Por “,” eu posso usar a seguinte:

N1.toFixed(2).replace(‘.’, ‘,’)

Sendo assim para de ser 1545.55 e passa a ser 1545,55

EXTRA:

Para transformar em valor monetário   
n1.toLocaleString(‘pt-BR’, {style: ‘currency’, currency: ‘BRL’})

R$ 1.545,55

Vamos falar de operadores

Aritméticos

Atribuição

Relacionados

Lógicos

Ternário

**Aritméticos e Atribuição**

**5 + 2: 10**

**5 – 2: 3**

**5 \* 2: 10**

**5 / 2: 2.5**

**5 % 2: 1 (resto da divisão)**

**5 \*\* 2: 25**

**Ordem de precedência**

**()**

**\*\***

**\* / %**

**+ -**

**var a = 5 + 3**

**var b = a % 5**

**var c = 5 \* b \*\* 2**

**var d = 10 – a / 2**

**var e = 6 \* 2 / d**

**var f = b % e + 4 / e**

**dicas**

**se você tiver uma varia e precisa fazer uma operação somente dela como por exemplo abaixo, você pode usar uns bizu**

**var n = 10**

**n = n-1, n vale agora 9**

**n = n+2, n vale 11 agora**

**e invés de seu fazer isso acima, eu posso fazer assim:**

**n-=1, n vale 9 agora**

**n+=2, n vale 11 agora**

**entendeu a logica ?**

**e tbm posso usar**

**n++**

**n—**

**Relacionais**

**5 > 2: true**

**7 < 4: false**

**8 >= 8: true**

**9 <= 7: false**

**5 == 5: true**

**4 != 4: false**

**Tbm podemos verificar se iguais idênticos e tals**

**Tipo assim**

**5 == 5: true**

**Pois 5 é igual a 5**

**5 == ‘5’: true**

**Mesmo sendo string 5 tem o mesmo valor de 5 porém com o “==” ele so verifica se o valor é igual**

**5 === ‘5’: false**

**Porque quando utilizamos o “===” nos perguntamos se o valor e tipo de 5 é igual ao valor e tipo de ‘5’ e da false**

**Pois 5 é um valor de 5 unidades e tem o tipo de number**

**Já**

**‘5’ tem o valor de 5 unidade mas o tipo é string**

**5 === 5: true (autoexplicativo o porque né)**

**LÓGICOS**

**!: negação(nosso “não”)**

**&&: conjunção (nosso “e”)**

**||: disjunção (nosso “ou”)**

**PRIMEIRO OS OPERADORES ARITMETICOS, DEPOIS OS OPERADORES RELACIONAIS E DEPOIS OPERADORES LOGICOS**

**var a = 5**

**var b = 8**

**a > b && b % 2 == 0**

**5 > 8 && 8 % 2 == 0**

**false: porque primeiro vamos fazer o resto da divisão de 8 % 2 == 0 ? Sim então é TRUE  
já depois vamos ver ver se A é > que B oque é FALSE porque o 5 não é maior que B**

**e como estamos usando o teste logico de “E” que é representado pelo && o resultado é false porque conforme ordem logica, precisamos TRUE e TRUE**

**se for FALSE e TRUE é FALSE**

**A ordem de precedência fica assim:**

**(): sempre oque esta parênteses primeiro**

**\*\*: sempre oque está sendo elevado a potencia**

**\* / %: após isso bem a multiplicação, divisão e resto de divisão, não nesta ordem, pois a ordem é de qual aparece primeiro**

**+ -: após vem a adição e subtração**

**> : maior que**

**< : menor que**

**>= : maior ou igual que**

**<= : menor ou igual a que**

**== : igual, porém sem identificar tipo**

**!= : Não é igual que**

**=== : igual identificando o tipo**

**!: Não**

**&&: e**

**||: ou**

**Operadores Ternários**

**Teste ? true : false**

**Exemplo**

**var x = 8**

**var res = x % 2 == 0 ? 5:9**

**aqui eu falo para a variável res receber 5 ou 9 depende se o resto da minha divisão, se o resto for 0 ele vai mostrar o 5 poois ele é true, mas se for 1 ele vai mostrar 9, com numeral é difícil mas imagine isso em string se for mair ou menor de idade**

**var idade = 19**

**var r = idade >= 18 ? ‘Maior’: ‘Menor’**

**vai aparecer Maior**

**ARVORE DOM**

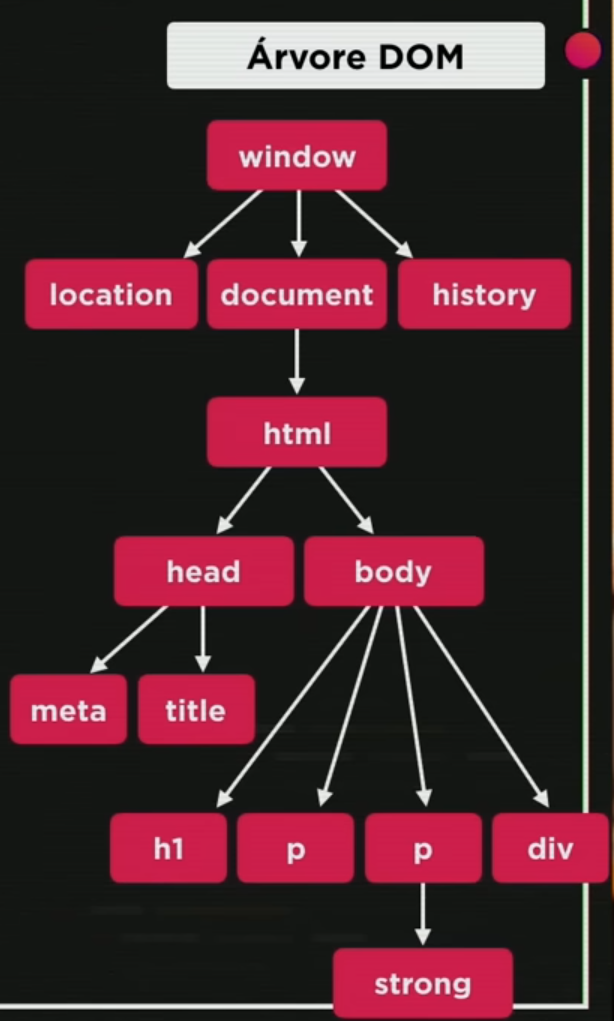
**window**

**location**

**document > html > head>body**

**history**

**quem está em cima é pai e quem está em baixo é child+**

****

**Eesses elementos podem ser selecionados por diversas maneiras, por exemplo:**

**Por Marca**

**document.getElementsByTagName(‘’)**

**Por ID**

**document.getElementById**

**Por Name**

**Por Class**

**Por Seletor**

para o Selector você precissa var d = document.querySelector('div#msg')

**EVENTOS DOM´S**

**Oque é DOM**

**Documento Object Model**

**Árvore DOM**

**Eventos DOM**

**<div>**

**É tudo aquilo que pode acontecer com esse elemento**

**Por exemplo**

**Mouseentrer quando chega o mouse dentro da div**

**Mousemove quando meu mouse esta movim,entando na div**

**Quando clico e seguro fica o mousedown**

**Quando eu solto o botão do mouse é o evento mouseup**

**Quando eu clico fica o “click”**

**Quando eu saio da div é o mouseout**

**Mas antes disso vamos aprender sobre FUNÇÕES**

**Elas serão executadas apenas quando evento acontecer**

**Para isso ocorrer a gente precisa configurar por exemplo**

**PRIMEIRO VAMOS CRIAR UM EVENTO DE CLIQUE DENTRO DO HTML**

**<body>**

**<div id=‘area’onclick=“clicar ()”>**

**Intereaja.....**

**</div>**

**<script>**

**function clicar(){**

**var a = document.getElementById(‘area’)**

**}**

**</script>**

**</body>**

**EXEMPLO NO HTML:**<!DOCTYPE html>

<html lang="pt-br">

<head>

    <meta charset="UTF-8">

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

    <title>Eventos DOM</title>

    <style>

        div#area{

            font: normal 20pt arial;

            background: forestgreen;

            color: white;

            width: 200px;

            height: 200px;

            line-height: 200px;

            text-align: center;

        }

    </style>

</head>

<body>

    <div id="area" onclick="clicar()" onmouseenter="entrar()" onmouseout="sair()">

        interaja .....

    </div>

    <script>

    var a = document.getElementById('area')

        function clicar(){

            a.innerText = 'Clicou !!'

            a.style.background ='red'

        }

        function entrar(){

            a.innerText = 'Entrou'

        }

        function sair(){

            a.innerText = 'Saiu!'

            a.style.background = 'green'

        }

    </script>

</body>

</html>

**EXEMPLO COM JAVASCRIPT**<!DOCTYPE html>

<html lang="pt-br">

<head>

    <meta charset="UTF-8">

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

    <title>Eventos DOM</title>

    <style>

        div#area{

            font: normal 20pt arial;

            background: forestgreen;

            color: white;

            width: 200px;

            height: 200px;

            line-height: 200px;

            text-align: center;

        }

    </style>

</head>

<body>

    <div id="area">

        interaja .....

    </div>

    <script>

    var a = document.getElementById('area')

    a.addEventListener('click', clicar)

    a.addEventListener('mouseenter', entrar)

    a.addEventListener('mouseout', sair)

        function clicar(){

            a.innerText = 'Clicou !!'

            a.style.background ='red'

        }

        function entrar(){

            a.innerText = 'Entrou'

        }

        function sair(){

            a.innerText = 'Saiu!'

            a.style.background = 'green'

        }

    </script>

</body>

</html>

**Fazendo um painel de SOMA**<!DOCTYPE html>

<html lang="pt-br">

<head>

    <meta charset="UTF-8">

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

    <title>Somando Números</title>

    <style>

        body{

            font: normal 18pt Arial;

        }

        input{

            font: normal 18pt Arial;

            width: 100px;

            text-align: center;

        }

    </style>

</head>

<body>

    <h1>Somando Valores</h1>

    <input type="number" name="txtn1" id="txtn1"> +

    <input type="number" name="txtn2" id="txtn2">

    <input type="button" value="Somar" onclick="somar()">

    <div id="res">Resultado</div>

    <script>

        function somar() {

            var tn1 = document.getElementById('txtn1')

            var tn2 = document.getElementById('txtn2')

            var res = document.getElementById('res')

            var n1 = Number(tn1.value)

            var n2 = Number(tn2.value)

            var s = n1 + n2

            res.innerHTML = `A soma entre ${n1} e ${n2} é igual a <strong>${s}</strong>`

        }

    </script>

</body>

</html>

**Aula**

**Condições**

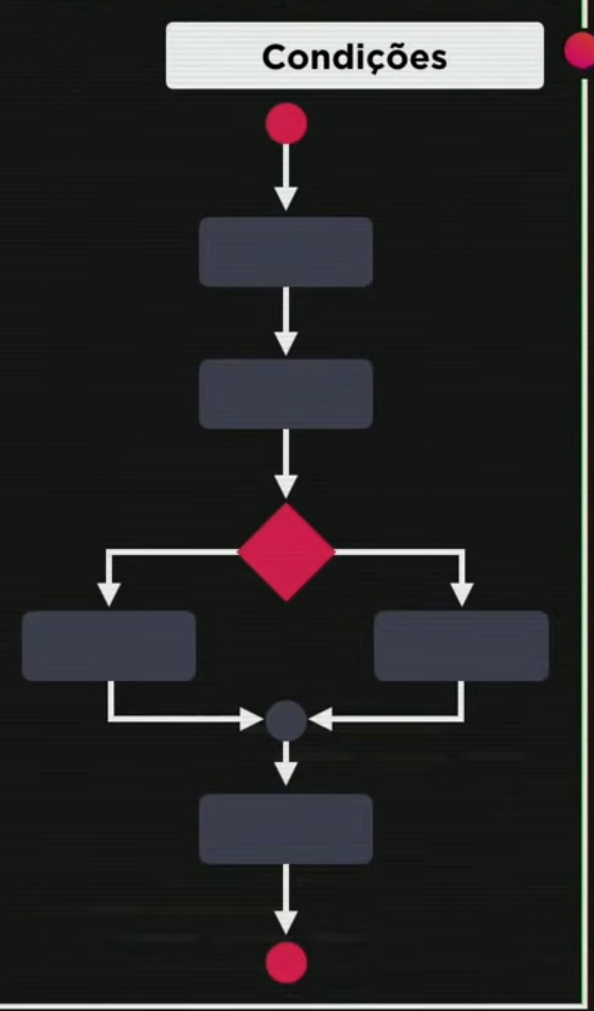
**SEQUÊNCIA**

**var n =**

**n +=2**

**window.alert(n)**

**Condições**

****

**O Losango vamos colocar como**

**PARA ABRIR UM BLOCO LEMBRER-SE QUE TEM QUE ABRIR E FECHAR CHAVES**

**if (condição){**

**true**

**} else {**

**false**

**}**

**Existem dois tipos de condições**

**Que ela só tem um tipo de bloco**

**É uma condição simples**

**if(condição){**

**true**

**}**

**Condição composta**

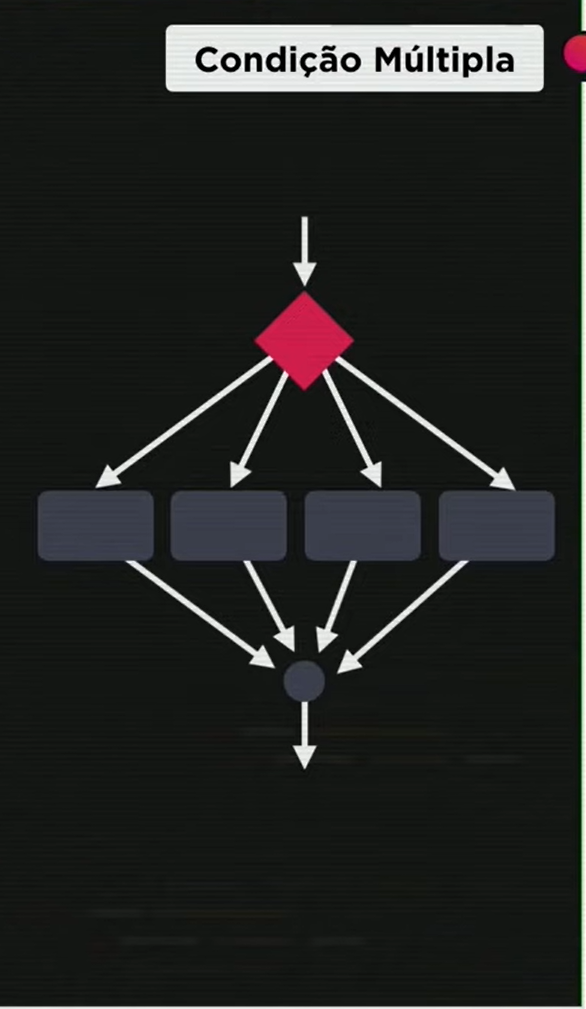
**if(condição){**

**true**

**} else {**

**false**

**}**

**Condições Múltiplas  
**

**Para efetuar usamos o**

**switch(expressão){**

**case valor 1:**

**no final sempre colocamos o break no final**

**case valor 2:**

**no final sempre colocamos o break no final**

**case valor 3:**

**no final sempre colocamos o break no final**

**default:**

**no final sempre colocamos o break no final**

**}**