* Para por uma mensagem de alerta antes de aparecer o conteúdo em si. Você usa a tag <script> e coloca depois window.alert(’ ’). Pode ver tbm na aula01

Exemplo>:

<script>

window.alert('minha primeira mensagem')

</script>

* Para por uma mensagem para confirmar algo referente ao site você usa a tag <script> window.confirm(‘ ‘)

Exemplo:

<script>

window.confirm('este site e para desenvolvedores de sistema, gostaria de proseguir')

</script>

* Se você quiser por uma mensagem perguntando nome, idade entre outros você coloca <script> window.prompt(‘ ‘)
* Exemplo:

<script>

window.prompt('qual e o seu nome?')

</script>

* = tem o significado em javascrip de receber, e o que for colocar depois dele, se não for numero e sim letras, deve colocar “ “ se for um objeto coloca{ }. Usado par guardar dados. Pode ver tbm na aula02 ex001

Exemplo:

Var nome = “Anderson” Var sofá = {20} var idade = 24

nome sofá idade

Anderson 20 24

Outro exemplo pratico

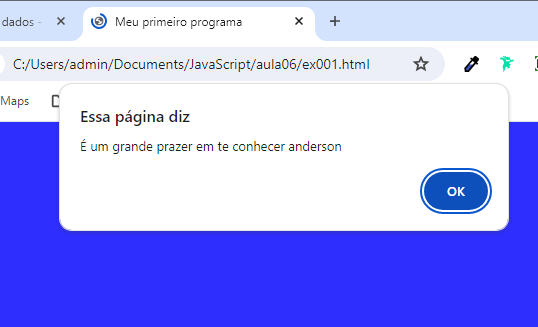
<script>

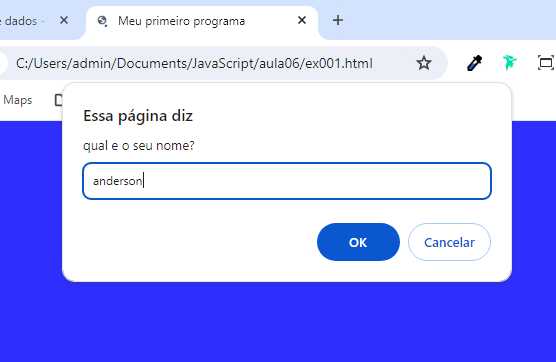
Var nome = window.prompt('qual e o seu nome?')

window.alert('É um grande prazer em te conhecer ' + nome)

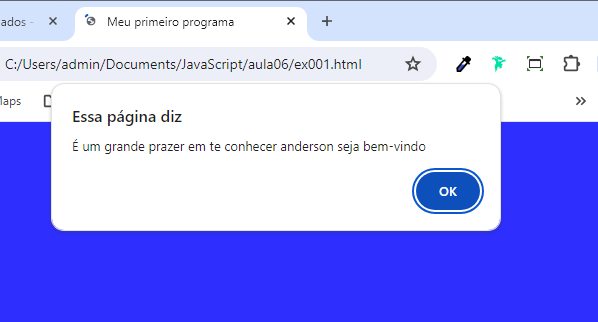
</script>

Se colocar o (+ nome) no final. depois que a pessoa escrever o nome dela, ira aparecer no (window alert) que foi escrito no exemplo mais nome da pessoa ficando a sim.





E se quiser colocar algo a mais depois nome da pessoa, da segunda imagem. E so colocar +’ ‘ e dentro das aspas você coloca o que você quer agregar a mais como + ’ seja bem-vindo’ ficara a sim



* + serve para adição e para concatenação, por isso as vezes tem que ser feito uma conversão que seria de string para number ou number para string. Tipos de conversão, Number.parseInt que converte de um numero par um numero inteiro, Number.parseFloat para usar números com ponto e virgula. Pode ver tbm na aula

Observação: a variante Number ira fazer o que as duas fazer automaticamente, as outras variantes, são usados caso que você queria forçar que so faça calculo com números inteiros ou quebrados. Number e mais usado

Exemplo de Number.parselInt:

<script>

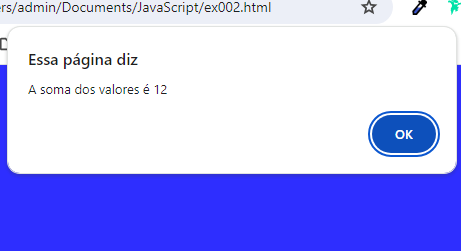
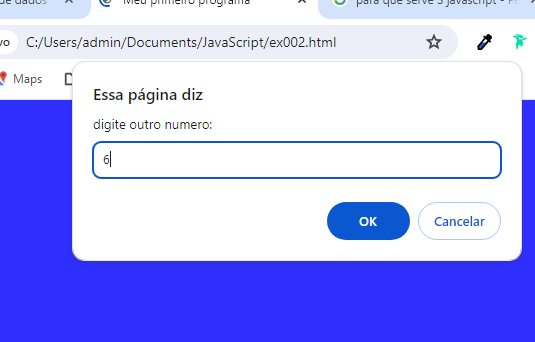
var n1 = Number.parseInt(window.prompt('digite um número:'))

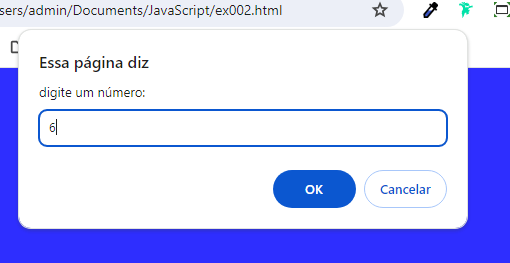
var n2 = Number.parseInt(window.prompt('digite outro numero:'))

var s = n1 + n2

window.alert('A soma dos valores é ' + s)

</script>





* E para números com ponto e virgula. exemplo de Number.Float

<script>

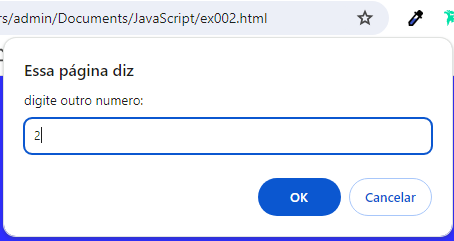
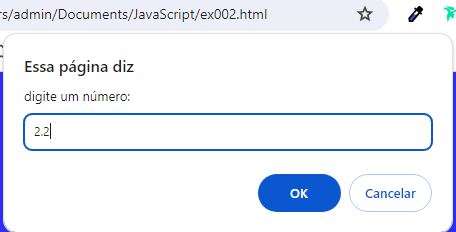
var n1 = Number.parseFloat(window.prompt('digite um número:'))

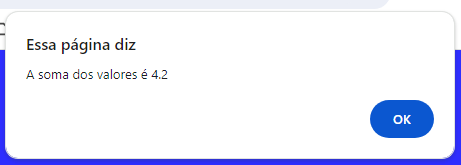
var n2 = Number.parseFloat(window.prompt('digite outro numero:'))

var s = n1 + n2

window.alert('A soma dos valores é ' + s)

</script>





Conversão de number para strig e feito com, string ou .tostring()

* Formatação de strings. Para deixar ele bem organizado e legível usasse tamplate string que e isso aqui `${ }` ai dentro das chaves colocasse a variante. Pode ver tbm na aula2 ex002

Exemplo de tamplate string:

<script>

var n1 = Number(window.prompt('digite um número:'))

var n2 = Number(window.prompt('digite outro numero:'))

var s = n1 + n2

window.alert(`A soma dos valores ${n1} e ${n2} é igual a ${s}`)

</script>

* E para colocar tudo em maiúsculas e .toUpperCase() e tudo minúsculas .to lowerCase(), lentgh para mostar numero de letras. E normalmente e sempre depois de document.write para mostrar quantas letras, se maiúsculo ou minúsculo. Aula2 ex003

<script>

var nome = window.prompt('Qual seu nome?')

document.write(`Olá, <strong>${nome}</strong> Seu nome tem ${nome.length} letras <br/>`)

document.write(`seu nome em Maiúsculo é ${nome.toUpperCase()} <br/>`)

document.write(`Seu nome em Minusculo é ${nome.toLowerCase()}`)

</script>

* .tofixed(2) para formatar números e para trocar o ponto por virgula e a variável .tofixed(2).replace(‘.’ , ‘,’)

Comando .tolocalstring(‘pt-br’, {style: ‘currecy’, currycy: ‘br’}) para mostrar o valor em reais do brasil, fazendo pontos e virgula no lugar certo e o R$

* Operadores:

Aritmético

atribuição

lógicos

ternário

* Aritmético

5 + 2 = 7 +somar

5 - 2 = 3 -diminui

5 \* 2 = 10 \*faz vezes

5 / 2 = 2.5 /dividi em números quebrados

5 % 2 = 1 % e para por o que resta da divisão feita

5 \*\* 2 = 25 \*\* faz ao quadrado

Obs: caso use todas esse meios, os calculas tem prioridade em quadrado, vezes e divisão. Para uma soma ou para diminuir ser feito primeiro deve colocar em ( )

Ordem de precedência

( )

\*\*

\*, /, %

+, -

* Auto-atribuições

Var n = 3 o n ele e 3 mais se você fizer um calculo com n com outros valores. Ficara no lugar do 3 por exemplo:

Var n = 3

n = n + 4 da 7 e sera o n no lugar do 3

n = n – 5 da 2 e 2 sera n no lugar do 7

você pode simplificar também, no lugar de n = n + 4. Da para colocar n + = 4, n - = 5. So o inicio do var que não da para fazer isso

* Relacionais

5 > 2 true por que 5 e maior que 2

7 < 4 false por que 7 não e menor que 4

8 >= 8 true por que fato que 8 e igual a 8

9 <= 7 false por que o 9 não e nem menor ou igual a 7

5 == 5 true por que 5 e igual a 5

5 === ‘5’ false por que ‘5’ não e idêntico ao 5

4 != 4 false por que 4 não e diferente de 4

> para ser true, o lado esquerdo de > tem que ser maior que o direito se não e false

< para ser true, o lado esquerdo de < tem que ser menor que o direito, se não e false

>= para ser true, o lado esquerdo de >= tem que ser maior ao igual, que o direito, se não e false

<= para ser true, o lado esquerdo de <= tem que ser menor ou igual, que o direito, se não e false

== para ser true, o lado esquerdo de == tem que ser igual ao direito, se não e false

=== para ser true, o lado esquerdo de === tem que ser idêntico ao direito se um for 5 e outro a sim ’5’ ou a sim “5” entre outros sera false

!= para ser true, o lado esquerdo de != não pode ser igual ao direito, se não e false

Exemplo mais pratico:

Preço >= 200.50 o preço e maior ou igual a 200.50?

Idade < 18 a idade é menor do que 18?

Curso == ‘javscript’ o curso e javascript

n1 != n2 o primeiro e diferente do segundo

Da para fazer um junção também de aritmético e relacionais por exemplo:

Var a = 8

Var b = 15

a <= b – 10

false por que 8 não e menor ou igual a 5

sempre e feito primeiro os aritméticos e depois relacionais

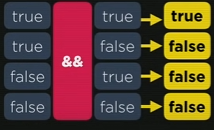
* lógicos

! para ser true, tem que conseguir qualquer coisa menos ao espefico, por exemplo, ! pede qualquer cor de caneta menos azul, se conseguir amarelo sera true, se for azul ser false

&& para ser true, tem que conseguir o que pede, esse && sempre pedi 2, e se pedir uma caneta azul e amarela. e conseguir essas canetas dessas cores sera true, mais se conseguir só uma das canetas ou nenhuma, serra false

|| para ser true basta conseguir 1 única coisa que pede, caso || pedir uma caneta azul e amarela, e vim só uma caneta azul ou amarela e true, e se não conseguir de nenhuma das core que queria sera false





Da para fazer um junção também de aritmético, relacionais e logicos por exemplo:

Var a = 5

Var b= 8

a > b && b % 2 == 0

false por que b % 2 que da 0 e 0==0 e true, só que o a não e maior que b então seria false && true, que da false

sempre e feito primeiro o aritméticos, relacionais e depois o logico

Exemplo mais pratico envolvendo relacionais e logico:

Idade >= 15 && idade <= 17 a idade esta entre 15 e 17?

Estado == ‘JR’ || estado ‘SP’ o estado e RJ ou SP?

Salario > 1500 && sexo != ‘M’

Ordem de precedência

( ), \*\*, /

>, <, >=

!, &&, ||

* Ternário

? bloco true

: bloco false

Exemplo:

Var media = 5.5

Media > 7 ?”aprovado” : “reprovado”

Reprovado por que a media 5.5 não e maior que 7

Da para fazer um junção também de aritmético, relacionais, logicos e ternário, por exemplo:

Var x=8

Var resultado = x % 2 == 0 ? 5 : 9

X

8

ressultado

5 deu true que 5, por que x % 2 e 0 e 0==0 e igual então o resultado e 5 por que ele não passou do 5 para o resultado ser 9

* Com o DOM da para ter acesso a vários componentes dentro do JavaScrip, a sim podendo navegar entre vários elementos do html. E para selecionar os elementos que deseja mexer, tem varias maneiras, como por marca, id, nome, classe e seletor. aula3 ex01

Exemplo:

Por marca e utilizado a tag getElementsByTagname( )

Var p1 = window.document.getElementsByTagname(‘p’) [0] o 0 significa que e o primeiro paragrafo se você quer mexer no segundo será 1 e a sim por diante

por id e utilizado a tag getElementByld( ) usado para selecionar um único elemento, e se usar o id em algum elemento, não precisar colocar em [] para escolher elemento você.

<div id=”msg”> aprendendo javascript </div>

Var d = window.document.getElementsByld(‘msg’)

por nome utiliza a tag getElementsByName( )

var h = window.document.getElementsByName(‘h1’)[1]

por classe e utilizado a tag getElementsByClassName( )

var h = window.document.getElementsByclassName(‘h1’)[1]

por seletor e utilizado a tag querySelector( ) que e para um único elemento, e para o plural querySelectorAll( )

<div id=”msg”> aprendendo javascript </div>

window.document.querySelector(‘div#msg’) o # e por que e um id se fosse uma classe seria .

var h = window.document.querySelectorAll(h1)[2]

* Eventos DOM de, fazer com que aparece algo escrito ou ate mesmo um favicom entre outros, quando você clicar ou passar o mouse na área. Aula4 ex01

mouseenter quando você chegar num elemento ele vai acionar o que você pós para aparecer

mousemove e para quando estiver movendo o mouse dentro do elemento

mousedown que e quando você clica no elemento

mouseup e quando você para de clicar o elemento

exemplo pratico:

<div id="area">interaja..</div>

<script> var a = window.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!'

a.style.background = 'blue'

function sair() {

a.innerText ='Saiu!'

a.style.background = 'rgb(42, 139, 42)'

</script>

mousedown que quando clica, fica em vermelho com a palavra Clicou!



 mouseenter que e quando passa ou mouse no elemento,ele fica em azul com a palavra Entrou!

 mouseout que quando tirar o mouse do elemento ficara verde escrito Saiu!

* function() executa uma ação (como um cálculo complexo) e retorna o resultado dessa ação como um valor. O valor de retorno pode ser um valor escalar (único) ou uma tabela. Aula4 ex02

Exemplo:

<h1>Somando valores</h1>

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

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

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

<div id="res">Resultado</div>

<script>

function somar() {

var tn1 = window.document.getElementById('txtn1')

var tn2 = window.document. querySelectoe ('input#txtn2')

window.document.querySelectoe('input#txtn2')

var res = window.document.getElementById('res')

var n1 = Number(tn1.value)

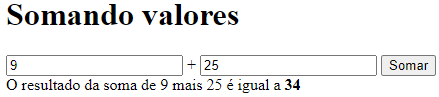
var n2 = Number(tn2.value)

var s = n1 + n2 res.innerHTML = `O resultado da soma de ${n1} mais ${n2} é igual a <strong>${s}</strong> `

}

</script>

Ficando a sim



* Document.write ser para escrever na tela, quando for dentro de html com <scrip>

Console.log você consegui digitar na tela, no javscript mesmo

* If significa: se por acaso. Aula5 ex01

Exemplo:

var vel = 60.5

console.log(`A velocidade do seu carro e ${vel}km/h`)

if (vel > 60) {

console.log('Voçê ultrapassou a velocidade permetida. Foi multado!')

}

Essa parte if (vel > 60) seria visto a sim. Se por acaso você ultrapassar a velocidade de 60

* Else significa: se não. Usado caso os if não tenha sido utilizados Aula 5 ex02

var pais ='EUA'

if (pais == 'Brasil') {

console.log('voçê e brasileiro')

} else {

console.log('Estrangeiro')

}

new Date ( ) e para por a hora exata

getHours para por a hora exata no sistema, ou getDay que e os dias da semana entre outros.

exemplo: aula6 ex02

var agora = new Date()

var hora = agora.getHours()

console.log(`Agora são extamente ${hora} horas`)

if (hora <12) {

console.log('Bom dia!')

} else if (hora <=18) {

consolelog(' Boa tarde')

} else {

console.log('Boa noite')

}

O **switch** controla o fluxo do programa permitindo ao programador especificar código diferente para ser executado em várias condições. Em seguida com o **case** você coloca o valor ou nome que for necessário e sempre a baixo do **case** tem que ter um **break**, e caso não tenha nada referente com o pede coloque **default** em baixo de **break** também. Usado mais em situações pontuais no lugar if, else.

Exemplo: aula6 ex03

var agora = new Date()

var diasem = agora.getDay()

console.log(diasem)

switch(diasem) {

case 0:

console.log('Domingo')

break

case 1:

console.log('Segunda')

break

case 2:

console.log('Terça')

        break

case 3:

console.log('Quarta')

break

case 4:

console.log('Quinta')

break

case 5:

console.log('Sexta')

break

case 6:

console.log('Sabado')

break

default:

console.log('[erro] Dia invalido')

break

}

* Img.src = ‘ ‘ e para por uma imagem. porem para aparecer em um certo momento, tipo no html foi posto uma imagem mais as 17horas quero que apareça outra imagem, deve criar um arquivo de script.js, e nesse arquivo você coloca img.src = ‘ ‘ ai coloca a pasta que imagem esta

Para por uma imagem só no javascrip como por exemplo, quero que apareça uma imagem de um jovem, adultou, ou idoso com forme sua idade, tem que por um document.creaElement e logo depois img.setAttribute(‘id’, ‘ ‘) para colocar o id e la no final da programação colocar um res,appendChild(img)

Exemplo pratico: aula08 ex01

var img = document.createElement('img')

img.setAttribute('id','foto')

if (sex[0].checked) {

genero = 'Homem'

if (idade >= 0 && idade < 10){

img.setAttribute('src', 'imagens/bebe-homem.png')

} else if (idade >= 10 && idade < 21)

res.innerHTML = `Detectamos ${genero} com ${idade} anos.`

res.appendChild(img)

obs: sempre cuida o tamanho da imagem, pode altera o posicionamento de um paragrafo e a foto caso, tenha colocado para ambos ficarem no meio e não estarem.

* Document.body.style.background = ‘ ‘ e para por uma caso precise em certo momento, tipo a cor esta vermelha, mais as 13 quero que fique azul
* Se você queira que o site carreguei diversas imagens em momentos do dia que você queria, na parte HTML em <body> Deve colocar onload() para carregar as imagens com forme a programação feita no javascript. <body onload=”carregar”>

Para por, para fazer uma verificação de alguma coisa como, botar para preencher os dados e a pessoa ver a sua verificação. Deve por no input buttom a tag onclick() para fazer uma verificação com forme a programação do javascript. <input type=”button” value=”verificação” onclick=”verificação()”>

* While( ) e uma estrutura de repetição.

Exemplo:

var c = 1

while (c <= 6) {

console.log(`passo ${c}`)

c = c + 1

}

Com essa programação fez, passo 1, passo 2, ate o 6 um embaixo do outro.

* For()
* Variável composta, para armazenar vários valores em uma mesmo estrutura em []

Exemplo: aula13 amb01

var valores = [2, 4, 8, 5, 7]

console.log(`${valores}`)

com essa programação ira aparecer os valores 2,4,8,5,7

e para por em ordem numérica tem que por um var que você escolheu e depois .sort()

exemplo:

var valores = [2, 4, 8, 5, 7,]

valores.sort()

console.log(`${valores}`)

com essa programação fez ficar a numeração em ordem a sim, 2,4,5,7,8

também da para adicionar um valor no vetor com .push()

* para ver os elementos use .length que e um atributo, que vai pegar todo o comprimento dele

exemplo: aula13 amb02

var valores = [2, 4, 8, 5, 7,]

valores.sort()

console.log(`${valores.length}`)

com essa programação, fez aparecer o numero de vetores que e 5

e para ver os vetores e os valores, sem precisar ficar colocando um por vez, e so usar for() e ele fica na vertical

exemplo:

var valores = [2, 4, 8, 5, 7,]

for(var pos = 0; pos < valores.length; pos++) {

console.log(`Aposição ${pos} tem o valor ${valores[pos]}`)

}

Com essa programação, ira fazer aparecer: a posição 0 tem o valor 2, a posição 1 tem o valor 4... e a sim por diante, ate o valor 7

A uma forma mais simplificada de fazer, para ver os vetores e os valores com in aula13 amb03

Exemplo: aula13 amb03

var valores = [2, 4, 8, 5, 7,]

valores.sort()

for(var pos in valores) {

console.log(`Aposição ${pos} tem o valor ${valores[pos]}`)

}

Com essa programação, ira fazer aparecer: a posição 0 tem o valor 2, a posição 1 tem o valor 4... e a sim por diante, ate o valor 7

Para buscar um valor dentro de um vetor use .index.Of() aii você coloca o valor que queira procurar e se não tiver ficara com -1

Exemplo: aula13 amb04

var valores = [2, 4, 8, 5, 7,]

console.log(valores)

var pos = valores.indexOf(8)

console.log(`Valor esta nesse vetor ${pos}`)

com essa programação mostrou que o valor 8 esta no vetor 2.

* Funções return, e para dar o retorno de algum com forme a programação feita.

Exemplo:

function parimpar(n) {

if (n%2 == 0) {return ' par'

} else {

return ' impar'

}

}

console.log(parimpar(23))

resultado deu impar por que 23 e um numero impar

* para soma também da

exemplo:

function soma(n1 = 0, n2 = 0) {

return n1 + n2

}

console.log(soma(2, 5))

deu o retorno de 7, porque a soma era 2 + 5

* também da para por uma função dentro de uma variável

exemplo:

var v = function(x) {

return x\*2

}

console.log(v(5))

retorno deu 10, porque fez 5 vezes 2

* calcular fatorial de um numero

exemplo:

function fatorial(n) {

var fat = 1

for(var c = n; c > 1; c--) {

fat \*= c

}

return fat

}

console.log(fatorial(5))

deu o retorno de 120, com o fatorial de 5x4x3x2x1

estudar sobre fatorial

* Da também para fazer uma fatorial recursiva

Exemplo:

function fatorial(n) {

if (n == 1){

return 1

} else {

return n \* fatorial(n - 1)

}

}

console.log(fatorial(5))

deu o retorno de 120, com o fatorial de 5x4x3x2x1

array para coisas simples, e declara com [], para declarar um objeto e com {},da para criar um item com (p){}.

exemplo:

var amigo = {nome: 'jose', sexo:'maculino', peso:'85.4', engordar(p = 0){

console.log('engordou')

this.peso += p

}}

amigo.engordar(2)

console.log(`${amigo.nome} pesa ${amigo.peso}kg`)

com essa programação fez isso:

engordou

jose pesa 85.42kg

estudar recursiva

var num = window.document.getElementById('inume')

var lista = window.document.getElementById('inumero')

var res = window.document.querySelector('div#res')

var valores = []

function isNumero(n) {

Função que isnumero que recebe numero

if(Number(n) >= 1 && Number(n) <= 100) {

Se por acaso o numero for maior ou igual 1 e menor ou igual a 100, ele so vai aceitar entre 1 a 100

return true Retornara true se for de 1 a 100

} else {

return false Se não for de 1 a 100 recebera false

}

}

function inLista(n, l) {

Função de inlista que vai ganhar um numero

if (l.indexOf(Number(n)) != -1) {

Se por acaso na lista o n for diferente de != -1 que significa que o valor não foi encontrado na lista

return true Retornara true se o numero não for encontrado na lista

} else {

return false Se foi o numero foi encontrado na lista e false

}

}

function adicionar() {

if(isNumero(num.value) && !inLista(num.value, valores)) {

E uma analise para ver se e um numero e se esta na lista

valores.push(Number(num.value))

Adiciona com push os números de num

var item = document.createElement('option')

crio o um elemento option com nome item

item.text = `Valor ${num.value} adicionado`

mostrara que o valor de num foi adicionado

lista.appendChild(item)

adicionou o item na lista

res.innerHTML = ''

quando for adicionado um elemento, o resultado será limpado

} else {

window.alert('Valor inválido ou já encontrado na lista.')

Se não for numero ou já esta na lista recebera o alert

}

num.value = ''

para esvaziar quando for adicionado um numero

num.focus()

para focar depois de adicionar um numero, e continuar adicionando

}

function finalizar() {

if (valores.length == 0) {

Se por acaso não tive nenhum valor

window.alert('Adicione valores antes de finalizar!')

Se não tiver valor aparecera’ adicione valores antes de finalizar

} else {

var tot = valores.length

Para saber quantos elementos o vetor tem

var maior = valores[0]

Maior valor 0

var menor = valores[0]

Menor valor 0

var soma = 0

var media = 0

for(var pos in valores) {

para cada posição em valores

soma += valores[pos]

para a Soma ficar recebendo vários valores

if (valores[pos] > maior)

se por acaso o valor pos for maior valor

maior = valores[pos]

se for maior será pos

if (valores[pos] < menor)

se por acaso o valor pos for menor valor

menor = valores[pos]

se for menor será pos

}

media = soma / tot

Media ira receber o valor de soma dividido por tot

res.innerHTML = ''

Resultado vazio

res.innerHTML += `<p> Ao todo, temos ${tot} numeros cadastrados</p>`

res.innerHTML += `<p> O maior valor informado foi ${maior}</p>`

res.innerHTML += `<p> O menor valor informado foi ${menor}</p>`

res.innerHTML += `<p> Somando todos os valores, teremos ${soma}</p>`

res.innerHTML += `<p> A média dos valores digitados é ${media} numeros cadastrados</p>`

}

}