



Técnico em

DESENVOLVIMENTO DE SISTEMAS

**PROGRAMAÇÃO WEB
FRONT-END**



Pauta:

- JavaScript externo
- Interagindo com campos de formulário
- innerHTML e innerText
- Exemplo e desafio
- Exercícios de fixação
- Desafios

Aula 04 de

JS



Vamos começar botando a mão na massa

1. Crie a pasta para a aula 4 de JavaScript.
2. Abra a sua pasta da aula 4 no VS Code (Atalho: Ctrl + K Ctrl + O)
3. Dentro da pasta da aula 4, crie uma pasta com o nome Exemplo1
4. Dentro da pasta Exemplo1, crie o arquivo exemplo1.html
5. Faça o código padrão html (Digite ! No início do código e aperte Enter)
6. Altere o atributo lang para "pt-br" e o title para Exemplo 1

```
<> exemplo1.html x
<> exemplo1.html > html
1  <!DOCTYPE html>
2  <html lang="pt-br">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport"
        content="width=device-width,
        initial-scale=1.0">
6      <title>Exemplo 1</title>
7  </head>
8  <body>
9
10 </body>
11 </html>
```



Criando campos de formulário

Dentro da tag body digite o seguinte código →

```
<body>
  <div class="box">
    <h1>Calculando Valores</h1>
    <label for="n1">Número 1</label>
    <input type="number" name="n1" id="n1">
    <br>
    <label for="n2">Número 2</label>
    <input type="number" name="n2" id="n2">
    <br>
    <button onclick="somar()">Somar</button>
    <label for="res">Resultado:"</label>
    <input type="number" name="res" id="res" readonly>
    <br>
  </div>
</body>
```

Nesse código temos uma div da classe box que será configurada para ficar no centro da página.

Dentro da div temos dois inputs onde serão inseridos valores, um botão que chamará uma função em JavaScript e um input somente leitura, não permitindo ao usuário digitar qualquer valor



Estilizando com CSS – Slide 1

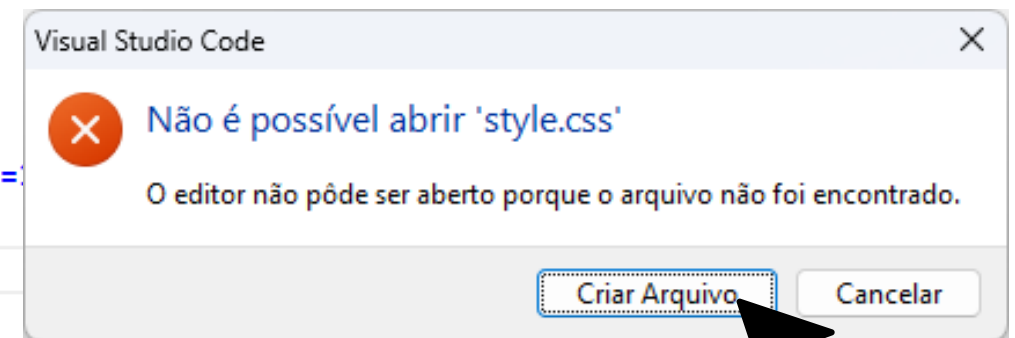
Dentro da tag head adicione a linha:

```
<link rel="stylesheet" href="style.css">
```






Agora, segure a tecla Ctrl e clique sobre o nome "**style.css**". O Visual Studio irá perguntar se deseja criar o arquivo, já que ele não existe. Clique em "Criar arquivo"


```
<head>  
  <meta charset="UTF-8">  
  <meta name="viewport" content="width=device-width initial-scale=1">  
  <title>Exemplo 1</title>  
  <link rel="stylesheet" href="style.css">  
</head>
```

Seguir o link (ctrl + clique)



Exemplo1 >  style.css > ...

```
1  body{
2      background-color:  grey;
3      color:  #FFF;
4      font: 1em Arial;
5      text-align: center;
6  }
7
8  .box{
9      width: 40%;
10     height: 80%;
11     margin: 50px auto;
12     border: 2px  black solid;
13     background-color:  #333;
14     border-radius: 10px;
15     box-shadow: 5px 5px 5px  #000;
16     text-align: center;
17 }
18
19 input {
20     height: 30px;
21     width: 80%;
22     margin-bottom: 20px;
23 }
```

```
24
25 #res {
26     font-size: larger;
27     color:  red;
28     text-align: center;
29 }
30
31 button {
32     width: 25%;
33     height: 50px;
34     border-radius: 10px;
35     margin: 10px 4px;
36 }
```



Fique à vontade para
estilizar de sua maneira!
Utilize Bootstrap se achar
melhor

Calculando Valores

Número 1

Número 2

Somar

Resultado:



Criando um arquivo JavaScript externo – slide 1

Podemos trabalhar com os scripts em um arquivo externo do tipo “.js”,

Isso traz mais organização ao nosso trabalho, pois, desta maneira podemos separar o código HTML do código JavaScript da mesma forma que fazemos com CSS.

Para carregar um arquivo JavaScript em nossa página devemos colocar a seguinte tag:

```
<script src="codigo.js">  
</script>
```

onde “codigo.js” deve ser o caminho até o seu código JavaScript.



Criando um arquivo JavaScript externo – slide 2

É uma boa prática carregar seus arquivos JavaScript externos no final do `<body>` do documento HTML para garantir que o HTML seja carregado antes de executar o JavaScript.

Portanto, adicione a seguinte linha no final do seu documento HTML
→

```
23     </div>
24     <script src="codigo.js"></script>
25 </body>
26 </html>
```

Repita o procedimento realizado no arquivo CSS para criar o arquivo “codigo.js”



Criando um arquivo JavaScript externo – slide 3

No arquivo `codigo.js` criado no slide anterior, digite o seguinte:

Exemplo1 > `JS` `codigo.js` > ...

```
1  let n1 = document.getElementById('n1')
2  let n2 = document.getElementById('n2')
3  let res = document.getElementById('res')
4
5  function somar(){
6      res.value = Number(n1.value) + Number(n2.value)
7  }
```



Agora é só testar...

The screenshot shows a web application interface with a dark gray background. At the top, the title 'Calculando Valores' is displayed in white. Below the title, there are two input fields. The first field is labeled 'Número 1' and contains the value '3'. The second field is labeled 'Número 2' and contains the value '4'. Below these fields is a light gray button with the text 'Somar'. At the bottom, there is a label 'Resultado:' followed by a white input field that contains the value '7' in red text.

Abra o seu arquivo index.html dentro da pasta da aula 4 e veja se ficou semelhante ao da imagem ao lado.

Teste digitando valores para o número 1 e número 2 e clique em somar.

Verifique se o resultado está aparecendo no campo resultado.



Comentando o código javascript

// Seleciona os elementos HTML com IDs específicos e atribui-os a variáveis.

```
let n1 = document.getElementById("n1")
```

```
let n2 = document.getElementById("n2")
```

```
let res = document.getElementById("res")
```

// Essa função será chamada quando o botão "Somar" for clicado.

```
function somar() {
```

```
    // Obtém os valores dos campos de entrada n1 e n2 e converte-os em números.
```

```
    // Realiza a operação de soma e atribui o resultado ao campo de entrada res.
```

```
    res.value = Number(n1.value) + Number(n2.value)
```

```
}
```



Exibindo o resultado em uma tag de parágrafo - Alterando o HTML

Vamos alterar um pouco nosso código para que, ao invés de exibir o resultado dentro de um input, exiba o resultado dentro de uma tag de parágrafo.

Altere o seu código HTML, retirando o campo input com id “res” (ou apenas comentando essa parte) e adicione um parágrafo com o id “res”

```
20      <label for="res">Resultado:</label>
21      <!-- <input type="number" name="res" id="res" readonly> -->
22      <p id="res">Aqui irá aparecer o resultado...</p>
```



Exibindo o resultado em uma tag de parágrafo - Alterando o JS

Agora vamos alterar o código JavaScript para que ele exiba o resultado na tag de parágrafo. Para isto, vamos utilizar a propriedade **innerHTML**. Veja como fica o nosso código:

```
function somar(){  
    res.innerHTML += `

O resultado da soma entre ${n1.value} e ${n2.value} é  
    ${Number(n1.value) + Number(n2.value)}</p>`  
}


```



Testando a calculadora

Calculando Valores

Número 1

Número 2

Resultado:

Aqui irá aparecer o resultado...

O resultado da soma entre 1 e 2 é 3

O resultado da soma entre 1 e 45 é 46

O resultado da soma entre 34 e 45 é 79

Teste digitando valores para o número 1 e número 2 e clique em somar.

Verifique se o resultados estão aparecendo no campo resultado.



Entendendo mais o **innerHTML**

O **innerHTML** é uma propriedade de elementos HTML.

Ele permite que você obtenha ou defina o conteúdo HTML de um elemento.

Quando você o utiliza para definir um valor, ele substitui o conteúdo existente do elemento pelo novo conteúdo HTML fornecido.

Existe também a propriedade **innerText**, que substitui somente o conteúdo de texto visível de um elemento HTML.

A diferença entre os dois é que o **innerHTML** faz interpretação de tags HTML e CSS, enquanto o **innerText** trata o conteúdo como texto simples.



Desafio

Adicione as seguintes funcionalidades em sua página:

- **Adicione um botão** para as demais operações matemáticas: (subtração, divisão, multiplicação) * não se esqueça que não existe divisão por 0
- Os resultados deverão ser **acumulados** na tag parágrafo, exibindo todos os resultados que forem acionados pelo usuário
- Adicionar um botão **“Limpar”** para resetar os valores.
- 🔍 📖 Pesquise sobre a biblioteca Math em JS e adicione as funções de **potência** e **raiz quadrada** em seu exercício.

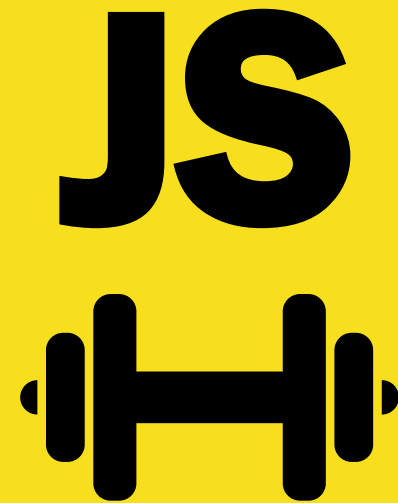
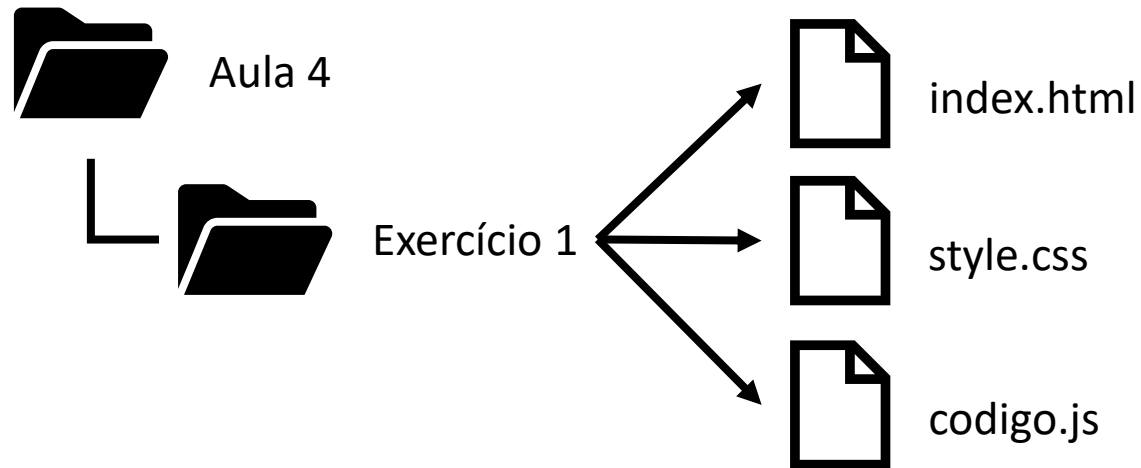




Exercícios de fixação

Crie uma pasta para cada um dos exercícios a seguir.

Cada pasta deverá conter o arquivo html, css e js do exercício.





Exercício 1 - CÁLCULO DE HORAS EXTRAS

Crie uma página para ajudar os(as) funcionários(as) a calcularem o valor que irão receber pelas horas extras trabalhadas:

O(a) colaborador(a) deverá inserir as seguintes informações:

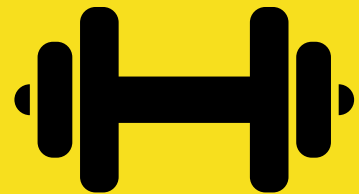
- Valor do salário mensal bruto;
- Quantidade de horas extras em dias normais (úteis);
- Quantidade de horas extras realizadas aos finais de semana;

Para calcular, divida o salário bruto por 200 (total de horas mensais) para descobrir o valor da hora trabalhada. Depois multiplique pela quantidade de horas extras normais.

O valor das horas extras aos finais de semana deverá receber um acréscimo 50%.

Some e mostre o valor à receber.

JS





Exercício 2 - AUMENTO DE SALÁRIO

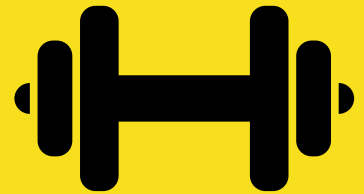
Crie um programa para atender a necessidade abaixo:

O gestor deverá informar o salário de um funcionário e o software calcular o valor do aumento conforme regra abaixo:

- Salários até R\$ 1.200,00: aumento de 16%.
- Salários de R\$ 1.200,01 até R\$ 2.100,00: aumento de 13%.
- Salários de R\$ 2.100,01 até R\$ 3.000,00: aumento de 10%.
- Salários acima de R\$ 3.000,00: aumento de 5%.

Exiba o novo salário após o aumento.

JS



%



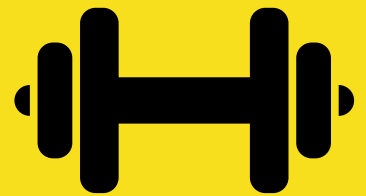
Exercício 3 - TRANSPORTE DE COLABORADORES

Desenvolva um programa onde o gerente deverá informar a quantidade de funcionários que necessitarão de transporte fretado e a quantidade de dias úteis em que o transporte deverá acontecer.

Mostre o custo mensal de acordo com a tabela abaixo:

QUANTIDADE DE PESSOAS	VALOR POR PESSOA / DIA
1 a 49	R\$ 4,50
50 a 99	R\$ 4,10
100 a 149	R\$ 3,80
150 a +	R\$ 3,60

JS





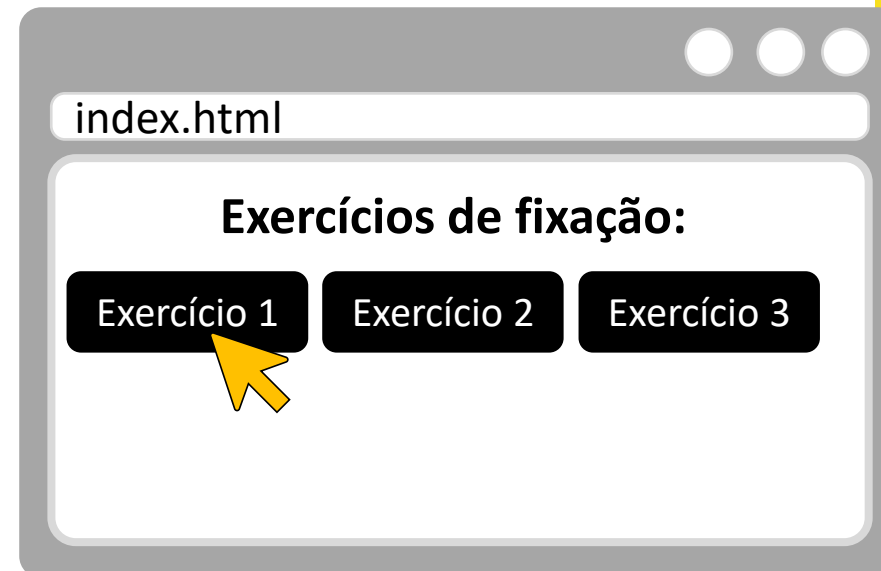
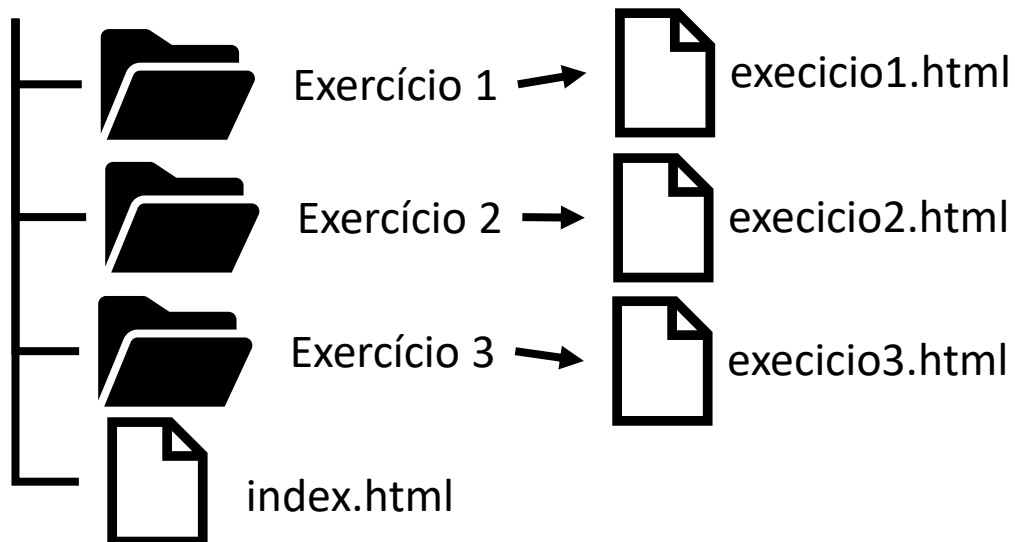
Desafio 1



Após realizar os 3 exercícios de fixação, construa uma página inicial que deverá conter 3 links, um para cada exercício. Nos exercícios, construa um link que o leve de volta para a página inicial.



Aula 5



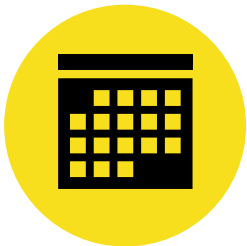
JS





Desafio 2

Adicione a data e a hora atual do sistema do usuário nas páginas HTML dos exercícios.



Para isto, pesquise sobre o objeto Date do Javascript e veja como você pode colocar essa informação em sua página

Caso peça auxílio ao ChatGPT para esta função, peça para que ele comente o código para melhor entendimento.

JS

