

Projeto 08

Header fixo e botões JavaScript



Disciplina:

Introdução a programação WEB

Projeto 08

Header fixo e botões JavaScript

01 – No Windows Explorer, criar uma pasta de nome “projetoshtml”
(caso a pasta já esteja criada, desconsidere esse passo)

02 - Abrir o Visual Studio Code.

03 - Selecionar a pasta dos projetos ou Workspace: “projetoshtml”

04 - Criar a subpasta do projeto: projeto08

05 - Criar o arquivo HTML: [index.html](#)

06 – Criar o “stub” (esboço ou estrutura básica) do arquivo html5.

07 – Confirmar ...

OBSERVAÇÃO: NO VS CODE

Use [CTRL+S](#) para salvar (gravar) o arquivo alterado;

Use [ALT+SHIFT+F](#) com o cursor dentro do <body> para ajustar o texto;

Use [CTRL+B](#) para esconder ou mostrar o explorador;

Use [CTRL+F5](#) (run without debugg) para executar o arquivo HTML;

Use [CTRL+;](#) para inserir um comentário;

Digite ! Seguido da tecla [`<enter>`](#) após criar o arquivo index.html (para gerar o esboço)

Digite ht, escolha HTML5 após criar o arquivo index.html (para gerar o esboço)

Use [CTRL + Shift + P](#), para recarregar a janela (no caso, por exemplo, de dependências de extensões do VS CODE)

Projeto 08

Header fixo e botões JavaScript

Arquivo index.html gerado e alterado:

```
1  <!DOCTYPE html>
2  <html lang="pt-BR">
3  <head>
4      <meta charset="UTF-8">
5      <meta http-equiv="X-UA-Compatible" content="IE=edge">
6      <meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1.0">
7      <title>Header fixo e botões</title>
8      <style>
9
10     </style>
11  </head>
12  <body>
13
14
15  </body>
16  </html>
```

OBSERVAÇÃO SOBRE O “MAXIMUM-SCALE=1.0”

O atributo `maximum-scale=1.0` é usado para controlar o nível máximo de zoom nos dispositivos móveis. É usado para: prevenir zoom excessivo, melhor experiência de usuário, evitar problemas de acessibilidade e impedir comportamentos indesejados em formulários.

O atributo `maximum-scale=1.0` deve ser usado com cautela para não comprometer a acessibilidade, porque ele impede que usuários com baixa visão ampliem o conteúdo da página acima de um limite (1.0).

08 – Alterar o tipo de linguagem, de “em” para “pt-BR”

09 – Alterar o título (aba) da página para: Header fixo e botões

OBSERVAÇÃO:

Usaremos o estilo internamente ao arquivo HTML

10 – Altere a viewport para:

```
<!-- SUGERIDO PARA EVITAR PROBLEMAS NA RESPONSIVIDADE -->
<meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1.0">
```

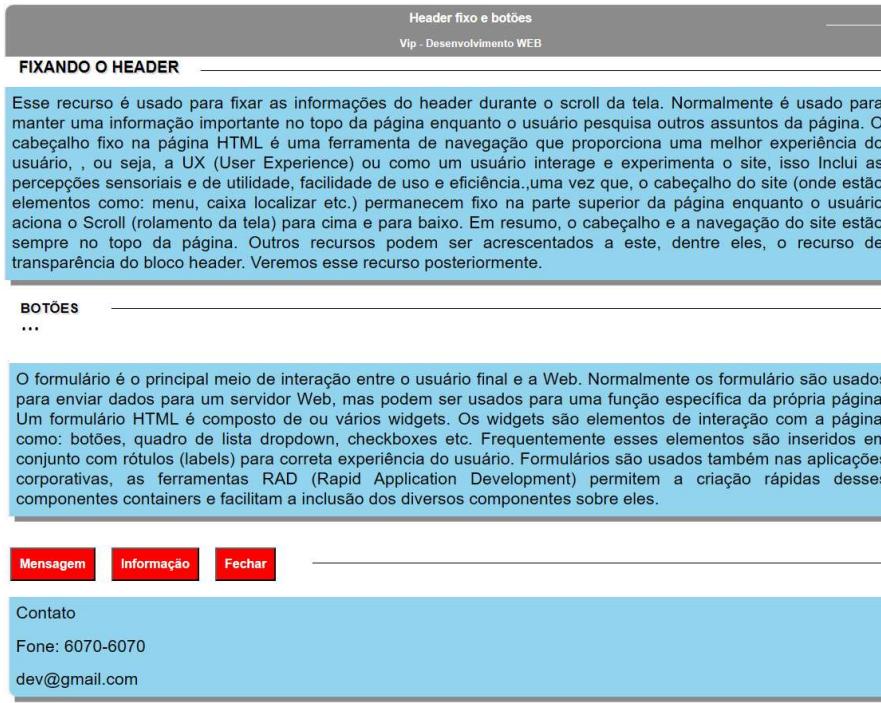
Inserir o texto fornecido no corpo (`<body>`) da página HTML.

Vamos trabalhar com estilo CSS na própria página HTML

Projeto 08

Header fixo e botões JavaScript

Layout da página a ser criada



* Seguir as indicações de marcações para para títulos, parágrafos e layout

11 – Inserir o texto fornecido, no corpo da página HTML:

12 – Criar o seletor universal (*) para definir as seguintes características:

```
* {  
    margin: 0px;  
    border: 0px;  
    padding: 0px;  
}
```

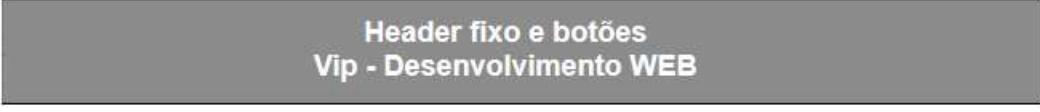
Projeto 08

Header fixo e botões JavaScript

13 – Vamos usar a unidade **REM**. Criar o seletor global (`html { ... }`) para definir o valor base da unidade relativa “rem” a ser usada na página. Usar `font-size` igual a **62.5%** para fazer o mapeamento de 16 pixels para 10 pixels;

```
html {  
    font-size: 62.5%;  
}
```

14 – Criar o header da página, conforme layout, usar os marcadores `<h3>` e `<h4>` respectivamente:



Header fixo e botões
Vip - Desenvolvimento WEB

15 – Criar o seletor `<header>` e inserir as seguintes propriedades:

```
header {  
    background-color: #rgba(128, 128, 128, 1);  
    Cor branco  
    Família de fonte Arial  
    Largura 100%  
    preenchimento (padding) 10 px  
    Alinhamento centralizado  
    Posição (position) fixado  
}
```

Posição fixado (`position: fixed`) Faz o header ficar fixo na tela
Alinhamento = Alinhamento do texto

SOBRE A FUNÇÃO RGB()

1 - A função `rgb(n1, n2, n3)` é usada para definir cores usando os valores: vermelho, verde e azul, respectivamente: n1, n2 e n3, valores no intervalo de 0 a 255.

2 - A função `rgba(n1, n2, n3, n4)` adiciona um quarto parâmetro, n4, que representa a transparência (opacidade) no intervalo de 0 a 1.

3 – `rgb()` é uma função do CSS (cascade Style Sheet)

Projeto 08

Header fixo e botões JavaScript

16 – Definir as propriedades para os parágrafos

```
p {  
    Família de fonte somente Arial  
    Tamanho do fonte 1.6 rem  
    Alinhamento do parágrafo justificado  
  
    Cor de fundo #87CEEB (SkyBlue)  
    box-shadow: 8px 8px #808080;  
  
    Largura 95%  
    Preehcimento de topo 10 pixels  
    preenchimento à esquerda 10 pixels  
    Preehcimento inferior 10 pixels  
    preenchimento à direita 10 pixels  
}
```

The code is annotated with curly braces to group properties into three categories:

- A brace on the right side groups the first three properties: "Família de fonte somente Arial", "Tamanho do fonte 1.6 rem", and "Alinhamento do parágrafo justificado". This group is labeled "Fontes".
- A brace on the right side groups the next two properties: "Cor de fundo #87CEEB (SkyBlue)" and "box-shadow: 8px 8px #808080;". This group is labeled "Cores e efeito de sobreamento do bloco".
- A brace on the right side groups the last four properties: "Largura 95%", "Preehcimento de topo 10 pixels", "preenchimento à esquerda 10 pixels", "Preehcimento inferior 10 pixels", and "preenchimento à direita 10 pixels". This group is labeled "Dimensionamento e preenchimento dos elementos do bloco".

Preenchimento = padding

box-shadow: deslocamento-x (horizontal) deslocamento-y (vertical) desfoque espalhamento cor do sobreamento;

Projeto 08

Header fixo e botões JavaScript

17 – Definir as propriedades para o título e os subtítulos

```
h1,  
h2 {  
    Família de fonte Arial  
    Alinhamento do texto à esquerda } Fontes  
  
    Preenchimento de topo 15 pixel  
    Preenchimento inferior 15 pixel  
    Preenchimento à esquerda 20 pixel } Dimensionamento e preenchimento  
    dos elementos do bloco  
  
    text-shadow: 2px 2px 0 #D3D3D3; } Sobreamento do título e subtítulos  
}  
  
h1 {  
    Tamanho do fonte: 2 rem } Formatação específica  
}  
  
h2 {  
    Tamanho do fonte: 1.6 rem }
```

FORMULÁRIOS

text-shadow

18 – Criar o rodapé da página com os seguintes dados abaixo. Criar uma classe para definir propriedades específicas para os parágrafos do rodapé.

```
<footer>  
    <p class="rodape">Contato:</p>  
    <p class="rodape">Fone: 60 6070-6070</p>  
    <p class="rodape">dev@gmail.com</p>  
</footer>
```

Observação:

A classe criada é necessária uma vez que as definições de estilo dos parágrafos do corpo da página `<body>` são replicadas nos parágrafos do `<footer>` (rodapé).

Projeto 08

Header fixo e botões JavaScript

19 – Definir, no estilo, as propriedades do rodapé:

```
.rodape {  
    background-color: #rgba(128, 128, 128, 1);  
    Cor branca  
  
    Família de fonte Arial  
    Tamanho da fonte 1.6 rem  
    Alinhamento do texto centralizado  
  
    box-shadow: none;  
    Largura 100%  
    Preenchimento 5 pixels  
}
```

Faça as alterações necessárias para espaçamento dos blocos, use padding ou quebra de linha
.

20 – Observe que criando o header fixo, o bloco do <header> ficar estático. Reduza a janela para ver esse resultado no scroll da janela.

```
header {  
    background-color: #rgba(128, 128, 128, 1);  
    Cor branca  
    Família de fonte Arial  
    Largura 100%  
    Preenchimento 10 pixels  
    Alinhamento do texto Centralizado  
    position: fixed; } 
```

OBSERVAÇÕES:

Faça as alterações necessárias no <header> da página.

Experimente alterar o último parâmetro da propriedade background-color para 0.4

Veja o resultado ...

```
background-color: #rgba(128, 128, 128, 0.4);
```

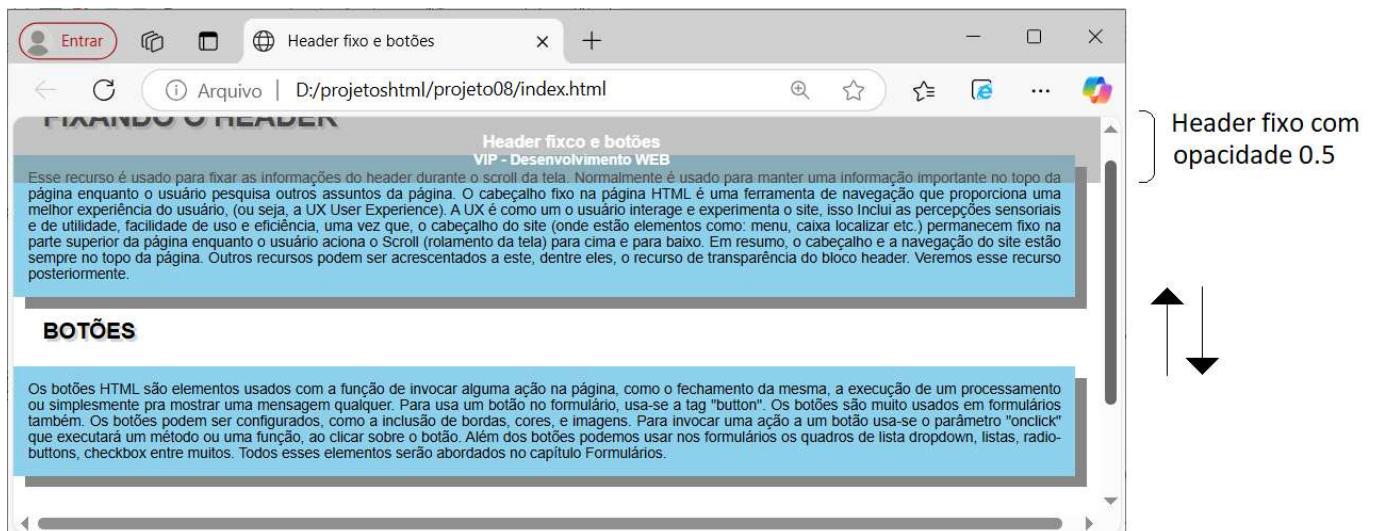
Projeto 08

Header fixo e botões JavaScript

21 – Para conseguir o efeito transparência no <header> há duas maneiras:

```
header {  
    background-color: rgba(128, 128, 128, 0.5);  
    color: #ffffff;  
    font-family: arial;  
    width: 100%;  
    padding: 10px;  
    text-align: center;  
    position: fixed;  
}  
  
... OU ...  
  
header {  
    background-color: rgba(128, 128, 128, 1);  
    color: #ffffff;  
    font-family: arial;  
    width: 100%;  
    padding: 10px;  
    text-align: center;  
    position: fixed;  
    opacity: 50%;  
}
```

Header fixo com opacidade (efeito transparência) 0.5 ou 50%



Projeto 08

Header fixo e botões JavaScript

22 – Inserir antes do <footer> um botão de comando para exibir uma mensagem na página.

Parte **HTML** do botão:

```
<br>
<button class="botoes" title="Mostrar mensagem da página"
name="btnMsg" value="OK" type="button"
onclick="mensagem()"> Mensagem </button>&ampnbsp&ampnbsp&ampnbsp&ampnbsp
<!-- Foram inseridos 4 espaços (nbsp = Non-breaking space )-->
<!-- entre esse botão e o próximo -->
```

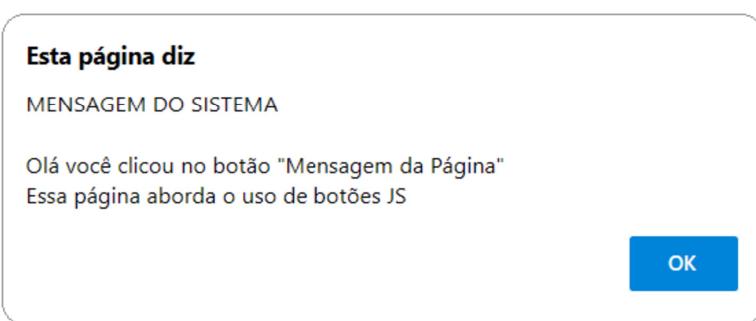
Parte **Script** do botão (Linguagem JavaScript):

```
<!-- Exemplo Alert() simples -->
<script>
    function mensagem() {
        alert("MENSAGEM DO SISTEMA\n\n" +
            "Olá você clicou no botão \"Mensagem da Página\" " +
            "\nEssa página aborda o uso de botões JS");
    }
</script>
```

O formulário é o principal meio de interação entre o usuário final e a Web. Normalmente os formulários são usados para enviar dados para um servidor Web, mas podem ser usados para uma função específica da própria página, com a página, como: botões, quadro de lista dropdown, checkboxes etc. Frequentemente esses elementos são inseridos em conjunto com rótulos (labels) para correta experiência do usuário. Formulários são usados também para criação rápidas desses componentes containers e facilitam a inclusão dos diversos componentes sobre eles.



Ao clicar no botão o seguinte resultado será mostrado:



Projeto 08

Header fixo e botões JavaScript

23 – Inserir entre o botão de comando inserido anteriormente e o <footer> um outro botão de comando com uma caixa de confirmação.

Parte **HTML** do botão:

```
<!-- Exemplo confirm() -->
<button class="botões" title="Confirmação"
name="btnConfirm" value="#" onclick="confirmar()"> Informação
</button>&nbsp&nbsp&nbsp&nbsp
```

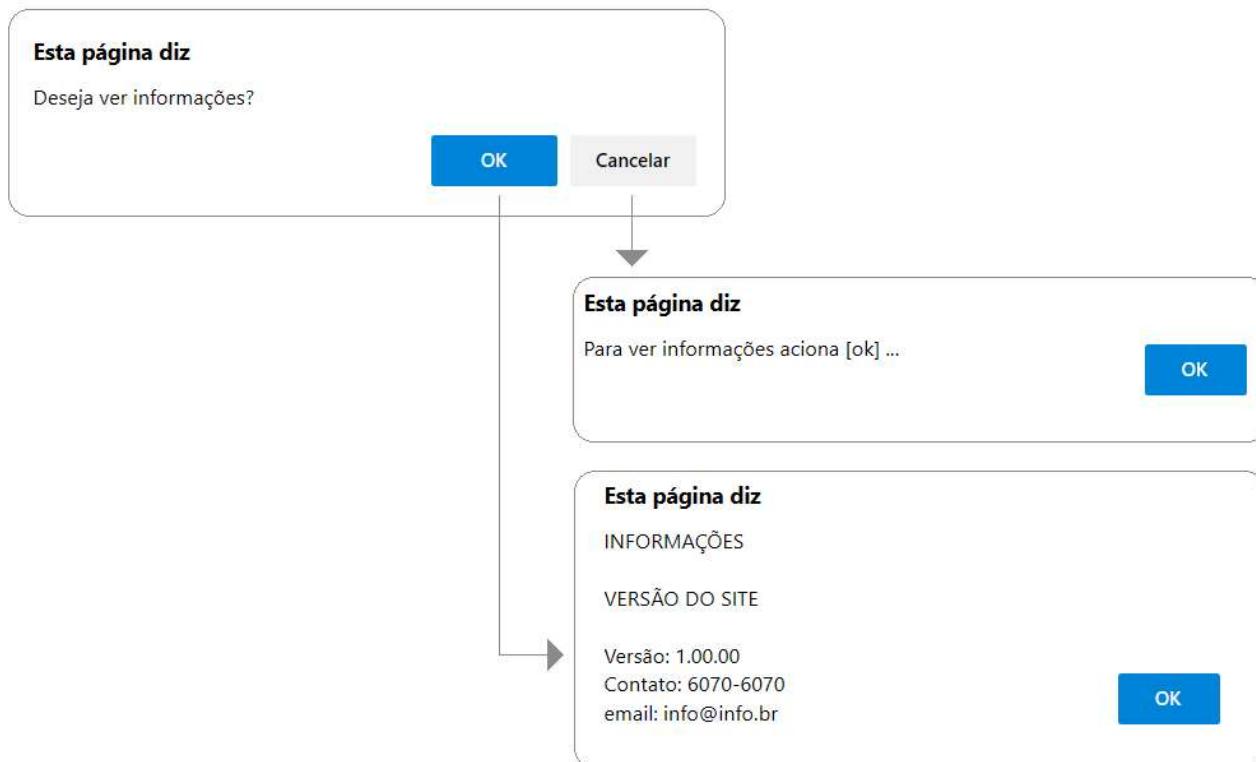
Parte **Script** do botão (Linguagem JavaScript):

```
<script>
    function confirmar() {
        var resultado = confirm("Deseja ver informações?");
        if (resultado == true) {
            alert("INFORMAÇÕES\n\nVERSÃO DO SITE\n\n" + "Versão: 1.00.00\n" +
            "Contato: 6070-6070\n" + "email: info@info.br");
        }
        else {
            alert("Para ver informações aciona [ok] ...")
        }
    }
</script>
```

Projeto 08

Header fixo e botões JavaScript

24 – Ao clicar no botão o seguinte resultado será mostrado:



Observe a estrutura **if/else** no script do botão.

Projeto 08

Header fixo e botões JavaScript

25 – Inserir entre o botão de comando inserido anteriormente e o <footer> um outro botão de comando para fechar o formulário.

Parte **HTML** do botão:

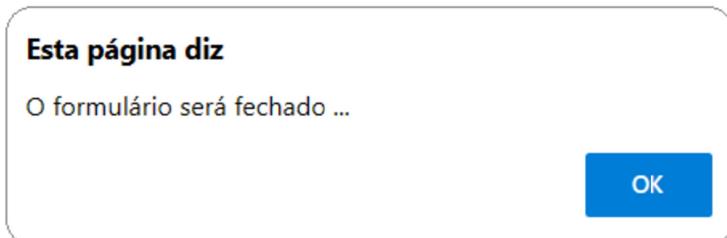
```
<!-- Exemplo do método Close() -->
<button class="botoes" title="Fechar o formulário ..."
name="btnFechar" value="OK" type="button" onclick="fechar()">
Fechar </button>
```

Parte **Script** do botão (Linguagem JavaScript):

```
<script>
    function fechar() {
        alert("O formulário será fechado ...")
        close()
    }
</script>
<br><br>
```



Ao clicar no botão o seguinte resultado será mostrado:



Projeto 08

Header fixo e botões JavaScript

26 – Aplicar os estilos no botão:

```
.botoes {  
    background-color: #ff0000;  
    color: #fff;  
    font-size: 16px;  
    text-align: center;  
    padding: 10px;  
    font-weight: bold;  
}
```

27 – Para inserir uma borda nos botões use a propriedade border no estilo (.botoes):

```
.botoes {  
    ...  
    border: 8px groove #D3D3D3; /* Light gray*/  
}
```

28 – Para inserir uma separação entre os botões use: (Non-breaking space) entre os botões.

```
<button class="botoes" title="Mostrar mensagem da página" name="btnMsg"  
       onclick="mensagem()>Mensagem</button>  
</button>&nbsp&nbsp&nbsp&nbsp  
        ↑  ↑  ↑  ↑
```

Aplique após o primeiro e o segundo botão.

29 – Para inserir espaços entre os blocos use
 ou crie um separador. Caso um dos blocos (<header>, <body> ou <footer>) fique desconfigurado, altere a propriedade width,

Projeto 08

Header fixo e botões JavaScript

PARA CRIAR O SEPARADOR:

(1) Crie a div e class no arquivo HTML

```
<div class="separador"></div>
```

(2) Crie o estilo no CSS

```
.separador {  
    padding-top: 10px;  
    padding-bottom: 10px;  
}
```

Botões JavaScript inseridos no formulário (após aplicar os estilos):

...

FORMULÁRIOS

O formulário é o principal meio de interação entre o usuário final e a Web. Normalmente os formulários são usados para enviar dados para um servidor Web, mas podem ser usados para uma função específica da própria página. Um formulário HTML é composto de vários widgets. Os widgets são elementos de interação com a página, como: botões, quadro de lista dropdown, checkboxes etc. Frequentemente esses elementos são inseridos em conjunto com rótulos (labels) para correta experiência do usuário. Formulários são usados também nas aplicações corporativas, as ferramentas RAD (Rapid Application Development) permitem a criação rápidas desses componentes containers e facilitam a inclusão dos diversos componentes sobre eles.



Contato:

Fone: 60 6070-6070

dev@gmail.com

Faça as alterações adicionais se necessário.

::fim