

# Tutorial Prof. Karize Viecelli



## Bootstrap



## Visual Code

Um guia bem direto e passo a passo para organizar o projeto no **Visual Studio Code**. 

---

### Passo a Passo: Criando a Estrutura do Projeto no VS Code

#### 1. Abrir o VS Code

- Clique no ícone do **Visual Studio Code** (VS Code) no seu computador.
  - Se não tiver o atalho, procure por "Visual Studio Code" no menu iniciar.
- 

#### 2. Selecionar a pasta do projeto (utilize uma pasta que já exista ou crie uma com nome Projeto)

- No menu superior, clique em **File** (Arquivo) → **Open Folder** (Abrir Pasta).
  - Escolha ou crie uma pasta nova para o projeto, por exemplo:  
 meu\_projeto.
- 

#### 3. Criar as pastas dentro do projeto

- No **Explorer** (painel lateral esquerdo do VS Code), clique com o botão direito sobre a pasta principal e selecione **New Folder** (Nova Pasta).
  - Crie as seguintes pastas:
    - css → vai guardar os estilos do site.
    - img → vai guardar as imagens.
- 

#### 4. Criar o arquivo style.css

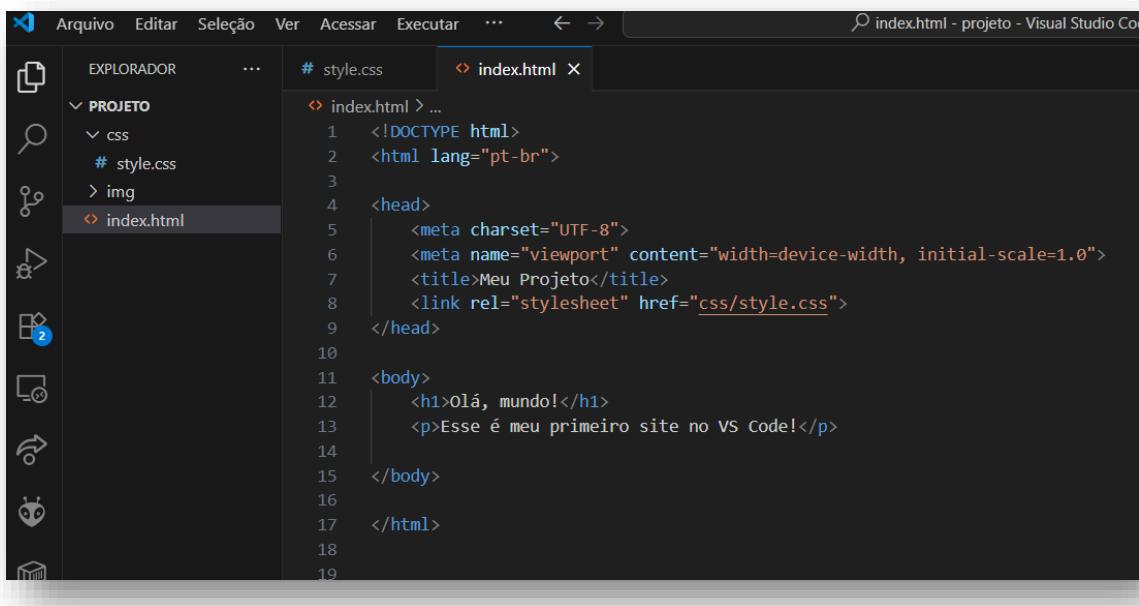
- Clique com o botão direito na pasta **css** → escolha **New File**.
  - Nomeie o arquivo como:  
 style.css.
- 

#### 5. Criar o arquivo principal index.html

- Clique com o botão direito na pasta principal do projeto → escolha **New File** (novo arquivo).
  - Nomeie o arquivo como:  
 index.html.
- 

#### 6. Estrutura básica do HTML

No index.html, digite ponto de exclamação e aperte o **enter**, deve gerar o código inicial e já faça o link com o style.css:



The screenshot shows the Visual Studio Code interface. The left sidebar has icons for Explorer, Search, Open, and others. The 'EXPLORADOR' section shows a 'PROJETO' folder containing 'css' (with 'style.css'), 'img', and 'index.html'. The 'index.html' file is selected and open in the main editor area. The code in 'index.html' is:

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Meu Projeto</title>
    <link rel="stylesheet" href="css/style.css">
</head>
<body>
    <h1>Olá, mundo!</h1>
    <p>Esse é meu primeiro site no VS Code!</p>
</body>
</html>
```

## 7. Testar no navegador

- Clique com o botão direito no arquivo index.html.
- Se tiver a extensão **Live Server** instalada, escolha **Open with Live Server**.
- Caso não tenha, abra a pasta do projeto no seu computador, dê duplo clique em index.html e ele abrirá no navegador.



👉 Pronto! Agora você tem a estrutura básica organizada:  
📁 meu\_projeto

📁 css → style.css

📁 img

📄 index.html

---

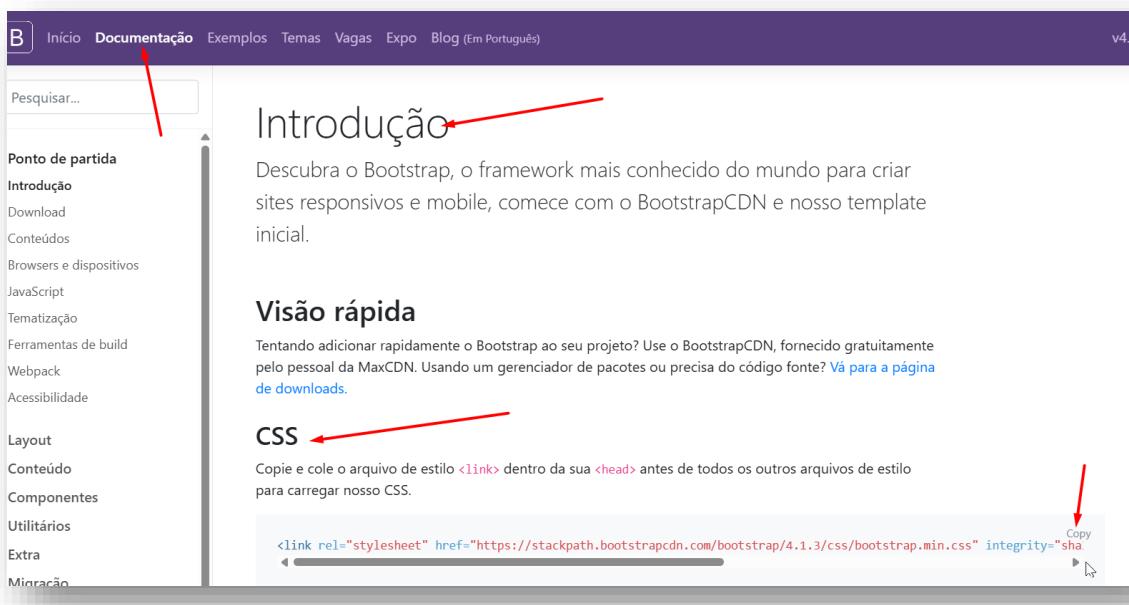
Adicionando Bootstrap:

### 💡 Passo a Passo: Usando o Bootstrap no Projeto

#### 1. Acessar o site oficial

- Entre no site: <https://getbootstrap.com.br/>
- Clique em **Documentação** -> **Introdução** -> **CSS**.
- Copie o link do arquivo CSS que aparece (é um CDN — Content Delivery Network).

Exemplo do link atual (pode mudar conforme versão):



#### 2. Inserir no index.html

No arquivo **index.html**, dentro da tag `<head>`, cole o link acima **antes do seu próprio style.css**. Assim, o Bootstrap carrega primeiro, e depois o aluno pode personalizar com o CSS próprio.

Exemplo completo com comentários:

```

# style.css          index.html
index.html > html > head > link
1  <!DOCTYPE html>
2  <html lang="pt-br">
3
4  <head>
5      <!-- Define o conjunto de caracteres (acentos, ç, etc.) --&gt;
6      &lt;meta charset="UTF-8"&gt;
7
8      <!-- Define a responsividade para dispositivos móveis --&gt;
9      &lt;meta name="viewport" content="width=device-width, initial-scale=1.0"&gt;
10
11     &lt;!-- Título da página --&gt;
12     &lt;title&gt;Meu Projeto&lt;/title&gt;
13
14     &lt;!-- Importando o CSS do Bootstrap via CDN --&gt;
15     &lt;link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/css/bootstrap.min.css"
16         integrity="sha384-MCw98/SFnGE8fJT3GXwEOngsV7Zt7NXFoaoApmYm81iuXoPkFOJwJ8ERdknPMO" crossorigin="anonymous"&gt;
17
18     &lt;!-- Link para o CSS próprio style.css - CSS externo --&gt;
19     &lt;link rel="stylesheet" href="css/style.css"&gt;
20
21 &lt;/head&gt;
</pre>

```

No arquivo **index.html**, antes de fechar a tag **</body>**, cole o link do Javascript do Bootstrap.

Assim, a página terá um script de Bootstrap mas pode também personalizar o seu próprio.

Exemplo completo com comentários:

**Ponto de partida**

- Introdução
- Download
- Conteúdos
- Browsers e dispositivos
- JavaScript
- Tematização
- Ferramentas de build
- Webpack
- Acessibilidade

**CSS**

Copie e cole o arquivo de estilo `<link>` dentro da sua `<head>` antes de todos os outros arquivos de estilo para carregar nosso CSS.

```
<link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/css/bootstrap.min.css" integrity="sha384-MCw98/SFnGE8fJT3GXwEOngsV7Zt7NXFoaoApmYm81iuXoPkFOJwJ8ERdknPMO" crossorigin="anonymous" Copy
```

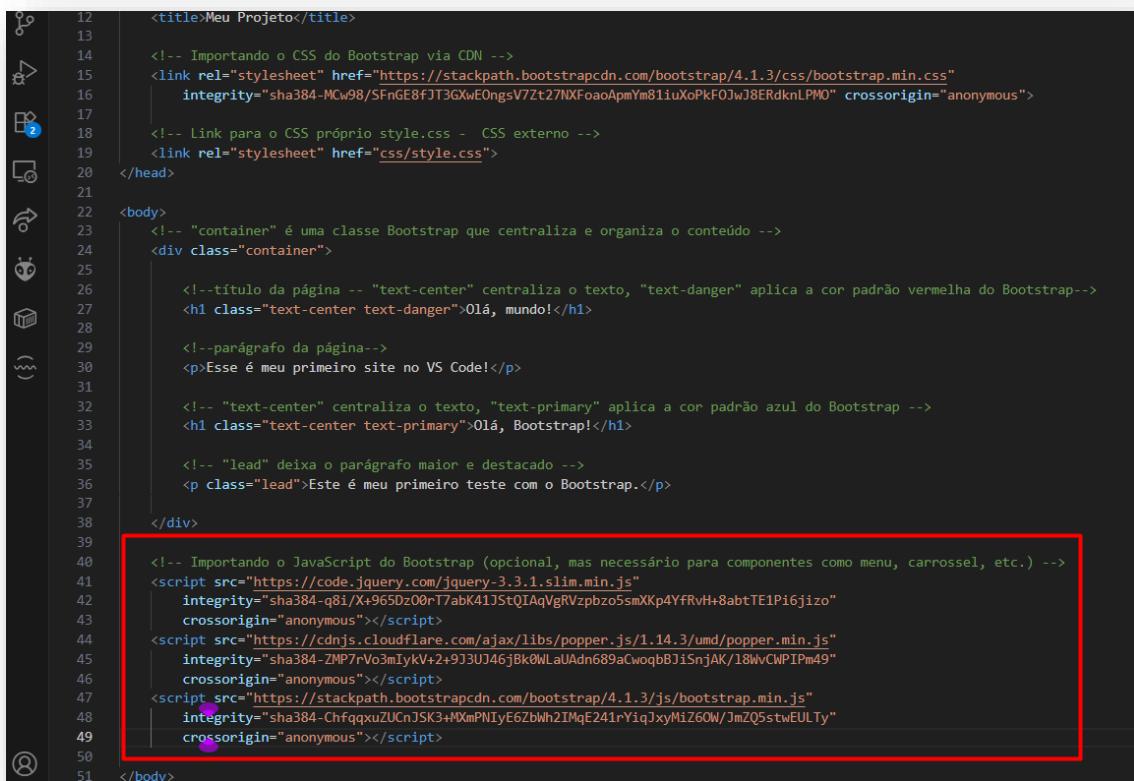
**JS**

Muitos dos nossos componentes precisam de JavaScript para funcionar. Mais especificamente, eles precisam do `jQuery`, `Popper.js` e de nossos próprios plugins JavaScript. Coloque os seguintes `<script>`s perto do final da sua página, logo antes do fechamento da tag `</body>`. jQuery tem que vir antes, depois o Popper.js e só depois nossos plugins JavaScript.

Nós usamos a `build slim do jQuery`, mas a versão completa também é suportada.

```
<script src="https://code.jquery.com/jquery-3.3.1.slim.min.js" integrity="sha384-q8i/X+965Dz0rT7abK41JStQIAqVgRVzbpbz" Copy
<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.3/umd/popper.min.js" integrity="sha384-ZMP7rVo3mIyyl" Copy
<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/js/bootstrap.min.js" integrity="sha384-ChfqxuzUCnJSK" Copy
```

Exemplo completo com comentários:



```
12 <title>Meu Projeto</title>
13
14 <!-- Importando o CSS do Bootstrap via CDN -->
15 <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/css/bootstrap.min.css"
16     integrity="sha384-MCw98/SFnGE8fJT3GXwEOngsV7zt27NXFaoApmYm8liuXoPkFOJw8ERdknPMO" crossorigin="anonymous">
17
18 <!-- Link para o CSS próprio style.css - CSS externo -->
19 <link rel="stylesheet" href="css/style.css">
20 </head>
21
22 <body>
23     <!-- "container" é uma classe Bootstrap que centraliza e organiza o conteúdo -->
24     <div class="container">
25
26         <!-- Título da página -- "text-center" centraliza o texto, "text-danger" aplica a cor padrão vermelha do Bootstrap-->
27         <h1 class="text-center text-danger">Olá, mundo!</h1>
28
29         <!-- Parágrafo da página -->
30         <p>Esse é meu primeiro site no VS Code!</p>
31
32         <!-- "text-center" centraliza o texto, "text-primary" aplica a cor padrão azul do Bootstrap -->
33         <h1 class="text-center text-primary">Olá, Bootstrap!</h1>
34
35         <!-- "lead" deixa o parágrafo maior e destacado -->
36         <p class="lead">Este é meu primeiro teste com o Bootstrap.</p>
37
38     </div>
39
40     <!-- Importando o JavaScript do Bootstrap (opcional, mas necessário para componentes como menu, carrossel, etc.) -->
41     <script src="https://code.jquery.com/jquery-3.3.1.slim.min.js"
42         integrity="sha384-q8i/Xi96DzO0rTTabK41JS0tQIAqVgRVzbzoSmXKp4YFRvH+8abTE1P16jizo"
43         crossorigin="anonymous"></script>
44     <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.3/umd/popper.min.js"
45         integrity="sha384-ZMP7rVo3IykVv2+913UJ46jbk0WLaUAdn689aCwoqbBJiSnjAK/l8WvCWIPm49"
46         crossorigin="anonymous"></script>
47     <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/js/bootstrap.min.js"
48         integrity="sha384-ChfquxUZCnJSK3+MXmPNyE6ZbWh2IMqE241rYiqJxyMz6OW/JmZQ5stwEULTy"
49         crossorigin="anonymous"></script>
50
51 </body>
```

### 3. O que está acontecendo em cada parte?

- **<meta charset="UTF-8">** → garante que acentos e caracteres especiais apareçam corretamente.
- **<meta name="viewport"...>** → faz o site se adaptar a telas de celular/tablet.
- **<link href="...bootstrap.min.css">** → importa o **CSS pronto do Bootstrap**, que traz estilos para botões, textos, tabelas, etc.
- **<link rel="stylesheet" href="css/style.css">** → adiciona o **CSS próprio do aluno**, para personalizar além do Bootstrap.
- **<div class="container">** → cria um espaço organizado e centralizado na tela.
- **class="text-center text-primary"** → aplica classes Bootstrap: centralizar texto e deixar azul.
- **<script src="...bootstrap.bundle.min.js">** → importa o **JavaScript do Bootstrap**, necessário para componentes interativos (menus, carrosséis, etc.).

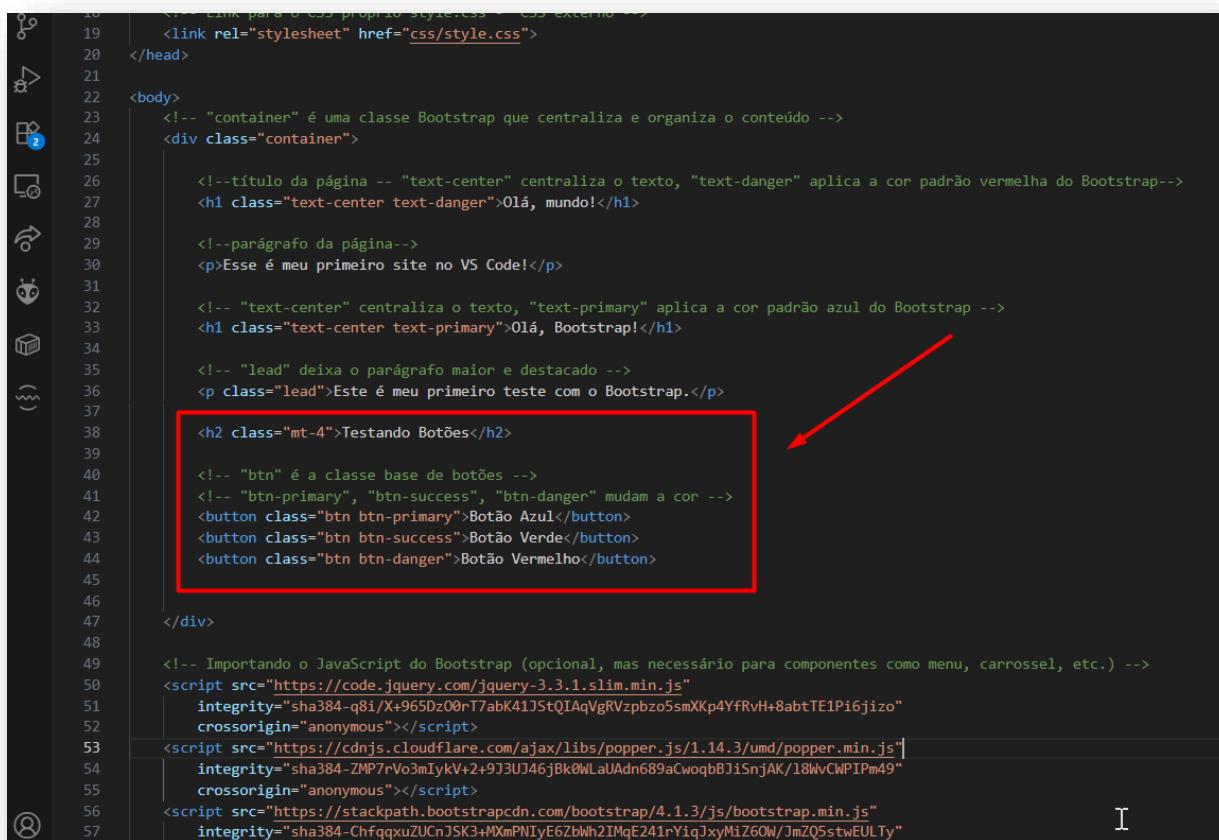
## Atividade Prática: Explorando o Bootstrap

### Objetivo

Testar as primeiras classes do Bootstrap (botões, cores de texto e grid responsivo) para entender como o framework funciona.

### 1. Criando botões estilizados

Dentro do `<div class="container">` no seu `index.html`, adicione:



```
16     <!-- Link para o CSS próprio style.css --> CSS externo -->
17     <link rel="stylesheet" href="css/style.css">
18 
19 </head>
20 
21 <body>
22     <!-- "container" é uma classe Bootstrap que centraliza e organiza o conteúdo -->
23     <div class="container">
24 
25         <!--título da página -- "text-center" centraliza o texto, "text-danger" aplica a cor padrão vermelha do Bootstrap-->
26         <h1 class="text-center text-danger">Olá, mundo!</h1>
27 
28         <!--parágrafo da página-->
29         <p>Esse é meu primeiro site no VS Code!</p>
30 
31         <!-- "text-center" centraliza o texto, "text-primary" aplica a cor padrão azul do Bootstrap -->
32         <h1 class="text-center text-primary">Olá, Bootstrap!</h1>
33 
34         <!-- "lead" deixa o parágrafo maior e destacado -->
35         <p class="lead">Este é meu primeiro teste com o Bootstrap.</p>
36 
37         <h2 class="mt-4">Testando Botões</h2>
38 
39         <!-- "btn" é a classe base de botões -->
40         <!-- "btn-primary", "btn-success", "btn-danger" mudam a cor -->
41         <button class="btn btn-primary">Botão Azul</button>
42         <button class="btn btn-success">Botão Verde</button>
43         <button class="btn btn-danger">Botão Vermelho</button>
44 
45 
46     </div>
47 
48     <!-- Importando o JavaScript do Bootstrap (opcional, mas necessário para componentes como menu, carrossel, etc.) -->
49     <script src="https://code.jquery.com/jquery-3.1.1.slim.min.js"
50            integrity="sha384-Zq8i/X+965Dz0rT7abK4JStQIAqVgRVzbzo5smXk4YfRvH+8abTE1Pi6jizo"
51            crossorigin="anonymous"></script>
52 
53     <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.3/umd/popper.min.js"
54            integrity="sha384-ZMP7rVo3MykV+2+9J3UJ46jBk0WLauAdn689aCwoqbBJiSnjAK/l8WvCPPIPm49"
55            crossorigin="anonymous"></script>
56 
57     <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/js/bootstrap.min.js"
58            integrity="sha384-Zq8i/X+965Dz0rT7abK4JStQIAqVgRVzbzo5smXk4YfRvH+8abTE1Pi6jizo"
59            crossorigin="anonymous"></script>
```

Ao aplicar a classe **btn + cor** (**btn-primary**, **btn-success**, etc.) o botão já fica estilizado.

## 2. Testando cores de texto

Logo abaixo, adicione:

```
<h1 class="text-center text-primary">Olá, Bootstrap!</h1>

<!-- "lead" deixa o parágrafo maior e destacado -->
<p class="lead">Este é meu primeiro teste com o Bootstrap.</p>

<h2 class="mt-4">Testando Botões</h2>

<!-- "btn" é a classe base de botões -->
<!-- "btn-primary", "btn-success", "btn-danger" mudam a cor -->
<button class="btn btn-primary">Botão Azul</button>
<button class="btn btn-success">Botão Verde</button>
<button class="btn btn-danger">Botão Vermelho</button>

<h2 class="mt-4">Testando Cores de Texto</h2>
<p class="text-primary">Texto azul (primary)</p>
<p class="text-success">Texto verde (success)</p>
<p class="text-danger">Texto vermelho (danger)</p>
<p class="text-muted">Texto cinza apagado (muted)</p>
<p class="text-warning">Texto amarelo (warning)</p>
<p class="text-info">Texto azul claro (info)</p>

</div>

<!-- Importando o JavaScript do Bootstrap (opcional, mas necessário para componentes como menu, carrosel, etc) -->
<script src="https://code.jquery.com/jquery-3.3.1.slim.min.js"
        integrity="sha256-q8i/X+95PnLjUdYXZ3QF9qaiFkzjVJXwA0PZUkxKuZC"
```

### Resultado

**Olá, mundo!**  
Esse é meu primeiro site no VS Code!

**Olá, Bootstrap!**  
Este é meu primeiro teste com o Bootstrap.

**Testando Botões**

Botão Azul    Botão Verde    Botão Vermelho

**Testando Cores de Texto**

- Texto azul (primary)
- Texto verde (success)
- Texto vermelho (danger)
- Texto cinza apagado (muted)
- Texto amarelo (warning)
- Texto azul claro (info)

Não

precisa criar estilos no CSS: o Bootstrap já traz uma paleta pronta.

### 3. Criando um grid responsivo

Agora, vamos organizar as colunas. Adicione:

```
<h2 class="mt-4">Testando Cores de Texto</h2>
<p class="text-primary">Texto azul (primary)</p>
<p class="text-success">Texto verde (success)</p>
<p class="text-danger">Texto vermelho (danger)</p>
<p class="text-muted">Texto cinza apagado (muted)</p>
<p class="text-warning">Texto amarelo (warning)</p>
<p class="text-info">Texto azul claro (info)</p>

<h2 class="mt-4">Testando Grid Responsivo</h2>
<div class="row">
    <!-- Cada div com col-4 ocupa 4 colunas de um total de 12 -->
    <div class="col-4 bg-primary text-white p-3">Coluna 1</div>
    <div class="col-4 bg-success text-white p-3">Coluna 2</div>
    <div class="col-4 bg-danger text-white p-3">Coluna 3</div>
</div>

</div>

<!-- Importando o JavaScript do Bootstrap (opcional, mas necessário para o menu, carrossel, etc.) -->
<script src="https://code.jquery.com/jquery-3.3.1.slim.min.js"
       integrity="sha384-"/>
/X+965Dz00rT7ahK411St0TAqVgRVzphzo5smXKn4YfRvH+8abTTF1Pi6iizo"
```

👉 Explicação:

- **row** cria uma linha.
- **col-4** divide a linha em 3 partes iguais (pois  $4+4+4 = 12$  colunas).
- As cores (bg-primary, bg-success, bg-danger) são cores de fundo prontas.
- **p-3** adiciona espaço interno (padding) em todos os lados.

---

### 4. O que observar

- Ao **diminuir a tela**, o grid irá se adaptar.
- Ao **trocar as classes de botões e cores**, existe uma lógica de nomes padronizados.
- Tudo isso **sem escrever nenhuma linha de CSS no style.css** ainda.

---

👉 Sugestão de desafio extra:

**Alterem uma cor ou mudem o tamanho das colunas** (por exemplo, col-6 + col-6, ou col-2 + col-5 + col-1+col-4), e observarem o resultado. (o total de colunas é 12, não pode ultrapassar este valor na soma total de colunas em uma div)

 **Conclusão**

Com apenas algumas linhas de código, você:

- ✓ Importou o Bootstrap.
- ✓ Criou botões prontos e responsivos.
- ✓ Usou cores pré-definidas.
- ✓ Montou um layout em grid.

Como organizar a página com Bootstrap, usando **container**, **container-fluid** e o sistema de organização de elementos.

---

### Conceito: Container x Container-Fluid

#### ◆ **container**

- Cria uma **largura fixa e centralizada** para o conteúdo.
- Essa largura se adapta conforme o tamanho da tela (mobile, tablet, desktop), mas nunca ocupa 100% da largura.
- Ideal quando você quer que o conteúdo **fique centralizado** e com margens laterais.

Exemplo:

```
<div class="container">
    <h1>Exemplo com Container</h1>
    <p>Este conteúdo fica centralizado na tela com margens laterais.</p>
</div>
```

---

#### ◆ **container-fluid**

- O conteúdo ocupa **100% da largura da tela**, de ponta a ponta.
- Útil quando você quer **aproveitar todo o espaço disponível** (ex.: cabeçalhos, rodapés, banners grandes).

Exemplo:

```
<div class="container-fluid bg-dark text-white p-3">
    <h1>Exemplo com Container-Fluid</h1>
    <p>Este conteúdo ocupa toda a largura da tela.</p>
</div>
```

---

### Organizando Elementos com Grid

O Bootstrap organiza a página em **linhas (row)** e **colunas (col)**.

- Uma linha (row) é dividida em **12 colunas**.
- Você pode distribuir os elementos conforme precisar:
  - col-12 → ocupa a linha inteira.
  - col-6 + col-6 → duas colunas iguais.
  - col-4 + col-8 → divisão desigual.

## Exemplo prático completo

```

16  <!-- Container fixo -->
17  <div class="container mt-4">
18      <h2 class="text-center">Exemplo com Container</h2>
19      <div class="row">
20          <div class="col-6 bg-primary text-white p-3">Coluna 1 (metade da tela)</div>
21          <div class="col-6 bg-success text-white p-3">Coluna 2 (metade da tela)</div>
22      </div>
23  </div>
24
25
26  <!-- Container-Fluid -->
27  <div class="container-fluid bg-dark text-white mt-4 p-4">
28      <h2>Exemplo com Container-Fluid</h2>
29      <div class="row">
30          <div class="col-4 bg-info p-3">Coluna 1</div>
31          <div class="col-4 bg-warning p-3">Coluna 2</div>
32          <div class="col-4 bg-danger p-3">Coluna 3</div>
33      </div>
34  </div>

```

## Resultado

### Exemplo com Container

Coluna 1 (metade da tela) Coluna 2 (metade da tela)

### Exemplo com Container-Fluid

Coluna 1 Coluna 2 Coluna 3

## ⭐ Explicação do código

- **container** → deixa o conteúdo centralizado, com margens automáticas.
- **container-fluid** → ocupa toda a largura da tela.
- **row** → cria uma linha para organizar colunas.
- **col-6** → divide a linha em duas colunas de mesmo tamanho ( $6+6 = 12$ ).
- **col-4** → divide a linha em três colunas iguais ( $4+4+4 = 12$ ).
- **bg-primary, bg-success, bg-danger...** → classes de cor de fundo do Bootstrap.
- **text-white** → cor do texto.
- **p-3 ou p-4** → espaçamento interno (padding).

## 🏆 Conclusão

- ✓ container → largura fixa e centralizada.
- ✓ container-fluid → ocupa toda a tela.
- ✓ row → cria uma linha.
- ✓ col → divide a linha em colunas proporcionais.

## Tipografia e espaçamentos no Bootstrap.

### Ficha de Exercício – Tipografia e Espaçamentos no Bootstrap

#### Objetivo

Explorar os recursos de **tipografia (títulos e textos)** e **espaçamentos (margens e paddings)** no Bootstrap. Você aprenderá como personalizar o visual sem escrever CSS.

#### ◆ Passo 1 – Testando títulos e textos

Adicione dentro do seu index.html:

```
<!-- Tipografia -->
<div class="container mt-4">

    <!-- Títulos e Parágrafos -->
    <h1 class="text-center">Título Principal (h1)</h1>

    <h2>Título Secundário (h2)</h2>
    <h3>Título Terciário (h3)</h3>

    <p>Este é um parágrafo normal.</p>
    <p class="lead">Este é um parágrafo de destaque usando a classe <code>.lead</code>.</p>
    <p class="text-muted">Este é um texto com estilo apagado (muted).</p>
</div>
```

#### Checklist:

- O título h1 aparece maior que os outros.
- O parágrafo com lead tem destaque.
- O texto com text-muted aparece mais claro.

## Título Principal (h1)

### Título Secundário (h2)

### Título Terciário (h3)

Este é um parágrafo normal.

Este é um parágrafo de destaque usando a classe `.lead`.

Este é um texto com estilo apagado (muted).

◆ Passo 2 – Trabalhando com alinhamento

Adicione:

```
<!-- Alinhamento de Texto -->
<div class="container mt-4">
    <h2 class="text-center">Alinhamento de Texto</h2>
    <p class="text-left">Texto alinhado à esquerda.</p>
    <p class="text-center">Texto centralizado.</p>
    <p class="text-right">Texto alinhado à direita.</p>
    <p class="text-justify">Ambitioni dedisse scrispsisse iudicaretur. Cras mattis iudicium purus sit amet fermentum. Donec sed odio operae, eu vulputate felis rhoncus. Praeterea iter est quasdam res quas ex communi. At nos hinc posthac, sitientis piros Afros. Petierunt uti sibi concilium totius Galliae in diem certam indicere. Cras mattis iudicium purus sit amet fermentum.</p>
</div>
```

Checklist:

- O texto centralizado fica no meio da tela.
- O alinhamento muda conforme a classe.

## Alinhamento de Texto

Texto alinhado à esquerda.

Texto centralizado.

Texto alinhado à direita.

Ambitioni dedisse scrispsisse iudicaretur. Cras mattis iudicium purus sit amet fermentum. Donec sed odio operae, eu vulputate felis rhoncus. Praeterea iter est quasdam res quas ex communi. At nos hinc posthac, sitientis piros Afros. Petierunt uti sibi concilium totius Galliae in diem certam indicere. Cras mattis iudicium purus sit amet fermentum.

◆ Passo 3 – Testando margens e paddings

As classes de espaçamento seguem o padrão:

👉 m = margin | p = padding | números de 0 a 5.

Exemplo prático:

```
<!-- Espaçamento -->
<div class="container mt-4">
    <div class="bg-primary text-white p-3 mb-3">Caixa com padding (p-3) e margem inferior (mb-3)</div>
    <div class="bg-success text-white p-5 mt-4">Caixa com padding (p-5) e margem superior (mt-4)</div>
</div>
```

Checklist:

- A primeira caixa tem menos espaço interno que a segunda.
- A margem superior/inferior afasta os elementos corretamente.

Caixa com padding (p-3) e margem inferior (mb-3)

Caixa com padding (p-5) e margem superior (mt-4)

#### ◆ Passo 4 – Espessura e italização de fontes

Rapidamente, modifique a espessura do texto ou italicize-o.

Adicione:

```
<!-- Peso e Estilo de Fonte -->
<p class="font-weight-bold">Texto grosso.</p>
<p class="font-weight-normal">Texto de espessura comum.</p>
<p class="font-weight-light">Texto de espessura leve.</p>
<p class="font-italic">Texto itálico.</p>
```

#### Resultado

**Texto grosso.**

Texto de espessura comum.

Texto de espessura leve.

*Texto itálico.*

#### ◆ Desafios Extras

1. Crie três parágrafos: um centralizado, um em itálico e um em negrito.
2. Use mt-5 e mb-5 para criar um espaço grande entre dois elementos.
3. Combine text-center com fw-bold para criar um **título centralizado em negrito**.

### Conclusão

- ✓ Bootstrap oferece classes prontas para **tipografia e alinhamento**.
  - ✓ Espaçamentos (m e p) ajudam a organizar os elementos.
  - ✓ Tudo isso **sem precisar escrever CSS manualmente**.
-

 **Ficha de Exercício – Componentes Básicos do Bootstrap** **Objetivo**

Explorar os **componentes prontos do Bootstrap**: botões avançados, alertas e cards.  
Entender como esses elementos funcionam e como usá-los na organização da página.

---

**◆ Passo 1 – Botões avançados**

Adicione dentro do container:

```
<h2 class="mt-4">Botões Avançados</h2>
<button class="btn btn-outline-primary mb-2">Botão Contornado Azul</button>
<button class="btn btn-lg btn-success mb-2">Botão Grande Verde</button>
<button class="btn btn-sm btn-danger mb-2">Botão Pequeno Vermelho</button>
<button class="btn btn-block btn-info">Botão Largura Total Azul Claro</button>
```

 **Checklist:**

- O botão azul tem só a borda colorida.
- O botão verde é maior que os outros.
- O botão vermelho é bem pequeno.

**Botões Avançados**[Botão Contornado Azul](#)[Botão Grande Verde](#)[Botão Pequeno Vermelho](#)[Botão Largura Total Azul Claro](#)

## ◆ Passo 2 – Alertas

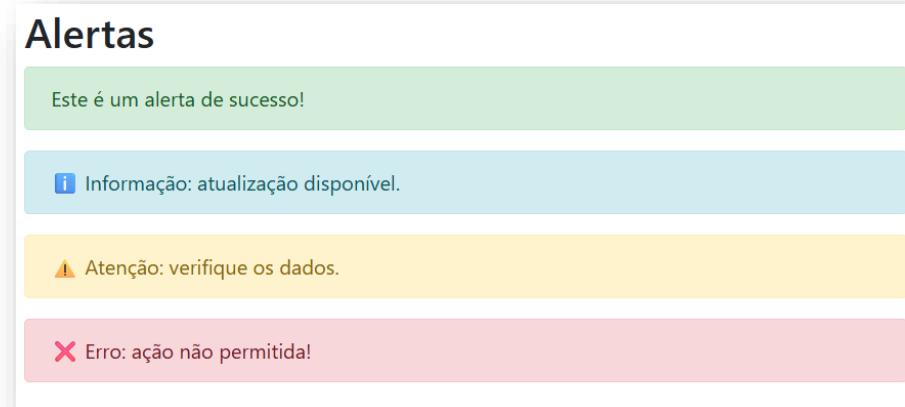
Os alertas servem para mensagens de destaque (avisos, erros, confirmações).

```
<h2 class="mt-4">Alertas</h2>
<div class="alert alert-success" role="alert">
| Este é um alerta de sucesso!
</div>

<div class="alert alert-info" role="alert">
| i Informação: atualização disponível.
</div>

<div class="alert alert-warning" role="alert">
| ⚠ Atenção: verifique os dados.
</div>
<div class="alert alert-danger" role="alert">
| ✖ Erro: ação não permitida!
</div>
```

## Alertas



### Checklist:

- O alerta verde aparece como confirmação.
- O alerta azul aparece como informação.
- O alerta amarelo aparece como aviso.
- O alerta vermelho aparece como erro.

### ◆ Passo 3 – Cards

Os cards são caixinhas prontas para organizar conteúdos (textos, imagens, links).

```
<h2 class="mt-4">Cards</h2>
<div class="row">
  <div class="col-4">
    <div class="card">
      
      <div class="card-body">
        <h5 class="card-title">Card 1</h5>
        <p class="card-text">Este é um exemplo de card com imagem, título e texto.</p>
        <a href="#" class="btn btn-primary">Saiba mais</a>
      </div>
    </div>
  </div>

  <div class="col-4">
    <div class="card">
      <div class="card-body">
        <h5 class="card-title">Card 2</h5>
        <p class="card-text">Um card sem imagem, apenas com texto.</p>
        <a href="#" class="btn btn-success">Continuar</a>
      </div>
    </div>
  </div>
</div>
```

#### ✓ Checklist:

- O primeiro card exibe uma imagem no topo.
- Ambos têm título, texto e botão.
- Os cards estão organizados em colunas lado a lado.

## Cards



### Card 1

Este é um exemplo de card com imagem, título e texto.

[Saiba mais](#)

### Card 2

Um card sem imagem, apenas com texto.

[Continuar](#)

---

◆ Desafios Extras 

1. Adicione um card com **alerta dentro** (misture os componentes).
  2. Teste mudar os tamanhos dos botões (btn-sm, btn-lg).
  3. Substitua a imagem do card por uma de sua escolha dentro da pasta img.
- 

 Conclusão

- ✓ Aprendemos botões avançados (btn-outline, tamanhos diferentes).
  - ✓ Usamos alertas para mensagens visuais.
  - ✓ Criamos cards para organizar conteúdo de forma bonita e responsiva.
-

Navbar (menu de navegação) com Bootstrap. Esse é um dos componentes mais usados em sites.

## Ficha de Exercício – Navbar no Bootstrap

### Objetivo

Aprender a criar uma **barra de navegação responsiva** usando o Bootstrap, com links, logotipo e menu que se adapta ao celular (menu hamburguer).

#### ◆ Passo 1 – Criando a Navbar básica

Adicione uma nova página: **home.html**

Gere o código básico do **html** com os links do **bootstrap, o css e o js**.

```
1  <!DOCTYPE html>
2  <html lang="pt-br">
3
4  <head>
5      <meta charset="UTF-8">
6      <meta name="viewport" content="width=device-width, initial-scale=1.0">
7      <title>Home</title>
8      <!-- Importando o CSS do Bootstrap via CDN -->
9      <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/css/bootstrap.min.css"
10         integrity="sha384-MCw98/SFnGE8fJT3XW0OaApmYm8iuXoPkFOJwJ8ERdknLPM0" crossorigin="anonymous">
11  </head>
12
13 <body>
14
15     <!-- Importando o JavaScript do Bootstrap (opcional, mas necessário para componentes como menu, carrossel, etc.) -->
16     <script src="https://code.jquery.com/jquery-3.3.1.slim.min.js"
17         integrity="sha384-q8i/x+965D20rT7abK41JSQIAqVgRVzbzo5smXKp4YfRvH+8abTE1Pi6jizo"
18         crossorigin="anonymous"></script>
19     <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.3/umd/popper.min.js"
20         integrity="sha384-ZMP7rVo3miIykV+2+9J3UJ46jBk0WLauAdn689aCwoqbBJiSnjAK/l8WvCWPIPm49"
21         crossorigin="anonymous"></script>
22     <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/js/bootstrap.min.js"
23         integrity="sha384-ChfqxuZUCnJSK3+MXmPNiY62bWh2IMqE241rYiqJxyMiZ60W/JmZQ5stwEULTy"
24         crossorigin="anonymous"></script>
25
26 </body>
27
28 </html>
```

Adicione logo abaixo da tag <body> no index.html:

```
<!-- Navbar básica -->
<nav class="navbar navbar-expand-lg navbar-dark bg-dark">
  <div class="container-fluid">
    <!-- Logo ou nome do site -->
    <a class="navbar-brand" href="#">Meu Site</a>

    <!-- Botão que aparece no celular (menu hamburguer) -->
    <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#menuNavegacao">
      <span class="navbar-toggler-icon"></span>
    </button>

    <!-- Links do menu -->
    <div class="collapse navbar-collapse" id="menuNavegacao">
      <ul class="navbar-nav ms-auto">
        <li class="nav-item">
          <a class="nav-link active" href="#">Início</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="#">Sobre</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="#">Contato</a>
        </li>
      </ul>
    </div>
  </div>
</nav>
```

Meu Site   Início   Sobre   Contato

### Explicando o código

- **navbar navbar-expand-lg** → cria a barra e define que será expansível em telas grandes (lg).
- **navbar-dark bg-dark** → define o estilo de cores (texto claro sobre fundo escuro).

- **navbar-brand** → logo ou nome do site.
  - **navbar-toggler** → botão de menu que aparece em telas pequenas.
  - **collapse navbar-collapse** → esconde o menu no celular e mostra em telas maiores.
  - **ms-auto** → joga os itens para o lado direito.
- 

#### ◆ Passo 2 – Alterando o estilo da Navbar

Troque a cor do fundo para experimentar outras classes:

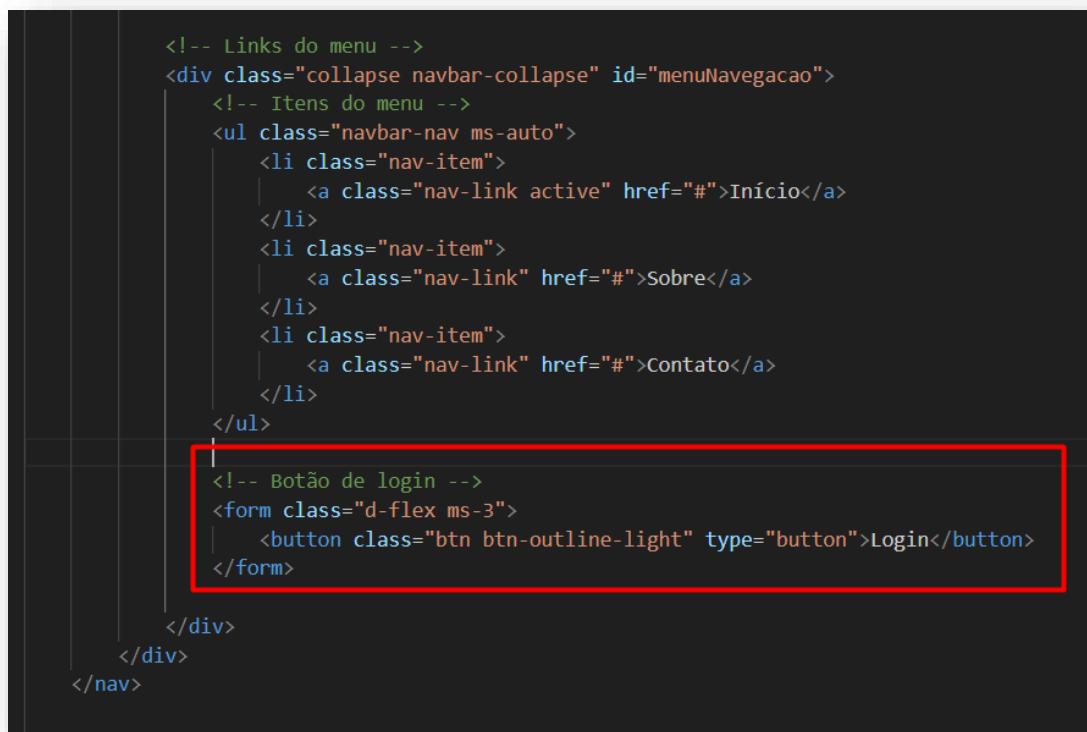
- bg-light navbar-light → fundo claro, texto escuro.
- bg-primary navbar-dark → fundo azul Bootstrap.
- bg-success navbar-dark → fundo verde.

 Checklist:

- O menu funciona em **telas grandes**.
  - O menu vira **hamburguer no celular**.
  - As cores mudam conforme a classe aplicada.
- 

#### ◆ Passo 3 – Adicionando um botão de ação

Dentro de <div class="collapse navbar-collapse">, adicione:



```
<!-- Links do menu -->
<div class="collapse navbar-collapse" id="menuNavegacao">
    <!-- Itens do menu -->
    <ul class="navbar-nav ms-auto">
        <li class="nav-item">
            <a class="nav-link active" href="#">Início</a>
        </li>
        <li class="nav-item">
            <a class="nav-link" href="#">Sobre</a>
        </li>
        <li class="nav-item">
            <a class="nav-link" href="#">Contato</a>
        </li>
    </ul>
    <!-- Botão de login -->
    <div class="d-flex ms-3">
        <form>
            <button class="btn btn-outline-light" type="button">Login</button>
        </form>
    </div>
</div>
</nav>
```

The code editor shows a snippet of HTML for a Bootstrap navigation bar. A red box highlights a section of the code where a new button is being added. The highlighted code is:

```
<!-- Botão de login -->
<div class="d-flex ms-3">
    <form>
        <button class="btn btn-outline-light" type="button">Login</button>
    </form>
</div>
```

## Checklist:

- O botão aparece no lado direito do menu.
  - O estilo btn-outline-light deixa o botão visível no fundo escuro.
- 

## Desafios Extras

1. Altere o texto "**Meu Site**" para o nome do seu projeto.
  2. Adicione mais um link no menu chamado **Serviços**.
  3. Troque o botão "Login" para a cor verde (btn-success).
  4. Faça o teste em tela de celular → o menu deve virar **hamburguer clicável**.
- 

## Conclusão

- ✓ Criamos uma **Navbar responsiva** com links.
  - ✓ Vimos como trocar **cores e estilos** facilmente.
  - ✓ Aprendemos a adicionar **botões de ação** dentro da barra.
-