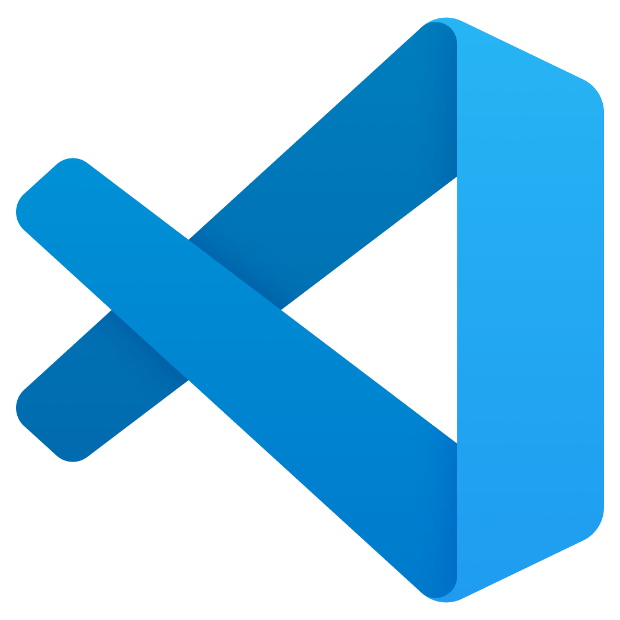
**Tutorial Prof. Karize Viecelli**



**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:

Texto

O conteúdo gerado por IA pode estar incorreto.

**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.

Interface gráfica do usuário, Texto, Aplicativo

O conteúdo gerado por IA pode estar incorreto.

👉 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):

Interface gráfica do usuário, Texto, Aplicativo, Email

O conteúdo gerado por IA pode estar incorreto.

**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:

Texto

O conteúdo gerado por IA pode estar incorreto.

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:

Interface gráfica do usuário, Texto, Aplicativo

O conteúdo gerado por IA pode estar incorreto.

Exemplo completo com comentários:

Texto

O conteúdo gerado por IA pode estar incorreto.

**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:

Texto

O conteúdo gerado por IA pode estar incorreto.

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

**2. Testando cores de texto**

Logo abaixo, adicione:

Texto

O conteúdo gerado por IA pode estar incorreto.

**Resultado**

Diagrama

O conteúdo gerado por IA pode estar incorreto. **Não precisa criar estilos no CSS**: o Bootstrap já traz uma paleta pronta.

**3. Criando um grid responsivo**

Agora, vamos organizar as colunas. Adicione:

Texto

O conteúdo gerado por IA pode estar incorreto.

👉 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**

Texto

O conteúdo gerado por IA pode estar incorreto.

**Resultado**

Uma imagem contendo Padrão do plano de fundo

O conteúdo gerado por IA pode estar incorreto.

**✨ 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:

Texto

O conteúdo gerado por IA pode estar incorreto.

✅ Checklist:

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

Interface gráfica do usuário, Texto, Aplicativo, Email

O conteúdo gerado por IA pode estar incorreto.

**🔹 Passo 2 – Trabalhando com alinhamento**

Adicione:

Texto

O conteúdo gerado por IA pode estar incorreto.

✅ Checklist:

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

Interface gráfica do usuário, Texto

O conteúdo gerado por IA pode estar incorreto.

**🔹 Passo 3 – Testando margens e paddings**

As classes de espaçamento seguem o padrão:  
👉 m = margin | p = padding | números de 0 a 5.

Exemplo prático:

Texto

O conteúdo gerado por IA pode estar incorreto.

✅ Checklist:

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

Uma imagem contendo Ícone

O conteúdo gerado por IA pode estar incorreto.

**🔹 Passo 4 – Espessura e italização de fontes**

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

Adicione:

Texto

O conteúdo gerado por IA pode estar incorreto.

**Resultado**

Interface gráfica do usuário, Texto, Aplicativo, chat ou mensagem de texto

O conteúdo gerado por IA pode estar incorreto.

**🔹 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:

Texto

O conteúdo gerado por IA pode estar incorreto.

✅ Checklist:

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

Uma imagem contendo Interface gráfica do usuário

O conteúdo gerado por IA pode estar incorreto.

**🔹 Passo 2 – Alertas**

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

Tela de computador com texto preto sobre fundo branco

O conteúdo gerado por IA pode estar incorreto.

Uma imagem contendo Tabela

O conteúdo gerado por IA pode estar incorreto.

✅ 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).

Texto

O conteúdo gerado por IA pode estar incorreto.

✅ Checklist:

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

Interface gráfica do usuário, Aplicativo

O conteúdo gerado por IA pode estar incorreto.

**🔹 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.**

Texto

O conteúdo gerado por IA pode estar incorreto.

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

Texto

O conteúdo gerado por IA pode estar incorreto.

Interface gráfica do usuário

O conteúdo gerado por IA pode estar incorreto.

**🔎 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:

Texto

O conteúdo gerado por IA pode estar incorreto.

✅ 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.