

**Professores**

Ana Beatriz Vilar e Philippe Correia

# **Dominando o Estilo da Web:**

## **De CSS a Bootstrap**



# O que é CSS?

- ✓ CSS: A Linguagem de Estilo da Web.
- ✓ CSS, ou Cascading Style Sheets (Folhas de Estilo em Cascata), é a **linguagem que usamos para dar estilo, cor, layout e vida às páginas da web.**
- ✓ Pense no HTML como o esqueleto de uma página e no CSS como as roupas, a maquiagem e a postura.

# Como o CSS funciona?

- ✓ A sintaxe do CSS é composta por:
  - ✓ **Seletor**: o elemento HTML que você deseja estilizar (ex: h1, p, .minha-classe).
  - ✓ **Declaração**: composta por uma propriedade e um valor, que definem o estilo a ser aplicado.
    - ✓ **Propriedade**: o que você quer mudar (ex: color, font-size, background-color).
    - ✓ **Valor**: como você quer mudar (ex: blue, 16px, #FFFFFF).

CSS

```
seletor {  
  propriedade: valor;  
}
```

# Três maneira de adicionar CSS a uma página

## **CSS Externo (recomendado):**

- ✓ Um arquivo .css separado, linkado no <head> do seu HTML.
- ✓ `<link rel="stylesheet" href="style.css">`
- ✓ Mantém o código organizado e reutilizável.

## **CSS Interno:**

- ✓ Código CSS dentro da tag <style> no <head> do HTML.
- ✓ Útil para estilos específicos de uma única página.

## **CSS Inline:**

- ✓ Estilos aplicados diretamente em uma tag HTML usando o atributo style.
- ✓ `<p style="color: red;">Este parágrafo é vermelho. </p>`
- ✓ Deve ser usado com moderação.

# Seletores Essenciais

## **SELETOR DE TIPO/TAG:**

- ✓ Seleciona todos os elementos de um determinado tipo.
- ✓ Ex: p, h2, div.

## **SELETOR DE CLASSE:**

- ✓ Seleciona elementos com um atributo de classe específico.
- ✓ Ex: .destaque.

## **SELETOR DE ID:**

- ✓ Seleciona um único elemento com um atributo de ID específico.
- ✓ Ex: #cabeçalho

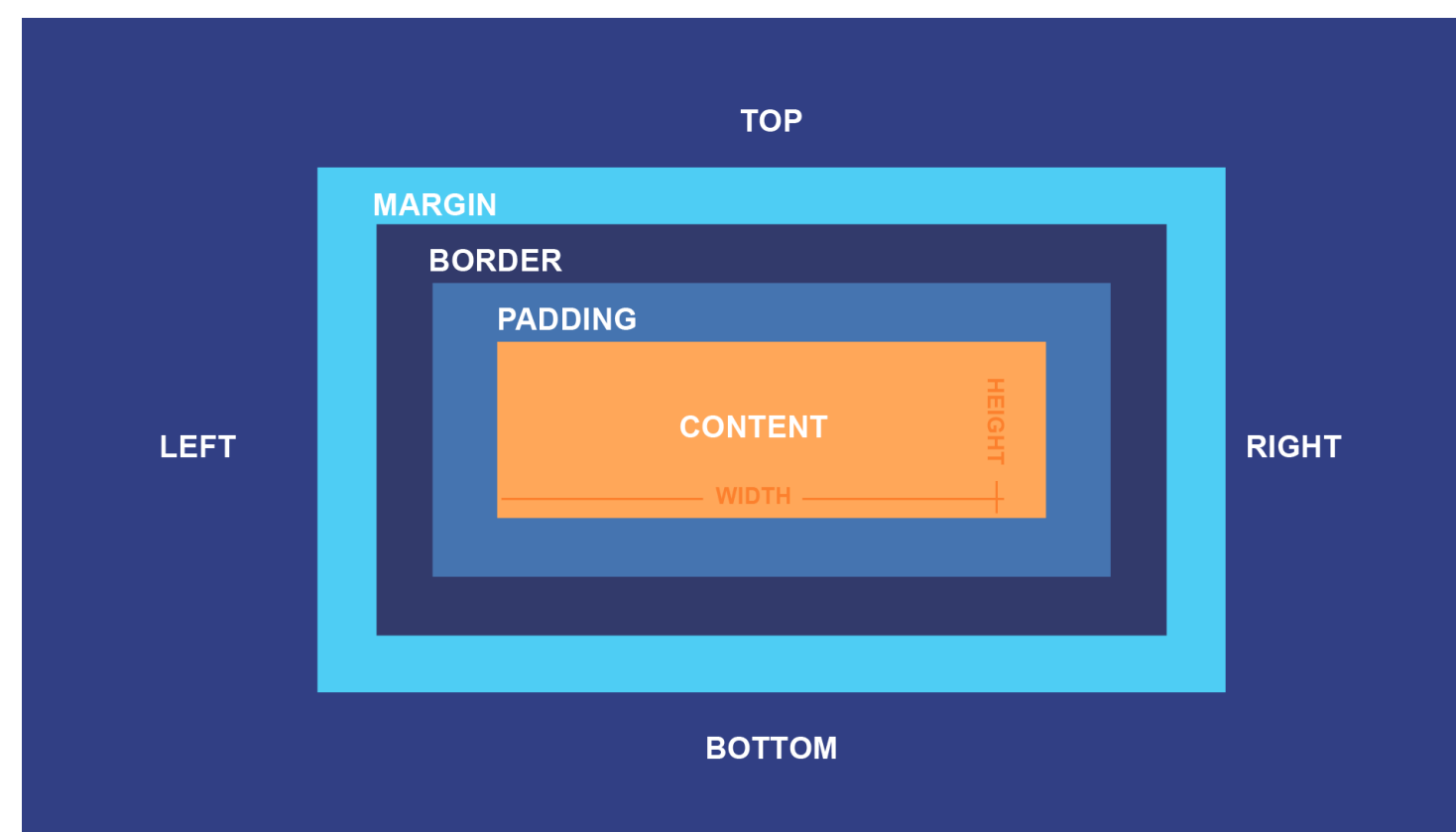
## **SELETOR DE ATRIBUTO:**

- ✓ Seleciona elementos com base em um atributo ou valor do atributo.
- ✓ Ex: [type="submit"].

# O modelo de caixa (Box Model)

## A base do Layout

- ✓ Todo elemento HTML é uma caixa retangular. O box model descreve como essa caixa é composta:
  - ✓ **Conteúdo:** o texto, imagem ou outro conteúdo.
  - ✓ **Padding:** espaçamento interno, entre o conteúdo e a borda.
  - ✓ **Borda (border):** a linha que envolve o padding.
  - ✓ **Margem (margin):** espaçamento externo, ao redor da borda.



# Propriedades Comuns para Começar

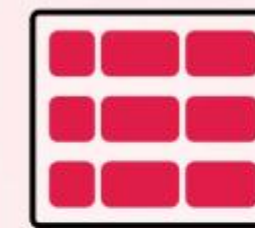
✓ <b>CORES E FUNDO:</b>	color	background-color	background-image	
✓ <b>TEXTO E FONTES:</b>	font-size	font-family	font-weight	text-align
✓ <b>ESPAÇAMENTOS:</b>	margin	padding		
✓ <b>DIMENSÕES:</b>	width	height		
✓ <b>BORDAS:</b>	border	border-radius		

# Layouts Modernos com Flexbox

- ✓ **FLEXBOX: Alinhamento e Distribuição Simplificados**
- ✓ Um modelo de layout unidimensional que facilita o alinhamento e a distribuição de itens em um contêiner.
- ✓ **Propriedades chave do contêiner:**
  - ✓ display: flex
  - ✓ justify-content
  - ✓ align-items
- ✓ **Propriedades chave dos itens:**
  - ✓ flex-grow
  - ✓ flex-shrink
  - ✓ flex-basis.

## CSS Flexbox

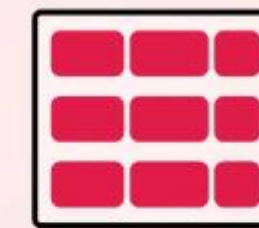
### flex-direction



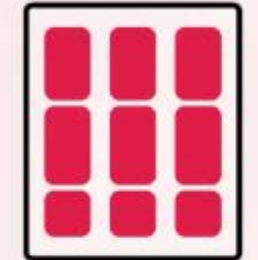
row



column



row-reverse

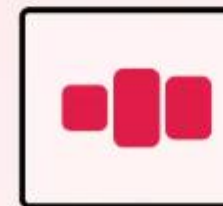


column-reverse

### align-items



flex-start



center



flex-end



stretch

### justify-content



flex-start



center



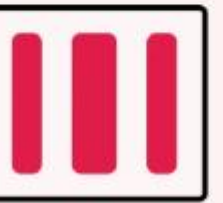
flex-end



space-between

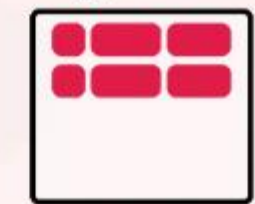


space-around

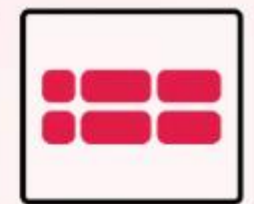


space-evenly

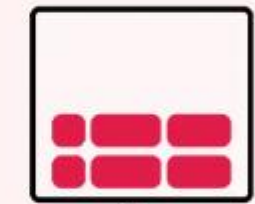
### align-content



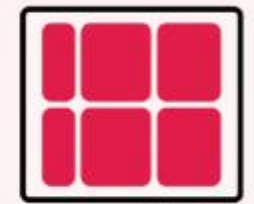
flex-start



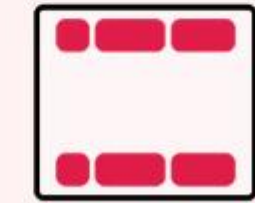
center



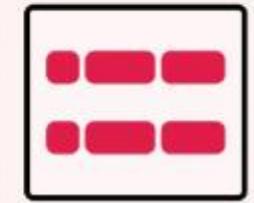
flex-end



stretch



space-between



space-around

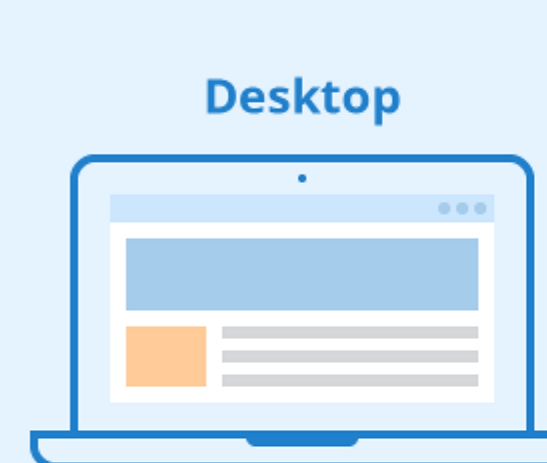


# Design Responsivo com Media Queries

- ✓ **Adaptando-se a qualquer tela**
- ✓ Media queries permite que você aplique diferentes estilos CSS para diferentes tamanhos de tela e dispositivos.
- ✓ Essencial para criar sites que funcionam bem em desktops, tablets e celulares.

## CSS

```
@media (max-width: 600px) {  
  body {  
    background-color: lightblue;  
  }  
}
```



@media screen and  
(min-width: 1024px)  
{...}



@media screen and  
(min-width: 768px) and  
(max-width: 1023px)  
{...}



@media screen and  
(max-width: 767px)  
{...}

# Acelerando o Desenvolvimento com Bootstrap

## Introdução ao Bootstrap

### O que é Bootstrap?

- ✓ Um framework front-end gratuito e de código aberto para desenvolver sites responsivos e mobile-first de forma rápida e fácil.
- ✓ Criado pelo Twitter, ele contém uma vasta coleção de componentes de interface pré-construídos e um poderoso sistema de grid.



# Acelerando o Desenvolvimento com Bootstrap

## Por que usar?

- ✓ **Agilidade no Desenvolvimento:** Componentes prontos para uso (menus, botões, formulários) que poupam tempo.
- ✓ **Responsividade Integrada:** O sistema de grid do Bootstrap facilita a criação de layouts que se adaptam a diferentes tamanhos de tela.
- ✓ **Consistência Visual:** Garante uma aparência coesa em todo o seu projeto.
- ✓ **Ampla Documentação e Comunidade:** Fácil de aprender e encontrar ajuda.
- ✓ **Compatibilidade entre Navegadores:** Minimiza problemas de compatibilidade.



# Acelerando o Desenvolvimento com Bootstrap

## Como adicionar Bootstrap ao seu Projeto?

- ✓ A Maneira Mais Fácil: Usando uma CDN (Content Delivery Network)
- ✓ Basta adicionar os links para os arquivos CSS e JavaScript do Bootstrap no seu arquivo HTML.

```
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css"  
      rel="stylesheet">
```

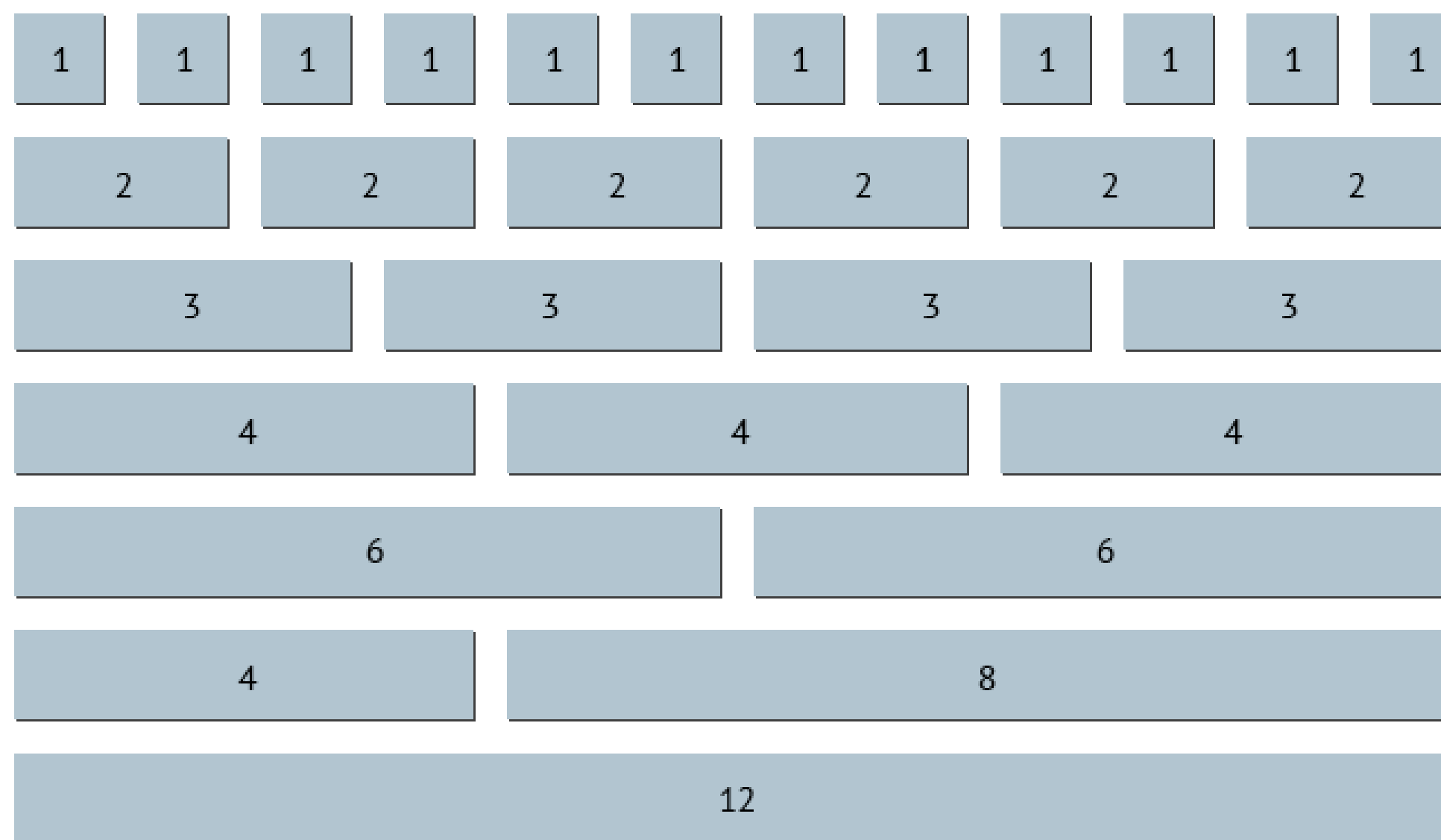
```
      <script  
src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></script>
```

# Acelerando o Desenvolvimento com Bootstrap

## O sistema de Grid do Bootstrap

### A BASE DE TUDO

- ✓ O Bootstrap utiliza um sistema de grid de 12 colunas para criar layouts.
- ✓ Você pode especificar quantas colunas um elemento deve ocupar em diferentes tamanhos de tela (breakpoints).
- ✓ Classes principais: `.container`, `.row`, `.col-*`.



# Acelerando o Desenvolvimento com Bootstrap

## Componentes Essenciais do Bootstrap

### Construindo sua Interface Rapidamente

- ✓ **Botões:** Vários estilos e tamanhos de botões pré-definidos.
- ✓ **Barras de Navegação (Navbars):** Menus de navegação responsivos e completos.
- ✓ **Cartões (Cards):** Contêineres flexíveis para exibir conteúdo.
- ✓ **Formulários:** Estilos e layouts para formulários.
- ✓ **Alertas e Modais:** Caixas de mensagem e janelas pop-up.
- ✓ Exemplo de um botão:  
`<button type="button" class="btn btn-primary">Botão Primário</button>`