

ENSINO MÉDIO INTEGRADO: INFORMÁTICA

Disciplina de Desenvolvimento Web

Aula 06: Framework Bootstrap - Sistema de Grade

Gil Eduardo de Andrade

Conceitos Preliminares

Framework Bootstrap (<https://getbootstrap.com/>)

O Bootstrap é uma ferramenta popular para o desenvolvimento de aplicações web, no que diz respeito ao front-end, oferecendo uma coleção abrangente de componentes e estilos CSS pré-construídos, além de funcionalidades JavaScript, que permitem a criação de interfaces de usuário responsivas e visualmente atraentes, de forma rápida e eficiente.

Criado originalmente no Twitter, em 2010, o Bootstrap foi projetado para unificar as ferramentas internas da empresa. Seu código aberto foi lançado em 2011, e rapidamente se tornou um dos frameworks front-end mais utilizados no mundo, dando suporte aos desenvolvedores na criação de sites e aplicações web consistentes e responsivas.

Principais Características e Funcionalidades

- **Sistema de Grid Responsivo**: utiliza um sistema de grid responsivo baseado em colunas, que se adapta automaticamente a diferentes tamanhos de tela, garantindo que o layout do site seja exibido corretamente em desktops, tablets e smartphones.
- **Componentes Reutilizáveis**: oferece uma ampla variedade de componentes pré-estilizados e personalizáveis, como botões, formulários, menus de navegação, alertas, tabelas e muito mais. Esses componentes podem ser facilmente incorporados aos projetos, economizando tempo e esforço no desenvolvimento.
- **Estilos CSS Globais**: fornece estilos CSS globais para elementos HTML básicos, como tipografia, cores, espaçamento e outros aspectos visuais. Isso garante uma aparência consistente e profissional em todo o site.

- **JavaScript Integrado**: além dos estilos CSS, também inclui plugins JavaScript que adicionam interatividade e funcionalidades avançadas aos componentes, como menus dropdown, modais, carrosséis e validação de formulários.

Vantagens da Utilização do Bootstrap

- **Agilidade no Desenvolvimento**: acelera o processo de desenvolvimento, fornecendo componentes e estilos pré-construídos que podem ser rapidamente integrados aos projetos.
 - **Consistência Visual**: garante uma aparência consistente em todo o site, seguindo um conjunto de estilos e diretrizes de design predefinidos.
 - **Responsividade**: facilita a criação de sites responsivos que se adaptam a diferentes tamanhos de tela, proporcionando uma experiência otimizada para todos os usuários.
 - **Comunidade Ativa**: possui uma comunidade de desenvolvedores ativa e engajada, que oferece suporte, compartilha recursos e contribui para o desenvolvimento contínuo do framework.
 - **Documentação Abrangente**: possui uma documentação completa e detalhada, que facilita o aprendizado e o uso do framework.
-

Redes de Distribuição de Conteúdo - CDNs

As CDNs (Content Delivery Networks), ou Redes de Distribuição de Conteúdo, são infraestruturas complexas de servidores estrategicamente localizados em diversos pontos do mundo. Seu principal objetivo é otimizar a entrega de conteúdo online, como páginas web, imagens, vídeos, aplicativos e outros tipos de dados, para usuários em diferentes locais geográficos.

Funcionamento

- **Armazenamento em cache**: armazenam cópias do conteúdo original em seus servidores, chamados de "servidores de borda" ou "pontos de

presença" (PoPs). Esses servidores estão mais próximos dos usuários do que os servidores de origem, onde o conteúdo foi criado.

- **Redirecionamento**: quando um usuário acessa um site ou aplicativo que utiliza uma CDN, a solicitação é direcionada para o servidor de borda mais próximo.
- **Entrega do conteúdo**: se o conteúdo solicitado estiver em cache no servidor de borda, ele é entregue diretamente ao usuário, sem a necessidade de buscar o conteúdo no servidor de origem.
- **Atualização**: caso o conteúdo não esteja em cache ou tenha sido atualizado, o servidor de borda busca a versão mais recente no servidor de origem e a armazena em cache para futuras solicitações.

Vantagens da Utilização de CDNs

- **Melhora a velocidade de carregamento**: ao entregar o conteúdo a partir de servidores próximos aos usuários, as CDNs reduzem a latência e o tempo de carregamento de páginas e aplicativos.
- **Reduz o consumo de largura de banda**: ao armazenar o conteúdo em cache, as CDNs diminuem a necessidade de transferir dados do servidor de origem, economizando largura de banda e custos de hospedagem.
- **Aumenta a disponibilidade e a escalabilidade**: as CDNs distribuem o conteúdo em vários servidores, o que garante que ele continue disponível mesmo que um servidor falhe ou haja picos de tráfego.
- **Proteção contra ataques DDoS**: as CDNs podem ajudar a proteger sites e aplicativos contra ataques de negação de serviço distribuído (DDoS), absorvendo o tráfego malicioso e evitando que ele chegue aos servidores de origem.

Casos de Uso

- **Sites de notícias e mídia**: para garantir que o conteúdo seja entregue rapidamente aos leitores em todo o mundo.

- **Plataformas de streaming de vídeo**: para oferecer vídeos de alta qualidade sem interrupções ou buffering.
- **Lojas online**: para melhorar a experiência de compra dos usuários e aumentar as conversões.
- **Empresas de software**: para distribuir downloads de aplicativos e atualizações de forma rápida e eficiente.

Exemplos de Uso

- Amazon CloudFront
- Google Cloud CDN
- Microsoft Azure CDN

Utilizando Bootstrap na Prática - CDN

SISTEMA DE GRADES

Adicionando as CNDs - Página HTML Padrão

JavaScript

```
<!DOCTYPE html>
<html lang="pt-br">
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1">
    <title>Bootstrap - CND</title>

    <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/boots
trap.min.css" rel="stylesheet"
integrity="sha384-QWTKZyjpPEjISv5WaRU90FeRpok6YctnYmDr5pNlyT2bRjX
h0JMHjY6hW+ALEwIH" crossorigin="anonymous">
  </head>
```

```
<body>
  <h1>Hello, world!</h1>

  <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstr
ap.bundle.min.js"
integrity="sha384-YvpcrYf0tY3lHB60NNkmXc5s9fDVZLESaAA55NDz0xhy9Gk
cIdslK1eN7N6jIeHz" crossorigin="anonymous"></script>
  </body>
</html>
```

Hello, world!

Utilizando a Classe “container” - Página HTML Padrão

JavaScript

```
<body>
  <div class="container">
    <h1>Olá mundo!</h1>
    <p>Lorem ipsum dolor sit amet...</p>
  </div>
  <hr>
  <div class="container-fluid">
    <h1>Olá mundo!</h1>
    <p>Lorem ipsum dolor sit amet consectetur...</p>
  </div>
</body>
```



Olá mundo!

Lorem ipsum dolor sit amet consectetur adipisicing elit. Eius omnis perspiciatis quasi reprehenderit illo voluptate dicta aspernatur, atque ratione iste ipsam vero nemo soluta natus odit, expedita explicabo. Culpa, perferendis!

Olá mundo!

Lorem ipsum dolor sit amet consectetur adipisicing elit. Eius omnis perspiciatis quasi reprehenderit illo voluptate dicta aspernatur, atque ratione iste ipsam vero nemo soluta natus odit. expedita explicabo. Culpa. perferendis!

Utilizando a Classe “container” - Breakpoints

JavaScript

```
<body>
  <div class="container-sm">
    <h1>Olá mundo!</h1>
    <p>Lorem ipsum dolor...</p>
  </div>
  <div class="container-md">
    <h1>Olá mundo!</h1>
    <p>Lorem ipsum dolor...</p>
  </div>
  <div class="container-lg">
    <h1>Olá mundo!</h1>
    <p>Lorem ipsum dolor...</p>
  </div>
  <div class="container-xl">
    <h1>Olá mundo!</h1>
    <p>Lorem ipsum dolor...</p>
  </div>
  <div class="container-xxl">
    <h1>Olá mundo!</h1>
```

```
<p>Lorem ipsum dolor...</p>  
</div>  
</body>
```

Olá mundo!

Lorem ipsum dolor sit amet co
ipsam vero nemo soluta natus

Olá mundo!

Lorem ipsum dolor sit amet co
ipsam vero nemo soluta natus

Olá mundo!

Lorem ipsum dolor sit amet co
ipsam vero nemo soluta natus

Olá mundo!

Lorem ipsum dolor sit amet co
ipsam vero nemo soluta natus

Olá mundo!

Lorem ipsum dolor sit amet consectetur adipis
expedita explicabo. Culpa, perferendis!

Entendendo o Código

Ao visualizar esta página em diferentes dispositivos ou tamanhos de tela, você notará que o conteúdo dentro de cada container se ajusta de acordo com o breakpoint correspondente. Os contêineres com sufixos como **-sm**, **-md**, **-lg**, **-xl** e **-xxl** terão larguras máximas diferentes em cada tamanho de tela, proporcionando uma experiência de visualização otimizada para cada dispositivo.

Valores de breakpoint padrão do Bootstrap:

- **xs (Extra small)**: menor que 576px

- **sm (Small)**: 576px ou maior
- **md (Medium)**: 768px ou maior
- **lg (Large)**: 992px ou maior
- **xl (Extra large)**: 1200px ou maior
- **xxl (Extra extra large)**: 1400px ou maior

O Bootstrap utiliza esses valores de breakpoint em conjunto com classes CSS e media queries para definir estilos e comportamentos diferentes para elementos HTML em diferentes tamanhos de tela.

Utilizando as Classes “row” e “col” - Sistema de Grades

JavaScript

<body>

```
<div class="container">
  <div class="row">
    <div class="col">
      Lorem ipsum dolor...
    </div>
    <div class="col">
      Lorem ipsum dolor...
    </div>
    <div class="col">
      Lorem ipsum dolor...
    </div>
    <div class="col">
      Lorem ipsum dolor...
    </div>
  </div>
  <div class="row">
    <div class="col-5">
      Foto
    </div>
```



```
<div class="col-3">  
    Logo  
</div>  
<div class="col-4">  
    Texto  
</div>  
</div>  
</div>  
</body>
```



Entendendo o Código

Sistema de Grid do Bootstrap

O Bootstrap usa um sistema de grid responsivo de 12 colunas, o que significa que é possível dividir uma linha em até 12 colunas, que se ajustam automaticamente ao tamanho da tela.

- **row:** representa uma linha na grade do Bootstrap, envolvendo as colunas (col-*) e garantindo que elas fiquem lado a lado.
- **col:** define uma coluna na grade do Bootstrap. Quando usada sem um número específico (ex: col-md-6), ela ocupa uma fração igual do espaço disponível na linha. Neste caso, como temos quatro colunas, cada uma ocupará 25% da largura da linha.

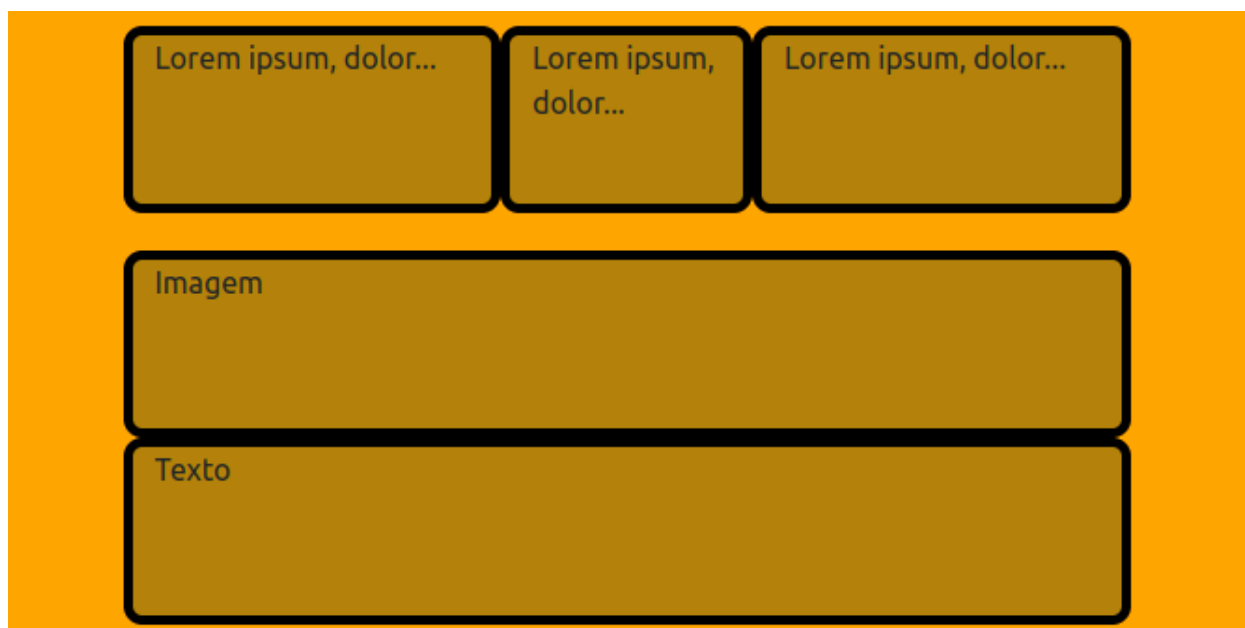
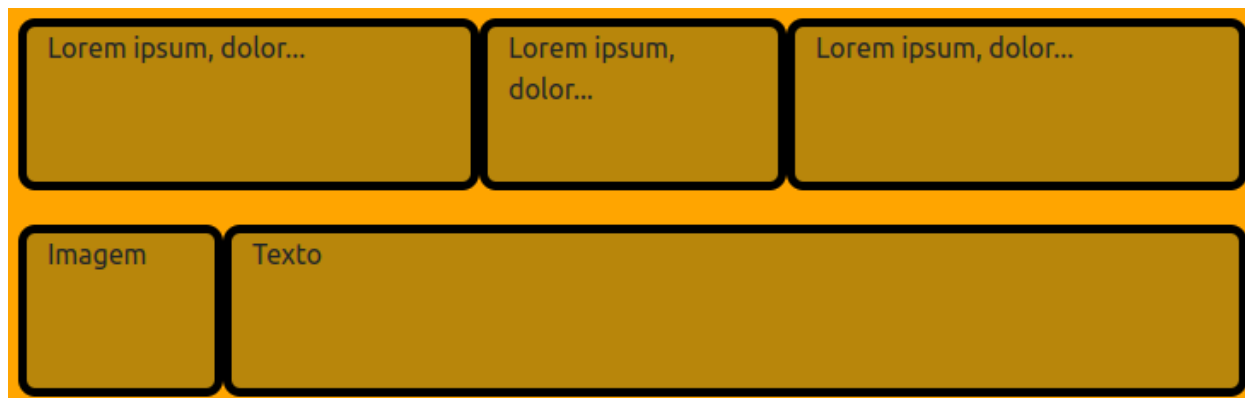
- **col-5, col-3, col-4**: definem colunas com larguras específicas. O número após o hífen indica quantas das 12 colunas do sistema de grade do Bootstrap a coluna deve ocupar.
 - col-5: Ocupa 5/12 da largura da linha.
 - col-3: Ocupa 3/12 da largura da linha.
 - col-4: Ocupa 4/12 da largura da linha.
-

Utilizando as Classes "row" e "col" - Sistema de Grades

JavaScript

```
<body>
  <div class="container">
    <div class="row">
      <div class="col">
        Lorem ipsum, dolor...
      </div>
      <div class="col-3">
        Lorem ipsum, dolor...
      </div>
      <div class="col">
        Lorem ipsum, dolor...
      </div>
    </div>

    <div class="row">
      <div class="col-md-2 col-lg-4">
        Imagem
      </div>
      <div class="col-md-10 col-lg-8">
        Texto
      </div>
    </div>
  </div>
</body>
```



Entendendo o Código

Primeira Linha <row>

- **col:** define uma coluna que ocupa uma fração igual do espaço disponível na linha. Neste caso, como temos três colunas, cada uma ocupará 1/3 da largura da linha.
- **col-3:** define uma coluna que ocupa 3/12 da largura da linha. Isso significa que ela ocupará 25% da largura da linha em telas menores que o breakpoint md (médio).

Segunda Linha <row>

- **col-md-2 col-lg-4**: define uma coluna que ocupa 2/12 da largura da linha em telas médias (md) e 4/12 da largura da linha em telas grandes (lg). Isso significa que ela ocupará 16.67% da largura da linha em telas médias e 33.33% da largura da linha em telas grandes.
 - **col-md-10 col-lg-8**: define uma coluna que ocupa 10/12 da largura da linha em telas médias (md) e 8/12 da largura da linha em telas grandes (lg). Isso significa que ela ocupará 83.33% da largura da linha em telas médias e 66.67% da largura da linha em telas grandes.
-

Utilizando a Classe “row-cols” - Sistema de Grades

JavaScript

```
<body>
  <div class="container">
    <div class="row row-cols-1 row-cols-md-2 row-cols-xl-4">
      <div class="col">
        Lorem ipsum, dolor...
      </div>
      <div class="col">
        Lorem ipsum, dolor...
      </div>
      <div class="col">
        Lorem ipsum, dolor...
      </div>
      <div class="col">
        Lorem ipsum, dolor...
      </div>
    </div>
  </div>
</body>
```

Lorem ipsum, dolor...	Lorem ipsum, dolor...
Lorem ipsum, dolor...	Lorem ipsum, dolor...



Lorem ipsum, dolor...
Lorem ipsum, dolor...
Lorem ipsum, dolor...
Lorem ipsum, dolor...

Entendendo o Código

As classes `row-cols-*` são usadas para especificar o número de colunas que você deseja em cada linha para diferentes tamanhos de tela (breakpoints).

- **row:** define um contêiner para as colunas.
- **row-cols-1:** em telas pequenas (extra small), a linha terá 1 coluna.
- **row-cols-md-2:** em telas médias (medium), a linha terá 2 colunas.

- **row-cols-xl-4**: em telas extra grandes (extra large), a linha terá 4 colunas.

Utilizando a Classe “row-cols” - Sistema de Grades

JavaScript

```
<div class="row">
  <div class="col">
    Lorem ipsum dolor...
  </div>
  <div class="col">
    <div class="row">
      <div class="col-4">
        Imagem
      </div>
      <div class="col-8">
        Texto
      </div>
      <div class="col-4">
        Imagem
      </div>
      <div class="col-8">
        Texto
      </div>
      <div class="col-4">
        Imagem
      </div>
      <div class="col-8">
        Texto
      </div>
      <div class="col-4">
        Imagem
      </div>
      <div class="col-8">
        Texto
      </div>
    </div>
  </div>
</div>
```

```
</div>  
</div>  
</div>  
</div>
```



Entendendo o Código

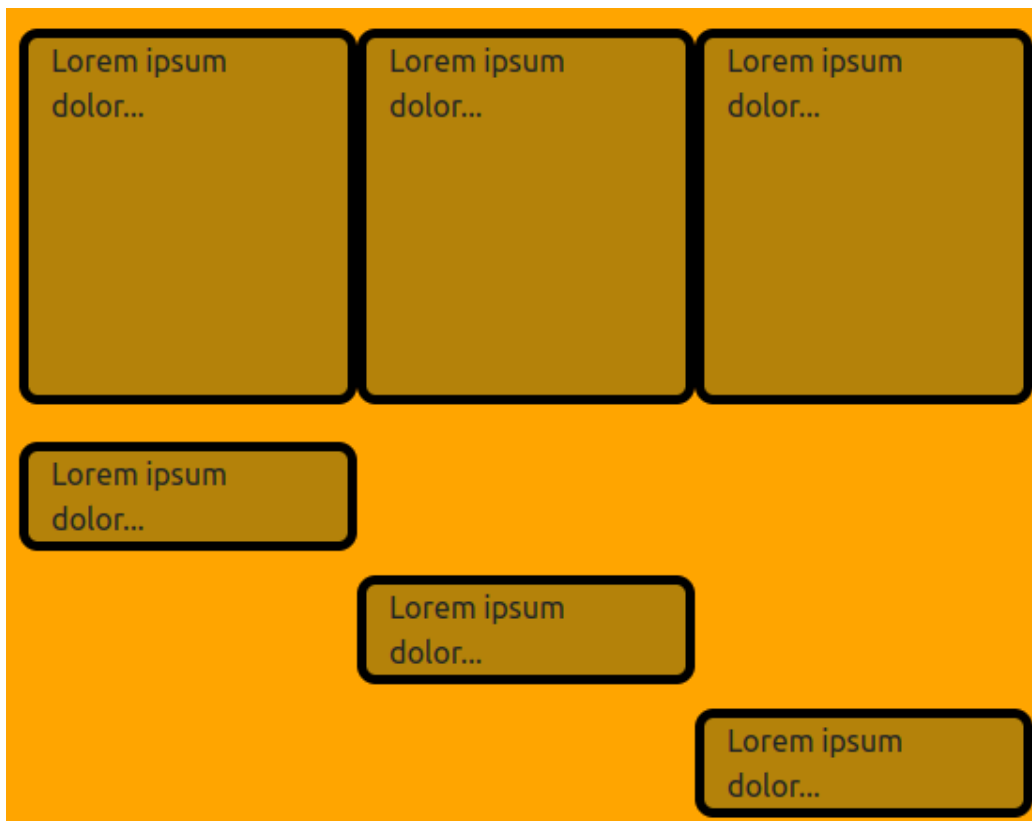
- Linha principal: divide o espaço horizontal disponível em duas colunas de largura igual.
 - Primeira coluna: Exibe o texto "Lorem ipsum dolor...".
 - Segunda coluna: Contém uma nova linha (row) aninhada.
- Linha aninhada: divide o espaço horizontal da segunda coluna em 12 colunas menores. Cada par de colunas (col-4 e col-8) organiza uma imagem e um texto lado a lado.
 - A classe col-4 ocupa 4 das 12 colunas (1/3 da largura) e exibe a imagem.
 - A classe col-8 ocupa 8 das 12 colunas (2/3 da largura) e exibe o texto.
 - Esse padrão se repete quatro vezes, criando quatro pares de imagem e texto.

Utilizando a Classe "row-cols" - Sistema de Grades



JavaScript

```
<div class="row align-items-auto">
  <div class="col">
    Lorem ipsum dolor...
  </div>
  <div class="col">
    Lorem ipsum dolor...
  </div>
  <div class="col">
    Lorem ipsum dolor...
  </div>
</div>
<div class="row">
  <div class="col align-self-start">
    Lorem ipsum dolor...
  </div>
  <div class="col align-self-center">
    Lorem ipsum dolor...
  </div>
  <div class="col align-self-end">
    Lorem ipsum dolor...
  </div>
</div>
```

Entendendo o Código

Primeira Linha:

- **row:** cria uma linha no sistema de grid do Bootstrap.
- **align-items-auto:** essa classe controla o alinhamento vertical dos itens dentro da linha. O valor auto permite que os itens se alinhem com base em seu conteúdo e tamanho.
- **col:** dentro da linha representa uma coluna. Como nenhuma largura específica é definida, as colunas se distribuem igualmente pelo espaço disponível na linha.

Segunda Linha:

- **row:** Cria uma linha no sistema de grid do Bootstrap.
 - **col:** cria uma coluna dentro da linha.

- **align-self-start, align-self-center, align-self-end**: essas classes controlam o alinhamento vertical individual de cada coluna dentro da linha:
 - **align-self-start**: alinha o item ao topo da coluna.
 - **align-self-center**: centraliza o item verticalmente na coluna.
 - **align-self-end**: alinha o item à parte inferior da coluna.
-

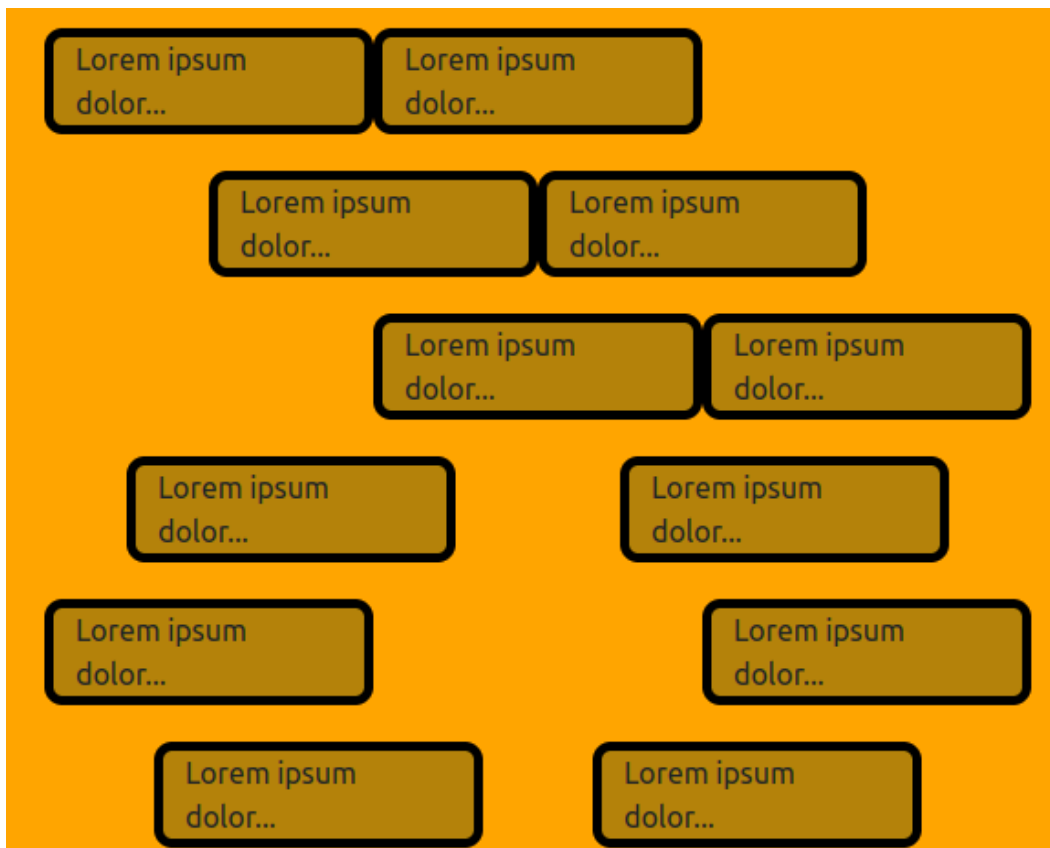
Utilizando a Classe "*justify-content*" - Sistema de Grades

JavaScript

```
<div class="row justify-content-start">
  <div class="col-4">
    Lorem ipsum dolor...
  </div>
  <div class="col-4">
    Lorem ipsum dolor...
  </div>
</div>
<div class="row justify-content-center">
  <div class="col-4">
    Lorem ipsum dolor...
  </div>
  <div class="col-4">
    Lorem ipsum dolor...
  </div>
</div>
<div class="row justify-content-end">
  <div class="col-4">
    Lorem ipsum dolor...
  </div>
  <div class="col-4">
    Lorem ipsum dolor...
  </div>
</div>
<div class="row justify-content-around">
```



```
<div class="col-4">
    Lorem ipsum dolor...
</div>
<div class="col-4">
    Lorem ipsum dolor...
</div>
</div>
<div class="row justify-content-between">
    <div class="col-4">
        Lorem ipsum dolor...
    </div>
    <div class="col-4">
        Lorem ipsum dolor...
    </div>
</div>
<div class="row justify-content-evenly">
    <div class="col-4">
        Lorem ipsum dolor...
    </div>
    <div class="col-4">
        Lorem ipsum dolor...
    </div>
</div>
```



Entendendo o Código

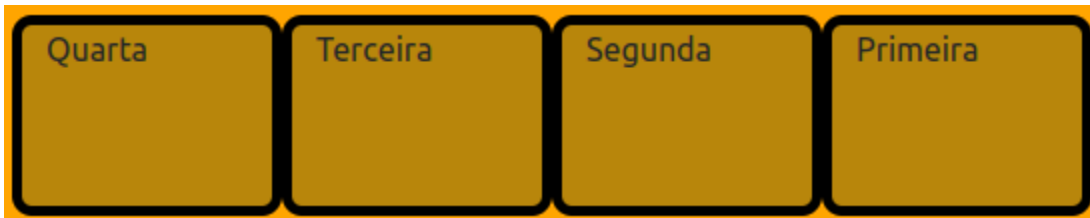
O código apresenta seis blocos (linhas), cada um utilizando um tipo diferente de alinhamento horizontal:

- **justify-content-start**: alinha as colunas ao início da linha.
- **justify-content-center**: centraliza as colunas na linha.
- **justify-content-end**: alinha as colunas ao final da linha.
- **justify-content-around**: distribui as colunas com espaço igual ao redor de cada coluna.
- **justify-content-between**: distribui as colunas com espaço igual entre elas, alinhando a primeira e a última coluna às extremidades da linha.
- **justify-content-evenly**: distribui as colunas com espaço igual entre elas e nas extremidades da linha.

Utilizando a Classe “*order*” - Sistema de Grades

JavaScript

```
<div class="row">
  <div class="col-3 order-last">
    Primeira
  </div>
  <div class="col-3 order-2">
    Segunda
  </div>
  <div class="col-3 order-1">
    Terceira
  </div>
  <div class="col-3 order-first">
    Quarta
  </div>
</div>
```



Entendendo o Código

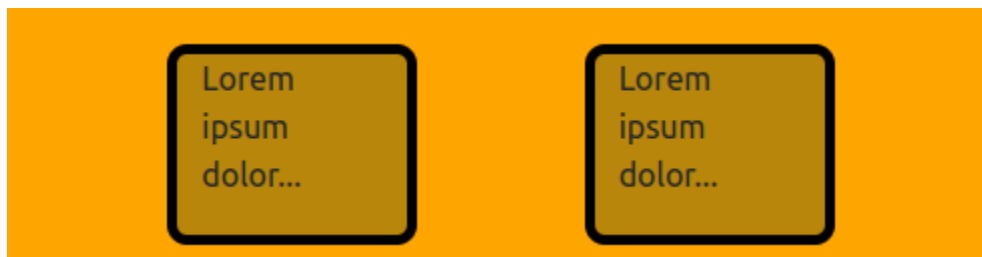
A classe `order` do Bootstrap afeta a ordem de exibição das colunas:

- **order-last**: esta coluna será exibida por último.
- **order-2**: esta coluna será exibida em segundo lugar (após a coluna com `order-1`).
- **order-1**: esta coluna será exibida em primeiro lugar.
- **order-first**: esta coluna será exibida antes de qualquer outra, mesmo aquelas sem uma classe `order` definida.

Utilizando a Classe “*offset*” - Sistema de Grades

JavaScript

```
<div class="row">
  <div class="col-3 offset-2">
    Lorem ipsum dolor...
  </div>
  <div class="col-3 offset-2">
    Lorem ipsum dolor...
  </div>
</div>
```



Entendendo o Código

A classe “*offset-**” é utilizada para adicionar um espaçamento à esquerda da coluna, deslocando-a para a direita.

- **row**: cria uma linha no sistema de grid do Bootstrap.
- **col-3**: cria uma coluna que ocupa 3 das 12 colunas disponíveis no sistema de grid do Bootstrap.
- **offset-2**: adiciona um espaçamento à esquerda da coluna, deslocando-a para a direita em 2 colunas.

Utilizando as Classes “*margin*” e “*padding*” - Sistema de Grades

JavaScript

```
<div class="row">
  <div class="col-4 m-auto py-md-4 py-xl-0">
    Lorem ipsum dolor...
  </div>
  <div class="col-4 mt-5">
    Lorem ipsum dolor...
  </div>
</div>
```



Entendendo o Código

- **row**: cria uma linha no sistema de grid do Bootstrap.
- **col-4 m-auto py-md-4 py-xl-0**:
 - **col-4**: cria uma coluna que ocupa 4 das 12 colunas disponíveis no sistema de grid do Bootstrap.
 - **m-auto**: centraliza a coluna horizontalmente dentro de seu contêiner pai.
 - **py-md-4**: adiciona preenchimento vertical (padding) de tamanho 4 (uma unidade de medida do Bootstrap) em telas de tamanho médio (md) e maiores.
 - **py-xl-0**: remove o preenchimento vertical em telas extra grandes (xl) e maiores.
- **col-4 mt-5**:
 - **col-4**: cria outra coluna que também ocupa 4 das 12 colunas disponíveis.

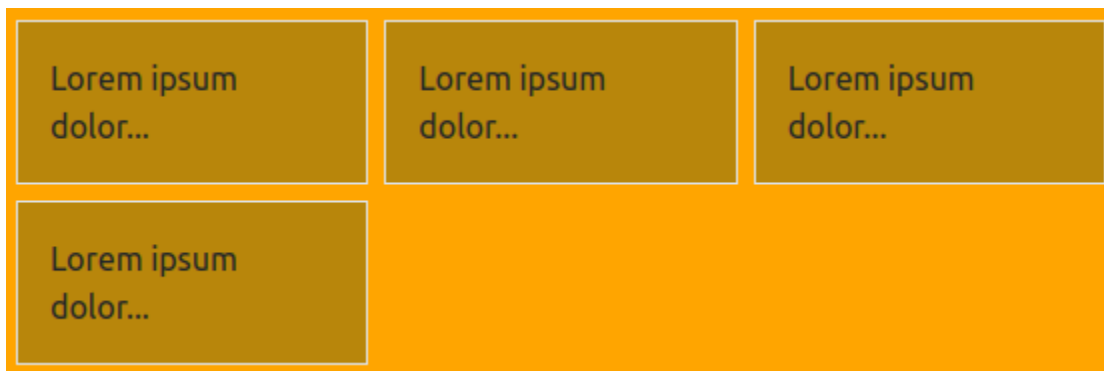


- mt-5: adiciona uma margem superior (margin-top) de tamanho 5 (a maior unidade de medida de margem do Bootstrap).

Utilizando a Classe “Gutter” - Sistema de Grades

JavaScript

```
<div class="row row-cols-3 g-2">
  <div class="col">
    <div class="bg-orange p-3 border">
      Lorem ipsum dolor...
    </div>
  </div>
  <div class="col">
    <div class="bg-orange p-3 border">
      Lorem ipsum dolor...
    </div>
  </div>
  <div class="col">
    <div class="bg-orange p-3 border">
      Lorem ipsum dolor...
    </div>
  </div>
  <div class="col">
    <div class="bg-orange p-3 border">
      Lorem ipsum dolor...
    </div>
  </div>
</div>
```

Entendendo o Código

- **row row-cols-3 g-2:**
 - **row**: cria uma linha no sistema de grid do Bootstrap.
 - **row-cols-3**: define que a linha terá 3 colunas por padrão. Isso significa que, em telas maiores, a linha será dividida em 3 colunas iguais. Em telas menores, as colunas se adaptarão para ocupar toda a largura disponível.
 - **g-2**: adiciona um espaçamento (gutter) de 2 unidades entre as colunas. O Bootstrap usa uma escala de 0 a 5 para definir o espaçamento.
 - **col**: cada "col" dentro da linha representa uma coluna. Como a classe row-cols-3 foi definida, as colunas se distribuirão igualmente pelo espaço disponível na linha, com um máximo de 3 colunas por linha.
 - **bg-orange p-3 border**:
 - **bg-orange**: define a cor de fundo do elemento como laranja.
 - **p-3**: adiciona um preenchimento (padding) de 3 unidades dentro do elemento.
 - **border**: adiciona uma borda ao elemento.
-

FONTES

Utilizando as Classes “h*” e “display-*” - Fontes

JavaScript

```
<h1>Título 01</h1>
<h2>Título 02</h2>
<h3>Título 03</h3>
<h4>Título 04</h4>
<h5>Título 05</h5>
<h6>Título 06</h6>
```

```
<p class="h1">Título 01</p>
<p class="h2">Título 02</p>
<p class="h3">Título 03</p>
<p class="h4">Título 04</p>
<p class="h5">Título 05</p>
<p class="h6">Título 06</p>
```

```
<h1 class="display-1">Título 01</h1>
<h1 class="display-2">Título 02</h1>
<h1 class="display-3">Título 03</h1>
<h1 class="display-4">Título 04</h1>
<h1 class="display-5">Título 05</h1>
<h1 class="display-6">Título 06</h1>
```

Título 01

Título 02

Título 03

Título 04

Título 05

Título 06

Título 01

Título 02

Título 03

Título 04

Título 05

Título 06

Título 01

Título 02

Título 03

Título 04

Título 05

Título 06

Entendendo o Código

Classes de Título (h1 a h6)

- **<p class="h1"> a <p class="h6">**: o Bootstrap oferece classes de título que podem ser aplicadas a elementos de parágrafo (<p>) para exibir títulos com o estilo correspondente às tags <h1> a <h6>. No código acima, cada parágrafo <p> com a classe h1 a h6 exibe um título com o mesmo texto e estilo visual que as tags <h1> a <h6> correspondentes.

Classes de Título Display (display-1 a display-6)

- **<h1 class="display-1"> a <h1 class="display-6">**: oferece classes de título "display" que fornecem opções adicionais para exibir títulos com tamanhos maiores e mais impactantes. No código acima, cada tag <h1> com a classe display-1 a display-6 exibe um título com o texto "Título 01" a "Título 06", respectivamente, mas com um estilo visual diferente das tags <h1> a <h6> padrão e das classes h1 a h6.

Observações

Usar as tags de título padrão (<h1> a <h6>) é a forma mais semântica de definir títulos em HTML, pois elas indicam a hierarquia do conteúdo para os mecanismos de busca e leitores de tela. É importante usar os títulos de forma consistente e hierárquica para garantir a acessibilidade e a organização do conteúdo.

Utilizando as Classes "text-*" e "blockquote-*" - Fontes

JavaScript

```
<figure>
  <blockquote class="blockquote">
    Encare os problemas...
  </blockquote>
  <figcaption class="blockquote-footer">
```

```
        Gil Eduardo de Andrade <cite
title="especialidade">Professor</cite>
    </figcaption>
</figure>

<figure class="text-center">
    <blockquote class="blockquote">
        Encare os problemas...
    </blockquote>
    <figcaption class="blockquote-footer">
        Gil Eduardo de Andrade <cite
title="especialidade">Professor</cite>
    </figcaption>
</figure>

<figure class="text-end">
    <blockquote class="blockquote">
        Encare os problemas...
    </blockquote>
    <figcaption class="blockquote-footer">
        Gil Eduardo de Andrade <cite
title="especialidade">Professor</cite>
    </figcaption>
</figure>
```

Encare os problemas como oportunidades para desenvolver o talento
necessário para resolvê-los

— Gil Eduardo de Andrade *Professor*

Encare os problemas como oportunidades para desenvolver o talento
necessário para resolvê-los

— Gil Eduardo de Andrade *Professor*

Encare os problemas como oportunidades para desenvolver o talento
necessário para resolvê-los

— Gil Eduardo de Andrade *Professor*

Entendendo o Código

Elemento <figure>: representa conteúdo independente, geralmente com uma legenda, como uma imagem, ilustração, diagrama, código, entre outros. No código acima ele está sendo usado para encapsular a citação e sua atribuição.

Classes “blockquote”, “blockquote-footer” e elemento <cite>:

- **blockquote**: classe que estiliza o elemento como uma citação em bloco, adicionando recuo à esquerda e margem inferior.
- **blockquote-footer**: classe que estiliza a linha que contém a fonte da citação. Geralmente, é usada em conjunto com o elemento <figcaption>.
- **cite**: tag HTML que define o título de uma obra criativa (livro, poema, música, filme). O atributo “title” pode ser usado para fornecer informações adicionais sobre a obra.

Classes “text-center” e “text-end”:

- **text-center**: classe que alinha o texto ao centro.
 - **text-end**: classe do Bootstrap alinha o texto à direita.
-

Utilizando as Classes “fs-*” e “fw-*” - Fontes



JavaScript

<h1>Font Size</h1>

<p class="fs-1">Font Size: fs-1</p>

<p class="fs-2">Font Size: fs-2</p>

<p class="fs-3">Font Size: fs-3</p>

<p class="fs-4">Font Size: fs-4</p>

<p class="fs-5">Font Size: fs-5</p>

<p class="fs-6">Font Size: fs-6</p>

<h1>Font Weight</h1>

<p class="fw-lighter">Font Weight: fw-lighter</p>

<p class="fw-light">Font Weight: fw-light</p>

<p class="fw-normal">Font Weight: fw-normal</p>

<p class="fw-bold">Font Weight: fw-bold</p>

<p class="fw-bolder">Font Weight: fw-bolder</p>

<h1>Font-Style</h1>

<p class="fst-italic">Font Style: fst-italic</p>

<p class="fst-normal">Font Style: fst-normal</p>

Font Size

Font Size: fs-1

Font Size: fs-2

Font Size: fs-3

Font Size: fs-4

Font Size: fs-5

Font Size: fs-6

Font Weight

Font Weight: fw-lighter

Font Weight: fw-light

Font Weight: fw-normal

Font Weight: fw-bold

Font Weight: fw-bolder

Font-Style

Font Style: fst-italic

Font Style: fst-normal

Entendendo o Código

- As classes ***"fs-1"*** a ***"fs-6"*** correspondem aos tamanhos de fonte do Bootstrap, onde ***"fs-1"*** é o maior tamanho e ***"fs-6"*** é o menor. Essas classes definem o tamanho da fonte com base em um sistema de escala responsivo, ajustando-se a diferentes tamanhos de tela.
- Os elementos `<p class="fw-lighter">...</p>` até `<p class="fw-bolder">...</p>` demonstram diferentes espessuras de fonte.
 - **fw-lighter**: Define a espessura da fonte como mais leve.
 - **fw-light**: Define a espessura da fonte como leve.
 - **fw-normal**: Define a espessura da fonte como normal.
 - **fw-bold**: Define a espessura da fonte como negrito.
 - **fw-bolder**: Define a espessura da fonte como mais negrito
- `<p class="fst-italic">...</p>`: define o estilo da fonte como itálico.
- `<p class="fst-normal">...</p>`: define o estilo da fonte como normal (não itálico).

CORES E ÍCONES

Utilizando as Classes ***"text-*"*** e ***"bg-*"*** - Cores

JavaScript

```
<p class="text-primary">Primary (Azul)</p>
<p class="text-secondary">Secondary (Cinza)</p>

<p class="text-success">Success (Verde)</p>
<p class="text-danger">Danger (Vermelho)</p>
<p class="text-warning">Warning (Laranja)</p>
<p class="text-info">Info (Azul Água)</p>
```

```
<p class="text-light bg-dark">Light (Branco)</p>
<p class="text-dark">Dark (Preto)</p>

<p class="text-black">Black (Preto)</p>
<p class="text-white bg-dark">White (Branco)</p>

<p class="text-black-50">Black - Escala 50</p>
<p class="text-white-50 bg-danger">White - Escala 50</p>
```

Texto

Primary (Azul)

Secondary (Cinza)

Success (Verde)

Danger (Vermelho)

Warning (Laranja)

Info (Azul Água)

Light (Branco)

Entendendo o Código

O Bootstrap fornece classes de utilitário para definir cores de texto e cores de fundo. Essas classes seguem uma nomenclatura consistente, facilitando a aplicação de estilos comuns.

- **<p class="text-primary">Primary (Azul)</p>**:
 - text-primary: aplica a cor primária do tema Bootstrap ao texto, que geralmente é um tom de azul.

- **<p class="text-secondary">Secondary (Cinza)</p>:**
 - text-secondary: aplica a cor secundária do tema Bootstrap ao texto, que geralmente é um tom de cinza.
- **<p class="text-success">Success (Verde)</p>:**
 - text-success: aplica a cor de sucesso ao texto, que geralmente é um tom de verde. Usado para indicar ações bem-sucedidas ou resultados positivos.
- **<p class="text-danger">Danger (Vermelho)</p>:**
 - text-danger: aplica a cor de perigo ao texto, que geralmente é um tom de vermelho. Usado para indicar erros, avisos ou ações perigosas.
- **<p class="text-warning">Warning (Laranja)</p>:**
 - text-warning: aplica a cor de aviso ao texto, que geralmente é um tom de laranja ou amarelo. Usado para indicar avisos ou alertas.
- **<p class="text-info">Info (Azul Água)</p>:**
 - text-info: aplica a cor de informação ao texto, que geralmente é um tom de azul claro ou ciano. Usado para fornecer informações adicionais ou contextuais.
- **<p class="text-light bg-dark">Light (Branco)</p>:**
 - text-light: aplica a cor clara (geralmente branco ou um tom muito claro) ao texto.
 - bg-dark: aplica a cor escura (geralmente preto ou um tom muito escuro) ao fundo do parágrafo.
- **<p class="text-dark">Dark (Preto)</p>:**
 - text-dark: Aplica a cor escura (geralmente preto ou um tom muito escuro) ao texto.
- **<p class="text-black">Black (Preto)</p>:**
 - text-black: aplica a cor preta ao texto.
- **<p class="text-white bg-dark">White (Branco)</p>:**
 - text-white: aplica a cor branca ao texto.
 - bg-dark: aplica a cor escura ao fundo do parágrafo.

- **<p class="text-black-50">Black - Escala 50</p>**:
 - text-black-50: aplica um tom de preto com 50% de opacidade ao texto. Isso cria um efeito de texto semitransparente.
- **<p class="text-white-50 bg-danger">White - Escala 50</p>**:
 - text-white-50: aplica um tom de branco com 50% de opacidade ao texto.
 - bg-danger: aplica a cor de perigo (vermelho) ao fundo do parágrafo.

Utilizando as Classes "bg-*" - Cores

JavaScript

```
<h1>Background</h1>
<div class="p-3 bg-primary">background</div>
<div class="p-3 bg-secondary">background</div>
<div class="p-3 bg-success">background</div>
<div class="p-3 bg-danger">background</div>
<div class="p-3 bg-warning">background</div>
<div class="p-3 bg-info">background</div>
<div class="p-3 bg-light">background</div>
<div class="p-3 bg-dark text-light">background</div>
<div class="p-3 bg-body">background</div>
<div class="p-3 bg-white">background</div>
<div style="background-image:
url('http://gileduardo.com.br/ifpr/dw/img/tulipa.jpg');">
<div class="p-3 bg-light bg-transparent
text-light">background</div>
</div>

<h1>Background - Gradient</h1>
<div class="p-3 bg-primary bg-gradient">background</div>
<div class="p-3 bg-secondary bg-gradient">background</div>
<div class="p-3 bg-success bg-gradient">background</div>
```

```
<div class="p-3 bg-danger bg-gradient">background</div>  
<div class="p-3 bg-warning bg-gradient">background</div>  
<div class="p-3 bg-info bg-gradient">background</div>  
<div class="p-3 bg-light bg-gradient">background</div>  
<div class="p-3 bg-dark bg-gradient text-light">background</div>
```



Background - Gradient



Entendendo o Código

- **bg-body**: define a cor de fundo do elemento div como a cor de fundo padrão do corpo do documento, geralmente um tom de branco ou cinza claro.
- **bg-light**: define a cor de fundo como um tom de cinza claro.
- **bg-transparent**: define o fundo como transparente, permitindo que o fundo do elemento pai seja visível.
- **bg-gradient**: aplica um gradiente linear à cor de fundo.
- **text-light**: define a cor do texto para um tom claro, geralmente branco.

Utilizando as Classes *"border-*"* - Bordas

JavaScript

```
<h1>Aditiva</h1>
<div class="box bg-light border"></div>
<div class="box bg-light border-top"></div>
<div class="box bg-light border-bottom"></div>
<div class="box bg-light border-start"></div>
<div class="box bg-light border-end"></div>

<h1>Subtrativa</h1>
<div class="box bg-light border"></div>
<div class="box bg-light border border-top-0"></div>
<div class="box bg-light border border-bottom-0"></div>
<div class="box bg-light border border-start-0"></div>
<div class="box bg-light border border-end-0"></div>

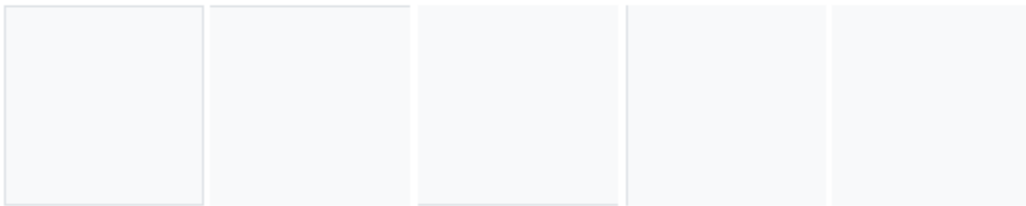
<h1>Cores</h1>
<div class="box bg-light border"></div>
<div class="box bg-light border border-primary"></div>
<div class="box bg-light border border-secondary"></div>
<div class="box bg-light border border-success"></div>
<div class="box bg-light border border-danger"></div>
<div class="box bg-light border border-warning"></div>
<div class="box bg-light border border-info"></div>
<div class="box bg-dark border border-light"></div>
<div class="box bg-light border border-dark"></div>
<div class="box bg-light border border-white"></div>

<h1>Tamanho</h1>
<div class="box bg-light border border-1"></div>
<div class="box bg-light border border-2"></div>
<div class="box bg-light border border-3"></div>
<div class="box bg-light border border-4"></div>
<div class="box bg-light border border-5"></div>

<h1>Arredondar</h1>
```

```
<div class="box bg-light border border-1 rounded-0"></div>  
<div class="box bg-light border border-1 rounded-1"></div>  
<div class="box bg-light border border-1 rounded-2"></div>  
<div class="box bg-light border border-1 rounded-3"></div>
```

Aditiva



Entendendo o Código

- **box**: classe criada para especificar o layout das caixas (quadrados) utilizadas nos exemplos de uso das classes de borda do Bootstrap.
- **border**: adiciona uma borda padrão à caixa.
- **border-top**: adiciona uma borda superior.
- **border-bottom**: adiciona uma borda inferior.
- **border-start**: adiciona uma borda à esquerda.
- **border-end**: adiciona uma borda à direita.
- **border-top-0**: remove a borda superior.
- **border-bottom-0**: remove a borda inferior.
- **border-start-0**: remove a borda esquerda.
- **border-end-0**: remove a borda direita.
- **rounded-0**: borda sem arredondamento.
- **rounded-1, rounded-2, rounded-3**: borda com diferentes níveis de arredondamento.

Utilizando as Classes "btn" e "btn-*" - Botões

JavaScript

```
<h1>Elementos</h1>
<button class="btn btn-dark">OK</button>
<input class="btn btn-dark" type="button" value="OK" />
<input class="btn btn-dark" type="submit" value="OK" />
<a href="#" class="btn btn-dark">OK</a>

<h1>Cores</h1>
<button class="btn btn-primary">OK</button>
<button class="btn btn-secondary">OK</button>
<button class="btn btn-success">OK</button>
<button class="btn btn-danger">OK</button>
<button class="btn btn-warning">OK</button>
<button class="btn btn-info">OK</button>
<button class="btn btn-light">OK</button>
<button class="btn btn-dark">OK</button>
<button class="btn btn-link">OK</button>

<h1>Cores</h1>
<button class="btn btn-outline-primary">OK</button>
<button class="btn btn-outline-secondary">OK</button>
<button class="btn btn-outline-success">OK</button>
<button class="btn btn-outline-danger">OK</button>
<button class="btn btn-outline-warning">OK</button>
<button class="btn btn-outline-info">OK</button>
<button class="btn btn-outline-light">OK</button>
<button class="btn btn-outline-dark">OK</button>

<h1>Tamanho</h1>
<button class="btn btn-primary btn-lg">OK</button>
<button class="btn btn-primary">OK</button>
<button class="btn btn-primary btn-sm">OK</button>

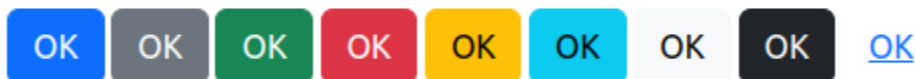
<h1>Desativado</h1>
```

```
<button class="btn btn-primary" disabled>OK</button>  
<button class="btn btn-secondary" disabled>OK</button>  
<button class="btn btn-success" disabled>OK</button>
```

Elementos



Cores



Cores



Entendendo o Código

A classe ***"btn"*** aplica os estilos básicos de botão do Bootstrap. As classes ***"btn-primary"***, ***"btn-secondary"***, ***"btn-success"***, ***"btn-danger"***, ***"btn-warning"***, ***"btn-info"***, ***"btn-light"***, ***"btn-dark"***, ***"btn-link"*** definem as cores de fundo dos botões.

- **primary**: Azul.
- **secondary**: Cinza.
- **success**: Verde.
- **danger**: Vermelho.
- **warning**: Amarelo.
- **info**: Azul claro.
- **light**: Cinza claro.
- **dark**: Cinza escuro/preto.

As classes ***"btn-outline-**** criam botões com contorno colorido e fundo transparente. O ***"*"*** representa as mesmas cores do parágrafo anterior.

- **btn-lg**: Torna o botão grande.
- **btn-sm**: Torna o botão pequeno.

O atributo ***"disabled"*** desativa os botões, tornando-os não clicáveis.

Utilizando as Classes ***"bi"*** e ***"bi-**** - Ícones

<https://icons.getbootstrap.com/>

JavaScript

```
<h1>Bootstrap Icons</h1>
<i class="bi bi-google fs-1 text-danger"></i>
<i class="bi bi-facebook fs-1 text-primary"></i>
<i class="bi bi-twitter fs-1 text-info"></i>
<i class="bi bi-github fs-1 text-dark"></i>
<i class="bi bi-instagram fs-1 text-warning"></i>
```

Bootstrap Icons



Entendendo o Código

- **<i> Tag**:
 - O elemento ***<i>*** é uma tag HTML que representa um ícone. Tradicionalmente, era usado para texto itálico, mas agora é comumente usado com bibliotecas de ícones.
- **class="bi bi-..."**:
 - ***bi***: indica que o ícone pertence à biblioteca Bootstrap Icons.

- bi-google, bi-facebook, bi-twitter, bi-github, bi-instagram: especificam o ícone a ser exibido. Cada um corresponde a um ícone da respectiva plataforma.
- **fs-1**:
 - Esta classe do Bootstrap define o tamanho da fonte do ícone. fs-1 é uma das maiores opções de tamanho de fonte no Bootstrap, resultando em ícones grandes.
- **text-danger, text-primary, text-info, text-dark, text-warning**:
 - Estas classes do Bootstrap definem a cor do texto (e, portanto, a cor do ícone).
 - text-danger: vermelho.
 - text-primary: azul.
 - text-info: azul claro.
 - text-dark: preto ou cinza escuro.
 - text-warning: laranja.

PROPRIEDADE - DISPLAY

O Bootstrap disponibiliza, também, classes de exibição que permitem indicar quando e como os elementos HTML devem ser apresentados, facilitando a construção de layouts responsivos.

NOTAÇÃO

As classes de exibição se aplicam a todos os pontos de interrupção, de “**xs**” a “**xl**”. Sendo assim, as classes são nomeadas usando o formato:

- ***.d-{value}*** para **xs**
- ***.d-{breakpoint}-{value}*** para **sm, md, lg, e xl**.

Onde “*value*” pode assumir os seguintes valores:

- *none*
- *inline*
- *inline-block*
- *block*
- *table*
- *table-cell*
- *table-row*
- *flex*
- *inline-flex*

As consultas de mídia afetam as larguras de tela com o ponto de interrupção fornecido ou maior. Por exemplo, *.d-lg-none* configura um *display: none*; em ambas as telas: *lg* e *xl*.

JavaScript

```
<hr>
<div class="d-inline p-2 bg-primary text-white">d-inline</div>
<div class="d-inline p-2 bg-dark text-white">d-inline</div>
<hr>
<span class="d-block p-2 bg-primary text-white">d-block</span>
<span class="d-block p-2 bg-dark text-white">d-block</span>
```

d-inline d-inline

d-block

d-block

Entendendo o Código

- **class="d-inline"**: altera o comportamento de exibição do elemento para display: inline. Elementos inline ocupam apenas a largura necessária para o seu conteúdo e fluem horizontalmente com outros elementos inline adjacentes.
- **class="d-block"**: altera o comportamento de exibição do elemento para display: block. Elementos block ocupam toda a largura disponível do seu contêiner pai e forçam outros elementos (tanto inline quanto block) a aparecerem em novas linhas abaixo e acima deles.

OCULTANDO ELEMENTOS

Pensando em uma dinâmica de desenvolvimento mais rápida e otimizada para dispositivos móveis, o Bootstrap disponibiliza classes de exibição responsivas que permitem exibir e ocultar elementos de acordo com o dispositivo (tamanho de tela).

Para ocultar elementos, basta usar a classe **.d-none** ou uma das classes **.d-{sm, md, lg, xl, xxl}-none** para qualquer variação de tela responsiva.

Para mostrar um elemento somente em um determinado intervalo de tamanhos de tela, você pode combinar uma classe **.d-*-none** com uma classe **.d-*-***. Por exemplo, utilizando as seguintes classes em conjunto: **.d-none .d-md-block .d-xl-none .d-xxl-none**, o elemento será ocultado para todos os tamanhos de tela, exceto médio e grande.

TAMANHO DE TELA	CLASSE
Oculto em todos os tamanhos	.d-none
Oculto apenas em “xs”	.d-none .d-sm-block

Oculto apenas em “ <i>sm</i> ”	.d-sm-none .d-md-block
Oculto apenas em “ <i>md</i> ”	.d-md-none .d-lg-block
Oculto apenas no “ <i>lg</i> ”	.d-lg-none .d-xl-block
Oculto apenas no “ <i>xl</i> ”	.d-xl-none .d-xxl-block
Oculto apenas em “ <i>xxl</i> ”	.d-xxl-none
Visível em todos	.d-block
Visível apenas em “ <i>xs</i> ”	.d-block .d-sm-none
Visível apenas em “ <i>sm</i> ”	.d-none .d-sm-block .d-md-none
Visível apenas em “ <i>md</i> ”	.d-none .d-md-block .d-lg-none
Visível apenas em “ <i>lg</i> ”	.d-none .d-lg-block .d-xl-none
Visível apenas em “ <i>xl</i> ”	.d-none .d-xl-block .d-xxl-none
Visível apenas em “ <i>xxl</i> ”	.d-none .d-xxl-block

JavaScript

```
<div class="d-lg-none text-bg-primary">
  Oculta em Telas "lg" e mais largas
</div>
<div class="d-none d-lg-block text-bg-success">
  Oculta em telas menores que "lg"
</div>
```

Oculto em Telas "lg" e mais largas



Oculto em telas menores que "lg"

Entendendo o Código

- **class="d-lg-none"**: classe de visibilidade responsiva do Bootstrap.
 - **d-:** indica que é uma classe de "display".
 - **lg:** indica o breakpoint *"large"* (telas com largura de 992 pixels ou mais).
 - **none:** indica que o elemento terá *"display: none;"* aplicado, ou seja, ficará oculto.
 - **Resumindo:** faz com que a div seja ocultada em telas com tamanho *"large"* (lg) e em telas ainda maiores (xl, xxl) - será visível apenas em telas menores que *"large"* (xs, sm, md);
- **class="d-none d-lg-block"**: duas classes de visibilidade responsiva aplicadas ao mesmo elemento.
 - **d-none:** faz com que o elemento seja ocultado por padrão em telas extra-pequenas (xs).
 - **d-lg-block:**
 - **d-:** indica que é uma classe de "display".
 - **lg:** indica o breakpoint *"large"* (telas com largura de 992 pixels ou mais).
 - **block:** indica que o elemento terá *"display: block;"* aplicado, ocupando toda a largura disponível e forçando quebras de linha.
 - **Resumindo:** faz com que a <div> seja exibida como um bloco (*display: block*) apenas em telas com tamanho *"large"* (lg) e em telas maiores (xl, xxl).