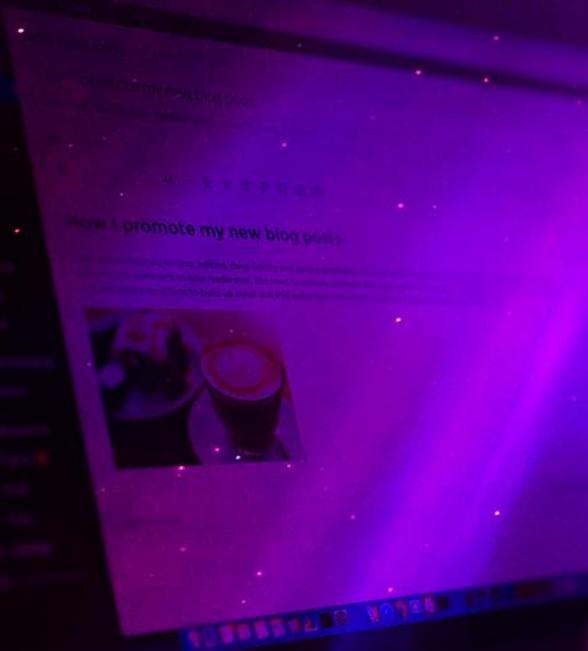


B

BOOTSTRAP 5

Guia Rápido para Iniciantes

DIEGO MARIANO



Edição

e-book: **BOOTSTRAP 5** – GUIA RÁPIDO PARA INICIANTES

DIEGO MARIANO

Bootstrap 5

GUIA RÁPIDO PARA INICIANTES

ALFAHELIX PUBLICAÇÕES



BOOTSTRAP 5

GUIA RÁPIDO PARA INICIANTES

© 2022 | Alfahelix, CNPJ: 37.524.984/0001-10

Lagoa Santa, MG, Brasil

www.alfahelix.com.br

Autor: Diego Mariano

Capa: adaptado de Hans Braxmeier | StockSnap por Pixabay | Bootstrap logo oficial by Bootstrap Authors

Páginas: 108

ISBN: 978-65-992753-4-0

Título: Bootstrap 5 – Guia Rápido para Iniciantes

*Este livro está publicado sob uma licença de compartilhamento Creative Commons Atribuição-NãoComercial 4.0 Internacional. Você pode utilizar qualquer conteúdo aqui apresentado, desde que cite: “Mariano, D. **Bootstrap 5 – Guia Rápido para Iniciantes.** 1^a ed. Alfabelix, Lagoa Santa. 2022”.*

Bootstrap está disponibilizado sob a licença MIT (<https://github.com/twbs/bootstrap/blob/main/LICENSE>) e a documentação sob a licença CC BY 3.0 (<https://creativecommons.org/licenses/by/3.0/>). Direitos reservados aos autores do Bootstrap (<https://getbootstrap.com/docs/5.1/about/team/>).

Sumário

INTRODUÇÃO	8
TEMPLATE INICIAL.....	8
CORES	10
TAMANHOS	10
LAYOUT.....	12
PONTOS DE QUEBRA (COLAPSO)	12
CONTÊINERES DE DADOS	13
SISTEMA DE GRADE	14
<i>Distribuindo colunas.....</i>	<i>17</i>
<i>Classes offset.....</i>	<i>19</i>
<i>Calhas (gutters).....</i>	<i>20</i>
<i>Margem e preenchimento</i>	<i>22</i>
<i>Posicionamento fixo e visibilidade</i>	<i>24</i>
<i>Altura e largura.....</i>	<i>29</i>
SOMBRAIS E OPACIDADE	31
CONTEÚDO.....	34
TIPOGRAFIA	34
<i>Tamanhos de textos.....</i>	<i>36</i>
IMAGENS E FIGURAS	38
TABELAS.....	43
<i>Tabelas coloridas</i>	<i>48</i>
<i>Tabelas responsivas</i>	<i>49</i>
FORMULÁRIOS.....	53
<i>Grupo de inputs.....</i>	<i>55</i>
<i>Desabilitando um campo (somente leitura)</i>	<i>56</i>
<i>Rótulos flutuantes.....</i>	<i>56</i>

<i>Validação de formulários</i>	57
COMPONENTES	59
1. CAIXAS EXPANSIVAS (SANFONA E CLASSE DE COLAPSO)	59
2. CAIXA DE ALERTAS (<i>ALERT</i>) E CAIXA DE NOTIFICAÇÕES (<i>TOAST</i>).....	65
3. EMBLEMAS (<i>BADGE</i>)	68
4. BOTÕES, GRUPO DE BOTÕES E DROPODOWN	69
<i>Alterando o tamanho de um botão</i>	71
<i>Estilo de botões: ativo, desabilitado, ocupando toda a linha e botão fechar</i>	72
<i>Menus suspensos (dropdown)</i>	74
<i>Grupo de botões</i>	75
5. CARDS	77
6. BARRAS DE PROGRESSO	81
7. CARROSEL DE IMAGENS.....	82
8. MODAL – CAIXAS DE DIÁLOGOS	86
9. POPOVERS E TOOLTIPS	88
10. MENUS: BARRAS DE NAVEGAÇÃO E ABAS	92
<i>Barra de navegação</i>	92
<i>List group</i>	95
<i>Offcanvas: menu lateral</i>	97
<i>Menu de rolagem (scrollspy)</i>	98
<i>Breadcrumb</i>	100
<i>Paginação</i>	101
BOOTSTRAP ICONS	103
CONCLUSÕES.....	107

Prefácio

Olá!

Bootstrap é um dos mais populares frameworks para construção de sites responsivos compatíveis tanto com computadores desktop, smartphones ou dispositivos com telas de diversos tamanhos.

Em sua quinta versão, Bootstrap traz uma série de novidades, como nomes de classes mais simplificados, novos recursos em JavaScript (jQuery não é mais necessário) e finalmente Bootstrap passou a ter uma biblioteca de ícones.

Neste manuscrito, vamos explorar algumas novidades do Bootstrap 5. É claro que este guia apenas aborda a superfície das funcionalidades e componentes do Bootstrap. Aqui, vamos explorar alguns dos recursos considerados mais importantes para iniciantes. É claro que nada se iguala a documentação oficial (<https://getbootstrap.com/docs>). Consulte-a sempre que achar necessário.

Bons estudos!



O framework CSS/JS mais querido da Web está mais poderoso do que nunca na sua 5^a versão. Pronto para adentrar no mundo de Bootstrap 5?

Introdução

Bootstrap é um dos mais populares frameworks para construção de sites responsivos compatíveis tanto com computadores desktop, smartphones ou dispositivos com telas de diversos tamanhos.

Em sua quinta versão, Bootstrap traz uma série de novidades, como nomes de classes mais simplificados, novos recursos em JavaScript (jQuery não é mais necessário) e finalmente Bootstrap passou a ter uma biblioteca de ícones: a Bootstrap icons.

Nas seções a seguir, vamos explorar algumas novidades do Bootstrap 5.

Template inicial

Vamos começar com a estrutura básica de um site feito com Bootstrap. Observe como ficaria o código fonte de uma página construída usando scripts armazenados em uma CDN:

```

1 <!doctype html>
2 <html lang="pt-BR">
3 <head>
4
5   <title>Bootstrap</title>
6
7   <meta charset="utf-8">
8   <meta name="viewport" content="width=device-width,
initial-scale=1, shrink-to-fit=no">
9

```

```
10      <!-- Bootstrap CSS -->
11      <link
12          href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-1BmE4kWBq78iYhFlqvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3" crossorigin="anonymous">
13      <link rel="stylesheet"
14          href="https://cdn.jsdelivr.net/npm/bootstrap-icons@1.8.1/font/bootstrap-icons.css">
15
16
17      <!-- Início do conteúdo -->
18
19      <!-- Fim do conteúdo -->
20
21      <!-- Bootstrap JavaScript -->
22      <script
23          src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js" integrity="sha384-ka7Sk0Gln4gmtz2MlQnikT1wXgYsOg+OMhuP+IlRH9sENB0LRn5q+8nbTov4+1p" crossorigin="anonymous"></script>
24
25 </body>
26 </html>
```

Para criarmos um site usando Bootstrap v5 são necessários três principais componentes:

- 1) Bootstrap CSS (linha 11)
- 2) Bootstrap icons (linha 12)
- 3) Bootstrap JavaScript Bundle (linha 22)

Opcionalmente, você pode baixar esses arquivos e executá-los localmente. Você pode ainda instalá-lo via npm com o comando:

npm install bootstrap

Cores

Antes de introduzirmos alguns conceitos mais complexos, precisamos apresentar um conceito que se estende para boa parte das classes do Bootstrap: as variações de classes de cores.

Observe a tabela a seguir:

Tabela 1. Cores no Bootstrap.

Palavra-chave (sufixo)	Cor	Exemplo
primary	Azul	Primary
secondary	Cinza	Secondary
success	Verde	Success
danger	Vermelho	Danger
warning	Laranja	Warning
info	Azul claro	Info
light	Branco (quase)	Light
dark	Preto (quase)	Dark

Fonte: adaptado de

<https://getbootstrap.com/docs/5.1/customize/color/>

Essas palavras-chave são usadas em muitas classes do Bootstrap. Por exemplo, se quiser colorir o fundo de laranja, podemos usar a classe *bg-warning* (sendo *bg* de *background* ou plano de fundo). Se quisermos alterar a cor de texto para verde, por exemplo, podemos usar *text-success*. A frente veremos outras classes.

Tamanhos

Outro padrão que devemos introduzir antes de falar de layout são as palavras-chave relacionadas a tamanhos. Observe a tabela a seguir:

Tabela 2. Sufixos de tamanho.

Tamanho	Sufixo da classe
Extra-pequeno (PP)	xs
Pequeno (P)	sm
Médio (M)	md
Grande (G)	lg
Extra-grande (GG)	xl
Extra extra-grande (GGG)	xxl

Fonte: adaptado de <https://getbootstrap.com/docs/5.1/>.

Para usar os sufixos de tamanho, em geral, basta informar o componente e adicionar o tamanho desejado separado por um hífen, como por exemplo, se quisermos um botão grande, podemos usar a classe btn-lg (nem todos os tamanhos estão disponíveis para todos os componentes).

Veremos alguns exemplos no próximo bloco.



Bootstrap lida com o layout de um website de maneira simples e elegante. Neste capítulo, você irá conhecer as principais classes relacionadas ao layout no Bootstrap.

Layout

Pontos de quebra (colapso)

Quando desenvolvemos sites responsivos, devemos prestar atenção nos chamados **Pontos de Interrupção** (*breakpoints*). Eles permitem controlar como o layout se adaptará em um determinado dispositivo. Por exemplo, não queremos que um layout planejado para ser exibido em uma tela Full-HD seja exibido da mesma forma em um celular: os elementos ficariam muito pequenos e o site não estaria legível. Por isso, o design responsivo visa oferecer ao cliente versões diferentes de um mesmo site dependendo do dispositivo que ele utiliza para acessar.

Bootstrap usa as chamadas *media queries* para aplicar estilos condicionalmente com base em um conjunto de parâmetros do navegador e do sistema operacional (um exemplo é a propriedade `min-width`). Segundo a documentação do Bootstrap, isso otimiza o código CSS, melhora o tempo de renderização e oferece uma ótima experiência para visitantes do site (<https://getbootstrap.com/docs/5.1/layout/breakpoints/>).

Tabela 3. Pontos de quebra do Bootstrap v5.

Breakpoint	Sufixo da classe	Dimensões
Extra-pequeno (PP)	-	<576px
Pequeno (P)	sm	≥576px
Médio (M)	md	≥768px
Grande (G)	lg	≥992px

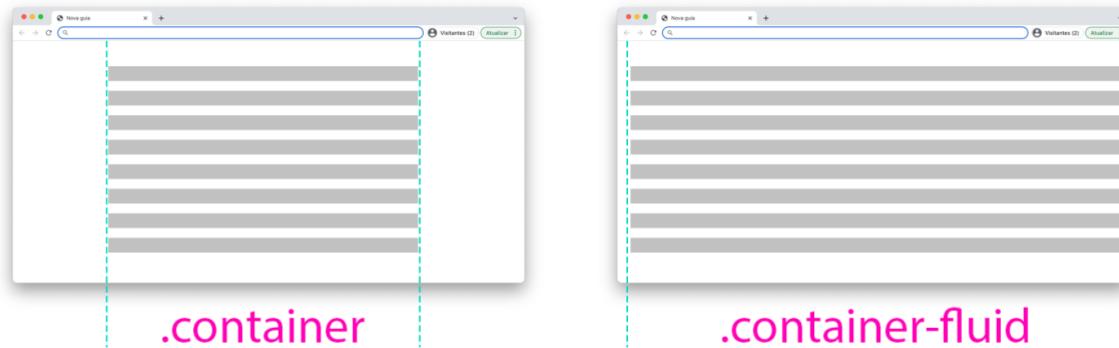
Extra-grande (GG)	xl	$\geq 1200\text{px}$
Extra extra-grande (GGG)	xxl	$\geq 1400\text{px}$

Fonte: traduzido de
[https://getbootstrap.com/docs/5.1/layout/breakpoints/.](https://getbootstrap.com/docs/5.1/layout/breakpoints/)

Contêineres de dados

Bootstrap permite delimitar o conteúdo em um contêiner que manterá as margens direita e esquerda homogêneas. Há basicamente três versões de classes contêiner: “.container”, “.container-fluid” e “container-*” (onde * pode ser um dos tipos sm, md, lg, xl ou xxl).

Figura 1. Classes container do Bootstrap 5.



Fonte: Wikicommons (CC-BY-SA-4.0). Disponível em
https://commons.wikimedia.org/wiki/File:Grid_system_Bootstrap_CSS_Framework.png. Acesso em 25 de mar. de 22.

A tabela a seguir apresenta o “ponto de colapso”, ou seja, quando uma classe container altera as margens do conteúdo com base nas especificações:

Tabela 4. Principais *tags* de contêiners.

Classe / tela	PP $<576\text{px}$	P $\geq 576\text{px}$	M $\geq 768\text{px}$	G $\geq 992\text{px}$	GG $\geq 1200\text{px}$	GGG $\geq 1400\text{px}$
<code>.container</code>	100%	540px	720px	960px	1140px	1320px
<code>.container-sm</code>	100%	540px	720px	960px	1140px	1320px
<code>.container-md</code>	100%	100%	720px	960px	1140px	1320px
<code>.container-lg</code>	100%	100%	100%	960px	1140px	1320px
<code>.container-xl</code>	100%	100%	100%	100%	1140px	1320px
<code>.container-xxl</code>	100%	100%	100%	100%	100%	1320px
<code>.container-fluid</code>	100%	100%	100%	100%	100%	100%

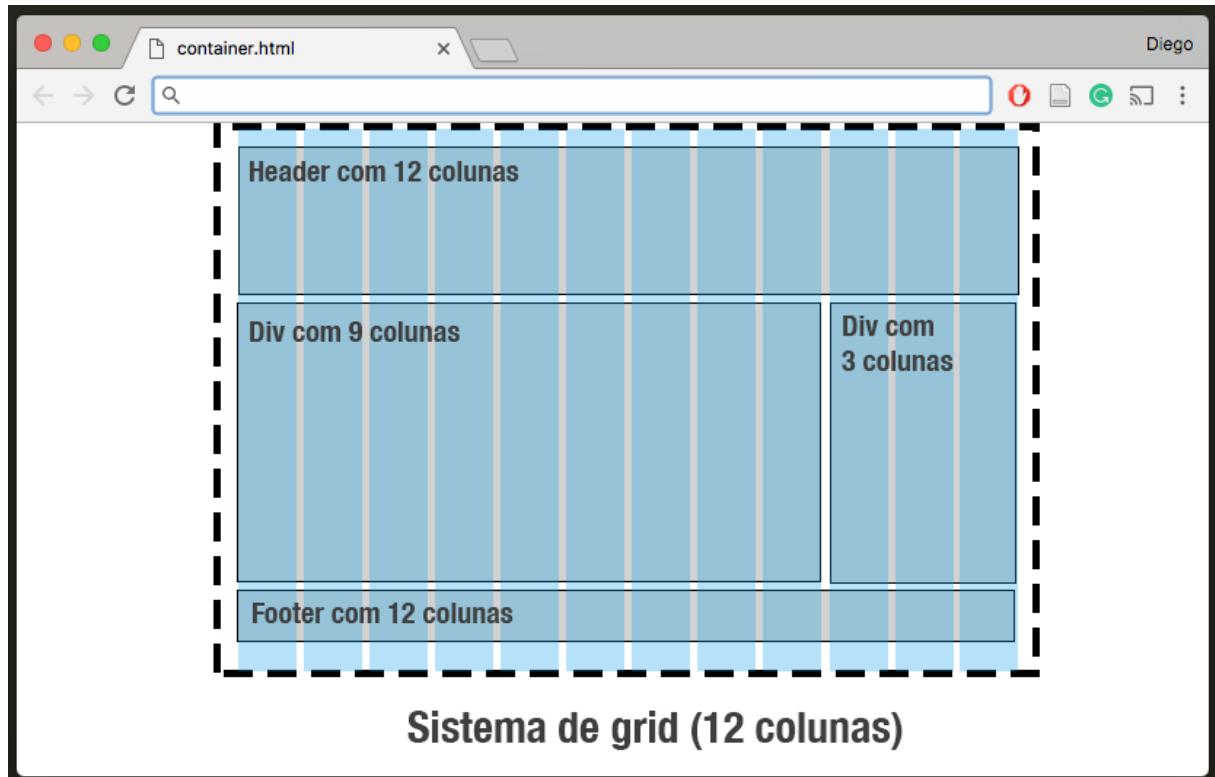
Fonte: traduzido de

<https://getbootstrap.com/docs/5.1/layout/containers/>.

Sistema de grade

O sistema de grade (*grid system*) do Bootstrap permite posicionar elementos com precisão. Para isso, Bootstrap divide a página em 12 colunas.

Tabela 5. Divisão em 12 colunas.



Fonte: Mariano & de Melo-Minardi (2017).

Ao estruturar uma página, devemos utilizar as seguintes classes:

- Contêiner (.container)
 - Linha (.row)
 - Coluna (.col-*)

Note que, neste caso, o * indica quantos espaços de colunas queremos alocar para determinado elemento (lembrando que o máximo é 12). Por exemplo, se quisermos dividir uma página em duas colunas, podemos alocar um tamanho 6 para cada coluna. Veja:

```

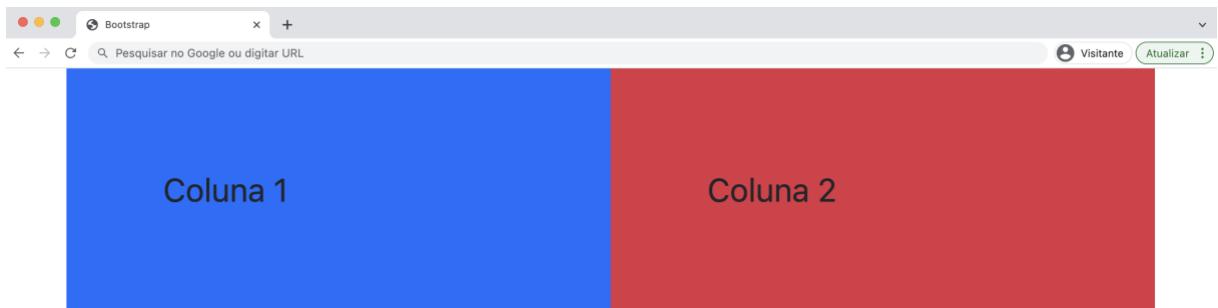
1 <div class="container">
2   <div class="row">
3     <div class="col-6 bg-primary p-5">Coluna 1</div>
4     <div class="col-6 bg-danger p-5">Coluna 2</div>

```

```
5    </div>
6 </div>
```

Note que declaramos um contêiner para delimitar o conteúdo, ou seja, uma linha e duas colunas dentro dessa mesma linha. Note que a soma dos valores das colunas dentro de uma linha deve ser igual a 12 (um valor acima disso provocaria uma quebra para uma outra linha, o que não é desejado). Não se preocupe com as classes *bg-primary* e *bg-danger*, pois elas foram incluídas apenas para inserir uma cor de fundo. O mesmo vale para a classe p-5, que foi incluída apenas para aumentar o espaçamento interno. Veja o resultado:

Figura 2. Duas colunas ocupando metade da página cada (classe col-6).



Fonte: próprio autor.

Note que Bootstrap divide cada elemento em 12 colunas. Logo, seria possível dividir a coluna 1 da figura em 12 outras partes.

Podemos definir um ponto de quebra para a coluna, informando os tamanhos desejados no meio da classe:

```
.col-[ponto-de-quebra]-[valor]
```

Por exemplo:

```
col-xl-6 col-sm-12
```

Essas duas classes aplicadas juntas indicam que, em uma tela grande (xl), o elemento deve ocupar metade do espaço. Já em uma tela pequena (sm), o elemento deve ocupar 100% do espaço disponível.

Distribuindo colunas

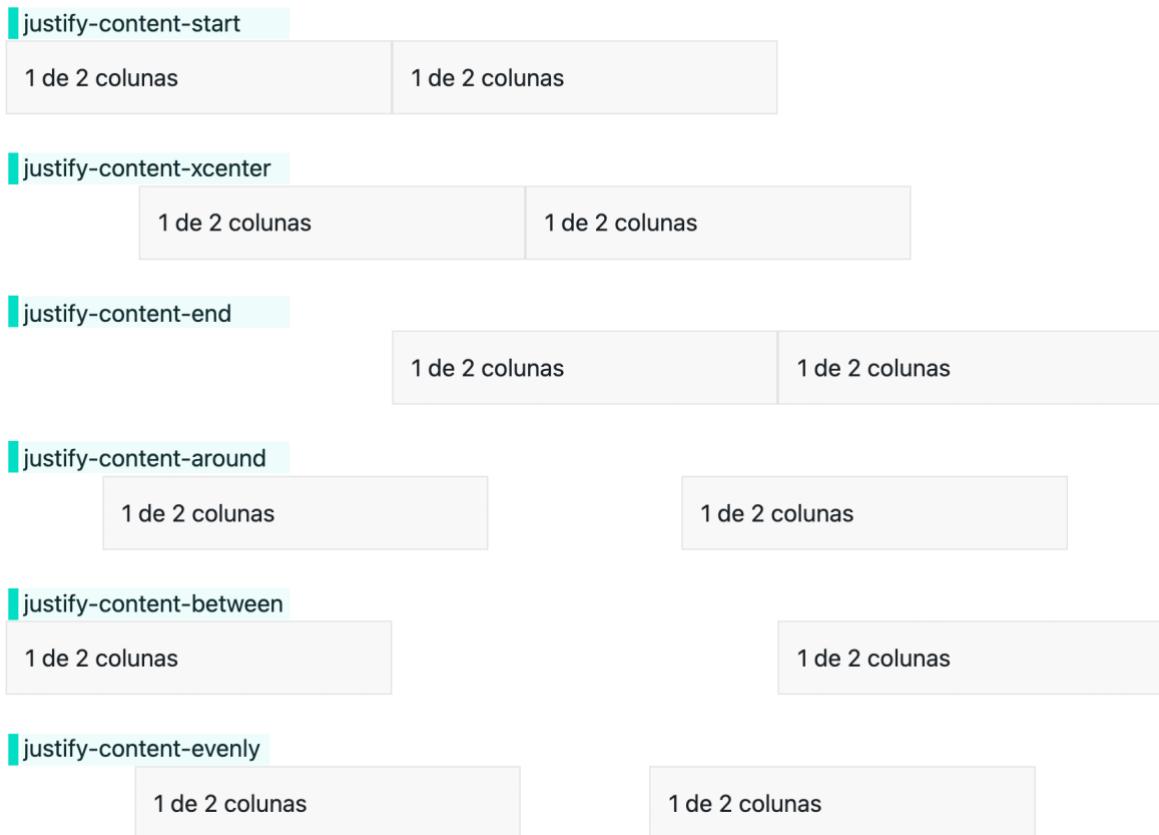
Digamos que desejamos adicionar colunas cujos valores de soma são inferiores a 12. Nesse caso, teríamos espaços remanescentes no final da linha. Caso desejemos distribuir essas colunas, podemos utilizar as classes de alinhamento.

Para isso, podemos distribuir as colunas usando, junto com a classe *row*, uma das seguintes classes:

- *justify-content-start*: alinha à esquerda (padrão);
- *justify-content-center*: centraliza;
- *justify-content-end*: alinha à direita;
- *justify-content-around*: margens à direita e à esquerda iguais;
- *justify-content-between*: alinha cada item em uma ponta (quando há dois itens);
- *justify-content-evenly*: parecido com o **-around*.

Observe como ficaria uma linha com duas colunas de tamanho col-4:

Figura 3. Colunas.



Fonte: Wikicommons (CC-BY-SA-4.0). Disponível em https://commons.wikimedia.org/wiki/File:Classes_bootstrap_alinhamento_o.png. Acesso em 25 de mar. de 22.

O código para gerar uma figura similar ficaria assim:

```
1 <div class="container">
2   <div class="row justify-content-*">
3     <div class="col-4">
4       1 de 2 colunas
5     </div>
6     <div class="col-4">
```

```
7      1 de 2 colunas
8      </div>
9      </div>
10 </div>
```

Substitua o * pela classe desejada.

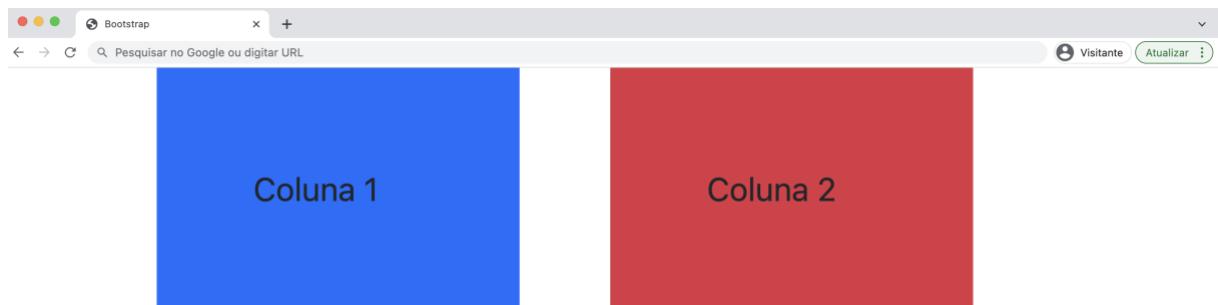
Classes offset

Podemos ainda adicionar o espaçamento antes de uma div usando as classes offset. No exemplo a seguir, vamos adicionar um desvio de 1 coluna (offset-1).

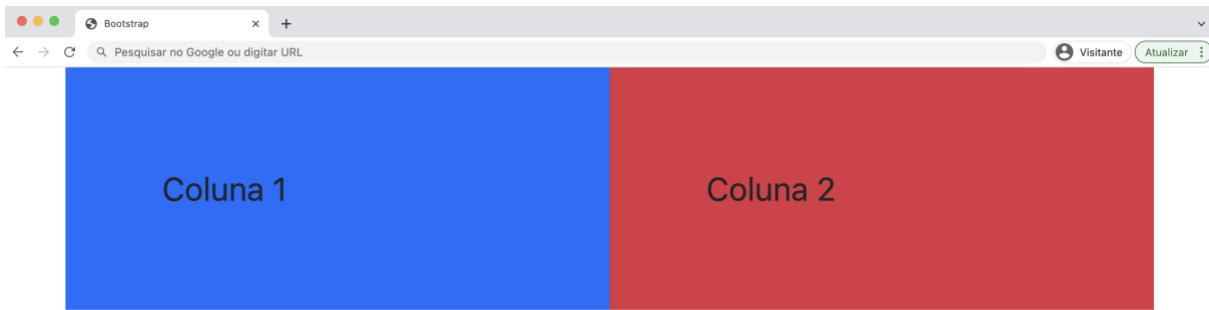
```
1 <div class="container">
2   <div class="row">
3     <div class="offset-1 col-4 bg-primary p-5">Coluna 1</div>
4     <div class="offset-1 col-4 bg-danger p-5">Coluna 2</div>
5   </div>
6 </div>
```

Observe o resultado:

Figura 4. Classe offset aplicada.



(A) Com offset (classes offset-1 col-4)



(B) Sem offset (class col-6)

Fonte: próprio autor.

As classes offset também suportam breakpoint, ou seja, pode-se usar offset-md-6, offset-sm-12, etc.

Calhas (*gutters*)

Calhas (*gutters*) representam o espaçamento entre colunas no sistema de grade. Por padrão, o espaçamento que separa colunas no Bootstrap 5 é de 1.5rem ou 24px. Entretanto, podemos alterar esses valores padrão usando as propriedades **g-***, sendo * um número que vai de 0 a 5, onde 0 representa uma remoção completa de margens, 3 o tamanho de margem padrão e 5 uma margem 3x maior (consulte a seção de margens e preenchimento para compreender como essa numeração funciona).

Por exemplo, vamos testar a margem máxima entre itens:

```

1 <div class="container">
2   <div class="row g-5">
3
4     <div class="col-6"><div class="bg-primary">1</div></div>
5     <div class="col-6"><div class="bg-danger">2</div></div>
6     <div class="col-6"><div class="bg-warning">3</div></div>
7     <div class="col-6"><div class="bg-secondary">4</div></div>
8     <div class="col-6"><div class="bg-success">5</div></div>
9     <div class="col-6"><div class="bg-info">6</div></div>

```

```
10  
11      </div>  
12  
13 </div>
```

Observe o resultado:

Figura 5. Calha de tamanho g-5.



Fonte: próprio autor.

Note que apenas os espaçamentos internos são alterados. A margem para o topo do primeiro item não pode ser modificada pela classe g-5.

Observe como podemos reduzir as calhas e ainda adicionar uma margem para o topo:

```
1 <div class="container">  
2   <div class="row g-2 mt-2">  
3  
4     <div class="col-6"><div class="bg-primary">1</div></div>  
5     <div class="col-6"><div class="bg-danger">2</div></div>  
6     <div class="col-6"><div class="bg-warning">3</div></div>  
7     <div class="col-6"><div class="bg-secondary">4</div></div>  
8     <div class="col-6"><div class="bg-success">5</div></div>
```

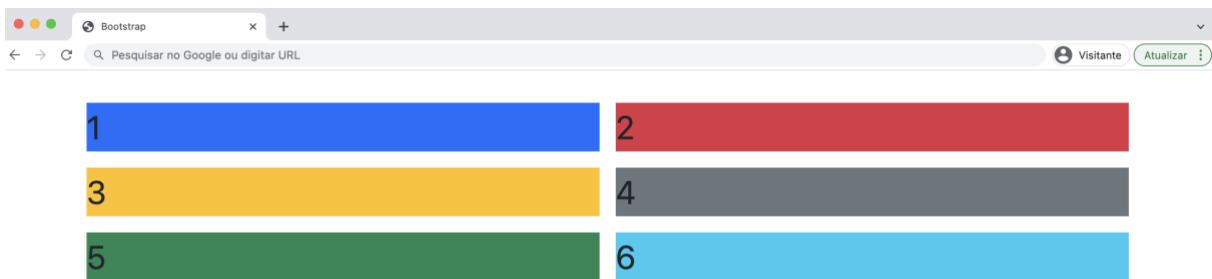
```

9   <div class="col-6"><div class="bg-info">6</div></div>
10
11   </div>
12 </div>

```

Para reduzir as calhas usamos g-2. Além disso, usamos mt-2, que aumenta a margem para o topo. Observe o resultado:

Figura 6. Calha de tamanho g-2.



Fonte: próprio autor.

Podemos ainda aplicar valores diferentes para calhas horizontais e calhas verticais. Para isso basta usar as classes gx-* e gy-*, e assim, alterar os valores de margem horizontal, eixo x, e margem vertical, eixo-y.

As variantes x e y podem ser usadas em outros contextos. Veremos isso na seção de margem e preenchimento.

Margem e preenchimento

No *box model*, um elemento HTML pode receber uma borda, além de margens internas (preenchimento ou *padding*) e margens externas (chamado apenas de margem ou *margin*). Bootstrap 5 trouxe uma maneira eficiente para lidar com espaçamentos. Para isso, usamos a seguinte sintaxe:

[tipo][posição]-[nível]

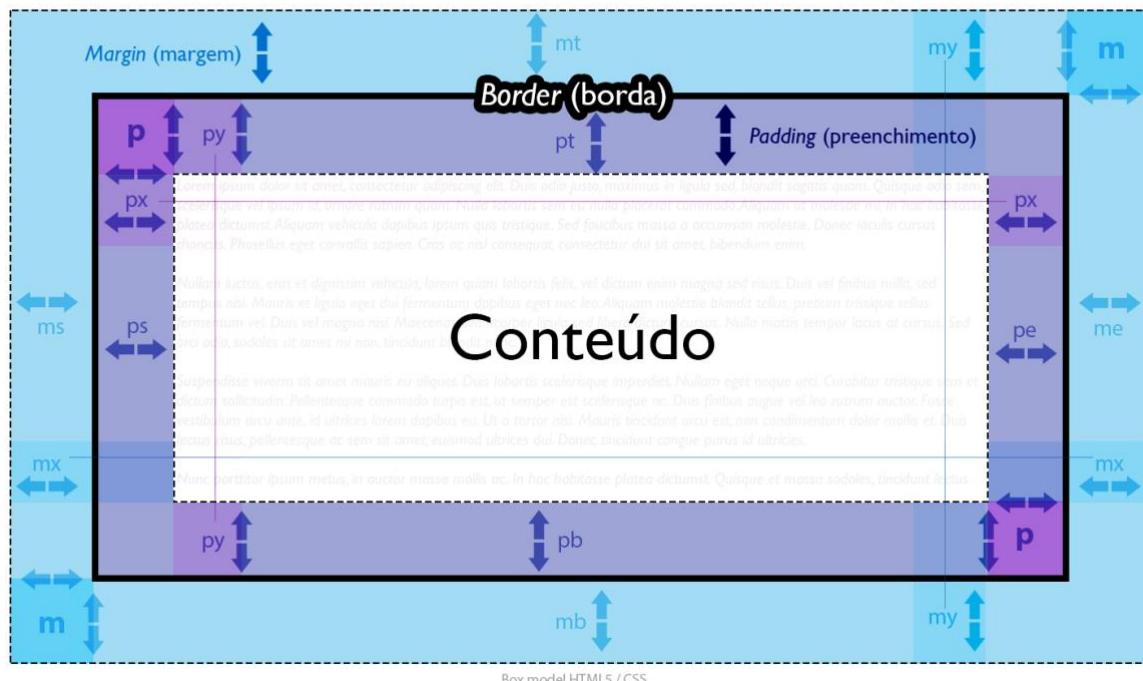
Onde tipo pode ser “p” ou “m”; posição pode ser “t” (top ou topo), “e” (end ou final), “b” (bottom ou rodapé), “s” (start ou início), “x” (eixo x: esquerda e direita), “y” (eixo y: topo e rodapé), ou ainda, pode-se deixar este valor em branco para aplicar aos quatro lados; por fim, nível pode ser 0, 1, 2, 3, 4 ou 5. Veja a figura:

Figura 7. Classes do Bootstrap para lidar com espaçamento.

Classes Bootstrap

Exemplos: `class="m-5"` (todas as margens nível 5), `mx-0` (margem esquerda e direita nível 0), `ms-1` (margem inicial nível 1), `p-2` (preenchimento nível 2), `pt-3` (preenchimento topo nível 3), `pe-0` (preenchimento final nível 0) e assim por diante.

Sintaxe: [tipo][posição]-[nível]
Tipo: p, m
Posição: t, e, b, s, x, y (opcional)
Nível: 0, 1, 2, 3, 4, 5



Fonte: Wikicommons (CC-BY 4.0). Disponível em https://commons.wikimedia.org/wiki/File:Classes_usadas_para_espa%C3%A7amento_no_Framework_Bootstrap.png. Acesso em 26 de março de 2022.

Em resumo, no Bootstrap podemos alterar a margem usando classes com a letra “m” e o preenchimento usando a letra “p”. Opcionalmente, podemos definir quais as posições queremos alterar. O nome da classe deve receber, separado por um hífen, um valor numérico, que vai de 0 a 5. A tabela a seguir ilustra esse valor numérico:

Tabela 6. Valores de margem e preenchimento.

Valor	Cálculo	Classe-exemplo	Descrição exemplo
0	Espaçamento-padrão x 0	m-0	Margem de 0px
1	Espaçamento-padrão x 0,25	pl-1	Preenchimento esquerdo (<i>left</i>) reduzido para $\frac{1}{4}$ do tamanho padrão.
2	Espaçamento-padrão x 0,5	my-1	Margem superior e inferior (eixo y) reduzida pela metade do tamanho padrão.
3	Espaçamento-padrão x 1	pt-1	Preenchimento superior (<i>top</i>) com valor padrão.
4	Espaçamento-padrão x 1,5	px-4	Preenchimento esquerdo e direito (eixo x) 1,5x maior que o tamanho padrão.
5	Espaçamento-padrão x 3	mb-5	Margem inferior (<i>bottom</i>) triplicada.
auto	Margem definida automaticamente	p-auto	Preenchimento automático.

Fonte: adaptado de

<https://getbootstrap.com/docs/5.1/utilities/spacing/>

Posicionamento fixo e visibilidade

Podemos fixar um elemento no topo ou no rodapé usando as classes “fixed-top” e “fixed-bottom”. Além disso, podemos esconder um elemento usando a classe “visually-hidden”. Observe o exemplo:

```
1 <div class="bg-warning col-2 fixed-top p-5">Fixo no
topo</div>
2 <div class="bg-success col-2 fixed-bottom p-5">Fixo no
rodapé</div>
3 <div class="bg-danger col-3 visually-hidden p-5">Texto
escondido</div>
```

Esse código irá produzir a seguinte interface:

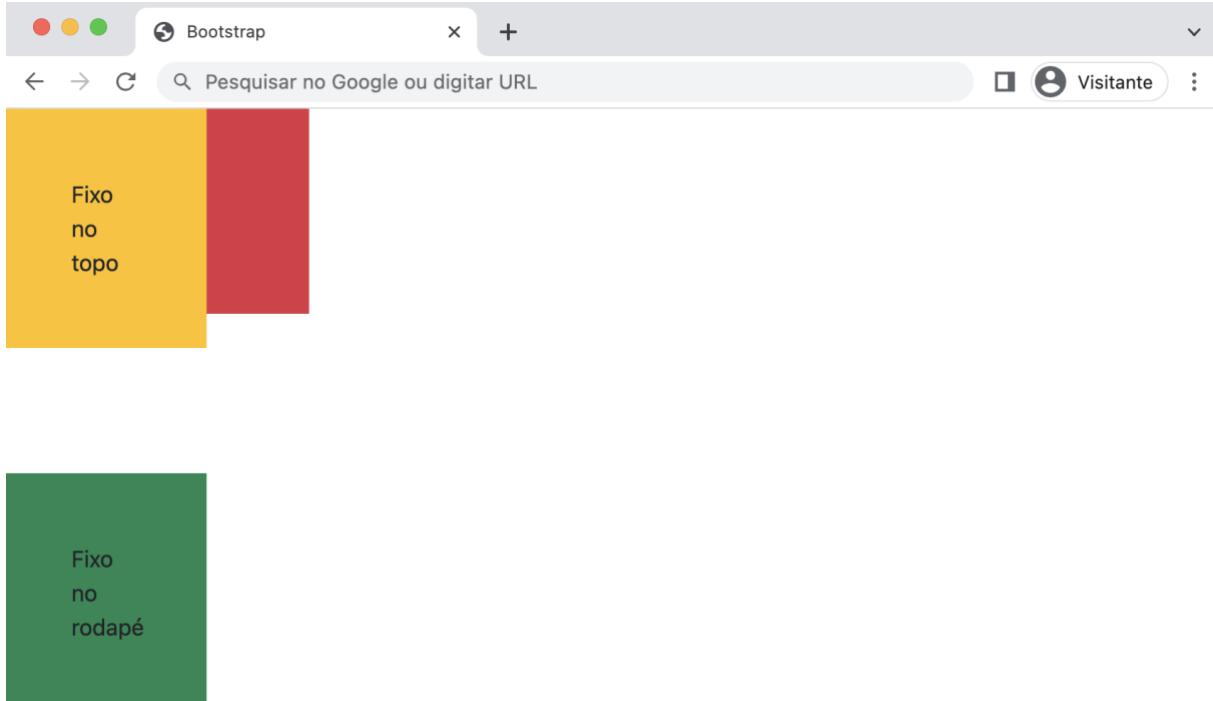
Figura 8. Classes fixed-top e fixed-bottom.



Fonte: próprio autor.

Note que o texto escondido não aparece. Tente remover a classe *visually-hidden* e você verá um fragmento da div logo abaixo da div fixada no topo:

Figura 9. A classe fixed-top sobrepõe itens. Portanto, deve-se considerar o espaço que ela ocupa na hora de construir os seus códigos.



Fonte: próprio autor.

Note que podemos alterar a visibilidade com as classes “*visible*” e “*invisible*”.

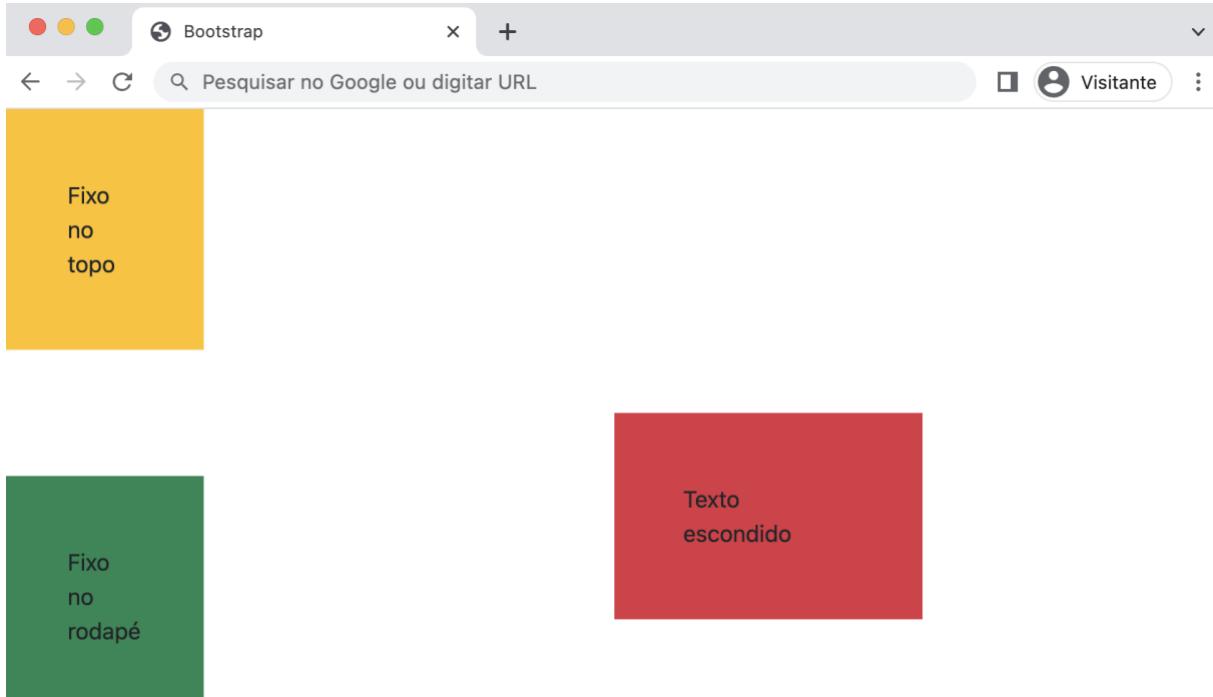
Além disso, podemos ainda alterar a posição com as classes “position-*”, onde * pode ser absolute, relative, fixed, static e stick. Ao usar a classe “position-absolute” podemos combinar isso com as propriedades *top* (superior), *end* (direita), *bottom* (inferior) e *start* (esquerda), seguido de uma propriedade numérica que indica a posição (0 indica começo da página, 50 indica meio e 100 indica final). Observe um exemplo:

```

1 <div class="bg-warning col-2 fixed-top p-5">Fixo no topo</div>
2 <div class="bg-success col-2 fixed-bottom p-5">Fixo no rodapé</div>
3 <div class="bg-danger col-3 p-5 position-absolute start-50 top-50">
4     Texto Escondido
5 </div>
```

Observe o resultado:

Figura 10. Classes start-50 e top-50.



Fonte: próprio autor.

Perceba que usamos start-50 e top-50 para posicionar um item no meio exato da tela. Mesmo assim, a div não parece centralizada. Isso acontece pois o alinhamento é feito com base na ponta superior esquerda. Podemos corrigir o alinhamento apenas adicionando a classe “translate-middle”:

```
1 <div class="bg-danger col-3 p-5 position-absolute start-50 top-50 translate-middle">
2     Texto escondido
3 </div>
```

Agora nosso item está corretamente centralizado:

Figura 11. Centralizando com translate-middle.



Fonte: próprio autor.

Poderíamos alinhar à direita usando a classe “end-0” e à esquerda com “start-0”. Podemos centralizar itens com “top-50” ou “bottom-50” (use a classe “translate-middle-y” ou “translate-middle-x” para corrigir o alinhamento horizontal e vertical). Observe um exemplo:

```

1 <div class="bg-warning col-2 fixed-top p-5 position-absolute
start-0 top-50 translate-middle-y">Fixo no topo</div>

2 <div class="bg-success col-2 p-5 position-absolute bottom-0
end-0">Fixo no rodapé</div>

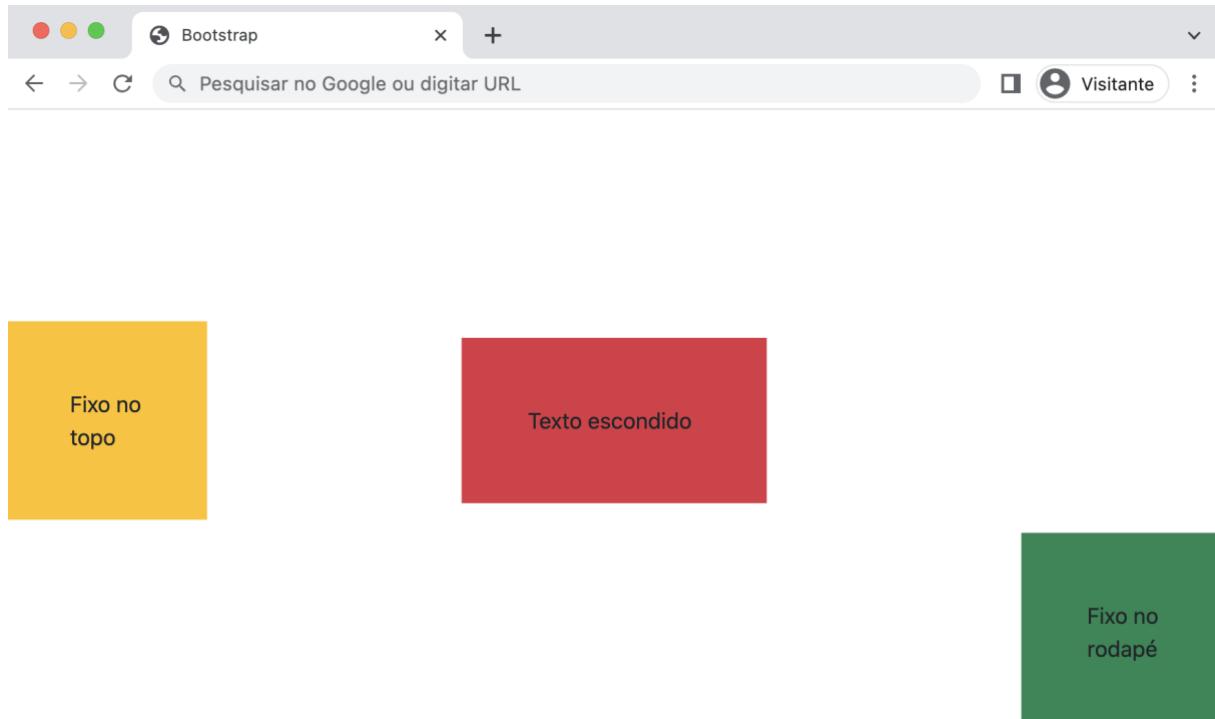
3 <div class="bg-danger col-3 p-5 position-absolute start-50
top-50 translate-middle">

4     Texto escondido

5 </div>

```

Figura 12. A div laranja foi alinhada ao centro do eixo y (classes “top-50” e “translate-middle-y”) e no começo (classe “start-0”) e a verde foi alinhada no canto inferior direito (classes “bottom-0” e “end-0”).



Fonte: próprio autor.

Por fim, podemos alterar o posicionamento com as classes “float-start” (alinhar à esquerda) ou “float-end” (alinhar à direita). Caso tenha problemas ao usar essas classes, experimente aplicar a classe “clearfix” ao próximo elemento. Perceba que há muitas formas de fazer a mesma coisa usando classes diferentes. A forma que você irá utilizar dependerá do seu gosto pessoal.

Altura e largura

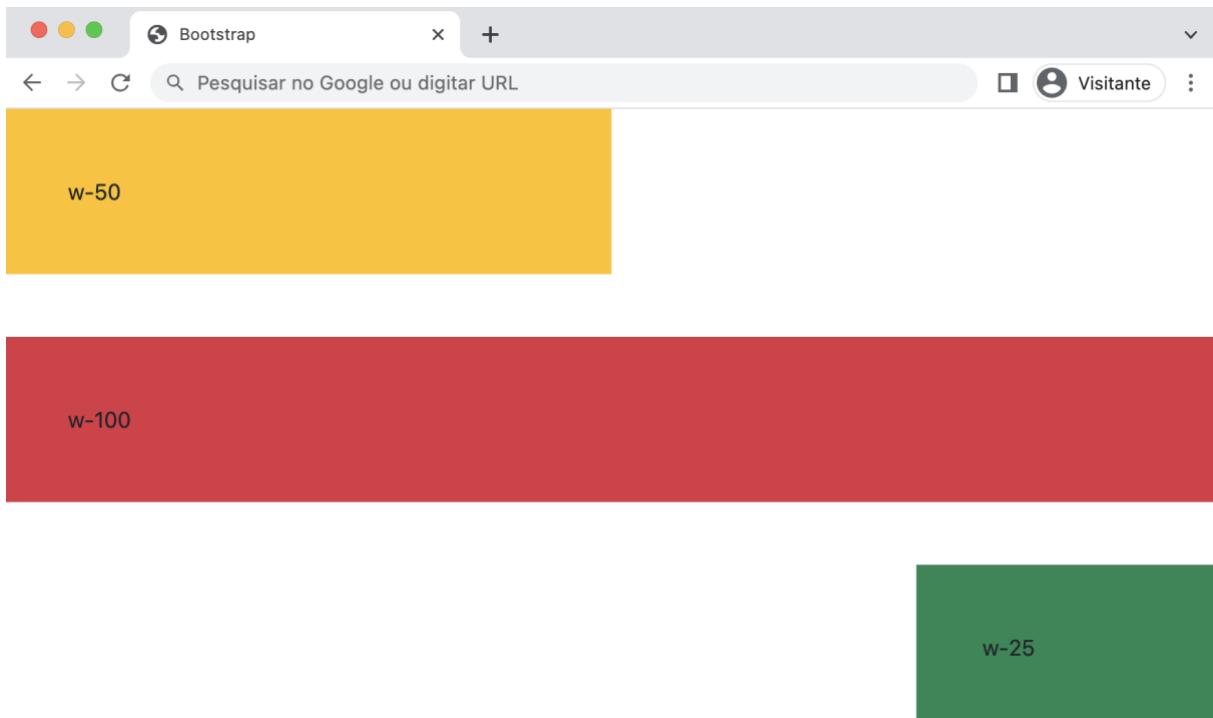
Podemos alterar a largura e a altura de um elemento usando as classes “w-*” e “h-*”, respectivamente, onde * representa o percentual de tamanho e pode ser 0, 25, 50, 75, 100 ou auto. Por exemplo:

```
1 <div class="bg-warning fixed-top p-5 w-50">w-50</div>
2 <div class="bg-success p-5 position-absolute bottom-0 end-0
w-25">
3   w-25
```

```
4 </div>
5 <div class="bg-danger p-5 position-absolute start-50 top-50
6     translate-middle w-100"> w-100
7 </div>
```

Este código gera os seguintes tamanhos:

Figura 13. Larguras w-50, w-100 e w-25.



Fonte: próprio autor.

Agora vamos brincar um pouco com a altura:

```
1 <div class="bg-warning fixed-top p-5 h-50 w-50">h-50</div>
2 <div class="bg-success p-5 position-absolute bottom-0 end-0
w-100 h-100">
3     w-25
4 </div>
5 <div class="bg-danger p-5 position-absolute start-50 top-50
```

```
6          translate-middle w-25 h-100">
7      w-100
8 </div>
```

Observe o resultado:

Figura 14. As classes de altura podem ser h-0, h-25, h-50, h-75 ou h-100.



Fonte: próprio autor.

Sombras e opacidade

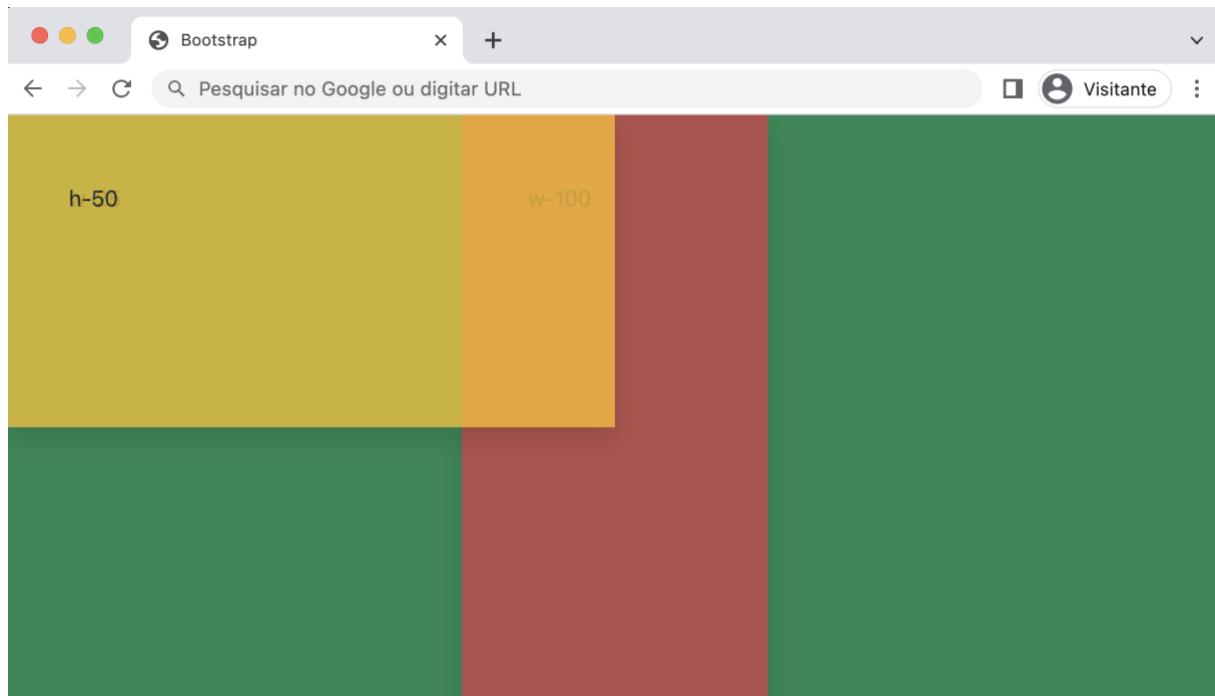
Podemos alterar a opacidade de um item usando as classes “opacity-*”, onde * indica o percentual (0, 25, 50, 75 ou 100). Além disso, vamos aproveitar este exemplo e testar a adição de sombras a elementos. Podemos incluir sombras usando a classe “shadow”. Veja:

```
1 <div class="bg-warning fixed-top p-5 h-50 w-50 shadow
opacity-75">
2     h-50
3 </div>
```

```
4 <div class="bg-success p-5 position-absolute bottom-0 end-0
5           w-100 h-100 shadow">
6           w-25
7 </div>
8 <div class="bg-danger p-5 position-absolute start-50 top-50
9           translate-middle w-25 h-100 opacity-75 shadow">
10          w-100
11 </div>
```

Observe o resultado:

Figura 15. As divs laranja e vermelha receberam uma opacidade de 50% (classe “opacity-50”), além de sombra (classe “shadow”).



Fonte: próprio autor.



No desenvolvimento front-end com Bootstrap, o conteúdo se refere a elementos HTML usualmente usados. Bootstrap possui classes para estilizá-los.

Conteúdo

Tipografia

A tipografia consiste nos aspectos que permeiam as fontes em um site. Bootstrap possui uma tipografia requintada que é aplicada automaticamente ao seu website.

Para aplicar alterações nas propriedades padrão, podemos usar as classes “text-*”, sendo * referente ao alinhamento ou à cor. Veja:

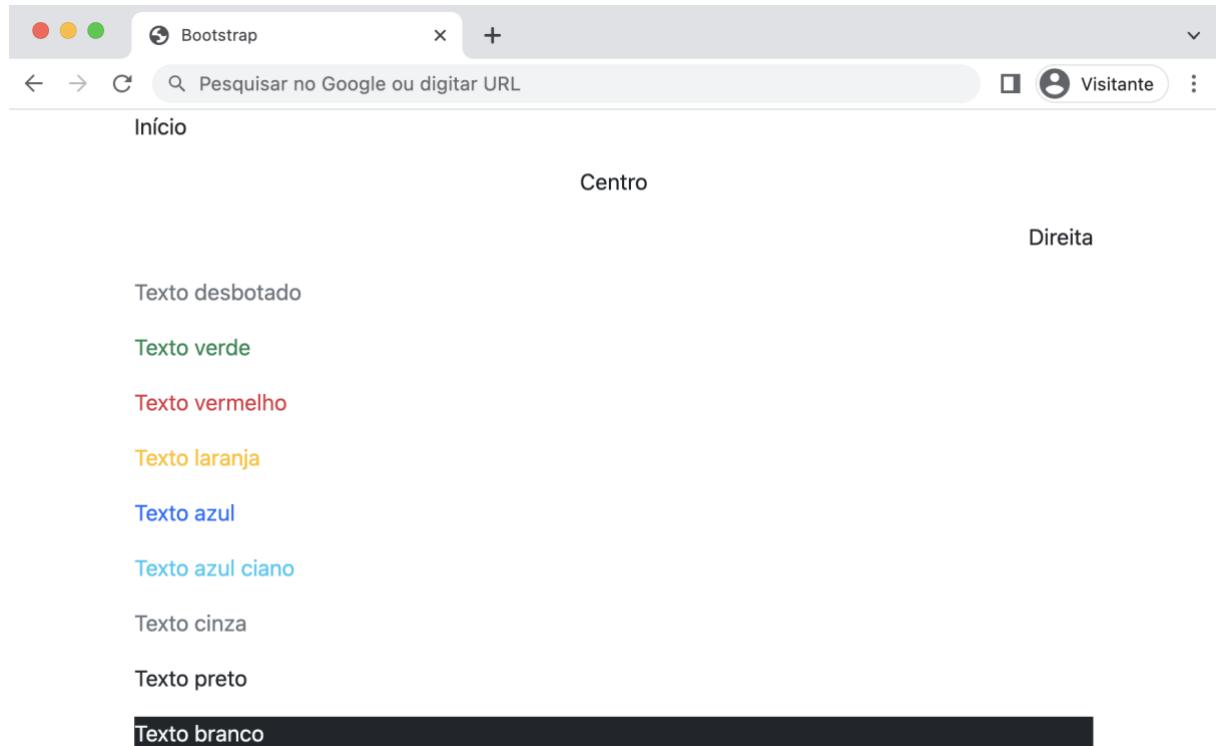
```

1 <div class="container">
2
3     <!-- posição -->
4     <p class="text-start">Início</p>
5     <p class="text-center">Centro</p>
6     <p class="text-end">Direita</p>
7
8     <!-- cor -->
9     <p class="text-muted">Texto desbotado</p>
10    <p class="text-success">Texto verde</p>
11    <p class="text-danger">Texto vermelho</p>
12    <p class="text-warning">Texto laranja</p>
```

```
13    <p class="text-primary">Texto azul</p>
14    <p class="text-info">Texto azul ciano</p>
15    <p class="text-secondary">Texto cinza</p>
16    <p class="text-dark">Texto preto</p>
17    <p class="text-light bg-dark">Texto branco</p>
18
19 </div>
```

Observe o resultado:

Figura 16. Propriedades de texto.



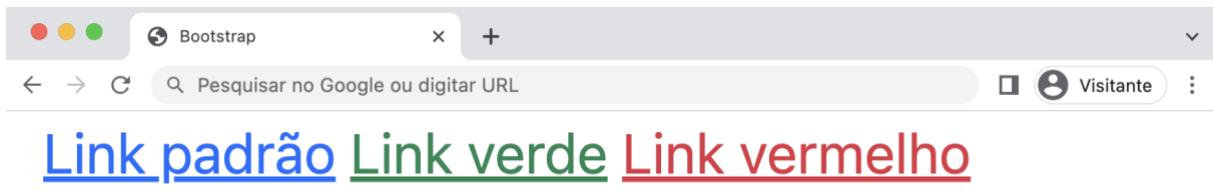
Fonte: próprio autor.

Caso queira aplicar essas propriedades de cores a links, troque “text-*” por “link-*”:

```
1 <a href="#">Link padrão</a>
2 <a href="#" class="link-success">Link verde</a>
3 <a href="#" class="link-danger">Link vermelho</a>
```

Compare as diferenças:

Figura 17. Links coloridos no Bootstrap.



Fonte: próprio autor.

Tamanhos de textos

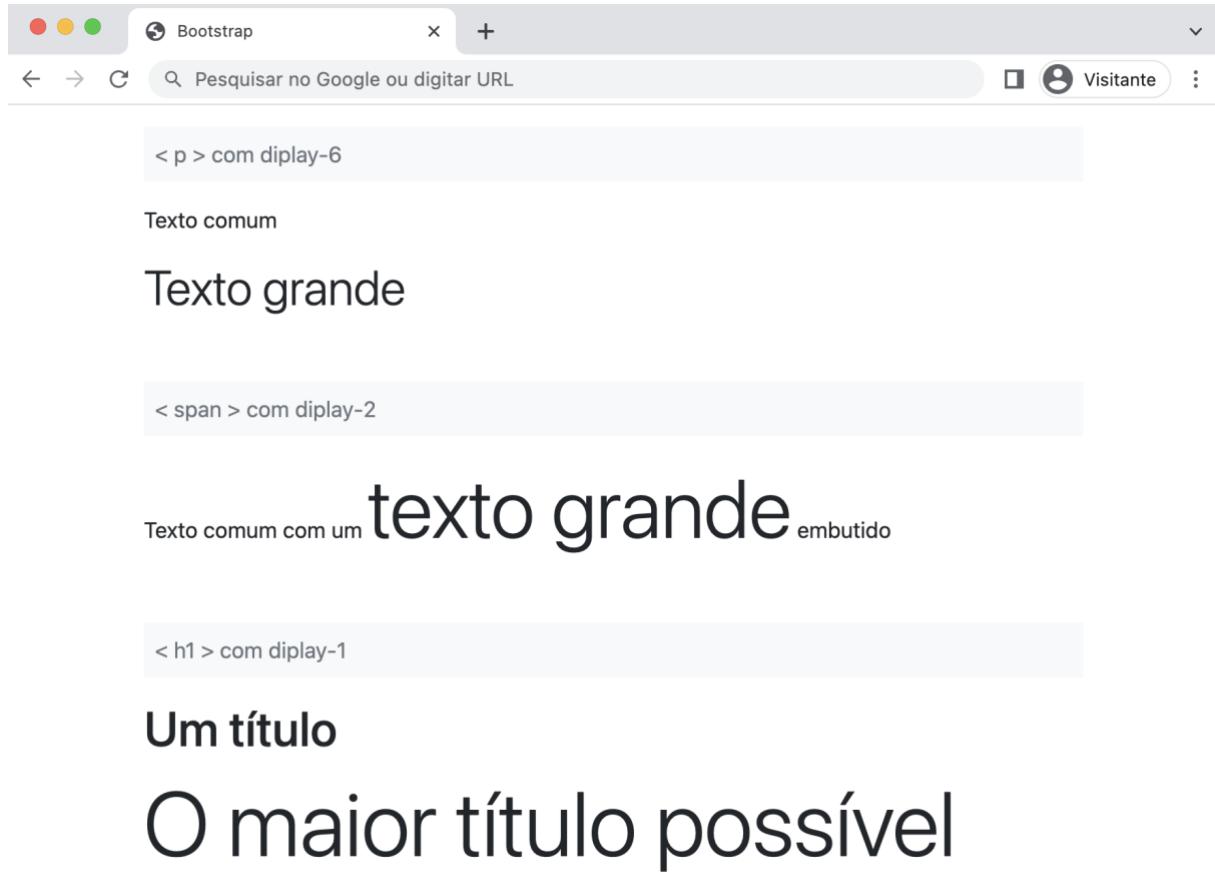
Se quisermos aumentar o tamanho de uma fonte, podemos usar as classes “display-*”, onde * representa um número que pode ser 0, 1, 2, 3, 4, 5 ou 6. Veja que podemos aplicar isso em simples tags de parágrafo (`<p>`), caixas delimitadoras de texto (``) ou até mesmo em títulos (`<h1>`). Veja:

```
1 <!-- < p > com display-6 -->
2 <p class="text-muted bg-light p-2">< p > com display-6</p>
3 <p>Texto comum</p>
4 <p class="display-6">Texto grande</p>
5
6 <!-- < span > com display-2 -->
7 <p class="text-muted bg-light p-2 mt-5">< span > com display-2</p>
8 <p>
9   Texto comum com um <span class="display-2">texto grande</span>
10 embutido </p>
11
```

```
12 <!-- < h1 > com diplay-1 -->
13 <p class="text-muted bg-light p-2 mt-5">< h1 > com diplay-1</p>
14 <h1>Um título</h1>
15 <h1 class="display-1">O maior título possível</h1>
```

Observe o resultado:

Figura 18. Tamanho de texto alterado com as classes “display-*”.

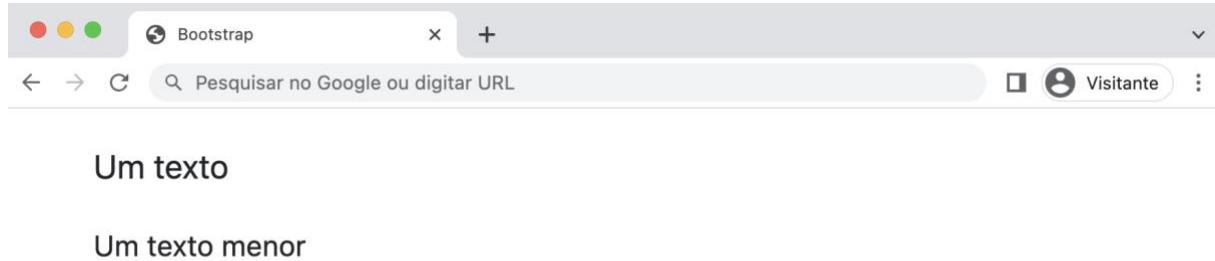


Fonte: próprio autor.

Já para reduzir o tamanho de um texto, utilize a classe “small”.

```
1 <p>Um texto</p>
2 <p class="small">Um texto menor</p>
```

Figura 19. A classe small reduz o tamanho da fonte.

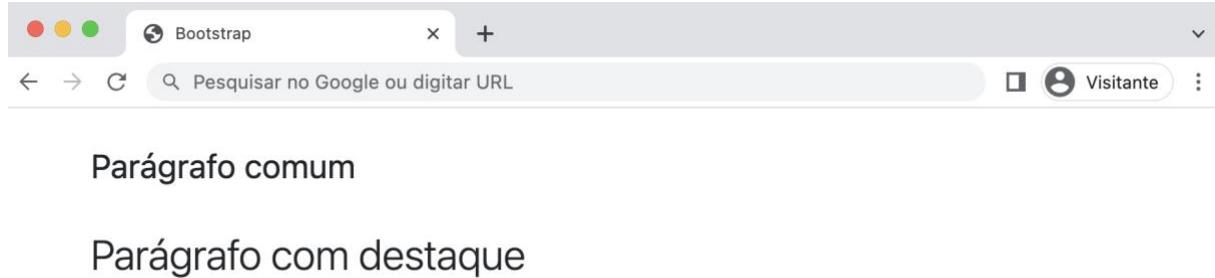


Fonte: próprio autor.

Podemos ainda adicionar destaque a um texto usando a classe “lead”. Observe:

```
1 <p>Parágrafo comum</p>
2 <p class="lead">Parágrafo com destaque</p>
```

Figura 20. A classe “lead” dá destaque ao texto.



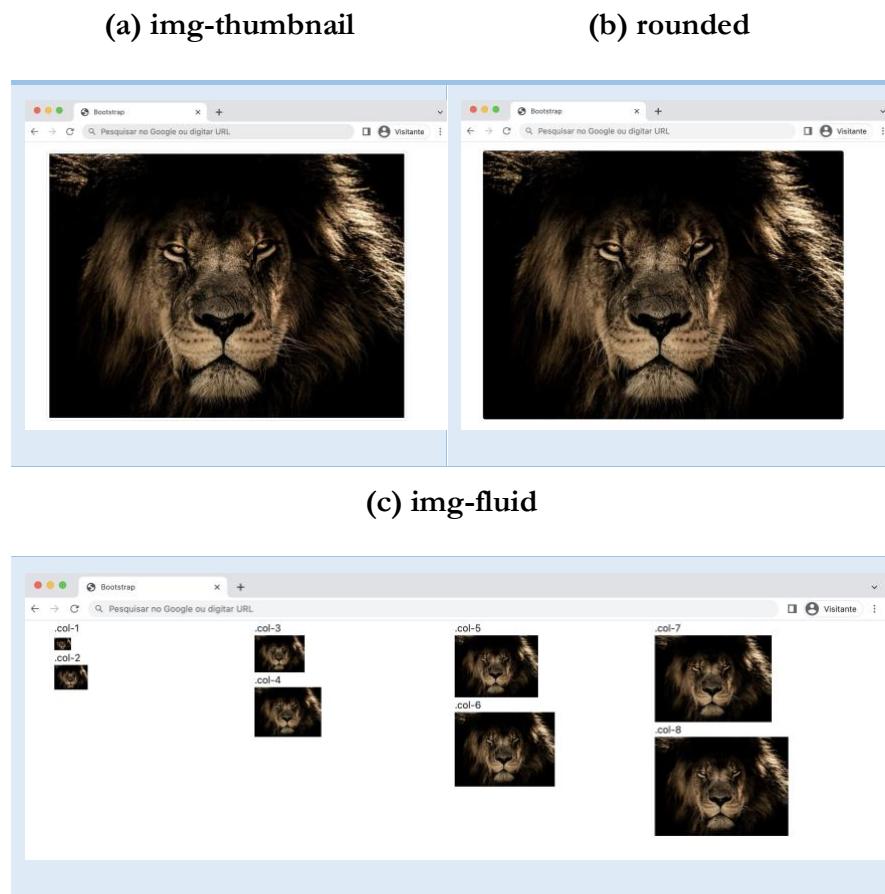
Fonte: próprio autor.

Imagens e figuras

Bootstrap fornece uma série de classes para estilizar figuras. São exemplos dessas classes:

- **img-thumbnail**: adiciona uma moldura à figura;
- **rounded**: adiciona bordas arredondadas;
- **img-fluid**: imagem responsiva (ocupa todo o espaço).

Figura 21. Estilizando imagens com Bootstrap. (A) classe img-thumbnail (miniatura) adiciona uma moldura ao redor da figura; (b) a classe rounded inclui bordas redondas na figura; (c) já img-fluid deixa a imagem responsiva, isto é, ela terá 100% do espaçamento do elemento HTML que ela ocupa, logo se ela está disposta em uma coluna de tamanho col-1, ela será pequena, mas se estiver em uma coluna maior, como col-8, ela ficará maior.



Fonte: Imagem de Ian Lindsay por Pixabay (disponível em <https://pixabay.com/pt/photos/le%c3%a3o-africano-gato-selvagem-felino-2888519/>).

Observe o código que gera as figuras acima:

```

1 <!-- Imagem responsive: img-fluid -->
2 <div class="container">
3   <div class="w-25 float-start".col-1
4     <div class="col-1">
5       

```

```
6      </div>
7
8      <div class="col-2">.col-2
9          
10     </div>
11
12     <div class="w-25 float-start">
13         <div class="col-3">.col-3
14             
15         </div>
16         <div class="col-4">.col-4
17             
18         </div>
19
20     <div class="w-25 float-start">
21         <div class="col-5">.col-5
22             
23         </div>
24         <div class="col-6">.col-6
25             
26         </div>
27
28     <div class="w-25 float-start">
29         <div class="col-7">.col-7
30             
31         </div>
32         <div class="col-8">.col-8
33             
34     </div>
```

```
35 </div>
36
37 <!-- Moldura: img-thumbnail -->
38 
39
40 <!-- Bordas redondas: rounded -->
41 
```

Para posicionar imagens podemos usar as classes:

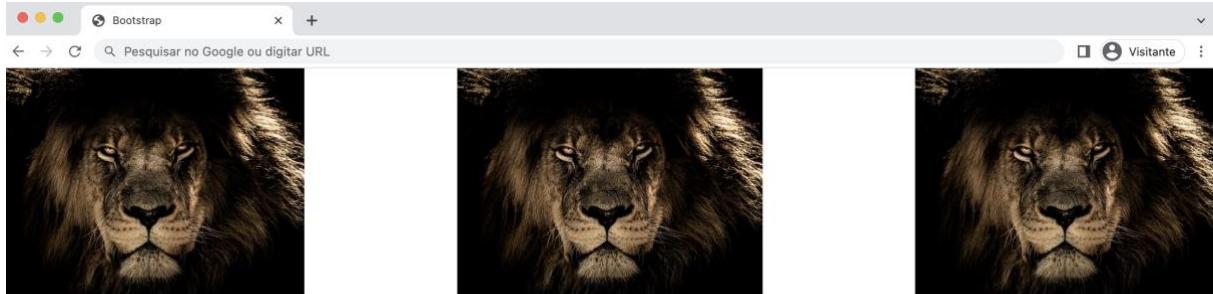
- Esquerda: float-start
- Direita: float-end
- Centro: mx-auto d-block

Veja como fica o código:

```
1 <!-- esquerda -->
2 
3
4 <!-- direta -->
5 
6
7 <!-- centro-->
8 
```

O resultado será:

Figura 22. Alinhando à esquerda (float-start), à direita (float-end) e ao centro (d-block mx-auto).



Fonte: Imagem de Ian Lindsay por Pixabay (disponível em <https://pixabay.com/pt/photos/le%c3%a3o-africano-gato-selvagem-felino-2888519/>).

Agora, digamos que você queira vincular sua figura a uma determinada legenda. Podemos fazer isso usando a tag <figure> como contêiner para imagem e texto, além da tag <figcaption> para legenda. Além disso, vamos precisar aplicar três classes:

- <figure> deve receber a classe “figure”;
- deve receber a classe “figure-img w-75 rounded”;
- <figcaption> deve receber a classe “figure-caption”.

Observe como ficaria o código:

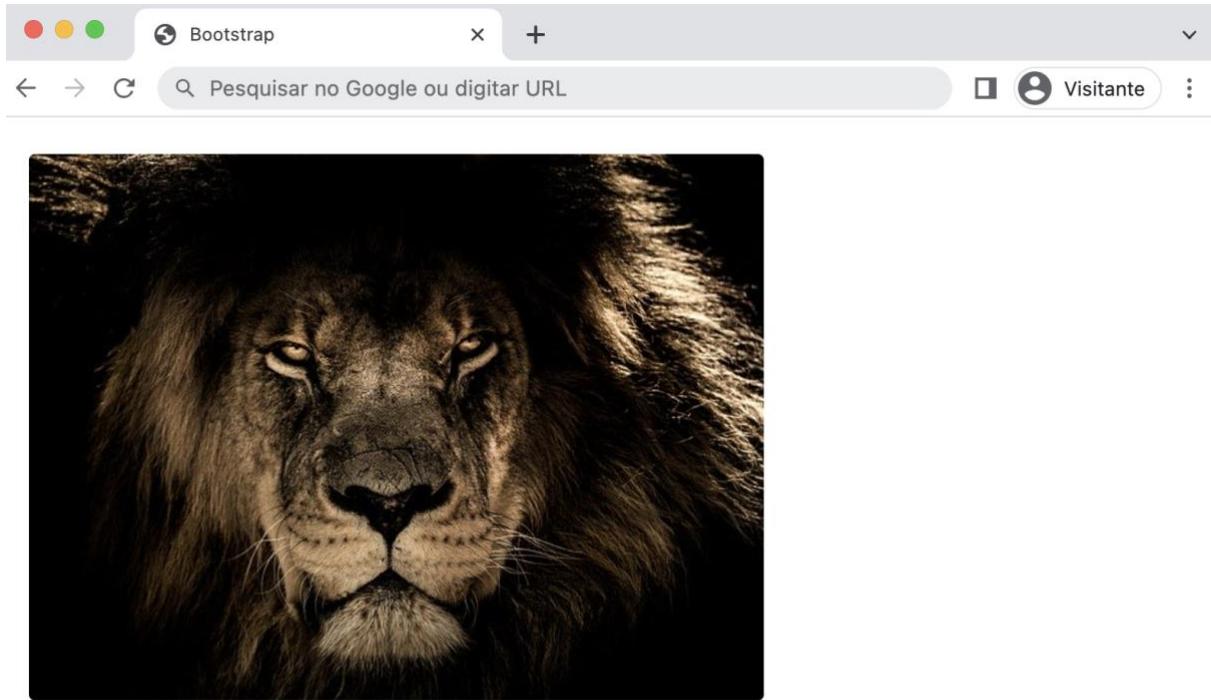
```

1 <figure class="figure p-4">
2   
3   <figcaption class="figure-caption">
4     Eu sou um leão.
5   </figcaption>
6 </figure>

```

Agora veja o resultado:

Figura 23. Figura com legenda. Perceba que a legenda é posicionada logo abaixo da figura.



Fonte: Imagem de Ian Lindsay por Pixabay (disponível em <https://pixabay.com/pt/photos/le%c3%a3o-africano-gato-selvagem-felino-2888519/>).

Tabelas

Bootstrap fornece uma série de classes para estilizar tabelas. Por padrão, toda a tabela deve receber a classe “table”. Essa classe pode ser complementada por outras classes como:

- **table-striped:** adiciona listas na tabela;
- **table-hover:** ao passar o mouse sobre uma linha, o navegador irá destacá-la;
- **table-sm:** tabela condensada;
- **table-bordered:** tabela com bordas;
- **table-borderless:** tabela sem bordas;

- **table-[*cor]**: colore a tabela (ver padrões de cores do Bootstrap, como *primary*, etc.);
- **table-responsive**: tabela responsiva.

Observe a tabela de exemplo a seguir:

```
1 <table class="">
2     <thead>
3         <tr>
4             <th>#</th>
5             <th>Número</th>
6         </tr>
7     </thead>
8     <tbody>
9         <tr>
10            <td>1</td>
11            <td>Um</td>
12        </tr>
13        <tr>
14            <td>2</td>
15            <td>Dois</td>
16        </tr>
17        <tr>
18            <td>3</td>
19            <td>Três</td>
20        </tr>
21    </tbody>
22 </table>
```

Observe como o navegador exibiria essa tabela por padrão:

Figura 24. Tabela sem qualquer formatação.

#	Número
1	Um
2	Dois
3	Três

Fonte: próprio autor.

Agora vamos alterar apenas a linha 1, testando as diferentes classes do Bootstrap. Vamos começar apenas inserindo a classe “table”:

```
1 <table class="table">
```

Observe como o estilo da tabela se altera por completo:

Figura 25. Classe table.

#	Número
1	Um
2	Dois
3	Três

Fonte: próprio autor.

Agora, vamos adicionar listras as linhas de nossa tabela. Podemos fazer isso com a classe “table-striped”:

```
1 <table class="table table-striped">
```

Figura 26. Classe table-striped (listras).

#	Número
1	Um
2	Dois
3	Três

Fonte: próprio autor.

Note que por padrão, Bootstrap não inclui bordas externas. Podemos adicioná-las usando a classe “table-bordered”:

```
1 <table class="table table-bordered">
```

Figura 27. Classe table-bordered.

#	Número
1	Um
2	Dois
3	Três

Fonte: próprio autor.

Ou podemos ainda remover todas as bordas usando “table-borderless”:

```
1 <table class="table table-borderless">
```

Figura 28. Classe table-borderless (tabela sem bordas).

#.	Número
1	Um
2	Dois
3	Três

Fonte: próprio autor.

Além disso, Bootstrap inclui um valor de preenchimento padrão, o que aumenta o tamanho final da tabela. Podemos criar tabelas mais condensadas usando “table-sm”:

```
1 <table class="table table-sm">
```

Para que você veja a diferença, vamos colocá-la lado a lado com o modelo padrão de tabela:

Figura 29. Tabela padrão vs. Tabela condensada (table-sm).

Tabela Padrão		Tabela Condensada (.table-sm)	
#.	Número	#.	Número
1	Um	1	Um
2	Dois	2	Dois
3	Três	3	Três

Fonte: próprio autor.

Tabelas coloridas

Tabelas também aceitam os atributos de cores. Para isso, basta chamar a classe “table-*”, onde * pode ser um dos tipos padrões de cores do Bootstrap (confira a imagem a seguir).

Figura 30. Colorindo tabelas.

Classe	Cabeçalho	Cabeçalho
Default	Célula	Célula
Primary	Célula	Célula
Secondary	Célula	Célula
Success	Célula	Célula
Danger	Célula	Célula
Warning	Célula	Célula
Info	Célula	Célula
Light	Célula	Célula
Dark	Célula	Célula

Fonte: adaptado de

<https://getbootstrap.com/docs/5.1/content/tables/>.

Esse mesmo padrão pode ser aplicado em toda a tabela, ou apenas em linhas ou células.

```

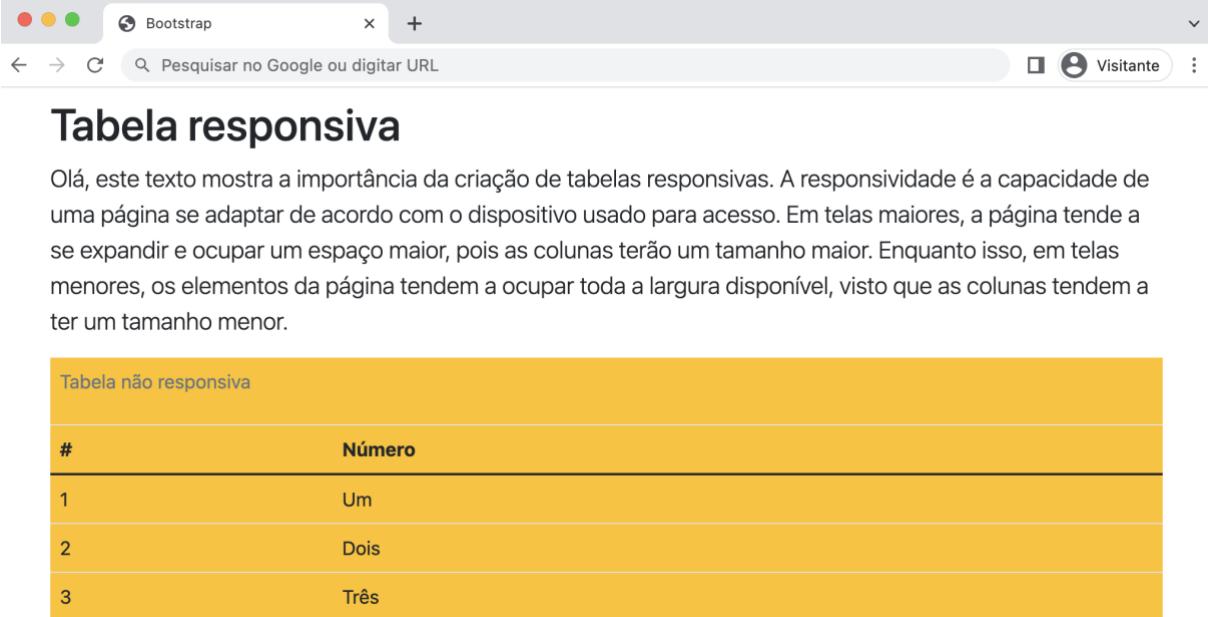
1 <!-- Tabelas <table> -->
2 <table class="table-primary">...</table>
3 <table class="table-secondary">...</table>
4 <table class="table-success">...</table>
5 <table class="table-danger">...</table>
6 <table class="table-warning">...</table>
7 <table class="table-info">...</table>
8 <table class="table-light">...</table>
```

```
9 <table class="table-dark">...</table>
10
11 <!-- Linhas <tr> -->
12 <tr class="table-primary">...</tr>
13 <tr class="table-secondary">...</tr>
14 <tr class="table-success">...</tr>
15 <tr class="table-danger">...</tr>
16 <tr class="table-warning">...</tr>
17 <tr class="table-info">...</tr>
18 <tr class="table-light">...</tr>
19 <tr class="table-dark">...</tr>
20
21 <!-- Células (<td> ou <th>) -->
22 <tr>
23   <td class="table-primary">...</td>
24   <td class="table-secondary">...</td>
25   <td class="table-success">...</td>
26   <td class="table-danger">...</td>
27   <td class="table-warning">...</td>
28   <td class="table-info">...</td>
29   <td class="table-light">...</td>
30   <td class="table-dark">...</td>
31 </tr>
```

Tabelas responsivas

Observe a tabela a seguir:

Figura 31. Exemplo de uma tabela (ainda) não-responsiva colorida de laranja.



A screenshot of a web browser window titled "Bootstrap". The address bar shows "Bootstrap" and a search bar with "Pesquisar no Google ou digitar URL". The main content area has a title "Tabela responsiva" and a paragraph about the importance of creating responsive tables. Below this is a table with a yellow background. The table has a caption "Tabela não responsiva" and two columns: "#". and "Número". It contains three rows with values 1, Um; 2, Dois; and 3, Três.

#	Número
1	Um
2	Dois
3	Três

Fonte: próprio autor.

Perceba que essa tabela possui uma cor de fundo dada por:

```
1 <table class="table bg-warning">
```

Note que há um texto acima, que se ajusta perfeitamente ao espaço da página. Agora veja o que acontece se o texto presente na tabela superar o espaço disponível:

Figura 32. Tabela não responsiva que ultrapassa a largura máxima.

The screenshot shows a web browser window titled "Bootstrap". The address bar says "Pesquisar no Google ou digitar URL". The page content includes a section titled "Tabela responsiva" with the following text:

Olá, este texto mostra a importância da criação de tabelas responsivas. A responsividade é a capacidade de uma página se adaptar de acordo com o dispositivo usado para acesso. Em telas maiores, a página tende a se expandir e ocupar um espaço maior, pois as colunas terão um tamanho maior. Enquanto isso, em telas menores, os elementos da página tendem a ocupar toda a largura disponível, visto que as colunas tendem a ter um tamanho menor.

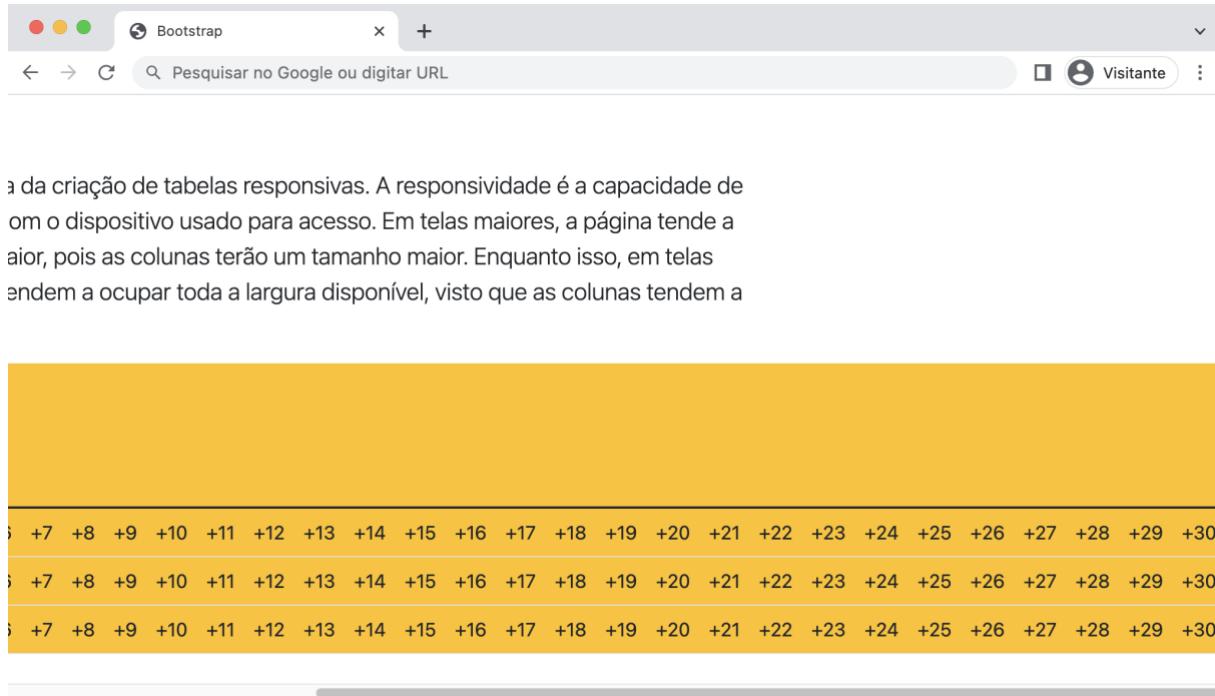
Below this text is a table with a yellow header row. The header row contains the text "Tabela não responsiva" and the column headers "# Número". The data rows are as follows:

		#	Número
1	Um	+1 +2 +3 +4 +5 +6 +7 +8 +9 +10 +11 +12 +13 +14 +15 +16 +17 +18 +19 +20 +21 +22	
2	Dois	+1 +2 +3 +4 +5 +6 +7 +8 +9 +10 +11 +12 +13 +14 +15 +16 +17 +18 +19 +20 +21 +22	
3	Três	+1 +2 +3 +4 +5 +6 +7 +8 +9 +10 +11 +12 +13 +14 +15 +16 +17 +18 +19 +20 +21 +22	

Fonte: próprio autor.

Uma barra horizontal de rolagem surgiu. Note que se arrastarmos a barra para o lado, nosso texto parcialmente desaparece:

Figura 33. Impacto de uma tabela não responsiva na leitura do conteúdo da página.



Fonte: próprio autor.

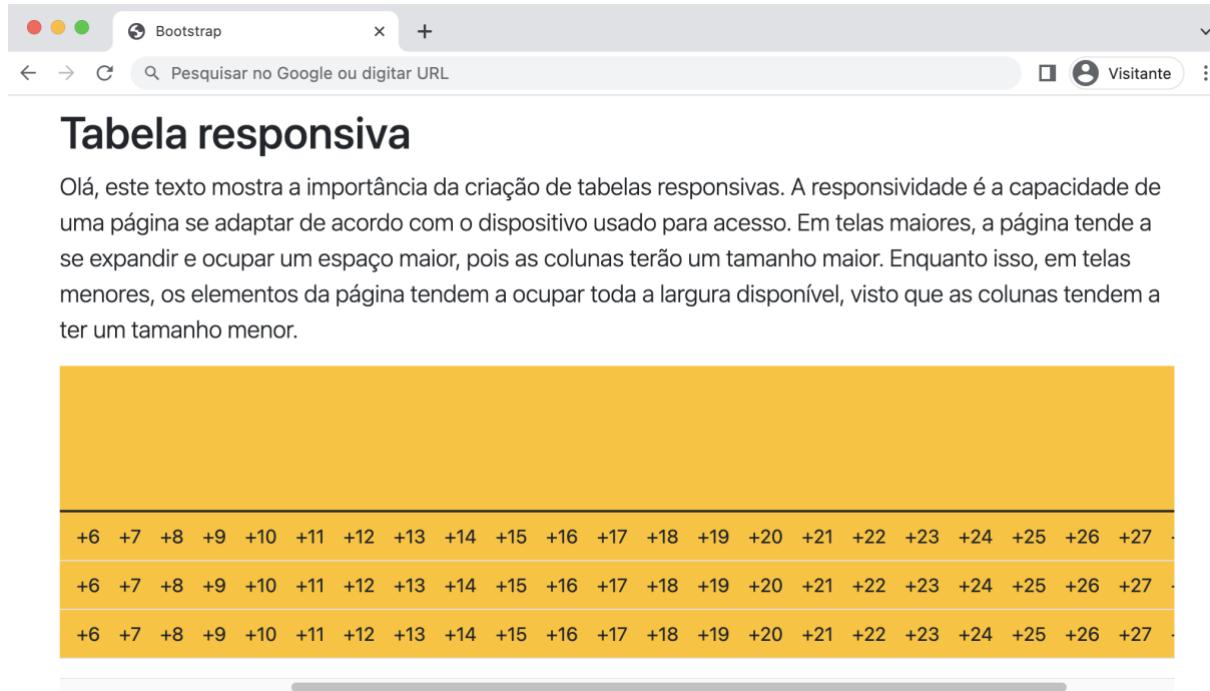
Em uma página responsiva, isso não é uma característica desejada.

Por isso, Bootstrap fornece a classe “table-responsive”. Note que a classe table-responsive não é aplicada diretamente à tabela, mas sim a uma div contêiner:

```
1 <div class="table-responsive">
2     <table class="table bg-warning">
3         [...]
4     </table>
5 </div>
```

Note que agora podemos navegar pelo conteúdo sem prejudicar a leitura dos outros elementos do site:

Figura 34. Tabela responsiva. Note que ao rolar pela barra lateral, os outros conteúdos não são impactados.



Fonte: próprio autor.

Formulários

Formulários são componentes HTML que permitem uma interação entre usuários de um site e o sistema. Entretanto, as configurações padrões de estilos de componentes de formulários não são muito atrativas. Por isso, Bootstrap fornece um conjunto de classes para formatação de campos de texto, caixas de seleção suspensos, caixa de marcação, dentre outros.

A principal classe utilizada é o “form-control”. Ela é aplicada a caixas de entrada de textos (`<input type="text">`) ou áreas de texto (`<textarea>`). Para botões do tipo checkbox ou radio, podemos usar a classe “form-input-check”. Já para caixas de seleção (`<select>`), podemos usar a classe “form-select”.

Observe alguns exemplos:

```
1 <form>
2         <input type="text" class="form-control"
placeholder="Digite aqui">
```

```
3  
4      <input type="checkbox" class="form-input-check">  
Checkbox  
5  
6      <input type="radio" name="" class="form-input-check">  
Radio  
7  
8      <select class="form-select">  
9          <option>Caixa de seleção</option>  
10     </select>  
11 </form>
```

Agora compare a diferença dos estilos padrão do navegador com o estilo aplicado pelo Bootstrap:

Figura 35. Formulários com (a) e sem (b) as classes do Bootstrap.

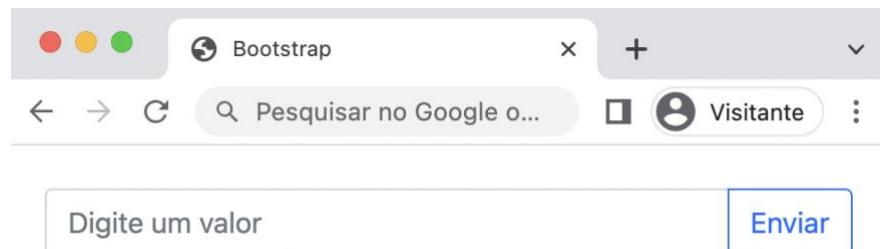


Fonte: próprio autor.

Grupo de inputs

Podemos agrupar botões usando a classe “input-group”. Observe:

Figura 36. Agrupando entradas.

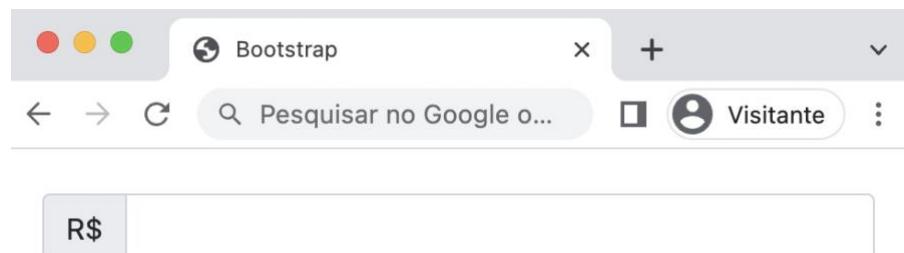


Fonte: próprio autor.

Podemos ainda agrupar campos de texto:

```
1 <!-- grupo de botões -->
2 <div class="input-group">
3
4     <!-- caixa cinza -->
5     <span class="input-group-text">R$</span>
6
7     <!-- entrada -->
8     <input type="text" class="form-control" aria-
label="Valor">
9 </div>
```

Figura 37. Texto agrupado com input.



Fonte: próprio autor.

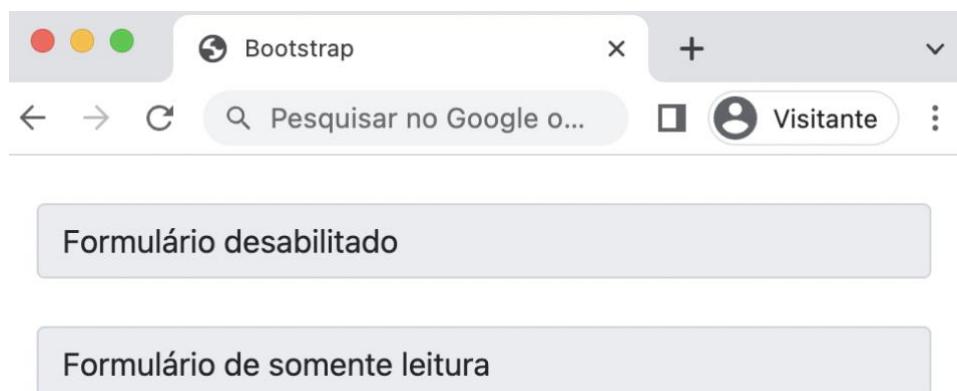
Desabilitando um campo (somente leitura)

Podemos desabilitar um campo de um formulário ou até mesmo configurá-lo como somente leitura usando os atributos *disabled* e *readonly*, respectivamente. Observe como podemos implementar isso:

```
1 <input type="text" class="form-control" value="Formulário
2     desabilitado" aria-label="Valor" disabled>
3
4 <input type="text" class="form-control" value="Formulário
5     somente leitura" aria-label="Valor" readonly>
```

O resultado da implementação desses atributos será bastante parecido:

Figura 38. Campos de formulários desabilitado e de somente leitura.



Fonte: próprio autor.

Rótulos flutuantes

Os rótulos flutuantes são uma nova característica do Bootstrap, que serve como um complemento ao atributo *placeholder*. Por padrão, *placeholder* permite exibir um texto em um campo *input*, que desaparece quando o usuário clica no campo. O atributo *placeholder* permite interfaces mais limpas, uma vez que aproveita o espaço dentro do campo para exibir informações relevantes (economizando espaços gastos com títulos e rótulos). Para visualizar o texto do *placeholder* novamente, o usuário deve clicar fora do campo de entrada, tendo em vista que nenhum valor deve ter sido digitado. Com os rótulos flutuantes isso não é necessário.

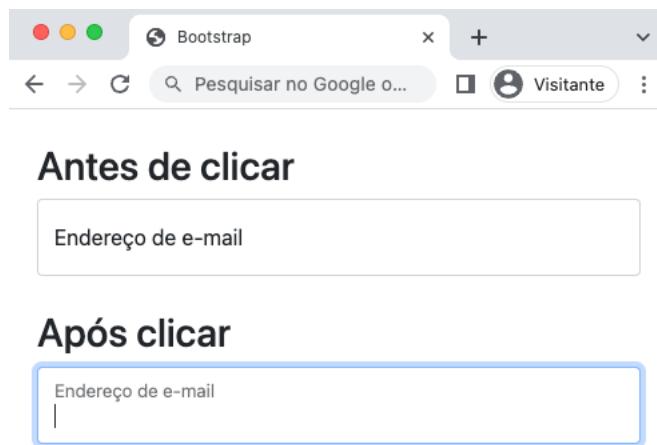
Rótulos flutuantes (*float labels*) permitem que um título para o campo seja exibido dentro da área do próprio campo quando o botão estiver ativo. Eles requerem que um campo <label> seja colocado junto com um campo <input>. Ambos devem ser agrupados por uma div contêiner que deve receber a classe “form-floating”. Observe o código:

```

1 <!-- div contêiner -->
2 <div class="form-floating">
3
4   <!-- campo de entrada -->
5   <input type="email" class="form-control"
6     id="rotulo-flutuante" placeholder="nome@email.com.br">
7
8   <!-- rótulo que será exibido acima do campo -->
9   <label for="rotulo-flutuante">Endereço de e-mail</label>
10
11 </div>
```

O resultado será:

Figura 39. Rótulos flutuantes.



Fonte: próprio autor.

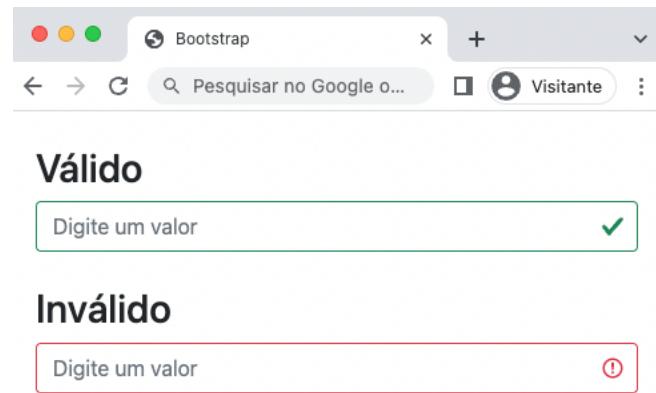
Validação de formulários

Podemos validar campos de um formulário com as classes “is-valid” e “is-invalid”. Veja:

```
1 <!-- Válido -->
2 <input type="text" class="form-control is-valid"
3      id="exemplo-1" placeholder="Digite um valor">
4
5 <!-- Inválido -->
6 <input type="text" class="form-control is-invalid"
7      id="exemplo-2" placeholder="Digite um valor">
```

Este código irá produzir a seguinte página:

Figura 40. Validação de formulários.



Fonte: próprio autor.



Bootstrap possui uma série de componentes que facilitam a construção de websites. Neste capítulo, serão apresentados os 10 componentes essenciais para construção de uma página completa.

Componentes

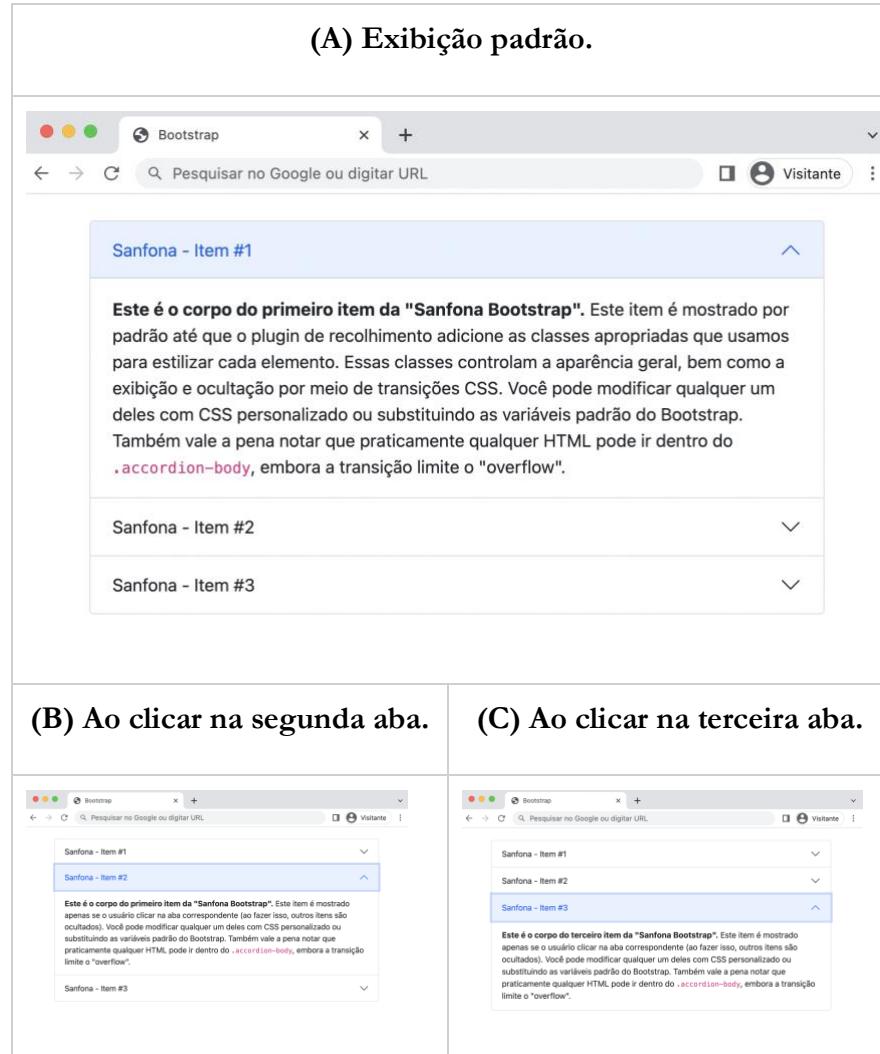
Em sua documentação, Bootstrap apresenta diversos componentes. De fato, alguns terão mais uso do que outros; por isso, neste capítulo serão elencados os 10 principais componentes:

1. Caixas expansivas (sanfona e classe de colapso)
2. Caixa de alertas (*alert*) e caixa de notificações (*toast*)
3. Emblemas (*badge*)
4. Botões, grupo de botões e *dropdown*
5. *Cards*
6. Barras de progresso
7. Carrossel de imagens
8. Modal – caixas de diálogos
9. *Popovers* e *tooltips*
10. Menus: barras de navegação e abas

1. Caixas expansivas (sanfona e classe de colapso)

Em muitos casos, desenvolvedores de interface precisam exibir informações sob demanda, isto é, a informação deve estar disponível apenas se o usuário clicar em determinado ponto ou interagir com determinado elemento. Em Bootstrap, isso é possível por meio de diversos componentes, como as “Sanfonas Bootstrap”. Observe a figura a seguir para analisar como esse componente funciona.

Figura 41. Sanfona Bootstrap.



Fonte: próprio autor.

O código a seguir, apresenta detalhadamente como implementar o componente sanfona:

```

1 <!-- Sanfona -->
2 <div class="accordion" id="acordeao-exemplo">
3
4   <!-- ITEM 1: aba do sanfona + conteúdo -->
5   <div class="accordion-item">
6
7     <!-- aba - botão de carregamento -->
8     <h2 class="accordion-header" id="titulo-1">
9       <button

```

```
10      class="accordion-button"
11      data-bs-toggle="collapse"
12      data-bs-target="#item-1"
13      aria-expanded="true"
14      aria-controls="item-1"
15      >
16      Sanfona - Item #1
17      </button>
18      </h2>
19
20      <!-- conteúdo -->
21      <div
22          id="item-1"
23          class="accordion-collapse collapse show"
24          aria-labelledby="titulo-1"
25          data-bs-parent="#acordeao-exemplo"
26      >
27      <div class="accordion-body">
28
29          <strong>Este é o corpo do segundo item da "Sanfona Bootstrap".</strong> Este item é mostrado por padrão até que o plugin de recolhimento adicione as classes apropriadas que usamos para estilizar cada elemento. Essas classes controlam a aparência geral, bem como a exibição e ocultação por meio de transições CSS. Você pode modificar qualquer um deles com CSS personalizado ou substituindo as variáveis padrão do Bootstrap. Também vale a pena notar que praticamente qualquer HTML pode ir dentro do <code>.accordion-body</code>, embora a transição limite o "overflow".
30
31      </div>
32      </div>
33      </div>
34
35      <!-- ITEM 2: aba do sanfona + conteúdo -->
36      <div class="accordion-item">
37
38          <!-- aba - botão de carregamento -->
39          <h2 class="accordion-header" id="titulo-2">
40              <button
41                  class="accordion-button"
42                  data-bs-toggle="collapse"
43                  data-bs-target="#item-2"
```

```
44      aria-expanded="true"
45      aria-controls="item-2"
46    >
47      Sanfona - Item #2
48    </button>
49  </h2>
50
51  <!-- conteúdo -->
52  <div
53    id="item-2"
54    class="accordion-collapse collapse"
55    aria-labelledby="titulo-2"
56    data-bs-parent="#acordeao-exemplo"
57  >
58    <div class="accordion-body">
59
60      <strong>Este é o corpo do primeiro item da "Sanfona Bootstrap".</strong> Este item é mostrado apenas se o usuário clicar na aba correspondente (ao fazer isso, outros itens são ocultados). Você pode modificar qualquer um deles com CSS personalizado ou substituindo as variáveis padrão do Bootstrap. Também vale a pena notar que praticamente qualquer HTML pode ir dentro do <code>.accordion-body</code>, embora a transição limite o "overflow".
61
62    </div>
63  </div>
64 </div>
65
66 <!-- ITEM 1: aba do sanfona + conteúdo -->
67 <div class="accordion-item">
68
69  <!-- aba - botão de carregamento -->
70  <h2 class="accordion-header" id="titulo-3">
71    <button
72      class="accordion-button"
73      data-bs-toggle="collapse"
74      data-bs-target="#item-3"
75      aria-expanded="true"
76      aria-controls="item-3"
77    >
78      Sanfona - Item #3
```

```

79      </button>
80  </h2>
81
82  <!-- conteúdo -->
83  <div
84      id="item-3"
85      class="accordion-collapse collapse"
86      aria-labelledby="titulo-3"
87      data-bs-parent="#acordeao-exemplo"
88  >
89      <div class="accordion-body">
90          <strong>Este é o corpo do terceiro item da "Sanfona Bootstrap".</strong> Este item é mostrado apenas se o usuário clicar na aba correspondente (ao fazer isso, outros itens são ocultados). Você pode modificar qualquer um deles com CSS personalizado ou substituindo as variáveis padrão do Bootstrap. Também vale a pena notar que praticamente qualquer HTML pode ir dentro do <code>.accordion-body</code>, embora a transição limite o "overflow".
91      </div>
92  </div>
93 </div>
94
95 </div>

```

Classe “.collapse”

O componente sanfona é baseado na classe “.collapse”. Essa classe é responsável por exibir ou esconder elementos aplicando um efeito de transição durante o processo. De modo generalizado podemos visualizar este componente da seguinte forma:

```

1 <!-- botão de ativação -->
2 <button
3     class="btn btn-primary"
4     data-bs-toggle="collapse"
5     data-bs-target="#item_secreto"
6 >
7     Exibir caixa oculta

```

```
8 </button>
9
10 <!-- Texto escondido -->
11 <div
12     class="collapse collapse-horizontal my-2"
13     id="item_secreto"
14 >
15     <span class="card p-4">Texto oculto</span>
16 </div>
```

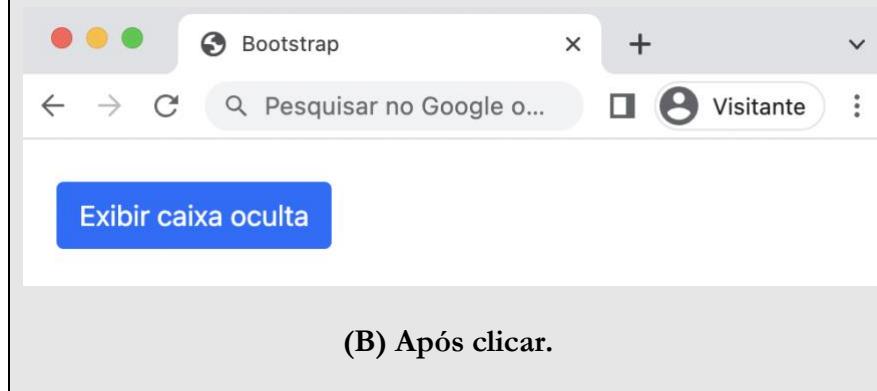
Note que o componente é composto por dois itens:

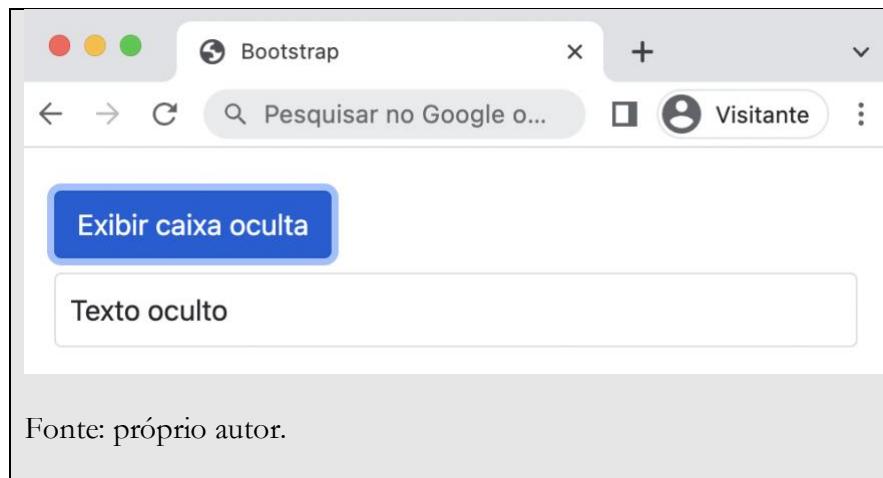
- (1) um botão de ativação/desativação;
- (2) um elemento HTML que fica escondido e só aparece após o clique no botão de ativação.

Agora observe a interface produzida:

Figura 42. Classe collapse.

(A) Antes de clicar.





2. Caixa de alertas (*alert*) e caixa de notificações (*toast*)

Caixas de alertas e de notificações são componentes que têm objetivos parecidos: mostrar mensagens para o usuário, mas por meio de estilos distintos.

As caixas de alertas (*alert*) são mais simples e podem ser carregados com a classe “alert”. Note que um elemento desse tipo também deve receber uma segunda classe indicando a cor “alert-*”, onde * pode ser primary, secondary, success, warning, danger, info, dark ou light. Links devem receber a classe “alert-link”.

Observe o exemplo a seguir:

```

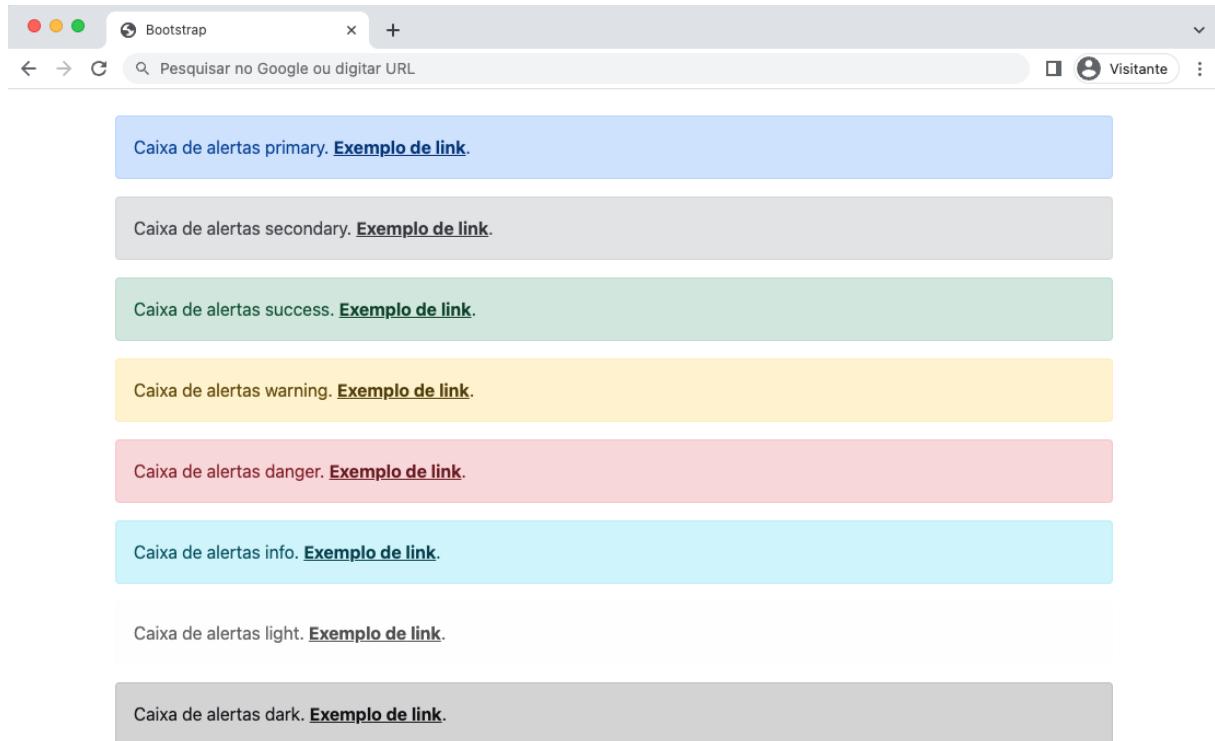
1 <div class="alert alert-primary" role="alert">
2   Caixa de alertas primary.
3   <a href="#" class="alert-link">Exemplo de link</a>.
4 </div>
5 <div class="alert alert-secondary" role="alert">
6   Caixa de alertas secondary.
7   <a href="#" class="alert-link">Exemplo de link</a>.
8 </div>
9 <div class="alert alert-success" role="alert">
10  Caixa de alertas success.
11  <a href="#" class="alert-link">Exemplo de link</a>.

```

```
12 </div>
13 <div class="alert alert-warning" role="alert">
14   Caixa de alertas warning.
15   <a href="#" class="alert-link">Exemplo de link</a>.
16 </div>
17 <div class="alert alert-danger" role="alert">
18   Caixa de alertas danger.
19   <a href="#" class="alert-link">Exemplo de link</a>.
20 </div>
21 <div class="alert alert-info" role="alert">
22   Caixa de alertas info.
23   <a href="#" class="alert-link">Exemplo de link</a>.
24 </div>
25 <div class="alert alert-light" role="alert">
26   Caixa de alertas light.
27   <a href="#" class="alert-link">Exemplo de link</a>.
28 </div>
29 <div class="alert alert-dark" role="alert">
30   Caixa de alertas dark.
31   <a href="#" class="alert-link">Exemplo de link</a>.
32 </div>
```

Este código produzirá a seguinte página:

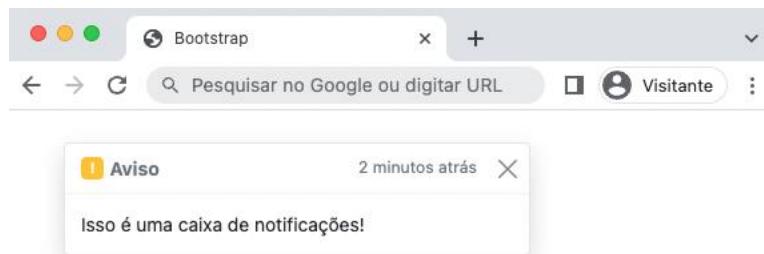
Figura 43. Caixas de alertas.



Fonte: próprio autor.

Por outro lado, as caixas de notificação têm um design mais sofisticado e podem ser usadas para envio de mensagens rápidas para o usuário. Caixas de notificação são também conhecidas como *toasts* ou “torradas” na tradução (acredito que tenha alguma relação com a forma como são utilizadas: elas surgem na tela de forma parecida com um pão saindo de uma torradeira). Observe o exemplo a seguir:

Figura 44. Exemplo de um componente *toast*. Pode-se fechar o componente clicando no X.



Fonte: próprio autor.

Observe o código usado para criá-lo:

```

1 <!-- Caixa de notificações -->
2 <div class="toast fade show" role="alert"
3           aria-live="assertive" aria-atomic="true">
4
5     <!-- Cabeçalho -->
6     <div class="toast-header">
7
8         <label class="badge bg-warning me-1">!</label>
9         <strong class="me-auto">Aviso</strong>
10        <small class="text-muted">2 minutos atrás</small>
11
12        <!-- Botão fechar -->
13        <button type="button" class="btn-close"
14          data-bs-dismiss="toast" aria-
15          label="Close"></button>
16    </div>
17
18    <!-- Corpo da caixa de notificações -->
19    <div class="toast-body">
20        Isso é uma caixa de notificações!
21    </div>
22 </div>

```

3. Emblemas (*badge*)

Badges (ou na tradução “emblemas”) são componentes bastante simples, mas que podem ser de grande valia a sua interface. Em geral, o componente *badge* é usado em elementos *inline* (como as tags ``, `` ou `<label>`) para aplicação de destaque e pode ser combinado com as classes de cores de plano de fundo (`bg-*`). Pode ainda ser usado em conjunto com a classe `rounded-pill` para inserção de bordas arredondadas.

O código a seguir ilustra como podemos implementá-lo:

```

1 <p>O componente <label class="badge
2   bg-primary">badge</label> destaca um trecho do texto.</p>
3

```

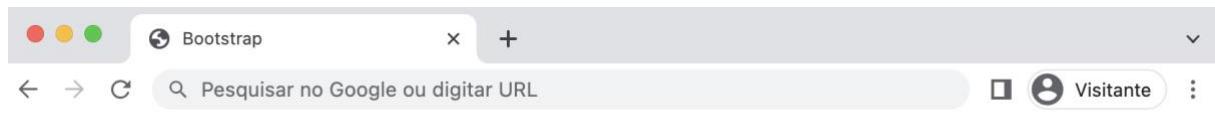
```

4 <p>O componente <label class="badge bg-primary
5     rounded-pill">badge</label> pode ser usado em conjunto
6     com a classe rounded-pill.</p>

```

O código produzirá a seguinte interface:

Figura 45. Componente badge. Pode-se alterar a cor de fundo trocando a classe bg-primary por outra classe bg-*.



O componente **badge** destaca um trecho do texto.

O componente **badge** pode ser usado em conjunto com a classe rounded-pill.

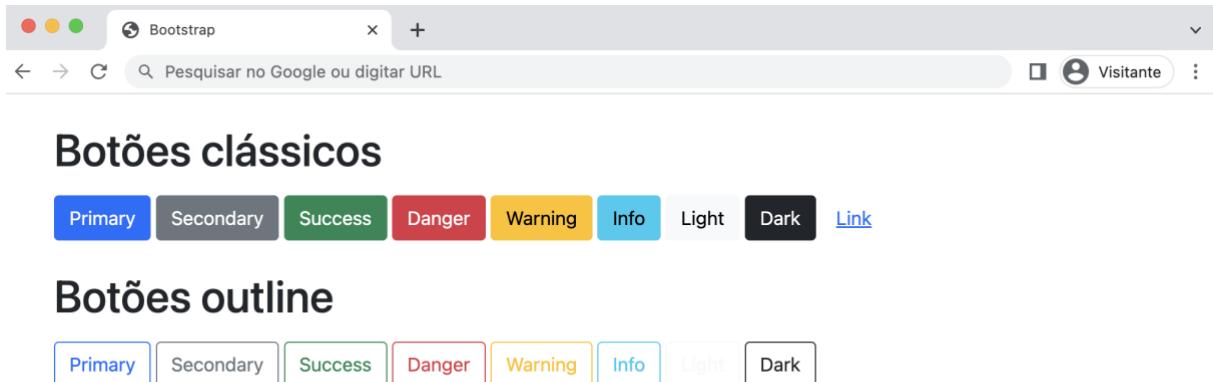
Fonte: próprio autor.

4. Botões, grupo de botões e dropdown

Botões são um dos componentes mais utilizados em Bootstrap. Eles podem ser carregados por meio da classe “.btn”, seguindo da classe “.btn-*”, onde * representa uma cor, sendo:

- **primary**: azul
- **secondary**: cinza
- **success**: verde
- **danger**: vermelho
- **warning**: laranja
- **info**: azul ciano
- **light**: branco gelo
- **dark**: preto acinzentado
- **link**: formata um botão como se fosse um link

Figura 46. Botões.



Fonte: próprio autor.

Podemos ter botões clássicos (com preenchimento com a cor correspondente) ou botões de *outline* (onde apenas a borda corresponde à cor informada). Para criar botões clássicos utilize o seguinte código:

```
1 <!-- Botões clássicos -->
2
3 <button class="btn btn-primary">Primary</button>
4 <button class="btn btn-secondary">Secondary</button>
5 <button class="btn btn-success">Success</button>
6 <button class="btn btn-danger">Danger</button>
7 <button class="btn btn-warning">Warning</button>
8 <button class="btn btn-info">Info</button>
9 <button class="btn btn-light">Light</button>
10 <button class="btn btn-dark">Dark</button>
11 <button class="btn btn-link">Link</button>
```

Para criar botões somente com borda (sem preenchimento), utilize as classes ".btn-outline-[cor]", como o seguinte exemplo:

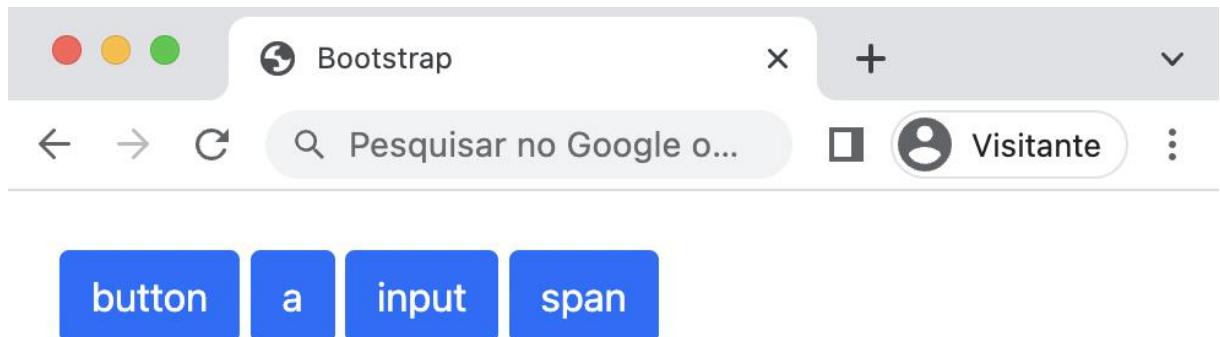
```
1 <!-- Botões outline -->
2
```

```
3 <button class="btn btn-outline-primary">Primary</button>
4 <button class="btn btn-outline-secondary"> Secondary </button>
5 <button class="btn btn-outline-success">Success</button>
6 <button class="btn btn-outline-danger">Danger</button>
7 <button class="btn btn-outline-warning">Warning</button>
8 <button class="btn btn-outline-info">Info</button>
9 <button class="btn btn-outline-light">Light</button>
10 <button class="btn btn-outline-dark">Dark</button>
```

Note que podemos aplicar a classe “.btn” a qualquer tipo de elemento, desde botões (<button>), links (<a>), inputs de submissão (<input>) ou até mesmo tags do tipo .

```
1 <button class="btn btn-primary">button</button>
2 <a class="btn btn-primary">a</a>
3 <input type="submit" class="btn btn-primary" value="input">
4 <span class="btn btn-primary">span</span>
```

Figura 47. Classe “.btn” aplicada a <button>, <a>, <input> ou .



Fonte: próprio autor.

Alterando o tamanho de um botão

Podemos ainda alterar o tamanho de um botão com as classes btn-lg e btn-sm. Veja:

```

1 <a class="btn btn-primary btn-sm">Botão pequeno</a>
2 <a class="btn btn-primary">Botão padrão</a>
3 <a class="btn btn-primary btn-lg">Botão grande</a>

```

Figura 48. Alterando tamanhos de botões.



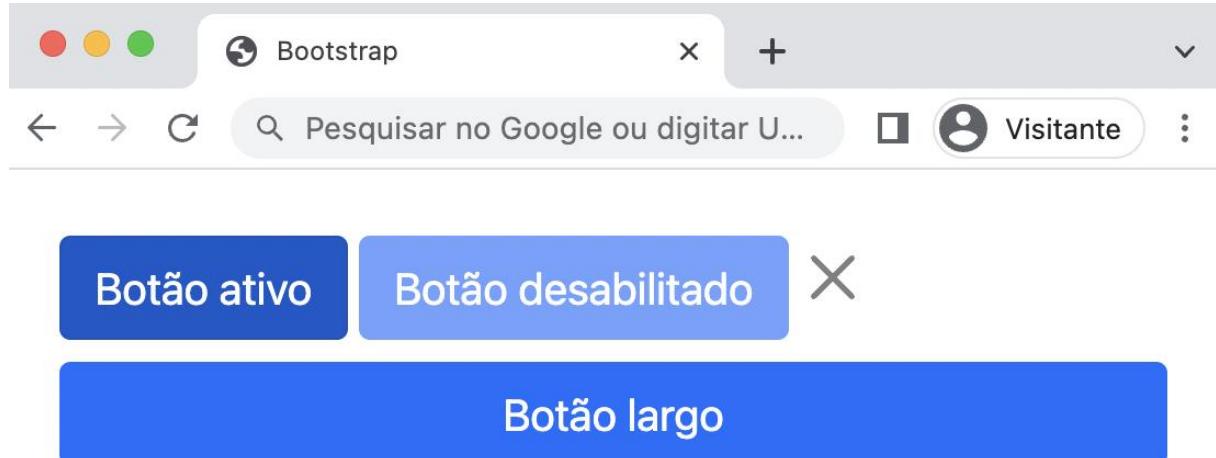
Fonte: próprio autor.

Estilo de botões: ativo, desabilitado, ocupando toda a linha e botão fechar

Agora vamos aplicar quatro outras classes para configurar o estilo de um botão:

- **active**: deixa o botão em um estado ativo (ou seja, ele fica com uma cor mais escura);
- **disabled**: desabilita o botão, ou seja, deixa o botão com uma cor desbotada (além disso, ao passar o mouse sobre ele, perceberá que as características de link do botão foram removidas);
- **btn-close**: cria um botão fechar (com um “x”). O botão fechar é utilizado em muitos outros componentes, por isso vale a pena citá-lo aqui;
- **d-grid**: faz com que o botão passe a ocupar todo o espaço de uma linha.

Figura 49. Botões ativo (classe “.active”), desabilitado (classe “.disabled”), fechar (classe “.btn-close”) e ocupando 100% da largura de uma linha (classe “.d-grid”).



Fonte: próprio autor.

Código que renderiza esses botões:

```

1 <!-- Botão ativo -->
2 <a class="btn btn-primary active">Botão ativo</a>
3
4 <!-- Botão desabilitado -->
5 <a class="btn btn-primary disabled">Botão desabilitado</a>
6
7 <!-- Botão fechar -->
8 <button type="button" class="btn-close"
9           aria-label="Close"></button>
10 <!-- Botão largo -->
11 <a class="btn btn-primary d-grid mt-2">Botão largo</a>
```

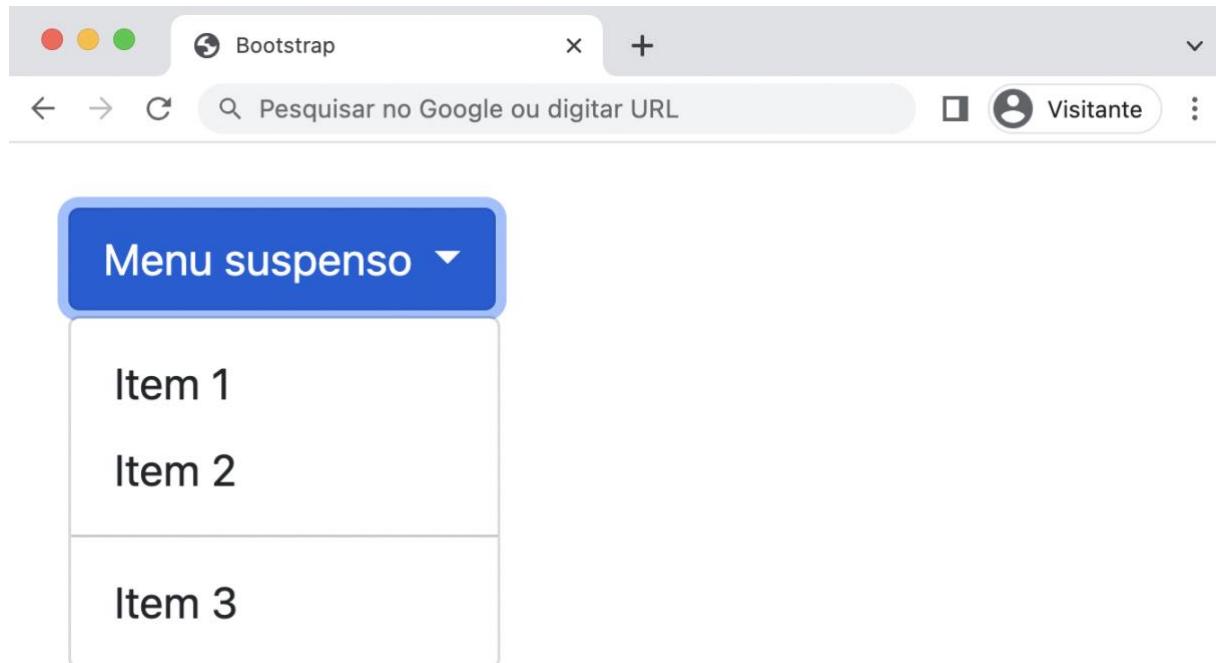
Menus suspensos (*dropdown*)

Menus suspensos podem ser criados pela classe “.dropdown”. Eles devem possuir um botão de ativação, que deve receber a classe “.dropdown-toggle” (responsável por inserir uma seta no fim do botão). A seguir, ainda dentro do contêiner “.dropdown”, um elemento do tipo deve receber a classe “.dropdown-menu” (esse elemento estará inicialmente oculto). Os elementos desta lista devem receber a classe “.dropdown-item”. Esse menu só estará disponível se o usuário clicar no botão de ativação. Observe o código:

```
1 <!-- Menu suspenso -->
2 <div class="dropdown">
3
4     <!-- Botão de ativação -->
5     <button
6         class="btn btn-primary dropdown-toggle"
7         id="menu-suspenso-oculto"
8         data-bs-toggle="dropdown">
9         Menu suspenso
10    </button>
11
12    <!-- itens do menu -->
13    <ul class="dropdown-menu">
14        <li><a class="dropdown-item" href="#">Item 1</a></li>
15        <li><a class="dropdown-item" href="#">Item 2</a></li>
16
17        <!-- Linha separadora -->
18        <li><hr class="dropdown-divider"></li>
19
20        <li><a class="dropdown-item" href="#">Item 3</a></li>
21    </ul>
22 </div>
```

Veja como fica nosso menu suspenso ao ser clicado:

Figura 50. Menu suspenso (*dropdown*) quando clicado.



Fonte: próprio autor.

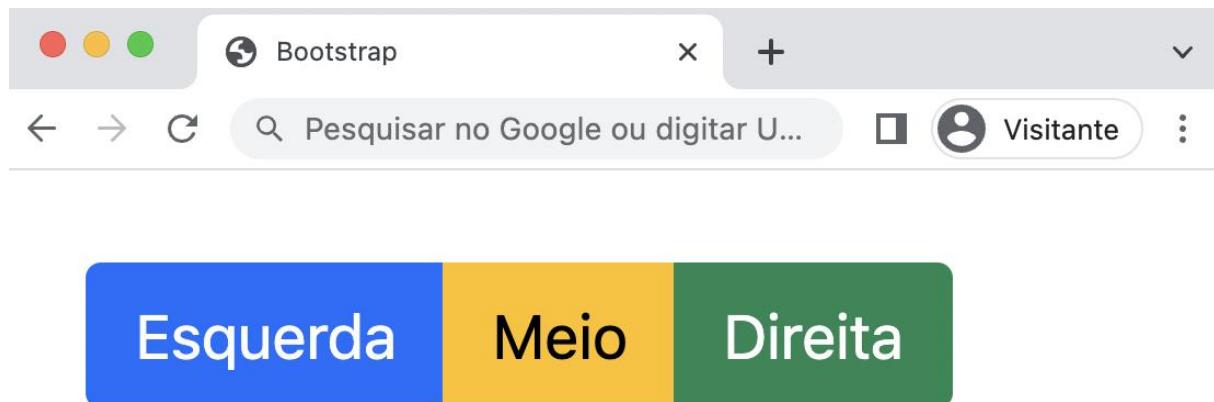
Grupo de botões

Pode-se agrupar botões usando uma div contêiner que recebe a classe “.btn-group”. Observe o código:

```
1 <!-- grupo de botões -->
2 <div class="btn-group" role="group">
3   <button type="button" class="btn btn-primary">Esquerda
</button>
4   <button type="button" class="btn btn-warning"> Meio
</button>
5   <button type="button" class="btn btn-success"> Direita
</button>
6 </div>
```

Ao abrir este código no navegador, você verá a seguinte interface:

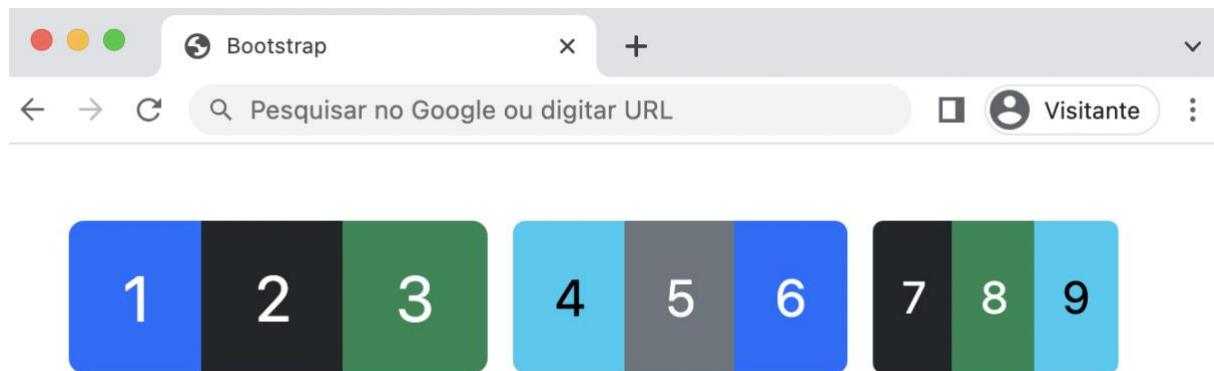
Figura 51. Agrupando botões (classe “.btn-group”). As cores dos elementos foram alteradas apenas ilustrativamente.



Fonte: próprio autor.

Note que podemos agrupar grupos de botões usando a classe “.btn-toolbar”, ou ainda, alterar a largura de botões usando as classes “.btn-group-lg” (grupo de botões mais largos) ou “.btn-group-sm” (grupo de botões pequenos).

Figura 52. Agrupando grupos de botões de tamanhos diferentes.



Fonte: próprio autor.

Observe o código usado para gerar esta figura:

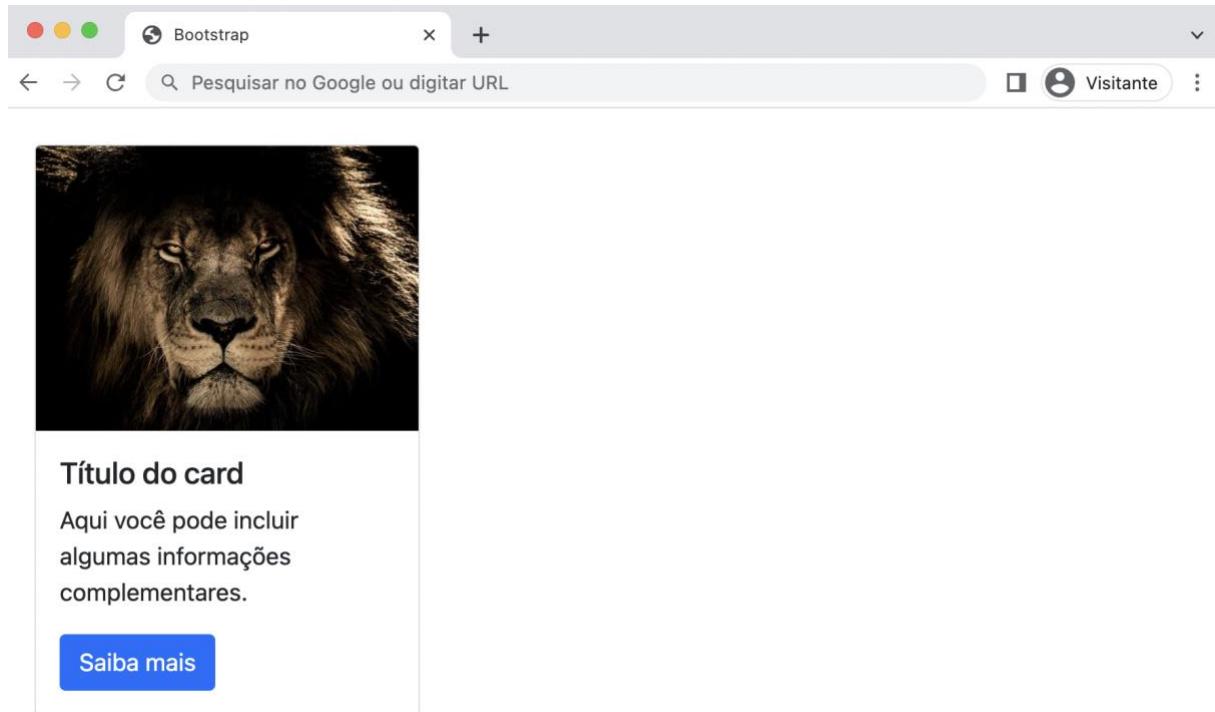
```
1 <!-- barra de ferramentas -->
```

```
2 <div class="btn-toolbar">
3
4     <!-- grupo de botões mais largos -->
5     <div class="btn-group btn-group-lg me-2" role="group">
6         <button type="button" class="btn btn-primary">1</button>
7         <button type="button" class="btn btn-dark">2</button>
8         <button type="button" class="btn btn-success">3</button>
9     </div>
10
11    <!-- grupo de botões padrão -->
12    <div class="btn-group me-2" role="group">
13        <button type="button" class="btn btn-info">4</button>
14        <button type="button" class="btn btn-secondary">5</button>
15        <button type="button" class="btn btn-primary">6</button>
16    </div>
17
18    <!-- grupo de botões menores -->
19    <div class="btn-group btn-group-sm" role="group">
20        <button type="button" class="btn btn-dark">7</button>
21        <button type="button" class="btn btn-success">8</button>
22        <button type="button" class="btn btn-info">9</button>
23    </div>
24 </div>
```

5. Cards

Cards são componentes que exibem uma formatação amigável para elementos listados que possuem uma imagem, título, texto complementar e um link de acesso, por exemplo. Observe um exemplo a seguir:

Figura 53. Componente de card tradicional.



Fonte: Pixabay (imagem do leão). Print de tela gerado pelo próprio autor.

Para que este código funcione, é necessário ter o arquivo “lion.jpg” (teste usando uma imagem qualquer). A imagem de exemplo foi obtida gratuitamente no Pixabay (<https://pixabay.com/>).

```
1 <!-- Cards Bootstrap -->
2 <div class="card col-4"> <!-- tamanho: col-4 -->
3
4   <!-- Imagem no topo -->
5   
6
7   <!-- Corpo do card -->
8   <div class="card-body">
9
10    <!-- Título -->
```

```

11     <h5 class="card-title">Título do card</h5>
12
13     <!-- Texto complementar -->
14     <p class="card-text">
15         Aqui você pode incluir algumas informações
16         complementares.
17     </p>
18     <!-- Link de acesso -->
19     <a href="#" class="btn btn-primary">Saiba mais</a>
20
21     </div> <!-- /Fim do Corpo do card -->
22
23 </div><!-- /Fim do card -->
```

Podemos ainda criar cards horizontais, como o seguinte:

Figura 54. Card horizontal.



Fonte: Pixabay (imagem do leão). Print de tela gerado pelo próprio autor.

Para isso, podemos utilizar colunas. Observe o código a seguir:

```
1 <!-- Card horizontal -->
```

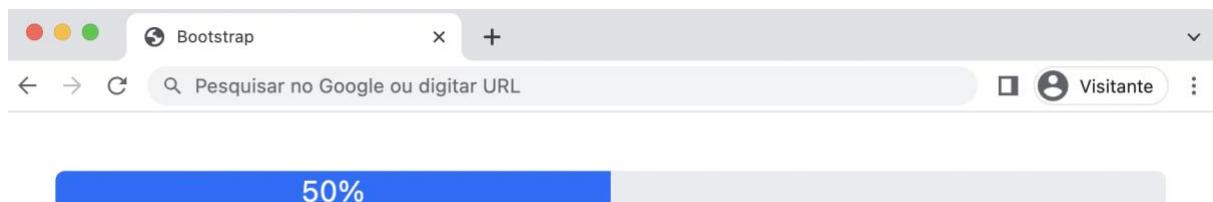
```
2 <div class="card mb-3">
3   <div class="row g-0">
4
5     <!-- coluna da imagem -->
6     <div class="col-4">
7       
8     </div>
9
10    <!-- coluna do texto -->
11    <div class="col-8">
12      <div class="card-body">
13        <h5 class="card-title">Titulo do card</h5>
14        <p class="card-text">
15          Um card horizontal tem uma maior largura.
16          Você pode usar este componente para descrever
17          melhor um determinado elemento de sua página.
18        </p>
19        <p class="card-text">
20          <small class="text-muted">
21            Atualizado há 3 minutos atrás
22          </small>
23        </p>
24      </div>
25    </div>
26  </div>
27 </div>
```

Você pode ainda alterar a cor dos cards usando as classes “bg-[cor]” descritas anteriormente.

6. Barras de progresso

Barras de progresso permitem acompanhar o carregamento de determinado item. Observe um exemplo de uma barra de progresso 50% preenchida:

Figura 55. Barra de progresso.



Fonte: próprio autor.

Podemos criá-la declarando uma div contêiner que recebe a classe “.progress” com uma div interna recebendo a barra de progresso (classe “.progress-bar”). Observe como ficaria o código:

```

1 <!-- barra de progresso -->
2 <div class="progress">
3
4   <div
5     class="progress-bar bg-primary"
6     role="progressbar"
7     style="width: 50%; /* largura da barra */"
8     aria-valuenow="50"
9     aria-valuemin="0"
10    aria-valuemax="100"
11  >
12    50% <!-- texto exibido dentro da barra -->
13  </div>
14
15 </div>
```

Altere a classe `bg-*` na linha 5 para alterar a cor desejada da barra. Você pode ainda adicionar listras a sua barra de progresso usando a classe `.progress-bar-striped` ou adicionar animações usando a classe `.progress-bar-animated` (na linha 5).

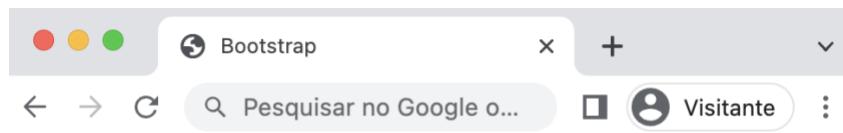
Note que Bootstrap é responsável apenas pela exibição e estilo dos componentes. Para fazer com que a animação esteja funcional, é necessário integrar sua interface ao *backend* da aplicação, por exemplo, por meio de códigos Ajax.

Além das barras de progresso, podemos adicionar ícones de carregamento, como os *spinners* (círculos giratórios). Veja um exemplo:

```
1 <div class="spinner-border text-primary" role="status">
2   <span class="visually-hidden">Carregando...</span>
3 </div>
```

Note que alteramos a cor de um *spinner* com a classe “text-primary”.

Figura 56. Ícone de carregamento (*spinner*).



Fonte: próprio autor.

7. Carrossel de imagens

O carrossel de imagens do Bootstrap combina JavaScript com uma interface amigável para exibir um *slider* de imagens. Para este exemplo, vamos coletar três imagens aleatórias da plataforma Pixabay (<https://pixabay.com/pt/>).

Um carrossel de imagens poderá ser composto por:

- (1) Um conjunto de imagens que se alterna na tela de tempos em tempos (no exemplo a seguir, vamos usar três);
- (2) Um par de botões de navegação que permitem trocar a imagem atual na tela (anterior e posterior);
- (3) Legenda: cada imagem pode ter um título e uma pequena descrição exibida;
- (4) Marcadores da posição da imagem: mostra qual a imagem atual, permitindo que o usuário acesse outras imagens clicando diretamente sobre este marcador.

Figura 57. Carrossel de imagens.



Fonte: próprio autor.

Agora, observe o código fonte e veja como podemos implementar um carrossel de imagens com Bootstrap (detalhes foram incluídos nos trechos comentados). É necessário que o script do Bootstrap esteja carregado no rodapé da página.

```
1 <!-- Carrossel de imagens - um slider -->
```

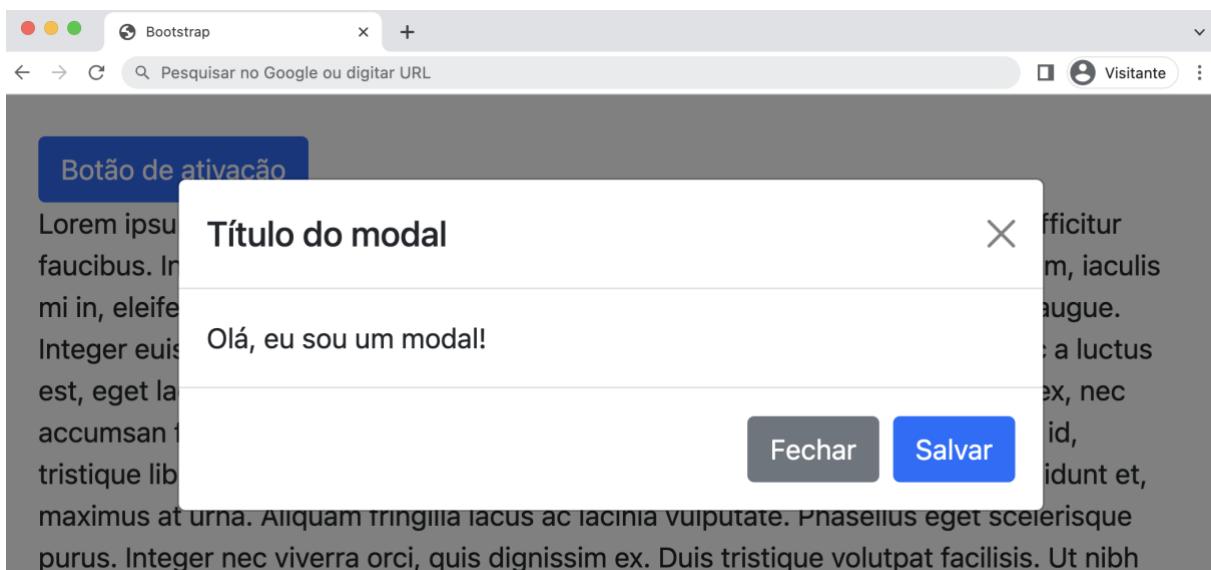
```
2 <div id="carrossel-de-imagens" class="carousel slide"
3     data-bs-ride="carousel">
4
5     <!-- traços marcadores -->
6     <div class="carousel-indicators">
7
8         <!-- primeiro marcador (ativo por padrão) -->
9         <button type="button" data-bs-target="#carrossel-
10             de-imagens" data-bs-slide-to="0" class="active" aria-
11                 current="true" aria-label="Slide 1"></button>
12
13         <!-- segundo marcador: link imagem 2 -->
14         <button type="button" data-bs-target="#carrossel-de-imagens"
15             data-bs-slide-to="1" aria-label="Slide 2"></button>
16
17         <!-- terceiro marcador: link imagem 3 -->
18         <button type="button" data-bs-target="#carrossel-de-imagens"
19             data-bs-slide-to="2" aria-label="Slide 3"></button>
20
21     </div>
22
23     <!-- Bloco de imagens -->
24     <div class="carousel-inner">
25
26         <!-- imagem 1 + legenda -->
27         <div class="carousel-item active">
28
29             <!-- imagem 1 -->
30             
31
32             <!-- legenda -->
33             <div class="carousel-caption d-none d-md-block">
34                 <h5>Leão</h5>
35                 <p>Exemplo de imagem.</p>
36             </div>
37         </div>
38
39         <!-- imagem 2 + legenda -->
40         <div class="carousel-item">
41
42             <!-- imagem 2 -->
43             
44
```

```
45      <!-- legenda -->
46      <div class="carousel-caption d-none d-md-block">
47          <h5>Cidade</h5>
48          <p>Exemplo de imagem.</p>
49      </div>
50  </div>
51
52  <!-- imagem 3 + legenda -->
53  <div class="carousel-item">
54
55      <!-- imagem 3 -->
56      
58
59      <!-- legenda -->
60      <div class="carousel-caption d-none d-md-block">
61          <h5>Montanha</h5>
62          <p>Exemplo de imagem.</p>
63      </div>
64  </div><!-- /fim bloco de imagens -->
65
66
67  <!-- Botões de navegação -->
68  <button class="carousel-control-prev" type="button"
69      data-bs-target="#carrossel-de-imagens" data-bs-slide="prev">
70      <span class="carousel-control-prev-icon" aria-
71      hidden="true"></span>
72      <span class="visually-hidden">Anterior</span>
73  </button>
74
75  <button class="carousel-control-next" type="button"
76      data-bs-target="#carrossel-de-imagens" data-bs-slide="next">
77      <span class="carousel-control-next-icon" aria-
78      hidden="true"></span>
79      <span class="visually-hidden">Próximo</span>
80  </button>
81
82 </div>
```

8. Modal – Caixas de diálogos

Modals (caixas de diálogos) são um dos componentes mais elegantes do Bootstrap. *Modals* usam o conceito de *lightboxes*, isto é, enquanto uma caixa de mensagens aparece no centro da tela, uma tela preta semitransparente cobre o fundo, dando destaque para a mensagem principal. Observe um exemplo:

Figura 58. Exemplo de *modal*. Observe que a caixa de diálogo do *modal* flutua sobre a página principal. Um plano de fundo preto semitransparente cobre toda a página, aumentando o destaque da mensagem exibida.



Fonte: próprio autor.

Um modal é composto por duas partes principais:

(1) Botão de ativação:

```

1 <!-- Botão de ativação -->
2 <button
3   class="btn btn-primary"
4   data-bs-toggle="modal"
5   data-bs-target="#meu-primeiro-modal"
6 >
7   Botão de ativação
8 </button>
```

(2) Div oculta do modal:

```
1 <!-- Modal - inclui fundo semitransparente -->
2 <div class="modal fade" id="meu-primeiro-modal"
tabindex="-1">
3
4     <!-- caixa de diálogo -->
5     <div class="modal-dialog">
6
7         <!-- conteúdo do modal -->
8         <div class="modal-content">
9
10            <!-- cabeçalho do modal -->
11            <div class="modal-header">
12                <h5 class="modal-title" id="titulo-modal">Título
do modal</h5>
13                <button type="button" class="btn-close" data-bs-
dismiss="modal"
14                    aria-label="Close"></button> <!-- botão fechar: x
-->
15            </div>
16
17            <!-- corpo do modal -->
18            <div class="modal-body">
19                Olá, eu sou um modal!
20            </div>
21
22            <!-- rodapé do modal -->
23            <div class="modal-footer">
24                <button type="button" class="btn btn-secondary"
25                    data-bs-dismiss="modal">Fechar</button><!-- botão
fechar -->
26                <button type="button" class="btn btn-
primary">Salvar</button>
27            </div>
28
29            </div> <!-- /modal-content -->
30        </div> <!-- /modal-dialog -->
31    </div> <!-- /modal -->
```

Você pode ainda alterar a largura de um modal usando as classes descritas na tabela a seguir junto com a div da caixa de diálogo e com a classe “.modal-dialog” (linha 5 do código apresentado anteriormente).

Tabela 7. Tamanhos de modais. Classes devem ser aplicadas junto com a classe “.modal-dialog”.

Tamanho	Classe	Largura máxima modal
Pequeno	.modal-sm	300px
Predefinição	Nenhum	500px
Grande	.modal-lg	800px
Extra grande	.modal-xl	1140px

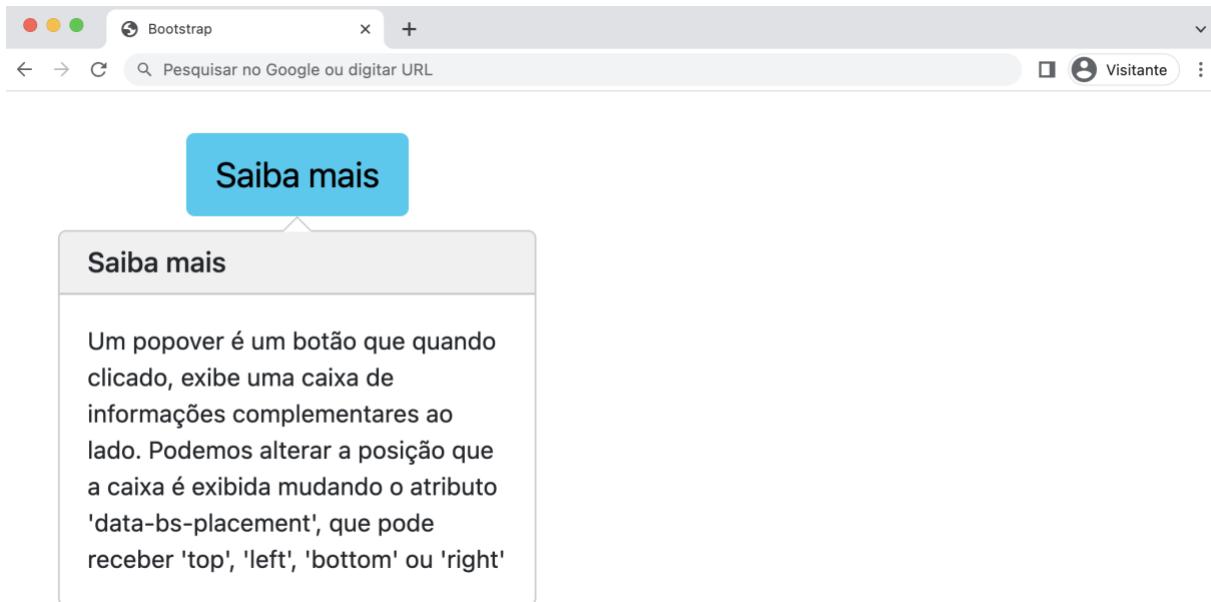
Fonte: Traduzido de
<https://getbootstrap.com/docs/5.1/components/modal/>.

9. Popovers e Tooltips

Popovers e *tooltips* são pequenos balões que aparecem ao lado de um elemento quando clicado (*popover*) ou quando sobreposto pelo *mouse* (*tooltip*).

Um *popover* é um componente que é ativado pelo clique em um determinado elemento, como um botão. Observe um exemplo de *popover*:

Figura 59. *Popovers* só são ativos quando clicados.



Fonte: próprio autor.

Neste caso, declaramos apenas um elemento HTML: um botão de ativação. Os dados exibidos no *popover* devem estar nos atributos “title” e “data-bs-content”. No atributo “data-bs-placement” podemos definir onde queremos que a caixa do *popover* seja posicionada: à esquerda (*left*), abaixo (*bottom*), à direita (*right*) ou acima (*top*). É claro que se o navegador definir que não há espaço disponível na posição informada, ele automaticamente posicionará a caixa onde for possível. Observe o código:

```

1 <button
2     class="btn btn-lg btn-info"
3     data-bs-toggle="popover"
4     title="Saiba mais"
5     data-bs-placement="bottom"
6     data-bs-content="Um popover é um botão que quando
7         clicado, exibe uma caixa de informações complementares ao
    lado. Podemos alterar a posição que a caixa é exibida mudando
    o atributo 'data-bs-placement', que pode receber 'top',
    'left', 'bottom' ou 'right'"
8 >Saiba mais</button>

```

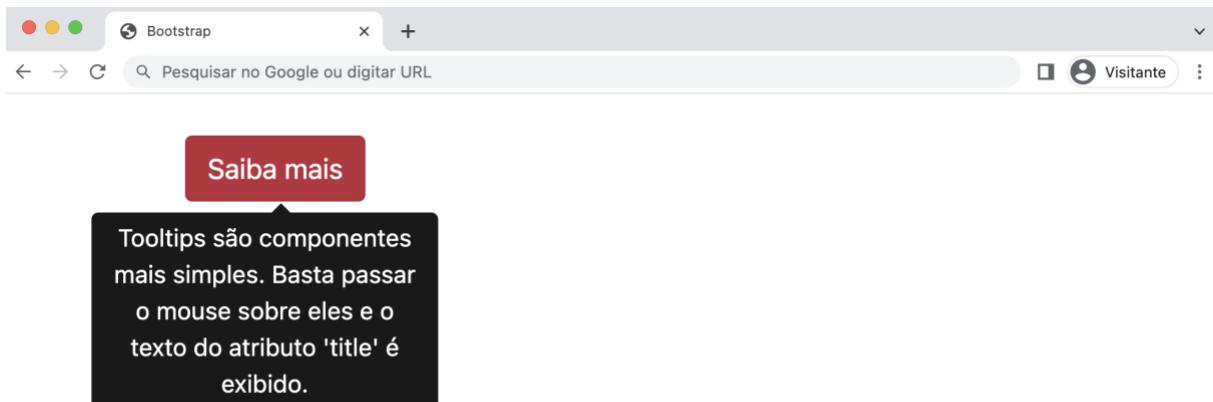
Note que ao incluir o elemento HTML do *popover*, o botão ainda não estará funcional. Isso se deve ao fato de *popovers* não serem ativados automaticamente uma vez que podem consumir uma grande quantidade

de recursos do navegador. Precisamos ativá-los usando JavaScript. O código abaixo, ativa todos os *popovers* em uma página (note que ele deve ser declarado após a declaração do Bootstrap JavaScript, em geral, antes do fim da página).

```
1 <!-- Bootstrap JavaScript -->
2 <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/boo-
tstrap.bundle.min.js" integrity="sha384-
ka7Sk0Gln4gmtz2MlQnikT1wXgYsOg+OMhuP+IlRH9sENBO0LRn5q+8nbTov4
+1p" crossorigin="anonymous"></script>
3
4
5 <!-- ativando os popovers -->
6 <script>
7     /* seleciona todos os popovers */
8     let popovers = document.querySelectorAll('[data-bs-
toggle="popover"]')
9
10    /* para cada popover */
11    for(let i of popovers){
12        new bootstrap.Popover(i); // ative-o
13    }
14 </script>
```

Os *tooltips* funcionam de maneira parecida. A diferença é que são ativos quando o mouse se posiciona sobre eles. Além disso, eles exibem apenas o valor presente no atributo “*title*” do botão de ativação. Observe:

Figura 60. Exemplo de *tooltip*. Quando o mouse passa por cima do botão, uma caixa de informações complementares é exibida logo abaixo.



Fonte: próprio autor.

Agora veja o código do botão de ativação:

```

1 <button
2     class="btn btn-danger"
3     data-bs-toggle="tooltip"
4     data-bs-placement="bottom"
5     title="Tooltips são componentes mais simples.
6             Basta passar o mouse sobre eles e o texto do
7             atributo 'title' é exibido."
8 >
9     Saiba mais
10 </button>
```

Note que os *tooltips* também precisam ser ativados via JavaScript. Então, no fim da página inclua o código:

```

1 <!-- ativando os tooltips -->
2 <script>
3     /* seleciona todos os tooltips */
4     let tooltips = document.querySelectorAll(
5         '[data-bs-toggle="tooltip"]'
6     )
7
8     /* para cada tooltip */
9     for(let i of tooltips){
10         new bootstrap.Tooltip(i); // ative-o
```

```

11      }
12 </script>

```

10. Menus: barras de navegação e abas

Os componentes de menus são essenciais para construção de páginas HTML completas com Bootstrap. De fato, esta seção engloba os vários tipos de menus, como barras de navegação (navbar), barras laterais em listas, menus laterais ocultos (offcanvas), além de identificadores de posição durante a rolagem (*scrollspy*), identificadores de caminhos (*breadcrumbs*) e barras de paginação.

Começaremos com um dos mais importantes: barras de navegação (navbar).

Barras de navegação

Barras de navegação (navbar), também conhecida como menu principal, é a barra superior onde geralmente inserimos o logo, principais links de acesso e barra de busca. Esse menu deve ser responsivo, isto é, ao ser aberto em um dispositivo com tela pequena, ele deve se ajustar automaticamente ao conteúdo (em geral, exibindo um menu de sanduíche).

O pseudocódigo a seguir ilustra como podemos organizar as tags em um menu:

```

1 <!-- PSEUDOCÓDIGO MENU -->
2 <NAVBAR classe="navbar navbar-[cor] bg-[cor] navbar-expand-[tamanho]">
3   <CONTAINER>
4     <LOGO classe="navbar-brand">
5     <MENU-SANDUÍCHE>
6       <DIV-COLAPSÁVEL classe="collapse navbar-collapse">
7         <LISTA-ESQUERDA classe="navbar-nav me-auto">
8           <ITEM-LISTA classe="nav-item">
9             <LINK classe="nav-link">
10            </ITEM-LISTA>
11          </LISTA-ESQUERDA>
12          <LISTA-DIREITA classe="d-flex">
13            <ITEM>

```

```

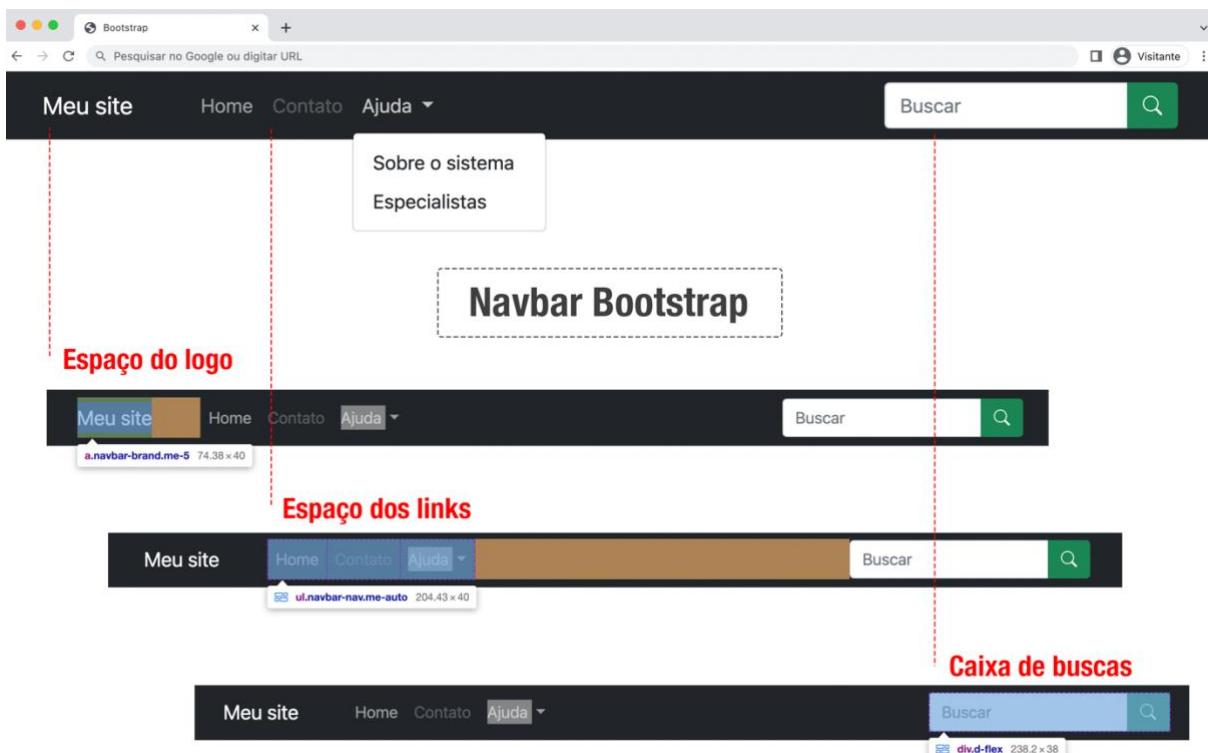
14          </LINKS-DIREITA>
15      </DIV-COLAPSÁVEL>
16  </CONTAINER>
17 </NAVBAR>

```

Note que podemos alterar a cor do menu mudando as classes expressas na linha 2. Note ainda que a classe navbar-[cor] prepara apenas as cores de links e outros elementos. Para alterar a cor do fundo, deve-se alterar a classe bg-[cor]. A classe navbar-expand-[tamanho] permite definir os níveis de responsividade (por exemplo, define em qual tamanho de tela será carregado o menu-sanduíche, assim, ocultando parte do menu).

Veja a seguir um exemplo de menu de navegação:

Figura 61. Exemplo de menu. Abaixo são demonstradas algumas partes do código.



Fonte: próprio autor.

Agora, observe o código completo que gera esse menu:

```

1 <!-- Barra de navegação -->
2 <nav class="navbar navbar-dark bg-dark navbar-expand-lg">
3
4    <!-- container -->

```

```

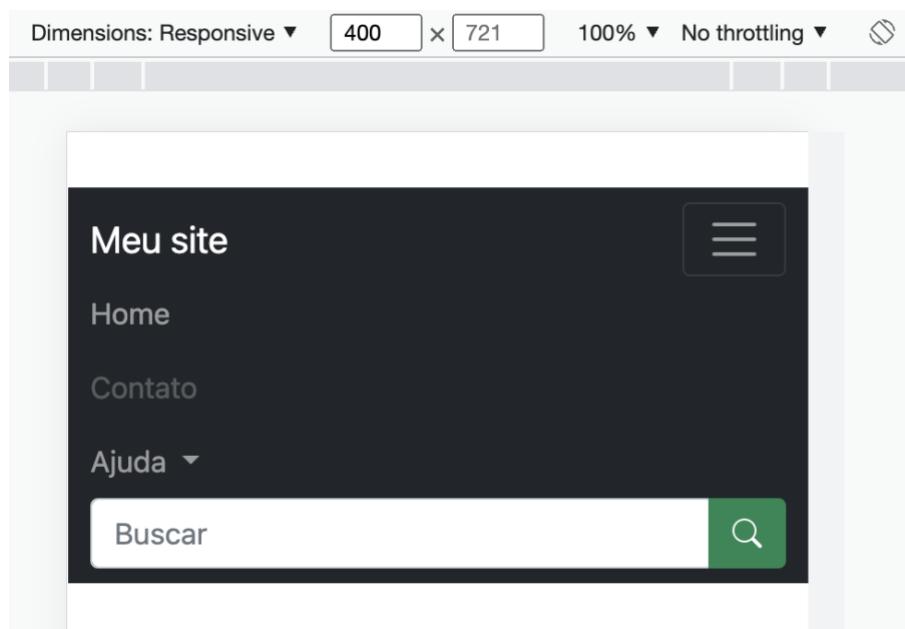
5      <div class="container">
6
7          <!-- logo -->
8          <a class="navbar-brand me-5" href="#">Meu site</a>
9
10         <!-- menu sanduíche (para smartphone) -->
11         <button class="navbar-toggler" data-bs-toggle="collapse"
12             data-bs-target="#menu-links">
13             <span class="navbar-toggler-icon"></span>
14         </button>
15
16         <!-- menu de links -->
17         <div class="collapse navbar-collapse" id="menu-links">
18
19             <!-- logo e link alinhados à esquerda -->
20             <ul class="navbar-nav me-auto">
21
22                 <!-- Link home -->
23                 <li class="nav-item">
24                     <a href="#" class="nav-link">Home</a>
25                 </li>
26
27                 <!-- Link desabilitado -->
28                 <li class="nav-item">
29                     <a href="#" class="nav-link disabled">Contato</a>
30                 </li>
31
32                 <!-- Menu suspenso -->
33                 <li class="nav-item dropdown">
34
35                     <!-- botão do menu suspenso -->
36                     <a class="nav-link dropdown-toggle" href="#"
37                         role="button" data-bs-toggle="dropdown">
38                         Ajuda
39                     </a>
40
41                     <!-- links ocultos -->
42                     <ul class="dropdown-menu">
43                         <li>
44                             <a class="dropdown-item" href="#">
45                             Sobre o sistema</a>
46                         </li>
47                         <li>
48                             <a class="dropdown-item" href="#">
49                             Especialistas</a>
50                         </li>
51                     </ul>
52
53                 </li> <!-- /nav-item dropdown -->
54
55             </ul> <!-- /navbar-nav -->
56
57             <!-- alinhado à direita -->
58             <div class="d-flex">
59                 <form class="input-group">
60                     <input type="text" class="form-control"
61                         placeholder="Buscar">
62                     <button class="btn btn-success">
```

```

63             <i class="bi bi-search"></i>
64         </button>
65     </form>
66   </div>
67 </div>
68
69 </div>
70 </nav>
```

Observe a seguir, como o mesmo código carregaria uma diferente interface em um smartphone:

Figura 62. Site carregado em um smartphone (dispositivo com tela pequena). Note que o botão do menu sanduíche (lado direito) foi carregado. Neste caso, o menu só será exibido se o usuário clicar nesse botão.



Fonte: próprio autor.

List group

Os grupos de lista (*list group*) podem ser utilizados para criar menus laterais. Observe o código a seguir:

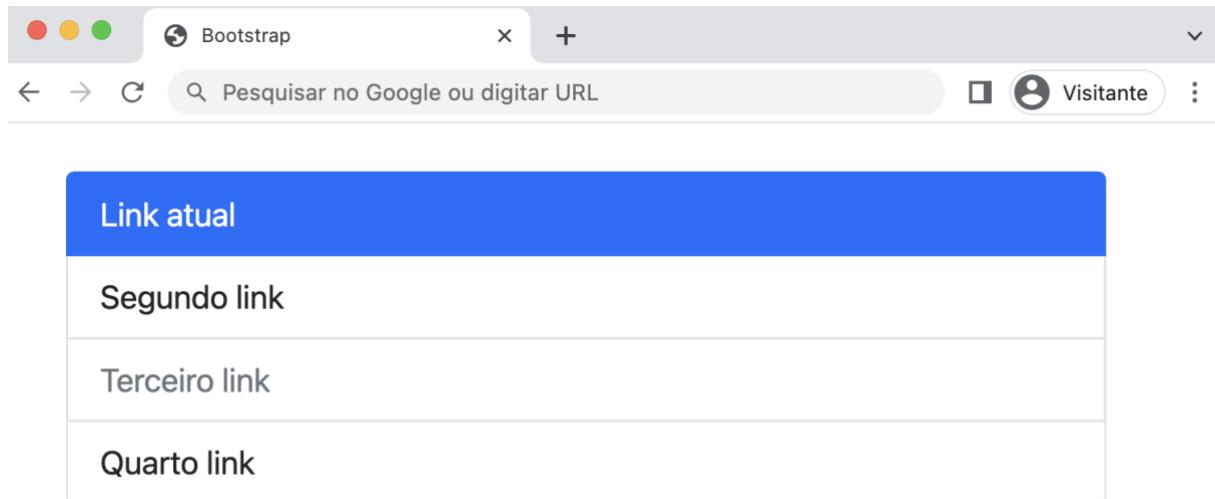
```

1 <!-- menu de grupo de listas -->
2 <div class="list-group">
3
4   <!-- Link ativo -->
5   <a href="#" class="list-group-item list-group-item-action active">
6     Link atual
```

```
7  </a>
8
9  <!-- Link -->
10 <a href="#" class="list-group-item list-group-item-action">
11   Segundo link
12 </a>
13
14 <!-- Link desabilitado -->
15 <a class="list-group-item list-group-item-action disabled">
16   Terceiro link
17 </a>
18
19 <!-- Link -->
20 <a href="#" class="list-group-item list-group-item-action">
21   Quarto link
22 </a>
23
24 </div>
```

Esse código produz a seguinte página:

Figura 63. Menu de grupo de listas.

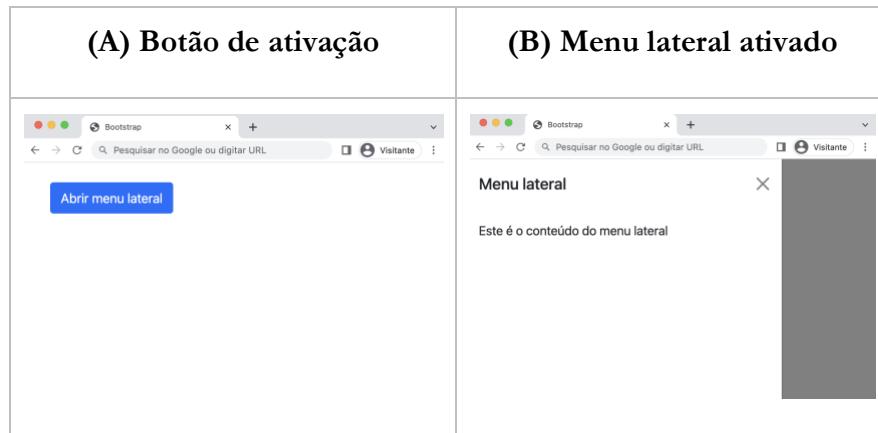


Fonte: próprio autor.

Offcanvas: menu lateral

Menus laterais (*offcanvas*) são menus que aparecem no canto esquerdo ao serem ativados, em geral, por meio do clique em um botão de ativação.

Figura 64. Menu lateral (*offcanvas*).



Fonte: próprio autor.

Podemos implementá-lo da seguinte forma:

```

1 <!-- botão de ativação -->
2 <a class="btn btn-primary" data-bs-toggle="offcanvas"
3   href="#menu-lateral">
4   Abrir menu lateral
5 </a>
6
7
8 <!-- menu lateral -->
9 <div class="offcanvas offcanvas-start" tabindex="-1" id="menu-
lateral">
10
11  <!-- cabeçalho -->
12  <div class="offcanvas-header">
13    <h5 class="offcanvas-title" id="offcanvasLabel">Menu
lateral</h5>
14    <button type="button" class="btn-close text-reset"
15      data-bs-dismiss="offcanvas" aria-label="Close"></button>
16  </div>
17
18  <!-- corpo -->
19  <div class="offcanvas-body">
20    Este é o conteúdo do menu lateral

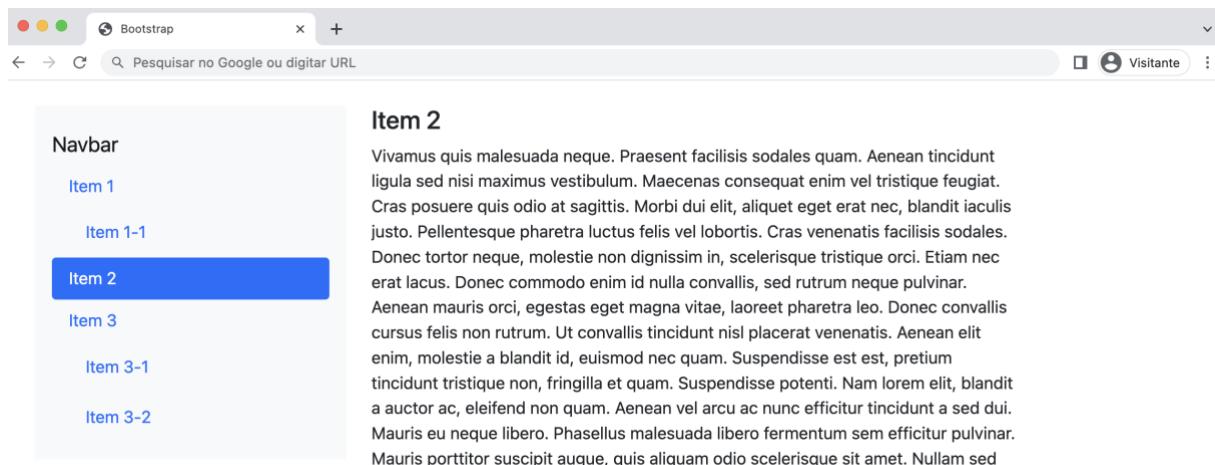
```

```
21  </div>
22
23 </div>
```

Menu de rolagem (*scrollspy*)

Scrollspy é um tipo de menu cujos itens são automaticamente marcados à medida que o usuário realiza a rolagem por certos títulos da página. Por exemplo:

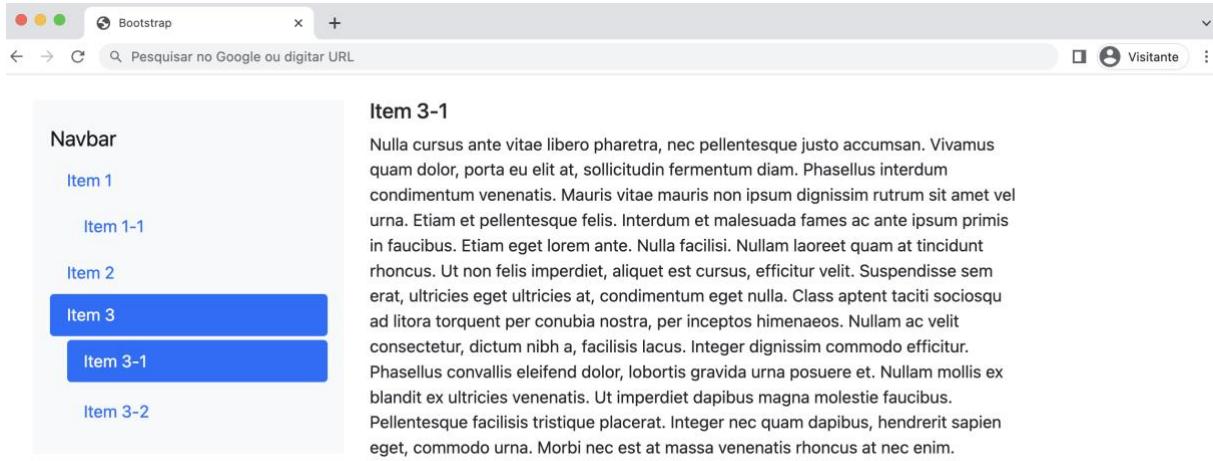
Figura 65. *Scrollspy*.



Fonte: próprio autor.

Aqui, o item 2 foi selecionado quando o usuário fez a rolagem da página. Ao continuar a descer, e ir até o item 3.1, por exemplo, a seguinte seleção é feita:

Figura 66. *Scrollpy* – rolagem por itens dentro da mesma página.



Fonte: próprio autor.

Para implementar isso, observe o código a seguir:

```

1 <!-- campo de rolagem -->
2 <div class="row">
3
4     <!-- coluna à esquerda -->
5     <div class="col-4">
6
7         <nav id="menu-rolagem" class="navbar navbar-light
8             bg-light flex-column align-items-stretch p-3">
9
10        <!-- Título -->
11        <a class="navbar-brand" href="#">Menu de rolagem</a>
12
13        <!-- Itens do menu -->
14        <nav class="nav nav-pills flex-column">
15            <a class="nav-link active" href="#item-1">Item 1</a>
16            <nav class="nav nav-pills flex-column">
17                <a class="nav-link ms-3 my-1" href="#item-1-1">Item 1-
18                1</a>
19            </nav>
20            <a class="nav-link" href="#item-2">Item 2</a>
21            <a class="nav-link" href="#item-3">Item 3</a>
22            <nav class="nav nav-pills flex-column">
23                <a class="nav-link ms-3 my-1" href="#item-3-1">Item 3-
24                1</a>

```

```

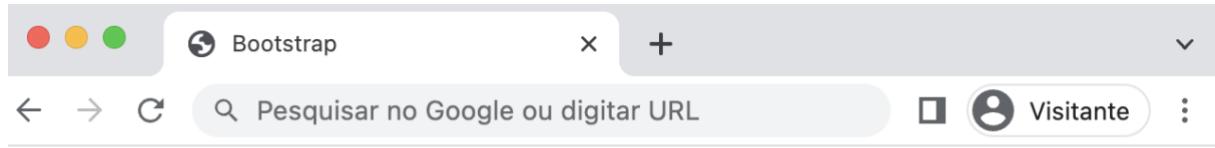
23           <a class="nav-link ms-3 my-1" href="#item-3-2">Item 3-
2</a>
24           </nav>
25       </nav>
26
27   </nav>
28 </div>
29
30   <!-- coluna à direita -->
31 <div class="col-8">
32     <div data-bs-spy="scroll" data-bs-target="#menu-rolagem"
33       data-bs-offset="0" tabindex="0"
34       style="position:relative; height:350px; overflow:auto;">
35
36     <h4 id="item-1">Item 1</h4>
37     <p>...</p> <!-- Preencher com texto -->
38     <h5 id="item-1-1">Item 1-1</h5>
39     <p>...</p>
40
41     <h4 id="item-2">Item 2</h4>
42     <p>...</p>
43
44     <h4 id="item-3">Item 3</h4>
45     <p>...</p>
46     <h5 id="item-3-1">Item 3-1</h5>
47     <p>...</p>
48     <h5 id="item-3-2">Item 3-2</h5>
49     <p>...</p>
50
51   </div>
52 </div>
53 </div>
```

Note que das linhas 38 a 49 há vários parágrafos apenas com “...”. Para testar essa funcionalidade, substitua isso por um texto maior (como por exemplo, o *lorem ipsum*, disponível em <https://www.lipsum.com/>).

Breadcumb

Menus de localização (*breadcrumb*) indicam a localização da página atual com base em uma hierarquia de navegação com barras separadoras adicionadas automaticamente via CSS.

Figura 67. *Breadcrumb*. A página atual se encontra em Home > Biblioteca > Livros.



[Home](#) / [Biblioteca](#) / [Livros](#)

Fonte: próprio autor.

Podemos implementar isso da seguinte forma:

```

1 <!-- Menus de localização -->
2 <nav aria-label="breadcrumb">
3   <ol class="breadcrumb">
4     <li class="breadcrumb-item"><a href="#">Home</a></li>
5     <li class="breadcrumb-item"><a href="#">Biblioteca</a></li>
6     <li class="breadcrumb-item active">Livros</li>
7   </ol>
8 </nav>
```

Paginação

Menus de paginação podem ser criados usando a classe “.pagination”. Cada item da lista deve receber a classe “.page-item” e os links a classe “.page-link”. Observe o código a seguir:

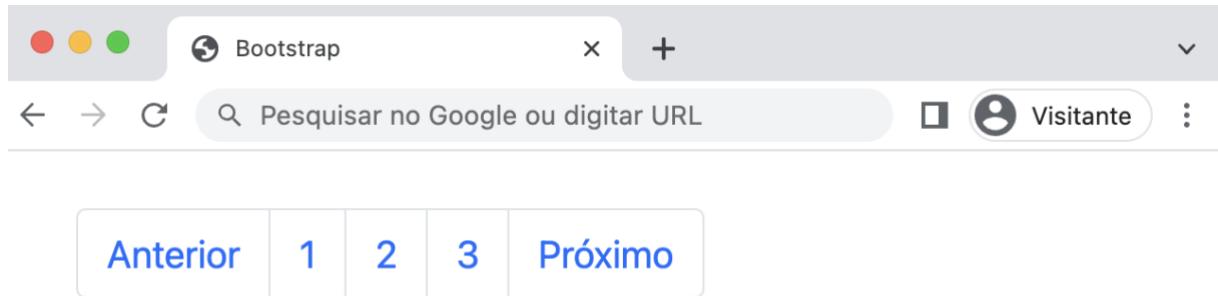
```

1 <!-- Paginação -->
2 <nav>
3   <ul class="pagination">
4     <li class="page-item">
5       <a class="page-link" href="#">Anterior</a>
6     </li>
7
8     <li class="page-item"><a class="page-link" href="#">1</a></li>
9     <li class="page-item"><a class="page-link" href="#">2</a></li>
10    <li class="page-item"><a class="page-link" href="#">3</a></li>
11
12    <li class="page-item">
```

```
13      <a class="page-link" href="#">Próximo</a>
14    </li>
15  </ul>
16 </nav>
```

Esse código produz a seguinte interface:

Figura 68. Paginação.



Fonte: próprio autor.



Bootstrap icons possui mais de 1600 opções de ícones para auxiliar você a construir seus sites.

Bootstrap icons

Bootstrap icons é o pacote de ícones disponibilizado inicialmente pela versão 5 do Bootstrap. Ele possui atualmente mais de 1600 ícones e pode ser obtido no site: <https://icons.getbootstrap.com/>.

Figura 69. Logo do Bootstrap icons.



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

Podemos declarar o Bootstrap icons para uso em uma página HTML com o seguinte código:

```
<link rel="stylesheet"  
      href="https://cdn.jsdelivr.net/npm/bootstrap-  
      icons@1.8.1/font/bootstrap-icons.css">
```

Há outras formas de realizar a instalação, como por exemplo, usando npm ou baixando os arquivos do código fonte. Acesse <https://icons.getbootstrap.com/#install> para saber mais detalhes ou obter o link de acesso da versão mais recente via CDN.

Para declarar um ícone, podemos usar a tag <i> da seguinte forma:

<i class=" [código-do-ícone]">

Por exemplo:

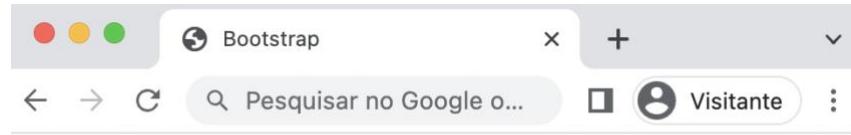
```

1 <!-- ícone de página inicial -->
2 <i class="bi bi-house-fill"></i>
3
4 <!-- ícone de gráfico -->
5 <i class="bi bi-bar-chart">
6
7 <!-- ícone de pessoa -->
8 <i class="bi bi-person-circle">
9
10 <!-- ícone de download -->
11 <i class="bi bi-arrow-down-circle-fill"></i>
12
13 <!-- ícone de ajuda -->
14 <i class="bi bi-question-circle-fill"></i>

```

Neste exemplo, apresentamos cinco diferentes ícones (consulte a documentação para encontrar outros tipos).

Figura 70. Exemplos de ícones.

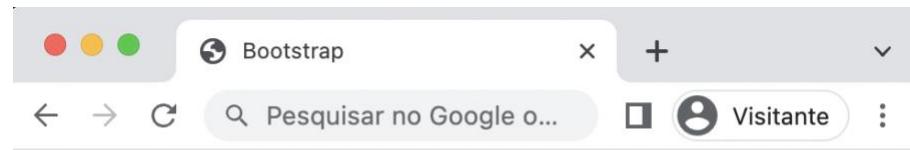


Fonte: próprio autor.

Você pode alterar a cor dos elementos mexendo nas propriedades de texto (por exemplo, utilize as classes `text-[cor]`, como a `text-primary`). Para alterar o tamanho mexa nas propriedades que alteram o tamanho da fonte, como `font-size`. Por exemplo:

```
1 <!-- ícone home -->
2 <i class="bi bi-house-fill text-primary"></i>
3
4 <!-- ícone de gráfico -->
5 <i class="bi bi-bar-chart text-warning">
6
7 <!-- ícone de pessoa -->
8 <i class="bi bi-person-circle text-success">
9
10 <!-- ícone de download -->
11 <i class="bi bi-arrow-down-circle-fill text-danger"></i>
12
13 <!-- ícone de ajuda => TAMANHO AUMENTADO -->
14 <i class="bi bi-question-circle-fill text-info"
15      style="font-size: 30px"></i>
```

Figura 71. Colorindo ícones.



Fonte: próprio autor.

Conclusões

Aqui apresentamos uma breve introdução à nova versão do framework Bootstrap: a versão 5. Com Bootstrap v5 produzir sites com *design* elegante e atrativo se tornou bem mais simples e fácil.

Para saber mais, acesse:

<https://getbootstrap.com/>

Referências bibliográficas

Bootstrap. **Documentação oficial.** Disponível em <https://getbootstrap.com/docs/5.1/getting-started/introduction/>. Acesso em 31 de mar. de 2022.

MARIANO, D.; DE MELO-MINARDI, R. **Introdução à Programação Web para Bioinformática: HTML, CSS, PHP and JavaScript.** [s.l.] Independently Published, 2017.