

BootStrap 4 – Resumo

1-Utilizando o Bootstrap na sua página

- 1-Incorporar o Bootstrap na sua página. Os arquivos são javascripts e CSS.
- 2-Atribuir as classes `class="x"` ao elemento

2-Recursos quanto ao tamanho da tela:

O browser que está exibindo a página tem um tamanho de tela (largura) que depende do seu hardware.
O Bootstrap classifica essa largura em:

Tamanho	Sigla	Tipo	Largura(px)	prefixo	Larg.máx. container(px)
Extra Small	-	C5	< 576	.col-	Não tem (auto)
Small	SM	C4	>= 576	.col-sm-x	540
Medium	MD	C3	>= 768	.col-md-x	720
Large	LG	C2	>= 992	.col-lg-x	960
Extra Large	XL	C1	>= 1200	.col-xl-x	1140

Nota: O Extra Small não tem sigla e é utilizado como default quando omitimos as letras SM, MD, LG e XL

x pode variar de 1 a 12.

Os tamanhos mais usados são:

320 pixels	Celular no modo retrato
480 pixels	Celular no modo paisagem
600 pixels	Tablets pequenos, 7 pol ou menos(kindle-600*800)
768 pixels	Tablets maiores que 7 pol no modo retrato(ipad-768*1024)
1024 pixels	Tablets no modo paisagem e monitores antigos
1200 pixels	Monitores wide screen

O número de colunas sempre é 12 (que é o GRID).

A largura da gutter (padding entre o elemento e o GRID) é de 30px (15 px de cada lado da coluna)
São alinháveis e podem ser ordenadas.

Importante: O conceito de Breakpoints é definido como sendo o ponto que a tela atinge tal largura, ou seja, tal tamanho. Por exemplo, o ponto sm é atingido quando a largura é maior que 575 e menor que e menor que 768. Portanto ao aumentar a tela e ela atingir 575 px é disparado o **breakpoint** sm e quando a tela está maior e diminuimos o tamanho dela ao atingir 767 ativa o **breakpoint** sm.

Portanto temos as seguintes faixas para os breakpoints

<u>Breakpoint</u>	<u>Min(px)</u>	<u>Max(px)</u>
Extra Small	0	575
Small	576	767
Medium	768	991
Large	992	1199
Extra Large	1200	>1200

Portanto:

- 1-Ao aumentarmos a tela e atingirmos a largura mínima atingimos um breakpoint
- 2-Ao diminuirmos a tela atingirmos a largura máxima atingimos um breakpoint

3-Importante – Maneira como o Bootstrap formata os elementos

O Bootstrap é **mobile first**, ou seja, para ele estamos sempre utilizando um dispositivo móvel para ver a página.

Portanto ele costuma adotar alguns padrões como colocar apenas um objeto por linha (um texto, um botão), dimensiona/formata o elemento para que ocupe toda a linha e com um visual muito bom e também quando redimensionamos a largura do browser/página ele dimensiona os objetos para que caibam no novo formato (encolhe-estica bordas, fundos, etc.).

4-Recursos de formatação de textos:

h1 a h6	define tamanho, altura espaçamento do item
display-1 a display-4	define tamanho, altura espaçamento do item como h1 a h6
lead	destaca o item do normal como se “liderasse”
text-monospace	formata o texto monoespaçado
text-truncate	trunca o texto no final da linha corrente (coloca ...)
text-uppercase	formata a fonte como maiúscula
text-lowercase	formata a fonte como minúscula
text-capitalize	formata a primeira letra como maiúscula e as demais minúsculas
text-right	alinha o texto à direita
text-left	alinha o texto à esquerda
text-center	alinha o texto ao centro
text-justify	alinha o texto justificado (mesma largura cobrindo toda a linha)
font-weight-bold	formata o texto como negrito
font-weight-normal	formato padrão de texto
font-italic	formata o texto como itálico
blockquote	destaca do blockquote padrão do HTML
blockquote-footer	formata o rodapé do blockquote bem melhor que o html
list-unstyled	retira o estilo da lista (ul li)
list-inline	coloca os elementos da lista na mesma linha (ul li) usada em conjunto com a classe list-inline-item para cada item.

5-Recursos de alinhamento de textos:

Nota: Caso a tela seja menor que a definição prevalece o alinhamento do html padrão

text-sm-right	aplica o alinhamento se a tela for sm (small)
text-md-right	aplica o alinhamento se a tela for md (médium)
text-lg-right	aplica o alinhamento se a tela for lg (large)
text-xg-right	aplica o alinhamento se a tela for xg (extra-large)

6-Recursos de Fundos:

bg-success	coloca a cor de fundo verde
bg-warning	coloca a cor de fundo laranja

7-Recursos de Formatação de exibição:

d-inline	exibe o elemento como se fosse um elemento de linha
d-block	exibe o elemento como se fosse um elemento de bloco
d-inline-block	exibe o elemento como se fosse um elemento de linha e em bloco Possui as qualidades do inline mas as formatações margin, padding, width, height são utilizadas (vide item F acima).
link	o objeto (um botão por exemplo) fica parecendo um link.

8-Recursos para elementos flutuantes:

float-left	faz o elemento flutuar a esquerda da página
float-right	faz o elemento flutuar a direita da página
clear-fix	limpa o float dos elementos filhos
float-none	define que o elemento não deve flutuar

Importante: Quando os elementos são exibidos como flutuantes a altura deles não é contada e o elemento a seguir pode ser exibido em cima do flutuante por espaçamento incorreto do elemento. Portanto, um item pode 'encavalhar' no outro, ou seja, um ser exibido em cima do outro.

9-Recursos para posicionamento fixo do elemento:

fixed-top	fixa o elemento no topo da página
fixed-bottom	fixa o elemento no rodapé da página

10-Recursos para float responsivo:

float-sm-right	faz o elemento flutuar à direita se a tela for SM(Pequena)
float-md-right	faz o elemento flutuar à direita se a tela for MD(Média)
float-lg-right	faz o elemento flutuar à direita se a tela for LG (Larga)
float-xg-right	faz o elemento flutuar à direita se a tela for XG (extralarga)
sticky-top	quando o elemento atingir o topo da página e o usuário deslocar a página para cima o elemento fica grudado no topo, não é rolado como os demais elemento da página, até que o deslocamento da página faça ele ser exibido fora do topo e aí ele volta a ser um elemento comum.

11-Cores padrão:

primary	azul claro
secondary	cinza médio
success	verde
info	ciano
warning	amarelo
danger	vermelho/púrpura
light	muito perto de branco – para fazer contraste com preto
dark	muito perto de preto – para fazer contraste com branco
white	branco – para fazer contraste com preto
black-50	preto com 50% de opacidade

As cores podem ser aplicadas em:

Texto e links	Exemplo: text-primary
Cor de fundo	Exemplo: bg-primary

12-Recursos de margin e padding:

Nota1: são utilizadas letras para designar o item:

t=top
b=bottom
l=left
r=right
x=margem no eixo x
y=margem no eixo y
m=margem (em todos os lados)
p=padding (em todos os lados)

Nota2: São utilizados os números de 0 a 5 para definir 0(menor) e 5(maior)

<u>Estilo</u>	<u>funcionalidade</u>	
mt-0	Margem topo 1	(não é zero px mas sim zero rem)
mb-1	Margem rodapé 1	
ml-2	Margem esquerda 2	
mr-3	Margem direita 3	
mx-4	Margem direita e esquerda 4	
my-5	Margem superior e inferior 5	
mm-1	todas as margens de 1 rem	

pt-0	Padding topo 1	(não é zero px mas sim zero rem)
pb-1	Padding rodapé 1	
pl-2	Padding esquerda 2	
pr-3	Padding direita 3	
px-4	Padding direita e esquerda 4	
py-5	Padding superior e inferior 5	
pp-1	Padding em todos os lados de 1	

13-Recursos para definir tamanhos dos elementos:

13.1-Larguras - Porcentagem

w-25	25%
w-50	50%
w-75	75%
w-100	100%
w-auto	toda a largura disponível

13.2-Alturas - Porcentagem

h-25	25%
h-50	50%
h-75	75%
h-100	100%
h-auto	100% da altura do elemento

14-Recursos de bordas – Exibir a borda

border	Todas
border-top	borda superior
border-bottom	borda inferior
border-left	borda esquerda
border-right	borda direita

15-Recursos de bordas – Cores

Border-primary	cor primária – azul claro
	Aplicável a todas as cores do item 8

16-Recursos de Bordas – Arredondamento

rounded	arredonda todas as laterais
rounded-top	arredonda apenas a parte superior
rounded-bottom	arredonda apenas a parte inferior
rounded-right	arredonda apenas a parte direita
rounded-left	arredonda apenas a parte esquerda
rounded-circle	arredonda todas as laterais mais acentuada que a rounded

17-Media types

São formatações típicas para um tipo de dispositivo.

Os tipos de dispositivos podem ser um celular, um tablet, um notebook, um desktop e a tela deles variam muito especialmente em largura.

Eu posso definir que se for para exibir a página no vídeo utilize uma css e se for imprimir utilize outra css.

Exemplos:

Exibição no vídeo:

```
<link rel="stylesheet" type="text/css" media="screen" href="estilotela.css">
```

Para impressão:

```
<link rel="stylesheet" type="text/css" media="print" href="estiloimp.css">
```

18-Media queries

São formatos que são aplicados apenas em determinada condição.

Ex1: @media (min-width:576px){ xxx}

Esta formatação será aplicada somente se o dispositivo tiver um tamanho maior ou igual a 576 pixels.

Ex2: @media (max-width:576px){ xxx}

Esta formatação será aplicada somente se o dispositivo tiver um tamanho menor que 576 pixels.

19-Botões

btn	define que um botão do bootstrap colocar este parâmetro sempre antes das tags de botões abaixo.
close	formata o botão para fechar alertas. Veja item 24. Caso deseje colocar um x bem estiloso nela use ×

19.1-Botões-Cores

Veja item 8

btn-primary	azul claro
btn-secondary	cinza médio
btn-success	verde
btn-info	ciano
btn-warning	amarelo
btn-danger	vermelho/púrpura
btn-light	muito perto de branco – para fazer contraste com preto
btn-dark	muito perto de preto – para fazer contraste com branco
btn-white	branco – para fazer contraste com preto
btn-black-50	preto com 50% de opacidade

19.2-Formatação de tags input

As tags input tipo submit, button, reset podem ser formatadas pelos itens 19.1 e ficam com cores (e alguns comportamentos) semelhantes a um botão comum.

19.3-Bordas de botões

Veja item 8

btn-outline-primary	azul claro
btn-outline-secondary	cinza médio
btn-outline-success	verde
btn-outline-info	ciano
btn-outline-warning	amarelo
btn-outline-danger	vermelho/púrpura
btn-outline-light	muito perto de branco – para fazer contraste com preto
btn-outline-dark	muito perto de preto – para fazer contraste com branco

20.1-Navegação simples 1

nav	define que são itens de navegação
nav-item	define um item de navegação
nav-link	define que o item de navegação deve ser tratado como um link

Exemplo:

```
<ul class="nav">
  <li class="nav-item"><a href="#" class="nav-link">Home</a></li>
  <li class="nav-item"><a href="#" class="nav-link">Sobre</a></li>
  <li class="nav-item"><a href="#" class="nav-link">Serviços</a></li>
</ul>
```

20.2-Navegação com item selecionado (pills)

nav-pills	exibem o item selecionado do menu como uma pílula, com um fundo diferenciado. Arredonda a borda do fundo do item.
active	define que o item é o item selecionado por default

Exemplo:

```
<ul class="nav nav-pills">
  <li class="nav-item active"><a href="#" class="nav-link">Home</a></li>
  <li class="nav-item"><a href="#" class="nav-link">Sobre</a></li>
  <li class="nav-item"><a href="#" class="nav-link">Serviços</a></li>
</ul>
```

20.3 – Navegação por abas (tabs)

nav-tabs	exibe os itens de navegação em abas distintas.
----------	--

Exemplo:

```
<ul class="nav nav-tabs">
  <li class="nav-item active"><a href="#" class="nav-link">Home</a></li>
  <li class="nav-item"><a href="#" class="nav-link">Sobre</a></li>
  <li class="nav-item"><a href="#" class="nav-link">Serviços</a></li>
</ul>
```

20.4-Navegação – Alinhamento da barra de navegação em relação a sua exibição

Nota: Os itens do menu são, por default, alinhados esquerda.

justify-content-center	exibe os itens do menu centralizado na página
justify-content-end	exibe os itens do menu alinhados a direita da página
flex-column	exibe os itens do menu em colunas, ou seja, um embaixo do outro. Utilizar junto com a classe nav-pills.
disabled	exibe o item do menu como inativo, desabilitado

Exemplo:

```
<ul class="nav nav-tabs justify-content-center">
  <li class="nav-item active"><a href="#" class="nav-link">Home</a></li>
  <li class="nav-item"><a href="#" class="nav-link">Sobre</a></li>
  <li class="nav-item"><a href="#" class="nav-link">Serviços</a></li>
</ul>
```

20.5-Navegação simples 2

nav-bar	define que o item é uma barra de navegação
navbar-expand-sm	a expansão da nav bar deve ser feita com a medida sm
navbar-expand-md	a expansão da nav bar deve ser feita com a medida md
navbar-expand-lg	a expansão da nav bar deve ser feita com a medida lg
navbar-expand-xl	a expansão da nav bar deve ser feita com a medida xl

navbar-dark	define o estilo para a barra de navegação (veja item 8)
bg-light	define a cor de fundo da barra de navegação (veja item 8)
nav-bar-nav	é o item que contém os itens da barra de navegação
ml-auto	define que a margem esquerda deve ser automática Alinha os itens da barra de navegação à esquerda (exceto o brand).
nav-item	define que é um item da barra de navegação
nav-link	define que o item da barra de navegação deve ser tratado como link.
Navbar-brand	define que é o item principal da barra de navegação. Este item é destacado em relação aos demais itens da barra de navegação.

Nota: é o expand que deixa os itens da barra de navegação na horizontal.
O default é que os itens da barra de navegação sejam exibidos na vertical.

21 - Barra de Navegação (ou Menu Responsivo)

É uma barra de navegação que é exibido de uma maneira quando a tela é pequena e de outra maneira quando a tela é maior.

Como opção quando a tela é muito pequena o menu é ocultado e fica no seu lugar um botão que a gente chama de menu 'hamburger' porque parece um hamburger e é muito resumido. Ao clicar no botão o menu é exibido e ao clicar novamente ele some.

Nota: a barra de navegação normalmente é colocado dentro de uma div com a classe="navbar-expand-sm" e com isto o menu é escondido e o botão só é exibido quando a tela é sm. Quando a largura da tela for md, lg, xl todo os itens do menu são exibidos.

navbar-toggler	é a classe a ser atribuída ao botão hamburger
navbar-toggler-icon	é a classe que exibe o botão hamburger
data-toggle="collapse"	classe a ser atribuída ao botão hamburger
data-target="#alvo"	é o id da div de menu que deve ser ser exibida ou escondida ao clicar no botão hamburger
collapse	é a classe a ser atribuída ao menu que deverá ser escondido ou exibido
navbar-collapse	esta classe faz com que o menu que esta sendo exibido/escondido seja exibido na vertical na próxima linha

Exemplo:

```
<nav class="navbar navbar-expand-sm navbar-dark bg-dark">
  <a href="#" class=navbar-brand">ItemFixoMenu</a>           se existir item fixo
  <button class="navbar-toggler" data-toggle="collapse" data-target="#nav-target">
    <span class="navbar-toggler-icon"></span>
  </button>
  <div class="collapse navbar-collapse" id="nav-target">
    <ul class="navbar-nav ml-auto">
      <li class="nav-item">
        <a href="#" class="nav-link">Home</a>
      </li>
      <li class="nav-item">
        <a href="#" class="nav-link">Sobre</a>
      </li>
      <li class="nav-item">
        <a href="#" class="nav-link">Serviços</a>
      </li>
    </ul>
  </div>
</nav>
```

21.1-Barra de navegação com formulário

Normalmente este formulário faz parte dos itens da barra de navegação e com isto ela será exibida/escondida junto com os itens do menu

form-inline	exibe os itens do formulário na mesma linha
form-control	controle que recebe os dados do form
placeholder	valor exibido de fundo da caixa de texto – dica de como preencher

Exemplo:

```
<nav class="navbar navbar-expand-sm navbar-dark bg-dark">
  <a href="#" class=navbar-brand">ItemFixoMenu</a> se existir item fixo
  <button class="navbar-toggler" data-toggle="collapse" data-target="#nav-target">
    <span class="navbar-toggler-icon"></span>
    <button>
      <div class="collapse navbar-collapse" id="nav-target">
        <ul class="navbar-nav ml-auto">
          <li class="nav-item">
            <a href="#" class="nav-link">Home</a>
          </li>
          <li class="nav-item">
            <a href="#" class="nav-link">Sobre</a>
          </li>
          <li class="nav-item">
            <a href="#" class="nav-link">Serviços</a>
          </li>
        </ul>
        <form class="form-inline" >
          <input type="text" class="form-control" placeholder="pesquisar...">
          <button class="btn btn-outline-success">ok</button>
        </form>
      </div>
    </button>
</nav>
```

21.2-Barra de navegação com menu DropDown

Um dos itens do menu tem um submenu que é exibido estilo dropdown.

dropdown	define que é o container do menu dropdown
dropdown-toggle	é o elemento que exhibe/esconde o menu dropdown
navbar	indica que é o elemento que contém os itens da barra de navegação
nav-item	indica que é um item do menu de navegação
data-toggle="dropdown"	indica que o item que exhibe/esconde o menu dropdown

Nota: Na barra de navegação usamos o data-toggle="collapse" porque queríamos o efeito de esconder(collapse) no menu.

Exemplo:

```
<nav class="navbar navbar-expand-sm navbar-dark bg-dark">
  <a href="#" class=navbar-brand">ItemFixoMenu</a> se existir item fixo
  <button class="navbar-toggler" data-toggle="collapse" data-target="#nav-target">
    <span class="navbar-toggler-icon"></span>
  </button>
  <div class="collapse navbar-collapse" id="nav-target">
    <ul class="navbar-nav ml-auto">
      <li class="nav-item">
        <a href="#" class="nav-link">Home</a>
      </li>
```

```

<li class="nav-item">
  <a href="#" class="nav-link">Sobre</a>
</li>

<li class="nav-item dropdown">
  <a href="#" class="nav-link dropdown-toggle" data-
toggle="dropdown">Serviços</a>
  <div>
    <a href="#" class="dropdown-item">Serviço1</a>
    <a href="#" class="dropdown-item">Serviço2</a>
    <a href="#" class="dropdown-item">Serviço3</a>
  </div>
</li>
</ul>
</div>
</nav>

```

21.3-Barra de navegação – Posicionamento

fixed-top fixa no topo da página
fixed-bottom fixa no rodapé da página
sticky-top cola no topo quando o elemento atingir o topo da tela
sticky-bottom cola no rodapé quando o elemento atingir o rodapé da tela

22-Listas

Formata as listas apresentadas na página.

Como os itens são agrupados, o primeiro e o último item tem as bordas superiores (primeiro) e inferiores (último) arredondados

Pode formatar:

```
ul            li
ul            a
```

list-group informa que uma lista agrupada de itens
list-group-item informa que é um item da lista agrupada de itens
list-group-item-estilo o estilo a ser aplicado é o mesmo do item 8 (primary, secondary,etc...) este estilo é aplicado no item, não só na cor e fundo.
list-group-flush apenas as bordas superior e inferior são exibidas, as laterais não.
Active determina que é o item ativo da lista

22.1-Listas-Badges

Badge é um item apendado como se fosse um crachá, determina alguma info do elemento.

badge indica que é um item tipo badge
badge-primary indica como o item deve ser exibido, cor, fundo, etc...

Exemplo:

```

<ul class="list-group">
  <li class="list-group-item"> Menu item de lista 1</li>
  <li class="list-group-item" > Menu item de lista 2</li>
  <li class="list-group-item" >
    Caixa de entrada
    <span class="badge badge-primary"120</span>
  </li>
  <li class="list-group-item" > Menu item de lista 4</li>

```

23-BreadCrumbs (do inglês, migalhas de pão)

São muito utilizados na navegação para informar aonde que o usuário está.
Exibe os itens na mesma linha separados por /.

breadcrumb	informa que deve ser exibido com o estilo breadcrumb (como um menu)
breadcrumb-item	informa que é um item do breadcrumb
active	indica que é o item selecionado do breadcrumb

Exemplo:

```
<ol class="breadcrumb">
  <li class="breadcrumb-item">Home</li>           item não clicável
</ol>
<ol class="breadcrumb">>
  <li class="breadcrumb-item">
    <a href="#">Home</a>           item clicável
  </li>
  <li class="breadcrumb-item">Noticias</li>       item não clicável
</ol>
```

24-Formulários

Notas:

1-Caso exista elementos agrupados com a mesma finalidade (por exemplo, um label com o texto 'Digite aqui o seu nome' e um textbox para que o usuário digite o nome) é correto agrupar os dois elementos numa mesma div o num elemento agrupador de itens para que ao aplicar o estilo css ou js ambos sejam formatados juntos e trabalhem juntos.

2-Quando os controles são formatados pelo Bootstrap normalmente eles ocuparão até o final da linha que estão sendo exibidos. Caso o browser seja redimensionado estes controles são redimensionados também ocupando o restante da linha na nova dimensão.

form-group	indica que o elemento é um elemento agrupador de itens do formulário
form-control	indica que é um controle de formulário. Pode ser aplicada nas tags: input, select, textareas entre outros. O tamanho do controle será o normal.
form-control-sm	indica que o controle de formulário deve ser exibido da menor maneira possível.
form-control-lg	indica que o controle de formulário deve ser exibido da maior maneira possível.
readonly	indica que o elemento deve ser exibido inativo para que o usuário não digite ou altere o conteúdo do elemento.
form-control-file	informa que o input tipo file deve ser formatado pelo Bootstrap
custom-file	substitui a classe form-group para elementos de file-upload
custom-file-input	define que o input tipo file deve ser formatado 'Custom' pelo Bootstrap.
custom-file-label	formata o label antes do input type file no estilo 'Custom'.
Custom-range	formata um input type range no estilo 'Custom' pelo Bootstrap

24.1-Formulários Inline

É quando todos os elementos do formulário devem ser exibidos em uma única linha.

É muito utilizado no topo da página quando temos a opção de login do usuário.

Fica tudo junto: inputs tipo texto e botões tipo submit

form-inline	indica que o form deve ser exibido com todos seus elementos na mesma linha
mr-2	margem a direita de 2 rem

25-Formatação de validação de formulários

Formata o controle como se tivesse sido validado e o resultado formata o controle, se sucesso ou se houve falha.

is-valid	define que o controle deve ser formatado como se fosse válido. A cor normal dele é verde.
is-invalid	define que o controle deve ser formatado como se fosse inválido. A cor normal dele é vermelho.
invalid-feedback	formata o controle relacionado a mensagem de feedback dada ao usuário quando a validação é inválida.
valid-feedback	formata o controle relacionado a mensagem de feedback dada ao usuário quando a validação é válida.

26-Input Groups

Muitos controles são usados juntos e o Bootstrap criou algumas maneiras de personalizar a exibição e utilização desses controles. É o caso de inputs de email, urls customizadas, valores monetários entre outros.

form-control	define que o controle é um elemento de um formulário
input-group	define que o element agrupa um conjunto de inputs
input-group-prepend	o conteúdo deve ser anexado no início do elemento, na parte esquerda e o elemento logo abaixo é colocado na mesma linha deste elemento.
Input-group-text	informa que o elemento é apenas um texto a ser agregado ao controle. Importante: Podemos aplicar esta classe numa div que contém um

Controle tipo checkbox ou radio. Neste caso o controle será renderizado da mesma maneira que o texto seria.

input-group-append	o conteúdo deve ser anexado no fim do elemento, na parte direita e o elemento logo abaixo é colocado na mesma linha deste elemento.
--------------------	---

27-Alertas

Não é um alerta como o do JavaScript mas sim um box arredondado com uma mensagem de texto. A finalidade desses alertas é emitir mensagens de feedback para os usuários a respeito de um fato relevante.

alert	define que o item é um alerta e modifica o espaçamento do item.
alert-primary	define a cor de fundo de um alerta. Veja o item 8 para as demais cores.
alert-dismissible	informa que o alerta pode ser escondido.

Exemplo:

```
<div class="alert" alert-success alert-dismissible">  
  <button class="close" type="button" data-dismiss="alert">  
    &times;  
  <button>  
    <strong>Fechar</strong>  
  Alerta com o botão fechar  
</div>
```

28-Barras de progresso

progress	define que é o container uma barra de progresso (uma div por exemplo)
----------	---

No Bootstrap podemos agrupar diversas barras de progresso na mesma linha.

progress-bar	define que o elemento é um progress-bar
--------------	---

w-75 define que o valor da barra de progresso é de 75% da área total ocupada pela barra de progresso. O número após o w- é sempre uma porcentagem.

Pode-se através do style definir o valor do progressbar – style="width:45%;"

Nota: se na div com a classe progressbar existir um texto ele será exibido junto e dentro da barra de progresso.

bg-success define uma cor de fundo para a barra de progresso.

As cores possíveis estão no item 8.

style="height:2px;" define que a altura da barra é de 2px...bem fininha. Logicamente neste caso o texto não pode ser exibido junto com a barra de progresso.

progress-bar-striped define que a barra de progresso é listrada

progress-bar-animated define que a barra de progresso é animada...as listas se movem.

Exemplo:

```
<div class="progress">  
  <div class="progress-bar w45">45%</div>  
</div>
```

29-Tabelas

Notas:

<table> define uma tabela

<thead> define o header/ cabeçalho da tabela

<th> define cada campo do <thead>

<tbody> define o corpo (conjunto de dados - conteúdo) da tabela

<tfoot> define o rodapé da tabela

Pode-se fazer o tbody ter um scroll horizontal enquanto que o thead não seja deslocado.

Na impressão os campos em thead são impressos no início de cada nova página.

table formata o container como uma tabela. Por default ela tem divisões nas linhas, fundo branco, mas não tem divisões nas colunas.

table-dark define que o tema dark para a tabela, ou seja, o fundo deve ser dark e os caracteres de texto com a cor branca.

Pode-se aplicar as demais opções de cores (veja item 8).

table-striped faz a tabela ficar com as linhas zebradas

table-bordered exhibe todas as bordas (coluna e linha) da tabela

table-borderless exhibe a tabela sem borda alguma

thead-dark define que deve ser aplicado o tema dark apenas no header da tabela.

table-success formata uma linha da tabela específica para o tema desejado.

Veja as cores no item 8.

table-sm formata a tabela para o tamanho sm, o menor

table-responsive Caso a tela seja insuficiente para exibir todas as colunas, as colunas mais a direita podem ser ocultadas e uma barra de rolagem horizontal permite deslocar a janela horizontalmente para que se possa ver as colunas que foram ocultadas.

29.1-Tabelas-Paginação

Caso a tabela tenha muitos itens um recurso comum é paginar, ou seja, exibir de 100 em 100 linhas, por exemplo, e criar um controle para navegar entre as páginas de dados.

O bootstrap tem formatos específicos para paginação.

O default do bootstrap é colocar o controle de paginação à esquerda da linha.

O caractere « é usado para exibir << (Anterior)

O caractere » é usado para exibir >> (Posterior)

pagination define que o controle é um container de itens de paginação (aplicar na ul)

justify-content-center	exibe os controles de paginação no centro da linha (ul)
justify-content-end	exibe os controles de paginação no fim da linha (à direita) (ul)
pagination-lg	exibe os controles de paginação com o formato grande
pagination-sm	exibe os controles de paginação com o formato pequeno
page-item	define que é um dos itens exibidos na paginação (aplicar no li)
page-link	define que o item é um link para a página n da paginação.
active	indica que é o item ativo da lista de itens de paginação.
disabled	exibe o item desabilitado, na cor cinza e o mouse hover não funciona.

30-Leitores de tela

São conteúdos que não são exibidos na página, mas o leitor de tela irá ler para o usuário o texto deste tipo de item. Por exemplo « e » (veja item 26.1) não são lidos pelo leitor de tela e devemos colocar um texto específico para estes itens se quisermos que o usuário saiba que eles existem.

sr-only	screen reader only. A mensagem não é exibida na tela mas o leitor de tela vai ler sem problemas esta mensagem.
---------	--

31-Cards - Cartões

Formatam o elemento de maneira que fique parecendo um cartão de visitas ou algo bem próximo. Também são muito parecidos com painéis que agrupam itens.

card	define que o elemento agrupa um conjunto de itens que devem ser exibidos como cartão. Coloca uma borda em volta dos objetos internos.
card-body	determina que as tags agrupadas por este elemento pertencem ao corpo do cartão
card-title	determina que o elemento é o título ou cabeçalho do cartão
card-subtitle	determina que o elemento é o sub-título ou cabeçalho do cartão
card-text	determina que o elemento é o texto do corpo do cartão
card-header	determina que o elemento é o cabeçalho do cartão
card-header-tabs	formata os itens do cabeçalho do cartão como tabs
card-img-bottom	informa que a imagem deve ser exibida no rodapé do cartão
card-img-top	informa que a imagem deve ser exibida no cabeçalho do cartão
card-img-overlay	informa que o texto do corpo do cartão deve ser escrito sobre a imagem anteriormente designada na tag anterior.
Card-footer	define que o elemento é o rodapé do cartão
nav	indica que os elementos agrupados pelo cartão são itens de navegação.
nav-item	indica que o item é um item de navegação (um link)
nav-link	indica que o item é um link de navegação pertencente a uma nav.
active	informa que o item é o item ativo dos links de navegação do cartão.
disabled	informa que o item deve ser exibido como desabilitado e o usuário não terá acesso a ele.

Nota: Utiliza text-center para alinhar o texto ao centro

Exemplo:

```
<h3 class="border-bottom">Cartão</h3>
<div class="card">
  <div class="card-header">Cabeçalho do cartão</div>
   <!-- imagem fundo -
  ->
  <div class="card-body">
    <h4 class="card-title">Título do cartão</h4>
    <h6 class="card-subtitle">Subtítulo do cartão</h6>
    <p class="card-text">Lorem ipsum dolor sit amet, consectetur adipiscing elit. Fusce sapien
    dui, feugiat et lorem quis, fringilla maximus nisl. Vivamus sed est pulvinar, aliquet purus
    ut, dictum elit.
    </p>
    <a class="btn btn-outline-primary" href="">Leia mais</a>
  </div>
</div>
```

```

</div>
```

Exemplo de cartão com navegação

```
<h3 class="border-bottom">Cartão com navegação</h3>
<div class="card">
  <div class="card-header">
    <ul class="nav nav-tabs card-header-tabs">
      <li class="nav-item">
        <a class="nav-link active" href="">Link 1</a>
      </li>
      <li class="nav-item">
        <a class="nav-link" href="">Link 2</a>
      </li>
      <li class="nav-item">
        <a class="nav-link disabled" href="">Link 3</a>
      </li>
    </ul>
  </div>
  <div class="card-body">
    <h4>Título do cartão</h4>
    <h6>Subtítulo do cartão</h6>
    <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Fusce sapien dui, feugiat
      et lorem quis, fringilla maximus nisl. Vivamus sed est pulvinar, aliquet purus ut,
      dictum elit.
    </p>
    <a class="btn btn-outline-danger btn-block" href="">Leia mais</a>
  </div>
</div>
```

31.1-Cartões com cor de fundo

`bg-primary` define a cor de fundo do objeto como Primary. Veja as cores no item 8.
`text-white` define que a cor do texto do elemento seja branca
`mb-2` margin bottom 2 – dá um espaçamento entre objetos

31.2-Cartão com bordas

`Border-primary` define que a cor da borda do objeto é primary. Veja cores item 8.

31.3-Cartão com colunas

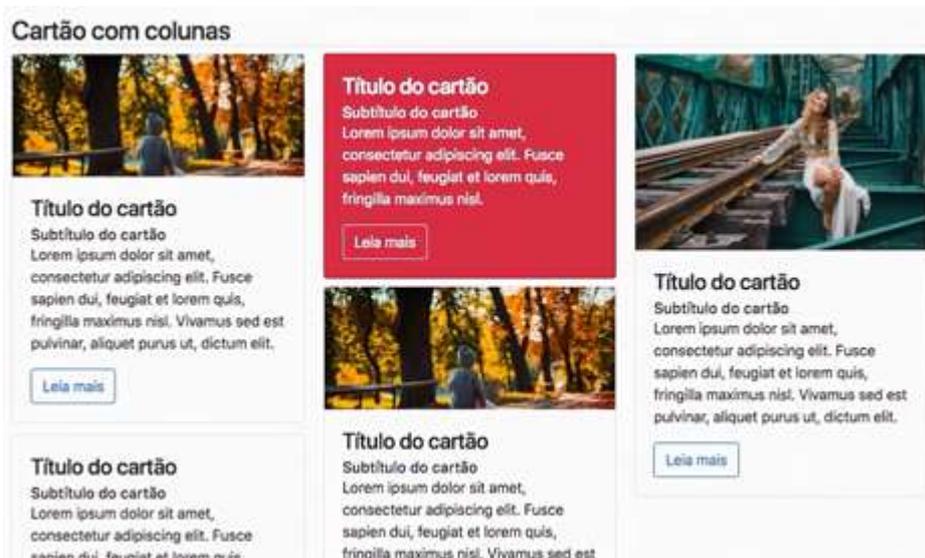
Basta colocar os cartões dentro de uma div agrupadora dos cartões.

Nesta div agrupadora temos uma única classe:

`card-columns` define que é um elemento agrupador de cards no formato de colunas.

Cada cartão agrupado é exibido, por default, da direita para a esquerda e, se não couber na linha, será exibido na linha de baixo.

Caso haja diversos cartões eles ocupam um espaçamento vertical variável dependendo do cartão e com isto cria um efeito interessante.



32-Jumbotron

Cria uma área de destaque para uma página do site com finalidade de destacar um elemento dentre os demais da página

- Jumbotron informa que o agrupamento dos itens feitos por este elemento devem ser bem destacados na página
- display-4 exibe o elemento com tamanho 4, como um header dentro do jumbotron
- lead formata uma tag p com um formato interessante para o jumbotron
- container deve ser atribuído ao elemento que contém todos os elementos do jumbotron. Com isto o alinhamento, espaçamento, distribuição dos elementos dentro do jumbotron ficam melhores.

33-Imagens

Agregando texto e imagens de maneira que fiquem da maneira que queremos na página.

- media Agrega a imagem e o texto dentro das classes agrupadas por este item. Coloca como se fosse inline.
- media-body Informa que os elementos agrupados por este elemento fazem parte do corpo da classe media. O media-body é normalmente é composto por um title, body e um footer. Normalmente antes do elemento media-body temos uma imagem.
- mr-3 exibe o elemento com uma margem direita de 3 rem
- start alinha os elementos componentes da media pelo topo do media-body.
- center alinha os elementos componentes da media pelo centro do media-body.
- end alinha os elementos componentes da media pelo rodapé do media-body.

Nota: O media-body alinha a imagem em relação ao media-body. Mas como podemos ter um título e uma mensagem no media-body parece que a imagem esta fora do centro em relação ao texto do medi-body e isto ocorre porque o media-body coloca uma margem bottom que desalinha (na aparência) a imagem do centro do media-body. Para acertar esse alinhamento coloque a classe mb-0 (margin bottom zero) no texto / parágrafo

33.1-Lista de imagens

Permite exibir uma lista de itens com imagens e texto num formato muito bom.

list-unstyled	remove o marcador de lista da tag ul. Aquele pontinho à esquerda do item.
media	aplicado a cada item da lista (li) indica que os elementos agrupados por este item são do tipo media (img, div, h1 a h6, p)
media-body	informa que as tags agrupadas por este item fazem parte do corpo do elemento media.
mr-2	margin-right 2 rem – separa a imagem do texto.
align-self-center	alinha o elemento ao centro
mb-0	margem inferior de 0 rem – Acerta o alinhamento da imagem com o media-body em relação ao centro.
border-bottom	exibe a borda inferior do elemento
pb-2	padding-bottom 2 rem – dá um espaçamento entre os elementos.
pt-2	padding-top 2 rem – dá um espaçamento entre os elementos.

34-Ícones

Os ícones sempre destacaram uma funcionalidade facilitando o reconhecimento e o acesso a elas. No Bootstrap os ícones podem ser incorporados ao texto de um botão, ter seu tamanho e cor ajustados entre outros efeitos.

Na página <https://getbootstrap.com/docs/4.0/extend/icons/>

existe um monte de sites que possuem um monte de ícones, muitos deles grátis que podem ser baixados e utilizados.

Os ícones são incorporados a página através da tag link, stylesheet e são arquivos .css comuns.

Cada fornecedor tem seu produto e cada um deles tem uma maneira de uso.

Por exemplo, o iconic você usa as classes:

oi	indica que o item é um ícone da open iconic
oi-bolt	indica qual o ícone da coleção que deseja utilizar.
display-1	define o tamanho do ícone na sua exibição
text-danger	define a cor do ícone

34.1-Adicionando o ícone a um botão

Exemplo:

```
<button class="btn btn-danger btn-lg">  
  <i class="oi oi-clock"></i>  
  Clique aqui  
</button>
```



Nota: O texto é opcional, pode ficar apenas o ícone.

35-Sistema de Grid

O Bootstrap divide todo o espaço horizontal da página em 12 grids (colunas) verticais, ou seja, divide verticalmente a página em 12 áreas verticais na mesma linha. A partir daí posso usar esses espaços para exibir meu elemento dividindo os 12 espaços em:

- 1 área de 12 espaços
- 2 áreas com 6 espaços
- 3 áreas com 4 espaços
- 4 áreas com 3 espaços
- 12 áreas de 1 espaço

Note que 12 colunas são a linha inteira do Grid. Você pode usar menos e o que sobrar ficará sem uso para o grid.

Essas colunas são de mesmo tamanho e são redimensionadas conforme redimensionamos a janela do browser. Veja item 3.

Importante: Podemos colocar GRID dentro de GRID, ou seja, dentro de um elemento GRID podemos colocar outro elemento GRID que dividirá o pai em 12 vezes proporcionais.

row	cria uma linha. Esta linha é que será dividida em 12 partes pelo GRID.
col-sm-2	informa ao browser que quando o dispositivo for do tipo sm este elemento deverá ocupar 2 das 12 colunas do grid.
Col-md-10	informa ao browser que quando o dispositivo for do tipo md este elemento deverá ocupar 10 das 12 colunas do grid.

Podemos combinar estas classes para que o mesmo elemento ocupe regiões diferentes em determinados tipos de telas.

Ex.: class= "col-sm-6 col-md-3" vai ocupar 6 colunas no modo sm e 3 no md

35.1-Configuração do Grid

container	agrupa todos elementos dentro do elemento atual e ocupa uns 8% da tela centralizando seu conteúdo
container-fluid	agrupa todos elementos dentro do elemento atual e ocupa 100% da largura da página.
col	define apenas uma coluna do GRID e tem a largura fixada pelo browser que toda a largura da linha repartida com as demais col. Se tiver só uma col, vai ocupar uma linha, se tiver 2 cols, as duas vão ocupar igualmente (50%-proporcionalmente) uma linha e assim por diante.
col-x	define que o elemento vai ocupar x colunas das 12 colunas do GRID. Nota: este elemento vai ocupar sempre o número de colunas definido mesmo ao redimensionar a largura do browser.
col-sm	a definição col é aplicada quando o elemento tiver a largura sm de tela. Vale para md, lg e xl.

Nota: uma col sempre será alinhada a seguir de uma outra row, na mesma linha a não ser que não caiba aí será colocada na linha abaixo.

order-1	informa que o elemento deverá ser a primeira coluna do grid. Podemos colocar de 1 a 12.
Offset-md-1	Quando o elemento for do tamanho md este elemento terá o espaçamento de 1 grid à esquerda, ou seja, este elemento será colocado do grid 2 em diante. O número 1 pode ser mudado para 1 a 12

Matéria extra: Veja Bootstrap mixins Sass. É um assunto longo para este documento.

35.2-Alinhamentos Vertical no Grid em toda linha

align-items-start	alinha os itens no topo do elemento. É o padrão.
align-items-center	alinha os itens no meio do elemento
align-items-end	alinha os itens no rodapé do elemento

35.3-Alinhamentos Vertical no Grid individual por elemento

align-self-start	alinha o item à esquerda da tela. Este é o padrão.
align-self-center	alinha o item no meio da tela
align-self-end	alinha o item à direita da tela.

35.4-Alinhamento Horizontal no Grid.

Utilizado quando não ocupamos as 12 colunas do grid e precisamos alinhar o conteúdo apresentado no GRID em relação a tela

justify-content-start	Alinha a apresentação do GRID à esquerda da tela. É o default.
justify-content-center	Alinha a apresentação do GRID ao centro da tela.
justify-content-end	Alinha a apresentação do GRID à direita da tela.

justify-content-around	Alinha o conteúdo do GRID distribuindo igualmente seus itens pela linha inteira contando com as laterais.
justify-content-between	Alinha o conteúdo do GRID distribuindo igualmente seus itens pela linha inteira sem contar com as laterais.

Importante: Os alinhamentos acima não funcionam só no GRID não...funcionam também em divs e outros elementos html.

36-Utilizando flexControls para alinhar conteúdo da página

No item 32 vimos como alinhar os itens do GRID utilizando o bootstrap.

No entanto, é possível fazer estes alinhamentos no elemento flexbox do html aplicando as mesmas classes do Grid como descrito abaixo.

36.1-Como os itens são apresentados

d-flex	informa que o elemento é um flex control
flex-column	os itens do agrupamento são exibidos um por linha.
flex-row	os itens do agrupamento são exibidos em colunas, um ao lado do outro. É o padrão. As colunas são exibidas da esquerda para a direita.
flex-row-reverse	os itens do agrupamento são exibidos em colunas, um ao lado do outro mas a primeira coluna é alinhado a partir da direita da tela.

36.2-Alinhamento horizontal

d-flex	informa que o elemento é um flex control
São as mesmas opções do item 32.4 – Alinhamento horizontal no Grid.	
Justify-content-Alinhamento (start, center, end, around, between	

36.3-Alinhamento horizontais de colunas

d-flex	informa que o elemento é um flex control
flex-column	Exibe os elementos um em cada linha (cada coluna ocupa uma linha inteira
flex-column-reverse	exibe as colunas na ordem inversa, de baixo para cima

36.4-Alinhamento vertical das colunas

d-flex	informa que o elemento é um flex control
align-items-start	alinham os itens na parte Superior do container. Default.
align-items-center	alinham os itens na parte central do container
align-items-end	alinham os itens na parte inferior do container
align-items-baseline	alinha o texto dos itens na mesma linha base.
align-items-stretch	estica o item para que ocupe todo o espaço da coluna.

36.5-Alinhamento vertical e individual de itens

d-flex	informa que o elemento é um flex control
align-self-start	exibe o elemento no topo do elemento
align-self-center	exibe o elemento no meio do elemento
align-self-end	exibe o elemento no rodapé do elemento
align-self-baseline	exibe o elemento alinhando os textos dos componentes
align-self-stretch	exibe o elemento esticando o item para que ocupe toda a coluna do item

37-Margem e Wrap

Wrap é o recurso de quando o elemento não cabe na largura atual do documento ele é deslocado para a linha inferior logo a seguir.

Quando utilizamos a classe nowrap a largura do documento é aumentada de maneira que todos os elementos caibam na linha.

mr-auto coloca uma margem a direita do elemento com uma largura maior possível de maneira que os elementos ocupem a linha inteira.

Nota: colocando mr-auto e ml-auto é uma das melhores maneiras de centralizar o elemento horizontalmente na linha.

mt-auto coloca uma margem no topo do elemento com a altura maior possível de maneira que os elementos ocupem a linha inteira.

mb-auto coloca uma margem no rodapé do elemento com a altura maior possível de maneira que os elementos ocupem a linha inteira.

Nota: colocando mt-auto e mb-auto é uma das melhores maneiras de centralizar o elemento verticalmente na linha.

d-flex informa que o elemento é um flex control

flex-nowrap expande a largura do documento para que todos elementos fiquem na mesma linha. Neste caso é muito comum exibir uma barra de rolagem horizontal para vermos todos os itens.

flex-wrap exhibe os elementos que cabem na largura atual do documento (sem expandir) e os que não puderem ser colocados na linha serão automaticamente jogados para a linha seguinte inferior.

38-Visibilidade de elementos

No html cansamos de colocar o estilo display="none" para esconder objetos. No bootstrap temos uma classe idêntica.

d-none não exhibe o elemento.

d-inline exhibe o elemento com o estilo em linha (todos juntos, um ao lado do outro)

d-block exhibe o elemento com o estilo bloco (ocupa toda linha)

d-md-block quando a tela atingir os breakpoints do md o browser irá exibir o elemento como bloco.

Nota: Podemos usar sm, md, lg, xl nesta classe.

39-Imagens

39.1-Imagem-Redimensionamento

Quando criamos uma imagem consideramos o tamanho da tela para que ela seja exibida com qualidade. No entanto, como a tela dos dispositivos variam muito de tamanho talvez fosse melhor se a imagem fosse redimensionada dependendo da tela do dispositivo (responsiva). Para isto temos a classe:

img-fluid redimensiona a imagem para que ocupe o espaço horizontal restante até o final da linha corrente ou do container corrente. O redimensionamento é proporcional (tanto na largura como na altura) fazendo com que a imagem não fique distorcida.

39.2-Imagem-Rotacionando

transform:rotate(30deg); este estilo colocada dentro de uma css faz com que a imagem rotacione 30 graus no sentido horário.

40-Barras de rolagem

overflow-x:hidden;	determina que mesmo que o conteúdo peça a barra de rolagem horizontal esta não será exibida. Colocar isto dentro de uma css.
overflow-x:visible;	determina que a barra de rolagem horizontal sempre será exibida mesmo que não seja necessária. Colocar isto dentro de uma css.
overflow-y:hidden;	determina que mesmo que o conteúdo peça a barra de rolagem vertical esta não será exibida. Colocar isto dentro de uma css.
overflow-y:visible;	determina que a barra de rolagem vertical sempre será exibida mesmo que não seja necessária. Colocar isto dentro de uma css.
overflow:hidden;	esconde ambas as barras de rolagem, vertical e horizontal.
overflow:visible;	exibe ambas as barras de rolagem, vertical e horizontal mesmo quando não necessárias.

Nota: Tanto no hidden horizontal quanto no vertical caso haja conteúdo e não esteja sendo exibido não há como visualizar este conteúdo.

Notas: Na aula 139 no final tem uma demonstração de como usar o inspecionar código nativo do browser.

41-Carousel ou Slider (Carrosel ou Slide show)

Este recurso é muito útil quando desejamos alternar a exibição de mais de um item no mesmo espaço. O Bootstrap troca automaticamente o conteúdo.

O conteúdo de cada slide pode ser texto, imagens entre outras coisas.

O carousel pode ser:

41.1-Simples – Alterna os itens sem qualquer interação com o usuário. O bootstrap troca as imagens de tempos em tempos.

41.2-Com controles – O usuário pode clicar num controle e ir para o slide anterior ou em outro controle e ir para o slide posterior.



41.3- Com indicadores - Além dos controles este tipo exibe uma caixa de indicadores dizendo qual slide está sendo apresentado slide 1, slide2, slide3



41.4-Com legendas – Ao trocar o slide é apresentado um novo texto ou mesmo um botão junto com a imagem.



Nota: O carousel utiliza o jquery, popper e bootstrap.

class="carousel slide" data-ride="carousel" indica que os itens agregados por este elemento são um carrossel.

Id="x"	fundamental para que os controles atuem neste item (prev, next).
carousel-inner	agrupa todos os itens do carousel
carousel-item	é um elemento a ser exibido dentro do carousel
active	indica o primeiro item a ser exibido pelo carousel
img-fluid	faz com que a imagem seja redimensionada para caber no container. O default é exibir a imagem original, no tamanho real.

Nota: Os controles são adicionados fora da classe inner e normalmente são links.

class="carousel-control-prev" data-slide="prev"	é o controle que permite voltar ao slide anterior
carousel-control-prev-icon	é o ícone para slide anterior
class="carousel-control-next" data-slide="next"	é o controle que permite ir para o slide posterior
carousel-control-next-icon	é o ícone para slide posterior

Nota: nos controles, tanto no next como no prev, eles estão ligados pelo id da classe carousel e pelo href do link:
Exemplo: href="#x"

Nota: os indicadores de qual slide esta sendo exibido devem ficar logo após a declaração da classe carousel e ficam dentro de uma ol.

carousel-indicators	é a classe que agrupa os indicadores do carousel
data-target="#idCarousel" data-slide-to="0"	
data-target	dentro da li vincula o item ao controle carousel
data-slide-to	indica para qual elemento o controle deve navegar quando clicamos no controle. Vai de zero a n-1.
active	indica que este elemento é o elemento inicial do carousel

Nota: Ao clicar no controle o indicador é também atualizado.

carousel-caption	informa que este elemento é a legenda do carousel.
------------------	--