

# Propriedades

display                      position  
[flexbox]                    visibility  
[CSS Grid]                  Texto  
float

## Formatação de texto

### ■ *font*

Propriedade	Exemplo	Observações
<b>font-family</b>	body{font-family:Arial, sans-serif;}	Podem ser definidas famílias genéricas: serif/sans serif; monospace/proportional; cursive (escrita manual); fantasy Não há limite para o número de fontes definidas.
	body{font-family:Tahoma, Geneva, sans-serif; }	Deve-se começar por fontes específicas e terminar com uma genérica, de forma a garantir a correta interpretação pelo browser.
<b>font-size</b>	h1{font-size: 1.5em;}	Unidades relativas
	h1{font-size: 150%;}	Unidades absolutas
	h1{font-size: x-large;}	keywords: xx-small, small, medium*, large, x-large, ...

## Formatação de texto

### ■ *font*

Propriedade	Exemplo	Observações
<b>font-weight</b>	h1 {font-weight: bold;}	Valores : normal*, bold, bolder, lighter, 100...900, inherit (herda o valor do elemento pai) Forma correcta de colocar elementos a bold, deve substituir o elemento HTML <b>; <strong>
<b>font-style</b>	h1 {font-style: italic;}	Valores : normal*, italic, oblique, inherit
<b>font-variant</b>	h1 {font-variant: small-caps;}	Valores: normal*, small-caps, inherit
<b>font</b>	h1 {font: italic bold inherit 1.5em Arial, sans-serif;}	Propriedade abreviada, onde a ordem dos valores é importante {font: style weight variant size font-family}

\* - default value

## Formatação de texto

### ■ *Download* de fontes

#### ■ @font-face rule

- Evita a necessidade da fonte utilizada ter de estar previamente instalada no cliente.
- A fonte é instalada no web server e sempre que necessário é efetuado o seu download.

```
@font-face {  
    font-family: newFont;  
    src: url('myfont.ttf')  
  
div{font-family:newFont;}
```

## Formatação de texto

Propriedade	Exemplo	Observações
<b>line-height</b>	p {line-height: 2;} p {line-height: 2em;}	Define o espaçamento entre linhas. Valores: number (factor de escala), length measurement, percentage, normal*, inherit
<b>text-indent</b>	p#1 {text-indent: 2em;} p#1 {text-indent: 20%;}	Define a indentação de um parágrafo. Valores : length measurement (0*), percentage, inherit
<b>text-align</b>	p#1 {text-align: left;}	Alinhamento horizontal do texto. Valores: left*, right, center, justify, inherit
<b>text-decoration</b>	a {text-decoration: none;}	Permite criar sublinhados, linhas sobrepostas ao texto,..
<b>text-transform</b>	h1 {text-transform: none;}	Define o texto em maiúsculas, minúsculas, ... Valores: none*, capitalize (apenas a 1ª letra de cada palavra), lowercase, uppercase, inherit
<b>letter-spacing</b>	p {letter-spacing: 8px;}	Define o espaçamento entre letras. Valores: length measurement, normal*, inherit
<b>word-spacing</b>	p {word-spacing: 8px;}	Define o espaçamento entre palavras. Valores: length measurement, normal*, inherit

## Formatação de texto (layout)

### ■ Múltiplas Colunas

```
p{
  background-color: orange;
  color:white;
  width: 600px;
  column-count: 3;
  font-size: 1.3em;
}
</style>
</head>
<body>
  <p>
    Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do ....
```


Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud	exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu	fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum
--	--	--

# Propriedades

display	position
[flexbox]	visibility
[CSS Grid]	Texto
float	<u>Cor</u>

## Cor

### ■ Propriedades: Formatação (cor)

Propriedade	Exemplo	Observações
color	<code>h1 {color: red;}</code>	Valores: color value (RGB / hexadecimal); color name: 
	<code>h1 {color: #FF0000;}</code>	
	<code>h1 {color: #F00;}</code>	
	<code>h1 {color: rgb(255,0,0);}</code>	
background-color	<code>h1 {background-color: #F00;}</code>	Valores: color value (RGB / hexadecimal); color name

## ■ Propriedades: Formatação (cor)

### ■ CSS color names (140)

- `p {color: orange;}`

### ■ RGB (red, green, blue)

- Decimal [0-255]
  - `p {color: rgb(255,153,0);}`

### ■ Hexadecimal # RRGGBB [0-9;A-F]

- `p {color: #FF9900}`
- `p {color: #F90}` (notação condensada quando a notação hexadecimal é composta por três pares de valores duplicados [utilizado por defeito em alguns editores de HTML/CSS ]

maroon #800000	red #ff0000	orange #ffa500	yellow #ffff00	olive #808000
purple #800080	fuchsia #ff00ff	white #ffffff	lime #00ff00	green #008000
navy #000080	blue #0000ff	aqua #00ffff	teal #008080	
black #000000	silver #c0c0c0	gray #808080		

*Conversão decimal – hexadecimal:*

*dividir o número original por 16, o 1º dígito é o quociente e o 2º dígito o resto da divisão  
 $200 = C8 = (16 \times 12 + 8)$*

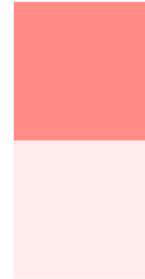
## ■ Existem vários *color palette generators* disponíveis:

- <https://www.materialpalette.com/light-green/purple>
- <https://digitalsynopsis.com/design/website-color-schemes-palettes-combinations/>
- <http://htmlcolorcodes.com/resources/best-color-palette-generators/>



- *alpha channel*
  - grau de transparência na definição da cor
    - Exemplo:
      - `h1 {color: #FF0000CC;}`
      - `rgba(255,0,0,0.8)`

```
.alpha{
  width:100px;height:100px;
  background-color: #ff6f69CC;
}
.alpha2{
  width:100px;height:100px;
  background-color: #ff6f6922;
}
</style>
</head>
<body>
  <div class="alpha"></div>
  <div class="alpha2"></div>
```



## Propriedades

display	position	<u>opacity</u>
[flexbox]	visibility	
[CSS Grid]	Texto	
float	Cor	

- A propriedade *opacity* é frequentemente utilizada
  - permite controlar o nível de transparência de um elemento

```
h1{color:darkorange}
#o1{opacity:1;}
#o2{opacity:0.5;}
#o3{opacity:0.25;}
</style>
</head>
<body>
  <h1 id="o1">Opacity</h1>
  <h1 id="o2">Opacity</h1>
  <h1 id="o3">Opacity</h1>
```

Opacity

Opacity

Opacity

## Propriedades

display

[flexbox]

[CSS Grid]

float

position

visibility

Texto

Cor

opacity

Imagens de Fundo

Propriedade	Observações
<b>background-image</b>	Imagem de fundo
<b>background-repeat</b>	Controla a repetição da imagem
<b>background-position</b>	Posição da imagem de fundo
<b>background-attachment</b>	Acompanha o deslocamento vertical dos conteúdos

## Imagens de fundo

- **{background-size: width height;}**
  - É possível definir várias dimensões para a imagem de fundo, eliminando a dependência das dimensões originais da imagem, admite vários valores:
    - **auto**: default (dimensões originais da imagem)
    - **length / percentage** (relativamente ao *container*)
    - **cover**: escala a imagem de modo a cobrir toda a área visível, algumas partes da imagem podem ficar invisíveis (as proporções da imagem original são mantidas)
    - **contain**: escala a imagem considerando a área visível (toda a imagem é visível). Caso seja necessário a imagem é repetida.

```
<style>
  body {color:white;
        background-image:url(back_Image.jpg);
        background-repeat:no-repeat;
        background-size:600px 100px;}
</style>
</head>
<body>
  <h1>Imagem de Fundo</h1>
```



# Propriedades

display	position	opacity
[flexbox]	visibility	Imagens de Fundo
[CSS Grid]	Texto	<u>Listas</u>
float	Cor	

## listas

Propriedade	Exemplo	Observações
<b>list-style-type</b>	<code>ul {list-style-type:square;}</code>	Permite escolher o símbolo de uma lista. Valores: none, disc*, circle, square, ...
<b>list-style-position</b>	<code>li {background-color: #666;}</code> <code>ul {list-style-position: outside;}</code>	Define se o símbolo da lista se encontra no interior ou exterior do background aplicado aos elementos da lista Valores : inside, outside, inherit
<b>list-style-image</b>	<code>ul {list-style-image: url(/image.png)</code> <code>list-style-position: outside;}</code>	Criar novos símbolos de lista (bullets) Valores: url, none*, inherit

\* - default value

# Propriedades

display	position	opacity
[flexbox]	visibility	Imagens de Fundo
[CSS Grid]	Texto	Listas
float	Cor	<u>Tabelas</u>

## Tabelas

### ■ Propriedades específicas para tabelas

Propriedade	Observações
<i>border-collapse</i>	Permite definir a separação entre células (borders separados, ou apenas um único border a separar células adjacentes). Valores: separate, collapse, inherit
<i>border-spacing</i>	Permite definir o espaçamento entre células. Valores: length, inherit
<i>empty-cells</i>	Permite definir se as células vazias são visíveis Valores: show, hide, inherit

```
<style>
  table{
    border-style:solid;
    border-width:1px;
    border-color:#900;
    border-collapse: separate;
    border-spacing:10px;
    empty-cells:hide;}

  th, td {border-style:solid;
    border-width:1px;
    border-color:#900;
    width:100px;
    text-align:center;}
</style>
```

```
<table>
  <tr><th>(1,1)</th>
    <th>(1,2)</th>
  </tr>
  <tr><td>(2,1)</td>
    <td>(2,2)</td>
  </tr>
  <tr><td>(3,1)</td>
    <td></td>
  </tr>
</table>
```

(1,1)	(1,2)
(2,1)	(2,2)
(3,1)	

## Propriedades

display

[flexbox]

[CSS Grid]

float

position

visibility

Texto

Cor

opacity

Imagem Fundo

Listas

Tabelas

Formulários

- Uma formatação correta dos elementos de um formulários é determinante para assegurar a sua usabilidade (objetivo principal)
  - Existem determinadas propriedades que só se aplicam a campos específicos

Elementos	Propriedades
text inputs (text, password, email, search, tel, url)	<i>width, height, background-color, background-image, border, border-radius, margin, padding, box-shadow, color, ...</i>
textarea	<i>line-height, resize</i>
inputs (submit, reset, button)	<i>width, height, border, background-color, margin, padding, box-shadow, ...</i>
select	<i>width, height, color, background-color ...</i>
fieldsets / legends	<i>border, background-color, margin, padding, ...</i>

## Propriedades

display	position	opacity	Formulários
[flexbox]	visibility	Imagem Fundo	<u>Transformações 2D/3D</u>
[CSS Grid]	Texto	Listas	
float	Cor	Tabelas	

## Transformações 2D

- Propriedade com múltiplos valores: [http://www.w3schools.com/cssref/css3\\_pr\\_transform.asp](http://www.w3schools.com/cssref/css3_pr_transform.asp)

- `seletor{ transform: rotate (deg)}`

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8" />
  <title></title>
  <style>
    div {width:200px;
          height:100px;
          background-color:darkorange;
          color:white;
          position:absolute;left:50px;top:100px;
          text-align:center;
          transform:rotate(30deg);
    }
  </style>
</head>
<body>
  <div>transform:ROTATE()</div>
</body>
</html>
```



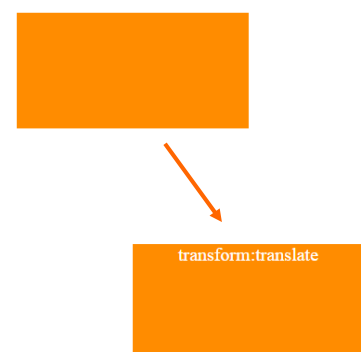
Por defeito, a imagem roda sobre o seu centro.  
A propriedade *transform-origin* permite alterar o centro de rotação.

## Transformações 2D

- `seletor{ transform: translate(xx,yy)}`

- A referência para o movimento de translação é a posição original do objecto

```
<!DOCTYPE html>
<html>
<head>
  <style>
    div {width:200px;
          height:100px;
          background-color:darkorange;
          color:white;
          position:absolute;left:50px;top:100px;
          text-align:center;}
    #d2 {transform:translate(100px,200px);}
  </style>
</head>
<body>
  <div></div>
  <div id="d2">transform:translate</div>
</body>
</html>
```



## Transformações 2D

■ `seletor{ transform: scale(xx,yy)}`

```
<!DOCTYPE html>
<html>
<head>
  <style>
    div {width:60px;
        height:20px;
        background-color:darkorange;
        color:white;
        position:absolute;left:50px;top:100px;
        text-align:center; }
    #d2 {left:150px;
        transform:scale(2,3);}
  </style>
</head>
<body>
  <div></div>
  <div id="d2">scale</div>
</body>
</html>
```



## Transformações 3D

■ `seletor{transform: rotate X}; seletor{transform: rotateY}`

```
...
    .dft{ background-color: darkorange;
        color:white;
        font-size: 2em;
        width:50px;
        height:100px;
        margin:10px;
        text-align: center}

    .r3d{ transform: rotateX(180deg);}
  </style>
</head>

<body>

  <div class="dft">1</div>
  <div class=" dft r3d">1</div>
  ...
```



# Propriedades

display	position	opacity	Formulários
[flexbox]	visibility	Imagem Fundo	Transformações 2D/3D
[CSS Grid]	Texto	Listas	<u>Transições</u>
float	Cor	Tabelas	

## Transições

- Suavização de alterações entre estados diferentes
  - Definir uma transição envolve:
    - Propriedade a alterar (obrigatório!)
    - Duração (obrigatório!)
    - A forma como se processa a aceleração da transição
    - A eventual existência de uma pausa antes de iniciar a transição.

# Transições

Propriedade	Exemplo	Observações
<b>transition-property</b>	{transition-property: width;}	Algumas propriedades que podem ser animadas: background-color; background-position; height, width, font-size; font-weight;...
<b>transition-duration</b>	{transition-duration: 1s;}	Os valores podem ser definidos em segundos (s) ou milisegundos (ms)
<b>transition-timing-function</b>	{transition-timing-function: ease;}	ease (início lento, rápido, final lento); linear (a velocidade mantém-se do início ao fim); ease-in (início lento, final rápido); ease-out (início rápido, final lento); ease-in-out (início lento, rápido, final lento).
<b>transition-delay</b>	{transition-delay: 0.3s;}	Os valores podem ser definidos em segundos (s) ou milisegundos (ms)
<b>transition</b>	{transition: width 1s ease-in-out 0.3s;}	{transition: property duration timing-function delay}

# Transições

```
<!DOCTYPE html>
<html>
<head>
  <style>
    div {width:100px;
        height:100px;
        background-color:orange;
        transition:width 3s; }
    div:hover {width:300px;}
  </style>
</head>
<body>
  <div></div>
</body>
</html>
```

## Transição width Elemento div



Uma vez o rato sobreposto ao elemento div, efetua-se a transição da **width** inicial para a **width** final



duração: 3s

## Transição width Elemento div



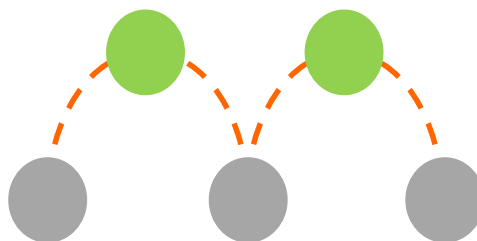


# Propriedades

display	position	opacity	Formulários
[flexbox]	visibility	Imagem Fundo	Transformações 2D/3D
[CSS Grid]	Texto	Listas	Transições
float	Cor	Tabelas	<u>Animações</u>

## Animação

- Baseada em *key frames* (*keyframe animation*)
  - Transições são animações baseadas em 2 *key frames* (inicial, final)
  - Animações mais complexas requerem a especificação de um maior número de *key frames*



*Keyframes*

`@keyframes nomeAnimação { ... }`

*Parâmetros*

## ▪ Definição de *key frames*

nome da animação (obrigatório)

```
@keyframes quadrados {  
  0% {background-color:red; left:0px; top:0px}  
  25% {background-color:yellow; left:200px; top:0px}  
  50% {background-color:blue; left:200px; top:200px}  
  75% {background-color:green; left:0px; top:200px}  
  100% {background-color:red; left:0px; top:0px}  
}
```

Definição das *key frames* (obrigatório)  
(neste exemplo foram definidas 5 *key frames*)

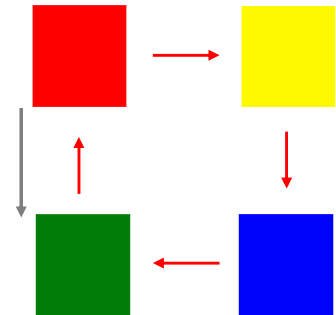
- Definir os parâmetros da animação
  - *animation-name* (**obrigatório!**)
  - *animation-duration*
    - Tempo, definido em segundos, que uma animação demora
      - o valor por defeito é o 0 (sem animação)
      - os valores negativos são tratados como 0
  - *animation-timing-function*
    - *ease* : Valor por defeito. A animação começa lenta, depois rápido, antes de terminar lento
    - *linear* : Velocidade constante ao longo da animação
    - *ease-in* : Animação inicia lenta
    - *ease-out* : Animação termina lenta
    - *ease-in-out* : Animação com um começo e um final lentos
    - *cubic-bezier(x1, y1, x2, y2)* : Controlo mais preciso da velocidade da animação

- *animation-delay*
  - define o tempo de intervalo (definido em s) entre a ocorrência do evento e o início da animação
- *animation-iteration-count*
  - define o número de vezes que a animação é executada
    - Infinite: animação é executada de forma ininterrupta
- *animation-direction*
  - normal
    - valor por defeito. A animação é sempre reproduzida na mesma direção
  - alternate
    - executa as iterações ímpares na direção normal e as iterações pares na direção inversa
- ...

## ■ Animação

- Parametrização da animação e associação com *key frames* definidas

```
<!DOCTYPE html>
<html>
<head>
  <style>
    div {width:100px;
        height:100px;
        background-color:red;
        position:relative;
        animation:quadrados 5s infinite alternate;}
    @keyframes quadrados {
      0% {background-color:red; left:0px; top:0px}
      25% {background-color:yellow; left:200px; top:0px}
      50% {background-color:blue; left:200px; top:200px}
      75% {background-color:green; left:0px; top:200px}
      100% {background-color:red; left:0px; top:0px}
    }
  </style>
</head>
<body>
  <h2>Animação Quadrados</h2>
  <div></div>
</body>
</html>
```

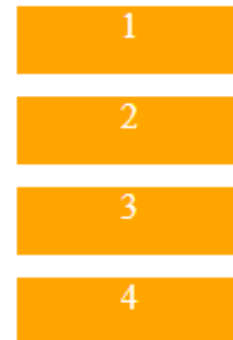


## Exercício *Flexbox*

## Exercício Flexbox

```
<ul>
  <li>1</li>
  <li>2</li>
  <li>3</li>
  <li>4</li>
</ul>
```

```
li{  background-color: orange;
      color:white;
      width:100px;
      height:30px;
      margin-top: 10px;
      text-align: center;
      list-style: none;
}
```



## Exercício Flexbox

```
<ul>
  <li>1</li>
  <li>2</li>
  <li>3</li>
  <li>4</li>
</ul>
```

```
ul{ display:flex;}

li{  background-color: orange;
      color:white;
      width:100px;
      height:30px;
      margin-top: 10px;
      text-align: center;
      list-style: none;
}
```



## Exercício Flexbox

```
<ul>
  <li>1</li>
  <li>2</li>
  <li>3</li>
  <li>4</li>
</ul>
```

```
li{background-color: orange;
  color:white;
  width:100px;
  height:30px;
  margin-top: 10px;
  text-align: center;
  list-style: none;

  margin:0 1px;
}

ul{ display:flex;}
```

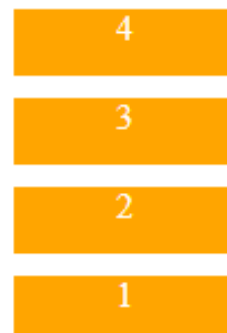


## Exercício Flexbox

```
<ul>
  <li>1</li>
  <li>2</li>
  <li>3</li>
  <li>4</li>
</ul>
```

```
li{background-color: orange;
  color:white;
  width:100px;
  height:30px;
  margin-top: 10px;
  text-align: center;
  list-style: none;
}

ul{ display:flex;
  flex-direction: column-reverse;}
```



## Exercício Flexbox

```
<ul>
  <li>1</li>
  <li>2</li>
  <li>3</li>
  <li>4</li>
</ul>
```

```
li{background-color: orange;
color:white;
width:100px;
height:30px;
text-align: center;
list-style: none;

margin: 2px;
}

ul{ display:flex;
  flex-direction: row;
}
```



## Exercício Flexbox

```
<ul>
  <li>1</li>
  <li>2</li>
  <li>3</li>
  <li>4</li>
</ul>
```

```
li{background-color: orange;
color:white;
width:100px;
height:30px;
text-align: center;
list-style: none;
margin: 2px;}

ul{ display:flex;
  flex-direction: row;
border:1px solid gray;
width:250px;
flex-wrap: nowrap;
justify-content: flex-start;
padding:0px;}
```



## Exercício Flexbox

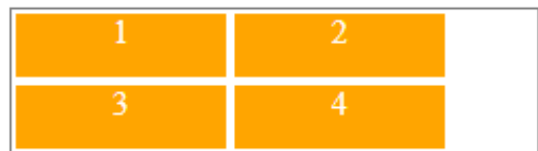
```
<ul>
  <li>1</li>
  <li>2</li>
  <li>3</li>
  <li>4</li>
</ul>
```

```
li{background-color: orange;
  color:white;
  width:100px;
  height:30px;
  text-align: center;
  list-style: none;

  margin: 2px;}

ul{ display:flex;
  flex-direction: row;
  border:1px solid gray;
  width:250px;
  justify-content:flex-start;
  padding:0px;

  flex-wrap: wrap;}
```



## Exercício Flexbox

```
<ul>
  <li>1</li>
  <li>2</li>
  <li>3</li>
  <li>4</li>
</ul>
```

```
li{background-color: orange;
  color:white;
  width:100px;
  height:30px;
  text-align: center;
  list-style: none;
  margin: 2px;

  flex-basis:20px;
}

ul{ display:flex;
  flex-direction: row;
  border:1px solid gray;
  width:250px;
  justify-content:flex-start;
  padding:0px;

  flex-wrap: wrap;}
```





```
li{background-color: orange;  
color:white;  
width:100px;  
height:30px;  
text-align: center;  
list-style: none;  
margin: 2px;  
flex-basis:20px;  
flex-grow: 1;}
```

```
<ul>  
  <li>1</li>  
  <li>2</li>  
  <li>3</li>  
  <li>4</li>  
</ul>
```



```
ul{ display:flex;  
flex-direction: row;  
border:1px solid gray;  
width:250px;  
justify-content:flex-start;  
padding:0px;  
flex-wrap: wrap;}
```

## Web Responsive

## Web Responsive

- Permite que uma aplicação seja utilizada de forma adequada independentemente do dispositivo que está a ser utilizado para aceder aos conteúdos
  - Todos os dispositivos acedem ao mesmo conteúdo usando o mesmo URL
  - São aplicados estilos diferentes, de acordo com o dispositivo, de forma a redimensionar componentes e otimizar a usabilidade (interação com o utilizador)

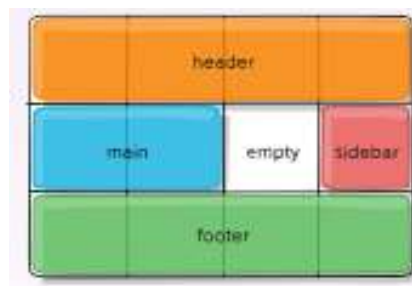


<https://sitefactory.com.br/blog/marketing-digital/o-que-e-um-site-responsivo-e-para-que-ele-serve/>

## Web Responsive

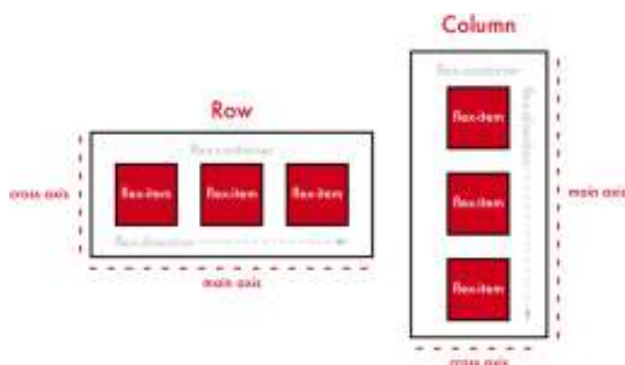
- Três componentes principais:
  - *layout flexível (flexible grid)*
    - As dimensões não podem ser estáticas, têm de ajustar (reduzir/aumentar) de acordo com o espaço disponível.
  - *Imagens flexíveis*
    - Imagens com capacidade de sofrer um efeito de escala (redução da dimensão) de acordo com a alteração do layout.
  - *CSS media queries*
    - método que permite a aplicação de estilos de acordo com o dispositivo em que a aplicação vai ser visualizada.

### Layout flexível



CSS Grid

### Flexbox Container



### Fluid Layout



### Imagens/media Flexíveis

- A imagem acompanha a redução do layout sendo que na situação contrária a sua dimensão original é o limite para a sua visualização.

```
img{max-width: 100%;}
```

- A propriedade *max-width* estabelece a largura máxima de um elemento
- Se for definida em %, a largura do elemento é **diretamente indexada às dimensões do respetivo container** até à dimensão original da imagem ( $\neq$  width:100%).
  - O elemento ajusta-se automaticamente à variação da largura do respetivo *container*
- Também aplicável a outro tipo de média

```
img, object, video {max-width:100%;}
```

### ■ CSS media queries

- Permite a aplicação de estilos de acordo com meio usado (*media type*) para visualização dos conteúdos web assim como em função das suas características (*media features*)

### ■ CSS media queries

#### ■ media types

- screen, print, all, ... <https://www.w3.org/TR/CSS21/media.html%23media-types> *Opcional!*

#### ■ media features

- width; height; device-width; ...

### ■ CSS media queries

#### ■ media features

Media feature	Observações
<i>width</i>	largura do viewport
<i>height</i>	altura do viewport
<i>orientation</i>	portrait/landscape
<i>aspect-ratio</i>	razão entre a largura e a altura do viewport
...	...

- Algumas propriedades podem ser testadas para valores limite, utilizando para tal prefixos:

#### ■ min, max

- min-width; max-width
- min-height; max-height

### ■ CSS media queries

- Efetuados diretamente na folha de estilos (a forma mais comum):

```
@media screen and (min-width:480px){  
    /* CSS Rules */  
}  
  
@media screen and (min-width:480px) and (orientation:landscape){  
    /* CSS Rules */  
}
```

- Em alternativa ser incorporados no html, incorporando os *media queries* na tag **<link>** através do atributo **media**

```
<head>  
  <link href="geral.css" rel="stylesheet">  
  <link href="colunas.css" rel="stylesheet" media="screen and (min-width:480px)">  
</head>
```

- Quando se utilizam *@media rules* a **ordem** das regras é muito importante (para seletores iguais prevalecem as ultimas regras a ser definidas):

- Estratégia:

- Especificam-se as regras aplicadas por defeito aos elementos (*baseline rules*)
  - Algumas dessas regras vão ser posteriormente substituídas por outras regras inseridas nas *@media* de forma a otimizar o conteúdo para determinados viewports.
  - A definição de *media queries* é baseada do conceito de *mobile-first*, ou seja:
    - Começa-se por definir os layouts para os dispositivos mais pequenos;
    - À medida que o espaço de visualização aumenta, são aplicados novos estilos para novas possibilidades de visualização.

```
@media screen and (min-width:480px){  
    /* CSS Rules */ }  
  
@media screen and (min-width:768px){  
    /* CSS Rules */ }
```

### ■ Definição das @media queries

- Escolha dos **breakpoints** (largura estabelecida no *media query* para definir novos estilos)

- <http://responsivedesign.is/develop/browser-feature-support/media-queries-for-common-device-breakpoints>
- <http://css-tricks.com/snippets/css/media-queries-for-standard-devices/>

```
/* ----- iPhone X ----- */

/* Portrait and Landscape */
@media only screen
  and (min-device-width: 375px)
  and (max-device-width: 812px)
  and (-webkit-min-device-pixel-ratio: 3) {

}

/* Portrait */
@media only screen
  and (min-device-width: 375px)
  and (max-device-width: 812px)
  and (-webkit-min-device-pixel-ratio: 3)
  and (orientation: portrait) {
```

### ■ CSS Pixel

- Corresponde ao pixel definido (abstração) nas declarações CSS

- width:200px; padding:10px;...

- <https://www.w3.org/TR/CSS2/syndata.html#length-units>

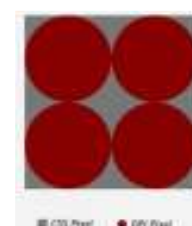
### ■ pixel (device)

- número de pixels existentes no dispositivo (hardware pixels - resolução)

### ■ Device Pixel Ratio (DPR)








- A relação que existe entre os pixels (device) e os pixels CSS.

- Exemplo: 1 CSS pixel corresponde a 4 device pixels (DPR = 2)



<http://blog.popupdesign.com.br/desenvolvimento-responsivo-e-viewport/>

### ■ Device pixel ratio (DPR)

		pixel (CSS) device width / device height	resolution pixels device		
	Apple iPhone 14	390 x 844	1170 x 2532	6.1"	3.0
	Apple iPhone 14 Plus	428 x 926	1284 x 2778	6.7"	3.0
	Apple iPhone 14 Pro	393 x 852	1179 x 2556	6.1"	3.0
	Apple iPhone 14 Pro Max	430 x 932	1290 x 2796	6.7"	3.0
	Apple iPhone 15	393 x 852	1179 x 2556	6.1"	3.0
	Apple iPhone 15 Plus	430 x 932	1290 x 2796	6.7"	3.0
	Apple iPhone 15 Pro	393 x 852	1179 x 2556	6.1"	3.0

<https://yesviz.com/viewport/>

### ■ Variação da Janela de Visualização (viewport)

#### ■ `<meta name="viewport" ... />`

- A especificação do *viewport* indica ao *browser* que deve ser aplicado um factor de escala à página para esta se ajustar às dimensões do dispositivo (screen)
- Deve ser incluído em todos os documentos html (responsive)

Property	Description
width	The width of the virtual viewport of the device.
device-width	The physical width of the device's screen.
height	The height of the "virtual viewport" of the device.
device-height	The physical height of the device's screen.
initial-scale	The initial zoom when visiting the page. 1.0 does not zoom.
minimum-scale	The minimum amount the visitor can zoom on the page. 1.0 does not zoom.
maximum-scale	The maximum amount the visitor can zoom on the page. 1.0 does not zoom.
user-scalable	Allows the device to zoom in and out. Values are yes or no.

- O *viewport* necessita de ser declarado apenas uma vez

*“A <meta> viewport element gives the browser instructions on how to control the page's dimensions and scaling”*

```
<meta name="viewport" content="width=device-width, initial-scale=1">
```

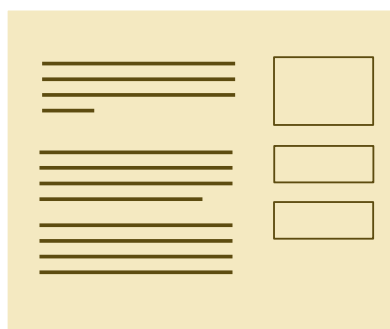
- **content = “width=device-width, ....”**
  - Largura do *viewport* igual à largura do dispositivo
- **content = “width=device-width, initial-scale=1”**
  - Garante o nível de zoom inicial quando é feito o download da página
  - Evita que seja visualizado apenas uma parte do conteúdo inicial

## Web Responsive

Exemplo mobile safari  
renderizado: 980px



```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```



screen: 320px



renderizado: 320px

Todo o espaço disponível  
é aproveitado.

O viewport é definido  
como sendo igual à  
largura do dispositivo

Permite uma correta  
aplicação das CSS media  
queries

screen: 320px



## Web Responsive

- Sem definição da meta *tag*, considerado por defeito um *viewport* de 980px largura
  - exemplo: iPhone 12 Pro



A imagem é renderizada a 980px e depois ajustada para a dimensão do dispositivo

Neste caso é desperdiçado cerca de 50% do espaço disponível

- Com ajuste do *viewport*

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```



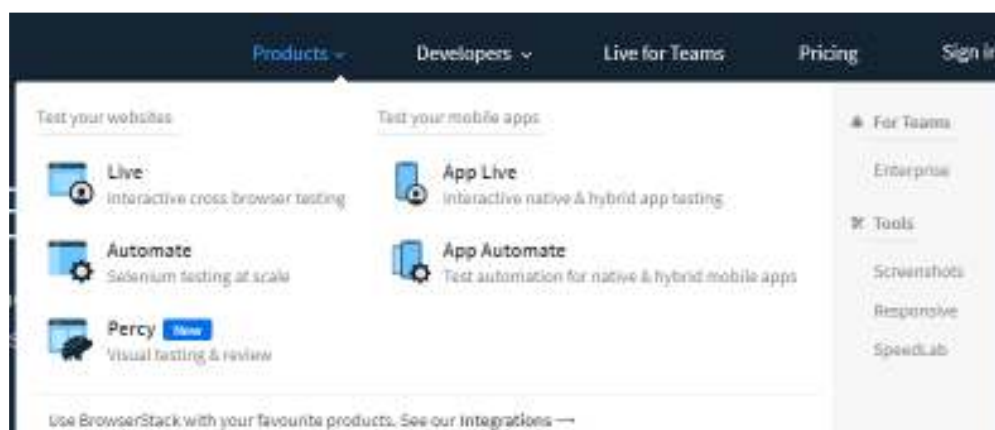
A imagem é renderizada considerando a largura do dispositivo

Optimização do espaço disponível

## Web Responsive

- Testing
  - Emuladores

- <https://www.browserstack.com/emulators-simulators>



# CSS Reset

## CSS Reset

### ■ CSS reset

- Aplicação de um conjunto de regras CSS que inviabilize a aplicação de estilos por defeito (browsers) de forma a tornar o ponto de partida tão neutro quanto possível
  - Melhora a portabilidade entre browsers
  - Existem disponíveis vários tipos de CSS reset
  - Se equacionada a sua utilização, um CSS reset deve ser copiado para o início da CSS de forma a garantir que é efetuado de forma correta.

```
html, body, div, span, applet, object, iframe,
h1, h2, h3, h4, h5, h6, p, blockquote, pre,
a, abbr, acronym, address, big, cite, code,
del, dfn, em, img, ins, kbd, q, s, samp,
small, strike, strong, sub, sup, tt, var,
b, u, i, center,
dl, dt, dd, ol, ul, li,
fieldset, form, label, legend,
table, caption, tbody, tfoot, thead, tr, th, td,
article, aside, canvas, details, embed,
figure, figcaption, footer, header, hgroup,
menu, nav, output, ruby, section, summary,
time, mark, audio, video {
    margin: 0;
    padding: 0;
    border: 0;
    font-size: 100%;
    font: inherit;
    vertical-align: baseline;
}

/* HTML5 display-role reset for older browsers */
article, aside, details, figcaption, figure,
footer, header, hgroup, menu, nav, section {
    display: block;
}
body {
    line-height: 1;
}
ol, ul {
    list-style: none;
}
blockquote, q {
    quotes: none;
}
blockquote:before, blockquote:after,
q:before, q:after {
    content: '';
    content: none;
}
table {
    border-collapse: collapse;
    border-spacing: 0;
}
```

<http://cssreset.com/>



Normalize.css

A modern, HTML5-ready alternative  
to CSS resets

*“Normalize.css makes browsers render all elements more consistently and in line with modern standards. **It precisely targets only the styles that need normalizing.**”*

```
/*! normalize.css v8.0.1 | MIT License | github.com/necolas/normalize.css */

/* Document
   ========================================================================== */

/**
 * 1. Correct the line height in all browsers.
 * 2. Prevent adjustments of font size after orientation changes in iOS.
 */

html {
  line-height: 1.15; /* 1 */
  -webkit-text-size-adjust: 100%; /* 2 */
}

/* Sections
   ========================================================================== */

/**
 * Remove the margin in all browsers.
 */

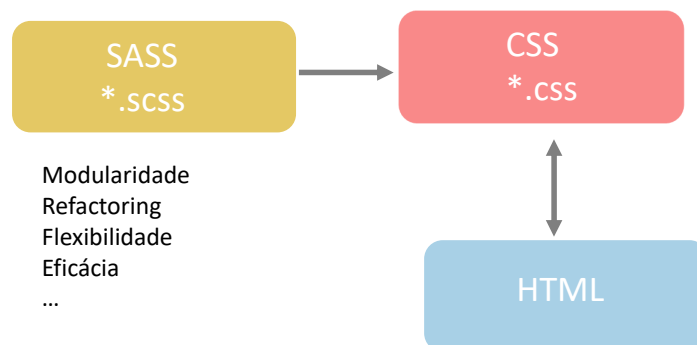
body {
  margin: 0;
}
```

<https://necolas.github.io/normalize.css/8.0.1/normalize.css>

## SASS

*Syntactically Awesome Style Sheets*

- Pré-processador código CSS
  - facilita a manutenção de código CSS, melhora a organização do código, mais escalável, ...
    - Variáveis,
    - Reutilização, ....
  - ficheiros de extensão \*.scss a partir dos quais são criados os ficheiros \*.css a ligar ao HTML (HTML **não reconhece** \*.scss)



## Sass: Características Principais

- Variáveis
  - permite a declaração de variáveis, as quais são precedidas de \$
    - Tipos de dados:
      - numbers
      - strings
      - colors
      - null
      - lists

```
$margin-values: 1px 2px 3px 4px;
```
      - maps

```
$map: (  
  key: value,  
  nextkey: nextvalue  
);
```
  - As variáveis devem:
    - possuir um significado semântico
    - seguir um padrão/convenção (ex: prefixos "header-" "footer-" "section-", ou sufixos "-color")

## Sass: Características Principais

### ■ Variáveis

- Importantes quando o mesmo valor é utilizado em vários sítios
  - Determinantes para a estruturação e manutenção do código, principalmente em projetos de grande dimensão



## Sass: Características Principais

### ■ Variáveis

- na realidade as CSS também permitem a declaração de variáveis (-- variável: valor) mas neste caso é necessário recorrer à função CSS (var)

```
<style>
  :root {--colorFg: orange}

  h1{color:var(--colorFg)}
  p{color:var(--colorFg)}
</style>
```

```
<body>
  <h1>Exemplo variáveis: h1</h1>
  <p>Exemplo variáveis: p</p>
```

**Exemplo variáveis: h1**

Exemplo variáveis: p

## Sass: Características Principais

### ■ Scope das variáveis

- Uma variável declarada num seletor é visível apenas nesse seletor.

```
variaveis.scss

$primaryColor: darkorange;

body {
  $primaryColor: lightgray;
  background-color: $primaryColor;
}

p {
  width: 200px;
  color: white;
  background-color: $primaryColor;
}
```

variaveis.scss

```
<body>
  <p>VARIABLES SCOPE</p>
</body>
```

```
variaveis.css

body {
  background-color: lightgray; }

p {
  width: 200px;
  color: white;
  background-color: darkorange; }
```

variaveis.css

VARIABLES SCOPE

## Sass: Características Principais

### ■ Scope das variáveis

- **!global** permite declarar uma variável global num seletor mas que é visível em todo o código

```
variaveis.scss

$primaryColor: darkorange;

body {
  $primaryColor: lightgray !global;
  background-color: $primaryColor;
}

p {
  width: 200px;
  color: white;
  background-color: $primaryColor;
}
```

variaveis.scss

```
<body>
  <p>VARIABLES SCOPE</p>
</body>
```

```
variaveis.css

body {
  background-color: lightgray; }

p {
  width: 200px;
  color: white;
  background-color: lightgray; }
```

variaveis.css

VARIABLES SCOPE

## Sass: Características Principais

### ■ Nesting

- Permite a criação de uma clara hierarquia visual, declarando diretamente os seletores enquadrados nos seletores que definem o contexto
  - Seletores contextuais em que o 1º nível de contexto é dado pelo **nest** que enquadra os outros seletores
  - A extensão desta dependência de seletores resulta em seletores cada vez mais específicos

sass\_1.scss

```
$primary-color: lightgray;
$width-element: 300px;

nav {
  ul {
    margin: 0;
    padding: 0;
    list-style: none;
  }

  li { display: inline-block; }

  a {
    display: block;
    padding: 6px 12px;
    text-decoration: none;
  }
}
```

sass\_1.css

```
nav ul {
  margin: 0;
  padding: 0;
  list-style: none; }
nav li {
  display: inline-block; }
nav a {
  display: block;
  padding: 6px 12px;
  text-decoration: none; }
```

sass\_1.css

## Sass: Características Principais

### ■ @import / Partials

- Sass permite importar pequenos blocos de código \*.scss (*partials*)
  - *Partial* é um ficheiro \*.scss cujo nome inicia com **underscore ( \_ )**.
    - O *underscore ( \_ )* indica que se trata de um *partial* (a ser importado por outro ficheiro \*.scss e não convertido diretamente num \*.css)
  - Os *Sass partials* são usados com a diretiva @import (só é especificado o nome do *partial*)
  - Torna o código mais modular (reset, variáveis, funções, ...) e consequentemente mais fácil de manter.

\_reset.scss

```
1 html,
2 body,
3 ul,
4 ol {
5     margin: 0;
6     padding: 0;
7 }
```

sass\_1.scss

```
1 @import 'reset';
2
3 body {
4     font: 100% Helvetica, sans-serif;
5     background-color: #efefef;
6 }
```

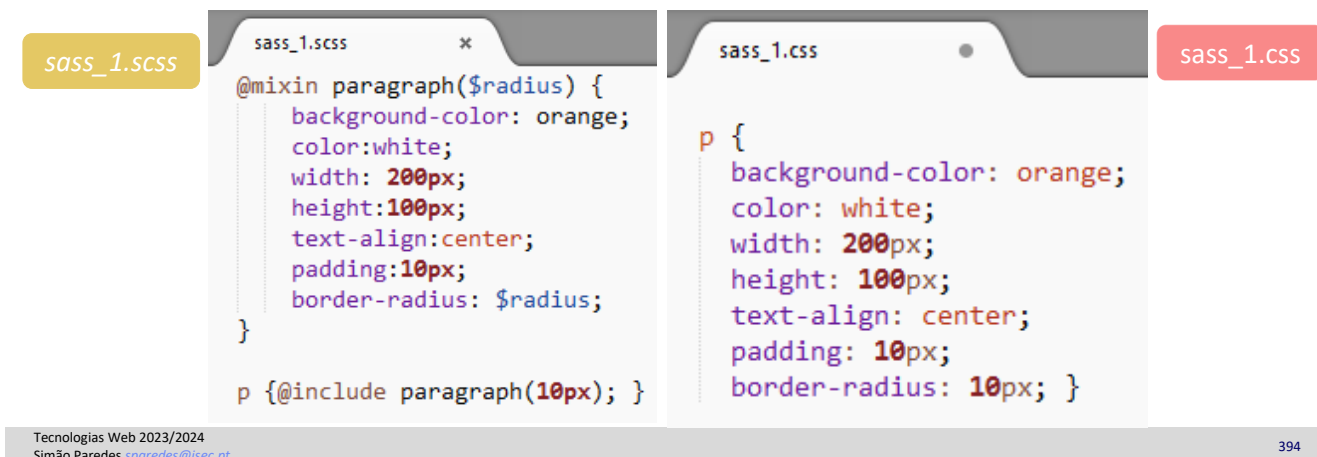
sass\_1.css

```
html,
body,
ul,
ol {
  margin: 0;
  padding: 0; }
body {
  font: 100% Helvetica, sans-serif;
  background-color: #efefef; }
```

## Sass: Características Principais

### ■ @mixin

- Conjunto de declarações CSS que se pretende reutilizar
- A criação do grupo de declarações CSS é baseado na diretiva **@mixin**
  - É obrigatório atribuir um nome ao @mixin
  - É possível a passagem de parâmetros ao @mixin
    - O parâmetro deve ser definido com (\$nome\_parâmetro)
  - @mixin é usado com base na diretiva **@include**
  - Os @mixin podem ser agrupados num *partial* de forma a libertar a \*.scss principal



## Sass: Características Principais

### ■ Inheritance (@extend)

- A diretiva **@extend** permite a partilha de declarações CSS entre diferentes seletores
- Minimiza a repetição de código em diferentes seletores





## Sass: Características Principais

### ■ @extend

- cada @extend pode ser aplicado a apenas um seletor

\*.SCSS

```
.error{
  color:red;
}

.critical-error{
  @extend .error;
  font-weight:bold;
  border:2px solid red;
}
```



```
.minor-error{ background-color:gray;}

.critical-error{
  @extend .error .minor-error;
  font-weight:bold;
  border:2px solid red;}

```



- múltiplos @extend no mesmo seletor

```
.critical-error{
  @extend .error;
  @extend .minor-error;
  font-weight:bold;
  border:2px solid red;}

```

- encadeamento de @extend

```
.main-error{
  @extend .critical-error;
  font-size: 125%;}

```

## Sass: Características Principais

### ■ @extend

- É possível definir um seletor exclusivamente para ser utilizado com @extend
  - Iniciam-se por %
  - Não podem ser compilados diretamente para CSS, só são compilados se forem alvo de um @extend

\*.SCSS

```
%error{color:red;}

.critical-error{
  @extend %error;
  font-weight:bold;
  border:2px solid red;}

```

\*.CSS

```
.critical-error {
  color: red; }

.critical-error {
  font-weight: bold;
  border: 2px solid red; }

```

- O @extend de um seletor inexistente, provoca um erro e inibe a geração do respetivo \*.css

- **!optional** permite contornar esta limitação

```
.critical-error{
  @extend %error;
  @extend %critical !optional;
  font-weight:bold;
  border:2px solid red;}

```

# Sass: Características Principais

## ■ @mixin vs. @extend

@mixin	@extend
reutilização código	reutilização código
permite passagem de parâmetros	não permite parâmetros
origina menos seletores CSS	Origina mais seletores CSS
repetição de declarações	minimiza repetição de declarações
	limitação no funcionamento com @media
tipicamente <b>mais rápida</b> que o @extend	

<https://tech.bellycord.com/blog/sass-mixins-vs-extends-the-data/>

# Sass: Características Principais

## ■ @media

- Permite a definição de CSS media queries
- Possibilita o *nest* das CSS *media queries*, no entanto no código \*.css a definição das *media queries* é sempre efetuada no top level

```
#main {  
  width:$content-width;  
  @media only screen and (max-width: 960px){  
    width:auto;  
    max-width:960px;  
  }  
}
```

\*.scss

\*.css

```
@media only screen and (max-width: 960px) {  
  #main {  
    width: auto;  
    max-width: 960px; } }  
}
```

# Sass: Características Principais

## ■ @media

- Não é possível efetuar o @extend de seletores definidos fora da media query

```
@media screen{  
  
  .error{color:red;}  
  
  .critical-error{  
    @extend .error;  
    font-weight:bold;  
    border:2px solid red;}  
}
```

# Sass: Características Principais

## ■ Operadores

- Conjunto de operadores matemáticos:

- +, -, \*, /, % ...

sass\_1.scss

```
sass_1.scss  
  
.container { width: 100%; }  
  
@mixin global{  
  color:white;  
  text-align:center;  
  height:50px;  
}  
  
article[role="main"] {  
  float: left;  
  width: 600px / 960px * 100%;  
  background-color:lightgray;  
  @include global;  
}  
  
aside[role="complementary"] {  
  float: right;  
  width: 300px / 960px * 100%;  
  background-color:darkorange;  
  @include global;  
}
```

sass\_1.css

```
sass_1.css  
  
@charset "UTF-8";  
.container {  
  width: 100%; }  
  
article[role="main"] {  
  float: left;  
  width: 62.5%;  
  background-color: lightgray;  
  color: white;  
  text-align: center;  
  height: 50px; }  
  
aside[role="complementary"] {  
  float: right;  
  width: 31.25%;  
  background-color: darkorange;  
  color: white;  
  text-align: center;  
  height: 50px; }
```

```
<body>  
  <article role="main">Main</article>  
  <aside role="complementary">Aside</aside>  
</body>
```



# Sass: Características Principais

## Operadores

```
sass_1.scss
$container-width: 100%;

.container {
  width: $container-width;
}

.col-4 {
  width: $container-width / 4;
  height: 60px;
  background-color: darkorange;
  color: white;
}
```

```
sass_1.css
@charset "UTF-8";
.container {
  width: 100%;
}

.col-4 {
  width: 25%;
  height: 60px;
  background-color: darkorange;
  color: white;
}
```

operador	obs.
+	adição
-	subtração
/	divisão
*	multiplicação
%	resto divisão inteira
==	igual
!=	diferente

<https://scotch.io/tutorials/getting-started-with-sass>

```
<link href="sass_1.css" rel="stylesheet" type="text/css" />
</head>
<body>
  <div class="col-4">Mathematical Operators</div>
</body>
```

Mathematical Operators

# Sass: Características Principais

## Funções

■ *Dois tipos de Funções: Built-in functions + Custom Functions*

■ *Built-in Functions:*

■ Conjunto de funções disponibilizadas pelo SASS <http://sass-lang.com/documentation/Sass/Script/Functions.html>

■ Algumas das funções mais comuns:

■ *darken(\$color, amount)*

- permite escurecer uma cor, tem a grande vantagem de já se conhecer a cor a escurecer

```
a {
  color: $link-color;
  &:hover {
    color: darken($link-color, 15%);
  }
}
```

■ *lighten(\$color, amount)*

- O efeito contrário, torna uma cor mais clara

■ *transparentize(\$color; amount)*

■ *opacity(\$color; amount)*

## Sass: Características Principais

### ■ Custom Funtions

#### ■ @function

- Permite múltiplos parâmetros (possibilita especificação dos valores por defeito)
- @return para retornar o valor

```
@function col-width($columns:12, $page-width:100%, $gap:1%){  
  @return ($page-width - $gap*($columns - 1))/ $columns;  
}
```

sass\_1.scss

- Chamada à função

```
#content {  
  float:left;  
  width:6*col-width(8);  
}  
  
#sidebar {  
  float:right;  
  width:2*col-width(8);  
}
```

sass\_1.scss

## Sass: Características Principais

### ■ Condições

- @if (...) {...} / @else if (...) {...} / @else {...}

\*.SCSS

```
$contrast:high;  
  
body {  
  @if ($contrast==high)  
    { color:black; }  
  @else if ($contrast==low)  
    { color:#DDD; }  
  @else{  
    color:$text-color;  
  }  
  font-family: sans-serif;  
  ...  
}
```

\*.CSS

```
body { color: black;  
       font-family: sans-serif; }
```

```
$contrast:low;
```

\*.CSS

```
body { color: #DDD;  
       font-family: sans-serif; }
```

## Sass: Características Principais

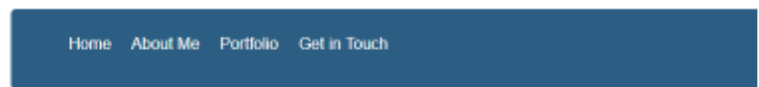
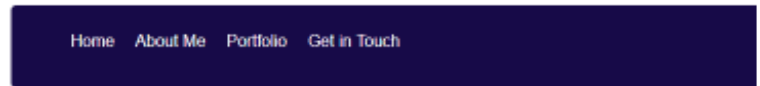
### ■ @if

- Podem condicionar de uma forma muito direta toda a formatação (ex.: paleta de cores)

```
// Allowed values theme: light; default
$theme:default;

$text-color:#222222;
$theme-color:#170a48;
$secondary-color:#f27731;
$ternary-color:#ccf962;
$menu-item-color:white;
$link-color:$secondary-color;

@if ($theme==light)
{
    $text-color:#d1bec2;
    $theme-color:#2b5e82;
    $secondary-color:#4b65c3;
    $ternary-color:#ccf962;
    $menu-item-color:white;
    $link-color:$secondary-color;
}
```



## Sass: Características Principais

### ■ Ciclos

#### ■ @for

- Repetir o processamento um número pré-determinado de vezes
- É possível utilizar variáveis em nomes de seletores, propriedades e respetivos valores recorrendo à notação `#{} (interpolation syntax)`

```
@for $i from 1 through 6 {
    .col-#{ $i }{
        width:$i*2em;
    }
}
```

\*.SCSS

```
.col-1 {
    width: 2em; }

.col-2 {
    width: 4em; }

.col-3 {
    width: 6em; }

.col-4 {
    width: 8em; }

.col-5 {
    width: 10em; }

.col-6 {
    width: 12em; }
```

\*.CSS

## Sass: Características Principais

### ■ Ciclos

#### ■ @each

- O mais versátil e o mais frequentemente utilizado

- Percorrer uma lista
- Percorrer um map

```
$font-sizes: (tiny:8px, small:11px, medium: 13px, large:16px);

@each $name, $size in $font-sizes{
  .#{$name}{
    font-size:$size;
  }
}
```

\*.scss

```
.tiny {
  font-size: 8px; }

.small {
  font-size: 11px; }

.medium {
  font-size: 13px; }

.large {
  font-size: 16px; }
```

\*.css

## Sass: Características Principais

### ■ Ciclos

#### ■ @while

- Exige que a variável que controla o ciclo seja incrementada, caso contrário origina-se um ciclo infinito

```
$j:2;

@while ($j<=8){
  .picture-#{$j}{
    width:$j * 10%;
  }
  $j:$j+2;
}
```

\*.scss

```
.picture-2 {
  width: 20%; }

.picture-4 {
  width: 40%; }

.picture-6 {
  width: 60%; }

.picture-8 {
  width: 80%; }
```

\*.css

- Apresentadas **apenas** as principais características deste pré-processador

- [http://sass-lang.com/documentation/file.SASS\\_REFERENCE.html#syntax](http://sass-lang.com/documentation/file.SASS_REFERENCE.html#syntax)

- Vantagens

- Organização/Modularidade do código
  - Refactoring
  - Funcionalidades não disponíveis nas CSS
    - Modularidade, reutilização de código; funções; operações matemáticas; ...
    - ...

- Desvantagens

- Curva de aprendizagem para implementar as novas features
  - ...

## Compressão CSS

*CSS Minify*



## Compressão CSS

### ■ CSS minify

- O objetivo principal é a redução do peso dos ficheiros CSS.
- Criado apenas um \*.css file de forma a minimizar os HTTP Requests
- Pode atingir uma redução de 20% relativamente ao código original
- Download mais rápido
- Torna mais confusa a leitura do código
  - Dificulta cópias ou processos de *reverse engineering*.

#### Before:

```
.ui-helper-hidden {  
    display: none;  
}  
.ui-helper-hidden-accessible {  
    border: 0;  
    overflow: hidden;  
}
```

#### After:

```
.ui-helper-hidden{display:none}.ui-helper-hidden-accessible{border:0; overflow: hidden;}
```

## Compressão CSS

### ■ CSS minify

- Remoção:
  - Comentários
  - Espaços em branco e novas linhas
  - Indentações
- Optimizar (reduzir) a designação dos seletores de ID, class, ...
- Ferramentas:



<https://wp-rocket.me/blog/best-free-css-and-javascript-minification-tools/>

# Exercício

## Exercício

- Qual é o output?

```
<style>
  #d1, #d2, #d3 {width:200px;height:100px;}

  #d1 {background-color:lightgray;}
  #d2 {background-color:orange;}
  #d3 {background-color:lightgreen;}
</style>
```

```
<div>
  <div id="d1"></div>
  <div id="d2"></div>
  <div id="d3"></div>
</div>
```



## Exercício

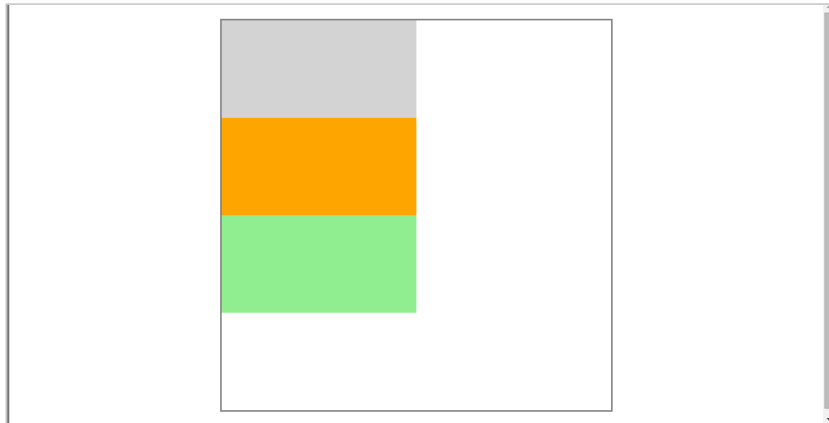
- Qual é o output?

```
<div id="container">
  <div id="d1"></div>
  <div id="d2"></div>
  <div id="d3"></div>
</div>
```

```
<style>
  #container { width:400px;height:400px;
               margin:20px auto;
               border: 2px gray solid;}

  #d1, #d2, #d3 {width:200px;height:100px;}

  #d1 {background-color:lightgray;}
  #d2 {background-color:orange;}
  #d3 {background-color:lightgreen;}
</style>
```



## Exercício

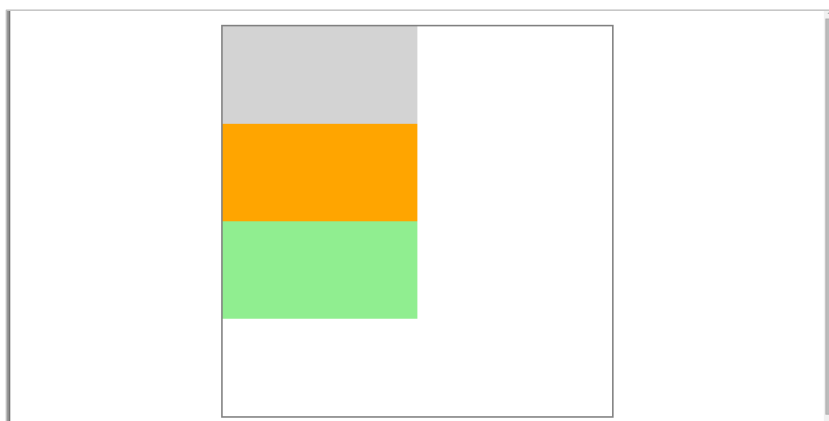
- Qual é o output?

```
<div id="container">
  <div id="d1"></div>
  <div id="d2"></div>
  <div id="d3"></div>
</div>
```

```
<style>
  #container { width:400px;height:400px;
               margin:20px auto;
               border: 2px gray solid;}

  #d1, #d2, #d3 {width:200px;height:100px;}

  #d1 {background-color:lightgray;}
  #d2 {background-color:orange;   left:200px}
  #d3 {background-color:lightgreen;}
</style>
```



## Exercício

- Qual é o output?

```
<style>
  #container { width:400px;height:400px;
               margin:20px auto;
               border: 2px gray solid;}
  #d1, #d2, #d3 {width:200px;height:100px;}

  #d1 {background-color:lightgray;}
  #d2 {background-color:orange; position: relative; left:200px}
  #d3 {background-color:lightgreen;}
</style>
```

```
<div id="container">
  <div id="d1"></div>
  <div id="d2"></div>
  <div id="d3"></div>
</div>
```



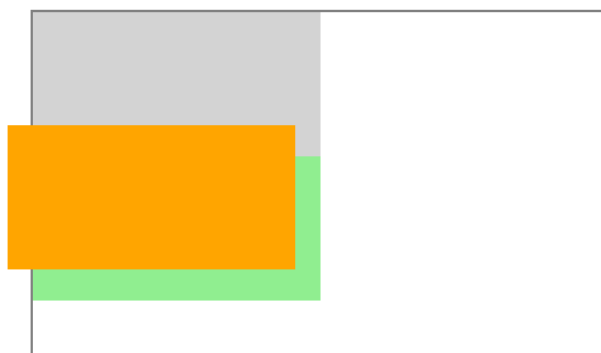
## Exercício

- Qual é o output?

```
<style>
  #container { width:400px;height:400px;
               margin:20px auto;
               border: 2px gray solid;}
  #d1, #d2, #d3 {width:200px;height:100px;}

  #d1 {background-color:lightgray;}
  #d2 {background-color:orange; position:absolute; left:200px; top:100px;}
  #d3 {background-color:lightgreen;}
</style>
```

```
<div id="container">
  <div id="d1"></div>
  <div id="d2"></div>
  <div id="d3"></div>
</div>
```



## Exercício

- Qual é o output?

```
<style>
  #container { width:400px;height:400px;
               margin:20px auto;
               border: 2px gray solid;
               position:relative; }
  #d1, #d2, #d3 {width:200px;height:100px;}

  #d1 {background-color:lightgray;}
  #d2 {background-color:orange; position:absolute; left:200px; top:100px;}
  #d3 {background-color:lightgreen;}
</style>
```



## Exercício

### Código?

(posicionamento absoluto)

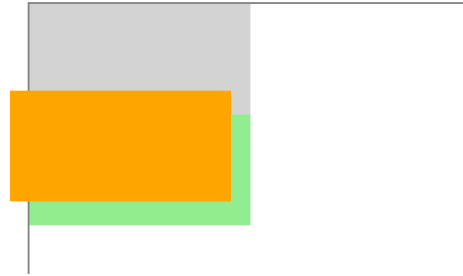


```
<div id="container">
  <div id="d1"></div>
  <div id="d2"></div>
  <div id="d3"></div>
</div>
```

```
<style>
  #container { width:400px;height:400px;
               margin:20px auto;
               border: 2px gray solid;
               position:relative; }
  #d1, #d2, #d3 {width:200px;height:100px;}

  #d1 {background-color:lightgray;}
  #d2 {background-color:orange; position:absolute; left:200px; top:100px;}
  #d3 {background-color:lightgreen; position:absolute; left:0px; top:200px;}
</style>
```

```
<div id="container">
  <div id="d1"></div>
  <div id="d2"></div>
  <div id="d3"></div>
</div>
```



```
#d1 {background-color:lightgray; position:relative; z-index:2;}
#d2 {background-color:orange; position:absolute; left:200px; top:100px; z-index:1;}
#d3 {background-color:lightgreen; position:relative; z-index:2;}
```

?



# Bootstrap5

*Frameworks CSS*

# Bootstrap

## ■ Bootstrap

- É um *framework* dedicado ao *Front-End development*.
  - Bootstrap5 é a nova versão lançada em janeiro/2020.
- Um conjunto de características que justificam a sua popularidade na implementação de aplicações Web
  - Responsive
  - Particularmente adaptado para ambientes *mobile*
  - Reduz o tempo de desenvolvimento
  - Assegura consistência
  - Assegura compatibilidade *cross browser*
  - Configurável
  - Possui uma comunidade de utilizadores muito alargada

# Bootstrap

## ■ Instalação

- 2 opções mais vulgarmente utilizadas:
  - Download

### Compiled CSS and JS

Download ready-to-use compiled code for **Bootstrap v5.0.2** to easily drop into your project, which includes:

- Compiled and minified CSS bundles (see [CSS files comparison](#))
- Compiled and minified JavaScript plugins (see [JS files comparison](#))

This doesn't include documentation, source files, or any optional JavaScript dependencies like Popper.

[Download](#)

- Incorporação via CDN

### CDN via jsDelivr

Skip the download with [jsDelivr](#) to deliver cached version of Bootstrap's compiled CSS and JS to your project.

```
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css" rel="stylesheet">  
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js" integrity="sha384-MQLwNkjR1J7sF/tf3keJ98ca+so2eQsJ0U2xtJ5R8q6p4tP9sPY5N+2rvNUyYhsYh57wQivlNbh8kpc/y" crossorigin="anonymous"></script>
```

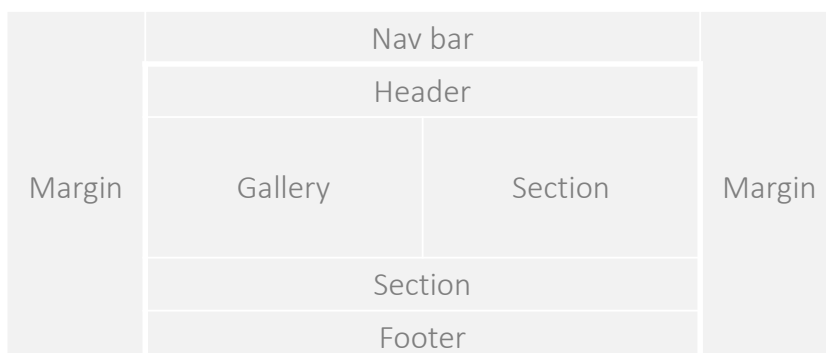
# Grid System

*Bootstrap5*

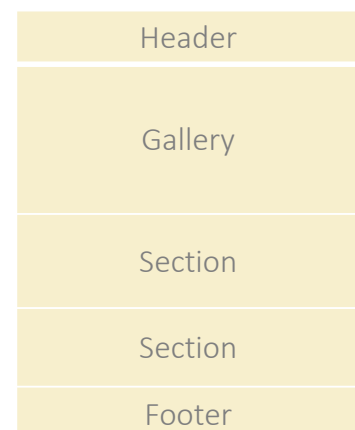
## Bootstrap: Grid System

- A construção de layouts é baseado no **Grid System**
  - Permite criar um *responsive layout*
  - Assume duas formas distintas de estruturar a disposição de informação

*Layout 1  
(desktop)*



*Layout 2 – 1 column  
(tablet / mobile)*



*“ ... mobile-first flexbox grid to build layouts of all shapes and sizes thanks to a twelve column system, five default responsive tiers, ....”*

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



## Bootstrap: Grid System

- A estrutura base disponibilizada pelo Bootstrap para implementação da Grid System é baseada em 3 conceitos chave:

- Container (C)

- O *wrapper* que permite o alinhamento horizontal do conteúdo e define a forma como a aplicação se adapta a diferentes viewports

- Exemplo: **.container-fluid** permite uma *width:100%* para todos os viewports

- Row (R)

- *Wrapper* de colunas, permite criar grupos de colunas.

- Columns

- Podem ser definidas 12 colunas por linha, às quais pode ser aplicado um agrupamento (span) de colunas (exemplo: **.col-4**).

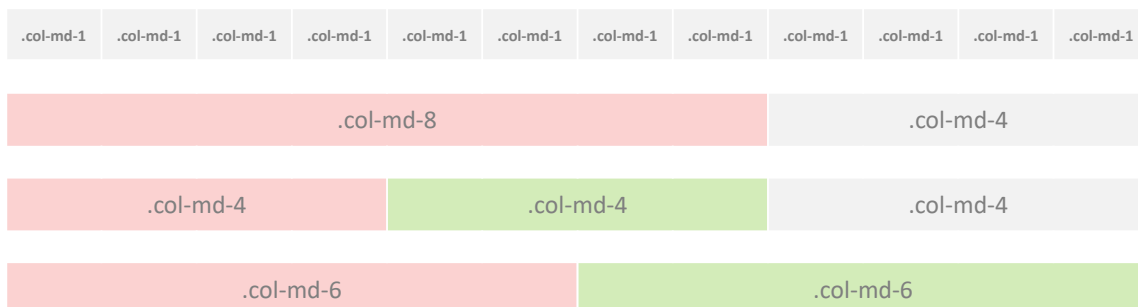
- É um elemento flexível cujo redimensionamento é automático.



## Bootstrap: Grid System

- Columns

- 12 colunas base, as quais podem ser sujeitas a um agrupamento horizontal (*spanning*)



- **.col-md-\***

- cria uma coluna com um agrupamento de colunas definido por \*
    - **md** especifica o breakpoint onde a coluna altera a sua largura
      - md →  $screen \geq 768px$  significa que as colunas irão ocupar 100% da largura disponível para ecrãs com largura inferior aos 768px (Layout 2)

# Bootstrap: Grid System

## Breakpoints

- definidos tendo por base:
  - múltiplos de 12
  - são representativos de um subconjunto de dispositivos e dimensões de viewport comuns
    - Não são específicos para nenhum dispositivo, mas asseguram que um funcionamento consistente para todos os dispositivos no mercado.
- definem em que classe de dispositivos se inicia a configuração definida:
  - Permitem escalar o layout definido para viewports de dimensão superior
  - Pelo contrário, para viewports de dimensão inferior (mobile), o layout 2 é assumido.

Breakpoints		
X-small	-	0-576px
Small	sm	≥ 576px
Medium	md	≥ 768px
Large	lg	≥ 992px
Extra Large	xl	≥ 1200px
Extra Extra large	xxl	≥ 1400px

# Bootstrap: Grid System

## Configurações Comuns:

- 2 colunas
  - As 2 colunas **iniciam-se ≥ 768px e escalam** para viewports de dimensão superior



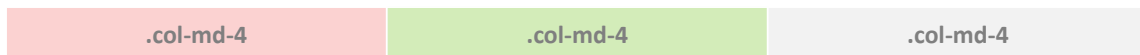
```
<div class="container">
  <div class="row">
    <div class="col-md-8">.col-md-8</div>
    <div class="col-md-4">.col-md-4</div>
  </div>
</div>
```

- para viewports inferiores a 768px, as colunas irão assumir 100% da largura disponível (layout 2)

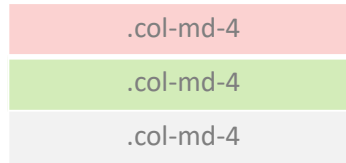


## Bootstrap: Grid System

- 3 colunas



- $vp \leq 768px$



```
<div class="container">
  <div class="row">
    <div class="col-md-4">.col-md-4</div>
    <div class="col-md-4">.col-md-4</div>
    <div class="col-md-4">.col-md-4</div>
  </div>
</div>
```

## Bootstrap: Grid System

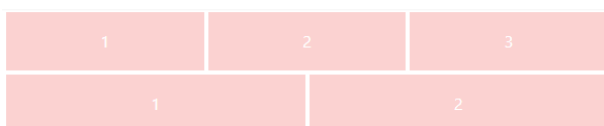
- Caso sejam utilizadas classes não baseadas nos *break points*, altera-se o comportamento **responsive**:

- Em *viewports* reduzidos mantém-se o número de colunas original

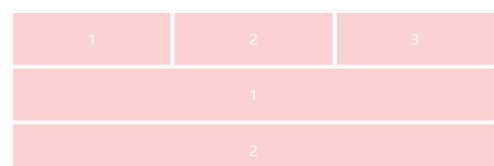
```
<div class="container-fluid">
  <div class="row">
    <div class="col">1</div>
    <div class="col">2</div>
    <div class="col">3</div>
  </div>
  <div class="row">
    <div class="col-sm">1</div>
    <div class="col-sm">2</div>
  </div>
</div>
```

class **col** mantém a definição original

class **col-sm** passam a estar sobrepostas (layout2)



Redução do viewport



- tipicamente utilizam-se classes baseadas nos *breakpoints* e com dimensão definida
  - exemplo: `.col-md-4`; `.col-md-8`; `.col-sm-6`, ...

## Containers

- Existem diversas classes pré-definidas para estabelecer o comportamento dos containers

**C**

- .container → 100% width ≤ 540px
- .container-sm → 100% width ≤ 540px
- .container-md → 100% width ≤ 720px
- .container-lg → 100% width ≤ 960px
- .container-xl → 100% width ≤ 1140px
- .container-xxl → 100% width ≤ 1320px
- .container-fluid → 100% width

```
<div class="container">.container</div>
<div class="container-sm">.container-sm</div>
<div class="container-md">.container-md</div>
<div class="container-lg">.container-lg</div>
<div class="container-xl">.container-xl</div>
<div class="container-xxl">.container-xxl</div>
<div class="container-fluid">.container-fluid</div>
```

## Containers

- Responsive para os diversos breakpoints

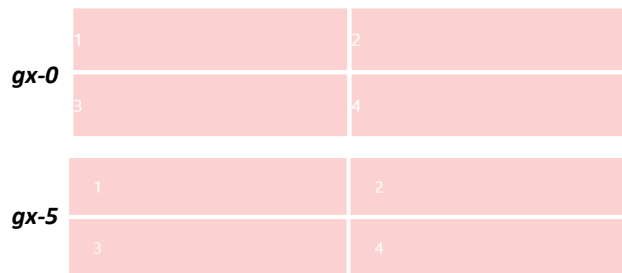
	Extra small <576px	Small ≥576px	Medium ≥768px	Large ≥992px	X-Large ≥1200px	XX-Large ≥1400px
.container	100%	540px	720px	960px	1140px	1320px
.container-sm	100%	540px	720px	960px	1140px	1320px
.container-md	100%	100%	720px	960px	1140px	1320px
.container-lg	100%	100%	100%	960px	1140px	1320px
.container-xl	100%	100%	100%	100%	1140px	1320px
.container-xxl	100%	100%	100%	100%	100%	1320px
.container-fluid	100%	100%	100%	100%	100%	100%

<https://getbootstrap.com/docs/5.0/layout/containers/>

## ■ gutter

### ■ **gx** define o padding

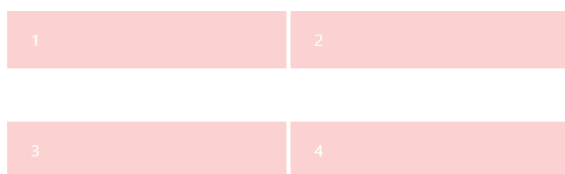
#### ■ gx-0 a gx-5



```
<div class="container">
  <div class="row gx-5">
    <div class="col-6">1</div>
    <div class="col-6">2</div>
    <div class="col-6">3</div>
    <div class="col-6">4</div>
  </div>
</div>
```

### ■ **gy** define margin vertical

#### ■ gy-0 a gy-5



#### ■ **g-5** é equivalente a **gx-5; gy-5**

```
<div class="container">
  <div class="row gx-5 gy-5">
    <div class="col-6">1</div>
    <div class="col-6">2</div>
    <div class="col-6">3</div>
    <div class="col-6">4</div>
  </div>
</div>
```

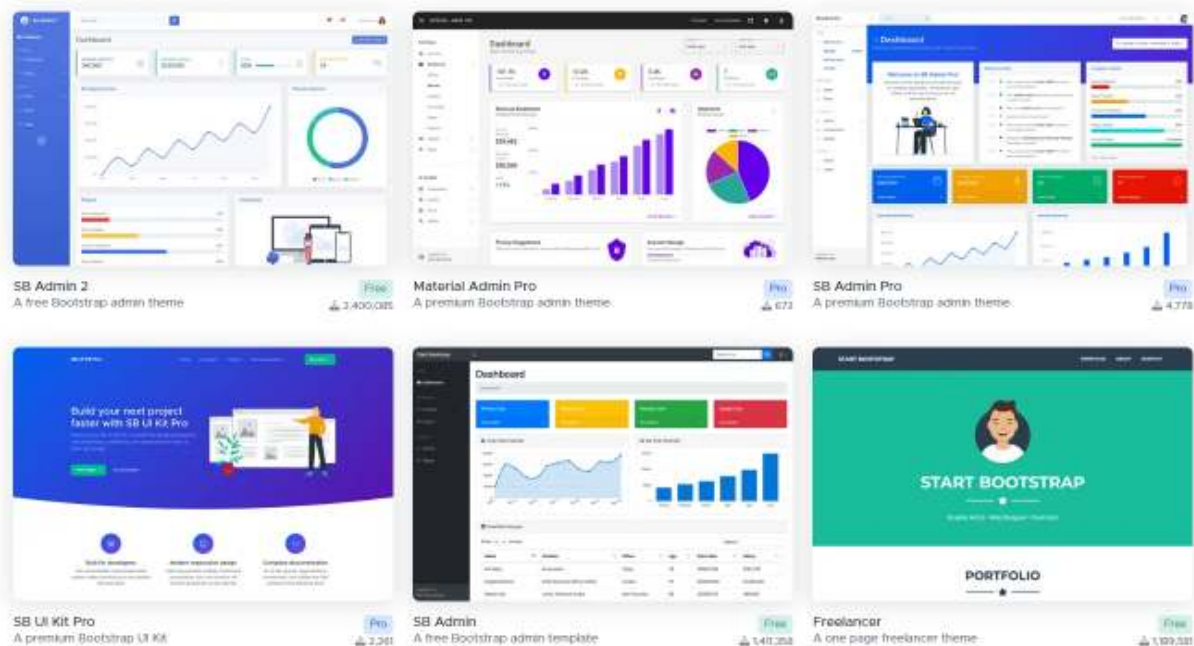
# Templates

*Bootstrap5*

# Bootstrap: Templates

- Existe um conjunto alargado de templates disponíveis para download

■ <https://startbootstrap.com/>



## Components

*Bootstrap5*

# Bootstrap: Components

- Elementos com formatação/funcionalidade definidas (reutilização)
  - Alguns componentes permitem configurações adicionais

■ <https://getbootstrap.com/docs/5.0/components/accordion/>

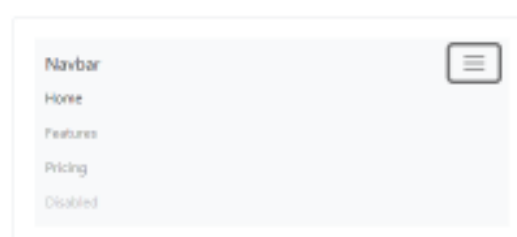
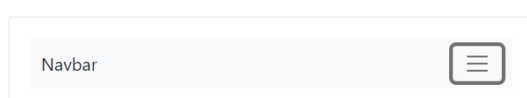
## Components

Accordion  
Alerts  
Badge  
Breadcrumb  
Buttons  
Button group  
Card  
Carousel  
Close button  
Collapse  
Dropdowns  
List group  
Modal  
Navs & tabs  
Navbar  
Offcanvas  
Pagination  
Popovers  
Progress  
Scrollspy  
Spinners



# Bootstrap: Components

- **Navbar**
  - Barra de navegação, totalmente configurável, responsiva em que nos *viewports* de menor dimensão assume um funcionamento em *sandwich button*



# Bootstrap: Components

## ■ Buttons

- É disponibilizado um conjunto diversificado de botões, os quais podem ser diretamente utilizados

■ <https://getbootstrap.com/docs/4.0/components/buttons/>

### Buttons



```
<div class="container" >
  <div class="row g-5">
    <h1>Buttons</h1>
    <div>
      <button type="button" class="btn btn-primary">Primary</button>
      <button type="button" class="btn btn-secondary">Secondary</button>
      ...
    </div>
  </div>
</div>
```

# Bootstrap: Components

## ■ Buttons

- São disponibilizadas várias classes que permitem definir cor e dimensão dos botões

### Buttons



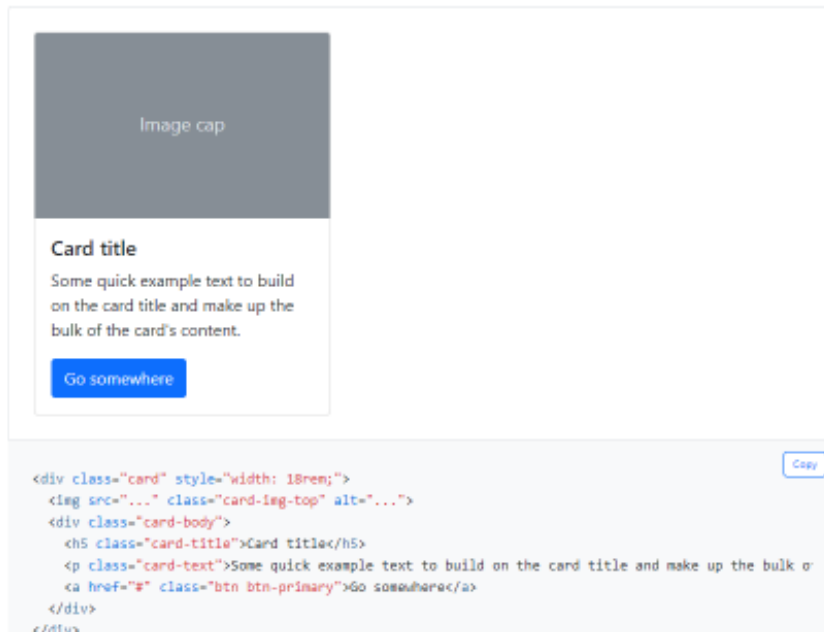
```
<div class="container" >
  <div class="row g-5">
    <h1>Buttons</h1>
    <div>
      <button type="button" class="btn btn-primary btn-lg">Primary</button>
      <button type="button" class="btn btn-secondary btn-sm">Secondary</button>
      <button type="button" class="btn btn-success">Success</button>
    </div>
  </div>
</div>
```



# Bootstrap: Components

## ■ Cards

- Componente que permite associar de forma direta imagem / título / conteúdo textual



# Bootstrap: Components

## ■ Carousel

- A inserção de um slide show é imediata e pode ser configurada de diversas formas
  - <https://getbootstrap.com/docs/5.1/components/carousel/>

```
<div id="carouselExampleControls" class="carousel slide" >
  <div class="carousel-inner">
    <div class="carousel-item active">
      
    </div>
    ....
    <a class="carousel-control-prev" href="#carouselExampleControls" role="button" data-slide="prev">
      <span class="carousel-control-prev-icon" aria-hidden="true"></span>
      <span class="sr-only">Previous</span>
    </a>
    <a class="carousel-control-next" href="#carouselExampleControls" role="button" data-slide="next">
      <span class="carousel-control-next-icon" aria-hidden="true"></span>
      <span class="sr-only">Next</span>
    </a>
  </div>
```



# Bootstrap: Components

## ■ Navs & Tabs

```
<ul class="nav">
  <li class="nav-item">
    <a class="nav-link active" aria-current="page" href="#">Active</a>
  </li>
  <li class="nav-item">
    <a class="nav-link" href="#">Link</a>
  </li>
  <li class="nav-item">
    <a class="nav-link" href="#">Link</a>
  </li>
  <li class="nav-item">
    <a class="nav-link disabled" href="#" tabindex="-1" aria-disabled="true">Disabled</a>
  </li>
</ul>
```

Active Link Link Disabled

```
<ul class="nav nav-tabs">
  <li class="nav-item">
    <a class="nav-link active" aria-current="page" href="#">Active</a>
  </li>
  <li class="nav-item">
    <a class="nav-link" href="#">Link</a>
  </li>
  <li class="nav-item">
    <a class="nav-link" href="#">Link</a>
  </li>
  <li class="nav-item">
    <a class="nav-link disabled" href="#" tabindex="-1" aria-disabled="true">Disabled</a>
  </li>
</ul>
```

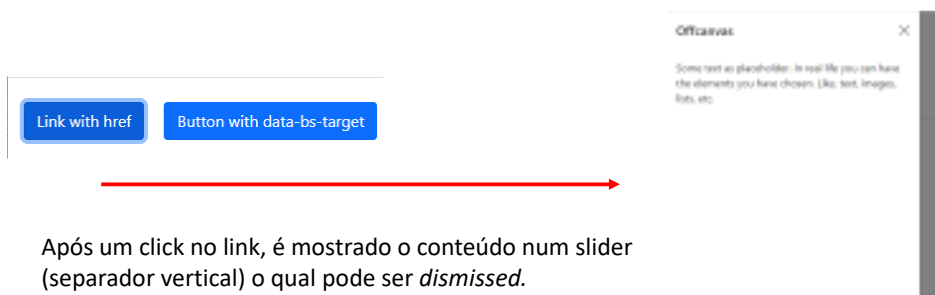
Active Link Link Disabled

# Bootstrap: Components

## ■ Offcanvas

- Novo componente no Bootstrap5, permite a visualização de conteúdo (definido offcanvas) através de um *slider*

```
<a class="btn btn-primary" data-bs-toggle="offcanvas" href="#offcanvasExample" role="button">Link with href </a>
<button class="btn btn-primary" type="button" data-bs-toggle="offcanvas" data-bs-target="#offcanvasExample">
  Button with data-bs-target
</button>
<div class="offcanvas offcanvas-start" tabindex="-1" id="offcanvasExample" aria-labelledby="offcanvasExampleLabel">
  <div class="offcanvas-header">
    <h3 class="offcanvas-title" id="offcanvasExampleLabel">Offcanvas</h3>
    <button type="button" class="btn-close text-reset" data-bs-dismiss="offcanvas" aria-label="Close"></button>
  </div>
  <div class="offcanvas-body">
    <div> Some text as placeholder. In real life you can have the elements you have chosen. Like, text, images, lists, etc.</div>
  </div>
</div>
```



Após um click no link, é mostrado o conteúdo num slider (separador vertical) o qual pode ser *dismissed*.

# Bootstrap: Components

## ■ *Acordion*

- Muito útil para a implementação de FAQ's

Accordion Item #1	▼
Accordion Item #2	▼
Accordion Item #3	▼

Accordion Item #1	^
<strong>This is the first item's accordion body.</strong> It is shown by default, until the collapse plugin adds the appropriate classes that we use to style each element. These classes control the overall appearance, as well as the showing and hiding via CSS transitions. You can modify any of this with custom CSS or overriding our default variables. It's also worth noting that just about any HTML can go within the <code>.accordion-body</code> , though the transition does limit overflow.	
Accordion Item #2	▼
Accordion Item #3	▼

```
<div class="accordion" id="accordionExample">
  <div class="accordion-item">
    <h2 class="accordion-header" id="headingOne">
      <button class="accordion-button" type="button" data-bs-toggle="collapse" data-bs-target="#collapseOne" aria-expanded="true"
        aria-controls="collapseOne"> Accordion Item #1 </button></h2>
    <div id="collapseOne" class="accordion-collapse collapse show" aria-labelledby="headingOne" data-bs-parent="#accordionExample">
      <div class="accordion-body">
        <strong>This is the first item's accordion body.</strong> It is shown by default, ....
      </div>
    </div>
  </div>
  ....
</div>
```

# Bootstrap: Components

## ■ Alerts

- Reforçar a interação com o utilizador.

```
<div class="alert alert-primary" role="alert">
  A simple primary alert—check it out!
</div>
```



- Eliminar o *alert*

```
<div class="alert alert-warning alert-dismissible fade show">
  Dismiss!!
  <button type="button" class="btn-close" data-bs-dismiss="alert" ></button>
</div>
```

Dismiss!!



# Utilities

## Bootstrap5

## Bootstrap: Utilities

- **Utilities** agrega um conjunto de classes específicas para definir texto, dimensões, cores, cores de fundo, espaçamentos, ....
  - <https://getbootstrap.com/docs/5.0/utilities/>

### ▼ Utilities

API

Background

Borders

Colors

Display

Flex

Float

Interactions

Overflow

Position

Shadows

Sizing

Spacing

Text

Vertical align

Visibility

.bg-primary

.bg-secondary

.bg-success

.bg-danger

.bg-warning

.bg-info

.bg-light

.bg-dark

# Bootstrap: Utilities

## Utilities: Spacing

### Notação:

- {property}{sides}-{size} para *breakpoint* xs
- {property}{sides}-{breakpoint}-{size} para *breakpoints* sm, md, lg, xl e xxl.

### property

- margin: **m**
- padding: **p**

### sides

t - for classes that set **margin-top** or **padding-top**  
b - for classes that set **margin-bottom** or **padding-bottom**  
s - (start) for classes that set **margin-left** or **padding-left**  
e - (end) for classes that set **margin-right** or **padding-right**  
x - for classes that set both **\*-left** and **\*-right**  
y - for classes that set both **\*-top** and **\*-bottom**  
blank - for classes that set a **margin** or **padding** on all 4 sides

### size

0 - for classes that eliminate the **margin** or **padding** by setting it to 0  
1 - (by default) for classes that set the **margin** or **padding** to \$spacer \* .25  
2 - (by default) for classes that set the **margin** or **padding** to \$spacer \* .5  
3 - (by default) for classes that set the **margin** or **padding** to \$spacer  
4 - (by default) for classes that set the **margin** or **padding** to \$spacer \* 1.5  
5 - (by default) for classes that set the **margin** or **padding** to \$spacer \* 3  
auto - for classes that set the **margin** to auto

<https://getbootstrap.com/docs/5.0/utilities/spacing/>

# Bootstrap: Utilities

## Utilities: Spacing

### Padding:

```
<div>
<p>
  Lorem ipsum dolor sit amet, consectetur adipisicing elit. ... suscipit ab.</p>
</div>
```

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Voluptatum unde incidunt nobis doloremque exercitationem nihil accusantium voluptatibus? Doloribus optio doloremque nihil veritatis, magni, blanditiis modi dicta temporibus voluptates suscipit ab.

### horizontal/vertical

```
<div class="px-5 py-3" >
<p>
  Lorem ipsum dolor sit amet, consectetur adipisicing elit. ... suscipit ab.</p>
</div>
```

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Voluptatum unde incidunt nobis doloremque exercitationem nihil accusantium voluptatibus? Doloribus optio doloremque nihil veritatis, magni, blanditiis modi dicta temporibus voluptates suscipit ab.

### Todas as direções

```
<div class="p-5" >
<p>
  Lorem ipsum dolor sit amet, consectetur adipisicing elit. ... suscipit ab.</p>
</div>
```

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Voluptatum unde incidunt nobis doloremque exercitationem nihil accusantium voluptatibus? Doloribus optio doloremque nihil veritatis, magni, blanditiis modi dicta temporibus voluptates suscipit ab.

# Bootstrap: Utilities

## ■ Utilities: Spacing

### ■ margin:

```
<div class="p-1" >
  <p>
    Lorem ipsum dolor sit amet, consectetur adipisicing elit. ... suscipit ab.</p>
  <p>
    Lorem ipsum dolor sit amet, consectetur adipisicing elit. ... suscipit ab.</p>
</div>
```

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Voluptatum unde incidunt nobis doloremque exercitationem nihil accusantium voluptatibus? Doloribus optio doloremque nihil veritatis, magni, blanditiis modi dicta temporibus voluptates suscipit ab.

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Voluptatum unde incidunt nobis doloremque exercitationem nihil accusantium voluptatibus? Doloribus optio doloremque nihil veritatis, magni, blanditiis modi dicta temporibus voluptates suscipit ab.

### ■ exemplo: (*mt* → *margin top*; *m* → *margin*)

```
<div class="p-1" >
  <p>
    Lorem ipsum dolor sit amet, consectetur adipisicing elit. ... suscipit ab.</p>
  <p class="mt-3">
    Lorem ipsum dolor sit amet, consectetur adipisicing elit. ... suscipit ab.</p>
</div>
```

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Voluptatum unde incidunt nobis doloremque exercitationem nihil accusantium voluptatibus? Doloribus optio doloremque nihil veritatis, magni, blanditiis modi dicta temporibus voluptates suscipit ab.

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Voluptatum unde incidunt nobis doloremque exercitationem nihil accusantium voluptatibus? Doloribus optio doloremque nihil veritatis, magni, blanditiis modi dicta temporibus voluptates suscipit ab.

# Bootstrap: Utilities

## ■ Utilities: Alinhamento de texto

```
<div class="p-2 text-center" >
  <p>
    Lorem ipsum dolor sit amet, consectetur adipisicing elit. ... suscipit ab.</p>
  <p>
    Lorem ipsum dolor sit amet, consectetur adipisicing elit. ... suscipit ab.</p>
</div>
```

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Voluptatum unde incidunt nobis doloremque exercitationem nihil accusantium voluptatibus? Doloribus optio doloremque nihil veritatis, magni, blanditiis modi dicta temporibus voluptates suscipit ab.

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Voluptatum unde incidunt nobis doloremque exercitationem nihil accusantium voluptatibus? Doloribus optio doloremque nihil veritatis, magni, blanditiis modi dicta temporibus voluptates suscipit ab.

```
<div class="w-25 p-3 text-center mx-auto my-4 border" >
  <p>
    Lorem ipsum dolor sit amet, consectetur adipisicing elit. ... suscipit ab.</p>
  <p>
    Lorem ipsum dolor sit amet, consectetur adipisicing elit. ... suscipit ab.</p>
</div>
```

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Voluptatum unde incidunt nobis doloremque exercitationem nihil accusantium voluptatibus? Doloribus optio doloremque nihil veritatis, magni, blanditiis modi dicta temporibus voluptates suscipit ab.

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Voluptatum unde incidunt nobis doloremque exercitationem nihil accusantium voluptatibus? Doloribus optio doloremque nihil veritatis, magni, blanditiis modi dicta temporibus voluptates suscipit ab.

- **w-25** (width 25%) {25, 50, 75, 100}
- **p-3** (padding nível 3) {0,1,2,3,4,5}
- **text-center** ( texto com alinhamento centrado)
- **mx-auto** (margens horizontais automaticas, elemento centrado)
- **my-4** (margens verticais de nível 4) {0,1,2,3,4,5}
- **border**

# Images

## Bootstrap5

## Bootstrap: Imagens

### ■ Imagens Fluídas

#### ■ O Bootstrap possui uma class especifica para tornar uma imagem flexível

##### ■ imagem sem formatação associada

```
<div class="w-25 p-3 text-center mx-auto my-4 border" >  
  <p>Lorem ipsum dolor sit amet, consectetur adipisicing elit. ... suscipi ab.</p>  
  <p>Lorem ipsum dolor sit amet, consectetur adipisicing elit. ... suscipi ab.</p>  
    
</div>
```



##### ■ imagem flexível com capacidade de ajustar automaticamente ao espaço disponível

```
<div class="w-25 p-3 text-center mx-auto my-4 border" >  
  <p>Lorem ipsum dolor sit amet, consectetur adipisicing elit. ... suscipi ab.</p>  
  <p>Lorem ipsum dolor sit amet, consectetur adipisicing elit. ... suscipi ab.</p>  
    
</div>
```



# Tables

## Bootstrap5

## Bootstrap: Tables

- Tabelas
  - Estão disponíveis um conjunto de classes associadas às tabelas que permitem uma formatação completa deste elemento
    - A estrutura HTML permanece inalterada

```
<div class="mx-auto w-50">
  <table class="table">
    <thead>
      <tr>
        <th
          scope="col">#</th>
        <th
          scope="col">First</th>
        <th
          scope="col">Last</th>
        <th
          scope="col">Handle</th>
      </tr>
    </thead>
    <tbody>
      <tr>
        ...
      </tr>
      ...
    </tbody>
  </table>
</div>
```

#	First	Last	Handle
1	Mark	Otto	@mdo
2	Jacob	Thornton	@fat
3	Larry the Bird		@twitter



## ■ Class específicas para tabelas

```
<div class="mx-auto w-50">  
  <table class="table table-light table-striped table-hover">  
    ...  
  </table>  
</div>
```

### ■ *table-light*

- Cor da tabela (tema)

### ■ *table-striped*

- Linhas alternadas

### ■ *table-hover*

- Efeito de sobreposição do cursor do rato

Class	Heading	Heading
Default	Cell	Cell
Primary	Cell	Cell
Secondary	Cell	Cell
Success	Cell	Cell
Danger	Cell	Cell
Warning	Cell	Cell
Info	Cell	Cell
Light	Cell	Cell
Dark	Cell	Cell

#	First	Last	Handle
1	Mark	Otto	@mdo
2	Jacob	Thornton	@fat
3	Larry the Bird		@twitter

# Forms

## Bootstrap5

# Bootstrap: Forms

- São disponibilizados diversos campos/elementos de formulários.
- Todos os elementos podem ser configuráveis

First name: Simao  
Last name: Paredes  
Username: @  
City:   
State: Choose...  
Zip:   
☐ Agree to terms and conditions  
Submit form

- Podem ser definidas validações locais, as quais também podem ser configuráveis

First name: Simao ✓ Looks good!  
Last name: Paredes ✓ Looks good!  
Username: @ ✓  
City: Please provide a valid city.  
State: Choose... Please select a valid state.  
Zip: Please provide a valid zip.  
☐ Agree to terms and conditions  
You must agree before submitting.  
Submit form

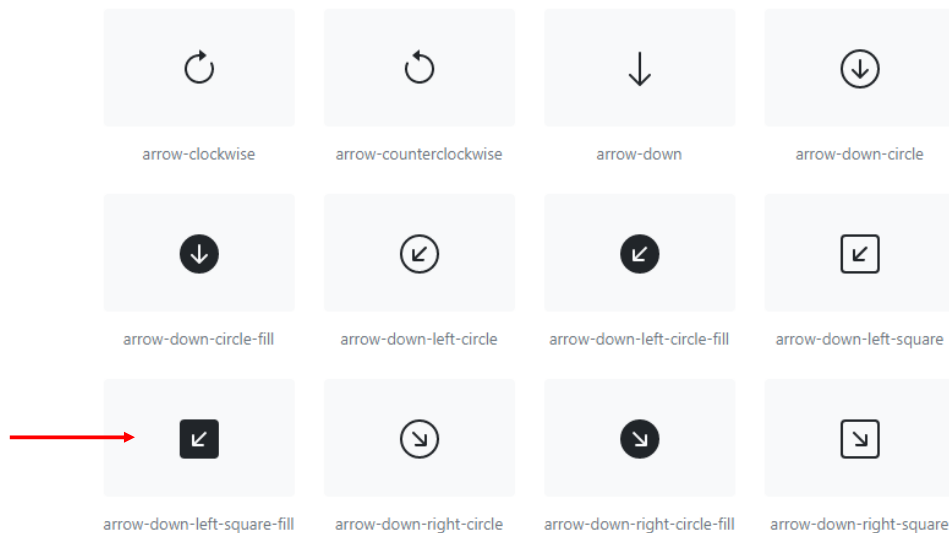
Validação do formulário é efetuada tendo por base duas pseudo-classes **:invalid** e **:valid**, as quais são aplicadas a todas as <tags> de inserção de dados.

## Icons

*Bootstrap5*

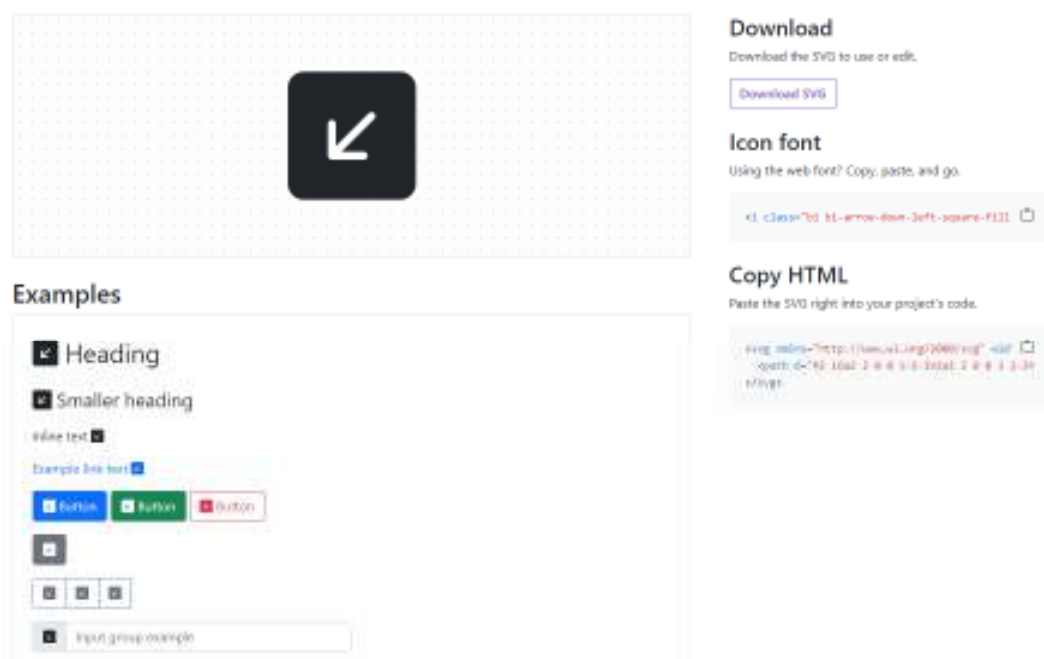
## Bootstrap: Icons

- São disponibilizadas centenas de icons implementados em SVG
  - <https://icons.getbootstrap.com/>
  - A inserção destes icons é direta, apenas é necessária a sua seleção e cópia do respetivo código SVG



## Bootstrap: Icons

- Após a seleção do *icon* pretendido, é possível editar diretamente o respetivo código SVG



# Customize

## Bootstrap5

## Bootstrap: Customize

- As formatações do bootstrap podem ser alteradas de forma a corresponder a necessidades específicas de uma aplicação.
  - este processo de customization tem várias opções disponíveis, sempre suportados pelos ficheiros Sass (\*.scss) originais.
    - <https://getbootstrap.com/docs/5.0/customize/overview/>

### ▼ Customize

#### Overview

#### Sass

#### Options

#### Color

#### Components

#### CSS variables

#### Optimize

## Sass

[View on GitHub](#)

Utilize our source Sass files to take advantage of variables, maps, mixins, and functions to help you build faster and customize your project.

## ■ Exemplo:

- Modificação do valor base das variáveis Sass, após o que são gerados os \*.css com as alterações produzidas.

Variable	Values	Description
<code>\$spacer</code>	<code>1rem</code> (default), or any value <code>&gt; 0</code>	Specifies the default spacer value to programmatically generate our <a href="#">spacer utilities</a> .
<code>\$enable-rounded</code>	<code>true</code> (default) or <code>false</code>	Enables predefined <code>border-radius</code> styles on various components.
<code>\$enable-shadows</code>	<code>true</code> or <code>false</code> (default)	Enables predefined decorative <code>box-shadow</code> styles on various components. Does not affect <code>box-shadows</code> used for focus states.
<code>\$enable-gradients</code>	<code>true</code> or <code>false</code> (default)	Enables predefined gradients via <code>background-image</code> styles on various components.
<code>\$enable-transitions</code>	<code>true</code> (default) or <code>false</code>	Enables predefined <code>transitions</code> on various components.
<code>\$enable-reduced-motion</code>	<code>true</code> (default) or <code>false</code>	Enables the <a href="#">prefers-reduced-motion media query</a> , which suppresses certain animations/transitions based on the users' browser/operating system preferences.
<code>\$enable-grid-classes</code>	<code>true</code> (default) or <code>false</code>	Enables the generation of CSS classes for the grid system (e.g. <code>.row</code> , <code>.col-md-1</code> , etc.).