

# Introdução às tecnologias Web - ITW

## Aula 3 – Estilos (CSS) & Marcadores HTML

# Sumário

Revisões

Seletores CSS

- Seletores básicos

- Combinadores

- Pseudo-classes, Pseudo elementos

Formatação de Texto em CSS

- Indentação, Alinhamento, Efeitos, Espaçamento, Transformação

CSS Icons

- Icon Fonts



# Estrutura base de um documento html

[relembrando...]

```
<!doctype html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="Author" content="Joaquim Sousa Pinto">
  <meta name="Keywords" content="exemplos">
  <meta name="Description" content="Exemplos das aulas">
  <link href="https://fonts.googleapis.com/css2?family=Open+Sans&display=swap" rel="stylesheet">
  <title>Titulo do Documento</title>
  <style>
    body { font-family: 'Open Sans', sans-serif; }
  </style>
</head>
<body>
</body>
</html>
```

**<head> </head>** - Cabeçalho do documento **[Este conteúdo não é representado.]**

- É aqui que definimos o conjunto de letras utilizado na página (UTF-8)
- Aqui podem ser colocados estilos css

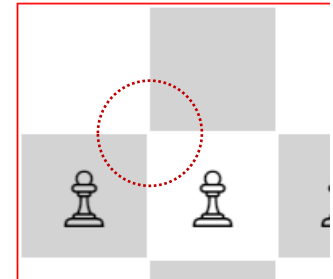
**<body> </body>** - Conteúdo do documento **[Este conteúdo é representado]**

- É aqui que deve ser inserido o código HTML criado

# Prática 2

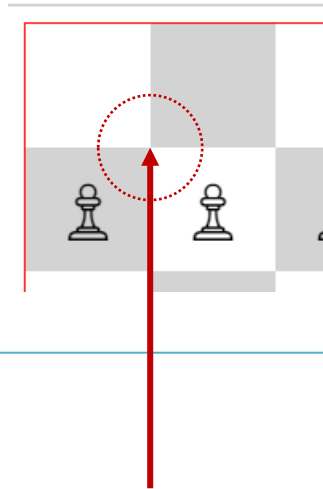
(dúvidas que ficaram no ar ...)

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8" />
  <title>Exercício 2 - Tabuleiro de xadrez</title>
  <style>
    table { margin: auto; border: solid 1px red; }
    td { text-align: center; vertical-align: middle; font-size: 60px; height: 80px; width: 80px; }
    td.branco { background-color: ivory; }
    td.preto { background-color: #888; }
  </style>
</head>
<body>
  <table>
    <tr>
      <td class="branco">#9814;</td><td class="preto">#9816;</td>
      <td class="branco">#9815;</td><td class="preto">#9812;</td>
      <td class="branco">#9813;</td><td class="preto">#9815;</td>
      <td class="branco">#9816;</td><td class="preto">#9814;</td>
    </tr>
    <tr>
      <td class="preto">#9817;</td><td class="branco">#9817;</td>
      <td class="preto">#9817;</td><td class="branco">#9817;</td>
      <td class="preto">#9817;</td><td class="branco">#9817;</td>
      <td class="preto">#9817;</td><td class="branco">#9817;</td>
    </tr>
  </table>
</body>
</html>
```



# Prática 2

(dúvidas que ficaram no ar ...)



```
<head>
  <meta charset="utf-8" />
  <title>Exercício 2 - Tabuleiro de xadrez</title>
  <style>
    table { margin: auto; border: solid 1px red; border-spacing: 0px; }
    td { text-align: center; vertical-align: middle; font-size: 60px; height: 80px; width: 80px; }
    td.branco { background-color: ivory; }
    td.preto { background-color: #888; }
  </style>
</head>
```

# Elementos HTML de agrupamento

div & span (revisões)

# Elementos HTML de agrupamento

Todos os elementos html são passíveis de posicionamento e agrupamento.

Assim, uma tabela (`<table></table>`) agrupa todos os elementos que se encontram dentro dela bem como um parágrafo (`<p></p>`) agrupa todas as palavras e letras nele contidas mas ...

Como fazer caso se pretenda **agrupar um conjunto de parágrafos**?

*ou*

Como **realçar apenas um conjunto de palavras** dentro de uma frase?

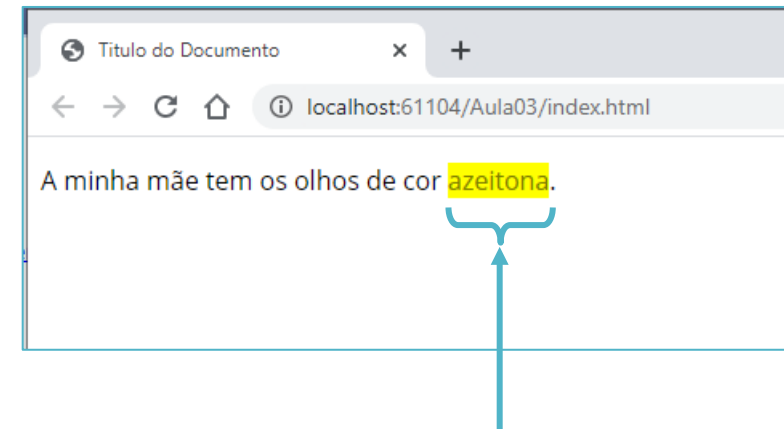
# Elementos HTML

`<span> </span>`

O marcador `<span>` é utilizado para agrupar elementos “inline” HTML.

Exemplo:

```
<p>A minha mãe tem os olhos de cor <span style="color: olive; background-color: yellow;">azeitona</span>.</p>
```



Note que o `<span> </span>` faz a o realce **dentro** do parágrafo `<p> </p>` e apenas na zona do marcador (azeitona).



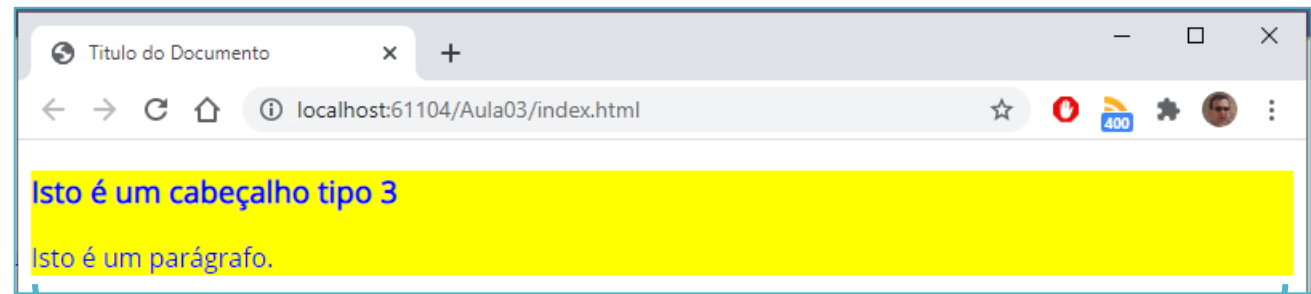
# Elementos HTML

`<div> </div>`

O marcador `<div> </div>` define uma divisão ou uma secção num documento HTML.

Exemplo:

```
<div style="color:#0000FF; background: #FFFF00;">  
  <h3>Isto é um cabeçalho tipo 3</h3>  
  <p>Isto é um parágrafo.</p>  
</div>
```



Note que o `<div> </div>` faz a o realce **a toda a largura do browser.**

Normalmente, os elementos <div> ocupam linhas inteiras e os browsers colocam uma quebra de linha antes e outra depois do elemento. Contudo, isso pode ser mudado através de CSS.

Exemplos:

```
<div style="width: 200px; height : 100px;">Eu sou um div</div>  
<div style="display: inline-block;">Eu também sou um div</div>  
<div style="float: left;">Eu sou o terceiro div</div>
```



# Seletores CSS

## Seletores de tipo (recordando...)

Os seletores de tipo permitem combinar estilos pelo nome da entidade.

Sintaxe:

```
element { style properties }
```

Exemplo:

```
<style>
  span {
    color: olive;
    background-color: yellow;
  }
</style>
```

Exemplo do slide 8 com estilos globais

```
<p>A minha mãe tem os olhos de cor <span>azeitona</span>.</p>
```

# Seletores CSS

## Seletores de classe

Num documento HTML, **os seletores de classe permitem configurar apenas um conjunto de elementos que pertençam a essa classe** e não todos os elementos de um determinado tipo.

Sintaxe:

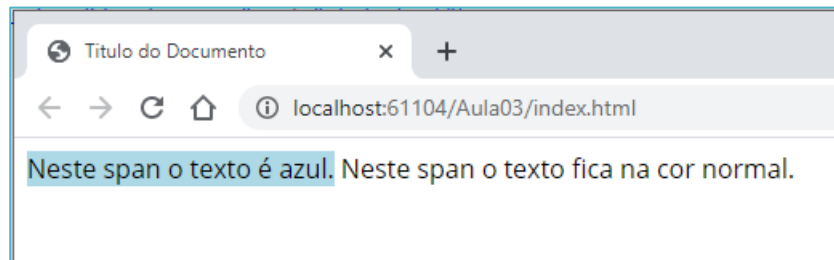
```
element.classname { style properties }
```

Exemplo:

```
span.inBlue { background-color: lightblue; }
```

```
<span class="inBlue">Neste span o texto é azul.</span>  
<span>Neste span o texto fica na cor normal.</span>
```

Resultado:



**Nota:** No exemplo anterior, quando se definiu um estilo global para o elemento `<span>`, todos os elementos desse tipo ficaram afetados por esse estilo; neste caso, usando uma **class**, apenas os elementos pertencentes a essa **class** foram afetados.

# Seletores CSS

## Seletores de classe

As classes nem sempre precisam de ficar associadas com um elemento html. Podem ser genéricas e aplicam-se a todos os elementos que possuam essa classe.

Sintaxe:

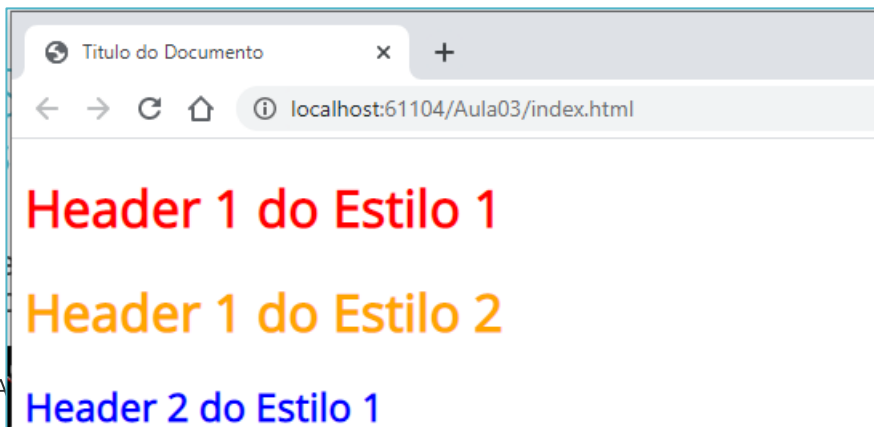
```
element.className { style properties }  
.classname { style properties }
```

Exemplo:

```
h1.estilo1 { color: red; }  
h1.estilo2 { color: orange; }  
.estilo1 { color: blue; }
```

```
<h1 class="estilo1">Header 1 do Estilo 1</h1>  
<h1 class="estilo2">Header 1 do Estilo 2</h1>  
<h2 class="estilo1">Header 2 do Estilo 1</h2>
```

Resultado:



**Nota importante:** na definição do estilo, o nome de uma classe (sem estar associada a uma entidade) começa **sempre** com um ponto (.)

# Seletores CSS

## Seletores de ID

Num documento HTML, os seletores de ID coincidem com nome de um elemento descrito no atributo ID.

O "ID de um elemento" deve ser único dentro num documento; assim, este estilo destina-se a ser aplicado apenas num elemento em particular.

Sintaxe:

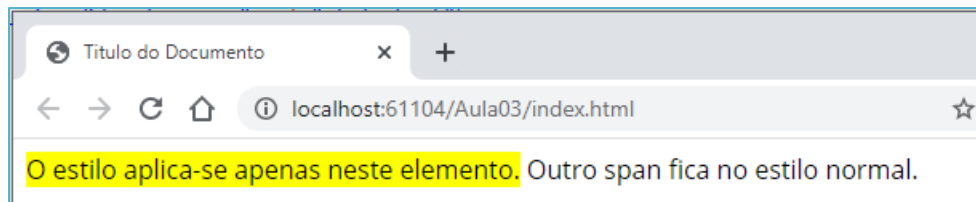
```
#id_value { style properties }
```

Exemplo:

```
#identified { background-color: yellow; }
```

```
<span id="identified">O estilo aplica-se apenas neste elemento.</span>  
<span>Outro span fica no estilo normal.</span>
```

Resultado:



**Nota importante:** o nome de um identificador, na definição do estilo, começa **sempre** com um cardinal (#). Os identificadores são sensíveis às maiúsculas e minúsculas. Assim, `identified` é diferente `Identified` de ou de `IDENTIFIED`.





# CSS

## Pseudo classes

Uma pseudo-classe CSS é uma palavra-chave precedida de dois pontos (:) que é adicionada no final dos seletores para especificar que se deseja modelar os elementos selecionados somente quando estes estiverem em determinado estado.

Sintaxe:

```
selector:pseudo-class { property:value; }
```

Exemplos: elemento a

```
a:link { color: #FF0000; } /* unvisited link */  
a:visited { color: #00FF00; } /* visited link */  
a:hover { color: #FF00FF; } /* mouse over link */  
a:active { color: #0000FF; } /* selected link */
```

# CSS

## Pseudo classes

Lista completa:

```
:active  
:any  
:checked  
:default  
:dir()  
:disabled  
:empty  
:enabled  
:first  
:first-child  
:first-of-type
```

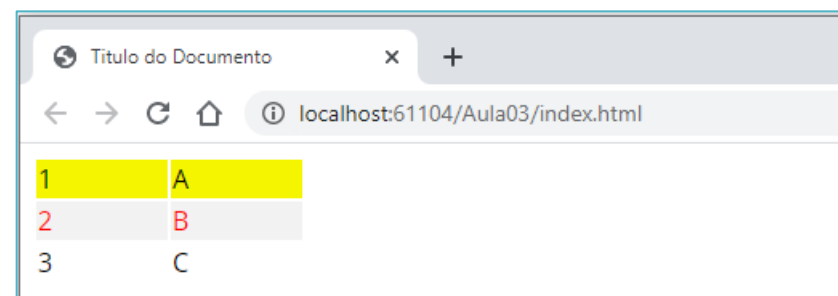
```
:fullscreen  
:focus  
:focus-within  
:hover  
:indeterminate  
:in-range  
:invalid  
:lang()  
:last-child  
:last-of-type
```

```
:left  
:link  
:not()  
:nth-child()  
:nth-last-child()  
:nth-last-of-type()  
:nth-of-type()  
:only-child  
:only-of-type  
:optional
```

```
:out-of-range  
:read-only  
:read-write  
:required  
:right  
:root  
:scope  
:target  
:valid  
:visited
```

Exemplos:

```
/* coloca as rows pares (even) com fundo cinzento */  
tr:nth-child(even) { background-color: #f2f2f2; color: red; }  
  
/* faz o efeito hover sobre as rows de uma tabela */  
tr:hover { background-color: #f5f500; }
```



Para uma consulta detalhada da aplicação e função de cada uma das pseudo classes, ver:

[https://developer.mozilla.org/en-US/docs/Learn/CSS/Introduction to CSS/Pseudo-classes and pseudo-elements](https://developer.mozilla.org/en-US/docs/Learn/CSS/Introduction%20to%20CSS/Pseudo-classes%20and%20pseudo-elements)

# CSS

## Pseudo elementos

Os pseudo-elementos são um tipo de CSS - desta vez precedidas por dois dois pontos (: :) - que se aplica somente a uma parte de um elemento, como por exemplo "a primeira letra de um elemento", ou "a primeira linha de um parágrafo" ou ainda "antes" ou "depois" de um elemento.

Sintaxe:

```
selector::pseudo-element { property:value; }
```

Lista completa:

```
::after  
::before  
::first-letter  
::first-line  
::selection  
::backdrop
```

Para uma consulta detalhada da aplicação e função de cada uma dos pseudo elementos, ver:

[https://developer.mozilla.org/en-US/docs/Learn/CSS/Introduction to CSS/Pseudo-classes and pseudo-elements](https://developer.mozilla.org/en-US/docs/Learn/CSS/Introduction%20to%20CSS/Pseudo-classes%20and%20pseudo-elements)

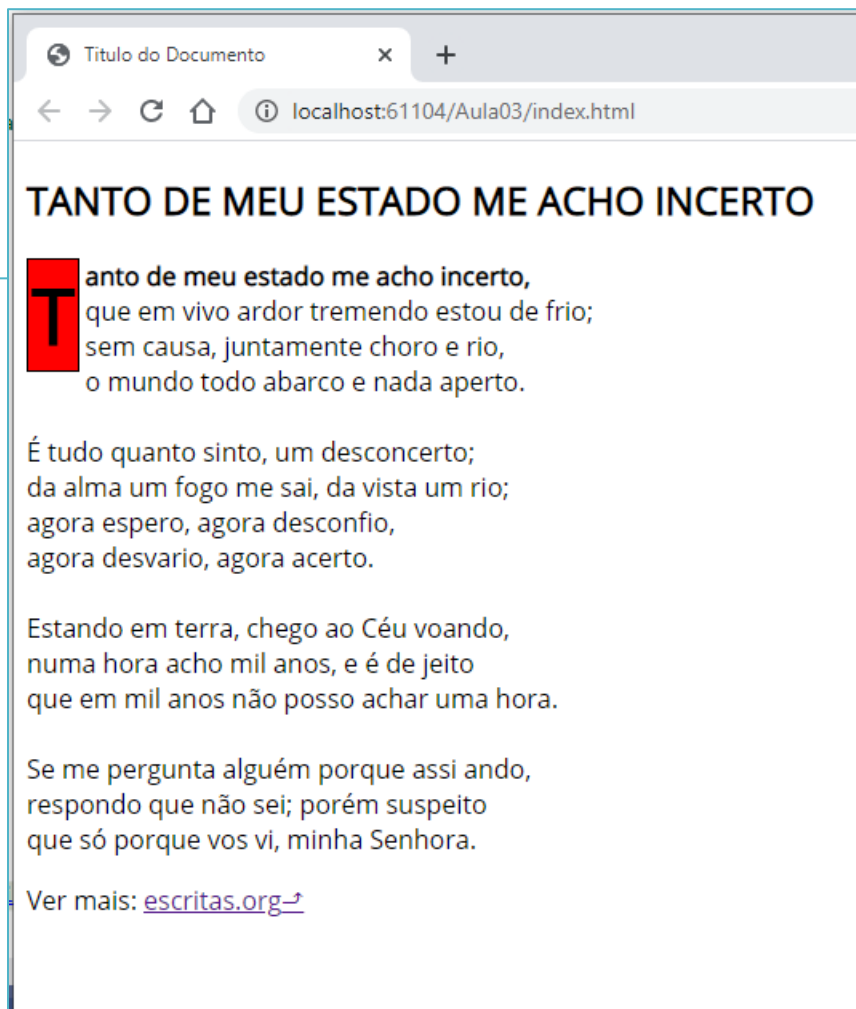
## Exemplo:

```
p.primeira::first-line { font-weight: bold; }  
p.primeira::first-letter { font-size: 3em; border: 1px solid black; background:  
red; display: block; float: left; padding: 2px; margin-right: 4px; }  
/* Todos os elementos com um atributo "href", cujo valor comece com "http", terão  
uma seta adicionada depois do seu conteúdo (indicação de ligação externa) */  
[href^=http]::after { content: '↗'; }
```

```
<h2>TANTO DE MEU ESTADO ME ACHO INCERTO</h2>  
<p class="primeira">  
  Tanto de meu estado me acho incerto,<br>  
  que em vivo ardor tremendo estou de frio;<br>  
  sem causa, juntamente choro e rio,<br>  
  o mundo todo abarco e nada aperto.<br>  
</p>  
<p>  
  É tudo quanto sinto, um desconcerto;<br>  
  da alma um fogo me sai, da vista um rio;<br>  
  agora espero, agora desconfio,<br>  
  agora desvario, agora acerto.<br>  
</p>  
<p>  
  Estando em terra, chego ao Céu voando,<br>  
  numa hora acho mil anos, e é de jeito<br>  
  que em mil anos não posso achar uma hora.<br>  
</p>  
<p>  
  Se me pergunta alguém porque assi ando,<br>  
  respondo que não sei; porém suspeito<br>  
  que só porque vos vi, minha Senhora.  
</p>  
<p>Ver mais: <a href="https://www.escritas.org/pt/t/9988/tanto-de-meu-estado-me-acho-incerto">escritas.org</a></p>
```

20/10/2020

©2014-20, JOAQUIM SOUSA PINTO



20

# Seletores CSS

## Seletor adjacente

O CSS para um “seletor adjacente” ou “próximo selector irmão” irá abranger apenas o elemento especificado que segue imediatamente o antigo elemento especificado.

Sintaxe:

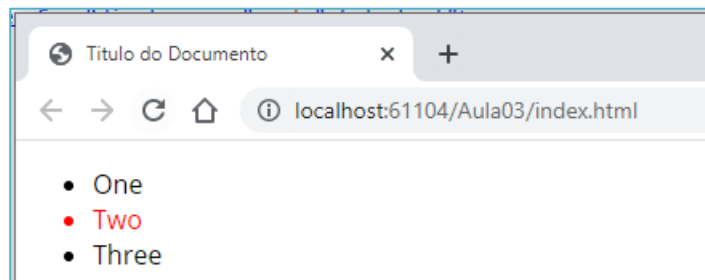
```
elemento_anterior + element_desejado { style properties }
```

Exemplo:

```
li:first-of-type + li { color: red; }
```

```
<ul>
  <li>One</li>
  <li>Two</li>
  <li>Three</li>
</ul>
```

Resultado:



# Seletores CSS

## Seletor irmão

O combinador “~” separa dois seletores e aplica-se ao segundo elemento somente se ele é precedido pelo primeiro, e se ambos compartilham um pai comum.

Sintaxe:

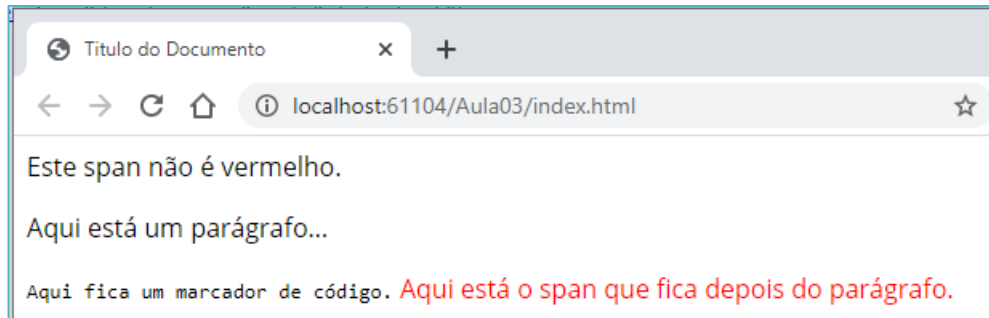
```
element ~ element { style properties }
```

Exemplo:

```
p ~ span { color: red; }
```

```
<span>Este span não é vermelho.</span>  
<p>Aqui está um parágrafo...</p>  
<code>Aqui fica um marcador de código.</code>  
<span>Aqui está o span que fica depois do parágrafo.</span>
```

Resultado:



# Seletores CSS

## Seletor filho

O combinador ">" separa dois seletores e corresponde apenas aos elementos que coincidem com o segundo selector que são filhos diretos de elementos que coincidem com o primeiro.

Sintaxe:

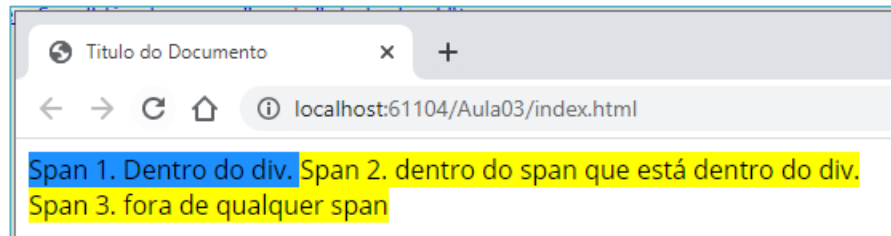
```
selector1 > selector2 { style properties }
```

Exemplo:

```
span { background-color: yellow; }  
div > span { background-color: DodgerBlue; }
```

```
<div>  
  <span>Span 1. Dentro do div.  
    <span>Span 2. dentro do span que está dentro do div.</span>  
  </span>  
</div>  
<span>Span 3. fora de qualquer span</span>
```

Resultado:



# Seletores CSS

## Seletor descendente (filho, neto, bisneto, ...)

O elemento de combinação " " (espaço) combina dois seletores de modo a que o selector combinado corresponde apenas aos elementos correspondentes ao segundo seletor para o qual há um elemento ancestral combinando o primeiro selector.

Seletores descendentes são semelhantes aos seletores filho, mas não exigem que a relação entre os elementos emparelhados seja estritamente pai-filho.

Sintaxe:

```
selector1 selector2 { style properties }
```

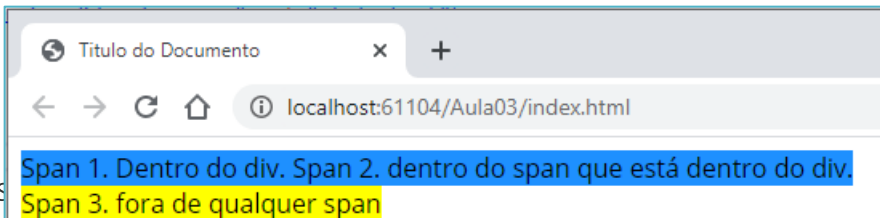
Exemplo:

```
span { background-color: yellow; }  
div span { background-color: DodgerBlue; }
```

```
<div>  
  <span>Span 1. Dentro do div. </span>  
  <span>Span 2. dentro do span que está dentro do div.</span>  
</span>  
</div>  
<span>Span 3. fora de qualquer span</span>
```

Seletor filho

Seletor neto



Note-se que neste estilo, ao contrário do definido no slide anterior, o `<span>` dentro do `<span>` (seletor neto) também é atingido pela formatação!



# Seletores CSS

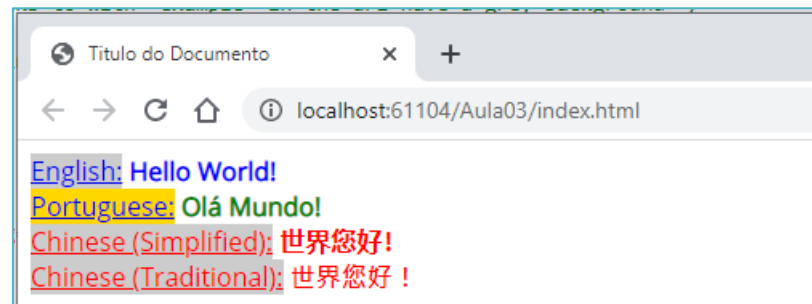
## Seletores de atributos

```
span[lang] { font-weight:bold; } /* All spans with a "lang" attribute are bold */
span[lang="pt"] { color:green; } /* All spans in Portuguese are green */
span[lang~="en-us"] { color: blue; } /* All spans in US English are blue */
span[lang|="zh"] { color: red; } /* Any span in Chinese is red, matches simplified (zh-CN) or traditional (zh-TW) */

a[href^="#"] { background-color:gold } /* All internal links have a gold background */
a[href$=".cn"] { color: red; } /* All links to urls ending in ".cn" are red */
a[href*="example"] { background-color: #CCCCCC; } /* All links to with "example" in the url have a grey background */
```

```
<a href="http://example.com">English:</a> <span lang="en-us en-gb en-au en-nz">Hello World!</span><br />
<a href="#portuguese">Portuguese:</a> <span lang="pt">Olá Mundo!</span><br />
<a href="http://example.cn">Chinese (Simplified):</a> <span lang="zh-CN">世界您好!</span><br />
<a href="http://example.cn">Chinese (Traditional):</a> <span lang="zh-TW">世界您好!</span><br />
```

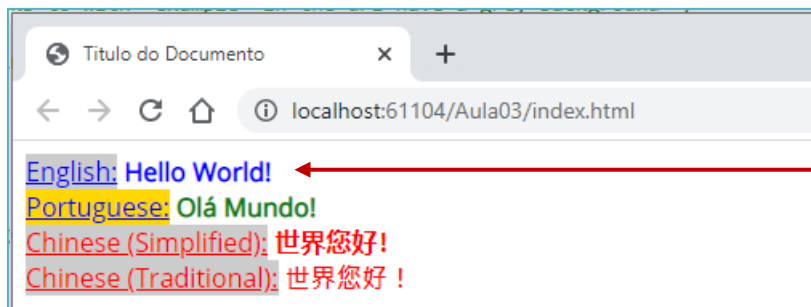
Fonte: [https://developer.mozilla.org/uk/docs/Web/CSS/Attribute\\_selectors](https://developer.mozilla.org/uk/docs/Web/CSS/Attribute_selectors)



```

<a href="http://example.com">English:</a> <span lang="en-us en-gb en-au en-nz">Hello World!</span><br />
<a href="#portuguese">Portuguese:</a> <span lang="pt">Olá Mundo!</span><br />
<a href="http://example.cn">Chinese (Simplified):</a> <span lang="zh-CN">世界您好 !</span><br />
<a href="http://example.cn">Chinese (Traditional):</a> <span lang="zh-TW">世界您好 !</span><br />

```



```

span[lang] { font-weight:bold; } /* All spans with a "lang" attribute are bold */
span[lang="pt"] { color:green; } /* All spans in Portuguese are green */
span[lang~="en-us"] { color: blue; } /* All spans in US English are blue */
span[lang|="zh"] { color: red; } /* Any span in Chinese is red, matches simplified (zh-CN) or traditional (zh-TW) */

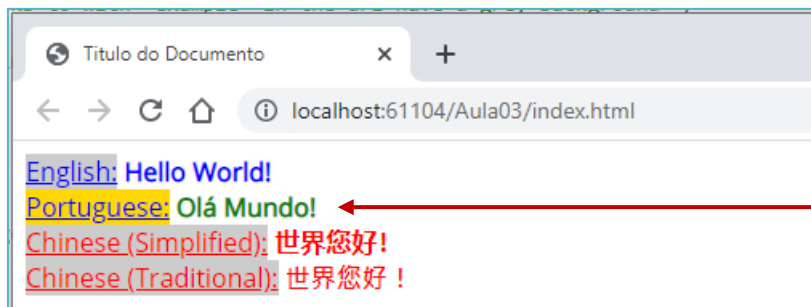
a[href^="#"] { background-color:gold } /* All internal links have a gold background */
a[href$=".cn"] { color: red; } /* All links to urls ending in ".cn" are red */
a[href*="example"] { background-color: #CCCCCC; } /* All links to with "example" in the url have a grey background */

```

```

<a href="http://example.com">English:</a> <span lang="en-us en-gb en-au en-nz">Hello World!</span><br />
<a href="#portuguese">Portuguese:</a> <span lang="pt">Olá Mundo!</span><br />
<a href="http://example.cn">Chinese (Simplified):</a> <span lang="zh-CN">世界您好 !</span><br />
<a href="http://example.cn">Chinese (Traditional):</a> <span lang="zh-TW">世界您好 !</span><br />

```



```

span[lang] { font-weight:bold; } /* All spans with a "lang" attribute are bold */
span[lang="pt"] { color:green; } /* All spans in Portuguese are green */
span[lang~="en-us"] { color: blue; } /* All spans in US English are blue */
span[lang|="zh"] { color: red; } /* Any span in Chinese is red, matches simplified (zh-CN) or traditional (zh-TW) */

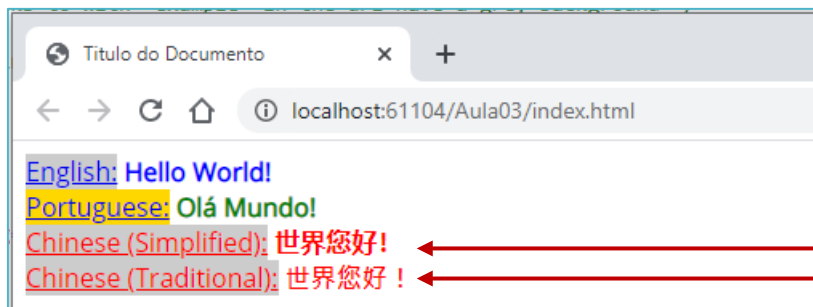
a[href^="#"] { background-color:gold } /* All internal links have a gold background */
a[href$=".cn"] { color: red; } /* All links to urls ending in ".cn" are red */
a[href*="example"] { background-color: #CCCCCC; } /* All links to with "example" in the url have a grey background */

```

```

<a href="http://example.com">English:</a> <span lang="en-us en-gb en-au en-nz">Hello World!</span><br />
<a href="#portuguese">Portuguese:</a> <span lang="pt">Olá Mundo!</span><br />
<a href="http://example.cn">Chinese (Simplified):</a> <span lang="zh-CN">世界您好 !</span><br />
<a href="http://example.cn">Chinese (Traditional):</a> <span lang="zh-TW">世界您好 !</span><br />

```



```

span[lang] { font-weight:bold; } /* All spans with a "lang" attribute are bold */
span[lang="pt"] { color:green; } /* All spans in Portuguese are green */
span[lang~="en-us"] { color: blue; } /* All spans in US English are blue */
span[lang|="zh"] { color: red; } /* Any span in Chinese is red, matches simplified (zh-CN) or traditional (zh-TW) */

a[href^="#"] { background-color:gold } /* All internal links have a gold background */
a[href$=".cn"] { color: red; } /* All links to urls ending in ".cn" are red */
a[href*="example"] { background-color: #CCCCCC; } /* All links to with "example" in the url have a grey background */

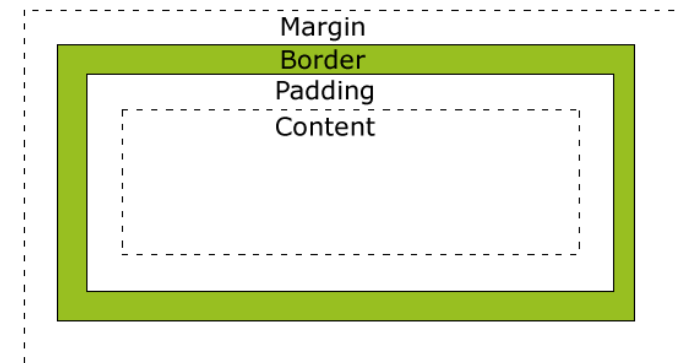
```

# Elementos CSS

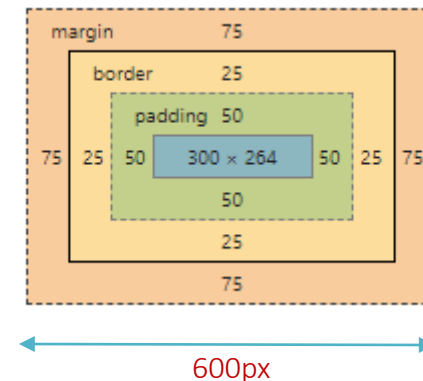
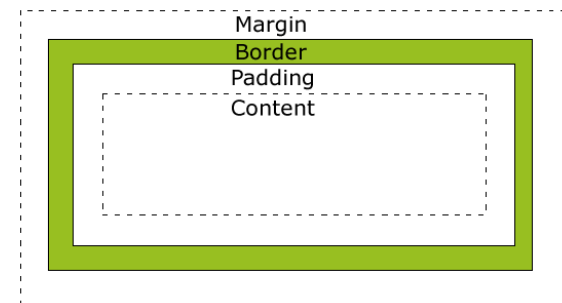
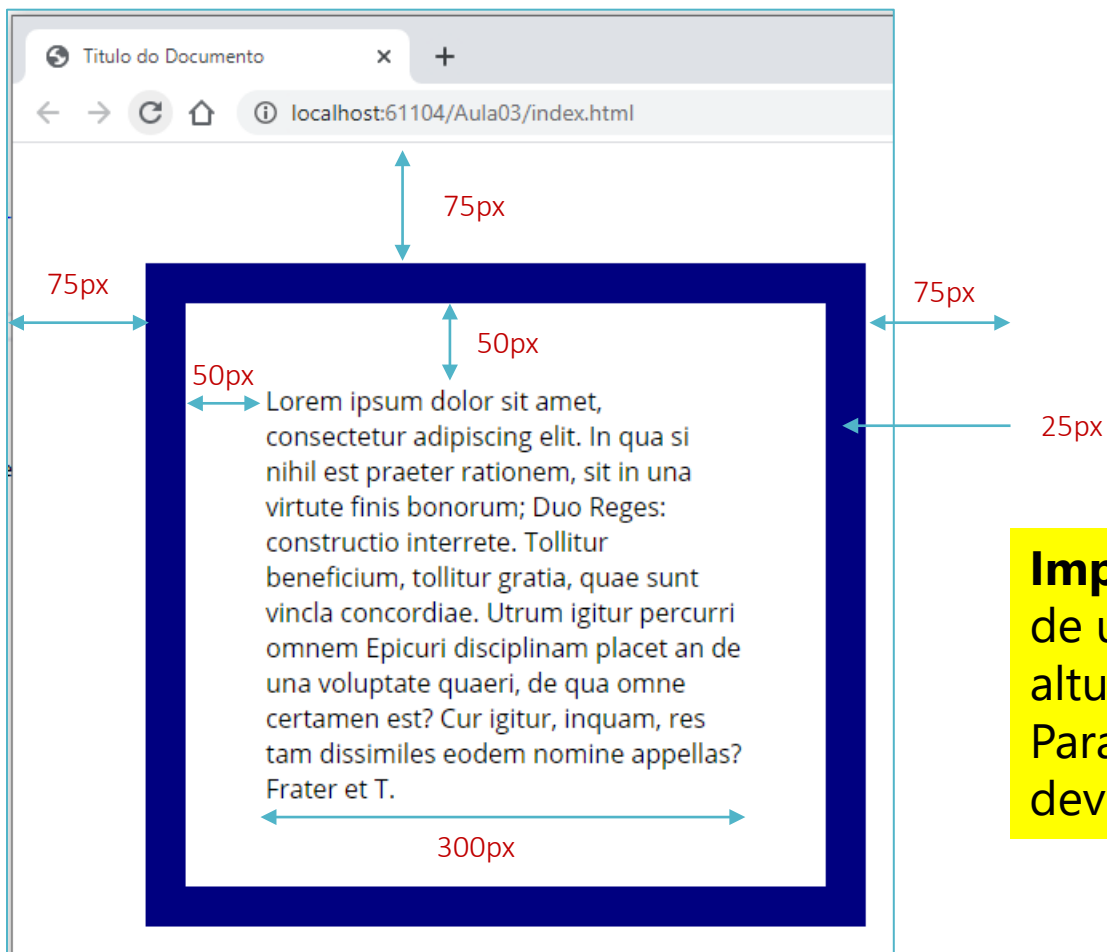
Todos os elementos HTML podem ser considerados como um conjunto de “caixas” ou como uma “cebola” que possui 4 camadas: Content, Padding, Border, Margin;

Cada uma das camadas pode ser configurada individualmente.

Content	conteúdo de um elemento de texto ou imagem
Padding	espaçamento entre o conteúdo e a bordadura
Border	bordadura à volta do elemento
Margin	distância entre o border e o limite exterior do elemento



```
p { width: 300px; padding: 50px; border: solid 25px navy; margin: 75px; }
```



**Importante:** ao definir as propriedades de largura e altura de um elemento com CSS, apenas define a largura e a altura da área de conteúdo / *content*. Para calcular o tamanho total de um elemento, também deve adicionar preenchimento, bordas e margens.

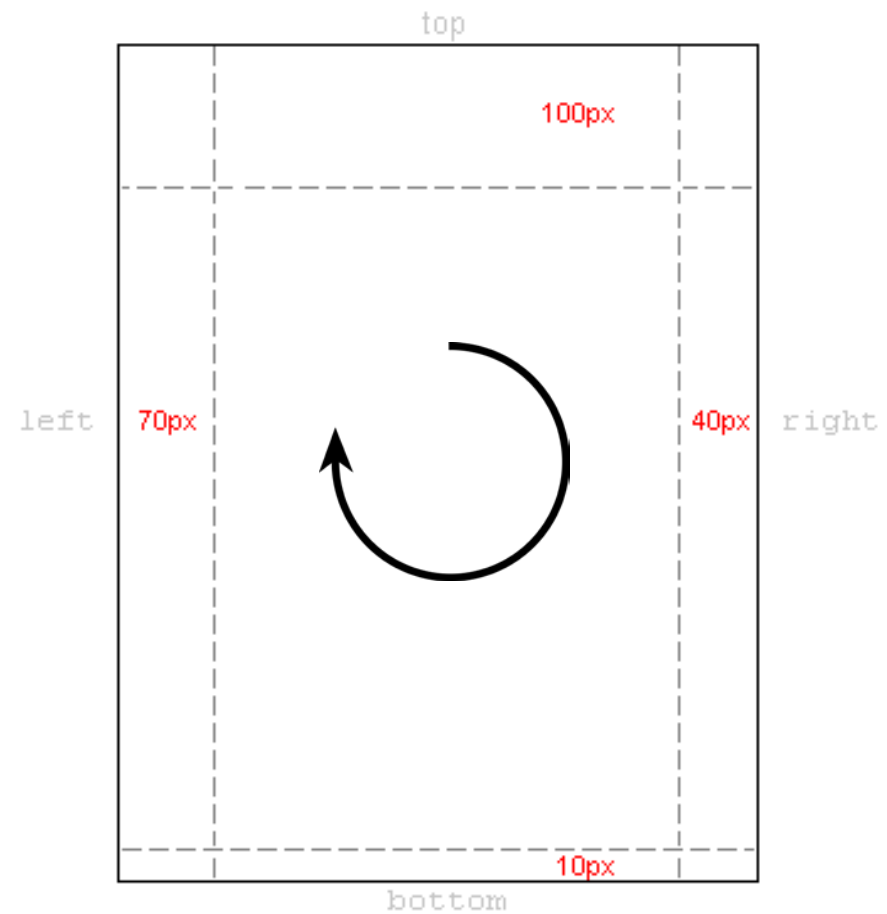
# Margin

Seleção individual de cada uma das variáveis

```
body {  
  margin-top: 100px;  
  margin-right: 40px;  
  margin-bottom: 10px;  
  margin-left: 70px;  
}
```

ou, na forma compacta,

```
body { margin: 100px 40px 10px 70px; }
```



# Padding

Seleção individual de cada uma das variáveis

```
p {  
  padding-top: 20px;  
  padding-right: 30px;  
  padding-bottom: 20px;  
  padding-left: 30px;  
}
```

ou, na forma compacta,

```
p { padding: 20px 30px 20px 30px; }
```



# Border

Seleção individual de cada uma das variáveis

```
1.border
2.border-block-end
3.border-block-end-color
4.border-block-end-style
5.border-block-end-width
6.border-block-start
7.border-block-start-color
8.border-block-start-style
9.border-block-start-width
10.border-bottom
11.border-bottom-color
12.border-bottom-left-radius
13.border-bottom-right-radius
14.border-bottom-style
15.border-bottom-width
16.border-collapse
17.border-color
```

```
18.border-image
19.border-image-outset
20.border-image-repeat
21.border-image-slice
22.border-image-source
23.border-image-width
24.border-inline-end
25.border-inline-end-color
26.border-inline-end-style
27.border-inline-end-width
28.border-inline-start
29.border-inline-start-color
30.border-inline-start-style
31.border-inline-start-width
32.border-left
33.border-left-color
34.border-left-style
```

```
35.border-left-width
36.border-radius
37.border-right
38.border-right-color
39.border-right-style
40.border-right-width
41.border-spacing
42.border-style
43.border-top
44.border-top-color
45.border-top-left-radius
46.border-top-right-radius
47.border-top-style
48.border-top-width
49.border-width
```

# Border

Seleção individual de cada uma das variáveis

```
border-style: none|hidden|dotted|dashed|solid|double|groove|ridge|inset|outset;
```

```
border-spacing: Length;
```

```
border-width: medium|thin|thick|Length;
```

```
border-collapse: separate|collapse;
```

Exemplos:

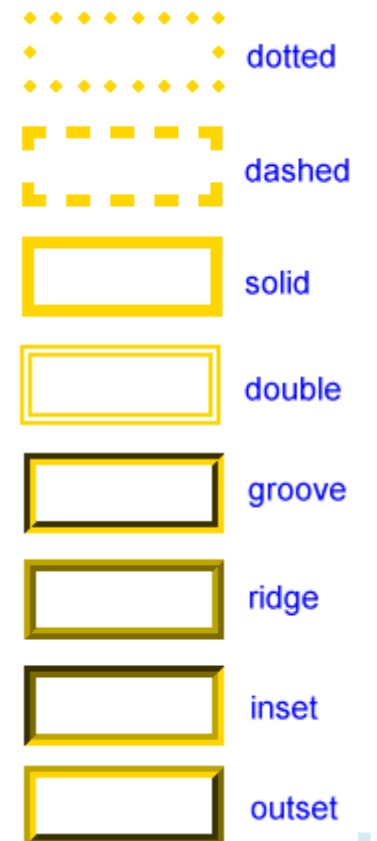
```
h1 { border-width: thick; border-style: dotted; border-color: gold; }  
h2 { border-width: 20px; border-style: outset; border-color: red; }  
p { border-width: 1px; border-style: dashed; border-color: blue; }
```

ou , na forma compacta,

```
border: border-width border-style border-color|initial|inherit;
```

```
h1 { thick dotted gold; }  
h2 { 20px outset red; }  
p { 1px dashed blue; }
```

Tipos de estilos de borders:



# Fontes

## Generalidades

Nas CSS, existem dois tipos de nomes de família de fontes:

família genérica - um grupo de famílias de fontes com uma aparência semelhante (como "Serif", "SansSerif" ou "Monospace")

família de fontes - uma família de fontes específica (como "Times New Roman" ou "Arial")



As fontes serifadas têm pequenas linhas nas extremidades de alguns caracteres (Times, Geneva, ...)

"Sans" significa sem - essas fontes não têm linhas no final dos caracteres (Arial, Verdana, ...)

Todos os caracteres monoespaçados têm a mesma largura (Courier, Lucida Console)

# Estilos para utilização de fontes

```
1.@font-face
2.font-family
3.font-feature-settings
4.@font-feature-values
5.font-kerning
6.font-language-override
7.font-size
8.font-size-adjust
9.font-stretch
10.font-style
11.font-synthesis
12.font-variant
13.font-variant-alternates
14.font-variant-caps
15.font-variant-east-Asian
16.font-variant-ligatures
17.font-variant-numeric
18.font-variant-position
19.font-weight
```

# Fontes

Versão compacta:

```
font: font-style font-variant font-weight font-size/line-height font-family|caption|icon|menu|message-box|small-caption|status-bar;
```

```
font-style: normal|italic|oblique|initial|inherit;
```

```
font-variant: normal|small-caps;
```

```
font-weight: normal|bold|bolder|lighter|number;
```

```
font-size:medium|xx-small|x-small|small|large|x-large|xx-large|smaller|larger|length;
```

```
font-family: family-name|generic-family|initial|inherit;
```

# Estilos para alinhamento de texto

- 1. text-align
- 2. text-align-last
- 3. text-orientation
- 4. text-decoration
- 5. text-decoration-color
- 6. text-decoration-line
- 7. text-decoration-style
- 8. text-indent
- 9. text-orientation
- 10. text-overflow
- 11. text-rendering
- 12. text-shadow
- 13. text-transform
- 14. text-underline-position

```
text-align: left|right|center|justify;
```

```
text-overflow: clip|ellipsis|string;
```

```
text-transform: none|capitalize|uppercase|lowercase;
```

```
text-justify: auto|inter-word|inter-character|none;
```

# CSS

## Posicionamento de elementos

Há 2 tipos principais de posicionamento:

absoluto – os pontos referência são a página;

relativo – os pontos referência são o elemento pai dentro do qual o elemento está.

```
position: static|absolute|fixed|relative|sticky;
```

```
.pos_abs { position: absolute; top: 50px; left: 50px; }
```

```
.pos_rel { position: relative; top: 50px; left: 50px; }
```

Outra forma de posicionamento relativo é conseguido através do estilo float:

```
float: none|left|right;
```

Depois de “flutuar elementos”, de modo a reiniciar o alinhamento, utiliza-se o estilo clear.

```
clear: none|left|right|both;
```

```
<div style="width: 50%; float: left; clear: left;">  
</div>
```

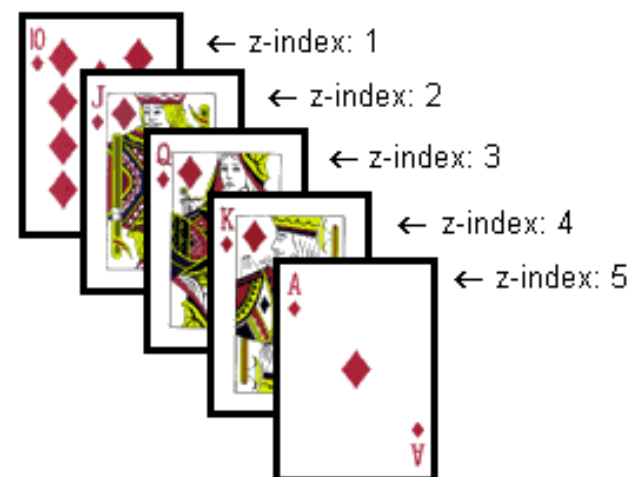
# CSS

## Posicionamento de elementos

```
z-index: auto|number|initial|inherit;
```

Exemplo:

```
.pos_1 {  
  position: absolute;  
  top: 50px;  
  left: 50px;  
  z-index: 1;  
}  
.pos_2 {  
  position: absolute;  
  top: 100px; left: 100px; z-index: 2;  
}  
.pos_3 {  
  position: absolute;  
  top: 150px;  
  left: 150px;  
  z-index: 3;  
}
```





# CSS fontes de ícones

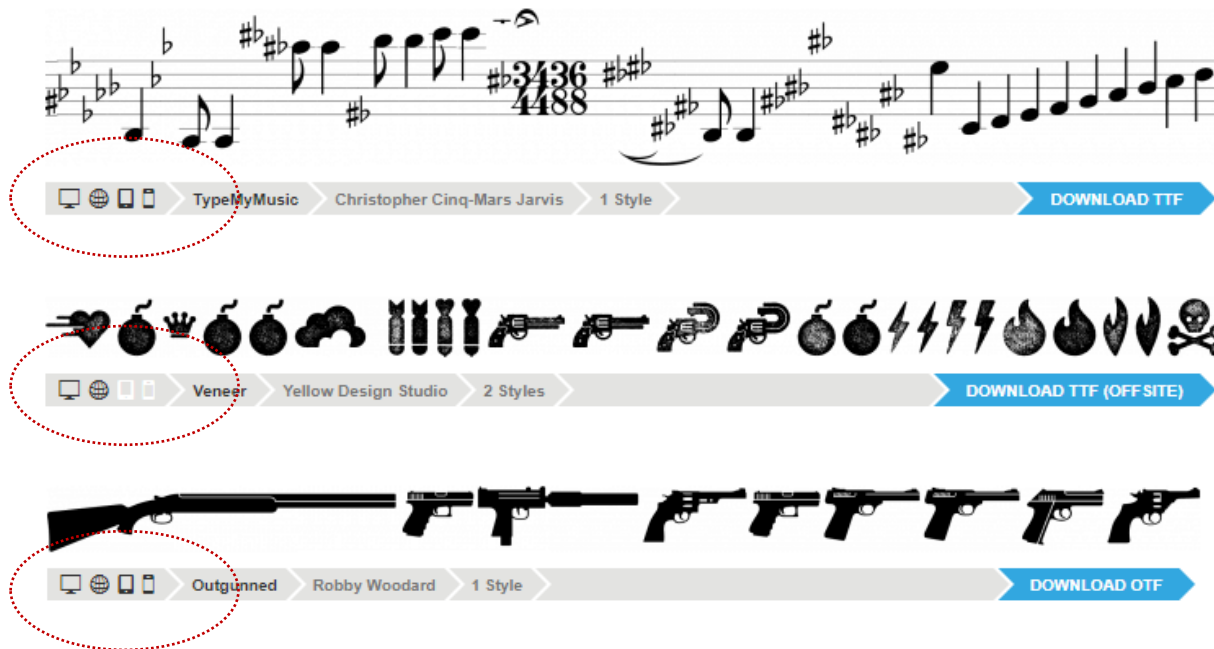


# CSS Icons

Squirrel (<https://www.fontsquirrel.com>)

Tal como referido na aula anterior, é possível a utilização de fontes externas para utilização nas páginas Web. Para além das fontes de texto, existem também fontes com imagens/icons.

Exemplos: [<https://www.fontsquirrel.com/fonts/list/classification/dingbat>]



# CSS Icons

Font Awesome (<http://fontawesome.com/>)

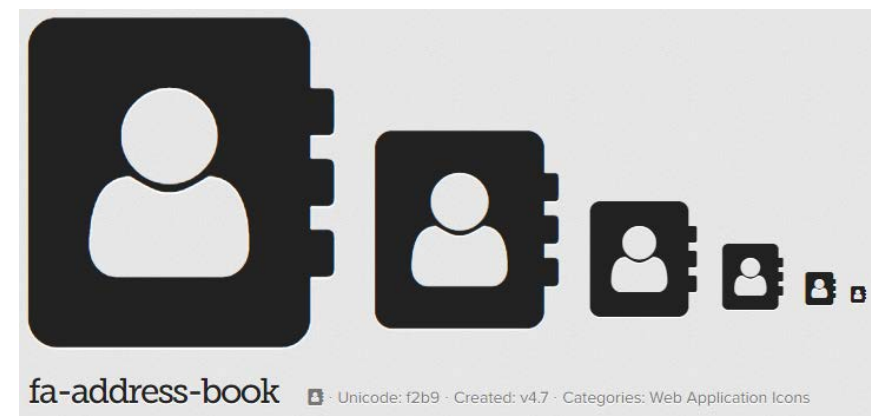
Uma das fontes de ícones mais utilizada é a "font awesome". A versão 6, em fase final de desenvolvimento (out20), terá 1.609 ícones (vetoriais) grátis acessíveis através de um ficheiro css.

Instalação através de endereço público da versão 4.7.0:

```
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css">
```

Exemplo:

```
<i class="fa fa-address-book" aria-hidden="true"></i>
```



```

<!DOCTYPE html>
<html>
<head>
  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css">
  <link href="https://fonts.googleapis.com/css?family=Roboto:300" rel="stylesheet">
  <style>
    body { font-family: 'Roboto', sans-serif; }
    .text-danger { color: red; }
  </style>
</head>
<body>
  <h2>Basic Icons</h2>
  <i class="fa fa-heart"></i>
  <i class="fa fa-car"></i>
  <i class="fa fa-file"></i>
  <i class="fa fa-bars"></i>
  <h4>Larger Icons</h4>
  <i class="fa fa-2x fa-cloud"></i>
  <h2>List Icons</h2>
  <ul class="fa-ul">
    <li><i class="fa-li fa fa-check-square"></i>List icons</li>
    <li><i class="fa-li fa fa-check-square"></i>can be used</li>
    <li><i class="fa-li fa fa-spinner fa-spin"></i>as bullets</li>
    <li><i class="fa-li fa fa-square"></i>in lists</li>
  </ul>
  <h2>Animated Icons</h2>
  <i class="fa fa-spinner fa-pulse fa-3x fa-fw"></i>
  <h2>Rotated & Flipped</h2>
  <i class="fa fa-shield"></i> normal<br>
  <i class="fa fa-shield fa-rotate-90"></i> fa-rotate-90<br>
  <i class="fa fa-shield fa-rotate-180"></i> fa-rotate-180<br>
  <i class="fa fa-shield fa-rotate-270"></i> fa-rotate-270<br>
  <i class="fa fa-shield fa-flip-horizontal"></i> fa-flip-horizontal<br>
  <i class="fa fa-shield fa-flip-vertical"></i> fa-flip-vertical
  <h2>Stacked Icons</h2>
  <span class="fa-stack fa-lg">
    <i class="fa fa-camera fa-stack-1x"></i>
    <i class="fa fa-ban fa-stack-2x text-danger"></i>
  </span> fa-ban on fa-camera
</body>
</html>

```

Permite a criação de novos elementos  
para representação de listas

Elementos animados  
(com movimento)

Rotação dos ícones

Sobreposição de ícones

## Icons



## Larger Icons



## List Icons

- ☒ List icons
- ☒ can be used
- as bullets
- in lists

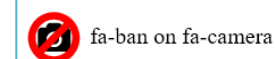
## Animated Icons



## Rotated & Flipped

- normal
- fa-rotate-90
- fa-rotate-180
- fa-rotate-270
- fa-flip-horizontal
- fa-flip-vertical

## Stacked Icons



# CSS Icons

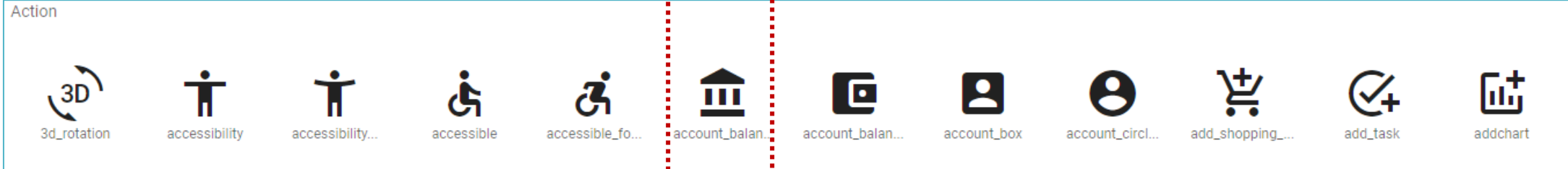
Google fonts (<https://material.io/resources/icons/?style=baseline>)

Esta fonte da google possui vários temas: filled, outlined, rounded, two-tone e sharp.

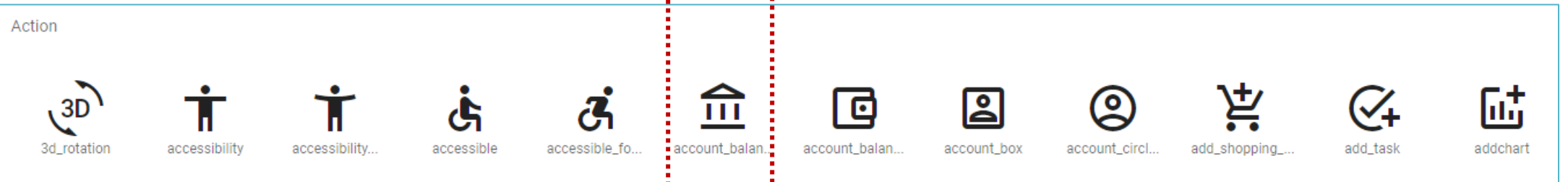
```
<!DOCTYPE html>
<html>
<head>
  <link rel="stylesheet" href="https://fonts.googleapis.com/icon?family=Material+Icons">
</head>
<body>
  <i class="material-icons">cloud</i>
  <i class="material-icons">favorite</i>
  <i class="material-icons">attachment</i>
  <i class="material-icons">computer</i>
  <i class="material-icons">traffic</i>
</body>
</html>
```

O modo de aplicação da fonte é distinto do utilizado na font awesome, na sua versão por classes.

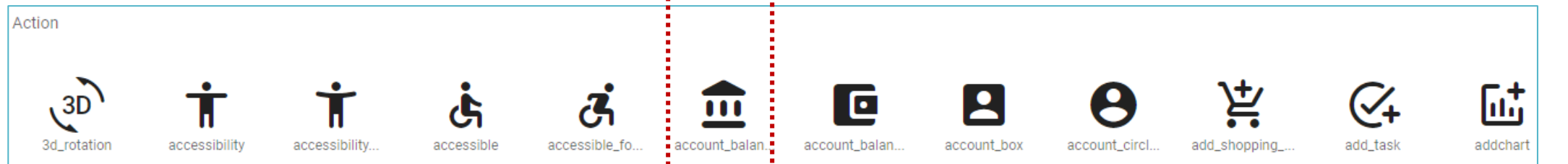
## Filled



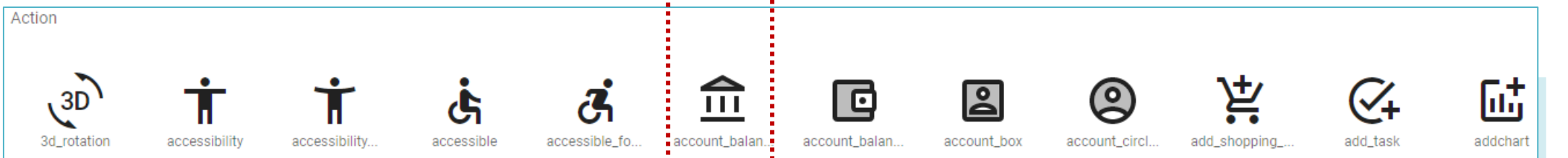
## Outlined



## Rounded



## Two-Tone



```

/* Google Material Icons Font
----- */
.material-icons-outlined,
.material-icons.material-icons--outlined,
.material-icons-two-tone,
.material-icons.material-icons--two-tone,
.material-icons-round,
.material-icons.material-icons--round,
.material-icons-sharp,
.material-icons.material-icons--sharp {
  font-family: 'Material Icons';
  font-weight: normal;
  font-style: normal;
  font-size: 24px; /* Preferred icon size */
  display: inline-block;
  line-height: 1;
  text-transform: none;
  letter-spacing: normal;
  word-wrap: normal;
  white-space: nowrap;
  direction: ltr;

  /* Support for all WebKit browsers. */
  -webkit-font-smoothing: antialiased;

  /* Support for Safari and Chrome. */
  text-rendering: optimizeLegibility;

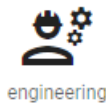
  /* Support for Firefox. */
  -moz-osx-font-smoothing: grayscale;

  /* Support for IE. */
  font-feature-settings: 'liga';
}

```

Exemplo:

```
<i class="material-icons-outlined md-18">engineering</i></a>
```



```

.material-icons-outlined,
.material-icons.material-icons--outlined {
  font-family: 'Material Icons Outlined';
}

.material-icons-two-tone,
.material-icons.material-icons--two-tone {
  font-family: 'Material Icons Two Tone';
}

.material-icons-round,
.material-icons.material-icons--round {
  font-family: 'Material Icons Round';
}

/* Rules for sizing the icon. */
.material-icons-outlined.md-18,
.material-icons.md-18.material-icons--outlined,
.material-icons-two-tone.md-18,
.material-icons.md-18.material-icons--two-tone,
.material-icons-round.md-18,
.material-icons.md-18.material-icons--round,
.material-icons.md-18 {
  font-size: 18px;
}

/* Rules for using icons as black on a light background. */
.material-icons.md-dark {
  color: rgba(0, 0, 0, 0.54);
}

.material-icons.md-dark.md-inactive {
  color: rgba(0, 0, 0, 0.26);
}

/* Rules for using icons as white on a dark background. */
.material-icons.md-light {
  color: rgba(255, 255, 255, 1);
}

.material-icons.md-light.md-inactive {
  color: rgba(255, 255, 255, 0.3);
}

```

# Utilizando CSS's para cenários mais complexos

Demonstração da separação entre HTML e CSS

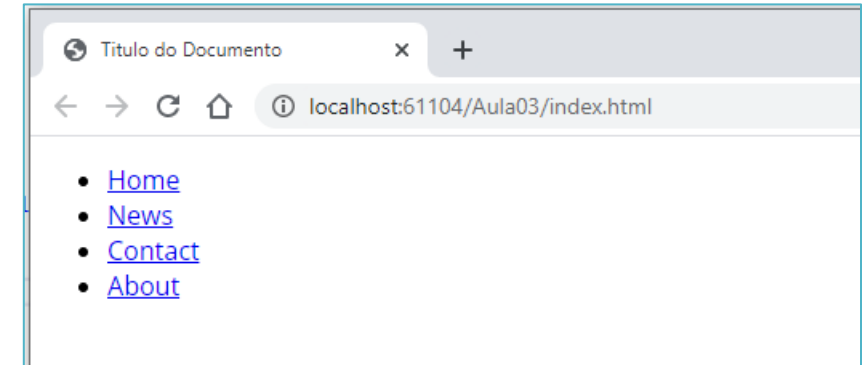
[Síntese do que foi aprendido nas 3 aulas]



# Barras de navegação utilizando listas

o HTML básico

```
<!DOCTYPE html>
<html>
<head>
</head>
<body>
  <ul>
    <li><a href="home.html" class="active">Home</a></li>
    <li><a href="news.html">News</a></li>
    <li><a href="contact.html">Contact</a></li>
    <li><a href="about.html">About</a></li>
  </ul>
</body>
</html>
```

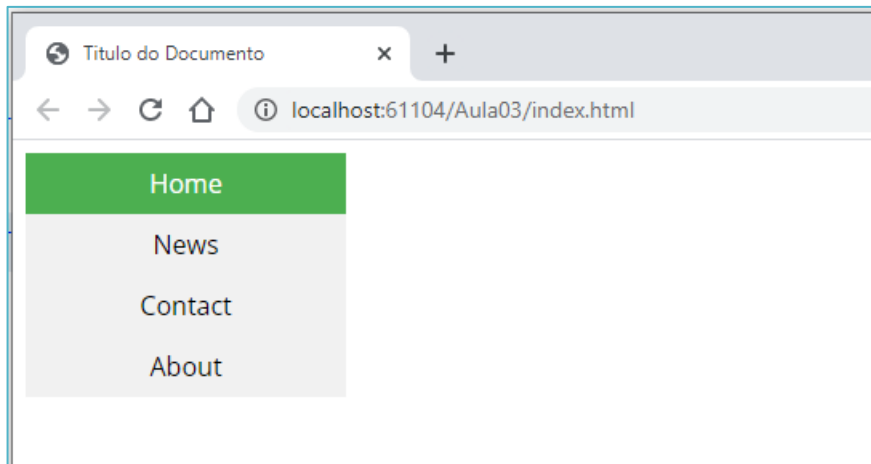


# Barras de navegação utilizando listas

## Menu vertical

Alterando os estilos CSS de modo a afetar a representação dos elementos `<ul>` e `<li>`:

```
ul { list-style-type: none; margin: 0; padding: 0; width: 200px; background-color: #f1f1f1; }  
li a { display: block; color: #000; padding: 8px 16px; text-decoration: none; text-align: center }  
/* Change the link color on hover */  
li a:hover { background-color: #555; color: white; }  
.active { background-color: #4CAF50; color: white; }
```

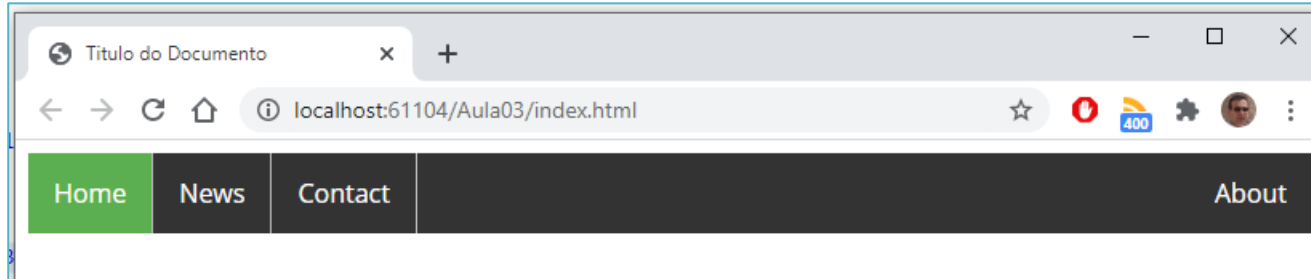


# Barras de navegação utilizando listas

## Menu horizontal

Ou ainda ....

```
ul { list-style-type: none; margin: 0; padding: 0; overflow: hidden; background-color: #333; }
li { float: left; border-right: 1px solid #bbb; }
    li:last-child { border-right: none; float: right; }
    li a { display: block; color: white; text-align: center; padding: 14px 16px; text-decoration: none; }
        li a:hover:not(.active) { background-color: #444; }
        li a.active:hover { color: #444; }
.active { background-color: #5CAF50; }
```



# Modelando imagens ao dispositivo

## A imagem

Imagem com 1800px por 900px (<https://www.pngonly.com/wp-content/uploads/2018/11/hd-avengers-png-10203.png>) (524Kb)



# Modelando imagens ao dispositivo

## Exemplo

O CSS:

```
<style>
  .banner { height: 600px; margin: auto;
             background-image: url('avengers.png');
             background-position: center center;
             background-repeat: no-repeat;
             background-size: cover;
          }
</style>
```

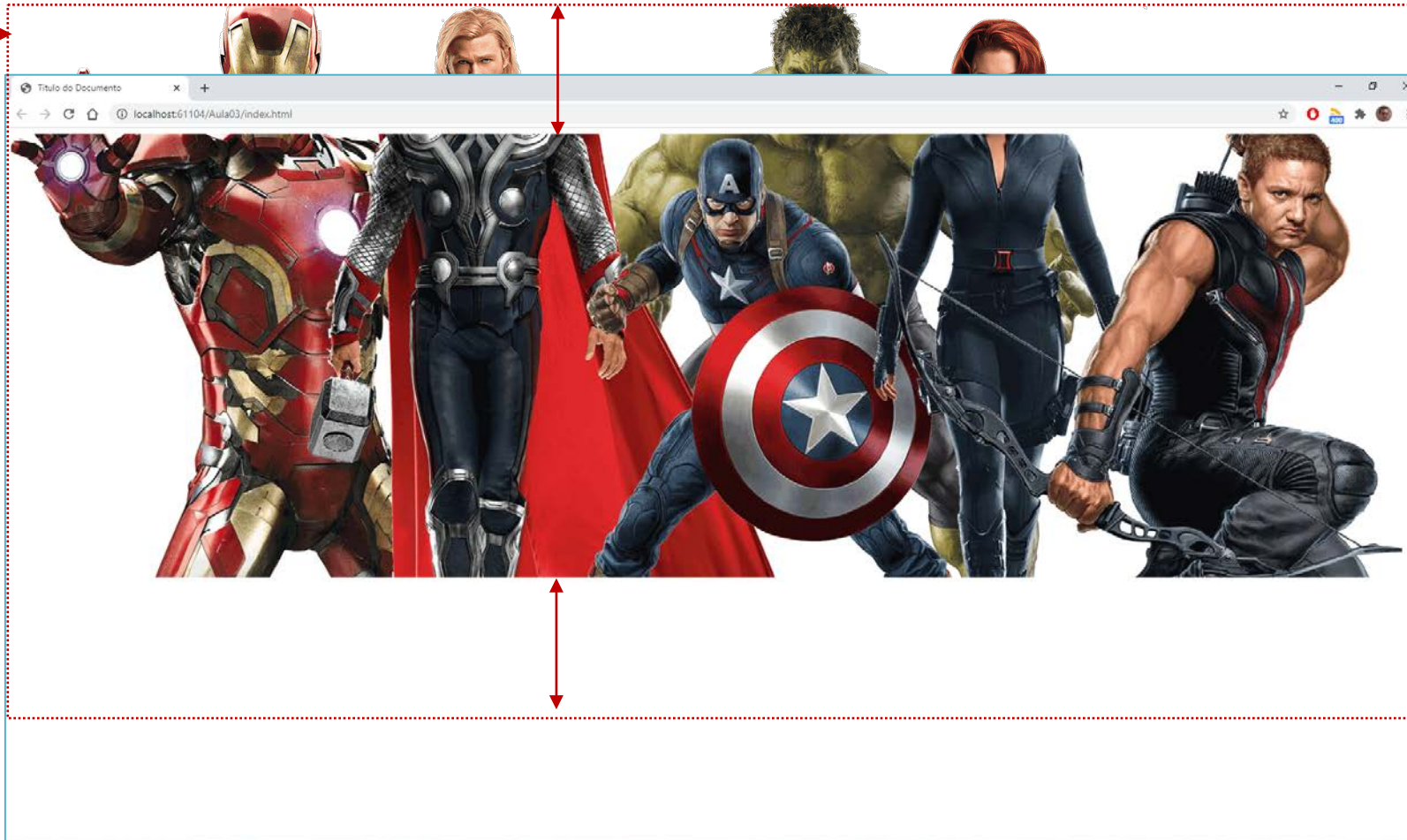
O HTML:

```
<div class="banner"></div>
```

# Modelando imagens ao dispositivo

dispositivo mais largo que a imagem – imagem é alargada e cortada em cima e em baixo

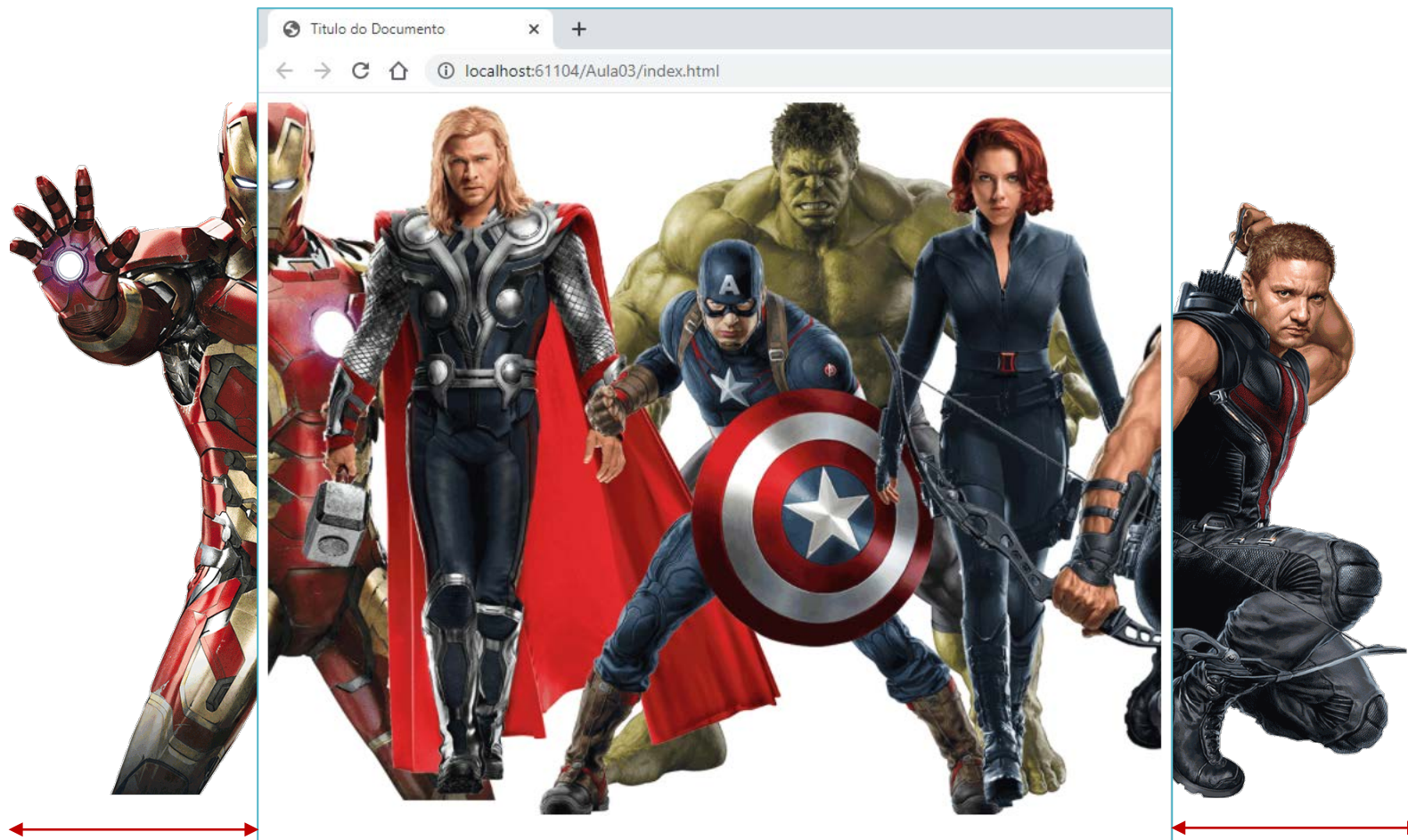
Imagem original





# Modelando imagens ao dispositivo

dispositivo mais estreito que a imagem – imagem é redimensionada e cortada nas pontas



# Modelando imagens ao dispositivo

dispositivo mais estreito que a imagem – imagem é redimensionada e cortada nas pontas

