

Desenvolvimento de Páginas Web Estáticas

HTML, HTML5, *Web Responsive*, *Bootstrap*, *CSS*



Programação e Serviços Web
1º Ano 2º Semestre

Cofinanciado por:



UNIÃO EUROPEIA
Fundo Social Europeu

Desenvolvimento de Páginas Web Estáticas

CSS

CSS – *Cascade Style Sheet*

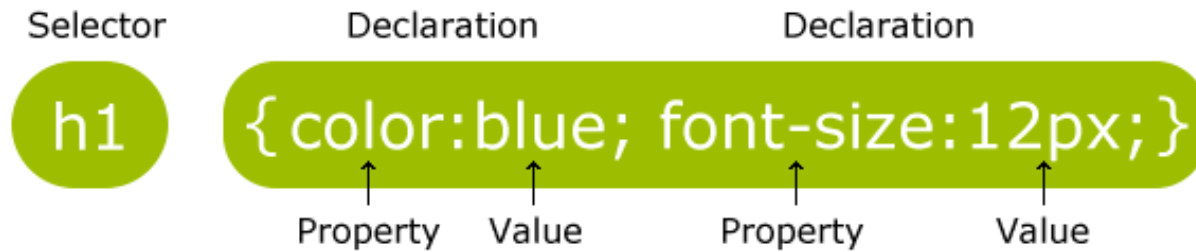
- CSS 3 é a versão atual do CSS;
- O CSS definem como os elementos **HTML são visualizados**;
- O CSS poupa imenso trabalho;
- Folhas de Estilo externas são armazenadas em ficheiros CSS;
- Novidades incorporadas:
 - Não utilizar mais imagens como *background* graças às propriedades de gradiente, transparência, bordas arredondadas;
 - Utilização de transições e animações;
 - **Seletores**.
- Mesmo ambiente do CSS 2.0;
- Até *browser* mais antigos têm compatibilidade (exceto IE 8, 7-).

CSS Exemplos

```
body {  
    background-color: #d0e4fe;  
}  
  
h1 {  
    color: orange;  
    text-align: center;  
}  
  
p {  
    font-family: "Times New Roman";  
    font-size: 20px;  
}
```

```
p {  
    color: red;  
    /* This is a single-line comment */  
    text-align: center;  
}
```

CSS



```
p {color:red;text-align:center;}
```

```
p {  
  color: red;  
  text-align: center;  
}
```

CSS - Selector

- *selector*:
 - é um elemento da marcação HTML identificado por:
 - nome
 - `<p>`, `<h1>`, `<form>`
 - nome de uma classe aplicada ao elemento da marcação HTML
 - `.topo`, `.principal`, `.menu`
 - nome de um identificador ID aplicado ao elemento da marcação HTML
 - `#tudo`, `#auxiliar`, `#rodape`
 - ou por qualquer outro identificador CSS de elementos ou trechos da marcação HTML, genericamente denominados de seletores CSS.

CSS - Selector

- Exemplos
 - Agrupamento – a regra é aplicada a mais do que um *selector*

```
h1 {  
  text-align: center;  
  color: red;  
}
```

```
h2 {  
  text-align: center;  
  color: red;  
}
```

```
p {  
  text-align: center;  
  color: red;  
}
```

```
h1, h2, p {  
  text-align: center;  
  color: red;  
}
```

CSS - Selector

- Exemplos

- **Classe** – podemos criar uma regra para uma classe específica e coloca-la no HTML como o atributo *class*. No CSS, utilizamos o nome da classe precedido por ponto.

```
.center {  
    text-align: center;  
    color: red;  
}
```

```
p.center {  
    text-align: center;  
    color: red;  
}
```

- **ID** – a grande diferença entre o *id* e *classe* é a unicidade. O *id* de determinado nome só pode ser definido a um e somente um elemento HTML dentro do documento. No caso do *id* é precedido por #.

```
#para1 {  
    text-align: center;  
    color: red;  
}
```


CSS - Selector

- Exemplos
 - `.s1{}`: estilo será aplicado na classe `.s1`
 - `#s1{}`: estilo será aplicado no id `#s1`
 - `s1 s2{}`: o estilo será aplicado dentro do elemento `s2` que está dentro de `s1`

Suponha que `s1` seja a `table` e `s2` a `td`, que tem a classe `padrao`:

```
.table td.padrao {  
  font-size: 10pt;  
  color: red;  
}
```

Todas as `td`, que tiverem a classe `padrao`, da `table` terão tamanho da fonte 10pt e cor vermelha.

CSS - *Selector*

- Exemplos

- `s1,s2{}`: aplica o estilo nos dois.
- `s1 *{}`: aplica o estilo para todos os elementos dentro de s1
- `s1{}`: aplica o estilo no s1, por exemplo, caso o s1 fosse o body:

```
body {  
  min-width: 650px;  
}
```

- `.s1 a:link{}`: aplica o estilo aos links dentro de s1

```
a:hover {  
  background-color: yellow;  
}
```

Muda a cor do background para amarelo quando passarmos com o rato por cima do link

Como utilizar CSS

- Três formas:
 - Ficheiro externo;
 - Interno à página;
 - Em linha.
- Ficheiro Externo

```
<head>  
<link rel="stylesheet" type="text/css" href="mystyle.css">  
</head>
```



Ficheiro
externo com
a CSS



Ficheiro atual

```
body {  
    background-color: lightblue;  
}  
  
h1 {  
    color: navy;  
    margin-left: 20px;  
}
```

Como utilizar CSS

- Interno à página

```
<head>
<style>
body {
    background-color: linen;
}

h1 {
    color: maroon;
    margin-left: 40px;
}
</style>
</head>
```

- Em linha

```
<h1 style="color:blue;margin-left:30px;">This is a heading.</h1>
```

Como utilizar CSS

- Quando há sobreposição de CSS

Externo

```
h1 {  
  color: navy;  
  margin-left: 20px;  
}
```

Interno

```
h1 {  
  color: orange;  
}
```

Final

```
color: orange;  
margin-left: 20px;
```

Border

- Adiciona uma linha de limite.
- *border-width*: espessura da linha;

```
<head>
  <style>
    p.dotted {border-style: dotted;}
    p.dashed {border-style: dashed;}
    p.solid {border-style: solid;}
    p.double {border-style: double;}
    p.groove {border-style: groove;}
    p.ridge {border-style: ridge;}
    p.inset {border-style: inset;}
    p.outset {border-style: outset;}
    p.none {border-style: none;}
    p.hidden {border-style: hidden;}
    p.mix {border-style: dotted dashed solid double;}
  </style>
</head>
<body>
  <p class="dotted">A dotted border.</p>
  <p class="dashed">A dashed border.</p>
  <p class="solid">A solid border.</p>
  <p class="double">A double border.</p>
  <p class="groove">A groove border.</p>
  <p class="ridge">A ridge border.</p>
  <p class="inset">An inset border.</p>
  <p class="outset">An outset border.</p>
  <p class="none">No border.</p>
  <p class="hidden">A hidden border.</p>
  <p class="mix">A mixed border.</p>
</body>
```

A dotted border.

A dashed border.

A solid border.

A double border.

A groove border.

A ridge border.

An inset border.

An outset border.

No border.

A hidden border.

A mixed border.

Border-Radius

- Adiciona uma linha de limite de forma arredondada.
- Sintaxe do *border-radius*:

```
<!DOCTYPE html>
<html>
<head>
<style>
  #example1 {
    border: 2px solid red;
    padding: 0px;
    border-radius: 25px;
  }
  #example2 {
    border: 2px solid red;
    padding: 20px;
    border-radius: 50px 20px;
  }
</style>
</head>
<body>

<h2>border-radius: 25px:</h2>
<div id="example1">
<p>The border-radius property defines the radius of the element's corners.</p>
</div>

<h2>border-radius: 50px 20px:</h2>
<div id="example2">
<p>If two values are set; the first one is for the top-left and bottom-right corner, the
second one for the top-right and bottom-left corner.</p>
</div>

</body>
</html>
```

Border-Radius

- Resultado:

border-radius: 25px:

The border-radius property defines the radius of the element's corners.

border-radius: 50px 20px:

If two values are set; the first one is for the top-left and bottom-right corner, the second one for the top-right and bottom-left corner.

Box-Shadow

- Faz a projeção da sombra do elemento em questão.

```
<head>
  <style>
    #exemplo {
      border: 1px solid;
      padding: 10px;
      box-shadow: 5px 10px 10px #888888;
    }
  </style>
</head>
<body>
  <h1>box-shadow</h1>
  <h3>box-shadow: 5px 10px 20px #888888:</h3>
  <div id="exemplo">
    <p>podemos definir a cor da sombra. neste caso é cinzenta.</p>
  </div>
</body>
```

box-shadow

box-shadow: 5px 10px 20px #888888:

podemos definir a cor da sombra. neste caso é cinzenta.

- Primeiro parâmetro: deslocamento horizontal
- Segundo parâmetro: deslocamento vertical
- Terceiro parâmetro: desfoque
- Quarto parâmetro: cor da sombra

Formatação de Texto

- *text-shadow*

- Adiciona sombra aos textos da página
- Sintaxe :
 - `text-shadow: 5px 5px 5px #FF0000;`
- Mesmas propriedades e funcionamento semelhante ao *box-shadow*.

- *font-face*

- Agora os *Web Designers* não estão limitados às fontes do navegador.
- Basta importar no projeto a fonte desejada.
- Sintaxe:
 - ```
@font-face {
 font-family: myFirstFont;
 src: url('Sansation_Light.ttf'),
 url('Sansation_Light.eot'); /* IE9+ */
}
```

## Formatação de Texto - Exemplos

---

```
body {
 color: blue;
}
```

```
h1 {
 color: #00ff00;
}
```

```
h2 {
 color: rgb(255,0,0);
}
```

```
h1 {
 text-align: center;
}
```

```
p.date {
 text-align: right;
}
```

```
p.main {
 text-align: justify;
}
```

```
h1 {
 text-decoration: overline;
}
```

```
h2 {
 text-decoration: line-through;
}
```

```
h3 {
 text-decoration: underline;
}
```

```
p.uppercase {
 text-transform: uppercase;
}
```

```
p.lowercase {
 text-transform: lowercase;
}
```

```
p.capitalize {
 text-transform: capitalize;
}
```

## CSS - Links

---

- *a:link*
- *a:visited*
- *a:hover*

```
/* unvisited link */
a:link {
 color: #FF0000;
}

/* visited link */
a:visited {
 color: #00FF00;
}

/* mouse over link */
a:hover {
 color: #FF00FF;
}
```

## Background

---

- *Background*

- Todas as propriedades do *background* numa só instrução

```
body {
 background: #ffffff url("img_tree.png") no-repeat right top;
}
```

- *background-color*

- Coloca a cor em *background* num elemento

```
body {
 background-color: #b0c4de;
}
```

```
h1 {
 background-color: #6495ed;
}
```

```
p {
 background-color: #e0ffff;
}
```

```
div {
 background-color: #b0c4de;
}
```

## CSS Layout - float and clear

---

```
<!DOCTYPE html>
<html>
<head>
<style>
img {
 float: right;
}
</style>
</head>
<body>
```

```
<p>In this example, the image will float to the right in the paragraph, and
the text in the paragraph will wrap around the image.</p>
```

```
<p>
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus
imperdiet, nulla et dictum interdum, nisi lorem egestas odio, vitae
scelerisque enim ligula venenatis dolor. Maecenas nisl est, ultrices nec
congue eget, auctor vitae massa. Fusce luctus vestibulum augue ut aliquet.
Mauris ante ligula, facilisis sed ornare eu, lobortis in odio. Praesent
convallis urna a lacus interdum ut hendrerit risus congue. Nunc sagittis
dictum nisi, sed ullamcorper ipsum dignissim ac. In at libero sed nunc
venenatis imperdiet sed ornare turpis. Donec vitae dui eget tellus gravida
venenatis. Integer fringilla congue eros non fermentum. Sed dapibus pulvinar
nibh tempor porta. Cras ac leo purus. Mauris quis diam velit.</p>

</body>
</html>
```

In this example, the image will float to the right in the paragraph, and the text in the paragraph will wrap around the image.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus imperdiet, nulla et dictum interdum, nisi lorem egestas odio, vitae scelerisque enim ligula venenatis dolor. Maecenas nisl est, ultrices nec congue eget, auctor vitae massa. Fusce luctus vestibulum augue ut aliquet. Mauris ante ligula, facilisis sed ornare eu, lobortis in odio. Praesent convallis urna a lacus interdum ut hendrerit risus congue. Nunc sagittis dictum nisi, sed ullamcorper ipsum dignissim ac. In at libero sed nunc venenatis imperdiet sed ornare turpis. Donec vitae dui eget tellus gravida venenatis. Integer fringilla congue eros non fermentum. Sed dapibus pulvinar nibh tempor porta. Cras ac leo purus. Mauris quis diam velit.



## CSS Layout - float and clear

---

In this example, the image will float to the left in the paragraph, and the text in the paragraph will wrap around the image.



Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus imperdiet, nulla et dictum interdum, nisi lorem egestas odio, vitae scelerisque enim ligula venenatis dolor. Maecenas nisl est, ultrices nec congue eget, auctor vitae massa. Fusce luctus vestibulum augue ut aliquet. Mauris ante ligula, facilisis sed ornare eu, lobortis in odio. Praesent convallis urna a lacus interdum ut hendrerit risus congue. Nunc sagittis dictum nisi, sed ullamcorper ipsum dignissim ac. In at libero sed nunc venenatis imperdiet sed ornare turpis. Donec vitae dui eget tellus gravida venenatis. Integer fringilla congue eros non fermentum. Sed dapibus pulvinar nibh tempor porta. Cras ac leo purus. Mauris quis diam velit.

```
<!DOCTYPE html>
<html>
<head>
<style>
img {
 float: left;
}
</style>
</head>
<body>
```

```
<p>In this example, the image will float to the left in the paragraph, and
the text in the paragraph will wrap around the image.</p>
```

```
<p>
```

```
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus
imperdiet, nulla et dictum interdum, nisi lorem egestas odio, vitae
scelerisque enim ligula venenatis dolor. Maecenas nisl est, ultrices nec
congue eget, auctor vitae massa. Fusce luctus vestibulum augue ut aliquet.
Mauris ante ligula, facilisis sed ornare eu, lobortis in odio. Praesent
convallis urna a lacus interdum ut hendrerit risus congue. Nunc sagittis
dictum nisi, sed ullamcorper ipsum dignissim ac. In at libero sed nunc
venenatis imperdiet sed ornare turpis. Donec vitae dui eget tellus gravida
venenatis. Integer fringilla congue eros non fermentum. Sed dapibus pulvinar
nibh tempor porta. Cras ac leo purus. Mauris quis diam velit.</p>
```

```
</body>
</html>
```

## CSS Layout - float and clear

---

```
.div1 {
 float: left;
 width: 100px;
 height: 50px;
 margin: 10px;
 border: 3px solid #73AD21;
}
```

### Without clear



div1

div2 - Notice that div2 is after div1 in the HTML code. However, since div1 floats to the left, the text in div2 flows around div1.

```
.div2 {
 border: 1px solid red;
}
```

```
.div3 {
 float: left;
 width: 100px;
 height: 50px;
 margin: 10px;
 border: 3px solid #73AD21;
}
```

### With clear



div3

div4 - Here, clear: left; moves div4 down below the floating div3. The value "left" clears elements floated to the left. You can also clear "right" and "both".

```
.div4 {
 border: 1px solid red;
 clear: left;
}
```

```
</style>
</head>
<body>
```

```
<h2>Without clear</h2>
```

```
<div class="div1">div1</div>
```

```
<div class="div2">div2 - Notice that div2 is after div1 in the HTML code. However,
since div1 floats to the left, the text in div2 flows around div1.</div>
```

```



```

```
<h2>With clear</h2>
```

```
<div class="div3">div3</div>
```

```
<div class="div4">div4 - Here, clear: left; moves div4 down below the floating div3.
The value "left" clears elements floated to the left. You can also clear "right" and
"both".</div>
```



## Background

---

- *background-image*
  - Coloca uma imagem em *background* de um elemento
- *background-repeat*
  - Como uma imagem em *background* se irá repetir
- *background-position*
  - Determina a posição de início de uma imagem em *background*

```
body {
 background-image: url("paper.gif");
}
```

```
body {
 background-image: url("gradient_bg.png");
 background-repeat: repeat-x;
}
```

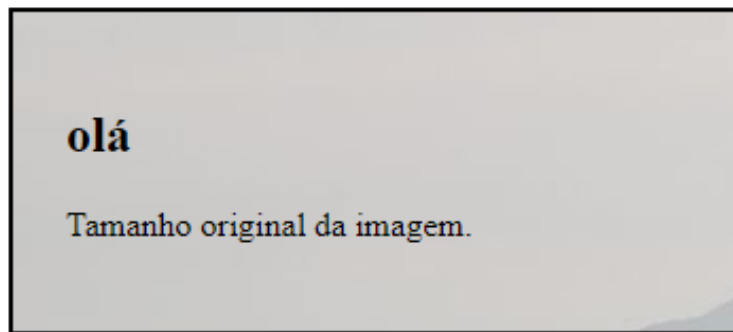
```
body {
 background-image: url("img_tree.png");
 background-repeat: no-repeat;
}
```

```
body {
 background-image: url("img_tree.png");
 background-repeat: no-repeat;
 background-position: right top;
}
```

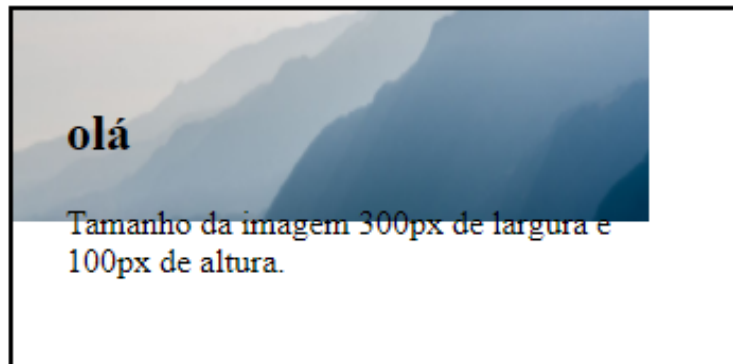
## Background

- O CSS3 contém novas propriedades de *background*:
  - *background-size* – tamanho da imagem

### **background-size: auto (default):**



### **background-size: 300px 100px:**



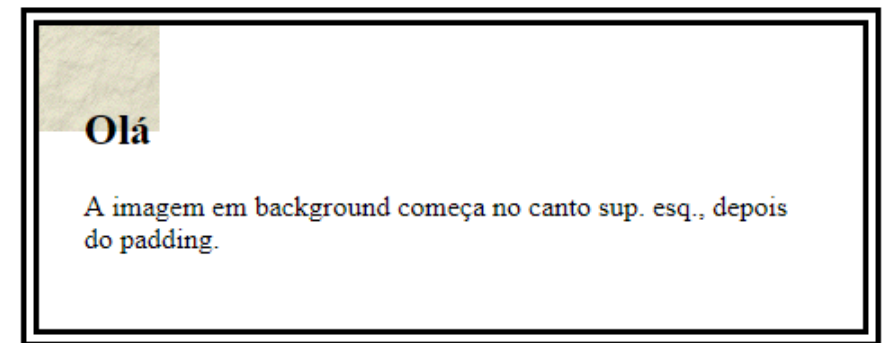
```
<!DOCTYPE html>
<html>
<head>
<style>
 #exemplo1 {
 border: 2px solid black;
 padding: 25px;
 background: url(mountain.jpg);
 background-repeat: no-repeat;
 background-size: auto;
 }
 #exemplo2 {
 border: 2px solid black;
 padding: 25px;
 background: url(mountain.jpg);
 background-repeat: no-repeat;
 background-size: 300px 100px;
 }
</style>
</head>
<body>
 <h2>background-size: auto (default):</h2>
 <div id="exemplo1">
 <h2>olá </h2>
 <p>Tamanho original da imagem.</p>
 </div>
 <h2>background-size: 300px 100px:</h2>
 <div id="exemplo2">
 <h2>olá</h2>
 <p>Tamanho da imagem 300px de largura e 100px de altura.</p>
 </div>
</body>
</html>
```

## Background

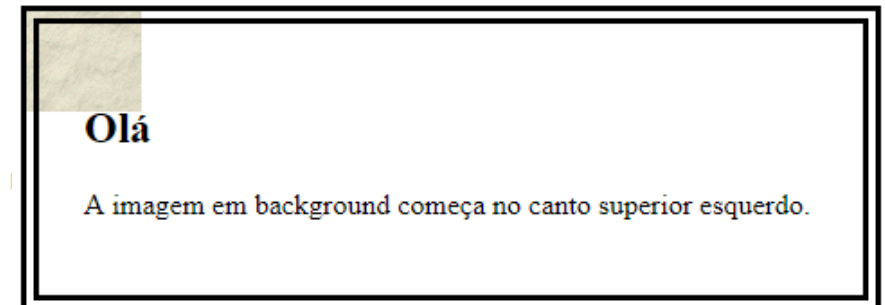
- O CSS3 contém novas propriedades de *background*:
  - *background-origin* – onde começa a imagem

```
<!DOCTYPE html>
<html>
<head>
 <style>
 #exemplo1 {
 border: 10px double black;
 padding: 25px;
 background: url(paper.gif);
 background-repeat: no-repeat;
 background-origin: padding-box;
 }
 #exemplo2 {
 border: 10px double black;
 padding: 25px;
 background: url(paper.gif);
 background-repeat: no-repeat;
 background-origin: border-box;
 }
 </style>
</head>
<body>
 <h2>background-origin: padding-box (default):</h2>
 <div id="exemplo1">
 <h2>Olá</h2>
 <p>A imagem em background começa no canto sup. esq., depois do
 </div>
 <h2>background-origin: border-box:</h2>
 <div id="exemplo2">
 <h2>Olá</h2>
 <p>A imagem em background começa no canto superior esquerdo.</p>
 </div>
</body>
</html>
```

### background-origin: padding-box (default):



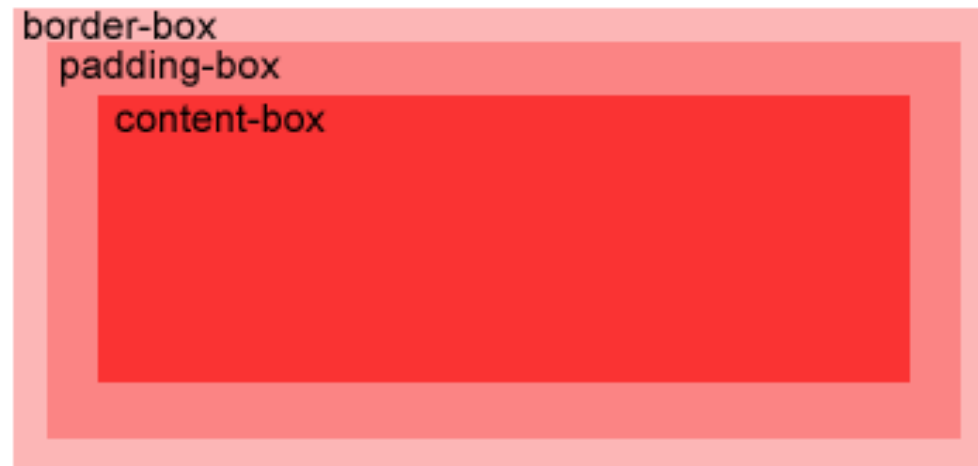
### background-origin: border-box:



## Background-origin

---

- A propriedade *background-origin* especifica o posicionamento das imagens de fundo .
- A imagem de fundo pode ser colocada dentro do *content-box*, *padding-box*, or *border-box*.



## Transições






---

- Para fazer pequenas animações para adicionar dinamismo a páginas.
- Se combinada com seletores e animações é possível criar *designs* realmente complexos e dinâmicos sem utilizar *javascript* ou *flash*.
- Propriedades:
  - *transition-property* – nome da propriedade que será afetada
  - *transition-duration* – duração da transição
  - *transition-timing-function* – tipo de transição, ex.: linear
  - *transition-delay* – quanto tempo demora até começar a animar.
- Exemplos:

```
transition-delay: 2s;
-moz-transition-delay: 2s; /* Firefox 4 */
-webkit-transition-delay: 2s; /* Safari and Chrome */
-o-transition-delay: 2s; /* Opera */
```

## Suporte de *browsers* para as Transições

- Os números antes de -webkit-, -moz-, ou -o- especificam a primeira versão que o prefixo funcionou.

Property					
transition	26.0 4.0 -webkit-	10.0	16.0 4.0 -moz-	6.1 3.1 -webkit-	12.1 10.5 -o-
transition-delay	26.0 4.0 -webkit-	10.0	16.0 4.0 -moz-	6.1 3.1 -webkit-	12.1 10.5 -o-
transition-duration	26.0 4.0 -webkit-	10.0	16.0 4.0 -moz-	6.1 3.1 -webkit-	12.1 10.5 -o-
transition-property	26.0 4.0 -webkit-	10.0	16.0 4.0 -moz-	6.1 3.1 -webkit-	12.1 10.5 -o-
transition-timing-function	26.0 4.0 -webkit-	10.0	16.0 4.0 -moz-	6.1 3.1 -webkit-	12.1 10.5 -o-

# Transições

---

- Para criar transições precisamos de:
  - A propriedade CSS que queremos adicionar um efeito
  - A duração do efeito

```
div {
 width: 100px;
 height: 100px;
 background: red;
 -webkit-transition: width 2s; /* Safari */
 transition: width 2s;
}
```

## Transições – exemplo1

---

```
<!DOCTYPE html>
<html>
<head>
<style>
div {
 width: 100px;
 height: 100px;
 background: red;
 -webkit-transition: width 2s; /* For Safari 3.1 to 6.0 */
 transition: width 2s;
}

div:hover {
 width: 300px;
}
</style>
</head>
<body>

<p>Note: This example does not work in Internet Explorer 9 and
earlier versions.</p>

<div></div>

<p>Hover over the div element above, to see the transition effect.</p>

</body>
</html>
```

[Resultado???](#)



## Transições – mais exemplos

---

- Transformar o exemplo1, de forma a crescer também na vertical, com os seguintes dados:
  - Altura de 300px;
  - Velocidade de transição na vertical de 4s;






[Resultado???](#)

- Transformar o exemplo1, de forma a crescer na vertical e na horizontal com rotação, com os seguintes dados:
  - Altura e largura final de 300px;
  - Velocidade de transição na horizontal e na vertical de 2s;
  - Rotação de 180graus;
  - Velocidade de transformação de 2s;

[Resultado???](#)

## Suporte de *browsers* para as Animações

- Os números antes de -webkit-, -moz-, ou -o- especificam a primeira versão que o prefixo funcionou.

Property					
@keyframes	43.0 4.0 -webkit-	10.0	16.0 5.0 -moz-	9.0 4.0 -webkit-	30.0 15.0 -webkit- 12.0 -o-
animation	43.0 4.0 -webkit-	10.0	16.0 5.0 -moz-	9.0 4.0 -webkit-	30.0 15.0 -webkit- 12.0 -o-

## Animações – exemplo4

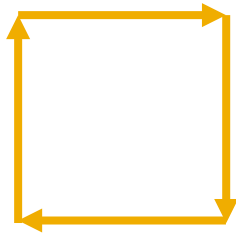
```
<!DOCTYPE html>
<html>
<head>
<style>
div {
 width: 100px;
 height: 100px;
 background-color: red;
 -webkit-animation-name: example; /* Safari 4.0 - 8.0 */
 -webkit-animation-duration: 4s; /* Safari 4.0 - 8.0 */
 animation-name: example;
 animation-duration: 4s;
}
/* Safari 4.0 - 8.0 */
@-webkit-keyframes example {
 0% {background-color: red;}
 25% {background-color: yellow;}
 50% {background-color: blue;}
 100% {background-color: green;}
}
/* Standard syntax */
@keyframes example {
 0% {background-color: red;}
 25% {background-color: yellow;}
 50% {background-color: blue;}
 100% {background-color: green;}
}
</style>
</head>
<body>
 <p>Note: This example does not work in Internet Explorer 9
 and earlier versions.</p>
 <div></div>
</body>
</html>
```

Resultado???

## Animações – mais exemplos

---

- Transformar o exemplo4, com os seguintes dados:
  - Reposicionamento do objeto relativamente a esquerda e topo (200px), de forma a que faça o seguinte trajeto:



Resultado???

# Animações

---

- *animation-iteration-count*

```
div {
 width: 100px;
 height: 100px;
 position: relative;
 background-color: red;
 animation-name: example;
 animation-duration: 4s;
 animation-iteration-count: infinite;
}
```

- Trajeto contínuo com 3 voltas

Resultado???

## Animações – mais exemplos

---

- Trajeto contínuo com 3 voltas, contudo no sentido contrário ao anteriormente definido

Resultado???

- Trajeto com 3 voltas, mas a cada volta inverter o sentido do trajeto

Resultado???

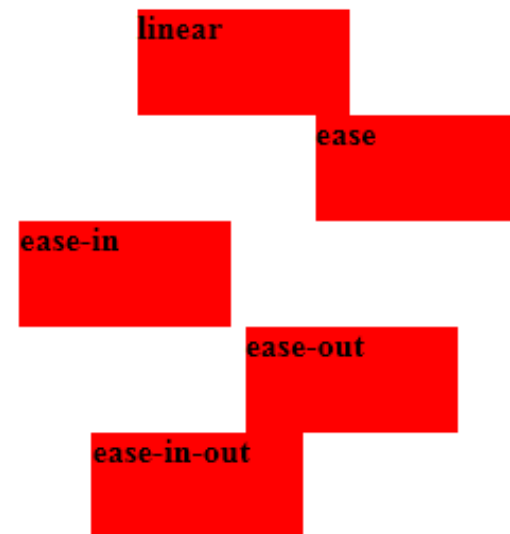
# Animações

---

## ○ *animation-timing-function*

- `ease` - specifies an animation with a slow start, then fast, then end slowly (this is default)
- `linear` - specifies an animation with the same speed from start to end
- `ease-in` - specifies an animation with a slow start
- `ease-out` - specifies an animation with a slow end
- `ease-in-out` - specifies an animation with a slow start and end
- `cubic-bezier(n,n,n,n)` - lets you define your own values in a cubic-bezier function

- Tentar reproduzir o seguinte exemplo:



## Pratica com apoio do W3Schools

---

- *Backgrounds*
- *Borders*
- *Boxmodel*
- Formatação de texto
- Tabelas
- Barras de navegação
- *Dropdown*
- Formulários
- Conceito de colunas
- *Tooltips*
- Botões
- *Media Queries*