



**INSTITUTO FEDERAL**

Catarinense

Campus Camboriú

# Desenvolvimento Web I

**Professor: Euclides Paim**

*euclides.paim@ifc.edu.br*



**INSTITUTO FEDERAL**

Catarinense

Campus Camboriú

# CSS3 - Transições e Animações

**Professor: Euclides Paim**

*euclides.paim@ifc.edu.br*



## CSS3: Transições e Animações

### Transições CSS

**Transições CSS** permitem alterar os valores de uma propriedade suavemente, durante um determinado período. Neste capítulo, você aprenderá sobre as seguintes propriedades:

- *transition*
- *transition-delay*
- *transition-duration*
- *transition-property*
- *transition-timing-function*



## **CSS3: Transições e Animações**

### Como usar Transições CSS?

Para criar um efeito de transição, é necessário especificar duas coisas:

- a propriedade CSS que você deseja adicionar o efeito e a
- a duração do efeito

**Nota:** Se a parte de duração não for especificado, a transição não terá nenhum efeito, porque o valor padrão é 0.

## CSS3: Transições e Animações

### Como usar Transições CSS?

O exemplo a seguir mostra um elemento `<div>` de 100 pixels x 100 pixels vermelho. O elemento `<div>` também tem especificado, um efeito de transição para a propriedade *width* (largura), com uma duração de 2 segundos:

```
div {  
  width: 100px;  
  height: 100px;  
  background: red;  
  transition: width 2s;  
}
```

O efeito de transição terá início quando a propriedade CSS especificada *width* (largura) mudar de valor.

## CSS3: Transições e Animações

### Como usar Transições CSS?

Agora, vamos especificar um novo valor para a propriedade de largura quando um usuário passar o mouse sobre o elemento <div>:

```
div:hover {  
    width: 300px;  
}
```

Observe que quando o cursor do mouse andar para fora do elemento, ele irá gradualmente mudar de volta para o seu estilo original. (Código nos arquivos da aula.)

## CSS3: Transições e Animações

### Alterar vários valores de propriedades

O exemplo a seguir acrescenta um efeito de transição para as propriedades largura e altura, com uma duração de 2 segundos para a largura e 4 segundos para que a altura;

```
div {  
  transition: width 2s, height 4s;  
}
```

Observe a alteração nas duas propriedades no ex2.

A propriedade ***transition-timing-function*** especifica a velocidade da curva de transição do efeito. A propriedade *transition-timing-function* pode ter os seguintes valores:

- *ease* - especifica um efeito de transição com um início lento e em seguida rápida, finalizando lentamente (este é o padrão).
- *linear* - especifica um efeito de transição com a mesma velocidade do início ao fim.
- *ease-in* - especifica um efeito de transição com um início lento.
- *ease-out* - especifica um efeito de transição com um final lento.
- *ease-in-out* - especifica um efeito de transição com início e fim lentos.
- *cubic-bezier(n,n,n,n)* - permite definir seus próprios valores em uma função cúbica-bezier

Mais informações sobre a Curva de Bézier: (<http://encurtador.com.br/nAMS2>)



## CSS3: Transições e Animações

### Especifique a velocidade da curva de transição

O exemplo a seguir mostra algumas das diferentes velocidades de transição que podem ser usadas como valor da propriedade *transition-timing-function*:

```
#div1 {transition-timing-function: linear;}  
#div2 {transition-timing-function: ease;}  
#div3 {transition-timing-function: ease-in;}  
#div4 {transition-timing-function: ease-out;}  
#div5 {transition-timing-function: ease-in-out;}
```

Exemplos disponíveis nos arquivos da aula (ex3).



## CSS3: Transições e Animações

### Retardar o efeito de transição

A propriedade *transition-delay* especifica um atraso (em segundos) para o efeito de transição. O exemplo seguinte tem um atraso de 1 segundo antes de iniciar:

```
div {  
    transition-delay: 1s;  
}
```

Ob.: Exemplo disponível nos arquivos da aula (ex4).



As propriedades CSS de transição podem ser especificadas uma por uma, como aqui:

```
div {  
    transition-property: width;  
    transition-duration: 2s;  
    transition-timing-function: linear;  
    transition-delay: 1s;  
}
```

ou resumidamente usando a forma abreviada *transition*:

```
div {  
    transition: width 2s linear 1s;  
}
```



## CSS3: Transições e Animações

### Animações CSS

CSS permite a animação de elementos HTML sem a utilização de *JavaScript* ou *Flash*. Neste capítulo, estudaremos as seguintes propriedades:

- *@keyframes*
- *animation-name*
- *animation-duration*
- *animation-delay*
- *animation-iteration-count*
- *animation-direction*
- *animation-timing-function*
- *animation-fill-mode*
- *animation*



## CSS3: Transições e Animações

### Animações CSS

Uma animação permite que um elemento mude gradualmente a partir de um estilo para outro. Você pode alterar um número indeterminado de propriedades CSS , quantas vezes você quiser. Para usar animações CSS, você deve primeiro especificar alguns quadros-chave (*keyframes*) para a animação.

Keyframes carregam os estilos que o elemento terá em determinado momento. Ao especificar estilos CSS dentro da regra **@keyframes**, a animação irá, em determinados momento, mudar gradualmente a partir do estilo atual para o novo estilo. Para que uma animação funcione, você deve vincular a animação a um elemento.



## CSS3: Transições e Animações

### Animações CSS

O exemplo a seguir vincula a animação "*example*" ao elemento `<div>`. A animação terá a duração de 4 segundos, mudando gradualmente a cor de fundo do elemento `<div>` de *red* para *yellow*:

```
/* O código da animação */
@keyframes example {
  from {background-color: red;}
  to {background-color: yellow;}
}

/* O elemento em que será aplicada a animação */
div {
  width: 100px;
  height: 100px;
  background-color: red;
  animation-name: example;
  animation-duration: 4s;
}
```

Código do exemplo disponível nos arquivos da turma (ex1).



## CSS3: Transições e Animações

### Animações CSS

**Nota:** A propriedade *animation-duration* define quanto tempo uma animação deve demorar para ser concluída. Se a propriedade *animation-duration* não for declarada, nenhuma animação irá ocorrer, porque o valor padrão é de 0 segundos.

No exemplo anterior especificamos quando o estilo vai mudar usando as palavras-chave "*from*" e "*to*" (que representam 0% (iniciar) e 100% (completo)). Também é possível usar porcentagem para adicionar outros estilos conforme veremos no próximo exemplo.



## CSS3: Transições e Animações

### Animações CSS

O exemplo a seguir irá mudar a cor de fundo do elemento <div> conforme a porcentagem, 25% completo, 50% completo, e novamente quando a animação estiver 100% concluída:

```
/* O código da animação */
@keyframes example {
  0%   {background-color: red;}
  25%  {background-color: yellow;}
  50%  {background-color: blue;}
  100% {background-color: green;}
}

/* O elemento em que será aplicada a animação */
div {
  width: 100px;
  height: 100px;
  background-color: red;
  animation-name: example;
  animation-duration: 4s;
}
```

Código do exemplo disponível nos arquivos da turma (ex2).





## CSS3: Transições e Animações

### Animações CSS

Este exemplo irá alterar tanto o *background-color* quanto a posição do elemento <div> quando a animação 25% completa, 50% completa, e quando a animação estiver 100% concluída:

```
/* O código da animação */
@keyframes example {
  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;}
}
```

```
/* O elemento em que será aplicada a animação */
div {
  width: 100px;
  height: 100px;
  position: relative;
  background-color: red;
  animation-name: example;
  animation-duration: 4s;
}
```

Código do exemplo disponível nos arquivos da turma (ex3).

## CSS3: Transições e Animações

### Delay de animação

A propriedade *animation-delay* especifica um atraso para o início de uma animação. O exemplo seguinte tem um atraso de 2 segundos, antes de iniciar a animação:

```
div {  
  width: 100px;  
  height: 100px;  
  position: relative;  
  background-color: red;  
  animation-name: example;  
  animation-duration: 4s;  
  animation-delay: 2s;  
}
```

Os valores negativos também são permitidos. Se usar valores negativos, a animação vai começar como se já tivesse rodando por *N* segundos.



## CSS3: Transições e Animações

### Animações CSS

Defina quantas vezes uma animação deve ser executada. A propriedade *animation-iteration-count* especifica o número de vezes que uma animação deve ser executada. O exemplo a seguir irá executar a animação 3 vezes antes de parar:

```
div {  
  width: 100px;  
  height: 100px;  
  position: relative;  
  background-color: red;  
  animation-name: example;  
  animation-duration: 4s;  
  animation-iteration-count: 3;  
}
```

O valor *"infinite"* pode ser usado para definir que a animação vai rodar continuamente.  
(Exemplo ex4)



## CSS3: Transições e Animações

### Animações CSS

Podemos executar a animação na direção inversa ou em ciclos alternados através da propriedade específica ***animation-direction***. O elemento pode ser reproduzida para a frente, para trás ou em ciclos alternados. A propriedade *animation-direction* pode ter os seguintes valores:

- *normal* - A animação é executada normalmente (para a frente). Este é o padrão.
- *reverse* - A animação é reproduzida em sentido inverso (para trás).
- *alternate* - A animação é reproduzida para a frente primeiro, depois para trás.
- *alternate-reverse* - A animação é reproduzida para trás em primeiro lugar, e em seguida para frente.

## CSS3: Transições e Animações

### Animações CSS

O exemplo a seguir utiliza o valor "*alternate-reverse*" para fazer a animação correr para trás em primeiro lugar e em seguida, para frente:

```
div {  
  width: 100px;  
  height: 100px;  
  position: relative;  
  background-color: red;  
  animation-name: example;  
  animation-duration: 4s;  
  animation-iteration-count: 2;  
  animation-direction: alternate-reverse;  
}
```

Código de exemplo nos arquivos da turma (ex5).



## CSS3: Transições e Animações

### Especifique o curva de velocidade da animação

A propriedade ***animation-timing-function*** especifica a curva de velocidade da animação. A propriedade *animation-timing-function* pode ter os seguintes valores:

- *ease* - Especifica uma animação com um início lento, em seguida, rápido, terminando lentamente (este é o padrão).
- *linear* - Especifica uma animação com a mesma velocidade do início ao fim.
- *ease-in* - Especifica uma animação com um início lento.
- *ease-out* - Especifica uma animação com uma final lento
- *ease-in-out* - Especifica uma animação com um início e fim lentos.
- *cubic-bezier(n,n,n,n)* - Permite definir os seus próprios valores em uma função cúbica-bezier

Mais informações sobre a Curva de Bézier: (<http://encurtador.com.br/nAMS2>)

## CSS3: Transições e Animações

### Animações CSS

O exemplo a seguir mostra a algumas das diferentes curvas de velocidade que podem ser utilizadas:

```
#div1 {animation-timing-function: linear;}  
#div2 {animation-timing-function: ease;}  
#div3 {animation-timing-function: ease-in;}  
#div4 {animation-timing-function: ease-out;}  
#div5 {animation-timing-function: ease-in-out;}
```

Código de exemplo nos arquivos da turma (ex6).



## CSS3: Transições e Animações

### Animações CSS

A propriedade ***shorthand*** (forma abreviada). O exemplo a seguir utiliza seis propriedades de animação:

```
div {  
  animation-name: example;  
  animation-duration: 5s;  
  animation-timing-function: linear;  
  animation-delay: 2s;  
  animation-iteration-count: infinite;  
  animation-direction: alternate;  
}
```

O mesmo efeito de animação acima pode ser conseguido utilizando a abreviação *animation*:

```
div {  
  animation: example 5s linear 2s infinite alternate;  
}
```





## **CSS3: Transições e Animações**

### **Exercícios de Fixação**

Para criar uma memória do conteúdo estudado, aplique estes conceitos no seu site:

1. inserindo estes novos recursos caso ainda não possua
2. atualizando elementos existentes caso já existam

**Nota:** A entrega deste material será considerada como presença para a presente aula



# Referências

## Referências Básicas

Livro NIEDERAUER, Juliano. Desenvolvendo websites com PHP: aprenda a criar websites dinâmicos e interativos com PHP e bancos de dados. 2. ed. rev. e atual. São Paulo: Novatec, 2011.

Livro SILVA, Maurício Samy. HTML 5: a linguagem de marcação que revolucionou a web. São Paulo: Novatec, 2011.

Livro SILVA, Maurício Samy. CSS3: desenvolva aplicações web profissionais com uso dos poderosos recursos de estilização das CSS3. São Paulo: Novatec, 2012.

## Referências Complementares

Livro DEITEL, Paul J. Ajax, . Rich Internet applications e desenvolvimento Web para programadores. . Pearson Prentice Hall. 2009

Livro DALL'OGGIO, Pablo. PHP: programando com orientação a objetos. . Novatec. 2009

Livro SOARES, Wallace. PHP 5: conceitos, programação e integração com banco de dados. . Érica. 2010

Livro SILVA, Maurício Samy. Criando sites com HTML: sites de alta qualidade com HTML e CSS. . Novatec. 2010

Livro FLANAGAN, David. o guia definitivo. . O Really. 2012 ★

## Referências na Internet

<https://www.w3schools.com>

<https://developer.mozilla.org/pt-BR/docs/Web>