

The logo for CSS3, featuring a gray square on the left and a red rectangle on the right with the text "CSS3" in white.

CSS3

Tecnologias Web
2016/2017

2

@font-face rule

- Com o CSS3, deixou de ser necessário depender das fontes instaladas no cliente
 - A fonte é disponibilizada no servidor, sendo efetuado o *download* para o cliente sempre que necessário (quando a fonte não está previamente instalada no cliente)

```
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8" />
<style type="text/css">
  @font-face
  {
    font-family: myFirstFont;
    src: url('Sansation_Light.ttf')
        ,url('Sansation_Light.eot');
  }
  div
  {
    font-family:myFirstFont;
  }
</style>
</head>
<body>
  <div>Download de novas fontes.</div>
</body>
</html>
```

*seletor{transform:****}*

- *transform*
 - permite mover, mudar a escala, rodar, e torcer os elementos
 - {transform:translate()}
 - {transform:rotate()}
 - {transform:scale()}
 - {transform:skew()}
 - ...



transform:translate()

- A referência para o movimento de translação é a posição original do objecto
 - Deslocamento horizontal positivo (to right)
 - Deslocamento vertical positivo (to bottom)

```
<!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>
```



transform:rotate()

7

- O elemento é rodado no sentido dos ponteiros do relógio
- Valores negativos rodam no sentido oposto

```
<!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>
```



transform	10	16	36	9	23
transform-origin	10	16	36	9	23

transform:scale()

8

- O elemento aumenta ou diminui de tamanho dependendo dos valores de escala (x,y)

```
<!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

9

- `seletor{transform: rotate X()}; seletor{transform: rotateY()}`

```
<!DOCTYPE html>
<html>
<head>
<style type="text/css">

    div{
        width:280px;
        height:190px;}
    #div2
    {
        transform:rotateX(180deg);
        -webkit-transform:rotateX(180deg); /* Safari and Chrome */
        -moz-transform:rotateX(180deg); /* Firefox */
    }
    img{
        width:280px;
        height:188px;
    }
</style>
</head>

<body>
    <div></div>
    <div id="div2"></div>
</body>
</html>
```



10

`seletor{transition:****}`

Transition

```
<!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



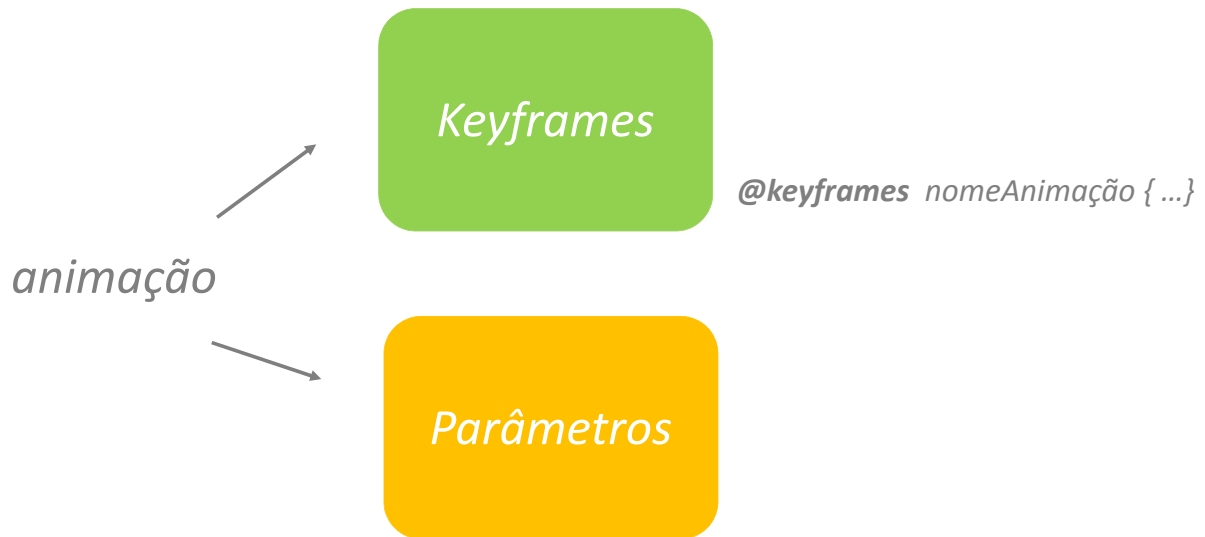
↓
Uma vez o rato sobreposto ao elemento div, efectua-se a transição da posição inicial para a posição final (duração 3s)

Transição width Elemento div



*seletor{animation:****}*

Animation



@keyframes

14

- O uso de keyframes implica definir o estado do objeto, ou das suas propriedades, nos diferentes passos da animação

```
@keyframes nome{controlo{estilos-CSS}}
```

- nome
 - obrigatório
- controlo
 - obrigatório. Aceita valores de 0 a 100% e os valores **from** (igual a 0%) e **to** (igual a 100%)
- estilos-CSS
 - obrigatório

@keyframes

15

- %
 - Permite a criação de um número variável de keyframes

```
@keyframes example {  
  0% {background-color: red;}  
  25% {background-color: yellow;}  
  50% {background-color: blue;}  
  100% {background-color: green;}  
}
```

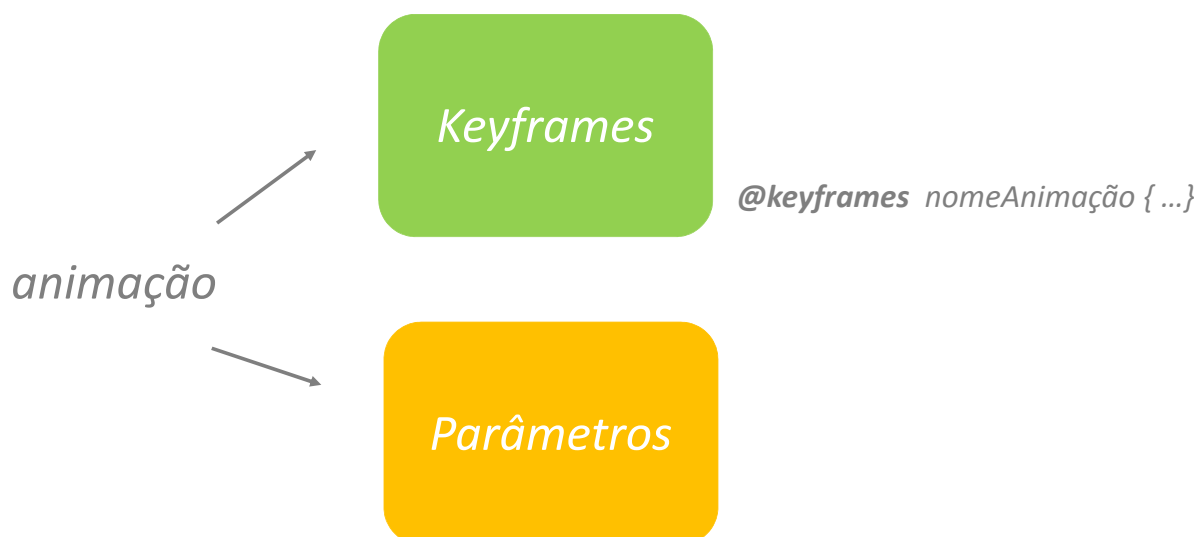
- *from/to*

```
@keyframes example {  
  from {background-color: red;}  
  to {background-color: green;}  
}
```

- Se o início e fim da animação são comuns é possível uma notação condensada

```
@keyframes example {  
  0%,100% {background-color: red;}  
  50% {background-color: green;}  
}
```

Animation



Parâmetros

17

```
animation: <name> <duration> <timing_function> <delay> <iteration_count>  
         <direction> ... ;
```

- <name>
 - nome da animação. **Obrigatório!**
- <duration>
 - tempo que uma animação demora a completar um ação, especificado em segundos
 - valor por defeito é 0 (sem animação)
 - valores negativos são tratados como 0
- <timing_function>
 - Define a velocidade da animação (constante, variável, ...)

Parâmetros

18

```
animation: <name> <duration> <timing_function> <delay> <iteration_count>  
         <direction> ... ;
```

- <delay>
 - define o intervalo de tempo entre o início da animação e o evento que a despoletou
 - valores diferentes de zero devem indicar a unidade – **s** (segundos) ou **ms** (milissegundos).
- <iteration_count>
 - especifica o número de vezes que a animação é executada. **infinite** significa é executada de forma repetida sem parar
- <direction>
 - determina a direção da animação
 - **normal**
 - o valor por defeito. A animação é sempre reproduzida na direção normal
 - **alternate**
 - executa as iterações ímpares na posição normal e as iterações pares na direção oposta

Parâmetros

19

- Propriedades Individuais

```
#box {  
  animation-name: exemplo;  
  animation-duration: 4s;  
  animation-iteration-count: 10;  
  animation-direction: alternate;  
  animation-timing-function: ease-out;  
  animation-delay: 2s;  
}
```

- Propriedade agregada

- diferentes valores separados por um espaço,
- as propriedades **duration** e **delay** devem ser especificadas por esta ordem, a ordem das restantes propriedades não é relevante

```
#box { animation: exemplo 4s 2s 10 alternate ease-out;}
```

Animações múltiplas

20

- Um **seletor** pode declarar múltiplas animações necessitando apenas que elas estejam separadas por vírgulas

```
.anima { animation:  
  primeira-animacao 3s infinite,  
  segunda-animacao 2s;  
}
```