

# Lenguaje de Marcas y Sistemas de Gestión

## UA 2.17 - Introducción CSS: Animaciones CSS



Centro Profesional  
**Universidad  
Europea**

Raúl Rodríguez Mercado

[raul.rodriguez@universidadeuropea.es](mailto:raul.rodriguez@universidadeuropea.es) / @raulrodriguezue

Dpto. Ciencias y Tecnología de la Informática y Comunicación

## UA 2.17: Introducción CSS: Animaciones CSS



### Objetivos

- Definir qué es CSS
- Conocer la estructura de las hojas de estilo y como se aplican a los documentos HTML



## UA 2.17: Introducción CSS: Animaciones CSS



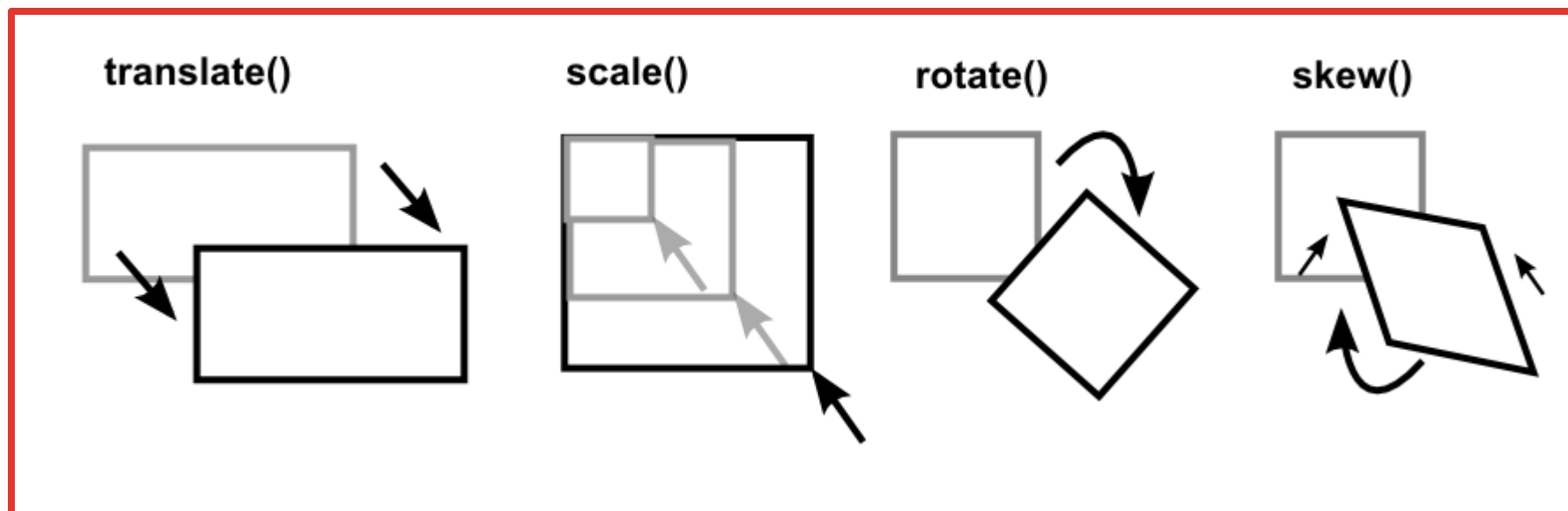
### Contenidos

- Transformaciones
- Transiciones
- Animaciones



### Transformaciones

- Las transformaciones es uno de los elementos más interesantes que se introducen en CSS3 para convertir el lenguaje de hojas de estilo en un sistema capaz de realizar todo tipo de efectos visuales, incluido 2D y 3D.
- Las funciones principales que aplican son las siguientes:



### Transformaciones

- ✓ **Translate:** Define el desplazamiento del objeto en horizontal y vertical. Viene dado por dos unidades separadas por comas, la primera será la que aplique horizontalmente, y la segunda verticalmente. También se puede definir únicamente un desplazamiento en horizontal con ***translateX*** o en vertical con ***translateY***.

Funciones	Significado
<code>translate( <i>x</i>, <i>y</i> )</code>	Traslada el elemento una distancia de <i>x</i> horizontalmente y de <i>y</i> verticalmente.
<code>translateX( <i>x</i> )</code>	Traslada el elemento una distancia de <i>x</i> horizontalmente.
<code>translateY( <i>y</i> )</code>	Traslada el elemento una distancia de <i>y</i> verticalmente.

## UA 2.17: Introducción CSS: Animaciones CSS



### Transformaciones

#### Ejemplo Translate



```
<div>Estático 1</div>  
<div class="moved">Movido</div>  
<div>Estático 2</div>
```



```
div {  
  width: 60px;  
  height: 60px;  
  background-color: skyblue;  
}  
  
.moved {  
  transform: translate(10px,20px);  
  background-color: pink;  
}
```



Estático  
1

Movido

Estático  
2

```
transform: translateX(10px);  
background-color: pink;
```

Estático  
1

Movido

Estático  
2

```
transform: translateY(15px);  
background-color: pink;
```

Estático  
1

Movido

Estático  
2

## UA 2.17: Introducción CSS: Animaciones CSS



### Transformaciones

- ✓ **Scale:** Define un cambio de la anchura y altura del objeto. Viene dado por dos unidades relativas separadas por comas, siendo la primera la que aplique al ancho, y la segunda a la altura.

Funciones	Significado
<code>scale( <i>fx</i>, <i>fy</i> )</code>	Reescala el elemento a un nuevo tamaño con factor <i>fx</i> horizontal y factor <i>fy</i> vertical.
<code>scaleX( <i>fx</i> )</code>	Reescala el elemento a un nuevo tamaño con factor <i>fx</i> horizontal.
<code>scaleY( <i>fy</i> )</code>	Reescala el elemento a un nuevo tamaño con factor <i>fy</i> vertical.

## UA 2.17: Introducción CSS: Animaciones CSS



### Transformaciones

#### Ejemplo Scale

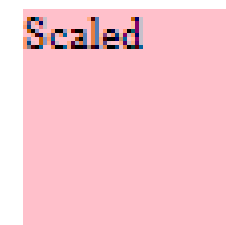
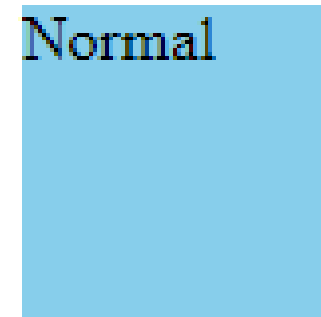


```
<body>
  <div>Normal</div>
  <div class="scaled">Scaled</div>
</body>
```



```
div {
  width: 80px;
  height: 80px;
  background-color: skyblue;
}

.scaled {
  transform: scale(0.7);
  background-color: pink;
}
```





## UA 2.17: Introducción CSS: Animaciones CSS



### Transformaciones

- ✓ **Rotate:** Define una rotación en grados del objeto. Viene dado por el valor de la rotación en grados.

Funciones	Significado
<code>rotate( <i>deg</i> )</code>	Establece una rotación 2D de <i>deg</i> grados.
<code>rotateX( <i>xdeg</i> )</code>	Establece una rotación 2D de <i>xdeg</i> grados sólo para el eje horizontal X.
<code>rotateY( <i>ydeg</i> )</code>	Establece una rotación 2D de <i>ydeg</i> grados sólo para el eje vertical Y.

- ✓ Podemos establecer únicamente una rotación en el eje horizontal empleando `rotateX` en lugar de `rotate` o en el eje vertical usando `rotateY`.

# UA 2.17: Introducción CSS: Animaciones CSS



## Transformaciones

### Ejemplo Rotate



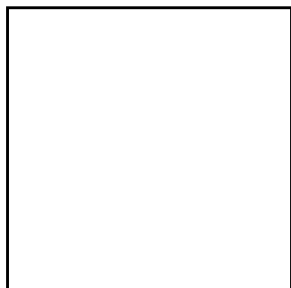
```
<body>
  <div class="col">
    <h3>Contenedor sin transform</h3>
    <div class="muestra"></div>
  </div>
  <div class="col">
    <h3>Contenedor con transform</h3>
    <div class="transform muestra"></div>
  </div>
</body>
```



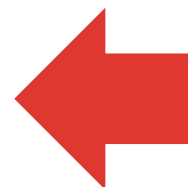
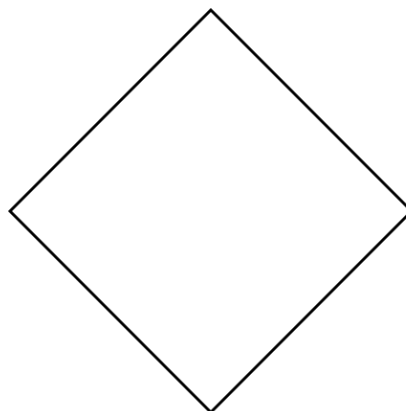
```
<style type="text/css">
  div {
    box-sizing: border-box;
    font-family: Arial, Helvetica, sans-serif;
    padding: .5em;
    font-size: 2em;
    text-align: center;
    width: 300px;
    height: 300px;

    col {
      width: 50%;
      height: 100%;
      float: left;
    }
  }
  .transform {
    -webkit-transform: rotate(45deg);
    transform: rotate(45deg);
  }
  .muestra {
    border: medium solid black;
    margin: 100px auto;
  }
</style>
```

Contenedor sin transform



Contenedor con transform



### Transformaciones

- ✓ **Skew:** Define una inclinación en grados del objeto. Viene dado por el valor de la inclinación en grados para el eje horizontal y vertical. Como ocurre en los otros casos, podemos definir únicamente una inclinación en el eje horizontal empleando skewX y en el eje vertical skewY en lugar de skew.

Funciones	Significado
skew( <i>xdeg</i> , <i>ydeg</i> )	Establece un ángulo para una deformación 2D
skewX( <i>xdeg</i> )	Establece un ángulo para una deformación 2D respecto al eje X
skewY( <i>ydeg</i> )	Establece un ángulo para una deformación 2D respecto al eje Y

# UA 2.17: Introducción CSS: Animaciones CSS



## Transformaciones

### Ejemplo Rotate

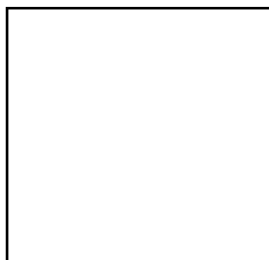


```
<body>
  <div class="col">
    <h3>Contenedor sin transform</h3>
    <div class="muestra"></div>
  </div>
  <div class="col">
    <h3>Contenedor con transform</h3>
    <div class="transform muestra"></div>
  </div>
</body>
```

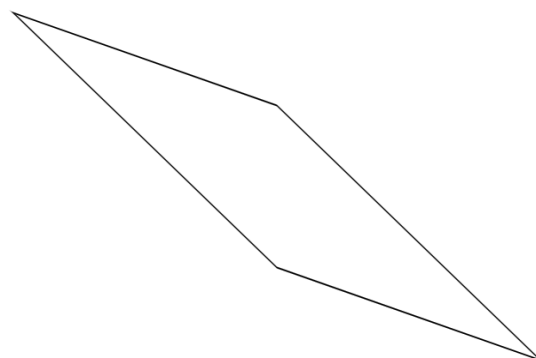


```
div {
  box-sizing: border-box;
  font-family: Arial, Helvetica, sans-serif;
  padding: .5em;
  font-size: 2em;
  text-align: center;
  width: 300px;
  height: 300px;
}
.col {
  width: 50%;
  height: 100%;
  float: left;
}
.transform {
  -webkit-transform: skew(45deg, 20deg);
  transform: skew(45deg, 20deg);
}
.muestra {
  border: medium solid black;
  margin: 100px auto;
}
```

Contenedor sin transform

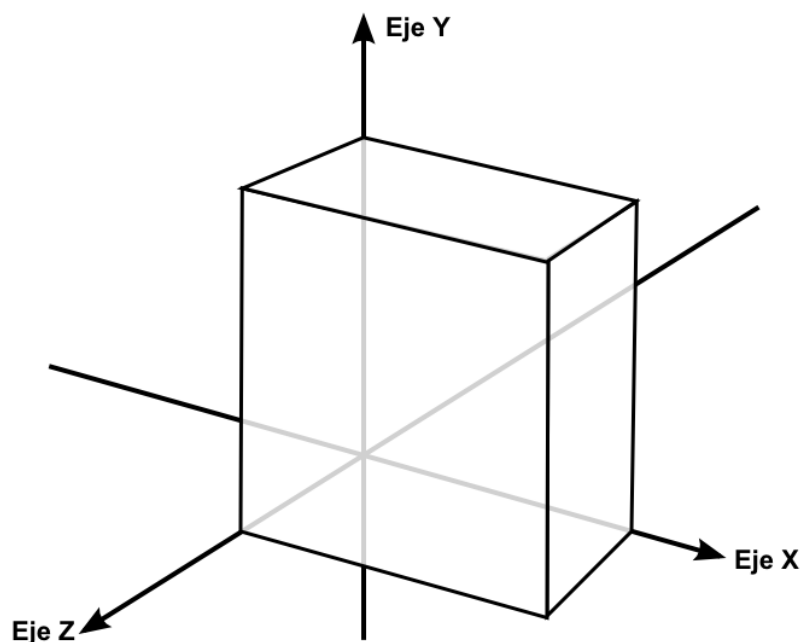


Contenedor con transform



### Transformaciones: 3D en CSS

- ✓ A todas las transformaciones anteriores se les puede añadir el conocido eje Z para definir una “tercera dimensión” al plano 2D con el que hemos trabajado hasta ahora. Recordar siempre que el **eje X** es el eje horizontal, el **eje Y** es el eje vertical y el **eje Z** es el eje de profundidad.



Funciones	Significado
<code>translateZ( <i>z</i> )</code>	Traslada el elemento una distancia de <i>z</i> en el eje de profundidad.
<code>translate3d( <i>x</i>, <i>y</i>, <i>z</i> )</code>	Establece una translación 3D, donde aplica los parámetros a cada eje.
<code>scaleZ( <i>fz</i> )</code>	Reescala el elemento a un nuevo tamaño con factor <i>fz</i> de profundidad.
<code>scale3d( <i>fx</i>, <i>fy</i>, <i>fz</i> )</code>	Establece un escalado 3D, donde aplica los factores a cada eje.
<code>rotateZ( <i>zdeg</i> )</code>	Establece una rotación 2D de <i>zdeg</i> grados sólo para el eje de profundidad Z.
<code>rotate3d( <i>x</i>, <i>y</i>, <i>z</i>, <i>deg</i> )</code>	Establece una rotación 3D, donde aplica a cada eje el número de grados <i>deg</i> .
<code>perspective( <i>n</i> )</code>	Establece una perspectiva 3D

### Transformaciones: 3D en CSS

#### Ejemplos básicos

### Efecto de Perspectiva



```
<html>
  <head>
    <link rel="stylesheet" type="text/css" href="perspectiva.css" />
  </head>
  <body>
    <div class="main">
      <h3>Efecto de Perspectiva</h3>
      <div class="transform muestra">3D</div>
    </div>
  </body>
</html>
```

```
div {
  box-sizing: border-box;
  padding: .5em;
  font-family: Arial, Helvetica, sans-serif;
  text-align: center;
}
.main {
  font-size: 2em;
  width: 50%;
  height: 100%;
  margin: 0 auto;
  /* perspective: 300px; */
}
.muestra {
  -webkit-transform: rotateX(45deg);
  transform: rotateX(45deg);
  background-color: gray;
  margin: 1em auto;
  width: 300px;
  height: 300px;
  color: white;
}
```

### Transformaciones: 3D en CSS

#### Ejemplos básicos

#### Efecto de Perspectiva



```
<html>
  <head>
    <link rel="stylesheet" type="text/css" href="perspectiva.css" />
  </head>
  <body>
    <div class="main">
      <h3>Efecto de Perspectiva</h3>
      <div class="transform muestra">3D</div>
    </div>
  </body>
</html>
```

```
div {
  box-sizing: border-box;
  padding: .5em;
  font-family: Arial, Helvetica, sans-serif;
  text-align: center;
}
.main {
  font-size: 2em;
  width: 50%;
  height: 100%;
  margin: 0 auto;
  perspective: 300px;
}
.muestra {
  -webkit-transform: rotateX(45deg);
  transform: rotateX(45deg);
  background-color: gray;
  margin: 1em auto;
  width: 300px;
  height: 300px;
  color: white;
}
```

## UA 2.17: Introducción CSS: Animaciones CSS



### Transformaciones: 3D en CSS

#### Ejemplos básicos

#### Efecto de Perspectiva



```
<html>
  <head>
    <link rel="stylesheet" type="text/css" href="perspectiva.css" />
  </head>
  <body>
    <div class="main">
      <h3>Efecto de Perspectiva</h3>
      <div class="transform muestra">3D</div>
    </div>
  </body>
</html>
```

```
div {
  box-sizing: border-box;
  padding: .5em;
  font-family: Arial, Helvetica, sans-serif;
  text-align: center;
}
.main {
  font-size: 2em;
  width: 50%;
  height: 100%;
  margin: 0 auto;
  perspective: 300px;
  perspective-origin: left center;
}
.muestra {
  -webkit-transform: rotateX(45deg);
  transform: rotateX(45deg);
  background-color: gray;
  margin: 1em auto;
  width: 300px;
  height: 300px;
  color: white;
}
```



### Transiciones

- Las **transiciones** se basan en un principio muy básico, conseguir un efecto suavizado entre un estado inicial y un estado final. Las propiedades relacionadas que existen son las siguientes:

Propiedades	Valor
transition-property	all   none   <u>propiedad</u>
transition-duration	0   <u>tiempo</u>
transition-timing-function	ease   linear   ease-in   ease-out   ease-in-out   cubic-bezier( <u>A</u> , <u>B</u> , <u>C</u> , <u>D</u> )
transition-delay	0   <u>tiempo</u>

### Transiciones

- **transition-property**: La propiedad de propiedad de transición especifica el nombre de la propiedad de CSS para la cual es el efecto de transición (el efecto de transición comenzará cuando cambie la propiedad de CSS especificada).  
Podemos especificar la propiedad concreta (***width o color***, por ejemplo) o simplemente especificar ***all*** para que se aplique a todos los elementos con los que se encuentre. Por otro lado, ***none*** hace que no se aplique ninguna transición.
- **transition-duration**: La propiedad de duración de transición especifica cuántos segundos o milisegundos (ms) tarda un efecto de transición en completarse. Para ello, especificaremos la **duración de la transición**, desde el inicio de la transición, hasta su finalización.

**Nota:** Se recomienda siempre comenzar con valores cortos, para que las transiciones sean rápidas y elegantes. Si establecemos una duración demasiado grande, el navegador realizará la transición con detención intermitentes, lo que hará que la transición vaya a golpes. Además, transiciones muy largas pueden resultar molestas a muchos usuarios.

### Transiciones

- **transition-timing-function**: La propiedad especifica la curva de velocidad del efecto de transición. Esta propiedad permite un efecto de transición para cambiar la velocidad durante su duración. Los valores que puede tomar la propiedad son los siguientes:

Valor	Inicio	Transcurso	Final
ease	Lento	Rápido	Lento
linear	Normal	Normal	Normal
ease-in	Lento	Normal	Normal
ease-out	Normal	Normal	Lento
ease-in-out	Lento	Normal	Lento

- Una función de tiempo **linear** siempre es constante, mientras que **ease** comienza suavemente, continua de forma más rápida y termina suavemente de nuevo. **Ease-in** y **ease-out** son variaciones que van más lento al principio o al final, y **ease-in-out** una mezcla de las dos.

## UA 2.17: Introducción CSS: Animaciones CSS

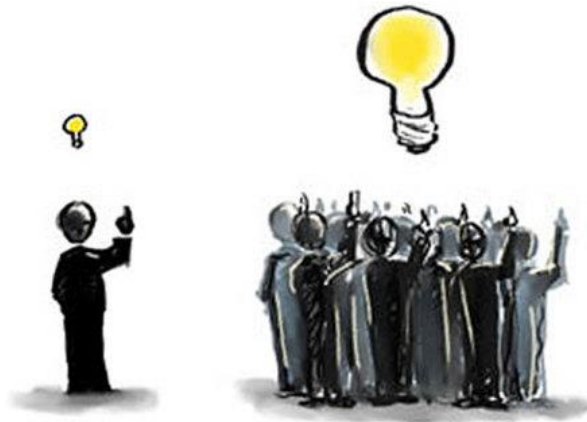


### Transiciones

- **transition-delay**: La propiedad de retardo de transición especifica cuándo se iniciará el efecto de transición. El valor de retardo de transición se define en segundos (s) o milisegundos (ms).

Como siempre, podemos resumir todas estas operaciones en una propiedad de atajo denominada transition. El orden de sus parámetros es el siguiente:

```
div {  
  transition: <property> <duration> <timing-function> <delay>  
}
```



Ejemplos que podéis consultar:  
[https://www.w3schools.com/css/css3\\_transitions.asp](https://www.w3schools.com/css/css3_transitions.asp)

## UA 2.17: Introducción CSS: Animaciones CSS



### Transiciones

#### transition-property y transition-duration

Ejemplo: Desplazarse sobre un elemento <div> y cambiar el ancho y la altura con un efecto de transición suave:

```
<!DOCTYPE html>
<html>
<head>
<style>
div {
  width: 100px;
  height: 100px;
  background: red;
  transition-property: width, height;
  transition-duration: 2s;
}

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

<p><b>Note:</b> 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>
```

## UA 2.17: Introducción CSS: Animaciones CSS



### Transiciones

#### transition-property y transition-duration

Ejemplo: Desplazarse sobre un elemento <div> y cambiar el ancho y la altura con un efecto de transición suave:

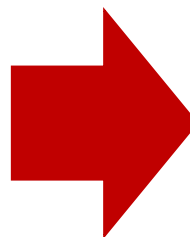
```
div {  
    transition-property: width, height;  
}  
  
div:hover {  
    width: 300px;  
    height: 300px;  
}
```

**Note:** This example does not work in Internet Explorer 9 and earlier versions.



Hover over the div element above, to see the transition effect.

**Note:** This example does not work in Internet Explorer 9 and earlier versions.



Hover over the div element above, to see the transition effect.

## UA 2.17: Introducción CSS: Animaciones CSS



### Transiciones

```
<html>
<head>
<title> Ejemplo Transition</title>
  <meta charset="utf-8">
  <link rel="stylesheet" type="text/css" href="ejemplotransicion.css" />
</head>
<body>
  <h2>CSS transition</h2>
  <div id="mainCont" style="">
    <div class="bloquePub">
      
    </div>
  </div>
</body>
</html>
```

```
/*{
  margin:0;
  padding:0;
  font-family: sans-serif;
}

.mainCont{
  border-style:none;
  border-width:0;
  background-color:white;
}

.bloquePub{
  border-radius: 40px;
  border: 5px solid;
  width: 260px;
  margin:60px 0 0 90px;
  background-color:yellow;
  font-size: 20px; text-align:center;
  padding-top: 20px; word-wrap:break-word;
  transition-property: color, transform;
  transition-duration: 6s, 4s;
  transition-timing-function: ease;
  transition-delay: 2s, 0s;
}

.bloquePub:hover {
  color: red;
  transform:rotate(360deg);
}

h2{
  margin: 15px 0 0 140px;
}
img {
  padding:10px;
}
```

## UA 2.17: Introducción CSS: Animaciones CSS



### Transiciones

CSS transition



Inicial...

CSS transition



Comenzando...

CSS transition



Progresando...

CSS transition



Terminando...

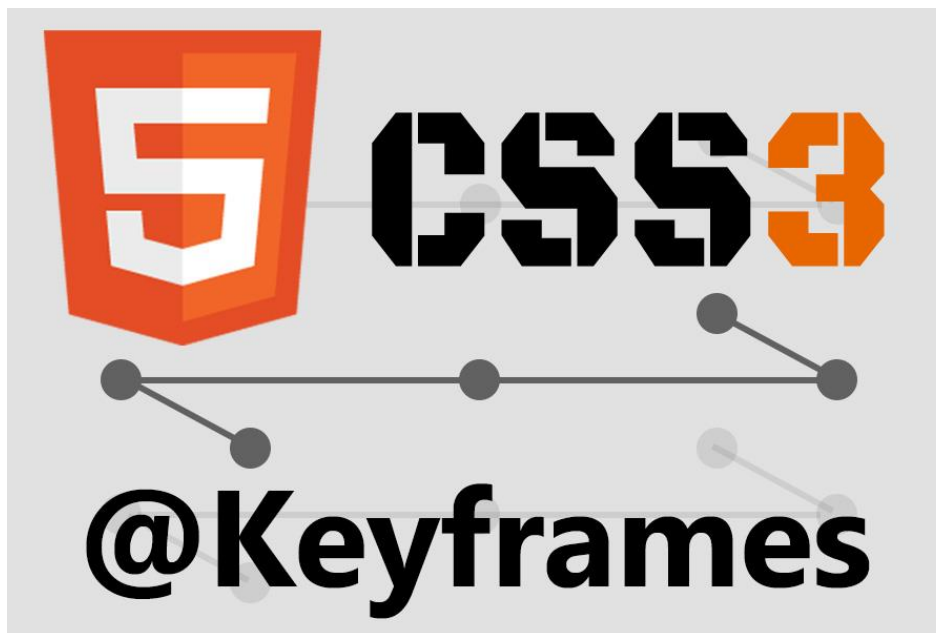


## UA 2.17: Introducción CSS: Animaciones CSS



### Animaciones

- Un movimiento o transformación consiste en ir de una posición o forma inicial a otra final. Estos dos puntos son los que definen la **animación**, son la clave, y se les conoce como tal: puntos clave o **keyframes**.
- Los puntos intermedios son los que se conocen como intercalación.



## UA 2.17: Introducción CSS: Animaciones CSS



### Animaciones

- Las propiedades que tenemos que tener en cuenta son las siguientes:
  - ✓ **Animation-name:** especifica el nombre del keyframe que quiere vincular al selector
  - ✓ **Animation-duration:** especifica cuánto tarda la animación en completarse
  - ✓ **Animation-timing-function:** especifica la curva de velocidad de la animación
  - ✓ **Animation-delay:** especifica un retardo antes del comienzo de la animación
  - ✓ **Animation-iteration-count:** especifica cuantas veces debe de reproducirse la animación.
  - ✓ **Animation-direction:** especifica alternativas al flujo de reproducción de la animación.
  - ✓ **Animation-fill-mode:** especifica valores que se aplicarán a al elemento cuando la reproducción se detenga
  - ✓ **Animation-play-state:** especifica si la reproducción está en marcha o pausada.

## UA 2.17: Introducción CSS: Animaciones CSS



### Animaciones

- Las propiedades que tenemos que tener en cuenta son las siguientes:

Propiedades	Valor
animation-name	<b>none</b>   <i>nombre</i>
animation-duration	<b>0</b>   <i>tiempo</i>
animation-timing-function	<b>ease</b>   linear   ease-in   ease-out   ease-in-out
animation-delay	<b>0</b>   <i>tiempo</i>
animation-iteration-count	<b>1</b>   infinite   <i>número</i>
animation-direction	<b>normal</b>   reverse   alternate   alternate-reverse
animation-fill-mode	<b>none</b>   forwards   backwards   both
animation-play-state	<b>running</b>   paused

## UA 2.17: Introducción CSS: Animaciones CSS



### Animaciones

#### *animation-direction*

Valor	Significado
normal	Los fotogramas se reproducen desde el principio al final.
reverse	Los fotogramas se reproducen desde el final al principio.
alternate	En iteraciones par, de forma normal. Impares, a la inversa.
alternate-reverse	En iteraciones impares, de forma normal. Pares, normal.

#### *animation-timing-function*

Valor	Inicio	Transcurso	Final
ease	Lento	Rápido	Lento
linear	Normal	Normal	Normal
ease-in	Lento	Normal	Normal
ease-out	Normal	Normal	Lento
ease-in-out	Lento	Normal	Lento

- Nota: las propiedades **animation-duration**, **animation-timing-function** y **animation-delay** funcionan exactamente igual que en las transiciones.

```
div {  
  animation: <name> <duration> <timing-function> <delay>  
            <iteration-count> <direction> <fill-mode> <play-state>  
}
```

## UA 2.17: Introducción CSS: Animaciones CSS



### Animaciones: regla @keyframes

- Para realizar animaciones en CSS3, tenemos que utilizar la regla **@keyframes**, la cuál es muy sencilla de utilizar y se basa en el siguiente esquema:

```
@keyframes nombre {  
  selectorkeyframe {  
    propiedad : valor ;  
    propiedad : valor  
  }  
}
```

- En primer lugar elegiremos un **nombre** para la animación con el fin de hacerlo posteriormente referencia, ya que podemos tener varias en una misma página. Tener en cuenta además, que podremos utilizar varios selectores para definir el transcurso de los fotogramas en la animación.

## UA 2.17: Introducción CSS: Animaciones CSS



### Animaciones: regla @keyframes

```
HTML
1 <!-- Learn about this code on MDN:
   https://developer.mozilla.org/es/docs/Web/CSS/CSS_Animations/Usando_animaciones_CSS -->
2
3 <p>The Caterpillar and Alice looked at each other for
   some time in silence:
4 at last the Caterpillar took the hookah out of its
   mouth, and addressed
5 her in a languid, sleepy voice.</p>
6

CSS
1 p {
2   animation-duration: 3s;
3   animation-name: slidein;
4   animation-iteration-count: infinite;
5 }
6 @keyframes slidein {
7   from {
8     margin-left: 100%;
9     width: 300%;
10  }
11
12  to {
13    margin-left: 0%;
14    width: 100%;
15  }
16 }
17
```

Este sencillo ejemplo da estilos al elemento **<p>** para que el texto se deslice por la pantalla entrando desde el borde derecho de la ventana del navegador pero de manera infinita.

## UA 2.17: Introducción CSS: Animaciones CSS



### Animaciones: regla @keyframes

```
HTML
1 <!-- Learn about this code on MDN:
   https://developer.mozilla.org/es/docs/Web/CSS/CSS_Animations/Usando_animaciones_CSS -->
2
3 <p>The Caterpillar and Alice looked at each other for
   some time in silence:
4 at last the Caterpillar took the hookah out of its
   mouth, and addressed
5 her in a languid, sleepy voice.</p>

CSS
1 p {
2   animation-duration: 3s;
3   animation-name: slidein;
4 }
5
6 @keyframes slidein {
7   from {
8     margin-left: 100%;
9     width: 300%
10  }
11
12   to {
13     margin-left: 0%;
14     width: 100%;
15  }
16 }
17
```

Este sencillo ejemplo da estilos al elemento **<p>** para que el texto se deslice por la pantalla entrando desde el borde derecho de la ventana del navegador.

## UA 2.17: Introducción CSS: Animaciones CSS



### Animaciones: regla @keyframes

```
<html>
  <head>
    <link rel="stylesheet" type="text/css" href="animabola1.css"/>
  </head>
  <body>
    <h2>Animando un círculo</h2>
    <div>
    <br></div>
  </body>
</html>
```



### Animando un círculo



Dirección  
del Movimiento



```
body {
  font-family: Arial;
  text-align: center;
}
h2 {
  font-size: 64px; /*16px son 1rem*/
}
div {
  width: 200px;
  height: 200px;
  background-color: bisque;
  position: relative;
  border-radius: 50%;
  animation-name: primeranimation;
  animation-duration: 5s;
  animation-timing-function: linear;
  animation-fill-mode: forwards;
}
img {
  width: 100px;
  height: 100px;
}

@keyframes primeranimation {
  from {
    left: 0;
  }
  to {
    left: calc(100% - 200px);
  }
}
```







**Universidad  
Europea**