

A decorative graphic consisting of a thin brown circle on the left side. A thick black left square bracket is positioned to the left of the circle's center. A thick brown right square bracket is positioned to the right of the circle's center. A horizontal bar with a light olive green gradient extends from the left edge of the slide, passing through the circle and ending at the right bracket.

# Estilos avanzados

CSS3

# [ CSS3 ]

- Bordes
  - Esquinas redondeadas
  - Bordes con imagen.
- Sombras
- Fondos
- Textos
  - Sombras
  - Envoltura
- Fuentes
- Transformaciones
  - 2D
  - 3D
- Transiciones
- Transiciones 3D. Efectos
- Animaciones





Bordes CSS3

# [ Esquinas redondeadas ]

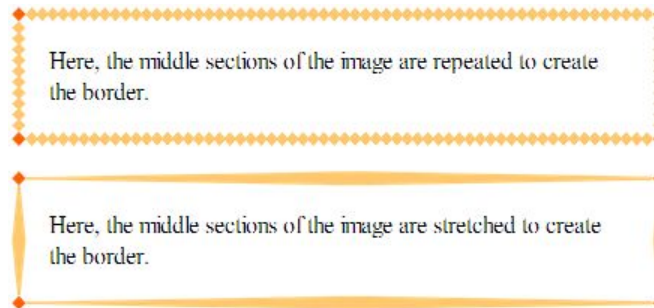
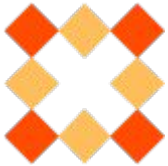
- ```
div
{
border:2px solid;
border-radius:25px;
-moz-border-radius:25px;
}
```

This box has rounded corners!

# [ Borde con imagen ]

*Sintaxis:* border-image:url(imagen) x y repetir;  
(*repetir:* stretch, repeat, round)

```
■ div{  
    border-width: 2px;  
    border-image:url(border.png) 30 30 round;  
    -moz-border-image:url(border.png) 30 30 round;  
    -webkit-border-image:url(border.png) 30 30 round;  
    -o-border-image:url(border.png) 30 30 round;  
}
```



# [ Sombra ]

- div  
`{box-shadow: 10px 10px 5px #888888;}`



- Para que la sombra sea interna al elemento hay que utilizar **inset**.  
Ej: `{box-shadow: inset 10px 10px 5px #888888;}`

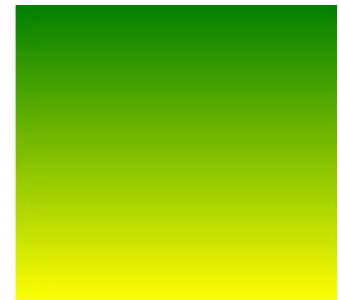


# Fondos CSS3

- Gradientes
- Imágenes

# [ Fondo color gradientes ]

```
#grad1 {  
  height: 200px;  
  background-color: red;  
  /* Para navegadores sin soporte para gradientes */  
  background-image: linear-gradient(green, yellow);  
  /* Sintaxis estándar (al final) */  
}
```



[https://www.w3schools.com/css/css3\\_gradients.asp](https://www.w3schools.com/css/css3_gradients.asp)



# [ Fondo imagen ]

Redimensionar la imagen de fondo:

```
■ div{  
    background:url(img_flwr.gif);  
    background-size:80px 60px;  
    background-repeat:no-repeat;  
}
```

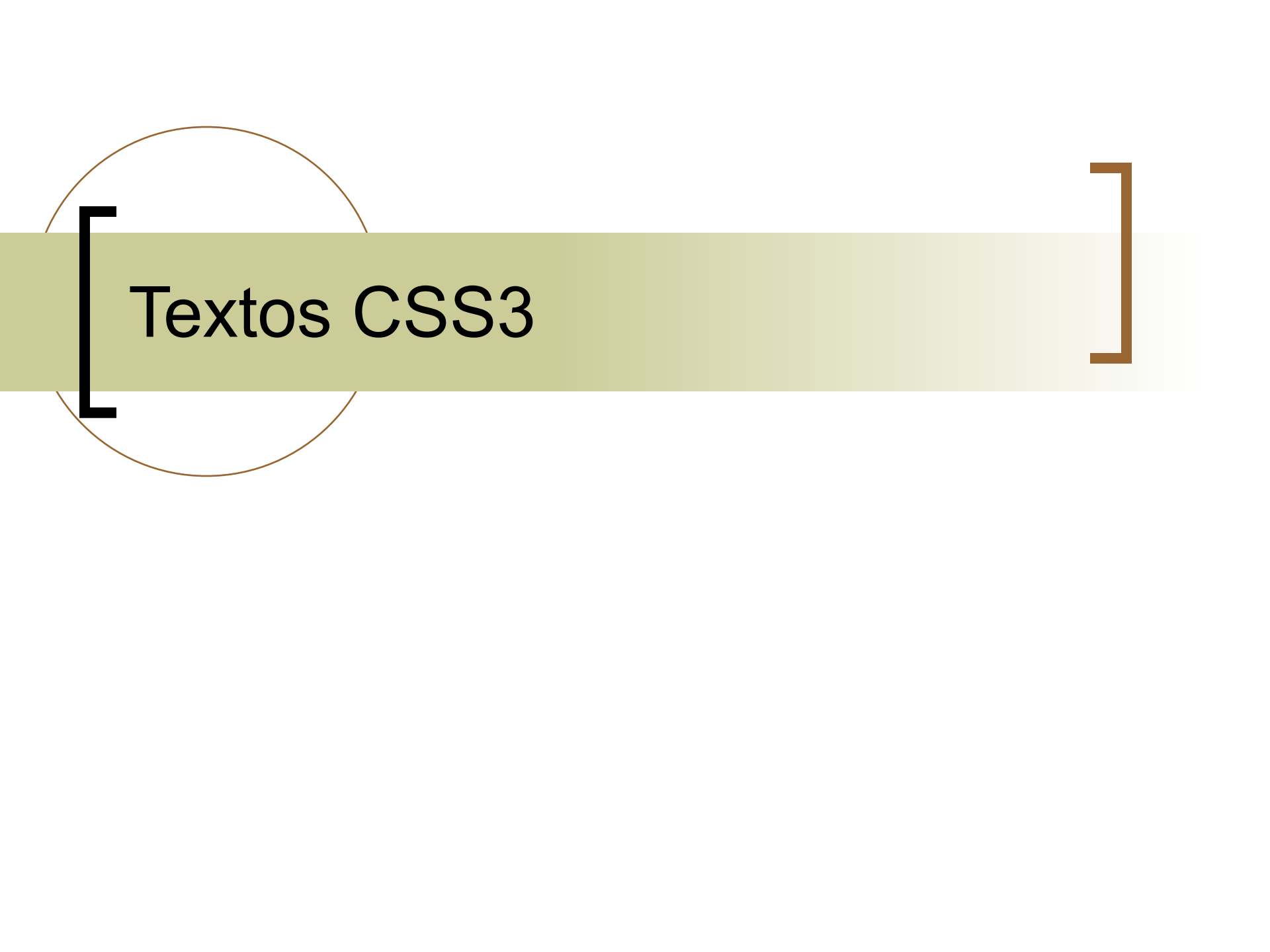
# Fondo imagen

- Posicionamiento imagen de fondo.
- La propiedad background-origin especifica el área de posicionamiento de las imágenes de fondo.

```
div{  
    background:url(img_flwr.gif);  
    background-repeat:no-repeat;  
    background-size:100% 100%;  
    background-origin:content-box;  
}
```

# [ Fondo imagen ]

- CSS3 permite el uso de varias imágenes de fondo de un elemento.
- ```
body{  
background-image:url(img_flwr.gif),url(img_tree.gif);  
}
```



Textos CSS3

# Sombra en texto

- Se especifica la sombra horizontal, la sombra vertical, la distancia de desenfoque, y el color de la sombra:
- ```
h1
{
text-shadow: 5px 5px 5px #44ff00;
}
```

# [ Wrap (envoltura) ]

- Si una palabra es demasiado larga para caber dentro de un área, se expande fuera.
- ```
p {  
    word-wrap:break-word;  
}
```

A decorative graphic consisting of a thin brown circle on the left and a horizontal bar extending to the right. The bar has a light beige gradient and is framed by a thick black left bracket and a thick brown right bracket.

# Fuentes CSS3

# [ @font-face ]

- Permite que la fuente no tenga por qué estar instalada en el ordenador.
- ```
@font-face{
font-family: myFirstFont;
src: url('Sansation_Light.ttf'),
      url('Sansation_Light.eot'); /* IE9+ */
}

div{
font-family:myFirstFont;
}
```



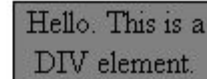
# [ @font-face (bold) ]

- @font-face
  - {
  - font-family: myFirstFont;
  - src: url('Sansation\_Bold.ttf'),  
url('Sansation\_Bold.eot');
  - font-weight:bold;
  - }



# Transformaciones 2D CSS3

# [ Rotate ]



Hello. This is a  
DIV element.



Hello. This is a  
DIV element.

```
<!DOCTYPE html>
<html>
<head>
</head>
<body>

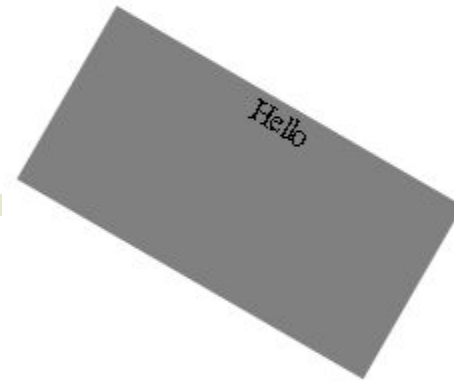
<center><div>Hello. This is a
  DIV
  element.</div></center>

<center><div id="div2">Hello.
  This is a DIV
  element.</div></center>

</body>
</html>
```

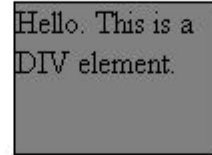
```
div
{
width:100px;
height:75px;
background-color:grey;
border:1px solid black;
}
div#div2
{
transform:rotate(30deg);
-ms-transform:rotate(30deg); /* IE 9 */
-moz-transform:rotate(30deg); /* Firefox */
-webkit-transform:rotate(30deg); /* Safari and Chrome */
-o-transform:rotate(30deg); /* Opera */
}
```

# [ Rotate



```
<!DOCTYPE html>  div
<html>           {
<head>           width:200px;
</head>          height:100px;
<body>           background-color:grey;
<div>Hello</div> /* Rotate div */
</body>          transform:rotate(30deg);
</html>          -ms-transform:rotate(30deg); /* IE 9 */
                  -moz-transform:rotate(30deg); /* Firefox */
                  -webkit-transform:rotate(30deg); /* Safari and
                  Chrome */
                  -o-transform:rotate(30deg); /* Opera */
                  }
```

# [ Translate



Hello. This is a  
DIV element.

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<style>
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<div>Hello. This is a DIV  
  element.</div>
```

```
<div id="div2">Hello. This is a DIV  
  element.</div>
```

```
</body>
```

```
</html>
```

```
div
```

```
{
```

```
width:100px;
```

```
height:75px;
```

```
background-color:grey;
```

```
border:1px solid black;
```

```
}
```

```
div#div2
```

```
{
```

```
transform:translate(50px,100px);
```

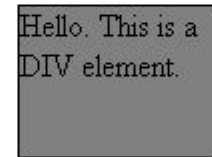
```
-ms-transform:translate(50px,100px); /* IE 9 */
```

```
-moz-transform:translate(50px,100px); /* Firefox */
```

```
-webkit-transform:translate(50px,100px); /* Safari and Chrome */
```

```
-o-transform:translate(50px,100px); /* Opera */
```

```
}
```



Hello. This is a  
DIV element.

# [ Scale

```
<!DOCTYPE html>
<html>
<head>
</head>
<body>
```

```
<div>Hello. This is a DIV
  element.</div>
```

```
<div id="div2">Hello. This is a DIV
  element.</div>
```

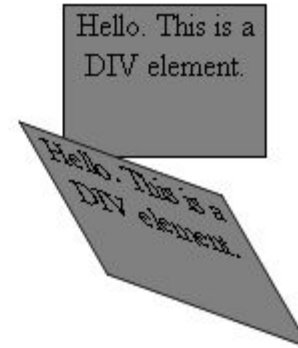
```
</body>
</html>
```

```
div
{
width:100px;
height:75px;
background-color:grey;
border:1px solid black;
}
div#div2
{
margin:100px;
transform:scale(2,4);
-ms-transform:scale(2,4); /* IE 9 */
-moz-transform:scale(2,4); /* Firefox */
-webkit-transform:scale(2,4); /* Safari and Chrome */
-o-transform:scale(2,4); /* Opera */
}
```

Hello. This is a  
DIV element.

Hello. This is a  
DIV element.

# [ Skew ]



```
<!DOCTYPE html>
<html>
<head>
</head>
<body>
```

```
<div>Hello. This is a DIV
  element.</div>
```

```
<div id="div2">Hello. This is a DIV
  element.</div>
```

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

```
div
{
width:100px;
height:75px;
background-color:grey;
border:1px solid black;
}
div#div2
{
transform:skew(30deg,20deg);
-ms-transform:skew(30deg,20deg); /* IE 9 */
-moz-transform:skew(30deg,20deg); /* Firefox */
-webkit-transform:skew(30deg,20deg); /* Safari and Chrome */
-o-transform:skew(30deg,20deg); /* Opera */
}
```



# Transformaciones 3D

CSS3



# [ RotateX ]

```
<!DOCTYPE html>
<html>
<head>
<style>
</style>
</head>
<body>

<div>Hello. This is a DIV
  element.</div>

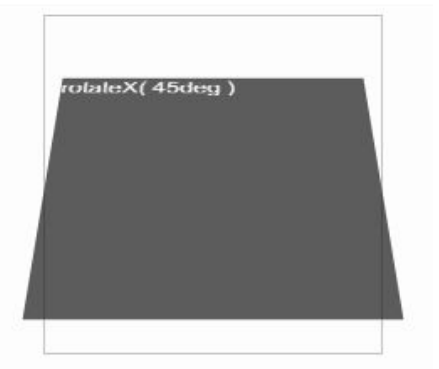
<div id="div2">Hello. This is a
  DIV element.</div>

</body>
</html>
```

```
div
{
width:100px;
height:75px;
background-color:grey;
border:1px solid black;
}
div#div2
{
transform:rotateX(40deg);
-webkit-transform:rotateX(40deg);
-moz-transform:rotateX(40deg);
}
```

Hello. This is a  
DIV element.

Hello. This is a  
DIV element.



# [ RotateY ]

```
<!DOCTYPE html>
<html>
<head>
<style>
</style>
</head>
<body>

<div>Hello. This is a DIV
  element.</div>

<div id="div2">Hello. This is a
  DIV element.</div>

</body>
</html>
```

```
div
{
width:100px;
height:75px;
background-color:grey;
border:1px solid black;
}
div#div2
{
transform:rotateY(130deg);
-webkit-transform:rotateY(130deg);
-moz-transform:rotateY(130deg);
}
```

# [ RotateZ ]

```
<!DOCTYPE html>
<html>
<head>
<style>
</style>
</head>
<body>

<div>Hello. This is a DIV
  element.</div>

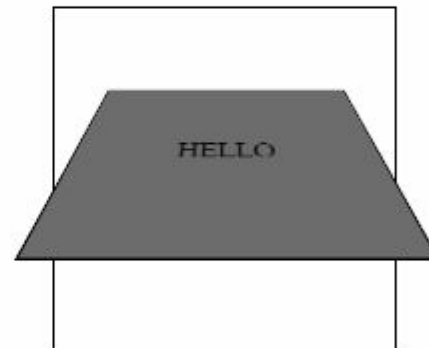
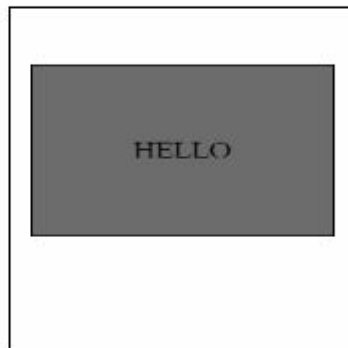
<div id="div2">Hello. This is a
  DIV element.</div>

</body>
</html>
```

```
div
{
width:100px;
height:75px;
background-color:grey;
border:1px solid black;
}
div#div2
{
transform:rotateZ(130deg);
-webkit-transform:rotateZ(130deg);
-moz-transform:rotateZ(130deg);
}
```

# [ Perspectiva ]

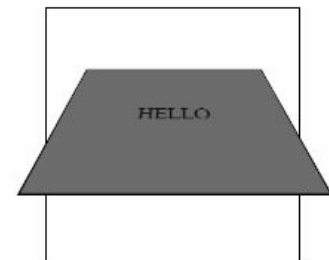
- La propiedad perspective establece la vista en perspectiva de un elemento.
- Se aplica a un contenedor y afecta a su contenido.
- Solo afecta a elementos con transformaciones 3D.



# [ Perspectiva: ejemplo ]

```
<html>
<head>
<style>
#div1 {
  position: relative;
  height: 150px;
  width: 150px;
  margin: 50px;
  padding: 10px;
  border: 1px solid black;
  perspective: 150px;
}
```

```
#div2 {
  padding: 50px;
  position: absolute;
  border: 1px solid black;
  background-color: red;
  -webkit-transform: rotateX(45deg);
  transform: rotateX(45deg);
}
</style>
</head>
<body>
<div id="div1">
  <div id="div2">HELLO</div>
</div>
</body>
</html>
```



# [ Perspectiva ]

- La propiedad perspective-origin establece el punto de fuga en la vista en perspectiva de un elemento.
- Ej: perspective-origin: 25% 75%;





# Transiciones CSS3

# [Transiciones]

- Las transiciones CSS3 son efectos que se aplican en el cambio gradual de un estilo a otro.
- Hay que especificar al menos:
  - La propiedad CSS a la cual se le quiere aplicar el efecto.
  - La duración del efecto (si no se especifica no se realizará, porque el valor por defecto es 0).



# [Transiciones]

## ■ Propiedades:

- Transition-property: Especifica las propiedades a la que se aplicará el efecto.
- Transition-duration: Define la duración de la transición.
- Transition-timing-function: Describe la función de tiempo que seguirá la transición:
  - Linear, ease, ease-in, ease-in-out,...
- Transition-delay: Retardo de tiempo en la aplicación de la transición.

# [Transiciones]

- No todas las propiedades pueden utilizarse en una transición.
- Lista de propiedades que se pueden modificar mediante transiciones:
  - background-color (color)
  - background-image (solo gradientes)
  - background-position (porcentaje y longitud)
  - border-bottom-color (color)
  - border-bottom-width (longitud)
  - border-color (color)
  - border-left-color (color)
  - border-left-width (longitud)
  - border-right-color (color)
  - border-right-width (longitud)
  - border-spacing (longitud)
  - border-top-color (color)
  - border-top-width (longitud)

# [Transiciones]

- border-width (longitud)
- bottom (longitud y porcentaje)
- color (color)
- crop (rectángulo)
- font-size (longitud y porcentaje)
- font-weight (número)
- grid-\* (diversos valores)
- height (longitud y porcentaje)
- left (longitud y porcentaje)
- letter-spacing (longitud)
- line-height (número, longitud y porcentaje)
- margin-bottom (longitud)
- margin-left (longitud)
- margin-right (longitud)
- margin-top (longitud)
- max-height (longitud y porcentaje)
- max-width (longitud y porcentaje)

# [Transiciones]


- min-height (longitud y porcentaje)
- min-width (longitud y porcentaje)
- opacity (número)
- outline-color (color)
- outline-offset (entero)
- outline-width (longitud)
- padding-bottom (longitud)
- padding-left (longitud)
- padding-right (longitud)
- padding-top (longitud)
- right (longitud y porcentaje)
- text-indent (longitud y porcentaje)
- text-shadow (sombra)
- top (longitud y porcentaje)
- vertical-align (palabras clave, longitud y porcentaje)
- visibility (visibilidad)
- width (longitud y porcentaje)
- word-spacing (longitud y porcentaje)
- z-index (entero)
- zoom (número)

# [Transiciones]

- Ejemplo declaración de transición:
  - `transition-property: all;`  
`transition-duration: 0.5s;`  
`transition-timing-function: ease;`  
`transition-delay: 0s;`
  - La transición afectará a todas las propiedades.

# [Transiciones]

- En este ejemplo se definen valores diferentes para cada propiedad
  - transition-property: opacity, left, top;  
transition-duration: 0.5s, 0.8s, 0.1s;  
transition-timing-function: ease, linear, ease;  
transition-delay: 0s, 0s, 1s;

A decorative graphic consisting of a thin brown circle on the left side. A thick black left square bracket is positioned to the left of the circle, and a thick brown right square bracket is positioned to the right of the circle. A horizontal bar with a light olive green to white gradient passes through the center of the circle and between the brackets.

# Transiciones 3D

Ejemplos

# [ Ejemplos: efecto flip ]

- Estructura HTML

```
<section class="container">
  <div id="card">
    <figure class="front">1</figure>
    <figure class="back">2</figure>
  </div>
</section>
```

- CSS para crear el espacio 3D.

```
.container {
  width: 200px;
  height: 260px;
  position: relative;
  perspective: 800px;
}
```



# [ Ejemplos: efecto flip ]

- CSS para el elemento que se va a transformar:

```
#card {  
  width: 100%;  
  height: 100%;  
  position: absolute;  
  transform-style: preserve-3d;  
  transition: transform 1s;  
}
```

# [ Ejemplos: efecto flip ]

- CSS para los elementos (dos caras) que componen el elemento a voltear:

```
#card figure {  
  display: block;  
  position: absolute;  
  width: 100%;  
  height: 100%;  
  backface-visibility: hidden;  
}
```

# [ Ejemplos: efecto flip ]

- CSS para las dos caras:

```
#card .front {  
    background: red;  
}  
#card .back {  
    background: blue;  
    transform: rotateY( 180deg );  
}
```

# [ Ejemplos: efecto flip ]

- Posibilidades para realizar la transición:
  - Añadir y quitar un clase al elemento #card (.flipped)

```
#card.flipped {  
    transform: rotateY( 180deg );  
}
```

- Utilizar el pseudoselector :hover

```
#card:hover {  
    transform: rotateY( 180deg );  
}
```

A decorative graphic consisting of a thin brown circle on the left and a horizontal bar with a light beige-to-white gradient on the right. A large black left square bracket is positioned on the left side of the bar, and a brown right square bracket is on the right side.

# Animaciones CSS3

# [Propiedades]

- **animation-name:**

- En esta propiedad debemos indicar el **nombre de la animación** a la que hacemos referencia.

- **animation-duration:**

- En esta propiedad especificamos el **número de segundos** o milisegundos de la animación, aceptando valores decimales. Siendo el valor por defecto 0, es decir sin animación.

- Ej: **animation-duration: 1s;**  
**animation-duration: 0.1s;**  
**animation-duration: 1.7ms;**

# [Propiedades]

## ■ **animation-iteration-count:**

- En esta propiedad debemos indicar cuantas veces queremos que se repita nuestra animación. Aceptando números enteros y la palabra **infinite** para que no deje de repetirse nunca.

## ■ **animation-direction:**

- Desde esta propiedad podemos definir si la dirección de nuestra animación es normal o por el contrario preferimos que sea a la inversa (reverse) en ciclos alternos (alternate).

# [Propiedades]

- **animation-timing-function:**
  - Linear, ease, ease-in, ease-in-out,...
- **animation-fill-mode:**
  - Permite definir el estilo que permanecerá al final la animación.
    - Forwards
    - Backwards
    - Both
    - none



# [Propiedades]

- **animation-delay:**

- En esta propiedad podemos definir en segundos o milisegundos con que retardo queremos activar nuestra animación.

- **animation-play-state:**

- Con esta propiedad podemos pausar (paused) y volver a poner en marcha nuestra animación (running).

# [@keyframes]

- Permite definir los fotogramas clave de una animación.
- Definición:
  - @-webkit-keyframes nombre-de-la-animacion {  
 from {  
 propiedades fotograma clave inicial;  
 }  
 to {  
 propiedades fotograma clave final;  
 }  
}

# [@keyframes]

- También se pueden definir fotogramas clave intermedios mediante porcentaje.
- Definición:
  - @-webkit-keyframes nombre-de-la-animacion  
{  
  0% { propiedades; }  
  20% { propiedades; }  
  63% { propiedades; }  
  ...Tantos como queramos...  
  100% { propiedades; }  
}