

Unidad 3: Propiedades Avanzadas

Contenido

| | |
|-----------------------------|----|
| Propiedades Avanzadas | 2 |
| @Font-face..... | 2 |
| Border-radius | 2 |
| Text-shadow..... | 4 |
| Box-shadow..... | 4 |
| Gradient | 4 |
| Opacity..... | 5 |
| Transform | 6 |
| Background | 8 |
| Background Repeat | 8 |
| Background Position | 8 |
| Background Size | 9 |
| Position | 10 |
| Position Static..... | 10 |
| Position Relative | 10 |
| Position Absolute..... | 11 |
| Position Fixed | 11 |
| z-index..... | 12 |

Propiedades Avanzadas

@Font-face

Para poder utilizar una fuente por fuera de las fuentes seguras (Arial, Times New Roman, Comic Sans, Tahoma, Verdana, etc), que son fuentes que siempre están instaladas en las computadoras de los usuarios, tenemos que tener nuestra fuente convertida a estos formatos:

- TTF (Chrome, Safari, Firefox, Opera)
- EOT (Explorer 6, 7 y 8)
- WOFF (Explorer 9, Firefox 3.6+, Chrome 5+)
- SVG (Opera, Chrome, iPhone iOS)

Una vez convertida, hay que crear un @font-face como primera instancia.

```
@font-face {  
  font-family: 'robotoregular';  
  src: url('Roboto.eot');  
  src: url('Roboto.svg');  
  src: url('Roboto.ttf');  
  src: url('Roboto.woff');  
}
```

Dentro de la función @font-face, hay que tener una propiedad font-family, con el nombre de la fuente entre comillas simples. Y un src, con la ubicación de cada archivo de la fuente en cada formato.

Una vez definida una fuente con @font-face, podemos usarla en cualquier etiqueta o clase, usando la propiedad font-family usando el nombre de la fuente establecido anteriormente entre comillas simples.

```
body{  
  font-family: 'robotoregular';  
}
```

Existen herramientas que permiten convertir las fuentes a estos formatos, como por ejemplo [Font-squirrel](#) o y también repositorios de fuentes @font-face, como el caso de google font, en donde directamente importamos la fuente y la referenciamos.

Border-radius

Esta propiedad permite agregarle a cualquier caja o imagen un borde redondeado. Si tiene un solo valor afecta a los cuatro lados. Por ejemplo

article{

```
border-radius: 20px;  
}
```



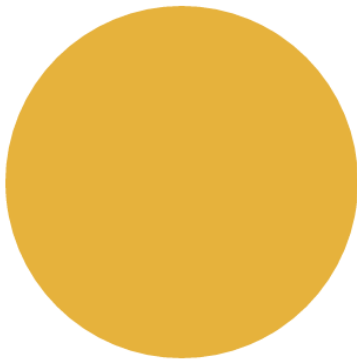
Si tiene varios valores, cada lado, puede ser independiente en su redondeado e incluso no tener redondeado como en el siguiente ejemplo:

```
article{  
border-radius: 150px 0 150px 0;  
}
```



Si queremos hacer un círculo, podemos hacerlo con border-radius, con el valor 50%.

```
article{  
border-radius: 50%;  
}
```



Text-shadow

La propiedad text-shadow, permite aplicar una sombra a un texto. Posee 4 valores, que representan la distancia en x, en y, la nitidez de la sombra, y el color expresado en rgba (para darle opacidad).

Sombra en Texto

```
h1{  
  text-shadow: 1px 1px 1px  
    rgba(0, 0, 0, 0.5);  
}
```

Box-shadow

Esta propiedad permite darle sombra a las cajas o a imágenes.

Puede tomar cinco parámetros: el color, el desplazamiento horizontal, el desplazamiento vertical, el valor de difuminación, y la palabra clave inset para generar una sombra interna.

```
article{  
  box-shadow: 10px 10px 0px -2px rgba(0,0,0,0.2);  
}
```

Gradient

Existen dos funciones para generar gradientes, linear-gradient y radial-gradient.

1. La función linear-gradient(posición inicio, color inicial, color final) puede ser aplicada a las propiedades background o background-image para generar un

gradiente lineal.

- Los atributos indican el punto inicial y los colores usados para crear el gradiente.
- El primer valor puede ser especificado en pixeles, en porcentaje o usando las palabras clave top, bottom, left y right.

```
body{
```

```
background: linear-gradient
```

```
(to bottom, rgba(147,206,222,1) 0%,
```

```
rgba(117,189,209,1) 41%,      rgba(73,165,191,1) 100%);
```

```
}
```

2. La función radial-gradient(posición inicio, forma, color inicial, color final) puede ser aplicada a las propiedades background o background-image para generar un gradiente radial.

- La posición de inicio es el origen y puede ser declarado en pixeles, porcentaje o como una combinación de las palabras clave center, top, bottom, left y right.

Para ambos tipos de gradientes, existen generadores visuales como

<https://www.cssmatic.com/es/gradient-generator>

Opacity

La propiedad opacity, permite modificar la opacidad de una caja u elemento. Va en valores decimales entre 0 y 1, siendo 0 sin opacidad y 1 opacidad al 100%. El valor por defecto de opacity es 1.

```
article{
```

```
opacity: 0.5; // en este ejemplo, la opacidad está al 50%.
```

```
}
```

Probando Opacidad de un elemento

Todo el elemento tendrá menor opacidad

Probando Opacidad de un elemento

Todo el elemento tendrá menor opacidad

Transform

Esta propiedad modifica la forma de un elemento. Utiliza cuatro funciones básicas: scale (escalar), rotate (rotar), skew (inclinarse), y translate (trasladar o mover).

- La función scale recibe solo un parámetro. Un valor negativo invierte el elemento, valores entre 0 y 1 reducen el elemento y valores mayores que 1 expanden el elemento (por ejemplo, transform: scale(1.5);).
- La función rotate usa solo un parámetro expresado en grados para rotar el elemento (por ejemplo, transform: rotate(20deg);).
- La función skew recibe dos valores, también en grados, para la transformación horizontal y vertical (por ejemplo, transform: skew(20deg, 20deg);).
- La función translate mueve el objeto tantos píxeles como sean especificados por sus parámetros (por ejemplo, transform: translate(20px);).

article{

transform: scaleX(1);

}

Scale (en decimales expresados con valores numéricos separados por decimales expresados con .)

Escala de 100%, que es el tamaño normal de un objeto, es 1.

Podemos escalar, pasando dos valores, eso representan x e y, o especificar en qué eje queremos escalar.

- transform:scale(x,y)
- transform:scale3d(x,y,z)
- transform:scaleX()
- transform:scaleY()
- transform:scaleZ()

Translate (se expresa en px o ems)

- transform:translate(x,y)
- transform:translate3D(x,y,z)
- transform:translateX()
- transform: translateY()
- transform:translateZ()

Skew (se expresa en deg)

- `transform:skew(x,y)`
- `transform:skewX()`
- `transform: skewY()`

Rotate (se expresa en deg)

- `transform:rotate(x,y)`
- `transform:rotateX()`
- `transform: rotateY()`

Background

Background Repeat

La propiedad repeat, sirve para controlar la repetición de un fondo. Por defecto tiene el valor repeat, con lo cual si tenemos un fondo de un div, o de un body y ese fondo es de menor tamaño se repite como retícula.

Los otros valores que puede tomar la propiedad repeat son:

- **repeat-none**
`background-repeat: repeat-none;`
Esto hace que no se repita. Es decir se muestra una sola vez
- **repeat-x**
`background-repeat: repeat-x;`
Repite el fondo solo en el eje x (es decir horizontalmente)
- **repeat-y**
`background-repeat: repeat-y;`
Repite el fondo solo en el eje y (es decir verticalmente)

Más info: http://www.w3schools.com/cssref/pr_background-repeat.asp

Background Position

La propiedad background position, establece la posición inicial de una imagen de fondo

Tiene dos valores que representan el eje x y el eje y

Estos valores pueden estar expresados en píxeles, porcentajes o valores en palabras.

Los valores posibles para el eje horizontal(x) son:

- left
- center
- right

Los valores posibles para el eje vertical(y) son:

- top
- center
- bottom

Ejemplos de combinaciones:

- `background-position: center top;`
- `background-position: left center;`
- `background-position: 30% top;`

- `background-position: center -100px;`

Más info: http://www.w3schools.com/cssref/pr_background-position.asp

Background Size

La propiedad background-size, nos permite controlar el tamaño de un fondo. Por defecto está en auto, es decir que un fondo se muestra en el tamaño que tiene la imagen. Con la propiedad background-size, podemos hacer que un fondo se adapte al ancho, al alto o al ancho y alto (acá se deformaría) de un contenedor.

Los otros valores que puede tomar la propiedad background-size son:

- **cover**
`background-size: cover;`
El fondo se adapta al ancho del contenedor ocupando todo su ancho, el alto es proporcional.
- **contain**
`background-size: contain;`
El fondo se adapta al alto del contenedor ocupando todo su alto, el ancho es proporcional.
- **porcentaje**
`background-size: 100% 100%;`
Se puede también establecer el tamaño en porcentajes o en píxeles, el primer valor representa el ancho y el segundo el alto. En este ejemplo, el valor representa que ocupa todo el ancho y todo el alto del contenedor, deformando el fondo (no mantiene la proporción de origen).

Más info: http://www.w3schools.com/cssref/css3_pr_background-size.asp

Position

Permite ubicar un elemento en un lugar exacto de la pantalla.

Valores posibles:

- static
- relative
- absolute
- fixed

Position Static

- Es el valor por defecto que tienen todos los elementos.
- Significa que ese elemento no está posicionado de ninguna manera, sino que está en su ubicación por defecto.

Position Relative

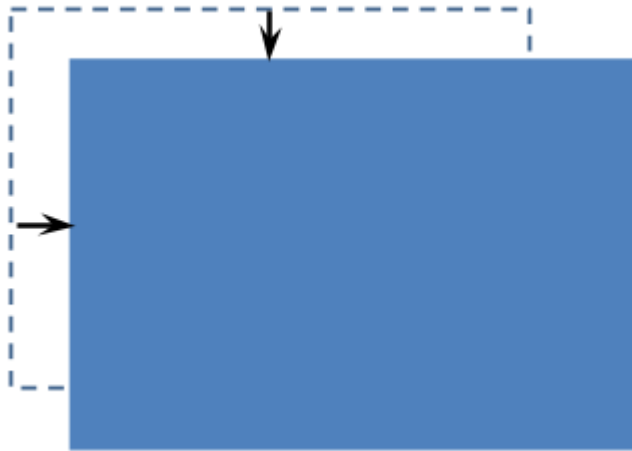
- Posiciona el elemento a partir de su posición original.
- Permite mover ese elemento desde su ubicación original, a la izquierda, derecha, arriba o abajo.

Para mover un elemento a la izquierda, arriba, abajo o derecha de su posición original, si ya tiene position relative, uso:

- top
- left
- right
- bottom

Ejemplo:

```
article{  
    position: relative;  
    top: 5px;  
    left: 5px;  
}
```



Position Absolute

Un elemento posicionado absolutamente está posicionado relativamente a otro elemento, llamado bloque contenedor.

Por defecto es el body, pero puede estar dentro de otro elemento con position relative.

Los elementos con position absolute, salen del flujo del html y pierden su width de 100%, se comportan como si .

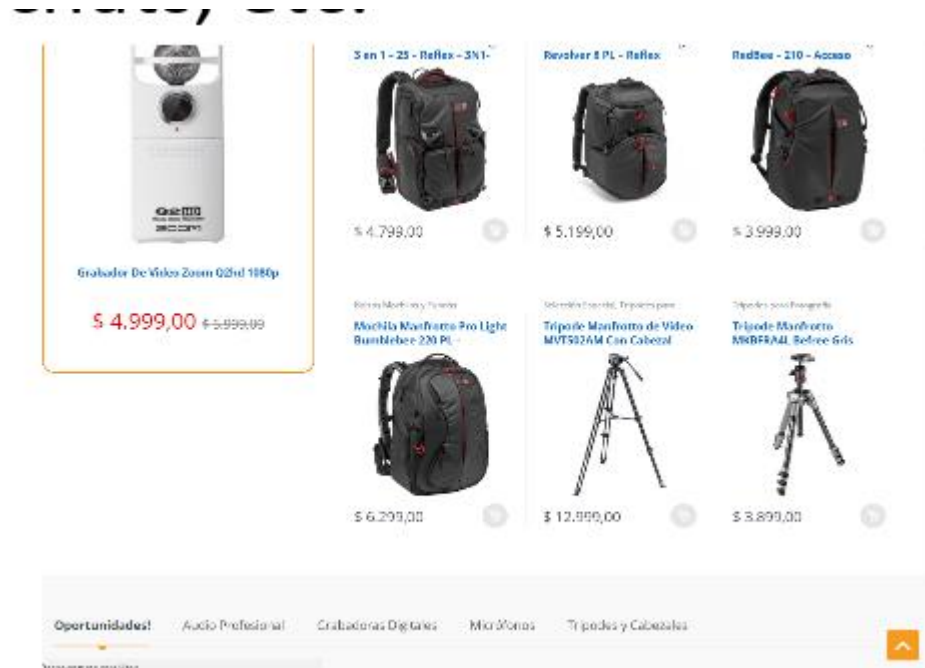
Para moverse en forma absoluta, también se utilizan las propiedades: top, left, right, bottom

Position Fixed

Propiedad que sirve para posicionar elementos en forma fija, relativa a la pantalla.

Si hago scroll, igual mantiene su posición.

Suele usarse para botón ir arriba, chats, etc.



z-index

Propiedad que sirve el orden en la superposición de elementos

Los valores posibles son numéricos.

El elemento que tenga el valor más alto de z-index, se pondrá por arriba del resto.

Por Ejemplo: un elemento con z-index:2; se posicionará por arriba de un elemento con z-index:1;

Solo los elementos html, que tengan position relative, absolute, o fixed, tienen la posibilidad de usar esta propiedad.