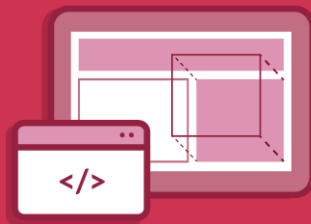


Diseño WEB

Clase 7

Animaciones, Transformaciones y Transiciones



Gradientes

CSS3 ha agregado la opción de **crear gradientes (fondos en degradé) sin la necesidad de usar imágenes.**

Los gradientes en CSS son de **dos tipos: lineales (*linear-gradient*) o radiales (*radial-gradient*)**. En el gradiente lineal, la transformación de color va avanzando línea a línea, mientras que en el radial, la transformación de color se produce debido a que sucesivos círculos concéntricos van cambiando de color.

La propiedad que utilizamos para realizar gradientes lineales, se llama *linear-gradient* y esta se agrega al atributo *background*, que ya vimos con anterioridad.

Esta propiedad maneja dos opciones de parámetros, podemos elegir el punto de inicio o sea si queremos que lo aplique arriba, abajo, a la derecha o a la izquierda de nuestra caja o podemos elegir los grados de inclinación que queremos que tenga nuestro gradiente.

Los puntos de inicio pueden ser *top*, *right*, *left* o *bottom*.

Entonces, supongamos que **queremos aplicarle a nuestra caja un gradiente que va de negro a gris desde la parte de arriba de la misma**. La sintaxis sería la siguiente:

```
div {  
  background: linear-gradient(to top, #000, #ccc)  
}
```

Si lo que queremos es aplicar **un gradiente de 90°**, entonces la sintaxis sería así:

```
div {  
  background: linear-gradient(90deg, #000, #ccc)  
}
```

Algunas cosas a tener en cuenta, la propiedad *linear-gradient* **no genera un color de fondo, sino una imagen sin dimensiones específicas**, esta se va a adaptar automáticamente para cubrir todo el espacio disponible.

Por otro lado, no todos los navegadores soportan esta propiedad.



Transformaciones



Una transformación es la **modificación en la forma en que se muestra un elemento**. Todo elemento transformado por CSS, cambia la forma en que se ve, pero no el lugar que ocupa. Los efectos que podemos lograr son:



- Mover un elemento de lugar (sin position!)
- Escalar el tamaño de un elemento.
- Voltear y girar elementos.
- Cambiar la perspectiva de un elemento.

El *transform* se encarga de modificar los objetos. El valor puede ser solo una propiedad modificada o varias, en cuyo caso cada una se separará por espacios de la siguiente. La transformación se puede aplicar a un objeto o al cambio de un estado. Y como se puede aplicar al cambio de estado, se lo puede transicionar para suavizar el cambio. En browsers viejos requiere prefijo propietario.

```
div {  
  transform: translate(10px, 20px);  
}
```

Rotación de objetos

La rotación permite **girar un objeto sin deformarlo**. Se hace con el *transform: rotate()*. Recibe entre paréntesis un número que representa la cantidad de grados a girar el objeto: Si es positivo, rota hacia la derecha (en sentido reloj). Si es negativo, rota hacia la izquierda (sentido anti horario). Por tratarse de grados, la unidad que acompaña el número será el deg (degrees).

```
div {  
  transform: rotate(45deg);  
}
```

Escalar objetos

transform: scale(), cambia la escala del objeto (como si fuese un zoom). Requiere dos números separados por coma: El primero es el ancho (Escala en eje X). El segundo el es alto (Escala en eje Y). Valores mayores a 1, agrandan. Valores entre 1 y 0, achican. Y valores negativos, escalan dado vuelta. Si solo se quiere cambiar un eje, existe *scaleX()* y *scaleY()*, cada uno solo recibe un número.



```
div {  
  transform:scale(5,1);  
  /* Igual que: scaleX(5) scaleY(1) */  
}
```

Trasladar (mover) objetos

transform:translate(), cambia la ubicación del objeto (como si fuese un **position**). Requiere dos números y su unidad separados por coma: El primero es el desplazamiento horizontal (eje X). El segundo el desplazamiento vertical (eje Y). Valores positivos, mueven a la derecha/abajo. Valores negativos, mueven a la izquierda/arriba. Sí, hay un *translateX()* y *translateY()*, cada uno solo recibe un solo número con su unidad.

```
div {  
  transform: translate(100px); /* Igual a  
  translateX(100px) */  
}
```



Animaciones



A diferencia de la transición, **una animación es un efecto que se loopea tantas veces como quieran**. No depende del cambio de estados (el elemento se animará desde la carga de la web). Una animación es la unión de dos partes. Por un lado una línea de tiempo (llamada keyframe) con la información de los cambios. Por otro lado, aplicar ese keyframe a un elemento que será el que se verá animado.

Animación – línea de tiempo

Es un elemento `@keyframes` con un nombre. Luego del nombre, y entre llaves se definen los puntos donde cambiará el CSS. Cada cambio pasa en un porcentaje de la animación. Por cada punto de inflexión, y entre llaves, van las reglas CSS que se aplicarán en ese momento. El cambio es paulatino de un porcentaje al otro.




```
@keyframes un_efecto{
  0%{ width: 100px; }
  25%{ width: 300px; }
  50%{ width: 200px;
    background-color: red; }
  75%{ width: 300px;}
  100%{ width: 100px;
    background-color:green;}
} /*esta llave cierra el un_efecto */
```

Animación – Aplicar el efecto

Ahora, ese *keyframe* hay que vincularlo a una etiqueta, dándole un *animation* que requiere: Cuantas veces repetir, esto puede ser un número (cantidad exacta) o *infinite*. Duración de toda la animación (este valor se usará en el *keyframe* como 100%). Se mide en seconds. Función de desaceleración. Delay entre que se carga la web y arranca la animación. También en seconds.

```
section{
  width: 100px;
  height: 100px;
  background-color: green;
  animation-name: un_efecto;
  animation-iteration-count: infinite;
  animation-timing-function: ease-in;
  animation-duration: 2s;
  animation-delay: 0s;
}
```



Transiciones

Transiciones CSS le permite cambiar los valores de propiedad sin problemas (de un valor a otro), durante un período determinado.

Ejemplo: al hacer hover sobre el elemento button

```
button {  
  width: 100px;  
  height: 100px;  
  background: red;  
  transition: width 10s;  
}  
  
div:hover {  
  width: 300px;  
}
```

Entonces al hacer Hover sobre el elemento button este crecerá a 300px de ancho. Ese crecimiento será atravesado por un efecto de transición entre una medida y otra.

Generadores

Existen distintos tipos de generadores de código CSS, tanto para animaciones, transformaciones y/o gradientes entre otros.

Alguno de ellos:

<https://www.cssmatic.com/>

<https://daneden.github.io/animate.css/>

<http://www.css3maker.com/css3-transform.html>





Desafío

- Crear sobre el último layout una animación, una transformación y un gradiente para alguno de sus elementos (*Duración : 30 min*).





Desafío

- Agregar animaciones, transformaciones y/o gradientes a nuestro proyecto
(Duracion : 30 min).

