

José Ángel Concepción Sánchez

CSS

Sistemas y Tecnologías Web

Curso 2015/2016

Índice de los principales puntos a tratar

1. ¿Qué es el CSS?
2. CSS2
3. CSS3
4. Introducción a animaciones en CSS3

¿Qué es el CSS?



Historia

- Las hojas de estilo aparecieron poco después del lenguaje de etiquetado SGML, alrededor de 1970.
- El gran boom de las hojas de estilo se produjo poco después de aparecer Internet y debido también al crecimiento exponencial de HTML.
- Entre finales de 1994 y 1995 se crea un lenguaje específico para HTML mezclando lo mejor de las dos propuestas ganadoras. Aparece el CSS.
- Desde entonces han aparecido tres grandes versiones: La primera en 1996, la segunda en 1998 (con una mejora en 2009) y la tercera, de la cual ya se han ido publicando módulos en el último tiempo.

¿Por qué utilizar hojas de estilo (CSS)?

- Nos permite crear y modificar el aspecto de una página web rápidamente y con menos trabajo.
- Mejora la accesibilidad a los contenidos presentes en nuestra página Web.
- Reduce la complejidad de mantenimiento de los sitios web y permite visualizar el mismo documento en infinidad de dispositivos diferentes, ya que es posible marcar de manera mucho más significativa los elementos presentes.

Ideas básicas del CSS

- Una hoja de estilo está formada por una o más reglas (selector/declaración).
 - **Selector:** Identifica al o los elementos a quienes se les aplica la regla.
 - **Declaración:** Especifica los estilos que se le aplican a los elementos. Son conjuntos propiedad/valor.

Ejemplo del aspecto de una “regla”

```
1) h1 { color: green; }
```

```
2) p { font-size: small; font-family: sans-serif; }
```

- Las formas para aplicar las reglas de estilo son:
 - **Hojas de estilo externas:** Todos los estilos CSS se incluyen en un archivo de tipo css que las páginas HTML enlazan mediante las etiquetas `<link>`.
 - **Hojas de estilo internas:** Las hojas de estilo de esta categoría funcionan solamente para el documento en el cual han sido insertadas empleando la etiqueta `<style>` en el head del HTML.
 - **Estilos en línea:** Se aplican propiedades y valores a elementos individuales dentro del HTML.

```
<h1 style="color: red; style=margin-top: 2em">Introducción</h1>
```

Prioridades dentro de CSS

- **Herencia:** Una regla para un elemento es heredada para todos los elementos dependientes de este.
- **Jerarquía:** Una determinada regla de estilo es omitida cuando resulta sobrescrita por una regla de estilo con mayor importancia. Cuanto más cerca esté la regla del elemento, generalmente tiene mayor importancia. (Se pueden cambiar las prioridades con el signo “!” en las reglas).
- **Orden:** Si hay conflicto entre reglas con el mismo nivel de importancia, se aplica la que se indicó en último lugar.

CSS2



Modelo de cajas

- Es el comportamiento de CSS que hace que todos los elementos de las páginas se representen mediante cajas rectangulares.
- Cada vez que se inserta una etiqueta HTML, se crea una nueva caja rectangular que encierra los contenidos de ese elemento.
- Los navegadores crean y colocan las cajas de forma automática, pero CSS permite modificar todas sus características.

width:	auto <i>tamaño</i>	height:	auto <i>tamaño</i>
max-width:	none <i>tamaño</i>	max-height:	none <i>tamaño</i>
min-width:	<i>tamaño</i>	min-height:	<i>tamaño</i>

Selectores en CSS

Ejemplo

El estándar de CSS 2.1, incluye una docena de tipos diferentes de selectores, que permiten seleccionar de forma muy precisa elementos individuales o conjuntos de elementos dentro de una página web.

No obstante, la mayoría de páginas de los sitios web se pueden diseñar usando solamente los cinco selectores básicos:

- Universal.
- De tipo o etiqueta.
- Descendente.
- De clase.
- De ID

Selector para clases en CSS

- La principal característica es que en una misma página HTML varios elementos diferentes pueden tener el mismo valor en el atributo *class*.
- Es uno de los más usados junto con el de ID que veremos posteriormente.

```
<body>
  <p class="destacado">Lorem ipsum dolor sit amet...</p>
  <p>Nunc sed lacus et <a href="#" class="destacado">est adipiscing</a> accumsan...</p>
  <p>Class aptent taciti <em class="destacado">sociosqu ad</em> litora...</p>
</body>
```

Ejemplo 1:

```
.destacado { color: red; }
```

Ejemplo 2:

```
p.destacado { color: red; }
```

- También podemos aplicar varias clases a un mismo elemento:

```
<p class="especial destacado error">Párrafo de texto...</p>
```

Selector para IDs en CSS

- Permite seleccionar un elemento de la página a través del valor de su atributo id.
- El valor del atributo id no se puede repetir en dos elementos diferentes de una misma página.

```
<p>Primer párrafo</p>  
<p id="destacado">Segundo párrafo</p>  
<p>Tercer párrafo</p>
```

Ejemplo 1:

```
#destacado { color: red; }
```

Ejemplo 2:

```
/* Todos los elementos de tipo "p" con atributo id="aviso" */  
p#aviso { ... }  
  
/* Todos los elementos con atributo id="aviso" que estén dentro  
de cualquier elemento de tipo "p" */  
p #aviso { ... }  
  
/* Todos los elementos "p" de la página y todos los elementos con  
atributo id="aviso" de la página */  
p, #aviso { ... }
```

Margen, relleno y bordes (I)

Ejemplo

- **Margen** (margin): Establece la separación entre el borde y el resto de las cajas adyacentes.

Ejemplo 1:

```
div img {  
  margin-top: .5em;  
  margin-bottom: .5em;  
  margin-left: 1em;  
  margin-right: .5em;  
}
```

Ejemplo 2:

```
div img {  
  margin: .5em .5em .5em 1em;  
}
```

- **Relleno** (padding): Establece la separación entre el contenido y los bordes laterales de la caja del elemento.

Ejemplo:

```
body {padding: 2em} /* Todos los rellenos valen 2em */  
body {padding: 1em 2em} /* Superior e inferior = 1em, Izquierdo y derecho = 2em */  
body {padding: 1em 2em 3em} /* Superior = 1em, derecho = 2em, inferior = 3em, izquierdo = 2em */  
body {padding: 1em 2em 3em 4em} /* Superior = 1em, derecho = 2em, inferior = 3em, izquierdo = 4em */
```

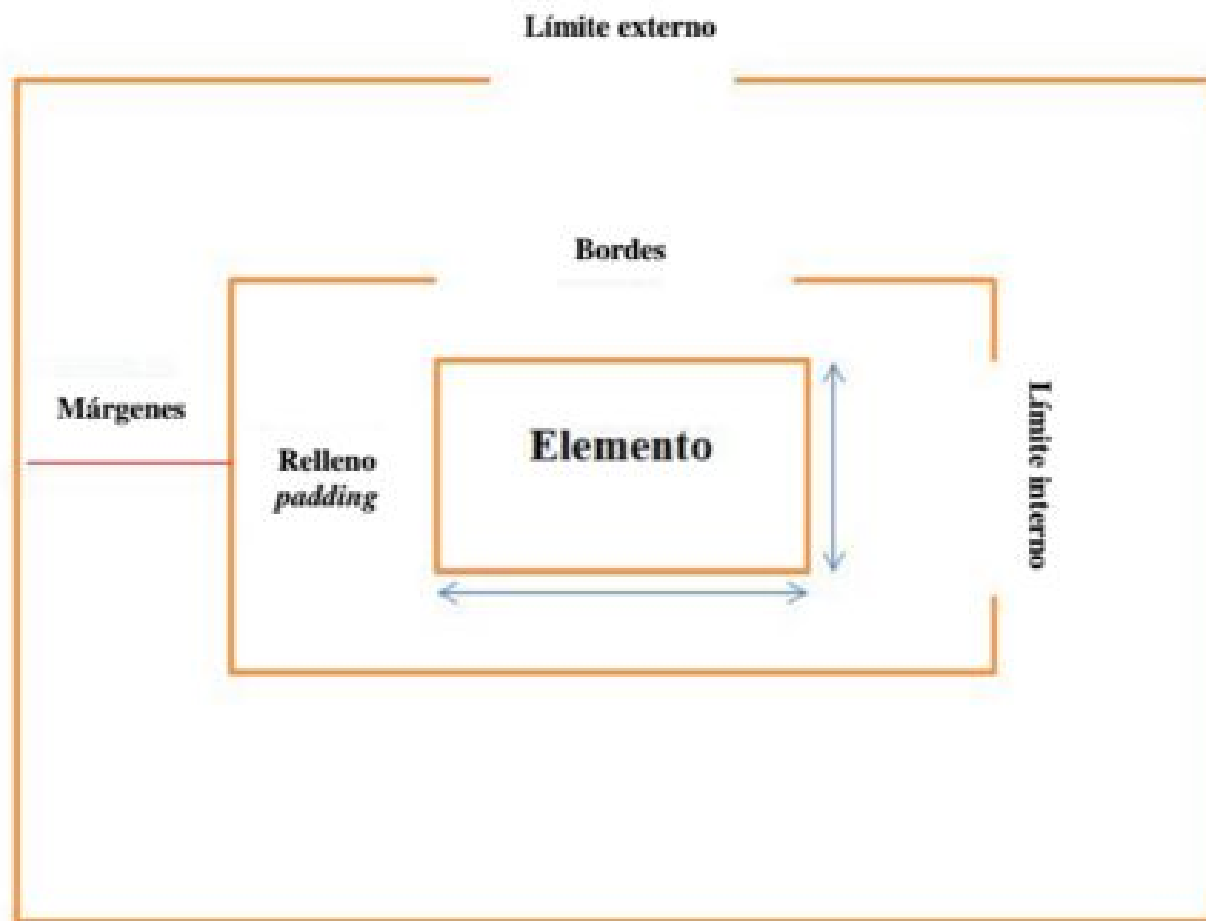
Margen, relleno y bordes (II)

- **Bordes (border):** Establece el borde que se encuentra entre los márgenes y el relleno.

```
div {  
  border-top: 1px solid red;  
  border-right: 1px solid red;  
  border-bottom: 1px solid red;  
  border-left: 1px solid red;  
}
```

es equivalente a: `div { border: 1px solid red; }`

- También se puede asignar un color a los bordes sin dar más parámetros (*border-color*) o la anchura (*border-width*), por ejemplo.



CSS3



Diferencias respecto a CSS2

- CSS3 funciona con módulos, lo que permite que pueda ser integrado en CSS y por tanto sean compatibles.
- Ahorra tiempo porque no es necesario el uso de otros programas para ciertos efectos.
- Carga mucho más rápido en la web.
- Posee nuevas propiedades.
- Define mejor el comportamiento de los medios.

Ejemplo 1 de animación CSS3

Ejemplo 2 de animación CSS3



Propiedades básicas de CSS3 (I)

- Esquinas redondeadas con “border-radius”

La regla número uno de la Web 2.0 es: si tiene esquinas redondeadas, es moderno. Gracias a la propiedad *border-radius* de CSS3, ya no hace falta que utilices imágenes ni Javascript para conseguir esquinas redondeadas. Un ejemplo:

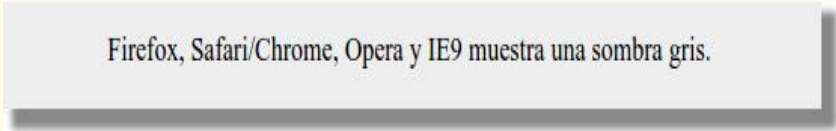


Ejemplo de esquinas redondeadas con CSS3

URL: [Bordes redondos](#)

- Crear sombras con “box-shadow”

Permite implementar fácilmente sombras tanto dentro como fuera de elementos de caja, especificando valores como el color, tamaño, borrosidad y offset. Un ejemplo:



Firefox, Safari/Chrome, Opera y IE9 muestra una sombra gris.

URL: [Sombras](#)

Propiedades básicas de CSS3 (II)

- Múltiples backgrounds

CSS3 nos permite especificar múltiples imágenes como background de una manera sencilla, ya que sólo hay que separar las URLs mediante comas.

Un ejemplo es:



URL: [Backgrounds](#)

- Opacidad en CSS3

Con *opacity*, la opacidad puede tener una valor de 0 a 1. Con el valor al 1 el color será al 100%, si es 0.2 será al 20% y así. Un ejemplo es:



URL: [Opacidad](#)

Introducción a animaciones en CSS3



Animaciones en CSS3

- Permiten animar la transición entre un estilo CSS y otro.
- Constan de dos componentes:
 - Un estilo que describe la animación.
 - Un conjunto de fotogramas que indican su estado inicial y final, así como posibles puntos intermedios.
- Constan de:
 - @keyframe: Permite crear una animación compuesta por varios fotogramas.
 - Elemento al que queremos aplicarle la animación.
- Hay que añadir prefijos si el navegador no está actualizado.

[URL del ejemplo](#)

```
@keyframes animacion1 {  
  from { left:0px }  
  to { left:500px }  
}
```

```
.anim {  
  position: relative;  
  background: #000;  
  color: #FFF;  
  animation-name: animacion1;  
  animation-duration: 2s;  
  animation-timing-function: ease;  
}
```

Propiedades de las animaciones

Propiedad	Valor	Uso
animation-name	none nombre	Nombre de la animación.
animation-duration	0 tiempo	Tiempo de la animación.
animation-timing-function	ease linear ease-in ease-out ...	Indica el ritmo de la animación.
animation-delay	0 tiempo	Tiempo que tarda en empezar tras cargarse.
animation-iteration-count	1 infinite número	Veces que se repite.
animation-fill-mode	none forwards backwards both	Valores finales tras animación.
animation-play-state	running paused	Pausa y reanuda la animación.
animation-direction	normal reverse alternate alternate-reverse	Indica si se empieza de nuevo por el principio o va hacia atrás.

Transformaciones 2D en CSS3

En CSS3 es posible utilizar transformaciones (*transform*) para alterar los elementos en un espacio 2D (eje X + eje Y) o 3D (eje X + eje Y + eje Z). Algunas cosas que se pueden hacer son:

- **Traslaciones** (*translate*): Realiza un movimiento relativo del elemento (píxeles).
- **Escalado** (*scale*): Realiza una ampliación o reducción de un elemento.
- **Rotación** (*rotate*): El elemento girará tanto grados como se le especifiquen.
- **Deformación** (*skwe*): Torcerá el elemento tantos grados como se le especifiquen.

Ejemplo de uso de transformaciones

Gracias por su atención.

Bibliografía

- Libros Web
- Ejemplos de CSS3
- Animaciones en CSS3
- Animaciones developers Mozilla
- Animaciones en 3D
- Curso online “Diseño Web” de Academia Mañana