

Tema 2. Selectores

Los selectores indican *"a quién hay que aplicar el estilo"*. Por lo tanto, los selectores son imprescindibles para aplicar de forma correcta los estilos CSS en una página.

Una misma regla puede aplicarse sobre varios selectores y un mismo selector se puede utilizar en varias reglas.

El estándar de CSS 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 utilizando solamente los cinco selectores básicos.

2.1. Selectores básicos

2.1.1. Selector universal

Se utiliza para seleccionar todos los elementos de la página. El siguiente ejemplo elimina el margen y el relleno de todos los elementos HTML (por ahora no es importante fijarse en la parte de la declaración de la regla CSS):

```
* {  
  margin: 0;  
  padding: 0;  
}
```

El selector universal se indica mediante un asterisco (*). No se utiliza habitualmente, ya que es difícil que un mismo estilo se pueda aplicar a todos los elementos de una página.

2.1.2. Selector de tipo

Selecciona todos los elementos de la página cuya etiqueta HTML coincide con el valor del selector. El siguiente ejemplo selecciona todos los párrafos de la página:

```
p {  
  ...  
}
```

Para utilizar este selector, solamente es necesario indicar el nombre de una etiqueta HTML (sin los caracteres < y >) correspondiente a los elementos que se quieren seleccionar.

El siguiente ejemplo aplica diferentes estilos a los titulares y a los párrafos de una página HTML:

```
h1 {  
  color: red;  
}
```

```
h2 {
  color: blue;
}

p {
  color: black;
}
```

En las hojas de estilo, es habitual agrupar las propiedades comunes de varios elementos en una única regla CSS y posteriormente definir las propiedades específicas de esos mismos elementos. El siguiente ejemplo establece en primer lugar las propiedades comunes de los títulos de sección (color y tipo de letra) y a continuación, establece el tamaño de letra de cada uno de ellos:

```
h1, h2, h3 {
  color: #8A8E27;
  font-weight: normal;
  font-family: Arial, Helvetica, sans-serif;
}

h1 { font-size: 2em; }
h2 { font-size: 1.5em; }
h3 { font-size: 1.2em; }
```

2.1.3. Selector descendiente

Selecciona los elementos que se encuentran dentro de otros elementos. Un elemento es descendiente de otro cuando se encuentra entre las etiquetas de apertura y de cierre del otro elemento.

El selector del siguiente ejemplo selecciona todos los elementos `` de la página que se encuentren dentro de un elemento `<p>`:

```
p span { color: red; }
```

Si el código HTML de la página es el siguiente:

```
<p>
  ...
  <span>texto1</span>
  ...
  <a href="">...<span>texto2</span></a>
  ...
</p>
```

El selector `p span` selecciona tanto `texto1` como `texto2`. El motivo es que en el selector descendiente, un elemento no tiene que ser "*hijo directo*" de otro. La única condición es que un elemento debe estar dentro de otro elemento, sin importar lo profundo que se encuentre.

Al resto de elementos `` de la página que no están dentro de un elemento `<p>`, no se

les aplica la regla CSS anterior.

La sintaxis formal del selector descendente se muestra a continuación:

```
| elemento1 elemento2 elemento3 ... elementoN
```

Los selectores descendentes siempre están formados por dos o más partes separadas entre sí por espacios en blanco. La última parte indica el elemento sobre el que se aplican los estilos y todas las partes anteriores indican el lugar en el que se tiene que encontrar ese elemento para que los estilos se apliquen.

En el siguiente ejemplo, el selector descendente se compone de cuatro partes:

```
| p a span em { text-decoration: underline; }
```

No debe confundirse el selector descendente con la combinación de selectores:

```
| /* El estilo se aplica a todos los elementos "p", "a", "span" y "em" */  
| p, a, span, em { text-decoration: underline; }
```

2.1.4. Selector de clase

Si se considera el siguiente código HTML de ejemplo:

```
| <body>  
|   <p>texto 1</p>  
|   <p> texto 2</p>  
|   <p> texto 3</p>  
| </body>
```

¿Cómo se pueden aplicar estilos CSS sólo al primer párrafo? El selector universal (*) no se puede utilizar porque selecciona todos los elementos de la página. El selector de tipo o etiqueta (p) tampoco se puede utilizar porque seleccionaría todos los párrafos. Por último, el selector descendente (body p) tampoco se puede utilizar porque todos los párrafos se encuentran en el mismo sitio.

Una de las soluciones más sencillas para aplicar estilos a un solo elemento de la página consiste en utilizar el atributo class de HTML sobre ese elemento para indicar directamente la regla CSS que se le debe aplicar:

```
| <body>  
|   <p class="grande">texto 1</p>  
|   <p class="medio"> texto 2</p>  
|   <p class="pequeño"> texto 3</p>  
| </body>
```

Para que el navegador no confunda este selector con los otros tipos de selectores, se prefija el valor del atributo class con un punto (.) tal y como muestra el siguiente ejemplo:

```
| p.grande {font-size: 50px;}  
| p.medio {font-size: 20px;}
```

```
p.pequeño {font-size: 10px;}
```

Supongamos el siguiente ejemplo:

```
<body>
  <h3 class="grande">texto 1</p>
  <p class="grande"> texto 2</p>
  <p class="pequeño"> texto 3</p>
</body>
```

En CSS podemos utilizar lo siguiente:

```
.grande {font-size: 50px;}
```

El selector `.grande` se interpreta como *"cualquier elemento de la página cuyo atributo `class` sea igual a `grande`"*, por lo que la regla CSS se le aplicara a la etiqueta `h3` y al primer párrafo.

Por último, es posible aplicar los estilos de varias clases CSS sobre un mismo elemento. La sintaxis es similar, pero los diferentes valores del atributo `class` se separan con espacios en blanco. En el siguiente ejemplo:

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

Al párrafo anterior se le aplican los estilos definidos en las reglas `.especial`, `.destacado` y `.error`, por lo que en el siguiente ejemplo, el texto del párrafo se vería de color rojo, en negrita y con un tamaño de letra de 15 píxel:

```
.error { color: red; }
.destacado { font-size: 15px; }
.especial { font-weight: bold; }
```

2.1.5. Selectores de ID

En ocasiones, es necesario aplicar estilos CSS a un único elemento de la página. Aunque puede utilizarse un selector de clase para aplicar estilos a un único elemento, existe otro selector más eficiente en este caso.

El selector de ID permite seleccionar un elemento de la página a través del valor de su atributo `id`. Este tipo de selectores sólo seleccionan un elemento de la página porque el valor del atributo `id` no se puede repetir en dos elementos diferentes de una misma página.

La sintaxis de los selectores de ID es muy parecida a la de los selectores de clase, salvo que se utiliza el símbolo de la almohadilla (`#`) en vez del punto (`.`) como prefijo del nombre de la regla CSS:

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

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

En el ejemplo anterior, el selector `#destacado` solamente selecciona el segundo párrafo.

La principal diferencia entre este tipo de selector y el selector de clase tiene que ver con HTML y no con CSS. Como se sabe, en una misma página, el valor del atributo `id` debe ser único, de forma que dos elementos diferentes no pueden tener el mismo valor de `id`. Sin embargo, el atributo `class` no es obligatorio que sea único, de forma que muchos elementos HTML diferentes pueden compartir el mismo valor para su atributo `class`.

De esta forma, la recomendación general es la de utilizar el selector de ID cuando se quiere aplicar un estilo a un solo elemento específico de la página y utilizar el selector de clase cuando se quiere aplicar un estilo a varios elementos diferentes de la página HTML.

Realizar ejercicio 1 subido a aules.

2.2. Selectores avanzados

Utilizando solamente los selectores básicos de la sección anterior, es posible diseñar prácticamente cualquier página web. No obstante, CSS define otros selectores más avanzados.

2.2.1. Selector de hijos

Se utiliza para seleccionar un elemento que es *hijo directo* de otro elemento y se indica mediante el "*signo de mayor que*" (`>`):

```
p > span { color: blue; }

<p><span>Texto1</span></p>
<p><a href="#"><span>Texto2</span></a></p>
```

En el ejemplo anterior, el selector `p > span` se interpreta como "*cualquier elemento `` que sea hijo directo de un elemento `<p>`*", por lo que el primer elemento `` cumple la condición del selector. Sin embargo, el segundo elemento `` no la cumple porque es descendiente pero no es hijo directo de un elemento `<p>`.

2.2.2. Selector adyacente

El selector adyacente utiliza el signo `+` y su sintaxis es:

```
| elemento1 + elemento2 { ... }
```

Selecciona todos los elementos de tipo elemento2 que cumplan las dos siguientes condiciones:

- elemento1 y elemento2 deben ser hermanos, por lo que su elemento padre debe ser el mismo.
- elemento2 debe aparecer inmediatamente después de elemento1 en el código HTML de la página.

En el siguiente ejemplo:

```
h1 + h2 { color: red }  
  
<body>  
<h1>Titulo1</h1>  
  
<h2>Subtítulo</h2>  
...  
  
<h2>Otro subtítulo</h2>  
...  
</body>
```

Los estilos del selector `h1 + h2` se aplican al primer elemento `<h2>` de la página, pero no al segundo `<h2>`, ya que:

- El elemento padre de `<h1>` es `<body>`, el mismo padre que el de los dos elementos `<h2>`. Así, los dos elementos `<h2>` cumplen la primera condición del selector adyacente.
- El primer elemento `<h2>` aparece en el código HTML justo después del elemento `<h1>`, por lo que este elemento `<h2>` también cumple la segunda condición del selector adyacente.
- Por el contrario, el segundo elemento `<h2>` no aparece justo después del elemento `<h1>`, por lo que no cumple la segunda condición del selector adyacente y por tanto no se le aplican los estilos de `h1 + h2`.

2.2.3. Selector de atributos

Permiten seleccionar elementos HTML en función de sus atributos y/o valores de esos atributos.

Los cuatro tipos de selectores de atributos son:

- `[nombre_atributo]`, selecciona los elementos que tienen establecido el atributo llamado `nombre_atributo`, independientemente de su valor. (puede tener otros atributos)
- `[nombre_atributo=valor]`, selecciona los elementos que tienen establecido un atributo llamado `nombre_atributo` con un valor igual a `valor`. (puede tener otros atributos)

- `[nombre_atributo~=valor]`, selecciona los elementos que tienen establecido un atributo llamado `nombre_atributo` y al menos uno de los valores del atributo es `valor`. (puede tener otros atributos)

A continuación se muestran algunos ejemplos de estos tipos de selectores:

```
/* Se muestran de color azul todos los enlaces que tengan  
un atributo "class", independientemente de su valor */  
a[class] { color: blue; }
```

```
/* Se muestran de color azul todos los enlaces que tengan  
un atributo "class" con el valor "externo" */  
a[class="externo"] { color: blue; }
```

```
/* Se muestran de color azul todos los enlaces que tengan  
un atributo "class" en el que al menos uno de sus valores  
sea "externo" */  
a[class~="externo"] { color: blue; }
```

2.2. Pseudo-clases

Selecciona elementos en algún estado especial y se pone la etiqueta a la que afecta, dos puntos y la pseudo-clase, todo seguido sin espacios. Hay muchas pero las principales son estas:

- `etiqueta:hover` → se aplica cuando se pasa el ratón por encima de la etiqueta indicada. Ejemplo: `p:hover`, `a:hover`
- `etiqueta:first-of-type` → se aplica a la primera etiqueta indicada. Ejemplo: `p:first-of-type`, `a:first-of-type`
- `etiqueta:first-child` → se aplica al primer hijo de la etiqueta indicada. Ejemplo: `p:first-child`, `strong:first-child`
- `etiqueta:last-child` → se aplica al último hijo de la etiqueta indicada. Ejemplo: `p:last-child`, `strong:last-child`
- `a:link` → se aplica a los enlaces
- `a:visited` → se aplica a los enlaces visitados
- `a:active` → se aplica a los enlaces que están siendo pulsados