

Capítulo 2

Selectores

Selectores

- La declaración indica *"qué hay que hacer"* y el selector indica *"a quién hay que hacérselo"*.
 - Selector { atributo:valor }
- Ejemplo:
 - h1 {color:blue}



Selectores básicos

Selector universal

- Se utiliza para seleccionar todos los elementos de la página.

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

Selector de tipo o etiqueta

- Selecciona todos los elementos de la página cuya etiqueta **HTML** coincide con el valor del selector.
- Ejemplo:

p {

...

}

Selector de tipo o etiqueta

- 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;
```


Selector de tipo o etiqueta

- Se pueden incluir varios selectores *separados por una coma (,)*:

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

Selector de tipo o etiqueta

- Este ejemplo establece en primer lugar las propiedades comunes de los títulos de sección 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; }
```


Ejercicio

- Crea un fichero HTML con dos textos, uno entre etiquetas h1 y otro entre etiquetas h2. Luego incluye en el head un estilo que afecte al color del texto de las etiquetas h1 y al color de fondo de los h2.

Selector descendente

- 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.
- Ejemplo:

p span { color: red; }

Selector descendente

- Ejemplo:

`<p>`

`...`

`texto1`

`...`

`...texto2`

`...`

`</p>`

- El selector `p span` selecciona tanto `texto1` como `texto2`.
- El motivo es que un elemento no tiene que ser "*hijo directo*" de otro.

Selector descendente

- La *sintaxis formal* del selector descendente es:

elemento1 elemento2 elemento3 ... elementoN

- Los *selectores descendentes* siempre están formados por dos o más partes separadas entre sí por espacios en blanco.

Selector descendente

- 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; }

/ El estilo se aplica solo a los elementos "em" que se encuentran dentro de "p a span" */*

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

Selector descendente

- Ejemplo:

```
p * a { color: red; }
```

```
<p><a href="#">Enlace</a></p>
```

```
<p><span><a href="#">Enlace</a></span></p>
```


Selector de clase

- Si se considera el siguiente código **HTML** de ejemplo:

```
<body>
```

```
<p>Lorem ipsum dolor sit amet...</p>
```

```
<p>Nunc sed lacus et est adipiscing accumsan...</p>
```

```
<p>Class aptent taciti sociosqu ad litora...</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.

Selector de clase

- Solución: utilizar el atributo class de HTML:

```
<body>
```

```
<p class="destacado">Lorem ipsum dolor sit amet...</p>
```

```
<p>Nunc sed lacus et est adipiscing accumsan...</p>
```

```
<p>Class aptent taciti sociosqu ad litora...</p>
```

```
</body>
```

Selector de clase

- Ejemplo de selectores de clase:

```
.aviso {  
    padding: 0.5em;  
    border: 1px solid #98be10;  
    background: #f6feda;  
}
```

```
.error {  
    color: #930;  
    font-weight: bold;  
}
```

```
<span class="error">...</span>
```


Selector de clase

- Si se considera de nuevo el ejemplo anterior:

```
<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 <span class="destacado">sociosqu ad</span>  
litora...</p>
```

```
</body>
```

Selector de clase

- ¿Cómo es posible aplicar estilos solamente al párrafo cuyo atributo **class** sea igual a destacado?

p.destacado { color: red }

Selector de clase

- No debe confundirse el selector de clase con los selectores anteriores:

/ Todos los elementos de tipo "p" con atributo class="aviso" */*

`p.aviso { ... }`

/ Todos los elementos con atributo class="aviso" que estén dentro*

*de cualquier elemento de tipo "p" */*

`p .aviso { ... }`

/ Todos los elementos "p" de la página y todos los*

Selector de clase

- Podemos aplicar los estilos de varias clases CSS sobre un mismo elemento.
- Los diferentes valores del atributo **class** se separan con espacios en blanco.
- Ejemplo:

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

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

```
.destacado { font-size: 15px; }
```

```
.especial { font-weight: bold; }
```

Selector de clase

- Es posible utilizar un selector más avanzado:

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

```
.error.destacado { color: blue; }
```

```
.destacado { font-size: 15px; }
```

```
.especial { font-weight: bold; }
```

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

- En el ejemplo anterior, el color de la letra del texto es *azul* y no rojo.
- Se ha utilizado un selector de clase múltiple **.error.destacado**, que se interpreta como *"aquellos elementos de la página que dispongan de un atributo class con al menos los valores*

Selectores de ID

- Este tipo de selectores sólo seleccionan un elemento de la página.
- Con estos selectores 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>
```


Selectores de ID

- Este ejemplo aplica la regla **CSS** solamente al elemento de tipo `<p>` que tenga un atributo `id` igual a “**aviso**”:

p#aviso { color: blue; }

- Un selector de tipo `p#aviso` sólo tiene sentido cuando el archivo **CSS** se aplica sobre muchas páginas **HTML** diferentes.

Selectores de ID

- No debe confundirse el selector de **ID** con los selectores anteriores:

/ 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 { ... }`

Combinación de selectores básicos

- La combinación de selectores puede llegar a ser todo lo compleja que sea necesario:

```
ul#menuPrincipal li.destacado a#inicio { ... }
```


Selector de hijos

- Se trata de un selector similar al selector descendente, pero muy diferente en su funcionamiento.
- 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>
```

Selector de hijos

- El siguiente ejemplo muestra las diferencias entre el selector descendente y el selector de hijos:

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

```
p > a { color: red; }
```

```
<p><a href="#">Enlace1</a></p>
```

```
<p><span><a href="#">Enlace2</a></span></p>
```

Selector adyacente

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

elemento1 + elemento2 { ... }

- El comportamiento de este selector 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.

Selector adyacente

- En el siguiente ejemplo:

```
h1 + h2 { color: red }
```

```
<body>
```

```
<h1>Titulo1</h1>
```

```
<h2>Subtítulo</h2>
```

```
...
```

```
<h2>Otro subtítulo</h2>
```

```
...
```

```
</body>
```

Selector adyacente

- El siguiente ejemplo es muy útil para los textos que se muestran como libros:

p + p { text-indent: 1.5em; }

- Suele ser habitual que la primera línea de todos los párrafos esté indentada, salvo la primera línea del primer párrafo.
- El selector *p + p* selecciona todos los párrafos de un elemento salvo el primer párrafo.

Selector de atributos

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

Selector de 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.
 - **[nombre_atributo=valor]**, selecciona los elementos que tienen establecido un atributo llamado nombre_atributo con un valor igual a valor.
 - **[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.

Selector de 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 apunten al sitio "http://www.ejemplo.com" */*

Selector de atributos

/ 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; }
```

/ Selecciona todos los elementos de la página cuyo atributo "lang" sea igual a "en", es decir, todos los elementos en inglés */*

```
*[lang=en] { ... }
```


Agrupación de reglas

- Cuando el selector de dos o más reglas **CSS** es idéntico, se deben agrupar las declaraciones de las reglas para hacer las hojas de estilos más eficientes:

```
h1 {  
    color: red;  
    font-size: 2em;  
    font-family: Verdana;  
}
```

Herencia

- Uno de los conceptos más característicos de CSS es la herencia de los estilos definidos para los elementos.
- Cuando se establece el valor de alguna propiedad en un elemento, todos sus descendientes heredan inicialmente ese mismo valor.

Herencia

- Si se indica por ejemplo un tipo de letra al elemento `<body>` de una página, todos los elementos de la página mostrarán ese tipo de letra, salvo que se indique lo contrario:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<title>Ejemplo de herencia de estilos</title>
<style type="text/css">
    body { font-family: Arial; color: black; }
    h1 { font-family: Verdana; }
    p { color: red; }
</style>
```


Colisiones de estilos

- En las hojas de estilos complejas, es habitual que varias reglas **CSS** se apliquen a un mismo elemento **HTML**.
- El problema de estas reglas múltiples es que se pueden dar colisiones como la del siguiente ejemplo:

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

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

```
<p>...</p>
```

Colisiones de estilos

- ¿De qué color se muestra el párrafo anterior?
- CSS tiene un *mecanismo de resolución de colisiones* y tiene en cuenta:
 - el tipo de hoja de estilo que se trate,
 - la importancia de cada regla y
 - lo específico que sea el selector.

Colisiones de estilos

- Se describe a continuación el método genérico seguido por CSS para resolver las colisiones:
 1. Determinar todas las declaraciones que se aplican al elemento para el medio CSS seleccionado.
 2. Ordenar las declaraciones según su origen (CSS de navegador, de usuario o de diseñador) y su importancia (palabra clave !important).
 3. Ordenar las declaraciones según lo específico que sea el selector. Cuanto más genérico es un selector, menos importancia tienen sus declaraciones.
 4. Si después de aplicar las normas anteriores existen dos o más reglas con la misma prioridad, se aplica la que se indicó en último lugar.

Colisiones de estilos

- La norma que se puede seguir es la de la "*especificidad*" del selector:
 1. Cuanto más específico sea un selector, más importancia tiene su regla asociada.
 2. A igual *especificidad*, se considera la última regla indicada.

Colisiones de estilos

- Ejemplo:

p { color: red; }

p#especial { color: green; }

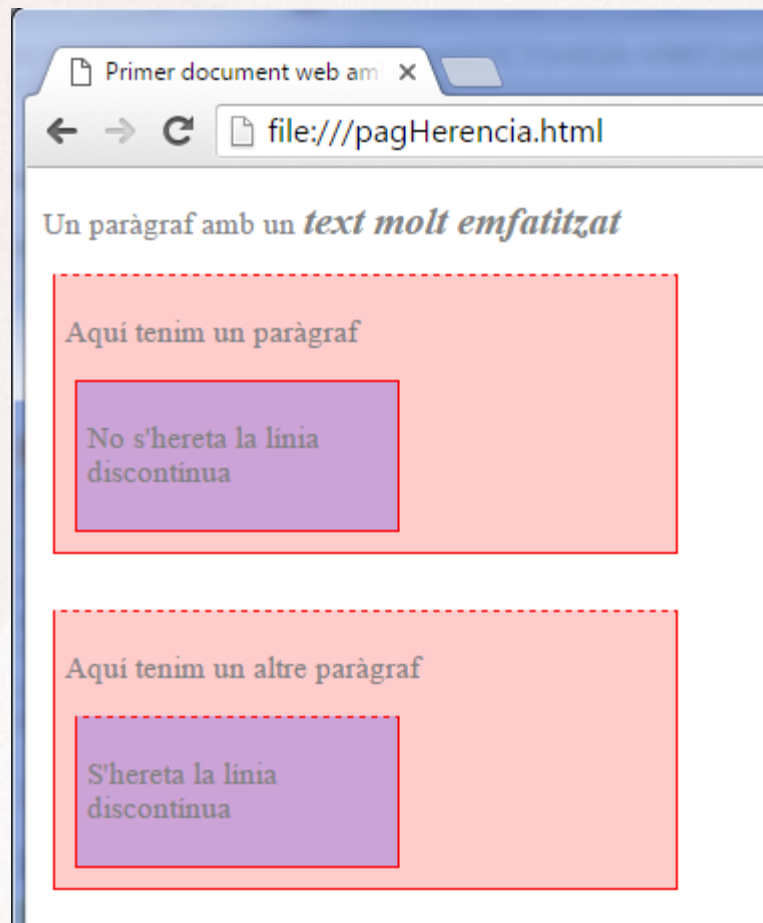
** { color: blue; }*

<p id="especial">...</p>

Colisiones de estilos - excepciones

- Todo elemento hereda las propiedades de sus antecesores. Sin embargo, se deben tener en cuenta las siguientes consideraciones:
 - No todas las propiedades se heredan. Esta característica se puede consultar en: www.w3.org/TR/css-2010/#properties.
- Si se quiere forzar la herencia de una propiedad de un elemento que por defecto no hereda se puede utilizar el valor inherit.
- Si se especifica un valor a una propiedad, este valor prevalece sobre el valor heredado.

Ejemplo:



La regla !important

- La palabra clave !important se utiliza para dar mayor prioridad a una instrucción, las instrucciones siguientes no puedan sobreescrirla.

- Ejemplo:

```
p {  
    color: #000066 !important;  
}
```


Resumen sobre selector de atributos

- [Atributo] Selecciona el elemento con este atributo.
- [Atributo = "valor"] El valor del atributo es valor.
- [Atributo ^ = "texto"] El valor del atributo comienza por texto.
- [Atributo \$ = "texto"] El valor del atributo acaba con texto.
- [Atributo * = "texto"] El valor del atributo contiene el texto.
- [Atributo ~ = "valor"] El valor del atributo es una lista de palabras separadas por espacios, una de las cuales es exactamente valor.
- [Atributo | = "valor"] El valor del atributo es una lista de palabras separadas por guiones, empezando por valor.

Resumen del selector de etiquetas

- $A + B$ Selectores adyacentes. Permite aplicar un estilo al elemento que se encuentre después (B) de un elemento específico (A).
- $A \sim B$ Selectores general de hermanos. Permite aplicar un estilo al elemento (B) que sea hermano del elemento (A).
- $A > B$ Selectores de hijos. Permite aplicar un estilo al primer elemento hijo (B) de un elemento específico (A).
- $A B$ Selectores descendientes. Permite aplicar un estilo a un elemento determinado que se encuentre dentro de un elemento específico.
- Ejemplo: <https://codepen.io/ioc-daw-m09/pen/BKdeKW>

Ejercicio 2

- Realizar una página web en HTML, con las reglas **CSS** necesarias, para que la página resultante tenga el mismo aspecto que el de la imagen en la última diapositiva.

Ejercicio 2

- Aunque la propiedad que modifica el color del texto se explica detalladamente en los próximos capítulos, en este ejercicio solamente es preciso conocer que la propiedad se llama color y que como valor se puede indicar directamente el nombre del color.
- Los nombres de los colores también están estandarizados y se corresponden con el nombre en inglés de cada color.
- En este ejercicio, se deben utilizar los colores: teal, red, blue, orange, purple, olive, fuchsia y green.

Lorem ipsum dolor sit amet

Nulla pretium. Sed tempus nunc vitae neque. **Suspendisse grvida**, metus a scelerisque sollicitudin, lacus velit ultricies nisl, nonummy tempus neque diam quis felis. **Etiam sagittis tortor** sed arcu sagittis tristique.

Aliquam tincidunt, sem eget volutpat porta

Vivamus velit dui, placerat vel, feugiat in, ornare et, urna. Aenean turpis metus, aliquam non, tristique in, pretium varius, sapien. Proin vitae nisi. Suspendisse **porttitor purus ac elit**. Suspendisse eleifend odio at dui. In in elit sed metus pretium elementum.

Título de la tabla

	Título columna 1	Título columna 2
Título fila 1	Donec purus ipsum	Curabitur <i>blandit</i>
Título fila 2	Donec purus ipsum	Curabitur blandit
	Título columna 1	Título columna 2

Donec purus ipsum, posuere id, venenatis at, placerat ac, lorem. Curabitur blandit, eros sed grvida aliquet, risus justo porta lorem, ut mollis lectus tortor in orci. Pellentesque nec augue.

Fusce nec felis eu diam pretium adipiscing. **Nunc elit elit, vehicula vulputate**, venenatis in, posuere id, lorem. Etiam sagittis, tellus in ultrices accumsan, diam nisi feugiat ante, eu congue magna mi non nisl.

Vivamus ultrices aliquet augue. Donec arcu pede, pretium vitae, rutrum aliquet, tincidunt blandit, pede. Aliquam in nisi. Suspendisse volutpat. Nulla facilisi. Ut ullamcorper nisi quis mi.

Ejercicio 2

- Algunas cuestiones a tener claras:
 - ¿Cómo ponemos un texto en negrita?
¿Y en cursiva? Diferencias entre **b**, *i*, `em` y `strong`.
 - Subrayado: ¿cómo hacerlo?

Ejercicio 2

- Ejemplos:
 - `El dinero` es importante pero `la salud` lo es más.
 - Con `<u>u</u>` subrayamos. Pero está desaprobado, lo que hay que usar es la propiedad CSS `text-decoration: underline`.

Ejercicio 2

- Otras posibilidades:

```
text-decoration: underline red;      /*  
Subrayado rojo */
```

```
text-decoration: underline wavy red; /*  
Subrayado rojo ondulado */
```

- También podemos usar:

```
text-decoration-color: currentcolor  
text-decoration-style: solid  
text-decoration-line: none
```