

HTML

Primeros pasos

April 29, 2022

¿Qué es HTML?

- HTML, siglas de *HyperText Markup Language*, es un lenguaje de marcado para la elaboración de páginas web.
- Un lenguaje de marcado es una forma de codificar un documento que, junto con el texto, incorpora etiquetas o marcas que contienen información adicional acerca de la estructura del texto o su presentación. Los lenguajes de marcado no son lenguajes de programación.
- Latex es otro lenguaje de marcado orientado a la creación de textos científicos.

¿Qué es HTML?

- SGML (*Standard Generalized Markup Language*). Consiste en un sistema para la organización y etiquetado de documentos. El lenguaje SGML sirve para especificar las reglas de etiquetado de documentos y no impone en sí ningún conjunto de etiquetas en especial.
- HTML es un estándar a cargo de la W3C (*World Wide Web Consortium*). W3C es un consorcio internacional que produce recomendaciones para la World Wide Web.
- W3C también ha editado XML (*eXtensible Markup Language*) para gestionar de forma estructurada los datos de tipo texto. El aspecto "extensible" permite al usuario definir sus propias marcas.

Aspectos fundamentales de HTML5

- HTML5 es una evolución de HTML. Es compatible con lo el desarrollo previo de HTML (los que no son tan compatibles son los navegadores). Es menos formal que XHTML(*eXtensible HyperText Markup Language*) pero respeta las normas básicas de etiquetado.
- El principio de separación entre contenido y representación se mantiene y se refuerza: muchas etiquetas y atributos han sido abandonados. HTML5 es inconcebible sin CSS3.
- Los *plug-ins* son menos necesarios: mejoras de audio y vídeo.

- Las *etiquetas* delimitan las instrucciones de HTML5. Son comandos interpretados por el navegador delimitados por los signos `<` y `>`. Ejemplo: `<etiqueta>`.
- Generalmente, a cada etiqueta de apertura le corresponde una etiqueta de cierre que marca el final de la instrucción o comando indicado. Ejemplo: `<etiqueta>...</etiqueta>`.
- Existen algunas instrucciones sin etiqueta de cierre con el siguiente formato `<etiqueta.../>`.

- En ocasiones es necesario indicar información adicional a la instrucción. Para tal caso las etiquetas poseen *atributos*.
- Sintaxis de una etiqueta con atributos:

```
<etiqueta atributo1='valor1' atributo2='valor2'>...</etiqueta>
```

- Es necesario que no haya espacios en blanco antes y después del igual de los atributos.
- Las etiquetas deben cerrarse en orden inverso de su apertura.

El documento HTML5 mínimo

- Todo documento HTML debe empezar por un *doctype* que especifica qué tipo de información contiene. En HTML5 es `<!DOCTYPE html>`.
- Sirve para indicar al navegador qué especificación de HTML se está usando.
- La etiqueta `<html>` indica al navegador que su contenido es un documento HTML.
- Puede tener un atributo que especifica el idioma de la información que contiene el documento:

```
1 <!DOCTYPE html>
2 <html lang="es">
3 <head>
4 </head>
5 </html>
```

El documento HTML5 mínimo

La etiqueta `<head>...</head>` permite indicar información sobre el documento:

- El título del documento con la etiqueta `<title>...</title>`.
- Declaraciones o llamadas a hojas de estilo CSS.
- Funciones o vínculos a archivos con instrucciones en JavaScript.
- Información para los motores de búsqueda (para su mejor indexación).
- Información a los navegadores.
- Meta-etiquetas.

El documento HTML5 mínimo

- La codificación de un documento html `<meta charset>` hace referencia a los caracteres representables en el documento:
 - ASCII: juego de caracteres de toda la vida. El más básico.
 - ISO-8859-1: es una extensión del anterior. También se denomina *Latin-1*. Sirve para expresar la información en las lenguas occidentales.
 - ISO-8859-15: es una actualización del anterior. Incluye, por ejemplo, el símbolo del euro.
 - UTF-8: permite expresar la información en casi cualquier alfabeto. Es el más recomendado por su compatibilidad.
- El cuerpo del documento, es decir, la información a visualizar junto con el resto de instrucciones se incluyen en el cuerpo de la página html. Su etiqueta es `<body>...</body>`.

El documento HTML5 mínimo

Juntándolo todo obtenemos nuestra primera página en HTML5:

```
<!DOCTYPE html>
<html lang="es">
<head>
<title>Primer página</title>
<meta charset="iso-8859-1">
</head>
<body>
Esta es mi primera página HTML5.
</body>
</html>
```

¿Qué son las hojas de estilo?

- Idea clave: separación entre el contenido y su presentación.
- Objetivo: permitir la programación de diferentes presentaciones para un mismo contenido dependiendo del dispositivo utilizado: ordenador, teléfono móvil, documento imprimible, etc.
- Las hojas de estilo son elementos agregados a HTML que especifican la presentación de la página web.
- Hoja de estilo en cascada o CSS (siglas en inglés de *Cascading Style Sheets*) es un lenguaje usado para definir la presentación de un documento estructurado escrito en HTML.

HTML4 vs HTML5

- HTML4 era rico en etiquetas con atributos de estilo, HTML5 no.
- HTML4 no obligaba a una separación tan estricta entre contenido y visualización. HTML5, por el contrario, proporciona una simplificación del código.
- HTML5 proporciona un mantenimiento del sitio web más sencillo.

- Las hojas de estilo tienen que ver con todo aquello relacionado con la presentación "estética" de un documento HTML: color de texto y de fondo, alineación, fuente, tamaño de fuente, etc.
- La declaración de un estilo se hace a través de declaraciones con la sintaxis:

`propiedad: valor;`

Por ejemplo: *text-align: center;* especifica que el texto va a estar centrado.

Selectores básicos:

- Los *selectores* son aquellos elementos de HTML que se ven afectados por la presentación. La especificación de un elemento de estilo se restringe a estos elementos de acuerdo con la siguiente sintaxis.
- Sintaxis: `selector {propiedad: valor;}`.
- Ejemplo: `h1 {text-align: center;}`, centra el texto de los elementos `h1`.

Selectores de clase:

- En muchas ocasiones no queremos aplicar el mismo estilo a todos los elementos de cierto tipo. Es posible un tratamiento particular de ellos utilizando la noción de clase (*class*).
- Para que declaración del estilo funcione como queremos es necesario que los elementos que queremos seleccionar se distingan de los demás.
- Sintaxis: `.nombre_clase {propiedad: valor;}`

Ejemplo:

- De todas las etiquetas `h1` solo queremos seleccionar el estilo de alguna de ellas.
- Aquellas que queramos seleccionar deben llevar la marca que permita distinguirla (cierto valor para `class`). Ejemplo: `<h1 class='efecto1'>...</h1>`.
- Con la instrucción `.efecto1 { text-align: center; }` se alinean todos los textos de los elementos que tengan la clase `efecto1`.
- Elementos de diferente tipo pueden tener la misma clase.

Selectores de identificador:

- Es una noción muy similar a la de clase. La diferencia consiste en que ahora no se desea que haya más de un elemento seleccionado.
- Sintaxis: `#identificador {propiedad: valor;}`
- Ejemplo:
 - Marcamos la etiqueta: `<h2 id='nombre'>...</h2>` y
 - Especificamos el estilo para este identificador: `#identificador {propiedad: valor;}`.
 - Se da por hecho que solo existe un elemento con el identificador *nombre*.

Dónde indicar el estilo

- Estilo en línea:
 - La instrucción del estilo va dentro de la instrucción HTML.
 - Ejemplo: `<h1 style=''text-align: center;''>...</h1>`
 - No respeta la regla de separación entre contenido y presentación. No es la forma recomendada. Solo uso excepcional.
- Estilo en la cabecera:
 - Todas las instrucciones de estilo van en la cabecera dentro de la etiqueta `<style>...</style>`.
 - Respeta la regla de separación entre contenido y presentación. Es la forma recomendada. Uso general.

Ejemplo:

```
<!DOCTYPE html>
<html lang="es">
<head>
<title>Primer página</title>
<meta charset="iso-8859-1">
<style type="text/css">
h1 { color: red; }
.efecto1 { font-style: italic; }
</style>
</head>
<body>
Esta es mi primera página HTML5.
</body>
</html>
```

- Todo el texto a visualizar en una página HTML debe ir dentro de la etiqueta `<body>...</body>`.
- Los retornos de carro o saltos de línea que aparecen en el código se ignoran. Para forzar un salto de línea es necesario utilizar la etiqueta `
`.
- La negrita y la cursiva se realizan con las etiquetas `...` y `<i>...</i>`.
- Para expresar exponentes y subíndices se utilizan las etiquetas: `^{...}` y `_{...}`.
- Para "tachar" cierto texto existe `...`.

El texto en HTML5

- La etiqueta `<pre>...</pre>` (texto preformateado) permite visualizar el texto tal y como se ha escrito (respetando los espacios, las tabulaciones y los saltos de línea).
- Comentarios: `<!-- Esto es un comentario -->`
- Los caracteres utilizados en la sintaxis de HTML no pueden ser utilizados directamente en la edición de texto. Es preciso especificarlos de otra forma.

Ejemplo:

<	<
>	>
"	"
&	&

Dando formato de texto con CSS

Negrita:

```
<!DOCTYPE html>
<html lang="es">
<head>
<title>Html5</title>
<meta charset="UTF-8">
<style type="text/css">
div { font-weight: bold;}
</style>
</head>
<body>
Texto en <div class="negrita">negrita</div>
</body>
</html>
```

Dando formato de texto con CSS

Cursiva:

```
<!DOCTYPE html>
<html lang="es">
<head>
<title>Html5</title>
<meta charset="UTF-8">
<style type="text/css">
.cursiva { font-style : italic;}
</style>
</head>
<body>
Texto en <div class="cursiva">cursiva</div>
</body>
</html>
```

Dando formato de texto con CSS

Tamaño de fuente:

```
<!DOCTYPE html>
<html lang="es">
<head>
<title>Html5</title>
<meta charset="UTF-8">
<style type="text/css">
#grande { font-size: 36px;}
</style>
</head>
<body>
<div id="grande">Texto en grande</div>
</body>
</html>
```


Dando formato de texto con CSS

Color:

```
<!DOCTYPE html>
<html lang="es">
<head>
<title>Html5</title>
<meta charset="UTF-8">
<style type="text/css">
h1 {color: red;}
</style>
</head>
<body>
<h1>Titulo en rojo</h1>
</body>
</html>
```

Dando formato de texto con CSS

Alineación:

```
<!DOCTYPE html>
<html lang="es">
<head>
<title>Html5</title>
<meta charset="UTF-8">
<style type="text/css">
#centrado { text-align: center;}
</style>
</head>
<body>
<div id="centrado">Texto alineado al centro</div>
</body>
</html>
```

Dando formato de texto con CSS

Subrayado:

```
<!DOCTYPE html>
<html lang="es">
<head>
<title>Html5</title>
<meta charset="UTF-8">
<style type="text/css">
.subrayado { text-decoration: underline;}
</style>
</head>
<body>
Texto <div class="subrayado">subrayado</div>
</body>
</html>
```

Estructura del documento

Títulos:

- HTML5 proporciona seis tamaños de letra para los títulos: desde el 1 (el más grande) hasta el 6 (el más pequeño).
- Sintaxis: `<hx>...</hx>`
- Ejemplo:

```
1  <!DOCTYPE html>
2  <html lang="es">
3  <head>
4  <title>Html5</title>
5  <meta charset="UTF-8">
6  </head>
7  <body>
8  <h1>Titulo de nivel 1</h1>
9  <h2>Titulo de nivel 2</h2>
10 <h3>Titulo de nivel 3</h3>
11 <h4>Titulo de nivel 4</h4>
12 <h5>Titulo de nivel 5</h5>
13 <h6>Titulo de nivel 6</h6>
14 </body>
15 </html>
```

Párrafos y citas:

- Los párrafos y citas añaden legibilidad a un texto.
- Sintaxis de los párrafos: `<p>...</p>`
- Sintaxis de las citas: `<blockquote>...</blockquote>`.
- A veces es muy útil una línea horizontal: `<hr>`.

Estructura del documento

Listas ordenadas:

- Sintaxis:

```
1 <ol>
2   <li> ... </li>
3   <li> ... </li>
4   ...
5   <li> ... </li>
6 </ol>
```

Estructura del documento

Listas ordenadas:

- Ejemplo:

```
1  <!DOCTYPE html>
2  <html lang="es">
3  <head>
4  <title>Html5</title>
5  <meta charset="UTF-8">
6  </head>
7  <body>
8  Listas en Html5:
9  <ol>
10 <li>Listas ordenadas</li>
11 <li>Listas no ordenadas</li>
12 <li>Listas anidadas</li>
13 <li>Listas de definición</li>
14 <li>Menú de listas</li>
15 </ol>
16 </body>
17 </html>
```

Estructura del documento

Listas no ordenadas:

- Sintaxis:

```
1 <ul>
2   <li> ... </li>
3   <li> ... </li>
4   ...
5   <li> ... </li>
6 </ul>
```


Estructura del documento

Listas no ordenadas:

- Ejemplo:

```
1  <!DOCTYPE html>
2  <html lang="es">
3  <head>
4  <title>Html5</title>
5  <meta charset="UTF-8">
6  </head>
7  <body>
8  Listas en Html5:
9  <ul>
10 <li>Listas ordenadas</li>
11 <li>Listas no ordenadas</li>
12 <li>Listas anidadas</li>
13 <li>Listas de definición</li>
14 <li>Menú de listas</li>
15 </ul>
16 </body>
17 </html>
```

Estructura del documento

Lista de definiciones:

- Muy útil para hacer glosarios de términos.

```
1  <!DOCTYPE html>
2  <html lang="es">
3  <head>
4  <title>Html5</title>
5  <meta charset="UTF-8">
6  </head>
7  <body>
8  <p>Etiquetas de lista disponibles:</p>
9  <dl>
10 <dt>kiwi</dt>
11 <dd>bla bla</dd>
12 <dt>ornitorrinco</dt>
13 <dd>bla bla</dd>
14 </dl>
15 </body>
16 </html>
```

- Los enlaces proporcionan un primer grado de interacción con el usuario. Los enlaces constituyen una red que constituye la Web.
- Sintaxis: `texto del enlace`
- El destino del enlace puede ser:
 - Un lugar de la propia página web.
 - Otra página del mismo sitio web.
 - Una página de otro sitio web.
 - Un archivo para descargar.

- Atributos de la etiqueta de enlace:
 - *href*: dirección (url) del enlace.
 - *hreflang*: idioma del documento del destino si es diferente al de origen.
 - *ping*: puede contener una lista de direcciones (separadas por espacios) que reciben una notificación cuando el usuario sigue el enlace. Muy útil para elaborar estadísticas.
 - *target*: especifica al navegador la forma de visualizar el destino de enlace:
 - `target='_blank'`: una nueva instancia el navegador.
 - `target='_self'`: una nueva pestaña en el navegador

Enlace hacia una ventana específica:

```
<!DOCTYPE html>
<html lang="es">
<head>
<title>Html5</title>
<meta charset="UTF-8">
</head>
<body>
Ir a <a href="pagina2.htm" target="_blank">Página 2</a>
</body>
</html>
```

Enlace con color y subrayado (alterando la apariencia por defecto):

```
<!DOCTYPE html>
<html lang="es">
<head>
<title>Html5</title>
<style type="text/css">
a { text-decoration: none;
    color: black;}
</style>
<meta charset="UTF-8">
</head>
<body>
Ir a <a href="pagina1.htm">Página 1</a><br>
Ir a <a href="pagina2.htm">Página 2</a><br>
Ir a <a href="pagina3.htm">Página 3</a><br>
</body>
</html>
```

Los enlaces pueden contener cualquier elemento de tipo bloque: `<div>` y `` por ejemplo.

```
<!DOCTYPE html>
<html lang="es">
<head>
<title>Html5</title>
<style>
div { width: 280px;
      height: 100px;
      border: 1px solid black;
      padding-left: 3px;}
a { text-decoration: none;
    color: black;}
</style>
<meta charset="UTF-8">
</head>
```

Los enlaces pueden contener cualquier elemento de tipo bloque: `<div>` y `` por ejemplo.

```
<body>
<p>Haga clic en el extracto para leer
el texto completo.</p>
<a href="pagina2.htm">
<div>Lorem ipsum dolor sit amet, consectetur
adipiscing elit. Sed non risus.
Suspendisse lectus tortor, dignissim sit amet,
adipiscing nec,
ultricies sed, dolor. Cras elementum
ultrices diam...</div>
</a>
</body>
</html>
```


- Una tabla (`<table>...</table>`) se compone de filas (`<tr>...</tr>`) que, a su vez, se componen de celdas (`<td>...</td>`).
- En cada celda se coloca la información a mostrar.
- Las celdas de encabezado (`<th>...</th>`) son un tipo particular de celdas con contenidos destacados visualmente. Se utilizan para indicar títulos en filas y columnas.
- La etiqueta `<caption>...</caption>` permite asociar un título o leyenda a una tabla.

Ejemplo de tabla:

```
<!DOCTYPE html>
<html lang="es">
<head>
<title>Html5</title>
<meta charset="UTF-8">
</head>
<body>
<table>
<tr>
<td>1</td><td>2</td>
</tr>
<tr>
<td>3</td><td>4</td>
</tr>
</table>
</body>
</html>
```

Modificando el formato de la tabla:

- El ancho de la tabla se define mediante la propiedad CSS width.
Ejemplo: `table { width: 250px; }`
- De forma análoga se especifica el alto de la tabla: `table { height: 250px; }`
- Tanto para las celdas como para las tablas enteras se puede modificar el "borde":
 - Tabla: `table { border: 1px solid black; }`
 - Celdas: `td { border: 1px solid black; }`

- Alineación de las celdas: con la propiedad *text-align*.

Ejemplo:

```
table { text-align: center; border: 1px solid black; border-collapse: collapse; width: 250px; }
```

- De forma similar se puede modificar las dimensiones de las celdas.
- Margen interior (la distancia del texto al borde de la celda) (*padding*).
Ejemplo: `td {padding: 15px;}`.

- En muchas ocasiones las tablas no tienen un aspecto de cuadrícula estricta.
- Fusión de filas y columnas:
 - Para fusionar columnas se modifica el atributo *colspan*.
 - Para fusionar filas *rowspan*.

```
<!DOCTYPE html>
<html lang="es">
<head>
<title>Html5</title>
<meta charset="UTF-8">
<style>
table { border: 1px solid black;
        border-collapse: collapse;
        width: 250px;}
td { border: 1px solid black;}
</style>
</head>
<body>
<table>
<tr>
<td colspan="3" style="text-align: center;">Título de la tabla</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>6</td>
</tr>
</table>
</body>
</html>
```

Tablas

```
<!DOCTYPE html>
<html lang="es">
<head>
<title>Html5</title>
<meta charset="UTF-8">
<style>
table { border: 1px solid black;
        border-collapse: collapse;
        width: 250px;}
td { border: 1px solid black;}
</style>
</head>
<body>
<table>
<tr>
<td rowspan="2">1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>5</td>
<td>6</td>
</tr>
</table>
</body>
</html>
```

- Para asegurar un tiempo de descarga razonable la Web exige los formatos de imagen: GIF, JPEG, PNG.
- Insertar una imagen. Ejemplo:

```
1  <!DOCTYPE html>
2  <html lang="es">
3  <head>
4  <title>Html5</title>
5  <meta charset="UTF-8">
6  </head>
7  <body>
8  
9  </body>
10 </html>
```


Insertar un enlace sobre una imagen. Ejemplo:

```
<!DOCTYPE html>
<html lang="es">
<head>
<title>Html5</title>
<meta charset="UTF-8">
</head>
<body>
<a href=globo.htm>

</a>
</body>
</html>
```

Cambiar el color de fondo:

```
<!DOCTYPE html>
<html lang="es">
<head>
<title>Html5</title>
<meta charset="UTF-8">
<style>
body {background-color: rgb(200,215,230);}
</style>
</head>
<body>
</body>
</html>
```

Insertar imagen de fondo:

```
<!DOCTYPE html>
<html lang="es">
<head>
<title>Html5</title>
<meta charset="UTF-8">
</head>
<style>
body {background-image: url('bg.gif');}
</style>
<body>
</body>
</html>
```

¿Para qué valen los formularios?

- Los formularios enriquecen la interacción de las páginas web con el usuario. Es una forma de envío de información estructurada por parte del usuario.
- La validación, el envío de información correcta, es muy importante. Existen varios niveles en los cuales realizar esta validación:
 - A nivel de HTML.
 - A nivel de JavaScript.
 - Desde el lado del servidor: con PHP, por ejemplo.
- En general, la información enviada con el formulario se procesa desde el lado del servidor mediante lenguajes como PHP y acceso a base de datos (como MySQL).

Estructura de los formularios

- La declaración de un formulario se realiza mediante la etiqueta `<form>...</form>`
- Ejemplo:

```
1  <!DOCTYPE html>
2  <html lang="es">
3      <head>
4          <title>Html5</title>
5          < meta charset="UTF-8">
6      </head>
7      <body>
8          <form action="">
9              Elementos del formulario
10         </form>
11     </body>
12 </html>
```

- Atributos de la etiqueta `<form>...</form>`:
 - *name*: permite asignar un nombre al formulario.
 - *action*: permite especificar qué tipo de acción se debe realizar sobre los datos:
 - Procesamiento de los datos por parte del servidor:
`action='dirección'`.
 - *enctype*: especifica el formato de los datos.
 - *method*: identifica el protocolo de transmisión de datos:
 - GET: transferencia de los datos en caracteres ASCII (máximo 100 caracteres).
 - POST: permite caracteres no ASCII (número ilimitado de caracteres).

Campos de texto

- Permiten la introducción de información alfanumérica.
- Sintaxis: `<input type='text'>`
- Ejemplo:

```
1 <!DOCTYPE html>
2 <html lang="es">
3   <head>
4     <title>Html5</title>
5     <meta charset="UTF-8">
6   </head>
7   <body>
8     <form action="">
9       Nombre: <input type="text">
10    </form>
11  </body>
12 </html>
```

- Atributos:

- *name*: permite asignar un nombre al campo de texto.
- *size*: determina el número de caracteres visibles. El valor por defecto es 20.
- *maxlength*: determina el número máximo de caracteres que se puede introducir.
- *value*: define el valor por defecto. Aparece cuando se carga la página.
- *required*: determina que obligatoriamente ese campo de texto no puede quedar en blanco.
- *pattern*: determina un formato para la información. Ejemplo: `<input type='text' pattern='[0-9]{5}'>` exige que haya 5 dígitos comprendidos entre el 0 y el 9.
- *height* y *width*: representan sus dimensiones.
- Si especificamos `<input type='password'>` tratará la información como si fuera una contraseña (aunque no protege los datos que se transmitirán si cifrar)
- Existen más atributos aunque no todos los navegadores los soportan.

Algunos valores para el atributo type:

Valor	Significado
password	Introducción de contraseñas
hidden	Texto oculto al usuario

Campos de texto

HTML5 además permite los siguientes valores para el atributo type:

Valor	Significado
color	Especifica un color
date	Especifica una fecha
datetime-local	Especifica una fecha y una hora
email	Exige que tenga formato válido
month	Especifica un mes
range	Especifica un rango
tel	Exige que tenga formato válido
time	Especifica una hora
url	Exige que tenga formato válido
week	Especifica una semana

Nota: no están soportados por todos los navegadores. Puedes ver más valores del atributo type en

https://www.w3schools.com/html/html_form_input_types.asp

Áreas de texto

- Permiten introducir textos más largos.
- Ejemplo:

```
1 <!DOCTYPE html>
2 <html lang="es">
3   <head>
4     <title>Html5</title>
5     <meta charset="UTF-8">
6   </head>
7   <body>
8     <form action="">
9       <p>Comentarios: </p>
10      <textarea rows="4" cols="30"></textarea>
11    </form>
12  </body>
13 </html>
```

- Atributos:
 - *name*: permite asignar un nombre al área de texto.
 - *cols* y *rows* define la filas y columna.
 - Los atributos *maxlength* y *required* también valen para las áreas de texto.

Listas desplegables

- Proporcionan al usuario una lista de opciones:
- Ejemplo:

```
1 <!DOCTYPE html>
2 <html lang="es">
3   <head>
4     <title>Html5</title>
5     <meta charset="UTF-8">
6   </head>
7   <body>
8     <form action="">
9       <p>Su navegador preferido:
10       <select>
11         <option value="1">Internet Explorer</option>
12         <option value="2">Firefox</option>
13         <option value="3">Safari</option>
14         <option value="4">Opera</option>
15         <option value="5">Google Chrome</option>
16       </select>
17     </p>
18   </form>
19 </body>
20 </html>
```

Atributos:

- *name*: permite asignar un nombre a la lista desplegable.
- *size*: número de elementos visibles.
- *multiple*: permite seleccionar más de un elemento.
- *selected*: especifica qué elemento es preseleccionado en la lista. Si no se especifica se preselecciona el primero.
- Si no se indica nada es el texto de la etiqueta `<option>` el transmitido. Si se quiere enviar otro distinto es necesario indicarlo con el atributo *value*.

Botones de selección única

- Permiten seleccionar una opción entre varias.
- Ejemplo:

```
1 <!DOCTYPE html>
2 <html lang="es">
3   <head>
4     <title>Html5</title>
5     <meta charset="UTF-8">
6   </head>
7   <body>
8     <form action="">
9       Sexo:<br>
10      <input type="radio" name="sexo">Masculino<br>
11      <input type="radio" name="sexo">Femenino
12    </form>
13  </body>
14 </html>
```

Atributos:

- *name*: permite asignar un nombre al conjunto de opciones. En este caso el nombre es obligatorio.
- *checked*: permite preseleccionar una opción.
- *value*: asigna un valor a una opción (para un procesamiento posterior).

Casillas de verificación

- Permiten seleccionar varias opciones.
- Ejemplo:

```
1  <!DOCTYPE html>
2  <html lang="es">
3    <head>
4      <title>Html5</title>
5      <meta charset="UTF-8">
6    </head>
7    <body>
8      <form action="">
9        Su pizza con:<br>
10       <input type="checkbox" name="n1"> Mozzarella<br>
11       <input type="checkbox" name="n2"> Jamón<br>
12       <input type="checkbox" name="n3" checked> Salsa de tomate<br>
13       <input type="checkbox" name="n4" checked> Pimientos<br>
14       <input type="checkbox" name="n5"> Champiñones<br>
15       <input type="checkbox" name="n6"> Olivas negras<br>
16       <input type="checkbox" name="n7"> Salsa picante
17     </form>
18   </body>
19 </html>
```


Atributos:

- *name*: permite asignar un nombre a el conjunto de opciones. En este caso el nombre es obligatorio.
- *checked*: permite preseleccionar una opción.
- *value*: asigna un valor a una opción (para un procesamiento posterior).

Botones

- De envío y cancelación: permiten enviar o cancelar la información de todos los campos:

```
1 <input type="submit" value="Enviar" />
2 <input type="reset" value="Borrar" />
```

- Genéricos: permiten asociarles código JavaScript. Pueden estar fuera del formulario.
- Ejemplo:

```
1 <!DOCTYPE html>
2 <html lang="es">
3   <head>
4     <title>Html5</title>
5     <meta charset="UTF-8">
6   </head>
7   <body>
8     <button onclick="alert('Hola')">Pinche aquí </button>
9   </body>
10 </html>
```

Etiquetado de los campos de formulario

- Facilitan la interacción con el formulario.
- Con la etiqueta `<label>...</label>` se permite seleccionar un botón de radio o casilla de verificación haciendo click en el texto adyacente.
- Ejemplo:

```
1  <!DOCTYPE html>
2  <html lang="es">
3    <head>
4      <title>Html5</title>
5      <meta charset="UTF-8">
6    </head>
7    <body>
8      <form action="">
9        <p>Forma de pago:<br>
10       <label for="cash1">Visa</label>
11       <input type="radio" name="cash" id="cash1"><br>
12       <label for="cash2">American Express</label>
13       <input type="radio" name="cash" id="cash2"><br>
14       <label for="cash3">Mastercard</label>
15       <input type="radio" name="cash" id="cash3">
16     </p>
17   </form>
18 </body>
19 </html>
```

Organización de los campos de un formulario

- Cuando el formulario es complejo es necesario estructurar la información mostrada al usuario.
- La etiqueta `<fieldset>` engloba campos de un formulario y los rodea con un borde.
- La etiqueta `<legend>` agrega una segunda leyenda.

Organización de los campos de un formulario

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>Html5</title>
    <meta charset="UTF-8">
  </head>
  <body>
    <form action="">
      <fieldset>
        <legend>Nuestros entrantes</legend>
        <input type="checkbox" name="n1"> Chikenito's<br>
        <input type="checkbox" name="n2"> Ensalada de temporada<br>
        <input type="checkbox" name="n3"> Buffalo wings<br>
      </fieldset>
      <fieldset>
        <legend>Nuestras pizzas</legend>
        <input type="checkbox" name="n4"> Clásica<br>
        <input type="checkbox" name="n5"> Campesina<br>
        <input type="checkbox" name="n6"> Diabólica<br>
      </fieldset>
      <fieldset>
        <legend>Nuestros postres</legend>
        <input type="checkbox" name="n7"> Helados variados<br>
        <input type="checkbox" name="n8"> Cookies<br>
        <input type="checkbox" name="n9"> Tarta de chocolate<br>
      </fieldset>
      <input type="submit"> <input type="reset">
    </form>
  </body>
</html>
```

Campo de texto con lista de sugerencias

- La etiqueta `<datalist>` agregada a un campo de texto abre una lista de sugerencias.
- Ejemplo:

```
1 <!DOCTYPE html>
2 <html lang="es">
3   <head>
4     <title>Html5</title>
5     <meta charset="UTF-8">
6   </head>
7   <body>
8     <form action="">
9       Escoja una fruta:
10      <input type="text" list="frutas">
11      <datalist id="frutas">
12        <option value="Naranja"></option>
13        <option value="Manzana"></option>
14        <option value="Pera"></option>
15        <option value="Ciruela"></option>
16      </datalist>
17    </form>
18  </body>
19 </html>
```

Campo de texto con formato numérico

- La etiqueta `<input type='number'>` permite contadores numéricos de lectura.
- Ejemplo:

```
1  <!DOCTYPE html>
2  <html lang="es">
3    <head>
4      <title>Html5</title>
5      <meta charset="UTF-8">
6    </head>
7    <body>
8      <form action="">
9        Numero de artículos (grupos de 2):<br>
10       <input type="number" name="q" min="2" max="10" step="2" value="2">
11     </form>
12   </body>
13 </html>
```

¿Dónde indicar el estilo?

Instrucciones CSS integradas en un elemento HTML5:

- Aunque no es lo deseable es posible agregar directamente el código de una declaración de estilo en una etiqueta determinada:

```
1  <!DOCTYPE html>
2  <html lang="es">
3    <head>
4      <title>Html5</title>
5      <meta charset="UTF-8">
6    </head>
7    <body>
8      <h1 style="font-style: italic;">Titulo 1</h1>
9      <h1>Titulo 2</h1>
10   </body>
11 </html>
```


¿Dónde indicar el estilo?

Instrucciones CSS internas en un documento HTML5:

- Dado que el alcance de las reglas de estilo es el documento entero, es posible especificarlas en la cabecera del documento:

```
1  <!DOCTYPE html>
2    <html lang="es">
3      <head>
4        <title>Html5</title>
5        <meta charset="UTF-8">
6        <style type="text/css">
7          h1 { font-style: italic;}
8        </style>
9      </head>
10     <body>
11       <h1>Titulo 1</h1>
12       <h1>Titulo 2</h1>
13     </body>
14  </html>
```

¿Dónde indicar el estilo?

Instrucciones CSS externas a un documento HTML5:

- Como es razonable pensar en aplicar el mismo estilo a varias páginas web, es posible almacenar el código de estilo en un archivo diferente e importarlo.
- De esta forma se puede cambiar rápidamente el estilo de una página web cambiando el archivo CSS importado.
- Un ejemplo:

```
1 <!DOCTYPE html>
2 <html lang="es">
3   <head>
4     <title>Html5</title>
5     <meta charset="UTF-8">
6     <link rel="stylesheet" type="text/css" href="ejemplo.css">
7   </head>
8   <body>
9     <h1>Titulo de nivel 1</h1>
10  </body>
11 </html>
```

¿Dónde indicar el estilo?

- Aquí el estilo importado es solamente la declaración: `h1 { text-decoration: overline; }`.
- Comentarios sobre: `<link rel="stylesheet" type="text/css" href="ejemplo.css">`:
 - `link`: indica al navegador que se trata de un enlace.
 - `rel="stylesheet" type="text/css"`: indica que se trata de una hoja de estilo CSS.
 - `href="ejemplo.css"`: indica qué fichero importar.

Noción de cascada (Cascading Style Sheets)

- Existen varias formas de introducir el estilo en nuestras páginas web: en línea, interna o externa.
- En páginas web suficientemente complejas surgirá la necesidad de combinar estas posibilidades.
- La noción de *cascada* determina la prioridad de las formas de introducir el estilo:
 - Primero se consideran las especificaciones de las hojas de estilo externas (ficheros CSS).
 - A continuación, las internas (las que están en la cabecera).
 - Y por último, la especificaciones en línea (dentro de cada etiqueta).
 - NOTA: Existe un cuarto nivel formado por las propiedades de estilo por defecto del navegador.
 - En caso de conflicto tenemos la siguiente prioridad (de menor a

mayor):

Máximo	especificaciones en línea
	especificaciones internas
	especificaciones externas
Mínimo	Propiedades por defecto del navegador

Tipografía

- CSS permite controlar fácilmente la tipografía del texto.
- Ejemplos: Especificación de la nombre de la fuente:

```
1 <!DOCTYPE html>
2 <html lang="es">
3   <head>
4     <title>Html5</title>
5     <meta charset="UTF-8">
6     <style type="text/css">
7       .serif {font-family: serif;}
8       .sans-serif {font-family: sans-serif;}
9       .cursive {font-family: cursive;}
10      .fantasy {font-family: fantasy;}
11      .monospace {font-family: monospace;}
12    </style>
13  </head>
14  <body>
15    <p class="serif">Texto en serif</p>
16    <p class="sans-serif">Texto en sans-serif</p>
17    <p class="cursive">Texto en cursive</p>
18    <p class="fantasy">Texto en fantasy</p>
19    <p class="monospace">Texto en monospace</p>
20  </body>
21 </html>
```

Especificación del estilo de la fuente:

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>Html5</title>
    <meta charset="UTF-8">
    <style type="text/css">
      .italic {font-style: italic;}
      .oblique {font-style: oblique;}
    </style>
  </head>
  <body>
    <h2 class="italic">Título en italic</h2>
    <h2 class="oblique">Título en oblique</h2>
  </body>
</html>
```

Diversas formas de especificar la negrita:

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>Html5</title>
    <meta charset="UTF-8">
    <style type="text/css">
      .bold { font-weight: bold;}
      .bolder { font-weight: bolder;}
      .bold900 { font-weight: 900;}
    </style>
  </head>
  <body>
    <p class="bold">Texto en negrita</p>
    <p class="bolder">Texto en negrita</p>
    <p class="bold900">Texto en negrita</p>
  </body>
</html>
```

Especificación del tamaño de la fuente:

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>Html5</title>
    <meta charset="UTF-8">
    <style type="text/css">
      .tamaño {font-size: 96px;}
    </style>
  </head>
  <body>
    <h1>En Html5</h1>
    <p class="tamaño">En CSS</p>
  </body>
</html>
```


Especificación "compacta":

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>Html5</title>
    <meta charset="UTF-8">
    <style type="text/css">
      p {font: italic bold small-caps 24pt Arial, sans-serif;}
    </style>
  </head>
  <body>
    <p>Fuente abreviada</p>
  </body>
</html>
```

Texto subrayado:

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>Html5</title>
    <meta charset="UTF-8">
    <style type="text/css">
      .deco1 { text-decoration: overline;}
      .deco2 { text-decoration: underline;}
      .deco3 { text-decoration: underline overline;}
      .deco4 { text-decoration: line-through;}
    </style>
  </head>
  <body>
    <p class="deco1">Texto con subrayado superior</p>
    <p class="deco2">Texto con subrayado inferior</p>
    <p class="deco3">Texto con subrayado inferior y subrayado superior</p>
    <p class="deco4">Texto tachado</p>
  </body>
</html>
```

Tratamiento de las mayúsculas:

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>Html5</title>
    <meta charset="UTF-8">
    <style type="text/css">
      .transformacion1 { text-transform: capitalize;}
      .transformacion2 { text-transform: uppercase;}
      .transformacion3 { text-transform: lowercase;}
    </style>
  </head>
  <body>
    <h2 class="transformacion1">texto en minúsculas</h2>
    <h2 class="transformacion2">texto en minúsculas</h2>
    <h2 class="transformacion3">TEXTO EN MAYÚSCULAS</h2>
  </body>
</html>
```

Indentación:

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>Html5</title>
    <meta charset="UTF-8">
    <style type="text/css">
      .indentacion { text-indent: 30px;}
    </style>
  </head>
  <body>
    <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed non risus.
      Suspendisse lectus...
    </p>
    <p class="indentacion">Lorem ipsum dolor sit amet, consectetur adipiscing elit.
      Sed non risus. Suspendisse lectus...
    </p>
  </body>
</html>
```

Interlineado:

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>Html5</title>
    <meta charset="UTF-8">
    <style type="text/css">
      .interlineado { line-height: 200%;}
    </style>
  </head>
  <body>
    <p class="interlineado">
      Lorem ipsum dolor sit amet, consectetur adipiscing<br>
      elit. Sed non risus. Suspendisse lectus tortor, dignissim<br>
      sit amet, adipiscing nec, ultricies sed, dolor. Cras<br>
      elementum ultrices diam. Maecenas ligula massa,...
    </p>
    <p>
      Lorem ipsum dolor sit amet, consectetur adipiscing<br>
      elit. Sed non risus. Suspendisse lectus tortor, dignissim<br>
      sit amet, adipiscing nec, ultricies sed, dolor. Cras<br>
      elementum ultrices diam. Maecenas ligula massa,...
    </p>
  </body>
</html>
```

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>Html5</title>
    <meta charset="UTF-8">
    <style type="text/css">
      .style1 { list-style-type:disc;}
      .style2 { list-style-type:circle;}
      .style3 { list-style-type:square;}
      .style4 { list-style-type:decimal;}
      .style5 { list-style-type: decimal-leading-zero;}
      .style6 { list-style-type:upper-roman;}
      .style7 { list-style-type:lower-roman;}
      .style8 { list-style-type:upper-alpha;}
      .style9 { list-style-type:lower-alpha;}
    </style>
  </head>
  <body>
    <ul>
      <li class="style1">Disco completo</li>
      <li class="style2">Círculo</li>
      <li class="style3">Cuadrado</li>
    </ul>
    <ol>
      <li class="style4">Decimal</li>
      <li class="style5">Decimal</li>
      <li class="style6">Romanos en mayúsculas</li>
      <li class="style7">Romanos en minúsculas</li>
      <li class="style8">Alfabeto en mayúsculas</li>
      <li class="style9">Alfabeto en minúsculas</li>
```

Ejemplo de la propiedad list-style-image:

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>Html5</title>
    <meta charset="UTF-8">
    <style type="text/css">
      ul { list-style-image: url(check.gif)}
    </style>
  </head>
  <body>
    <ul>
      <li>Elemento 1</li>
      <li>Elemento 2</li>
      <li>Elemento 3</li>
      <li>Elemento 4</li>
    </ul>
  </body>
</html>
```

Ejemplo de la propiedad list-style-position:

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>Html5</title>
    <meta charset="UTF-8">
    <style type="text/css">
      .posicion1 { list-style-position: outside}
      .posicion2 { list-style-position: inside}
    </style>
  </head>
  <body>
    <ul>
      <li class="posicion1">Elemento 1<br />
        Siguiete elemento 1<br />
        Siguiete elemento 1</li>
    </ul>
    <ul>
      <li class="posicion2">Elemento 2<br />
        Siguiete elemento 2<br />
        Siguiete elemento 2</li>
    </ul>
  </body>
</html>
```


Ejemplo de la propiedad border-spacing:

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>Html5</title>
    <meta charset="UTF-8">
    <style type="text/css">
      table { width: 200px;
              text-align: center;
              border : 1px solid gray;
              border-spacing: 10px;}
      td { border : 1px solid gray;
           background-color: rgb(213,230,245)}
    </style>
  </head>
  <body>
    <table>
      <tr>
        <td>1</td><td>2</td><td>3</td>
      </tr>
      <tr>
        <td>4</td><td>5</td><td>6</td>
      </tr>
      <tr>
        <td>7</td><td>8</td><td>9</td>
      </tr>
    </table>
  </body>
</html>
```

Ejemplo de la propiedad border-collapse:

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>Html5</title>
    <meta charset="UTF-8">
    <style type="text/css">
      table { width: 200px;
        text-align: center;
        border : 1px solid gray;
        border-collapse: collapse;}
      td { border : 1px solid gray;}
    </style>
  </head>
  <body>
    <table>
      <tr>
        <td>1</td><td>2</td><td>3</td>
      </tr>
      <tr>
        <td>4</td><td>5</td><td>6</td>
      </tr>
      <tr>
        <td>7</td><td>8</td><td>9</td>
      </tr>
    </table>
  </body>
</html>
```

Ejemplo de la propiedad empty-cells:

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>Html5</title>
    <meta charset="UTF-8">
    <style type="text/css">
      table { width: 200px;
              text-align: center;}
      td { border : 1px solid gray;
          background-color: rgb(213,230,245);}
      .show {empty-cells: show;}
      .hide {empty-cells: hide;}
    </style>
  </head>
  <body>
    <table>
      <tr>
        <td class="show"></td><td>2</td><td>3</td>
      </tr>
      <tr>
        <td>4</td><td>5</td><td>6</td>
      </tr>
      <tr>
        <td>7</td><td>8</td><td class="hide"></td>
      </tr>
    </table>
  </body>
</html>
```

Ejemplo de la propiedad caption-side:

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>Html5</title>
    <meta charset="UTF-8">
    <style type="text/css">
      table { width: 200px;
        text-align: center;
        border : 1px solid gray;
        border-collapse: collapse;}
      td { border : 1px solid gray;}
      caption { caption-side: bottom;}
    </style>
  </head>
  <body>
    <p></p>
    <table>
      <caption>Leyenda de la tabla</caption>
      <tr>
        <td>1</td><td>2</td><td>3</td>
      </tr>
      <tr>
        <td>4</td><td>5</td><td>6</td>
      </tr>
      <tr>
        <td>7</td><td>8</td><td>9</td>
      </tr>
    </table>
```

Color de fondo:

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>Html5</title>
    <meta charset="UTF-8">
    <style type="text/css">
      body { background-color: rgb(215,230,245);}
    </style>
  </head>
  <body>
  </body>
</html>
```

Inserción de una imagen de fondo:

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>Html5</title>
    <meta charset="UTF-8">
    <style type="text/css">
      body { background-image: url(burbujas.jpg);}
    </style>
  </head>
  <body>
  </body>
</html>
```

Inserción de una imagen de fondo con repetición:

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>Html5</title>
    <meta charset="UTF-8">
    <style type="text/css">
      body { background-image: url(gris.jpg);
            background-repeat: repeat-x;}
    </style>
  </head>
  <body>
  </body>
</html>
```


- El aspecto del documento se puede personalizar más aún especificando los siguientes parámetros:
 - *margin*: separación opcional existente entre la caja y el resto de cajas adyacentes o de los límites del documento.
 - *border*: línea que encierra completamente el contenido y su relleno.
 - *padding*: espacio libre existente entre el contenido y el borde.

Propiedades de caja

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>Html5</title>
    <meta charset="UTF-8">
    <style type="text/css">
      p { width: 200px;
          border: 10px solid rgb(125,165,205);
          padding: 20px;
          margin: 20px;
          background-color: rgb(215,230,245);
        }
    </style>
  </head>
  <body>
    <p>Noción de caja en CSS</p>
  </body>
</html>
```

- Las especificaciones de estilo tienen dos limitaciones:
 - 1 Solamente se pueden aplicar a etiquetas HTML5. ¿Cómo aplicar un estilo particular a la primera letra de un párrafo?.
 - 2 Solamente se pueden aplicar a atributos que existen. ¿Cómo asignar estilo a un enlace visitado?.
- Las *pseudoclasses* son un intento de superar este problema.
- Sintaxis: `selector:Pseudoclase { declaraciones de estilo }`.
- También se pueden utilizar con clases. Sintaxis: `selector.clase:Pseudoclase { declaraciones de estilo }`.
- Las pseudoclasses, se pueden definir en los mismos sitios que las clases: en línea, internas o externas a una página HTML5.

Pseudoclasas de enlace

a:link permite definir la apariencia de un enlace no seleccionado:

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>Html5</title>
    <meta charset="UTF-8">
    <style type="text/css">
      a:link { text-decoration: none;
              color: black;}
    </style>
  </head>
  <body>
    <p><a href="test.htm">¡ Esto es, no obstante, un enlace !</a></p>
  </body>
</html>
```

Pseudoclasas de enlace

a:visited permite definir la apariencia de un enlace visitado:

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>Html5</title>
    <meta charset="UTF-8">
    <style type="text/css">
      a:visited {text-decoration: line-through;
        color: black;}
    </style>
  </head>
  <body>
    <p><a href="test.htm">Esto es un enlace visitado.</a></p>
  </body>
</html>
```

Pseudoclasas de enlace

a:hover permite definir la apariencia cuando el ratón le sobrevuela:

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>Html5</title>
    <meta charset="UTF-8">
    <style type="text/css">
      a:hover { font-weight: bold;
                font-style: italic;
                font-size: 1.5em;}
    </style>
  </head>
  <body>
    <p><a href="test.htm">Esto es un enlace.</a></p>
  </body>
</html>
```

Pseudoclasas de enlace

`a:active` permite definir la apariencia cuando se hace click en él:

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>Html5</title>
    <meta charset="UTF-8">
    <style type="text/css">
      a:active {text-decoration: underline overline;}
    </style>
  </head>
  <body>
    <p><a href="test.htm">Esto es un enlace.</a></p>
  </body>
</html>
```

Pseudoclasas de enlace

- Se suelen utilizar de forma conjunta.
- `:hover` se puede utilizar con otros elementos además de con los enlaces. Ejemplo:

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>Html5</title>
    <style type="text/css">
      div { width: 200px; height: 50px;
            border: 1px solid black;}
      div:hover { background-color: rgb(125,165,205)}
    </style>
  </head>
  <body>
    <div></div>
  </body>
</html>
```


Pseudoclasas de párrafo

:first-letter permite definir la apariencia de la primera letra de un párrafo (estilo para la letra capital):

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>Html5</title>
    <meta charset="UTF-8">
    <style type="text/css">
      #box { border: 1px solid black;
            margin-left: 30px;
            width: 200px;
            padding: 10px;}
      .letracapital:first-letter { margin-right: 5px;
                                   font-size: 4em;}
    </style>
  </head>
  <body>
    <div id="box">
      <div class="letracapital">He aquí un texto que ilustra el efecto de letra capital
        que podemos obtener con: first-letter.</div>
    </div>
  </div>
</body>
</html>
```

Pseudoclasas de párrafo

:first-line permite definir la apariencia de la primera línea de un párrafo:

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>Html5</title>
    <meta charset="UTF-8">
    <style type="text/css">
      .box { border: 1px solid black;
            margin-left: 30px;
            width: 200px;
            padding: 10px; }
      .linea1:first-line { font-variant: small-caps;
                          font-weight: bold; }
    </style>
  </head>
  <body>
    <div class="box">
      <p class="linea1">He aquí un texto que ilustra el efecto de estilo que podemos
        obtener con: first-line.</p>
    </div>
  </body>
</html>
```

Pseudoclasas para insertar contenido

:before permite, junto con la propiedad content, junto con la propiedad content insertar un contenido antes de un elemento:

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>Html5</title>
    <meta charset="UTF-8">
    <style type="text/css">
      h1:before {content:url(arrow.gif);}
    </style>
  </head>
  <body>
    <h1>Capitulo 1</h1>
    <p>Lorem ipsum dolor sit amet...</p>
    <h1>Capitulo 2</h1>
    <p>Lorem ipsum dolor sit amet...</p>
  </body>
</html>
```

Pseudoclases para insertar contenido

:before permite, junto con la propiedad content, insertar un contenido después de un elemento:

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>Html5</title>
    <meta charset="UTF-8">
    <style type="text/css">
      h2:after {content:url(new.gif);}
    </style>
  </head>
  <body>
    <h2>Capitulo 1</h2>
    <p>Lorem ipsum dolor sit amet...</p>
    <h2>Capitulo 2</h2>
    <p>Lorem ipsum dolor sit amet...</p>
  </body>
</html>
```

- CSS3 proporciona además una gran versatilidad para referirse a elementos concretos de la página web.

Selectores jerárquicos:

- `A B`: selecciona cualquier elemento B que es descendiente de A.
- `A > B`: selecciona cualquier elemento B que es hijo directo de A.
- `A + B`: selecciona cualquier elemento B precedido directamente de A.

Selectores de CSS3

#division h1:

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>Html5</title>
    <meta charset="UTF-8">
    <style type="text/css">
      #division h1 { width: 200px;
                      background-color: #9cf;}

      h1 { margin-top: 5px;
          margin-bottom: 5px;
          font-size: 20px;}

      p { margin-top: 5px;
          margin-bottom: 5px;}
    </style>
  </head>
  <body>
    <div id="division">
      <h1>Titulo 1</h1>
      <div id="contenido" style="padding-left: 20px;">
        <h1>Titulo 1 bis</h1>
        <p>Párrafo 1</p>
      </div>
      <h1>Titulo 2</h1>
    </div>
  </body>
</html>
```

Selectores de CSS3

#division > h1:

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>Html5</title>
    <meta charset="UTF-8">
    <style type="text/css">
      #division > h1 { width: 200px;
                        background-color: #9cf;}
      h1 { margin-top: 5px;
          margin-bottom: 5px;
          font-size: 20px;}
      p { margin-top: 5px;
          margin-bottom: 5px;}
    </style>
  </head>
  <body>
    <div id="division">
      <h1>Titulo 1</h1>
      <div id="contenido" style="padding-left: 20px;">
        <h1>Titulo 1 bis</h1>
        <p>Párrafo 1</p>
      </div>
      <h1>Titulo 2</h1>
    </div>
  </body>
</html>
```

Selectores de CSS3

li.uno + li:

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>Html5</title>
    <meta charset="UTF-8">
    <style type="text/css">
      li.uno + li { width: 200px;
                    background-color: #9cf;}
    </style>
  </head>
  <body>
    <ul>
      <li class="uno">Item 1</li>
      <li>Item 2</li>
      <li>Item 3</li>
      <li>Item 4</li>
      <li>Item 5</li>
    </ul>
  </body>
</html>
```


Pseudoclases de selección

- `:empty`: elementos vacío y que no tienen hijos.
- `:only-child`: devuelve el hijo único (si lo hubiera).
- `:first-child`: devuelve el primer hijo.
- `:last-child`: devuelve el último hijo.
- `:nth-child(n)`: devuelve el enésimo hijo (la numeración comienza en 1).
- `:nth-last-child(n)`: devuelve el enésimo hijo contando a partir del último.
- `:only-of-type`: devuelve el único elemento de ese tipo.
- `:first-of-type`: devuelve el primer elemento de ese tipo.
- `:last-of-type`: devuelve el último elemento de ese tipo.

Pseudoclasas de selección

Partiendo de la página web siguiente:

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>Html5</title>
    <meta charset="UTF-8">
  </head>
  <body>
    <ul>
      <li>Item 1</li>
      <li>Item 2</li>
      <li>Item 3</li>
      <li>Item 4</li>
      <li>Item 5</li>
    </ul>
  </body>
</html>
```

Pseudoclasses de selección

Consideramos los siguientes ejemplos:

- `ul:first-child {background-color: #9cf; } o
li:first-of-type {background-color: #9cf; }`
- `ul:last-child {background-color: #9cf; } o
li:last-of-type {background-color: #9cf; }`
- `ul:nth-child(3) {background-color: #9cf; } o
li:nth-of-type(3) {background-color: #9cf; }`
- `ul:nth-last-child(2) {background-color: #9cf; } o
li:last-nth-of-type(2) {background-color: #9cf; }`

Selectores de atributos

Consideremos en siguiente ejemplo:

```
<!DOCTYPE html>
<html lang="es">
<head>
  <title>Html5</title>
  <meta charset="UTF-8">
</head>
<body>
  <div class="titulo"> Selectores de atributo </div>
  <div name="uno"> Bla bla bla </div>
  <div title="dos"> Bla bla bla 2 </div>
  <div title="tres"> tres tres tres </div>
  <div name="ultimo"> fin del fin </div>
</body>
</html>
```

Selectores de atributos

- `[atr]`: selecciona un elemento que contiene el atributo `atr`.
Ejemplo: `div[name] {background-color: #9cf; }`
- `[atr='valor']`: selecciona un elemento que contiene el atributo `atr` con el valor indicado. Ejemplo: `div[class='titulo'] {background-color: #9cf; }`
- `[atr^='valor']`: selecciona un elemento que contiene el atributo `atr` que comienza con el prefijo indicado. Ejemplo:
`div[name^='ult'] {background-color: #9cf; }`
- `[atr$='valor']`: selecciona un elemento que contiene el atributo `atr` que termine con el sufijo indicado. Ejemplo:
`div[title$='os'] {background-color: #9cf; }`
- `[atr*='valor']`: selecciona un elemento que contiene el atributo `atr` que contenga el valor indicado. Ejemplo: `div[title*='re'] {background-color: #9cf; }`