



ELEARNING TOTAL

Desarrollo Web HTML5 y CSS3 – Unidad 6

Diseño Web Responsive HTML5 y CSS3

Unidad 6: Multimedia y Formularios



Indice

Unidad 6: Multimedia y Formularios

Elementos eliminados en HTML5

Elementos reinterpretados

Elementos interactivos HTML5



Objetivos

Que el alumno logre:

- Conocer los nuevos elementos semánticos incorporados en la versión HTML5.





Elementos eliminados en HTML5

Algunos de los objetivos de HTML5 son evitar el código innecesario y facilitar la tarea a los desarrolladores de navegadores, a los fabricantes de editores de html y fundamentalmente a los desarrolladores de los sitios web.

Para lograr este fin, se eliminaron algunos elementos que se consideraban innecesarios.

Una recomendación importante para tener en cuenta es que todo lo que pueda ser resuelto empleando estilos, debe ser realizado mediante CSS. De esta forma, logramos la finalidad de abstraer la estructura del documento y los datos de la representación que nos proveen los estilos.

Etiquetas de representación:

- `<basefont>`
- `<big>`
- `<center>`
- ``
- `<s>`
- `<strike>`
- `<tt>`
- `<u>`

También se eliminan las etiquetas de diagramación mediante marcos, ya que afectan la usabilidad, la accesibilidad y la navegabilidad del sitio:

- `<frameset>`
- `<frame>`
- `<noframes>`

Además se eliminan:

- `<acronym>`
 - en su lugar se utiliza `<abbr>`
- `<applet>`
 - se utiliza `<object>`



- `<dir>`
 - su función se reemplaza con la listas (ul por ejemplo)

En relación a los atributos, dejan de ser soportados:

- `<html>` deja de soportar el atributo **versión**
- `` deja de soportar los atributos **longdesc** y **name**
- `<link>` deja de soportar el atributo **target**
- `<meta>` deja de soportar el atributo **scheme**
- `<object>` deja de soportar los atributos **archive**, **classid**, **codebase**, **codetype**, **declare** y **standby**.
- `<a>` deja de soportar los atributos **rev** y **charset**.

Algunas recomendaciones generales en cuanto a etiquetas y atributos en HTML5:

- No se recomienda el uso del atributo **aling** (como atributo de html), ya que su función puede ser afectada mediante el uso de CSS.
- Para la etiqueta `body`, no es recomendable usar los atributos **alink**, **link**, **text**, **vlink**, **background** y **bgcolor**.
- Para elementos de tabla, no se recomienda emplear los atributos **bgcolor**, **border**, **cellpadding**, **valign**, **height**, **width**, **nowrap**, **frame** y **char**.
- En el caso de listas y menús, no se recomienda emplear los atributos **type** y **compact**.
- Tampoco es recomendable utilizar **hspace** y **vspace** en las etiquetas `` y `<object>`



Elementos reinterpretados

Con el fin de aportar una nueva estructura de página con mayor valor semántico, alguno de los elementos ya utilizados en HTML se resignificaron:

- `<dl>`
- `<cite>`
- `<address>`
- ``
- ``
- `<small>`
- `<i>`
- ``
- `
`
- `<hr>`



Nuevos elementos

Podemos encontrar una amplia variedad de nuevas etiquetas desarrolladas para HTML 5 (y algunas otras que aún están en desarrollo). Algunas de ellas son:

- <article>
- <aside>
- <audio>
- <canvas>
- <command>
- <details>
- <dialog>
- <embed>
- <figure>
- <footer>
- <header>
- <hgroup>
- <keygen>
- <mark>
- <meter>
- <nav>
- <output>
- <progress>
- <rp>
- <rt>
- <ruby>
- <section>
- <source>
- <time>
- <video>

HTML 5 introduce nuevos elementos "inline" muy útiles para aumentar nuestro existente arsenal de "span, strong, em, abbr, etc". A partir de ahora a estos elementos ya no se les llamará "inline", sino "text-level semantics."



Mark

Cuando revisamos un listado de una búsqueda en una web, usualmente vemos el término por el que hemos buscado iluminado o resaltado dentro de cada resultado. Por lo general se marca cada instancia del término de búsqueda con un elemento `span`, pero `span` desde un punto de vista semántico, es algo impreciso, ya que realmente sirve para poco más que aplicarle una clase específica a un elemento dentro de un estilo ya definido.

Se podría utilizar `em` o `strong` pero semánticamente no sería correcto ya que no querrías otorgarle una “importancia” al término de búsqueda, simplemente queremos que de alguna manera quede resaltado.

Para eso utilizamos el elemento `mark`:

```
<h1>Resultado de búsqueda para 'e-learning'</h1>
<ol>
<li>
<a href="http://www.sceu.frba.utn.edu.ar/">Cursos de <mark>e-leraning</mark> UTN.</a>
</li>
</ol>
```

El elemento `mark` no lleva implícito ninguna importancia para el contenido salvo el mostrarlo como algo de interés en el contexto en que esté.

Según la especificación, `mark` denota fragmento de texto de un documento marcado o iluminado con fines de referencia debido a su importancia en otro contexto.

Resultado de búsqueda para 'e-learning'

1. [Cursos de e-leraning UTN.](http://www.sceu.frba.utn.edu.ar/)



Meter

El elemento **meter** puede ser utilizado para marcar medidas, siempre que esas medidas sean parte de una escala con valores mínimos y máximos.

La función de la nueva etiqueta **meter** de la especificación HTML5 es la de **indicar una medida escalar dentro de un rango conocido**, o un valor fraccional.

Es importante aclarar que **meter** es usado para representar un rango, o sea que no sería correcto usarlo para mostrar nada más que un simple número, a no ser que ese número pueda ser acotado por un mínimo y un máximo.

```
<meter min="0" max="120" value="40">
```

```
<span>1/3</span>
```

```
</meter>
```

Aquí también tenemos la posibilidad de usar contenido alternativo para aquellos navegadores que todavía no soportan dicho elemento.



Visualización de la etiqueta meter en Google Chrome

Progress

Mientras el elemento **meter** es muy bueno para describir algo que ya ha sido medido. El elemento **progress** te permite marcar un valor que está en proceso de cambio.

Este elemento puede ser usado (en conjunto con Javascript) para mostrar el progreso de una tarea o de un proceso que esté ocurriendo en la página web, como por ejemplo un archivo o recurso que se está subiendo o descargando. Su estructura es la siguiente:

Su perfil está **<progress>60%</progress>** completado.

Otra vez se pueden utilizar los atributos **min**, **max**, y **value**:



```
<progress min="0" max="100" value="60"></progress>
```

El elemento **progress** es más útil cuando se utiliza combinado con DOM Scripting. Se puede utilizar JavaScript para actualizaciones dinámicas del valor, permitiendo al navegador comunicar el cambio al usuario. Muy útil para cargar archivos con Ajax.

Barra de progreso con etiqueta progress



Un punto a notar es que **ambos elementos (meter y progress) son modificables en tamaño (tanto ancho como alto) mediante CSS.**

Imágenes

La directiva **<figure>** se trata de una nueva incorporación de HTML 5.

Es un elemento que engloba a la imagen y a la información adicional (título, descripción, referencias) que pueda acompañarla.

Se trata de un bloque como **<section>** o **<article>** que permite darle un tratamiento especial (incluso sacándolo del contenido principal y llevarlo, por ejemplo a una barra lateral).

<figure> se utiliza en conjunto con el elemento **<figcaption>**

Hay que tener en cuenta que una figura en HTML puede contener cualquier contenido que ayude a ilustrar el contenido principal, lo que incluye fragmentos de código, animaciones, objetos multimedia, o textos adicionales. No está restringido únicamente a imágenes.

El elemento **<figcaption>** representa un título o leyenda a una figura establecida por **<figure>**.



<figure> con

```
<figure>
  <img src = "orang-utan.jpg" alt = "Orangután bebe colgando de una cuerda"
>
</ figure>
```



<figure> con y <figcaption>

El figcaption funciona como una leyenda de la imagen.



Un macaco descarada, Baja Kintaganban Río, Borneo. Original por [Richard Clark](#)



```
<figure>
  <img src = "macaque.jpg" alt = "Macaco en los árboles" >
  <figcaption> Baja Kintaganban Río, Borneo. Original por <a href =
"http://www.flickr.com/photos/rclark/" > Richard Clark </ a> </ figcaption>
</ figure>
```

<figure> con varias imágenes

```
<figure>
  <img src = "/" kookaburra.jpg" alt = "Kooaburra" >
  <img src = "/" pelican.jpg" alt = "Pelican estaba en la playa" >
  <img src = "/" lorikeet.jpg" alt = "Cheeky busca del arco iris Lorikeet" >
  <figcaption> Aves de Australia. De izquierda a derecha, Kookburra, Pelican y
Lorikeet Arco Iris. Originales de <a href =
"http://www.flickr.com/photos/rclark/" > Richard Clark </ a> </ figcaption>
</ figure>
```



Aves de Australia. De izquierda a derecha, Kookburra, Pelican y Lorikeet Arco Iris. Originales de [Richard Clark](http://www.flickr.com/photos/rclark/)



Elementos Interactivos HTML 5

Uno de los grandes cambios de HTML 5 con respecto a versiones anteriores es que ha simplificado la forma de introducir contenidos multimedia que antes requerían de extensiones específicas como Flash.

<DETAILS>

Un elemento que recoge información adicional que puede mostrarse u ocultarse. Hay una cierta divergencia en la forma de entender este elemento, por lo que debe usarse con precaución: para el W3C contendrá un atributo <LEGEND> que por algunos navegadores, contiene elementos <DT> y <DD> como si fuera una lista.

Ante un elemento <DETAILS>, el navegador incorpora un símbolo que permite ampliar/reducir detalles.

```
<details>
<summary>Copyright 1999-2011.</summary>
<p>All pages and graphics on this web site are the property of the company.</p>
</details>
```

<MENU> y <COMMAND>

<MENU> define un menú con sus opciones. Cada elemento del menú se define con <COMMAND> que contiene los atributos @OnClick, con la acción que debe desencadenarse al pulsar sobre el elemento del menú, y @label e @icon con la etiqueta y la imagen asociados a ese elemento.

<MENU> cuenta con el atributo @label para dar un nombre al menú, y @type que permite indicar el tipo de menú de comandos. Hay definidos 3 valores: “context menú” para un menú contextual; “toolbar” para una barra de herramientas; y “list” para una lista de comandos.

Dentro de <MENU> puede utilizarse para separar elementos, como si fuera una lista.

Cada elemento del menú define con directivas ya conocidas como <BUTTON> indicando el tipo al que pertenecen (con @type).

Los elementos de <COMMAND>, tan solo se muestran si están contenidos dentro de un <MENU>.



```
<menu>
<li><input type="checkbox" />Red</li>
<li><input type="checkbox" />blue</li>
</menu>
```

Atributos de la etiqueta command

type

- Indica el tipo de comando y el valor por defecto en el mando. **valores: checkbox, radio, command**

checked

- Se utiliza sólo cuando el tipo es una radio o una casilla de verificación y define si el botón de comando respectivo se comprueba o no. **valor: checked**

disabled

- Este es un atributo booleano que indica que el comando no está disponible en la forma actual. **valor: disabled**

label

- Da el nombre del comando, ya que sería muestra al usuario. Debe mencionarse explícitamente y no debe estar vacío. **valores: command, name**

icon

- Da una imagen o una imagen que representa el comando. Debe contener una dirección válida. **valor: url**

radiogroup

- Puede ser utilizado solamente si el comando es una radio. **valor: radiogroup**



Resumen

En esta Unidad...

Trabajamos con los nuevos elementos multimediales de HTML5.

En la próxima Unidad...

Comenzaremos a trabajar con Tipografías, Íconos y Animación en CSS3.