

Repaso XHTML y CSS

¿Qué es HTML?

HTML HyperText Markup Language no es un lenguaje de programación, no se programa. Es un lenguaje de marcado, para **maquetar** tu página web y presentar la información

¿Qué es XHTML?

XHTML (eXtensible HyperText Markup Language), es básicamente HTML expresado como XML válido.

Diferencias entre HTML y XHTML

La siguiente lista muestra algunas reglas de XHTML 1.0 que lo diferencian de HTML 4.01.

→ Los elementos vacíos deben cerrarse siempre:

➤ **Incorrecto:** `
`

➤ **Correcto:** `
</br>` ó `
` ó `
`

Nota: Cualquiera de las tres formas es válida en XHTML. Por compatibilidad, **debe usarse** `
`.

→ Los elementos *no vacíos* también deben cerrarse siempre:

➤ **Incorrecto:** `<p>Primer párrafo<p>Segundo párrafo`

➤ **Correcto:** `<p>Primer párrafo</p><p>Segundo párrafo</p>`

→ Los elementos anidados deben tener un correcto orden de apertura/cierre (el que se abre último, debe cerrarse primero).

➤ **Incorrecto:** `Texto`

➤ **Correcto:** `Texto`

→ Los valores de los atributos deben siempre ir encerrados entre comillas (simples o dobles).

➤ **Incorrecto:** `<td rowspan=3>`

➤ **Correcto:** `<td rowspan="3">`

→ Los nombres de elementos y atributos deben ir en minúsculas.

➤ **Incorrecto:** `Domname`

➤ **Correcto:** `Domname`

→ No está permitida la minimización de atributos (se usa el nombre del atributo como valor).

➤ **Incorrecto:** `<textarea readonly>Solo-lectura</textarea>`

➤ **Correcto:** `<textarea readonly="readonly">Solo-lectura</textarea>`

→ Los atributos desaprobados en HTML 4.01 no forman parte de XHTML.

➤ **Incorrecto:** `Blue text`

➤ **Correcto:** `Blue text`

Reglas para DTDs (document type definition) estrictos:

→ El texto no debe ser insertado directamente en el cuerpo (dentro de la etiqueta *body*).

➤ **Incorrecto:** `<body>Texto plano</body>`

➤ **Correcto:** `<body>Texto plano</body>`

→ No se deben insertar elementos de bloque dentro de elementos de línea.

➤ **Incorrecto:** `<h2>Título</h2>`

➤ **Correcto:** `<h2>Título</h2>`

CARACTERES ESPECIALES

Las páginas web pueden conllevar problemas de todo tipo con los caracteres **especiales**. Suelen escribirse en el código HTML de una forma especial: empiezan por ‘&’ seguido de un nombre establecido y acaban con un punto y coma.

á → ´	é → é	í → í	ó → ó	ú → ú	ñ → ñ
-------------	--------------	--------------	--------------	--------------	--------------

Carácter	En el código	Inglés
<	<	Less Than
>	>	Greater Than
&	&	Ampersand
“	"	Quotation Marks
€	€	Euro
·	·	Middle Dote
Espacio en blanco	 	Non Breakable Space

Una forma de regular la visualización de este tipo de caracteres sin tener que escribirlos constantemente de este modo, es **utilizar una codificación de caracteres**. Si no es la adecuada, se verán de forma incorrecta.

*Por ejemplo, no podemos representar directamente el carácter hebreo ‘**א**’ en un documento si se utiliza una codificación **ISO 8859-1** (que cubre los idiomas de Europa Occidental). Una forma de indicar que desea incluir dicho carácter es utilizar el escape **א**.*

UTF-8 es el sistema de codificación más extendido y con más soporte a lo largo y ancho del mundo, por eso será el que utilizaremos.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN" "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
    <title>Este es mi título</title>
  </head>
  <body>
    <h2>El año del éxito.</h2>
  </body>
</html>
```

DOCTYPE

Le indica al navegador qué tipo de fichero está interpretando. En nuestro caso, siempre le diremos que se trata de un fichero escrito en **xhtml**. Por tanto situaremos al inicio de nuestra página una expresión como la siguiente:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3c.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
```

ETIQUETAS XHTML

- Son un conjunto de palabras reservadas que van entre los símbolos `<` y `>`.
- Deben estar escritas **en minúsculas**.

Hay dos tipos de etiquetas. En primer lugar aquellas que tienen una etiqueta de apertura y otra cierre. Las etiquetas de cierre se diferencian de las de apertura simplemente en que llevan la barra inclinada `/` antes del nombre de la etiqueta. *Ejemplo:*

`<p>` Párrafo `</p>`

`<div>` Caja o Contenedor `</div>`

`<h1>` Cabecera `</h1>`

Entre apertura y cierre se sitúa el contenido de la etiqueta, que puede incluir **texto** u otras **etiquetas** que también se deben abrir y cerrar.

Este ejemplo de estructura sería correcto:

`<p>` Párrafo `` con `` un texto `</p>`

Y este otro sería incorrecto:

`<p>` Párrafo `` con un texto `</p>`

Las etiquetas que se abren en primer lugar, se deben cerrar después de las que se abren posteriormente.

El otro tipo de etiquetas son aquellas que **no necesitan una etiqueta de cierre**, y simplemente incluyen la barra inclinada / al final de la única etiqueta que las compone. Ejemplo:

```
<hr />
```

```
<br />
```

```

```

Las etiquetas **pueden incluir** modificadores de su significado llamados **parámetros**. Se sitúan a continuación del nombre de la etiqueta, separados por espacios y siempre tienen a continuación el símbolo igual (=) y un valor que va entre comillas dobles (" "). *Ejemplo:*

```
<p style="color:red;">Párrafo de texto en color rojo</p>
```

```
<h1 style="color:blue;">Cabecera en color azul</h1>
```

```

```

```
<div id="contenido">Caja identificada con el nombre contenido</div>
```


HEAD

Entre `<head>` y `</head>` se añaden otras etiquetas que aportan información **sobre la página** y su contenido **no** aparecerá a la ventana del navegador. Algunas de estas etiquetas más importantes son:

<code><title>Nom de la página</title></code>	Inserta el nombre de la página en la barra de título de la ventana del navegador web.
<code><link rel="stylesheet" href="estilos.css" type="text/css" /></code>	Indica al navegador en qué fichero están los estilos a aplicar a la página: estilos.css , que en este caso deberá estar a la misma carpeta que el fichero .html .
<code><meta http-equiv="content-type" content="text/html; charset=utf-8" /></code>	Indica el conjunto de caracteres que se utilizará al contenido de la página. Se pueden utilizar otros charset , como por ejemplo iso-8859-1 .
<code><script language="Javascript" src="ficheroScripts.js" type="texto/javascript"></script></code>	Indica al navegador en qué fichero está el código javascript que se puede ejecutar a la página: ficheroScripts.js , que en este caso deberá estar a la misma carpeta que el fichero .html .

BODY

Entre **<body>** y **</body>** se incluye todo lo que se desea visualizar a la ventana del navegador.

El contenido se introduce generalmente dentro de etiquetas **<div>** que introduciremos **para crear nuestra estructura** en la página. Las etiquetas crean un espacio en la página que llamaremos **cajas** (rectangulares). Algunas de las etiquetas más importantes son las siguientes:

Etiqueta	Función
<div> </div>	Crea una caja , por poder estructurar el contenido de la página.
<h1> Texto </h1>	Cabecera, título principal.
<h2> </h2>... <h6> </h6>	Cabeceras con medidas progresivamente menores .
<p> Párrafo de texto </p>	Párrafo , donde meter texto dentro de la página.
 texto 	Negrita
 texto 	Cursiva

Etiqueta	Función
<code><hr /></code>	Inserta una línea horizontal en la página.
<code></code>	Inserta la imagen foto.jpg dentro de la página. En este caso el fichero foto.jpg debe estar en la misma carpeta que el fichero .html
<code>
</code>	Inserta un salto de línea en la página.
<code><table> </table></code>	Indica la presencia de una tabla en la página.
<code><tr> </tr></code>	Inserta una fila en una mesa.
<code><td> </td></code>	Inserta una casilla dentro de la fila de una tabla.

Etiqueta	Función
<code> </code>	Lista ordenada
<code> </code>	Lista no ordenada
<code> </code>	Inserta un elemento dentro de una lista.

Etiqueta	Función
<code> Google </code>	Asigna un enlace al texto Google, que abre la página web www.google.es .
<code> Documento </code>	Asigna un enlace al texto Documento, que abre el documento que contiene el archivo document.pdf.
<code> Página web</code>	Asigna un enlace al texto Página web, que abre otro fichero .html que nosotros hemos creado.

ESTRUCTURA

Conviene estructurar una página web para organizar mejor la información que queremos que se visualice. Toda etiqueta crea una caja, pero la etiqueta `<div>`, se utiliza exclusivamente por crear cajas de contenidos. *Ejemplo:*

```
<body>
  <div id="contenedor">
    <div id="cabecera">
      Esto es cabecera.
    </div>
    <div id="principal">
      Esto es principal.
    </div>
    <div id="pie">
      Esto es pie.
    </div>
  </div>
</body>
```

ESTILOS

Se puede cambiar el aspecto de la página web, introduciendo los estilos en otro fichero con extensión **.css**. (*Recomendado*), indicando los estilos dentro de cada etiqueta o indicando en el apartado **head** de la página.

En el head:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01//EN">
<html>
  <head>
    <title>Mi primera página con estilo</title>
    <style type="text/css">
      body {
        color: purple;
        background-color: #d8da3d
      }
      h1 {
        font-family: Helvetica, Geneva, Arial
      }
    </style>
  </head>
<body>
  ...
```

Dentro de cada etiqueta:

```
<p style="font-family: tahoma, arial, sans-serif; color: #59C; ">...</p>
```

En una nueva línea en la sección de cabecera `<head>`, con la etiqueta `<link>`:

```
<link type="text/css" rel="stylesheet" href="estilos.css" />
```

Ya hemos visto donde se definirán los estilos de la página. Ahora veremos que hay **tres maneras para asociar** el estilo con las etiquetas html:

id: identificador **único** en toda la página web.

En el archivo .html	En el archivo .css
<pre><div id="cabecera"> <h1> Cabecera de la página </h1> <p> párrafo de texto de la página </p> </div></pre>	<pre>#cabecera { background-color:#0000FF; color:#FFFFFF; }</pre>

class: conjunto de características que se pueden asignar a varios elementos de una página:

En el archivo .html	En el archivo .css
<code><p class="claseParrafo"> párrafo de texto con el aspecto definido en la clase claseParrafo definido en un archivo .css </p></code>	<code>.claseParrafo{ background-color:#00FF00; color:#FFFF00; }</code>

nombre de etiqueta: Usando directamente el tipo de etiqueta que queremos cambiar en todo el documento.

En el archivo .html	En el archivo .css
<code><h1> Título </h1></code>	<code>h1 { background-color:#FF0000; color:#FF00FF; }</code>

Si se producen dos o más instancias de diferente tipo, la preferencia se determinará como sigue:

1. ID (*ejemplo*: #capa1)
2. Clase y Pseudoclase (*ejemplos*: .aviso y :hover)
3. Selector de tipo simple (*ejemplo*: p)

Si se desea que siempre se aplique un estilo, se utiliza la propiedad **important**:

```
h2 {
    color : blue !important;
}
```