Los lenguajes de marcas son llamados así porque los resultados se obtienen a partir de unas etiquetas fijas, llamadas marcas. En el lenguaje de marcas html la marca <html> indica el inicio de un documento de este tipo y la marca <body> indica el inicio de la parte visible del documento final.

Salvo que se indique lo contrario, una <marca> deberá ser cerrada de la forma </marca>

Las marcas pueden tener características propias. Por ejemplo podemos escribir:

<body bgcolor=pink>

A la marca body se le ha añadido la característica bgcolor, que cambia el color del fondo del documento.

Estas características tienen argumentos, de tipo string.

**Lista de marcas**

* **<br>:** salto de línea (branch).
* **<hn>:** Titula un párrafo con un tamaño de letra dependiente de n ( 1<= n <= 6). Mayor valor de n implica texto más pequeño.
* **<p(n)>:** Agrupa el texto en un párrafo, que puede numerarse opcionalmente mediante un número n (no afecta al tamaño de texto).**No introduce salto de línea.**
* **<!--:** Inicia un comentario
* **-->:** Termina un comentario
* **<b>:** Texto en negrita. Está en desuso, sustituyéndose por <strong>
* **<i>:** Texto en cursiva.
* **<u>:** Texto subrayado.

**Lista de características para algunas marcas**

* **bgcolor=”color”**: cambia el color del fondo del documento a “color”. Este argumentos toma valores RGB en hexadecimal (en el rango #000000 a #ffffff), aunque hay colores preestablecidos, como pink, yellow, red…
* **text=”color”:** como antes pero para texto.
* **background=”path\image”:** fija como fondo una imagen image, localizada en la ruta path.
* **align=””:** alinea el texto a “left”, “right” o “center”

**Algunas marcas especiales**

**<font>:** permite modificar aspectos de la tipografía a partir de las características

* **size=n:** cambia el tamaño de la fuente, desde 1 hasta 7
* **color=:**ya visto
* **size=n:** cambia la tipografía a n

**Elementos Especiales**

**Listas**

**<ol>:** lista ordenada, es decir los elementos siguen una sucesión ordenada

**<ul>:** lista NO ordenada, es decir los elementos son individuales

**<li>:** elemento de cualquiera de estas listas

**Imágenes**

**<imgsrc=”path”>:** imagen localizada en path. NO TIENE MARCA DE CIERRE.

* **width/heigth:**si sólo se indica una, la otra se autoajusta para mantener la escala
* **color**

Width= / height= : si solo se pone una, se reescala ella sola

Border= el borde

Alt el texto alternativo

Enlaces

<a> (con cierre)

Se puede usar con href(link) o con name para hacer etiquetas

Tablas

Son cuadrangulares que sirven para ordenaar.

Se definen a 3 niveles:

Marca general (<table>

Nivel Fila <tr>

Nivel Celda <td>

Propiedades de la tabla

Border=1

Bgcolor

Align

Width: pixeles o porcentual

Cellpadding: espacio entre pared de las celdas y su contenido

Cellspacing: entre celdas

Valign: alineación vertical: no tiene sentido para todas las marcas.

Para tabla no vale, pero para una fila por ejemplo sí (top, bottom, middle)

Día 1

Listas

Pueden ser ordenadas (por cualquier tipo) o no ordenadas (por puntito, cuadrado, de este tipo>

Litas ordenadas: <ol>

Listas desordenadas <ul>

Los ítems son <li>

Imágenes

NO TIENE CIERRE

<img>

Fundamental: src=”path”

Width= / height= : si solo se pone una, se reescala ella sola

Border= el borde

Alt el texto alternativo

Enlaces

<a> (con cierre)

Se puede usar con href(link) o con name para hacer etiquetas

Tablas

Son cuadrangulares que sirven para ordenaar.

Se definen a 3 niveles:

Marca general (<table>

Nivel Fila <tr>

Nivel Celda <td>

Propiedades de la tabla

Border=1

Bgcolor

Align

Width: pixeles o porcentual

Cellpadding: espacio entre pared de las celdas y su contenido

Cellspacing: entre celdas

Valign: alineación vertical: no tiene sentido para todas las marcas.

Para tabla no vale, pero para una fila por ejemplo sí (top, bottom, middle)

Día 2

La forma cuadrangular de la tabla nunca podrá alterarse, pero sí la distribución de las celdas.

Podemos expandir una celda . Por ejemplo, expandir una celda para que ocupe 2 filas.

<td rowspan=n> siendo n el numero de filas expandidas (mas la suya)

Analogamente existe colspan

Ejemplos de tablas para crear (ej1)

Formularios

Son programas de dialogo para introducir datos (la pagina web hace preguntas y tu las contestas a través de unos lugares donde escribir -> input

Marca de formulario <form>

Dentro iran los inputs

Marca input: <input>: la normal, sin más. Cerrada, que sublime no lo hace. NOH ACE FALTA

Propiedades:

Type=: hay varios tipos. El defecto es text (metes lo que quieras, poerl o guarda como text

Size=: el tamaño (da en caracteres, no en pixels)

Maxlength=lo máximo que puede meterse

Id=”nombre”

Name=”nombre”:

Estas dos ultimas se pueden aplicar a cualquier objeto

Las diferencias ya las iremos viendo. Ambas sirven para identificar el objeto. La id es más para datos externos (lenguajes de programcion varios) y el name para dentro (javascript).

Otro tipo de input: <select>: menú desplegable sencillo dentro ocn variaso ocioenes. Cada opción es <option>

Si que necesuta la marca de cierre

Las opciones tienen 2 partes: el valor y lo que ves. Ej: tu ves Zaragoza, pero a nivel de sotf se recibe otra cosa, por ejemplo 50 ue es es código postal.

Por defecto, coinciden.

Se puede meter selected para seleccionar el predeterminado (esta es boolena)

Otros tipos de <input>

Type=”radio”: eliges solo uno de varias opciones, descartando los demás. Ejemplo: metes edad con un botonocito.

Analogia como vector: objeto edad, con elemento 0 edad1, elemento1 edad2….

Normalmente, no va a guardas “20 años” sino la opción, como a o 3. La opción que vemos se pone fuera de la marca donde queramos: por ejemplo, delante

Problema: por defecto deja seleccionar varios y no deja deseleccionarlos: hay que darles un name común (para qie sean un mimso objeto).

Analogamente a selected, aquí se mete checked

<checkbox> Permite seleccionar varios o ninguno. Se aplican propiedades anteriores, como selected por ejemplo.

<textarea> cuadro de texto

`propiedades:

Cols, rows: lo que ocupa en estas nidades

Type submit: eu n botón especifico que hce algo con el formulario; el value es lo que aparece en el botón.Lo envía arribap ro fedcto

Hay dos formas de eniar datos:

Por get y por post

Por get manda lo de los formularios y desaparece. Lo envía a la propiedad action, que puede incorporarse al formulario (<form action=”séptimo.html”>

Es el defecto, pero está por post también (se especifica con method=post

Así va, pero no los pasar por la barra

Hay otro botón (reset que lo que hace esb orrrar lo que había en el formulario

FRAMES

En htnml5 se usan iframes. Esto esta desfasado, pero bueno, hay que entenderlo o que?

Un frame es una ventana: dentro del a ventana principal vamos a meter otras ventanas (frames).

Set: configuración

Empezamos <frameset>

2 propiedades ppales: rows y cols

Cols=”\*, 250”: hay 2 columnas: la primera ocupa lo que quiera y la segunda 250

<frame>

Propiedad fundamental: src: lo que abre

Name: lo de siempre