

UT2. HTML

UT2. HTML

2.5. Contenido del cuerpo

- 2.5.1 Manejo del texto

- 2.5.2 Listas

- 2.5.3 Enlaces

- 2.5.4 Agrupación del contenido

- 2.5.5 Imágenes

- 2.5.6 Tablas

- 2.5.7 Marcos

- 2.5.8 Objetos

2.5. Contenido del cuerpo

En HTML la mayoría de elementos del cuerpo se pueden clasificar en:

- **Elementos de bloque.**

Son elementos que crean estructuras más grandes que pueden contener a los elementos de línea y a otros elementos de bloque. Además siempre comienzan en una línea nueva.

- **Elementos de línea.**

Estos elementos solo pueden contener datos y a otros elementos de línea. No comienzan con línea nueva.

2.5.1. Manejo del texto

h1 h2 h3 h4 h5 h6

Para empezar vamos a ver los elementos h1 al h6 que sirven para establecer encabezados para las secciones.

Hay seis niveles de encabezados: h1, h2, h3, h4, h5 Y h6 ordenados estrictamente por el tamaño. Los encabezados son elementos de bloque.

Las etiquetas de apertura y cierre son obligatorias.

2.5.1. Manejo del texto

h1 h2 h3 h4 h5 h6

Ejemplo:

```
<body>  
  <h1>Encabezado H1</h1>  
  <h2>Encabezado H2</h2>  
  <h3>Encabezado H3</h3>  
  <h4>Encabezado H4</h4>  
  <h5>Encabezado H5</h5>  
  <h6>Encabezado H6</h6>  
</body>
```

2.5.1. Manejo del texto

<p>

Otro de los elementos de bloque más típicos es `<p>` de **párrafo**. De la misma forma que distribuimos el texto mediante párrafos en un documento Word, las líneas de texto en un documento HTML deben agruparse dentro de `<p>`.

Cada párrafo en HTML se compone de una única línea de texto, salvo que introduzcamos un retorno de carro explícito usando el elemento `
`. El navegador se encarga de que la línea de texto ocupe todo el ancho disponible, y si el ancho cambia, el texto se ajusta al nuevo ancho ocupando más o menos líneas.

Todo texto de un documento HTML, que no sea encabezado, lista o celda de tabla debe estar dentro del elemento `<p>`.

La etiqueta de cierre es opcional. No puede contener elementos de bloque.

2.5.1. Manejo del texto

p

Ejemplo:

```
<body>
  <h1>Evolución de Networking</h1>
  <h3>Fasel: Ordenadores personales</h3>
  <p>La aparición de ordenadores personales en las empresas
fue lenta al principio, pero el lanzamiento de programas
de gestión como Lotus y otras aplicaciones específicas de
empresa, impulsaron el desarrollo del PC.</p>
</body>
```

2.5.1. Manejo del texto

p

Un asunto a tener en cuenta es el espacio en blanco, si queremos insertar un espacio más del normal entre dos palabras, no vale utilizar la barra espaciadora porque es ignorada por el navegador.

Es necesario insertar el carácter ** ** (no-break space) por cada espacio que queramos añadir.

2.5.1. Manejo del texto

br

A veces es preciso introducir un **salto de línea** dentro del mismo párrafo, en ese caso debemos insertar la etiqueta `
` de "break" que sirve para "romper" la línea de texto.

No sirve insertar un salto de línea usando `enter` porque también es ignorado por el navegador.

Usando el ejemplo anterior, para evitar que una frase quede partida y mejorar la legibilidad del texto, podríamos introducir saltos de línea detrás de punto y seguido.

Sin etiqueta de cierre.

2.5.1. Manejo del texto

hr

Otra forma posible de separar el texto es utilizar una línea visible horizontal.

Sin etiqueta de cierre (V).

2.5.1. Manejo del texto

**** *<i>* <small> <s> <u>

Son los llamados estilos de fuente:

- b: negrita
- i: itálica o cursiva
- small: tamaño pequeño,
- s: texto tachado mediante una línea a media altura
- u: subrayado con una línea por debajo.

Las etiquetas de apertura y de cierre son obligatorias.

Se trata de elementos de línea, por tanto, no hacen salto de línea y se pueden introducir dentro de un párrafo <p>. También pueden contener a otros combinándolos entre sí.

2.5.1. Manejo del texto

**** *<i>* <small> <s> <u>

Son los llamados estilos de fuente:

- b: negrita
- i: itálica o cursiva
- small: tamaño pequeño,
- s: texto tachado mediante una línea a media altura
- u: subrayado con una línea por debajo.

Las etiquetas de apertura y de cierre son obligatorias.

Se trata de elementos de línea, por tanto, no hacen salto de línea y se pueden introducir dentro de un párrafo <p>. También pueden contener a otros combinándolos entre sí.

2.5.1. Manejo del texto

pre

Texto con formato previo. Sirve para que el navegador visualice el texto tal y como aparece en el contenido de esta etiqueta, respetando tabuladores, espacios y saltos de línea.

La única excepción es que introduzcamos en el contenido, otros **elementos HTML**, los cuales **se ejecutarán también**.

Las etiquetas de inicio y final son obligatorias.

Para representar un bloque de código se puede usar el elemento pre con el elemento code. Para representar la salida de un bloque de código se puede usar el elemento pre junto con el elemento samp. También se puede usar el elemento pre junto con el elemento kbd para indicar texto que el usuario debe introducir.

2.5.1. Manejo del texto

pre

```
<body>
  <h1>Programación en Java</h1>
  <h2>
    <pre>
class HolaMundo{

    public static void main (String arg[]){
        System.out.println("&lt;Hola Mundo&gt;")
    }
    </pre>
  </h2>
</body>
```

Para evitar que los caracteres "<" y ">" se interpreten como etiqueta debemos utilizar:

< = "<" (less than)

> = ">" (greater than)

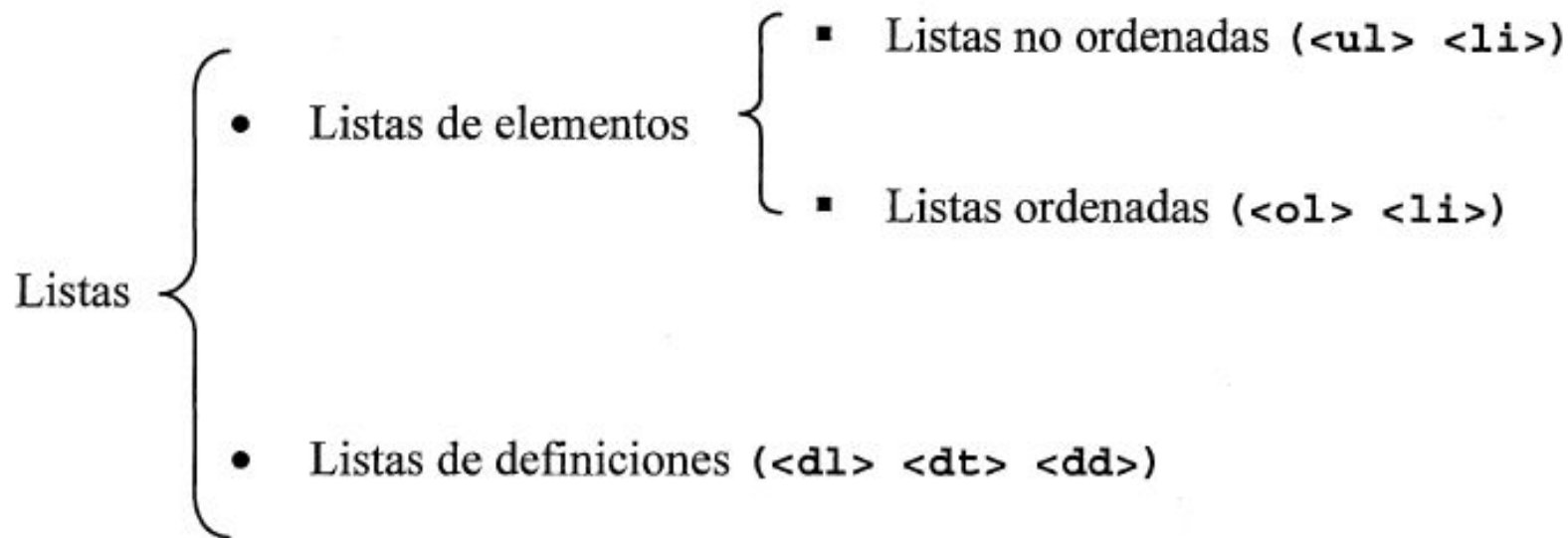
2.5.2. Listas

Tenemos 2 tipos fundamentales de listas: **listas de elementos** y **listas de definiciones**.

A su vez las listas de elementos pueden estar ordenadas o no.

Las listas son un recurso muy común en los textos, como se puede apreciar en este esquema, en el que se han utilizado listas no ordenadas.

2.5.2. Listas



2.5.2. Listas

ul

Listas de elementos **no ordenados (unordered)**.

Se construyen con el elemento **** y su contenido está formado exclusivamente por elementos ****.

Estos elementos **** son los que contienen a su vez el texto.

Las etiquetas de apertura y cierre son obligatorias.

2.5.2. Listas

ol

Listas de elementos ordenados.

Se construyen con el elemento `` y su contenido está formado exclusivamente por elementos ``.

Las etiquetas de apertura y cierre son obligatorias.

2.5.2. Listas

ol

Listas de elementos **ordenados (ordered)**.

Se construyen con el elemento **** y su contenido está formado exclusivamente por elementos ****.

Las etiquetas de apertura y cierre son obligatorias.

2.5.2. Listas

ol

Listas de elementos **ordenados (ordered)**.

Se construyen con el elemento **** y su contenido está formado exclusivamente por elementos ****.

Las etiquetas de apertura y cierre son obligatorias.

2.5.2. Listas

li

Elemento de lista.

Objeto de lista, que puede contener otros elementos de bloque como párrafos o directamente el texto.

El cierre es opcional.

2.5.2. Listas

Ejemplo:

```
<body>
  <h1>Lenguajes de Marcas</h1> <hr>
  <h2>Lista ordenada</h2>
  <ol>
    <li>HTML</li>
    <li>XHTML</li>
    <li>XML</li>
  </ol>
  <h2>Lista no ordenada</h2>
  <ul>
    <li>HTML</li>
    <li>XHTML</li>
    <li>XML</li>
  </ul>
</body>
```

2.5.2. Listas

Listas anidadas.

La mayor utilidad de las listas resulta cuando se combinan, es decir, cuando cada elemento de una lista es a su vez otra lista:

```
<body>
<h1>Listas Anidadas</h1>
<hr>
<h2>Lenguajes de Marcas</h2>
<ol>
  <li><b>Páginas Web</b>
    <ul>
      <li>HTML</li>
      <li>XHTML</li>
    </ul>
  </li>
  <li><b>Documentos</b>
    <ul>
      <li>RTF</li>
      <li>TeX</li>
      <li>MathML</li>
    </ul>
  </li>
  <li><b>Gráficos</b> <ul>
    <li>SVG</li>
    <li>STEP</li>
  </ul>
</li>
</ol>
</body>
```

Listas Anidadas

Lenguajes de Marcas

1. Páginas Web

- HTML
- XHTML

2. Documentos

- RTF
- TeX
- MathML

3. Gráficos

- SVG
- STEP

2.5.2. Listas

Listas de definiciones.

dl

Se trata de listas que no tienen índices ni símbolos de ninguna clase como las anteriores, sino que se componen de dos elementos: el **término** y la **definición**, a modo de diccionario. Todo el texto aparece con sangría.

Se construyen con el elemento `<dl>` y contienen solamente elementos de tipo `<dt>` y `<dd>` en número variable.

Las etiquetas de apertura y cierre obligatorias.

2.5.2. Listas

Listas de definiciones.

dt

Es el término que se quiere definir, normalmente una palabra o un texto breve y se trata de un elemento de línea.

La etiqueta de cierre opcional.

dd

Es la definición del término anterior, normalmente un texto largo y se trata de un elemento de bloque.

La etiqueta de cierre opcional.

2.5.2. Listas

Listas de definiciones.

Ejemplo:

```
<body>
  <h1>Lenguajes de Marcas</h1>
  <dl>
    <dt>HTML</dt>
    <dd>HyperText Markup Language </dd>
    <dd>Lenguaje de Marcas de Hipertexto</dd>
    <dt>WML</dt> <dd>Wireless Markup Language </dd>
    <dt>XHTML MP</dt>
    <dd>Extensible HyperText Markup Language Mobile Policy</dd>
    <dt>RSS</dt>
    <dd>Rich Site Summary</dd>
    <dd>Really Simple Syndication</dd>
    <dt>RTF</dt>
    <dd>Rich Text Format</dd>
  </dl>
</body>
```

Lenguajes de Marcas

HTML

HyperText Markup Language

Lenguaje de Marcas de Hipertexto

WML

Wireless Markup Language

XHTML MP

Extensible HyperText Markup Language Mobile Policy

RSS

Rich Site Summary

Really Simple Syndication

RTF

Rich Text Format

2.5.3 Enlaces

Los enlaces o hipervínculos son los elementos más característicos del HTML, ya que fueron el motivo que inspiró su desarrollo, para vincular información procedente de muchas fuentes distintas.

Un hipervínculo es en realidad una zona del documento, que al realizar alguna acción sobre ella (como hacer clic o colocar el ratón encima) nos redirecciona a otro punto del mismo documento o de otro distinto.

El hipervínculo en principio es invisible, lo que vemos son los elementos que insertamos en su contenido, como texto, imagen, objetos, etc.

2.5.3 Enlaces

a

Este elemento sirve para establecer el origen o el destino del hipervínculo.

Las etiquetas de apertura y cierre son obligatorias.

Atributos:

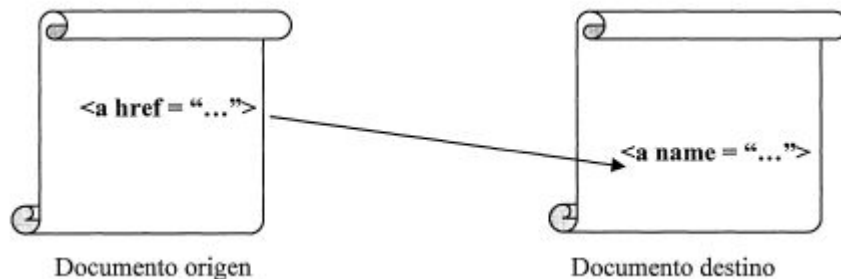
- **name:** sirve para nombrar un ancla o marcador que será el destino del enlace.
- **href:** sirve para especificar la URI destino del hipervínculo.
- **target:** establece el marco de destino (por defecto se toma la ventana actual). Valores posibles: `_blank`, `_self`, `_parent`, `_top`

2.5.3 Enlaces

a

Se trata de un **elemento en línea** por lo que puede estar dentro de un párrafo, lista, etc., a su vez `<a>` puede contener otros elementos en línea excepto a sí mismo.

Los atributos **name** y **href** son opuestos en el sentido de que, o bien creamos el origen con `<a href>`, o bien el destino con `<a name>`.



2.5.3 Enlaces

a

Destino en la misma página.

- Insertamos el marcador en el destino: ``
- Insertar el hipervínculo en el origen: ``
- Es preciso usar el carácter almohadilla # para indicar, que el destino es un marcador de posición interno al documento.

2.5.3 Enlaces

a

Destino en la misma página.

- Insertamos el marcador en el destino: ``
- Insertar el hipervínculo en el origen: ``
- Es preciso usar el carácter almohadilla # para indicar, que el destino es un marcador de posición interno al documento.
- También es posible utilizar como marcador de posición, un elemento ya existente en la página, en lugar de crear un nuevo marcador. En este ejemplo usamos el atributo **id**. El resultado sería el mismo que con el marcador `<a name>`.

Por ejemplo: `<h2 id="X"> XHTML </h2>`

2.5.3 Enlaces

a

Destino en la misma página.

- Insertamos el marcador en el destino: ``
- Insertar el hipervínculo en el origen: ``
- Es preciso usar el carácter almohadilla # para indicar, que el destino es un marcador de posición interno al documento.
- También es posible utilizar como marcador de posición, un elemento ya existente en la página, en lugar de crear un nuevo marcador. En este ejemplo usamos el atributo **id**. El resultado sería el mismo que con el marcador `<a name>`.

Por ejemplo: `<h2 id="X"> XHTML </h2>`

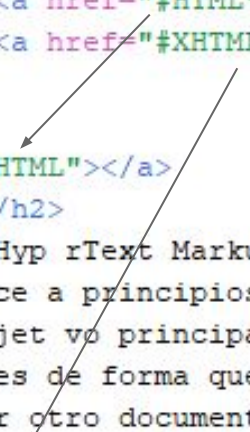
2.5.3 Enlaces

a

Destino en
la misma página.

```
<body>
  <h1>Lenguajes de marcas </h1>
  <ul>
    <li> <a href="#HTML"> HTML </a></li>
    <li> <a href="#XHTML"> XHTML </a> </li>
  </ul>

  <a name="HTML"></a>
  <h2>HTML</h2>
  <p>HTML (HyperText Markup Language)
    Aparece a principios de los años 90.
    El objetivo principal era enlazar los documentos mediante
    enlaces de forma que al hacer clic sobre algún texto, se
    abriera otro documento relacionado con la información </p>
  <a name="XHTML"></a>
  <h2>XHTML</h2>
  <p>XHTML (Extensible HyperText Markup Language)
    Es un dialecto de XML que contiene todos los elementos de
    HTML pero que se ajusta a las normas sintácticas de XML.
  </p>
</body>
```

The diagram consists of two arrows. The first arrow starts at the href attribute of the first list item, which is "#HTML", and points down to the name attribute of the first anchor, which is "HTML". The second arrow starts at the href attribute of the second list item, which is "#XHTML", and points down to the name attribute of the second anchor, which is "XHTML".

2.5.3 Enlaces

a

Destino en otra página del mismo servidor.

Supongamos ahora que tenemos una página `indice.html` que es el origen de todos los hipervínculos. También tenemos una segunda página llamada `definiciones.html` que contiene los marcadores de posición, que son el destino de los hipervínculos.

El marcador estará en la página destino: `</ a>`

El hipervínculo estará en el origen: ``

Simplemente hay que añadir el nombre de la página destino en el atributo `href`.

2.5.3 Enlaces

a

Destino en otra página del mismo servidor.

```
<ul>
  <li> <a href="Definiciones.html#H"> HTML </a> </li>
  <li> <a href="Definiciones.html#X"> XHTML </a> </li>
  <li> <a href="Definiciones.html#W"> WML </a> </li>
</ul>
```

Indice.html

```
<a name="X"></a>
<h2>XHTML</h2>
<p>XHTML (Extensible HyperText Markup Language)
  Es un lenguaje que contiene todos los elementos de HTML
  pero que se ajusta a las normas sintácticas de XML.
</p>
```

Definiciones.html

2.5.3 Enlaces

a

Destino en otro servidor (hipervínculo externo)

En este caso no necesitamos marcadores sino especificar la URI completa de la página destino.

Ejemplo:

```
<ul>
  <li> <a href="http://www.w3.org/TR/html401/"> HTML </a> </li>
  <li> <a href="http://www.w3.org/TR/xhtml1/"> XHTML </a> </li>
  <li> <a href="http://www.wapforum.org"> WML </a> </li>
</ul>
```

2.5.4 Agrupación del contenido

div

Sirve para agrupar varios elementos dentro de un bloque y de esta forma asignarle nombre, clase y estilo.

Las etiquetas de apertura y cierre obligatorias.

Se trata de un elemento de bloque, por tanto, puede contener elementos en línea y otros elementos de bloque, cómo otro elemento <div>.

2.5.4 Agrupación del contenido

div

Ejemplo:

```
<!DOCTYPE html>
<html>
  <head>
    <title> Industrias </title>
    <meta charset="UTF-8">
  </head>
  <body>
    <!-- Primer bloque llamado Header -->
    <div id="Header" style="background-color: #D5EDB3">
      <h2>LATON MAN</h2>
      <p>Industrias StarFuerte</p>
    </div>
    <!-- Segundo bloque llamado Content -->
    <div id="Content" style="background-color: #EE3">
      <p>Industrias StarFuerte presenta su nuevo prototipo de
        armadura para combate cuerpo a cuerpo contra múltiples
        enemigos. <br>
        Características:
      </p>
      <ul>
        <li>Resistencia al fuego</li>
        <li>Capacidad para volar</li>
        <li>Gran arsenal de proyectiles</li>
      </ul>
    </div>
    <p>Solo se puede adquirir por teléfono. <br>
      De regalo un set de cuchillos. <br>
      Aproveche esta magnífica oferta.
    </p>
  </body>
</html>
```

LATON MAN

Industrias StarFuerte

Industrias StarFuerte presenta su nuevo prototipo de armadura para combate cuerpo a cuerpo contra múltiples enemigos.

Características:

- Resistencia al fuego
- Capacidad para volar
- Gran arsenal de proyectiles

Solo se puede adquirir por teléfono.

De regalo un set de cuchillos.

Aproveche esta magnífica oferta.

2.5.4 Agrupación del contenido

span

Sirve para identificar un elemento en línea y de esta forma asignarle nombre, clase y estilo.

Las etiquetas de apertura y cierre son obligatorias.

Con `` podemos cambiar entre otros, el tamaño de la fuente, el color, el alto de línea, la posición del texto dentro de la línea, etc.

Mediante la etiqueta `` podemos distinguir partes concretas del texto y aplicarle un estilo diferente a cada una, sin necesidad de cambiar de línea.

2.5.4 Agrupación del contenido

span

Ejemplo:

```
<!DOCTYPE html>
<html>
  <head>
    <title> Industrias </title>
    <meta charset="UTF-8">
  </head>
  <body>
    <!-- Primer bloque llamado Header -->
    <div id="Header" style="background-color: #D5EDB3"
      align="center">
      <h2>Primer Bloque</h2>
      <p>Este párrafo no tiene estilo específico</p>
    </div>
    <!-- Segundo bloque llamado Content -->
    <div id="Content" style="background-color: #EE3"
      align="center">
      <h2>Segundo Bloque</h2>
      <p>Este párrafo tiene elementos en línea con estilo propio:
        <br>
        <span style="font-size: 16pt"> Tamaño de fuente 16
        </span>
        <span style="font-style: italic">Cursiva </span>
        <span style="font-weight: bolder"> Negrita I </span>
      </p>
    </div>
  </body>
</html>
```

Primer Bloque

Este párrafo no tiene estilo específico

Segundo Bloque

Este párrafo tiene elementos en línea con estilo propio:
Tamaño de fuente 16 *Negrita I*

2.5.5 Imágenes

img

Sirve para insertar la imagen determinada por el atributo `src`.

Se trata de un elemento en línea. Sin etiqueta de cierre.

Atributos: **src**, **alt**, **height**, **width**, **loading**, **usemap**

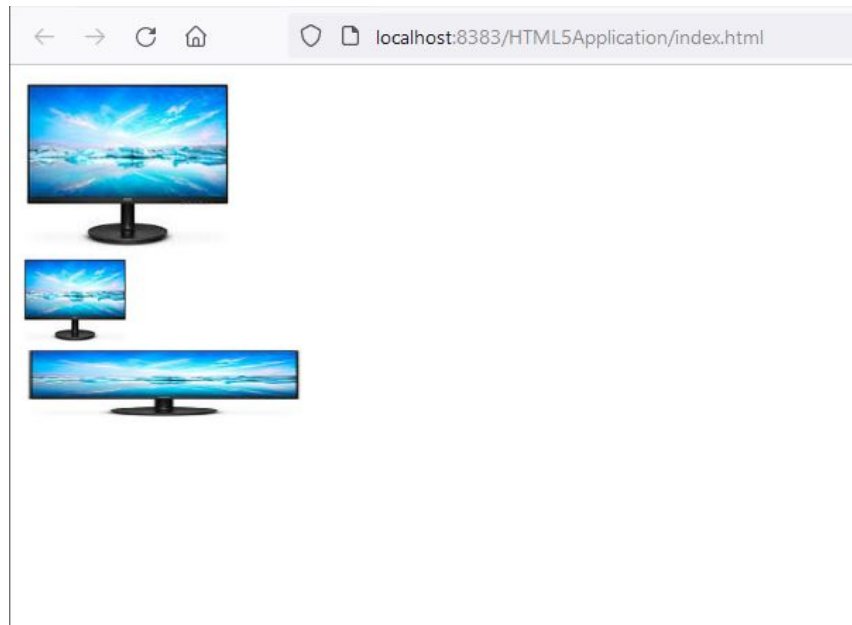
- **src**: Indica la URL de acceso a la imagen.
- **alt**: Texto con breve descripción de la imagen, que se presenta en caso de no poder visualizar la imagen.
- **height**: Altura de la imagen.
- **width**: Anchura de la imagen.
- **loading**: Si se va a postergar o no la carga de la imagen (eager, lazy).
- **usemap**: Si va a usar un mapa de areas.

2.5.5 Imágenes

img

Ejemplo:

```
<!DOCTYPE html>
<html>
  <head>
    <title> Industrias </title>
    <meta charset="UTF-8">
  </head>
  <body>
    
    <br>
    
    <br>
    
  </body>
</html>
```



2.5.5 Imágenes

map

Sirve para crear un mapa de imágenes en el lado del cliente. Las etiquetas inicial y final son obligatorias.

Ejemplo: [map](#)

2.5.5 Imágenes

area

Sirve para especificar una región geométrica del mapa y el vínculo asociado a esa región. Sin etiqueta final.

Atributos: **shape**, **coords**, **href**, **alt**.

- **shape**: especifica el tipo de región con los valores: default, rect, circle y poly.
 - **default** (región completa)
 - **rect** (rectangular) = (x,y) vértice superior izquierdo, (x,y) vértice inferior derecho.
 - **circle** (circular) = (x, y) centro, radio.
 - **poly** (poligonal) = (x,y) 1º punto, (x,y) 2º punto, ..., (x,y) último punto.
- **coords**: indica la posición en pantalla, son valores de longitud partiendo de la esquina superior izquierda de la pantalla y separados por comas.
- **href**: URL del enlace.
- **alt**: Texto alternativo para el area.

2.5.6 Tablas

table

Inserta una tabla de tamaño variable que puede contener filas, columnas, fila/s de encabezado, fila/s de cuerpo y resumen.

A su vez cada fila y columna puede estar formada por cero o más celdas. Las etiquetas inicial y final son obligatorias.

Contenido de table: **<tr>**, **<td>**, **<th>**, **<thead>**, **<tfoot>**, **<tbody>**, **<colgroup>**, **<col>** y **<caption>**.

2.5.6 Tablas

caption

Representa el título de la tabla.

Ejemplo:

```
<caption>
  <p>Table 1. </p>
  <p>This table shows the total score obtained from rolling two
    six-sided dice. The first row represents the value of the first die,
    the first column the value of the second die. The total is given in
    the cell that corresponds to the values of the two dice. </p>
</caption>
```

Table 1.

This table shows the total score obtained from rolling two six-sided dice. The first row represents the value of the first die, the first column the value of the second die. The total is given in the cell that corresponds to the values of the two dice.

	1	2	3	4	5	6
1	2	3	4	5	6	7
2	3	4	5	6	7	8
3	4	5	6	7	8	9
4	5	6	7	8	9	10
5	6	7	8	9	10	11
6	7	8	9	10	11	12

2.5.6 Tablas

tr

Sirve para crear una fila dentro de una tabla. La etiqueta final es opcional.

td, th

Sirve para crear una celda dentro de una fila (<th> para la celda de una cabecera). La etiqueta final es opcional.

Atributos: **colspan**, **rowspan**

- **colspan**: Expande esta celda para que ocupe un número de columnas.
- **rowspan**: Expande esta celda para que ocupe un número de filas.

thead, tbody, tfoot

Sirven para agrupar filas en la cabecera, el pie o en el cuerpo de una tabla.

2.5.6 Tablas

colgroup

Sirve para crear un grupo de columnas. La etiqueta final es opcional.

col

Sirve para dar estilo a parte de la tabla.

Atributos: **span**

- **span**: determina el número de columnas del grupo.

Ejemplo:

```
<table>
  <colgroup>
    <col span="2" style="background-color: #D6EEEE">
  </colgroup>
  <tr>
    <th>MON</th>
    <th>TUE</th>
    <th>WED</th>
    <th>THU</th>
    <th>FRI</th>
    <th>SAT</th>
    <th>SUN</th>
  </tr>
  ...
```

MON	TUE	WED	THU	FRI	SAT	SUN
1	2	3	4	5	6	7

2.5.7 Marcos

Antiguamente casi todas las páginas contenían marcos, porque son una manera fácil de crear diferentes áreas de navegación.

La separación entre cabecera, pie, menú e información se hacía mediante marcos. Sin embargo, la aparición de las hojas de estilo CSS los ha relegado a segundo plano.

Con las hojas de estilo podemos conseguir diseños más flexibles y variados, además tienen ciertos inconvenientes que los han relegado casi por completo.

Un documento con marcos no tiene sección **body**, en su lugar tiene una sección **<frameset>**, y una sección **<noframes>** en caso de no poder visualizar los marcos.

No soportados en la última versión de HTML.

En su lugar usa **<iframe>**.

2.5.7 Marcos

iframe

Un marco en línea (inline frame) se utiliza para incrustar otro documento dentro del documento HTML actual. Usa CSS para diseñar el <iframe>.

Es una buena práctica incluir siempre un atributo de título para el <iframe>. Los lectores de pantalla lo utilizan para leer cuál es el contenido del <iframe>.

Atributos: **allowfullscreen**, **height**, **width**, **sandbox**, **src**

- **allowfullscreen**: Si el iframe puede activar la pantalla completa (true o false).
- **height**: Altura del marco en línea.
- **width**: Anchura del marco en línea.
- **sandbox**: Establece restricciones extra para el iframe.
 - allow-downloads
 - allow-forms
 - allow-modals
 - allow-orientation-lock
 - allow-pointer-lock
 - allow-popups
 - allow-popups-to-escape-sandbox
 - allow-presentation
 - allow-same-origin
 - allow-scripts
 - allow-top-navigation
 - allow-top-navigation-by-user-activation
 - allow-top-navigation-to-custom-protocols
- **src**: URL del documento a insertar.

2.5.7 Marcos

iframe

Ejemplo:

```
<!DOCTYPE html>
<html>
  <head>
    <title> Marca </title>
    <meta charset="UTF-8">
  </head>
  <body>
    <h1> Página de Marca </h1>
    <iframe src="http://marca.com" width="800" height="600" style="border:1px solid black;">
    </iframe>
  </body>
</html>
```



2.5.8 Objetos

object

Inserta un objeto en el documento.

El tipo de objeto viene determinado por sus atributos y puede ser una imagen, un applet de Java, un vídeo, una animación flash, otro documento HTML, etc.

Las etiquetas de apertura y cierre son obligatorias.

Para imágenes es mejor usar el elemento ****, para HTML es mejor usar el elemento **<iframe>**, para audio es mejor usar el elemento **<audio>**, para video es mejor usar el elemento **<video>**.

Atributos: **data**, **type**, **width**, **height**

- **data:** Establece la URL del recurso usado por el objeto.
- **type:** Establece el tipo de media del objeto ([IANA media types](#))

2.5.8 Objetos

object

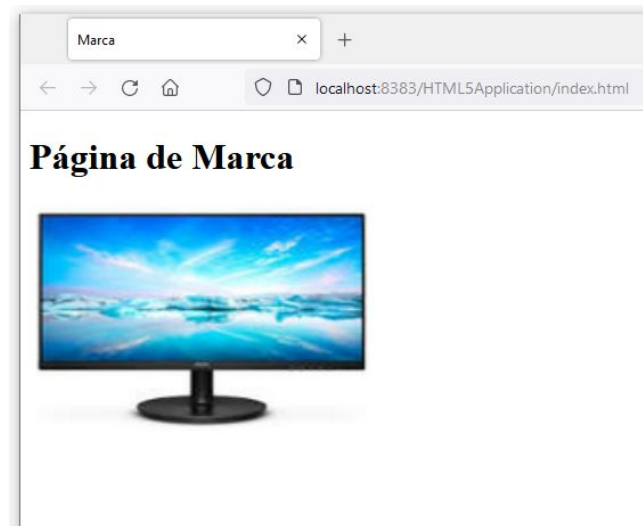
Ejemplo:

```
<!DOCTYPE html>
<html>
  <head>
    <title> Monitores S.A. </title>
    <meta charset="UTF-8">
  </head>
  <body>

    <h1> Monitores en stock </h1>

    <object data="monitor.png" width="300" height="200"></object>

  </body>
</html>
```



2.5.8 Objetos

param

Sirve para inicializar variables de objetos, que serán utilizadas en tiempo de ejecución. Sin etiqueta de cierre.

Atributos: **name**, **value**.

- **name**: indica el nombre de un parámetro de ejecución.
- **value**: indica el valor inicial del parámetro especificado en name.

Ejemplo: Insertar un video local.

```
<object type="video/mpeg" width="300" height="200" data="videos/carrera.mpg">  
  <param name="FileName" value="videos/carrera.mpg">  
  <param name="quality" value="high">  
</object>
```