

LMSGI – UD02

IMÁGENES Y OBJETOS

Imágenes.

Tipos:

- Imágenes de contenido → proporcionan información y complementan la información textual → Incluidas en el código HTML.
- Imágenes de adorno → se utilizan para hacer bordes redondeados, para mostrar pequeños iconos en las listas de elementos, para mostrar fondos de página, etc. → Pueden ir en hojas de estilos.

`` → Incluir una imagen en un documento.

- `src = "url"` - Indica la URL de la imagen que se muestra.
- `alt = "texto"` - Descripción corta de la imagen.
- `longdesc = "url"` - Indica una URL en la que puede encontrarse una descripción más detallada de la imagen.
- `name = "texto"` - Nombre del elemento imagen.
- `height = "unidad_de_medida"` - Indica la altura (en píxeles o en %) con la que se debe mostrar la imagen (no es obligatorio que coincida con la altura original de la imagen).
- `width = "unidad_de_medida"` - Indica la anchura (en píxeles o en %) con la que se debe mostrar la imagen (no es obligatorio que coincida con la anchura original de la imagen).
 - Si sólo se especifica uno de los dos (altura, anchura) el navegador calcula el otro valor para mantener la proporcionalidad de la imagen.

Mapas de imagen.

- Permite definir diferentes zonas "pinchables" dentro de una imagen, cada una de ellas puede apuntar a una URL diferente.
- Las zonas o regiones que se pueden definir en una imagen se crean mediante rectángulos, círculos y polígonos.
- Pasos:
 1. Se inserta la imagen original mediante la etiqueta ``.
 2. Se utiliza la etiqueta `<map>` para definir las zonas o regiones de la imagen.
 3. Cada zona se define mediante la etiqueta `<area>`.

`<map>` → Definir un Mapa de imagen.

- `name = "texto"` - Nombre que identifica de forma única al mapa definido.

Si una imagen utiliza un mapa de imagen, debe indicarlo mediante el atributo `usemap`. El valor del atributo debe ser el nombre del mapa de imagen definido en otra parte del mismo documento HTML:

```

...
<map name="continentes">
...
</map>
```

`<area>` → Definir un área de un mapa de imagen.

- `href = "url"` - URL a la que se accede al pinchar sobre el área.
- `nohref = "nohref"` - Se emplea para las áreas que no son seleccionables.
- `shape = "default | rect | circle | poly"` - Indica el tipo de área que se define (toda la imagen, rectangular, circular o poligonal).
- `coords = "lista de números"` - Se trata de una lista de números separados por comas que representan las coordenadas del área.
 - Rectangular = `x1,y1,x2,y2` (coordenadas X e Y del vértice superior izquierdo y coordenadas X e Y del vértice inferior derecho).
 - Circular = `x1,y1,R` (coordenadas X e Y del centro y radio del círculo).
 - Poligonal = `x1,y1,x2,y2,...,xn,yn` (coordenadas de los vértices del polígono. Si las últimas coordenadas no son iguales que las primeras, se cierra automáticamente el polígono uniendo ambos vértices).

Objetos.

applets de Java,
vídeos en formato *QuickTime* o *Flash*,
etc. ...

➔ *plugins* (el navegador no interpreta estos elementos directamente).

`<object>` ➔ Para incluir objetos en documentos.

- `data = "url"` - Indica la URL de los datos que utiliza el objeto.
- `classid`, `codebase`, `codetype` - Información específica que depende del tipo de objeto.
- `type = "Tipo_de_contenido"` - Indica el tipo de contenido de los datos.
- `height = "unidad_de_medida"` - Indica la altura con la que se debe mostrar el objeto.
- `width = "unidad_de_medida"` - Indica la anchura con la que se debe mostrar el objeto.

También se pueden incluir varias versiones alternativas de un mismo contenido. Así, si el navegador no es capaz de interpretar el formato por defecto, puede optar por otros formatos alternativos:

```
<object title="La Tierra vista desde el espacio"
classid="http://www.observer.mars/TheEarth.py">

  <!-- Formato alternativo en forma de vídeo -->
  <object data="PlanetaTierra.mpeg" type="application/mpeg">

    <!-- Otro formato alternativo mediante una imagen GIF -->
    <object data="PlanetaTierra.gif" type="image/gif">

      <!-- Si el navegador no soporta ningún formato, se muestra el
      texto -->
      La <strong>Tierra</strong> vista desde el espacio.

    </object>

  </object>

</object>

</object>
```

A los objetos también se les puede pasar información adicional en forma de parámetros:

`<param>` ➔ Indica el valor de los parámetros de un objeto.

- `name = "texto"` - Indica el nombre del parámetro.
- `value = "texto"` - Indica el valor del parámetro.

```
<object data="..." type="...">
  <param name="parametro1" value="40" />
  <param name="parametro2" value="20" />
  <param name="parametro3" value="texto de prueba" />
</object>
```

Uno de los principales inconvenientes de `<object>` es la forma de incluir vídeos en formato Flash en las páginas HTML:

```
<object data="nombre_video.swf" type="application/x-shockwave-
flash"></object>
```

Algunos navegadores pueden no visualizar el vídeo hasta que se ha descargado completamente. Solución alternativa → `<embed>` (etiqueta no válida en XHTML):

`<embed>` → Para “embeber” objetos en documentos.

- `src = "url"` - Indica la URL del archivo u objeto que se incluye en la página.
- `type = "tipo_de_contenido"` - Indica el tipo de contenido del objeto (*flash, quicktime, java*, etc.).
- `height = "unidad_de_medida"` - Indica la altura con la que se debe mostrar el objeto.
- `width = "unidad_de_medida"` - Indica la anchura con la que se debe mostrar el objeto.

```
<object width="425" height="350">
  <param name="movie"
    value="http://www.youtube.com/v/MsH0rBWCYjs">
  </param>
  <param name="wmode"
    value="transparent">
  </param>
  <embed src="http://www.youtube.com/v/MsH0rBWCYjs"
    type="application/x-shockwave-flash" wmode="transparent"
    width="425" height="350">
  </embed>
</object>
```