

## ***U.D. 3 Listas, imágenes, objetos y tablas en XHTML 1.0 Y HTML 5***

### **Unidad Didáctica 3**

Listas , imágenes, objetos y tablas

<a href="#">1. Listas .....</a>	<a href="#">2</a>
<a href="#">1.1. Listas no ordenadas .....</a>	<a href="#">2</a>
<a href="#">1.2. Listas ordenadas .....</a>	<a href="#">4</a>
<a href="#">1.3. Listas de definición .....</a>	<a href="#">5</a>
<a href="#">2. Imágenes y objetos .....</a>	<a href="#">9</a>
<a href="#">2.1. Imágenes .....</a>	<a href="#">9</a>
<a href="#">2.2. Mapas de imagen .....</a>	<a href="#">12</a>
<a href="#">2.3. Objetos .....</a>	<a href="#">15</a>
<a href="#">3. Tablas .....</a>	<a href="#">18</a>
<a href="#">7.1. Tablas básicas .....</a>	<a href="#">19</a>
<a href="#">7.2. Tablas avanzadas .....</a>	<a href="#">31</a>

# 1. Listas

En ocasiones, *es posible agrupar determinadas palabras o frases en un conjunto de elementos que tienen más significado de forma conjunta*. El menú de navegación de un sitio web por ejemplo está formado por un grupo de palabras. Aunque cada palabra por separado tiene sentido, de forma conjunta constituyen el menú de navegación de la página, por lo que su significado conjunto es mayor que por separado.

El lenguaje HTML define tres tipos diferentes de listas para agrupar los elementos: **listas no ordenadas** (se trata de una colección simple de elementos en la que no importa su orden), **listas ordenadas** (similar a la anterior, pero los elementos están numerados y por tanto, importa su orden) y **listas de definición** (un conjunto de términos y definiciones similar a un diccionario).

## 1.1. Listas no ordenadas

Las listas no ordenadas son las más sencillas y las que más se utilizan. Una lista no ordenada es un conjunto de elementos relacionados entre sí pero para los que no se indica un orden o secuencia determinados. La etiqueta <ul> encierra todos los elementos de la lista y la etiqueta <li> cada uno de sus elementos.

<ul>	Lista no ordenada
Atributos comunes	básicos, i18n y eventos
Atributos específicos	-
Tipo de elemento	Bloque
Descripción	Se emplea para definir listas no ordenadas

<li>	Elemento de una lista
Atributos comunes	básicos, i18n y eventos
Atributos específicos	-
Tipo de elemento	Bloque
Descripción	Se emplea para definir los elementos de las listas (ordenadas y no ordenadas)

El siguiente código HTML muestra un ejemplo sencillo de lista no ordenada:

```
<html>

  <head>

    <title>Ejemplo de etiqueta UL</title>

  </head>

  <body>

    <h1>Menú</h1>

    <ul>
      <li>Inicio</li>
      <li>Noticias</li>
      <li>Artículos</li>
      <li>Contacto</li>
    </ul>

  </body>

</html>
```



Figura. Ejemplo de uso de la etiqueta ul

El navegador por defecto muestra los elementos de la lista tabulados y con una pequeña viñeta formada por un círculo negro. Como ya se sabe, el aspecto con el que se muestran los elementos de las listas se puede modificar mediante las hojas de estilos CSS.

## 1.2. Listas ordenadas

Las listas ordenadas son casi idénticas a las listas no ordenadas, salvo que en este caso los elementos relacionados se muestran siguiendo un orden determinado.

Se define mediante la etiqueta `<ol>`. Los elementos de la lista se definen mediante la etiqueta `<li>`, la misma que se utiliza en las listas no ordenadas.

<b>&lt;ol&gt;</b>	Lista ordenada
<b>Atributos comunes</b>	básicos, id y eventos
<b>Atributos específicos</b>	-
<b>Tipo de elemento</b>	Bloque
<b>Descripción</b>	Se emplea para definir listas ordenadas

El siguiente código HTML muestra un ejemplo sencillo de lista ordenada:

```
<html>
  <head>
    <title>Ejemplo de etiqueta OL</title>
  </head>
  <body>
    <h1>Instrucciones</h1>

    <ol>
      <li>Enchufar correctamente</li>
      <li>Comprobar conexiones</li>
      <li>Encender el aparato</li>
    </ol>

  </body>
</html>
```

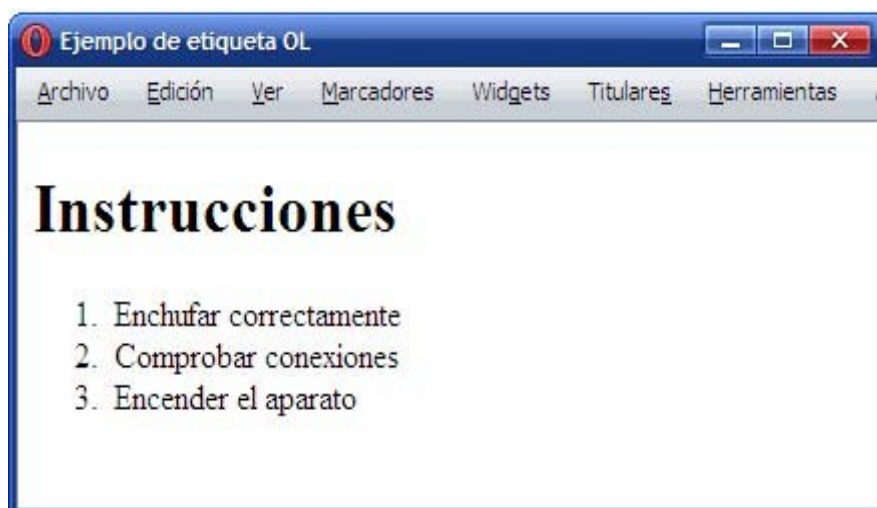


figura 5.2. Ejemplo de uso de la etiqueta ol

El navegador muestra la lista de forma muy parecida a las listas no ordenadas, salvo que en este caso no se emplean viñetas gráficas en los elementos, sino que se numeran de forma consecutiva. El tipo de numeración empleada también se puede modificar aplicando hojas de estilos CSS a los elementos de la lista.

### 1.3. Listas de definición

Las listas de definición apenas se utilizan en la mayoría de páginas HTML.

Su funcionamiento es similar al de un diccionario, ya que cada elemento de la lista está formado por términos y definiciones. La etiqueta `<dl>` crea la lista de definición y las etiquetas `<dt>` y `<dd>` definen respectivamente el término y la descripción de cada elemento de la lista.

### U.D. 3 Listas, imágenes, objetos y tablas en XHTML 1.0 Y HTML 5

<b>&lt;dt&gt;</b>	Término de una definición
<b>Atributos comunes</b>	básicos, i18n y eventos
<b>Atributos específicos</b>	-
<b>Tipo de elemento</b>	Bloque
<b>Descripción</b>	Se emplea para definir los términos de los elementos de una lista de definición

<b>&lt;dd&gt;</b>	Descripción de una definición
<b>Atributos comunes</b>	básicos, i18n y eventos
<b>Atributos específicos</b>	-
<b>Tipo de elemento</b>	Bloque
<b>Descripción</b>	Se emplea para indicar las definiciones de los elementos de una lista de definición

<b>&lt;dl&gt;</b>	Lista de definición
<b>Atributos comunes</b>	básicos, i18n y eventos
<b>Atributos específicos</b>	-
<b>Tipo de elemento</b>	Bloque
<b>Descripción</b>	Se emplea para definir listas de definición

El siguiente código HTML muestra un ejemplo sencillo de lista de definición:

```
<html>

  <head>

    <title>Ejemplo de etiqueta DL</title>

  </head>

  <body>

    <h1>Metalenguajes</h1>

    <dl>
      <dt>SGML</dt>
      <dd>Metalenguaje para la definición de otros lenguajes de
        marcado</dd>

      <dt>XML</dt>
      <dd>Lenguaje basado en SGML y que se emplea para describir
        datos</dd>
    </dl>
  </body>
</html>
```

### U.D. 3 Listas, imágenes, objetos y tablas en XHTML 1.0 Y HTML 5

```
<dt>RSS</dt>
<dt>GML</dt>
<dt>XHTML</dt>
<dt>SVG</dt>
<dt>XUL</dt>
<dd>Lenguajes derivados de XML para determinadas aplicaciones</dd>
</dl>

</body>
</html>
```

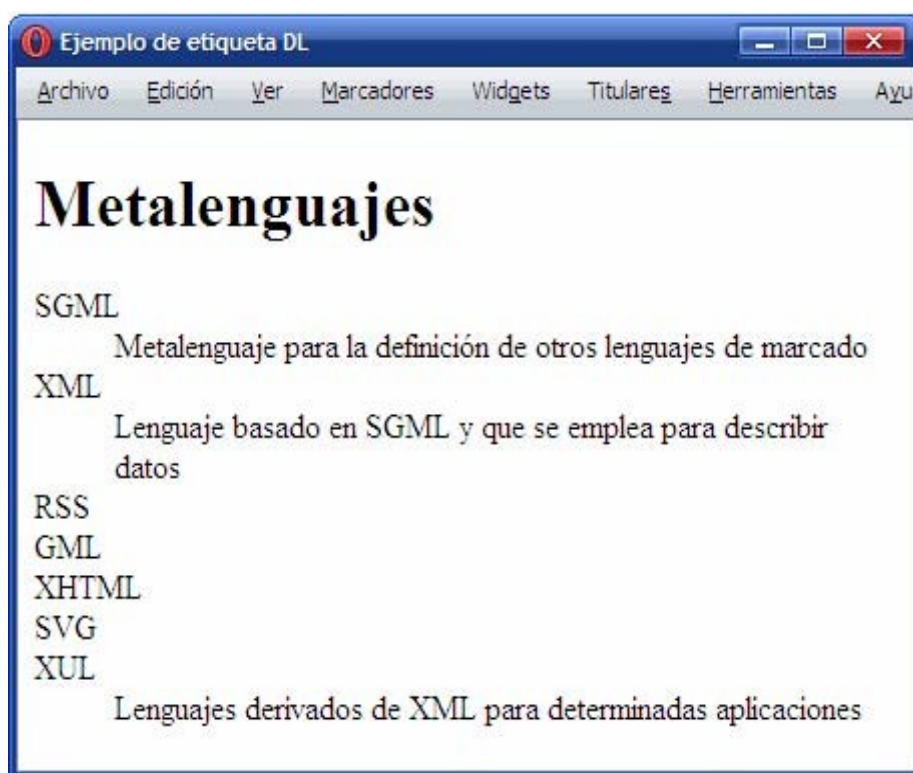


Figura 1.3. Ejemplo de uso de la etiqueta dl

Los navegadores formatean las listas de definición de forma similar a las otras listas, tabulando la definición y alineando a la izquierda los términos. Aunque no es habitual, cada término puede tener asociada más de una definición y cada definición puede tener asociada varios términos.

Ejercicio 2 bol2 :Determinar el código HTML que corresponde a la siguiente lista anidada simple.



Figura . Ejemplo de lista anidada simple de dos niveles



## 2. Imágenes y objetos

### 2.1. Imágenes

Dentro de las imágenes que se pueden incluir en una página HTML se deben distinguir dos tipos: las **imágenes de contenido** y las **imágenes de adorno**.

Las imágenes de contenido son las que proporcionan información y complementan la información textual. Las imágenes de contenido se incluyen directamente en el código HTML mediante la etiqueta `<img>`.

Las imágenes *de adorno* son las que se utilizan para hacer bordes redondeados, para mostrar pequeños iconos en las listas de elementos, para mostrar fondos de página, etc. Las imágenes de adorno no se deberían incluir en el código HTML, sino que deberían emplearse hojas de estilos CSS para mostrarlas.

A continuación se muestra la definición de la etiqueta `<img>`, utilizada para incluir las imágenes en las páginas HTML:

<b>&lt;img&gt;</b>	Imagen
<b>Atributos comunes</b>	básicos, i18n y eventos
<b>Atributos específicos</b>	<ul style="list-style-type: none"><li>▪ <code>src</code> = "url" - Indica la URL de la imagen que se muestra</li><li>▪ <code>alt</code> = "texto" - Descripción corta de la imagen</li><li>▪ <code>longdesc</code> = "url" - Indica una URL en la que puede encontrarse una descripción más detallada de la imagen</li><li>▪ <code>name</code> = "texto" - Nombre del elemento imagen</li><li>▪ <code>height</code> = "unidad_de_medida" - Indica la altura con la que se debe mostrar la imagen (no es obligatorio que coincida con la altura original de la imagen)</li><li>▪ <code>width</code> = "unidad_de_medida" - Indica la anchura con la que se debe mostrar la imagen (no es obligatorio que coincida con la anchura original de la imagen)</li></ul>
<b>Tipo de elemento</b>	En línea y etiqueta vacía
<b>Descripción</b>	Se emplea para incluir imágenes en los documentos

Los dos atributos requeridos son `src` y `alt`. El atributo `src` es similar al atributo `href` de los enlaces, ya que establece la URL de la imagen que se va a mostrar en la página. Las URL indicadas pueden ser absolutas o relativas.

El atributo `alt` permite describir el contenido de la imagen mediante un texto breve. Las descripciones deben tener una longitud inferior a 1024 caracteres y son útiles

para las personas y dispositivos discapacitados que no pueden acceder a las imágenes.

Ejemplo sencillo para incluir una imagen:

```

```

Como `<img>` es una etiqueta vacía, no tiene etiqueta de cierre. Como ya se explicó anteriormente, para cerrar una etiqueta vacía se incluyen los caracteres `>` al final de la etiqueta.

HTML no impone ninguna restricción sobre el formato gráfico que se puede utilizar en las imágenes, por lo que en principio la etiqueta `<img>` puede incluir cualquier formato gráfico existente. Sin embargo, si la imagen utiliza un formato poco habitual, todos o algunos navegadores no serán capaces de mostrar esa imagen.

La recomendación es utilizar uno de los tres siguientes formatos gráficos que entienden todos los navegadores modernos: GIF, JPG y PNG. El formato PNG presenta el inconveniente de que los navegadores obsoletos como Internet Explorer 6 no muestran correctamente las imágenes con transparencias de 24 bits.

El atributo `longdesc` no se utiliza de forma habitual, pero permite indicar la URL en la que se puede encontrar más información sobre la imagen. Como el atributo `alt` sólo permite incluir descripciones de hasta 1024 caracteres, el atributo `longdesc` se emplea con las imágenes complejas que necesitan mucha información para ser descritas:

```

```

```

```

En el ejemplo anterior, las dos imágenes se encuentran en el mismo directorio del servidor (`/imagenes/`). Se trata de una estrategia habitual en la mayoría de sitios web: guardar todas las imágenes de contenido en un directorio especial independiente del resto de contenidos HTML. Además, el directorio siempre suele llamarse de la misma manera: "imagenes" o "images" en inglés.

Los atributos `width` y `height` se utilizan para indicar la anchura y altura con la que se muestran las imágenes, por lo que son los más contradictorios. Como ya se ha comentado, HTML estructura de forma correcta los contenidos de la página y CSS define el aspecto gráfico con el que se muestran los contenidos. En principio, la anchura y la altura con la que se muestra una imagen es parte de su aspecto gráfico, por lo que debería ser propio de CSS y no de XHTML.

### U.D. 3 Listas, imágenes, objetos y tablas en XHTML 1.0 Y HTML 5

Sin embargo, en la práctica no es viable establecer la anchura y altura de todas las imágenes de contenidos mediante CSS. Si el sitio web dispone de muchas imágenes, la sobrecarga de estilos diferentes que debería definir CSS sería contraproducente. Por este motivo, los atributos width y height son la excepción a la norma de que el código HTML no haga referencia al aspecto de los contenidos.

```

```

```

```

Si el valor del atributo width o height se indica mediante un número entero, el navegador supone que hace referencia a la unidad de medida píxel. Por tanto, en el ejemplo anterior, la primera foto se muestra con una anchura de 200 píxel y una altura de 350 píxel.

También es posible indicar la anchura y altura en forma de porcentaje. En este caso, el porcentaje hace referencia a la altura/anchura del elemento en el que está contenida la imagen. Si la imagen no se encuentra dentro de ningún otro elemento, hace referencia a la anchura/ altura total de la página.

```
<div>
  
</div>
```

El ejemplo anterior mezcla los dos tipos de medidas que se pueden utilizar, para indicar que la foto tiene una anchura igual al 30% de la anchura del elemento <div> que la contiene y una altura de 350 píxel.

La anchura/altura con la que se muestra una imagen no tiene que coincidir obligatoriamente con la anchura/altura real de la imagen. Sin embargo, cuando estos valores no coinciden, las imágenes se muestran deformadas y el aspecto final es muy desagradable.

Si solamente se establece la altura de la imagen, el navegador calcula la anchura necesaria para que se mantenga la proporción de la imagen. De la misma forma, si sólo se establece la anchura de la imagen, el navegador calcula la altura que hace que la imagen se siga viendo con las mismas proporciones.

**Ejercicio 5 bol2** Modificar la página de índice del portfolio de los ejercicios 6 y 7 para mostrar directamente las imágenes de los proyectos.



Figura . Nueva página del portfolio que muestra la imagen de cada uno de los proyectos

## 2.2. Mapas de imagen

Aunque el uso de los mapas de imagen se ha reducido drásticamente en los últimos años, aún se utilizan en algunos sitios especializados. Muchas agencias de viaje y sitios relacionados utilizan mapas geográficos para seleccionar el destino del viaje.

Un mapa de imagen permite definir diferentes zonas "pinchables" dentro de una imagen. El usuario puede pinchar sobre cada una de las zonas definidas y cada una de ellas puede apuntar a una URL diferente. Siguiendo el ejemplo anterior, una sola imagen que muestre un mapa de todos los continentes puede definir una zona diferente para cada continente. De esta forma, el usuario puede pinchar sobre la zona correspondiente a cada continente para que el navegador muestre la página que contiene los viajes disponibles a ese destino.

Las zonas o regiones que se pueden definir en una imagen se crean mediante rectángulos, círculos y polígonos.

Para crear un mapa de imagen, en primer lugar se inserta la imagen original mediante la etiqueta <img>. A continuación, se utiliza la etiqueta <map> para definir las zonas o regiones de la imagen. Cada zona se define mediante la etiqueta <area>.

<b>&lt;map&gt;</b>	Mapa de imagen
<b>Atributos comunes</b>	básicos, i18n y eventos
<b>Atributos específicos</b>	<ul style="list-style-type: none"> <li>▪ name = "texto" - Nombre que identifica de forma única al mapa definido (es obligatorio indicar un nombre único)</li> </ul>
<b>Tipo de elemento</b>	Bloque y en línea
<b>Descripción</b>	Se emplea para definir mapas de imagen

<b>&lt;area&gt;</b>	Area de un mapa de imagen
<b>Atributos comunes</b>	básicos, i18n, eventos y foco
<b>Atributos específicos</b>	<ul style="list-style-type: none"> <li>▪ href = "url" - URL a la que se accede al pinchar sobre el área</li> <li>▪ nohref = "nohref" - Se emplea para las áreas que no son seleccionables</li> <li>▪ shape = "default   rect   circle   poly" - Indica el tipo de área que se define (toda la imagen, rectangular, circular o poligonal)</li> <li>▪ 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,...,XnYn (coordenadas de los vértices del polígono. Si las últimas coordenadas no son iguales que las primeras, se cierra automáticamente el polígono uniendo ambos vértices)</li> </ul>
<b>Tipo de elemento</b>	Etiqueta vacía
<b>Descripción</b>	Se emplea para definir las distintas áreas que forman un mapa de imagen

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>
```

### U.D. 3 Listas, imágenes, objetos y tablas en XHTML 1.0 Y HTML 5

Las áreas se definen mediante el atributo `shape` que indica el tipo de área y `coords` que es una lista de coordenadas cuyo significado depende del tipo de área definido. El enlace de cada área se define mediante el atributo `href`, con la misma sintaxis y significado que para los enlaces normales.

El siguiente ejemplo muestra una imagen sencilla que contiene cuatro figuras geométricas: Utilizando un círculo, dos rectángulos y un polígono se pueden definir fácilmente cuatro zonas *pinchables* en la imagen mediante el siguiente código HTML:

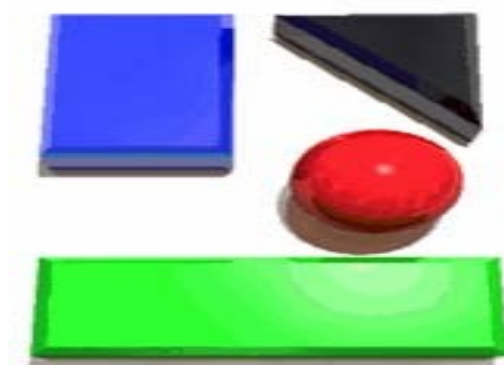


figura . Ejemplo de imagen que incluye un mapa de imagen

```
<p>
<map id="mapa_zonas" >
  <area shape="rect" coords="20,25,84,113" alt="r" href="cuadrado.html"
    onclick='alert("Has pulsado en el nmero dos");'/>
  <area shape="poly" coords="90,25,162,26,163,96,89,25,90,24" href="triangulo.html"alt="r" />
  <area shape="circle" coords="130,114,29" href="circulo.html"alt="r" />
  <area shape="rect" coords="19,156,170,211" href="mailto:rectangulo@direccion.com"alt="r"/>
  <area shape="default" nohref="nohref" alt="r"/>
</map></p>
```

## 2.3. Objetos

Además de las imágenes, HTML permite incluir en las páginas web otros elementos mucho más complejos, como *applets* de Java y vídeos en formato QuickTime o Flash. La mayoría de este tipo de contenidos no los interpreta el navegador directamente, sino que hace uso de pequeños programas llamados *plugins* y que se encargan de tratar con este tipo de elementos complejos.

La etiqueta <object> es la que permite "*embeber*" o incluir en las páginas HTML cualquier tipo de contenido complejo:

<b>&lt;object&gt;</b>	Objeto
<b>Atributos comunes</b>	básicos, i18n y eventos
<b>Atributos específicos</b>	<ul style="list-style-type: none"> <li>▪ <code>data = "url"</code> - Indica la URL de los datos que utiliza el objeto</li> <li>▪ <code>classid</code>, <code>codebase</code>, <code>codetype</code> - Información específica que depende del tipo de objeto</li> <li>▪ <code>type</code> - Indica el tipo de contenido de los datos</li> <li>▪ <code>height = "unidad_de_medida"</code> - Indica la altura con la que se debe mostrar el objeto</li> <li>▪ <code>width = "unidad_de_medida"</code> - Indica la anchura con la que se debe mostrar el objeto</li> </ul>
<b>Tipo de elemento</b>	Bloque y en línea
<b>Descripción</b>	Se emplea para embeber objetos en los documentos

El atributo `data` se emplea para indicar la URL del recurso que se va a incluir.

El atributo `type` indica el tipo de contenido de los datos del objeto. Los posibles valores de `type` están estandarizados y coinciden con los del atributo `type` de la etiqueta `<a>` que se explicó anteriormente.

El propio estándar de HTML incluye ejemplos de uso de esta etiqueta. Incluir un vídeo



### U.D. 3 Listas, imágenes, objetos y tablas en XHTML 1.0 Y HTML 5

en formato MPEG:

```
<object data="PlanetaTierra.mpeg" type="application/mpeg" />
```

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 cualquiera de los 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 siguiente texto -->
  La <strong>Tierra</strong> vista desde el espacio. </object> </object> </object>
```

A los objetos también se les puede pasar información adicional en forma de parámetros mediante la etiqueta <param>:

<param>	Parámetros de un objeto
Atributos comunes	id
Atributos específicos	<ul style="list-style-type: none"><li>▪ name = "texto" - Indica el nombre del parámetro</li><li>▪ value = "texto" - Indica el valor del parámetro</li></ul>
Tipo de elemento	Etiqueta vacía
Descripción	Se emplea para indicar el valor de los parámetros del objeto

Las etiquetas <param> siempre se incluyen en el interior de las etiquetas <object>:

```
<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. Si se utiliza el siguiente código:

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

El elemento anterior es correcto desde el punto de vista técnico, pero provoca que algunos navegadores como Internet Explorer no visualicen el vídeo hasta que se ha descargado completamente. Si se trata de un vídeo largo, esta solución no es válida para el usuario.



### U.D. 3 Listas, imágenes, objetos y tablas en XHTML 1.0 Y HTML 5

Por este motivo, se utiliza una solución alternativa para incluir vídeos Flash en las páginas HTML: el uso de la etiqueta <embed>. Aunque esta solución funciona correctamente, no se trata de una solución válida desde el punto de vista del estándar de XHTML, por lo que las páginas que incluyan esta solución no pasarán correctamente el proceso de validación.

Este es el motivo por el que los sitios web más populares de vídeos en formato Flash proporcionan un código similar al siguiente para incluir sus vídeos en las páginas HTML:

```
<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>
```

Una vez más, se debe tener en cuenta que la solución anterior de utilizar la etiqueta <embed> es correcta desde el punto de vista del usuario (no tiene que esperar a que el vídeo se descargue completamente para poder verlo) pero no es una solución técnicamente válida, ya que la etiqueta <embed> no es parte del estándar XHTML.

<b>&lt;embed&gt;</b>	Embeber objetos
<b>Atributos comunes</b>	básicos, i18n y eventos
<b>Atributos específicos</b>	<ul style="list-style-type: none"><li>▪ <code>src = "url"</code> - Indica la URL del archivo u objeto que se incluye en la página</li><li>▪ <code>type = "tipo_de_contenido"</code> - Indica el tipo de contenido del objeto (flash, quicktime, java, etc.)</li><li>▪ <code>height = "unidad_de_medida"</code> - Indica la altura con la que se debe mostrar el objeto</li><li>▪ <code>width = "unidad_de_medida"</code> - Indica la anchura con la que se debe mostrar el objeto</li></ul>
<b>Tipo de elemento</b>	Bloque
<b>Descripción</b>	Se emplea para embeber objetos en los documentos

## 3. Tablas

Desde sus primeras versiones, HTML incluyó el soporte para crear tablas de datos en las páginas web. Además de ser sencillo, el modelo definido por HTML es muy flexible y bastante completo.

El diagrama muestra una tabla con el título 'Cursos de diseño gráfico'. Las etiquetas con flechas indican: 'título de tabla' (punto al título), 'cabecera de columna' (punto a la fila de encabezados), 'cabecera de tabla' (punto a la primera columna), 'fila' (punto a una fila de datos), 'cabecera de fila' (punto a la primera fila de datos) y 'columna' (punto a una columna de datos).

Nombre	Horas	Plazas	Horario
Introducción a XHTML	20	20	09:00 – 13:00
CSS avanzado	40	15	16:00 – 20:00
Taller de usabilidad	40	10	16:00 – 20:00
Introducción a AJAX	60	20	08:30 – 12:30

Figura. Partes que componen una tabla compleja

Las tablas en HTML utilizan los mismos conceptos de filas, columnas, cabeceras y títulos que los que se utilizan en cualquier otro entorno de publicación de documentos:

Las tablas de HTML puede contener elementos simples, agrupaciones de filas y de columnas, cabeceras y pies de tabla, subdivisiones, cabeceras múltiples y otros elementos complejos.

Aunque parezca obvio, las tablas se deben utilizar para mostrar información tabular.

Hasta hace unos años, las tablas también se utilizaban para definir la estructura de las páginas web. La cabecera de la página era una fila de una gran tabla, el pie de página era otra fila de esta tabla y la zona de contenidos estaba formada por varias columnas dentro de esa gran tabla.

Aunque algunos malos diseñadores siguen utilizando hoy en día las tablas para definir la estructura completa de las páginas web, se trata de una técnica obsoleta y nada recomendable. El motivo es que se complica en exceso el código HTML y su mantenimiento es muy complejo. La solución correcta para definir la estructura de las páginas consiste en la utilización de hojas de estilos CSS.

## 7.1. Tablas básicas

Las tablas más sencillas de HTML se definen con tres etiquetas:

- <table> para crear la tabla,
- <tr> para crear cada fila y
- <td> para crear cada columna.

A continuación se muestra el código HTML de una tabla sencilla:

```
<html>
  <head>
    <title>Ejemplo de tabla sencilla</title>
  </head>
  <body>
    <h1>Listado de cursos</h1>

    <table>
      <tr>
        <td><strong>Curso</strong></td>
        <td><strong>Horas</strong></td>
        <td><strong>Horario</strong></td>
      </tr>

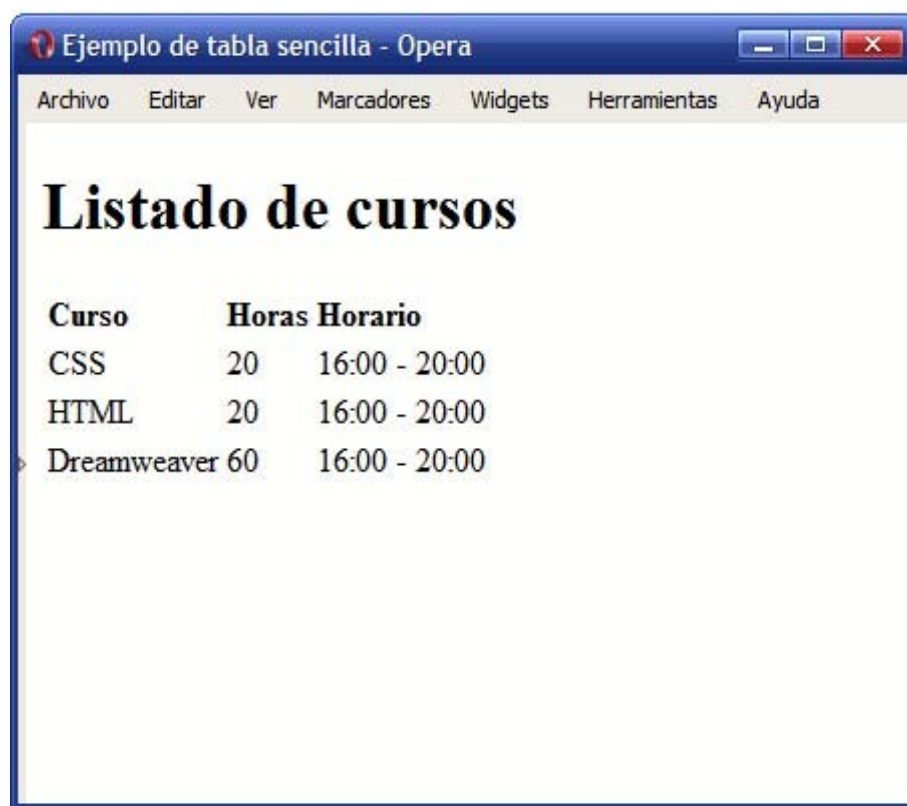
      <tr>
        <td>CSS</td>
        <td>20</td>
        <td>16:00 - 20:00</td>
      </tr>

      <tr>
        <td>HTML</td>
        <td>20</td>
        <td>16:00 - 20:00</td>
      </tr>

      <tr>
        <td>Dreamweaver</td>
        <td>60</td>
        <td>16:00 - 20:00</td>
      </tr>
    </table>

  </body>
</html>
```

Si se visualiza el código anterior en cualquier navegador, se obtiene una tabla como la que muestra la siguiente imagen:



Curso	Horas	Horario
CSS	20	16:00 - 20:00
HTML	20	16:00 - 20:00
Dreamweaver	60	16:00 - 20:00

Figura Ejemplo de tabla sencilla creada con las etiquetas table, tr y td

La etiqueta <table> encierra todas las filas y columnas de la tabla. Las etiquetas <tr> (del inglés *"table row"*) definen cada fila de la tabla y encierran todas las columnas. Por último, la etiqueta <td> (del inglés *"table data cell"*) define cada una de las columnas de las filas, aunque realmente HTML no define columnas sino *celdas de datos*.

Al definir una tabla, se debe pensar en primer lugar en las filas que la forman y a continuación en las columnas. El motivo es que HTML procesa primero las filas y por eso las etiquetas <tr> aparecen antes que las etiquetas <td>.

### U.D. 3 Listas, imágenes, objetos y tablas en XHTML 1.0 Y HTML 5

<b>&lt;table&gt;</b>	Tabla
<b>Atributos comunes</b>	básicos, i18n y eventos
<b>Atributos específicos</b>	<ul style="list-style-type: none"> <li>▪ <code>summary</code> = "texto" - Permite describir el contenido de la tabla (lo utilizan los buscadores y las personas discapacitadas)</li> </ul>
<b>Tipo de elemento</b>	Bloque
<b>Descripción</b>	Se emplea para definir tablas de datos
<b>&lt;tr&gt;</b>	Fila de tabla
<b>Atributos comunes</b>	básicos, i18n y eventos
<b>Atributos específicos</b>	-
<b>Tipo de elemento</b>	Bloque
<b>Descripción</b>	Se emplea para definir cada fila de las tablas de datos

<b>&lt;td&gt;</b>	Celda de tabla
<b>Atributos comunes</b>	básicos, i18n y eventos
<b>Atributos específicos</b>	<ul style="list-style-type: none"> <li>▪ <code>abbr</code> = "texto" - Permite describir el contenido de la celda (se emplea sobre todo con los navegadores de voz utilizados por personas discapacitadas)</li> <li>▪ <code>headers</code> = "lista_de_id" - Indica las celdas que actúan como cabeceras para esta celda (los títulos de las columnas y filas). Se indica como una lista de valores del atributo "id" de celdas</li> <li>▪ <code>scope</code> = "col, row, colgroup, rowgroup" - Indica las celdas para las que esta celda será su cabecera. Ej: <code>scope="col"</code> indica que esta celda es la cabecera de todas las demás celdas que están en la misma columna</li> <li>▪ <code>colspan</code> = "numero" - Número de columnas que ocupa esta celda</li> <li>▪ <code>rowspan</code> = "numero" - Número de filas que ocupa esta celda</li> </ul>
<b>Tipo de elemento</b>	Bloque
<b>Descripción</b>	Se emplea para definir cada una de las celdas que forman las filas de una tabla, es decir, las columnas de la tabla

De todos los atributos disponibles para las celdas, los más utilizados son `rowspan` y `colspan`, que se emplean para construir tablas complejas como las que se ven más adelante. Entre los demás atributos, sólo se utiliza de forma habitual el atributo `scope`, sobre todo con las celdas de cabecera que se ven a continuación.

Normalmente, algunas de las celdas de la tabla se utilizan como cabecera de las demás celdas de la fila o de la columna. En este caso, HTML define la etiqueta `<th>` (del inglés *"table header cell"*) para indicar que una celda es cabecera de otras celdas.

<b>&lt;th&gt;</b>	Celda cabecera de tabla
<b>Atributos comunes</b>	básicos, i18n y eventos
<b>Atributos específicos</b>	<ul style="list-style-type: none"><li>▪ <code>abbr</code> = "texto" - Permite describir el contenido de la celda (se emplea sobre todo con los navegadores de voz utilizados por personas discapacitadas)</li><li>▪ <code>headers</code> = "lista_de_id" - Indica las celdas que actúan como cabeceras para esta celda (los títulos de las columnas y filas). Se indica como una lista de ID de celdas</li><li>▪ <code>scope</code> = "col, row, colgroup, rowgroup" - Indica las celdas para las que esta celda será su cabecera. Ej: <code>scope="col"</code> indica que esta celda es la cabecera de todas las demás celdas que están en la misma columna</li><li>▪ <code>colspan</code> = "numero" - Número de columnas que ocupa esta celda</li><li>▪ <code>rowspan</code> = "numero" - Número de filas que ocupa esta celda</li></ul>
<b>Tipo de elemento</b>	Bloque
<b>Descripción</b>	Se emplea para definir las celdas que son cabecera de una fila o de una columna de la tabla

Los atributos de la etiqueta `<th>` son idénticos que los atributos definidos para la etiqueta `<td>`. En este caso, el atributo más utilizado es `scope`, que permite indicar si la celda es cabecera de la fila o de la columna (`<th scope="row">` y `<th scope="col">` respectivamente).

Por otra parte, HTML define la etiqueta `<caption>` para establecer la leyenda o título de una tabla. La etiqueta debe colocarse inmediatamente después de la etiqueta `<table>` y cada tabla sólo puede incluir una etiqueta `<caption>`.

<b>&lt;caption&gt;</b>	Leyenda o título de tabla
<b>Atributos comunes</b>	básicos, i18n y eventos
<b>Atributos específicos</b>	-
<b>Tipo de elemento</b>	En línea
<b>Descripción</b>	Se emplea para definir la leyenda o título de una tabla

**Ejercicio 6 bol2:** Determinar el código HTML necesario para crear la tabla que se muestra en la siguiente imagen:



Nombre producto	Precio unitario	Unidades	Subtotal
Reproductor MP3 (80 GB)	192.02	1	192.02
Fundas de colores	2.50	5	12.50
Reproductor de radio & control remoto	12.99	1	12.99
<b>TOTAL</b>	-	7	<b>207.51</b>

Figura Tabla sencilla con celdas de cabecera

**ejercicio 7 bol2:** Utilizar las celdas de cabecera donde sea necesario y añadir todos los atributos posibles.

Determinar el código HTML necesario para crear la tabla que se muestra en la siguiente imagen. Utilizar las celdas de cabecera donde sea necesario y añadir todos los atributos posibles.





Resultado de la búsqueda	
Imagen	Datos
	<p><a href="#">Portátil - 3 GHz - 4 GB RAM</a></p> <p><a href="#">Comprar:</a> <del>2.990 €</del> 2.599 €</p>
	<p><a href="#">Videocámara - Alta definición 1080p - 60 GB</a></p> <p><a href="#">Comprar:</a> <del>1.099 €</del> 999 €</p>
	<p><a href="#">Televisor - 46" - Full HD</a></p> <p><a href="#">Comprar:</a> <del>1.999 €</del> 1.799 €</p>
	<p><a href="#">Móvil - 3G - Wi-Fi - 8 GB</a></p> <p><a href="#">Comprar:</a> <del>399 €</del> 349 €</p>

Figura 7.4. Tabla con los resultados de una búsqueda

Las tablas complejas suelen disponer de una estructura irregular que une varias columnas para formar una columna ancha o une varias filas para formar una fila más alta que las demás. Para fusionar filas o columnas, se utilizan los atributos rowspan y colspan respectivamente.

La siguiente imagen muestra una tabla compleja que ha fusionado dos columnas simples para formar una columna más ancha:



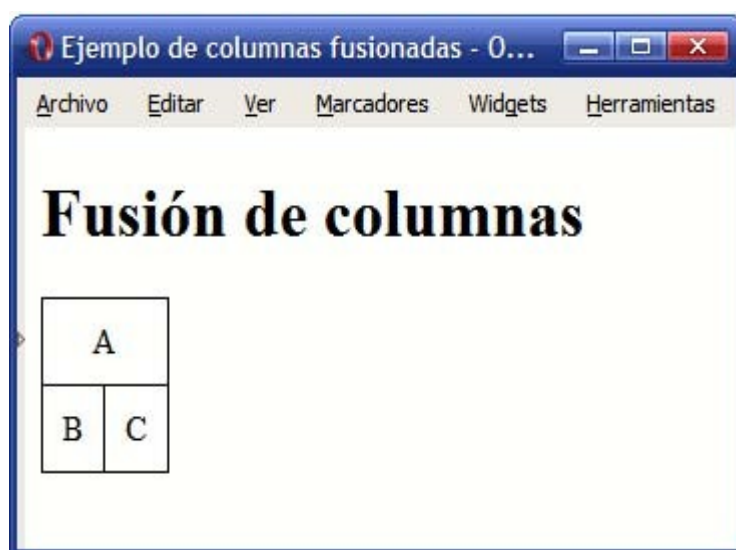


Figura . Ejemplo sencillo de fusión de columnas

Para obtener una tabla como la de la imagen anterior, se debe utilizar el siguiente código:

```
<table>

  <tr>

    <td colspan="2">A</td>

  </tr>

  <tr>

    <td>B</td>
    <td>C</td>

  </tr>

</table>
```

La primera fila de la tabla está formada sólo por una columna, mientras que la segunda fila está formada por dos columnas. En principio, podría pensarse en utilizar el siguiente código HTML para definir la tabla:

```
<table> <tr> <td>A</td> </tr>

<tr> <td>B</td> <td>C</td>
</tr> </table>
```

Sin embargo, si se utiliza el código anterior, el navegador visualiza de forma incorrecta la tabla, ya que las tablas en HTML deben disponer de una estructura regular. En otras palabras, todas las filas de una tabla HTML deben tener el mismo número de

columnas.

Por lo tanto, si se quieren mostrar menos columnas en una fila, se fusionan mediante el atributo **colspan**, que **indica el número de columnas simples que va a ocupar una determinada celda**.

En el ejemplo anterior, la celda de la primera fila debe ocupar el espacio de dos columnas simples, por lo que el código HTML debe ser `<td colspan="2">A</td>`.

De forma equivalente, si se quiere diseñar una tabla HTML que fusiona filas como la de la siguiente imagen:



Figura Ejemplo sencillo de fusión de filas

El código HTML que se debe utilizar para obtener la tabla de la imagen anterior es:

```
<table>
<tr>
  <td>A</td>
  <td rowspan="2">B</td>

</tr>

<tr>
  <td>C</td>

</tr>
</table>
```

De forma análoga a la fusión de columnas del ejemplo anterior, la fusión de filas debe

indicarse de forma especial. Como las tablas HTML tienen que ser regulares, todas las columnas deben tener el mismo número de filas. Así, si en el ejemplo anterior se utilizara el siguiente código:

```
<table>
  <tr>
    <td>A</td>
    <td>B</td>
  </tr>

  <tr>
    <td>C</td>
  </tr> </table>
```

La tabla anterior no se visualizaría correctamente. Como la segunda columna de la tabla ocupa el espacio de las dos filas, el código HTML debe indicar claramente que esa celda va a ocupar dos filas, de manera que todas las columnas de la tabla cuenten con el mismo número de filas.

Utilizando los atributos `rowspan` y `colspan`, es posible diseñar tablas tan complejas como las que se muestran en los siguientes ejemplos.

**Ejemplo de fusión de múltiples columnas:** El código HTML necesario para fusionar las columnas de la tabla anterior se muestra a continuación:

```
<html>

  <head><title>Ejemplo de columnas fusionadas</title></head>

  <body>

    <h1>Fusión de columnas</h1>

    <table>
      <tr> <td colspan="3">A</td>
        <td>B</td>
      </tr>

      <tr> <td>C</td>
        <td colspan="2">D</td>
        <td>E</td>
      </tr>

      <tr> <td colspan="4">F</td>
      </tr>

      <tr> <td>G</td>
        <td>H</td>
        <td>I</td>
        <td>J</td> </tr>
```

```
</table>  
</body> </html>
```

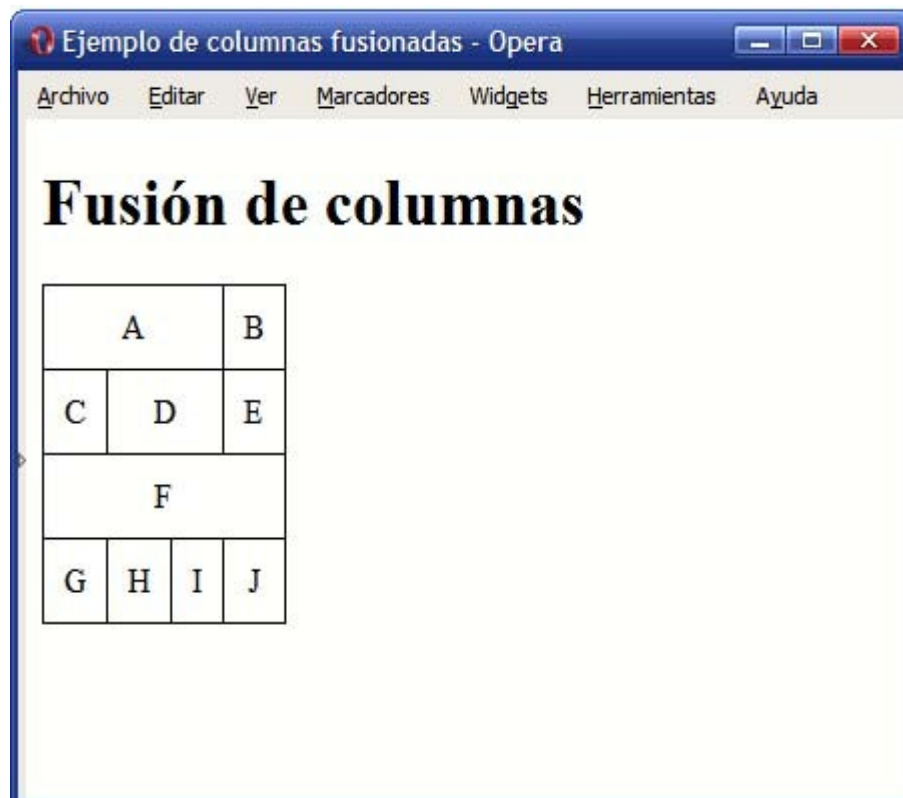


Figura . Ejemplo complejo de fusión de columnas

El código HTML necesario para fusionar las filas de la tabla siguiente se muestra a continuación:

A	B	C	D
E	F		G
	H		
I	J	K	

```
<html> <head><title>Ejemplo de filas fusionadas</title></head> <body>
```

```
<h1>Fusión de filas</h1>
```

```
<table>
```

```
<tr>
```

```
<td>A</td>
```

```
<td>B</td>
```

```
<td rowspan="3">C</td>
```

```
<td>D</td>
```

```
</tr>
```

```
<tr>
```

```
<td rowspan="2">E</td>
```

```
<td>F</td>
```

```
<td rowspan="3">G</td>
```

```
</tr>
```

```
<tr>
```

```
<td>H</td>
```

```
</tr>
```

```
<tr>
```

```
<td>I</td>
```

```
<td>J</td>
```

### ***U.D. 3 Listas, imágenes, objetos y tablas en XHTML 1.0 Y HTML 5***

```
<td>K</td>
</tr>

</table>

</body> </html>
```







Ejercicio 8 bol2: Determinar el código HTML necesario para crear la tabla que se muestra en la siguiente imagen: Emplear las etiquetas <table>, <tr>, <td>, <th>, <caption> y los atributos colspan, rowspan, abbr, scope, summary.

Ejemplo de tabla compleja - Opera

Archivo Editar Ver Marcadores Widgets Herramientas Ayuda

### Comparativa de reproductores MP3

Tabla comparativa de las características técnicas de los reproductores MP3

	 <b>MP3 mini</b>			 <b>MP3 grande</b>	
<b>Capacidad de almacenamiento</b>	4GB (1.000 canciones)	8GB (2.000 canciones)	16GB (4.000 canciones)	30GB (7.500 canciones)	80GB (20.000 canciones)
<b>Colores</b>					
<b>Pantalla</b>	LCD de 3 cm (diagonal) con retroiluminación			LCD de 6 cm (diagonal) con retroiluminación	
<b>Tiempo de carga</b>	Unas 3 horas			Unas 4 horas	
				Unas 2 horas para alcanzar el 80% de la capacidad	

## 7.2. Tablas avanzadas

Algunas tablas complejas están formadas por más elementos que filas y celdas de datos. Así, es común que las tablas más avanzadas dispongan de una sección de cabecera, una sección de pie y varias secciones de datos. Además, también es posible agrupar varias columnas de forma lógica para poder aplicar estilos similares a un

CONSOLIDATED BALANCE SHEET	2008			
	March	June	September	December
<b>Non-current assets</b>	<b>87.249</b>	<b>87.126</b>	<b>90.426</b>	<b>91.269</b>
Intangible assets	21.810	21.145	20.986	20.758
Goodwill	17.914	19.660	21.828	21.739
Property, plant and equipment and investment property	33.245	32.332	33.428	33.888
Long-term financial assets and other non-current assets	5.723	5.687	5.981	6.183
Deferred tax assets	8.557	8.303	8.202	8.702
<b>Current assets</b>	<b>18.042</b>	<b>17.979</b>	<b>19.128</b>	<b>17.713</b>
Inventories	1.154	1.134	1.052	1.012
Trade and other receivables	9.244	9.495	9.709	9.666
Current tax receivable	1.288	1.565	1.468	1.555
Short-term financial investments	1.877	1.803	1.788	1.679
Cash and cash equivalents	4.468	3.557	5.101	3.792
Non-current assets classified as held for sale	11	425	9	9
<b>Total Assets = Total Equity and Liabilities</b>	<b>105.291</b>	<b>105.106</b>	<b>109.554</b>	<b>108.982</b>
<b>Equity</b>	<b>15.714</b>	<b>15.072</b>	<b>19.185</b>	<b>20.001</b>
Equity attributable to equity holders of the parent	11.932	12.085	16.397	17.178
Minority interest	3.782	2.987	2.788	2.823
<b>Non-current liabilities</b>	<b>54.053</b>	<b>66.406</b>	<b>63.908</b>	<b>62.644,0</b>
Long-term financial debt	41.665	54.263	51.647	50.675
Deferred tax liabilities	4.868	4.617	4.727	4.700
Long-term provisions	6.466	6.507	6.545	6.287
Other long-term liabilities	1.054	1.020	988	982
<b>Current liabilities</b>	<b>35.523</b>	<b>23.628</b>	<b>26.462</b>	<b>26.337,0</b>
Short-term financial debt	19.507	7.466	8.975	8.382
Trade and other payables	8.792	8.259	8.782	8.533
Current tax payable	2.007	2.324	2.529	2.841
Short-term provisions and other liabilities	5.218	5.212	6.176	6.580
Liabilities associated with non-current assets classified as held for sale	0	367	0	0
<b>Financial Data</b>				
Net Financial Debt (1)	53.510	54.922	52.239	52.145

determinado grupo de columnas.

Un ejemplo clásico de tablas avanzadas es el de las tablas utilizadas en contabilidad, como por ejemplo la tabla que muestra el balance de una empresa:

Figura Ejemplo de tabla compleja correspondiente al balance de una empresa



<b>&lt;thead&gt;</b> <b>&lt;tbody&gt;</b> <b>&lt;tfoot&gt;</b>	Cabecera de tabla Sección de una tabla Pie de tabla
Atributos comunes	básicos, i18n y eventos
Atributos específicos	-
Tipo de elemento	Bloque
Descripción	Se emplean para agrupar varias filas en una cabecera (thead) un pie (tfoot) o una sección (tbody) de una tabla

Las partes que componen las tablas complejas se definen mediante las etiquetas <thead>, <tbody> y <tfoot>.

- La cabecera de la tabla se define con la etiqueta <thead>.
- el pie de la tabla se define mediante <tfoot>
- y cada sección de datos se define con una etiqueta <tbody>.

Cada tabla puede contener solamente una cabecera y un pie, pero puede incluir un número ilimitado de secciones. Si se define una cabecera y/o un pie, las etiquetas <thead> y/o <tfoot> deben colocarse inmediatamente antes que cualquier etiqueta <tbody>.

La siguiente imagen muestra una tabla avanzada con cabecera, pie y una sección de datos: El código HTML necesario para crear la tabla de la imagen anterior hace uso de las etiquetas <thead>, <tbody> y <tfoot>:

Análisis de ventas anuales				
AÑO	Expansión de ventas			
	Producto A	Producto B	Producto C	Producto D
N-3	-	-	-	-
N-2	3	5	8	4
N-1	4	4	7	3
N	5	7	6	2
AÑO	Producto A	Producto B	Producto C	Producto D
	Expansión de ventas			

Figura 7.11. Ejemplo de tabla avanzada con cabecera, pie y secciones

```
<html>
  <head>
    <title>Ejemplo de tabla avanzada</title>
  </head>
  <body>
    <h3>Análisis de ventas</h3>
    <table summary="Análisis de ventas anuales">
      <caption>Análisis de ventas anuales</caption>
      <thead>
        <tr>
          <th rowspan="2" scope="col">AÑO</th>
          <th colspan="4" scope="col">Expansión de ventas</th>
        </tr>
      </thead>
    </table>
  </body>
</html>
```

```
<tr>
    <th scope="col">Producto A</th>
    <th scope="col">Producto B</th>
    <th scope="col">Producto C</th>
    <th scope="col">Producto D</th>
</tr>
</thead>
<tfoot>
    <tr>
        <th rowspan="2" scope="col">AÑO</th>
        <th scope="col">Producto A</th>
        <th scope="col">Producto B</th>
        <th scope="col">Producto C</th>
        <th scope="col">Producto D</th>
    </tr>
    <tr> <th colspan="4" scope="col">Expansión de ventas</th>
    </tr>
</tfoot>
<tbody>
    <tr>
        <th scope="row">N-3</th>
        <td>-</td><td>-</td><td>-</td><td>-</td>
    </tr>
    <tr>
        <th scope="row">N-2</th>
        <td>3</td><td>5</td><td>8</td><td>4</td>
    </tr>
    <tr>
        <th scope="row">N-1</th>
        <td>4</td><td>4</td><td>7</td><td>3</td>
    </tr>
    <tr>
        <th scope="row">N</th>
        <td>5</td><td>7</td><td>6</td><td>2</td>
    </tr>
</tbody>
</table>
</body>
</html>
```

Aunque al principio resulta extraño, el elemento <tfoot> siempre se escribe antes que cualquier elemento <tbody> en el código HTML. De hecho, si la etiqueta <tfoot> aparece después de un elemento <tbody>, la página no se considera válida.

### U.D. 3 Listas, imágenes, objetos y tablas en XHTML 1.0 Y HTML 5

La etiqueta <tbody> permite realizar agrupaciones de filas, pero en ocasiones se necesitan agrupar columnas. Aunque su uso no es muy común, HTML define dos etiquetas similares para agrupar columnas: <col> y <colgroup>.

La etiqueta <col> se utiliza para asignar los mismos atributos a varias columnas de forma simultánea. De esta forma, la etiqueta <col> no agrupa columnas, sino que sólo asigna atributos comunes a varias columnas.

Una columna se define con el elemento COL, al principio de la tabla antes de crear las filas. La definición de una columna NO CREA CELDAS, las celdas se crean exclusivamente con los elementos TD y TH dentro de las filas. La definición de una columna es solamente una forma de poder referenciarla.

El primer elemento COL hace referencia a la primera columna de la tabla, el segundo elemento COL, a la segunda columna de la tabla, y así sucesivamente.

```
<TABLE>

  <COL width="10%">

    <!-- La 1a columna ocupará un 10% del total de la tabla -->

  <COL width="20%"/>

    <!-- La 2a columna ocupará un 20% del total de la tabla -->

    <!-- Del resto de columnas (si las hay) no dice nada -->

  <TR>

    <TD> <!-- Primera celda --> </TD>

    <!-- Resto de filas y celdas .....

  </TABLE>
```

Si las características que deseamos aplicar son las mismas para varias columnas consecutivas, podemos hacerlo con un solo COL, usando el atributo span:

```
<COL width="10%" span="3"/>
```

significa que la anchura del 10% se aplicará a las 3 columnas consecutivas.

NOTA: Obsérvese que a pesar de la similitud con colspan y rowspan, el significado de span es totalmente distinto: el valor de span indica el número de columnas a las que afectan las características que estamos detallando.

De la misma forma que hacíamos grupos de filas, también podemos hacer grupos de columnas. El elemento que nos permite definir los grupos de columnas es COLGROUP. Un elemento COLGROUP agrupa todos los elementos COL que formarán parte del grupo.

```
<COLGROUP>

    <COL width="10%"/>

    <COL width="10%"/>

    <COL width="10%"/>

</COLGROUP>

<COLGROUP width="70%">
```

El código anterior se aplica sobre una tabla de 4 columnas: las 3 primeras (que están agrupadas) tendrán una anchura del 10% del total de la anchura de la tabla cada una. La cuarta columna (que pertenece a otro grupo) tendrá una anchura del 70% de la tabla.

Este mismo resultado se puede conseguir con un código más pequeño como el que sigue:

```
<COLGROUP>

    <COL span="3" width="10%"/>

</COLGROUP>

<COLGROUP width="70%">
```

El primer elemento COL indica mediante el atributo span que la anchura del 10% afecta a 3 columnas.

También se puede conseguir el mismo resultado con el siguiente código:

```
<COLGROUP span="3" width="10%"> </COLGROUP>

<COLGROUP width="70%"> </COLGROUP>
```

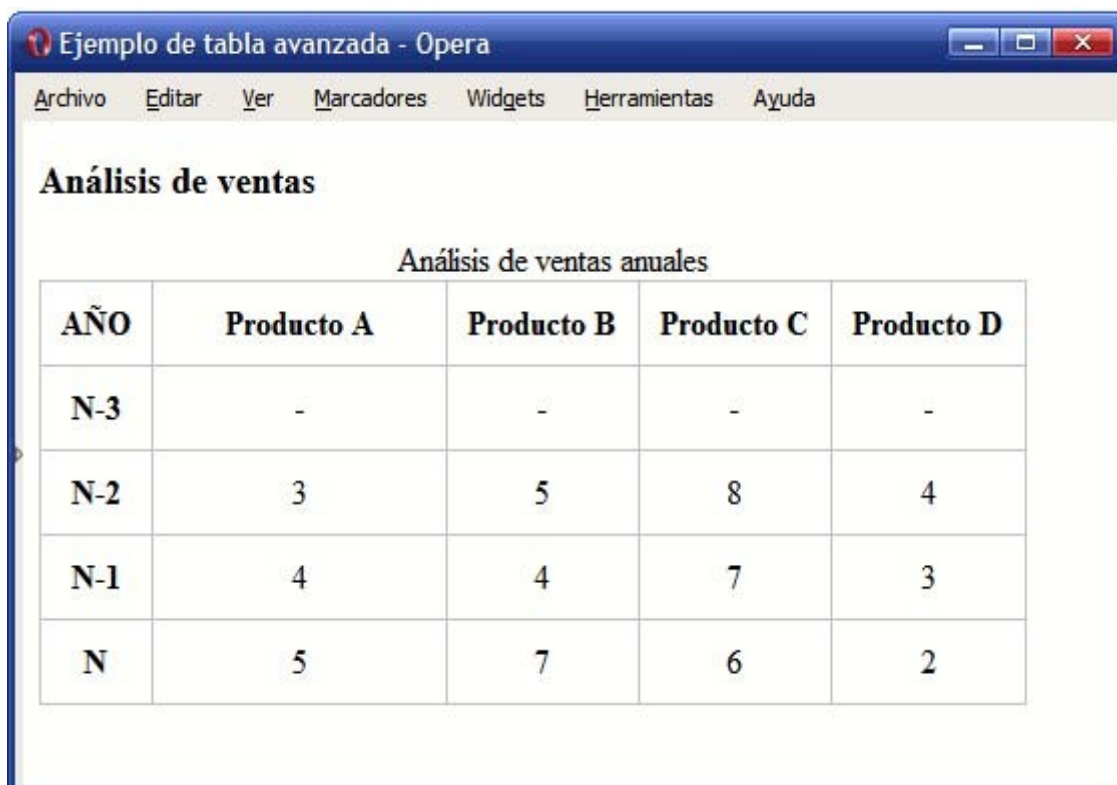
donde vemos que, cuando todas las columnas de un grupo son iguales, los atributos

### U.D. 3 Listas, imágenes, objetos y tablas en XHTML 1.0 Y HTML 5

se pueden expresar directamente en el elemento COLGROUP, omitiendo COL.

NOTA: La etiqueta final de COL está prohibida (siempre es un elemento vacío). La etiqueta final de COLGROUP es opcional.

La siguiente imagen muestra una tabla que hace uso de la etiqueta <col>:



Análisis de ventas anuales				
AÑO	Producto A	Producto B	Producto C	Producto D
N-3	-	-	-	-
N-2	3	5	8	4
N-1	4	4	7	3
N	5	7	6	2

Figura . Ejemplo de tabla avanzada que usa la etiqueta col

### U.D. 3 Listas, imágenes, objetos y tablas en XHTML 1.0 Y HTML 5

El código HTML necesario para crear la tabla anterior se muestra a continuación:

```
<table summary="Análisis de ventas anuales">

<caption>Análisis de ventas anuales</caption>

<col style="width:10%;" />

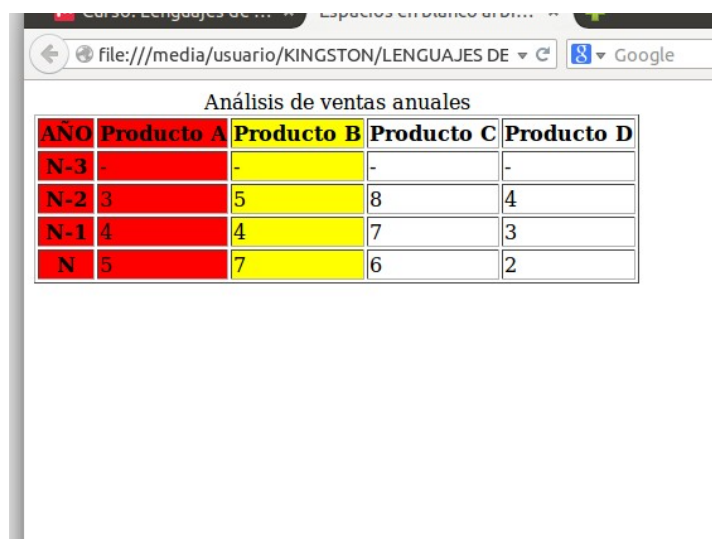
<col style="width:30%;" />

<thead>
  <tr>
    <th scope="col">AÑO</th>
    <th scope="col">Producto A</th>
    <th scope="col">Producto B</th>
    <th scope="col">Producto C</th>
    <th scope="col">Producto D</th>
  </tr>
</thead>
<tbody>
  <tr>
    <th scope="row">N-3</th>
    <td>-</td><td>-</td>
    <td>-</td>
    <td>-</td>
  </tr>
  <tr>
    <th scope="row">N-2</th>
    <td>3</td>
    <td>5</td>
    <td>8</td>
    <td>4</td>
  </tr>
  <tr>
    <th scope="row">N-1</th>
    <td>4</td>
    <td>4</td>
    <td>7</td>
    <td>3</td>
  </tr>
  <tr>
    <th scope="row">N</th>
    <td>5</td>
    <td>7</td>
    <td>6</td>
    <td>2</td>
  </tr>
</tbody>
</table>
```

### U.D. 3 Listas, imágenes, objetos y tablas en XHTML 1.0 Y HTML 5

Por otra parte, la etiqueta <colgroup> se emplea para agrupar de forma estructural varias columnas de la tabla. La forma habitual de indicar el número de columnas que abarca la agrupación es utilizar el atributo span, que establece el número de columnas de cada agrupación.

La siguiente imagen muestra una tabla avanzada con una agrupación de columnas realizada con la etiqueta <colgroup>:



The image shows a web browser window with a file:// URL. The page title is 'Análisis de ventas anuales'. It contains a table with 5 columns: 'AÑO', 'Producto A', 'Producto B', 'Producto C', and 'Producto D'. The first column 'AÑO' has a rowspan of 4. The second and third columns, 'Producto A' and 'Producto B', are grouped together using a <colgroup> tag with a span of 2. The table data is as follows:

AÑO	Producto A		Producto B	Producto C	Producto D
N-3	-	-	-	-	-
N-2	3	5	8	4	
N-1	4	4	7	3	
N	5	7	6	2	

Figura . Ejemplo de tabla avanzada que usa la etiqueta colgroup



El código HTML necesario para crear la tabla anterior se muestra a continuación:

```
<table summary="Análisis de ventas anuales" border="1">
  <caption>Análisis de ventas anuales</caption>
  <colgroup>
    <col span="2" style="background-color:red"/>
    <col style="background-color:yellow"/>
  </colgroup>
  <thead>
    <tr>
      <th scope="col">AÑO</th>
      <th scope="col">Producto A</th>
      <th scope="col">Producto B</th>
      <th scope="col">Producto C</th>
      <th scope="col">Producto D</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <th scope="row">N-3</th>
      <td>-</td>
      <td>-</td>
      <td>-</td>
      <td>-</td>
    </tr>
    <tr>
      <th scope="row">N-2</th>
      <td>3</td>
      <td>5</td>
      <td>8</td>
      <td>4</td>
    </tr>
    <tr>
      <th scope="row">N-1</th>
      <td>4</td>
      <td>4</td>
      <td>7</td>
      <td>3</td>
    </tr>
    <tr>
      <th scope="row">N</th>
      <td>5</td>
      <td>7</td>
      <td>6</td>
      <td>2</td>
    </tr>
  </tbody>
</table>
```

El uso de las etiquetas `<col>` y `<colgroup>` no está muy extendido, debido a que la mayoría de navegadores no soportan muchas de sus funcionalidades.

#### Representación incremental de tablas

La representación incremental de una tabla significa la posibilidad de que se comience a representar antes de que llegue toda la información. Si una tabla es demasiado grande y no permite la representación incremental, el usuario deberá esperar demasiado tiempo para ver su contenido.

No es necesario decir que una tabla estará mejor construida si permite la representación incremental.

Lo que necesita un agente de usuario para poder representar una tabla incrementalmente será la anchura total de la tabla (atributo `width` en el elemento `TABLE`) y el número y anchura de las columnas, por tanto tendremos que incluir la definición de columnas con `COL` y `COLGROUP`.

Como ya sabemos, hay 2 formas de expresar la anchura de columnas en HTML:

- > En términos absolutos (número de pixels que ocupa)
- > En términos porcentuales (% de la columna con respecto al total de la tabla)

EJEMPLO: la siguiente tabla tendrá dos columnas: la primera ocupa un ancho de 275 pixels y la segunda de 150 pixels:

```
<COL width="275"/>
```

```
<COL width="150"/>
```

EJEMPLO: la siguiente tabla tendrá 3 columnas: la primera ocupa la mitad del total, la segunda y la tercera una cuarta parte del total cada una:

```
<COL width="50%"/>
```

```
<COL width="25%"/>
```

```
<COL width="25%"/>
```

Estas dos formas de indicar la anchura se pueden usar conjuntamente. En el siguiente ejemplo la primera columna tiene 200 pixels de ancho, la segunda ocupa el 25% del total y el resto de columnas (si las hay) se reparten el espacio que queda.

```
<COL width="200"/>
```

```
<COL width="25%"/>
```

NOTA: Habrá que tener cuidado para que las anchuras (las de las distintas columnas y la anchura global de la tabla) no entren en contradicción, pues no será revisado y el resultado puede ser inesperado.

#### ANCHURAS EN HTML

Hay una tercera forma de expresar las anchuras en HTML: en términos proporcionales (expresando con el símbolo asterisco (\*) el número de partes que ocupa la columna del total de partes de la tabla).

Una LONGITUD PROPORCIONAL tiene la forma `"i*"`, donde `"i"` es un entero. Los agentes de usuario, cuando reparten espacio entre los elementos que compiten por ese espacio, adjudican primero las longitudes en píxeles y en porcentajes, y después dividen el espacio sobrante entre las longitudes relativas. Cada longitud relativa recibe una porción del espacio disponible que es proporcional al entero que precede al `"*"`. El valor `"*"` es equivalente a `"1*"`.

Así, si hay disponibles 60 píxeles de espacio después de haber adjudicado el agente de usuario el espacio en píxeles y en porcentajes, y las longitudes relativas que deben asignarse son `"1*"`, `"2*"` y `"3*"`, se asignarán 10 píxeles al `1*`, 20 píxeles al `2*` y 30 píxeles al `3*`.

### ***U.D. 3 Listas, imágenes, objetos y tablas en XHTML 1.0 Y HTML 5***

No obstante, no existe esta alternativa cuando se habla de anchuras de columnas, que sólo puede expresarse en pixels o porcentajes.