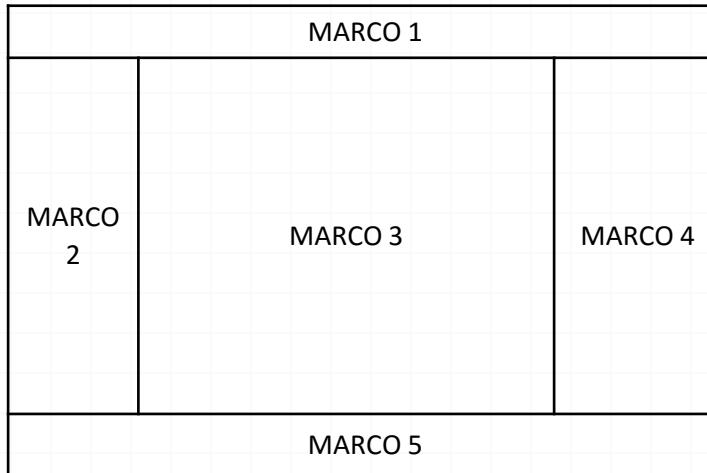




Frames

# Introducción

- Los frames, o marcos, son una herramienta que nos permite dividir la ventana del navegador en subventanas que se pueden tratar de forma independiente.



- Cada uno de los marcos de una página, contiene un documento HTML diferente.
- **NOTA.** Los marcos son un recurso obsoleto. Hoy en día contamos con herramientas mucho mejores para conseguir los mismos resultados.

# Creación de frames

# Creación de marcos

- 0 La estructura de un documento con frames es algo diferente a la de un documento HTML normal. En estos documentos, se sustituye las etiqueta <body> de apertura y cierre por las etiqueta <frameset> de apertura y cierre.

Documento HTML normal	Documento HTML con frames
<pre>&lt;html&gt;   &lt;head&gt; ... &lt;/head&gt;   &lt;body&gt;     ...   &lt;/body&gt; &lt;/html&gt;</pre>	<pre>&lt;html&gt;   &lt;head&gt; ... &lt;/head&gt;   &lt;frameset cols="..." rows="..."&gt;     ...   &lt;/frameset &gt; &lt;/html&gt;</pre>

Documento HTML normal	Documento HTML con frames
<pre>&lt;html&gt;   &lt;head&gt; ... &lt;/head&gt;   &lt;body&gt;     ...   &lt;/body&gt; &lt;/html&gt;</pre>	<pre>&lt;html&gt;   &lt;head&gt; ... &lt;/head&gt;   &lt;frameset cols="..." rows="..."&gt;     ...   &lt;/frameset &gt; &lt;/html&gt;</pre>

Atributo	Valores que puede tomar	Permite cambiar el...
cols	Un número (que puede expresar un valor absoluto en px o relativo si se añade el signo de %). También se puede usar un *	Tamaño de cada una de las subventanas verticales. cols="100,150,*" crea 3 subventanas la primera de las cuales ocupa el 100 píxeles, la segunda 150 píxeles y la tercera el espacio restante de la ventana.
rows	Un número (que puede expresar un valor absoluto en px o relativo si se añade el signo de %). También se puede usar un *	Tamaño de cada una de las subventanas horizontales. rows="50%,25%,25%" crea 3 subventanas la primera de las cuales ocupa el 50% del tamaño total de la ventana y las otras dos el 25% cada una.

0 Dentro de las etiquetas <frameset> de cierre y apertura definiremos los marcos que constituyen nuestra página y para ello utilizaremos la etiqueta <frame>.

```
<frameset rows="filas" cols="columnas">  
  <frame scr="URL de la frame 1" name="nombre frame 1">  
  <frame scr="URL de la frame 2" name="nombre frame 2">  
  ...  
  <frame scr="URL de la frame n" name="nombre frame n">  
</frameset>
```

Atributo	Valores que puede tomar	Efecto
src	url. Puede ser una url en valor absoluto o relativo	Muestra el archivo especificado por la url en el marco indicado
Name	Texto	Permite especificar un nombre por el que nos referiremos a esa subventana



# Manos a la obra

## Código

```
<frameset rows="10%,*,80">
  <frame src="web1.html">
  <frame src="web2.html">
  <frame src="web3.html">
</frameset>
```

## Resultado

Página con Frames

Aquí irá el logotipo de nuestra empresa

Aquí irá toda la información de nuestra página

Aquí irá la información de contacto de nuestra empresa

```
<frameset cols="10%,*,150">
  <frame src="web1.html">
  <frame src="web2.html">
  <frame src="web3.html">
</frameset>
```

Página con Frames verificadas

Aquí irá el logotipo de nuestra empresa

Aquí irá toda la información de nuestra página

Aquí irá la información de contacto de nuestra empresa

Ventajas e  
inconvenientes  
de los frames

# Ventajas del uso de frames

- La navegación por la página podría ser más rápida ya que aunque la primera vez no hubiera diferencia, las cargas sucesivas podrían ser más rápidas ya que algunos marcos se podrían guardar.
- Algunos elementos de la página se mantienen fijos lo que puede facilitar el uso por parte del usuario.
- Esos elementos fijos, nos ayudan a crear una identidad del sitio.

# Inconvenientes

- Algunos navegadores no los soportan.
- Los frames fijos hacen que se pierda espacio de la pantalla. Este problema, se puede volver insostenible cuando el usuario esté visualizando nuestra página desde un dispositivo con pantalla pequeña (como una tablet o un Smartphone).
- El uso de frames puede afectar al posicionamiento de la página.
- El botón “Atrás” de los navegadores en ocasiones no funciona de la manera deseada al utilizar frames.
- En ocasiones, la opción de “Añadir a favoritos” de los navegadores no funciona correctamente.



Soporte de los  
navegadores

# Soporte de los navegadores

- Algunos navegadores no soportan frames. Esto es un inconveniente ya que si el usuario utiliza uno de estos navegadores e intenta abrir una página que contiene frames, el navegador mostrará una página vacía.
- Para solventar este problema tenemos las etiquetas <noframes> y </noframes> que nos permiten definir el código de la página que deberá mostrarse si la página se estuviera cargando en un navegador que no permita el uso de frames.

# Uso de la etiqueta <noframes>

```
<html>
  <head>
    ...
  </head>

  <frameset rows="filas" cols="columnas">
    <frame ...>
    <frame ...>
    ...
    <frame ...>
  </frameset>

  <noframes>
    <body>
      Aquí irá el código html que contendrá la página a
      mostrar para el caso de que el navegador del usuario no
      soporte marcos.
    </body>
  </noframes>
</html>
```

Aquí irá el código html que contendrá la página a mostrar para el caso de que el navegador del usuario no soporte marcos.

0 Algunos diseñadores en el código que va entre las etiquetas <noframes> se limitan a poner un texto similar al siguiente:

La página que está intentando visualizar utiliza frames. Necesita utilizar un navegador que los soporte para poder verla.

0 En otras ocasiones, se crea una página web que no se use marcos y se incluye un enlace entre las etiquetas <noframes> para que nos lleve a esa página.

La página que está intentando visualizar utiliza frames. Pique aquí para ir a una versión soportada por su navegador.

Formatado de  
marcos

0 Aparte de especificar el alto y ancho de los marcos con los atributos cols y rows vistos en apartados anteriores, tanto las etiquetas <frameset> como <frame> permiten el uso de varios argumentos que nos servirán para configurar la apariencia de los frames.

# Atributos de la etiqueta <frameset>

Atributo	Valores que puede tomar	Permite cambiar el...
frameborder	yes / no	Si se muestra o no el borde
framespacing	Un número	Separación entre los marcos
border	Un número (puede ser un valor absoluto en px o relativo si se añade el signo de %)	Grosor del borde
bordercolor	Cualquiera de las formas vistas para expresar colores	Color del borde

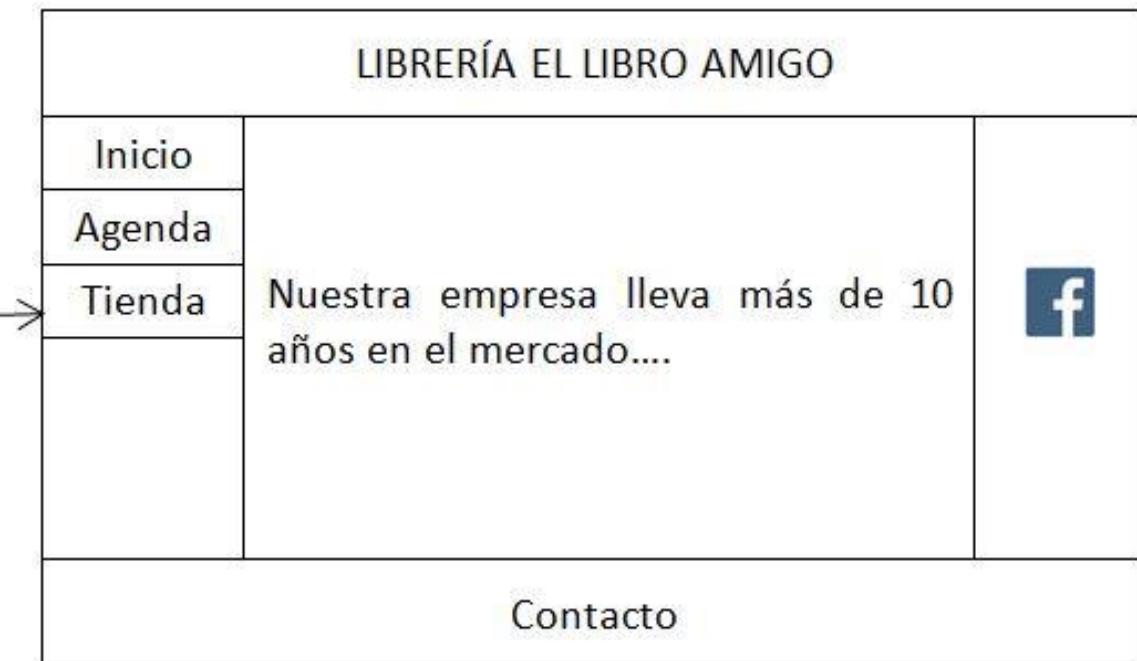
# Atributos de la etiqueta <frame>

Atributo	Valores que puede tomar	Permite cambiar el...
marginwidth	número	Margen horizontal del frame en pixels. Si ponemos 0 el contenido de la página que esté en ese frame se verá pegado completamente al borde del frame.
marginheight	número	Margen vertical del frame expresado en píxeles.
scrolling	yes / no / auto El valor por defecto es auto	Permite indicar si queremos barras de desplazamiento en el marco. La opción auto hace que se muestre la barra sólo cuando el contenido del frame no se vea por completo.
noresize		Si se pone este atributo el frame tendrá un tamaño fijo y no se podrá cambiar. Si no se pone, el tamaño del frame se podrá modificar. Por defecto, los frames se pueden redimensionar.
frameborder	yes / no	Si ponemos no, los bordes del frame se eliminarán.
bordercolor	Cualquiera de las formas vistas para expresar colores	Color del borde

Enlaces entre  
contenidos de  
marcos

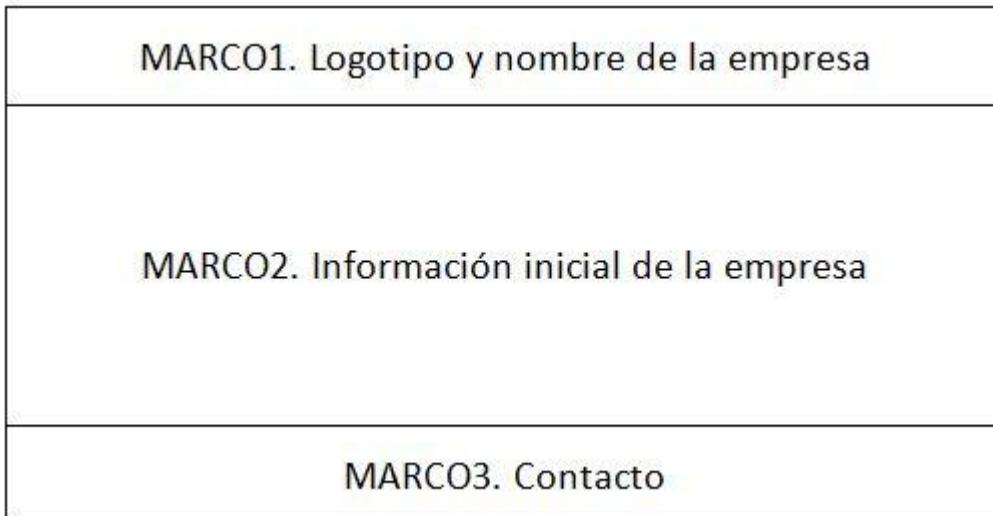
0 Como ya hemos comentado anteriormente, el contenido de los marcos son páginas web y como tales páginas web pueden contener enlaces hacia otras páginas. Cuando el usuario hace clic sobre un enlace, el nuevo contenido se mostrará en el frame donde esté colocado ese enlace.

Si el usuario hace clic sobre el botón Agenda o Tienda, la nueva página se abriría en el marco pequeño de la izquierda donde está el botón, cuando lo ideal sería que se abriera en la zona del centro.



# Abrir un enlace en un marco diferente

1. Dar nombres al marco o marcos donde se deba mostrar el contenido cuando un usuario haga clic sobre los enlaces que haya en cualquier otro frame.  
O Supongamos una estructura dividida en marcos como la siguiente:



En MARCO3 tenemos un texto que pone Contacto con un enlace a los datos de Contacto de la empresa. Cuando el usuario pica sobre el botón Contacto debería de abrirse los datos de contacto en MARCO2

## 0 Definir la estructura en marcos:

```
<frameset rows="10%,*,80">
  <frame src="pagina1.html">
  <frame src="pagina2.html" name="principal">
  <frame src="pagina3.html">
</frameset>
```

## 2. Definir los enlaces para que al abrir el recurso enlazado se dirija hacia el marco adecuado.

- En nuestro ejemplo, pagina3.html contendrá un enlace a una página llamada contacto.html que será la que mostrará los datos de contacto de la empresa, por lo que contendrá un código similar al siguiente:

```
...
<body>
...
<a href="contacto.html"
target="principal">Contacto</a>
...
</body>
```

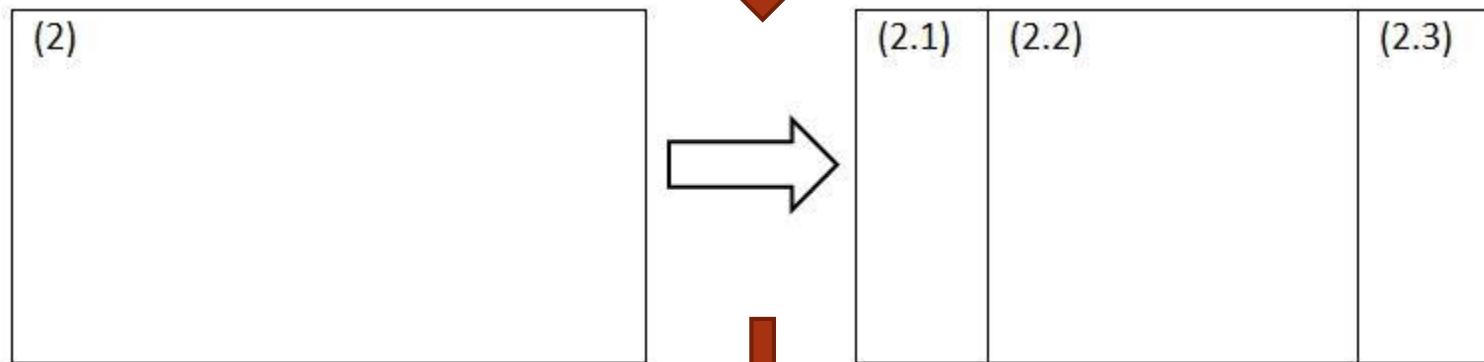
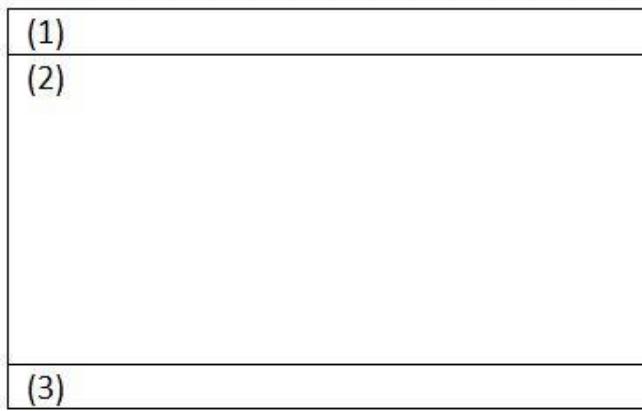
- El atributo target puede tomar varios valores:

- \_blank. El nuevo contenido se abrirá en una nueva ventana.
- \_self. El nuevo contenido se abrirá en el mismo frame donde está situado el enlace. Este es el valor por defecto.
- \_parent. El nuevo contenido se abrirá sobre el padre. Si no hubiera, se abriría sobre el marco actual.
- \_top. El contenido se carga a pantalla completa, eliminando todos los frames que tengamos definidos.



Marcos anidados

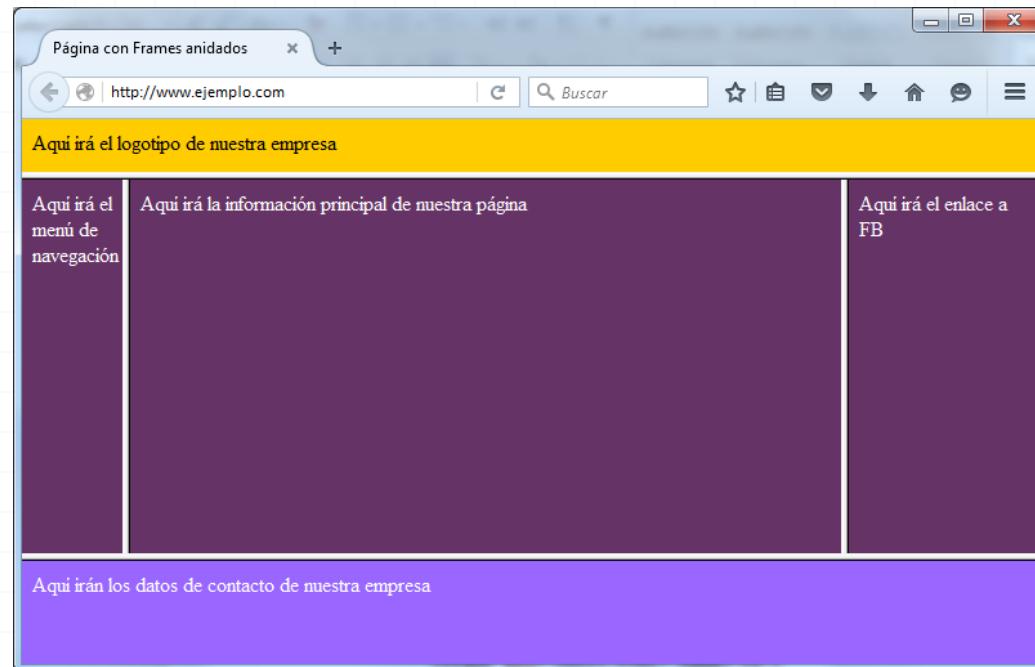
- En los ejemplos vistos a lo largo de este apartado, hemos visto cómo definir marcos en filas o en columnas.
- Sin embargo, lo más habitual en una página web es que se mezclen marcos en filas y marcos en columnas. Para conseguir esto, debemos anidar etiquetas <frameset>.
- Para anidar etiquetas <frameset>, debemos empezar por la estructura más externa y seguir hacia las más internas.





# Manos a la obra

```
<frameset rows="10%, *, 80">
    <frame src="web1.html" name="encabezado">
    <frameset cols="10%, *, 150">
        <frame src="web21.html" name="menu">
        <frame src="web22.html" name="principal">
        <frame src="web23.html" name="FB">
    </frameset>
    <frame src="web3.html" name="pie">
</frameset>
```





Marcos  
incrustados  
(iframes)

# Definir un iframe

- Un iframe (inline frame) sirve para crear una ventana secundaria dentro de la página actual. Esto nos permite mostrar una página web dentro de la página actual.
- La sintaxis para definir un iframe es:

```
<iframe src="url"> texto </iframe>
```

- *src*: especificará la url de la página que se quiere mostrar en el marco.
- *texto*: Si el navegador no soporta iframes, se mostrará el contenido que se halle entre las etiquetas `<iframe>` de apertura y cierre.

```
<iframe src="http://www.google.com" >
```

Su navegador no permite el uso de iframes

```
</iframe>
```

# Establecer la alineación del iframe

- Para definir la alineación del iframe podemos utilizar el atributo align.

```
<iframe src="url" align="alineacion"> </iframe>
```

- alineacion. Puede tomar los valores "left" o "right" para ubicar el frame en la parte izquierda o derecha de la ventana del navegador.



# Manos a la obra

```
<body>
```

```
  <p>Línea 1 de la página principal</p>
```

```
  <iframe>
```

```
    src="http://www.google.com"
```

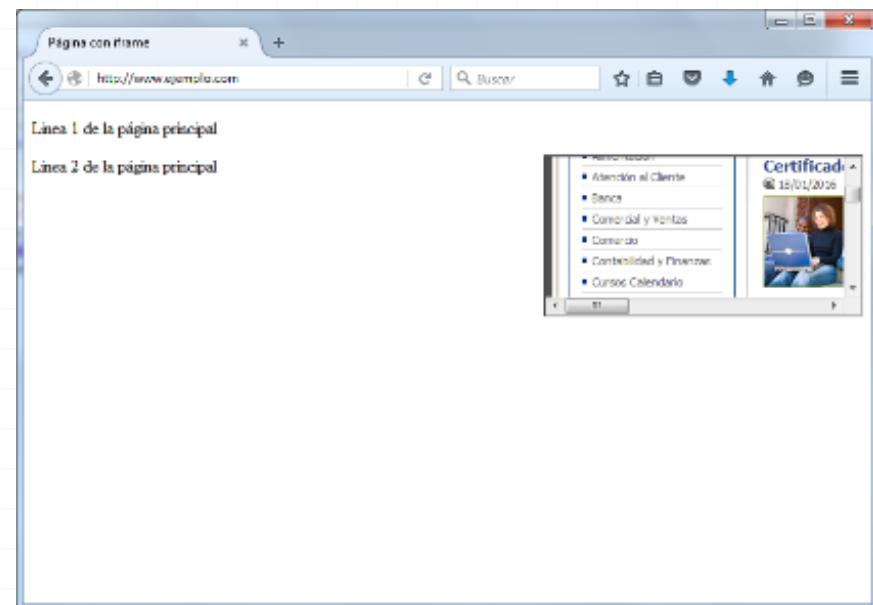
```
    align="right">
```

```
      Su navegador no permite el uso de iframes
```

```
  </iframe>
```

```
  <p>Línea 2 de la página principal</p>
```

```
</body>
```



# Definir el tamaño del iframe

- Para definir el tamaño que va a ocupar el iframe, utilizaremos los atributos height y width.

```
<iframe src="url" width="value" height="value"></iframe>
```

- value. Por defecto, se especifica en píxeles pero también se puede indicar en tanto por ciento, añadiendo el signo de % a continuación del número.



# Manos a la obra

<body>

<p>Línea 1 de la página principal</p>

<iframe>

src="http://www.google.com"

align="right"

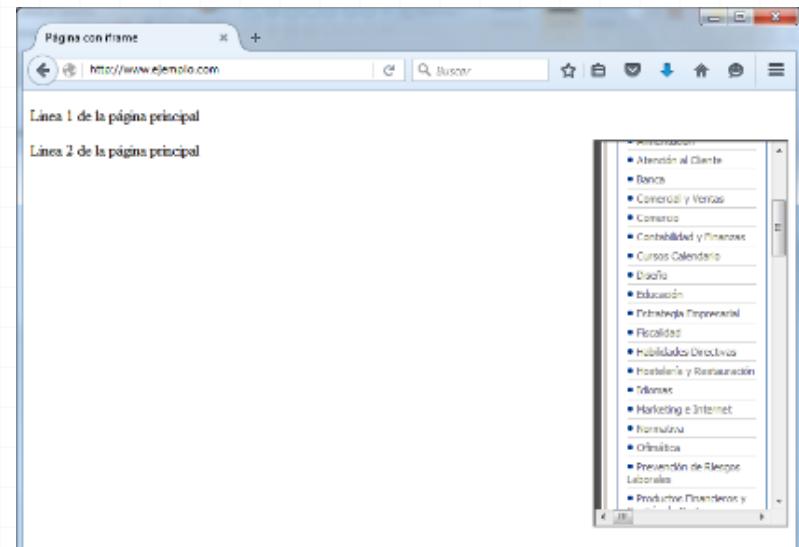
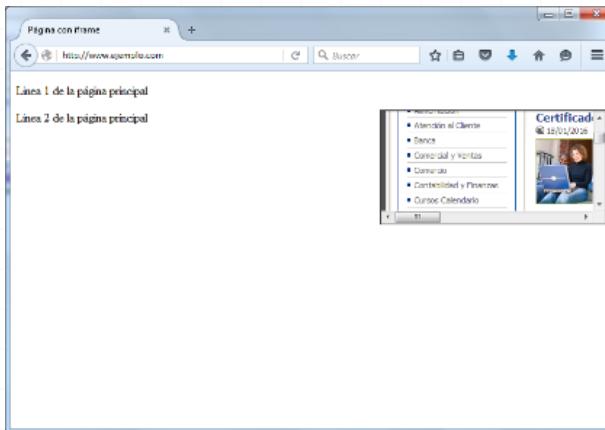
width="200" height="400">

Su navegador no permite el uso de iframes

</iframe>

<p>Línea 2 de la página principal</p>

</body>



# Utilizar un iframe como destino de un link

- Podemos hacer que al hacer clic sobre un enlace, la nueva ventana se abra en el iframe. Para ello, cuando creamos el iframe, debemos identificarlo con un nombre mediante el atributo name.

```
<iframe  
    src="url"  
    width="value"  
    height="value"  
    name="nombre"></iframe>
```

- nombre. Es el nombre que queremos asignar al iframe y que luego utilizaremos como argumento en el atributo target cuando definamos el enlace.



# Manos a la obra

## Código

```
<body>

<h1>Abrir un enlace en un iframe</h1>

<iframe align="right" width="200" height="400" name="marco">
</iframe>
<p> Pincha <a href="https://www.google.com" target="marco"> aquí</a>
para ir a la página</p>
</body>
```

## Resultado



Al hacer clic sobre la palabra aquí, veremos la página en el iframe.



# Quitar el borde a un iframe

- Los iframes aparecen rodeados por un borde (como puede observarse en la imagen de la derecha).
- Podemos indicar si queremos que se visualice este borde o no se visualice con el argumento frameborder.

```
<iframe src="url" frameborder="0"></iframe>
```

- El argumento puede tomar dos valores:

- 0. El borde no se debe mostrar
  - 1. Se debe mostrar el borde

Por defecto, el valor de frameborder es 1



- 0 También podemos quitar el borde utilizando el atributo style y utilizando la propiedad border de CSS.

```
<iframe src="url" style="border:none"></iframe>
```

- 0 Con esta misma propiedad border podemos cambiar el tamaño, estilo y color del borde del iframe.

```
<iframe src="url" style="border:5px dotted red"></iframe>
```

# Insertar un video de youtube

- Quizás una de las aplicaciones más utilizadas de los iframes es insertar un video de youtube en nuestra página.
- Para insertar un video de youtube en nuestra página:
  - Abrir youtube y buscar el video que deseemos.

2. Hacer clic sobre el botón Compartir

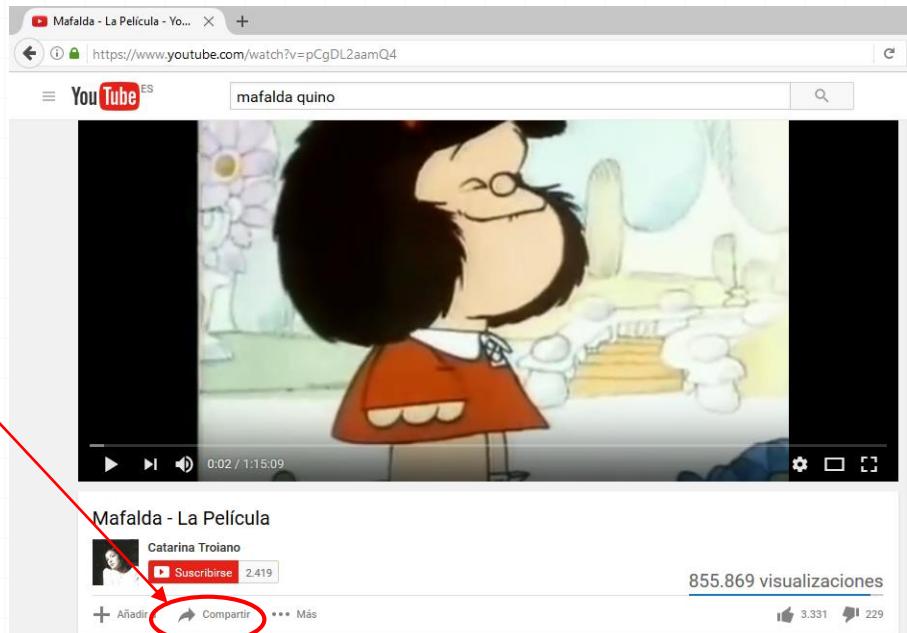


3. En la lista que se abre hacer clic sobre insertar

Compartir **Insertar** Enviar por correo

```
<iframe width="560" height="315" src="https://www.youtube.com/embed/pCgDL2amQ4" frameborder="0" allowfullscreen></iframe>
```

MOSTRAR MÁS

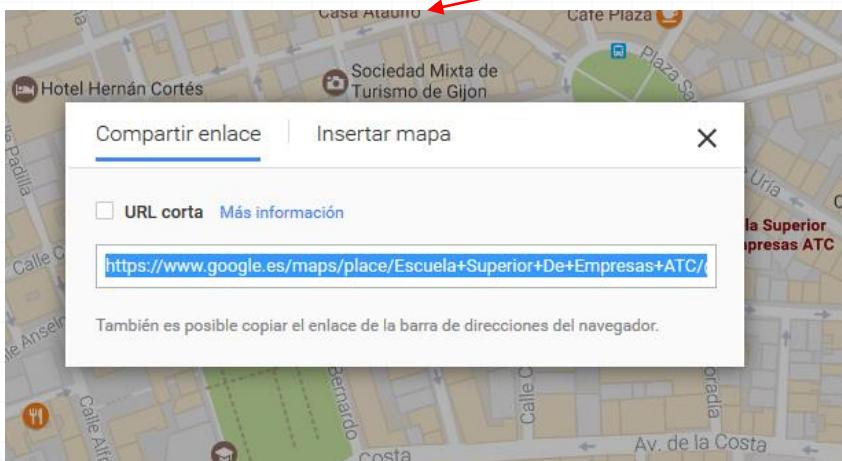
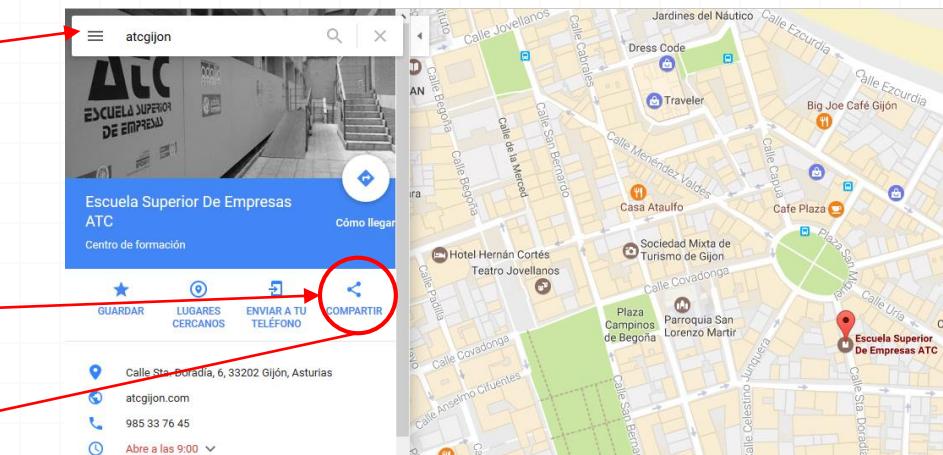


Copiar en enlace que aparece en el cuadro de texto en nuestra página en el lugar donde queramos que aparezca el video

# Insertar un mapa de google maps

0 Otro uso de los iframes es cuando queremos insertar un mapa tomado de google maps en nuestra página web para indicar una localización

Abrimos google maps y buscamos la localización que queremos incluir en nuestro mapa y le damos a buscar. Cuando google maps nos muestre la localización en el mapa, haremos clic sobre el botón Compartir



En la ventana que se abre, copiar el enlace que aparece (Podemos marcar URL corta para utilizar un acortador de URL)  
Pegar el enlace en nuestra página en el lugar donde queramos que aparezca el mapa

# Formularios

# Descripción General

- Un formulario es un elemento que nos permitirá recoger datos introducidos por el usuario.
- A través de un formulario podemos crear cuestionarios, encuestas para el usuario, apartados de comentarios, etc. a través de los cuales podremos interactuar con el usuario.
- Para definir un formulario utilizaremos la etiqueta `<form>` que tendrá la siguiente sintaxis:

```
<form>
```

...

*Elementos del formulario*

...

```
</form>
```

- *Elementos del formulario.* Son aquellos elementos que nos permitirán recoger la información del usuario. Pueden ser cuadros de diálogo, listas con opciones seleccionables (excluyentes o no excluyentes), botones, etc.
- A la etiqueta `<form>` le podemos añadir algunos parámetros para determinar cómo va a ser su funcionamiento. Los principales parámetros que podemos incluir serán:

# Elementos de un formulario

# Elemento <input>

- Es el elemento más importante de un formulario. Su sintaxis básica es la siguiente:

```
<input type="tipo" name="nombre_control">
```

- type. Es el tipo de control. Según el tipo seleccionado, el usuario podrá introducir un tipo u otro de información.
- name. El atributo name es muy importante ya que el nombre que ponemos en este campo es el que utilizará la aplicación del servidor para obtener el valor introducido en este campo del formulario. Si un campo no incluye este atributo, sus datos no se enviarán al servidor.

# <input type="text">

- Es quizás el elemento más utilizado. Permite crear un cuadro de texto vacío donde el usuario puede introducir cualquier cadena de caracteres.
- Su sintaxis es la siguiente:

```
<input type="text" name="nombre_control" value="valor"  
       size="tamaño" maxlength="valor">
```

- **value**. Este atributo nos permitirá establecer un valor inicial para el cuadro de texto. Si no se quiere asignar ningún valor inicial al cuadro de texto podemos no incluir el atributo value o poner el valor vacío mediante la sentencia value="".
- **size**. Nos permite establecer el tamaño en caracteres con el que queremos que el navegador muestre el cuadro de texto que estamos definiendo.
- **maxlength**. Nos permite establecer el número máximo de caracteres que el usuario puede introducir en un cuadro de texto.



# Manos a la obra

```
<form>
  <p>
    Nombre:
      <input type="text" name="nombre"><br>
  </p>
  <p>
    Apellidos:
      <input type="text" name="apellidos">
  </p>
</form>
```

A screenshot of a web browser window titled "Página con Formulario". The address bar shows the URL "http://www.ejemplo.com". The main content area displays a form titled "Ficha personal". The form has two input fields: one for "Nombre" and one for "Apellidos", both represented by empty text input boxes.

# <input type="password">

- Permite definir un cuadro para contraseñas.
- Este elemento admite los mismos atributos que el elemento text.
- La única diferencia con el cuadro de texto visto en el punto anterior, es que los datos que el usuario introduce en un cuadro de contraseña no se ven por pantalla, en su lugar, el navegador los sustituye por asteriscos (\*) o puntos (•).



# Manos a la obra

```
<form>
```

```
    <p>
```

Usuario:

```
            <input type="text" name="usuario">
```

```
    </p>
```

```
    <p>
```

Contraseña:

```
            <input type="password" name="psw">
```

```
    </p>
```

```
</form>
```

Observa que al introducir la contraseña los caracteres que el usuario teclea no son legibles (se sustituyen por \* o •)



Página con Formulario

http://www.ejemplo.com

Datos de acceso

Usuario: pepito

Contraseña: \*\*\*\*\*

# <input type="checkbox">

○ El checkbox es una opción seleccionable que puede presentar dos estados: activado o desactivado.

○ El formato de este elemento será:

```
<input type="checkbox" name="nombre_control" value="valor">
```

○ El valor del atributo `value`, junto con el valor del atributo `name`, es la información que llega al servidor cuando el usuario envía el formulario.

○ Con la instrucción anterior, crearemos un checkbox no activado, es decir que presentará el cuadro sin seleccionar. Si deseáramos crear un cuadro seleccionable pero activado, entonces añadiremos el argumento `checked` dentro de la sentencia `<input type>`.

```
<input type="checkbox" name="nombre_control" value="valor" checked>
```



# Manos a la obra

<form>

    Marcar todos los estudios terminados que se posean <br/>  
    <input type="checkbox" name="primarios" value="ESO"> ESO  
    <input type="checkbox" name="secundarios"  
        value="Bachiller" checked> Bachiller o FP2  
    <input name="universitarios" type="checkbox"  
        value="Superiores"> Licenciatura o Diplomatura

</form>

Observa que por defecto aparece marcado Bachiller porque es la que tiene el argumento checked.

Para que al lado de cada cuadrado aparezca un texto, hay que escribirlo después del cierre de la sentencia input



Página con Formulario

http://www.ejemplo.com

Datos académicos

Marcar todos los estudios terminados que se posean

ESO  Bachiller o FP2  Licenciatura o Diplomatura

# <input type="radio">

- Los elementos de tipo radio son similares a los del tipo checkbox con la diferencia de que el carácter que muestran es un círculo y que cuando tenemos una lista con varias opciones éstas son excluyentes entre sí.



# Manos a la obra

<form>

```
    <input type="radio" name="sexo" value="hombre" checked>  
        Hombre<br/>
```

```
    <input type="radio" name="sexo" value="mujer"> Mujer
```

</form>

Observa que por defecto aparece marcado hombre porque es la que tiene el argumento checked.

De todas las opciones mostradas, solo podemos elegir una. Al marcar Mujer, se desmarcaría Hombre

Página con Formulario

http://www.ejemplo.com

Datos personales

Sexo

Hombre  
 Mujer

# <input type="submit">

- Generalmente, todos los formularios incluyen un botón para enviar los datos al servidor.
- Para generar este botón definiremos un elemento de tipo submit.



# Manos a la obra

```
<form>
  <p>
    Usuario:
    <input type="text" name="usuario">
  </p>
  <p>
    Contraseña:
    <input type="password" name="psw">
  </p>
  <input type="submit" value="Enviar">
</form>
```

Observa que el texto que se muestra en el botón es el que se ha puesto dentro del atributo value cuando se ha definido el tipo submit. Si se omite el atributo value, el navegador mostrará un texto dentro del botón que pone Enviar consulta

The screenshot shows a web browser window titled "Página con Formulario". The address bar displays "http://www.ejemplo.com". The main content area contains a form with the title "Datos de acceso". It has two input fields: one for "Usuario" and one for "Contraseña". Below the fields is a "Enviar" button. An arrow points from the text in the previous paragraph to this "Enviar" button.

# <input type="reset">

- Este tipo de campo nos permite crear un botón que borre todos los datos introducidos por el usuario en el formulario devolviendo el formulario a su estado inicial sin ningún campo cubierto.

```
<input type="reset" name="limpiar" value="Limpiar datos" />
```

- Hace unos años, el uso de este botón era muy común en los formularios, pero en la actualidad ya no es tan habitual su empleo.

# <input type="button">

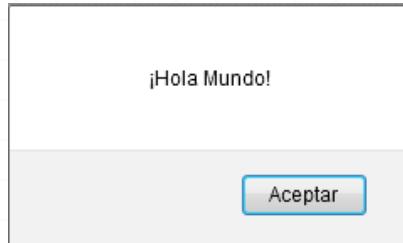
- El tipo button nos permite definir un botón.
- En el caso de definir un botón, suele ser necesario incluir algún atributo del tipo onclick al que se asocie una acción que deberá llevarse a cabo cuando el usuario haga clic sobre el botón.

```
<input type="button" onclick="alert('¡Hola Mundo!')" value="Pica aquí">
```

- La sentencia anterior insertará un botón en nuestro formulario como el que mostramos a continuación:

Pica aquí

- Si el usuario hace clic sobre el botón, el sistema ejecutará la sentencia definida dentro del parámetro onclick. En este caso, mostrará una ventana emergente al usuario con el mensaje ¡Hola Mundo!



# <input type="file">

- Este tipo de input nos permite definir un campo para que el usuario pueda adjuntar un archivo.

```
<input type="file" name="foto">
```

- Al utilizar este tipo de input, el navegador mostrará un cuadro de texto con un botón al lado que permitirá al usuario navegar por los directorios y archivos de su equipo para seleccionar el archivo que quiere adjuntar.

Si lo deseas puedes añadir tu foto

Examinar...

No se ha seleccionado ningún archivo.

- En principio no existe ninguna limitación sobre el tipo o tamaño del archivo a adjuntar pero, generalmente, los servidores añaden restricciones sobre los archivos que aceptan por motivos de seguridad.

# <input type="number">

- Este tipo de input es nuevo en HTML5 y es soportado por las últimas versiones de los principales navegadores (Internet explorer, Firefox, Chrome, Safari y Opera)
- Se usa para campos en los que el usuario debe introducir un valor numérico, de manera que no acepta otro valor que no sea un número. Además, se pueden establecer restricciones en los valores aceptados.

```
<input type="number" name="cantidad" min="2"  
       max="10" step="2" value="4">
```



Para introducir el valor en el campo de tipo number podemos, escribirlo directamente en el cuadro o utilizar las flechas de incremento y decrecimiento.

**0 min.** Es el valor más pequeño que vamos a poder introducir en este campo (en nuestro ejemplo, 2). Si intentamos introducir un valor más pequeño el navegador nos marcará el cuadro en rojo y si ponemos el cursor sobre él nos mostrará un mensaje indicándonos que debemos teclear un valor superior al mínimo.

Cantidad:

Seleccione un valor que no sea menor de 2.

**0 max.** Igual que el anterior pero nos permite definir el valor máximo que nuestro campo aceptará.

Cantidad:

Seleccione un valor que no sea mayor de 10.

**0 step.** Especifica el rango de valores admitidos. En el ejemplo, hemos puesto que el valor mínimo es 2 y que step=2 lo que implica que los valores admitidos serán 2, 4, 6, 8,...

Cantidad:

Seleccione un valor válido. Los dos valores válidos más cercanos son 6 y 8.

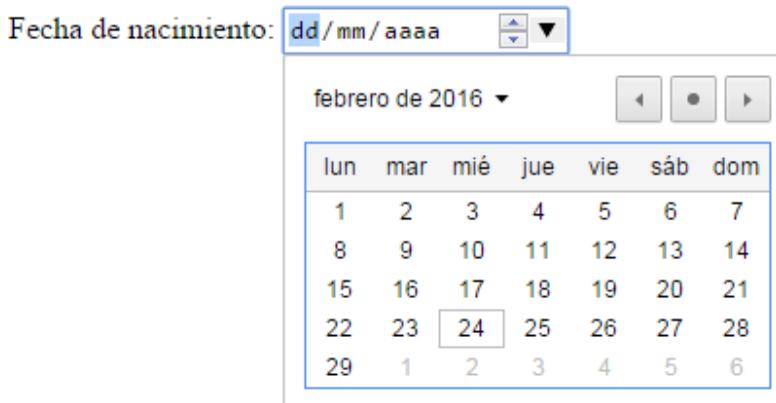
**0 value.** Especifica el valor por defecto del campo. En nuestro ejemplo, hemos puesto 4 por eso cuando vemos el campo de formulario aparece el valor 4.

# <input type="date">

- Este tipo de input es nuevo en HTML5. Internet explorer y Firefox no soportan este tipo de input.
- Se utiliza para campos que contienen fechas.

```
<input type="date" name="fechanac">
```

- Dependiendo del navegador, puede mostrarse un selector de fecha al lado del campo de tipo fecha.



- Al igual que en el tipo número, este tipo de datos permite especificar un valor mínimo y máximo para la fecha a introducir, un valor por defecto, etc.

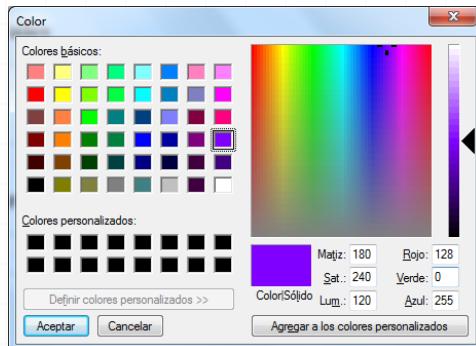
# <input type="color">

- Este tipo de input es nuevo en HTML5. Internet explorer y Safari no soportan este tipo de input.
- Se utiliza para los campos que deben contener un color. Dependiendo del navegador, puede aparecer un selector de color al lado del campo.

```
<input type="color" name="cfav">
```

Dinos tu color preferido:

- Al picar sobre el selector de color se mostrará una ventana para poder elegir el color:



# <input type="email">

- Este tipo de input es nuevo en HTML5 y es soportado por las últimas versiones de los principales navegadores excepto Safari.

```
<input type="email" name="email">
```

- Cuando el usuario introduce su correo electrónico en un campo de tipo email, algunos navegadores validan la información introducida y si no contiene una dirección válida de correo (nombre, signo @ y nombre de dominio) no permitirá la introducción del dato.

Correo de contacto

Introduzca una dirección de correo.

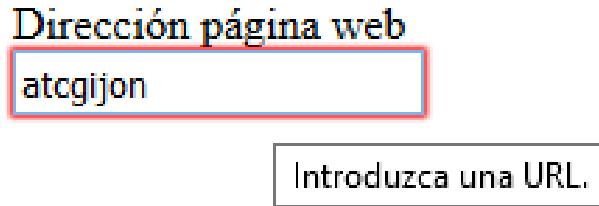
- La mayoría de los smartphones, reconocen el tipo email y cuando el usuario se sitúa en un campo de este tipo, agregan automáticamente ".com" al teclado para facilitar la introducción del correo.

# <input type="url">

- Este tipo de input es nuevo en HTML5 y es soportado por las últimas versiones de los principales navegadores excepto Safari.

```
<input type="url" name="web">
```

- Este tipo de input se utiliza para que el usuario, al llenar el formulario, pueda introducir una url en un formato válido. Una vez introducida algunos navegadores validarán de forma automática la dirección escrita por el usuario y si ésta no tuviera un formato correcto emitirán un mensaje de error:



- La mayoría de los smartphones, reconocen el tipo url y cuando el usuario se sitúa en un campo de este tipo, agregan automáticamente ".com" al teclado para facilitar la introducción del final de la url.

# Elemento <textarea>

- Este elemento es similar al tipo text con la diferencia de que el usuario puede escribir textos más extensos que pueden contar con varias filas y columnas.

```
<textarea name="nombre"  
         cols="valor"  
         rows="valor">  
</textarea>
```

- cols. Número de caracteres que se pueden escribir en cada fila como máximo
- rows. Filas de texto que serán visibles



# Manos a la obra

```
<form>
```

```
    <p>
```

```
        Déjanos tu comentario:<br/>
```

```
        <textarea name="comentario" cols="50" rows="5">
```

```
    </textarea>
```

```
    </p>
```

```
</form>
```

Déjanos tu comentario:

Enviar

# Elemento <select>

- El elemento <select> nos permite crear listas seleccionables, es decir, una lista de opciones de entre las que el usuario puede elegir una.
- Cuando definamos una lista de opciones seleccionables, necesitaremos dos etiquetas:
  - Por un lado necesitaremos la etiqueta <select> de apertura que indique el comienzo de la lista y la etiqueta </select> de cierre que indique el final de la misma.
  - Por otro lado, necesitaremos las etiquetas <option> de apertura y </option> de cierre, para identificar cada una de las opciones de la lista.

## 0 Etiqueta <select>

```
<select size="tamaño" [multiple]>  
    Lista de opciones  
</select>
```

- 0 **size**. Es el número de filas de la lista que se muestran. Por defecto sólo se muestra una (lista desplegable)
- 0 **multiple**. Si se incluye esta opción el usuario podrá seleccionar más de una opción de la lista. Por defecto, sólo se puede elegir una opción de la lista.

## 0 Etiqueta <option>

```
<option value="valor" [selected]>  
    Texto mostrado al usuario  
</option>
```

- 0 **value**. Es el valor que se envía al servidor cuando el usuario elige esta opción.
- 0 **Selected**. Si se incluye este argumento, indica la opción por defecto.

Selecciona el área en el que estás interesado/a:

Contabilidad ▾



# Manos a la obra

<form>

Selecciona el área en el que estás interesado/a:

<select name="area">

<option value="conta">Contabilidad</option>

<option value="lab">Laboral</option>

<option value="inf">Ofimática</option>

</select>

</form>

Selecciona el área en el que estás interesado/a:

Contabilidad ▾

Vemos que en este caso, se muestra una lista desplegable. Como no se ha puesto el argumento selected la opción que aparece visible es la primera



# Manos a la obra

```
<form>
```

Selecciona el área en el que estás interesado/a:

```
<select id="area" name="area">  
  <option value="conta">Contabilidad</option>  
  <option value="lab">Laboral</option>  
  <option value="inf">Ofimática</option>  
  <option value="sel" selected>-Selecciona-</option>  
</select>
```

```
</form>
```

Selecciona el área en el que estás interesado/a:

-Selecciona- ▾

Al agregar el atributo selected a la última opción (-Selecciona-) es la que se muestra al usuario, a pesar de que en este caso esté la última



# Manos a la obra

```
<form>
```

Selecciona el área en el que estás interesado/a:

```
<select id="area" name="area" size="5">
  <option value="conta">Contabilidad</option>
  <option value="lab">Laboral</option>
  <option value="inf">Ofimática</option>
</select>
</form>
```

Selecciona el área en el que estás interesado/a:

A screenshot of a dropdown menu window. The menu has a light gray background and a thin gray border. Inside, there are three items listed vertically: "Contabilidad" in red, "Laboral" in blue, and "Ofimática" in green. Each item is preceded by a small square checkbox. At the bottom right of the menu, there are two small arrows pointing upwards and downwards, indicating scrollable content.

- Contabilidad
- Laboral
- Ofimática

Al poner size="5" se muestra una lista con un tamaño para mostrar 5 opciones

# <optgroup>

- 0 Esta etiqueta permite agrupar las opciones de una lista cuando ésta es muy extensa. La sintaxis de esta etiqueta es la siguiente:

```
<optgroup label="Titulo_seccion">  
    Lista de opciones  
</optgroup>
```



# Manos a la obra

Selecciona la materia:<br>

```
<select id="materia" name="materia">
    <option selected>---Selecciona una opción---</option>
    <optgroup label="Área Contable">
        <option value="conta1">Contabilidad Básica</option>
        <option value="conta2">Contabilidad Avanzada</option>
        <option value="conta3">Contabilidad Financiera</option>
    </optgroup>
    <optgroup label="Área Laboral">
        <option value="lab1">Nóminas</option>
        <option value="lab2">Seguros Sociales</option>
        <option value="lab3">IRPF</option>
    </optgroup>
    <optgroup label="Área Ofimática">
        <option value="ofi1">Word</option>
        <option value="ofi2">Excel</option>
        <option value="ofi3">Powerpoint</option>
    </optgroup>
</select>
```

Selecciona la materia:

---Selecciona una opción---



Selecciona la materia:

---Selecciona una opción---

---Selecciona una opción---

**Área Contable**

- Contabilidad Básica
- Contabilidad Avanzada
- Contabilidad Financiera

**Área Laboral**

- Nóminas
- Seguros Sociales
- IRPF

**Área Ofimática**

- Word
- Excel
- Powerpoint

# Procesamiento de formularios

# Envío de datos

- Una vez rellenados los campos de un formulario por parte de un usuario, es necesario que los datos lleguen al responsable de la página para que pueda recoger los datos que el usuario ha introducido.
- Para ello, debemos añadir algunos atributos a la etiqueta `<form>`:

```
<form action="url" method="post | get">
```

- `action`. Este atributo permite definir la acción que deberá realizarse cuando se envía el formulario. Generalmente, indicará la URL donde se encuentra ubicada la aplicación del servidor encargada de procesar los datos introducidos por el usuario.
- `method`. Este atributo permite definir la forma en la que los datos se envían desde el formulario al servidor. Este atributo suele tomar dos valores: `get` y `post`.

`method="get"`

`method="post".`

# get y post

- A la hora de elegir cuál de los dos métodos debemos utilizar deberemos tener en cuenta algunas consideraciones:
  - get admite como máximo el envío de unos 500 bytes de información
  - get no permite el envío de archivos adjuntos con el formulario.
  - Los datos enviados mediante get se ven en la barra de direcciones del navegador (se añaden al final de la URL de la página) mientras que con post no.
- Como "truco" para elegir el método apropiado podemos considerar que usaremos get en aquellos formularios que no modifican la información (por ejemplo en un formulario de búsqueda) y deberíamos usar post en aquellos formularios que modifican la información original (formularios que permitan insertar, modificar o borrar alguna información).

# Ejemplo

```
<form action="alta_usuario.php" method="post">
```

- 0 Además, debemos de tener en cuenta que si se ha incluido algún campo para adjuntar archivos, necesitaremos añadir el atributo enctype en la etiqueta <form>. Por lo que en los formularios que permiten adjuntar archivos, la etiqueta <form> quedaría como sigue:

```
<form action="alta_usuario.php" method="post"  
enctype="multipart/form-data"
```

# Formateo de formularios

# Etiqueta <fieldset>

- Podemos agrupar de manera lógica los campos de un formulario mediante la etiqueta <fieldset>.
- Un formulario, puede contener tantos grupos como se consideren necesarios. A la hora de crear estos grupos podemos seguir diferentes criterios de agrupación.
- Cuando creamos un grupo de campos con la etiqueta <fieldset> podemos añadir un título para el grupo con la etiqueta <legend>.

```
<fieldset>
    <legend>Título</legend>
        Lista de campos que forman parte del grupo
    </legend>
</fieldset>
```

- El navegador muestra por defecto un borde para remarcar los campos que pertenecen al mismo grupo.



# Manos a la obra

```
<form>
  <fieldset>
    <legend>Datos personales</legend>
    Nombre:
    <input type="text" name="nombre" >
    Apellidos:
    <input type="text" name="apellidos" >
    Edad:
    <input type="number" name="edad" min="18" max="100">
    Sexo <br/>
    <input type="radio" name="sexo"
           value="hombre" checked> Hombre<br/>
    <input type="radio" name="sexo" value="mujer" /> Mujer
  </fieldset>
  ...
</form>
```

A screenshot of a web browser window titled "Página con Formulario". The address bar shows "http://www.ejemplo.com". The page content is a form titled "Datos personales" containing the following fields:

- Nombre:
- Apellidos:
- Edad:
- Sexo:
  - Hombre
  - Mujer

# Etiqueta <label>

- En todo el código HTML que hemos estado utilizando para crear los campos de los formularios utilizados en los ejemplos anteriores, hemos visto que para agregar un texto que acompañara a los campos, lo hemos hecho sin más mediante texto normal. No existe una etiqueta que nos permita establecer un título o texto asociado a los controles que se incluyen en los formularios.
- Aunque no exista una etiqueta específica que nos permita establecer una relación entre el campo y el texto que lo acompaña, HTML cuenta con la etiqueta <label> que permite definir un título para el campo.
- Su sintaxis es:

```
<label for="id_campo">Texto</label>
```

```
<input type="tipo" id="id_campo" name="nombre_campo">
```

0 Visualmente no apreciamos diferencia entre utilizar la etiqueta <label> o simplemente teclear un texto al lado del campo. La ventaja de utilizar <label> es que esta etiqueta hace que mejora la accesibilidad de nuestra web. Además, si hacemos clic sobre el texto definido mediante un <label>, el puntero del ratón se posicionará automáticamente sobre el campo asociado a esa etiqueta.



Otras opciones

# Atajos de teclado

0 Los atajos de teclado sirven para definir una combinación de teclas mediante la cual el usuario podrá acceder a un elemento en cuestión utilizando el teclado en lugar del ratón. El uso de estos atajos del teclado permite mejorar la accesibilidad de nuestra página.

0 Para crear un atajo de teclado utilizaremos la etiqueta `<accesskey>`:

```
<form action="alta.php" method="post">
    <label for="usuario" accesskey="u">Nombre Usuario</label>
    <input type="text" value="usuario">
</form>
```

# Orden de edición

- Hay usuarios que, para desplazarse a través de los campos de un formulario, en lugar del ratón prefieren utilizar el teclado usando, generalmente, la tecla "*Tabulador*". En este caso, lo habitual es que cada vez que el usuario pulsa la tecla "*Tabulador*" el navegador desplace el foco al siguiente campo del formulario.
- Pero, habrá veces en que querremos alterar este orden y que, si el usuario utiliza el teclado para desplazarse por los campos del formulario, lo haga en un orden que no sea el de ubicación de los campos.
  - En este caso será necesario definir el orden de tabulación.

# Formas para definir el orden de tabulación

- Cuando un usuario abre el formulario, generalmente debe hacer clic sobre el primer campo (o pulsar la tecla “*Intro*”) para poder empezar a rellenarlo. Podemos utilizar el atributo `autofocus` dentro del elemento `<input>` para indicar que se debe dirigir el foco hacia ese campo cuando se cargue el formulario.

Correo electrónico: `<input type="email" name="correo" autofocus>`

- Este atributo es incorporado a partir de HTML5.
- Utilizar el atributo `tabindex`. Este atributo permite especificar la posición de un elemento dentro del orden de tabulación del formulario, es decir, el orden en el que el navegador saltará de un campo a otro del formulario cuando el usuario utilice la tecla “*Tabulador*” para cambiar de campo.

## Ejemplo

```
<form>  
  <input tabindex="3" type="text" name="campo_a">  
  <input tabindex="1" type="text" name="campo_b">  
  <input tabindex="2" type="text" name="campo_c">  
  <input type="text" name="campo_d" autofocus>  
</form>
```

Cuando el navegador cargue el formulario, el usuario verá el cursos parpadeando sobre el campo\_d que es el que tiene definido el atributo autofocus.

A partir de aquí, cuando el usuario pulse la tecla “*Tabulador*” cambiará al campo\_b, luego al campo\_c y por último al campo\_a ya que tienen definido el orden 1, 2 y 3 respectivamente).

# Vídeo en HTML5

# Vídeos en nuestras páginas web

- Una de las grandes novedades de HTML5 es precisamente la inclusión de video en nuestras páginas web ya que nos permite prescindir de tener que utilizar ciertos programas de terceros (como por ejemplo, flash).
- Los navegadores ya soportan algunos formatos de video y además proporcionan características a la interfaz de los vídeos.
- Lo primero que necesitamos conocer son los formatos de video que soportan los navegadores y que a día de hoy son:
  - MP4. Está sujeto a licencia comercial
  - Ogg. Es un formato libre.
- Existen conversores gratuitos (tanto online como descargables)que te permiten convertir cualquier formato (como por ejemplo .avi) a ogg o mp4. Incluso, te permitirán convertir uno de los formatos al otro, es decir, ogg a mp4 y viceversa.

# Etiqueta para incluir video

- La etiqueta html5 que nos permite incluir videos en nuestras páginas web es la etiqueta <video> de apertura y </video> de cierre.
- Esta etiqueta cuenta con una serie de atributos de los cuales los más importantes son:
  - **src**. Permite especificar dónde se encuentra el vídeo
  - **controls**. Permite especificar si nuestro video va a contar con la típica botonera con los botones de play, pause, etc.
  - **autoplay**. Permite especificar al navegador si nuestro video va a reproducirse automáticamente al cargar la página o no
  - **loop**. Permite especificar al navegador si al finalizar la reproducción de nuestro video, éste comenzará de nuevo. Es decir, si se reproducirá indefinidamente.
  - **poster**. Permite incluir una imagen jpg a modo de portada o presentación que aparecerá antes de que empiece la reproducción del video.
  - **preload**. Permite especificar algunas características importantes antes de cargarlo (por ejemplo, cuántos fotogramas tiene el video, duración, etc.)



# Manos a la obra

- Vamos a hacer una página web sencilla donde vamos a incluir un video.
- Vamos a empezar creando la estructura de carpetas.
  - Crear una carpeta para el sitio.
  - Dentro de esta carpeta crear una carpeta que llamaremos video donde ubicaremos el video.
  - Dentro de esta carpeta video, crear otra carpeta que llamaremos portada y donde incluiremos la imagen que más adelante vamos a utilizar como portada del video.
  - Crearemos la estructura de un documento html5 básica y lo guardamos con el nombre index en la carpeta del sitio creada en el punto 1.

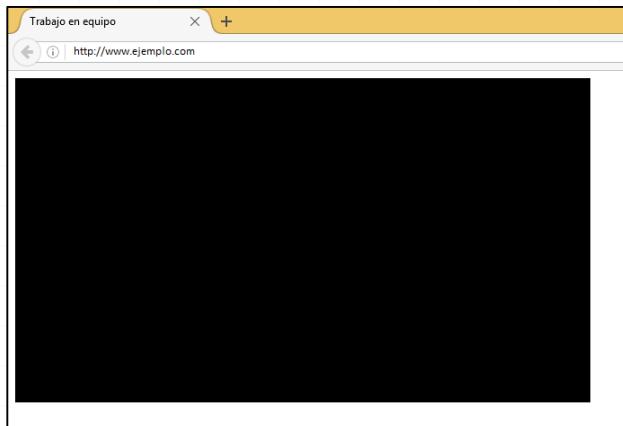
# Incluir el vídeo

- Para incluir el video, utilizaremos la etiqueta <video> en nuestro body.

```
<div id="mivideo">  
    <video src="video/video1.mp4" > </video>  
</div>
```

- Como no todos los navegadores soportan los mismos formatos, deberíamos definir varios formatos de video para que cada navegador tome el que es capaz de reproducir (por lo menos el mp4 y el ogg ó ogv):

```
<video>  
    <source src="video/video1.mp4" type="video/mp4">  
    <source src="video/video1.ogv" type="video/ogv">  
    <source src="video/video1.webm" type="video/webm">  
    Lo siento, tu navegador no soporta vídeos en formato HTML5  
</video>
```



Lo anterior sería suficiente para incluir el video.

Sin embargo, si ejecutamos la página, veremos que el video no se ve. Falta reproducir el video de alguna forma.

# Reproducir el vídeo automáticamente

- Vamos a hacer que el vídeo se reproduzca automáticamente al cargar la página:

```
<video src="video/video1.mp4" autoplay>  
</video>
```

- Si utilizamos la segunda de las formas vistas, el autoplay lo deberemos de poner dentro de la etiqueta video.
- Al usar autoplay hemos conseguido que el video se cargue automáticamente y se visualice pero no tenemos forma de detenerlo antes de que finalice. Vamos a mostrar los controles nativos del navegador sobre el vídeo (botones play, pause, la barra de avance rápido, botón de pantalla completa y botón de audio)

- Para ello utilizaremos el atributo controls de la etiqueta video

```
<video src="video/video1.mp4" autoplay controls>
```

- Nota. Cada navegador proporciona una botonera diferente. El aspecto de la botonera dependerá del navegador con el que estemos viendo el vídeo.

# Tamaño del video

- Como no hemos especificado qué tamaño ha de tener el video, el tamaño con el que aparece el navegador será el que tenga el video.
- Este tamaño lo podemos cambiar con los atributos width y height.

```
<video src="video/video1.mp4"  
       autoplay controls  
       width="520">  
</video>
```

- Al cambiar el ancho sólo, el navegador asume que se quiere manipular el height de manera proporcional.
- Si ponemos el atributo loop, cuando el vídeo llega a su final comenzará la reproducción de nuevo y así mientras el usuario se mantenga en la página.
- Podemos añadir una imagen para que se muestre una imagen hasta que comience la reproducción del video. (Para este caso debemos quitar el autoplay si no, no tendríamos tiempo de ver la imagen)

```
<video src="video/video1.mp4"  
       controls width="520"  
       poster="video/portada/inicial.png">  
</video>
```



Aplicando estilos  
CSS al video

- Con css podemos modificar el estilo del vídeo, su colocación, etc.
- En nuestro ejemplo, vamos a crear un archivo para el archivo que va a contener los estilos.
- Vamos a dar los estilos al div que contiene el vídeo:

```
#mivideo{  
    width: 60%;  
    margin: 20px auto;  
    background: orange;  
    padding: 5px;  
    -moz-border-radius: 5px;  
    -webkit-border-radius: 5px;  
    border-radius: 5px;  
    border: navy 2px solid;  
}  
  
video {  
    width: 100%;  
}  
  
@media (max-width: 480px) {  
    #mivideo{  
        width: 95%;  
    }  
}
```

Colocar un video  
como fondo de  
pantalla

- La primera recomendación para incorporar un video como fondo de pantalla es quitarle la voz.
- Vamos a incorporar el video a nuestro documento html:

```
<body>
  <video autoplay loop muted>
    <source src="video/video1.mp4" type=video/mp4>
    <source src="video/video1.ogv" type=video/ogv>
    <source src="video/video1.webm" type=video/webm>
    Lo siento, tu navegador no soporta vídeos en formato HTML5
  </video>
</body>
```

# Aplicar estilos

```
* {  
    margin:0;  
    padding: 0;  
}
```

Lo ponemos fixed para que el video no  
haga scroll

```
video {
```

```
    position: fixed; ← Lo ponemos 100% el alto y el ancho  
    min-width: 100%;           para que ocupe toda la ventana  
    min-height: 100%;
```

Coloca el video por detrás de la  
z-index:-1; imagen

```
}
```

# Audio en HTML5

# Audios en nuestras páginas web

- Incluir audio en nuestras páginas web es casi idéntico a incluir video.
- Los formatos de video más habituales que soportan casi todos los navegadores a día de hoy son:
  - MP3. Está sujeto a licencia comercial
  - Ogg. Es un formato libre.
- Casi todos los navegadores hoy en día admiten el formato mp3.
- Existen conversores gratuitos (tanto online como descargables)que te permiten convertir cualquier formato (como por ejemplo .avi) a ogg o mp3. Incluso, te permitirán convertir uno de los formatos al otro, es decir, ogg a mp3 y viceversa.

# Etiqueta para incluir audio

- La etiqueta html5 que nos permite incluir audios en nuestras páginas web es la etiqueta `<audio>` de apertura y `</audio>` de cierre.
- Esta etiqueta cuenta con una serie de atributos de los cuales los más importantes son:
  - `src`. Permite especificar dónde se encuentra el video
  - `controls`. Permite especificar si nuestro video va a contar con la típica botonera con los botones de play, pause, etc.
  - `autoplay`. Permite especificar al navegador si nuestro video va a reproducirse automáticamente al cargar la página o no
  - `loop`. Permite especificar al navegador si al finalizar la reproducción de nuestro video, éste comenzará de nuevo. Es decir, si se reproducirá indefinidamente.
  - `preload`. Permite especificar algunas características importantes antes de cargarlo (por ejemplo, duración, etc.)

# Imágenes

# Imágenes en páginas web

- En ocasiones, una imagen resultará mucho más descriptiva que largos párrafos de texto que pueden llegar a cansar al usuario, ya lo dice el refrán “Vale más una imagen que mil palabras”.
- Debemos evitar abusar de las imágenes, utilizarlas en su justa medida y cuando su uso esté justificado.
- Para incluir una imagen en una página web utilizaremos la etiqueta `<img>` con sus atributos correspondientes.
- La sintaxis básica de la etiqueta `<img>` es la siguiente:

```

```

- url. Es la url de la imagen que se quiere mostrar. Esta url puede ser absoluta o relativa.
  - Por ejemplo,
- ```

```
- La etiqueta `<img>` es una etiqueta vacía, es decir, no delimita ningún texto por lo que podemos utilizar el atajo ya comentado para otras etiquetas para cerrarla: ``

# Formatos de imagen

## ○ GIF

- Diseñado para comprimir las imágenes digitales y, así, reducir su peso.
- Admite una paleta de hasta 256 colores. Esto nos permite optimizar el tamaño del archivo, pero no resulta muy adecuado para imágenes con cierta calidad.
- Permite transparencias e imágenes animadas.
- Es uno de los formatos más utilizados para publicar imágenes en la web por el bajo peso de las mismas.

## ○ JPG-JPEG

- Admite una paleta de hasta 16 millones de colores.
- Es el formato más común junto con el GIF para publicar imágenes en la web.
- Permite la compresión de imágenes, lo que hace que resulten imágenes de poco peso, pero esta compresión genera pérdida de calidad en la imagen (que será más apreciable cuantas más veces se guarde la imagen).

## ○ PNG

- Es un formato relativamente reciente que se está imponiendo rápidamente como alternativa a los dos formatos anteriores, ya que incorpora todas sus ventajas y evita sus inconvenientes.
- Permite comprimir las imágenes sin pérdida de calidad.
- Permite manejar transparencia.
- Permite imágenes animadas.



# Manos a la obra

```
<body>
```

```
    <h1>El león</h1>
```

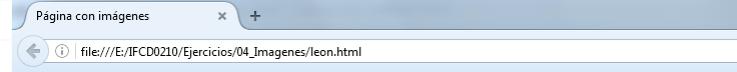
```
    
```

```
    <p>
```

De toda la familia de gatos, el león siempre ha sido considerado por el hombre como el rey de los animales.

```
    </p>
```

```
</body>
```



## EL LEÓN



De toda la familia de gatos, el león siempre ha sido considerado por el hombre como el Rey de los animales.

# Tamaño de imagen

- Si no indicamos otra cosa, al incluir una imagen en nuestra página, esta se mostrará con el tamaño original de la imagen. Sin embargo, podemos especificar un ancho y un alto para mostrar la imagen.
- Esto lo podemos conseguir de dos formas: mediante los atributos Width y Height o mediante el atributo Style.

# Atributos width y height

- Estos dos atributos nos permiten indicar al navegador el ancho y el alto, en píxeles, que deberá tener la imagen.

```

```

- value1, value2. Será un número y representa el tamaño en píxeles que queremos asignar de ancho y alto a la imagen

```

```

- Si solo ponemos uno de los dos parámetros, el navegador ajustará la medida del otro de manera proporcional.

```

```

```

```

```

```

- Los tres comandos anteriores, hacen que se visualice la imagen con igual tamaño.
- También se puede representar el ancho y alto en % respecto al tamaño de la pantalla

# Atributo style

- Otra forma de especificar el ancho y alto de una imagen es mediante el uso del atributo style:

```

```

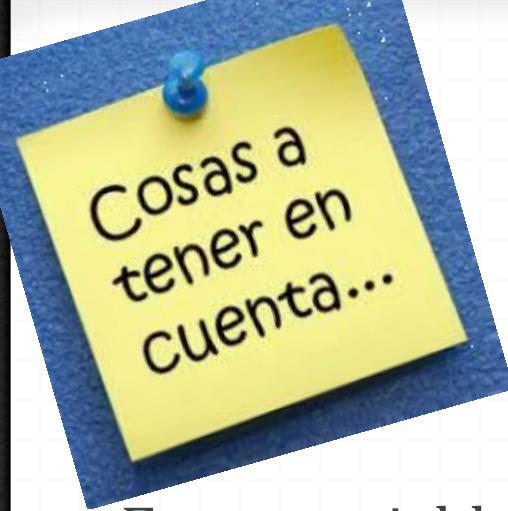
- value1, value2. Será un número y representa el tamaño en píxeles que queremos asignar de ancho y alto a la imagen.

```

```

- Al igual que en el apartado anterior, si sólo se indica uno de los dos parámetros (width o height) y se omite el otro, el navegador ajustará el tamaño que no se indica a la misma proporción que el parámetro especificado.

# Consejos



Cosas a tener en cuenta...

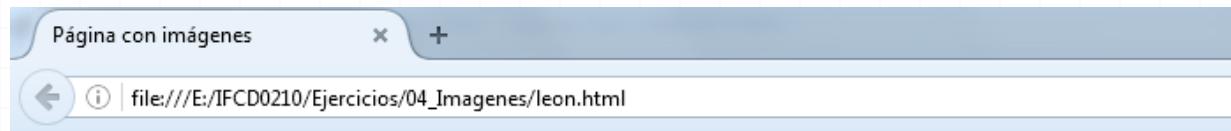
- Es aconsejable que crees las imágenes con el tamaño adecuado para la web.
- Tanto los atributos width y height, como el atributo style son válidos en HTML5. Sin embargo, es más aconsejable utilizar el atributo style ya que evita que las hojas de estilos cambien el tamaño original de las imágenes.



# Manos a la obra

0 Añadir la sentencia adecuada a la página del león para que tome un ancho de 200 px y actualice el alto proporcionalmente

```
<IMG src="imagenes/leon.png" style="width:200px;" />
```



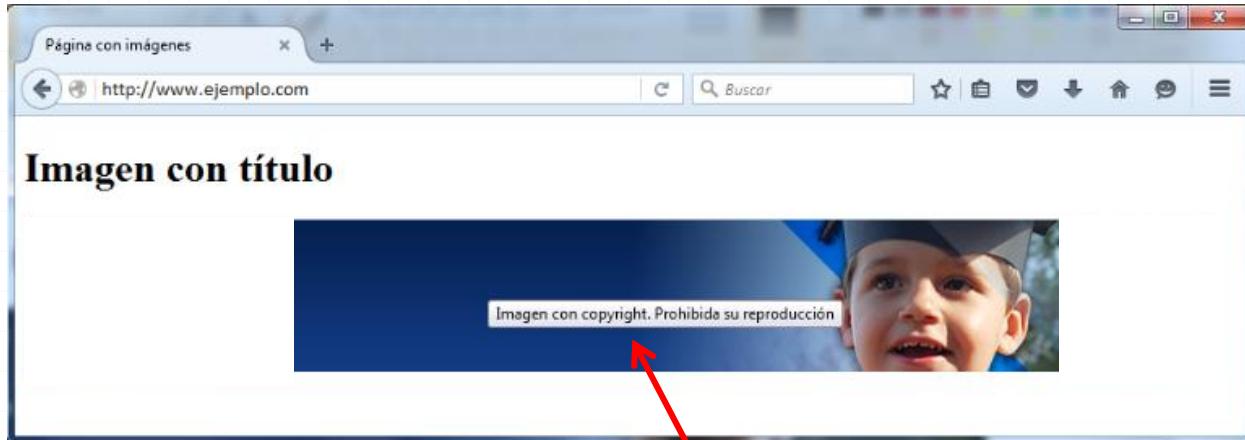
## EL LEÓN



De toda la familia de gatos, el león siempre ha sido considerado por el hombre como el Rey de los animales.

# Titulo de la imagen

- Este atributo se utilizará para ofrecer una descripción emergente de la imagen.
- 
- En el ejemplo anterior, al colocar el puntero del ratón sobre la imagen veríamos un mensaje emergente que pondría "*Imagen con copyright. Prohibida su reproducción*".



Mensaje emergente que aparece al colocar el puntero del ratón sobre el enlace

# Textos alternativos

- El atributo alt permite especificar un texto alternativo para una imagen, para el caso de que ésta no se pueda visualizar (ya sea porque la imagen no se cargue por alguna causa o por limitaciones del usuario, como por ejemplo una persona invidente).
- Si una imagen no puede mostrar la imagen, se mostrará el texto definido en el atributo alt.
- A continuación mostramos un ejemplo del uso de este atributo:

```

```



○ En esta página debería verse el logotipo de la academia, pero por algún problema el navegador no es capaz de mostrarlo, por lo que muestra el texto que se ha definido dentro del atributo alt y que no es otro que el de “*Logotipo de la academia*”

0 Las personas ciegas o con dificultades visuales, en muchas ocasiones utilizan screen readers (lectores de pantalla). Estos lectores de pantalla no son más que programas de software que pueden leer lo que el navegador está mostrando en la pantalla. Los lectores de pantalla son capaces de leer el texto incluido dentro del atributo alt.

# Diferencias entre title y alt

- El atributo alt muestra información sustitutiva de la imagen para el caso de que esta no esté visible mientras que el atributo title se emplea para proporcionar información adicional o complementaria, esta información no sustituye a la imagen.
- Siguiendo las recomendaciones del W3C para hacer una página accesible, el alt debería ser obligatorio en todas nuestras imágenes sin embargo el title es opcional.
- Tampoco es correcto incluir la misma información en los dos atributos.



# Manos a la obra

- Definir un título y un texto alt para la imagen del león.
  - El título debe poner: Todos los derechos reservados
  - El texto alternativo debe poner: Dibujo de un león

```

```

# Enlaces en imágenes

- Como comentábamos en el apartado 3.5 de este mismo tema, podemos asociar un enlace a una imagen de manera que al hacer clic sobre ella, nos lleve a otro recurso.
- Para asociar un enlace a una imagen utilizaremos la etiqueta `<a>` ya comentada:

```
<a href="url"></a>
```



# Manos a la obra

- Poner un enlace en la imagen del león para que al picar sobre ella nos lleve a la página de Wikipedia:

[https://es.wikipedia.org/wiki/Panthera\\_leo](https://es.wikipedia.org/wiki/Panthera_leo)

- Este enlace se deberá abrir en una pestaña nueva



# Manos a la obra

- Crear una página que incluya los logos de yahoo, google y bing.
- Cada uno de los logos deberá incluir un title que diga: “Ir a xxxx” donde xxx será el nombre del buscador correspondiente al logo.
- Cada logo deberá incluir también un texto alternativo que diga “Logotipo de xxx”
- Cada logotipo deberá tener un enlace al correspondiente buscador que se deberá abrir en una etiqueta nueva.
  - <https://www.google.es>
  - <http://www.bing.com/>



## Principales buscadores

**bing**™

**Google**

# Mapas de ímagenes

# Mapas de imagen

- A veces deseamos que una misma imagen tenga diferentes enlaces según la zona de la imagen sobre la que pique el usuario. Esto se consigue mediante los mapas de imágenes.
- Para mapear una imagen, necesitaremos varias etiquetas:
  - Primeramente, deberemos añadir el atributo usemap a la etiqueta `<img>` para indicar que la imagen es un mapa.

```

```
  - Luego, necesitaremos la etiqueta `<map> </map>` que nos permite definir el mapeo de la imagen. Esta etiqueta llevará un nombre que deberá ser el mismo que se ha especificado en el atributo usemap.

```
<map name="mapa1">  
  . . .  
</map>
```

# Atributo <área>

- Entre las etiquetas <map> de apertura y </map> de cierre incluiremos las etiquetas <area> que nos permitirá definir las áreas que van a contener los enlaces dentro de la imagen. Esta etiqueta cuenta con varios atributos:
  - shape="rect/poly/circle". Permite definir el tipo de área. Puede contener los valores rect (área rectangular), poly (área poligonal) o circle (área ovalada).
  - coords="coordenadas". Permite definir las coordenadas del área. En las áreas rectangulares deben especificarse las coordenadas de la esquina superior izquierda y las de la esquina inferior derecha. En las poligonales especificaremos las coordenadas de todos los vértices del área. En las circulares indicaremos las coordenadas del centro del círculo y el valor del radio
  - href="URL". Indica la dirección de la página web a la que se accede si pinchamos en un área predeterminada.

# Ejemplo

```
  
<map name="mapa">  
  <area shape="rect" coords="18,124,111,219" href="pagina1.html" />  
  <area shape="circle" coords="128,60,50" href="pagina2.html" />  
  <area shape="poly" coords="139,200, 193,110, 236,199"  
    href="pagina3.html" />  
</map>
```

- En el caso de la forma rect, las coordenadas son las de la esquina superior izquierda y la esquina inferior derecha del cuadrado o rectángulo.
- En el caso de circle, las dos primeras coordenadas definen el centro del círculo y la tercera es el radio.
- En el caso de poly especificaremos las coordenadas de dos en dos de cada uno de los puntos que forma el polígono.



# Manos a la obra

```
<body>
```

```
    
```

```
    <map name="mapa1">
```

```
        <area
```

```
            shape="rect" coords="18,124,111,219"  
            href="http://www.abc.es"
```

```
        />
```

```
        <area
```

```
            shape="circle" coords="128,60,50"  
            href="http://www.elpais.com"
```

```
        />
```

```
        <area
```

```
            shape="poly"  
            coords="139,200, 193,110, 236,199"  
            href="http://www.marca.com"
```

```
        />
```

```
    </map>
```

```
</body>
```

Imágenes de  
fondo

# Imagen de fondo de una página

- Generalmente, para poner una imagen de fondo en una página utilizaremos CSS pero aquí vamos a ver cómo lo haríamos utilizando exclusivamente html.
- Para añadir una imagen de fondo desde html utilizaremos el atributo background de la etiqueta body:

```
<body background="fondo.jpg">
```

- donde fondo.jpg es el nombre del archivo de imagen.
- Dependiendo de dónde esté guardada la imagen, bastará con incluir el nombre y la extensión o deberá especificarse la ruta donde se encuentra la imagen.



0 Más adelante veremos cómo hacer que la imagen se adapte a la pantalla a través de CSS.

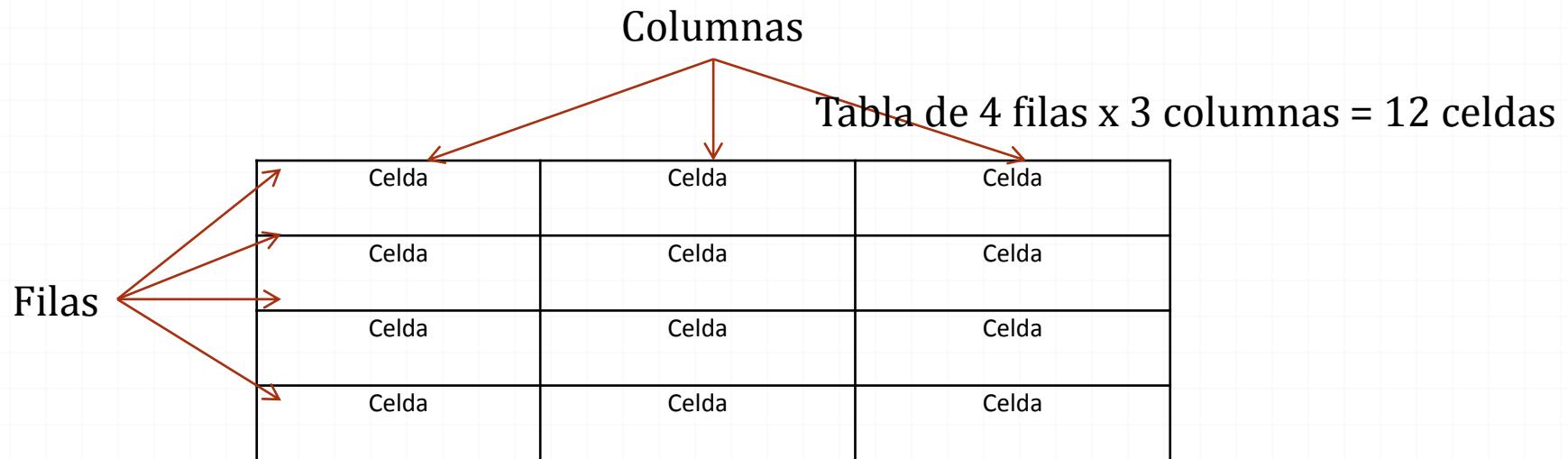
# Tablas

# Introducción

- Una tabla es un cuadro donde organizamos la información en filas y columnas.
- Hoy en día, HTML permite crear tablas con gran variedad de funciones y usos, lo que hace que sean una herramienta muy potente.

# Estructura básica

- 0 Llamamos fila a cada una de las divisiones en horizontal y columna a cada una de las divisiones en vertical. A la intersección de una fila con una columna es a lo que llamamos celda. Las celdas de la tabla son las que realmente abarcan el contenido que se quiere incluir en la tabla.



El contenido de cada celda puede ser muy diverso: texto o imágenes o incluso otra tabla

# Crear una tabla

- A la hora de crear una tabla con HTML deberemos utilizar la etiqueta `<table>` para indicar dónde comenzará la tabla y `</table>` para indicar el final de la tabla.
- La etiqueta `<table>` delimita el espacio de una tabla, pero no es suficiente por sí sola para crear la tabla. Para poder crear una tabla deberemos definir las filas mediante las etiquetas `<tr>` de apertura y `</tr>` de cierre.
- Y, para cada fila, deberemos definir las columnas mediante las etiquetas `<td>` de apertura y `</td>` de cierre.



# Manos a la obra

```
<table>
<tr>
  <td>
    Celda 1.1
  </td>
  <td>
    Celda 1.2
  </td>
</tr>
<tr>
  <td>
    Celda 2.1
  </td>
  <td>
    Celda 2.2
  </td>
</tr>
<tr>
  <td>
    Celda 3.1
  </td>
  <td>
    Celda 3.2
  </td>
</tr>
</table>
```

## Ejemplo de tabla

Celda 1.1	Celda 1.2
Celda 2.1	Celda 2.2
Celda 3.1	Celda 3.2

# Bordes

- En el ejemplo anterior, observamos hemos creado una tabla de 3 filas y 2 columnas (en total 6 celdas).
- Como podemos ver en la imagen, vemos que por defecto, las celdas no tienen bordes que marquen la separación entre sí. Si quisiéramos que el navegador mostrar una línea para separar las celdas, deberíamos añadir el atributo border dentro de la etiqueta <table>.
- <table border="1">

Celda 1.1	Celda 1.2
Celda 2.1	Celda 2.2
Celda 3.1	Celda 3.2

# Título de tabla

- Además de filas y columnas, la tabla puede incluir un título. En caso de incluirlo, el título debe indicar el contenido de la tabla.
- Para añadir un título a la tabla, utilizaremos la etiqueta `<caption>`. Esta etiqueta debe ir inmediatamente después de la etiqueta `<table>`

```
<table>
    <caption> Título de la tabla</caption>
    ...
</table>
```

Título de la tabla	
Celda 1.1	Celda 1.2
Celda 2.1	Celda 2.2
Celda 3.1	Celda 3.2



# Crear la siguiente tabla

## Libreria "El libro amigo"

Listado de libros infantiles

Momo	Michael Ende	17,00 €
El Principito	Antoine de Saint-Exupéry	15,00 €

# Fila de título

- Al crear una tabla, ésta puede contener filas y columnas con diferente tipo de información.
- Es muy habitual que se quiera identificar el tipo de contenido de las celdas mediante un título.
- Con este objetivo HTML nos ofrece la etiqueta <th>.



Modificar la tabla del ejercicio anterior para que tenga la estructura siguiente

## Libreria "El libro amigo"

Listado de libros infantiles

Titulo	Autor	Precio
Momo	Michael Ende	17,00 €
El Principito	Antoine de Saint-Exupéry	15,00 €

```
<table border="1">
  <caption>
    Listado de libros infantiles
  </caption>
  <tr>
    <th>Titulo</th>
    <th>Autor</th>
    <th>Precio</th>
  </tr>
  <tr>
    <td>Momo</td>
    <td>Michael Ende</td>
    <td>17,00 €</td>
  </tr>
  <tr>
    <td>El Principito</td>
    <td>Saint-Exupéry</td>
    <td>15,00 €</td>
  </tr>
</table>
```

# Columnas de título

- 0 La etiqueta <th> se puede usar tanto para definir filas como columnas de títulos



# Modificar la tabla del ejercicio anterior para que tenga la estructura siguiente

## Libreria "El libro amigo"

Listado de libros infantiles

	Titulo	Autor	Precio
Libro 1	Momo	Michael Ende	17,00 €
Libro 2	El Principito	Antoine de Saint-Exupéry	15,00 €

```
<table border="1">
  <caption>
    Listado de libros infantiles
  </caption>
  <tr>
    <th> </th>
    <th>Titulo</th>
    <th>Autor</th>
    <th>Precio</th>
  </tr>
  <tr>
    <td>Libro 1</td>
    <td>Momo</td>
    <td>Michael Ende</td>
    <td>17,00 €</td>
  </tr>
  <tr>
    <td>Libro 2</td>
    <td>El Principito</td>
    <td>Saint-Exupéry</td>
    <td>15,00 €</td>
  </tr>
</table>
```

# Cabecera, Cuerpo y pie de una tabla

- En ocasiones, la información que contiene una tabla es muy extensa.
- En este caso, sería útil contar con herramientas que nos permitieran mejorar la organización de la información. Con este propósito, HTML nos ofrece las etiquetas `<thead>`, `<tbody>` y `<tfoot>`.
- Las tres etiquetas anteriores nos permiten dividir la tabla en tres secciones: cabeza, cuerpo y pie.



```
<table border="1">
  <caption>
    Listado de libros infantiles
  </caption>
  <thead>
    <tr>
      <th></th>
      <th>Titulo</th>
      <th>Autor</th>
      <th>Precio</th>
    </tr>
  </thead>
  <tfoot>
    <tr>
      <td colspan="4">
        Listado libros infantiles
      </td>
    </tr>
  </tfoot>
  <tbody>
    <tr>
      <th>Libro 1</th>
      <td>Momo</td>
      <td>Michael Ende</td>
      <td>17,00 €</td>
    </tr>
    <tr>
      <th>Libro 2</th>
      <td>El Principito</td>
      <td>Saint-Exupéry</td>
      <td>15,00 €</td>
    </tr>
  </tbody>
</table>
```

## Libreria "El libro amigo"

Listado de libros infantiles

	Titulo	Autor	Precio
Libro 1	Momo	Michael Ende	17,00 €
Libro 2	El Principito	Saint-Exupéry	15,00 €

Listado libros infantiles

Si nos fijamos en el código, vemos que la etiqueta `<tfoot>` se ubica después de la etiqueta `</thead>`, sin embargo se visualiza en el pie de la tabla.

\* Colspan lo veremos más adelante

# Resumiendo

Título de la tabla

Celda

Fila

Columna

Cabecera

Cuerpo

Pie

	Titulo	Autor	Precio
Libro 1	Momo	Michael Ende	17,00 €
Libro 2	El Principito	Saint-Exupéry	15,00 €
Listado libros infantiles			

# Formato de tablas

# Borde de tabla

- Como comentamos anteriormente, por defecto al definir una tabla, las celdas no tendrán ningún tipo de borde. Si queremos que las celdas aparezcan delimitadas mediante una línea, deberemos utilizar el atributo border.

```
<table border="px">
```

- px representa el grosor del borde en píxeles.
- También podemos definir un color para el borde con el atributo bordercolor:

```
<table bordercolor="color">
```

- color es el color que queremos asignar al borde. Para especificar el color podremos utilizar cualquiera de los métodos vistos anteriormente.

# Alineación

- Podemos definir una alineación para la tabla mediante el atributo align.

```
<table align="alineación">
```

- Donde alineación tomará diferentes valores según la alineación que queramos definir:

Valor align	Efecto
align="center"	Mostrará la tabla centrada dentro de la ventana del navegador
align="right"	Mostrará la tabla alineada a la derecha dentro de la ventana del navegador
align="left"	Mostrará la tabla alineada a la izquierda dentro de la ventana del navegador. Esta es la opción por defecto.

# Tamaño de la tabla

- Por defecto, a medida que vamos tecleando texto en una tabla, ésta se expandirá para dar cabida al contenido de sus celdas. Si la tabla llegara al límite del espacio disponible en la ventana del navegador, el texto que exceda dicho ancho pasará a la nueva línea.
- No obstante, podemos asignar un tamaño fijo a la tabla para que ésta no se expanda. Para ello utilizaremos el atributo width.

```
<table width="tamaño">
```

- tamaño. Será la anchura de la tabla en píxeles o porcentaje.

# Fondo de la tabla

○ Se puede definir un fondo para la tabla utilizando los atributos background o bgcolor:

○ bgcolor. Nos permite definir un color de fondo para la tabla:

```
<table bgcolor="color">
```

○ color. Definiremos el color que queramos utilizar como fondo para la tabla con cualquiera de los formatos vistos anteriormente.

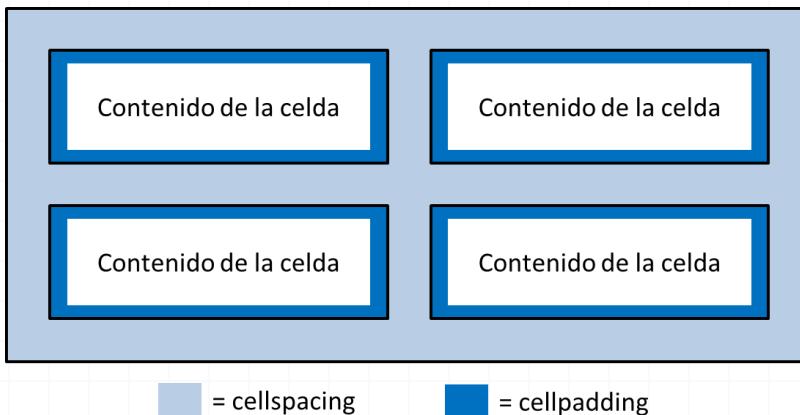
○ background. Nos permite colocar una imagen como fondo en una tabla.

```
<table background="url">
```

○ url. Es la ruta donde se encuentra la imagen almacenada. Esta url puede ser absoluta o relativa, siguiendo el formato visto en puntos anteriores.

# cellpadding y cellspacing

- 0 Estos dos atributos se añaden a la etiqueta <table> para modificar la anchura interior de la celda y/o el espacio entre celdas (respectivamente).





# Manos a la obra

Crear un documento con 4 tablas cada una de las cuales debe tener la siguiente configuración:

Código

Efecto

```
<table  
border="1">
```

Momo	Michael Ende	17,00 €
El Principito	Antoine de Saint-Exupéry	15,00 €

```
<table  
border="1"  
cellpadding="12">
```

Momo	Michael Ende	17,00 €
El Principito	Antoine de Saint-Exupéry	15,00 €

```
<table  
border="1"  
cellspacing="12">
```

Momo	Michael Ende	17,00 €
El Principito	Antoine de Saint-Exupéry	15,00 €

```
<table  
border="1"  
cellspacing="12"  
>
```

Momo	Michael Ende	17,00 €
El Principito	Antoine de Saint-Exupéry	15,00 €

Si quisieramos quitar ese efecto de doble línea que se muestra como borde de la tabla, podemos poner cellspacing="0". Crear una quinta tabla con cellspacing="0"

**NOTA.** En la actualidad, todos los atributos vistos en este apartado no se suelen utilizar y se sustituyen por estilos CSS.

# Formato de contenido de las celdas

- En el apartado anterior, hemos visto cómo aplicar diferentes formatos a la tabla al completo. Si en lugar de aplicar los atributos mencionados a la etiqueta `<table>`, los aplicamos a las etiquetas `<td>` o `<tr>`.

```
<tr atributo="valor">  
<td atributo="valor">
```

- Si lo aplicamos a la etiqueta `<td>` cambiaremos el aspecto de una única celda, si lo aplicamos a la etiqueta `<tr>` cambiamos el formato de todas las celdas que se encuentran en esa fila.

**NOTA.** Al igual que en el caso anterior, la mayoría de los formatos descritos en este apartado hoy en día se consiguen utilizando estilos CSS.

Atributo	Valores que puede tomar	Efecto
width	Puede ser un número expresado en píxeles o se puede acompañar del signo % y en ese caso será un porcentaje	Permite cambiar el ancho de la celda
align	Left (izquierda. Valor por defecto) Right (derecha) Center (centro)	Permite cambiar la alineación horizontal del texto de la celda
valign	Baseline (línea de base) Bottom (inferior) Middle (medio) Top (superior)	Permite cambiar la alineación vertical del contenido de la celda
bgcolor	Cualquiera de las formas vistas para definir colores	Permite cambiar el color de fondo de la celda
background	url de la imagen	Permite poner una imagen de fondo en la celda
bordercolor	Cualquiera de las formas vistas para definir colores	Permite cambiar el color del borde de la celda



Agrupamiento de  
filas y columnas

# Expandir celdas

- En ocasiones necesitamos unir dos o más celdas en una sola que ocupe el espacio de las celdas que hemos unido. Es lo que llamamos expandir una celda y lo podemos hacer en horizontal (expandir columna) o en vertical (expandir fila). Para ello contamos con los colspan y rowspan, respectivamente.

Colspan

Esta celda ocupa el espacio de cinco hacia la derecha

Rowspan

Esta celda ocupa el espacio de las tres siguientes hacia abajo

Gastos mensuales primer semestre 2015				
Trimestre	Mes	Luz	Agua	Teléfono
Primer trimestre	Enero	10 €	15 €	20 €
	Febrero	10 €	15 €	20 €
	Marzo	10 €	15 €	20 €
Segundo trimestre	Abril	10 €	15 €	20 €
	Mayo	10 €	15 €	20 €
	Junio	10 €	15 €	20 €

Gastos mensuales primer semestre 2015				
Trimestre	Mes	Luz	Agua	Teléfono
Primer trimestre	Enero	10 €	15 €	20 €
	Febrero	10 €	15 €	20 €
	Marzo	10 €	15 €	20 €
Segundo trimestre	Abril	10 €	15 €	20 €
	Mayo	10 €	15 €	20 €
	Junio	10 €	15 €	20 €

```

<table border="1" cellspacing="0">
  <tr>
    <th colspan="5">Gastos mensuales</th>
  </tr>
  <tr>
    <td>Trimestre</td>
    <td>Mes</td>
    <td>Luz</td>
    <td>Agua</td>
    <td>Teléfono</td>
  </tr>
  <tr>
    <td rowspan="3">Primer trimestre</td>
    <td>Enero</td>
    <td>10 €</td>
    <td>15 €</td>
    <td>20 €</td>
  </tr>

```

```

    <tr>
      <td>Febrero</td>
      <td>10 €</td>
      <td>15 €</td>
      <td>20 €</td>
    </tr>
    <tr>
      <td>Marzo</td>
      <td>10 €</td>
      <td>15 €</td>
      <td>20 €</td>
    </tr>
    <tr>
      <td rowspan="3">Segundo trimestre</td>
      <td>Marzo</td>
      <td>10 €</td>
      <td>15 €</td>
      <td>20 €</td>
    </tr>
    <tr>
      <td>Abril</td>
      <td>10 €</td>
      <td>15 €</td>
      <td>20 €</td>
    </tr>
    <tr>
      <td>Mayo</td>
      <td>10 €</td>
      <td>15 €</td>
      <td>20 €</td>
    </tr>
  </table>

```



# Tablas anidadas

- HTML nos da la posibilidad de anidar tablas. Es decir, podemos insertar una tabla dentro de otra.
- 
- Para anidar una tabla en otra, bastará con incluir la estructura completa de la tabla dentro de una celda.

```
<table>
  <tr>
    <td>
      <table>
        ...
      </table>
    </td>
    ...
  </tr>
  ...
</table>
```

- A la hora de anidar tablas deberemos ser extremadamente cuidadosos, ya que es un proceso complejo y complicado, además de que cuantas más tablas anidemos unas dentro de otras, más aumentaremos el tiempo que el navegador deberá emplear para cargar nuestra página. Por esto, se aconseja evitar la anidación de tablas.



Buenas prácticas  
en el uso de  
tablas

- Como comentamos en el apartado anterior, el anidamiento de tablas puede ralentizar la carga de la página por lo que conviene evitar el anidamiento de tablas o, por lo menos, reducir el número de niveles lo más posibles. Lo ideal es que no hubiera tablas anidadas.
- En ocasiones se emplea el anidamiento de tablas para conseguir ciertos efectos como bordes dobles, distancias a márgenes, etc. Esto es un gran error, lo correcto sería utilizar estilos CSS.
- Es recomendable utilizar el atributo width (bien sea en valor absoluto o relativo) para que el navegador sepa dónde termina la tabla, esto ayudará a que el navegador cargue la tabla más rápidamente.
- Evitar el uso del atributo height para especificar la altura de una celda. Es mejor utilizar estilos CSS. Y en el caso de que nos veamos obligados, utilizarlo a nivel de fila, nunca de celda.
- Evitar los atributos que permiten dar formato a las tablas, filas y columnas codificados directamente en HTML. Es preferible hacerlo a través de CSS.
- Tener mucho cuidado de que ninguna etiqueta <tr>, <td>, <th> o <caption> se queden sin cerrar.