

Aplicaciones Web y Lenguajes de Marcas
**Tema 1b2: Documentos Web con Lenguajes de
Marcas => HTML**

2º Parcial - LMSGI
1º de Ciclo Superior de DAM

les de Teis (Vigo) – Curso 2020/21

Tema 1b2: Documentos Web con Lenguajes de Marcas => HTML

Objetivos: saber utilizar HTML para crear y modificar documentos web.

- Identificar las diferencias entre HTML y XHTML.
- Versiones de HTML.
- Software necesario para crear páginas web.
- Identificar la estructura de un documento HTML:
 - Sintaxis
 - Semántica
 - Reconocer los principales elementos de HTML.
- Utilizar HTML para crear y modificar páginas web:
 - Títulos
 - Listas
 - Imágenes
 - Hipervínculos
 - Tablas
 - **Formularios**
 - **Audio y vídeo**

Formularios

- **Formularios:**

*Elemento fundamental en las páginas web para **recopilar datos** de los usuarios, **enviando** dicha información a un **servidor web** para que la **procese***

- Imprescindibles para obtener información de los usuarios y adaptar los contenidos de las páginas web a sus necesidades.
 - Ej: solicitud de información, registro de usuarios, encuestas, buscador, compra de un producto, etc.
- La **interactuación** entre una página web y los usuarios **permite**:
 - 1) Recabar datos de los usuarios.
 - 2) Enviar dicha información a un servidor (web/de aplicaciones) indicando su dirección URL.
 - 3) El servidor, a través de un script, procesa los datos y envía un mensaje respuesta al usuario.
- Elementos básicos de un formulario:
 - Texto y **elementos de Html**.
 - **Controles** de formulario:
 - ✓ **Cajas de texto**
 - ✓ **Botones de selección**
 - ✓ **Menú de opciones**

Formularios

- **Etiquetas para crear un formulario:**

- **<form>** y **</form>**: definición general del **formulario**. Dentro definiremos el contenido del mismo utilizando **controles** y **texto**.

- **Atributos imprescindibles:**

- ✓ **action**: indica la **acción** que realizará nuestro formulario => **URL** con la dirección de la *página web o servicio al cual se enviarán los datos* recogidos a través de los diferentes controles del formulario para **procesarlos**.

- **Sintaxis:**

```
<form action="ruta/recurso.php">
....
</form>
```

- ✓ **method**: indica el **método http** utilizado para **transferir los datos** recopilados en el formulario. Los métodos de **paso de datos** más utilizados son:

- a) **GET** o **get**: los **datos** se pasan en la propia **URL**, ya que genera una cadena de búsqueda en la URL que contiene los datos del formulario.

- **Sintaxis:**

```
<form method="get">
```

- b) **POST** o **post**: los **datos** se pasan internamente, en la cabecera http, de forma invisible para el usuario.

- **Sintaxis:**

```
<form method="post">
```

Formularios

- **Ventajas de utilizar GET:** se pueden almacenar las URL en favoritos
- **Inconvenientes de utilizar GET:**
 - ✓ Limitación de la URL (256 caracteres).
 - ✓ **NO** permite el envío **de archivos adjuntos** con el formulario.
 - ✓ Los datos enviados se ven en la barra de dirección del navegador.
 - Ej: `http://www.servidor.es?campo1=valor1&campo2=valor2...`
- **Ventajas de utilizar POST:** no tiene ninguno de los inconvenientes del método GET. Tampoco su ventaja.
- **¿Cuándo utilizar GET y cuando POST?**
 - **GET:** en formularios que **no modifican** la información.
 - Ej: formularios de búsqueda.
 - **POST:** en formularios que **sí modifican** la información que se almacena en una bd.
 - Ej: formularios para insertar, modificar o borrar datos en una bd.

```
<form action="http://servidor.es/formulario1.php" method="post">
```

Formularios: <input>

- **Elementos de un formulario:**
 - a) **<input>**: permite crear la mayoría de los *controles* que se añaden a un formulario para que los usuarios cubran los datos que se piden.
 - **Atributos de <input>**:
 - ✓ **type**: imprescindible, ya que indica el **tipo de control** que queremos crear:
 - a) **text**: para crear cuadros de texto
 - b) **password**: para crear cuadros de contraseña
 - c) **checkbox**: para crear casillas de verificación
 - d) **radio**: para crear botones de radio
 - e) **submit**: para crear botón de envío
 - f) **reset**: para crear botón de restablecer
 - g) **file**: para adjuntar un archivo
 - h) **hidden**: para crear campos ocultos
 - i) **image**: para personalizar los botones
 - j) **button**: para crear botones adicionales
 - Ej: `<input type="text"/>`

Formularios: atributos de <input>

- **Atributos de <input>:**
- ✓ **name:** asigna un nombre interno al control para **identificar** el elemento.
 - No asignar caracteres problemáticos.
 - **Imprescindible** para procesar la información del formulario.
- ✓ **value:** valor inicial del control (valor predeterminado).
- ✓ **size:** **tamaño** inicial del control. Se puede indicar como **número de caracteres** (en campos de *texto* y de *password*) y en **px** para el resto de controles.
- ✓ **maxlength:** nº máximo de caracteres para los controles de *texto* y de *password*. Permite limitar el tamaño del valor introducido. Muy útil para campos con formato predefinido y limitado.
- ✓ **checked:** permite indicar qué opción aparece preseleccionada en los controles *checkbox* y *radiobutton*. Valor único "**checked**".
- ✓ **readonly:** el contenido del control no se puede modificar. Valor "**readonly**".
- ✓ **disabled:** deshabilita el control y su valor no se envía al servidor. Valor "**disabled**".
- ✓ **src:** indica la URL de la imagen que se emplea como botón de formulario. Valor "**url**".
- ✓ **alt:** descripción del control.
- ✓ **id:** identificador del control.

Formularios: <input>

- **Cuadro de texto:**

- Tiene la apariencia de un rectángulo vacío.
- El usuario puede escribir cualquier texto o número (que se interpreta como texto).
- El valor de **type** es **text**.
- Ej: `Escriba su nombre <input type="text" name="nombre" />`

- **Cuadro de contraseña:**

- Tiene la apariencia de un rectángulo vacío.
- El usuario puede escribir cualquier texto y los navegadores ocultan dicho texto, mostrando en pantalla asteriscos o círculos.
- Ideal para escribir contraseñas o datos sensibles.
- El valor de **type** es **password**.
- Ej: `Escriba su contraseña <input type="password" name="contrasena" />`

- **Atributos comunes para cuadros de texto y de contraseña:**

- ✓ **type**
- ✓ **name**
- ✓ **value**
- ✓ **size**
- ✓ **maxlength**

Formularios: <input>

- **Botones de radio:**

- Tiene la apariencia de un círculo.
- El usuario puede seleccionar **una** de las opciones relacionadas que se le ofrecen, ya que son **excluyentes**.
- La información que se envía al servidor es el **name** y **value** de la opción seleccionada.
- El valor de **type** es **radio**.
- Ej: `Hombre <input type="radio" name="sexo" value="hombre"/>`

- **Casillas de verificación:**

- Tiene la apariencia de un pequeño rectángulo.
- El usuario puede seleccionar **una** o varias opciones relacionadas que se le ofrecen, ya que **no** son **excluyentes**.
- La información que se envía al servidor es el **name** y **value** de la opción seleccionada.
- El valor de **type** es **checkbox**.
- Ej: `<input type="checkbox" name="pasatiempo" value="leer" />Leer`

- **Atributos comunes para botones de radio y casillas de verificación:**

- ✓ **type**
- ✓ **name**
- ✓ **value**
- ✓ **checked**

Formularios: <input>

- **Botón de acción: envío de datos -submit-**
 - Tiene la apariencia de un botón rectangular.
 - Permite realizar la **comunicación** entre la página que contiene el formulario y la página destino que procesará los datos que se hayan cubierto:
 - Al pulsar el botón, todos los datos recabados en los controles del formulario se envían a la página receptora del formulario (**action**).
 - El valor de **type** es **submit**.
- **Botón de acción: reestablecer datos -reset-**
 - Tiene la apariencia de un rectángulo vacío.
 - Borra los datos que haya introducido el usuario en los controles del formulario, dejándolos con sus **valores por defecto**.
 - El valor de **type** es **reset**.
 - Ej: `<input type= "reset" />`
- **Atributos comunes de los botones de acción:**
 - ✓ **value**: modifica el texto predeterminado del botón.
 - Ej: `<input type="submit" value="Buscar" />`

Formularios: <input>

- **Ej9a:** Crea una página web que incluya un formulario con los controles, etiquetas y botones vistos hasta ahora.
- Comprueba el funcionamiento de los elementos incluidos.
- Ej:

Formulario de contacto

Escriba su nombre:

Escriba su e-mail:

Escriba su teléfono:

¿Le gusta nuestra web?

☒ Sí

☐ No

¿Cómo podemos contactar con Usted?

☐ Por teléfono

☐ Por email

Formularios: <label>

- **Etiqueta de controles:** **<label>** y **</label>**
 - Se utiliza para definir el **título** o **leyenda** de los controles de los formularios.
 - Ayudan al usuario de la página web a entender mejor la función de cada control:
 - **Cuadros de texto:** describen el contenido.
 - **Checkbox y Radiobuttons:** dan nombre a las diferentes opciones.
 - **Funcionamiento al hacer clic sobre el texto que contiene la etiqueta:**
 - **Cuadros de texto:** el puntero del ratón se posiciona en la caja de texto para escribir.
 - **Checkbox y Radiobuttons:** se activa la opción seleccionada.
- **Ventajas:**
 - Código mejor estructurado.
 - Mejora la **usabilidad**.
 - Con el atributo **for** se relaciona el título con el control -a través del **id**-.

▪ **Ej:**

```
<label>Introduzca su nombre:  
    <input type="text" name="nombre" />  
</label>
```

```
<label for="nombre">Introduzca su nombre:</label> <br/>  
    <input type="text" id="nombre" name="nombre" />
```

Formularios

➤ Sol. Ej9a: página web con un formulario y utilizando etiquetas <label>

```
<form action="servidor.com/gestion.php" method="get">
  <p> <!--uso d label relacionando el título con el control gracias a la etiqueta d cierre de label-->
    <label>Escriba su nombre:
    <input type="text" name="nombre" id="nombre" />
    </label>
  </p>
  <p>
    <label>Escriba su e-mail:
    <input type="text" name="correo" id="apellido" />
    </label>
  </p>
  <p> <!--uso d label relacionando el valor de for con el id del control-->
    <label for="telefono">Escriba su teléfono:</label>
    <input type="text" name="telefono" id="telefono" />
  </p>
  <p>¿Le gusta nuestra web? <br />
    <label>
      <input type="radio" name="gusta" id="si" value="si" checked="true" /> Sí
    </label>
    <br />
    <label>
      <input type="radio" name="gusta" id="no" value="no" /> No
    </label>
  </p>
  <p>¿Cómo podemos contactar con Usted? <br />
    <label>
      <input type="checkbox" name="contactar" id="contactar_telefono" value="contactar_telefono"/>
      Por teléfono
    </label>
    <br />
    <label>
      <input type="checkbox" name="contactar" id="contactar_email" value="contactar_email" /> Por email
    </label>
  </p>
  <br />
  <input type="submit" value="Enviar" />
  <input type="reset" value="Restablecer" />
</form>
```

Formularios: <input>

▪ Cuadro de Selección de archivo:

- Tiene la apariencia de un botón, con el texto “**Examinar...**”.
- Al pulsar el botón, se puede elegir un fichero para adjuntar en el formulario.
- La ruta local del fichero seleccionado es lo que se guarda para enviar junto con el resto de campos del formulario.
- El valor de **type** es **file**.
- Ej: `Adjunte su cv <input type="file" name="cv" />`

✓ **enctype**: atributo de <form>, obligatorio añadirlo, valor “**multipart/form-data**”

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

▪ Atributos específicos:

- ✓ **accept**: permite indicar el **tipo MIME** de archivos que aceptará el servidor. Se pueden indicar varios tipos.
 - Ej: **image/*** -indica que se acepta cualquier formato de imagen-, **audio/***, **video/***, **image/jpeg** -indica un tipo MIME completo-.

```
<input type="file" name="imagensubida" accept="image/png, .jpeg, .jpg, image/gif">
```

- ✓ **multiple**: permite seleccionar más de un archivo para subir al servidor. Para que funcione correctamente, el valor de name debe terminar con [].

```
<input type="file" name="archivossubidos[]" multiple>
```

Formularios

- **Ej9b:** Crea una página web que incluya un formulario que permita adjuntar ficheros.
 - Revisa los atributos del formulario para que se permita dicho envío:
 - Método de envío
 - Tipo de encriptación de datos
 - Comprueba el funcionamiento del cuadro de adjuntar ficheros si se permite:
 - Un único fichero adjunto
 - Varios
- **Ej:**

Envío de ficheros

Fichero a enviar: No se ha seleccionado ningún archivo.

Fichero a enviar: adsf.pdf

Fichero a enviar: 2 archivos seleccionados.

Formularios: <input>

▪ Botón de imagen:

- Permite personalizar los botones con la apariencia de la imagen elegida mediante **src**.
- Funciona como un **submit**.
- El valor de **type** es **image**.

- Ej: `<input type="image" src="imagenes/icono_pino.png" />`



▪ Botón genérico:

- Tiene la apariencia de un cuadro de texto, que simula un botón rectangular.
- Sirve para capturar su pulsación (mediante JavaScript) y responder en consecuencia.
- El valor de **type** es **button**.
- Actualmente se recomienda crear un botón con la etiqueta **<button>**

- Ej: `<input type="button" value="Haz clic" size="8" />`

Haz clic

Formularios: <select>

- **Listas desplegables o cuadros combinados:** <select> y <option>
- Al hacer click sobre la lista desplegable, se muestran varias opciones de las que podremos escoger una o varias:
 - a) Muestra una opción en pantalla y permite elegir sólo una.
 - b) Muestra varias opciones en pantalla y permite elegir sólo una.
 - c) Muestra varias opciones en pantalla y permite elegir varias.
- <select> y </select>: es la etiqueta padre, sirve para definir la lista y encerrar todas las opciones que muestra.
 - **Atributos:**
 - ✓ **name:** identificación del control
 - ✓ **size:** define el nº de opciones a mostrar
 - ✓ **multiple:** permite elegir varias opciones. Valor único “multiple”.
- <option> y </option>: para definir cada opción de la lista
 - **Atributos:**
 - ✓ **value:** valor del control escogido
- **Ej:**

```
<select name="provincia">
  <option value="av">Ávila</option>
  <option value="bu">Burgos</option>
  <option value="za">Zamora</option>
</select>
```

Formularios: <select>

- Listas desplegables o cuadros combinados:
- <optgroup> y </optgroup> :
 - Permite agrupar opciones dentro de la lista desplegable
 - ✓ **label**: atributo para darle un título a cada grupo de opciones. Dicho título no se puede seleccionar como opción.
 - Ej:

```
<select name="provincia">
  <optgroup label="Castilla y León">
    <option value="av">Ávila</option>
    <option value="bu">Burgos</option>
    <option value="za">Zamora</option>
  </optgroup>
</select>
```

Lista desplegable de Provincias

Elija su provincia:

Castilla y León

- Ávila
- Burgos
- León
- Palencia
- Salamanca
- Segovia
- Soria
- Valladolid
- Zamora

Galicia

- La Coruña
- Lugo
- Orense
- Pontevedra

Formularios: <textarea>

- **Áreas de Texto -cuadros de texto multilínea-:** <textarea> y </textarea>
 - Tiene la apariencia de un rectángulo.
 - Son cuadros de texto creados para añadir textos largos, para sugerencias u observaciones, ya que permiten introducir varias líneas.
 - Se puede especificar el tamaño del área de texto mediante la especificación de nº de líneas y nº de columnas.
 - El contenido de este elemento representa el **valor inicial**.
 - **Atributos:**
 - ✓ **cols:** controla la **anchura** total de los caracteres que se podrán escribir como máximo en cada fila.
 - ✓ **rows:** indica la **altura** en líneas de texto del cuadro
- **Ej:**

```
¿Le gusta nuestra web? <br />  
<textarea rows="10" cols="50" name="texto">  
Me parece una web muy interesante porque ...  
</textarea>
```

Cuadro de texto multilínea

¿Le gusta nuestra web?
Me parece una web muy interesante porque ...

Formularios: <fieldset>

- **Agrupación de controles:** <fieldset> y </fieldset>
 - Hay situaciones en la que los campos de los formularios deben mostrarse agrupados: por significado, por importancia, ...
 - Con <fieldset> podemos **agrupar visualmente** una serie de **elementos relacionados** del formulario permitiendo diferenciar partes del formulario.
 - Muestra un rectángulo que contiene todos los elementos y controles incluidos en él. Ocupa el ancho de toda la pantalla.
 - <legend>: para añadir un título al grupo de controles agrupados.

▪ Ej:

```
<fieldset >
  <legend>Datos personales</legend>
  <input type="radio" name="sexo" value="hombre"/>Hombre<br />
  <input type="radio" name="sexo" value="mujer" />Mujer<br />
  Estado civil: Casado <input type="checkbox" name="estadocivil"
value="casado" /> <br />
</fieldset>
```

Datos personales

☐ Hombre
☒ Mujer

Estado civil: Casado ☒

Formularios

➤ **Ej9c:** Utiliza la página web anterior e incluye los últimos controles vistos:

- Botones de imagen y genérico
- Listas desplegables con y sin agrupaciones de opciones
- Áreas de texto
- Agrupación de controles relacionados

Datos personales

Escribe tu nombre

Escribe el e-mail

Escribe tu teléfono

Cuestionario

¿Te gusta nuestra web?



Mostrar varias opciones

Cómo valorarías nuestra web?

muy buena

buena

normal

mala

muy mala

Mostrar y elegir varias opciones

Cómo valorarías nuestra web?

muy buena

buena

normal

mala

muy mala

Enviar

Restablecer

Formularios: nuevos controles

■ Novedades en los formularios en Html5: nuevos controles

- ✓ Mejoran la experiencia de los usuarios.
- ✓ Cuadros de texto especializados que **facilitan** al usuario la entrada de datos.
- ✓ En los dispositivos móviles, el teclado que se muestre para rellenar cada campo se **adapta** al tipo de *input*.
- ✓ **Validaciones nativas**: la validación de los campos se realiza **sin** necesidad de **JavaScript** y **antes** de enviar los datos recopilados al servidor.

■ Tipos de validaciones:

- ✓ Campos **obligatorios**: verifica si el campo posee algún valor.
- ✓ Campos de tipo **email**: verifica que los datos introducidos en el campo se corresponden con una dirección de correo válida.
- ✓ Campos de tipo **URL**: verifica que los datos introducidos en el campo se corresponden con una URL.
- ✓ Campos que sólo permiten **números**: no permite la entrada de texto ni de caracteres especiales.
- ✓ Campos que sólo permiten letras/números con un determinado formato: si cumple el **pattern**.

Formularios: nuevos controles

- **Nuevos tipos de `<input>` en Html5:**
- ✓ Facilitan el desarrollo de los formularios al realizar **validaciones** de los datos introducidos en los controles de los formularios.
- a) **email:** para añadir una/varias dirección de correo electrónico con formato válido.
 - Ej: `<input type="email" multiple />`
- b) **url:** para añadir una dirección URL con formato válido.
 - Ej: `<input type="url" />`
- c) **date:** para añadir una fecha con formato válido (ofrecido por el navegador).
 - Ej: `<input type="date" />`
- d) **time:** para añadir una hora en formato 24h válido (ofrecido por el navegador).
 - Ej: `<input type="time" />`
- e) **datetime, datetime-local:** para añadir una fecha-hora con formato válido.
 - Ej: `<input type="datetime" />`
- f) **month:** para elegir un mes, con nº o nombre (ofrecido por el navegador).
 - Ej: `<input type="month" />`
- g) **week:** para elegir una semana, entre 1 y 53 (ofrecido por el navegador).
 - Ej: `<input type="week" />`

Formularios: nuevos controles <input>

- **Nuevos tipos de <input> en Html5:**

h) **number:** para añadir un dato numérico. Con los atributos **min**, **max** y **step** respectivamente, se puede validar que sea un nº entre un valor mínimo y un máximo, con un valor de incremento.

- **Ej:** `<input type="number" min="0" max="20" step="2" />`

i) **range:** muestra un control horizontal deslizante para elegir un valor numérico. Utiliza los atributos **min**, **max** y **step**.

- **Ej:** `<input type="range" min="0" max="100" step="10" />`

j) **tel:** para añadir un nº de teléfono. NO se valida, pero la interface muestra un teclado numérico.

- **Ej:** `<input type="tel" />`

k) **search:** para añadir un término de búsqueda. Como un campo de texto pero con interface mejorada (permite borrar directamente lo escrito -desde la X que se muestra a la derecha del control-). Hay navegadores que muestran de la misma forma los campos de texto.

- **Ej:** `<input type="search" />`

l) **color:** para elegir un color a partir de la paleta de colores que muestra el navegador.

- **Ej:** `<input type="color" />`

Formularios: nuevos controles <input>

- **Ej9d:** Crea una nueva página web que incluya un formulario que contenga los **nuevos controles *input*** ofrecidos en Html5.
- Comprueba el funcionamiento de los diferentes controles desde al menos dos navegadores, incluyendo la **validación** en aquellos controles donde se haga.
- ¿Cómo se muestran los controles desde un navegador si no los reconoce?
- **Ej:**

Utilizando nuevos controles de input en Html5

Hoja de inscripción

Nombre: E-mail:

Teléfono: Url:

Fecha: ☐ Hora: ☐ Fecha y hora:

Mes: de ☐ Semana: ☐

Número: Rango:

Búsqueda: Color:

Formularios: nuevos controles

- **Sol. Ej9d:** los controles nuevos que un navegador no es capaz de identificar los muestra como un cuadro de texto normal, ej, el datetime y el de búsqueda.
- **Ej:** visualización desde *Firefox* y desde *Ópera*

Hoja de inscripción

Nombre: E-mail:

Teléfono: Url:


Fecha: Hora: Fecha y hora:

Mes: Semana:

Número: Rango:

Búsqueda: Color:

Nombre: E-mail:

 Incluye un signo "@" en la dirección de correo electrónico. La dirección "adgfadf" no incluye el signo "@".

Fecha: Hora: Fecha y hora:

Mes: Semana:

Número: Rango:

Formularios: nuevos controles

- **Crear una lista de valores para un `<input>`: `list` y `<datalist>` `</datalist>`:**
- Permite **predefinir** los posibles valores que tendrá un cuadro de texto `<input>`.
- Si no se muestra la opción deseada, se puede escribir un **nuevo** valor.
- Para conseguirlo, hay que utilizar el atributo `list` de `<input>`:
 1. Con el valor que se le asigne a `list` asociamos el cuadro de texto a la lista de valores predefinida.
 2. Dentro del elemento `<datalist>` definimos las opciones que nos interese, utilizando la etiqueta `<option>`
 3. Para asociar el `<input>` con el `<datalist>`, el `id` del `<datalist>` debe tener el mismo valor que el `list` del `<input>`.
- La apariencia es la de un cuadro de texto que ayuda al usuario a **elegir un valor**, ya que al teclear una letra, muestra las opciones que contienen dicha letra.

▪ **Sintaxis:**

```
<input list="identificador" />  
  <datalist id="identificador">  
    <option value="valor1" />  
    <option value=" valor2" />  
    <option value=" valor3"/>  
  </datalist>
```

Formularios: nuevos controles

- Novedades en los formularios en Html5: nuevos controles
- Crear una lista de valores para un `<input>`: **list** y **<datalist>** `</datalist>`:
- Ej:

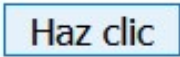
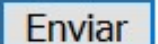
```
Elija un editor de código:<input list="editor" />  
<datalist id="editor">  
  <option value="Atom" />  
  <option value="NotePad++" />  
  <option value="VsCode" />  
  <option value="Sublime" />  
  <option value="Brackets" />  
</datalist>
```

Elija un editor de código:

a
Atom
NotePad++
Brackets

- **Ej9d2:** Añade en la página web anterior un control `<input>` con **<datalist>** y comprueba su funcionamiento.

Formularios: nuevos controles <button>

- **Novedades en los formularios en Html5: nuevos controles**
- **Botón genérico: <button> y </button>**
 - Tiene la apariencia de un botón rectangular.
 - Sirve para capturar su pulsación (mediante JavaScript) y responder en consecuencia.
 - La **recomendación** de **Html5** es utilizar esta etiqueta para crear los diferentes **botones**.
 - El **atributo** más importante es **type**. Los posibles valores son:
 - **button**: para crear un botón genérico.
 - En versiones anteriores se hacía con **<input type="button"/>**
 - **reset**: funciona como el cuadro **input** con **type=reset**.
 - **submit**: funciona como el cuadro **input** con **type=submit**.
- **Ej1:** `<button type="button" >Haz clic</button>` 
- **Ej2:** `<button type="submit" >Enviar</button>` 

Formularios: nuevos atributos en Html5

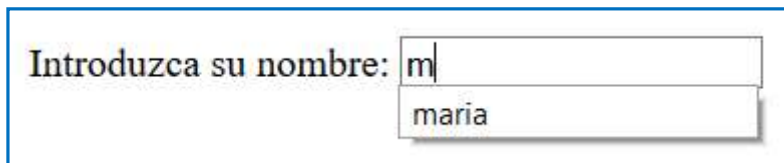
- **Novedades en los formularios en Html5: nuevos atributos de formulario e inputs**
- ✓ **autocomplete**: autocompletar.
- ✓ **autofocus**: pone el foco en el campo deseado cuando se carga el formulario.
- ✓ **form**: para relacionar un control con un formulario.
- ✓ **formaction**: para cambiar el **action** de un formulario.
- ✓ **formmethod**: para cambiar el **method** de un formulario.
- ✓ **formnovalidated**: para no validar ninguno de los controles de un formulario (inputs nuevos con validación).
- ✓ **height y width**: para redimensionar la imagen del botón **image**.
- ✓ **list**: para asociar un **<datalist>** con un input.
- ✓ **min, max, step**: valor mínimo, máximo e incremento (inputs numéricos o fecha).
- ✓ **multiple**: para añadir varios valores.
- ✓ **novalidate**: para no validar un determinado **input** que tenga validación.
- ✓ **placeholder**: valor predeterminado.
- ✓ **required**: campo obligatorio.
- ✓ **pattern**: patrones de validación mediante **expresiones regulares**.

Formularios: nuevos atributos en Html5

■ Novedades en los formularios en Html5: nuevos atributos

✓ **autocomplete**: autocompletar.

- Permite **activar** (valor **on**) o desactivar (valor **off**) el **autocompletado** en ciertos **campos** o en todo el **formulario**.
- Esta opción muestra entradas habituales que se han escrito anteriormente, basándose en la actividad del navegador, ayudando así a recordar o sugerir dichos valores.
- Se puede utilizar en casi todos los tipos de **<input>**:
 - text
 - password
 - En los type nuevos (search, number, url, tel, email, date, ...)
- Conviene **desactivarlo** cuando lo que se escribe son **datos sensibles**: nombre de usuario, contraseñas, ...
- **Ej:** Introduzca su nombre: `<input type="text" name="nombre" autocomplete="on" />`



Introduzca su nombre:

maria

Formularios: nuevos atributos en Html5

- **Novedades en los formularios en Html5: nuevos atributos**

- ✓ **multiple**: varios valores.

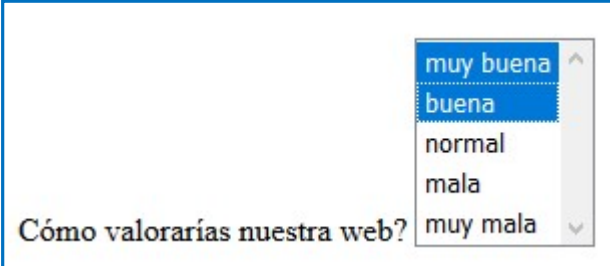
- Permite añadir varios valores en el control en el que se usa.
- Para separar los diferentes valores, utilizar el carácter “,”.
- Se puede utilizar en los **<select>** y en los siguientes tipos de **<input>**:
 - email
 - file

- Ej:

```
<select name="provincia" size="3" multiple>
```

```
<input type="email" name="email" id="email" multiple/>
```

```
<input name="ficheros[]" type="file" multiple />
```



Cómo valorarías nuestra web?

muy buena
buena
normal
mala
muy mala

The image shows a web form with a label "Cómo valorarías nuestra web?". Below the label is a dropdown menu. The dropdown menu is open, showing five options: "muy buena", "buena", "normal", "mala", and "muy mala". The "muy buena" option is currently selected and highlighted in blue. The dropdown menu has a small upward arrow on the left and a small downward arrow on the right.

Formularios: nuevos atributos en Html5

■ Novedades en los formularios en Html5: nuevos atributos

✓ **placeholder**: valor predeterminado.

- Muestra un **texto inicial** dentro de los controles donde se puede escribir texto.
- Dicho texto **desaparece** cuando el control obtiene el foco y el usuario comienza a escribir en él.
- Vuelve a aparecer si se le da al botón de **reset**.
- **Ayuda** a rellenar un cuadro de un formulario, especialmente los cuadros de texto que tengan asociado un *patrón*.
- Se puede utilizar en los `<textarea>` y en los siguientes tipos de `<input>`:
 - text, password
 - search, url, email
 - number, tel

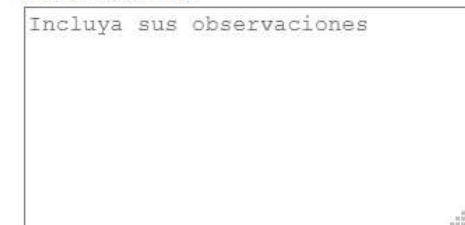
■ **Ej1:** `<input type="text" name="nombre" placeholder="inserta tu nombre" />`



■ **Ej2:** `<label>Observaciones:

 <textarea rows="7" cols="30" name="observaciones"
 placeholder="Incluya sus observaciones"></textarea>
</label>`

Observaciones:



Formularios: nuevos atributos en Html5

- **Novedades en los formularios en Html5: nuevos atributos**

- ✓ **required**: campo obligatorio.

- Obliga a rellenar con algún valor el control en el que se usa.
- Si no se cubre => **error de validación** y se ejecuta la acción por defecto que tenga el navegador.
 - **Ej:** aparece un pequeño globo emergente conteniendo un texto de advertencia.



- Utilizar el atributo **title** para proporcionar ayuda adicional a los usuarios.

- **Ej:**

Introduzca su nombre:

debe introducir un nombre

```
<label>Introduzca su nombre:
  <input type="text" required title="debe introducir un nombre">
</label>
```

Formularios: nuevos atributos en Html5

- **Novedades en los formularios en Html5: nuevos atributos**

- ✓ **pattern**: valida información.

- Su función es validar la información introducida por los usuarios en campos de texto de formulario.
- Se puede utilizar en los siguientes tipos de **<input>**:
 - text, password
 - search, url, email
 - number, tel
- La validación se basa en **expresiones regulares** => el valor del campo debe cumplir un **patrón** que se define mediante **expresiones regulares**.
- Se suele utilizar junto a **required** y **placeholder**
- Ej:

- **Expresiones regulares:**

patrones utilizados para encontrar una determinada combinación de caracteres dentro de una cadena de texto

Formularios: nuevos atributos en Html5

- **Expresiones regulares:**

- Se compone de caracteres simples, como `/abc/`, o una combinación de caracteres simples y especiales, como `/ab*c/` o `/Chapter (\d+)\.\d*/`.
- Los paréntesis se emplean como un recurso de memoria (almacenar la coincidencia encontrada para un uso posterior).

- **Patrones simples**

- Caracteres para los que se desea una **coincidencia exacta**.
- **Ej:** `/abc/` coincidirá sólo con esa secuencia y orden de caracteres.

- **Caracteres especiales:**

- Requiere más que una coincidencia exacta: encontrar una o más 'b', encontrar espacios en blanco, etc.
- **Ej:** `/ab*c/` coincidirá con cualquier secuencia de caracteres en la cual una 'a' preceda a cualquier cantidad de 'b' y le siga inmediatamente una 'c'.

- **Ej:** `<input type="text" pattern="[a-zA-Z]{5}[0-9]{3}" .../>`

➤ **Amplía:** busca documentación sobre expresiones regulares desde https://developer.mozilla.org/es/docs/Web/JavaScript/Guide/Regular_Expressions#special-word-boundary y para **verificarlas** desde <http://www.contadordecaracteres.info/prueba-expresiones-regulares.html> o <http://html5pattern.com/>

Formularios

- Al poco tiempo de salir Html5, este era el nivel de soporte en los principales navegadores:

Browser Support for New HTML5 Input Types

		 Firefox	 Safari	 Safari	 Chrome	 Opera	 IE	 Android
Email	?	4+	5+	3.1+	6+/10+	10.6+	10+	4+
Tel	?	4+	5+	3.1+	6+	10.6+	10+	2.3+
Url	?	4+	5+	3.1+	6+/10+	10.6+	10+	2.3+
Search	?	4+	5+	4+	6+	10.6+	9/10+	4+
Color	?	29+	8+	8-	20+	11+	11-	4.4+
Number	?	29+	5+	3.2+	7+	9+	10+	2.3+
Range	?	23+	4+	5+	6+	11+	10+	4.2+
Date	?	32-	7-	5+	20+	9+	11-	4.4+

- **Averigua** cuál es el soporte actual de los nuevos tipos de controles y atributos de los formularios de Html5 desde la web html5test.com.

Audio y Vídeo

- **Antecedentes sobre los elementos multimedia en las páginas web:**

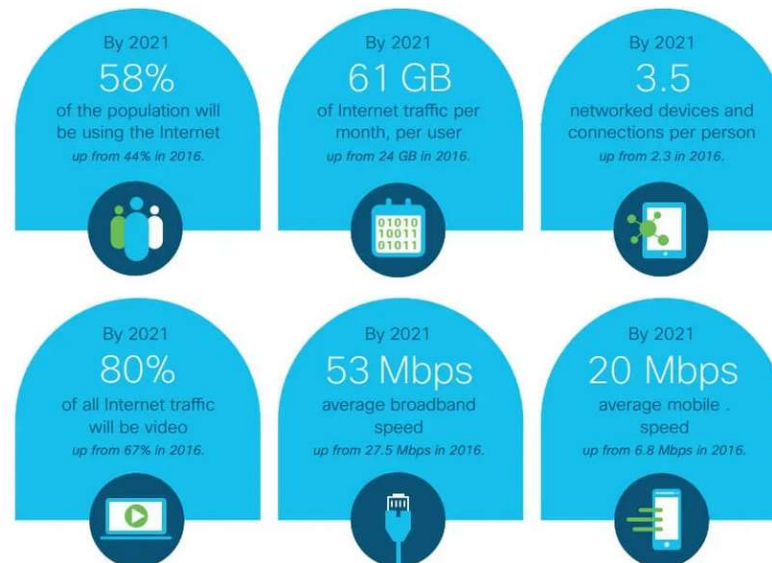
- Cambio de uso de las páginas web:

“De compartir investigaciones científicas a reenviar de forma viral vídeos de gatitos tocando el piano...”

“Según Cisco, se prevee que en 2022 el 82% del tráfico mundial de Internet provenga de la transmisión y descarga de vídeos”

- Hasta Html5 **NO** se ofrecía soporte para audio y vídeo.

➤ **Consulta:** lee los siguientes artículos sobre la evolución y uso de vídeo a través de Internet: <https://www.adslzone.net/2017/06/08/20170608internet-2021/> y desde <https://www.ituser.es/actualidad/2020/05/la-tasa-de-trafico-de-datos-supera-los-91-terabits-decix-analiza-la-evolucion-de-internet>



Audio y Vídeo

▪ Evolución del vídeo en la Web:

1) <embed>

- Se utilizaba para colocar en una página web elementos no pertenecientes al lenguaje HTML: animaciones Flash, vídeo y audio.
- El **navegador** creaba una ventana de vídeo que utilizaba un **reproductor externo** (*Windows Media Player, Apple Quicktime...*).
- **Sin capacidad de controlar la ejecución**, ni de descargar en un **buffer** el vídeo para evitar esperas mientras se descargaba, ni de saber si el vídeo se podía reproducir correctamente en todos los navegadores/s.o.
- Ej: `<embed src="ruta/light.swf" type="application/x-shockwave-flash" width="180" height="350" />`

2) Plug-in de Adobe Flash

- **Garantizaba** que los vídeos Flash **funcionasen** en todos los navegadores donde se hubiese instalado dicho plug-in.
- **Permitía controlar la ejecución**, hacer predescargas, ...

3) A través de un **enlace**.

- El navegador debe buscar si dispone del **plug-in** apropiado para ver el vídeo, en caso de no tenerlo, se descargará en el ordenador.
- Ej: `Ver vídeo`

- Evolución del vídeo en la Web:

- 4) **<object>**

- Sustituye a **<embed>**.
 - Permite incorporar cualquier tipo de contenido externo a una página web.
 - Con **<param>** se pueden especificar los parámetros relacionados con los plug-in de los navegadores encargados de mostrar el objeto, facilitando las **instrucciones sobre la reproducción**.

- Ej:

```
<object data="selva.wmv" type="video/x-ms-wmv" width="300">  
  <param name="autoplay" value="true" />  
</object>
```

- Por problemas de compatibilidad entre navegadores, se recomienda seguir utilizando **<embed>** y duplicar ciertos parámetros.

- Ej:

```
<object data="ruta/luz.swf" type="application/x-shockwave-flash" width="180" height="350">  
  <param name="movie" value="ruta/luz.swf" />  
  <param name="quality" value="high"/>  
  <param name="wmode" value="transparent"/>  
  <embed src="ruta/luz.swf" type="application/x-shockwave-flash"  
    allowscriptaccess="always" allowfullscreen="true" width="180" height="350" />  
</object>
```


Audio y Vídeo: Evolución

5) <iframe>

- Se utiliza para **mostrar contenidos externos** al sitio web como si fueran parte del mismo sitio => **incrusta contenido**, ya que coloca una página html dentro de otra.
- Elemento casi desaparecido, que se ha recuperado en el estándar de HTML5.
- Popularidad actual gracias a *youtube*.
- **Algunos Atributos:**
 - ✓ **src**: URL al recurso que se desea mostrar.
 - ✓ **height** y **width**: altura y anchura de la ventana que mostrará el recurso.
 - ✓ **scrolling**: para mostrar o no barras de scroll en la ventana del iframe.
- **Sintaxis:**

```
<iframe src="ruta completa/recurso" width="450" height="450" scrolling="yes" />
```

- **Ej:**

```
<iframe src="https://belennadales.com/pequena-historia-de-la-musica-fernando-argenta"  
width="450" height="450" scrolling="yes" />
```

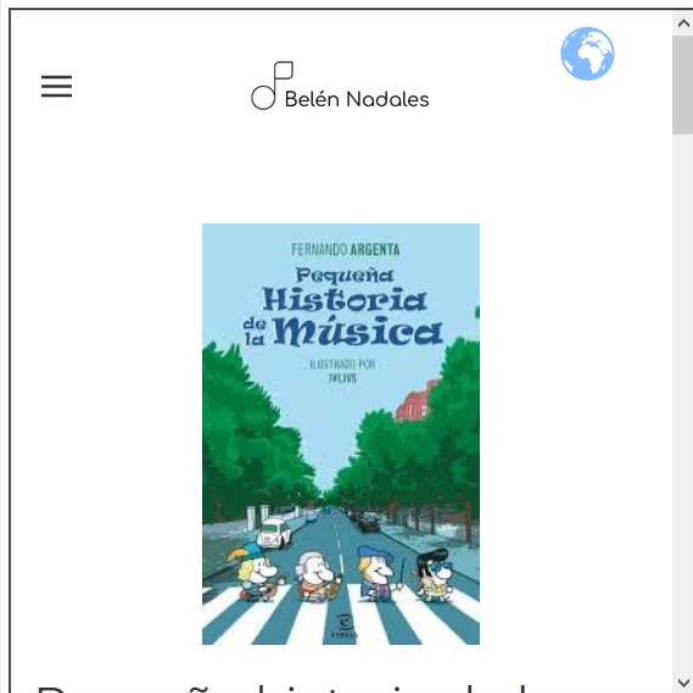
Audio y Vídeo: Evolución

5) <iframe>

- La forma más sencilla para insertar un vídeo de *youtube* en una página web es **copiando** la **url** que se genera automáticamente utilizando la opción de **compartir**.
- Ej:

```
<iframe width="560" height="315" src="https://www.youtube.com/embed/EZiiSgcCxAO" frameborder="0" allow="accelerometer; autoplay; clipboard-write; encrypted-media; gyroscope; picture-in-picture" allowfullscreen></iframe>
```

Añadiendo contenido externo a la página web: iframe



Insertar un vídeo de youtube gracias a la etiqueta Iframe

Censura de redes sociales

EL BLOQUEO DE LAS REDES SOCIALES A DONALD TRUMP: Censura o límites a la libertad de expresión



Audio y Vídeo antes de Html5

- **Antecedentes sobre elementos multimedia:**
 - **Problemática existente para añadir elementos multimedia:**
 - ✓ **No** hay un **códec standard**: numerosos formatos de vídeo y audio.
 - ✓ Cada navegador tiene distintas capacidades.
 - ✓ Necesario instalar plug-in (extensiones) para poder ver vídeos y oír música
 - ✓ **Internet = caos** desde el punto de vista de un desarrollador web.
 - **2005**: aparición de **youtube**
 - ✓ Plataforma destinada a contener vídeos.
 - ✓ Al alojar los vídeos en esa plataforma, ya no se necesita configurar servidores de vídeo propios.
 - **Ventajas:**
 - ✓ Todos los usuarios acceden a esa plataforma.
 - ✓ Ahorro de espacio.
 - ✓ Ahorro de tiempo en implementaciones de códecs.
 - **Inconvenientes:**
 - ✓ Funcionaba con **Flash Player**
 - ✓ Cumplir condiciones legales para publicar: ¿**censura** de contenido?.

Audio y Vídeo en Html5

- **Incorporación de elementos multimedia en Html5:**

- Se crearon nuevas etiquetas a partir de una **idea** simple:

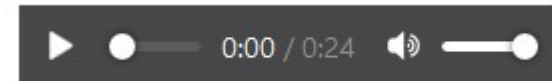
*Igual que puedes añadir imágenes a una página web con el elemento ****, deberías poder insertar sonido con un elemento **<audio>** y vídeo con un elemento **<video>***

*... y **HTML5** los creó...*

- El código es **legible** por cualquier dispositivo: móvil, ipad, tablet android, ordenador, etc.
- Elementos totalmente **integrados** en la web.
- Pertenecen al estándar de **HTML5**, no dependiendo de software de terceros (reproductores externos).

- **Ej:**

```
<audio src="cancion.mp3" controls></audio>
```



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



Vídeo en Html5

- **<video>** y **</video>**:
 - Sólo disponible a partir de **HTML5**.
 - **Atributos**:
 - ✓ **src**: URL al vídeo que se desea mostrar.
 - ✓ **height** y **width**: altura y anchura de la ventana que mostrará el vídeo en la página.
 - ✓ **autoplay**: valor fijo **autoplay** para indicar que el vídeo se inicia automáticamente en cuanto se descargue.
 - ✓ **loop**: valor fijo **loop** para indicar que el vídeo se reproduzca automáticamente una y otra vez.
 - ✓ **controls**: valor **controls** para que el vídeo muestre la barra de reproducción con los controles *-pausa, play,...-*.
 - ✓ **preload**: indicación sobre cómo debemos realizar la descarga:
 - **auto**: el vídeo se descarga en cuanto se carga la página.
 - **none**: el vídeo no se descarga. Cuando el usuario pulse *play*, se descargará.
 - ✓ **poster**: permite indicar la dirección URL a una imagen que se mostrará mientras el vídeo no se está reproduciendo.
 - Si no se usa este atributo, se usa el primer fotograma del vídeo como póster.

Vídeo en Html5

- **<video>** y **</video>**:
 - **Codecs de vídeo:**
 - **Códec:** software que se usa para crear los vídeos.
 - 1) Un vídeo actual usa un formato **comprimido** de datos, por lo que para crearlos se **codifican** los datos y para mostrarlo, se **decodifica**.
 - 2) Para ver un vídeo en un formato concreto, se necesita el **códec apropiado**.
 - 3) Cada **navegador** incorpora unos **codecs diferentes**.
 - **Solución:** incorporar el elemento **<source>** para dar soporte a diferentes formatos
 - 1) Codificar el vídeo en **varios formatos**.
 - 2) Referenciarlos dentro de la etiqueta **<video>**
 - 3) El navegador usará el formato que sea capaz de traducir (del que disponga el códec adecuado).
 - **Atributos:**
 - ✓ **src:** URL al vídeo que se desea mostrar.
 - ✓ **type:** tipo MIME del vídeo.
 - **opcional:** indicar los **codecs** concretos que necesitamos para decodificar el vídeo

Vídeo en Html5

- **Sintaxis:** `<source src="video.mp4" type="video/mp4" />`

- **Ej:**

```
<video autoplay="autoplay" controls="controls" poster="foto1.jpg" >  
  <source src="video.mp4" type="video/mp4;codecs='avc1.42E01E,  
  mp40a.40.2' " />  
  <source src="video.ogv" type="video/ogg;codecs='theora, vorbis' " />  
  El navegador no puede mostrar el vídeo  
</video>
```

- **Funcionamiento:**

- 1) Si el primer formato no se reconoce (primer elemento del **source**), se intenta con el segundo y así sucesivamente.
- 2) Si ningún formato es reproducible por el navegador actual, éste mostrará la frase final tras el último source.



Vídeo en Html5

- **Formatos de archivos multimedia:**
- El standard HTML5 no requiere soporte para ningún formato específico de vídeo.
- Cada navegador elige los formatos que quieren soportar.
 - **Formatos estándar de vídeo:**
 - Soportados por los navegadores Html5.
 - **Extensiones recomendadas:**

FORMAT	DESCRIPTION	COMMON FILE EXTENSION	MIME TYPE
H.264	The industry standard for video encoding, particularly when dealing with high-definition video. Used by consumer devices (like Blu-ray players and camcorders), web sharing websites (like YouTube and Vimeo), and web plug-ins (like Flash and Silverlight).	.mp4	video/mp4
Ogg Theora	A free, open standard for video by the creators of the Vorbis audio standard. Byte for byte, the quality and performance doesn't match H.264, although it's still good enough to satisfy most people.	.ogv	video/ogg
WebM	The newest video format, created when Google purchased VP8 and transformed it into a free standard. Critics argue that the quality isn't up to the level of H.264 video—yet—and that it may have unexpected links to other people's patents, which could lead to a storm of lawsuits in the future.	.webm	video/webm

Audio en Html5

- **<audio>** y **</audio>**:
 - Sólo disponible a partir de **HTML5**.
 - Elemento **comparable** al de **<video>**: permite incorporar audio a una página web.
 - **Atributos**:
 - ✓ **src**: URL al audio que se desea escuchar.
 - ✓ **autoplay**: valor fijo **autoplay** para indicar que el audio se inicia automáticamente en cuanto se descargue.
 - ✓ **loop**: valor fijo **loop** para indicar que el audio se reproduzca automáticamente una y otra vez.
 - ✓ **controls**: valor **controls** para que el audio muestre la barra de reproducción con los controles *-pausa, play,...-*.
 - ✓ **preload**: indicación sobre cómo debemos realizar la descarga:
 - **auto**: el audio se descarga en cuanto se carga la página.
 - **none**: el audio no se descarga. Cuando el usuario pulse *play*, se descargará.
 - Mismo **problema** de **codecs** y **formatos**.
 - **Solución**: convertir el audio a distintos formatos y dar opciones dentro de la etiqueta **<audio>** gracias a la etiqueta **<source>**.

Audio en Html5

- **<audio>** y **</audio>**:

- **Sintaxis:**

```
<audio controls autoplay>
```

```
<source src="audio1.ogg" type="audio/ogg">
```

```
<source src="audio2.mp3" type="audio/mpeg">
```

No se puede reproducir el archivo de audio

```
</audio>
```

- **Ej:**

Audio

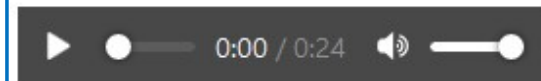


El navegador no puede reproducir el audio

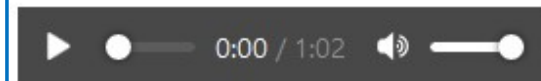
Audio



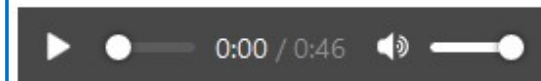
Presentación: Infinito



Título: Me lo como



Título: Relax



Audio en Html5

- **Formatos de archivos multimedia:**
- El standard HTML5 no requiere soporte para ningún formato específico de vídeo.
- Cada navegador elige los formatos que quieren soportar.
 - **Formatos estándar de audio:**
 - Soportados por los navegadores Html5.
 - **Extensiones recomendadas:**

FORMAT	DESCRIPTION	COMMON FILE EXTENSION	MIME TYPE
MP3	The world's most popular audio format.	.mp3	audio/mp3
Ogg Vorbis	A free, open standard that offers high-quality, compressed audio comparable to MP3.	.ogg	audio/ogg
WAV	The original format for raw digital audio. Doesn't use compression, so files are staggeringly big and unsuitable for most web uses.	.wav	audio/wav

Audio y Vídeo en Html5

- Al poco tiempo de salir Html5, este era el nivel de compatibilidad entre formatos de archivos multimedia y los navegadores:

TABLE 5-2 *Browser support for HTML5 audio formats*

	IE	FIREFOX	CHROME	SAFARI	OPERA	SAFARI IOS	ANDROID
MP3	9	21	5	3.1	-	3	2.3
Ogg Vorbis	-	3.6	5	-	10.5	-	-
WAV	-	3.6	8	3.1	10.5	-	-

TABLE 5-3 *Browser support for HTML5 video formats*

	IE	FIREFOX	CHROME	SAFARI	OPERA	SAFARI IOS	ANDROID
H.264 Video	9	21	5	3.1	-	4*	2.3
Ogg Theora	-	3.5	5	-	10.5	-	-
WebM	-	4	6	-	10.6	-	2.3

- **Averigua** cuál es el nivel de compatibilidad actual entre formatos de archivos multimedia y las últimas versiones de los navegadores.