

# RESUMEN HTML5

Por Javier Peiro Aguado (2DAW)

## 1- Introducción

**HTML 5** es la quinta revisión importante del lenguaje HTML, diseñada para mejorar el desarrollo de aplicaciones web. Introduce nuevas características y establece criterios claros de conformidad para los navegadores, buscando mejorar la interoperabilidad.

## 2-DOCTYPE

La declaración DOCTYPE (Document Type Declaration) se tiene que colocar en la primera línea del archivo HTML, antes de la etiqueta `<html>`.

En **HTML 5**, la declaración se simplifica a:

```
<!DOCTYPE HTML>
```

## 3-AUDIO

El elemento `<audio>` en HTML5 permite incluir archivos de audio en una página web sin necesidad de complementos como Flash o Java. Algunos de los formatos de audio soportados dependen del navegador que usas, como mp3, wav, ogg, o au.

Ej:

```
<audio src="sonido.ogg" autoplay controls loop></audio>
```

Propiedades de `<audio>`:

- **src**: URL del archivo de audio.
- **autoplay**: El audio se reproduce automáticamente al cargar la página.
- **loop**: El audio se repite una vez que termina.
- **controls**: Muestra los controles para reproducir, pausar y ajustar el volumen.
- **autobuffer**: El archivo se descarga antes de comenzar a reproducirse.

Además, podemos agregar varias fuentes de audio para asegurar la compatibilidad con diferentes navegadores:

```
<audio controls autoplay loop>
  <source src="sonido.ogg">
  <source src="sonido.mp3">
</audio>
```

## 4-VIDEO

El elemento <video> en HTML5 permite incluir videos en una página web sin necesitar Flash. Los formatos más comunes soportados por los navegadores son mp4, webm y ogv.

Propiedades de <video>:

- **src**: Dirección del archivo de video.
- **controls**: Muestra los controles del video (play, pausa, barra de avance).
- **autoplay**: El video comienza automáticamente al cargar la página.
- **width**: Ancho del video en píxeles.
- **height**: Alto del video en píxeles.

Al igual que <audio>, se pueden agregar varias fuentes de video para asegurar la compatibilidad con diferentes navegadores:

```
<video width="640" height="360" controls>
  <source src="http://videos.mozilla.org/firefox/3.5/overview/overview.ogv">
  <source src="http://videos.mozilla.org/firefox/3.5/overview/overview.mp4">
</video>
```

## 5-Elementos HTML semánticos

HTML5 introduce varios elementos semánticos que mejoran el desarrollo de páginas web y su análisis en los motores de búsqueda. Estos elementos ayudan a estructurar el contenido de manera clara y significativa.

Estos son los más importantes:

- **<header>**: Marca la cabecera de una página, que puede incluir el logotipo, imágenes, y elementos de búsqueda. También puede anidarse dentro de otras secciones.
- **<nav>**: Representa una sección de navegación, que agrupa enlaces importantes que dirigen a otras partes del sitio o páginas externas.

- **<section>**: Se utiliza para representar una sección temática de un documento, como un capítulo o una subsección de contenido, con su propio encabezado.
- **<footer>**: Define el pie de página, que contiene información como el autor, derechos de autor, y enlaces relacionados.
- **<aside>**: Representa contenido relacionado de forma tangencial con el contenido principal, como barras laterales o notas explicativas.
- **<article>**: Define una entrada independiente dentro de un sitio, como una entrada de blog, artículo de revista, o una noticia. Puede contener contenido anidado, como los comentarios de los usuarios en un blog.

Estos elementos semánticos facilitan la comprensión y organización de la web.

## 6-FORM (Autofocus)

En HTML5, la propiedad autofocus se utiliza en los formularios para especificar qué control tendrá el foco (donde aparecerá el cursor) cuando se cargue la página. Antes, esto se hacía con JavaScript, pero con HTML5 se puede hacer directamente.

Para usar esta propiedad, simplemente ponemos **autofocus** al control de formulario sin asignarle un valor.

Ej:

```
<input type="text" id="edad" autofocus>
```

## **7-FORM (placeholder)**

La propiedad placeholder en los formularios HTML muestra un texto dentro de un campo de entrada como guía para el usuario sobre qué dato debe ingresar. Este texto desaparece cuando el usuario empieza a escribir. Se debe usar solo cuando ayude a aclarar qué se debe ingresar.

```
<input type="text" placeholder="Introduce aquí tu nombre">
```

## **8-FORM (required)**

Required en HTML5 permite marcar un campo de formulario como obligatorio. Al agregarla al control, el navegador valida automáticamente que el campo esté lleno antes de enviarlo al servidor, sin necesidad de utilizar JavaScript.

## **9-FORM (pattern)**

Pattern en HTML5 permite validar que el contenido de un formulario siga un patrón específico, utilizando expresiones regulares. Esta propiedad es útil para asegurarse de que los datos ingresados (como un correo electrónico) tengan una estructura adecuada, sin necesidad de usar JavaScript.

Ej:

```
<input type="text" id="numero" required pattern="[01]{8}">
```

## 10-FORM (input type="email")

`<input type="email">` se utiliza solamente para ingresar direcciones de correo electrónico. Este tipo de control ya tiene incorporada la validación de la estructura de un email, lo que elimina la necesidad de JavaScript para verificar el formato.

## 11-FORM (input type="range")

`<input type="range">` permite seleccionar un valor dentro de un rango mediante una barra deslizante. Se pueden definir las propiedades min y max para establecer los valores mínimo y máximo seleccionables.

## 12-FORM (input type="date/datetime-local/month/time,week")

HTML5 tiene algunos tipos de controles de entrada para manejar fechas y horas de forma más sencilla:

1. **`<input type="date">`**: Permite seleccionar una fecha mediante un calendario.
2. **`<input type="datetime-local">`**: Permite ingresar tanto fecha como hora.
3. **`<input type="month">`**: Facilita la selección de un mes y un año.
4. **`<input type="time">`**: Permite ingresar solo una hora.
5. **`<input type="week">`**: Permite seleccionar una semana del año.

Cada navegador puede tener una interfaz diferente para estos controles, pero todos proporcionan una forma de ingresar datos de fecha y hora sin necesidad de JavaScript.

## 13-FORM (input type="color")

El control `<input type="color">` en HTML5 permite que los usuarios seleccionen un color desde un cuadro de diálogo.

Ej:

```
<form action="#">  
<p>Seleccione un color:  
<input type="color" id="color1" name="color1">
```

```
</p>
<p><input type="submit" value="Confirmar"></p>
</form>
```

Este control se muestra en los navegadores como un rectángulo de color, y al hacer clic se abre un cuadro de diálogo para elegir el color. Si deseas establecer un valor predeterminado, puedes usar el atributo value, como se muestra a continuación con el color rojo:

```
<input type="color" id="color1" name="color1" value="#ff0000">
```

## 14-FORM (input type="number")

<input type="number"> en HTML5 permite ingresar un valor numérico. Además, puedes especificar los valores mínimo y máximo que el usuario puede introducir utilizando los atributos min y max.

Ej:

```
<input type="number" id="valor" name="valor" min="1" max="10">
```

En este caso, el usuario solo puede ingresar un valor entre 1 y 10.

## 15-FORM (input type="url")

<input type="url"> en HTML5 se utiliza para introducir una URL. Valida que el formato de la dirección web sea correcto, y si el usuario introduce una URL incorrecta, el navegador muestra un mensaje de error.

Ej:

```
<input type="url" id="sitio" name="sitio" placeholder="https://ejemplo.com">
```

## 16-FORM (input-datalist)

El elemento <datalist> permite proporcionar una lista de sugerencias al usuario en un campo de texto. Este elemento se asocia al control <input> mediante el atributo list, y proporciona opciones que el usuario puede seleccionar sin tener que escribir todo el texto.

Ej:

```
<form action="#">
  Ingresa el nombre de navegador:
  <input type="text" id="navegador" list="listanavegadores">
  <datalist id="listanavegadores">
    <option label="Chrome" value="Chrome">
    <option label="Firefox" value="Firefox">
  </datalist>
  <br>
  <input type="submit" value="Confirmar">
</form>
```

## 17-FORM (input novalidate)

Novalidate se utiliza en el formulario <form> para desactivar todas las validaciones del cliente, como las de los atributos required, pattern o validate.

Esta propiedad es útil cuando queremos hacer una validación del lado del servidor sin la interferencia de las validaciones previas en el cliente, lo que permite depurar los datos antes de que se envíen al servidor.

Ej:

```
<form action="#" novalidate>
  <!-- Campos del formulario -->
</form>
```