

# Tema 02 - HTML (Formato de texto y Listas)

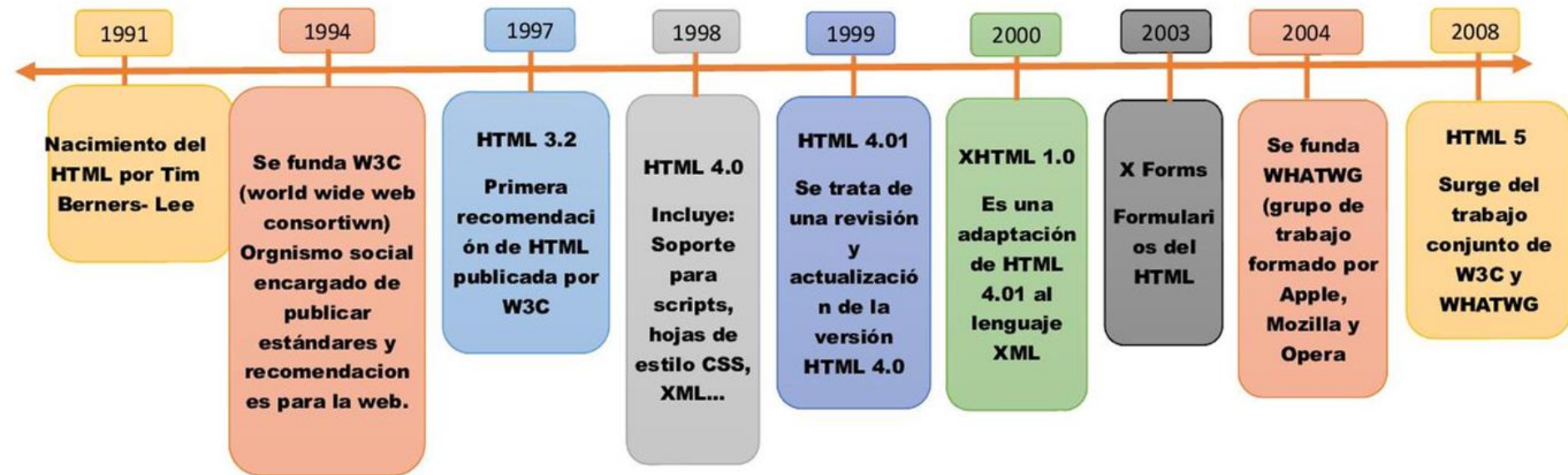
Parte 1

# 1. Evolución de HTML

HTML es el lenguaje utilizado para crear la mayor parte de las páginas web. Es un estándar reconocido en todos los navegadores, por lo tanto, todos ellos visualizan una página HTML de forma muy similar independientemente del sistema operativo sobre el que se ejecutan.



# 1. Evolución de HTML



# 1. Evolución de HTML

- **El origen de HTML** fue un sistema de hipertexto para compartir documentos electrónicos en 1980. La primera propuesta oficial para convertir HTML en un estándar se realizó en 1993. Aunque ninguna de las dos propuestas de estándar que se hicieron consiguieron convertirse en estándar oficial.
- **HTML 2.0** fue la primera versión oficial de HTML el IETF publicó el estándar en septiembre de 1995.
- **HTML 3.2** se publicó el 14 de Enero de 1997 por el W3C. Incorpora los applets de Java y texto alrededor de las imágenes.
- **HTML 4.0** se publicó el 24 de Abril de 1998. Entre las novedades que presenta se encuentran las hojas de estilos CSS y la posibilidad de incluir pequeños programas en las páginas web.

# 1. Evolución de HTML

- **HTML 4.01** es la última especificación oficial de HTML se publicó el 24 de diciembre de 1999. Es una actualización de la versión anterior. En ese momento el W3C detuvo la actividad de estandarización de HTML hasta marzo de 2007, momento en que se retoma debido a la fuerza de las empresas que forman el grupo WHATWG y a la publicación de los borradores de HTML 5.0, que será la siguiente versión de este lenguaje.

Tras la publicación del estándar HTML 4.01 se detecta su incompatibilidad con herramientas basadas en XML. Para evitar estos problemas se crea lenguaje XHTML que combina la sintaxis de HTML 4.0 con la de XML.

- **XHTML 1.0** fue la primera versión, se publicó el 26 de Enero de 2000. Es una adaptación de HTML 4.01 al lenguaje XML, por lo que mantiene sus características, y añade algunas restricciones y elementos de XML.
- La versión **XHTML 1.1** ya ha sido publicada en forma de borrador y pretende modularizar XHTML.
- El borrador de **XHTML 2.0** ya ha sido publicado, que presenta grandes novedades respecto de las anteriores versiones.

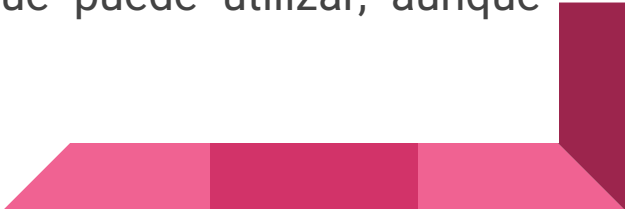
## 2. Estructura de un documento HTML

Un documento HTML está formado por etiquetas y atributos.

Aunque HTML define una gran cantidad de etiquetas, estas no son suficientes para crear páginas complejas ya que la definición completa de ciertos elementos, como las imágenes y los enlaces, requiere información adicional. Como no es posible crear una etiqueta por cada elemento diferente, se añade la información adicional a las etiquetas mediante los atributos dando lugar a los elementos.

Para cada uno de los atributos hay definido un conjunto de valores que se le puede asignar, si el valor de un atributo no es válido, el navegador le ignora.

Cada una de las etiquetas HTML define los atributos que puede utilizar, aunque algunos de ellos son comunes a muchas etiquetas.



### 3. Etiquetas estructurales principales

Son aquellas que crean la estructura para el documento (la página web):

- “**<!DOCTYPE html>**”: esta etiqueta define el tipo de documento y es obligatoria. Se encarga de indicar al navegador qué tipo de lenguaje va a tratar.
- **<html> y </html>**: Todo el documento tiene que ir entre estas etiquetas de apertura y de cierre. Todo lo que esté fuera de estas etiquetas no se tendrá en cuenta. Todo documento html tiene dos partes: head y body.
- **El head** se declara como **<head> y </head>** y dentro puede incluir elementos estructurales de la página como **<title>** (que indica el título de la página).
- **El body** se declara como **<body> y </body>** y en su interior se encuentra el contenido real de toda la página.

### 3. Etiquetas estructurales principales

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <meta charset="utf-8">
5      <title></title>
6  </head>
7  <body>
8
9  </body>
10 </html>
```



## 4. Etiquetas para dar formato al texto

- Las etiquetas `<h1>`, `<h2>`, `<h3>`, `<h4>`, `<h5>` y `<h6>`: sirven para indicar encabezados de la página web. `<h1>` se utiliza para títulos de primer nivel, `<h2>` para títulos de segundo nivel... y así sucesivamente.

```
<div>  
  <h1>Encabezado 1 IEDA</h1>  
  <h2>Encabezado 2 IEDA</h2>  
  <h3>Encabezado 3 IEDA</h3>  
  <h4>Encabezado 4 IEDA</h4>  
  <h5>Encabezado 5 IEDA</h5>  
  <h6>Encabezado 6 IEDA</h6>  
</div>
```



**Encabezado 1 IEDA**

**Encabezado 2 IEDA**

**Encabezado 3 IEDA**

**Encabezado 4 IEDA**

**Encabezado 5 IEDA**

**Encabezado 6 IEDA**

## 4. Etiquetas para dar formato al texto

- La etiqueta **<b>**: Pone el texto en negrita.
- La etiqueta **<i>**: Pone el texto en cursiva.
- La etiqueta **<u>**: Subraya el texto.
- La etiqueta **<del>**: Tacha el texto.
- La etiqueta **<tt>**: imita el estilo de una máquina de escribir.
- La etiqueta **<mark>**: Subraya el texto como si fuera un subrayador.
- Las etiquetas **<sup>** y **<sub>**: sirven para crear superíndices y subíndices (principalmente utilizadas en matemáticas).
- La etiqueta **<p>**: Sirve para indicar el comienzo y el final de un párrafo.
- La etiqueta **<br>**: es un salto de línea.

Además, todas estas etiquetas se pueden anidar unas con otras para así conseguir texto en negrita y subrayado, cursiva y negrita...

## 5. Creación de listas

Tenemos tres tipos de listas:

### Listas no ordenadas

```
<p>países del cono sur:</p>
<ul>
  <li>Argentina</li>
  <li>Chile</li>
  <li>Paraguay</li>
  <li>Uruguay</li>
</ul>
```

Se utilizan cuando el orden de los elementos no tiene importancia.

- La lista se crea utilizando las etiquetas `<ul>` y `</ul>`.
- Cada uno de los elementos de esta lista se designa con `<li>` y `</li>`.
- Además, podemos hacer que la lista esté definida por el atributo `type` con:

**disc:** Es el valor por defecto. Consiste en un pequeño círculo relleno.

**circle:** Consiste en un pequeño círculo con el interior vacío

**square:** Consiste en un pequeño cuadrado.

**none:** Este valor elimina la viñeta, dejando la lista sin viñetas.

## 5. Creación de listas

### Listas ordenadas

```
<p>Principales ciudades de España</p>
<ol>
  <li>Madrid</li>
  <li>Barcelona</li>
  <li>Valencia</li>
  <li>Sevilla</li>
</ol>
```

Son muy parecidas a las anteriores pero, en este caso, el orden de los elementos sí importa.

- La lista se crea utilizando las etiquetas `<ol>` y `</ol>`.
- Cada elemento también se designa con `<li>` pero, en este caso, el type puede ser:

**type="1"**: listas con números: (1, 2, 3, ...)

**type="a"**: listas con letras minúsculas: (a, b, c, ...)

**type="A"**: listas con letras mayúsculas: (A, B, C, ...)

**type="I"**: listas con números romanos en mayúsculas: (I, II, III, ...)

**type="i"**: listas con números romanos en minúsculas: (i, ii, iii, ...)

## 5. Creación de listas

### Listas ordenadas

#### *Modificar el principio de la lista*

En ocasiones nos interesa que el número del principio de la lista no sea el uno (1), sino empezar por otro número. Para ello usaremos el atributo `start="num"`. El valor de este atributo será siempre un número: el número por el que empieza la lista. Aunque la lista esté en letras o en números romanos, pondremos siempre el número correspondiente, y el programa ya se encargará de traducirlo al que le corresponda.

```
<ol type "A" start="5" >  
  <li>Zaragoza</li>  
  <li>Bilbao</li>  
  <li>Málaga</li>  
  <li>La Coruña</li>  
</ol>
```



E. Zaragoza  
F. Bilbao  
G. Málaga  
H. La Coruña

## 5. Creación de listas

### Listas de definición

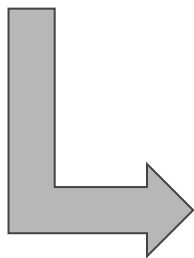
Además de las listas no ordenadas y listas ordenadas en HTML tenemos también las listas de definición.

- Las listas de definición son aquellas en que los términos aparecen como en las entradas de un diccionario. Toda la lista va encerrada en la etiqueta `<dl> ...</dl>`.
- Los elementos de la lista pueden ser de dos clases: **términos y definiciones**.
- Los **términos** van delimitados por la etiqueta `<dt> ... </dt>` y las **definiciones** van delimitadas por la etiqueta `<dd> ... </dd>`.

## 5. Creación de listas

### Listas de definición

```
<p>Relieves de las costas</p>
<dl>
  <dt>Cabo</dt>
  <dd>Sector de costa que se adentra en el mar</dd>
  <dt>Golfo</dt>
  <dd>Sector del mar que se adentra en la costa</dd>
  <dt>Playa</dt>
  <dd>Costa baja formada por una acumulación de material sedimentario de arenas y gravas.</dd>
  <dt>Acantilado</dt>
  <dd>Escarpe litoral modelado por la acción erosiva del oleaje y de las corrientes.</dd>
</dl>
```



- Relieves de las costas
- Cabo
    - Sector de costa que se adentra en el mar
  - Golfo
    - Sector del mar que se adentra en la costa
  - Playa
    - Costa baja formada por una acumulación de material sedimentario de arenas y gravas.
  - Acantilado
    - Escarpe litoral modelado por la acción erosiva del oleaje y de las corrientes.

## 5. Creación de listas

### Listas anidadas

Si incluimos una lista dentro de otra, tendremos unas listas anidadas. La lista que está dentro se mostrará desplazada a la derecha respecto a la que la contiene.

```
<p>Regiones y Provincias del norte</p>
<ul>
  <li>Galicia</li>
  <ul>
    <li>La coruña/A Coruña</li>
    <li>Lugo</li>
    <li>Orense/Ourense</li>
    <li>Pontevedra</li>
  </ul>
  <li>Asturias</li>
  <li>Cantabria</li>
  <li>Pais Vasco/Euskadi</li>
  <ol>
    <li>Vizcaya/Bizkaia</li>
    <li>Guipuzcoa/Guipuzkoa</li>
    <li>Alava/Araba</li>
  </ol>
</ul>
```

- Regiones y Provincias del norte
- Galicia
    - La coruña /A Coruña
    - Lugo
    - Orense /Ourense
    - Pontevedra
  - Asturias
  - Cantabria
  - Pais Vasco/Euskadi
    1. Vizcaya /Bizkaia
    2. Guipuzcoa /Guipuzkoa
    3. Alava /Araba



# Tema 02 - HTML (Tablas)


Parte 2

# 1. Creación de tablas

Una tabla no es otra cosa más que un medio de organizar datos en filas y columnas.

En documentos HTML una tabla puede ser considerada, resumidamente, como **un grupo de filas donde cada una contiene a un grupo de celdas**. Esto es conceptualmente distinto a un grupo de columnas que contiene a un grupo de filas, y esta diferencia tendrá un impacto en la composición y comportamiento de la tabla.

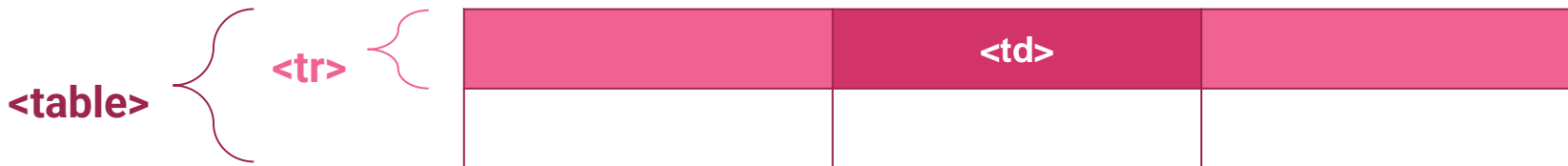
Como muchas otras estructuras de HTML, las tablas son construidas utilizando elementos.



# 1. Creación de tablas

## Elementos de una tabla

- **<table></table>**: Sirve para declararse la tabla, es el contenedor principal.
- **<caption>.</caption>**: se emplea para insertar un título a la tabla. Debe colocarse justo después de la etiqueta <table>, y antes de la primera etiqueta <tr>
- **<tr></tr>**: filas contenedoras de las celdas.
- **<td></td>**: celdas dentro de una fila.



# 1. Creación de tablas

## ¿Cómo se crea una tabla?

1. El contenido de la tabla debe estar incluido entre las etiquetas **<table>** ... **</table>**
2. Cada fila debe estar incluida entre las etiquetas **<tr>** ...**</tr>**
3. Dentro de la etiqueta de fila (**<tr>** ...**</tr>**) incluiremos las etiquetas de celda: **<td>** (**contenido de la celda**)**</td>**. por cada celda que haya en la fila incluiremos una etiqueta **<td>** ...**</td>**, incluso si la celda está vacía.
4. Debemos poner en todas las filas (Etiquetas **<tr>**) el mismo número de celdas (etiquetas **<td>**) para que la tabla quede de forma homogénea.

# 1. Creación de tablas

```
<table>
  <tr>
    <td>Fila 1, columna 1</td>
    <td>Fila 1, columna 2</td>
    <td>Fila 1, columna 3</td>
  </tr>
  <tr>
    <td>F 2, C 1</td>
    <td>F 2, C 2</td>
    <td>F 2, C 3</td>
  </tr>
  <tr>
    <td>F 3, C 1</td>
    <td>F 3, C 2</td>
    <td>F 3, C 3</td>
  </tr>
</table>
```



Fila 1, Columna 1	Fila 1, Columna 2	Fila 1, Columna 3
F 2, C 1	F 2, C 2	F 2, C 3
F 3, C 1	F 3, C 2	F 3, C 3

# 1. Creación de tablas

## Celdas de encabezado <th></th>

Se emplea igual que la etiqueta <td> ... </td> y sustituye a ésta para indicar que una celda es cabecera de tabla.

```
<table>
  <caption>TABLA DE EJEMPLO</caption>
  <tr>
    <td></td>
    <th>COLUMNA 1</th>
    <th>COLUMNA 2</th>
    <th>COLUMNA 3</th>
  </tr>
  <tr>
    <th>FILA 1</th>
    <td>F 1, C 1</td>
    <td>F 1, C 2</td>
    <td>F 1, C 3</td>
  </tr>
  <tr>
    <th>FILA 2</th>
    <td>F 2, C 1</td>
    <td>F 2, C 2</td>
    <td>F 2, C 3</td>
  </tr>
  <tr>
    <th>FILA 3</th>
    <td>F 3, C 1</td>
    <td>F 3, C 2</td>
    <td>F 3, C 3</td>
  </tr>
</table>
```



	COLUMNA 1	COLUMNA 2	COLUMNA 3
FILA 1	F 1, C 1	F 1, C 2	F 1, C 3
FILA 2	F 2, C 1	F 2, C 2	F 2, C 3
FILA 3	F 3, C 1	F 3, C 2	F 3, C 3

# 1. Creación de tablas

## Bordes

Las celdas de la tabla aparecen por defecto sin un borde definido. esto se puede cambiar usando el atributo border en la etiqueta <table>, el valor de este atributo es un número que nos indicará el grosor del borde medido en pixels.

Importante!!! → El atributo border debe utilizarse únicamente con la etiqueta table ya que con las otras etiquetas de la tabla no tiene ningún efecto.

*Ejemplo:* `<table border="2"> ... </table>`

# 1. Creación de tablas

## Tamaño de las celdas

Podemos modificar el tamaño de las celdas de la tabla, para ello utilizaremos los atributos height (alto) y width (ancho). Los atributos pueden colocarse en cualquiera de las celdas, pero hay que tener en cuenta que si se modifica el ancho de una celda, se modifica también el tamaño de la columna, y al modificar el alto de una celda, modificamos el alto de la fila.

Los valores de los atributos height (alto) y width (ancho) pueden ser números o porcentajes. Si escribimos un número, indicará píxeles, y si escribimos un porcentaje, éste está referido al total del ancho o alto de la tabla.



# 1. Creación de tablas

## Tamaño de las celdas

Si usamos los atributos height (alto) y width (ancho) con la etiqueta table definiremos la anchura o la altura de toda la tabla, y las celdas se adaptarán proporcionalmente al tamaño de la tabla.

Si modificamos el tamaño de dos celdas de la misma fila o columna, la tabla tomará siempre el valor más alto de los dos y lo aplicará a esa fila o columna.

```
<table border="1">
<caption>Tabla de ejemplo</caption>
<tr>
  <td height="50" width="150">Mi tabla</td>
  <th width="200">Primera Columna</th>
  <th width="200">Segunda Columna</th>
</tr>
<tr>
  <th height="75">Primera fila</th>
  <td>A-1</td>
  <td>A-2</td>
</tr>
<tr>
  <th height="75">Segunda fila</th>
  <td>B-1</td>
  <td>B-2</td>
</tr>
</table>
```



Tabla de ejemplo

Mi tabla	Primera Columna	Segunda Columna
Primera fila	A-1	A-2
Segunda fila	B-1	B-2

# 1. Creación de tablas

## Alineación

- Podemos alinear el texto de las celdas mediante el atributo **align**. Los valores que puede tomar el atributo son left (izquierda) center (centrado) o right (derecha) aplicado a las etiquetas td o th .
- También podemos alinear el texto de la celda en vertical, mediante el atributo **valign**. Los valores que puede tomar el atributo son top (posición superior), middle (en el centro) o bottom (posición inferior). lo aplicaremos igualmente a las etiquetas td o th.
- Si queremos alinear de igual modo todos los elementos de una fila podemos usar los atributos align y valign en la etiqueta tr correspondiente a la fila. Todas las celdas de esa fila se alinearán de la manera indicada.
- Si utilizamos el atributo align en la etiqueta table la tabla entera quedará alineada respecto a su elemento contenedor (en su defecto respecto al fondo de la página), pero no se alinean los elementos del interior de las celdas.

# 1. Creación de tablas

## Color de fondo y bordes

- **bordercolor:** modifica el color de los bordes de la tabla mediante el atributo. Este atributo debemos insertarlo en la etiqueta table y afecta a toda la tabla.
- **bgcolor:** cambia el color de fondo de las celdas de la tabla. Puede insertarse en las etiquetas de celda (td o th) o en las etiquetas de fila (tr), en este último caso el cambio de color afectará a toda la fila.
- background: Para poner una imagen de fondo a la tabla, fila o celda.

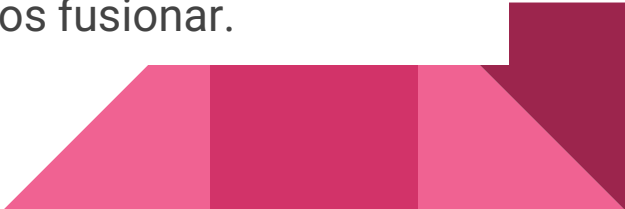
Tabla de ejemplo

Mi tabla	Primera Columna	Segunda Columna
Primera fila	A-1	A-2
Segunda fila	B-1	B-2

# 1. Creación de tablas

## Unificación de celdas

La unificación de celdas es un mecanismo mediante el cual un autor puede combinar o fusionar dos o más celdas adyacentes (td y th) en una tabla.

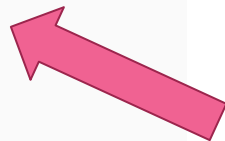
- **colspan:** Para que una celda ocupe dos o más columnas dentro de la misma fila. Su valor es el número de celdas que queremos fusionar.
  - **rowspan:** Para que una celda ocupe dos o más filas dentro de la misma columnas. Su valor es el número de celdas que queremos fusionar.
- 

# 1. Creación de tablas

## Unificación de celdas

```
<table width="80%" border="1" cellspacing="0" cellpadding="5">
  <tr>
    <th>Columna 1</th>
    <th>Columna 2</th>
    <th>Columna 3</th>
    <th>Columna 4</th>
  </tr>
  <tr>
    <td> F1-C1</td>
    <td colspan="2"> Fusión celdas F1-C2 y F1-C3</td>
    <td> F1-C4</td>
  </tr>
  <tr>
    <td colspan="2">Fusión celdas F2-C1 y F2-C2</td>
    <td colspan="2">Fusión celdas F2-C3 y F2-C4</td>
  </tr>
  <tr>
    <td colspan="4">Fusión celdas F3-C1, F3-C2, F3-C3 y F3-C4</td>
  </tr>
</table>
```

Columna 1	Columna 2	Columna 3	Columna 4
F1-C1	Fusión celdas F1-C2 y F1-C3		F1-C4
Fusión celdas F2-C1 y F2-C2		Fusión celdas F2-C3 y F2-C4	
Fusión celdas F3-C1, F3-C2, F3-C3 y F3-C4			



¡Si en una celda (etiqueta td o th) hay un atributo colspan="2" la celda está ocupando dos columnas, por lo que deberemos poner una celda menos para que el número de columnas sea el mismo que en las demás filas!

# 1. Creación de tablas

## Unificación de celdas

```
<table width="80%" border="1" cellspacing="0" cellpadding="5">
  <tr>
    <th>Fila 1</th>
    <td>F1-C1</td>
    <td rowspan="2">Fusión F1-C2 y F2-C2</td>
    <td rowspan="3">Fusión F1-C3, F2-C3 y F3-C3</td>
    <td rowspan="4">Fusión F1-C4, F2-C4, F3-C4 y F4-C4</td>
  </tr>
  <tr>
    <th>Fila 2</th>
    <td rowspan="2">Fusion F2-C1 y F3-C1</td>
  </tr>
  <tr>
    <th>Fila 3</th>
    <td rowspan="2">Fusión F3-C2 y F4-C2</td>
  </tr>
  <tr>
    <th>Fila 4</th>
    <td>F4-C1</td>
    <td>F4-C3</td>
  </tr>
</table>
```

¡Si una fila está ya ocupada por una celda fusionada con otra de una fila superior, no debemos marcarla con ninguna etiqueta, ya que está indicada mediante el atributo **rowspan** en la fila superior!

Fila 1	F1-C1	Fusión F1-C2 y F2-C2	Fusión F1-C3, F2-C3 y F3-C3	Fusión F1-C4, F2-C4, F3-C4 y F4-C4
Fila 2	Fusion F2-C1 y F3-C1			
Fila 3		Fusión F3-C2 y F4-C2		
Fila 4	F4-C1	F4-C3		

# Tema 02 - HTML (Formularios)

Parte 3

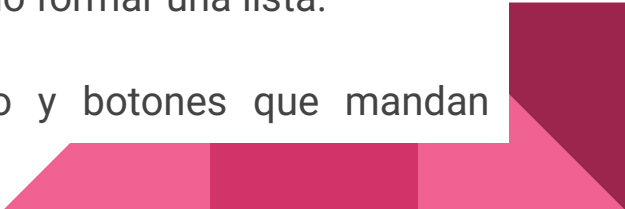
# 1. Creación de un formulario

El formulario es la forma en que el visitante de la página Web manda información a su administrador.

Mediante los formularios se pueden mandar datos, rellenar una encuesta, enviar comentarios, etc. Se usan cuando se quiere que el visitante de la página nos envíe informaciones concretas.

El formulario consta de unos mecanismos en donde el usuario responde a una serie de preguntas. Para ello puede escribir en una serie de cajas la información solicitada, y/o también elegir entre una o varias opciones disponibles, las cuales pueden o no formar una lista.

El lenguaje HTML puede crear formularios con cajas de texto y botones que mandan información






# 1. Creación de un formulario

**Un formulario debe ir siempre dentro de la etiqueta `<form> .... </form>`.** Tanto los elementos para mandar información como los adicionales (textos, imágenes, elementos explicativos sobre la información que se debe mandar, etc.) deben ir encerrados dentro de estas etiquetas.

Los elementos de formulario que recogen información, como botones y cuadros de texto, se denominan "campos de formulario" o "controles de formulario".



## 2. Campos de texto

### Texto corto

Nombre de la etiqueta	Atributos obligatorios	Atributos opcionales
input	<ul style="list-style-type: none"><li>- type="text"</li><li>- name=" "</li></ul>	<ul style="list-style-type: none"><li>- maxlength: Define el máximo de caracteres que puedes introducir en el formulario.</li><li>- size: define el tamaño de la caja a lo ancho.</li><li>- placeholder: Este atributo indica el texto que aparecerá escrito en la caja por defecto cuando se cargue la página.</li><li>- required: Este atributo indica que es obligatorio rellenar este campo.</li></ul>

```
<p>nombre: <input type="text" name="nombre"/></p>
```

nombre:

## 2. Campos de texto

### Texto oculto

Nombre de la etiqueta	Atributos obligatorios	Atributos opcionales
input	<ul style="list-style-type: none"><li>- type="password"</li><li>- name=" "</li></ul>	<ul style="list-style-type: none"><li>- maxlength: Define el máximo de caracteres que puedes introducir en el formulario.</li><li>- size: define el tamaño de la caja a lo ancho.</li><li>- required: Este atributo indica que es obligatorio rellenar este campo.</li></ul>

## 2. Campos de texto

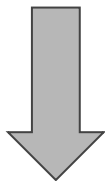
### Texto largo

Nombre de la etiqueta	Atributos obligatorios	Atributos opcionales
<code>textarea</code>	<ul style="list-style-type: none"><li>- <code>name=" "</code></li></ul>	<ul style="list-style-type: none"><li>- <code>rows</code>: define el número de líneas de texto que caben.</li><li>- <code>cols</code>: define el número de columnas que se pueden escribir por línea.</li><li>- <code>placeholder</code>: Este atributo indica el texto que aparecerá escrito en la caja por defecto cuando se cargue la página.</li><li>- <code>required</code>: Este atributo indica que es obligatorio rellenar este campo.</li></ul>

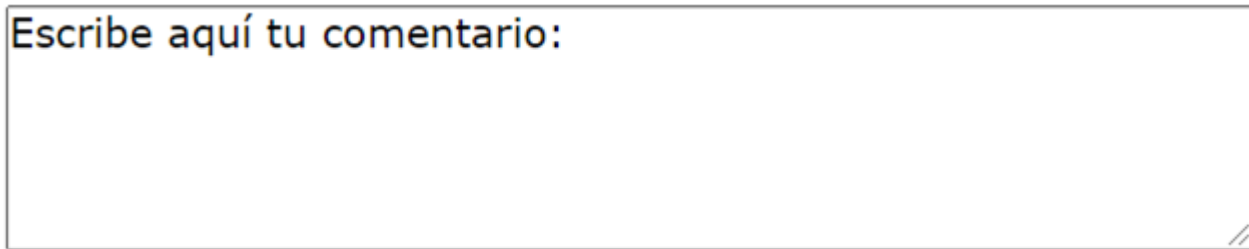
## 2. Campos de texto

### Texto largo

```
<p><textarea name="comentario" rows="5" cols="50">Escribe aquí tu comentario: </textarea></p>
```



Escribe aquí tu comentario:

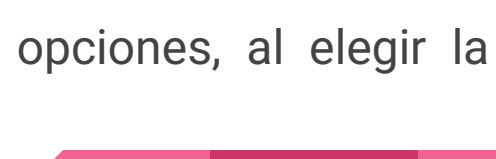


## 2. Campos de texto

### Botones radio

Además de poder enviar texto mediante un formulario, podemos hacer que el usuario elija entre varias opciones que le presentemos. Para ello se le presentan una serie de opciones o casilleros entre los cuales el usuario puede marcar las que le interesen.

**En el caso de los botones radio se le presentan varias opciones de la que deberá marcarse sólo una.** Si se intentan elegir dos opciones, al elegir la segunda se desactiva la primera.



### 3. Botones

#### Botones radio

Nombre de la etiqueta	Atributos obligatorios	Atributos opcionales
input	<ul style="list-style-type: none"><li>- type="radio"</li><li>- name=" " → <b>su valor debe ser el mismo en todas las etiquetas relacionadas en las que queremos que sólo se pueda marcar una opción.</b></li><li>- value: valor de la etiqueta, ya que mediante este atributo sabremos cual es el valor que el usuario ha marcado (en javascript)</li></ul>	<ul style="list-style-type: none"><li>- checked="checked": la opción aparece ya marcada al cargarse la página.</li><li>- required: Este atributo indica que es obligatorio rellenar este campo.</li></ul>

### 3. Botones

#### Botones radio

```
<form ...>

  <p>Elige una estación del año</p>
  <p><input type="radio" name="estacion" value="primavera"/> Primavera </p>
  <p><input type="radio" name="estacion" value="verano"/> Verano </p>
  <p><input type="radio" name="estacion" value="otoño"/> Otoño </p>
  <p><input type="radio" name="estacion" value="invierno"/> Invierno </p>

</form>
```



Elige una estación del año

- ☐ Primavera
- ☐ Verano
- ☐ Otoño
- ☐ Invierno



### 3. Botones

#### Cajas de validación (checkbox)

Las cajas de validación son también botones de opciones, pero en este caso podemos marcar uno o varios a la vez, ya que al marcar uno no se desactivan los otros, pues cada botón es independiente del resto.

Nombre de la etiqueta	Atributos obligatorios	Atributos opcionales
input	<ul style="list-style-type: none"><li>- type="checkbox"</li><li>- name=" " → <b>su valor debe ser distinto en cada uno de los botones</b></li></ul>	<ul style="list-style-type: none"><li>- checked="checked": la opción aparece ya marcada al cargarse la página.</li><li>- required: Este atributo indica que es obligatorio rellenar este campo.</li></ul>

### 3. Botones

#### Cajas de validación (checkbox)

```
<form ...>  
  
  <p>Suscribirse a las siguientes secciones:</p>  
  <p><input type="checkbox" name="deportes"> Deportes</p>  
  <p><input type="checkbox" name="noticias"> Noticias</p>  
  <p><input type="checkbox" name="cultura"> Cultura</p>  
  <p><input type="checkbox" name="economia"> Economía</p>  
  
</form>
```



Suscribirse a las siguientes secciones:

- ☐ Deportes
- ☐ Noticias
- ☐ Cultura
- ☐ Economía

### 3. Botones

#### Botón de envío

Es el único que es obligatorio, ya que todo formulario debe ser enviado a algun correo o página de procesamiento. su código HTML es el siguiente:

**`<input type="submit" value="texto">`**

- La etiqueta y atributo `input type="submit"` es la que indica que éste es el botón de envío.
- El atributo `value` indica cual es el texto que incluiremos dentro del botón

```
<input type="submit" value="Enviar el formulario">
```



Enviar el formulario

### 3. Botones

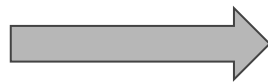
#### Botón de borrado

Este botón sirve para borrar todos los campos del formulario y dejarlos en sus valores predeterminados. Caso de haberse equivocado, o querer cambiar algunos datos una vez relleno, el usuario puede pulsar este botón y volver a empezar a rellenarlo.

El código HTML de este botón es parecido al del anterior, lo único que cambia es el valor del **atributo type el cual es reset**. por tanto su código HTML será el siguiente:

**`<input type="reset" value="texto">`**

`<input type="reset" value="borrar todo">`



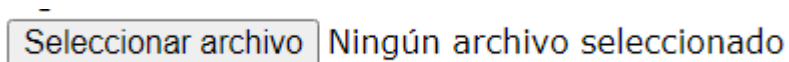
Borrar todo

### 3. Botones

#### Envío de archivos adjuntos

Podemos pedir al usuario que nos envíe junto con el formulario un archivo adjunto, para ello usaremos el código `input type="file"` el atributo `name` también es imprescindible para poder recibir correctamente los datos. El código HTML de este tipo de campo será el siguiente:

**`<input type="file" name="archivo" value="texto">`**



**Importante** → Al incluir un control para adjuntar archivos, es obligatorio añadir el atributo `enctype` en la etiqueta `<form>` del formulario. El valor del atributo `enctype` en este caso debe ser `multipart/form-data`

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

## 3. Botones

### Botones normales

Su aspecto es similar al de los botones de envío y borrado. El código de el botón anterior es el siguiente:

```
<input type="button" value="botón normal" name="boton1">
```

El atributo value indica el texto que veremos en el botón. Por otra parte debemos insertar el atributo name para enlazar con el código javascript u otro tipo de programas que den utilidad a este botón.



botón normal

## 4. Listas de opciones

### Listas desplegables

Dentro de un formulario podemos poner también una lista desplegable en la que el usuario puede elegir una o varias opciones.

- La lista entera debe estar encerrada dentro de la etiqueta **<select> ... </select>**. Esta etiqueta puede llevar el atributo `multiple = "multiple"` para indicar que se pueden elegir varias opciones.
- Dentro de la etiqueta **<select> ... </select>** escribiremos las opciones de la lista mediante las etiquetas **<option> texto\_de\_la\_opción </option>**. Esta etiqueta puede llevar el atributo `selected="selected"` para indicar que es el valor que se ve por defecto.
- Al igual que en los campos anteriores debemos insertar el **atributo name** en la etiqueta `select` para tener una referencia en los datos recibidos del campo al que se refieren.

## 4. Listas de opciones

### Listas desplegables

```
<p>Elige un color:  
  <select name="color">  
    <option>rojo</option>  
    <option>amarillo</option>  
    <option>verde</option>  
    <option>azul</option>  
  </select>  
</p>
```



Elige un color:





## 5. Agrupación de etiquetas

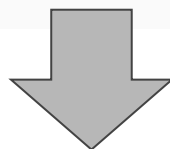
Es posible mejorar el aspecto de un formulario agrupando varias etiquetas en un mismo apartado.

- **Para ello utilizaremos la etiqueta `<fieldset> ... </fieldset>`.** Además también podemos poner una cabecera o título al cuadro que forma la etiqueta `fieldset` mediante la etiqueta: `<legend>texto</legend>`, la cual debe estar también incluida en la etiqueta `fieldset`
- Dentro de `<fieldset> ... </fieldset>` incluiremos los distintos campos del formulario.



## 5. Agrupación de etiquetas

```
<form ....>  
  
<fieldset>  
<legend>datos personales</legend>  
<p>nombre: <input type="text" name="nombre"/></p><br/>  
<p>Apellido: <input type="text" name="apellido"/></p><br/>  
<p>Correo electrónico: <input type="text" name="correo"/></p>  
</fieldset>  
...  
</form>
```



datos personales

nombre:

Apellido:

Correo electrónico:

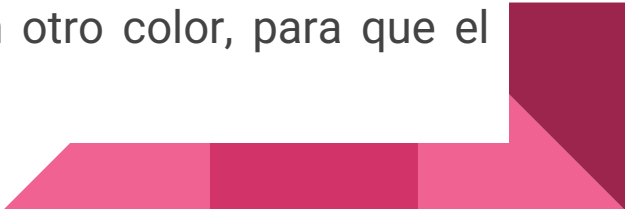
# Tema 02 - HTML (Enlaces)

Parte 4

# 1. Tipos de enlaces

Los enlaces permiten conectar entre sí las páginas de un sitio web, así como con otros sitios Web. Éstos permiten que el navegante pueda saltar de una página a otra y de un sitio a otro rápidamente, con sólo hacer un clic.

Los enlaces suelen detectarse fácilmente. Basta normalmente con deslizar el puntero del ratón sobre las imágenes o el texto que contiene un enlace y la forma del puntero cambia, transformándose por regla general en una mano con un dedo señalador. Además suelen ir subrayados o destacados en otro color, para que el usuario no tenga dificultad en reconocerlos.



# 1. Tipos de enlaces

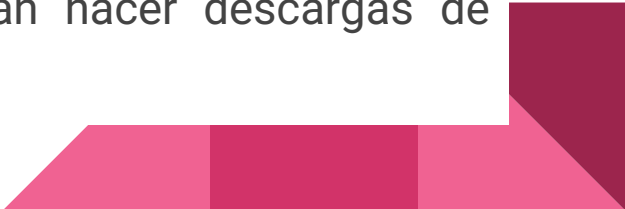
- Los enlaces se crean con la etiqueta `<a>...</a>`
- Esta etiqueta lleva de manera obligatoria el atributo `href="..."` en el cual introduciremos el destino del enlace.

**`<a href="destino">texto_del_enlace</a>`**

- El "destino" o "ruta" es el sitio a donde apunta el enlace y puede ser una página, un correo, o un archivo.
- El "texto\_del\_enlace" es el texto en el que debemos hacer clic para que el enlace funcione, aunque también puede ser una imagen u otro elemento.

# 1. Tipos de enlaces

En función de su destino o lugar al que nos dirigen, los enlaces se clasifican en:

- **Enlaces externos:** los dirigidos hacia páginas de otros sitios web.
  - **Enlaces locales:** los que se dirigen a otras páginas del mismo sitio web.
  - **Enlaces internos:** los que se dirigen a otras partes dentro de una misma página.
  - **Enlaces con direcciones de correo:** para crear un mensaje de correo dirigido a una dirección.
  - **Enlaces con archivos:** para que los usuarios puedan hacer descargas de ficheros.
- 

## 2. Enlaces externos

Son los enlaces que se dirigen hacia páginas que se encuentran fuera de nuestro sitio web, es decir, cualquier otra página que no forma parte de nuestro sitio.

En este caso el destino será la URL de la página (lo que escribimos en la ventana superior del navegador). Éste lo escribiremos como el valor del atributo href.

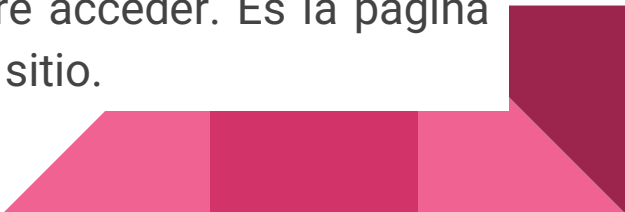
**<a href="http://google.com">Enlace a Google </a>**



[Enlace a google](http://google.com)

## 2. Enlaces externos

La URL es único para cada página, se debe escribir completo dentro del valor de la etiqueta href y consta de las siguientes partes:

- **Protocolo (http://):** Es el mecanismo que utiliza el navegador para acceder a ese recurso. Todas las páginas web utilizan el protocolo http://.
  - **Servidor (www.wikipedia.org):** se trata del ordenador en el que se encuentra guardada la página a la que se quiere acceder. Podríamos definirlo como el nombre del sitio.
  - **Ruta (/aaa/bbb.html):** camino que se debe seguir una vez que se ha llegado al servidor para localizar la página concreta que se quiere acceder. Es la página concreta que queremos ver una vez hemos accedido al sitio.
- 




### 3. Enlaces locales

Son aquellos que nos llevan a otra página del mismo sitio web que estamos visualizando.

Los enlaces locales tienen la misma sintaxis que los enlaces externos, lo único que cambia es el valor del atributo href que en lugar de indicarnos una URL o dirección web, debe indicarnos la ruta o camino que debemos seguir para encontrar ese archivo.

**`<a href="ruta">Texto_del_enlace</a>`**

Al estar el archivo de partida y el de destino en el mismo sitio Web, basta con escribir la ruta. Esta ruta la podemos escribir de forma absoluta o relativa.



### 3. Enlaces locales

#### URL relativas vs. URL absolutas

Las rutas y URLs en los enlaces externos son absolutas. Una URL absoluta indica todas las partes de la URL (protocolo, servidor y ruta completa). La URL relativa tiene en cuenta la ubicación de la página de origen, y prescinde de todas las partes de la URL absoluta que coinciden con la URL del archivo o página de origen.

Es por esto por lo que en los enlaces locales no necesitamos escribir ni el protocolo ni el servidor. Además sólo debemos indicar el camino (carpetas de distancia y nombre de ellas) que hay entre el origen y el destino. Esto es mucho más simple que escribir la URL completa para cada enlace interno (lo cual también funcionaría).

## 4. Enlaces internos

Se denominan enlaces internos aquellos que llevan a otra parte de la misma página. Para ello la página debe ser lo bastante larga para que no quepa entera en la pantalla, y haya que acceder a gran parte de ella utilizando la barra de desplazamiento, generalmente la vertical.

En este caso se pueden colocar unos enlaces, generalmente al principio de la página, que pulsándolos nos lleven a las diferentes partes de la página.

Aunque estos enlaces suelen colocarse al principio de la página, en realidad podemos colocarlos en cualquier sitio, y hacer que apunten a cualquier otro lugar de la página. el punto de la página al que apuntan aparecerá lo más cerca posible de la esquina superior izquierda de la pantalla

Los enlaces internos constan de dos partes: ya que al estar los dos en la misma página, hay que indicar no sólo el origen, sino también el destino.

## 4. Enlaces internos

### Origen y destino

1. El punto de origen es donde pulsaremos para activar el enlace. Ahí debemos colocar la etiqueta de origen que será la siguiente:

**`<a href="#destino">Texto_del_enlace</a>`**

La etiqueta que usamos es la misma que para un enlace normal, pero en el valor del atributo href escribiremos un nombre para el destino, siempre precedido del signo #.

1. El punto de destino es donde queremos que nos envíe el enlace: ahí debemos colocar la etiqueta de destino, que será la siguiente:

**`<a id="destino">Texto_del_destino</a>`**

Aquí la etiqueta `<a>` no lleva el atributo href ya que no es de origen, sin embargo lleva el atributo name que es el que indica dónde está el destino. El valor del atributo name debe ser el mismo que el valor del atributo href en la etiqueta de origen, pero sin #.

# Tema 02 - HTML (Multimedia)

Parte 5

# 1. Imágenes

Para incluir una imagen en una página HTML debemos guardarla previamente, o bien tener localizada la URL o dirección de la imagen en otra página Web.

Para incluir imágenes usaremos la etiqueta `<img>` que llevará obligatoriamente el atributo `src="URL_de_la_imagen"`.

Otro atributo que, aunque no es obligatorio, es muy recomendable es el atributo `alt="texto"`. Como valor se suele escribir un texto corto que es el que aparecerá en el caso de que el navegador no pueda cargar la imagen. Este texto suelen mostrarlo algunos navegadores en un recuadro al pasar el ratón por encima de la imagen.

La sintaxis de una etiqueta de imagen en su forma básica será la siguiente:

```

```

# 1. Imágenes

Las etiquetas de imagen no tienen etiquetas de cierre, por lo que suelen escribirse con una barra inclinada al final. Esta barra inclinada no es obligatoria, sin embargo es recomendable ponerla.

Los tipos de imágenes más usados son:

- **PNG:** Soporta transparencia. Compresión sin pérdidas.
- **JPG:** Compresión con pérdidas. Ideal para imágenes con texturas.
- **SVG:** Formato vectorial. Ideal para imágenes escalables.
- **GIF:** Formato para imágenes pequeñas y animadas.
- **WEBP:** Alternativa libre de Google al JPEG. Soporta transparencias.

# 1. Imágenes

## Tamaño de la imagen

Como no siempre nos coincide el tamaño real de la imagen con el tamaño que queremos que tenga en nuestra página, muchas veces tenemos que modificar el tamaño de las imágenes para adaptarlas.

Lo más normal es usar los atributos **height=" "** (alto) y **width=" "** (ancho) en la etiqueta HTML, para definir el alto y el ancho de la imagen.

Los valores de los atributos height=" " y width=" " pueden ser:

- **números:** el navegador los interpreta como píxeles.
- **porcentajes:** indican el tanto por ciento de espacio que ocupa la imagen dentro de su elemento contenedor, caso de que no tuviera elemento contenedor, éste sería la página (elemento body).



# 1. Imágenes

## Tamaño de la imagen

Como no siempre nos coincide el tamaño real de la imagen con el tamaño que queremos que tenga en nuestra página, muchas veces tenemos que modificar el tamaño de las imágenes para adaptarlas.

Lo más normal es usar los atributos **height=" "** (alto) y **width=" "** (ancho) en la etiqueta HTML, para definir el alto y el ancho de la imagen.

Los valores de los atributos height=" " y width=" " pueden ser:

- **números:** el navegador los interpreta como píxeles.
- **porcentajes:** indican el tanto por ciento de espacio que ocupa la imagen dentro de su elemento contenedor, caso de que no tuviera elemento contenedor, éste sería la página (elemento body).

# 1. Imágenes

## Alineación de imágenes

Lo habitual es integrar la imagen en el texto de forma que la imagen quede a un lado de la página y el texto cubriendo todo el otro lado. Esto lo conseguimos con el atributo **align="left"** (imagen a la izquierda), o **align="right"** (imagen a la derecha).

****

La opción de colocar la imagen centrada con el texto a ambos lados no se obtiene con el atributo align="center", ya que éste muestra la imagen a la izquierda y a su derecha una sólo línea de texto centrada en vertical respecto de la imagen. **Para esta opción debemos utilizar un método más complejo mediante código CSS.**

# 1. Imágenes

## Bordes y márgenes

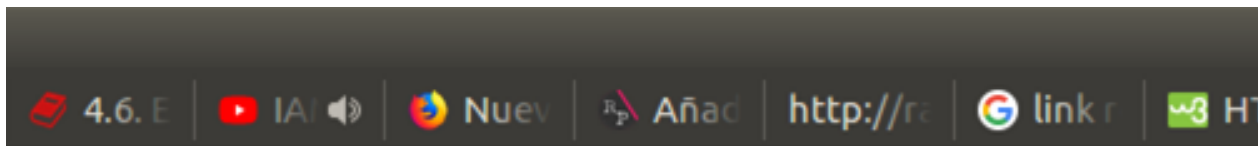
- El atributo **border**="..." define un borde del tamaño indicado alrededor de la imagen. Su valor es un número que expresa los píxeles de grosor que tendrá el borde.
- Los atributos **vspace**="..." y **hspace**="..." indican el espacio libre, en píxeles, que tiene que colocarse entre la imagen y los otros elementos que la rodean. vspace indica los márgenes laterales, mientras que hspace indica el margen superior e inferior.

La mayoría de los navegadores modernos ignoran los atributos `vspace="..."` y `hspace="..."` por lo que para definir los márgenes debemos emplear código CSS.

## 2. Favicon

Un favicon es la pequeña imagen que se muestra en la pestaña del navegador o en la lista de marcadores (favoritos). En la barra de dirección, el tamaño del favicon es bastante reducido, 16x16 píxeles, pero en otros lugares como los marcadores puede tener un tamaño mayor (24x24, 32x32, 48x48 o 64x64).

**<link rel="icon" type="image/png" href="URL\_DE\_LA\_IMAGEN"  
sizes="TAMAÑO">**



## 2. Favicon

- **<link/>**: es una etiqueta vacía y tiene que ir en el <head> de nuestro documento HTML.
- **El atributo rel**: siempre es "icon".
- **El atributo type**: depende de la extensión de la imagen que queramos poner: png, gif, jpg ...
- **El atributo href**: es la ruta de donde se encuentra la imagen.
- **El atributo sizes**: es opcional. Sirve para poner el tamaño sizes="16x16 24x24 36x36 48x48". Para imágenes escalables, como SVG, se puede utilizar el valor any.

Puedes descargar algunos desde aquí → <https://icon-icons.com/>

### 3. Vídeo y audio

Para incluir un vídeo en una página HTML debemos tener descargado el archivo o tener localizada la URL o dirección del archivo en otra página Web.

Para incluir vídeos usaremos la etiqueta `<video></video>` que llevará obligatoriamente el atributo `src="URL_del_video"`.

**`<video src = "URL_del_video"></video>`**

Para incluir audios usaremos la etiqueta `<audio></audio>` que llevará obligatoriamente el atributo `src="URL_del_audio"`.

**`<audio src = "URL_del_audio"></audio>`**

```
<audio src="sintonia.mp3" autoplay loop ></audio>
```

```
<video src="mivideo.mp4" controls autobuffer widht="300"></video>
```

### 3. Vídeo y audio

#### Atributos

- **autoplay:** Indica que el archivo debe reproducirse automáticamente al cargarse la página. Este atributo no tiene valores.
- **controls:** Indica que deben mostrarse los controles para que el usuario pueda iniciar, detener , pausar el archivo. Este atributo no tiene valores.
- **loop:** El archivo vuelve a reproducirse cuando acaba, una y otra vez. Este atributo tampoco tiene valores.
- **autobuffer:** El archivo tiene que descargarse completamente antes de empezar a reproducirse. Este atributo tampoco tiene valores.
- **width="num":** Sólo para video; su valor indica en píxeles el ancho de la pantalla en que se reproduce el video.
- **height="num":** Sólo para video; su valor indica en pixels el alto de la pantalla en que se reproduce el video.

### 3. Vídeo y audio

Los formatos que soportan los principales navegadores para la etiqueta video de HTML5:

Formatos	Firefox	Chrome	I.E. 10	Safari	Opera
.ogv	si	si	no	no	si
.mp4	si	si	si	si	no
.webm	si	si	no	no	si

Los formatos que soportan los principales navegadores para la etiqueta audio de HTML5:

Formatos	Firefox	Chrome	I.E. 10	Safari	Opera
.mp3	si	si	si	si	no
.wav	si	si	no	si	si
.ogg	si	si	no	no	si



## 4. Etiqueta source

No todos los navegadores soportan todos los formatos de audio o video, por lo que podemos poner en la página enlaces al mismo archivo en varios formatos, para que el navegador detecte el formato que soporta. Para ello utilizamos la etiqueta source:

**<audio autoplay loop>**

**<source src="sintonia.mp3"/>**

**<source src="sintonia.wav"/>**

**<source src="sintonia.au"/>**

**</audio>**

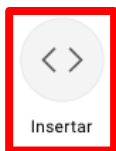
La etiqueta es válida tanto para audio como para video. Dentro de la etiqueta source indicamos la ruta a diferentes archivos. El navegador intentará reproducir el archivo indicado en la primera etiqueta, si no lo consigue pasará a la segunda, y así sucesivamente hasta que consiga reproducir alguno de ellos.

## 5. Introducir vídeos y listas de reproducción desde

1. En un ordenador, ve al vídeo o a la lista de reproducción de YouTube que quieras insertar.
2. Haz clic en COMPARTIR Compartir.
3. En la lista de opciones que se muestra, haz clic en Insertar.
4. Copia el código HTML del cuadro que aparece.
5. Pégallo en el código HTML de tu sitio web.



Compartir



WhatsApp



Pinterest



Facebook



Twitter

[https://youtu.be/WSFT4OQF\\_mA](https://youtu.be/WSFT4OQF_mA)

☐ Empezar en 0:24

Insertar vídeo

```
<iframe width="560" height="315"
src="https://www.youtube.com/embed/WS
FT4OQF_mA" title="YouTube video
player" frameborder="0"
allow="accelerometer; autoplay;
clipboard-write; encrypted-media;
gyroscope; picture-in-picture"
allowfullscreen></iframe>
```

☐ Empezar en 0:24

COPIAR

## Ejercicio

Pon en práctica qué has aprendido:

- Crea un archivo HTML sobre la arquitectura de Von Neumann en la que introduzcas un favicon.
- Introduce un pequeño párrafo en la que describas brevemente qué es la arquitectura Von Neumann.
- Añade una tabla similar a la siguiente en la que introduzcas imágenes, audio y vídeos.

Imagen	Imagen	
	Vídeo con source	Vídeo Youtube
	Audio	

- Un vídeo tiene que ser introducido con la etiqueta <video> y <source> y otro desde Youtube
- Utiliza los atributos vistos tanto en audio como en vídeo.

# Tema 02 - HTML (Div y Span)

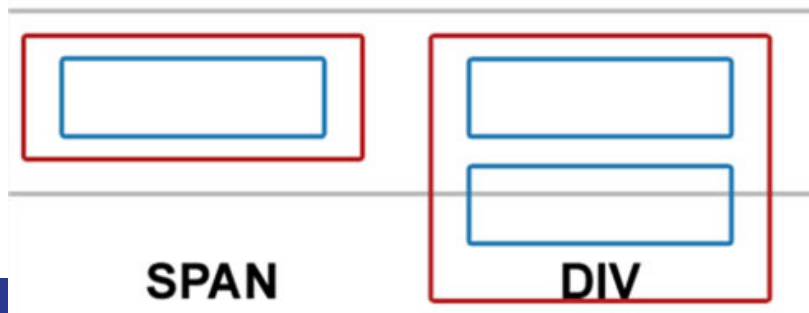
Parte 6

# 1. Elementos de bloque vs. Elementos en línea

El lenguaje HTML5, define mediante etiquetas, elementos dentro de un documento web. Estos elementos pueden ser de dos tipos denominados, **elementos de bloque** y **elementos en línea**.

Su nombre es debido a la forma en que ocupan espacio dentro del documento que se está creando.

Un elemento en bloque significa que el elemento ocupa todo el ancho del documento disponible produciendo un retorno de línea al final del mismo.



# 1. Elementos de bloque vs. Elementos en línea

- El elemento en bloque ocuparía todo el ancho de página del documento.
- Los elementos en línea solo utilizan el espacio que ocupan y se colocan en línea.

El párrafo es un elemento en bloque.

Por el contrario un [enlace](#) es un elemento en línea, como puede observar en este párrafo.

Las etiquetas <div> y <span> no tienen ningún tipo de significado especial, solo que la primera se define como un elemento de bloque y por tanto el navegador mostrará un salto de línea antes y después de la misma. Mientras que la segunda es un elemento de línea.

**Un uso muy común de las mismas era también para establecer el diseño del documento mediante hojas de estilo.**

## 2. Etiqueta DIV

La etiqueta <div> viene de división, prácticamente funciona y sirve de contenedor de bloque, es decir puede contener uno o varios elementos HTML dentro de ella (incluso otros div), es bastante utilizada para agrupar varios elementos que luego serán apuntados con CSS con la finalidad de dar un estilo adecuado a dicho grupo, o también puede servir para organizar mejor nuestra página web.

```
1 <div id="rojo" style="background-color: red;">
2
3 <h1> Titulo en contenedor rojo</h1><br>
4
5 <div id="azul" style="background-color: blue;">
6 <p> Este es un párrafo dentro de un contenedor div con fondo rojo y que a su
  vez esta dentro de otro contenedor div con fondo azul</p>
7 </div>
8
9 <br><br>
10
11 </div><br>
12
13 <div id="verde" style="background-color: green;">
14 <h2> Subtitulo en contenedor verde</h2>
15 
16 <p> Este es otro parrado que esta dentro de un contenedor div con fondo
  verde</p></div>
```



## 2. Etiqueta SPAN

La etiqueta `<span>` también funciona como un contenedor pero en este caso será un contenedor en línea. Por ejemplo para destacar una palabra o expresión dentro de un párrafo utilizaremos la etiqueta `<span>` para encerrar esa palabra o expresión y luego ponerle un estilo mediante un atributo.

```
1 <p> Este es un párrafo en el que insertaremos un elemento span a continuación  
para diferenciarlo <span style="background: yellow; color: red;">Texto  
especial dentro de etiqueta span con fondo amarillo y color rojo</span> ese  
texto especial sigue dentro del párrafo y en línea, forma parte del párrafo,  
pero podemos especificar alguna característica </p>
```



Este es un párrafo en el que insertaremos un elemento span a continuación para diferenciarlo **Texto especial dentro de etiqueta span con fondo amarillo y color rojo** ese texto especial sigue dentro del párrafo y en línea, forma parte del párrafo, pero podemos especificar alguna característica



## Ejercicio

- Crea la página principal de tu revista insertando divs y span para indicar las partes principales (cabecera, cuerpo, pie...)
- Crea una segunda página que sea la página de un artículo. Introduce fotos, texto, vídeo...

**Importante** → *Ten en cuenta que una revista online (como cualquier otra página web) mantiene una apariencia similar entre todas sus páginas, es decir, son lo más similares posibles, modificando principalmente la parte del contenido.*