



# ENTORNOS GRÁFICOS

## HTML

Recopilación realizada por :  
Prof. Ing. Daniela Díaz

Año  
2019

<html>

## HTML

Sigla en inglés de *HyperText Markup Language*

Lenguaje de marcas de hipertexto

Es un lenguaje de marcado para la elaboración de páginas web.

Es un estándar que sirve de referencia del software que conecta con la elaboración de páginas web en sus diferentes versiones.

Define una estructura básica y un código (denominado código HTML) para la definición de contenido de una página web, como texto, imágenes, videos, juegos, entre otros

# HTML

Es un estándar a cargo del *World Wide Web Consortium* (W3C) o Consorcio WWW, organización dedicada a la estandarización de casi todas las tecnologías ligadas a la web, sobre todo en lo referente a su escritura e interpretación.

Se considera el lenguaje web más importante siendo su invención crucial en la aparición, desarrollo y expansión de la World Wide Web (WWW).

Es el estándar que se ha impuesto en la visualización de páginas web y es el que todos los navegadores actuales han adoptado.

El Consorcio WWW, en inglés: World Wide Web Consortium (W3C), es un consorcio internacional que genera recomendaciones y estándares que aseguran el crecimiento de la World Wide Web a largo plazo.

La **World Wide Web (WWW)** o red informática mundial es un sistema de distribución de documentos de hipertexto o hipermedia interconectados y accesibles vía Internet.

Con un navegador web, un usuario visualiza sitios web compuestos de páginas web que pueden contener textos, imágenes, vídeos u otros contenidos multimedia, y navega a través de esas páginas usando hiperenlaces.



La World Wide Web (Web), en castellano "Tejido Mundial", es una red de recursos de información.

La Web se basa en tres mecanismos para hacer que estos recursos estén listos y a disposición de la mayor audiencia posible:

1. Un esquema uniforme de nombres para localizar recursos en la Web (p.ej., URIs).
2. Protocolos, para acceder a recursos con nombre en la Web (p.ej., HTTP).
3. Hipertexto, para navegar fácilmente entre recursos (p.ej., HTML).

# Introducción a los URIs

Todos los recursos disponibles en la Web (documentos HTML, imágenes, videoclips, programas, etc.) tienen una dirección que puede ser codificada mediante un *Universal Resource Identifier*, o "URI", es decir, un Identificador Universal de Recursos.

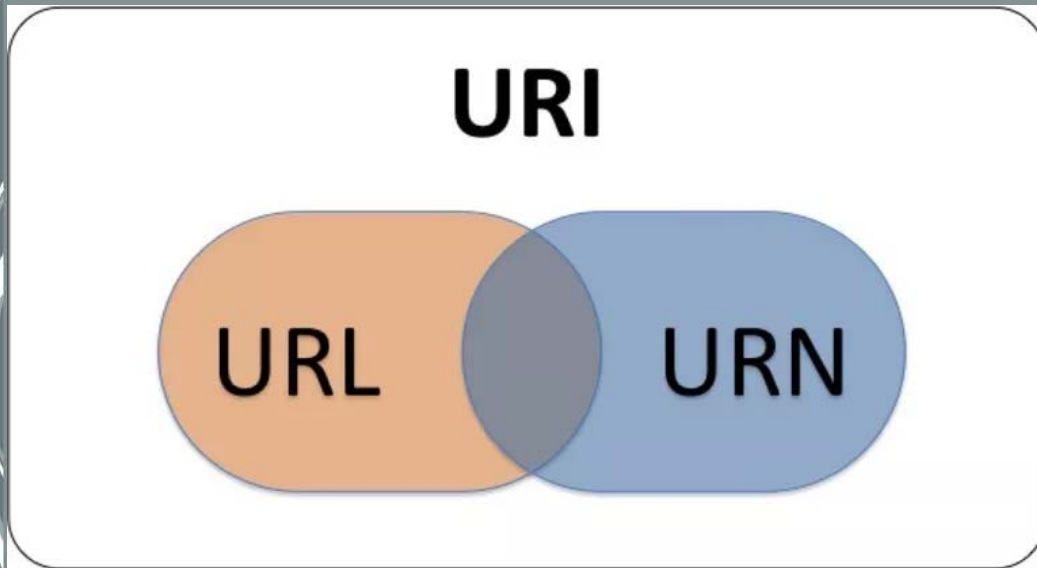
Los URIs se componen normalmente de tres partes:

1. El esquema de nombres del mecanismo usado para acceder al recurso.
2. El nombre de la máquina que aloja el recurso.
3. El nombre en sí del recurso, dado en forma de "path" o "ruta de acceso".

*Observación:* La mayoría estará familiarizado con el término "URL" y no con el término "URI".

*Los URLs forman un subconjunto del esquema de nombres URI, más gral*

# Introducción a los URIs



URI: siglas en inglés de Uniform Resource Identifier (identificador uniforme de recursos), que sirve para identificar recursos en Internet.

Dicho identificador de recursos tiene un formato estándar definido y su propósito es permitir interacción entre recursos disponibles en Internet -o en alguna red de cómputo-, recursos como lo son páginas, servicios, imágenes, vídeos, etc.

# Introducción a los URIs

El **URI** no se debe confundir con el **URL**, ya que el **URI**, es un concepto que incluye al **URL**.

*La diferencia fundamental es que los **URI** identifican y los **URL** localizan* y, como las localizaciones también sirven para identificar, se puede decir que todos los ***URL son también URI*** y que, por otro lado, hay URI que no son URL.

Por ejemplo...

Mi nombre, Luis Castro, es una identificación (sería como un URI) pero no dice nada de cómo localizarme, es más, sirve para identificar a mucha más gente con el mismo nombre.

En cambio mi dirección dice exactamente cómo localizarme (sería como un URL) y así es posible diferenciarme de los otros Luis Castro en el mundo y establecer contacto conmigo.





# Introducción a los URIs

Según el gráfico anterior:

*“un URI puede se un URL, puede ser un URN o puede ser ambos”.*

## URN

URN son las siglas de Uniform Resource Name, que sirve también para identificar a un recurso en una red pero usando un nombre.

Los nombres usados son menos amigables que un nombre propio, pero sí deben identificar de forma única a un recurso.

Los URN son poco usados por los cibernautas y los navegadores que empleamos.

# URIs, URL, URN

Formato de URI

Un URI tiene un formato estándar, que es:

*esquema://máquina/directorio/archivo#fragmento*

Consideremos el URI que designa la página de Informes Técnicos del W3C:

**http://www.w3.org/TR**

Este URI puede leerse de la siguiente manera:

"Hay un documento disponible a través del protocolo HTTP que se encuentra en la máquina `www.w3.org`, accesible a través de la ruta `/TR`".

# URIs, URL, URN

## Esquema de URI

El esquema en un URI sirve para fines de clasificación.

La especificación de los elementos que siguen después de los dos puntos varía en función del esquema.

Algunos ejemplos son comúnmente encontrados en el uso diario de Internet:

**http**, que es el esquema más frecuentemente encontrado al navegar en Internet.

**https**, que es el esquema usado para páginas seguras de Internet, conocido como SSL.

**mailto**, esquema usado para direcciones de correo electrónico

**ftp**, esquema usado para el protocolo de transferencia de archivos ftp.

# Volviendo al HTML...

Para publicar información y distribuirla globalmente, se necesita un lenguaje entendido universalmente, una especie de lengua franca de publicación que todas las computadoras puedan comprender potencialmente.

El lenguaje de publicación usado por la World Wide Web es el HTML (acrónimo de Hyper Text Markup Language, Lenguaje para el Formato de Documentos de Hipertexto).

El HTML da a los autores las herramientas para:

- ☐ Publicar documentos en línea con encabezados, textos, tablas, listas, fotos, etc.
- ☐ Obtener información en línea a través de vínculos de hipertexto, haciendo clic con el botón de un ratón.
- ☐ Diseñar formularios para realizar transacciones con servicios remotos, para buscar información, hacer reservas, pedir productos, etc.
- ☐ Incluir hojas de cálculo, videoclips, sonidos, y otras aplicaciones directamente en sus documentos.

# Breve historia del HTML

El HTML fue desarrollado originalmente por Tim Berners-Lee y fue popularizado por el navegador Mosaic.

Durante los años 90 ha proliferado con el crecimiento explosivo de la Web.

Durante este tiempo, el HTML se ha desarrollado de diferentes maneras.

La Web depende de que los autores de páginas Web y las compañías compartan las mismas convenciones de HTML. Esto ha motivado el trabajo colectivo en las especificaciones del HTML.

Los documentos HTML deberían funcionar bien en diferentes navegadores y plataformas.

Gracias a la interoperabilidad se reducen gastos, ya que sólo se debe desarrollar una versión de cada documento.

# Breve historia del HTML

El HTML ha sido desarrollado con la premisa de que cualquier tipo de dispositivo debería ser capaz de usar la información de la Web:

PCs con pantallas gráficas con distintas resoluciones y colores,  
teléfonos móviles,  
dispositivos de mano,  
dispositivos de salida y entrada por voz,  
computadoras con anchos de banda grandes o no, etc.

# HTML 5

**HTML 5** es la quinta revisión más importante que se hace al lenguaje HTML.

En este trabajo, se introducen nuevas características para ayudar a los autores de aplicaciones Web, y se ha prestado especial atención a la definición de claros criterios de conformidad para los agentes de usuario (navegadores) en un esfuerzo por mejorar la interoperabilidad *(la habilidad de dos o más sistemas o componentes para intercambiar información y utilizar la información intercambiada)*.

HTML es el lenguaje de marcado principal de la World Wide Web. Originalmente, HTML fue diseñado principalmente como un lenguaje para describir semánticamente documentos científicos. Su diseño general, sin embargo, ha permitido que se adapte, en los años posteriores, para describir varios otros tipos de documentos e incluso aplicaciones.



## ¿Qué es un lenguaje de marcado?

Un **lenguaje de marcado** hace referencia a aquellos lenguajes que emplean etiquetas.

Estas etiquetas ya están predefinidas dentro del lenguaje respectivo y contienen la información que “ayudan” a leer el texto. Es decir, tanto para los desarrolladores como para las plataformas que pueden leer este lenguaje, las etiquetas contienen información adicional de la estructura del texto.

*Su principal diferencia con los lenguajes de programación es que éstos últimos poseen funciones aritméticas o variables, mientras que los lenguajes de marcado no.*

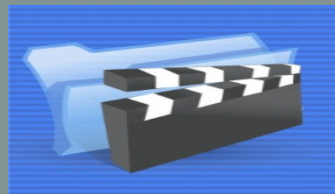


# ¿Para que sirve HTML5?

## ¿Cuáles son los principales usos?

El **lenguaje HTML5** se usa para definir la estructura básica de una **página web**. Sin embargo, una de sus más grandes adiciones en HTML 5 es poder añadir audio y video sin necesidad de usar Flash u otro reproductor multimedia.

Por medio de las etiquetas `<video>` y `<audio>` de HTML5, permite añadir videos o audio sin necesidad de usar Adobe Flash o cualquier otro plugin de tercero.



## ¿Para que sirve HTML5?

## ¿Cuáles son los principales usos?

También, pueden subirse videos a páginas de terceros como Youtube e incrustarlos en el nuevo sitio web, esta es una de las opciones más viables pues a pesar de colocar elementos multimedia, el peso final del archivo no se ve afectado.



La **geolocalización** permite al sitio detectar la ubicación de cada usuario que ingresa al sitio web. Por ejemplo, permite ofrecer opción de idiomas según el lugar de ubicación del usuario. Esta opción no se puede activar si el usuario no lo aprueba.

## ¿Para que sirve HTML5?

## ¿Cuáles son los principales usos?

Con HTML5 se pueden crear animaciones en 2D gracias a la etiqueta `<canvas>`. La API para esta etiqueta permite dibujar elementos en 2D y animarlos.



También la API permite añadir eventos de teclado, ratón y cualquier otro mando que desee incluir.

La gran ventaja de desarrollar aplicaciones HTML5 es que el resultado final es completamente accesible, es decir, se puede acceder a esta aplicación desde un ordenador, tablet o móvil.

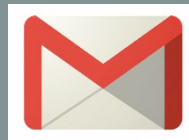
Incluso al cambiar de dispositivo, se puede acceder a la aplicación web mediante la URL respectiva, cosa que no sucede con una aplicación móvil.

# ¿Para que sirve HTML5?

## ¿Cuáles son los principales usos?

La gran parte de aplicaciones web funcionan desde la nube. Un ejemplo común son los clientes de correo como Gmail, que también cuenta con una aplicación móvil.

Es probable que muchos usuarios prefieran la aplicación móvil, pero le da la facilidad a sus usuarios de elegir la opción que les atraiga más.



## Ventajas de usar HTML5

### □ Es gratuito:

No necesita ningún tipo de programa especial para empezar a programar en HTML5, incluso puede hacerlo en un bloc de notas, guardar el documento como HTML y visualizarlo desde cualquier navegador. Sin embargo, aunque esto es posible no es realmente recomendable pues en un bloc de notas no separa las etiquetas del contenido y puede ser más complicado realizar correcciones.

Puede usar un editor de código gratuito como Notepad++ que ofrece funciones básicas como diferenciación por color entre etiquetas y contenido.

# Ventajas de usar HTML5

## ❑ Código más ordenado

Debido a la adición de nuevas etiquetas que ayudan a nombrar partes de la estructura básica de toda página web (como <header>, por ejemplo), así como la eliminación de ciertas etiquetas, el código HTML se puede separar fácilmente entre etiquetas y contenido, permitiendo así que el desarrollador pueda trabajar de manera más efectiva y detectar errores de manera más rápida.

Las etiquetas son claras y descriptivas, de modo que el desarrollador puede comenzar a codificar sin ningún problema.

# Ventajas de usar HTML5

## ❑ Compatibilidad en navegadores

Los navegadores modernos y populares como Chrome, Firefox, Safari y Opera soportan HTML5. El único problema sería considerar a usuarios que emplean navegadores más antiguos, ya que en éstos no todas las nuevas funciones y etiquetas de HTML5 están disponibles.

## ❑ Almacenamiento mejorado

Otra nueva adición en HTML5 ha sido el almacenamiento local que se define a sí mismo como “mejor que las cookies” pues la información nunca se transfiere al servidor. De esta manera, la información se mantiene segura.

Permite que la información se mantenga almacenada incluso después de haber cerrado el navegador y como funciona desde el lado de cliente, la información se mantiene a salvo incluso si el usuario decide borrar sus cookies.

# Ventajas de usar HTML5

## □ HMTL5 y el diseño adaptativo

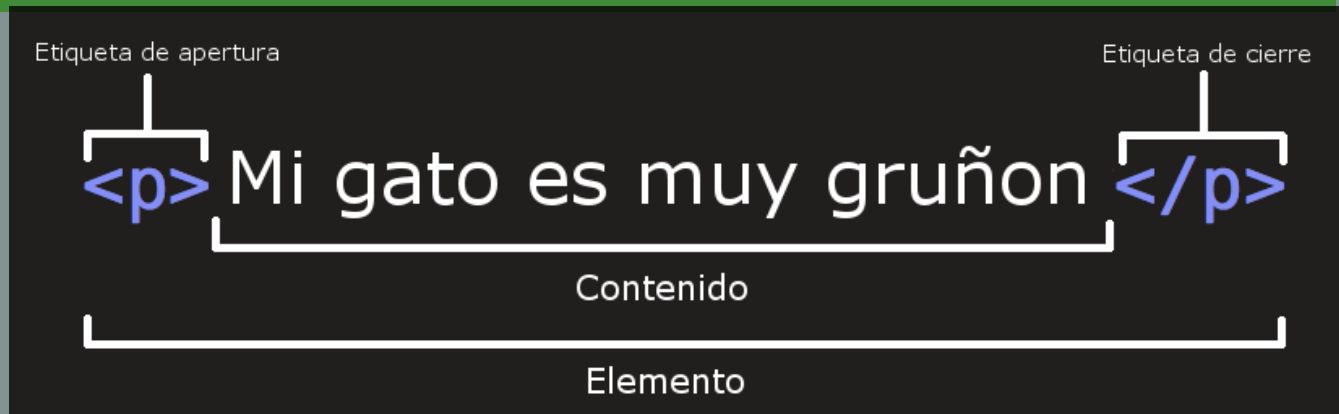
HTML5 es compatible con los navegadores móviles, de modo que cada página realizada en HTML5 que se ve en ordenadores, también se puede adaptar a los dispositivos móviles.

Esta especificación para móviles puede hacerla desde el mismo documento HTML o puede emplear una framework especializada para mejorar tu productividad.

.



# Anatomía de un elemento HTML



Las partes principales del elemento son:

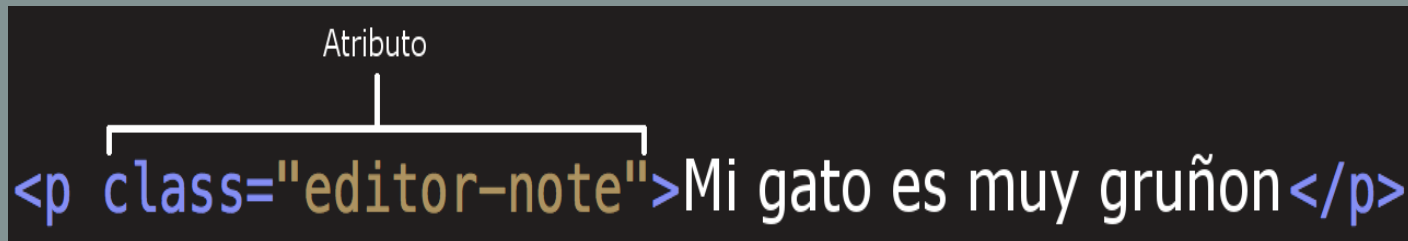
- **La etiqueta de apertura**: consiste en el nombre del elemento (en este caso, `p`), encerrado por **paréntesis angulares** (`<` `>`) de apertura y cierre. **Establece dónde comienza o empieza a tener efecto el elemento**, en este caso, dónde es el comienzo del párrafo.

# Anatomía de un elemento HTML

- **La etiqueta de cierre:** es igual que la etiqueta de apertura, excepto que incluye una barra de cierre (/) antes del nombre de la etiqueta. Establece dónde termina el elemento, en este caso, dónde termina el párrafo.
- **El contenido:** este es el contenido del elemento, que en este caso es sólo texto.
- **El elemento:** la etiqueta de apertura, más la etiqueta de cierre, más el contenido equivale al elemento.

# Anatomía de un elemento HTML

Los elementos pueden también tener atributos, que se ven así:



Los atributos contienen información adicional acerca del elemento, la cual no deseamos que aparezca en el contenido real del elemento.

En este caso, el atributo *class* permite darle al elemento un nombre identificativo, que puede ser usado luego para apuntarle al elemento información de estilo y demás.

# Anatomía de un elemento HTML

Un atributo debe tener siempre:

- Un espacio entre éste y el nombre del elemento (o del atributo previo, si el elemento ya posee uno o más atributos).
- El nombre del atributo, seguido por un signo de igual (=).
- Comillas de apertura y de cierre, encerrando el valor del atributo

## Anidando elementos

Se pueden colocar elementos dentro de otros elementos, esto se llama anidamiento.

Por ejemplo, resaltar una palabra del texto (en nuestro ejemplo la palabra "muy"), podemos encerrarla en un elemento `<strong>`, que significa que dicha palabra debe ser enfatizada:

# Anatomía de un elemento HTML

## Anidando elementos

`<p>Mi gato es <strong>muy</strong> gruñon.</p>`

Debes asegurarte que los elementos estén correctamente anidados: en el ejemplo de abajo, creamos la etiqueta de apertura del elemento `<p>` primero, luego la de `<strong>`, por lo tanto, debemos cerrar ésta etiqueta primero, y luego la de `<p>`. Esto es incorrecto:

`<p>Mi gato es <strong>muy gruñon.</p> </strong>`

Los elementos deben abrirse y cerrarse ordenadamente, de forma tal que se encuentren claramente dentro o fuera el uno del otro.

# Anatomía de un elemento HTML

## Elementos vacíos

Algunos elementos no poseen contenido, y son llamados **elementos vacíos**.

Ejemplo: el elemento [<img>](#) de nuestro HTML:

```

```

Posee dos atributos, pero no hay etiqueta de cierre `</img>` ni contenido encerrado. Esto es porque un elemento de imagen no encierra contenido al cual afectar. Su propósito es desplegar una imagen en la página HTML, en el lugar en que aparece.

### Observación:

- “**src**” es imprescindible para poder colocar una imagen. Indica dónde se encuentra alojada la imagen a mostrar. Se escribe: ``, (“x” la dirección o la url dónde se encuentra situada la imagen).
- Puede agregarse una descripción de la imagen dentro de la misma para pueda leerse al mantener el ratón encima de ella. Se usa el atributo “**alt**” .

# Esquema mínimo de un documento HTML

- Un documento HTML comienza siempre con la etiqueta `<HTML>`, que indica que el documento en cuestión está construido con dicho lenguaje.
- La mayoría de las etiquetas son pareadas, es decir, `<...>` corresponde al principio de la acción y `</...>` indica el fin de dicha acción.
- Una página web estará siempre contenida entre las etiquetas `<HTML>` y `</HTML>`.

Por otra parte, todo documento HTML consta de dos partes:

la cabecera (**head**) y el cuerpo del documento (**body**).

- La cabecera contiene básicamente información destinada al browser (o navegador), que queda oculta al usuario. Su etiqueta (pareada) es `<HEAD>`.
- El cuerpo es el documento que ve el usuario. Su etiqueta (pareada) es `<BODY>`.

# La etiqueta básica

La etiqueta básica, que controla el tipo de fuente utilizado, es `<FONT>`.

`<FONT>` es una etiqueta pareada.

Afecta a los caracteres introducidos entre `<FONT>` y `</FONT>`.

Se puede especificar el tipo de letra (es decir, la fuente de caracteres) añadiéndolo a la etiqueta, de la siguiente manera:

```
<FONT FACE="Arial">...</FONT>
```



# La etiqueta básica

Se puede especificar el color de la fuente con la etiqueta

`<FONT COLOR="?">...</FONT>`

Para especificar los colores:

- Cada color posee su correspondiente código.
- Hay dos códigos: hexadecimal, lo habitualmente usado o RGB. En ambos casos se trata de expresar las proporciones de tres colores básicos: rojo, verde y azul.
- Los valores hexadecimales de los tres colores se expresan de la siguiente manera:
  - Rojo: `color:#ff0000;`
  - Verde: `color:#00ff00;`
  - Azul: `color:#0000ff;`

# La etiqueta básica

- Los mismos colores, expresados con los valores RGB (red, green, blue), son los siguientes:

- Rojo: `color:rgb(255,0,0);`
- Verde: `color:rgb(0,255,0);`
- Azul: `color:rgb(0,0,255);`

- Todos los colores se expresan mediante combinaciones de estos tres colores básicos.

Los colores más básicos pueden especificarse sin código, simplemente escribiendo la palabra correspondiente (en inglés).

Por ejemplo:

`<FONT COLOR="blue">...</FONT>`, `<FONT COLOR="red">...</FONT>`

# La etiqueta básica

Se puede especificar el tamaño de la fuente con la etiqueta

`<FONT SIZE="?">...</FONT>`

El valor del tamaño ("?") deseado puede suministrarse de manera absoluta (un número del 1 al 7) o relativa (+1, +2, -1, -2, etc., teniendo en cuenta que el valor por defecto es 3).

Todas estas etiquetas se pueden combinar, de modo que con una sola se controla

Ejemplo:

el tipo, el tamaño y el color de los caracteres:

`<FONT FACE="Arial" COLOR="#0000FF" SIZE="5">...</FONT>.`

# La etiqueta básica

Luego, la Sintaxis básica de las etiquetas HTML:

`<ETIQUETA [ATRIBUTO="valor"]>.`

Lo anterior quiere decir que una etiqueta HTML (por ejemplo, FONT) puede llevar uno o varios atributos (FACE, COLOR, SIZE), cuyos valores se colocan entrecomillados tras el signo igual (= "arial", = "#0000FF", etc.).

# Etiquetas de formato de caracteres

Etiquetas más comunes:

Código	Función
<code>&lt;B&gt;...&lt;/B&gt;</code>	Negrita
<code>&lt;I&gt;...&lt;/I&gt;</code>	Cursiva
<code>&lt;U&gt;...&lt;/U&gt;</code>	Subrayado
<code>&lt;SUB&gt;...&lt;/SUB&gt;</code>	Subíndice
<code>&lt;SUP&gt;...&lt;/SUP&gt;</code>	Superíndice

- Existe otra etiqueta similar a `<B>`, menos utilizada porque depende de las opciones de configuración del navegador: `<STRONG>...</STRONG>`.
- Existe otra etiqueta similar a `<I>`, menos utilizada porque depende de las opciones de configuración del navegador: `<EM>...</EM>`.
- Conviene tener cuidado a la hora de manejar el subrayado, puesto que éste se utiliza convencionalmente para indicar los hiperenlaces, y podría resultar confuso.

# Otras Etiquetas de caracteres

Hay muchas otras etiquetas que afectan a los caracteres.

- La etiqueta `<PRE>...</PRE>` indica que el texto afectado está preformateado, de modo que el navegador sí tiene en cuenta todos los espacios y saltos de línea definidos en la pantalla. Luego el texto a la vista en el navegador será igual a la del bloc de notas donde lo escribiéramos.
- La etiqueta `<S>...</S>` genera texto tachado.
- La etiqueta `<TT>...</TT>` genera texto similar al de una máquina de escribir (fuente courier de paso fijo).
- La etiqueta `<BIG>...</BIG>` aumenta el tamaño de letra (equivale a `<FONT SIZE="+1">`).
- La etiqueta `<SMALL>...</SMALL>` disminuye el tamaño de letra (equivale a `<FONT SIZE="-1">`).

# Los párrafos de un documento HTML

La etiqueta `<BR>` inserta un salto de línea, pero no un salto de párrafo.

En cambio, la etiqueta `<P>...</P>` inserta un salto de línea y una línea en blanco, por lo que, en la práctica, equivale a un nuevo párrafo.

Tanto `<BR>` como `<P>...</P>` pueden repetirse varias veces seguidas para dejar más distancia entre las líneas y los párrafos.

Un párrafo puede alinearse introduciendo los siguientes atributos en la etiqueta habitual:

Código	Función
<code>&lt;P align="left"&gt;...&lt;/P&gt;</code>	Párrafo alineado a la izquierda
<code>&lt;P align="center"&gt;...&lt;/P&gt;</code>	Párrafo centrado
<code>&lt;P align="right"&gt;...&lt;/P&gt;</code>	Párrafo alineado a la derecha
<code>&lt;P align="justify"&gt;...&lt;/P&gt;</code>	Párrafo justificado

# Los párrafos de un documento HTML

Las etiquetas anteriores permiten alinear los párrafos enteros, pero siempre dejando una línea vacía delante de cada uno de ellas.

Para alinear independientemente líneas o grupos de líneas:

Código	Función
<code>&lt;DIV align="left"&gt;...&lt;/DIV&gt;</code>	Línea(s) alineada(s) a la izquierda
<code>&lt;DIV align="center"&gt;...&lt;/DIV&gt;</code>	Línea(s) centrada(s)
<code>&lt;DIV align="right"&gt;...&lt;/DIV&gt;</code>	Línea(s) alineada(s) a la derecha
<code>&lt;DIV align="justify"&gt;...&lt;/DIV&gt;</code>	Línea(s) alineada(s) justificadas



# Los párrafos de un documento HTML

Pueden utilizarse hasta seis tipos de títulos en un documento HTML, ordenados jerárquicamente por medio de la etiqueta `<Hn>...</Hn>`, donde  $n = 1$  a 6.

Las etiquetas de los títulos introducen una línea superior en blanco.

Como lo habitual después de un título es un nuevo párrafo (`<P>`), que también introduce su propia línea, el resultado final suele ser que una línea de título tiene una línea en blanco antes y otra después.

Las etiquetas de los títulos admiten indicaciones de alineación, de la misma manera que la etiqueta de párrafo.

# Los párrafos de un documento HTML

## EJEMPLO

```
<HTML>
<HEAD>
<TITLE>Ejemplo 7</TITLE>
</HEAD>
<BODY>
<H1> Título 1</H1>
<H2> Título 2</H2>
<H3> Título 3</H3>
<H4> Título 4</H4>
<H5> Título 5</H5>
<H6> Título 6</H6>

<H7> Título 7 (como no existe, no supone ningún cambio)</H7>

<P>Además, los títulos pueden recibir indicaciones de alineación, como en los siguientes ejemplos</P>

<H3 align="center"> Título centrado</H3>
<H3 align="right"> Título a la derecha</H3>
</BODY>
</HTML>
```

Título uno

Título dos

Título tres

Título cuatro

Título cinco

Título seis

Título siete (como no existe, no supone ningún cambio)

Además, los títulos pueden recibir indicaciones de alineación, como en los siguientes ejemplos:

Título centrado

Título a la derecha

# Los párrafos de un documento HTML

## Las listas

Las listas son elementos que, como los títulos, permiten presentar niveles jerarquizados de texto.

Los documentos HTML permiten dos tipos de listas:

- Listas no ordenadas: su etiqueta principal es `<UL>...</UL>`.
- Listas ordenadas o numeradas: su etiqueta principal es `<OL>...</OL>`.
- Listas de definiciones: su etiqueta principal es `<DL>...</DL>`

Una vez definido el tipo de lista deseada, los elementos que la componen se marcan de distinta manera según el tipo:

- En las listas ordenadas (`<OL>...</OL>`) o no ordenadas (`<UL>...</UL>`), cada elemento de dicha lista se marca con la etiqueta `<LI>...</LI>` (list item).
- En las listas de definición (`<DL>...</DL>`), se utilizan dos: cada elemento que se define se marca con `<DT>...</DT>` y el texto de cada definición se marca con `<DD>...</DD>`.

```
<HTML>
<HEAD>
<TITLE>Ejemplo 8</TITLE>
</HEAD>
<BODY>
<H3>Los meses de primavera</H3>
<UL>
<LI>Abril</LI>
<LI>Mayo</LI>
<LI>Junio</LI> </UL>
<BR>
<H3>Los meses de verano</H3>
<OL>
<LI>Julio</LI>
<LI>Agosto</LI>
<LI>Septiembre</LI> </OL> <BR>
<H3>Los meses del año</H3>
<UL>
<LI>Los meses de primavera
<OL>
<LI>Abril</LI>
<LI>Mayo</LI>
<LI>Junio</LI> </OL> </LI>
```

Listas no ordenadas

Listas ordenadas

```
<LI>Los meses de verano
<OL>
<LI>Julio</LI>
<LI>Agosto</LI>
<LI>Septiembre</LI>
</OL> </LI> </UL>
<BR>
<H3>Refranes de los meses</H3>
<DL>
<DT>Marzo</DT>
<DD>El sol de marzo da con el mazo.</DD>
<DT>Abril</DT>
<DD>El abril, aguas mil.</DD>
<DT>Mayo</DT>
<DD>Hasta el 4 de mayo no te quites el sayo.
</DD>
</DL>
</BODY>
</HTML>
```

Listas de definiciones

### **Los meses de primavera**

- Abril
- Mayo
- Junio

### **Los meses de verano**

1. Julio
2. Agosto
3. Septiembre

### **Los meses del año**

- Los meses de primavera
  1. Abril
  2. Mayo
  3. Junio
- Los meses de verano
  1. Julio
  2. Agosto
  3. Septiembre

### **Refranes de los meses**

Marzo

El sol de marzo da con el mazo.

Abril

En abril, aguas mil.

Mayo

Hasta el 40 de mayo no te quites el sayo.

# Los párrafos de un documento HTML

## Las rayas horizontales

Las rayas horizontales son otro de los procedimientos habituales para separar los contenidos de un documento HTML.

Se introducen simplemente con la etiqueta `<HR>`.

Características:

- El tamaño: la etiqueta `<HR>` introduce una raya que ocupa todo el ancho de la pantalla.

Pueden especificarse distintas anchuras añadiendo el valor deseado, expresado en píxeles (`<HR width="n">`) o en términos porcentuales (`<HR width="n%">`)

- La alineación: cuando una raya ocupa sólo una parte de la pantalla, puede especificarse su alineación por medio de las etiquetas habituales:

`<HR align="left">`

`<HR align="center">`

`<HR align="right">`

# Los párrafos de un documento HTML

- La altura: se expresa en píxeles por medio de la etiqueta `<HR size="n">`  
Puede adoptar el valor 0 (la raya más fina).
- El color : por medio del parámetro habitual: `<HR color="?">`
- El sombreado: la raya posee habitualmente un pequeño efecto de sombreado, que puede eliminarse con la etiqueta `<HR noshade>`.
- Lo habitual suele ser la combinación de varios parámetros en una misma etiqueta.

Por ejemplo:

```
<HR width="50%" size="0" align="center">
```

# Los enlaces de un documento HTML

Un hiperenlace es un vínculo que une un elemento del documento activo (por ejemplo, una palabra) con otro, de manera que activamos ese segundo elemento simplemente con un click del ratón en el primero.

La sintaxis general de la etiqueta es muy simple:

```
<A HREF="destino">...</A>.
```

(Los puntos suspensivos representan el elemento enlazado.)

Para que el destino se abra en una ventana nueva, hay que indicar el siguiente atributo:

```
<A HREF="destino" target="_blank">...</A>
```



# Los enlaces de un documento HTML

El destino de un hiperenlace puede ser de tres tipos:

- Enlace externo: otro documento HTML situado en un ordenador remoto.
- Enlace local: otro documento HTML situado en el mismo ordenador.
- Ancla: otro lugar dentro del mismo documento HTML.

## Los enlaces externos

Desde cualquier punto de un documento HTML puede establecerse un enlace con cualquier recurso de Internet.

Los recursos más habituales de Internet son:

- La WWW, es decir, las páginas web o documentos HTML.

La etiqueta correspondiente es:

```
<A HREF="http://servidor/camino.../fichero">...</A>
```

# Los enlaces de un documento HTML

- El correo electrónico.

La etiqueta correspondiente es:

```
<A HREF="mailto:usuario@servidor">...</A>
```

- La transferencia de ficheros (servidores FTP).

La etiqueta correspondiente es:

```
<A HREF="ftp://servidor/camino.../fichero">...</A>
```

## **Los enlaces locales**

La mayor parte de las "páginas web" son, en rigor, una colección de páginas, pues to que la organización clásica de los sitios web consiste en un conjunto ordenado de documentos HTML entrelazados y residentes en un mismo ordenador.

# Los enlaces de un documento HTML

## Los enlaces locales

Dentro de un mismo ordenador, los documentos HTML pueden residir en un mismo directorio o en varios:

- Cuando todos los documentos residen en el mismo directorio no hay ningún problema, y la etiqueta será, simplemente:

`<A HREF="fichero.htm">...</A>`

- Cuando los ficheros residen en diferentes directorios, hay que facilitar la dirección completa del fichero, lo cual puede hacerse de manera absoluta o relativa:

- Direccionamiento absoluto: se indica la trayectoria completa del archivo en cuestión: `<A HREF="file:///c:/directorio/subdirectorios/fichero.htm">...</A>`

- Direccionamiento relativo: no se indican los nombres de los directorios, :  
`<A HREF="../..fichero.htm">...</A>`

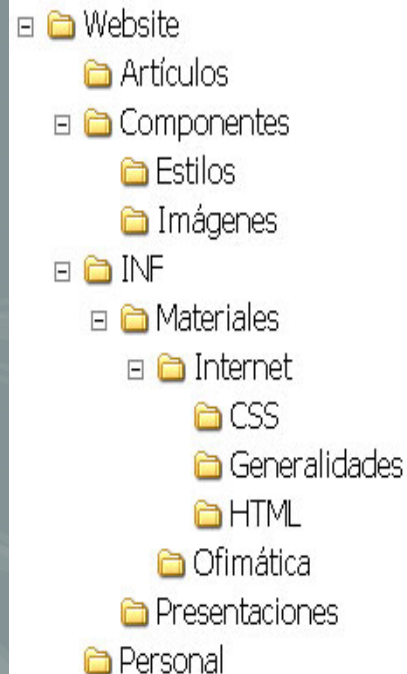
# Los enlaces de un documento HTML

## Los enlaces locales

Aun utilizando un direccionamiento relativo, es necesario indicar los nombres de los directorios "descendentes".

Por ejemplo, en una web ordenada, un enlace desde un documento situado en el directorio <HTML> a un recurso situado en el directorio <Imágenes> debería especificarse así:

```
<A HREF=../../../Componentes/Imágenes/fichero.htm">...</A>
```



# Los enlaces de un documento HTML

## Las anclas

Especialmente cuando el documento es largo, conviene utilizar enlaces que apunten hacia otro lugar dentro del mismo documento. Estos enlaces se denominan **anclas**, **anclajes** o **apuntadores**.

El procedimiento de enlace es doble:

- En primer lugar, hay que establecer el punto de anclaje en el lugar del documento que vaya a ser el destino del enlace.

Su etiqueta es `<A NAME=nombre>...</A>`.

- Después sólo queda enlazar los caracteres deseados con el ancla anterior.

Su etiqueta es `<A HREF=#nombre>...</A>`.

# Los enlaces de un documento HTML

## Las anclas

Algunas observaciones sobre las anclas:

- El nombre del ancla es arbitrario, pero lo lógico es que guarde alguna relación con el destino, especialmente cuando hay muchas en un mismo documento.
- Aunque la etiqueta del ancla sea pareada, lo normal es que no incluya nada entre sus dos partes (`<A NAME=nombre>...</A>`), puesto que lo importante es marcar el punto de inicio del ancla, no su final.

Nota:

- <sup>...</sup> indica un superíndice
- <hr> para separar un texto de otro o un párrafo de otro podemos utilizar una línea horizontal de un tamaño o un grosor determinado

```
</BODY>
</HTML>
```

# DOCTYPE

La declaración del tipo de documento (DTD Document Type Declaration), esta sección se ubica en la primera línea del archivo HTML, es decir antes de la marca html.

En HTML 5 se simplifica esta sección con la siguiente sintaxis:

```
<!DOCTYPE HTML>
```

Es importante agregar el DOCTYPE, de esta forma el navegador puede saber que estamos utilizando la especificación del HTML 5.



# TABLAS

La primera etiqueta para crear tablas es `<table>` `</table>`.  
Esta es la etiqueta que define el comienzo y final de una tabla.  
Es una etiqueta de bloque, así que tendrá que situarse fuera de un párrafo.

Ejemplo:

```
<p>Esto es un párrafo anterior a la tabla.</p>
```

```
<table>
```

```
  <!-- Inserta aquí el contenido de la tabla -->
```

```
</table>
```

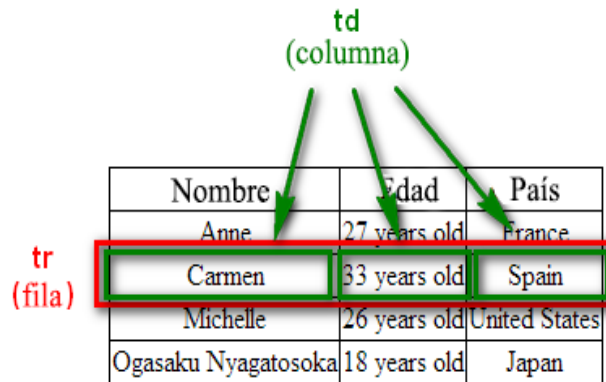
```
<p>Esto es un párrafo posterior a la tabla.</p>
```

# TABLAS

- `<tr>` `</tr>`: define el comienzo y el final de una fila de la tabla.
- `<td>` `</td>`: define el comienzo y el final del contenido de una celda.

En HTML, las tablas se crean fila por fila.

En cada fila (`<tr>`) se especifican los contenidos de las celdas (`<td>`).  
Una tabla se construye básicamente:



Nombre	Edad	País
Anne	27 years old	France
Carmen	33 years old	Spain
Michelle	26 years old	United States
Ogasaku Nyagatosoka	18 years old	Japan

Una tabla, con celdas en línea

# TABLAS

Carmen 33 years old Spain

Michelle 26 years old United States

```
<table>
  <tr>
    <td>Carmen</td>
    <td>33 years old</td>
    <td>Spain</td>
  </tr>
  <tr>
    <td>Michelle</td>
    <td>26 years old</td>
    <td>United States</td>
  </tr> </table>
```



# TABLAS

## Hojas de Estilo

Las hojas de estilo simplifican el código HTML y lo liberan de las responsabilidades de presentación.

Esto da tanto a los autores como a los usuarios control sobre la presentación de los documentos: fuentes, alineación, colores, etc. La información de estilo puede especificarse para elementos individuales o para grupos de elementos.

La información de estilo puede especificarse en un documento HTML o en hojas de estilo externas.

El mecanismo para asociar una hoja de estilo con un documento es independiente del lenguaje de la hoja de estilo.



# TABLAS

## Hojas de Estilo

Antes de la llegada de las hojas de estilo, los autores utilizaban las tablas y las imágenes como medio de organizar la presentación de sus páginas.

Al ofrecer las hojas de estilo mecanismos de presentación más potentes, el W3C declarará obsoletos en el futuro muchos de los elementos y atributos de presentación del HTML, anteriores.

A lo largo de esta especificación dichos elementos y atributos se marcan como "desaprobados".

# TABLAS

## Hojas de Estilo

Retomando el desarrollo de la tabla, observamos que una tabla sin CSS tiene una visualización bastante vacía.

Luego puede usarse una **hoja de estilo** para darle un aspecto más parecido a una tabla:

Código de la Hoja de Estilo:

```
td /* Todas las celdas de la tabla... */  
{  
    border: 1px solid black; /* tendrá un borde de 1 px */  
}
```

Carmen	33 years old	Spain
Michelle	26 years old	United States

Cada celda tiene su propio borde

# TABLAS

## Hojas de Estilo

Hay una propiedad específica para tablas en CSS,

**border-collapse**, que implica «pegar los bordes».

Esta propiedad puede tener dos valores:

**collapse**: los bordes se pegan, es el efecto que estamos buscando;

**separate**: los bordes se separarán (valor por defecto).

table

{

border-collapse: collapse; /\* Los bordes de la tabla se pegarán (mejor aspecto) \*/

}

td

{

border: 1px solid black;

}

Carmen	33 years old	Spain
Michelle	26 years old	United States

Los bordes están pegados

# TABLAS

## La fila de encabezado

Falta añadir la fila de encabezado a la tabla.

En el siguiente ejemplo, los encabezados son «Nombre», «Edad» y «País».

La fila de encabezado se ha creado con una etiqueta `<tr>`, pero las celdas que contiene están esta vez enmarcadas por etiquetas `<th>` y no `<td>`.

```
<table>
  <tr>
    <th>Nombre</th>
    <th>Edad</th>
    <th>País</th>
  </tr>

  <tr>
    <td>Carmen</td>
    <td>33 años</td>
    <td>España</td>
  </tr>
  <tr>
    <td>Michelle</td>
    <td>26 años</td>
    <td>Estados Unidos</td>
  </tr>
</table>
```



# TABLAS

## Hojas de Estilo

La fila de encabezado es fácil de reconocer por dos razones:

- las celdas contienen etiquetas `<th>`, en lugar de las habituales etiquetas `<td>`;
- es la primera fila de la tabla .

Puesto que el nombre de la celda es ligeramente diferente en el encabezado, se debe recordar actualizar el código CSS para indicarle que aplique un borde en celdas normales y en el encabezado.

```
table
{
    border-collapse: collapse;
}
td, th /* Asigna un borde a las etiquetas td Y th */
{
    border: 1px solid black;}
```

Name	Age	Country
Carmen	33 years old	Spain
Michelle	26 years old	United States

Una tabla con encabezado

# TABLAS

## Título de tabla

Normalmente, todas las tablas de ben tener título.

El título informa rápidamente al visitante sobre el contenido de la tabla.

**<caption>**: esta etiqueta se coloca al comienzo de la tabla, justo antes del encabezado. Es la etiqueta que contiene el título de la tabla.

```
<table>
```

```
<caption>Passengers of flight 377</caption>
```

```
<tr>
```

```
<th>Name</th>
```

```
<th>Age</th>
```

```
<th>Country</th>
```

```
</tr>
```

```
<tr>
```

```
<td>Carmen</td>
```

```
<td>33 years old</td>
```

```
<td>Spain</td>
```

```
</tr>
```

```
<tr>
```

```
<td>Michelle</td>
```

```
<td>26 years old</td>
```

```
<td>United States</td>
```

```
</tr> </table>
```

Passengers of flight 377

Name	Age	Country
Carmen	33 years old	Spain
Michelle	26 years old	United States

Una tabla con título

# TABLAS

## Una tabla estructurada

- Las tablas grandes pueden dividirse en tres partes:
  - Encabezado;
  - Cuerpo de la tabla;
  - Pie de tabla.
- En algunas tablas, puede que necesite combinar celdas.

## Dividir una tabla grande

Si la tabla es bastante grande, lo mejor es dividirla en varias partes.

Para esto, las etiquetas HTML se utilizan para definir los tres «campos» de la tabla:

- el encabezado (al principio): se define con las etiquetas `<thead>` `</thead>`;
- el cuerpo (en el centro): se define con las etiquetas `<tbody>` `</tbody>`;
- el pie de tabla (al final): se define con las etiquetas `<tfoot>` `</tfoot>`.

# TABLAS

¿Qué se ubica en el pie de la tabla?

Si es una tabla larga, normalmente se copirán los encabezados.

Esto permite ver a lo que se refiere cada columna, incluso al final de la tabla.

Fundamentalmente, una tabla se divide en tres partes.

<b>table header</b>	<b>Name</b>	<b>Age</b>	<b>Country</b>	<b>&lt;thead&gt;</b>
	Carmen	33 years old	Spain	
	Michelle	26 years old	United States	
<b>table body</b>	Francois	43 years old	France	<b>&lt;tbody&gt;</b>
	Martine	34 years old	France	
	Jonathan	13 years old	Australia	
	Xu	19 years old	China	
<b>table footer</b>	<b>Name</b>	<b>Age</b>	<b>Country</b>	<b>&lt;tfoot&gt;</b>

Una tabla dividida en varias partes

# TABLAS

Es recomendable escribir las etiquetas en el orden siguiente:

**<thead>**

**<tfoot>**

**<tbody>**

Por lo tanto, en el código, primero se define la parte superior, luego la parte final y, por último, la parte principal (<tbody>).

# TABLAS

```
<table>
  <caption>Pasajeros del vuelo 377</caption>

  <thead> <!-- Pasajeros del vuelo 377 -->
    <tr>
      <th>Nombre</th>
      <th>Edad</th>
      <th>País</th>
    </tr>
  </thead>

  <tfoot> <!-- Pie de tabla -->
    <tr>
      <th>Nombre</th>
      <th>Edad</th>
      <th>País</th> </tr> </tfoot>

  <tbody> <!-- Cuerpo de la tabla -->
    <tr>
      <td>Carmen</td>
      <td>33 años</td>
      <td>España</td> </tr>
```

```
<tr>
  <td>Michelle</td>
  <td>26 años</td>
  <td>Estados Unidos</td>
</tr>
<tr>
  <td>François</td>
  <td>43 años</td>
  <td>Francia</td>
</tr>
<tr>
  <td>Martine</td>
  <td>34 años</td>
  <td>Francia</td>
</tr>
<tr>
  <td>Jonathan</td>
  <td>13 años</td>
  <td>Australia</td>
</tr>
<tr>
  <td>Xu</td>
  <td>19 años</td>
  <td>China</td> </tr> </tbody> </table>
```

# TABLAS

No es necesario siempre utilizar estas tres etiquetas (<thead>, <tbody>, <tfoot>) en todas las tablas.

De hecho, las utilizará principalmente si la tabla es muy grande y necesita organizarla de forma más clara.

En tablas «pequeñas», no es necesario

# TABLAS

## Combinar celdas:

Movie title	For children?	For teenagers?
Chainsaw massacre	No, too violent	Yes
The care bears go skiing	Yes, suitable	Not violent enough...
Lucky Luke, goes it alone	For the whole family!	

Una tabla que contiene títulos de películas y su audiencia

Para combinar, se añade un atributo a la etiqueta `<td>`.

Hay dos tipos de combinar:

- Combinar columnas: las celdas se combinan horizontalmente. Se utiliza el atributo **colspan**.
- Combinar filas: en este caso, dos filas se combinan en una. Las celdas se combinan verticalmente. Se utiliza el atributo **rowspan**.

```
<table>
  <tr>
    <th>Título de la película</th>
    <th>¿Para niños?</th>
    <th>¿Para adolescentes?</th>
  </tr>
  <tr>
    <td>La matanza de Texas</td>
    <td>No, demasiado violenta</td>
    <td>Sí</td>
  </tr>
  <tr>
    <td>Los osos amorosos se van a esqui  
ar</td>
    <td>Sí, apropiada</td>
    <td>No hay violencia suficiente...</td>
  </tr>
  <tr>
    <td>Lucky Luke el intrépido</td>
    <td colspan="2">¿Para toda la familia!</td>
  </tr>
</table>
```



# TABLAS

## Combinar verticalmente con rowspan

<b>Film title</b>	Chainsaw massacre	The care bears go skiing	Lucky Luke, goes it alone
<b>For children?</b>	No, too violent	Yes, suitable	For the whole family!
<b>For adolescents?</b>	Yes	Not violent enough...	

Las celdas se han combinado verticalmente

```
<table>
```

```
<tr>
```

```
<th>Título de la película</th>
```

```
<td>La matanza de Texas</td>
```

```
<td>Los osos amorosos se van a  
esquiar</td>
```

```
<td>Lucky Luke el intrépido</td>
```

```
</tr>
```

```
<tr>
```

```
<th>¿Para niños?</th>
```

```
<td>No, demasiado violenta</td>
```

```
<td>Sí, apropiada</td>
```

```
<td rowspan="2">¿Para toda la familia!
```

```
</td>
```

```
</tr>
```

```
<tr>
```

```
<th>¿Para adolescentes?</th>
```

```
<td>Sí</td>
```

```
<td>No hay violencia suficiente...</td>
```

```
</tr> </table>
```

# TABLAS

En resumen:

- Una tabla se inserta con la etiqueta `<table>` y se define fila por fila utilizando la etiqueta `<tr>`.
- Cada fila contiene celdas `<td>` (celdas normales) o `<th>` (celdas de encabezado)
- El título de la tabla se define mediante `<caption>`.
- Puedes añadir un borde a las celdas de la tabla mediante `border`. Para combinar bordes, se usa la propiedad de CSS `border-collapse`.
- Una tabla puede dividirse en tres secciones: `<thead>` (encabezado), `<tbody>` (cuerpo) y `<tfoot>` (pie de la tabla). El uso de estas etiquetas no es obligatorio.
- Puedes combinar celdas horizontalmente mediante el atributo `colspan` o verticalmente con `rowspan`. Debe especificarse cuántas celdas tienen que combinarse.



# A TRABAJAR....

