



UNIDAD 2.-Unidad 2.- HTML y CSS Básico. Parte 1 HTML.

1.- LENGUAJES DE MARCAS, HTML5

- ✓ **HTML** es el lenguaje con el que se escriben las páginas web que pueden ser vistas en un navegador.
- ✓ Este lenguaje nos permite aglutinar textos, sonidos e imágenes y combinarlos o referencias a otras páginas.
- ✓ Un archivo HTML (una página) no es más que un texto. Es por ello que para programar en HTML necesitamos un editor de textos.
- ✓ Actualmente los navegadores ya están diseñados para manejar la mayoría de las funcionalidades recogidas en HTML5. Para comprobar si el código HTML de una página web valida con las reglas HTML5 podemos visitar <http://validator.w3.org>
- ✓ El World Wide Web Consortium (**W3C**) es una comunidad internacional que desarrolla estándares, que desarrolla reglas y convenciones, para la red informática mundial. Liderado por Tim Berners-Lee y su misión es guiar a la Web hacia su máxima potencial.



1.1.- ESTÁNDARES Y VERSIONES

- ✓ HTML fue creado en **1980** por **Tim Berners-Lee** cuando trabajaba en el CERN, pero no fue hasta **1993** que llegó la primera propuesta de estandarización.
- ✓ En noviembre de **1995** se aprobó el estándar **HTML 2.0**, versión del HTML carecía de muchas herramientas que permitieran controlar el diseño de las páginas y añadir contenido multimedia.
- ✓ El borrador de **HTML 3.0** resultó demasiado extenso y complejo, y varias compañías se unieron para formar un nuevo comité encargado de establecer los estándares del HTML.
- ✓ En enero de **1997** se aprobó el estándar **HTML 3.2**. Este nuevo estándar incluía las mejoras propuestas por los navegadores Internet Explorer y Netscape Navigator.
- ✓ En diciembre de **1998** se aprobó el estándar **HTML 4.0**, creado para estandarizar los marcos (frames), las hojas de estilo y los scripts.
- ✓ En septiembre de **2001** se aprobó el **estándar HTML 4.01**.
- ✓ La versión del documento aparece al inicio del archivo:
 - `<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Frameset//EN"
"http://www.w3.org/TR/html4/frameset.dtd">`



1.1.- ESTÁNDARES Y VERSIONES

- ✓ En octubre de **2014** se aprobó el **estándar HTML 5**.
- ✓ HTML 5 es tiene muchas ventajas entre ellas: posibilita la inserción de vídeos y audio de forma directa, la geolocalización del usuario, permite realizar diseños adaptables a distintos dispositivos (web, tablets, móviles...), etc.
- ✓ Para HTML 5 la declaración de la etiqueta se simplifica hasta:
 - `<!DOCTYPE>`

1.2.- NORMAS FUNDAMENTALES

✓ ETIQUETAS

- ✓ El HTML es un lenguaje que basa su sintaxis en etiquetas.
- ✓ Generalmente las etiquetas son **cerradas** y presenta dos partes:
 - ✓ Una apertura de forma general <etiqueta>
 - ✓ Un cierre de tipo </etiqueta>

```
<p> Párrafo bien cerrado </p>  
<b>Esto está en negrita</b>
```

Párrafo bien cerrado

Esto está en negrita

- ✓ Las etiquetas **abiertas** constan solo de apertura y no tienen etiqueta de cierre Ejemplos:
, <hr>, <input>.
- ✓ Nota: En algunas versiones de HTML, como XHTML 1.0 lo correcto sería poner un barra / justo antes del cierre.
.

1.2.- NORMAS FUNDAMENTALES

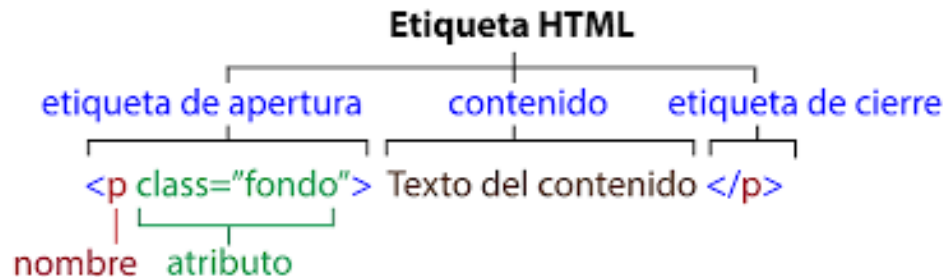
✓ ATRIBUTOS

- ✓ Las etiquetas pueden contener atributos para configurar características de las propias etiquetas donde se indican.

```
<body background="fondo.jpg">  
<a href="mi_fichero.zip">Descarga mi_fichero.zip</a>  
<input type="radio" name="sexo" value="Hembra"> Mujer
```

[Descarga mi fichero.zip](#) ☒ Mujer

- ✓ Estructura de una etiqueta completa HTML:



- ✓ Las mayúsculas o minúsculas son indiferentes al escribir etiquetas y atributos, aunque es conveniente utilizar minúsculas ya que otras tecnologías que sí las diferencian que pueden convivir con HTML (XML por ejemplo).
- ✓ El valor del atributo puede escribirse sin comillas si no contiene espacios, Ya que los espacios separan atributos. ¡Mejor utilizar comillas!

1.2.- NORMAS FUNDAMENTALES

- ✓ **COMENTARIOS**
- ✓ Son líneas de texto que no son interpretadas por el navegador, y sirven para poner aclaraciones en el código.

```
<!-- Esto es un comentario  
|   <p>Esto no se verá en el documento</p>  
-->
```


1.3.- ESTRUCTURA BÁSICA

- ✓ La estructura básica de un documento HTML es la siguiente:

```
<html>
  <head>
    <!-- Etiquetas y contenidos del encabezado
    Datos que no aparecen en nuestra página pero que
    son importantes para catalogarla:
    Título, palabras clave,... -->
    <title>
    </title>
  </head>
  <body>
    <!-- Etiquetas y contenidos del cuerpo
    Parte del documento que será mostrada por el navegador:
    Texto e imágenes-->
  </body>
</html>
```


1.3.- ESTRUCTURA BÁSICA

- CABECERA

- ✓ `<head>...</head>`: delimita la cabecera del documento.
- ✓ Dentro de la cabecera es importante definir el título de la página por medio de la directiva `<title></title>`, que aparecerá en la pestaña de nuestro visor de páginas Web.

```
<title> Mi página Web de videojuegos </title>
```

Mi página Web de videojuegos



1.3.- ESTRUCTURA BÁSICA

- CABECERA

- ✓ Dentro de la cabecera de nuestro documento podemos incluir otras directivas adicionales.
- ✓ La directiva **<meta>** indica al visor de Internet las palabras clave y contenido de nuestra página Web. Utilizado por buscadores (Google, Bing, etc) para incluir la página en sus bases de datos. Utiliza generalmente dos atributos:
 - ✓ **name** que indica qué tipo de etiqueta se usará. Ej:
 - ✓ "description" indica al visor una descripción de la página.
 - ✓ "keywords" indica al visor las palabras clave para los buscadores de Internet.
 - ✓ **content** el contenido de una etiqueta concreta.
 - ✓ **http-equiv**: otro atributo útil para cargar otra web (o la misma) pasado un tiempo.
- ✓ La directiva **<base>** la localización de los ficheros, gráficos, sonidos, etc... Si no se incluye entiende que están en el mismo lugar que el html.

```
<head>
  <title> Mi página Web de videojuegos </title>
  <meta name ="description" content = "blog personal de videojuegos">
  <meta name ="keywords" content = "videogames Xbox PS5">
  <meta http-equiv="refresh" content ="10; url=https://www.playstation.com/es-es/">
  <base href ="http://www.imagenes.com/videojuegos/">
</head>
```



1.3.- ESTRUCTURA BÁSICA

- CUERPO

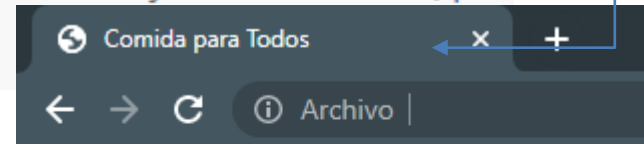
- ✓ La directiva **<body>...</body>** indica el inicio y final de nuestra página Web, y donde pongamos los contenidos de nuestra página, textos, gráficos, enlaces, etc....
- ✓ Antiguamente junto con la etiqueta **<body>** se añadían atributos de apariencia global del documento, tales como: background, bgcolor, text, link, vlink, etc. Estos ya no son soportados en HTML5, dado que la idea es separar el contenido de la forma en que se presenta utilizando CSS.

1.3.- ESTRUCTURA BÁSICA

- **EJEMPLO: Tu primera página**

- ✓ Crea una página web similar a esta con el contenido que prefieras. Guárdala con el nombre de tu página y extensión “.html”
- ✓ **Consejo:** Utiliza nombres en tus archivos que tengan algunas normas básicas para ahorrarte disgustos y líos. El consejo es que no utilices acentos ni espacios ni otros caracteres que no sean números, letras o guiones.

```
<html>
  <head>
    <title> Comida para Todos </title>
  </head>
  <body>
    <p>Estás en la página Comida para Todos.</p>
    <br>
    <p>Aquí aprenderás recetas fáciles y deliciosas.</p>
  </body>
</html>
```



Estás en la página Comida para Todos.

Aquí aprenderás recetas fáciles y deliciosas.

1.4 COLORES

- ✓ Cada color es definido por un número hexadecimal de 6 dígitos que está compuesto a su vez de tres partes (de 8 bits). Cada parte representa la contribución del rojo, verde y azul al color en cuestión (RGB). Algunos colores básicos se pueden utilizar por nombre (entre parentesis).
- ✓ **Nota: a pesar de que los estamos viendo aquí, el formato de color se deja para las hojas de estilo CSS, lo mismo ocurrirá con los estilos .**

Nombres de colores y valores #RGB



Black = "#000000" (Negro)



Silver = "#C0C0C0" (Plateado)



Gray = "#808080" (Gris)



White = "#FFFFFF" (Blanco)



Maroon = "#800000" (Marrón)



Red = "#FF0000" (Rojo)



Purple = "#800080" (Púrpura)



Fuchsia = "#FF00FF" (Fucsia)



Green = "#008000" (Verde)



Lime = "#00FF00" (Verde lima)



Olive = "#808000" (Verde oliva)



Yellow = "#FFFF00" (Amarillo)



Navy = "#000080" (Azul marino)



Blue = "#0000FF" (Azul)



Teal = "#008080" (Azul verdoso)



Aqua = "#00FFFF" (Celeste)

1.5 ESPACIADOS, SALTOS DE LÍNEA Y PÁRRAFOS

- **Parrafos.**

- ✓ La etiqueta `<p>` introduce un salto y deja una línea en blanco antes de continuar con el resto del documento.
- ✓ La etiqueta `
` realiza un simple retorno de carro

```
<p>Este es el texto que forma el primer párrafo de la página.  
Los párrafos pueden ocupar varias líneas y el navegador se encarga  
de ajustar su longitud al tamaño de la ventana.</p>
```

```
<p>El segundo párrafo de la página también se define encerrando  
su texto con la etiqueta p. El navegador también se encarga de  
separar automáticamente cada párrafo.</p>
```

```
<br><br>
```

```
<p>Un tercer párrafo separado con una etiqueta br.</p>
```

Nota.

- ✓ cambiar de línea en nuestro documento HTML sin introducir alguna de estas u otras etiquetas no implica un cambio de línea en la página visualizada.

Este es el texto que forma el primer párrafo de la página. Los párrafos pueden ocupar varias líneas y el navegador se encarga de ajustar su longitud al tamaño de la ventana.

El segundo párrafo de la página también se define encerrando su texto con la etiqueta p. El navegador también se encarga de separar automáticamente cada párrafo.

Un tercer párrafo separado con dos etiquetas br.

1.5 ESPACIADOS, SALTOS DE LÍNEA Y PÁRRAFOS

- Espaciado.

- ✓ Cualquier espacio, tabulación, o salto de línea, o conjunto de ellos:
 - ✓ Fuera de una etiqueta, no se mostrará en el navegador.
 - ✓ Dentro de una etiqueta se mostrarán como un único espaciado.

```
<p>Este primer párrafo no contiene saltos de línea ni otro tipo de espaciado.</p>
```

```
<p>Este segundo párrafo sí que contiene saltos  
de      línea  
y  otro  tipo  de  espaciado que no será mostrado.</p>
```

```
<p>Este tercer párrafo sí que contiene saltos <br/>  
de <br/>  
línea <br/>  
y &nbsp;&nbsp;&nbsp;&nbsp;&nbsp; otro &nbsp;&nbsp;&nbsp; tipo &nbsp;&nbsp;&nbsp; de espaciado &nbsp;&nbsp;&nbsp; que se mostrará.</p>
```

Un tercer párrafo separado con una etiqueta br.

Este primer párrafo no contiene saltos de línea ni otro tipo de espaciado.

Este segundo párrafo sí que contiene saltos de línea y otro tipo de espaciado que no será mostrado.

Este tercer párrafo sí que contiene saltos
de
línea
y otro tipo de espaciado que se mostrará.

Nota: Si queremos añadir varios espacios dentro de una etiqueta se utiliza ** ** (Non Breaking Space - espacio sin ruptura).

1.5 ESPACIADOS, SALTOS DE LÍNEA Y PÁRRAFOS

- **Parrafos.**

- ✓ Los párrafos delimitados por etiquetas <p>.
- ✓ Mediante CSS podrán ser justificados a la izquierda, centro o derecha. Veremos cómo hacerlo

```
<p>Texto alineado a la izquierda</p>  
<p>Texto alineado al centro</p>  
<p>Texto alineado a la derecha</p>
```

Texto alineado a la izquierda

Texto alineado al centro

Texto alineado a la derecha

Ejemplo práctico 1:

Para practicar un poco lo que acabamos de ver, utiliza tu primera página que hiciste para incorporar por este orden:

- 1 Párrafos centrados
- 1 Párrafos alineados a la derecha
- Un salto de línea triple
- 1 párrafo alineado a la izquierda

1.5 ESPACIADOS, SALTOS DE LÍNEA Y PÁRRAFOS

- Texto preformateado.
- ✓ HTML elimina cualquier espacio en blanco adicional que se inserte excepto cuando utilizamos la etiqueta `<pre>.....</pre>`. Aunque convertirá el texto afectado a fuente monoespaciada (posiblemente Courier).

```
La etiqueta pre
respeta los espacios en blanco
y
muestra el texto
tal y como
está escrito
```

```
<pre>
La etiqueta pre
respeta los espacios en blanco
y
muestra el texto
tal y como
está escrito
</pre>

<p>
La etiqueta pre
respeta los espacios en blanco
y
muestra el texto
tal y como
está escrito
</p>
```

La etiqueta pre respeta los espacios en blanco y muestra el texto tal y como está escrito

1.5 ESPACIADOS, SALTOS DE LÍNEA Y PÁRRAFOS

- **Contenedores o Layers (capa): <div>**
 - ✓ Es una etiqueta que permite agrupar elementos dentro de ella, los cuales tendrán el formato que tenga <div>, a menos que se especifique otra cosa.

```
<div>
  <p>Parrafo1 mantiene alienación que tenga de div (por defecto izquierda)</p>
  <p>Parrafo2 igual a Parrafo1</p>
  <p>Parrafo3 se ha especificado con alineación a la derecha (mediante CSS).</p>
</div>
```

Parrafo1 mantiene alienación que tenga de div (por defecto izquierda)

Parrafo2 igual a Parrafo1

Parrafo3 se ha especificado con alineación a la derecha (mediante CSS).

1.5 ESPACIADOS, SALTOS DE LÍNEA Y PÁRRAFOS

- **Contenedores o Layers (capa): **
 - ✓ Es un contenedor en línea.
 - ✓ Sirve para aplicar estilo al texto o agrupar elementos en línea
 - ✓ Crea una caja
 - ✓ Es habitual usarla junto con los atributos id y class para modificar posteriormente su aspecto con CSS, en lugar de hacerlo en el HTML (como en la imagen).

```
<div>
  <p>Este div tiene background gris
  mediante CSS.<br>
  Los span de los párrafos
  tienen color gracias a CSS.
  </p>
  <p><span>Este</span> es el
  Parrafo1</p>
  <p>Este <span>es el</span>
  Parrafo2</p>
  <p>Este es el <span>
  Parrafo3</span></p>
</div>
```

Este div tiene background gris mediante CSS.
Los span de los párrafos tienen color gracias a CSS.

Este es el Parrafo1

Este es el Parrafo2

Este es el Parrafo3

1.6.- ENCABEZADOS

- `<hX>`
- ✓ Existen otras etiquetas para definir párrafos especiales, formateados como títulos. Son los encabezados (header) en inglés, que tienen un tamaño mayor de letra y colocan el texto en negrita.
- ✓ Hay varios tipos de encabezados con diferentes tamaños desde `<h1>` para los encabezados más grandes hasta `<h6>` para los encabezados más pequeños.

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

Nota: todos los encabezados se colocarán en un párrafo independiente.

Encabezado de nivel 1

Encabezado de nivel 2

Encabezado de nivel 3

Encabezado de nivel 4

Encabezado de nivel 5

Encabezado de nivel 6

Titular de la página

Párrafo de introducción...

La primera sub-sección

Párrafo de contenido...

Otra subsección

Más párrafos de contenido...



1.7.-LISTAS DE ELEMENTOS

- ✓ Las listas son utilizadas para citar, numerar y definir objetos. También son utilizadas corrientemente para desplazar el comienzo de línea hacia la derecha.
- ✓ Podemos distinguir tres tipos de listas:
 - ✓ Listas desordenadas
 - ✓ Listas ordenadas
 - ✓ Listas de definición

1.7.-LISTAS DE ELEMENTOS

- Listas desordenadas

- ✓ Delimitadas por las etiquetas ` ... ` (unordered list). Cada uno de los elementos de la lista es citado por medio de una etiqueta `` (list ítem).
- ✓ La viñeta podrá ser modificada mediante CSS.

```
<p>Países del mundo</p>
<ul>
  <li>Francia</li>
  <li>Alemania</li>
  <li>Gran Bretaña</li>
  <li>España</li>
</ul>
```

Países del mundo

- Francia
- Alemania
- Gran Bretaña
- España

1.7.-LISTAS DE ELEMENTOS

- Listas ordenadas

- ✓ Delimitadas por las etiquetas ` ... ` (ordered list) Cada elemento será igualmente precedido de su etiqueta ``.
- ✓ Es posible fijar el valor con el se empieza con el atributo **start**. Mediante CSS será posible también utilizar numeración, alfabeto en mayúsculas o minúsculas o números romanos.

```
<p>Ordenamos por numeros</p>
<ol type="1">
  <li>Elemento 1</li>
  <li> Elemento 2</li>
</ol>
```

```
<p>Ordenamos por números
empezando por el 10</p>
<ol start="10">
  <li>Elemento 10</li>
  <li> Elemento 11</li>
</ol>
```

Ordenamos por numeros

1. Elemento 1
2. Elemento 2

Ordenamos por números empezando por el 10

10. Elemento 10
11. Elemento 11

1.7.-LISTAS DE ELEMENTOS

- Listas de definición

- ✓ Delimitadas por las etiquetas `<dl> ... </dl>` (definition list). Cada elemento es presentado junto con su definición. La etiquetas del elemento y su definición son `<dt>` (definition term) y `<dd>` (definition definition).

```
<p>Different languages:</p>
<dl>
  <dt>HTML</dt>
  <dd>is a markup language</dd>
  <dt>Java</dt>
  <dd>is a programming language and platform</dd>
  <dt>JavaScript</dt>
  <dd>is a scripting language</dd>
  <dt>SQL</dt>
  <dd>is a query language</dd>
</dl>
```

Different languages:

HTML

is a markup language

Java

is a programming language and platform

JavaScript

is a scripting language

SQL

is a query language

1.7.-LISTAS DE ELEMENTOS

- Anidando listas
 - ✓ Es posible anidar listas de forma mixta.

```
<p>Países y ciudades del mundo</p>
<ol>
  <li>Francia
    <ul>
      <li>París</li>
      <li>Toulouse</li>
    </ul>
  </li>

  <li>Alemania
    <ul>
      <li>Berlín</li>
      <li>Bonn</li>
    </ul>
  </li>

  <li>España
    <ul>
      <li>Valencia</li>
      <li>Madrid</li>
    </ul>
  </li>
</ol>
```

Países y ciudades del mundo

1. Francia
 - París
 - Toulouse
2. Alemania
 - Berlín
 - Bonn
3. España
 - Valencia
 - Madrid

A decorative vertical bar on the left side of the slide, composed of numerous overlapping circles of various sizes and colors, including blue, green, yellow, orange, and pink, creating a vibrant, bubbly effect.

Práctica

- Realiza la Práctica 1

1.8.- CARACTERES ESPECIALES

- ✓ Una página web se ha de ver en países distintos, que usan conjuntos de caracteres distintos. El lenguaje HTML nos ofrece un mecanismo para que los caracteres especiales se vayan a ver bien en todos los ordenadores del mundo, y navegador independientemente de su juego de caracteres.

Result	Description	Entity Name
	non-breaking space	
<	less than	<
>	greater than	>
&	ampersand	&
"	double quotation mark	"
'	single quotation mark (apostrophe)	'
¢	cent	¢
£	pound	£
¥	yen	¥
€	euro	€
©	copyright	©
®	registered trademark	®

Nota: Para acentos nos basta con utilizar un character set compatible:
<meta http-equiv="Content-Type" content="text/html" charset="UTF-8" />

1.9.- VÍNCULOS O HIPERENLACES

- ✓ Los enlaces se detectan en una página, ya que al deslizar el puntero del ratón sobre las imágenes o el texto, cambia generalmente de puntero a señalador.
- ✓ Si no especificamos lo contrario, estos enlaces de texto estarán subrayados y coloreados en azul. En el caso de las imágenes que sirvan de enlace, veremos que están delimitadas por un marco azul por defecto.
- ✓ Para colocar un enlace se utiliza etiquetas `<a> ... `. Dentro de la etiqueta el atributo **href** = “destino” especifica el destino del enlace, y contenido un texto o una imagen .

```
<a href="destino">contenido</a>
```

- ✓ Además, entre otros atributos, el más interesante es **target** que especifica si la nueva página se cargará una nueva pestaña (`_blank`), en el frame superior al del enlace (`_parent`), en el frame del enlace (`_self`), o en la pestaña actual (`_top`).
- ✓ Según este son agrupados del siguiente modo:
 - ✓ Enlaces internos: los que se dirigen a otras partes dentro de la misma página.
 - ✓ Enlaces locales: los que se dirigen a otras páginas del mismo sitio web.
 - ✓ Enlaces remotos: los dirigidos hacia páginas de otros sitios web.
 - ✓ Enlaces con direcciones de correo: para crear un mensaje de correo dirigido a una dirección.
 - ✓ Enlaces con archivos: para que los usuarios puedan hacer download de ficheros.

1.9.- VÍNCULOS O HIPERENLACES

- **Enlaces internos**

- ✓ Son los enlaces que apuntan a un lugar diferente dentro de la misma página. Son utilizados documentación o índices para ir la sección deseada.
- ✓ Como destino se utiliza: “#nombre_sección”. La sección en cuestión deberá tener el in identificador **id** “nombre_sección”.

```
<p>Sección 1</p>
<a href="#seccion2">Ir Sección 2</a>
<br>.<br>.<br>.<br>.<br>.<br>
<p>Algo de texto de la sección 1.</p>
<br>.<br>.<br>.<br>.<br>.<br>

<p id="seccion2">Sección 2</a>
```

Sección 1

[Ir Sección 2](#)

.
. .
. .
. .
. .

Algo de texto de la sección 1.

.
. .
. .
. .
. .

Sección 2

1.9.- VÍNCULOS O HIPERENLACES

- **Enlaces locales**

- ✓ Como hemos dicho, un sitio web está constituido de páginas enlazadas.
- ✓ Los sitios suelen estar ordenados por directorios, a los que accederemos a otras páginas, imágenes, sonidos...
- ✓ En los enlaces tendremos que especificar la ruta al recurso, utilizando barras “/”:
 - ✓ Si el archivo está en el mismo directorio de la página no es necesario especificar la ruta.
 - ✓ ./ indica el directorio actual
 - ✓ ../ indica el directorio superior
 - ✓ Si el archivo está un subdirectorio, navegaremos hasta el con “/”.

```
<p>Estoy en la página: pagina1.html</p>
<a href="pagina2.html">Este enlace va a pagina2.html que está en
    el mismo directorio que pagina1.html</a><br>
<a href="../dir/pagina3.html">Este enlace va a pagina3.html en subdirectorio</a><br>
<a href="../../pagina4.html">Este enlace va a pagina4.html que está en el directorio
    superior al de pagina1.html</a>
```

Estoy en la página: [pagina1.html](#)

[Este enlace va a pagina2.html que está en el mismo directorio que pagina1.html](#)

[Este enlace va a pagina3.html en subdirectorio](#)

[Este enlace va a pagina4.html que está en el directorio superior al de pagina1.html](#)

1.9.- VÍNCULOS O HIPERENLACES

- Enlaces remotos

- ✓ Son los enlaces que se dirigen hacia páginas que se encuentran fuera de nuestro sitio web, es decir, cualquier otro documento que no forma parte de nuestro sitio.
- ✓ todas las direcciones web (URLs) empiezan por **http://** También se puede especificar protocolo con el que se pueda acceder, por ejemplo ftp://.

```
<a href="http://www.elpais.com">ir a elpais.com</a>
```

[ir a elpais.com](http://www.elpais.com)

1.9.- VÍNCULOS O HIPERENLACES

- **Enlaces a direcciones de correo**

- ✓ al pincharlos nos abre la aplicación de correo por defecto del sistema con un nuevo mensaje de correo electrónico dirigido a una dirección de mail determinada.
- ✓ Son muy habituales en las páginas web para ofrecer una vía de contacto con el propietario de la página.
- ✓ Utiliza el atributo **mailto:** para indicar la dirección de correo a la que se debe dirigir el enlace. Opcionalmente se puede indicar el asunto (**subject:**) , o la copia de carbón (**cc:**)

- **Enlaces a archivos**

- ✓ Basta con incluir la ruta al archivo.

```
<a href="mailto:professor.ies.serpis@gmail.com?subject=contacto a través de la pagina">
  professor.ies.serpis@gmail.com</a>
<br>
<a href="./musica/music.wav">Un archivo de música.</a>
```

professor.ies.serpis@gmail.com
[Un archivo de música.](#)

1.10.-IMÁGENES

- ✓ Para insertar una imagen se utiliza la etiqueta **** (image). No utiliza cierre y en ella hemos de especificar obligatoriamente la ruta (local o en Internet) con el atributo **src** (source).
- ✓ Otros atributos opcionales son:
 - ✓ **alt**: es una descripción de la imagen. El navegador mostrara esta descripción, cuando la imagen no ha sido todavía cargada, o si no puede encontrarla. También las utilizan aplicaciones para discapacitados o teléfonos vocales que no muestran imágenes.
 - ✓ **height y width**: especifican la altura y anchura respectivamente de la imagen en pixeles o % que indica el porcentaje del área de trabajo del navegador que utilizará. Puede no coincidir con los de la imagen original, aunque es aconsejable para evitar pixelados o transferencia innecesaria de imágenes muy grandes.
 - ✓ **title**: añade un titulo que se muestra al pasar el ratón por la imagen.
- ✓ Para utilizar imágenes como enlaces se pone la imagen como contenido del enlace.

```
<div align="center">  
  <a href="https://www.w3schools.com/">  
    </a>  
</div>
```



1.10.-IMÁGENES

- Alineación de imágenes con HTML

- ✓ Para alinear una imagen horizontalmente con un párrafo, incluimos la imagen (img) en un párrafo (p).

```
<p>imagen con texto a la derecha</p>  
<p>imagen con texto a la izquierda</p>
```










imagen con texto a la derecha



imagen con texto a la izquierda

1.10.-IMÁGENES

✓ Respecto a los tipos de imágenes, aquí tenemos una comparativa.

Formato de imagen	Categoría	Paleta de colores	Transparencia	Animación	Uso
 JPG	Lossy	Millones de colores	✗	✗	Imágenes sin animación Fotografía
 GIF	Lossless	Máximo 256 colores	Binario	✓	Animaciones simples Gráficos con colores planos Gráficos sin degradados
 PNG ₈	Lossless	Máximo 256 colores	✓	✗	Uso similar al formato .GIF Mejor transparencia pero sin animación Es perfecto para iconos
 PNG ₂₄	Lossless	Millones de colores	✓	✗	Uso similar al formato .PNG-8 Imágenes sin animación y con transparencia
 BMP	Lossless	Millones de colores	✗	✗	Formato de imagen antiguo No hay nada en lo que despunte
 SVG	Vector/Lossless	Millones de colores	✓	✗	Gráficos y logotipos para páginas web Retina / pantallas de alta resolución
 WEBP	Lossy/Lossless	Millones de colores	✓	✓	Gráficos y logotipos para páginas web Retina / pantallas de alta resolución

A decorative vertical bar on the left side of the slide, composed of numerous overlapping circles of various sizes and colors, including blue, green, yellow, orange, and pink, creating a vibrant, bubbly effect.

Práctica

- Realiza la Práctica 2.

1.11.- TABLAS

- ✓ Una tabla es un conjunto de celdas organizadas dentro de las cuales podemos alojar distintos contenidos. Está delimitada por la etiqueta **<table> ... </table>**.
- ✓ Nos puede ayudar a generar texto en columnas como los periódicos, prefijar los tamaños ocupados por distintas secciones de la página o poner de una manera sencilla un pie de foto a una imagen.
- ✓ Las tablas son descritas por filas de izquierda a derecha definidas por la etiqueta **<tr> ... </tr>** (table row)
- ✓ Dentro de cada fila, habrá diferentes columnas definidas por la etiqueta **<td> ... </td>** (table data cell) donde colocaremos su contenido. Aunque no muy utilizado, podemos utilizar **<th> ...</th>** para darle carácter de cabecera en negrita y centrado.

```
<table border="1px">
  <tr>
    <th>Cabecera Celda 1</th>
    <th>Cabecera Celda 2 </th>
  </tr>
  <tr>
    <td>Fila 1, Celda 1</td>
    <td>Fila 1, Celda 2 </td>
  </tr>
  <tr>
    <td>Fila 2, Celda 1</td>
    <td>Fila 2, Celda 2</td>
  </tr>
</table>
```

Cabecera Celda 1	Cabecera Celda 2
Fila 1, Celda 1	Fila 1, Celda 2
Fila 2, Celda 1	Fila 2, Celda 2

Nota: a pesar de que es un estilo, y ya que queremos ver la distribución de celdas, utilizamos el atributo border en el html, algo que no es normativo en HTML5, ya que debería estar en el CSS.

1.11.- TABLAS

- ✓ Los siguientes atributos permiten combinar o expandir celdas de la misma forma que se hace en las tablas de un documento de Word o Excel:
 - ✓ **colspan** Expande una celda horizontalmente. (Efecto “combinar celdas”)
 - ✓ **rowspan** Expande una celda verticalmente.

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

  <tr>
    <td colspan="2">Esta celda tiene colspan 2</td>
  </tr>
  <tr>
    <td rowspan="2">Esta celda tiene row 2</td>
    <td>Fila 2, Celda 2</td>
  </tr>
  <tr>
    <td>Fila 3, Celda 2</td>
    <td>Fila 3, Celda 2</td>
  </tr>
</table>
```

Esta celda tiene colspan 2		
Esta celda tiene row 2	Fila 2, Celda 2	
	Fila 3, Celda 2	Fila 3, Celda 2

Nota: aunque las tablas permiten ordenar los contenidos de una página web, no deben ser utilizadas como herramientas de diseño.

1.11.- TABLAS

- Anidamiento de tablas.

✓ Al igual que con las listas, es posible anidar tablas.

```
<table border="1">
  <tr>
    <td>
      Celda de la tabla principal
    </td>
    <td>
      <table border="1">
        <tr>
          <td>Tabla anidada, celda 1</td>
          <td>Tabla anidada, celda 2</td>
        </tr>
        <tr>
          <td>Tabla anidada, celda 3</td>
          <td>Tabla anidada, celda 4</td>
        </tr>
      </table>
    </td>
  </tr>
</table>
```

Celda de la tabla principal	Tabla anidada, celda 1	Tabla anidada, celda 2
	Tabla anidada, celda 3	Tabla anidada, celda 4

A decorative vertical bar on the left side of the slide, composed of numerous overlapping circles of various sizes and colors, including blue, green, yellow, orange, and pink, creating a vibrant, bubbly effect.

Práctica

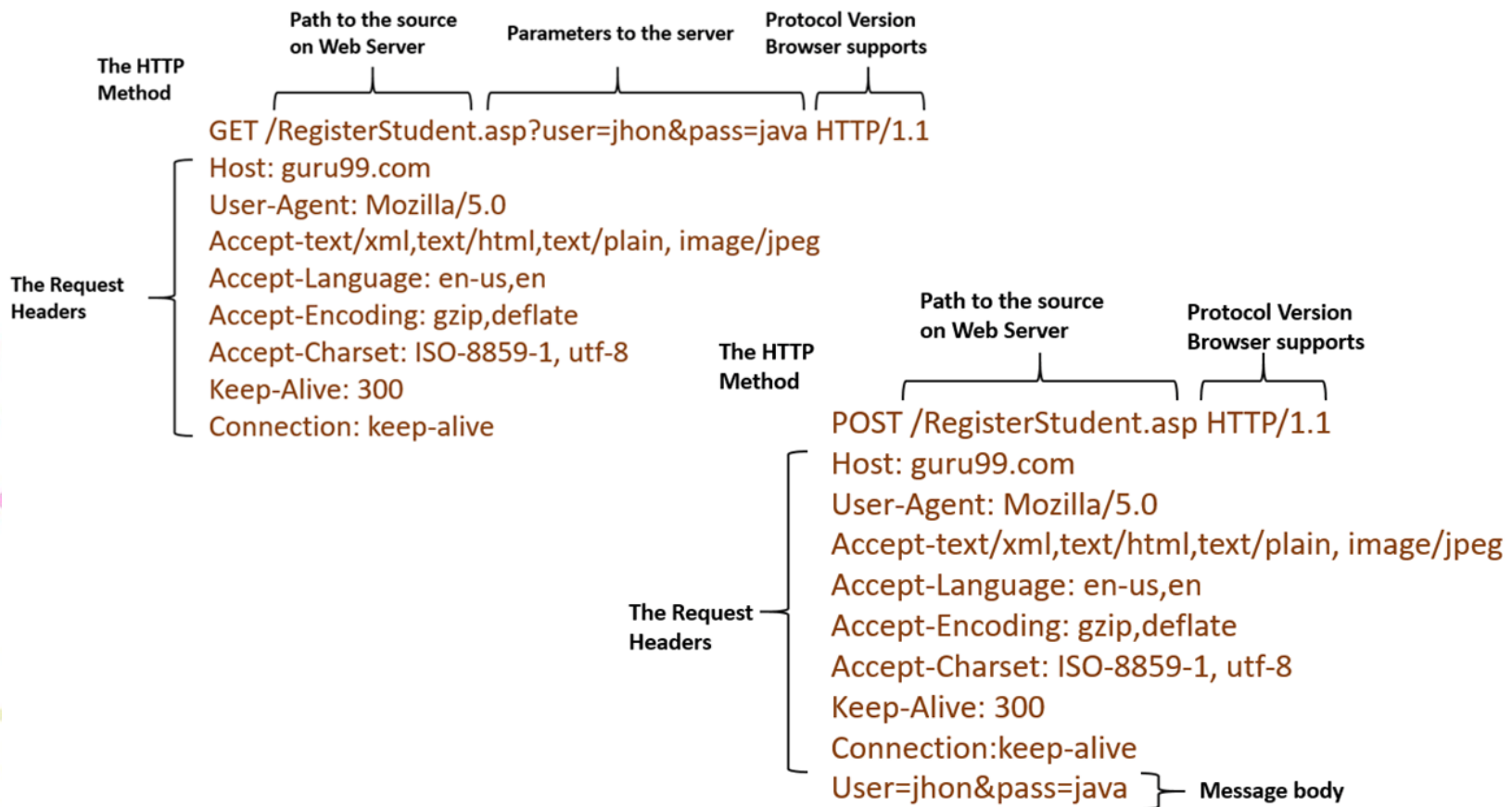
- Realiza la Práctica 3.

1.12.- FORMULARIOS

- ✓ Su utilizan para que el usuario envíe una información precisa, rellenando ciertos campos. Los valores de esos campos serán enviados a una dirección de correo, o como parámetros a una web, generalmente PHP o ASP, con la que el servidor web es capaz de procesar los datos.
- ✓ Los formularios son definidos con las etiquetas **<form> ... </form>**. En ella deberemos especificar algunos atributos:
 - ✓ **action**: define el tipo de acción a llevar a cabo con el formulario. Puede ser un correo o una página.
 - ✓ **method**: especifica la forma en la que el formulario es enviado, mediante dos posibles valores: **post** o **get**.
 - ✓ **enctype**: especifica la forma en que los datos son codificados al enviarlos al servidor. Puede tener los valores:
 - ✓ **application/x-www-form-urlencoded**: Valor por defecto. Todos los caracteres se codifican antes de enviarse (los espacios se convierten en símbolos "+" y los caracteres especiales se convierten en valores ASCII HEX)
 - ✓ **multipart/form-data**: cuando se sube un archivo con el formulario.
 - ✓ **text/plain**: Sin codificación. No recomendado si es correo.

1.12.- FORMULARIOS

- Diferencias entre una solicitud GET y POST
- ✓ Con GET los parámetros del formulario se mandan en la URL de la petición. Con POST se mandan en el cuerpo de la petición.





1.12.- FORMULARIOS

- Diferencias entre una solicitud GET y POST

GET	POST
Los valores son visibles en la URL.	Los valores no son visibles en la URL.
Tiene una limitación en la longitud de los valores, generalmente 255 caracteres.	No tiene ninguna limitación en la longitud de los valores.
Los resultados de GET son mejores en comparación con POST debido a la simple naturaleza de agregar los valores en la URL.	Tiene un rendimiento más bajo en comparación con el método GET debido al tiempo dedicado a incluir valores POST en el cuerpo HTTP.
Este método solo admite tipos de datos de cadena.	Este método admite diferentes tipos de datos, como cadenas, numéricos, binarios, etc.
Los resultados de GET se pueden marcar como favoritos.	Los resultados de POST no se pueden marcar como favoritos.
La solicitud GET a menudo se puede almacenar en caché.	La solicitud POST apenas se puede almacenar en caché.
Los parámetros GET permanecen en el historial del navegador web	Los parámetros no se guardan en el historial del navegador web.

1.12.- FORMULARIOS

- **Elementos de Formularios.**
- ✓ Cualquier elemento de formulario tendrá la etiqueta abierta **<input>** en ella se debe especificar el atributo **name**, que identifica el campo del formulario, **type**, el tipo de campo, aparte de otros opcionales como los siguientes:
 - ✓ **value**: especifica el valor por defecto para un tipo input. Es el valor que será mandado junto con el nombre al servidor.
 - ✓ **size**: tamaño de la entrada en número de caracteres.
 - ✓ **maxlength**: número máximo de caracteres.
 - ✓ **id**: permite emplear un identificador. Suele hacerse coincidir con el nombre y se utiliza para asociarlo a un <label>.
- ✓ Los elementos del formulario se pueden agrupar en campos **<fieldset> ... </fieldset>**. Suele ser utilizado para agrupar campos que son relacionados de alguna manera. Dentro de este, con la etiqueta **<legend> ... </legend>**, damos una leyenda o nombre a la agrupación.
- ✓ Las entradas de formulario suelen etiquetarse con la etiqueta **<label> ... </label>**. El atributo **for** permite asociar la etiqueta con el campo si tiene el mismo valor que el atributo **id** del input.

1.12.- FORMULARIOS

- Elementos de Formularios.

- ✓ Con type podremos utilizar los diferentes tipos de campos:

- ✓ **text**: un campo de texto.

Nombre:

- ✓ **password**: texto oculto en el que se muestra "*" al rellenarlo. Se suele utilizar con contraseñas.

Contraseña:

- ✓ **radio**: permite una selección con botones de radio. Sólo se podrá seleccionar una opción. Para que varios elementos pertenezcan a la misma selección, deberán tener el mismo nombre (**name**). Una de la opciones puede utilizar el atributo **checked** sin parámetros (opción por defecto).

Estado civil:

☒ soltero

☐ casado

- ✓ **checkbox**: permite seleccionar o no un elemento.). Permite utilizar el atributo **checked** sin parámetros (opción por defecto). La información que nos llega será nombre_elemento=on o =off.

☒ Quiero recibir información.

1.12.- FORMULARIOS

- Elementos de Formularios.

- ✓ Existen otros elementos de formulario que no son `<input>`:
 - ✓ `<textarea>...</textarea>`: espacio de texto de varias líneas. Opcionalmente podemos utilizar los atributos **rows** (número de filas), **cols** (número de columnas), o el valor por defecto que queda entre las etiquetas.

Comentarios sobre su satisfacción personal:

Escriba tu comentario...

- ✓ `<select> ... </select>`: menú desplegable que nos permiten elegir una (o varias) de las múltiples opciones. Opcionalmente podremos utilizar los atributos **size** para indicar el número de valores mostrados de la lista (el resto pueden ser vistos por medio de la barra lateral de desplazamiento), o **multiple** (sin parámetros) que permitirá seleccionar varios elementos con las teclas ctrl o shift. . Cada opción será incluida en una línea precedida de la etiqueta `<option>`. ara que sea la Una de ellas puede utilizar el atributo **selected** (sin parámetros) popción por defecto. Además podemos elegir el valor con **value**, de forma que se el dato enviado no el texto de `<option>`

Estado civil: /

casado ▾

casado

soltero

1.12.- FORMULARIOS

- Elementos de Formularios.

- ✓ Todos los elementos del formulario pueden utilizar el atributo **disabled**, si lo hacen el elemento estará deshabilitado y no se podrá utilizar o pinchar.

```
<form action="">  
  <label>Nombre:</label><input type="text" value="Mickey"><br>  
  <label>Apellido:</label><input disabled type="text" value="Mouse"><br>  
</form>
```

Nombre:

Apellido:

- ✓ Además existen otros elementos que permiten el envío o borrado del formulario. Se utilizan como un tipo `<input>` con type:
 - ✓ **submit**: botón que permite enviar el formulario y hacerlo llegar a su destino. Es obligatorio para que el formulario pueda ser enviado. El atributo **value** dicta el texto que aparece el botón.
 - ✓ **reset**: botón que permite el borrado del formulario. Es opcional.

1.12.- FORMULARIOS

- Ejemplo de formulario.

```
<form action="login.php" method="get">
  <fieldset>
    <legend>Información personal.</legend>
    <label for="user">Nombre: </label>
    <input type="text" id="user" name="user" value="su nombre"><br>
    <label for="mail">Correo: </label>
    <input type="text" id="mail" name="mail" value="ejemplo@ejemplo.com">
    <br>
    Contraseña: <input type="password" name="pass"><br>
  </fieldset>
  <fieldset>
    <legend>Información laboral.</legend>
    Experiencia:<br>
    <select multiple name="tec">
      <option selected value="1">HTML</option>
      <option value="2">CSS</option>
      <option value="3">JavaScript</option>
    </select><br>
    Lenguaje de programación preferido:<br>
    <input type="radio" name="lenguaje" value="1">C++<br>
    <input type="radio" name="lenguaje" value="2">Java<br>
    <input type="radio" name="lenguaje" value="3">Phyton<br>

    Explique su experiencia como desarrollador:<br>
    <textarea name="comentario" rows="10" cols="40">
      Escriba tu comentario...
    </textarea><br>
    <input type="checkbox" name="newsletter" checked>
      Quiero recibir información.<br>
  </fieldset>
  <input type="submit" value="Envio">
  <input type="reset" value="Borrar">
</form>
```

Información personal.

Nombre:

Correo:

Contraseña:

Información laboral.

Experiencia:

☒ HTML
☐ CSS
☐ JavaScript

Lenguaje de programación preferido:

☐ C++
☐ Java
☐ Phyton

Explique su experiencia como desarrollador:

☒ Quiero recibir información.



1.12.- FORMULARIOS

- **Ejemplo de formulario.**
- En el caso de utilizar el método get, al rellenar el formulario y pinchar en Envío, los datos se envían en la propia URL.
- `/login.php?user=Bill+Gates&mail=bill.gates%40ieserraperenxisa.com&pass=windows&tec=1&tec=2&tec=3&lenguaje=1&comentario=Toda+la+vida+creando+windows...&newsletter=on`
- En el caso de utilizar el método post, la URL sería login.php, y los datos irían en el cuerpo (body) del mensaje HTML.

A decorative vertical bar on the left side of the slide, composed of numerous overlapping circles of various sizes and colors, including blue, green, yellow, orange, and pink, creating a vibrant, bubbly effect.

Práctica

- Realiza la Práctica 4.

1.13.- FRAMES

- ✓ Los **frames** (marcos) permiten tener en una misma página espacios independientes los unos de los otros, de modo que en cada espacio se coloca una página distinta que se codifica en un fichero HTML distinto.
- ✓ Fueron incluidas en el estándar HTML 4, y sustituidos por **<iframe>** en HTML5, por lo que los veremos ligeramente
- ✓ Al ser cada marco es independiente, tendrán sus propias barras de desplazamiento, horizontales y verticales, por separado.
- ✓ Para especificar la disposición de los marcos en la ventana se utiliza la etiqueta **<frameset> ... </frameset>**. Esta sustituye a **<body>...</body>**. En ella podemos utilizar el atributo **cols**, o **rows** (no a la vez) para definir cómo se dividirá la página en filas o columnas, bien en píxeles o porcentaje. Además se puede utilizar el comodín “*” para que se rellene el resto del espacio.
- ✓ La etiqueta **<frame>** permite especificar el código HTML fuente del frame con su atributo **src**.

1.13.- FRAMES

```
<frameset rows="15%,*" border="4" bordercolor="#00FFFF">
  <frame noresize src="1.13 Frames1.html">
  <frameset cols="20%,*">
    <frame scrolling="no" src="1.13 Frames2.html">
    <frame src="1.13 Frames31.html" name="principal">
  </frameset>
</frameset>
```

Esta es la página 1

**Esta es la
página 2**

[pagina 3.1](#)
[pagina 3.2](#)
[pagina 3.3](#)

Esta es la página 3

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

`<p>Esta es la página 2</p>`

```
<a href="1.13 Frames31.html" target="principal">pagina 3.1</a> <br>
<a href="1.13 Frames32.html" target="principal">pagina 3.2</a> <br>
<a href="1.13 Frames33.html" target="principal">pagina 3.3</a>
```