

Clase 2

¿Qué es HTML?

DOCTYPE y estructura básica

Etiquetas

- De bloque

- De línea

- De apertura y cierre

- De cierre

Atributos

Lenguajes para Desarrollo Web

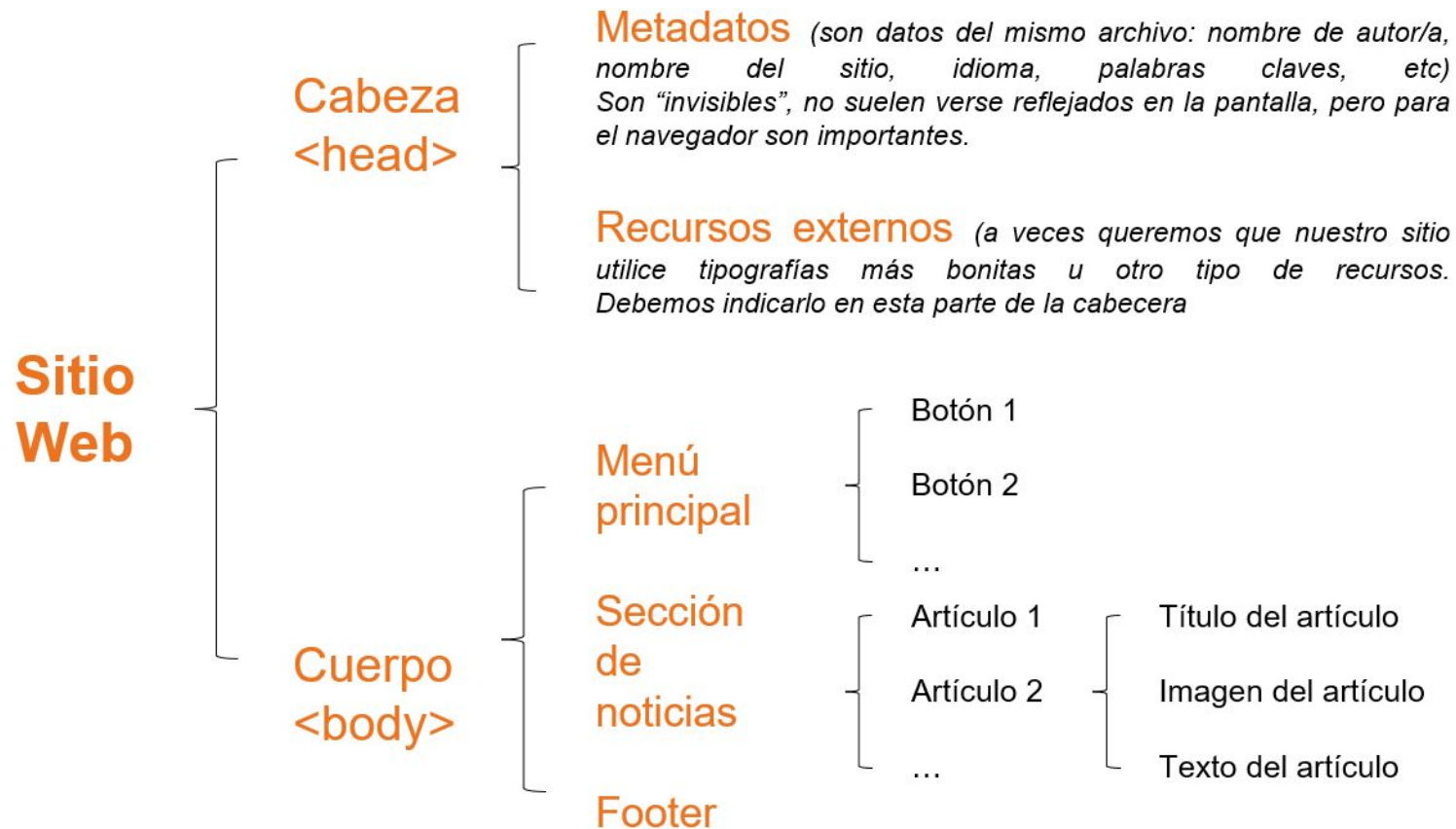
Los lenguajes de programación son un conjunto de elementos, códigos y reglas que permiten crear una serie de instrucciones para comunicar a los sistemas informáticos las acciones a realizar.

Para el diseño y desarrollo web existen diferentes lenguajes que nos permiten llevar el diseño en papel (sketch) a una estructura que pueda interpretar un computador. En este curso se verán dos lenguajes base: **HTML** y **CSS**. Más adelante veremos **JavaScript**.



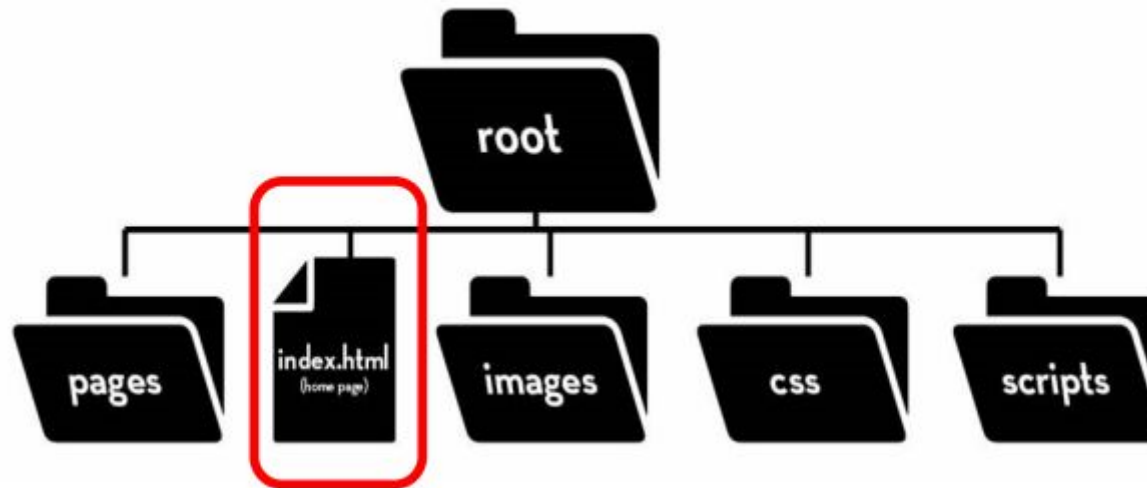
¿Qué es HTML?

Es un lenguaje de marcado que estructura los elementos que componen un sitio web (textos, imágenes, botones, etc.) de una forma jerárquica.



Nombrar y crear archivos

- ☐ Los nombres no deben tener espacios, acentos, ñ, ni símbolos.
- ☐ Si son varias palabras usar guiones " - " o " _ " ej: "mi -pagina-web"
- ☐ Tienen que estar escritos en minúsculas.
- ☐ Los archivos de página deben tener la extensión ".html" (es la forma en que el servidor web sabe que se trata un documento web).
- ☐ Al guardar archivos en Windows, verificar que no se guarden con la doble extensión de archivos (archivo.html.txt)



Etiquetas HTML

Están delimitadas por un inicio y un final de cada elemento. Lo que se encuentra dentro de la etiqueta (el contenido) es a lo que se le da formato.

Todas las etiquetas aceptan atributos: cualquier característica que pueda ser diferente entre una etiqueta y la otra.

Cada etiqueta puede tener más de un atributo, separados entre sí por espacios.

Los atributos sólo se escriben en la etiqueta de apertura y su valor se escribe entre comillas.

```
<etiqueta atributo="valor">  
Contenido  
</etiqueta>
```

Las etiquetas cerradas tienen un contenido, por lo general texto, mientras que las abiertas se utilizan para incluir elementos como imágenes, líneas, entre otros.

Listado de Etiquetas

Etiquetas de Estructura Básica

`<!DOCTYPE html>`: determina qué versión de HTML utiliza el documento, información clave para los navegadores. La interpretación y representación será más rápida y consistente.

`<html>`: etiqueta inicial, define que el documento está bajo el estándar de HTML. Tiene apertura y cierre.

`<head>`: es la parte privada del documento, y se utiliza como espacio de comunicación entre el sitio web y el navegador. Esta etiqueta también envuelve etiquetas relacionadas con la importación de documentos CSS y JS.

`<meta>`: añade información sobre la página de la cual pueden valerse los buscadores. También puede definir el idioma y la codificación en la cual está escrita la página.

`<title>`: título de la página. Será visualizado en la solapa del navegador.

`<body>`: contiene toda la información visible de la página.

Etiquetas de Estructura Básica

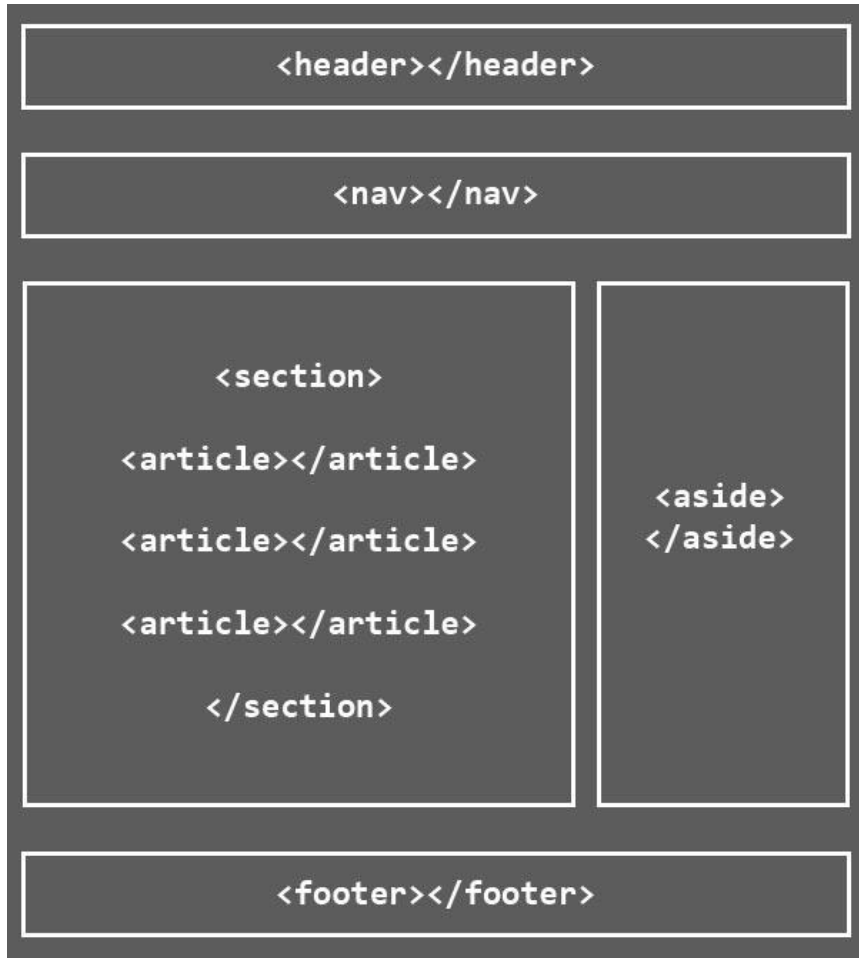
```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
  </head>
  <body>

  </body>
</html>
```

Las etiquetas del **<body>** se dividen en dos grupos:

- **Elementos de bloque:** son aquellos que, sin ser modificados por CSS, ocupan el 100% del ancho del contenedor, y se mostrarán uno abajo del otro.
- **Elementos de línea:** sólo ocupan el ancho que diga el contenido, y se verán uno al lado del otro.

HTML5 - Etiquetas semánticas



Generan estructura y definen su contenido.

`<section>`: Define una sección en un documento.

`<nav>`: Define un bloque que contiene enlaces de navegación, como por ejemplo el menú.

`<article>`: Define contenido autónomo que podría existir independientemente del resto del contenido.

`<aside>`: Define contenidos vagamente relacionados con el resto del contenido de la página.

`<main>`: Define el contenido principal o importante en el documento. Solamente existe uno en el documento.

`<header>`: Define la cabecera de una página o sección.

`<footer>`: Define el pie de una página o sección.

HTML - Algunas otras etiquetas

`<h1></h1>` a `<h6></h6>`: son etiquetas de bloque que permiten crear una estructura para el contenido de una página web. Son esenciales para cumplir con la correcta organización de dicho contenido.

`<p></p>`: permiten agrupar texto dentro de un párrafo para que el contenido sea más fácil de leer y esté bien organizado.

`<a> `: para crear un enlace a una página web. El atributo principal de la etiqueta HTML es href, donde pondremos el enlace al que queremos conectar. Otro atributo muy usado es target, el cual nos sirve para indicar si el enlace se abrirá en una nueva ventana o en la misma.

`
`: indica al navegador que viene un salto de línea.

`<hr/>`: representa un cambio de tema entre párrafos.

Multimedia en HTML

```

```

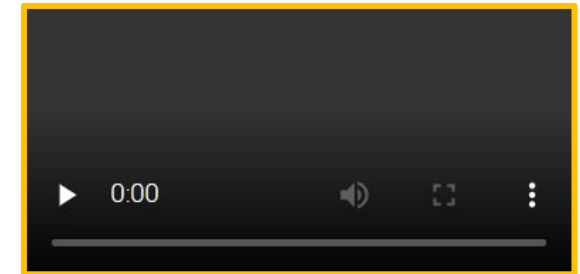
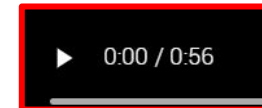
El atributo “alt” es un texto que debe representar la foto que se está visualizando. Tiene que ser conciso y breve, y dejar en claro de qué se trata la imagen.

Audio y video:

```
<video src="./video.mp4" controls></video>
<audio src="./hello.wav" controls></audio>
```

Atributos de audio y video...

- ☐ **Controls:** controles para manejar el audio.
- ☐ **Autoplay:** reproducción automática.
- ☐ **Loop:** repetición automática.
- ☐ **Preload:** almacenamiento temporal del audio.
 - ☐ **"none"**: no almacena el archivo.
 - ☐ **"auto"**: almacenamiento temporal del archivo.
 - ☐ **"metadata"**: almacena temporalmente sólo los metadatos del archivo.

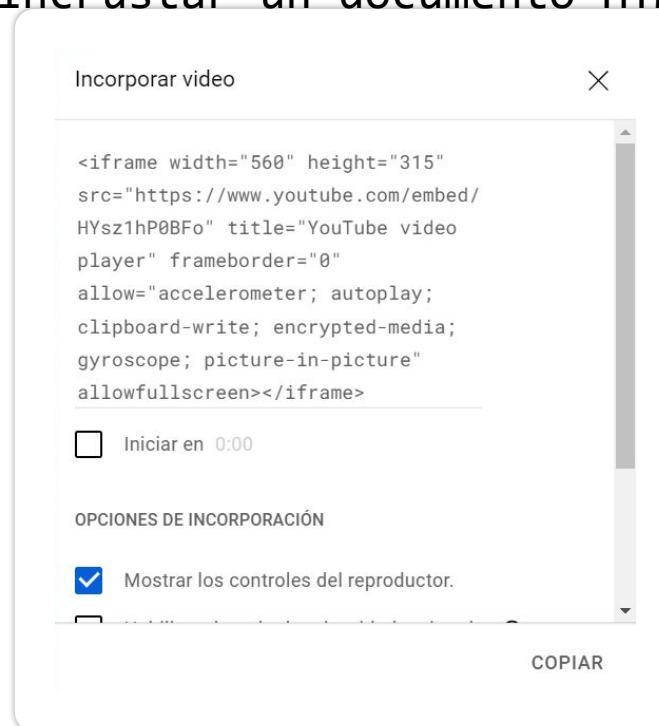


Multimedia en HTML

<source>: especificar múltiples fuentes de archivos con el fin de proporcionar vídeos/audios codificados en formatos diferentes, para diversos navegadores.

```
<source src="./hello.wav" type="audio/wav">  
<source src="./hello.mp3" type="audio/mp3">
```

<iframe>: permite insertar o incrustar un documento HTML dentro del principal.



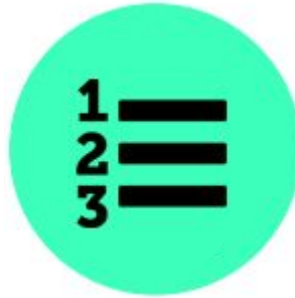
HTML - Listas



Listas de definición

Listas en las que tenemos “valor” y “definición”. Suelen utilizarse para definir términos.

```
<dl>
  <dt>Término1</dt>
  <dd>Definición 1</dd>
  <dt>Término 2</dt>
  <dd>Definición 2</dd>
  ...
  <dt>Término N</dt>
  <dd>Definición N</dd>
</dl>
```



Listas ordenadas

Cada uno de los elementos irá precedido de un número o letra que establezca su orden.

```
<ol>
  <li>Elemento 1</li>
  <li>Elemento 2</li>
  ...
  <li>Elemento N</li>
</ol>
```



Listas no ordenadas

Los elementos estarán precedidos por una viñeta.

```
<ul>
  <li>Elemento 1</li>
  <li>Elemento 2</li>
  ...
  <li>Elemento N</li>
</ul>
```

HTML - Formularios

Los formularios permiten a los usuarios la introducción de datos, que generalmente se envían a un servidor web o a un archivo para su procesamiento y almacenamiento.

`<form></form>`: elemento contenedor que define formalmente un formulario. Todos sus atributos son opcionales, pero es una práctica estándar establecer siempre al menos los siguientes atributos:

- **“action”**: define la ubicación (URL) donde se envían los datos que el formulario ha recopilado cuando se validan.
- **“method”**: define con qué método HTTP (o a dónde) se envían los datos. Puede ser:
 - **GET**: la información viajará por la barra de direcciones a continuación del nombre del archivo. Si no se indica en “method”, será GET por defecto.
 - **POST**: la información viajará junto a los encabezados del HTML (será “invisible”). No se obtienen los datos adjuntos en la dirección URL, sino que son incluidos en el cuerpo de la petición.

Ejemplo de formulario

```
<form action="" method="">
  <label for="nombre">Nombre: </label>
  <input type="text" name="nombre" id="nombre"/>
  <br>

  <label for="area">Texto largo: </label>
  <textarea name="area" id="area" cols="30" rows="10"></textarea>
  <br>

  <label for="telefono">Teléfono: </label>
  <input type="number" name="telefono" id="telefono"/>
  <br>

  <label for="email">Correo Electrónico: </label>
  <input type="email" name="email" id="email"/>
  <br>

  <label for="comentarios">Comentarios: </label>
  <input type="text" name="comentarios" id="comentarios"/>
  <br>

  <input type="submit"/>
  <input type="reset" value="Borrar Campos"/>
</form>
```

Nombre:

Texto largo:

Teléfono:

Correo Electrónico:

Comentarios:

Formularios - input, textarea, label

Algunos controles generales para el ingreso de texto:

- `<input type="text">`: Cajas de texto de una sola línea (no acepta el uso de la tecla Enter).
- `<input type="password">`: Cajas para el ingreso de contraseñas (el contenido no será visible).
- `<textarea>`: Cajas para contenido multilínea. Puede ser una o muchas líneas de texto.

Más información sobre atributos en `<input>`

`<label>`: Define formalmente a cada elemento de un formulario. Esta etiqueta es de mucha ayuda para generar un formulario accesible.

Su principal atributo es "for", que debe ser igual al valor de "id" o "name".

```
<form>
|   <label for="nombre_apellido">Nombre: </label>
|   <input type="text" name="nombre_apellido"/>
</form>
```


Formularios - Botones

Los botones disparan las acciones del formulario. Hay 3 tipos:

- “submit”: envía los datos al archivo indicado como ***Action***.
- “reset”: vacía todo lo ingresado y resetea los campos.
- “button”: lo utilizaremos con JavaScript más adelante.

```
<form>
  <input type="submit" value="Enviar formulario"/>
  <input type="reset" value="Limpiar formulario"/>
  <input type="button" value="Sin acciones"/>
</form>
```


Formularios – Controles de Selección

En estos casos, el usuario no puede ingresar libremente un texto, sino que el programador le da una lista predefinida. El dato que llega al elegir una opción se define desde el atributo “value”.

□ Botones de radio: sólo se puede elegir una opción.

```
<div>Opción 1</div>
<input type="radio" name="pregunta 1" value="opciónuno"/>
<div>Opción 2</div>
<input type="radio" name="pregunta 1" value="opcióndos"/>
```

Opción 1
☐
Opción 2
☐

□ Casillas de chequeo: de toda la lista de opciones, el usuario puede optar por una, todas o ninguna opción.

```
<div>Acepta términos y condiciones</div>
<input type="checkbox" name="acepta" value="1"/>
```

Acepta términos y condiciones
☐

□ Menú desplegable: sólo es posible seleccionar una opción.

```
<select name="talles">
<option value="L">Large</option>
<option value="M">Medium</option>
<option value="S">Small</option>
</select>
```

Formularios - Fieldset y Legend

`<fieldset>` y `<legend>` se utilizan en conjunto y sirven respectivamente para definir y etiquetar grupos lógicos de elementos de formularios.

Al formar varios grupos de elementos se puede crear una estructura más fácil de asimilar por el usuario, especialmente para formularios con muchos elementos.

`<fieldset>`: agrupa los elementos. Se utiliza con su respectiva etiqueta de cierre y lo que hace es crear un recuadro que rodea a los elementos de formulario colocados dentro de ella.

`<legend>` sirve para nombrar o etiquetar un grupo creado con FIELDSET. Añade una nota de aclaración sobre qué tipo de información se está agrupando en el recuadro. Puede ayudar al usuario y mejorar la interfaz y la claridad de los formularios.

Formularios - Fieldset y Legend

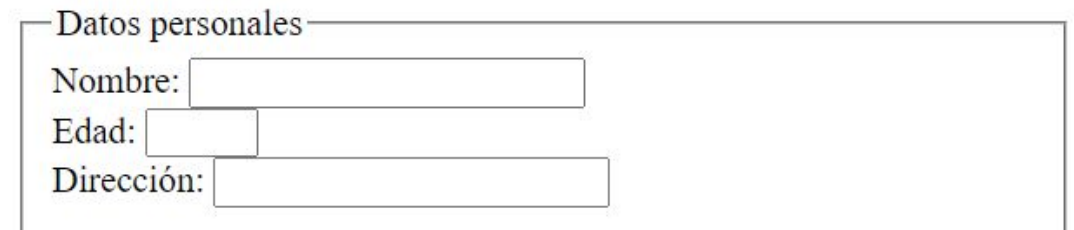
Solo `<fieldset>`:

```
<form action="">
  <fieldset>
    <!--Elemento de formulario-->
    <input type="text" name="elemento1">
    <br>
    <!--Otro elemento-->
    <input type="text" name="otro">
  </fieldset>
</form>
```

A visual representation of a form fieldset. It consists of a large rectangular box with a thin border. Inside this box, at the top-left corner, are two smaller rectangular text input fields stacked vertically. The first input field is slightly wider than the second one.

`<fieldset>` con `<legend>`:

```
<form>
  <fieldset>
    <legend>Datos personales</legend>
    Nombre: <input type="text" name="nombre">
    <br>
    Edad: <input type="text" name="edad" size="2">
    <br>
    Dirección: <input type="text" name="direccion">
  </fieldset>
</form>
```

A visual representation of a form fieldset with a legend. The fieldset is a large rectangle with a thin border. At the top-left, the text "Datos personales" is followed by a horizontal line. Below this, there are three labeled text input fields: "Nombre:" followed by a wide input field, "Edad:" followed by a narrow input field, and "Dirección:" followed by a wide input field. Each label is positioned to the left of its corresponding input field.

Enlaces absolutos y relativos

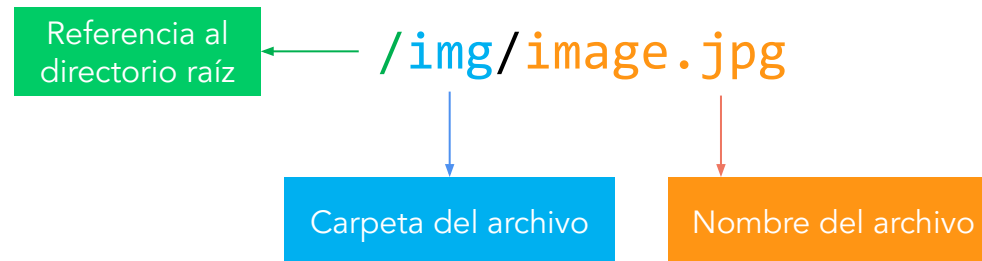
Los enlaces están formados por una serie de rutas, donde se le indica al navegador la dirección a la que tiene que ir cuando pulsamos sobre ese link.

Tenemos dos tipos de ruta:

Ruta absoluta: incluye el nombre del dominio.



Ruta relativa: indica el nombre del directorio.



Enlaces internos

Los enlaces internos permiten referenciar secciones de una misma página. Utilizaremos la etiqueta `<a>`, junto con el atributo `“href”`.

Antes en el valor de `“href”` poníamos un enlace web o de otro archivo. Para indicarle al navegador que se trata de un lugar dentro de la misma página utilizaremos los marcadores, que inician con un `#`.

El nombre que tendrán estos marcadores dependerá del `id` del elemento o sección a la que haré referencia.

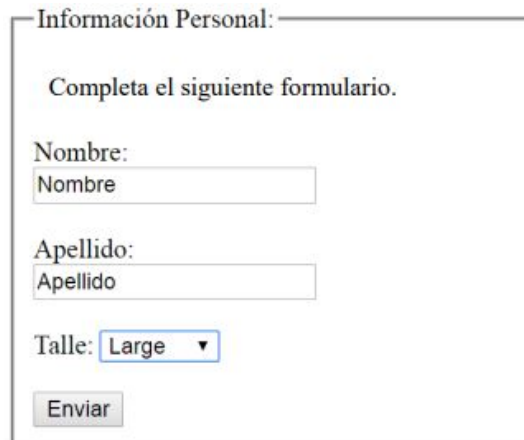
La sintaxis quedará así:

```
<a href="#pie">Ir al pie de página</a>
...
<footer id="pie"></footer>
```

¡A practicar!

Ejercicio 1: Crear una página que contenga las etiquetas de estructura básica, una imagen y un iframe de YouTube.

Ejercicio 2: Crear un formulario de contacto como indica la imagen a continuación, usando los códigos vistos.



Información Personal:

Completa el siguiente formulario.

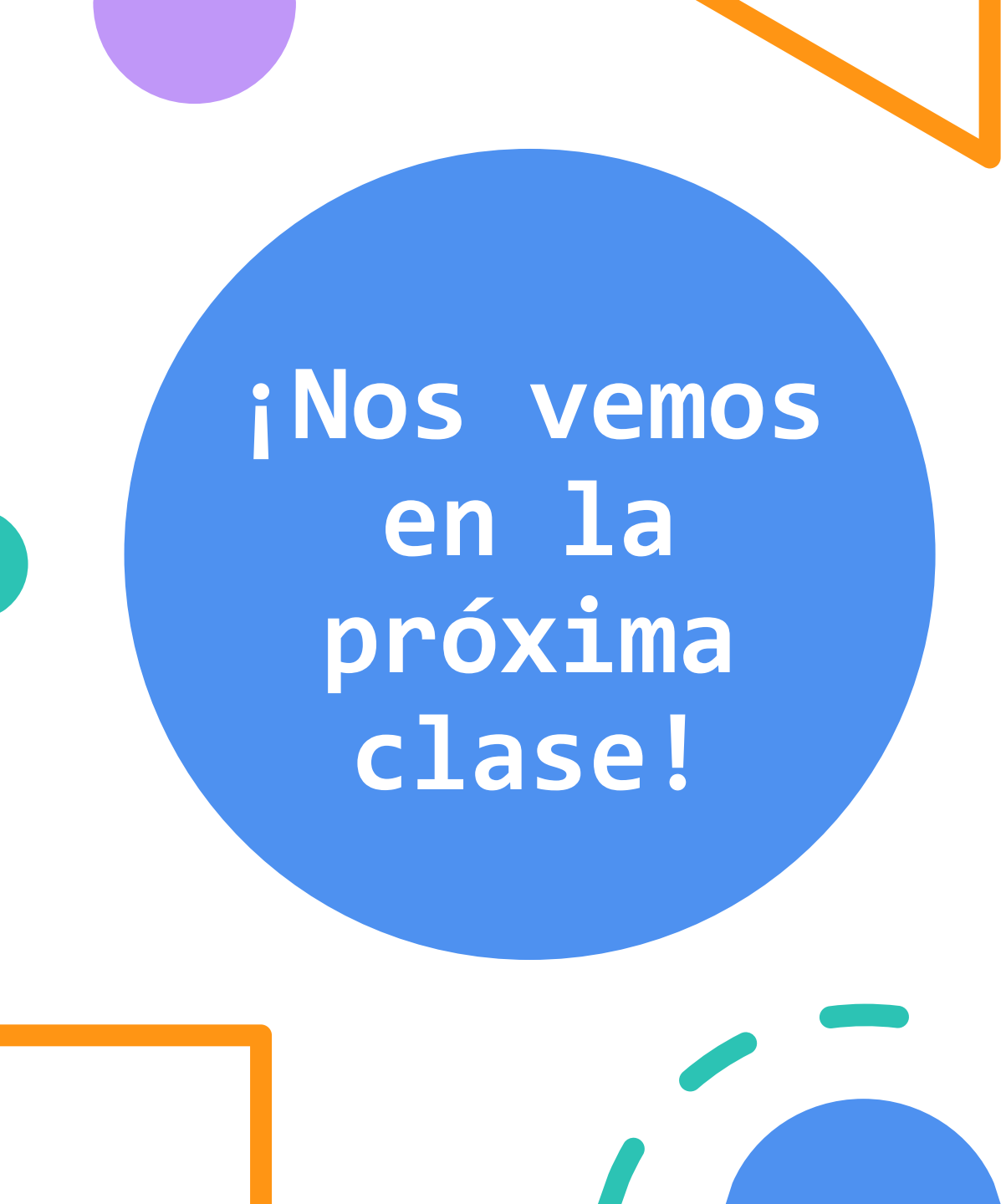
Nombre:

Apellido:

Talle:

Por el momento no nos preocuparemos por el estilo.

¡No se olviden de implementar lo aprendido en la clase de Git y GitHub!



**¡Nos vemos
en la
próxima
clase!**

Rocío Decurgez

rociodecurgez.bam@gmail.com

1150968244