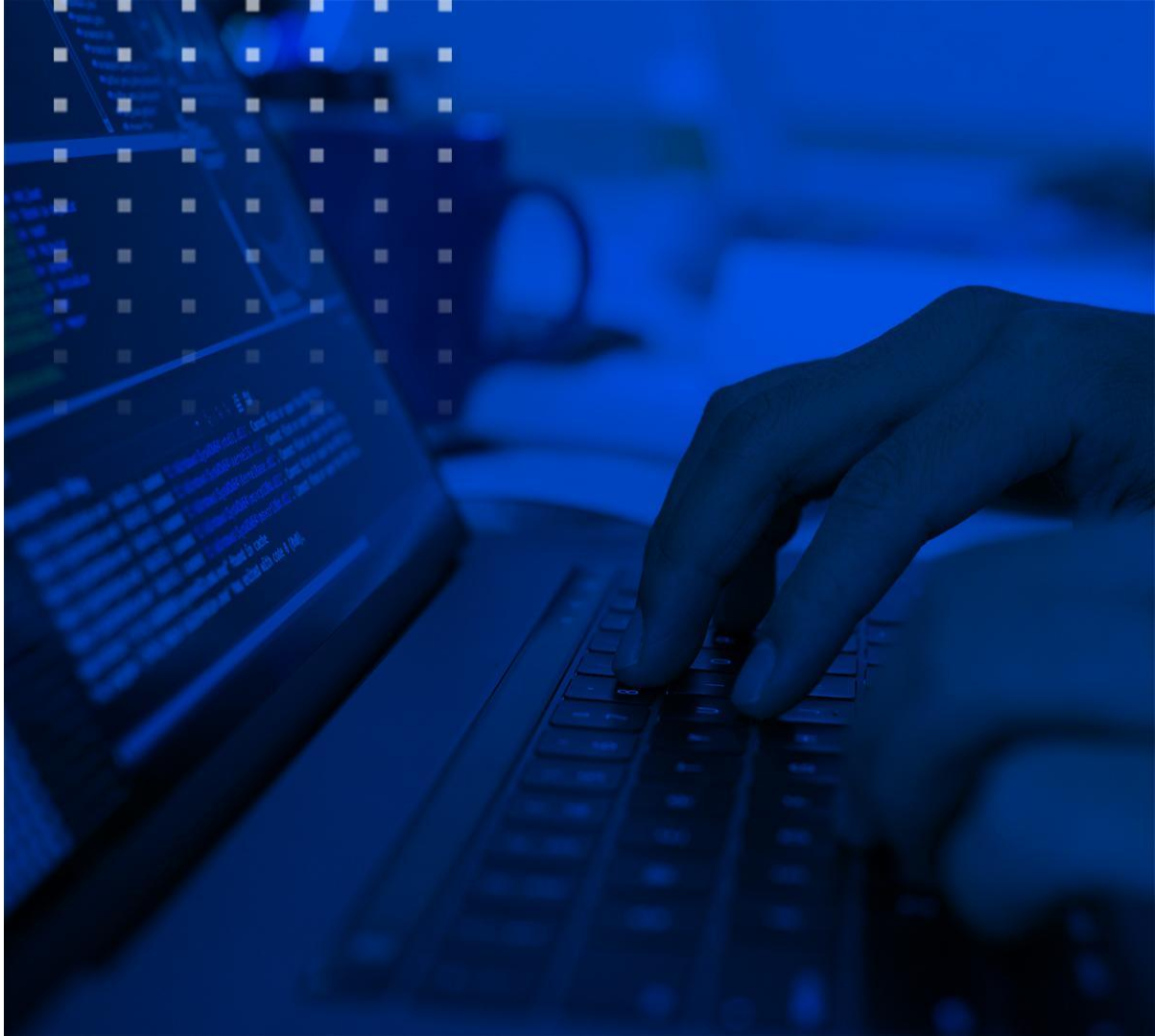
incluyeme.com

UTN.BA
UNIVERSIDAD TECNOLÓGICA NACIONAL
FACULTAD REGIONAL BUENOS AIRES

CURSO

FULL STACK DEVELOPER
NIVEL INICIAL



UNIDAD 1
Front-End: Maquetación web con HTML

Full Stack Developer Inicial

Front-End: Maquetación web con HTML

Objetivos del módulo...

- Conocer conceptos básicos del lenguaje HTML, su sintaxis y etiquetas.
- Diagramar la estructura básica de un documento HTML.

Lenguajes para el desarrollo web



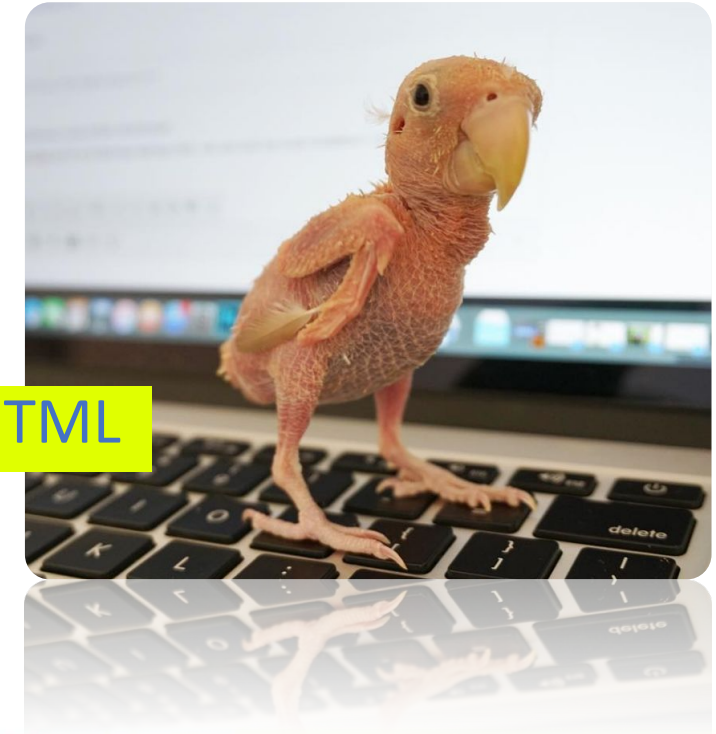
Existen diferentes lenguajes que permiten llevar el diseño en papel (sketch) a una estructura que pueda interpretar un computador.

Editores de código









HTML

Es un **lenguaje de marcado de hipertexto**, que permite crear documentos para web:

- Etiqueta
- Atributo
- Estructura
- Elemento



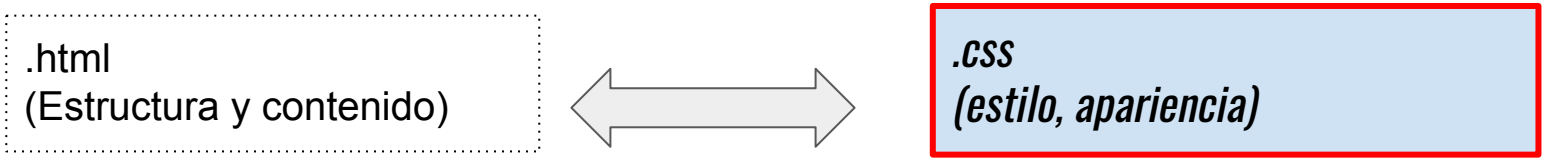
CCS

El CSS (hojas de estilo en cascada) es un **lenguaje de hojas de estilo**, que permite controlar la apariencia de una página web:

- Selectores
- Estilo
- Reglas
- Medidas

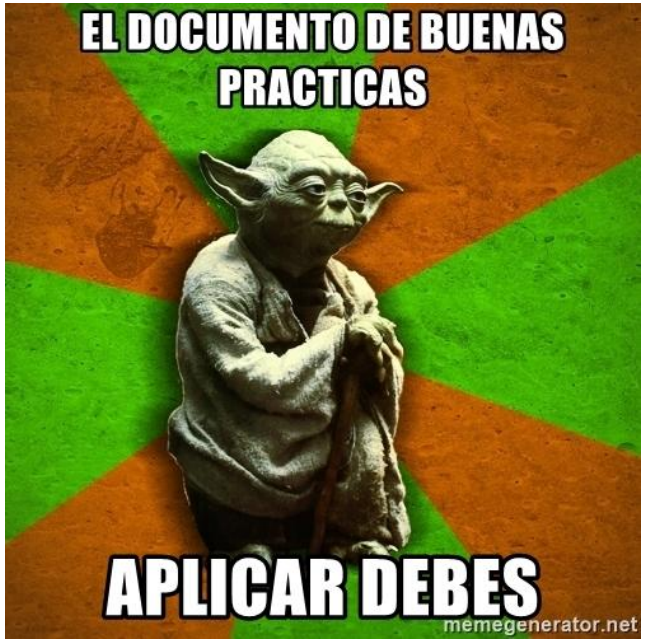
HTML Y CSS se complementan

- HTML (hyper text markup language) da estructura y contenido.
- CSS (cascading style sheets) da estilo al contenido.



Buena Práctica

- En el archivo con extensión html, solo habrá HTML.
- En el archivo con extensión css, solo habrá CSS.



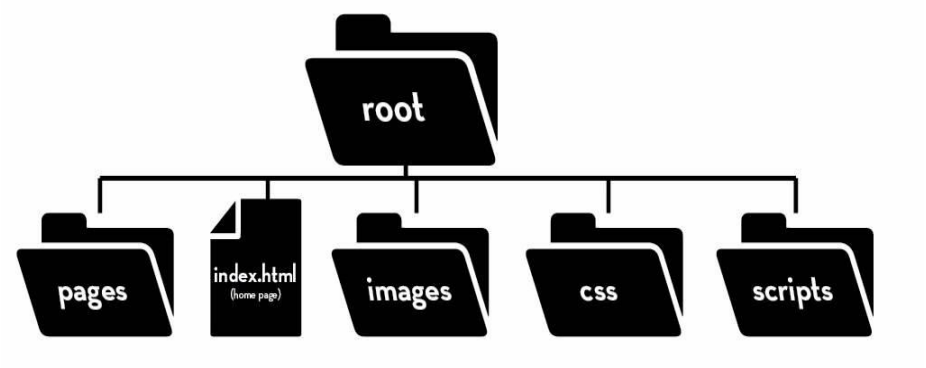
Convenciones para nombrar archivos

- No deben tener espacios, acentos, eñes, ni símbolos. Usar guiones(“-” ,“_”) ej: “mi-pagina-web”.
- Tienen que estar escritos en minúsculas.
- Deben tener la extensión “.html”.

Convenciones para nombrar archivos

Atención: al guardar archivos en Windows, verificar que no se guarden con la doble extensión de archivos (**archivo.html.txt**).

Documento predeterminado



Es el archivo que se carga cuando todavía no has hecho **click** en ningún link.

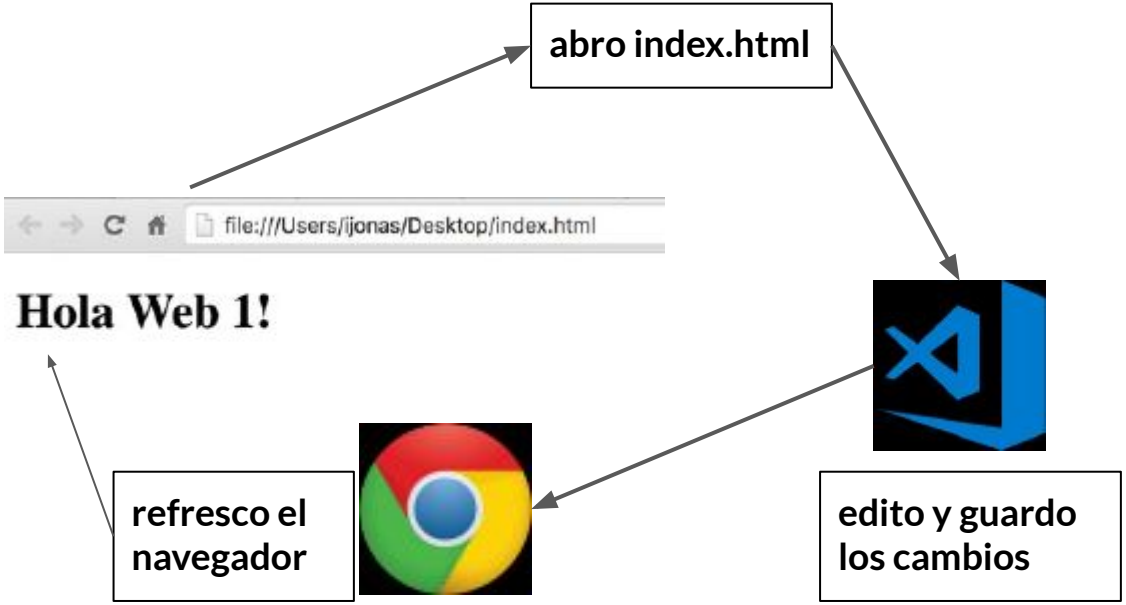
Debe llamarse **index.html**, ya que es el nombre estandarizado.

Documento predeterminado

Es el archivo que se carga cuando todavía no has hecho click en ningún link.

Debe llamarse **index.html**, ya que es el nombre estandarizado.

Ciclo de Desarrollo



Ciclo de Desarrollo

- Abro el archivo de extensión .html en VSCode
- Doble click en Chrome u otro navegador.
- Edito el archivo en VSCode.
- Guardo los cambios.
- Refresco el navegador.
- Veo los cambios.

Etiquetas HTML

Las etiquetas HTML están delimitadas por un inicio **<** y un final **>**.

Por regla general toda etiqueta es un juego de pares: una abre, otra cierra.

Etiquetas HTML

Un elemento HTML está definido por la etiqueta de apertura, un contenido y la etiqueta de cierre. Elemento es “todo desde la etiqueta de apertura a la de cierre”.

```
<etiqueta>
  Contenido
</etiqueta>
```

Atributos de las etiquetas

Todas las etiquetas aceptan atributos, esto es, cualquier característica que pueda ser diferente entre una etiqueta y la otra. El valor irá entre comillas.

Atributos de las etiquetas

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

Los mismos sólo van en la etiqueta de de apertura.

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

Etiquetas cerradas y abiertas

Las primeras encierran un contenido, por lo general texto, mientras que las abiertas sirven, entre otras cosas, para incluir elementos como imágenes, líneas, saltos de línea, etc.

Etiquetas cerradas y abiertas

El signo “/” se utiliza para las etiquetas de cierre. En estas últimas se pone a continuación del signo “<”, mientras que en las abiertas delante del signo “>”.

```
<p>Este es un párrafo con texto en su interior - Etiqueta Cerrada </p>
<br/> <!-- Esta etiqueta inserta un salto de línea en el documento -->
<hr/> <!-- Esto es una línea horizontal - Etiqueta abierta -->
```

Etiquetas abiertas en HTML5

Ya no es una obligación poner el signo “ / ”. Por ejemplo, **<img src=””>** funcionará correctamente, y lo mismo sucederá con **
, **<hr>, los meta tag.

Estructura básica

- **<head>**: es la parte privada del documento, que se utiliza como un espacio de comunicación entre el sitio web y el navegador. Envuelve **<title>**, etiquetas **<meta>** y las relacionadas con la importación de documentos CSS y JS.

Estructura básica

- **<body>**: encierra el contenido propiamente dicho del sitio.

Ambos deben estar dentro de un elemento principal: la etiqueta <html>.

Estructura básica

- **<html>:** Define que el documento está bajo el estándar de HTML. Abre y cierra, por lo tanto es fundamental no olvidar la etiqueta **</html>** al finalizar el documento.

Estructura básica

- **<title>:** Define el título de la página, el cual será visualizado en la solapa del navegador.

Estructura básica

- **<meta>:** Añade información sobre la página, desde la cual pueden valerse los buscadores, puede definir el idioma y la codificación en la cual está escrita la página.

DOCTYPE

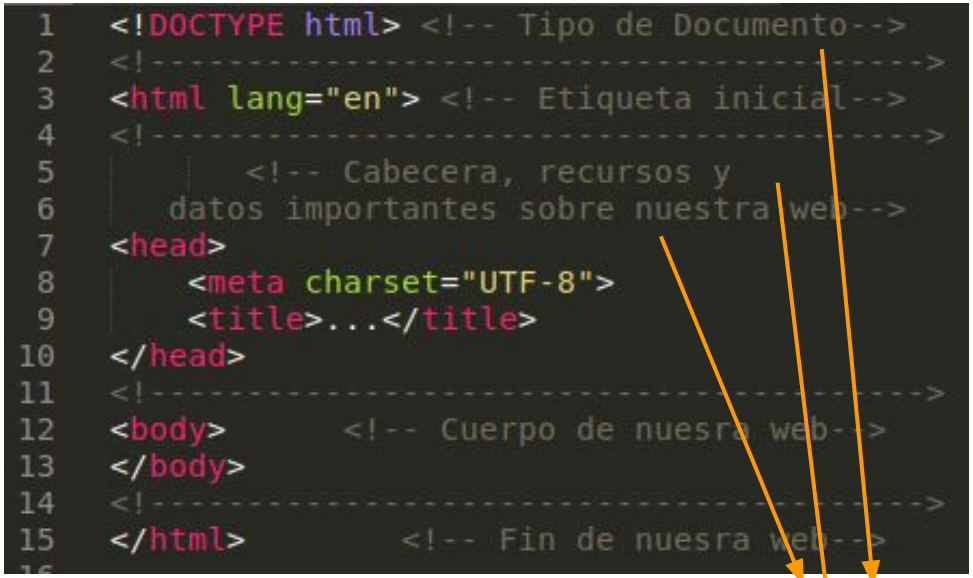
Cuando escribes tu documento **HTML**, lo primero que debes hacer es escribir el **DOCTYPE**, el cual declara el tipo de documento.

```
<!DOCTYPE html>
```

Tips: https://www.w3.org/QA/Tips/Doctype.html.es

DOCTYPE

Sirve para indicar que tu documento está escrito siguiendo la estructura determinada por una **DTD** (Document Type Definition) concreta.


<!DOCTYPE html> <!-- Tipo de Documento-->
<!-->
<html lang="en"> <!-- Etiqueta inicial-->
<!-->
<!-- Cabecera, recursos y
datos importantes sobre nuestra web-->
<head>
<meta charset="UTF-8">
<title>...</title>
</head>
<!-->
<body> <!-- Cuerpo de nuesra web-->
</body>
<!-->
</html> <!-- Fin de nuesra web-->
COMENTARIOS
- Se definen dentro de: <!-- -->
- No se muestran en el navegador.

Tipos de etiquetas: Grupo general

Todas las etiquetas que van dentro del **<body></body>** se dividen en dos grupos:

- **Elementos de bloque**
- **Elementos de línea**

```
<h1>
    Elemento de  Bloque
</h1>
<h2>
    Elemento de  Bloque
</h2>
```

<h1> a <h6> (de bloque): un encabezado es el texto que encabeza o titula el contenido que sigue. Se puede tratar de un artículo, un texto o una sección del documento que estamos viendo.

```
<h1>
    Elemento de  Bloque
</h1>
<h2>
    Elemento de  Bloque
</h2>
```

Existen 6 niveles:

del <h1></h1> al <h6></h6>

Esta jerarquía se debe respetar **en cada documento HTML** que forme parte del sitio web.

Etiquetas HTML:



-
-
 (de bloque)
- (de línea)
- (de línea)
- (de bloque)
- (de línea):

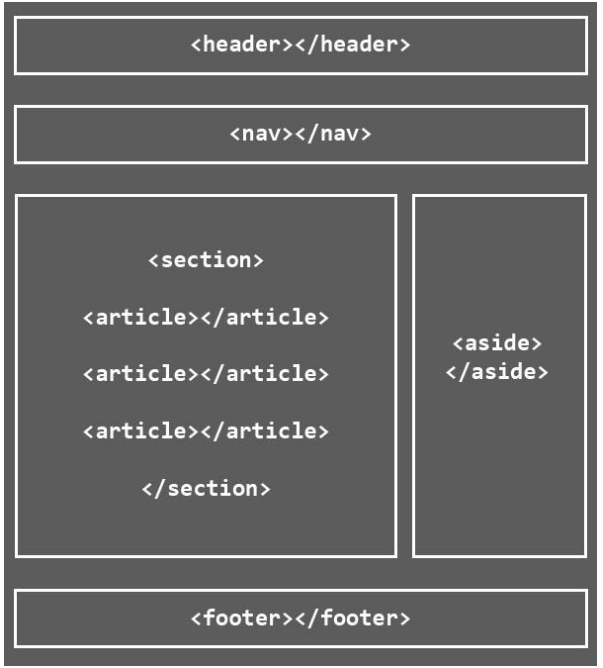
Utilizaremos todas estas etiquetas en nuestro primer ejemplo de estructura web:

```
<!DOCTYPE html>
<html>
 <head>
  <meta charset="utf-8">
  <title>Mi primer sitio web</title>
 </head>
 <body>
  <h1>Encabezado nivel 1</h1>
        <h2>Mi nombre es:</h2> Tu  Nombre y Apellido
        <h2>Objetivos del curso:</h2>  Lorem ipsum
        <h3>Mini bio:</h3>  Lorem ipsum

 </body>
</html>
```

HTML5

HTML5 incorpora **etiquetas semánticas** que no sólo generan estructura, sino que también indican el tipo de contenido al desarrollador y al navegador web.



Etiquetas HTML5

- **<section>:** Define una sección en un documento.
- **<nav>:** Define un bloque que contiene enlaces de navegación.
- **<article>:** Define contenido autónomo que podría existir independientemente del resto del contenido.

Etiquetas HTML5

- **<aside>:** Define contenidos relacionados con el resto del contenido de la página.
- **<main>:** Define el contenido importante en el documento. Solamente existe un elemento <main> en el documento.

Etiquetas HTML5

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

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

- **<form>:** Define un formulario

Multimedia en html

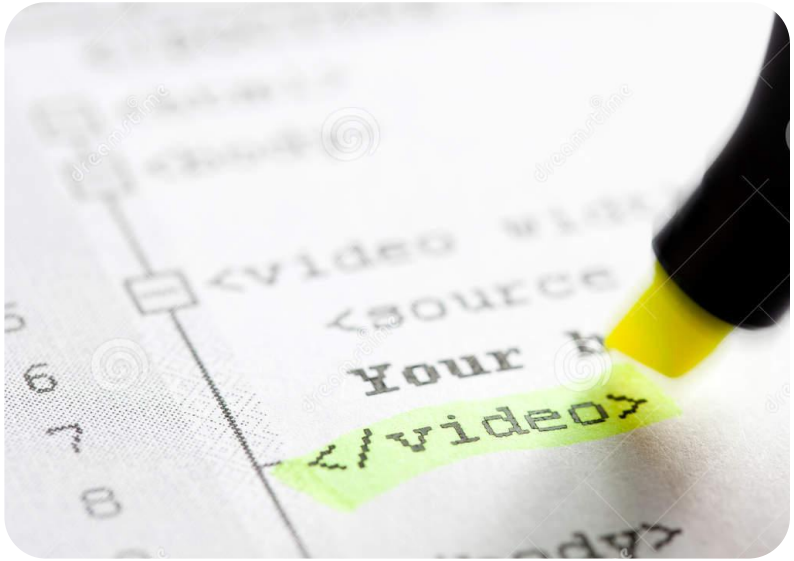
ALT

El “alt” es un texto que debe representar la foto que se está visualizando.

Debe ser conciso y breve, dejando en claro de qué se trata la imagen.

```
<img src="smiley.gif" alt="Carita sonriente" />
```

Etiqueta video




```
<video src="mivideo.mp4" controls>

  Tu navegador no implementa el elemento
<code>video</code>.

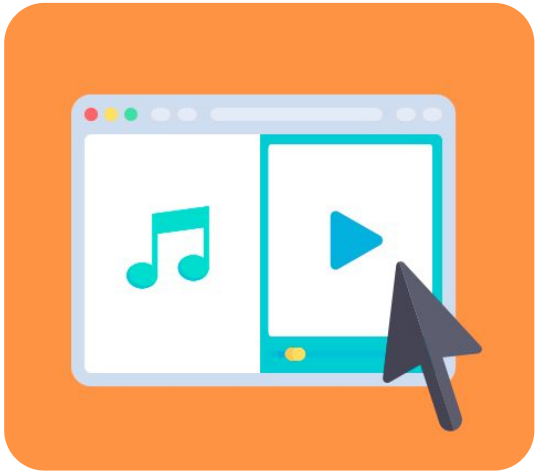
</video>
```

Etiqueta de audio



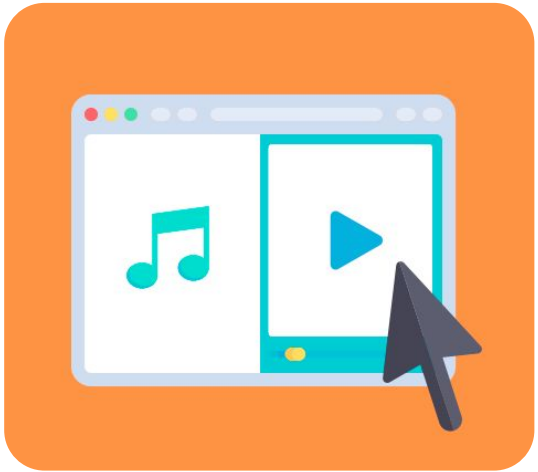
```
<audio src="perfect.mp3" preload="auto"
controls></audio>
```

Atributos de la etiqueta <AUDIO>



- Controls
- Autoplay
- Loop
- Preload

Atributos de la etiqueta <AUDIO>



- *"none"*
- *"auto"*
- *"metadata"*

Se pueden especificar múltiples fuentes de archivos usando el elemento **<source>**, para proporcionar vídeo o audio codificados en formatos diferentes.

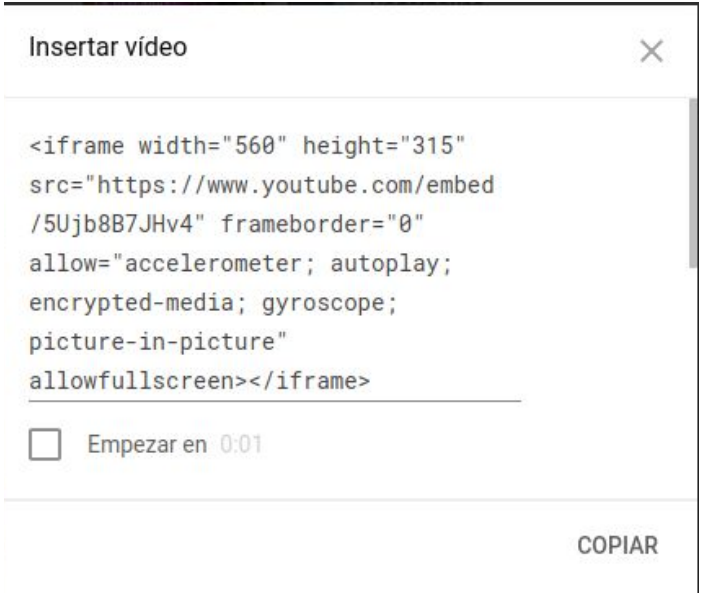
```
<video controls>
  <source src="foo.ogg" type="video/ogg">
  <source src="foo.mp4" type="video/mp4">
  Tu navegador no implementa el elemento <code>video</code>.
</video>
```

Etiqueta IFRAME

Es un elemento HTML que permite insertar o incrustar un documento HTML dentro de un documento HTML principal.

```
<iframe src="pagina_fuente.html" width=290 height=250>Texto para cuando el navegador no conoce la etiqueta iframe</iframe>
```

Etiqueta IFRAME



HTML: Listas y formularios

OBJETIVOS

- Comprender la sintaxis de HTML.
- Conocer las diferentes etiquetas y el uso de cada una.
- Dominar el concepto de web semántica, y las etiquetas HTML5 de estructura.

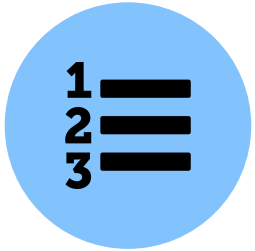
- **HTML:** es un "lenguaje" de marcado de etiquetas, que permite crear documentos para web.
- **Doctype:** Sirve para indicar que tu documento está escrito siguiendo la estructura determinada por un DTD concreto.

Listas

TIPOS DE LISTAS



Listas no ordenadas



Listas ordenadas



Listas de descripción

Tipos de listas

Casos de uso: paso a paso de un procedimiento (como una receta de cocina); características de un elemento; galería de imágenes; el menú de una página web, etc.

¿Viñetas o números?

- Las **listas numéricas** establecen un orden en la lectura de sus ítems.
- Las **listas de viñetas** no representan ningún orden o importancia entre sus ítems. Son elementos compuestos.

¿Viñetas o números?

- : define una lista ordenada de artículos (numéricas).
- : define una lista de artículos sin orden (viñetas).
- : define un artículo de una lista.

Ejemplo

Ambas listas se deben insertar mediante la etiqueta (list-item).

(empresa con listado/sin orden)

```
<ul>
  <li>Empresa</li>
  <li>Producto</li>
  <li>Servicios</li>
  <li>Contacto</li>
</ul>
```

- Empresa
- Producto
- Servicios
- Contacto

Anidar listas

- Computadoras portátiles:
  - Procesador I4.
  - Procesador I5.
- Computadoras de escritorio:
  - Procesador Pentium.
  - Procesador Celeron.

Listas de descripción

Una lista de descripción es una lista de términos, con una descripción de cada uno.

La etiqueta <dl> define la lista de descripción; la etiqueta <dt> define el término, nombre o ítem y la etiqueta <dd> describe cada término, nombre o ítem.

Listas de descripción

```
<dl>
  <dt>Café</dt>
  <dd>bebida que proviene
de una planta</dd>

  <dt>Leche</dt>
  <dd>bebida que se
extrae de un
mamífero</dd>
</dl>
```

Formularios

Formularios



Formularios

El elemento HTML **<form>** se usa para **capturar datos ingresados por el usuario.** Su propósito principal es permitir al usuario seleccionar o insertar información y enviarla al servidor para ser procesada.

Formularios

El formulario es un contenedor de distintos tipos de **<input>** (text fields, checkboxes, radio buttons, submit buttons, etc)

Formularios

Los formularios requieren de ciertos **atributos** para determinar cómo se envía la información al servidor:

- **name**—Especifica el nombre del formulario.
- **method**—Determina el método a utilizar para enviar la información al servidor.

Formularios

method:

- GET se usa para enviar una cantidad limitada de información de forma pública (los datos son incluidos en la URL)
- POST se utiliza para enviar una cantidad ilimitada de información de forma privada

Formularios

- **action**—Declara la URL del archivo en el servidor.
- **target**—Determina dónde se mostrará la respuesta recibida desde el servidor: _blank (nueva ventana), _self (mismo recuadro), _parent (recuadro padre), y _top (la ventana que contiene el recuadro). El valor _self se declara por defecto.

Formularios

- **enctype**—Este atributo declara la codificación aplicada a los datos que envía el formulario. Puede tomar tres valores: 1)**application/x-www-form-urlencoded (por defecto)**, 2)**multipart/form-data**, 3)**text/plain**.

Formularios

- **accept-charset**—Este atributo declara el tipo de codificación aplicada al texto del formulario. Los valores más comunes son UTF-8 e ISO-8859-1. El valor por defecto se asigna al documento con el elemento **<meta>**, ubicado en la sección **<head>**.

Ejemplo

```
<!DOCTYPE html>
<html lang="es">
<head>
<meta charset="utf-8">
<title>Formularios</title>
</head>
<body>
        <section>
                <form  name="formulario"  method="get"  action="procesar.php">
                </form>
        </section>
</body>
</html>
```

Elementos

Un formulario puede incluir diferentes herramientas para permitir al usuario seleccionar o insertar información. HTML incluye múltiples elementos para crear estas herramientas.

Elementos más utilizados

- **<input>**—Este elemento crea un campo de entrada.
- **<textarea>**—Crea un campo de entrada para insertar múltiples líneas de texto. El tamaño se puede declarar en números enteros usando los atributos **rows** y **cols**, o en píxeles con estilos CSS.

Elementos más utilizados

- **<select>**—Este elemento crea una lista de opciones que el usuario puede elegir. Trabaja junto con el elemento **<option>** para definir cada opción y el elemento **<optgroup>** para organizar las opciones en grupos.

Elementos más utilizados

- **<button>**—Crea un botón. Incluye el atributo **type** para definir el propósito del botón. Los valores disponibles son **submit** para enviar el formulario (por defecto), **reset** para reiniciar el formulario, y **button** para realizar tareas personalizadas.

Elementos más utilizados

- **<label>**—Crea una etiqueta para identificar un elemento de formulario.
- **<fieldset>**— Agrupa otros elementos de formulario para crear secciones dentro de formularios extensos. Puede contener un elemento **<legend>**.

Elemento INPUT

Puede adoptar diferentes características y aceptar varios tipos de valores dependiendo del valor de su atributo **type**.

Valores más comunes de INPUT

- **text**—Entrada para texto genérico.
- **email**—Entrada para cuentas de correo.
- **search**—Entrada para términos de búsqueda.
- **tel**—Entrada para números de teléfono.
- **number**—Entrada para insertar números.

Valores más comunes de INPUT

- **range**—Entrada para un rango de números.
- **date**—Entrada para insertar una fecha.
- **datetime-local**—Entrada para insertar fecha y hora.
- **week**—Entrada para número de la semana.
- **month**—Entrada para insertar el número del mes.

Valores más comunes de INPUT

- **time**—Entrada para insertar horas y minutos.
- **hidden**—Oculta el campo de entrada.
- **password**—Entrada para insertar una clave.
- **color**—Entrada para insertar un color.

Valores más comunes de INPUT

- **checkbox**—Permite al usuario activar o desactivar una opción.
- **radio**—Botón de opción.
- **file**—Entrada para seleccionar un archivo en la computadora o dispositivo del usuario.

Valores más comunes de INPUT

- **button**—Este valor genera un botón. El botón trabaja como el elemento **<button> de tipo button**. No realiza ninguna acción por defecto; la acción debe ser definida desde JavaScript.

Valores más comunes de INPUT

- **submit**—Botón para enviar el formulario.
- **reset**—Botón para reiniciar el formulario.
- **image**—Este valor carga una imagen que se usa como botón para enviar el formulario, *debe incluir el atributo src para especificar la URL de la imagen*.

Valores más comunes de INPUT

Para incluir un formulario en nuestro documento, tenemos que declararlo con el elemento <form>, y luego incorporar en su interior todos los elementos que el usuario necesitará para insertar la información y enviarla al servidor.

Ejemplo

Si queremos que el usuario inserte su nombre y edad, tenemos que incluir dos campos de entrada para texto y un tercer elemento para crear el botón con el que enviar el formulario.

Ejemplo

```
<form name="formulario" method="get" action="procesar.php">
    <input type="text" name="nombre">
    <input type="text" name="edad">
    <input type="submit">
</form>
```

Atributo “VALUE”

Representa la etiqueta del botón, la cual es normalmente mostrada por los navegadores dentro de éste.

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

Atributo “VALUE”

- Input de tipo “*submit*”: envía el formulario.
- Input de tipo “*reset*”: resetea el formulario.
- Input de tipo “*button*”: no tiene acciones por defecto.

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

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”*.

Grupos de controles de selección

- **Botones de radio:** sólo se puede elegir una opción.
- **Casillas de chequeo:** de toda la lista de opciones, el usuario puede optar por una, todas o ninguna opción.
- **Menú desplegable:** sólo es posible seleccionar una opción.

Atributo “VALUE”

Es el valor que se enviará al enviarse el formulario.

Botones de radio:

```
<form>
  <div>hombre</div>
  <input type="radio" name="genero" value="hombre" />
  <div>mujer</div>
  <input type="radio" name="genero" value="mujer" />
 <div>otro</div>
  <input type="radio" name="genero" value="otro" />
</form>
```

Atributo “VALUE”

Casillas de chequeo:

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

Etiqueta <LABEL>

Define formalmente a cada elemento de un formulario. Ayuda a generar un formulario accesible.

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

Etiqueta <LABEL>

Su principal atributo es "for", que va a referenciar a "label" con su elemento del formulario. El valor "for" debe ser igual al valor "id" o "name" del elemento.

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

Menú desplegable

Es el llamado **combo-box, selector o menú**. De toda la lista, se puede elegir una opción (aunque tiene un atributo que permite cambiarlo).

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

Conjunto de campo

Las etiquetas <fieldset> y <legend> se utilizan en conjunto. La primera, crea grupos de elementos del formulario que posean un mismo propósito; la segunda, define formalmente el propósito del elemento fieldset.

Conjunto de campo

Se estructuran de la siguiente manera:

```
<form>
 <fieldset>
  <legend>Talle de remera</legend>
   <!-- Aquí irán los elementos de formulario -->
 </fieldset>
</form>
```

Atributos globales

HTML define atributos globales que son exclusivos de elementos de formulario:

- required.
- disabled.
- readonly.

Atributos globales

- placeholder.
- autocomplete.
- novalidate.
- multiple.
- autofocus.
- pattern.
- spellcheck.

Atributos globales

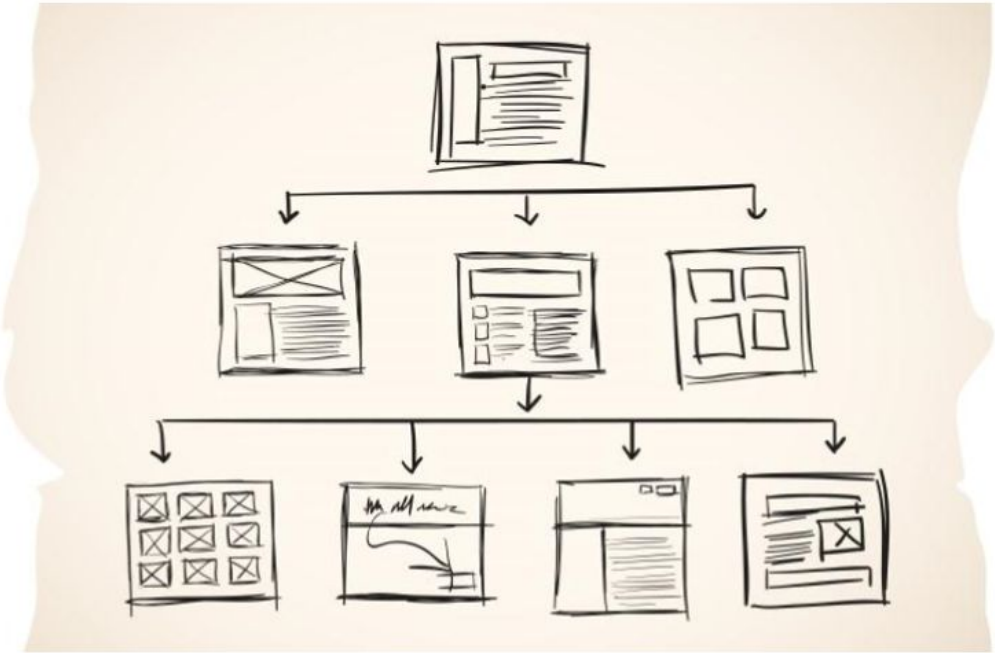
Los atributos **disabled** y **readonly** no permiten al usuario interactuar con el elemento.

- **disabled:** Mostrar al usuario que el elemento puede estar disponible en otras condiciones.

Atributos globales

- **readonly:** Se implementa cuando solo existe un valor posible y no queremos que el usuario lo cambie.

ENLACES



Arquitectura de un sitio

Los **enlaces,** también conocidos como links o anchors, se utilizan para relacionar partes del mismo documento. Por defecto, se visualizan azules y subrayados.

Arquitectura de un sitio

Para crear uno, es necesario utilizar la etiqueta de ancla **<a>** con el atributo *"href"*, que establecerá el destino al que apunta. Por ejemplo:

```
<a href="productos.html">Productos</a>
```

ENLACES RELATIVOS, ABSOLUTOS E INTERNOS

Enlaces relativos

Los **enlaces relativos** son aquellos que apuntan a páginas ubicadas dentro del mismo proyecto.

```
<a href="contacto.html">Contacto</a>
```

Enlaces relativos

En caso de que el archivo se encuentre en un directorio específico, el mismo deberá ser mencionado.

```
<a href="imagenes/mapa.jpg">ver mapa</a>
```

Enlaces absolutos

Los **enlaces absolutos** son aquellos cuyo destino apunta a un documento que está fuera del sitio, y debe ser especificado utilizando la URL completa:

```
<a href="http://www. fullstack.com/frontend">Desarrollo Web</a>
```

Enlaces internos

Los **enlaces internos** permiten referenciar secciones de tu página, para lo cual se utiliza el **id**:

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

Enlaces internos

También puedes usar como destino una sección específica que se encuentre en una página distinta de la actual:

```
<a href="contacto.html#formulario">Formulario de contacto</a>
```