**# Editores de texto**

Lo primero que vas a necesitar para iniciar el bootcamp es un **\*\*editor de texto\*\***. Un **\*\*editor de texto\*\*** es una aplicación que nos permite crear y editar archivos de texto como su nombre lo indica.

A diferencia de un procesador de palabras (como Microsoft Word), un editor de texto se utiliza para crear archivos de texto **\*\*sin formato\*\***.

Cada sistema operativo trae algún editor de texto como **\*\*Bloc de Notas\*\*** en Windows o **\*\*TextEdit\*\*** en Mac. Sin embargo, existen editores de texto especializados para programadores que tienen varias ventajas sobre el editor que viene en tu sistema operativo:

- Resalta la sintaxis de acuerdo al lenguaje de programación para facilitar su lectura.

- Ayuda a auto completar el código.

- Es altamente personalizable a través de archivos de configuración o plugins.

  Ejemplos de editores de texto para programadores incluyen [Sublime Text](https://www.sublimetext.com/), [Atom](https://atom.io/) y [VSCode](https://code.visualstudio.com/), entre muchos otros.

Para efectos de ejemplo te invitamos a instalar **\*\*VSCode\*\*** ingresa [**\*\*aquí\*\***](https://code.visualstudio.com/download) y selecciona tu sistema operativo para continuar con el proceso de instalación.

**# Introducción a HTML**

**\*\*Con HTML puedes hacer visible tu contenido en las páginas web\*\***.

Las páginas web son visualizadas a través de navegadores web como chrome, firefox, brave entre otros, estos son clientes web que se encargan entre otras cosas de interpretar el HTML para visualizar a los usuarios los contenidos de una manera estructurada.

**## ¿Qué es HTML?**

Es un lenguaje muy sencillo compuesto de elementos, que se pueden aplicar a piezas de texto para darles un significado diferente en un documento, por medio de etiquetas puedes identificar si es un párrafo, si es una lista, si es una tabla de contenidos, si quieres insertar imagenes o videos, entre otras muchas otras posibilidades que HTML ofrece.

HTML (Lenguaje de Marcas de Hipertexto, del inglés \\_HyperText Markup Language) es el componente más básico de la web y no es un lenguaje de programación.

**## Anatomía de un documento HTML**

Como venimos hablando, HTML está compuesto de múltiples elementos que por si solos no logran dar significado a un texto, para que estos elementos sean interpretados de manera correcta, necesitan de un contexto, de un elemento superior que los enmarque globalmente y es por esto que hablamos de la anatomía de un documento HTML como lo vemos a continuación:

    <!DOCTYPE html>

    <html>

        <head>

            <meta charset="utf-8">

            <title>Mi pagina de prueba</title>

        </head>

        <body>

            <img src="images/firefox-icon.png" alt="Mi imagen de prueba">

        </body>

    </html>

En la estructura básica tienes los siguientes elementos:

- `<!DOCTYPE html>`: el tipo de documento. Es un preámbulo requerido. Hoy día es simplemente un artefacto antiguo que a nadie le importa, pero que debe ser incluido para que todo funcione correctamente. Por ahora, eso es todo lo que necesitas saber.

- `<html></html>` — el elemento [`<html>`](https://developer.mozilla.org/es/docs/Web/HTML/Element/html) encierra todo el contenido de la página entera y, en ocasiones, se le conoce como el elemento raíz.

- `<head></head>` — el elemento [`<head>`](https://developer.mozilla.org/es/docs/Web/HTML/Element/head) actúa como un contenedor de todo aquello que quieres incluir en la página HTML que no es contenido visible por los visitantes de la página. Incluye cosas como palabras clave, una descripción de la página que quieres que aparezca en resultados de búsquedas, código CSS para dar estilo al contenido, declaraciones del juego de caracteres, etc.

- `<meta charset="utf-8">` — [`<meta>`](https://developer.mozilla.org/es/docs/Web/HTML/Element/meta). Este elemento establece el juego de caracteres que tu documento usará en `utf-8`, que incluye casi todos los caracteres de todos los idiomas humanos.

- `<title></title>` — el elemento [`<title>`](https://developer.mozilla.org/es/docs/Web/HTML/Element/title) establece el título de tu página, que es el título que aparece en la pestaña o en la barra de título del navegador cuando la página es cargada, y se usa para describir la página cuando es añadida a los marcadores o como favorita.

- `<body></body>` — el elemento [`<body>`](https://developer.mozilla.org/es/docs/Web/HTML/Element/body) Encierra todo el contenido que deseas mostrar a los usuarios web que visiten tu página, ya sea texto, imágenes, videos, juegos, pistas de audio reproducibles, y demás.

**## Anatomía de un elemento HTML:**

Un elemento se compone de varias partes en su definición, a continuación detallamos las principales:

1.  **\*\*La etiqueta de apertura\*\***: consiste en el nombre del elemento encerrado por **\*\*paréntesis angulares\*\*** (< >) de apertura y cierre. Establece dónde comienza o empieza a tener efecto el elemento.

2.  **\*\*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.

3.  **\*\*El contenido\*\***: este es el contenido del elemento.

4.  **\*\*El elemento\*\***: la etiqueta de apertura, más la etiqueta de cierre, más el contenido equivale al elemento.

Un elemento puede tener uno o muchos atributos para caracterizar o definir parte del comportamiento de cada uno, o para agregar identificadores entre otros, los atributos siempre deben tener:

1.  Un espacio entre este y el nombre del elemento (o del atributo previo, si el elemento ya posee uno o más atributos).

2.  El nombre del atributo, seguido por un signo de igual (**\*\*=\*\***).

3.  Comillas de apertura y de cierre, encerrando el valor del atributo.

Los atributos siempre se incluyen en la etiqueta de apertura de un elemento, nunca en la de cierre.

**## Elementos básicos de HTML:**

Los elementos de HTML los podemos agrupar según su intencionalidad en 5 grandes grupos, elementos semánticos, de multimedia, de cabecera, de formularios y de scripting; a continuación describiremos cada uno de estos:

**### Semánticos:**

En esta agrupación encontramos aquellos elementos que le pueden llegar a dar significado a los textos que queremos transmitir, al igual que funcionalidad para inducir alguna acción en los usuarios y a su vez podemos subdividirlo en 5 grupos así:

1.  **\*\*De texto\*\***: Permiten dotar de un significado especial al contenido como por ejemplo.

| **\*\*Elemento\*\*** | **\*\*Descripción\*\***                                                               |

| ------------ | ----------------------------------------------------------------------------- |

| `<strong>`   | Fragmento de texto **\*\*importante\*\*** o **\*\*palabras clave\*\***                        |

| `<em>`       | Fragmento de texto *\_enfatizado\_* respecto a la frase que lo contiene.          |

| `<mark>`     | Fragmento de texto resaltado, simulando estar marcado con rotulador amarillo. |

| `<i>`        | Fragmento de texto con voz o tono alternativo al resto.                       |

| `<b>`        | Fragmento de texto sin importancia destacable (*\_fines utilitarios\_*).          |

| `<u>`        | Fragmento de texto para nombres propios o escritura incorrecta intencionada.  |

Los anteriores son solo algunos ejemplos de los elementos de texto que existen en el lenguaje, puedes ampliar tu conocimiento consultando [aquí](https://developer.mozilla.org/es/docs/Web/HTML/Element)

2.  **\*\*De agrupación\*\***: se utilizan para agrupar y organizar información, veamos algunos ejemplos:

| **\*\*Elemento\*\*** | **\*\*Descripción\*\***                                                  |

| ------------ | ---------------------------------------------------------------- |

| `<div>`      | Capa o división utilizada para agrupar varias etiquetas HTML.    |

| `<p>`        | Define un párrafo de texto.                                      |

| `<pre>`      | Establece un texto pre formateado.                               |

| `<ul>`       | Define una lista sin orden. Se trata de la etiqueta contenedora. |

| `<ol>`       | Define una lista numerada (con orden). Etiqueta contenedora.     |

| `<li>`       | Define el elemento de una lista.                                 |

3.  **\*\*Enlaces o hipervínculos\*\***: La idea es establecer una referencia a una dirección o URL donde está alojado ese otro documento de destino, que puede ser una página web, un archivo PDF, una imagen o cualquier otro tipo de documento. Para este caso tenemos el elemento `<a>`

4.  **\*\*De sección\*\***: En versiones anteriores a HTML 5 utilizábamos el elemento de agrupación `<div>` en con junto con sus atributos para definir secciones de las páginas web como podemos ver a

continuación:

    <div class="articulo">

      <h1>Título del artículo</h1>

      <p class="intro">Pequeña introducción.</p>

      <p class="contenido">Aquí va el texto del artículo en cuestión con sus detalles.</p>

      <p class="pie">Escrito por Manz</p>

    </div>>

con la introducción de los elementos de sección podemos escribir de una manera semánticamente más clara lo anterior de la siguiente forma:

    <article>

        <header>

            <h1>Título del artículo</h1>

            <p class="intro">Pequeña introducción.</p>

        </header>

        <p class="contenido">Aquí va el texto del artículo en cuestión con sus detalles.</p>

        <footer>

            <p>Escrito por Manz.</p>

        </footer>

    </article>

De esta forma, preparamos nuestro documento HTML para que cualquier navegador, robot de buscador o aplicación o sistema informático sea capaz de leer el documento HTML y conocer perfectamente la naturaleza del contenido de dicha sección.

5.  **\*\*De tablas\*\***: son una forma sencilla de mostrar datos claramente. Además, si se construyen de forma semántica y correctamente, es muy sencillo darle estilos desde CSS y cambiar su diseño con unas cuantas propiedades, puesto que mediante las etiquetas que la componen se puede hacer referencia a cada parte de la misma, a continuación los elementos básicos:

| **\*\*Elemento\*\*** | **\*\*Descripción\*\***                                                   |

| ------------ | ----------------------------------------------------------------- |

| `<table>`    | Capa o división utilizada para agrupar varias etiquetas HTML.     |

| `<tr>`       | **\*\*Table Row\*\***. Etiqueta contenedora de cada fila de la tabla.     |

| `<td>`       | **\*\*Table Data\*\***. Cada una de las celdas de la tabla.               |

| `<th>`       | **\*\*Table Header\*\***. Cada una de las celdas de cabecera de la tabla. |

**### De multimedia:**

En esta agrupación encontramos aquellos elementos que nos permiten cargar contenido de multimedia a las paginas web como lo pueden ser imágenes, videos, audios, subtítulos, embebidos de otros sitios, entre otros, a continuación describimos algunas de ellas:

1.  Para imágenes `<img>`: si la imagen pertenece al contenido y tema tratado en esa página, debería incluirse mediante una etiqueta HTML `<img>` un ejemplo básico puede ser `<img src="https://lenguajehtml.com/img/logo.png" alt="Logotipo de HTML5" width="400" height="453" />`

    en la versión 5.1 se incluyen los elementos `<picture> y <source>` que le dan una gran versatilidad a la carga de imágenes permitiendo definir múltiples source para que dependiendo del navegador seleccione el que más se adapte al mismo.

    Los atributos del elemento `<img>` son:

    `src` Indica el nombre o la URL de la imagen a mostrar. **\*\*Atributo obligatorio\*\***.

    `alt` Establece un texto alternativo que describa la imagen a mostrar. **\*\*Atributo obligatorio\*\***.

    `width`Indica el ancho de la imagen en píxels (sin la unidad). Se puede hacer desde CSS.

    `height`Indica el alto de la imagen en píxels (sin la unidad). Se puede hacer desde CSS.

2.  Para contenido externo: Las siguientes etiquetas sirven para incrustar contenido externo:

    `<iframe>`- Permite incrustar contenido externo en «vivo», sus atributos son los siguientes `src`, `srcdoc`, `name`, `width`, `height`.

    `<embed>`- Permite incrustar contenido interactivo, sus atributos son los siguientes `src`, `type`, `width`, `height`

    `<object>`- Permite incrustar contenido externo con fallbacks, sus atributos son los siguientes `data`, `type`, `name`, `form`, `width`, `height`

    `<param>`- Define parámetros de un elemento `<object>`, sus atributos son `name`, `value`

3.  Para video `<video>`: En HTML5 se introduce la interesante posibilidad de **\*\*mostrar videos directamente\*\*** desde nuestro navegador. De hecho, si arrastramos un video a la ventana del navegador, veremos que comienza a reproducirse en él. Para poder insertar videos en nuestras páginas HTML tenemos que utilizar la etiqueta `<video>`, que junto a la etiqueta `<source>` podremos utilizar estas capacidades multimedia de HTML5.

    los atributos del elemento video son los siguientes:

    `src` *\_Dirección URL\_* del Video a reproducir. Obligatoria si actúa como etiqueta contenedora.

    `poster` *\_Dirección URL\_* de una imagen de muestra a modo de presentación.

    `preload` **\*\*auto\*\*** | `metadata` | `none`, Indica como realizar la precarga del video.

    `mediagroup`*\_nombre\_*, Establece un nombre para un grupo de contenidos multimedia.

    `autoplay`Comienza a reproducir el video automáticamente.

    `loop`Vuelve a iniciar el video cuando finaliza su reproducción (bucle).

    `muted`Establece el video sin sonido (silenciado).

    `controls`Muestra los controles de reproducción. Por defecto no se muestran.

    `width`*\_tamaño\_* Indica el tamaño de ancho del video.

    `height`*\_tamaño\_* Indica el tamaño de alto del video.

    Un ejemplo puede ser `<video src="video.mp4" width="640" height="480"></video>`

4.  Para audio: En este caso, utilizaremos la etiqueta `<audio>` que funciona exactamente igual que `<video>`, pero con relación a archivos de audio. los atributos del elemento son:

    `src`\_Dirección URL\_del audio a reproducir. Obligatoria si actúa como etiqueta contenedora.

    `preload`**\*\*auto\*\*** | `metadata` | none, Indica como realizar la precarga del audio.

    `mediagroup`\\_nombre\_, Establece un nombre para un grupo de contenidos multimedia.

    `autoplay` Comienza a reproducir el audio automáticamente.

    `loop`Vuelve a iniciar el audio cuando finaliza su reproducción (bucle).

    `muted`Establece el audio sin sonido (silenciado).

    `controls`Muestra los controles de reproducción. Por defecto no se muestran.

    Un ejemplo puede ser `<audio src="audio.mp3" preload="none" controls></audio`.

**### De cabecera:**

Hay dos etiquetas HTML principales: la cabecera de la página o `<head>` y el cuerpo de la página o `<body>` . El primero de ellos, del que hablaremos en este apartado, se encarga de contener **\*\*etiquetas de metadatos\*\*** (*\_información sobre el documento\_*) así como establecer **\*\*relaciones con otros documentos\*\***. Existen varios elementos a utilizar en la cabecera y los listamos a continuación:

`<title>`Título de la página (pestaña o título del navegador o en buscadores).

`<base>`- URL base del documento (*\_usado para gestionar rutas relativas\_*), sus atributos son `href`, `target`.

`<link>`- Establece una relación del documento actual con otro externo, sus atributos son `href`, `hreflang`, `rel`, `media`, `type`.

`<meta>`- Establece un metadato específico en el documento actual, sus atributos son `name`, `content`, `http-equiv`, `charset`.

`<style>` Crea estilos CSS que afectarán únicamente al documento actual, sus atributos son `media`, `type`.

`<script>` Indica un script a cargar o ejecutar en la página actual, sus atributos son `src`, `type`, `charset`, `async`, `defer`.

La etiqueta con mayor número de posibilidades en la cabecera de un documento HTML es la etiqueta `<meta>`. En ella, y a través de los atributos **\*\*name\*\*** y **\*\*content\*\*** podemos indicar una gran cantidad de metadatos al documento. Veamos los más conocidos:

`description`*\_texto\_*,Indica la descripción de la página que aparece en buscadores.

`keywords`*\_palabras clave\_*, Lista de palabras clave separadas por comas. Google no la tiene en cuenta.

`author`*\_nombre\_*, Indica el nombre del autor de la página.

`language`*\_idioma\_*, [Código ISO 639-1](https://es.wikipedia.org/wiki/ISO\_639-1) del idioma del documento HTML.

`generator`*\_software\_*, Indica el software utilizado para crear la página web.

`theme-color`*\_color\_*,[Color en formato hexadecimal](https://lenguajecss.com/css/colores-y-fondos/colores-css/) de la barra de navegación del navegador.

`viewport`*\_parámetros\_*, Comportamiento de la región visible del navegador.

Al igual que vimos como establecer un título y descripción para un documento HTML, es posible definir una serie de **\*\*etiquetas de metadatos\*\*** orientadas a **\*\*redes sociales\*\*** como Facebook o Twitter. De esta forma, al margen de los datos del documento HTML, podemos indicar títulos o descripciones diferentes, orientadas al público de cada red social como lo son [Open Graph](http://ogp.me/) y [Twitter Cards](https://dev.twitter.com/cards/overview)

**### De formularios:**

HTML5 proporciona una serie de etiquetas interesantes para utilizar en formularios (*\_además de las ya existentes en versiones anteriores\_*). Los **\*\*formularios\*\*** son una forma sencilla de establecer mecanismos para que el usuario pueda introducir información en una página web de forma sencilla e intuitiva, y el sitio web sea capaz de procesarla correctamente y añadirla en una base de datos de la página, enviarla por email o procesarla para mostrar información final al usuario.

Nuestros **\*\*objetivos al crear un formulario\*\*** HTML5 deben ser los siguientes:

- Hacer lo más **\*\*sencillo\*\*** posible el proceso de inserción de datos por parte del usuario.

- Intentar que la experiencia de usuario sea lo más **\*\*agradable\*\*** posible.

- Intentar que los datos introducidos estén en un **\*\*formato predecible\*\*** y esperado.

- Reducir los errores al introducir datos en la medida de lo posible.

- Comunicar claramente si ocurren errores al introducir datos.

Para comenzar con la creación de un formulario, el primer paso es indicar una etiqueta contenedora `<form>`, que incluirá toda la información que se quiere recoger en ese formulario.

La etiqueta `<form>` dispone de varios **\*\*atributos\*\*** para utilizar:

`action` - *\_URL\_*, Dirección URL del back-end donde se enviará la información del formulario.

`method`- **\*\*get\*\*** | `post`, Método HTTP de envío. `GET` a través de URL, `POST` para envío extenso.

`name`-*\_nombre\_*. Nombre del formulario. Útil para procesar posteriormente.

`target`-*\_destino\_*,Nombre del lugar donde se abrirá el formulario. `\\_blank` para nueva pestaña.

`enctype`-*\_tipo\_*,Codificación para el envío del formulario. Importante para envío de archivos.

`accept-charset`-*\_codificación\_*, Fuerza a utilizar una codificación en los parámetros de texto del formulario.

`autocomplete`-**\*\*on\*\*** | `off`,Activa o desactiva el autocompletado para todos los campos del formulario.

`novalidate`Con este atributo presente, el formulario obvia la validación HTML5.

Un ejemplo de un primer contener de formulario podría ser `<form name="formulario" method="post" action="/send.php"></form>`.

Para camputar información en los formularios HTML nos provee un elemento en particular para cada tipo, a continuación listamos algunos de gran utilidad:

| **\*\*Elemento\*\*** | **\*\*Ejemplo\*\***| **\*\*Información a obtener\*\***|

| --- | --- | -- |

| `<input>` `<textarea>` | Nombres, apellidos, direcciones físicas... | Información de texto |

| `<input>` para números | Edades, precios, cuantías... | Números o cantidades |

| `<input>` para fechas | Fecha de nacimiento, inicio de evento... | Fechas u horas |

| `<input type="checkbox">` | Si/No, Opción A/B, ON/OFF... | Verdadero/falso|

| `<input type="radio"><select>` | (Elegir una opción de 2 ó más posibles)| Opción única|

| `<select multiple><input type="checkbox">` | (Elegir varias opciones de 2 ó más posibles)| Varias opciones|

| `<datalist>` | (Elegir una opción o indicar una propia)| Opción única abierta|

| `<input type="color">` | (Escoger un color o tonalidad)| Selección de color|

| `<input type="file">` | (Escoger un archivo para enviar)| Selección de archivo|