# Primeras Lineas de Html y Css

Estimad@s, espero que estén a gusto con el curso. Como verán, la clase estuvo muy enfocada en HTML y CSS, porque son fundamentales para el avance en las diferentes herramientas a utilizar. Intento a través de esta pequeña guía que tengan un soporte para todo lo que avancen de manera individual maquetando con HTML y CSS.

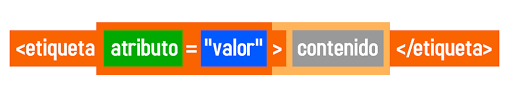
# HTML

## ¿Qué es HTML?

Primero y principal, es necesario tener en claro que HTML no es un lenguaje de programación, es un lenguaje de marcado de hipertexto o “**H**yper**T**ext **M**arkup **L**anguage” por el desarrollo de sus iniciales en inglés, básicamente este lenguaje se escribe en su totalidad con elementos, estos elementos están constituidos por etiquetas, contenido y atributos.

Cuando un cliente (por ejemplo, navegador web) ingresa a un website, interpreta todo el contenido escrito en HTML, y a partir de allí va renderizando el sitio según el contenido de las etiquetas, con sus atributos.

La estructura básica de un elemento HTML es la siguiente:



Cierre del elemento

Indica al cliente el inicio de un elemento en particular

Contenido que será renderizado por el navegador, en base a la etiqueta y los atributos de la misma

Los atributos pueden ser muchos y variados. Los más comunes son las clases (atributo **class**=”nombre de clase”) y los identificadores (atributo **id**=”identificador”).

## Estructura Básica de un documento HTML

De manera general, una página web construida con HTML tiene el siguiente esquema:

<HTML>

<HEAD>

<TITLE>Título de la página</TITLE>

Aquí se ingresan los contenidos relevantes para el navegador, importación de archivos externos y demás.

</HEAD>

<BODY>

Contenido

</BODY>

<FOOTER>

     Pié de página

</FOOTER>

</HTML>

Entre las etiquetas <html> y <head> se suelen colocar otras opcionales, como por ejemplo:

<meta name="description" content="Descripcion que será renderizada por los motores de búsqueda">

<meta name="keywords" content="palabras,claves,del,sitio">

La cabecera es la sección comprendida entre <head> y </head>.

En ella se encuentra necesariamente el título (entre las etiquetas <title> y </title>).

El título de la página debe describir su contenido por ejemplo:

<TITLE> Mi Sitio Web - Bienvenido </TITLE>

Dentro de la cabecera podemos importar los archivos externos que necesitemos (Si lo que deseamos es importar archivos JS, lo mejor es hacerlo en el footer para optimizar la carga del sitio).

Ejemplo de una importación de archivos CSS

<link rel="stylesheet" href="estilo.css" type="text/css" >

A través de la etiqueta link (No necesita cierre) le avisamos al navegador sobre la utilización de un link externo. Con la etiqueta rel=”stylesheet” indicamos que estaremos cargando una hoja de estilo. Con type=”text/css”, indicamos que el lenguaje del mismo estará en CSS, y con la etiqueta href=”” indicamos la ubicación del mismo.

### Encabezados y Párrafos HTML

Los encabezados, de manera similar a un editor de texto, nos permiten establecer títulos y comienzos de las diferente secciones. Su sintaxis es la siguiente:

<h1>Encabezado 1</h1>

<h2>Encabezado 2</h2>

<h3>Encabezado 3</h3>

<h4>Encabezado 4</h4>

<h5>Encabezado 5</h5>

<h6>Encabezado 6</h6>

Un párrafo, en cambio, tiene la siguiente sintáxis.

<p> Contenido del parrafo. </p>

## Estilos de Fuentes en HTML

Como verán, la lógica de HTMl es similar a la de un editor de texto.   
Estas son algunas de las etiquetas más comunes para manipular estilos de fuente

|  |  |
| --- | --- |
| <B>Texto en Negrita</b>  <I>Itálica</i>  <B><I>Negrita e Itálica</i></b>  <U>Subrayado</u>  <EM>Enfatizado</em>  <STRONG>Fuerte</strong>  <CODE>Code Texto</code>  <CITE> Citation Text</cite>  <KBD>Keyboard Text</kbd>  <SAMP>Sample Text</samp>  <TT>Teletype Text</tt>  <VAR>Variable Element Text</var>  <BIG>Texto grande</big>  <SMALL>Texto pequeño</small>  <SUB>Subindice</SUB>  <SUP>Superíndice</SUP>  <STRIKE>Texto tachado</STRIKE> | * **Texto en Negrita** * *Itálica* * ***Negrita e Itálica*** * Subrayado Solo Explorer * *Enfatizado* * **Fuerte** * Code Texto * *Citation Text* * Keyboard Text * Sample Text * Teletype Text * *Variable Element Text* * Texto grande * Texto pequeño * Subíndice * Superíndice * ~~Texto tachado~~ |

Estas etiquetas pueden introducirse en un Encabezado, párrafo, etc. El contenido que se encuentre dentro de la misma recibirá un estilo particular.

## Links (Hipervínculos)

Nos sirve para crear un enlace a otras páginas de internet, archivos o ubicaciones dentro de la misma página, direcciones de correo, o cualquier otra URL. Se realiza a través de la etiqueta <a> (anchor).

<a href="link al que llevará a hacer clic">Texto visible</a>

<!-- Dentro del href podemos ingresar URLS, o diferentes directrices, por ejemplo -->

<a href="https://google.com">Haciendo clic aquí se abrirá Google</a>

<a href="mailto:direccion@email.com">Haciendo clic aquí se abrirá nuestro cliente predeterminado de email</a>

<a href="https://wa.me/3543640112">Haciendo clic aquí se abrirá una conversación de whatsapp al número indicado</a>

Asimismo, una de las etiquetas más utilizadas en los hipervículos es target. La misma especificará si el link debe abrirse en una ventana diferente o en la misma.

<a href="http://google.com" target="\_blank">El atributo \_blank lo abrirá en una nueva ventana/pestaña</a>

<a href="http://google.com" target="\_self ">El atributo \_self lo abrirá en la misma ventana (opción por defecto) </a>

## Listas ordenadas y no ordenadas

De forma similar a cualquier editor de texto, nos permiten organizar el contenido en listas.

### Lista Ordenada

Se organizará siguiendo un órden determinado

<ol>

<li>Elemento 1 </li>

<li>Elemento 2 </li>

<li>Elemento 3 </li>

</ol>

Mostrará:

1. Elemento 1
2. Elemento 2
3. Elemento 3

### Lista No Ordenada

Se organizará sin seguir un órden determinado, sino con viñetas

<ul>

<li>Elemento 1 </li>

<li>Elemento 2 </li>

<li>Elemento 3 </li>

</ul>

Mostrará:

* Elemento 1
* Elemento 2
* Elemento 3

## Divs en HTML

Nos permiten organizar el contenido en contenedores, sobre los cuales podremos aplicar clases que afecten a todos los elementos contenidos, como así también trabajar de manera estructurada en nuestro código.

<div id="id\_de\_la\_div" class="clase\_de\_la\_div" >

<h1>Titulo</h1>

<p>Este es un párrafo...</p>

</div>

## Algunos de los elementos más utilizados

Los elementos HTML son muchos, en el siguiente link, podrás ver una lista de los principales que utilizarás en tu camino del desarrollo:

<http://www.htmlquick.com/es/reference/tags.html>

CSS

## ¿Qué es CSS?

Cascading Style Sheets (CSS), traducido literalmente al español, como Hojas de estilo en cascada, es un lenguaje para especificar cómo los documentos se presentan a los usuarios. En nuestro caso, los documentos son los sitios en HTML que, principalmente, a través de las clases, reciben diferentes estilos.

El navegador ingresa a un sitio HTML, posteriormente busca las clases que tienen las diferentes etiquetas.

La sintáxis de CSS, es la siguiente

selector { propiedad: declaración; declaraciones};

/\* Ejemplo de un estilo aplicado a una clase \*/

.cajaRoja { background : red; font-size: 20px};

/\* Ejemplo de un estilo aplicado a un id\*/

#elementoUnico { background : blue; font-size: 12px};

### Aplicando estilos a elementos HTML

p {

    text-align: center;

    color: red;

}

Esto aplicará una regla de estilo a todos los elementos p de nuestro HTML.

### Aplicando estilo a clases

A través de la etiqueta class=”nombre” en un elemento HTML, podemos aplicar estilo desde el css.

Ejemplo:

En primer lugar, definimos una clase desde el HTML:

<h1 class="tituloRojo"> Este título será rojo </h1>

Posteriormente, le aplicamos estilo desde el CSS, llamando a la clase creada. Para hacerlo, se escribe el nombre de la clase, comenzando por un punto.

.tituloRojo {

color:red

}

### Aplicando estilo a identificadores

A través de la etiqueta id=”nombre” en un elemento HTML, podemos aplicar estilo desde el css.

Ejemplo:

En primer lugar, definimos un ID desde el HTML.

<h1 id="tituloUnico"> Este título será rojo </h1>

Posteriormente, le aplicamos estilo desde el CSS, llamando al ID creado. Para hacerlo, escribimos el nombre del ID anteponiendo un #

#tituloUnico {

color:red

}

## Principales propiedades CSS

Como verán, CSS tiene palabras reservadas para las propiedades, admitiendo valores propios de cada una. Este es el resumen de las principales propiedades en CSS. A las mismas ya se le incluyen declaraciones a modo de demostración.

### Color

color:red

Este ya lo conocemos. Define el color del texto. El color deseado puede escribirse directamente, en inglés, o puede utilizarse el sistema RGB en hexadecimal. Por ejemplo, este mismo color rojo en RGB sería: #FF0000; Ciertos valores, como el rojo, pueden escribirse de forma abreviada: #F00; Otra forma de declarar el color es mediante la función rgb() que puede parametrizarse en decimal o en tantos por ciento. He aquí un ejemplo con las cuatro formas del color rojo:

color: #ff0000

color: #f00

color: rgb(255,0,0)

color: rgb(100%, 0%, 0%)

### Font-Size

font-size:20px;

También muy conocido: define el tamaño de los caracteres. Además del tamaño definido por el usuario, expresado en cualquiera de las unidades de medida conocidas, puede tener alguno de los siguientes valores absolutos: xx-large x-large large medium small x-small xx-small.

### Font-Family

font-family:Courier;

Indica el nombre (o nombres) del tipo de letra que se va a emplear. En el ejemplo se ha utilizado el tipo "Courier", pero pueden escribirse varios separados por comas, lo que indica al navegador que si no existe en la máquina el primer tipo, utilice el segundo, y si tampoco, el tercero, etc. Por ejemplo: font-family:Courier,Verdana,Arial;

Cuando definas tipos de letra, recuerda que muchos programas y algunos drivers de impresora instalan sus propias fuentes en el sistema sin avisarte, y puede que en tu máquina haya tipos que no son estándar en Windows. No se deben utilizar tipos extraños que difícilmente se encontrarán en las máquinas de los clientes, es decir, el tipo de fuente indicado en la hoja simplemente le indica al navegador qué fuente debe usar, NO se la sirve. En la versión 3 de CSS parece que se podrá indicar al cliente dónde obtener una fuente que no tenga instalada.

### Font-Weight

font-weight:bold;

Aquí se especifica el peso o grosor de la fuente. Pueden emplearse literales como lighter, normal (por defecto) o bold. También se pueden utilizar números entre 100 y 900, escritos de 100 en 100 (no sirve 190, pero sí 200). Los pesos no tienen los mismos resultados en todas las máquinas, ya que depende mucho de la calidad de su pantalla, de su resolución, del navegador que emplea... En cualquier caso, no deben hacerse combinaciones extrañas, como definir un size muy pequeño y darle peso 900, ya que lo único que conseguirás es un borrón perfectamente ilegible.

Font-style

font-style:italic;

Solamente tiene dos posibilidades: normal (por defecto) o italic que es el empleado en el ejemplo que estamos analizando, y que hace que el texto tenga cierta inclinación.

### Line-Height

line-height:30px;

Sirve para establecer la distancia entre líneas consecutivas de un mismo párrafo. Además de en píxels o alguna de las unidades de medida que ya conocemos, puede indicarse con un simple número que viene a indicar, aproximadamente, cuantas anchuras de línea se tomarán como unidad de medida para separar una línea de otra. Así, por ejemplo, si pones 0 la segunda línea se suporpone a la primera.

### Letter-Spacing

letter-spacing:5px;

Establece la separación entre los caracteres de la línea.

### Word-Spacing

word-spacing:5px;

Establece la separación entre las palabras de la línea.

### Text-Decoration

text-decoration:underline;

Con este parámetro, que no influye en el tamaño, puedes acompañar al texto de una delgada línea decorativa que puede estar en tres posiciones distintas, como underline (el típico subrayado debajo de la línea), through (en el centro de la línea -tachado-) o overline (encima del texto). Para que no aparezca se utiliza none (por defecto). El valor none se puede utilizar para eliminar el efecto de subrayado que ponen otros elementos, como ocurre en los links.

### Text-Transform

text-transform:capitalize;

Curioso efecto que provee de cuatro posibilidades: capitalize que convierte en mayúscula la primera letra de cada palabra del texto, como puedes ver en el ejemplo, donde el texto original no es así. uppercase para convertir todas las letras a mayúsculas y lowercase para lo contrario, es decir, convertir todas las letras a minúsculas. El valor por defecto es none, que como ya habrás supuesto, deja el texto tal como está escrito.

### Text-Align

text-align:left;

Alinea el texto según convenga. Con left a la izquierda (por defecto), right a la derecha, center centrado, y justify justificado, es decir, igualando todas las líneas en longitud a izquierda o derecha.

### Text-Indent

text-indent:30px

Produce que la primera línea del párrafo se escriba adentrada (indentada) un cierto espacio hacia la derecha o hacia la izquierda, dependiendo de la alineación activa.

### Borders

Nos permiten cambiar el borde de un element, también podemos definir el grosor y estilo del mismo:

.clase {

border-style: solid;

border-width: 5px;

border-color: red;

}

### Background

Nos permiten cambiar el fondo de un elemento, podemos utilizar nombres de colores, valores hexadecimales, o imágenes

.clase {

background: red ;

}

.otraClase {

background: url(‘direcciondelaimagen’)

}

## Más propiedades

Como verán, las propiedades HTML son palabras dedicadas y hay un gran número.

Puede ver un resumen de las utilizadas principalmentes en la web en W3 Schools

<https://www.w3schools.com/css/default.asp>