**ESTRUCTURA BÁSICA HTML**

<!DOCTYPE html>

<**html**>

<**head**>

<**title**>HTML anidado</**title**>

</**head**>

<**body**>

<**h1**>Laboratoria</**h1**>

<**p**>Código <**em**>que</**em**> <**strong**>transforma</**strong**></**p**>

</**body**>

</**html**>

**ELEMENTOS**

Inician con una etiqueta + contenido + cierre de etiqueta

**Los elementos a nivel de bloque**

**Ejm <p>**

**Los elementos en línea**

**Ejm <em> <strong> <mark> <a>**

1. Encabezados para títulos y subtítulos (tip: headings)
2. Un link
3. Una tabla
4. Una lista
5. Una imagen
6. Una línea divisoria (tip: horizontal rule)
7. Una cita
8. Resaltar un texto (tip: mark)

**Atributos**

**Ejem** href / target / target="\_blank" / title

Los atributos pueden modificar / especializar / ó identificar más a la etiqueta que estamos .

dan información adicional sobre el contenido de un elemento que no queremos que aparezca en el contenido actual.

<**a** href="http://laboratoria.la/" target="\_blank" title="Página de inicio de Laboratoria">Ir a Laboratoria</**a**>

\* agregan características y/o comportamiento a tu elemento HTML.

- aparecen en la **etiqueta de apertura**: un nombre y un valor, separados por un signo de igual (=).

**atributos comunes**  a todos los elementos html  **ejm** style

<**p** style="color:red; background-color: yellow; border: 1px solid black;">Código <**em**>que</**em**> <**strong**>transforma</**strong**></**p**>

Un atributo de estilo puede contener una o más declaraciones.

Cada declaración está compuesta por una**propiedad** (como el color) seguida de dos puntos y un **valor** (como rojo)

Cuando tenemos más de una declaración, cada una debe estar separada por punto y coma (;). Por ejemplo: "color:red; background-color: yellow;".

Hay un montón de aspectos que pueden ser influenciados por el estilo. Por ejemplo, la propiedad displaycontrola si un elemento se muestra como un bloque o en línea, o inclusive no mostrarlo:

<**p**>

Un texto se puede mostrar <**strong**>en línea</**strong**>,

<**strong** style="display: block">como bloque</**strong**>, y

<**strong** style="display: none">no visualizarse</**strong**>.

</**p**>

**atributos específicos** que son asociados a un elemento en particular

<ul> lista desordenada con viñetas

<ol> lista ordenada con números

<li > elementos a listar

## Etiquetas vacías

<**img** src="http://cde.2.trome.pe/ima/0/1/1/8/5/1185397.jpg">

**ESTRUCTURA CSS**

### **CSS Inline**

**Para mejorar el uso de style**

<**h1** style="color: blue;

background-color: yellow;

border: 1px solid black;">¡Hola Mundo!</**h1**>

### **2. CSS en el head**

<!DOCTYPE html>

<**html**>

<**head**>

<**title**>Entendiendo CSS</**title**>

<**style**>

**h1** {

**color**: blue;

**background-color**: yellow;

**border**: 1px solid black;

}

</**style**>

</**head**>

<**body**>

<**h1**>¡Hola Mundo!</**h1**>

</**body**>

</**html**>

### **3. CSS stylesheet externo**

#### Archivo index.html

<!DOCTYPE html>

<**html**>

**<head>**

<**title**>Entendiendo CSS</**title**>

<**link** rel="stylesheet" href="style.css">

**</head>**

<**body**>

<**h1**>Hello World!</**h1**>

</**body**>

</**html**>

#### Archivo style.css

**h1** {

**color**: blue;

**background-color**: yellow;

**border**: 1px solid black;

}

La etiqueta <link>

cuenta con el atributo rel para indicar la relación del documento enlazado con el actual.

Valorstylesheet.

El atributo href

Nombre del archivo style.css.

**Sintaxis css**

* Propiedades: son identificadores que indican a las personas qué característica de estilo (ancho, color de fondo, fuente) queremos cambiar.
* Valores: a cada propiedad se le da un valor, que indica cómo queremos cambiar ésta característica (por ejemplo qué fuente, qué ancho o qué color usar).

Propiedad +valor = declaración

Varias declaraciones= bloque de declaración

un **selector**. Para definir a qué elemento de la página web se debe aplicar el estilo que especifica el bloque de declaración

los bloques de declaraciones emparejados con selectores = reglas css

**selector**

|

**p** { **color**: red; **font-size**: 15px; }

| | | |

**prop** **valor** **prop** **valor**

\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_

| |

**declaraci**ó**n** **declaraci**ó**n**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

|

**bloque** **de** **declaraci**ó**n**

/\* ejemplo de una regla css \*/

**p** {

**color**: red;

**font-size**: 15px;

}

## Selector de elementos (selector de tipos)

Este selector hacer referencia directamente a un tipo de elemento HTML. Es la manera más sencilla para hacer referencia a todos los elementos de un mismo tipo. Veamos el ejemplo:

<p>What color do you like?</p>

<div>I like blue.</div>

<p>I prefer red!</p>

/\* All p elements are red \*/

p {

color: red;

}

/\* All div elements are blue \*/

div {

color: blue;

}

## Selectores de clases

## Clase( atributos)

El selector de clase se forma con un punto, '.', seguido de un nombre de clase. Un nombre de clase puede ser cualquier valor sin espacios usado dentro de un atributo HTML [class](https://developer.mozilla.org/es/docs/Web/HTML/Global_attributes" \l "attr-class). Podemos elegir el nombre que deseemos para la clase. Es conveniente saber que en un mismo documento varios elementos pueden compartir el mismo valor de clase y que un elemento puede tener varios nombres de clases separados por un espacio en blanco. Veamos un rápido ejemplo:

Sea el siguiente código HTML:

<ul>

<li class="first done">Create an HTML document</li>

<li class="second done">Create a CSS style sheet</li>

<li class="third">Link them all together</li>

</ul>

Y un sencillo documento de estilos:

/\* The element with the class "first" is bolded \*/

.first {

font-weight: bold;

}

/\* All the elements with the class "done" are strike through \*/

.done {

text-decoration: line-through;

}

## Selectores ID

## (id atributos)

El selector ID está formdo por una almohadilla (#), seguida del nombre ID de determinado elemento. Cualquier elemento puede tener un único nombre ID fijado con el atributo [id](https://developer.mozilla.org/es/docs/Web/HTML/Global_attributes#attr-id). Podemos usar cualquier nombre de nuestra elección para el ID. Es la forma más efectiva de selecionar un solo elemento.

Veamos un rápido ejemplo — dado el código HTML:

<p id="polite"> — "Good morning."</p>

<p id="rude"> — "Go away!"</p>

Y un sencillo documento de estilos:

#polite {

font-family: cursive;

}

#rude {

font-family: monospace;

text-transform: uppercase;

}

## Selector Universal

selector universal (\*) es el comodín. Nos permite seleccionar todos los elementos de una página, se suele usar en combinación con otros selectores (ver [Combinators](https://developer.mozilla.org/es/docs/Learn/CSS/Introduction_to_CSS/Selectores_simples" \l "Combinators) más adelante).

**Importante**: Cuidado al utilizar el selector universal. Al afectar a todos los elementos, usarlo en grandes páginas web grandes puede afectar al rendimiento, ralentizando el tiempo de carga de las páginas más de lo esperado. No tendremos necesidad de usarlo muy a menudo

hora como ejemplo; dado un código HTML:

<div>

<p>I think the containing box just needed

a <strong>border</strong> or <em>something</em>,

but this is getting <strong>out of hand</strong>!</p>

</div>

Y su correspondiente hoja de estilos:

\* {

padding: 5px;

border: 1px solid black;

background: rgba(255,0,0,0.25)

}

## Combinaciones

En CSS, las combinaciones nos permiten combinar varios selectores juntos, con lo que podremos seleccionar elementos contenidos en otros elementos, o elementos adyacentes a otros. Disponemos de cuatro tipos:

* El **selector descendiente** —  (espacio) — permite seleccionar un elemento anidado en alguna parte dentro de otro elemento (no tiene por qué ser un hijo; puede ser un nieto, por ejemplo)
* El **selector hijo** — > — permite seleccionar un elemento que es hijo directo de otro elemento.
* El **selector hermano** — + — permite seleccionar un elemento que es hermano directo de otro elemento (a la derecha por ejemplo, en el mismo nivel jerárquico).
* El **selector hermano en general** — ~ — permite seleccionar cualquier elemento hermano de otro (por ejemplo en el mismo nivel jerárquico, pero no necesariamente adyacente a él).

Veamos un rápido ejemplo para ver el funcionamiento:

<section>

<h2>Heading 1</h2>

<p>Paragraph 1</p>

<p>Paragraph 2</p>

<div>

<h2>Heading 2</h2>

<p>Paragraph 3</p>

<p>Paragraph 4</p>

</div>

</section>

section p {

color: blue;

}

section > p {

background-color: yellow;

}

h2 + p {

text-transform: uppercase;

}

h2 ~ p {

border: 1px dashed black;

}

Los selectores funcionarán como sigue:

* section p selecciona todos los elementos <p>  — los dos primeros son hijos directos del elemento <section>, y los dos siguientes son nietos del elemento  <section> (ya que también están dentro de <div>). Todo el texto del párrafo será de color azul.
* section > p selecciona solo los dos primeros elementos <p>, que son hijos directos del elemento <section> (pero no los dos siguientes, pues no son hijos directos). Por tanto, solo los dos primeros párrafos tendrán realzado amarillo.
* h2 + p selecciona solo los elementos <p> que están directamente después del elemento <h2> en el mismo nivel jerárquico — en este caso el primer y el tercer párrafo. Así estos tendrán el texto en mayúsculas
* h2 ~ p selecciona cualquier elemento <p> en el mismo nivel jerárquico desde los elementos <h2>— en este caso todos los párrafos. Todos ellos tendrán el borde punteado.

**SELECTORES DE ATRIBUTO**

Los selectores de atributo son un tipo especial de selector que combinará elementos en función de sus [atributos](https://developer.mozilla.org/en-US/docs/Glossary/attribute) y valores de atributo. Su sintaxis genérica consta de corchetes ( []) que contienen un nombre de atributo seguido de una condición opcional para coincidir con el valor del atributo. Los selectores de atributos se pueden dividir en dos categorías dependiendo de la forma en que coincidan con los valores de atributo: selectores de atributo de **presencia y valor** y selectores de atributo de **valor de subcadena** .

https://developer.mozilla.org/en-US/docs/Learn/CSS/Introduction\_to\_CSS/Attribute\_selectors