

# Unidad 3 — HTML y CSS

## Contenidos

HTML .....	1
Elementos y atributos .....	2
Estructura y semántica de una página web .....	2
Texto, listas y tablas .....	3
Links y URLs .....	5
Imágenes .....	5
Formularios .....	5
CSS .....	6
Selectores y reglas .....	6
Selectores .....	7

Esta unidad es una introducción al desarrollo web con dos de los lenguajes que entiende el navegador: HTML y CSS.

## HTML

HTML quiere decir *HyperText Markup Language*: Lenguaje de Marcado de HiperTexto. Cuando se propuso la creación de la *World Wide Web* el protocolo de comunicación propuesto fue HTTP (*HyperText Transfer Protocol*). El concepto de hipertexto se refiere a lo que llamamos comunmente *links*. El usuario navega de una página a otra en la web a través del hipertexto, los links que contienen una referencia a otras páginas y sitios de la web.

HTML es el lenguaje que se estableció para definir la **estructura** y la **semántica** de una página web. El código HTML consiste de distintos elementos arreglados en una estructura de árbol donde todos los elementos tienen una raíz en común, el elemento `<html>`.

Veamos un ejemplo básico.

```
<!DOCTYPE html>
<html lang="es" dir="ltr">
  <head>
    <meta charset="utf-8" />
    <title>Hola HTML</title>
  </head>
  <body>
    <h1>Hola HTML!</h1>
    <p>Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor
    incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud
    exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.</p>
  </body>
</html>
```

Si se presta atención a los niveles de indentación, el elemento `<html>` es el “padre” de los elementos `<head>` y `<body>`. A su vez `<head>` tiene dos hijos, un elemento `<meta>` y un elemento `<title>`. El elemento `<body>` también tiene dos hijos, un encabezado y un párrafo. A esto nos referimos con una estructura de árbol. La primer línea no es un elemento, sino una declaración de que el documento que sigue es un documento de HTML5, el último estándar del lenguaje.

## Elementos y atributos

Los elementos de HTML se ven en el código por sus **etiquetas** o *tags*. La gran mayoría de los elementos llevan dos etiquetas, una de apertura y una de cierre. Las etiquetas comienzan con `<` y terminan con `>`, con el nombre del elemento en el medio. Las etiquetas de cierre comienzan con `</`. Algunos elementos no llevan etiquetas de cierre porque no tienen texto adentro. Por ejemplo el elemento `<meta />` o el elemento `<img />`.

Los elementos a menudo tienen atributos, que aparecen en la etiqueta de apertura. Por ejemplo el elemento `` tiene un atributo `src` que es igual a una URL. Los atributos siempre se especifican con el nombre, un igual y su valor entre comillas dobles. Distintos atributos aplican a distintos elementos, pero hay dos atributos importantes que se pueden usar en cualquier elemento que son `id` y `class`.

Los atributos de un elemento siempre forman pares de nombre y valor, con el valor del atributo encerrado en comillas después del signo igual.

## Estructura y semántica de una página web

La disposición de los elementos HTML en el código le dan la estructura a la página web. En el nivel más general, todo lo que está adentro del elemento `<body>` es lo que se ve en la ventana del navegador. Los elementos a menudo están anidados unos dentro de otros, marcando una jerarquía entre ellos.

El elemento `<head>` que aparece antes de `<body>` no contiene elementos que se muestran en la ventana del navegador, sino que contiene información acerca de la página web.

La semántica de una web se refiere al significado del texto y otros medios de comunicación que

contiene. Los elementos de HTML dotan de significado a estos medios. Se dice que HTML es un lenguaje de marcado (*markup language*) porque justamente las etiquetas HTML de alguna manera indican o marcan el significado de distintas porciones de texto, este texto es un encabezado, este otro texto es un párrafo.

## Texto, listas y tablas

En esta sección hacemos una revisión de los elementos más comunes para trabajar con texto.

### Encabezados y párrafos

Los encabezados (*headings*) vienen en seis tamaños distintos desde `<h1>` hasta `<h6>`. Indican que el texto que encierran son títulos y subtítulos.

```
<h1>Título</h1>
<h2>Subtítulo</h2>
<h3>Subsubtítulo</h3>
```

El elemento `<p>` indica que el texto es un párrafo. Por defecto el navegador muestra cada párrafo separado por una línea en blanco.

```
<p><b>Lorem ipsum</b> dolor sit amet, consectetur adipisicing elit, sed do eiusmod
tempor incididunt ut labore et <i>dolore magna aliqua</i>. Ut enim ad minim veniam,
quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.</p>

<p> Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu
fugiat <u>nulla pariatur</u>. Excepteur sint occaecat cupidatat non proident, sunt in
culpa qui officia deserunt mollit anim id est laborum.</p>
```

Dentro de un párrafo podemos usar `<b>` para texto en **negrita**, `<i>` para texto en *cursiva*, y `<u>` para texto subrayado. El elemento `<i>` hoy en día es usado para insertar un ícono con fuentes como Font Awesome también. Todos estos elementos a diferencia de los encabezados y párrafos que son elementos en **bloque**, son **en línea**. Es decir, que no empiezan una nueva línea.

Otros elementos que no empiezan una nueva línea y que además no tienen etiqueta de cierre son `<br />` (*break*) que es el salto de línea, y `<hr />` (*horizontal rule*) una línea o regla horizontal.

### Listas

Hay tres tipos de listas en HTML, ordenadas, no ordenadas y listas de definiciones. Las listas consisten de dos elementos, la lista en sí y una cantidad de ítems en cada lista (*list item*) marcados con el elemento `<li>` en las listas ordenadas y no ordenadas. En cambio en las listas de definiciones el término que se va a definir se marca con `<dt>` y la definición con `<dd>`.

Ejemplos de listas: no ordenada primero, ordenada y por último lista de definiciones. En la primer lista hay una sublista también.

```

<ul>
  <li>Item 1</li>
  <li>Item 2</li>
  <li>Item 3
    <ul>
      <li>Subitem 1</li>
      <li>Subitem 2</li>
    </ul>
  </li>
</ul>

<ol>
  <li>Item 1</li>
  <li>Item 2</li>
  <li>Item 3</li>
</ol>

<dl>
  <dt>HTML</dt>
  <dd>Es un lenguaje de marcado, <i>HyperText Markup Language</i></dd>
</dl>

```

## Tablas

Las tablas en HTML se definen dentro del elemento `<table>`. En el elemento tabla podemos definir de manera opcional un encabezado y un cuerpo: `<thead>` y `<tbody>`. Cada fila de la tabla debe definirse en un `<tr>` (*table row*), y cada celda dentro de esa fila con un `<td>` (*table data*). Este último se cambia por un `<th>` (*table header*) cuando estamos escribiendo el encabezado de una columna.

```

<table>
  <thead>
    <th>Alumno</th>
    <th>Nota</th>
  </thead>
  <tbody>
    <tr>
      <td>Juan</td>
      <td>7</td>
    </tr>
    <tr>
      <td>María</td>
      <td>8</td>
    </tr>
  </tbody>
</table>

```

# Links y URLs

Una URL de *Uniform Resource Locator* es una cadena de caracteres que identifica de manera única un recurso en la web. Por recurso entendemos generalmente un archivo (HTML, PNG, GIF, CSS, etc.). Por ejemplo la URL `./img/conocimiento.jpg` hace referencia a una imagen de esta página web.

La etiqueta `<a>` de *anchor* (ancla) nos permite tener vínculos desde nuestra página a otros documentos y recursos.

```
<h1 id="top">Links</h1>
<a href="./img/conocimiento.jpg">Una imagen</a>
<a href="https://www.google.com.ar" target="_blank">Un link a Google</a>
<a href="mailto:ejemplo@ejemplo.com">Contactar</a>
<a href="#top">Ir arriba</a>
```

Las URL pueden ser relativas, como en el primer ejemplo que la dirección es relativa a la ubicación de la página desde donde se linkea. O también pueden ser absolutas como en el segundo ejemplo que simplemente es un link a la página de Google. Para abrir el link en una nueva pestaña podemos hacer uso del atributo `target`.

Podemos tener links para enviar un mail a una dirección especificada como URL con `mailto:.` También podemos linkear dentro de la misma página usando el atributo `id` de distintos elementos, generalmente encabezados, anteponiendo el símbolo `#` en el atributo `href` del link.

## Imágenes

El elemento `<img />` sirve para cargar imágenes en una página. Los dos atributos más importantes de este elemento son `src` (*source*), la URL de la imagen y `alt` un texto alternativo para los lectores de pantalla o por si la imagen no carga.

```

```

Se recomienda usar JPG para fotografías y GIF o PNG para ilustraciones o imágenes con pocos colores diferentes.

## Formularios

Un formulario es un grupo de elementos típicos de una interfaz de usuario con un mismo propósito, como enviar una petición para registrarse en un sitio, o sacar un turno para el médico. Los elementos típicos de un formulario son similares a los de un formulario en papel, con la salvedad de que algunas cosas que se pueden hacer en la computadora como el elemento `<select>` no se pueden en papel.

```

<form>
  <label for="nombre">Nombre</label>
  <input type="text" name="nombre"> <br />
  <label for="email">Correo electrónico</label>
  <input type="text" name="email"> <br />
  <input type="radio" name="radio">Masculino <br />
  <input type="radio" name="radio">Femenino <br />
  <select name="select">
    <option>Profesor</option>
    <option selected>Alumno</option>
  </select> <br />
  <input type="submit" name="" value="Enviar">
</form>

```

Generalmente, aunque no necesariamente, los formularios web tienen un botón al final que envía los datos del formulario a un servidor para ser procesados. El elemento `<input>` usa el atributo `type` para definir que tipo de entrada se desea: un botón, un campo de texto, un *checkbox*, etc. El elemento `<label>` (etiqueta) sirve para dar una descripción de los elementos de entrada.

## CSS

CSS por *Cascading Style Sheets* es el lenguaje de diseño de la web. Su propósito es doble, por un lado le da estilo a los sitios web manejando aspectos como el color y la tipografía. Por otro lado define el *layout*, la disposición de los elementos HTML en una página controlando aspectos como posición, márgenes, relleno, etc.

## Selectores y reglas

Un archivo CSS es un conjunto de reglas de estilo. Cada regla comienza con un selector (o varios) y contiene dentro una serie de estilos que se le aplican a ese selector. Los estilos vienen especificados como pares de **propiedades** y **valores**.

En el siguiente ejemplo hay una regla que selecciona todos los `<h3>` y les aplica cuatro estilos: fuente, subrayado, alineación al centro y un color.

```

h3 {
  font-family: 'Times New Roman', serif;
  color: #323232;
  text-decoration: underline;
  text-align: center;
}

```

Una manera de aplicar reglas de estilo a un archivo HTML es usando la etiqueta `<style>` como en el siguiente ejemplo.

```

<!DOCTYPE html>
<html lang="es" dir="ltr">
  <head>
    <meta charset="utf-8">
    <title>Ejemplo</title>
    <style>
      h1 {
        color: red;
        font-size: 24px;
      }
    </style>
  </head>
  <body>
    <h1>CSS dentro del HTML</h1>
    <p>Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor
    incididunt ut labore et dolore magna aliqua.</p>
  </body>
</html>

```

Generalmente se prefiere usar un archivo separado para el estilo de una página, así es más fácil reutilizar un diseño en distintos sitios o en las distintas páginas de un mismo sitio. Para enlazar una hoja de estilos a un archivo HTML se utiliza la etiqueta `<link>` en el `<head>` de la página.

```

<!DOCTYPE html>
<html lang="es" dir="ltr">
  <head>
    <meta charset="utf-8">
    <title>Ejemplo</title>
    <link rel="stylesheet" href="style.css">
  </head>
  <body>
    <h1>CSS dentro del HTML</h1>
    <p>Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor
    incididunt ut labore et dolore magna aliqua.</p>
  </body>
</html>

```

## Selectores

Los selectores disponibles son varios, pero los más usados son:

- `*` el selector universal.
- `h1`, `h2`, `p` selectores para cada elemento HTML.
- `.nav` selector de clase, comienzan con punto.
- `#.btn-1` selector de ID, comienzan con numeral.
- `p a` selector de descendientes, en este caso elige los elementos *anchor* dentro de párrafos.

Las reglas aplicadas a cada elemento de la página se definen por la **especificidad** de cada selector. Muchas veces hay varias reglas que aplican a los mismos elementos. La regla que gana es la más específica.

Si dos selectores son idénticos, como en `p { color:red; }` y `p { color:blue; }` la última regla definida en el orden del archivo es la que se aplica. Pero si no son idénticos gana el más específico. Las reglas que seleccionan por ID siempre son más específicas que cualquier otra regla. Las reglas que seleccionan por clase son más específicas que las que seleccionan elementos de la página como `h1`.

Algunas propiedades de los elementos pueden ser **heredadas** de un elemento a sus descendientes. Por ejemplo la tipografía es heredada. Por lo tanto si definimos `body { font-family: serif; }` esta regla va a ser aplicada a todos los elementos con texto de la página salvo que digamos lo contrario.