

Desarrollo Web Para Principiantes

Introducción

El propósito de este documento es orientar mediante ejemplos y consejos que me han servido para adentrarme poco a poco en el mundo de las páginas web. Se iniciará con lo necesario para poder programar en HTML, pasando por CSS y JavaScript. Además de otras herramientas como NodeJS, Nginx, etc.

HTML es un lenguaje de programación basado en etiquetas usadas para declarar lo que será mostrado al usuario en el navegador. HTML no permite definir cómo será mostrado, es meramente el esqueleto de las páginas web. Para poder estilizar y dar funcionalidad a los elementos, serán necesarios otros lenguajes de programación como CSS para estilizar y JavaScript para agregar funcionalidad a la página.

Si se desea estudiar más en profundidad lo visto durante la guía, será posible descargar desde el Github todos los archivos y pdf's recomendados para iniciar.

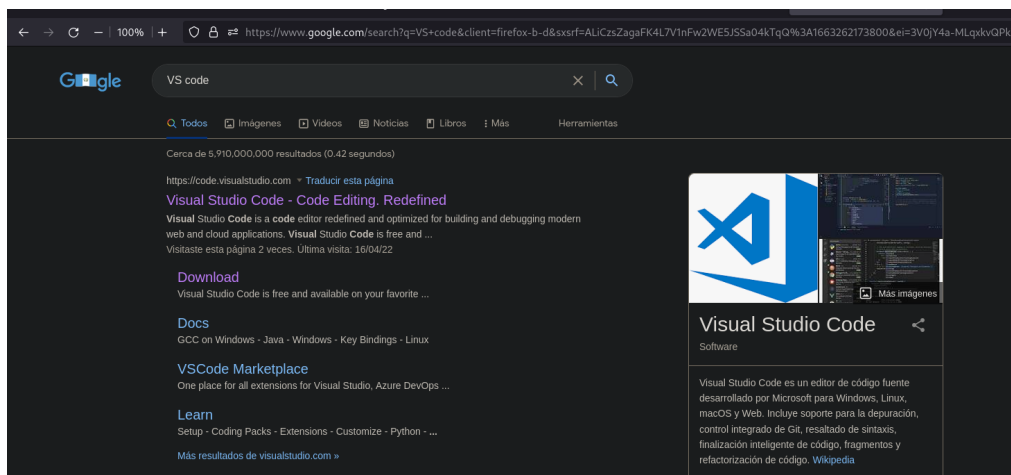
Capítulo 1

Elegir un entorno de desarrollo

Para este curso será necesaria una herramienta para poder programar de forma comoda y eficiente. Se puede usar algo sencillo y ligero como el bloc de notas, pero durante la guía se usará VS Code, debido a la cantidad de extensiones y herramientas que ofrece a la hora de programar.

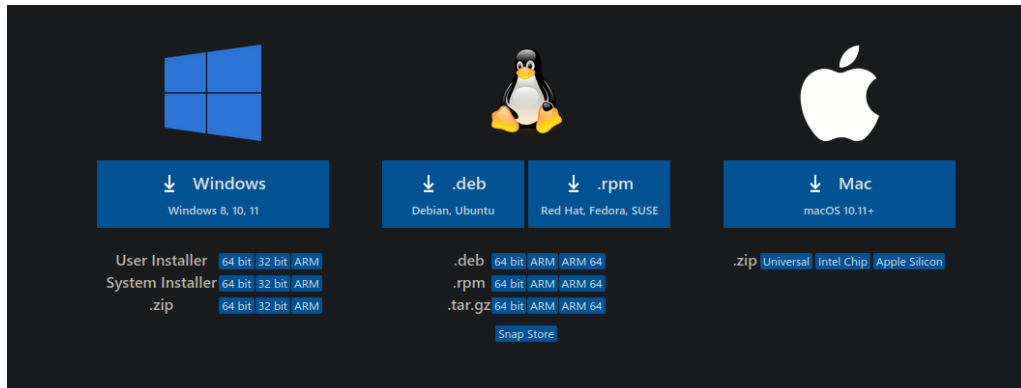
Paso 1:

- Ve al navegador de tu elección y busca VS Code o Visual Studio Code
- Ingresa a la página web



Paso 2:

- Según tu sistema operativo (Windows, Mac o Linux), selecciona el archivo para instalar, descarga e instalalo.

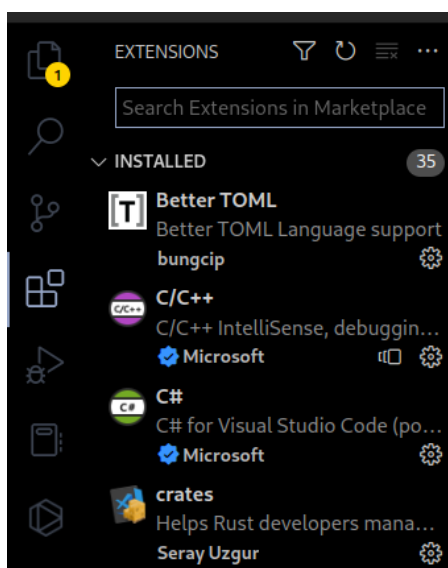


Paso 3:

- Ya instalado vamos a crear una carpeta donde iremos guardando los archivos durante toda la guía. No importa la ubicación de dicha carpeta.

Paso 4:

- Abrir VS Code
- Ir a File (Archivo)
- Open Folder (Abrir carpeta) y se selecciona la carpeta creada anteriormente



Etiquetas básicas en HTML5

DOCTYPE: Esta etiqueta no es completamente necesaria, pero como buena práctica coloca al inicio del documento html, para así hacerle saber al navegador que tipo de archivo estará leyendo.

TITLE: Permite modificar el texto del título de la pestaña del navegador.

HEAD: Etiqueta que contiene información que no será mostrada en pantalla, pero que será necesaria para el funcionamiento de la página web. Desde importación de elementos de terceros, hasta importar archivos css y javascript.

HEADER: Etiqueta que permite separar los menús principales del resto del cuerpo de la página web.

BODY: Esta etiqueta engloba a la mayoría de elementos en la página.

FOOTER: Etiqueta utilizada más como buena práctica, es usada para colocar información de contacto, direcciones físicas, etc.

P: Etiqueta para añadir párrafos completos al html de la página. (Es posible agregar párrafos con otras etiquetas, pero como buena práctica se recomienda usar la etiqueta **p**).

B: Texto en formato **bold** o **negrita**.

H1 hasta H6: Son quizás algunas de las etiquetas más utilizadas, ya que son las mejores para títulos, subtítulos y textos puntuales que no se repetirán.

BR: Etiqueta que permite agregar líneas vacías al html.

I: Etiqueta de texto en formato *cursiva*.

SUMMARY: Etiqueta utilizada en el texto principal de la etiqueta **Details**.

DETAILS: Etiqueta que permite “esconder” texto en un menú desplegable (Ver ejemplos más adelante)

STRIKE / DEL: Etiqueta que permite “tachar” un texto. (ejemplo: ~~texto tachado~~)

SMALL: Texto en un menor tamaño. (ejemplo: texto más pequeño)

SECTION: Etiqueta que permite seccionar los párrafos en varias partes. (No afecta visualmente, pero es una buena práctica).

SUP y SUB: Permite colocar texto arriba y debajo de otros elementos. (Ejemplo: $hola^{exponente}$ y $base_{base}$)

ASIDE: Etiqueta para indicarle al navegador que es un texto secundario (No afecta visualmente, más es una buena práctica).

ARTICLE: Etiqueta mayormente utilizada como una buena práctica ya que funciona similar a la etiqueta **P**, ya que permite añadir grandes textos. (Es usada para hacerle saber que el texto es solamente una pequeña parte del contenido de los párrafos).

NAV: Etiqueta utilizada para englobar grandes bloques de links (No afecta visualmente, es usada como buena práctica).

UL: Lista sin un orden específico. Un buen ejemplo sería una lista con viñetas.

OL: Lista ordenada, lista que tiene un orden específico, mediante distintos elementos.

LI: Etiqueta para agregar elementos a la lista.

A: Etiqueta utilizada para crear elementos con links.

IMG: Etiqueta para poder crear elementos de tipo imagen. Será posible hacer ciertas modificaciones mediante el atributo width (se recomienda usar porcentajes para poder mantener las proporciones). Y el atributo alt para indicar que debería ir en caso de que la imagen no pueda ser desplegada.

VIDEO: Etiqueta para añadir elementos multimedia de tipo vídeo. Será posible modificar elementos mediante los atributos controls, loop y autoplay.

AUDIO: Etiqueta multimedia de tipo audio. Permite modificar elementos mediante los atributos autoplay, controls y loop.

TABLE: Etiqueta que permite crear un elemento de tipo tabla (ejemplo: tablas de Excel).

TR: Etiqueta para generar filas en las tablas. (Table row)

TH: Etiqueta para generar los encabezados de las columnas en las tablas. (Table header)

TD: Etiqueta para añadir elementos a las tablas. (Table data).

CAPTION: Etiqueta usada para añadir títulos a las tablas.

DIV: Etiqueta usada para englobar secciones de código (No afecta visualmente, es usada como una buena práctica, para mantener lo más organizado el código).

LABEL: Etiqueta usada como buena práctica al lado de elementos de formularios.

FORM: Etiqueta usada para englobar y crear elementos de input en la página web. Existen varios atributos que le indican al navegador que tipo de input se realizará.

- **Type text:** Cuadros de inputs vacíos para textos simples
- **Type password:** Le indica al navegador que se ingresará una contraseña, lo que sustituye los inputs por asteriscos.

- **Type number:** Indica que se deberán ingresar números solamente.
- **Type email:** Indica que deberá ingresar un texto en formato email (hola@algo.algo).
- **Type Color:** Crea un recuadro que permite elegir un color.
- **Type range:** Crea una barra para deslizar en vez de ingresar el número directamente.
- **Type date:** Crea elementos que permiten seleccionar una fecha.
- **Type time:** Crea un recuadro que permite seleccionar una hora
- **Type textarea:** Permite generar un recuadro para ingresar textos grandes.
 - Además, se puede crear el recuadro con un tamaño ya establecido con los atributos row y cols

- **Type radio:** Permite crear un menú del tipo rellenar un círculo. Mediante los atributos name para agrupar y value para definir la opción dentro del menú.
- **Type checkbox:** Permite producir un menú de opciones que permite seleccionar más de un elemento. Las opciones se generarán con los atributos name y value.
- **Type button:** Elemento que genera un botón para crear un botón e interactuar con el formulario.
- **Type submit:** Elemento que genera un botón para enviar los resultados del formulario.

SELECT: Etiqueta que creará un menú de tipo acordeon.

OPTION: Etiqueta para añadir elementos al menú en la etiqueta **SELECT**.

<!-- -->: Etiqueta que permite “comentar” una sección de código.

PROGRESS: Permite mostrar una barra de “progreso” o carga.

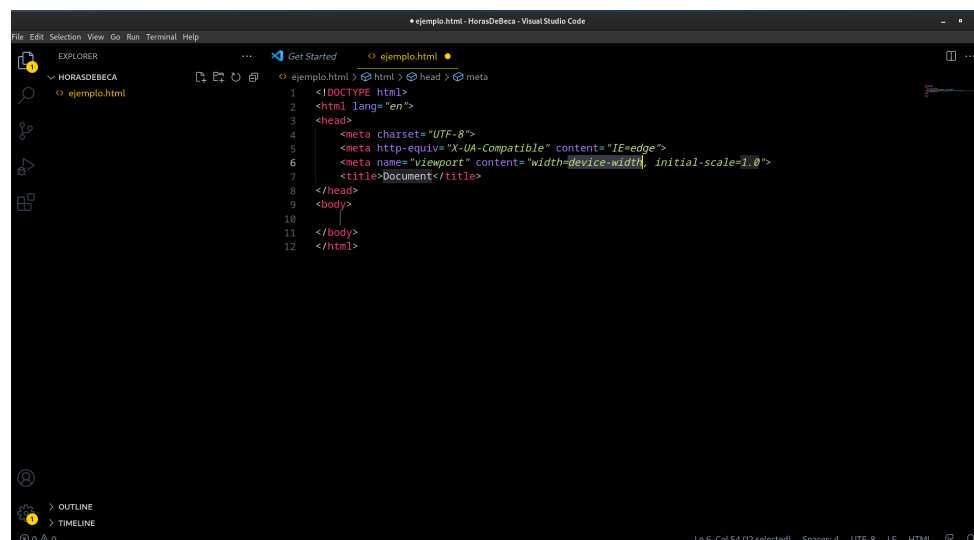
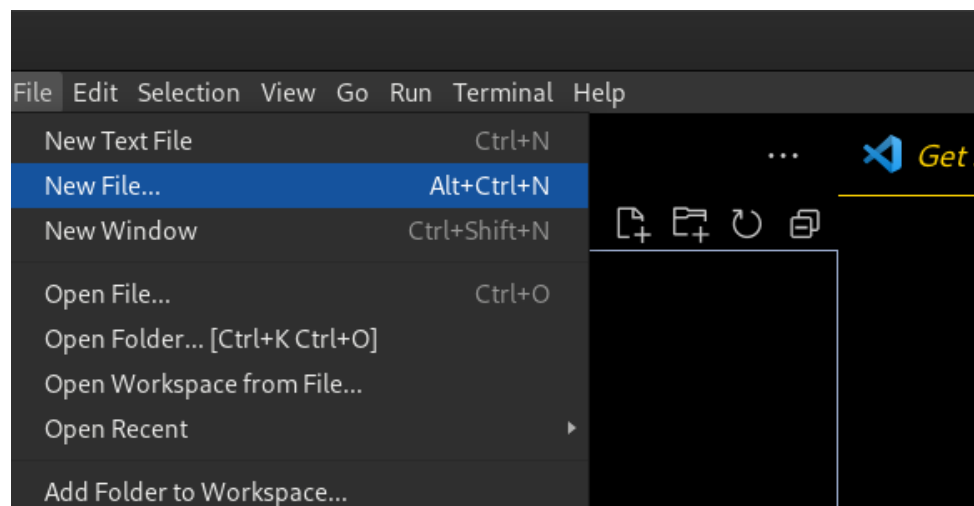
Capítulo 2

A lo largo de este capítulo iremos creando diversos mini-proyectos escritos solamente en html. Comenzando con una pequeña historia interactiva, pasando por una página web de una tienda y por último un blog, siempre haciendo uso de varias de las etiquetas vistas en el capítulo anterior.

Inicio del proyecto

Para iniciar, crearemos el tan conocido “Hola Mundo!”. Por lo que debemos crear un archivo, los pasos a seguir son:

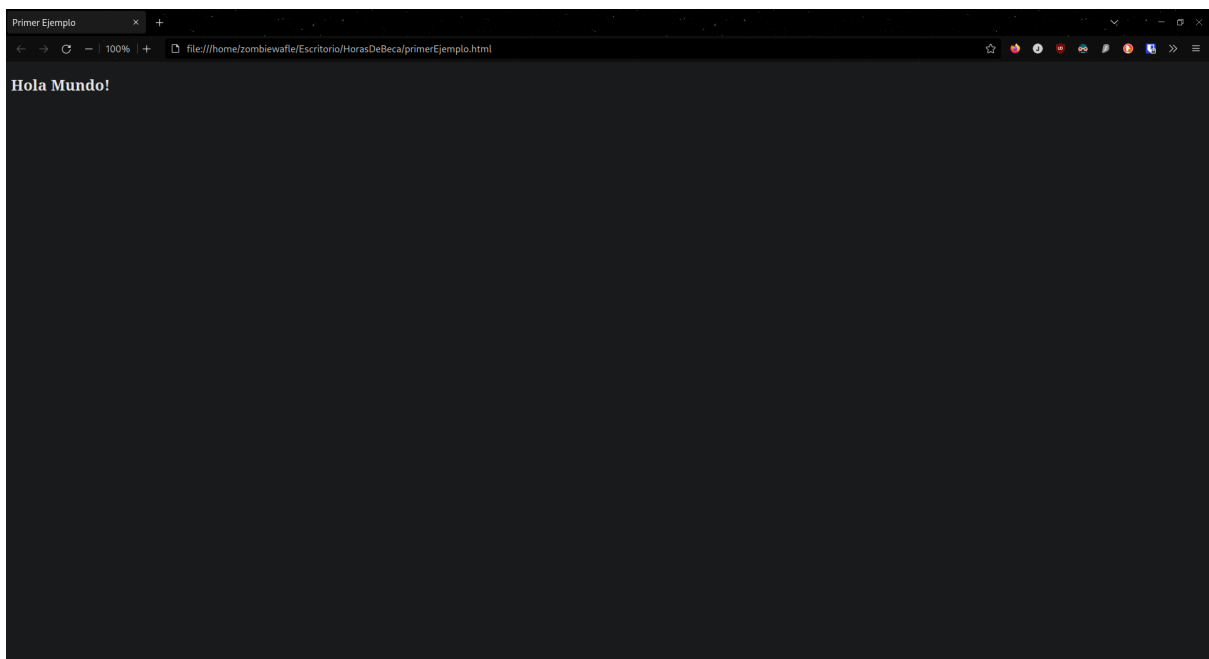
- Ir a crear un archivo desde **Archivo** (File)
- Ir a **Nuevo Archivo** (New File)
- Se ingresa el nombre del documento
- Para generar la estructura básica de un html podemos escribir html y seleccionar la opción html:5



Ya generada la estructura básica, ha llegado el momento de crear nuestra primera página web en HTML5 (solamente con HTML). Para este ejemplo solamente tendremos una etiqueta H2 con el texto “¡Hola Mundo!”.

Dentro de la etiqueta **body** manejaremos la mayor parte del tiempo en este primer proyecto. Esto solamente para seguir las buenas prácticas de la programación de páginas web. Además de que usaremos la etiqueta **title** para indicar el título en la pestaña del navegador.

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta http-equiv="X-UA-Compatible" content="IE=edge">
6      <meta name="viewport" content="width=device-width, initial-scale=1.0">
7      <title>Primer Ejemplo</title>
8  </head>
9  <body>
10     <h2>Hola Mundo!</h2>
11
12 </body>
13 </html>
```



Historia Interactiva

Esta vez vamos a crear una historia de fantasía (el tema queda a discreción de cada uno), en donde se tengan múltiples opciones. Dependiendo de la situación se deberán crear varios caminos y finales.

Nota:

Para este ejemplo haremos uso de las etiquetas `<p>`, `<a>` y del atributo `href` para poder redirigir a los distintos archivos en el proyecto.

Primero necesitamos generar la estructura básica de la primera sección de la página.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Historia Interactiva No.1</title>
</head>
<body>

</body>
</html>
```

Para facilitar la comprensión será necesaria una página de inicio, con instrucciones básicas de cómo moverse a lo largo de la historia.

```
start.html index.html x
index.html > ...
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Historia Interactiva No.1</title>
8 </head>
9 <body>
10
11   <h2>¡Bienvenidos</h2>
12   <p>A continuación encontrarás una historia interactiva, la cual cambiará basandose en tus decisiones. Algunas te llevaran al mismo punto,
13     además de que siempre podrás regresar. Este es solamente un ejemplo basado en los videojuegos basados en texto, por lo que si deseas
14     seguir expandiendo la historia, siempre podrás acceder al código fuente e ir agregando contenido.
15   </p>
16   ¡Espero que lo disfrutes y suerte!
17 </body>
18 </html>
```

Para mantener un orden se irá construyendo la historia en distintos archivos. Empezando siempre con el archivo `index.html`. Será muy importante saber nombrar los archivos que vayamos creando, para facilitar el trabajo, además de hacer más legible nuestro código.

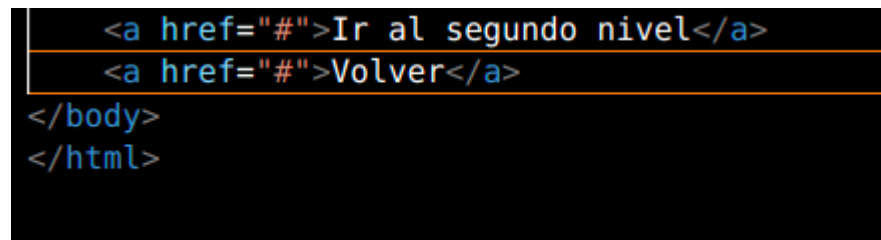
Nota: Se aconseja usar una estructura de nombrado de archivos del tipo CamelCase, ya que facilita su lectura, pero esto queda a discreción de cada uno.



```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4 <meta charset="UTF-8">
5 <meta http-equiv="X-UA-Compatible" content="IE=edge">
6 <meta name="viewport" content="width=device-width, initial-scale=1.0">
7 <title>Entrada</title>
8 </head>
9 <body>
10
11 <p>Has avanzado por el estrecho pasadizo, avanzas como puedes y llegas a la parte trasera de la casa. ¿Ahora que harás? ¿Intentarás entrar a
12
13 <a href="#">Romper la puerta corrediza y entrar a la casa</a>
14 <a href="#">Volver</a>
15
16 </body>
17 </html>
```

Para facilitar el trabajo más adelante, solamente iremos agregando el símbolo “#” en el atributo **href**, esto para que el navegador reconozca que es un link, pero sin funcionar realmente.

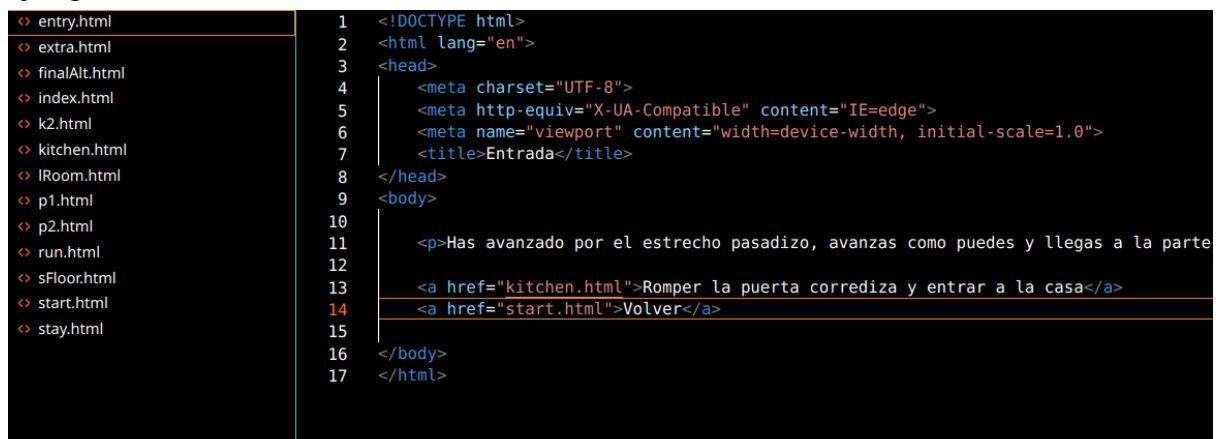
Ejemplo:



```
<a href="#">Ir al segundo nivel</a>
<a href="#">Volver</a>
</body>
</html>
```

Para poder redirigir a diversos archivos html. Será necesario agregar el nombre y dirección de dicho archivo. Repetir hasta haber completado la historia.

Ejemplo:



```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4 <meta charset="UTF-8">
5 <meta http-equiv="X-UA-Compatible" content="IE=edge">
6 <meta name="viewport" content="width=device-width, initial-scale=1.0">
7 <title>Entrada</title>
8 </head>
9 <body>
10
11 <p>Has avanzado por el estrecho pasadizo, avanzas como puedes y llegas a la parte
12
13 <a href="kitchen.html">Romper la puerta corrediza y entrar a la casa</a>
14 <a href="start.html">Volver</a>
15
16 </body>
17 </html>
```

Capítulo 3

Para este capítulo veremos cómo estilizar un HTML de manera básica, mediante el uso de CSS. Durante el capítulo iremos repasando paso a paso y con una descripción de los atributos más recurrentes en la mayoría de páginas web de hoy en día. Además de una breve descripción de lo que es el lenguaje CSS y sus derivados.

En el caso de CSS, los elementos que deseamos modificar serán seleccionados mediante su ID, clase, o etiqueta. Podremos acceder a ellos diversas maneras, en el caso de los ID que hayamos colocado en el html, será mediante el # seguido del ID. Las clases mediante un punto seguido del nombre de la clase (.claseEjemplo).

En CSS y derivados se programa en un formato de cascada, ya que conforme vamos bajando, se podrán sobrescribir los cambios que vayamos haciendo. Por ejemplo:

```
# general.css > ...
1  body{
2  |    background-color: □black;
3  |  }
4
5  body{
6  |    background-color: ■white;
7  |  }
8
9  |
```

El navegador tomará en cuenta solamente el último cambio que hayamos hecho. En este caso sería cambiar el color del fondo a blanco.

Atributos Básicos en CSS

COLOR: Permite cambiar el color de un texto.

LINE-HEIGHT: Permite cambiar la distancia entre las líneas.

LETTER-SPACING: Permite cambiar el espaciado entre caracteres.

TEXT-ALIGN: Permite posicionar el texto en distintos puntos dentro de su contenedor.

TEXT-DECORATION: Permite cambiar la decoración del texto, como subrayados, viñetas, etc.

TEXT-INDENT: Permite alterar la sangría o indentado del primer carácter de un texto.

TEXT-TRANSFORM: Permite modificar cosas como todo en mayúsculas, minúsculas, primera en mayúsculas o sin ningún tipo de alteraciones.

LIST-STYLE: Altera el estilo de las listas.

LIST-STYLE-IMAGE: Permite convertir una imagen en un elemento de la lista.

LIST-STYLE-POSITION: Permite cambiar la posición de los elementos de la lista.

LIST-STYLE-TYPE: Permite cambiar el estilo de la lista.

BORDER: Permite alterar elementos de los bordes como color, estilo, etc.

BORDER-BOTTOM: Permite cambiar las propiedades del borde inferior del contenedor.

BORDER-TOP: Permite cambiar las propiedades del borde superior del contenedor.

BORDER-LEFT: Permite cambiar las propiedades del borde izquierdo del contenedor.

BORDER-RIGHT: Permite cambiar las propiedades del borde derecho del contenedor.

BORDER-COLOR: Permite cambiar el color del borde de todos los lados del contenedor.

BORDER-LEFT-COLOR: Permite cambiar el color del borde izquierdo.

BORDER-RIGHT-COLOR: Permite cambiar el color del borde derecho.

BORDER-TOP-COLOR: Permite cambiar el color del borde superior.

BORDER-BOTTOM-COLOR: Permite cambiar el color del borde inferior.

BORDER-BOTTOM-STYLE: Permite cambiar el estilo del borde inferior.

BORDER-BOTTOM-WIDTH: Permite cambiar el grosor del borde inferior.

BORDER-TOP-STYLE: Permite cambiar el estilo del borde superior.

BORDER-TOP-WIDTH: Permite cambiar el grosor del borde superior.

BORDER-LEFT-STYLE: Permite cambiar el estilo del borde izquierdo.

BORDER-LEFT-WIDTH: Permite cambiar el grosor del borde izquierdo.

BORDER-RIGHT-STYLE: Permite cambiar el estilo del borde derecho.

BORDER-RIGHT-WIDTH: Permite cambiar el grosor del borde derecho.

BORDER-STYLE: Permite cambiar el estilo del borde, en todos sus lados.

FONT: Permite cambiar el estilo de la letra, desde su grosor, estilo, tipografía, etc.

FONT-FAMILY: Permite crear o cambiar la tipografía.

FONT-SIZE: Permite especificar el tamaño de la letra.

FONT-STYLE: Permite especificar el estilo de la letra, ya sea normal, itálica, negrita, etc.

FONT-VARIANT: Permite especificar si el texto deberá ser mostrado en minúsculas o no.

FONT-WEIGHT: Permite especificar el grosor de la letra, desde normal, bold (negrita), lighter (delgada), etc.