

«Talento Tech»

Desarrollo Web 1

Clase 03





Clase N° 3 | Conceptos básicos

Temario:

- Estructura de CSS
- Implementación de CSS
- Propiedades de texto y fondos
- Selectores CSS
- Características de “box sizing”

¿Qué es CSS?

CSS (Cascading Style Sheets) es un lenguaje de estilo utilizado para describir la presentación y el diseño de un documento **HTML o XML**. Se utiliza para definir cómo se ven los elementos y componentes de una página web, incluyendo el color, la tipografía, el tamaño, el espaciado, el diseño y otros aspectos visuales. El primer paso en este curso fue completar el esqueleto del sitio web. Ahora, llega el momento de darle color a todo esto. El primer paso es conocer la estructura de **CSS**, cómo se escribe. Luego, veremos cómo incorporarlo al esqueleto.

Implementando CSS



Existen varias maneras de relacionar, entrelazar, juntar, linkear el esqueleto del sitio con su cuerpo, el **HTML** con el **CSS**, y varias de estas maneras son incorrectas. Aun así es importante conocerlas ya que puede suceder que te topes con algún sitio web que aún las utilice o que al momento de trabajar sobre otro proyecto ya existente tenga estas metodologías. En este apartado veremos las formas correctas e incorrectas de implementar **CSS**.

CSS

CSS es un lenguaje que permite darle estilo a nuestro sitio. Cuando decimos estilo nos referimos a poder darle colores, tipos de letras, cambiar el tamaño de letras, imágenes y todo elemento que se nos ocurra, generar animaciones, etc, etc, etc.

A esta altura ya sabés que para crear un título principal, usás la etiqueta <h1> en el html y si quisieras que esa etiqueta tuviera un color, tendrías que agregarlo por medio de CSS.

La pregunta es ¿Cómo lo implemento?

Existen tres maneras de implementar CSS al sitio web, dos de las cuales son incorrectas. Aunque de estas tres opciones una sola sea la adecuada, es importante que conozcas las tres.

1. Estilos en línea
2. Etiqueta Style
3. Etiqueta link

Vamos a ver estas diferentes formas de implementar CSS con ejemplos.

Estilo en línea

La primera forma se llama “**inline styles**” o “**estilos en línea**”. En esta forma lo que hacemos es agregar un atributo “style” a la etiqueta y dentro de este atributo, las propiedades CSS

```
<h1 style="background-color: red;">Titulo Principal</h1>
```

El problema con esta metodología es que el código CSS se ve mezclado con el HTML haciendo su lectura muy compleja y dificultando futuras modificaciones que se quieran realizar.

✗Este método no es recomendado.

Etiqueta Style

¿Te acordás de la etiqueta **<head>**?. Dentro de la etiqueta **<html>** había dos grandes etiquetas, la de **<head>** y la de **<body>**. En la de **<body>** va información que se va a ver en la web y en la de **<head>** van metadatos para el navegador.

Existe una etiqueta que se llama **<style>** en donde podemos agregar estilos **CSS**. Esta etiqueta va dentro de la etiqueta **<head>** y queda de la siguiente manera:

```
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width,
initial-scale=1.0">
  <title>Document</title>
  <style>
    h1{
      background-color: gray;
    }
  </style>
</head>
```

En el primer ejemplo vimos el uso de CSS en línea y el principal problema era la mezcla entre códigos dificultando su lectura.

En este segundo caso se separa código HTML y CSS, ubicando las propiedades de CSS dentro del **<head>** del HTML.

Aún así, este método tampoco es recomendado ya que quedan grandes columnas de código. Ambas estructuras son igual de importantes y es preciso trabajar con ellas al mismo tiempo. Este método tampoco resulta poco práctico en el uso cotidiano.

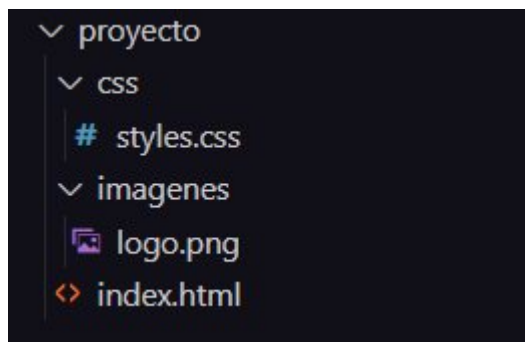
✗Este método no es recomendado.

Etiqueta Link

La etiqueta **<link>** es un elemento utilizado en la sección **<head>** de un documento HTML para enlazar o referenciar recursos externos, como archivos CSS, fuentes, iconos, archivos JavaScript, entre otros. Su función principal es establecer una conexión entre el documento HTML y los recursos externos necesarios para su correcta presentación y funcionalidad.

Al tratarse de un archivo externo, en este caso, un archivo .css, debemos crearlo y ubicarlo del directorio o carpeta principal, como vemos en el siguiente gráfico.

Este método si es recomendado.



Al tratarse de un archivo externo, en este caso un archivo .css, debemos crearlo y ubicarlo del directorio o carpeta principal, como vemos en el siguiente gráfico.
Esto podemos hacerlo desde el VS Code.

Así se vería la etiqueta link en el head (la ruta depende de la ubicación del archivo)

```
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width,
initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="./css/style.css">
</head>
```

✓ Este método es el recomendado.

Creando archivo desde VS Code

En este ejemplo vemos cómo crear la carpeta que guardará los archivos .css que vas a crear y el archivo CSS, desde **VS Code**. La cantidad de archivos puede ser la necesaria, pero como primer paso haremos un solo archivo. Aunque sea uno solo, igualmente debe estar alojado en la carpeta correspondiente.

Dentro del **<head>** de nuestro archivo HTML escribimos la etiqueta **<link>** y el programa mostrará sugerencias, de las cuales debemos seleccionar **"link:css"**. Los atributos que nos aparecen son:

- **rel:** Especifica la relación entre el documento actual y el recurso externo enlazado
- **href:** Especifica la ubicación del recurso externo que se va a enlazar. Con escribir solo un punto, dará las sugerencias de qué carpeta y archivo buscar.
 - No olvidemos ./ para ingresar a un archivo o carpeta y ../ para salir

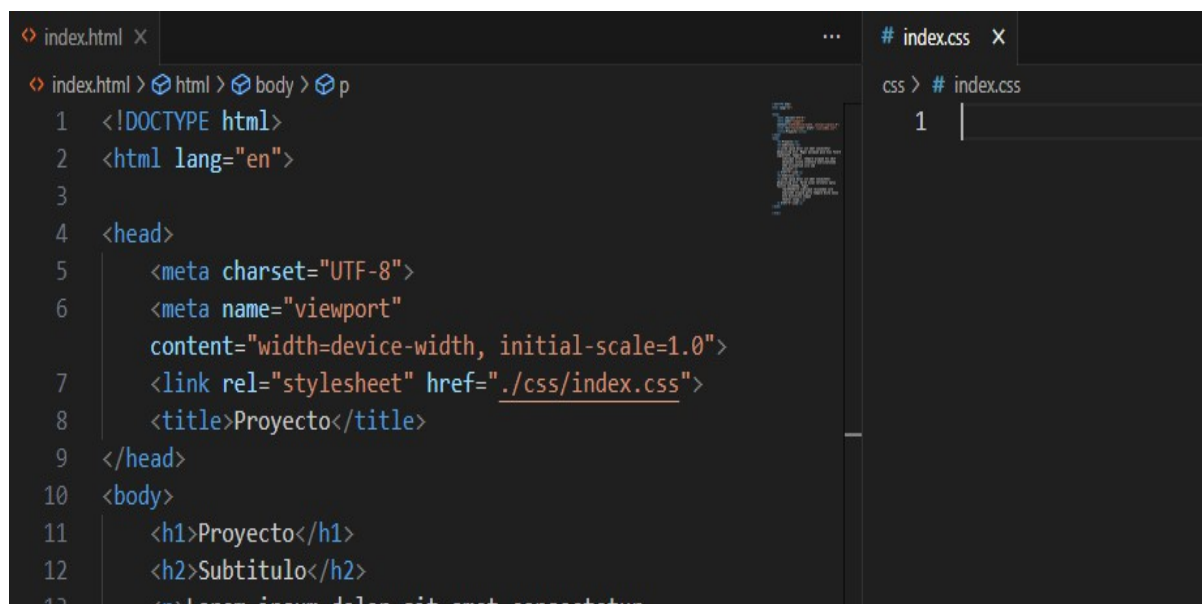
Algunas recomendaciones:

Estas recomendaciones son para ayudarte a trabajar de manera más cómoda las modificaciones que realices en ambos archivos, tanto .html como .css.

Primera recomendación: En la parte superior derecha de **VS Code**, encontrarás este ícono.



Al hacerle clic te permitirá ver dos archivos a la vez, de la siguiente manera:

The image shows a screenshot of the Visual Studio Code editor interface. On the left, the 'index.html' file is open, displaying HTML code with line numbers 1 through 13. The code includes a DOCTYPE declaration, an HTML lang attribute, a head section with meta tags for charset and viewport, a link to a stylesheet, and a title. The body section contains two heading tags. On the right, the '# index.css' file is open, showing a single line of code. The editor's breadcrumb navigation at the top indicates the current file is 'index.html' and the current view is 'css > # index.css'.

Segunda recomendación: Utilizando el atajo **Alt+Z** se logra que todo el texto entre en una ventana más angosta, evitando que se deba scrollear hacia los costados.

Tercera recomendación: es para configurar la posibilidad de hacer zoom en el código.

Archivos > Preferencias > Configuración

En el buscador de la pestaña “**Configuración**”, buscamos zoom y dejamos marcada la última casilla “**Mouse Wheel Zoom**”.

Esta última configuración permite que al hacer clic en **Ctrl+rueda** del mouse, puedas hacer zoom sobre el trabajo.

¡Listo!

Ahora si, el programa VS Code está aún más cómodo de utilizar para esta siguiente etapa en la que se aplicarán estilos al sitio web desde el archivo CSS linkeado al HTML.

Propiedades para textos y fondos

En este tercer tema veremos con mayor profundidad algunas propiedades de las más utilizadas que son las de color y fondo. Ambas son las primeras en retocarse y las que comienzan a definir la “personalidad” o el “humor” del sitio web. Para esto podemos usar una paleta de colores, que es una colección de colores seleccionados y organizados de manera armoniosa.

Se utiliza en diseño gráfico, diseño web y otras disciplinas creativas para elegir y combinar colores de forma coherente y estética. Además, ayuda a transmitir emociones, establecer la identidad de una marca y crear una experiencia visual cohesiva. Teniendo una paleta seleccionada es más fácil su aplicación desde el código. Por otro lado también se verá cómo modificar otros aspectos de los fondos y otros aspectos de los textos, como su tamaño, su peso, etc.

Color:

Ya tenemos preparado el archivo **CSS** linkeado al archivo **HTML**. Ahora sí, vamos a darle color al proyecto.

Antes de comenzar te recomendamos tener presente tu boceto. Aunque en un primer momento no era necesario agregar color a los wireframes de las primeras clases, ahora que aplicamos estilo, es importante definir una paleta de color representativa para tu sitio web.

Te dejamos algunos sitios donde podés generarla.

- [colorhunt](#)
- [colorhail](#)
- [color Adobe](#)
- [canva colors](#) (Genera una paleta de color a partir de una imagen)

- [blend](#) (Genera gradientes rectos o circulares)
- [htmlcolorcode](#)

¡A tener en cuenta!

Existen tres maneras de escribir colores.

Veamos cada una de ellas.

HSL: (Hue, Saturation, Lightness) Se utiliza para definir colores de manera más intuitiva y flexible. Se basa en la combinación de componentes de rojo, verde y azul, el modelo HSL se basa en tres propiedades

RGB: (Red, Green, Blue) Se utiliza para definir colores mezclando diferentes cantidades de los componentes rojo, verde y azul. Cada componente puede tener un valor entre 0 y 255, donde 0 representa la ausencia del color y 255 representa la intensidad máxima del color.

HEX: Utiliza combinaciones de números y letras. En CSS, los colores se pueden especificar utilizando una notación de seis dígitos, precedidos por el símbolo "#".

✓ **Esta última es la más utilizada.**

Ya seleccionada la paleta de colores, es cuestión de ver en qué sectores se visualiza mejor en relación al contraste pensado en el diseño.

Propiedades para texto

Ya vimos los primeros cambios con colores. Veamos qué otras propiedades podemos aplicar a los textos. Algunas de las más utilizadas son:

font-family: Define la fuente o tipo de letra a utilizar. (lo veremos más adelante)

font-size: Especifica el tamaño del texto.

font-weight: Indica el grosor o peso del texto.

font-style: Define el estilo del texto, como normal, cursiva o inclinada.

text-align: Controla la alineación del texto dentro de su contenedor, como izquierda, derecha, centrado o justificado.

text-decoration: Permite agregar decoraciones al texto, como subrayado, tachado o resaltado.

text-transform: Cambia la capitalización del texto, como mayúsculas, minúsculas o capitalización inicial.

line-height: Establece la altura de línea, es decir, el espacio vertical entre líneas de texto.

letter-spacing: Controla el espaciado entre caracteres del texto.

Para estas propiedades puede que no conozcamos o no aprendamos todos los valores que pueden abarcar, pero emmet nos puede predecir algunas opciones y gracias a **live server**, podemos ver cómo se ven esos cambios en tiempo real mientras lo probamos.

Propiedades de fondo

Como vimos antes, CSS nos brinda una serie de propiedades para que podamos personalizar el fondo de los elementos HTML. Todas estas se desprenden de una propiedad llamada background.



De momento, lo que probamos fue cambiarle el color de fondo del body, pero podemos cambiarle el color de fondo a diferentes elementos. Por ejemplo a los párrafos.

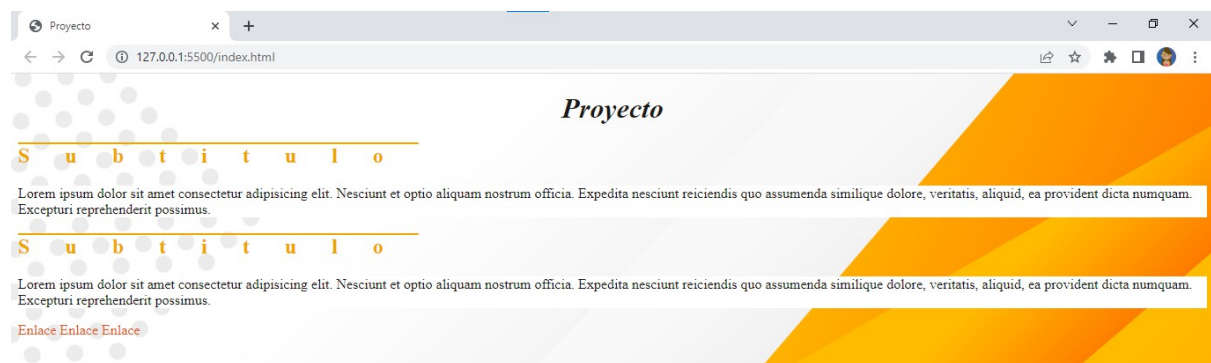
Existe una posibilidad aún más interesante que es la de poder agregar imágenes como fondos. Estas imágenes pueden ser de cualquier temática, desde algo formal hasta algo abstracto, o hasta texturas. Se utiliza background-image y hay dos maneras de colocar imágenes como fondos.

La primera es utilizando alguna imágenes que se tenga en la carpeta de imágenes y utilizando la siguiente regla CSS:

```
background-image: url(../imagenes/fondo.jpg);
```

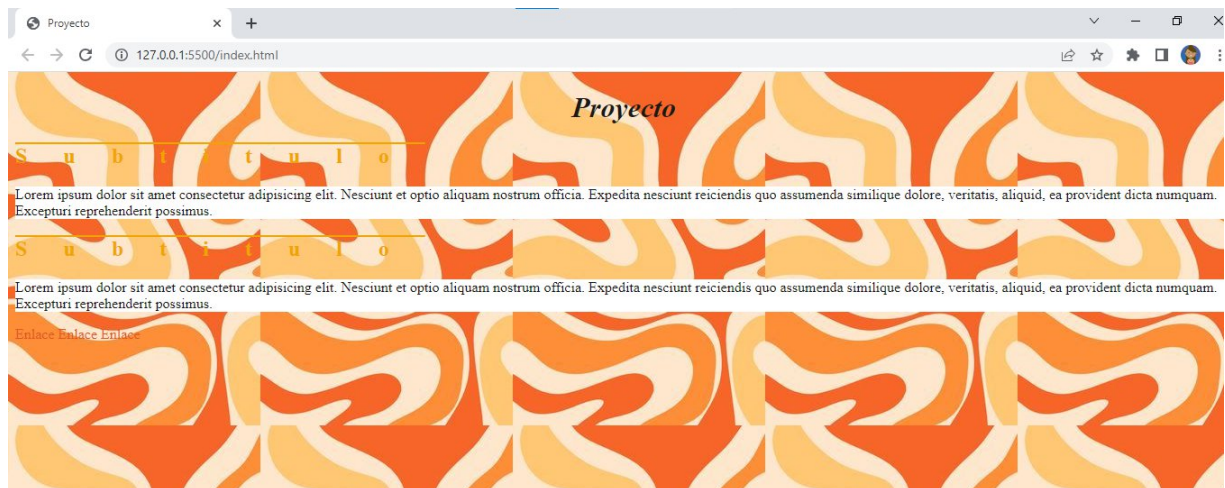
Al estar dentro de la carpeta de css, debemos colocar dos puntos al inicio para indicar que salimos de esa carpeta para ingresar en otra, que es “**imágenes**”.

Resultado



La segunda forma es copiando directamente la dirección de la imagen encontrada en internet entre los paréntesis.

Resultado:



¿Ves el problema?

Ya sea por el tamaño, la posición, el color de fondo, etc. de la imagen probablemente sea preciso ajustar alguna propiedad extra. Es por eso que CSS nos trae un conjunto de propiedades complementarias al background-image para optimizar la forma en la que se ve nuestro fondo.



background-repeat

background-repeat: controla si la imagen de fondo se repite en el área disponible. Puede tener los siguientes valores:

repeat: la imagen se repite tanto en el eje horizontal como en el vertical (por defecto).

repeat-x: la imagen se repite solo en el eje horizontal.

repeat-y: la imagen se repite solo en el eje vertical.

no-repeat: la imagen no se repite.

background-position

background-position: especifica la posición inicial de la imagen de fondo dentro del elemento. Puede utilizar valores como píxeles, porcentajes o palabras clave como:

left (izquierda)

right (derecha)

top (arriba)

bottom (abajo)

center

background-size

background-size: establece el tamaño de la imagen de fondo. Puede utilizar valores como:

píxeles

porcentajes

auto

cover (ajustar la imagen para que cubra completamente el área)

contain (ajustar la imagen para que se ajuste completamente dentro del área sin recortarse)

background-attachment

La propiedad background-attachment en CSS se utiliza para controlar si una imagen de fondo se desplaza con el contenido de la página o se mantiene fija en su posición mientras se desplaza el contenido.

Los posibles valores para **background-attachment** son:

scroll: Es el valor por defecto. La imagen de fondo se desplaza junto con el contenido de la página. Si el usuario se desplaza hacia arriba o hacia abajo, la imagen de fondo se moverá con el contenido.

fixed: La imagen de fondo permanece fija en su posición relativa a la ventana del navegador. A medida que el usuario se desplaza por la página, la imagen de fondo no se moverá.

local: La imagen de fondo se desplaza junto con el contenido de su contenedor. Si el contenido es más largo que el contenedor y se produce un desplazamiento, la imagen de fondo se moverá con el contenido dentro del contenedor, pero no se desplazará más allá de los límites del contenedor.



Recapitulando:

Hasta ahora hemos visto las primeras propiedades que modificamos en CSS

- Colores en texto y fondo
- Propiedades de los textos (**font-style**, **text-align**, **text-decoration**, **text-transform**, **etc**)
- Propiedades para fondo que nos permiten ajustar las imágenes



¡No te preocupes!

Lo bueno de tener un boceto es que nos ayuda a marcar el camino por el que debemos guiarnos. Puede que en el proceso el diseño cambie a algo diferente y eso está muy bien.

Conoceremos más propiedades CSS, pero la buena noticia es que no todas deben aplicarse, sino que debes usar las que se ajusten mejor a tus necesidades, a tu boceto y a tu proyecto.

Lo importante es que conozcas todas estas herramientas que tenés a disposición, cómo se las utiliza o al menos conozcas sus nombres para poder investigar por tu cuenta.



Clases

Ya pusimos en práctica algunas propiedades y como vimos, muchos elementos cambian características a la vez. Quizás existan elementos en específico que no queramos que se vean así, como todos los `<h2>`, o todos los `<p>`. Para hacer que tengan otras características se utilizan las clases, que son selectores especiales asignados a cada elemento en particular para que posea propiedades únicas. Son muy fáciles de usar y serán necesarias a lo largo de todo el curso.

Para conocer más te invitamos a ver la siguiente presentación.

[*Insertar presentacion Clase 03 - Clases](#)

Selectores de Clase

En **CSS**, las clases son selectores que se utilizan para aplicar estilos a uno o más elementos HTML. Una clase se define mediante el uso del atributo `class` en el elemento HTML, seguido de un nombre de clase único que se elige para identificar y agrupar elementos específicos.

Las clases permiten aplicar estilos a múltiples elementos en un documento HTML sin tener que repetir el mismo estilo para cada uno de ellos. Al asignar una clase a un elemento, se le puede aplicar una serie de estilos definidos en la hoja de estilo CSS correspondiente a esa clase.

Su mejor función es la de seleccionar elementos específicos y asignarle sólo a ellos alguna propiedad en particular.

Recordemos el ejemplo del tema anterior. Veamos cómo le cambiamos el color a dos párrafos diferentes.

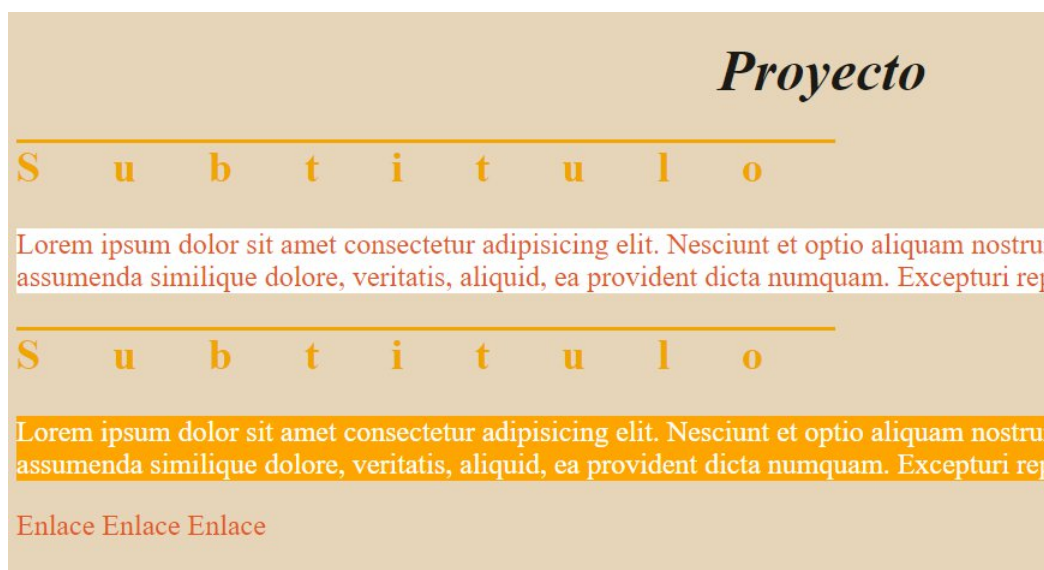
En el ejemplo anterior, todos los elementos `<p>` cambiaban al color asignado en el CSS. Con ayuda del atributo `“class”`, lo que puedo hacer es asignarles diferentes propiedades a cada elemento por separado. En este ejemplo le asignaremos una clase a cada elemento `<p>` y le daremos diferentes atributos.

Una se llamará `“parrafoUno”`, y la otra `“parrafoDos”`

Luego en el CSS, le daremos las diferentes propiedades. Para `“llamarlas”` o nombrarlas, las debemos mencionar con un punto adelante y luego el nombre. De la siguiente manera:

Resultado:

Podemos ver cómo a cada párrafo le dimos propiedades por separado con ayuda de las clases.



A tener en cuenta:

- Se pueden utilizar la cantidad de clases que se crean necesarias. Es muy probable que todos los elementos tengan clases y eso está bien.
- Evitar que el nombre de las clases contenga números, caracteres extraños como la letra Ñ, acentos, guiones, puntos, etc.
- El nombre de la clase debe comenzar con una letra.
- Si el nombre de la clase tiene dos nombres te sugerimos usar el método camelCase, que indica que la primera letra de la segunda palabra debe ir en mayúsculas.

Ahora ¡A sumar clases!

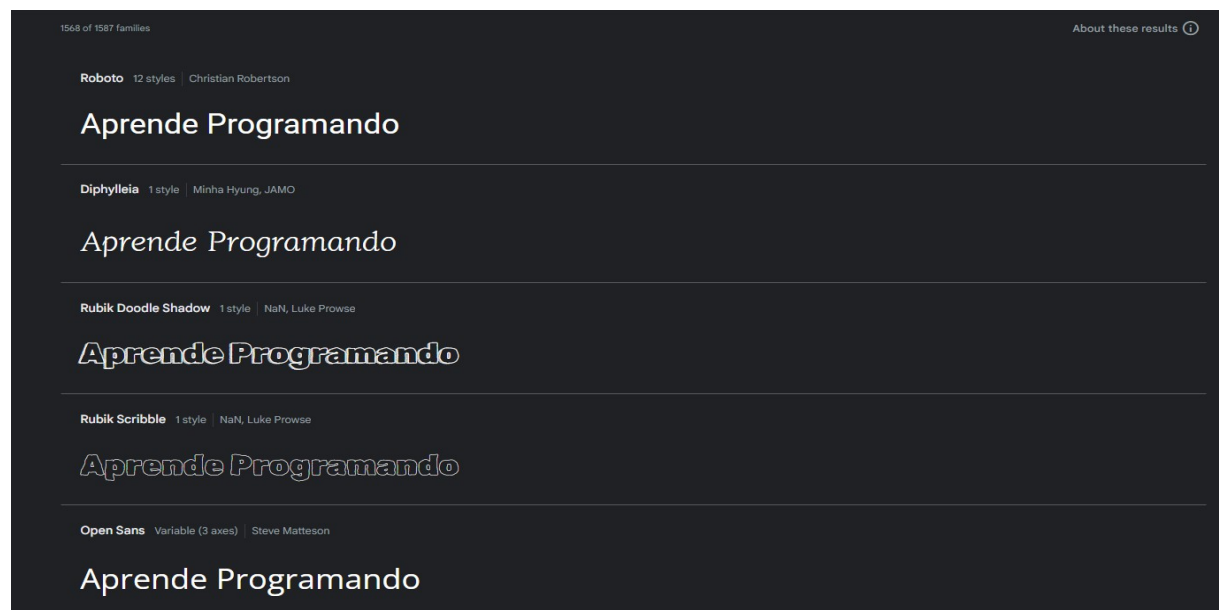
Tipografías en CSS

La tipografía desempeña un papel crucial en el diseño web, afectando la legibilidad y la estética de una página. En esta sección, aprenderemos cómo incorporar y manejar tipografías en **CSS** para darle una identidad única a nuestra página web.

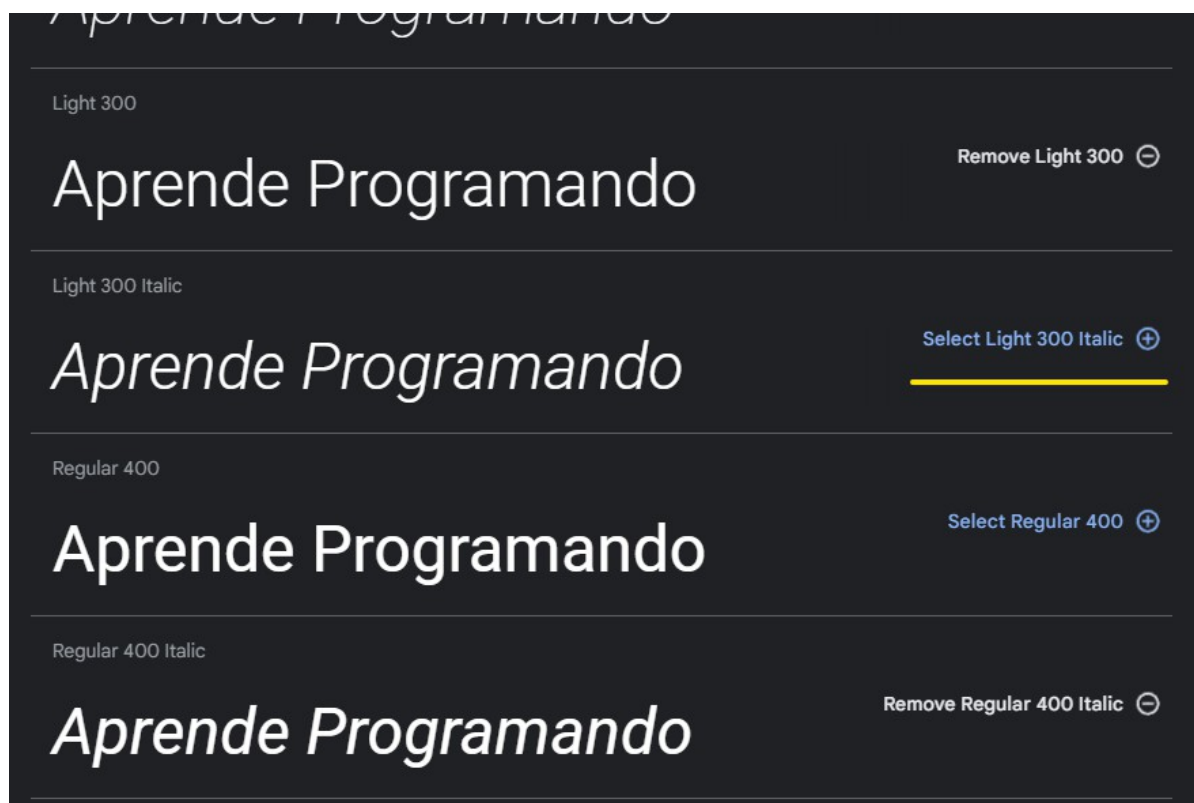
La propiedad **font-family en CSS** nos permite definir qué tipo de fuente queremos aplicar a nuestro texto. Una excelente fuente para ello es **Google Fonts**. Veamos cómo importar e incorporar una fuente desde **Google Fonts**.

Google Fonts

Primero, seleccionamos la fuente que deseas en **Google Fonts**. <https://fonts.google.com/>
Para el ejemplo trabajaremos con la fuente Roboto. Pero pueden investigar y utilizar las que mas les guste.



Una vez que elegimos la familia de la fuente le haremos click, se abrirá una nueva vista en la que podremos seleccionar los “pesos” (font-weight) que definen el grosor del trazo de cada carácter, también podemos seleccionar el estilo cursiva (italic). Para ir seleccionando las que queremos usar, debemos hacer click en el botón azul con el símbolo “+”. De esta forma iremos agregándolas a nuestra selección.



Luego, hacemos clic en el ícono de la bolsa de compras en la parte superior y seguimos las instrucciones para incorporarla en nuestra página. (ojo! no cierras la página todavía!)

Aquí se irán guardando
las fuentes que elijamos

The screenshot shows the Google Fonts website interface. At the top, there is a search bar and a 'Sort by: Trending' dropdown. A blue arrow points from the handwritten text 'Aquí se irán guardando las fuentes que elijamos' to the 'Download family' button. Below the search bar, there are tabs for 'tester', 'Glyphs', and 'About & license'. A 'Download family' button is highlighted with a blue circle. The main area displays three font preview cards, each with the word 'Programando' in a different font style. The first card is 'Thin 100', the second is 'Thin 100 Italic', and the third is 'Light 300'. On the right side, there is a 'Selected family' panel. It has a 'Review' section showing the selected font family 'Roboto' and its styles: 'Light 300', 'Light 300 Italic', and 'Regular 400 Italic'. Below this is a 'Use on the web' section with two radio buttons: '<link>' (selected) and '@import'. Below the radio buttons is a code block containing the HTML code for embedding the font. A blue arrow points from the handwritten text 'Estos links debemos copiarlos en el head de nuestro archivo HTML' to the code block.

Search fonts Sort by: Trending

tester Glyphs About & license Download family

Select preview text: Continent Language

48px

Programando Select Thin 100

Programando Select Thin 100 Italic

Programando Remove Light 300

Selected family

Review

Roboto

Light 300

Light 300 Italic

Regular 400 Italic

Add more styles Remove all

Use on the web

To embed a font, copy the code into the <head> of your html

<link> @import

```
<link rel="preconnect" href="https://fonts.googleapis.com">
<link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
<link href="https://fonts.googleapis.com/css2?family=Roboto:ital,wght@0,300;1,300;1,400&display=swap" rel="stylesheet">
```

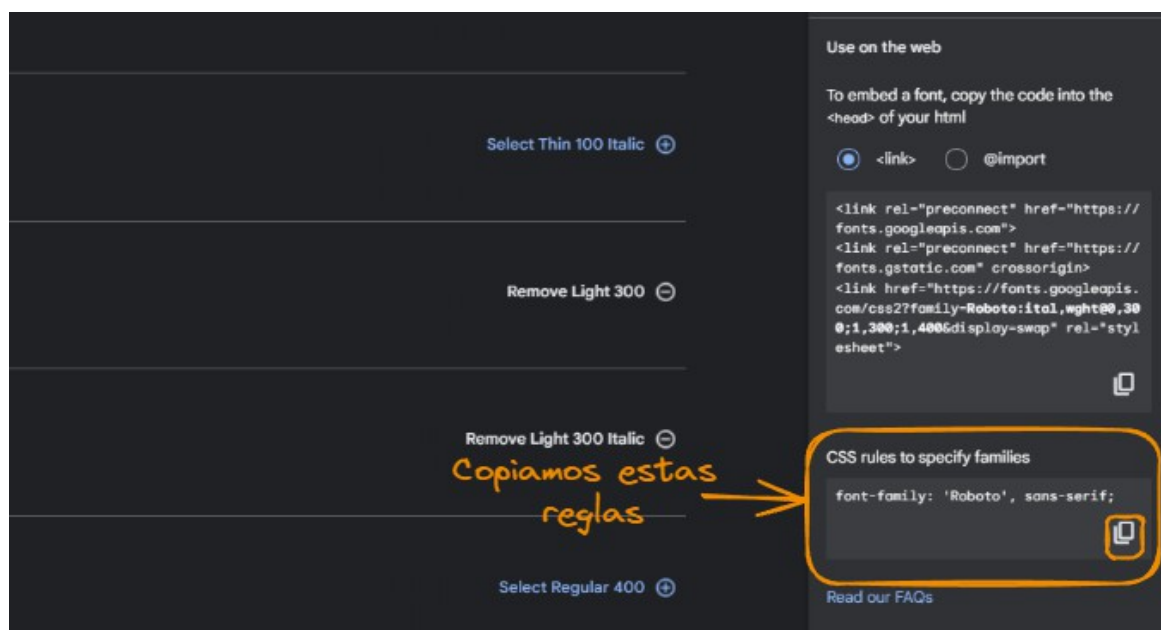
Estos links debemos copiarlos en el head de nuestro archivo HTML

```
index.html > ...
1 <!DOCTYPE html>
2 <html lang="es">
3
4 <head>
5   <meta charset="UTF-8">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <link rel="stylesheet" type="text/css" href="./index.css">
8
9   <link rel="preconnect" href="https://fonts.googleapis.com">
10  <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
11  <link href="https://fonts.googleapis.com/css2?family=Roboto:ital,wght@0,300;1,300;1,400&display=swap" rel="stylesheet">
12
13
14 <title>Ejemplo tipografías</title>
15 </head>
16
17 <body>
18   <header>
19     <h1>Aprendiendo tipografías</h1>
20   </header>
21
22 </body>
23
24 </html>
25
```

Los pegamos aquí

Y listo! ¡Ya podemos usar la tipografía Roboto en nuestra página web! ¿Cómo hacemos?

Nos falta el último paso para que podamos ver la fuente en nuestro navegador. Debemos definir la familia de la fuente en el archivo **CSS**. Para eso volveremos a Google Fonts para buscar las reglas de css que necesitaremos. Las encontramos debajo de los links que copiamos recién.



```
body{  
  font-family: 'Roboto', sans-serif;  
}
```

Aplicaremos la propiedad “**font-family**” al elemento body, de esta forma la fuente afectará a todos los elementos de nuestra página.

¡Ahora sí! Ya podemos empezar a trabajar con distintas tipografías y darle a nuestro sitio una apariencia única e increíble!

Pro tip: Pueden usar todas las fuentes que quieran, pero es importante evitar el “efecto cocoliche”.

Cuando se trata de fuentes, menos es más.

Combinar tipografías para que queden agradables a la vista es un arte en sí mismo y puede ser muy difícil.

Pro tip II: Dos tipografías suelen ser más que suficiente para casi cualquier diseño.



Íconos en HTML

A medida que la tecnología avanza y el ritmo de vida se acelera, los íconos se han convertido en una forma de simplificar mensajes y representar situaciones de la vida cotidiana. En la actualidad, prácticamente todos los logotipos, marcas y aplicaciones utilizan íconos que, gracias a sus colores y formas, transmiten significados reconocibles a simple vista.

Esta simplicidad se basa en una cultura de comunicación masiva en la que la mayoría de las personas comprende el significado de muchos íconos, lo que los convierte en algo cercano a lo universal. Aunque parezcan obvios, los íconos también tienen reglas de uso. No se puede reducir toda la información a unos pocos dibujos simples, pero cuando se utilizan correctamente, pueden transmitir un mensaje claro y lograr que los usuarios se sientan atraídos e interactúen con un sitio web.

Desafío N° 3:

Al archivo **HTML** creado previamente, incluye:

- Un **h1** y un **h2**.
- Agrega un div con un párrafo lleno de texto, y asígnale la clase “practicando”.
- Crea un archivo **CSS** y utilizando el selector de clase, llama a la clase “practicando”; y asígnale la propiedad color con algún valor. utiliza el selector de elementos para darle tamaño y color a los **h1 y h2**. experimenta con los diferentes tipos de valores de color.
- Utiliza una fuente desde <https://fonts.google.com/>





Buenos Aires
aprende
Agencia de Actividades para el Futuro

BA Buenos
Aires
Ciudad