

***RECUERDA PONER A GRABAR LA
CLASE***





¿DUDAS DEL ON-BOARDING?

MIRALO AQUI



Clase 03. DESARROLLO WEB

INCLUYENDO CSS A NUESTRO PROYECTO



OBJETIVOS DE LA CLASE

- Agregar multimedia con HTML.
- Comprender la sintaxis de CSS.
- Incluir CSS en el proyecto.

GLOSARIO:

Clase 2

Listas: HTML permite agrupar elementos que tienen más significado de forma conjunta. Aunque cada palabra por separado tiene sentido, de forma conjunta constituyen el menú de navegación de la página, por lo que su significado conjunto es mayor que por separado. Esto se denomina listas.

Tablas: son un conjunto de celdas organizadas, dentro del cual es posible alojar distintos contenidos. Sirven para representar información tabulada, en filas y columnas.

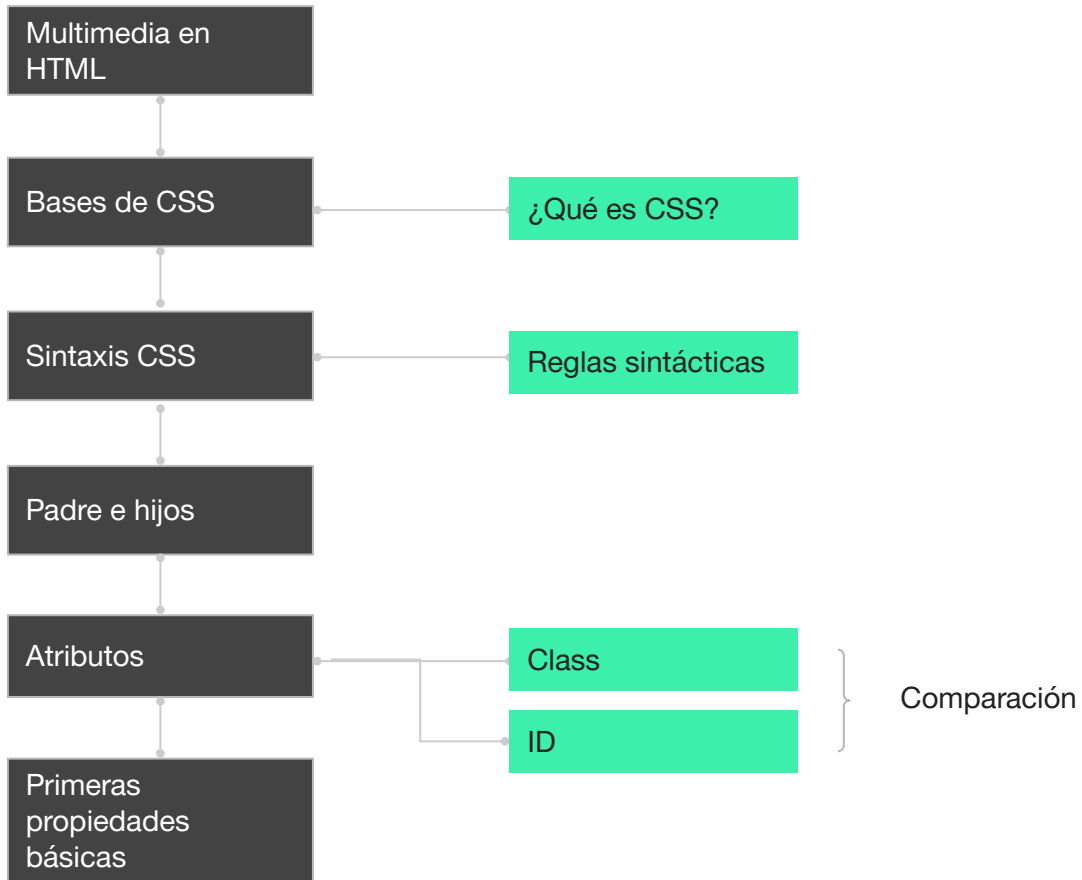
Formularios: son etiquetas donde el usuario ingresará o seleccionará valores, que serán enviados a un archivo encargado de procesar la información.

Enlaces: también conocidos como links o anchors, se utilizan para relacionar partes del mismo documento. Por defecto, se visualizan azules y subrayados.

MAPA DE CONCEPTOS

MAPA DE CONCEPTOS CLASE 3

¡Para
recordar!



CRONOGRAMA DEL CURSO

Clase 2



Primeros pasos con HTML



LISTAS



PRÁCTICAS DE LO
VISTO EN CLASE



FORMULARIOS



ESTRUCTURA HTML DEL
PROYECTO

Clase 3



Incluyendo CSS a nuestro proyecto



PRÁCTICAS DE LO
VISTO EN CLASE



ATRIBUTOS



AGREGANDO CSS A
NUESTRO HTML

Clase 4



CSS - Medidas, colores, fuentes



PRÁCTICAS DE LO
VISTO EN CLASE



ESTILOS



TIPOGRAFÍA



ASIGNANDO ESTILOS



PRIMERA ENTREGA
DEL PROYECTO FINAL

MULTIMEDIA EN HTML

IMÁGENES

- **Enriquecen el HTML:** las imágenes son elementos que enriquecen la experiencia de los usuarios.
- **Insertar imágenes:** se insertan con la etiqueta ``, que pertenece al grupo de las etiquetas que se cierran a sí mismas (con la barra al final). Para funcionar requiere, como mínimo, indicar en dónde está el archivo a mostrar. Eso se hace con el atributo “src” (el source o fuente), que respeta todas las reglas de ruteo vistas en los links. Se comportan como elementos de línea, es significa que se verán una al lado de la otra.

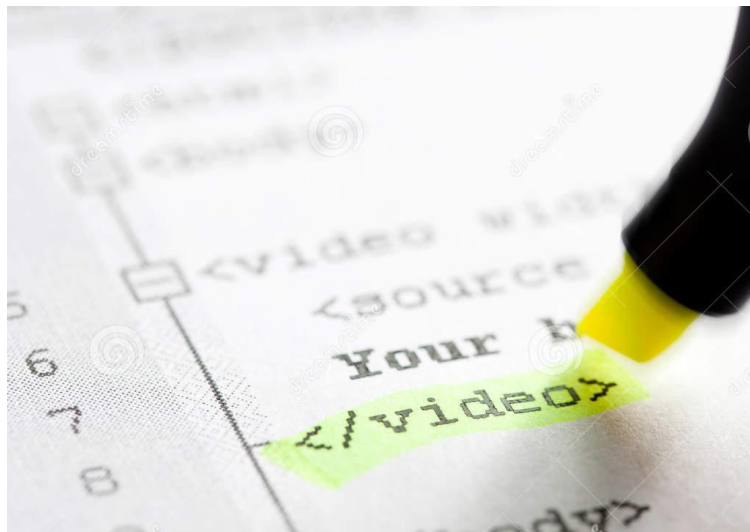
ALT

El "alt" es un texto que debe representar la foto que se está visualizando. Tiene que ser conciso y breve, pero dejar en claro de qué se trata la imagen.

```

```

ETIQUETA VIDEO



```
<video src="mivideo.mp4"  
controls>
```

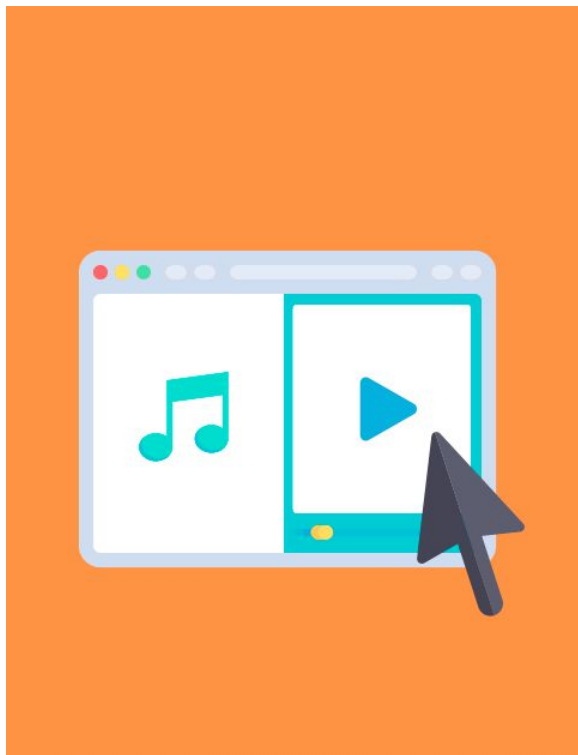
Tu navegador no implementa el
elemento `<video>`.
`</video>`

ETIQUETA AUDIO



```
<audio src="perfect.mp3" preload="auto"  
controls></audio>
```

ATRIBUTOS DE LA ETIQUETA <AUDIO>



- **Controls:** controles para manejar el audio.
- **Autoplay:** reproducción automática.
- **Loop:** repetición automática.
- **Preload:** almacenamiento temporal del audio.
 - "none" : no almacena el archivo.
 - "auto" : almacenamiento temporal del archivo.
 - "metadata" : almacena temporalmente sólo los metadatos del archivo.

Ejemplo
en vivo



¡VAMOS A PRACTICAR LO VISTO!

CODER HOUSE

Se pueden especificar múltiples fuentes de archivos usando el elemento **<source>**, con el fin de proporcionar vídeo o audio codificados en formatos diferentes, para diversos navegadores.


```
<video controls>
  <source src="foo.ogg" type="video/ogg">
  <source src="foo.mp4" type="video/mp4">
  Tu navegador no implementa el elemento
<code>video</code>.
</video>
```


ETIQUETA IFRAME

Es un elemento HTML que permite insertar o incrustar un documento HTML dentro de un documento HTML principal.

```
<iframe src="pagina_fuente.html" width=290  
height=250>Texto para cuando el navegador no  
conoce la etiqueta iframe</iframe>
```

ETIQUETA IFRAME



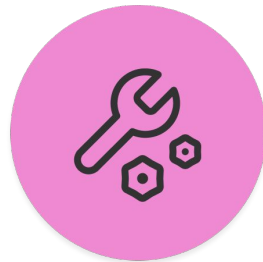
CÓDER Cómo buscar empleo por LinkedIn: "Buscar tra...
Ver más ta... Compartir

Insertar vídeo

```
<iframe width="560" height="315"
src="https://www.youtube.com/embed
/5Ujb8B7JHv4" frameborder="0"
allow="accelerometer; autoplay;
encrypted-media; gyroscope;
picture-in-picture"
allowfullscreen></iframe>
```

☐ Empezar en 0:01

COPIAR



HTML

Crea un archivo HTML.



¡A PRACTICAR!

Crea un archivo HTML que contenga: un video, un audio (puedes descargar ambos de Drive), y un iframe que muestre el video de [Coderhouse](https://www.coderhouse.com). Tienes 15 minutos para realizar la actividad.

BASES DE CSS

PREMISAS

CSS (Cascading Style Sheets) es un lenguaje web para aplicar formato visual (color, tamaño, separación y ubicación) al HTML.

Con él puedes cambiar por completo el aspecto de cualquier etiqueta HTML.

Bienvenidos a Diseño Web

Que la fuerza te acompañe!

- Inicio
- Blog
- Contactos

Sobre el profesor

El profesor explica, hace y ustedes practican

Sobre ustedes

Aprenderan a diseñar una página web



BIENVENIDOS A DISEÑO WEB



Que la fuerza te acompañe!

SOBRE EL PROFESOR



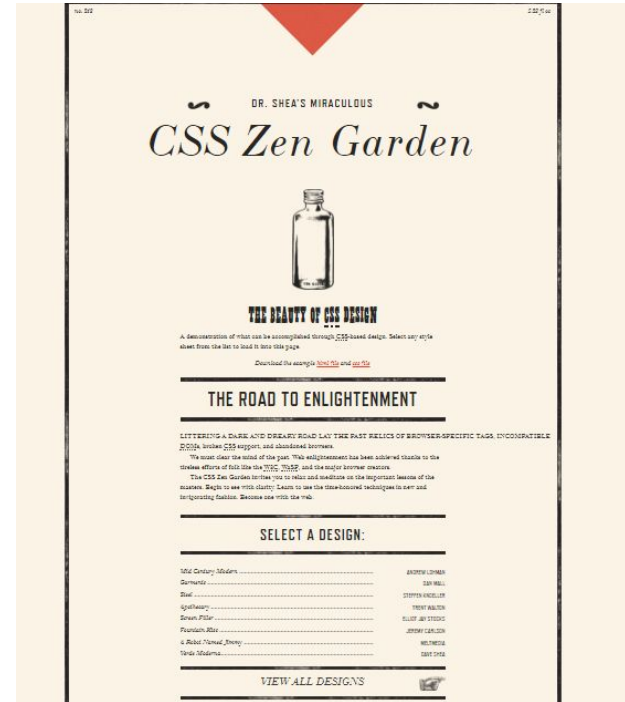
El profesor explica, hace y ustedes practican

SOBRE USTEDES



Aprenderan a diseñar una página web

CSS bien implementado permite cambiar **todo el diseño** de un sitio web, sin modificar el HTML. Las siguientes dos imágenes corresponden al mismo código HTML pero distinto CSS:



SINTAXIS CSS

SINTAXIS

```
selector {  
    propiedad1: valor;  
    propiedad2: valor;  
}
```



```
h1 {  
    color: red;  
}
```

REGLAS SINTÁCTICAS

- Cada declaración CSS está formada por un juego de pares **propiedad: valor;**
- No se ve afectado por el espacio en blanco. Las propiedades se pueden escribir de corrido o una debajo de la otra.
- Siempre que la propiedad represente un número, el valor debe indicar en qué unidad se expresa.

PADRE E HIJOS

PADRE E HIJOS

Cuando tienes una etiqueta “dentro” de otra, lo que haces es aplicar el concepto de padres e hijos.

En este caso, **section** es padre de **article** y, a su vez, **article** es padre del **h2** y del **p**.

```
<section>
  <article>
    <h2>Título</h2>
    <p> Lorem ipsum
elit. Temporibus
="#"> possimus <
debitis!
  </p>
</article>
</section>
```

PADRE E HIJOS

Esto te habilita a agregar atributos específicos a “hijos”, sin alterar los del “padre”. Un padre puede tener muchos hijos, y todos ellos heredan sus características, pudiendo tener también características particulares.

Selector **HIJO**



Selector **PADRE**



```
section article {  
    background-color: #cccccc;  
    width: 500px;  
    height: 500px;  
}
```

PADRE E HIJOS

En este caso, se observa la forma correcta de declarar cada estilo. Cuando quieres seleccionar una etiqueta, debes incluir las etiquetas padre/s para que sean más específicas a la hora de aplicar estilos.

HTML

```
<section>
  <article>
    <h2>Título</h2>
    <p>Lorem ipsum dolor sit amet,
    consectetur adipisicing elit.
    Temporibus cum magnam eos a commodi
    sequi possimus iure, eius nemo
    voluptates fuga debitis!
    </p>
  </article>
</section>
```

```
section{
  padding: 50px 30px 20px 60px;
  margin-left: 40px;
}
section article {
  background-color: #cccccc;
  width: 500px;
  height: 500px;
}
section article p{
  line-height: 4;
}
```

CSS

CODER HOUSE

INSERTAR CSS

INSERTAR CSS EN EL HTML

Forma **EXTERNA**: dentro de la etiqueta **<head>**, llamas al archivo CSS que necesites (recuerda el uso de rutas relativas y absolutas).

```
<link rel="stylesheet" href="archivo.css" />
```

1

INSERTAR CSS EN EL HTML

Forma **INTERNA**: es recomendable que esté dentro de la etiqueta **<head>**. Puede estar en **<body>**, pero sería más desprolijo.

```
<style>  
    /* comentario de CSS, dentro de esta etiqueta, va el  
    codigo CSS, */  
</style>
```

2

INSERTAR CSS EN EL HTML

Otra forma **INTERNA**, muy poco recomendable, consiste en usar para “parches” específicos, o pruebas. Se hace difícil mantenerlo.

```
<h1>Un encabezado sin formato</h1>  
<h2 style="CODIGO CSS">H2 con formato CSS</h2>  
  
<p>Párrafo sin formatear</p>  
<p style="CODIGO CSS">Párrafo formateado</p>  
<p>Otro párrafo sin formatear</p>
```

3

Ejemplo
en vivo



¡VAMOS A PRACTICAR LO VISTO!

CLASS

CLASS

Generalmente se utiliza para **darle estilos a cierta parte del código**. Por ejemplo, si quieres que una imagen tenga bordes, y que además sean redondeados.

CLASS DESDE CSS

Desde CSS, **puedes usar los nombres que quieras**, siempre y cuando empiecen con **LETRAS**, y pongas un “.” adelante. Lo recomendable es poner un nombre que haga referencias a los estilos que tendrá. Por ejemplo:

```
.bordesRedondeados {  
  /* codigo CSS */  
}
```

HTML: ATRIBUTO CLASS=""

En el HTML, para aplicar una clase debes usar el atributo **“class”**, y luego colocar en el **valor** el **nombre de la clase** (que has especificado en CSS).

```
<img src="" class="bordesRedondeados" />
```


MÁS DE UNA CLASS

Puedes aplicar **más de una clase** a cada etiqueta separada por un espacio. De esta manera, podrás tener estilos diferenciados para cada clase.

```
<img src="" class="bordesRedondeados imgChica" />
```

ATRIBUTO ID

ID

- Generalmente se usa para nombrar porciones de código y sectores, como por ejemplo cuando quieres nombrar distintas secciones.
- Es posible ponerle ID a cualquier elemento HTML para darle un "nombre". Y así como el ID, todos los elementos también aceptan el atributo class="".
- Dicha clase se utiliza cuando quieres aplicar el mismo estilo a más de un elemento, y la búsqueda por etiqueta no sirve para lograrlo. No necesitas escribir varias veces el mismo CSS, ni repetir el ID.

ID DESDE CSS

Desde CSS, **puedes usar los nombres que quieras**, siempre y cuando empiecen con **LETRAS**, y pongas un “#” adelante. Lo recomendable es poner un nombre que haga referencias a los estilos que tendrá. Por ejemplo:

```
#productos {  
  /* codigo CSS */  
}
```

HTML: ATRIBUTO ID=""

Para aplicar un ID en el HTML, debes usar el atributo “id”, y luego en el valor el nombre del ID (que has especificado en CSS). Por ejemplo:

```
<section id="productos">
```

```
</section>
```


COMPARACIÓN CLASS VS. ID

COMPARACIÓN

	¿Se puede reutilizar su nombre en el HTML?	¿Se puede usar varias veces en un atributo en el HTML?	¿Cuándo lo uso?
ID	NO	NO	Nombrar secciones, divisiones de código
CLASS	SI	SI	Especificar diseño aparte del código
Ejemplo ID	id="productos" id="productos2"		<section id="productos">
Ejemplo CLASS	class="bordes" class="bordes"	class="bordes destacado"	<p class="destacado">


EJEMPLO

HTML:



```
<section id= "prod">  
  <article class= "rojo">  
  </article>  
  <article id= "prod">  
  </article>  
</section>
```

HTML:



```
<section id= "prod">  
  <article class= "rojo">  
  </article>  
  <article class= "rojo">  
  </article>  
</section>
```

Tanto **ID** como **Class** pueden ser utilizadas dentro del html en diferentes etiquetas. Sin embargo, los nombres otorgados a las clases se pueden repetir, mientras que utilizados en los IDs no.

Ejemplo
en vivo



¡VAMOS A PRACTICAR LO VISTO!

HERENCIA Y CASCADA

HERENCIA

En general, estas propiedades son intuitivas. Por ejemplo, podrás heredar de un elemento padre el tamaño de letra y color de la misma, *a menos que el elemento hijo tenga otros estilos aplicados*. Puedes ver más al respecto [aquí](#).

```
div {  
    color: red;  
}  
  
<div>  
    <p>Este párrafo quedará en rojo,  
    por herencia</p>  
</div>
```

CASCADA

El navegador lee de arriba hacia abajo (forma de cascada)
¿qué color crees que se aplicará al párrafo (p) al ver el
siguiente código?

```
p {  
    color: red;  
}  
  
p {  
    color: green;  
}
```

EJEMPLO DE BUENA PRÁCTICA



HTML:

```
<ul>
  <li class="rojo">Item
R1</li>
  <li class="rojo">Item
R2</li>
  <li class="rojo">Item
R3</li>
  <li class="rojo">Item
R4</li>
</ul>
```

CSS:

```
.rojo{ color: red; }
.azul{ color: blue; }
```

HTML:

```
<ul>
  <li>Item R1</li>
  <li>Item R2</li>
  <li>Item R3</li>
  <li>Item R4</li>
</ul>
<ol>
  <li>Item A1</li>
  <li>Item A2</li>
</ol>
```

CSS:

```
ul li{ color: red; }
ol li{ color: blue; }
```



Ejemplo
en vivo



¡VAMOS A PRACTICAR LO VISTO!

PRECEDENCIA DE DECLARACIONES

Cuando reglas distintas apuntan al mismo objeto:

- Si son **propiedades distintas**, se suman (se combinan).
- Si tienen **alguna propiedad repetida**, sólo una queda.

Esto es lo que se denomina *precedencia*.

- **ID** pisa cualquier otra regla.
- **Class** sobrescribe las reglas de etiqueta, pero no las de **ID**.
- **Etiquetas** tienen la menor precedencia.

ID > Class > Etiquetas

ESTILOS INLINE

Si utilizas estilos inline, sobrescribirán cualquier estilo de las páginas externas de CSS. Se podría decir que los estilos inline son los que tienen una mayor especificidad, por lo tanto, no es recomendable utilizarlos en tu página.

```
<p style= "color: red">Párrafo rojo</p>
```


ESPECIFICIDAD

En este gráfico se resume cuán importante es cada selector:



Estilo aplicado
a la **Etiqueta**.



Estilo aplicado
a la **Class**.



Estilo aplicado
al **ID**.



Estilo aplicado
Inline.

!IMPORTANT;

- Si tienes 3 reglas CSS, es poco probable que “choquen”, pero en un CSS extenso es más común.
- La declaración ***!important;*** corta la precedencia. Se escribe después del valor de la propiedad CSS que se quiere convertir en la más importante. Se utiliza un ***!important;*** por cada valor a pisar.

Si necesitas más de 5 *!important;* en todo tu CSS, algo estás haciendo mal.



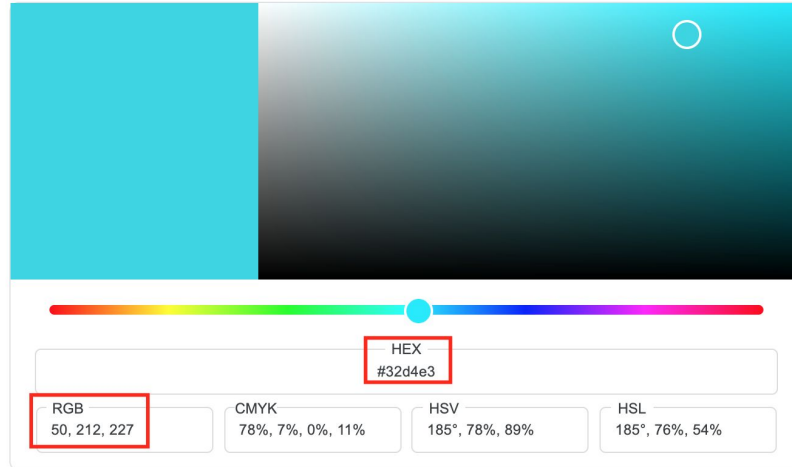
PRIMERAS PROPIEDADES BÁSICAS

PROPIEDAD: COLOR

Mediante esta propiedad, podrás agregar color a los textos de tu sitio, pero... ¿cómo se eligen los colores?

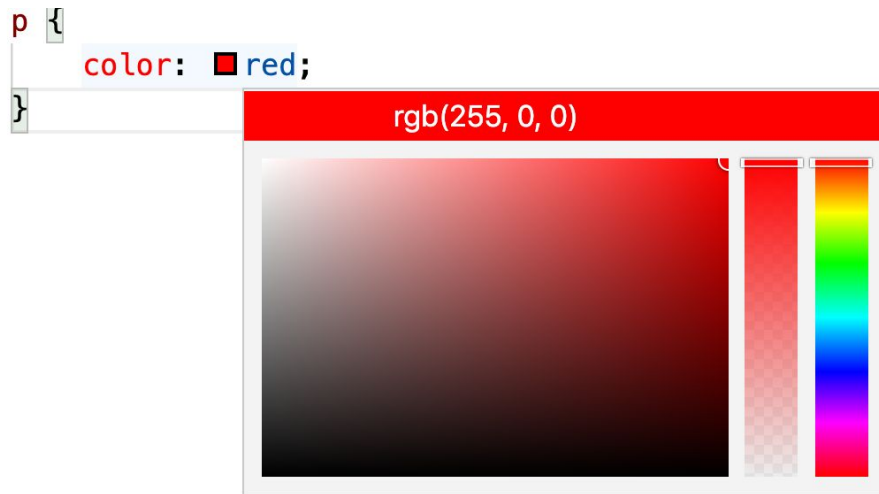
PROPIEDAD: COLOR

Desde Google, puedes buscar “[color picker](#)”
(alternativa [w3schools](#)).



PROPIEDAD: COLOR

Desde Visual Studio Code, simplemente te “paras” sobre el color. Por ejemplo, escribe “red” y haz la prueba:



TIPOS DE VALORES PARA COLOR

Existen distintos valores, pero nos centraremos en 3:

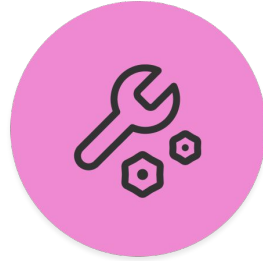
- Por nombre del color (ej: red).
- Hexadecimal (ej: #ffffff).
- RGB (por ejemplo: 50, 212, 227). Si agregas un valor más, puedes manejar su opacidad. (red, green, blue) cada color permite hasta 256 valores.

Ejemplo
en vivo



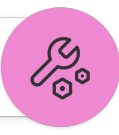
¡VAMOS A PRACTICAR LO VISTO!

CODER HOUSE



ATRIBUTOS

Agrega atributos a los archivos.



¡A PRACTICAR!

Al archivo HTML creado previamente, agrega un div con un párrafo lleno de texto, y asígnale la clase “desafio1”. Además, crea un archivo CSS con una clase llamada “desafio1”; y asígnale la propiedad color con valor naranja (orange). Tienes 15 minutos para completar la actividad.



AGREGANDO CSS A NUESTRO HTML

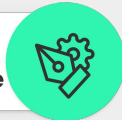
Comienza a utilizar CSS en tu proyecto.

AGREGANDO CSS A NUESTRO HTML

Formato: archivo HTML y CSS. Debe tener el nombre “Idea+Apellido”.

Sugerencia: crear carpeta en formato zip o rar, con el/los archivos HTML y CSS.

Desafío
entregable



>> Consigna: crea un archivo CSS, y linkéalo al HTML entregado en el desafío de la clase anterior, esto es, “Primeros pasos con HTML”. Asigna color a títulos, párrafos y listas, mediante clases y etiquetas.

>>Aspectos a incluir en el entregable:

Detalle completo acerca de lo que se espera que el estudiante entregue. Se sugiere detallar cada uno de los ítems.

>>Ejemplo:

[Carpeta comprimida con los archivos](#)



CSS EN TU PROYECTO

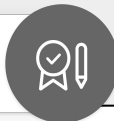
Comenzar a utilizar CSS en tu proyecto.

CSS

Formato: archivo html y css

Sugerencia: carpeta en formato zip o rar con el/los archivos html y css.

Desafío
Complementario

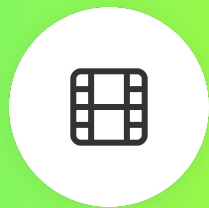


>> Consigna:

1. Tomando como base las estructuras creadas en HTML de la clase dos para tu proyecto, dar formato al menú, color a los títulos, formato a las listas a través de etiquetas.
2. Agregar videos e imágenes a alguna de las secciones del proyecto y darle formato a los títulos de las tablas, agregando color.
3. Se evaluará el estilo y estructura del código.

¿PREGUNTAS?





***¿QUIERES SABER MÁS? TE DEJAMOS
MATERIAL AMPLIADO DE LA CLASE***

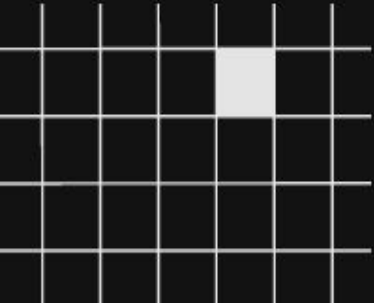


- [Patrones sutiles](#) | **Toptal**
- [Recursos de Dominio Público](#) | **Internet Archive**
- [Jardin Zen CSS](#) | **CSS Zen Garden**



¡MUCHAS GRACIAS!

Resumen de lo visto en clase hoy:

- Multimedia con HTML.
 - CSS: Sintaxis, Reglas, Atributos class & ID.
- 



OPINA Y VALORA ESTA CLASE