

***Preguntas Frecuentes***

***Desarrollo Web***

***CODERHOUSE***

## ***Clase 0***

### **No soy muy experto en computadoras ¿Qué necesito saber antes de empezar el curso?**

No te preocupes, desde Coderhouse nos preocupamos porque la experiencia de todos nuestros estudiantes durante la cursada, sin importar su nivel de conocimientos previos, sea lo más amigable posible. Es por esto que te dejamos este pequeño tutorial acá: <https://www.youtube.com/watch?v=g9Y6nZGtTD0>

### **¿Qué puedo hacer con HTML?**

Usarás HTML para crear toda la estructura y contenido de tu proyecto, como títulos, párrafos, listas, imágenes, videos, entre otros. Así como podrás ver en el siguiente link, crearás el “cuerpo” de tu sitio web: [https://docs.google.com/presentation/d/1e\\_IQjfXt18eeTluVQlkwVUQLaonANi4K9O6cja4qfGc/edit#slide=id.gea953b5762\\_0\\_687](https://docs.google.com/presentation/d/1e_IQjfXt18eeTluVQlkwVUQLaonANi4K9O6cja4qfGc/edit#slide=id.gea953b5762_0_687)

### **¿Qué puedo hacer con CSS?**

Le darás estilos a tu sitio web, entre ellos, colores, tamaños de letras e imágenes, también podrás posicionar tus elementos y separaciones.

Así como se observa en el siguiente link, le “pondrás la ropa” a tu sitio web y quedará hecho a tu medida:

[https://docs.google.com/presentation/d/1e\\_IQjfXt18eeTluVQlkwVUQLaonANi4K9O6cja4qfGc/edit#slide=id.gea953b5762\\_0\\_700](https://docs.google.com/presentation/d/1e_IQjfXt18eeTluVQlkwVUQLaonANi4K9O6cja4qfGc/edit#slide=id.gea953b5762_0_700)

### **¿Cuál es la diferencia entre un sitio web y una página web?**

El sitio web es todo lo que se muestra en el navegador cuando ingresas a una dirección. La página web se refiere a las diferentes secciones dentro de ese contenido. Por ejemplo, supongamos que existe un sitio web llamado lacompra.com, y dentro de este hay diferentes pestañas: inicio, productos, contacto y compras. La mezcla de estas cuatro secciones es el sitio web, pero individualmente, cada una de ellas es una página web.

## **Veo varias opciones de editores de texto ¿Cuál debo usar?**

Si es tu primer contacto con el Desarrollo Web puedes usar cualquiera de las opciones, pero los que más recomendamos son:

Sublime Text. Descárgalo acá <https://www.sublimetext.com/download>

Visual Studio Code. Descárgalo acá:

<https://code.visualstudio.com/download>

Es recomendable que uses el mismo editor de texto que tu profe, así podrás seguir mejor las clases.

## **No logro instalar Balsamiq ¿Qué hago?**

Podés usar la versión web. Solamente tenés que crear una cuenta y podrás hacer exactamente lo mismo que en la app. Podés verla acá:

<https://balsamiq.cloud/>

## **¿Qué sitios web recomiendan para consultar información confiable sobre Desarrollo Web?**

Dos de los sitios más actualizados y más completos para información de este tema son los siguientes:

- <https://www.w3schools.com/>
- <https://developer.mozilla.org/es/>

En ellos podrás encontrar información importante y bien explicada para seguir investigando y aprendiendo.

## ***Clase 1***

### **¿Cómo recomiendan organizarse durante la cursada?**

La organización es la base para una buena cursada. Para que empieces con el pie derecho, te preparamos un pequeño video tutorial con un paso a paso, que podés consultar acá:

<https://www.loom.com/share/17486ff2242743468520bcb1f600e0f8>

¡Recordá ingresar periódicamente a la plataforma de Coderhouse! Allí encontrarás las clases planificadas, el material de clase y las fechas de los desafíos.

**No termino de entender la utilidad de hacer un sketch y un wireframe.**

Si bien no estamos en clases de diseño gráfico, el hacer un **sketch** primero en papel nos ayuda a organizar la estructura visual de nuestro proyecto. Es decir, nos permite previsualizar donde queremos colocar el menú, las imágenes, el texto, etc. Luego se le da un poco más de forma con el **wireframe**.

El poder ver la idea plasmada sobre un soporte físico nos facilita la escritura del código ya que este trabajo previo sirve a modo de guía.

**Veo que existen demasiadas etiquetas en HTML ¿Debo memorizarlas todas?**

La realidad es que no. Cada etiqueta cumple un papel y una función diferente en nuestros códigos y las irás aprendiendo con el uso y práctica.

Te invitamos a leer más en el material complementario de esta clase en el siguiente link:

<https://docs.google.com/document/d/1TXXDSUxpHqLob5GZo6NZfCgp9IEOwSew4pog6i2Ztbc/edit>

**¿Cuándo debo usar etiquetas semánticas y cuándo uso <div>?**

Las etiquetas semánticas son aquellas que definen ciertas partes de nuestro sitio web, por ejemplo **<header>** define la cabecera donde normalmente tenemos nuestro menú de navegación y algún logo; **<nav>** sirve para indicar que lo que está dentro de ella es una navegación interna; **<main>**, **<section>**, **<aside>** y **<article>** entre otras, se usan para ciertas partes de nuestros sitios.

Cuando algún contenido que vayamos a escribir no califique para alguna de esas etiquetas, entonces será un **<div>**.

Sobre este tema te dejamos el link a una consultoría gratuita que fue dictada por uno de nuestros tutores:

<https://plataforma.coderhouse.com/contenidos/videos/semantica-en-html5>

## ***Clase 2***

### **¿Es siempre necesario que escriba mi menú de navegación dentro de una lista?**

Si bien funciona igual en caso que no lo hagas, se considera una buena práctica escribirlo de esta forma ya que se trata de una colección de elementos del mismo tipo, tal como se vió en la clase “*Primeros pasos con HTML*”. Podés volver a ver la ppt al respecto aquí: [https://docs.google.com/presentation/d/1pP80RdXH3hIZHGhyV7pggWDMNc\\_cWt240N0SiY3G4aE/edit#slide=id.gcb4c2b4ce8\\_0\\_679](https://docs.google.com/presentation/d/1pP80RdXH3hIZHGhyV7pggWDMNc_cWt240N0SiY3G4aE/edit#slide=id.gcb4c2b4ce8_0_679)

### **¿Se podrían usar las tablas para organizar todo nuestro sitio web?**

Hace varios años, antes de la entrada de **HTML5** y **CSS3**, se estilaba organizar o mejor dicho: maquetar sitios web usando tablas. Ahora estos dos lenguajes se han hecho tan poderosos que ya no son necesarias para esto. Al día de hoy, se considera una mala práctica usar las tablas para esos fines.

### **¿El formulario de mi proyecto va a poder usarse para enviar información real?**

Si bien en este curso veremos la estructura básica de un formulario y cómo funciona, para hacerlo totalmente funcional se requieren de lenguajes de programación más avanzados que te permitirán recibir toda la información que ingrese el usuario.

Te invitamos a consultar el material complementario de la clase: <https://docs.google.com/document/d/17WqMgdAUSTtPIboUQUZktgOqTZFERMLKkesSHL3eals/edit>

Además, en la carpeta de tu comisión encontrarás un tutorial que puede ayudarte al momento de hacer un formulario: <https://drive.google.com/drive/u/0/folders/1XSJdE-9T3OAxUCNjBBQqLD3XGr0s5Pz>

## ¿Cómo puedo hacer que un enlace me abra en otra pestaña del navegador?

Solo debes agregarle el atributo **target="\_blank"**. Por ejemplo:

```
<a href="http://www.coderhouse.com/frontend" target="_blank">Curso  
de Frontend</a>
```

Recuerda que esto se recomienda solo para enlaces fuera de tu proyecto que no te pertenecen, como por ejemplo, un perfil de alguna red social o escribir por WhatsApp. Es decir, podés usar este tip para cualquier sitio web que no sea el tuyo 🍌

## ¿Es obligatorio usar el atributo "alt" en las imágenes?

Este atributo, junto con el atributo "title", ayuda a que tu sitio web sea más eficiente e inclusivo para todas las personas, incluyendo para quienes presentan dificultades visuales.

A continuación, te compartimos una presentación que realizamos específicamente para abordar esta pregunta:

[https://docs.google.com/presentation/d/1IPsbjQhTPdV8OkZkTnVBjpiZsA0vuDMv2XuJ8dyBP14/edit#slide=id.g10113c4e71e\\_0\\_2](https://docs.google.com/presentation/d/1IPsbjQhTPdV8OkZkTnVBjpiZsA0vuDMv2XuJ8dyBP14/edit#slide=id.g10113c4e71e_0_2)

## Quiero agregar un mapa desde Google Maps pero no logro hacerlo:

Debés usar el mismo principio en el que se trabajó durante el encuentro sincrónico sobre los videos de YouTube. En la carpeta de clase te dejamos un video tutorial con el paso a paso. Lo mismo debes hacer si quieres agregar, por ejemplo, tu álbum favorito de Spotify.

Para insertar mapa seguí el siguiente video tutorial:

<https://drive.google.com/file/d/1gNK7B69UWH-e2OvmK9HW-GugF7INU966/view?usp=sharing>

Para insertar tu álbum de Spotify seguí el siguiente video tutorial:

<https://drive.google.com/file/d/1lscwE2ylgLCmpDzhwV2EJku5kZmXc06T/view?usp=sharing>

## ¿Hay alguna forma de escribir código más rápido en Visual Studio Code?

Visual Studio Code es uno de los tantos editores de código que existen y hoy día, es muy popular.

Un editor de código es una herramienta que se usa para escribir en los distintos lenguajes usados tanto para desarrollo front end como backend. Por lo tanto es la herramienta que más usarás durante el curso para lograr tu cometido. 💪

¿Recuerdas el block de notas de windows? Pues, básicamente un editor de código es un block de notas repotenciado, y por eso, tienes atajos del teclado, algunos vistos en la clase 0: [https://docs.google.com/presentation/d/1e\\_IQjfXt18eeTluVQlkwVUQLaonANi4K9O6cja4qfGc/edit#slide=id.gf327733b10\\_0\\_40](https://docs.google.com/presentation/d/1e_IQjfXt18eeTluVQlkwVUQLaonANi4K9O6cja4qfGc/edit#slide=id.gf327733b10_0_40)

Además, te compartimos un mini tutorial al respecto en la carpeta de la clase ¡Creemos que te será muy útil para aprender a codear! [https://drive.google.com/file/d/1Zc8-J8XzZmD7wi\\_WyPjbzEYdi6ntSgV8/view](https://drive.google.com/file/d/1Zc8-J8XzZmD7wi_WyPjbzEYdi6ntSgV8/view)

## ***Clase 3***

### **¿Qué significa CSS? ¿Por qué son hojas “en cascada”?**

Son las siglas para Cascading Style Sheets, es decir, Hojas de Estilo en Cascada en inglés.

La característica de cascada viene dada porque los navegadores leen los estilos que escribamos en CSS de arriba hacia abajo. Podríamos graficarlo como agua cayendo de una cascada.

Te invitamos a leer un poco más en el material complementario de la clase:

[https://docs.google.com/document/d/1eaN24\\_h2c1\\_jt2giP\\_IYBwi0TDpNaaf5jTrykPlsUM/edit#heading=h.61eqnzumm8ww](https://docs.google.com/document/d/1eaN24_h2c1_jt2giP_IYBwi0TDpNaaf5jTrykPlsUM/edit#heading=h.61eqnzumm8ww)

### **Veo que hay varias formas de usar CSS, ¿Cuál debo aplicar?**

Si bien existen varias formas de usarlo, durante este curso usarás la forma externa, es decir, linkeando tu archivo CSS a los HTML. Esta es la manera que hoy día se recomienda aplicar como buena práctica.

## **¿Debo crear un archivo CSS para cada HTML, o sea, si tengo 5 archivos HTML debo crear 5 CSS?**

No, deberás crear un solo archivo CSS y este mismo linkearlo a todos tus HTML.

## **No logro entender como linkear mi archivo CSS para hacerlo funcionar.**

Debes hacerlo dentro de tu etiqueta **<head>**. Recuerda cuidar las rutas de los archivos para que funcionen correctamente.

Podés consultar el paso a paso en el siguiente mini video tutorial:

<https://drive.google.com/drive/folders/1c3seoUpo53sM-rwvUzhLER5UMNkURJCj>

## **¿Cuándo debo usar clases y cuándo ID?**

Las clases en CSS están destinadas a reutilizarse, puedes usarlas en todas las etiquetas que necesites sin importar su tipo y todas las veces que quieras dentro de un mismo archivo HTML, ya sea para dar color a letras, espacios, tamaños, etc.

Los IDs están destinados a ser únicos en cada sección y no deben utilizarse más de una vez en cada archivo HTML.

Te compartimos una tabla comparativa con las diferencias:

[https://docs.google.com/presentation/d/1IlicAiTHpc7Osd6k9KrTleuz5-87PtKuPPzuZ4lumWU/edit#slide=id.geed3b6cf48\\_0\\_348](https://docs.google.com/presentation/d/1IlicAiTHpc7Osd6k9KrTleuz5-87PtKuPPzuZ4lumWU/edit#slide=id.geed3b6cf48_0_348)

Para seguir practicando cuándo usar etiquetas, clases o ID, te recomendamos el siguiente juego: <http://flukeout.github.io/> 😊

## **¿Hay alguna herramienta que me ayude a definir una paleta de colores para mi proyecto?**

¡Si! Aquí te dejamos algunos recursos donde podés elegir un color y se selecciona una paleta con otros tonos que funcionan juntos, esperamos que te sean útiles a la hora de pensar la imagen visual de tu proyecto:

<https://www.materialpalette.com/>

<https://coolors.co/337ca0-3ec300-fffc31-ff1d15-e13700>



## ¿Qué unidades de medida debo usar en mi proyecto?

Las más usadas hoy en día y mas recomendadas son:

- Para anchos y altos de imágenes (width y height): **píxeles (px)** o **porcentajes (%)** según la necesidad.
- Para anchos y altos de contenedores: **Porcentajes (%)**
- Para textos: **Rem.**

## Clase 4

### Además del método BEM y camelCase ¿Hay otros métodos para nombrar clases o IDs?

Si, existen otras como kebab-case o Train-Case. Acá puede investigar un poco más:

<https://adrianalonsodev.medium.com/convenci%C3%B3n-de-nombres-desde-el-camelcase-hasta-el-kebab-case-787e56d6d023>

### ¿Cuál es la diferencia entre margin y padding?

La diferencia es que **margin** refiere al espacio hacia afuera de las cajas y **padding**, en cambio, al espacio hacia adentro de las cajas.

Para entenderlo mejor, es recomendable que mires el siguiente video que las muestra en correcto funcionamiento:

<https://drive.google.com/file/d/1cpqFRND0VyzNc-65o4LQOzLCICf2KCUA/view?usp=sharing>

### ¿Cuáles son los elementos de bloque y cuáles los de línea?

Los elementos de **línea** son aquellos que ocupan únicamente el espacio que necesitan dentro de la pantalla, y por esta razón se colocan uno al lado del otro, ejemplo **<span>** 🖱️

Los elementos de **bloque**, son los que ocupan todo el ancho de la pantalla, y todo lo que escribas luego se va a ir hacia abajo de estos.

Ejemplo: **<div>**, **<h1>**, **<p>**. 🖱️

Te dejamos el link a la presentación con el listado de cada uno:

[https://docs.google.com/presentation/d/1JV\\_mdo\\_iLHnoilNDHP67Zk5LwVlj6xUncaq8BE81sOE/edit#slide=id.gf309bfa2be\\_0\\_554](https://docs.google.com/presentation/d/1JV_mdo_iLHnoilNDHP67Zk5LwVlj6xUncaq8BE81sOE/edit#slide=id.gf309bfa2be_0_554)

### ¿Dónde puedo ver más sobre el tema de position?

Este recurso te permitirá entenderlo mejor:

<https://developer.mozilla.org/es/docs/Web/CSS/position>

Sin embargo, lo mejor para entenderlo es practicar, practicar y practicar. A veces es necesario romper el código para entender cómo arreglarlo ¡Hacelo sin miedo!

## ***Clase 5***

### **¿Qué quiere decir que flexbox sea una “caja flexible”?**

Se llama así porque todos los elementos y etiquetas HTML se comportan como una caja, es decir, dentro de ellas colocas lo que quieres que contengan y con flexbox les indicas cómo deben comportarse y moverse. Ya sea que quieras que se pongan en columna o en filas, centradas, a la izquierda o a la derecha, con flexbox lo puedes hacer.

Te compartimos un video tutorial en dónde explicamos esto en detalle:

[https://drive.google.com/file/d/120NYITrIb8-4ISj4hjtTtE\\_M2qxDIXBu/view?usp=sharing](https://drive.google.com/file/d/120NYITrIb8-4ISj4hjtTtE_M2qxDIXBu/view?usp=sharing)

### **¿Siempre que use flexbox debo usar todas las propiedades que vimos?**

No, solo deberás usar las que precises para que tu proyecto se vea como quieres.

### **¿Por qué se dice que debemos dejar de usar Float?**

Si bien Float solucionó algunos problemas por un tiempo, comenzó a generar otros. Entre ellos, comportamientos extraños dependiendo del navegador y dificultad para posicionar y centrar. Con la llegada de Flexbox se solucionaron todos estos problemas, combinando menos código con más soporte de los navegadores, por lo tanto, Float está a un paso de quedar obsoleto. Es una buena práctica evitar su uso.

### **¿Si uso Flexbox ya no debería usar display block, inline o inline-block?**

Flexbox no choca con estas propiedades, por lo tanto podrías por ejemplo, usar flexbox en un div, y block en otra etiqueta. A diferencia de

Float, block, inline e inline-block siguen en vigencia. ¡Usa la que mejor se adapte a cada sección de tu proyecto!

### **¿Hay algún juego o sitio interactivo donde pueda practicar Flexbox?**

Si bien la mejor forma de entenderlo es practicarlo en tu código, te dejamos estos divertidos juegos que pueden ayudarte a internalizar el conocimiento más rápido. ¿Estás listo? ⚡

<https://flexboxfroggy.com/#es>

<http://www.flexboxdefense.com/>

## ***Clases 6***

### **¿Qué es CSS Grid?**

Es una tecnología de CSS que, junto con Flexbox, sirven para organizar todos los elementos de nuestro sitio web. Como ya vimos, Flexbox es una “caja flexible” y Grid un sistema de grillas formadas por filas y columnas, algo así como una tabla de Excel super poderosa. 🦸

En la siguiente ppt te lo explicamos más profundamente:

[https://docs.google.com/presentation/d/1Re3G4IKPsCubVg1QvzbhQx9nT1nRkeEmkTORT4a1-Qg/edit#slide=id.gefbddf7b0e\\_0\\_116](https://docs.google.com/presentation/d/1Re3G4IKPsCubVg1QvzbhQx9nT1nRkeEmkTORT4a1-Qg/edit#slide=id.gefbddf7b0e_0_116)

### **¿Cuál es la diferencia entre Grid y Flexbox?**

La diferencia principal es que con flexbox se logra posicionamiento unidimensional (una dimensión), es decir, solo arriba y abajo o a los lados, y grid, en cambio, es bidimensional (dos dimensiones). Esto significa que nos permite manipular los elementos en cualquier dirección y además moverlos a cualquier lugar de tu sitio. Podés ver un ejemplo ingresando a la siguiente presentación:

[https://docs.google.com/presentation/d/1Re3G4IKPsCubVg1QvzbhQx9nT1nRkeEmkTORT4a1-Qg/edit#slide=id.gefbddf7b0e\\_0\\_152](https://docs.google.com/presentation/d/1Re3G4IKPsCubVg1QvzbhQx9nT1nRkeEmkTORT4a1-Qg/edit#slide=id.gefbddf7b0e_0_152)

Si necesitás más información, podés ingresar a la siguiente fuente confiable:

[https://developer.mozilla.org/es/docs/Web/CSS/CSS\\_Grid\\_Layout/Relationship\\_of\\_Grid\\_Layout](https://developer.mozilla.org/es/docs/Web/CSS/CSS_Grid_Layout/Relationship_of_Grid_Layout)

### **¿Si uso flexbox puedo usar grids o viceversa?**

Totalmente, estas no son tecnologías que choquen entre sí, al contrario, pueden convivir perfectamente para el mejor funcionamiento de tu proyecto.

### **¿Cómo sé cuándo usar grid-areas, fraction, y demás métodos de CSS Grid?**

Generalmente, depende del tamaño de lo que quieras hacer. Si quieres posicionar los elementos de forma general o macro, te conviene usar áreas y podrás lograr algo como esto:

[https://docs.google.com/presentation/d/1Re3G4IKPsCubVg1QvzbhQx9nT1nRkeEmkTORT4a1-Qg/edit#slide=id.gefbddf7b0e\\_0\\_198](https://docs.google.com/presentation/d/1Re3G4IKPsCubVg1QvzbhQx9nT1nRkeEmkTORT4a1-Qg/edit#slide=id.gefbddf7b0e_0_198)

Y las fracciones (fraction) cuando quieras hacer algo más pequeño como una sección en particular, por ejemplo, una galería de imágenes. A medida que vayas desarrollando y practicando, te darás cuenta más rápidamente qué se amolda mejor para lograr tu objetivo.

### **¿Debo usar absolutamente todas las propiedades vistas en la clase para resolver el desafío?**

No, mínimamente lo que se pide en la ppt del mismo, si quieres agregar algo más será bien recibido. Recuerda que ante cualquier duda de este tipo debes consultar con tu profe o tu tutor/a.

### **¿Dónde puedo practicar Grid antes de aplicarlo en mi proyecto?**

Si quieres verlo aplicado de forma más realista, puedes hacerlo desde acá: <https://grid.layoutit.com/>

Si quieres hacerlo de forma más divertida, puedes practicarlo con este simpático juego: <http://cssgridgarden.com/#es>

## ***Clase 7***

### **¿Por qué debería desarrollar Mobile First?**

Hoy día el desarrollo Mobile First no solo es una buena práctica, sino que es indispensable. Si nos ponemos a pensar, ¿Desde qué tipo de dispositivos accedemos habitualmente a internet? Pues si, desde nuestros celulares, por lo tanto, debemos darle una prioridad especial a que nuestros sitios funcionen bien en celulares.

Te recomendamos esta lectura complementaria sobre el tema:

<https://blackpooldigital.com/blog/mobile-first-que-es-y-por-que-implementarlo-en-tu-web>

### **¿Tengo que rehacer todo mi proyecto y ponerle a todo Media Queries?**

No, las media queries deberás aplicarla únicamente donde las requieras. Por ejemplo, si para pantallas de celulares algunos de tus elementos se ven unos debajo de los otros, en las media queries solo deberás modificar lo necesario para que se pongan uno al lado del otro. Te dejamos un artículo interesante de nuestro foro escrito por uno de nuestros tutores:

<https://plataforma.coderhouse.com/foro//media-queries-y-mobile-first>

### **¿Qué medidas de pantallas debo usar para mis media queries?**

Si bien no existe un estándar 100% definido lo más comunes son 768px, 992px, 1024px y 1200px. Sin embargo, puede que sea necesario que uses tamaños más chicos o más grandes según la necesidad de tu proyecto. Te dejamos por acá una tabla con los breakpoints más usados:

[https://docs.google.com/presentation/d/1k4STrf27eZOpQSBorLU-sftNDaTiIJAKFW2hddDDSF8/edit#slide=id.gf30eb74637\\_0\\_149](https://docs.google.com/presentation/d/1k4STrf27eZOpQSBorLU-sftNDaTiIJAKFW2hddDDSF8/edit#slide=id.gf30eb74637_0_149)

### **¿En qué parte de mi CSS debo escribir las media queries?**

Hoy día se ven dos formas:

1. Acumularlas y escribirlas todas al final de tu CSS.
2. Escribirlas justo debajo del elemento que quieras modificar con media queries.

Si bien puedes ubicarlas donde mejor te parezca, por temas de buena práctica y además orden de código, se recomienda colocarlas justo

debajo del elemento que quieras modificar. Esto te permitirá ubicarlas más rápido según el orden en que escribas tu código.

Además, si las colocas al final recuerda que CSS lee tus estilos “en cascada”. Entonces, si las pones todas al final, obligas al navegador a bajar hasta el final de tu archivo de estilos para aplicar lo que corresponda a las media queries. Esto puede hacer que tu sitio cargue algunos segundos más lento, si bien no será mucha diferencia, cada segundo de carga cuenta. 🕒

### **¿Cómo puedo combinar flexbox, grid y media queries?**

Puedes por ejemplo, usar flexbox para que tu galería de imágenes se vea de arriba hacia abajo en celulares, y usar grid para que en tablet y en pantallas más grandes se vean una al lado del otro. Acá te dejamos un pequeño video que te puede ayudar a entenderlo gráficamente:

<https://drive.google.com/file/d/18RP4D4S0vIUe9SbjSauDKhxpibb6iQBf/view?usp=sharing>

## ***Clase 8***

### **¿Qué son las animaciones?**

Las animaciones son una forma de cambiar un elemento de un estado a otro sin necesidad que el usuario interactúe con este, ya sea mediante clic, teclado etc. Se suele utilizar para dar movimiento fluido a algunos elementos o para cambios en colores de fondo infinitos. Por ejemplo, usando una animación podemos lograr que una imagen aparezca “de la nada” de izquierda a derecha y crear una barra de carga de porcentaje. Existen infinidad de animaciones, pero ojo, úsalas con consciencia para que tu sitio web no se vea sobresaturado, como el sitio web de Homero Simpson 😄. Te dejamos más información en los siguientes links:

- <https://developer.mozilla.org/es/docs/Web/CSS/animation>
- [https://www.w3schools.com/css/css3\\_animations.asp](https://www.w3schools.com/css/css3_animations.asp)

### **¿Qué son las transiciones?**

Las transiciones, así como las animaciones, son una forma de cambiar un elemento de un estado a otro. Sin embargo, para la transición es necesaria una interacción de usuario, por ejemplo, que cuando se pose el mouse sobre un link el fondo de este cambie de color. Sin las transiciones el cambio de color sería instantáneo y brusco, en cambio, con ellas podemos definir que este cambio tse demore y sea más amigable al usuario. Te dejamos más información en los siguientes links:

- [https://developer.mozilla.org/es/docs/Web/CSS/CSS\\_Transitions/Using\\_CSS\\_transitions](https://developer.mozilla.org/es/docs/Web/CSS/CSS_Transitions/Using_CSS_transitions)
- [https://www.w3schools.com/css/css3\\_transitions.asp](https://www.w3schools.com/css/css3_transitions.asp)

### **¿Cuáles son las diferencias entre animaciones y transiciones CSS?**

En sí, la más notoria es la necesidad de interacción del usuario. La animación no necesita que el usuario haga alguna acción en el sitio para funcionar, puede ejecutarse sola cuando carga el sitio web, y hasta infinitamente si así lo decidimos. En cambio, las transiciones necesitan alguna interacción del usuario para funcionar, ya sea posar el mouse sobre un elemento (hover), hacer clic, etc.

## ***Clase 9***

### **¿Cómo instalo GIT en Windows?**

Primero debes entrar al sitio oficial de GIT: <https://git-scm.com/>

Una vez que hayas hecho esto, podés seguir el siguiente paso a paso:

[https://docs.google.com/presentation/d/154OiXVU\\_aXjdi-ulf6rv7fIWzFLKzfsIm4wJ74xnlg/edit#slide=id.gf31549e164\\_0\\_183](https://docs.google.com/presentation/d/154OiXVU_aXjdi-ulf6rv7fIWzFLKzfsIm4wJ74xnlg/edit#slide=id.gf31549e164_0_183)

### **¿Cómo instalo GIT en MAC?**

Primero debes entrar al sitio oficial de GIT: <https://git-scm.com/>

Una vez que hayas hecho esto, podés seguir el siguiente paso a paso:

[https://docs.google.com/presentation/d/154OiXVU\\_aXjdi-ulf6rv7fIWzFLKzfsIm4wJ74xnlg/edit#slide=id.gf31549e164\\_0\\_256](https://docs.google.com/presentation/d/154OiXVU_aXjdi-ulf6rv7fIWzFLKzfsIm4wJ74xnlg/edit#slide=id.gf31549e164_0_256)

### **¿Cómo configuro GIT para empezar a usarlo?**

Lo principal que debes tener en cuenta es que debes pensar un nombre de usuario y usar un e-mail, el cual deberá ser el mismo que uses para registrarte en GitHub.

Te recomendamos usar, por ejemplo, tu nombre de usuario en instagram, en twitter o en cualquier otra red social para asegurarte de no olvidarlo.

A continuación, te compartimos la ppt de la clases con el paso a paso de la configuración:

[https://docs.google.com/presentation/d/154OiXVU\\_aXjdi-ulf6rv7fIWzFLKzfsIm4wJ74xnlg/edit#slide=id.gf31549e164\\_0\\_325](https://docs.google.com/presentation/d/154OiXVU_aXjdi-ulf6rv7fIWzFLKzfsIm4wJ74xnlg/edit#slide=id.gf31549e164_0_325)

### **¿Debo escribir git init cada vez que vaya a trabajar sobre mi repositorio ya creado?**

No, el comando git init solo se usa la primera vez que creas un repositorio, luego solo deberás trabajar con add, commit y próximamente con push.

### **¿Para qué sirven git add y git commit?**

- Con git add + el nombre de tu archivo le indicas a git que tome ese archivo para sus modificaciones. Como por ejemplo, **git add index.html**
- Otra forma del add, es escribir simplemente: add . (add, espacio y un punto) con esto le indicamos a git que tome en cuenta todos los archivos que hayan sido modificados sin necesidad de hacer add uno por uno.
- Con git commit escribes un mensaje indicativo sobre los cambios que hiciste, así, además, confirmas los cambios añadidos con el add, por ejemplo:  
**git commit -m "creé mi primer repositorio git"**

En resumen,



## ***Comandos***

### **git init**

Para inicializar tu repositorio (lo escribes una sola vez por proyecto)

### **git add**

Para añadir cambios.

### **git commit**

Para confirmar los cambios con un mensaje indicativo.

### **¿Para qué sirven las ramas de GIT?**

Con las ramas haces una copia de tu rama principal (master o main) dentro del mismo repositorio de GIT, sin necesidad de crear otra carpeta desde tu pc. Así, evitamos las famosas carpetas “Proyecto con GIT”, “Proyecto con GIT 2”, “Proyecto con GIT este si”, “Proyecto con GIT final”. 😞

Creando una rama, podés hacer modificaciones libremente sin que estas afecten a tu archivo principal hasta que vos mismo lo decidas al hacer merge.

### **¿Debo usar GIT si o si desde la interfaz de Bash o puedo usarlo desde mi editor de código?**

Ambas opciones son viables y dependerá de vos elegir la que te resulte más práctica.

### **¿Puedo ver algún paso a paso de todos los comandos aplicados?**

¡Claro que sí! Preparamos un video instructivo que te puede ayudar:

[https://drive.google.com/file/d/1qGGAIMk2cBJt3yQBw4VA\\_nZwKT7-nt8I/view?usp=sharing](https://drive.google.com/file/d/1qGGAIMk2cBJt3yQBw4VA_nZwKT7-nt8I/view?usp=sharing)

## ***Clase 10***

### **¿Puedo usar cualquier email para registrarme en GitHub?**

No, debes registrarte con la misma dirección de email al momento de crear tu usuario de GIT, de lo contrario no va a haber forma de que ambas tecnologías se conecten con tu proyecto.

### **¿Qué es realmente Github?**

Github es una especie de red social para desarrolladores en la cual ellos no solo pueden mostrar sus códigos, sino que, además, permite trabajar en conjunto con otras personas en un proyecto en común, sin importar en qué parte del mundo se encuentran.

Además te servirá a modo de portfolio para demostrar tus habilidades como desarrollador al momento de buscar un trabajo en el rubro. Los reclutadores IT suelen consultar Github al momento de contratar.

### **¿Dónde puedo ver un paso a paso para subir mi repositorio a GitHub?**

Podés volver a revisar la presentación de la clase: [https://docs.google.com/presentation/d/1p7ANbYTbk4Efw3hx6PXX5m-Oa4rSEKKbToQFgf6EpdQ/edit#slide=id.gf4d51016ba\\_0\\_117](https://docs.google.com/presentation/d/1p7ANbYTbk4Efw3hx6PXX5m-Oa4rSEKKbToQFgf6EpdQ/edit#slide=id.gf4d51016ba_0_117)

También preparamos un mini video tutorial por si prefieres verlo en acción 🖥️:

[https://drive.google.com/file/d/1MXt8gzeOFjW-UtH3o\\_fIEn1mBaLhFU4A/view?usp=sharing](https://drive.google.com/file/d/1MXt8gzeOFjW-UtH3o_fIEn1mBaLhFU4A/view?usp=sharing)

## ***Clase 11***

### **¿Qué es Bootstrap?**

Bootstrap es un framework o marco de trabajo desarrollado por Twitter que tiene como objetivo facilitar y optimizar el tiempo en el desarrollo de código, todo esto mediante la publicación y uso gratuito de sus códigos para distintas partes de tus proyectos. Por lo tanto esto incluye códigos HTML, CSS y JavaScript que con solo copiarlos y pegarlos ya puedes utilizarlos. Ojo, el reto está en lograr personalizarlos.

## ¿Cuál es la mejor forma de usar Bootstrap, descargando los archivos o a través del CDN?

Ambas formas son válidas, todo depende de cómo te sientas mejor trabajando. Tené en cuenta lo siguiente antes de tomar una decisión:

1. Si lo descargas y agregas a tu proyecto: Ventaja: Mayor control y te aseguras que siempre funcione. Desventaja: puede aumentar el peso de tu proyecto.
2. Si usas el CDN: Ventaja: No te suma peso a tu proyecto. Es solo agregar links a tu código. Desventaja: Si los servidores de Bootstrap se llegan a caer por un tiempo, tu sitio se quedará sin gran parte de sus estilos. Pasa muy poco pero siempre es una posibilidad latente.

## ¿Por qué cuando linkeo los archivos de Bootstrap se rompen parte de mi proyecto?

Esto suele deberse a dos motivos. El primero es que dentro de los `<head>` de tus archivos html hayas linkeado los archivos de Bootstrap por debajo de tus propios archivos. Recuerda que CSS lee los estilos en cascada, y si linkeas los de Bootstrap después de los tuyos, esos son los últimos que va a aplicar.

Para asegurarte de linkear correctamente, podés usar el siguiente modelo:

```
1 <head>
2   <meta charset="UTF-8">
3   <meta http-equiv="X-UA-Compatible" content="IE=edge">
4   <meta name="viewport" content="width=device-width, initial-scale=1.0">
5   <!------- Link de Bootstrap ----->
6   <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet"
7     integrity="sha384-1BmE4kWBq78iYhF1dvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3" crossorigin="anonymous">
8
9   <!------- Link a tu archivo CSS ----->
10  <link rel="stylesheet" href="css/styles.css">
11  <title>Tu proyecto</title>
12 </head>
```

Y la segunda razón, es que Bootstrap tiene estilos predefinidos para muchas cosas, por ejemplo, tiene tipo de fuente predeterminado, tamaños de fuente específicos, márgenes o paddings propios, etc. La mejor forma de asignar tus propios estilos es haciéndolo dentro de tu propio CSS y así los lograrás resetear.

### **Linkeé el archivo de Bootstrap antes que mi propio CSS y los estilos que yo asigno no se reflejan ¿Qué puede estar pasando?**

Así como Bootstrap tiene muchos estilos predefinidos, también tiene algunos `!important` incluidos, estos podrían estar yuxtapuestos a tus estilos.

La mejor forma de localizar este problema sería: en el navegador con tu proyecto abierto, abrir el inspector presionando F2 y allí “rastrear” el elemento que desees personalizar y fijarte si tiene algún `!important` de Bootstrap que pueda estar afectando.

Usando el inspector del navegador, podemos ver mucho más gráficamente el funcionamiento de nuestro código y así reparar errores que se nos pueden estar pasando por alto en nuestro editor de código.

### **¿Qué versión de Bootstrap debo usar?**

Las más modernas y usadas hoy día son las versiones 4.6 y la 5.1. Sin embargo, uses la que uses, es importante que si decides usar la 5.1, los links, códigos o archivos descargados que agregues a tu proyecto deben ser todos de esta versión. Por ejemplo, si tomas un código html de la versión 5.1 pero usas los CSS y los JavaScript de la 4.6, no te va a funcionar. Intenta que todo corresponda siempre a la misma versión.

### **¿Qué quiere decir que Bootstrap funcione bajo un sistema de grillas?**

Esta afirmación simplemente quiere decir que Bootstrap usa CSS Grid para que puedas ubicar tus elementos usando las clases `row` o `col` con sus determinados sufijos.

Ten en cuenta que esta grilla consiste en 12 columnas, así que todos los elementos donde vayas a agregar estas clases deben sumar 12.

Podés profundizar consultando el material complementario de clase:  
[https://docs.google.com/document/d/1NIPra3pMPxplkSugb9yJH8\\_NC\\_Nx-8AUkSDfqWWAaVBE/edit#heading=h.pm90vwgtm1m3](https://docs.google.com/document/d/1NIPra3pMPxplkSugb9yJH8_NC_Nx-8AUkSDfqWWAaVBE/edit#heading=h.pm90vwgtm1m3)

### **¿Si o si debo usar el sistema de grillas de Bootstrap?**

No necesariamente. Puedes usar solo el código que creas necesario, no es obligatorio usarlo todo.

### **¿Dónde puedo ver todas las clases que usa Bootstrap?**

Acá te dejamos un cheatsheet con el listado de clases y para qué sirve cada una: <https://bootstrap-cheatsheet.themeselection.com/>

También te compartimos un link con los componentes donde puedes verlos en funcionamiento junto con su documentación oficial:

<https://getbootstrap.com/docs/5.0/examples/cheatsheet/>

## ***Clase 12***

### **¿Debo usar media queries propias aunque use Bootstrap para hacer mi sitio responsive?**

Seguramente habrá algún contenido que no puedas acomodar como quieras usando los breakpoints de Bootstrap y debas acudir a crear una propia. Todo esto dependerá de cómo tengas pensado tu proyecto.

### **¿Cuándo debo usar min-width y cuándo max-width en mis proyectos?**

- Usando min-width indicas a tu proyecto que cuando la pantalla tenga mínimamente el tamaño que le pongas, haga determinada cosa, es decir, de aquí para arriba.

Por ejemplo: Si usas min-width: 768px, estas diciendo que cuando la pantalla mida un mínimo de 768px haga algo, es decir, desde 768 para arriba.

- En cambio usando max-width indicas que tu sitio haga algo **hasta** que la pantalla mida el ancho que le pongas.

Usando el mismo ejemplo anterior si indicas max-width: 768px, indicas que tu sitio haga algo hasta que la pantalla mida un máximo de 768px, es decir, servirá para tamaños desde 0 hasta 768px. Ya de 769px para arriba deberás indicar otro comportamiento.

### **¿Qué son las pseudoclasses?**

Las pseudoclasses se usan para dar ciertos estilos y funcionamientos a tus elementos, dependiendo del estado en que estén. Como por ejemplo: cuando posas tu mouse sobre un enlace quieres que el fondo cambie de color (pseudoclase :hover); otro ejemplo, cuando dentro de un grupo de elementos solo quieres darle estilos a los pares o impares.

### **¿Hay algún video tutorial que recomienden para ver las pseudoclasses en funcionamiento?**

Si, te dejamos un video tutorial grabado por uno de los tutores de nuestro staff:

<https://drive.google.com/file/d/1S3OFr12UenfsWoMmlBAhtEjDKFgAtGCt/view?usp=sharing>

### **¿Cuántas pseudoclasses existen?**

Son muchas, y cada una con una utilidad y funcionamiento distinto. Conocerlas te permitirá elegir la que mejor se adecúe a lo que necesitas lograr.

Te compartimos un listado con las distintas pseudoclasses existentes:

<https://developer.mozilla.org/es/docs/Web/CSS/Pseudo-classes>

En el siguiente enlace podrás acceder a una explicación interactiva:

<https://lenguajecss.com/css/selectores/pseudoclasses/>

## ***Clase 13***

### **¿Por qué tendría que usar SASS si ya existe CSS?**

La sintaxis de SASS está programada de forma que puedas optimizar mucho tu escritura de código. Esto porque te permite, por ejemplo, definir una sola vez algún código que vayas a usar repetidas veces

como colores de fuentes o de fondos, tamaños de fuentes, y hasta varias líneas de código como cuando alineas con CSS, y usarlos solo colocando alguna palabra clave.

Una vez que usas esta palabra clave en alguna parte, esta se replicará en tu archivo CSS de una forma legible.

Acá te dejamos el link al sitio oficial de SASS donde verás información 100% oficial: <https://sass-lang.com/>

### **¿Qué necesito para comenzar a usar SASS?**

En principio, se requiere que instales node.js en tu computadora. Acá te dejamos el paso a paso:

[https://docs.google.com/presentation/d/19urJmC3Z\\_-aVHLTV8Gu0R0I1MUlIFUGMRa3kWrF\\_ill/edit#slide=id.gfed2a768a\\_0\\_166](https://docs.google.com/presentation/d/19urJmC3Z_-aVHLTV8Gu0R0I1MUlIFUGMRa3kWrF_ill/edit#slide=id.gfed2a768a_0_166)

### **¿Cómo comienzo a trabajar con SASS?**

Una vez que ya hayas instalado node.js tendrás que hacer el siguiente paso a paso:

[https://docs.google.com/presentation/d/19urJmC3Z\\_-aVHLTV8Gu0R0I1MUlIFUGMRa3kWrF\\_ill/edit#slide=id.gfed2a768a\\_0\\_166](https://docs.google.com/presentation/d/19urJmC3Z_-aVHLTV8Gu0R0I1MUlIFUGMRa3kWrF_ill/edit#slide=id.gfed2a768a_0_166)

También te dejamos este video tutorial por si quieres ver el paso a paso en acción:

<https://drive.google.com/file/d/1h78SEaTE1MqAPUqqZsjaGaiYtD3peNWh/view?usp=sharing>

### **¿Tengo que reescribir todo mi código CSS dentro de SASS para que pueda funcionar?**

No. Por la instancia del curso en la que estamos, podrías comenzar copiando el código CSS más actualizado que tengas y pegarlo en un archivo principal de SASS. Luego, de a poco, ir usando solo lo que requieras de SASS para optimizar tu tiempo de código.

**La consola se me puso toda roja como con mensajes de error y no me compiló SASS ¿Qué puede estar pasando?**

En principio, la consola de SASS te puede generar errores por dos razones principales. La primera, si tienes el autoguardado de Visual Studio Code, esto hace que todo cambie se guarde en tiempo real y se va actualizando con cada tecla que presiones. Esto hará que te vaya dando error hasta que termines de escribir correctamente.

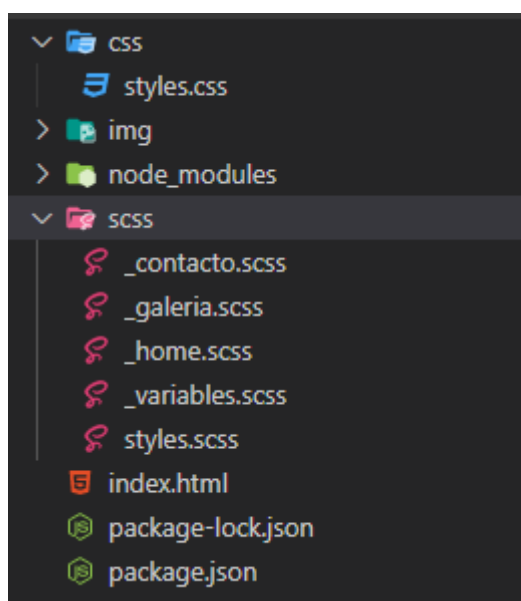
La segunda, es que sin querer hayas escrito mal algo, ya sea una llave de cierre, un punto y coma que te falte o alguna propiedad mal escrita.

Sea cual sea el error no te preocupes, no te dejes alarmar por el color rojo de la consola. Acá te dejamos una presentación donde te enseñamos cómo detectar la razón de los errores:

<https://docs.google.com/presentation/d/1BIHBoTDi-zVIMMiWLJPqEDzVleLsE8eV/edit#slide=id.p1>

### **¿Puedo crear varios archivos SASS por cada una de mis secciones?**

En el caso de SASS, se considera una buena práctica tener archivos de estilos separados que te van a permitir trabajar más organizadamente. Solo debes tener en cuenta que es necesario tener un SASS principal (el que indicas en el archivo json que modificas al principio), y este es el que se va a conectar directo con tu archivo CSS único. El resto puedes separarlo por secciones, variables, utilidades, o de la forma que te sientas mejor trabajando. Abajo te dejamos un ejemplo:





En este caso, el archivo principal es el `styles.scss`, y los demás, están separados por los estilos para las secciones, y uno de variables útiles. Ten en cuenta que los archivos individuales deben llevar un guión bajo antes del nombre, es decir: `_variables.scss`, `_galeria.scss`, etc.

Para este caso, nuestro archivo scss principal quedará así:



Con los `@import` indicamos que recopile o tome todos los estilos que están en los demás archivos y los compile en nuestro CSS principal. Ten en cuenta que todos los archivos de utilidades como las variables, deberán ir siempre **antes** que los estilos de tus secciones. Esto porque todas las secciones de tu sitio se alimentarán de ellas (colores de fuentes, tamaños, márgenes, paddings, etc). Si las pones al final o al medio se te puede romper todo, y eso no es lo que queremos para ti (recuerda el funcionamiento en Cascada de CSS).

## ***CLASE 14***

**Veo que en las diapositivas de la clase nombran mucho el SASS de Bootstrap. ¿Es obligatorio usarlo en mi proyecto?**

No, no es obligatorio. La intención de enseñarte esto es que veas como tecnologías tan poderosas como Bootstrap usan SASS para construirse de una forma más ordenada.

Si te sientes bien con la idea de trabajar con los SASS de Bootstrap, puedes intentarlo pero no será obligatorio para aprobar tus desafíos. Cualquier duda, consulta siempre con tu profe o tutor, recuerda que ellos llevan el liderazgo de tu comisión.

### **¿Hay algún límite para el uso de lo visto en esta clase?**

No, puedes usar todas las variables, condicionales, maps, each, extends y mixins que requieras siempre que te ahorren tiempo y te hagan más fácil y amigable la tarea de escribir código CSS a través de SASS.

### **No termino de entender cómo aplicar todos estos conocimientos a mi proyecto.**

No pasa nada, a veces SASS puede parecer un poco abrumador. Con un poco de práctica te irá saliendo.

Por acá te dejamos un video tutorial por si quieres ver las distintas operaciones y formas de funcionamiento de SASS implementado en tiempo real:

[https://drive.google.com/file/d/1p25al6Z\\_TSMZv8pr\\_ANKUuba8Ga-xVj9/view?usp=sharing](https://drive.google.com/file/d/1p25al6Z_TSMZv8pr_ANKUuba8Ga-xVj9/view?usp=sharing)

## ***CLASE 15***

### **¿Para qué me sirve toda esta información sobre servidores y seguridad?**

En principio, debemos recalcar que internet es un espacio totalmente público, por lo tanto, así como puedes encontrarte cosas muy buenas, puedes encontrarte con cosas muy malas que pueden afectar de distintas formas que van desde, dañarte tu computadora o celular, o robar tus datos sensibles como número de documento, cuentas bancarias o tarjetas de crédito.

Al momento que desarrollas un sitio web ya sea para ti o para un cliente, debes asegurarte que dicho sitio cumpla con todas las medidas necesarias para evitar el ingreso de virus o robo de datos.

Para esto es necesario que sepas cómo funciona internet y por qué es tan importante blindar a través de seguridad nuestros sitios.

Te dejamos el material complementario donde podrás encontrar más información:

<https://docs.google.com/document/d/1WKJMaCe--TO5zASIXylan9PWGgXRZSw7DE-RbglSQ3U/edit#>

### **Las imágenes hacen que mi sitio cargue muy lento ¿Cómo puedo reducirles el peso y que no pierdan calidad?**

Para lograr esto tienes 2 opciones, ambas válidas y dependerá totalmente de tu preferencia.

**La primera opción:** Optimizar las imágenes a través de herramientas, entre las cuales hay algunas online, donde puedes subir tus imágenes y te las descarga con mucho menos peso y poca o nula pérdida de calidad, entre las cuales están:

Aplicaciones de escritorio como Photoshop o <https://imageoptim.com/api>.

Entre las herramientas online más populares están:

<https://tinypng.com/> y <https://squoosh.app/>

**La segunda opción:** Subir tus imágenes a un servidor gratuito como <https://imgbb.com/> y usar el enlace que este te genera dentro del atributo src de tus imágenes. Con esto te evitas tener las imágenes dentro de las carpetas de tu proyecto.

Esta forma es parecida a insertar un video de YouTube con <iframe>. Funciona y no quita espacio de tus carpetas.

### **¿Para qué sirve el SEO?**

El SEO u Optimización de Motores de Búsqueda por sus siglas en inglés, sirve para que tu sitio web poco a poco se vaya posicionando bien en los motores de búsqueda. Por ejemplo, para que tu sitio pueda

estar en la página 1 del buscador de Google debe tener un muy buen nivel de SEO entre otras cosas.

### **¿Cómo logro un buen posicionamiento SEO?**

Entre las cosas más importantes para hacer está el escribir código de forma correcta, respetando lo más posible la semántica y la accesibilidad web que hemos visto en clases anteriores.

Te dejamos el link al material complementario donde hay información para que profundices mucho más:

<https://docs.google.com/document/d/1WKJMaCe--TO5zASIXylan9PWGgXRZSw7DE-RbgISQ3U/edit#heading=h.s9otq5e207r7>

## ***CLASE 16***

### **¿Debo contratar un hosting pago para mi proyecto final?**

No, no es obligatorio que pagues nada en este momento. De hecho, no recomendamos que lo hagas hasta tanto tengas la suficiente práctica o necesites alojar un sitio real de un cliente o alguna empresa.

Por lo pronto puedes usar webhost, Github Pages o cualquier hosting gratuito de tu preferencia.

### **¿Qué debería tener en cuenta al momento de querer contratar un hosting?**

Más allá del precio, entre las cosas más importantes que deberías tener en cuenta están: el espacio que te brinden, que te incluyan el certificado de seguridad SSL, y que tengan atención de soporte las 24hs al momento que te presente algún problema.

Algunos también te incluyen el dominio de tu sitio web, pero ya allí está en tu decisión si contratarlo con el hosting o con el ente de control de esto en tu país, en argentina es <https://nic.ar/>

### **Además de webhost y GitHub Pages ¿Qué otros hosting gratuitos recomiendan?**

Te recomendamos:

Netlify: <https://www.netlify.com/>

Vercel: <https://vercel.com/>

Si bien hay muchos más, estos dos son muy amigables y además son los hosting gratuitos más seguros hoy día.

## ***CLASE 17***

### **¿Cuánto debo cobrar como freelance a un cliente para desarrollar un sitio web con los conocimientos que adquirí en este curso?**

La realidad es que no podemos ponerle precio a tu trabajo y a tu dedicación, sin embargo, te recomendamos poner un precio realista por hora y con base en eso, estudiar lo mejor posible cuántas horas tardarás aproximadamente en terminar el proyecto y allí hacer tu cuenta.

### **Si busco trabajo en alguna empresa ¿A qué puestos puedo aspirar con los conocimientos adquiridos en este curso y cuál podría ser mi sueldo base?**

Con los conocimientos adquiridos acá y siempre y cuando te mantengas practicando y en movimiento puedes aspirar a cargos como Maquetador(a) Jr o Front End Developer Trainee o Junior. Sin embargo, recuerda que depende mucho de la necesidad de la empresa y de tus ganas de aprender y crecer.

En cuánto al sueldo, es variable dependiendo de cada empresa. Como referencia puedes ingresar al portal de Glassdoor según tu país, y allí filtrar por empresas o por cargos y así ver en promedio cuánto están pagando en la industria en puestos que coincidan con tus habilidades.

El sitio de Glassdoor para Argentina es:  
<https://www.glassdoor.com.ar/index.htm>

### **¿Qué sitios web puedo usar para buscar trabajo freelance?**

Hay muchos, pero entre los más destacados están:

<https://www.upwork.com/>

<https://www.workana.com/es/> y,  
<https://www.freelancer.com.ar/>

## ***Clase 18***

### **Ya terminé este curso ¿Cuál seguiría el siguiente paso para continuar con mi formación?**

Lo siguiente a aprender si quieres crecer en el mundo del Desarrollo Web Front End sería JavaScript. Es un lenguaje de programación muy usado, muy potente y por lo tanto muy buscado a nivel laboral.

Recuerda que en Coderhouse tenemos nuestro propio curso de JavaScript.

Antes de arrancarlo, practica un poco más sobre lo aprendido en este curso y sobre todo, repasa o aprende lo básico de fundamentos de la programación. Te vendrá muy bien.

### **¿Cómo se verá mi proyecto final una vez que esté en línea?**

Te invitamos a ver algunos proyectos de nuestros estudiantes en nuestro sitio web para que puedas analizarlos y/o inspirarte:  
<https://www.coderhouse.com/proyectos>

Además, preparamos un análisis en profundidad en video de uno de esos proyectos. Nos enfocamos en ver qué uso la autora de lo aprendido para realizar su proyecto:

[https://drive.google.com/file/d/1nSfADeLIm5cbgjOGYN\\_rlmfMZ3x29kFK/view?usp=sharing](https://drive.google.com/file/d/1nSfADeLIm5cbgjOGYN_rlmfMZ3x29kFK/view?usp=sharing)