A graphic on the left side of the slide. It features a 3D effect with four overlapping rectangular blocks in purple, orange, yellow, and blue. The text 'Agencia de Aprendizaje a lo largo de la vida' is written across these blocks in white. An orange arrow points to the right from the orange block.

Agencia de  
Aprendizaje  
a lo largo  
de la vida

# FULL STACK JAVA

## Clase 1

VSCoDe y HTML

# Les damos la bienvenida

Vamos a comenzar a grabar la clase

# Temas a tratar:

¿Qué es VSCode (Visual Studio Code)?

- Lenguajes: Maquetación y programación
- ¿Qué es Electron?
- ¿Qué es un Framework?
- ¿Qué es una librería?

Instalación de Visual Studio Code

- En Windows/Mac y Linux
- Extensiones

¿Qué es HTML?

- Estructura semántica del lenguaje
- ¿Qué es una etiqueta?
- Etiqueta <head>
- Etiqueta <body>
- Etiquetas básicas
- Atributos

# ¿Qué es Visual Studio Code?

**Visual Studio Code es un editor de código (fuente) publicado en el año 2015 y desarrollado en Electron, creado inicialmente para HTML y Javascript, hoy soporta la mayoría de los lenguajes de programación (Java, Javascript, Python, Golang, Rust, Typescript, Solidity, entre otros).**

# Lenguajes: Maquetación y Programación

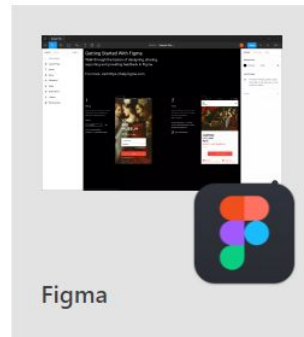
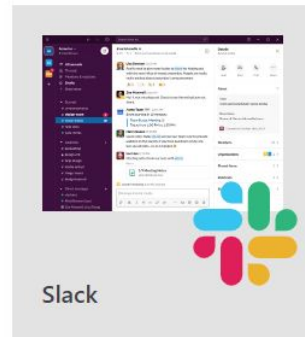
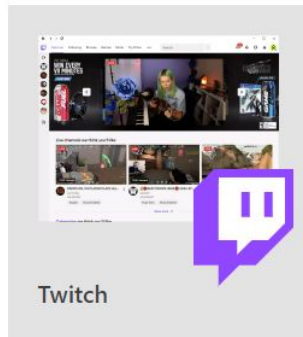
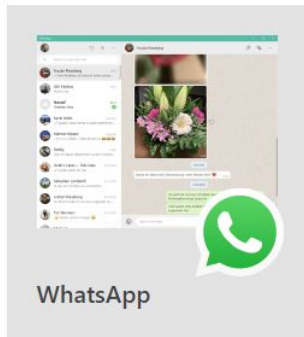
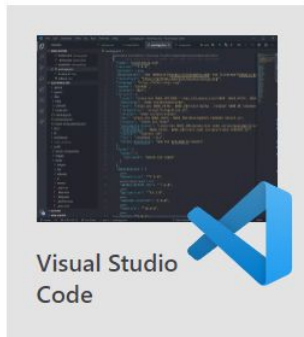
**Maquetación:** Permiten crear un aspecto visual mediante código simple, con acceso a algunas interacciones por parte del usuario (ej. *HTML, CSS*).

**Compilados:** Lenguajes que pueden convertir tu código escrito a código binario, lenguaje de máquina (ej. *C#, C++, Rust, F#, Go, Java, Cobol, Pascal, Fortran, Basic (inicialmente), Visual Basic*)

**Interpretados/Transpilados:** Lenguajes que necesitan de un intérprete, un mediador para entender tu código escrito (Ej: *Python, Javascript, Typescript, Ruby, Perl, Erlang, PHP, Basic*).

# ¿Qué es Electron?

**Electron Js es un framework para Javascript que también puede hacer uso de HTML y CSS. Entre los productos realizados con Electron podemos encontrar a:**



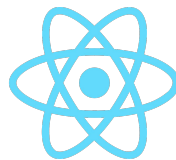
# ¿Qué es un Framework?



Un framework es:

- un marco de trabajo
- una estructura o una manera de proceder con el lenguaje
- unas reglas que debemos respetar a cambio de determinados beneficios, como tener un producto más escalable, más ordenado, más comprensible para otros usuarios que manejen el mismo framework.

# ¿Qué es una Librería?



anime

**Una librería es:**

- **un conjunto de implementaciones funcionales**
- **una herramienta de desarrollo, con fines varios**
- **una forma de simplificar tareas para aumentar la productividad**



# Visual Studio Code Instalación

[Link al sitio de descarga](#)

[¿Qué sistema operativo de Windows tengo?](#)



Version 1.69 is now available! Read about the new features and fixes from June.

# Code editing. Redefined.

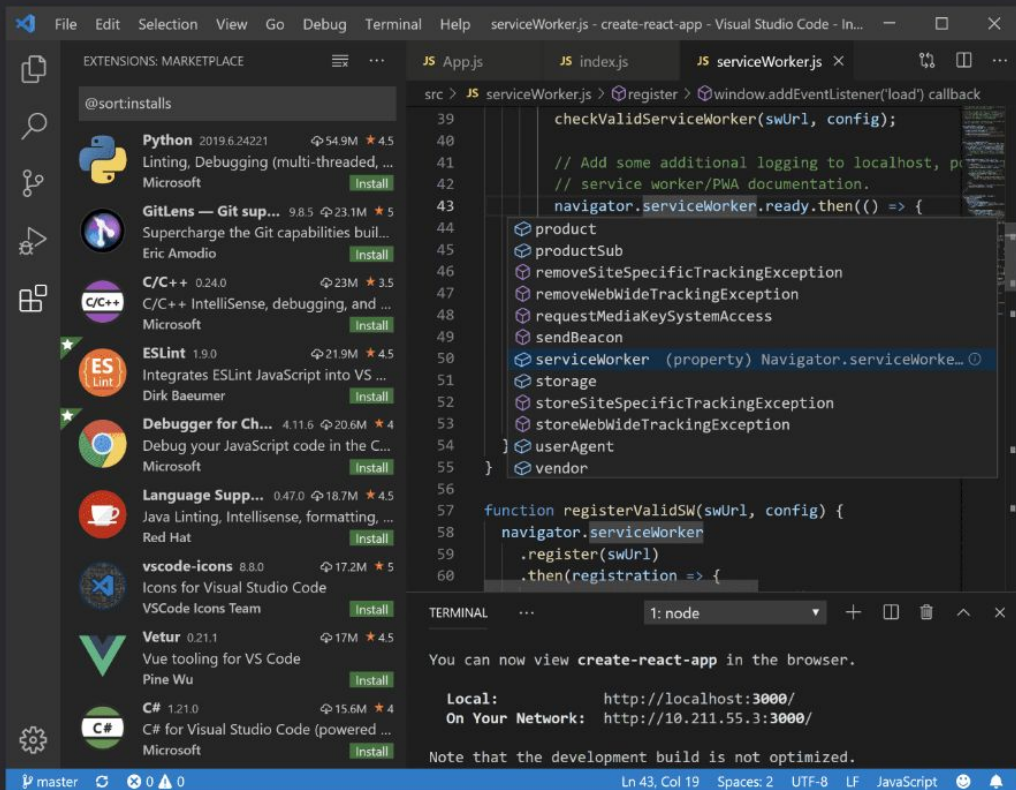
Free. Built on open source. Runs everywhere.

## Download for Windows

Stable Build

Stable Insiders

		Stable	Insiders
macOS	Universal	<a href="#">↓</a>	<a href="#">↓</a>
Windows x64	User Installer	<a href="#">↓</a>	<a href="#">↓</a>
Linux x64	.deb	<a href="#">↓</a>	<a href="#">↓</a>
	.rpm	<a href="#">↓</a>	<a href="#">↓</a>
Other downloads or open on web			



Version 1.69 is now available! Read about the new features and fixes from June.

## Stable: Versión estable (recomendada)

Insider: Versión beta, con más novedades pero con más probabilidad de errores (no recomendada)



Free. Built on open source. Runs everywhere.

Download for Windows  
Stable Build

Debian/Ubuntu  
Red Hat/Fedora

macOS Universal

Windows x64 User Installer

Linux x64 .deb .rpm

Other downloads  
or open on web

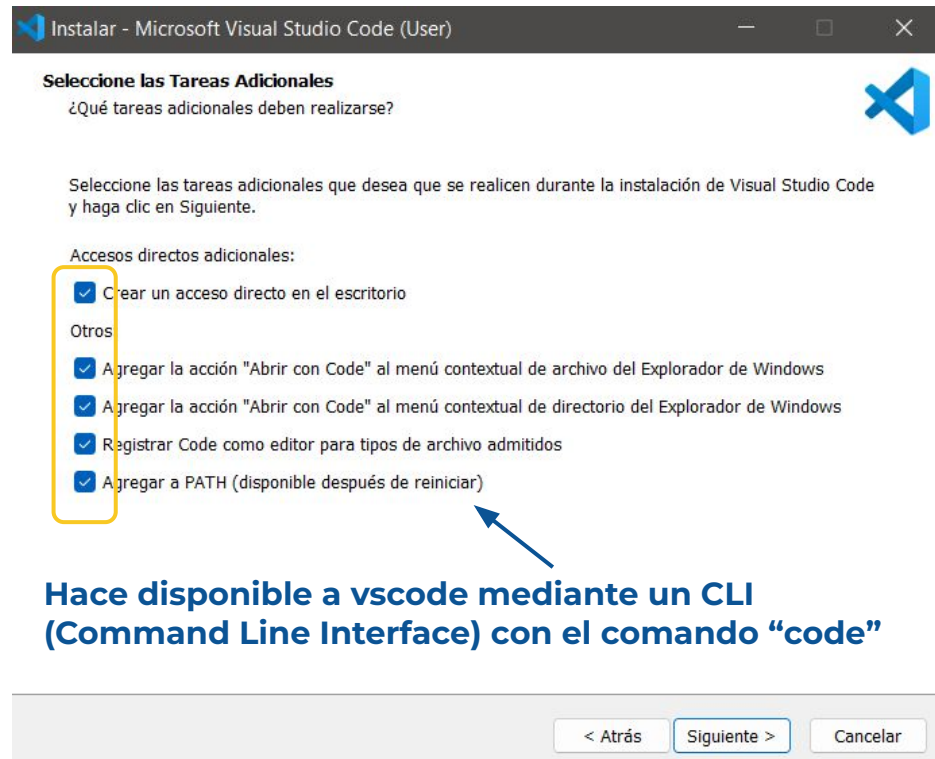
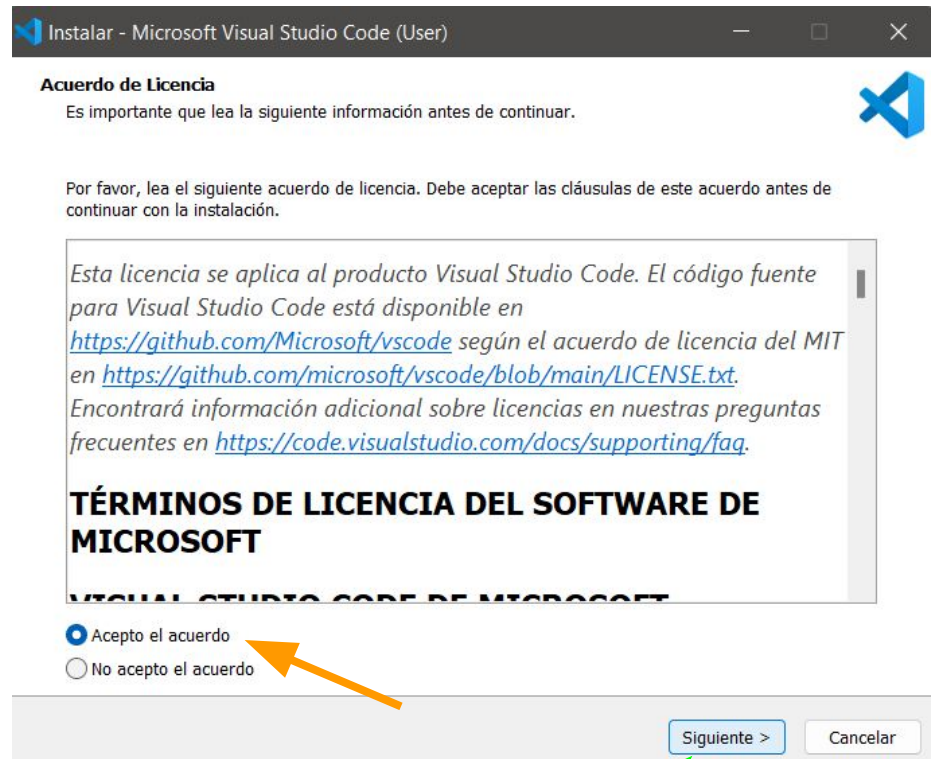
```
sudo dpkg -i <archivo_descargado>  
Ej: sudo dpkg -i code_1.69.2.1658162013_amd64.deb
```

**sudo (Super User DO):** Brinda permisos de “Administrador” de manera temporal, de super usuario para poder instalar aplicaciones en Linux.

**dpkg:** Gestor de paquetes que permiten el control de instalación, compilación, configuración, información y eliminación de archivos en el sistema.

**-i:** Install (Instalar)

# VSCode: Instalación



# VSCode: Instalación



## Listo para Instalar

Ahora el programa está listo para iniciar la instalación de Visual Studio Code en su sistema.



Haga clic en Instalar para continuar con el proceso o haga clic en Atrás si desea revisar o cambiar alguna configuración.

### Tareas Adicionales:

#### Accesos directos adicionales:

Crear un acceso directo en el escritorio

#### Otros:

Agregar la acción "Abrir con Code" al menú contextual de archivo del Explorador de Windows

Agregar la acción "Abrir con Code" al menú contextual de directorio del Explorador de Windows

Registrar Code como editor para tipos de archivo admitidos

Agregar a PATH (disponible después de reiniciar)



## Completando la instalación de Visual Studio Code

El programa completó la instalación de Visual Studio Code en su sistema. Puede ejecutar la aplicación utilizando los accesos directos creados.

Haga clic en Finalizar para salir del programa de instalación.

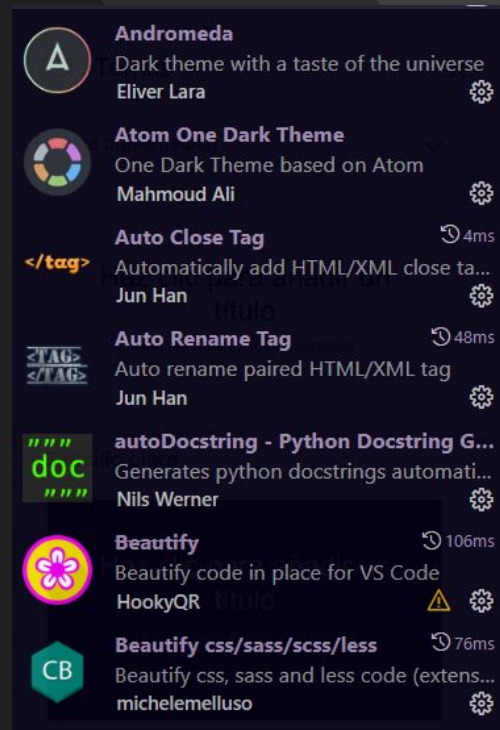


☒ Ejecutar Visual Studio Code

< Atrás **Instalar** Cancelar

**Finalizar**

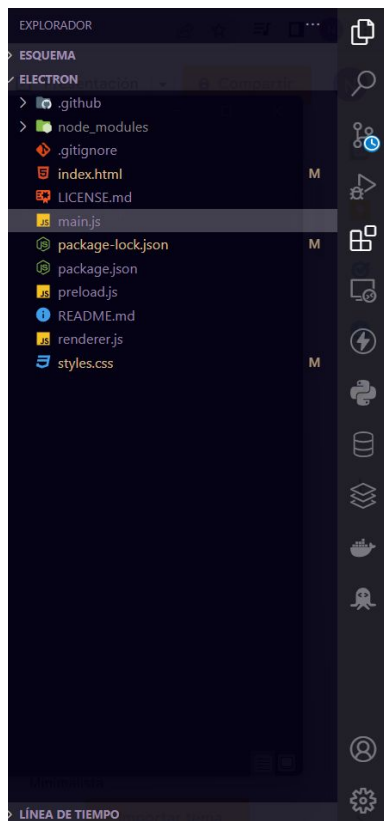
# Extensiones



A screenshot of the Visual Studio Code extension marketplace interface. The background is dark purple. The list of extensions is as follows:

- Andromeda**  
Dark theme with a taste of the universe  
Eliver Lara [Settings icon]
- Atom One Dark Theme**  
One Dark Theme based on Atom  
Mahmoud Ali [Settings icon]
- Auto Close Tag** 4ms  
Automatically add HTML/XML close ta...  
Jun Han [Settings icon]
- Auto Rename Tag** 48ms  
Auto rename paired HTML/XML tag  
Jun Han [Settings icon]
- autoDocstring - Python Docstring G...**  
Generates python docstrings automati...  
Nils Werner [Settings icon]
- Beautify** 106ms  
Beautify code in place for VS Code  
HookyQR [Warning icon] [Settings icon]
- Beautify css/sass/scss/less** 76ms  
Beautify css, sass and less code (extens...  
michelemelluso [Settings icon]





**Explorador:** Estructura de archivos y carpetas del proyecto abierto.

**Buscar:** Permite buscar palabras, comandos y frases en TODO el proyecto (carpetas, archivos) y reemplazar sus valores si se desea.

**Extensiones:** Podremos buscar, instalar y gestionar extensiones para el editor.

**Cuentas:** Permite vincular un usuario de [GitHub.com](https://github.com) con tu sesión de VSCode, para luego sincronizar y vincular tus configuraciones, extensiones y todo lo que involucre al editor con otras máquinas.

**Administrar (configuración):** Permite configurar atajos de teclado, propiedades del editor, abrir la paleta de comandos (ctrl + shift + p), snippets propios, temas, etc.

# Extensiones

Son herramientas que mejoran la productividad

## Las que deberías tener:

- **Live Server:** Actualiza tu código en el navegador en tiempo real.
- **Beautify:** Permite dar formato a tu código de manera automática
- **Prettier:** Similar a Beautify
- **IntelliCode:** Autocompletado
- **IntelliSense for CSS**
- **Javascript (ES6) Code Snippets**
- **Bootstrap 4, Font Awes... Snippets:**

## Extensiones que suman:

- **Auto Close Tag:** Cierra las etiquetas de manera automática.
- **Auto Rename Tag:** Cambia el nombre de apertura y cierre en simultáneo.
- **Better Comments:** Hacer comentarios con variedad de colores y fondos (configurable)
- **Bracket Pair Colorizer:** Colorea pares de llaves, paréntesis y corchetes de manera uniforme
- **Material Theme:** Temas
- **css-auto-prefix:** Genera código CSS para mayor compatibilidad con exploradores
- **Error Lens:** Marca los errores o advertencias dentro del editor
- **Indent Rainbow:** Separa la indentación con colores
- **Material Icon Theme:** Temas para iconos
- **Path Intellisense:** Autocompleta rutas de archivos



**HTML**



# ¿Qué es HTML5?

## HTML: HyperText Markup Language

Las siglas del lenguaje hacen referencia a un Lenguaje de Etiquetas y (Markup) texto interconectado, entrelazado con contenido variado, como ser imágenes, audio, otros archivos (HyperText).

# Estructura semántica del lenguaje HTML

Como a la hora de comunicar algo en nuestro lenguaje verbal respetamos una estructura del mensaje para ser entendidos: En nuestro código debemos seguir ciertas pautas para que el mismo pueda ser comprendido con claridad por nuestra comunidad.

# Estructura semántica del lenguaje HTML

Esta práctica no sólo se limita a HTML ni tampoco perjudica el funcionamiento de nuestro programa, sin embargo marcará nuestro perfil profesional en el rubro, recordemos que nosotros “codeamos” para las personas, para nosotros, y no para las máquinas.

# Estructura semántica del lenguaje HTML

Nuestro código debe ser lo más *legible* posible, *sencillo*, *concreto*, que cumpla su cometido.

Seguir estas normas pautadas de manera colectiva en la comunidad nos lleva a tener un código más *limpio*, *escalable*, *estable*, *coherente*, menos propenso a errores. Este conjunto de normas las denominamos **buenas prácticas** y las estaremos desarrollando a lo largo de la cursada.

## ¿Qué es una etiqueta?

Las etiquetas (tags) son las unidades que manejamos en el lenguaje HTML para ofrecer propiedades puntuales a lo que esté dentro de ella o a lo que esté involucrada con la misma (sea texto, imagen, sonido, video u otros archivos).

**`<p>Hola!</p>`**

Por lo general se conforman por una apertura (encapsulada entre los signos <>) y un cierre (igual a la apertura pero con una barra / al comienzo)

**``**

Algunos casos puntuales (como la etiqueta img) se pueden cerrar en la misma etiqueta de apertura, y en otros como <br> o <hr> no los necesitan

# Etiqueta `<head></head>`

**<head>** Representa lo que el cargador cargará, pero no mostrará en el navegador (no se renderiza). Dentro de ella podremos encontrar las siguientes etiquetas:

**<meta>** Incluye información como codificación de caracteres, compatibilidad, campo de renderizado, temas, elementos de SEO y redes sociales.

**<link>** Rutas de archivos a cargar para la página (CSS, iconos, web manifest) y el momento de carga de archivos(preload)

**<title>**Título de la página**</title>**  
**</head>**

# Etiqueta `<body></body>`

**<body>** La etiqueta body comprende todos los aspectos visuales del sitio, la cual incluye dos etiquetas principales que ayudan a comprender la semántica del sitio:

**<header>** Aquí solemos incluir los elementos de cabecera de la página, como el menú, logo, accesos al carrito, al perfil del usuario.**</header>**

**<main>** Esta es una tercera etiqueta que no tiene un uso marcado como las anteriores, sin embargo ayuda mucho a organizar un proyecto cuando ese a gran escala. Va antes del footer, y es el cuerpo del sitio.**</main>**

**<footer>** Los elementos que encontramos al pie de la página suelen incluirse acá, links de interés, datos de la empresa, un mapa del sitio, links externos entre otros**</footer>**

**<script src='./src/js/archivo.js'>**Luego de las principales header y footer en caso de usar javascript se utiliza este tag, acompañado del atributo src que representa la fuente/ruta del archivo js (javascript)**</script>**  
**</body>**



# Etiqueta `<body></body>`

**<body>** La etiqueta body comprende todos los aspectos visuales del sitio, la cual incluye dos etiquetas principales que ayudan a comprender la semántica del sitio: header y footer.

**<header>** Aquí solemos incluir los elementos de cabecera de la página, como el menú, logo, accesos al carrito, al perfil del usuario.

**<nav>** Permite crear menús de navegación, se asocia con otras etiqueta como **<ul>** (“unordered list”=“lista desordenada”) u **<ol>** (“ordered list”=“lista ordenada”) y dentro de ellas varias etiquetas **<li>** (item list).  
**</header>**

**<main>** Esta es una tercera etiqueta que no tiene un uso marcado como las anteriores, sin embargo ayuda mucho a organizar un proyecto cuando este crezca. Se encuentra entre header y footer,, y es el cuerpo del sitio.

**<article>** Se utiliza generalmente para blogs, donde agrupa por la temática de artículos (se utiliza para dar un mejor orden a la hora de leer el código)**</article>**

# Otras etiquetas para el `<body></body>`

**<section>** Similar al tag article pero con un concepto semántico más general, podemos decir que puede incluir tags articles dentro suyo, separa secciones de la página.**</section>**

**<aside>** El contenido que no tiene una inclusión directa en la temática de la página, como puede ser un banner publicitario iría en este tag semántico**</aside>**  
**</main>**

**<footer>** Los elementos que encontramos al pie de la página suelen incluirse acá, links de interés, datos de la empresa, un mapa del sitio, links externos entre otros**</footer>**

**<script src=“./src/js/archivo.js”>**Luego de las principales header y footer en caso de usar javascript se utiliza este tag, acompañado del atributo src que representa la fuente/ruta del archivo js (javascript)**</script>**  
**</body>**

# Etiquetas básicas

Dentro del body podremos hacer uso de las siguientes etiquetas (no olvidar cerrar las mismas con los signos de mayor y menor, sin olvidar la barra al inicio, de esta manera:

**<h1>**Este es el texto dentro de la etiqueta h1**</h1>**

La mayoría de las etiquetas funcionan como bloques, es decir: Cuando se cierra un tag el siguiente no comenzará a su lado, sino debajo. Existen excepciones a esto, como la etiqueta span y las que dan estilo al texto (negrita, cursiva, tachada, subrayada).

**<h1><h2><h3><h4><h5><h6>:** Estas etiquetas se usan para otorgar un tamaño al tipo de letra (de mayor a menor respectivamente), destacarse del resto del texto y también se usa como estrategia de marketing (SEO).

# Etiquetas básicas

**<p>**: Representa un párrafo, con un tamaño de letra genérico.

**<b>**: Convierte el contenido en letra negrita.

**<i>**: Convierte su contenido en letra itálica, o como también la conocemos: cursiva.

**<u>**: Subraya el texto incluido entre este tag.

**<hr>**: Genera una línea que recorre el ancho de la página. No necesita cerrarse.

**<br>**: Produce un salto de línea (saltea un renglón). No necesita cerrarse.

**<p>**: Representa un párrafo, con un tamaño de letra genérico.

**<b>**: Convierte el contenido en letra negrita.

**<i>**: Convierte su contenido en letra itálica, o como también la conocemos: cursiva.

**<u>**: Subraya el texto incluido entre este tag.

**<del>**: Tacha el texto dentro del tag.

# Etiquetas

**<span>** Las etiquetas como p, h1, h2... generan bloques de texto, una etiqueta p seguida de otra etiqueta p hará que el texto continúe en el renglón siguiente. Esto no sucede con span, el cual permite conectar diferentes textos en el mismo renglón.

**<div>** Uno de los tags más populares, administra su contenido como bloques, pero permite una gran organización del material cuando se asocia con CSS

**<audio>** Permite agregar audios a la página, entre sus tantos atributos podemos encontrar el autoplay, controls, type que nos permitirán gestionar la reproducción del archivo.

# Etiquetas

**<video>** Esta etiqueta permite ingresar videos a nuestra página, podemos usar atributos como loop, autoplay, controls para manejar su funcionamiento.

**** Permite ingresar una imagen a la página, no olvidar que siempre debe tener una fuente (link), con su atributo *src* (source), como también el atributo *alt* donde podemos completar con una palabra o frase que haga referencia a ella (se la utiliza por temas de SEO y en caso de personas no videntes).

# Atributos

Los atributos son propiedades que pueden recibir las etiquetas, ellas modificarán las propiedades del contenido de las tags.

Hay que estar atentos a que no todas las etiquetas se les puede pasar cualquier atributo, para ello empezaremos a recurrir a la documentación del lenguaje!

[Documentación HTML MDN \(español\)](#)

[Documentación HTML W3S](#)

# Atributos

Por último tendremos un atributo especial que veremos en el ejemplo de esta clase llamado “style”, este será nuestro primer paso al mundo CSS, ya que éste asigna un estilo a la etiqueta que lo incluye, pero ojo, lo más recomendable es evitar usarlo, a futuro nos puede generar problemas al usar CSS!



# Links de interés

Regístrate en Github!

[Github.com](https://github.com)

[CMDer: ¿Qué es? ¿Cómo instalarlo?  
¿Dónde bajarlo \(En su descripción\)?](#)

[Descarga Visual Studio Code](#)

[Revisa tu versión del sistema Windows](#)

Documentación HTML

[MDN Mozilla Development Network](#)

[W3Schools](#)

Comunidad de ayuda mundial

[Stackoverflow](#)

[Stackoverflow \(español\)](#)

Buscador Dev

[You.com](#)

## **Recordá:**

- **Revisar la Cartelera de Novedades.**
- **Hacer tus consultas en el Foro.**

**Todo en el Aula Virtual.**