

# **Desarrollo Web**

## **de Principiante a**

## **Profesional con Paga**

*Volumen I*

**Construye tu portafolio mientras aprendes  
HTML5, CSS y JavaScript  
paso a paso con apoyo**

Por

*Ojula Technology Innovations*

**Desarrollo Web**  
**de Principiante a Profesional con Paga**

*Volumen 1*

Construye tu portafolio mientras aprendes HTML, CSS y JavaScript paso a paso con apoyo



Copyright © [Ojula Technology Innovations](#)

Todos los derechos reservados

**ISBN:** 9791221326369

**Publicado en los Estados Unidos**

**Limitación de responsabilidad/Exención de garantía**

Este libro contiene información obtenida de fuentes auténticas y de gran prestigio. Se han realizado esfuerzos considerables para publicar datos e información fiables, pero el autor y el editor no pueden asumir la responsabilidad por la validez de todos los contenidos ni las consecuencias de su uso. Toda la información que se ofrece en este libro se basa en la propia investigación del autor y no constituye ningún asesoramiento técnico, financiero o profesional. El autor y el editor han intentado localizar a los propietarios de los derechos de autor de todo el material incluido en esta publicación, y piden disculpas a los propietarios correspondientes si no se ha obtenido el permiso para publicarlo de esta forma. Si algún material con derechos de autor no ha sido reconocido, por favor escríbanos y háganoslo saber para que podamos corregirlo en alguna reimpresión futura.

Excepto por los casos permitidos por la Ley de Derechos de Autor de los Estados Unidos, ninguna parte de este libro puede ser reimpressa, reproducida, transmitida o utilizada en cualquier forma por cualquier medio electrónico, mecánico o de otro tipo, conocido o por inventarse en un futuro, incluyendo fotocopias, microfilmación y grabación, o en cualquier sistema de almacenamiento o recuperación de información, sin el permiso por escrito del editor.

## Tabla de contenidos

Cómo aprender bien HTML, CSS y JavaScript

- 0. Comenzando con herramientas de programación gratis
- 1. HTML básico y recursos útiles
  - 1.1. Saluda a los elementos HTML
  - 1.2. Encabezados con el elemento H2
  - 1.3. Comunica con el elemento de Párrafo
  - 1.4. Llena los espacios en blanco con texto Placeholder
  - 1.5. Comentar/Quitar comentarios en HTML
  - 1.6. Comentar en HTML
  - 1.7. Eliminar elementos HTML
  - 1.8. Introducción a los elementos HTML5
  - 1.9. Agregar imágenes a tu sitio web
  - 1.10. Enlaces a páginas externas con el elemento Ancla
  - 1.11. Enlaces a secciones internas de una página con el elementos Ancla
  - 1.12. Anidar un elemento Ancla dentro de un párrafo
  - 1.13. Crear enlaces muertos usando el signo de numeral
  - 1.14. Convertir una imagen en un enlace
  - 1.15. Crear una lista desordenada
  - 1.16. Crear una lista ordenada
  - 1.17. Crear un campo de texto
  - 1.18. Agregar texto Placeholder a un campo de texto
  - 1.19. Formularios: Crear un elemento Formulario
  - 1.20. Agregar un botón de envío a un Formulario
  - 1.21. Usar HTML5 para campos obligatorios

- 1.22. Crear un conjunto de botones de opción
- 1.23. Crear casillas de opción múltiple
- 1.24. Menú desplegable, botón deslizante, área de texto, validación de formulario y botón para subir archivo
- 1.25. Anidar elementos con Div y Span
- 1.26. Declarar el tipo de documento en HTML
- 1.27. Definir la cabecera y el cuerpo de un documento HTML
- 1.28. Cambiar el color del texto
- 1.29. Introducción al inspector de Chrome
- 1.30. Tablas en HTML
  - 1.30.1. Cómo hacer una tabla básica
  - 1.30.2. Cómo agregar encabezados a una tabla
  - 1.30.3. Cómo agregar un título a una tabla
  - 1.30.4. Cómo usar el atributo de rango
  - 1.30.5. Cómo modificar el rango de las celdas
  - 1.30.6. Cómo agregar cabecera, cuerpo y pie de página a una tabla
2. Bases de CSS (Hoja de estilo en cascada)
  - 2.1. Usar selectores CSS para dar estilo a los elementos
  - 2.2. Usar una clase CSS para dar estilo a un elemento
  - 2.3. Estilizar múltiples elementos con una clase CSS
  - 2.4. Selector universal \*{ }
  - 2.5. Cambiar el tamaño de letra de un elemento
  - 2.6. Configurar el tipo de letra de un elemento
  - 2.7. Importar tipos de letra con Google Font
  - 2.8. Especificar cómo los tipos de letra deben degradarse

- 2.9. Cómo redimensionar tus imágenes
- 2.10. Agregar bordes alrededor de tus elementos
- 2.11. Agregar esquinas redondeadas
- 2.12. Crear imágenes circulares
- 2.13. Configurar el color de fondo en un elemento Div
- 2.14. Asignar un identificador a un elemento
- 2.15. Usar identificadores para estilizar elementos
- 2.16. Ajustar el relleno de un elemento
- 2.17. Ajustar el margen de un elemento
- 2.18. Agregar un margen negativo a un elemento
- 2.19. Agregar rellenos diferentes a cada lado de un elemento
- 2.20. Agregar diferentes márgenes a cada lado de un elemento
- 2.21. Usar notaciones abreviadas para especificar el relleno de un elemento
- 2.22. Usar notaciones abreviadas para especificar el margen de un elemento
- 2.23. Usar selectores de atributo para estilizar elementos
- 2.24. Comprendiendo las unidades Absoluto y Relativo (EM y REM)
- 2.25. Estilizar el cuerpo del HTML
- 2.26. Herencia de estilos desde el cuerpo del HTML
- 2.27. Prioridad de un estilo sobre otro
- 2.28. Sobrescribir estilos en CSS posterior
- 2.29. Sobrescribir declaraciones de clase con estilo de atributo de identificador
- 2.30. Sobrescribir declaraciones de clase con estilo Inline
- 2.31. Sobrescribir todos los demás estilos utilizando Important
- 2.32. Usar código hexadecimal para colores específicos
- 2.33. Usar código hexadecimal para mezclar colores

- 2.34. Usar código hexadecimal abreviado
  - 2.35. Usar valores RGB para los colores de elementos
  - 2.36. Usar valores RGB para mezclar colores
  - 2.37. Lista de selectores
  - 2.38. Selectores descendientes
  - 2.39. Selector adyacente (Combinador)
  - 2.40. Selector de hijo/descendiente directo
  - 2.41. Pseudoclases
  - 2.42. Pseudoelementos
3. Avance del Volumen 2
- 3. 1. Continuar con tu entrenamiento en desarrollo web
  - 4. Apuntes de repaso, portafolio de desarrollador, ejercicios de programación y proyectos de práctica
    - 4.1. Introducción
    - 4.2. Descargar los recursos de entrenamiento
    - 4.3. Cómo conseguir más ayuda
  - 5. Más recursos útiles

## Cómo aprender bien HTML, CSS y JavaScript

Sé que existen más que suficientes recursos en internet para enseñar a los más principiantes todo lo que necesitan para aprender desarrollo web desde cero, pero este libro es diferente porque aborda algunos problemas desarrollo web. La credibilidad de un desarrollador web es mucho más que sus credenciales. En realidad, se trata del peso de su portafolio.

Las oportunidades de empleo suelen venir de las muestras de trabajo y de habilidades concretas, no tanto de un título universitario. En este sentido, es necesario aprender lo suficiente. Este libro no solo te ayudará a aprender bien y a crear un portafolio impresionante, sino que también te proporcionará ayuda y apoyo continuo.

Con este libro y mi **dedicado equipo de ayuda y soporte 24/7**, no hay nada que temer. He ayudado a muchos desarrolladores web a actualizar sus habilidades de desarrollo, iniciar carreras exitosas y ser contratados para trabajos remotos. Me ha dado cuenta de que incluso los principiantes más ambiciosos pueden enfrentarse a problemas, como el de no poder decidir por dónde comenzar. A veces se pierden por completo en el camino y, por tanto, necesitan ayuda.

La verdad es que todo el mundo necesita ayuda en un momento u otro para aprender HTML, CSS o JavaScript adecuadamente en su recorrido de desarrollo web. En este libro he plasmado suficientes retos de programación y sus soluciones. Obtendrás **hojas de repaso actualizadas regularmente, ejercicios de programación, un sorprendente portafolio de desarrollador web para mostrar tus muestras de trabajo, proyectos para practicar, gráficos y capturas de pantalla utilizados en el libro. Asimismo, obtendrás más ayuda si utilizas el enlace de soporte gratuito al final de este libro. Te ayudaré a aprender bien.**

Por último, si compras este libro, te añadiré a mi comunidad de desarrollo web para que puedas seguir aprendiendo y obtener más ayuda hasta que te conviertas en un profesional bien pagado.

*Bolakale Aremu  
CEO, Ojula Technology Innovations  
Desarrollador Web e Ingeniero en Software  
[Ojulaweb.com](http://Ojulaweb.com)*

## 0. Comenzando con herramientas de programación gratis

HTML se encuentra en su quinta versión, por lo que se conoce como HTML5. CSS está en su tercera, por lo que se conoce como CSS3. Independientemente de que utilices un sistema operativo Windows, Mac o Linux, necesitas un navegador moderno y actualizado, además de un editor de texto/código, como el bloc de notas, para comenzar tu viaje de programación. Sin embargo, te recomiendo encarecidamente que descargas e instalas el [editor de Visual Studio Code](#) y uses [Google Chrome](#) como navegador antes de comenzar. Ambos son bastante fáciles de instalar y cuentan con un gran número de herramientas para desarrolladores preinstaladas. Adicionalmente, en YouTube hay diversos videos sobre cómo instalar las últimas versiones. Una vez que los hayas instalado, ¡es hora de comenzar!

En tu escritorio o en cualquier otra ubicación que quieras en tu computadora:

1. Crea una nueva carpeta y ponle un nombre apropiado, como **Desarrollo-web** (es recomendado evitar nombrar haciendo uso de ciertos caracteres especiales, como letras tildadas). Abre esta carpeta.
2. Dentro de la carpeta, crea otra y ponle nombre, por ejemplo: **Curso\_html**. Abre esta carpeta.
3. Dentro, crea otra carpeta y ponle un nombre como **Leccion\_01**. Abre esta carpeta.
4. En esta carpeta crea un documento de texto, cambia el nombre por defecto a *index* y su extensión original (.txt) a .html. Véase la ilustración 0.1.

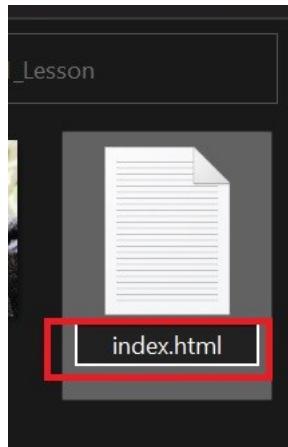


Ilustración 0.1: Renombrando un nuevo documento de texto (Bloc de notas)

Luego de confirmar el cambio de nombre, recibirás una advertencia indicando que el documento puede quedar inutilizable. (Nota: En Windows, para esta opción se debe tener seleccionada la casilla **Extensiones de nombre de archivo**, en la pestaña **Vista** del Explorador de archivos). Selecciona Sí para confirmar que quieras cambiar el nombre y la extensión del documento a **index.html**. Este nuevo documento es lo que se suele usar como un tipo de **página de inicio** estándar o **página principal** de una aplicación. ¡La primera parte de tu página web está lista! Ahora el archivo debería mostrar el logo de Google Chrome o el de cualquier navegador que hayas seleccionado como predeterminado, como se muestra en la ilustración 0.2.



Ilustración 0.2: El documento ahora muestra el logo de Google Chrome.

Ahora abre Visual Studio Code (su descarga se mencionó anteriormente). A continuación se mostrarán pasos adicionales para cambiar el idioma del editor a español, en caso de que así lo prefieras. (1) Ir a la pestaña de Extensiones (Extensions), (2) dentro de la barra de búsqueda, digitar *Spanish Language Pack for Visual Studio Code*, (3) seleccionar la extensión correspondiente de entre los resultados y, por último, (4) hacer clic en *Install*. Para realizar los cambios, debemos reiniciar Visual Studio Code, lo que podremos hacer desde una ventana de notificación que aparecerá en la esquina inferior derecha.



Ilustración 0.3: Cambiando el idioma de la interfaz a español.

Abre el menú **Archivo** (1) en la parte superior izquierda; al hacer clic aparecerá el panel mostrado en la Ilustración 0.4. Más adelante en este libro podrás seleccionar la primera opción (**Nuevo archivo de texto**) para crear un archivo sin nombre (*untitled*) para escribir tu código.

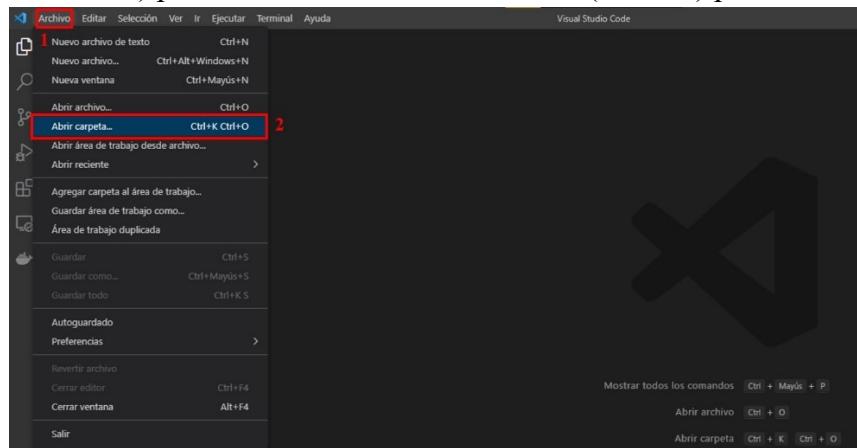


Ilustración 0.4: Cómo abrir una nueva carpeta en Visual Studio Code

Por ahora solo seleccionaremos **Abrir carpeta** (2), lo que abrirá la caja de diálogo en la ilustración 0.5. Navega hasta la carpeta **Leccion\_01** y haz clic sobre ella para seleccionarla (1), luego en el botón **Seleccionar carpeta** (2) para cerrar la ventana de diálogo.

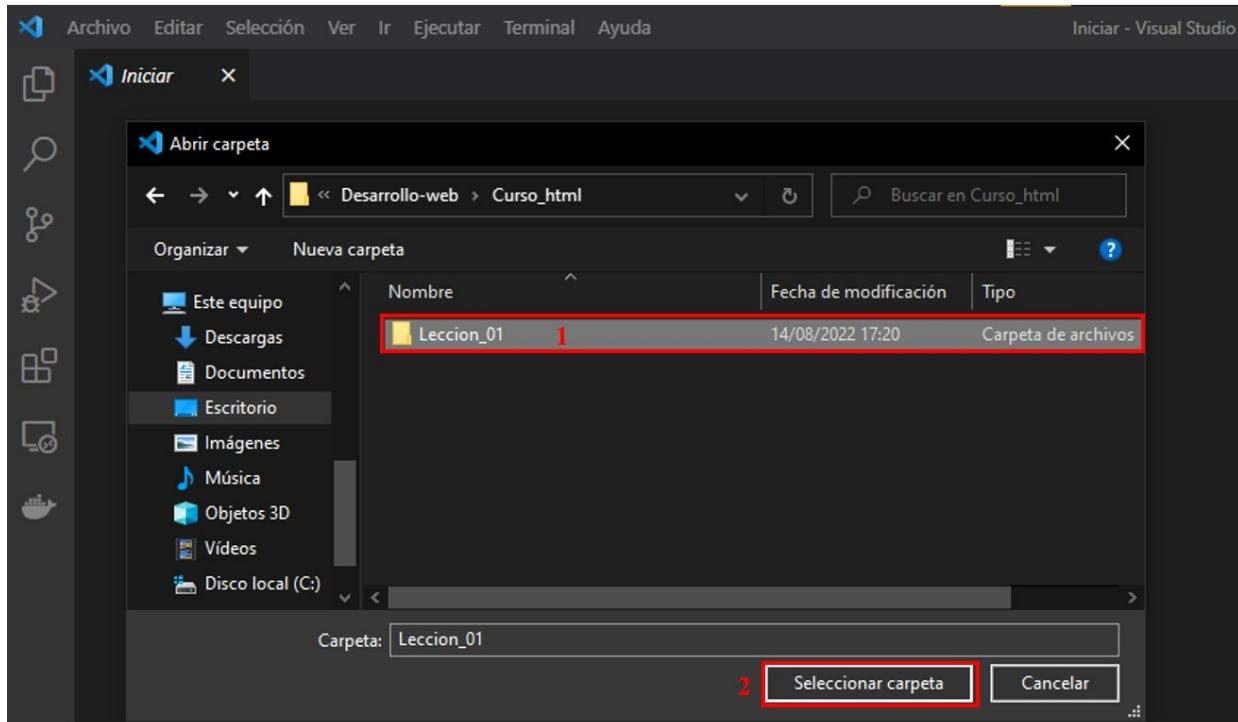


Ilustración 0.5: Cómo navegar y seleccionar la carpeta en tu computadora.

Ahora deberías ver la carpeta (1) y el archivo **index.html** (2) en el panel izquierdo de Visual Studio Code (se suele abreviar como VSC o VS Code), como se muestra abajo; dale clic para abrirlo.

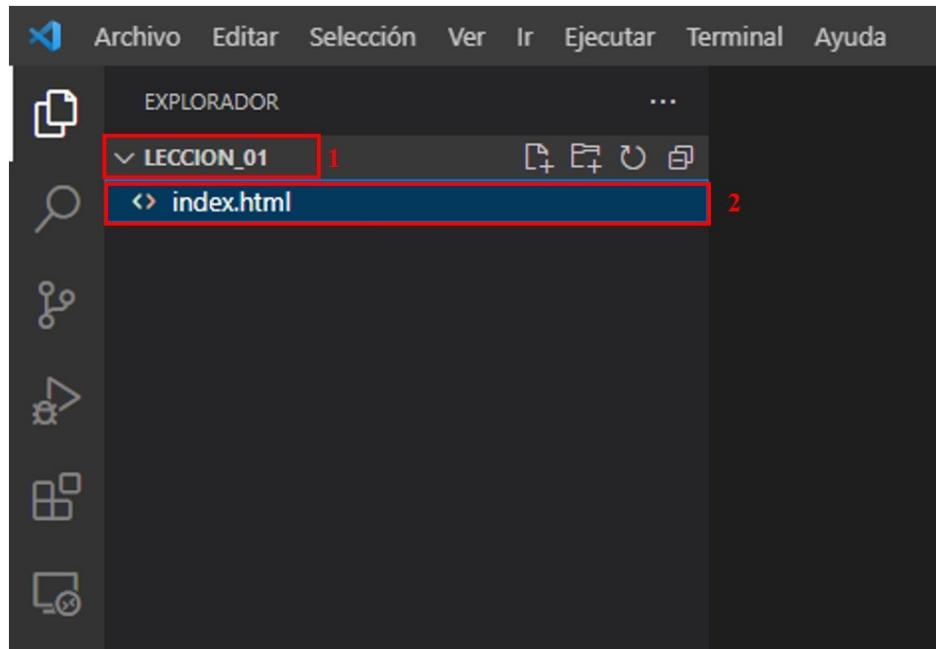


Ilustración 0.6: Cómo seleccionar archivos desde el panel del explorador de archivos de Visual Studio Code.

El paso anterior abrirá una ventana en la parte derecha con el título del archivo (*index.html*) (1). A partir de aquí todo el código que escribas será en una ventana como esta. El número 1 (2) que ves debajo de la pestaña con el nombre indica que esa es la primera línea de tu código.

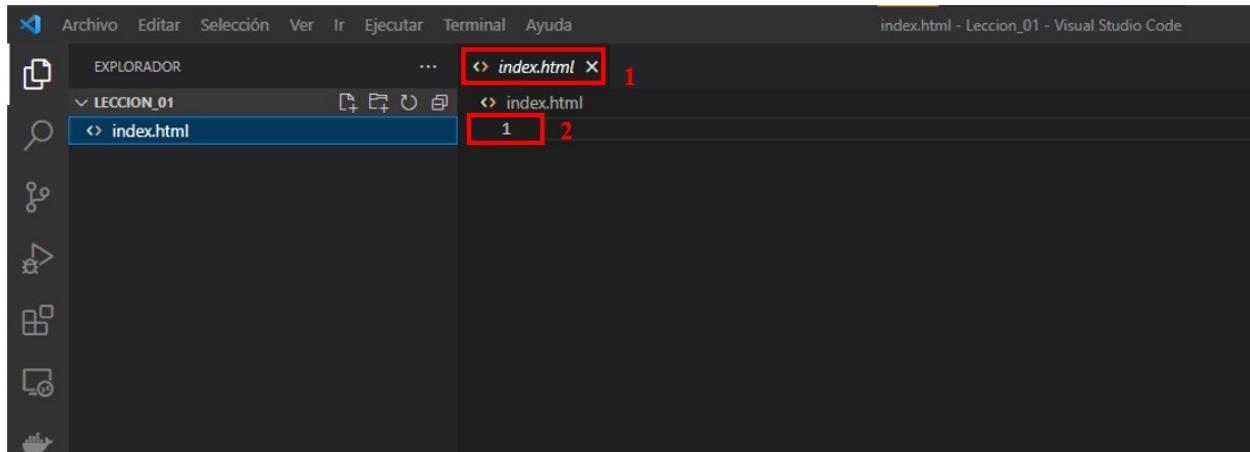


Ilustración 0.7: Ventana del editor de Visual Studio Code (derecha) mostrando la primera linea del código.

Entonces, ¿estás listo para comenzar a escribir tu código? ¡Avancemos al siguiente capítulo!

## 1. HTML básico y recursos útiles

¿Qué significa HTML5? He visto mucha confusión al respecto con estudiantes, y por una buena razón. Para algunos, HTML5 suena como una versión que se descarga, se instala o se actualiza, similar a un videojuego; sin embargo, HTML5 es un poco diferente.

Si leemos la [entrada en MDN](#) sobre ello, nos dice:

«*El término HTML5 es básicamente una palabra de moda que se refiere a un conjunto de tecnologías web modernas*». (Traducido del inglés).

Es la última evolución del estándar que define el HTML. El término abarca dos conceptos diferentes: una nueva versión de un lenguaje de HTML con nuevos elementos, atributos y comportamientos; y un grupo más amplio de tecnologías web.

No se trata de nada a lo que debas prestarle mucha importancia por ahora, excepto que debes saber que incluye varias tecnologías que nos ayudan a hacer cosas como construir sitios web con gráficos en 3D, almacenar información en el navegador y otras cosas que, técnicamente, no son HTML, pero que seguimos llamando HTML. Todo ese nuevo conjunto de funciones se llamó HTML5 o HTML5 y amigos. No te preocupes si todo esto parece confuso.

Cuando digo HTML5, me refiero solo a la última evolución del **estándar** que define HTML. Pero, ¿qué significa «estándar» y por qué es un estándar que define el HTML? Bueno, la forma en que funciona HTML es que no hay una copia oficial que se descarga (nunca instalamos HTML). Existe un documento muy extenso llamado [\*El Estándar HTML\*](#) (documento en inglés) que explica cómo debe funcionar el HTML.

Así que, cuando decimos HTML5, no es una versión de algo que descargues e instales, sino que se trata de una versión de este documento e incluye nuevos elementos, como elementos de vídeo y audio. Estos no existían hace 15 años en HTML, pero ahora forman parte de este documento, lo que hace posible que los navegadores (Chrome, Firefox, Edge o el navegador que estés usando) los implementen.

Podrías haber seguido con normalidad sin saber nada de esto, pero quería aclararlo porque escuché a los estudiantes hablar de actualizar a HTML5 y cosas así. Como mencioné, esto no es algo que hagas manualmente; tu navegador y la organización responsable de él son los que implementan características basadas en la última versión de este estándar.

En palabras técnicas, HTML es un lenguaje de marcado que utiliza una sintaxis o notación especial para describir la estructura de una página web al navegador. Los elementos HTML suelen tener lo que llamamos «etiquetas» de apertura y de cierre que rodean y dan significado al contenido. Por ejemplo, diferentes elementos pueden describir el texto como un título, un párrafo o una lista.

Ahora, me gustaría presentarte rápidamente uno de los recursos más útiles que utilizo con frecuencia en este libro, y te recomiendo que lo utilices también. Se llama [Mozilla Developer Network](#) o [MDN](#) (se encuentra parcialmente traducida al español). Ciertamente es un recurso que te salva de apuros, ya que abarca los fundamentos de las tecnologías web y mucho más. La ilustración 1.0 es una captura de pantalla de MDN.



Ilustración 1.0:Página de referencia de HTML en el sitio web de [MDN](#)

Es un recurso escrito por y para desarrolladores. Considéralo como la Wikipedia de HTML, CSS, JavaScript, navegadores y muchas más cosas que explican las reglas de HTML. Está llena de tutoriales, referencias y guías, lo que la convierte en una documentación o recurso que te ayuda a entender y aprender sobre las tecnologías web.

No obstante, ten en cuenta que Mozilla no es el creador de HTML, CSS o JavaScript, y que existen otros recursos similares a MDN. Esta es una especie de enciclopedia, una recopilación de conocimiento, pero no un documento oficial ni nada parecido. Es de código abierto y está creada por desarrolladores.

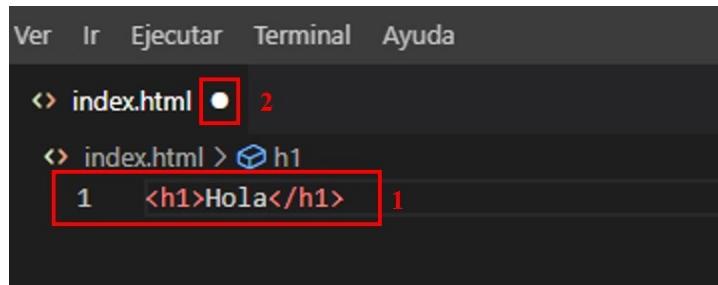
## 1.1. Saluda a los elementos HTML

En este volumen construirás una aplicación web con fotos de gatos para aprender algunos de los elementos HTML más comunes. Piensa en esto como los bloques de construcción de cualquier página web.

Bienvenido a mis desafíos de HTML; estos te guiarán a través de tu viaje de desarrollo web, paso a paso. Lo primero que harás será crear una página web sencilla utilizando HTML. Para esto, regresa al editor en VSC (Visual Studio Code) y escribe lo siguiente:

```
<h1>Hola</h1>
```

Se debe parecer a lo que se muestra en la ilustración 1.1.1.



```
<h1>Hola</h1>
```

Ilustración 1.1.1: La primera línea del marcado está completa.

El marcado (*markup*) en tu editor de código que dice `<h1>Hola</h1>` (1) es un elemento HTML. El punto blanco junto al nombre (2) significa que tienes cambios sin guardar en tu archivo.

Para guardar tu código, haz clic en Archivo (1) y luego en Guardar (o con la combinación Ctrl+S) (2). Después de guardar tu código, el punto blanco cambiará a una X (3). Ver la ilustración 1.1.2.

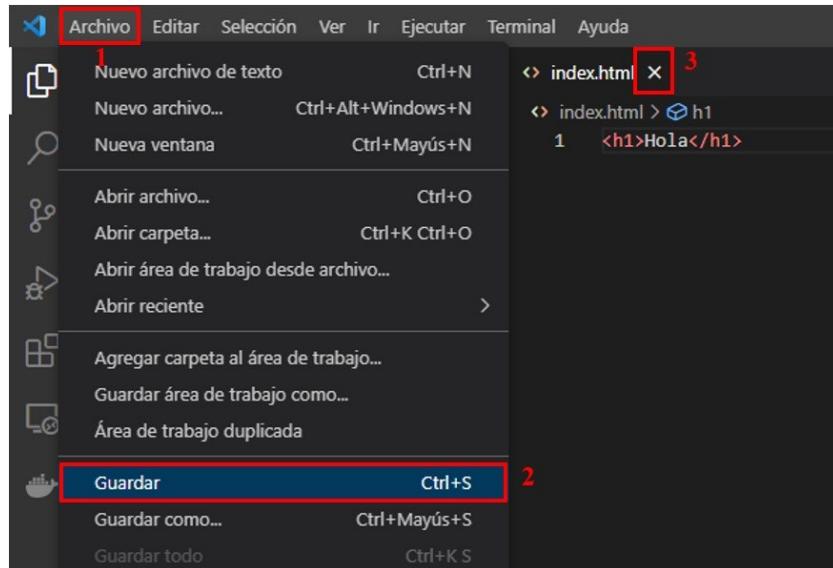


Ilustración 1.1.2: Cómo guardar tu código en el editor.

Como se mencionó anteriormente, los elementos HTML tienen una **etiqueta de apertura** y una **etiqueta de cierre**.

Las etiquetas de apertura se ven así:

```
<h1>
```

Las etiquetas de cierre se ven así:

```
</h1>
```

La única diferencia es que la etiqueta de cierre tiene una barra diagonal después de la llave de apertura (<). Posiciona el puntero al final de la etiqueta de cierre (</h1>), haz clic y presiona Enter. Esto creará automáticamente la segunda línea para seguir con tu código.

Como se explicará en breve, cada vez que quieras probar/ejecutar tu código, simplemente haz clic derecho en cualquier parte del editor y seleccionar **Open with Live Server**. En caso de no tener Live Server instalado, o si prefieres no utilizarlo, entonces haz clic en **Ejecutar** (1). Aquí se te presentarán diferentes formas de ejecutar tu código: **Iniciar depuración** (2) (*Debugging* en inglés) y **Ejecutar sin depuración** (3).

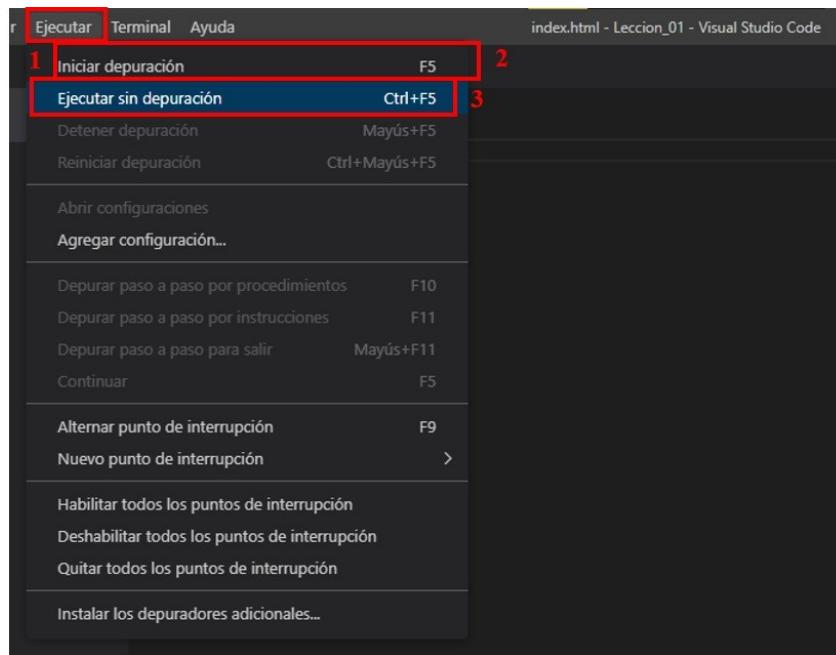


Ilustración 1.1.3: Cómo correr el código en tu editor.

Selecciona **Ejecutar sin depuración**, lo que significa que solo quieras ver el resultado de tu código, pero sin interés de comprobar la existencia de errores (comúnmente llamados *bugs*) en él. Con **Iniciar depuración** indicas que quieres comprobar si hay errores cuando ejecutas el código. Por ahora, ya que estamos seguros de que no hay errores en este marcado sencillo, nos quedaremos con **Ejecutar sin depuración**.

VSC te preguntará cuál navegador deseas utilizar para visualizar el resultado del código (1 o 2, como se muestra en la ilustración 1.1.4).

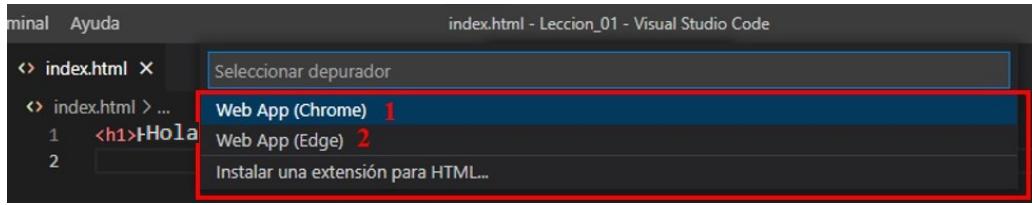


Ilustración 1.1.4: Cómo seleccionar un navegador para mostrar tu resultado.

Si no, tu navegador por defecto (Google Chrome en mi caso) debería abrirse y mostrarte el resultado de este sencillo código: la palabra Hola, así como en la ilustración 1.1.5.

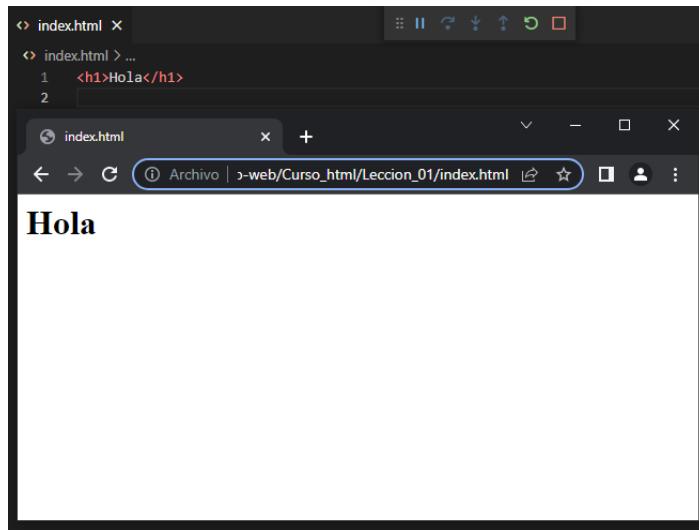


Ilustración 1.1.5: Visualizando el resultado del código en Google Chrome.

Como mencioné antes, otra forma de visualizar el resultado de tu código es utilizar **Live Server**, la cual es una extensión muy útil de VSC. Si aún no lo has instalado, puedes descargarla desde el marketplace de VSC (pestaña de Extensiones, como se vio en el capítulo anterior). Aquí encontrarás muchas otras extensiones útiles y con diversas funcionalidades que puedes descargar e instalar para mejorar tu experiencia en VSC.

Hay distintas formas en las que puedes descargar e instalar Live Server con la ayuda del siguiente enlace: <https://marketplace.visualstudio.com/items?itemName=ritwickdey.LiveServer>. Ver ilustración 1.1.6.

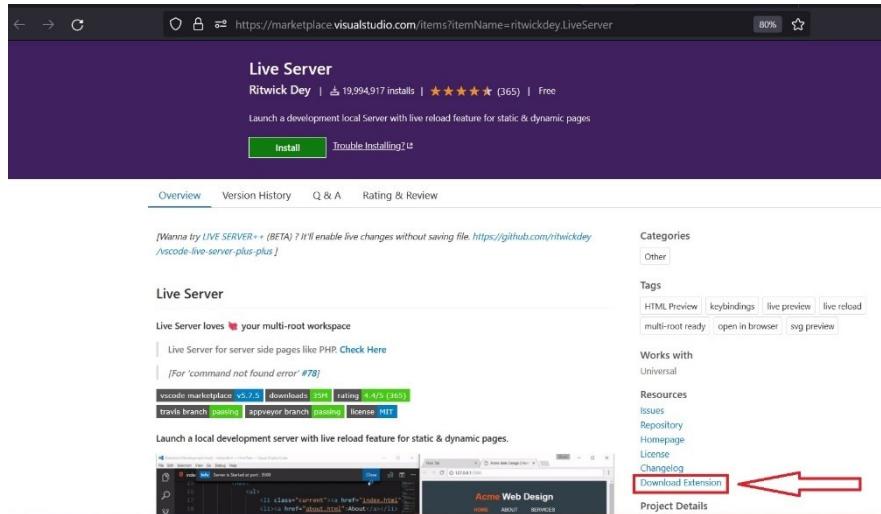


Ilustración 1.1.6: Página para descargar Live Server.

Esta es la primera manera de instalar la extensión. Sigue los pasos en la ilustración 1.1.7.

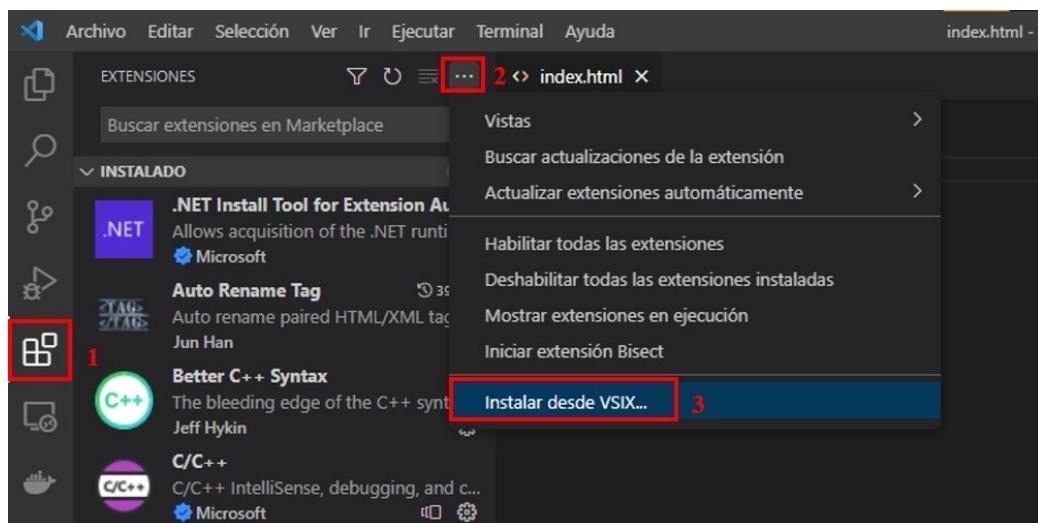


Ilustración 1.1.7: Cómo instalar Live Server con archivo VSIX.

Paso 1: Selecciona el ícono de **Extensiones** (1) en el panel de la izquierda.

Paso 2: Haz clic en el ícono de tres puntos (...) (2) para mostrar más opciones.

Paso 3: Selecciona **Instalar desde VSIX...** (3).

Cuando se abra la ventana de diálogo, navega hasta la carpeta donde se guardó la extensión que descargaste de Live Server. Selecciona el archivo y haz clic en el botón **Instalar** para comenzar. La instalación debería terminar en un par de segundos, y tal vez necesites reiniciar VSC antes de poder usarla.

La segunda forma es seleccionar el botón **Install** (1) en la página de descarga y luego aceptar

## Abrir Visual Studio Code (2).

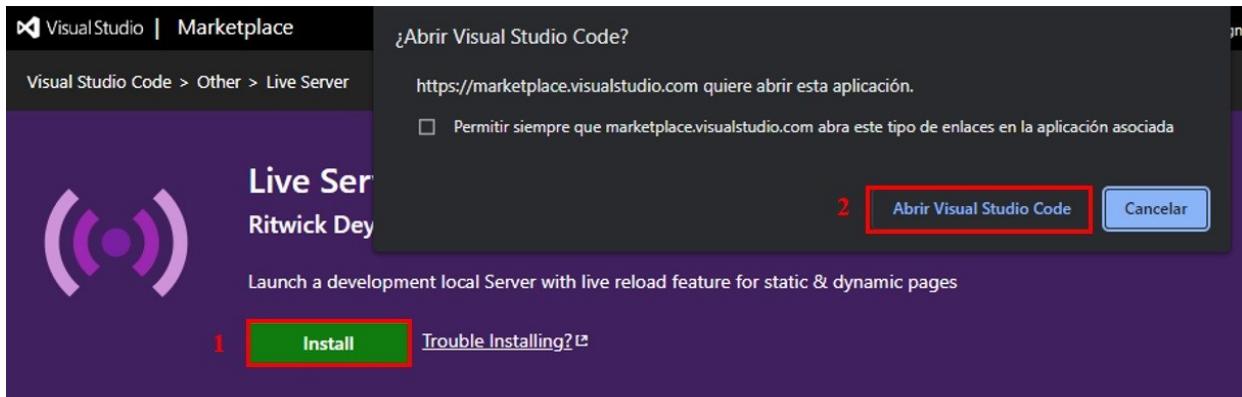


Ilustración 1.1.8: Segunda forma para descargar Live Server.

Esto debería abrir el apartado de Extensiones en VSC, donde debes hacer clic en **Instalar** y esperar a que se complete, y después reiniciar VSC.

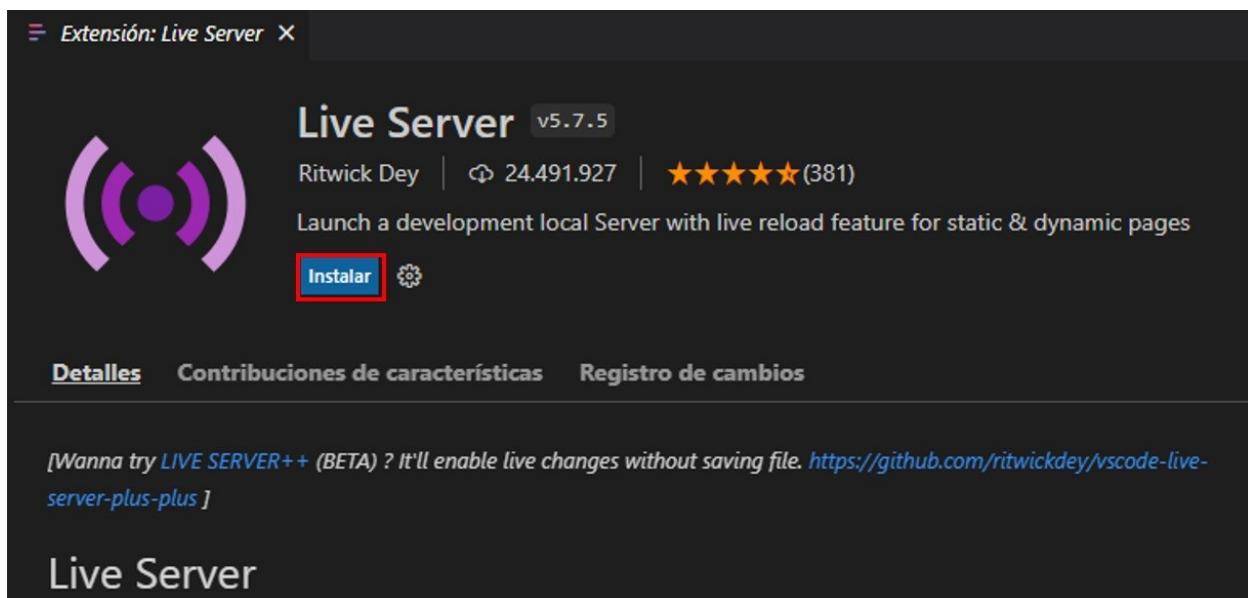


Ilustración 1.1.9: Instalando Live Server desde Extensiones en Visual Studio Code.

Ahora cada vez que quieras probar tu código, podrás hacerlo haciendo clic derecho en el editor y seleccionando **Open with Live Server**. Ver ilustración 1.1.10.

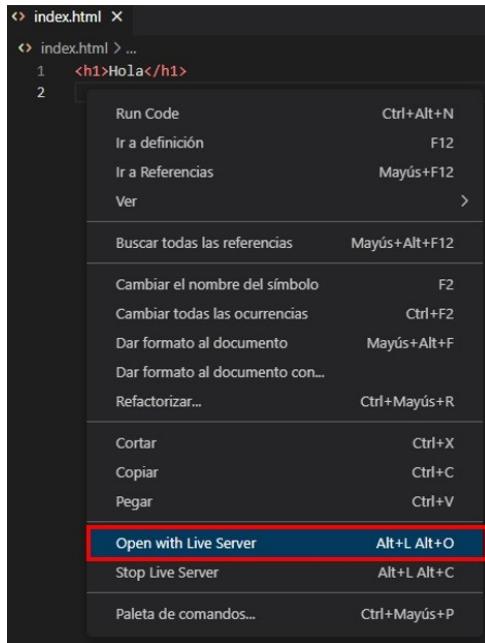


Ilustración 1.1.10: Cómo abrir el navegador con la extensión Live Server.

Tu navegador se abrirá y mostrará el resultado, como en la ilustración 1.1.11. Como puedes ver en la barra del enlace, notarás que la dirección web es la de un servidor local: <http://127.0.0.1:5500/index.html>, a diferencia de en la ilustración 1.1.5, donde la dirección era solamente la ubicación del archivo dentro de tu computadora.

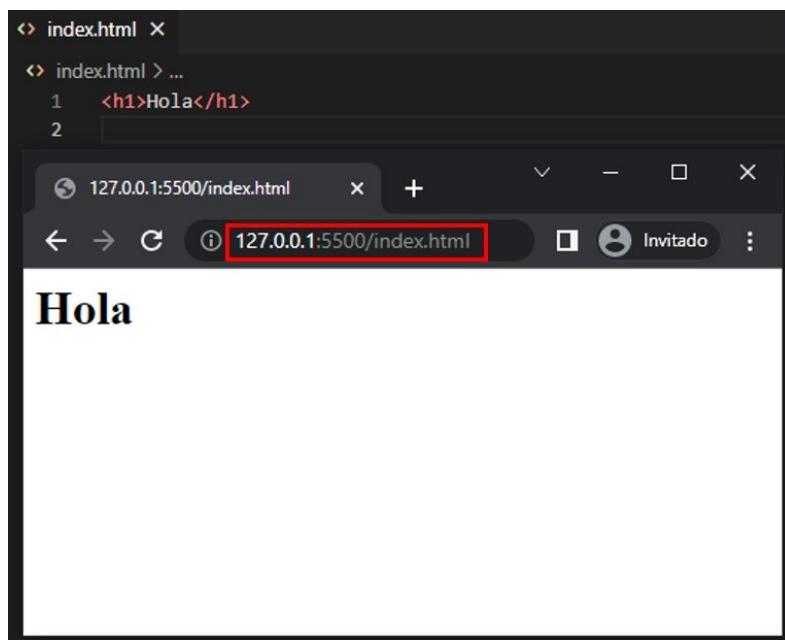


Ilustración 1.1.11: Visualizando el resultado del código con Live Server en Google Chrome.

A continuación, tendrás tu primer reto: cambiar el texto del elemento h1 para que diga «Hola Mundo», guardar los cambios y probar el código. Si quieres intentar otra

forma de crear archivos, entonces puedes hacerlo desde el panel del explorador de archivos de VSC. Tienes las opciones de hacerlo ya sea con clic izquierdo sobre el **ícono de Nuevo archivo** (1) o con clic derecho dentro del panel y seguidamente escogiendo la opción **Nuevo archivo** (2); ponle el nombre que quieras, por ejemplo **CodigoDePrueba.html**, y presiona Enter para terminar. Ver las ilustraciones 1.1.12 y 1.1.13.



Ilustración 1.1.12: Cómo crear un nuevo archivo desde Visual Studio Code.

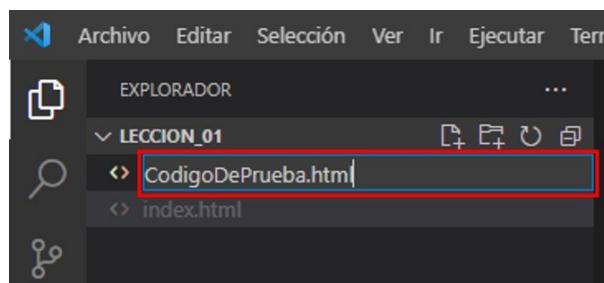


Ilustración 1.1.13: Cómo nombrar un nuevo archivo creado desde Visual Studio Code.

**Nota:** No olvides guardar siempre tu código antes de ejecutarlo; de lo contrario, es posible que no se actualice cuando lo quieras visualizar.

A medida que vayamos avanzando, te enseñaré más elementos de encabezado (elementos h, del inglés *headline*) muy similares. Ya te mostré el primero, el elemento h1, que es el encabezado de nivel superior para toda página web; todos los elementos comunes de encabezado o para encabezado de sección son: h1, h2, h3, h4, h5 y h6.

Sin embargo, no deberías usar un elemento h3 en tu página sin un h2 o un h1; del mismo modo, nunca deberías tener un h6 sin un h5 o h4, y así sucesivamente. En realidad, estos encabezados no existen para aumentar de tamaño las palabras, sino para dar estructura a la página y para marcar el tema principal y los subtemas.

En pocas palabras, estos encabezados son importantes (¡los usamos todo el tiempo!), pero recuerda **no utilizarlos por el tamaño** y procurar usarlos correctamente. Por ejemplo, lo recomendado es tener un elemento h1 como máximo en una página.

## 1.2. Encabezados con el elemento H2

A lo largo de las siguientes lecciones, construiremos una aplicación web con fotos de gatos en HTML5, parte por parte. El elemento h2 en este paso agregará un encabezado de nivel dos a la página web.

Este elemento le comunica al navegador la estructura de tu página web. Es habitual que los elementos h1 se utilicen para los encabezados principales y los h2 para los subencabezados. Los demás elementos h (h3, h4, h5, h6) sirven para indicar diferentes niveles de subtítulos.

Agrega una etiqueta h2 con el texto «Fotos de gatos» para crear un segundo elemento HTML debajo del elemento h1 con «Hola Mundo».

## 1.3. Comunica con el elemento de Párrafo

El elemento p es el elemento preferido para mostrar texto en forma de párrafo en las páginas web. p es por «párrafo» (*paragraph* en inglés).

Puedes crearlo de la siguiente manera:

```
<p>¡Soy una etiqueta p!</p>
```

Crea un elemento p debajo de tu elemento h2 y escribe el texto «Hola Párrafo».

**Nota:** Como costumbre, todas las etiquetas HTML se escriben en minúsculas, por ejemplo: <p></p> en vez de <P></P>.

## 1.4. Rellena los espacios en blanco con texto Placeholder

Es bastante común que los desarrolladores web usen `lorem ipsum` como texto placeholder. Pero antes de continuar, ¿qué es el texto placeholder y qué es `lorem ipsum`? Primero aclaremos placeholder. Si bien *placeholder* en español significa «marcador de posición», es más frecuente el uso de su nombre en inglés (también es conocido como «texto provisional»). Se utiliza para llenar el espacio donde irá el texto oficial y poder visualizar la estructura de estos elementos. No se debe confundir con el atributo `placeholder`.

Ahora, acerca de `lorem ipsum`, se trata de un texto que se extrae al azar de un famoso pasaje del filósofo romano Cicerón. Los tipógrafos lo han utilizado desde el siglo XVI, y esta tradición persiste en la web.

Crear texto placeholder con `lorem ipsum` dentro de VSC es fácil. Puedes hacerlo escribiendo la palabra «`lorem`» y presionando la tecla Enter; las palabras son escogidas aleatoriamente, y este fue mi resultado:

`Lorem, ipsum dolor sit amet consectetur adipisicing elit.  
 Delectus harum labore, molestias tempore a tenetur. Tempora id  
 voluptatem rem ducimus odio repellat saepe veniam aperiam. Quasi  
 illo ratione velit expedita!`

VSC te permite modificar el texto a generar. Por ejemplo, si escribes «`lorem5`», se generarán solamente cinco palabras, con «`lorem100`» se generarán 100 palabras; si escribes «`lorem*3`», se generarán 3 líneas de texto placeholder. Si quieres personalizarlo aún más, entonces puedes escribir algo como «`lorem10*5`», lo que generará 5 líneas de texto con 10 palabras cada una.

## 1.5. Comentar/Quitar comentarios en HTML

Crear comentarios (o comentar) es la manera en que puedes dejar notas/comentarios para otros desarrolladores dentro de tu código, sin afectar el resultado que verá el usuario final. Además, es muy conveniente para ignorar líneas de código que no necesitemos, pero sin borrarlas.

Los comentarios en HTML comienzan con `<!--` y terminan con `-->`.

Puedes comentar varias líneas de código al mismo tiempo así:

```
<!--  
<h1>Hola Mundo</h1>  
  
<h2>Fotos de gatos</h2>  
  
<p>Lorem, ipsum dolor sit amet consectetur adipisicing elit.  
 Delectus harum labore, molestias tempore a tenetur. Tempora id  
 voluptatem rem ducimus odio repellat saepe veniam aperiam. Quasi
```

```
illo ratione velit expedita!</p>
```

```
-->
```

## 1.6. Comentar en HTML

Recuerda que para comentar necesitas comenzar con `<!--` y `-->` para cerrar el comentario; esto también aplica para líneas no consecutivas. A continuación, comenta los elementos h1 y p, pero no el h2, similar al código abajo:

### Código:

```
<!--  
<h1>Hola Mundo</h1>  
-->  
<h2>Fotos de gatos</h2>  
<!--  
<p> Lorem, ipsum dolor sit amet consectetur adipisicing elit.  
Delectus harum labore, molestias tempore a tenetur. Tempora id  
voluptatem rem ducimus odio repellat saepe veniam aperiam. Quasi  
illo ratione velit expedita!</p>  
-->
```

## 1.7. Eliminar elementos HTML

Imagina que nuestro celular no tiene mucho espacio vertical. Vamos a remover los elementos innecesarios para comenzar a construir nuestra aplicación.

Eliminar elementos es tan fácil como crearlos, sencillamente borra la línea (incluyendo las etiquetas). En este ejemplo hemos borrado el elemento h1 para simplificar nuestra vista:

```
<h2>Fotos de gatos</h2>  
  
<p>Lorem, ipsum dolor sit amet consectetur adipisicing elit.  
Delectus harum labore, molestias tempore a tenetur. Tempora id  
voluptatem rem ducimus odio repellat saepe veniam aperiam. Quasi  
illo ratione velit expedita!</p>
```

## 1.8. Introducción a los elementos HTML5

HTML5 introduce etiquetas HTML más descriptivas. Entre ellas encontramos:

- `main` («principal» en inglés),
- `header` («cabecera» en inglés),
- `footer` («pie de página» en inglés),
- `nav` («navegación», del inglés *navigation*),
- `video`,
- `article` («artículo» en inglés),
- `section` («sección» en inglés),
- y muchas otras.

Estas etiquetas proporcionan una estructura descriptiva a tu HTML, volviéndolo más fácil de leer, otorgando mayor accesibilidad y mejorando el SEO (Optimización para motores de búsqueda, del inglés *Search Engine Optimization*). La etiqueta `main` ayuda a los motores de búsqueda y a otros desarrolladores a encontrar el contenido principal de tu página. Aquí hay un ejemplo de su uso, con un elemento `main` y dos elementos hijos (*child*) anidados:

```
<main>  
  <h1>Hola Mundo</h1>  
  <p>Hola Párrafo</p>  
</main>
```

**Nota:** Muchas de las nuevas etiquetas HTML5 y sus ventajas están contempladas en la sección *Accesibilidad aplicada* (capítulo 4).

Ahora crea un segundo elemento `p` con texto placeholder (o copia el de abajo):

  Lorem ipsum dolor sit amet consectetur adipisicing elit. Vero  
  asperiores optio quod, facere eos deserunt impedit est sequi  
  error corporis!

Después crea un elemento `main` y anida únicamente los dos elementos `p` en su interior. Con anidar nos referimos a colocar un elemento dentro de otro.

```
<h2>Fotos de gatos</h2>  
  
<main>  
  <p>Lorem, ipsum dolor sit amet consectetur adipisicing elit.  
  Delectus harum labore, molestias tempore a tenetur. Tempora id  
  voluptatem rem ducimus odio repellat saepe veniam aperiam. Quasi
```

illo ratione velit expedita!</p>

<p>Lorem ipsum dolor sit amet consectetur adipisicing elit.  
Vero asperiores optio quod, facere eos deserunt impedit est  
sequi error corporis!</p>

</main>