**Proceso instalación Visual Studio Code**

**Paso 1:** Escribimos en Google Visual Studio Code y seleccionamos donde dice “Download”.



**Paso 2:** Seleccionamos el sistema operativo que tenemos y lo descargamos.



**Paso 3:** Al darle clic nos descargará un .exe, al cual le daremos clic encima.



**Paso 4:** Lee y acepta el acuerdo de licencia. Haz clic en Next para continuar.



**Paso 5:** Puedes cambiar la ubicación de la carpeta de instalación o mantener la configuración predeterminada. Haz clic en Next para continuar.



**Paso 6:** Elige si deseas cambiar el nombre de la carpeta de accesos directos en el menú Inicio o si no deseas instalar accesos directos en absoluto. Haz clic en Next.



**Paso 7:** Selecciona las tareas adicionales, por ej. crear un icono en el escritorio o añadir opciones al menú contextual de Windows Explorer. Haz clic en Next.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

**Paso 8:** Haz clic en Install para iniciar la instalación.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

**Paso 9:** El programa está instalado y listo para usar. Haz clic en Finish para finalizar la instalación y lanzar el programa.



**Creación de archivos y carpetas**

**Paso 1:** Creamos una carpeta raíz. En nuestro caso la llamaremos “etiquetas”.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente con confianza media

**Paso 2:** Dentro de esta, crearemos otras dos carpetas llamadas “css”, la cual contendrá todo lo estético de nuestra página, y otra llamada “js” la cual contendrá la lógica de nuestra página.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

**Etiqueta**

**Paso 1:** Para abrir el Visual Studio Code, haremos lo siguiente: Dentro de la carpeta raíz, en la barra superior, escribiremos cmd y le damos enter.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

**Paso 2:** Eso nos abrirá una terminal, solo tendremos que escribir “code .”.

Texto

Descripción generada automáticamente

**Paso 3:** Esto nos abrirá el visual studio code. Ahora, ya que estamos dentro, crearemos un archivo llamado “index.html” dándole en la carpeta raíz dando click donde indica la flecha.

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

**Paso 4:** Llamaremos el archivo “index.html”

Captura de pantalla de un celular

Descripción generada automáticamente

**Paso 5:** Una vez en el archivo recién creado, colocamos el código “!” y escogemos la primera opción.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

**Paso 6:** Una vez colocada la estructura, cambiaremos el titulo (línea 6) por “Etiquetas”

Captura de pantalla de computadora

Descripción generada automáticamente

**Paso 7:** Ahora dentro del body crearemos un div principal para nuestra página (Linea 10) le pondremos un id llamado “app”, y además crearemos otro div dentro de ese mismo div (línea 12) y le pondremos el id “input-tag”.

Texto

Descripción generada automáticamente

**Paso 8:** Ahora crearemos un nuevo archivo llamado “app.js” dando click derecho a la carpeta llamada “js” y new file

Captura de pantalla de un celular

Descripción generada automáticamente

Captura de pantalla de un celular

Descripción generada automáticamente

**Paso 9:** Una vez creado el archivo vamos a vincular el javascript en nuestro html (Línea 16)

Texto

Descripción generada automáticamente

**Paso 10:** Una vez vinculado el javascript iremos al archivo “app.js”, donde crearemos un array para almacenar las etiquetas creadas por el usuario (línea 2)

Además vamos a llamar el id del html (“input-tag”) que creamos anteriormente (línea 3) además crearemos 2 constantes que se van a encargar de crear un contenedor de etiqueta y de entrada en el html cuando sea necesario (línea 4-5)

Texto

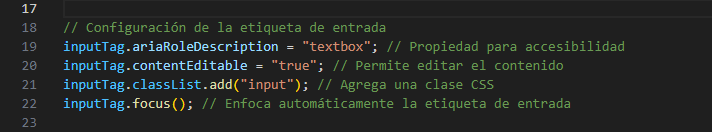
Descripción generada automáticamente

**Paso 11:** Ahora crearemos una función “inputTagContainer” (línea 8) el cual funcionará para los contenedores de las etiquetas, comenzaremos creando un condicional (línea 10), y en caso de que encuentre el id “input-tag” (línea 11) o cualquier elemento secundario de la clase “tag-container” (línea 12), en caso de que se cumpla cualquiera va a enfocar la tarjeta de entrada para editarla (línea 14)

Texto

Descripción generada automáticamente

**Paso 12:** Ahora crearemos un código el cual configura una etiqueta de entrada. Establece su propiedad ariaRoleDescription (línea 19)para mejorar la accesibilidad, permite la edición del contenido con contentEditable(línea 20), agrega la clase CSS "input" (linea21) a la etiqueta y la enfoca automáticamente para que el usuario pueda comenzar a editarla de inmediato (línea 22).

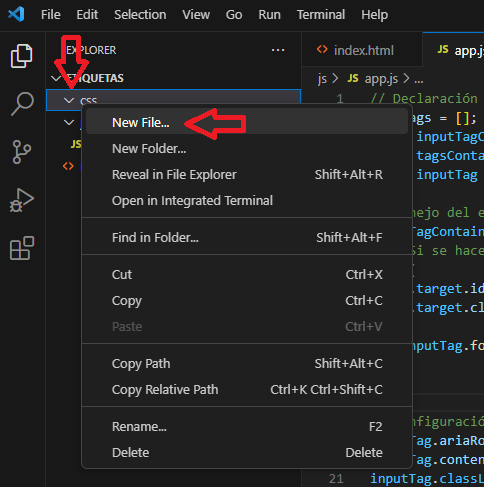


**Paso 13:** Ahora vamos a añadir clases css a los elementos del HTML (línea 25-26)

Texto

Descripción generada automáticamente

**Paso 14:** Ahora miraremos el diseño de nuestra página, lo primero que haremos será dar click derecho en la carpeta “css” y darle a new file o nuevo archivo

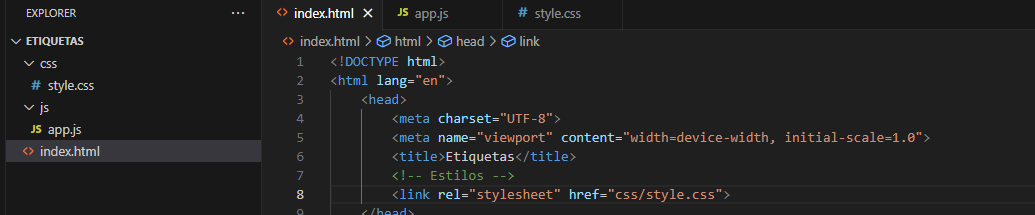


**Paso 15:** Llamaremos el archivo como “style.css”

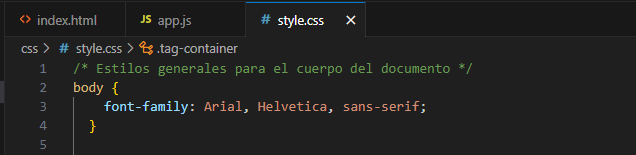
Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

**Paso 16:** Ahora iremos a nuestro archivo “index.html” y vincularemos nuestro css a nuestro html (línea 8)



**Paso 17:** Iremos a nuestro archivo “style.css” y pondremos unas fuentes (línea 2-4)



**Paso 18:** Ahora vamos a crear un diseño para la etiqueta donde escribimos

Texto

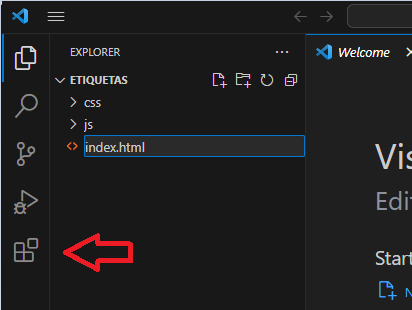
Descripción generada automáticamente

**Paso 19:** Ahora vamos a modificar el diseño de nuestro contenedor de etiquetas.

Texto

Descripción generada automáticamente

**Paso 20:** Ahora para visualizar nuestro proyecto daremos click a la zona inferior izquierda como apunta la flecha.



**Paso 21:** Nos saldrá esta pestaña y buscaremos live server, en la barra de búsqueda.

**Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

Descripción generada automáticamente**

**Paso 22:** Ahora que encontramos lo que necesitamos, le damos click a “install”

**Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente**

**Paso 23:** Ahora que tenemos instalado el live server nos debería de aparecer algo así donde podremos volver a nuestro proyecto dando click al explorador.

**Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

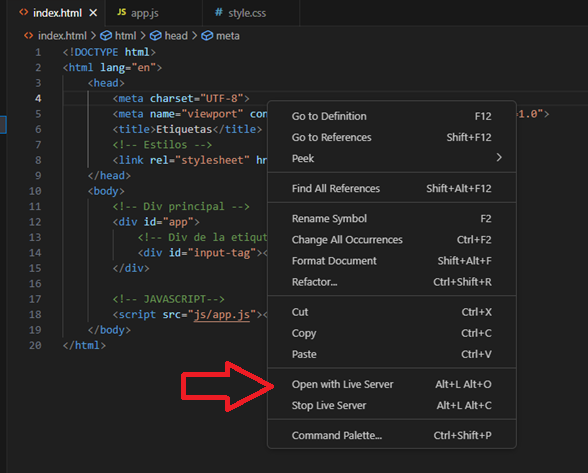
Descripción generada automáticamente**

**Paso 24:** Una vez te sale el explorador volvemos al Index.html

**Captura de pantalla de un celular

Descripción generada automáticamente**

**Paso 25:** Una vez volvemos al index.html damos click derecho en cualquier parte del código y damos click izquierdo a “open with live server”

****

**Paso 26:** Ahora nuestra página estará así

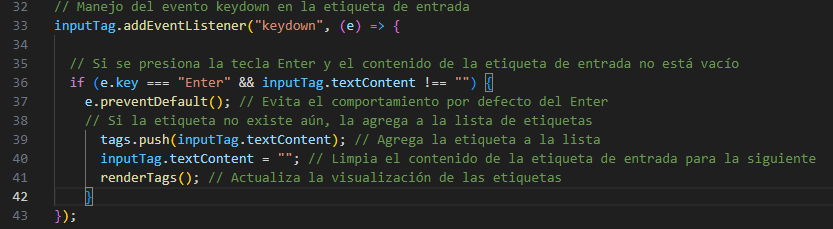
Forma

Descripción generada automáticamente

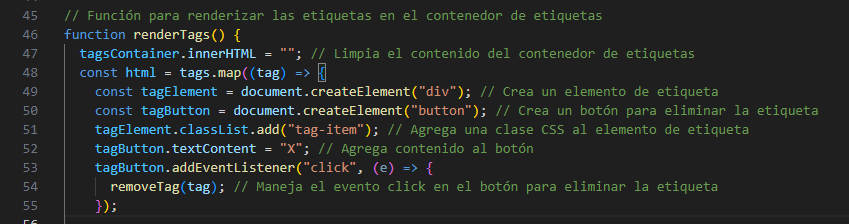
Forma

Descripción generada automáticamente

**Paso 27:** Ahora volveremos a nuestro archivo “app.js” y vamos a añadir un evento (línea 33) y una condicional de que en caso de que se presione la tecla “enter” y el contenido no sea igual a vacio va a quitar el comportamiento por defecto (línea 37) y vamos a agregar la etiqueta en la lista (línea 39) también limpiaremos la lista para que no se esté generando constantemente (línea 40) y vamos a llamar la función “renderTags” que aun no esta creada pero sirve para actualizar la visión y se vea la etiqueta que recién creamos (línea 41).



**Paso 28:** Ahora vamos a crear la función “renderTags” línea (línea 46) y comenzaremos limpiando el contenido del contenedor (línea 47) y vamos a recorrer el array “tags” con la función “map” (línea 48) y vamos a crear un elemento de etiqueta que será un div (línea 49) y un botón (línea 50) a esos elementos le agregaremos la clase (tag-item), (línea 51) y al botón le pondremos de contenido una “X” (línea 52) y por añadiremos un evento para que cuando pulse el botón que acabamos de crear se elimine la etiqueta (línea 53 a 54).



**Paso 29:** Ahora vamos a añadir el contenido de la etiqueta al elemento div que hermos creado anteriormente con el texto de la etiqueta (línea 57), y además vamos a añadir la “X” (línea 58)

Texto

Descripción generada automáticamente

**Paso 30:** Ahora dentro de la misma función “renderTags” vamos a hacer que por cada elemento añada un elemento de etiqueta (línea 63-64), también agregaremos la etiqueta “inputTag” a todas las etiquetas generadas, para poder agregar nuevas etiquetas luego (línea 66) y por último colocamos para que el usuario pueda añadir otra etiqueta sin siquiera clicar (línea 67)

Texto

Descripción generada automáticamente

**Paso 31:** Ahora mismo nuestro proyecto y veremos que podemos añadir etiquetas sin embargo aún se pueden repetir las etiquetas.

Forma

Descripción generada automáticamente  
**Paso 32:** Ahora volvemos a nuestro “app.js” y crearemos una función llamada “existTag” que se encargara de ver si ya existe (linea 70) una etiqueta mirando en caso de que el valor ya este incluido (linea 71)

Texto

Descripción generada automáticamente

**Paso 33:** Ahora vamos a añadir la función “removeTag” para eliminar la etiqueta y actualizamos la visualización (línea 75), filtramos las que no se van a eliminar (línea 76) y renderizamos (línea 77)

Texto

Descripción generada automáticamente

**Paso 34:** Ahora iremos de nuevo de nuevo al evento en caso de apretar “Enter” (Paso 27) y colocaremos un condicional que en caso de que la etiqueta no exista la coloque (línea 39)

Texto

Descripción generada automáticamente

**Paso 35:** Y ahora en caso de que se aprete la tecla “backspace” (línea 45) y no tenga contenido dentro (línea 46) y que haya mínimo 1 etiqueta (línea 47), va a eliminar la etiqueta (línea 50) y va a llamar la función “renderTags” (línea 51)

Texto

Descripción generada automáticamente

**Paso 36:** Ahora miramos nuestro proyecto y podemos agregar etiquetas, tampoco podremos colocar 2 etiquetas iguales y podremos eliminar las etiquetas tanto con el botón como con la tecla de retroceso

Forma, Rectángulo

Descripción generada automáticamente

**Paso 37:** Ahora iremos a nuestro archivo “style.css” y comenzaremos con el diseño del contenedor.

Texto

Descripción generada automáticamente

**Paso 38:** Así va quedando por ahora

Forma

Descripción generada automáticamente

**Paso 39:** Ahora hacemos un poco de diseño para la caja de etiquetas.

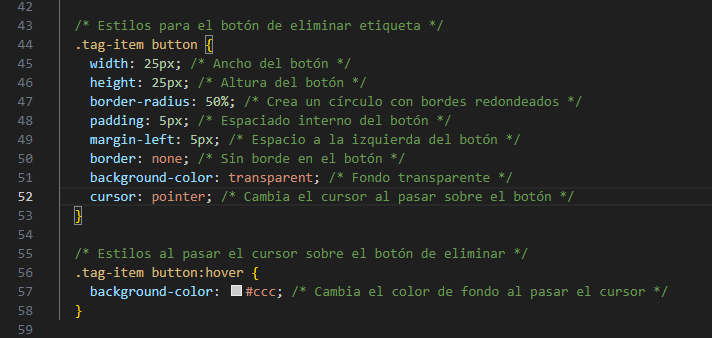
Texto

Descripción generada automáticamente

Forma

Descripción generada automáticamente**Paso 40:** Así va por ahora

**Paso 41:** Ahora cambiaremos el diseño del botón de eliminar la etiqueta. Y usaremos el “.tag-item button:hover” para cambiar el color cuando pasas el ratón del cursor por encima.



**Paso 42:** Así ha quedado nuestro proyecto.

Forma

Descripción generada automáticamente