**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 carpetas y archivos**

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



**Paso 2:**Dentro de esta, crearemos tres carpetas. Uno llamado css: Donde irán los archivos que se encargaran del diseño de la pagina; el segundo llamado js: Donde irán los archivos que contengan toda la lógica e interacción del aplicativo, y el tercero llamado img que es donde meteremos imágenes

Texto

Descripción generada automáticamente

**Paso 3**:En el buscador de archivos escribimos “cmd” y se nos abrirá la consola.

Tabla

Descripción generada automáticamente con confianza media

**Paso 4:**Escribimos en la consola “code .”.

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

**Paso 5:** Se abre la carpeta en el visual estudio code

Captura de pantalla de un celular

Descripción generada automáticamente

**Paso 6:** Creamos un nuevo archivo en la carpeta raíz llamado “index.html” y en este escribimos “html:5”

Interfaz de usuario gráfica, Texto, Aplicación, Teams

Descripción generada automáticamente

**Paso 7:**Esto habrá creado el código base de un archivo html en el cual podremos empezar a trabaja

Captura de pantalla de computadora

Descripción generada automáticamente

**Paso 8:** creamos un archivo style.css en la carpeta css.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

**Paso 9:** enlazamos el archivo index.html con el style.css

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

**Paso 10:** creamos un archivo en la carpeta js llamado app.js

Interfaz de usuario gráfica, Texto, Aplicación

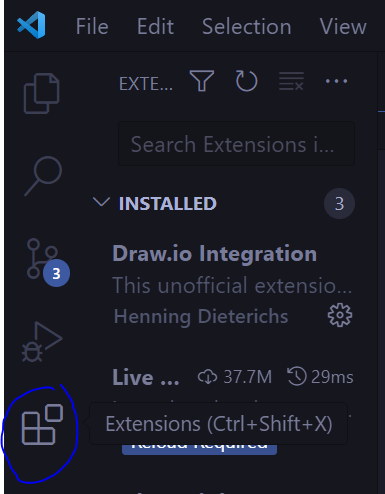
Descripción generada automáticamente

**Paso 11:** enlazamos el archivo index.html con el app.js

Texto

Descripción generada automáticamente con confianza media

**Paso 12:** en el lado derecho encontraremos un apartado de extensiones en el visual studio



**Paso 13:** en las extensiones buscamos “live server” click en la primera opción y los descargamos

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

**Estructura HTML**

**Paso 14:** En el body del archivo html crearemos un div que será el contenedor principal

Texto

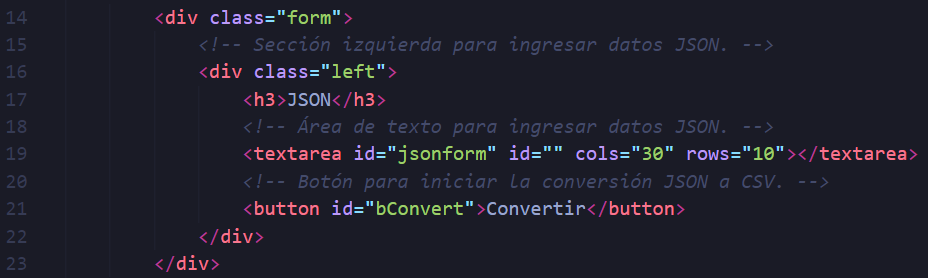
Descripción generada automáticamente

**Paso 15:** dentro del contenedor principal crearemos otro div con la clase “form”

**Texto

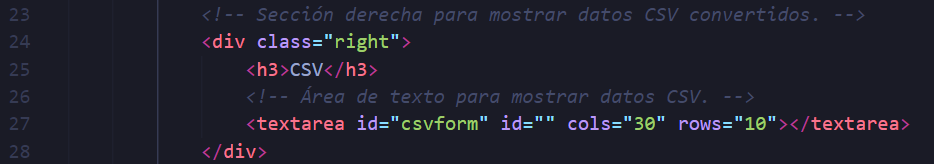
Descripción generada automáticamente**

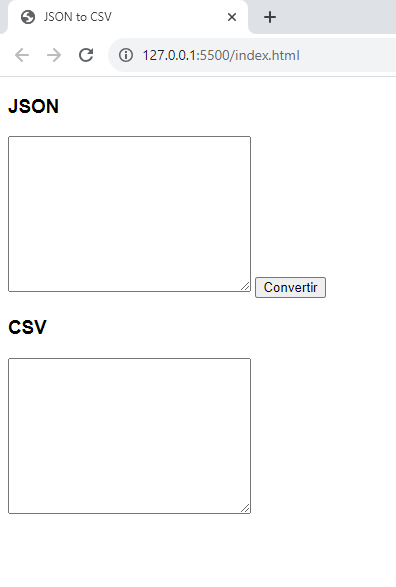
**Paso 16:** Dentro del div form crearemos un div con la clase “left” este contendrá un titulo un área de texto donde ingresaremos el json y un botón que da inicio a la conversión de json a cvs(formatos separados por comas)



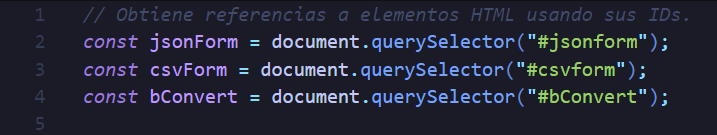
Interfaz de usuario gráfica

Descripción generada automáticamente con confianza media

**Paso 17:** ahora dentro div form creamos un div con la clase “right” donde pondremos un titulo y un área de texto donde saldrá el json en formato csv



**Estructura JS**

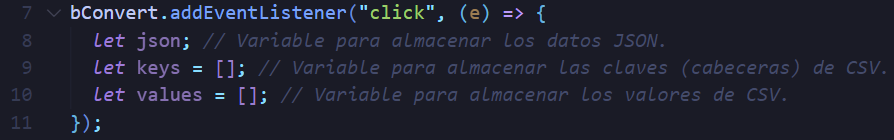
**Paso 18:** Referenciamos elementos html con el querySelector usando el id

**Paso 19:** Agregamos un listener al botón de convertir al darle click correrá todo el código que lleve adentro

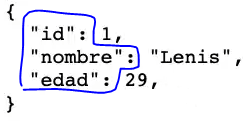
Texto

Descripción generada automáticamente

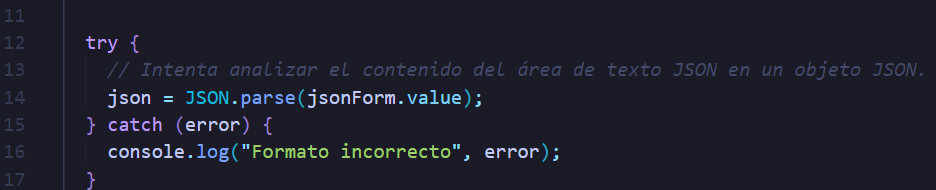
**Paso 20:** iniciamos tres variables la primera para guardar el json, la segunda para guardas las cabeceras y por ultimo el valor de esas cabeceras.

****

lo que esta dentro de la línea azul son las cabeceras lo que sigue después de los dos puntos es el valor asignado



**Paso 21:** ahora validamos si la cadena de texto que se nos dan es un json, si es un json sigue con el código de lo contrario manda un error

****

**Paso 22:** se crea un condicional para ver si el json viene en un array

Texto

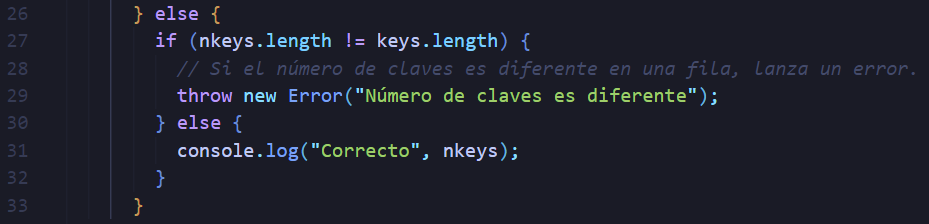
Descripción generada automáticamente con confianza media

**Paso 23:** si la condición anterior se cumple se recorre cada elemento dado que la primera iteración el array keys está vacío ósea igual a 0, keys se le agregan las propiedades de nkeys.

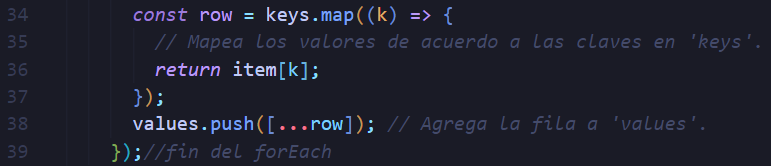
Texto

Descripción generada automáticamente

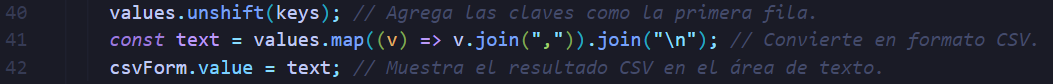
**Paso 24:** si la condición anterior no se cumple se abre un condicional donde nkeys y keys son diferentes ósea que no tienen las mismas propiedades y manda el error de lo contrario sale correcto y sigue



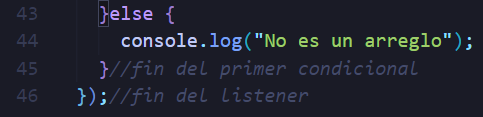
**Paso 25:** una vez tenemos las propiedades en keys lo recorremos y obtenemos lo valores y con el método push lo agregamos a value, y se cierra el forEach|.



**Paso 26**: ahora se agrega las cabeceras en el array values con el método unshift (línea 40), y se convierte en formato csv (línea 41) y por ultimo se muestra en el area de texto(línea 42)



**Paso 27:** en caso de que el primer condicional que evalúa si el json es un array se falso que muestre que no es un arreglo en la consola.



**Estructura CSS**

**Paso 28**: en el archivo css en el body cambiamos la fuente de la letra

Texto

Descripción generada automáticamente

**Paso 29**: la propiedad flex  indica la capacidad de un elemento flexible para alterar sus dimensiones y llenar el espacio disponible y la propiedad gap especifica los canales entre las filas y las columnas de la cuadrícula

Texto

Descripción generada automáticamente con confianza media

Interfaz de usuario gráfica, Aplicación, Word

Descripción generada automáticamente

**Paso 30**: se le da un ancho del 50% de la pagina a la izquierda y derecha del formulario

Texto

Descripción generada automáticamente

Interfaz de usuario gráfica, Aplicación, Word

Descripción generada automáticamente

**Paso 31:** el box-sizing establece cómo se calcula el ancho y alto total de un elemento con el valor border-box le dice al navegador que tenga en cuenta cualquier borde y relleno en los valores que especifique para el ancho y alto de un elemento

Texto

Descripción generada automáticamente

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente