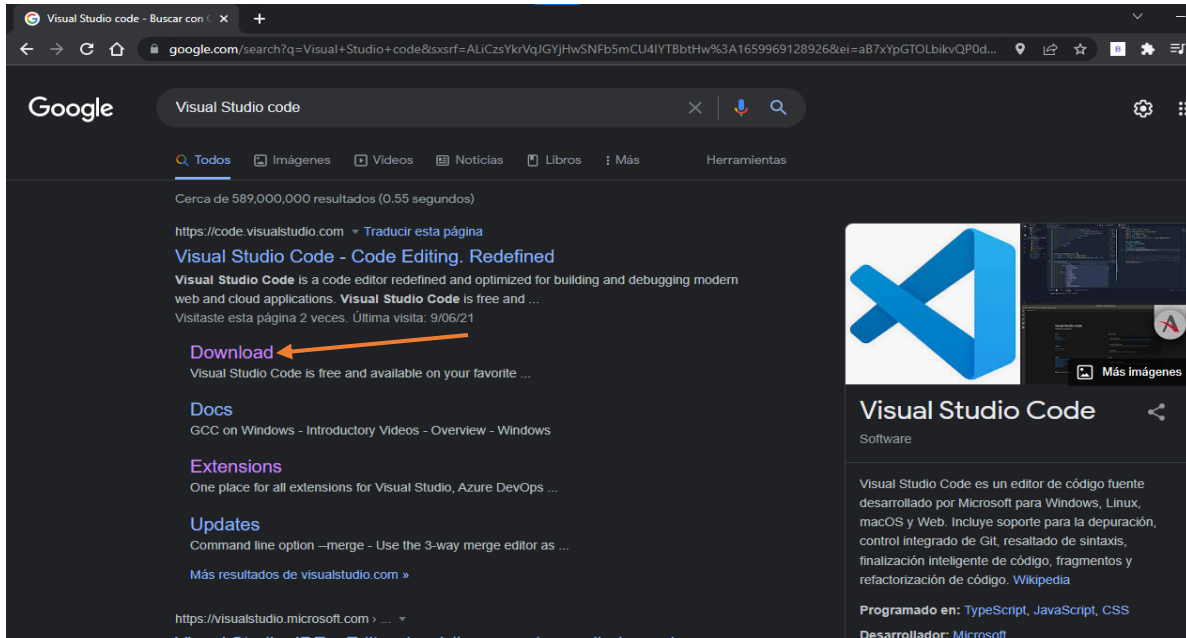
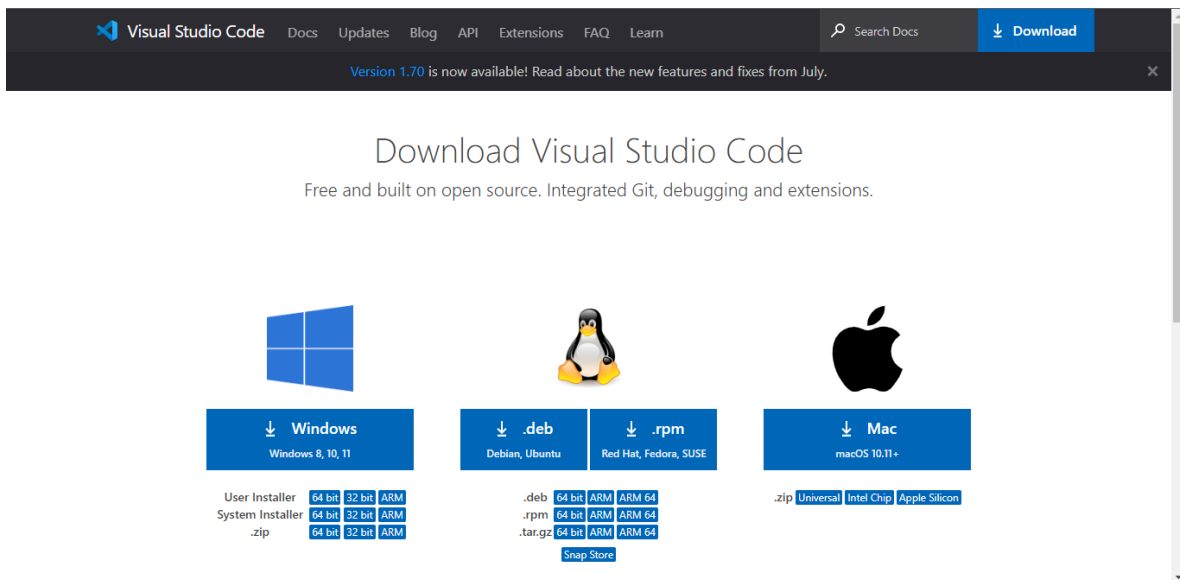


Proceso instalación Visual Studio Code

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



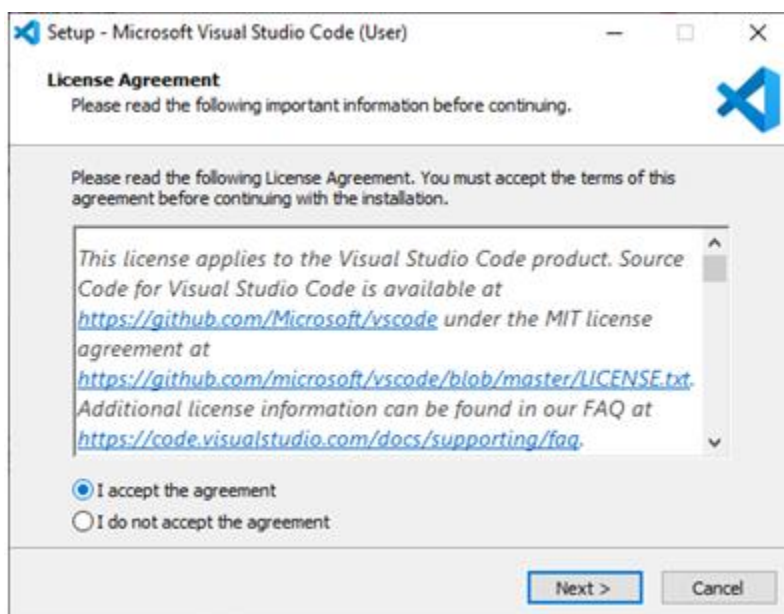
Paso 2: Seleccionamos el sistema operativo que tenemos y lo descargamos.



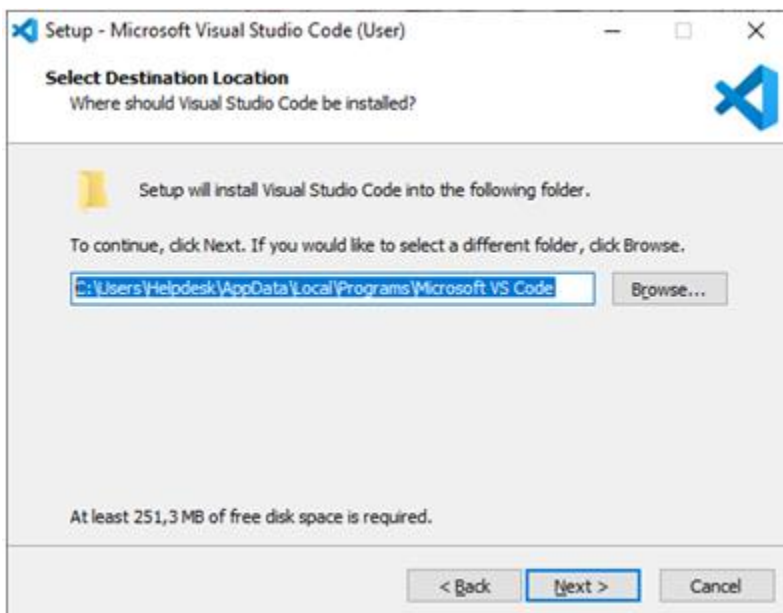
Paso 3: Al darle clic nos descargará un .exe, al cual le daremos clic encima.



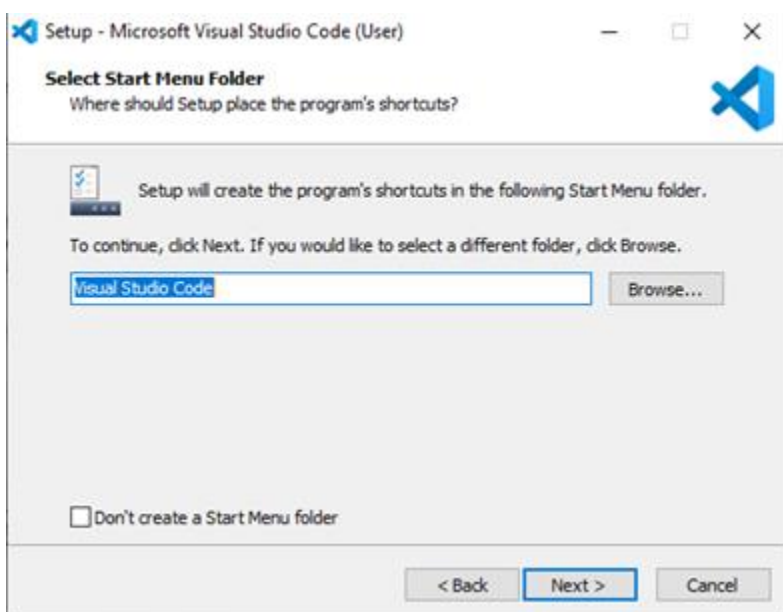
Paso 4: Lee y acepta el acuerdo de licencia. Haz clic en Next para continuar.



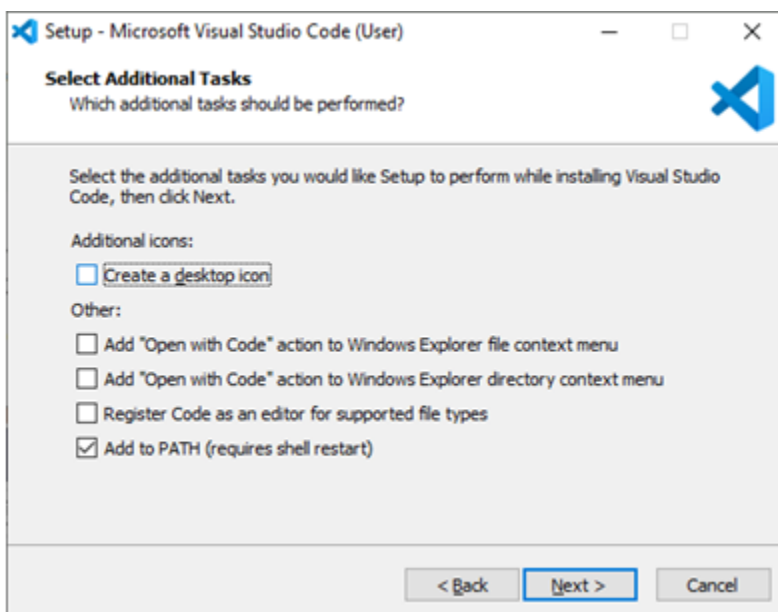
Paso 5: Puedes cambiar la ubicación de la carpeta de instalación o mantener la configuración predeterminada. Haz clic en Next para continuar.



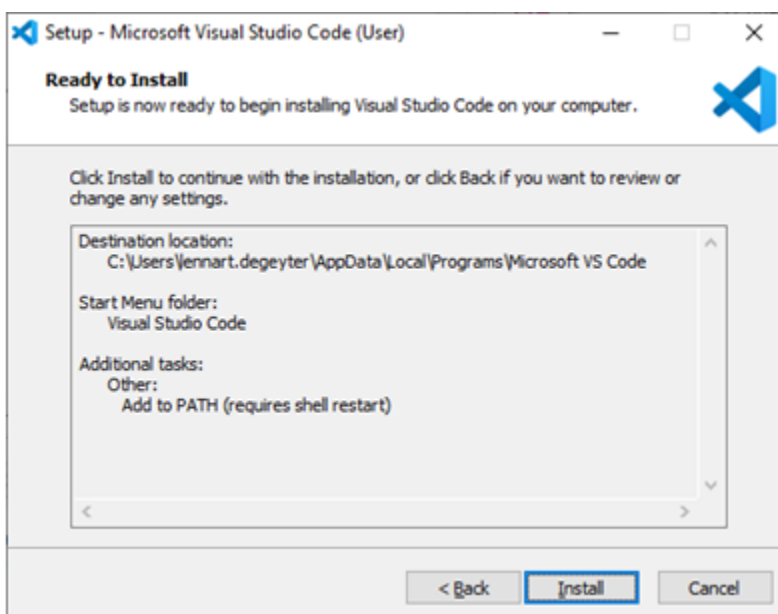
Paso 6: Elige si deseas cambiar el nombre de la carpeta de accesos directos en el menú Inicio o si no deseas instalar accesos directos en absoluto. Haz clic en Next.



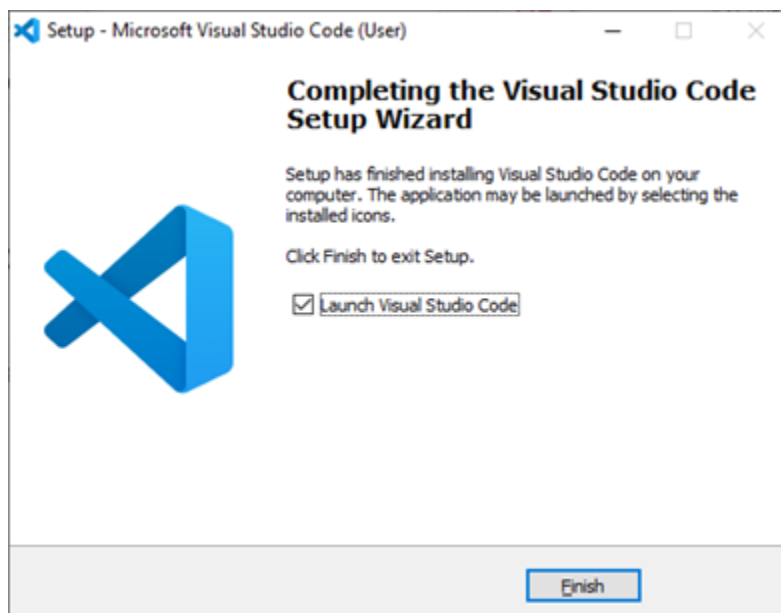
Paso 7: Selecciona las tareas adicionales, por ej. crear un icono en el escritorio o añadir opciones al menú contextual de Windows Explorer. Haz clic en Next.



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




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 “rating”.

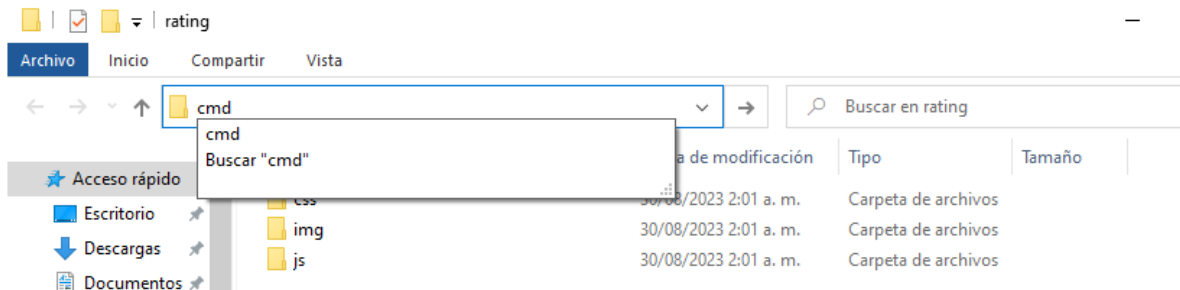
Nombre	Fecha de modificación	Tipo	Tamaño
 rating	30/08/2023 2:12 a. m.	Carpeta de archivos	

Paso 2: Dentro de esta, crearemos otras tres carpetas llamadas “css”, la cual contendrá todo lo estético de nuestra página, otra llamada “js” la cual contendrá la lógica de nuestra página y por último una llamada “img” la cual contendrá la imagen de nuestras estrellas.

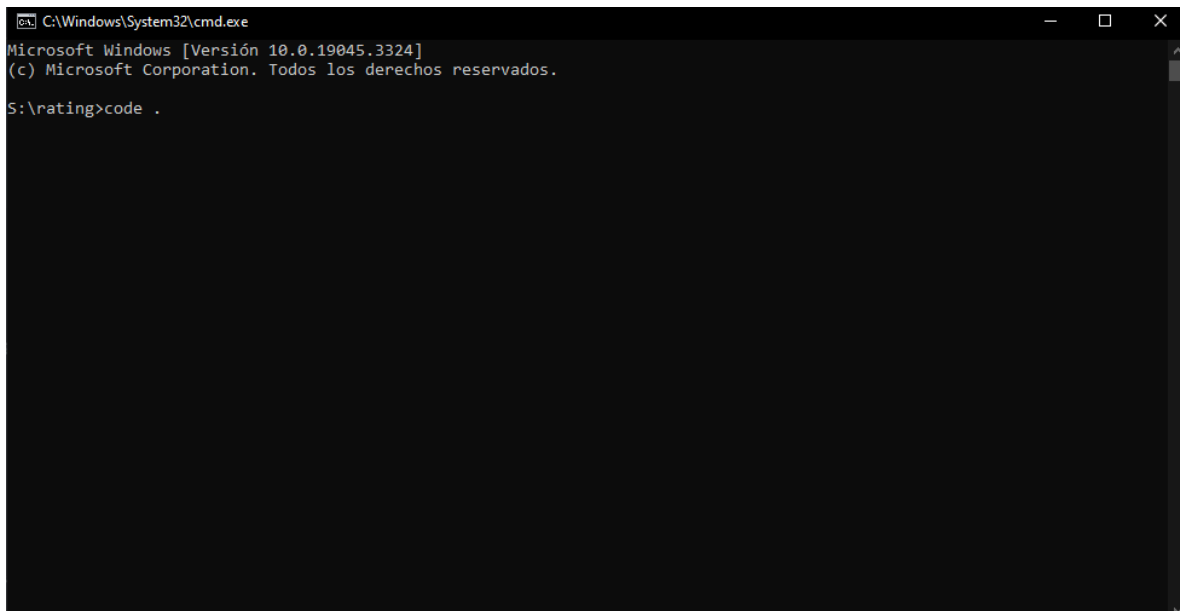
> Este equipo > adso (S:) > rating				Buscar en rating	
	Nombre	Fecha de modificación	Tipo	Tamaño	
ion	 css	18/08/2023 10:04 a. m.	Carpeta de archivos		
ón	 documentación	30/08/2023 2:00 a. m.	Carpeta de archivos		
ón	 img	18/08/2023 11:07 a. m.	Carpeta de archivos		

Tarjeta

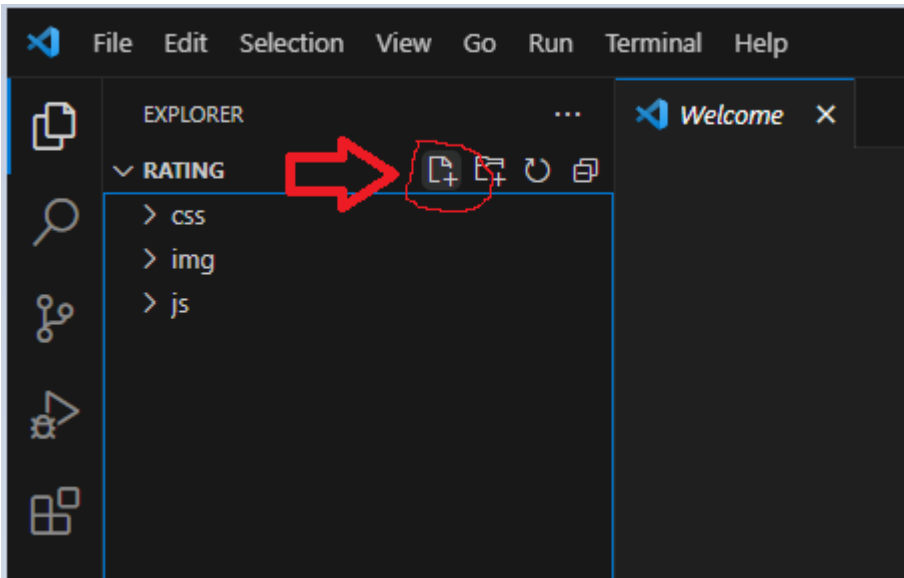
Paso 1: Para abrir el Visual Studio Code, haremos lo siguiente: Dentro de la carpeta raíz, en la barra superior, escribiremos cmd y le damos enter.



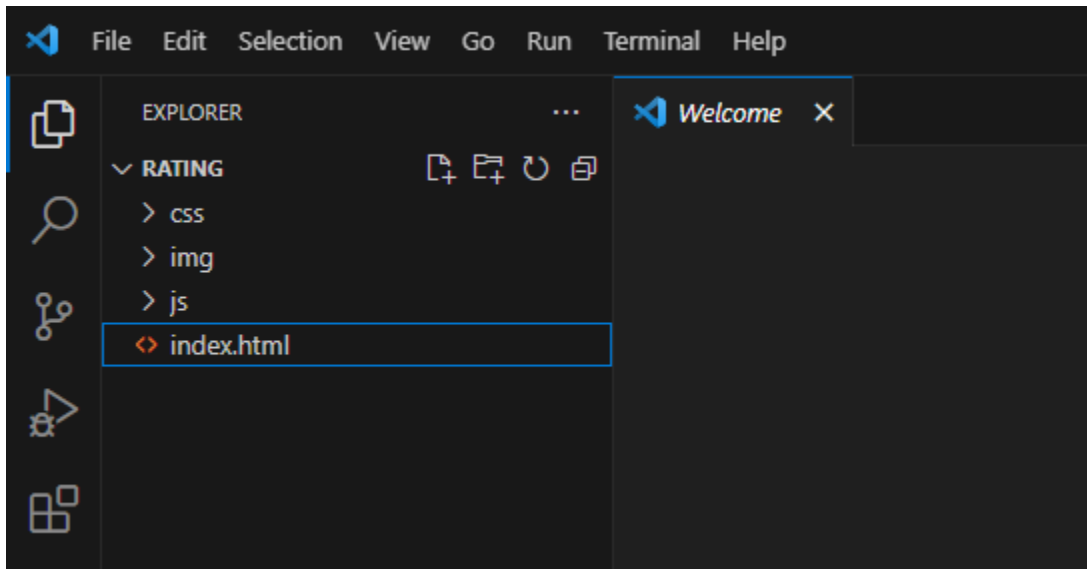
Paso 2: Eso nos abrirá una terminal, solo tendremos que escribir “code .”.



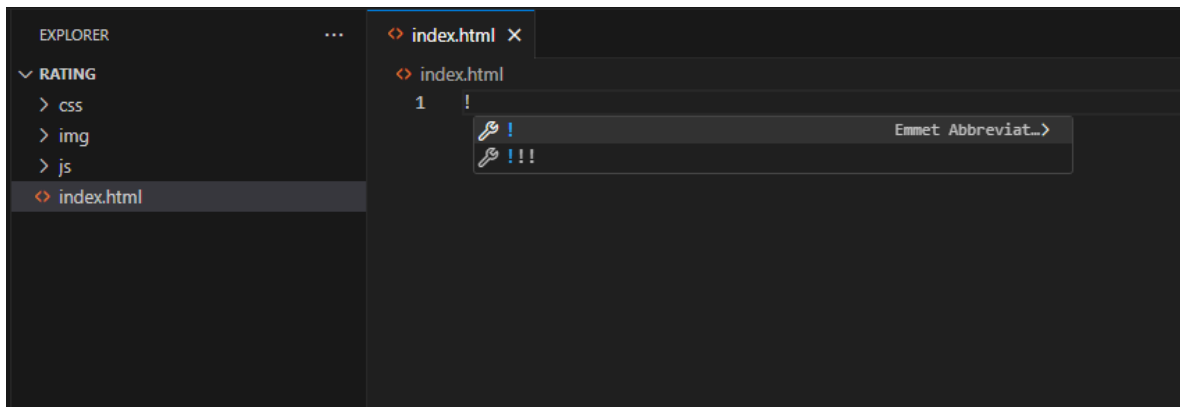
Paso 3: Esto nos abrirá el visual studio code. Ahora, ya que estamos dentro, crearemos un archivo llamado “index.html” dándole en la carpeta raíz dando click donde indica la flecha.



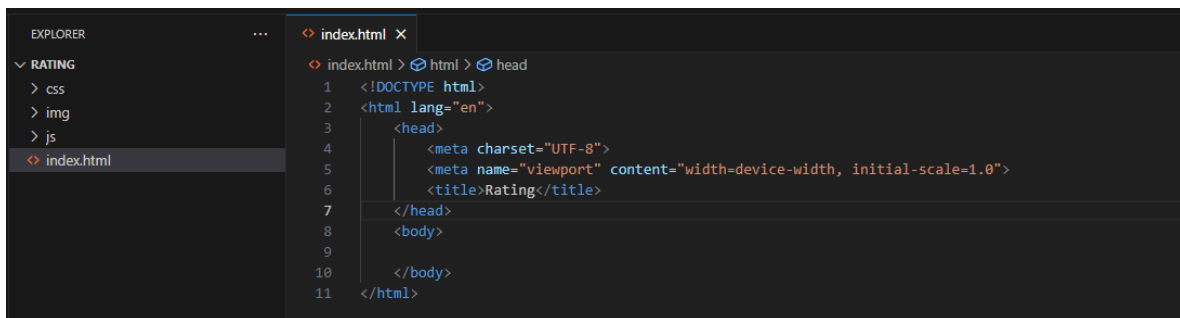
Paso 4: Llamaremos el archivo “index.html”



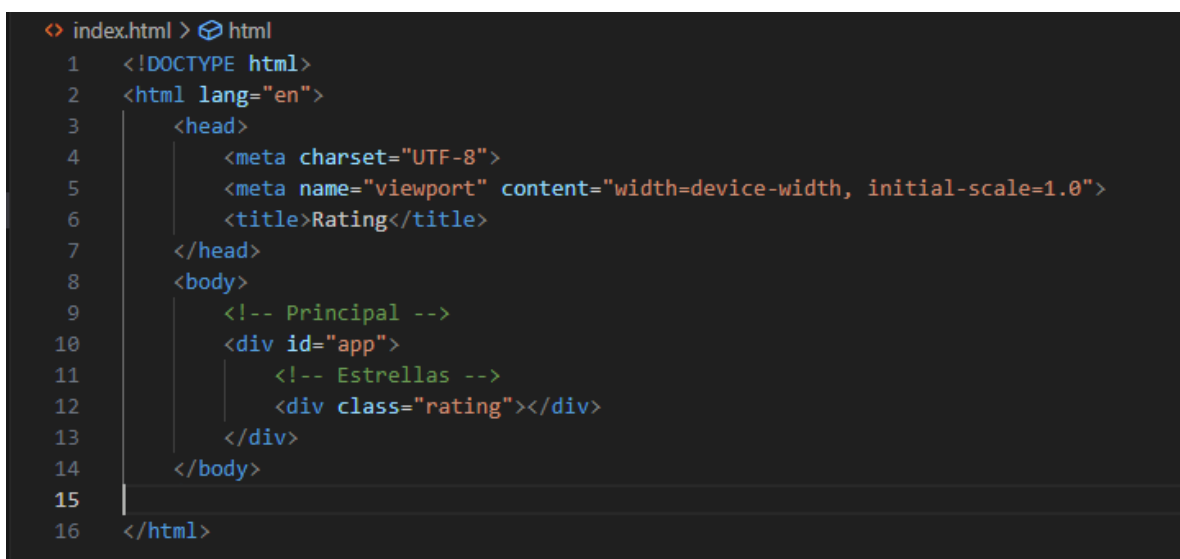
Paso 5: Una vez en el archivo recién creado, colocamos el código “!” y escogemos la primera opción.



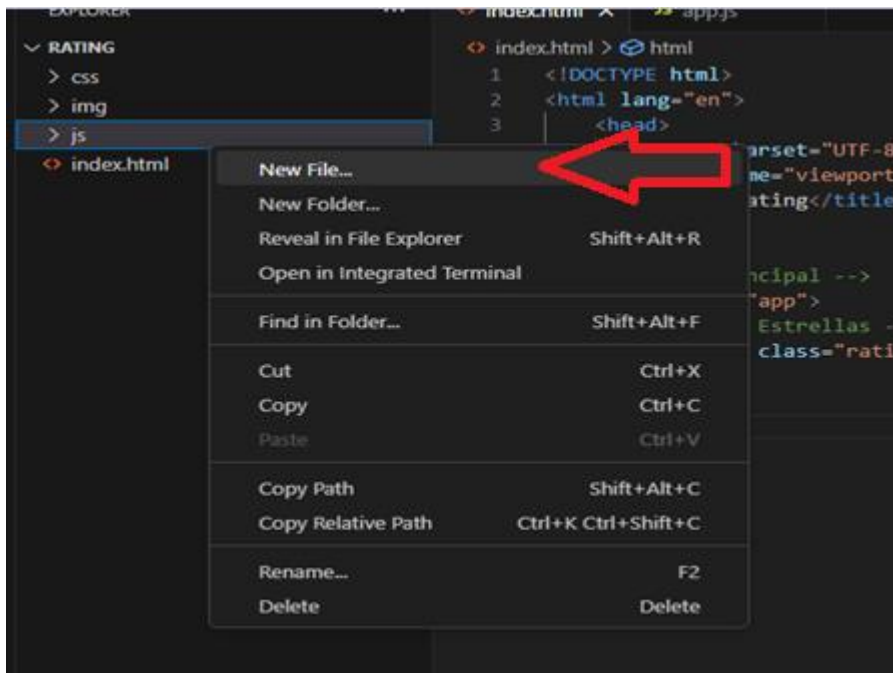
Paso 6: Una vez colocada la estructura, cambiaremos el título (línea 6) por “Rating”



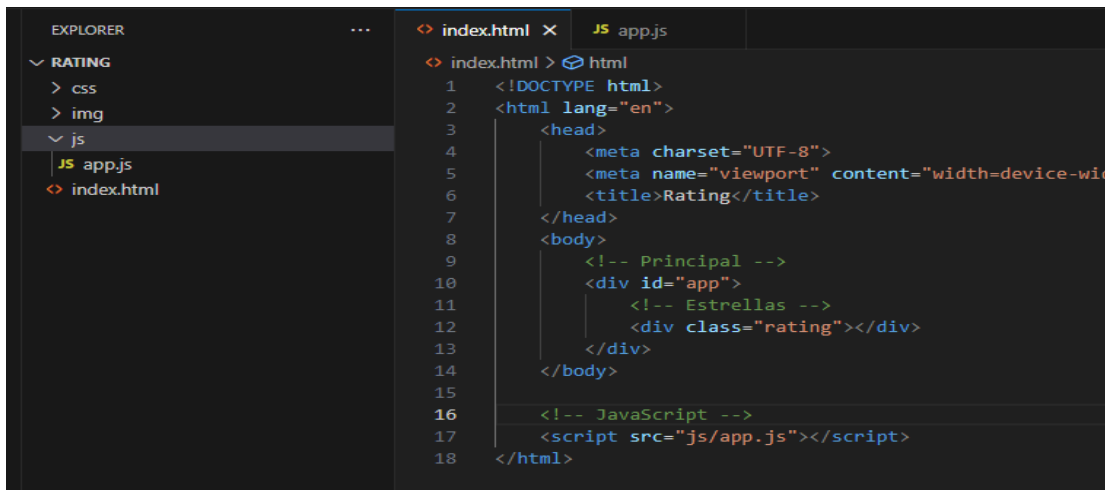
Paso 7: Ahora dentro del body crearemos un div principal para nuestra página (Línea 10) le pondremos un id llamado “app”, y además crearemos otro div dentro de ese mismo div (línea 12) y le pondremos la clase “rating”.



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



Paso 9: Una vez creado el archivo vamos a vincular el javascript en nuestro html (Línea 17)



Paso 10: Una vez vinculado el javascript iremos al archivo “app.js” y vamos a referenciar la clase “rating” de nuestro html y guardándolo en una variable (línea 3) También crearemos una variable para ver la cantidad de estrellas actuales (línea 5)

Y por último una constante la cual nos va a determinar la cantidad máxima de estrellas (línea 7).

```
index.html JS app.js X
js > JS app.js > ...
1 // Creamos una representación visual de una calificación mediante elementos div.
2 // Referenciar la clase rating del HTML, guardandolo en una variable.
3 const ratingContainer = document.querySelector(".rating");
4 // Variable para saber cual es la cantidad de estrellas actuales.
5 let currentValue = 0;
6 // Variable que va a determinar la cantidad máxima de estrellas
7 const limit = 10;
8
```

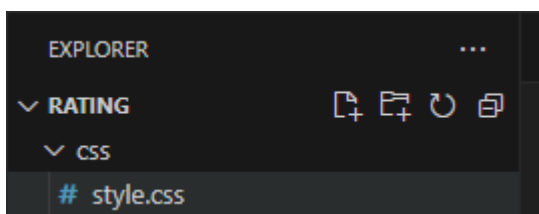
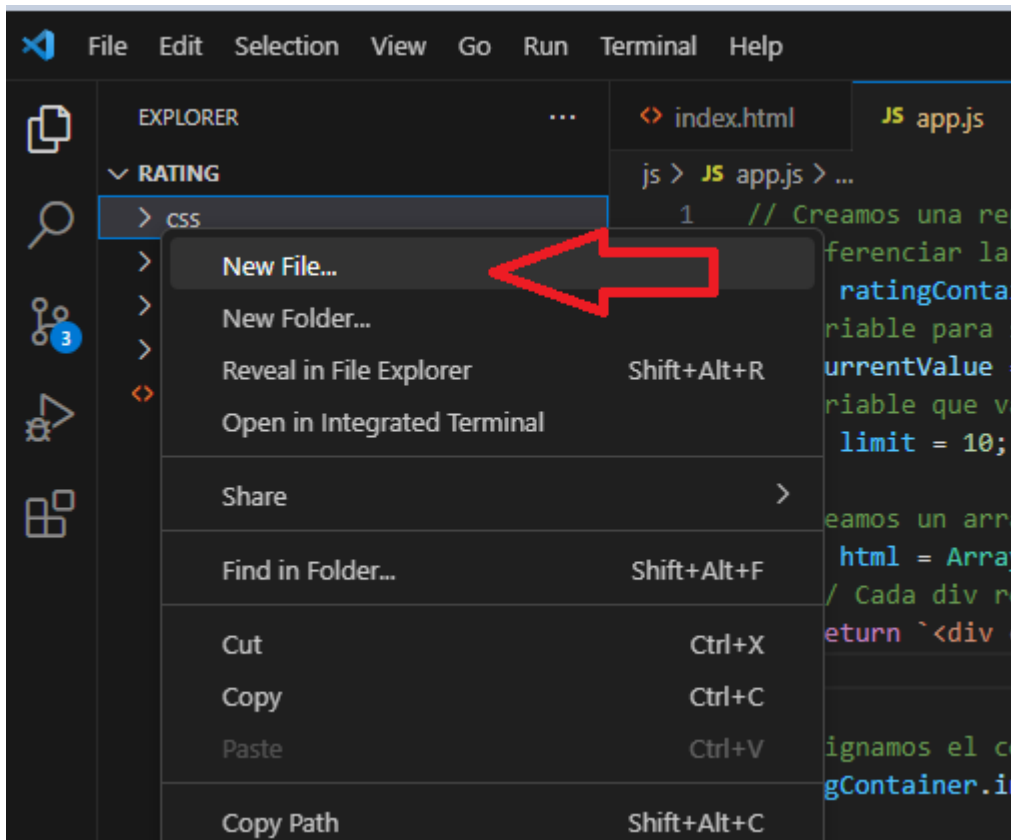
Paso 11: Ahora una variable el cual tendrá un array de elementos div usando el método map para recorrerlo (línea 10)

Dentro de ese array crearemos un código para representar cada div con una clase y atributos únicos (línea 12)

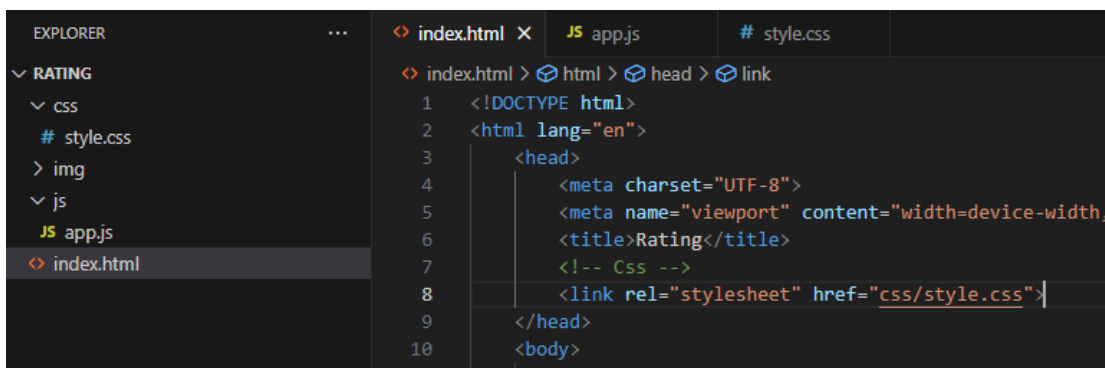
Por último, asignamos el contenido generado en el array “html” en el archivo “index.html” (línea 16)

```
9 // Creamos un array de elementos div usando el método map para recorrer el array.
10 const html = Array.from(Array(limit)).map((item, i) => {
11     // Cada div representa un elemento de calificación y tiene clases y atributos únicos.
12     return `<div class="item item-${i}" data-pos="${i}"></div>`;
13 });
14
15 // Asignamos el contenido generado en el array "html" al contenedor de calificación en el HTML.
16 ratingContainer.innerHTML = html.join("");
17
```

Paso 12: Ahora vamos a visualizar como está quedando, el primer paso será comenzar el “CSS” iremos en la barra de la izquierda a nuestra carpeta llamada “css” y crearemos un nuevo archivo



Paso 13: El archivo se llamará “style.css” y lo vinculamos en nuestro archivo llamado “index.html” (línea 8)

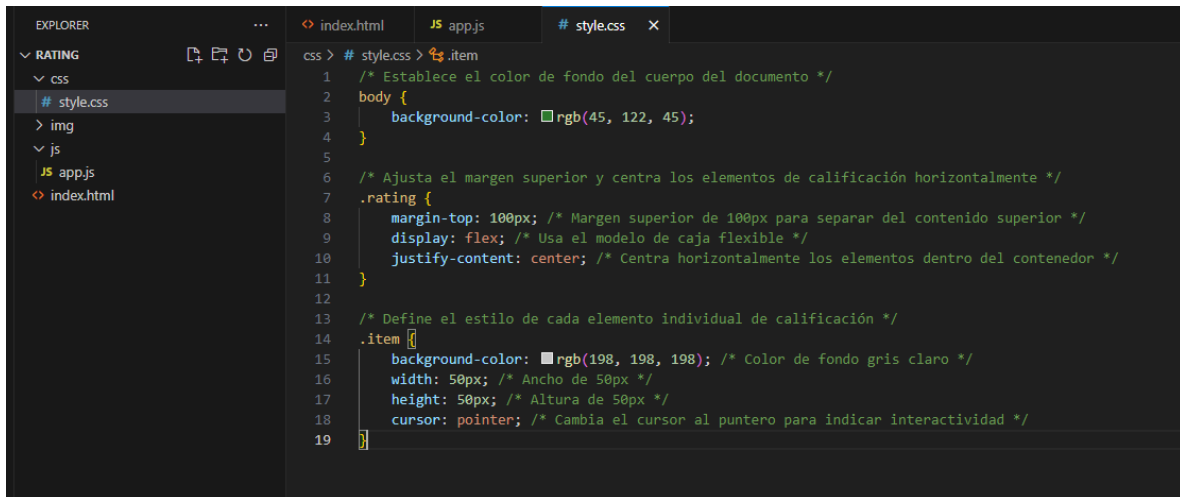


Paso 14: Ahora iremos al archivo que acabamos de crear ("style.css") y comenzamos con el diseño de nuestra página.

Lo primero que modificaremos será el color del fondo (línea 2)

Ahora vamos a cambiar un poco el diseño de rating para ajustarlo (desde la línea 7 a la 11)

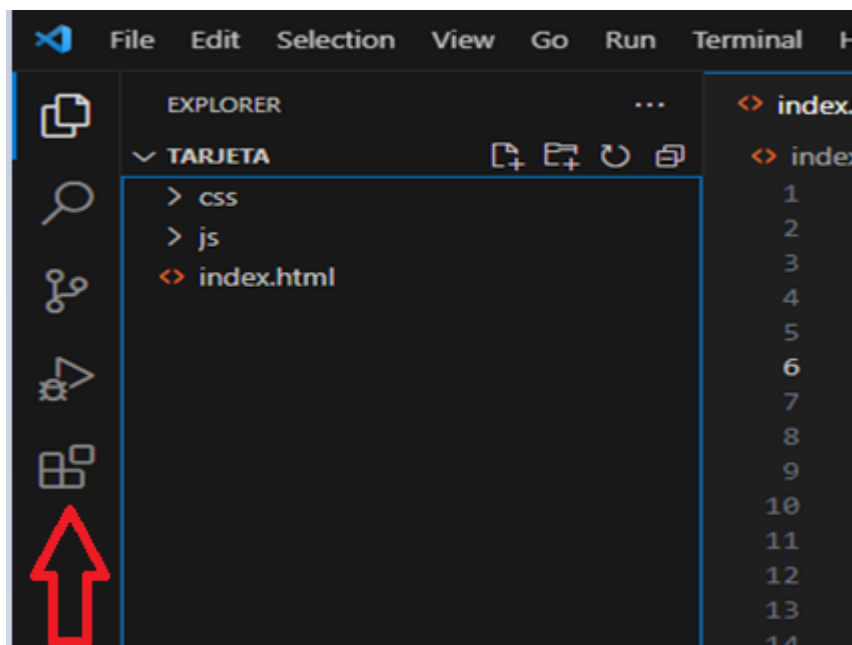
Por último, vamos a definir el estilo de los items los cuales se verán reflejados en las estrellas (Desde la línea 14 a la 19)



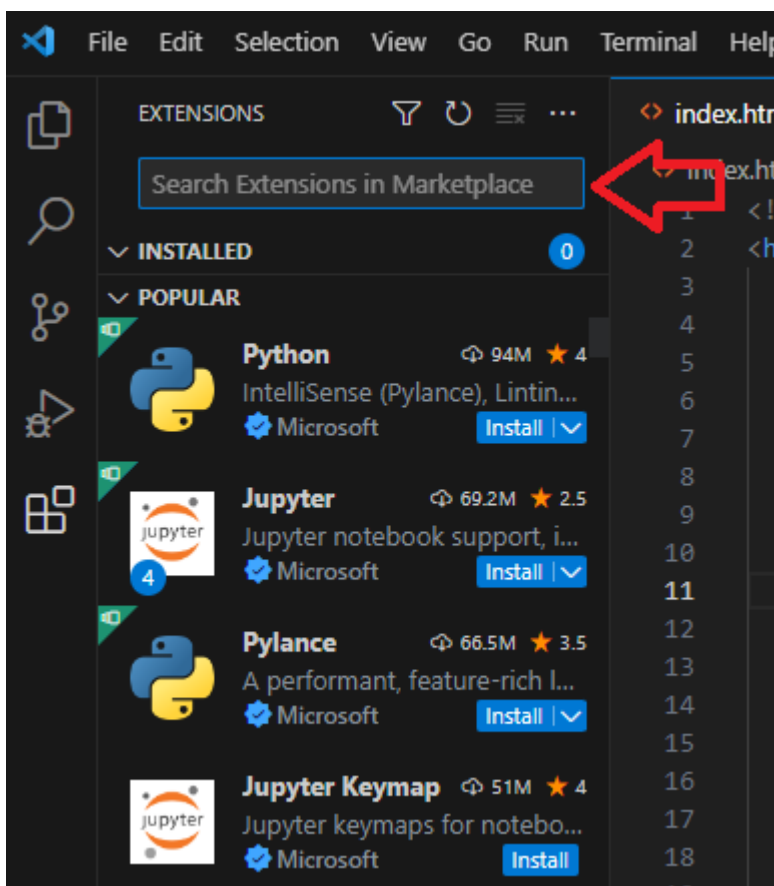
```
EXPLORER
  < index.html
  JS app.js
  # style.css
  > img
  > js
  JS app.js
  < index.html

css > # style.css > .item
1  /* Establece el color de fondo del cuerpo del documento */
2  body {
3    background-color: rgb(45, 122, 45);
4  }
5
6  /* Ajusta el margen superior y centra los elementos de calificación horizontalmente */
7  .rating {
8    margin-top: 100px; /* Margen superior de 100px para separar del contenido superior */
9    display: flex; /* Usa el modelo de caja flexible */
10   justify-content: center; /* Centra horizontalmente los elementos dentro del contenedor */
11  }
12
13  /* Define el estilo de cada elemento individual de calificación */
14  .item {
15    background-color: rgb(198, 198, 198); /* Color de fondo gris claro */
16    width: 50px; /* Ancho de 50px */
17    height: 50px; /* Altura de 50px */
18    cursor: pointer; /* Cambia el cursor al puntero para indicar interactividad */
19  }
```

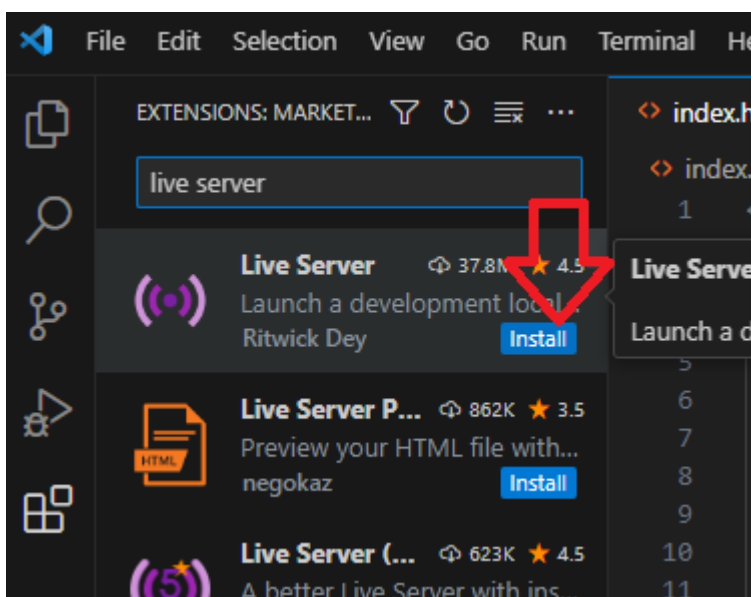
Paso 15: Ahora para poder visualizar como se esta viendo nuestro proyecto le daremos click al icono de abajo a la izquierda.



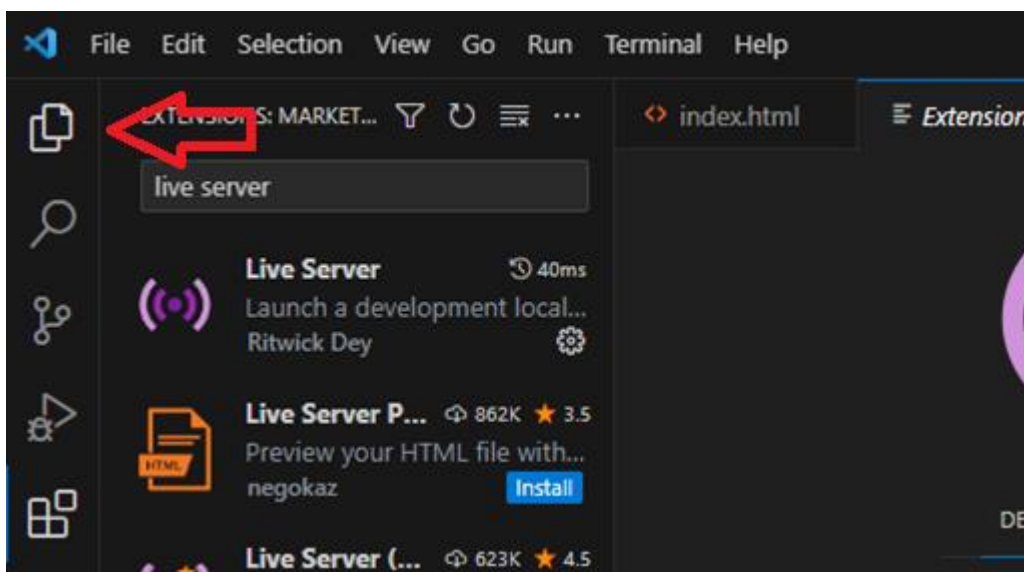
Paso 16: Nos saldrá esta pestaña y buscaremos live server, en la barra de búsqueda.



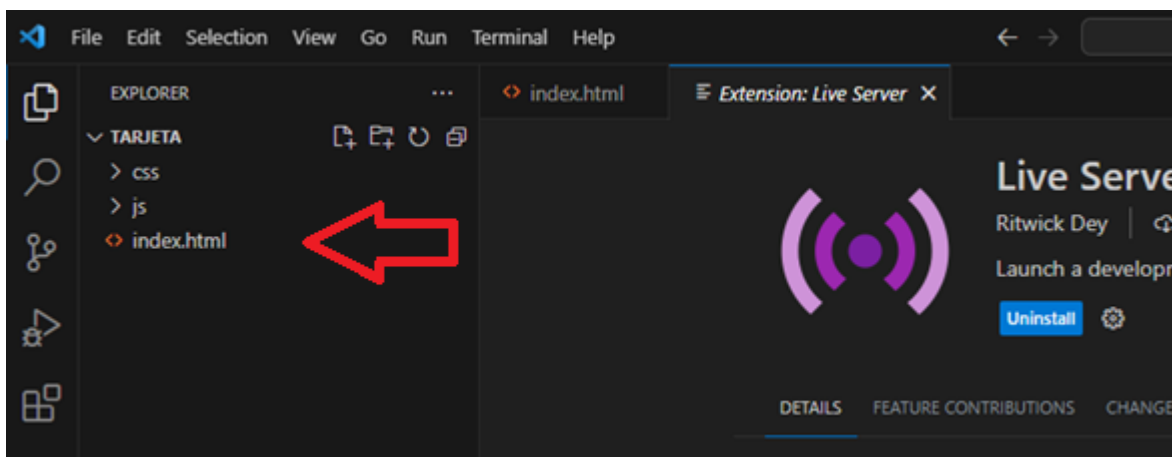
Paso 17: Ahora que encontramos lo que necesitamos, le damos click a “install”



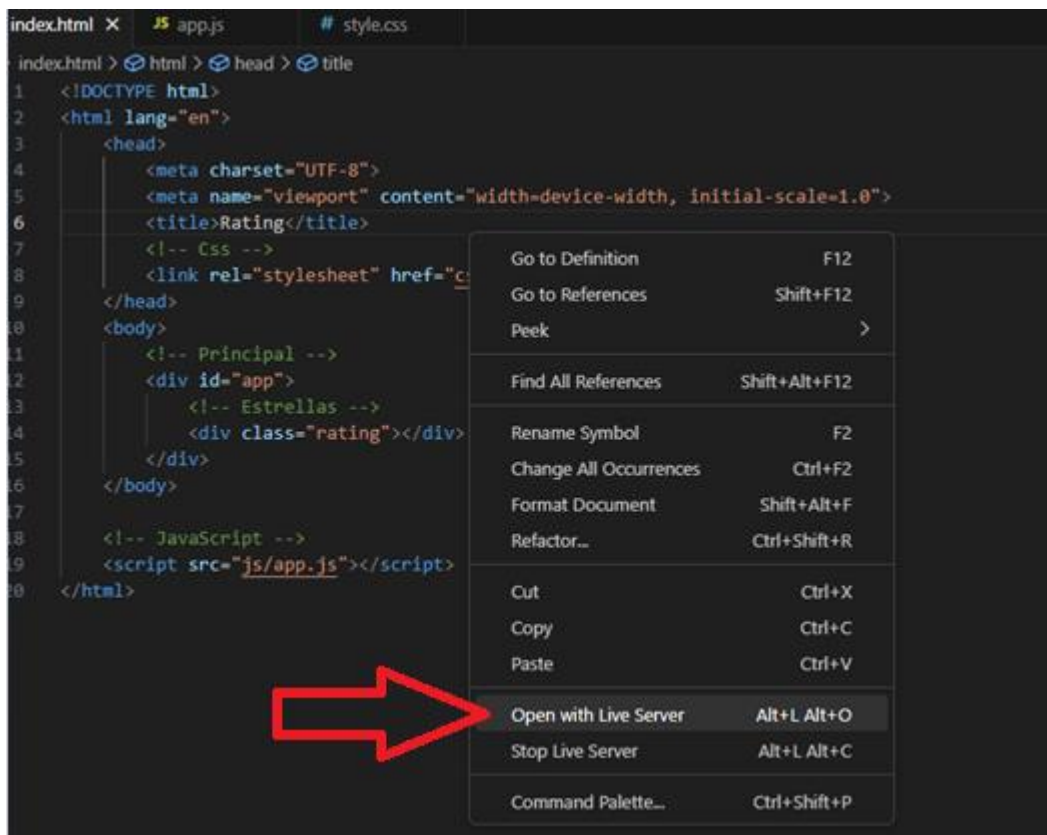
Paso 18: Ahora que tenemos instalado el live server nos debería de aparecer algo así donde podremos volver a nuestro proyecto dando click al explorador.



Paso 19: Una vez te sale el explorador volvemos al Index.html



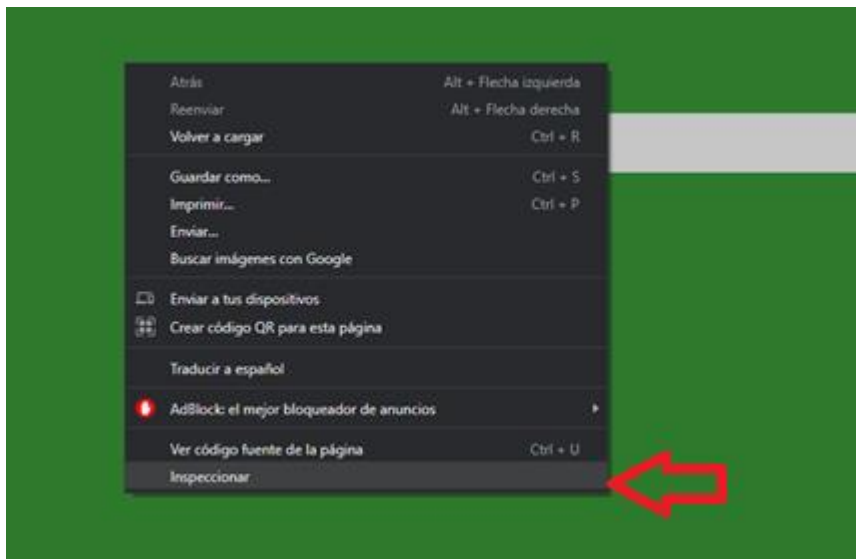
Paso 20: Una vez volvemos al index.html damos click derecho en cualquier parte del código y damos click izquierdo a “open with live server”



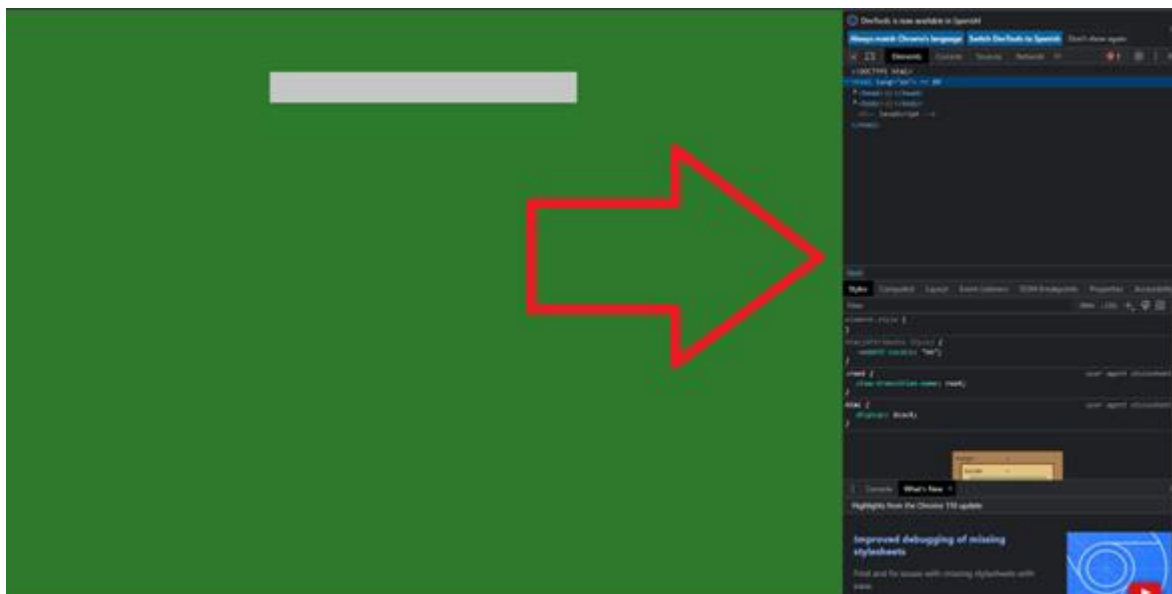
Paso 21: Ahora nuestra página estará así



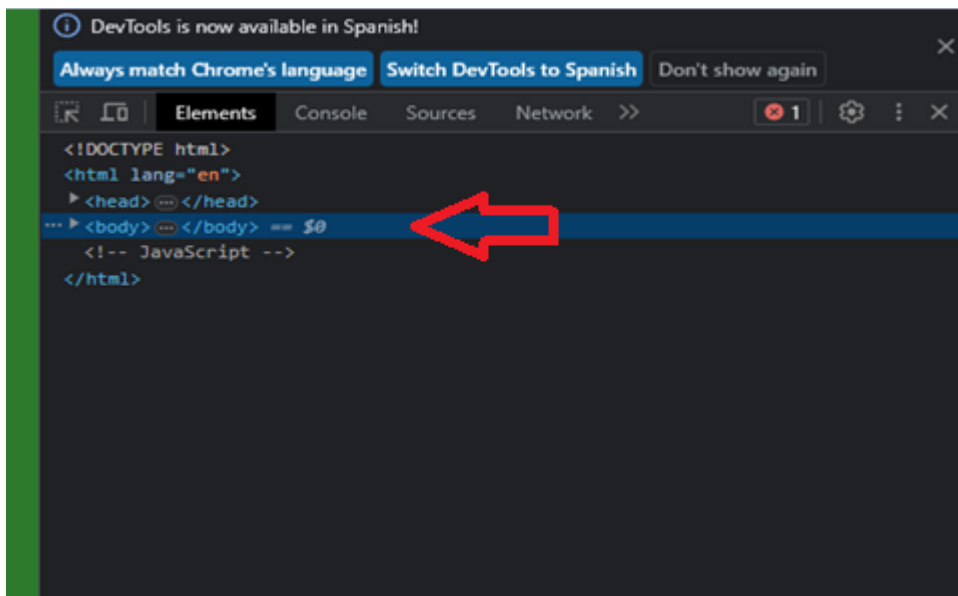
Paso 22: Vamos a mirar si la barra de la mitad (Donde irán las estrellas) si está funcionando correctamente. Para esto daremos click en cualquier parte de la pagina y escogemos la opción llamada (inspeccionar)



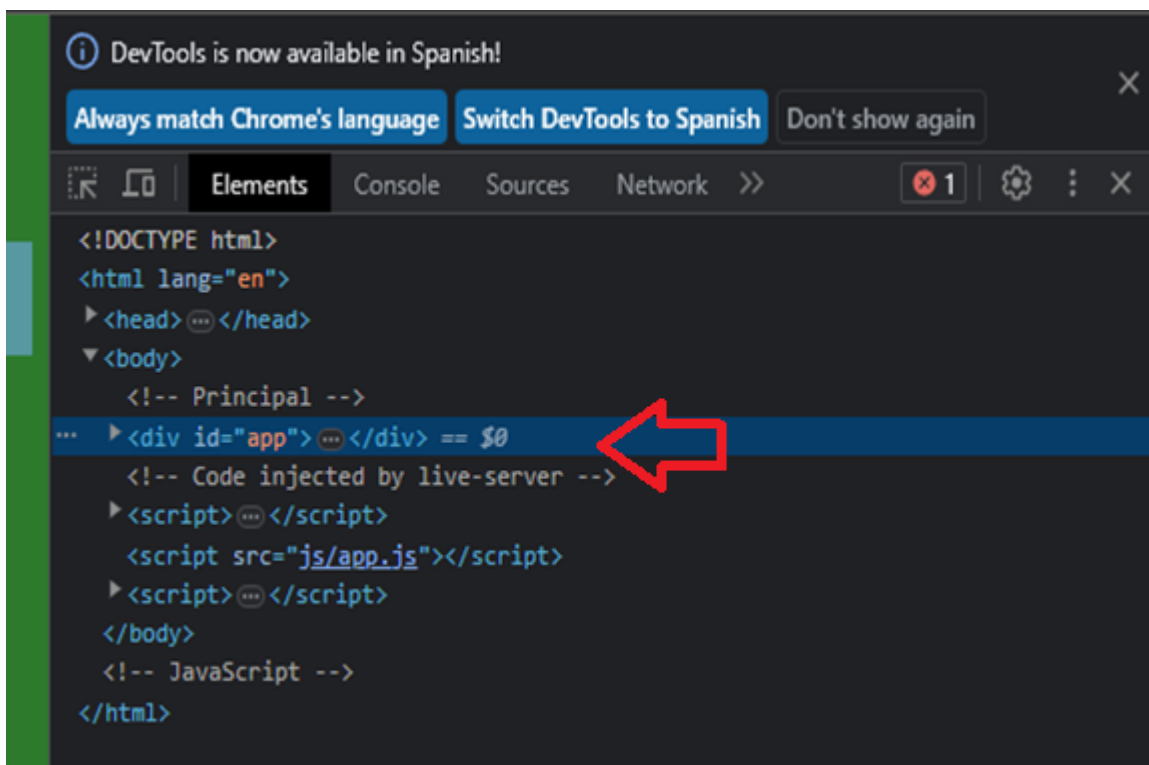
Paso 23: Se nos abrirá este panel a la derecha



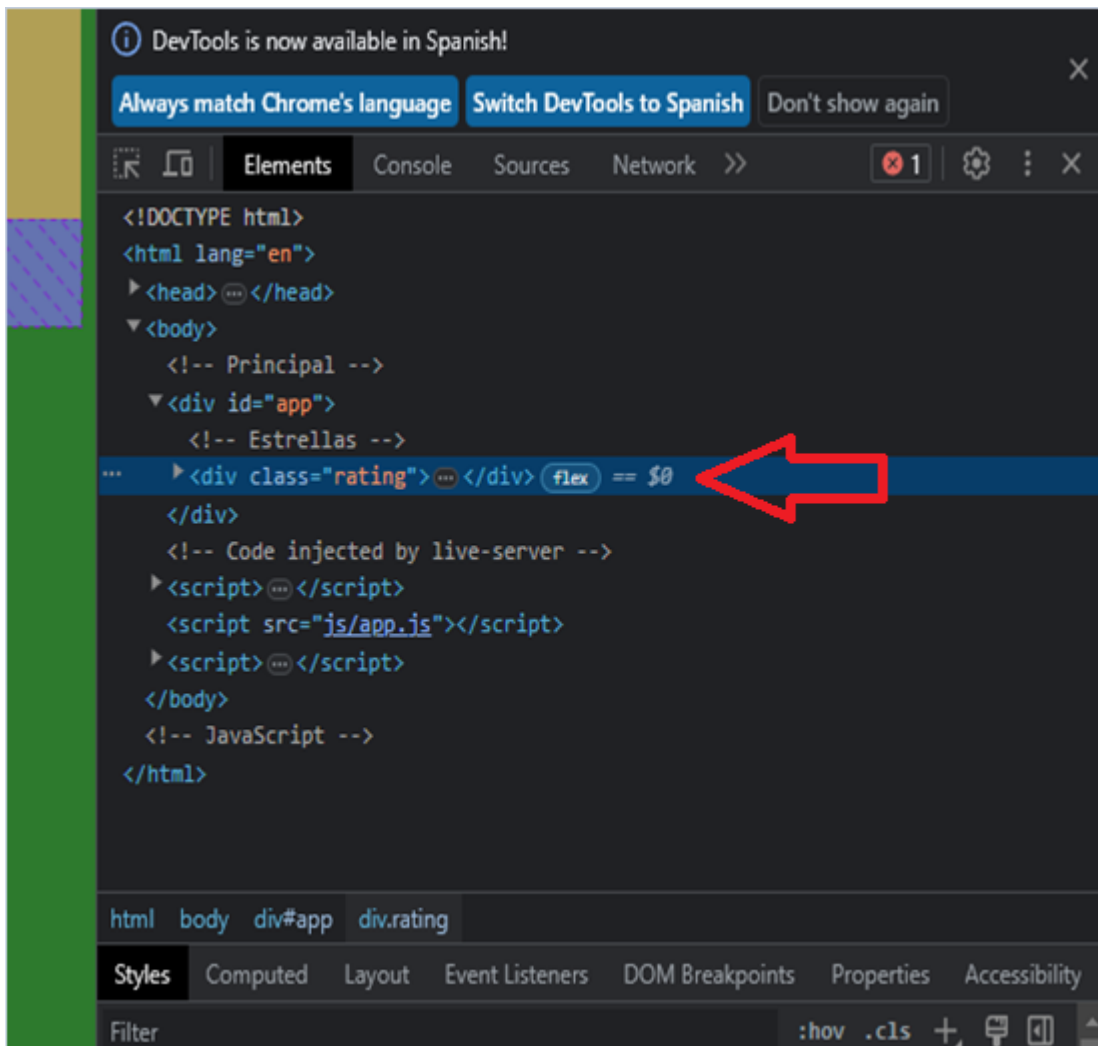
Paso 24: Ahora daremos click en la opción “body” para desplegar el código



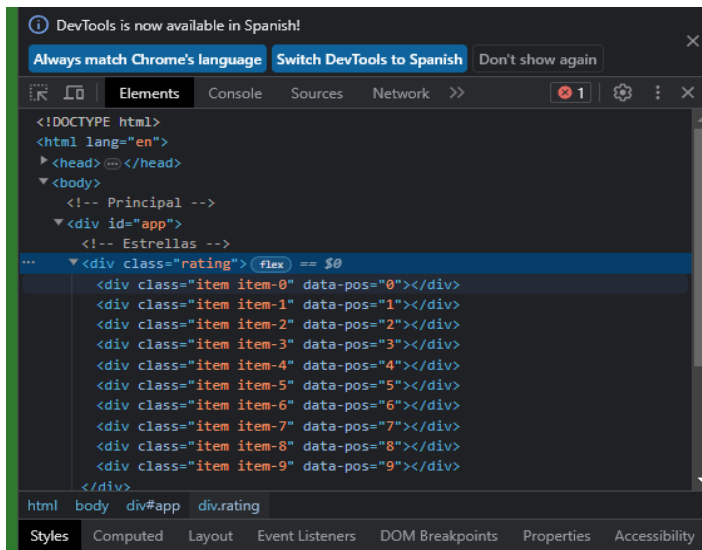
Paso 25: Una vez abierto el código dentro de “body” iremos a buscar el div con el id “app” y desplegamos su código.



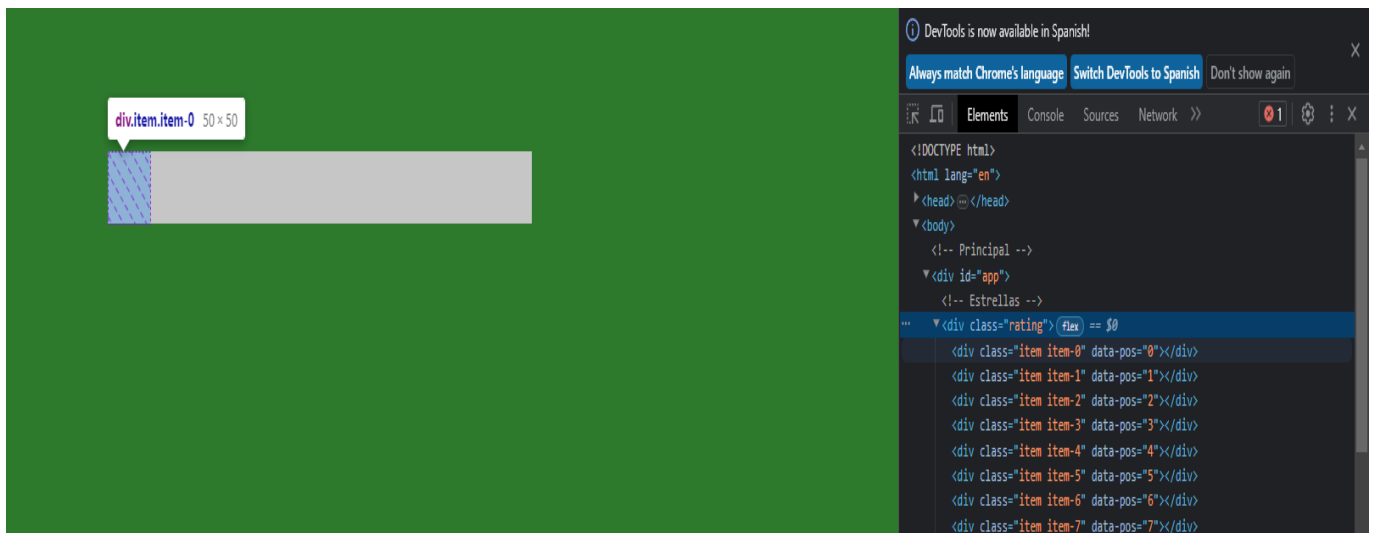
Paso 26: Una vez desplegado el código de app iremos al div con la clase “rating” y la desplegamos



Paso 27: Ahora que desplegamos el div podemos ver que dentro hay varios items individuales con la con la cantidad que pusimos en la cantidad máxima. (Paso 10), y se nos debe de estar viendo algo así.



The screenshot shows the Chrome DevTools 'Elements' panel. The DOM tree is expanded to show a `<div class="rating">` element. Inside this div, there are ten `<div class="item item-0" data-pos="0">` through `<div class="item item-9" data-pos="9">` elements. The 'Styles' pane at the bottom shows the 'flex' property is set to `== $0`. The breadcrumb at the bottom indicates the path: `html > body > div#app > div.rating`.



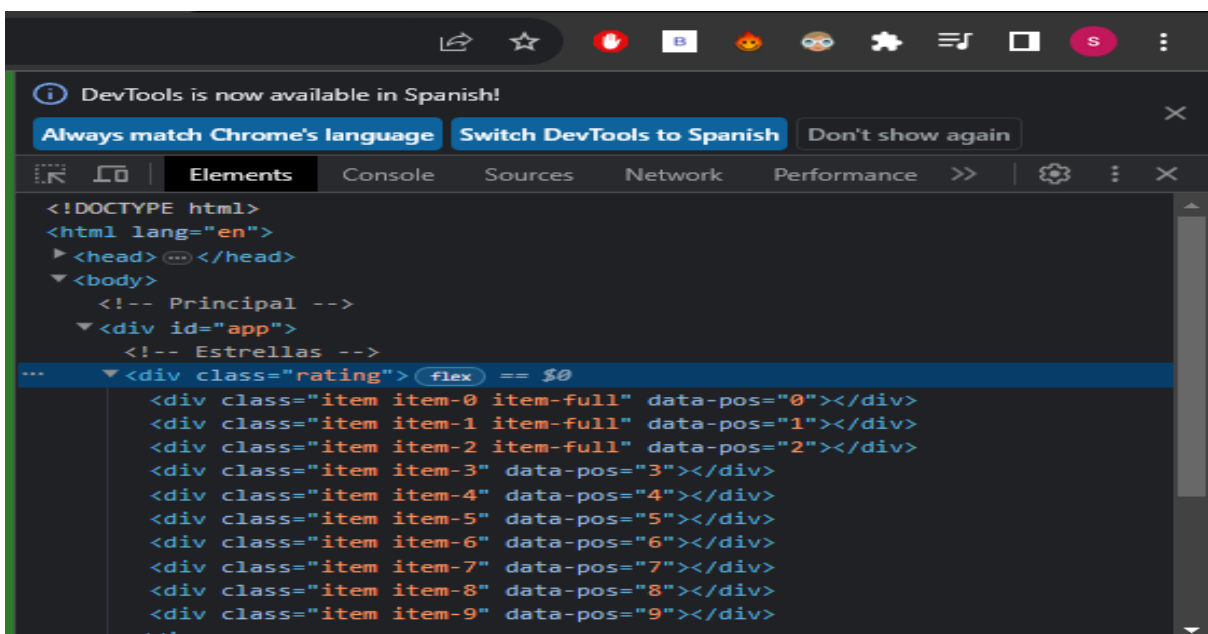
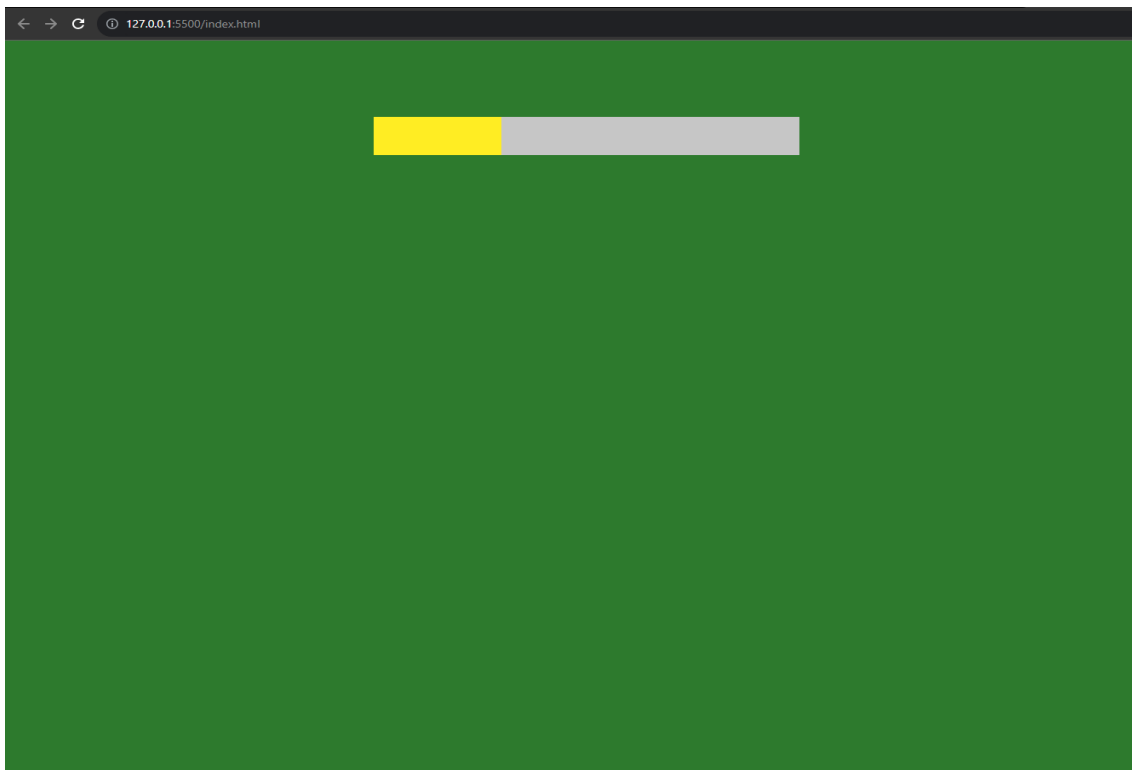
Paso 28: Una vez vimos cómo va quedando nuestra página iremos de nuevo al archivo “app.js” , comenzaremos asignando el contenido generado en el array al contenedor de calificación del HTML (línea 16) ahora seleccionamos los objetos con la clase “.item” y le agregamos un evento al pasar el mouse por encima (línea 19 a línea 20), ahora obtenemos dentro de ese evento la posición del elemento (línea 22) y creamos un condicional para que en caso se pase la posición no haga nada(línea 25)

```
15 // Asignamos el contenido generado en el array "html" al contenedor de calificación en el HTML.
16 ratingContainer.innerHTML = html.join("");
17
18 // Selecciona todos los elementos con la clase ".item" y agrega un oyente para el evento "mouseover"
19 document.querySelectorAll(".item").forEach((item) => {
20     item.addEventListener("mouseover", (e) => {
21         // Obtiene la posición del elemento en función del atributo "data-pos"
22         const pos = item.getAttribute("data-pos");
23
24         // Si el valor actual es igual a la posición más 1, no hace nada
25         if (currentValue === parseInt(pos) + 1) {
26             return;
27         }
28     });
29 });
30
```

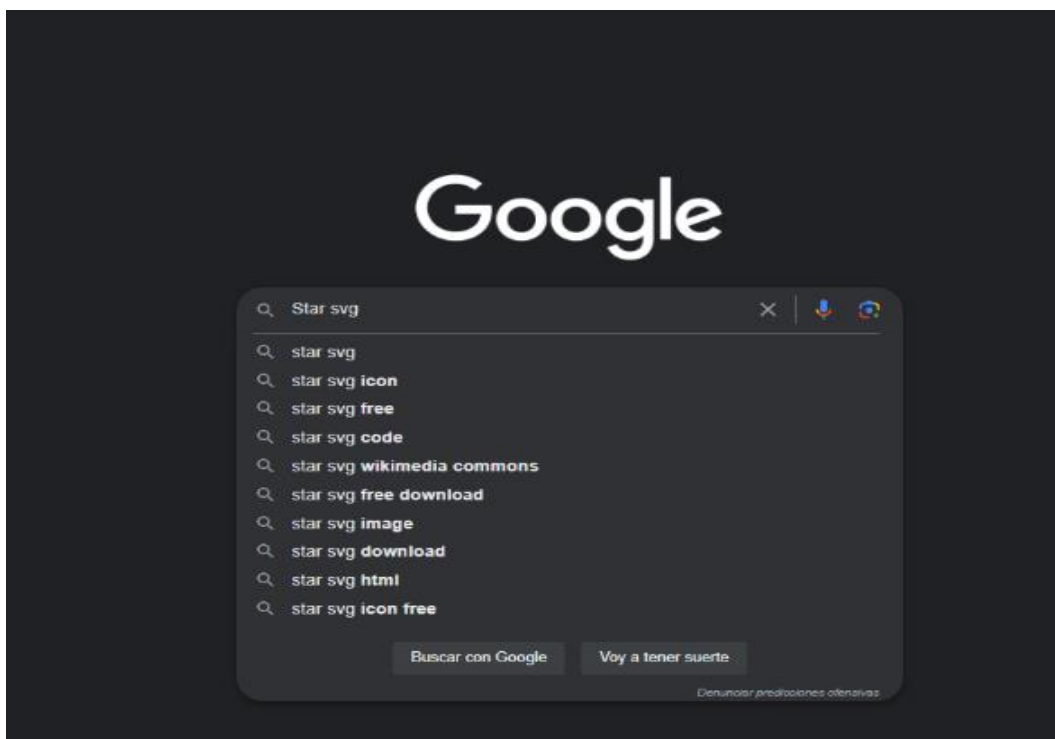
Paso 29: Ahora dentro de ese condicional vamos a seleccionar los items y les vamos a remover la clase “item-full” a todos los elementos con la clase “.item” cuando quitamos el ratón de encima

```
23
24 // Si el valor actual es igual a la posición más 1, no hace nada
25 if (currentValue === parseInt(pos) + 1) {
26     return;
27 }
28
29 // Remueve la clase "item-full" de todos los elementos con la clase ".item"
30 document.querySelectorAll(".item").forEach((item) => {
31     if (item.classList.contains("item-full")) {
32         item.classList.remove("item-full");
33     }
34 });
35
36 });
37
```

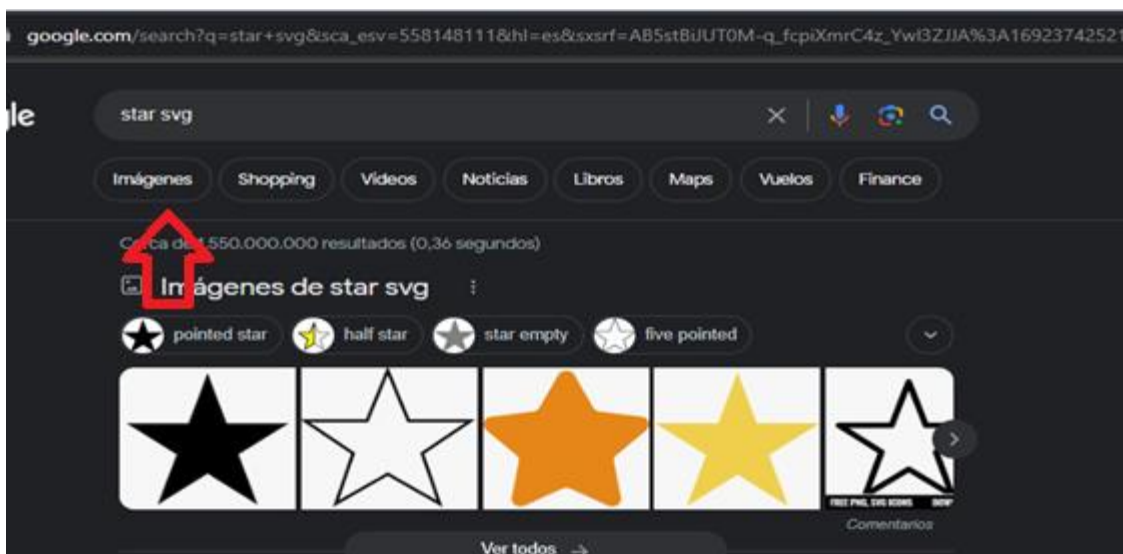
Paso 30: Ahora miramos como va quedando nuestro proyecto (paso 20) y en la consola podremos observar los elementos como se quita la clase o se pone según si ponemos el ratón encima o lo quitamos



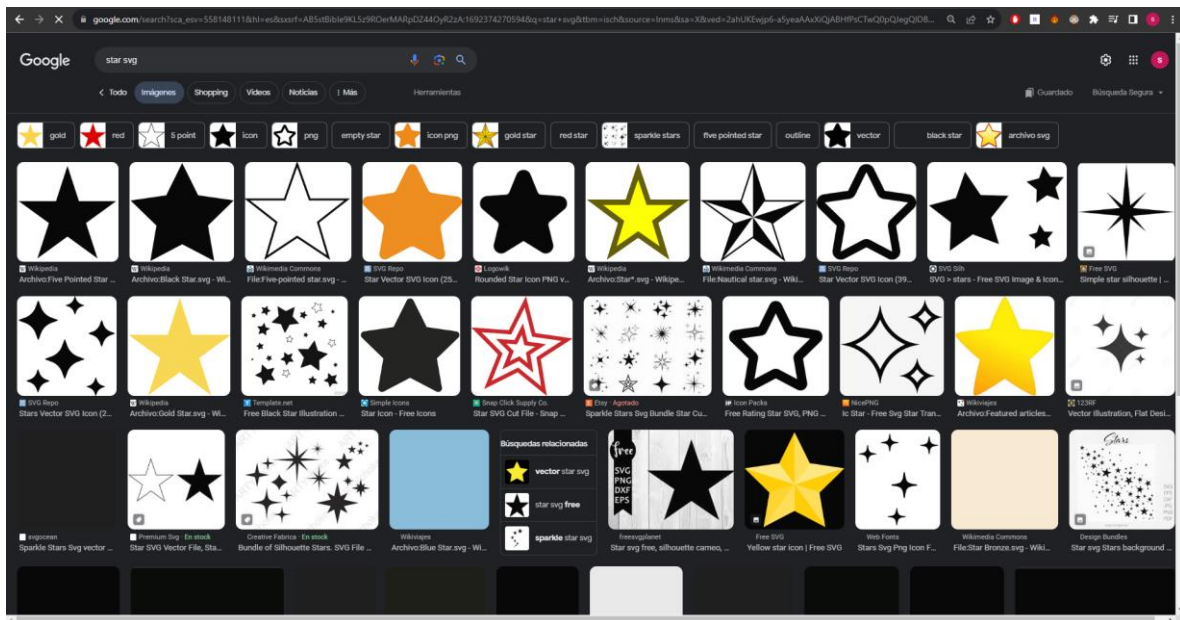
Paso 31: Ahora vamos a crear el diseño para la puntuación con unas estrellas, lo primero que haremos será ir a Google y buscaremos una estrella importante que sea SVG para que lo detecte bien el código.



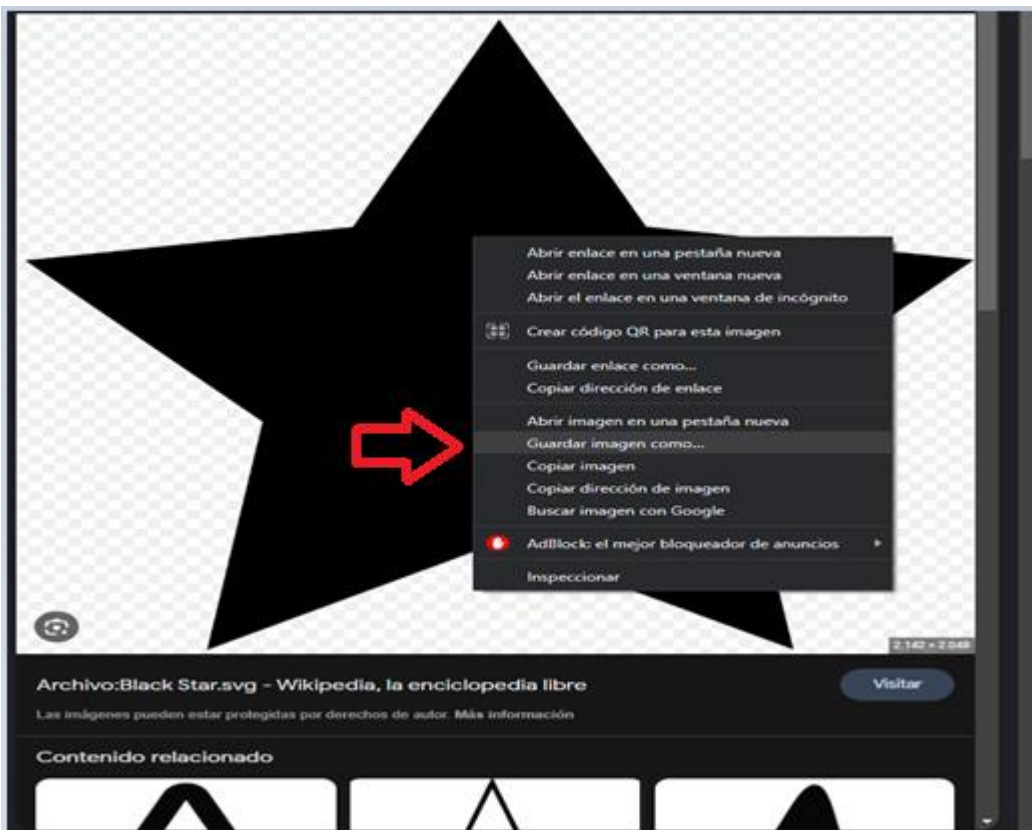
Paso 32: Iremos a imágenes.



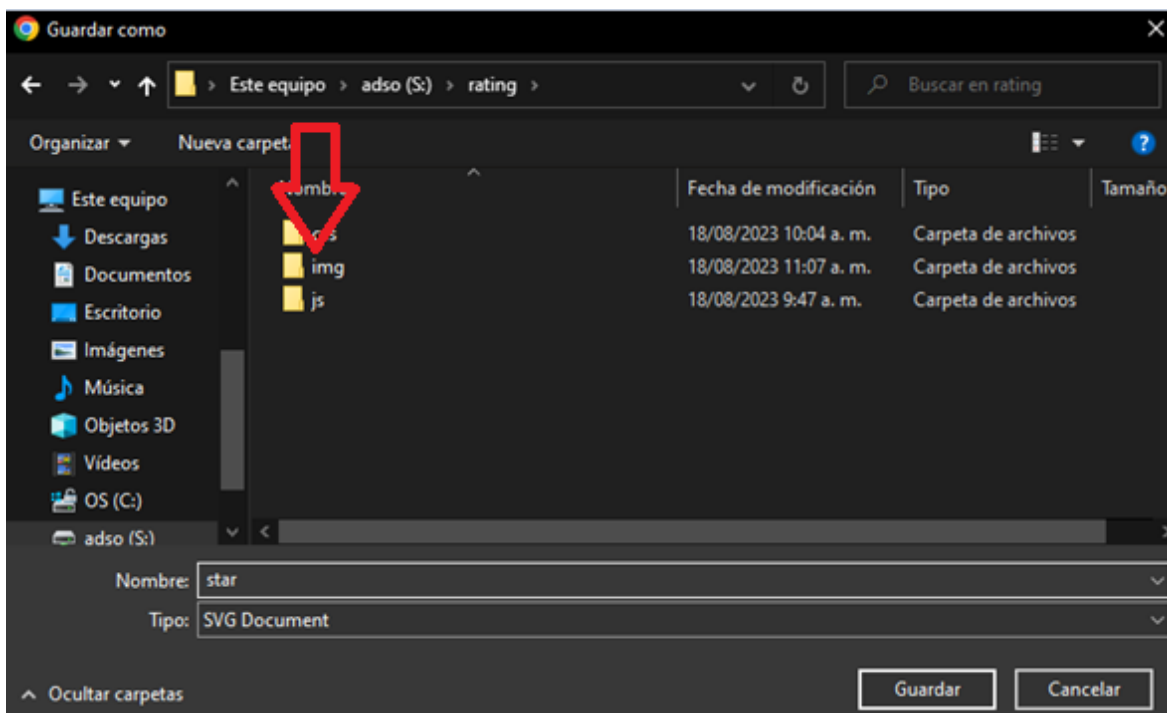
Paso 33: Ahora buscamos la estrella que nos parezca más bonita, recuerda que es mejor si termina en .svg



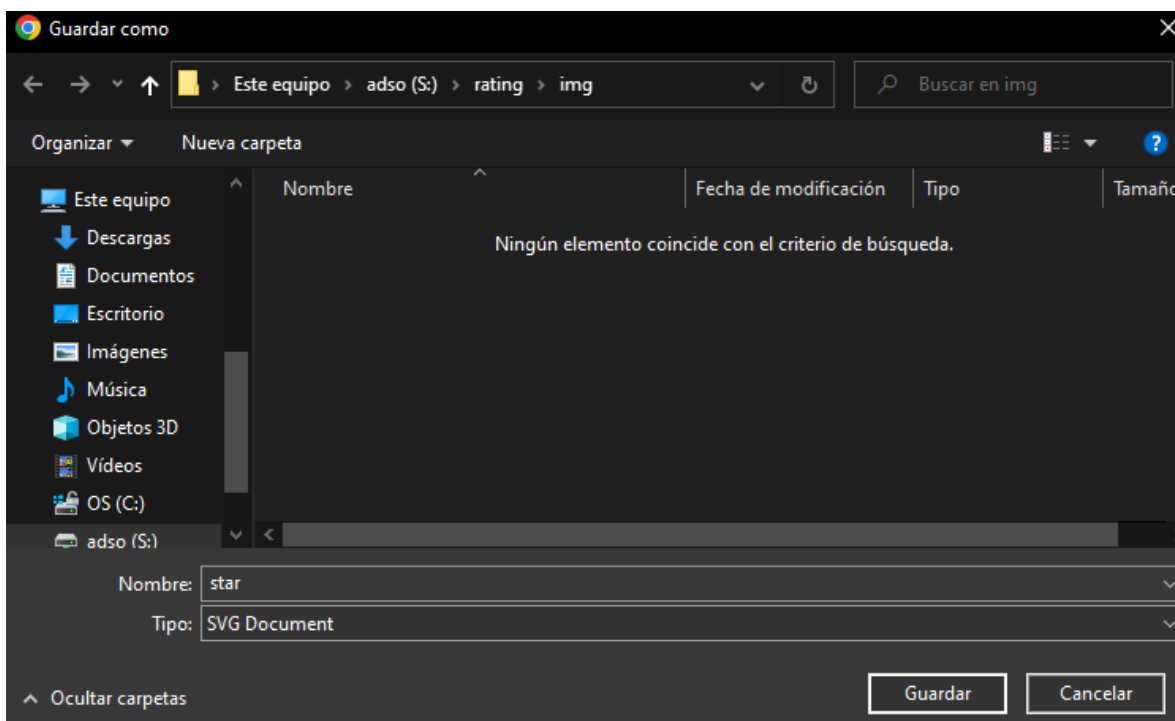
Paso 34: Ahora que escogimos una estrella, la seleccionamos con click izquierdo y a la imagen de la derecha le damos click derecho y seleccionamos “Guardar imagen como...”



Paso 35: Buscaremos la carpeta donde esta nuestro proyecto y seleccionamos la carpeta “img”



Paso 36: Una vez dentro de la carpeta le damos click a guardar luego de poner el nombre.



Paso 37: Ahora iremos a nuestro archivo “style.css” y vamos a añadir nuestras estrellas en la clase “.item” añadiéndolo como una máscara (línea 17), y vamos a colocar 2 puntos antes de la ruta debido a que estamos dentro de la carpeta “css” y toca salir a la carpeta anterior y por eso se usan 2 “..” y por ultimo aplicamos la mascara a la imagen “línea 19”

```
13  /* Define el estilo de cada elemento individual de calificación */
14  .item {
15      background-color: ■ rgb(198, 198, 198); /* Color de fondo gris claro */
16      /* Aplica una máscara a la imagen utilizando propiedades específicas para navegadores WebKit */
17      -webkit-mask-image: url(..../img/star.svg);
18      /* Aplica una máscara a la imagen utilizando la propiedad mask-image */
19      mask-image: url(..../img/star.svg);
20      width: 50px; /* Ancho de 50px */
21      height: 50px; /* Altura de 50px */
22      cursor: pointer; /* Cambia el cursor al puntero para indicar interactividad */
23  }
24
25  /* Define el estilo de un elemento de calificación lleno */
26  .item-full {
27      background-color: ■ rgb(255, 237, 35); /* Cambia el color de fondo a un tono de amarillo */
28  }
```

Paso 38: Ahora miramos como quedo nuestro proyecto terminado (paso 20)

