VueJS

Es un framework de JS para creacion de interfaces de usuario

Construido sobre HTML, CSS, JS, tiene una sintaxis basada en componenetes que te permite desarrollar de forma eficiente interfaces de usuario

No consulta una db, tampoco seguridad, ni css, hay que descargar librerias para poder hacer esto

1) instalar vite npm i vite@latest

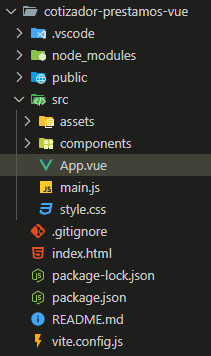
2) moverte al proyecto

3) correr vite npm run dev

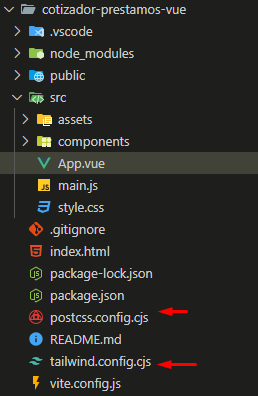
4) Instalar npm i -D tailwindcss@3.1.8 postcss@8.4.18 autoprefixer@10.4.12 para trabajar con tailwin

5) npx tailwindcss init -p para generar el archivo de configuracion de tailwind

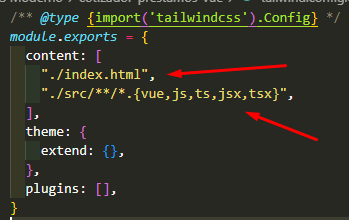
Proyecto de Vue ya eliminando el componente de inicio



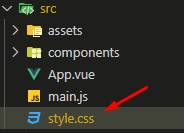
npx tailwindcss init -p para generar los archivos de configuración de tailwind

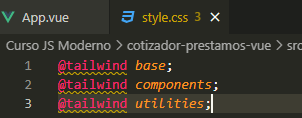


Configuracion necesaria para que tailwnd lo interprete y sepa donde interpretar el código de css, que to lo encuentre en index.html o en el directorio src que tengan la extencion vue,js, ts, jsx, tsx y tengan código de tailwind lo interprete



Agregar las directivas de tailwind





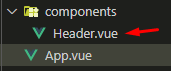
Los componentes de Vue se pueden escribit en 2 API's diferentes Options API y Composition API

Option API tu aplicacion utiliza una sintaxis de tipo Objetos.

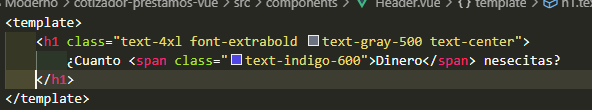
Con Composition API se definen los componentes utilizando imports y escribiendo las funciones directamente en el Componente

Creando un componente con Composition

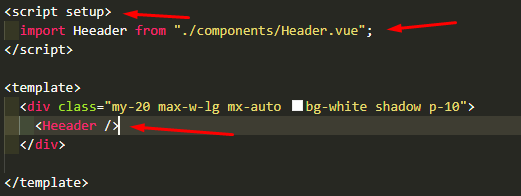
Creamos un componente con la extensio .vue dentro de la carpeta componentes



Codigo del componente

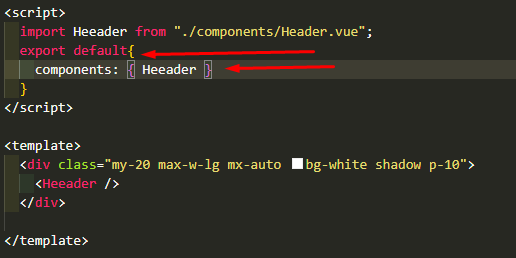


Lo importamos, debe de llevar el setup en el script primera linea y lo utilizamos

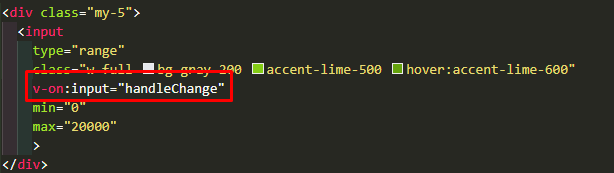


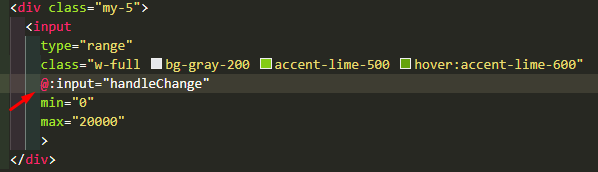
Option solo es la forma en como se llama

Le quitamos el setup, creamos un export default y dentro tiene el components que es un objeto y dentro pasamos el componente

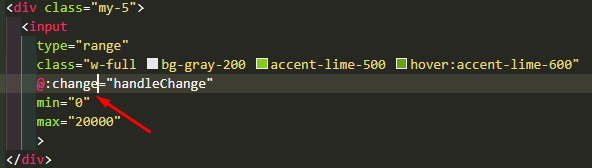


Generar eventos como change, submit, donde v-on:input es el evento y handleChange es la función que reacciona a dicho evento, hay otra forma con una @

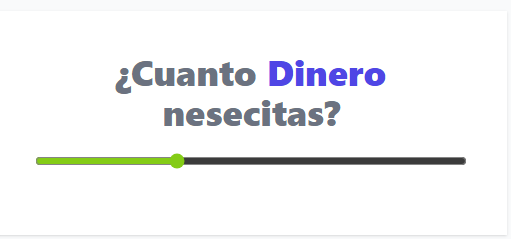




Change también funciona pero aquí en vue parece hasta que arrastramos y soltamos la barra, pero el input cualquier movimiento lo registra



La barra de desplazamiento es la capturamos el evento cuando se arrastra

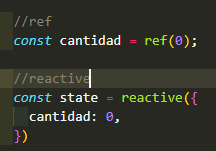


Administrar el State

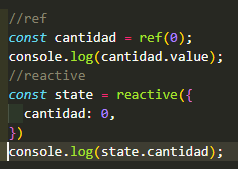
Las dos formas mas sencillas de manejar un state en Vue son con ref y reactive, ambos se importan desde vue

Ref toma valores primitivos mientras que reactive toma objetos

En un equipo de trabajo o un proyecto grande se recomienda adminsitrar el state con una herramienta llamada Pinia(sustituto de VueX)

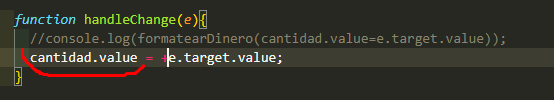


Accediendo a su valor

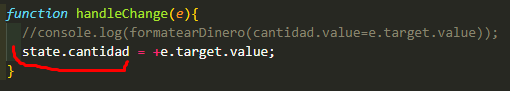


Como modificar estos valores

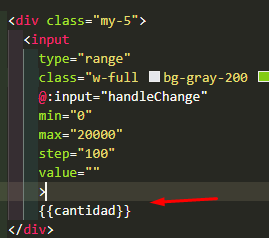
Cambiando el estado de un ref, solo se le asigna el valor del evento



Cambiando el state con reactive



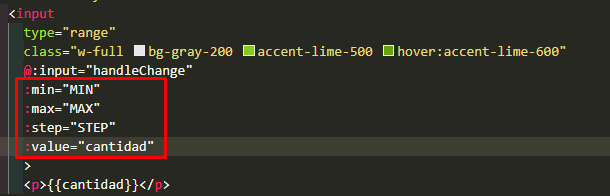
Mostrar variables en el html



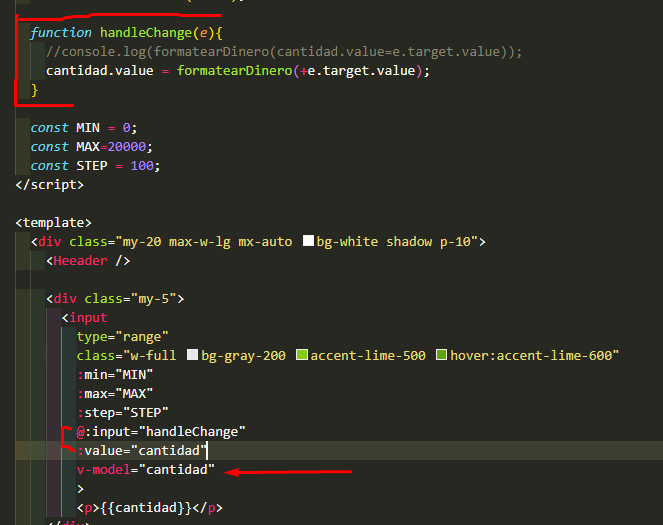
Ahí se muestran las dos formas pero me gusta la primera



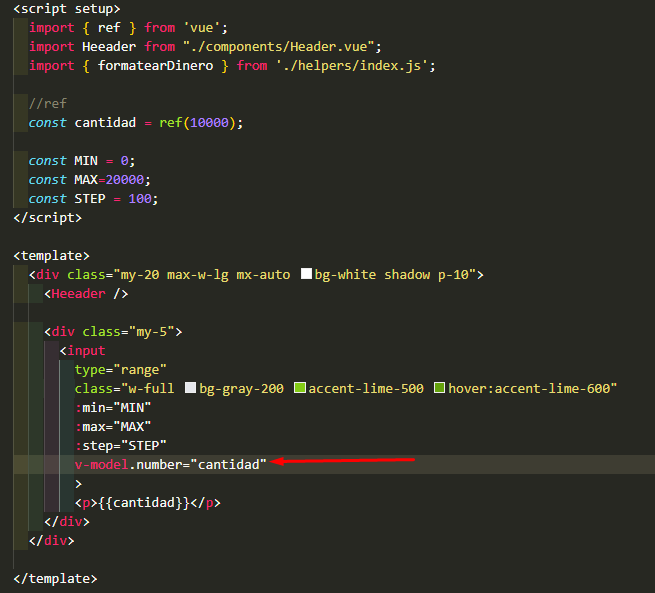
Atributos dinámicos, se le anteponen los dos punto MIN, MAX, STEP, cantidad son variables



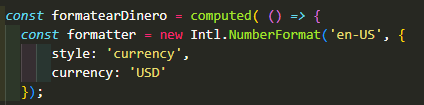
v-model=”cantidad“ simplifica el hanldeChange el value y la función



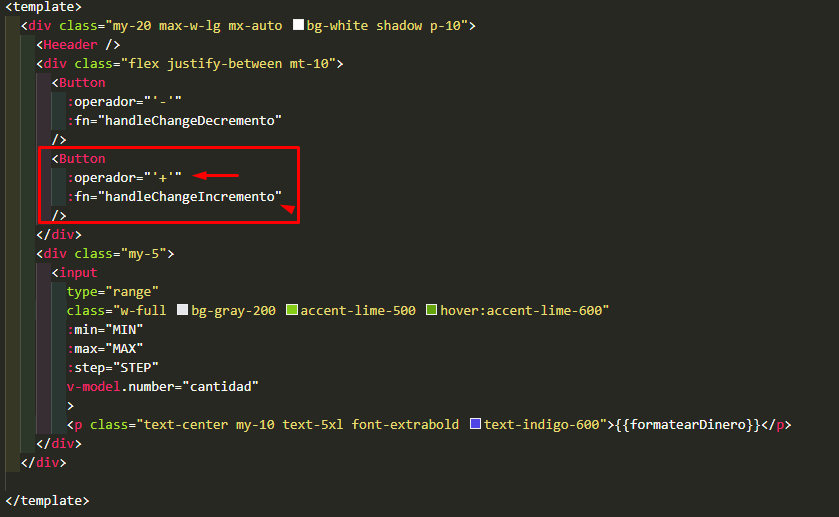
Con una sola línea simplificamos muchas cosas y también lo convierte a numero ya que un valor de un html siempre es string



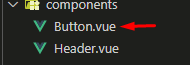
Compute Properties, me gusta mas como helper por que aquí no se le podrían pasar parametros



Paso de parámetros a los Props



Se creo otro componente



Se creo la estructura del script y se creo la estructura del template, en el script se define el prop y este e sun objeto que se le pasa un tipo de dato. En el template se le pasan los parametros

