



Plantillas y rendering en Vue

Estilos condicionales

Utilizar la sintaxis de templates de Vuex para el despliegue de valores y variables que den solución a un requerimiento.

- Unidad 1: Introducción a Componentes Web y Vue Js
- Unidad 2: Binding de formularios
- Unidad 3: Templates y rendering en Vue
- Unidad 4: Manejo de eventos y reutilización de componentes
- Unidad 5: Consumo de datos desde una API REST



Te encuentras aquí



¿Qué aprenderás en esta sesión?

- *Aplica enlace de estilos a un componente para resolver un problema planteado.*

¿Cómo crearías una interfaz de usuario que permita modificar los estilos de un elemento?



¿Cuál es la diferencia
entre v-if y v-show?



/* Binding de estilos */

Binding de estilos

:style

Cómo ya hemos visto, podemos enlazar los atributos de las etiquetas HTML con valores del estado como bien lo hicimos con imágenes y la propiedad **:src**.

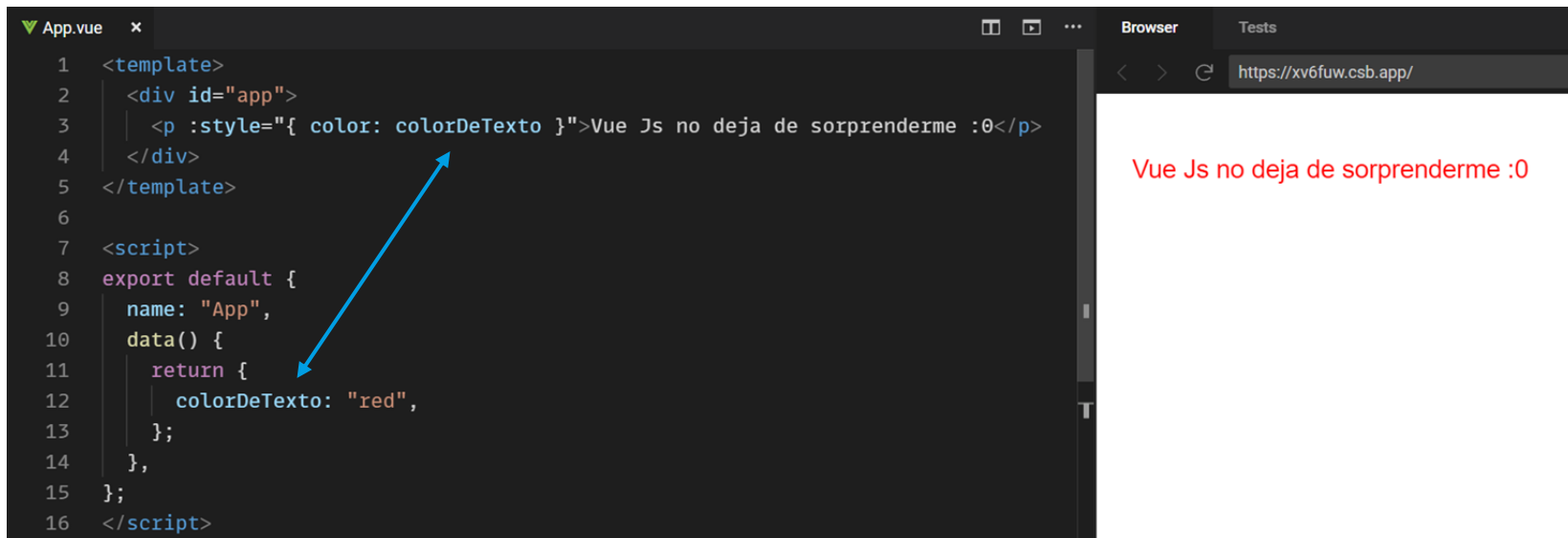
Al atributo **style** podemos asignarle un objeto con propiedades CSS cuyos valores sean variables del estado:

```
<p :style="{ <propiedad css>: <variable del estado> }">
```

Binding de estilos

:style

Veamos el siguiente ejemplo donde le asignamos un color de texto a un párrafo:



The screenshot shows a code editor with a file named 'App.vue'. The code is as follows:

```
1 <template>
2   <div id="app">
3     <p :style="{ color: colorDeTexto }">Vue Js no deja de sorprenderme :0</p>
4   </div>
5 </template>
6
7 <script>
8   export default {
9     name: "App",
10    data() {
11      return {
12        colorDeTexto: "red",
13      };
14    },
15  };
16 </script>
```

A blue arrow points from the 'colorDeTexto' property in the data() function (line 12) to the 'color: colorDeTexto' binding in the template (line 3).

To the right of the code editor is a browser window showing the rendered output: 'Vue Js no deja de sorprenderme :0' in red text. The browser's address bar shows 'https://xv6fuw.csb.app/'.

Ejercicio guiado 1



Ejercicio guiado

Agrega una imagen cuyo ancho sea un “100%” definido en el estado

Este ejercicio es similar a cambiar el color de un párrafo. Entonces, debemos realizar lo siguiente:

1. Agregar una imagen con un src (puede ser una URL online) e incluir el style binding con el atributo width que contenga como valor una variable del estado ancho.
1. Agregar una variable en el estado ancho igual a “100%”.



Ejercicio propuesto



Ejercicio propuesto

¡Sigamos practicando!

De acuerdo al ejercicio guiado 1:

- Agrega una imagen cuyo ancho sea un “80%” definido en el estado



Binding de estilos

:style

Ya que podemos asignar un objeto como valor al atributo style, éste puede tener todas las propiedades que necesitamos.

```
1 <template>
2   <div id="app">
3     
11  </div>
12 </template>
13
14 <script>
15 export default {
16   name: "App",
17   data() {
18     return {
19       url: "https://cdn.pixabay.com/photo/2013/10/02/23/03/mountains-190055_960_720.jpg",
20       sombra: "5px 5px 10px 5px black",
21       bordeRedondeado: "50%",
22       alto: "300px",
23     };
24   },
25 };
26
```



Binding de estilos

:style

Cuando tengamos que definir una propiedad CSS cuyo nombre está separado por un guión (-), podemos encerrarla entre comillas o usar la nomenclatura camelCase.

camelCase es un tipo de nomenclatura que podemos ocupar para unir palabras iniciando la primera en minúscula y a partir de la segunda palabra, todas deben iniciar con una mayúscula.

Por ejemplo:

- 'font-size' === fontSize
- 'letter-spacing' === letterSpacing
- 'z-index' === zIndex

{desafío}
latam_

Ejercicio guiado 2



Ejercicio guiado

Crea un formulario con 3 inputs que le asignen el: color de texto, color de fondo y bordes a un párrafo

1. Agregar 3 inputs enlazados a 3 variables del estado diferentes: color, fondo, borde
1. Crear 3 variables en el estado diferentes: color, fondo, borde
1. Agregar un párrafo en el template con un style binding que asigne las 3 propiedades definidas en cada input



Ejercicio guiado

*Crea un formulario con 3 inputs que le asignen el:
color de texto, color de fondo y bordes a un párrafo*

¿Cómo nos fue?

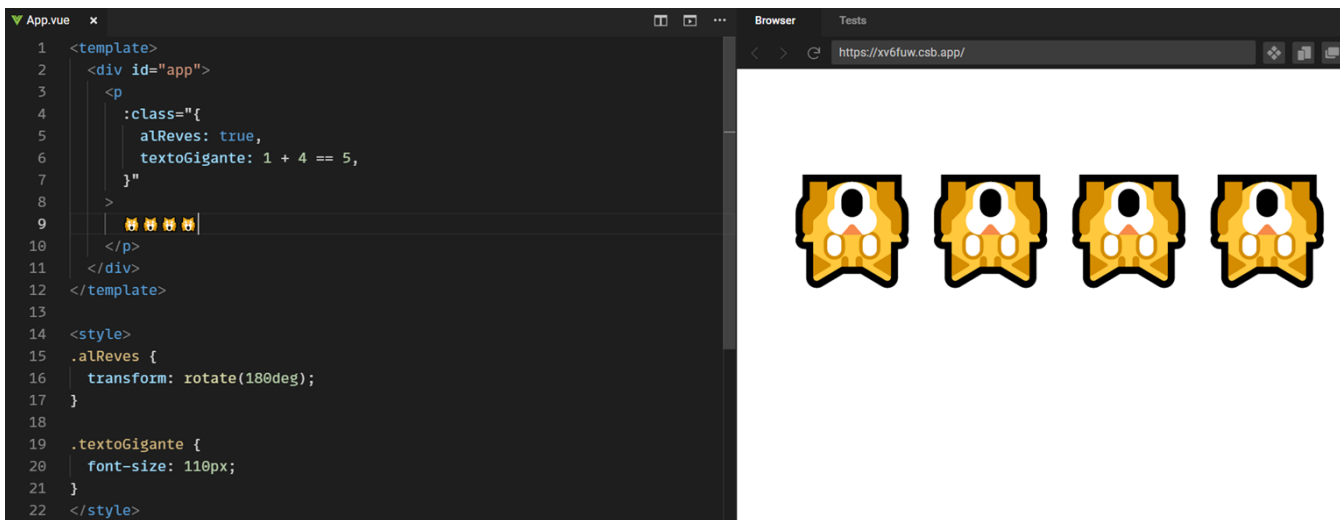
Compartamos nuestro desarrollo y comentemos



Binding de estilos

:class

En el caso del atributo class, podemos condicionar la asignación de clases a partir de expresiones o valores booleanos.



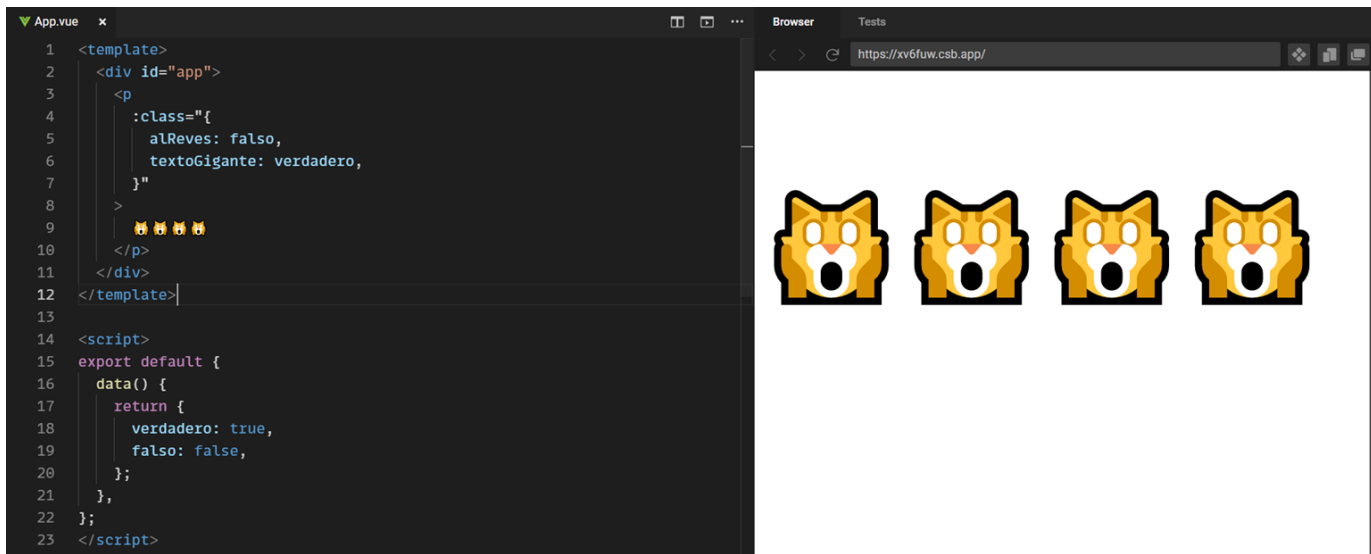
```
1 <template>
2   <div id="app">
3     <p
4       :class="{
5         alReves: true,
6         textoGigante: 1 + 4 == 5,
7       }"
8     >
9     🐱🐱🐱🐱
10  </p>
11 </div>
12 </template>
13
14 <style>
15   .alReves {
16     transform: rotate(180deg);
17   }
18
19   .textoGigante {
20     font-size: 110px;
21   }
22 </style>
```

The browser window shows the rendered output of the Vue.js application. It displays four cat emojis (🐱) in a row. The first two cat emojis are rotated 180 degrees, and the last two are in a larger font size, demonstrating the conditional class binding in the template.

Binding de estilos

:class

Y por supuesto también podemos asignar variables del estado como valores



{desafío}
latam_

Ejercicio guiado 3



Ejercicio guiado

Crea un formulario con 5 checkbox que asignen diferentes clases a un título HTML

1. Agregar 5 inputs de tipo checkbox enlazados a 5 variables del estado.
1. Crear 5 clases que asignen estilos diferentes a un elemento de texto.
1. Crear 5 variables en el estado relacionadas a las clases CSS creadas.
1. Usar el Class Binding para según el estado de los checkbox, las diferentes clases de CSS creadas se asignen o no al título HTML.



Ejercicio guiado

Crea un formulario con 5 checkbox que asignen diferentes clases a un título HTML

¿Cómo nos fue?

Compartamos nuestro desarrollo y comentemos



La experiencia de usuario
mejora notoriamente
cuando el usuario
puede ver inmediatamente
los cambios que produce.





Próxima sesión...

- *Guía de ejercicios.*

{desafío}
latam_

*Academia de
talentos digitales*

