

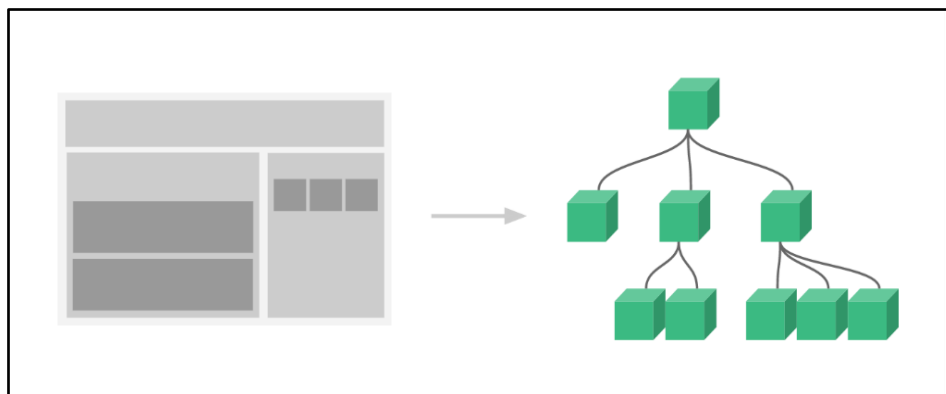
TEXT CLASS REVIEW

TEMAS A TRATAR EN EL CUE:

- ¿Qué son los componentes?
- ¿Qué son las props?
- ¿Como invocar un component con Props?
- Eventos.

¿QUÉ SON LOS COMPONENTES?

Son instancias reutilizables de Vue con un nombre. Podemos usarlos como elementos HTML personalizado. Éstos son una de las características más poderosas en Vue, pues permiten encapsular código HTML para ser reutilizable.



¿Por qué utilizar componentes?

Al poder dividir una misma página web en distintos componentes, éstos se hacen más fáciles de mantener y cambiar, ya que son responsables de partes específicas de nuestra aplicación. También, nos ayudan a la reutilización de código, pues si tenemos un componente NavBar, por ejemplo, éste puede ser utilizado en diferentes vistas de nuestra aplicación, sin necesidad de repetirlo.

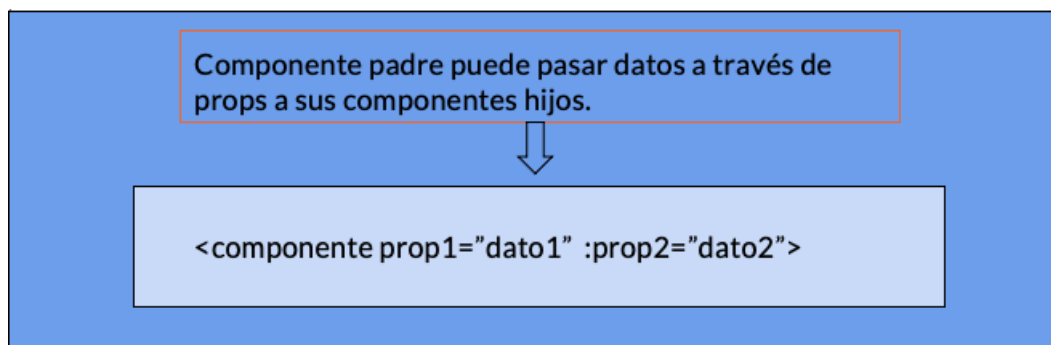
Componente padre e hijos:

Un componente puede tener uno o más componentes dentro de él, lo que nos permite poder anidarlos. Esto es una ventaja, ya que al tener el código dividido en componentes, podemos enfocarnos en la funcionalidad específica de cada uno.

¿QUÉ SON LOS PROPS?

O properties, son propiedades a nuestra medida, que nos permiten enviar datos entre un componente padre, hacia un componente hijo

En esencia, son datos que pueden ser pasados a los componentes, para que éstos realicen cierta acción con ellos.



¿Qué tipos de datos puedo pasar como props?

```
1 props: {  
2   title: String,  
3   likes: Number,  
4   isPublished: Boolean,  
5   commentIds: Array,  
6   author: Object,  
7   callback: Function,  
8   contactsPromise: Promise // or any other constructor  
9 }
```



Formas de declarar props

- **Como Array:** es la forma más simple de declarar una props, con esto, solo le indicamos al componente que necesitamos una como entrada; en caso de requerir más de una, se debe anteponer una coma y el valor.

```
1 props: ['title', 'likes', 'isPublished'],
```

- **Como Objeto:** esto nos permite ingresar el tipo de dato que necesitamos recibir.

```
1 props: {  
2   title: String,  
3   likes: Number,  
4   isPublished: Boolean,  
5 },
```

- **Props como objeto de un objeto:** si quisiéremos ser más específicos, podríamos generar un objeto y poner si el valor es requerido, o si tiene un valor por default:

```
1 props: {  
2   title: {  
3     type: String,  
4     required: true,  
5   },  
6   likes: {  
7     type: Number,  
8     default: 1,  
9   },  
10 },
```

¿CÓMO INVOCAR UN COMPONENTE CON PROPS?

Si llamáramos a nuestro componente lo haríamos de esta forma.

```
1 <div id="app">
2   <MyComponent
3     title="Mi titulo desde una prop"
4     :likes='32'
5     :isPublished="true"
6   ></MyComponent>
7 </div>
```

EVENTOS

Nos sirven para poder comunicarnos con un elemento padre.

Para que un componente hijo pueda comunicarse con el componente padre, es necesario que el éste lo escuche.

Padre puede escuchar al evento gatillado por el hijo , esto lo hace a través de ligar el evento a un metodo dentro de sí mismo.



Elemento hijo puede gatillar un evento , para que el padre ejecute alguna acción

Emitiendo Eventos:

```
eliminar: function(id){  
  this.$emit("eliminar", {id: id});  
},
```



Método dentro del
componente hijo.

- Para emitir un evento, es necesario ocupar la función \$emit de vue.
- \$emit nos permite entregar 2 argumentos. El primero, hace referencia al nombre del evento, y el segundo(opcional) corresponde al (payload), en el cual podemos retornar datos de ser necesario.