ViewJS  
<https://www.vuemastery.com/courses/intro-to-vue-3/intro-to-vue3/>

Tip: Using the console to change reactive properties  
Tip: Adding Methods  
Tip: Adding a computed property  
Tip: Creating components  
Tip: Adding props to our re-usable component  
Tip: Emitting events to child components (i:e button events) – event binding  
Tip: Form and Field binding  
Tip: Alerts  
Tip: Importing components and the scope made available to all components

Tip: Using the console to change reactive properties  
When you browse your app, you can open developer tools and change a property reactively  
Text

Description automatically generated  
Graphical user interface, text, application

Description automatically generated  
This works with any property on our data object for setting properties using the chrome console  
A screenshot of a computer

Description automatically generated with medium confidence  
  
  
Tip: Adding Methods  
Text

Description automatically generated  
  
Text

Description automatically generated

Tip: Adding a computed property

A screenshot of a computer

Description automatically generated with medium confidence  
A screenshot of a computer

Description automatically generated with medium confidence

Tip: Creating components

A screenshot of a computer

Description automatically generated with medium confidence  
  
Text

Description automatically generated

One thing to note, you have to have the /\*html\*/ directive in order to get the markup not think it is a standard string. This was made possible by using the VS Code Extension:  
Text

Description automatically generated

Text

Description automatically generated  
Above we cut and pasted our markup and code from our main.js and pasted it into our component  
Also note:  
In the main.js file, we still need to have the code:

A screenshot of a computer

Description automatically generated with medium confidence  
This is the default method for vue  
  
And now in our Index.html file  
Text

Description automatically generated  
We import the component via script tag and reference the tag like in angular

Tip: Adding props to our re-usable component  
Text

Description automatically generated  
  
Graphical user interface, text

Description automatically generated  
  
Text

Description automatically generated  
  
Text

Description automatically generated  
  
To check

Graphical user interface, text, application, website

Description automatically generated

Tip: Emitting events to child components (i:e button events) – event binding  
In order to fix the button event, we have to create a event binder using emit  
Over in our component:  
Text

Description automatically generated  
  
In our index.html file:  
Text

Description automatically generated  
  
Then in our main.js file for our methods:  
Graphical user interface, text

Description automatically generated  
it's pretty straight forward

Tip: Alerts  
Text

Description automatically generated  
Graphical user interface, application

Description automatically generated

Tip: Form and field binding:  
Two way binding accomplished as shown below  
A screenshot of a computer screen

Description automatically generated with medium confidence

Tip: Importing components and the scope made available to all components

One interesting thing about imports in Vue:  
You import all of your components in the index.html file only

Text

Description automatically generated  
  
Then in any of your components, you can reference them  
Text

Description automatically generated