

Lecture Code Walkthrough - Intro to Vue

Create a new project and walk the students through the process:

```
vue create vue-product-reviews
```

Choose the default project.

Walk the students through the new project that was created.

The **src** folder is where all of the component code will live. Our components will live in the **components** folder. Each component will be put into its own file that ends in a **.vue** extension.

Make a new file there called **ProductReview.vue**. Add HTML to the top:

```
<template>
  <div class="main">
    <h2>Product Reviews for </h2>

    <p class="description"></p>
  </div>
</template>
```

This is the HTML of the component, the display to the user. When the component is included, the **template** tags won't be included but everything inside the **template** tags will be.

The curly brackets signify One-way Data Binding. We will eventually be creating data properties that will hold values that we want to show in those spots and the curly brackets is how we tie the values of those properties to those elements.

Add the following CSS:

```
<style scoped>
div.main {
  margin: 1rem 0;
}
</style>
```

This is the component's default style. This is where you will put anything that works with the component's layout or look.

Then the JavaScript section:

```
<script>
export default {
  name: 'product-review',
  data() {
    return {
```

```
      name: 'Cigar Parties for Dummies',
      description: 'Host and plan the perfect cigar party for all of your
squirrely friends.'
    }
  }
}
</script>
```

This defines our Vue JavaScript Object. Everything after the `export default` is a JavaScript object that holds all the data properties and methods that the component will use.

Here, we've defined a name for the component, `product-review`, and data properties that we data bound to the view.

Warning::`data()` must return an object

One thing to point out to students is that the data method on the Vue object must return another object that holds the data properties. This is because every component needs to have its own data properties. This will be a standard pattern that they will always use for a component's `data`.

That's all you need for a working component.

We can now add that to a page by adding it to the main `App.vue` component. This will be the component's parent component.

```
<template>
  <div id="app">
    <product-review></product-review>
    
    <HelloWorld msg="Welcome to Your Vue.js App"/>
  </div>
</template>

<script>
import HelloWorld from './components/HelloWorld.vue'
import ProductReview from './components/ProductReview.vue'

export default {
  name: 'app',
  components: {
    HelloWorld,
    ProductReview
  }
}
</script>
```

Now, if you run `npm run serve`, you will see your component at the top of the page. It will have the product name in a larger `h2` tag and the description underneath that. Inspect the code and show how the component's HTML has been added to the DOM.