Congratulations on completing this lesson! In this lesson, we covered a number of different techniques for dynamically styling Vue elements.

We learned how to dynamically add inline styles using v-bind:style with a style object or an array of style objects. We then learned how to dynamically add classes using v-bind:class with a class object or an array of class objects and class name strings.

It may seem like any one of these techniques would be sufficient for dynamically styling a front-end app — and that’s true! As you continue learning about Vue, you will see advantages and use cases for each technique.

The important thing to take away at this stage in your learning journey is that you should aim to use the technique that keeps your code the most readable and leaves your app with the least repetitive code.

**Instructions**

As mentioned at the beginning of this lesson, you now possess the skills to style all of the remaining form fields in the app. However, doing this would require separate inline style or class logic for every single field, since you would need to check the value of each field individually when determining what state it is in. This would result in a ton of repetitive code!

For now, if you need to use dynamic styling in your work, this will work. However, later in your learning journey, you will learn how to use your current dynamic styling skills with components in order to re-use your styling logic without copy and pasting your code. This will make your code easier to read and more maintainable.

Congratulations on learning all of these new styling techniques! They may seem overwhelming right now, but after a little practice, you’ll gain the confidence to know what tools you have at your disposal and when to use them. Nice work!