

v-bind to bind multiple attributes

To bind multiple attributes using `v-bind` in Vue.js, you can pass an object where the keys are the attribute names and the values are the corresponding data properties or expressions.

Example:

Consider a scenario where you want to dynamically bind the `id`, `class`, and `title` attributes to an HTML element.

Code

```
<div id="app">
  <button v-bind="buttonAttributes">Click Me</button>
</div>

<script src="https://unpkg.com/vue@3/dist/vue.global.js"></script>
<script>
  const app = Vue.createApp({
    data() {
      return {
        buttonAttributes: {
          id: 'my-button',
          class: 'primary-button',
          title: 'This is a dynamic button',
          'data-custom': 'some-value' // Example of a data attribute
        }
      };
    }
  });

  app.mount('#app');
</script>
```

In this example:

A `div` with the ID `app` serves as the root element for the Vue application.

A `button` element uses `v-bind="buttonAttributes"` to bind multiple attributes.

In the Vue instance's `data` option, an object named `buttonAttributes` is defined.

This `buttonAttributes` object contains key-value pairs where the keys (`id`, `class`, `title`, `data-custom`) represent the attribute names, and the values are the desired values for those attributes.

Vue automatically applies these attributes to the `button` element when the component is mounted.

This approach simplifies binding multiple attributes, especially when dealing with a large number of dynamic attributes or when the attributes are logically grouped together in your data

Revision #2

Created 29 October 2025 02:43:36 by AI API

Updated 1 December 2025 14:01:42 by AI Channel