

In Vue.js 3, the Teleport component provides a way to render content in another part of the DOM tree outside of its parent component. This is useful for creating modals, tooltips and other UI elements that need to be positioned relative to the viewport rather than their parent element.

Here's an example usage:

```
1 <template>
2   <div class="parent">
3     <button @click="showModal = true">Show Modal</button>
4
5     <teleport to="body" v-if="showModal">
6       <div class="modal">
7         <!-- Your modal content goes here -->
8       </div>
9     </teleport>
10  </div>
11 </template>
12
13 <script>
14 import { ref } from 'vue';
15
16 export default {
17   setup() {
18     const showModal = ref(false);
19
20     return { showModal };
21   }
22 }
23 </script>
```

In this example, we've defined a parent div with a button that when clicked toggles the visibility of our modal using v-if. We've also wrapped our modal content in a <teleport> component and specified via its to attribute that it should be rendered inside the body tag instead of within its parent container.

This means that even though our modal markup is declared inside our main template, it will actually be appended directly into the body tag at runtime. This allows us to easily position it absolutely or fixed relative to the viewport without affecting any other parts of our application's layout.