What are the different form states, and why are they important?

In HTML, form controls, like inputs, can be in different stages or conditions like a focused state, readonly state, or disabled state.

The first state would be considered the default state. The default state of an email address input is a blank input. That is what the email input looks like when it is first rendered on the page. At this point, the user has not input any information.

When the user clicks on a form control or selects it with the keyboard's tab key, then that means it is in the focused state. When an input is in the focused state, most browsers will show a blue highlighted border around the input. But you can choose to add additional styles in CSS.

Another form state is the disabled state. This state shows users that an input cannot be focused or activated. To disable an input, you can add the disabled boolean attribute to the element like this:

<input disabled type="email" name="email" id="email" />

If the user tries to click on the input, then the focus will not be enabled. Similar to the focused state, you can choose to add additional styles for the disabled state using CSS.

Another type of form state is the readonly state. This is when a form control, like an input, is not editable by the user. Here is an example of setting an email input to read only:

<input

readonly

type="email"

name="email"

id="email"

value="example@email.com"

/>

The value attribute is used to set the value shown inside the input field. If you tried to click on the input, you would not be able to edit the existing value.

A key difference between the disabled state and readonly state is that readonly can be focused while the disabled state cannot.

Understanding the different form states is important because they ensure a smooth user experience by providing clear feedback and guidance while handling errors.