

Unstyled form control

The `FormControlUnstyled` is a utility that lets you associate a form input with auxillary components, such as labels, error indicators or helper text.

```
{{"component": "modules/components/ComponentLinkHeader.js", "design": false}}
```

FormControlUnstyled

`FormControlUnstyled` wraps an input and other components, enabling to reflect the input's state in these other components.

For instance, you may want to show an additional element asking the user to enter a value if the input is empty, or display a warning icon if the entered value is incorrect.

```
{{"demo": "BasicFormControl.js"}}
```

The `FormControlUnstyled` provides a context that can be read by the `useFormControl` hook.

useFormControl hook

The `useFormControl` hook can be used to enable integration between custom form inputs and `FormControlUnstyled`. Additionally, you can read the form control's state and react to its changes in a custom component.

The demo below shows both. `CustomInput` is a wrapper around a native HTML `input` that adds `FormControlUnstyled` integration. `ControlStateDisplay` reads the state of the form control and displays it as text.

```
{{"demo": "UseFormControl.js", "defaultCodeOpen": false}}
```

Note that even though `FormControlUnstyled` supports both controlled and uncontrolled-style API (i.e. it accepts `value` and `defaultValue` props), `useFormControl` returns only the controlled `value`. This way, you don't have to implement both in your custom input - `FormControlUnstyled` does this for you.

`useFormControl` returns an object with the following fields:

Name	Type	Description
<code>disabled</code>	boolean	Represents the value of the <code>FormControlUnstyled</code> 's <code>disabled</code> prop.
<code>error</code>	boolean	Represents the value of the <code>FormControlUnstyled</code> 's <code>error</code> prop. Note that it is not calculated automatically (i.e. it's not set when <code>required: true</code> and <code>value: ''</code>)
<code>filled</code>	boolean	Set to <code>true</code> if <code>value</code> is not empty.
<code>focused</code>	boolean	Set to <code>true</code> if the wrapped input has received focus.
<code>required</code>	boolean	Represents the value of the <code>FormControlUnstyled</code> 's <code>required</code> prop.
<code>value</code>	unknown	The current value of the form control.

Additionally, the following callbacks are a part of the returned object. They are meant to be used when creating custom inputs.

Name	Type	Description
------	------	-------------

onChange	React.ChangeEvent => void	Value change handler. Should be forwarded to the inner input.
onBlur	() => void	Focus change handler. Should be forwarded to the inner input.
onFocus	() => void	Focus change handler. Should be forwarded to the inner input.