

Checkbox

Checkboxes allow the user to select one or more items from a set.

Checkboxes can be used to turn an option on or off.

If you have multiple options appearing in a list, you can preserve space by using checkboxes instead of on/off switches. If you have a single option, avoid using a checkbox and use an on/off switch instead.

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

Basic checkboxes

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

Label

You can provide a label to the `Checkbox` thanks to the `FormControlLabel` component.

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

Size

Use the `size` prop or customize the font size of the svg icons to change the size of the checkboxes.

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

Color

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

Icon

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

Controlled


You can control the checkbox with the `checked` and `onChange` props:

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

Indeterminate

A checkbox input can only have two states in a form: checked or unchecked. It either submits its value or doesn't. Visually, there are **three** states a checkbox can be in: checked, unchecked, or indeterminate.

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

 When `indeterminate` is set, the value of the `checked` prop only impacts the form submitted values. It has no accessibility or UX implications.

FormGroup

`FormGroup` is a helpful wrapper used to group selection control components.

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

Label placement

You can change the placement of the label:

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

Customization

Here is an example of customizing the component. You can learn more about this in the [overrides documentation page](#).

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

🔗 If you are looking for inspiration, you can check [MUI Treasury's customization examples](#).

When to use

- [Checkboxes vs. Radio Buttons](#)
- [Checkboxes vs. Switches](#)

Accessibility

(WAI-ARIA: <https://www.w3.org/TR/wai-aria-practices/#checkbox>)

- All form controls should have labels, and this includes radio buttons, checkboxes, and switches. In most cases, this is done by using the `<label>` element ([FormControlLabel](#)).
- When a label can't be used, it's necessary to add an attribute directly to the input component. In this case, you can apply the additional attribute (e.g. `aria-label`, `aria-labelledby`, `title`) via the `inputProps` prop.

```
<Checkbox
  value="checkedA"
  inputProps={{
    'aria-label': 'Checkbox A',
  }}
/>
```