

# Componente React para Botões de opção

## Botões de opção

Botões de opção permitem o usuário selecionar uma das opções de um conjunto.

Use botões de opção quando um usuário precisar ver todas as opções disponíveis. Se várias opções podem ser suprimidas da visualização, considere usar um menu suspenso (dropdown), pois ele utiliza menos espaço.

Os botões de opção devem ter a opção comum e mais usada selecionada por padrão.

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

## Radio group

O componente `RadioGroup` é um wrapper utilizado para agrupar componentes `Radio`, fornece uma API mais simples e adequada para a acessibilidade por teclado.

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

## Direção

To lay out the buttons horizontally, set the `row` prop:

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

## Controlled

You can control the radio with the `value` and `onChange` props:

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

## Botões de opção independentes

O componente `Radio` também pode ser usado de forma independente, sem o wrapper `RadioGroup`.

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

## Tamanho

Use the **size** prop or customize the font size of the svg icons to change the size of the radios.

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

## Cor

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

## Posicionamento do rótulo

Você pode alterar o posicionamento do rótulo com o componente `FormControlLabel` na propriedade `labelPlacement`:

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

## Exibição de erro

Em geral, os botões de opção devem ter um valor selecionado por padrão. Se esse não for o caso, você pode exibir um erro se nenhum valor estiver selecionado quando o formulário for submetido:

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

## Opções customizadas

Aqui está um exemplo de customização do componente. Você pode aprender mais sobre isso na página de documentação de sobrecargas.

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

## useRadioGroup

Para situações de uso avançadas de customização, um hook `useRadioGroup()` é exposto. Ele retorna o valor do contexto do grupo de botões de opção pai. O componente de botões de opção usa este hook internamente.

## API

```
import { useRadioGroup } from '@material-ui/core/RadioGroup';
```

**Retornos** `value` (*object*):

- `value.name` (*string* [opcional]): O nome usado para fazer referência ao valor do controle.
- `value.onChange` (*func* [opcional]): Callback fired when a radio button is selected.
- `value.value` (*any* [opcional]): Valor do botão de opção selecionado.

**Exemplo** `{{“demo”: “UseRadioGroup.js”}}`

## Quando usar

- Caixas de Seleção vs. Botões de Opção

## Acessibilidade

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

- Todos os controles de formulário devem ter rótulos, e isso inclui os botões de opção, caixas de seleção e interruptores. Na maioria dos casos, isso é feito usando o elemento `<label>` (`FormControlLabel`).
- Quando um rótulo não pode ser usado, é necessário adicionar um atributo diretamente no componente de entrada. Nesse caso você pode aplicar um atributo adicional (por exemplo, `aria-label`, `aria-labelledby`, `title`) através da propriedade `inputProps`.

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