

Menu

Os menus exibem uma lista de opções em superfícies temporárias.

O menu exibe uma lista de opções em uma superfície temporária. Aparece quando o usuário interage com um botão ou outro controle.

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

Menu básico

Um menu básico abre sobre o elemento âncora por padrão (esta opção pode ser [alterada](#) através de propriedades). Quando estão perto de uma borda da tela, menus básicos realinham verticalmente para garantir que todos os itens do menu fiquem completamente visíveis.

Escolhendo uma opção deve confirmar imediatamente a opção e fechar o menu.

Desambiguação: Em contraste com menus simples, uma caixa de diálogo simples pode apresentar detalhes adicionais relacionados às opções disponíveis para um item da lista ou fornecer navegação ou ações indiretas relacionada à tarefa principal. Although they can display the same content, simple menus are preferred over simple dialogs because simple menus are less disruptive to the user's current context.

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

Menu selecionado

No viewport do desktop, o preenchimento é aumentado para dar mais espaço ao menu.

```
{{"demo": "IconMenu.js", "bg": true}}
```

Menu posicionado

For the menu that has long list and long text, you can use the `dense` prop to reduce the padding and text size.

```
{{"demo": "DenseMenu.js", "bg": true}}
```

Composição de MenuList

Se usado para seleção de itens, quando abertos, menus simples colocam o foco inicial no item do menu selecionado.

O item de menu atualmente selecionado é definido usando a propriedade `selected` (de [ListItem](#)). Para usar um item do menu selecionado sem impactar o foco inicial, defina a propriedade `variante` para "menu".

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

Menu customizado

Devido ao componente `Menu` usar o componente `Popover` para se posicionar, você pode usar as mesmas [propriedades de posicionamento](#) para posicioná-lo. Por exemplo, você pode exibir o menu abaixo da âncora:

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

Composição de MenuList

O componente `Menu` usa o componente `Popover` internamente. No entanto, você pode querer usar uma estratégia de posicionamento diferente ou não bloquear a rolagem. Para atender a essas situações, disponibilizamos um componente `MenuList` que você pode compor com o uso do `Popper`, veja o exemplo a seguir.

A principal responsabilidade do componente `MenuList` é manipular o foco.

```
{{"demo": "MenuListComposition.js", "bg": true}}
```

Limitações

Se a altura de um menu impede que todos os itens de menu sejam exibidos, o menu terá internamente a opção de rolagem.

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

Trocar transição

Aqui está um exemplo de customização do componente. Você pode aprender mais sobre isso na [página de documentação de sobrescritas](#).

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

O `MenuItem` é um encapsulador em torno de `Listitem` com alguns estilos adicionais. Você pode usar os mesmos recursos de composição de lista com o componente `MenuItem`:

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

Menu de contexto

Se a altura de um menu impede que todos os itens de menu sejam exibidos, o menu terá internamente a opção de rolagem.

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

Projetos Complementares

Existe [um problema com flexbox](#) que impede `text-overflow: ellipsis` de funcionar em um leiaute flexível. Você pode usar o componente `Typography` com `noWrap` para solucionar esse problema:

```
{{"demo": "TypographyMenu.js", "bg": true}}
```

Trocar transição

Use uma transição diferente.

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

Menu de contexto

Aqui está um exemplo de um menu de contexto. (Clique com o botão direito para abrir.)

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

Complementary projects

Para situações de uso mais avançadas, você pode tirar proveito com:

PopupState helper

Existe um pacote de terceiros [material-ui-popup-state](#) que cuida do estado do menu para você na maioria das situações.

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