Lists

Lists are continuous, vertical indexes of text or images.

Lists are a continuous group of text or images. They are composed of items containing primary and supplemental actions, which are represented by icons and text.

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

Basic List

{{"demo": "BasicList.js", "bg": true}}

The last item of the previous demo shows how you can render a link:

```
<ListItemButton component="a" href="#simple-list">
  <ListItemText primary="Spam" />
</ListItemButton>
```

You can find a demo with React Router following this section of the documentation.

Nested List

{{"demo": "NestedList.js", "bg": true}}

Folder List

{{"demo": "FolderList.js", "bg": true}}

Interactive

Below is an interactive demo that lets you explore the visual results of the different settings:

{{"demo": "InteractiveList.js", "bg": true}}

Selected ListItem

{{"demo": "SelectedListItem.js", "bg": true}}

Align list items

When displaying three lines or more, the avatar is not aligned at the top. You should set the `alignItems="flex-start"` prop to align the avatar at the top, following the Material Design guidelines:

{{"demo": "AlignItemsList.js", "bg": true}}

List Controls

Checkbox

A checkbox can either be a primary action or a secondary action.

The checkbox is the primary action and the state indicator for the list item. The comment button is a secondary action and a separate target.

{{"demo": "CheckboxList.js", "bg": true}}

The checkbox is the secondary action for the list item and a separate target.

{{"demo": "CheckboxListSecondary.js", "bg": true}}

Switch

The switch is the secondary action and a separate target.

{{"demo": "SwitchListSecondary.js", "bg": true}}

Sticky subheader

Upon scrolling, subheaders remain pinned to the top of the screen until pushed off screen by the next subheader. This feature relies on CSS sticky positioning. (⚠️ no IE 11 support)

{{"demo": "PinnedSubheaderList.js", "bg": true}}

Inset List Item

The `inset` prop enables a list item that does not have a leading icon or avatar to align correctly with items that do.

{{"demo": "InsetList.js", "bg": true}}

Gutterless list

When rendering a list within a component that defines its own gutters, `ListItem` gutters can be disabled with `disableGutters`.

{{"demo": "GutterlessList.js", "bg": true}}

Virtualized List

In the following example, we demonstrate how to use react-window with the `List` component. It renders 200 rows and can easily handle more. Virtualization helps with performance issues.

{{"demo": "VirtualizedList.js", "bg": true}}

The use of react-window when possible is encouraged. If this library doesn't cover your use case, you should consider using react-virtualized, then alternatives like react-virtuoso.

Customization

Here are some examples of customizing the component. You can learn more about this in the overrides documentation page.

{{"demo": "CustomizedList.js"}}

🎨 If you are looking for inspiration, you can check MUI Treasury's customization examples.