



Search...

Ctrl+K



Typography

Use typography to present your design and content as clearly and efficiently as possible.



Check out the latest remote job listings from the leading job board for designers, developers, and creative pros.

ads via Carbon

[View as Markdown](#)[Feedback](#)[Bundle size](#)[Source](#)[Material Design](#)[Figma](#)[Sketch](#)

Roboto font

Material UI uses the [Roboto](#) font by default. Add it to your project via Fontsource, or with the Google Fonts CDN.

[npm](#) [pnpm](#) [yarn](#)

```
npm install @fontsource/roboto
```

[Copy](#)

Then you can import it in your entry point like this:

```
import '@fontsource/roboto/300.css';
import '@fontsource/roboto/400.css';
import '@fontsource/roboto/500.css';
import '@fontsource/roboto/700.css';
```

[Copy](#)

- Fontsource can be configured to load specific subsets, weights, and styles. Material UI's default typography configuration relies only on the 300, 400, 500, and 700 font weights.

Google Web Fonts

To install Roboto through the Google Web Fonts CDN, add the following code inside your project's `<head>` tag:

```
<link rel="preconnect" href="https://fonts.googleapis.com" />
<link rel="preconnect" href="https://fonts.gstatic.com" crossorigin />
<link
```

[Copy](#)

```
rel="stylesheet"
href="https://fonts.googleapis.com/css2?family=Roboto:wght@300;400;500;700&display=swap"
/>
```

Component

Usage

The Typography component follows the [Material Design typographic scale](#) that provides a limited set of type sizes that work well together for a consistent layout.

h1. Heading

h2. Heading

h3. Heading

h4. Heading

h5. Heading

h6. Heading

subtitle1. Lorem ipsum dolor sit amet, consectetur adipisicing elit.

Quos blanditiis tenetur

subtitle2. Lorem ipsum dolor sit amet, consectetur adipisicing elit. Quos
blanditiis tenetur

body1. Lorem ipsum dolor sit amet, consectetur adipisicing elit.

Quos blanditiis tenetur unde suscipit, quam beatae rerum inventore
consectetur, neque doloribus, cupiditate numquam dignissimos
laborum fugiat deleniti? Eum quasi quidem quibusdam.

body2. Lorem ipsum dolor sit amet, consectetur adipisicing elit. Quos
blanditiis tenetur unde suscipit, quam beatae rerum inventore consectetur,
neque doloribus, cupiditate numquam dignissimos laborum fugiat deleniti?
Eum quasi quidem quibusdam.

BUTTON TEXT

caption text

OVERLINE TEXT

 Edit in Chat

JS

TS

[Hide code](#)



```
import Box from '@mui/material/Box';
import Typography from '@mui/material/Typography';

export default function Types() {
  return (
    <Box sx={{ width: '100%', maxWidth: 500 }}>
      <Typography variant="h1" gutterBottom>
        h1. Heading
      </Typography>
    </Box>
  )
}
```

```
<Typography variant="h2" gutterBottom>
  h2. Heading
</Typography>
<Typography variant="h3" gutterBottom>
  h3. Heading
</Typography>
<Typography variant="h4" gutterBottom>
  h4. Heading
</Typography>
<Typography variant="h5" gutterBottom>
  h5. Heading
</Typography>
<Typography variant="h6" gutterBottom>
  h6. Heading
</Typography>
<Typography variant="subtitle1" gutterBottom>
  subtitle1. Lorem ipsum dolor sit amet, consectetur adipisicing elit. Quos
  blanditiis tenetur
</Typography>
<Typography variant="subtitle2" gutterBottom>
  subtitle2. Lorem ipsum dolor sit amet, consectetur adipisicing elit. Quos
  blanditiis tenetur
</Typography>
<Typography variant="body1" gutterBottom>
  body1. Lorem ipsum dolor sit amet, consectetur adipisicing elit. Quos
  blanditiis tenetur unde suscipit, quam beatae rerum inventore consectetur,
  neque doloribus, cupiditate numquam dignissimos laborum fugiat deleniti? Eum
  quasi quidem quibusdam.
</Typography>
```

Theme keys

In some situations you might not be able to use the Typography component. Hopefully, you might be able to take advantage of the [typography](#) keys of the theme.

THIS DIV'S TEXT LOOKS LIKE THAT OF A BUTTON.

 [Edit in Chat](#) JS TS

[Collapse code](#)



```
import { styled } from '@mui/material/styles';

const Div = styled('div')(({ theme }) => ({
  ...theme.typography.button,
  backgroundColor: (theme.vars || theme).palette.background.paper,
  padding: theme.spacing(1),
}));

export default function TypographyTheme() {
```

```
return <Div>{"This div's text looks like that of a button."}</Div>;  
}
```

Customization

Adding & disabling variants

In addition to using the default typography variants, you can add custom ones, or disable any you don't need. See the [Adding & disabling variants](#) page for more info.

Changing the semantic element

The Typography component uses the `variantMapping` prop to associate a UI variant with a semantic element. It's important to realize that the style of a typography component is independent from the semantic underlying element.

To change the underlying element for a one-off situation, like avoiding two `h1` elements in your page, use the `component` prop:

```
<Typography variant="h1" component="h2">  
  h1. Heading  
</Typography>
```

[Copy](#)

To change the typography element mapping globally, [use the theme](#):

```
const theme = createTheme({  
  components: {  
    MuiTypography: {  
      defaultProps: {  
        variantMapping: {  
          h1: 'h2',  
          h2: 'h2',  
          h3: 'h2',  
          h4: 'h2',  
          h5: 'h2',  
          h6: 'h2',  
          subtitle1: 'h2',  
          subtitle2: 'h2',  
          body1: 'span',  
          body2: 'span',  
        },  
      },  
    },  
  },  
};
```

[Copy](#)

System props



- ⓘ System props are deprecated and will be removed in the next major release. Please use the `sx` prop instead.

```
- <Typography mt={2} />  
+ <Typography sx={{ mt: 2 }} />
```

Copy

Accessibility



Key factors to follow for an accessible typography:

- **Color.** Provide enough contrast between text and its background, check out the minimum recommended [WCAG 2.0 color contrast ratio ↗](#) (4.5:1).
- **Font size.** Use [relative units \(rem\)](#), instead of pixels, to accommodate the user's browser settings.
- **Heading hierarchy.** Based on [the W3 guidelines ↗](#), don't skip heading levels. Make sure to [separate the semantics from the style](#).

API



See the documentation below for a complete reference to all of the props and classes available to the components mentioned here.

- [`<Typography />`](#)

Edit this page

Was this page helpful?

[`< Tooltip`](#)

[`> Alert`](#)