

Divider

The Divider component provides a thin, unobtrusive line for grouping elements to reinforce visual hierarchy.

>>

🏠

🔗

✍️

💬

📄

T

✉️

📏

Design and Development tips in your inbox. Every weekday.
ads via Carbon

- View as Markdown
- Feedback
- Bundle size
- Source
- Material Design
- Figma
- Sketch

Introduction

The Material UI Divider component renders as a dark gray `<hr>` by default, and features several useful props for quick style adjustments.

Toothbrush\$4.50

Pinstriped cornflower blue cotton blouse takes you on a walk to the park or just down the hall.

Select type

SoftMediumHard

Edit in ChatJSTS

Hide code⚡📦📄🔍🔄⋮

```
import Card from '@mui/material/Card';
import Box from '@mui/material/Box';
import Chip from '@mui/material/Chip';
import Stack from '@mui/material/Stack';
import Divider from '@mui/material/Divider';
import Typography from '@mui/material/Typography';

export default function IntroDivider() {
  return (
    <Card variant="outlined" sx={{ maxWidth: 360 }}>
      <Box sx={{ p: 2 }}>
```

```

<Stack
  direction="row"
  sx={{ justifyContent: 'space-between', alignItems: 'center' }}
>
  <Typography gutterBottom variant="h5" component="div">
    Toothbrush
  </Typography>
  <Typography gutterBottom variant="h6" component="div">
    $4.50
  </Typography>
</Stack>
<Typography variant="body2" sx={{ color: 'text.secondary' }}>
  Pinstriped cornflower blue cotton blouse takes you on a walk to the park or
  just down the hall.
</Typography>
</Box>
<Divider />
<Box sx={{ p: 2 }}>
  <Typography gutterBottom variant="body2">
    Select type
  </Typography>
  <Stack direction="row" spacing={1}>
    <Chip color="primary" label="Soft" size="small" />
    <Chip label="Medium" size="small" />
    <Chip label="Hard" size="small" />
  </Stack>
</Box>

```

Basics



```
import Divider from '@mui/material/Divider';
```

Copy

Variants



The Divider component supports three variants: `fullWidth` (default), `inset`, and `middle`.

Full width variant below

Inset variant below

Middle variant below

List item

Edit in Chat

JS

TS

Hide code



```
import List from '@mui/material/List';
import ListItem from '@mui/material/ListItem';
import ListItemText from '@mui/material/ListItemText';
import Divider from '@mui/material/Divider';

const style = {
  py: 0,
  width: '100%',
  maxWidth: 360,
  borderRadius: 2,
  border: '1px solid',
  borderColor: 'divider',
  backgroundColor: 'background.paper',
};

export default function DividerVariants() {
  return (
    <List sx={style}>
      <ListItem>
        <ListItemText primary="Full width variant below" />
      </ListItem>
      <Divider component="li" />
      <ListItem>
        <ListItemText primary="Inset variant below" />
      </ListItem>
      <Divider variant="inset" component="li" />
      <ListItem>
        <ListItemText primary="Middle variant below" />
      </ListItem>
      <Divider variant="middle" component="li" />
      <ListItem>
        <ListItemText primary="List item" />
      </ListItem>
    </List>
  );
}
```

Orientation



Use the `orientation` prop to change the Divider from horizontal to vertical. When using vertical orientation, the Divider renders a `<div>` with the corresponding accessibility attributes instead of `<hr>` to adhere to the WAI-ARIA [spec](#).


[Edit in Chat](#)

JS

TS

[Collapse code](#)


```
import FormatAlignLeftIcon from '@mui/icons-material/FormatAlignLeft';
import FormatAlignCenterIcon from '@mui/icons-material/FormatAlignCenter';
import FormatAlignRightIcon from '@mui/icons-material/FormatAlignRight';
import FormatBoldIcon from '@mui/icons-material/FormatBold';
import Box from '@mui/material/Box';
import Divider, { dividerClasses } from '@mui/material/Divider';

export default function VerticalDividers() {
  return (
    <Box
      sx={{
        display: 'flex',
        alignItems: 'center',
        border: '1px solid',
        borderColor: 'divider',
        borderRadius: 1,
        bgcolor: 'background.paper',
        color: 'text.secondary',
        '& svg': {
          m: 1,
        },
        [`& .${dividerClasses.root}`]: {
          mx: 0.5,
        },
      }}
    >
      <FormatAlignLeftIcon />
      <FormatAlignCenterIcon />
      <FormatAlignRightIcon />
      <Divider orientation="vertical" flexItem />
      <FormatBoldIcon />
    </Box>
  );
}
```

Flex item

Use the `flexItem` prop to display the Divider when it's being used in a flex container.

B | ***I***

[Edit in Chat](#)

JS

TS

[Collapse code](#)



```
import FormatBoldIcon from '@mui/icons-material/FormatBold';
import FormatItalicIcon from '@mui/icons-material/FormatItalic';
import Box from '@mui/material/Box';
import Divider from '@mui/material/Divider';

export default function FlexDivider() {
```

```

return (
  <Box
    sx={{
      display: 'inline-flex',
      alignItems: 'center',
      border: '1px solid',
      borderColor: 'divider',
      borderRadius: 2,
      bgcolor: 'background.paper',
      color: 'text.secondary',
      '& svg': {
        m: 1,
      },
    }}
  >
    <FormatBoldIcon />
    <Divider orientation="vertical" variant="middle" flexItem />
    <FormatItalicIcon />
  </Box>
);
}

```

With children



Use the `textAlign` prop to align elements that are wrapped by the Divider.

Lorem ipsum dolor sit amet, consectetur adipiscing elit.

CENTER

Lorem ipsum dolor sit amet, consectetur adipiscing elit.

LEFT

Lorem ipsum dolor sit amet, consectetur adipiscing elit.

RIGHT

Lorem ipsum dolor sit amet, consectetur adipiscing elit.

Chip

Lorem ipsum dolor sit amet, consectetur adipiscing elit.

[Edit in Chat](#)

JS

TS

[Collapse code](#)



```

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

```

```
const Root = styled('div')(({ theme }) => ({
  width: '100%',
  ...theme.typography.body2,
  color: (theme.vars || theme).palette.text.secondary,
  '& > :not(style) ~ :not(style)': {
    marginTop: theme.spacing(2),
  },
}));

export default function DividerText() {
  const content = (
    <p>{`Lorem ipsum dolor sit amet, consectetur adipiscing elit.`}</p>
  );

  return (
    <Root>
      {content}
      <Divider>CENTER</Divider>
      {content}
      <Divider textAlign="left">LEFT</Divider>
      {content}
      <Divider textAlign="right">RIGHT</Divider>
      {content}
      <Divider>
        <Chip label="Chip" size="small" />
      </Divider>
      {content}
    </Root>
  );
}
```

Customization

Use with a List

When using the Divider to separate items in a List, use the `component` prop to render it as an `` — otherwise it won't be a valid HTML element.

Inbox
Drafts
Trash
Spam

```
import List from '@mui/material/List';
import ListItem from '@mui/material/ListItem';
import ListItemText from '@mui/material/ListItemText';
import Divider from '@mui/material/Divider';

const style = {
  p: 0,
  width: '100%',
  maxWidth: 360,
  borderRadius: 2,
  border: '1px solid',
  borderColor: 'divider',
  backgroundColor: 'background.paper',
};

export default function ListDividers() {
  return (
    <List sx={style} aria-label="mailbox folders">
      <ListItem>
        <ListItemText primary="Inbox" />
      </ListItem>
      <Divider component="li" />
      <ListItem>
        <ListItemText primary="Drafts" />
      </ListItem>
      <Divider component="li" />
      <ListItem>
        <ListItemText primary="Trash" />
      </ListItem>
      <Divider component="li" />
      <ListItem>
        <ListItemText primary="Spam" />
      </ListItem>
    </List>
  );
}
```

Icon grouping



The demo below shows how to combine the props `variant="middle"` and `orientation="vertical"`.



```
import Card from '@mui/material/Card';
import Divider, { dividerClasses } from '@mui/material/Divider';
```

```
import FormatAlignLeftIcon from '@mui/icons-material/FormatAlignLeft';
import FormatAlignCenterIcon from '@mui/icons-material/FormatAlignCenter';
import FormatAlignRightIcon from '@mui/icons-material/FormatAlignRight';
import FormatBoldIcon from '@mui/icons-material/FormatBold';

export default function VerticalDividerMiddle() {
  return (
    <Card
      variant="outlined"
      sx={{
        display: 'flex',
        color: 'text.secondary',
        '& svg': {
          m: 1,
        },
        [`& .${dividerClasses.root}`]: {
          mx: 0.5,
        },
      }}
    >
      <FormatAlignLeftIcon />
      <FormatAlignCenterIcon />
      <FormatAlignRightIcon />
      <Divider orientation="vertical" variant="middle" flexItem />
      <FormatBoldIcon />
    </Card>
  );
}
```

Accessibility

Due to its implicit role of `separator`, the `Divider`, which is a `<hr>` element, will be announced by screen readers as a "Horizontal Splitter" (or vertical, if you're using the `orientation` prop).

If you're using it as a purely stylistic element, we recommend setting `aria-hidden="true"` which will make screen readers bypass it.

```
<Divider aria-hidden="true" />
```

Copy

If you're using the `Divider` to wrap other elements, such as text or chips, we recommend changing its rendered element to a plain `<div>` using the `component` prop, and setting `role="presentation"`. This ensures that it's not announced by screen readers while still preserving the semantics of the elements inside it.

```
<Divider component="div" role="presentation">
  <Typography>Text element</Typography>
</Divider>
```

Copy

Anatomy



The Divider component is composed of a root `<hr>`.

```
<hr class="MuiDivider-root">
  <!-- Divider children goes here -->
</hr>
```


Copy

API



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

- `<Divider />`

 [Edit this page](#)

Was this page helpful?  

[< Chip](#)

[Icons >](#)