

Spacing

A wide range of shorthand responsive margin and padding utility classes to modify an element's appearance.

Notation

The space utility converts shorthand margin and padding props to margin and padding CSS declarations. The props are named using the format `{property}{sides}`.

Where *property* is one of:

- `m` - for classes that set *margin*
- `p` - for classes that set *padding*

Where *sides* is one of:

- `t` - for classes that set *margin-top* or *padding-top*
- `b` - for classes that set *margin-bottom* or *padding-bottom*
- `l` - for classes that set *margin-left* or *padding-left*
- `r` - for classes that set *margin-right* or *padding-right*
- `x` - for classes that set both **-left* and **-right*
- `y` - for classes that set both **-top* and **-bottom*
- blank - for classes that set a margin or padding on all 4 sides of the element

Transformation

Depending on the input and the theme configuration, the following transformation is applied:

- input: `number` & theme: `number` : the prop value is multiplied by the theme value.

```
const theme = {
  spacing: 8,
}

<Box sx={{ m: -2 }} /> // margin: -16px;
<Box sx={{ m: 0 }} /> // margin: 0px;
<Box sx={{ m: 0.5 }} /> // margin: 4px;
<Box sx={{ m: 2 }} /> // margin: 16px;
```

- input: `number` & theme: `array` : the prop value is used as the array index.

```
const theme = {
  spacing: [0, 2, 3, 5, 8],
}

<Box sx={{ m: -2 }} /> // margin: -3px;
<Box sx={{ m: 0 }} /> // margin: 0px;
<Box sx={{ m: 2 }} /> // margin: 3px;
```

- input: `number` & theme: `function` : the function is called with the prop value.

```
const theme = {
  spacing: value => value * 2,
}

<Box sx={{ m: 0 }} /> // margin: 0px;
<Box sx={{ m: 2 }} /> // margin: 4px;
```

- input: `string` : the prop value is passed as raw CSS value.

```
<Box sx={{ m: "2rem" }} /> // margin: 2rem;
<Box sx={{ mx: "auto" }} /> // margin-left: auto; margin-right: auto;
```

Example

```
{{"demo": "Demo.js", "defaultCodeOpen": false, "bg": true}}
```

```
<Box sx={{ p: 1 }}>...
<Box sx={{ m: 1 }}>...
<Box sx={{ p: 2 }}>...
```

Horizontal centering

The CSS flex and grid display properties are often used to align elements at the center. However, you can also use `margin-left: auto;`, `margin-right: auto;`, and a width for horizontally centering:

```
{{"demo": "HorizontalCentering.js", "defaultCodeOpen": false, "bg": true}}
```

```
<Box sx={{ mx: "auto", width: 200 }}>...
```

API

```
import { spacing } from '@mui/system';
```

Import name	Prop	CSS property	Theme key
spacing	m	margin	spacing
spacing	mt	margin-top	spacing
spacing	mr	margin-right	spacing
spacing	mb	margin-bottom	spacing
spacing	ml	margin-left	spacing
spacing	mx	margin-left, margin-right	spacing
spacing	my	margin-top, margin-bottom	spacing

spacing	p	padding	spacing
spacing	pt	padding-top	spacing
spacing	pr	padding-right	spacing
spacing	pb	padding-bottom	spacing
spacing	pl	padding-left	spacing
spacing	px	padding-left, padding-right	spacing
spacing	py	padding-top, padding-bottom	spacing

Some people find the prop shorthand confusing, you can use the full version if you prefer:

```
-<Box sx={{ pt: 2 }} />
+<Box sx={{ paddingTop: 2 }} />
```

```
-<Box sx={{ px: 2 }} />
+<Box sx={{ paddingX: 2 }} />
```