

# Badge

Badge generates a small badge to the top-right of its child(ren).

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

## Basic badge

Examples of badges containing text, using primary and secondary colors. The badge is applied to its children.

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

## Color

Use `color` prop to apply theme palette to component.

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

## Customization

Here is an example of customizing the component. You can learn more about this in the [overrides documentation page](#).

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

## Badge visibility

The visibility of badges can be controlled using the `invisible` prop.

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

The badge hides automatically when `badgeContent` is zero. You can override this with the `showZero` prop.

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

## Maximum value

You can use the `max` prop to cap the value of the badge content.

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

## Dot badge

The `dot` prop changes a badge into a small dot. This can be used as a notification that something has changed without giving a count.

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

## Badge overlap

You can use the `overlap` prop to place the badge relative to the corner of the wrapped element.

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

## Badge alignment

You can use the `anchorOrigin` prop to move the badge to any corner of the wrapped element.

```
{{"demo": "BadgeAlignment.js", "hideToolbar": true}}
```

## Accessibility

You can't rely on the content of the badge to be announced correctly. You should provide a full description, for instance, with `aria-label` :

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