

Badge

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



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](#)[Figma](#)[Sketch](#)

Basic badge



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

[Edit in Chat](#)

JS

TS

[Collapse code](#)

```
import Badge from '@mui/material/Badge';
import MailIcon from '@mui/icons-material/Mail';

export default function SimpleBadge() {
  return (
    <Badge badgeContent={4} color="primary">
      <MailIcon color="action" />
    </Badge>
  );
}
```

Color



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

[Edit in Chat](#)

JS

TS

[Collapse code](#)

```
import Badge from '@mui/material/Badge';
import Stack from '@mui/material/Stack';
import MailIcon from '@mui/icons-material/Mail';

export default function ColorBadge() {
  return (
    <Stack spacing={2} direction="row">
      <Badge badgeContent={4} color="secondary">
        <MailIcon color="action" />
      </Badge>
      <Badge badgeContent={4} color="success">
        <MailIcon color="action" />
      </Badge>
    </Stack>
  );
}
```

Customization



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

[Edit in Chat](#)

JS

TS

[Collapse code](#)

```
import Badge, { BadgeProps } from '@mui/material/Badge';
import { styled } from '@mui/material/styles';
import IconButton from '@mui/material/IconButton';
import ShoppingCartIcon from '@mui/icons-material/ShoppingCart';

const StyledBadge = styled(Badge)(({ theme }) => ({
  '& .MuiBadge-badge': {
    right: -3,
    top: 13,
    border: `2px solid ${theme.palette.background.paper}`,
    padding: '0 4px',
  },
}));

export default function CustomizedBadges() {
```

```

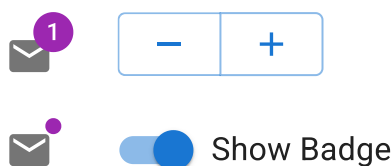
return (
  <IconButton aria-label="cart">
    <StyledBadge badgeContent={4} color="secondary">
      <ShoppingCartIcon />
    </StyledBadge>
  </IconButton>
);
}

```

Badge visibility



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



[Edit in Chat](#)

JS

TS

[Hide code](#)



```

import * as React from 'react';
import Box from '@mui/material/Box';
import Badge from '@mui/material/Badge';
import ButtonGroup from '@mui/material/ButtonGroup';
import Button from '@mui/material/Button';
import AddIcon from '@mui/icons-material/Add';
import RemoveIcon from '@mui/icons-material/Remove';
import MailIcon from '@mui/icons-material/Mail';
import Switch from '@mui/material/Switch';
import FormControlLabel from '@mui/material/FormControlLabel';

export default function BadgeVisibility() {
  const [count, setCount] = React.useState(1);
  const [invisible, setInvisible] = React.useState(false);

  const handleBadgeVisibility = () => {
    setInvisible(!invisible);
  };

  return (
    <Box
      sx={{
        color: 'action.active',
        display: 'flex',
        flexDirection: 'column',
        '& > *': {
          marginBottom: 2,
        },
      }}
    >

```

```

    '& .MuiBadge-root': {
      marginRight: 4,
    },
  }}
>
<div>
  <Badge color="secondary" badgeContent={count}>
    <MailIcon />
  </Badge>
  <ButtonGroup>

```

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



Edit in Chat

JS

TS

Collapse code



```

import Stack from '@mui/material/Stack';
import Badge from '@mui/material/Badge';
import MailIcon from '@mui/icons-material/Mail';

export default function ShowZeroBadge() {
  return (
    <Stack spacing={4} direction="row" sx={{ color: 'action.active' }}>
      <Badge color="secondary" badgeContent={0}>
        <MailIcon />
      </Badge>
      <Badge color="secondary" badgeContent={0} showZero>
        <MailIcon />
      </Badge>
    </Stack>
  );
}

```

Maximum value



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



Edit in Chat

JS

TS

Collapse code



```
import Stack from '@mui/material/Stack';
import Badge from '@mui/material/Badge';
import MailIcon from '@mui/icons-material/Mail';

export default function BadgeMax() {
  return (
    <Stack spacing={4} direction="row" sx={{ color: 'action.active' }}>
      <Badge color="secondary" badgeContent={99}>
        <MailIcon />
      </Badge>
      <Badge color="secondary" badgeContent={100}>
        <MailIcon />
      </Badge>
      <Badge color="secondary" badgeContent={1000} max={999}>
        <MailIcon />
      </Badge>
    </Stack>
  );
}
```

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.


[✦ Edit in Chat](#)
[JS](#)
[TS](#)
[Collapse code](#)


```
import Box from '@mui/material/Box';
import Badge from '@mui/material/Badge';
import MailIcon from '@mui/icons-material/Mail';

export default function DotBadge() {
  return (
    <Box sx={{ color: 'action.active' }}>
      <Badge color="secondary" variant="dot">
        <MailIcon />
      </Badge>
    </Box>
  );
}
```

Badge overlap



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

[Edit in Chat](#)[JS](#)[TS](#)[Collapse code](#)

```
import Box from '@mui/material/Box';
import Stack from '@mui/material/Stack';
import Badge from '@mui/material/Badge';

const shapeStyles = { bgcolor: 'primary.main', width: 40, height: 40 };
const shapeCircleStyles = { borderRadius: '50%' };
const rectangle = <Box component="span" sx={shapeStyles} />;
const circle = (
  <Box component="span" sx={{ ...shapeStyles, ...shapeCircleStyles }} />
);
export default function BadgeOverlap() {
  return (
    <Stack spacing={3} direction="row">
      <Badge color="secondary" badgeContent=" " >
        {rectangle}
      </Badge>
      <Badge color="secondary" badgeContent=" " variant="dot">
        {rectangle}
      </Badge>
      <Badge color="secondary" overlap="circular" badgeContent=" " >
        {circle}
      </Badge>
      <Badge color="secondary" overlap="circular" badgeContent=" " variant="dot">
        {circle}
      </Badge>
    </Stack>
  );
}
```

Badge alignment



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

Vertical

Horizontal

☒ Top

☒ Right



```
<Badge
  anchorOrigin={{
    vertical: 'top',
    horizontal: 'right',
  }}
/>
```

Copy

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`:



Edit in Chat

JS

TS

Collapse code




```
import IconButton from '@mui/material/IconButton';
import Badge from '@mui/material/Badge';
import MailIcon from '@mui/icons-material/Mail';



function notificationsLabel(count: number) {
  if (count === 0) {
    return 'no notifications';
  }
  if (count > 99) {
    return 'more than 99 notifications';
  }
  return `${count} notifications`;
}

export default function AccessibleBadges() {
  return (
    <IconButton aria-label={notificationsLabel(100)}>
      <Badge badgeContent={100} color="secondary">
        <MailIcon />
      </Badge>
    </IconButton>
  );
}
```

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

- `<Badge />`

 [Edit this page](#)

Was this page helpful?  

[< Avatar](#)

[Chip >](#)