Button Group

The ButtonGroup component can be used to group related buttons.

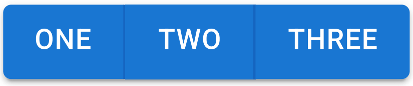
Check out the latest remote job listings from the leading job board for designers, developers, and creative pros.

ads via Carbon

View as Markdown Feedback Bundle size Source Figma Sketch

Basic button group

The buttons can be grouped by wrapping them with the `ButtonGroup` component. They need to be immediate children.

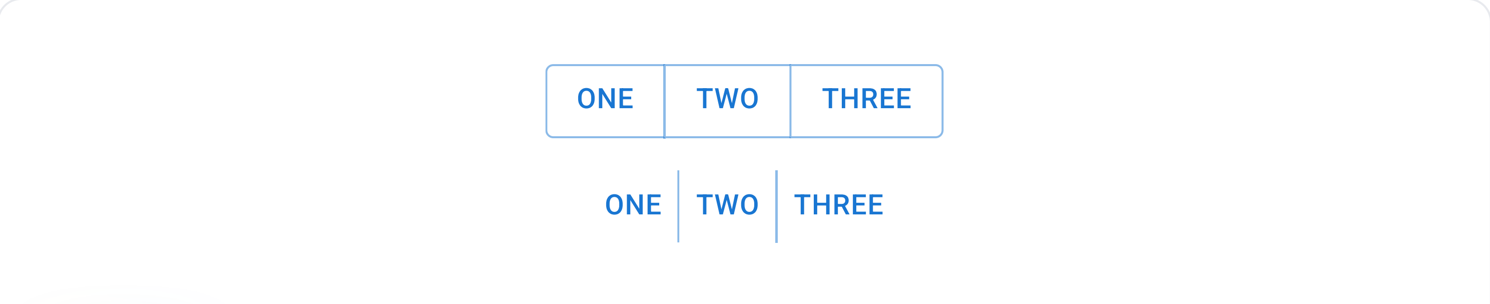


Edit in Chat JS TS Collapse code

```
import Button from '@mui/material/Button';
import ButtonGroup from '@mui/material/ButtonGroup';

export default function BasicButtonGroup() {
  return (
    <ButtonGroup variant="contained" aria-label="Basic button group">
      <Button>One</Button>
      <Button>Two</Button>
      <Button>Three</Button>
    </ButtonGroup>
  );
}
```

Button variants

All the standard button variants are supported.

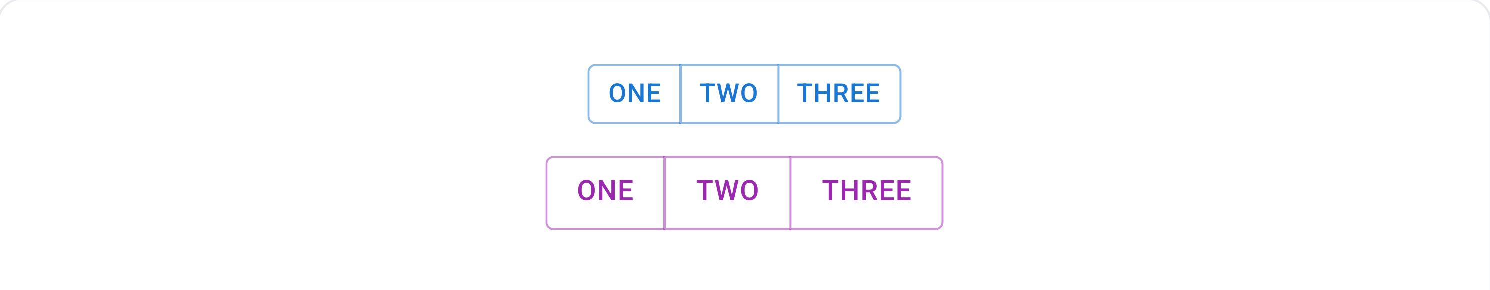


Edit in Chat JS TS Collapse code

```
import Button from '@mui/material/Button';
import ButtonGroup from '@mui/material/ButtonGroup';
import Box from '@mui/material/Box';

export default function VariantButtonGroup() {
  return (
    <Box
      sx={{
        display: 'flex',
        flexDirection: 'column',
        alignItems: 'center',
        '& > *': {
          m: 1,
        },
      }}
    >
      <ButtonGroup variant="outlined" aria-label="Basic button group">
        <Button>One</Button>
        <Button>Two</Button>
        <Button>Three</Button>
      </ButtonGroup>
      <ButtonGroup variant="text" aria-label="Basic button group">
        <Button>One</Button>
        <Button>Two</Button>
        <Button>Three</Button>
      </ButtonGroup>
    </Box>
  );
}
```

Sizes and colors

The `size` and `color` props can be used to control the appearance of the button group.

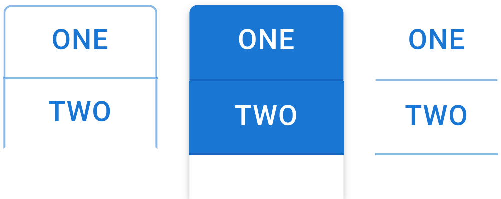


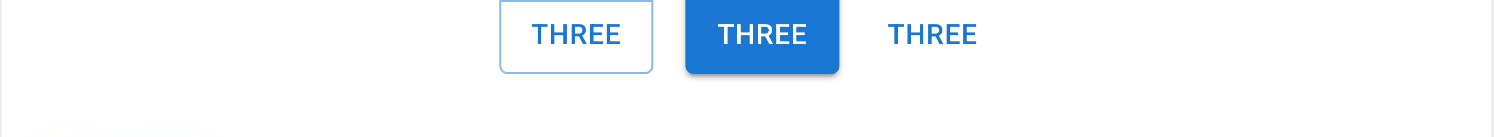
Edit in Chat JS TS Collapse code

```
import Button from '@mui/material/Button';
import Box from '@mui/material/Box';
import ButtonGroup from '@mui/material/ButtonGroup';

const buttons = [
  <Button key="one">One</Button>,
  <Button key="two">Two</Button>,
  <Button key="three">Three</Button>,
];

export default function GroupSizesColors() {
  return (
    <Box
      sx={{
        display: 'flex',
        flexDirection: 'column',
        alignItems: 'center',
        '& > *': {
          m: 1,
        },
      }}
    >
      <ButtonGroup size="small" aria-label="Small button group">
        {buttons}
      </ButtonGroup>
      <ButtonGroup color="secondary" aria-label="Medium-sized button group">
        {buttons}
      </ButtonGroup>
      <ButtonGroup size="large" aria-label="Large button group">
        {buttons}
      </ButtonGroup>
    </Box>
  );
}
```

Vertical group

The button group can be displayed vertically using the `orientation` prop.



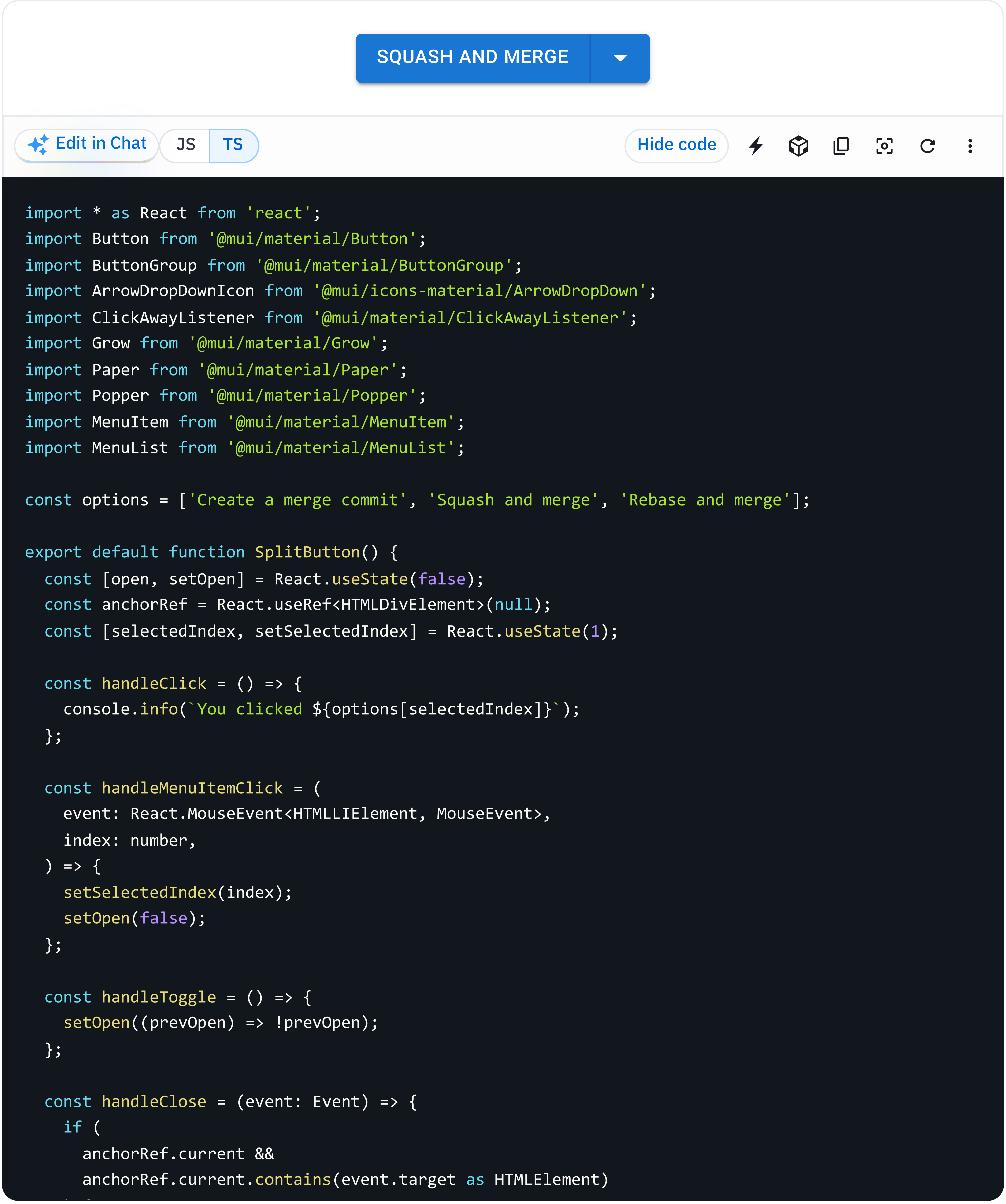


Edit in Chat JS TS Hide code

```tsx
import Button from '@mui/material/Button';
import ButtonGroup from '@mui/material/ButtonGroup';
import Box from '@mui/material/Box';

const buttons = [
  <Button key="one">One</Button>,
  <Button key="two">Two</Button>,
  <Button key="three">Three</Button>,
];

export default function GroupOrientation() {
  return (
    <Box
      sx={{
        display: 'flex',
        '& > *': {
          m: 1,
        },
      }}
    >
      <ButtonGroup orientation="vertical" aria-label="Vertical button group">
        {buttons}
      </ButtonGroup>
      <ButtonGroup
        orientation="vertical"
        aria-label="Vertical button group"
        variant="contained"
      >
        {buttons}
      </ButtonGroup>
      <ButtonGroup
        orientation="vertical"
        aria-label="Vertical button group"
        variant="text"
      >
        {buttons}
      </ButtonGroup>
    </Box>
```

Split button

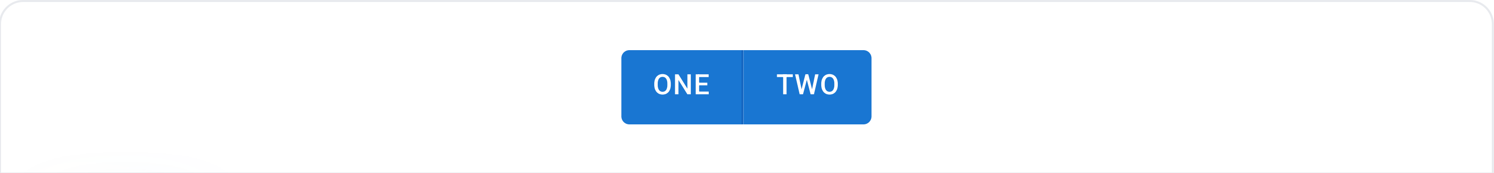
`ButtonGroup` can also be used to create a split button. The dropdown can change the button action (as in this example) or be used to immediately trigger a related action.



```tsx
import * as React from 'react';
import Button from '@mui/material/Button';
import ButtonGroup from '@mui/material/ButtonGroup';
import ArrowDropDownIcon from '@mui/icons-material/ArrowDropDown';
import ClickAwayListener from '@mui/material/ClickAwayListener';
import Grow from '@mui/material/Grow';
import Paper from '@mui/material/Paper';
import Popper from '@mui/material/Popper';
import MenuItem from '@mui/material/MenuItem';
import MenuList from '@mui/material/MenuList';

const options = ['Create a merge commit', 'Squash and merge', 'Rebase and merge'];

export default function SplitButton() {
  const [open, setOpen] = React.useState(false);
  const anchorRef = React.useRef<HTMLDivElement>(null);
  const [selectedIndex, setSelectedIndex] = React.useState(1);

  const handleClick = () => {
    console.info(`You clicked ${options[selectedIndex]}`);
  };

  const handleMenuItemClick = (
    event: React.MouseEvent<HTMLLIElement, MouseEvent>,
    index: number,
  ) => {
    setSelectedIndex(index);
    setOpen(false);
  };

  const handleToggle = () => {
    setOpen((prevOpen) => !prevOpen);
  };

  const handleClose = (event: Event) => {
    if (
      anchorRef.current &&
      anchorRef.current.contains(event.target as HTMLElement)
```

Disabled elevation

You can remove the elevation with the `disableElevation` prop.

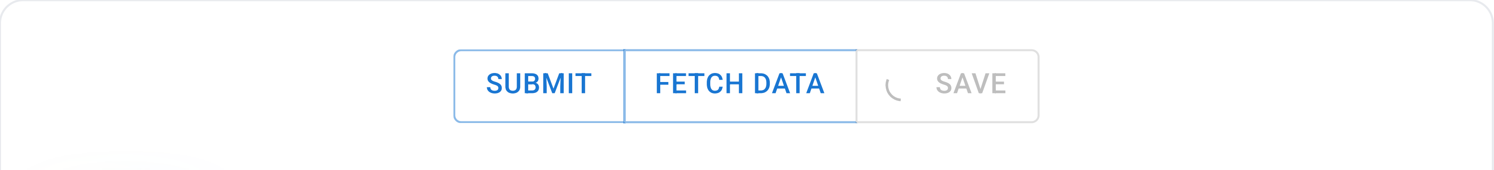


Edit in Chat JS TS Collapse code

```
import ButtonGroup from '@mui/material/ButtonGroup';
import Button from '@mui/material/Button';

export default function DisableElevation() {
  return (
    <ButtonGroup
      disableElevation
      variant="contained"
      aria-label="Disabled button group"
    >
      <Button>One</Button>
      <Button>Two</Button>
    </ButtonGroup>
  );
}
```

Loading

Use the `loading` prop from `Button` to set buttons in a loading state and disable interactions.



Edit in Chat JS TS Collapse code

```
import ButtonGroup from '@mui/material/ButtonGroup';
import Button from '@mui/material/Button';
import SaveIcon from '@mui/icons-material/Save';

export default function LoadingButtonGroup() {
  return (
    <ButtonGroup variant="outlined" aria-label="Loading button group">
      <Button>Submit</Button>
      <Button>Fetch data</Button>
      <Button loading loadingPosition="start" startIcon={<SaveIcon />}>
        Save
      </Button>
    </ButtonGroup>
  );
}
```

API

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

- `<Button />`
- `<ButtonGroup />`

Edit this page

Was this page helpful?

‹ Button

Checkbox ›