When To Use

When there are more than a few options to choose from, you can wrap them in a `Dropdown`. By hovering or clicking on the trigger, a dropdown menu will appear, which allows you to choose an option and execute the relevant action.

Examples

Basic

```
import React from 'react';
import { DownOutlined, SmileOutlined } from '@ant-design/icons';
import type { MenuProps } from 'antd';
import { Dropdown, Space } from 'antd';

const items: MenuProps['items'] = [
  {
    key: '1',
    label: (
      <a target="_blank" rel="noopener noreferrer"
href="https://www.antgroup.com">
        1st menu item
      </a>
    ),
  },
  {
    key: '2',
    label: (
      <a target="_blank" rel="noopener noreferrer"
href="https://www.aliyun.com">
        2nd menu item (disabled)
      </a>
    ),
    icon: <SmileOutlined />,
    disabled: true,
  },
  {
    key: '3',
    label: (
      <a target="_blank" rel="noopener noreferrer"
href="https://www.luohanacademy.com">
        3rd menu item (disabled)
      </a>
    ),
    disabled: true,
  },
```

```
    {
      key: '4',
      danger: true,
      label: 'a danger item',
    },
];

const App: React.FC = () => (
  <Dropdown menu={{ items }}>
    <a onClick={(e) => e.preventDefault()}>
      <Space>
        Hover me
        <DownOutlined />
      </Space>
    </a>
  </Dropdown>
);

export default App;
```

Extra node

v5.21.0

```
import React from 'react';
import { DownOutlined, SettingOutlined } from '@ant-design/icons';
import type { MenuProps } from 'antd';
import { Dropdown, Space } from 'antd';

const items: MenuProps['items'] = [
  {
    key: '1',
    label: 'My Account',
    disabled: true,
  },
  {
    type: 'divider',
  },
  {
    key: '2',
    label: 'Profile',
    extra: '⌘P',
  },
  {
    key: '3',
    label: 'Billing',
```

```
    extra: '⌘B',
  },
  {
    key: '4',
    label: 'Settings',
    icon: <SettingOutlined />,
    extra: '⌘S',
  },
];

const App: React.FC = () => (
  <Dropdown menu={{ items }}>
    <a onClick={(e) => e.preventDefault()}>
      <Space>
        Hover me
        <DownOutlined />
      </Space>
    </a>
  </Dropdown>
);

export default App;
```

Placement

```
import React from 'react';
import type { MenuProps } from 'antd';
import { Button, Dropdown, Space } from 'antd';

const items: MenuProps['items'] = [
  {
    key: '1',
    label: (
      <a target="_blank" rel="noopener noreferrer"
href="https://www.antgroup.com">
        1st menu item
      </a>
    ),
  },
  {
    key: '2',
    label: (
      <a target="_blank" rel="noopener noreferrer"
href="https://www.aliyun.com">
        2nd menu item
      </a>
```

```
    ),
  },
  {
    key: '3',
    label: (
      <a target="_blank" rel="noopener noreferrer"
href="https://www.luohanacademy.com">
        3rd menu item
      </a>
    ),
  },
];

const App: React.FC = () => (
  <Space direction="vertical">
    <Space wrap>
      <Dropdown menu={{ items }} placement="bottomLeft">
        <Button>bottomLeft</Button>
      </Dropdown>
      <Dropdown menu={{ items }} placement="bottom">
        <Button>bottom</Button>
      </Dropdown>
      <Dropdown menu={{ items }} placement="bottomRight">
        <Button>bottomRight</Button>
      </Dropdown>
    </Space>
    <Space wrap>
      <Dropdown menu={{ items }} placement="topLeft">
        <Button>topLeft</Button>
      </Dropdown>
      <Dropdown menu={{ items }} placement="top">
        <Button>top</Button>
      </Dropdown>
      <Dropdown menu={{ items }} placement="topRight">
        <Button>topRight</Button>
      </Dropdown>
    </Space>
  </Space>
);

export default App;
```

Arrow

```
import React from 'react';
import type { MenuProps } from 'antd';
```

```tsx
import { Button, Dropdown, Space } from 'antd';

const items: MenuProps['items'] = [
  {
    key: '1',
    label: (
      <a target="_blank" rel="noopener noreferrer"
href="https://www.antgroup.com">
        1st menu item
      </a>
    ),
  },
  {
    key: '2',
    label: (
      <a target="_blank" rel="noopener noreferrer"
href="https://www.aliyun.com">
        2nd menu item
      </a>
    ),
  },
  {
    key: '3',
    label: (
      <a target="_blank" rel="noopener noreferrer"
href="https://www.luohanacademy.com">
        3rd menu item
      </a>
    ),
  },
];

const App: React.FC = () => (
  <Space direction="vertical">
    <Space wrap>
      <Dropdown menu={{ items }} placement="bottomLeft" arrow>
        <Button>bottomLeft</Button>
      </Dropdown>
      <Dropdown menu={{ items }} placement="bottom" arrow>
        <Button>bottom</Button>
      </Dropdown>
      <Dropdown menu={{ items }} placement="bottomRight" arrow>
        <Button>bottomRight</Button>
      </Dropdown>
    </Space>
    <Space wrap>
```

```
            <Dropdown menu={{ items }} placement="topLeft" arrow>
              <Button>topLeft</Button>
            </Dropdown>
            <Dropdown menu={{ items }} placement="top" arrow>
              <Button>top</Button>
            </Dropdown>
            <Dropdown menu={{ items }} placement="topRight" arrow>
              <Button>topRight</Button>
            </Dropdown>
          </Space>
        </Space>
    );

export default App;
```

Other elements

```
import React from 'react';
import { DownOutlined } from '@ant-design/icons';
import type { MenuProps } from 'antd';
import { Dropdown, Space } from 'antd';

const items: MenuProps['items'] = [
  {
    label: (
      <a target="_blank" rel="noopener noreferrer"
href="https://www.antgroup.com">
        1st menu item
      </a>
    ),
    key: '0',
  },
  {
    label: (
      <a target="_blank" rel="noopener noreferrer"
href="https://www.aliyun.com">
        2nd menu item
      </a>
    ),
    key: '1',
  },
  {
    type: 'divider',
  },
  {
    label: '3rd menu item (disabled) ',
```

```
      key: '3',
      disabled: true,
    },
  ];

  const App: React.FC = () => (
    <Dropdown menu={{ items }}>
      <a onClick={(e) => e.preventDefault()}>
        <Space>
          Hover me
          <DownOutlined />
        </Space>
      </a>
    </Dropdown>
  );

  export default App;
```

Arrow pointing at the center

```
import React from 'react';
import type { MenuProps } from 'antd';
import { Button, Dropdown, Space } from 'antd';

const items: MenuProps['items'] = [
  {
    key: '1',
    label: (
      <a target="_blank" rel="noopener noreferrer"
href="https://www.antgroup.com">
        1st menu item
      </a>
    ),
  },
  {
    key: '2',
    label: (
      <a target="_blank" rel="noopener noreferrer"
href="https://www.aliyun.com">
        2nd menu item
      </a>
    ),
  },
  {
    key: '3',
    label: (
```

```
      <a target="_blank" rel="noopener noreferrer"
href="https://www.luohanacademy.com">
        3rd menu item
      </a>
    ),
  },
];

const App: React.FC = () => (
  <Space direction="vertical">
    <Space wrap>
      <Dropdown menu={{ items }} placement="bottomLeft" arrow={{
pointAtCenter: true }}>
        <Button>bottomLeft</Button>
      </Dropdown>
      <Dropdown menu={{ items }} placement="bottom" arrow={{ pointAtCenter:
true }}>
        <Button>bottom</Button>
      </Dropdown>
      <Dropdown menu={{ items }} placement="bottomRight" arrow={{
pointAtCenter: true }}>
        <Button>bottomRight</Button>
      </Dropdown>
    </Space>
    <Space wrap>
      <Dropdown menu={{ items }} placement="topLeft" arrow={{
pointAtCenter: true }}>
        <Button>topLeft</Button>
      </Dropdown>
      <Dropdown menu={{ items }} placement="top" arrow={{ pointAtCenter:
true }}>
        <Button>top</Button>
      </Dropdown>
      <Dropdown menu={{ items }} placement="topRight" arrow={{
pointAtCenter: true }}>
        <Button>topRight</Button>
      </Dropdown>
    </Space>
  </Space>
);

export default App;
```

Trigger mode

```
import React from 'react';
import { DownOutlined } from '@ant-design/icons';
import type { MenuProps } from 'antd';
import { Dropdown, Space } from 'antd';

const items: MenuProps['items'] = [
  {
    label: (
      <a href="https://www.antgroup.com" target="_blank" rel="noopener
noreferrer">
        1st menu item
      </a>
    ),
    key: '0',
  },
  {
    label: (
      <a href="https://www.aliyun.com" target="_blank" rel="noopener
noreferrer">
        2nd menu item
      </a>
    ),
    key: '1',
  },
  {
    type: 'divider',
  },
  {
    label: '3rd menu item',
    key: '3',
  },
];

const App: React.FC = () => (
  <Dropdown menu={{ items }} trigger={['click']}>
    <a onClick={(e) => e.preventDefault()}>
      <Space>
        Click me
        <DownOutlined />
      </Space>
    </a>
  </Dropdown>
);

export default App;
```

Click event

```
import React from 'react';
import { DownOutlined } from '@ant-design/icons';
import type { MenuProps } from 'antd';
import { Dropdown, message, Space } from 'antd';

const onClick: MenuProps['onClick'] = ({ key }) => {
  message.info(`Click on item ${key}`);
};

const items: MenuProps['items'] = [
  {
    label: '1st menu item',
    key: '1',
  },
  {
    label: '2nd menu item',
    key: '2',
  },
  {
    label: '3rd menu item',
    key: '3',
  },
];

const App: React.FC = () => (
  <Dropdown menu={{ items, onClick }}>
    <a onClick={(e) => e.preventDefault()}>
      <Space>
        Hover me, Click menu item
        <DownOutlined />
      </Space>
    </a>
  </Dropdown>
);

export default App;
```

Button with dropdown menu

```
import React from 'react';
import { DownOutlined, UserOutlined } from '@ant-design/icons';
import type { MenuProps } from 'antd';
import { Button, Dropdown, message, Space, Tooltip } from 'antd';
```

```tsx
const handleButtonClick = (e: React.MouseEvent<HTMLButtonElement>) => {
  message.info('Click on left button.');
  console.log('click left button', e);
};

const handleMenuClick: MenuProps['onClick'] = (e) => {
  message.info('Click on menu item.');
  console.log('click', e);
};

const items: MenuProps['items'] = [
  {
    label: '1st menu item',
    key: '1',
    icon: <UserOutlined />,
  },
  {
    label: '2nd menu item',
    key: '2',
    icon: <UserOutlined />,
  },
  {
    label: '3rd menu item',
    key: '3',
    icon: <UserOutlined />,
    danger: true,
  },
  {
    label: '4rd menu item',
    key: '4',
    icon: <UserOutlined />,
    danger: true,
    disabled: true,
  },
];

const menuProps = {
  items,
  onClick: handleMenuClick,
};

const App: React.FC = () => (
  <Space wrap>
    <Dropdown.Button menu={menuProps} onClick={handleButtonClick}>
      Dropdown
```

```jsx
      </Dropdown.Button>
      <Dropdown.Button menu={menuProps} placement="bottom" icon=
{<UserOutlined />}>
        Dropdown
      </Dropdown.Button>
      <Dropdown.Button menu={menuProps} onClick={handleButtonClick} disabled>
        Dropdown
      </Dropdown.Button>
      <Dropdown.Button
        menu={menuProps}
        buttonsRender={([leftButton, rightButton]) => [
          <Tooltip title="tooltip" key="leftButton">
            {leftButton}
          </Tooltip>,
          React.cloneElement(rightButton as React.ReactElement<any, string>,
{ loading: true }),
        ]}
      >
        With Tooltip
      </Dropdown.Button>
      <Dropdown menu={menuProps}>
        <Button>
          <Space>
            Button
            <DownOutlined />
          </Space>
        </Button>
      </Dropdown>
      <Dropdown.Button menu={menuProps} onClick={handleButtonClick} danger>
        Danger
      </Dropdown.Button>
    </Space>
);

export default App;
```

Custom dropdown

```jsx
import React from 'react';
import { DownOutlined } from '@ant-design/icons';
import { Button, Divider, Dropdown, Space, theme } from 'antd';
import type { MenuProps } from 'antd';

const { useToken } = theme;
```

```
const items: MenuProps['items'] = [
  {
    key: '1',
    label: (
      <a target="_blank" rel="noopener noreferrer"
href="https://www.antgroup.com">
        1st menu item
      </a>
    ),
  },
  {
    key: '2',
    label: (
      <a target="_blank" rel="noopener noreferrer"
href="https://www.aliyun.com">
        2nd menu item (disabled)
      </a>
    ),
    disabled: true,
  },
  {
    key: '3',
    label: (
      <a target="_blank" rel="noopener noreferrer"
href="https://www.luohanacademy.com">
        3rd menu item (disabled)
      </a>
    ),
    disabled: true,
  },
];

const App: React.FC = () => {
  const { token } = useToken();

  const contentStyle: React.CSSProperties = {
    backgroundColor: token.colorBgElevated,
    borderRadius: token.borderRadiusLG,
    boxShadow: token.boxShadowSecondary,
  };

  const menuStyle: React.CSSProperties = {
    boxShadow: 'none',
  };

  return (
```

```
      <Dropdown
        menu={{ items }}
        dropdownRender={(menu) => (
          <div style={contentStyle}>
            {React.cloneElement(
              menu as React.ReactElement<{
                style: React.CSSProperties;
              }>,
              { style: menuStyle },
            )}
            <Divider style={{ margin: 0 }} />
            <Space style={{ padding: 8 }}>
              <Button type="primary">Click me!</Button>
            </Space>
          </div>
        )}
      >
        <a onClick={(e) => e.preventDefault()}>
          <Space>
            Hover me
            <DownOutlined />
          </Space>
        </a>
      </Dropdown>
  );
};

export default App;
```

Cascading menu

```
import React from 'react';
import { DownOutlined } from '@ant-design/icons';
import type { MenuProps } from 'antd';
import { Dropdown, Space } from 'antd';

const items: MenuProps['items'] = [
  {
    key: '1',
    type: 'group',
    label: 'Group title',
    children: [
      {
        key: '1-1',
        label: '1st menu item',
      },
```

```
      {
        key: '1-2',
        label: '2nd menu item',
      },
    ],
  },
  {
    key: '2',
    label: 'sub menu',
    children: [
      {
        key: '2-1',
        label: '3rd menu item',
      },
      {
        key: '2-2',
        label: '4th menu item',
      },
    ],
  },
  {
    key: '3',
    label: 'disabled sub menu',
    disabled: true,
    children: [
      {
        key: '3-1',
        label: '5d menu item',
      },
      {
        key: '3-2',
        label: '6th menu item',
      },
    ],
  },
];

const App: React.FC = () => (
  <Dropdown menu={{ items }}>
    <a onClick={(e) => e.preventDefault()}>
      <Space>
        Cascading menu
        <DownOutlined />
      </Space>
    </a>
  </Dropdown>
```

```
  );

  export default App;
```

Cascading menu

Debug

```
import React from 'react';
import { DownOutlined } from '@ant-design/icons';
import type { MenuProps } from 'antd';
import { Dropdown, Space } from 'antd';

const items: MenuProps['items'] = [
  {
    key: '1',
    type: 'group',
    label: 'Group title',
    children: [
      {
        key: '1-1',
        label: '1st menu item',
      },
      {
        key: '1-2',
        label: '2nd menu item',
      },
    ],
  },
  {
    key: '2',
    label: 'sub menu',
    children: [
      {
        key: '2-1',
        label: '3rd menu item',
      },
      {
        key: '2-2',
        label: '4th menu item',
      },
    ],
  },
  {
    key: '3',
    label: 'disabled sub menu',
```

```
      disabled: true,
      children: [
        {
          key: '3-1',
          label: '5d menu item',
        },
        {
          key: '3-2',
          label: '6th menu item',
        },
      ],
    },
];

const App: React.FC = () => (
  <div style={{ height: 200 }}>
    <Dropdown menu={{ items, openKeys: ['2'] }} open autoAdjustOverflow=
{false}>
      <a onClick={(e) => e.preventDefault()}>
        <Space>
          Cascading menu
          <DownOutlined />
        </Space>
      </a>
    </Dropdown>
  </div>
);

export default App;
```

The way of hiding menu.

```
import React, { useState } from 'react';
import { DownOutlined } from '@ant-design/icons';
import type { DropdownProps, MenuProps } from 'antd';
import { Dropdown, Space } from 'antd';

const App: React.FC = () => {
  const [open, setOpen] = useState(false);

  const handleMenuClick: MenuProps['onClick'] = (e) => {
    if (e.key === '3') {
      setOpen(false);
    }
  };
```

```
    const handleOpenChange: DropdownProps['onOpenChange'] = (nextOpen, info)
=> {
    if (info.source === 'trigger' || nextOpen) {
      setOpen(nextOpen);
    }
  };

  const items: MenuProps['items'] = [
    {
      label: 'Clicking me will not close the menu.',
      key: '1',
    },
    {
      label: 'Clicking me will not close the menu also.',
      key: '2',
    },
    {
      label: 'Clicking me will close the menu.',
      key: '3',
    },
  ];

  return (
    <Dropdown
      menu={{
        items,
        onClick: handleMenuClick,
      }}
      onOpenChange={handleOpenChange}
      open={open}
    >
      <a onClick={(e) => e.preventDefault()}>
        <Space>
          Hover me
          <DownOutlined />
        </Space>
      </a>
    </Dropdown>
  );
};

export default App;
```

Context Menu

```tsx
import React from 'react';
import type { MenuProps } from 'antd';
import { Dropdown, theme } from 'antd';

const items: MenuProps['items'] = [
  {
    label: '1st menu item',
    key: '1',
  },
  {
    label: '2nd menu item',
    key: '2',
  },
  {
    label: '3rd menu item',
    key: '3',
  },
];

const App: React.FC = () => {
  const {
    token: { colorBgLayout, colorTextTertiary },
  } = theme.useToken();

  return (
    <Dropdown menu={{ items }} trigger={['contextMenu']}>
      <div
        style={{
          color: colorTextTertiary,
          background: colorBgLayout,
          height: 200,
          textAlign: 'center',
          lineHeight: '200px',
        }}
      >
        Right Click on here
      </div>
    </Dropdown>
  );
};

export default App;
```

Loading

```tsx
import React, { useState } from 'react';
import { DownOutlined } from '@ant-design/icons';
import type { MenuProps } from 'antd';
import { Dropdown, Space } from 'antd';

const items: MenuProps['items'] = [
  {
    label: 'Submit and continue',
    key: '1',
  },
];

const App: React.FC = () => {
  const [loadings, setLoadings] = useState<boolean[]>([]);

  const enterLoading = (index: number) => {
    setLoadings((state) => {
      const newLoadings = [...state];
      newLoadings[index] = true;
      return newLoadings;
    });

    setTimeout(() => {
      setLoadings((state) => {
        const newLoadings = [...state];
        newLoadings[index] = false;
        return newLoadings;
      });
    }, 6000);
  };

  return (
    <Space direction="vertical">
      <Dropdown.Button type="primary" loading menu={{ items }}>
        Submit
      </Dropdown.Button>
      <Dropdown.Button type="primary" size="small" loading menu={{ items }}>
        Submit
      </Dropdown.Button>
      <Dropdown.Button
        type="primary"
        loading={loadings[0]}
        menu={{ items }}
        onClick={() => enterLoading(0)}
      >
```

```
        Submit
      </Dropdown.Button>
      <Dropdown.Button
        icon={<DownOutlined />}
        loading={loadings[1]}
        menu={{ items }}
        onClick={() => enterLoading(1)}
      >
        Submit
      </Dropdown.Button>
    </Space>
  );
};

export default App;
```

Selectable Menu

```
import React from 'react';
import { DownOutlined } from '@ant-design/icons';
import type { MenuProps } from 'antd';
import { Dropdown, Space, Typography } from 'antd';

const items: MenuProps['items'] = [
  {
    key: '1',
    label: 'Item 1',
  },
  {
    key: '2',
    label: 'Item 2',
  },
  {
    key: '3',
    label: 'Item 3',
  },
];

const App: React.FC = () => (
  <Dropdown
    menu={{
      items,
      selectable: true,
      defaultSelectedKeys: ['3'],
    }}
  >
```

```jsx
    <Typography.Link>
      <Space>
        Selectable
        <DownOutlined />
      </Space>
    </Typography.Link>
  </Dropdown>
);

export default App;
```

Menu full styles

Debug

```tsx
import React from 'react';
import { AppstoreOutlined, DownOutlined, MailOutlined, SettingOutlined }
from '@ant-design/icons';
import type { MenuProps } from 'antd';
import { Dropdown, Space } from 'antd';

type MenuItem = Required<MenuProps>['items'][number];

function getItem(
  label: React.ReactNode,
  key: React.Key,
  icon?: React.ReactNode,
  children?: MenuItem[],
  type?: 'group',
): MenuItem {
  return {
    key,
    icon,
    children,
    label,
    type,
  } as MenuItem;
}

const items: MenuItem[] = [
  getItem(
    'Item Group',
    'group',
    null,
    [getItem('Option 0', '01'), getItem('Option 0', '02')],
    'group',
```

```
  ),
  getItem('Navigation One', 'sub1', <MailOutlined />, [
    getItem('Item 1', 'g1', null, [getItem('Option 1', '1'),
getItem('Option 2', '2')], 'group'),
    getItem('Item 2', 'g2', null, [getItem('Option 3', '3'),
getItem('Option 4', '4')], 'group'),
  ]),
  getItem('Navigation Two', 'sub2', <AppstoreOutlined />, [
    getItem('Option 5', '5'),
    getItem('Option 6', '6'),
    getItem('Submenu', 'sub3', null, [getItem('Option 7', '7'),
getItem('Option 8', '8')]),
  ]),
  getItem('Navigation Three', 'sub4', <SettingOutlined />, [
    getItem('Option 9', '9'),
    getItem('Option 10', '10'),
    getItem('Option 11', '11'),
    getItem('Option 12', '12'),
  ]),
  // Not crash
  null as any,
];

const App: React.FC = () => (
  <Dropdown
    menu={{
      items,
      selectedKeys: ['1'],
      openKeys: ['sub1'],
    }}
  >
    <a onClick={(e) => e.preventDefault()}>
      <Space>
        Hover to check menu style
        <DownOutlined />
      </Space>
    </a>
  </Dropdown>
);

export default App;
```

_InternalPanelDoNotUseOrYouWillBeFired

Debug

```jsx
import React from 'react';
import { SmileOutlined } from '@ant-design/icons';
import { Dropdown } from 'antd';

const { _InternalPanelDoNotUseOrYouWillBeFired: InternalDropdown } =
Dropdown;

const menu = [
  {
    key: '1',
    label: (
      <a target="_blank" rel="noopener noreferrer"
href="https://www.antgroup.com">
        1st menu item
      </a>
    ),
  },
  {
    key: '2',
    label: (
      <a target="_blank" rel="noopener noreferrer"
href="https://www.aliyun.com">
        2nd menu item (disabled)
      </a>
    ),
    icon: <SmileOutlined />,
    disabled: true,
  },
  {
    key: '3',
    label: (
      <a target="_blank" rel="noopener noreferrer"
href="https://www.luohanacademy.com">
        3rd menu item (disabled)
      </a>
    ),
    disabled: true,
  },
  {
    key: '4',
    danger: true,
    label: 'a danger item',
  },
];

const App: React.FC = () => <InternalDropdown menu={{ items: menu }} />;
```

```
export default App;
```

Icon debug

Debug

```jsx
import React from 'react';
import { DownOutlined } from '@ant-design/icons';
import { Dropdown, Space } from 'antd';

const App: React.FC = () => (
  <Space>
    <Dropdown.Button icon={<DownOutlined />} menu={{ items: [] }}>
      Submit
    </Dropdown.Button>
  </Space>
);

export default App;
```

API

Common props ref: [Common props](#)

Dropdown

Property	Description	Type	Defau
arrow	Whether the dropdown arrow should be visible	boolean | { pointAtCenter: boolean }	false
autoAdjustOverflow	Whether to adjust dropdown placement automatically when dropdown is off screen	boolean	true
autoFocus	Focus element in overlay when opened	boolean	false
disabled	Whether the dropdown menu is disabled	boolean	-

destroyPopupOnHide	Whether destroy dropdown when hidden	boolean	false
dropdownRender	Customize dropdown content	(menus: ReactNode) => ReactNode	-
getPopupContainer	To set the container of the dropdown menu. The default is to create a div element in body, but you can reset it to the scrolling area and make a relative reposition. Example on CodePen	(triggerNode: HTMLElement) => HTMLElement	() => document
menu	The menu props	MenuProps	-
overlayClassName	The class name of the dropdown root element	string	-
overlayStyle	The style of the dropdown root element	CSSProperties	-
placement	Placement of popup menu: `bottom` `bottomLeft` `bottomRight top` `topLeft` `topRight`	string	`bottomLe`
trigger	The trigger mode which executes the dropdown action. Note that hover can't be used on touchscreens	Array<click|hover|contextMenu>	[hover]
open	Whether the dropdown menu is currently open.	boolean	-

	Use `visible` under 4.23.0 ([why?](#))		
onOpenChange	Called when the open state is changed. Not trigger when hidden by click item. Use `onVisibleChange` under 4.23.0 ([why?](#))	(open: boolean, info: { source: 'trigger' | 'menu' }) => void	-

Dropdown.Button

Same props from Dropdown. And includes additional props:

Property	Description	Type	Default	Versi
buttonsRender	Custom buttons inside Dropdown.Button	(buttons: ReactNode[]) => ReactNode[]	-	
loading	Set the loading status of button, the same as [Button](#)	boolean | { delay: number, icon: ReactNode }	false	icon: 5.23.(
danger	Set the danger status of button	boolean	-	4.23.(
icon	Icon (appears on the right)	ReactNode	-	
size	Size of the button, the same as [Button](#)	`large` | `middle` | `small`	`middle`	
type	Type of the button, the same as [Button](#)	`primary` | `dashed` | `link` | `text` | `default`	`default`	
onClick	The same as [Button](#): called when you click the button on the left	(event: React.MouseEvent<HTMLElement, MouseEvent>) => void	-	

Note

Please ensure that the child node of `Dropdown` accepts `onMouseEnter`, `onMouseLeave`, `onFocus`, `onClick` events.

Design Token

FAQ

How to prevent Dropdown from being squeezed when it exceeds the screen horizontally?

You can use `width: max-content` style to handle this. ref [#43025](#).