

何时使用 {#when-to-use}

当页面上的操作命令过多时，用此组件可以收纳操作元素。点击或移入触点，会出现一个下拉菜单。可在列表中进行选择，并执行相应的命令。

- 用于收罗一组命令操作。
- Select 用于选择，而 Dropdown 是命令集合。

代码演示

基本

```
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;

```

额外节点

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: '⌵',
  },
  {
    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;

```

弹出位置

```

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;

```

箭头

```

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>
        <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>
);

```

```

    </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;

```

其他元素

```

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;

```

箭头指向

```

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;

```

触发方式


```

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
norereferrer">
        1st menu item
      </a>
    ),
    key: '0',
  },
  {
    label: (
      <a href="https://www.aliyun.com" target="_blank" rel="noopener
norereferrer">
        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;

```

触发事件

```
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;
```

带下拉框的按钮

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

```
const handleClick = (e: React.MouseEvent<HTMLButtonElement>) => {
  message.info('Click on left button.');
```

console.log('click left button', e);

```
};

const handleClick: 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: handleClick,
};

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

```

    </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;

```

扩展菜单

```

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;

```

多级菜单

```

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;
```

多级菜单

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;

```

菜单隐藏方式

```

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;

```

右键菜单

```
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;
```

加载中状态

```

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;

```

菜单可选选择

```

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'],
    }}
  >

```

```

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

export default App;

```

Menu 完整样式

Debug

```

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

```

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

```
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

通用属性参考: [通用属性](#)

Dropdown

| 参数 | 说明 | 类型 | 默认值 |
|--------------------|---|---|----------------|
| arrow | 下拉框箭头是否显示 | boolean { pointAtCenter: boolean } | false |
| autoAdjustOverflow | 下拉框被遮挡时自动调整位置 | boolean | true |
| autoFocus | 打开后自动聚焦下拉框 | boolean | false |
| disabled | 菜单是否禁用 | boolean | - |
| destroyPopupOnHide | 关闭后是否销毁 Dropdown | boolean | false |
| dropdownRender | 自定义下拉框内容 | (menus: ReactNode) => ReactNode | - |
| getPopupContainer | 菜单渲染父节点。默认渲染到 body 上，如果你遇到菜单滚动定位问题，试试修改 | (triggerNode: HTMLElement) => HTMLElement | () => document |

| | | | |
|------------------|---|---|------------|
| | 为滚动的区域，并相对其定位。 示例 | | |
| menu | 菜单配置项 | MenuProps | - |
| overlayClassName | 下拉根元素的类名称 | string | - |
| overlayStyle | 下拉根元素的样式 | CSSProperties | - |
| placement | 菜单弹出位置： bottom bottomLeft bottomRight top topLeft topRight | string | bottomLeft |
| trigger | 触发下拉的行为，移动端不支持 hover | Array<click hover contextMenu> | [hover] |
| open | 菜单是否显示，小于 4.23.0 使用 visible (为什么?) | boolean | - |
| onOpenChange | 菜单显示状态改变时调用，点击菜单按钮导致的消失不会触发。小于 4.23.0 使用 onVisibleChange (为什么?) | (open: boolean, info: { source: 'trigger' 'menu' }) => void | - |

Dropdown.Button

属性与 Dropdown 的相同。还包含以下属性：

| 参数 | 说明 | 类型 | 默认值 | 版本 |
|---------------|--------------------------------------|--|-------|--------------|
| buttonsRender | 自定义左右两个按钮 | (buttons: ReactNode[]) => ReactNode[] | - | |
| loading | 设置按钮载入状态，和 Button 一致 | boolean { delay: number, icon: ReactNode } | false | icon: 5.23.0 |
| danger | 设置危险按钮 | boolean | - | 4.23.0 |

| | | | | |
|---------|--|--|---------|--|
| icon | 右侧的 icon | ReactNode | - | |
| size | 按钮大小, 和 Button 一致 | large middle small | middle | |
| type | 按钮类型, 和 Button 一致 | primary dashed link text default | default | |
| onClick | 点击左侧按钮的回调, 和 Button 一致 | (event: React.MouseEvent<HTMLElement, MouseEvent>) => void | - | |

注意

请确保 `Dropdown` 的子元素能接受 `onmouseenter`、`onmouseleave`、`onfocus`、`onclick` 事件。

主题变量 (Design Token)

FAQ

Dropdown 在水平方向超出屏幕时会被挤压该怎么办？

你可以通过 `width: max-content` 来解决这个问题, 参考 [#43025](#)。