

## 何时使用 {#when-to-use}

- 用于网站上的全局功能；
- 无论浏览到何处都可以看见的按钮。

## 代码演示

### 基本

```
import React from 'react';
import { FloatButton } from 'antd';

const App: React.FC = () => <FloatButton onClick={() =>
  console.log('onClick')} />;

export default App;
```

### 类型

```
import React from 'react';
import { QuestionCircleOutlined } from '@ant-design/icons';
import { FloatButton } from 'antd';

const App: React.FC = () => (
  <>
    <FloatButton icon={<QuestionCircleOutlined />} type="primary" style={{
      insetInlineEnd: 24 }} />
    <FloatButton icon={<QuestionCircleOutlined />} type="default" style={{
      insetInlineEnd: 94 }} />
  </>
);

export default App;
```

### 形状

```
import React from 'react';
import { CustomerServiceOutlined } from '@ant-design/icons';
import { FloatButton } from 'antd';

const App: React.FC = () => (
  <>
    <FloatButton
      shape="circle"
      type="primary"
    />
  </>
);
```

```

        style={{ insetInlineEnd: 94 }}
        icon={<CustomerServiceOutlined />}
      />
    <FloatButton
      shape="square"
      type="primary"
      style={{ insetInlineEnd: 24 }}
      icon={<CustomerServiceOutlined />}
    />
  </>
);

export default App;

```

## 描述

```

import React from 'react';
import { FileTextOutlined } from '@ant-design/icons';
import { FloatButton } from 'antd';

const App: React.FC = () => (
  <>
    <FloatButton
      icon={<FileTextOutlined />}
      description="HELP INFO"
      shape="square"
      style={{ insetInlineEnd: 24 }}
    />
    <FloatButton description="HELP INFO" shape="square" style={{
insetInlineEnd: 94 }} />
    <FloatButton
      icon={<FileTextOutlined />}
      description="HELP"
      shape="square"
      style={{ insetInlineEnd: 164 }}
    />
  </>
);

export default App;

```

## 含有气泡卡片的悬浮按钮

```

import React from 'react';
import { FloatButton } from 'antd';

```

```
const App: React.FC = () => <FloatButton tooltip={<div>Documents</div>} />;

export default App;
```

## 浮动按钮组

```
import React from 'react';
import { QuestionCircleOutlined, SyncOutlined } from '@ant-design/icons';
import { FloatButton } from 'antd';

const App: React.FC = () => (
  <>
    <FloatButton.Group shape="circle" style={{ insetInlineEnd: 24 }}>
      <FloatButton icon={<QuestionCircleOutlined />} />
      <FloatButton />
      <FloatButton.BackTop visibilityHeight={0} />
    </FloatButton.Group>
    <FloatButton.Group shape="square" style={{ insetInlineEnd: 94 }}>
      <FloatButton icon={<QuestionCircleOutlined />} />
      <FloatButton />
      <FloatButton icon={<SyncOutlined />} />
      <FloatButton.BackTop visibilityHeight={0} />
    </FloatButton.Group>
  </>
);

export default App;
```

## 菜单模式

```
import React from 'react';
import { CommentOutlined, CustomerServiceOutlined } from '@ant-
design/icons';
import { FloatButton } from 'antd';

const App: React.FC = () => (
  <>
    <FloatButton.Group
      trigger="click"
      type="primary"
      style={{ insetInlineEnd: 24 }}
      icon={<CustomerServiceOutlined />}
    >
      <FloatButton />
    </FloatButton.Group>
  </>
);
```

```

      <FloatButton icon={<CommentOutlined />} />
    </FloatButton.Group>
    <FloatButton.Group
      trigger="hover"
      type="primary"
      style={{ insetInlineEnd: 94 }}
      icon={<CustomerServiceOutlined />}
    >
      <FloatButton />
      <FloatButton icon={<CommentOutlined />} />
    </FloatButton.Group>
  </>
);

export default App;

```

## 受控模式

```

import React, { useState } from 'react';
import { CommentOutlined, CustomerServiceOutlined } from '@ant-
design/icons';
import { FloatButton, Switch } from 'antd';

const App: React.FC = () => {
  const [open, setOpen] = useState<boolean>(true);
  return (
    <>
      <Switch onChange={setOpen} checked={open} style={{ margin: 16 }} />
      <FloatButton.Group
        open={open}
        trigger="click"
        style={{ insetInlineEnd: 24 }}
        icon={<CustomerServiceOutlined />}
      >
        <FloatButton />
        <FloatButton />
        <FloatButton icon={<CommentOutlined />} />
      </FloatButton.Group>
      <FloatButton.Group
        open={open}
        shape="square"
        trigger="click"
        style={{ insetInlineEnd: 88 }}
        icon={<CustomerServiceOutlined />}
      >
        <FloatButton />

```

```

        <FloatButton />
        <FloatButton icon={<CommentOutlined />} />
      </FloatButton.Group>
    </>
  );
};

export default App;

```

## 弹出方向

v5.21.0

```

import React from 'react';
import {
  CommentOutlined,
  DownOutlined,
  LeftOutlined,
  RightOutlined,
  UpOutlined,
} from '@ant-design/icons';
import { Flex, FloatButton } from 'antd';

const BOX_SIZE = 100;
const BUTTON_SIZE = 40;

const wrapperStyle: React.CSSProperties = {
  width: '100%',
  height: '100vh',
  overflow: 'hidden',
  position: 'relative',
};

const boxStyle: React.CSSProperties = {
  width: BOX_SIZE,
  height: BOX_SIZE,
  position: 'relative',
};

const insetInlineEnd: React.CSSProperties['insetInlineEnd'][] = [
  (BOX_SIZE - BUTTON_SIZE) / 2,
  -(BUTTON_SIZE / 2),
  (BOX_SIZE - BUTTON_SIZE) / 2,
  BOX_SIZE - BUTTON_SIZE / 2,
];

```

```

const bottom: React.CSSProperties['bottom'][] = [
  BOX_SIZE - BUTTON_SIZE / 2,
  (BOX_SIZE - BUTTON_SIZE) / 2,
  -BUTTON_SIZE / 2,
  (BOX_SIZE - BUTTON_SIZE) / 2,
];

const icons = [
  <UpOutlined key="up" />,
  <RightOutlined key="right" />,
  <DownOutlined key="down" />,
  <LeftOutlined key="left" />,
];

const App: React.FC = () => (
  <Flex justify="space-around" align="center" style={wrapperStyle}>
    <div style={boxStyle}>
      {[['top', 'right', 'bottom', 'left'] as const}.map((placement, i) =>
        {
          const style: React.CSSProperties = {
            position: 'absolute',
            insetInlineEnd: insetInlineEnd[i],
            bottom: bottom[i],
          };
          return (
            <FloatButton.Group
              key={placement}
              trigger="click"
              placement={placement}
              style={style}
              icon={icons[i]}
            >
              <FloatButton />
              <FloatButton icon={<CommentOutlined />} />
            </FloatButton.Group>
          );
        })
    </div>
  </Flex>
);

export default App;

```

[回到顶部](#)

```
import React from 'react';
import { FloatButton } from 'antd';

const App: React.FC = () => (
  <div style={{ height: '300vh', padding: 10 }}>
    <div>Scroll to bottom</div>
    <div>Scroll to bottom</div>
    <div>Scroll to bottom</div>
    <div>Scroll to bottom</div>
    <div>Scroll to bottom</div>
    <div>Scroll to bottom</div>
    <div>Scroll to bottom</div>
    <FloatButton.BackTop />
  </div>
);

export default App;
```

## 徽标数

```
import React from 'react';
import { QuestionCircleOutlined } from '@ant-design/icons';
import { FloatButton } from 'antd';

const App: React.FC = () => (
  <>
    <FloatButton shape="circle" style={{ insetInlineEnd: 24 + 70 + 70 }}
      badge={{ dot: true }} />
    <FloatButton.Group shape="circle" style={{ insetInlineEnd: 24 + 70 }}>
      <FloatButton
        href="https://ant.design/index-cn"
        tooltip={<div>custom badge color</div>}
        badge={{ count: 5, color: 'blue' }}
      />
      <FloatButton badge={{ count: 5 }} />
    </FloatButton.Group>
    <FloatButton.Group shape="circle">
      <FloatButton badge={{ count: 12 }} icon={<QuestionCircleOutlined />}
    />
    <FloatButton badge={{ count: 123, overflowCount: 999 }} />
    <FloatButton.BackTop visibilityHeight={0} />
  </FloatButton.Group>
</>
);
```

```
export default App;
```

## 调试小圆点使用

Debug

```
import React, { useState } from 'react';
import { ConfigProvider, FloatButton, Slider } from 'antd';
import type { ConfigProviderProps, GetProp } from 'antd';

type AliasToken = GetProp<ConfigProviderProps, 'theme'>['token'];

const App: React.FC = () => {
  const [radius, setRadius] = useState<number>(0);

  const token: Partial<AliasToken> = {
    borderRadius: radius,
  };

  return (
    <>
      <Slider min={0} max={20} style={{ margin: 16 }} onChange={setRadius} />
      <ConfigProvider theme={{ token }}>
        <FloatButton shape="square" badge={{ dot: true }} />
      </ConfigProvider>
    </>
  );
};

export default App;
```

## **`_InternalPanelDoNotUseOrYouWillBeFired`**

Debug

```
import React from 'react';
import { CustomerServiceOutlined, QuestionCircleOutlined, SyncOutlined }
from '@ant-design/icons';
import { FloatButton } from 'antd';

/** Test usage. Do not use in your production. */
const { _InternalPanelDoNotUseOrYouWillBeFired: InternalFloatButton } =
FloatButton;
```



```
const App: React.FC = () => (  
  <div style={{ display: 'flex', columnGap: 16, alignItems: 'center' }}>  
    <InternalFloatButton backTop />  
    <InternalFloatButton icon={<CustomerServiceOutlined />} />  
    <InternalFloatButton  
      icon={<QuestionCircleOutlined />}  
      description="HELP"  
      shape="square"  
      type="primary"  
    />  
    <InternalFloatButton  
      shape="square"  
      items={[  
        { icon: <QuestionCircleOutlined /> },  
        { icon: <CustomerServiceOutlined /> },  
        { icon: <SyncOutlined /> },  
      ]}  
    />  
    <InternalFloatButton  
      open  
      icon={<CustomerServiceOutlined />}  
      trigger="click"  
      items={[  
        { icon: <QuestionCircleOutlined /> },  
        { icon: <CustomerServiceOutlined /> },  
        { icon: <SyncOutlined /> },  
      ]}  
    />  
  </div>  
)  
);  
  
export default App;
```

## API

通用属性参考: [通用属性](#)

自 `antd@5.0.0` 版本开始提供该组件。

### 共同的 API

参数	说明	类型	默认值	版本
icon	自定义图标	ReactNode	-	
description	文字及其它内容	ReactNode	-	

tooltip	气泡卡片的内容	ReactNode   () => ReactNode	-	
type	设置按钮类型	default   primary	default	
shape	设置按钮形状	circle   square	circle	
onClick	点击按钮时的回调	(event) => void	-	
href	点击跳转的地址，指定此属性 button 的行为和 a 链接一致	string	-	
target	相当于 a 标签的 target 属性，href 存在时生效	string	-	
htmlType	设置 button 原生的 type 值，可选值请参考 <a href="#">HTML 标准</a>	submit   reset   button	button	5.21.0
badge	带徽标数字的悬浮按钮（不支持 status 以及相关属性）	<a href="#">BadgeProps</a>	-	5.4.0

**FloatButton.Group**

参数	说明	类型	默认值	版本
shape	设置包含的 FloatButton 按钮形状	circle   square	circle	
trigger	触发方式（有触发方式为菜单模式）	click   hover	-	
open	受控展开，需配合 trigger 一起使用	boolean	-	
closeIcon	自定义关闭按钮	React.ReactNode	<CloseOutlined />	
placement	自定义菜单弹出位置	top   left   right   bottom	top	5.21.0
onOpenChange	展开收起时的回调，需配合 trigger 一起使用	(open: boolean) => void	-	
onClick	点击按钮时的回调（仅在菜单模式中有效）	(event) => void	-	5.3.0

**FloatButton.BackTop**

参数	说明	类型	默认值	版本
duration	回到顶部所需时间（ms）	number	450	
target	设置需要监听其滚动事件的元素	() => HTMLElement	() => window	
visibilityHeight	滚动高度达到此参数值才出现 BackTop	number	400	
onClick	点击按钮的回调函数	() => void	-	

主题变量（Design Token）