

何时使用 {#when-to-use}

- 对评价进行展示。
- 对事物进行快速的评级操作。

代码演示

基本

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

const App: React.FC = () => <Rate />;

export default App;
```

半星

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

const App: React.FC = () => <Rate allowHalf defaultValue={2.5} />;

export default App;
```

文案展现

```
import React, { useState } from 'react';
import { Flex, Rate } from 'antd';

const desc = ['terrible', 'bad', 'normal', 'good', 'wonderful'];

const App: React.FC = () => {
  const [value, setValue] = useState(3);
  return (
    <Flex gap="middle" vertical>
      <Rate tooltips={desc} onChange={setValue} value={value} />
      {value ? <span>{desc[value - 1]}</span> : null}
    </Flex>
  );
};

export default App;
```

只读

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

const App: React.FC = () => <Rate disabled defaultValue={2} />;

export default App;
```

清除

```
import React from 'react';
import { Flex, Rate } from 'antd';

const App: React.FC = () => (
  <Flex gap="middle" vertical>
    <Flex gap="middle">
      <Rate defaultValue={3} />
      <span>allowClear: true</span>
    </Flex>
    <Flex gap="middle">
      <Rate defaultValue={3} allowClear={false} />
      <span>allowClear: false</span>
    </Flex>
  </Flex>
);

export default App;
```

其他字符

```
import React from 'react';
import { HeartOutlined } from '@ant-design/icons';
import { Flex, Rate } from 'antd';

const App: React.FC = () => (
  <Flex vertical gap="middle">
    <Rate character={<HeartOutlined />} allowHalf />
    <Rate character="A" allowHalf style={{ fontSize: 36 }} />
    <Rate character="好" allowHalf />
  </Flex>
);

export default App;
```

自定义字符

```

import React from 'react';
import { FrownOutlined, MehOutlined, SmileOutlined } from '@ant-
design/icons';
import { Flex, Rate } from 'antd';

const customIcons: Record<number, React.ReactNode> = {
  1: <FrownOutlined />,
  2: <FrownOutlined />,
  3: <MehOutlined />,
  4: <SmileOutlined />,
  5: <SmileOutlined />,
};

const App: React.FC = () => (
  <Flex gap="middle" vertical>
    <Rate defaultValue={2} character={({ index = 0 }) => index + 1} />
    <Rate defaultValue={3} character={({ index = 0 }) => customIcons[index
+ 1]} />
  </Flex>
);

export default App;

```

组件 Token

Debug

```

import React from 'react';
import { ConfigProvider, Rate } from 'antd';

/** Test usage. Do not use in your production. */
export default () => (
  <ConfigProvider
    theme={{
      components: {
        Rate: {
          starColor: 'blue',
          starSize: 40,
          starHoverScale: 'scale(2)',
          starBg: 'red',
        },
      },
    }}
  >
    <Rate defaultValue={2.5} />

```

```
    </ConfigProvider>
  );
```

API

通用属性参考：[通用属性](#)

属性	说明	类型	默认值	版本
allowClear	是否允许再次点击后清除	boolean	true	
allowHalf	是否允许半选	boolean	false	
autoFocus	自动获取焦点	boolean	false	
character	自定义字符	ReactNode (RateProps) => ReactNode	<StarFilled />	function(): 4.4.0
className	自定义样式类名	string	-	
count	star 总数	number	5	
defaultValue	默认值	number	0	
disabled	只读，无法进行交互	boolean	false	
keyboard	支持使用键盘操作	boolean	true	5.18.0
style	自定义样式对象	CSSProperties	-	
tooltips	自定义每项的提示信息	string[]	-	
value	当前数，受控值	number	-	
onBlur	失去焦点时的回调	function()	-	
onChange	选择时的回调	function(value: number)	-	
onFocus	获取焦点时的回调	function()	-	
onHoverChange	鼠标经过时数值变化的回调	function(value: number)	-	

onKeyDown	按键回调	function(event)	-	
-----------	------	-----------------	---	--

方法

名称	描述
blur()	移除焦点
focus()	获取焦点

主题变量（Design Token）