

When To Use

- Show evaluation.
- A quick rating operation on something.

Examples

Basic

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

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

export default App;
```

Half star

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

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

export default App;
```

Show copywriting

```
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;
```

Read only

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

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

export default App;
```

Clear star

```
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;
```

Other Character

```
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;
```

Customize character

```

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;

```

Component 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

Common props ref: [Common props](#)

Property	Description	type	Default	Version
allowClear	Whether to allow clear when click again	boolean	true	
allowHalf	Whether to allow semi selection	boolean	false	
autoFocus	If get focus when component mounted	boolean	false	
character	The custom character of rate	ReactNode (RateProps) => ReactNode	<StarFilled />	function(): 4.4.0
className	The custom class name of rate	string	-	
count	Star count	number	5	
defaultValue	The default value	number	0	
disabled	If read only, unable to interact	boolean	false	
keyboard	Support keyboard operation	boolean	true	5.18.0
style	The custom style object of rate	CSSProperties	-	
tooltips	Customize tooltip by each character	string[]	-	
value	The current value	number	-	
onBlur	Callback when component lose focus	function()	-	
onChange	Callback when select value	function(value: number)	-	

onFocus	Callback when component get focus	function()	-	
onHoverChange	Callback when hover item	function(value: number)	-	
onKeyDown	Callback when keydown on component	function(event)	-	

Methods

Name	Description
blur()	Remove focus
focus()	Get focus

Design Token