

When To Use

- If you need to represent the switching between two states or on-off state.
- The difference between `Switch` and `Checkbox` is that `Switch` will trigger a state change directly when you toggle it, while `Checkbox` is generally used for state marking, which should work in conjunction with submit operation.

Examples

Basic

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

const onChange = (checked: boolean) => {
  console.log(`switch to ${checked}`);
};

const App: React.FC = () => <Switch defaultChecked onChange={onChange} />;

export default App;
```

Disabled

```
import React, { useState } from 'react';
import { Button, Space, Switch } from 'antd';

const App: React.FC = () => {
  const [disabled, setDisabled] = useState(true);

  const toggle = () => {
    setDisabled(!disabled);
  };

  return (
    <Space direction="vertical">
      <Switch disabled={disabled} defaultChecked />
      <Button type="primary" onClick={toggle}>
        Toggle disabled
      </Button>
    </Space>
  );
};

export default App;
```

Text & icon

```
import React from 'react';
import { CheckOutlined, CloseOutlined } from '@ant-design/icons';
import { Space, Switch } from 'antd';

const App: React.FC = () => (
  <Space direction="vertical">
    <Switch checkedChildren="开启" uncheckedChildren="关闭" defaultChecked
  />
    <Switch checkedChildren="1" uncheckedChildren="0" />
    <Switch
      checkedChildren={<CheckOutlined />}
      uncheckedChildren={<CloseOutlined />}
      defaultChecked
    />
  </Space>
);

export default App;
```

Two sizes

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

const App: React.FC = () => (
  <>
    <Switch defaultChecked />
    <br />
    <Switch size="small" defaultChecked />
  </>
);

export default App;
```

Loading

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

const App: React.FC = () => (
  <>
    <Switch loading defaultChecked />
  </>
);
```

```

    <br />
    <Switch size="small" loading />
  </>
);

export default App;

```

Custom component token

Debug

```

import React from 'react';
import { ConfigProvider, Space, Switch } from 'antd';

const App: React.FC = () => (
  <ConfigProvider
    theme={{
      components: {
        Switch: {
          trackHeight: 14,
          trackMinWidth: 32,
          // opacityLoading: 0.1,
          colorPrimary: 'rgb(25, 118, 210, 0.5)',
          trackPadding: -3,
          handleSize: 20,
          handleBg: 'rgb(25, 118, 210)',
          handleShadow:
            'rgba(0, 0, 0, 0.2) 0px 2px 1px -1px, rgba(0, 0, 0, 0.14) 0px
1px 1px 0px, rgba(0, 0, 0, 0.12) 0px 1px 3px 0px',
          // innerMinMargin: 4,
          // innerMaxMargin: 8,
        },
      },
    }}
  >
    <Space>
      <Switch defaultChecked />
    </Space>
  </ConfigProvider>
);

export default App;

```

API

Common props ref: [Common props](#)

Property	Description	Type	Default	Version
autoFocus	Whether get focus when component mounted	boolean	false	
checked	Determine whether the Switch is checked	boolean	false	
checkedChildren	The content to be shown when the state is checked	ReactNode	-	
className	The additional class to Switch	string	-	
defaultChecked	Whether to set the initial state	boolean	false	
defaultValue	Alias for defaultChecked	boolean	-	5.12.0
disabled	Disable switch	boolean	false	
loading	Loading state of switch	boolean	false	
size	The size of the Switch, options: default small	string	default	
uncheckedChildren	The content to be shown when the state is unchecked	ReactNode	-	
value	Alias for checked	boolean	-	5.12.0
onChange	Trigger when the checked state is changing	function(checked: boolean, event: Event)	-	
onClick	Trigger when clicked	function(checked: boolean, event: Event)	-	

Methods

Name	Description
blur()	Remove focus

focus()	Get focus
---------	-----------

Design Token

FAQ

Why not work in Form.Item?

Form.Item default bind value to `value` property, but Switch value property is `checked` . You can use `valuePropName` to change bind property.

```
<Form.Item name="fieldA" valuePropName="checked">
  <Switch />
</Form.Item>
```