自 `antd@4.20.0` 版本开始提供该组件。

何时使用 {#when-to-use}

- 用于展示多个选项并允许用户选择其中单个选项；
- 当切换选中选项时，关联区域的内容会发生变化。

代码演示

基本

```tsx
import React from 'react';
import { Segmented } from 'antd';

const Demo: React.FC = () => (
  <Segmented<string>
    options={['Daily', 'Weekly', 'Monthly', 'Quarterly', 'Yearly']}
    onChange={(value) => {
      console.log(value); // string
    }}
  />
);

export default Demo;
```

垂直方向

v5.21.0

```tsx
import React from 'react';
import { AppstoreOutlined, BarsOutlined } from '@ant-design/icons';
import { Segmented } from 'antd';

const Demo: React.FC = () => (
  <Segmented
    vertical
    options={[
      { value: 'List', icon: <BarsOutlined /> },
      { value: 'Kanban', icon: <AppstoreOutlined /> },
    ]}
  />
);

export default Demo;
```

Block 分段选择器

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

const Demo: React.FC = () => (
  <Segmented options={[123, 456, 'longtext-longtext-longtext-longtext']}
block />
);

export default Demo;
```

胶囊形状

v5.24.0

```
import React, { useState } from 'react';
import { MoonOutlined, SunOutlined } from '@ant-design/icons';
import { Flex, Segmented } from 'antd';
import type { SizeType } from '../../config-provider/SizeContext';

const Demo: React.FC = () => {
  const [size, setSize] = useState<SizeType>('middle');
  return (
    <Flex gap="small" align="flex-start" vertical>
      <Segmented
        options={['small', 'middle', 'large']}
        value={size}
        onChange={(value) => setSize(value as SizeType)}
      />
      <Segmented
        size={size}
        shape="round"
        options={[
          { value: 'light', icon: <SunOutlined /> },
          { value: 'dark', icon: <MoonOutlined /> },
        ]}
      />
    </Flex>
  );
};

export default Demo;
```

不可用

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

const App: React.FC = () => (
  <Flex gap="small" align="flex-start" vertical>
    <Segmented options={['Map', 'Transit', 'Satellite']} disabled />
    <Segmented
      options={[
        'Daily',
        { label: 'Weekly', value: 'Weekly', disabled: true },
        'Monthly',
        { label: 'Quarterly', value: 'Quarterly', disabled: true },
        'Yearly',
      ]}
    />
  </Flex>
);

export default App;
```

受控模式

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

const Demo: React.FC = () => {
  const [value, setValue] = useState<string | number>('Map');

  return <Segmented options={['Map', 'Transit', 'Satellite']} value={value}
onChange={setValue} />;
};

export default Demo;
```

自定义渲染

```
import React from 'react';
import { UserOutlined } from '@ant-design/icons';
import { Avatar, Flex, Segmented } from 'antd';

const App: React.FC = () => (
  <Flex gap="small" align="flex-start" vertical>
    <Segmented
      options={[
        {
          label: (
            <div style={{ padding: 4 }}>
              <Avatar src="https://api.dicebear.com/7.x/miniavs/svg?seed=8"
/>
              <div>User 1</div>
            </div>
          ),
          value: 'user1',
        },
        {
          label: (
            <div style={{ padding: 4 }}>
              <Avatar style={{ backgroundColor: '#f56a00' }}>K</Avatar>
              <div>User 2</div>
            </div>
          ),
          value: 'user2',
        },
        {
          label: (
            <div style={{ padding: 4 }}>
              <Avatar style={{ backgroundColor: '#87d068' }} icon=
{<UserOutlined />} />
              <div>User 3</div>
            </div>
          ),
          value: 'user3',
        },
      ]}
    />
    <Segmented
      options={[
        {
          label: (
            <div style={{ padding: 4 }}>
              <div>Spring</div>
              <div>Jan-Mar</div>
            </div>
          ),
          value: 'spring',
        },
        {
          label: (
            <div style={{ padding: 4 }}>
              <div>Summer</div>
              <div>Apr-Jun</div>
            </div>
          ),
          value: 'summer',
        },
        {
          label: (
            <div style={{ padding: 4 }}>
              <div>Autumn</div>
              <div>Jul-Sept</div>
            </div>
          ),
          value: 'autumn',
        },
        {
          label: (
            <div style={{ padding: 4 }}>
              <div>Winter</div>
              <div>Oct-Dec</div>
            </div>
          ),
          value: 'winter',
        },
      ]}
    />
  </Flex>
);

export default App;
```

动态数据

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

const Demo: React.FC = () => {
  const [options, setOptions] = useState(['Daily', 'Weekly', 'Monthly']);
  const [moreLoaded, setMoreLoaded] = useState(false);

  const handleLoadOptions = () => {
    setOptions((prev) => [...prev, 'Quarterly', 'Yearly']);
    setMoreLoaded(true);
  };

  return (
    <Flex gap="small" align="flex-start" vertical>
```

```
      <Segmented options={options} />
      <Button type="primary" disabled={moreLoaded} onClick=
{handleLoadOptions}>
        Load more options
      </Button>
    </Flex>
  );
};

export default Demo;
```

三种大小

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

const App: React.FC = () => (
  <Flex gap="small" align="flex-start" vertical>
    <Segmented size="large" options={['Daily', 'Weekly', 'Monthly',
'Quarterly', 'Yearly']} />
    <Segmented options={['Daily', 'Weekly', 'Monthly', 'Quarterly',
'Yearly']} />
    <Segmented size="small" options={['Daily', 'Weekly', 'Monthly',
'Quarterly', 'Yearly']} />
  </Flex>
);

export default App;
```

设置图标

```
import React from 'react';
import { AppstoreOutlined, BarsOutlined } from '@ant-design/icons';
import { Segmented } from 'antd';

const Demo: React.FC = () => (
  <Segmented
    options={[
      { label: 'List', value: 'List', icon: <BarsOutlined /> },
      { label: 'Kanban', value: 'Kanban', icon: <AppstoreOutlined /> },
    ]}
  />
);
```

```
export default Demo;
```

只设置图标

```
import React from 'react';
import { AppstoreOutlined, BarsOutlined } from '@ant-design/icons';
import { Segmented } from 'antd';

const Demo: React.FC = () => (
  <Segmented
    options={[
      { value: 'List', icon: <BarsOutlined /> },
      { value: 'Kanban', icon: <AppstoreOutlined /> },
    ]}
  />
);

export default Demo;
```

配合 name 使用

v5.23.0

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

const Demo: React.FC = () => (
  <Segmented<string> options={['Daily', 'Weekly', 'Monthly', 'Quarterly',
'Yearly']} name="group" />
);

export default Demo;
```

受控同步模式

Debug

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

const Demo: React.FC = () => {
  const [foo, setFoo] = useState<string | number>('AND');
  return (
    <>
      <Segmented value={foo} options={['AND', 'OR', 'NOT']} onChange=
{setFoo} />
      &nbsp;&nbsp;
      <Segmented value={foo} options={['AND', 'OR', 'NOT']} onChange=
{setFoo} />
    </>
  );
};

export default Demo;
```

统一高度

Debug

```
import React from 'react';
import { Button, Flex, Input, Segmented, Select } from 'antd';

const App: React.FC = () => (
  <Flex gap="small" vertical>
    <div>
      <Segmented
        size="large"
        style={{ marginInlineEnd: 6 }}
        options={['Daily', 'Weekly', 'Monthly']}
      />
      <Button type="primary" size="large">
        Button
      </Button>
    </div>
    <div>
      <Segmented
        size="middle"
        style={{ marginInlineEnd: 6 }}
        options={['Daily', 'Weekly', 'Monthly']}
      />
      <Input placeholder="default size" style={{ width: 150 }} />
    </div>
    <div>
      <Segmented
        size="small"
        style={{ marginInlineEnd: 6 }}
        options={['Daily', 'Weekly', 'Monthly']}
      />
      <Select size="small" defaultValue="lucy" style={{ width: 150 }}>
        <Select.Option value="lucy">Lucy</Select.Option>
      </Select>
    </div>
  </Flex>
);

export default App;
```

自定义组件 Token

Debug

```
import React from 'react';
import { ConfigProvider, Segmented } from 'antd';

const Demo: React.FC = () => (
  <ConfigProvider
    theme={{
      components: {
        Segmented: {
          itemColor: '#222',
          itemHoverColor: '#333',
          itemHoverBg: 'rgba(0, 0, 0, 0.06)',
          itemSelectedBg: '#aaa',
          itemActiveBg: '#ccc',
          itemSelectedColor: '#fff',
        },
      },
    }}
  >
    <Segmented options={['Daily', 'Weekly', 'Monthly', 'Quarterly', 'Yearly']} />
  </ConfigProvider>
);

export default Demo;
```

API

通用属性参考：通用属性

> *自 `antd@4.20.0` 版本开始提供该组件。*

Segmented

参数	说明	类型	默认值	版本
block	将宽度调整为父元素宽度的选项	boolean	false	

defaultValue	默认选中的值	string | number		
disabled	是否禁用	boolean	false	
onChange	选项变化时的回调函数	function(value: string | number)		
options	数据化配置选项内容	string[] | number[] | SegmentedItemType[]	[]	
size	控件尺寸	`large` | `middle` | `small`	`middle`	
vertical	排列方向	boolean	`false`	5.21.0
value	当前选中的值	string | number		
shape	形状	`default` | `round`	`default`	5.24.0
name	Segmented 下所有 `input[type="radio"]` 的 `name` 属性。若未设置，则将回退到随机生成的名称	string		5.23.0

SegmentedItemType

属性	描述	类型	默认值	版本
label	分段项的显示文本	ReactNode	-	
value	分段项的值	string | number	-	
icon	分段项的显示图标	ReactNode	-	
disabled	分段项的禁用状态	boolean	false	
className	自定义类名	string	-	

主题变量（Design Token）