

何时使用 {#when-to-use}

- 可以水平或垂直地分隔区域。
- 当需要自由拖拽调整各区域大小。
- 当需要指定区域的最大最小宽高时。

代码演示

基本用法

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

const Desc: React.FC<Readonly<{ text?: string | number }>> = (props) => (
  <Flex justify="center" align="center" style={{ height: '100%' }}>
    <Typography.Title type="secondary" level={5} style={{ whiteSpace:
'nowrap' }}>
      {props.text}
    </Typography.Title>
  </Flex>
);

const App: React.FC = () => (
  <Splitter style={{ height: 200, boxShadow: '0 0 10px rgba(0, 0, 0, 0.1)'
}}>
    <Splitter.Panel defaultSize="40%" min="20%" max="70%">
      <Desc text="First" />
    </Splitter.Panel>
    <Splitter.Panel>
      <Desc text="Second" />
    </Splitter.Panel>
  </Splitter>
);

export default App;
```

受控模式

```
import React from 'react';
import { Button, Flex, Splitter, Switch, Typography } from 'antd';

const Desc: React.FC<Readonly<{ text?: string | number }>> = (props) => (
  <Flex justify="center" align="center" style={{ height: '100%' }}>
    <Typography.Title type="secondary" level={5} style={{ whiteSpace:
'nowrap' }}>
      {props.text}
    </Typography.Title>
  </Flex>
);
```

```

    </Typography.Title>
  </Flex>
);

const App: React.FC = () => {
  const [sizes, setSizes] = React.useState<(number | string) []>(['50%', '50%']);
  const [enabled, setEnabled] = React.useState(true);
  return (
    <Flex vertical gap="middle">
      <Splitter
        onResize={setSizes}
        style={{ height: 200, boxShadow: '0 0 10px rgba(0, 0, 0, 0.1)' }}
      >
        <Splitter.Panel size={sizes[0]} resizable={enabled}>
          <Desc text="First" />
        </Splitter.Panel>
        <Splitter.Panel size={sizes[1]}>
          <Desc text="Second" />
        </Splitter.Panel>
      </Splitter>
      <Flex gap="middle" justify="space-between">
        <Switch
          value={enabled}
          onChange={() => setEnabled(!enabled)}
          checkedChildren="Enabled"
          uncheckedChildren="Disabled"
        />
        <Button onClick={() => setSizes(['50%', '50%'])}>Reset</Button>
      </Flex>
    </Flex>
  );
};

export default App;

```

垂直方向

```

import React from 'react';
import { Flex, Splitter, Typography } from 'antd';

const Desc: React.FC<Readonly<{ text?: string | number }>> = (props) => (
  <Flex justify="center" align="center" style={{ height: '100%' }}>
    <Typography.Title type="secondary" level={5} style={{ whiteSpace: 'nowrap' }}>
      {props.text}
    </Typography.Title>
  </Flex>
);

```

```

    </Typography.Title>
  </Flex>
);

const App: React.FC = () => (
  <Splitter layout="vertical" style={{ height: 300, boxShadow: '0 0 10px
  rgba(0, 0, 0, 0.1)' }}>
    <Splitter.Panel>
      <Desc text="First" />
    </Splitter.Panel>
    <Splitter.Panel>
      <Desc text="Second" />
    </Splitter.Panel>
  </Splitter>
);

export default App;

```

可折叠

```

import React from 'react';
import { Flex, Splitter, Typography } from 'antd';
import type { SplitterProps } from 'antd';

const Desc: React.FC<Readonly<{ text?: string | number }>> = (props) => (
  <Flex justify="center" align="center" style={{ height: '100%' }}>
    <Typography.Title type="secondary" level={5} style={{ whiteSpace:
    'nowrap' }}>
      {props.text}
    </Typography.Title>
  </Flex>
);

const CustomSplitter: React.FC<Readonly<SplitterProps>> = ({ style,
...restProps }) => (
  <Splitter style={{ boxShadow: '0 0 10px rgba(0, 0, 0, 0.1)', ...style }}
  {...restProps}>
    <Splitter.Panel collapsible min="20%">
      <Desc text="First" />
    </Splitter.Panel>
    <Splitter.Panel collapsible>
      <Desc text="Second" />
    </Splitter.Panel>
  </Splitter>
);

```

```
const App: React.FC = () => (
  <Flex gap="middle" vertical>
    <CustomSplitter style={{ height: 200 }} />
    <CustomSplitter style={{ height: 300 }} layout="vertical" />
  </Flex>
);

export default App;
```

多面板

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

const Desc: React.FC<Readonly<{ text?: string | number }>> = (props) => (
  <Flex justify="center" align="center" style={{ height: '100%' }}>
    <Typography.Title type="secondary" level={5} style={{ whiteSpace:
'nowrap' }}>
      Panel {props.text}
    </Typography.Title>
  </Flex>
);

const App: React.FC = () => (
  <Splitter style={{ height: 200, boxShadow: '0 0 10px rgba(0, 0, 0, 0.1)'
}}>
    <Splitter.Panel collapsible>
      <Desc text={1} />
    </Splitter.Panel>
    <Splitter.Panel collapsible={{ start: true }}>
      <Desc text={2} />
    </Splitter.Panel>
    <Splitter.Panel>
      <Desc text={3} />
    </Splitter.Panel>
  </Splitter>
);

export default App;
```

复杂组合

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

```

const Desc: React.FC<Readonly<{ text?: string | number }>> = (props) => (
  <Flex justify="center" align="center" style={{ height: '100%' }}>
    <Typography.Title type="secondary" level={5} style={{ whiteSpace:
'nowrap' }}>
      {props.text}
    </Typography.Title>
  </Flex>
);

const App: React.FC = () => (
  <Splitter style={{ height: 300, boxShadow: '0 0 10px rgba(0, 0, 0, 0.1)'
}}>
    <Splitter.Panel collapsible>
      <Desc text="Left" />
    </Splitter.Panel>
    <Splitter.Panel>
      <Splitter layout="vertical">
        <Splitter.Panel>
          <Desc text="Top" />
        </Splitter.Panel>
        <Splitter.Panel>
          <Desc text="Bottom" />
        </Splitter.Panel>
      </Splitter>
    </Splitter.Panel>
  </Splitter>
);

export default App;

```

标签页中嵌套

Debug

```

import React from 'react';
import { Flex, Splitter, Tabs, Typography } from 'antd';

const Desc: React.FC<Readonly<{ text?: string | number }>> = (props) => (
  <Flex justify="center" align="center" style={{ height: '100%' }}>
    <Typography.Title type="secondary" level={5} style={{ whiteSpace:
'nowrap' }}>
      {props.text}
    </Typography.Title>
  </Flex>
);

```

```

const App: React.FC = () => {
  const SplitterContent = (
    <Splitter
      style={{
        height: 200,
        boxShadow: '0 0 10px rgba(0, 0, 0, 0.1)',
      }}
    >
      <Splitter.Panel collapsible>
        <Desc text={1} />
      </Splitter.Panel>
      <Splitter.Panel
        collapsible={{
          start: true,
        }}
      >
        <Desc text={2} />
      </Splitter.Panel>
      <Splitter.Panel>
        <Desc text={3} />
      </Splitter.Panel>
    </Splitter>
  );
  return (
    <Tabs
      defaultActiveKey="1"
      items={[
        {
          key: '1',
          label: 'General',
          children: 'Content of Tab Pane 1',
        },
        {
          key: '2',
          label: 'Splitter Tab',
          children: SplitterContent,
        },
      ]}
    />
  );
};

export default App;

```

延迟渲染模式

v5.23.0

```
import React from 'react';
import { Flex, Space, Splitter, Typography } from 'antd';

const Desc: React.FC<Readonly<{ text?: string | number }>> = (props) => (
  <Flex justify="center" align="center" style={{ height: '100%' }}>
    <Typography.Title type="secondary" level={5} style={{ whiteSpace:
'nowrap' }}>
      {props.text}
    </Typography.Title>
  </Flex>
);

const App: React.FC = () => (
  <Space direction="vertical" style={{ width: '100%' }}>
    <Splitter lazy style={{ height: 200, boxShadow: '0 0 10px rgba(0, 0, 0,
0.1)' }}>
      <Splitter.Panel defaultSize="40%" min="20%" max="70%">
        <Desc text="First" />
      </Splitter.Panel>
      <Splitter.Panel>
        <Desc text="Second" />
      </Splitter.Panel>
    </Splitter>
    <Splitter
      lazy
      layout="vertical"
      style={{ height: 200, boxShadow: '0 0 10px rgba(0, 0, 0, 0.1)' }}
    >
      <Splitter.Panel defaultSize="40%" min="30%" max="70%">
        <Desc text="First" />
      </Splitter.Panel>
      <Splitter.Panel>
        <Desc text="Second" />
      </Splitter.Panel>
    </Splitter>
  </Space>
);

export default App;
```

调试

Debug

```

import React from 'react';
import { Flex, Splitter, Typography } from 'antd';

const Desc: React.FC<Readonly<{ text?: string | number }>> = (props) => (
  <Flex justify="center" align="center" style={{ height: '100%' }}>
    <Typography.Title type="secondary" level={5} style={{ whiteSpace:
'nowrap' }}>
      Panel {props.text}
    </Typography.Title>
  </Flex>
);

const App: React.FC = () => (
  <Flex vertical gap="middle">
    <Typography.Title level={3}>[true, 0, false]</Typography.Title>
    <Splitter style={{ height: 200, boxShadow: '0 0 10px rgba(0, 0, 0,
0.1)' }}>
      <Splitter.Panel>
        <Desc text={1} />
      </Splitter.Panel>
      <Splitter.Panel defaultSize={0}>
        <Desc text={2} />
      </Splitter.Panel>
      <Splitter.Panel resizable={false}>
        <Desc text={3} />
      </Splitter.Panel>
    </Splitter>
    <Typography.Title level={3}>[false, 0, true]</Typography.Title>
    <Splitter style={{ height: 200, boxShadow: '0 0 10px rgba(0, 0, 0,
0.1)' }}>
      <Splitter.Panel resizable={false}>
        <Desc text={1} />
      </Splitter.Panel>
      <Splitter.Panel defaultSize={0}>
        <Desc text={2} />
      </Splitter.Panel>
      <Splitter.Panel>
        <Desc text={3} />
      </Splitter.Panel>
    </Splitter>
    <Typography.Title level={3}>Start have min & max</Typography.Title>
    <Splitter style={{ height: 200, boxShadow: '0 0 10px rgba(0, 0, 0,
0.1)' }}>
      <Splitter.Panel min={50} max={100}>
        <Desc text={1} />
      </Splitter.Panel>

```



```

    <Splitter.Panel>
      <Desc text={2} />
    </Splitter.Panel>
  </Splitter>
  <Typography.Title level={3}>End have min & max</Typography.Title>
  <Splitter style={{ height: 200, boxShadow: '0 0 10px rgba(0, 0, 0, 0.1)' }}>
    <Splitter.Panel>
      <Desc text={1} />
    </Splitter.Panel>
    <Splitter.Panel min="20%" max="70%">
      <Desc text={2} />
    </Splitter.Panel>
  </Splitter>
</Flex>
);

export default App;
```

API

通用属性参考：[通用属性](#)

Splitter 组件需要通过子元素计算面板大小，因而其子元素仅支持 *Splitter.Panel*。

Splitter

参数	说明	类型	默认值	版本
layout	布局方向	horizontal vertical	horizontal	-
onResizeStart	开始拖拽之前回调	(sizes: number[]) => void	-	-
onResize	面板大小变化回调	(sizes: number[]) => void	-	-
onResizeEnd	拖拽结束回调	(sizes: number[]) => void	-	-
lazy	延迟渲染模式	boolean	false	5.23.0

Panel

参数	说明	类型	默认值	版本
defaultSize	初始面板大小，支持数字 px 或者文字 '百分比%' 类型	number string	-	-

min	最小阈值，支持数字 px 或者文字 '百分比%' 类型	number string	-	-
max	最大阈值，支持数字 px 或者文字 '百分比%' 类型	number string	-	-
size	受控面板大小，支持数字 px 或者文字 '百分比%' 类型	number string	-	-
collapsible	快速折叠	boolean { start?: boolean; end?: boolean }	false	-
resizable	是否开启拖拽伸缩	boolean	true	-

主题变量（Design Token）