When To Use

- Avoid components clinging together and set a unified space.
- Use Space.Compact when child form components are compactly connected and the border is collapsed (After version `antd@4.24.0` Supported).

Difference with Flex component

- Space is used to set the spacing between inline elements. It will add a wrapper element for each child element for inline alignment. Suitable for equidistant arrangement of multiple child elements in rows and columns.
- Flex is used to set the layout of block-level elements. It does not add a wrapper element. Suitable for layout of child elements in vertical or horizontal direction, and provides more flexibility and control.

Examples

Basic Usage

```
import React from 'react';
import { UploadOutlined } from '@ant-design/icons';
import { Button, Popconfirm, Space, Upload } from 'antd';

const App: React.FC = () => (
  <Space>
    Space
    <Button type="primary">Button</Button>
    <Upload>
      <Button icon={<UploadOutlined />}>Click to Upload</Button>
    </Upload>
    <Popconfirm title="Are you sure delete this task?" okText="Yes"
cancelText="No">
      <Button>Confirm</Button>
    </Popconfirm>
  </Space>
);

export default App;
```

Vertical Space

```
import React from 'react';
import { Card, Space } from 'antd';

const App: React.FC = () => (
  <Space direction="vertical" size="middle" style={{ display: 'flex' }}>
    <Card title="Card" size="small">
```

```
      <p>Card content</p>
      <p>Card content</p>
    </Card>
    <Card title="Card" size="small">
      <p>Card content</p>
      <p>Card content</p>
    </Card>
    <Card title="Card" size="small">
      <p>Card content</p>
      <p>Card content</p>
    </Card>
  </Space>
);


export default App;
```

Space Size

```
import React, { useState } from 'react';
import { Button, Radio, Slider, Space } from 'antd';
import type { ConfigProviderProps } from 'antd';

type SizeType = ConfigProviderProps['componentSize'];

const App: React.FC = () => {
  const [size, setSize] = useState<SizeType | [SizeType, SizeType] |
'customize'>('small');
  const [customSize, setCustomSize] = React.useState<number>(0);
  return (
    <>
      <Radio.Group value={size} onChange={(e) => setSize(e.target.value)}>
        {['small', 'middle', 'large', 'customize'].map((item) => (
          <Radio key={item} value={item}>
            {item}
          </Radio>
        ))}
      </Radio.Group>
      <br />
      <br />
      {size === 'customize' && (
        <>
          <Slider value={customSize} onChange={setCustomSize} />
          <br />
        </>
      )}
      <Space size={size !== 'customize' ? size : customSize}>
```

```
        <Button type="primary">Primary</Button>
        <Button>Default</Button>
        <Button type="dashed">Dashed</Button>
        <Button type="link">Link</Button>
      </Space>
    </>
  );
};


export default App;
```

Align

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

const App: React.FC = () => (
  <div className="space-align-container">
    <div className="space-align-block">
      <Space align="center">
        center
        <Button type="primary">Primary</Button>
        <span className="mock-block">Block</span>
      </Space>
    </div>
    <div className="space-align-block">
      <Space align="start">
        start
        <Button type="primary">Primary</Button>
        <span className="mock-block">Block</span>
      </Space>
    </div>
    <div className="space-align-block">
      <Space align="end">
        end
        <Button type="primary">Primary</Button>
        <span className="mock-block">Block</span>
      </Space>
    </div>
    <div className="space-align-block">
      <Space align="baseline">
        baseline
        <Button type="primary">Primary</Button>
        <span className="mock-block">Block</span>
      </Space>
    </div>
```

```
    </div>
);

export default App;
```

Wrap

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

const App: React.FC = () => (
  <Space size={[8, 16]} wrap>
    {Array.from({ length: 20 }).map((_, index) => (
      // eslint-disable-next-line react/no-array-index-key
      <Button key={index}>Button</Button>
    ))}
  </Space>
);

export default App;
```

Split

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

const App: React.FC = () => (
  <Space split={<Divider type="vertical" />}>
    <Typography.Link>Link</Typography.Link>
    <Typography.Link>Link</Typography.Link>
    <Typography.Link>Link</Typography.Link>
  </Space>
);

export default App;
```

Compact Mode for form component

```
import React from 'react';
import { CopyOutlined } from '@ant-design/icons';
import {
  AutoComplete,
  Button,
  Cascader,
```

```jsx
  ColorPicker,
  DatePicker,
  Input,
  InputNumber,
  Select,
  Space,
  TimePicker,
  Tooltip,
  TreeSelect,
} from 'antd';

const { Option } = Select;
const { TreeNode } = TreeSelect;

const App: React.FC = () => (
  <Space direction="vertical">
    <Space.Compact block>
      <Input style={{ width: '20%' }} defaultValue="0571" />
      <Input style={{ width: '30%' }} defaultValue="26888888" />
    </Space.Compact>
    <Space.Compact block size="small">
      <Input style={{ width: 'calc(100% - 200px)' }}
defaultValue="https://ant.design" />
      <Button type="primary">Submit</Button>
    </Space.Compact>
    <Space.Compact block>
      <Input style={{ width: 'calc(100% - 200px)' }}
defaultValue="https://ant.design" />
      <Button type="primary">Submit</Button>
    </Space.Compact>
    <Space.Compact block>
      <Input
        style={{ width: 'calc(100% - 200px)' }}
        defaultValue="git@github.com:ant-design/ant-design.git"
      />
      <Tooltip title="copy git url">
        <Button icon={<CopyOutlined />} />
      </Tooltip>
    </Space.Compact>
    <Space.Compact block>
      <Select defaultValue="Zhejiang" allowClear>
        <Option value="Zhejiang">Zhejiang</Option>
        <Option value="Jiangsu">Jiangsu</Option>
      </Select>
      <Input style={{ width: '50%' }} defaultValue="Xihu District,
Hangzhou" />
```

```
    </Space.Compact>
    <Space.Compact block>
      <Select allowClear mode="multiple" defaultValue="Zhejianggggg" style=
{{ width: '50%' }}>
        <Option value="Zhejianggggg">Zhejianggggg</Option>
        <Option value="Jiangsu">Jiangsu</Option>
      </Select>
      <Input style={{ width: '50%' }} defaultValue="Xihu District,
Hangzhou" />
    </Space.Compact>
    <Space.Compact block>
      <Input.Search style={{ width: '30%' }} defaultValue="0571" />
      <Input.Search allowClear style={{ width: '50%' }}
defaultValue="26888888" />
      <Input.Search style={{ width: '20%' }} defaultValue="+1" />
    </Space.Compact>
    <Space.Compact block>
      <Select defaultValue="Option1">
        <Option value="Option1">Option1</Option>
        <Option value="Option2">Option2</Option>
      </Select>
      <Input style={{ width: '50%' }} defaultValue="input content" />
      <InputNumber defaultValue={12} />
    </Space.Compact>
    <Space.Compact block>
      <Input style={{ width: '50%' }} defaultValue="input content" />
      <DatePicker style={{ width: '50%' }} />
    </Space.Compact>
    <Space.Compact block>
      <DatePicker.RangePicker style={{ width: '70%' }} />
      <Input style={{ width: '30%' }} defaultValue="input content" />
      <Button type="primary">查询</Button>
    </Space.Compact>
    <Space.Compact block>
      <Input style={{ width: '30%' }} defaultValue="input content" />
      <DatePicker.RangePicker style={{ width: '70%' }} />
    </Space.Compact>
    <Space.Compact block>
      <Select defaultValue="Option1-1">
        <Option value="Option1-1">Option1-1</Option>
        <Option value="Option1-2">Option1-2</Option>
      </Select>
      <Select defaultValue="Option2-2">
        <Option value="Option2-1">Option2-1</Option>
        <Option value="Option2-2">Option2-2</Option>
      </Select>
    </Select>
```

```jsx
    </Space.Compact>
    <Space.Compact block>
      <Select defaultValue="1">
        <Option value="1">Between</Option>
        <Option value="2">Except</Option>
      </Select>
      <Input style={{ width: 100, textAlign: 'center' }}
placeholder="Minimum" />
      <Input
        className="site-input-split"
        style={{
          width: 30,
          borderLeft: 0,
          borderRight: 0,
          pointerEvents: 'none',
        }}
        placeholder="~"
        disabled
      />
      <Input
        className="site-input-right"
        style={{
          width: 100,
          textAlign: 'center',
        }}
        placeholder="Maximum"
      />
    </Space.Compact>
    <Space.Compact block>
      <Select defaultValue="Sign Up" style={{ width: '30%' }}>
        <Option value="Sign Up">Sign Up</Option>
        <Option value="Sign In">Sign In</Option>
      </Select>
      <AutoComplete
        style={{ width: '70%' }}
        placeholder="Email"
        options={[{ value: 'text 1' }, { value: 'text 2' }]}
      />
    </Space.Compact>
    <Space.Compact block>
      <TimePicker style={{ width: '70%' }} />
      <Cascader
        style={{ width: '70%' }}
        options={[
          {
            value: 'zhejiang',
```

```
            label: 'Zhejiang',
            children: [
              {
                value: 'hangzhou',
                label: 'Hangzhou',
                children: [
                  {
                    value: 'xihu',
                    label: 'West Lake',
                  },
                ],
              },
            ],
          },
          {
            value: 'jiangsu',
            label: 'Jiangsu',
            children: [
              {
                value: 'nanjing',
                label: 'Nanjing',
                children: [
                  {
                    value: 'zhonghuamen',
                    label: 'Zhong Hua Men',
                  },
                ],
              },
            ],
          },
        ]}
        placeholder="Select Address"
      />
    </Space.Compact>
    <Space.Compact block>
      <TimePicker.RangePicker />
      <TreeSelect
        showSearch
        style={{ width: '60%' }}
        value="leaf1"
        dropdownStyle={{ maxHeight: 400, overflow: 'auto' }}
        placeholder="Please select"
        allowClear
        treeDefaultExpandAll
        onChange={() => {}}
      >
```

```jsx
        <TreeNode value="parent 1" title="parent 1">
          <TreeNode value="parent 1-0" title="parent 1-0">
            <TreeNode value="leaf1" title="leaf1" />
            <TreeNode value="leaf2" title="leaf2" />
          </TreeNode>
          <TreeNode value="parent 1-1" title="parent 1-1">
            <TreeNode value="leaf3" title={<b style={{ color: '#08c'
}}>leaf3</b>} />
          </TreeNode>
        </TreeNode>
      </TreeSelect>
      <Button type="primary">Submit</Button>
    </Space.Compact>
    <Space.Compact>
      <Input placeholder="input here" />
      <InputNumber placeholder="another input" addonBefore="$" />
      <InputNumber placeholder="another input" addonAfter="$" />
    </Space.Compact>
    <Space.Compact>
      <Input placeholder="input here" />
      <ColorPicker />
    </Space.Compact>
    <Space.Compact>
      <Button type="primary">Button</Button>
      <Input placeholder="input here" addonAfter="$" />
    </Space.Compact>
  </Space>
);

export default App;
```

Button Compact Mode

```jsx
import React from 'react';
import {
  CommentOutlined,
  DownloadOutlined,
  EllipsisOutlined,
  HeartOutlined,
  LikeOutlined,
  MailOutlined,
  MobileOutlined,
  ShareAltOutlined,
  StarOutlined,
  WarningOutlined,
} from '@ant-design/icons';
```

```jsx
import { Button, Dropdown, Space, Tooltip } from 'antd';

const App: React.FC = () => (
  <div>
    <Space.Compact block>
      <Tooltip title="Like">
        <Button icon={<LikeOutlined />} />
      </Tooltip>
      <Tooltip title="Comment">
        <Button icon={<CommentOutlined />} />
      </Tooltip>
      <Tooltip title="Star">
        <Button icon={<StarOutlined />} />
      </Tooltip>
      <Tooltip title="Heart">
        <Button icon={<HeartOutlined />} />
      </Tooltip>
      <Tooltip title="Share">
        <Button icon={<ShareAltOutlined />} />
      </Tooltip>
      <Tooltip title="Download">
        <Button icon={<DownloadOutlined />} />
      </Tooltip>
      <Dropdown
        placement="bottomRight"
        menu={{
          items: [
            {
              key: '1',
              label: 'Report',
              icon: <WarningOutlined />,
            },
            {
              key: '2',
              label: 'Mail',
              icon: <MailOutlined />,
            },
            {
              key: '3',
              label: 'Mobile',
              icon: <MobileOutlined />,
            },
          ],
        }}
        trigger={['click']}
      >
```

```
        <Button icon={<EllipsisOutlined />} />
    </Dropdown>
</Space.Compact>
<br />
<Space.Compact block>
    <Button type="primary">Button 1</Button>
    <Button type="primary">Button 2</Button>
    <Button type="primary">Button 3</Button>
    <Button type="primary">Button 4</Button>
    <Tooltip title="Tooltip">
        <Button type="primary" icon={<DownloadOutlined />} disabled />
    </Tooltip>
    <Tooltip title="Tooltip">
        <Button type="primary" icon={<DownloadOutlined />} />
    </Tooltip>
</Space.Compact>
<br />
<Space.Compact block>
    <Button>Button 1</Button>
    <Button>Button 2</Button>
    <Button>Button 3</Button>
    <Tooltip title="Tooltip">
        <Button icon={<DownloadOutlined />} disabled />
    </Tooltip>
    <Tooltip title="Tooltip">
        <Button icon={<DownloadOutlined />} />
    </Tooltip>
    <Button type="primary">Button 4</Button>
    <Dropdown
        placement="bottomRight"
        menu={{
            items: [
                {
                    key: '1',
                    label: '1st item',
                },
                {
                    key: '2',
                    label: '2nd item',
                },
                {
                    key: '3',
                    label: '3rd item',
                },
            ],
        }}
```

```
                trigger={['click']}
            >
                <Button type="primary" icon={<EllipsisOutlined />} />
            </Dropdown>
        </Space.Compact>
    </div>
);

export default App;
```

Vertical Compact Mode

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

const App: React.FC = () => (
    <Space>
        <Space.Compact direction="vertical">
            <Button>Button 1</Button>
            <Button>Button 2</Button>
            <Button>Button 3</Button>
        </Space.Compact>
        <Space.Compact direction="vertical">
            <Button type="dashed">Button 1</Button>
            <Button type="dashed">Button 2</Button>
            <Button type="dashed">Button 3</Button>
        </Space.Compact>
        <Space.Compact direction="vertical">
            <Button type="primary">Button 1</Button>
            <Button type="primary">Button 2</Button>
            <Button type="primary">Button 3</Button>
        </Space.Compact>
    </Space>
);

export default App;
```

Input addon debug

Debug

```
import React, { useState } from 'react';
import { CopyOutlined, DownloadOutlined, SettingOutlined } from '@ant-
design/icons';
import {
    Button,
```

```jsx
  Cascader,
  DatePicker,
  Divider,
  Drawer,
  Dropdown,
  Input,
  InputNumber,
  Modal,
  Popover,
  Select,
  Space,
  Tooltip,
} from 'antd';

const { Option } = Select;

const selectBefore = (
  <Select defaultValue="http://" className="select-before">
    <Option value="http://">http://</Option>
    <Option value="https://">https://</Option>
  </Select>
);
const selectAfter = (
  <Select defaultValue=".com" className="select-after">
    <Option value=".com">.com</Option>
    <Option value=".jp">.jp</Option>
    <Option value=".cn">.cn</Option>
    <Option value=".org">.org</Option>
  </Select>
);

const App: React.FC = () => {
  const [showModal, setShowModal] = useState(false);
  const [showDrawer, setShowDrawer] = useState(false);
  return (
    <Space direction="vertical">
      <Space.Compact block>
        <Button>default Button</Button>
        <Button danger>danger Button</Button>
        <Button type="dashed">dashed Button</Button>
        <Button type="text">text Button</Button>
        <Button type="link">Link Button</Button>
        <Tooltip title="Tooltip">
          <Button icon={<DownloadOutlined />} disabled />
        </Tooltip>
      </Space.Compact>
```

```jsx
      <br />
      <Space.Compact>
        <Button>Prefix</Button>
        <Input addonBefore="http://" addonAfter=".com"
defaultValue="mysite" />
        <Button type="primary">Submit</Button>
      </Space.Compact>
      <Space.Compact>
        <Input placeholder="prefix" />
        <Input addonBefore={selectBefore} addonAfter={selectAfter}
defaultValue="mysite" />
        <Button icon={<CopyOutlined />} />
      </Space.Compact>
      <Space.Compact>
        <Input.Search />
        <Input.Search />
        <Button icon={<CopyOutlined />} />
      </Space.Compact>
      <Space.Compact>
        <Input addonAfter={<SettingOutlined />} defaultValue="mysite" />
        <Button type="primary">Submit</Button>
        <Input placeholder="suffix" addonAfter={<SettingOutlined />} />
      </Space.Compact>
      <Space.Compact>
        <Input addonBefore="http://" suffix=".com" defaultValue="mysite" />
        <Button type="primary">Submit</Button>
      </Space.Compact>
      <Space.Compact>
        <Button>Prefix</Button>
        <Input
          addonBefore={<Cascader placeholder="cascader" style={{ width: 150
}} />}
          defaultValue="mysite"
        />
        <Button type="primary">Submit</Button>
      </Space.Compact>
      <Space.Compact>
        <Input addonBefore="Prefix" defaultValue="mysite" showCount />
        <Button type="primary">Submit</Button>
        <Input
          addonBefore="Prefix"
          defaultValue="mysite"
          showCount
          addonAfter={<SettingOutlined />}
        />
        <Input addonBefore="Prefix" defaultValue="mysite" showCount />
```

```
      </Space.Compact>
      <br />
      <Space.Compact>
        <Button onClick={() => setShowModal(true)}>debug Modal
context</Button>
        {showModal && (
          <Modal title="Basic Modal" open={showModal} onCancel={() =>
setShowModal(false)}>
            <Button>normal button A</Button>
            <Button>normal button B</Button>
            <br />
            <br />
            <Input />
            <br />
            <br />
            <Space.Compact>
              <Button>compact button A</Button>
              <Button>compact button B</Button>
            </Space.Compact>
          </Modal>
        )}
      </Space.Compact>
      <Space.Compact>
        <Dropdown.Button
          menu={{
            items: [
              {
                key: '1',
                label: <Button>menu button</Button>,
              },
              {
                key: '2',
                label: 'normal menu item',
              },
            ],
          }}
        >
          debug Dropdown.Button context
        </Dropdown.Button>
      </Space.Compact>
      <Space.Compact>
        <Button onClick={() => setShowDrawer(true)}>debug Drawer
context</Button>
        {showDrawer && (
          <Drawer
            title="Basic Drawer"
```

```jsx
          placement="right"
          onClose={() => setShowDrawer(false)}
          open={showDrawer}
        >
          <Button>normal button A</Button>
          <Button>normal button B</Button>
          <br />
          <br />
          <Space.Compact>
            <Button>compact button A</Button>
            <Button>compact button B</Button>
          </Space.Compact>
        </Drawer>
      )}
    </Space.Compact>
    <Space.Compact>
      <Input placeholder="Debug Popover context" />
      <Popover
        content={
          <>
            <Input placeholder="Left Border" />
            <Divider />
            <DatePicker />
            <Divider />
            <InputNumber />
            <Divider />
            <Select />
          </>
        }
        trigger={['click']}
        placement="bottom"
      >
        <Button>Settings</Button>
      </Popover>
    </Space.Compact>
    <Space.Compact>
      <InputNumber addonBefore="+" addonAfter="$" defaultValue={100} />
    </Space.Compact>
    <Space.Compact>
      <Select defaultValue="Sign Up">
        <Option value="Sign Up">Sign Up</Option>
        <Option value="Sign In">Sign In</Option>
      </Select>
    </Space.Compact>
    <Space.Compact>
      <DatePicker.RangePicker style={{ width: '70%' }} />
```

```
    </Space.Compact>
    <Space.Compact>
      <InputNumber defaultValue={12} />
    </Space.Compact>
    <Space.Compact>
      <Cascader
        style={{ width: '70%' }}
        options={[
          {
            value: 'zhejiang',
            label: 'Zhejiang',
            children: [
              {
                value: 'hangzhou',
                label: 'Hangzhou',
                children: [
                  {
                    value: 'xihu',
                    label: 'West Lake',
                  },
                ],
              },
            ],
          },
          {
            value: 'jiangsu',
            label: 'Jiangsu',
            children: [
              {
                value: 'nanjing',
                label: 'Nanjing',
                children: [
                  {
                    value: 'zhonghuamen',
                    label: 'Zhong Hua Men',
                  },
                ],
              },
            ],
          },
        ]}
        placeholder="Select Address"
      />
    </Space.Compact>
    <Space.Compact direction="vertical">
      <Button>vertical compact button A</Button>
```

```jsx
        </Space.Compact>
      </Space>
    );
  };


  export default App;
```

Nested Space Compact

Debug

```jsx
import React from 'react';
import { CopyOutlined, SearchOutlined } from '@ant-design/icons';
import { Button, Cascader, Input, InputNumber, Select, Space, TimePicker }
from 'antd';

const { Option } = Select;

const App: React.FC = () => (
  <>
    <Space.Compact block>
      <Space.Compact>
        <Space.Compact>
          <Input style={{ width: 90 }} placeholder="Typing..." />
          <Button icon={<SearchOutlined />} />
        </Space.Compact>
        <Space.Compact>
          <InputNumber defaultValue={12} />
          <Select defaultValue="Option1">
            <Option value="Option1">Opt1</Option>
            <Option value="Option2">Opt2</Option>
          </Select>
        </Space.Compact>
      </Space.Compact>
      <Button type="primary">Separator</Button>
      <Space.Compact>
        <Space.Compact>
          <Input.Search style={{ width: 110 }} placeholder="Search" />
          <Button type="primary">Submit</Button>
        </Space.Compact>
        <Space.Compact>
          <Input defaultValue="mysite" />
          <Button icon={<CopyOutlined />} />
        </Space.Compact>
      </Space.Compact>
    </Space.Compact>
```

```jsx
<>
  <br />
  <Space.Compact block>
    <Space.Compact>
      <TimePicker />
      <Button type="primary">Submit</Button>
    </Space.Compact>
    <Space.Compact>
      <Cascader
        options={[
          {
            value: 'zhejiang',
            label: 'Zhejiang',
            children: [
              {
                value: 'hangzhou',
                label: 'Hangzhou',
                children: [
                  {
                    value: 'xihu',
                    label: 'West Lake',
                  },
                ],
              },
            ],
          },
          {
            value: 'jiangsu',
            label: 'Jiangsu',
            children: [
              {
                value: 'nanjing',
                label: 'Nanjing',
                children: [
                  {
                    value: 'zhonghuamen',
                    label: 'Zhong Hua Men',
                  },
                ],
              },
            ],
          },
        ]}
        placeholder="Select Address"
      />
      <Button type="primary">Submit</Button>
```

```
          </Space.Compact>
        </Space.Compact>
      </>
    </>
  );

export default App;
```

Diverse Child

Debug

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

const App: React.FC = () => (
  <Space>
    <>
      Button
      <Button>Button</Button>
    </>
    Button
    <Popconfirm title="Are you sure delete this task?" okText="Yes"
cancelText="No">
      <Button>Delete</Button>
    </Popconfirm>
    <Popconfirm title="Are you sure delete this task?" okText="Yes"
cancelText="No">
      <Button disabled>Delete</Button>
    </Popconfirm>
    {null}
    {false}
    {1}
    Button
    {null}
    {undefined}
  </Space>
);

export default App;
```

Flex gap style

Debug

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

const style: React.CSSProperties = {
  width: 150,
  height: 100,
  background: 'red',
};

const App: React.FC = () => {
  const [singleCol, setSingleCol] = useState(false);

  return (
    <>
      <Switch
        checked={singleCol}
        onChange={() => {
          setSingleCol(!singleCol);
        }}
      />
      <div style={{ boxShadow: '0 0 5px green' }}>
        <Space style={{ width: singleCol ? 307 : 310, background: 'blue' }} size={[8, 8]} wrap>
          <div style={style} />
          <div style={style} />
          <div style={style} />
          <div style={style} />
        </Space>
      </div>
    </>
  );
};

export default App;
```

API

Common props ref：[Common props](#)

Property	Description	Type	Default	Version
align	Align items	`start` | `end` |`center` |`baseline`	-	4.2.0
classNames	Semantic className	[Record<SemanticDOM, string>](#)	-	

direction	The space direction	`vertical|horizontal`	`horizontal`	4.1.0
size	The space size	[Size](#) | [Size[]](#)	`small`	4.1.0 | Array: 4.9.0
split	Set split	ReactNode	-	4.7.0
styles	Semantic style	[Record<SemanticDOM, CSSProperties>](#)	-	
wrap	Auto wrap line, when `horizontal` effective	boolean	false	4.9.0

Size

```
'small' | 'middle' | 'large' | number
```

Space.Compact

Use Space.Compact when child form components are compactly connected and the border is collapsed. The supported components are:

- Button
- AutoComplete
- Cascader
- DatePicker
- Input/Input.Search
- InputNumber
- Select
- TimePicker
- TreeSelect

Property	Description	Type	Default	Version
block	Option to fit width to its parent's width	boolean	false	4.24.0
direction	Set direction of layout	`vertical|horizontal`	`horizontal`	4.24.0
size	Set child component size	`large|middle|small`	`middle`	4.24.0

Semantic DOM

演示

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

import SemanticPreview from '../../../.dumi/components/SemanticPreview';
import useLocale from '../../../.dumi/hooks/useLocale';

const locales = {
  cn: {
    item: '包裹的子组件',
  },
  en: {
    item: 'Wrapped item element',
  },
};

const App: React.FC = () => {
  const [locale] = useLocale(locales);
  return (
    <SemanticPreview
      componentName="Space"
      semantics={[{ name: 'item', desc: locale.item, version: '5.6.0' }]}
    >
      <Space>
        <Button type="primary">Primary</Button>
        <Button>Default</Button>
        <Button type="dashed">Dashed</Button>
      </Space>
    </SemanticPreview>
  );
};

export default App;
```

Design Token