

## Examples

### Basic

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

const App: React.FC = () => (
  <Space direction="vertical" size={16}>
    <Space wrap size={16}>
      <Avatar size={64} icon={<UserOutlined />} />
      <Avatar size="large" icon={<UserOutlined />} />
      <Avatar icon={<UserOutlined />} />
      <Avatar size="small" icon={<UserOutlined />} />
      <Avatar size={14} icon={<UserOutlined />} />
    </Space>
    <Space wrap size={16}>
      <Avatar shape="square" size={64} icon={<UserOutlined />} />
      <Avatar shape="square" size="large" icon={<UserOutlined />} />
      <Avatar shape="square" icon={<UserOutlined />} />
      <Avatar shape="square" size="small" icon={<UserOutlined />} />
      <Avatar shape="square" size={14} icon={<UserOutlined />} />
    </Space>
  </Space>
);

export default App;
```

### Type

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

const url =
'https://gw.alipayobjects.com/zos/rmsportal/KDpgvguMpGfqaHPjicRK.svg';

const App: React.FC = () => (
  <Space size={16} wrap>
    <Avatar icon={<UserOutlined />} />
    <Avatar>U</Avatar>
    <Avatar size={40}>USER</Avatar>
    <Avatar src={url} />
    <Avatar src={<img src={url} alt="avatar" />} />
  </Space>
);
```

```

    <Avatar style={{ backgroundColor: '#fde3cf', color: '#f56a00'
}}>U</Avatar>
    <Avatar style={{ backgroundColor: '#87d068' }} icon={<UserOutlined />}
/>
  </Space>
);

export default App;

```

## Autoset Font Size

```

import React, { useState } from 'react';
import { Avatar, Button } from 'antd';

const UserList = ['U', 'Lucy', 'Tom', 'Edward'];
const ColorList = ['#f56a00', '#7265e6', '#ffbf00', '#00a2ae'];
const GapList = [4, 3, 2, 1];

const App: React.FC = () => {
  const [user, setUser] = useState(UserList[0]);
  const [color, setColor] = useState(ColorList[0]);
  const [gap, setGap] = useState(GapList[0]);

  const changeUser = () => {
    const index = UserList.indexOf(user);
    setUser(index < UserList.length - 1 ? UserList[index + 1] :
UserList[0]);
    setColor(index < ColorList.length - 1 ? ColorList[index + 1] :
ColorList[0]);
  };

  const changeGap = () => {
    const index = GapList.indexOf(gap);
    setGap(index < GapList.length - 1 ? GapList[index + 1] : GapList[0]);
  };

  return (
    <>
      <Avatar style={{ backgroundColor: color, verticalAlign: 'middle' }}
size="large" gap={gap}>
        {user}
      </Avatar>
      <Button
        size="small"
        style={{ margin: '0 16px', verticalAlign: 'middle' }}
        onClick={changeUser}

```

```

      >
        ChangeUser
      </Button>
      <Button size="small" style={{ verticalAlign: 'middle' }} onClick=
{changeGap}>
        changeGap
      </Button>
    </>
  );
};

export default App;

```

## With Badge

```

import React from 'react';
import { UserOutlined } from '@ant-design/icons';
import { Avatar, Badge, Space } from 'antd';

const App: React.FC = () => (
  <Space size={24}>
    <Badge count={1}>
      <Avatar shape="square" icon={<UserOutlined />} />
    </Badge>
    <Badge dot>
      <Avatar shape="square" icon={<UserOutlined />} />
    </Badge>
  </Space>
);

export default App;

```

## Avatar.Group

```

import React from 'react';
import { AntDesignOutlined, UserOutlined } from '@ant-design/icons';
import { Avatar, Divider, Tooltip } from 'antd';

const App: React.FC = () => (
  <>
    <Avatar.Group>
      <Avatar src="https://api.dicebear.com/7.x/miniavs/svg?seed=1" />
      <a href="https://ant.design">
        <Avatar style={{ backgroundColor: '#f56a00' }}>K</Avatar>
      </a>
    </Avatar.Group>
  </>
);

```

```

        <Tooltip title="Ant User" placement="top">
          <Avatar style={{ backgroundColor: '#87d068' }} icon={<UserOutlined
/>} />
        </Tooltip>
        <Avatar style={{ backgroundColor: '#1677ff' }} icon=
{<AntDesignOutlined />} />
      </Avatar.Group>
      <Divider />
      <Avatar.Group
        max={{
          count: 2,
          style: { color: '#f56a00', backgroundColor: '#fde3cf' },
        }}
      >
        <Avatar src="https://api.dicebear.com/7.x/miniavs/svg?seed=2" />
        <Avatar style={{ backgroundColor: '#f56a00' }}>K</Avatar>
        <Tooltip title="Ant User" placement="top">
          <Avatar style={{ backgroundColor: '#87d068' }} icon={<UserOutlined
/>} />
        </Tooltip>
        <Avatar style={{ backgroundColor: '#1677ff' }} icon=
{<AntDesignOutlined />} />
      </Avatar.Group>
      <Divider />
      <Avatar.Group
        size="large"
        max={{
          count: 2,
          style: { color: '#f56a00', backgroundColor: '#fde3cf' },
        }}
      >
        <Avatar src="https://api.dicebear.com/7.x/miniavs/svg?seed=3" />
        <Avatar style={{ backgroundColor: '#f56a00' }}>K</Avatar>
        <Tooltip title="Ant User" placement="top">
          <Avatar style={{ backgroundColor: '#87d068' }} icon={<UserOutlined
/>} />
        </Tooltip>
        <Avatar style={{ backgroundColor: '#1677ff' }} icon=
{<AntDesignOutlined />} />
      </Avatar.Group>
      <Divider />
      <Avatar.Group
        size="large"
        max={{
          count: 2,
          style: { color: '#f56a00', backgroundColor: '#fde3cf', cursor:

```

```

'pointer' },
    popover: { trigger: 'click' },
  }}
>
  <Avatar
src="https://zos.alipayobjects.com/rmsportal/ODTLcjxAfvqbxHnVXCYX.png" />
  <Avatar style={{ backgroundColor: '#f56a00' }}>K</Avatar>
  <Tooltip title="Ant User" placement="top">
    <Avatar style={{ backgroundColor: '#87d068' }} icon={<UserOutlined
/>} />
  </Tooltip>
  <Avatar style={{ backgroundColor: '#1677ff' }} icon=
{<AntDesignOutlined />} />
  </Avatar.Group>
  <Divider />
  <Avatar.Group shape="square">
    <Avatar style={{ backgroundColor: '#fde3cf' }}>A</Avatar>
    <Avatar style={{ backgroundColor: '#f56a00' }}>K</Avatar>
    <Avatar style={{ backgroundColor: '#87d068' }} icon={<UserOutlined
/>} />
    <Avatar style={{ backgroundColor: '#1677ff' }} icon=
{<AntDesignOutlined />} />
  </Avatar.Group>
</>
);

export default App;

```

## Calculate text style when hiding

Debug

```

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

type SizeType = 'large' | 'small' | 'default' | number;

const App: React.FC = () => {
  const [hide, setHide] = useState(true);
  const [size, setSize] = useState<SizeType>('large');
  const [scale, setScale] = useState(1);

  const toggle = () => {
    setHide(!hide);
  };
};

```



```
export default App;
```

## Responsive Size

```
import React from 'react';
import { AntDesignOutlined } from '@ant-design/icons';
import { Avatar } from 'antd';

const App: React.FC = () => (
  <Avatar
    size={{ xs: 24, sm: 32, md: 40, lg: 64, xl: 80, xxl: 100 }}
    icon={<AntDesignOutlined />}
  />
);

export default App;
```

## Fallback

Debug

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

const App: React.FC = () => (
  <Space>
    <Avatar shape="circle" src="http://abc.com/not-exist.jpg">
      A
    </Avatar>
    <Avatar shape="circle" src="http://abc.com/not-exist.jpg">
      ABC
    </Avatar>
  </Space>
);

export default App;
```

## Component Token

Debug

```
import React from 'react';
import { AntDesignOutlined, UserOutlined } from '@ant-design/icons';
import { Avatar, Badge, ConfigProvider, Space, Tooltip } from 'antd';
```

```

const App: React.FC = () => (
  <ConfigProvider
    theme={{
      components: {
        Avatar: {
          containerSize: 60,
          containerSizeLG: 30,
          containerSizeSM: 16,

          textFontSize: 18,
          textFontSizeLG: 28,
          textFontSizeSM: 12,

          borderRadius: 10,
          groupOverlapping: -10,
          groupBorderColor: '#eee',
        },
      },
    }}
  >
    <Space>
      <Avatar shape="circle" src="http://abc.com/not-exist.jpg">
        A
      </Avatar>
    </Space>
    <Space>
      <Avatar.Group
        max={{
          count: 2,
          style: { color: '#f56a00', backgroundColor: '#fde3cf' },
        }}
      >
        <Avatar src="https://api.dicebear.com/7.x/miniavs/svg?seed=2" />
        <Avatar style={{ backgroundColor: '#f56a00' }}>K</Avatar>
        <Tooltip title="Ant User" placement="top">
          <Avatar style={{ backgroundColor: '#87d068' }} icon=
{<UserOutlined />} />
        </Tooltip>
        <Avatar style={{ backgroundColor: '#1890ff' }} icon=
{<AntDesignOutlined />} />
      </Avatar.Group>
    </Space>
    <Space>
      <Badge count={1}>
        <Avatar shape="square" icon={<UserOutlined />} />
      </Badge>
    </Space>
  </ConfigProvider>
)

```



```

    <Badge dot>
      <Avatar shape="square" icon={<UserOutlined />} />
    </Badge>
  </Space>
</ConfigProvider>
);

export default App;

```

## API

Common props ref: [Common props](#)

### Avatar

Property	Description	Type	Default	Version
alt	This attribute defines the alternative text describing the image	string	-	
gap	Letter type unit distance between left and right sides	number	4	4.3.0
icon	Custom icon type for an icon avatar	ReactNode	-	
shape	The shape of avatar	circle   square	circle	
size	The size of the avatar	number   large   small   default   { xs: number, sm: number, ... }	default	4.7.0
src	The address of the image for an image avatar or image element	string   ReactNode	-	ReactNode: 4.8.0
srcSet	A list of sources to use for different screen resolutions	string	-	
draggable	Whether the picture is allowed to be dragged	boolean   'true'   'false'	true	
crossOrigin	CORS settings attributes	'anonymous'   'use-credentials'   ''	-	4.17.0

onError	Handler when img load error, return false to prevent default fallback behavior	() => boolean	-	
---------	--	---------------	---	--

*Tip: You can set `icon` or `children` as the fallback for image load error, with the priority of `icon` > `children`*

#### Avatar.Group 4.5.0+

Property	Description	Type	Default	Version
max	Set maximum display related configurations, Before 5.18.0 you can use <a href="#">parameters</a>	{ count?: number; style?: CSSProperties; popover?: PopoverProps }	-	5.18.0
size	The size of the avatar	number   large   small   default   { xs: number, sm: number, ...}	default	4.8.0
shape	The shape of the avatar	circle   square	circle	5.8.0

## Design Token