

设计师专属

安装 [Kitchen Sketch 插件](#) , 一键填充高逼格头像和文本。

代码演示

基本

```
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;
```

类型

```
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>
  </Space>
);
```

```

    <Avatar size={40}>USER</Avatar>
    <Avatar src={url} />
    <Avatar src={<img src={url} alt="avatar" />} />
    <Avatar style={{ backgroundColor: '#fde3cf', color: '#f56a00'
}}>U</Avatar>
    <Avatar style={{ backgroundColor: '#87d068' }} icon={<UserOutlined />}
/>
  </Space>
);

export default App;

```

自动调整字符大小

```

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;

```

带徽标的头像

```

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" />
    </Avatar.Group>
  </>
);

```

```

    <a href="https://ant.design">
      <Avatar style={{ backgroundColor: '#f56a00' }}>K</Avatar>
    </a>
    <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;

```

隐藏情况下计算字符对齐

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);
};

const toggleSize = () => {
    const sizes = ['small', 'default', 'large'] as SizeType[];
    let current = sizes.indexOf(size) + 1;
    if (current > 2) {
        current = 0;
    }
    setSize(sizes[current]);
};

const changeScale = () => {
    setScale(scale === 1 ? 2 : 1);
};

return (
    <>
        <Space wrap>
            <Button onClick={toggle}>Toggle Avatar visibility</Button>
            <Button onClick={toggleSize}>Toggle Avatar size</Button>
            <Button onClick={changeScale}>Change Avatar scale</Button>
        </Space>
        <div style={{ textAlign: 'center', transform: `scale(${scale})`,
marginTop: 24 }}>
            <Avatar size={size} style={{ background: '#7265e6', display: hide ?
'none' : '' }}>
                Avatar
            </Avatar>
            <Avatar
                size={size}
                src="invalid"
                style={{ background: '#00a2ae', display: hide ? 'none' : '' }}
            >
                Invalid
            </Avatar>
            <div style={{ display: hide ? 'none' : '' }}>
                <Avatar size={size} style={{ background: '#7265e6' }}>
                    Avatar
                </Avatar>
                <Avatar size={size} src="invalid" style={{ background: '#00a2ae'
}}>
                    Invalid
                </Avatar>
            </div>
        </div>
    </>

```

```
    </>
  );
};

export default App;
```

响应式尺寸

```
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;
```

图片不存在时

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;
```

组件 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>
  </ConfigProvider>
)

```



```

    <Space>
      <Badge count={1}>
        <Avatar shape="square" icon={<UserOutlined />} />
      </Badge>
      <Badge dot>
        <Avatar shape="square" icon={<UserOutlined />} />
      </Badge>
    </Space>
  </ConfigProvider>
);

export default App;
```

API

通用属性参考: [通用属性](#)

Avatar

参数	说明	类型	默认值	版本
alt	图像无法显示时的替代文本	string	-	
gap	字符类型距离左右两侧边界单位像素	number	4	4.3.0
icon	设置头像的自定义图标	ReactNode	-	
shape	指定头像的形状	circle square	circle	
size	设置头像的大小	number large small default { xs: number, sm: number, ...}	default	4.7.0
src	图片类头像的资源地址或者图片元素	string ReactNode	-	ReactNode: 4.8.0
srcSet	设置图片类头像响应式资源地址	string	-	
draggable	图片是否允许拖动	boolean 'true' 'false'	true	
crossOrigin	CORS 属性设置	'anonymous' 'use-credentials' ''	-	4.17.0
onError	图片加载失败的事件, 返回 false 会关	() => boolean	-	

	闭组件默认的 fallback 行为			
--	-----------------------	--	--	--

Tip: 你可以设置 `icon` 或 `children` 作为图片加载失败的默认 `fallback` 行为, 优先级为 `icon` > `children`

Avatar.Group 4.5.0+

参数	说明	类型	默认值	版本
max	设置最多显示相关配置, 5.18.0 前可使用 参数	{ count?: number; style?: CSSProperties; popover?: PopoverProps }	-	5.18.0
size	设置头像的大小	number large small default { xs: number, sm: number, ...}	default	4.8.0
shape	设置头像的形状	circle square	circle	5.8.0

主题变量 (Design Token)