语义化的矢量图形。使用图标组件，你需要安装 `@ant-design/icons` 图标组件包：

```
npm install --save @ant-design/icons
```

设计师专属

安装 Kitchen Sketch 插件 💎，就可以一键拖拽使用 Ant Design 和 Iconfont 的海量图标，还可以关联自有项目。

图标列表

```
import IconDisplay from 'site/theme/template/IconDisplay';
ReactDOM.render(<IconDisplay />, mountNode);
```

API

从 4.0 开始，antd 不再内置 Icon 组件，请使用独立的包 `@ant-design/icons` 。

通用图标

参数	说明	类型	默认值	版本
className	设置图标的样式名	string	-	
rotate	图标旋转角度（IE9 无效）	number	-	
spin	是否有旋转动画	boolean	false	
style	设置图标的样式，例如 `fontSize` 和 `color`	CSSProperties	-	
twoToneColor	仅适用双色图标。设置双色图标的主要颜色	string (十六进制颜色)	-	

其中我们提供了三种主题的图标，不同主题的 Icon 组件名为图标名加主题做为后缀。

```
import { StarOutlined, StarFilled, StarTwoTone } from '@ant-design/icons';

<StarOutlined />
<StarFilled />
<StarTwoTone twoToneColor="#eb2f96" />
```

自定义 Icon

参数	说明	类型	默认值	版本
component	控制如何渲染图标，通常是一个渲染根标签为 `<svg>` 的 React 组件	ComponentType<CustomIconComponentProps>	-	
rotate	图标旋转角度（IE9 无效）	number	-	

spin	是否有旋转动画	boolean	false	
style	设置图标的样式，例如 `fontSize` 和 `color`	CSSProperties	-	

关于 SVG 图标

在 `3.9.0` 之后，我们使用了 SVG 图标替换了原先的 font 图标，从而带来了以下优势：

- 完全离线化使用，不需要从 CDN 下载字体文件，图标不会因为网络问题呈现方块，也无需字体文件本地部署。
- 在低端设备上 SVG 有更好的清晰度。
- 支持多色图标。
- 对于内建图标的更换可以提供更多 API，而不需要进行样式覆盖。

更多讨论可参考：#10353。

所有的图标都会以 `<svg>` 标签渲染，可以使用 `style` 和 `className` 设置图标的大小和单色图标的颜色。例如：

```
import { MessageOutlined } from '@ant-design/icons';

<MessageOutlined style={{ fontSize: '16px', color: '#08c' }} />;
```

双色图标主色

对于双色图标，可以通过使用 `getTwoToneColor()` 和 `setTwoToneColor(colorString)` 来全局设置图标主色。

```
import { getTwoToneColor, setTwoToneColor } from '@ant-design/icons';

setTwoToneColor('#eb2f96');
getTwoToneColor(); // #eb2f96
```

自定义 font 图标

在 `3.9.0` 之后，我们提供了一个 `createFromIconfontCN` 方法，方便开发者调用在 iconfont.cn 上自行管理的图标。

```
import { createFromIconfontCN } from '@ant-design/icons';

const MyIcon = createFromIconfontCN({
  scriptUrl: '//at.alicdn.com/t/font_8d5l8fzk5b87iudi.js', // 在 iconfont.cn 上生成
});

ReactDOM.render(<MyIcon type="icon-example" />, mountedNode);
```

其本质上是创建了一个使用 `<use>` 标签来渲染图标的组件。

options 的配置项如下：

参数	说明	类型	默认值	版本
extraCommonProps	给所有的 `svg` 图标 `<Icon />` 组件设置额外的属性	{ [key: string]: any }	{}	
scriptUrl	iconfont.cn 项目在线生成的 js 地址，`@ant-design/icons@4.1.0` 之后支持 `string[]` 类型	string | string[]	-	

在 `scriptUrl` 都设置有效的情况下，组件在渲染前会自动引入 iconfont.cn 项目中的图标符号集，无需手动引入。

见 iconfont.cn 使用帮助 查看如何生成 js 地址。

自定义 SVG 图标

如果使用 `webpack`，可以通过配置 @svgr/webpack 来将 `svg` 图标作为 `React` 组件导入。`@svgr/webpack` 的 `options` 选项请参阅 svgr 文档。

```
// webpack.config.js
{
  test: /\.svg(\?v=\d+\.\d+\.\d+)?$/,
  use: [
    {
      loader: 'babel-loader',
    },
    {
      loader: '@svgr/webpack',
      options: {
        babel: false,
        icon: true,
      },
    },
  ],
}
```

```
import Icon from '@ant-design/icons';
import MessageSvg from 'path/to/message.svg'; // path to your '*.svg' file.
// in create-react-app:
// import { ReactComponent as MessageSvg } from 'path/to/message.svg';

ReactDOM.render(<Icon component={MessageSvg} />, mountNode);
```

`Icon` 中的 `component` 组件的接受的属性如下：

字段	说明	类型	只读值	版本
className	计算后的 `svg` 类名	string	-	
fill	`svg` 元素填充的颜色	string	`currentColor`	

height	`svg` 元素高度	string | number	`1em`	
style	计算后的 `svg` 元素样式	CSSProperties	-	
width	`svg` 元素宽度	string | number	`1em`	