

Button使用

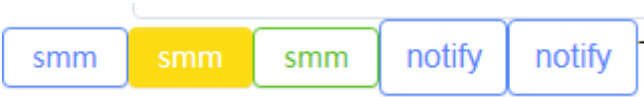
通过设置 Button 的属性来产生不同的按钮样式。

按钮的属性说明如下：

属性	说明	类型	默认值	版本
disabled	按钮失效状态	boolean	false	1.0
href	点击跳转的地址，指定此属性 button 的行为和 a 链接一致	string	-	1.0
size	设置按钮大小	lg	md	sm
btnType	设置按钮类型	primary	info	success
onClick	点击按钮时的回调	(event) => void	-	1.0

支持原生 button 的其他所有属性。

```
<Button className="btn-class" btnType="warning" size="ssm"></Button>
```



Alert提醒框使用

参数	说明	类型	默认值	版本
description	警告提示的辅助性文字介绍	ReactNode	-	1.0
message	警告提示内容	ReactNode	-	1.0
showIcon	是否显示辅助图标	boolean	false	1.0
type	指定警告提示的样式，有四种选择 success、info、warning、error	string	info , banner 模式下默认值为 warning	1.0
onClose	关闭时触发的回调函数	() => void	-	1.0

[illegible]

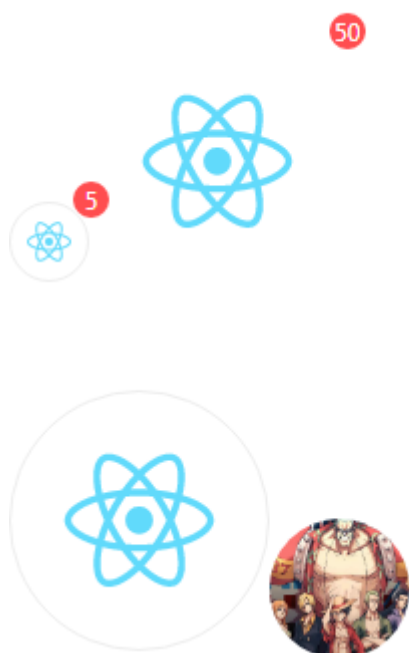
参数	说明	类型	默认值	版本
alt	图像无法显示时的替代文本	string	-	1.0
size	设置头像的大小	lg <code>ssm</code>	<code>sm</code>	<code>default</code>
href	图片头像的资源地址	string	-	1.0
border	边框	boolean	false	1.0

```
<Avatar href="xxxxxxxxxxxx" border="true" size="ssm"></Avatar>
```



参数	说明	类型	默认值	版本
bgColor	字体颜色	string	-	1.0
count	展示的数字，大于 overflowCount 时显示为 <code>\${overflowCount}+</code> ，为 0 时隐藏	number	0	1.0
overflowCount	展示封顶的数字值	number	99	1.0

```
<Badge count={5} overflowCount={99}>
  <Avatar href={sv} border="true" size="ssm"></Avatar>
</Badge>
```



Breadcrumb面包屑

参数	说明	类型	默认值	版本
separator	分隔符	string	"/"	1.0

Breadcrumb.Item

参数	说明	类型	默认值	版本
to	路由地址，有传递则设置为可跳转	string		1.0

```
<Breadcrumb>
  <Breadcrumb.Item to="/aaa">Ant Design</Breadcrumb.Item>
  <Breadcrumb.Item to="/aaa">Component</Breadcrumb.Item>
  <Breadcrumb.Item>General</Breadcrumb.Item>
</Breadcrumb>
```

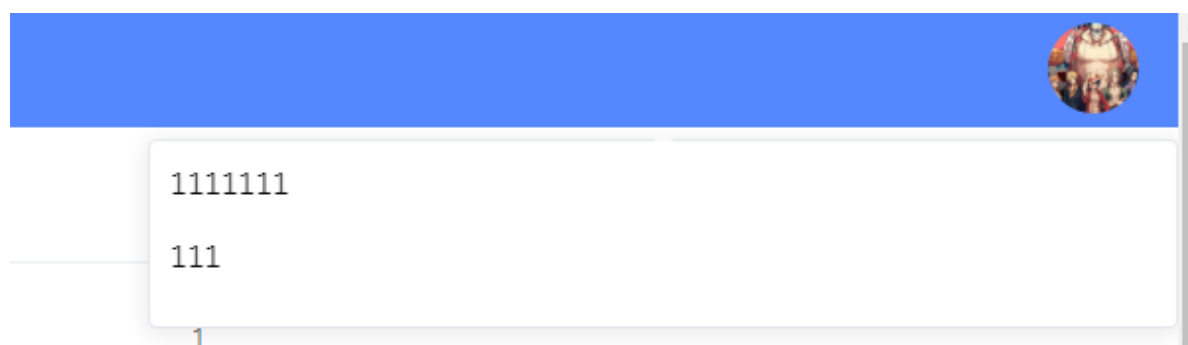
DropDown下拉列表

参数	说明	类型	默认值	版本
overlay	menu项，暂定包裹全部元素	DropDownMenu		1.0
positionClass	下拉菜单定位类名			

DropDownMenu

参数	说明	类型	默认值	版本
children	menu内容	ReactNode		1.0

```
<DropDown overlay={
  <DropDownMenu className="drop-menu">
    1111111
    <p>111</p>
  </DropDownMenu>}>
  <div className="nav-notify">
    <span><Icon type="nc-u-notice-header"></Icon></span>
    <Avatar size="sm"
href="https://image.lceda.cn/avatars/2020/10/e36x5ZmcEpm2ibm91N30rs2CzoHeGz8Ew0eR6dKD.jpeg"></Avatar>
  </div>
</DropDown>
```



Icon

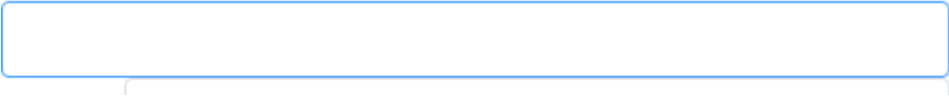
参数	说明	类型	默认值	版本
type	图标类型，参考原网站图标类型	string		1.0
iconType	引入iconfont图标类型	string		1.0
color	图标颜色	string		1.0
style	样式	CSSProperties		1.0
onClick	点击事件	()=>void		1.0

```
<Icon type="nc-u-search"></Icon>
```

Input输入框

参数	说明	类型	默认值	版本
type	声明 input 类型，同原生 input 标签的 type 属性，请直接使用 <code>TextArea</code> 代替 <code>type="textarea"</code>	string		1.0
value	输入框内容	string		1.0
disabled	是否禁用	false		1.0
onChangeInput	点击事件	(value)=>void		1.0

```
<Input className="myCalss" placeholder="name" onChangeInput={
(val:string)=>{console.log(val);
}}></Input>
```



Message提示消息

参数	说明	类型	默认值	版本
type	信息类型	string	default	1.0
message	消息内容	string		1.0
close	关闭按钮	boolean	false	1.0
style	样式	CSSProperties		1.0
closeCallback	点击事件	()=>void		1.0

```
<Message
  type="error"
  close
  message="当前账号尚未当前账号箱"
></Message>
```



Modal模态框

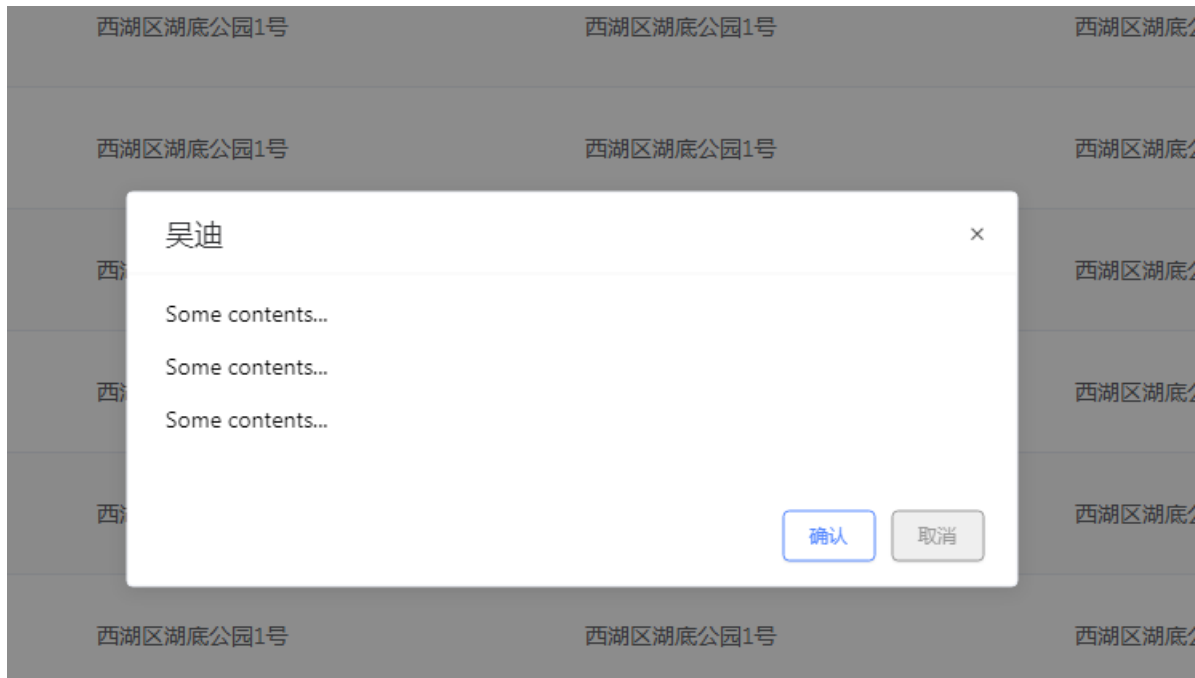
参数	说明	类型	默认值	版本
visible	模态框消失隐藏	boolean	false	1.0
title	模态框标题	ReactNode		1.0
showClose	是否显示关闭按钮	boolean	false	1.0
onCancel	取消回调	() => void		1.0
onOk	确认回调	() => void		1.0
onCancelText	取消按钮信息	string	取消	1.0
onOkText	确认按钮信息	string	确定	1.0

```
<Modal visible={vis}
  className="modal-big"
  showClose
  title="吴迪" onOk={() => {
    console.log("ok");
    setVis(false)
  }}
  onCancel={() => {
    setVis(false)
    console.log("onCancel");
  }} >
  <p>Some contents...</p>
  <p>Some contents...</p>
  <p>Some contents...</p>
</Modal>
```

Notification.open通知消息

参数	说明	类型	默认值	版本
type	通知类型	boolean	false	1.0
message	通知信息	string		1.0
duration	通知消失时间	number（单位秒）	3	1.0

```
<Button onClick={() => {  
  Notification  
    .open({  
      message: "error" + Math.random(),  
      type: "error",  
      duration: 3  
    })  
}}>  
  notify  
</Button>
```



Pagination分页

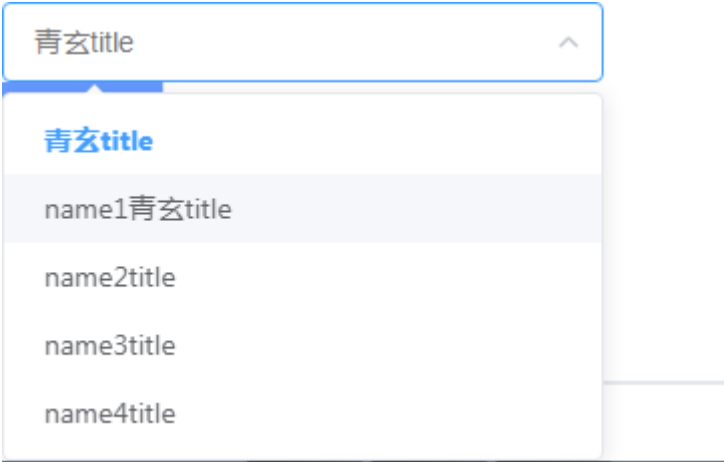
参数	说明	类型	默认值	版本
defaultCurrent	默认页	number	1	1.0
current	当前页	number	1	1.0
defaultPageSize	默认页面大小	number	20	1.0
pageSize	页面数据大小	number	20	1.0
total	数据总数	number	0	1.0
groupCount	显示页码大小	number	7	
onChange	页码改变的回调	(page: number, pageSize: number) => void		

```
<Pagination total={400} onChange={(page, pageSize) => {  
  console.log(page, pageSize);  
}}></Pagination>
```

Select选择器

参数	说明	类型	默认值	版本
defaultValue	默认option 的value, 输入框显示option对应的值	string		1.0
disabled	是否禁用	boolean	false	1.0
onFocus	获得焦点事件	() => void		1.0
onBlur	失去焦点事件	() => void		1.0
onChange	option改变事件	() => void		1.0
onSearch	取消按钮信息	string	取消	1.0

```
<Select defaultValue="青玄"
  onChange={value: string => {
    console.log(value);
  }}>
  <Option value="青玄">青玄title</Option>
  <Option value="name1">name1青玄title</Option>
  <Option value="name2">name2title</Option>
  <Option value="name3">name3title</Option>
  <Option value="name4">name4title</Option>
</Select>
```

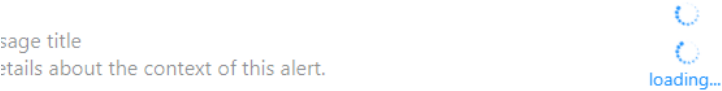


Spin 加载Loading

参数	说明	类型	默认值	版本
tip	loading提示	string		1.0
spinning	是否显示	boolean	false	1.0

```
<div className="">
  <Spin spinning={true} ></Spin>
</div>
<Spin spinning={spin} tip="loading...">
  <div data-show="true" className="eda-alert ant-alert-info ant-alert-with-description ant-alert-no-icon" role="alert">
    <div className="ant-alert-content">
      <div className="ant-alert-message">
        Alert message title</div>
      <div className="ant-alert-description">
        Further details about the context of this alert.
      </div>
    </div>
  </div>
</Spin>
```

Spin



Switch开关

参数	说明	类型	默认值	版本
defaultChecked	默认状态	boolean	false	1.0
disabled	是否禁用	boolean	false	1.0
onChange	状态改变回调	(value: boolean) => void		1.0

```
<Switch disabled defaultChecked></Switch>
<Switch defaultChecked onChange={(checked) => {
  console.log(checked);
}}></Switch>
```



Table表格

参数	说明	类型	默认值	版本
columns	默认状态	boolean	false	1.0
dataSource	是否禁用	boolean	false	1.0
pagination	是否显示页码	IPagination		1.0

```
interface IColumns {
  title: string,
  dataIndex: string,
  key: string,
  width?: number,
  render?: (item: string) => ReactNode
}
//dataSource 数据中的key须与IColumms中的key的值相对应
<Table pagination={{
  total: 400, defaultPageSize: 50, onChange: (page, pageSize) => {
    let data = dataSource.filter((item, index) => {
      if((index >= (page-1) * pageSize)&&(index < (page) * pageSize)){
        return true
      }else{
        return false
      }
    })
    console.log(data);

    setDataList(data)

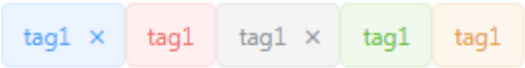
  }
}} stripe columns={columns} dataSource={dataList}>
</Table>
```

序号	操作	操作时间	积分变化	积分值
1	兑换优惠券	time	-1	1
2	32	西湖区湖底公园1号	西湖区湖底公园1号	西湖区湖底公园1号
2	32	西湖区湖底公园1号	西湖区湖底公园1号	西湖区湖底公园1号
2	32	西湖区湖底公园1号	西湖区湖底公园1号	西湖区湖底公园1号
2	32	西湖区湖底公园1号	西湖区湖底公园1号	西湖区湖底公园1号
2	32	西湖区湖底公园1号	西湖区湖底公园1号	西湖区湖底公园1号
2	32	西湖区湖底公园1号	西湖区湖底公园1号	西湖区湖底公园1号
2	32	西湖区湖底公园1号	西湖区湖底公园1号	西湖区湖底公园1号

Tabs标签页

参数	说明	类型	默认值	版本
message	标签信息	string		1.0
type	标签类型	React.ReactNode		1.0
closable	是否显示关闭按钮	boolean	false	1.0
close	关闭回调	()=>void		1.0

```
<Tag closable>tag1</Tag>
<Tag type="danger">tag1</Tag>
<Tag type="info" closable>tag1</Tag>
<Tag type="success">tag1</Tag>
<Tag type="warning">tag1</Tag>
```



Textarea多行文本

参数	说明	类型	默认值	版本
value	标签信息	string		1.0
rows	行长度	number		1.0
cols	列长度	number	false	1.0
disabled	是否禁用	boolean	false	1.0
resize	是否可拉伸	boolean	true	1.0
onChangeInput	输入框改变的回调	(value:string)=>void		1.0

```
<Textarea resize={false} value="默认值" placeholder="请输入"></Textarea>
```

默认值