



链滴

JSX 简介 --- JSX 语法注意事项

作者: [BYwangbing](#)

原文链接: <https://ld246.com/article/1571228728607>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

```
import React from 'react'
import ReactDOM from 'react-dom'

const element = <h1>Hello, world!</h1>;
```

  这种有趣的标签语法既不是字符串也不是HTML。

  这就是 JSX，他是 JavaScript 的一种扩展语法。我们推荐在 React 中使用种语法来描述 UI 信息。JSX 可能会让你想起某种模板语言，但是它具有 JavaScript 的全部能力。

  JSX 可以生成 React “元素”。

1. 什么是JSX语法：

  就是符合 xml 规范的 JS 语法；（语法格式相对来说，要比HTML严谨很多）

2. 如何启用 jsx 语法

  请移步：[Webpack4.0如何启用 jsx 语法](#)

3. jsx 语法的本质：

  并不是直接把 jsx 渲染到页面上，而是 内部先转换成了 createElement 形式，再渲染的

4. 在 jsx 中混合写入 js 表达式

  在 jsx 语法中，要把 JS代码写到 {} 中。

  什么情况下需要使用 {} 呢?当我们需要在 Jsx 控制的区域内写Js表达式了,则需要把Js代码写到{}中

● 渲染数字

```
let num = 1997;
```

```
ReactDOM.render(<div>{num + '1009'}</div>, document.getElementById('app'));
```

● 渲染字符串

```
let str = "你好, Smile";
```

```
ReactDOM.render(<div>{str}</div>, document.getElementById('app'));
```

● 渲染布尔值

```
let bool = true;
```

```
ReactDOM.render(<div>
  {bool.toString()}
</div>
```

```
<hr />
```

```
{bool?"条件为真":"条件为假"}
```

```
</div>, document.getElementById('app'));
```

- 为属性绑定值

```
let title = 'This is a h3';
```

```
ReactDOM.render(<div>
```

```
  <h3 title={title}>独自走在繁华的都市，擦身而过的是熙熙攘攘的人群。路灯拉长孤独的身影，热  
  都是旁人的，你总是一个人...</h3>
```

```
</div>, document.getElementById('app'));
```

- 渲染jsx元素

```
const myH3 = <h3>独自走在繁华的都市，擦身而过的是熙熙攘攘的人群。路灯拉长孤独的身影，  
闹都是旁人的，你总是一个人...</h3>;
```

```
ReactDOM.render(<div>{myH3}</div>, document.getElementById('app'));
```

- 渲染jsx元素数组

```
const arr = [  
  <h3>斯人若彩虹，遇上方知有</h3>，  
  <h3>管他熙熙攘攘阳关道，我偏一条独木桥走到黑</h3>  
];
```

```
ReactDOM.render(<div>{arr}</div>, document.getElementById('app'));
```

- 将普通字符串数组，转为jsx数组并渲染到页面上【两种方案】

React中，需要把key添加给被forEach或map或for循环直接控制的元素

```
// React中，需要把key添加给被forEach或map或for循环直接控制的元素
```

```
const arrStr = ['魏无羡', '蓝忘机', '舅舅', '师姐', '小天使', '夷陵老祖'];  
const nameArr = [];  
arrStr.forEach(item => {  
  const temp = <h4 key={item}>{item}</h4>;  
  nameArr.push(temp)  
});  
ReactDOM.render(<div>{nameArr}</div>, document.getElementById('app'));
```

```
const arrStr = ['魏无羡', '蓝忘机', '舅舅', '师姐', '小天使', '夷陵老祖'];  
  
ReactDOM.render(<div>  
  {arrStr.map(item => <div key={item}><h5>{item}</h5></div> )}  
</div>, document.getElementById('app'));
```

在 jsx 中写注释：推荐使用`{/* 这是注释 */}`，eg:

```
{/*arr*/}
{
  // 这是注释，你看不见我
}
```

5. 为 jsx 中的元素添加class类名：需要使用`className`来替代`class`；`html` or 替换`label`的`for`属性

`class`和`for`为js中的关键字

6.在JSX创建DOM的时候，所有的节点，必须有唯一的根元素进行包裹；

7.在 jsx 语法中，标签必须 成对出现，如果是单标签，则必须自闭和！

当 编译引擎，在编译JSX代码的时候，如果遇到了`<`那么就把它当作 HTML代码去编译，如果遇到了`}`就把 花括号内部的代码当作 普通JS代码去编译；