

# 夏夏的react教程

## 1. 区别

react和jquery到底有什么区别呢？

jquery: 命令式的操作元素。 react: 声明时的描述元素排列。

## 2. 举个例子

假定有一个tab页，tab页可以切换，我们来看一下这个tab页在这两个库下是如何做的。

首先来看jquery式的操作：

html:

```
<div id="app"></div>
```

javascript:

```
var tpl = '<ul class="tab">'
+ '<li class="tab-item active">足球</li>'
+ '<li class="tab-item">篮球</li>'
+ '</ul>';
var html = template.compile(tpl)();
$('#app').append(html);
$('.tab').on('click', '.tab-item', function() {
  var $self = $(this);
  $self.siblings().removeClass('active');
  $self.addClass('active');
});
```

总结：一般来说，都会借助一个模板工具，生成自己想要的html，然后将html插入到文档中。最后对元素绑定事件，根据点击的元素来增加class。

然后我们再来看一下react式的操作：

html:

```
<div id="app"></div>
```

javascript:

```
import React, { Component } from 'react';
import ReactDOM from 'react-dom';
```

```
class Tab extends Component {
  state = {
    active: 'football'
  }
  tabChange = (tabName) => {
    this.setState({
      activeName: tabName
    })
  }
  render() {
    const activeName = this.state.activeName;
    const footballClassName = activeName === 'football' ? 'tab-item
active' : 'tab-item';
    const basketballClassName = activeName === 'basketball' ? 'tab-item
active' : 'tab-item'
    return (
      <ul className="tab">
        <li onClick={() => this.tabChange('football')} className=
{footballClassName}>足球</li>
        <li onClick={() => this.tabChange('basketball')} className=
{basketballClassName}>篮球</li>
      </ul>
    )
  }
}

ReactDOM.render(<Tab/>, document.getElementById('#app'));
```

总结：react会先描述出这个结构的样子，比如这个例子中看到的，两个li标签，并且他们的状态，是根据一个activeName字段来判断的，当activeName字段为football的时候，足球的标签被激活，当activeName字段为basketball的时候，篮球的标签被激活。active这个class不是根据点击结果，我们自己判断去加上的。而是当tab被点击之后，我们只去更新active这个字段的值，至于是不是带有active这个class，react重新render的时候，会根据activeName字段来判断，哪个标签需要加上active。

### 3. 变化分析

jquery版本：

最终的完整html很容易想到：

```
<div id="app">
  <ul class="tab">
    <li class="tab-item active">足球</li>
    <li class="tab-item">篮球</li>
  </ul>
</div>
```

当我们点击**篮球**这个标签后，首先判断出你点击的是篮球这个标签，然后移除所有的active的类名。也就是说，这时候html会变成这样：

```
<div id="app">
  <ul class="tab">
    <li class="tab-item">足球</li>
    <li class="tab-item">篮球</li>
  </ul>
</div>
```

然后再根据你点击的tab，给他加上active。

```
<div id="app">
  <ul class="tab">
    <li class="tab-item">足球</li>
    <li class="tab-item active">篮球</li>
  </ul>
</div>
```

从而得到最终的结果。

react版本：

最终的完整html很容易想到：

```
<div id="app">
  <ul class="tab">
    <li class="tab-item active">足球</li>
    <li class="tab-item">篮球</li>
  </ul>
</div>
```

仍然是这样一个结构，但是这只是表面看到的，react内部还维护了一个变量activeName。所以完整的来说是这样：

```
<div id="app">
  <ul class="tab">
    <li class="tab-item active">足球</li>
    <li class="tab-item">篮球</li>
  </ul>
</div>
```

```
state = {
  activeName: 'football'
}
```

这时候呢，当我们点击了篮球这个标签，我们没有做class上面的任何变更，我们只是将state中的activeName从football改成了basketball，然后立刻重新渲染。也就是之前的这个ul标签，整个替换掉，这时候会生成一个新的html，而这个新的html，正是我们从期望的结果，并且仍然带着state，结果如下

```
<div id="app">
  <ul class="tab">
    <li class="tab-item">足球</li>
    <li class="tab-item active">篮球</li>
  </ul>
</div>
```

```
state = {
  activeName: 'basketball'
}
```