React 官方文档

特点

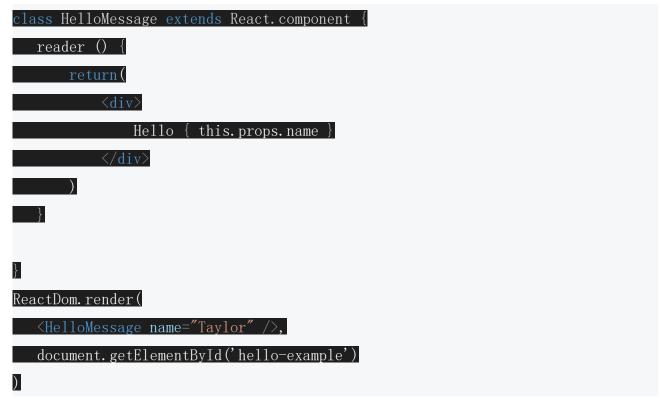
- 声明式
- 组件化
- 一次学习, 随处编写

原生js

- class / extends / constructor /supeer
- bind

简单组件

reander() 方法返回vitural-dom, this.props接受外部传入的数据.
使用 React 的时候也可以不使用 JSX 语法。尝试使用 Babel REPL, 了解 JSX 被编译成原 生 JavaScript 代码的步骤。



有状态组件

```
class Timer extends React.component {
 constructor (props) {
super(props);
this.state = { seconds: 0 }
tick () {
this.setState(state => {
  seconds: state.seconds + 1
})
componentDidMount () {
this.interval = setInterval(() => this.tick, 1000)
componentWillUnMount () {
 cleanInterval(this.interval)
render () {
return (
<div>
Seconds: {this.state.seconds}
</div>
ReactDOM.render(
<Timer />,
document.getElementById('timer-example')
)
```

TODO-应用

```
class TodoApp extends React.component {
 constructor(props) {
 super (props)
 <u>this.state = {}</u>
 this. handleChange = this. handleChange. bind(this)
 this. handleSubmit = this. handleSubmit. bind(this)
render() {
 return (
<div>
 <h3>TODO</h3>
</div>
}
JSX
JSX: JavaScript + XML
// no JSX
const heading = <hl classNmae="site-heaing">Hello, React!</hl>
// use JSX
const heading = React.createElement('h1', { className: 'site-heading'}, 'Hello,
React!')
简单组件
const SimpleComponent = () => {
 return <div>Example</div>
}
```

类组件

class ClassComponent extends Component {
 render() {
 return <div>Example</div>
 }
}