

Review ReactJS

Isumi
Batam, Jan 2020

Material:

- **ReactJS Basic**
- **ReactJS Lifecycle**
- **ReactJS Networking**
- **ReactJS Debugging**

QA..?

:: NOTES ::

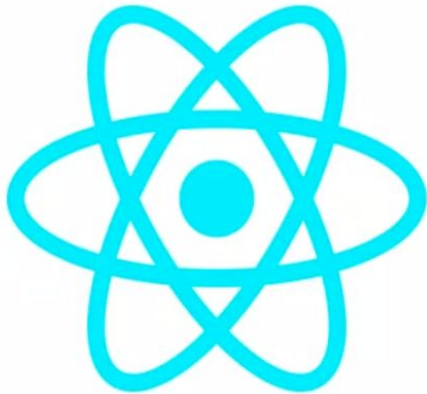
HTML

Declarative for static content

React

Declarative for dynamic data

For more details: <https://jscomplete.com/learn/why-react>



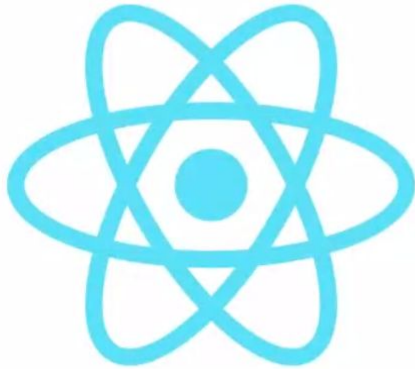
The “virtual” browser (vs. DOM API)

“Just JavaScript”

React Native (for the win)

Battle-tested

Declarative language (model UI and state)



1) Components

- Like functions
- Input: props, state | Output: UI
- Reusable and composable
- `<Component />`
- Can manage a private state

2) Reactive updates

- React will react
- Take updates to the browser

3) Virtual views in memory

- Generate HTML using JavaScript
- No HTML template language
- Tree reconciliation

React Components

Function Component

Class Component

For more details: <https://reactjs.org/docs/components-and-props.html#function-and-class-components>

JSX Is NOT HTML

```
class Hello extends React.Component {  
  render () {  
    return (  
      <div className="container">  
        <h1>Getting Started</h1>  
      </div>  
    );  
  }  
}  
  
ReactDOM.render(<Hello />, mountNode);
```

```
class Hello extends React.Component {  
  render () {  
    return (  
      React.createElement("div", { className: "container"},  
        React.createElement("h1", null, "Getting Started")  
      )  
    );  
  }  
}  
  
ReactDOM.render(React.createElement(Hello, null), mountNode);
```

Explore:

- **ReactJS Router:** <https://reacttraining.com/react-router/web/guides/quick-start>
- **ReactJS Navigation:** <https://reactnavigation.org/docs/en/navigating.html>
- **ReactJS UI Framework:**
<https://hackernoon.com/23-best-react-ui-component-libraries-and-frameworks-250a81b2ac42>

CODE REVIEW