REACT CHEATSHEET

INSTALLATION

React

```
<script src="https://unpkg.com/react@15/dist/react.js"></script>
$ $ npm install react --save
$ $ bower install react --save
```

React DOM

```
<script src="https://unpkg.com/react-dom@15/dist/react-dom.js"></script>
$ npm install react-dom
$ bower install react-dom --save
```

RENDERING

ES5

```
ReactDOM.render(
  React.createElement(
    Link,
    {href: 'https://Node.University'}
  ),
  document.getElementById('menu')
)
```

ES5+JSX

```
ReactDOM.render(
  <Link href='https://Node.University'/>,
  document.getElementById('menu')
)
```

Server-side rendering

```
const ReactDOMServer = require('react-dom/server')
ReactDOMServer.renderToString(Link, {href: 'https://Node.University'})
ReactDOMServer.renderToStaticMarkup(Link, {href:
'https://Node.University'})
```

propTypes

Types available under React.PropTypes:
any array bool element func
node number object string
To make a property required (warning only), append .isRequired.

More methods

```
instanceOf(constructor)
oneOf(['News', 'Photos'])
oneOfType([propType, propType])
```

Custom validation

```
propTypes: {
  customProp: function(props, propName, componentName) {
    if (!/regExPattern/.test(props[propName])) {
      return new Error('Validation failed!');
    }
  }
}
```

SPECIAL PROPERTIES

key—Unique identifier for an element to turn arrays/lists into hashes for better performance. For example: key={id}.

ref—Reference to an element via this.refs.NAME. For example: ref="email" will create a this.refs.email DOM node or ReactDOM.findDOMNode(this.refs.email).

style—Accepts an object for camelCased CSS styles instead of a string (immutable since v0.14). For example: style={{color: red}}.

className—HTML class attribute. For example: className="btn".

htmlFor—HTML for attribute. For example: htmlFor="email".

dangerouslySetInnerHTML—Sets inner HTML to raw HTML by providing an object with the key html.

children—Sets the content of the element via this.props.children. For example: this.props.children[0].

data-NAME—Custom attribute. For example: data-tooltip-text="...".

LIFECYCLE EVENTS

```
componentWillMount function()
componentDidMount function()
componentWillReceiveProps function(nextProps)
shouldComponentUpdate function(nextProps, nextState)  -> bool
componentWillUpdate function(nextProps, nextState)
componentDidUpdate function(prevProps, prevState)
componentWillUnmount function()
```

COMPONENT PROPERTIES AND METHODS

Properties

this.refs—Lists components with a ref property.

this.props—Lists any properties passed to an element (immutable).

this.state—Lists states set by setState and getInitialState (mutable). Avoid setting state manually with this.state=...

this.isMounted—Flags whether the element has a corresponding DOM node.

Methods

setState(changes)—Changes state (partially) to this.state, and triggers a rerender

replaceState(newState)—Replaces this.state, and triggers a rerender

forceUpdate()—Triggers an immediate DOM rerender

REACT ADDONS

As npm modules:

```
react-addons-css-transition-group
react-addons-perf
react-addons-test-utils
react-addons-pure-render-mixin
react-addons-linked-state-mixin
react-addons-clone-with-props
react-addons-create-fragment
react-addons-css-transition-group
react-addons-linked-state-mixin
react-addons-pure-render-mixin
react-addons-shallow-compare
react-addons-transition-group
react-addons-update
```