



## INSTALLATION

### React

```
<script src="https://fb.me/react-0.14.0.js"></script>
```

```
$ npm install react --save
```

```
$ bower install react --save
```

### React DOM

```
<script src="https://fb.me/react-dom-0.14.0.js"></script>
```

```
$ npm install react-dom
```

```
$ bower install react-dom --save
```

## RENDERING

### Rendering (ES5)

```
ReactDOM.render(  
  React.createElement(Link, {name: 'HackHall.com'}),  
  document.getElementById('menu')  
)
```

### Rendering (ES5+JSX)

```
ReactDOM.render(  
  <Link name='HackHall.com' />,  
  document.getElementById('menu')  
)
```

### Server-side Rendering

```
var ReactDOMServer = require('react-dom/server')  
ReactDOMServer.renderToString(Link, {name: 'HackHall.com'})  
ReactDOMServer.renderToStaticMarkup(Link, {name: 'HackHall.com'})
```

## COMPONENTS

### ES5

```
var Link = React.createClass({  
  displayName: 'Link',  
  render: function() {  
    return React.createElement('a', {className: 'btn', title:  
this.props.name}, 'Click ->', this.props.name);  
  }  
});
```

### ES5 + JSX

```
var Link = React.createClass({  
  render: function() {  
    return <a className='btn' title={this.props.name}>Click ->  
this.props.name</a>  
  }  
});
```

### ES6 + JSX

```
class Link extends React.Component {  
  render() {  
    return <a className='btn' title={this.props.name}>Click ->  
this.props.name</a>  
  }  
}
```

## LIFECYCLE EVENTS

```
componentWillMount function()
```

```
componentDidMount function()
```

```
componentWillReceiveProps function(nextProps)
```

```
shouldComponentUpdate function(nextProps, nextState)-> bool
```

```
componentWillUpdate function(nextProps, nextState)
```

```
componentDidUpdate function(prevProps, prevState)
```

```
componentWillUnmount function()
```

## ADVANCED COMPONENTS

### Options (ES5)

`propTypes` `object` : Type validation in development mode

`getDefaultProps` `function()` : object of default props

`getInitialState` `function()` : object of the initial state

### ES5:

```
var Link = React.createClass({  
  propTypes: { name: React.PropTypes.string },  
  getDefaultProps: function() {  
    return { initialCount: 0 }  
  },  
  getInitialState: function() {  
    return {count: this.props.initialCount};  
  },  
  tick: function() {  
    this.setState({count: this.state.count + 1});  
  },  
  render: function() {  
    return React.createElement(  
      'a',  
      {className: 'btn', href: '#', title: this.props.name, onClick:  
this.tick.bind(this)},  
      'Click ->',  
      (this.props.name? this.props.name : 'webapplog.com'),  
      (Clicked: '+this.state.count+')  
    );  
  }  
});
```

### ES5 + JSX:

```
var Link = React.createClass({  
  propTypes: { name: React.PropTypes.string },  
  getDefaultProps: function() {  
    return { initialCount: 0 }  
  },  
  getInitialState: function() {  
    return {count: this.props.initialCount};  
  },  
  tick: function() {  
    this.setState({count: this.state.count + 1});  
  },  
  render: function() {  
    return (  
      <a onClick={this.tick.bind(this)} href="#" className="btn"  
title={this.props.name}>  
        Click -> {(this.props.name? this.props.name : 'webapplog.com')}  
        (Clicked: {this.state.count})  
      </a>  
    );  
  }  
});
```

### ES6 + JSX:

```
export class Link extends React.Component {  
  constructor(props) {  
    super(props);  
    this.state = {count: props.initialCount};  
  }  
  tick() {  
    this.setState({count: this.state.count + 1});  
  }  
  render() {  
    return (  
      <a onClick={this.tick.bind(this)} href="#" className="btn"  
title={this.props.name}>  
        Click -> {(this.props.name? this.props.name : 'webapplog.com')}  
        (Clicked: {this.state.count})  
      </a>  
    );  
  }  
}  
Link.propTypes = { initialCount: React.PropTypes.number };  
Link.defaultProps = { initialCount: 0 };
```