



React Without JSX

JSX is not a requirement for using React. Using React without JSX is especially convenient when you don't want to set up compilation in your build environment.

Each JSX element is just syntactic sugar for calling `React.createElement(component, props, ...children)`. So, anything you can do with JSX can also be done with just plain JavaScript.

For example, this code written with JSX:

```
class Hello extends React.Component {
  render() {
    return <div>Hello {this.props.toWhat}</div>;
  }
}

ReactDOM.render(
  <Hello toWhat="World" />,
  document.getElementById('root')
);
```

can be compiled to this code that does not use JSX:

```
class Hello extends React.Component {
  render() {
    return React.createElement('div', null, `Hello ${this.props.toWhat}`);
  }
}

ReactDOM.render(
  React.createElement(Hello, {toWhat: 'World'}, null),
  document.getElementById('root')
);
```



If you're curious to see more examples of how JSX is converted to JavaScript, you can try out [the online Babel compiler](#).

The component can either be provided as a string, or as a subclass of `React.Component`, or a plain function for stateless components.

If you get tired of typing `React.createElement` so much, one common pattern is to assign a shorthand:

```
const e = React.createElement;

ReactDOM.render(
  e('div', null, 'Hello World'),
  document.getElementById('root')
);
```

If you use this shorthand form for `React.createElement`, it can be almost as convenient to use React without JSX.

Alternatively, you can refer to community projects such as [react-hyperscript](#) and [hyperscript-helpers](#) which offer a terser syntax.

DOCS

Installation

Main Concepts

Advanced Guides

API Reference

Contributing

FAQ

CHANNELS

GitHub [↗](#)

Stack Overflow [↗](#)

Discussion Forum [↗](#)

Reactiflux Chat [↗](#)

DEV Community [↗](#)

Facebook [↗](#)

Twitter 

COMMUNITY

Community Resources

Tools

MORE

Tutorial

Blog

Acknowledgements

React Native 

Copyright © 2018 Facebook Inc.