

Stack Overflow is a community of 7.7 million programmers, just like you, helping each other. Join them; it only takes a minute:

Sign up 

## What is the meaning of `{...this.props}` in Reactjs

What is the meaning of

```
{...this.props}
```

I am trying to use it like that

```
<div {...this.props}> Content Here </div>
```

reactjs

asked Feb 11 '15 at 10:47



Swaraj ghosh

434 2 6 16

### 4 Answers

It's called [spread attributes](#) and it's aims for making passing of props easier.

Let us imagine that you have a component that accepts N number of properties. Passing these down can be tedious and unwieldy if the number grows.

```
<Component x={} y={} z={} />
```

so instead you do this, wrap them up in an object and use the spread notation

```
var props = { x: 1, y: 1, z:1 };
<Component {...props} />
```

which will unpack it into the props on your component, i.e., you "never" use `{... props}` inside your `render()` function, only when you pass the props down to another component. Use your unpacked props as normal `this.props.x`.

edited May 27 at 4:30

answered Feb 11 '15 at 10:50



Henrik Andersson

20.8k 6 61 74

1 Just to add, it can help to think of it as a replacement to `this.transferPropsTo` which was deprecated in React 0.12.x and will be removed in 0.13.x. It of course allows much more advanced usage however simply translating React 0.11.x's `this.transferPropsTo(<Foo />)` to `<Foo {...this.props} />` is most useful for people making that transition. – [Mike Driver](#) Feb 11 '15 at 14:53

Good clarification, thanks mate :) – [Henrik Andersson](#) Feb 11 '15 at 15:04

It is ES-6 feature. It means you extract all the properties of props in `div`. `{...}` operator is used to extract properties of object.

answered Apr 11 at 5:34



Mayank Soni

1

### It's ES6 `Spread_operator` & `Destructuring_assignment`

[https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/Spread\\_operator](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/Spread_operator)

[https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/Destructuring\\_assignment](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/Destructuring_assignment)

```
<div {...this.props}> Content Here </div>
```

it's equal to

### class component

REPL: <https://jscomplete.com/repl/>

```
const person = {
  name: "xgqfrms",
  age: 23,
  country: "China"
};

class TestDemo extends React.Component {
  render() {
    const {name, age, country} = {...this.props};
    // const {name, age, country} = this.props;
    return (
      <div>
        <h3> Person Information: </h3>
        <ul>
          <li>name={name}</li>
          <li>age={age}</li>
          <li>country={country}</li>
        </ul>
      </div>
    );
  }
}

ReactDOM.render(
  <TestDemo {...person}/>
  , mountNode
);
```

Run code snippet

Expand snippet

```
const person = {
  name: "xgqfrms",
  age: 23,
  country: "China"
};

class TestDemo extends React.Component {
  render() {
    const {name, age, country} = {...this.props};
    // const {name, age, country} = this.props;
    return (
      <div>
        <h3> Person Information: </h3>
        <ul>
          <li>name={name}</li>
          <li>age={age}</li>
          <li>country={country}</li>
        </ul>
      </div>
    );
  }
}

ReactDOM.render(
  <TestDemo {...person}/>
  , mountNode
);
```

CSS

Person Information:

- name=xgqfrms
- age=23
- country=China

## function component

REPL: <https://jscomplete.com/repl/>

```
const props = {
  name: "xgqfrms",
  age: 23,
  country: "China"
};

const Test = (props) => {
  return(
    <div>
      name={props.name}
      age={props.age}
      country={props.country}>
      Content Here
    <ul>
      <li>name={props.name}</li>
      <li>age={props.age}</li>
      <li>country={props.country}</li>
    </ul>
    </div>
  );
};

ReactDOM.render(
  <div>
    <Test {...props}/>
    <hr/>
    <Test
      name={props.name}
      age={props.age}
      country={props.country}
    />
  </div>
  , mountNode
);
```

Run code snippet

Expand snippet

```
const props = {
  name: "xgqfrms",
  age: 23,
  country: "China"
};

const Test = (props) => {
  return(
    <div>
      name={props.name}
      age={props.age}
      country={props.country}>
      Content Here
      <ul>
        <li>name={props.name}</li>
        <li>age={props.age}</li>
        <li>country={props.country}</li>
      </ul>
    </div>
  );
};

ReactDOM.render(
  <div>
    <Test {...props}/>
  </div>
  , mountNode
);
```

Content Here

- name=xgqfrms
- age=23
- country=China

CSS

for more details, seeing the following links:

- <https://babeljs.io/docs/plugins/transform-object-rest-spread/>
- <https://github.com/gildata/RAIO/issues/136#issuecomment-327361743>
- <https://facebook.github.io/react/docs/components-and-props.html>

edited Sep 6 at 5:26

answered Sep 6 at 3:35  
user8202629


When we use our defined components, we can add attributes called props. These attributes are available in our component as this.props and can be used in our render method to render dynamic data:

```
var MyComponent = React.createClass({
  render: function(){
    return (
      <h1>Hello, {this.props.name}!</h1>
    );
  }
});


React.render(<MyComponent name="Handsome" />, document.getElementById('myDiv'));
```

also you can checkout here [reactjs](#)

edited Dec 23 '15 at 6:26

 [Serjik](#)  
3,381 5 34 53

answered Dec 23 '15 at 6:17

 [Paddy lock](#)  
5 1