React Props

In this lesson, we learn to modify our components and make them more dynamic using props, which is a fundamental concept in React.

What are Props?

Props are like arguments you can pass to a function. Imagine them to be options that can be passed to a component to customize its functionality. For example, the following will always print 'Welcome to Rect', but if you want it to be able to print whatever greeting you want, you can use props.

```
import React from 'react';
require('./style.css');

import ReactDOM from 'react-dom';
import App from './app.js';

ReactDOM.render(
  <App />,
  document.getElementById('root')
);
```

The code block below uses props to pass a greeting. Two greetings are passed as props on *Line 8* of the index.js file where `greeting` and `anotherGreeting` are variables that could be named anything. They are used on *lines 5 and 8* of the app.js file. Essentially, you can access whatever prop you define when using the component you as `this.props.nameGivenToProp` if used like `<componentName nameGivenToProp = 'prop value' secondProp = 'second value' />`.

You can also pass Numbers, Functions & javascript expressions to your prop by using {}. As on *line 24*.

```
import React from 'react';
require('./style.css');

import ReactDOM from 'react-dom';
import App from './app.js';

ReactDOM.render(
```

```
    <App greeting = "greeting #1" anotherGreeting = "greeting #2"/>,
    document.getElementById('root')
);
```

Now you know how props work. In the next lesson, we'll discuss state in React.