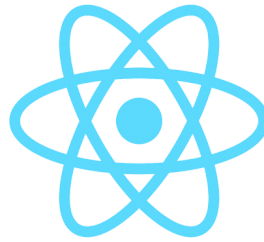


[Topics](#)[Pro](#)[About](#)[Sign in with Github](#)
or via email

Get
exclusive
content,
resources,
and
more!

you(

Help
spread
the
word
about
this
tutorial!



Changing State via User Input

Learn how to capture data from an input field and update a component's state.



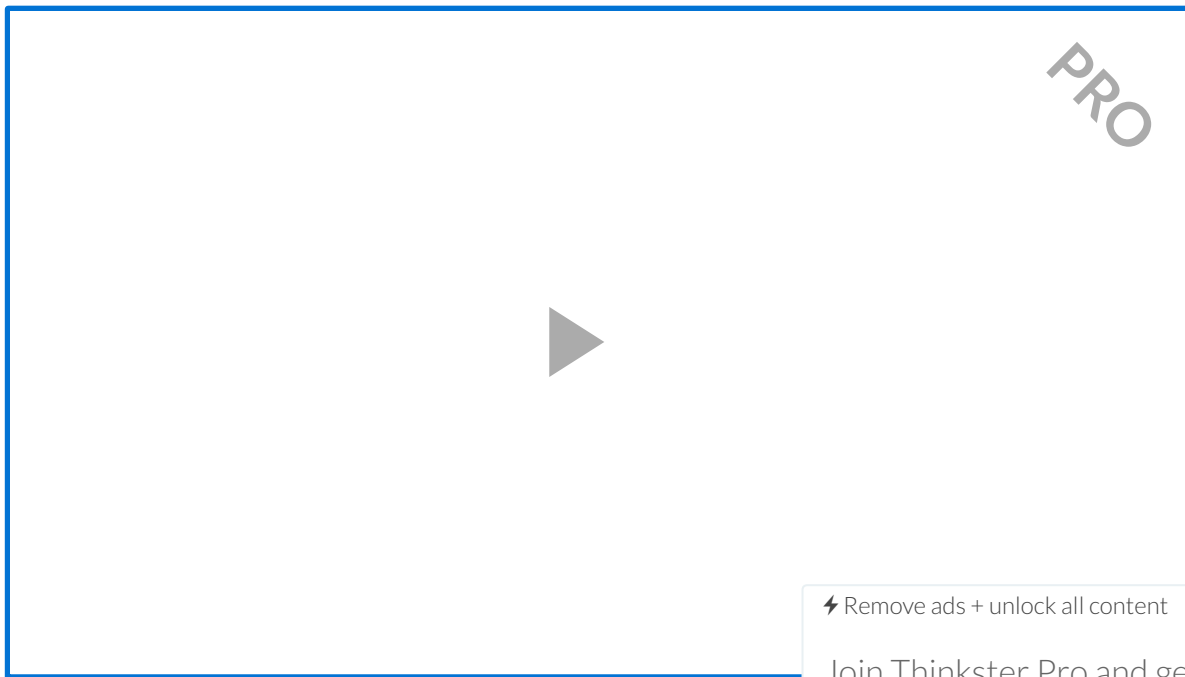
Eric Simons



Up to date (Aug '17)



React



⚡ Go Pro

Unlock all content & remove ads

⚡ Remove ads + unlock all content

AD

[Join Thinkster Pro and get
videos, source code, and no ads](#)

[Topics](#)[Pro](#)[About](#)[Sign in with Github](#)[or via email](#)

Get
exclusive
content,
resources,
and
more!

Help
spread
the
word
about
this
tutorial!



SendGrid

Delivery. Start sending
for Free with a 5-min
Integration.

Join
our
newsletter!



SUBSCRIBE

1-2 emails
per week,
no spam

⚡ Remove ads + unlock all content

AD

[Join Thinkster Pro and get
videos, source code, and no ads](#)

[Topics](#)[Pro](#)[About](#)[Sign in with Github](#)[or via email](#)

Get
exclusive
content,
resources,
and
more!

you(

Help
spread
the
word
about
this
tutorial!



⚡ Remove ads + unlock all content

AD

[Join Thinkster Pro and get
videos, source code, and no ads](#)



framing the problem

Lets start with [the code](#) from our [last post](#):

```
var Water = React.createClass({
  getInitialState: function() {
    return {
      currentTemp: 10
    };
  },

  render: function() {
    // empty variable that will hold either "Liquid", "Solid", or "Gas"
    var stateOfMatter;

    // If temp is on/below freezing, it's a solid
    if (this.state.currentTemp <= 32) {
      stateOfMatter = 'Solid';

    // if temp is on/above boiling, it's a gas
    } else if (this.state.currentTemp >= 212) {
      stateOfMatter = 'Gas';

    // otherwise it's just a liquid
    } else {
      stateOfMatter = 'Liquid';
    }

    return (
      <div>
        <p>At { this.state.currentTemp }°F, water is considered to be a "{ stateOfMatter }"
      </div>
    );
  }
});
```

[⚡ Remove ads + unlock all content](#)

AD

[Join Thinkster Pro and get videos, source code, and no ads](#)



create an input field where you can simply type in a temperature and have it update in real time. To do this, we'll need to figure out two things:

- How do you capture data from input fields in React components?
- How do you update the state of a React component?

Capturing data from an input field

To help standardize cross browser (and platform) development, React provides a handful of helper components and methods for interacting with forms/inputs. For our use case, the React docs show how you can create a “controlled component” input that uses the “onChange” property to pass along the input’s value (amongst other things) to another method.

Lets implement this in our code. In render's return statement, lets add an input field whose value is tied to `this.state.currentTemp` and onChange calls a method called `setTemperature`:

```
return (  
  <div>  
    <input type="text" onChange={ this.setTemper  
    <p>At { this.state.currentTemp }°F, water is  
  </div>  
);
```

⚡ Remove ads + unlock all content

Join Thinkster Pro and get
videos, source code, and no ads