**Stateless Components Inherit From Stateful Components**

Let's learn our first *programming pattern!*

In this lesson, we'll take a look at a simple version of a programming pattern. The following lessons will expand upon that lesson, and by the end, we'll have a programming pattern in its full complexity.

Our programming pattern uses two React components: a *stateful* component, and a *stateless* component. "Stateful" describes any component that has a state property; "stateless" describes any component that does not.

In our pattern, a *stateful* component passes its state down to a *stateless* component.

Click Next to walk through an example!

# Build a Stateful Component Class

Let's make a stateful component pass its stateto a stateless component.

To make that happen, you need two component classes: a stateful class, and a stateless class.

# Build a Stateless Component Class

Great! You just made a stateful component class named Parent.

Now, let's make our stateless component class.

# Pass a Component's State

A <Parent /> is supposed to pass its state to a <Child />.

Before a <Parent /> can pass anything to a <Child />, you need to import Child into **Parent.js**.

# Don't Update props

Great work! You just passed information from a stateful component to a stateless component. You will be doing a lot of that.

You learned earlier that a component can change its state by calling this.setState(). You may have been wondering: how does a component change its props?

The answer: it doesn't!

A component should never update this.props. Look at **Bad.js** to see an example of what not to do.

This is potentially confusing. props and statestore dynamic information. Dynamic information can change, by definition. If a component can't change its props, then what are props for?

**A React component should use props to store information that can be changed, but can only be changed by a different component.**

**A React component should use state to store information that the component itself can change.**

If that's a bit confusing, don't worry! The next two lessons will be examples.