Unit 3: State, Props, Props Validation

State

State is the place where the data comes from. You should always try to make your state as simple as possible and minimize number of stateful components. If you have, for example, ten components that need data from the state, you should create one container component that will keep the state for all of them.

State

```
import React from 'react';

class App extends React.Component {
   constructor(props) {
      super(props);

      this.state = {
         header: "Header from state...",
         "content": "Content from state..."
      }
   }

   render() {
      return (
         <div>
            <h1>{this.state.header}</h1>
            <h2>{this.state.content}</h2>
         </div>
      );
   }
}

export default App;
```

Props

The main difference between state and props is that **props** are immutable. This is why container component should define state that can be updated and changed, while the child components should only pass data from the state using props.

2 ways to define props:

- default Props:
- pass to children components from parent components

Props

set defautProps:

```
import React from 'react';

class App extends React.Component {
   render() {
      return (
         <div>
            <h1>{this.props.headerProp}</h1>
            <h2>{this.props.contentProp}</h2>
         </div>
      );
   }
}

App.defaultProps = {
   headerProp: "Header from props...",
   contentProp:"Content from props..."
}

export default App;
```

Props

pass to children components from parent components (eg in the code sample)

Props Validation

(eg in the code sample)