Klasy:

Komponent:

function ContactItem({ login, name, department }) {

const imgUrl = `https://api.adorable.io/avatars/55/${login}.png`;

return (

<li className="item">

<img src={imgUrl} className="ui mini rounded image" />

<div className="content">

<h4 className="header">{name}</h4>

<div className="description">{department}</div>

</div>

</li>

);

}

Klasa:

class ContactItem extends React.Component {

render() {

const { login, name, department } = this.props

const imgUrl = `https://api.adorable.io/avatars/55/${login}.png`;

return (

<li className="item">

<img src={imgUrl} className="ui mini rounded image" />

<div className="content">

<h4 className="header">{name}</h4>

<div className="description">{department}</div>

</div>

</li>

);

}

}

Zdarzenia:

function MyComponent() {

return (

<button onClick={() => alert('Kliknięto!')}>Kliknij!</button>

);

}

function onClickHandler() {

alert('Kliknięto!');

}

function MyComponent() {

return (

<button onClick={onClickHandler}>Kliknij!</button>

);

}

class App extends React.Component {

render() {

return <button onClick={this.onClickHandler}>Kliknij!</button>;

}

onClickHandler() {

alert("Kliknięto!");

}

}

STATE

Do przechowywania stanu komponentu. Przykład licznik:

class App extends React.Component {  
 constructor() {  
 super();  
 this.state = {counter: 0};  
 }  
   
 render() {  
 return (  
 <div>  
 <button onClick={this.increment.bind(this)}>+</button>  
 <output>{this.state.counter}</output>  
 <button onClick={this.decrement.bind(this)}>-</button>  
 </div>  
 );  
 }  
   
 increment() {  
 this.setState({  
 counter: this.state.counter + 1  
 })  
 }  
   
 decrement() {  
 this.setState({  
 counter: this.state.counter - 1  
 })  
 }  
}  
ReactDOM.render(<App />, document.getElementById("app"));