

REACT - FORMULÁŘE

Defaultní akce při odeslání formuláře je přesměrovat uživatele na jinou stránku. V Reactu se nám ale často hodí zpracovávat formuláře jinak.

```
function handleSubmit(e) {  
  // formulář se neodešle a uživatel nebude přesměrován na jinou stránku  
  e.preventDefault();  
}
```

CONTROLLED KOMPONENTY

V HTML mají input elementy jako je input, textarea, nebo select vlastní stav (anglicky state). V Reactu má komponenta také vlastní state. Tyto dva stavy můžeme zkombinovat a nechat tak state komponenty být jediným zdrojem pravdy.

Pokud má input stejnou hodnotu jako state a při jeho změně se state mění, jedná se o controlled komponentu.

```
class NameForm extends React.Component {  
  /* ... */  
  
  handleChange(e) {  
    // když se změní input, tak se změní  
    // i state  
    this.setState(  
      {value: e.target.value}  
    );  
  }  
  
  render() {  
    return (  
      <form onSubmit={this.handleSubmit}>  
        <label>Jméno: </label>  
        <input type="text"  
          value={this.state.value}  
          onChange={this.handleChange}/>  
        <input type="submit"  
          value="Odeslat"/>  
      </form>  
    );  
  }  
}
```

ZPRACOVÁNÍ VÍCE INPUTŮ

Nemusíme mít pro kontrolování každého inputu samostatnou metodu. Můžeme mít třeba něco takového:

```
handleInputChange(e) {  
  const target = e.target;  
  const value = (  
    target.type === 'checkbox'  
    ? target.checked  
    : target.value  
  );  
  const name = target.name;  
  
  this.setState({  
    [name]: value  
  });  
}
```

ATRIBUT VALUE

Některým inputům se v HTML nastavují hodnoty pomocí atributu value a některým jiným způsobem. Inputu textarea se například v HTML nastavuje hodnota dovnitř uzavíracího a otevíracího tagu. V Reactu se hodnota inputu textarea nastavuje pomocí atributu value. Stejně je to například i u selectu.

```
<textarea  
  value={this.state.value}  
  onChange={this.handleChange}  
>
```