

Le gestionnaire d'événement du tag `<input>` `onChange` prend un objet javascript qui effectue une fonction fléchée qui retourne une modification de l'état du composant.

En suite le paramètre `'value='` est valorisé par le nouvelle état du composant.

```
JS index.js JS App.js JS Form.js x JS ItemList.js
1  import React from 'react';
2
3  class Form extends React.Component{
4
5      state = {
6          name: ' ',
7          quantity: 0
8      };
9
10     render(){
11         return(
12             <div>
13                 <h3>Ajouter des articles à acheter</h3>
14                 <form>
15                     <input type="number" placeholder="quantité"
16                         value={this.state.quantity} onChange={(oEvt)=> this.setState({quantity: oEvt.target.value})}/>
17                     <input type="text" placeholder="article" value=""
18                         value={this.state.name} onChange={(oEvt)=> this.setState({name: oEvt.target.value})}/>
19                     <button type="submit">ajouter</button>
20                 </form>
21             </div>
22         )
23     }
24 }
25
26 export default Form;
27
```

Si besoin dans une fonction on peut réinitialiser l'état du composant a sa valeur initial, dans le cas ci dessous après l'ajout d'un article dans un tableau item on remet le champs 'name' et 'quantity' à vide ce qui permet à 'REACT' de remettre les valeurs à vide au niveau de la vue, car les champs value des <input> sont lié a l'état du composant form.

```
<form onSubmit={this._handleSubmit} >
  <input type="number" placeholder="quantité"
    value={this.state.quantity} onChange={(oEvt) => this.setState({quantity: oEvt.target.value})}/>
  <input type="text" placeholder="article" value=""
    value={this.state.name} onChange={(oEvt) => this.setState({name: oEvt.target.value})}/>
  <button type="submit">ajouter</button>
</form>
```

```
JS App.js JS Form.js x
1 import React from 'react';
2
3 class Form extends React.Component{
4
5   state = {
6     name: '',
7     quantity: 0
8   };
9
10  _handleSubmit = (oEvt) =>{
11    oEvt.preventDefault();
12    console.log(this.state);
13    this.props._addArticle(this.state);
14    this.setState({name: '', quantity: 0});
15  }
```

Liste de course en construction

Ajouter un article



0 article ajouter

The image shows a web form with two input fields. The first field contains the number '0' and the second field contains the word 'article'. To the right of these fields is a button labeled 'ajouter'. A red arrow points from the '0' in the first input field to the 'this.setState({name: '', quantity: 0});' line in the code block on the left. Another red arrow points from the 'article' in the second input field to the 'this.setState({name: '', quantity: 0});' line in the same code block.