

3. modification by reference

```
example.js — ~/Projects/temp
example.js
28   handleNoteAdd = (newNote) => {
29     const notes = this.state.notes;
30     notes.push(newNote);
31
32     this.setState({
33       notes
34     });
35   }
36
37   render() {
38     return (
39       <div className="app">
40         <h2 className="app__header">NotesApp</h2>
41
42         <NoteEditor onNoteAdd={this.handleNoteAdd} />
43
44         <NotesGrid
45           notes={this.state.notes}
46           onNoteDelete={this.handleNoteDelete}
47         />
48       </div>
49     );
50   }
51 }
```

↑ the same reference

3. modification by reference

```
example.js — ~/Projects/temp
example.js
10  shouldComponentUpdate(nextProps) {
11      return this.props.notes !== nextProps.notes; // false
12  }
13
14  render() {
15      const {
16          notes,
17          onDelete
18      } = this.props;
19
20      return (
21          <Masonry
22              className='grid'
23              options={MASONRY_OPTIONS}
24          >
25              {
26                  notes.map(note =>
27                      <Note
28                          key={note.id}
29                          id={note.id}
```

↑ so, shallow comparison produces 'false'