Nay tiếp tục, mình sẽ xây dựng một ứng dụng **CRUD trong React**. Với các bài viết trước ta đã hiểu biết được một phần cơ bản về**React**, nếu bạn có xem qua, thì bài viết này cũng tương đối đơn giản thôi, không quá phức tạp lắm

Tạo project React name "**crud-react**"

[view source](https://hoanguyenit.com/create-crud-in-react.html#viewSource)[print](https://hoanguyenit.com/create-crud-in-react.html#printSource)[?](https://hoanguyenit.com/create-crud-in-react.html#about)

|  |
| --- |
| npx create-react-app crud-react |
| cd crud-react |

|  |
| --- |
| npm install bootstrap@4.2.1 |

Ok, tôi giả sử đã có một mảng dữ liệu và cần phải hiển thị ra ngoài cho người dùng xem

[view source](https://hoanguyenit.com/create-crud-in-react.html#viewSource)[print](https://hoanguyenit.com/create-crud-in-react.html#printSource)[?](https://hoanguyenit.com/create-crud-in-react.html#about)

|  |
| --- |
| items:[ |
| { |

|  |
| --- |
| name:"Iphone 7 Plus",price:500.0 |
| }, |

|  |
| --- |
| { |
| name:"Iphone 8 Plus",price:1400.0 |

|  |
| --- |
| }, |
| { |

|  |
| --- |
| name:"Iphone 10",price:1600.0 |
| }, |

|  |
| --- |
| { |
| name:"Iphone 10 Pro",price:1700.0 |

|  |
| --- |
| } |
| ] |

Đồng thời tôi cần thiết lập các biến như (***index,name,price,action***)

+**index:** lưu vị trí của một item khi bạn bấm edit , default=0  
+ **name:** lưu tên của item  
+ **price:** là giá tiền của item đó  
+ **action:** là hành động "ADD ITEM" hoặc "UPDATE ITEM", default="ADD ITEM"

Ok, giờ bạn mở file **src/App.js** lên và cài đặt code như sau:

[view source](https://hoanguyenit.com/create-crud-in-react.html#viewSource)[print](https://hoanguyenit.com/create-crud-in-react.html#printSource)[?](https://hoanguyenit.com/create-crud-in-react.html#about)

|  |
| --- |
| import React from 'react'; |
| import './App.css'; |

|  |
| --- |
| import 'bootstrap/dist/css/bootstrap.css'; |
| class App extends React.Component{ |

|  |
| --- |
| constructor(props){ |
| **super**(props); |

|  |
| --- |
| **this**.state = { |
| index:0, |

|  |
| --- |
| name:"", |
| price:0, |

|  |
| --- |
| action:"ADD ITEM",//default ADD ITEM |
| items:[ |

|  |
| --- |
| { |
| name:"Iphone 7 Plus",price:500.0 |

|  |
| --- |
| }, |
| { |

|  |
| --- |
| name:"Iphone 8 Plus",price:1400.0 |
| }, |

|  |
| --- |
| { |
| name:"Iphone 10",price:1600.0 |

|  |
| --- |
| }, |
| { |

|  |
| --- |
| name:"Iphone 10 Pro",price:1700.0 |
| } |

|  |
| --- |
| ] |
| } |

|  |
| --- |
| **this**.changeName = **this**.changeName.bind(**this**); |
| **this**.changePrice = **this**.changePrice.bind(**this**); |

|  |
| --- |
|  |
| } |

|  |
| --- |
| changeName = (e)=>{ |
| **this**.setState({ |

|  |
| --- |
| name:e.target.value |
| }) |

|  |
| --- |
| } |
| changePrice = (e)=>{ |

|  |
| --- |
| **this**.setState({ |
| price:e.target.value |

|  |
| --- |
| }) |
| } |

|  |
| --- |
|  |
| addItem =()=>{ |

|  |
| --- |
| **if**(!**this**.state.items.find(item=>item.name===**this**.state.name)){ |
| **this**.setState({ |

|  |
| --- |
| items:[ |
| ...**this**.state.items, |

|  |
| --- |
| { name:**this**.state.name,price:**this**.state.price} |
| ], |

|  |
| --- |
| name:"", |
| price:"" |

|  |
| --- |
| }) |
| } |

|  |
| --- |
| } |
| Edit = (item,index)=>{ |

|  |
| --- |
| **this**.setState({ |
| action:'UPDATE ITEM', |

|  |
| --- |
| name:item.name, |
| price:item.price, |

|  |
| --- |
| index:index |
| }); |

|  |
| --- |
| } |
| updateItem = ()=>{ |

|  |
| --- |
| let data = **this**.state.items; |
| data.map((item,index)=>{ |

|  |
| --- |
| **if**(**this**.state.index===index){ |
| item.name = **this**.state.name; |

|  |
| --- |
| item.price = parseInt(**this**.state.price); |
| } |

|  |
| --- |
| }) |
| //set update items |

|  |
| --- |
| **this**.setState({ |
| items:data, |

|  |
| --- |
| name:"", |
| price:"", |

|  |
| --- |
| action:'ADD\_ITEM' |
| }) |

|  |
| --- |
|  |
| } |

|  |
| --- |
|  |
| deleteItem=(name)=>**this**.setState({ |

|  |
| --- |
| items:**this**.state.items.filter(item=>item.name!=name) |
| }) |

|  |
| --- |
| render(){ |
| **return** ( |

|  |
| --- |
| <div className="container"> |
| <div className="row"> |

|  |
| --- |
| <div className="col-md-4"> |
| <h1>{**this**.state.action}</h1> |

|  |
| --- |
| <div className="form-group"> |
| <label>Name</label> |

|  |
| --- |
| <input type="text" name="" className="form-control" onChange={**this**.changeName} value={**this**.state.name}/> |
| </div> |

|  |
| --- |
| <div className="form-group"> |
| <label>Price</label> |

|  |
| --- |
| <input type="text" name=""  className="form-control" onChange={**this**.changePrice} value={**this**.state.price}/> |
| </div> |

|  |
| --- |
| <div className="form-group"> |
| <button type="button" className="btn btn-primary"  onClick={**this**.state.action=='ADD ITEM'?**this**.addItem:**this**.updateItem}>Add</button> |

|  |
| --- |
| </div> |
| </div> |

|  |
| --- |
|  |
| <div className="col-md-8"> |

|  |
| --- |
| <h1>List Products</h1> |
| <table className="table"> |

|  |
| --- |
| <thead> |
| <tr> |

|  |
| --- |
| <th>SDT</th> |
| <th>Name</th> |

|  |
| --- |
| <th>Price</th> |
| <th>Modify</th> |

|  |
| --- |
| <th>Remove</th> |
| </tr> |

|  |
| --- |
| </thead> |
| <tbody> |

|  |
| --- |
| { |
| **this**.state.items.map((item,index)=>( |

|  |
| --- |
| <tr key={index}> |
| <td>{index}</td> |

|  |
| --- |
| <td>{item.name}</td> |
| <td>{item.price} $</td> |

|  |
| --- |
| <td><label className="badge badge-warning" onClick={()=>**this**.Edit(item,index)}>modify</label></td> |
| <td><label className="badge badge-danger" onClick={()=>**this**.deleteItem(item.name)}>remove</label></td> |

|  |
| --- |
| </tr> |
| )) |

|  |
| --- |
| } |
| </tbody> |

|  |
| --- |
| </table> |
| </div> |

|  |
| --- |
| </div> |
| </div> |

|  |
| --- |
| ); |
| } |

|  |
| --- |
| } |
|  |

|  |
| --- |
| export **default** App; |

Nếu bạn có xem qua các bài viết trước thì bài này cũng giống giống vậy, chỉ khác hàm (**addItem,updateItem,deleteItem**)  
+ **addItem()** : khi gọi hàm này thì, một item sẽ được thêm vào mảng items

[view source](https://hoanguyenit.com/create-crud-in-react.html#viewSource)[print](https://hoanguyenit.com/create-crud-in-react.html#printSource)[?](https://hoanguyenit.com/create-crud-in-react.html#about)

|  |
| --- |
| addItem =()=>{ |
| **if**(!**this**.state.items.find(item=>item.name===**this**.state.name)){ |

|  |
| --- |
| **this**.setState({ |
| items:[ |

|  |
| --- |
| ...**this**.state.items, |
| { name:**this**.state.name,price:**this**.state.price} |

|  |
| --- |
| ], |
| name:"", |

|  |
| --- |
| price:"" |
| }) |

|  |
| --- |
| } |
| } |

Đầu tiên ta sẽ xét, xem biến name có trùng trong mảng items không, nếu không ta sẽ thêm vào mảng items  
Với lại bạn sẽ thấy bên trên có dấu**(...)**, được gọi là **Spread operator (…) trong ES6**, bạn có thể xem thêm tại đây:  
*https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/Spread\_syntax*

+ **updateItem():** ta dùng cập nhật một item, đầu tiên ta cần tìm vị trí index của item được chọn, sau đó cập nhật tại giá trị item đó

[view source](https://hoanguyenit.com/create-crud-in-react.html#viewSource)[print](https://hoanguyenit.com/create-crud-in-react.html#printSource)[?](https://hoanguyenit.com/create-crud-in-react.html#about)

|  |
| --- |
| updateItem = ()=>{ |
| let data = **this**.state.items; |

|  |
| --- |
| data.map((item,index)=>{ |
| **if**(**this**.state.index===index){ |

|  |
| --- |
| item.name = **this**.state.name; |
| item.price = parseInt(**this**.state.price); |

|  |
| --- |
| } |
| }) |

|  |
| --- |
| //set update items |
| **this**.setState({ |

|  |
| --- |
| items:data, |
| name:"", |

|  |
| --- |
| price:"", |
| action:'ADD\_ITEM' |

|  |
| --- |
| }) |
|  |

|  |
| --- |
| } |

+ **deleteItem():** ta dùng xóa một item ra khỏi mảng items, ta dùng filter để thực hiện điều này!

[view source](https://hoanguyenit.com/create-crud-in-react.html#viewSource)[print](https://hoanguyenit.com/create-crud-in-react.html#printSource)[?](https://hoanguyenit.com/create-crud-in-react.html#about)

|  |
| --- |
| deleteItem=(name)=>**this**.setState({ |
| items:**this**.state.items.filter(item=>item.name!=name) |

|  |
| --- |
| }) |



