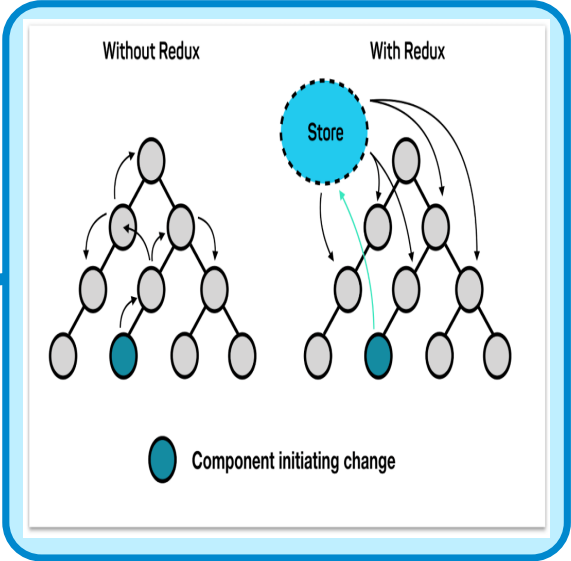


Bai15Redux

Why



Toán tử ba chấm

Gán biến bằng nhau nhưng không trở đến cùng 1 vùng nhớ

```
render() {  
  var p1 = {  
    text: "man hinh",  
    gia: 100  
  }  
  console.log(p1);  
  var p2 = [...p1];  
  p2.gia = 100;  
  console.log(p2);  
  console.log(p1);  
  
  var s1 = [1,2,3,4,5];  
  var s2 = [...s1];  
  console.log(s1);  
  s2[0] = 100;  
  console.log(s2);  
  console.log(s1);  
}
```

Khi gán p1 = p2 thì cả 2 đều trở đến 1 vùng nhớ nên để trở về 2 vùng nhớ khác nhau thì thêm dấu 3 chấm

là mảng thì p1 = [...p2]

là đối tượng thì p1 = {...p2}

Cấu trúc của Redux

Action

là những chỉ thị gồm thuộc tính bắt buộc là type

type: tên chỉ thị

item: phân tử thao tác

action: không trực tiếp hiện mà chỉ đưa ra các chỉ thị

reducer :là người thực hiện chi tiết

Reducer

Reducer là 1 hành động

hành động này trả về 1 state

có 2 tham số

old state

chỉ thị action

action dùng để thay đổi state cũ và tính ra state mới

state

khi khai báo reducer nhận vào tham số là action thì

1. action đó có thể đặt tên được thông qua action.type

2. tác dụng của action là chỉ thị để tính ra state tiếp theo

Store

LÀ 1 đối tượng quản lý reducer và state, vì state là private, không thể thay đổi trực tiếp được mà phải thông qua đối tượng quản lý của nó là store

Cài đặt redux

npm install redux --save

```
render() {  
  
  var redux = require('redux');  
  var oldState =  
  {  
    num: ["man hinh", "chuột", "Bàn phím"],  
    editStatus:true  
  }  
  var reducer1 = (state=oldState, action) =>  
  {  
    switch (action.type) {  
      case "CHANGE_EDIT_STATE":  
        return {...state,editStatus:!state.editStatus};  
    }  
  }  
  // reducer quản lý state, store quản lý reducer  
  // thông qua store mới truy cập được state  
  // action chỉ là cái tên thôi  
  var store1 = redux.createStore(reducer1);  
  store1.dispatch({type:"CHANGE_EDIT_STATE"});  
  console.log(store1.getState());  
}
```