--1) Lenh 1 : Cai dat 1 lan : npm install -g create-react-app

--2) Lenh 2 : cai dat khung reactjs : create-react-app vd01

--3) lenh 3 : chay app -> npm start

--4) lenh 4 : react router : npm install react-router-dom

--4) Luu y ; khi gheo giao dien front-end : %PUBLIC\_URL%/ -> vao thu muc public

--5) Tai lieu : https://reacttraining.com/react-router/web/guides/quick-start

--6) gui du lieu trong from : thi k duoc load lai trang, nhan du lieu ngay tren trang do -> url redirect - target

--7 ) uuid : de id tu tang -> npm install uuid

--react thuan : thi phai su dung truyen qua truyen lai

Luu y : state luu hanh noi bo trong component day thoi - xem tromg image

project user : vd05\_user\_manage

VD : button them moi va dong lai neu o trong cung component AddUser thi chi can lam state la ok

truong hop neu nam khac component Vd 2 nut button nam o component searchForm thi phai lam : - su dung props

2 component nay phai lien ket thong qua thang cha

- SearchForm ket noi voi app.js - component cha

- AddUser ket noi voi app.js - component cha

luu y : bo muon truyen cho con thong qua props

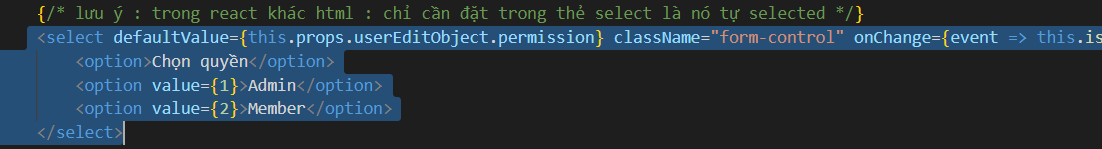
con muon truyen cho bo : tham so cai ham trong props

- su dung ham map phai co return

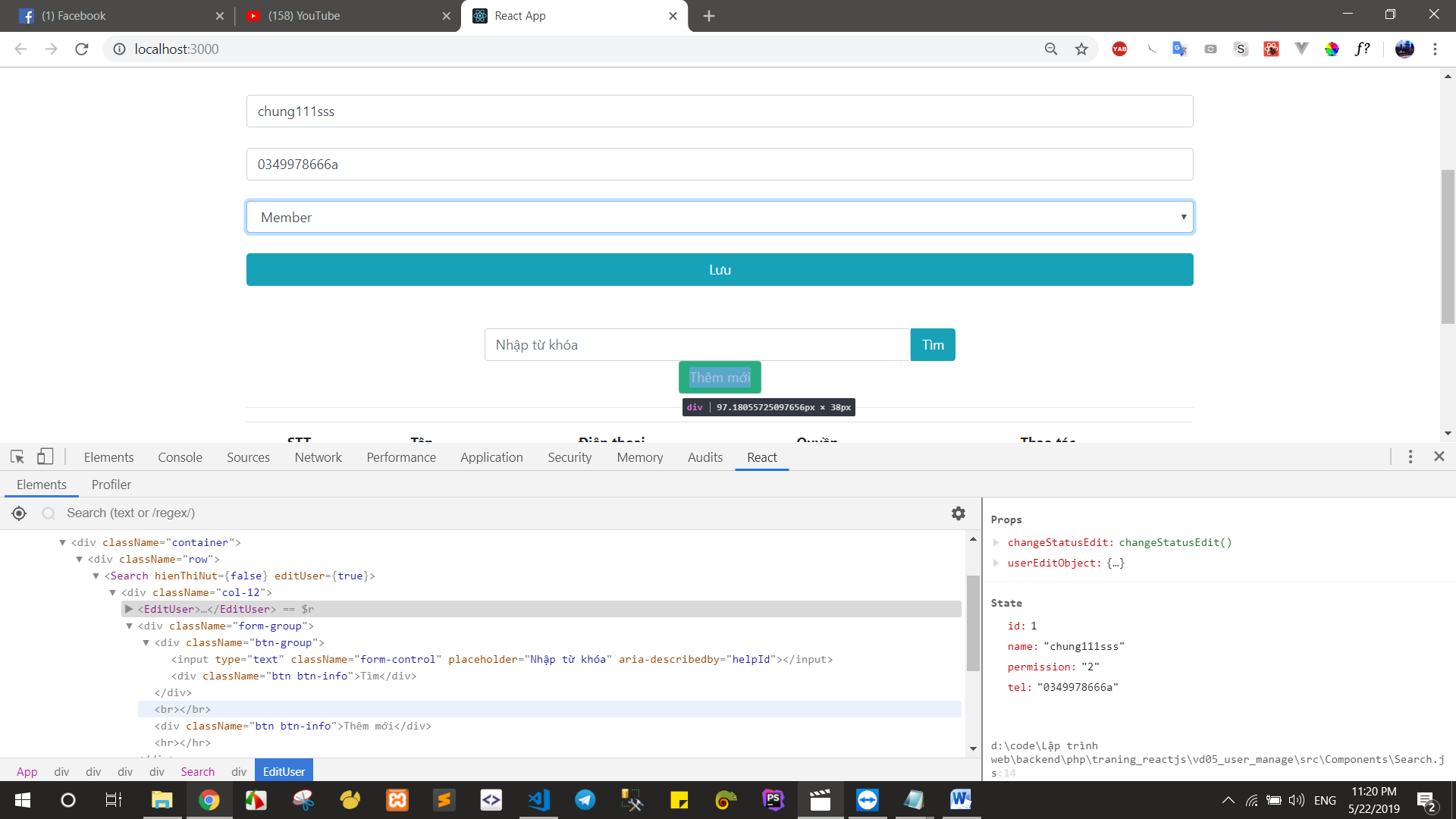
- foreach : khong co return

--uuid : tu dong tang id : giong auto increment : https://www.npmjs.com/package/uuid

{/\* luu ý : trong react khác html : thẻ select option : selected chỉ cần đặt trong select chứ k cần trong option

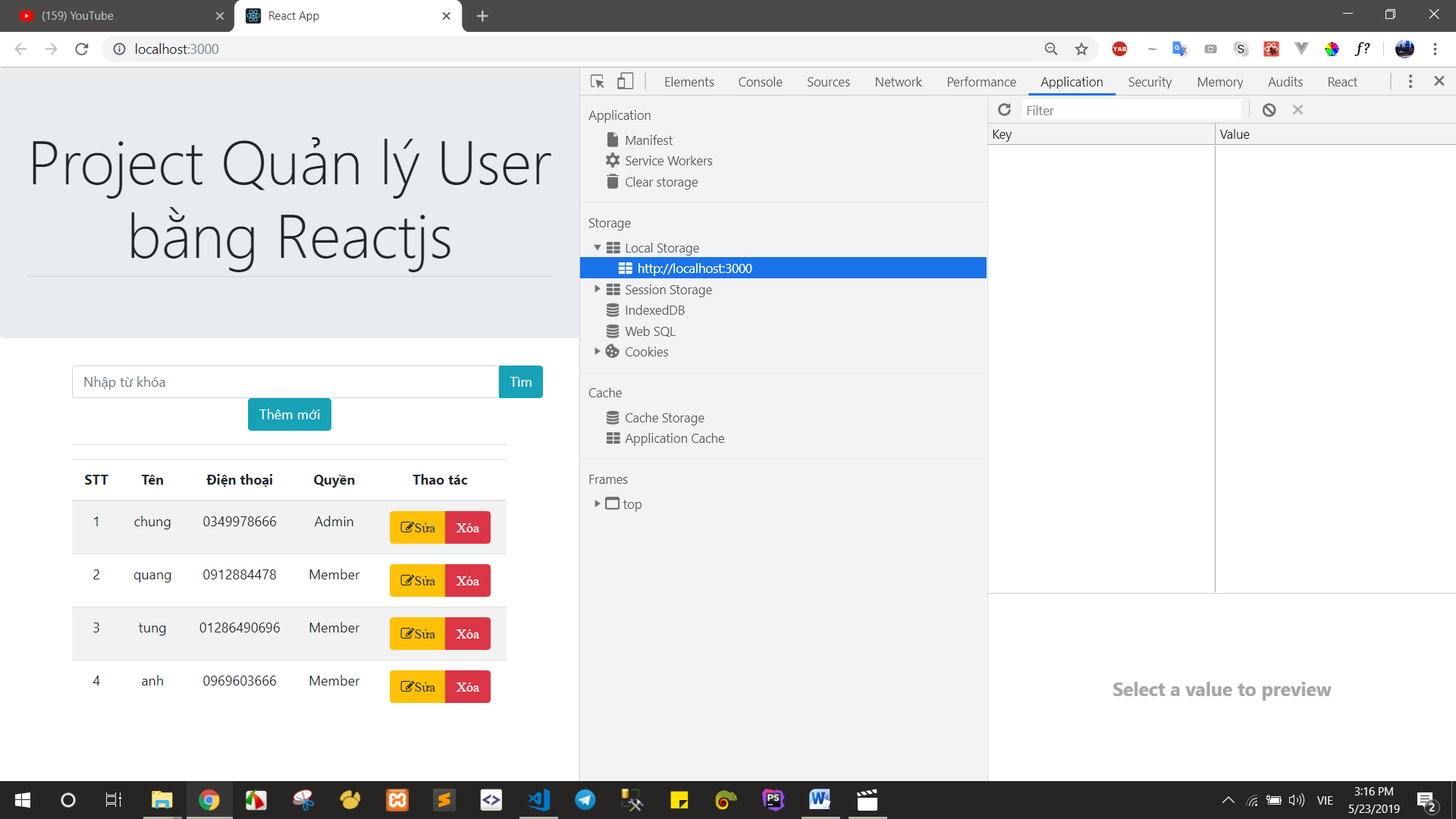


* Cách xem dữ liệu trực tiếp – k cần phải console.log : react developer tool

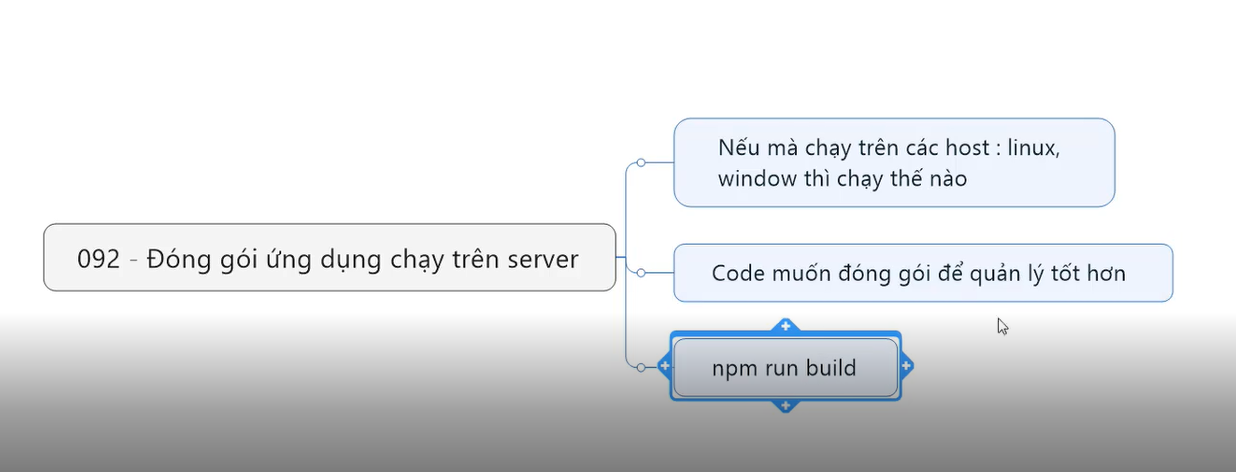


* Dung local storage : Viet trong trong render app.js roi chay xem thu
* localStorage.setItem('key1','haha');
* console.log(localStorage.getItem('key1'));
* localStorage.removeItem('key1');



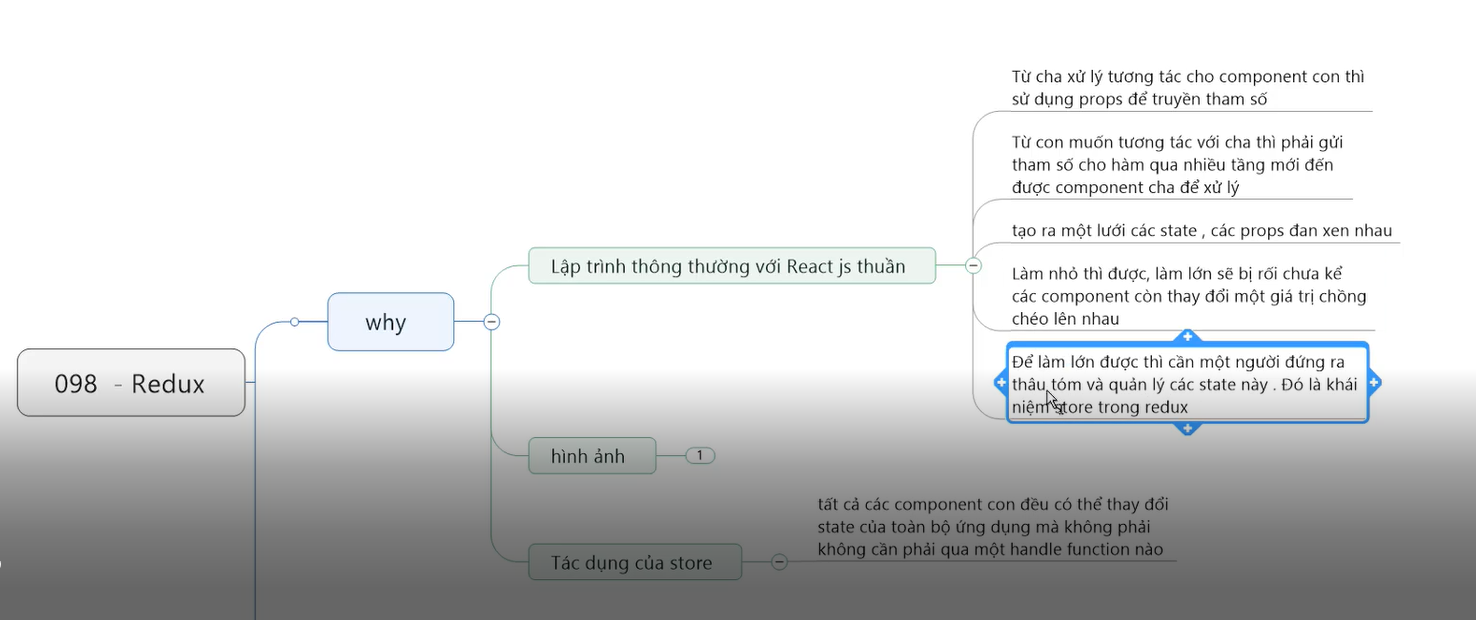


* Đóng gói ứng dụng để phát triển lên server: sẽ có folder tên là build
* Chạy trên local xampp: user.react trong htdocs
* Có thể chạy = dòng lệnh : chuột phải giữ shift rồi mở folder trong htdocs
  + npm install -g serve
  + serve -s build – lưu ý chạy trong build thì thôi bỏ build – chạy luôn
* 18) dowve xong **npm install** trỏ đến đúng folder mới chạy -> vì tải về có kb chưa đủ thư viện

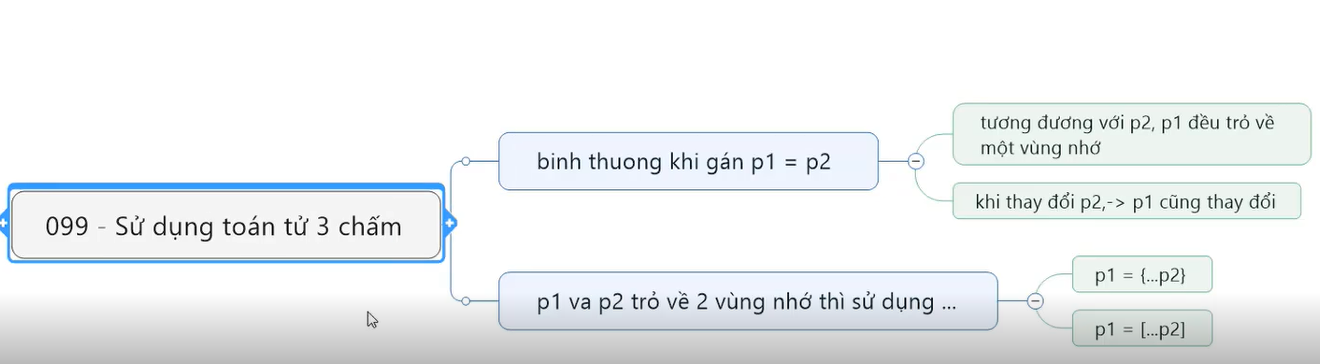


------------------------------redux-----------------------------------

1. KN



1. Toán tử 3 chấm
   * Create-react-app vd06\_redux



* + VD
  + Sau khi gán p2 = p1
* var p1 = {
* ten : "man hinh",
* gia : 90
* }
* console.log(p1);
* var p2 = {...p1};
* p2.gia = 100;
* console.log(p1);
* console.log(p2);
* Viết tắt toán tử 3 chấm và thêm phần tử

var b1 = {

num : [12,13,14],

status : true

}

var b2 = {...b1,status:false};

console.log(b2);

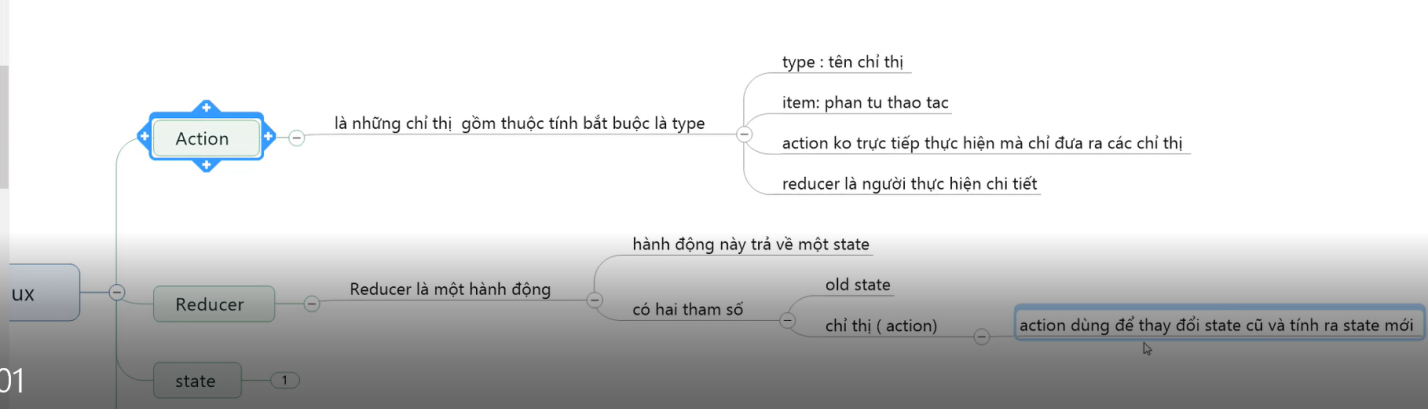
//thêm phần tử vào mảng num

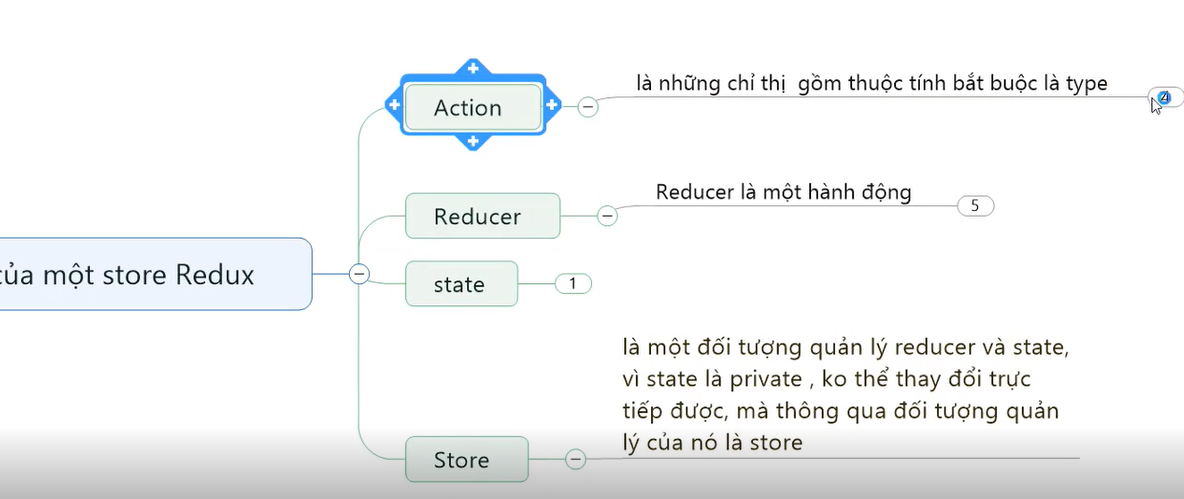
var b3 = {...b1,num:[...b1.num,100]}; //viết tắt

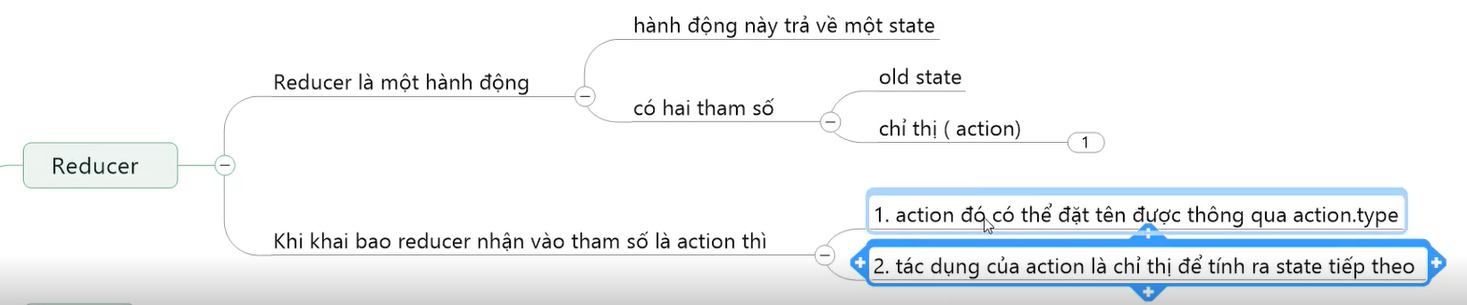
//b3.num[3] = 100; viết thường

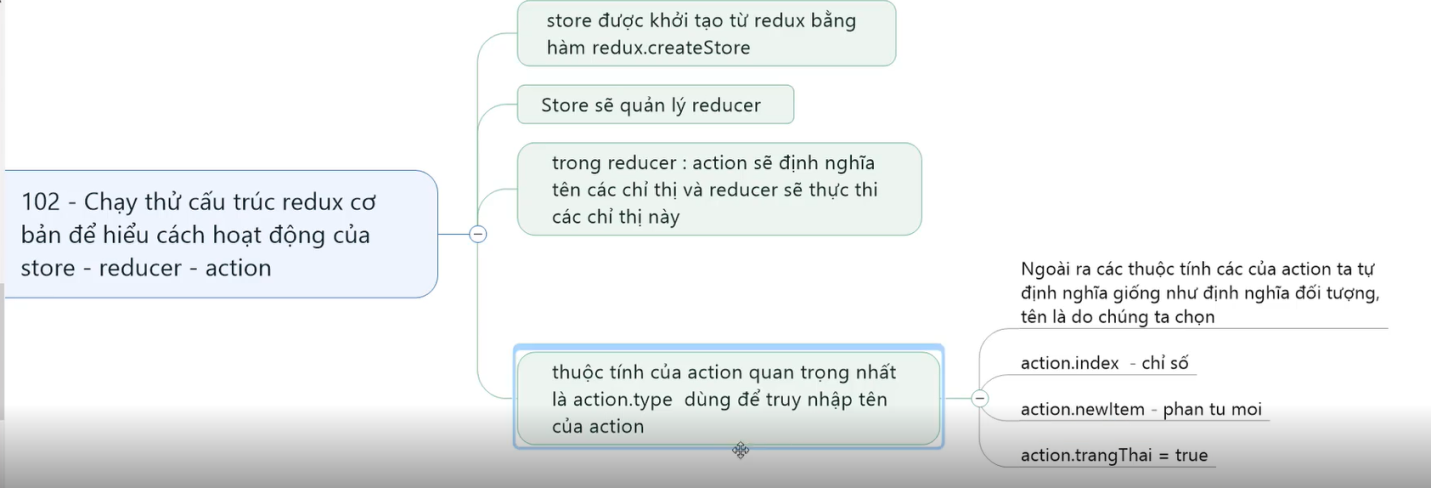
console.log(b3);

1. KN Redux
   * Npm install redux –save
   * React Redux ES6 Snippets – cài plugin trong visual code



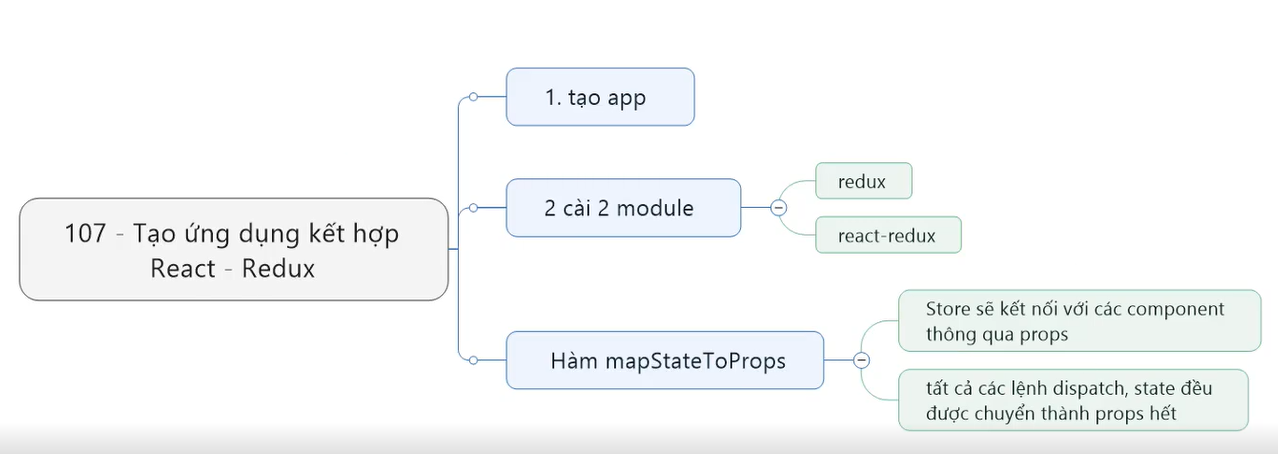




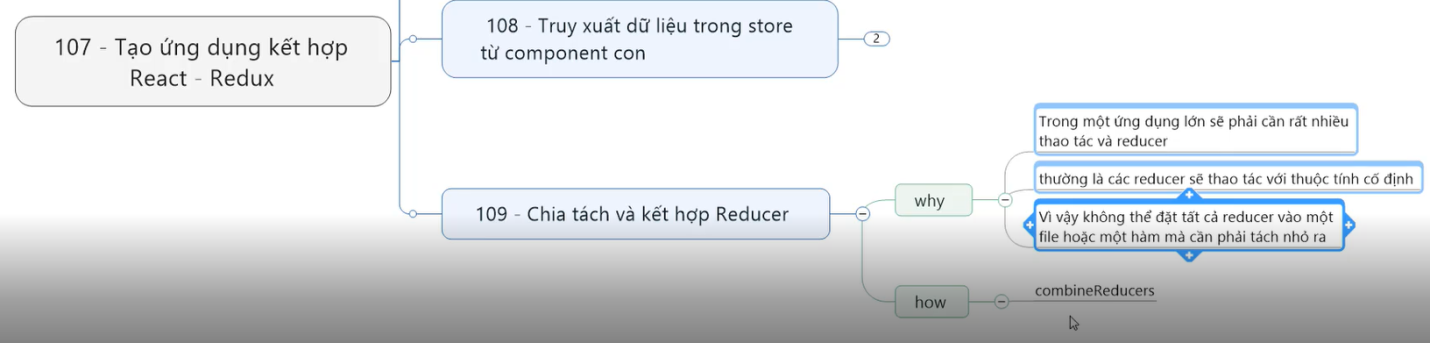


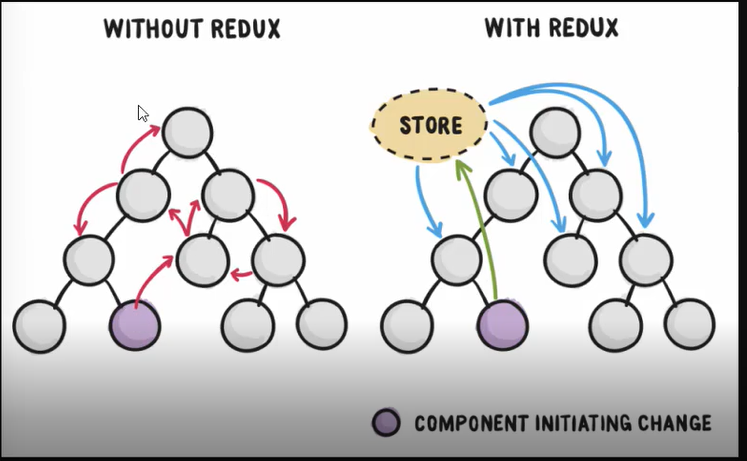
* Store
  + createStore : quản lý reducer và state
  + subscribe : theo dõi sự biến đổi của state mỗi khi chạy dispatch
  + dispatch : tiến hành làm công việc mà reducer giao

1. REACT VÀ REDUX
   * Create-react-app vd07\_react\_redux
   * Npm install redux – trỏ đến file vd07 rồi mới cài
   * Npm install react-redux – trỏ đến file vd07 rồi mới cài



* Tách nhỏ reducer

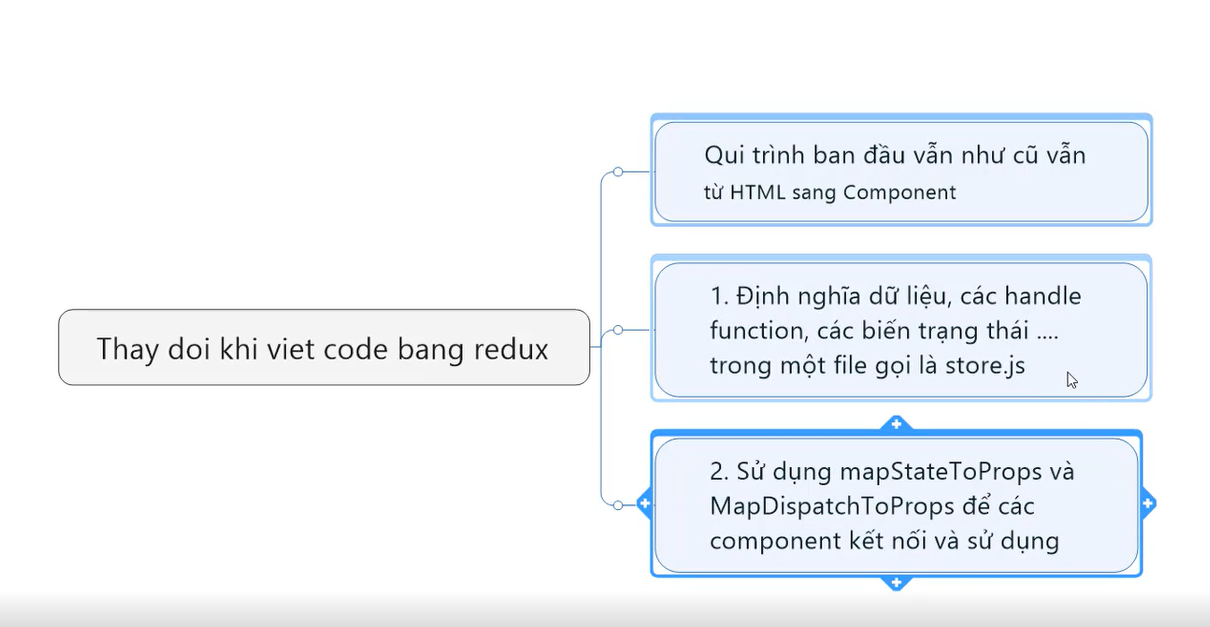




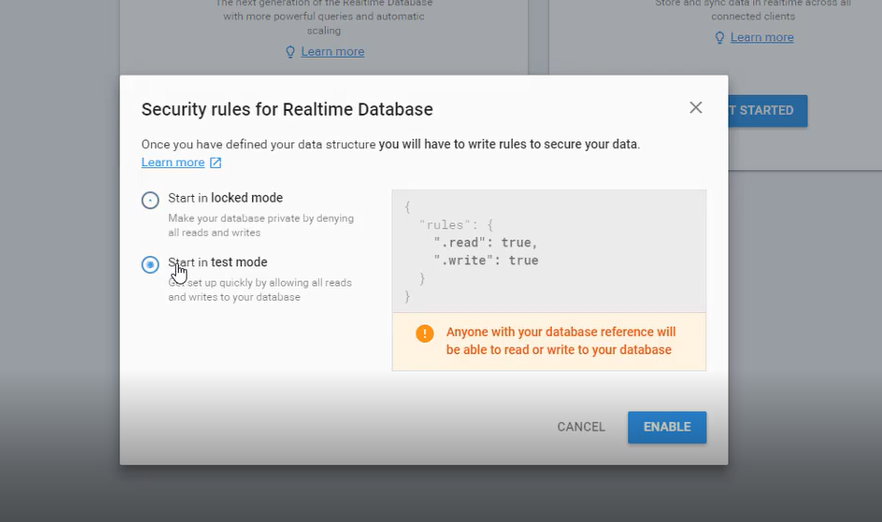
* Từ store truyền sang component thì chuyển hết state thành props mới truyền được (mapStateToProps)
* Từ component lên hàm trong store(2 cách): kết nối xong truy nhập dispatch

userEditStatus : là hàm để khi click thì thay đổi trạng thái

* + Cách 1 : this.userEditStatus
* userEditStatus = () => {
* /\* var dispatch = this.props.dispatch; \*/
* //ngắn gọn hơn
* var {dispatch} = this.props; //được truyền dưới dạng props
* dispatch({
* type: "CHANGE\_EDIT\_STATUS"
* })
* }
* Cách 2:
  + Sử dụng hàm mapDispatchToProps : this.props.userEditStatus
* const mapDispatchToProps = (dispatch, ownProps) => {
* return {
* userEditStatus: () => {
* dispatch({
* type: "CHANGE\_EDIT\_STATUS"
* })
* }
* }
* }
* Tổng kết phần react và redux

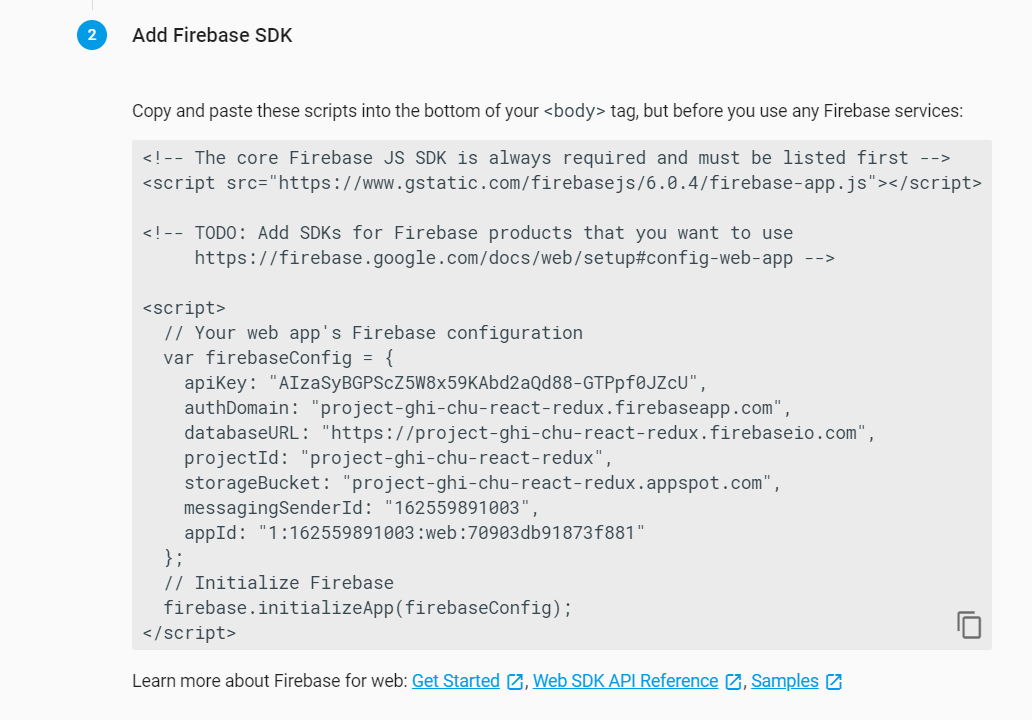


1. Project quản lý ghi chú
   * Chú ý : khi tạo giao diện bootstrap thì chia:col và col-4 để khi k có col-4 thì cái col kia nó tự co giãn đẹp hơn

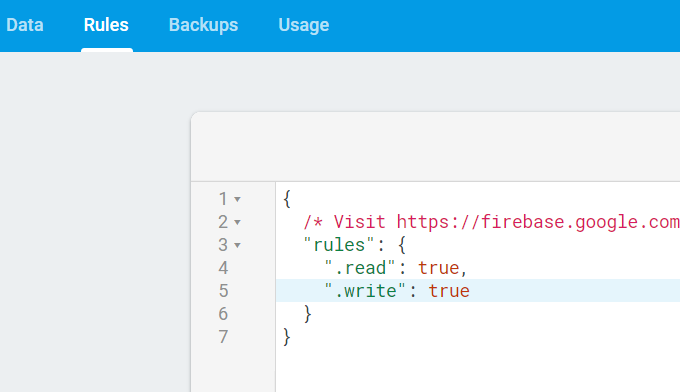
* <div class="container">
* <div class="row">
* <div class="col">
* Lorem ipsum dolor sit amet consectetur adipisicing elit. Sunt deserunt voluptatem non culpa nam nisi cupiditate earum eaque! Aliquid impedit rem vitae sed corrupti quae natus fugiat dolore, maxime in.
* </div>
* <div class="col-4">
* Lorem ipsum dolor sit, amet consectetur adipisicing elit. Vero aperiam ea, voluptate sequi possimus reiciendis illum modi eum quisquam culpa doloribus a dicta laborum tenetur magni, quo ipsam doloremque in?
* </div>
* </div>
* </div>
* Sử dụng firebase
  + 
* Tạo dữ liệu với firebase và có thể xuất filew JSON ra để xem – lưu ý chọn realtime database



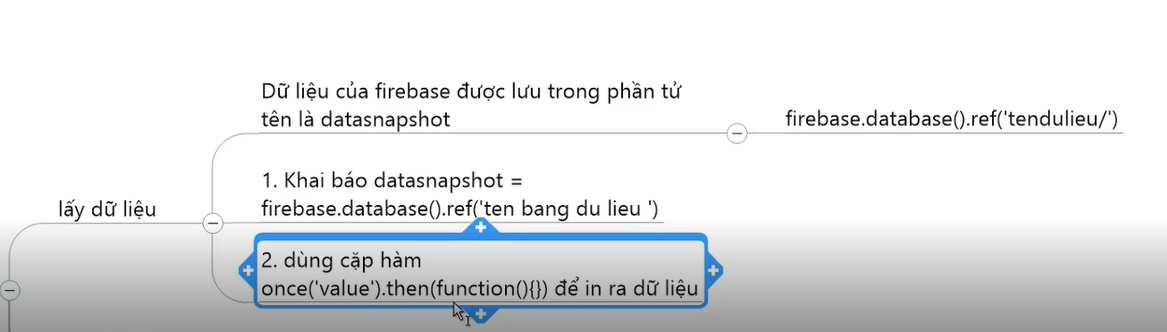
* Kết nối firebase với react
  + Trỏ đến folder -> npm install firebase –save
  + Vào phần cài đặt trong firebase -> chọn web rồi copy đoạn js paste vào ứng dụng
  + Import rồi console.log ra xem



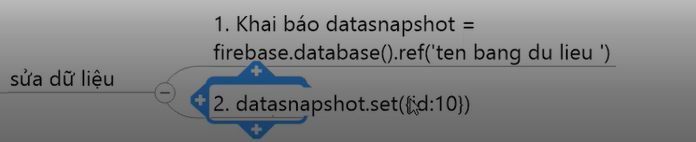
* Nếu lỗi k lấy được dữ liệu false -> true



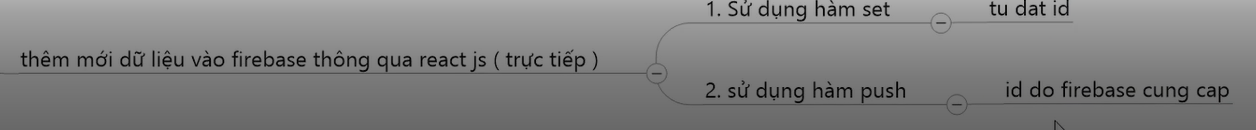
* Lấy dữ liệu từ firebase trong code



* Sửa dữ liệu firebase trong code : nếu k có thì nó thêm luôn,còn có rồi thì nó sửa(chức năng thêm luôn, nhưng thêm có 1 cách nữa) – k nên dùng để thêm tại vì phải ghi cả id



* Thêm dữ liệu : push – k nên dùng set



* Xóa dữ liệu



---------------------------------xong 121

redux : framework : store quan ly nen de dang hon

--------------------------------------------------------------

PHIM TAT TRONG VISUAL CODE

--1) Tao component class : rcjs

--2) Tao component : rcc

--3) sst : set state

--4) cwm : componentwillmount

--5) cdm : componentdidmount

--6) scu : shouldComponentUpdate

--7) componentWillUpdate

--8) componentDidUpdate

--9) componentWillReceiveProps

--10\_ shif alt a : comment nhiều dòng