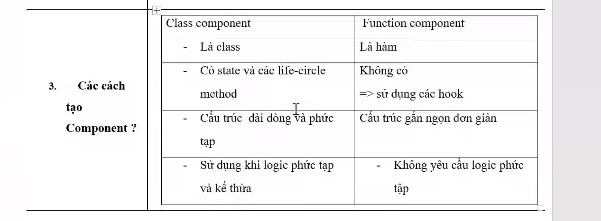


Compoent có thể tái sử dung(mục đích)

* Dễ dàng tái sử dụng



* Chủ yếu sử dụng function compoent

App.js là 1 compoent có sẵn(func)

Tạo compoeent

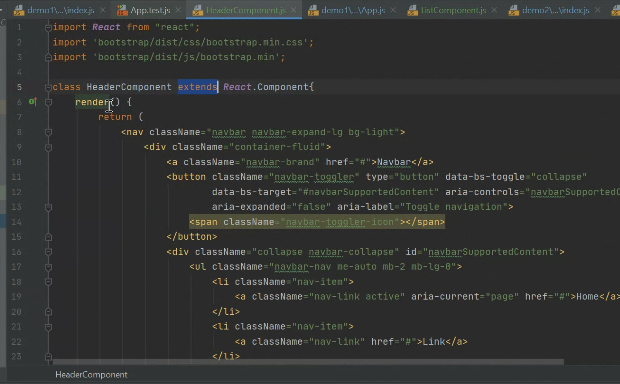
* Tạo thư mục compoent
* Tạo file js trong thư mục



Vẫn cần import boostrap

**class**

Cần extend và render(){ return



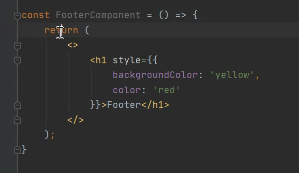
Phải có



Để sang file khác có thể dùng

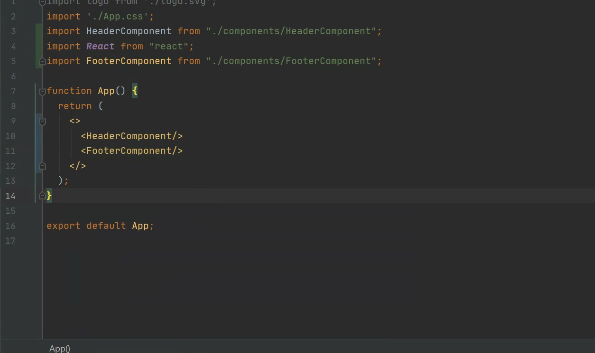
**Function(ví dụ footer)**

**Arrow** funcetion chỉ cần return

****

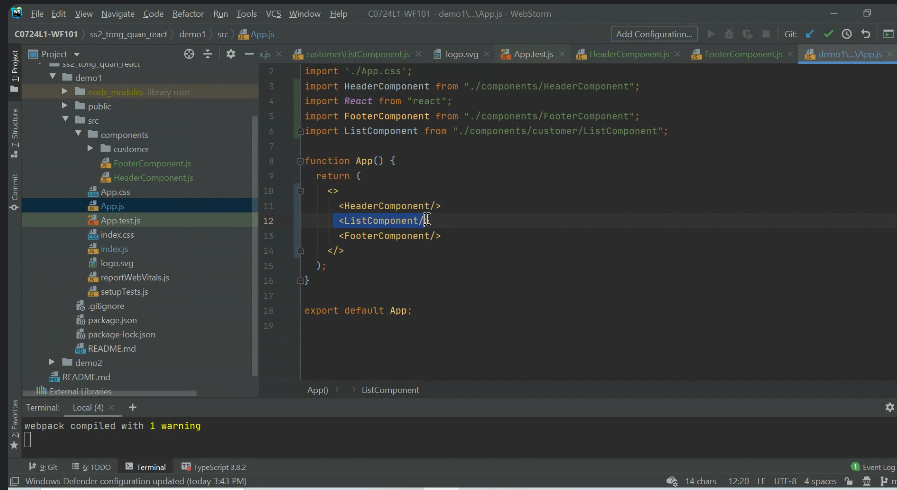
****

**Sử dụng:**

****

và trong file js chỉ việc render ( <App/>)

sử dụng list



Cần import nếu k ở ngay file hiện hành

**Props là một đối tượng**

Muốn sử dụng list import vào compoent

 (props



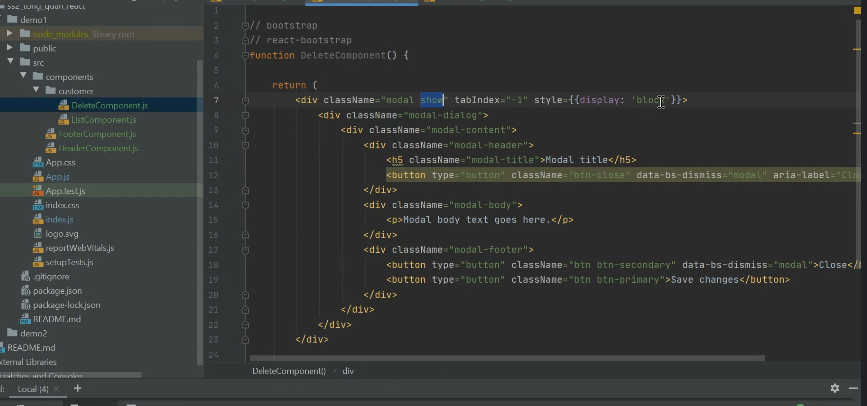
Props là một đối tượng

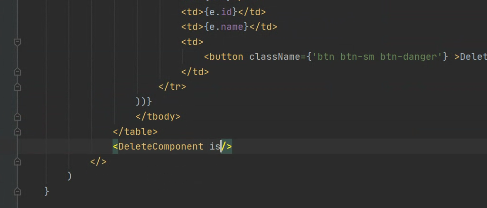
List,size là một thuộc tính

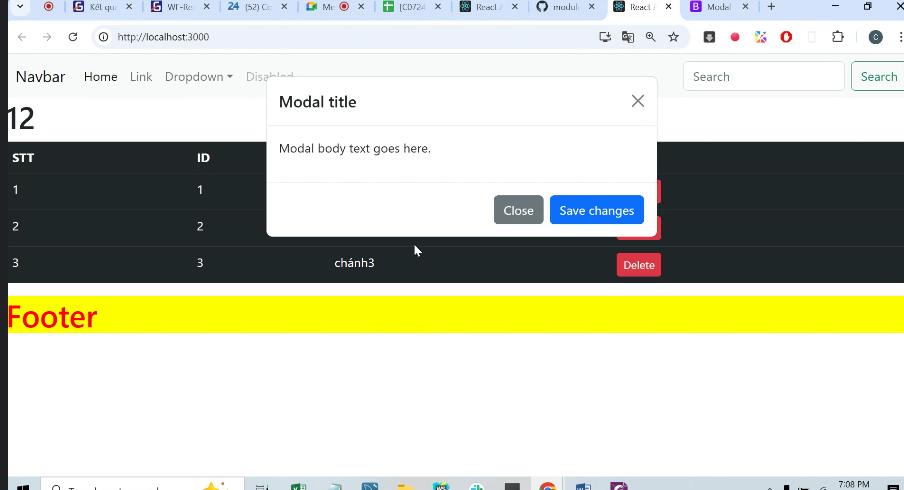
* Truy xuất props



Tạo hộp thoại xóa:

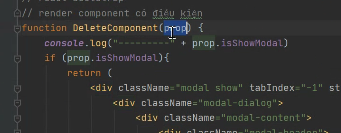


Truyền vào list: 



Truyền vào thuộc tính

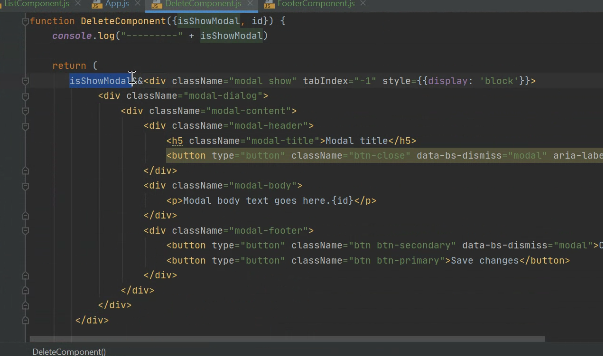




(điều kiện để hiển thị modal)

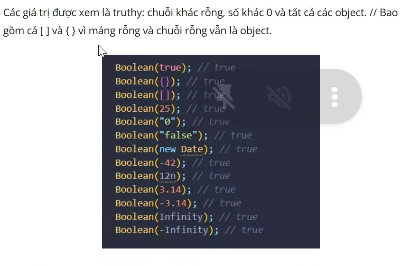
Cách thường dùng(k dùng if) => toán tử &&

(falsy và truthy)



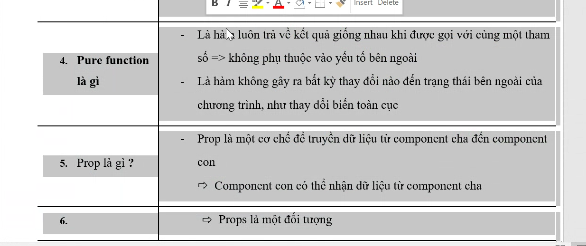
* **isShowModa&&…**

nếu đúng



=> thì hiển thị

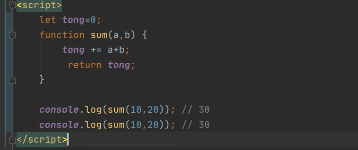
**còn nếu trả về false hoặc giá trị k xđ => k trả về gì**

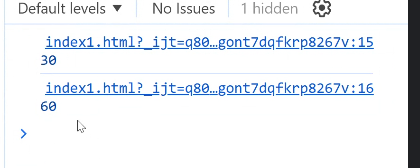


Pure function => cùng một biến trả về cùng 1 giá trị(vd return a+b)

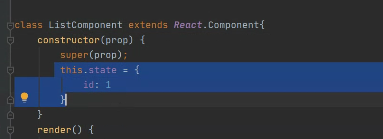
Còn k phải => 2 hoặc nhiều lần gọi sẽ trả về giá trị khác nhau:

Ví dụ hàm k phải pure function





State,life -circle chỉ class có:



Function k có nên phải sử dụng hook

