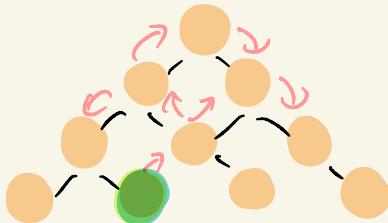
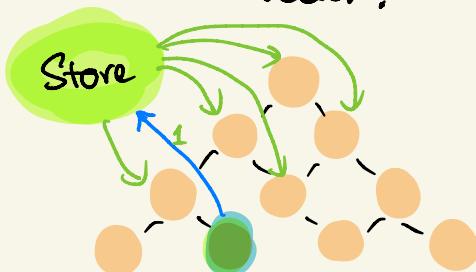


Mục đích Redux: lưu dữ liệu trong store để kiểm soát quá trình code tốt hơn.

Ko có Redux:



redux:



Điểm chung dữ liệu tùy biến

Mục đích: anh Sơn muốn xuất ra UI 1 cái list data xe
và có thể thêm, sửa, xóa Siêu xe của anh

Bước 1: tạo file Reducer.js.

Reducer
.js

Data (xe)

hàm Reduce để tùy chỉnh data dựa vào hành động.

hành động?
(thêm, sửa, xóa)

có
trả về
data mới

Không
trả về data
ban đầu

Nói chung Reducer.js sẽ trả về data ở đây là xe

Bước 2: Core.js

Core.js

hàm html: hàm này để mai một input HTML vào cho tiến logic hàm này nếu bạn không hiểu thì học lùi bài 189 và 192 Khác cơ bản

Create Store

CreateStore Sẽ nhận tham số là Reducer

State = Reducer() : function này là hàn̄m reducer đã được chạy.
⇒ nó là data xe

Roots là 1 cái Object mà Key của nó có thể là DOM

Roots: { root: component, ... } function nếu chạy sẽ xuất ra
hàn̄m Render ra UI : thành phần innerHTML của root là component()
HTML

hàn̄m CreateStore sẽ xuất ra 1 obj gồm 3 hàn̄m :

attach(component, root) : tiến hành render() cho 2 tham số nòi nhận

Connect(selector = State) : hàn̄m này mục đích của nó là chạy hàn̄m component khi data bị thay đổi qua hàn̄m reducer.

→ Cái Connect này thật chất là component()
Khi data thay đổi

dispatch(action, arg) : update State Khi có hành động mới từ user
(action)

Bước 3:

Store.JS

thông nay chẳng làm gì ngoài việc
chạy hàn̄m CreateStore với đối số là Reducer
chạy xong thì nó xuất ra 3 hàn̄m attach
Connect để xài
dispatch

Bước 4:

App.JS

cái này để viết hàn̄m Component (đặt tên nó là app)
hàn̄m app này thât ra chính là cái mã HTML
mà ta mong muốn sẽ bỏ vào root

trong mã HTML Sẽ có' nút để' thực hiện hành động và hành động
đó mang lối sử thay đổi trong **reducer**
⇒ ta sẽ sử dụng hàm **dispatch** để' báo cho hàm **reducer** biết là
nó Sẽ thay đổi data thế nào

Sau đó ta sẽ dùng **connect** để' chạy hàm app với data dưới
cập nhập từ reducer

Các bạn chú ý là hàm **connect** thực ra là chạy lây hàm app
với data mới thôi.

Bước 5 :

Script.
JS

đến đây thì dễ rõ, tiến hành render ra
UI với **attach**(**Component**, **root**), truyền đối
số vào cho **attach**:



→ Xong