**Up Github:**

* Tạo thư mục trên github ở phần Your repositories -> New repositories
* Right click -> Git Bash here -> git clone *đường link thư mục git*
* Right click -> Git Bash here vào trong thư mục mới clone về -> git status (Kiểm tra file đó có được theo dõi chưa)
* git add \* (để úp lên hết) hoặc git add tên file (để úp lên 1 file) : Để theo dõi file
* git commit –m ’’Mô tả file đưa lên’’ : đưa lên kho chứa
* git push : đưa lên github

**Để quay lại 1 thời điểm nào đó mình muốn:**

* git log : Chọn thời điểm mình muốn quay lại, copy id ở phần comit
* git revert idcomit
* Shift: q : để đi ra lại

Khi sửa đổi trên git muốn đồng bộ xuống thư mục của mình thì sử dụng *git pull*

**Host free:**

https://vn.000webhost.com/

<https://freevnn.com/>

https://byet.host/

**Tiện ích chrome:**

+ react-detector: Cho phép trang nào đang dùng react

+ react developer tools: F12 -> Chọn Reactjs kiểm tra code

+ show me the react:

**Cài đặt:**

+ Node js

+ Gitbash (git scm)

**Kiểm tra:**

Vào cmd (Window + R) -> node - - version

**Thực hiện Project:**

Tạo thư mục project - > right click - > git pash here - > npm init -> đặt tên, thông tin,…

Cài thêm gói nodejs: npm install express ejs

Hoặc:

npm install -g create-react-app

create-react-app my-app

cd my-app

npm start

**Set port cho project:**

Trong file package.json -> "start": "set port=4200 && react-scripts start",

**JSX:**

- JSX (Javascript Syntax Extension – JS XML): Viết theo kiểu XML. Thay thế cho React.createElement().

- JSX không phải là HTML

- JSX có thể viết như HTML

- Thuộc tính class => *className*

- Search google: HTML to JSX

**Props:**

- Là các thuộc tính của 1 component

- Truyền dữ liệu từ cha -> con, theo dạng *key= “value”* (Value nhận vào có kiểu dữ liệu là chuỗi). Bỏ trong dấu { } để truyền đúng kiểu dữ liệu

- Nhận lại thông qua từ khóa: *this.props.key*

- Nhận nội dung bên trong thẻ: *this.props.children*

**Refs:**

- Lấy giá trị thông qua ref. (VD lấy giá trị ô input, textarea, select, …)

- Cú pháp: thêm thuộc tính ref= “key”

- Lấy giá trị: this.refs.key.value

**State:**

- Là trạng thái của component

- Khai báo những giá trị cần lưu trữ của riêng component đó.

- Tạo state tại constructor. Gọi this.state = { key:value, key1:value1,… }

- Gọi state: this.state.key

- Thay đổi state: this.setState({ key:value, key1:value1,… })

- Khi setState được gọi => hàm render được gọi

|  |  |
| --- | --- |
| Props | State |
| Nhận dữ liệu từ bên ngoài | Dữ liệu nội bộ |
| Không thể thay đổi giá trị | Có thể thay đổi giá trị |
|  | Phạm vi Private trong component |

.preventDefault(): ngăn load lại trang

{new Date().toLocaleTimeString()} : Hiển thị giờ phút giây hiện tại

**LocalStorage && SessionStorage:**

- Giúp lưu trữ dữ liệu tạm thời ở trình duyệt. Tab Application

- Tương tự cookie. Lưu ở trình duyệt

- Lưu ~5MB

- Mất khi xóa lịch sử hoặc sửa dụng lệnh để xóa

- SessionStorage mất khi đóng trình duyệt

- Các phương thức chính:

+ \_setItem(‘key’,value)

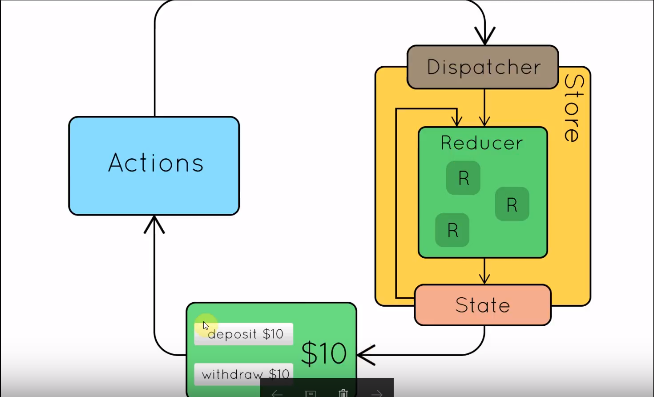
+ \_getItem(‘key’)

+ \_removeItem(‘key’)

**Build Project: npm run build**

**REDUX**

**Install redux:** npm install redux –save



|  |  |
| --- | --- |
|  | |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  | |
|  |  |