*https://reactjs.org/tutorial/tutorial.html*

**Buổi 01:**

* K/n:

+ Javascript: ES6, Class ( biến , hàm), JSX

+ Reactjs: ưu điểm (khi nào cần sd)

What Is React?

React is a declarative, efficient, and flexible JavaScript library for building user interfaces. It lets you compose complex UIs from small and isolated pieces of code called “components”.

* Setup và chạy chương trình đầu tiên:

+ NodeJS (runtime environment.)

+ NPM (node package manager): dùng cài đặt các dependencies (tự cài khi cài nodejs) . CP: npm install/I [package name] –save . Kiểm tra phiên bản: node -v

+ visual studio code + các extentions reactjs, html, css

+ Setup create-reacta-app:

*https://reactjs.org/docs/create-a-new-react-app.html#create-react-app*

npm i -g create-react-app

tạo project: create-react-app [ten app] 🡪 npm start

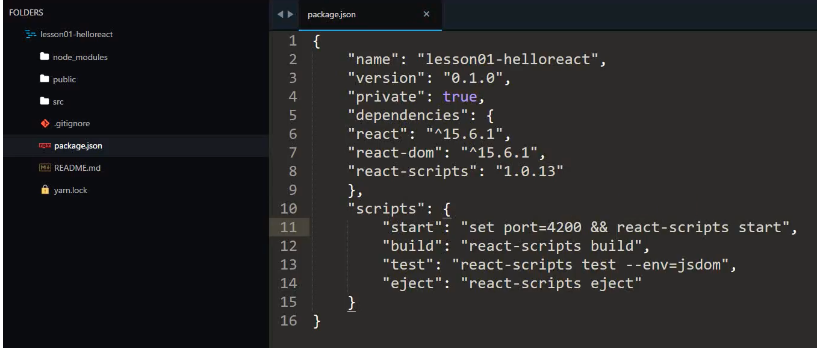
or:

npx create-react-app my-app

cd my-app

npm start

+ đổi port tại package.json



+ Cấu trúc cơ bản. Các sử dụng lồng nhau giữa các component

* Bài tập: Tạo giao diện todo list (phân chia component)

**Buổi 02:**

* Tìm hiểu props và state:

*https://viblo.asia/p/reactjs-bai-2-component-props-state-07LKXYLkZV4*

+ Props ở đây chính là properties của một component, chúng ta có thể thay đổi props của component bằng cách truyền dữ liệu từ bên ngoài vào. Props có thể là 1 object, funtion, string, number....

+ State biểu diễn trạng thái của component, state là private chỉ có thể thay đổi bên trong bản thân của chính component đó. Chúng ta có thể change states bằng cách gọi this.setState()

* Truyền dữ liệu từ component cha -> con:

+ Dùng props : key=”value”

* Truyền dữ liệu từ component con -> cha

+ Dùng hàm.

+ Dùng ref

* Truyền dữ liệu giữa các component ngang cấp: best way: redux
* Handle events: search doc: event
* Bài tập: Hoàn thiện bt todo list với logic

Life circle hook

https://viblo.asia/p/reactjs-life-cycle-hooks-XL6lAAoNlek