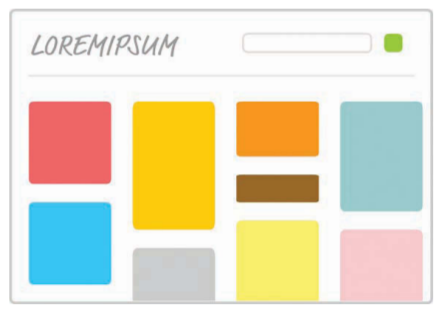
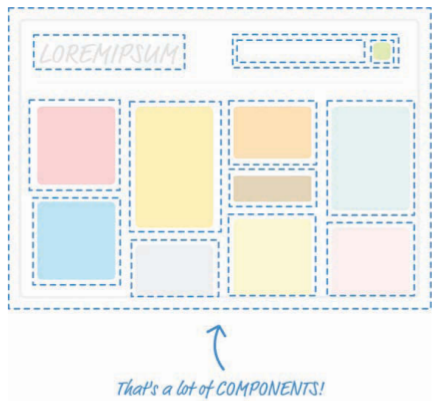
**Chapter 3: Các thành phần trong React**

* **Components là gì?**
* Tất cả những gì các bạn nhìn thấy trên web đều là những Component.
* Như bài giới thiệu về React cũng đã có đề cập, React khuyến khích việc chia nhỏ các thành phần tạo nên một trang web và các thành phần đó được gọi là Components.



* **Ôn lại JavaScript**
* Biến:

Các biến có thể được coi là các thùng chứa có tên. Bạn có thể đặt dữ liệu vào các vùng chứa này và sau đó muốn sử dụng nó thì gọi tên của thùng chứa đó.

* Khai báo biến:

Để khai báo một biến ta sử dụng từ khóa const, let, var.

+ var khai báo biến có thể truy cập ở phạm vi hàm số hoặc bên ngoài hàm số, toàn cục.

+ const được sử dụng để khai báo 1 hằng số, và giá trị của nó không thay đổi trong suốt chương trình.

+ let khai báo biến chỉ có thể truy cập được trong block bao quanh nó được xác định bằng cặp {}.

Đọc thêm về const, let, var tại [đây](https://viblo.asia/p/phan-biet-kieu-bien-var-let-va-const-trong-javascript-ORNZqaOnZ0n).

* Kiểu dữ liệu của biến:

Các kiểu dữ liệu thường gặp:

+ string: ‘/chuỗi/’

+ number: /số/

+ boolean: true/false

+ object: { key: ‘value’ }

+ string array: [‘value1’, ‘value2’]

+ object array: [{ key: ‘value1’ }, { key: ‘value2’ }]

+ null

+ undefined

* Hàm (function):

+ Function (hàm, chức năng), gọi chung là subprogram (chương trình con) có thể được gọi ở bên ngoài hoặc bên trong chính nó.

+ Các bạn có thể coi hàm như là một nhà máy. Các bạn bỏ nguyên liệu vào (tham số) sẽ cho ra thành phẩm nào đó (kết quả return).

Ví dụ:

function tiemTraSua(tra, sua) {

const ly = tra + sua;

return ly;

}

Như ví dụ trên ta đã có một function tiệm trà có nguyên liệu đầu vào là [trà] và [sữa], công thức là [trà + sữa] và thành phẩm là [ly trà sữa].

Vậy để tạo một ly trà sữa truyền tham số giá trị của tra và sua vào function tiemTraSua(tra, sua) như sau:

tiemTraSua(400, 600); (400g trà và 600g sữa)

-> Chúng ta có thành phẩm là ly trà sữa 1000g.

+ Function không nhất thiết phải có tham số và kết quả return.

* **Function Component**
* Function component giống như một function bình thường nhưng kết quả return của nó là JSX.
* Để sử dụng một component thì bạn phải export component đó và import nó vào một component khác (component mẹ). Khi đó các bạn gọi nó như một thẻ HTML thông thường.

Ví dụ:

// Item.jsx

import React from 'react';

function Item(){

return (

<div>Item</div>

);

}

export default Item;

// List.jsx

import React from 'react';

import Item from '../Item';

function List(props) {

return (

<div>

<Item />

</div>

);

}

export default List;

* Props và state:

+ State như là một biến chỉ dùng trong function component và nó có thể thay đổi theo trạng thái của function component (lúc khởi tạo/ thay đổi nội dung/ rời đi/ ...). Chúng ta sẽ nói rõ hơn về state ở bài sau.

+ Props là tham số truyền vào mặt định của function components. Nó có kiểu dữ liệu là object và key và value do các bạn tự định nghĩa.

+ Props chỉ được truyền từ mẹ sang con và không thể truyền ngược lại.

Ví dụ:

// Item.jsx

import React from 'react';

function Item(props){

const name = props.name;

return (

<div>{name}</div>

);

}

export default Item;

// List.jsx

import React from 'react';

import Item from '../Item';

function List(props) {

return (

<div>

<Item name=”Lê Bá Thanh Tuấn”/>

</div>

);

}

export default List;