**MENU**

[**1.** **SPA và MPA** 3](#_Toc138535508)

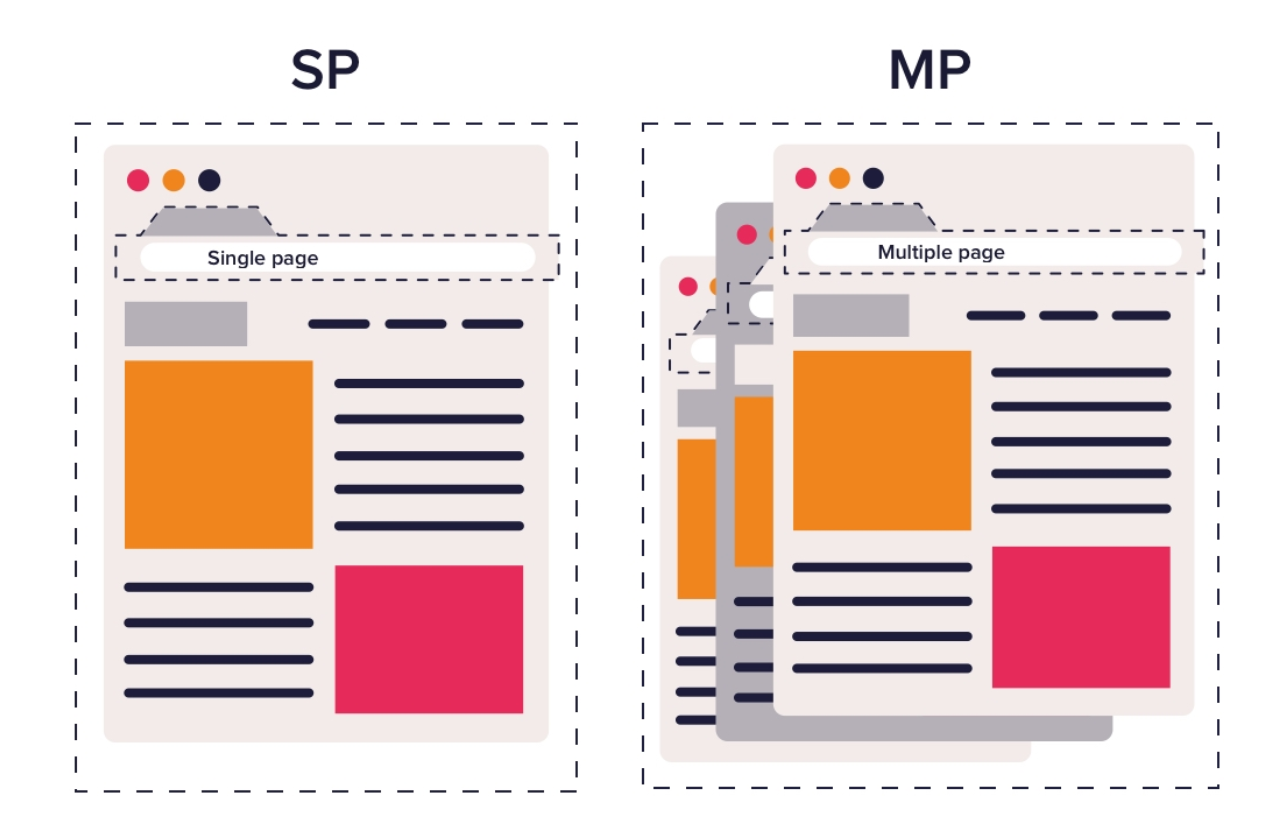
[**1.1** **Giới thiệu** 3](#_Toc138535509)

[**1.2** **So sánh** 3](#_Toc138535510)

[**1.3** **Lựa chọn SPA hay MPA** 4](#_Toc138535511)

1. **SPA và MPA**

**-**Là 2 kiến trúc phát triển web hiện nay



* 1. **Giới thiệu**

**SPA**

-Single page application được cho là cách tiếp cận hiện đại hơn

-Không yêu cầu tải lại trang trong quá trình sử dụng

**MPA**

-Multi page application là một các tiếp cận cổ điển (vẫn phổ biến đến nay)

-Tải lại trang trong quá trình sử dụng (click vào đường link, chuyển trang, …)

* 1. **So sánh**

**+Tốc độ**

-**SPA** nhanh hơn khi sử dụng. Phần lớn tài nguyên được tải trong lần đầu. Trang chỉ tải thêm dữ liệu mới khi cần

-**MPA** chậm hơn khi sử dụng. Luôn tải lại toàn bộ trang khi truy cập và chuyển hướng

**+Bóc tách Code**

-**SPA** có phần Front-end riêng biệt

-**MPA** Front-end và Back-end phụ thuộc nhau nhiều hơn, được đặt trong cùng 1 dự án

**+SEO**

**-**SEO (Search Engine Optimization) là quá trình tối ưu hóa công cụ tìm kiếm, nhằm cải thiện vị trí và hiển thị của trang web trong kết quả tìm kiếm tự nhiên của các công cụ tìm kiếm như Google, Bing, Yahoo, và nhiều hơn nữa. Mục tiêu của SEO là tăng lượng lưu lượng truy cập tự nhiên (organic traffic) đến trang web, đồng thời cải thiện khả năng hiển thị và sắp xếp của trang web trong kết quả tìm kiếm.

Quá trình SEO bao gồm việc tối ưu hóa các yếu tố nội dung và cấu trúc trang web, tạo liên kết (link building), cải thiện trải nghiệm người dùng, nghiên cứu từ khóa, và theo dõi và đánh giá hiệu quả của các chiến lược tối ưu hóa. SEO đòi hỏi sự hiểu biết về cách hoạt động của các công cụ tìm kiếm, quy tắc và tiêu chuẩn của ngành, và việc áp dụng các kỹ thuật phù hợp để tăng tính thân thiện với công cụ tìm kiếm của trang web.

Qua việc tối ưu hóa SEO, trang web có cơ hội cao hơn để xuất hiện trên các trang kết quả tìm kiếm hàng đầu, thu hút được lượng lớn người dùng tiềm năng và tăng khả năng tiếp cận đối tượng khách hàng mục tiêu.

-**SPA** không thân thiện với SEO như **MPA**

**+UX**

-**SPA** cho trải nghiệm tốt hơn, nhất là các thao tác chuyển trang

**-SPA** đem lại trải nghiệm tốt hơn trên thiết bị di động

**+Quá trình phát triển**

**-SPA** dễ dàng tái sử dụng code (chia component) => làm 1 giao diện có thể sử dụng API cho web và mobile luôn

-**MPA** có thể chia cimponent nhưng sẽ phức tạp hơn => phải tạo nhiều API cho mỗi thiết bị sử dụng

**+SPA( CSR ) và MPA (SSR)**

-**SPA** còn được biết đến là **CSR –** Client Side Rendering ! Lý do là khi chúng ta gửi một request thì trình duyệt sẽ tải (render) trang web trên máy của chúng ta (phía client)

-**MPA** còn được biết là **SSR** – Server Sider Rendering ! Lý do là khi chúng ta gửi request thì phía server sẽ render trang web và trả về trình duyệt ! Sau đó mới hiển thị lên máy chúng ta

**+Phụ thuộc Javascript**

-**SPA** phụ thuộc hoàn toàn vào Javascript (không có JS là phế luôn). Vì khi chúng ta phát triển theo **SPA** thì trang web chỉ có một file.html duy nhất và tất cả các trang còn lại được xây lên nhờ Javascript !

-**MPA** có thể không cần Javascript ! mà vẫn có thể sử dụng nhiều file.html tương tác với nhau vẫn oke ! Nhưng thời điểm này mà không có JS thì sao gọi là web nữa

**+Làm việc với server**

-**SPA** thì khi nhiều người tương tác thì trình duyệt sẽ render trên từng máy của client nên sẽ giúp ích cho server rất nhiều

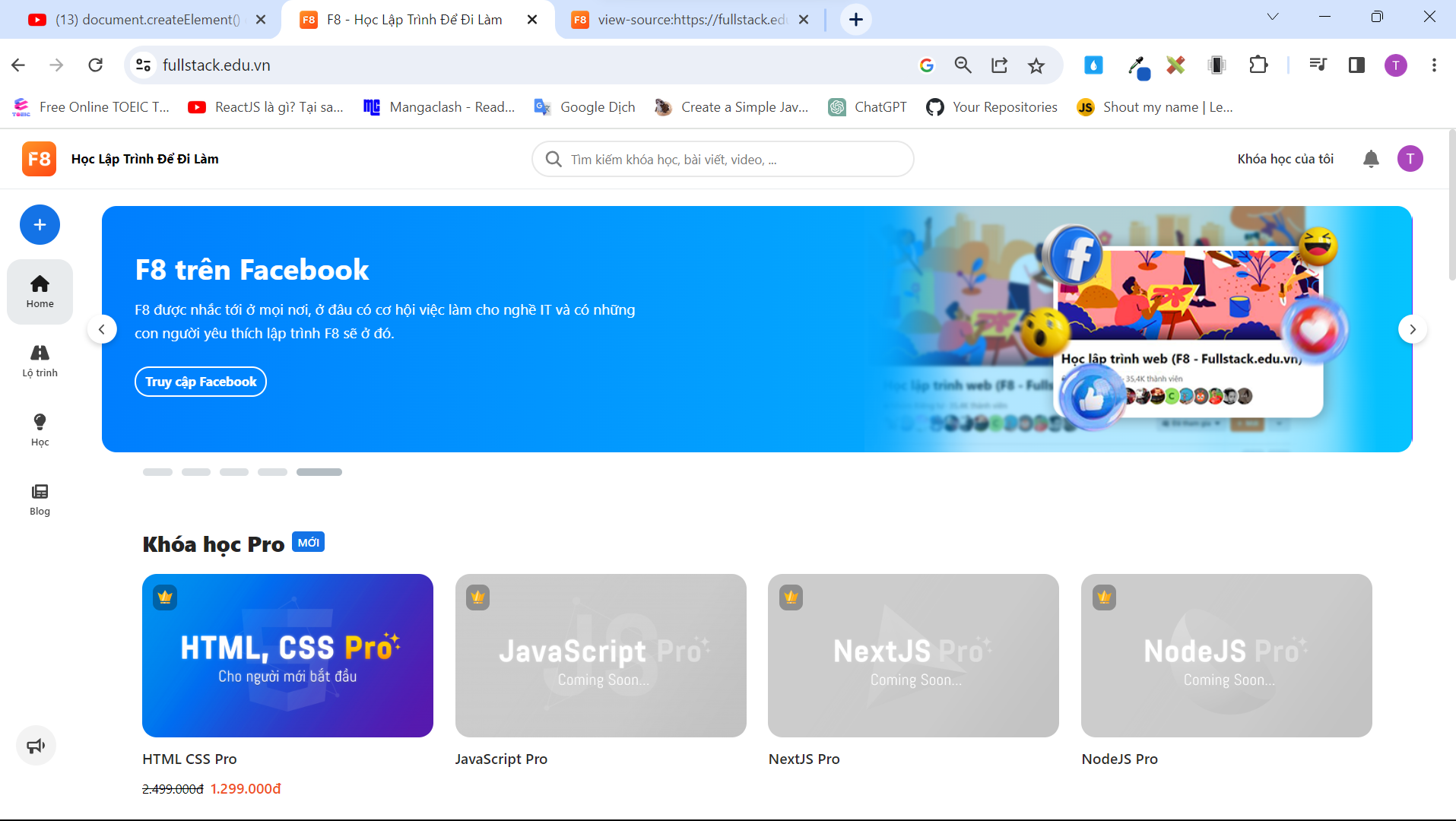
-**MPA** thì khi mỗi người thực hiện một request thì server phải nhận và render từng cái một ! Khiến server làm việc nhiều hơn nên chi phí khá tốn !

* 1. **Lựa chọn SPA hay MPA**

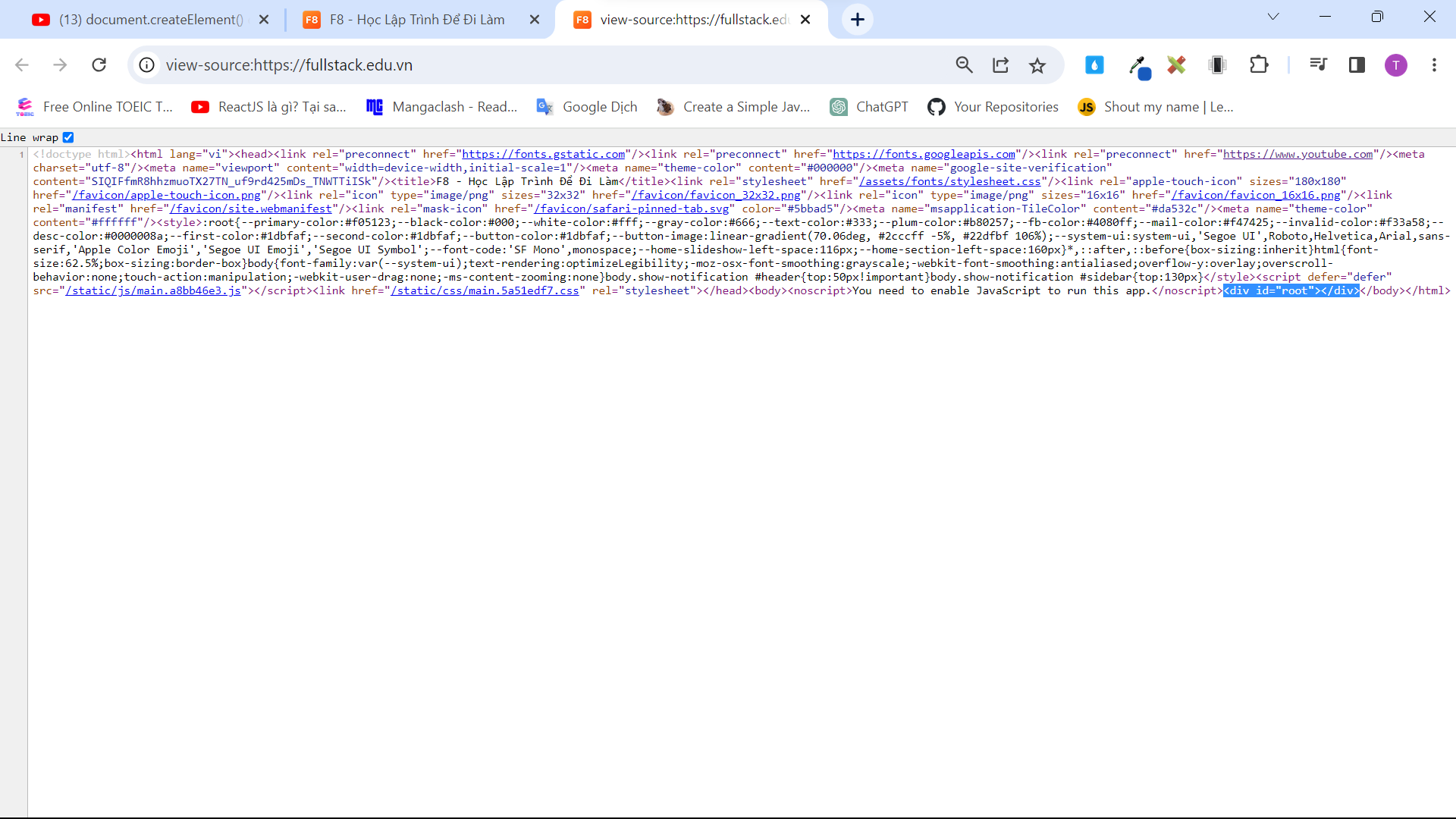
**-**Trường hợp nếu khách hàng yêu cầu built một trang web về giới thiệu công ty, sản phẩm,… mà không có nhu cầu mở rộng hoặc phát triển trang web hay chạy trên mobile thì chúng ta nên sài **MPA** để có thể phát triển ngay và luôn !

**-**Còn nếu như khách hàng muốn mở rộng thêm nhiều tính năng mới lạ, phát triển luôn bên mobile và muốn trang web phải mượt mà, trải nghiệm người dùng tốt nhất thì chúng ta hãy dùng **SPA**

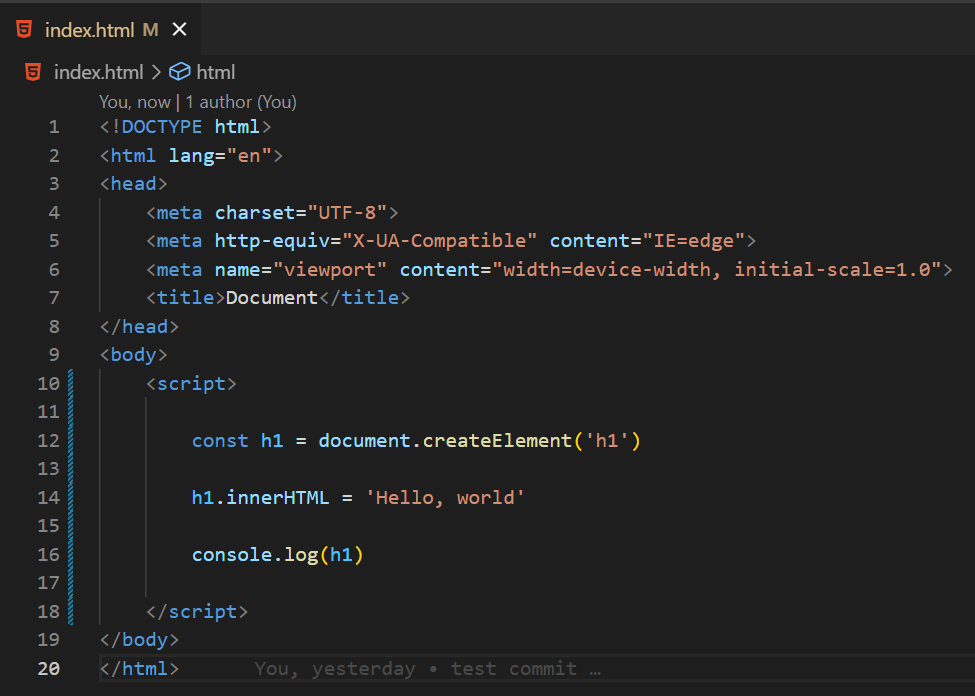
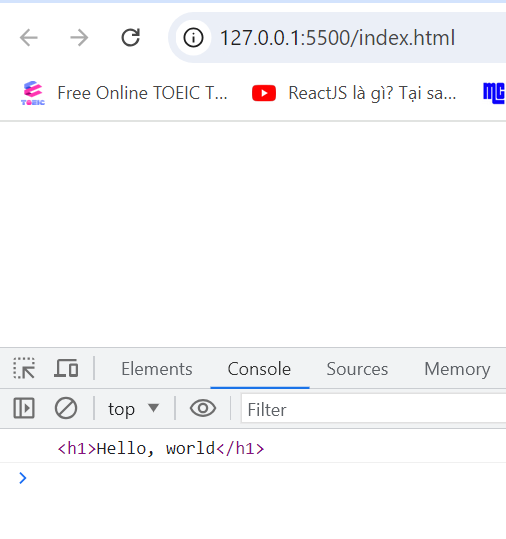
1. **Document.creatElement() (Tạo Element trong thẻ <script>)**



-Chúng ta có trang web của fullstack.edu.vn ! Có thể thao tác qua lại trên thanh điều hướng



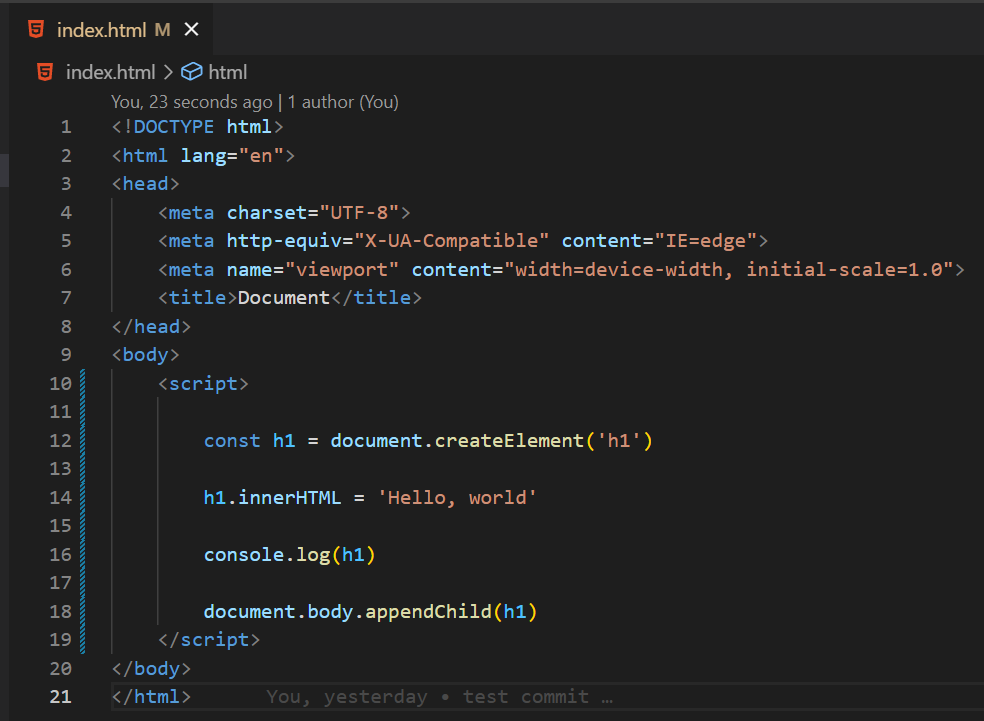
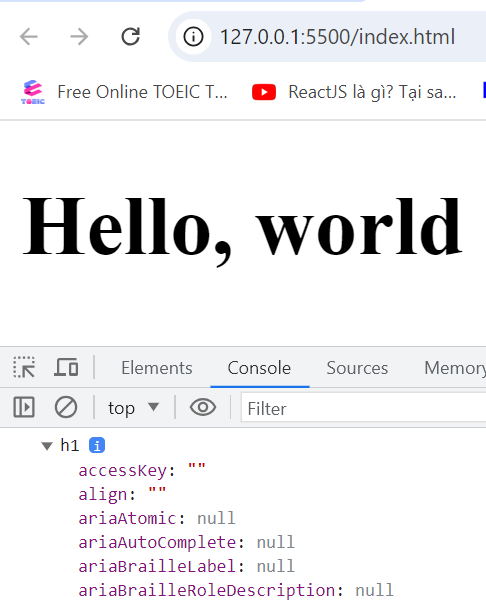
-Nhưng điều thú vị ở đây là ! trong phần source code của trang web chúng ta không hề thấy cả núi code HTML cả mà đa phần là JavaScript (chỉ vài dòng)! Và chỉ có duy nhất 1 thẻ <div> có id là **root** ! Để code được giao diện trên chắc cả ngàn dòng code mà sao lại chỉ hiện chó 1 thẻ <div> duy nhất ! đó là cách reactJS làm việc nói chính xác hơn là **SPA**

** **

**-**Để mô phỏng lại thì chúng ta sẽ làm ví dụ nhỉ ! Chúng ta tạo một element là <h1> bằng phương thức **document.creatElement()** và cho nó text Node = ‘Hello, world’ thông qua thuộc tính **innerHTML** sau đó in ra màn hình console !

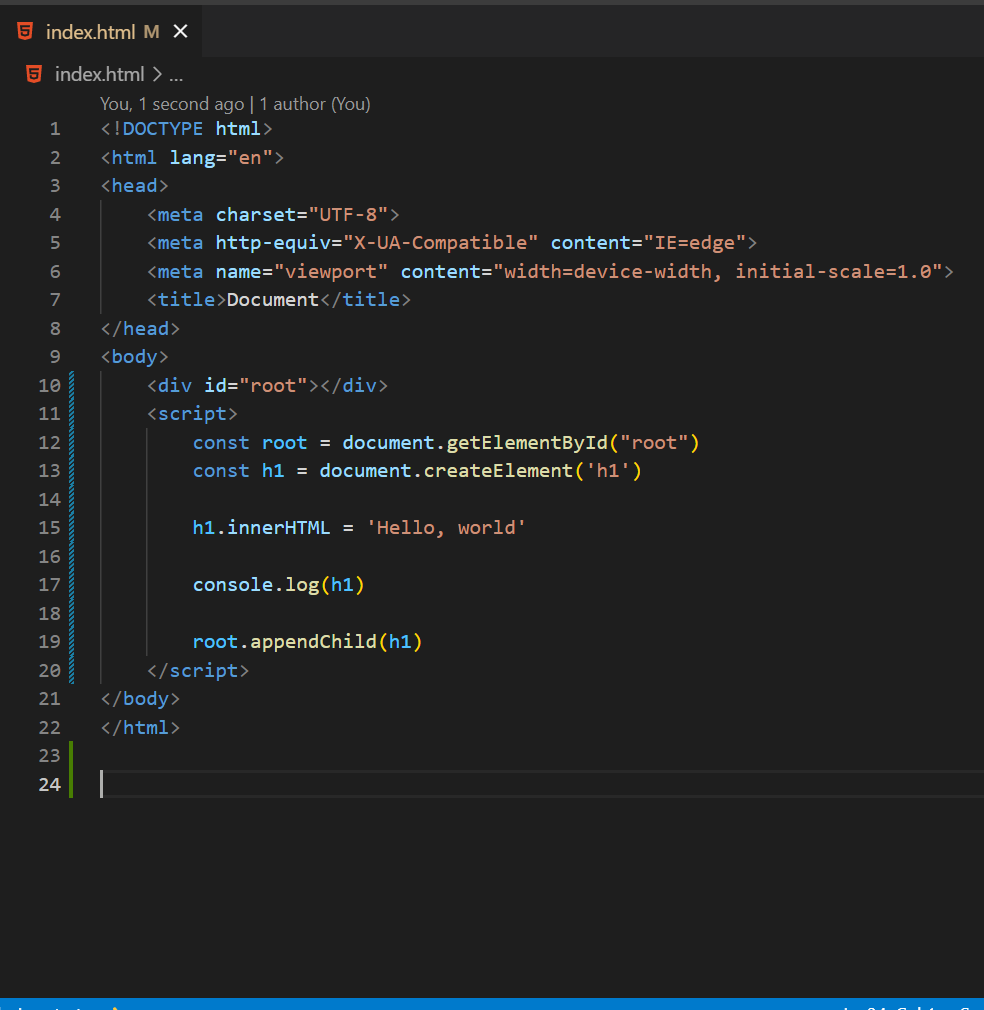
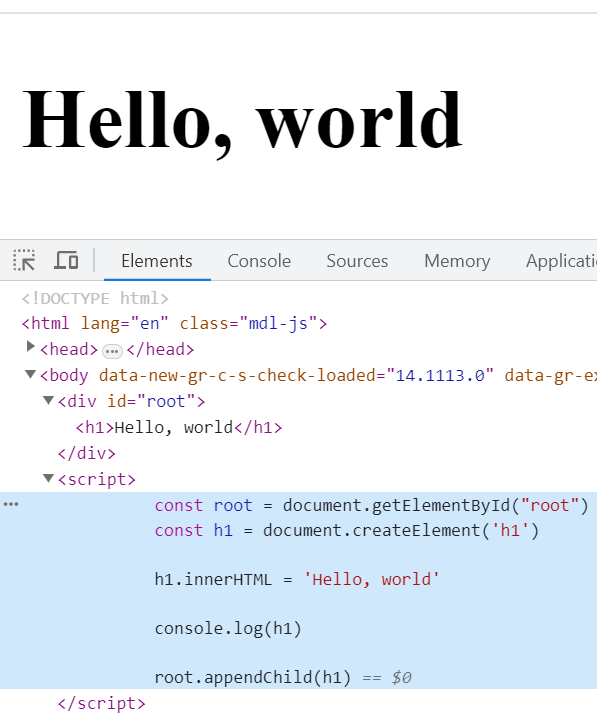
**-**Nhưng để ý là phía trên màn hình giao diện của chúng ta vẫn trắng tinh chưa có gì !

* 1. **Document.appendChild() (Thêm element con vào element cha)**

-Để thêm thẻ <h1> vừa tạo hiển thị lên trang web thì ta sử dụng phương thức **appendChild()** và **document.body** có nghĩa là đối tượng **document** đi đến thẻ **<body>** và thêm con cho nó (kiến thức DOM tree)

-Nếu mở source code trang lên thì sẽ thấy không có dòng code **<h1>** nào cho đến khi **file.js** được tạo mới hiển thị **<h1>**

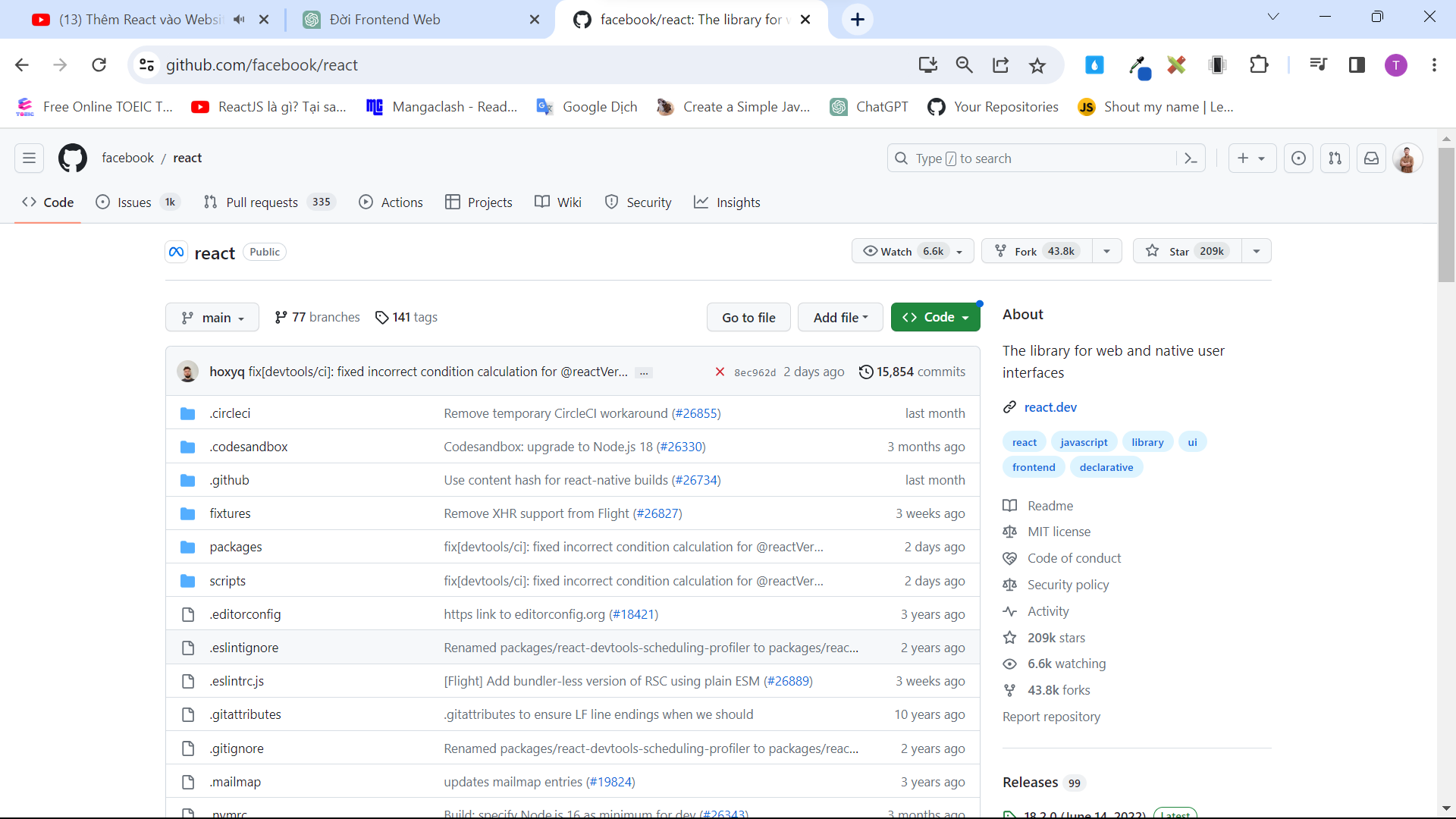
 

-Để làm giống với fullstact.edu.vn thì chúng ta chỉ việc tạo một the **<div>** có **id = “root”** và gọi đến Element root đó thay **document.body** thành **root** thôi

-Đó là cách **SPA** làm việc ! và thư viên ReactJS cũng hoạt động như vậy

1. **Thêm thư viện ReactJS vào Website**
   1. **Cách thư viện được lưu trữ và phân phối**

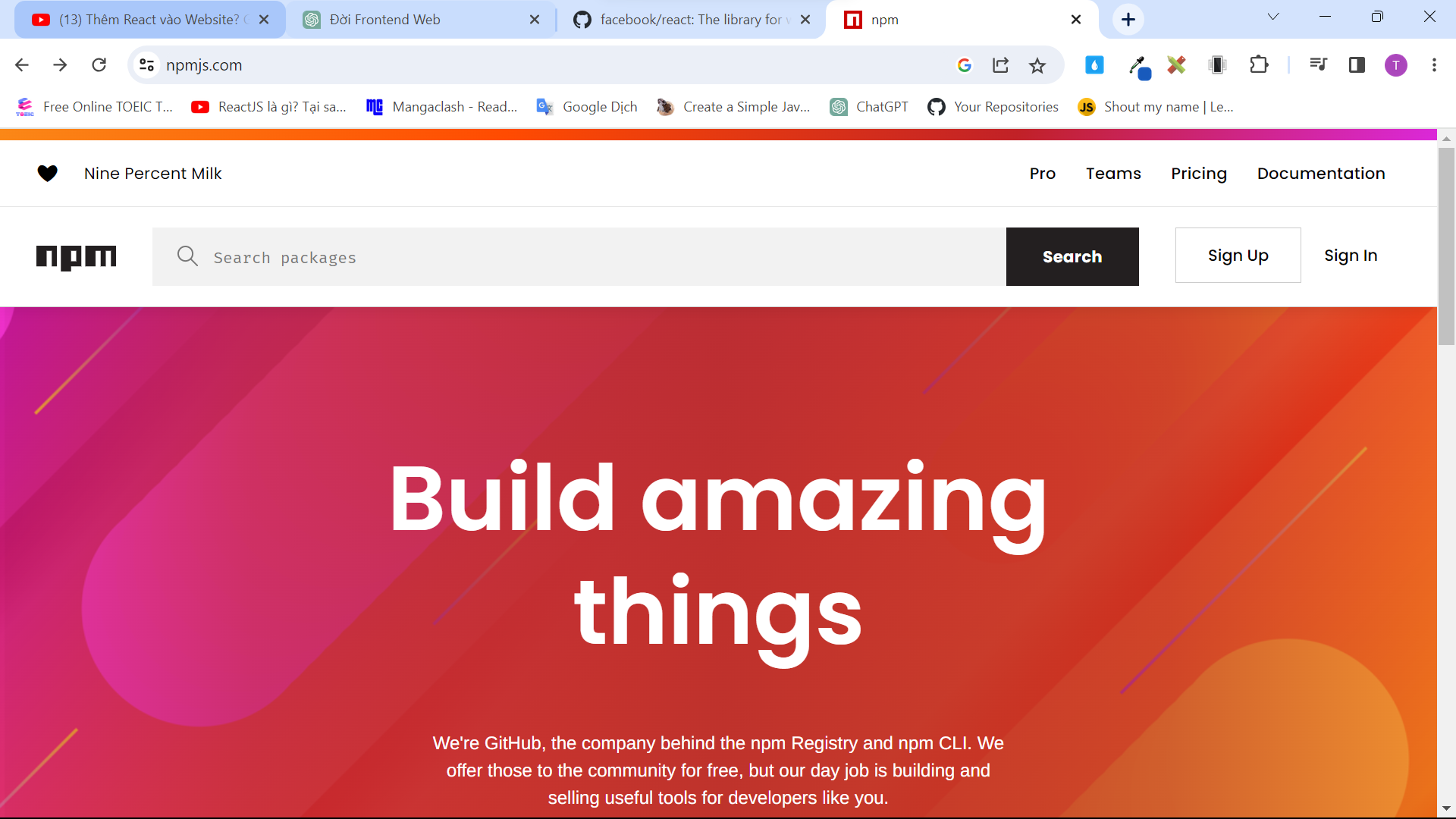
**+Github**

****

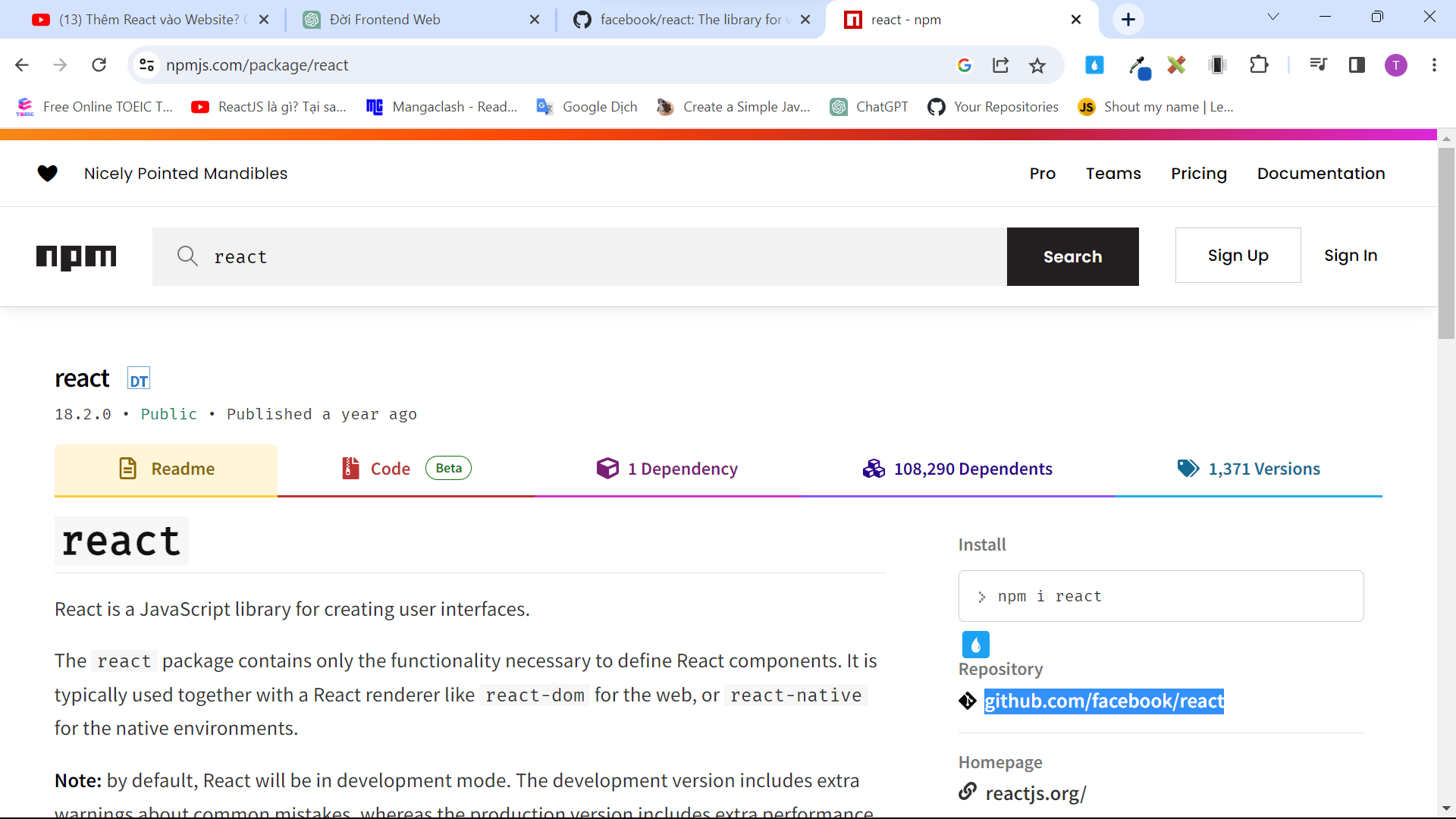
-Là nơi mà lưu trữ gốc của thư viện React ! Chúng ta có thể lấy về và tạo ra thư viện react riêng cho mình cũng được !

-Nhưng không nhất thiết phải tải hết của người ta về ! mà chỉ sử dụng một vài phương thức thôi !

**+NPMJS**

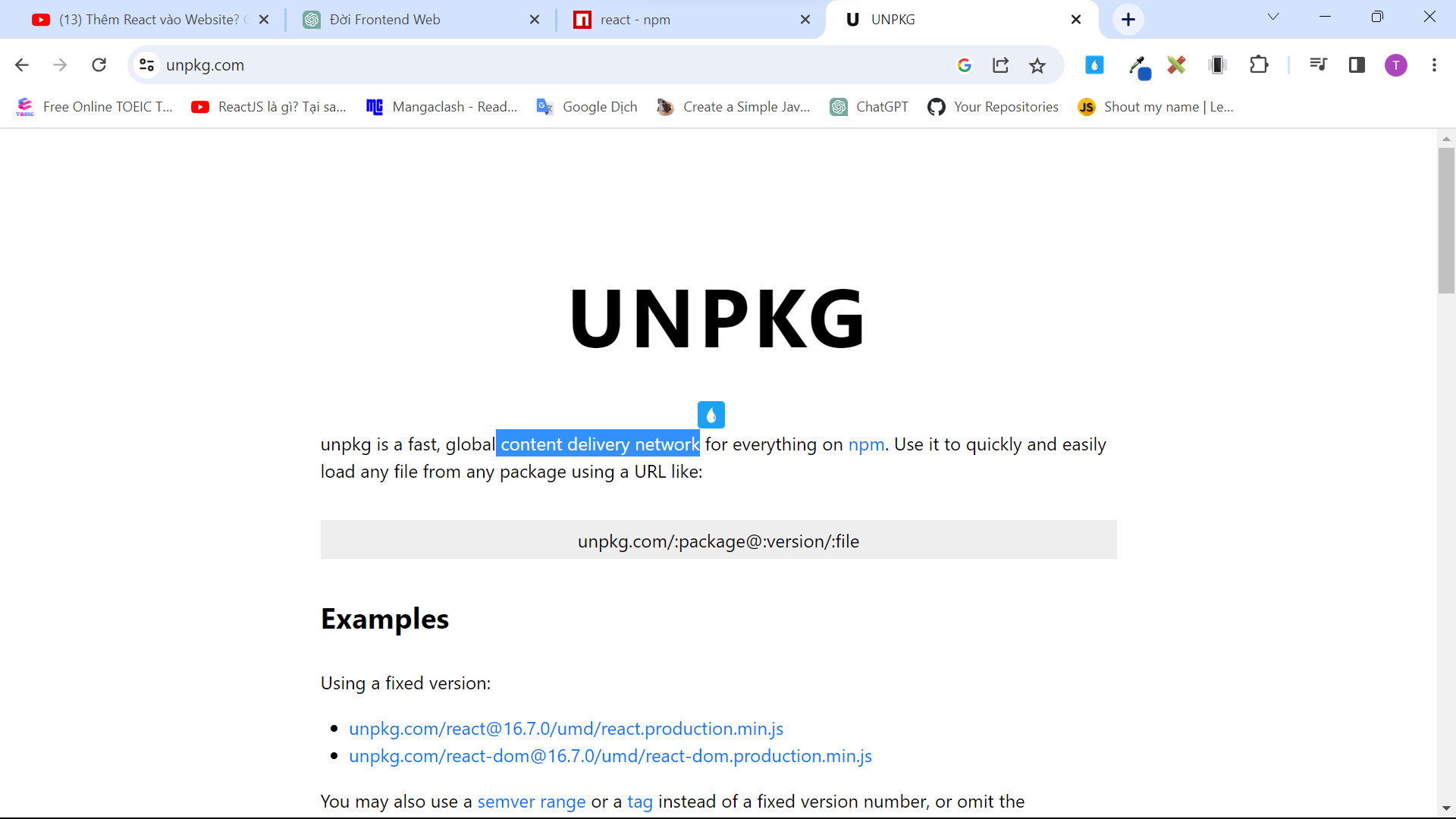
****

**-NPM** là Node Package Manager là nơi quản lý thư việnvà công bố cho người dùng tải về



-Chúng ta nhập từ khóa react vào và chọn thư viện react !để ý nguồn đúng với link github cho thấy cả 2 là một thôi !

**+UNPKG**

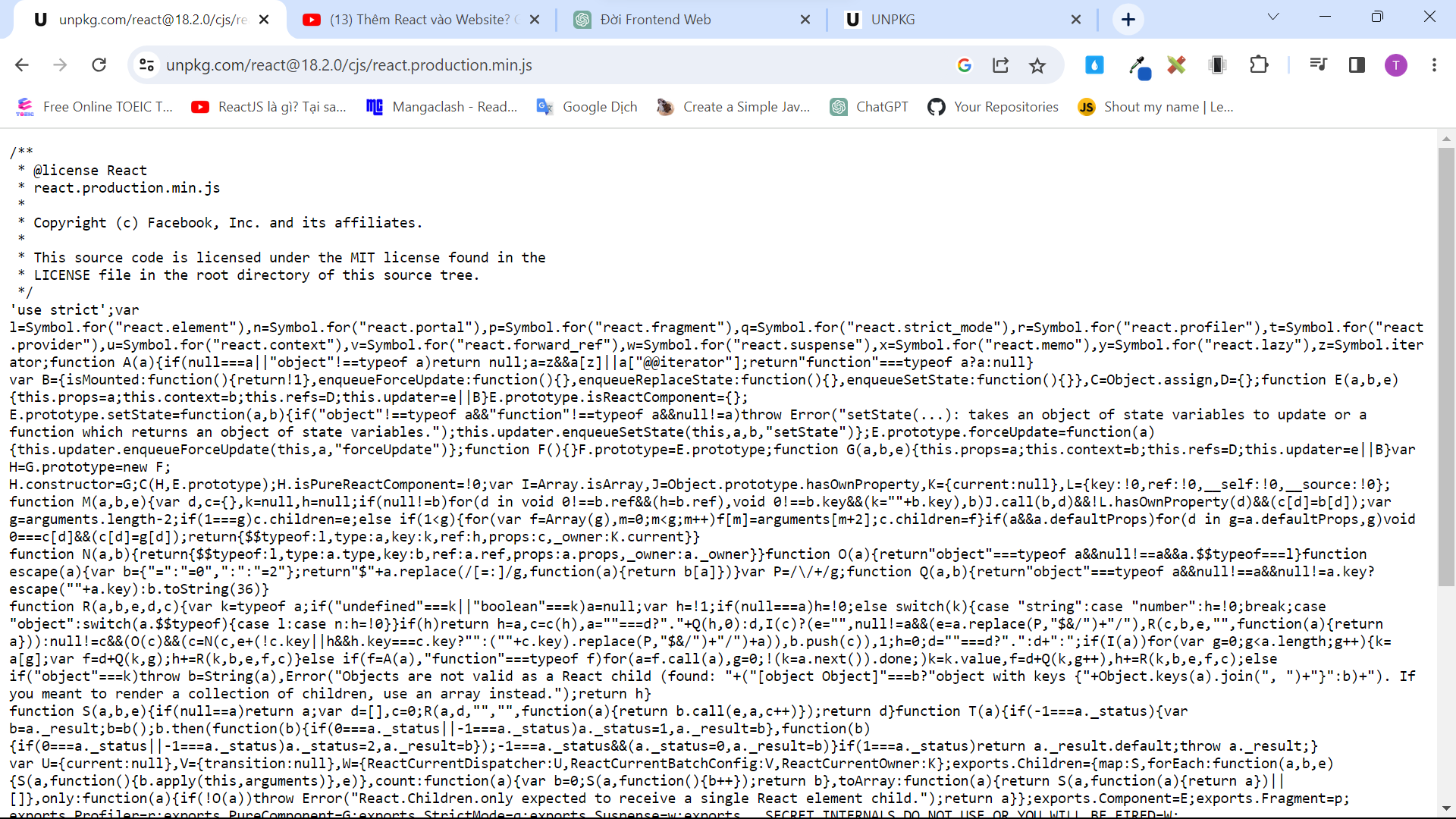
****

**-**Có nhiều cách để tải thư viện bên **npm** về lắm nhưng nhanh nhất vẫn là sài **cdn (**tự tìm hiểu nhé**)**

****

**-**Chúng ta có thể gõ **unpkg.com/tên thư viện** để có thể biết nó có những phiên bản gì

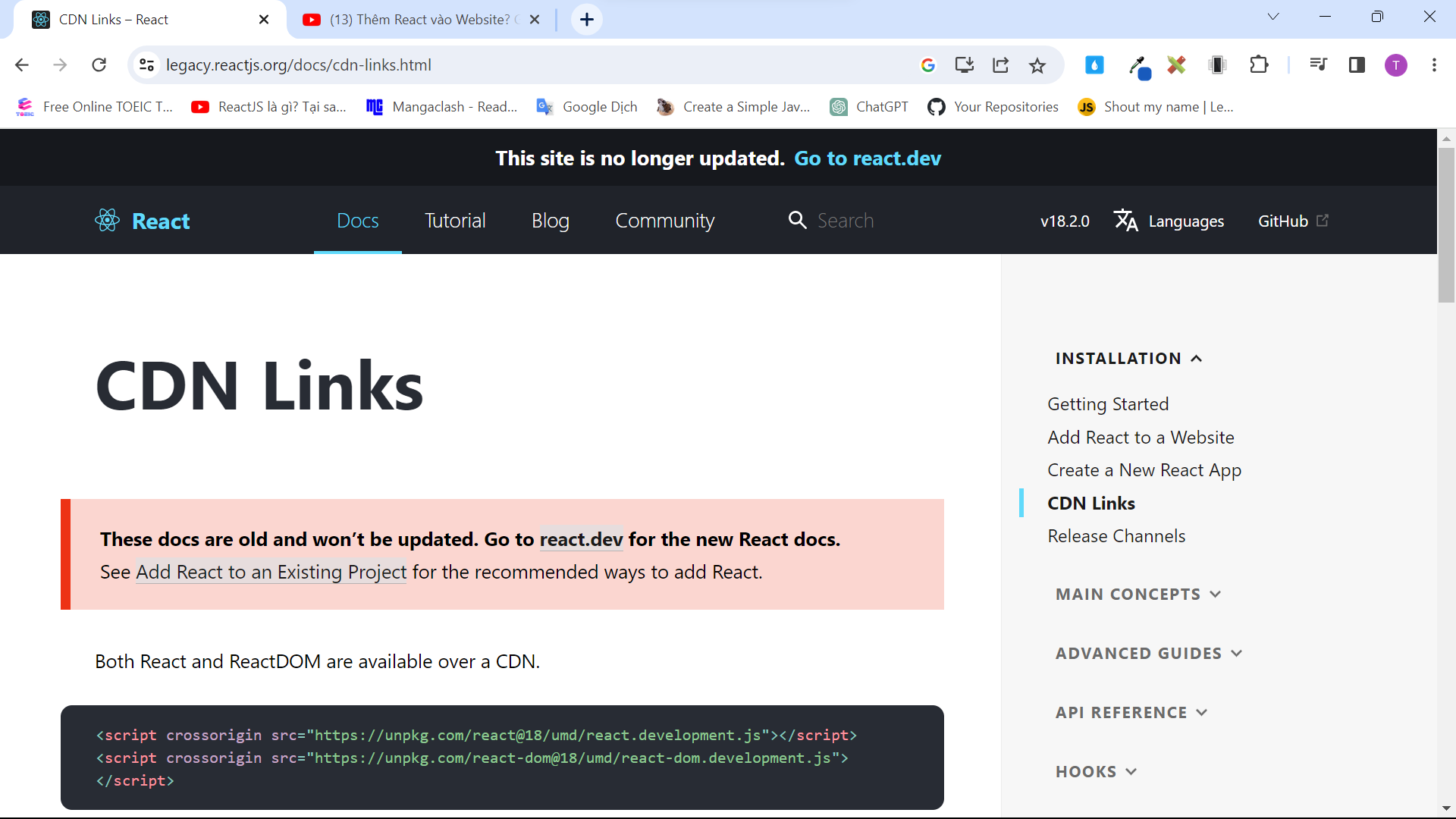
**-**Ở đây gồm phiên bản cho dev (báo lỗi, bug) và cho người dùng cuối (gọn nhẹ, dễ dàng sử dụng) !

****

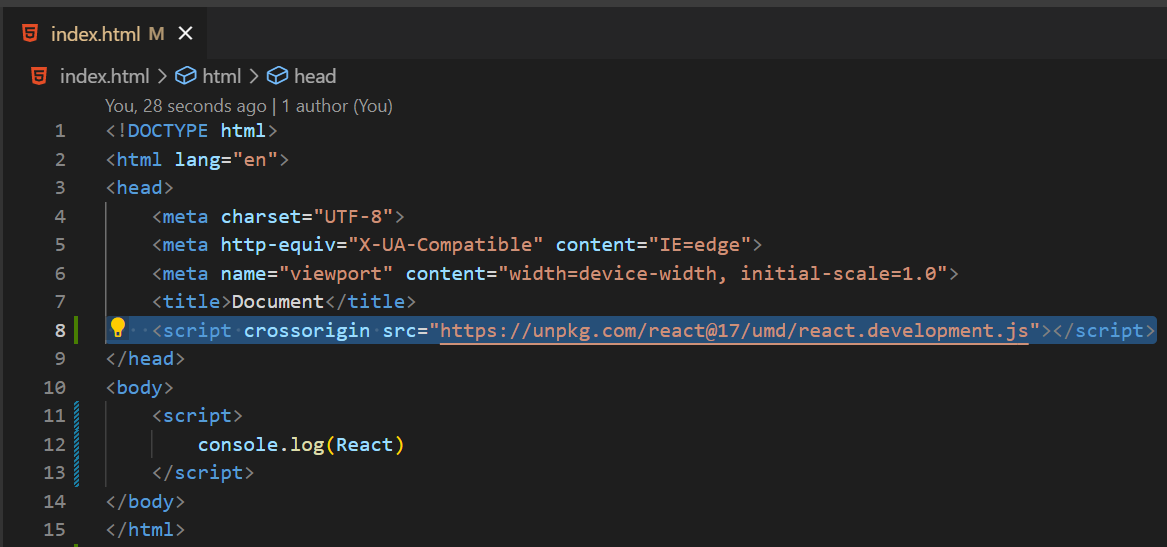
**-**Chúng ta có thể đi sâu hơn để thấy toàn bộ thư viện React chỉ vỏn vẹn một file ! Nhưng đội ngũ kỹ sư của facebook đã bỏ rất nhiều kinh nghiệm và công sức mới có được cho chúng ta sài

**-**Vì vậy suy cho cùng chúng ta chỉ có thể sử dụng các phương thức của thư viện để build lên sản phẩm của mình chứ không ai code giùm ta gì cả và hoàn toàn là 100% code Javascript ! nên chưa có nền tản cứng về Javascript thì gãy !

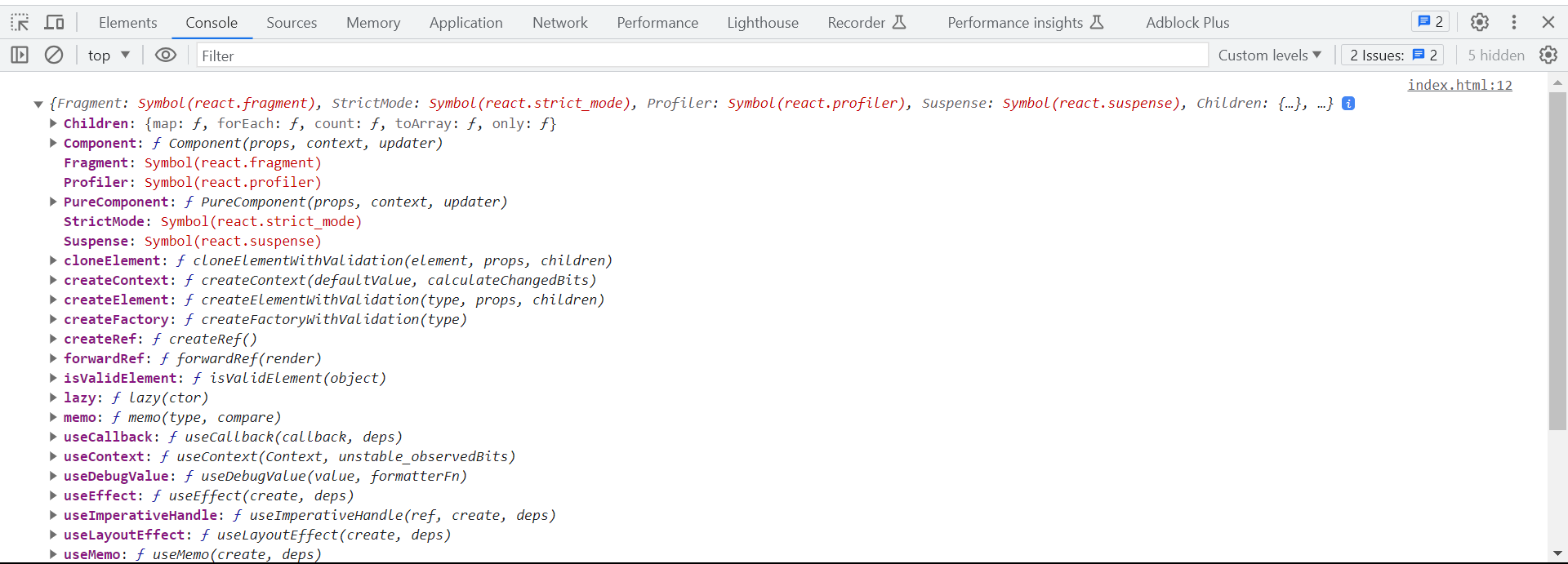
* 1. **Thêm thư viện ReactJS**

****

**-**Để thêm thư viện **ReactJS** thì chúng ta sẽ truy cập vào trình duyệt gõ cdn reactJSlà ra ! (**TRONG TÀI LIỆU VỀ REACTJS THÌ SẼ ÁP DỤNG PHIÊN BẢN 17.0.2**)

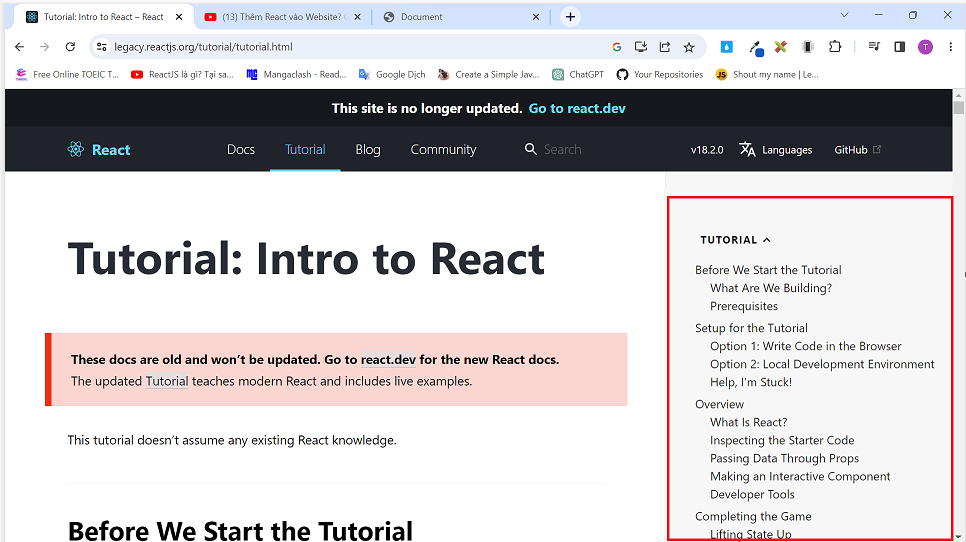


-Sau đó dán **cdn** vào **file.html** để kiểm tra add thư viện thành công thì chúng ta thử in đối tượng React xem ! ( =)) nếu khác phiên bản thì đổi thành 17 sài nhé )



-Nếu ra được tất cả thuộc tính và phương thức của React thì đã add thành công ! Sau đó chúng ta chỉ việc chõ đến sài thôi !

* 1. **React official docs**



**-**Trong trang chủ react thì có phần Tutorial khá bài bản và đi từ gốc đến ngọn ! nên đọc để nắm rỏ react làm việc như thế nào !