# Vai trò của HTML, CSS, Javascript

HTML (HyperText Markup Language - Ngôn ngữ đánh dấu siêu văn bản), CSS (Cascading Style Sheets - Các tập tin định kiểu theo tầng) và JavaScript là 3 ngôn ngữ để phát triển web mà trình duyệt có thể hiểu. Chúng là các ngôn ngữ khác nhau nhưng có quan hệ mật thiết với nhau, mỗi ngôn ngữ được thiết kế cho những nhiệm vụ cụ thể. Việc hiểu rõ được cách chúng làm việc với nhau sẽ giúp bạn sớm trở thành một nhà phát triển web chuyên nghiệp.

Trước tiên, bạn cần nắm được:

* HTML dùng để thêm ý nghĩa cho nội dung bằng cách đánh dấu nó
* CSS dùng để định dạng cho nội dung đã được đánh dấu bởi HTML
* JavaScript dùng để thêm chức năng giúp người dùng có thể tương tác nhiều hơn với trang web

Hãy tưởng tượng HTML là phần khung xượng của một trang web, CSS là những gì bạn nhìn thấy trên trang web và JavaScript là các hành vi có thể thao tác trên trang web.



HTML là phần cấu trúc, CSS hoàn thiện về thẩm mỹ, JavaScript làm tính năng tương tác.

Ví dụ: bạn có thể đánh dấu một đoạn văn bản với code HTML như này:

<p id="paragraph">Hôm nay là một ngày đẹp trời!</p>

Sau đó, bạn có thể đặt kích thước và màu sắc cho đoạn văn bản với CSS sau:

#paragraph {

font-size: 18px;

color: red;

}

Cuối cùng, nếu bạn muốn nhấn vào đoạn văn bản thì nội dung của nó sẽ thay đổi, thêm code JavaScript sau:

const p = document.getElementById('paragraph');

p.onclick = function() {

this.innerText = 'Bạn đã CLICK vào đoạn văn bản!';

};

Tóm tắt :

* HTML, CSS và JavaScript là 3 ngôn ngữ duy nhất mà trình duyệt có thể hiểu
* HTML viết tắt của HyperText Markup Language, nghĩa là Ngôn ngữ đánh dấu siêu văn bản
* CSS viết tắt của Cascading Style Sheets, nghĩa là Các tập tin định kiểu theo tầng
* JavaScript là một ngôn ngữ lập trình, giúp thêm tính tương tác cho trang web
* Ví HTML như phần khung của một ngôi nhà, thì CSS đóng vai trò trang trí cho ngôi nhà trở nên đẹp mắt, và JavaScript giúp ngôi nhà có nhiều tiện nghi hiện đại hơn.

# chi tiết về ngôn ngữ html

## Giới thiệu HTML

HTML (HyperText Markup Language, ý nghĩa: Ngôn ngữ đánh dấu siêu văn bản). Đúng vậy, HTML là ngôn ngữ đánh dấu, nó cung cấp cú pháp để đánh dấu nội dung và không chứa logic lập trình chẳng hạn như if...else.

HTML là ngôn ngữ đánh dấu, HTML không phải là một ngôn ngữ lập trình.

## Ngôn ngữ đánh dấu

Hiểu đơn giản, ngôn ngữ đánh dấu sẽ giúp bạn có thể đánh dấu nội dung theo cách mang lại ý nghĩa cho nội dung đó. Hãy tưởng tượng, bạn đang sử dụng một trình soạn thảo và muốn in nghiêng một đoạn văn bản.

Việc bạn chọn đoạn văn và in nghiêng chúng chính là một ví dụ cho việc đánh dấu, và trong HTML, chúng ta sẽ đánh dấu nội dung bằng cách sử dụng thẻ mở và thẻ đóng.

Hôm nay là một ngày đẹp trời

Tương tự như ví dụ trước, nhưng sử dụng HTML để đánh dấu:

Hôm nay là <i>một ngày đẹp trời!</i>

## Siêu văn bản

Siêu văn bản (hypertext) là loại văn bản chứa nhiều loại dữ liệu khác nhau như: văn bản, âm thanh, hình ảnh, video, … và các siêu liên kết đến các siêu văn bản khác. Trong khi đó, văn bản thông thường (không phải siêu văn bản) chỉ bao gồm nội dung văn bản và hình ảnh. Ví dụ: một tờ báo, các trang trong một cuốn sách, … không phải là siêu văn bản.

## Tóm tắt

* HTML là ngôn ngữ đánh dấu siêu văn bản (HyperText Markup Language)
* HTML là ngôn ngữ đánh dấu, HTML không phải là một ngôn ngữ lập trình
* Siêu văn bản (hypertext) là loại văn bản chứa nhiều loại dữ liệu khác nhau như: văn bản, âm thanh, hình ảnh, video, … và các siêu liên kết
* Cú pháp cơ bản của HTML bao gồm thẻ mở và thẻ đóng. Ví dụ: Hôm nay là <i>một ngày đẹp trời!</i> thì <i> là thẻ mở, </i> là thẻ đóng.

# cấu trúc tiêu chuẩn file html

## DOCTYPE

Đây là khai báo kiểu cho tài liệu HTML, còn được gọi là DOCTYPE, là dòng code đầu tiên cần được khai báo trong mọi file HTML.

Khai báo DOCTYPE có ý nghĩa hướng dẫn cho trình duyệt web hiểu về phiên bản HTML mà trang web đang sử dụng. Điều này đảm bảo rằng trang web của bạn được xử lý giống nhau trên nhiều trình duyệt khác nhau.

DOC: document

TYPE: loại

Đảm bảo rằng bạn đặt DOCTYPE ngay đầu tài liệu HTML của mình mà không có bất cứ nội dung gì trước đó để đảm bảo khai báo này hoạt động đúng đắn trên mọi trình duyệt.

## Thẻ HTML

Toàn bộ trang web được bao bọc trong cặp thẻ html (gồm thẻ mở <html> và thẻ đóng </html>). Thẻ <html> (hay phần tử <html>) còn được gọi là phần tử gốc (root), đây là phần tử cấp cao nhất vì tất cả các phần tử khác đều là con của phần tử này.

Trong thực tế, các trang web sẽ sử dụng thêm thuộc tính lang trong thẻ mở html, điều này nhằm chỉ định ngôn ngữ được sử dụng trong nội dung của trang web.

A screenshot of a computer

Description automatically generated

<https://www.w3schools.com/tags/ref_language_codes.asp>

## Thẻ HEAD

Thẻ <head> là nơi chứa các thẻ meta, là các thẻ cung cấp thêm thông tin hiển thị của trang web. Ví dụ: tiêu đề trang web, các đường link tới file CSS, khả năng thu/phóng của trang web, …

## Thẻ BODY

Thẻ <body> là nơi chứa toàn bộ nội dung hiển thị của trang web. Mọi nội dung bạn từng nhìn thấy trên các trang web đều nằm trong thẻ <body>.

## Tóm tắt

* Mỗi file HTML có đuôi mở rộng là html, ví dụ: index.html. Cấu trúc gồm 4 phần: DOCTYPE, <html>, <head> và <body>
* DOCTYPE là dòng đầu tiên cần được khai báo trong mọi file HTML.
* Thẻ <html> chứa toàn bộ nội dung của trang web.
* Thẻ <head> chứa các thẻ meta, thẻ <body> chứa nội dung được hiển thị của trang web.
* Thuộc tính lang sử dụng để chỉ định ngôn ngữ của trang web. Các công cụ dịch ngôn ngữ và trình đọc màn hình sẽ sử dụng thuộc tính này.

Lưu ý: tại sao phải đặt file là index.html – Vì mặc định web server sẽ tải file index.html

# tìm hiểu thẻ meta

Thẻ <meta> được sử dụng để cung cấp thêm thông tin và dữ liệu cho trang web.

Thẻ meta là một trong số thẻ tự đóng (self-closing tag), cách viết của nó chỉ đơn giản như ví dụ sau:

<meta />

Khi sử dụng thẻ <meta> chúng ta cần khai báo thêm thuộc tính để việc sử dụng thẻ này trở nên có ý nghĩa.

Một số trường hợp sử dụng thẻ meta phổ biến:

* Chỉ định bảng mã ký tự cho trang web
* Cung cấp mô tả trang web cho công cụ tìm kiếm
* Chỉ định cách hiển thị trang web trên nhiều thiết bị (PC, tablet, mobile)
* Cung cấp thông tin của trang web cho trình thu thập dữ liệu của mạng xã hội (như Facebook)

A screenshot of a computer code

Description automatically generated

## Meta charset

Đây là thẻ cần được khai báo tại vị trí đầu tiên trong thẻ <head>:

<head>

<meta charset="UTF-8" />

...

</head>

Thẻ meta với charset="UTF-8" có tác dụng hỗ trợ hiển thị các ký tự nằm ngoài khoảng từ A - Z. Ví dụ: tiếng Việt có dấu, tiếng Trung, tiếng Nhật, riếng Ả Rập, …

Nếu bạn không khai báo meta charset tại vị trí đầu tiên trong thẻ <head> thì trong một số trường hợp nội dung trang web chứa ký tự phức tạp, trình duyệt sẽ cố gắng đoán bảng mã ký tự và có thể gây ra lỗi hiển thị không mong muốn.

Điều trên ngày nay khó xảy ra. Các trang web ngày nay hầu hết được triển khai trên các máy chủ web có hỗ trợ UTF-8. Vì vậy, khai báo <meta charset="UTF-8" /> là không cần thiết trong trường hợp này.

Tuy nhiên, không vì lý do trên mà chúng ta không sử dụng thẻ meta charset. Thực tế, vẫn có những máy chủ không mặc định hỗ trợ charset. Để đảm bảo tính thống nhất về bảng mã ký tự cho trang web của bạn, mặc định chúng ta luôn sử dụng thẻ meta charset nhé.

## Meta title

Tuy thẻ này không có từ meta nhưng vẫn nằm trong nhóm thẻ meta. Đây là thẻ chứa tiêu đề của trang web, tiêu đề này được sử dụng trong một số trường hợp:

Hiển thị tiêu đề trang web trên thanh tab của trình duyệt

Hiển thị tiêu đề trong kết quả của công cụ tìm kiếm (ví dụ: Google Search)

Ví dụ sử dụng thẻ meta <title>:

<title>HTML, CSS từ Zero đến Hero | by F8</title>

## Meta viewport

Để tối ưu hóa việc hiển thị giao diện web trên nhiều thiết bị có kích cỡ màn hình khác nhau (như PC, tablet, mobile) bạn sẽ cần sử dụng thẻ meta viewport.

Ví dụ sử dụng thẻ meta viewport:

<meta name="viewport" content="width=device-width, initial-scale=1.0" />

Nếu trang web của bạn không có thẻ meta viewport thì khi mở trên thiết bị di động nội dung sẽ bị thu nhỏ, bạn phải phóng to để dễ dàng đọc được. Các trang web được xây dựng vào khoảng những năm 2010 thường gặp vấn đề này.

Sau đó, trào lưu thiết kế web đáp ứng (hay: responsive web design) xuất hiện. Tới ngày nay, hầu hết mọi trang web đều được tối ưu hiển thị trên nhiều thiết bị, bao gồm cả mobile. Đương nhiên, các trang web được tối ưu cho đa thiết bị đều sử dụng thẻ meta viewport.

Ví dụ:

* Bản thiết kế giao diện trang web:
* [https://www.figma.com/file/n8EZ1ywmY8YJQnS2JjW3Bg/[Medium]----Findtrend-Webflow-(Free-Code)](https://www.figma.com/file/n8EZ1ywmY8YJQnS2JjW3Bg/%5bMedium%5d----Findtrend-Webflow-(Free-Code))
* Trang web bị thiếu thẻ meta viewport:
* <https://htmlcss-projects.fullstack.edu.vn/project-5/missing-meta-viewport.html>
* Trang web có sử dụng thẻ meta viewport:
* <https://htmlcss-projects.fullstack.edu.vn/project-5/>

Trên thiết bị di động, trang web bị thiếu meta viewport chữ bị thu nhỏ, phải phóng to mới dễ dàng đọc được

Quét mã QR trên điện thoại để truy cập (trang thiếu meta viewport):

A qr code on a white background

Description automatically generated

Quét mã QR trên điện thoại để truy cập (trang có meta viewport):

A qr code on a white background

Description automatically generated

Trong thẻ meta viewport, giá trị width=device-width nhằm yêu cầu trình duyệt đặt chiều rộng của trang web bằng chiều rộng của thiết bị (PC, tablet, mobile). Giá trị

initial-scale=1.0 nhằm giữ đúng kích thước/tỉ lệ mặc định của trang web.

## Meta open graph

Đây là các thẻ meta được trình thu thập nội dung của Facebook sử dụng, nhằm lấy thông tin của trang web phục vụ việc hiển thị khi trang web được chia sẻ trên mạng xã hội này.



Khi chia sẻ trang web trên mạng xã hội Facebook:

A screenshot of a social media post

Description automatically generated

<meta

property="og:url"

content="https://fullstack.edu.vn/blog/tong-hop-cac-san-pham-cua-hoc-vien-tai-f8.html"

/>

<meta property="og:type" content="article" />

<meta property="og:title" content="Tổng hợp các sản phẩm của học viên tại F8 | by Sơn Đặng | F8" />

<meta

property="og:description"

content="Bài viết này nhằm tổng hợp lại các dự án mà học viên F8 đã hoàn thành và chia sẻ trên nhóm Học lập trình web F8. Các dự án dưới..."

/>

<meta

property="og:image"

content="https://files.fullstack.edu.vn/f8-prod/blog\_posts/65/6139fe28a9844.png"

/>

Tất nhiên, đây mới chỉ là phần giới thiệu để các bạn nắm được ý nghĩa của các thẻ meta thường được sử dụng. Để có thể triển khai được một trang web lên mạng Internet và có thể chia sẻ trang web đó lên Facebook các bạn sẽ cần hoàn thành khóa học này.

## Tóm tắt

* Thẻ meta được sử dụng để cung cấp thêm thông tin và dữ liệu cho trang web. Các thẻ meta thường được sử dụng bởi trình duyệt, các công cụ tìm kiếm và trình thu thập dữ liệu của các mạng xã hội.
* Thẻ meta là một trong số thẻ tự đóng (self-closing tag), nhờ vậy không cần thêm thẻ đóng khi sử dụng thẻ meta.
* Thẻ meta charset nhằm chỉ định bảng mã ký tự cho trang web (ví dụ: <meta charset="UTF-8" />). UTF-8 hỗ trợ hiển thị các ký tự ngoài khoảng A - Z. Luôn nhớ đặt thẻ meta charset tại vị trí đầu tiên trong thẻ <head>.
* Thẻ meta <title> cung cấp tiêu đề trang web hiển thị trên tab trình duyệt, hiển thị trong kết quả tìm kiếm (ví dụ: Google Search).
* Thẻ meta viewport để hỗ trợ trang web hiển thị trên các thiết bị có kích cỡ màn hình khác nhau (PC, tablet, mobile). Nếu thiếu thẻ meta viewport, khi truy cập trên thiết bị di động, nội dung có thể bị thu nhỏ và cần phải phóng to để dễ dàng đọc được.
* Thẻ meta Open Graph cung cấp thông tin cho trình thu thập dữ liệu của Facebook, hỗ trợ hiển thị thông tin trang web khi được chia sẻ trên mạng xã hội này.

# Boilerplate

## Boilerplate là gì?

Trong lập trình, boilerplate được coi là những đoạn code được tiêu chuẩn hóa và tái sử dụng ở nhiều nơi mà ít khi có sự thay đổi. Thay vì chúng ta phải viết đi viết lại những đoạn code đó nhiều lần thì boilerplate thường được tạo sẵn bằng một cách nào đó giúp tiết kiệm thời gian hơn khi sử dụng (ví dụ như: phím tắt, cú pháp tắt, vv).

## Tạo cấu trúc file HTML

Qua những bài học trước bạn phải tạo cấu trúc file HTML bằng cách viết code thủ công (ví dụ như: DOCTYPE, <html>, <head> và <body>). Tin vui là bạn không cần phải viết lặp lại chúng nhiều lần mỗi khi tạo file HTML mới nữa.

A screenshot of a computer

Description automatically generated

A screen shot of a computer program

Description automatically generated

Đây là một trong số tính năng của Emmet mà bạn sẽ được học sau một vài chương tới.

Trong hầu hết mọi trường hợp bạn chỉ cần thay đổi lại thuộc tính lang và nội dung thẻ <title>.

## Tóm tắt

Boilerplate được coi là những đoạn code được tiêu chuẩn hóa và tái sử dụng ở nhiều nơi mà ít khi có sự thay đổi

Có thể tạo nhanh cấu trúc HTML cơ bản bằng cách gõ ! và sau đó nhấn phím Tab (được hỗ trợ bởi Emmet).

# các thẻ thông dụng trong html

## Thẻ H1 và H2

Trong HTML có 6 thẻ được dùng để làm tiêu đề từ thẻ h1 - h6. Trong đó chữ h được viết tắt của từ heading (ý nghĩa: tiêu đề, phần mở đầu). Trong bài này, mình giới thiệu trước cho các bạn 2 thẻ là <h1> và <h2> vì đây là 2 thẻ được sử dụng phổ biến nhất trong số các thẻ tiêu đề.

A black screen with white text

Description automatically generated

A screenshot of a computer screen

Description automatically generated

Thẻ <h1> được sử dụng làm tiêu đề chính, ví dụ tiêu đề của một bài blog. Các thẻ <h2> được sử dụng làm tiêu đề phụ, ví dụ dùng làm các tiêu đề thể hiện ý chính trong bài blog.

Tiêu đề chính thường thấy trong: Tiêu đề của bài blog, tên sản phẩm trên trang TMĐT, tên một bài học, …

Tiêu đề phụ thường thấy trong các ý chính của một bài viết.

## Thẻ p

Thẻ <p> để thể hiện các đoạn văn, p là viết tắt của paragraph (ý nghĩa: đoạn văn bản). Thẻ <p> được sử dụng để thể hiện các đoạn văn bản trong nội dung bài viết như: mô tả bài blog, các đoạn văn bản trong bài blog, các đoạn văn bản trong nội dung sản phẩm, nội dung các bình luận trên mạng xã hội, …

A screenshot of a computer

Description automatically generated

## Thẻ b, i, u

Đây là 3 thẻ có chức năng khá quen thuộc nếu các bạn đã sử dụng các trình soạn thảo văn bản. Trong đó b viết tắt của bold (ý nghĩa: in đậm), i viết tắt của italic (ý nghĩa: in nghiêng) và u viết tắt của underline (ý nghĩa: gạch chân).

A screenshot of a computer

Description automatically generated

Trường hợp sử dụng các thẻ này trên nội dung trang web cũng tương tự như khi soạn văn bản trên trình soạn thảo. Thông thường, ta sẽ in đậm các từ hoặc đoạn văn muốn nhấn mạnh, in nghiêng cho các đoạn văn mang tính lưu ý và gạch chân cho những đoạn văn muốn nhấn mạnh hoặc đoạn văn chứa liên kết có thể nhấn vào.

A screenshot of a computer screen

Description automatically generated

## Thẻ a

Khi truy cập các trang web bạn gần như luôn bắt gặp các thẻ a, thẻ này giúp tạo ra một siêu liên kết tới các trang web, các file, email, … Trong đó, a viết tắt của anchor (ý nghĩa: điểm neo, neo một liên kết).

A screenshot of a computer

Description automatically generated

## Thẻ img

Đây là thẻ dùng để hiển thị hình ảnh trên trang web. Trong đó img viết tắt của image (ý nghĩa: hình ảnh). Trong một trang web, hình ảnh được sử dụng rất phổ biến: logo trang web, ảnh đại diện, ảnh bìa, hình ảnh các mặt hàng trên trang web TMDT, …

A screenshot of a computer

Description automatically generated

## Thẻ ul, li

Thẻ ul và li được sử dụng để tạo giao diện dạng danh sách, 2 thẻ này được sử dụng rất phổ biến trong nội dung các bài viết.

Các bạn có bao giờ thấy danh sách như này?

A screenshot of a computer

Description automatically generated

Trong đó, ul viết tắt của unordered list (ý nghĩa: danh sách không có thứ tự) và li viết tắt của list item (ý nghĩa: mục trong danh sách).

A screenshot of a computer

Description automatically generated

## Thẻ ol, li

Gần giống như cặp thẻ ul, li. Đây cũng là 2 thẻ để tạo giao diện danh sách. Điểm khác biệt duy nhất nằm ở ol, ol viết tắt của ordered list (ý nghĩa: danh sách có số thứ tự). Sử dụng thẻ này trong trường hợp bạn muốn thể hiện rõ số thứ tự của các mục trong danh sách.

A screenshot of a computer

Description automatically generated

## Tóm tắt

* Thẻ h1 sử dụng làm tiêu đề chính, thẻ h2 sử dụng làm tiêu đề phụ
* Thẻ p sử dụng để chứa các đoạn văn bản
* Thẻ b, i, u để in đậm, in ngiêng và gạch chân
* Thẻ a tạo ra các liên kết có thể nhấn vào, sử dụng thuộc tính href để chứa liên kết
* Thẻ img để hiển thị hình ảnh, liên kết tới hình ảnh được khai báo trong thuộc tính src
* Thẻ ul, li để hiển thị danh sách không có số thứ tự
* Thẻ ol, li để hiển thị danh sách có số thứ tự

# thuộc tính (attribute)

## Thuộc tính là gì?

Thuộc tính (attribute) được sử dụng để bổ sung thêm thông tin và dữ liệu cho các thẻ HTML.Các thuộc tính có dạng key="value" và luôn được viết trong thẻ mở của các thẻ HTML (không bao giờ được viết trong thẻ đóng).

Thực tế, các bạn đã được làm quen với một số thuộc tính của các thẻ HTML rồi đó.

A black and grey rectangular object

Description automatically generated

Ví dụ trên có 1 thuộc tính là lang="vi", nó gồm 2 thành phần:

lang

vi

Trong đó, lang được gọi là khóa của thuộc tính, vi được gọi là giá trị của thuộc tính.

Cùng quan sát một số ví dụ sử dụng thuộc tính thường gặp khác:

A screenshot of a computer program

Description automatically generated

## Nhiều thuộc tính

Bạn có thể sử dụng đồng thời nhiều thuộc tính trong một thẻ HTML, mỗi thuộc tính (mỗi cặp key="value") được cách nhau bởi một khoảng trắng.

A screenshot of a computer

Description automatically generated

Trong ví dụ trên có 2 thuộc tính là:

name="viewport"

content="width=device-width, initial-scale=1.0"

Trong thuộc tính thứ nhất có:

key: name

value: viewport

Trong thuộc tính thứ hai có:

key: content

value: width=device-width, initial-scale=1.0

## Thuộc tính toàn cục

Các thuộc tính được đề cập trong phần này có thể sử dụng cho toàn bộ các thẻ HTML. Vì vậy, chúng được gọi là thuộc tính toàn cục (Global Attributes).

### Thuộc tính lang

Mặc định thuộc tính lang không được sử dụng cho trang web. Khi bạn sử dụng thuộc tính này cho thẻ <html> thì mặc định các thẻ trong trang web sẽ thừa kế lại thuộc tính lang này.

A screen shot of a computer code

Description automatically generated

Chính vì điều này, nên nếu trang web của bạn có nội dung là tiếng Việt thì chỉ cần thêm <html lang="vi"> là đủ.

Tuy nhiên, có một số tình huống trang web của bạn vẫn đề cập tới một ngôn ngữ khác ngôn ngữ chính của trang web. Trường hợp này, bạn có thể sử dụng thêm thuộc tính lang cho thẻ chứa nội dung là ngôn ngữ khác.

A computer screen shot of white text

Description automatically generated

### Thuộc tính hidden

Các thẻ HTML có thuộc tính hidden sẽ được ẩn khỏi giao diện trang web (bao gồm cả các thẻ con và nội dung của nó).

A black and white picture of a person

Description automatically generated

Sau này, bạn có thể dùng JavaScript để thêm/bớt thuộc tính hidden để làm cho một thẻ HTML được ẩn/hiện trên giao diện trang web.

### Thuộc tính title

Đây là thuộc tính bổ sung thông tin cho một thẻ HTML, nội dung của thuộc tính này sẽ hiển thị khi bạn đưa con trỏ chuột vào phần nội dung của nó.

A screenshot of a computer code

Description automatically generated

A screenshot of a chat

Description automatically generated

Ngoài ra, còn một số thuộc tính toàn cục chưa được đề cập tới trong bài này. Bạn có thể tham khảo thêm tại: <https://www.w3schools.com/tags/ref_standardattributes.asp>

## Tóm tắt

* Thuộc tính (attribute) được dùng để bổ sung thêm thông tin và dữ liệu cho các thẻ HTML.
* Các thuộc tính luôn luôn được thêm vào phần thẻ mở, không bao giờ được thêm vào phần thẻ đóng.
* Có thể sử dụng nhiều thuộc tính trong một thẻ HTML. Ví dụ: <meta name="viewport" content="width=device-width, initial-scale=1.0" />
* Các thuộc tính toàn cục có thể sử dụng cho mọi thẻ HTML. Ví dụ: lang, hidden, title

# mẹo để viết HTML tốt hơn

## Viết thẻ mở và thẻ đóng trước

Tuân thủ theo cách viết này, bạn sẽ không bao giờ bị quên thẻ đóng.

## Trình bày code ngay ngắn

Một bài văn tốt cần đúng chính tả. Người viết văn là người hành văn, người viết code là người hành code. Vậy viết code HTML như thế nào cho đúng “chính tả”?

## Thẻ con thụt lề 1 dấu tab so với thẻ cha

Để đảm bảo tính dễ đọc, tránh nhầm lẫn khi số lượng code HTML nhiều lên, các bạn luôn thụt lề cho thẻ con bằng 1 dấu Tab (4 dấu cách) so với thẻ cha nhé.

A screen shot of a computer

Description automatically generated

Thực tế, việc không ngay hàng thẳng lối có rất nhiều hình thái khác nhau, tùy vào cách viết của mỗi người. Nhưng đều gây ra vấn đề chung đó là rất khó đọc và khó xác định được các thẻ mở và thẻ đóng. Có thể dẫn tới việc viết thừa/thiếu thẻ mở/thẻ đóng, gây lỗi giao diện trang web.

## Không để dư các khoảng trắng

Trong HTML, việc viết dư thừa các khoảng trắng hay các dấu ngắt dòng không gây lỗi giao diện. Tuy nhiên, chúng ta không có lý do gì để viết dư các khoảng trắng. Luôn đảm bảo viết code tối giản, đơn giản, không dư thừa các khoảng trắng, dấu ngắt dòng không cần thiết.

A screenshot of a computer program

Description automatically generated

## Ngắt dòng hợp lý

Để đảm bảo tính dễ đọc cho code của các bạn khi viết HTML, các bạn cần tìm những điểm ngắt dòng hợp lý.

A screenshot of a computer

Description automatically generated

## Đặt các thẻ đúng vị trí

Có một số trường hợp các thẻ HTML được thiết kế để luôn nằm trong một thẻ cha khác.

A screenshot of a computer program

Description automatically generated

Có 2 điểm sử dụng chưa đúng:

* Thẻ <body> đang nằm trong thẻ <head>
* Thẻ <h1> đang là con trực tiếp của thẻ <html>

A screenshot of a computer program

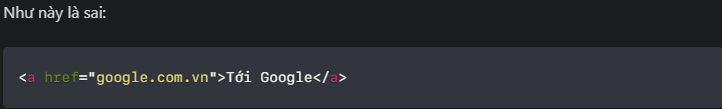
Description automatically generated

## Một số sai lầm thường gặp khác

Khi mới học, chúng ta thường gặp khó khăn trong việc học cú pháp. Nhiều khi, chỉ một lỗi cú pháp nhỏ có thể lấy đi thời gian cả ngày của bạn.

### Quên http, https cho thuộc tính href

Ví dụ, để chuyển tới trang web google.com.vn khi nhấn vào thẻ <a> bạn sẽ làm như nào?



Vì một URL cần bắt đầu với phần giao thức (http, https). Trường hợp sử dụng đúng sẽ như sau:



### Mở/đóng thẻ sai thứ tự

Ví dụ trong trường hợp sử dụng kết hợp nhiều thẻ một lúc.

A screenshot of a computer

Description automatically generated

## Tóm tắt

* Luôn viết cú pháp mở và đóng thẻ trước, sau đó mới viết nội dung
* Sử dụng thụt lề bằng 1 dấu Tab cho các thẻ con
* Không viết dư thừa các khoảng trắng và các dấu ngắt dòng
* Sử dụng dấu ngắt dòng hợp lý, đảm bảo code dễ đọc
* Đặt các thẻ đúng vị trí. Ví dụ: thẻ <li> luôn là con trực tiếp của thẻ <ul>, <ol>
* Luôn nhớ sử dụng https cho các URL trong thuộc tính href

# Các thẻ tiêu đề

Trong HTML có 6 thẻ tiêu đề từ **h1 - h6**. Con số trong tên thẻ thể hiện cấp độ (Level) của thẻ, cấp độ ở đây là việc ưu tiên thứ tự sử dụng thẻ. Trong đó cấp 1 là ưu tiên cao nhất, cấp 6 là ưu tiên thấp nhất.

Thẻ <h1> được sử dụng làm tiêu đề chính cho trang web. Đây là phần tiêu đề nổi bật nhất, dễ thấy nhất, mang thông điệp ngắn gọn giúp người dùng và công cụ tìm kiếm hiểu trang web của bạn nói về điều gì.

## Thẻ H1

### Giúp công cụ tìm kiếm hiểu nội dung trang web

[John Mueller của Google](https://www.seroundtable.com/google-h1-tags-rankings-28305.html) nói rằng thẻ <h1> giúp Google hiểu cấu trúc & nội dung của một trang web. Vì vậy, bạn nên sử dụng thẻ <h1> cho tiêu đề chính cho trang web của bạn nhé.

Tham khảo thêm: [Đề xuất sử dụng thẻ H1 từ Google](https://developers.google.com/style/headings)

### Nâng cao trải nghiệm người dùng

Trong nhiều trường hợp, thẻ <h1> là một phần trong cấu trúc phân cấp của trang web. Trong đó thẻ <h1> là tiêu đề nổi bật nhất, sau đó là: <h2>, h3, …

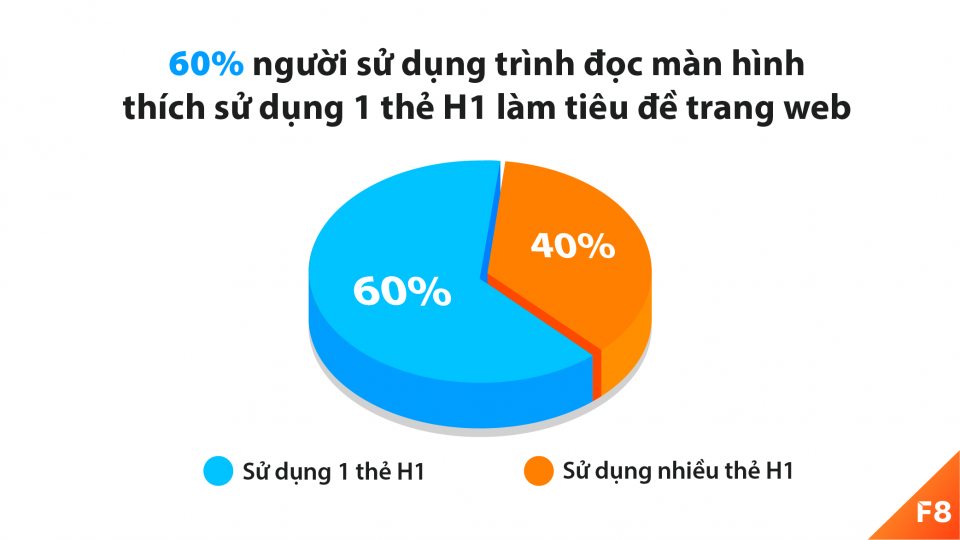


### Nâng cao khả năng tiếp cận

Khả năng tiếp cận cho trang web (Web accessibility) đề cập đến việc thiết kế các trang web đáp ứng cho mọi người, bao gồm cả người khuyết tật.

Vì vậy, ta có thể hiểu ngắn gọn “Web Accessibility” là các phương án để làm cho mọi người, bao gồm cả người khuyết tật, đều có thể dễ dàng truy cập và sử dụng trang web.

Theo [nghiên cứu này](https://webaim.org/projects/screenreadersurvey7/#heading) của webAIM, 60% người dùng trình đọc màn hình thích tiêu đề trang web là 1 thẻ <h1>.



Ở gần cuối khóa học, mình có một chương nói riêng về “Web accessibility” các bạn nhé.

### Tóm tắt

* Nội dung thẻ <h1> giúp Google hiểu về cấu trúc & nội dung trang web
* Chỉ nên sử dụng tối đa 1 thẻ <h1> trong 1 trang
* Nên sử dụng thẻ <h1> cho tất cả các trang quan trọng
* Thẻ <h1> giúp nâng cao trải nghiệm người dùng và khả năng tiếp cận (Web accessibility) cho trang web. Giúp mọi người, bao gồm cả người khuyết tật có thể dễ dàng sử dụng trang web.

## Thẻ H2

### Tìm hiểu thẻ h2

Thẻ <h2> được sử dụng làm tiêu đề các phần chính trong một trang web. Thẻ <h2> được sử dụng làm tiêu đề con của thẻ <h1>.

A black rectangular object with white text

Description automatically generated

Lưu ý, thẻ <h2> được dùng làm tiêu đề con của thẻ <h1> ở đây nghĩa là về mặt trình bày và ý nghĩa sử dụng, không phải đề cập tới việc đưa thẻ <h2> vào bên trong thẻ <h1>.

A screenshot of a computer

Description automatically generated

Chúng ta sẽ không lồng các thẻ h vào nhau khi sử dụng.

### Cấu trúc phân cấp

Sự kết hợp của các thẻ h trong một trang web giúp tạo nên hệ thống phân cấp về mặt nội dung. Giúp người sử dụng và các công cụ tìm kiếm sẽ dễ dàng xác định được bố cục nội dung của một trang web.



Tham khảo: <https://laravel.com/docs/9.x/eloquent-relationships>

Để tự xác định được cấu trúc các thẻ h, đơn giản bạn chỉ cần nghĩ như lúc đang viết một bài blog:

* Tiêu đề chính của bài blog dùng thẻ h1
* Các phần chính trong bài blog dùng thẻ h2
* Các ý phụ trong mỗi phần dùng thẻ h3
* Tương tự như vậy cho tới thẻ h6

Trong thực tế, cũng hiếm khi chúng ta phải sử dụng tới thẻ h6. Vì các nội dung thông thường có từ 1 - 4 cấp độ, vậy nên chúng ta sẽ thường thấy thẻ h1 - h4 được sử dụng. Trong đó, thẻ h1 - h2 thường xuyên được sử dụng hơn cả.

Xác định các thẻ tiêu đề bằng ý nghĩa nội dung của nó, không phải dựa trên kích thước của nó. Bởi vì kích thước còn phụ thuộc vào thiết kế và cách sử dụng CSS, CSS có thể thay đổi kích thước của các thẻ.

### Tóm tắt

* Sử dụng thẻ <h2> làm tiêu đề các phần chính trong trang web, đây được coi là tiêu đề con của tiêu đề <h1>
* Không sử dụng các thẻ h lồng nhau
* Các thẻ h được sử dụng để tạo hệ thống phân cấp nội dung
* Xác định các thẻ tiêu đề bằng ý nghĩa nội dung của nó, không phải dựa trên kích thước của nó

## Thẻ H3, H4, H5, H6

### Thẻ tiêu đề h3 - h6

Qua các bài học trước, chúng ta đã được tìm hiểu chi tiết về thẻ <h1>, <h2>. Trong bài này, chúng ta sẽ tìm hiểu về các thẻ h còn lại:

* Thẻ <h3> là tiêu đề con của thẻ <h2>
* Thẻ <h4> là tiêu đề con của thẻ <h3>
* Thẻ <h5> là tiêu đề con của thẻ <h4>
* Thẻ <h6> là tiêu đề con của thẻ <h5>

Các thẻ càng gần với h6 càng ít được sử dụng. Bạn cũng không cần phải sử dụng tất cả các thẻ h, nó tùy thuộc vào nội dung bạn muốn thể hiện.

Mục tiêu của việc sử dụng các thẻ h là để tạo ra cấu trúc phân cấp nội dung cho trang web của bạn. Giúp người dùng và các công cụ tìm kiếm có thể dễ dàng hiểu được nội dung mà trang web đang đề cập.

### Tránh bỏ qua các cấp độ

Khi sử dụng các thẻ h, bạn nên tránh bỏ qua các cấp độ tiêu đề. Nếu bạn đang sử dụng thẻ h1, thì sau đó nên là thẻ h2. Tránh bỏ qua thẻ h2 mà đi tới luôn thẻ h3. Tương tự như vậy cho tới thẻ h6.

## Các thẻ tiêu đề và SEO

Tối ưu hóa công cụ tìm kiếm (Search Engine Optimization - SEO) là hoạt động tối ưu trang web nhằm nâng cao thứ hạng trang web trong kết quả của công cụ tìm kiếm.

Các tiêu đề có vai trò quan trọng trong SEO, nó cung cấp nội dung từ khóa và cấu trúc phân cấp nội dung trang web cho các công cụ tìm kiếm. Từ đó, các công cụ tìm kiếm như Google Search có thể đánh giá, phân tích và xếp hạng cho trang web.

## Tóm tắt

* Trong HTML có 6 thẻ tiêu đề từ h1 - h6. Trong đó h1 là cấp cao nhất, h6 là cấp thấp nhất
* Các thẻ càng gần với h6 càng ít được sử dụng. Bạn cũng không cần phải sử dụng tất cả các thẻ h, nó tùy thuộc vào nội dung bạn muốn thể hiện
* Tránh bỏ qua các cấp độ tiêu đề. Nếu bạn đang sử dụng thẻ h1, thì sau đó nên là thẻ h2. Tránh bỏ qua thẻ h2 mà đi tới luôn thẻ h3
* Các tiêu đề có vai trò quan trọng trong SEO, nó cung cấp nội dung từ khóa và cấu trúc phân cấp nội dung trang web cho các công cụ tìm kiếm

## Những sai lầm thường gặp

### Sử dụng nhiều h1 trong cùng một trang

Thẻ <h1> nên được sử dụng cho tiêu đề chính cho trang web hiện tại của bạn và nên đặt ở đầu phần nội dung của bạn. Nếu bạn có một trang nói chi tiết về sản phẩm Laptop DELL XPS 9710x thì bạn nên sử dụng thẻ <h1> với tên sản phẩm đó:



Nếu sau đó bạn có đề cập tới một chiếc laptop khác là Laptop Razer 15 2022 thì bạn nên có một trang riêng cho sản phẩm đó với tên của nó nằm trong thẻ <h1>.

### Sử dụng thẻ h vì nó phù hợp với kích thước phông chữ

Lỗi này gặp chủ yếu do các nhà phát triển trang web chưa biết cách sử dụng đúng các thẻ tiêu đề. Ví dụ họ không sử dụng thẻ <h1> chỉ vì nó quá lớn so với thiết kế mà họ đang tìm kiếm, nên họ chọn sử dụng thẻ <h2> hoặc <h3>.

Tuy nhiên, bạn chỉ cần sử dụng CSS để tạo kiểu cho các tiêu đề theo cách chúng ta muốn, vì vậy kích thước phông chữ không nên được dùng để xem xét khi quyết định lựa chọn các thẻ tiêu đề.

### Không tuân thủ cấu trúc tiêu đề

Tất cả các tiêu đề nên tuân theo một cấu trúc rõ ràng. Vì vậy, một <h2> phải theo sau một <h1>, một <h3> phải theo sau một <h2>, … Bạn tránh quên điều này và hãy nhớ thêm rằng bạn có thể dễ dàng thay đổi kích thước của các thẻ tiêu đề bằng CSS.

Ví dụ, sau đây là một cấu trúc tiêu đề tốt:

<h1>Thú cưng phổ biến ở Việt Nam</h1>

<h2>Chó Husky ở Việt Nam</h2>

<h3>Chế độ ăn kiêng của chó Husky</h3>

<h3>Đặc điểm của chó Husky</h3>

<h2>Chuột Hamster ở Việt Nam</h2>

<h3>Chế độ ăn kiêng cho chuột Hamster</h3>

<h3>Đặc điểm của chuột Hamster</h3>

### Hoàn toàn không sử dụng thẻ tiêu đề

Điều này thường xảy ra trong quá khứ, các nhà phát triển trang web thường sử dụng các “tricks” để “đánh lừa” công cụ tìm kiếm. Bởi vì họ biết rằng các công cụ tìm kiếm sẽ ưu tiên cho các nội dung trong thẻ h, họ thường spam các từ khóa cần SEO trong các thẻ h trên trang web.

Ngày nay, các công cụ tìm kiếm đã rất thông minh. Các “tricks” spam từ khóa trong các thẻ h không mang lại tác dụng, thậm chí còn phản tác dụng.

Nếu bạn cần lời khuyên cho việc làm SEO tốt. Mình khuyên các bạn nên chú trọng tới việc mang lại nội dung có giá trị cho người sử dụng. Các công cụ tìm kiếm ngày nay đủ thông minh để đánh giá được trang web có thực sự tốt cho người dùng hay không, từ đó nâng cao thứ hạng của trang web trong kết quả tìm kiếm.

### Tóm tắt

* Nên sử dụng tối đa một thẻ <h1> trong cùng một trang
* Không sử dụng thẻ h chỉ vì nó phù hợp với kích thước phông chữ
* Nên tuân thủ cấu trúc các thẻ h để tạo ra cấu trúc phân cấp cho nội dung trang web
* Làm SEO tốt là mang lại giá trị nội dung đích thực cho người sử dụng. Không nên spam các từ khóa trong các thẻ tiêu đề nhắm “đánh lừa” công cụ tìm kiếm

# Thẻ đoạn văn

## Thẻ p

### Tìm hiểu về thẻ p

Các đoạn văn bản chiếm một phần lớn trong nội dung của các trang web. Trong bài này, chúng ta sẽ tìm hiểu chi tiết hơn về cách sử dụng thẻ <p> để làm việc với các đoạn văn bản.

Chữ p viết tắt của paragraph (ý nghĩa: đoạn văn). Viết nhanh thẻ này với Emmet: gõ p và nhấn tab.



Thẻ <p> được dùng để thể hiện một đoạn văn bản trên trang web. Ví dụ như một bài blog được tạo ra bởi nhiều đoạn văn bản.

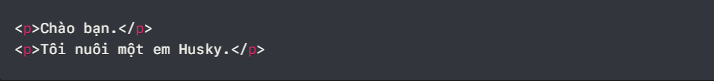
### Ví dụ về cách sử dụng thẻ p

Mỗi đoạn văn bản có thể nằm trong một thẻ <p>, một đoạn văn có thể bao gồm nhiều câu.

A screenshot of a computer

Description automatically generated

Thẻ <p> cũng có thể được sử dụng cho các đoạn văn bản ngắn, không nhất thiết phải sử dụng với các đoạn văn bản dài và nhiều câu văn.



Trong thực tế, phần lớn các bài viết được tạo nên từ nhiều đoạn văn khác nhau. Mỗi đoạn văn này sẽ nằm trong một thẻ <p> khác nhau.

Bạn sẽ nhận thấy rằng một đoạn văn luôn bắt đầu trên một dòng mới. Ngoài ra, các trình duyệt sẽ tự động thêm một số khoảng trắng phía trên và dưới để ngăn cách giữa các đoạn văn.

Bạn sẽ dễ dàng thấy điều này khi chúng ta có nhiều đoạn văn liên tiếp.

A screenshot of a computer

Description automatically generated

Khoảng cách giữa các thẻ <p> được gọi là khoảng cách lề (margin), đây là khoảng cách mặc định, chúng ta có thể sửa lại khoảng cách này khi sử dụng CSS.

### Tóm tắt

* Thẻ <p> được sử dụng để thể hiện một đoạn văn bản trên trang web.
* Một đoạn văn không nhất thiết phải nhiều hơn một câu, có thể sử dụng thẻ <p> với các đoạn văn bản ngắn.
* Mỗi thẻ <p> sẽ luôn nằm trên một dòng mới. Ngoài ra, trình duyệt sẽ tự động thêm khoảng lề trên và dưới để ngăn cách giữa các thẻ <p>.

## Thẻ br

### Giới thiệu thẻ br

Thẻ ngắt dòng (Line Break element) trong HTML là <br />, được sử dụng để tạo ra một dòng mới (ngắt dòng) trong cùng một đoạn văn. Đây là một thẻ tự đóng.

Trong thực tế, thẻ <br> rất hữu ích để tạo điểm ngắt dòng cho một bài thơ hoặc một địa chỉ.

A close up of a sign

Description automatically generated

Trong HTML, nhiều khoảng trắng hoặc dấu ngắt dòng chỉ được coi là một khoảng trắng (một dấu cách). Vì vậy, nếu bạn sử dụng dấu ngắt dòng trong code thì kết quả có thể không như ý.

Ví dụ không sử dụng <br>:

A black and white screen

Description automatically generated

Có thể bạn sẽ nghĩ, mỗi thẻ <p> luôn bắt đầu trên một dòng mới. Vậy tại sao không sử dụng mỗi dòng bằng một thẻ <p>? Chúng ta sẽ tìm hiểu điều này trong phần tiếp theo nhé.

### Cách sử dụng thẻ br

Hãy xem xét hai đoạn mã HTML qua hai ví dụ dưới đây.

Ví dụ, ngắt dòng bằng thẻ <br>:

A screenshot of a computer

Description automatically generated

Ví dụ, ngắt dòng bằng việc sử dụng thẻ <p>:

A black and white square with text

Description automatically generated

Kết quả hiển thị trên trình duyệt của hai đoạn mã trên gần như giống hệt nhau. Tuy nhiên, chúng khác nhau về ngữ nghĩa sử dụng và khả năng truy cập (Web accessibility).

Tuy hiển thị giống nhau. Nhưng theo bạn, đâu mới là cách sử dụng đúng đắn?

Câu trả lời là:

Thẻ <br> không nên được sử dụng để phân tách hai đoạn văn hoặc hai nội dung khác nhau để đảm bảo tính ngữ nghĩa. Trường hợp duy nhất bạn nên sử dụng thẻ <br> đó là để tạo dòng mới trong cùng một đoạn văn.

### Tóm tắt

* Thẻ ngắt dòng (Line Break element) trong HTML là <br />, được sử dụng để tạo ra một dòng mới trong một đoạn văn. Đây là một thẻ tự đóng.
* Trong thực tế, thẻ <br> rất hữu ích để tạo điểm ngắt dòng cho một bài thơ hoặc một địa chỉ.
* Trong HTML, nhiều khoảng trắng hoặc dấu ngắt dòng chỉ được coi là một khoảng trắng (một dấu cách).
* Thẻ <br> không nên được sử dụng để phân tách hai đoạn văn hoặc hai nội dung khác nhau, để đảm bảo tính ngữ nghĩa. Trường hợp duy nhất bạn nên sử dụng thẻ <br> đó là để tạo dòng mới trong cùng một đoạn văn.

## Tóm tắt thẻ p và br

* Thẻ <p> được sử dụng để thể hiện một đoạn văn bản trên trang web.
* Một đoạn văn không nhất thiết phải nhiều hơn một câu, có thể sử dụng thẻ <p> với các đoạn văn bản ngắn.
* Mỗi thẻ <p> sẽ luôn nằm trên một dòng mới. Ngoài ra, trình duyệt sẽ tự động thêm khoảng lề trên và dưới để ngăn cách giữa các thẻ <p>.
* Thẻ ngắt dòng (tiếng Anh: Line Break element) trong HTML là <br />, được sử dụng để tạo ra một dòng mới trong một đoạn văn. Đây là một thẻ tự đóng.
* Trong thực tế, thẻ <br> rất hữu ích để tạo điểm ngắt dòng cho một bài thơ hoặc một địa chỉ.
* Trong HTML, nhiều khoảng trắng hoặc dấu ngắt dòng chỉ được coi là một khoảng trắng (một dấu cách).
* Mỗi thẻ <p> là một đoạn văn độc lập. Không có tình huống một đoạn văn lại nằm trong một đoạn văn khác.
* Luôn sử dụng thẻ <p> cho các đoạn văn bản để đảm bảo tính ngữ nghĩa và tối ưu khả năng truy cập.
* Thẻ <br> chỉ sử dụng với mục đích ngắt dòng trong các đoạn văn, không sử dụng để tạo khoảng cách lề giữa các nội dung.
* Tính ngữ nghĩa là rất quan trọng trong HTML, bản thân mỗi thẻ HTML được tạo ra là để đánh dấu giúp nội dung có ý nghĩa hơn.

# Chữ in đậm, chữ in nghiêng

## Thẻ strong và thẻ b

**Giống nhau:**

* Thẻ <b> và <strong> giống nhau về cách hiển thị, các trình duyệt hiển thị nội dung của 2 thẻ này dưới dạng chữ in đậm.

**Khác nhau:**

* Thẻ <b> và <strong> khác nhau về ngữ nghĩa và mục đích sử dụng.
* Thẻ <b> thường được sử dụng để in đậm một đoạn văn bản, nhằm thu hút sự chú ý của người đọc vào nội dung này.
* Thẻ <strong> sử dụng khi muốn thể hiện sự quan trọng, nghiêm trọng, khẩn cấp, v.v

**Trường hợp sử dụng:**

Phần lớn thời gian chúng ta sử dụng thẻ <b> và <strong> là vì chúng ta muốn in đậm một phần nội dung. Trong thực tế, đôi khi việc lựa chọn thẻ <b> hay <strong> có thể là sự ngẫu hứng của người viết code (trong trường hợp chưa nắm được ngữ nghĩa của thẻ).

**Vậy sử dụng 2 thẻ này như thế nào cho đúng?**

Các bạn có thể tuân theo cách sau:

* Khi chỉ muốn **in đậm nội dung**: sử dụng thẻ <b> hoặc CSS, với mục đích in đậm để gây sự chú ý thông thường thì sử dụng thẻ <strong> chưa hợp lý về ngữ nghĩa.
* Khi muốn thể hiện **nội dung là quan trọng**: sử dụng thẻ <strong> là hợp lý, thẻ <b> hoặc CSS sẽ chưa hợp lý vì không có ngữ nghĩa thể hiện sự quan trọng.

**Tóm tắt:**

* Nội dung trong thẻ <b> và <strong> đều được trình duyệt hiển thị dưới dạng chữ in đậm, kết quả hiển thị là giống nhau.
* Sự khác khác giữa <b> và <strong> là về mặt ngữ nghĩa sử dụng
* Thẻ <b> sử dụng khi đơn giản là muốn gây sự chú ý vào một đoạn nội dung
* Thẻ <strong> sử dụng khi muốn thể hiện sự quan trọng, nghiêm trọng, khẩn cấp, v.v

## Thẻ em và i

**Giống nhau**

* Trình duyệt hiển thị nội dung trong 2 thẻ này giống nhau, đều là in nghiêng.

**Khác nhau**

* Thẻ <i> và <em> khác nhau về ngữ nghĩa và mục đích sử dụng.
* Thẻ i:
* Được sử dụng để thể hiện các phần nội dung trong một đoạn văn bản dưới kiểu chữ in nghiêng. Ví dụ như trong đoạn văn xuất các từ ngữ chuyên ngành kỹ thuật, tên một tác phẩm, trích dẫn một câu nói, nhắc tới các từ ở ngôn ngữ khác (bài viết tiếng Việt mà nhắc tới tiếng Pháp chẳng hạn), v.v.
* Thẻ em
* Thẻ này ở thị trường VN thường ít được sử dụng hơn thẻ <i>. Thẻ này mang ngữ nghĩa được dùng để nhấn mạnh một phần nội dung trong đoạn văn (ví dụ như nhấn trọng âm trong tiếng Anh).
* Các trình đọc màn hình cũng có thể phân biệt được nội dung trong thẻ <em> để nhấn mạnh hơn vào đó.

**Trường hợp sử dụng**

* Mặc dù thẻ <i> và thẻ <em> đều thể hiện đoạn văn dưới kiểu in nghiêng, nhưng đây lại không phải là lý do chính để chúng ta quyết định sử dụng thẻ <i> hay <em>.
* Đó là theo lý thuyết, theo đúng ngữ nghĩa của 2 thẻ này. Tuy nhiên, với ngữ nghĩa này thì thị trường Việt Nam hoặc các nước không có thị phần sử dụng trình đọc màn hình hoặc không đề cao việc nhấn trọng âm thì việc tuân theo đúng ngữ nghĩa của 2 thẻ này là không cần thiết.

**Trong thực tế, các bạn có thể sử dụng như sau:**

* Để **in nghiêng** nội dung: ưu tiên dùng thẻ <i> hoặc CSS sẽ hợp lý hơn là dùng thẻ <em>.
* Để **nhấn mạnh** từ: dùng thẻ <em>, vì dùng thẻ <i> hoặc CSS sẽ không có ý nghĩa nhấn mạnh.

**Tóm tắt:**

* Cả thẻ <i> và <em> đều được trình duyệt hiển thị nội dung dưới dạng chữ in nghiêng.
* Nếu xét về yếu tố ngữ nghĩa (phù hợp hơn với thị trường dùng trình đọc màn hình hoặc ngôn ngữ có quy ước nhấn trọng âm như tiếng Anh):
* Thẻ <i> mang ngữ nghĩa thể hiện các thuật ngữ chuyên ngành, tên một tác phẩm, trích dẫn một câu nói, nhắc tới một ngôn ngữ khác trong đoạn văn, v.v.
* Thẻ <em> mang ngữ nghĩa thể hiện việc nhấn mạnh nội dung (nhấn mạnh trọng âm, điều này hoạt động trên cả trình đọc màn hình), v.v
* Xét về thực tế thì bạn có thể sử dụng như sau:
* Để in nghiêng nội dung: ưu tiên dùng thẻ <i> hoặc CSS sẽ hợp lý hơn là dùng thẻ <em>.
* Để nhấn mạnh từ: dùng thẻ <em>, vì dùng thẻ <i> hoặc CSS sẽ không mang ý nghĩa nhấn mạnh.