**MỤC LỤC**

[CHƯƠNG 1. CẤU TRÚC HTML5. 2](#_Toc200634662)

[1.1. Giới thiệu 2](#_Toc200634663)

[1.2. Tags 2](#_Toc200634664)

[1.2.1. Cấu trúc file html 5. 3](#_Toc200634665)

[1.2.2. Thẻ Head. 3](#_Toc200634666)

[1.2.2.1 Thẻ meta 3](#_Toc200634667)

[1.2.2.2 Thẻ link 3](#_Toc200634668)

[1.2.2.3 Thẻ title 4](#_Toc200634669)

[1.2.3. Thẻ Body. 4](#_Toc200634670)

[1.3. Sematic TAGS 5](#_Toc200634671)

[1.4. Attribute 6](#_Toc200634672)

[1.4.1. Thuộc tính toàn cục 6](#_Toc200634673)

[1.4.2. Thuộc tính cụ thể của từng thẻ (Specific Attributes) 6](#_Toc200634674)

[1.5. FROMS 7](#_Toc200634675)

[1.5.1. Input types 7](#_Toc200634676)

[1.5.2. Validation 7](#_Toc200634677)

[1.5.3. Aria 7](#_Toc200634678)

[1.6. SEO CƠ BẢN 8](#_Toc200634679)

[1.6.1. Meta tags 8](#_Toc200634680)

[1.6.2. Alt text 9](#_Toc200634681)

[CHƯƠNG 2. BÀI TẬP 11](#_Toc200634682)

**MỤC LỤC HÌNH ẢNH**

[Hình 1.1 Cấu trúc khi dùng Sematic tags 6](#_Toc200634754)

[Hình 1.2 Ví dụ cho thẻ meta tags 9](#_Toc200634755)

[Hình 1.3 Ví dụ minh họa thẻ meta với thuộc tính Viewport 10](#_Toc200634756)

**MỤC LỤC BẢNG BIỂU**

[Bảng 1.1 Loại tags phổ biến 3](#_Toc200638105)

[Bảng 1.2 Các thẻ sematic. 8](#_Toc200638106)

# HTML5.

## Giới thiệu

HTML (HyperText Markup Language) hay còn gọi ngôn ngữ đánh dấu siêu văn bản.

Chú ý :

Thường file html 5 sẽ có đuôi .html nhưng vẫn có trường hợp .htm chương trình vẫn nhận ra đó là các hệ điều hành cũ (như Windows 3.1, DOS), tên file chỉ được dùng tối đa **3 ký tự** cho phần mở rộng.

## Tags

Trong HTML, tag (thẻ) là một cặp ký hiệu đặc biệt dùng để xác định các phần tử (elements) của trang web — ví dụ như đoạn văn, hình ảnh, tiêu đề, liên kết, biểu mẫu, v.v.  
Có 3 kiểu thẻ phổ biến.

|  |  |  |
| --- | --- | --- |
| **Loại thẻ** | **Ví dụ** | **Ghi chú** |
| Thẻ mở + đóng | <p>...</p> | Có cả mở và đóng |
| Thẻ tự đóng | <img /> | Không cần đóng |
| Thẻ rỗng | <br> | Thẻ rỗng vì ko chứa nội dung |

Bảng 1.1 Loại tags phổ biến

* Lưu ý: html5 không phân biệt chữ hoa hay chữ thường.
* Các thẻ có thể lồng nhau để tạo cấu trúc dom.

### Cấu trúc file html 5.

Một file html thường có cấu trúc 2 phần :

<!DOCTYPE html> : Khai báo cho trình duyệt biết đây là tài liệu HTML5.

Html:lly g Đây là thẻ gốc (root element) của toàn bộ trang HTML. Tất cả các nội dung khác của trang web đều nằm trong thẻ <html> này.

* Head: Chứa các thông tin metadata (siêu dữ liệu) về trang web, không hiển thị trực tiếp trên trình duyệt nhưng lại rất quan trọng.
* Body: Chứa tất cả nội dung thực sự hiển thị trên trang web mà người dùng có thể nhìn thấy và tương tác.

### Thẻ Head.

#### Thẻ meta

T**hẻ <meta>** trong HTML là thành phần quan trọng trong phần <head> của tài liệu, dùng để cung cấp **siêu dữ liệu (metadata)** cho trình duyệt.

* **Metadata** (dịch sang tiếng Việt là **"siêu dữ liệu"**) là **dữ liệu mô tả thông tin về dữ liệu khác**. Nói đơn giản, nó không phải là dữ liệu chính, mà là những thông tin **mô tả, giải thích, hoặc cung cấp ngữ cảnh** cho dữ liệu chính.

Thẻ <meta> là một thẻ tự đóng (self-closing tag).

#### Thẻ link

**Thẻ <link>** dùng để **liên kết (kết nối)** tài liệu HTML hiện tại với các **tài nguyên bên ngoài** như: file CSS, icon, font, hoặc các file khác.

Không giống như thẻ <a> dùng để tạo siêu liên kết (hyperlink) trong nội dung trang web. thẻ <link> chủ yếu được đặt trong phần <head> của tài liệu và không hiển thị trực tiếp trên trang

Các thuộc tính bắt buộc của thẻ link

* rel : Dùng xác định **mối quan hệ** giữa tài liệu HTML hiện tại và tài nguyên được liên kết.
* stylesheet : Xác định mối liên kết với file css.
* Icon: Chỉ định biểu tượng trang web (favicon) hiển thị trên tab trình duyệt, thanh địa chỉ, hoặc bookmark.
* Preload: Hướng dẫn trình duyệt **tải trước một tài nguyên cụ thể** càng sớm càng tốt mà không làm chặn quá trình render của trang. Rất hữu ích để tối ưu hiệu suất tải trang ⇨ ví dụ như font, file css
* Preconnect: Cho trình duyệt biết rằng tài liệu hiện tại dự định **thiết lập kết nối đến một tên miền (origin)** mà nó cần các tài nguyên từ đó. Điều này giúp giảm độ trễ (latency) khi tải tài nguyên từ nguồn đó sau này. ⇨ ví dụ như serve.
* href : Xác định đường dẫn (URL) của tài nguyên được liên kết.

#### Thẻ title

Thẻ <title> là một phần tử HTML được đặt bên trong phần **<head>** của tài liệu. Nó định nghĩa **tiêu đề của trang web**.

### Thẻ Body.

Thẻ <body> là một trong những thẻ cơ bản và quan trọng nhất trong cấu trúc HTML. Nó đại diện cho **toàn bộ nội dung hiển thị** của một tài liệu HTML. Nói cách khác, **mọi thứ nhìn thấy và tương tác trên một trang web** (văn bản, hình ảnh, video, liên kết, nút bấm, biểu mẫu, v.v.) đều nằm bên trong thẻ <body>.

## Sematic TAGS

### Giới thiệu về các thẻ ngữ nghĩa

Khi thiết kế một trang web, **chúng ta không chỉ làm cho mắt người thấy đẹp**, mà còn phải để **máy hiểu được nội dung và cấu trúc**.

Trước HTML5, các nhà phát triển thường sử dụng các thẻ <div> và <span> kết hợp với các thuộc tính id hoặc class để định nghĩa cấu trúc và ý nghĩa của các phần tử trên trang.

Mặc dù vẫn hoạt động, cách này không cung cấp thông tin ngữ nghĩa thực sự cho trình duyệt hoặc công cụ tìm kiếm.

A screenshot of a computer

AI-generated content may be incorrect.

Hình 1.1 Tại sao phải dùng Sematic

* **"Semantic tags"** là **các thẻ HTML5 có ý nghĩa rõ ràng về mặt nội dung** — giúp trình duyệt, công cụ tìm kiếm, và lập trình viên **hiểu rõ chức năng của từng phần trên trang.**

A grid of a website

AI-generated content may be incorrect.

Hình 1.2 Cấu trúc khi dùng Sematic tags

### Ý nghĩa của các thẻ sematic.

|  |  |  |
| --- | --- | --- |
| **Thẻ Semantic** | **Mô tả** | **Vai trò & Mục đích sử dụng điển hình** |
| <header> | Đại diện cho phần đầu của một tài liệu, section, hoặc article. | Thường chứa logo, tiêu đề trang (<h1>), thanh điều hướng (<nav>), hoặc các yếu tố giới thiệu khác. Giúp công cụ tìm kiếm và trình đọc màn hình nhận diện phần mở đầu. |
| <nav> | Đại diện cho một tập hợp các liên kết điều hướng chính. | Chứa các menu chính, mục lục, hoặc liên kết quan trọng giúp người dùng di chuyển trong trang/website. Quan trọng cho khả năng tiếp cận (screen readers). |
| <main> | Đại diện cho nội dung chính, độc lập và duy nhất của tài liệu. | Là "trái tim" của trang, chứa nội dung cốt lõi mà không bao gồm các phần tử lặp lại như sidebar, header, footer. Mỗi tài liệu chỉ nên có một thẻ <main>. Tốt cho SEO và khả năng tiếp cận. |
| <article> | Đại diện cho một phần nội dung độc lập, khép kín, có thể phân phối lại. | Dùng cho bài blog, tin tức, bình luận, bài đăng diễn đàn. Mỗi <article> nên có tiêu đề riêng (<h1>-<h6>). Giúp công cụ tìm kiếm hiểu rõ các mục nội dung riêng biệt trên trang. |
| <section> | Đại diện cho một phần chung chung của tài liệu, thường có tiêu đề. | Dùng để nhóm các nội dung có liên quan theo chủ đề (ví dụ: "Giới thiệu", "Sản phẩm", "Liên hệ"). Phù hợp khi một <div> không đủ để truyền tải ý nghĩa ngữ nghĩa. Không dùng chỉ để định kiểu. |
| <aside> | Đại diện cho nội dung liên quan nhưng có thể độc lập với nội dung chính. | Thường là thanh bên (sidebar), khối quảng cáo, liên kết liên quan, thông tin thêm không phải là phần cốt lõi của nội dung chính. |
| <footer> | Đại diện cho phần cuối của một tài liệu, section, hoặc article. | Thường chứa thông tin bản quyền, liên hệ, liên kết bổ sung, thông tin tác giả. Giúp định vị phần cuối của một khối nội dung. |
| <figure> | Đại diện cho nội dung tự chứa (ảnh, biểu đồ, video, code). | Dùng để nhóm nội dung đa phương tiện và chú thích của nó. Nội dung trong <figure> có thể được di chuyển mà không ảnh hưởng đến ngữ cảnh chính của tài liệu. |
| <figcaption> | Cung cấp chú thích hoặc mô tả cho nội dung trong thẻ <figure>. | Luôn nằm trong thẻ <figure>, cung cấp văn bản mô tả cho hình ảnh, biểu đồ, v.v. Cải thiện khả năng tiếp cận và giúp hiểu ngữ cảnh của nội dung đa phương tiện. |
| <time> | Đại diện cho một thời điểm cụ thể hoặc một khoảng thời gian. | Cung cấp ý nghĩa ngữ nghĩa cho ngày tháng/thời gian, có thể kèm thuộc tính datetime để máy móc dễ dàng đọc và hiểu. Ví dụ: <time datetime="2025-06-11">Ngày 11 tháng 6 năm 2025</time>. |
| <mark> | Đánh dấu văn bản có liên quan hoặc nổi bật trong ngữ cảnh cụ thể. | Giúp người đọc dễ dàng nhận ra các phần quan trọng của văn bản, thường được trình duyệt hiển thị mặc định bằng màu vàng. Ví dụ: highlight kết quả tìm kiếm. |
| <details> | Tạo một widget tiết lộ (disclosure widget). | Cho phép người dùng mở rộng hoặc thu gọn nội dung. Hữu ích cho các phần FAQ, thông tin chi tiết mà không muốn hiển thị ngay lập tức. |
| <summary> | Cung cấp tiêu đề hoặc chú thích cho widget tiết lộ trong <details>. | Luôn nằm trong thẻ <details>, là phần văn bản mà người dùng nhấp vào để mở rộng/thu gọn nội dung của <details>. |

Bảng 1.2 Các thẻ sematic.

### Lưu ý

* Mục đích chính của thẻ ngữ nghĩa là cung cấp ý nghĩa cấu trúc, không phải để áp dụng CSS.
* Một trang chỉ nên có **một <main>**
* Luôn có tiêu đề (<h1>-<h6>) cho <section> và <article>
* Phân biệt rõ <article> và <section>:

<article>: Dành cho nội dung có thể tự đứng một mình, độc lập, và có thể được phân phối lại (ví dụ: bài blog, tin tức). Nếu bạn có thể lấy nội dung ra và đặt vào một trang khác mà nó vẫn có ý nghĩa đầy đủ, đó là một <article>.

<section>: Dành cho các phần có liên quan về mặt chủ đề trong một tài liệu hoặc trong một <article>. Một <section> thường có một tiêu đề để xác định chủ đề của nó.

* Sử dụng đúng cấu trúc lồng ghép

A screenshot of a chat

AI-generated content may be incorrect.

Hình 1.3 Lưu ý khi sử dụng sematic

Chú ý: Với non-Sematic

**Khi nào dùng <div>?**

* Dùng <div> khi không có thẻ ngữ nghĩa nào phù hợp hơn. <div> là một phần tử nhóm chung chung.
* Thường dùng <div> để tạo các container cho mục đích định kiểu (ví dụ: bao quanh một nhóm các phần tử để áp dụng flexbox/grid) hoặc để nhóm các phần tử mà không có ý nghĩa ngữ nghĩa cụ thể nào khác.

## Attribute

Khi viết một **thẻ HTML**, ví dụ như <img>, <a>, <input>, các thẻ này không thể hiện đầy đủ ý nghĩa nếu **không có thông tin đi kèm**.

🧐 **Vậy làm sao để nói cho trình duyệt biết:**

* Ảnh đó nằm ở đâu?
* Liên kết đó dẫn tới đâu?
* Ô nhập liệu có loại là gì?

📌 Đó chính là lý do HTML có **"attribute" – thuộc tính** để **cung cấp thông tin bổ sung** cho phần tử.

* Vậy **thuộc tính** để cung cấp thêm thông tin về thẻ đó hoặc để điều chỉnh hành vi, giao diện của nó. Các thuộc tính này được định nghĩa trong **thẻ mở** **hoặc thẻ tự đóng** của một phần tử HTML.

Lưu ý:

* Luôn sử dụng dấu nháy kép cho giá trị thuộc tính (""):

Lý do: Mặc dù một số trình duyệt có thể chấp nhận dấu nháy đơn ('') hoặc thậm chí không có dấu nháy cho các giá trị đơn giản, nhưng việc sử dụng dấu nháy kép là tiêu chuẩn khuyến nghị của W3C. Điều này giúp tránh lỗi khi giá trị chứa dấu cách hoặc các ký tự đặc biệt, và cải thiện khả năng đọc mã.

### Thuộc tính toàn cục

Là những thuộc tính có thể được sử dụng trên **hầu hết mọi thẻ HTML**. Chúng cung cấp các chức năng chung cho các phần tử.

### Thuộc tính cụ thể của từng thẻ (Specific Attributes)

## FROMS

### Input types

### Validation

### Aria

## SEO CƠ BẢN

### Meta tags

Thẻ <meta> giúp **SEO, responsive, chia sẻ mạng xã hội, xác minh thông tin...**

A screenshot of a web page

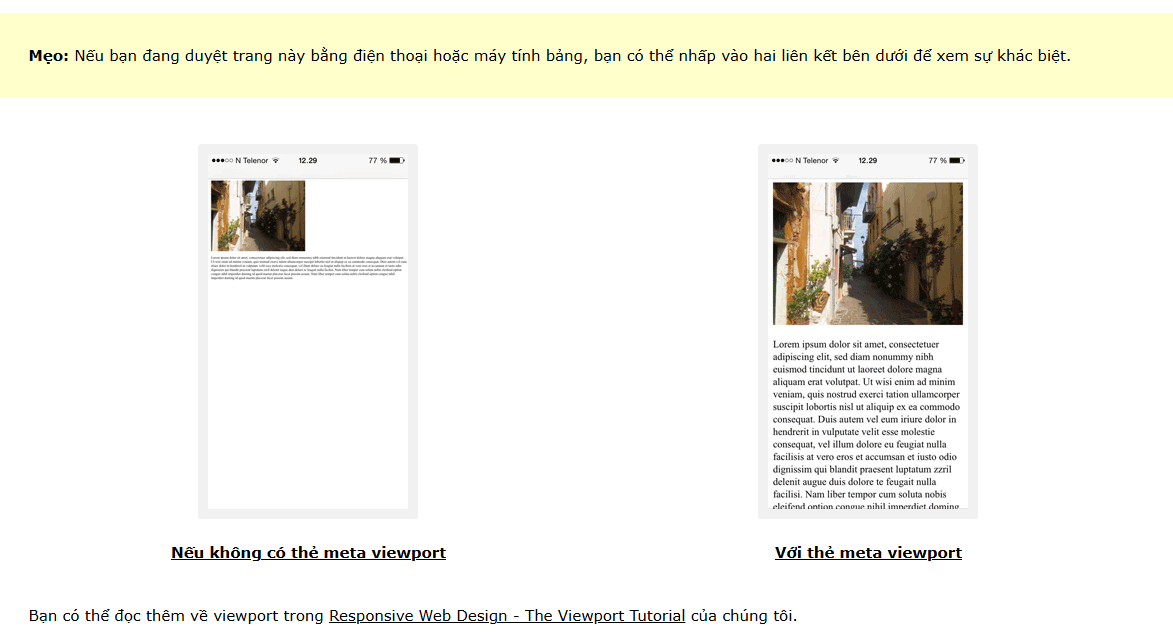
AI-generated content may be incorrect.

Hình 1.4 Ví dụ cho thẻ meta tags

Nhưng thành phần của một thẻ meta.

* Charset : Xác định bộ mã ký tự cho tài liệu HTML. Thường được dùng UTF-8
* Name … content :
* Viewport : Dùng để hiển thị trên thiết bị di động có hai thuộc tính nhỏ “width=device-width”: Chiều rộng của viewport (vùng hiển thị) sẽ bằng chiều rộng của thiết bị (ví dụ: màn hình điện thoại).

initial-scale=1.0: Mức độ zoom ban đầu khi trang được tải là 100% (không phóng to hay thu nhỏ).



Hình 1.5 Ví dụ minh họa thẻ meta với thuộc tính Viewport

* Description : Cung cấp một đoạn tóm tắt ngắn gọn về nội dung của trang.
* Author : Cung cấp tên tác giả hoặc chủ sở hữu của trang web/tài liệu
* Keywords : Liệt kê các từ khóa liên quan đến nội dung của trang

Lưu ý:

Không bắt buộc phải dùng hết, nhưng **charset và viewport là tối thiểu nên có.**

### Alt text

**Alt text** (viết tắt của "alternative text") là một đoạn văn bản mô tả ngắn gọn nội dung của một hình ảnh. Đoạn văn bản này được đặt trong thuộc tính alt của thẻ <img>.

* **Tối ưu hóa công cụ tìm kiếm (SEO):** Các công cụ tìm kiếm (như Google) không thể "nhìn" được hình ảnh. Chúng dựa vào alt text để hiểu nội dung và ngữ cảnh của hình ảnh đó.
* Bằng cách cung cấp alt text mô tả và có liên quan đến từ khóa, bạn giúp công cụ tìm kiếm lập chỉ mục hình ảnh của mình hiệu quả hơn, từ đó cải thiện thứ hạng tìm kiếm cho cả hình ảnh và trang web của bạn.

**Cách viết Alt text hiệu quả:**

* Mô tả chính xác và ngắn gọn: Tập trung vào những gì quan trọng nhất trong hình ảnh.
* **Tránh "Hình ảnh của..." hoặc "Ảnh..."**: Trình đọc màn hình đã tự động nhận diện đây là một hình ảnh, nên việc lặp lại "Hình ảnh của..." là không cần thiết.
* **Độ dài phù hợp:** Cố gắng giữ alt text dưới 125 ký tự để đảm bảo các trình đọc màn hình không cắt bớt.

# BÀI TẬP

Bài Tập: Xây Dựng Trang "Hồ Sơ Cá Nhân Tối Giản" Với HTML5

Mục tiêu: Tạo một trang web hồ sơ cá nhân đơn giản nhưng có cấu trúc rõ ràng và sử dụng các thẻ ngữ nghĩa (semantic tags) của HTML5, cùng với một số tính năng đa phương tiện và biểu mẫu cơ bản.

Yêu cầu:

Cấu trúc chung của trang:

* Sử dụng thẻ <!DOCTYPE html> để khai báo HTML5.
* Phần head phải có charset và title.
* Phần body sẽ chứa toàn bộ nội dung hiển thị.

Phần đầu trang (Header):

* Sử dụng thẻ <header>.
* Bên trong header, có một tiêu đề chính giới thiệu về bạn (ví dụ: <h1>Tên Của Bạn - Hồ Sơ Cá Nhân</h1>).
* Có một đoạn văn giới thiệu ngắn gọn (sử dụng <p>).

Thanh điều hướng (Navigation):

* Sử dụng thẻ <nav>.
* Bên trong nav, tạo một danh sách không có thứ tự (<ul>) với các mục điều hướng (liên kết) như "Về tôi", "Kỹ năng", "Dự án", "Liên hệ".
* Sử dụng thẻ <a> cho các liên kết và trỏ đến các ID tương ứng trong trang (ví dụ: <a href="#ve-toi">Về tôi</a>).

Các phần nội dung chính (Sections):

* Phần "Về tôi":
* Sử dụng thẻ <section> với id="ve-toi".
* Có một tiêu đề cấp 2 (<h2>Về tôi</h2>).
* Một đoạn văn giới thiệu chi tiết hơn về bản thân, sở thích, mục tiêu.
* Sử dụng thẻ <blockquote> để trích dẫn một câu nói yêu thích của bạn.
* Phần "Kỹ năng":
* Sử dụng thẻ <section> với id="ky-nang".
* Có một tiêu đề cấp 2 (<h2>Kỹ năng</h2>).
* Liệt kê các kỹ năng của bạn dưới dạng danh sách không có thứ tự (<ul>) hoặc có thứ tự (<ol>).
* Ví dụ: "HTML5", "CSS3", "JavaScript", "Python", v.v.
* Phần "Dự án" (Sử dụng <article>):
* Sử dụng thẻ <section> với id="du-an".
* Có một tiêu đề cấp 2 (<h2>Dự án</h2>).
* Bên trong phần này, tạo ít nhất 2 <article> riêng biệt, mỗi <article> đại diện cho một dự án.
* Mỗi <article> phải có:
* Tiêu đề dự án (<h3>).
* Mô tả ngắn gọn về dự án (<p>).
* Một liên kết đến dự án (nếu có) hoặc hình ảnh đại diện (sử dụng <a> và <img>).
* Phần "Liên hệ":
* Sử dụng thẻ <section> với id="lien-he".
* Có một tiêu đề cấp 2 (<h2>Liên hệ</h2>).
* Tạo một biểu mẫu liên hệ đơn giản (<form>) với các trường sau:
* Tên (<label> và <input type="text">).
* Email (<label> và <input type="email">).
* Tin nhắn (<label> và <textarea>).
* Một nút gửi (<input type="submit">).
* Thêm thuộc tính HTML5 cho input: placeholder, required cho Email.

Phần chân trang (Footer):

* Sử dụng thẻ <footer>.
* Bên trong footer, có thông tin bản quyền hoặc liên hệ cơ bản (ví dụ: &copy; 2025 Tên Của Bạn).
* Sử dụng <small> để hiển thị văn bản bản quyền nhỏ hơn.
* Sử dụng đa phương tiện (Media):
* Video: Trong phần "Về tôi" hoặc một phần mới, nhúng một video ngắn (có thể là video mẫu) sử dụng thẻ <video>. Đặt thuộc tính controls để hiển thị điều khiển phát lại.
* Audio: Trong phần "Kỹ năng" hoặc một phần mới, nhúng một đoạn âm thanh ngắn (có thể là âm thanh mẫu) sử dụng thẻ <audio>. Đặt thuộc tính controls.