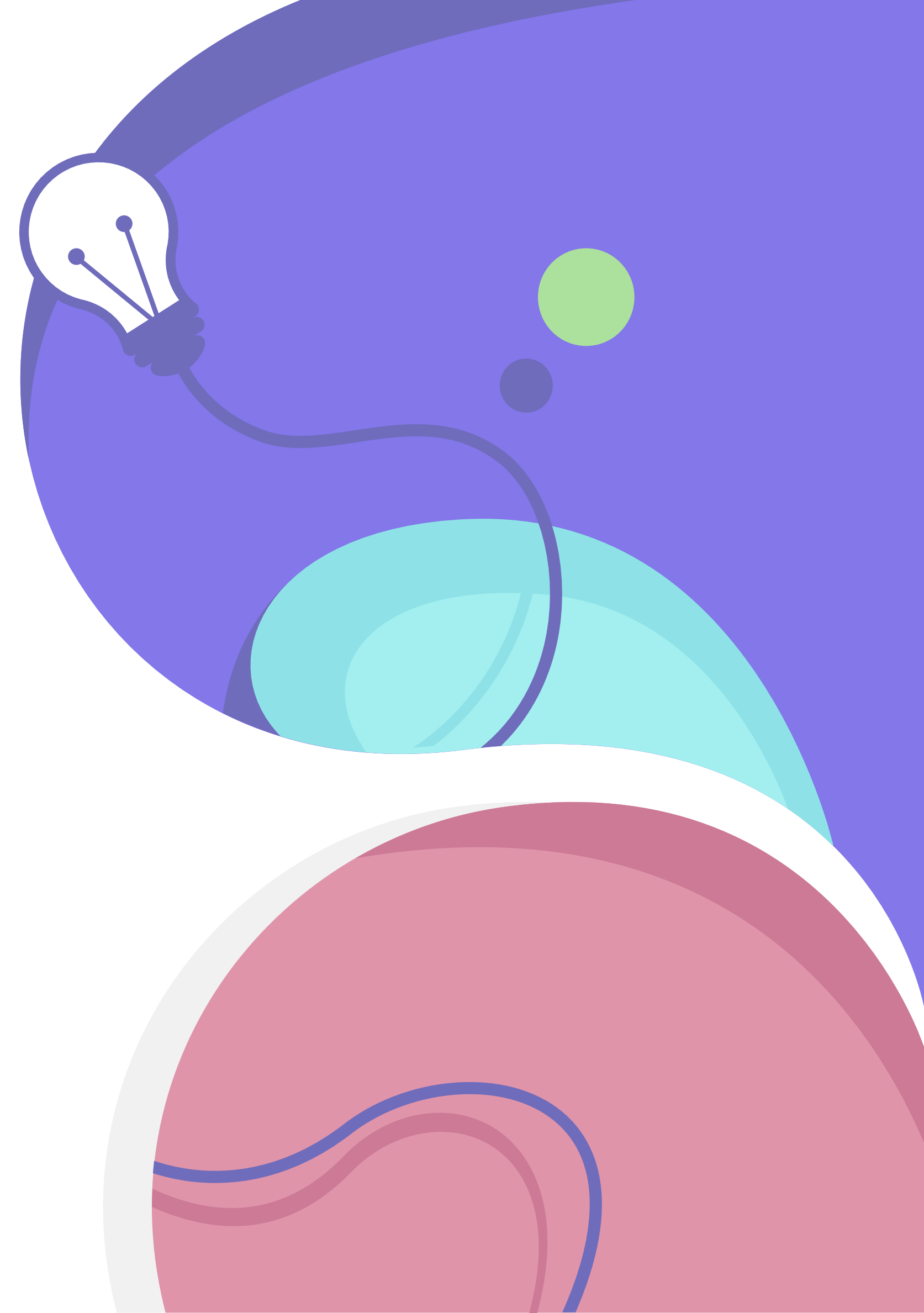


HTML CSS

Ali Duc Huy





THẺ <audio> TRONG HTML



Thẻ **<audio>** dùng để chèn (đưa) một tệp âm thanh vào trang web.
Trình duyệt sẽ hiển thị bộ điều khiển âm thanh để người dùng có thể phát, tạm dừng, chỉnh âm lượng,...

1. CÚ PHÁP CƠ BẢN

```
<audio controls>
  <source src="music.mp3" type="audio/mpeg">
</audio>
```

- Giải thích:**
- <audio>**: Thẻ chứa âm thanh.
 - controls**: Thuộc tính hiển thị bộ điều khiển (play, pause, volume,...).
 - <source>**: Thẻ khai báo nguồn âm thanh.
 - src**: Đường dẫn đến tệp âm thanh.
 - type**: Định dạng của tệp âm thanh (audio/mpeg là MP3).

2. VÍ DỤ MINH HỌA

Ví dụ 1: Phát tệp MP3

```
<audio controls>
  <source src="music.mp3" type="audio/mpeg">
</audio>
```



Ví dụ 2: Hỗ trợ nhiều định dạng

```
<audio controls>
  <source src="music.mp3" type="audio/mpeg">
  <source src="music.ogg" type="audio/ogg">
  <source src="music.wav" type="audio/wav">
</audio>
```

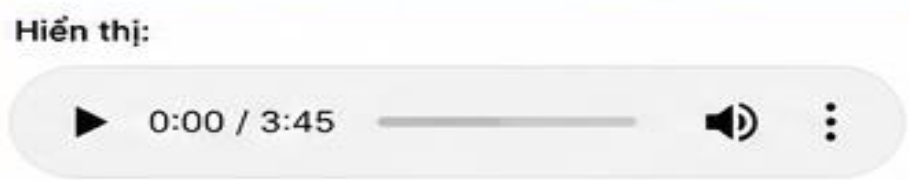


3. CÁC THUỘC TÍNH THƯỜNG DÙNG

Thuộc tính	Mô tả	Giá trị
controls	Hiển thị bộ điều khiển âm thanh	controls
autoplay	Tự động phát khi trang được tải	autoplay
loop	Tự động phát lại khi kết thúc	loop
muted	Tắt âm thanh khi phát	muted

4. VÍ DỤ KẾT HỢP CÁC THUỘC TÍNH

```
<audio controls autoplay loop muted>
  <source src="music.mp3" type="audio/mpeg">
</audio>
```



5. ĐỊNH DẠNG ÂM THANH HỖ TRỢ PHỔ BIẾN

Định dạng	Ghi chú
MP3	Phổ biến nhất, tương thích hầu hết trình duyệt
OGG	Mã nguồn mở, chất lượng tốt
WAV	Chất lượng cao, dung lượng lớn
AAC	Chất lượng cao, thường dùng cho iOS/Safari

6. LƯU Ý

- ✓ Luôn cung cấp nhiều định dạng âm thanh để đảm bảo tương thích.
- ✓ Sử dụng thuộc tính controls để người dùng dễ điều khiển.
- ✓ Không nên tự động phát âm thanh (tránh làm phiền người dùng).
- ✓ Dung lượng tệp âm thanh nên hợp lý để trang web tải nhanh.

7. <source> LÀ GÌ?

<source> là thẻ dùng để khai báo nguồn (file) âm thanh cho thẻ **<audio>**.

src: đường dẫn đến file âm thanh.

type: định dạng của file (giúp trình duyệt biết đây là loại âm thanh gì).

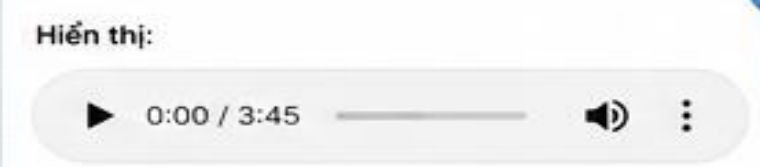
Có bắt buộc phải dùng <source> không?

KHÔNG bắt buộc.

Bạn có thể chèn âm thanh trực tiếp bằng thuộc tính **src** của thẻ **<audio>**. Tuy nhiên, khi muốn hỗ trợ nhiều định dạng, bạn nên dùng **<source>**.

Cách 1: Không dùng <source> (Đơn giản)

```
<audio controls src="music.mp3"></audio>
```



- Ưu điểm:**
- ✓ Mã ngắn gọn, dễ viết.
 - ✓ Phù hợp khi chỉ có 1 định dạng âm thanh.
 - ✓ Được tất cả trình duyệt hiện đại hỗ trợ.

Cách 2: Có dùng <source> (Khuyến khích)

```
<audio controls>
  <source src="music.mp3" type="audio/mpeg">
  <source src="music.ogg" type="audio/ogg">
  <source src="music.wav" type="audio/wav">
</audio>
```



- Ưu điểm:**
- ✓ Hỗ trợ nhiều định dạng âm thanh.
 - ✓ Tăng khả năng tương thích với trình duyệt cũ.
 - ✓ Là cách viết chuyên nghiệp.

GỢI Ý CHO HỌC SINH

Khi mới học HTML, bạn có thể dùng cách 1 (đơn giản). Sau khi hiểu rõ hơn, hãy chuyển sang cách 2 để viết code tốt hơn và chuyên nghiệp hơn.

Ví dụ đơn giản:

```
<audio controls src="music.mp3"></audio>
```

Ví dụ chuyên nghiệp:

```
<audio controls>
  <source src="music.mp3" type="audio/mpeg">
  <source src="music.ogg" type="audio/ogg">
</audio>
```



THẺ `<video>` TRONG HTML



Thẻ `<video>` dùng để nhúng (chèn) một đoạn video vào trang web.

Trình duyệt sẽ hiển thị bộ điều khiển video để người dùng có thể phát, tạm dừng, tua, điều chỉnh âm lượng, toàn màn hình,...

1. CÚ PHÁP CƠ BẢN

`<video controls>`

`<source src="video.mp4" type="video/mp4">`

Trình duyệt của bạn không hỗ trợ thẻ video.

`</video>`

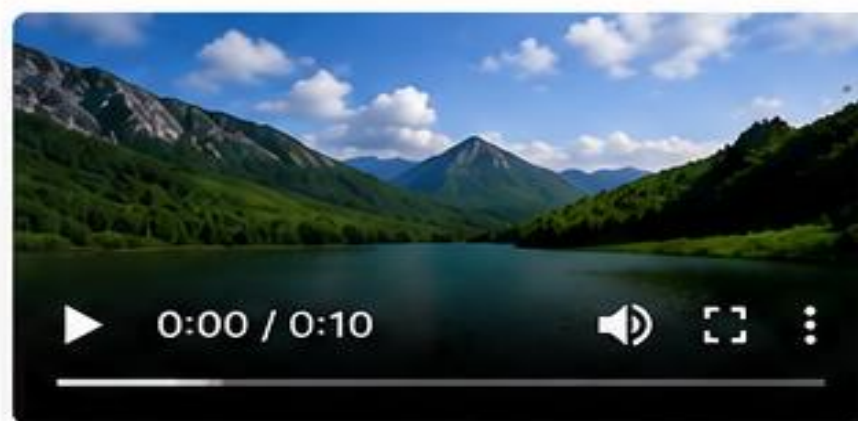
Giải thích:

- `<video>`**: Thẻ chứa video.
- `controls`**: Hiển thị bộ điều khiển (play, pause, volume, ...).
- `<source>`**: Chỉ định nguồn video và định dạng.
- Dòng chữ bên trong `<video>` sẽ hiển thị nếu trình duyệt không hỗ trợ.

2. VÍ DỤ MINH HỌA

Ví dụ 1: Video MP4

```
<video controls width="480">
  <source src="video.mp4" type="video/mp4">
</video>
```



Ví dụ 2: Hỗ trợ nhiều định dạng

```
<video controls width="480">
  <source src="video.mp4" type="video/mp4">
  <source src="video.webm" type="video/webm">
  <source src="video.ogv" type="video/ogg">
</video>
```



Ví dụ 3: Tự động phát (không khuyến khích)

```
<video controls autoplay muted loop width="480">
  <source src="video.mp4" type="video/mp4">
</video>
```



3. CÁC THUỘC TÍNH THƯỜNG DÙNG

Thuộc tính	Mô tả	Giá trị
controls	Hiển thị bộ điều khiển video	controls
autoplay	Tự động phát khi trang được tải	autoplay
loop	Tự động phát lại khi kết thúc	loop
muted	Tắt âm thanh khi phát	muted
width	Chiều rộng video	pixel (vd: 480) hoặc %
height	Chiều cao video	pixel (vd: 270) hoặc %
poster	Ảnh hiển thị trước khi phát	URL ảnh

4. VÍ DỤ SỬ DỤNG POSTER

```
<video controls width="480" poster="thumbnail.jpg">
  <source src="video.mp4" type="video/mp4">
</video>
```



5. GIÁ TRỊ THUỘC TÍNH PRELOAD (ĐÃ BỎ)

- i** Thuộc tính preload đã bị loại bỏ trong HTML5. Trình duyệt hiện nay tự quyết định việc tải video dựa trên kết nối mạng và thiết bị của người dùng.

6. KHÔNG DÙNG `<SOURCE>` (CHỈ 1 ĐỊNH DẠNG)

Nếu chỉ có 1 định dạng video (ví dụ MP4), bạn có thể dùng trực tiếp thuộc tính src:

```
<video controls width="480" src="video.mp4">
  Trình duyệt của bạn không hỗ trợ thẻ video.
</video>
```



7. DÒNG CHỮ DỰ PHÒNG (FALLBACK TEXT)

Nếu trình duyệt không hỗ trợ thẻ `<video>` thì dòng chữ bên trong sẽ hiển thị.

```
<video controls width="480">
  <source src="video.mp4" type="video/mp4">
  Trình duyệt của bạn không hỗ trợ thẻ video.
</video>
```

8. LƯU Ý

- ✓ Luôn cung cấp nhiều định dạng (MP4, WebM, OGG) để tăng khả năng tương thích.
- ✓ Không nên tự động phát video có âm thanh.
- ✓ Sử dụng **poster** để hiển thị hình ảnh hấp dẫn trước khi phát.
- ✓ Dùng **controls** để người dùng dễ dàng điều khiển video.



THẺ `<iframe>` TRONG HTML



Thẻ `<iframe>` (inline frame) dùng để nhúng một trang web khác vào trong trang hiện tại.
Nội dung từ trang được nhúng sẽ hiển thị như một khung nằm trong trang của bạn.

1. CÚ PHÁP CƠ BẢN

```
<iframe src="URL" width="rộng" height="cao"></iframe>
```

Trong đó:

- src** : Đường dẫn đến trang cần nhúng.
- width** : Chiều rộng của khung iframe.
- height** : Chiều cao của khung iframe.

3. CÁC THUỘC TÍNH THƯỜNG DÙNG

Thuộc tính	Mô tả	Giá trị
src	Đường dẫn đến trang cần nhúng	URL
width	Chiều rộng của iframe	px hoặc %
height	Chiều cao của iframe	px
title	Tiêu đề mô tả nội dung iframe	Văn bản
frameborder	Hiển thị hoặc ẩn viền khung	0 (ẩn) hoặc 1 (hiện)
scrolling	Hiển thị thanh cuộn	yes, no, auto
allowfullscreen	Cho phép hiển thị toàn màn hình	(không có giá trị)

5. LƯU Ý QUAN TRỌNG

- ⚠ Một số trang web có thể chặn iframe để bảo vệ nội dung. Nếu trang bị chặn, bạn sẽ không thể nhúng được.
- ⚠ Ảnh hưởng đến tốc độ tải trang nếu nhúng quá nhiều.
- ⚠ Nên sử dụng iframe hợp lý để đảm bảo trải nghiệm tốt cho người dùng và SEO.

2. VÍ DỤ MINH HỌA

Ví dụ 1: Nhúng một trang web

```
<iframe src="https://www.w3schools.com" width="600" height="300"></iframe>
```



Ví dụ 2: Nhúng Google Maps

```
<iframe src="https://www.google.com/maps" width="600" height="300"></iframe>
```



Ví dụ 3: Nhúng video YouTube

```
<iframe width="560" height="315" src="https://www.youtube.com/embed/dQw4w9WgXcQ" title="YouTube video" frameborder="0" allowfullscreen></iframe>
```



4. LÀM SAO BIẾT TRANG WEB CÓ HỖ TRỢ IFRAME HAY KHÔNG?

Bạn có thể kiểm tra theo các cách sau:

1 Kiểm tra bằng cách nhúng thử

Thử nhúng trang web vào iframe.

- Nếu nội dung hiển thị bình thường → Trang web đó **cho phép iframe**.
- Nếu chỉ hiện thông báo lỗi hoặc trang trống → Trang web đó **chặn iframe**.

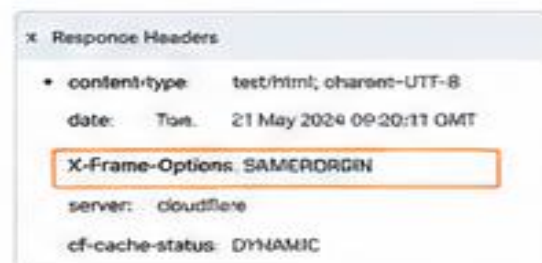


2 Kiểm tra bằng trình duyệt (X-Frame-Options)

Mở trang web cần kiểm tra → Nhấn F12 → Chọn tab "Network" → Tải lại trang (F5) → Chọn một file bất kỳ → Xem mục "Headers".

Tìm dòng X-Frame-Options:

SAMEORIGIN	: Chỉ cho phép nhúng trong cùng domain.
DENY	: Không cho phép nhúng.
Không có	: Thường là cho phép nhúng.



3 Dùng công cụ online

Truy cập các trang như:

- <https://iframetester.net/>
- <https://iframe.ly/embed>

Nhập URL trang web, công cụ sẽ kiểm tra giúp bạn.

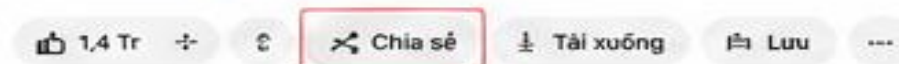


6. LÀM SAO LẤY LINK IFRAME TRÊN YOUTUBE?

1 Mở video YouTube bạn muốn nhúng.



2 Bấm nút "Chia sẻ" (Share) bên dưới video.



3 Chọn "Nhúng" (Embed).



4 Sao chép đoạn mã iframe được cung cấp.

```
<iframe width="560" height="315" src="https://www.youtube.com/embed/dQw4w9WgXcQ" title="YouTube video" frameborder="0" allowfullscreen></iframe>
```



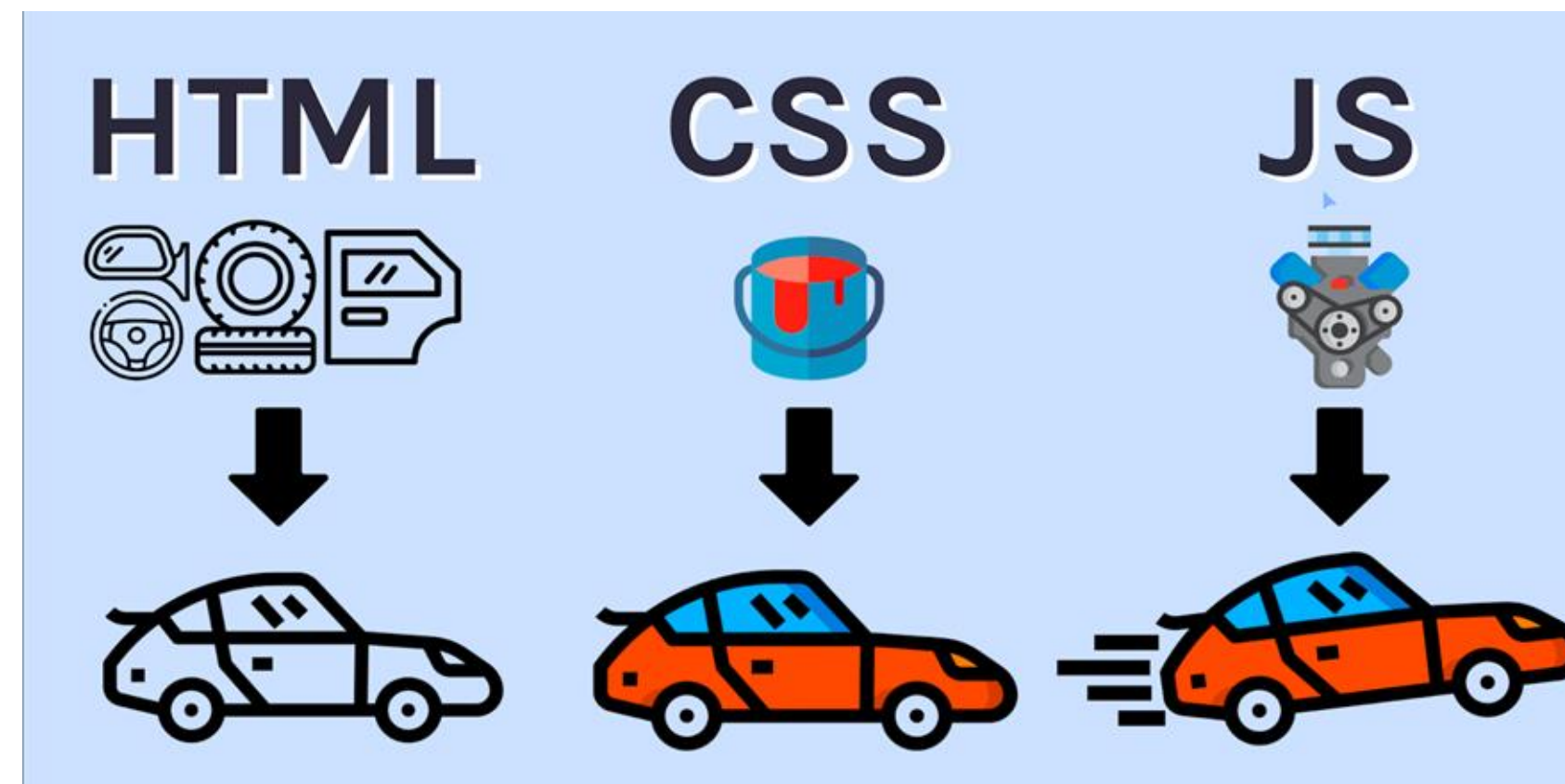
5 Dán đoạn mã này vào trang HTML của bạn.



Tổng quan CSS: **Khái niệm**

CSS là viết tắt của **Cascading Style Sheets**, một ngôn ngữ để định dạng / định kiểu / style cho tài liệu HTML.

Dễ hiểu thì CSS dùng để **làm đẹp trang web**, sử dụng nó để thay đổi **màu sắc, bố cục, phong chữ, kích thước...vv** của mọi thứ trên trang web.





Tổng quan CSS: **Cú pháp**

Cú pháp chung - tổng quan:

```
selector {  
  property: value;  
}
```

Các selector cơ bản:

Theo thẻ html: `p {...}`

Theo class: `.class-name {...}`

Theo id: `#id {...}`



Tổng quan CSS: Cách áp dụng

Có ba cách áp dụng CSS vào HTML đơn giản nhất:

1. Inline: Viết CSS trong các thẻ HTML.

- Cách này nhanh nhất, độ ưu tiên áp dụng của css cũng cao nhất. Nhưng chỉ nên dùng khi cần test nhanh và demo đơn giản. **Lạm dụng sẽ gây khó bảo trì khi scale lớn.**

1. Internal (CSS Nội tuyến) : Viết CSS trong thẻ `<style>` bên trong thẻ `<head>`

- Là một cách làm nhanh CSS ngay trong file html nếu trang web không có gì phức tạp như dự án nhỏ hoặc bài tập đơn giản mà chỉ có 1 file HTML duy nhất.

1. External (CSS Bên Ngoài): Dùng tách biệt file css riêng và gọi nó ở file html.

- Cách làm chuẩn nhất, đặc biệt trong teamwork, dễ tái sử dụng, mở rộng và bảo trì.



CÁC LOẠI CSS



1. CSS NỘI TUYẾN (INLINE CSS)

CSS được viết trực tiếp trong thẻ HTML bằng thuộc tính **style**.

CÁCH VIẾT

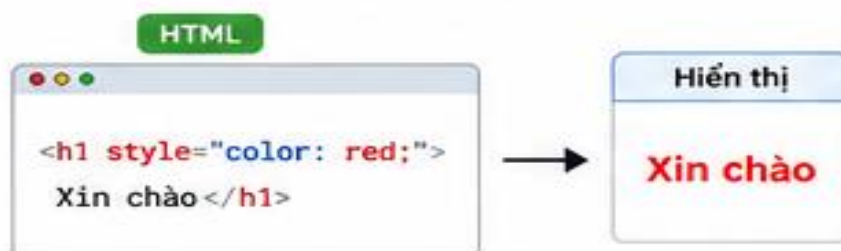
```
<h1 style="color: red;
font-size: 24px;
text-align: center;">
Xin chào các bạn!
</h1>
```

KẾT QUẢ

Xin chào các bạn!

ĐẶC ĐIỂM

- ✓ Áp dụng cho duy nhất một thẻ HTML.
- ✓ Ưu tiên cao nhất.
- ✓ Phù hợp khi cần chỉnh nhanh một thẻ.
- ✓ Không khuyến khích sử dụng nhiều.



2. CSS NỘI BỘ (INTERNAL / EMBEDDED CSS)

CSS được viết trong thẻ **<style>** bên trong phần **<head>** của trang HTML.

CÁCH VIẾT

```
<head>
<style>
  h1 {
    color: blue;
    font-size: 24px;
    text-align: center;
  }
</style>
</head>
<body>
  <h1>Xin chào các bạn!</h1>
</body>
```

KẾT QUẢ

Xin chào các bạn!

ĐẶC ĐIỂM

- ✓ Áp dụng cho toàn bộ trang HTML.
- ✓ Không ảnh hưởng đến các trang khác.
- ✓ Phù hợp cho trang đơn hoặc nội dung nhỏ.
- ✓ Dễ quản lý hơn Inline CSS.



3. CSS BÊN NGOÀI (EXTERNAL CSS) CÁCH 1: DÙNG <link> (KHUYẾN KHÍCH)

CSS được viết trong một tệp tin **.css** riêng, rồi liên kết vào trang HTML bằng thẻ **<link>**.

CÁCH VIẾT

```
index.html
<head>
  <link rel="stylesheet"
        href="style.css">
</head>
<body>
  <h1>Xin chào các bạn!</h1>
</body>

style.css
h1 {
  color: green;
  font-size: 24px;
  text-align: center;
}
```

KẾT QUẢ

Xin chào các bạn!

ĐẶC ĐIỂM

- ✓ Áp dụng cho nhiều trang HTML.
- ✓ Dễ bảo trì và quản lý.
- ✓ Giảm dung lượng HTML.
- ✓ Được khuyến khích sử dụng.



4. CSS BÊN NGOÀI (EXTERNAL CSS) CÁCH 2: DÙNG @import

CSS được liên kết từ file CSS khác bằng cú pháp **@import**.

CÁCH VIẾT

```
index.html
<head>
  <style>
    @import url("style.css");
  </style>
</head>
<body>
  <h1>Xin chào các bạn!</h1>
</body>

style.css
h1 {
  color: purple;
  font-size: 24px;
  text-align: center;
}
```

KẾT QUẢ

Xin chào các bạn!

ĐẶC ĐIỂM

- ✓ Cũng áp dụng cho nhiều trang HTML.
- ✓ Có thể import nhiều file CSS trong cùng một file CSS.
- ✓ Tốc độ tải chậm hơn cách dùng <link>.
- ✓ Ít được khuyến khích hơn <link>.



SO SÁNH NHANH

1. Inline CSS
2. Internal CSS
3. External CSS - <link>
4. External CSS - @import

CÁCH DÙNG

Một thẻ HTML
Toàn bộ trang HTML
Nhiều trang HTML
Nhiều trang HTML

PHẠM VI ÁP DỤNG

Không
Không
Có
Có (nhiều file)

TÁI SỬ DỤNG

Không
Không
Không
Không

ẢNH HƯỞNG ĐẾN TRANG KHÁC

Không
Không
Không
Không

TỐC ĐỘ TẢI

Nhanh
Nhanh
Nhanh nhất
Chậm hơn

ƯU TIÊN

Cao nhất
Thứ 2
Thứ 3
Thứ 4



THỨ TỰ ƯU TIÊN CSS: 1. Inline CSS > 2. Internal CSS > 3. External CSS (<link>) > 4. External CSS (@import)



MẪU CSS SAU THIẾT LẬP ĐỊNH DẠNG CHỮ ĐỎ CHO ĐỒNG THỜI CÁC THẺ **h1, h2, h3**

```
h1, h2, h3 { color: red; }
```



1. Ý NGHĨA

CSS ở bên sẽ áp dụng thuộc tính

color: red;

cho tất cả các thẻ **<h1>**, **<h2>**, **<h3>**.

Các thẻ được viết cách nhau bởi dấu phẩy (,).

2. CẤU TRÚC HTML

```
<body>
```

```
<h1>Tiêu đề cấp 1</h1>
```

```
<h2>Tiêu đề cấp 2</h2>
```

```
<h3>Tiêu đề cấp 3</h3>
```

```
<p>Đây là đoạn văn bản  
thường.</p>
```

```
</body>
```

3. CSS

```
1 h1, h2, h3 {  
2   color: red;  
3 }  
4 /* Áp dụng màu đỏ cho */  
5 /* tất cả các thẻ h1, h2, h3 */  
6 /* cùng một lúc */
```

4. KẾT QUẢ HIỂN THỊ TRÊN TRÌNH DUYỆT

Tiêu đề cấp 1

Tiêu đề cấp 2

Tiêu đề cấp 3

Đây là đoạn văn bản thường.



Kết quả: Các thẻ **h1, h2, h3 đều có màu chữ là **đỏ**.**

5. LƯU Ý

- Dấu phẩy (,) trong CSS dùng để chọn nhiều phân tử cùng lúc.
- Các thẻ nào được chọn sẽ chịu chung một kiểu định dạng.
- Có thể áp dụng nhiều thuộc tính khác, không chỉ color.
- Nếu một thẻ xuất hiện nhiều lần, tất cả đều bị ảnh hưởng.

6. MỞ RỘNG

Ví dụ chọn nhiều thẻ khác:

- **h1, h2, h3, p** { color: blue; } → **h1, h2, h3** và **p** đều màu xanh
- **div, span** { font-size: 18px; } → **div** và **span** có cỡ chữ 18px
- **h1, .title, #heading** { color: green; } → **h1**, **class="title"** và **id="heading"** màu xanh lá



ĐỊNH DẠNG VĂN BẢN BẰNG CSS

Sử dụng CSS để định dạng, trình bày văn bản đẹp hơn và dễ đọc hơn.

1. VÍ DỤ HTML

```
<div class="text-demo">
  <h1>Tiêu đề cấp 1</h1>
  <p class="p1">Đoạn văn bản căn giữa.</p>
  <p class="p2">Đoạn văn bản gạch chân.</p>
  <p class="p3">Đoạn văn bản viết hoa.</p>
  <p class="p4">Đoạn văn bản nghiêng.</p>
  <p class="p5">Đoạn văn bản cỡ chữ 20px.</p>
  <p class="p6">Đoạn văn bản có khoảng cách chữ.</p>
  <p class="p7">Đoạn văn bản có khoảng cách dòng.</p>
  <p class="p8">Đoạn văn bản thụt đầu dòng.</p>
</div>
```

2. CSS

```
h1 { color: blue; text-align: center; }
.p1 { text-align: center; }
.p2 { text-decoration: underline; }
.p3 { text-transform: uppercase; }
.p4 { font-style: italic; }
.p5 { font-size: 20px; }
.p6 { letter-spacing: 4px; }
.p7 { line-height: 1.8; }
.p8 { text-indent: 40px; }
```

3. KẾT QUẢ HIỂN THỊ

Tiêu đề cấp 1

Đoạn văn bản căn giữa.

Đoạn văn bản gạch chân.

ĐOẠN VĂN BẢN VIẾT HOA.

Đoạn văn bản nghiêng.

Đoạn văn bản cỡ chữ 20px.

Đoạn văn bản có khoảng cách chữ.

Đoạn văn bản có khoảng cách dòng.

Đoạn văn bản thụt đầu dòng.

	THUỘC TÍNH	MÔ TẢ	GIÁ TRỊ PHỔ BIẾN	VÍ DỤ CSS	KẾT QUẢ
≡	text-align	Căn chỉnh vị trí văn bản (theo chiều ngang)	left, right, center, justify	text-align: center;	Đoạn văn bản căn giữa.
U	text-decoration	Trang trí văn bản	none, underline, overline, line-through	text-decoration: underline;	<u>Đoạn văn bản gạch chân.</u>
Tt	text-transform	Chuyển đổi kiểu chữ	none, uppercase, lowercase, capitalize	text-transform: uppercase;	ĐOẠN VĂN BẢN VIẾT HOA.
I	font-style	Kiểu chữ	normal, italic, oblique	font-style: italic;	<i>Đoạn văn bản nghiêng.</i>
A	font-size	Kích thước chữ	px, em, rem, %, pt, ...	font-size: 20px;	Đoạn văn bản cỡ chữ 20px.
A+A	letter-spacing	Khoảng cách giữa các chữ (theo chiều ngang)	normal, số px, em, rem, ...	letter-spacing: 4px;	Đoạn văn bản có khoảng cách chữ.
↑A	line-height	Khoảng cách giữa các dòng	normal, số, số px, em, rem, ...	line-height: 1.8;	Đoạn văn bản có khoảng cách dòng.
¶	text-indent	Thụt đầu dòng	số px, em, rem, %	text-indent: 40px;	Đoạn văn bản thụt đầu dòng.
abc	text-shadow	Thêm bóng cho chữ	none, h-shadow v-shadow blur color	text-shadow: 2px 2px 4px gba(0,0,0,0.3);	Đoạn văn bản có bóng chữ.



GHI CHÚ: Các thuộc tính trên thường áp dụng cho: h1, h2, h3, h4, h5, h6, p, span, div, a, li, ...

CSS – THUỘC TÍNH color

Dùng để đặt **màu chữ** cho văn bản.

1. CÚ PHÁP

```
selector {  
    color: giá_trị;  
}
```

2. CÁC CÁCH KHAI BÁO MÀU

Cách khai báo	Ví dụ	Mô tả
Tên màu	red	Sử dụng tên màu có sẵn trong CSS.
Mã HEX	#ff0000	Sử dụng mã màu HEX (bắt đầu bằng #).
RGB	rgb(255, 0, 0)	Sử dụng giá trị RGB (đỏ, xanh lá, xanh dương).
RGBA	rgba(255, 0, 0, 0.5)	RGB có thêm độ trong suốt (0 đến 1).

3. VÍ DỤ MINH HỌA

Giá trị	CSS	Kết quả hiển thị	Màu hiển thị
red	p { color: red; }	Đây là đoạn văn bản minh họa.	 Đỏ
#00ff00	p { color: #00ff00; }	Đây là đoạn văn bản minh họa.	 Xanh lá
rgb(0, 0, 255)	p { color: rgb(0, 0, 255); }	Đây là đoạn văn bản minh họa.	 Xanh dương
rgba(255, 0, 0, 0.5)	p { color: rgba(255, 0, 0, 0.5); }	Đây là đoạn văn bản minh họa.	 Đỏ (50% trong suốt)
#ff9900	p { color: #ff9900; }	Đây là đoạn văn bản minh họa.	 Cam
purple	p { color: purple; }	Đây là đoạn văn bản minh họa.	 Tím

 **GHI CHÚ:** Thuộc tính **color** chỉ áp dụng cho **màu chữ** (văn bản và các phần tử dạng inline).

CSS – THUỘC TÍNH **text-align**

Thuộc tính text-align dùng để căn chỉnh vị trí văn bản theo **chiều ngang**.

1. CÚ PHÁP

```
selector {  
    text-align: giá_trị;  
}
```

2. CÁC GIÁ TRỊ

Giá trị	Ý nghĩa
left	Căn trái văn bản.
right	Căn phải văn bản.
center	Căn giữa văn bản.
justify	Căn đều hai bên (kéo giãn các khoảng cách giữa các từ).

3. VÍ DỤ MINH HỌA

Giá trị	HTML	CSS	Kết quả hiển thị
left	<p>Đây là đoạn văn bản minh họa. Đoạn văn bản này dùng để minh họa thuộc tính text-align.</p>	<pre>p { text-align: left; }</pre>	Đây là đoạn văn bản minh họa. Đoạn văn bản này dùng để minh họa thuộc tính text-align.
right	<p>Đây là đoạn văn bản minh họa. Đoạn văn bản này dùng để minh họa thuộc tính text-align.</p>	<pre>p { text-align: right; }</pre>	Đây là đoạn văn bản minh họa. Đoạn văn bản này dùng để minh họa thuộc tính text-align.
center	<p>Đây là đoạn văn bản minh họa. Đoạn văn bản này dùng để minh họa thuộc tính text-align.</p>	<pre>p { text-align: center; }</pre>	Đây là đoạn văn bản minh họa. Đoạn văn bản này dùng để minh họa thuộc tính text-align.
justify	<p>Đây là đoạn văn bản minh họa. Đoạn văn bản này dùng để minh họa thuộc tính text-align.</p>	<pre>p { text-align: justify; }</pre>	Đây là đoạn văn bản minh họa. Đoạn văn bản này dùng để minh họa thuộc tính text-align.



GHI CHÚ: Thuộc tính text-align thường áp dụng cho các phần tử dạng khối (block): **p**, **div**, **h1**, **h2**, **h3**, ...

CSS – THUỘC TÍNH **font-size (px)**

Dùng để xác định kích thước chữ. Ở đây chỉ minh họa với **đơn vị px**.

1. CÚ PHÁP

```
selector {  
    font-size: kích_thước_px;  
}
```

2. Ý NGHĨA

Giá trị là số kèm đơn vị px.

Ví dụ: **12px, 16px, 20px, 24px, ...**

3. VÍ DỤ MINH HỌA

Giá trị (px)	HTML	CSS	Kết quả hiển thị
12px	<p> Đây là đoạn văn bản minh họa. </p>	<pre>p { font-size: 12px; }</pre>	Đây là đoạn văn bản minh họa.
16px	<p> Đây là đoạn văn bản minh họa. </p>	<pre>p { font-size: 16px; }</pre>	Đây là đoạn văn bản minh họa.
20px	<p> Đây là đoạn văn bản minh họa. </p>	<pre>p { font-size: 20px; }</pre>	Đây là đoạn văn bản minh họa.
24px	<p> Đây là đoạn văn bản minh họa. </p>	<pre>p { font-size: 24px; }</pre>	Đây là đoạn văn bản minh họa.
32px	<p> Đây là đoạn văn bản minh họa. </p>	<pre>p { font-size: 32px; }</pre>	Đây là đoạn văn bản minh họa.



GHI CHÚ:

Đơn vị px (pixel) là đơn vị tuyệt đối, cố định kích thước chữ trên màn hình.

CSS text-shadow

Tạo bóng cho chữ (đổ bóng văn bản)

1. KHÁI NIỆM

text-shadow dùng để tạo bóng cho chữ.

2. CÚ PHÁP

`text-shadow: h-offset v-offset blur-radius color;`

- h-offset** : Khoảng cách bóng theo chiều ngang
(+) sang phải, (-) sang trái
- v-offset** : Khoảng cách bóng theo chiều dọc
(+) xuống dưới, (-) lên trên
- blur-radius**: Độ mờ/nhòe của bóng (càng lớn càng mờ)
0 = bóng sắc nét
- color** : Màu của bóng

3. VÍ DỤ CƠ BẢN

CSS	KẾT QUẢ
<code>text-shadow: 2px 2px 0 red;</code>	Xin chào!
<code>text-shadow: -2px -2px 0 blue;</code>	Xin chào!
<code>text-shadow: 2px -2px 0 green;</code>	Xin chào!
<code>text-shadow: -2px 2px 0 orange;</code>	Xin chào!

4. VÍ DỤ VỚI ĐỘ MỜ (BLUR-RADIUS)

CSS	KẾT QUẢ
<code>text-shadow: 2px 2px 5px gray;</code>	Xin chào!
<code>text-shadow: 2px 2px 10px blue;</code>	Xin chào!
<code>text-shadow: 2px 2px 20px red;</code>	Xin chào!
<code>text-shadow: 0 0 10px purple;</code>	Xin chào!

5. VÍ DỤ NHIỀU BÓNG (MULTIPLE SHADOWS)

Có thể tạo nhiều bóng cùng lúc, các bóng cách nhau bởi dấu phẩy.

```
text-shadow: 1px 1px 0 red,  
            -1px -1px 0 blue,  
            1px -1px 0 green,  
            -1px 1px 0 orange;
```

Xin chào!

6. VÍ DỤ THỰC TẾ

Bóng nhẹ Hello <code>text-shadow: 1px 1px 2px #ccc;</code>	Bóng đậm Hello <code>text-shadow: 2px 2px 4px #666;</code>	Bóng xa Hello <code>text-shadow: 5px 5px 15px #999;</code>
Bóng màu Hello <code>text-shadow: 2px 2px 5px hotpink;</code>	Bóng phát sáng Hello <code>text-shadow: 0 0 8px #00f;</code>	Bóng vàng Hello <code>text-shadow: 2px 2px 4px #ffc107;</code>

7. LƯU Ý

- ✓ Giá trị blur-radius không bắt buộc.
- ✓ Có thể viết: `text-shadow: 2px 2px red;`
(mặc định `blur-radius = 0`)
- ✓ Màu có thể viết theo tên, mã HEX, RGB, RGBA...
- ✓ Có thể dùng nhiều bóng để tạo hiệu ứng đẹp.
- ✓ Bóng chữ khác với bóng hộp (`box-shadow`).

8. TỔNG KẾT CÚ PHÁP

`text-shadow: h-offset v-offset blur-radius color;`
 Ngang Dọc Độ mờ Màu bóng

Ví dụ đầy đủ:

```
text-shadow: 3px 3px 10px rgba(0,0,0,0.5);
```

Xin chào!

CSS – THUỘC TÍNH **text-decoration**



Dùng để trang trí văn bản: **gạch chân**, **gạch ngang**, **gạch trên** hoặc không trang trí.

1. CÚ PHÁP

```
selector {  
    text-decoration: giá_trị;  
}
```

2. CÁC GIÁ TRỊ

Giá trị	Ý nghĩa
none	Không trang trí gì.
underline	Gạch chân dưới văn bản.
overline	Gạch trên đầu văn bản.
line-through	Gạch ngang giữa văn bản.

3. VÍ DỤ MINH HỌA

Giá trị	HTML	CSS	Kết quả hiển thị
none	<code><p>Đây là đoạn văn bản minh họa.</p></code>	<pre>p { text-decoration: none; }</pre>	Đây là đoạn văn bản minh họa.
underline	<code><p>Đây là đoạn văn bản minh họa.</p></code>	<pre>p { text-decoration: underline; }</pre>	<u>Đây là đoạn văn bản minh họa.</u>
overline	<code><p>Đây là đoạn văn bản minh họa.</p></code>	<pre>p { text-decoration: overline; }</pre>	<u>Đây là đoạn văn bản minh họa.</u>
line-through	<code><p>Đây là đoạn văn bản minh họa.</p></code>	<pre>p { text-decoration: line-through; }</pre>	Đây là đoạn văn bản minh họa.



GHI CHÚ: Thuộc tính text-decoration thường áp dụng cho các phần tử dạng khối (**p**, **h1**, **h2**, **div**, ...) và cả dạng nội tuyến (**span**, **a**, ...).

CSS – THUỘC TÍNH **line-height**

Dùng để điều chỉnh khoảng cách giữa các dòng trong văn bản.

CÚ PHÁP

```
selector {  
  line-height: giá_trị;  
}
```

GIÁ TRỊ

- **number**: Số không đơn vị (ví dụ: 1.5, 2, 2.5)
- **length**: Có đơn vị (ví dụ: 20px, 30px, ...)
- **%**: Phần trăm (ví dụ: 150% = 1.5)



Giá trị mặc định: **normal** (khoảng 1.2)

VÍ DỤ MINH HỌA

line-height: 1	line-height: 1.5	line-height: 2	line-height: 2.5	line-height: 3
Đây là đoạn văn bản minh họa gồm có nhiều dòng.	Đây là đoạn văn bản minh họa gồm có nhiều dòng.	Đây là đoạn văn bản minh họa gồm có nhiều dòng.	Đây là đoạn văn bản minh họa gồm có nhiều dòng.	Đây là đoạn văn bản minh họa gồm có nhiều dòng.
Khoảng cách nhỏ	Khoảng cách vừa	Khoảng cách rộng	Khoảng cách rất rộng	Khoảng cách cực rộng

CSS – THUỘC TÍNH **letter-spacing**

Dùng để điều chỉnh khoảng cách giữa các ký tự trong văn bản.

CÚ PHÁP

```
selector {  
  letter-spacing: giá_trị;  
}
```

GIÁ TRỊ

- **normal**: Mặc định (khoảng cách bình thường)
- **length**: Có đơn vị (ví dụ: 1px, 2px, 3px, ...)



Giá trị dương: tăng khoảng cách giữa các ký tự.

Giá trị âm: giảm khoảng cách giữa các ký tự.

VÍ DỤ MINH HỌA

letter-spacing: normal	letter-spacing: 1px	letter-spacing: 2px	letter-spacing: 4px	letter-spacing: -1px	letter-spacing: -2px
Đây là đoạn văn bản.	Đây là đoạn văn bản.	Đây là đoạn văn bản.	Đ â y l à đ o a n v ă n b ả n .	Đâylàđoạnvăn bản.	Đâylàđoạnvăn bản.
Khoảng cách bình thường	Tăng nhẹ khoảng cách	Tăng vừa khoảng cách	Tăng nhiều khoảng cách	Giảm nhẹ khoảng cách	Giảm nhiều khoảng cách



GHI CHÚ: Hai thuộc tính line-height (khoảng cách dòng) và letter-spacing (khoảng cách ký tự) giúp văn bản dễ đọc và trình bày đẹp hơn.

CSS – THUỘC TÍNH font-family

Dùng để chỉ định **kiểu chữ (font)** cho văn bản.

1. CÚ PHÁP

```
selector {  
  font-family: tên_font, tên_font_dự_phòng, ...;  
}
```

2. LƯU Ý

- Nếu máy tính không có tên font đầu tiên, trình duyệt sẽ dùng tên font dự phòng tiếp theo.
- Nên kết thúc bằng generic-family (serif, sans-serif, monospace, cursive, fantasy) để đảm bảo hiển thị.
- Nếu tên font có khoảng trắng, cần đặt trong dấu ngoặc kép.

3. VÍ DỤ MINH HỌA

Giá trị font-family	CSS	Kết quả hiển thị
Arial, Helvetica, sans-serif	<pre>p { font-family: Arial, Helvetica, sans-serif; }</pre>	Đây là đoạn văn bản minh họa. (Hiển thị bằng font Arial hoặc Helvetica)
"Times New Roman", Times, serif	<pre>p { font-family: "Times New Roman", Times, serif; }</pre>	Đây là đoạn văn bản minh họa. (Hiển thị bằng Times New Roman hoặc Times)
"Poppins", Arial, sans-serif	<pre>p { font-family: "Poppins", Arial, sans-serif; }</pre>	Đây là đoạn văn bản minh họa. (Hiển thị bằng Poppins hoặc Arial)
Courier New, monospace	<pre>p { font-family: Courier New, monospace; }</pre>	Đây là đoạn văn bản minh họa. (Hiển thị bằng Courier New)
cursive	<pre>p { font-family: cursive; }</pre>	<i>Đây là đoạn văn bản minh họa.</i> (Hiển thị bằng font kiểu cursive bất kỳ)
fantasy	<pre>p { font-family: fantasy; }</pre>	Đây là đoạn văn bản minh họa. (Hiển thị bằng font kiểu fantasy bất kỳ)

 **GHI CHÚ:** Nên liệt kê nhiều font dự phòng để đảm bảo văn bản hiển thị đúng trên nhiều thiết bị khác nhau.

CSS – THUỘC TÍNH font-family

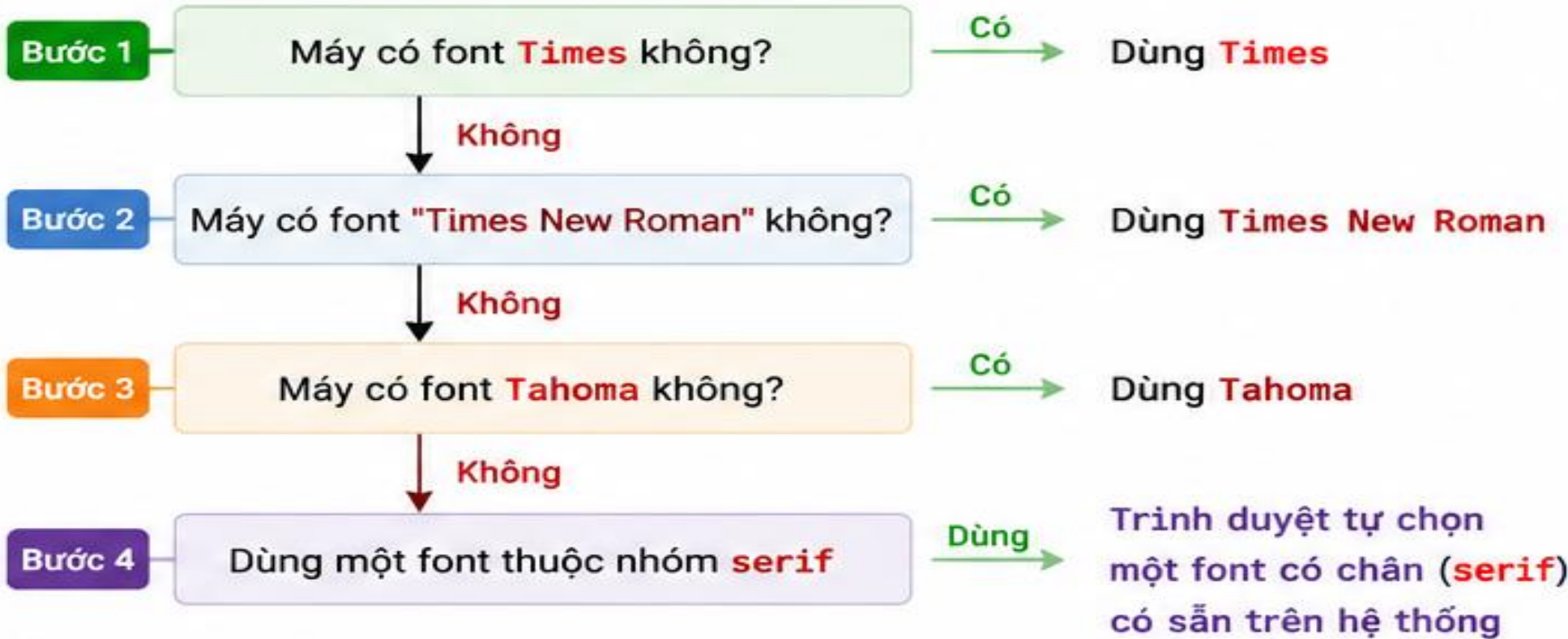
Dùng để chỉ định **kiểu chữ (font)** cho văn bản.

font-family: Times, "Times New Roman", Tahoma, serif;

→ Đây là danh sách font dự phòng
(**font stack**)

1 CÁCH HOẠT ĐỘNG

Trình duyệt sẽ thử từng font theo thứ tự **từ trái sang phải**.
Font đầu tiên có sẵn trên máy sẽ được dùng.



2 GIẢI THÍCH CÁC THÀNH PHẦN

Thành phần	Loại	Ý nghĩa
Times	Tên font cụ thể	Ưu tiên 1: nếu máy có font Times thì dùng font này.
"Times New Roman"	Tên font cụ thể	Ưu tiên 2: nếu không có Times thì dùng Times New Roman.
Tahoma	Tên font cụ thể	Ưu tiên 3: nếu không có hai font trên thì dùng Tahoma.
serif	Nhóm font (generic family)	Nếu không có các font trên, trình duyệt sẽ chọn một font thuộc nhóm serif (font có chân).

3 VÌ SAO "TIMES NEW ROMAN" PHẢI CÓ DẤU NGOẶC KÉP?

ĐÚNG ✓

```
font-family: "Times New Roman";
```

Vì tên font có dấu cách.

SAI ✗

```
font-family: Times New Roman;
```

CSS sẽ hiểu là 3 font khác nhau:
Times, New, Roman

4 CÁC NHÓM FONT (GENERIC FAMILY)

serif

Abc

Font có chân
Ví dụ: Times New Roman, Georgia, Garamond

sans-serif

Abc

Font không chân
Ví dụ: Arial, Helvetica, Calibri

monospace

Abc

Mỗi ký tự có cùng chiều rộng
Ví dụ: Courier New, Consolas

cursive

Abc

Kiểu chữ viết tay
Ví dụ: Brush Script, Lucida Handwriting

fantasy

Abc

Kiểu chữ trang trí
Ví dụ: Impact, Comic Sans MS



LƯU Ý:

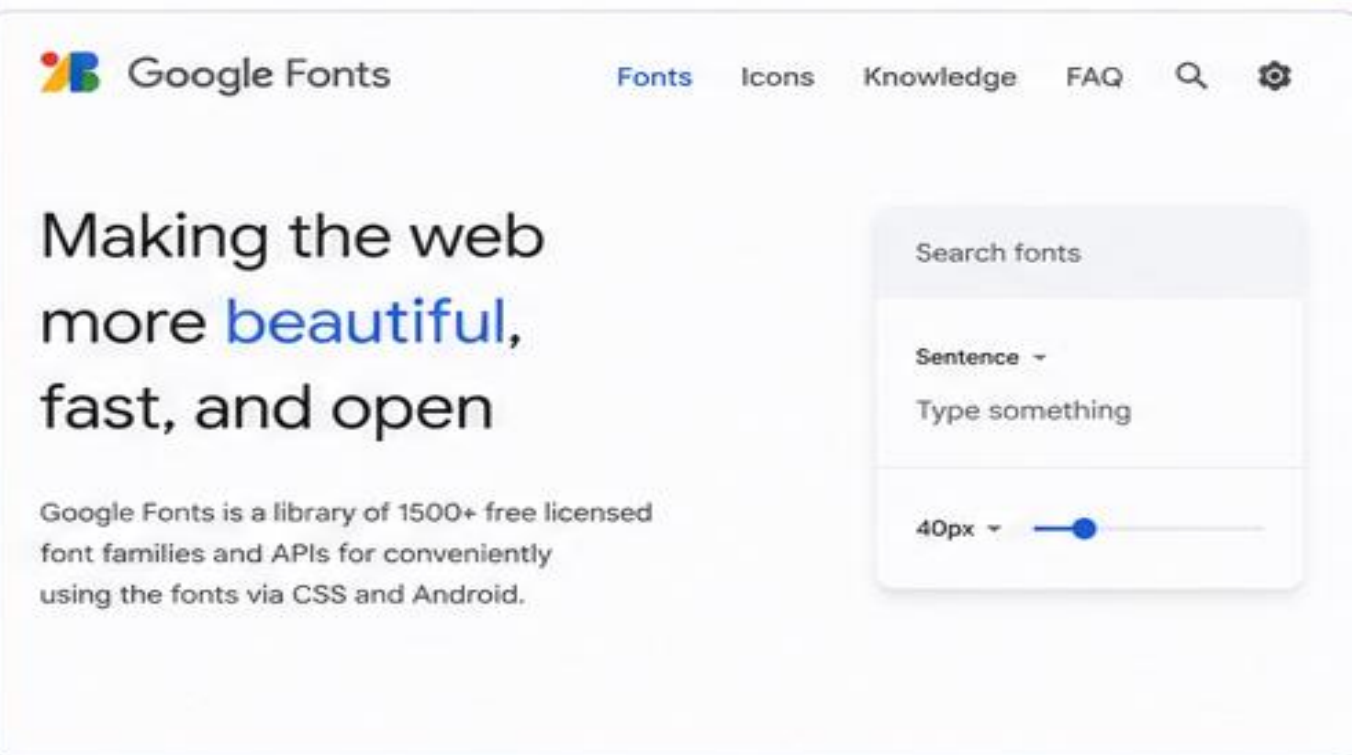
Luôn khai báo nhiều font theo thứ tự ưu tiên để đảm bảo văn bản hiển thị đẹp trên mọi thiết bị và mọi hệ điều hành.

HƯỚNG DẪN DÙNG GOOGLE FONT TRÊN WEBSITE

Các bước: Tìm font → Lấy link embed → Nhúng vào <head> → Sử dụng trong CSS

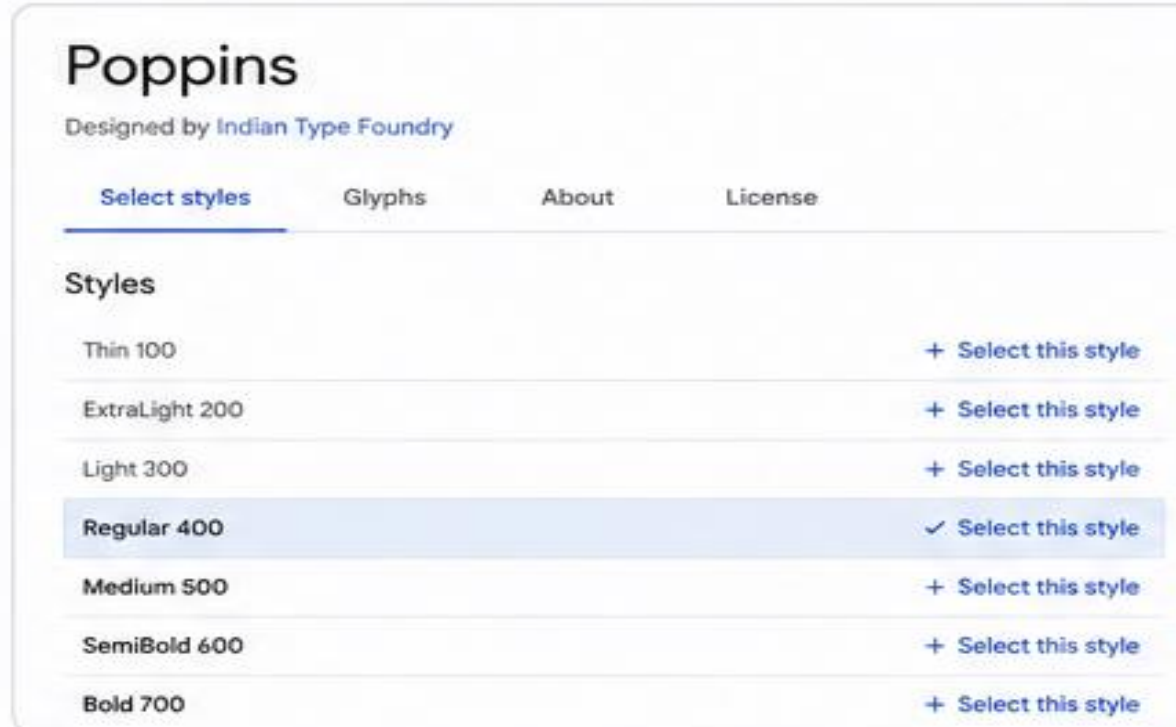
1 VÀO TRANG GOOGLE FONTS

Truy cập: <https://fonts.google.com>



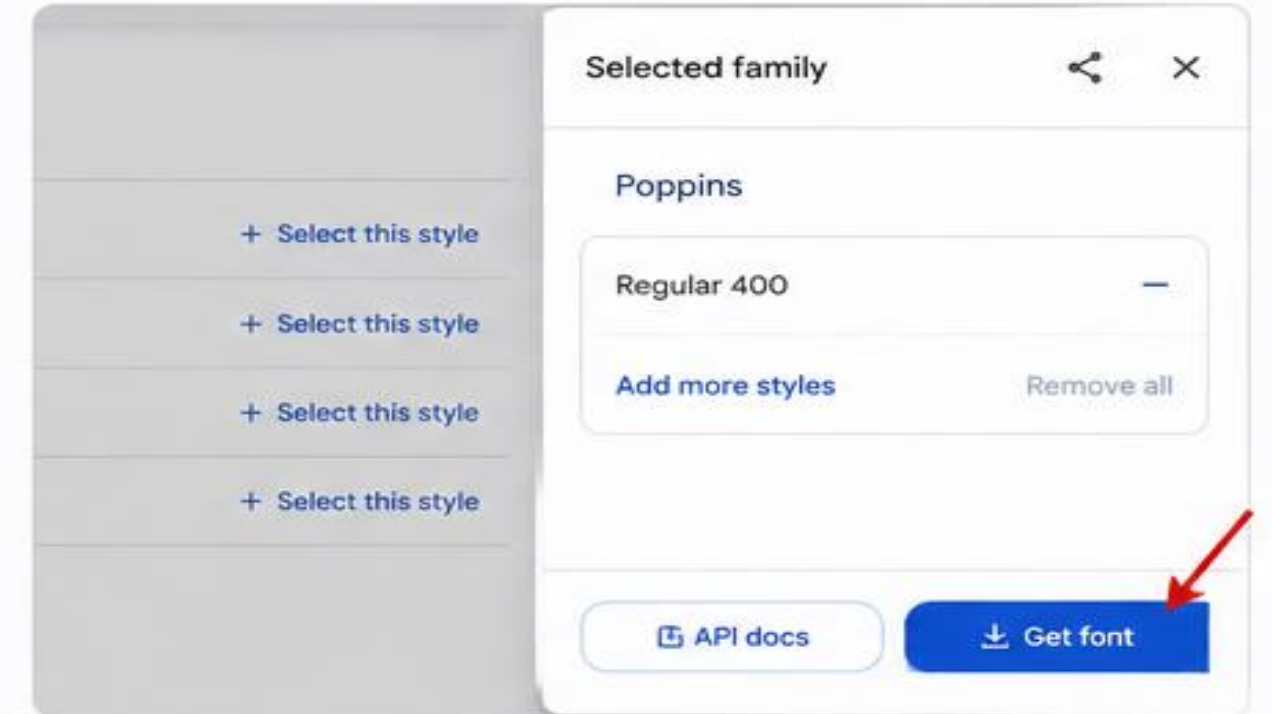
2 TÌM VÀ CHỌN FONT

- Gõ tên font bạn muốn, ví dụ: "Poppins"
- Chọn style (độ đậm, nghiêng) cần dùng



3 NHẤN "GET FONT"

- Sau khi chọn style, nhấn nút "Get font" ở góc phải.
- Một cửa sổ bên phải sẽ hiện ra.



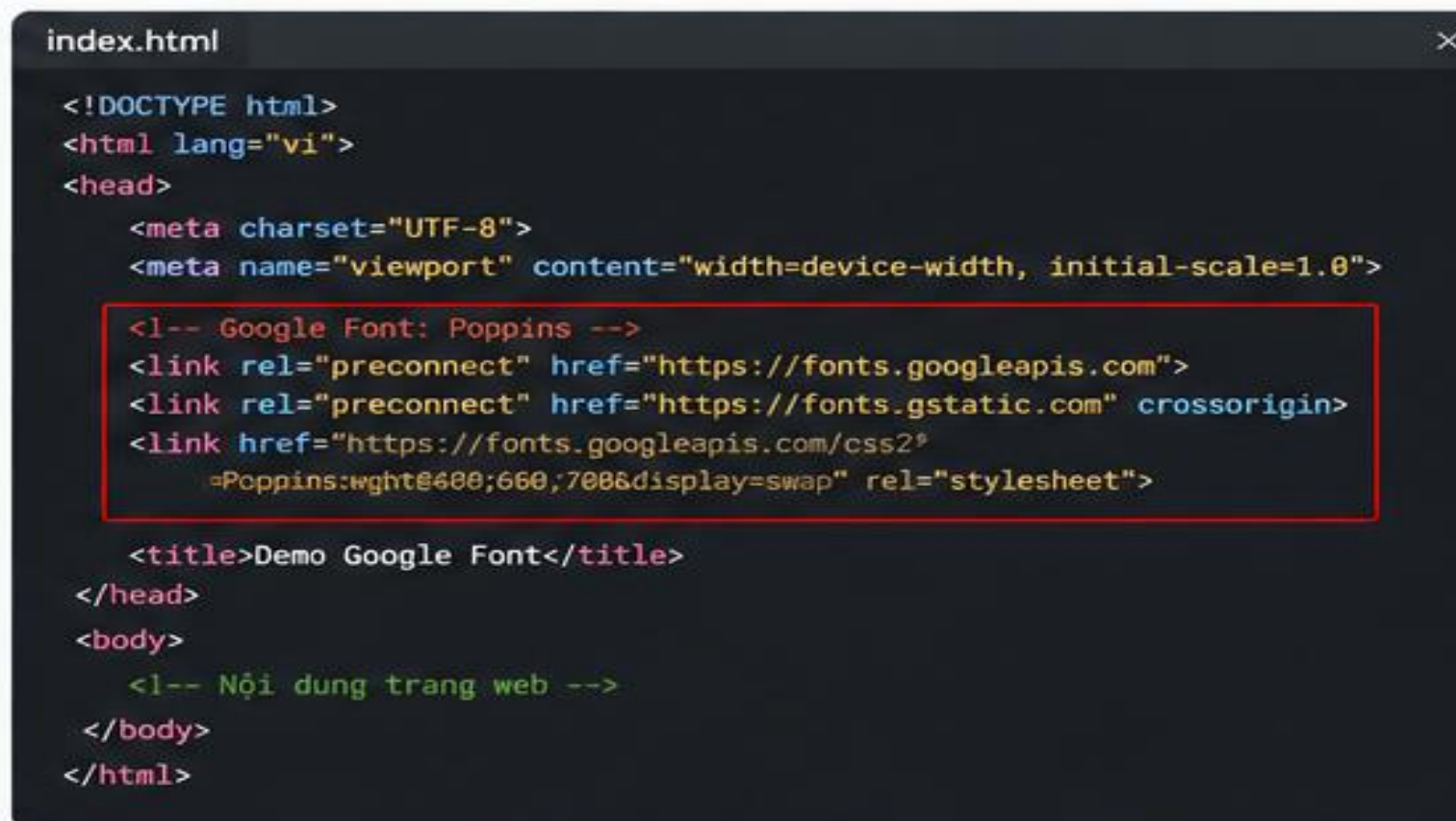
4 CHỌN "EMBED" VÀ SAO CHÉP LINK

- Chọn tab "Embed".
- Sao chép đoạn <link> trong phần <head>.



5 DÁN LINK VÀO <HEAD>

Dán đoạn link vừa sao chép vào phần <head> của file HTML.



6 SỬ DỤNG FONT TRONG CSS

Dùng tên font vừa chọn trong CSS.



✓ KẾT QUẢ

Đây là tiêu đề

Đây là đoạn văn bản sử dụng Google Font **Poppins**.

CSS – CLASS VÀ ID

Dùng để **chọn phần tử** HTML và áp dụng **định dạng** CSS.

1. CLASS LÀ GÌ?

- Class dùng để **nhóm** các phần tử có **chung kiểu dáng**.
- Có thể dùng lại cho nhiều phần tử.
- Trong HTML: dùng thuộc tính `class="ten-class"`
- Trong CSS: chọn bằng dấu chấm `.ten-class`

VÍ DỤ CLASS

HTML

```
<h1 class="title">Tiêu đề</h1>
<p class="text">Đoạn văn 1</p>
<p class="text">Đoạn văn 2</p>
```

CSS

```
.title {
  color: blue;
  font-size: 24px;
}
.text {
  color: gray;
  line-height: 1.6;
}
```

KẾT QUẢ HIỂN THỊ

Tiêu đề

Đoạn văn 1

Đoạn văn 2

2. ID LÀ GÌ?

- ID dùng để chọn **một phần tử duy nhất**.
- Mỗi ID chỉ được dùng cho **1 phần tử** trong toàn bộ trang.
- Trong HTML: dùng thuộc tính `id="ten-id"`
- Trong CSS: chọn bằng dấu `#ten-id`

VÍ DỤ ID

HTML

```
<h1 id="header">Tiêu đề trang</h1>
<p>Đoạn văn bên dưới tiêu đề.</p>
<div id="footer">Chân trang</div>
```

CSS

```
#header {
  color: darkred;
  text-align: center;
}
#footer {
  background: #f0f0f0;
  padding: 10px;
  text-align: center;
}
```

KẾT QUẢ HIỂN THỊ

Tiêu đề trang

Đoạn văn bên dưới tiêu đề.

Chân trang

3. SO SÁNH CLASS VÀ ID

Tiêu chí	CLASS	ID
Cách khai báo trong HTML	<code>class="ten-class"</code>	<code>id="ten-id"</code>
Trong CSS chọn bằng	<code>.ten-class</code>	<code>#ten-id</code>
Số lượng sử dụng	Dùng cho nhiều phần tử	Chỉ dùng cho 1 phần tử
Tính duy nhất	Không duy nhất	Duy nhất
Mục đích sử dụng	Nhóm phần tử có cùng kiểu dáng	Xác định 1 phần tử duy nhất
Ưu tiên (Specificity)	Thấp hơn ID	Cao hơn CLASS

4. LƯU Ý

- ✓ Nên dùng **CLASS** cho các nhóm phần tử giống nhau.
- ✓ Dùng **ID** khi cần định dạng cho **1** phần tử duy nhất (ví dụ: header, footer, main, ...).
- ✓ Trong một trang HTML, không được trùng **ID**.
- ⚠ Tránh lạm dụng **ID** vì độ ưu tiên của **ID** cao, sẽ khó ghi đè style.



TÓM LẠI: **CLASS** dùng để tái sử dụng cho nhiều phần tử. **ID** dùng để xác định duy nhất một phần tử.

VÌ SAO KHÔNG ĐƯỢC TRÙNG ID TRONG HTML?

ID dùng để xác định **DUY NHẤT** một phần tử trong toàn bộ trang HTML.
Vì vậy, mỗi **ID** chỉ được phép xuất hiện **1 lần**.

1. ID LÀ GÌ?

- ID (viết tắt của Identifier) dùng để chọn một phần tử **duy nhất**.
- Trong HTML: dùng thuộc tính `id="ten-id"`
- Trong CSS: chọn bằng dấu `#ten-id`



VÍ DỤ ĐÚNG (ID DUY NHẤT)

HTML

```
<h1 id="header">Tiêu đề</h1>
<p id="content">Nội dung...</p>
<div id="footer">Chân trang</div>
```

CSS

```
#header { color: blue; }
#content { color: green; }
#footer { color: gray; }
```

KẾT QUẢ

Tiêu đề

Nội dung...

Chân trang



VÍ DỤ SAI (TRÙNG ID)

HTML

```
<h1 id="title">Tiêu đề 1</h1>
<h2 id="title">Tiêu đề 2</h2>
<p id="title">Đoạn văn</p>
```

```
<!-- TRÙNG ID -->
<!-- TRÙNG ID -->
```

CSS

```
#title {
  color: purple;
  font-size: 24px;
}
```

KẾT QUẢ (KHÔNG ĐÁNG TIN CẬY)

Trình duyệt chỉ áp dụng style cho phần tử đầu tiên tìm thấy.
Kết quả hiển thị có thể khác nhau giữa các trình duyệt!

Tiêu đề 1

Tiêu đề 2

Đoạn văn

2. LÝ DO KHÔNG ĐƯỢC TRÙNG ID

1. ĐẢM BẢO TÍNH DUY NHẤT



ID được thiết kế để xác định duy nhất một phần tử. Trùng ID làm mất đi ý nghĩa này.

2. CSS HOẠT ĐỘNG KHÔNG CHÍNH XÁC



CSS chọn #id sẽ chỉ áp dụng cho phần tử đầu tiên được tìm thấy (theo thứ tự trong DOM).

3. JAVASCRIPT HOẠT ĐỘNG SAI



`document.getElementById("id")` chỉ trả về phần tử đầu tiên. Các xử lý khác sẽ bị sai lệch.

4. ẢNH HƯỞNG ĐẾN LIÊN KẾT (#id)



Liên kết dạng `<a href="#id">` sẽ nhảy đến phần tử đầu tiên có ID đó.

5. GÂY LỖI & KHÓ BẢO TRÌ



Code khó hiểu, khó debug, dễ gây xung đột style, ảnh hưởng trải nghiệm người dùng.

3. KHI NÀO DÙNG CLASS THAY VÌ ID?



Khi bạn muốn áp dụng style hoặc hành vi cho **hiều** phần tử giống nhau, hãy dùng **CLASS** (.) thay vì ID.

Ví dụ:

```
<!-- HTML -->
<p class="note">Ghi chú 1</p>
<p class="note">Ghi chú 2</p>
<p class="note">Ghi chú 3</p>
```

```
/* CSS */
.note {
  color: orange;
  font-style: italic;
}
```

KẾT QUẢ

Ghi chú 1
Ghi chú 2
Ghi chú 3



TÓM LẠI

ID dùng để xác định **DUY NHẤT** một phần tử.
Không được trùng ID trong cùng một trang HTML!

BÀI TẬP HTML & CSS – THỰC HÀNH ĐỊNH DẠNG VĂN BẢN

Áp dụng các thuộc tính: line-height, font-family, letter-spacing, text-decoration, font-size, text-align, color, text-indent, text-shadow

1. GIỚI THIỆU BẢN THÂN

GIỚI THIỆU BẢN THÂN

Xin chào!
Mình tên là Nguyễn Văn A.
Mình rất thích học lập trình Web.
Mục tiêu của mình là trở thành lập trình viên.

Yêu cầu:

- Tiêu đề màu xanh.
- Tiêu đề căn giữa.
- Font chữ Arial.
- Tiêu đề 36px.
- Đoạn văn 20px.
- Đầu dòng thụt vào 40px.
- Khoảng cách giữa các dòng 1.8.
- Có bóng chữ cho tiêu đề.

2. TRÍCH DẪN NỔI TIẾNG

THÀNH CÔNG

"Muốn thành công hãy bắt đầu từ những việc nhỏ."

— Khuyết danh

Yêu cầu:

- Tiêu đề màu đỏ.
- Tiêu đề có bóng.
- Câu nói in nghiêng.
- Gạch chân tên tác giả.
- Khoảng cách chữ 2px.

3. NỘI QUY LỚP HỌC

NỘI QUY LỚP HỌC

1. Đi học đúng giờ.
2. Không làm việc riêng.
3. Giữ vệ sinh lớp học.
4. Tích cực phát biểu.

Yêu cầu:

- Tiêu đề căn giữa.
- Font Times New Roman.
- Nội dung 22px.
- Khoảng cách dòng 2.
- Thụt đầu dòng.

4. BÀI THƠ

QUÊ HƯƠNG

Quê hương là chùm khế ngọt
Cho con trèo hái mỗi ngày
Quê hương là đường đi học
Con về rợp bướm vàng bay

Yêu cầu:

- Tiêu đề màu xanh.
- Font Georgia.
- Chữ 24px.
- Khoảng cách dòng 2.
- Căn giữa toàn bộ bài thơ.

5. QUẢNG CÁO

SIÊU KHUYẾN MÃI

GIẢM GIÁ 50%

Chỉ áp dụng đến hết ngày 30/08.

Nhanh tay mua ngay!

Yêu cầu:

- Tiêu đề lớn 48px.
- Màu đỏ.
- Có bóng chữ.
- "Giảm giá 50%" màu cam.
- Căn giữa.

6. BÁO ĐIỆN TỬ

TIN CÔNG NGHỆ

HTML và CSS là nền tảng quan trọng để xây dựng Website.

Việc luyện tập mỗi ngày sẽ giúp bạn thành thạo nhanh hơn.

Yêu cầu:

- Font Poppins.
- Tiêu đề 36px.
- Nội dung 20px.
- Thụt đầu dòng.
- Line-height 1.8.
- Letter-spacing 1px.

7. BÀI TỔNG HỢP

TRUNG TÂM TIN HỌC NEXTBASE

Khóa học HTML & CSS

Lập trình Web từ cơ bản đến nâng cao

Hotline: 0909 999 999

Yêu cầu:

Bắt buộc sử dụng tất cả các thuộc tính: **color, font-family, font-size, text-align, text-indent, line-height, letter-spacing, text-shadow, text-decoration.**

8. POSTER (KHÓ)

WELCOME TO HTML

Học lập trình Web thật dễ dàng!

Bắt đầu ngay hôm nay.

www.nextbase.edu.vn

Yêu cầu:

- Font Google (Poppins hoặc Roboto).
- Tiêu đề 50px.
- Có bóng chữ.
- Căn giữa.
- Đối màu từng dòng.
- Dòng cuối gạch chân.
- Điều chỉnh khoảng cách chữ.
- Điều chỉnh khoảng cách dòng.

9. GIỚI THIỆU KHÓA HỌC

KHÓA HỌC HTML & CSS

Bạn sẽ học được:

- HTML cơ bản
- CSS cơ bản
- Thiết kế Website
- Responsive

Đăng ký ngay hôm nay!

Yêu cầu:

Sử dụng tối thiểu: **font-family, font-size, color, text-align, text-shadow, letter-spacing, line-height, text-indent, text-decoration.**

10. THỬ THÁCH CUỐI BUỔI

★ HTML & CSS ★

Khóa học lập trình Web

HTML là nền tảng của Website.

CSS giúp Website đẹp hơn.

Đăng ký ngay hôm nay!

Yêu cầu:

- Không dùng Margin, Padding, Border, Background, Flexbox, Grid.
- Chỉ sử dụng các thuộc tính đã học.
- Thiết kế giống mẫu.



GHI NHỚ:

- **CLASS** dùng cho nhiều phần tử giống nhau.
- **ID** dùng để xác định duy nhất 1 phần tử.



Hãy thực hành để thành thạo các thuộc tính CSS đã học!