



21

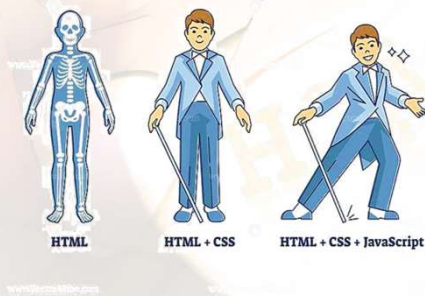
Lịch sử

- 1991 **HTML** (Tim Berners-Lee tạo 20 thẻ HTML đầu tiên)
- 1994 **HTML 2**
- 1996 **CSS 1** + **JavaScript**
- 1997 **HTML 4**
- 1998 **CSS 2**
- 2000 **XHTML 1**
- 2002 **Tableless Web Design**
- 2005 **AJAX**
- 2009 **HTML 5** (Chính thức 2014)

22

Khái niệm

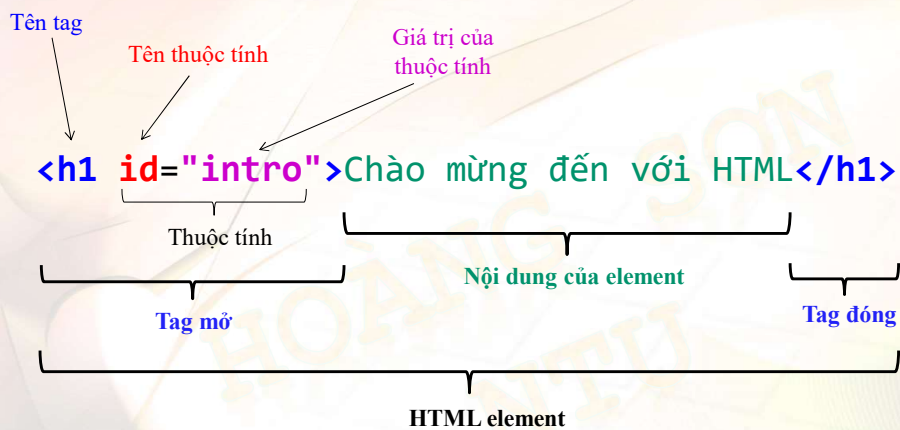
❖ **HTML** (HyperText Markup Language) : ngôn ngữ được dùng để mô tả **cấu trúc** và **nội dung** của trang web



- Không phải là ngôn ngữ lập trình.
- Không phải là ngôn ngữ "định dạng" (tạo hình thức)
- HTML sử dụng những mã "đánh dấu" – **phần tử (element)** – để mô tả các thành phần của văn bản siêu liên kết.

23

HTML Element

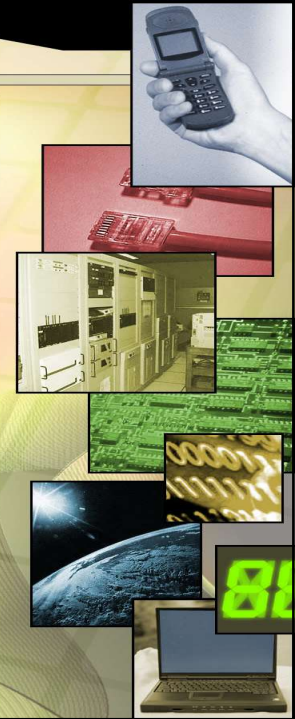


☞ **Các phần tử HTML có thể lồng vào nhau (phần nội dung của phần tử này chứa phần tử khác)**

24

HTML Element (tt)

- ❖ Tên tag của phần tử HTML không phân biệt chữ hoa-thường, nhưng nên là **chữ thường**.
- ❖ Một số phần tử HTML không có tag đóng: **
, **<input>,... Tuy nhiên để tương thích với các hệ thống cũ thì nên sử dụng thêm ký tự **/**, ví dụ như **
, **<input />,...
- ❖ Trong phần nội dung của phần tử HTML, khi hiển thị trên trình duyệt **các từ chỉ cách nhau tối đa 1 khoảng trắng** (dù trong code có nhiều khoảng trắng). Sử dụng **Enter** (xuống dòng) trong code thì hiển thị **tương đương với 1 khoảng trắng** (không xuống dòng)



25

Phân loại HTML Element

- ❖ Mỗi phần tử HTML được phân loại **hiển thị nội dung** trên trình duyệt mặc định theo 2 loại: phần tử khối (**block**) và phần tử dòng (**inline**)
- ❖ **Phần tử khối** luôn **bắt đầu trên một dòng mới** và **chiếm toàn bộ chiều rộng có thể có** (không phụ thuộc nội dung phần tử).
- ❖ **Phần tử dòng** không bắt đầu dòng mới và có **độ rộng tùy vào nội dung của phần tử**.



26

Phân loại HTML Element (tt)

❖ Một số phần tử khối

<address>, <article>, <aside>, <blockquote>,
<canvas>, <dd>, <div>, <dl>, <dt>,
<fieldset>, <figcaption>, <figure>, <footer>,
<form>, <h1> - <h6>, <header>, <hr>, ,
<main>, <nav>, <noscript>, , <p>, <pre>,
<section>, <table>, <tfoot>,

27

Phân loại HTML Element (tt)

❖ Một số phần tử dòng

<a>, <abbr>, <audio>, , <bdo>,
,
<button>, <cite>, <code>, , <i>, ,
<input>, <kbd>, <label>, <map>, <object>,
<output>, <q>, <s>, <samp>, <script>,
<select>, <small>, , , <sub>,
<sup>, <style>, <textarea>, <time>, <var>

👉 **Lưu ý:** Một phần tử dòng không chứa phần tử khối khác, trừ ngoại lệ thẻ <a> có thể chứa các phần tử khối.

28

Class & Id

❖ Class

- Thuộc tính class dùng để **tạo một hoặc nhiều nhóm** các phần tử và có thể dùng trong CSS hoặc JavaScript thực hiện một số tác vụ nhất định cho phần tử có class đó.
- Một class có thể **áp dụng cho một hoặc nhiều** phần tử, và các phần tử này không nhất thiết phải cùng loại thẻ.
- Một class có thể **nhận được nhiều giá trị**, mỗi giá trị cách nhau bằng một khoảng trắng
- Giá trị class có **phân biệt chữ hoa chữ thường**.

```
<h1 class="this that">Hello HTML</h1>
```

29

Class & Id (tt)

❖ Id

- Thuộc tính id dùng để xác định một id **duy nhất cho một phần tử** và có thể dùng trong CSS hoặc JavaScript thực hiện một số tác vụ nhất định cho phần tử có id đó.
- Thường được dùng để **phân biệt** các phần tử có **cùng tên thẻ**.
- Trong một trang các phần tử không được phép trùng giá trị id.
- Giá trị id có **phân biệt chữ hoa chữ thường**.

```
<div id="content">Vùng nội dung trang</div>
```

👉 **Lưu ý:** tên của class và id chỉ bao gồm **chữ cái, chữ số**, dấu **_**, dấu **-** và **không được bắt đầu bằng chữ số**.

30

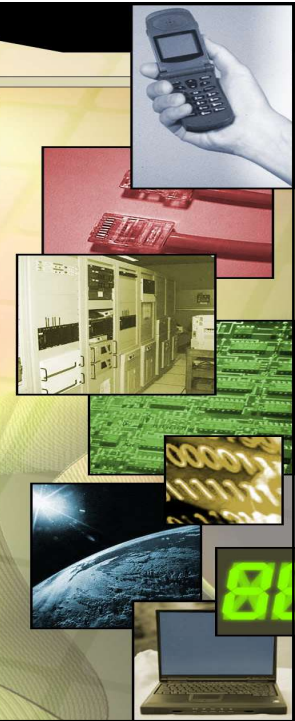
Class & Id (tt)

Thuộc tính id

- Một phần tử chỉ nhận một giá trị id.
- Một id chỉ có thể dùng cho một phần tử duy nhất.
- Khi muốn chọn một phần tử dựa theo id thì ta dùng cú pháp **#giatri-id**

Thuộc tính class

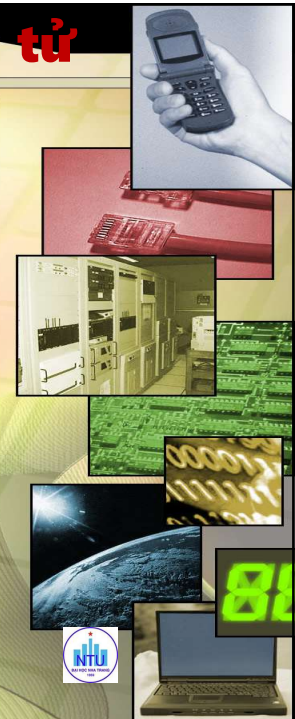
- Một phần tử có thể nhận một hoặc nhiều giá trị class.
- Một class có thể dùng cho nhiều phần tử khác nhau.
- Khi muốn chọn một phần tử dựa theo class thì ta dùng cú pháp **.giatri-class**



31

Ký tự đặc biệt trong nội dung phần tử

Kết quả hiển thị	Ý nghĩa	Code trong văn bản HTML	Code số
	một khoảng trắng	&nbsp;	&#160;
<	nhỏ hơn	&lt;	&#60;
>	lớn hơn	&gt;	&#62;
&	dấu và	&amp;	&#38;
"	nháy kép	&quot;	&#34;
'	nháy đơn	&apos;	&#39;



32

Ký tự đặc biệt trong nội dung phần tử (tt)

Kết quả hiển thị	Ý nghĩa	Code trong văn bản HTML	Code số
¢	cent	¢	¢
£	pound	£	£
¥	yen	¥	¥
§	section	§	§
©	copyright	©	©
®	registered trademark	®	®
×	nhân	×	×
÷	chia	÷	÷

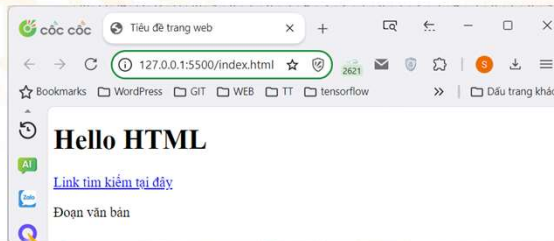
33

Cấu trúc tài liệu HTML

```

1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Tiêu đề trang web</title>
7 </head>
8 <body>
9   <h1>Hello HTML</h1>
10  <a href="http://www.google.com.vn">Link tìm kiếm tại đây</a>
11  <p>Đoạn văn bản</p>
12 </body>
13 </html>

```

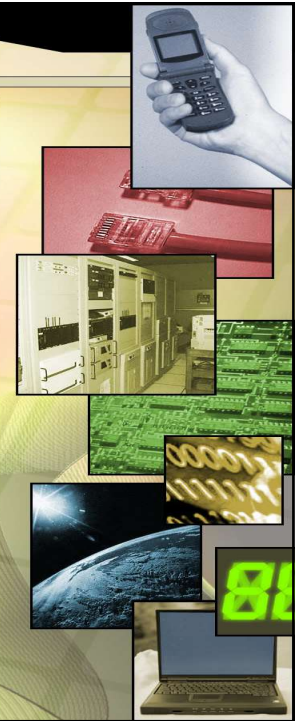


Tạo nhanh 1 tài liệu HTML trong VS Code: gõ !

34

Cấu trúc tài liệu HTML (tt)

- **<!DOCTYPE html>**: định nghĩa phiên bản cụ thể của HTML, cụ thể là phiên bản 5
- **<html></html>**: phần tử bắt buộc, cấp cao nhất, định nghĩa một tài liệu HTML (trang web)
- **<head></head>**: khai báo các siêu dữ liệu (metadata) bổ sung của trang web như: tiêu đề trang, bảng mã...
- **<title></title>**: dùng để khai báo tiêu đề của trang web
- **<body></body>**: chứa các phần tử sẽ được hiển thị trên màn hình trình duyệt.
- **<h1></h1>**: phần tử xác định một tiêu đề.
- **<a>**: phần tử xác định một liên kết
- **<p></p>**: phần tử xác định một đoạn văn bản.



35

Chú thích (comment) trong tài liệu HTML

- ❖ Thông tin được ghi thêm vào code trong tài liệu HTML nhưng không ảnh hưởng đến hoạt động của trang web, không hiển thị nội dung trên trình duyệt
- ❖ Có tác dụng giải thích, ghi chú đoạn code để sau này khi đọc lại hoặc khi sửa code sẽ dễ dàng hơn
- ❖ Cú pháp: **<!-- Nội dung comment -->**
- ❖ Phím tắt: **Ctrl + /** (Windows) hoặc **Cmd + /** (Mac)



36

Các phần tử cơ bản của HTML

Tag	Mô Tả
<code><div>...</div></code>	Xác định một vùng
<code><h1>...</h1></code> <code><h6>...</h6></code>	Xác định tiêu đề văn bản từ 1 đến 6 (1 lớn nhất, 6 nhỏ nhất)
<code><p>...</p></code>	Xác định một đoạn văn bản
<code>
</code>	Chèn một dòng trắng (xuống dòng)
<code><hr /></code>	Chèn một đường thẳng

37

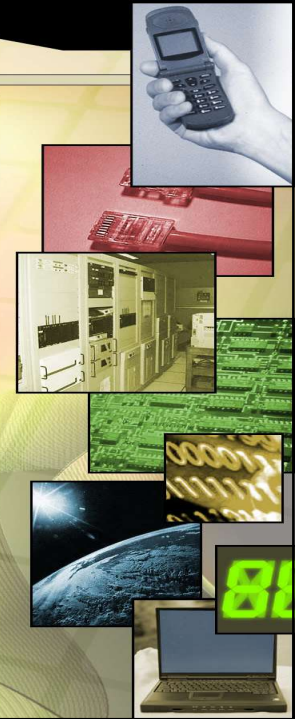
Các phần tử cơ bản của HTML (tt)

Tag	Mô Tả
<code>...</code> <code>...</code>	Văn bản in đậm
<code><i>...</i></code> <code>...</code>	Văn bản in nghiêng
<code><small>...</small></code>	Văn bản chữ nhỏ
<code><sup>...</sup></code>	Văn bản chỉ số trên, ví dụ x^2
<code><sub>...</sub></code>	Văn bản chỉ số dưới, ví dụ x_2

38

Semantic Elements

- ❖ Là những phần tử có nội dung bên trong có ý nghĩa tương ứng với tên phần tử đó.
- ❖ Mô tả ý nghĩa về cấu trúc của phần tử đó đối với các trình duyệt và lập trình viên
- ❖ Hỗ trợ tốt hơn đối với các công cụ tìm kiếm
- ❖ Thay thế thẻ `<div>` trong bố cục nội dung tài liệu HTML

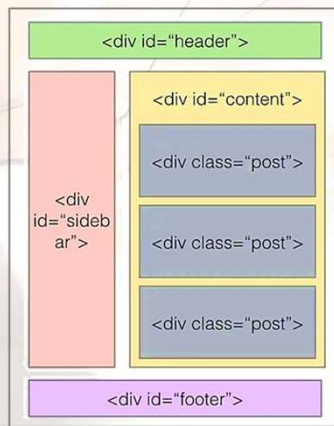


39

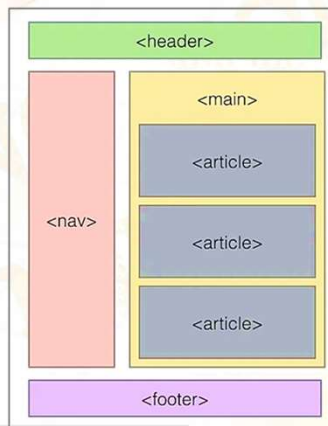
Semantic Elements (tt)

HTML4 vs HTML5 Page Structure on a Blog

HTML4: Lots of Classes/IDs



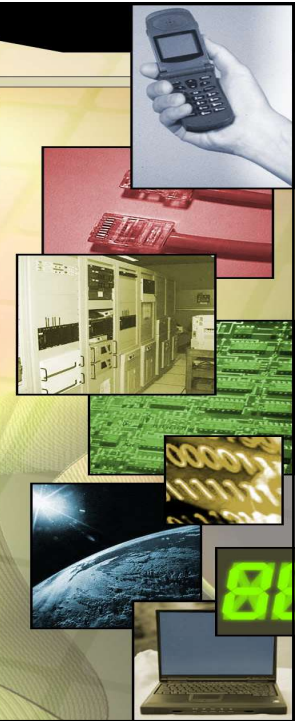
HTML5: Semantic Tags/Sections



40

Semantic Elements (tt)

Tag	Mô tả
<code><article>...</article></code>	Xác định một bài viết/bài báo
<code><aside>...</aside></code>	Xác định nội dung nằm ở rìa trang
<code><details>...</details></code>	Xác định vùng ẩn/hiện nội dung
<code><figcaption>...</figcaption></code>	Chú thích cho phần tử <figure>
<code><figure>...</figure></code>	Vùng đánh dấu nội dung ảnh trong tài liệu
<code><footer>...</footer></code>	Xác định phần cuối của một tài liệu/section
<code><header>...</header></code>	Xác định phần đầu của một tài liệu/section
<code><main>...</main></code>	Chỉ định vùng nội dung chính của một tài liệu
<code><mark>...</mark></code>	Xác định vùng đánh dấu/làm nổi bật văn bản
<code><nav>...</nav></code>	Xác định vùng chứa các link điều hướng (menu)
<code><section>...</section></code>	Xác định một phần (vùng) trong tài liệu
<code><summary>...</summary></code>	Hiển thị tiêu đề cho phần tử <details>
<code><time>...</time></code>	Xác định ngày/giờ

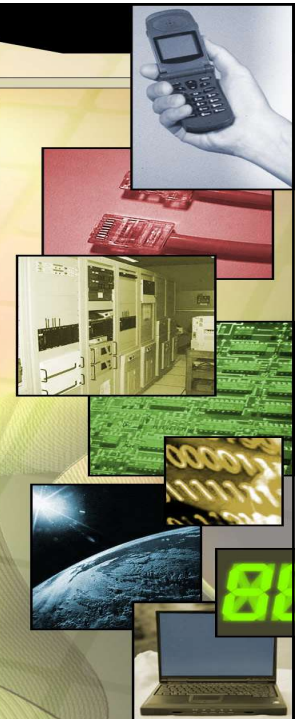


41

Thẻ liên kết

`Text_Liên_kết`

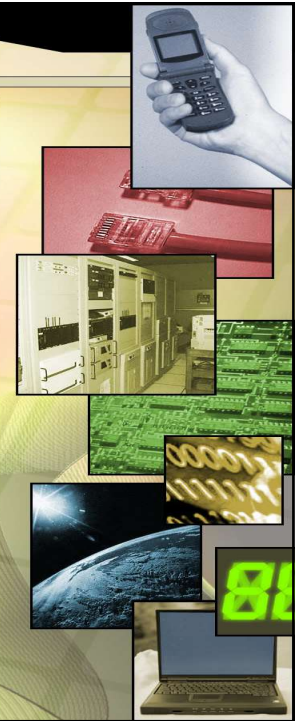
- ❖ Tạo liên kết để chuyển hướng đến tài nguyên web khác
- ❖ Thuộc tính:
 - **href** (Hypertext **R**eference): thuộc tính xác định địa chỉ liên kết.
 - **target**: thuộc tính xác định nơi mà tài liệu được mở.
 - **_self**: (mặc định) mở tài liệu ở tab hiện tại.
 - **_blank**: mở tài liệu trong tab mới.
 - **title**: Thông tin bổ sung cho liên kết (xuất hiện khi di chuyển chuột lên liên kết)
 - **name**: đánh dấu vị trí của liên kết trong tài liệu (sử dụng trong liên kết nội)



42

Thẻ liên kết (ví dụ 1)

```
<a href="https://www.google.com.vn">Tìm kiếm bằng Google</a>
<a href="../../../myfolder/sample.htm">Link tới trang sample</a>
<a href="mailto:abc@yahoo.com">
  Link tới email abc@yahoo.com
</a>
```



43

Thẻ liên kết (ví dụ 2)

```
<body>
  <p><a href="#C2">Tới chapter 2</a></p>
  <h2>Chapter 1</h2>
  <p>Chapter 1 là ..... </p>
  <h2><a name="C2">Chapter 2</a></h2>
  <p>Chapter 2 là ..... </p>
</body>
```



44

Chèn ảnh

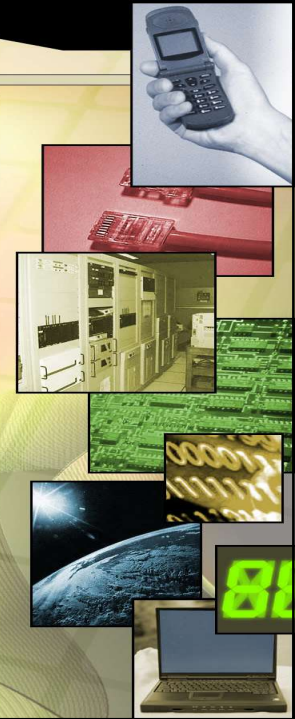
```

```

- ❖ Nhúng ảnh vào trang web.
- ❖ Phần tử không có nội dung, không có thẻ đóng và là phần tử dòng.
- ❖ Thuộc tính:
 - **src** (source - nguồn): đường dẫn của ảnh.
 - **alt**: (alternate - thay thế): văn bản thay thế cho hình ảnh, hiển thị khi đường dẫn ảnh bị lỗi.

```

```



45

Chèn video

```
<video src="đường_dẫn_video"></video>
```

hoặc

```
<video>
```

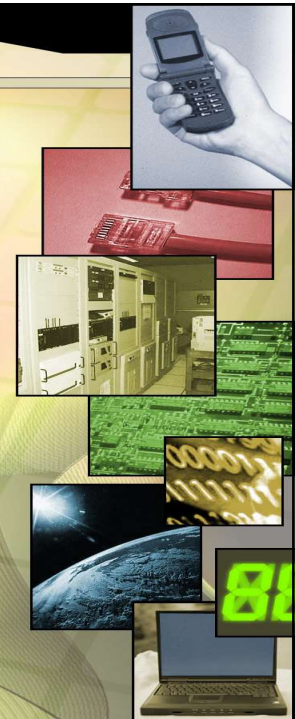
```
<source src="media/movie.mp4" type="video/mp4">
```

```
<source src="media/movie.ogg" type="video/ogg">
```

Your browser does not support the video tag.

```
</video>
```

- ❖ Nhúng video vào trang web.
- ❖ Phần tử dòng

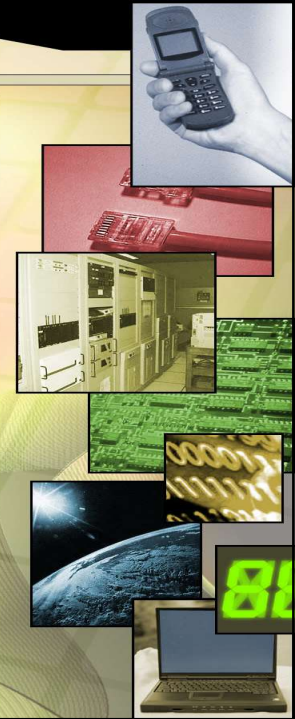


46

Chèn video (tt)

❖ Thuộc tính:

- **src** : đường dẫn của video.
- **controls** : thêm điều khiển cho video (mở, dừng, tua tới-lùi...)
- **autoplay** : tự động phát video khi trang web mở.
- **poster** : ảnh đại diện của video (độc lập với nội dung video)
 - **url** : đường dẫn ảnh đại diện video
- **loop**: phát video ở chế độ lặp.
- **muted**: phát video ở chế độ câm (sử dụng cho Google Chrome nếu muốn thêm **autoplay**)
- **preload**: chế độ tải trước video
 - **auto**: tự động tải trước
 - **none**: không tự động tải trước



47

Chèn audio

```
<audio src="đường_dẫn_audio"></audio>
```

hoặc

```
<audio>
```

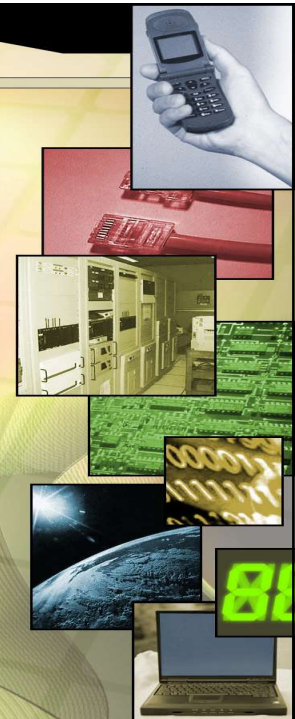
```
  <source src="media/audio.mp3" type="audio/mpeg">
```

```
  <source src="media/audio.ogg" type="audio/ogg">
```

Your browser does not support the audio tag.

```
</audio>
```

- ❖ Nhúng audio vào trang web.
- ❖ Phần tử dòng
- ❖ Các thuộc tính tương tự như phần tử **<video>** (không có thuộc tính **poster**)



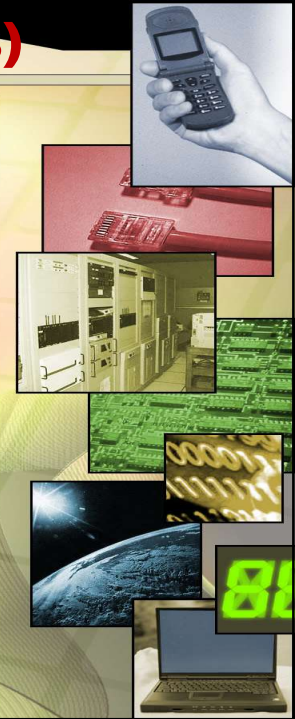
48

Đường dẫn tài nguyên (file paths)

- ❖ **Absolute File Paths** (Đường dẫn tuyệt đối): là URL đầy đủ của file tài nguyên trên Internet

```

```



49

Đường dẫn tài nguyên (file paths) - tt

- ❖ **Relative File Paths** (Đường dẫn tương đối): đường dẫn liên quan đến tài liệu hiện tại (tài liệu **đang viết code**). Thường dùng để tham chiếu đến các tài nguyên cùng website

```

```

```
<!-- file logo.jpg cùng folder với trang hiện tại -->
```

```

```

```
<!-- file logo.jpg trong folder con media (folder media cùng folder với trang hiện tại) -->
```

```

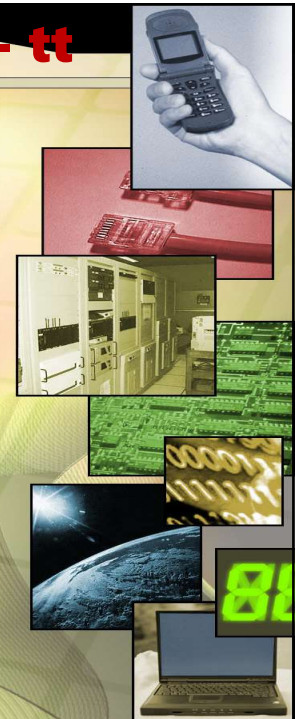
```

```
<!-- file logo.jpg trong folder cao hơn 1 cấp với trang hiện tại -->
```

```

```

```
<!-- file logo.jpg trong folder gốc của website (không nên sử dụng) -->
```



50

Bảng

Tag	Mô Tả
<code><table>...</table></code>	Tạo bảng
<code><tr>...</tr></code>	Hàng trong bảng
<code><td>...</td></code>	Ô trong hàng
<code><th>...</th></code>	Hàng đầu của bảng (trong phần tử <code><tr></code>)
<code><caption>...</caption></code>	Nhãn của bảng (trong phần tử <code><table></code>)

51

Bảng (tt)

```

<html>
<body>
  <h4>Nhóm 2 ô hàng ngang thành 1 ô</h4>
  <table border="1">
    <tr>
      <td>Name</td>
      <td colspan="2">Telephone</td>
    </tr>
    <tr>
      <td>Bill Gates</td>
      <td>555 77 854</td>
      <td>555 77 855</td>
    </tr>
  </table>
</body>
</html>

```

Nhóm 2 ô hàng ngang thành 1 ô

Name	Telephone	
Bill Gates	555 77 854	555 77 855

52

Bảng (tt)

```
<html>
<body>
  <h4>Nhóm 2 ô hàng dọc thành 1 ô</h4>
  <table border="1">
    <tr>
      <td>First Name:</td>
      <td>Bill Gates</td>
    </tr>
    <tr>
      <td rowspan="2">Telephone:</td>
      <td>555 77 854</td>
    </tr>
    <tr>
      <td>555 77 855</td>
    </tr>
  </table>
</body>
</html>
```

Nhóm 2 ô hàng dọc thành 1 ô

First Name:	Bill Gates
Telephone:	555 77 854
	555 77 855

53

Danh sách

Tag	Mô Tả
...	Danh sách có sắp xếp
...	Danh sách không sắp xếp
...	1 phần tử trong danh sách

```
<ul><strong>Thực đơn điểm tâm</strong>
  <li type="circle">Bò beefsteak</li>
  <li type="disc">Salad trộn</li>
  <li type="square">Nước trái cây ép</li>
</ul>
```

Thực đơn điểm tâm

- Bò beefsteak
- Salad trộn
- Nước trái cây ép

54

Danh sách (tt)

❖ Định dạng danh sách

➤ Định dạng cho toàn bộ danh sách

```
<ol type="giá_trị">...</ol>
```

```
<!-- giá_trị: A, a, I, i -->
```

```
<ol start="n">...</ol>
```

```
<!-- bắt đầu danh sách với thứ tự n -->
```

➤ Định dạng cho từng mục

```
<li type="giá_trị">...</li>
```

```
<!-- Tại vị trí phần tử sẽ đánh chỉ số theo giá_trị -->
```

```
<li value="n">...</li>
```

```
<!-- Tại vị trí phần tử sẽ đánh thứ tự n-->
```

55

Danh sách (tt)

II. HTML

- A. Minimal Page
- B. Headings
- C. Tags
- D. Lists
 - i. Unordered
 - ii. Ordered
 - iii. Definition
 - iv. Nested

- E. Links
 - i. Absolute
 - ii. Relative

- F. Images

III. CSS

- A. Anatomy
- B. Basic Selectors
 - i. Element
 - ii. Class
 - iii. ID
 - iv. Group
- C. The DOM
- D. Advanced Selectors
- E. Box Model

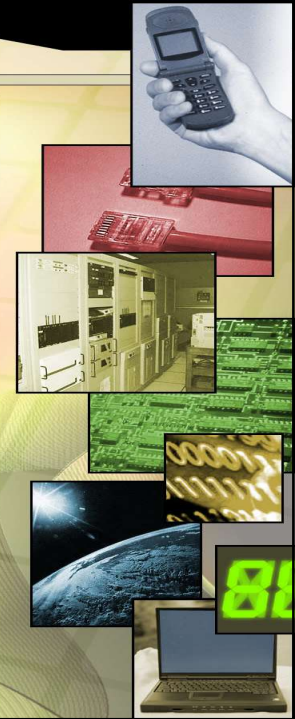
56

Danh sách (tt)

```

<ol type="I" start="2">
  <li>HTML
    <ol type="A">
      <li>Minimal Page</li>
      <li>Headings</li>
      <li>Tags</li>
      <li>List
        <ol type="i">
          <li>Unordered</li>
          <li>Ordered</li>
          <li>Definition</li>
          <li>Nested</li>
        </ol>
      </li>
      <li>Links
        <ol type="i">
          <li>Absolute</li>
          <li>Relative</li>
        </ol>
      </li>
      <li>Images</li>
    </ol>
  </li>
  <li>CSS
    <ol type="A">
      <li>Anatomy</li>
      <li>Basic Selectors
        <ol type="i">
          <li>Element</li>
          <li>Class</li>
          <li>Id</li>
          <li>Group</li>
        </ol>
      </li>
      <li>The DOM</li>
      <li>Advanced Selectors</li>
      <li>Box Model</li>
    </ol>
  </li>
</ol>

```



57

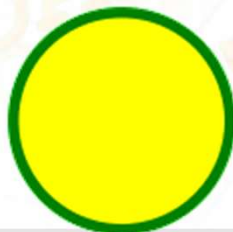
Đồ họa

<svg>...</svg> : tạo vùng vẽ đối tượng đồ họa cơ bản

```

<svg width="100" height="100">
  <circle cx="50" cy="50" r="40" stroke="green"
    stroke-width="4" fill="yellow"/>
</svg>

```

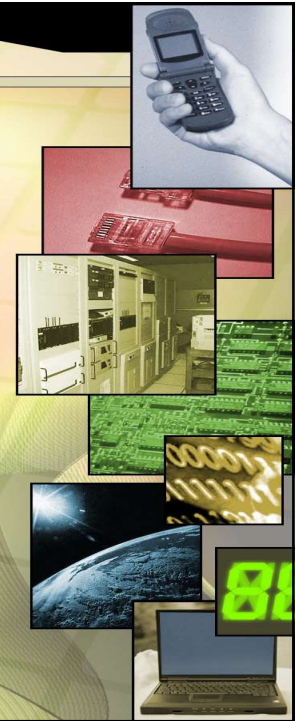


58

Đồ họa (tt)

`<canvas>...</canvas>` : tạo vùng vẽ đồ họa tự do

```
<canvas id="myCanvas"></canvas>
<script>
  var c = document.getElementById("myCanvas");
  var ctx = c.getContext("2d");
  ctx.beginPath();
  ctx.arc(50, 50, 40, 0, 2*Math.PI);
  ctx.fillStyle = "yellow";
  ctx.fill();
  ctx.strokeStyle = "green";
  ctx.lineWidth = 4;
  ctx.stroke();
</script>
```



59

Biểu mẫu - Form

- ❖ Phần tử `<form>` xác định hình thức thu thập dữ liệu đầu vào của người dùng. Đầu vào của người dùng thường được gửi đến máy chủ để xử lý.

```
<form action="action_page.php" method="post">
  <label for="fullName">Họ tên:</label>
  <br />
  <input type="text" id="fullName" name="full_name" />
  <br />
  <label for="email">Email:</label>
  <br />
  <input type="text" id="email" name="e_mail" />
  <br /><br />
  <input type="submit" value="Gửi" />
</form>
```




60

Biểu mẫu - Form (tt)

Họ tên:

Email:




61

Thuộc tính form

➤ **action**: định nghĩa việc gì sẽ xảy ra khi biểu mẫu được gửi đi. Giá trị là đường dẫn chỉ đến file chứa code xử lý dữ liệu đầu vào mà form gửi tới.

➤ **target**: xác định nơi hiển thị trang phản hồi của form

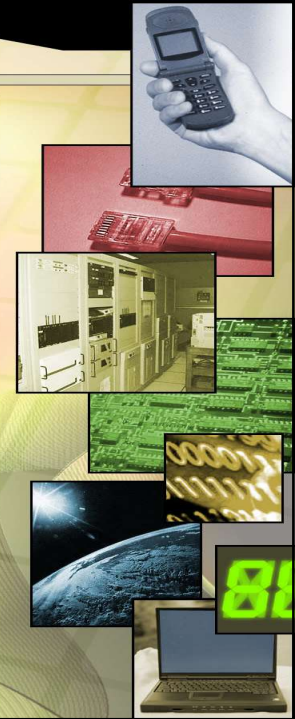
- **_self**: (mặc định) ở tab hiện tại.
- **_blank**: ở tab mới.



62

Thuộc tính form (tt)

- **method**: xác định phương pháp HTTP (GET hay POST) sẽ được dùng khi gửi dữ liệu trong biểu mẫu đến máy chủ
 - **GET**: (mặc định) nối dữ liệu biểu mẫu vào URL, theo cặp **name=value** nhìn thấy được trong URL của trang phản hồi.
`/action_page.php?firstname=James&lastname=Bond`
 - ✓ Độ dài URL bị giới hạn (2048 ký tự) → dữ liệu đầu vào bị hạn chế
 - ✓ Không dùng GET để gửi các dữ liệu nhạy cảm (do nhìn thấy được trong URL)
 - ✓ Hữu ích khi người dùng muốn đánh dấu (bookmark) trang kết quả.
 - **POST**: không hiển thị dữ liệu lên địa chỉ trang phản hồi
 - ✓ Không giới hạn kích thước dữ liệu đầu vào.
 - ✓ Không đánh dấu được trang phản hồi



63

Thuộc tính form (tt)

- **autocomplete**: Khi bật tính năng tự động điền, trình duyệt sẽ tự động gợi ý các giá trị dựa trên các giá trị mà người dùng đã nhập trước đó.
 - **on** (mặc định)
 - **off**

Thuộc tính autocomplete có thể áp dụng ở các phân tử nhập liệu của form
- **novalidate**: Thuộc tính này đặc tả dữ liệu form không cần phải kiểm tra tính chính xác dữ liệu khi gửi đi.



64

Phần tử Form

Tag	Mô Tả
<code><input ... /></code>	Phần tử nhập liệu cơ bản
<code><textarea>...</textarea></code>	Vùng nhập liệu văn bản có nhiều hàng
<code><select>...</select></code>	Nhập liệu kiểu danh sách chọn
<code><button>...</button></code>	Nút bấm
<code><label>...</label></code>	Nhãn của 1 phần tử form
<code><fieldset>...</fieldset></code>	Nhóm các phần tử nhập liệu
<code><legend>...</legend></code>	Nhãn của 1 fieldset

65

Phần tử Form (tt)

Lưu ý:

- Tất cả các phần tử nhập liệu của form **phải có thuộc tính name**, nếu không dữ liệu đó sẽ không thể gửi đi.

```
<input type="text" id="fullName" name="f_name" />
```

- Giá trị thuộc tính **for** của thẻ `<label>` phải giống giá trị thuộc tính **id** của phần tử nhập liệu để **liên kết** chúng lại với nhau. Khi đó, **click vào <label>** thì con trỏ sẽ **tự động focus** vào phần tử nhập liệu.

```
<label for="email">Email:</label>
```

```
<br />
```

```
<input type="text" id="email" name="e_mail" />
```

66

Phần tử Form - select

```
<select name="cars">
  <option value="1">Toyota</option>
  <option value="2" selected>Honda</option>
  <option value="3">Mercedes</option>
  <option value="4">Ford</option>
</select>

<select name="cars">
  <optgroup label="Swedish Cars">
    <option value="volvo">Volvo</option>
    <option value="saab">Saab</option>
  </optgroup>
  <optgroup label="German Cars">
    <option value="mercedes">Mercedes</option>
    <option value="audi">Audi</option>
  </optgroup>
</select>
```

67

Phần tử Form - select (tt)

- **multiple**: cho phép người dùng chọn nhiều giá trị.
- **size**: xác định số lượng lựa chọn được hiển thị (mặc định là 1)

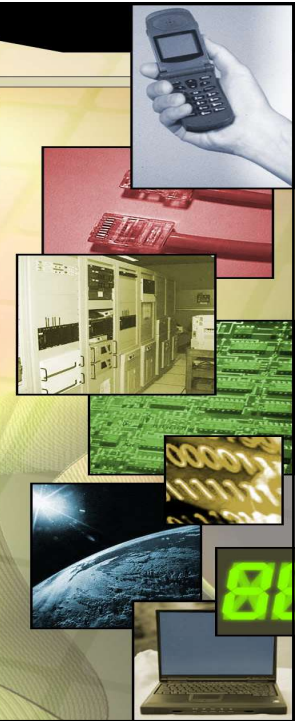
```
<select name="cars" multiple size="4">
  <option value="1">Toyota</option>
  <option value="2">Honda</option>
  <option value="3">Mercedes</option>
  <option value="4">Ford</option>
  <option value="5">Audi</option>
  <option value="6">Vinfast</option>
</select>
```

68

Phần tử Form - input

- ❖ Phần tử quan trọng nhất của form, có thể hiển thị theo nhiều cách, tùy vào giá trị của thuộc tính **type**
- ❖ Các loại input type **<input type="..." />**

text	password	radio	checkbox
button	submit	image	reset
file	hidden		
color	date	datetime-local	email
month	number	range	search
tel	time	url	week

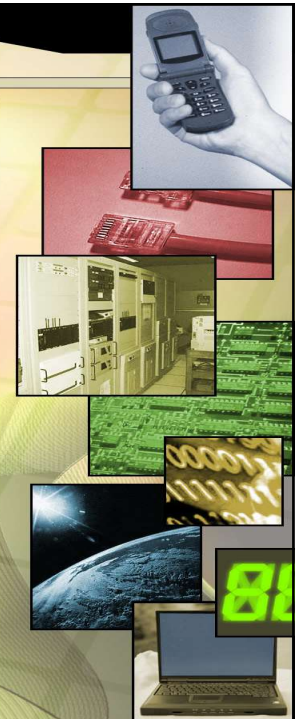


69

Phần tử Form - input (tt)

❖ Một số thuộc tính nhập liệu

- **value**: chỉ định giá trị ban đầu cho (giá trị nhập sẵn).
- **readonly**: trường đầu vào là chỉ đọc và giá trị sẽ được gửi khi gửi biểu mẫu.
- **disabled**: chỉ định rằng một trường đầu vào sẽ bị vô hiệu hóa và giá trị sẽ không được gửi khi gửi biểu mẫu.
- **maxlength**: chỉ định số lượng ký tự tối đa được phép nhập vào.
- **min** và **max**: chỉ định các giá trị tối thiểu và tối đa cho một trường đầu vào.
- **placeholder**: chỉ định một gợi ý ngắn để mô tả giá trị dự kiến của trường đầu vào.
- **required**: xác định rằng trường đầu vào phải có giá trị (được nhập) trước khi gửi biểu mẫu.

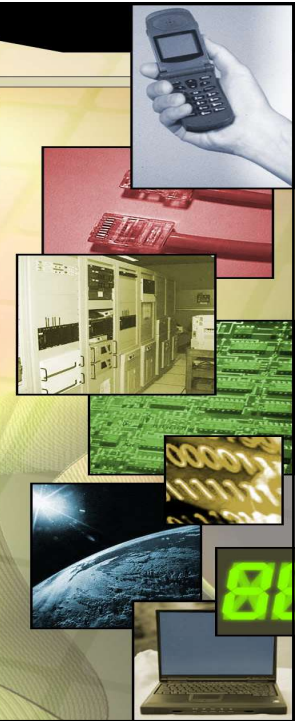


70

Phần tử Form - input (tt)

❖ Một số thuộc tính nhập liệu (tt)

- **autofocus**: tự động focus trường nhập liệu khi load trang.
- **step**: xác định bước nhảy trong các trường nhập liệu số
- **pattern**: xác định khuôn dạng dữ liệu nhập vào.
- **form**: xác định trường nhập liệu thuộc 1 form có *id xác định* (mặc dù form đó không chứa trường nhập liệu này).
- **list**: xác định giá trị trường nhập liệu là một danh sách (sử dụng thẻ `<datalist>` để tạo danh sách). Tương tự như `<select>`



71

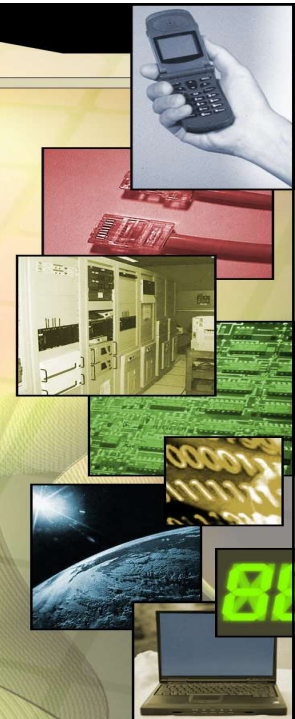
Phần tử Form - input (tt)

```
<form>
  <input type="radio" name="sex" value="male" checked /> Male
  <br>
  <input type="radio" name="sex" value="female" /> Female
</form>
```

☒ Male
☐ Female

```
<form>
  <input type="checkbox" name="bike" checked /> I have a bike
  <br>
  <input type="checkbox" name="car" /> I have a car
</form>
```

☒ I have a bike
☐ I have a car



72

Phần tử Form - input (tt)

```
<label for="country_code">Country code:</label>
<input type="text" id="country_code" name="country_code"
pattern="[A-Za-z]{3}" title="Three letter country code" />
```

Country code:

Three letter country code

```
<label for="browser">Choose browser:</label>
<input list="browsers" name="browser" id="browser" />
<datalist id="browsers">
  <option value="Edge"></option>
  <option value="Firefox"></option>
  <option value="Chrome"></option>
  <option value="Safari"></option>
</datalist>
<input type="submit" />
```

Choose browser: Submit

- Edge
- Firefox
- Chrome
- Safari

73

Phần tử Form - input (tt)

```
<form action="action_page.php" id="this_form">
  <label for="num">Input number (between 0 and 50): </label>
  <input type="number" id="num" name="num" min="0" max="50" step="5"
value="20" title="Input number" required />
  <input type="submit">
</form>
<label for="name">Input name: </label>
<input type="text" id="name" name="name"
form="this_form" maxlength="10" autofocus />
<br />
<label for="color">Input color: </label>
<input type="color" id="color"
name="color" form="this_form" />
```

Input number (between 0 and 50): 20 Submit

Input name: ABCDEFGHIJ

Input color:

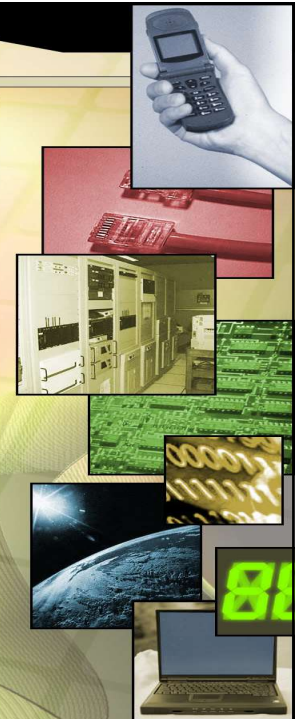
106 43 222
R G B

74

Thẻ (tag) đặc biệt - <meta .../>

- ❖ Cung cấp thêm "thông tin về trang web" cho trình duyệt và các công cụ tìm kiếm.
- ❖ Đặt trong thẻ <head>

```
<head>
  <meta charset="UTF-8" />
  <meta name="description" content="Free Web tutorials" />
  <meta name="keywords" content="HTML,CSS,XML,JavaScript" />
  <meta name="author" content="Hoàng Sơn" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <!-- Thiết lập khung nhìn cho trang web trên tất cả các thiết bị. -->
  <meta http-equiv="refresh" content="10" />
  <meta http-equiv="refresh" content="30; url=https://ntu.edu.vn" />
</head>
...
```



75

Thẻ (tag) đặc biệt - <script>...</script>

- ❖ Bổ sung code kịch bản phía máy khách, mặc định là JavaScript

❖ Cú pháp

- Nhúng code script vào trang web

```
<script>
  document.write("Hello world!");
</script>
```

- Liên kết đến file script bên ngoài

```
<script src="js/myScript.js"></script>
```

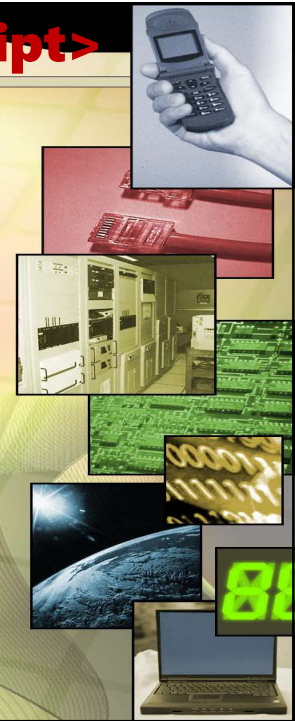


76

Thẻ (tag) đặc biệt - `<script>...</script>`

❖ Thuộc tính

- **src** : chỉ định đường dẫn đến file javascript bên ngoài.
- **async** : xác định code kịch bản được thực hiện **không đồng bộ**, được sử dụng khi script này là **độc lập, không phụ thuộc** vào các script khác (**tùy chọn**)
- **defer** : xác định script chỉ được thực hiện sau khi tài liệu HTML đã được phân tích cú pháp.

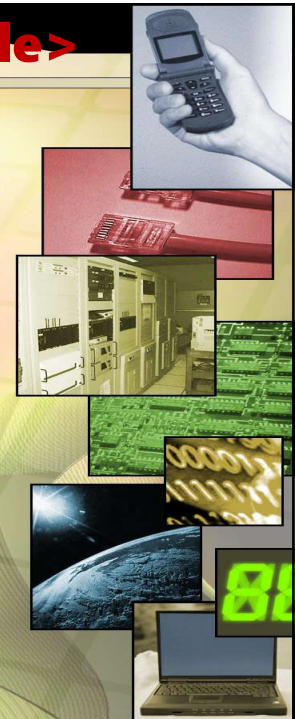


77

Thẻ (tag) đặc biệt - `<style>...</style>`

- ❖ Bổ sung code định dạng (CSS) cho các phần tử HTML trong nội bộ trang web.
- ❖ Bắt buộc phải ở trong thẻ `<head>`

```
<head>
  <style>
    div, #myId {
      background-color: gray;
      color: #0000FF;
    }
    .myclass { color: red;}
  </style>
</head>
```



78

Thẻ (tag) đặc biệt - <link ... />

❖ Xác định mối quan hệ giữa tài liệu HTML và các tài nguyên bên ngoài, được dùng phổ biến nhất là kết nối tới tài liệu css để bổ sung code định dạng cho các phần tử HTML.

❖ Bắt buộc phải ở trong thẻ <head>

```
<head>
```

```
  <link rel="stylesheet" href="theme.css" />
```

```
</head>
```

