**FRONTEND**

**I. HTML (Hyper Text Markup Language):**

***1. HTML là gì?***

- HTML là chữ viết tắt của Hypertext Markup Language, là ngôn ngữ được sử dụng rộng rãi nhất để viết các trang Web.

- Hypertext là cách mà các trang Web (các tài liệu HTML) được kết nối với nhau. Và như thế, đường link có trên trang Web được gọi là Hypertext.

- Như tên gọi đã gợi ý, HTML là ngôn ngữ đánh dấu bằng thẻ (Markup Language), nghĩa là bạn sử dụng HTML để đánh dấu một tài liệu text bằng các thẻ (tag) để nói cho trình duyệt Web cách để cấu trúc nó để hiển thị ra màn hình.

\* Cấu trúc cơ bản của một tài liệu HTML:

<!DOCTYPE html>

<html>

<head>

<title>Day la tieu de</title>

</head>

<body>

<h1>Day la dau de</h1>

<p>Phan noi dung cua tai lieu ...</p>

</body>

</html>

***2. Các thẻ cơ bản của HTML:***

***\*Các cặp thẻ mở và thẻ đóng tạo thành các phần tử HTML.***

- <!DOCTYPE html>: cần ở đầu mỗi tài liệu HTML, đánh dấu cho trình duyệt biết nó đang thực hiện đọc file HTML, đây không hẳn là một thẻ HTML.

- <html>…</html>: nói với trình duyệt rằng nó đang đọc file HTML, cần cả hai thẻ trên.

- <head>: Đối với các trang cơ bản, thẻ head sẽ chứa tiêu đề.

- <title>: Thẻ này xác định tiêu đề của trang, đặt tên vào thẻ này và đóng nó, đây là tiêu đề tab khi nó được hiển thị khi mở trong một trình duyệt.

<head><title> Tiêu đề </title> </head>

- <meta>: thẻ này được đặt trong khu vực tiêu đề của trang, được sử dụng bởi các công cụ tìm kiếm, có nhiều trường meta khác nhau, một số trường cơ bản như sau:

\* cú pháp: <meta name = “field\_name” content = “field\_value”>

+ description: mô tả cơ bản của trang.

<meta name="description" content="A basic HTML tutorial">

+ keywords: lựa chọn từ khoá cho trang.

<meta name="keywords" content="HTML,code,tags">

+ author: tác giả của trang.

<meta name="author" content="Nguyen Van Dong">

+ viewport: một thẻ đảm bảo rằng trang này có thể hiển thị trên tất cả thiết bị. Thẻ này sẽ luôn có: “width=device-width, initial-scale=1.0” để đảm bảo nội dung trang hiển thị tốt trên các thiết bị di động và máy tính để bàn.

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

- <body>: mọi thứ của trang web đều được đặt trong thẻ này ngoại trừ tiêu đề.

<body> Everything you want displayed on your page. </body>

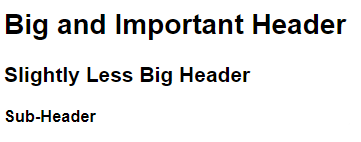
- <h1>: xác định tiêu đề cấp một trên trang, thẻ <h2> xác định tiêu đề cấp 2, ...

<h1>Big and Important Header</h1>

<h2>Slightly Less Big Header</h2>

<h3>Sub-Header</h3>

=> Kết quả:



- <p>: Thẻ đoạn văn bắt đầu một đoạn mới, thường bao gồm 2 dấu ngắt dòng.

<p>Your first paragraph.</p>

<p>Your second paragraph.</p>

\*Có thể sử dụng các kiểu CSS trong các đoạn văn như sau:

<p style="font-size: 120%;">20% larger text</p>

- <br>: Thẻ ngắt dòng chèn một đoạn ngắt dòng đơn.

<p>The first line.<br> The second line (close to the first one).</p>

- <strong>: Thẻ này xác định các đoạn văn bản quan trọng, dùng để in đậm văn bản, có thể sử dụng thẻ <b> cho chữ in đậm.

<strong>Very important things you want to say.</strong>

- <em>: Thẻ này xác định đoạn văn bản được in nghiêng, giống với thẻ <i>

<em>An emphasized line.</em>

- <href>: Thuộc tính xác định đích đến của liên kết, trong nhiều trường hợp, đây sẽ là một trang web khác, cũng có thể là một tập tin, hình ảnh, tài liệu, ... Các thuộc tính bao gồm: target và title, thuộc tính target dùng để mở liên kết trong một tab hoặc cửa sổ mới.

<a href="//www.quantrimang.com/>Go to Quantrimang</a>

<a href="//www.quantrimang.com/" target="blank">Go to Quantrimang in a new tab</a>

- <img>: Thẻ này dùng để nhúng hình ảnh vào tranh web, có thể thêm nguồn để hiển thị hình ảnh như sau:

<img src="/photos/image/2017/10/23/code-html-3.jpg">

Có thể thêm các thuộc tính width và height để đặt kích thước hình ảnh, nếu chỉ quy định một chiều, chiều còn lại sẽ được scale theo còn nếu quy định cả hai chiều thì hình ảnh sẽ bị méo mó ☺

<img src="/photos/image/2017/10/23/code-html-3.jpg" width="320" alt="the name of your image">

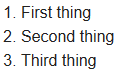
Thẻ alt cho biết văn bản nào sẽ được hiển thị nếu trình duyệt kết nối chậm, cũ, họ sẽ thấy nội dung text trên trang của bạn.

ư

- <ol>: Cho phép tạo một danh sách có thứ tự, nhìn chung là tạo được một danh sách có đánh số, mỗi mục trong danh sách được đánh dấu là bằng thẻ danh mục: <li>

<ol> <li>First thing</li> <li>Second thing</li> <li>Third thing</li> </ol>

=> kết quả:



Trong HTML5, có thể sử dụng, thẻ <ol reversed> để đảo ngược thứ tự các con số trong bảng, thuộc tính type dùng để quy định loại ký tự nào sẽ được dùng trong danh sách, có thể là “1”, “A”, “I” hoặc “i”.

- <ul>: Cho phép tạo một danh sách không có số thứ tự.

<ul> <li>First item</li> <li>Second item</li> <li>Third item</li> </ul>

Thuộc tính type lúc này có thể đặt thành disc, circle hoặc square.

- <table>: thẻ này dùng để phân đoạn thông tin trên trang.

<table> <tbody> <tr> <th>First column</th> <th>Second column</th> </tr> <tr> <td>Row 1, column 1</td> <td>Row 1, column 2</td> </tr> <td>Row 2, column 1</td> <td>Row 2, column 2</td> </tbody> </table>

Mỗi thẻ <th> bao gồm một tiếu đề, đôi khi chúng được đặt trong thẻ <thead>, thẻ <tbody> dùng để tạo ra phần thân của bảng, thẻ <tr> tạo ra một hàng mới, và thẻ <td> xác định một ô trong hàng đó.

- <blockquote>: Khi trích dẫn một lời nói thì nên sử dụng thẻ này để tạo nên sự khác biệt.

<blockquote>I must not fear. Fear is the mind-killer. Fear is the little-death that brings total obliteration. I will face my fear. I will permit it to pass over me and through me. And when it has gone past I will turn the inner eye to see its path. Where the fear has gone there will be nothing. Only I will remain.</blockquote>

- <button>: tạo nút bấm trên trang.

<button>Click vào đây</button>

- <center> <p> ... </p></center>: căn giữa văn bản.

- <hr />: tạo một dòng ngang từ vị trí hiện tại đến lề phải => tạo một dòng ngắt, nếu không có khoảng trắng và dấu /, sẽ không có hiệu lực trong XHTML.

- <pre>: giữ nguyên định dạng trong tài liệu, các dữ liệu nằm trong <pre> </pre> sẽ giữ nguyên định dạng trong tài liệu.

- <u>: thẻ này dùng để hiển thị nội dung trong thẻ kèm theo dấu gạch dưới.

the u trong HTML.

- <strike>: thẻ này dùng để hiển thị nội dung trong thẻ kém theo dấu gạch ngang.  ~~the strike trong HTML~~.

- <tt>: nội dung trong thẻ này được hiển thị dưới dạng font đơn cách (độ dài mọi ký tự là như nhau).

- <sup>: nội dung trong thẻ được hiển thị dưới dạng chỉ số trên.

- <sub>: nội dung trong thẻ được hiển thị dưới dạng chỉ số dưới.

- <ins>: nội dung trong thẻ được hiển thị dưới dạng bị chèn.

- <del>: nội dung trong thẻ được hiển thị dưới dạng bị xoá.

- <big> và <small>: nội dung trong thẻ được hiển thị to/nhỏ hơn so với xung quanh.

- <div>: chia các phần tử thành nhóm, có thể đính kèm style cho phần tử này.

<!DOCTYPE html>

<html>

<head>

<title>Vi du the div trong HTML</title>

</head>

<body>

<div id="menu" align="middle" >

<a href="../index.jsp">HOME</a> |

<a href="../lien-he/">CONTACT</a> |

<a href="../index.jsp">ABOUT</a>

</div>

<div id="content" align="left" bgcolor="white">

<h5>Vi du the div</h5>

<p>Hoc HTML tai VietJack.....</p>

</div>

</body>

</html>

=> Kết quả:

/about/contact\_us.jsp

[HOME](https://vietjack.com/index.jsp) | [CONTACT](https://vietjack.com/lien-he/) | [ABOUT](https://vietjack.com/index.jsp)

**Vi du the div**

Hoc HTML tai VietJack....

- <span>: chia nhóm các phần tử trên một dòng, dùng để phân nhóm các từ trong câu.

<!DOCTYPE html>

<html>

<head>

<title>Vi du the span</title>

</head>

<body>

<p>Day la vi du ve the <span style="color:green">span</span> va the <span style="color:red">div</span> co su dung CSS</p>

</body>

</html>

=> Kết quả: Day la vi du ve the span va the div co su dung CSS

- <mark>: nội dung trong thẻ được đánh dấu bằng mực vàng.

<!DOCTYPE html>

<html>

<head>

<title>Danh dau van ban trong HTML</title>

</head>

<body>

<p>Cum tu sau duoc danh dau bang <mark>the mark trong HTML</mark> voi mau vang</p>

</body>

</html>

=> Kết quả: Cum tu sau duoc danh dau bang the mark trong HTML voi mau vang

- <abbr>: cung cấp thêm thông tin hữu ích trong trình duyệt, thuộc tính title trong thẻ sử dụng để lưu ý nghĩa đầy đủ của nội dung này.

<!DOCTYPE html>

<html>

<head>

<title>The abbr trong HTML</title>

</head>

<body>

<p>Toi thich hoc <abbr title="Ngon ngu lap trinh Java">Java</abbr>.</p>

</body>

</html>

=> Kết quả (không được giống lắm): Toi thich hoc Java.

- <acronym>: tên lược danh, mà trình duyệt không xử lý gì hết ☺

- <bdo>: quyết định phương hướng của văn bản hiện tại.

<!DOCTYPE html>

<html>

<head>

<title>Vi du the bdo trong HTML</title>

</head>

<body>

<p>Phan van ban se hien thi tu trai sang phai.</p>

<p><bdo dir="rtl">Phan van ban se hien thi tu phai qua trai.</bdo></p>

</body>

</html>

=> Kết quả:

Phan van ban se hien thi tu trai sang phai.

Phan van ban se hien thi tu phai qua trai.

- <dfn>: chỉ rõ ra rằng đây là một mục đặc biệt, nội dung bên trong thường được hiển thị nghiêng như <i> hoặc <em>.

- <q>: thêm nháy kép trong nội dung câu.

- <cite>: hiển thị câu trích dẫn, in nghiêng theo mặc định.

- <code>: thông thường nội dung trong thẻ này được hiển thị bằng font đơn cách.

- <kbd>: dùng để hiển thị nội dung là những gì được chỉ rõ để gõ vào văn bản.

- <var>: thường được dùng chung với thẻ <pre> để chỉ rõ rằng nội dung trong phần tử là một biến.

- <samp>: chỉ kết quả đầu ra mẫu của chương trình.

- <address>: chứa địa chỉ (nơi ở).

<!DOCTYPE html>

<html>

<head>

<title>Vi du the address trong HTML</title>

</head>

<body>

<address>So 11, Pho Le Thai To, Thu Do Ha Noi</address>

</body>

</html>

=> Kết quả: *So 11, Pho Le Thai To, Thu Do Ha Noi*

***\* Các thẻ meta trong HTML (Mở rộng phần trên):***

- Thẻ meta dùng để cung cấp các thông tin thêm, thẻ này là một phần tử trống và không có thẻ đóng, mang thông tin trong các thuộc tính. Thẻ meta là một trong các nhân tố quan trọng trong SEO. Tối ưu hoá thẻ meta giúp cho web nhanh lên top tìm kiếm.

- Các thuộc tính trong thẻ meta:

+ name: tên của thuộc tính

+ content: giá trị thuộc tính

+ scheme: giản đồ giải thích giá trị thuộc tính.

+ http-equiv: thiết lập cho header nhằm thông báo các phản hồi HTTP: ví dụ như thiết lập cookie.

- Thẻ meta keywords: dùng để chỉ rõ các từ khoá quan trọng trong quá trình tìm kiếm và từ khoá này sẽ được sử dụng trong các công cụ tìm kiếm.

- Thẻ meta description: một miêu tả ngắn gọn về tài liệu, trợ giúp các công cụ tìm kiếm trong mục lập chỉ mục trang web các công cụ cần tìm kiếm.

- Thẻ meta revised: Cung cấp thông tin ngày cập nhật cuối cùng mà tài liệu được cập nhật.

- Thẻ meta http-equiv = “refresh”: dùng để xác định thời gian mà sau đó tranh web sẽ refresh một cách tự động.

Chuyển hướng trang: xác đinh url: <meta http-equiv="refresh" content="5; url=https://vietjack.com" />

- Thẻ meta http-equiv = “cookie”: Sử dụng thẻ này để bảo quản các cookie trên Client sau đó Server có thể sử dụng thông tin này để theo dõi lượng khách truy cập.

*Cookie là dữ liệu, được lưu giữ trong các file văn bản nhỏ trên máy tính và nó được trao đổi giữa các trình duyệt Web và Sever để theo dõi các khu vực của thông tin dựa trên ứng dụng của bạn.*

*Nếu bạn không tính đến ngày hết hạn, Cookie sẽ được coi là một session (phiên) và sẽ bị xóa khi người dùng thoát trình duyệt.*

- Thẻ meta author: Sử dụng thẻ này để thiết lập tên tác giả của một trang web

- Xác định bộ ký tự: ví dụ về bộ ký tự UTF-8 <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />

***3. Thuộc tính trong HTML:***

- Thuộc tính được sử dụng để xác định đặc trưng của một phần tử HTML và được đặt trong thẻ mở của phần tử đó. Tất cả các thuộc tính được chia thành 2 thành phần là **tên** và một **giá trị.**

- Tên là tên của thuộc tính, ví dụ align, bạn có thể sử dụng thuộc tính này để căn chỉnh đoạn văn trong văn bản.

- Giá trị là giá trị cần gán cho thuộc tính ví dụ như với align thì có left, center, right.

- Tên và giá trị thuộc tính không phân biệt chữ hoa và chữ thường (nên dùng chữ thường).

- Trong phần lớn các phần tử HTML, có 4 thuộc tính cốt lõi đó là:

+ id: dùng để nhận diện duy nhất bất kỳ phần tử nào, như là một định danh duy nhất. Nếu có 2 phần tử cùng tên trong một trang web, có thể dùng id để phân biệt được chúng.

<p id="html">Vi du 1 ve thuoc tinh id trong HTML</p>

<p id="css">Vi du 2 ve thuoc tinh id trong HTML</p>

+ title: cung cấp tiêu đề cho một phần tử, chế độ xử lý tuỳ thuộc vào phần tử mang nó là gì, thường là một tooltip khi rê chuột qua.

+ class: dùng để liên kết một phần tử với một style sheet và xác định kiểu của phần tử (tìm hiểu thêm Cascadign Style Sheet – CSS). Giá trị của phần tử này là một danh sách tên các class được ngăn bởi khoảng trắng.

class="tenClass1 tenClass2 tenClass3"

+ style: quy định các quy tắc CSS trong phần tử.

<!DOCTYPE html>

<html>

<head>

<title>Vi du thuoc tinh style trong HTML</title>

</head>

<body>

<p style="font-family:arial; color:#FF0000;">Hoc HTML tai VietJack...</p>

</body>

</html>

=> Kết quả: Hoc HTML tai VietJack...

***\* Các phần tử Global trong HTML: (Các thuộc tính này nằm trong thẻ mở <html>)***

- Thuộc tính dir: cho phép thiết lập phương hướng mà văn bản hiển thị. Giá trị “ltr”: từ trái qua phải hoặc “rtl”: phải qua trái.

<!DOCTYPE html>

<html dir="rtl">

<head>

<title>Vi du thuoc tinh dir trong HTML</title>

</head>

<body>

Phan van ban nay duoc hien thi tu phai sang trai.

</body>

</html>

- lang trong HTML và xml:lang trong XHTML: chi phép chỉ rõ ngôn ngữ chính sử dụng trong một tài liệu.

<!DOCTYPE html>

<html lang="vi">

<head>

<title>Vi du thuoc tinh lang trong HTML</title>

</head>

<body>

Trang VietJack duoc viet bang tieng Viet.

</body>

</html>

***\* Các thuộc tính hay sử dụng trong HTML:***

-

|  |  |  |
| --- | --- | --- |
| align | right, left, center | Các thẻ căn chỉnh theo chiều ngang |
| valign | top, middle, bottom | Các thẻ căn chỉnh theo chiều dọc trong một phần tử HTML |
| bgcolor | Giá trị số, thập lục phân, RGB | Đặt màu nền phía sau một phần tử |
| background | URL | Đặt ảnh nền phía sau một phần tử |
| id | Người dùng tự định nghĩa | Đặt tên một phần tử để sử dụng với Cascading Style Sheets |
| class | Người dùng tự định nghĩa | Phân loại một phần tử để sử dụng với Cascading Style Sheets |
| width | Giá trị số | Xác định độ rộng của bảng, hình ảnh hoặc ô trong bảng |
| height | Giá trị số | Xác định chiều cao của bảng, hình ảnh hoặc ô trong bảng |
| title | Người dùng tự định nghĩa | "Pop-up" tiêu đề của phần tử |

***4. Comment trong HTML:***

- Comment trong HTML được đặt trong thẻ <!.. ..>:

<!DOCTYPE html>

<html>

<head> <!-- Vi du comment 1 -->

<title>Tieu de tai lieu</title>

</head> <!-- Vi du comment 2 -->

<body>

<p>Phan noi dung chinh cua tai lieu.....</p>

</body>

</html>

=> Kết quả: Phan noi dung chinh cua tai lieu…..

- Comment hợp lệ và không hợp lệ trong HTML:

+ Các comment không được lồng vào nhau.

+ Không được có dấu cách < !-- ... -->

- Comment nhiều dòng: đặt <!-- ở đầu dòng đầu tiên và --> ở dòng cuối cùng:

<!--

Vi du mot comment co nhieu dong. Phan comment nay

co the trai rong tren nhieu dong.

-->

- Comment cho script code: Nếu bạn đang sử dụng JavaSript hoặc VB Scrip trong code HTML của bạn thì nó sẽ gợi ý đặt đoạn Script code đó trong một comment HTML thích hợp để các trình duyệt cũ làm việc đúng cách.

<!DOCTYPE html><html>

<head>

<title>Comment cho Script code</title>

<script>

<!--

document.write("Hello World!")

//-->

</script>

</head>

<body>

<p>Hello , World!</p>

</body>

</html>

=> Kết quả:

Hello World!

Hello , World!

- Comment cho CSS: Nếu bạn sử dụng Casecading Style Sheet trong mã hóa HTML của bạn thì nó sẽ gợi ý đặt mã hóa Style Sheet vào trong một comment thích hợp để các trình duyệt cũ có thể làm việc đúng cách.

<!DOCTYPE html><html>

<head>

<title>Comment cho Style Sheet</title>

<style>

<!--

.example {

border:1px solid #4a7d49;

}

//-->

</style>

</head>

<body>

<div class="example">Hello , World!</div>

</body>

</html>

=> Kết quả (có khung xanh):

***5. Hình ảnh trong HTML: (Mở rộng phần trên)***

- Cú pháp cơ bản:

<img src="đường dẫn url tới nơi chứa hình ảnh" ... danh-sách-thuộc-tính/>

- Thẻ <img> là một thẻ trống, nghĩa là nó có thể chứa danh sách thuộc tính và không có thẻ đóng.

<!DOCTYPE html>

<html>

<head>

<title>Hinh anh trong HTML</title>

</head>

<body>

<p>Vi du the img trong HTML.</p>

<img src="./images/logo3.png" alt="Hình ảnh trong HTML" />

</body>

</html>

=> Kết quả: (lấy kết quả của page ngta)



- Phải gõ tên file ảnh trong thuộc tính src, có phân biệt chữ hoa chữ thường.

- Thuộc tính alt là một thuộc tính ủy quyền, nếu hình ảnh không hiển thị, nó sẽ xác định một văn bản thay thế cho hình ảnh đó.

- Thiết lập chiều cao bằng thuộc tính height, chiều rộng bằng width, căn chỉnh bằng thuộc tính align, tạo viền bằng thuộc tính border.

***6. Bảng trong HTML (Mở rộng phần trên):***

- Bảng cho phép lập trình viên sắp xếp các thông tin dưới dạng bảng (xếp vào các ô trong bảng.

- Dùng thẻ <table> để tạo bảng, sử dụng <tr> để tạo ra hàng mới, thẻ td dùng để tạo các ô.

<!DOCTYPE html>

<html>

<head>

<title>Vi du bang trong HTML</title>

</head>

<body>

<table border="1">

<tr>

<td>Row 1, Column 1</td>

<td>Row 1, Column 2</td>

</tr>

<tr>

<td>Row 2, Column 1</td>

<td>Row 2, Column 2</td>

</tr>

</table>

</body>

</html>

Chạy đoạn mã trên sẽ tạo bảng sau:

|  |  |
| --- | --- |
| Row 1, Column 1 | Row 1, Column 2 |
| Row 2, Column 1 | Row 2, Column 2 |

- border là một thuộc tính của table, nếu không cần viền thì có thể đặt border = “0”.

*\* Tiêu đề bảng trong HTML:*

- Tiêu đề được xác định bằng thẻ <th>, thẻ này thế chỗ cho thẻ <td>, có thể sử dụng để làm tên cột, đại diện cho các ô dữ liệu. Thông thường, th được đặt ở các ô trên cùng, nhưng thực tế có thể đặt ở bất cứ ô nào trong cột.

<!DOCTYPE html>

<html>

<head>

<title>Vi du tieu de bang</title>

</head>

<body>

<table border="1">

<tr>

<th>Ten nhan vien</th>

<th>Luong</th>

</tr>

<tr>

<td>Minh Chinh</td>

<td>5000</td>

</tr>

<tr>

<td>Duy Manh</td>

<td>7000</td>

</tr>

</table>

</body>

</html>

Kết quả hiển thị là:

|  |  |
| --- | --- |
| **Ten nhan vien** | **Luong** |
| Minh Chinh | 5000 |
| Duy Manh | 7000 |

*\* Thuộc tính cellpading và cellspacing:*

- cellspacing dùng để xác định độ rộng khoảng các với border, cellpading thay cho khoảng cách giữa border của ô và nội dung trong ô.

<!DOCTYPE html>

<html>

<head>

<title>Vi du the cellpadding va cellspacing trong HTML</title>

</head>

<body>

<table border="1" cellpadding="5" cellspacing="5">

<tr>

<th>Ten nhan vien</th>

<th>Luong</th>

</tr>

<tr>

<td>Minh Chinh</td>

<td>5000</td>

</tr>

<tr>

<td>Duy Manh</td>

<td>7000</td>

</tr>

</table>

</body>

</html>

Kết quả như sau:

|  |  |
| --- | --- |
| **Ten nhan vien** | **Luong** |
| Minh Chinh | 5000 |
| Duy Manh | 7000 |

*\*Hợp nhất các hàng và cột trong bảng:*

- colspan: dùng để hợp nhất các ô khác cột.

- rowspan: dùng để hợp nhất các ô khác hàng.

- Hai thuộc tính này sẽ được để ở trong thẻ mở <td>.

<!DOCTYPE html>

<html>

<head>

<title>Vi du the colspan va rowspan trong HTML</title>

</head>

<body>

<table border="1">

<tr>

<th>Column 1</th>

<th>Column 2</th>

<th>Column 3</th>

</tr>

<tr><td rowspan="2">Row 1 Cell 1</td><td>Row 1 Cell 2</td><td>Row 1 Cell 3</td></tr>

<tr><td>Row 2 Cell 2</td><td>Row 2 Cell 3</td></tr>

<tr><td colspan="3">Row 3 Cell 1</td></tr>

</table>

</body>

</html>

Kết quả hiển thị là:

|  |  |  |
| --- | --- | --- |
| **Column 1** | **Column 2** | **Column 3** |
| Row 1 Cell 1 | Row 1 Cell 2 | Row 1 Cell 3 |
| Row 2 Cell 2 | Row 2 Cell 3 |
| Row 3 Cell 1 | | |

*\* Background cho bảng trong HTML:*

- bgcolor: thuộc tính thiết lập màu nền cho cả bảng hoặc chỉ cho một ô.

- background: thuộc tính thiết lập ảnh nền cho cả bảng hoặc chỉ cho một ô.

- bordercolor: thuộc tính thiết lập màu cho border.

<!DOCTYPE html>

<html>

<head>

<title>Vi du tao background cho bang trong HTML</title>

</head>

<body>

<table border="1" bordercolor="green" bgcolor="yellow">

<tr>

<th>Column 1</th>

<th>Column 2</th>

<th>Column 3</th>

</tr>

<tr><td rowspan="2">Row 1 Cell 1</td><td>Row 1 Cell 2</td><td>Row 1 Cell 3</td></tr>

<tr><td>Row 2 Cell 2</td><td>Row 2 Cell 3</td></tr>

<tr><td colspan="3">Row 3 Cell 1</td></tr>

</table>

</body>

</html>

Nó sẽ tạo ra kết quả sau:

|  |  |  |
| --- | --- | --- |
| **Column 1** | **Column 2** | **Column 3** |
| Row 1 Cell 1 | Row 1 Cell 2 | Row 1 Cell 3 |
| Row 2 Cell 2 | Row 2 Cell 3 |
| Row 3 Cell 1 | | |

*\*Chiều cao và độ rộng của bảng trong HTML:*

- Sử dụng thuộc tính width và height để thiết lập độ rộng và chiều cao của bảng, có thể thiết lập bằng pixel hoặc phần trăm của khu vực đặt hình.

- <!DOCTYPE html>

<html>

<head>

<title>Chieu cao va do rong bang trong HTML</title>

</head>

<body>

<table border="1" width="400" height="150">

<tr>

<td>Row 1, Column 1</td>

<td>Row 1, Column 2</td>

</tr>

<tr>

<td>Row 2, Column 1</td>

<td>Row 2, Column 2</td>

</tr>

</table>

</body>

</html>

Nó sẽ tạo ra kết quả sau:

|  |  |
| --- | --- |
| Row 1, Column 1 | Row 1, Column 2 |
| Row 2, Column 1 | Row 2, Column 2 |

*\* Phụ đề bảng trong HTML:*

- Sử dụng thẻ caption để thiết lập tiêu đề cho bảng hoặc lời giải thích cho bảng được đặt ở đầu bảng.

<!DOCTYPE html>

<html>

<head>

<title>Vi du phu de bang trong HTML</title>

</head>

<body>

<table border="1" width="100%">

<caption>Vi du phu de bang</caption>

<tr>

<td>row 1, column 1</td><td>row 1, columnn 2</td>

</tr>

<tr>

<td>row 2, column 1</td><td>row 2, columnn 2</td>

</tr>

</table>

</body>

</html>

Đoạn mã này sẽ tạo ra kết quả sau:

|  |  |
| --- | --- |
| Vi du phu de bang | |
| row 1, column 1 | row 1, columnn 2 |
| row 2, column 1 | row 2, columnn 2 |

*\*Phần đầu, thân và chân bảng:*

- Bảng chia làm 3 phần, sử dụng các thẻ để xác định:

+ thead: tạo một phần tử tiêu đề bảng riêng biệt.

+ tbody: Nội dung chính của bảng.

+ tfoot: Tạo một phần chân bảng riêng biệt.

- Có thể có nhiều phần tử tbody nhưng các thẻ này cần phải nằm sau phần thead và tfoot.

<!DOCTYPE html>

<html>

<head>

<title>Vi du bang trong HTML</title>

</head>

<body>

<table border="1" width="100%">

<thead>

<tr>

<td colspan="4">Day la phan tieu de bang</td>

</tr>

</thead>

<tfoot>

<tr>

<td colspan="4">Day la phan chan bang</td>

</tr>

</tfoot>

<tbody>

<tr>

<td>Cell 1</td>

<td>Cell 2</td>

<td>Cell 3</td>

<td>Cell 4</td>

</tr>

</tbody>

</table>

</body>

</html>

Kết quả là:

| Day la phan tieu de bang | | | |
| --- | --- | --- | --- |
| Day la phan chan bang | | | |
| Cell 1 | Cell 2 | Cell 3 | Cell 4 |

*\*Lồng bảng trong HTML:*

- Có thể lồng một bảng vào trong một bảng khác, trong thẻ td có thể sử dụng hầu hết các thẻ.

<!DOCTYPE html>

<html>

<head>

<title>Vi du bang trong HTML</title>

</head>

<body>

<table border="1" width="100%">

<tr>

<td>

<table border="1" width="100%">

<tr>

<th>Ten nhan vien</th>

<th>Luong</th>

</tr>

<tr>

<td>Minh Chinh</td>

<td>5000</td>

</tr>

<tr>

<td>Duy Manh</td>

<td>7000</td>

</tr>

</table>

</td>

</tr>

</table>

</body>

</html>

Nó sẽ cho kết quả như sau:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Ten nhan vien** | **Luong** | | Minh Chinh | 5000 | | Duy Manh | 7000 | |

***7. Danh sách trong HTML (Mở rộng phần trên):***

- Trong HTML gồm có 3 loại danh sách, đi kèm với các thẻ sau:

+ ul: một danh sách không có thứ tự, tuy nhiên sẽ có các bullet ở đầu, thường mặc định là chấm tròn.

+ ol: một danh sách có thứ tự, thường mặc định là 1. 2. 3.

+ dl: danh sách tự định nghĩa, nhìn chung là các sắp xếp các bullet hay số thứ tự có thể thay bằng các ký tự tuỳ thích.

*\*Danh sách không thứ tự: (sử dụng thẻ ul)*

- Mặc định thì bullet là chấm tròn (disc), sử dụng thuộc tính type để thiết lập các loại bullet này trong thẻ ul, các giá trị có thể của type là:

+ disc: chấm tròn (hình tròn đặc).

+ circle: viền hình tròn (hình tròn rỗng).

+ square: hình vuông đặc.

*\*Danh sách có thứ tự: (sử dụng thẻ ol)*

- Mặc định là 1. 2. 3, sử dụng thuộc tính type để thiết lập các “loại thứ tự”, các loại này nằm trong thẻ ol, các giá trị có thể của type là:

+ 1: Hiển thị theo thứ tự 1. 2. 3.

+ i: Hiển thị theo thứ tự các chữ số La Mã bằng chữ thường.

+ I: Hiển thị theo thứ tự các chữ số La Mã bằng chữ hoa.

+ a: Hiển thị theo thứ tự a. b. c.

<!DOCTYPE html>

<html>

<head>

<title>Vi du danh sach da qua sap xep</title>

</head>

<body>

<ol>

<li>Beetroot</li>

<li>Ginger</li>

<li>Potato</li>

<li>Radish</li>

</ol>

</body>

</html>

B. C.

*\*Danh sách tự định nghĩa: (sử dụng thẻ dl) => ít xài nên bỏ qua)*

***8. Text Link trong HTML (Mở rộng phần trên):***

- Thẻ a trong HTML:

Cú pháp cơ bản:

<a href="link.com" target="\_self">Click</a>

=> Lúc này sẽ hiển thị một từ “Click” với link là link.com khi được click vào.

*\* Thuộc tính target: dùng để xác định vị trí nơi đường link được mở*

- Các giá trị có thể của target:

+ \_blank: mở trong một tab mới.

+ \_self: mở trong tab chứa liên kết.

+ \_parent: mở cửa sổ trong khung nguồn chứa đường link. //chưa hiểu lắm (really không hiểu)

+ \_top: mở trang liên kết trong cửa sổ toàn màn hình. //chưa hiểu luôn.

+ targetframe: mở trang liên kết trong targetframe ☺ // cái này còn có thể nghĩ là mình sẽ hiểu nó trong tương lai.

*\*Base link trong HTML:*

- Website đoạn mã bên dưới có base link là "https:///example.com/public/"

- Trong trang web có ba liên kết:

+ Liên kết 1 là đường dẫn tương đối, nên nó được kết hợp với base link để tạo thành đường dẫn tuyệt đối https://example.com/public/default.html

+ Liên kết 2 là đường dẫn tương đối, nên nó được kết hợp với base link để tạo thành đường dẫn tuyệt đối https://example.com/public/file/flower.png

+ Liên kết 3 là đường dẫn tuyệt đối nên nó được giữ nguyên.

<!DOCTYPE html>

<html>

<head>

<base href="https:///example.com/public/">

</head>

<body>

<a href="default.html" target="\_blank">Liên kết 1</a>

<hr>

<a href="file/flower.png" target="\_blank">Liên kết 2</a>

<hr>

<a href="http://webcoban.vn/javascript/default.html" target="\_blank">Liên kết 3</a>

</body>

</html>

- Ngoài ra thẻ base còn dùng để xác định kiểu mở link (thiết lập thuộc tính target mặc định) chung cho các link khác nếu chúng không override lại:

<!DOCTYPE html>

<html>

<head>

<base target="\_blank">

</head>

<body>

<a href="http://webcoban.vn/html/default.html">Liên kết 1</a>

<hr>

<a href="http://webcoban.vn/css/default.html">Liên kết 2</a>

<hr>

<a href="http://webcoban.vn/javascript/default.html" target="\_self">Liên kết 3</a>

</body>

</html>

*\*Liên kết tới một khu vực trong trang:*

- Đặt tên khu vực muốn liên kết tới trong trang sử dụng thẻ name.

<h1>Vi du text link trong HTML <a name="top"></a></h1>

- Sau đó tạo liên kết tới khu vực này.

<a href="../html/text\_link\_trong\_html.jsp#top">Ve dau trang</a>

*\*Thiết lập màu cho liên kết:*

- Sử dụng các thuộc tính alink, link, vlink trong thẻ body để thiết lập màu cho active link, link và visited link.

*\*Có thể tạo ra một liên kết tới các file pdf, doc, zip để khi người dùng bấm vào đó thì tài liệu được tải, chỉ cần cung cấp đầy đủ URL của file đó.*

***9. Image Link trong HTML (Có thể gộp với mục Image):***

- Sử dụng một hình ảnh bên trong một Hyperlink để tạo một image link:

<!DOCTYPE html>

<html>

<head>

<title>Vi du image link trong HTML</title>

</head>

<body>

<p>Click vao link sau:</p>

<a href="https://vietjack.com" target="\_self">

<img src="./images/logo3.png" alt="Vietjack Homepage" border="0"/>

</a>

</body>

</html>

=> Khi bấm vào hình ảnh sẽ dẫn tới link mà ảnh đó chứa như một hyperlink.

*\*Hình ảnh nhạy chuột trên HTML:*

- HTML và XHTML cho phép nhúng nhiều link khác nhau trong cùng một hình ảnh đơn (các link khác nhau nằm trên các toạ độ khác nhau của hình ảnh). Những hình ảnh như vậy được gọi là Image Map.

- Image Map trên Server: (Tìm hiểu thêm xíu nữa nha)

+ Sử dụng thuộc tính ismap để tạo các hình ảnh đặc biệt. Khi người dùng nhấn vào 1 địa điểm trong ảnh thì trình duyệt truyền song song toạ độ của con trỏ và URL trong thẻ <a> tới Server. Server thông qua toạ độ con trỏ xác định tài liệu nào sẽ được trả về cho trình duyệt. Toạ độ được xác định là góc trên bên trái (0,0) và được trả về với một dấu ? đằng trước cách biệt với URL.

- Image Map trên Client:

+ Image Map trên client được cho phép bởi thuộc tính usermap trong thẻ <img /> và được định nghĩa mở rộng bởi các thẻ <map> và <area>. Giá trị của thuộc tính usemap dùng để liên kết thẻ <map> và thẻ <img>. Thẻ <map> và area được dùng để xác định các toạ độ ảnh và link tương ứng.

+ Thẻ area trong thẻ map được dùng để xác định các khu vực toạ độ và ranh giới các khu vực nhấp chuột.

<!DOCTYPE html>

<html>

<head>

<title>Vi du USEMAP trong HTML</title>

</head>

<body>

<p>Tim kiem va click vao mot vi tri tren hinh anh sau</p>

<img src="./images/logo3.png" alt="HTML Map" border="0" usemap="#html"/>

<!-- Tao Image Map -->

<map name="html">

<area shape="circle"

coords="80,80,20" href="../css/index.jsp" alt="CSS Link" target="\_self" />

<area shape="rect"

coords="5,5,40,40" alt="jQuery Link" href="../jquery/index.jsp" target="\_self" />

</map>

</body>

</html>

***10. Email link trong HTML:***

- Thẻ <a> cung cấp một tính năng gửi mail bằng cách chọn một địa chỉ và gửi mail. Cú pháp cơ bản như sau:

<a href= "mailto:abc@example.com">Send Email</a>

=> Kết quả: [Send Email](mailto:abc@example.com)

(Bấm vào sẽ redirect tới phần mềm Email)

*\*Thiết lập mặc định cho email trong HTML:*

- Có thể thiết lập sẵn email subject và email body trong href theo cú pháp như sau:

<a href="mailto:abc@example.com?subject=Feedback&body=Message">

Send Feedback

</a>

=> Kết quả: 

***11. Frame trong HTML:***

- Frame dùng để phân chia trang web thành các khu vực riêng để chứa các tài liệu HTML riêng.

- Hạn chế của Frame trong HTML:

+ Các màn hình nhỏ không thể giải quyết được các frame này vì màn hình không đủ lớn để chia thành các frame.

+ Đôi khi trang web hiển thị khác nhau trên các trang khác nhau tuỳ loại màn hình.

+ Nút quay lại trang trước có thể không làm việc như người dùng mong muốn.

+ Một vài trình duyệt không hỗ trợ sử dụng frame.

- Để sử dụng frame trong HTML, sử dụng thẻ <frameset> thay cho thẻ body. Thẻ này xác định cách chia các frame. Mỗi frame sẽ được chỉ dẫn bởi một thẻ frame và nó xác định tài liệu nào sẽ được mở trong frame này.

<!DOCTYPE html>

<html>

<head>

<title>Ví dụ tao Frame trong HTML</title>

</head>

<frameset rows="10%,80%,10%">

<frame name="top" src="../html/top\_frame.htm" />

<frame name="main" src="../html/main\_frame.htm" />

<frame name="bottom" src="../html/bottom\_frame.htm" />

<noframes>

<body>

Trình duyet cua ban khong ho tro Frame.

</body>

</noframes>

</frameset>

</html>

Cần một số file khác để hiển thị kết quả mong muốn, ở đây chỉ để như vậy mà thôi.

=> Kết quả:



Thay thuộc tính rows của đoạn code trên thành columns và thay đổi một số giá trị, ta được kết quả như sau:

<!DOCTYPE html>

<html>

<head>

<title>Vi du Frame trong HTML</title>

</head>

<frameset cols="25%,50%,25%">

<frame name="left" src="../html/top\_frame.htm" />

<frame name="center" src="../html/main\_frame.htm" />

<frame name="right" src="../html/bottom\_frame.htm" />

<noframes>

<body>

Trinh duyet cua ban khong ho tro Frame.

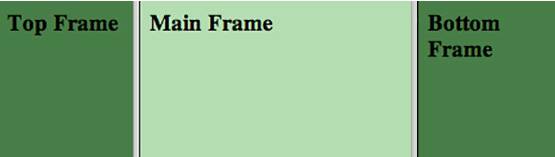
</body>

</noframes>

</frameset>

</html>

=> Kết quả:



*\*Một số thuộc tính của thẻ <frameset>:*

|  |  |
| --- | --- |
| cols | Xác định số cột được chứa trong frameset và kích cỡ của mỗi cột. Bạn có thể xác định độ rộng của mỗi cột theo một trong 4 cách sau:   * Theo giá trị tuyệt đối Pixels. Ví dụ để tạo ra 3 cột dọc, sử dụng *cols="100, 500,100"*. * Phần trăm của cửa sổ trình duyệt. Ví dụ để tạo ra thiết lập 3 Frame dọc: *cols="10%, 80%,10%"*. * Sử dụng biểu tượng gán giá trị (wildcard). Ví dụ để tạo 3 Frame dọc, bạn sử dụng *cols="10%, \*,10%"*. Trong trường hợp này thì wildcard sẽ chiếm phần còn lại. * Độ rộng quan hệ của cửa sổ trình duyệt. Ví dụ để tạo 3 Frame dọc, sử dụng *cols="3\*,2\*,1\*"*. Đây là cách thay thế cho việc sử dụng phần trăm. Ở trường hợp này, cửa sổ windows sẽ được chia thành 6 phần, cột đầu sẽ chiếm một nửa màn hình, cột thứ hai chiếm 1/3 và cột thứ ba chiếm 1/6. |
| rows | Thuộc tính này giống như thuộc tính cols và sử dụng các giá trị tương tự, nhưng nó được sử dụng để xác định các hàng của frameset. Ví dụ để tạo 2 Frame ngang, bạn sử dụng *rows="10%, 90%"*. Bạn có thể xác định chiều cao của mỗi hàng theo cách tương tự như các cột. |
| border | Thuộc tính này xác định độ rộng của Border (đường viền) của mỗi Frame theo Pixel. Ví dụ như border="5". Một giá trị bằng không có nghĩa là không có Border. |
| frameborder | Thuộc tính này xác định có hay không Border được hiển thị giữa các Frame. Nó nhận giá trị 1 hoặc 0. Ví dụ frameborder="0" thì xác định là không có Border. |
| framespacing | Xác định số lượng khoảng cách giữa các Frame trong frameset. Nó nhận giá trị nguyên. Ví dụ framespcing="10" nghĩa là có 10 pixel khoảng cách giữa các Frame. |

*\*Các thuộc tính của thẻ <frame>:*

|  |  |
| --- | --- |
| src | Thuộc tính này được sử dụng để đặt tên của file sẽ được tải trong Frame. Giá trị là bất cứ địa chỉ URL. Ví dụ, src="../html/top\_frame.jsp" sẽ tải một file HTML có trong danh mục HTML. |
| name | Thuộc tính này cho phép bạn đặt tên cho một Frame. Nó được sử dụng để chỉ rõ Frame nào của tài liệu sẽ được tải trong đó. Nó đặc biệt quan trọng khi bạn muốn tạo các link trong một Frame mà tải các trang web trong một Frame khác, trong trường hợp đó, Frame thứ hai cần một tên để xác định chính nó như là đích tới của link. |
| frameborder | Thuộc tính xác định có hay không Border trong Frame đã được chỉ; thuộc tính này được quyền ưu tiên trước giá trị được gán trong thuộc tính **frameborder** của thẻ <frameset>. Và nó nhận giá trị 1 (Yes) hoặc 0 (No). |
| marginwidth | Thuộc tính này cho phép bạn xác định độ rộng của khoảng cách giữa bên trái và phải của Border Frame với nội dung trong Frame. Giá trị là pixel. Ví dụ marginwidth="10". |
| marginheight | Thuộc tính cho phép bạn xác định chiều cao của khoảng cách giữa trên và dưới của Border Frame và nội dung của nó. Giá trị là pixel. Ví dụ marginheight="10". |
| noresize | Theo mặc định thì bạn có thể đặt lại kích thước cho Frame bằng cách nhấn chuột vào Border Frame và kéo. Thuộc tính noresize ngăn cản người sử dụng đặt lại kích cỡ. Ví dụ noresize="noresize". |
| scrolling | Thuộc tính này điều khiển sự xuất hiện của thanh cuốn xuất hiện trên Frame. Nó nhận giá trị "yes", "no", hoặc "auto". Ví dụ scrolling="no" nghĩa là sẽ không có thanh cuốn (scrollbar). |
| longdesc | Thuộc tính này cho phép bạn cung cấp một đường link tới trang web khác chứa một sự miêu tả dài trong nội dung của Frame. Ví dụ longdesc="framedescription.jsp". |

*\*Trình duyệt không hỗ trợ Frame:*

- Nếu trình duyệt không hỗ trợ frame thì nội dung trong thẻ <noframes> sẽ được hiển thị tới người dùng.

*\*Thuộc tính name và target của Frame:*

<!DOCTYPE html>

<html>

<head>

<title>Vi du Target Frame trong HTML</title>

</head>

<frameset cols="200, \*">

<frame src="../html/menu.htm" name="menu\_page" />

<frame src="../html/main.htm" name="main\_page" />

<noframes>

<body>

Trinh duyet cua ban khong ho tro Frame.

</body>

</noframes>

</frameset>

</html>

menu.html:

<!DOCTYPE html>

<html>

<body bgcolor="#4a7d49">

<a href="http://www.google.com" target="main\_page">Google</a>

<br /><br />

<a href="http://www.microsoft.com" target="main\_page">Microsoft</a>

<br /><br />

<a href="http://news.bbc.co.uk" target="main\_page">BBC News</a>

</body>

</html>

main.html:

<!DOCTYPE html>

<html>

<body bgcolor="#b5dcb3">

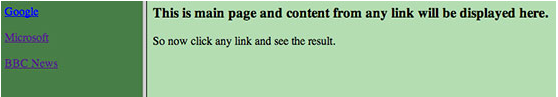
<h3>This is main page and content from any link will be displayed here.</h3>

<p>So now click any link and see the result.</p>

</body>

</html>

=> Kết quả:



- Thuộc tính của target giống với target đã trình bày trong thẻ <a>:

|  |  |
| --- | --- |
| \_self | Tải trang web trong Frame hiện tại. |
| \_blank | Tải một trang trong cửa sổ trình duyệt mới. |
| \_parent | Tải trang trong cửa sổ chính, trong trường hợp frameset đơn là cửa sổ trình duyệt chính. |
| \_top | Tải trang trong cửa sổ trình duyệt, thay cho các trang hiện tại. |
| targetframe | Tải trang trong một Frame mục tiêu đã đặt tên. |

***12. Iframe:***

- Có thể xác định một Iframe với thẻ <iframe>, thẻ này không liên quan tới thẻ <frameset>. Nó có thể xuất hiện ở bất kỳ đâu trong tài liệu của bạn. Các thẻ iframe xác định một khu vực trong trang mà tại đó trang có thể hiển thị một trang riêng biệt, bao gồm cả thanh cuốn và viền. Thẻ này dùng để nhúng một trang khác vào trang hiện tại.

- Thuộc tính src dùng để xác định địa chỉ URL của trang mà Iframe sẽ chứa nó.

<!DOCTYPE html>

<html>

<head>

<title>Vi du the Iframe trong HTML</title>

</head>

<body>

<p>Phan noi dung tai lieu...</p>

<iframe src="../html/menu.htm" width="555" height="200">

Rat tiec vi trinh duyet cua ban khong ho tro Iframe.

</iframe>

<p>Phan noi dung tai lieu...</p>

</body>

</html>

=> Kết quả:



*\*Các thuộc tính của thẻ iframe:*

|  |  |
| --- | --- |
| src | Thuộc tính này đưa ra tên file mà được tải trong Frame. Giá trị của nó có thể là bất cứ địa chỉ URL. Ví dụ src="/htm/top\_frame.jsp" sẽ tải một file HTML có trong thư mục HTML. |
| name | Thuộc tính này cho phép bạn đặt tên cho một Frame. Nó được sử dụng để chỉ rõ Frame nào của tài liệu sẽ được tải trong đó. Nó đặc biệt quan trọng khi bạn muốn tạo các link trong một Frame mà tải các trang web trong một Frame khác, trong trường hợp đó, Frame thứ hai cần một tên để xác định chính nó như là đích tới của link. |
| frameborder | Thuộc tính xác định có hay không Border trong Frame đã được chỉ; thuộc tính này được quyền ưu tiên trước giá trị được gán trong thuộc tính *frameborder* của thẻ <frameset>. Và nó nhận giá trị 1 hoặc 0. |
| marginwidth | Thuộc tính này cho phép bạn xác định độ rộng của khoảng cách giữa bên trái và phải của Border Frame với nội dung trong Frame. Giá trị là pixel. Ví dụ marginwidth="10". |
| marginheight | Thuộc tính cho phép bạn xác định chiều cao của khoảng cách giữa trên và dưới của Border Frame và nội dung của nó. Giá trị là pixel. Ví dụ marginheight="10". |
| noresize | Theo mặc định thì bạn có thể đặt lại kích thước cho Frame bằng cách nhấn chuột vào Border Frame và kéo. Thuộc tính *noresize* ngăn cản người sử dụng đặt lại kích cỡ. Ví dụ noresize="noresize". |
| scrolling | Thuộc tính này điều khiển sự xuất hiện của thanh cuốn xuất hiện trên Frame. Nó nhận giá trị "yes", "no", hoặc "auto". Ví dụ scrolling="no" nghĩa là sẽ không có thanh cuốn (scrollbar). |
| longdesc | Thuộc tính này cho phép bạn cung cấp một đường link tới trang web khác chứa một sự miêu tả dài trong nội dung của Frame. Ví dụ longdesc="framedescription.jsp". |

***13. Block trong HTML:***

- Trong các ***phần tử*** HTML được chia làm 2 loại: Các phần tử cấp khối và các phần tử nội tuyến.

- Các phần tử cấp khối thường có các thẻ: **<p>, <h1>, <h2>, <h3>, <h4>, <h5>, <h6>, <ul>, <ol>, <dl>, <pre>, <hr />, <blockquote>** và **<address>.** Chúng phải bắt đầu bằng một dòng mới và bất cứ những gì theo sau phải nằm trên các dòng này.

- Các phần tử nội tuyến là các phần tử xuất hiện trong một câu và không cần phải xuống dòng. Các phần tử này gồm các thẻ: **<b>, <i>, <u>, <em>, <strong>, <sup>, <sub>, <big>, <small>, <li>, <ins>, <del>, <code>, <cite>, <dfn>, <kbd>,** và **<var>**.

*\*Tạo nhóm các phần tử trong HTML:* Có hai thẻ quan trọng hay dùng để tạo nhóm các phần tử trong HTML là <div> và <span>:

- Thẻ <div>: là một thẻ cấp khối có vai trò rất quan trong việc tạo ra các nhóm thẻ trong HTML và sử dụng nhiều cùng với CSS. Hiện nay thì thẻ này đang được dùng để bố trí các layout trong trang web.

<!DOCTYPE html>

<html>

<head>

<title>Vi du the div trong HTML</title>

</head>

<body>

<!-- Nhom so 1 -->

<div style="color:red">

<h4>Day la nhom dau tien</h4>

<p>Danh sach cac loai rau</p>

<ul>

<li>Beetroot</li>

<li>Ginger</li>

<li>Potato</li>

<li>Radish</li>

</ul>

</div>

<!-- Nhom so 2 -->

<div style="color:green">

<h4>Day la nhom thu hai</h4>

<p>Danh sach cac loai trai cay</p>

<ul>

<li>Apple</li>

<li>Banana</li>

<li>Mango</li>

<li>Strawberry</li>

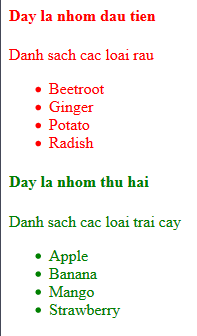
</ul>

</div>

</body>

</html>

=> Kết quả:



- Thẻ <span>: thẻ này là một thẻ nội tuyến được dùng trong tài liệu HTML, dùng để phân chia các nhóm ngay trong một câu:

<!DOCTYPE html>

<html>

<head>

<title>Vi du the span trong HTML</title>

</head>

<body>

<p>Day la <span style="color:red">mau do</span> va day la <span style="color:green">mau xanh</span></p>

</body>

</html>

=> Kết quả:

Day la mau do va day la mau xanh

***14. Thiết lập background:***

- Theo mặc định, màu nền trang web sẽ là màu trắng, có thể thay đổi điều này bằng hai cách:

+ Thay đổi màu nền.

+ Thay đổi hình nền.

- Thuộc tính bgcolor (thiết lập màu nền): Thuộc tính này được sử dụng để thay đổi màu nền của một phần tử HTML và riêng biệt cho các phần của phần thân trang (body) và bảng (table).

Cú pháp cơ bản: <tên-thẻ bgcolor="giá-trị-màu"...>

<!DOCTYPE html>

<html>

<head>

<title>Thiet lap mau nen trong HTML</title>

</head>

<body>

<!-- Dinh dang 1 - su dung ten mau -->

<table bgcolor="yellow" width="100%">

<tr><td>

Mau nen la mau vang

</td></tr>

</table>

<!-- Dinh dang 2 - Su dung gia tri hex -->

<table bgcolor="#6666FF" width="100%">

<tr><td>

Mau nen la xanh da troi

</td></tr>

</table>

<!-- Dinh dang 3 - su dung gia tri mau bang khai niem RGB -->

<table bgcolor="rgb(255,0,255)" width="100%">

<tr><td>

Mau nen la xanh la cay

</td></tr>

</table>

</body>

</html>

=> Kết quả:

|  |
| --- |
| Mau nen la mau vang |

|  |
| --- |
| Mau nen la xanh da troi |

|  |
| --- |
| Mau nen la xanh la cay |

- Thuộc tính background (thiết lập hình nền): dùng để thiết lập hình nền cho một phần tử HTML và riêng biệt cho phần thân và bảng. Cú pháp cơ bản như sau: <tên-thẻ background="đường dẫn url tới hình ảnh"...>

<!DOCTYPE html>

<html>

<head>

<title>Vi du hinh nen trong HTML</title>

</head>

<body>

<!-- Thiet lap hinh nen cho bang -->

<table background="./images/html.gif" width="100%" height="100">

<tr><td>

Bang nay se co mot hinh nen.

</td></tr>

</table>

</body>

</html>

*\*Các hình hoa và văn trong suốt: (Cái này thực hành rồi biết)*

***15. Màu sắc trong HTML:***

- Sử dụng thẻ bgcolor trong body hoặc thiết lập trong từng phần tử khác nhau trong tài liệu bằng thẻ này (đã nói ở phần 14).

- Trong thẻ body có thể sử dụng các phần tử sau để thiết lập các màu khác nhau:

+ bgcolor: Thiết lập màu cho nền của trang.

+ text: thiết lập màu cho văn bản.

+ alink: thiết lập màu cho các active link.

+ link: Thiết lập màu cho link.

+ vlink: Thiết lập màu cho visited link.

- Các cách mã hoá màu trong HTML:

+ Sử dụng tên màu:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | Black |  | Gray |  | Silver |  | White |
|  | Yellow |  | Lime |  | Aqua |  | Fuchsia |
|  | Red |  | Green |  | Blue |  | Purple |
|  | Maroon |  | Olive |  | Navy |  | Teal |

+ Sử dụng mã hexa (6 bit đại diện cho một màu).

+ Giá trị thập phân và phần trăm xác định thông qua thuộc tính rgb().

***16. Font trong HTML:***

- Sử dụng thẻ font và basefont để thiết lập font trong tài liệu.

*\*Thẻ font:*

- Thẻ font có 3 thuộc tính là: size, color và face.

- Thuộc tính size có giá trị được chấp nhận từ 1 – 7, mặc định là 3.

Font size="1"  
Font size="2"  
Font size="3"  
Font size="4"  
Font size="5"  
Font size="6"  
Font size="7"

- Thuộc tính face: (thiết lập loại font)

<!DOCTYPE html>

<html>

<head>

<title>Font Face trong HTML</title>

</head>

<body>

<font face="Times New Roman" size="5">Times New Roman</font><br />

<font face="Verdana" size="5">Verdana</font><br />

<font face="Comic sans MS" size="5">Comic Sans MS</font><br />

<font face="WildWest" size="5">WildWest</font><br />

<font face="Bedrock" size="5">Bedrock</font><br />

</body>

</html>

=> Kết quả:

Times New Roman  
Verdana  
Comic Sans MS  
WildWest  
Bedrock

- Thuộc tính color (thiết lập màu chữ):

<!DOCTYPE html>

<html>

<head>

<title>Thiet lap mau font trong HTML</title>

</head>

<body>

<font color="#FF00FF">Phan van ban nay co mau hong.</font><br />

<font color="red">Phan van ban nay co mau do.</font>

</body>

</html>

=> Kết quả:

Phan van ban nay co mau hong.  
Phan van ban nay co mau do.

*\*Phần tử basefont:*

- Phần tử <basefont> này được sử dụng để thiết lập một kích cỡ, màu, và bề mặt font mặc định cho bất kỳ phần nào trong tài liệu. Tuy nhiên, phần tử <font> có quyền ưu tiên cao hơn và sẽ ghi đè các thiết lập trong <basefont>.

<!DOCTYPE html>

<html>

<head>

<title>Thiet lap mau boi su dung basefont</title>

</head>

<body>

<basefont face="arial, verdana, sans-serif" size="2" color="#ff0000">

<p>Day la font mac dinh cua trang.</p>

<h2>Vi du phan tu &lt;basefont&gt;</h2>

<p><font size="+2" color="darkgray">

Vi du cac thiet lap boi su dung phan tu font.

</font></p>

<p><font face="courier" size="-1" color="#000000">

Vi du courier font.

</font></p>

</body>

</html>

- Day la font mac dinh cua trang.

## Vi du phan tu <basefont>

Vi du cac thiet lap boi su dung phan tu font.

Vi du courier font.

***17. Form trong HTML: (Cái này để tìm hiểu xem cái nào tối ưu hơn thì làm)***

- Thẻ **<form>** được sử dụng để tạo một Form trong HTML và thẻ này có cú pháp:

<form action="Script URL" method="GET|POST">

các phần tử font, chẳng hạn như input, textarea ...

</form>

|  |  |
| --- | --- |
| action | Ứng dụng quản trị back-end sẵn sàng để xử lý dữ liệu từ site khách. |
| method | Phương thức để tải dữ liệu lên. Thường sử dụng là GET và POST. |
| target | Xác định cửa sổ hoặc frame để hiển thị kết quả. Thuộc tính có thể nhận các giá trị như \_blank, \_self, \_parent…. |
| enctype | Bạn sử dụng thuộc tính này để xác định cách mà trình duyệt mã hóa dữ liệu trước khi nó gửi tới Server. Các giá trị có thể nhận là:   * **application/x-www-form-urlencoded** - Đây là phương thức tiêu chuẩn mà hầu hết các form sử dụng. * **mutlipart/form-data** - Nó được sử dụng khi bạn muốn tải lên dữ liệu nhị phân trong mẫu form của các file như ảnh, word…. |

- Các form trong HTML:

Text Input

Checkbox

Radio Box

Select Box

File Select Box

Submit

*\*Text Input:*

<!DOCTYPE html>

<html>

<head>

<title>Vi du Text Input trong HTML</title>

</head>

<body>

<form >

First name: <input type="text" name="first\_name" />

<br>

Last name: <input type="text" name="last\_name" />

</form>

</body>

</html>

First name:   
Last name: 

|  |  |
| --- | --- |
| type | Xác định kiểu của dữ liệu nhập vào và với Text Input thì thuộc tính này sẽ được thiết lập là **text**. |
| name | Xác định tên cho trường được gửi tới Server để xác định và nhận giá trị đó. |
| value | Xác định giá trị đầu tiên của form. |
| size | Cho phép xác định độ rộng các ký tự của Text Input. |
| maxlength | Cho phép xác định số lượng ký tự lớn nhất mà người sử dụng có thể nhập vào Text Input. |

*\*Password Input:*

Tương tự như Input đơn dòng, nhưng nó giấu các ký tự ngay sau khi người sử dụng nhập mật khẩu vào. Form này được tạo ra bằng cách sử dụng thẻ <input> và thuộc tính type được thiết lập là **password**.

Sau đây là một ví dụ đơn giản về Password Input được sử dụng để nhận mật khẩu người dùng:

<!DOCTYPE html>

<html>

<head>

<title>Ví dụ Password Input trong HTML</title>

</head>

<body>

<form >

User ID : <input type="text" name="user\_id" />

<br>

Password: <input type="password" name="password" />

</form>

</body>

</html>

User ID :     
Password: 

- Các thuộc tính:

|  |  |
| --- | --- |
| type | Xác định kiểu của dữ liệu nhập vào và với Password Input thì thuộc tính này được thiết lập là **password**. |
| name | Xác định tên trường được gửi tới Server để xác định và nhận giá trị. |
| value | Xác định giá trị dữ liệu đầu vào ban đầu của form. |
| size | Xác định độ rộng của các ký tự |
| maxlength | Xác định số lượng ký tự lớn nhất mà người sử dụng có thể nhập vào form. |

*\*Text input nhiều dòng:*

<!DOCTYPE html>

<html>

<head>

<title>Vi du Text Input đa dong</title>

</head>

<body>

<form>

Mieu ta: <br />

<textarea rows="5" cols="50" name="description">

Ban nhap phan mieu ta tai day...

</textarea>

</form>

</body>

</html>

=> Kết quả:

Mieu ta:   


-

|  |  |
| --- | --- |
| name | Cung cấp tên cho trường được gửi tới Server để xác định và nhận giá trị. |
| rows | Xác định số hàng. |
| cols | Xác định số cột. |

- ...

***18. Nhúng đa phương tiện trong HTML:***

- Sử dụng thẻ <embed> để nhúng đa phương tiện vào trang web của mình.

- Sử dụng thẻ <noembed> để hiển thị nội dung thay thế cho các trình duyệt không hỗ trợ thẻ <embed>.

<!DOCTYPE html>

<html>

<head>

<title>Thẻ embed trong HTML</title>

</head>

<body>

<embed src="../html/yourfile.mid" width="100%" height="60" >

<noembed><img src="yourimage.gif" alt="Alternative Media" ></noembed>

</embed>

</body>

</html>

Tài liệu cần được nhúng sẽ bỏ trong thẻ src.

*\*Thuộc tính của thẻ <embed>:*

|  |  |
| --- | --- |
| lign | Căn chỉnh đối tượng: **center, left** hoặc **right**. |
| autostart | Thuộc tính logic này xác định có hay không đa phương tiện nên khởi động tự động. Giá trị là *true* hoặc *false*. |
| loop | Xác định nếu âm thanh sẽ được tiếp tục chơi (thiết lập loop là true), hay chơi trong một thời gian nhất định (một giá trị xác định), hoặc không (false). |
| playcount | Xác định lượng thời gian để âm thanh được chơi. Đây là tùy chọn thay thế cho tùy chọn *loop* nếu bạn đang sử dụng IE. |
| hidden | Xác định đối tượng đa phương tiện sẽ được hiển thị trên trang. Giá trị false nghĩa là không và true là có. |
| width | Độ rộng của đối tượng thể hiện bằng pixel. |
| height | Chiều cao của đối tượng thể hiện bằng pixel. |
| name | Tên được sử dụng để tham chiếu của đối tượng. |
| src | Địa chỉ URL của đối tượng được nhúng. |
| volume | Điều khiển âm lượng của âm thanh. Giá trị từ 0 (tắt) đến 100 (âm lượng to nhất). |

- Các định dạng video hỗ trợ:

* .swf files - là các file được tạo bởi chương trình Macromedia's Flash.
* .wmv files - được tạo bởi Microsoft's Window's Media Video.
* .mov files - được tạo bởi định dạng Apple's Quick Time Movie.
* .mpeg files - được tạo bởi Moving Pictures Expert Group.

*\*Sử dụng thẻ <bgsound> làm nhạc nền.*

<!DOCTYPE html>

<html>

<head>

<title>Vi du the embed trong HTML</title>

</head>

<body>

<bgsound src="../html/yourfile.mid">

<noembed><img src="yourimage.gif" ></noembed>

</bgsound>

</body>

</html>

*\*Thẻ object trong HTML:*

- Phần tử object cung cấp giải pháp chung cho tất cả các đối tượng.

+ Nhúng một đối tượng tài liệu HTML vào một tài liệu HTML:

<object data="data/test.htm" type="text/html" width="300" height="200">

alt : <a href="data/test.htm">test.htm</a>

</object>

Tại đây thì thuộc tính *alt* sẽ được sử dụng cho hình ảnh nếu trình duyệt không hỗ trợ thẻ *object*.

+ Nhúng một tài liệu pdf:

<object data="data/test.pdf" type="application/pdf" width="300" height="200">

alt : <a href="data/test.pdf">test.htm</a>

</object>

Có thể thiết lập một số tham số thông qua thẻ <param> như sau:

<object data="data/test.wav" type="audio/x-wav" width="200" height="20">

<param name="src" value="data/test.wav">

<param name="autoplay" value="false">

<param name="autoStart" value="0">

alt : <a href="data/test.wav">test.wav</a>

</object>

+ …

***19. Tìm hiểu thêm một số thẻ trong Header:***

- Thẻ <title> được sử dụng để xác định tiêu đề của tài liệu HTML. Dưới đây là ví dụ của thẻ này.

<!DOCTYPE html>

<html>

<head>

<title>Vi du the title trong HTML</title>

</head>

<body>

<p>Hello, World!</p>

</body>

</html>

- Thẻ <meta> cung cấp siêu dữ liệu về tài liệu HTML mà bao gồm các thông tin về tên tác giả của trang, danh sách các từ khóa, sự miêu tả trang, ...

<!DOCTYPE html>

<html>

<head>

<title>Vi du the Meta trong HTML</title><!-- The meta cung cap cac tu khoa -->

<meta name="keywords" content="HTML, C, C++, Java, PHP, Perl, Python"><!-- The meta cung cap phan mieu ta trang -->

<meta name="description" content="Hoc lap trinh co ban va nang cao tai VietJack"><!-- The meta cung cap thong tin tac gia -->

<meta name="author" content="VietJack"><!-- The meta xac dinh content type -->

<meta http-equiv="content-type" content="text/html; charset=UTF-8"><!-- The meta xac dinh thoi gian refresh -->

<meta http-equiv="refresh" content="30"><!-- The meta xac dinh thoi gian Expire -->

<meta http-equiv="expires" content="Wed, 21 June 2016 14:25:27 GMT"><!-- The meta noi cho robots khong lap index trang -->

<meta name="robots" content="noindex, nofollow"></head>

<body>

<p>Hello, World!</p>

</body>

</html>

- Thẻ <base> được sử dụng để xác định địa chỉ URL cơ bản cho tất cả các địa chỉ URL liên quan trong một trang, nghĩa là tất cả các địa chỉ URL khác sẽ được kết nối với địa chỉ URL cơ bản trong khi đặt vị trí cho các mục.

<!DOCTYPE html>

<html>

<head>

<title>Vi du the base trong HTML</title>

<base href="../html/" />

</head>

<body>

<img src="./images/logo3.png" alt="Logo Image"/>

<a href="../html/index.jsp" title="Hoc HTML co ban va nang cao"/>Bai huong dan HTML</a> </body>

</html>

- Thẻ <link> được sử dụng để xác định mối quan hệ giữa tài liệu hiện tại và nguồn bên ngoài. Sau đây là một ví dụ để kết nối với một Style Sheet có sẵn bên ngoài trong thư mục phụ **css** trong trang gốc.

<!DOCTYPE html>

<html>

<head>

<title>Vi du the link trong HTML</title>

<base href="../html/" />

<link rel="stylesheet" type="text/css" href="../css/style.css">

</head>

<body>

<p>Hello, World!</p>

</body>

</html>

- Thẻ <style> được sử dụng để xác định Style Sheet của tài liệu HTML hiện tại. Dưới đây là một ví dụ xác định một vài quy tắc Style Sheet trong thẻ <style>:

<!DOCTYPE html>

<html>

<head>

<title>Vi du the style trong HTML</title>

<base href="../html/" />

<style type="text/css">

.myclass{

background-color: #aaa;

padding: 10px;

}

</style>

</head>

<body>

<p class="myclass">Hello, World!</p>

</body>

</html>

- Thẻ <script> được sử dụng hoặc để bao một hoặc nhiều tệp script bên ngoài hoặc để xác định script bên trong của tài liệu HTML. Dưới đây là một ví dụ chúng tôi sử dụng Javascript để xác định một chức năng Javascript đơn giản:

<!DOCTYPE html>

<html>

<head>

<title>Vi du the script trong HTML</title>

<base href="../html/" />

<script type="text/javascript">

function Hello(){

alert("Hello, World");

}

</script>

</head>

<body>

<input type="button" onclick="Hello();" name="ok" value="OK" />

</body>

</html>

***20. CSS trong HTML:***

- Khi sử dụng HTML thuần, có thể tạo ra một text có màu như sau:

<!DOCTYPE html>

<html>

<head>

<title>Vi du HTML - CSS</title>

</head>

<body>

<p><font color="green" size="5">Hello, World!</font></p>

</body>

</html>

- Có thể sử dụng CSS để code lại cơ bản như sau:

<!DOCTYPE html>

<html>

<head>

<title>Vi du HTML - CSS</title>

</head>

<body>

<p style="color:green;font-size:24px;">Hello, World!</p>

</body>

</html>

=> Kết quả: Hello, World!

*\*Ba cách sử dụng CSS trong tài liệu HTML:*

* **External CSS** - Xác định các quy tắc của Style Sheet trong một file riêng rẽ và sau đó bao gồm file đó trong tài liệu HTML bằng cách sử dụng thẻ <link>.
* **Internal CSS** - Xác định các quy tắc của khu vực Header của tài liệu HTML bằng cách sử dụng thẻ <style>.
* **Inline CSS** - Xác định các quy tắc của Style Sheet một cách trực tiếp song song với các phần tử HTML bằng cách sử dụng thuộc tính **style**.

*\*External CSS:*

- Nếu muốn sử dụng chung một style nào đó trong tài liệu, bạn có thể code chúng trong file riêng đặt tên đuôi .css

Giả sử có 1 file style.css như sau:

.red{

color: red;

}

.thick{

font-size:20px;

}

.green{

color:green;

}

- Khi đó để sử dụng chúng trong HTML, ta làm như sau:

<!DOCTYPE html>

<html>

<head>

<title>Su dung External CSS trong HTML</title>

<link rel="stylesheet" type="text/css" href="../html/style.css">

</head>

<body>

<p class="red">This is red</p>

<p class="thick">This is thick</p>

<p class="green">This is green</p>

<p class="thick green">This is thick and green</p>

</body>

</html>

=> Kết quả:

This is red

This is thick

This is green

This is thick and green

*\*Sử dụng internal CSS:*

- Nếu muốn áp dụng một style CSS cho một tài liệu riêng lẻ, bạn có thể thiết lập những quy tắc này trong phần header của tài liệu như sau bằng cách sử dụng thẻ style:

<!DOCTYPE html>

<html>

<head>

<title>Su dung Internal CSS trong HTML</title>

<style type="text/css">

.red{

color: red;

}

.thick{

font-size:20px;

}

.green{

color:green;

}

</style>

</head>

<body>

<p class="red">This is red</p>

<p class="thick">This is thick</p>

<p class="green">This is green</p>

<p class="thick green">This is thick and green</p>

</body>

</html>

*\*Sử dụng Inline CSS:*

- Khi muốn thay đổi style của một phần tử riêng biệt, bạn có thể thực hiện thay đổi đó trong thuộc tính style của phần tử đó:

<!DOCTYPE html>

<html>

<head>

<title>Su dung Inline CSS trong HTML</title>

</head>

<body>

<p style="color:red;">This is red</p>

<p style="font-size:20px;">This is thick</p>

<p style="color:green;">This is green</p>

<p style="color:green;font-size:20px;">This is thick and green</p>

</body>

</html>

***21. JavaScript trong HTML:***

- Script là một phần nhỏ trong trang web nhằm tăng tính tương tác của trang web.

- Bạn có thể lưu JavaScript vào một tệp riêng biệt và sau đó include nó ở bất cứ đâu trong trang web của bạn, hoặc có thể thiết lập các tính năng này bên trong tài liệu HTML của bạn.

*\*Internal Script trong HTML:*

- Một tệp JavaScript có đuôi dạng .js và và được bao trong các tệp HTML bằng thẻ <script>.

- Ví dụ: trong file script dưới đây có một hàm: (dùng để hiển thị một alert box)

function Hello()

{

alert("Hello, World");

}

Trong tài liệu HTML lại có nội dung như sau:

<!DOCTYPE html>

<html>

<head>

<title>Vi du Javascript External Script</title>

<script src="../html/script.js" type="text/javascript"/></script>

</head>

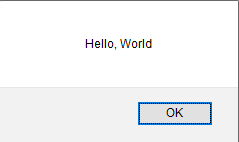
<body>

<input type="button" onclick="Hello();" name="ok" value="Click Me" />

</body>

</html>

=> Kết quả là có một nút bấm khi được bấm thì hiện lên alert box như trong hàm đã viết. (Ở đây không thể hiện được sự kiện)



*\*Internal Script trong HTML:*

- Có thể code một script trực tiếp trong tài liệu HTML và thông thường code trong header nhưng thực tế thì không giới hạn việc đặt code trong body.

<!DOCTYPE html>

<html>

<head>

<title>Su dung Internal Script trong HTML</title>

<base href="../html/" />

<script type="text/javascript">

function Hello(){

alert("Hello, World");

}

</script>

</head>

<body>

<input type="button" onclick="Hello();" name="ok" value="Click Me" />

</body>

</html>

*\*Xử lý sự kiện trong HTML:*

- Xử lý sự kiện thường gắn với chuột và bàn phím.

<!DOCTYPE html>

<html>

<head>

<title>Vi du Event Handler trong HTML</title>

<base href="../html/" />

<script type="text/javascript">

function EventHandler(){

alert("Vi du ham event handler!!");

}

</script>

</head>

<body>

<p onmouseover="EventHandler();">Re chuot qua dong nay va xem ket qua.</p>

</body>

</html>

- Nếu trình duyệt không hỗ trợ Java Script bạn có thể dùng thẻ <noscript> để hiển thị nội dung thay thế.

**Ví dụ cho JavaScript:**

<script type="text/javascript">

<!--

document.write("Hello Javascript!");

//-->

</script>

<noscript>Trinh duyet cua ban khong ho tro Javascript!</noscript>

**Ví dụ cho VBScript:**

<script type="text/vbscript">

<!--

document.write("Hello VBScript!")

'-->

</script>

<noscript>Trinh duyet cua ban khong ho tro VBScript!</noscript>

*\*Xác định ngôn ngữ Script mặc định trong HTML:*

- Có thể thiết lập ngôn ngữ script mặc định cho tất cả nội dung trong trang bằng thẻ thuộc tính content của thẻ <meta/>

<meta http-equiv="Content-Script-Type" content="text/JavaScript" />

**II. CSS:**

***1. CSS là gì?***

- CSS viết tắt là Cascade Style Sheet, là một Design Language đơn giản được sử dụng để đơn giản hoá quá trình thiết kế các trang web.

- CSS tiết kiệm được thời gian và công sức.

- CSS được tạo thành từ các style rule, các rule này được thông dịch bởi trình duyệt và sau đó được “áp dụng” trên file HTML.

- Style Rule gồm 3 phần:

+ Selector: (bộ chọn phần tử) Một selector là một thẻ HTML mà tại đó style được áp dụng, đó có thể là bất kỳ thẻ nào.

+ Property: (thuộc tính) Là thuộc tính của kiểu thẻ HTML, là tất cả các thuộc tính trong HTML được chuyển thành CSS Property.

+ Value: Là giá trị được gán cho property.

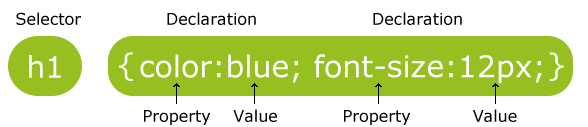
- Cú pháp chung: selector { property: value }

p {

color: red;

text-align: left;

}



*\*Các loại Selector trong CSS:*

- Element Selector: Bộ chọn này chọn phần tử dựa trên tên của chúng.

p {

text-align: center;

color: red;

}

- Type Selector:

h1 {

color: #36CFFF;

}

- Universal Selector:

\* {

color: #000000;

}

- Descendent Selector: (chưa hiểu)

- Class Selector: Áp dụng các Style Rule cho các phần tử dựa trên thuộc tính class của chúng.

Để chọn các phần tử với một class cụ thể, viết một dấu “.” trước tên class đó.

.center {

text-align: center;

color: red;

}

Với trường hợp chỉ muốn áp dụng các rule cho một phần tử trong class có nhiều phần tử:

div.center {

text-align: center;

color: red;

}

Với trường hợp các phần từ tham chiếu nhiều class, nhiều style rule sẽ được áp dụng trên phần tử đó. (Ở đây thuộc tính của div sẽ được lấy từ 2 rule của 2 class là center và black.

<div class="center black">

DIV này sẽ được tạo style bởi hai class là center và black.

</div>

- ID Selector: Sử dụng thuộc tính ID của một phần tử để thiết lập Rule, do trong một page không có ID duy nhất nên ID Selector được sử dụng để thiết lập cho một phần tử duy nhất.

Để lựa chọn một phần tử với một id cụ thể, bạn viết một ký tự # (đọc là ký tự hash) ở trước id của phần tử đó.

<!DOCTYPE html>

<html>

<head>

<style>

#a{

color: red;

/\* Vi du ve comment tren mot dong \*/

text-align: center;

}

/\* Vi du ve comment

tren nhieu dong

trong CSS

\*/

</style>

</head>

<body>

<p id = "a">Hoc CSS co ban tai VietJack!</p>

</body>

</html>

Có thể thể hiện một cách chi tiết hơn về thiết lập thuộc tính cho một phần tử có ID bằng cách sau: (phần màu xanh)

<!DOCTYPE html>

<html>

<head>

<style>

p#a{

color: red;

/\* Vi du ve comment tren mot dong \*/

text-align: center;

}

/\* Vi du ve comment

tren nhieu dong

trong CSS

\*/

</style>

</head>

<body>

<p id = "a">Hoc CSS co ban tai VietJack!</p>

</body>

</html>

- Child Selector: Áp dụng cho các phần tử là con trực tiếp của phần tử khác:

body > p {

color: #000000;

}

Rule này sẽ áp dụng cho tất cả các phần tử p là con trực tiếp của body, không áp dụng cho các phần tử bên trong nó.

- Multi Style Rule trong CSS: (sử dụng dấu “;” để ngăn cách các thuộc tính).

h1 {

color: #36C;

font-weight: normal;

letter-spacing: .4em;

margin-bottom: 1em;

text-transform: lowercase;

}

- Group Selector:

h1, h2, h3 {

color: #36C;

font-weight: normal;

letter-spacing: .4em;

margin-bottom: 1em;

text-transform: lowercase;

}

Hoặc:

#content, #footer, #supplement {

position: absolute;

left: 510px;

width: 200px;

}

- Comment trong CSS:

Comment trong CSS nằm trong cặp: /\* \*/

p {

color: red;

/\* vi du ve comment nam tren mot dong \*/

text-align: center;

}

/\* vi du ve

comment nam

tren nhieu dong

\*/

***2 Include CSS vào HTML:***

*\*Internal CSS:*

- Cách này áp dụng cho những site nhỏ, là cách sử dụng CSS vào tài liệu HTML thông qua thẻ <style> đặt bên trong các thẻ <head>.

- Style Rule trong trang sẽ chỉ áp dụng cho trang đó.

<!DOCTYPE html>

<html>

<head>

<style type = "text/css" media = "all">

body {

background-color: linen;

}

h1 {

color: maroon;

margin-left: 40px;

}

</style>

</head>

<body>

<h1>Day la phan tieu de cho tai lieu</h1>

<p>Day la vi du mot doan van trong tai lieu.</p>

</body>

</html>

=> Kết quả:



*\*Các thuộc tính trong thẻ <style>:*



*\*Inline CSS:*

- Chỉ cách nhúng CSS vào từng phần tử trong tài liệu HTML bằng cách sử dụng thuộc tính style trong phần tử đó.

- Cú pháp chung:

<tenPhanTu style = "...style rules....">

<html>

<head>

</head>

<body>

<h1 style = "color:#36C;"> Vi du ve Inline CSS </h1>

</body>

</html>

=> Kết quả: 

*\*External CSS:*

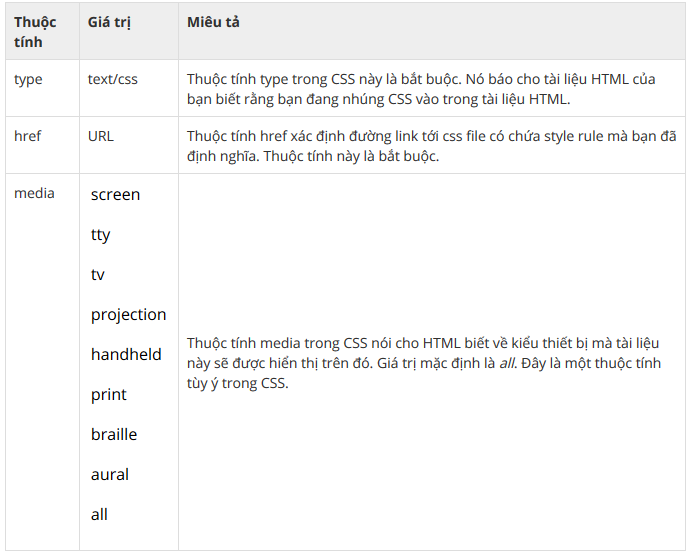
- Chỉ cách nhúng CSS vào tài liệu HTML bằng cách sử dụng một file đuôi .css để định nghĩa tất cả các style rule trong đó và sử dụng thẻ <link> để liên kết chúng trong tài liệu HTML.

- Cú pháp nhúng:

<head>

<link type = "text/css" href = "..." media = "..." />

</head>



- Ví dụ:

Nội dung trong file style.css:

h1, h2, h3 {

color: #36C;

font-weight: normal;

letter-spacing: .4em;

margin-bottom: 1em;

text-transform: lowercase;

}

Nội dung trong file jsp:

<head>

<link type = "text/css" href = "vietjack.css" media = " all" />

</head>

- Có thể sử dụng @import thay cho thẻ link để thực hiện external CSS:

<head>

@import "style.css";

</head>

*\*Độ ưu tiên import CSS: Inline -> Internal -> External (theo thứ tự cao tới thấp).*

***3. Kích thước trong CSS:***

|  |  |  |
| --- | --- | --- |
| **Đơn vị** | **Miêu tả** | **Ví dụ** |
| % | Đơn vị %, là đơn vị tham chiếu tỉ lệ so với một phần tử mẹ dựa vào kích thước. Ví dụ: bạn có một khung mẹ có kích thước là 400px và bạn muốn đặt một khung khác vào bên trong khung mẹ này. Tùy theo lựa chọn thì khung con này sẽ có nhiều kích thước, và tương ứng với kích thước khung mẹ thì bạn sẽ có tỉ lệ tương ứng. Ví dụ: nếu là 50% thì khung con có kích thước là 200px | p {font-size: 16pt; line-height: 125%;} |
| cm | Đơn vị centimet | div {margin-bottom: 2cm;} |
| em | Một đơn vị được sử dụng cho chiều cao của font. Một đơn vị em tương đương với kích cỡ của một font đã cho, nếu bạn gán kích cỡ của font là 12pt thì mỗi đơn vị em sẽ là 12pt, do đó 2em sẽ là 24pt | p {letter-spacing: 7em;} |
| ex | Đơn vị được sử dụng cho chiều cao của chữ cái x thường (hay là x-height) | p {font-size: 24pt; line-height: 3ex;} |
| in | Đơn vị inch (=2.54 cm) | p {word-spacing: .15in;} |
| mm | Đơn vị milimet | p {word-spacing: 15mm;} |
| pc | Đơn vị pica. Một pica = 12 point và do đó 1 inch = 6 pica | p {font-size: 20pc;} |
| pt | Đơn vị point. 1 inch = 72 point | body {font-size: 18pt;} |
| px | Đơn vị được sử dụng trên màn hình hiển thị, một px sẽ tương đương với một điểm ảnh trên màn hình. Chất lượng của điểm ảnh sẽ hoàn toàn khác nhau trên một số thiết bị, ví dụ như một điểm ảnh trên các thiết bị in ấn sẽ khác một điểm ảnh trên các thiết bị màn hình có độ phân giải thấp. | p {padding: 25px;} |

- Có thể kết hợp các đơn vị em và & để thiết lập kích thước font size tỷ lệ với màn hình.

<html>

<head>

<style>

body {

font-size: 100%;

}

h1 {

font-size: 2.5em;

}

h2 {

font-size: 1.875em;

}

p {

font-size: 0.875em;

}

</style>

</head>

<body>

<h1>Phan tieu de 1</h1>

<h2>Phan tieu de 2</h2>

<p>Day la mot doan van.</p>

<p>Xac dinh kich co font bang % va em la lua chon hieu qua nhat cho tat ca trinh duyet!</p>

</body>

</html>

***4. Màu trong CSS (Tham khảo trong mục HTML):***

***5. Background trong CSS:***

*­­*- Các thuộc tính CSS liên quan tới Background:

* Thuộc tính **background-color**: thuộc tính này được sử dụng để thiết lập màu nền của một phần tử.
* Thuộc tính **background-image**: thuộc tính này được sử dụng để thiết lập hình nền cho một phần tử.
* Thuộc tính **background-repeat**: thuộc tính này được sử dụng để điều khiển sự lặp đi lặp lại của một hình ảnh nền theo chiều dọc hoặc chiều ngang.
* Thuộc tính **background-position**: thuộc tính này được sử dụng để điều khiển vị trí của một hình ảnh nền.
* Thuộc tính **background-attachment**: thuộc tính này được sử dụng để xác định xem có hay không một hình nền là cố định hoặc có thể scroll với phần còn lại của trang.
* Thuộc tính **background**: sử dụng thuộc tính này nếu bạn muốn viết ít code hơn mà vẫn xác định được tất cả các thuộc tính liên quan tới background ở trên cho hình nền.

- Background color:

<html>

<head>

<body>

<p **style = "background-color:yellow;"**>

Phan noi dung nay co Background Color la mau vang.</p>

</body>

</head>

<html>

=> Kết quả:



- Background image:

<html>

<head>

<style>

body {

background-image: url("../css/images/css.jpg");

background-repeat: repeat;

}

</style>

</head>

<body>

<p>Vi du ve gia tri mac dinh cua thuoc tinh background-repeat trong CSS.</p>

</body>

</html>

- Thuộc tính background-repeat: lặp lại của hình nền chỉ việc hình nền sẽ được lặp lại khi kích thước màn hình lớn hơn kích thước ảnh nền. Có thể thiết lập điều này bằng thuộc tính repeat trong CSS với các giá trị sau:

+ repeat: Giá trị mặc định. Hình nền sẽ được lặp đi lặp lại theo cả chiều dọc lẫn chiều ngang.

+ repeat-x: Hình nền sẽ chỉ được lặp đi lặp lại theo chiều ngang.

+ repeat-y: Hình nền sẽ chỉ được lặp đi lặp lại theo chiều dọc.

+ no-repeat: Hình nền sẽ không được lặp đi lặp lại.

- Thuộc tính background-position và background-attachment:

+ background-position: dùng để thiết lập vị trí của hình nền, đơn vị thường là px.

Cú pháp: background-position: kc\_chieu\_ngang kc\_chieu\_doc;

background-position:100px 200px;

+ background-attachment: dùng để thiết lập việc hình sẽ được kéo theo khi rê chuột hay giữ im mang các giá trị là: fixed và scroll.

- Thuộc tính background: dùng để làm giảm số lượng code bằng cách đặt tất cả các thuộc tính trong một thuộc tính duy nhất là background nhưng yêu cầu các thuộc tính này phải được **viết theo đúng thứ tự:**

+ background-color

+ background-image

+ background-repeat

+ background-attachment

+ background-position

<p **style="background:url(../css/images/hinh\_anh\_nen.gif) repeat fixed;"**>

Doan van nay co hinh nen co dinh (gia tri fixed) va co the lap di lap lai (gia tri repeat)!

</p>

***6. Font trong CSS:***

- Các thuộc tính của font trong CSS gồm có:

+ Thuộc tính **font-family**: được sử dụng để thay đổi bề mặt font

+ Thuộc tính **font-style**: được sử dụng để tạo một font chữ nghiêng hoặc chếch

+ Thuộc tính **font-variant**: được sử dụng để tạo những chữ hoa nhỏ (small-cap)

+ Thuộc tính **font-weight**: được sử dụng để tăng giảm độ đậm của font

+ Thuộc tính **font-size**: được sử dụng để xác định kích cỡ font

+ Thuộc tính **font**: sử dụng thuộc tính này nếu bạn muốn viết ít code hơn mà vẫn xác định được các thuộc tính liên quan tới font ở trên.

- Thuộc tính font-family trong CSS:

+ Trong CSS, có hai loại Font Family:

generic family: một nhóm các font family có font face khá tương tự nhau.

font family: là một font family cụ thể.

+ Khi muốn thay đổi font-face trong CSS ta có thể sử dụng thuộc tính font-family, khi sử dụng thuộc tính này nên thiết lập nhiều hơn 1 giá trị để khi trình duyệt không thể hỗ trợ được font thứ nhất thì sẽ lấy font tiếp theo đê thực hiện.

<html>

<head>

</head>

<body>

<p **style="font-family:georgia,garamond,serif;"**>

Phan noi dung nay duoc hien thi voi mot trong cac font: georgia, garamond, hoac gia tri mac dinh la serif

phu thuoc vao trinh duyet cua ban.

</p>

</body>

</html>

- Thiết lập font-style trong CSS: Thuộc tính này mang 3 giá trị là normal (bình thường), italic (nghiêng) và oblique (giống với Italic) nhưng ít được hỗ trợ hơn.

<html>

<head>

</head>

<body>

<p **style="font-style:italic;"**>

Doan van nay se duoc hien thi duoi dang in nghieng.

</p>

</body>

</html>

- Thiết lập font-variant: Thuộc tính này dùng để tạo các chữ hoa nhỏ, mang hai giá trị là normal (bình thường) và small-caps.

<html>

<head>

</head>

<body>

<p **style="font-variant:small-caps;"**>

Doan van nay se duoc hien thi duoi dang cac chu hoa nho.

</p>

</body>

</html>

=> Kết quả: 

- Thiết lập font-weight trong CSS: Dùng để thiết lập độ đậm của văn bản mang các giá trị như sau: *normal, bold, bolder, lighter, 100, 200, 300, 400, 500, 600, 700, 800, 900*.

<html>

<head>

</head>

<body>

<p **style="font-weight:bold;"**>Doan van nay duoc hien thi duoi dang in dam.</p>

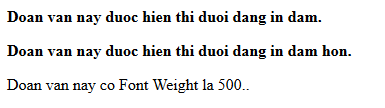
<p **style="font-weight:bolder;"**>Doan van nay duoc hien thi duoi dang in dam hon.</p>

<p **style="font-weight:500;"**>Doan van nay co Font Weight la 500..</p>

</body>

</html>

=> Kết quả:



- Thiết lập thuộc tính font-size trong CSS: Dùng để thiết lập kích thước chữ, có thể mang các giá trị như sau:  *x-small, x-small, small, medium, large, x-large, xx-large, smaller, larger, size in pixels hoặc in %.*

- Thiết lập thuộc tính font-size-adjust trong CSS: Có thể dùng để thiết lập kích cỡ văn bản của một phần tử, giúp thiết lập chiều rộng của chữ:

<**html**>

<**head**>

</**head**>

<**body**>

<**p** style="font-size-adjust:*0.61*;">

Doan van nay su dung thuoc tinh font-size-adjust co gia tri 0.61.

</**p**>

</**body**>

</**html**>



- Thiết lập font-size trong CSS bằng pixel: (có gì đó sai sai)

<**html**>

<**head**>

</**head**>

<**body**>

<**p** style="font-stretch:*ultra-expanded*;">

Neu gia tri cua thuoc tinh nay khong co hieu qua, tuc la may tinh cua ban

khong ho tro hai gia tri: condensed hoac expanded.

</**p**>

</**body**>

</**html**>

- Thiết lập font-size bằng em: Nhiều người sử dụng em để thiết lập kích thước văn bản thay cho px. Đơn vị này được W3C khuyến nghị sử dụng. Một em bằng kích thước font văn bản hiện tại. Kích cỡ mặc định là 16 px.

<**html**>

<**head**>

<**style**>

h1 {

font-size: *2.5em*; /\* 40px/16=2.5em \*/

h2 {

font-size: *1.875em*; /\* 30px/16=1.875em \*/

p {

font-size: *0.875em*; /\* 14px/16=0.875em \*/}

</**style**>

</**head**>

<**body**>

<**h1**>Phan tieu de 1</**h1**>

<**h2**>Phan tieu de 2</**h2**>

<**p**>Day la mot doan van.</**p**>

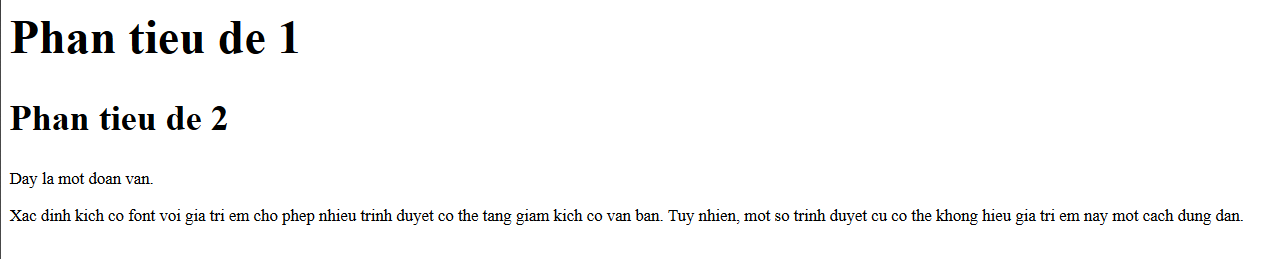
<**p**>Xac dinh kich co font voi gia tri em cho phep nhieu trinh duyet

co the tang giam kich co van ban. Tuy nhien, mot so trinh duyet cu co

the khong hieu gia tri em nay mot cach dung dan.</**p**>

</**body**>

</**html**>



- Sử dụng kết hợp em và % trong CSS: Là giải pháp để hiển thị kích thước văn bản hài hoà trên mọi trình duyệt. Khi thiết lập kích thước cho văn bản, nên thiết lập một font-size mặc định bằng 100% cho phần tử body:

<**html**>

<**head**>

<**style**>

body {

font-size: *100%*;

}

h1 {

font-size: *2.5em*;

}

h2 {

font-size: *1.875em*;

}

p {

font-size: *0.875em*;

}

</**style**>

</**head**>

<**body**>

<**h1**>Phan tieu de 1</**h1**>

<**h2**>Phan tieu de 2</**h2**>

<**p**>Day la mot doan van.</**p**>

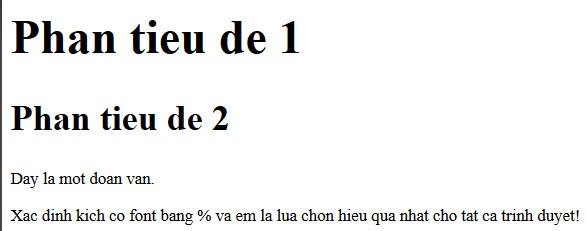
<**p**>Xac dinh kich co font bang % va em la lua chon hieu qua nhat cho

tat ca trinh duyet!</**p**>

</**body**>

</**html**>

=> Kết quả:



- Thuộc tính font: Dùng khi muốn rút gọn code, các giá trị thuộc tính bên trong cần được viết theo thứ tự như sau:

font-style font-variant font-weight font-size/line-height font-family

hoặc:

font: font-style font-variant font-weight font-size/line-height

font-family|caption|icon|menu|small-caption|status-bar;

<**html**>

<**head**>

</**head**>

<**body**>

<**p** style="font: *italic small-caps bold 15px georgia*;">Bang Style

Rule tren, ban co the xac dinh gia tri cho tat ca cac thuoc tinh lien

quan toi font.</**p**>

</**body**>

</**html**>

***7. Text trong CSS:***

- Text trong CSS có các thuộc tính sau:

+ Thuộc tính **color** được sử dụng để thiết lập màu cho văn bản.

+ Thuộc tính **direction** được sử dụng để thiết lập hướng cho văn bản.

+ Thuộc tính **letter-spacing** được sử dụng để thêm hoặc bớt khoảng cách giữa các chữ cái trong một từ.

+ Thuộc tính **word-spacing** được sử dụng để tăng hoặc giảm khoảng cách giữa các từ trong một câu.

+ Thuộc tính **text-indent** được sử dụng để tạo độ thụt của văn bản trong một đoạn văn.

+ Thuộc tính **text-align** được sử dụng để căn chỉnh văn bản trong một tài liệu.

+ Thuộc tính **text-decoration** được sử dụng để tạo cách dấu gạch ở chân, ở trên, ở giữa văn bản.

+ Thuộc tính **text-transform** được sử dụng để chuyển văn bản thành chữ hoa hoặc chữ thường.

+ Thuộc tính **white-space** được sử dụng để định dạng và điều khiển phần khoảng trắng của văn bản.

+ Thuộc tính **text-shadow** được sử dụng để thiết lập hình bóng (shadow như trong word) xung quanh văn bản.

- Thiết lập thuộc tính color cho text: giá trị là bất kỳ tên màu nào hoặc định dạng màu hợp lệ nào.

**html**>

<**head**>

</**head**>

<**body**>

<**p** style="color: *red*;">Phan van ban nay duoc viet bang mau do.</**p**>

</**body**>

</**html**>

=> Kết quả:



- Thiết lập hướng văn bản bằng thuộc tính direction: thuộc tính này có hai giá trị là ltr (left to right) và rtl (right to left).

<**html**>

<**head**>

</**head**>

<**body**>

<**p** style="direction:*rtl*;">

Van ban se duoc hien thi theo huong tu phai qua trai.

</**p**>

</**body**>

</**html**>

=> Kết quả: 

- Thiết lập khoảng cách giữa các chữ cái trong một từ bằng thuộc tính letter-spacing: thuộc tính này có thể nhận giá trị là normal (mặc định) hoặc một số cụ thể:

<**html**>

<**head**>

</**head**>

<**body**>

<**p** style="letter-spacing:*5px*;">

Cac chu cai trong mot tu se co khoang cach la 5 pixel.

</**p**>

</**body**>

</**html**>

=> Kết quả: 

- Thiết lập khoảng cách giữa các từ (hai từ liền nhau) bằng thuộc tính word-spacing: thuộc tính này có thể nhận giá trị normal hoặc một số cụ thể.

<**html**>

<**head**>

</**head**>

<**body**>

<**p** style="word-spacing: *5px*;">Khoang cach giua hai tu lien ke nhau

la 5 pixel.</**p**>

</**body**>

</**html**>

=> Kết quả:



- Thiết lập độ thụt văn bản (thụt dòng đầu tiên của mỗi đoạn văn) với thuộc tính text-indent trong CSS: thuộc tính này có thể nhận các giá trị % hoặc một số cụ thể.

<**html**>

<**head**>

</**head**>

<**body**>

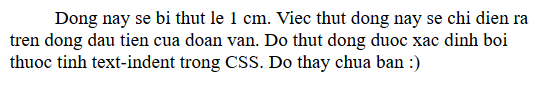
<**p** style="text-indent: *1cm*;">Dong nay se bi thut le 1 cm. Viec thut

dong nay se chi dien ra tren dong dau tien cua doan van. Do thut dong

duoc xac dinh boi thuoc tinh text-indent trong CSS. Do thay chua ban :)</**p**>

</**body**>

</**html**>



- Căn chỉnh văn bản với thuộc tính text-align: thuộc tính này có 4 giá trị là: left, right, center.

<**html**>

<**head**>

</**head**>

<**body**>

<**p** style="text-align: *right*;">Can chinh ben phai (gia tri right).</**p**>

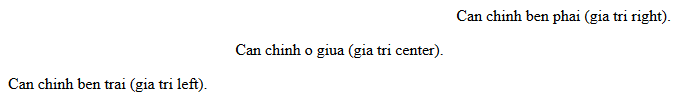
<**p** style="text-align: *center*;">Can chinh o giua (gia tri center).</**p**>

<**p** style="text-align: *left*;">Can chinh ben trai (gia tri left).</**p**>

</**body**>

</**html**>

=> Kết quả:



- Trang trí văn bản với thuộc tính text-decoration: Thuộc tính này mang các giá trị như sau: *none, overline (dấu gạch ở trên), underline (gạch chân), line-through (gạch ngang).*

<**html**>

<**head**>

</**head**>

<**body**>

<**p** style="text-decoration: *underline*;">Doan van nay se co duong

gach ngang duoi chan.</**p**>

<**p** style="text-decoration: *line-through*;">Doan van nay se bi gach

ngang.</**p**>

<**p** style="text-decoration: *overline*;">Doan van nay se co duong gach

ngang tren dau.</**p**>

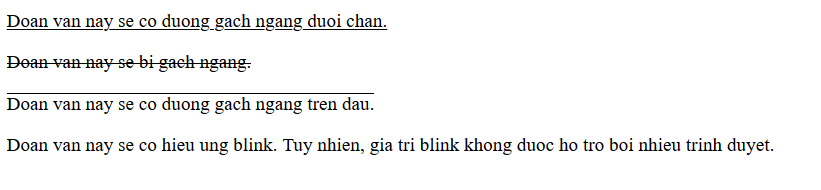
<**p** style="text-decoration: *blink*;">Doan van nay se co hieu ung

blink. Tuy nhien, gia tri blink khong duoc ho tro boi nhieu trinh

duyet.</**p**>

</**body**>

</**html**>

=> Kết quả: 

- Thiết lập chữ hoa và chữ thường trong CSS: thuộc tính này mang 4 giá trị là none, capitalize (viết hoa chữ cái đầu), uppercase (viết hoa toàn bộ), lowercase (viết thường toàn bộ).

<**html**>

<**head**>

</**head**>

<**body**>

<**p** style="text-transform: *capitalize*;">Viet hoa chu cai dau tien.</**p**>

<**p** style="text-transform: *uppercase*;">Viet hoa toan bo chu cai.</**p**>

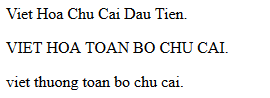
<**p** style="text-transform: *lowercase*;">Viet thuong toan bo chu cai.

</**p**>

</**body**>

</**html**>

=> Kết quả:



- Thiết lập khoảng cách giữa các khoảng trắng trong CSS: normal, pre, nowrap

- Tạo shadow trong CSS: Cú pháp để tạo bóng trong CSS như sau:

text-shadow: *h-shadow v-shadow blur-radius* color|none|initial|inherit;

<**html**>

<**head**>

</**head**>

<**body**>

<**p** style="text-shadow: *4px 4px 8px blue*;">Neu trinh duyet cua ban

khong ho tro thuoc tinh text-shadow trong CSS, thi phan van ban se co

Blue Shadow.</**p**>

</**body**>

</**html**>

=> Kết quả:



***8. Hình ảnh trong CSS:***

- Các thuộc tính cho hình ảnh trong CSS:

+ Thuộc tính **border**: Thiết lập độ rộng của đường viền bao quanh hình ảnh.

+ Thuộc tính **height**: Thiết lập chiều cao của hình ảnh.

+ Thuộc tính **width**: Thiết lập độ rộng của hình ảnh.

+ Thuộc tính **–moz-opacity**: thiết lập độ trong suốt của hình ảnh.

- Thiết lập thuộc tính border: nhận giá trị đo bằng px hoặc %

<**html**>

<**head**>

</**head**>

<**body**>

<**img** style="border: *0px*;"

src=*"https://i.imgur.com/inBsikg.png"* />

<**br** />

<**img** style="border: *3px dashed red*;"

src=*"https://i.imgur.com/inBsikg.png"* />

</**body**>

</**html**>

=> Kết quả:



- Thuộc tính height, width: giá trị là pixel hoặc %

- Thuộc tính -moz-opacity, opacity (được hỗ trợ):

<**html**>

<**head**>

</**head**>

<**body**>

<**img** style="border: *0px*;" src=*"https://i.imgur.com/inBsikg.png"* />

<**br** />

<**img**

style="border: *3px dashed red*; -moz-opacity: *0.1*; opacity: *0.4*;"

src=*"https://i.imgur.com/inBsikg.png"* />

</**body**>

</**html**>

=> Kết quả:



***9. Trang trí link:***

- Link trong CSS có 4 trạng thái:

+ Trạng thái :**link** – biểu thị rằng trang web này trình duyệt chưa lưu (tức người sử dụng lần đầu tiên click vào đường dẫn này).

+ Trạng thái :**visited** – biểu thị rằng đường dẫn tới trang web này đã được lưu bởi trình duyệt (tức là người sử dụng đã click vào đường dẫn này trước đó rồi).

+ Trạng thái :**hover** – biểu thị rằng khi người sử dụng di chuyển chuột qua phần tử mà chứa link đó (tức là phần tử đó là một link khi người sử dụng di chuyển chuột qua phần tử đó).

+ Trạng thái :**active** – biểu thị đường link là active khi người sử dụng click chuột vào.

- Khi lập trình CSS nên lưu ý như sau:

+ Trạng thái :hover PHẢI theo sau các trạng thái :link và :visited trong định nghĩa CSS thì mới có hoạt động. Nếu thiếu một trong hai trạng thái trên, thì các Style Rule bạn đã định nghĩa cho :hover sẽ không có giá trị.

+ Trạng thái :active PHẢI theo sau trạng thái :hover trong định nghĩa CSS.

<**style** type=*"text/css"*>

a:link {color: *#000000*}

a:visited {color: *#006600*}

a:hover {color: *#FFCC00*}

a:active {color: *#FF00CC*}

</**style**>

***10. Border trong CSS:***

- Trong CSS, có các thuộc tính sau đây để thiết lập giá trị cho border:

* Thuộc tính **border-color** xác định màu của đường viền.
* Thuộc tính **border-style** giúp bạn xác định xem đường viền có style gì, có thể mang các giá trị như: **none, solid, dotted, dashed, double, groove, ridge, inset, outset, hidden.**
* Thuộc tính **border-width** giúp bạn xác định độ rộng của đường viền, có thể có các giá trị: px, pt hoặc cm hoặc có thể là *thin, medium hoặc thick.*
* Thuộc tính **border-bottom-color** giúp bạn thay đổi màu của đáy đường viền.
* Thuộc tính **border-top-color** giúp bạn thay đổi màu của phần trên đường viền.
* Thuộc tính **border-left-color** giúp bạn thay đổi màu của cạnh trái đường viền.
* Thuộc tính **border-right-color** giúp bạn thay đổi màu của cạnh phải đường viền.
* Thuộc tính **border-bottom-style** thay đổi style cho cạnh dưới của đường viền.
* Thuộc tính **border-top-style** thay đổi style cho cạnh trên của đường viền.
* Thuộc tính **border-left-style** thay đổi style cho cạnh trái của đường viền.
* Thuộc tính **border-right-style** thay đổi style cho cạnh phải của đường viền.

***11. Căn lề (Margin):***

- Các thuộc tính trong CSS liên quan margin (xác định khoảng cách phần nằm ngoài ngoài đường viền):

+ Thuộc tính **margin**: sử dụng thuộc tính này bạn có thể thiết lập tất cả style liên quan tới việc căn lề chỉ trong một khai báo CSS.

+ Thuộc tính **margin-bottom** căn lề dưới của một phần tử.

+ Thuộc tính **margin-top** căn lề trên của một phần tử.

+ Thuộc tính **margin-left** căn lề trái của một phần tử.

+ Thuộc tính **margin-right** căn lề phải của một phần tử.

- Tất cả các thuộc tính liên quan tới Margin trên đều có thể nhận các giá trị sau:

**+ auto**: Trình duyệt tự động ước lượng việc căn lề cho phần tử.

**+ length**: Xác định độ rộng (đơn vị px, pt, cm, …) của lề. Giá trị mặc định là 0.

**+ %**: Xác định mối quan hệ giữa lề với độ rộng của phần tử chứa lề.

**+ inherit**: Kế thừa thuộc tính này từ phần tử cha chứa phần tử có thuộc tính margin này.

- Thuộc tính margin:

p {

margin: 100px 150px 100px 80px;

}

**margin: 25px 50px 75px 100px;**

* Lề trên là 25px
* Lề phải là 50px
* Lề dưới là 75px
* Lề trái là 100px

**margin: 25px 50px 75px;**

* Lề trên là 25px
* Lề phải và trái là 50px
* Lề dưới là 75px

**margin: 25px;**

* Các lề trên, lề dưới, lề trái, và lề phải là 25px

<**html**>

<**head**>

</**head**>

<**body**>

<**p** style="margin: *15px*; border: *1px solid black*;">Ca 4 le: tren,

duoi, trai va phai deu co do rong le la 15px</**p**>

<**p** style="margin: *10px 2%*; border: *1px solid black*;">Le tren va le

duoi la 10px. Do rong cua le trai va le phai bang 2% tong do rong cua

trang.</**p**>

<**p** style="margin: *10px 2% -10px*; border: *1px solid black*;">Le tren

la 10px. Do rong cua le trai va le phai bang 2% tong do rong cua

trang. Le duoi la -10px</**p**>

<**p** style="margin: *10px 2% -10px auto*; border: *1px solid black*;">Le

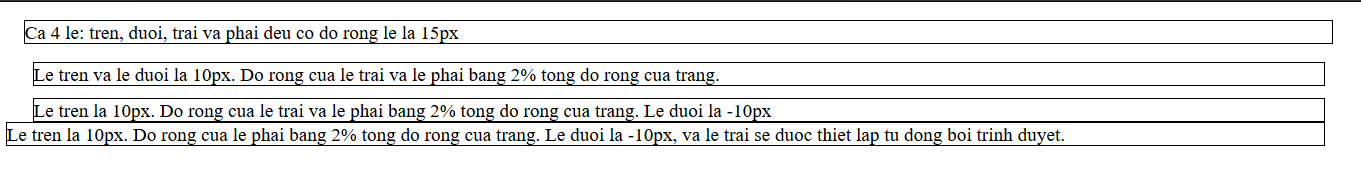
tren la 10px. Do rong cua le phai bang 2% tong do rong cua trang. Le

duoi la -10px, va le trai se duoc thiet lap tu dong boi trinh duyet.</**p**>

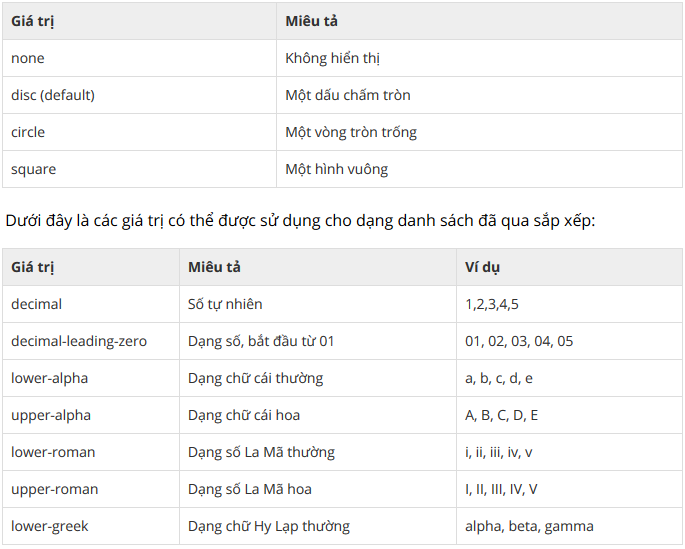
</**body**>

</**html**>

=> Kết quả:



***12. List và marker:***



***13. Padding trong CSS:***

- Padding là khoảng cách giữa phần nội dung và phần viền của nội dung đó.

- Các thuộc tính liên quan tới padding trong CSS:

+ Thuộc tính **padding-bottom** xác định phần padding bên dưới của một phần tử.

+ Thuộc tính **padding-top** xác định phần padding bên trên của một phần tử.

+ Thuộc tính **padding-left** xác định phần padding bên trái của một phần tử.

+ Thuộc tính **padding-right** xác định phần padding bên phải của một phần tử.

+ Thuộc tính **padding** giúp bạn xác định toàn bộ các thuộc tính trên, tức là có thể xác định toàn bộ phần không gian padding bao quanh một phần tử.

***14. Outline trong CSS:***

- Outline là một đường vẽ nằm ngay bên ngoài hình và outline không chiếm giữ không gian, không làm thay đổi kích thước của hình.

- Các thuộc tính trong CSS hỗ trợ outline:

+ Thuộc tính **outline-width** xác định độ rộng của outline.

+ Thuộc tính **outline-style** xác định line style cho outline, có thể có các giá trị:

**none**: tức là không có border (tương đương với border-width: 0;).

**solid**: đường viền liền đơn.

**dotted**: đường viền là một dãy các chấm nhỏ.

**dashed**: đường viền là một dãy các đoạn thẳng ngắn.

**double**: đường viền là hai đường dạng solid.

**groove**: đường viền trông giống như được khắc vào trong trang.

**ridge**: đường viền trông ngược với dạng groove.

**inset**: đường viền trông giống như là nó được nhúng vào trong trang.

**outset**: ngược lại với dạng inset.

**hidden**: giống như none.

+ Thuộc tính **outline-color** xác định màu của outline.

+ Thuộc tính **outline** để xác định tất cả thuộc tính của outline trên.

***15. Kích thước trong CSS:***

- Có thể thay đổi kích thước các box trong CSS bằng các thuộc tính như sau:

+ Thuộc tính **height** xác định chiều cao của hộp.

+ Thuộc tính **width** xác định độ rộng của hộp.

+ Thuộc tính **line-height** xác định chiều cao của một dòng văn bản.

+ Thuộc tính **max-height** thiết lập chiều cao tối đa của một hộp.

+ Thuộc tính **min-height** thiết lập chiều cao tối thiểu của một hộp.

+ Thuộc tính **max-width** thiết lập độ rộng tối thiểu của một hộp.

+ Thuộc tính **min-width** thiết lập độ rộng tối thiểu của một hộp.

- Các thuộc tính height và width:

<**html**>

<**head**>

</**head**>

<**body**>

<**p** style="margin: *15px*; border: *1px solid black*;">Ca 4 le: tren,

duoi, trai va phai deu co do rong le la 15px</**p**>

<**p** style="margin: *10px 2%*; border: *1px solid black*;">Le tren va le

duoi la 10px. Do rong cua le trai va le phai bang 2% tong do rong cua

trang.</**p**>

<**p** style="margin: *10px 2% -10px*; border: *1px solid black*;">Le tren

la 10px. Do rong cua le trai va le phai bang 2% tong do rong cua

trang. Le duoi la -10px</**p**>

<**p** style="margin: *10px 2% -10px auto*; border: *1px solid black*;">Le

tren la 10px. Do rong cua le phai bang 2% tong do rong cua trang. Le

duoi la -10px, va le trai se duoc thiet lap tu dong boi trinh duyet.</**p**>

</**body**>

</**html**>

<**html**>

<**head**>

</**head**>

<**body**>

<**p**

style="width: *400px*; height: *100px*; border: *1px solid red*; padding: *5px*; margin: *10px*;">

Doan van nay co do rong la 400 pixel va chieu cao la 100 pixel</**p**>

</**body**>

</**html**>

- Thuộc tính line-height: thuộc tính này thay đổi khoảng cách giữa các dòng văn bản:

<**html**>

<**head**>

</**head**>

<**body**>

<**p** style="width:*400px*; height:*100px*; border:*1px solid red*; padding:*5px*; margin:*10px*; line-height:*30px*;">

Doan van co do rong la 400 px va chieu cao la 100 px. Trong do, line height la 30 px.

Doan van co do rong la 400 px va chieu cao la 100 px. Trong do, line height la 30 px.

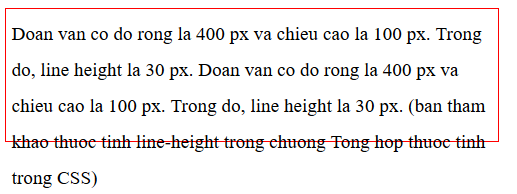
(ban tham khao thuoc tinh line-height trong chuong Tong hop thuoc tinh trong CSS)

</**p**>

</**body**>

</**html**>

=> Kết quả:



- Các thuộc tính max-height, min-height, max-width, max-height trong CSS: Nhằm thiết lập các kích thước tối đa và tối thiểu của một box.

***16. Scrollbar trong CSS:***

- CSS cung cấp thuộc tính overflow để yêu cầu trình duyệt sẽ xử lý như thế nào khi nội dung tràn khỏi phần không gian đã cấp cho nó. Thuộc tính này có thể có các giá trị như sau:

+ visible: cho phép nội dung tràn qua phần không gian chứa nó.

+ hidden: phần nội dung sẽ chỉ hiển thị tới phần viền không gian chứa nó và không có thanh cuốn.

+ scroll: kích cỡ của phần tử không thay đổi, nhưng sẽ có thêm thanh cuốn (scroll bar) cho phép người dùng thực hiện scroll để xem hết phần nội dung.

+ auto: Thanh cuốn sẽ chỉ hiển thị khi nội dung lớn hơn phần không gian chứa nó và không hiển thị khi nội dung ít hơn phần không gian chứa nó.

<**html**>

<**head**>

</**head**>

<**style** type=*"text/css"*>

.scroll {

display: *block*;

border: *1px solid red*;

padding: *5px*;

margin-top: *5px*;

width: *600px*;

height: *50px*;

overflow: *scroll*;

}

.auto {

display: *block*;

border: *1px solid red*;

padding: *5px*;

margin-top: *5px*;

width: *600px*;

height: *50px*;

overflow: *auto*;

}

</**style**>

<**body**>

<**p**>Vi du gia tri scroll cua thuoc tinh overflow trong CSS:</**p**>

<**div** class=*"scroll"*>Tra loi bao chi hoi dau thang, Pho chu tich

thuong truc VFF Tran Quoc Tuan va HLV Nguyen Huu Thang tung de cap toi

kha nang mo cua cho cau thu nhap tich len tuyen. Theo ong Tran Quoc

Tuan, nhung cau thu nay neu dap ung duoc yeu cau ve chuyen mon do HLV

Nguyen Huu Thang dat rat hi deu co kha nang len tuyen.</**div**>

<**br** />

<**p**>Vi du gia tri auto cua thuoc tinh overflow trong CSS:</**p**>

<**div** class=*"auto"*>Tra loi bao chi hoi dau thang, Pho chu tich

thuong truc VFF Tran Quoc Tuan va HLV Nguyen Huu Thang tung de cap toi

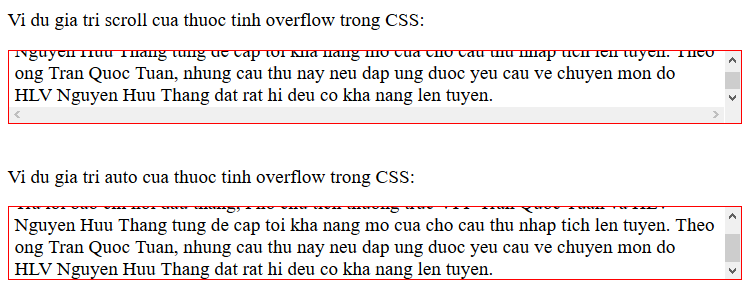
kha nang mo cua cho cau thu nhap tich len tuyen. Theo ong Tran Quoc

Tuan, nhung cau thu nay neu dap ung duoc yeu cau ve chuyen mon do HLV

Nguyen Huu Thang dat rat hi deu co kha nang len tuyen.</**div**>

</**body**>

</**html**>

=> Kết quả:

***17. Một số thuộc tính và thẻ nâng cao:***

*\*Thuộc tính visible:* Thuộc tính cho phép ẩn một phần tử nào đó. Sử dụng thuộc tính này cùng với JS có thể tạo ra những menu và layout đa tính năng. Thuộc tính này có thể nhận các giá trị:

+ visible: Hiển thị nội dung.

+ hidden: Ẩn nội dung.

+ collapse: Chỉ sử dụng với các hàng và cột trong bảng.

<**html**>

<**head**>

</**head**>

<**body**>

<**p**>Doan van nay la nhin thay nhu binh thuong.</**p**>

<**p** style="visibility: *hidden*;">Doan van nay la khong nhin thay (bi

an di).</**p**>

</**body**>

</**html**>

=> Kết quả:



*\*Pseudo Class và Pseudo Element:*

- active Pseudo class: được sử dụng trong CSS để thêm hiệu ứng đặc biệt cho phần tử đã được activate, có thể nhận bất kỳ giá trị màu nào.

- focus Pseudo class: được sử dụng để thêm các hiệu ứng đặc biệt cho một phần tử được focus.

- hover Pseudo class: được sử dụng để thêm các hiệu ứng đặc biệt cho một phần tử khi chuột di chuyển qua nó.

- link Pseudo class: được sử dụng để thêm các hiệu ứng đặc biệt cho một unvisited link, nhận bất kỳ giá trị màu hợp lệ nào.

- visited Pseudo class: được sử dụng để thêm hiệu ứng đặc biệt cho một visited link, nhận bất kỳ giá trị màu hợp lệ nào.

- first-child Pseudo class: được dùng để thêm hiệu ứng cho phần tử là phần tử con đầu tiên của một phần tử khác.

<**html**>

<**head**>

<**style** type=*"text/css"*>

div>p:first-child {

text-indent: *25px*;

}

</**style**>

</**head**>

<**body**>

<**div**>

<**p**>Doan van thu 1. Doan van nay se bi thut dau dong (thuoc tinh

text-indent).</**p**>

<**p**>Doan van thu 2. Doan van nay se khong bi thut dau dong.</**p**>

</**div**>

<**p**>Phan tu p phai la phan tu con DAU TIEN trong the div thi moi bi

tac dong boi thuoc tinh text-indent, cac doan van khac se khong bi anh

huong:</**p**>

<**div**>

<**h3**>Vi du khac</**h3**>

<**p**>Doan van dau tien trong the div. Doan van nay khong bi anh

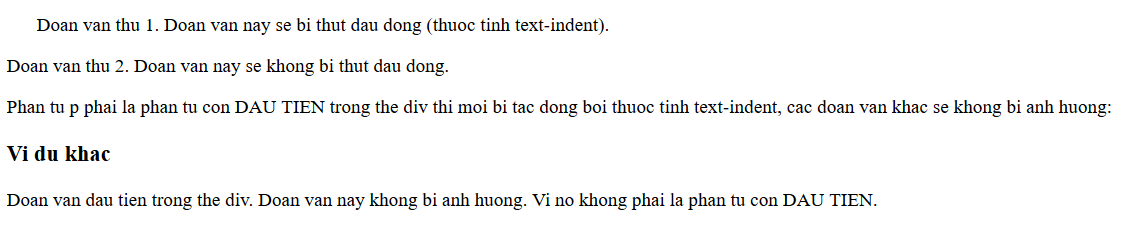
huong. Vi no khong phai la phan tu con DAU TIEN.</**p**>

</**div**>

</**body**>

</**html**>

=>Kết quả:



- lang Pseudo Class trong CSS được sử dụng để xác định một ngôn ngữ cho một phần tử nào đó.

<**html**>

<**head**>

<**style** type=*"text/css"*>

/\* Hai cach trich dan khac nhau cho tung loai ngon ngu \*/

:lang(en) {

quotes: *'"' '"' "'" "'"*;

}

:lang(fr) {

quotes: *"<<" ">>" "<" ">"*;

}

</**style**>

</**head**>

<**body**>

<**p**>

...<**q** lang=*"fr"*>Vi du mot doan van duoc trich dan</**q**>...

</**p**>

</**body**>

</**html**>

=> Kết quả:



- first-letter Pseudo class: được sử dụng để thêm style đặc biệt cho các chữ cái đầu tiên (ký tự đầu tiên) của đoạn văn trong một selector. Các thuộc tính của pseudo class này gồm có:

+ Nhóm thuộc tính font

+ Nhóm thuộc tính color

+ Nhóm thuộc tính background

+ Nhóm thuộc tính margin

+ Nhóm thuộc tính padding

+ Nhóm thuộc tính border

+ Thuộc tính text-decoration

+ Thuộc tính vertical-align

+ Thuộc tính text-transform

+ Thuộc tính line-height

+ Thuộc tính float

+ Thuộc tính clear

<**html**>

<**head**>

<**style** type=*"text/css"*>

p:first-letter {

font-size: *3em*;

}

p.normal:first-letter {

font-size: *10px*;

}

</**style**>

</**head**>

<**body**>

<**p** class=*"normal"*>Chu cai dau tien van se duoc hien thi theo binh

thuong va co Font Size la 10 px.</**p**>

<**p**>Chu cai dau tien cua doan van se co Font Size la 5em va co mau

den nhu trong CSS rule tren. Cac ky tu con lai trong doan van se van

hien thi binh thuong theo Font Size mac dinh. Day la vi du minh hoa

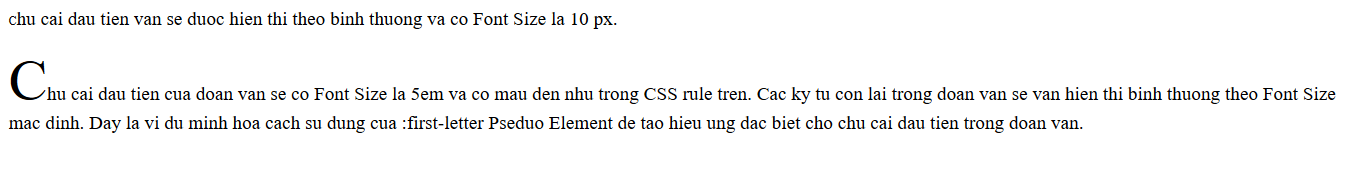
cach su dung cua :first-letter Pseduo Element de tao hieu ung dac biet

cho chu cai dau tien trong doan van.</**p**>

</**body**>

</**html**>

=> Kết quả:



- first-line pseudo class: Sử dụng để thêm các style đặc biệt cho hàng đầu tiên của đoạn văn bản.

- before và after pseudo class: Sử dụng để chèn thêm nội dung vào trước một phần tử nào đó:

<**html**>

<**head**>

<**style** type=*"text/css"*>

p:before {

display: *inline-block*;

width: *16px*;

height: *16px*;

margin-right: *5px*;

content: *""*;

background: *url("https://i.imgur.com/sRaUbMc.png") no-repeat 0 0*;

background-size: *100%*;

}

</**style**>

</**head**>

<**body**>

<**p**>Dong text nay se duoc dat truoc boi mot Bullet.</**p**>

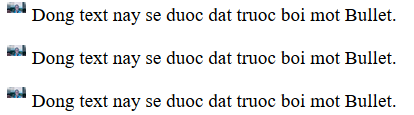
<**p**>Dong text nay se duoc dat truoc boi mot Bullet.</**p**>

<**p**>Dong text nay se duoc dat truoc boi mot Bullet.</**p**>

</**body**>

</**html**>

=> Kết quả:



- Các lưu ý khi sử dụng pseudo class trong CSS: a:hover phải theo sau a:link và a:visited nếu không sẽ không có hiệu quả. a:active phải theo sau a:hover nếu không sẽ không có hiệu quả.