**NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN**

*Trà Vinh, ngày ….. tháng …… năm ……*

**Giáo viên hướng dẫn**

*(Ký tên và ghi rõ họ tên)*

*Trà Vinh, ngày ….. tháng …… năm ……*

**Giáo viên hướng dẫn**

*(Ký tên và ghi rõ họ tên)*

**LỜI CẢM ƠN**

**NHẬN XÉT CỦA THÀNH VIÊN HỘI ĐỒNG**

*Trà Vinh, ngày ….. tháng …… năm ……*

**Thành viên hội đồng**

*(Ký tên và ghi rõ họ tên)*

MỤC LỤC

DANH MỤC HÌNH ẢNH – BẢNG BIỂU

# TÓM TẮT

# MỞ ĐẦU

# CHƯƠNG 1: TÌM HIỂU HTML, CSS VÀ JAVASCRIPT

## Ngôn ngữ HTML

### Tổng quan HTML

HTML (HyperText Markup Language hay "Ngôn ngữ Đánh dấu Siêu văn bản") là một [ngôn ngữ đánh dấu](https://vi.wikipedia.org/wiki/Ng%C3%B4n_ng%E1%BB%AF_%C4%91%C3%A1nh_d%E1%BA%A5u) được thiết kế ra để tạo nên các [trang web](https://vi.wikipedia.org/wiki/Website) với các mẩu thông tin được trình bày trên [World Wide Web](https://vi.wikipedia.org/wiki/World_Wide_Web). Cùng với [CSS](https://vi.wikipedia.org/wiki/CSS) và [JavaScript](https://vi.wikipedia.org/wiki/JavaScript), HTML tạo ra bộ ba nền tảng kỹ thuật cho [World Wide Web](https://vi.wikipedia.org/wiki/World_Wide_Web). HTML đã trở thành một chuẩn [Internet](https://vi.wikipedia.org/wiki/Internet) do tổ chức [World Wide Web Consortium](https://vi.wikipedia.org/wiki/W3C) (W3C) duy trì. Phiên bản chính thức mới nhất của HTML là HTML5.

HTML mô tả cấu trúc của một trang web. HTML bao gồm một tập hợp các phần tử. Các phần tử sẽ chỉ định cách hiển thị nội dung trên trình duyệt. Các phần tử được diễn tả bởi các thẻ (tag). Trình duyệt không hiển thị các thẻ, nhưng sử dụng chúng để hiển thị nội dung của trang web.

#### Trình duyệt web

Trình duyệt web hay Web Browser là một phần mềm ứng dụng được gọi tắt là trình duyệt, trình duyệt web giúp người dùng có thể nhìn thấy và tương tác với các văn bản, hình ảnh, nhạc, video, các trò chơi và nhiều thông tin khác xuất hiện trên một trang web bất kì.

Mỗi trình duyệt sẽ có một cách đọc và hiển thị HTML riêng, nên có thể là cùng một trang web nhưng khi truy cập bằng hai trình duyệt khác nhau ta sẽ thấy hai cách hiển thị khác nhau trên hai trình duyệt.

Các trình duyệt web phổ biến hiện nay như: Google Chrome, Cốc Cốc, Firefox, Safari, Internet Explorer...

#### Trình soạn thảo mã nguồn

Trình soạn thảo mã nguồn hay Code Editor là công cụ không thể thiếu trong việc thiết kế xây dựng website. Trình soạn thảo mã nguồn giúp chúng ta làm nổi bật code (Highlight code), tự động hoàn thiện (Auto complete), sửa lỗi (Debug) và các tính năng khác giúp cho việc soạn thảo mã nguồn được tiện lợi hơn, nhanh chóng hơn và dễ dàng tìm kiếm, sửa lỗi.

Một số trình soạn thảo hỗ trợ việc soạn thảo mã nguồn:

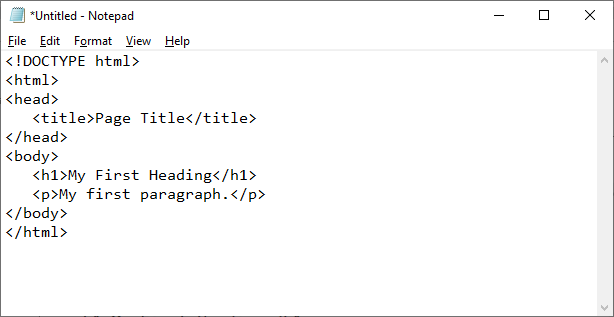
* Notepad
* Notepad++
* Sublime Text
* Visual Studio Code
* TextMate
* Komodo Edit

### Tạo một trang web đơn giản

Để tạo một trang web đơn giản, ta thực hiện các bước sau:

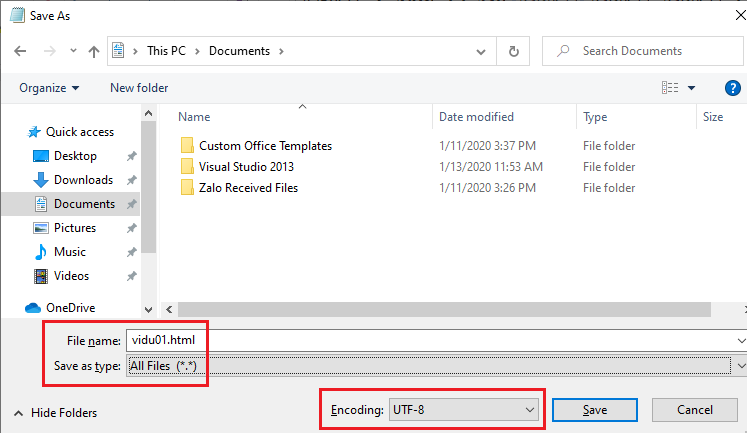
Bước 1: Ta sẽ sử dụng trình soạn thảo Notepad trên máy tính Window để xây dựng trang web. Vào **Menu Start >** gõ **Notepad >** nhấn **Enter**.

Bước 2: Nhập các dòng code như hình bên dưới.



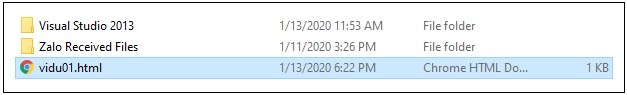
Hình : Soạn thảo mã nguồn tạo trang web đơn giản bằng Notepad.

Bước 3: Chọn **File > Save** lưu lại tập tin với phần mở rộng “**.html**” hoặc “**.htm**” và chọn **Encoding** là **UTF-8** như hình.



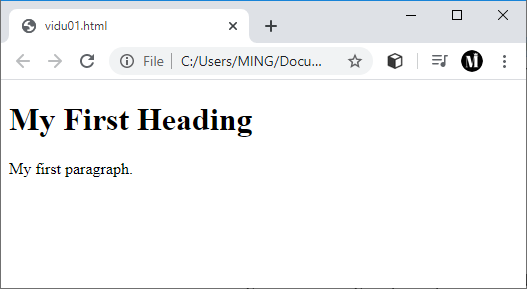
Hình : Lưu tập tin với định dạng html và UTF-8.

Bước 4: Kiểm tra xem chúng ta đã lưu đúng định dạng tập tin hay chưa. Nếu biểu tượng tập tin là một trình duyệt bất kì có trên máy tính thì chúng ta đã tạo thành công một trang web đơn giản.



Hình : Kiểm tra tập tin có được lưu đúng định dạng hay không.

Bước 5: Xem trang web bằng cách nhấn đúp chuột vào tập tin html và mở bằng trình duyệt mà chúng ta muốn.



Hình : Xem trang web trên trình duyệt.

### Cấu trúc trang HTML

<!DOCTYPE html>  
<html>  
<head>  
 <title>Page Title</title>  
</head>  
<body>  
 <h1>My First Heading</h1>  
 <p>My first paragraph.</p>  
</body>  
</html>

**Trong đó:**

* <!DOCTYPE html> là phần khai báo phiên bản HTML.
* <html> là thẻ bao bọc nội dung của trang HTML.
* <head> là thẻ khai báo thông tin của trang web.
* <title> là thẻ dùng để khai báo tiêu đề của trang web.
* <body> là thẻ chứa nội dung trang và được hiển thị trên trình duyệt.
* <h1> là thẻ dùng để khai báo độ lớn của tiều đề.
* <p> là thẻ khai báo đoạn văn bản.

### Các thẻ HTML cơ bản

Một thẻ HTML (hay còn gọi là một phần tử) thường bao gồm một thẻ **mở** và một thẻ **đóng**, nội dung được thêm vào giữa hai thẻ.

Ví dụ: **<tên\_thẻ>** Nội dung hiển thị... **</tên\_thẻ>**

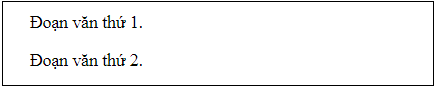
#### Thẻ đoạn văn (Paragraphs)

Trong HTML, thẻ **<p>** dùng để xác định các đoạn văn. HTML sẽ tự động cách dòng giữa các thẻ **<p>**.

<p> Đây là đoạn văn thứ 1. </p>

<p> Đây là đoạn văn thứ 2. </p>

Kết quả khi hiển thị trên trình duyệt:



Hình : Xác định các đoạn văn với thẻ <p>.

#### Thẻ tiêu đề (Heading)

Tiêu đề trong HTML được định nghĩa với các thẻ **<h1>** đến **<h6>**.

<h1>Tiêu đề 1</h1>

<h2>Tiêu đề 2</h2>

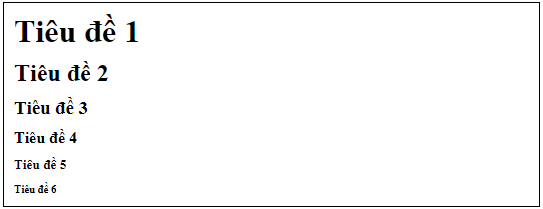
<h3>Tiêu đề 3</h3>

<h4>Tiêu đề 4</h4>

<h5>Tiêu đề 5</h5>

<h6>Tiêu đề 6</h6>

Kết quả khi hiển thị trên trình duyệt:



Hình : Sử dụng thẻ <h1> đến <h6> để xác định kích thước tiêu đề.

#### Thẻ liên kết (Links)

Dùng thẻ **<a>** để chèn một liên kết trong trang HTML. Địa chỉ liên kết được gán vào thuộc tính **href** trong thẻ **<a>**.

<p> Liên kết trong HTML:

<a href="https://www.w3schools.com"> w3schools.com </a>

</p>

Kết quả khi hiển thị trên trình duyệt:



Hình : Thẻ <a> dùng để chèn liên kết vào trang web.

#### Các thẻ định dạng văn bản (Formatting)

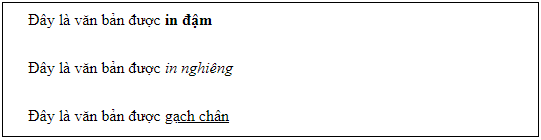
Các thẻ **<b>**, **<i>**, **<u>** được dùng để định dạng cho văn bản như: bôi đậm, in nghiêng, gạch chân...

<p>Đây là văn bản được <b>in đậm</b></p>

<p>Đây là văn bản được <i>in nghiêng</i></p>

<p>Đây là văn bản được <u>gạch chân</u></p>

Kết quả khi hiển thị trên trình duyệt:



Hình : Sử dụng các thẻ định dạng để làm nổi bật văn bản.

Ngoài ra còn một số thẻ định dạng có công dụng giống như thẻ **<b>**, **<i>**, nhưng lại khác nhau về mặt ý nghĩa.

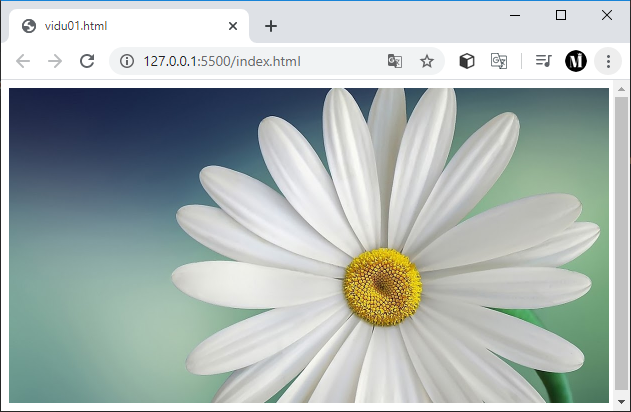
Thẻ **<strong>** và thẻ **<em>** có chức năng in đậm và in nghiêng văn bản như thẻ **<b>** và thẻ **<i>** nhưng về mặt ý nghĩa thẻ **<strong>**, **<em>** được dùng để nhấn mạnh tầm quan trọng của văn bản về mặt ngữ nghĩa còn thẻ **<b>**, **<i>** không có bất kì tầm quan trọng nào.

#### Thẻ hình ảnh (Images)

Để chèn hình ảnh vào trang web, ta sử dụng thẻ **<img>** với thuộc tính **src** để trỏ đến nơi lưu trữ hình ảnh cần chèn vào web.

<img src="images/flower.jpg" alt="Hình ảnh bông hoa">

Kết quả khi hiển thị trên trình duyệt:



Hình : Chèn hình ảnh vào trang web bằng thẻ <img>.

#### Thẻ danh sách (Lists)

Danh sách trong HTML có 2 loại:

* Danh sách có sắp xếp được định nghĩa với thẻ **<ol>**.
* Danh sách không sắp xếp được định nghĩa với thẻ **<ul>**.

<ol>

<li>item 1</li>

<li>item 2</li>

<li>item 3</li>

</ol>

<ul>

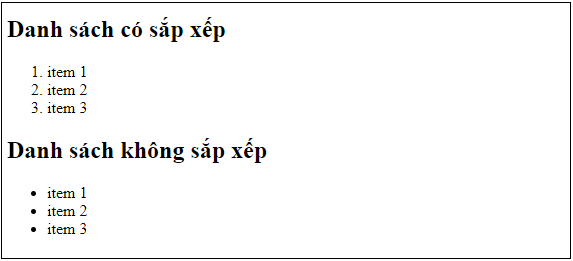
<li>item 1</li>

<li>item 2</li>

<li>item 3</li>

</ul>

Kết quả khi hiển thị trên trình duyệt:



Hình : Tạo danh sách với thẻ <ol> và <ul> trong HTML.

#### Thẻ tạo bảng (Table)

Bảng trong HTML được xác định bằng thẻ **<table>**.

Mỗi hàng trong bảng được định nghĩa với thẻ **<tr>**, mỗi cột trong bảng được định nghĩa với thẻ **<td>.**

<table>

<tr>

<td> Dòng 1 Cột 1 </td>

<td> Dòng 1 Cột 2 </td>

</tr>

<tr>

<td> Dòng 2 Cột 1 </td>

<td> Dòng 2 Cột 2 </td>

</tr>

<tr>

<td> Dòng 3 Cột 1 </td>

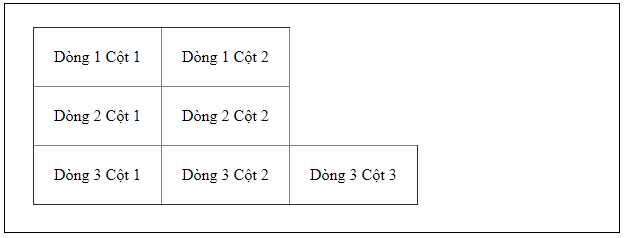
<td> Dòng 3 Cột 2 </td>

<td> Dòng 3 Cột 3 </td>

</tr>

</table>

Kết quả khi hiển thị trên trình duyệt:



Hình : Tạo bảng trình bày dữ liệu bằng thẻ <table>.

Thuộc tính class

Class là thuộc tính HTML được sử dụng để xác định các kiểu bằng nhau cho các phần tử có cùng tên lớp.

Tất cả các thành phần HTML có cùng class sẽ có cùng kiểu.

Các thẻ khác nhau có thể có cùng tên class

<style>

.name1{

background-color: red;

}

</style>

<body>

<div class="name1">

<h1>Thẻ h1 trong class 1 có tên name1</h1>

</div>

<div class="name1">

<h1>Thẻ h1 trong class 2 có tên name1</h1>

</div>

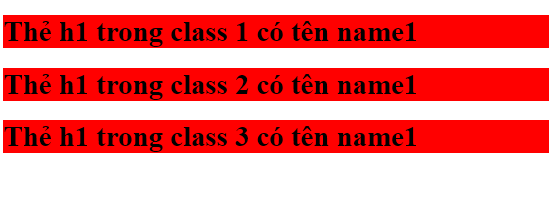
<div class="name1">

<h1>Thẻ h1 trong class 3 có tên name1</h1>

</div>

</body>

## Kết quả hiển thị trên trình duyệt:



## Thuộc tính id

Thuộc tính **id** chỉ định một id duy nhất cho một phần tử HTML.

Để chọn một **id** cụ thể, ta viết **#** sau đó là **id của phần tử**.

<style>

#name1{

color: red;

background-color:green;

}

</style>

<body>

<div id="name1">

<h1>Nội dung trong thẻ id có tên là name1</h1> </div>

</body>

## Kết quả hiển thị trên trình duyệt:



## Iframe

Một iframe HTML được xác định bằng thẻ iframe.

Các thuộc tính sử dụng cho iframe: height, width, boder,…

<h1>HTML iframe</h1>

<iframe src="" height="200" width="200"></iframe>

Kết quả hiển thị trên trình duyệt:



Đường dẫn tệp HTML

<img src="picture.jpg"> : ảnh nằm trong cùng tệp với trang HTML.

<img src="images/picture.jpg"> : nằm trong thư mục images trong thư mục hiện tại.

<img src="/images/picture.jpg"> : nằm trong thư mục images trong thư mục gốc.

<img src="../picture.jpg"> : nằm trong thư mục tăng một cấp với thư mục hiện tại.

Đường dẫn tệp mô tả vị trí của tệp trong cấu trúc thư mục của một trang web.

Đường dẫn tệp được sử dụng khi liên kết với các tệp bên ngoài như:

+ Trang web

+ Hình ảnh

+ Javascripts

Đường dẫn tệp tuyệt đối:

<h1>Đường dẫn tệp tuyệt đối</h1>

<img src="https://www.w3schools.com/images/picture.jpg" alt="Moutain">

Kết quả hiển thị trên trình duyệt:



Đường dẫn tệp tương đối nằm trong thư mục hiện tại

<h1>Đường dẫn tệp tương đối nằm trong thư mục hiện tại</h1>

<img src="picture\_it.jpg" alt="IT">



Form

Phần tử <form> HTML xác định một biểu mẩu được sử dụng để thu thập dữ liệu nhập vào của người dùng

Phần tử <input>

+ Là phần tử hình thức quan trọng nhất.

+ Có thể hiển thị theo nhiều cách tùy vào loại thuộc tính.

<input type="text">: xác định nhập văn bản.

<input type="radio">: xác định nút radio (chọn một trong nhiều lựa chọn).

<input type="submit">: xác định nút gửi.

Nhập văn bản

**<input type="text">** xác định nhập một dòng cho **văn bản**.

<form>

Họ:<br>

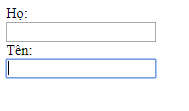
<input type="text" name="ho"><br>

Tên:<br>

<input type="text" name="ten">

</form>

Kết quả hiển thị trên trình duyệt:



*Lưu ý:* chiều rộng mặc định của văn bản là 20 ký tự.

Nút Radio

**<input type="radio">** xác định một nút **radio .**

<h1>Nút Radio</h1>

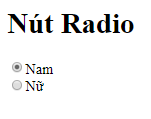
<form>

<input type="radio" name="gioiTinh" value="nam" checked="">Nam<br>

<input type="radio" name="gioiTinh" value="nu">Nữ

</form>

Kết quả hiển thị trên trình duyệt:



Nút gửi

**<input type="submit">** định nghĩa nút gửi dữ liệu biểu mẩu đến trình xử lý biểu mẩu.

Trình xử lý biểu mẫu thường là một trang máy chủ có tập lệnh để xử lý dữ liệu đầu vào.

Trình xử lý biểu mẫu được chỉ định trong thuộc tính hành động của biểu mẫu :

<h1>Nút gửi</h1>

<form>

Họ:<br>

<input type="text" name="ho" value="Khánh"><br>

Tên:<br>

<input type="text" name="ten" value="Duy"><br><br>

<input type="submit" value="Submit">

</form>

Kết quả hiện thị trên trình duyệt:



Nhóm dữ liệu biểu mẩu với **<fieldset>**

Phần **<fieldset>**tử được sử dụng để nhóm dữ liệu **liên quan trong một biểu mẫu**.

Phần **<legend>**tử xác định chú thích cho **<fieldset>** phần tử.

<form>

<fieldset>

<legend>Thông tin cá nhân:</legend>

Tên:<br>

<input type="text" name="ten" value="Duy"><br>

Họ:<br>

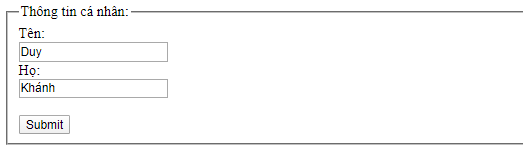
<input type="text" name="ho" value="Khánh"><br><br>

<input type="submit" value="Submit">

</fieldset>

</form>

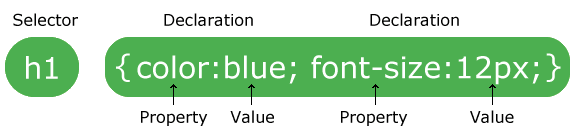
Kết quả hiện thị trên trình duyệt:



## Bảng định kiểu CSS

Cú pháp CSS

Bao gồm các Selector và Declaration:



## Declaration: chứa một hoặc nhiều declaretion được phân tách bằng dấu chấm phẩy.

Mỗi Delaration bao **gồm tên thuộc tính (property) CSS và một gía trị (value) được phân tách bằng dấu hai chấm.**

**Mỗi Delaration được kết thúc bằng dấu chấm phẩy và các khối khai báo được bao quanh bởi cặp dấu ngoặc nhọn.**

<style>

h1 {

color: red;

text-align: center;

}

</style>

</head>

<body>

<h1>Hello World!</h1>

</body>

Kết quả hiển thị trên trình duyệt:



CSS Selectors

CSS Selectors được sử dụn để tìm hoặc chọn các thành phần HTML mà bạn muốn tạo kiểu.

Các loại CSS Selectors:

* **Simple selectors** : chọn dựa trên **name, id, class**.
* **Compinator selectors**: chọn dựa trên các mối quan hệ cụ thể.
* **Pseudo-class selectors**: chọn dựa trên một trạn thái nhất định.
* **Pseudo-elements selectors**: chọn và định kiểu một phần của phần tử.
* **Attribute selectors**: chọn thành phần dựa trên ía trị thuộc tính hoặc thuộc tính.

CSS **name** Selectors

Tất cả các phần tử <p> trên trang sẽ được căn giữa, với màu văn bản màu đỏ:

<style>

p {

text-align: center;

color: red;

}

</style>

</head>

<body>

<p>Xin chào</p>

<p id="para1">Hello</p>

<p>HelloWord</p>

</body>

Kết quả hiển thị trên trình duyệt:



CSS **id** Selectors

**Selector id** sử dụn thuộc tính id của một phần tử HTML để chọn một phần tử cụ thể.

Id của một thành phần là duy nhất tron một tran, vì vậy **selector id** được sử dụn để chọn một thành phần duy nhất.

Để chọn một phần tử có id cụ thể, hãy viết ký tự #, theo sau là id của phần tử.

<style>

#name1 {

color: red;

}

</style>

</head>

<body>

<p>Xin chào</p>

<p id="name1">Hello</p>

<p>HelloWord</p>

</body>

Kết quả hiển thị trên trình duyệt:



Lưu ý: Tên id khôn thể bắt đầu bằn số.

CSS **class** Selectors

Class selectors chọn các thành phần HTML với thuộc tính cụ thể.

Để chọn các phần tử với một class cụ thể, ta viết một ký tự (.), theo sau là tên class.

<style>

.name1 {

color: blue;

}

</style>

</head>

<body>

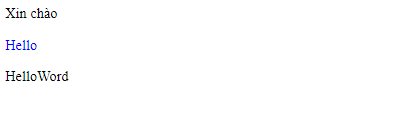
<p>Xin chào</p>

<p class="name1">Hello</p>

<p>HelloWord</p>

</body>

Kết quả hiển thị trên trình duyệt:



Các phần tử HTML cụ thể sẽ bị ảnh hưởn bởi một class.

VD: Các phần tử tron thẻ <p> với class = “name2“ sẽ có màu xanh.

<style>

.name2 {

color: blue;

}

</style>

</head>

<body>

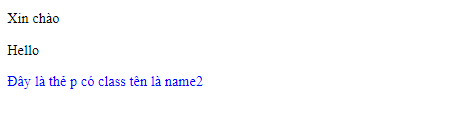
<p>Xin chào</p>

<p class="name1">Hello</p>

<p class="name2">Đây là thẻ p có class tên là name2</p>

</body>

Kết quả hiển thị trên trình duyệt:



Các phần tử HTML cũn có thể tham chiếu đến nhiều hơn một class.

<style>

p.name1 {

color: red;

}

p.name2 {

font-size: 300%;

}

</style>

</head>

<body>

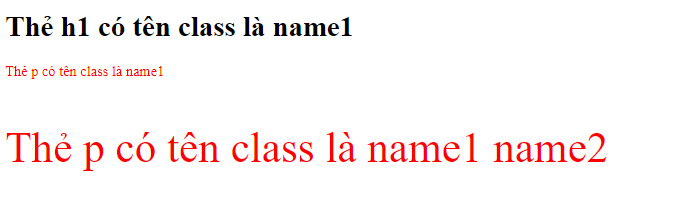
<h1 class="name1">Thẻ h1 có tên class là name1</h1>

<p class="name1">Thẻ p có tên class là name1</p>

<p class="name1 name2">Thẻ p có tên class là name1 name2</p>

</body>

Kết quả hiển thị trên trình duyệt:



Insert CSS

Có 3 cách để chèn CSS:

* External CSS
* Internal CSS
* Inline CSS

External CSS

Với định kiểu bên noài (External CSS) ta có thể thay đổi iao diện của toàn bộ tran web bằn cách thay đổi chỉ một tệp

Mỗi tran HTML phải bao ồm một tham chiếu đến tệp định kiểu bên noài nằm tron phần tử **<link>** bên tron phần **<head>**

<head>

<meta charset="UTF-8">

<title>Howto CSS</title>

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

</head>

<body>

<h1>Đây là thẻ h1</h1>

<p>Đây là thẻ p</p>

</body>

Định kiểu CSS bên noài được viết bất kì tron trình soạn thảo văn bản nào và phải được lưu với phần mở rộn là **.css**

Tệp **.css**  bên noài khôn được chứa bất kì thẻ HTML nào

Tệp “css.css”

body {

background-color: lightblue;

}

h1 {

color: red;

margin-left: 20px;

}

Kết quả hiển thị trên trình duyệt



Lưu ý: Khôn thêm khoản trắn iữa ía trị thuộc tính và đơn vị (chẳn hạn như: margin-left: 20 px;). Cách chính xác là margin-left: 20px;

Internal CSS (CSS nội bộ)

Được sử dụng nếu một trang HTML duy nhất có một kiểu duy nhất.

Kiểu bên trong được xác định bên trong phần tử **<style>,** bên trong phần **<head>**.

<head>

<meta charset="UTF-8">

<title>Howto CSS</title>

<style>

h1{

color: red;

}

p{

margin-left: 20px;

color: blue;

}

</style>

</head>

<body>

<h1>Đây là thẻ h1</h1>

<p>Đây là thẻ p</p>

</body>

Kết quả hiển thị trên trình duyệt:



Inline CSS (CSS nội tuyến)

Được sử dụng để áp dụng một kiểu duy nhất cho một phần tử.

Để sử dụng kiểu nội tuyến, hãy thêm thuộc tính style cho thành phần có liên quan. Thuộc tính style có thể chứa bất kỳ thuộc tính CSS nào.

<h1 style="color: blue; margin-left: 20px;">Đây là thẻ h1</h1>

<p style="color: green;">Đây là thẻ p</p>

Kết quả hiển thị trên trình duyệt:



Màu CSS

Màu sắc được chỉ định bằn cách sử dụn tên mà, các ía trị RB, HEX, HSL, RBA, HSLA.

Tên màu

Tron CSS một màu có thể được chỉ định bằn cách sử dụn tên màu

<h1 style="background-color:Tomato;">Tomato</h1>

<h1 style="background-color:Orange;">Orange</h1>

<h1 style="background-color:DodgerBlue;">DodgerBlue</h1>

<h1 style="background-color:MediumSeaGreen;">MediumSeaGreen</h1>

<h1 style="background-color:Gray;">Gray</h1>

<h1 style="background-color:SlateBlue;">SlateBlue</h1>

<h1 style="background-color:Violet;">Violet</h1>

<h1 style="background-color:LightGray;">LightGray</h1>

Kết quả hiển thị trên trình duyệt:



Màu cho văn bản

<h1 style="color: red;">Hello Word</h1>

Kết quả hiển thị trên trình duyệt:



Màu cho khun viền

<h1 style="border: 2px solid blue;">Hello Word</h1>

Kết quả hiển thị trên trình duyệt:



Màu dựa trên ía trị Hex

Một màu có thể được chỉ định bằng cách sử dụng giá trị thập lục phân ở dạng:

# rrggbb.

Trong đó rr (đỏ), gg (xanh lá cây) và bb (xanh lam) là các giá trị thập lục phân trong khoảng từ 00 đến ff (giống như số thập phân 0-255).

<h1 style="color:#ffc747;">Hello Word</h1>

Kết quả hiển thị trên trình duyệt:



Nền CSS

Các thuộc tính nền CSS được sử dụn để xác định hiệu ứn nền cho các phần tử.

* background-color: màu nền
* background-image: ảnh nền
* background-repeat: ảnh nền lặp lại
* background-attachment: đính kèm nền
* background-position: vị trí nền

Màu nền (background-color)

<style>

body{

background-color: lightblue;

}

</style>

</head>

<body>

<h1>Hello Word</h1>

</body>

Kết quả hiển thị trên trình duyệt:



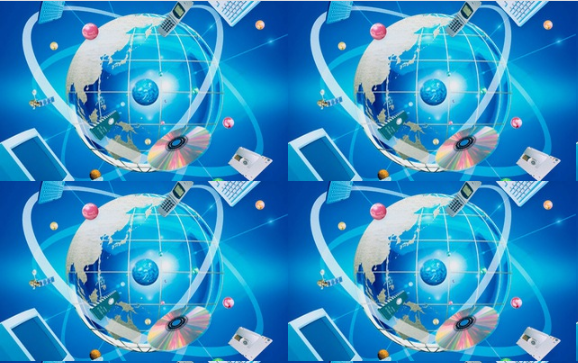
Ảnh nền

body{

background-image: url(picture\_it.jpg);

}

Kết quả hiển thị trên trình duyệt:



Lặp lại nền

Theo mặc định, ảnh nền được lặp lại theo cả chiều nan (repeat-x) và chiều dọc(repeat-y).

body{

background-image: url(picture\_it.jpg);

background-repeat: repeat-x;

}

Kết quả hiển trị trên trình duyệt:



Khôn lặp lại ảnh nền

body{

background-image: url(picture\_it.jpg);

background-repeat: no-repeat;

}

Kết quả hiển thị trên trình duyệt:



Vị trí ảnh nền

body{

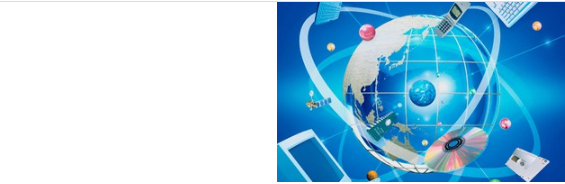
background-image: url(picture\_it.jpg);

background-repeat: no-repeat;

background-position: right top;

}

Kết quả hiển thị trên trình duyệt:



Đườn viền (Boder)

Các boder CSS cho phép ta chỉ định kiểu dán, chiều rộn và màu sắc của đườn viền của một phần tử

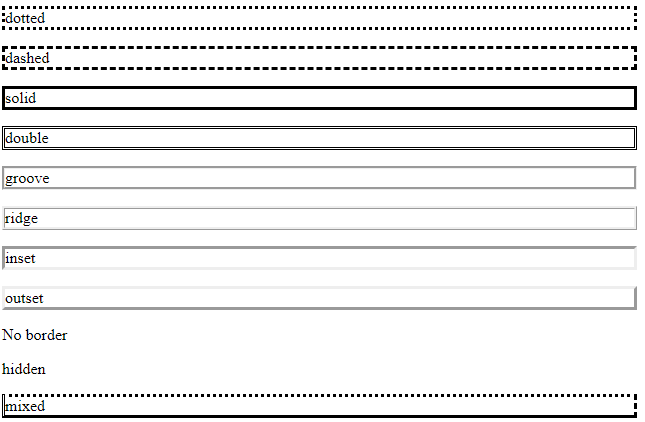
Kiểu đườn viền (boder-style)

Các đườn viền qui định cụ thể nhữn loại đườn viền để hiển thị.

* **dotted :** Xác định đường viền chấm.
* **dashed :** Xác định đường viền nét đứt.
* **solid :** Xác định đường viền nét liền.
* **double :** Xác định đường viền đôi.
* **groove :** Xác định đường viền có rãnh 3D. Hiệu ứng phụ thuộc vào giá trị màu viền.
* **ridge :** Xác định đường viền có viền 3D. Hiệu ứng phụ thuộc vào giá trị màu viền.
* **inset :** Xác định đường viền chèn 3D. Hiệu ứng phụ thuộc vào giá trị màu viền.
* **outset :** Xác định đường viền đầu 3D. Hiệu ứng phụ thuộc vào giá trị màu viền.
* **none :** Xác định không đườn viền.
* **hidden :** Xác định đường viền ẩn.

p.dotted {border-style: dotted;}  
p.dashed {border-style: dashed;}  
p.solid {border-style: solid;}  
p.double {border-style: double;}  
p.groove {border-style: groove;}  
p.ridge {border-style: ridge;}  
p.inset {border-style: inset;}  
p.outset {border-style: outset;}  
p.none {border-style: none;}  
p.hidden {border-style: hidden;}  
p.mix {border-style: dotted dashed solid double;}

Kết quả hiển thị trên trình duyệt:



Các thuộc tính có thể kết hợp với đườn viền:

**border-width:** chiều rộn của đườn viền.

p.solid {border-style: solid; border-width: 5px;}

Kết quả hiển thị trên trình duyệt:



**boder-color:** màu của đườn viền.

p.solid {border-style: solid; border-width: 5px; border-color: red;}

Kết quả hiển thị trên trình duyệt:



Các cạnh riên lẻ của đườn viền

p {

border-top-style: dotted;

border-right-style: solid;

border-bottom-style: dotted;

border-left-style: solid;

}

Kết quả hiển thị trên trình duyệt:



Thuộc tính Margin

Là khoảng cách giữa một phần tử với các phần tử khác trong trang web.

p {

border: 1px solid black;

margin-top: 100px;

margin-bottom: 100px;

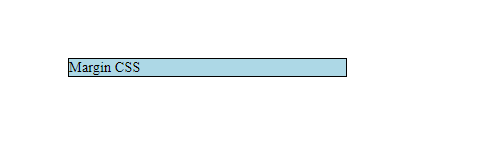
margin-right: 150px;

margin-left: 80px;

background-color: lightblue;

}

Kết quả hiển thị trên trình duyệt:



Cách viết ngắn mã

Nếu thuộc tính **margin** có bốn giá trị:

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

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

p {

margin: 25px 50px 75px 100px;

}

Kết quả hiển thị trên trình duyệt:

Nếu thuộc tính **margin** có ba giá trị:

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

* **lề trên** là 25px
* **lề phải, trái** là 50px
* **lề dưới** là 75px

p {  
  margin: 25px 50px 75px;  
}

Nếu thuộc tính **margin** có hai giá trị:

**margin**: 25px 50px;

* **lề trên, dưới** là 25px
* **lề phải, trái** là 50px

p {  
  margin: 25px 50px;  
}

Nếu thuộc tính **margin** có một giá trị:

**margin**: 25px;

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

p {  
  margin: 25px;  
}

**Padding**

Là thuộc tính để tạo khôn ian xun quanh nội dun của một phần tử, bên tron bất kỳ đườn viền được xác định nào.

div {

border: 1px solid black;

background-color: lightblue;

padding-top: 50px;

padding-right: 30px;

padding-bottom: 50px;

padding-left: 80px;

}

Kết quả hiển thị trên trình duyệt:



Nếu thuộc tính**padding** có bốn giá trị:

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

* **hàng trên** là 25px
* **bên phải** là 50px
* **dưới** là 75px
* **bên trái** là 100px

div {  
  padding: 25px 50px 75px 100px;  
}

Nếu thuộc tính**padding** có ba giá trị:

**padding**: 25px 50px 75px;

* **hàng trên** là 25px
* **bên phải, trái** là 50px
* **dưới** là 75px

div {  
  padding: 25px 50px 75px;  
}

Nếu thuộc tính**padding** có hai giá trị:

**padding**: 25px 50px;

* **hàng trên, dưới** là 25px
* **bên phải, trái** là 50px

div {  
  padding: 25px 50px;  
}

Nếu thuộc tính**padding** có một giá trị:

**padding**: 25px;

* **hàng trên, dưới, trái, phải** là 25px

div {  
  padding: 25px;  
}

## Ngôn ngữ Javascript

# DANH MỤC TÀI LIỆU THAM KHẢO