

## WEB TĨNH – MỤC LỤC BÀI HỌC

<b>A. NGÔN NGỮ HTML .....</b>	<b>10</b>
<b>I. Một số khái niệm.....</b>	<b>10</b>
1. Trang web (Webpage) .....	10
2. Hyperlink .....	10
3. Website .....	10
4. Browser (Trình duyệt web).....	10
5. WebServer .....	10
6. Http .....	11
7. Web tĩnh.....	11
8. Web động.....	11
<b>II. Các loại trang web trong 1 website .....</b>	<b>12</b>
1. Trang chủ (Homepage) .....	12
2. Trang chuyên đề .....	12
3. Trang đơn vị.....	13
4. Trang phản hồi (Feedback) .....	13
5. Trang web động .....	13
<b>III. Các thành phần thường có trong 1 trang web .....</b>	<b>14</b>
1. Chữ.....	14
2. Hình ảnh.....	14
3. Banner .....	14
4. Logo .....	14
5. Hit Counter .....	14
6. Search form.....	14
7. Navigator (Menu) .....	14
8. Multimedia.....	14
<b>IV. Giới thiệu HTML .....</b>	<b>15</b>
<b>V. Cấu trúc của một trang web.....</b>	<b>15</b>
<b>VI. Soạn thảo trang web .....</b>	<b>16</b>
1. Mở Notepad++ và gõ nội dung sau .....	16

2. Xem Source HTML của trang .....	17
3. Thêm lệnh xuống hàng giữa Họ tên và Email .....	17
4. Đưa hình (h.jpg) vào trang web: .....	17
5. Chỉnh kích thước và title, cách canh và đường viền cho hình: .....	17
6. Thêm liên kết vào trang (liên kết nằm trong tag a) .....	18
<b>VII. Các tag HTML căn bản .....</b>	<b>18</b>
<b>B. CĂN BẢN VỀ DREAMWEAVER .....</b>	<b>22</b>
<b>I. Giới thiệu về Dreamweaver .....</b>	<b>22</b>
<b>II. Thao tác với trang .....</b>	<b>22</b>
1. Tạo trang web .....	22
a. Tạo trang HTML trống .....	22
b. Tạo trang PHP .....	22
2. Xem thử trang web .....	22
3. Chuyển chế độ làm việc .....	22
a. Chế độ Code .....	22
b. Chế độ Design .....	22
c. Chế độ Split .....	23
d. Chuyển chế độ của trang .....	23
4. Lưu trang web .....	23
5. Mở trang .....	23
6. Đổi tên file .....	23
<b>III. Màn hình Dreamweaver .....</b>	<b>24</b>
1. Toolbar Document .....	24
2. Thanh Properties .....	24
3. Thanh Tag .....	24
4. Panel group .....	25
a. Insert bar .....	25
b. Css bar .....	25
c. Css bar .....	25
d. Dom bar .....	25
<b>C. CHÈN THÔNG TIN VÀO TRANG .....</b>	<b>26</b>
<b>I. Paragraph .....</b>	<b>26</b>

<b>II. Hình ảnh.....</b>	<b>26</b>
1. Các loại hình ảnh chèn vào vào trang.....	26
2. Chèn hình vào trang.....	27
3. Chỉnh thuộc tính của hình.....	27
<b>III. Liên kết.....</b>	<b>28</b>
1. Các loại địa chỉ .....	28
a. Địa chỉ tuyệt đối .....	28
b. Địa chỉ tương đối.....	28
c. Địa chỉ gốc (Ký hiệu là /) .....	29
2. Tạo liên kết .....	30
<b>IV. Chèn phim từ Youtube.....</b>	<b>30</b>
<b>V. Chèn bản đồ vào trang.....</b>	<b>32</b>
1. Chèn bản đồ .....	32
2. Tìm vĩ độ, kinh độ của 1 địa điểm bất kỳ .....	34
<b>VI. Table.....</b>	<b>35</b>
1. Chèn table .....	35
2. Các tag trong một table.....	36
<b>VII. Danh sách.....</b>	<b>37</b>
1. UnOdered list.....	37
2. Ordered list .....	37
3. Definition list .....	38
<b>VIII. Các tag quan trọng: div, span, iframe .....</b>	<b>38</b>
1. Tag DIV .....	38
2. Tag Span .....	39
3. Tag Heading.....	39
4. Tag nav .....	39
5. Tag header .....	39
6. Tag Footer.....	39
7. Tag main .....	40
8. Tag section.....	40
<b>D. CASCADING STYLE SHEET (CSS) .....</b>	<b>41</b>
<b>I. Giới thiệu .....</b>	<b>41</b>

<b>II. Tạo Style.....</b>	<b>41</b>
1. Định dạng cho 1 tag trong toàn trang .....	41
a. Tên style .....	41
b. Ví dụ: .....	41
2. Định dạng cho 1 tag cụ thể có tên riêng .....	42
a. Tên style .....	42
b. Ví dụ: .....	42
3. Định dạng cho tag bên trong 1 đối tượng có tên riêng .....	42
a. Tên style .....	42
b. Ví dụ .....	42
4. Class.....	42
a. Tên style .....	43
b. Ví dụ .....	43
c. Set Class .....	43
<b>III. Nhúng file css bên ngoài vào trang web .....</b>	<b>43</b>
<b>IV. Định dạng trực tiếp css trong tag html .....</b>	<b>43</b>
<b>V. Các thuộc tính CSS .....</b>	<b>43</b>
1. Các thuộc tính thường dùng để định dạng text.....	43
2. Các thuộc tính thường dùng để định dạng background.....	44
3. Các thuộc tính thường dùng để định dạng cho khối chữ .....	44
4. Các thuộc tính thường dùng để định dạng 1 khối.....	45
5. Các thuộc tính thường dùng để định dạng đường viền.....	45
6. Các thuộc tính định dạng danh sách list .....	46
7. Các thuộc tính định dạng vị trí .....	46
<b>VI. Định dạng liên kết.....</b>	<b>46</b>
<b>E. HOSTING VÀ DOMAIN.....</b>	<b>47</b>
<b>I. Khái niệm .....</b>	<b>47</b>
1. Hosting.....	47
2. Domain.....	47
<b>II. Tìm mua domain và hosting.....</b>	<b>47</b>
<b>III. Những điều phải biết khi mua hosting .....</b>	<b>47</b>
1. Những điều phải biết.....	47

2. Quản trị hosting .....	48
3. Upload website lên mạng internet .....	49
<b>IV. Mua domain .....</b>	<b>49</b>
<b>F. FORM.....</b>	<b>50</b>
<b>I. Chèn form.....</b>	<b>50</b>
<b>II. Các thuộc tính của Form.....</b>	<b>51</b>
<b>III. Các thành phần trên Form .....</b>	<b>52</b>
1. Text field.....	52
2. Radio group .....	52
3. Check Box .....	53
4. List/Menu.....	53
5. File field.....	54
6. Text area .....	54
7. Button.....	54
8. Image Field .....	55
9. Hidden field .....	55
<b>G. JAVASCRIPT .....</b>	<b>56</b>
<b>I. Giới thiệu .....</b>	<b>56</b>
<b>II. Viết mã Javascript vào trang .....</b>	<b>56</b>
1. Viết trực tiếp trong trang với tag script .....	56
2. Viết mã javascript trong file riêng .....	56
a. Tạo file javascript .....	56
b. Liên kết file js đến trang html .....	56
3. Sử dụng javascript trong sự kiện của các tag .....	57
a. Đưa chuột vào hình trái tim, sẽ đổi thành hình khác, chuột ra trở lại hình cũ .....	57
b. Nhấp chuột vào textfield tìm kiếm, chữ tìm kiếm trong đó sẽ biến mất.....	57
4. Lab sử dụng một số javascript hữu dụng.....	57
5. Website javascript.....	58
<b>III. Lập trình javascript .....</b>	<b>58</b>
1. Các sự kiện cơ bản trong Javascript.....	58
2. Biến .....	58
3. Các phép toán trên các biến.....	58

4. Phát biểu if...else .....	59
5. Vòng lặp for.....	59
6. Vòng lặp while .....	59
7. Vòng lặp do...while .....	60
8. Truy xuất đối tượng trong trang web .....	60
9. Hàm trong javascript .....	61
10. Đối tượng window .....	61
11. Đối tượng document .....	62
12. Đối tượng screen.....	62
13. Đối tượng history .....	63
14. Đối tượng location .....	63
15. Đối tượng Navigator.....	64
<b>H. JQUERY .....</b>	<b>65</b>
<b>I. Giới thiệu .....</b>	<b>65</b>
<b>II. Chèn jQuery vào trang .....</b>	<b>65</b>
<b>III. Chọn phần tử trong trang.....</b>	<b>65</b>
1. jQuery Selectors.....	66
2. jQuery Attribute Selectors .....	66
<b>III. Một số hàm thường dùng trong JQuery .....</b>	<b>66</b>
<b>IV. Hiệu ứng trong jquery .....</b>	<b>68</b>
<b>V. Sự kiện trong JQuery.....</b>	<b>69</b>
<b>VI. JQuery Ajax .....</b>	<b>70</b>
<b>VII. Một số Plugin JQuery hữu dụng .....</b>	<b>71</b>
<b>I. HTML5 CSS3, RESPONSIVE .....</b>	<b>72</b>
<b>I. HTML5 .....</b>	<b>72</b>
1. Những gì mới trong html5 .....	72
2. Cấu trúc 1 trang html5 .....	72
3. Những element mới trong html5 và những tag bị loại bỏ .....	72
4. HTML5 Form Elements .....	73
a. <datalist> .....	73
b. <output> .....	74

5. Các kiểu input mới trong HTML5 .....	74
6. Các thuộc tính quy định giá trị nhập; .....	74
<b>II. CSS3 .....</b>	<b>75</b>
1. Rounded Corners .....	75
2. Background Image .....	76
3. Gradients .....	76
4. Shadow .....	77
5. Text.....	77
6. Web Fonts.....	78
a. Dùng @font-face .....	78
b. Dùng @import.....	79
7. 2D Transforms.....	79
8. 3D Transforms.....	80
9. Transitions .....	81
10. Animations.....	81
11. Multi-column .....	82
12. Resizing .....	82
13. Box Sizing .....	83
14. Media Query .....	84
a. Xác định media.....	84
b. Các loại media types trong css3 .....	85
c. Ví dụ .....	85
<b>III. Reponsevice .....</b>	<b>85</b>
1. Giới thiệu .....	85
2. Khai báo Viewport.....	86
3. Các nguyên tắc khi thực hiện responsive .....	87
a. Responsive Images .....	87
b. Responsive Text Size .....	88
c. Media query .....	88
4. Các nền tảng hỗ trợ Responsive Web Design .....	89
<b>J. BOOTSTRAP 4 .....</b>	<b>90</b>
<b>I. Sơ lược về Bootstrap.....</b>	<b>90</b>

1. Giới thiệu .....	90
2. Các ưu điểm của Bootstrap .....	90
<b>II. Download và nhúng bootstrap vào trang .....</b>	<b>91</b>
1. Downloaf bootstrap .....	91
2. Nhúng bootstrap vào trang web .....	91
3. Thiết lập cho website hỗ trợ giao diện Mobile .....	91
4. Một số lưu ý khi sử dụng Bootstrap .....	92
<b>III. Sử dụng các thành phần trong bootstrap.....</b>	<b>92</b>
1. Chuẩn bị.....	92
2. Color .....	93
a. Text color.....	93
b. Background color .....	93
3. Form.....	94
a. Form inline .....	94
b. Form horizontal .....	94
4. Navigation.....	95
a. Navigation 1 cấp.....	95
b. Navigation collapse .....	96
c. Navigation có dropdown .....	96
d. Fixed navigation .....	97
5. Badges.....	97
6. Table .....	98
7. Button.....	100
8. Pagination .....	100
9. Card.....	101
10. Image .....	102
11. Alert .....	103
12. Carousel .....	103
13. Modal .....	105
14. Collapse .....	105
15. Jumbotron .....	106
16. Text/Typography .....	106
17. Flex .....	107



---

18. Tooltip.....	109
19. Popover .....	109
20. Scrollspy .....	110
21. Grid trong Bootstrap .....	112
<b>K. THIẾT KẾ LAYOUT CÙNG BOOTSTRAP.....</b>	<b>114</b>
<b>I. Một số quy định .....</b>	<b>114</b>
<b>II. Cấu trúc cơ bản của trang.....</b>	<b>114</b>
<b>III. Một số dạng layout .....</b>	<b>115</b>
1. Layout 3 cột rộng bằng nhau .....	115
2. Layout 3 cột rộng khác nhau .....	115
3. Layout 2 cột, lồng nhau .....	115
4. Layout hỗn hợp cho Mobile, Tablet và Desktop .....	116
<b>IV. Những cách để có một layout đẹp. ....</b>	<b>116</b>
1. Tự thiết kế .....	116
2. Tự thiết kế với tool .....	117
3. Tìm/mua template trên mạng .....	117

---

## A. NGÔN NGỮ HTML

### I. Một số khái niệm

#### 1. Trang web (Webpage)

Trang web (tĩnh) là một file dạng text chứa dữ liệu và các tag HTML. Khi hiển thị trong trình duyệt web, dữ liệu sẽ được hiển thị theo quy định của các tag mà nó nằm trong. Dữ liệu trong trang web có thể là văn bản, hình ảnh, âm thanh, video...

Một định nghĩa khác - nếu nhìn ở kết quả trong trình duyệt - chúng ta có thể nói một cách nôm na rằng: trang web là một trang thông tin giống như một trang báo, nhưng súc tích hơn vì có đủ các dạng thông tin: chữ, hình, âm thanh, video.

#### 2. Hyperlink

Hyperlink là một liên kết chỉ đến một trang web khác. 1 trang web có thể chứa nhiều link.

#### 3. Website

Là một tập hợp nhiều trang web thể hiện thông tin của một tổ chức, một chủ đề nào đó.

Mỗi website có một trang web gọi là trang chủ, trang chủ này sẽ chứa các hyperlink chỉ đến các trang web khác trong website.

Người xem sẽ vào website bắt đầu từ trang chủ, từ trang này, nhờ các link trong đó mà họ sẽ đến được các trang khác trong toàn website.

#### 4. Browser (Trình duyệt web)

Là chương trình dùng để xem các trang web. Ví dụ : Internet Explorer, Firefox, Chrome...

#### 5. WebServer

Là các máy “phục vụ web”, đây là các máy tính trên Internet/Intranet có cài chương trình webserver, 2 chương trình webserver nổi tiếng nhất là: IIS và Apache.

Webserver sẽ trả về cho người sử dụng trang web mà họ yêu cầu để họ xem. Webserver liên lạc với browser qua giao thức http.

Một Webserver có thể chứa nhiều website.

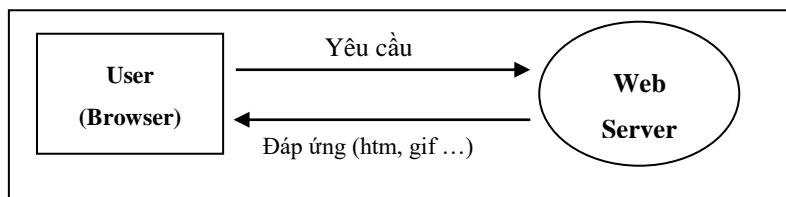
## 6. Http

Khi bạn gõ một địa chỉ trong thanh Address của browser hoặc nhấp một liên kết, giữa browser của bạn và webserver sẽ diễn ra hàng loạt các liên lạc/ trao đổi với nhau để hiển thị trang web cho bạn xem. Những hoạt động liên lạc này là giao thức http

## 7. Web tĩnh

Là trang web chỉ có tag html và dữ liệu, tất cả đều gõ trực tiếp trong trang chứ không đặt ở nơi khác. File có tên mở rộng là .html hoặc .htm

Mô hình web tĩnh:

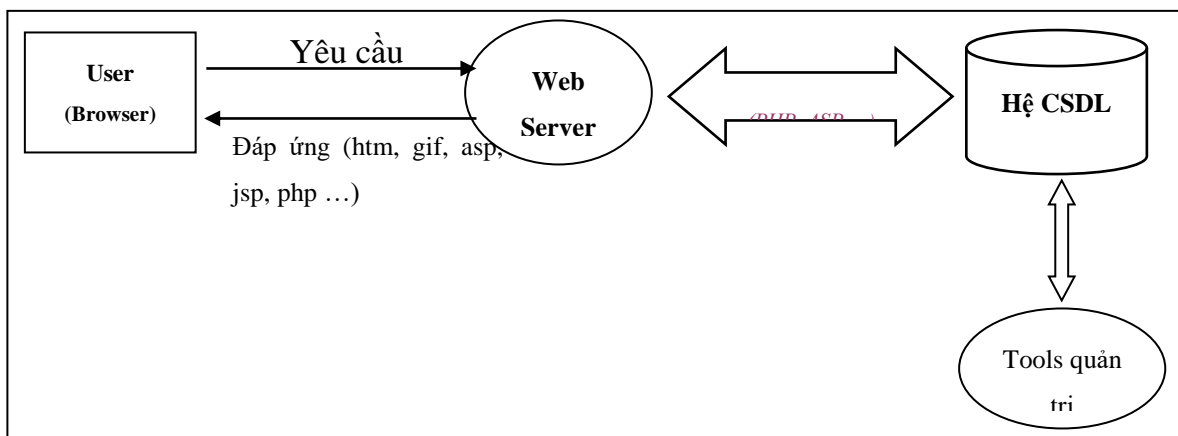


Trong mô hình web tĩnh, user yêu cầu 1 trang web html, trang web này đã được thiết kế sẵn và đặt trên webserver, trang web không hề có tương tác đến CSDL hay hệ thống. Webserver chỉ việc lấy file htm trả về cho user. Vậy là xong.

## 8. Web động

Là trang web có truy xuất đến cơ sở dữ liệu (Database) hoặc có tương tác với webserver để thực hiện một chức năng cao cấp nào đó. Một trang web động có thể trả về những kết quả khác nhau tùy theo yêu cầu của người sử dụng. Thiết kế web động đòi hỏi người thiết kế có nhiều kiến thức: HTML, Javascript, Database, WebServer, ... tốn nhiều công sức và thời gian.

Mô hình Web động:



Trong mô hình web động, Webserver sẽ tương tác với các chương trình “hậu trường” phía sau nó (PHP, ASP...) để thực hiện 1 số việc nào đó, (thường là kết nối cơ sở dữ liệu), các chương trình này lấy dữ liệu trong hệ quản trị cơ sở dữ liệu và thực hiện định dạng (nếu cần) rồi đưa về cho webserver, webserver sẽ trả về cho user.

Các chương trình “hậu trường” như PHP, ASP... là chương trình trung gian, là cầu nối giữa Webserver và cơ sở dữ liệu. Sở dĩ có chúng là vì tương tác với cơ sở dữ liệu không phải là mục tiêu của webserver, nhiệm vụ chính của WebServer là tương tác với user để trả về trang web (qua giao thức http).

## II. Các loại trang web trong 1 website

### 1. Trang chủ (Homepage)

Là trang đầu tiên xuất hiện khi website được gọi tên. Trang chủ thường chứa các thông tin căn bản nhất của website và các liên kết đến các trang chuyên đề. Trang chủ có thể có các tên như index.html, default.htm

### 2. Trang chuyên đề

Là trang lớn mang các liên kết đến các trang đơn vị có cùng tính chất. Ví dụ: Trang chuyên đề giới thiệu lãnh đạo công ty liên kết đến các trang về giám đốc, phó giám đốc,

trường phòng.... Trang chuyên đề giới thiệu các dịch vụ liên kết đến các trang đơn vị có nội dung giới thiệu từng dịch vụ ...

Trong website tin tức, trang chuyên đề là các trang hiển thị tin trong một loại.

### 3. Trang đơn vị

Thường được bố trí vào các thư mục có cùng chủ đề, chứa nội dung cụ thể của website. Một trang đơn vị có thể xuất hiện trên một hay nhiều trang chuyên đề tùy theo tính chất của trang đó.

Các trang đơn vị thường là một trang web tĩnh với đuôi htm hay html, cũng có thể là những trang web động với kết quả hiển thị là những thông tin trích xuất từ database.

Trong website tin tức, trang đơn vị là các trang hiển thị chi tiết một tin nào đó.

### 4. Trang phản hồi (Feedback)

Chứa các form thu nhận thông tin từ người xem. Đây là đặc trưng của Internet: thông tin hai chiều. Người xem web không chỉ nhận thông tin từ web mà còn có thể giao tiếp với chủ website thông qua trang web. Thông tin do người đọc nhập vào có thể hiển thị ngay tại trang web hoặc có thể chuyển đến một địa chỉ email nhất định nào đó hoặc đưa vào cơ sở dữ liệu.

Có nhiều loại trang phản hồi tùy mục đích sử dụng form thu nhận thông tin. Ví dụ trang góp ý, trang đặt câu hỏi thắc mắc...

### 5. Trang web động

Là trang web có truy xuất đến cơ sở dữ liệu (Database) hoặc có tương tác với webserver để thực hiện 1 chức năng cao cấp nào đó. Một trang web động có thể trả về những kết quả khác nhau tùy yêu cầu của người sử dụng. Thiết kế web động đòi hỏi người thiết kế có nhiều kiến thức: HTML, Javascript, Database, WebServer, ....

---

### III. Các thành phần thường có trong 1 trang web

#### 1. Chữ

Là những thông tin chính trình bày trong trang web.

#### 2. Hình ảnh

Có 3 loại hình ảnh dùng để chèn vào trang web, đó là gif, png, jpg. Gif nhẹ ký nhưng màu sắc đơn giản, jpg và png nặng ký hơn và màu sắc mịn đẹp. Gif giúp tạo các hình animation chuyển động nhúc nhích làm vui mắt. Hình ảnh nhiều làm cho việc load web chậm. Gif và png có thể tạo hiệu ứng transparent (trong suốt) cho hình.

#### 3. Banner

Là một file ảnh dùng để quảng cáo hay tiêu đề website.

#### 4. Logo

Là biểu tượng của website, hoặc của cơ quan chủ quản website.

#### 5. Hit Counter

Là thông tin đếm số người truy cập website.

#### 6. Search form

Là hộp thoại giúp người xem tìm kiếm thông tin cần tìm. Search form có thể dùng để tìm thông tin trong một trang, một site hay tất cả các site trên toàn cầu.

#### 7. Navigator (Menu)

Là tập hợp những liên kết dẫn đến các trang chuyên đề.

#### 8. Multimedia

Là các file ảnh, video hay âm thanh lồng trong trang.

## IV. Giới thiệu HTML

HTML (*H*yper *T*ext *M*arkup *L*anguage) là một ngôn ngữ để quy định cách hiển thị thông tin trong trang web. HTML gồm nhiều lệnh, mỗi lệnh gọi là 1 tag. Mỗi tag quy định một cách thức hiển thị dữ liệu trong trang web. Ví dụ như: chữ đậm, chữ nghiêng, màu chữ ... Người xem trang web không thấy các tag mà chỉ thấy các dữ liệu được định dạng bởi các tag. Nói đơn giản : **HTML Là 1 ngôn ngữ dùng để tạo ra các trang web.**

Các tag cùng với dữ liệu trong đó được lưu trong 1 file text, gọi là trang web. File này thường có tên mở rộng là .html hoặc .htm. Ví dụ: Nếu bạn gõ như sau khi tạo trang web:

```
Lớp: <b>Web doanh nghiệp</b><br>  
Họ tên: <u><i>Nguyễn Văn Tèo</i></u>
```

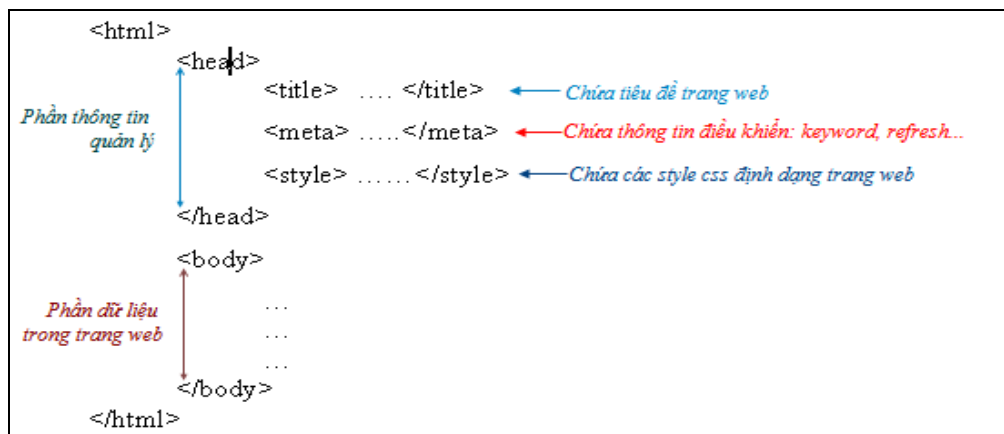
thì kết quả hiện trong Browser sẽ thế này:

```
Lớp: Web doanh nghiệp  
Ho tên: Nguyễn Văn Tèo
```

Tên tag bạn nên viết bằng chữ thường, đặt trong 2 dấu < >. Tag thường có mở và đóng, một số tag chỉ có mở.

## V. Cấu trúc của một trang web

- Một trang web thường có mở đầu và kết thúc bởi tag **<html>**
- Tag **head** chứa những thông tin để quản lý và hoạt động nội tại bên trong trang web, không hiện ra cho user xem.
- Tag **title** là tiêu đề của trang web, bao giờ cũng nằm trong tag head
- Tag **body** chứa dữ liệu hiện ra trong trang web cho user xem.

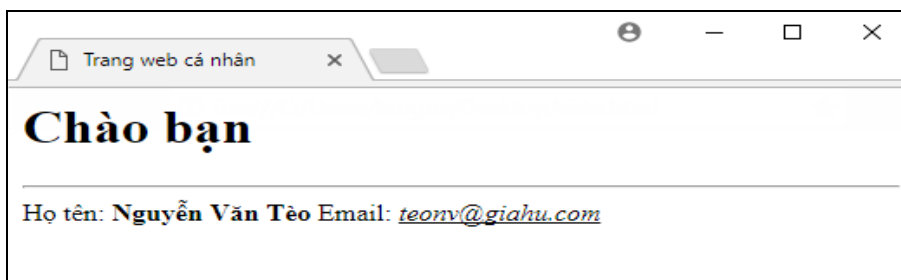


## VI. Soạn thảo trang web

### 1. Mở Notepad++ và gõ nội dung sau

```
<html>
<head>
    <title>Trang web cá nhân</title>
</head>
<body>
<h1> Chào bạn </h1>
<hr>
Họ tên: <b> Nguyễn Văn Tèo</b>
Email: <i><u> teonv@giahu.com </u></i>
</body>
</html>
```

- Nhấp menu File → Save để lưu file. Lưu vào **Desktop**, tên file là **vidu.html** (Encoding là **UTF-8**)
- Nhấp đúp vào file **vidu.html** trên Desktop để xem kết quả





## 2. Xem Source HTML của trang

Trong trình duyệt, nhấp phải chuột → View page Source, bạn sẽ thấy code HTML của trang. Hoặc hay hơn là nhấp nút phải tại tên file .html → **Edit With Notepad++**

## 3. Thêm lệnh xuống hàng giữa Họ tên và Email

- Thêm sau tag </b> lệnh xuống hàng: **<br/>**
- Đóng Notepad và lưu lại.
- Nạp lại trang vidu.html (phím F5) để xem những thay đổi

## 4. Đưa hình (h.jpg) vào trang web:

- Trong source HTML của trang, nhấp sau tag <hr> và gõ như sau:

```

```

- Mở thêm 1 cửa sổ Browser. Vào Google, tìm 1 hình nào đó (cỡ lớn) và lưu vào Desktop với tên h.jpg (Chú ý: Trong trang google có chỗ để bạn tìm hình theo kích thước, theo màu, theo loại hình, theo thời gian.)
- Nạp lại trang vidu.html để xem kết quả

## 5. Chỉnh kích thước và title, cách canh và đường viền cho hình:

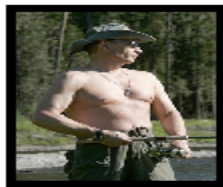
- Xem Source HTML của trang.
- Nhấp sau chữ <img> và bổ sung để được như sau:

```

```

- Xong thì lưu lại
- Nạp lại trang vidu.html để xem kết quả

Chào bạn



Họ tên: **Nguyễn Văn Tèo**  
Email: [teonv@giayhu.com](mailto:teonv@giayhu.com)

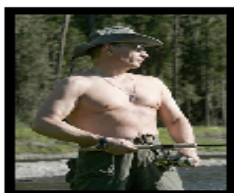
- Sửa chữ **left** thành **right** và xem kết quả

## 6. Thêm liên kết vào trang (liên kết nằm trong tag a)

- Trong source HTML của tran, nhấp sau chữ `</i>` , Enter xuống hàng vào nhập vào code sau rồi lưu lại:

```
<p>Website thường xem: </p>
<a href=http://nhatnghe.com> Nhất nghệ</a><br>
<a href=http://google.com target=_blank> Google </a>
```

- Nạp lại trang vidu.html để xem kết quả



Họ tên: **Nguyễn Văn Tèo**  
Email: [teonv@giayhu.com](mailto:teonv@giayhu.com)

Website thường xem:

[Nhất nghệ](http://nhatnghe.com)  
[Google](http://google.com)

## VII. Các tag HTML căn bản

Tên Tag	Cú pháp	Định nghĩa
<code>&lt;!--...--&gt;</code>	<code>&lt;!-- Nội dung chú thích --&gt;</code>	Chú thích trong HTML
<code>&lt;!DOCTYPE&gt;</code>	<code>&lt;!doctype html public "-//ietf//dtd html Level 1//en"&gt;</code>	Ghi thông tin về version HTML áp dụng trong tài liệu Web
<code>&lt;a&gt;</code>	<code>&lt;a href=abc.html&gt; Tên hiển thị&lt;/a&gt;</code>	Tạo liên kết đến trang abc.html
<code>&lt;b&gt;</code>	<code>&lt;b&gt; Nội dung đoạn văn&lt;/b&gt;</code>	In đậm
<code>&lt;blockquote&gt;</code>		Định nghĩa một khối trích dẫn

<b>&lt;body&gt;</b>	<b>&lt;BODY&gt;</b> Nội dung trang HTML <b>&lt;/BODY&gt;</b>	Tag chứa toàn bộ nội dung trang
<b>&lt;br&gt;</b>	Nội dung đoạn văn <b>&lt;br&gt;</b> bắt đầu một dòng mới	Xuống dòng, không qua đoạn mới
<b>&lt;caption&gt;</b>	<pre> &lt;table&gt; &lt;caption&gt;Bảng điểm &lt;/caption&gt; &lt;tr&gt;&lt;td&gt;Tên&lt;/td&gt;&lt;td&gt;Điểm&lt;/td&gt;&lt;/tr&gt; &lt;tr&gt;&lt;td&gt;Tèo&lt;/td&gt;&lt;td&gt;9&lt;/td&gt;&lt;/tr&gt; &lt;/table&gt; </pre>	Nhãn của 1 table
<b>&lt;dl&gt;</b> <b>&lt;dt&gt;</b> <b>&lt;dd&gt;</b>	<pre> ,&lt;dl&gt; ,&lt;dt&gt;Definition of the term&lt;/dt&gt; ,&lt;dd&gt;Definition of the term&lt;/dd&gt; ,&lt;dt&gt;Definition of the term&lt;/dt&gt; ,&lt;dd&gt;Definition of the term&lt;/dd&gt; &lt;/dl&gt; </pre>	<b>dl:</b> Tạo danh sách dạng list <b>dt:</b> phần tiêu đề trong 1 mục <b>dd:</b> phần mô tả trong 1 mục
<b>&lt;div&gt;</b>	<b>&lt;div&gt;.....&lt;/div&gt;</b>	div chứa 1 vùng dữ liệu trong trang
<b>&lt;em&gt;</b>	<b>&lt;em&gt;</b> Nội dung <b>&lt;/em&gt;</b>	Định dạng kiểu chữ nghiêng
<b>&lt;embed&gt;</b>	<pre> &lt;embed src="yourfile.mid" autostart="true" hidden="false" loop="false"&gt; </pre>	Nhúng đối tượng (nhạc, phim..)
<b>&lt;fieldset&gt;</b>		Định nghĩa một fieldset, dùng để bao quanh 1 vùng dữ liệu (cho đẹp)
<b>&lt;form&gt;</b>	<b>&lt;form method=post action="xuly.php"&gt;</b> <b>&lt;/form&gt;</b>	Tạo 1 form
<b>&lt;input&gt;</b>	<pre> &lt;form method=post action="xuly.php"&gt; &lt;input type="text" size="10" maxlength="30"&gt; &lt;input type="Submit" value="Submit"&gt; &lt;/form&gt; </pre>	<b>form:</b> Tạo 1 form <b>input:</b> dùng để chèn các text field, button, checkbox, radio button, image field trong form
<b>&lt;h1&gt; to &lt;h6&gt;</b>	<pre> &lt;h1&gt;Tiêu đề 1&lt;/h1&gt; &lt;h2&gt;Tiêu đề 2&lt;/h2&gt; &lt;h3&gt;Tiêu đề 3&lt;/h3&gt; </pre>	Tạo tiêu đề (cấp 1 đến cấp 6)

	<p>&lt;h4&gt;Tiêu đề 4 &lt;/h4&gt; &lt;h5&gt;Tiêu đề 5 &lt;/h5&gt; &lt;h6&gt;Tiêu đề 6 &lt;/h6&gt;</p>	
<p>&lt;head&gt; &lt;meta&gt;</p>	<p>, &lt;head&gt; &lt;meta name="Description" content="Mô tả website"&gt; &lt;meta name="keywords" content="Từ khóa 1, từ khóa 2,..."&gt; &lt;/head&gt;</p>	<p><b>head:</b> Phần thông tin quản lý <b>meta:</b> Tag Meta chứa các thông tin quản lý trang web. Ví dụ: Mô tả trang, khai báo keyword, từ khóa, động chuyển hướng, tạo hiệu ứng chuyển trang ...</p>
<hr>	<hr color="#FF0000">	Tạo một đường gạch ngang
<i>	<i> Nội dung </i>	Chữ in <i>ngghiêng</i>
<iframe>	<p>&lt;iframe name="content_frame" width="488" height="244" src="welcome.htm"&gt; &lt;/iframe&gt;</p>	Tạo 1 iframe (iframe là 1 vùng trong trang chứa 1 trang khác)
<img>	<p>&lt;img src="hinh.gif" width="41" height="41" border="0" alt="Mô tả về hình ảnh"&gt;</p>	Chèn hình vào văn bản.
<legend>		Title của một fieldset
<p>&lt;ol&gt;, &lt;li&gt;</p>	<p>&lt;ol&gt; &lt;li&gt;Danh mục 1&lt;/li&gt; &lt;li&gt;Danh mục 2&lt;/li&gt; &lt;li&gt;Danh mục 3&lt;/li&gt; &lt;/ol&gt;</p>	<p><b>ol:</b> Tạo danh sách dạng unordered list <b>li:</b> Tạo 1 mục trong danh sách</p>
<link>	<link href="c1.css" rel="stylesheet" type="text/css" />	Liên kết đến 1 file css
<ol>	<p>&lt;ol&gt; &lt;li&gt;Danh mục 1&lt;/li&gt; &lt;li&gt;Danh mục 2&lt;/li&gt; &lt;li&gt;Danh mục 3&lt;/li&gt; &lt;/ol&gt;</p>	Tạo 1 danh sách dạng ordered list (mỗi mục có 1 số thứ tự ở đầu)
<p>&lt;option&gt;, &lt;select&gt;</p>	<p>&lt;select&gt; &lt;option value="giá trị 1"&gt;Option1&lt;/option&gt; &lt;option value="giá trị 2"&gt;Option2&lt;/option&gt;</p>	<p><b>select:</b> dùng để tạo 1 select box <b>option:</b> Tạo 1 item trong 1 select box</p>

	<code>&lt;/select&gt;</code>	
<code>&lt;p&gt;</code>	<code>&lt;p&gt;Nội dung đoạn văn bản.&lt;/p&gt;</code>	Paragraph
<code>&lt;script&gt;</code>	<code>&lt;script type="text/javascript" src="a.js"&gt;</code> <code>&lt;/script&gt;</code>	Liên kết đến 1 file javascript (a.js)
<code>&lt;span&gt;</code>	<code>&lt;span&gt; Nội dung văn bản&lt;/span&gt;</code>	Bao quanh 1 vùng text để định dạng
<code>&lt;strong&gt;</code>	<code>&lt;strong&gt;Nội dung đoạn văn bản&lt;/strong&gt;</code>	Chữ đậm
<code>&lt;style&gt;</code>	<code>&lt;style&gt;body { color : white}</code> <code>P { color : blue; font-size : 12pt; }</code> <code>H1 { color : red; font-size : 18pt}</code> <code>&lt;/style&gt;</code>	Tag chứa các style CSS trong vùng head
<code>&lt;sub&gt;</code>	<code>&lt;sub&gt;Nội dung đoạn văn bản&lt;/sub&gt;</code>	Chữ subscript (chữ xuống dưới+nhỏ)
<code>&lt;sup&gt;</code>	<code>&lt;sup&gt;Nội dung đoạn văn bản&lt;/sup&gt;</code>	Chữ superscript (chữ lên cao+nhỏ)
<code>&lt;table&gt;,&lt;br&gt;&lt;td&gt;,&lt;br&gt;&lt;tr&gt;</code>	<code>&lt;table&gt;</code> <code>&lt;tr&gt; &lt;th&gt;Họ tên&lt;/th&gt; &lt;th&gt;Điểm&lt;/th&gt;</code> <code>&lt;/tr&gt;</code> <code>&lt;tr&gt; &lt;td&gt;Tèo&lt;/td&gt; &lt;td&gt;9&lt;/td&gt;&lt;/tr&gt;</code> <code>&lt;tr&gt; &lt;td&gt;Tý&lt;/td&gt; &lt;td&gt;10&lt;/td&gt;&lt;/tr&gt;</code> <code>&lt;table&gt;</code>	<b>table:</b> Tạo 1 bảng <b>tr:</b> Tạo 1 hàng trong bảng <b>td:</b> Tạo 1 ô dữ liệu trong bảng <b>th:</b> Tạo 1 ô tiêu đề trong bảng
<code>&lt;textarea&gt;</code>	<code>&lt;textarea&gt;...&lt;/textarea&gt;</code>	Tạo 1 đối tượng trong form để user gõ nhiều hàng chữ
<code>&lt;u&gt;</code>	<code>&lt;u&gt;Nội dung đoạn văn bản&lt;/u&gt;</code>	Gạch dưới
<code>&lt;ul&gt;</code>	<code>&lt;ul&gt;</code> <code>&lt;li&gt;Danh mục 1&lt;/li&gt;</code> <code>&lt;li&gt;Danh mục 2&lt;/li&gt;</code> <code>&lt;/ul&gt;</code>	Tạo 1 danh sách dạng unordered list (mỗi mục có 1 số ở đầu)

---

## B. CĂN BẢN VỀ DREAMWEAVER

### I. Giới thiệu về Dreamweaver

Dreamweaver là chương trình dùng để tạo trang web tĩnh, web động rất hay. Dreamweaver là một thành phần trong bộ sản phẩm của hãng Adobe gồm nhiều sản phẩm như: Photoshop, Dreamweaver, Fireworks, Acrobat Pro,...

Mở Dreamweaver: Start → (All) Programs → Adobe Dreamweaver ...

### II. Thao tác với trang

#### 1. Tạo trang web

##### a. Tạo trang HTML trống

Menu File → New → Blank page → HTML → None → Create

##### b. Tạo trang PHP

Menu File → New → Blank Page → PHP → None → Create

#### 2. Xem thử trang web

Gõ phím F12 để xem thử

#### 3. Chuyển chế độ làm việc

##### a. Chế độ Code

Là chế độ để xem và làm việc với code HTML của trang.

##### b. Chế độ Design

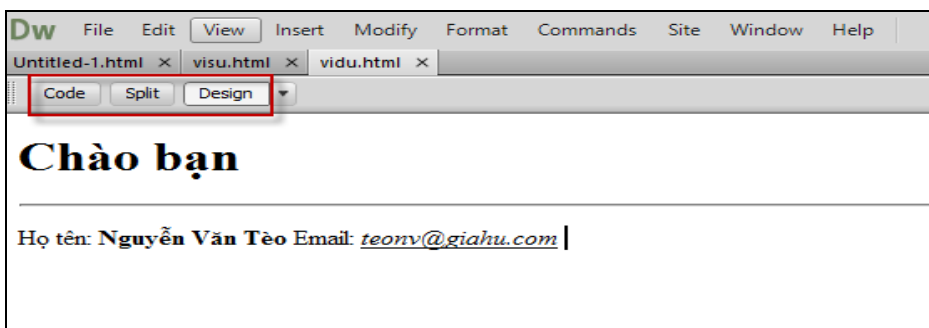
Là chế độ chủ yếu để xem và làm việc với trang. Trang web sẽ hiện gần giống hệt theo ý của bạn ở chế độ này.

### c. Chế độ Split

Chia đôi màn hình. Phía trên hiện code HTML, phía dưới là chế độ Design.

### d. Chuyển chế độ của trang

Nhấp nút Code, Design, Split ở trên vùng soạn thảo (thanh Document)



## 4. Lưu trang web

Nhấp menu File → Save /Save As / Save All

## 5. Mở trang

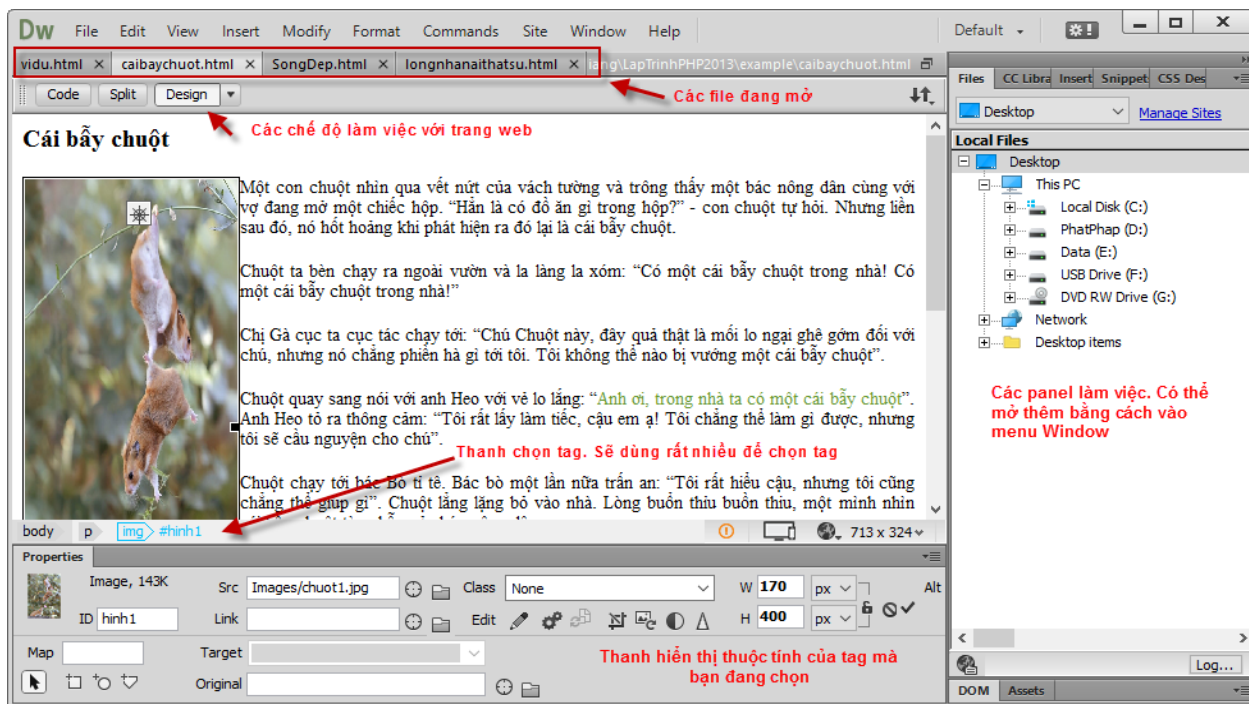
Nhấp đúp vào tên file trong panel Files.

## 6. Đổi tên file

Nhấp tên file trong panel Files → gõ phím F2 → gõ tên file mới



### III. Màn hình Dreamweaver



#### 1. Toolbar Document

Là thanh toolbar chứa các thao tác với tài liệu hiện hành. Toolbar này chứa các nút **Design**, **Code**, **Split** mà bạn đã dùng.

#### 2. Thanh Properties

Là thanh nằm bên dưới vùng soạn thảo. Thanh Properties là nơi hiện các thuộc tính của đối tượng mà bạn đang chọn. Bạn dùng thanh này để hiệu chỉnh thuộc tính của đối tượng.

Ẩn hiện thanh Properties: Nhấp menu Window → Properties

#### 3. Thanh Tag

Là thanh nằm ngang, ngay phía trên thanh Properties. Đây là nơi hiện ra các tag chứa đối tượng mà bạn đang chọn. Nghĩa là nếu bạn muốn biết một đối tượng nào đó nằm trong các tag nào, hãy chọn nó rồi nhìn vào thanh này.



---

## 4. Panel group

Là vùng nằm dọc bên phải của vùng soạn thảo. Đây là nơi chứa rất nhiều thông tin mà bạn có thể dùng để tạo và định dạng trang, các panel thường dùng trong đây là Insert, CSS, Files

### *a. Insert bar*

Là toolbar dùng để chèn các đối tượng vào trang web (table, hình, form, div...). Ẩn hiện thanh này bằng cách vào menu **Window → Insert**

### *b. Css bar*

Là toolbar dùng để định dạng các tag bằng kỹ thuật CSS. Ẩn hiện thanh này bằng cách vào menu **Window → CSS Designer**

### *c. Files bar*

Là toolbar dùng để quản lý các file trong website. Ẩn hiện thanh này bằng cách vào menu **Window → Files**

### *d. Dom bar*

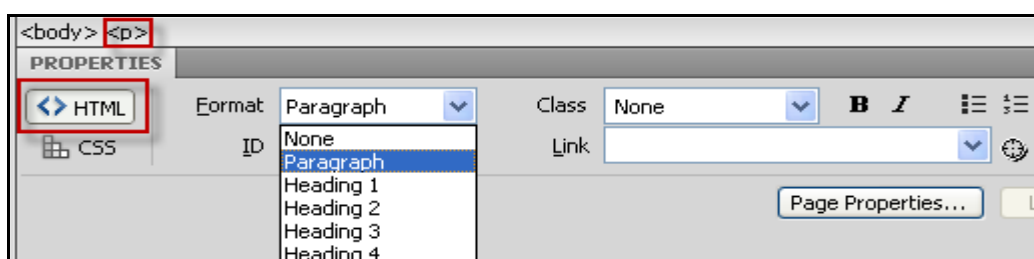
Là toolbar dùng để quản lý cấu trúc các tag trong trang theo dạng cấu trúc cây thư mục. Dùng thanh này để chọn tag, duplicate tag, di chuyển, xóa tag... rất hay, nhất là với các trang web có cấu trúc phức tạp (dựa trên bootstrap) mà bạn sẽ học sau. Ẩn hiện thanh này bằng cách vào menu **Window → DOM**

## C. CHÈN THÔNG TIN VÀO TRANG

### I. Paragraph

Paragraph là 1 đoạn text gồm một hoặc nhiều hàng chữ nằm trong tag p, tag h1-h6 . Khi nhập chữ rồi Enter xuống hàng, Dw sẽ tự động bao quanh vùng text vừa gõ 1 tag p. Có thể thay đổi tag bao quanh đoạn như sau:

- Chọn đoạn text
- Trên thanh Properties/mục **Format**: chọn tag (Heading 1 → Heading 6). Tag h1-h6 thường dùng khi định dạng các tiêu đề chương, bài, đề mục



### II. Hình ảnh

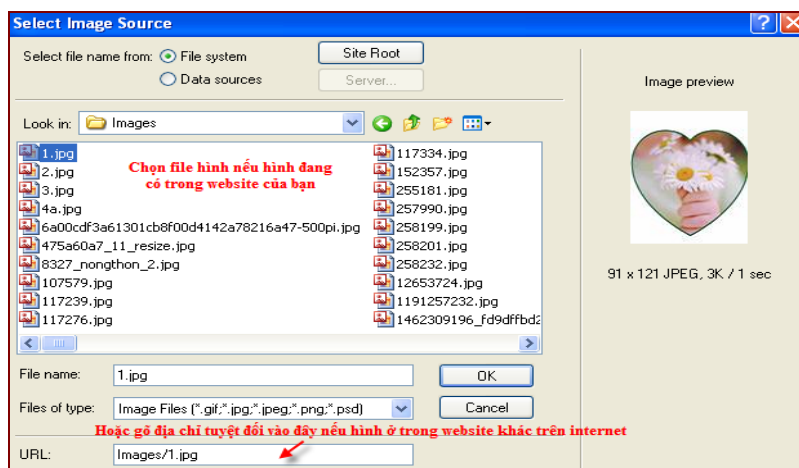
#### 1. Các loại hình ảnh chèn vào vào trang

Hình ảnh làm cho trang web thêm sống động. Có 3 loại hình có thể dùng để chèn vào trang:

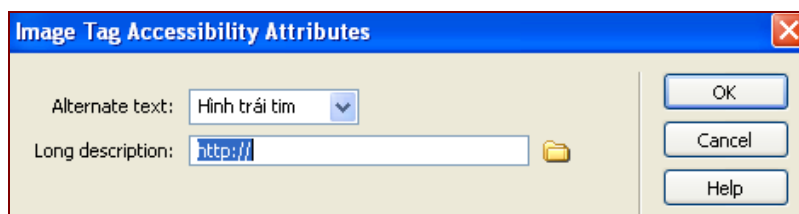
- Dạng gif: chất lượng thấp, ít màu, có thể là hình động, có thể tạo được hiệu ứng Transparent
- Dạng jpg: chất lượng tốt, nhiều màu, không thể là hình động, không tạo được hiệu ứng Transparent
- Dạng png: chất lượng tốt, nhiều màu, có thể là hình động, có thể tạo được hiệu ứng Transparent, tỷ lệ nén tốt nên file size thường nhỏ hơn jpg. Nên dùng hình dạng này để chèn vào trang.

## 2. Chèn hình vào trang

- Nhấp vị trí chèn hình rồi vào menu Insert → Image
- Chọn file hoặc gõ địa chỉ của hình trong mục URL

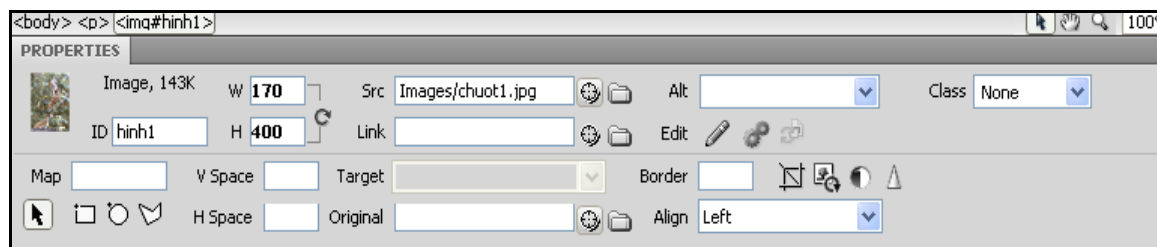


- Alternate Text: Text thay thế cho hình → OK



## 3. Chỉnh thuộc tính của hình

Trên thanh Properties, có thể chỉnh :



- Vspace/Hspace: Khoảng cách từ hình tới chữ theo chiều dọc/ chiều ngang

- W: bề rộng của hình.
- H: bề cao của hình.
- Src: địa chỉ, tên file hình.
- Link: địa chỉ trang web sẽ hiện khi user nhấp vào hình.
- Alt: thay thế cho hình.
- Border: độ dày của đường viền quanh hình. Nếu gõ 0 thì hình sẽ không có viền.
- Align: cách canh của hình so với văn bản xung quanh (Left, Right,...)

### III. Liên kết

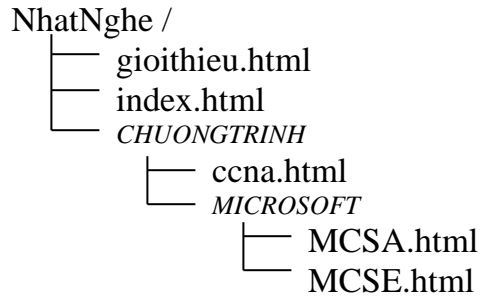
#### 1. Các loại địa chỉ

##### a. Địa chỉ tuyệt đối

- Là địa chỉ bao gồm đầy đủ 3 phần **protocol**, **domain**, **file**. Địa chỉ tuyệt đối thường dùng khi liên kết sang website khác
- Ví dụ: <http://www.nhatnghe.com/ChuongTrinh/CCNA.html>
  - Protocol là http
  - Domain là [www.nhatnghe.com](http://www.nhatnghe.com)
  - File là ChuongTrinh/CCNA.html

##### b. Địa chỉ tương đối

- Là địa chỉ không đầy đủ. Không có protocol và domain
- Thường dùng để liên kết giữa các trang trong nội bộ website
- Khi dùng địa chỉ tương đối, phải biết mình đang ở đâu, link đến đâu.
- Ví dụ: Site bạn đang thực hiện là NhatNghe



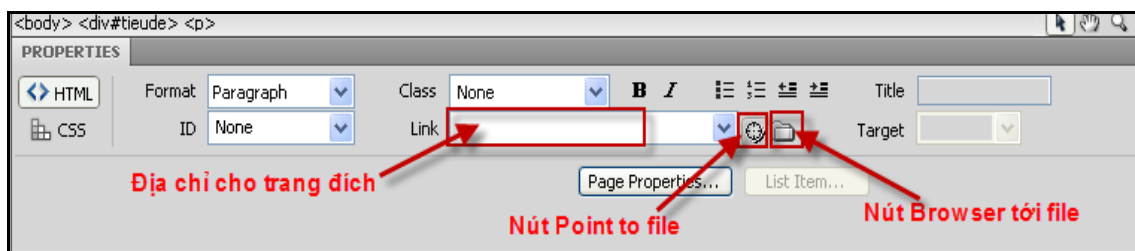
- a. Nếu bạn đang soạn file gioithieu.html, muốn tạo liên kết đến file index.html thì ghi địa chỉ là index.html
- b. Nếu bạn đang soạn file MCSE.html, muốn tạo liên kết đến file MCSA.html thì ghi địa chỉ là MCSE.html
- c. Nếu bạn đang soạn file gioithieu.html, muốn tạo liên kết đến file CCNA.html thì ghi địa chỉ là ChuongTrinh/ccna.html
- d. Nếu bạn đang soạn file gioithieu.html, muốn tạo liên kết đến file MCSA.html thì ghi địa chỉ là ChuongTrinh/Microsoft/MCSA.html
- e. Nếu bạn đang soạn file CCNA.html, muốn tạo liên kết đến file index.html thì ghi địa chỉ là ../index.html
- f. Nếu bạn đang soạn file MCSA.html, muốn tạo liên kết đến file index.html thì ghi địa chỉ là ../../index.html

### c. Địa chỉ gốc (Ký hiệu là /)

- Là địa chỉ bắt đầu bằng dấu / (tính từ gốc của website)
- Dùng để link trong nội bộ website
- Không quan tâm đến vị trí hiện tại, do đó:
  - a. Từ bất cứ trang nào trong site, muốn tạo liên kết đến file index.html thì ghi địa chỉ là /index.html

- b. Từ bất cứ trang nào trong site, muốn tạo liên kết đến file CCNA.html thì ghi địa chỉ là /ChuongTrinh/CCNA.html
- c. Từ bất cứ trang nào trong site, muốn tạo liên kết đến file MCSA.html thì ghi địa chỉ là /ChuongTrinh/Microsoft/MCSA.html

## 2. Tạo liên kết



Cách 1: Chọn text hoặc hình rồi gõ địa chỉ trong mục Link trên thanh Properties

Cách 2: Chọn text hoặc hình rồi nhấp nút Browse (trên thanh Properties) để chọn file

Cách 3: Chọn text hoặc hình rồi vào menu Insert → Hyperlink

- Text: Dòng text hyperlink
- Link: trang đích
- Target: cửa sổ mà trang đích hiện
- Title: tiêu đề của link

## IV. Chèn phim từ Youtube

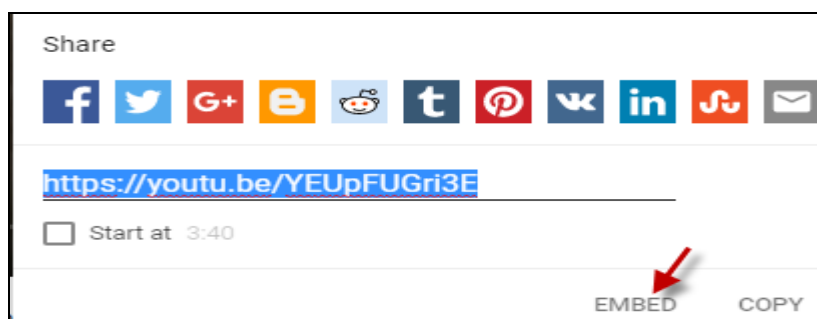
1. Xem 1 Video nào đó trên Youtube

<https://www.youtube.com/watch?v=YEUpFUGri3E>

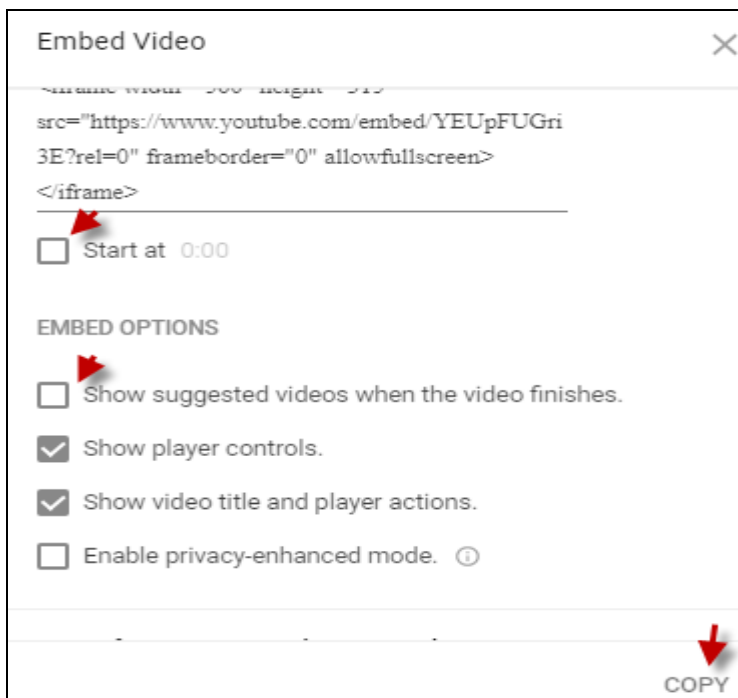
2. Nhấp nút Share (xem hình dưới)



3. Nhấp nút **Embed** (hình dưới)



4. Chọn các thông số rồi nhấp nút Copy (hình dưới)



5. Trong DREAMWEAVER, Paste tại vị trí muốn chèn video

```
<iframe width="560" height="315"
src="https://www.youtube.com/embed/YEUpFUGri3E?rel=0&start=30"
frameborder="0"
allowfullscreen></iframe>
```

Bạn chỉnh lại độ rộng, độ cao cho thích hợp

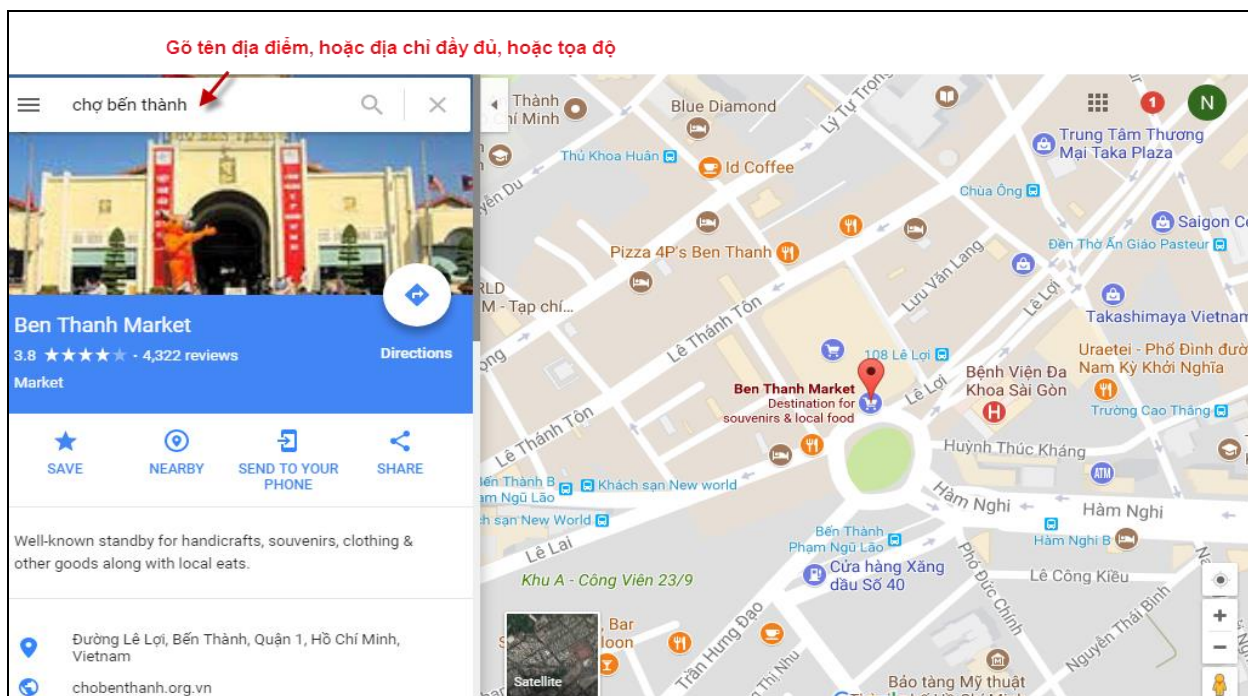
## V. Chèn bản đồ vào trang

### 1. Chèn bản đồ

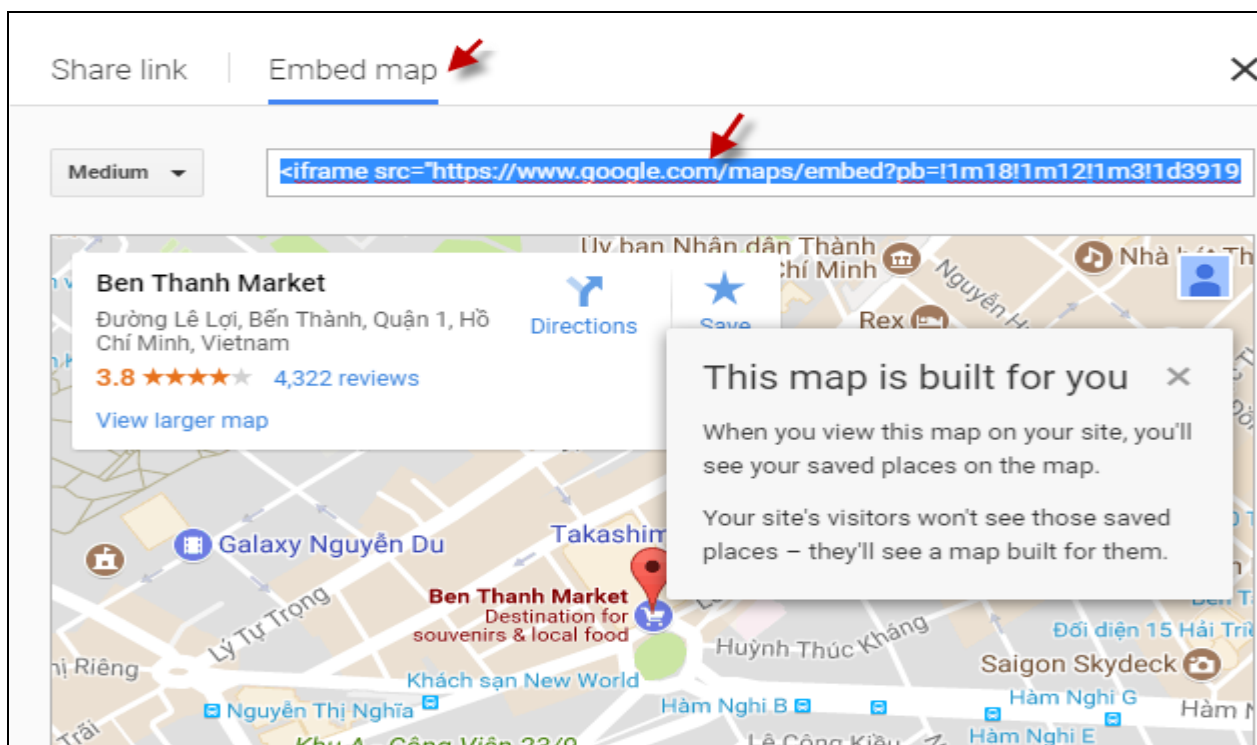
#### 1. Vào google.com/maps

Nếu địa điểm là nổi tiếng, chỉ cần bạn gõ tên, còn không thì gõ đầy đủ địa chỉ (số nhà, đường, phường, quận, tp). Hoặc bạn gõ tọa độ chính xác của chỗ cần hiện theo kiểu (vĩ độ , kinh độ)





2. Nhấp Nút **Share** (xem hình trên) rồi nhấp **Embed map**



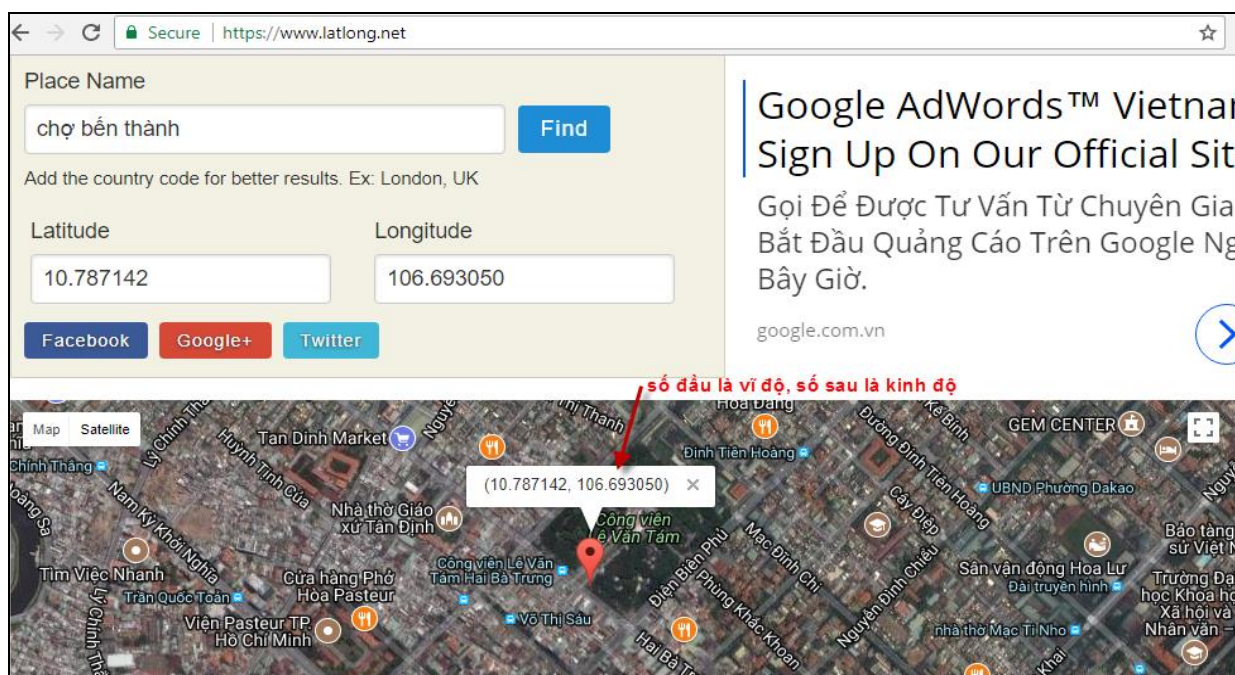
3. Copy địa chỉ hiện trong hình ( <iframe src=... ) rồi paste vào trong trang web.

```
<iframe
src="https://www.google.com/maps/embed?pb=!1m18!1m12!1m3!1d3919.5118593805914!2d106.696
13491439935!3d10.772053492324517!2m3!1f0!2f0!3f0!3m2!1i1024!2i768!4f13.1!3m3!1m2!1s0x317
52f3f3129e64d%3A0x8d6b2d79522c7f30!2sBen+Thanh+Market!5e0!3m2!1sen!2s!4v1507187219981
" width="600" height="450" frameborder="0" style="border:0" allowfullscreen></iframe>
```

Bạn có thể hiệu chỉnh độ rộng, độ cao của bản đồ

## 2. Tìm vĩ độ, kinh độ của 1 địa điểm bất kỳ

- Vào trang latlong.net
- Nhập 1 địa chỉ nổi tiếng nào đó. VD: Chợ bến thành
- Bạn phóng to và kéo bản đồ đến vị trí cần dùng rồi nhấp vào ➔ sẽ hiện vĩ độ và kinh độ

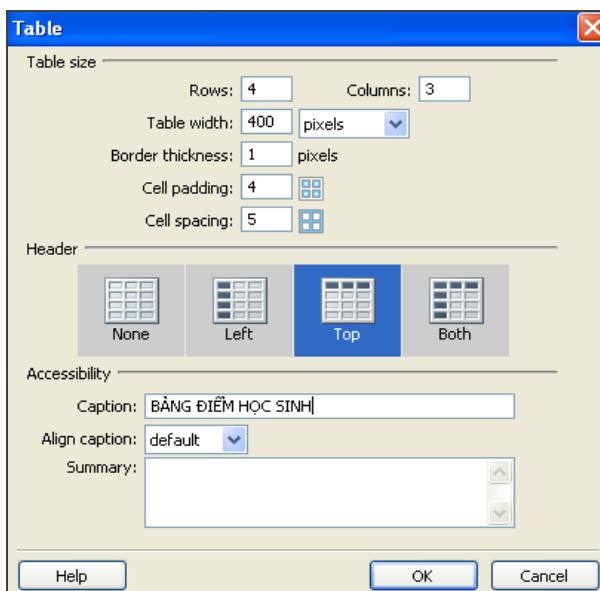


## VI. Table

Bạn dùng table khi muốn trình bày một vùng dữ liệu dạng bảng. hoặc bố cục dữ liệu thành dạng nhiều hàng cột.

### 1. Chèn table

- Nhấp vị trí muốn chèn rồi vào menu Insert → chọn **Table**
  - **Rows**: số hàng. **Columns**: số cột
  - **Table Width**: độ rộng table (tính bằng pixel hoặc %)
  - **Cell Padd**: khoảng cách từ đường viền của ô đến nội dung trong ô
  - **Cell Space**: khoảng cách giữa các ô
  - **Header**: cho biết table có vùng header hay không (top, left, both, none)
  - **Caption**: Nhãn của table
  - Nhấp OK
- Thêm xóa hàng, cột
  - Thêm hàng, cột: Nhấp 1 ô trong hàng/cột hiện hành rồi nhấp phải → Table → chọn **Insert Row** hoặc **Insert Column**
  - Xóa hàng cột: Nhấp 1 ô trong hàng/cột muốn xóa rồi nhấp phải → Table → Delete Row hoặc Delete Column
- Merge cell: Chọn các ô cần merge → Nhấp phải → Table → Merge Cell
- Split cell: Chọn ô cần chia → Nhấp phải → Table → Split Cell



- Nhập dữ liệu: Nhấp từng ô muốn nhập và gõ dữ liệu vào

Ho tên	Toán	Văn
Tèo	6	7
Tý	9	10
Lượng	8	5

- Cấu trúc HTML đơn giản, mỗi table có cấu trúc HTML như sau

```
<table>
<tr>    <!--Mỗi hàng trong tag tr-->
<th> Dữ liệu trong ô 1</th> <th> Dữ liệu trong ô 2 </th> ...
</tr>
<tr>    <!-- Mỗi ô trong tag th hoặc td -->
<td> Dữ liệu trong ô 1</td> <td> Dữ liệu trong ô 2 </td> ...
</tr>
</table>
```

## 2. Các tag trong một table

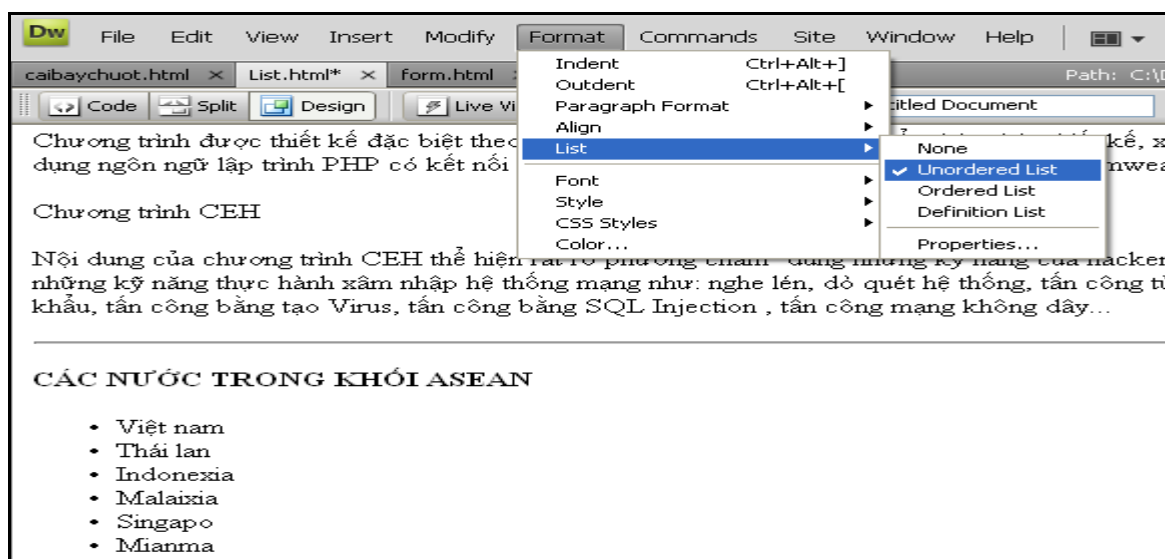
1. <table>: là tag bao trùm toàn bộ 1 table

2. <tr> : là tag bao quanh 1 hàng dữ liệu
3. <td>: là tag bao quanh 1 ô dữ liệu
4. <th>: là tag bao quanh 1 ô tiêu đề, chữ trong ô tiêu đề mặc định là đậm và canh giữa trong ô
5. <thead>: là tag bao quanh các tr dùng làm dòng tiêu đề trên.
6. <tbody>: là tag bao quanh các tr dùng làm dòng dữ liệu.
7. <tfoot> là tag bao quanh các tr dùng làm dòng tiêu đề dưới.

## VII. Danh sách

### 1. UnOrdered list

- Là list gồm nhiều mục dạng liệt kê. Mỗi mục có 1 ký hiệu (bullet) ở đầu.
- Thực hiện : Chọn text rồi vào menu Format → List → UnOrdered List



List dạng này nằm trong tag UL, mỗi mục nằm trong tag li . Bạn qua code để xem nhé.

### 2. Ordered list

- Là list gồm nhiều mục dạng liệt kê. Mỗi mục có 1 con số ở đầu

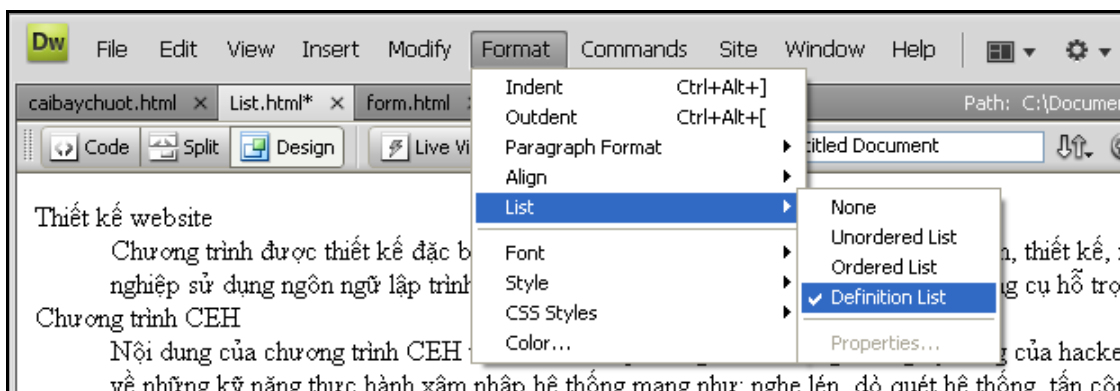


- Thực hiện: Chọn text rồi vào menu Format → List → Ordered List

**Chú ý:** List dạng này nằm trong tag OL, mỗi mục nằm trong tag li.

### 3. Definition list

- Là list dùng khi diễn tả nhiều item, mỗi item có mô tả của nó
- Thực hiện : Chọn text rồi vào menu Format → List → Definition List



**Chú ý :** List dạng này nằm trong tag dl, mỗi mục nằm trong tag dt ,dd.

## VIII. Các tag quan trọng: div, span, iframe

Phân trên là những tag quan trọng phải biết , đó là tag a, img, table, iframe. Dưới đây trình bày thêm một số tag nữa:

### 1. Tag DIV

Là tag có thể chứa được mọi đối tượng html bên trong nó (table, liên kết, hình ảnh...). Trong trang web, tag div đóng vai trò “container” (nghĩa là bao chứa 1 vùng dữ liệu nào đó). Ví dụ: vùng menu, vùng tin xem nhiều, vùng form đăng nhập, vùng tìm kiếm... Ví dụ:

```
<div id="loihayydep">
<p>Đừng để một ai chẳng nhận được gì khi rời bạn, dù bạn biết không bao
giờ gặp lại.</p>
<p>Đừng mua thứ hữu ích mà hãy mua thứ cần thiết </p>
<p>Nghệ thuật sống với nhau chính là nghệ thuật giữ khoảng cách </p>
</div>
```

## 2. Tag Span

Tag Span dùng để bao quanh một vùng text. Mặc định, tag span “chung sống hòa bình” với các text khác trong một hàng. Nghĩa là ở đầu và cuối tag span không có xuống hàng (khác với div và p)

## 3. Tag Heading

Là các tag h1, h2, h3, h4, h5, h6. Các tag này gọi là tag Heading. Bạn thường dùng nó bao quanh các tiêu đề chương, tiêu đề bài... Chữ trong các tag này mặc định là đậm.

## 4. Tag nav

Thường dùng tag này bao quanh vùng menu, paper links, topic path. Ví dụ:

```
<nav>
<ul> <li><a href="#">Trang chủ</a></li>
      <li><a href="#">Giới thiệu</a></li>
      <li><a href="#">Sản phẩm</a></li>
      <li><a href="#">Liên hệ</a></li>
</ul>
</nav>
```

## 5. Tag header

Là tag định nghĩa những gì bên trong nó là phần đầu của trang web. Thường chứa giới thiệu, tiêu đề, thanh navigation. Ví dụ:

```
<header>
<h1> PHP cơ bản</h1>
<p>Kiến thức kiến thức cần thiết để làm website </p>
</header>
```

## 6. Tag Footer

Là tag định nghĩa những gì bên trong nó là phần cuối của trang. Thường chứa các thông tin liên lạc, tên tác giả, nguồn gốc của bài viết. Ví dụ:

```
<footer>
<p>Học PHP cơ bản</p>
<address>
Email: <a href="mailto:support@nhatnghe.com">support@nhatnghe.com</a>
<br />
Địa chỉ: 105 Bà Huyện Thanh Quan.<br /> Điện thoại: 028 9322 735
</address>
</footer>
```

## 7. Tag main

Là tag dùng để xác định nội dung chính của trang. Tag <main> thường sử dụng kết hợp với tag <header> & tag <footer> để tạo thành cấu trúc của 1 trang web hoàn chỉnh. Mỗi trang web chỉ nên có 1 tag main.

```
<head>
...
</head>
<body>
  <header>
    ...
  </header>
  <main>
    ...
  </main>
  <footer>
    ...
  </footer>
</body>
```

## 8. Tag section

Tag <section> định nghĩa một khu vực (vùng bao) trong trang web. Thường section để quy định các vùng trong 1 trang web rộng.

```
<section>
<h2>Tiêu đề section</h2>
<p>Nội dung section</p>
</section>
```



## D. CASCADING STYLE SHEET (CSS)

### I. Giới thiệu

- CSS là 1 kỹ thuật dùng để định nghĩa cách thể hiện của các đối tượng trong trang web. CSS cho phép bạn định dạng các đối tượng với rất nhiều đặc điểm (thuộc tính) mở rộng mà HTML thông thường không có. Bạn không thể định dạng 1 trang web cho đẹp khi không có sự am hiểu nhiều về kỹ thuật CSS.
- Style: Là 1 tập hợp các đặc điểm định dạng .
- Sheet: Là 1 tập hợp nhiều style
- Vị trí lưu style: Các style có thể lưu trong trang web hiện hành hoặc ở file riêng bên ngoài (.css). Nếu đặt trong trang web thì các style được đặt trong tag <style>. Nếu style được đặt ở file bên ngoài thì dùng tag link để nhúng vào trang web.

### II. Tạo Style

#### 1. Định dạng cho 1 tag trong toàn trang

##### a. Tên style

Để định dạng cho tag nào đó trong toàn trang (như định dạng cho các tag p trong trang chẳng hạn) thì bạn tạo style giống như tên của tag muốn định dạng. Cụ thể như muốn định dạng cho tag p thì tạo style tên là **p**

##### b. Ví dụ:

```
<style>  
p { font-size:20px; text-align:justify; }  
a {color: red; text-decoration: none}  
</style>
```

## 2. Định dạng cho 1 tag cụ thể có tên riêng

Với mỗi tag trong trang, bạn có thể đặt cho nó 1 tên riêng nào đó (thông qua thuộc tính id của nó)

Ví dụ :  ➔ **hinh1** là tên riêng của tag img này.

### a. Tên style

Với 1 tag có tên riêng, bạn định dạng cho nó bằng cách tạo 1 style như sau:

**#TênRiêng** (Là dấu # kèm với tên riêng của tag muốn định dạng)

### b. Ví dụ:

Muốn định dạng cho 1 tag có tên riêng là quangcao thì tạo style như sau:

```
#quangcao {width:500px; background-color:red; }
```

## 3. Định dạng cho tag bên trong 1 đối tượng có tên riêng

### a. Tên style

**#TênRiêng TagCon** (là #TênRiêng kèm với tag bên trong muốn định dạng)

### b. Ví dụ

Định dạng tag **img** trong 1 tag có tên riêng là **quangcao** thì tạo style là

```
#quangcao img {width:200px ; }
```

## 4. Class

Class là 1 cái tên bạn dùng để đại diện cho 1 nhóm các thuộc tính định dạng. Thường dùng class để định dạng cho những tag nằm rời rạc (xa nhau) nhưng muốn định dạng giống nhau, như tiêu đề các box chẳng hạn.

#### a. Tên style

**.TênClass** (Là “.” kèm với tênClass)

#### b. Ví dụ

```
.tieude {color: blue; }
```

#### c. Set Class

Để ấn định class cho 1 tag nào đó, bạn thêm thuộc tính class. Ví dụ :

```
<li class="active"><a href=#>Trang chủ</a></li>
```

Chú ý: Bạn cũng có thể ấn định nhiều class cho 1 tag cũng được

```
<li class="active menu"><a href=#>Trang chủ</a></li>
```

### III. Nhúng file css bên ngoài vào trang web

Các style định dạng css bạn có thể đặt ở 1 file (có tên mở rộng .css) mà không nhất thiết phải đặt trong trang web. Khi đó để nhúng file .css vào trang html thì bạn dùng tag link

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

### IV. Định dạng trực tiếp css trong tag html

Bạn có thể định dạng 1 tag html nào đó bằng cách dùng thuộc tính style trong mở đầu của tag đó. Xem ví dụ sau:

```
<div id="tinxemnhiều" style="background-color:darkblue; "> ... </div>
```

### V. Các thuộc tính CSS

#### 1. Các thuộc tính thường dùng để định dạng text

- font-family: Font chữ. Mỗi lần 3 font để dự phòng máy user không có font.

- font-size : Cỡ chữ. Có thể dùng các đơn vị px, em
- font-weight : độ đậm của chữ. Thường dùng **Normal** và **Bold**.
- font-style: chữ nghiêng hay không. Thường dùng *italic*
- text-transform: Biến đổi thành chữ hoa/chữ thường. Thường dùng Uppercase, lowercase, capitalize
- line-height: độ cao của 1 hàng chữ trong paragraph. (>100% là giãn ra, <100% là co lại)
- text-decoration: cách gạch ngang chữ. Thường dùng none, underline
  - Underline: gạch dưới chữ
  - None: không gạch
- color: màu chữ.

## 2. Các thuộc tính thường dùng để định dạng background

- background-color: màu nền.
- background-image:url(DiachiHinh) : ảnh nền
- background-repeat: cách lặp lại ảnh nền (repeat-x, repeat-y, repeat: 2 chiều)
- background-attachment: (fixed, scroll) ảnh nền sẽ cuộn theo chữ hay cố định. Mặc định là scroll.
- background-position: vị trí ảnh nền theo chiều ngang/dọc.

## 3. Các thuộc tính thường dùng để định dạng cho khối chữ

- word-spacing: Khoảng cách giữa các từ.
- letter-spacing: Khoảng cách giữa các ký tự
- text-align: (left, right, center, justify) cách canh chữ

- white-space: (normal/pre/nowrap) bình thường/ giữ nguyên định dạng/không cuộn chữ xuống hàng
- display: (block, none, inline, table): cách hiển thị 1 tag
  - block: quy định tag sẽ là 1 khối tách biệt với những cái khác ở trên và dưới nó. Giống paragraph. Có thể apply cho link để toàn chiều rộng có thể click thay vì user chỉ click vào text của link
  - inline: các tag được định dạng display:inline sẽ nằm cùng 1 hàng. Vd: các li cùng 1 hàng, để làm menu ngang.
  - none: tag bị ẩn, không xuất hiện trong trang web.

#### 4. Các thuộc tính thường dùng để định dạng 1 khối

- width: độ rộng. Có thể dùng đơn vị px hoặc %.
- height: độ cao. Có thể dùng đơn vị px hoặc %.
- float: (left, right) Cho đối tượng trôi qua trái hoặc qua phải.
- clear: (left, right, both). Thuộc tính bỏ qua float của tag phía trên tag đang định dạng.
- padding: : Khoảng cách từ lề của đối tượng với nội dung bên trong
- margin: Khoảng cách từ lề của đối tượng với những đối tượng bên ngoài

#### 5. Các thuộc tính thường dùng để định dạng đường viền

- border-style: (solid, dash, dotted, double) kiểu đường viền.
- border-width: độ dày. Có thể dùng thick, medium, thin hoặc 1 con số
- border-color: Màu đường viền.

## 6. Các thuộc tính định dạng danh sách list

- list-type: (*disc, circle, square, decimal, lower-roman, upper-roman, lower-alpha, upper-alpha*) Kiểu bullet của danh sách
- list-type-image:url(DiaChiHinh) hình dùng thay thế ký tự bullet
- list-type-position: (*inside, outside*) vị trí của ký hiệu bullet (ở ngoài hay bên trong lề trái của các item)

## 7. Các thuộc tính định dạng vị trí

Các thuộc tính liên quan đến vị trí, kích thước đối tượng ...

- position: (*static, relative, absolute, fixed*): cách đặt đối tượng theo tọa độ top, left, right, bottom
- top: tọa độ dọc của cạnh trên
- left: tọa độ trái
- z-Index: để cho tag được định dạng nằm trên hoặc dưới so với những tag khác (tag nào có giá trị i\_index cao hơn sẽ nằm trên)

## VI. Định dạng liên kết

Tên style	Ý nghĩa
<b>A:link</b>	định dạng cho tất cả liên kết chưa được nhấp trong trang
<b>A:visited</b>	định dạng cho tất cả liên kết đã được nhấp trong trang
<b>A:hover</b>	định dạng cho liên kết trong trang đang được đưa chuột vào
<b>A</b>	định dạng cho tất cả liên kết trong trang
<b>#menu A:link</b>	định dạng liên kết chưa được nhấp trong vùng có tên là menu
<b>#menu A:visited</b>	định dạng liên kết đã được nhấp trong vùng có tên là menu
<b>#menu A:hover</b>	định dạng liên kết đang đưa chuột vào trong vùng có tên là menu
<b>#menu A</b>	định dạng cho tất cả liên kết trong vùng có tên là menu

## E. HOSTING VÀ DOMAIN

### I. Khái niệm

#### 1. Hosting

Hosting là chỗ trên mạng internet để chứa các trang web của bạn. Để có hosting thì bạn phải mua chỗ. Nếu không muốn mua chỗ thì bạn có thể đăng ký ở những chỗ cho miễn phí (nhưng bạn sẽ bị hạn chế nhiều thứ, như tốc độ, các tính năng hay, thời gian...)

#### 2. Domain

Domain (hay tên miền) là địa chỉ website của bạn. Ví dụ: **abc.com** hay **abc.com.vn**. Để có được địa chỉ này thì bạn phải mua. Điều kiện để mua (ngoài tiền) là tên domain mà bạn muốn chưa có ai mua. Thường thì tiền mua domain sẽ trả theo năm, hết hạn phải gia hạn tiếp. Nếu không thì bạn sẽ bị mất domain và người khác có thể mua tên đó.

Mỗi domain sẽ được trở vào 1 hosting do bạn đã mua trước đó. Và bạn có thể mua nhiều domain để trở vào một website.

### II. Tìm mua domain và hosting

Ở Việt nam hiện có rất nhiều nhà cung cấp dịch vụ domain và hosting. Ví dụ như mắt bão, pa viet nam, nhân hòa, và nhiều nơi khác nữa. Thường để cho tiện thì bạn nên mua hosting và domain ở cùng một nhà cung cấp, để họ setup hết các thông số cho bạn.

### III. Những điều phải biết khi mua hosting

#### 1. Những điều phải biết

Khi đăng ký mua hosting ở đâu đó, bạn phải quan tâm đến các thông tin sau:

- Gói hosting bạn định mua chạy hệ điều hành nào? Bạn phải chọn Linux nhé, vì chúng ta đang dùng PHP để làm web động.
- Gói hosting bạn định mua chạy php version mấy. Ít nhất phải version php 5.5 trở về sau.

- Gói hosting bạn định mua chạy cho tạo bao nhiêu database mysql? Điều này quan trọng để bạn sử dụng database sau này. Ít nhất phải là 1 database, nếu bạn không để ý điều này mà mua gói rẻ, nó không cho tạo database thì khổ.
- Gói hosting bạn định mua cho phép bạn lưu thông tin với dung lượng là bao nhiêu? Dung lượng hosting gồm tổng kích thước file trong website, kích thước database và mail đó nhé.
- Gói hosting bạn định mua cho phép băng thông là bao nhiêu? Có gói không hạn chế nhưng nhiều gói là hạn chế đó nha. Băng thông là tổng dung lượng dữ liệu mà server sẽ trả về cho các người xem trong 1 tháng. Nếu trong tháng mà server đổ về đủ băng thông thì nó ngưng lại.
- Bạn mua hosting ở đâu thì khả năng support của kỹ thuật là rất quan trọng. Việc support của họ bạn rất cần khi muốn được giúp đỡ để hiệu chỉnh hay hỏi han gì đó từ nhà cung cấp hosting.
- Khả năng bảo mật của nhà cung cấp hosting cũng là thông tin cần quan tâm, Tuy nhiên yếu tố này cần tới niềm tin của bạn nhiều hơn.
- Backup data: Việc backup dữ liệu cho website của bạn được định kỳ bao lâu 1 lần? Đề khi cần lấy lại dữ liệu cũ, bạn có thể yêu cầu.

## 2. Quản trị hosting

Khi mua hosting xong, bạn sẽ được nhà cung cấp tạo cho 1 tài khoản và mật khẩu. Họ sẽ gửi cho bạn địa chỉ server + tài khoản + mật khẩu để bạn đưa website của bạn lên chỗ của họ.

Những gì phải làm khi quản trị hosting? Khi quản trị hosting, những việc sau đây bạn thường hay làm:



- Quản trị File: bao gồm upload file, tạo file, xóa file, đổi tên, di chuyển, bung nén, nén file...
- Quản trị Database: bao gồm tạo database, tạo user kết nối database, ấn định quyền truy cập vào database, import và export dữ liệu.
- Quản trị FTP: bao gồm tạo các user để ftp từ các chương trình FTP client như FlashFXP, WS\_FTP, WinSCP...
- Quản trị domain và subdomain
- Quản trị các mail box

### 3. Upload website lên mạng internet

Việc upload lên mạng bạn sẽ có lab riêng. Bạn thực thi theo lab riêng của Thầy cho.

## IV. Mua domain

Bạn có thể mua domain từ các nhà cung cấp trong nước hoặc nước ngoài. Khi mua xong domain, bạn cũng sẽ được cung cấp username, password, và địa chỉ website để vào quản lý domain.

Bạn làm gì để quản lý domain? Thường làm nhất là trỏ tên miền đến nhà cung cấp hosting của bạn. Ngoài ra còn 1 số các thao tác khác liên quan đến việc tạo/chỉnh các MX record, Cname record...

## F. FORM

Form dùng để tạo các biểu mẫu thu thập thông tin của người xem trang web. Một Form có thể chứa nhiều thành phần như TextField, CheckBox, RadioButton, ComboBox, List/Menu, TextArea...

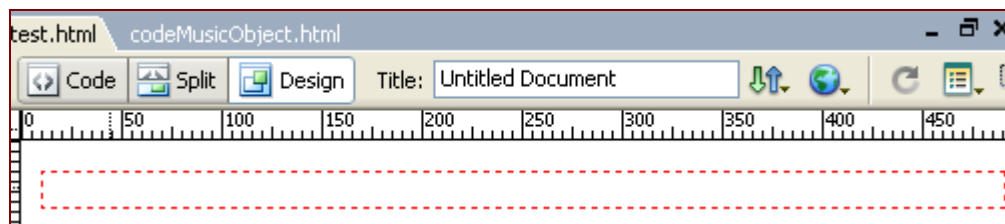
ĐĂNG KÝ THÀNH VIÊN CLB YÊU NHẠC

Họ tên	Nguyễn Văn Tèo
Tên đăng nhập	teonv
Mật khẩu	.....
Gỡ lại mật khẩu	.....
Phái	<input checked="" type="radio"/> Nam <input type="radio"/> Nữ
Ngoại ngữ	<input checked="" type="checkbox"/> Anh <input type="checkbox"/> Pháp <input type="checkbox"/> Hoa
Nghề nghiệp	Học sinh-Sinh viên
Hình	missonha.jpg <input data-bbox="1036 951 1166 982" type="button" value="Browse..."/>
Sở thích	Ngắm trăng, uống trà, vuốt râu, nhìn mưa rơi, nghe chim hót, ngắm thơ
<input data-bbox="735 1129 841 1161" type="button" value="Đăng ký"/> <input data-bbox="857 1129 963 1161" type="button" value="Làm lại"/>	

- Mỗi form **phải có ít nhất** 1 nút đóng vai trò **submit** (đưa dữ liệu lên server)
- Mỗi form **có thể có** 1 nút giúp xóa hết dữ liệu được gõ trong form (nút Reset)
- Mỗi form nằm trong 1 tag form. Tag form có thuộc tính method (post, get : là phương thức đưa dữ liệu lên server) và thuộc tính action (là trang web sẽ tiếp nhận dữ liệu từ form gửi lên)
- Một trang web có thể có nhiều Form nhưng các form không được lồng nhau.

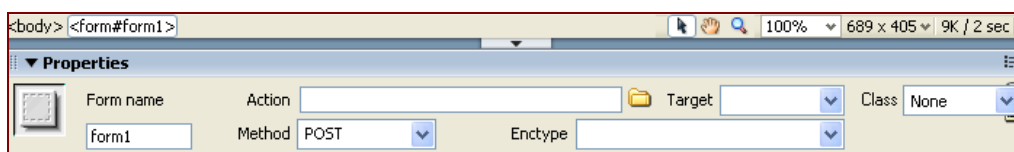
## I. Chèn form

Nhấp vị trí muốn chèn form rồi vào menu Insert ➔ Form ➔ Form



## II. Các thuộc tính của Form

Chọn tag form trên thanh tag, khi đó trên thanh Properties sẽ hiện các thuộc tính:



- **Form Name:** là tên của form, duy nhất, không khoảng trắng
- **Action:** tên của file tiếp nhận dữ liệu của Form (URL).
- **Method:** Là phương thức truyền dữ liệu lên server. Có 2 phương thức: Post và get

**Get:** Kết hợp theo từng cặp tên-giá trị vào phía sau URL của file nhận dữ liệu. Ví dụ: action là **login.php** thì khi submit sẽ ghép như sau **login.php?username=abc&pass=123**

**Post :** được dùng nhiều hơn Get. Với Post, dữ liệu được ẩn đi khi truyền về server (cũng theo theo dạng key-value).

- **Enctype:** có giá trị là :

**Application :** mặc định, không cần gán

**Mutipart:** dùng khi form có 1 file cần upload. Mỗi lần thêm 1 file field vào form, dw sẽ tự động chỉnh entype thành multipart

### III. Các thành phần trên Form

#### 1. Text field

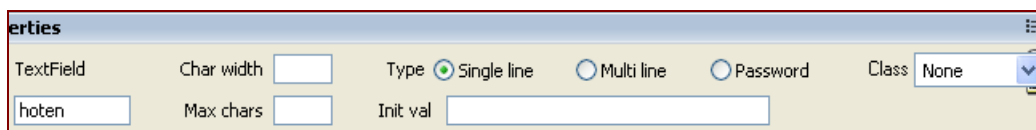
- Là thành phần trong form cho phép user gõ 1 hàng chữ
- Cách chèn: Nhấp vị trí muốn chèn rồi vào menu Insert ➔ Form ➔ Text Field

**ID:** Tên của text field

**Label:** Nhãn của text field

**Style:** cách đặt nhãn.

- Thuộc tính:



**Char width:** độ rộng text field tính theo ký tự

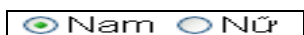
**Max chars:** Số ký tự tối đa mà user có thể nhập

**Type:** Single line/Multi line/Password ~ 1 hàng chữ/nhiều hàng chữ/mật khẩu

**Init val:** Giá trị ban đầu.

#### 2. Radio group

- Sử dụng khi muốn tạo nhóm các mục chọn và chỉ cho phép user chọn 1 mục



- Cách chèn: Nhấp vị trí muốn chèn rồi vào menu Insert ➔ Form ➔ Radio Group

**Name:** Tên chung của các mục

**Label:** Nhãn của từng mục

**Value:** Giá trị của mỗi mục

**Layout:** chọn xuống hàng dùng br hoặc table

### 3. Check Box

- Sử dụng khi muốn tạo nhiều mục và user có thể chọn nhiều mục.

☒ Anh ☐ Pháp ☐ Hoa

- Cách chèn: Nhấp vị trí trong form muốn chèn rồi vào Insert ➔ Form ➔ Checkbox

### 4. List/Menu

- List/menu chứa nhiều item (mục) bên trong, user có chọn hoặc nhiều mục trong box.
- Cách chèn: Nhấp vị trí (trong form) muốn chèn vào menu Insert ➔ Form ➔ List/Menu
- Thuộc tính:

Properties

List/Menu Type: ☒ Menu Height: 1 List Values... Class: None

ngheghiep ☐ List Selections: ☐ Allow multiple

Initially selected: Học sinh - Sinh viên  
Giáo viên  
Công nhân viên

**List Values:** Thêm bớt mục (name, value). Ví dụ: Thêm các item sau:

List Values

Item Label Value

Học sinh - Sinh viên	1
Giáo viên	2
Công nhân viên	3
Khác	4

OK Cancel

**Type:** cách hiển thị

**List:** box có thể cao nhiều hàng. User có thể chọn nhiều

**Menu:** box chỉ cao 1 hàng. Khi user nhấp vào sẽ xổ xuống các mục bên trong. User chỉ chọn được 1 mục.

**Height:** Chiều cao box

**Allow multiple:** cho phép user chọn nhiều mục

- Chú ý:

- Chiều rộng của box tự co giãn theo item dài nhất, nhưng có thể fix độ rộng = css
- Nếu xem code html, bạn sẽ thấy mỗi item nằm trong tag option. Có value và text
- Thêm giá trị **selected** trong 1 option để chỉ định item được chọn mặc định

## 5. File field

- File Field là đối tượng giúp user chọn file trên máy của họ để đưa lên server.
- Cách chèn: Nhấp vị trí muốn chèn rồi vào menu Insert → Form → File Field

## 6. Text area

- Là thành phần trong form cho phép user gõ nhiều hàng chữ
- Cách chèn: Nhấp vị trí muốn chèn rồi vào menu Insert → Form → Text Area

ID: Tên của Text Area

Label: Nhãn của Text Area

- Thuộc tính:

The screenshot shows a 'Properties' window for a 'TextField' control. The 'Char width' is set to 45, 'Num lines' is 5, 'Type' is 'Multi line' (selected), 'Password' is unselected, 'Class' is 'None', and 'Wrap' is 'Default'. The 'ID' field contains the text 'sothich'. There is also an 'Init val' field which is currently empty.

Char width: độ rộng tính theo ký tự

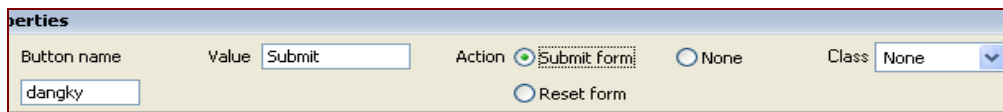
Num lines: độ cao tính theo số dòng text

Init val: Giá trị ban đầu.

## 7. Button

- Button là nút lệnh, có thể dùng để gửi dữ liệu trong form lên server (submit), xóa dữ liệu trong form (reset), hoặc thực hiện 1 tác vụ nào đó.
- Cách chèn: Nhấp vị trí (trong form) muốn chèn rồi vào menu Insert → Form → Button

- Thuộc tính:



Value: Dòng chữ trong nút.

Action: chức năng của nút submit, reset, none

## 8. Image Field

- Image Field là hình đóng vai trò như nút submit, dùng để gửi dữ liệu trong form lên server. Bạn có thể dùng image field thay thế cho nút submit nếu muốn có định dạng đẹp hơn trong cách trình bày form.
- Cách chèn: Nhấp vị trí muốn chèn, vào menu Insert → Form → Image Field → Chọn file.
- Chú ý: Đối tượng này là tag input, không phải là hình thông thường (*Hình thông thường nằm trong tag img*) và không có Image field đóng vai trò reset

## 9. Hidden field

- Hidden field là đối tượng ẩn, user không thấy, chỉ người lập trình mới biết.
- Cách chèn: Nhấp vị trí muốn chèn rồi vào menu Insert → Form → Hidden Field
- Thuộc tính: **Hidden Field**: Tên của hidden field

**Value**: Giá trị của hidden field

## G. JAVASCRIPT

### I. Giới thiệu

- Là ngôn ngữ lập trình chạy trên browser, nhằm tạo khả năng tương tác với user.
- Phân biệt chữ hoa, chữ thường
- Mỗi lệnh kết thúc bằng dấu ;
- Ghi chú: 1 dòng //
- Ghi chú nhiều dòng : /\* ..... \*/
- Có thể viết mã lệnh javascript trong tag script của trang, hoặc viết riêng trong file .js rồi nhúng vào trang html, hoặc viết mã lệnh javascript trong sự kiện của các tag html.

### II. Viết mã Javascript vào trang

#### 1. Viết trực tiếp trong trang với tag script

```
<script>
    alert("Đã thêm sản phẩm vào giỏ hàng");
</script>
```

#### 2. Viết mã javascript trong file riêng

##### a. Tạo file javascript

- Nhấp menu File → New → Javascript → Create
- Gõ mã lệnh javascript . Ví dụ: gõ

```
hoten=prompt("Bạn ơi bạn tên gì?");
alert("Chào bạn " + hoten);
```

- Lưu file với tên mở rộng là .js

##### b. Liên kết file js đến trang html

- Mở 1 file html.



- Nhấp vị trí muốn chèn (thường trong tag head) rồi vào Insert → HTML → Script Object → Script → Chọn file javascript

### 3. Sử dụng javascript trong sự kiện của các tag

Mỗi tag đều có một số sự kiện, bạn có thể sử dụng javascript khi các sự kiện này xảy ra.

Mở file js1.html



#### a. Đưa chuột vào hình trái tim, sẽ đổi thành hình khác, chuột ra trở lại hình cũ

Nhấp hình trái tim rồi qua code, bổ sung vào tag img như sau :

```

```

#### b. Nhấp chuột vào textfield tìm kiếm, chữ tìm kiếm trong đó sẽ biến mất

Nhấp text field tìm rồi qua code, bổ sung để được như sau :

```
<input onclick="this.value=''" onblur="this.value=''" name=tim type=text id=tim
value="Tìm kiếm" />
```

### 4. Lab sử dụng một số javascript hữu dụng

Bạn mở file **js\_HuongDan.doc** để xem

## 5. Website javascript

- [www.dynamicdrive.com](http://www.dynamicdrive.com): Nhiều javascript hay
- [www.w3schools.com](http://www.w3schools.com): Nhiều bài học, ví dụ rất dễ hiểu và javascript, php, css, xml...
- <http://www.htmlgoodies.com/>

## III. Lập trình javascript

### 1. Các sự kiện cơ bản trong Javascript

- onClick → Khi Click chuột vào đối tượng
- onMouseover → Khi con trỏ vào đối tượng
- onMouseout → Khi con trỏ ra ngoài đối tượng
- onKeyPress → Khi nhấn phím
- onSubmit → Khi submit form
- onChange → Khi thay đổi dữ liệu trong list/menu
- onBlur → Khi chọn đối tượng khác

### 2. Biến

Biến dùng để chứa một giá trị, biến có thể thay đổi giá trị sau khi khai báo

```
<script>
var i = 100; // kiểu số nguyên
var j = 1.5; // kiểu số thực (double)
var str="Lập trình Javascript"; // biến kiểu chuỗi
var n= new Date;
</script>
```

### 3. Các phép toán trên các biến

- Các phép toán số học: +, -, \*, /, ++, --, %

- Phép gán: =
- Các phép toán so sánh: ==, !=, >, <, >=, <=
- Các phép toán logic : &&, ||, !

#### 4. Phát biểu if...else

Biểu thức	Ví dụ
<pre>if(điều kiện) {     Các lệnh thực hiện khi điều kiện đúng }else{     Các lệnh thực hiện khi điều kiện sai }</pre>	<pre>&lt;script&gt; var n=5; if(n%2==0) {     document.write(n + " Là số chẵn"); } else{     document.write(n + " Là số lẻ"); } &lt;/script&gt;</pre>

#### 5. Vòng lặp for

Biểu thức	Ví dụ
<pre>for(i=0;i&lt;10;i++) {     Các lệnh được lặp }</pre> <p><i>/*</i> <i>i=0: giá trị khởi tạo</i> <i>i&lt;10: biểu thức điều kiện</i> <i>i++: Tăng biến đếm lên 1 */</i></p>	<pre>&lt;script&gt; for (i=0;i&lt;10;i++){     document.write(i + "&lt;br/&gt;"); } &lt;/script&gt;</pre>

#### 6. Vòng lặp while

Biểu thức	Ví dụ
<pre>while(điều kiện) {</pre>	<pre>&lt;script&gt;</pre>

<p>Các lệnh được lặp</p> <pre> } /* kiểm tra điều kiện, nếu đúng thì lặp, sai thì kết thúc lặp */ </pre>	<pre> var i=0; while (i&lt;10) { document.write(i + "&amp;nbsp;"); i++; } &lt;/script&gt; </pre>
--	--

## 7. Vòng lặp do...while

Biểu thức	Ví dụ
<pre> do{ Các lệnh được lặp }while(điều kiện) /* Thực thi các lệnh trong vòng lặp rồi test điều kiện, nếu đúng thì lặp lại, nếu sai thì kết thúc*/ </pre>	<pre> &lt;script&gt; var i=0 do{ i++; document.write(i + "&amp;nbsp;"); }while(i&lt;50); &lt;/script&gt; </pre>

## 8. Truy xuất đối tượng trong trang web

- Truy xuất đối tượng theo id

```
document.getElementById('tênđối tượng')
```

- Truy xuất đối tượng trong form

```
document.tênform.tênđối tượng
```

- Ví dụ:

```

document.getElementById('thongbao').innerHTML="Username bạn chọn
đã có người dùng";
var username= document.formlogin.username.value;
var soluong= document.formchonsp.soluong.value;

```

## 9. Hàm trong javascript

Hàm là một nhóm lệnh để giải quyết một việc nào đó. Mỗi hàm có 1 tên. Khi gọi tên của hàm, các lệnh trong đó sẽ thực thi

- Tạo hàm mới:

```
function Tên hàm([tham so]) {  
    // Các câu lệnh javascript bên trong hàm  
    [return giá trị;]  
}
```

- Phần tham số có thể có hoặc không. Hàm có thể trả về 1 giá trị sau khi tính

```
function checkPassword(pass, repass) {  
    if (pass.value=='') {  
        alert('Bạn chưa nhập mật khẩu');  
        pass.select();  
        return false;  
    }  
    if (pass.value!=repass.value) {  
        alert('Mật khẩu không giống nhau. Mời nhập lại');  
        pass.select();  
        return false;  
    }  
    return true;  
}
```

## 10. Đối tượng window

Là đối tượng có sẵn trong Javascript để diễn tả cửa sổ hiện hành. Các hàm thường dùng:

<a href="#">alert()</a>	Hiện hộp thông báo có 1 nút OK
<a href="#">close()</a>	Đóng cửa sổ hiện hành
<a href="#">confirm()</a>	Hiện box với 1 thông báo, nút OK và Cancel

<a href="#">focus()</a>	Sets focus cho window
<a href="#">open()</a>	Mở 1 window mới
<a href="#">print()</a>	In cửa sổ hiện hành
<a href="#">prompt()</a>	Hiện hộp thoại để user nhập thông tin
<a href="#">setInterval()</a>	Dùng để thực thi 1 hàm sau 1 thời khoảng (milli giây)
<a href="#">setTimeout()</a>	Gọi 1 hàm sau 1 khoảng thời gian chỉ định (milli giây)

## 11. Đối tượng document

Là đối tượng có sẵn trong Javascript dùng để mô tả tài liệu hiện hành. Mỗi trang HTML khi nạp vào trình duyệt sẽ trở thành đối tượng document. Thông qua đối tượng này, bạn có thể truy xuất vào mọi thành phần HTML trong trang.

- Các Properties:

<a href="#">cookie</a>	Trả về tất cả các cookie trong tài liệu (name/value)
<a href="#">domain</a>	Trả về domain name của server nạp tài liệu
<a href="#">referrer</a>	Trả về URL của trang web liên kết đến trang hiện tại
<a href="#">title</a>	Set hoặc get tiêu đề của trang
<a href="#">URL</a>	url của tài liệu

- Các Methods

<a href="#">getElementById()</a>	Truy xuất đến phần tử theo id (phần tử đầu tiên)
<a href="#">getElementsByName()</a>	Truy xuất đến các phần tử theo tag

## 12. Đối tượng screen

Là đối tượng có sẵn trong Javascript, chứa thông tin về màn hình của user. Các thuộc tính

<a href="#">availHeight</a>	Cho biết độ cao của màn hình (không tính Windows Taskbar)
<a href="#">availWidth</a>	Cho biết độ rộng màn hình (không tính Windows Taskbar)
<a href="#">height</a>	Cho biết độ cao của màn hình
<a href="#">width</a>	Cho biết độ rộng của màn hình

Ví dụ:

```
<script>
document.write("<p>availHeight= " + screen.availHeight+"</p>");
document.write("<p>availWidth= " + screen.availWidth+"</p>");
document.write("<p>height= " + screen.height + "</p>");
document.write("<p>width= " + screen.width+ "</p>");
document.write("<p>pixelDepth= " + screen.pixelDepth+ "</p>");
</script>
```

### 13. Đối tượng history

Là đối tượng có sẵn trong Javascript dùng để lưu địa chỉ các trang web mà user đã xem. Dùng đối tượng này, bạn có thể chuyển tới trang trước , trang sau. Các phương thức:

<a href="#">back()</a>	Trở lại trang trước
<a href="#">forward()</a>	Trời trang sau
<a href="#">go()</a>	Nhảy đến trang chỉ định trong history list

Ví dụ:

```
<script> history.back(); </script>
```

### 14. Đối tượng location

Là đối tượng có sẵn trong Javascript chứa thông tin về URL hiện hành. Đây là con của đối tượng window và được truy xuất với cú pháp **window.location**. Các properties:



hostname	Cho biết hostname của URL
href	Trả về toàn thể URL
search	Trả về phần query string trong URL

- Các methods

reload()	Nạp lại trang hiện hành
----------	-------------------------

- Ví dụ:

```
<script>
document.write(location.host);
document.write(location.href);
</script>
```

## 15. Đối tượng Navigator

Là đối tượng chứa các thông tin về browser. Các Properties

<a href="#">appName</a>	Trả về tên của browser
<a href="#">appVersion</a>	Trả về version của browser
<a href="#">platform</a>	Trả về hệ điều hành mà browser đang chạy trên đó
<a href="#">userAgent</a>	Trả về user-agent mà browser gửi tới server

Ví dụ:

```
<script>
document.write("Name: <b>" + navigator.appName + "</b><br/>");
document.write("Version: <b>" + navigator.appVersion + "</b><br/>");
document.write("Platform: <b>" + navigator.platform + "</b><br/>");
document.write("User-agent: <b>" + navigator.userAgent + "</b><br/>");
</script>
```

---

## H. JQUERY

### I. Giới thiệu

- JQuery là một thư viện Javascript giúp đơn giản cách viết JavaScript trong trang web.
- JQuery có thể giúp định dạng lại và thay đổi thông tin trong trang web.
- JQuery có thể giúp bạn tạo nhiều hiệu ứng như mờ dần, chạy dọc chạy ngang v.v.. bạn có thể tạo thêm các hiệu ứng khác.
- jQuery cho phép tạo ra các Plugin nếu cần..

### II. Chèn jQuery vào trang

- Vào trang chủ của jQuery và download phiên bản mới nhất (file js), chép vào folder website của bạn, rồi insert vào trang web bằng tag script quen thuộc.
- Hoặc chèn JQuery từ site chính thức của JQuery:

```
<html>
<head>
<script src="http://code.jquery.com/jquery-latest.js"></script>
<script>
$(document).ready(function() {
    // Mã jquery của bạn
});
</script>
</head>
```

### III. Chọn phần tử trong trang

Thao tác cơ bản của JQuery là chọn các phần tử trong tài liệu HTML và thực hiện một việc gì đó bằng cách sử dụng hàm \$(). Lệnh cơ bản như sau:

**\$(query).action()**

- \$ là kí hiệu đặc biệt, xác định đây là câu lệnh jQuery.
- **action** là hàm sẽ tác động lên các phần tử được chọn (click, change...)
- **query** diễn tả phần tử sẽ chọn trong trang, có nhiều cách chọn

## 1. jQuery Selectors

Chọn các phần tử (selector) kiểu như CSS:

<code>\$("a")</code>	chọn tất cả các tag <a>
<code>\$(".div.intro")</code>	chọn tất cả các tag <div> có class là "intro".
<code>\$("#p#tieude")</code>	chọn tất cả các tag <p> có id là "tieude".
<code>\$(this)</code>	Diễn tả phần tử hiện hành
<code>\$(".tensp")</code>	Chọn các phần tử có class là tensp
<code>\$("#left")</code>	Chọn phần tử có tên là left

## 2. jQuery Attribute Selectors

jQuery có thể chọn các phần tử với các thuộc tính cho trước.

<code>\$("[href]")</code>	chọn tất cả các phần tử có thuộc tính href.
<code>\$("[href='#']")</code>	chọn tất cả các phần tử với href ="#".
<code>\$("[href!='#']")</code>	chọn tất cả các phần tử với chuỗi href khác "#".
<code>\$("[src\$='.jpg']")</code>	chọn tất cả các phần tử mà src chứa ".jpg".

## III. Một số hàm thường dùng trong JQuery

### a. **addClass**: Thêm một class CSS

Để set class cho đối tượng, dùng lệnh **addClass**. Để bỏ class, dùng **removeClass**. Ví dụ:

```
$("#tinxemnhieu span").addClass("caption");  
$("#tinxemnhieu p").addClass("rows");
```

## b. **removeClass**: gỡ một class CSS

```
$("#tinxemnhieu a").removeClass("abc");
```

## c. **css** : định dạng css cho tag được chọn.

Cú pháp: **\$(selector).css(name,value)** hoặc **\$(selector).css({properties})**

```
$("#[tieude]").css("color", "#003399");  
$("#[tieude]").css("font-size", "24px");  
$("#[tieude]").css({"margin": "0", "padding": "5px", "background-color": "#FFFFCC"});
```

Tham khảo thêm: [http://www.w3schools.com/jquery/jquery\\_ref\\_css.asp](http://www.w3schools.com/jquery/jquery_ref_css.asp)

d. **html** : Thay đổi nội dung (thuộc tính innerHTML) của thành phần được chọn.

e. **append**: Thêm nội dung vào innerHTML (ở cuối) của thành phần được chọn

f. **prepend**: Thêm nội dung vào innerHTML (ở đầu) của thành phần được chọn

g. **after**: Chèn nội dung vào ở trước thành phần được chọn

h. **before**: Chèn nội dung vào ở phía sau thành phần được chọn

Ví dụ:

```
$("#tinnoibat span").html("Tin nổi bật");  
$("#tinnoibat").prepend("<p><a href=#>Cá sấu cắn voi</a></p>");  
$("#tinnoibat").append("<p><a href=#>Loài cá biết bay </a></p>");  
$("#tinnoibat").before("<hr/>");  
$("#tinnoibat").after("<hr/>");
```

Tham khảo thêm: [http://www.w3schools.com/jquery/jquery\\_ref\\_html.asp](http://www.w3schools.com/jquery/jquery_ref_html.asp)

## IV. Hiệu ứng trong jquery

Jquery giúp bạn tạo các hiệu ứng như Hide, Show, Toggle, Slide, Fade, and Animate. (Ẩn, hiện, bật tắt, trượt, mờ đi và hoạt hình).

Các hàm tạo hiệu ứng trên nhận 2 đối số: speed và callback. Cú pháp như sau:

<code>\$(selector).hide(speed, callback)</code>	→ Ẩn phần tử
<code>\$(selector).show(speed, callback)</code>	→ Hiện phần tử
<code>\$(selector).toggle(speed, callback)</code>	→ Ẩn hiện phần tử
<code>\$(selector).slideDown(speed, callback)</code>	
<code>\$(selector).slideUp(speed, callback)</code>	
<code>\$(selector).slideToggle(speed, callback)</code>	
<code>\$(selector).fadeIn(speed, callback)</code>	→ Mờ dần
<code>\$(selector).fadeOut(speed, callback)</code>	→ Tò dần
<code>\$(selector).fadeTo(speed, opacity, callback)</code>	→ Làm cho trong suốt
<code>\$(selector).animate({params}, [duration], [easing], [callback])</code>	→ Tạo hiệu ứng động

**speed** có thể là "slow", "fast", "normal", hoặc số milligiây

**callback** tên của hàm sẽ thực thi sau khi hiệu ứng thực thi xong

**params:** Thuộc tính được biến đổi. Ví dụ:  
`animate({width:"70%",opacity:0.4,marginLeft:"0.6in",fontSize:"3em"})`

**duration:** Thời gian diễn ra hiệu ứng: "fast", "slow", "normal", hoặc miligiây.

Tham khảo thêm: [http://www.w3schools.com/jquery/jquery\\_ref\\_effects.asp](http://www.w3schools.com/jquery/jquery_ref_effects.asp)

Ví dụ 1:

```
$("#btnan").click(function() { $("#tinxemnhieu").hide(2000); } );
```

Ví dụ 2:

```
$("#bttnhien").click(function() { $("#tinxemnhieu").show(3000); } );
```

Ví dụ 3:

```
$("#bttnanhien").click(function() {  
    $("#tinxemnhieu").toggle(1000);  
});
```

Ví dụ 4:

```
$("#btnup").click(function() {  
    $("#tinxemnhieu").slideUp(1000);  
});  
$("#btndown").click(function() {  
    $("#tinxemnhieu").slideDown(1000);  
});  
$("#btnupdown").click(function() {  
    $("#tinxemnhieu").slideToggle(1000);  
});
```

## V. Sự kiện trong JQuery

Dưới đây là một số hàm xử lý sự kiện phổ biến, hay dùng trong jquery

```
$(document).ready(function) : Sự kiện trang web nạp xong  
$(selector).click(function) : nhấp vào phần tử được chọn  
$(selector).dblclick(function) : nhấp đúp phần tử được chọn  
$(selector).focus(function) : phần tử được chọn có focus  
$(selector).mouseover(function): đưa chuột vào phần tử được chọn  
$(selector).mouseout(function): đưa chuột ra khỏi p.tử được chọn  
$(selector).change(function): chuyển giá trị của phần tử được chọn  
$(selector).blur(function) : phần tử được chọn mất focus
```

**Có thể có sự xung đột về tên:** jQuery sử dụng ký hiệu \$ như là cách phân biệt mã jQuery với code javascript thông thường. Nhưng một số thư viện khác (như Prototype) cũng dùng ký hiệu này. jQuery có một phương thức gọi là **noConflict** nhằm khắc phục vấn đề này.

```
var $jq=jQuery.noConflict();
```

⇒ Khi đó JQuery sẽ sử dụng ký hiệu \$jq thay cho \$ mặc định

Tham khảo thêm: [http://www.w3schools.com/jquery/jquery\\_events.asp](http://www.w3schools.com/jquery/jquery_events.asp)

## VI. JQuery Ajax

AJAX (*Asynchronous JavaScript and XML*) là 1 kỹ thuật để cập nhật một phần của trang web với một lượng nhỏ dữ liệu mà không cần phải nạp lại toàn bộ trang. Với jQuery AJAX, bạn có thể các request lên server để lấy dữ liệu. **Các hàm tạo jQuery AJAX Requests**

Hàm	Mô tả
<code>\$(selector).load(url,data,callback)</code>	Tải dữ liệu từ server vào phần tử được chọn
<code>\$.ajax(options)</code>	Tải dữ liệu từ server vào XMLHttpRequest object
<code>\$.get(url,data,callback,type)</code>	Tải dữ liệu từ xa sử dụng HTTP GET
<code>\$.post(url,data,callback,type)</code>	Tải dữ liệu từ xa sử dụng HTTP POST
<code>\$.getScript(url,callback)</code>	Tải và thực thi 1 JavaScript file

**selector** Phần tử được chọn

**url** Địa chỉ trang web sẽ request dữ liệu

**data** : Dữ liệu dạng cặp đôi Key/value sẽ gửi lên server

**callback**: hàm sẽ thực thi khi request xong

**type** Kiểu của dữ liệu sẽ send (html,xml,json,jsonp,script,text)



**options:** Tất cả các cặp giá trị key/value để tạo request như: định nghĩa url, datatype, filter, charset, timeout, error function

## VII. Một số Plugin JQuery hữu dụng

1. Cycle Plugin : <http://malsup.com/jquery/cycle/>
2. jqGrid Plugin : <http://www.trirand.com/blog/> Trình bày dữ liệu dạng bảng, có phân trang, tìm kiếm...
3. JQuery Media: <http://jquery.malsup.com/media/#download>
4. AnythingSlider: <http://css-tricks.com/examples/AnythingSlider/#panel-3>
5. Creating a Slick Auto-Playing Featured Content Slider  
<http://css-tricks.com/creating-a-slick-auto-playing-featured-content-slider/>
6. Create Featured Content Slider Using jQuery UI  
<http://webdeveloperplus.com/jquery/featured-content-slider-using-jquery-ui/>
7. s3Slider jQuery plugin: <http://www.serie3.info/s3slider/>

## I. HTML5 CSS3, RESPONSIVE

### I. HTML5

Html5 ra đời năm 2014 và hiện là phiên bản html mới nhất để làm web. HTML5 được bổ sung thêm nhiều tag mới, nhiều tính năng mới so với html4.

#### 1. Những gì mới trong html5

Khai báo doctype đơn giản hơn như sau ở đầu mỗi trang html5

```
<!DOCTYPE html>
```

Khai báo charset cũng đơn giản hơn trong tag head

```
<meta charset="UTF-8">
```

#### 2. Cấu trúc 1 trang html5

```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Title of the document</title>
</head>

<body>
Content of the document.....
</body>
</html>
```

#### 3. Những element mới trong html5 và những tag bị loại bỏ

Những tag mới cho layout: <header>, <footer>, <article>, and <section>.

Những form elements mới : number, date, time, calendar, range.

Thành phần đồ họa : <svg> and <canvas>.

Những tag media : <audio> , <video>.

Những tag cũ không còn dùng trong html5

Bỏ tag <acronym> thay bằng <abbr> , bỏ tag <applet> thay bằng <object>, bỏ tag <dir> thay bằng <ul>

Bỏ tag hoàn toàn các <basefont>, <big> , <center> , <font> , <frame> , <frameset> , <noframes> , <strike>, <tt>. Tác dụng của tag này đã có sẵn trong css.

#### 4. HTML5 Form Elements

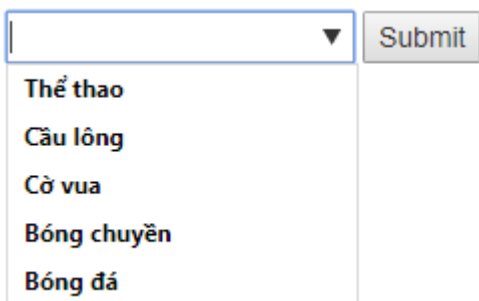
HTML5 thêm 2 tag form element mới : <datalist> , <output>

##### a. <datalist>

Tag <datalist> chứa danh sách các mục cho 1 tag input. Người dùng sẽ thấy các giá trị từ danh sách này và chọn thay vì phải nhập giá trị.

Thuộc tính **list** của tag <input> phải giống với thuộc tính **id** của <datalist>

```
<form action="">
  <input list="lstloaitin" name="loaitin">
  <datalist id="lstloaitin">
    <option value="Thể thao">
    <option value="Cầu lông">
    <option value="Cờ vua">
    <option value="Bóng chuyền">
    <option value="Bóng đá">
  </datalist>
  <input type="submit">
</form>
```



## b. <output>

Tag <output> dùng để diễn tả kết quả tính toán

```
<form action="" oninput="x.value=parseInt(a.value)+parseInt(b.value)" style="max-width:200px">
<input type="range" id="a" name="a" value="1" min="1" max="20" >
<input type="number" id="b" name="b" value="10"><br/>
<output name="x" for="a b"></output> <br/>
<input type="submit">
```

## 5. Các kiểu input mới trong HTML5

HTML5 thêm nhiều kiểu input mới trong form: color, date, datetime-local, email, month, number, range, search, tel, time, url, week

Tuy nhiên, vì nhiều lý do, các trình duyệt chưa hỗ trợ hết các loại này. Mời bạn test vài cái được support trong 3 trình duyệt nổi tiếng nhất (Chrome, Firefox, IE):

```
<form>
  Chọn màu: <input type="color" name="mau"> <br/>
  E-mail: <input type="email" name="email"> <br/>
  Tháng: <input type="number" name="thang" min="1" max="12"> <br/>
  Website: <input type="url" name="homepage">
  <input type="submit">
</form>
```

## 6. Các thuộc tính quy định giá trị nhập;

Sau đây là 1 số giá trị giúp bạn quy định cách nhập trong form của user:

Thuộc tính	Mô tả
disabled	Khóa tag input, không cho user nhập
max	Giá trị lớn nhất mà user có thể nhập trong tag input
maxlength	Số ký tự lớn nhất mà user có thể nhập trong tag input

min	Giá trị nhỏ nhất mà user có thể nhập trong tag input
pattern	Chỉ định biểu thức regular expression để test giá trị user nhập
readonly	Specifies that an input field is read only (cannot be changed)
required	Bắt buộc user phải nhập giá trị vào
size	Độ rộng của tag (tính theo ký tự)
step	Chỉ định bước nhảy của giá trị trong tag input
value	Giá trị mặc định trong tag input

Ví dụ:

```
<form>
  Số lượng:
  <input type="number" name="diem" min="0" max="100" step="10" value="30">
</form>
```

## II. CSS3

CSS3 chuẩn mới nhất để định dạng trang web với rất nhiều đặc điểm định dạng.

### 1. Rounded Corners

CSS3 có thuộc tính border-radius giúp bo tròn góc của các tag

```
<!DOCTYPE html><html><head>
<style>
#rcorners4 {
  border-radius: 15px 50px 30px 5px;
  background: #73AD21; padding: 20px; width: 200px; height: 150px;
}
</style>
</head>
<body><p id="rcorners4"></p></body></html>
```



## 2. Background Image

Bạn có thể chỉ định nhiều hình nền khi định dạng

```
<!DOCTYPE html><html> <head>
<style>
#box1{
    background-image: url(cha.jpg), url(nen.jpg);
    background-position: left center, right top;
    background-repeat: no-repeat, repeat; text-align:right;
    padding: 15px; width:700px; height:320px; font-size:1.5em; }
</style>
</head>
<body>
<div id="box1">
<h1>Cha tôi</h1>
Cha đưa cả tấm lưng gầy<br/> Chờ che con được tới ngày hôm nay.<br/>
Cha tôi lắm nỗi gian nan<br/> Vì con cơ cực chẳng màng tâm thân.<br/>
Cha ơi bóng cả cây cao<br/> Chờ che con những lao đao cuộc đời.<br/>
</div>
</body></html>
```



## 3. Gradients

Gradients giúp bạn tạo nên hiệu ứng màu nền giữa 2 màu (nhiều màu cũng được)

```
<!DOCTYPE html><html><head>
<style>
#box2 {
    height: 200px; background: linear-gradient(to right, red , yellow, green);
}
</style>
```

```
</style>
</head>
<body>
<div id="box2"></div>
</body></html>
```



#### 4. Shadow

Có 2 thuộc tính tạo bóng: bóng theo viền của khung (box-shadow) và bóng cho ký tự (text-shadow)

```
<!DOCTYPE html><html><head>
<style>
#box3 {
width: 600px; height: 100px; padding: 20px; font-size:2em;
background-color: yellow; text-align:center;
box-shadow: 10px 10px 5px grey;
text-shadow:3px 2px 1px #2196F3; }
</style>
</head>
<body>
<div id="box3">Mênh mông bát ngát đại dương<br/>Cũng không sánh được tình
thương mẹ hiền</div>
</body></html>
```

Mênh mông bát ngát đại dương  
Cũng không sánh được tình thương mẹ hiền

#### 5. Text

Ba thuộc tính mới định dạng text: text-overflow, word-wrap, word-break



Thuộc tính **text-overflow** chỉ định cách hiển thị text khi nội dung tràn vùng hiển thị

```
<!DOCTYPE html><html><head><style>
p.test1 {
  white-space:nowrap; width:380px; border:1px solid #000000; overflow:hidden;
  text-overflow: clip;
}
p.test2 {
  white-space:nowrap; width:380px; border:1px solid #000000; overflow:hidden;
  text-overflow: ellipsis;
}
</style></head>
<body>
<p class="test1">Mẹ ơi con hiểu một điều. Khi con mất mẹ như điều đứt dây</p>
<p class="test2">Mẹ ơi con hiểu một điều. Khi con mất mẹ như điều đứt dây</p>
</body></html>
```

Mẹ ơi con hiểu một điều. Khi con mất mẹ như điều đứt dây

Mẹ ơi con hiểu một điều. Khi con mất mẹ như điều đứt ...

## 6. Web Fonts

### a. Dùng @font-face

Web fonts cho phép bạn dùng các font mà máy tính của người dùng không có. Tức là bạn nhúng font từ server của bạn vào trang web để đưa về máy tính của user. Bạn nhúng font bằng chỉ chỉ **@font-face**

```
<!DOCTYPE html><html><head>
<style>
@font-face { font-family: fontcuaTui; src: url(DancingScript-Bold.ttf); }
p.test1 { width: 500px; font-size:2em; font-family:fontcuaTui; }
</style>
</head>
<body>
<p class="test1">Mẹ ơi con chịu bao điều<br/>Nhưng không chịu nổi thân điều
đứt dây.</p>
</body></html>
```

Mẹ ơi con chịu bao điều

Nhưng không chịu nổi thân điều đứt dây.



## b. Dùng @import

Dùng chỉ thị import, bạn cũng có thể nhúng font mà máy người dùng không có. Các font có thể lấy từ host khác chứ không hẳn lấy từ server của bạn.

```
<!DOCTYPE html><html><head>
<style>
@import url('https://fonts.googleapis.com/css?family=Pattaya');
.test1 {width:500px; font-size:2em; font-family:'Pattaya',sans-serif;}
</style></head>
<body>
<p class="test1">Đời người như một giấc mơ, <br/>
Trăm năm nào khác cuộc cờ đổi thay.
</p>
</body></html>
```

*Đời người như một giấc mơ,  
Trăm năm nào khác cuộc cờ đổi thay.*

## 7. 2D Transforms

Transforms là hiệu ứng dùng để di chuyển quay, phóng to thu nhỏ, làm nghiêng. Dùng transform bạn có thể làm thay đổi shape, size và position. CSS3 hỗ trợ 2 cách supports transform: 2D và 3D.

2D Transforms có các hàm translate(), rotate(), scale(), skewX(), skewY(), matrix()

```
<!DOCTYPE html><html><head>
<style>
div {width:300px;height:100px;background-color:yellow;border:1px solid black}
div#myDiv {transform: rotate(-20deg); }
</style></head>
<body>
<div> Thương anh nhiều vô kể<br/>Giữa chợ đời bán mua<br/>Có lúc được lúc
thua<br/>Anh mỉm cười: "Thôi kệ!</div>
<div id="myDiv">Thương anh nhiều vô kể<br/>Thôi kệ là hiền nhân<br/>Thôi kệ là
nghịệp lành<br/>Nổi tình thân huynh đệ</div>
</body>
</html>
```

Thương anh nhiều vô kể  
Giữa chợ đời bán mua  
Có lúc được lúc thua  
Anh mỉm cười: "Thôi kệ!"

Thương anh nhiều vô kể  
Thôi kệ là hiền nhân  
Thôi kệ là nghiệp lành  
Nổi tình thân huynh đệ

## 8. 3D Transforms

3D transform có 3 hàm :rotateX() ,rotateY() , rotateZ() giúp bạn quay nội dung đối tượng.

```
<!DOCTYPE html><html><head>
<style>
div {
width:300px;height:100px; background-color:yellow;border:1px solid black;
}
div#myDiv { transform: rotateY(180deg); }
</style></head>
<body>
<div>Đừng quên hi vọng, sự hi vọng cho bạn sức mạnh để tồn tại ngay khi bạn
đang bị bỏ rơi.</div>
<div id="myDiv">Đừng quên hi vọng, sự hi vọng cho bạn sức mạnh để tồn tại ngay
khi bạn đang bị bỏ rơi.</div>
</body></html>
```

Đừng quên hi vọng, sự hi vọng cho bạn sức  
mạnh để tồn tại ngay khi bạn đang bị bỏ rơi.

Đừng quên hi vọng, sự hi vọng cho bạn sức  
mạnh để tồn tại ngay khi bạn đang bị bỏ rơi.

## 9. Transitions

Transitions cho phép bạn thay đổi giá trị 1 thuộc tính rất mượt trong 1 khoảng thời gian.

```
<!DOCTYPE html><html><head>
<style>
#dn { width:250px; height:200px; background: red; font-size:1.5em; text-align:center; transition:width 2s, height 5s;}
#dn:hover { width:400px; height:120px}
</style></head>
<body>
<div id="dn"> Đừng lấy của cải vật chất để đo lường thành công hay thất bại, tâm hồn của mỗi con người mới xác định được mức độ "giàu có" trong cuộc sống của mình.</div>
</body></html>
```

Đừng lấy của cải vật chất để đo lường thành công hay thất bại, tâm hồn của mỗi con người mới xác định được mức độ “giàu có” trong cuộc sống của mình.

## 10. Animations

Animation giúp bạn tạo các hiệu ứng hoạt hình cho các tag mà không cần tới javascript hay Flash.

```
<!DOCTYPE html><html><head>
<style>
#box5 {
width: 200px; height: 100px; background-color: red; text-align:center;
padding:15px; font-size:1.2em;
animation-name:chuyenmau; animation-duration:4s; animation-iteration-count:3;
}
@keyframes chuyenmau {
from {background-color: red;}
to {background-color: yellow;}
}
</style></head>
<body>
<div id="box5"> Đôi mi nào khép lại<br/>Phong kín một bông hồng<br/>
Khum tay che ngọn gió<br/>Tay gõ vào cửa không </div>
```

</body></html>

Đôi mi nào khép lại  
Phong kín một bông hồng  
Khum tay che ngọn gió  
Tay gõ vào cửa không

[https://www.w3schools.com/css/css3\\_animations.asp](https://www.w3schools.com/css/css3_animations.asp)

## 11. Multi-column

CSS3 cho phép bạn trình bày chiều cột chữ rất dễ dàng. Các thuộc tính để định dạng nhiều cột gồm: multi-column properties: column-count , column-gap , column-rule-style , column-rule-width, column-rule-color, column-rule, column-span, column-width

```
<!DOCTYPE html><html><head>
<style>
.newspaper {
width:700px; background:#F44336;color:white;font-size:1.2em;line-height:150%;
column-count:3; column-gap:40px; column-rule:1px solid lightblue;
}
</style></head>
<body>
<div class="newspaper">
Đôi mi nào khép lại<br/>Phong kín một bông hồng<br/>
Khum tay che ngọn gió<br/>Tay gõ vào cửa không<br/>
Cách chia hàng giậu nhỏ<br/>Một nhịp cầu tơ sương<br/>
Trăng đã về bên ấy<br/>Đợi chi giữa vô thường<br/>
Am cỏ trắng vắng vặc<br/>Cây lá đứng lặng im<br/>
Áo mỏng mờ sương khói<br/>Giữa vòng tay vô biên
</div>
</body></html>
```

Đôi mi nào khép lại	Cách chia hàng giậu nhỏ	Am cỏ trắng vắng vặc
Phong kín một bông hồng	Một nhịp cầu tơ sương	Cây lá đứng lặng im
Khum tay che ngọn gió	Trăng đã về bên ấy	Áo mỏng mờ sương khói
Tay gõ vào cửa không	Đợi chi giữa vô thường	Giữa vòng tay vô biên

## 12. Resizing

CSS3 có thuộc tính resize cho phép user chỉnh kích thước đối tượng . resize

```
<!DOCTYPE html><html><head>
<style>
#tho {
border:2px solid; padding:10px;width:300px; resize:horizontal; overflow:auto;
}
</style></head>
<body> <div id="tho">
<p>Sao hôm tìm sao mai.
Ơi người đi bắt bóng
Vùng trăng treo trước ngực
Bóng đỏ dài sau lưng.</p>
<p>Đôi mi nào khép lại
Phong kín một bông hồng
Khum tay che ngọn gió
Tay gõ vào cửa không</p>
</div>
</body></html>
```

Sao hôm tìm sao mai. Ơi người đi bắt bóng  
Vùng trăng treo trước ngực Bóng đỏ dài sau  
lưng.  
Đôi mi nào khép lại Phong kín một bông hồng  
Khum tay che ngọn gió Tay gõ vào cửa không

### 13. Box Sizing

CSS3 có thuộc tính tên **box-sizing** rất quan trọng, nó giúp bạn bao gồm luôn padding và border vào trong vùng độ rộng và cao của đối tượng. Giúp bạn tránh rắc rối khi đối tượng chạy lung tung khi chỉnh padding, border.

```
<!DOCTYPE html><html><head>
<style>
.div1 {width: 300px; height: 100px; border: 1px solid blue;}
.div2 { width: 300px; height: 100px; border: 1px solid red; padding: 20px;}
.div3 { width: 300px; height: 100px; border: 1px solid green; padding: 20px;
      box-sizing:border-box}
</style></head>
<body>
<div class="div1">
Bạn ta bảy chục tuổi xuân<br/>Giờ ngồi xem kiến hành quân thay người<br/>
May còn con kiến để chơi<br/>Có khi vạn sự ở đời con con</div> <br/>
<div class="div2">
Dễ từng lấp biển dời non<br/>Lung còng mới được lon ton tuổi già<br/>
Góc vườn đàn kiến bò ra<br/>Kéo đi hàng dọc như là Trường Sơn.</div> <br/>
<div class="div3">
```

Nhớ ơi đồng đội chập chờn  
Chiến tranh bỏ lại cô đơn hòa bình  
Ngồi chơi với kiến giết mình  
Một ta mà cả đội hình ngày xưa.

Bạn ta bảy chục tuổi xuân  
Giờ ngồi xem kiến hành quân thay người  
May còn con kiến để chơi  
Có khi vạn sự ở đời con con

Dễ từng lấp biển đời non  
Lung còng mới được lon ton tuổi già  
Góc vườn đàn kiến bỏ ra  
Kéo đi hàng dọc như là Trường Sơn.

Nhớ ơi đồng đội chập chờn  
Chiến tranh bỏ lại cô đơn hòa bình  
Ngồi chơi với kiến giết mình  
Một ta mà cả đội hình ngày xưa.

## 14. Media Query

Media query trong CSS3 giúp bạn xác định kiểu của thiết bị mà người dùng đang sử dụng để xem trang web của bạn. Media query xác định kiểu thiết bị dựa vào các tiêu chí như: width và height của viewport, width và height của thiết bị, orientation (hướng dọc/ngang), độ phân giải.

Sử dụng media query là cách phổ biến để định dạng trang web cho phù hợp với thiết bị.

### a. Xác định media

Cú pháp media query trong trang html

```
@media not|only mediatype and (expressions) {  
    CSS-Code;  
}
```

Cú pháp nhúng file css

```
<link rel="stylesheet" media="mediatype and|not|only (expresion)"  
href="abc.css" >
```

## b. Các loại media types trong css3

Value	Description
all	Mọi loại thiết bị
print	Sử dụng để in ấn
screen	Dùng cho màn hình máy tính., tablets, smart-phones ...
speech	Sử dụng cho screenreaders để đọc trang

## c. Ví dụ

```
<!DOCTYPE html><html><head>
<style>
body { background-color: pink;}
@media screen and (min-width: 480px) {
    body { background-color: lightgreen; }
}
</style></head>
<body><h1>Vành nón nghiêng che</h1>
<p>Vành nón đàn bà<br/> thường rộng hơn vành mũ đàn ông<br/>
Phải chăng phái nữ cần nhiều hơn bóng mát?<br/>
Thương mẹ quá, nắng sương pha tóc bạc<br/>
Bóng mát đời mình, mẹ nhường hết cho con<br/> <br/>
Vành nón đàn bà<br/> thường rộng hơn vành mũ đàn ông<br/>
Phải chăng phái nữ cần nhiều hơn bóng mát?<br/>
In dáng mẹ, em mưa nắng đi, về<br/>Thường quen tay nghiêng nón<br/>
Che cho con cho chồng thêm bóng yêu thương</p>
</body></html>
```

## III. Reponsevice

### 1. Giới thiệu

Thiết kế web responsive là làm cho trang web nhìn tốt trên các loại thiết bị khác nhau (desktops, tablets, phones).

Để làm được điều này, bạn sử dụng HTML và CSS để chỉnh lại kích thước, ẩn bớt, phóng to, thu nhỏ, hoặc di chuyển nội dung các thành phần trong trang để làm cho trang web xem tốt được trên nhiều kích thước màn hình.



## 2. Khai báo Viewport

Viewport là vùng user nhìn thấy của trang web. Viewport khác nhau tùy theo loại thiết bị, nó nhỏ với điện thoại và là lớn với màn hình máy tính.

Khi tạo chức năng responsive cho trang, bạn phải thêm lệnh khai báo viewport trong tag head:

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

Lệnh trên dùng để chỉ cho trình duyệt cách thức để điều khiển độ rộng của trang cũng như điều chỉnh các thành phần trong trang cho phù hợp ngữ cảnh màn hình. Trong đó:

**width=device-width** ấn định độ rộng của trang theo độ rộng của màn hình thiết bị.

**initial-scale=1.0** ấn định tỷ lệ phóng (zoom) ban đầu là 1 khi trang được nạp.

Sau đây là ví dụ không có và có tag meta viewport



### Không khai báo tag meta viewport



### Có khai báo meta viewport



Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat. Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio

## 3. Các nguyên tắc khi thực hiện responsive

### a. Responsive Images

- **Sử dụng thuộc tính max-width :** Responsive images là làm cho các hình có tỷ lệ đẹp theo kích cỡ của browser. Nếu ấn định thuộc tính **max-width** là 100% thì các hình sẽ nhỏ lại nếu cần thiết nhưng chúng sẽ không lớn hơn kích thước gốc. Ví dụ:

```

```

- **Hiện các hình khác nhau tùy theo độ trong của browser :** Dùng tag <picture> giúp bạn chỉ định các hình khác nhau theo kích thước trình duyệt.

```
<picture>
  <source srcset="a.jpg" media="(max-width: 480px)">
  <source srcset="b.jpg" media="(max-width: 600px)">
  <source srcset="c.jpg">
  <img src="">
</picture>
```

## b. Responsive Text Size

Cỡ chữ có thể được chỉ định bằng đơn vị "vw", nghĩa là "viewport width". Ấn định này làm cho cỡ chữ thay đổi theo cỡ của browser:

```
<p style="font-size:5vw">Hạnh phúc là khi ta tìm ra và trân trọng những niềm vui, những giá trị giản dị từ cuộc sống</p>
```

## c. Media query

Media query là cách thức trong CSS3 giúp bạn dò môi trường hiện tại để biết mà định dạng trang web. Bằng cách sử dụng chỉ thị **@media**

```
<!DOCTYPE html><html><head>
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<style>
body {background-color: lightgreen;}
@media only screen and (max-width: 500px) {
    body {background-color: lightblue; }
}
</style>
</head>
<body>
<p>Đừng nên thờ ơ với những gì đã quá quen thuộc với bạn.Hãy giữ chắc lấy chúng như những gì quan trọng nhất,vì sẽ có lúc bạn cảm thấy tiếc nuối khi những điều thân thuộc ấy mất đi.</p>
<p>Đừng hạ thấp giá trị của mình bằng cách so sánh bản thân mình với người khác.Mỗi chúng ta là một con người khác nhau và đều có những giá trị khác nhau.</p>
</body></html>
```

Bằng cách dùng media query, bạn có thể định nghĩa cách hiển thị trang web của mình trên các môi trường khác nhau.

Ví dụ 2:

```
<!DOCTYPE html> <html>
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<style>
* { box-sizing: border-box;}
.left { background-color:#2196F3; padding:20px; float:left; width:20%; }
.main { background-color:#f1f1f1; padding:20px; float:left; width:60%; }
.right{ background-color:#4CAF50; padding:20px; float:left; width:20%; }

@media (max-width:800px) {
    .left, .main, .right { width:100%; }
}
```

```
}  
</style>  
<body>  
<div class="left"> Left Menu<br><br><br><br> </div>  
<div class="main"> Main Content<br><br><br><br> </div>  
<div class="right"> Right Content<br><br><br><br> </div>  
</body> </html>
```

#### 4. Các nền tảng hỗ trợ Responsive Web Design

Việc thiết kế trang web có responsive là rất hay, rất cần thiết nhưng không hề dễ dàng, rất cực nếu như bạn tự thực hiện.

Bạn nên sử dụng sự hỗ trợ của các bộ thư viện giúp cho việc này thì hay hơn. Có nhiều bộ thư viện như W3.CSS, Bootstrap... sẽ giúp bạn thực hiện này nhanh và hiệu quả hơn.

## J. BOOTSTRAP 4

### I. Sơ lược về Bootstrap

#### 1. Giới thiệu

Bootstrap là một nền tảng cho phép phát triển các website hỗ trợ responsive. Nó định nghĩa sẵn rất nhiều class CSS, icon font, các hàm tiện ích chạy trên jquery... để tạo ra các kiểu chữ, button, menu và các thành phần giao diện trong trang.

Bootstrap hỗ trợ Responsive với nhiều kích thước màn hình khác nhau

Ký hiệu lớp	Thiết bị	Độ rộng lớp container	Chú thích
.col-xs-\$	Extra small	Auto	Dùng cho điện thoại kích thước nhỏ hơn 768px
.col-sm-\$	Small devices	750px	Dùng cho tablets kích thước $\geq 768$ px
.col-md-\$	Medium devices	970px	Dùng cho desktop ( $\geq 992$ px)
.col-lg-\$	Large devices	1170px	Dùng cho desktops $\geq 1200$ px

Trong đó \$ là một số nằm trong khoảng từ 1-12

#### 2. Các ưu điểm của Bootstrap

1. Chuẩn hiển thị trên mọi màn hình, mọi trình duyệt.
2. Tự động điều chỉnh kích thước trang theo kích thước trình duyệt.
3. Tốc độ load tốt.
4. Hỗ trợ nhiều kiểu hiển thị sẵn, nhiều kiểu box, nhiều kiểu form đẹp và sang trọng.
5. Hỗ trợ LESS và SASS giúp lập trình viên nhanh chóng tích hợp Bootstrap.
6. Việc tích hợp Bootstrap vào website rất dễ dàng.

## II. Download và nhúng bootstrap vào trang

### 1. Downloaf bootstrap

- Vào <https://getbootstrap.com/docs/4.2/getting-started/download/> rồi nhấp **Download**
- Chép file zip vừa down vào website của bạn rồi giải nén (Extract Here) , bạn sẽ được 1 folder
- Vào folder vừa chép, sẽ thấy các folder css, js. Đó là những gì chúng ta cần

### 2. Nhúng bootstap vào trang web

Cách 1: Mở 1 trang trong website của bạn rồi nhúng code css và js của botstrap vào:

```
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js"> </script>
<link rel="stylesheet" type="text/css" href="bootstrap-4.2.1-dist/css/bootstrap.min.css">
<script src="bootstrap-4.2.1-dist/js/bootstrap.min.js"></script>
```

Do bootstrap chạy trên nền jquery nên bạn phải kèm thêm jquery như trên. Nhúng jquery trực tiếp từ CDN như ví dụ trên hoặc download jquery về website rồi nhúng vào cũng được.

Cách 2: Nhúng bootstrap trực tiếp từ CDN, không cần down về máy, đoạn code có sẵn trên trang download bootstrap

```
<!-- Latest compiled and minified CSS -->
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css"
integrity="sha384-BVYiiSIFeK1dGmJRAkycuHAHRg32OmUcww7on3RYdg4Va+PmSTsz/K68vbdEjh4u" crossorigin="anonymous">

<!-- Optional theme -->
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap-theme.min.css"
integrity="sha384-rHyoN1iRsVXV4nD0JutlnGaslCJuC7uwjduW9SVrLvRYooPp2bWYgmgJQIXwl/Sp" crossorigin="anonymous">

<!-- Latest compiled and minified JavaScript -->
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js" integrity="sha384-Tc5IQib027qvyjSMfHjOMaLkfuWxZxUPnCJA7l2mCWNIpG9mGCD8wGNICPD7Txa" crossorigin="anonymous"></script>
```

### 3. Thiết lập cho website hỗ trợ giao diện Mobile

Để chắc chắn giao diện của website có hỗ trợ mobile. Đặt code sau trong tag <head>:

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

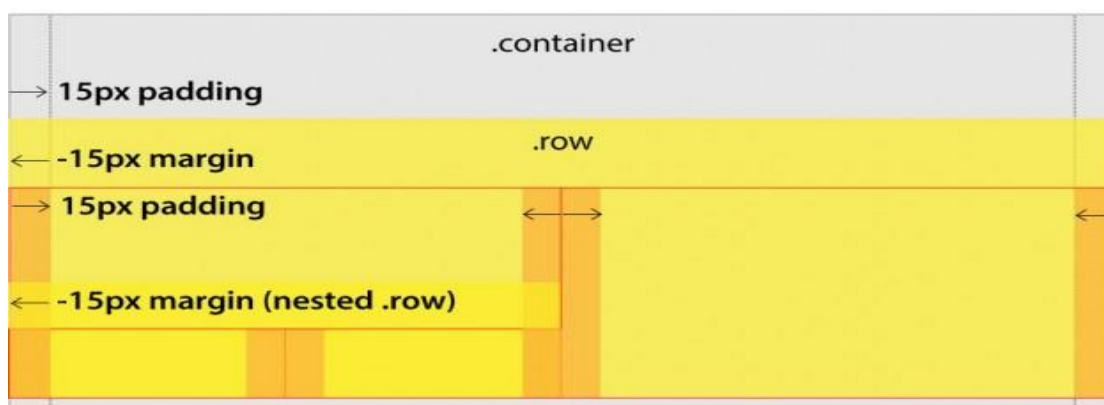
Tag này dùng để co giãn độ rộng trang theo độ rộng của thiết bị. 1,0 là thiết lập mức độ phóng ban đầu khi trang được trình duyệt tải lần đầu tiên.

#### 4. Một số lưu ý khi sử dụng Bootstrap

class **Container**: được fix sẵn độ rộng tùy theo độ phân giải, padding sang hai bên là 15px

class **Container-fluid**: có độ rộng full màn hình

class **Row**: Margin là số âm: -15px nếu muốn thẻ nào đó sát bên lề hai bên, trái ngược với Container, bạn chỉ cần đặt container bọc ngoài thẻ Row là thẻ trong đó sẽ nằm đúng lề,



### III. Sử dụng các thành phần trong bootstrap

#### 1. Chuẩn bị

Để học/thực tập các thành phần trong bootstrap, bạn cần chuẩn bị như sau:

1. Tạo 1 trang html/php và lưu với tên gì đó, chẳng hạn **testBT.html**
2. Chèn code jquery, code bootstrap vào trang của mình. Dưới đây là ví dụ:

```
<!doctype html> <html><head>
<meta charset="utf-8">
<script src=
"https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js"> </script>
<link rel="stylesheet" type="text/css" href="bootstrap-4.2.1-
dist/css/bootstrap.min.css">
```

```
<script src="bootstrap-4.2.1-dist/js/bootstrap.min.js"></script>
</head><body>
<div class="container">

</div>
</body> </html>
```

Code màu đỏ ở trên bạn có thể sửa cho thích hợp với site của bạn. Và từ đây trở xuống , các đoạn code exampe bạn hãy copy đặt trong body để xem thử

## 2. Color

### a. Text color

Các class dùng để định dạng màu chữ gồm **.text-muted** , **.text-primary** , **.text-success** , **.text-info** , **.text-warning** , **.text-danger** , **.text-secondary** , **.text-white** , **.text-dark** , **.text-body** , **.text-light**

```
<p class="text-muted">This text is muted.</p>
<p class="text-primary">This text is important.</p>
<p class="text-success">This text indicates success.</p>
<p class="text-info">This text represents some information.</p>
<p class="text-warning">This text represents a warning.</p>
<p class="text-danger">This text represents danger.</p>
<p class="text-secondary">Secondary text.</p>
<p class="text-dark">This text is dark grey.</p>
<p class="text-body">Default body color (often black).</p>
<p class="text-light">This text is light grey (on white background).</p>
<p class="text-white">This text is white (on white background).</p>
```

Có thể thêm 50% opacity cho màu chữ đen/trắng khi dùng thêm class **.text-black-50** , **.text-white-50**

```
<p class="text-black-50">Black text with 50% opacity on white background</p>
<p class="text-white-50 bg-dark">White text with 50% opacity on black bg</p>
```

### b. Background color

Các class màu nền gồm: **.bg-primary** , **.bg-success** , **.bg-info** , **.bg-warning** , **.bg-danger** , **.bg-secondary** , **.bg-dark** , **.bg-light**

```
<p class="bg-primary text-white">This text is important.</p>
<p class="bg-success text-white">This text indicates success.</p>
```



```
<p class="bg-info text-white">This text represents some information.</p>
<p class="bg-warning text-white">This text represents a warning.</p>
<p class="bg-danger text-white">This text represents danger.</p>
<p class="bg-secondary text-white">Secondary background color.</p>
<p class="bg-dark text-white">Dark grey background color.</p>
<p class="bg-light text-dark">Light grey background color.</p>
```

### 3. Form

Khi tạo form có dùng các định dạng của bootstrap, bạn cần nhớ:

- Dùng `<div class="form-group">` bao quanh mỗi form control và nhãn của nó
- Thêm class `.form-control` vào các tag `<input>`, `<textarea>` và `<select>`
- Tag form có 1 class đặc biệt do bạn chỉ định sẽ quyết định cách hiện của form

#### a. Form inline

Địa chỉ Email:  Mật khẩu:  ☐ Ghi nhớ

```
<form class="form-inline">
  <div class="form-group">
    <label for="email">Địa chỉ Email:</label>
    <input type="email" class="form-control" id="email">
  </div>
  <div class="form-group">
    <label for="pwd">Mật khẩu:</label>
    <input type="password" class="form-control" id="pwd">
  </div>
  <div class="checkbox">
    <label><input type="checkbox"> Ghi nhớ</label>
  </div>
  <button type="submit" class="btn btn-default">Đăng nhập</button>
</form>
```

#### b. Form horizontal

Địa chỉ Email:

Mật khẩu:

☐ Ghi nhớ



```
<form class="form-horizontal">
  <div class="form-group">
    <label for="email">Địa chỉ Email:</label>
    <input type="email" class="form-control" id="email">
  </div>
  <div class="form-group">
    <label for="pwd">Mật khẩu:</label>
    <input type="password" class="form-control" id="pwd">
  </div>
  <div class="checkbox">
    <label><input type="checkbox"> Ghi nhớ</label>
  </div>
  <button type="submit" class="btn btn-default ">Đăng nhập</button>
</form>
```

## 4. Navigation

### a. Navigation 1 cấp

Thanh navigation hay menu được tạo rất dễ dàng với <nav class="navbar navbar-default">

```
<nav class="navbar navbar-expand-sm navbar-light bg-info">
  <ul class="navbar-nav">
    <a class="navbar-brand" href="#">Sống đẹp</a>
    <li class="nav-item">
      <a class="nav-link" href="#">Bài học</a>
    </li>
    <li class="nav-item">
      <a class="nav-link" href="#">Suy ngẫm</a>
    </li>
    <li class="nav-item">
      <a class="nav-link" href="#">Yêu thương</a>
    </li>
  </ul>
</nav>
```

Sống đẹp   Bài học   Suy ngẫm   Yêu thương

- Class **navbar-expand-xl|lg|md|sm** để chỉ định sẽ hiện dọc trên loại thiết bị.
- Class **bg-info** là màu nền, có thể đổi thành **bg-primary** , **bg-success** , **bg-info** , **bg-warning** , **bg-danger** , **bg-secondary** , **bg-dark** , **bg-light**
- Class **navbar-light** là màu chữ, có thể đổi thành **navbar-dark**

- Nếu thêm class `justify-content-center` vào tag nav thì thanh sẽ canh giữa
- Class `navbar-brand` là logo, có thể bỏ

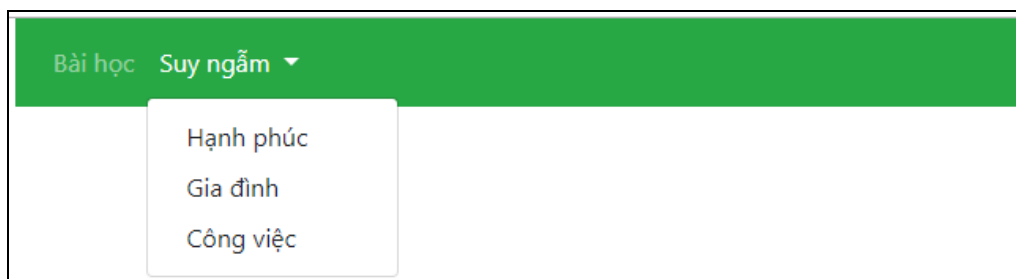
### b. Navigation collapse

Là thanh sẽ tự động co lại khi màn hình nhỏ

```
<nav class="navbar navbar-expand-md bg-dark navbar-dark">
  <a class="navbar-brand" href="#">Chuyên mục</a>
  <button class="navbar-toggler" type="button" data-toggle="collapse" data-
target="#collapsibleNavbar">
    <span class="navbar-toggler-icon"></span>
  </button>
  <div class="collapse navbar-collapse" id="collapsibleNavbar">
    <ul class="navbar-nav">
      <li class="nav-item">
        <a class="nav-link" href="#">Tin tức</a>
      </li>
      <li class="nav-item">
        <a class="nav-link" href="#">Sản phẩm</a>
      </li>
      <li class="nav-item">
        <a class="nav-link" href="#">Giới thiệu</a>
      </li>
    </ul>
  </div>
</nav>
```

### c. Navigation có dropdown

```
<nav class="navbar navbar-expand-sm bg-success navbar-dark">
  <ul class="navbar-nav">
    <li class="nav-item">
      <a class="nav-link" href="#">Bài học</a>
    </li>
    <li class="nav-item dropdown">
      <a class="nav-link dropdown-toggle" href="#" id="navbardrop" data-
toggle="dropdown">
        Suy ngẫm
      </a>
      <div class="dropdown-menu">
        <a class="dropdown-item" href="#">Hạnh phúc</a>
        <a class="dropdown-item" href="#">Gia đình</a>
        <a class="dropdown-item" href="#">Công việc</a>
      </div>
    </li>
  </ul>
</nav>
```



#### d. Fixed navigation

```
<h1>Thơ hay</h1>
<nav class="navbar navbar-expand-sm bg-dark navbar-dark fixed-top">
  <a class="navbar-brand" href="#">Logo</a>
  <ul class="navbar-nav">
    <li class="nav-item">
      <a class="nav-link" href="#">Link</a>
    </li>
    <li class="nav-item">
      <a class="nav-link" href="#">Link</a>
    </li>
  </ul>
</nav>
<p>Bạn<p>ta<p>bảy<p>chục<p>tuổi<p>xuân<br/>Giờ<p>ngồi<p>xem<p>kiến<p>hành<p>qu
ân<p>thay<p>người<p>
May<p>còn<p>con<p>kiến<p>để<p>chơi<br/>Có<p>khi<p>vạn<p>sự<p>ở<p>đời<p>con<p>c
on<p>
<p>Để<p>tùng<p>lấp<p>biển<p>dời<p>non<br/>Lung<p>còng<p>mới<p>được<p>lon<p>ton
<p>tuổi<p>già<p>
Góc<p>vuòn<p>đàn<p>kiến<p>bò<p>ra<br/>Kéo<p>đi<p>hàng<p>dọc<p>nhu<p>là<p>Trườn
g<p>Sơn.
```

- **fixed-top** : canh theo cạnh trên
- **fixed-bottom** : canh theo cạnh dưới
- **sticky-top** : Khi user cuộn, thanh navigation dính cạnh trên thì dừng lại

## 5. Badges

- Badges dùng để thông tin về 1 trạng thái, 1 điều gì đó.

```
<h4><span class="badge badge-default">Default Badge</span>
<span class="badge badge-primary">Tin nổi bật</span>
<span class="badge badge-success">Tin đang hiện</span>
<span class="badge badge-info">Còn hàng</span>
<span class="badge badge-warning">Sản phẩm bị ẩn</span>
<span class="badge badge-danger">Hết hàng</span>
</h4>
```

Default Badge Tin nổi bật Tin đang hiện Còn hàng Sản phẩm bị ẩn Hết hàng

- Nếu muốn bo tròn góc thì thêm class **badge-pill**
- Có thể dùng badge trong các element khác

```
<button type="button" class="btn btn-primary">
  Ý kiến <span class="badge badge-light">14</span>
</button>
<button type="button" class="btn btn-danger">
  Mục con <span class="badge badge-light">7</span>
</button>
```

## 6. Table

Các class trong bootstrap để định dạng table

.table	Là class cơ bản để định dạng 1 table trong bootstrap.
.table-striped	định dạng màu nền các hàng chẵn lẻ khác nhau
.table-bordered	Bổ sung border cho các ô
.table-hover	Định dạng để đưa chuột vào ô sẽ có màu thay đổi
.table-responsive	Tạo table responsive
.table-dark	Nền đen cho table
.table-borderless	Table không có viền
table-sm	Cho padding của các ô giảm 1 nữa
.table-primary .table-success .table-danger .table-info .table-warning .table-active .table-secondary .table-light .table-dark	Màu nền cho table, tr, td
thead-dark thead-light	Màu nền trong tag thead

### a. Bootstrap có class **.table** để định dạng table cơ bản

```
<table class="table">
  <thead>
    <tr> <th>Họ tên</th> <th>Email</th> </tr>
  </thead>
  <tbody>
    <tr> <td>Nguyễn Văn Tèo</td> <td>teo@giahu.com</td> </tr>
    <tr> <td>Phan Thị Luøm</td> <td>luom@gomeo.com</td> </tr>
    <tr> <td>Văn Thị Út Vẹo</td> <td>utveo@yahoo.com</td> </tr>
  </tbody>
</table>
```

Họ tên	Email
Nguyễn Văn Tèo	teo@giahu.com
Phan Thị Luøm	luom@gomeo.com
Văn Thị Út Vẹo	utveo@yahoo.com

### b. Bootstrap có class **.table-striped** để định dạng table dạng stripped

```
<table class="table table-striped">
  <thead>
    <tr> <th>Họ tên</th> <th>Email</th> </tr>
  </thead>
  <tbody>
    <tr> <td>Nguyễn Văn Tèo</td> <td>teo@giahu.com</td> </tr>
    <tr> <td>Phan Thị Luøm</td> <td>luom@gomeo.com</td> </tr>
    <tr> <td>Văn Thị Út Vẹo</td> <td>utveo@yahoo.com</td> </tr>
  </tbody>
</table>
```

Họ tên	Email
Nguyễn Văn Tèo	teo@giahu.com
Phan Thị Luøm	luom@gomeo.com
Văn Thị Út Vẹo	utveo@yahoo.com

### c. Bootstrap có class **.table-bordered** để định dạng các ô có đường viền

```
<table class="table table-bordered">
  <thead> <tr> <th>Họ tên</th> <th>Email</th> </tr> </thead>
  <tbody>
    <tr> <td>Nguyễn Văn Tèo</td> <td>teo@giahu.com</td> </tr>
    <tr> <td>Phan Thị Luøm</td> <td>luom@gomeo.com</td> </tr>
    <tr> <td>Văn Thị Út Vẹo</td> <td>utveo@yahoo.com</td> </tr>
  </tbody>
```

</table>

Họ tên	Email
Nguyễn Văn Tèo	teo@giahu.com
Phan Thị Lụa	luom@gomeo.com
Văn Thị Út Vẹo	utveo@yahoo.com

Xem thêm: [www.w3schools.com/bootstrap4/bootstrap\\_tables.asp](http://www.w3schools.com/bootstrap4/bootstrap_tables.asp)

## 7. Button

Sau đây là 8 class để tạo các button với bootstrap: btn , btn-default , btn-primary , btn-success, btn-info , btn-warning , btn-danger , btn-link

```
<button type="button" class="btn">Basic</button>
<button type="button" class="btn btn-default">Default</button>
<button type="button" class="btn btn-primary">Primary</button>
<button type="button" class="btn btn-success">Success</button>
<button type="button" class="btn btn-info">Info</button>
<button type="button" class="btn btn-warning">Warning</button>
<button type="button" class="btn btn-danger">Danger</button>
<button type="button" class="btn btn-link">Link</button>
```



## 8. Pagination

Bootstrap có class **.pagination** (gán cho tag ul ) để định dạng thanh phân trang. Và vài class khác

.pagination	Là class cơ bản để định dạng 1 thanh phân trang.
.active	Gán cho tag li diễn tả trang hiện hành
.disabled	Gán cho tag li diễn tả cấm click
.pagination-lg	Cho thanh phân trang cỡ lớn
.pagination-sm	Cho thanh phân trang cỡ nhỏ

```
<ul class="pagination">
  <li class="page-item"><a class="page-link" href="#">Previous</a></li>
```

```
<li class="page-item"><a class="page-link" href="#">1</a></li>
<li class="page-item"><a class="page-link" href="#">2</a></li>
<li class="page-item"><a class="page-link" href="#">3</a></li>
<li class="page-item"><a class="page-link" href="#">Next</a></li>
</ul>
```

Previous 1 2 3 Next

## 9. Card

```
<div class="card">
  <div class="card-header">Cung oán ngâm khúc</div>
  <div class="card-body">
    Chìm đáy nước cá lù dừ lặn, <br/>
    Lừng lừng trời nhận ngấn ngơ sa, <br/>
    Hương trời đắm nguyệt say hoa, <br/>
    Tây Thi mất vía, Hằng Nga giệt mình<br/>
  </div>
  <div class="card-footer">
    Trích đoạn bài thơ của rất nổi tiếng của Nguyễn Gia Thiều (1741-1798)
  </div>
</div>
```

Cung oán ngâm khúc

Chìm đáy nước cá lù dừ lặn,  
Lừng lừng trời nhận ngấn ngơ sa,  
Hương trời đắm nguyệt say hoa,  
Tây Thi mất vía, Hằng Nga giệt mình

Trích đoạn bài thơ của rất nổi tiếng của Nguyễn Gia Thiều (1741-1798)

- Nếu không cần, có thể bỏ heading, footer của panel
- Có thể thêm các class sau để chỉ định màu nền: **.bg-primary**, **.bg-success**, **.bg-info**, **.bg-warning**, **.bg-danger**, **.bg-secondary**, **.bg-dark**, **.bg-light**
- Class **.card-columns** dùng để tạo 1 grid cards. layout sẽ tự động điều chỉnh khi có nhiều card. Grid sẽ hiển thị dạng dọc trên màn hình small ( $\leq 576px$ ):
- Dùng class **.card-deck** để các card = nhau width và height, có một khoảng cách giữa các card
- Dùng class **.card-group** để các card = nhau width và height, không có 1 khoảng cách giữa các card

```
<h2>Cards Columns</h2>
```

```
<div class="card-columns">
  <div class="card bg-primary">
    <div class="card-body text-center">
      <p class="card-text">Some text inside the first card</p>
    </div>
  </div>
  <div class="card bg-warning">
    <div class="card-body text-center">
      <p class="card-text">Some text inside the second card</p>
    </div>
  </div>
  <div class="card bg-success">
    <div class="card-body text-center">
      <p class="card-text">Some text inside the third card</p>
    </div>
  </div>
  <div class="card bg-danger">
    <div class="card-body text-center">
      <p class="card-text">Some text inside the fourth card</p>
    </div>
  </div>
  <div class="card bg-light">
    <div class="card-body text-center">
      <p class="card-text">Some text inside the fifth card</p>
    </div>
  </div>
  <div class="card bg-info">
    <div class="card-body text-center">
      <p class="card-text">Some text inside the sixth card</p>
    </div>
  </div>
</div>
```

## 10. Image

```
 <!--img bình thường-->


 <hr/>
 <!--height=auto, width=100% -->
```



- Class **img-fluid** : responsive, tức max-width=100% , height=auto



- Class **mx-auto d-block** : để cho hình canh giữa (1 mình)
- Class **float-left/ float-right**: cho hình dạt qua trái/phải

## 11. Alert

Alert là hiện ra 1 thông báo cho user xem. Thông báo có thể ở những mức độ thông tin khác nhau như success, warning, danger... Các class có thể dùng : **alert-success** , **alert-info**, **alert-warning** , **alert-danger** , **alert-primary** , **alert-secondary** , **alert-light** , **alert-dark**

```
<div class="alert alert-success" >
    <strong>Lưu đơn hàng thành công!</strong> Cảm ơn bạn đã mua hàng.
</div>
<div class="alert alert-info">
    <strong>Thông tin!</strong> Đơn hàng của bạn có số tiền quá nhỏ.
</div>
<div class="alert alert-warning">
    <b>Báo động!</b> Việc chỉnh quyền, phải thoát ra vào lại mới có tác dụng.
</div>
<div class="alert alert-danger">
    <strong>Nguy hiểm!</strong> Nếu xóa loại tin, các tin trong đó sẽ mất hết.
</div>
```

Lưu đơn hàng thành công! Cảm ơn bạn đã mua hàng.

Thông tin! Đơn hàng của bạn có số tiền quá nhỏ.

Báo động! Việc chỉnh quyền, phải thoát ra vào lại mới có tác dụng.

Nguy hiểm! Nếu xóa loại tin, các tin trong đó sẽ mất hết.

## 12. Carousel

```
<div id="demo" class="carousel slide" data-ride="carousel">
    <!-- Indicators -->
    <ul class="carousel-indicators">
        <li data-target="#demo" data-slide-to="0" class="active"></li>
        <li data-target="#demo" data-slide-to="1"></li>
        <li data-target="#demo" data-slide-to="2"></li>
    </ul>
    <!-- The slideshow -->
    <div class="carousel-inner">
```

```
<div class="carousel-item active">
  
</div>
<div class="carousel-item">
  
  <div class="carousel-caption">
    <h3>Vịnh Hạ Long</h3>
    <p>Kỳ quan thế giới ai cũng nên viếng thăm!</p>
  </div>
</div>
<div class="carousel-item">
  
</div>
<!-- Left and right controls -->
<a class="carousel-control-prev" href="#demo" data-slide="prev">
  <span class="carousel-control-prev-icon"></span>
</a>
<a class="carousel-control-next" href="#demo" data-slide="next">
  <span class="carousel-control-next-icon"></span>
</a>
</div>
```

Các class

Class	Mô tả
.carousel	Tạo 1 carousel
.carousel-indicators	Hiện các nút chuyển tới 1 slide, giống thanh phân trang
.carousel-inner	Chứa các slides
.carousel-item	Chứa 1 slide
.carousel-control-prev	Nút chuyển qua slide trái khi user nhấp
.carousel-control-next	Nút chuyển qua slide phải khi user nhấp
.carousel-control-prev-icon	Icon qua trái
.carousel-control-next-icon	Icon qua phải
.slide	Hiệu ứng transition và animation khi chuyển slide

. carousel-caption

Nhấn cho từng slide

### 13. Modal

Hiện 1 cửa sổ popup để hiện thông tin, nằm trên trang web.

```
<button type="button" class="btn btn-primary" data-toggle="modal" data-
target="#myModal">Open modal</button>

<div class="modal" id="myModal">
<div class="modal-dialog">
<div class="modal-content">
  <!-- Modal Header -->
  <div class="modal-header">
    <h4 class="modal-title">Modal Heading</h4>
    <button type="button" class="close" data-dismiss="modal">&times;</button>
  </div>
  <!-- Modal body -->
  <div class="modal-body">
    Modal body..
  </div>
  <!-- Modal footer -->
  <div class="modal-footer">
    <button type="button" class="btn btn-danger" data-dismiss="modal">Close
  </button>
  </div>
</div>
</div>
</div>
```

- Thêm hiệu ứng fade : Thêm class fade cho div modal

```
<!-- Fading modal -->
<div class="modal fade"></div>
```

- Chỉ định size: Thêm class **modal-sm** hoặc **modal-lg** cho div **modal-dialog**

```
<div class="modal-dialog modal-sm">
```

- Modal giữa màn hình: Thêm class **modal-dialog-centered** cho div **modal-dialog**

```
<div class="modal-dialog modal-dialog-centered">
```

### 14. Collapse

Collapse rất hữu dụng khi bạn muốn ẩn hiện 1 lượng lớn nội dung.

```
<button data-toggle="collapse" class="btn btn-primary" data-target=
"#demo" >
Collapsible
</button>
<div id="demo" class="collapse">
Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod
tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim
veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea
commodo consequat.
</div>
```

## 15. Jumbotron

Jumbotron là 1 grey box để hiện thông tin quan trọng nào đó

```
<div class="jumbotron">
  <h1>Bootstrap Tutorial</h1>
  <p>Bootstrap is the most popular HTML, CSS...</p>
</div>
```

Nếu muốn full width thì thêm class `.jumbotron-fluid`

## 16. Text/Typography

- `<mark>`: Tag `<mark>` được định dạng màu nền vàng và padding

```
<p>Use the mark element to <mark>highlight</mark> text.</p>
```

- `<abbr>`: Tag `<abbr>` được định dạng border-bottom dạng dotted

```
<p>The <abbr title="World Health Organization">WHO</abbr> was founded
in 1948.</p>
```

- `<blockquote>`: Tag `<blockquote>` với class `blockquote` được dùng khi trích nguyên văn

```
<blockquote class="blockquote">
<p>For 50 years, WWF has been protecting the future of nature. The
world's leading conservation organization, WWF works in 100 countries
and is supported by 1.2 million members in the United States and close
to 5 million globally.</p>
<footer class="blockquote-footer">From WWF's website</footer>
</blockquote>
```

- Định dạng chữ

Class	Description
-------	-------------

<code>.font-weight-bold</code>	Chữ đậm
<code>.font-italic</code>	Chữ nghiêng
<code>.font-weight-light</code>	Chữ đậm sáng
<code>.font-weight-normal</code>	Chữ bình thường
<code>.lead</code>	Làm paragraph chữ lớn là sáng ra 1 chút
<code>.small</code>	Chữ nhỏ hơn, (khoảng 85% so với cha)
<code>.text-left</code>	Chữ canh trái
<code>.text-*-left</code>	Chữ canh trái trên màn hình small, medium, large , xlarge
<code>.text-center</code>	Chữ canh giữa
<code>.text-*-center</code>	Chữ canh giữa trên màn hình small, medium, large , xlarge
<code>.text-right</code>	Chữ canh phải
<code>.text-*-right</code>	Chữ canh phải trên màn hình small, medium, large , xlarge
<code>.text-justify</code>	Chữ canh đều hai biên
<code>.text-monospace</code>	CHỮ Monospaced
<code>.text-nowrap</code>	Không cho chữ cuộn xuống khi đựng lè phải
<code>.text-lowercase</code>	Hiện ra dạng chữ thường
<code>.text-uppercase</code>	Hiện ra dạng chữ hoa
<code>.text-capitalize</code>	Hiện ra dạng chữ hoa đầu mỗi từ

## 17. Flex

Các class flex trong BS4 dùng để tạo layout theo kỹ thuật flex – kỹ thuật làm layout tốt nhất hiện nay

- Tạo flex với class d-flex

```
<div class="d-flex p-3 bg-secondary text-white">
  <div class="p-2 bg-info">Flex item 1</div>
  <div class="p-2 bg-warning">Flex item 2</div>
  <div class="p-2 bg-primary">Flex item 3</div>
</div>
```

- Tạo flex với class d-inline-flex

```
<div class="d-inline-flex p-3 bg-secondary text-white">
  <div class="p-2 bg-info">Flex item 1</div>
  <div class="p-2 bg-warning">Flex item 2</div>
  <div class="p-2 bg-primary">Flex item 3</div>
</div>
```

- Flex horizontal: Dùng class `.flex-row` (mặc định) hoặc `.flex-row-reverse`

```
<div class="d-flex flex-row bg-secondary">
  <div class="p-2 bg-info">Flex item 1</div>
  <div class="p-2 bg-warning">Flex item 2</div>
  <div class="p-2 bg-primary">Flex item 3</div>
</div>
<hr>
<div class="d-flex flex-row-reverse bg-secondary">
  <div class="p-2 bg-info">Flex item 1</div>
  <div class="p-2 bg-warning">Flex item 2</div>
  <div class="p-2 bg-primary">Flex item 3</div>
</div>
```

- Flex vertical: Dùng class `.flex-column` hoặc `.flex-column-reverse`

```
<div class="d-flex flex-column">
  <div class="p-2 bg-info">Flex item 1</div>
  <div class="p-2 bg-warning">Flex item 2</div>
  <div class="p-2 bg-primary">Flex item 3</div>
</div>
<hr>
<div class="d-flex flex-column-reverse">
  <div class="p-2 bg-info">Flex item 1</div>
  <div class="p-2 bg-warning">Flex item 2</div>
  <div class="p-2 bg-primary">Flex item 3</div>
</div>
```

- Flex fill/equal width: Dùng class `.flex-fill`

```
<div class="d-flex">
  <div class="p-2 bg-info flex-fill">Flex item 1</div>
  <div class="p-2 bg-warning flex-fill">Flex item 2</div>
</div>
```

```
<div class="p-2 bg-primary flex-fill">Flex item 3</div>
</div>
```

- Flex grow: Dùng class `.flex-grow-1` trong 1 item để nó chiếm không gian còn lại

```
<div class="d-flex">
  <div class="p-2 bg-info">Flex item 1</div>
  <div class="p-2 bg-warning">Flex item 2</div>
  <div class="p-2 bg-primary flex-grow-1">Flex item 3</div>
</div>
```

- Flex order: Dùng class `.order` để sắp xếp các item

```
<div class="d-flex bg-secondary">
  <div class="p-2 bg-info order-3">Flex item 1</div>
  <div class="p-2 bg-warning order-1">Flex item 2</div>
  <div class="p-2 bg-primary order-2">Flex item 3</div>
</div>
```

## 18. Tooltip

Tooltip là 1 popup nhỏ hiện ra khi user đưa chuột vào 1 tag.

```
<a href="#" data-toggle="tooltip" data-placement="top" title="Hooray!">
Hover
</a>
<script>
$(document).ready(function() {
  $('[data-toggle="tooltip"]').tooltip();
});
</script>
```

- Chú ý: Để tooltip chèn thêm popper nữa nhé (trước jquery)

```
<script src=
"https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.3/umd/popper.min.js">
</script>
```

- Position tooltip: có thể thay **top** trong data-placement thành **bottom**, **left**, **right** để chỉ định vị trí.

## 19. Popover

Giống như tooltip nhưng nội dung phong phú hơn.

```
<h1>Test Popover</h1> <h2>Test Popover</h2><hr>
```

```
<a href="#" data-toggle="popover" title="Popover Header" data-
content="Some content inside the popover" data-placement="top">
Toggle popover
</a>
<script>
$(document).ready(function() {
    $('[data-toggle="popover"]').popover();
});
</script>
```

- **Chú ý:** Để tooltip chèn thêm popper nữa nhé (trước jquery)

```
<script src=
"https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.3/umd/popper.min.js">
</script>
```

- **Position tooltip:** có thể thay **top** trong data-placement thành **bottom, left, right** để chỉ định vị trí.
- **Đóng popover:** thêm thuộc tính sau để đóng popover khi user lick bên ngoài

```
<a href="#" data-toggle="popover" title="Popover Header" data-
content="Some content inside the popover" data-trigger="focus" data-
placement="top">
Toggle popover
</a>
```

## 20. Scrollspy

Scrollspy được dùng để tự động cập nhật links trong thanh navigation theo vị trí dữ liệu cuộn.

```
<body data-spy="scroll" data-target=".navbar" data-offset="50">
<nav class="navbar navbar-expand-sm bg-dark navbar-dark fixed-top">
<ul class="navbar-nav">
    <li class="nav-item"><a class="nav-link" href="#section1">Section 1</a></li>
    <li class="nav-item"><a class="nav-link" href="#section2">Section 2</a></li>
    <li class="nav-item"><a class="nav-link" href="#section3">Section 3</a></li>
    <li class="nav-item dropdown">
        <a class="nav-link dropdown-toggle" href="#" id="navbardrop" data-
toggle="dropdown"> Section 4 </a>
        <div class="dropdown-menu">
            <a class="dropdown-item" href="#section41">Link 1</a>
            <a class="dropdown-item" href="#section42">Link 2</a>
        </div>
    </li>
</ul>
</nav>
```



```
<div id="section1" class="container-fluid bg-success" style="padding-top:70px;padding-bottom:70px">
  <h1>Section 1</h1>
  <p>Try to scroll this section and look at the navigation bar while scrolling! Try to scroll this section and look at the navigation bar while scrolling!</p>
  <p>Try to scroll this section and look at the navigation bar while scrolling! Try to scroll this section and look at the navigation bar while scrolling!</p>
</div>
<div id="section2" class="container-fluid bg-warning" style="padding-top:70px;padding-bottom:70px">
  <h1>Section 2</h1>
  <p>Try to scroll this section and look at the navigation bar while scrolling! Try to scroll this section and look at the navigation bar while scrolling!</p>
  <p>Try to scroll this section and look at the navigation bar while scrolling! Try to scroll this section and look at the navigation bar while scrolling!</p>
</div>
<div id="section3" class="container-fluid bg-secondary" style="padding-top:70px;padding-bottom:70px">
  <h1>Section 3</h1>
  <p>Try to scroll this section and look at the navigation bar while scrolling! Try to scroll this section and look at the navigation bar while scrolling!</p>
  <p>Try to scroll this section and look at the navigation bar while scrolling! Try to scroll this section and look at the navigation bar while scrolling!</p>
</div>
<div id="section41" class="container-fluid bg-danger" style="padding-top:70px;padding-bottom:70px">
  <h1>Section 4 Submenu 1</h1>
  <p>Try to scroll this section and look at the navigation bar while scrolling! Try to scroll this section and look at the navigation bar while scrolling!</p>
  <p>Try to scroll this section and look at the navigation bar while scrolling! Try to scroll this section and look at the navigation bar while scrolling!</p>
</div>
<div id="section42" class="container-fluid bg-info" style="padding-top:70px;padding-bottom:70px">
  <h1>Section 4 Submenu 2</h1>
  <p>Try to scroll this section and look at the navigation bar while scrolling! Try to scroll this section and look at the navigation bar while scrolling!</p>
  <p>Try to scroll this section and look at the navigation bar while scrolling! Try to scroll this section and look at the navigation bar while scrolling!</p>
</div>
</body>
```

## 21. Grid trong Bootstrap

Giao diện trong Bootstrap dạng lưới (grid) và được chia làm 12 cột (column) được đặt trong một class **row** . Nếu không muốn dùng tới 12 cột riêng lẻ, bạn có thể nhóm các cột lại để có các cột rộng hơn.

span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1
span 4				span 4				span 4			
span 4				span 8							
span 6						span 6					
span 12											

Hệ thống lưới trong bootstrap là responsive, nghĩa là các cột sẽ được sắp xếp lại tùy theo cỡ của màn hình thiết bị đang dùng để xem trang. Hệ thống lưới trong bootstrap có 4 class cơ bản sau:

- xs (for phones)
- sm (for tablets)
- md (for desktops)
- lg (for larger desktops)

Các class này sẽ được dùng phối hợp cùng nhau khi định dạng các tag để tạo nên sự linh động trong cách trình bày layout cho trang web. Cụ thể một chút: Mỗi cột trong grid sẽ có Padding là 15px sang hai bên và ký hiệu như sau:

.col-xs- : sử dụng cho màn hình có độ phân giải < 768px (Extra small devices – Phones)

.col-lg- : sử dụng cho màn hình có độ phân giải ≥ 1200px (Large devices – Desktop)

.col-md- : sử dụng cho màn hình có độ phân giải ≥ 992px (Medium devices – Desktop)

.col-sm- : sử dụng cho màn hình có độ phân giải  $\geq 768$ px (Small devices – Tablets)

**Ví dụ : Sau đây là 1 vài class**

.col-md-1 : Cột rộng 1 phần (1 /12) với màn hình trung bình (Medium devices)

.col-md-6: Cột rộng một nửa màn hình (6 /12) với màn hình trung bình (Medium devices)

.col-sm-12: Cột rộng toàn màn hình (12 /12) với màn hình máy tính bảng (Small devices)

.col-xm-12: Cột rộng toàn màn hình (12 /12) với màn hình điện thoại (Extra small devices)

	Extra small devices Phones (<768px)	Small devices Tablets ( $\geq 768$ px)	Medium devices Desktops ( $\geq 992$ px)	Large devices Desktops ( $\geq 1200$ px)
Grid behavior	Horizontal at all times	Collapsed to start, horizontal above breakpoints		
Container width	None (auto)	750px	970px	1170px
Class prefix	.col-xs-	.col-sm-	.col-md-	.col-lg-
# of columns	12			
Column width	Auto	~62px	~81px	~97px
Gutter width	30px (15px on each side of a column)			
Nestable	Yes			
Offsets	Yes			
Column ordering	Yes			

Grid đối với các loại thiết bị

## K. THIẾT KẾ LAYOUT CÙNG BOOTSTRAP

### I. Một số quy định

- Các dòng phải đặt trong class **.container** (fixed-width) hoặc **.container-fluid** (full-width)
- Sử dụng các dòng để nhóm các cột nằm ngang nhau.
- Nội dung thông tin của bạn sẽ đặt bên trong các (div) columns. Các columns là con trực tiếp của các (div) rows.
- Sử dụng các class có sẵn trong bootstrap như **.row** hay **.col-sm-4** để nhanh chóng tạo nên layouts chứ đừng tạo riêng class của bạn.
- Khoảng cách giữa các cột đã được tạo sẵn qua thuộc tính padding.
- Tổng các cột trong hệ thống lưới bootstrap là 12, bạn có thể phối hợp các cột để tạo nên 1 cột rộng hơn. Ví dụ : 3 cột phối hợp nhau sẽ là class **.col-sm-4**

### II. Cấu trúc cơ bản của trang

Sau đây là cấu trúc cơ bản của trang layout dùng bootstrap:

```
<div class="container">
  <div class="row">
    <div class="col-**-*"></div>
  </div>
  <div class="row">
    <div class="col-**-*"></div>
    <div class="col-**-*"></div>
    <div class="col-**-*"></div>
  </div>
  <div class="row">
    ...
  </div>
</div>
```

Trong trang layout, tạo tag **<div class="container">** ở ngoài cùng. Kế đó là tạo các row với tag **<div class="row">**. Rồi định nghĩa các cột với các div có class dạng như **.col-\*\*-\***. Với chú ý rằng tổng các số trong **.col-\*\*-\*** luôn là 12 trên mỗi dòng.

### III. Một số dạng layout

#### 1. Layout 3 cột rộng bằng nhau

```
<div class="container-fluid">  
  <div class="row">  
    <div class="col-sm-4" style="background:lavender;">.col-sm-4</div>  
    <div class="col-sm-4" style="background:lavenderblush;">.col-sm-4</div>  
    <div class="col-sm-4" style="background:lavender;">.col-sm-4</div>  
  </div>  
</div>
```

.col-sm-4 .col-sm-4 .col-sm-4

#### 2. Layout 3 cột rộng khác nhau

```
<div class="container-fluid">  
  <div class="row">  
    <div class="col-sm-3" style="background:lavender;">.col-sm-3</div>  
    <div class="col-sm-6" style="background:lavenderblush;">.col-sm-6</div>  
    <div class="col-sm-3" style="background:lavender;">.col-sm-3</div>  
  </div>  
</div>
```

.col-sm-3 .col-sm-6 .col-sm-3

#### 3. Layout 2 cột, lồng nhau

```
<div class="container-fluid">  
  <div class="row">  
    <div class="col-sm-8" style="background:lavender;">  
      <div class="row">  
        <div class="col-sm-6" style="background:lightcyan;">.col-sm-6</div>  
        <div class="col-sm-6" style="background:lightgray;">.col-sm-6</div>  
      </div>  
    <div class="col-sm-4" style="background:lavenderblush;">.col-sm-4</div>  
  </div>  
</div>
```

.col-sm-6 .col-sm-6 .col-sm-4

## 4. Layout hỗn hợp cho Mobile, Tablet và Desktop

Bạn cần nhớ có 4 loại class cho các 4 loại thiết bị khác nhau đó nhé: **xs** (phones), **sm** (tablets), **md** (desktops) và **lg** (larger desktops). Có thể tổ hợp các loại class này lại để tạo nên các dạng thức thay đổi cho layout trên các môi trường thiết bị khác nhau.

```
<div class="container-fluid">  
<div class="row">  
<div class="col-sm-3 col-md-6 col-lg-4 bg-success">sm-3 md-6 lg-4</div>  
<div class="col-sm-9 col-md-6 col-lg-8 bg-warning">sm-9 md-6 lg-8</div>  
</div>  
</div>
```

Xem thử khi phóng to trình duyệt:



Có cửa sổ trình duyệt lại 1 chút, bạn sẽ thấy khác:



Có cửa sổ trình duyệt lại 1 chút, nữa bạn sẽ thấy khác:



## IV. Những cách để có một layout đẹp.

Để có 1 layout đẹp,, bạn cần làm gì?

### 1. Tự thiết kế

Bạn vẽ layout cụ thể theo ý tưởng (trên giấy) trước tiên. Đồng thời quyết định các yếu tố: màu sắc, số cột, các thành phần như: logo, menu, slider, footer, ... Rồi tự làm trang html theo ý tưởng với nền tảng là các kiến thức ở trên.

Bạn cũng có thể vẽ layout bằng photoshop rồi xuất sang html. Với trang html có được từ photoshop, bạn tiếp tục custom để nó chạy trên bootstrap là hay nhất.

---

Với việc tự thiết kế, bạn sẽ gặp nhiều trở ngại nếu như khả năng mỹ thuật chưa cao, và bạn phải chịu khó nghiên cứu thêm về html, css, bootstrap.

## 2. Tự thiết kế với tool

Có nhiều phần mềm giúp bạn thiết kế layout đẹp. Bạn có thể tìm trên Google với từ khóa như **tools design layout bootstrap**. Sẽ có nhiều tools giúp bạn làm việc này (1 tool hay có tên **Artisteer** ).

## 3. Tìm/mua template trên mạng

Trên internet có rất nhiều template miễn phí và có phí được thiết kế dựa trên bootstrap. Bạn tìm chúng chọn và dùng là cách tiết kiệm thời gian nhất. Có rất nhiều template đẹp đó nhé. Từ khóa bạn có thể dùng để search : **template html bootstrap free**