

# **GIÁO TRÌNH**

## **HTML VÀ THIẾT KẾ WEBSITE**

**(Mã số giáo trình: 3CI3)**



HÀ NỘI. 2005

## LỜI MỞ ĐẦU

Ngôn ngữ Siêu văn bản HTML (**H**yper **T**ext **M**arkup **L**anguage) là ngôn ngữ biểu diễn văn bản cho phép ta đưa vào một văn bản nhiều thuộc tính cần thiết để có thể truyền thông quảng bá trên mạng toàn cục WWW (**W**orld **W**ide **W**eb). HTML cho phép ta đưa hình ảnh đồ họa vào văn bản, thay đổi cách bày trí của văn bản, và tạo những tài liệu siêu văn bản có khả năng đối thoại tương tác với người dùng.

HTML chủ yếu xoay quanh khái niệm “thẻ” (tag) làm nền tảng. Một ví dụ về tag là <B>. Hầu hết các chức năng trên HTML có thể mở và thẻ đóng tạo thành một cặp giới hạn một đoạn văn bản. Ví dụ <B> và </B> là thẻ “**bold**” nghĩa là chữ béo. Toàn bộ đoạn văn bản giữa <B> và </B> sẽ được thể hiện dưới dạng chữ béo (hay chữ đậm) khi văn bản đó được xem bằng một trình duyệt tương ứng. Ví dụ <B> xin chào </B> sẽ được hiện lên là **xin chào**.

Để tạo một siêu văn bản, ta có thể dùng bất cứ một chương trình soạn thảo nào (ví dụ: NC-Norton Commander, EDIT - của DOS, NotePad hay Write - của Windows 3.x, WordPad của Win95, WinWord 2.0 hay 6.0, FoxPro, Borland C++ IDE, Borland Pascal IDE, FrontPage, TextPad, v.v...), chỉ cần nắm được các thẻ của HTML, và khi ghi vào đĩa thì cần lưu dưới dạng file text. Tên file có đuôi mở rộng là HTM (hoặc HTML). Song có một hạn chế là dạng văn bản khi ta soạn với khi ta xem sau này (trên WWW) không giống nhau. Chính vì lẽ đó mà nhiều hãng tung ra phần mềm soạn siêu văn bản **What You See Is What You Get** (WYSIWYG- cái ta thấy cũng là cái ta có được). Hãng Microsoft cũng đã tung ra một tiện ích được sử dụng rộng rãi. Ngoài ra, còn có nhiều tiện ích chuyển đổi từ dạng RTF (có thể soạn bằng WinWord hay WordPad) sang HTML, hay các phiên bản sau này của trình duyệt Web như NetScape có sẵn luôn chức năng này. Các tiện ích đó đều giống nhau ở chỗ cho phép ta gõ trực tiếp các thẻ vào văn bản nhưng cũng có thể dùng tổ hợp của thanh công cụ (toolbar), hộp thoại (dialog), thực đơn (menu) hay danh sách các lựa chọn (pop-up list). Để gạch dưới đoạn văn bản trong ví dụ vừa rồi, ta có thể gõ vào xin chào, dùng chuột chọn, rồi bấm vào nút Underline trên thanh công cụ. Văn bản sẽ tự động được chuyển thành <U>xin chào</U> (ở đây thì <U> và </U> là cặp mở/đóng của tag có chức năng giới hạn đoạn văn bản cần được gạch dưới).

*Ngày nay, do sự phát triển như thác lũ của mạng toàn cục, HTML cũng ngày càng trở nên phức tạp và hoàn thiện hơn để đáp ứng được những yêu cầu mới nảy sinh trong quá trình phát triển đó (như âm thanh, hình ảnh động, v.v...). Người ta gọi đó là những phiên bản của HTML và đánh số để biểu thị. HTML 2, HTML 2+, HTML 3, ... là để chỉ những phiên bản sau này.*

*Một trong những điểm mạnh của HTML là một văn bản bất kỳ nếu tuân thủ tiêu chuẩn HTML đều có thể hiện lên màn hình hay in ra, tóm lại là hiểu được, bởi bất kỳ loại phần mềm hay máy tính nào mà người đọc có, không phân biệt trình duyệt nào (NetScape trên Windows hay Lynx trên UNIX, thậm chí cho người khiếm thị bằng phần mềm đặc biệt).*

*Người ta còn đang tranh cãi nhiều trên Internet về HTML “tốt”. Định hướng nguyên thủy của HTML là tạo ra một phương pháp vạn năng để lưu giữ và thể hiện thông tin. Sau này người ta coi HTML như một ngôn ngữ định hướng nội dung – “trong tài liệu có gì” quan trọng hơn nhiều so với “tài liệu có đẹp không”.*

*Tác giả rất mong nhận được sự góp ý từ bạn đọc*

*Xin chân thành cảm ơn.*

- Tên môn học: **HTML VÀ THIẾT KẾ WEB**
- Mã số môn học: 3CI3
- Thời gian: Lý thuyết + thực hành 45T
- Mục tiêu: Cung cấp cho học viên các kiến thức cơ bản liên quan đến ngôn ngữ HTML và thiết kế Web.
- Những kiến thức cần phải được trang bị trước khi học: Có kiến thức về sử dụng máy tính.
- Nội dung môn học:

**Chương I: NHỮNG KHÁI NIỆM CƠ SỞ.**

**Chương II: TRÌNH BÀY TRANG.**

**Chương III: DANH SÁCH VÀ BẢNG TRONG HTML.**

**Chương IV: ĐƯA HÌNH ẢNH VÀO TÀI LIỆU HTML.**

**Chương V: CÁC MỐI LIÊN KẾT SIÊU VĂN BẢN.**

**Chương VI: BÀY TRÍ NỀN VÀ KHUNG.**

**Chương VII: BIỂU MẪU STYLE VÀ CASCADING STYLE SHEET.**

**Chương VIII: CÔNG CỤ SOẠN THẢO TRỰC QUAN WEB (Microsoft FrontPage 2003)**

- Đối tượng học: Những người cần biết các kiến thức cơ bản về HTML và Website để thiết kế Website tĩnh.
- Biên soạn: Bộ môn Các hệ thống thông tin, Khoa Công nghệ thông tin, Trường ĐH Công Nghệ, ĐHQGHN.

## MỤC LỤC

<b>LỜI MỞ ĐẦU</b>	<b>1</b>
<b>CHƯƠNG I. NHỮNG KHÁI NIỆM CƠ SỞ</b>	<b>9</b>
<b>I.1. World Wide Web là gì?</b>	<b>9</b>
<b>I.2. HTML là gì?</b>	<b>12</b>
<b>I.3. Các đặc điểm của siêu văn bản</b>	<b>13</b>
I.3.1. Độc lập với phần cứng và phần mềm	13
I.3.2. Độc lập với khái niệm trang và thứ tự các trang	14
I.3.3. Website và trang chủ - homepage	14
<b>I.4. Soạn thảo văn bản - những vấn đề chung</b>	<b>15</b>
I.4.1. Trang mã nguồn HTML và trang Web	15
I.4.2. Các thẻ HTML	16
I.4.3. Các quy tắc chung	17
I.4.4. Cấu trúc của một tài liệu HTML	18
I.4.5. Các phần tử HTML (HTML element)	19
<b>Bài tập</b>	<b>19</b>
<b>CHƯƠNG II. TRÌNH BÀY TRANG</b>	<b>20</b>
<b>II.1. Tạo tiêu đề</b>	<b>20</b>
<b>II.2. Thẻ trình bày trang</b>	<b>21</b>
II.2.1. Một số thẻ chính	21
II.2.2. Các thuộc tính của thẻ trình bày trang	25
<b>Bài tập</b>	<b>27</b>
<b>CHƯƠNG III. DANH SÁCH VÀ BẢNG TRONG HTML</b>	<b>28</b>
<b>III.1. Các kiểu danh sách</b>	<b>28</b>
III.1.1. Danh sách không đánh số thứ tự	28
III.1.2. Danh sách đánh số thứ tự	29
III.1.3. Danh sách các định nghĩa	31
III.1.4. Danh sách phối hợp, lồng nhau	31

<b>III.2. Bảng biểu .....</b>	<b>32</b>
III.2.1. Khung cấu trúc.....	32
III.2.2. Một số lưu ý về bảng.....	33
III.2.3. Các ví dụ.....	37
<b>Bài tập.....</b>	<b>40</b>
<b>CHƯƠNG IV. ĐƯA HÌNH ẢNH VÀO TÀI LIỆU HTML .....</b>	<b>41</b>
<b>IV.1. Hình ảnh tĩnh .....</b>	<b>41</b>
IV.1.1. Tập ảnh.....	41
IV.1.2. Thẻ <IMG...>.....	41
<b>IV.2. Các thuộc tính của thẻ chèn hình ảnh.....</b>	<b>41</b>
IV.2.1. Thuộc tính ALT .....	41
IV.2.2. Thuộc tính WIDTH và HEIGHT .....	42
IV.2.3. Thuộc tính ALIGN .....	43
IV.2.4. Thuộc tính VSPACE và HSPACE .....	43
<b>Bài tập.....</b>	<b>44</b>
<b>CHƯƠNG V. CÁC MỐI LIÊN KẾT SIÊU VĂN BẢN.....</b>	<b>45</b>
<b>V.1. Thẻ neo và mối liên kết .....</b>	<b>45</b>
V.1.1. Thuộc tính HREF .....	45
V.1.2. Liên kết ra ngoài – External Links .....	45
V.1.3. Địa chỉ tuyệt đối.....	46
V.1.4. Địa chỉ tương đối .....	46
V.1.5. Liên kết nội tại – Internal Link.....	46
<b>V.2. Dùng hình ảnh làm đầu mối liên kết .....</b>	<b>47</b>
V.2.1. Thay chữ bằng hình.....	47
V.2.2. Image Map - thẻ AREA .....	47
<b>V.3. Đưa âm thanh vào tài liệu .....</b>	<b>48</b>
V.3.1. Liên kết đến tệp âm thanh .....	48
V.3.2. Tạo âm thanh nền.....	48

<b>V.4. Đưa Video vào tài liệu .....</b>	<b>49</b>
V.4.1. Chèn tệp Video .....	49
V.4.2. Nhúng tệp video.....	49
<b>Bài tập.....</b>	<b>50</b>
<b>CHƯƠNG VI. BÀY TRÍ NỀN VÀ KHUNG .....</b>	<b>51</b>
<b>VI.1. Màu nền và văn bản .....</b>	<b>51</b>
VI.1.1. Đặt màu nền.....	51
VI.1.2. Màu chữ của văn bản .....	51
VI.1.3. Màu của đầu mối liên kết - Thuộc tính LINK, VLINK và ALINK.....	51
VI.1.4. Thuộc tính và mã màu .....	52
<b>VI.2. Nạp hình ảnh làm nền cho trang văn bản .....</b>	<b>53</b>
VI.2.1. Thuộc tính BACKGROUND .....	53
VI.2.2. Water mark .....	53
VI.2.3. Hãy ký tên vào tài liệu của mình.....	54
<b>VI.3. Khung – Frames.....</b>	<b>55</b>
VI.3.1. Trang trí khung.....	55
VI.3.2. Thành phần FRAMESET .....	56
<b>VI.4. Thiết lập Target, thẻ NOFRAME và IFRAME .....</b>	<b>59</b>
VI.4.1. Thiết lập Target .....	59
VI.4.2. Thẻ NOFRAMES .....	60
VI.4.2. Nhúng frame - thẻ IFRAME .....	60
<b>Bài tập.....</b>	<b>61</b>
<b>CHƯƠNG VII. BIỂU MẪU STYLE VÀ CASCADING STYLE SHEET .....</b>	<b>62</b>
<b>VII.1. FORM.....</b>	<b>62</b>
VII.1.1. FORM là gì?.....	62
VII.1.2. Các thành phần trong FORM.....	63
VII.1.3. Thêm tính cấu trúc cho FORM .....	70
<b>VII.2. Cascading style sheet.....</b>	<b>72</b>

VII.2.1. Inline Style .....	72
VII.2.2. Giới thiệu Style Sheet.....	74
VII.2.3. Javascript Style Sheet .....	75
VII.2.4. Thuật ngữ Style Sheet .....	77
VII.2.5. Các chú thích trong Style Sheet .....	92
VII.2.6. Lợi ích của các Style Sheet.....	92
VII.2.7. Kết hợp Style Sheet với HTML .....	93
VII.2.8. Thứ tự ưu tiên của các style (Cascading) .....	97
<b>CHƯƠNG VIII. CÔNG CỤ SOẠN THẢO TRỰC QUAN WEB (MICROSOFT FRONTPAGE 2003) .....</b>	<b>100</b>
<b>VIII.1. Tạo một trang Web.....</b>	<b>100</b>
VIII.1.1. Bắt đầu sử dụng FrontPage2003 .....	100
VIII.1.2. Tạo một trang từ một template .....	101
VIII.1.3. Tạo và lưu một trang mới.....	102
VIII.1.4. Tạo một đề mục .....	103
VIII.1.5. Chọn font và màu.....	104
<b>VIII.2. Tổ chức một trang với các liên kết, danh sách và bảng .....</b>	<b>107</b>
VIII.2.1. Thêm một hyperlink vào một trang Web .....	107
VIII.2.2. Tạo một danh sách .....	110
VIII.2.3. Tổ chức một trang với các bảng .....	112
VIII.2.4. Hiện thị hình ảnh trên một trang Web .....	120
VIII.2.5. Tạo một Web site mới.....	133
VIII.2.5.Khai thác site mới .....	135
<b>VIII.3. Phát triển nhanh một site với các template .....</b>	<b>142</b>
VIII.3.1. Chọn một template Web site .....	143
VIII.3.2. Tạo một Web site mới.....	144
VIII.3.3. Tạo tùy biến Web site mới của ta .....	145
VIII.3.4. Thêm và loại bỏ các lời chú thích.....	146



VIII.3.5. Khai thác template Personal Web Site.....	147
VIII.3.6. Thêm một tem thời gian vào một trang Web .....	148
VIII.3.7. Lưu các thay đổi sang một site .....	149
<b>VIII.4. Tạo một site với sự trợ giúp của Wizard trong FrontPage 2003 .....</b>	<b>150</b>
VIII.4.1. Mở một wizard tạo site.....	150
VIII.4.2. Nhập một site hiện có vào FrontPage .....	152
VIII.4.3. Chọn một phương pháp import.....	152
VIII.4.4. Chọn vị trí để lưu site .....	155
VIII.4.5. Thu thập thông tin phản hồi từ các khách tham quan Web site của ta .....	156
VIII.4.6. Lưu thông tin phản hồi của khách tham quan sang một file.....	158
VIII.4.7. Nhận thông tin phản hồi của khách tham quan bằng email .....	160
<b>Bài tập.....</b>	<b>161</b>
<b>BẢNG CÁC TỪ VIẾT TẮT .....</b>	<b>162</b>
<b>TÀI LIỆU THAM KHẢO.....</b>	<b>163</b>

# CHƯƠNG I. NHỮNG KHÁI NIỆM CƠ SỞ

## I.1. World Wide Web là gì?

World Wide Web (WWW) là một mạng các tài nguyên thông tin. WWW dựa trên 3 cơ chế để các tài nguyên này trở nên sẵn dùng cho người xem càng rộng rãi nhất càng tốt:

- Cơ chế đặt tên cùng dạng đối với việc định dạng các tài nguyên trên WWW (như các URL)
  - Các giao thức, để truy nhập tới các tài nguyên qua WWW (như HTTP)
  - Siêu văn bản, để dễ dàng chuyển đổi giữa các tài nguyên (như HTML).
- Các ràng buộc giữa ba cơ chế được nêu rõ dưới đây

### Giới thiệu về URL:

Mọi tài nguyên sẵn dùng trên WWW – tài liệu HTML, ảnh, video clip, chương trình,... - có một địa chỉ mà có thể được mã hóa bởi một URL.

Các URL thường gồm 3 phần:

- Việc đặt tên của các cơ chế dùng để truy nhập tài nguyên
- Tên của máy tính lưu trữ (tổ chức) tài nguyên
- Tên của bản thân tài nguyên, như một đường dẫn

Ví dụ coi URL chỉ rõ trang W3C Technical Reports:

`http://www.w3.org/TR`

URL này có thể được đọc như sau: Có một tài liệu sẵn dùng theo giao thức HTTP, đang lưu trong máy `www.w3.org`, có thể truy nhập theo đường dẫn “/TR”. Các cơ chế khác ta có thể thấy trong các tài liệu HTML bao gồm “mailto” đối với thư điện tử và “ftp” đối với FTP.

Đây là một ví dụ khác về URL. Ví dụ này ám chỉ tới hộp thư (mailbox) của người dùng:

....đây là văn bản ...

Mọi góp ý, xin gửi thư tới

`<A ref="mailto:joe@someplace.com">Joe Cool</A>`

### Các định danh đoạn (fragment identifiers):

Một số URL ám chỉ tới việc định vị một tài nguyên. Kiểu này của URL kết thúc với “#” theo sau bởi một dấu hiệu kết nối (gọi là *các định danh đoạn*). Ví dụ, đây là một URL đánh dấu một móc tên là section\_2:

```
http://somesite.com/html/top.html#section_2
```

### Các URL tương đối:

Một URL tương đối không theo cơ chế đặt tên. Đường dẫn của nó thường tham chiếu tới một tài nguyên trên cùng một máy như tài liệu hiện tại. Các URL tương đối có thể gồm các thành phần đường dẫn tương đối (như “..” nghĩa là một mức trên trong cấu trúc được định nghĩa bởi đường dẫn), và có thể bao gồm các dấu hiệu đoạn.

Các URL được giải quyết để cho toàn các URL sử dụng một URL gốc. Như một ví dụ của giải pháp URL tương đối, giả sử chúng ta có URL gốc

“http://www.acme.com/support/intro.html”. URL tương đối trong đánh dấu dưới đây cho một liên kết siêu văn bản:

```
<A href="suppliers.html">Suppliers</A>
```

sẽ mở rộng thành URL đầy đủ

“http://www.acme.com/support/suppliers.html” trong khi URL tương đối trong việc đánh dấu cho một ảnh dưới đây

```
<IMG src="../icons/logo.gif" alt="logo">
```

sẽ mở rộng thành URL đầy đủ “http://www.acme.com/icons/logo.gif”

Trong HTML, các URL được dùng để:

- Liên kết tới tài liệu hoặc tài nguyên khác, (xem A và LINK)
- Liên kết tới kiểu dạng bên ngoài hoặc kịch bản (script) (xem LINK và SCRIPT)
- Gồm một ảnh, đối tượng, hoặc applet trong một trang, (xem IMG, OBJECT, APPLET và INPUT)
- Tạo một bản đồ ảnh (xem MAP và AREA)
- Tạo một form (xem FORM)
- Tạo một khung tài liệu (xem FRAME và IFRAME)
- Trích dẫn một chỉ dẫn bên ngoài (xem Q, BLOCKQUOTE, INS và DEL)
- Tham khảo các quy ước siêu dữ liệu mô tả một tài liệu (xem HEAD)

Ngày nay người ta dùng máy tính như một công cụ rất hữu ích để truy nhập Internet, chủ yếu là tìm kiếm thông tin. Các thông tin này có thể là các văn bản, hình ảnh âm thanh hay thông tin đa phương tiện... Với giao diện thân thiện bởi việc sử dụng các ký hiệu, biểu tượng rất gợi tả gần giống với các hình ảnh đời thường và chỉ cần những thao tác đơn giản ta đã có ngay thông tin cần tìm kiếm ở trước mặt.

Nhu cầu sử dụng máy tính để truy cập Internet tìm kiếm thông tin ngày càng nhiều và người sử dụng máy tính có trình độ tin học và tiếng Anh để hiểu các thông báo của máy khác nhau. Làm thế nào để mọi người có thể dễ dàng sử dụng máy tính để truy cập Internet như một công cụ phục vụ đắc lực cho việc tra cứu tìm kiếm thông tin trên mạng thông tin rộng lớn nhất toàn cục.

Việc này trở nên dễ dàng hơn bởi ý tưởng siêu văn bản (Hypertext). Siêu văn bản là các văn bản “thông minh” có thể giúp người đọc tìm và cung cấp cho họ các tài liệu liên quan. Người đọc không phải mất công tìm kiếm trong kho thông tin Internet vô tận.

Khái niệm siêu văn bản do nhà tin học Ted Nelson đề xuất vào năm 1965 như một ước mơ (“Computer Dreams”) về khả năng của máy tính trong tương lai. Ông hy vọng về các máy tính có trí tuệ như con người, biết cách truy tìm các thông tin cần thiết.

Dự án thực hiện siêu văn bản là của một kỹ sư trẻ người Anh tên là Tim Berners – Lee. Sau khi tốt nghiệp Đại học Oxford (Anh) năm 1976, năm 1980 Tim đã viết một chương trình mô phỏng mối liên kết hai chiều bất kỳ trên một đồ thị như kiểu liên kết siêu văn bản. Năm 1989, trong khi làm việc tại Viện nghiên cứu kỹ thuật hạt nhân châu Âu (CERN) tại Berne (Thụy sỹ), thấy các đồng nghiệp rất vất vả trong việc tra tài liệu, Tim đã đưa ra một đề án lưu trữ siêu văn bản trên máy tính sao cho dễ dàng tìm kiếm tài liệu hơn.

Trong thế giới siêu văn bản WWW, người sử dụng có thể dễ dàng đi từ tài liệu này sang tài liệu khác thông qua các mối liên kết. Như vậy ta có thể du lịch trong xa lộ thông tin phong phú trong khi vẫn ngồi tại nhà. Kỹ thuật siêu văn bản

giúp cho ta không phải sang tận Luvrơ ở Paris mà vẫn có thể chiêm ngưỡng được các kiệt tác hội họa. Chính nó đã góp phần tạo ra bước phát triển bùng nổ của Internet trong những năm gần đây.

## **I.2. HTML là gì?**

Để phổ biến thông tin trên toàn cục, cần một ngôn ngữ phổ biến và dễ hiểu, một kiểu việc phổ biến tiếng mẹ đẻ mà toàn bộ các máy tính có thể hiểu được. Ngôn ngữ phổ biến dùng bởi World Wide Web là HTML (Hyper Text Markup Language).

HTML cho tác giả các ý nghĩa để:

Phổ biến các tài liệu trực tuyến với các heading, văn bản, bảng, danh sách, ảnh,..v.v...

Truy tìm thông tin trực tuyến theo các liên kết siêu văn bản bằng việc kích vào một nút

Thiết kế các định dạng cho việc kiểm soát các giao dịch (transaction) với các thiết bị từ xa, đối với người dùng trong việc tìm kiếm thông tin, tạo các sản phẩm, đặt hàng,..v.v...

Bao gồm spread-sheets, video clips, sound clips, và các ứng dụng trực tiếp khác trong các tài liệu của họ.

Các trang Web đầy sinh động mà bạn thấy trên World Wide Web là các trang siêu văn bản được viết bằng ngôn ngữ đánh dấu siêu văn bản hay *HTML - HyperText Markup Language*. *HTML cho phép bạn tạo ra các trang phối hợp hài hòa văn bản thông thường với hình ảnh, âm thanh, video, các mối liên kết đến các trang siêu văn bản khác...*

*Tên gọi ngôn ngữ đánh dấu siêu văn bản phản ánh đúng thực chất của công cụ này:*

***Đánh dấu (Markup):*** *HTML là ngôn ngữ của các thẻ đánh dấu - Tag. Các thẻ này xác định cách thức trình bày đoạn văn bản tương ứng trên màn hình.*

***Ngôn ngữ (Language):*** *HTML là một ngôn ngữ tương tự như các ngôn ngữ lập trình, tuy nhiên đơn giản hơn. Nó có cú pháp chặt chẽ để viết các lệnh thực*

hiện việc trình diễn văn bản. Các từ khoá có ý nghĩa xác định được cộng đồng Internet thừa nhận và sử dụng. Ví dụ *b* = bold, *ul* = unordered list, ...

**Văn bản (Text):** HTML đầu tiên và trước hết là để trình bày văn bản và dựa trên nền tảng là một văn bản. Các thành phần khác như hình ảnh, âm thanh, hoạt hình.. đều phải "cắm neo" vào một đoạn văn bản nào đó.

**Siêu văn bản (Hyper):** HTML cho phép liên kết nhiều trang văn bản rải rác khắp nơi trên Internet. Nó có tác dụng che dấu sự phức tạp của Internet đối với người sử dụng. Người dùng Internet có thể đọc văn bản mà không cần biết đến văn bản đó nằm ở đâu, hệ thống được xây dựng phức tạp như thế nào. HTML thực sự đã vượt ra ngoài khuôn khổ khái niệm văn bản cổ điển.

### **I.3. Các đặc điểm của siêu văn bản**

#### **I.3.1. Độc lập với phần cứng và phần mềm**

HTML độc lập với phần cứng và phần mềm. Một tài liệu HTML được viết bằng một phần mềm soạn thảo cụ thể bất kỳ, trên một máy cụ thể nào đó đều có thể đọc được trên bất kì một hệ thống tương thích nào.

Điều này có nghĩa là các tệp siêu văn bản có thể được trình duyệt hiển thị trên MAC hay PC tùy ý mà không phải sửa chữa thay đổi gì. Sở dĩ có được tính chất này là vì các thẻ chỉ dẫn đạt yêu cầu cần phải làm gì chứ không cụ thể cần làm như thế nào.

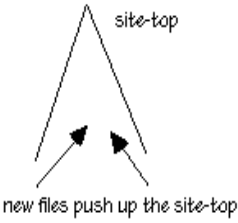
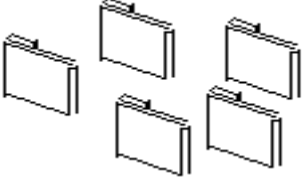
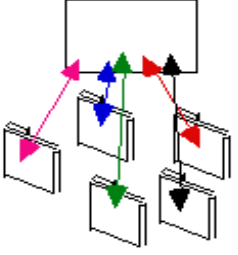
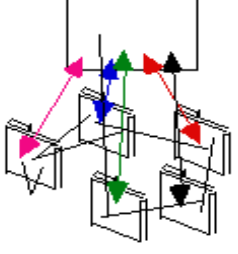
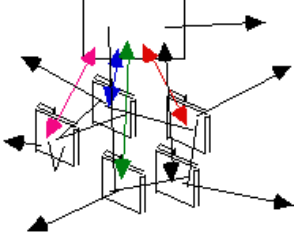
Cũng vì lẽ đó mà bạn không thể chắc chắn trang tài liệu siêu văn bản của bạn sẽ hiện lên màn hình chính xác là như thế nào vì còn tùy theo trình duyệt thể hiện các thành phần HTML ra sao.

Trong thực tiễn, HTML chỉ thực sự độc lập đối với phần cứng, chưa hoàn toàn độc lập đối với phần mềm. Chỉ phần cốt lõi là chuẩn hoá, còn các phần mở rộng do từng nhà phát triển xây dựng thì không hoàn toàn tương thích nhau.

### 1.3.2. Độc lập với khái niệm trang và thứ tự các trang

Một tính chất nữa là HTML độc lập với khái niệm trang. Văn bản được trình bày tùy theo kích thước của cửa sổ hiển thị: cửa sổ rộng bề ngang thì sẽ thu ngắn hơn, cửa sổ hẹp bề ngang thì sẽ được kéo dài ra để hiển thị cho hết nội dung. Độ dài của văn bản HTML thực sự không bị hạn chế.

### 1.3.3. Website và trang chủ - homepage

	<p>Một website là một bó các trang web liên kết với nhau và liên kết với các trang ở bên ngoài chẳng chịt như mạng nhện. Hàng triệu Website liên kết với nhau tạo thành World Wide Web – WWW</p>
	<p>Có thể tưởng tượng như một ngọn núi nhỏ các trang web mỗi ngày một cao thêm, được phát triển bằng cách thêm vào nhiều trang web lý thú khác nữa</p>
	<p>Có thể minh họa hình ảnh của một website như trong hình vẽ bên</p>
	<p>Trang chủ hay trang chính - 'home page' có thể hiểu là cửa chính - 'front door' để thâm nhập vào kho thông tin liên kết chằng chịt ấy.</p>
	<p>Vậy home page là trang web mà bộ duyệt sẽ mở ra đầu tiên mỗi khi người dùng bắt đầu thăm website.</p>

## **I.4. Soạn thảo văn bản - những vấn đề chung**

### **I.4.1. Trang mã nguồn HTML và trang Web**

*Trang mã nguồn HTML là một tệp văn bản bình thường gồm các kí tự ASCII, có thể được tạo ra bằng bất cứ trình soạn thảo thông thường nào.*

*Theo quy ước, tất cả các tệp mã nguồn của trang siêu văn bản phải có đuôi là .html hoặc .htm.*

*Khi trình duyệt (browser) đọc trang mã nguồn HTML, nó sẽ dịch các thẻ lệnh và hiển thị lên màn hình máy tính thì ta thường gọi là trang Web. Vậy trang web không tồn tại trên đĩa cứng của máy tính cục bộ. Nó là cái thể hiện của trang mã nguồn qua việc xử lý của trình duyệt. Như sau này ta sẽ thấy, các trình duyệt khác nhau có thể hiển thị cùng một trang mã nguồn không hoàn toàn giống nhau.*

*Nói soạn thảo siêu văn bản tức là tạo ra trang mã nguồn HTML đúng quy định để độ duyệt hiểu được và hiển thị đúng.*

Sử dụng HTML để soạn thảo các trang siêu văn bản, về nguyên tắc cũng không khác mấy so với dùng các bộ soạn thảo văn bản thông thường. Chẳng hạn, trong soạn thảo văn bản thông thường, để làm nổi bật các tiêu đề ta phải đánh dấu nó và chọn cỡ to, căn chính giữa... Chương trình soạn thảo văn bản sẽ chèn các dấu hiệu thích hợp (ta không nhìn thấy được) vào đầu và cuối đoạn tiêu đề được chọn để thể hiện nó theo yêu cầu.

Với HTML cũng tương tự như vậy. Để làm nổi bật các tiêu đề ta cần đánh dấu điểm bắt đầu và điểm kết thúc của đoạn tiêu đề bằng cặp thẻ Heading, ví dụ <H1>... </H1>. Bộ duyệt sẽ hiển thị đoạn này với cỡ chữ to hơn và căn chính giữa.

Ví dụ, dòng sau đây trong tài liệu HTML

```
<H1>Tiêu đề mức 1</H1>
```

sẽ được trình duyệt hiển thị thành

**Tiêu đề mức 1**



Hiện nay có nhiều công cụ soạn thảo siêu văn bản mạnh như *Microsoft FrontPage, Dream Weaver..* với giao diện trực quan và tự động sinh mã HTML, cho phép soạn thảo siêu văn bản như soạn thảo thông thường.

Tuy nhiên, việc tìm hiểu cú pháp của HTML, nắm vững ý nghĩa của các thẻ khác nhau vẫn rất cần thiết để có thể tạo ra các trang Web động, tương tác với người sử dụng, tức là các ứng dụng Web sau này.

#### **1.4.2. Các thẻ HTML**

Các thẻ dùng để báo cho trình duyệt cách thức trình bày văn bản trên màn hình hoặc dùng để chèn một mối liên kết đến các trang khác, một đoạn chương trình khác...

Mỗi thẻ gồm một từ khoá - KEYWORD - bao bọc bởi hai dấu "bé hơn" (<) và "lớn hơn" (>).

Hầu hết các lệnh thẻ hiện bằng một cặp hai thẻ: thẻ mở (<KEYWORD>) và thẻ đóng (</KEYWORD>). Dấu gạch chéo ("/") kí hiệu thẻ đóng. Lệnh sẽ tác động vào đoạn văn bản nằm giữa hai thẻ.

<KEYWORD> Đoạn văn bản chịu tác động của lệnh</KEYWORD>

Một số thẻ không có cặp, chúng được gọi là các thẻ **rỗng** hay thẻ **đơn**. Chỉ có thẻ mở <KEYWORD> mà thôi.

Nhiều thẻ có kèm các thuộc tính (attribute), cung cấp thêm các tham số chi tiết hơn cho việc thực hiện lệnh. Các thuộc tính được chia làm hai loại: thuộc tính bắt buộc và thuộc tính không bắt buộc hay tùy chọn.

Một thuộc tính là bắt buộc nếu như phải có nó thì thẻ lệnh mới thực hiện được. Ví dụ, để chèn một hình ảnh vào trang tài liệu ta dùng thẻ <IMG> (Image). Tuy nhiên, cần chỉ rõ cái ảnh nào sẽ được dán vào đây. Điều này được thiết lập bằng thuộc tính SRC="địa chỉ của tệp ảnh". Thuộc tính SRC là bắt buộc phải có đối với thẻ <IMG>.

### 1.4.3. Các quy tắc chung

Một số điều cần lưu ý khi soạn thảo siêu văn bản bằng HTML:

- Nhiều dấu cách liên nhau cũng chỉ có tác dụng như một dấu cách. Bạn phải sử dụng thẻ để thể hiện nhiều dấu giãn cách liên nhau.
- Gõ Enter để xuống dòng được xem như một dấu cách, để xuống hàng thì chúng ta phải sử dụng thẻ tương ứng
- Có thể viết tên thẻ không phân biệt chữ in thường và in hoa.
- Vì các ký tự dấu lớn hơn ">", dấu nhỏ hơn "<" đã được dùng làm thẻ đánh dấu, do đó để hiển thị các ký tự này HTML quy định cách viết: &gt; &lt;

Nói chung, quy tắc viết các ký tự đặc biệt trong HTML là tên quy định của ký tự nằm giữa dấu ampersand - & và dấu chấm phẩy ' ; '

&tên\_quy\_định;

5 ký tự đặc biệt hay dùng

Ký tự	Cách viết
<	&lt;
>	&gt;
“	&quot;
Ký tự trắng	&nbsp;
&	&amp;

Có thể chèn các dòng bình luận, chú thích... vào trang mã nguồn bằng cách đặt giữa cặp dấu chú thích <!-- và -->. Trình duyệt sẽ bỏ qua không xét đến phần mã nằm giữa cặp dấu đó:

<!-- Dòng chú thích -->

#### 1.4.4. Cấu trúc của một tài liệu HTML

Mọi tài liệu HTML đều có khung cấu trúc như sau:

```
<HTML>
<HEAD><TITLE></TITLE></HEAD>
<BODY>
</BODY>
</HTML>
```

Hãy xem trình duyệt hiển thị tài liệu trên như thế nào. Dĩ nhiên là một trang trắng chưa có nội dung gì cả !

Giữa cặp thẻ tiêu đề <TITLE>... </TITLE> là dòng chữ sẽ hiện lên trên thanh tiêu đề của cửa sổ khi trình duyệt đọc tài liệu. Nếu bỏ trống thì trình duyệt sẽ cho hiện tên tệp thay vào đó.

Toàn bộ nội dung của tài liệu nằm giữa hai thẻ xác định thân của trang <BODY>... </BODY>. Các dòng văn bản, hình ảnh, âm thanh, video, các mối liên kết... tạo nên trang Web đều phải nằm ở đây.

Ví dụ như tài liệu HTML đơn giản dưới đây:

```
<HTML>
<HEAD><TITLE>Chỗ này là tiêu đề</TITLE></HEAD>
<BODY>
  Toàn bộ nội dung của tài liệu nằm ở đây: Các đoạn
  văn bản xen lẫn hình ảnh, âm thanh, video, các liên
  kết đến vị trí khác, tài liệu khác...
</BODY>
</HTML>
```

Hãy xem trình duyệt trình bày tài liệu trên như thế nào.

Nhớ lại rằng nhiều dấu cách chỉ được coi như một, dấu xuống dòng chỉ được thể hiện như một dấu cách nên tài liệu trên hoàn toàn tương đương với tài liệu sau đây:

```
<HTML>
<HEAD>
<TITLE>Chỗ này là tiêu đề </TITLE>
```

```
</HEAD>
<BODY>
  Toàn bộ nội dung của tài liệu nằm ở đây: Các đoạn
  văn bản xen lẫn hình ảnh, âm thanh, video, các liên
  kết đến vị trí khác, tài liệu khác...<P>
</BODY>
</HTML>
```

Tuy nhiên, để dễ theo dõi và phát hiện lỗi, nên trình bày như trong văn bản trước: đóng thẳng cột từng cặp thẻ, xuống dòng khi cần thiết, đặt các thẻ vào nơi hợp lý nhất.

#### **1.4.5. Các phần tử HTML (HTML element)**

*Một tài liệu HTML tạo nên từ nhiều thành phần HTML. Một thành phần HTML được đánh dấu bằng một cặp thẻ mở và thẻ đóng.*

*Các thành phần HTML có thể cấu trúc phân cấp hình cây, thành phần "mẹ" chứa nhiều thành phần "con" khác lồng bên trong nó.*

*Có thành phần rỗng, chỉ có thẻ mở.*

*Để soạn thảo tài liệu HTML đúng cú pháp, cần nắm vững cấu trúc của từng thành phần HTML.*

*Không giống như khi lập trình, nếu bạn mắc lỗi cú pháp HTML sẽ không có một thông báo lỗi nào báo cho biết mà trình duyệt sẽ hiểu lầm và trình bày trang Web không như bạn muốn mà thôi.*

#### **Bài tập**

1. Nêu các đặc điểm của siêu văn bản (HTML)
2. Thế nào là trang Web?

## CHƯƠNG II. TRÌNH BÀY TRANG

### II.1. Tạo tiêu đề

Mở đầu các trang văn bản thường là các tiêu đề cần làm nổi bật từng phần của văn bản như Chương, Mục,... cũng cần có đề mục rõ ràng khác với phần thân để người đọc theo dõi cho thuận tiện.

Có 6 mức tiêu đề trong HTML. Cách thể hiện các tiêu đề phụ thuộc vào trình duyệt nhưng thông thường thì:

- Tiêu đề mức 1

Thẻ định nghĩa có dạng: `<H1>...</H1>`

Ví dụ:

`<H1>Tiêu đề 1</H1>` cho ta tiêu đề tương ứng **Tiêu đề 1**

- Tiêu đề mức 2

Thẻ định nghĩa có dạng: `<H2>...</H2>`

Ví dụ:

`<H2>Tiêu đề 2</H2>` cho ta tiêu đề tương ứng **Tiêu đề 2**

- Tiêu đề mức 3

Thẻ định nghĩa có dạng: `<H3>...</H3>`

Ví dụ:

`<H3>Tiêu đề 3</H3>` cho ta tiêu đề tương ứng **Tiêu đề 3**

- Tiêu đề mức 4

Thẻ định nghĩa có dạng: `<H4>...</H4>`

Ví dụ:

`<H4>Tiêu đề 4</H4>` cho ta tiêu đề tương ứng **Tiêu đề 4**

- Tiêu đề mức 5

Thẻ định nghĩa có dạng: <H5>...</H5>

Ví dụ:

<H5>Tiêu đề 5</H5> cho ta tiêu đề tương ứng **Tiêu đề 5**

- Tiêu đề mức 6

Thẻ định nghĩa có dạng: <H6>...</H6>

Ví dụ:

<H6>Tiêu đề 6</H6> cho ta tiêu đề tương ứng **Tiêu đề 6**

## **II.2. Thẻ trình bày trang**

### **II.2.1. Một số thẻ chính**

Các thành phần trình bày trang để định dạng cả một đoạn văn bản và phải nằm trong phần thân của tài liệu. Có nhiều thẻ được sử dụng nhưng trong tài liệu này chỉ trình bày một số thẻ trình bày chính: định dạng phần địa chỉ (<ADDRESS>), đoạn văn bản (<P>), xuống dòng (<BR>), căn chính giữa (<CENTER>), đường kẻ ngang (<HR>), đoạn văn bản đã định dạng sẵn (<PRE>), trích dẫn nguồn tài liệu (<BLOCKQUOTE>)

#### **a. Định dạng phần địa chỉ**

*Cho biết thông tin như địa chỉ, danh thiếp và tác giả, thường đặt ở đầu hay cuối tài liệu.*

Thẻ định dạng: <ADDRESS>...</ADDRESS>

Ví dụ:

```
<ADDRESS>
Newsletter editor <BR>
J.R. Brown<BR>
JimquickPost Neus, Jumquick, CT 01234<BR>
Tel (123) 456 7890
</ADDRESS>
```

Kết quả thu được:

*Newsletter editor  
J.R. Brown  
JimquickPost News, Jumquick, CT 01234  
Tel (123) 456 7890*

## **b. Đoạn văn bản**

Thẻ này dùng để xác định một đoạn văn bản. Thẻ <P> (Paragraph) có thẻ dùng kèm thuộc tính để ấn định cách trình bày đoạn văn bản.

Chỉ là giới hạn một đoạn paragraph, cách bày trí do các thành phần khác tạo thành. Thường chỉ có khoảng trống khoảng một dòng hay nửa dòng trước paragraph, trừ khi nằm trong phần địa chỉ <ADDRESS>...</ADDRESS>. Một số trình duyệt thể hiện dòng đầu của paragraph thụt vào.

Thẻ định nghĩa dạng: <P>...</P>  
<P align="left|center|right">...</P>

Một đoạn văn bản rỗng là một dòng trắng.

Vì đầu dòng CR (Carriage Return) không có hiệu lực xuống dòng mới mà chỉ có tác dụng như một dấu cách, do đó có thể tạo một dòng dẫn cách (một dòng trắng) giữa các đoạn văn bản cần phải sử dụng thẻ <P>. Trường hợp này chỉ cần dùng thẻ <P> đơn lẻ, không cần thẻ đóng.

Chú ý: một số thẻ khác như các thẻ tiêu đề <H1>, . . . , <H6>, dòng kẻ ngang <HR>, danh sách, bảng biểu,... đã kèm luôn việc xuống dòng thành một đoạn văn bản mới. Không cần dùng thêm thẻ <P> trước và sau các thẻ này.

## **c. Xuống dòng**

*Thẻ này dùng để xuống dòng mới. Bắt buộc xuống dòng tại vị trí gặp từ khóa này. Dòng mới được căn lề như dòng được bẻ tự động khi dòng đó quá dài*

Thẻ định nghĩa dạng: <BR>

Nếu không muốn chèn một dòng trắng mà chỉ đơn thuần muốn xuống dòng mới thì cần sử dụng thẻ <BR> (Break). Thẻ Break không cần có thẻ đóng kèm theo.

#### **d. Đường kẻ ngang**

Thẻ này tạo ra đường kẻ ngang (Horizontal Rule) ngăn cách giữa các phần trong tài liệu.

Thẻ định nghĩa dạng: <HR>

<HR width=n%|n(pixel) size=n align=left|center|right noshade>

Ví dụ:

<HR>

<ADDRESS>February 8, 1995, CERN</ADDRESS>

Kết quả thu được:



#### **e. Căn chính giữa**

*Thẻ này dùng để căn chỉnh đoạn văn bản ở giữa chiều rộng trang văn bản.*

Thẻ định nghĩa dạng: <CENTER>...</CENTER>

Thẻ này cũng có tác dụng xuống dòng khi kết thúc đoạn văn bản.

#### **f. Đoạn văn bản đã định dạng sẵn**

Giới hạn đoạn văn bản đã được định dạng sẵn (pre-formatted) cần được thể hiện bằng phong chữ có độ rộng ký tự không đổi (do phải thẳng cột). Nếu không có thuộc tính WIDTH đi cùng thì bề rộng mặc định là 80 ký tự/dòng. Bề rộng 40,80 và 132 được thể hiện tối ưu, còn các bề rộng khác có thể được làm tròn.

Thẻ định nghĩa dạng: <PRE>...</PRE>

Trong các thành phần trước:

Dấu xuống dòng sẽ có ý nghĩa chuyển sang dòng mới (chứ không còn là dấu cách)

<P> không dùng. Nếu nó sẽ được coi như xuống dòng

Được phép dùng các thành phần liên kết nhấn mạnh

Không được chứa các thành phần định dạng paragraph (tiêu đề, địa chỉ,...)



Ký tự TAB (có mã US-ASCII và ISO-8859-1 là 9) phải hiểu là số dấu cách nhỏ nhất sao cho đến ký tự tiếp theo ở vị trí là bội của 8. Tuy nhiên không nên dùng.

Ví dụ:

```
<PRE WIDTH="80">
Nguyễn Văn AThọ rên
Trần Thị B   Thọ Sơn
</PRE>
```

Kết quả thu được:

Nguyễn Văn AThọ rên Trần Thị B   Thọ Sơn
---------------------------------------------

### **g. Trích dẫn nguồn tài liệu khác**

*Dùng để trích dẫn một đoạn văn bản, thường được thể hiện bằng chữ nghiêng có căn lề thụt vào trong (như một paragraph) và thường có một dòng trống trên và dưới.*

Thẻ định nghĩa dạng:

```
<BLOCKQUOTE>...</BLOCKQUOTE>
```

Ví dụ:

```
I think the poem ends
<BLOCKQUOTE>
<P>Soft you now, the fair Ophelia. Nymph, in thy orisons, be
all my sins remembred. </BLOCKQUOTE> but I am not sure.
```

Kết quả thu được:

I think the poem ends  Soft you now, the fair Ophelia. Nymph, in thy orisons, be all my sins remembred.  but I am not sure.
-----------------------------------------------------------------------------------------------------------------------------------------------

## II.2.2. Các thuộc tính của thẻ trình bày trang

### a. Thuộc tính ALIGN của thẻ Paragraph

Thẻ <P> dùng để xác định một đoạn văn bản (như trình bày ở phần trước). Dưới đây ta tìm hiểu kỹ thêm một số các thuộc tính kèm theo (ALIGN) của nó. Có thể căn lề trái (left - mặc định), căn giữa (center) hoặc căn lề phải (right).

- Căn lề trái:      <P ALIGN=LEFT>...</P>

Cả đoạn văn bản được căn lề trái của trang.

Ví dụ:

```
<P ALIGN=LEFT>  
Có lần tôi thấy một người đi  
Chẳng biết về đâu nghĩ ngợi gì  
Chân bước hững hờ theo bóng lẻ  
Một mình làm cả cuộc phân ly  
</P>
```

Kết quả trả về một đoạn văn bản được căn lề bên trái như sau:

Có lần tôi thấy một người đi  
Chẳng biết về đâu nghĩ ngợi gì  
Chân bước hững hờ theo bóng lẻ  
Một mình làm cả cuộc phân ly

- Căn giữa:                      <P ALIGN=CENTER>...</P>

Cả đoạn văn bản được căn chính giữa trang.

Ví dụ:

```
<P ALIGN=CENTER>  
Có lần tôi thấy một người đi  
Chẳng biết về đâu nghĩ ngợi gì  
Chân bước hững hờ theo bóng lẻ  
Một mình làm cả cuộc phân ly  
</P>
```

Kết quả trả về một đoạn văn bản được căn giữa như sau:

Có lần tôi thấy một người đi  
Chẳng biết về đâu nghĩ ngợi gì  
Chân bước hững hờ theo bóng lẻ  
Một mình làm cả cuộc phân ly

- **Căn lề phải:** <P ALIGN=RIGHT>...</P>

Cả đoạn văn bản được căn lề bên phải của trang.

Ví dụ:

```
<P ALIGN=RIGHT>
Có lần tôi thấy một người đi
Chẳng biết về đâu nghĩ ngợi gì
Chân bước hững hờ theo bóng lẽ
Một mình làm cả cuộc phân ly
</P>
```

Kết quả trả về một đoạn văn bản được căn lề bên phải như sau:

Có lần tôi thấy một người đi Chẳng biết về đâu nghĩ ngợi gì Chân bước hững hờ theo bóng lẽ Một mình làm cả cuộc phân ly
----------------------------------------------------------------------------------------------------------------------------------

## **b. Thuộc tính Clear của thẻ xuống dòng**

Thẻ xuống dòng <BR> cũng có 3 thuộc tính kèm theo như sau:

```
<BR CLEAR=ALL>
<BR CLEAR=LEFT>
<BR CLEAR=RIGHT>
```

Các thẻ được sử dụng khi chèn hình ảnh, để các dòng chữ xuất hiện bên dưới, bên trái hay bên phải của hình.

## **c. Các kiểu đường kẻ ngang khác nhau**

Như ở phần trên đã giới thiệu, thẻ <HR> tạo một đường kẻ ngang chạy suốt chiều rộng của sổ màn hình. Các đường kẻ này có thể được thay đổi độ đậm (mảnh), ngắn, dài, căn lề trái, căn lề phải,... bằng cách sử dụng các thuộc tính của chúng.

```
<HR WIDTH=n% SIZE=n ALIGN=LEFT|RIGHT NOSHADE>
```

Trong đó:

<b>Thẻ, thuộc tính</b>	<b>Miêu tả</b>
<HR>	Chèn dòng kẻ ngang suốt chiều rộng cửa sổ màn hình
WIDTH = n%	Thay đổi độ dài của đường kẻ, chiếm n% độ rộng cửa sổ màn hình. Nếu không có % đằng sau thì độ dài tính theo đơn vị pixel
SIZE = n	Thay đổi độ đậm hay mảnh của đường kẻ. n là số pixel
ALIGN=LEFT RIGHT	Căn lề trái phải. Đường kẻ ngang mặc định được căn chính giữa
NOSHADE	Không có bóng mờ, đường kẻ thành màu đen

### **Bài tập**

1. Tiêu đề trong trang Web có mấy mức chính?
2. Nêu các thuộc tính của thẻ Paragraph (<P>). Hãy tạo một trang web đơn giản trong đó có sử dụng tiêu đề mức 2 và thẻ Paragraph (<P>).
3. Tạo một trang web đơn giản tự giới thiệu về mình, bạn bao nhiêu tuổi, bạn làm gì, sở thích của bạn.

## CHƯƠNG III. DANH SÁCH VÀ BẢNG TRONG HTML

### III.1. Các kiểu danh sách

Việc liệt kê danh sách sẽ làm cho văn bản trở nên sáng sủa và dễ theo dõi. Trong HTML có một số kiểu danh sách như danh sách đánh số thứ tự, danh sách không đánh số thứ tự, danh sách định nghĩa, danh sách kiểu bảng chọn và danh sách kiểu thư mục.

Một số trình duyệt không hỗ trợ danh sách kiểu bảng chọn và danh sách kiểu thư mục.

#### III.1.1. Danh sách không đánh số thứ tự

Danh sách không đánh số thứ tự liệt kê các mục bằng một chấm tròn ở đầu mỗi mục.

Cú pháp:

```
<UL>
<LI> Mục thứ 1
<LI> Mục thứ 2
.....
</UL>
```

Cặp thẻ `<UL>...</UL>` đánh dấu bắt đầu và kết thúc danh sách. Nó có tác dụng lùi lề trái danh sách vào sâu hơn.

Các thẻ `<LI>` có tác dụng thể hiện bắt đầu một dòng mới với một chấm tròn.

Ví dụ:

```
<UL>
<LI> Mục thứ 1
<LI> Mục thứ 2
<LI> Mục thứ 3
</UL>
```

Kết quả trả về một danh sách sau:

- Mục thứ 1
- Mục thứ 2
- Mục thứ 3

Theo mặc định, mỗi mục thuộc mức thứ nhất của danh sách không đánh số thứ tự bắt đầu bằng một chấm tròn, còn mỗi mục thuộc mức thứ hai sẽ bắt đầu bằng một vòng tròn.

Ta vẫn có thể thay đổi bằng cách sử dụng thuộc tính TYPE

Cú pháp:

```
<UL TYPE=DISC | CIRCLE | SQUARE>  
hoặc
```

```
<UI TYPE=DISC | CIRCLE | SQUARE>
```

Ví dụ:

```
<UL>  
<LI TYPE=CIRCLE> Mục thứ 1  
<LI TYPE=CIRCLE> Mục thứ 2  
<LI TYPE=CIRCLE> Mục thứ 3  
</UL>
```

Kết quả trả về một danh sách sau:

- |                                                                                                     |
|-----------------------------------------------------------------------------------------------------|
| <ul style="list-style-type: none"><li>○ Mục thứ 1</li><li>○ Mục thứ 2</li><li>○ Mục thứ 3</li></ul> |
|-----------------------------------------------------------------------------------------------------|

### **III.1.2. Danh sách đánh số thứ tự**

Danh sách đánh số thứ tự giống như danh sách không đánh số thứ tự nhưng các chấm tròn ở đầu mỗi mục sẽ được thay bởi việc đánh số thứ tự các mục đó.

Cú pháp:

```
<OL>  
<LI> Mục thứ 1  
<LI> Mục thứ 2  
.....  
</OL>
```

Cặp thẻ <OL>...</OL> đánh dấu bắt đầu và kết thúc danh sách. Nó có tác dụng lùi lề trái danh sách vào sâu hơn.

Các thẻ <LI> có tác dụng thể hiện bắt đầu một dòng mới với một số thứ tự tương ứng.

Ví dụ:

```
<OL>
<LI> Mục thứ 1
<LI> Mục thứ 2
<LI> Mục thứ 3
</OL>
```

Kết quả trả về một danh sách sau:

- |                                                                                                        |
|--------------------------------------------------------------------------------------------------------|
| <ol style="list-style-type: none"><li>1. Mục thứ 1</li><li>2. Mục thứ 2</li><li>3. Mục thứ 3</li></ol> |
|--------------------------------------------------------------------------------------------------------|

Có thể dùng thuộc tính TYPE để lựa chọn các cách đánh thứ tự khác nhau khi bắt đầu mỗi mục trong một danh sách có đánh số thứ tự:

Chữ in hoa A,B,C,... (TYPE=A)

Chữ in thường a,b,c,... (TYPE=a)

Chữ số La mã lớn I,II,III,... (TYPE=I)

Chữ số La mã nhỏ i, ii, iii,... (TYPE=i)

Số thứ tự 1,2,3,4,... (TYPE=1), đây là lựa chọn mặc định

Cú Pháp:

```
<OL TYPE=A|a|I|i|1>
```

Hoặc

```
<LI TYPE=A|a|I|i|1>
```

Ví dụ:

```
<UL>
<LI TYPE=I> Mục thứ 1
<LI TYPE=I> Mục thứ 2
<LI TYPE=I> Mục thứ 3
</UL>
```

Kết quả trả về một danh sách sau:

- |                                                 |
|-------------------------------------------------|
| I. Mục thứ 1<br>II. Mục thứ 2<br>III. Mục thứ 3 |
|-------------------------------------------------|

### **III.1.3. Danh sách các định nghĩa**

Danh sách các định nghĩa trông giống như một bảng từ vựng hay bảng giải thích các thuật ngữ lùi vào trong.

Cú pháp:

```
<OL TYPE=A|a|I|i|1>
```

hoặc

```
<LI TYPE=A|a|I|i|1>
```

Ví dụ:

```
<DL>
```

```
<DT> Mục thứ 1
```

```
<DD> Nội dung mục thứ 1
```

```
<DT> Mục thứ 2
```

```
<DD> Nội dung mục thứ 2
```

```
<DT> Mục thứ 3
```

```
<DD> Nội dung mục thứ 3
```

```
</DL>
```

Kết quả trả về một danh sách sau:

- |                                 |
|---------------------------------|
| Mục thứ 1<br>Nội dung mục thứ 1 |
| Mục thứ 2<br>Nội dung mục thứ 2 |
| Mục thứ 3<br>Nội dung mục thứ 3 |

### **III.1.4. Danh sách phối hợp, lồng nhau**

Các kiểu danh sách nêu trên đều có thể lồng nhau và lồng nhau nhiều mức.

Ví dụ:



```

<UL>
<LI>Mức 1
<UL>
<LI>Mức 1.1
<LI>Mức 1.2
</UL>
<LI>Mức 2
<UL>
<LI>Mức 2.1
<LI>Mức 2.2
</UL>
</UL>

```

Kết quả trả về:

- |                                                                                                                                                                                                                                            |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <ul style="list-style-type: none"> <li>• Mức 1 <ul style="list-style-type: none"> <li>• Mức 1.1</li> <li>• Mức 1.2</li> </ul> </li> <li>• Mức 2 <ul style="list-style-type: none"> <li>• Mức 2.1</li> <li>• Mức 2.2</li> </ul> </li> </ul> |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

## III.2. Bảng biểu

### III.2.1. Khung cấu trúc

Giới hạn bảng: `<TABLE>...</TABLE>`

Định nghĩa một hàng: `<TR>...</TR>`

Định nghĩa một ô: `<TD>...</TD>`

Ô chứa tiêu đề: `<TH>...</TH>`. (H=Header)

Đầu đề của bảng: `<CAPTION>...</CAPTION>`

Cú pháp tạo khung cấu trúc của một bảng:

```

<TABLE>
<TR> <TH> Tên cột 1 </TH>
      <TH> Tên cột 2 </TH>
      ...
</TR>
<TR>
      <TD> Ô 1 dòng 1 </TD>

```

```

        <TD> Ô 2 dòng 1 </TD>
        .....
    </TR>
    .....
</TABLE>

```

**Ví dụ:**

```

<TABLE>
  <TR>
    <TH> Họ và tên </TH>
    <TH> Lớp </TH>
    <TH> Quê quán </TH>
  </TR>
  <TR>
    <TD> Bùi Thanh Trúc </TD>
    <TD> K50CA </TD>
    <TD> Hà Nội </TD>
  </TR>
  <TR>
    <TD> Nguyễn Mai Hương </TD>
    <TD> K50CB </TD>
    <TD> Thanh Hóa </TD>
  </TR>
</TABLE>

```

Kết quả thu được bảng như sau:

Họ và tên	Lớp	Quê quán
Bùi Thanh Trúc	K50CA	Hà Nội
Nguyễn Mai Hương	K50CB	Thanh Hóa

### **III.2.2. Một số lưu ý về bảng**

Ô rỗng thì không có đường bao. Muốn ô rỗng có đường bao phải dùng dấu cách không bẻ dòng:

```

<TD>&nbsp;</TD> hoặc <TD><BR></TD>

```

Có thể lợi dụng ROWSPAN và COLSPAN để tạo bảng có ô chồng chéo lên nhau nhưng không nên lạm dụng.

Đôi lúc hình ảnh nằm ngoài ô ta muốn đưa hình ảnh đó vào. Nguyên nhân là do HTML được viết như sau:

```
<TD>  
<IMG SRC="url">  
</TD>
```

Dấu xuống dòng ở đây coi như dấu cách do đó có sự chòng chéo hình ảnh với ký tự trắng đó. Cách khắc phục như sau:

```
<TD><IMG SRC="url"></TD>
```

## <TABLE>...</TABLE>

Mặc định bảng không có đường bao nếu không đi với thuộc tính BORDER. Bảng có các thuộc tính sau:

### **BORDER**

Đề định nghĩa một bảng có đường bao. Tuy nhiên không làm ảnh hưởng tới độ rộng của bảng.

**BORDER**=<giá trị>

Cho phép khai báo độ dày của đường bao cho phép chỉnh đường bao ngoài đậm hơn đường bao trong để dễ nhìn hơn, mặt khác nếu giá trị bằng không lại cho phép dành chỗ cho số liệu bên trong, nhất là đối với những bảng compact.

CELLSPACING=<giá trị>

Khoảng cách giữa các ô.

CELLPADDING=<giá trị>

Khoảng cách giữa nội dung của ô và đường bao.

```
<TABLE BORDER=0 CELLSPACING=0 CELLPADDING=0>
```

Cho ta bảng compact nhất có thể có (dành tất cả cho nội dung).

**WIDTH**=<giá trị hay phần trăm>

Nếu đi cùng với <TABLE> thuộc tính này có ý nghĩa mô tả chiều rộng mong muốn của bảng (tính tuyệt đối bằng pixels hay theo phần trăm so với bề rộng của tài liệu). Thường thì trình duyệt tự tính toán sao cho bảng được bố trí hợp lý. Dùng

thuộc tính này bắt buộc trình duyệt cố gắng làm sao xếp được các ô vào bảng có bề rộng mong muốn đó.

Nếu đi cùng với <TH> hay <TD> lại có ý nghĩa tương tự đối với một ô.

## **ROWSPAN**

Cho biết ô hiện thời chiếm mấy hàng của bảng, mặc định là 1.

## **ALIGN**

Nếu nằm trong <CAPTION> nó có ý nghĩa đối với đầu đề của hàng nằm trên hay nằm dưới, có giá trị hoặc TOP hoặc BOTTOM (mặc định là TOP).

Nếu nằm trong <TR>, <TH> hay <TD> có giá trị LEFT, CENTER hay RIGHT và điều khiển việc đặt nội dung của ô căn bên trái, vào giữa ô hay căn bên phải.

## **VALIGN**

Nếu nằm trong <TR>, <TH> hay <TD> có thể có các giá trị TOP, MIDDLE, BOTTOM hay BASELINE để điều khiển việc đặt nội dung của ô lên trên, vào giữa (theo chiều dọc) hay xuống dưới và cũng có thể là tất cả cùng các ô trong hàng cùng căn theo một đường nằm ngang.

## **<TD ...>...</TD>**

Có nghĩa là dữ liệu bảng (Table Data), chỉ được xuất hiện trong một hàng của bảng. Mỗi hàng không nhất thiết phải có cùng số ô vì dòng ngắn hơn sẽ được chấp thêm ô rỗng vào bên phải. Mỗi ô chỉ được chứa các thành phần bình thường khác nằm trong phần thân của tài liệu. Các thuộc tính mặc định là: ALIGN=left và VALIGN=middle. Các mặc định này có thể thay đổi bởi các thuộc tính trong <TR> và lại thay đổi tiếp bởi thuộc tính ALIGN hoặc VALIGN khai báo riêng cho từng ô. Bình thường mặc định thì nội dung sẽ được bẻ dòng cho vừa vào khổ rộng của từng ô. Dùng thuộc tính NOWRAP trong <TD> để cấm việc đó.

<TD...>...</TD> cũng có thể chứa các thuộc tính NOWRAP, COLSPAN và ROWSPAN.

## **NOWRAP**

Như nói ở trên, cần thận trọng khi dùng thuộc tính này để phòng có ô quá rộng.

## COLSPAN

Có thể xuất hiện trong bất kỳ ô nào (<TH> hay <TD>) và chỉ rằng ô đó bành trướng ra mấy cột của bảng, mặc định là 1.

**<TH ...> ... </TH>**

Có nghĩa là tiêu đề của bảng (Table Header), các ô này tương tự như ô bình thường khác được định nghĩa bằng <TD>, có điều font chữ đậm và có thuộc tính mặc định là ALIGN=Center. <TH ...>...</TH> cũng có thể chứa thuộc tính VALIGN, NOWRAP, COLSPAN và ROWSPAN.

**<CAPTION ...> ... </CAPTION>**

Đặt tiêu đề cho một bảng nên phải nằm trong một cặp <TABLE> song không được nằm trong hàng hay cột. Thuộc tính mặc định là ALIGN=Top (tiêu đề đặt ở đầu bảng), song có thể đặt là ALIGN=Bottom (cuối bảng). Tiêu đề có thể chứa bất kỳ thành phần nào một ô có thể chứa và luôn được căn lề vào giữa bảng (theo chiều ngang) và có thể cũng có bẻ dòng cho phù hợp.

Dưới đây là bảng tóm tắt liệt kê các thuộc tính của thẻ xây dựng bảng:

Thẻ/Thuộc tính	Ý nghĩa
<TABLE BORDER=n CELLSPACING=n CELLPADDING=n BGCOLOR="#rrggbb">	TABLE - Bắt đầu bảng BORDER - Đặt khung nổi 3D xung quanh bảng. Đặt BORDER=0 sẽ làm mất biên bao quanh. CELLSPACING - Đặt độ dày của dòng kẻ ngang trong bảng. CELLPADDING - Đặt kích thước của một ô. BGCOLOR - Đặt màu nền của bảng.
<TR>	B3/4t Đầu một dòng của bảng – Table row.
<TD>	B3/4t Đầu một ô của bảng (bắt đầu cột trong một bảng).
<TH>	Thẻ <TH> giống như <TD> nhưng cho chữ đậm và căn chính giữa - (heading).
<CAPTION>	Đặt tiêu đề (đầu đề) cho một bảng
ALIGN=MIDDLE   RIGHT	Thuộc tính dùng kèm bên trong các thẻ <TR>, <TH> hay <TD> để căn lề chữ trong một ô của bảng theo chiều ngang
VALIGN=TOP   BOTTOM	Thuộc tính dùng kèm các thẻ <TR>, <TH> hay <TD>

	để giãn một ô theo chiều ngang chiếm nhiều cột hơn các ô trên dòng khác.
WIDTH=n%	Thuộc tính dùng kèm các thẻ <TH> hay <TD> để đặt độ rộng của ô.
COLSPAN=n	Thuộc tính dùng kèm các thẻ <TH> hay <TD> để giãn một ô theo chiều ngang chiếm nhiều cột hơn các ô trên dòng khác.
ROWSPAN=n	Thuộc tính dùng kèm các thẻ <TH> hay <TD> để giãn một ô theo chiều đứng chiếm nhiều dòng hơn các ô ở cột khác
</TABLE>	Kết thúc bảng

### III.2.3. Các ví dụ

Ví dụ một bảng cơ bản gồm 2 dòng và 3 cột:

```
<TABLE BORDER>
<TR>
    <TD>A</TD><TD>B</TD><TD>C</TD>
</TR>
<TR>
    <TD>D</TD><TD>E</TD><TD>F</TD>
</TR>
</TABLE>
```

Kết quả trả về:

A	B	C
D	E	F

Ví dụ một bảng có Item 2 (tiêu thức 2) chiếm 2 hàng (dòng):

```
<TABLE BORDER>
<TR>
    <TD>Item 1</TD>
    <TD ROWSPAN=2>Item 2</TD>
    <TD>Item 3</TD>
</TR>
<TR>
    <TD>Item 4</TD> <TD>Item 5</TD>
</TR>
```

</TABLE>

Kết quả trả về:

Item 1	Item 2	Item 3
Item 4		Item 5

Ví dụ một bảng có Item 1 (tiêu thức 2) chiếm 2 hàng (dòng):

```
<TABLE BORDER>
<TR>
    <TD ROWSPAN=2>Item 1</TD>
    <TD>Item 2</TD> <TD>Item 3</TD> <TD>Item 4</TD>
</TR>
<TR>
    <TD>Item 5</TD> <TD>Item 6</TD> <TD>Item 7</TD>
</TR>
</TABLE>
```

Kết quả trả về:

Item 1	Item 2	Item 3	Item 4
	Item 5	Item 6	Item 7

Ví dụ bảng có Item 2 (tiêu thức 2) chiếm 2 cột:

```
<TABLE BORDER>
<TR>
    <TD>Item 1</TD>
    <TD COLSPAN=2>Item 2</TD>
</TR>
<TR>
    <TD>Item 3</TD> <TD>Item 4</TD> <TD>Item 5</TD>
</TR>
</TABLE>
```

Kết quả trả về:

Item 1	Item 2	
Item 3	Item 4	Item 5

Ví dụ một bảng có tiêu đề:

```
<TABLE BORDER>
  <TR>
    <TH>Head 1</TH> <TH>Head 2</TH> <TH>Head 3</TH>
  </TR>
  <TR>
    <TD>D</TD> <TD>E</TD> <TD>F</TD>
  </TR>
</TABLE>
```

Kết quả trả về:

Head 1	Head 2	Head 3
D	E	F

Ví dụ một bảng có tiêu đề chiếm 2 cột:

```
<TABLE BORDER>
  <TR>
    <TH COLSPAN=2>Head 1</TH>
    <TH COLSPAN=2>Head 2</TH>
  </TR>
  <TR>
    <TD>A</TD> <TD>B</TD> <TD>C</TD> <TD>D</TD>
  </TR>
  <TR>
    <TD>E</TD> <TD>F</TD> <TD>G</TD> <TD>H</TD>
  </TR>
</TABLE>
```

Kết quả trả về:

Head 1		Head 2	
A	B	C	D
E	F	G	H



## Bài tập

1. Sửa phần giới thiệu của bạn (bài tập 3, chương 2) sao cho các mục đó đưa về dạng danh sách và liệt kê 5 thứ bạn thích nhất dưới dạng số.
2. Áp dụng những gì đã học ở trên, hãy tạo một danh sách như sau:
  - Top level 1
  - Top level 2
    1. Second Level
      - Fact 1
      - Fact 2
  - Top level 3
3. Áp dụng những gì đã học ở trên, hãy tạo bảng giống như sau:

<b>Tổng kết năm học 2003 - 2004</b>			
<b>Thống kê</b>	<b>HS giỏi</b>	<b>HS khá</b>	<b>HS trung bình</b>
Lớp 1A	35	4	0
Lớp 1B	20	10	12
Lớp 1C	5	25	20

## CHƯƠNG IV. ĐƯA HÌNH ẢNH VÀO TÀI LIỆU HTML

### IV.1. Hình ảnh tĩnh

#### IV.1.1. Tập ảnh

Mỗi ảnh được lưu giữ trong máy tính thành một tệp riêng biệt. Tệp ảnh có nhiều định dạng khác nhau, thể hiện qua phần mở rộng (đuôi) của tên tệp: \*.bmp, \*.gif, \*.jpeg, \*.jpg,...

Tuy nhiên các trình duyệt chỉ có thể “hiểu” được các tệp ảnh dưới dạng GIF hay JPEG. Hình ảnh dạng GIF được sử dụng rộng rãi nhất trên Internet, sau đó là đến các ảnh dạng JPEG.

#### IV.1.2. Thẻ <IMG...>

Cú pháp chèn ảnh vào trang Web:

```
<IMG SRC="URL">
```

IMG (Image), thuộc tính SRC (Source) là đường dẫn đến nơi lấy tệp ảnh. Giá trị của thuộc tính SRC được gán bằng URL là một địa chỉ trên máy tính địa phương hay trên Internet. Nó chỉ ra nơi lưu trữ tệp ảnh cần chèn vào.

Địa chỉ URL báo cho trình duyệt biết cần lấy tệp ảnh ở đâu.

Nếu tệp ảnh nằm ngay trên máy tính địa phương thì cần ghi rõ đường dẫn

Nếu tệp ảnh lấy từ một nơi khác trên Internet thì phải ghi đầy đủ địa chỉ URL

Ví dụ:

```
<IMG SRC="logo.gif">
```

### IV.2. Các thuộc tính của thẻ chèn hình ảnh

Trong thẻ IMG còn có một số thuộc tính khác như: ALT, WIDTH, HEIGHT, ALIGN, VSPACE, HSPACE, BORDER

#### IV.2.1. Thuộc tính ALT

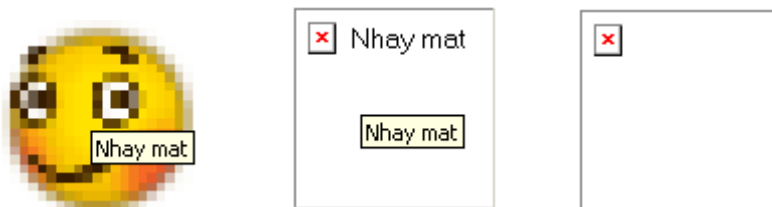
Thuộc tính ALT – ALternative cho phép ta chèn một đoạn chữ thay thế vào chỗ có hình và hiện lên tiêu chú thích khi đưa trỏ chuột vào ảnh. Thuộc tính này dùng trong trường hợp trình duyệt không hiển thị được hình ảnh thì sẽ hiện dòng văn bản thay thế. Làm như vậy để những người sử dụng trình duyệt không có khả

năng đọc ảnh hay đã tắt chức năng đọc ảnh để tăng tốc độ có thể biết được đó là hình ảnh gì và họ có thể chọn xem sau nếu có thời gian.

Nên luôn sử dụng thuộc tính này khi đưa hình ảnh vào trang Web vì nhiều người muốn có tốc độ cao, lướt nhanh qua các thông tin là chính đã tắt chức năng đọc ảnh của trình duyệt vẫn có thể biết được ảnh đó mang nội dung gì.

Ví dụ:

```
<IMG SRC="face.gif" ALT="Nhảy mát">
```



Hiện ảnh      Ảnh (không hiện lên) khi có và không sử dụng thuộc tính ALT

Nếu không sử dụng thuộc tính ALT thì tại chỗ có hình sẽ hiện từ IMAGE hoặc biểu tượng ảnh bị khuyết.

#### **IV.2.2. Thuộc tính WIDTH và HEIGHT**

Thuộc tính WIDTH và HEIGHT dùng để xác định chiều rộng và chiều cao của ảnh. Giá trị này có thể tính theo phần trăm (%) hoặc pixel.

Chú ý:

Nếu đặt chiều rộng và chiều cao không chuẩn sẽ làm hình trong ảnh bị co giãn méo đi. Để khắc phục nhược điểm này, ta nên xử lý ảnh để kích thước phù hợp với yêu cầu.

Ví dụ:

```
<IMB SRC="face.gif" WIDTH=100 HEIGHT=100>
```



```
<IMB SRC="face.gif" WIDTH=50 HEIGHT=50>
```





### IV.2.3. Thuộc tính ALIGN

Thuộc tính ALIGN cho phép chỉnh lại vị trí của ảnh theo cả hai chiều ngang và chiều dọc.

Theo chiều dọc: ALIGN=MIDDLE, ALIGN=TOP, ALIGN=BOTTOM

Ben trai  Ben phải	Ben trai  Ben phải	Ben trai  Ben phải
ALIGN=TOP	ALIGN=MIDDLE	ALIGN=BOTTOM

Theo chiều ngang: ALIGN=LEFT, ALIGN=RIGHT

Day la hình nhảy mát 	Day la hình nhảy mát 
ALIGN=LEFT	ALIGN=RIGHT

Nếu như không có thuộc tính này thì mặc định ảnh được đặt tại đúng nơi đặt thẻ IMG đóng trên cùng dòng chữ, theo cạnh dưới của khung nhìn.

### IV.2.4. Thuộc tính VSPACE và HSPACE

Khi sử dụng thuộc tính ALIGN, ảnh được chèn vào dòng văn bản với các chữ dính sát liền. Thuộc tính VSPACE và HSPACE dùng để tạo khoảng trống nhỏ viền xung quanh ảnh (tính theo đơn vị pixel).

VSPACE=n                      Thêm khoảng trống theo chiều dọc

HSPACE=n                      Thêm khoảng trống theo chiều ngang

Ví dụ:

```
<IMG SEC="face.gif" VSPACE=20 HSPACE=20>
```



## Bài tập

1. Hãy cho biết tại sao các ảnh dưới đây có thể được nhúng vào trong một tài liệu HTML? Viết đoạn code được dùng để nhúng các ảnh đó.



2. Tạo một trang web và chèn các ảnh sao cho ảnh nằm ở phía bên trái, và các đoạn văn bản nằm bên phải.
3. Tạo một trang web và chèn các ảnh sao cho ảnh được căn giữa và các đoạn văn bản nằm ở trên hoặc ở bên dưới.

## CHƯƠNG V. CÁC MỐI LIÊN KẾT SIÊU VĂN BẢN

### V.1. Thẻ neo và mối liên kết

Các liên kết siêu văn bản trong một tài liệu HTML là để tham chiếu đến một tài liệu khác hay một phần tài liệu khác nằm trong tài liệu đó.

Liên kết siêu văn bản gồm có ba phần: giao thức, cái neo và địa chỉ hay URL

Để tạo một mối liên kết thì việc đầu tiên cần làm là thiết lập cái neo. Cặp thẻ tạo neo là <A>...</A> (Anchor). Thẻ này có nhiều thuộc tính bắt buộc phải xác định rõ, ta sẽ tìm hiểu kỹ hơn ở phần sau.

#### V.1.1. Thuộc tính HREF

HREF (Hypertext REFerence – tham chiếu siêu văn bản) được dùng để liên kết đến:

Một tài liệu khác (external link) hay

Một phần khác nằm trong chính tài liệu đang đọc (internal link)

Giao thức để tham chiếu HREF là HTTP. Nếu là liên kết nội tại – internal link thì không cần phải có phần giao thức.

#### V.1.2. Liên kết ra ngoài – External Links

Để liên kết đến một tài liệu khác, ta cần phải biết địa chỉ URL của tài liệu đích. Cũng cần phải chỉ chỗ, thường là một nhóm vài từ để làm đầu mối liên kết. Nhóm từ này sẽ đổi màu để phân biệt, con chuột trở vào sẽ có hình bàn tay và khi nhấn chuột thì trang siêu văn bản đích sẽ được hiển thị.

Nên chọn các từ làm đầu mối sao cho gợi tả và sao cho tự nhiên, không ảnh hưởng đến nội dung và ý nghĩa của câu văn. Tránh sử dụng lặp đi lặp lại câu "nhấn vào đây" khi tạo đầu mối liên kết.

Cú pháp để tạo ra một mối liên kết tới tài liệu khác - liên kết ra ngoài - là như sau:

```
<A HREF="URL"> Nhóm từ đầu mối liên kết</A>
```

### **V.1.3. Địa chỉ tuyệt đối**

Khi liên kết tới một tệp nằm ở ngoài máy tính cục bộ ta phải nêu rõ giao thức và đầy đủ địa chỉ URL của tệp đích.

```
<A HREF="full URL">... </A>
```

Ví dụ:

```
<A HREF="http://www.vnu.edu.vn/index.html">ĐHQG Hà Nội</A>
```

```
<A HREF="http://www.srl.mit.edu.au/sa/vicsa.html">Victorian  
Subject Associations on Internet</A>
```

### **V.1.4. Địa chỉ tương đối**

Nếu liên kết đến một tệp ở ngay trên máy chủ của bạn thì không cần dùng địa chỉ URL tuyệt đối mà dùng địa chỉ tương đối. Thay cho URL là tên tệp cùng với đường dẫn đến thư mục nơi chứa tệp đích.

```
<A HREF="path & filename">... </A>
```

Ví dụ:

```
<A HREF="examples/exam1.htm">Ví dụ 1</A>
```

### **V.1.5. Liên kết nội tại – Internal Link**

Cũng có thể tạo mối liên kết trong bản thân tài liệu siêu văn bản, từ phần này đến phần khác. Điều này rất có ích khi tài liệu có kích thước lớn. Ta có thể tạo mục lục toàn bộ nội dung trên phần đầu trang gồm tên các chương và liên kết từng tên chương đến phần nội dung tương ứng.

Để làm điều này, cần đánh dấu điểm đích - book mark- của liên kết bằng thẻ neo

```
<A NAME="Tên điểm đích">
```

còn trong thẻ neo tại điểm đầu của mỗi liên kết thì thay URL bằng tên nói trên.

```
<A HREF="Tên điểm đích">
```

Lưu ý có thêm dấu "#".

Ví dụ, để tạo liên kết nội tại từ đây nhảy về đầu trang ta cần tạo một book mark ở đầu trang với tên là **Top** chẳng hạn. Sau đó, tạo thẻ neo liên kết ở dòng dưới như sau

<A HREF="#Top"> về đầu trang </A>

## V.2. Dùng hình ảnh làm đầu mỗi liên kết

### V.2.1. Thay chữ bằng hình

Như ở phần trên, cấu trúc thành phần HTML mỗi liên kết siêu văn bản là:

```
<A HREF="Đầu mỗi liên kết">
```

Đề thay “đầu mỗi liên kết” bởi một biểu tượng, hay một hình ảnh... Ta chỉ cần chèn một hình ảnh hay biểu tượng đó vào vị trí của “đầu mỗi liên kết”.

Ví dụ:

```
<A HREF="http://www.vnu.edu.vn"><IMG SRC="Logo.gif"></A>
```

### V.2.2. Image Map - thẻ AREA

Thử hình dung ta có một bản đồ địa lý hành chính. Khi nhấn chuột vào vùng diện tích của một tỉnh thì một trang thông tin tương ứng với tỉnh đó sẽ hiện ra. Đó chính là Image Map.

Với Image Map, “đầu mỗi liên kết” là một vùng trên một hình ảnh có hình dạng tùy ý nào đó.

Các công cụ soạn thảo trực quan cho phép dễ dàng tạo Image Map.

Một ví dụ về mã nguồn HTML của thành phần Image Map như sau:

```
<MAP NAME="Map01">
<AREA HREF="chapter1.html" SHAPE="rect" coords="6, 22, 167,
169">
<AREA HREF="chapter2.html" SHAPE="circle" coords="259, 106,
36">
<AREA HREF="chapter3.html" SHAPE="polygon" coords="123, 196,
56, 211, 278, 201">
</MAP>
<IMG BORDER="0" SRC="image.gif" WIDTH="400" HEIGHT="600"
USEMAP="#Map01">
```

Đoạn mã trên sẽ tạo 3 đầu mỗi liên kết là ba vùng hình chữ nhật (rect), hình tròn (circle) và hình tam giác (polygon) trên nền ảnh image.gif. Khi nhấn chuột vào phạm vi 3 vùng này thì các tệp tương ứng chapter1.html, chapter2.html và chapter3.html sẽ được nạp về để hiển thị.



### V.3. Đưa âm thanh vào tài liệu

#### V.3.1. Liên kết đến tệp âm thanh

Để chèn một đoạn âm thanh vào tài liệu HTML ta cũng theo đúng quy tắc tạo mỗi liên kết thông thường. Trong thẻ neo, tại địa điểm URL bây giờ là địa chỉ của tệp âm thanh.

Dưới đây là một ví dụ về việc chèn âm thanh vào tài liệu. Khi nhấn chuột vào mỗi liên kết này, tệp âm thanh sẽ được phát lại.

```
<A HREF="music/papa.rmi">Papa</A>
```

#### V.3.2. Tạo âm thanh nền

Không những có thể chèn tệp âm thanh vào tài liệu HTML mà còn có thể nhúng âm thanh vào tài liệu, nghĩa là âm thanh nền phát ra khi trang tài liệu bắt đầu hiển thị.

Để nhúng âm thanh vào tài liệu ta dùng thẻ

```
<EMBED SRC="URL" HEIGHT="n" WIDTH="n"
AUTOSTART=true|false LOOP=true|false>
```

Trong đó:

Thuộc tính	Mô tả
SRC="URL"	Đường dẫn đến tệp âm thanh
HEIGHT=n	Chiều cao của khung nhìn cho giao diện điều khiển của phần mềm phát âm thanh
WIDTH=n	Chiều rộng của khung nhìn cho giao diện điều khiển của phần mềm phát âm thanh
AUTOSTART=true false	Âm thanh tự động được kích hoạt khi bắt đầu nạp tài liệu hiển thị (nên đặt là true)
LOOP=true false	Thiết lập việc lặp lại nhiều lần hay chỉ một lần

Đặt thẻ <EMBED> ở đâu thì khung nhìn của giao diện điều khiển việc phát tệp âm thanh xuất hiện ở đó.

Ví dụ:

```
<EMBED SRC="music/papa.rmi" HEIGHT=50 WIDTH=200
AUTOSTART=true LOOP=true>
```

## V.4. Đưa Video vào tài liệu

### V.4.1. Chèn tệp Video

Việc chèn tệp video vào tài liệu HTML cũng giống như chèn tệp âm thanh.

Ví dụ đoạn mã sau đây sẽ chèn một đoạn phim video vào tài liệu. Nó sẽ tạo ra một mối liên kết tới tệp video đã chọn và khi ta nhấn chuột vào mối liên kết thì sẽ được xem lại tệp video đó.

```
<A HREF="film.mpg">Film</A>
```

Chú ý: Các tệp video đều có kích thước khá lớn nên hãy cân nhắc về tốc độ đường truyền của người dùng.

### V.4.2. Nhúng tệp video

Thẻ <EMBED> cũng dùng để nhúng tệp video vào tài liệu:

Ví dụ:

```
<EMBED SRC="film/film.mov" HEIGHT=144 WIDTH=160>
```

Cũng tương tự như các thuộc tính của việc chèn tệp âm thanh:

Thuộc tính	Mô tả
SRC="URL"	Tên và đường dẫn đến tệp video
HEIGHT=n	Chiều cao của khung hình chiếu video
WIDTH=n	Chiều rộng của khung hình chiếu video
AUTOPLAY=true false	Đặt cho video tự chạy khi bắt đầu nạp tài liệu (không bắt buộc)
CONTROLLER=true false	Cho hiện giao diện điều khiển của phần mềm video ở liền ngay phía dưới khung hình. Thuộc tính này không bắt buộc. Mặc định là true.
LOOP=true false	Thiết lập việc lặp lại nhiều lần hay chỉ một lần

Vị trí đặt thẻ <EMBED> là nơi xuất hiện khung hình video.

## **Bài tập**

1. Tạo một trang khác giới thiệu chi tiết hơn về mình và có liên kết tới trang giới thiệu chính của mình.
2. Chèn 2 ảnh vào trong trang giới thiệu chính sao cho khi click vào ảnh đó thì trang giới thiệu chi tiết được mở ra. Yêu cầu: ảnh thứ nhất sử dụng đường dẫn tuyệt đối, ảnh thứ hai sử dụng đường dẫn tương đối tới trang cần liên kết.
3. Tạo một trang thứ ba trong đó có chèn một đoạn Video clips ở giữa trang.

## CHƯƠNG VI. BÀY TRÍ NỀN VÀ KHUNG

### VI.1. Màu nền và văn bản

#### VI.1.1. Đặt màu nền

Để văn bản trở nên đẹp và hấp dẫn, đôi khi đặt nền cho trang siêu văn bản.

Thuộc tính BGCOLOR (Background Color). Dùng thuộc tính này kèm thẻ <BODY> để đặt màu nền cho văn bản. Cú pháp như sau:

```
<BODY BGCOLOR="#rrggbb">  
Nội dung của tài liệu  
</BODY>
```

trong đó "#rrggbb" là *red-green-blue*, bộ ba số hai chữ số hệ đếm 16, xác định mã màu.

#### VI.1.2. Màu chữ của văn bản

Thuộc tính TEXT. Thuộc tính này để thiết lập màu cho các con chữ trong văn bản, trừ các đầu mối liên kết phải có màu khác đi.

```
<BODY TEXT="#rrggbb">  
Nội dung của tài liệu  
</BODY>
```

#### VI.1.3. Màu của đầu mối liên kết - Thuộc tính LINK, VLINK và ALINK

Ba thuộc tính trên để đặt màu của các đầu mối siêu liên kết.

- LINK - đặt màu hiển thị trước khi nhấn chuột vào để đến thăm đích liên kết.
- VLINK - Đặt màu sau khi đích liên kết đã được đến thăm (visited).
- ALINK - đặt màu khi bạn kích hoạt, đang nhấn chuột vào (active).

Các màu mặc định là: LINK=blue, VLINK=purple and ALINK=red

```
<BODY LINK="#rrggbb" VLINK="#rrggbb" ALINK="#rrggbb">  
Nội dung của tài liệu  
</BODY>
```

Kết hợp cùng với nhau

```

<HTML>
<HEAD>
  <TITLE>Color          Control          Example</TITLE>
</HEAD>
<BODY  BGCOLOR="#000000"  TEXT="#F0F0F0"  LINK="#FFFF00"
VLINK="#22AA22"  ALINK="#0077FF">
  Đây là một tài liệu làm ví dụ về thiết đặt màu. Các
chữ màu xám sáng trên nền đen, và <A
HREF="nowhere.htm">đầu mỗi liên kết</A> đầu tiên có màu
vàng, chuyển màu xanh lá cây sáng khi được kích hoạt và
có màu xanh nhạt khi điểm đích đã được thăm
  </BODY>
</HTML>

```

Lưu ý: Tổ hợp màu như trên chỉ được thiết đặt một lần cho toàn văn bản. Hãy thận trọng khi lựa chọn để đảm bảo dễ đọc, dễ nhìn.

#### **VI.1.4. Thuộc tính và mã màu**

```

<BODY  BGCOLOR="#rrggbb"  TEXT="#rrggbb"  LINK="#rrggbb"
VLINK="#rrggbb"  ALINK="#rrggbb">

```

<b>Thuộc tính</b>	<b>Mô tả</b>
BGCOLOR	Đặt màu nền
TEXT	Đặt màu các con chữ, trừ các mối nối.
LINK	Đặt màu ban đầu của đầu mỗi liên kết khi chưa kích hoạt
VLINK	Đặt màu đầu mỗi liên kết khi đã thăm đích
ALINK	Đặt màu đầu mỗi liên kết khi kích hoạt

Một vài mã màu hay dùng:

<b>Màu</b>	<b>Mã</b>
black	#000000
white	#FFFFFF
pale grey	#DBDDE5
dark green	#306020
green	#405060
pale blue/green	#00FFFF
pale blue	#A5D6F7
navy blue	#000020

burgundy	#800040
yellow	#FFFF00
purple	#601050
red	#FF0000
orange red	#FF2400
orange	#FF7F00
neon blue	#4D4DFE
pale purple/orchid	#9370DB
maroon	#8E236B
scarlet	#8C1717
pale purple	#6B238E
firebirch	#8E2323

Tuy nhiên hiện nay nhiều trình duyệt chấp nhận dùng tên màu tiếng Anh thay cho các chữ số rất khó nhớ ở trên.

## VI.2. Nạp hình ảnh làm nền cho trang văn bản

### VI.2.1. Thuộc tính **BACKGROUND**

Có thể dùng thuộc tính này để tạo một ảnh nền cho trang tài liệu siêu văn bản. Thay cho xác định màu ta cần chỉ ra tên tệp hình ảnh kèm đường dẫn.

```
<BODY BACKGROUND ="đường dẫn tới tệp ảnh">
Nội dung tài liệu
</BODY>
```

Ví dụ:

```
<BODY BACKGROUND="images/backgnd.gif">
```

Khi dùng ảnh làm nền cho trang Web cần phải nạp một tệp hình ảnh. Như ta đã biết tệp ảnh thường có kích thước lớn, làm chậm việc hiển thị trang tài liệu. Do đó cần chọn tệp ảnh có kích thước nhỏ làm ảnh nền. Toàn bộ trang văn bản sẽ được "lát nền" bằng hình ảnh này như ta lát nền nhà bằng gạch hoa.

### VI.2.2. **Water mark**

Nhiều trang web có nền trang trí gắn chặt cố định, còn phần văn bản sẽ cuộn trôi bên trên mỗi khi ta di chuyển thanh trượt. Hiệu ứng này được tạo ra nhờ thiết lập thêm thuộc tính cho ảnh nền là

bgproperties="fixed"

Ví dụ:

```
<BODY BACKGROUND="images/backgnd.gif"
bgproperties="fixed">
```

### **VI.2.3. Hãy ký tên vào tài liệu của mình**

Một thông lệ nên tuân theo là "kí tên" vào tài liệu. Nó giúp cho người đọc biết được những thông tin tối thiểu về tác giả soạn ra tài liệu, thời gian cập nhật... Việc đưa thêm địa chỉ của tài liệu Web vào cuối trang sẽ giúp cho người đọc lưu lại được xuất xứ của trang tài liệu. Đó là chưa nói đến ý nghĩa quan trọng của phần chữ kí này trong các tài liệu chính thức hoặc có tính thương mại.

Phần chữ kí thường gồm các thông tin sau:

- Ngày khởi tạo lần đầu.
- Ngày sửa chữa cập nhật gần nhất.
- Tên (và e-mail) của tác giả
- Tuyên bố về bản quyền (nếu cần )
- URL

Đoạn mã HTML của phần chữ kí đại loại có thể như sau:

```
<HR SIZE=1>
<FONT SIZE=-1>
  Ngày viết: 20 September 1998 <BR>
  Ngày cập nhật: 20 October 2001<BR>
  Tác giả: <A HREF="mailto:webmaster@vitti.vnu.edu.vn">
  Nguyễn Văn Hùng</A> email:
  webmaster@vitti.vnu.edu.vn"><BR>
<P>
  Copyright &copy; Viện Đào tạo công nghệ thông tin -
  VITTI, ĐHQG Hà nội, 1998.
<HR SIZE=1>
  URL: http://www.vnu.edu.vn/index.htm
</FONT>
```

Thẻ `<A HREF="mailto:webmaster@vitti.vnu.edu.vn">...</A>` là mối liên kết đến dịch vụ thư điện tử, khi trỏ chuột vào đây sẽ kích hoạt dịch vụ e-mail để gửi đến địa chỉ nêu sau lệnh **mailto**.

Hãy tập thói quen thêm chữ kí vào tài liệu của mình!

### VI.3. Khung – Frames

HTML có các thẻ trình bày cho phép chia vùng hiển thị của cửa sổ trình duyệt thành nhiều khung, mỗi khung là một cửa sổ độc lập, hiển thị một tài liệu HTML khác nhau.

Khung cho phép người thiết kế hiển thị đồng bộ nhiều tài liệu HTML khác nhau để tiện theo dõi, so sánh. Ví dụ, trong khung bên trái hiển thị các nút bấm, còn khung bên phải hiển thị tài liệu tương ứng.

#### VI.3.1. Trang trí khung

Trang HTML thực hiện bày trí các khung (gọi là *frameset document*) có cấu trúc khác trang thông thường, không có khung.

Trang thường có 2 phần, HEAD và BODY. Trang bày trí khung có HEAD và FRAMESET thay cho BODY.

Thành phần FRAMESET tổ chức các khung trong cửa sổ trình duyệt. Nó cũng có thể chứa thẻ NOFRAMES để xử lí trường hợp trình duyệt không hỗ trợ frame.

Các thành phần thông thường khác vốn nằm trong BODY không được xuất hiện trước thẻ mở FRAMESET. Nếu không, thành phần FRAMESET sẽ bị bỏ qua.

Ví dụ:

Dưới đây là một ví dụ đơn giản.

```
<HTML>
<HEAD>
<TITLE>A simple frameset document</TITLE>
</HEAD>
<FRAMESET cols="20%, 80%">
  <FRAMESET rows="100, 200">
    <FRAME src="frame1.html">
    <FRAME src="frame2.gif">
```

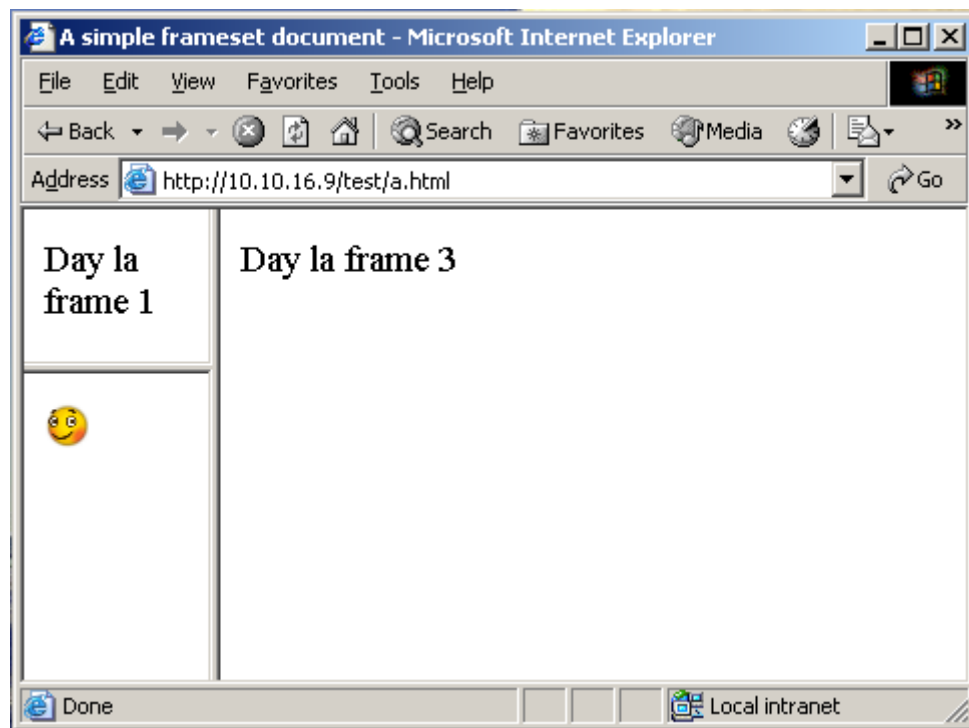


```

</FRAMESET>
  <FRAME src="frame3.html">
  <NOFRAMES>
    <P>This frameset document contains:
    . . . . .
  </NOFRAMES>
</FRAMESET>
</HTML>

```

Đoạn mã trên sẽ tạo 3 khung, được bài trí như dưới đây.



*Hình VI.1. Kết quả chạy đoạn code ví dụ*

Khi trình duyệt khách không hỗ trợ khung thì các khung sẽ không được hiển thị mà thành phần NOFRAMES sẽ được xử lí.

### **VI.3.2. Thành phần FRAMESET**

Thẻ FRAMESET dùng để phân chia vùng hiển thị trong cửa sổ trình duyệt thành các khung hình chữ nhật. Mỗi khung hình chữ nhật gọi là một frame, được định nghĩa bằng thẻ FRAME .

#### **a. Các thuộc tính và ví dụ minh họa**

*rows* = Danh sách các độ cao của các khung

Danh sách gồm nhiều phần tử, cách nhau dấu phẩy. Mỗi phần tử xác định độ cao (số dòng !) của một khung. Chia chiều đứng thành bao nhiêu khung thì danh sách có bấy nhiêu phần tử.

Chiều cao thể hiện bằng

- số pixel,
- tỷ lệ phần trăm chiều cao màn hình
- hay tỷ lệ phần chiều cao còn lại.

Giá trị mặc định là 100%, tức chỉ có một khung theo chiều ngang.

*cols* = Danh sách các độ rộng của các khung.

ý nghĩa tương tự như trên.

Giá trị mặc định là 100%, tức chỉ có một khung theo chiều dọc.

Thuộc tính *row* thiết lập việc chia khung theo chiều ngang trong một frameset. Nếu không định nghĩa, thì các cột trong khung sẽ chiếm toàn bộ chiều cao vùng hiển thị.

Thuộc tính *cols* thiết lập việc chia khung theo chiều đứng trong một frameset. Nếu không định nghĩa, thì các dòng trong khung sẽ chiếm toàn bộ chiều rộng vùng hiển thị.

Phối hợp hai thuộc tính sẽ tạo ra ô lưới các khung.

Các ví dụ.

1- Chia màn hình thành hai nửa: nửa trên và nửa dưới:

```
<FRAMESET rows="50%, 50%">  
...the rest of the definition...  
</FRAMESET>
```

2- Chia màn hình thành 3 cột. Cột giữa rộng 250 pixels. Cột đầu chiếm 25% của phần còn lại và cột thứ 3 chiếm 75% của phần độ rộng còn lại.

```
<FRAMESET cols="1*, 250, 3*">  
...the rest of the definition...  
</FRAMESET>
```

3- Tạo lưới gồm  $2 \times 3 = 6$  khung.

```
<FRAMESET rows="30%, 70%" cols="33%, 34%, 33%">  
...the rest of the definition...
```

```
</FRAMESET>
```

4- Chia chiều đứng màn hình thành 4 khung. Khung thứ nhất chiếm 30% của chiều cao vùng hiển thị. Khung thứ hai có chiều cao cố định 400 pixel. Dấu sao có nghĩa là hai khung thứ 3, thứ 4 chia nhau phần còn lại. Khung thứ 4 có chiều cao là "2\*", gấp đôi khung thứ 3 (vì "\*" tương đương với 1\*).

Nếu chiều cao vùng hiển thị là 1000 pixel thì độ cao của các khung 1,2,3,4 lần lượt là: 300, 400, 100, 200 pixel !.

```
<FRAMESET rows="30%, 400, *, 2*">  
...the rest of the definition...  
</FRAMESET>
```

## **b. Chia khung lồng nhau và thành phần FRAME**

Việc chia khung có thể lồng nhau nhiều mức.

Ví dụ: chia chiều rộng thành 3 khung đứng, sau đó khung ở giữa lại được chia thành 2 phần trên và dưới.

```
<FRAMESET cols="33%, 33%, 34%">  
...contents of first frame...  
  <FRAMESET rows="40%, 50%">  
    ...contents of second frame, first row...  
    ...contents of second frame, second row...  
  </FRAMESET>  
...contents of third frame...  
</FRAMESET>
```

Thẻ FRAME định nghĩa một khung hình cụ thể (trong nhiều khung hình của frameset).

### **Các thuộc tính:**

***name*** = Tên của khung.

Có thể dùng tên này để làm đích của mỗi siêu liên kết.

***src*** = URI

Trở đến trang tài liệu sẽ hiển thị trong khung.

### *noresize*

Không cho phép co giãn lại kích thước

*scrolling* = auto | yes | no

Thiết lập thanh cuộn.

- auto: Xuất hiện thanh cuộn khi cần thiết. Đây là giá trị mặc định.
- yes: Luôn có thanh cuộn.
- no: Luôn không có thanh cuộn.

*frameborder* = 1 | 0

Thiết lập đường biên.

- 1: Có đường biên giữa khung đang xét với các khung kề nó. Đây là giá trị mặc định.
- 0: Không có đường biên giữa khung đang xét với các khung kề nó.

*marginwidth* = số pixel

Thiết lập độ rộng lề chiều rộng = khoảng trống giữa phần hiển thị nội dung và biên trái, biên phải. Giá trị mặc định tùy theo bộ duyệt.

*marginheight* = số pixel

Thiết lập độ rộng lề chiều cao = khoảng trống giữa phần hiển thị nội dung và biên trên, biên dưới. Giá trị mặc định tùy theo trình duyệt.

Lưu ý: Nội dung trong một frame không được thuộc về chính trang tài liệu định nghĩa frameset.

## **VI.4. Thiết lập Target, thẻ NOFRAME và IFRAME**

### **VI.4.1. Thiết lập Target**

Thuộc tính target là để xác định tệp tài liệu HTML sẽ hiển thị trong khung.

*target* = tên khung đích.

Thiết lập tên của khung mà tài liệu sẽ mở ra trong khung đó.

Thuộc tính này dùng với các thành phần tạo mối liên kết: (A, LINK), image map (AREA), và FORM.

## VI.4.2. Thẻ NOFRAMES

Thành phần NOFRAMES thiết lập nội dung cần hiển thị khi trình khách không hỗ trợ frame hoặc đã tắt chức năng hiển thị frame.

Thành phần NOFRAMES đặt ở phần cuối của thành phần FRAMESET.

Ví dụ:

```
<HTML>
<HEAD>
<TITLE>A frameset document with NOFRAMES</TITLE>
</HEAD>
<FRAMESET cols="50%, 50%">
  <FRAME src="main.html">
  <FRAME src="table_of_contents.html">
  <NOFRAMES>
  <P>Here is the <A href="main-noframes.html">
    non-frame based version of the document.</A>
  </NOFRAMES>
</FRAMESET>
</HTML>
```

## VI.4.2. Nhúng frame - thẻ IFRAME

Thành phần IFRAME cho phép người thiết kế chèn một frame vào giữa một khối văn bản text và hiển thị một tài liệu HTML khác bên trong.

Thuộc tính SRC thiết lập tài liệu nguồn để hiển thị trong frame.

### Các thuộc tính:

**name** = tên. để tham chiếu trong tài liệu

**width** = Độ rộng của inline frame.

**height** = Độ cao của inline frame.

Ví dụ:

```
<IFRAME src="foo.html" width="400" height="500"
  scrolling="auto" frameborder="1">
[Your user agent does not support frames or is
currently configured
not to display frames. However, you may visit
<A href="foo.html">the related document.</A>]
</IFRAME>
```

Inline frames mặc định là không co giãn được, không cần phải nêu rõ noresize.

### **Bài tập**

1. Sửa trang giới thiệu tóm tắt về mình sao cho nền của trang đó là một ảnh bất kỳ, trang giới thiệu chi tiết có nền màu xanh nước biển nhạt.
2. Dùng thẻ IFRAME để nhúng trang thứ 3 vào trong trang giới thiệu tóm tắt (trang chủ).

## CHƯƠNG VII. BIỂU MẪU STYLE VÀ CASCADING STYLE SHEET

### VII.1. FORM

#### VII.1.1. FORM là gì?

##### a. Chức năng của FORM

Để mở rộng khả năng phục vụ người sử dụng, dịch vụ Web cần phải sử dụng các ứng dụng khác bên ngoài. Ví dụ NSD yêu cầu tìm kiếm trong cơ sở dữ liệu, lấy các thông tin tức thời, luôn được cập nhật... Để làm được điều này dịch vụ Web phải chuyển yêu cầu của NSD đến một ứng dụng khác. Ứng dụng này sẽ thực hiện yêu cầu và trả lại kết quả cho Web server để chuyển tiếp đến NSD.

Form là một cách để chuyển dữ liệu từ NSD đến cho Web Server xử lý. Forms được sử dụng rộng rãi trên WWW. Các forms rất tiện lợi cho người dùng điền các yêu cầu tìm kiếm, các biểu mẫu điều tra, nhập dữ liệu đầu vào cho các ứng dụng..

Có nhiều thành phần khác nhau (gọi là các điều khiển *control*) trong một form. Tùy theo yêu cầu giao tiếp với NSD cần chọn thành phần thích hợp nhất.

Khi tạo form bạn cũng cần phải chỉ rõ cho máy chủ dịch vụ biết cách xử lý form. Có nhiều loại chương trình ứng dụng khác nhau trong máy chủ dịch vụ để làm việc này: các chương trình CGI, ISAPI, các script ASP, JSP, Java Bean, Servlet...

Tạo form là khâu đầu tiên trong việc xây dựng giao tiếp giữa NSD với các ứng dụng Internet / Intranet.

##### b. Thành phần của FORM

Cặp thẻ để tạo Form là `<FORM...>...</FORM>`.

Mọi thành phần của form như sẽ trình bày dưới đây đều phải nằm trong phạm vi giới hạn bởi cặp thẻ này.

Công thức khung để tạo form là:

```
<FORM METHOD=POST ACTION="URL">
```

```
Toàn bộ các thành phần khác bên trong form nằm ở đây  
</FORM>
```

Thuộc tính METHOD chỉ ra phương thức trao đổi dữ liệu giữa trình duyệt Web và máy chủ Web. Có 2 phương thức là POST, GET. Đối với form phương thức thường là POST.

Thuộc tính ACTION để thiết lập địa chỉ URL của chương trình sẽ nhận và xử lý dữ liệu gửi từ form.

### **VII.1.2.Các thành phần trong FORM**

Như đã nêu trên, có nhiều loại thành phần khác nhau trong một Form. Dưới đây sẽ trình bày lần lượt những điểm cơ bản nhất.

#### **a. Hộp văn bản – TextBox**

Hộp văn bản là nơi để gõ vào một dữ liệu kiểu xâu kí tự. Hộp văn bản sẽ được hiển thị như sau:

Bottom of Form 1

Thẻ để tạo ra nó là:

```
<INPUT TYPE="TEXT" NAME="Tên" SIZE="n">
```

Trong đó TEXT là từ khoá ứng với kiểu là Text box, Tên là tên của hộp văn bản này còn n là một số nguyên chỉ ra chiều dài của trường. Tên phải duy nhất trong trang, không được trùng nhau.

#### **b. Hộp mật khẩu – Password**

Hộp mật khẩu là nơi để gõ vào một mật khẩu kiểu xâu kí tự. Hộp mật khẩu được hiển thị như sau:

Thẻ để tạo ra nó là:

```
<INPUT TYPE="PASSWORD" SIZE="n">
```

Trong đó PASSWORD là từ khoá để tạo hộp mật khẩu, n là số nguyên chỉ ra chiều dài của hộp.

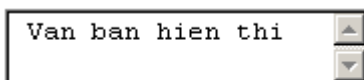
Hộp mật khẩu khác hộp văn bản ở chỗ khi gõ các kí tự từ bàn phím thì nó không hiển thị kí tự tương ứng trong khung mà thay bằng các dấu sao \*.



Lưu ý: Việc hiển thị kí tự dấu \* để thay thế chỉ có tác dụng che giấu với người nhìn tại máy cục bộ. Password vẫn được gửi vào mạng dưới dạng rõ, không mã hoá, có thể bị xem trộm. Nhiệm vụ mã là của giao thức mạng.

### c. Vùng văn bản – Text Windows

Vùng văn bản, để hiển thị nhiều dòng văn bản. Vùng văn bản có dạng như sau:



Công thức viết là:

```
<TEXTAREA NAME= OTenVung" ROWS=m COLS=n> Van ban hien  
thi</TEXTAREA>
```

trong đó, *Tên vùng* là tên của vùng văn bản, m và n là các số nguyên, chỉ ra chiều cao và chiều rộng của vùng văn bản. Nếu bạn để trống phần "đoạn văn bản sẽ hiển thị" thì sẽ không có văn bản nào xuất hiện trong cửa sổ.

Ví dụ:

```
<TEXTAREA NAME="COMMENTS" ROWS=5 COLS=50> Bạn cần phải  
thường xuyên cập nhật các mẫu virus mới nhất</TEXTAREA>
```

### d. Các lựa chọn – Radio Button

Các nút chọn radio là một nhóm nút tròn, chỉ cho phép bạn được chọn một trong nhiều nút. Nút radio có dạng như sau:

- Nhan nut 1
- Nhan nut 2
- Nhan nut 3

Công thức viết:

```
<INPUT TYPE="RADIO" NAME="Tên_nhóm_nút "  
VALUE="giá_trị_được_chọn"> Nhãn nút
```

Trong đó RADIO là từ khoá ứng với kiểu nút chọn Radio, *nhãn nút i* là xâu kí tự sẽ xuất hiện cạnh nút.

Ví dụ:

- Master Card
- Visa
- Bank Card

Đoạn mã để tạo nút radio đầu tiên - Mastercard - là:

```
<INPUT TYPE="RADIO" NAME="PAYMENT"
VALUE="Mastercard">Master Card
```

Trong ví dụ trên, thuộc tính NAME được gán trị PAYMENT là tên nhóm nút Radio, gồm 3 nút cho phép lựa chọn một trong 3 cách thanh toán, thuộc tính VALUE để nhận kết quả câu trả lời ứng với lựa chọn "Mastercard". Lưu ý phân biệt giữa Mastercard là giá trị của lựa chọn với Mastercard là nhãn nút, tức cụm chữ hiện cạnh nút radio. Ta có thể thay nhãn nút bằng một xâu bất kì, ví dụ "Thẻ tín dụng MasterCard".

Dĩ nhiên cứ một nút radio lại tương ứng với một dòng mã HTML như trên. Lưu ý rằng thuộc tính NAME phải như nhau cho toàn bộ các nút radio của cùng một nhóm. Trong ví dụ trên thì tất cả các nút đều phải cùng có NAME = [PAYMENT].

Trong một nhóm nút radio, một nút có thể có thuộc tính CHECKED, nghĩa là nó được đánh dấu chọn sẵn khi mở trang Web.

#### e. Ô đánh dấu – CheckBoxes

Ô đánh dấu, để đánh dấu một hoặc vài lựa chọn. Ô đánh dấu có dạng như sau:

- Nhan o 1
- Nhan o 2
- Nhan o 3

Công thức viết:

```
<INPUT TYPE="checkbox" NAME="Tên nhóm nút" VALUE="giá
trị được chọn"> Nhan o i
```

Ví dụ:

- Overnight
- Ship to avobe address
- Call before shipping

Khác với nút radio bạn có thể đánh dấu chọn nhiều ô đồng thời trong cùng một nhóm ô đánh dấu.

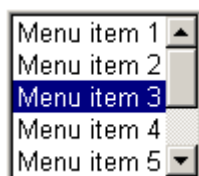
Ví dụ công thức để tạo ra checkbox đầu tiên ở trên là:

```
<INPUT TYPE="CHECKBOX" NAME="DELIVERY"
VALUE="Overnight">Overnight
```

 ý nghĩa của các thuộc tính cũng tương tự như của nút radio.

Trong một nhóm ô đánh dấu, một số ô có thể có thuộc tính CHECKED và được đánh dấu chọn sẵn khi mở trang Web.

## f. Danh sách lựa chọn – thành phần SELECT



Công thức để tạo menu là:

```
<SELECT NAME="MENU" SIZE="5">
<OPTION>Menu item 1
<OPTION>Menu item 2
<OPTION SELECTED>Menu item 3
<OPTION>Menu item 4
<OPTION>Menu item 5
<OPTION>Menu item 6
<OPTION>Menu item 7
<OPTION>Menu item 8
</SELECT>
```

trong đó thuộc tính SIZE xác định số mục chọn có thể nhìn thấy trong cửa sổ cuộn. OPTION SELECTED xác định lựa chọn mặc định ban đầu.

Có thể tạo menu buông xuống có dạng như sau nếu ta đặt thuộc tính SIZE=1:



Để tạo bảng chọn kiểu này chỉ cần bỏ thuộc tính SIZE trong đoạn mã trên.

Mặc định, danh sách chọn SELECT chỉ cho phép chọn một mục chọn. Thuộc tính MULTIPLE cho phép đánh dấu chọn nhiều mục.

## g. Hộp chọn tệp

Cho phép hiển thị hộp chọn tệp hoặc nhập tên tệp trực tiếp. Ví dụ.

Khi nhấn chuột vào nút Browse thì hộp thoại Choose File sẽ mở và cho phép chọn tệp.

Cách viết:

```
<INPUT TYPE="FILE" NAME="fileName">
```

Khi nhập tên tệp vào trường text hay chọn tệp (bằng nút Browse) thì tên tệp sẽ được gán cho thuộc tính VALUE của thành phần này.

#### **h. Nút Send và Clear**

Khi người sử dụng đã điền xong thì phải hoặc gửi kết quả đi, hoặc xoá sạch và điền lại từ đầu:

Send - gửi kết quả đi.

Clear - xoá sạch và làm lại từ đầu.

Hai nút để làm việc này có dạng:

Mã để tạo ra hai nút này là:

```
<INPUT TYPE="SUBMIT" VALUE="Send form">
```

```
<INPUT TYPE="RESET" VALUE="Clear form">
```

trong đó hai chuỗi ký tự trong ngoặc kép Send form, Clear form gán cho thuộc tính VALUE sẽ hiển thị trên nút tương ứng. Dĩ nhiên ta có thay bằng chuỗi nào khác tùy ý, chẳng hạn “gửi đi”, “điền lại”.

#### **i. Nút nhấn – Push Button**

Ngoài hai nút kiểu "Submit" và "Reset" luôn gắn với form và có chức năng quy định trước như trên, có thể tạo nút nhấn để gắn với những hành động xử lý khác do ta tự thiết kế. Mã nguồn để tạo nút nhấn tổng quát kiểu này là.

```
<Button type="button" name="Tên nút"> Nhấn nút  
</BUTTON>
```

Tên nút để tham chiếu, còn *nhãn nút* là nhãn sẽ hiển thị trên nút. Nếu muốn trang trí một biểu tượng hay hình ảnh trên nút chỉ cần thay thế phần *nhãn nút* bằng hình ảnh.

```
<BUTTON type="button" name="Tên nút"><IMG SRC="Tập ảnh" ALT="Dòng chữ thay thế"></BUTTON>
```

Lưu ý: phải cung cấp dòng chữ thay thế cho thành phần IMG. Không được gắn một *image map* với thành phần IMG chứa trong một thẻ BUTTON.

Để gắn các hàm chức năng xử lý với một nút nhấn, ta dùng các thuộc tính về sự kiện nội tại của nút như onclick, ondblclick, onmousedown, onmouseup, onmouseover, onmousemove, onmouseout... Phần trình bày về Scripting sẽ nêu rõ vấn đề này.

## j. Nút nhấn bằng hình ảnh

Có thể tạo nút nhấn trực tiếp bằng một hình ảnh.

```
<INPUT type="image" NAME = "Tên nút" SRC="Tên ảnh">
```



Khi nhấn nút, form được gửi đi cùng với các tọa độ x,y tính bằng pixel, kể từ góc trên - trái của hình ảnh. Các số liệu này được gửi đi dưới dạng.

*name.x=x-value name.y=y-value*

ở đây name là giá trị của thuộc tính name của nút; *x-value*, *y-value* là các tọa độ.

Nếu việc xử lý khác nhau phụ thuộc tọa độ của điểm nhấn chuột mà trình khách không hỗ trợ hình ảnh thì phải dùng giải pháp khác: dùng nhiều button hoặc *image map* và script.

## k. Các trường ẩn

Các trường ẩn không hiển thị trên form nhưng được dùng để gửi thông tin cho Server.

Ví dụ:

```
<INPUT TYPE="HIDDEN" NAME="previous-url" VALUE="URL">
```

Đoạn mã trên để tạo một liên kết trên trang cho phép NSD nhấn nút để tới một trang Web tại địa chỉ URL nhất định nào đó chứ không phải dùng phím BACK.

Một ví dụ nữa là để điền tự động mục Subject trong email có thể dùng một trường ẩn như sau:

```
<INPUT TYPE="HIDDEN" NAME="subject" VALUE="membership form">
```

Đoạn mã này điền xâu kí tự đã gán cho thuộc tính VALUE vào dòng chủ đề - subject trong email.

## 1. Nhãn

### Thẻ label

Các nút đã có nhãn ngầm định, gán bằng thuộc tính *value*

Các thành phần còn lại như *text fields*, *checkboxes*, *radio buttons*, *menus* không có nhãn ngầm định. Có thể hiển thị một xâu kí tự kề bên thích hợp để làm "nhãn" như ta vẫn làm ở trên.

Tuy nhiên, đặc tả HTML có thành phần LABEL để gán nhãn cho các thành phần điều khiển khác.

Cặp thẻ <LABEL>... </LABEL> dùng để định nghĩa thành phần LABEL.

### Thuộc tính FOR của thẻ label

```
<LABEL for="Id">
```

Thuộc tính này nhằm gán nhãn cho thành phần điều khiển trở bởi Id. Id là tên định danh của thành phần điều khiển.

Nếu không có thuộc tính for thì ngầm định là gán cho thành phần điều khiển chứa trong cặp thẻ LABEL đang xét.

Có thể nhiều hơn một LABEL cho cùng một thành phần điều khiển

Ví dụ:

```
<FORM action="..." method="post">
<TABLE>
  <TR>
    <TD><LABEL for="fname">First Name</LABEL>
    <TD><INPUT type="text" name="firstname" id="fname">
  <TR>
    <TD><LABEL for="lname">Last Name</LABEL>
    <TD><INPUT type="text" name="lastname" id="lname">
</TABLE>
```

</FORM>

### Gán nhãn không dùng thuộc tính FOR

Để gán nhãn cho một thành phần điều khiển mà không dùng thuộc tính FOR, thành phần điều khiển phải nằm bên trong thành phần LABEL. Lúc này, một thành phần LABEL chỉ chứa một thành phần điều khiển. Các chữ làm nhãn có thể đặt trước hay sau thành phần điều khiển.

Ví dụ: gán nhãn cho 2 thành phần text field.

```
<FORM action="..." method="post">
<P>
<LABEL>
  First Name
  <INPUT type="text" name="firstname">
</LABEL>
<LABEL>
  <INPUT type="text" name="lastname">
  Last Name
</LABEL>
</P>
</FORM>
```

## **VII.1.3. Thêm tính cấu trúc cho FORM**

### **a. FIELDSET và LEGEND**

Thành phần FIELDSET cho phép nhóm một số thành phần điều khiển có liên quan với nhau. NSD dễ hiểu hơn, còn người thiết kế trang cũng dễ xử lý di chuyển focus hơn.

Thành phần LEGEND cho phép gán một dòng chú giải cho một FIELDSET

#### Các thuộc tính

`align = top|bottom|left|right`

- `top`: legend đặt ở phía trên của fieldset. Đây là giá trị mặc định.
- `bottom`: legend đặt ở phía dưới của fieldset.
- `left`: legend đặt ở phía trái.
- `right`: legend đặt ở phía phải của fieldset.

#### Vi dụ

```
<FORM action="..." method="post">
<P>
<FIELDSET>
```

```

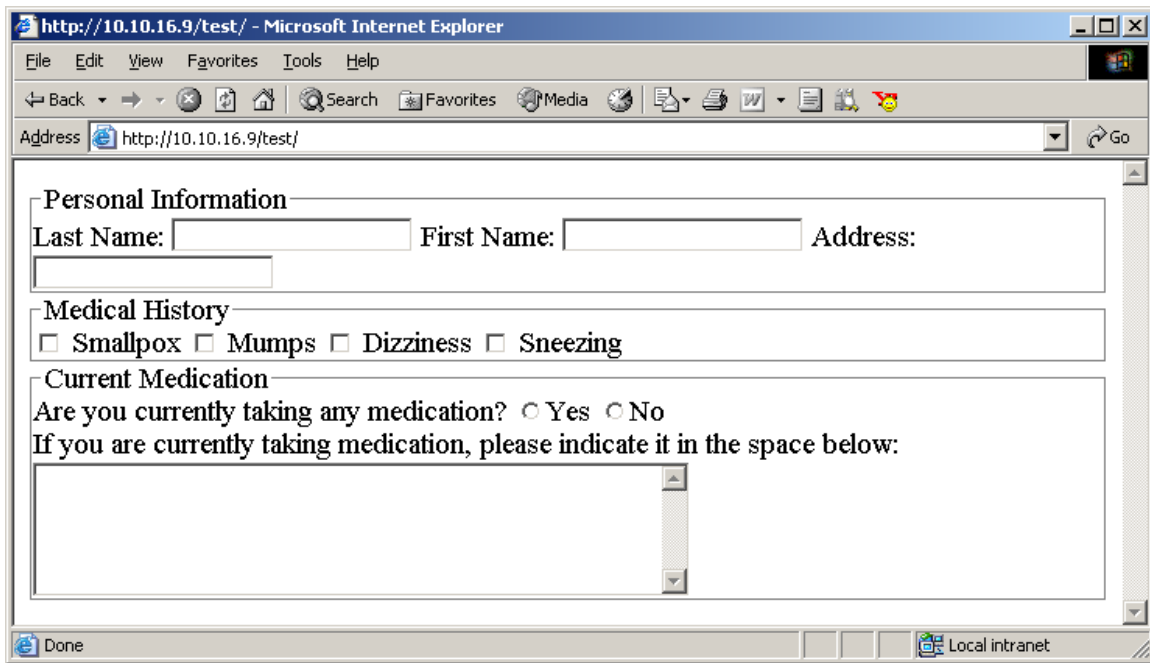
<LEGEND>Personal Information</LEGEND>
  Last Name: <INPUT name="personal_lastname"
type="text">
  First Name: <INPUT name="personal_firstname"
type="text">
  Address:   <INPUT name="personal_address"
type="text">
    </FIELDSET>
    <FIELDSET>
      <LEGEND>Medical History</LEGEND>
      <INPUT name="history_illness" type="checkbox"
value="Smallpox"> Smallpox
      <INPUT name="history_illness" type="checkbox"
value="Mumps"> Mumps
      <INPUT name="history_illness" type="checkbox"
value="Dizziness"> Dizziness
      <INPUT name="history_illness" type="checkbox"
value="Sneezing"> Sneezing
    </FIELDSET>

  <FIELDSET>
    <LEGEND>Current Medication</LEGEND>
    Are you currently taking any medication?
    <INPUT name="medication_now" type="radio"
value="Yes">Yes
    <INPUT name="medication_now" type="radio"
value="No">No
    <BR> If you are currently taking medication, please
indicate it in the space below:
    <TEXTAREA name="current_medication" rows="5"
cols="50">
    </TEXTAREA>
  </FIELDSET>
</FORM>

```

Trình duyệt sẽ hiển thị như sau:





Hình VII.1. Kết quả ví dụ

## VII.2. Cascading style sheet

Style Sheets là một đặc tính quan trọng mà có thể được dùng trong HTML động (Dynamic HTML). Mặc dù trang Web không thật sự cần phải có một style sheet, nhưng việc dùng nó có nhiều lợi ích. Trong phần này, ta sẽ làm quen về công nghệ style sheet dùng để nâng cao việc thiết kết và phát triển trang Web.

### VII.2.1. Inline Style

Một đặc tính quan trọng của Dynamic HTML là các style động. Ta có thể thay đổi style của nhiều phần tử HTML trên trang sau khi nó được hiển thị trong trang. Sự thay đổi có thể thể hiện bằng cách đáp lại tương tác của người dùng hoặc sự kiện thay đổi trạng thái (như sự kiện thay đổi kích thước).

Có hai cách thay đổi style của trang:

Thay đổi style bên trong

Viết các scripts để thay đổi style

Sử dụng các inline style ta có thể viết các style động mà không phải viết thêm script vào trang.

Một “inline style” là một style mà được gán vào một phần tử đặc biệt. Style không hỗ trợ cho tất cả các phần tử của một kiểu hoặc một lớp nào đó. Một inline style được định nghĩa dùng thuộc tính STYLE cho tag của các phần tử. Ví dụ, nếu ta

muốn gán màu của <H1> (Heading 1) thành màu đỏ, ta có thể gán thuộc tính STYLE của thẻ H1.

```
<H1 STYLE="color:red">
```

Nếu ta muốn dùng script để sửa một inline style bất kỳ lúc nào, khi đó ta phải dùng đối tượng style (Style Object). Style Object hỗ trợ mọi thuộc tính mà CSS hỗ trợ cho các style. Để dùng thuộc tính trong scrip:

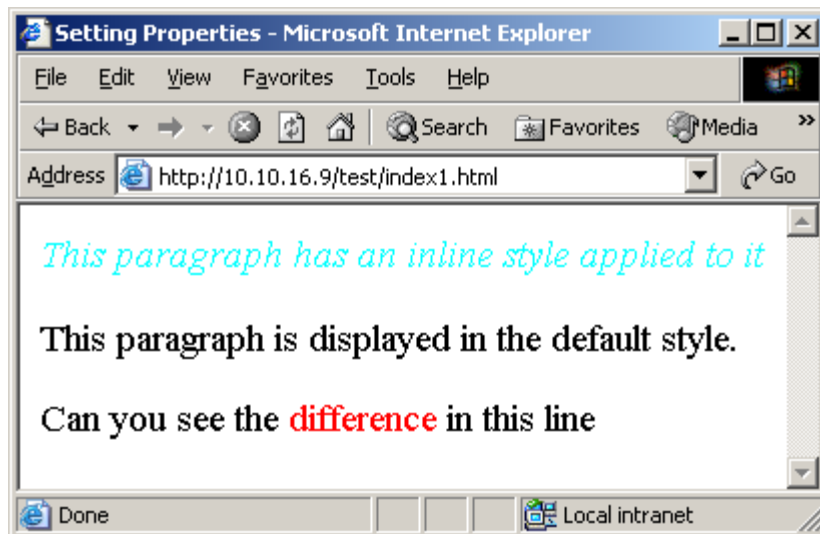
Loại bỏ gạch nối từ tên thuộc tính style CSS

Thay đổi chữ đầu tiên của từ sau gạch nối thành chữ viết hoa

Ví dụ, thuộc tính CSS font-weight thành fontWeight trong DHTML. Hoặc text-align thành textAlign.

Ta hãy xem một ví dụ để xóa mọi thứ. Ta sẽ dùng inline style để gán các thuộc tính của nhân tố <P>. Xem đoạn code dưới đây trong Internet Explorer:

```
<HTML>
<HEAD>
<TITLE>Setting Properties</TITLE>
</HEAD>
<BODY>
<P style = "color:aqua; font-Style:italic; text-
Align:center;"> This paragraph has an inline style
applied to it
<BR>
<P> This paragraph is displayed in the default style.
<BR>
<P>Can you see the <SPAN style =
color:red>difference</SPAN>in this line
</BODY>
<HTML>
```



Hình VII.2. Sử dụng Inline Style

## VII.2.2. Giới thiệu Style Sheet

Style sheet là nơi mà ta quản lý và điều khiển các style. Style sheet mô tả diện mạo và việc biểu diễn của một tài liệu HTML như nó sẽ được đưa ra trên màn hình, hoặc in ra ngoài. Trong thực tế, ta có thể nghĩ về nó như một khuôn mẫu để căn cứ vào các tài liệu HTML của ta. Đặc biệt ta cũng có thể định rõ vị trí và sự xuất hiện của các phần tử trên một trang và tạo các hiệu ứng đặc biệt. Ví dụ ta có thể tạo một style sheet định nghĩa cho thẻ <H1> là chữ béo và nghiêng và màu xanh. Hoặc thẻ phần tử <P> được đưa ra với màu đỏ với font Arial.

Ta có thể định nghĩa thông tin style mà ta muốn trong một vị trí – style sheet. Khi đó style sheet có thể được liên kết để tất cả các trang trong một Web site để tạo một diện mạo thích hợp, đồng nhất qua site bên trong. Ta có thể định nghĩa nhiều style sheet và liên kết chúng thành một tập như nhau của các trang Web để mềm dẻo hơn nhiều trong khi tạo các trang Web.

Một trang Web có thể không có style sheet, nếu không có style sheet được xác định, khi đó trình duyệt sẽ sử dụng cách hiển thị theo quy ước ngầm định của nó. Ví dụ, một thẻ <H1> luôn được hiển thị cùng một định dạng, các đoạn và các lề cũng như vậy. Dù sao việc dùng style sheet ta có thể xác định cách mà thẻ <H1> được hiển thị trong trình duyệt.

Về mặt lý thuyết, ta có thể dùng bất kỳ công nghệ style sheet nào với HTML. Dù sao thực tế thì chỉ công nghệ Cascading Style Sheet (CSS) được các trình duyệt

hỗ trợ tốt. CSS là một chuẩn được phát triển bởi World Wide Web Consortium (thường gọi là “W3C”) dùng trong các trình duyệt Web.

### **VII.2.3. Javascript Style Sheet**

Netscape hỗ trợ các cascading style sheet. Nó cũng hỗ trợ style sheet mà được viết trông giống JavaScript và dùng mô hình đối tượng tài liệu – Document Object Model (DOM). Khi ra định nghĩa một style sheet ta có thể khai báo kiểu như:

“text/CSS” - - Trong trường hợp này nó tìm cú pháp CSS.

“text/JavaScript” – Trong trường hợp này nó tìm cú pháp JavaScript style sheet.

Khi dùng cú pháp JavaScript trong phần tử <STYLE>, ta phải xác định thuộc tính các thẻ của đối tượng tài liệu.

Trong mô hình đối tượng tài liệu, các phần tử HTML được đối xử như các đối tượng. Một đối tượng có các thuộc tính và có thể được truy nhập để định nghĩa diện mạo và style của phần tử. Một thuộc tính trả về có thể hoạt động như một đối tượng với một tập các thuộc tính. Ví dụ, một trang Web được đối xử như một đối tượng tài liệu. Đối tượng tài liệu có một thuộc tính các thẻ. Thuộc tính các thẻ có một đối tượng H2. Đối tượng H2 có một thuộc tính gọi là color mà có thể được truy nhập và thay đổi khi được yêu cầu.

```
Document.tags.H2.color = "limegreen"
```

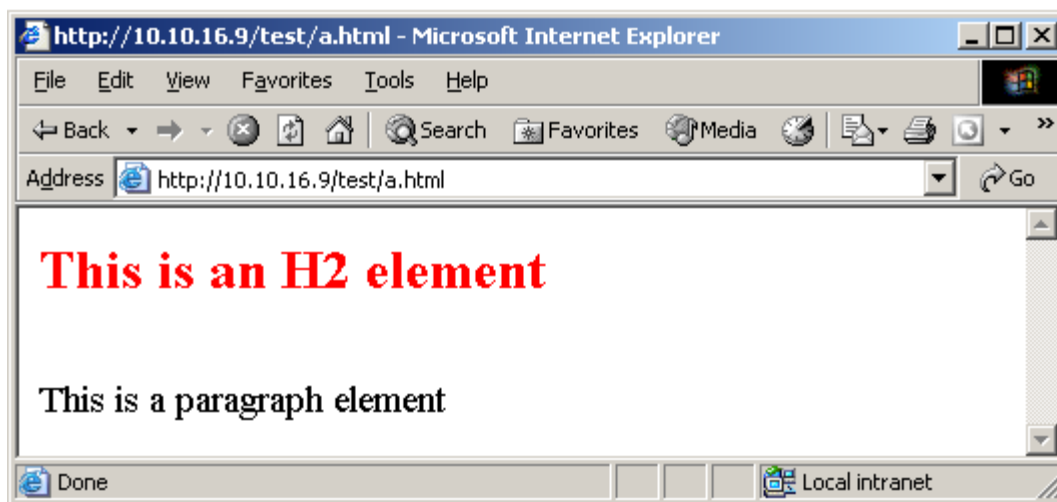
Ví dụ:

```
<HTML>
<HEAD>
<STYLE TYPE = "text/javascript">
tags.P.fontSize = "25pt";
tags.P.marginLeft = "15pt";
tags.H2.color = "hotpink";
</STYLE>
</HEAD>
<BODY>
<H2>This is an H2 element with a style applied</H2>
<BR>
<P>This is a paragraph element
</BODY>
</HTML>
```

Tất cả các phần tử khác đều thừa kế từ phần tử <BODY>. Nếu ta muốn cung cấp một style mặc định cho tất cả các phần tử trong một tài liệu, ta có thể đặt style yêu cầu trong phần tử <BODY>. Đoạn mã sau chạy trên trình duyệt netscape:

Ví dụ:

```
<HTML>
<HEAD>
<STYLE TYPE = "text/javascript">
tags.body.color="red";
</STYLE>
</HEAD>
<BODY>
<H2>This is an H2 element</H2>
<BR>
<P>This is a paragraph element
</BODY>
</HTML>
```



*Hình VII.3. Hỗ trợ một style bình thường*

Ví dụ:

```
<HTML>
<HEAD>
<STYLE TYPE = "text/css">
BODY {color:limegreen}
</STYLE>
</HEAD>
<BODY>
```

```
<H2>This is an H2 element</H2>
<BR>
<P>This is a paragraph element</P>
</BODY>
</HTML>
```

Nếu ta phải đặt nhiều thuộc tính cho một phần tử, ta có thể dùng cú pháp để rút ngắn lại mã:

```
<HTML>
<HEAD>
<STYLE TYPE = "text/javascript">
with (tags.P) {
color="hotpink";
fontStyle="italic";
fontFamily="helvetica";
fontSize="20pt";
}
</STYLE>
</HEAD>
<BODY>
<P>This is a paragraph element</P>
</BODY>
</HTML>
```

#### **VII.2.4. Thuật ngữ Style Sheet**

Một cascading style sheet định nghĩa các style mà có thể được cung cấp cho các trang hoặc các phần tử trang.

**Luật style** - - Một cascading style sheet là một tập các luật. Một luật định nghĩa định dạng của tài liệu. Ví dụ, ta có thể tạo một luật style để quy định tất cả các <H2> xuất hiện với màu xanh.

**Style Sheet** - - là một danh sách các luật. Nó có thể được nhúng vào trong tài liệu HTML. Trong trường hợp đó nó được gọi là một style sheet nhúng. Một style sheet cũng có thể được tạo như một file bên ngoài và liên kết với tài liệu HTML. Các luật style có thể được cung cấp để lựa chọn các phần của một trang Web. Ví dụ, ta có thể quy định một đoạn văn bản xuất hiện dạng chữ đậm và nghiêng. Việc này được gọi là khai báo bên trong style (inline style declaration) nhờ đó các style được cung cấp để chia các phần tử HTML trên một trang Web.

**Các luật** - - Một style sheet có thể gồm một hoặc nhiều luật. Phần đầu của luật được gọi là Selector. Mỗi selector có cá thuộc tính và các giá trị kết hợp với nó.

Ví dụ:

```
A RuleSelector{Declarationsproperty: value; property:
value; ... }
```

Phần của luật kèm theo bên trong các ngoặc nhọn được gọi là Khai báo (Declaration). Một khai báo có hai phần, phần trước dấu hai chấm là thuộc tính (Property) và phần sau dấu hai chấm là giá trị (Value) của thuộc tính đó.

Các khai báo phân chia ra bởi dấu chấm phẩy (;). Ta không cần đặt một dấu phẩy chấm sau khai báo cuối cùng.

Ví dụ: H1 {color:blue}

H1 là Selector, color:blue là khai báo.

Bên trong khai báo:

```
{Thuộc tính:Giá trị}
```

Color là thuộc tính, blue là giá trị.

Ta có thể đặt mỗi luật tách rời bên trong các thẻ STYLE. Trong trường hợp đó các header của ta sẽ được liệt kê.

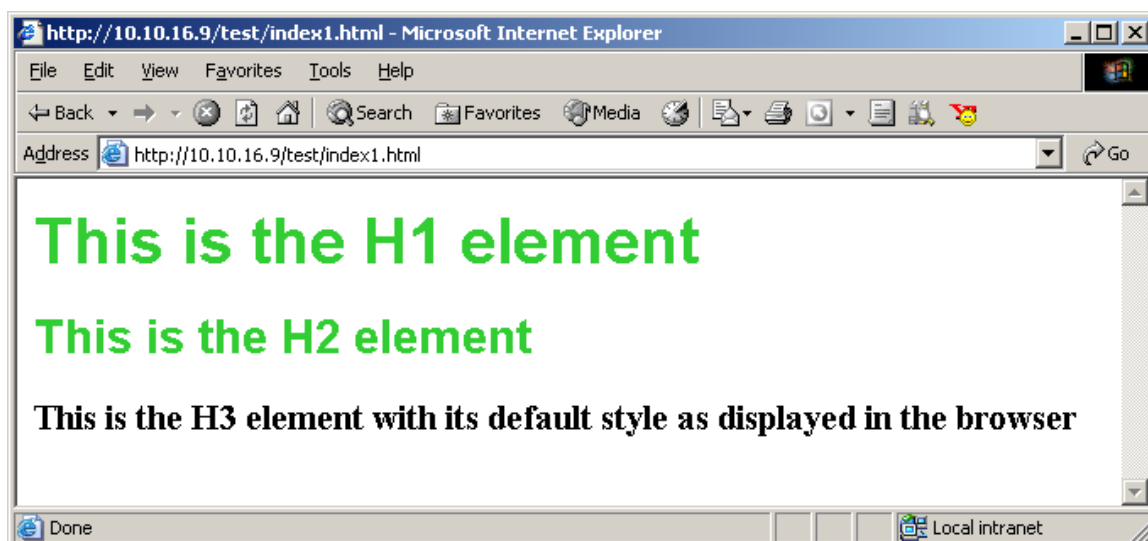
Ví dụ:

```
<HTML>
<HEAD>
<STYLE TYPE = "text/css">
H1{ color:limegreen }
H1{ font-family: Arial }
    { color: limegreen }
H2{ font-family:Arial }
</STYLE>
</HEAD>
<H1>This is the H1 element</H1>
<H2>This is the H2 element</H2>
<H3>This is the H3 element with its default style as
displayed in the browser</H3>
</HTML>
```

Ta cũng có thể nhóm các luật. Mỗi khai báo được phân chia bởi một dấu chấm phẩy.

Ví dụ:

```
<HTML>
<HEAD>
<STYLE TYPE = "text/css">
    H1, H2{ color: limegreen;font-family: Arial;}
</STYLE>
</HEAD>
<H1>This is the H1 element</H1>
<H2>This is the H2 element</H2>
<H3>This is the H3 element with its default style as
displayed in the browser</H3>
</HTML>
```



*Hình VII.4. Luật nhóm*

## **Các Selector**

Selector có thể được định nghĩa như, “một xâu mà quy định các luật nào hỗ trợ cho các phần tử nào.

Có hai kiểu cơ bản của các selector

- **Các selector đơn giản (Simple selectors)**
  - HTML Selectors
  - Class selectors



- ID Selectors
- **Các selector ngữ cảnh (Contextual selectors)**

#### a. Các selector đơn giản (Simple selectors)

Sử dụng dễ nhất. Nó mô tả một phần tử không kể vị trí của nó trong cấu trúc tài liệu. Quy định H1 là một selector đơn giản.

Ví dụ: `H1 {color:blue}`

#### HTML Selector

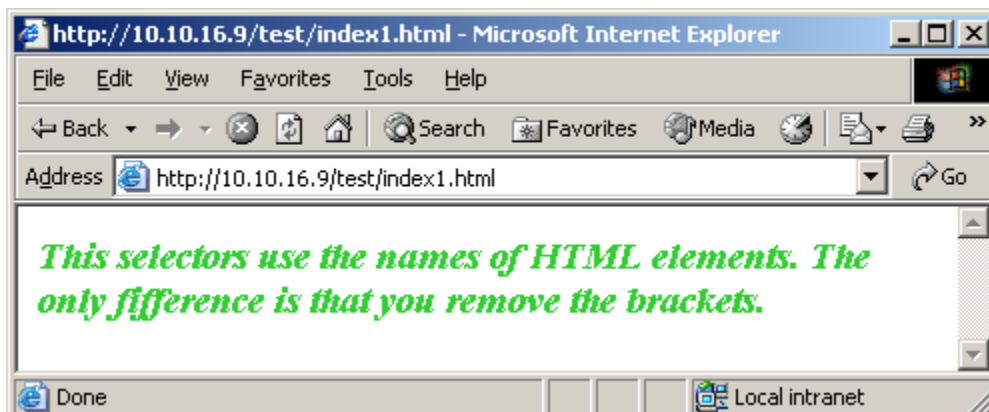
Selector này sử dụng các tên của các phần tử HTML. Chỉ khác là ta loại bỏ các móc nhọn. Như vậy, thẻ `<P>` HTML trở thành `P`. Trong ví dụ dưới đây, trong khi định nghĩa style, phần tử `P` không có các móc nhọn. Điều này bởi vì, phần tử `<P>` HTML được đối xử như một selector.

Ví dụ:

```
<HTML>
<HEAD>
<STYLE TYPE="text/css">
P{font-style:italic; font-weight:bold;color:limegreen}
</STYLE>
</HEAD>
```

Ở đây các nội dung của `<P>` được xác định. Phần tử `<P>` được đối xử như một phần tử HTML.

```
<P>This selectors use the names of HTML elements. The
only fifferece is that you remove the brackets.</P>
</BODY>
</HTML>
```



Hình VII.5. Sử dụng một HTML selector

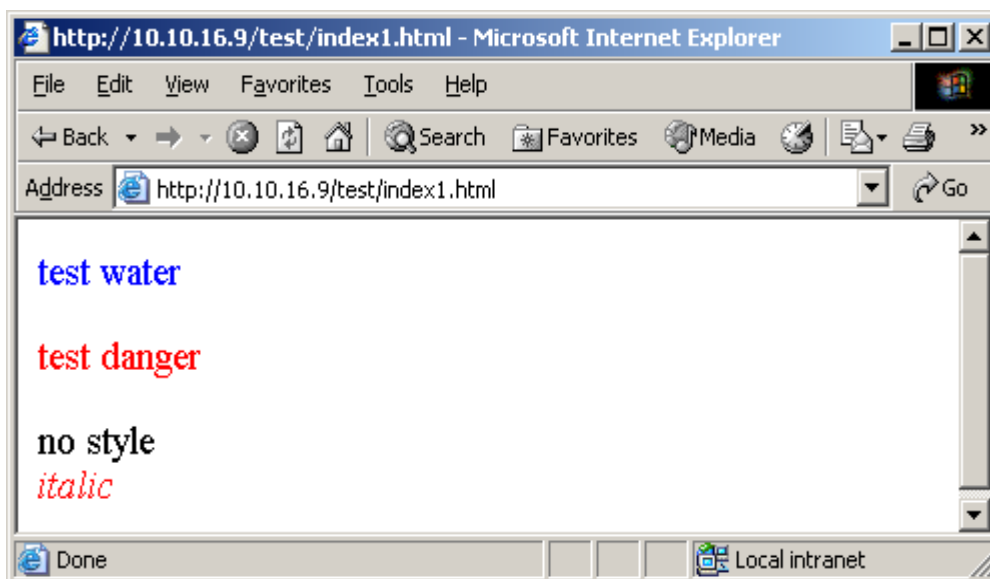
## Class Selector

Các selector này dùng thuộc tính CLASS của các phần tử HTML. Mọi phần tử, mà thấy được, có một thuộc tính CLASS mà được dùng để gán một quy định. Thay vì ta có thể gán một quy định lớp cho nhiều phần tử của một kiểu đơn khi ta muốn hiển thị nhiều màu khác nhau. Trong trường hợp đó ta sẽ dùng một quy định lớp cho <H2>.

CLASS Selector được bắt đầu với một dấu chấm (.) gọi là ký tự cờ (flag), theo sau bởi ‘tên lớp’ của selector. Sẽ tốt hơn để chọn các tên lớp theo mục đích hơn là một tên mà mô tả màu hoặc style của họ. Ví dụ, ta có thể muốn đoạn A xuất hiện dạng chữ nghiêng, các đoạn khác xuất hiện với style khác, trong trường hợp đó đoạn A có thể có một quy định lớp.slant.

Ví dụ:

```
<STYLE TYPE="text/css">
.water { color: blue }
.danger { color: red }
</STYLE>
<P class="water">test water
<P class="danger">test danger
<P>no style
<BR>
<EM class="danger">italic</EM>
```



Hình VII.6. Sử dụng CLASS selector

Ví dụ:

```
<HTML>
<HEAD>
<STYLE TYPE="text/javascript">
    classes.water.all.color = "blue"
    classes.danger.all.color = "red"
</STYLE>
</HEAD>
<P class="water">test water
<P class="danger">test danger
<P>no style
<BR>
<EM class="danger">italic</EM>
</BODY>
</HTML>
```

Khi dùng cú pháp JavaScript, ta không thể dùng gạch nối bên trong các tên lớp. Bởi vì JavaScript đọc gạch nối như một dấu trừ (-) mà là một toán tử JavaScript. Các tên lớp không thể gồm bất kỳ toán tử JavaScript nào.

Khi định nghĩa một style class:

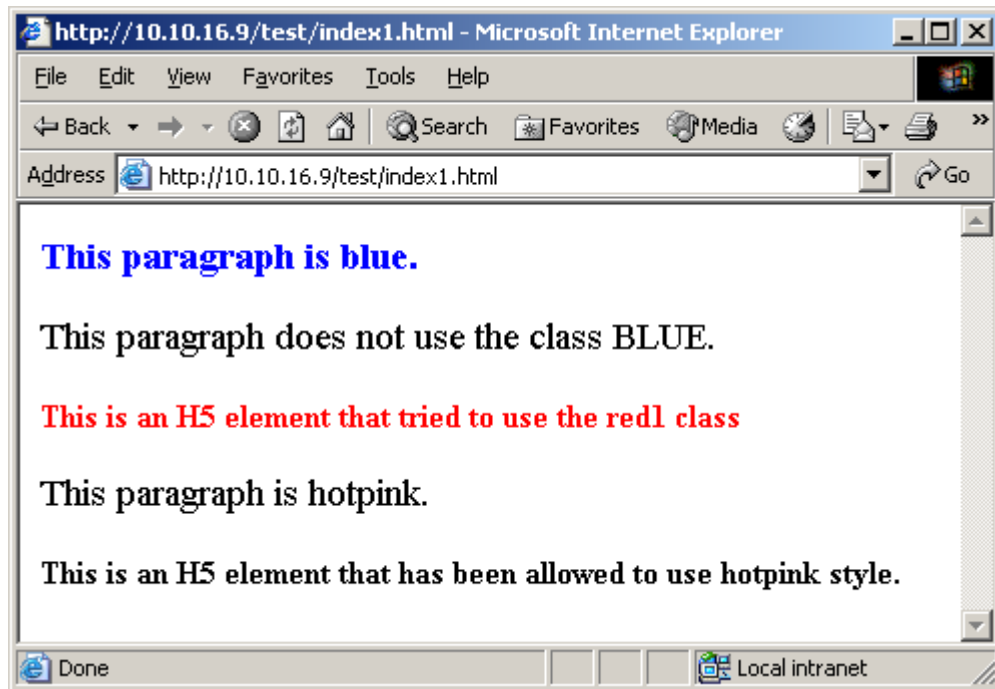
Ta có thể xác định các phần tử HTML nào có thể dùng style này

Ta có thể dùng từ khóa tất cả để cho tất cả các phần tử có thể dùng nó.

Ví dụ:

```
<HTML>
<HEAD>
<STYLE TYPE="text/css">
all.hotpink {color:hotpink;}
P.BLUE {color: blue; font-weight:bold;}
H5.red1 {color:red; font-weight:bold;}
</STYLE>
</HEAD>
<BODY bgColor = lavender>
<P CLASS=BLUE>This paragraph is blue.</P>
<P>This paragraph does not use the class BLUE.</P>
<H5 CLASS=red1>This is an H5 element that tried to use
the red1 class</H5>
<P CLASS=hotpink>This paragraph is hotpink.
<H5 CLASS=hotpink>This is an H5 element that has been
allowed to use hotpink style.</H5>
</BODY>
```

</HTML>



*Hình VII.7. Xác định các phần tử*

Ví dụ:

```
<HTML>
<HEAD>
<STYLE TYPE="text/javascript">
  classes.HOTPINK.all.color="hotpink";
  classes.BLUE.P.color = "blue";
  classes.BLUE.P.fontWeight = "bold";
  classes.red1.H5.color = "red";
  classes.red1.H5.fontWeight = "bold";
</STYLE>
</HEAD>
<BODY bgColor = lavender>
<P CLASS=BLUE>This paragraph is blue.</P>
<P>This paragraph does not use the class BLUE.</P>
<H5 CLASS=hotpink>This an H5 element that tried to use
the red1 class</H5>
<P CLASS=hotpink>This paragraph is hot pink.
<H5 CLASS=hotpink>This is an H5 element that has been
allowed to use hotpink style.</H5>
</BODY>
```

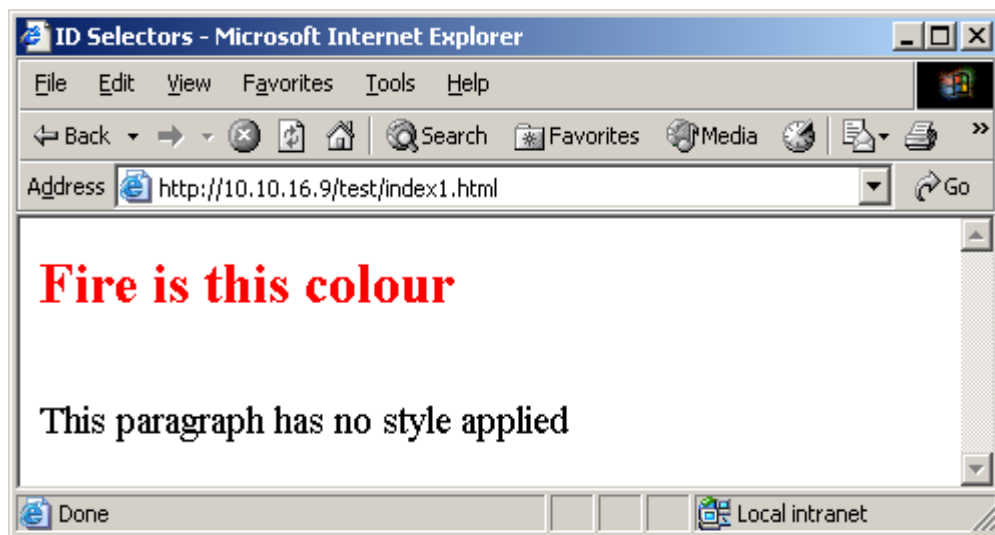
</HTML>

## ID Selector

Một ID Selector dùng thuộc tính ID của một phần tử HTML. Một ID selector được dùng để cung cấp một style cho một phần tử cụ thể trên trang Web. Ví dụ, ta có thể dùng một ID selector để cung cấp cho một tiêu đề <H2>. Không có nghĩa là cùng một style sẽ được cung cấp cho sự kiện khác của một phần tử <H2> trên cùng một trang, trừ khi chỉ rõ. Nó tương tự như việc dùng một inline style do đó một style được cung cấp cho một phần tử xác định. Một ID selector được bắt đầu bởi một dấu thăng (#). Khi dùng cú pháp JavaScript, ta phải dùng thuộc tính ID.

Ví dụ:

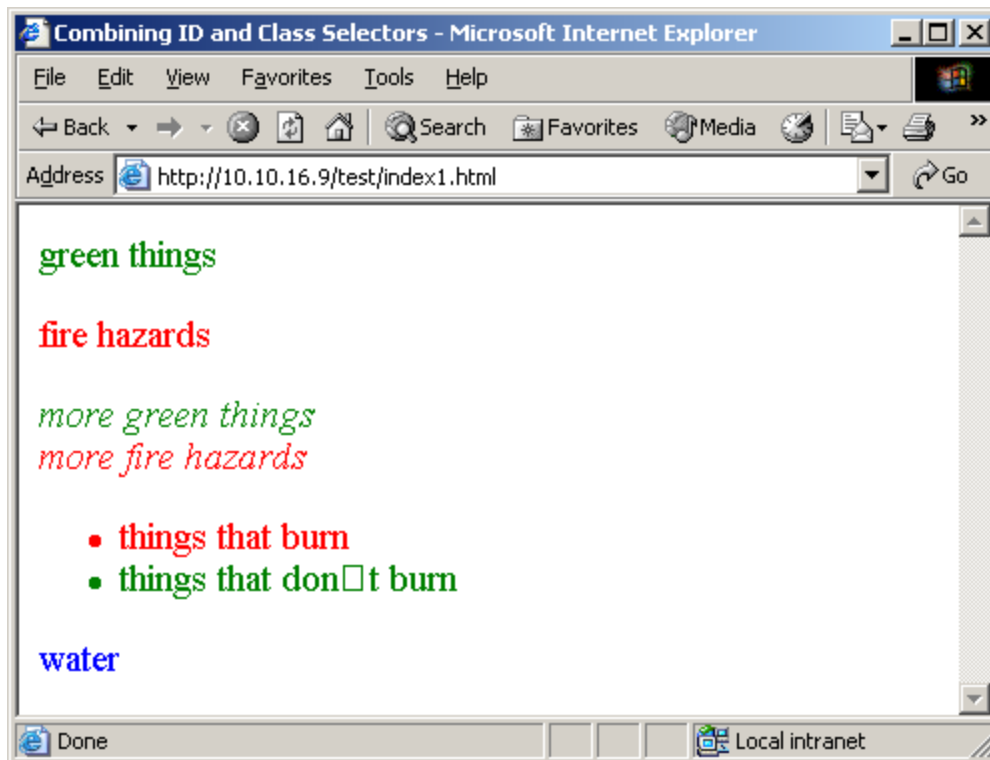
```
<HTML>
<HEAD>
<TITLE> ID Selectors</TITLE>
</HEAD>
<BODY>
<STYLE TYPE="text/css">
    #control {color: red}
</STYLE>
<H2 id="control">Fire is this colour</H2>
<BR>
<P>This paragraph has no style applied
</BODY>
</HTML>
```



Hình VII.8. Sử dụng ID selector

## Ví dụ:

```
<HTML>
<HEAD>
<TITLE>Combining ID and Class Selectors</TITLE>
</HEAD>
<BODY>
<STYLE TYPE="text/css">
    .forest {color: green }
    .danger {color: red }
    #control {color: blue }
</STYLE>
<P class="forest">green things
<P class="danger">fire hazards
</P>
<EM class="forest">more green things</EM>
<BR>
<EM class="danger">more fire hazards</EM>
<UL>
<LI class="danger">things that burn
<LI class="forest">things that don't burn
</UL>
<P id="control">water</P>
</BODY>
</HTML>
```



*Hình VII.9. Kết hợp ID và Class selector*

Ví dụ:

```

<HTML>
<HEAD>
<TITLE>Combining ID and Class Selectors</TITLE>
</HEAD>
<BODY>
<STYLE TYPE="text/javascript">
With (classes.forest.all) {
Color = "green";
}
With (classes.danger.all){
Color = "red";
}
idss.control.color = "blue";
</STYLE>
<P class="forest">green things
<P class="danger">fire hazards
</P>
<EM class="forest">more green things</EM>
<BR>
<EM class="danger">more fire hazards</EM>
<UL>

```

```

<LI class="danger">things that burn
<LI class="forest">things that don't burn
</UL>
<P id="control">water</P>
</BODY>
</HTML>

```

## b. Các selector ngữ cảnh (Contextual Selectors)

Một contextual selector đề cập đến ngữ cảnh của các phần tử. Để làm mọi thứ sáng sủa hơn, đôi khi ta có hai phần tử với cùng một giá trị. Phần tử chính hoặc phần tử cha có một phần tử con bên trong nó. Trong trường hợp đó, để thay đổi style của phần tử con ta cần dùng một contextual selector.

Điều này dựa trên khái niệm kế thừa, nơi mà phần tử con kế thừa style được gán cho phần tử cha.

Một ví dụ thông thường là phần tử <BODY>. Khi ta thêm một phần tử vào mỗi phần tử <BODY> chứa bên trong những kế thừa của <BODY>.

```
<p>...<b>...</b>...</p>
```

P là cha và B là con.

Bây giờ thì xem nó được điều khiển ra sao? Sau khi ta không muốn tất cả các phần tử trên trang Web xuất hiện cùng một style. Như vậy, ta phải ghi đè sự kế thừa. Trong trường hợp đó ta sẽ phải thay đổi style của phần tử con. Xem đoạn code dưới đây:

Ví dụ:

```

<HTML>
<HEAD>
<TITLE>Contextual selectors</TITLE>
</HEAD>
<STYLE TYPE="text/css">
BODY { color:blue;
Background:lavender;
Font-family:Arial;
}
UL {color:red}
</STYLE>

```

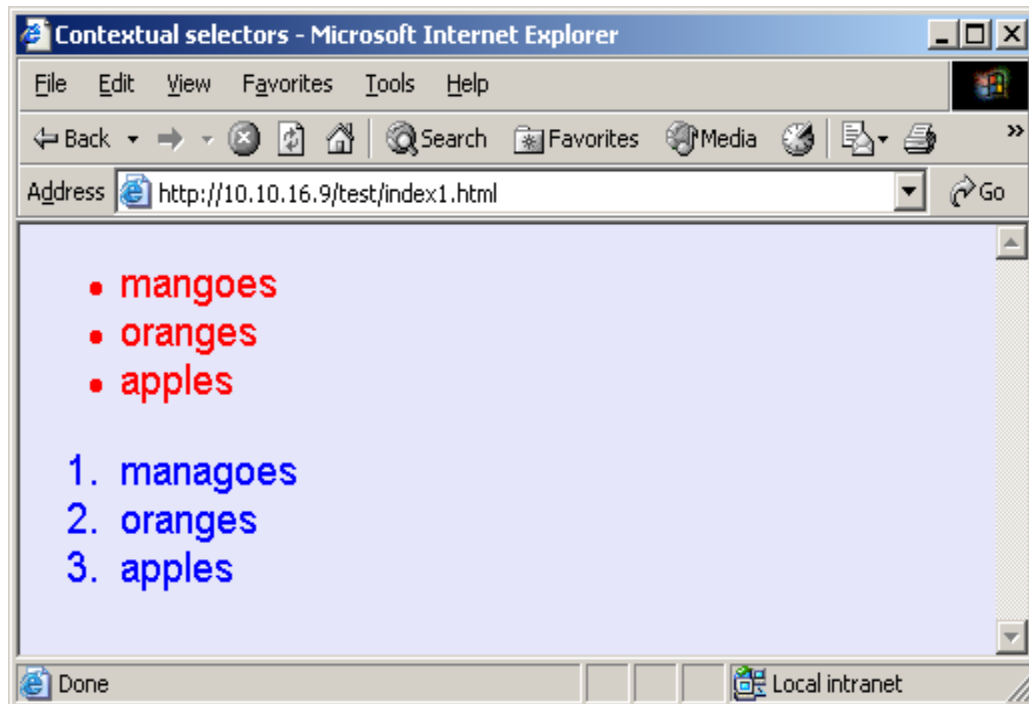
Selector UL trong style sheet xác định rằng danh sách không đánh số màu đỏ, do đó tất cả các phần tử LI sẽ là màu đỏ trừ khi nó được ghi đè sự kế thừa này.



```
<UL>
<LI> mangoes
<LI> oranges
<LI> apples
</UL>
```

Selector UL Selector trong style sheet xác định rằng các chỉ mục danh sách sẽ là màu đỏ. Chúng thừa kế font Arial từ khai báo BODY, màu đỏ từ khai báo UL. Nếu ta xác định một font family trong khai báo UL, nó sẽ ghi đè khai báo selector BODY. Không có selector OL trong style sheet, do đó danh mục OL kế thừa các thuộc tính của nó từ selector BODY.

```
<OL>
<LI> managoes
<LI> oranges
<LI> apples
</OL>
</BODY>
</HTML>
```



*Hình VII.10. Contextual Selectors*

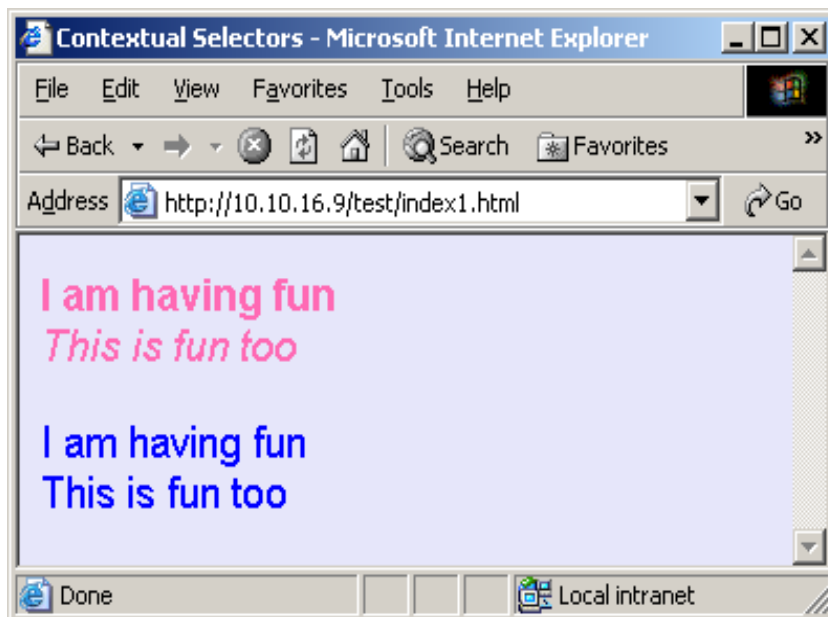
Các phần tử không được kế thừa trong style sheet như các selector sẽ kế thừa các thuộc tính của phần tử cha nó. Trong ví dụ dưới đây, các phần tử B và I đang kế thừa các thuộc tính của selector P.

Ví dụ:

```
<HTML>
<HEAD>
<TITLE>Contextual Selectors</TITLE>
</HEAD>
<STYLE TYPE="text/css">
BODY { color:blue;
      background:lavender;
font-family:Arial;
      }
P {color:hotpink}
</STYLE>
<BODY>
<P>
<B>I am having fun</B><BR>
<I>This is fun too</I><BR>
```

Để sự trả về của các thuộc tính của selector P, đóng nó lại

```
</P>
I am having fun
<BR>
This is fun too
</BODY>
</HTML>
```



Hình VII.11. Contextual Selectors

Ví dụ:

```
<HTML>
<STYLE TYPE="text/javascript">
With (tags.BODY) {
    color="blue";
    backgroundColor="lavender";
    fontFamily="Arial";
}
tags.P.color="hotpink";
</STYLE>
<BODY>
<P>
<B>I am having fun</B><BR>
<I>This is fun too</I><BR>
<P>
I am having fun
<BR>
This is fun too
</BODY>
</HTML>
```

## Thay đổi các luật

Ta phải tìm hiểu để chỉnh sửa style của một phần tử cụ thể. Có thể các kế thừa khi ta muốn làm các thứ trên phạm vi toàn cục, cho tất cả các trang trong web site. Ta có thể muốn chỉnh sửa một style định nghĩa bởi một style sheet như một lớp hoặc style toàn cục. Khi ta làm điều này, sự sửa đổi có một hiệu ứng gọn sóng. Đó là, style của tất cả các phần tử trên trang mà đưa ra style sheet đặc biệt được thay đổi.

Ví dụ, ta có một định nghĩa style sheet mà xác định hai style:

Một style toàn cục mà cung cấp toàn bộ phần tử <H2> (green, arial font, normal size)

Một lớp style chung gọi là cảnh báo (red, bold, italic) mà sẽ cung cấp bất kỳ một phần tử nào mà sử dụng lớp đó.

Hai ví dụ sau đây miêu tả việc dùng style sheet gọi là **sheet1.css** mà được liên kết với 1.htm đầu tiên và sau đó là 2.htm

Ví dụ:

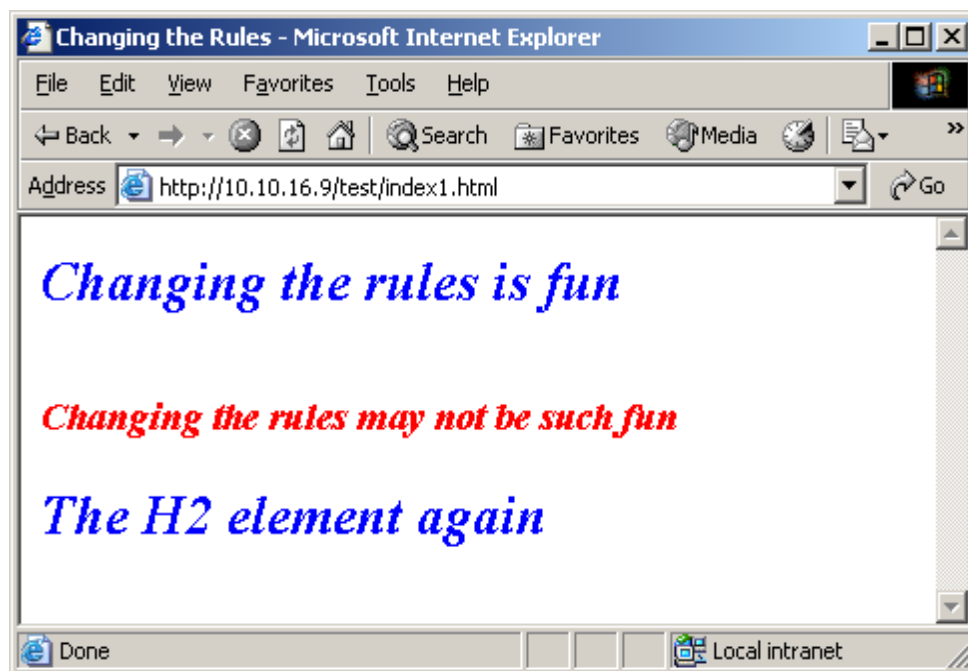
```
sheet1.css:
H2 {color:blue; font-style:italic;}
.warning {color:red; font-weight:bold; font-
style:italic;}
```

Copy đoạn code này và ghi với tên “**sheet1.css**”

Sau đó trong file f1.htm ta sử dụng nó như sau:

f1.htm

```
<HTML>
<HEAD>
<TITLE>Changing the Rules</TITLE>
  <LINK REL=stylesheet HREF="sheet1.css"
  TYPE="text/css">
</HEAD>
  <H2>Changing the rules is fun</H2>
  <BR>
  <P class="warning">Changing the rules may not be
  such fun
  <H2>The H2 element again<p>
</HTML>
```



Hình VII.12. Sử dụng một style sheet thông thường

Trong file f2 . htm ta cũng sử dụng:

f2 . htm

```
<HTML>
<HEAD>
<TITLE>Changing the Rules</TITLE>
  <LINK REL=stylesheet HREF="sheet1.css"
    TYPE="text/css">
</HEAD>
  <H2>This document uses the sheet1 style
sheet<H2>
  <BR>
  <P class="warning">Selecting this option could
delete all your files
  <H2>The H2 element again<P>
</HTML>
```

### **VII.2.5. Các chú thích trong Style Sheet**

Các chú thích (comments) là một đặc tính mà hầu hết các ngôn ngữ phát triển đều hỗ trợ. Ta cũng có thể thêm các chú thích vào một style sheet để giúp lưu giữ vết của các style mà được hỗ trợ thông qua trang. Các chú thích được bao quanh bởi các dấu /\*. Không được đặt lồng vào nhau.

Với cascading style sheet, cú pháp là:

```
H1 {color:blue;} /*Các phần tử H1 màu xanh*/
tags.H1.color="blue"; /*Các phần tử H1 màu xanh*/
```

Với JavaScript style sheet, cú pháp là:

```
tags.H1.color="blue"; //Các phần tử H1 màu xanh
```

### **VII.2.6. Lợi ích của các Style Sheet**

Nếu ta muốn sử dụng điều khiển lớn hơn với các trang Web ta nên sử dụng style. Ta có thể sử dụng style sheet cho:

#### **a. Ghi đè các ngầm định trình duyệt**

Mỗi trình duyệt có cách thể hiện các trang Web riêng của nó. Ngày nay, các chuyên viên phát triển không có quyền với trang được thể hiện trong một trình duyệt. Ta không biết được trình duyệt nào trên toàn cầu được sử dụng. Với các style sheet ta có thể ghi đè các mặc định truyền thống của trình duyệt và gán nó thành của riêng ta. Ví dụ, ta có thể xác định style trong phần tử <H1> như sau:

```
<H1><FONT SIZE=3 COLOR=AQUA>  
<B><I>Overriding the browser</I></B>  
</FONT></H1>
```

## **b. Bố trí trang**

Các style sheet có thể được dùng để hiển thị font, thay đổi màu sắc,... mà không thay đổi cấu trúc của trang web. Nghĩa là một người thiết kế có thể tách rời các yêu cầu thiết kế trực quan từ cấu trúc logic của trang Web và đánh địa chỉ khác nhau.

Việc dùng các độ đo tương đối trong style sheet của ta, ta có thể biểu diễn tài liệu để trông đẹp hơn trên màn hình tại bất kỳ độ phân giải nào.

## **c. Các Style Sheet có thể được tái sử dụng**

Sau khi ta định nghĩa thông tin style, ta có thể nhúng style sheet vào trong tài liệu HTML. Ta cũng có thể liên kết toàn bộ các trang trên web site tới style sheet. Điều này đảm bảo rằng các trang web của ta có một diện mạo đồng bộ khi chúng được hiển thị. Như vậy ta có thể có một nền thông thường, logo công ty và một số thông tin chuẩn trong một style sheet. Điều này sẽ đảm bảo rằng một cái nhìn thông thường và cảm giác trên Web site. Hãy tưởng tượng nếu có hàng trăm trang và ta phải định nghĩa style của mỗi trang riêng biệt.

## **d. Tạo một lần**

Ta có thể tạo một style sheet và liên kết nhiều tài liệu tới nó. Tất cả các tài liệu sẽ có một diện mạo. Dù sao quan trọng nhất là khi ta tạo một thay đổi tới style sheet tất cả các tài liệu liên kết tới style sheet sẽ mang lại sự thay đổi.

## **VII.2.7. Kết hợp Style Sheet với HTML**

Có nhiều cách để kết hợp style sheet với HTML

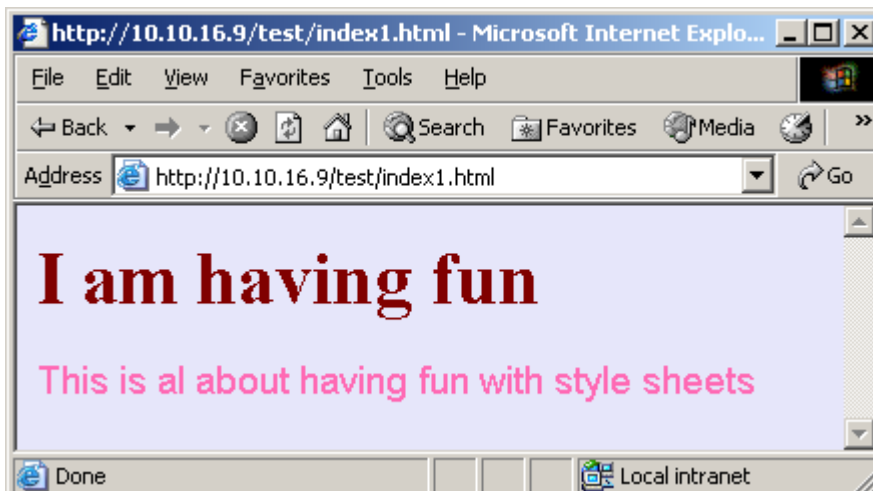
- Phần tử STYLE
- Thuộc tính Style
- Phần tử liên kết

### **Phần tử STYLE (STYLE element)**

Phần tử STYLE được chèn vào trong phần tử <HEAD> của một tài liệu với toàn bộ luật đặt giữa các thẻ mở và đóng. Khi các trang được hiển thị, chỉ tài liệu mà có STYLE được nhúng vào mới bị tác động. Thẻ <STYLE> có một tham số --TYPE. Tham số này xác định kiểu Internet Media như “text/css”.

Ví dụ:

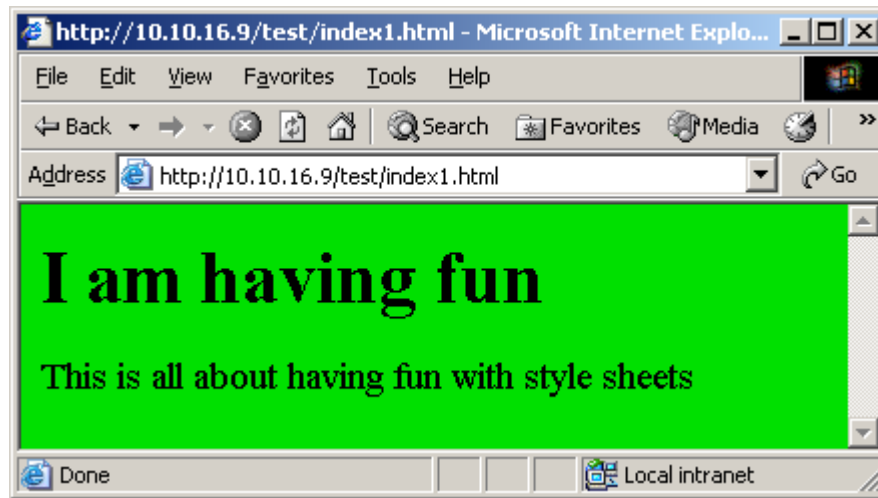
```
<HTML>
<STYLE TYPE="text/css">
  H1 {color:maroon;}
  P {color:hotpink;
    Font-family:Arial;
  }
</STYLE>
<BODY bgcolor="lavender">
  <H1>I am having fun</H1>
  <P>This is al about having fun with style
sheets</P>
</BODY>
</HTML>
```



*Hình VII.13. Sử dụng phân tử STYLE*

```
<HTML>
<HEAD>
  <STYLE TYPE="text/javascript">
    tags.H1.color="maroon";
    with (tags.P) {
      color="hotpink";
      fontFamily="Arial";
    }
  </STYLE>
</HEAD>
<BODY bgcolor="lavender">
  <H1>I am having fun</H1>
  <P>This is all about having fun with style
sheets</P>
```

```
</BODY>
</HTML>
```



*Hình VII.14. Sử dụng phân tử STYLE*

## **Thuộc tính STYLE**

Thuộc tính STYLE được dùng để cung cấp style sheet cho các phần tử riêng lẻ. Một Style Sheet có thể nhỏ bằng một luật. Sử dụng thuộc tính Style ta có thể bỏ qua phần tử Style và đặt các khai báo trực tiếp vào trong các thẻ bắt đầu riêng lẻ.

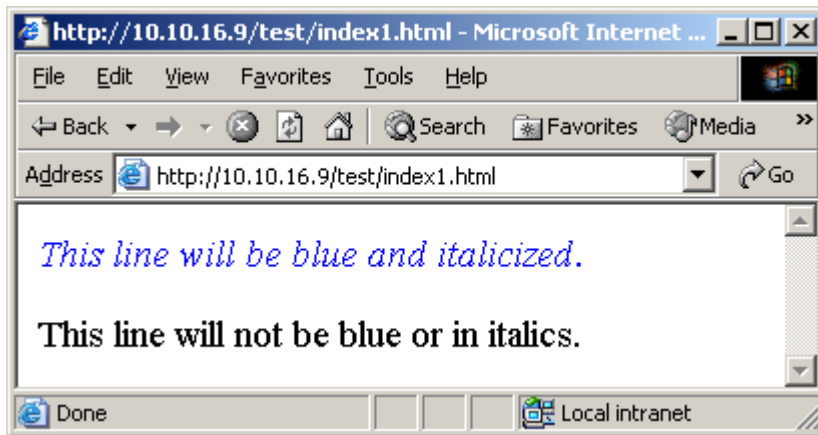
```
<H2 style="color:green;
font-family: Arial">
```

Thông thường, ta có thể chỉ dùng thuộc tính này nếu ta đang thay đổi style của một phần tử đặc biệt. Nếu ta có ý định cung cấp cùng một style trên toàn bộ tài liệu, khi đó đây không thật sự là cách hay để đạt được điều này.

Ví dụ:

```
<HTML>
  <P STYLE="color:blue; font-style:italic">
    This line will be blue and italicized.
  <P>
    This line will not be blue or in italics.
</HTML>
```





Hình VII.15. Sử dụng thuộc tính STYLE

### Phần tử liên kết (Link Element)

Nếu ta muốn dùng phần tử liên kết, khi đó Style Sheet phải được tách khỏi tài liệu. Khi đó địa chỉ Web với style sheet có thể được thêm vào.

```
<LINK REL=stylesheet  
      HREF="stylesmine.css"  
      Type="text/css"  
>
```

Phải có cả thuộc tính “rel=stylesheet” hoặc trình duyệt sẽ không tải style sheet.

Ví dụ:

Sheet1.css

```
H2 {color:blue; font-style:italic;}  
P {color:yellow;}
```

File.htm

```
<HTML>  
<HEAD>  
  <TITLE>Linking external style sheets</STYLE>  
  <LINK REL=STYLESHEET TYPE="text/css"  
    HREF="sheet1.css">  
</HEAD>  
<BODY>  
  <H2>This is an H2 element</H2>  
  <BR>
```

```
<P>This is a paragraph
</BODY>
</HTML>
```

Sheet1.css

```
tags.H2.color="blue";
tags.H2.fontStyle="Italic";
tags.H2.color="yellow";
```

File.htm

```
<HTML>
<HEAD>
<TITLE>A Good Title</TITLE>
<LINK REL=STYLESHEET TYPE="text/javascript"
  HREF="sheet1.css">
</HEAD>
<BODY>
  <H2>This is an H2 element</H2>
  <BR>
  <P>This is a paragraph
</BODY>
</HTML>
```

### **VII.2.8. Thứ tự ưu tiên của các style (Cascading)**

Khi ta đang dùng nhiều kỹ thuật để cung cấp các style (liên kết và nhúng) vào trang Web, có một cách để trình duyệt quyết định chọn style nào. Trình duyệt xếp qua tất cả các luật được xác định và chọn một luật quan trọng nhất. Style sheet cục bộ có độ ưu tiên cao nhất. Tiếp theo là style sheet toàn cục với style sheet liên kết tại phía trên.

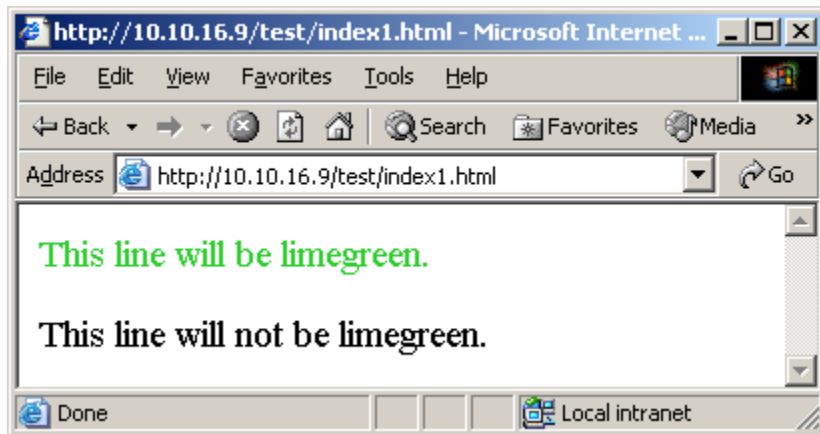
Ví dụ, nếu một dòng trên trang Web định nghĩa <P> là màu đỏ qua một style sheet liên kết, tất cả các đoạn văn sẽ là màu đỏ trừ <P> ở trên, mà sẽ là màu xanh dương.

```
<HTML>
<LINK REL=stylesheet
  HREF="sheet1.css"
  Type="text/css">
```

```

<P STYLE="color:limegreen">
    This line will be limegreen.
<P>
    This line will not be limegreen.
</HTML>

```

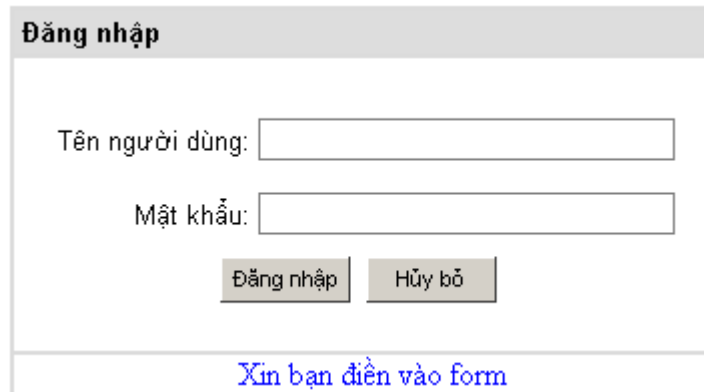


*Hình VII.16. Cascading style*

Thuộc tính	Tên CSS	Thuộc tính Scripting
Font properties	Font	font
	font-size	fontSize
	font-style	fontStyle
Text properties	Text-align	textAlign
	Text-indent	textIndent
	vertical-align	verticalAlign
Box properties	border	border
	border-width	borderWidth
	border-bottom	borderBottom
	border-color	borderColor
Positioning properties	Clip	clip
	height	height
	Left	left
	Top	top
	z-index	zIndex

## **Bài tập**

1. Tạo ra một trang web mới dùng để đăng nhập có giao diện như ở hình dưới (sử dụng FORM).



The image shows a login form with a grey header bar containing the text "Đăng nhập". Below the header, there are two input fields: "Tên người dùng:" followed by a text box, and "Mật khẩu:" followed by a text box. Below the input fields are two buttons: "Đăng nhập" and "Hủy bỏ". At the bottom of the form, there is a blue link that says "Xin bạn điền vào form".

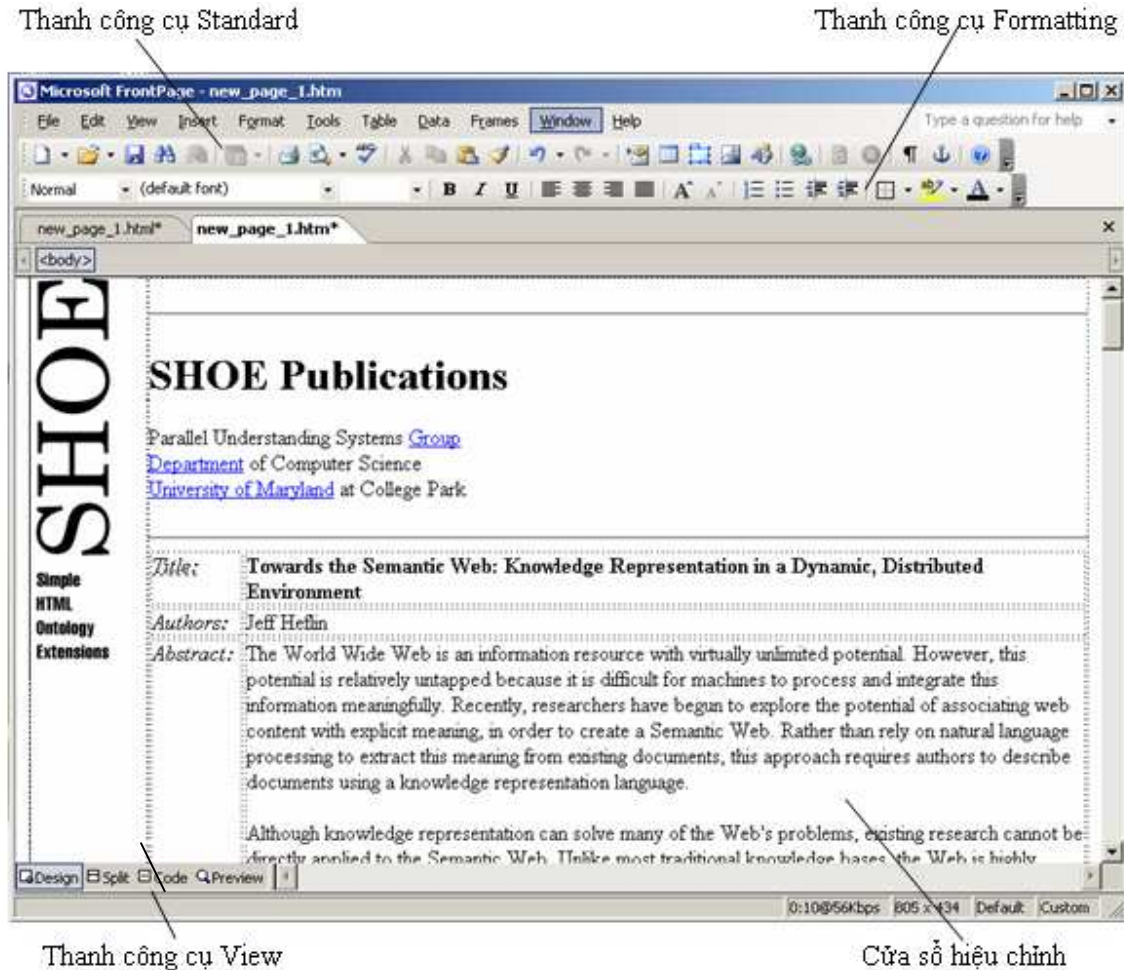
2. Tạo một file css để đặt mặc định kiểu font và màu cho các phần giới thiệu của bạn.

# CHƯƠNG VIII. CÔNG CỤ SOẠN THẢO TRỰC QUAN WEB (MICROSOFT FRONTPAGE 2003)

## VIII.1. Tạo một trang Web

### VIII.1.1. Bắt đầu sử dụng FrontPage2003

Front Pages2003 chứa tất cả các công cụ mà ta cần để tạo, xuất, và quản lý một World Wide Web site chuyên nghiệp, bắt mắt và thú vị.



Hình VIII.1. Giao diện đồ họa

Ta sẽ thấy giao diện đồ họa người dùng của chương trình, giao diện này giống như một bộ xử lý văn bản với một vài nút, thanh công cụ, và menu phụ.

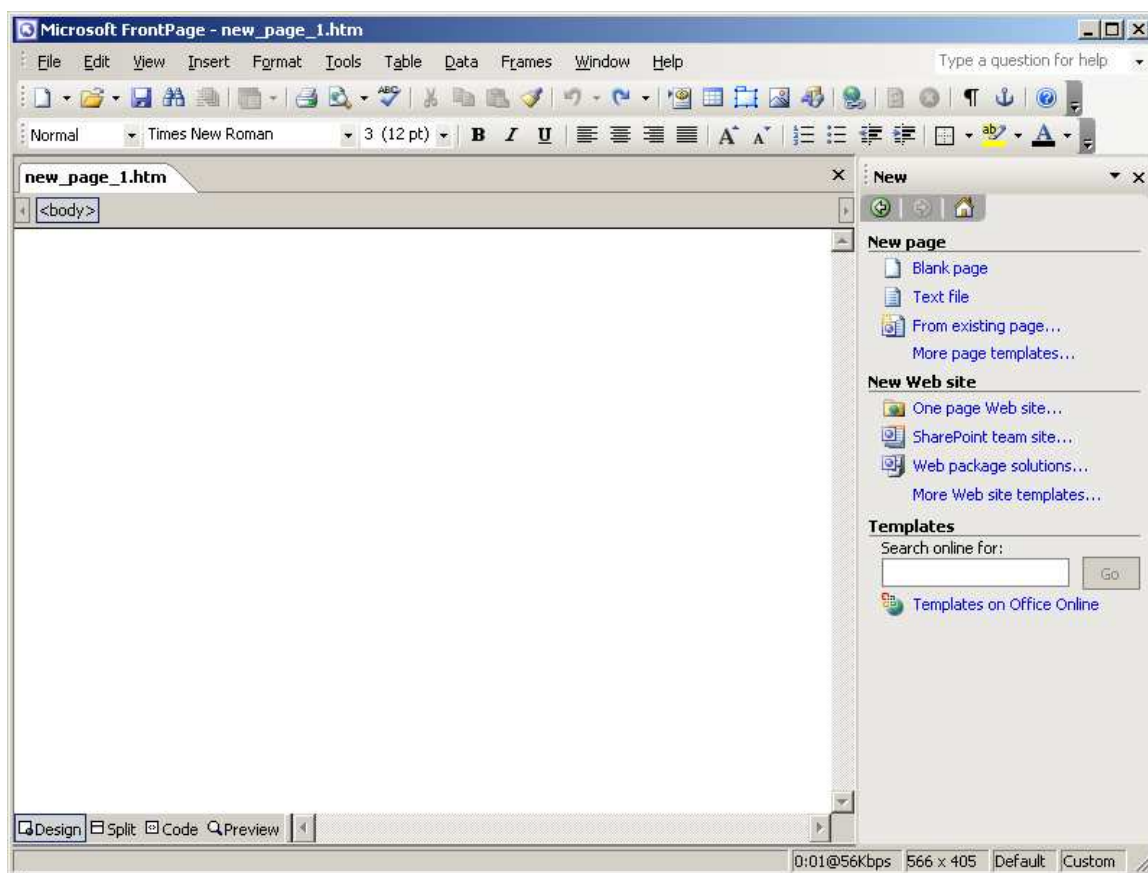
Khi ta chạy FrontPage lần đầu tiên, chương trình tạo một trang Web mới để ta có thể bắt đầu làm việc ngay lập tức. Khi làm việc, ta có thể thấy chính xác diện mạo của trang Web như thế nào khi một người nào đó xem trang bằng một trình duyệt Web.

Nếu ta muốn xem Internet Explorer sẽ hiển thị trang Web như thế nào, ta có thể nhấn nút Preview trên thanh công cụ Views. Khi đó cửa sổ hiệu chỉnh của ta sẽ được thay bằng một trình duyệt mini. Để đóng trình duyệt mini và quay về công việc của ta, nhấn nút Design trên thanh công cụ đó.

### **VIII.1.2. Tạo một trang từ một template**

Bây giờ ta đã biết đôi chút về các thanh công cụ và cửa sổ hiệu chỉnh, ta có thể sẵn sàng tạo trang Web riêng cho mình:

1. Mở menu Start, sau đó tìm và chạy FrontPage2003. FrontPage mở ra, một trang trống sẽ được tạo ra dành cho ta
2. Để tạo một trang Web mới, chọn menu File → New. Khung new mở ra nằm bên phải cửa sổ như trong hình



*Hình VIII.2. Khung New*

3. Click vào “More page template” trong phần New page. Hộp thoại Page Template mở ra, ta có thể liệt kê tất cả các template ta có thể sử dụng khi nạp trang Web mới.

4. Để tìm hiểu thêm về một template, click vào nó một lần và đọc phần Description của hộp thoại. Mỗi trang Web Frontpage phải được dựa vào một template
5. Để bắt đầu với trang hoàn toàn trống, chọn biểu tượng Normal Page; nếu không, chọn một trong các biểu tượng khác
6. Click vào nút OK. Hộp thoại Page Template đóng lại. FronPage sẽ tạo trang Web mới và mở nó trong cửa sổ biên tập. Ta có thể bắt đầu làm việc với nó ngay lập tức.

### **VIII.1.3. Tạo và lưu một trang mới**

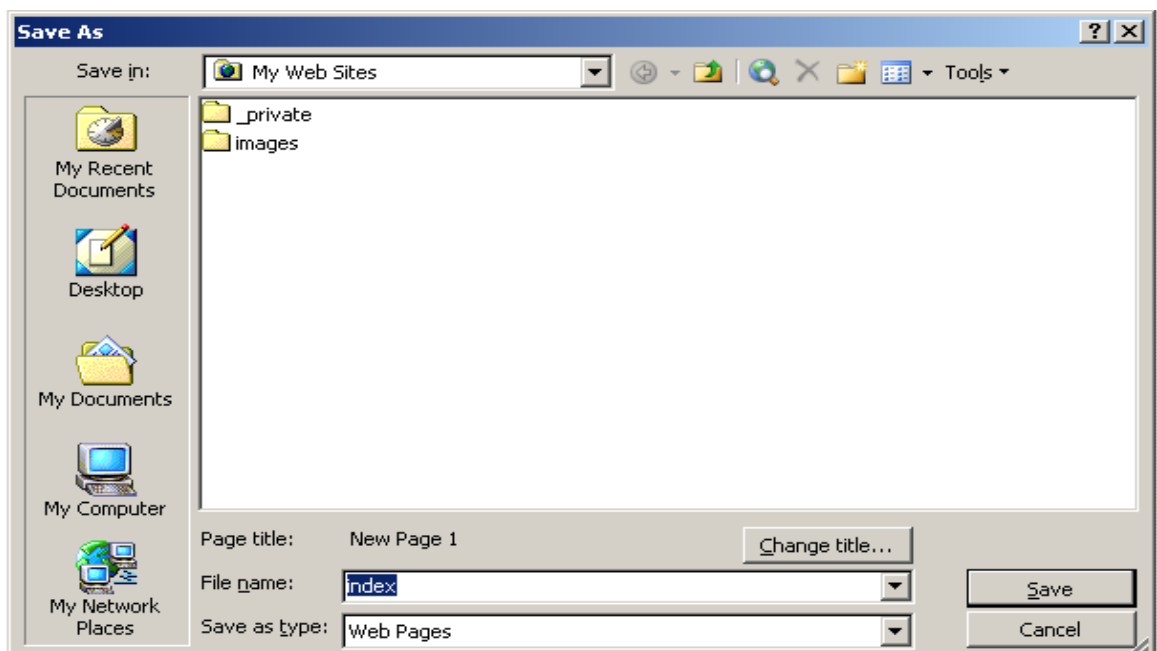
Khi ta lưu một file lần đầu tiên, ta có thể chọn một tên nào đó thích hợp hơn tên và tiêu đề hiện có.

Việc chọn một tiêu đề ngắn gọn, có tính mô tả là quan trọng vì hai lý do: Nó giúp người ta sử dụng trang của ta và nó giúp người khác tìm thấy nó.

Tiêu đề trang được hiển thị trong thanh tiêu đề của trình duyệt Web. Nó cũng được sử dụng bởi các công cụ tìm kiếm. Ví dụ, khi ta tìm kiếm về một thông tin nào đó trên Google tại <http://www.google.com>, mỗi trang Web xuất hiện được liệt kê theo tiêu đề của nó.

Ta đặt cho trang của ta một tên và tiêu đề mới khi ta lưu nó lần đầu tiên:

1. Click nút Save trên thanh công cụ Standard. Hộp thoại Save As sẽ xuất hiện (hình vẽ dưới)



*Hình VIII.3. Cửa sổ Save As*

2. Sử dụng hộp thoại này, tìm và mở thư mục nơi mà ta sẽ lưu các trang Web mà ta tạo.
3. Trong hộp text File name, đặt một tên mới cho file, phải chắc chắn là nó có phần mở rộng.htm (hoặc.html)
4. Phần Page Title hiển thị tiêu đề hiện hành. Để chọn một tiêu đề mới, click vào nút Change Title. Hộp thoại Set Page Title sẽ mở ra.
5. Tiêu đề của một trang nên có tính mô tả và xúc tích. Nhập một tiêu đề mới trong hộp text Page Title và click OK. Ta sẽ quay trở về hộp thoại Save As.
6. Click nút Save.

Sau khi bạn đã đặt tên và lưu một trang Web, ta có thể lưu lại nó một cách nhanh chóng bằng cách click nút Save trên thanh công cụ Standard. Tên file ta chọn trước đó sẽ được sử dụng lại một cách tự động.

#### **VIII.1.4. Tạo một đề mục**

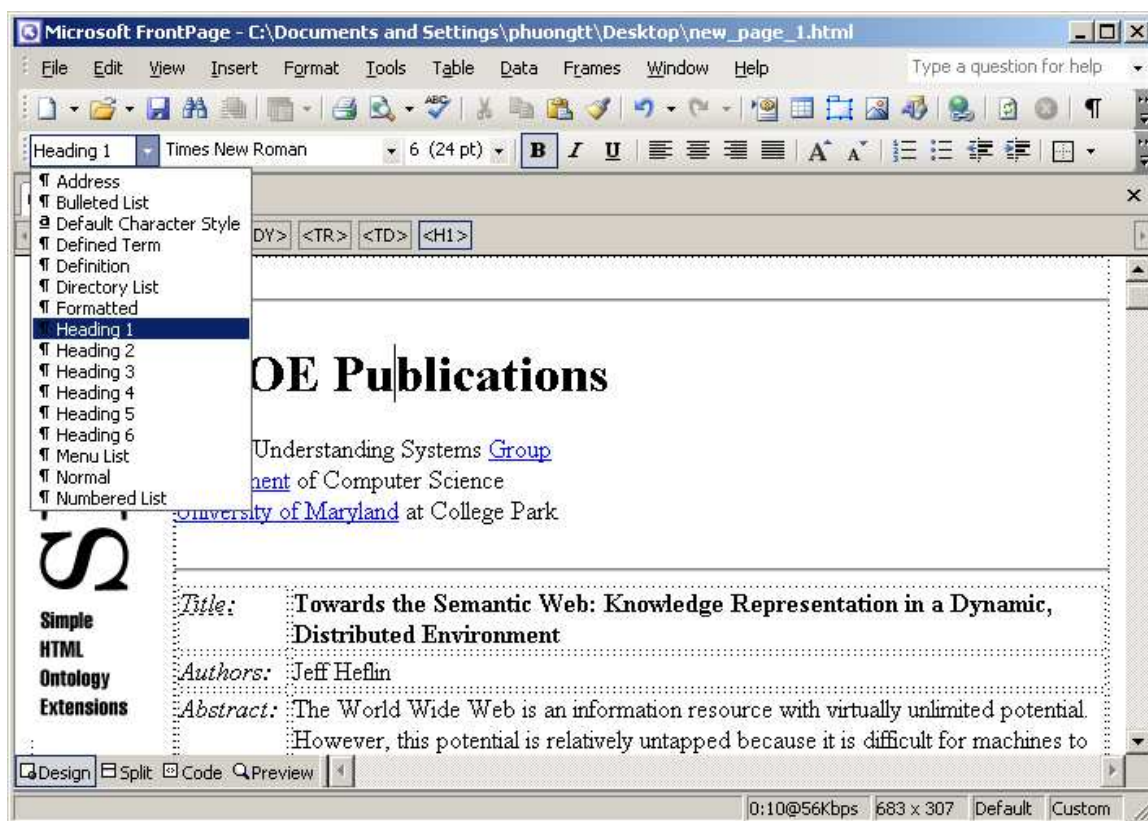
Text trên một trang Web có thể được tách biệt với các text khác bằng cách chuyển đổi nó thành một tiêu đề chính, một đề mục bắt mắt nổi bật với text xung quanh.

Như đã giới thiệu trong phần “Tiêu đề trong HTML” ở trên, các tiêu đề chính được sắp xếp theo kích cỡ từ 1 (lớn nhất) đến 6 (nhỏ nhất), và chúng được sử dụng cho với cùng một mục đích như một đề mục trong một bài báo, một đoạn văn bản,... nào đó. Chúng cũng có thể được dùng làm các tiêu đề con với một mục lớn hơn, làm các đoạn trích dẫn được phóng lớn, và cho những mục đích khác nhằm thu hút sự chú ý.

Để chuyển đổi text thành một tiêu đề chính, thực hiện các bước sau đây:

1. Chọn text bằng cách kéo chuột lên trên nó. Text được bật sáng.
2. Mở menu Style xuống trên thanh công cụ Formatting để chọn một trong sáu kích cỡ tiêu đề chính, như được chứng minh họa trong hình dưới





Hình VIII.4. Hộp thoại Style Formatting

Text được chọn (và bất kỳ text trong cùng một đoạn) sẽ trở thành một tiêu đề chính.

Kích cỡ hiển thị của một tiêu đề chính phụ thuộc vào trình duyệt Web được sử dụng bởi một khách tham quan đến site của ta và cách nó được cấu hình như thế nào. Nhưng theo luật chung, ta có thể tin cậy vào một hệ thống phân loại kích cỡ từ 1 đến 6.

Các tiêu đề chính có thể được sử dụng theo nhiều cách nhất như thể chúng là text. Một điều ngoại lệ là một tiêu đề chính phải chiếm khoảng riêng của nó.

Để thấy trực tiếp được điều này, hãy bật sáng một từ trong một đoạn và chuyển đổi nó thành một tiêu đề chính. Mọi thứ trong đoạn trở thành một tiêu đề chính.

### VIII.1.5. Chọn font và màu

Vào thời điểm này, text mà ta đã thêm vào một trang Web sẽ được hiển thị bằng cách sử dụng cùng một màu và font mặc định như trong Internet Explorer: VNI-Times, 12 point, màu đen.

Ta có thể thay đổi từng khía cạnh này của text (và các khía cạnh khác).

Một font có thể được chọn cho tất cả các text trên một trang, các đoạn cụ thể, thậm chí phần của một đoạn.

Bất kỳ font hiện có trên máy tính của ta có thể sử dụng và sẽ trông tuyệt vời khi ta xem trang Web trên máy tính đó.

Tuy nhiên, những người tham quan đến Web site của ta sẽ không xem trang trên máy tính của ta. Các trang của ta tạo sẽ trông khác biệt với các trang của người khác.

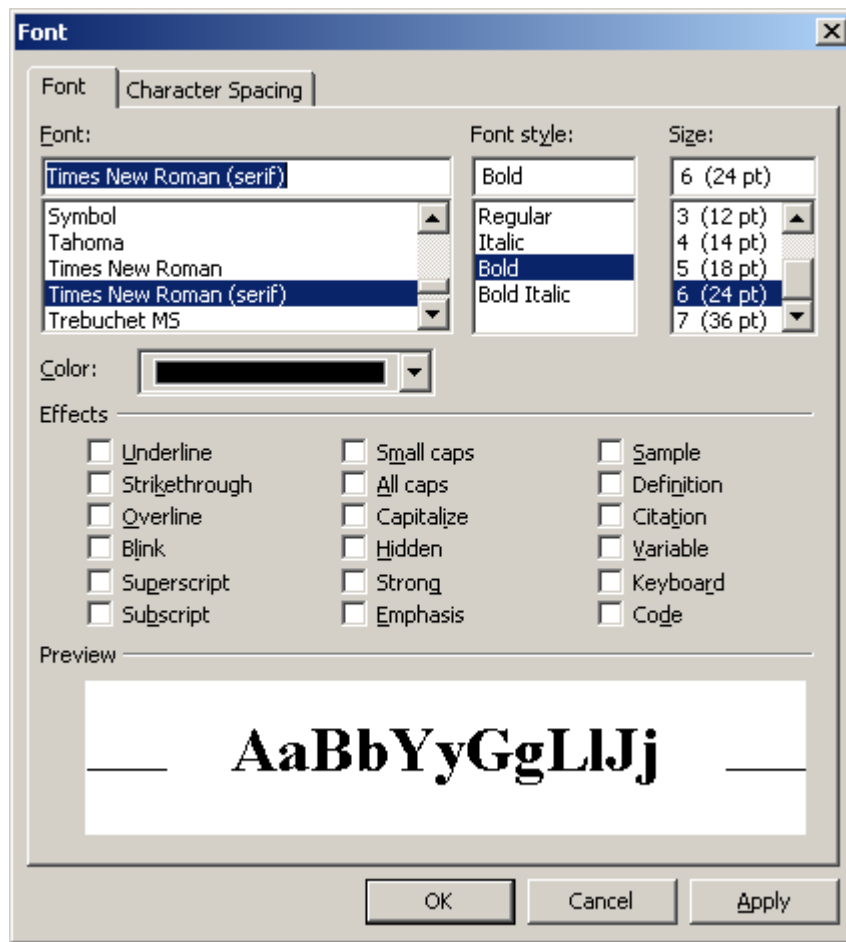
Khi ta đang tạo các trang Web, các font mà ta chọn có thể sẽ có sẵn cho càng nhiều khách tham quan của ta càng tốt. Nhiều font sẽ riêng biệt cho một hệ điều hành cụ thể - những người dùng Windows, Mac, và Linux không chia sẻ nhiều font.

Nếu một font không có sẵn, một font mặc định, chẳng hạn Arial, Helvetica, Times Roman, hoặc Verdana sẽ được sử dụng cho font đó.

Kích cỡ của font có thể được chỉ định làm một kích cỡ point hoặc một tỷ lệ từ 1 (nhỏ nhất) đến 7 (lớn nhất).

Để định dạng một font trên một trang Web mà ta đang biên soạn, hãy thực hiện các bước sau:

1. Chọn đoạn text muốn thay đổi
2. Trên thanh công cụ Formatting, sử dụng menu Font thả xuống để chọn một font cụ thể. Text được hiển thị ngay lập tức bằng font mới. Ta chỉ có thể chọn một font với menu này (xem bước 6 để tìm cách xác định nhiều font)
3. Trên cùng thanh công cụ, sử dụng menu Size thả xuống để xác định một kích cỡ font.
4. Để chọn một màu text khác và thực hiện các thay đổi khác, chọn lệnh Format, Font. Hộp thoại Font sẽ hiện ra (xem hình vẽ dưới)



Hình VIII.5. Hộp thoại Font

5. Sử dụng menu Color thả xuống để chọn một màu cho text. Nếu ta chọn Automatic thay vì một màu, màu mặc định được sử dụng.
6. Để xác định danh sách các font, nhập chúng vào trong hộp danh sách Font, được tách biệt bởi các dấu phẩy.
7. Để xem chức năng của một hiệu ứng cụ thể, hãy chọn hộp kiểm của nó trong khung Effects. Khung Preview hiển thị diện mạo của text mẫu với định dạng được chọn - kể cả hiệu ứng được chọn.
8. Khi ta hài lòng với diện mạo của text, hãy click nút OK.

Bởi vì FrontPage cho ta thấy dáng vẻ của một trang trông như thế nào trong khi nó được biên soạn, ta thấy được ngay lập tức font và style được chọn.

Nếu ta muốn xem một dòng text đã được định dạng như thế nào, hãy click vào nó. Thanh công cụ Formatting sẽ được cập nhật để cho biết những gì mà ta đã áp dụng vào text đó.

## **VIII.2. Tổ chức một trang với các liên kết, danh sách và bảng**

Bây giờ ta đã quen thuộc với việc thêm text vào một trang Web, ta sẽ sẵn sàng thực hiện bước kế tiếp và tổ chức nội dung của trang để trình bày theo nhiều cách khác nhau.

Các danh sách (list) là các nhóm thông tin có liên quan được tách biệt bằng các dấu bullet, số, hoặc những ký hiệu khác.

Các bảng (table) là các hộp chứa text, đồ họa hoặc ngay cả các table nhỏ hơn.

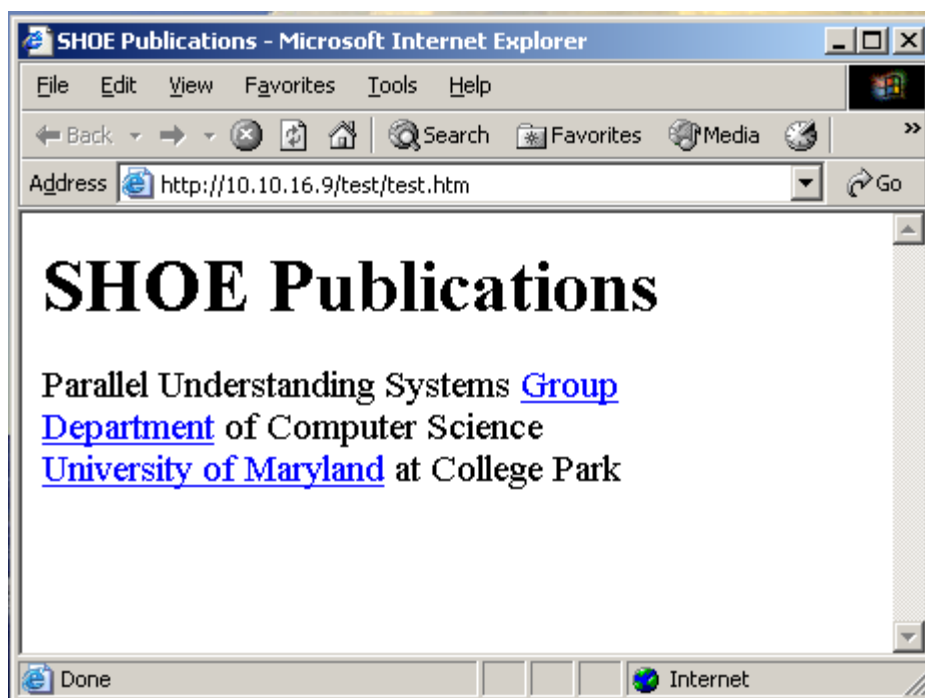
Siêu liên kết (hyperlink) là các liên kết có thể click có thể được sử dụng để tải một trang mới trong một trình duyệt Web.

### ***VIII.2.1. Thêm một hyperlink vào một trang Web***

Các tài liệu trên World Wide Web có thể được nối kết lại với nhau bằng các hyperlink – các vùng có thể click của một trang Web nhằm làm cho một trang mới hoặc một loại file khác được mở trong một trình duyệt Web.

Các hyperlink thường được kết tới text, nhưng điều này không phải là một yêu cầu. Ta có thể đặt một liên kết trên bất kỳ phần nào của một trang Web: các ảnh, nút, applet Java, các phim QuickTime, các file âm thanh MP3.

Các text hyperlink được hiển thị theo một cách nhằm tách biệt chúng với text khác trên một trang. Cách thông thường nhất mà các hyperlink được trình bày trong một trình duyệt Web là bằng cách gạch dưới text được liên kết. Hình dưới minh họa một trang Web chứa ba hyperlink được nhận biết bằng các đường gạch dưới.



*Hình VIII.6. Ví dụ Hyperlink trong trang Web*

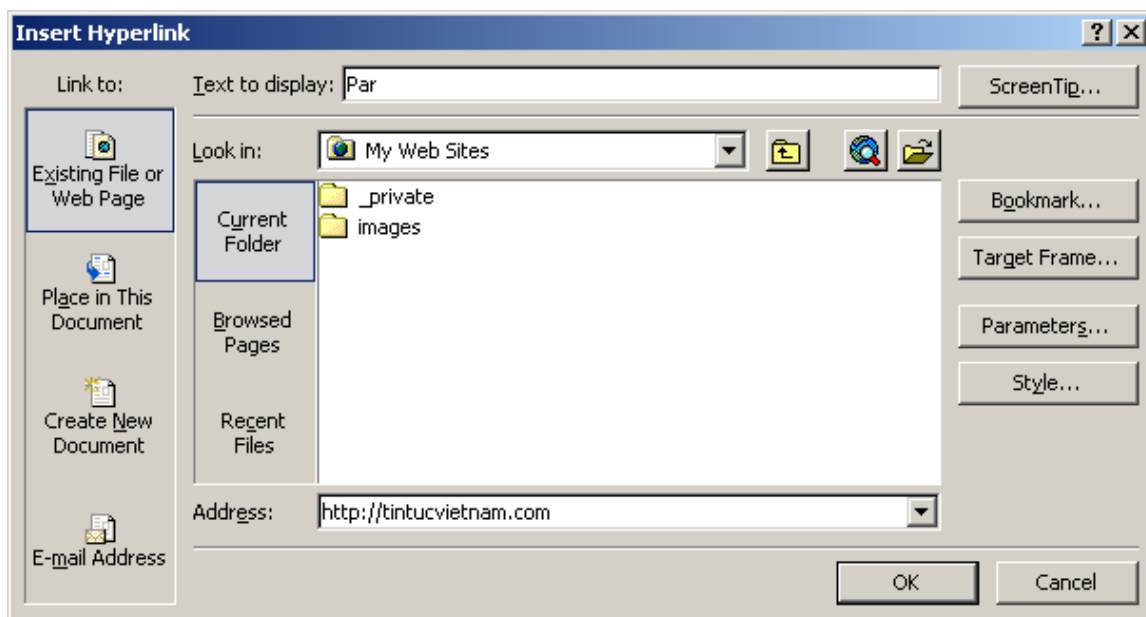
Một hyperlink chứa một URL, đây là dạng viết tắt của Uniform Resource Locator, nhưng có một cách đơn giản hơn nhiều để mô tả chúng: một URL là một địa chỉ dẫn đến một điều nào đó trên Web.

Các trình duyệt Web thường hiển thị địa chỉ của trang mà ta đang xem trong thanh Address của trình duyệt. Một số URL mẫu bao gồm trang chủ Sams Publishing tại <http://www.sampublishing.com>, công cụ tìm kiếm của Google tại <http://www.google.com> /unclesam, và server Knowledge Base FTP của Microsoft tại <http://ftp.microsoft.com>

Tại hyperlink cũng có thể được sử dụng để gửi email mới - những địa chỉ này bắt đầu bằng `mailto:` và theo sau là một địa chỉ email. Ví dụ như `mailto:bernie@mail.house.gov`

Để tạo một hyperlink, thực hiện những bước sau:

1. Trên một trang Web đang mở trong cửa sổ biên soạn, rê chuột lên trên phần của trang mà sẽ được kết hợp với liên kết. Phần đó của trang sẽ được bật sáng để cho thấy nó đã được chọn.
2. Chọn Insert, Hyperlink (hoặc click nút Insert Hyperlink trên thanh công cụ Standard). Hộp thoại Insert Hyperlink xuất hiện (hình vẽ dưới)



*Hình VIII.7. Hộp thoại Insert Hyperlink*

Các hyperlink có thể được kết hợp với các file trên máy tính của ta hoặc bất kỳ địa chỉ trên World Wide Web:

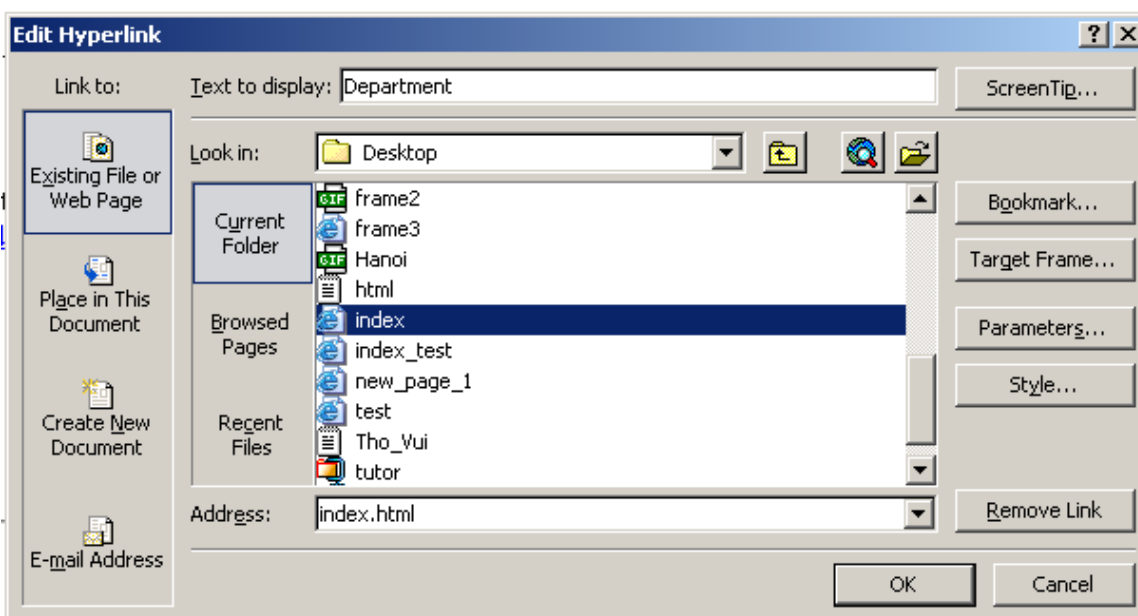
- Nếu ta liên kết với một file trên máy tính của ta, hãy sử dụng hộp thoại để tìm và chọn file đó
- Nếu ta liên kết tới một địa chỉ Web, nhập địa chỉ đó trong trường text Address (hoặc dán nó từ Windows Clipboard, nếu nó đã được sao chép ở đó từ trình duyệt Web). Sau đó click OK.

Nếu hộp thoại Insert Hyperlink đóng lại, ta sẽ quay trở về cửa sổ biên soạn.

Nếu Hyperlink được kết hợp với text, nó sẽ được gạch dưới trên trang Web.

Sau khi thêm một hyperlink vào một trang, ta có thể thay đổi nó bằng cách click chuột phải liên kết. Sau đó trên menu ngữ cảnh vừa xuất hiện, chọn Hyperlink Properties.

Hộp thoại Edit Hyperlink mở ra. Để loại bỏ hoàn toàn một liên kết, click chuột vào Remove Link.



Hình VIII.8. Hộp thoại Edit Hyperlink

### VIII.2.2. Tạo một danh sách

Trong hầu hết các trình duyệt Web, các đoạn text được tách biệt bằng các dòng trắng và được canh thẳng bên lề trái, nhưng không được thụt vào. Một cách khác để tổ chức text là biến đổi nó thành một danh sách.

Trên một trang Web, các danh sách là các nhóm mục liên quan được tách biệt với phần còn lại của trang bằng các số, bullet, hoặc các ký hiệu tương tự. Ta có thể sử dụng hai loại danh sách:

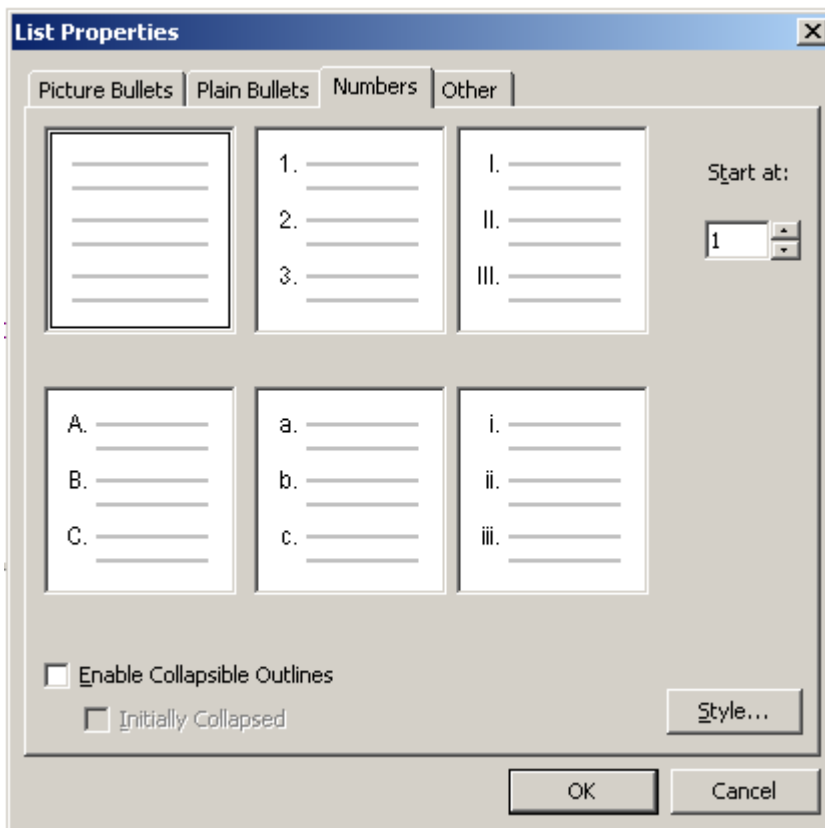
- Các danh sách được đánh số, với mỗi mục có một số duy nhất đứng trước
- Các danh sách không được đánh số, với mỗi mục có một dấu bullet, dấu tròn hoặc một ký tự khác đứng trước.

Text đứng trước là một danh sách hai mục không được đánh số. Ký tự “.” Tương tự như các dấu bullet thường được hiển thị trên trang Web.

Để chuyển đổi các dòng text thành một danh sách, thực hiện những bước sau đây:

1. Chọn text sẽ được chuyển đổi thành danh sách.
2. Click nút Unnumbered List hoặc nút Numbered List trên thanh công cụ Formatting
3. Khi một danh sách được biên soạn, click phím Enter để thêm một mục mới. Một bullet hoặc số xuất hiện trên một dòng mới.

4. Để thay đổi cách một danh sách được trình bày, đặt con trỏ trên một mục danh sách, click chuột phải và chọn List Properties từ menu tắt vừa xuất hiện. hộp thoại List Properties xuất hiện (hình vẽ dưới)



*Hình VIII.9. Hộp thoại List Properties*

Hộp thoại này có thể được sử dụng để xác định một số khởi đầu khác cho một danh sách được đánh số và diện mạo của các bullet trong một danh sách không được đánh số.

Các danh sách có thể được đặt bên trong các danh sách khác. Hình dưới minh họa một vài danh sách được sắp xếp lồng nhau.



## EVENTS CALENDAR

- Jacksonville Jazz Fest
  - 4/11/2003
  - Jacksonville, FL
  - Performers
    - Tony Bennett,
    - Joshua Redman
    - Boney James
- Great Hawaiian Jazz BlowOut
  - 5/3/2003
  - Honolulu, HI
  - Performers
    - Local jazz musicians

### *Hình VIII.10. Ví dụ về danh sách*

Như được minh họa ở trên, các bullet hiển thị kế bên các mục trong một danh sách không được đánh số cung cấp một gợi ý về danh sách mà chúng thuộc về danh sách đó. Nếu một danh sách không được đánh số được đặt bên trong một danh sách khác thì hai danh sách có các kiểu bullet khác nhau.

Để đặt một mục danh sách bên trong một danh sách khác, bật sáng mục và click nút Increase Indent (trên thanh công cụ) hai lần.

Để đẩy một mục ra khỏi một danh sách khác, bật sáng nó và nhấp nút Decrease Indent hai lần.

Ta có thể đặt một danh sách vào sâu bên trong một danh sách khác bao nhiêu cấp tùy ý.

### **VIII.2.3. Tổ chức một trang với các bảng**

Một trong những điều làm bối rối những nhà thiết kế trang Web lần đầu tiên là trạng thái hay thay đổi của một trang Web. Text, đồ họa và những phần khác hay thay đổi của một trang Web di chuyển phụ thuộc vào cách chúng được trình bày. Cùng một trang có thể trông khác nhau đáng kể ở trong hai trình duyệt khác nhau trên các máy tính khác nhau. Thậm chí nó có thể thay đổi diện mạo trên một máy tính nếu cửa sổ trình duyệt được định lại kích cỡ.

Những người thiết kế Web có thể kiểm soát được diện mạo của các thành phần trang một cách nhiều hơn bằng cách đặt chúng trong các bảng.

Các bảng là các lưới hình chữ nhật được chia thành các ô riêng biệt. Thông tin có thể được đặt vào từng ô này để canh thẳng chúng theo chiều dọc hay chiều ngang với thông tin trong các ô khác.

Nếu ta cảm thấy khó hình thành khái niệm về một bảng khi nó có liên quan đến một trang Web, hãy nghĩ đến một lịch treo tường:

SUN	MON	TUE	WED	THU	FRI	SAT
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30			

*Hình VIII.11. Sử dụng bảng để sắp xếp lịch*

Một lịch giống như trên là một bảng hình chữ nhật chứa một nhóm ô.

Trên một lịch treo tường, mỗi ngày chiếm một ô riêng của nó trong bảng. Tên của mỗi ngày từ SUN đến SAT, cũng chiếm ô riêng của nó.

Các bảng được chia thành các cột dọc và các hàng ngang. Lịch treo tường được minh họa có bảy cột và sáu hàng.

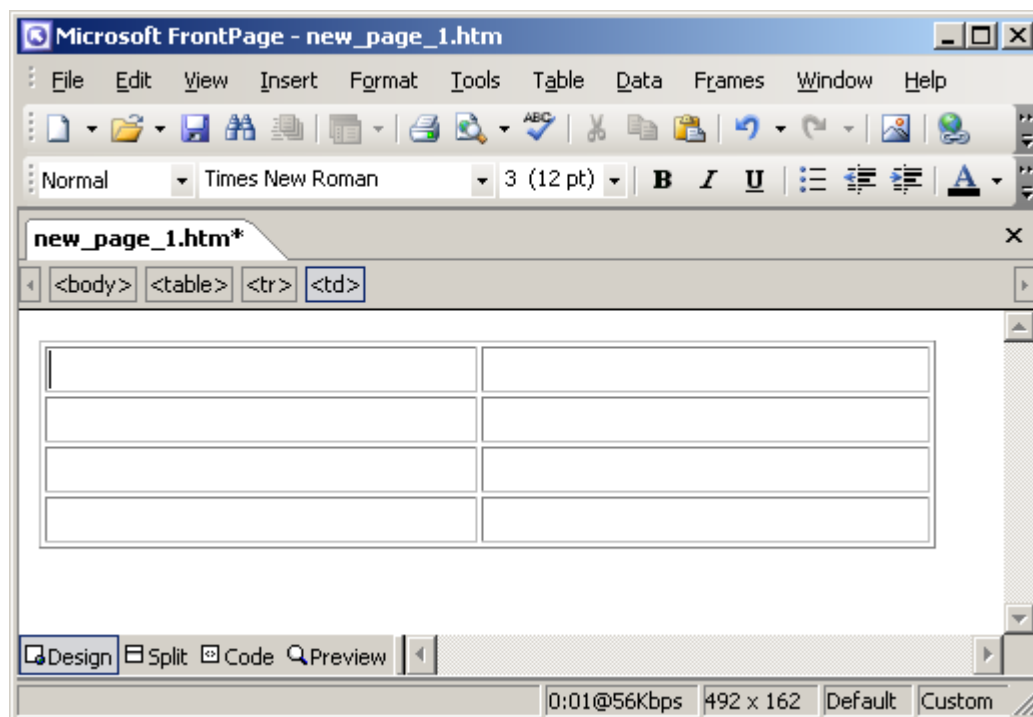
Mục đích chính của các bảng là tổ chức thông tin vốn phải được căn thẳng với các hàng và các cột. Ta có thể sử dụng các bảng để hiển thị dữ liệu chẳng hạn như báo cáo chi phí trong các cột để đọc.

### **Căn thẳng text trong bảng**

Mặc dù các bảng hữu dụng cho việc trình bày trang, nhưng chúng cũng là một cách hiệu quả để trình bày text trong các hàng và cột dạng bảng.

Bởi vì ta có thể tạo một bảng với FrontPage, ta phải quyết định bao nhiêu hàng và cột mà nó sẽ chứa, các cột được đếm từ trái sang phải và các hàng được đếm từ trên xuống dưới.

Các bảng được hiển thị dưới dạng một lưới rỗng khi chúng được thêm vào một trang Web. Hình dưới minh họa một bảng mới được tạo chứa hai cột và bốn hàng.



*Hình VIII.12. Tạo bảng*

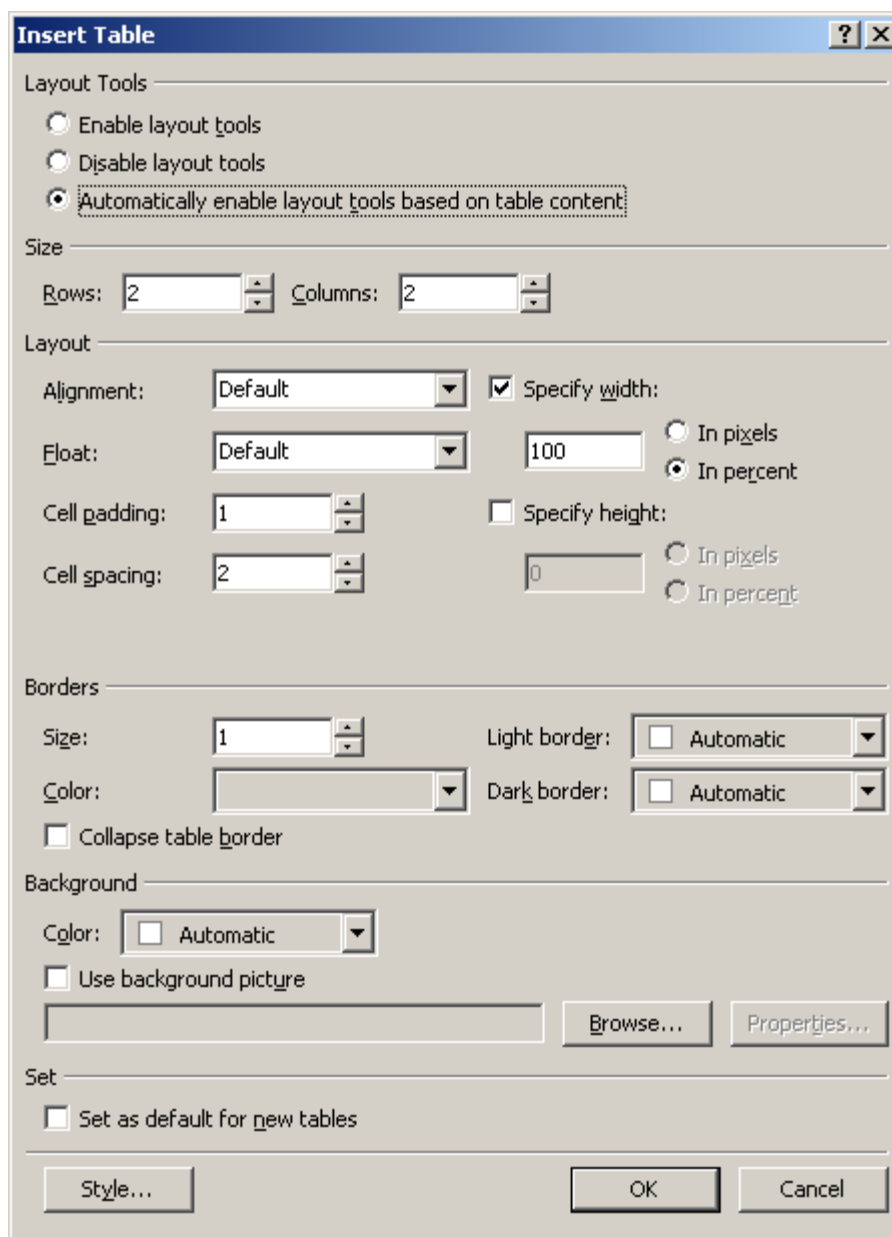
Số hàng và số cột trong một bảng quyết định số ô ban đầu mà nó chứa. Bảng ở hình trên chứa 8 ô.

### **Thêm một bảng vào một trang**

Một bảng rộng khi nó được tạo trên một trang Web. Sau đó ta điền vào các ô riêng lẻ của nó.

Để thêm một bảng vào một trang, thực hiện những bước sau đây:

1. Với các trang mở trong cửa sổ biên soạn, click vào nơi mà bảng sẽ được chèn.
2. Trên thanh menu chọn Table → Insert → Table. Hộp thoại Insert Table mở ra như được minh họa ở hình dưới



Hình VIII.13. Hộp thoại Insert Table

- Trong phần Size, sử dụng các hộp danh sách Rows and Columns để xác lập kích thước của bảng. Lựa chọn mà ta thực hiện không nhất thiết phải cố định. Ta có thể thêm và bớt các hàng và các cột ra khỏi bảng khi ta làm việc với nó. Do đó, các giá trị ban đầu không quan trọng. Phần Size xác lập các hàng, cột, và số ô trong bảng, nhưng nó không quyết định bao nhiêu khoảng trống mà table chiếm trên trang Web khi nó hiển thị. Một bảng thường chiếm càng nhiều phần của trang càng cần thiết để hiển thị nội dung của các ô.

4. Để tạo lượng khoảng trống mà một số ô sẽ chiếm, chọn hộp kiểm Specify Width và chọn một đơn vị đo:
  - a. Đối với một chiều rộng cụ thể, chọn một tùy chọn In Pixels và nhập chiều rộng mà ta muốn vào trường text nằm gần kề
  - b. Để xác lập chiều rộng dưới dạng tỷ lệ phần trăm của khoảng trống có sẵn trên trang (được đo từ cạnh này sang cạnh kia), chọn tùy chọn In Percent và gõ nhập một tỷ lệ phần trăm từ 1 đến 100 vào trường text nằm gần kề.
  - c. Nếu ta chọn một chiều rộng 100%, bảng sẽ chiếm tất cả mà nó có thể chiếm.
5. Để xác lập chiều cao của bảng, lặp lại bước 4 với trường Height.

Có thêm ba cách để tinh chỉnh một bảng là xác lập các giá trị đường viền, khoảng cách đệm ô và khoảng cách ô của nó.

Đường viền (border) xác định kích thước của đường viền bao quanh bảng. Nếu nó được xác lập sang 0, đường viền và tất cả đường lưới của nó sẽ biến mất. Các ô của bảng sẽ vẫn căn thẳng một cách chính xác, nhưng sẽ không rõ ràng cho thấy rằng một bảng đang được sử dụng trên trang.

Khoảng cách đệm ô (cell padding) là lượng khoảng trống bao quanh nội dung của mỗi ô. Nếu ta tăng padding từ giá trị mặc định là 1, các ô sẽ trở nên lớn hơn trong khi nội dung của nó sẽ vẫn giữ cùng một kích cỡ.

Khoảng cách ô (cell spacing) là lượng khoảng trống trong đường viền lưới giữa ô. Điều này làm cho chiều rộng và chiều cao của các trường lưới trở nên lớn hơn, nếu đường viền có thể nhìn thấy được. Khi khoảng cách đường tăng, bảng mở rộng trong khi các ô vẫn giữ cùng một kích cỡ.

1. Trong hộp thoại Insert Table, sử dụng các hộp danh sách Border, Cell Padding, và Cell Spacing để xác lập các giá trị này khi cần thiết.
2. Khi ta hoàn tất việc xác lập bảng, click nút OK.

### **Thêm dữ liệu vào một bảng**

Khi ta có một bảng trên một trang, ta có thể thêm các text vào bất kỳ ô của nó: click con trỏ trong một ô và bắt đầu nhập. Ta cũng có thể sử dụng các lệnh cắt, sao chép và dán hoặc rê và thả để điền đầy một ô.

Các bảng bắt đầu với tất cả ô và hàng có cùng một kích cỡ và FrontPage cố gắng làm cho chúng có cùng một kích cỡ khi ta thêm text. Từ bao bọc xung quanh mép phải của một ô như thể nó nằm trên lề phải của một trang.

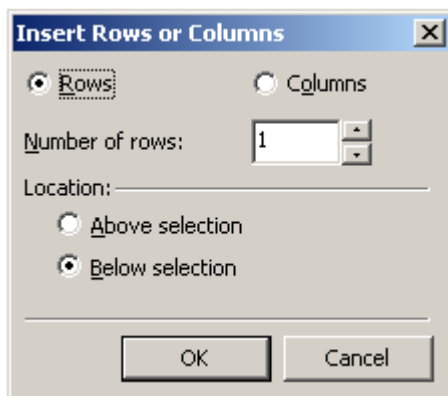
Khi ta điền đầy một bảng, nhấn phím Tab để nhảy đến ô kế tiếp nằm bên phải (hoặc hàng kế tiếp) hoặc nhấn các phím Shift và Tab cùng một lúc để di chuyển theo hướng ngược lại.

Một điều khác thường xảy ra nếu ta nhấn Tab khi ta ở ô sau cùng trong bảng (ô ở hàng dưới cùng và cột nằm ở tận cùng bên phải): FrontPage tạo một hàng mới và di chuyển con trỏ vào ô đầu tiên trên hàng này.

Điều này cho phép ta tiếp tục thêm dữ liệu mới vào một bảng ngay cả nếu ta đánh giá thấp số hàng mà ta cần khi nó được tạo.

Để thêm một hoặc nhiều cột hoặc hàng vào một bảng, thực hiện các bước sau đây:

1. Click một ô nằm kề nơi các ô mới sẽ được chèn vào.
2. Trên thanh menu chọn Table → Insert → Rows or Columns. Hộp thoại Insert Rows or Columns xuất hiện (hình dưới)



Hình VIII.14. Hộp thoại Insert Rows or Columns

3. Chọn các tùy chọn Columns hoặc Rows
4. Chọn số cột hoặc hàng để chèn trong hộp danh sách Number Of. Chúng được thêm ở kế bên ô được chọn ở bước 1.
5. Trong phần Position, chọn một tùy chọn để quyết định chính xác nơi mà các cột hoặc hàng mới sẽ được đặt ở đó.
6. Click OK.

## **Loại bỏ các hàng hoặc cột ra khỏi bảng**

Các hàng và cột cũng có thể bị xóa sau khi ta đã chọn chúng:

1. Đặt con trỏ lên trên đường viền bảng bên ngoài của một hàng hoặc cột mà ta muốn xóa. Di chuyển con trỏ xung quanh đường viền đó đến khi nó thay đổi thành một mũi tên nhỏ màu đen trỏ vào bảng.
2. Click một lần. Hàng hoặc cột theo hướng của mũi tên được bật sáng.
3. Để xóa nó, nhấn phím Delete hoặc click chuột phải vào vùng được bật sáng và chọn Delete Rows hoặc Delete Columns từ menu tắt vừa xuất hiện.

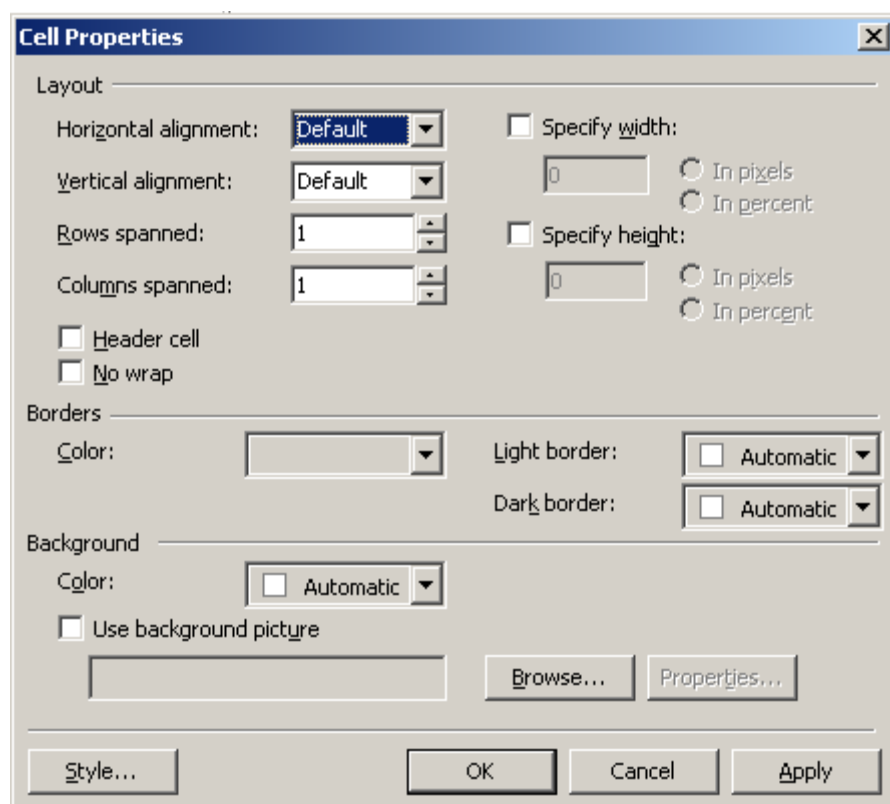
## **Thay đổi kích cỡ của một bảng**

Theo luật chung, FrontPage định kích cỡ của một bảng để tất cả các ô chiếm cùng một lượng khoảng trống trừ khi một số trong chúng chứa text quá lớn đến nỗi không thể thực hiện được điều này.

Một cách dễ dàng để định dạng một bảng sao cho nó chiếm ít khoảng trống hơn là thu nhỏ nó để nó chiếm lượng khoảng trống tối thiểu cần thiết: click ở bất cứ nơi nào trên bảng và chọn (trên thanh menu) Table → AutoFit to Contents.

Ta cũng có thể định lại kích cỡ một bảng để các hàng và cột cụ thể có chiều rộng pixel hoặc tỷ lệ phần trăm cụ thể:

1. Click chuột phải vào một trong các ô trong hàng hoặc cột đó.
2. Trong menu tắt vừa mở ra, chọn lệnh menu Cell Properties. Hộp thoại Cell Properties xuất hiện, hộp thoại này có thể được sử dụng để xác lập chiều rộng của một ô theo cùng một cách như một bảng được cấu hình.



*Hình VIII.15. Hộp thoại Cell Properties*

3. Để xác lập chiều rộng của ô đó, chọn hộp kiểm Specify Width, chọn In Pixels hoặc In Percent, và sau đó nhập một giá trị vào trường text nằm gần kề.
4. Đối với chiều cao, chọn Specify Height và lặp lại các hướng dẫn ở bước 3.
5. Nếu ta muốn loại bỏ một giá trị chiều cao hoặc chiều rộng cho một ô, xóa dấu kiểm ra khỏi hộp Specify Height hoặc hộp Specify Width. Nếu các ô khác trong hàng hoặc cột đó không được xác lập các giá trị, tất cả các ô được hiển thị với cùng một kích cỡ.
6. Click OK.

FrontPage có thể sử dụng các kích thước mới của ô để quyết định các ô khác trong cùng một hàng và cột sẽ được định kích cỡ như thế nào.

Luật chung sau đã được áp dụng: Nếu ô là ô lớn nhất trong hàng hoặc cột của nó, thì các ô khác sẽ được mở rộng để có cùng một kích cỡ.

Khi ta xác lập được kích thước cho một ô bảng cụ thể, ta không nên sử dụng lại Table → AutoFit to Contents nếu không ta sẽ hủy những thay đổi của ta.



#### **VIII.2.4. Hiện thị hình ảnh trên một trang Web**

Làm việc với các ảnh theo các dạng khác nhau

Trước khi ta bắt đầu làm việc với hình ảnh đồ họa, điều quan trọng là ta phải biết loại nào của các định dạng ta nên sử dụng. Sự minh họa bằng hình ảnh có thể được biểu diễn trên một máy tính theo hàng chục dạng khác nhau nhưng người thiết kế Web cần quen thuộc với ba dạng: GIF, JPEG, và PNG.

Ta đã có một vài trong số các file này trên máy tính của Graphic Interchange Format (GIF) hoặc Joint Photographic Experts Group (JPEG). Một dạng mới hơn mà nó đang trở nên phổ biến là Portable Network Graphics (PNG).

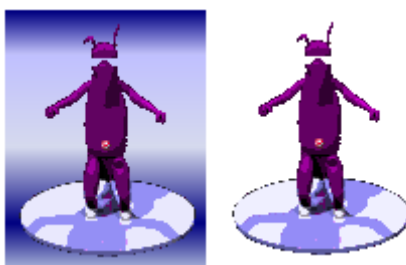
##### Dạng GIF

Dạng GIF chứa các ảnh được giới hạn chỉ 256 màu. Dạng này lý tưởng cho các ảnh có nhiều màu đồng nhất (các nút menu), các ảnh đồ họa nhỏ (các biểu tượng, quảng cáo), và những hình ảnh khác vốn không đòi hỏi chi tiết rõ nét.

Nếu một ảnh chụp được hiển thị dưới dạng GIF, đầu tiên nó phải được làm giảm sao cho không quá 256 màu xuất hiện khác nhau trong ảnh.

Dạng GIF hỗ trợ hai hiệu ứng đặc biệt thường thấy trên Web: độ trong suốt và hoạt hình.

Độ trong suốt (transparency) là một kỹ thuật nhằm làm cho một phần của một ảnh hòa hợp với nền của trang, nền có thể là một màu đồng nhất hoặc một ảnh. Độ trong suốt sẽ làm việc bằng cách chỉ định một màu trong một ảnh GIF làm màu trong suốt, màu này sẽ không được hiển thị khi ảnh được hiển thị trên một trang Web. Để thấy được trực tiếp điều này, hãy xem hai ảnh GIF trên một trang trong hình dưới



*Hình VIII.16. Ảnh trong suốt và ảnh không trong suốt*

Một ảnh GIF có thể được tạo đồng bằng cách hiển thị một số ảnh GIF có liên quan theo trình tự. Những ảnh này được lưu trữ cùng với nhau trong một file đơn với thông tin về khoảng thời gian bao lâu để hoàn thành một ảnh, thứ tự của sự hiển thị, và số lần để lặp lại qua trình tự.

Chắc ta đã thấy hàng trăm ảnh động khi ta duyệt Web; chúng cũng là một công nghệ mà các nhà quảng cáo ưa thích.

### Dạng JPEG

Dạng JPEG chứa các ảnh có chất lượng ảnh chụp vốn có thể chứa hàng ngàn màu khác nhau. Để làm cho kích cỡ file trở nên hợp lý, làm cho ảnh tải nhanh hơn trên Web hoặc được truyền tải bởi phương tiện khác, JPEG sử dụng một kỹ thuật nén dữ liệu nhằm làm cho kích cỡ file nhỏ hơn nhưng bù lại chất lượng ảnh bị mất.

Khi một file JPEG được tạo ra bởi một camera kỹ thuật số, máy quét hoặc phần mềm, thì người ta cần đạt được sự cân bằng giữa kích cỡ và chất lượng. Độ rõ nét và chiều sâu ảnh càng cao thì kích cỡ của file càng lớn.

Do nén, JPEG thường là lựa chọn thích hợp nhất cho các ảnh phức tạp với số màu lớn. Các file JPEG thích hợp đối với việc hiển thị các ảnh chụp được quét vốn không có các vùng màu đồng nhất.

JPEG thường là lựa chọn kém cho các ảnh với các vùng lớn có một màu đơn. Do cách xử lý việc nén, các đường gợn sóng (thường được gọi là “răng cưa”) sẽ xuất hiện trông mờ hơn.

### Dạng PNG

Dạng PNG đã được giới thiệu để thay thế và cải tiến dạng GIF và JPEG. Nó có thể được sử dụng để hiển thị các ảnh có 256 màu hoặc ít hơn, giống như một dạng GIF (dạng PNG-8), các ảnh có hàng ngàn màu, giống như một dạng JPEG (dạng PNG-24). Các ảnh PNG cũng có thể hỗ trợ độ trong suốt và các hiệu ứng đặc biệt khác.

Những nhà thiết kế Web site đã thông thả trong việc chọn PNG bởi vì nó không luôn được hỗ trợ trong các trình duyệt Web. Internet Explorer 4 và Netscape Navigator 4 những ấn bản đầu tiên của một trong hai chương trình này vốn có thể hiển thị các ảnh đồ họa PNG mà không cần trợ giúp của một plug-in (một chương trình mở rộng các tính năng của một trình duyệt).

Ngày nay, các phiên bản hiện hành của năm trình duyệt thông dụng nhất – Internet Explorer, Netscape Navigator, Mozilla, Opera, và Safari - hỗ trợ PNG, mặc dù hầu hết các trình duyệt không hỗ trợ tất cả các tính năng của nó.

### Chọn dạng thích hợp

Bằng cách sử dụng FrontPage, ta có thể thêm các ảnh đồ họa sang một site theo nhiều dạng khác mà ta có thể quen thuộc: BMP (Windows Bitmaps), EPS (Encapsulated PostScript), RAS (Raster), TGA (Truevision Targa Graphics Adapter), Tiff (Tagged Image File), và WMF (Windows Meta File).

Khi ta thêm một trong các ảnh này vào một trang và sau đó lưu nó, FrontPage sẽ chuyển đổi nó sang dạng GIF nếu nó chứa 256 màu hoặc ít hơn, hoặc dạng JPEG nếu nó chứa nhiều hơn.

Phần mềm khuyến khích ta sử dụng GIF hoặc JPEG.

Quy tắc chung là sử dụng JPEG cho các ảnh chụp và sử dụng GIF hoặc PNG cho mọi thứ khác. Khi một ảnh JPEG phức tạp hoặc lớn đến mức kích cỡ file của nó là 30KB hoặc lớn hơn, ta hãy làm cho ảnh trở nên đơn giản hơn hoặc thay thế nó bằng một ảnh khác. Điều này ngăn ảnh chiếm một lượng thời gian quá nhiều để xuất hiện khi một người dùng Web xem nó trên một trang bằng một môi liên kết modem quay số.

### **Thêm một ảnh vào một trang**

Bây giờ ta đã biết đôi chút về các ảnh đồ họa, ta sẵn sàng đặt chúng lên trên các trang Web riêng của ta.

Trong FrontPage, hình ảnh được thêm vào một trang Web khi nó đang được hiệu chỉnh. Ta có thể thực hiện điều này bằng nhiều cách: Ta có thể rê một file từ một thư mục hoặc Windows Explorer; sao chép nó sang Clipboard và dán nó trên trang; chọn Insert → Picture → From File; hoặc nhấp một nút.

Tùy chọn sau cùng là dễ nhất, do đó nó được sử dụng trong phần này.

Để thêm một ảnh vào một trang Web, hãy thực hiện các bước sau:

1. Mở trang mà ta muốn hiệu chỉnh
2. Đặt con trỏ tại vị trí trên trang nơi ảnh sẽ hiển thị
3. Trên menu hệ thống, chọn Insert → Picture → From File trên thanh công cụ Standard. Hộp thoại Picture xuất hiện. Sử dụng hộp thoại này để tìm thư mục chứa ảnh. Ta không thể nhớ tên của ảnh mà ta muốn phải

không ? Hãy click vào bên phải của nút Views và chọn Thumbnails từ menu bật lên vừa xuất hiện. Các ảnh có kích cỡ thu nhỏ của mỗi hình ảnh sẽ được hiển thị, như được minh họa trong hình dưới.



*Hình VIII.17. Chọn ảnh*

4. Chọn ảnh và click insert. File được hiển thị như một phần của trang trong cửa sổ hiệu chỉnh, cho phép ta xem diện mạo của nó ngay lập tức.

Khi một ảnh được đặt trên một trang, phần nội dung trang sẽ di chuyển để dành chỗ trống. Text và các thành phần trang khác chảy xung quanh hoặc bên dưới trang. Để di chuyển ảnh, nhấp kéo rê nó đến một vị trí mới.

Mặc dù dường như ảnh đã được trộn với trang Web, nhưng nó vẫn nằm trong file riêng của nó – file mà ta đã chọn bằng hộp thoại Picture.

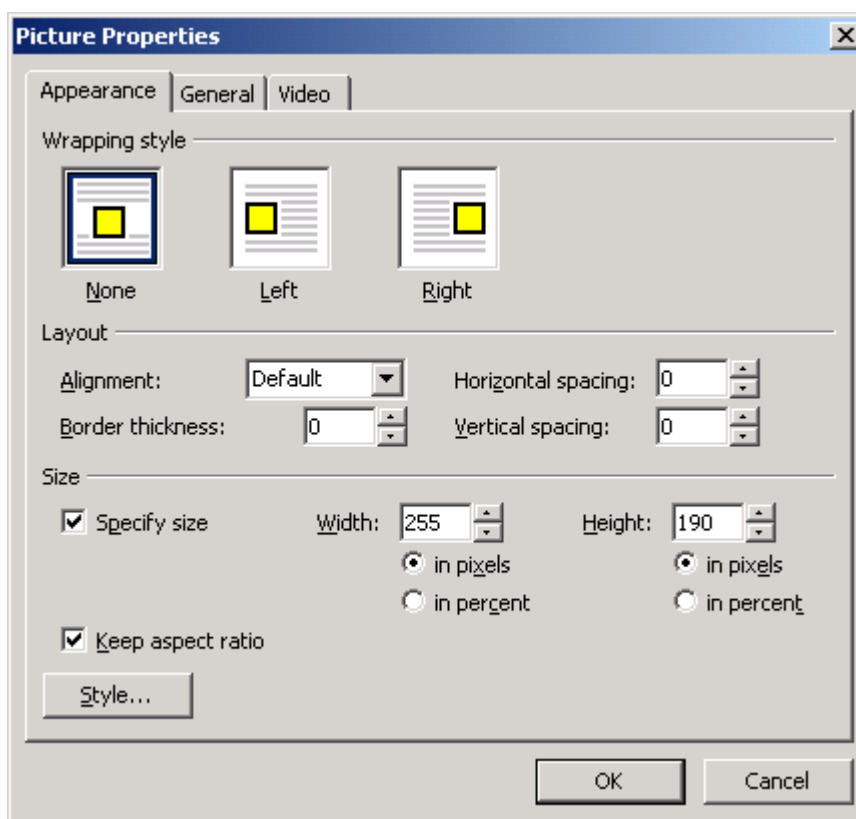
### **Trình bày hình ảnh với text**

Lần đầu tiên ta thêm một ảnh vào một phần của text trong FrontPage, có thể ta không hài lòng với diện mạo của nó. Text chạy một cách vụng về từ mép của ảnh, để lại nhiều khoảng trống xung quanh nó.

Việc chọn một kiểu bao bọc (wrapping style) mới cho ảnh sẽ xử lý vấn đề này. Kiểu bao bọc là một xác lập click để chọn ảnh sẽ được hiển thị như thế nào tương ứng với text nằm gần kề và nội dung khác trên trang.

Để chọn kiểu bao bọc ảnh, hãy thực hiện các bước sau:

1. Click đúp vào ảnh. Hộp thoại Picture Properties mở ra (xem hình dưới)



*Hình VIII.18. Hộp thoại Picture Properties*

2. Nếu tab Appearance không xuất hiện, click tên tab đó để đưa nó lên phía trên.
3. Chọn một trong các tùy chọn nằm bên dưới tiêu đề chính Wrapping Style – click biểu tượng nằm trên các nhãn None, Left, hoặc Right. Các biểu tượng cho biết cách ảnh sẽ được đặt như thế nào tương ứng với text.
4. Để thêm hoặc giảm lượng khoảng trống giữa các cạnh của ảnh và text, hãy điều chỉnh xác lập Horizontal Spacing.
5. Để bổ sung hoặc loại bỏ một số khoảng trống nằm trên đỉnh và đáy, hãy điều chỉnh phần Vertical Spacing.

6. Click OK. Khi hộp thoại Picture Properties đóng lại và ta sẽ thấy kết quả trên trang của ta.

Kiểu bao bọc của một ảnh chỉ quyết định cách nó được hiển thị bên text nằm gần kề.

Để thêm một khoảng dòng trống trước và sau một ảnh, hãy đặt con trỏ nằm bên trái hoặc bên phải của ảnh và nhấn Enter để chèn một ngắt đoạn. Để chèn một ngắt dòng nhỏ hơn, hãy nhấn Shift + Enter.

Một cách khác để tạo tùy biến vị trí của một ảnh là sử dụng xác lập căn chỉnh của nó.

Xác lập này tương tự như kiểu bao bọc, quyết định cách một ảnh sẽ được căn thẳng kế bên các ảnh và text khác có chiều bao bọc.

Để xác lập kiểu căn chỉnh của một ảnh, hãy thực hiện các bước sau:

1. Click đúp vào ảnh. Hộp thoại Picture Properties mở ra.
2. Nếu tab Appearance không xuất hiện, click tên của tab để đưa nó lên phía trên.
3. Bên dưới tiêu đề chính Layout, chọn một trong các tùy chọn của hộp danh sách Alignment:
  - a. Left alignment và Right alignment – làm cho ảnh xuất hiện phía bên trái hoặc bên phải của text xung quanh.
  - b. Top alignment – căn thẳng các mép đỉnh của ảnh và text. Bottom alignment – căn thẳng các mép đáy của ảnh và text. Đây cũng là kiểu căn chỉnh mặc định cho một ảnh mới.
  - c. Middle alignment – căn thẳng phần chính của ảnh so với đáy của text.
  - d. Absolute Middle alignment – căn thẳng phần chính giữa của ảnh với phần chính giữa của text.

Có nhiều tùy chọn khác, nhưng tất cả chúng tương tự như sáu tùy chọn này. Ta có thể sử dụng các tùy chọn căn chỉnh text, hình ảnh, hoặc bất cứ những gì khác vốn đủ nhỏ để hiển thị.

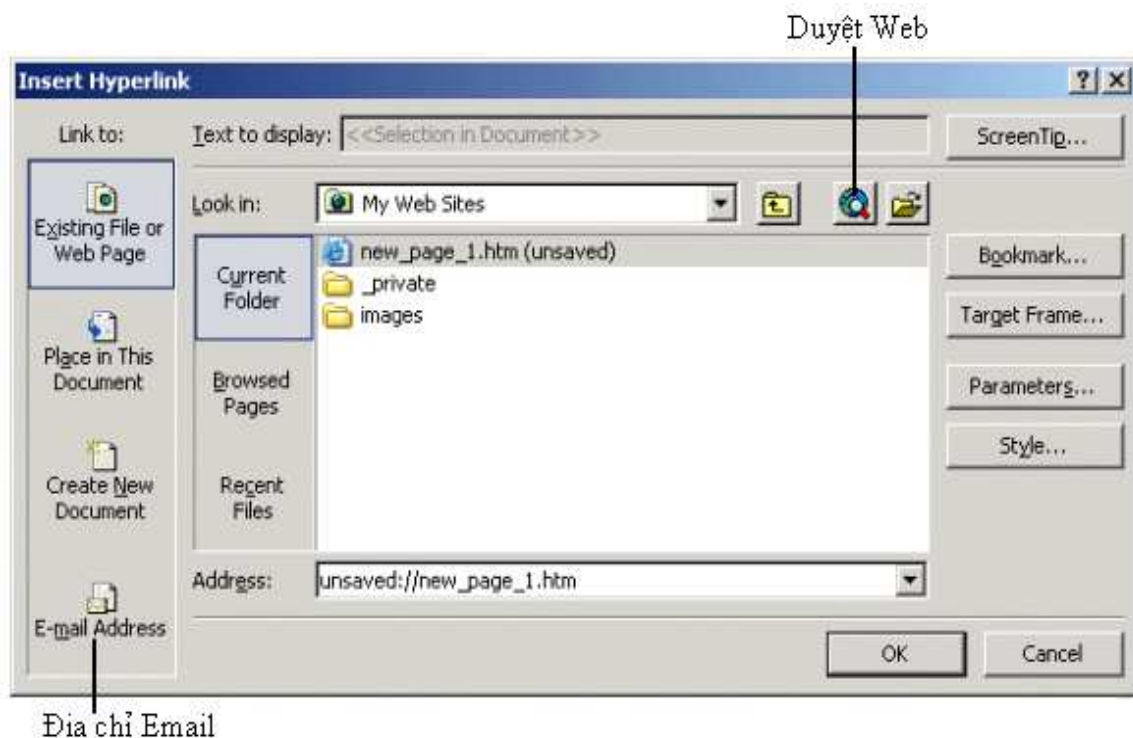
4. Click OK.

## **Thêm một liên kết (Hyperlink)**

Bất kỳ text hoặc đồ họa ở trên một trang Web có thể có một hyperlink được kết hợp với nó. Liên kết này có thể trở vào một trang hoặc một file trên cùng một Web site, một site trên Web, hoặc bất kỳ nguồn tài nguyên khác vốn có một địa chỉ internet.

Để thêm một hyperlink vào một ảnh, hãy thực hiện các bước sau:

1. Click vào ảnh. Các nút nhỏ xuất hiện tại các góc và cạnh của ảnh để biểu thị rằng nó đã được chọn cho việc hiệu chỉnh.
2. Click nút Insert Hyperlink trên thanh công cụ Standard. Hộp thoại Insert Hyperlink mở ra, như được minh họa ở hình dưới. Hộp này có thể được sử dụng để chọn nhiều loại liên kết:
  - a. Nếu liên kết dẫn đến một trang Web hoặc file khác trên máy tính của ta, sử dụng hộp thoại này để tìm và nhập file đó. Tên của nó xuất hiện trong trường Address.
  - b. Nếu liên kết là một địa chỉ email, click biểu tượng Email Address trong khung Link To. Hộp thoại sẽ hiển thị trường Email Address và trường subject. Điền thông tin vào những trường này.
  - c. Nếu liên kết là một địa chỉ Web khác, nhập nó vào trường Address (hoặc sao chép địa chỉ từ thanh Address của trình duyệt Web và dán nó vào trường).



Hình VIII.19. Hộp thoại Insert Hyperlink

3. Để mở trình duyệt và tìm đúng địa chỉ, click nút Browse the Web (hình trên).
4. Click OK.

### Thêm một chú thích

Một trong những cách để làm cho các trang Web của ta trở nên khả dụng hơn là cung cấp cho mỗi ảnh một chú thích nhằm mô tả ảnh.

Khi một trang đang được tải xuống, một số trình duyệt Web hiển thị chú thích của một ảnh trong vùng được chiếm bởi ảnh đó.

Nếu ảnh đang được sử dụng là một nút menu hoặc cho một số mục đích định hướng nào đó, chú thích cho phép những người dùng của ta tận dụng nó trước khi ảnh được tải xuống. Những người sử dụng một kết nối Internet chậm (56.6K hoặc thấp hơn) sẽ cảm kích cách cư xử tao nhã này, đặc biệt nếu ảnh lớn.

Các nội dung mô tả text cũng là cách duy nhất mà một trình duyệt Web chỉ text chẳng hạn như Linux có thể làm cho bất kỳ ảnh trở nên có ý nghĩa. Nếu một ảnh phải được nạp để hướng Web của ta, thì nó nên có text để mô tả mục đích của nó.



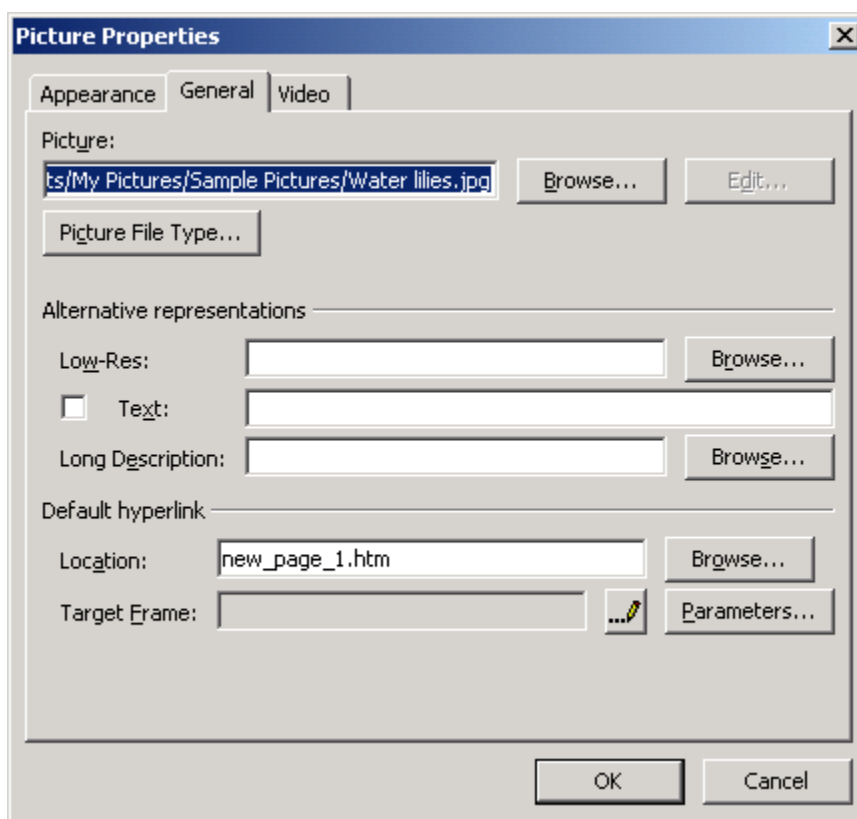
Bằng cách cung cấp text này, ta cung cấp thêm thông tin về các nội dung của trang mà các công cụ tìm kiếm có thể tận dụng. Dịch tìm kiếm Google Images tại [images.google.com](http://images.google.com), hiển thị các ảnh phù hợp với một hoặc nhiều từ khóa, tận dụng các chú thích.

Ta cũng cung cấp thông tin cần thiết cho những người dùng bị khuyết tật để sử dụng Web site của ta, nhằm tăng khả năng truy cập.

Mục đích truy cập, là một trong những chủ đề nóng bỏng nhất của những nhà thiết kế Web ngày nay, là bảo đảm rằng Website có thể được sử dụng với các bộ đọc màn hình đồ họa công nghệ hỗ trợ khác. Bằng cách cung cấp chú thích cho mỗi ảnh - đặc biệt các ảnh được kết hợp các hyperlink - ta mở rộng những người xem tương lai đối với một site.

Để thêm một chú thích vào một ảnh hoặc hiệu chỉnh một chú thích hiện có, hãy thực hiện các bước sau:

1. Hãy click đúp ảnh. Hộp thoại Picture Properties sẽ mở ra.
2. Click tab General để đưa nó lên phía trước. Tab này có thể được sử dụng để thay đổi hoặc thay thế một ảnh, chọn một hyperlink, hoặc cung cấp một chú thích và một thông tin mô tả khác.
3. Nhập một chú thích ngắn gọn cho ảnh trong trường text (hoặc thay thế chú thích hiện có, nếu có), như được minh họa trong hình dưới



Hình VIII.20. Thêm một chú thích vào một ảnh

4. Click OK.

Để xem trang của ta trông giống như thế nào mà không có ảnh, hầu hết các trình duyệt Web có thể được cấu hình để ngưng hiển thị chúng. Trong Internet Explorer 6, hãy thực hiện các bước sau:

1. Trên menu hệ thống chọn Tool → Internet Options. Hộp thoại Internet Options được mở ra.
2. Click tab Advanced để đưa nó lên phía trước.
3. Cuộn xuống danh sách Setting cho đến khi ta tìm thấy phần Multimedia.
4. Xóa dấu kiểm kê bên hộp kiểm Show Picture.
5. Click OK.

Các ảnh sẽ không được hiển thị trong các trang Web mà ta tải sau điểm đó, mặc dù một số trang trong cache hoặc bộ nhớ đệm vẫn còn cung cấp các hình ảnh đồ họa.

Ta có thể mở lại chế độ hiển thị ảnh bằng cách chọn hộp kiểm Show Pictures.

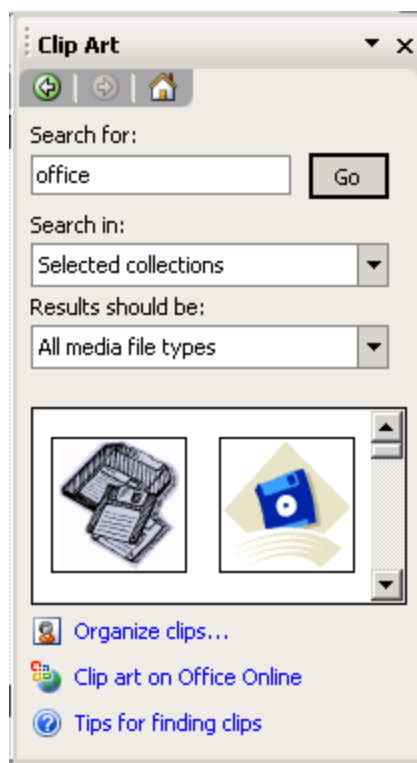
### **Thêm một Clip Art vào một ảnh**

FrontPage 2003 có một thư viện chứa hàng ngàn ảnh clip art, ảnh chụp, và các file đa phương tiện khác. Những ảnh này có thể được kết nhập vào trong các trang Web riêng của ta.

Thư viện clip art chứa FrontPage và bất kỳ sản phẩm Of-fice khác mà ta đã sử dụng, chẳng hạn như các phiên bản trước của FrontPage. Nó cũng có thể chứa một số ảnh đồ họa, ảnh chụp kỹ thuật số, và các ảnh khác trên máy tính của ta – Microsoft Clip Organizer được đính kèm với Office tìm kiếm trên máy tính của ta dành cho cá file ảnh đồ họa và đa phương tiện khác.

Để tìm và thêm clip art vào một trang, thực hiện các bước sau:

1. Đặt con trỏ tại vị trí nơi ảnh sẽ được chèn vào.
2. Trên menu hệ thống chọn Insert → Picture → Clip Art. Khung Clip Art mở ra nằm bên phải của bộ biên soạn (xem hình dưới). Ta cũng có thể được hỏi xem Clip Organizer có nên được lập hạng mục cho các file đa phương tiện trên hệ thống của ta hay không. Điều này mất nhiều thời gian và có thể được hiện vào bất cứ lúc nào, do đó không cần phải thực hiện nó ngay lập tức.



Hình VIII.21. Tìm kiếm Clip Art

3. Trong khung Clip Art, nhập một hoặc nhiều từ vào trong hộp Search For nhằm mô tả loại ảnh mà ta tìm.
4. Để thực hiện một tìm kiếm, ta hãy sử dụng hộp danh sách Search In. FrontPage có thể tìm kiếm tập hợp clip art riêng của nó, ảnh Office khác, các clip mà ta đã tổ chức, hoặc tất cả ba thứ này cùng một lúc.
5. Để thu hẹp thêm nó, hãy sử dụng hộp danh sách Results Should Be. Thư viện clip art chứa các loại đa phương tiện: các bản vẽ clip art, các ảnh chụp, phim và âm thanh. Ta có thể tìm kiếm một loại tất cả chúng.
6. Click nút Go. Các file phù hợp với việc tìm kiếm của ta được hiển thị trong khung. Các ảnh thumbnail của mỗi ảnh clip art và các ảnh chụp được hiển thị - sử dụng thanh cuộn để xem chúng. Hình trên thể hiện kết quả tìm kiếm.
7. Khi ta tìm được một ảnh mà ta thích, hãy click đúp vào nó, ảnh sẽ xuất hiện trên trang, thường với kích cỡ lớn hơn nhiều so với những gì mà ta mong muốn. Ta sẽ có thể định lại kích cỡ ảnh này khi cần thiết.

Clip Art từ thư viện thường lớn hơn nhiều so với những gì ta mong đợi. Điều này là do phần thiết kế bởi vì ta có thể dễ dàng thu nhỏ một ảnh sang một kích cỡ mà ta muốn mà không làm mất chất lượng ảnh của nó. Việc tăng kích cỡ và duy trì một chất lượng thì khó hơn nhiều (thường không thực hiện được).

Khi ta thêm clip art vào một trang, ảnh gốc vẫn nằm trong thư viện. Những thay đổi mà ta thực hiện với ảnh sẽ không thay đổi bản gốc của ảnh.

Nếu ta không thể tìm thấy clip art phù hợp trong thư viện được đính kèm với FrontPage 2003, thì các file đa phương tiện khác dành cho các trang Web của ta có sẵn trên Web site Microsoft Office Online. Để đi đến đó, hãy thực hiện các bước sau:

1. Nếu khung Clip Art không được mở, hãy chọn Insert → Picture → Clip Art.
2. Click liên kết Clip Art on Office Online nằm ở cuối cùng.

Trình duyệt Web của ta mở ra tại Microsoft Office Clip Art and Media site, nơi ta có thể duyệt theo từng hạng mục hoặc tìm kiếm các từ khóa cụ thể.

Các file mà ta chọn từ Office Online được nhập tự động vào thư viện clip art của FrontPage 2003.

## Thay đổi kích cỡ của một ảnh

Kích cỡ của bất kỳ ảnh nào trên một trang Web đều có thể thay đổi được - được tăng đến một kích cỡ lớn hơn, được thu nhỏ thành một kích cỡ nhỏ, hoặc ngay cả được biến dạng để chiều rộng có một tỷ lệ khác với chiều cao.

Kích cỡ của một ảnh có thể được thay đổi theo hai cách trong FrontPage. Ta có thể định lại kích cỡ ảnh thực, nhằm thay đổi file chứa ảnh, hoặc thay đổi kích cỡ hiển thị của ảnh.

Các ảnh được biểu diễn trên một trang Web có thể được hiển thị ở các kích cỡ lớn hơn hoặc nhỏ hơn kích cỡ thực của nó, điều này làm cho cùng một ảnh được trình bày với những kích cỡ khác nhau trên hai trang khác nhau của một Web site.

Bởi vì lý do này, nên điều quan trọng là ta phải lưu ý đến sự khác biệt giữa việc định lại kích cỡ một ảnh – và việc chỉ hiển thị ảnh với một kích cỡ khác, nhằm không thay đổi file đồ họa gốc.

Nếu ta không bao giờ hiển thị ảnh với một kích cỡ lớn hơn, thì việc định lại kích cỡ ảnh là một lựa chọn tốt hơn vì nó sẽ tải nhanh hơn khi một khách tham quan xem trang Web của ta. (Nếu ảnh không bắt nguồn từ thư viện clip art, hãy lưu một bản sao của ảnh lớn hơn bất kỳ vị trí nào).

Các ảnh clip art trông đẹp hơn khi chúng được định lại kích cỡ một cách cố định thay vì được hiển thị với một kích cỡ lớn hơn bởi vì FrontPage làm một công việc hiệu chỉnh ảnh nào đó trên file để làm nhẵn các mép và làm rõ nét các chi tiết. Tiến trình này được gọi là việc lấy lại mẫu (resampling) ảnh.

Để định lại kích cỡ một ảnh, thực hiện các bước:

1. Click vào ảnh. Các nút chọn xuất hiện xung quanh các mép của ảnh, như được minh họa trong hình dưới. Khi ta đặt chuột lên trên một trong các nút này, con trỏ biến thành một mũi tên, điều này biểu thị rằng ta có thể kéo các nút xung quanh.
2. Để thay đổi kích cỡ của ảnh, kéo một nút đến một vị trí mới:
  - a. Để thay đổi chiều cao, giữ nút tại phần chính giữa của đỉnh hoặc đáy của ảnh.
  - b. Để thay đổi chiều rộng, giữ nút tại phần chính giữa của cạnh trái và phải.

- c. Để thay đổi kích cỡ trong khi duy trì chiều rộng và chiều cao, giữ bất kỳ góc nào.

Ảnh thay đổi kích cỡ khi các nút được di chuyển, cho phép ta xem trước việc thay đổi trước khi nó được hoàn tất.

3. Nhả nút. Một hộp nhỏ xuất hiện kế bên ảnh. Click hộp này để mở một menu thả xuống vốn quyết định ảnh sẽ được định lại kích cỡ như thế nào.
4. Chọn Only Modify Size Attributes để thay đổi kích cỡ hiển thị của ảnh.
5. Chọn Resample Picture to Match Size để định lại kích cỡ ảnh một cách cố định.

Để thay đổi kích cỡ, hãy kéo hộp đến một vị trí mới để làm cho ảnh lớn hoặc nhỏ hơn,

Sau khi clip art đã được thêm vào một trang, nó có thể được xem như bất kỳ ảnh khác. Ta có thể di chuyển nó vòng quanh, thêm một chú thích, và thực hiện những thay đổi khác.

### ***VIII.2.5. Tạo một Web site mới***

Ở trên ta đã có thể thực hiện các công việc thao tác đơn lẻ trên một trang Web. Bây giờ ta sẽ tìm hiểu và học cách tạo một Web site mới ra sao.

Việc nhập (importing) các file trở nên đặc biệt tiện lợi nếu ta có một số tài liệu được tạo bằng Microsoft Word hoặc chương trình khác mà ta muốn đưa lên World Wide Web.

#### **Tạo một Web site mới**

FrontPage cung cấp nhiều tính năng nhằm giúp ta dễ dàng tạo, quản lý, và xuất các Web site.

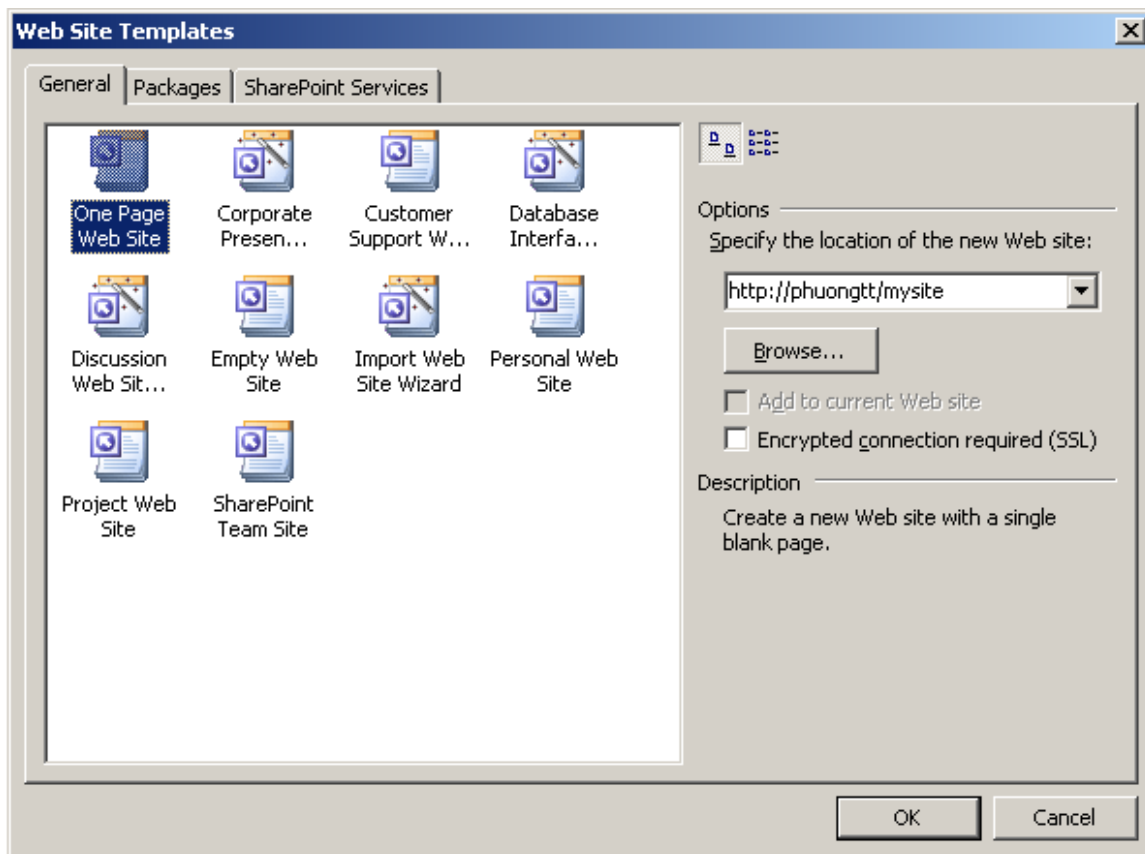
Các site được quản lý bởi FrontPage được tổ chức với việc sử dụng các file thư mục của máy tính. Một site chiếm một thư mục chính và một số thư mục con, mỗi thư mục con phục vụ một mục đích khác nhau.

Khi một thư mục đã được chỉ định làm một thư mục chính cho một site, thì nó không được sử dụng cho bất kỳ site nào khác nữa.

Khi FrontPage được cài đặt, một thư mục My Web Sites được tạo bên trong My Documents thuộc máy tính của ta. Đây là một nơi tốt để lưu trữ các thư mục riêng lẻ mà sẽ chứa các site ta tạo ra.

Để tạo một Web site mới ngay từ đầu, chạy FrontPage và thực hiện các bước sau:

1. Trên menu hệ thống chọn File → New. Khung New mở ra nằm dọc theo cạnh phải của giao diện FrontPage, kể bên cửa sổ nơi các trang Web được biên soạn.
2. Click hyperlink More Web Site Template.



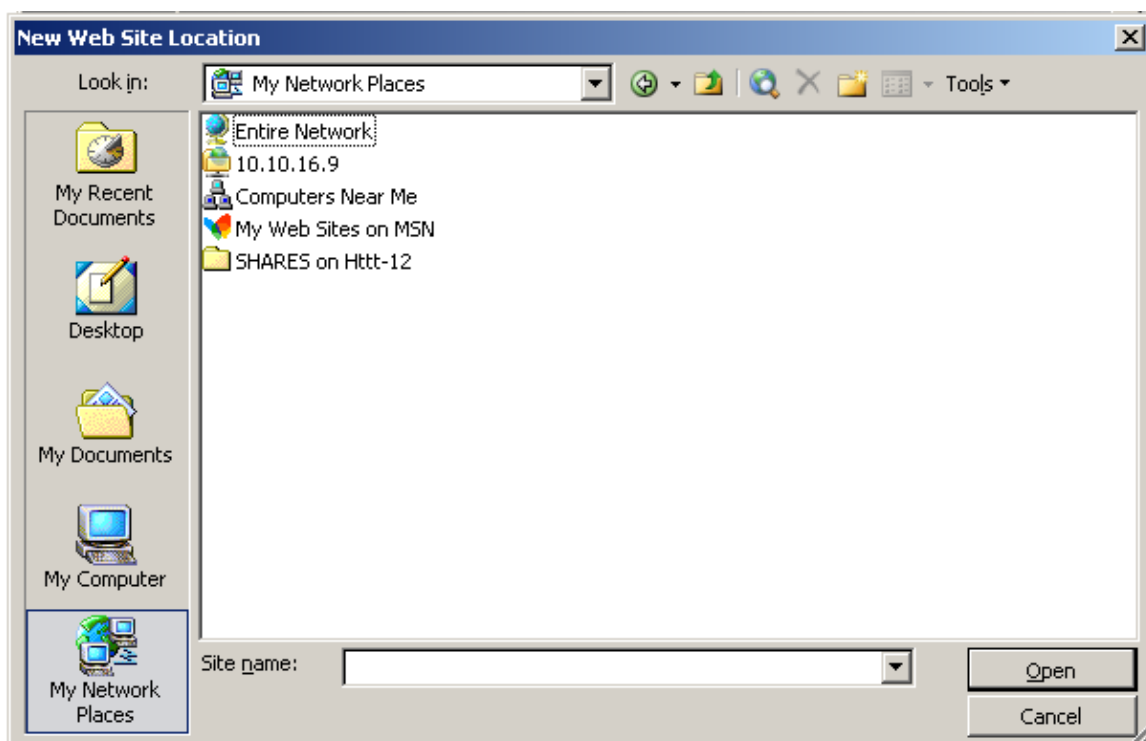
*Hình VIII.22. Hộp thoại Web Site Templates*

Các Web site được tạo trong FrontPage bằng cách sử dụng các template, các phiên bản dự trữ của một số Web site mà ta có thể tạo tùy biến. Ta sẽ Tìm hiểu kỹ hơn trong phần sau.

3. Chọn biểu tượng của template On-Page Web Site, template này được sử dụng để tạo một site đơn giản, không chứa gì cả ngoại trừ một trang Web trống.

Kế tiếp, ta có thể tạo một thư mục mới trên máy tính của ta nơi các site sẽ được lưu ở đó.

1. Click nút Browse. Hộp thoại New Web Site Location mở ra, như được minh họa trong hình dưới.



*Hình VIII.23. Hộp thoại New Web Site Locations*

2. Mỗi Web site được quản lý bởi FrontPage phải được cung cấp bởi thư mục riêng của nó. Trước khi ta có thể thực hiện điều đó, hãy tìm và mở thư mục này nơi ta sẽ lưu giữ thư mục này – một vị trí thích hợp là thư mục My Web Site bên trong My Documents.
3. Để tạo một thư mục cho site mới của ta, hãy click nút Create New Folder, được minh họa trong hình trên. Hộp thoại New Folder mở ra.
4. Nhập tên thư mục mới trong trường Name và click nút OK. Thư mục được mở và hiển thị trong menu Look In thả xuống của hộp thoại New Web Site Location.

Để mở một trang Web trong cửa sổ biên soạn, click đúp file. Click đúp index.htm để mở trang chủ của site vừa mới được tạo để hiệu chỉnh.

Để quay trở về khung xem Folder

### **VIII.2.5. Khai thác site mới**

Khi ta tạo một Web site mới, FrontPage sẽ thêm tất cả file và thư mục cần thiết kể vào thư mục chính của site.

Để cho ta một trang rõ ràng về những gì mà site chứa, nó được hiển thị trong khung xem Folders, được minh họa ở hình dưới. (Có thể ta không nhìn thấy khung nhìn Folder List nằm bên trái khung nhìn Folders).





Những trang này có thể được thêm vào một site, như được đề cập ở phần kế tiếp.

Ta cũng có thể tạo một trang mới vốn được thêm vào site mà ta đang hiệu chỉnh:

1. Trên menu hệ thống chọn File → New để mở khung New nếu nó chưa được mở.
2. Chọn loại trang ta muốn tạo:
  - Đối với một trang hoàn toàn trống, click hyperlink Blank Page.
  - Đối với các loại trang khác, click liên kết More Page Templates.

Hộp thoại Page Templates xuất hiện, liệt kê các loại template mà trang mới có thể được dựa vào đó. Chọn biểu tượng của template mà ta muốn và lick OK.

Trong thư mục chính của site, FrontPage tạo một trang mới bằng cách sử dụng template và đặt cho nó một tên chẳng hạn như new\_page\_1.htm hoặc new\_page\_2.htm. Tên trang sẽ có một ký tự dấu sao (“\*”) theo sau khi nó chứa những thay đổi chưa được lưu.

Click nút Save trong thanh công cụ Standard để lưu file đang hiển thị trong cửa sổ biên soạn. Trên menu hệ thống chọn File → Save All để lưu tất cả các file đang mở hiệu chỉnh.

Để đóng một Web site, chọn File → Close. Nếu bất kỳ các trang được hiệu chỉnh chưa được lưu, một hộp thoại được mở ra dành cho mỗi trang hỏi có muốn lưu file đó hay không. Click Yes để lưu file hoặc No để loại bỏ các thay đổi.

### **Nhập các file vào một site**

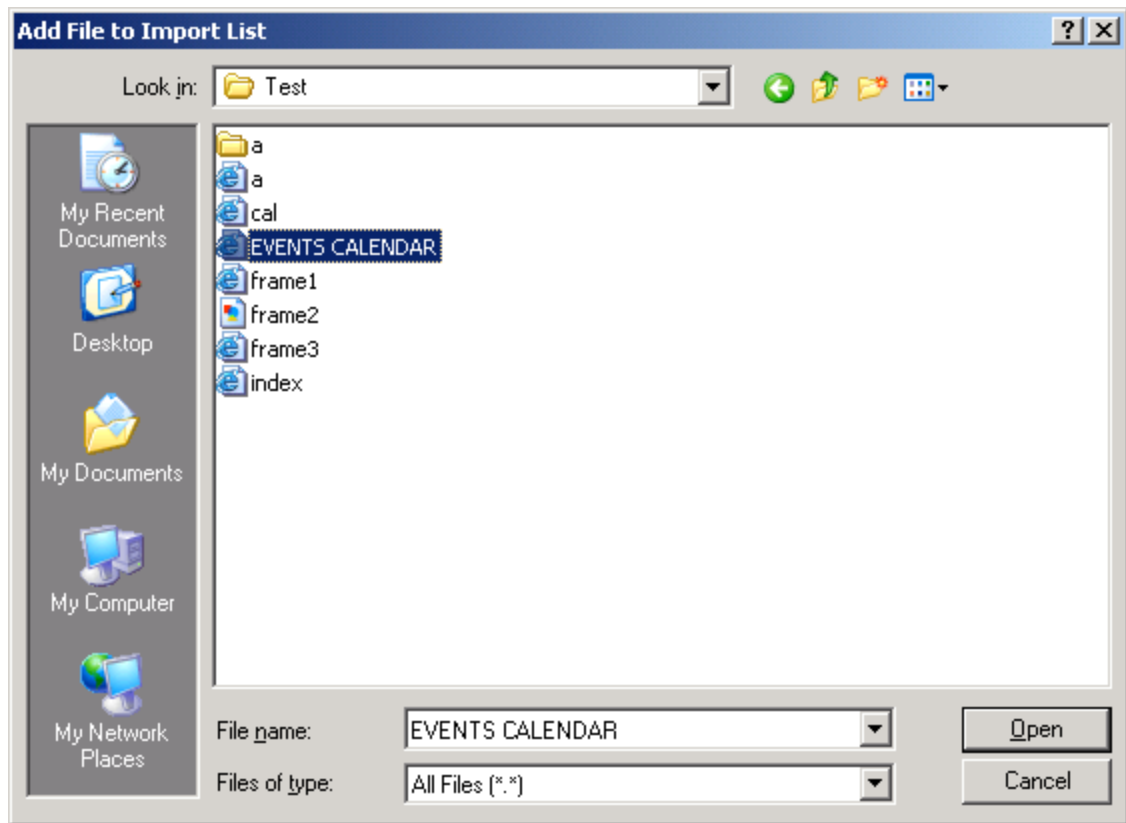
World Wide Web có thể trình bày các loại tài liệu khác ngoài các trang Web:

- Các file ảnh đồ họa dạng GIF, JPEG, PNG, và các dạng khác
- Các file âm thanh chẳng hạn như các file WAV, MIDI và MP3
- Các file Movie theo dạng AVI, MOB, MPD, và QuickTime
- Các chương trình Macromedia Shockwave và Flash, và các chương trình tương tác khác.

Khi ta tạo một site mới, thư mục của nó sẽ chứa một thư mục con images trong đó nơi hình ảnh của một site có thể được lưu trữ. Nếu ta đang làm việc với các loại media khác, ta có thể muốn tạo các thư mục dành cho chúng - chẳng hạn như sound cho bất kỳ file âm thanh mà ta đang sử dụng.

Một cách để kết hợp những file media này và các file khác vào trong site của ta là nhập (import) chúng vào trong site:

1. Với site đang mở, chọn File → Import. Hộp thoại mở ra, liệt kê các file mà ta đã chọn để nhập vào site. Nếu ta chưa nhập, thì danh sách rỗng.
2. Click nút Add File. Hộp thoại Add File to Import List mở ra.



*Hình VIII.25. Hộp thoại Add File to Import List*

3. Sử dụng hộp thoại này để tìm và mở thư mục chứa file đó. Sau đó click file. File được liệt kê trong hộp thoại Import.
4. Click Add File để chọn một file khác để nhập, hoặc click OK để nhập các file được chọn vào site của ta.

Các file được nhập trở thành một phần của site. Bất cứ khi nào ta sao chép site sang một thư mục mới hoặc xuất bản nó lên World Wide Web, file đó sẽ được đưa vào.

Một file nhập có thể được kéo đến một thư mục khác thuộc site này từ khung nhìn Folders hoặc khung Folder List.

### **Xóa một site**

Một Web site được quản lý bởi FrontPage có thể bị xóa theo hai cách, một sự khác biệt cực kỳ quan trọng mà ta cần lưu ý trước khi xóa một site.

Cách thứ nhất để xóa một site là loại bỏ nó hoàn toàn: Điều này sẽ xóa sạch thư mục chứa site, tất cả các thư mục con của nó, cùng tất cả các trang, hình ảnh, và các file khác vốn tạo nên site.

Cách thứ hai là chỉ xóa các file và thư mục mà FrontPage sử dụng để quản lý site, để lại nguyên vẹn mọi thứ khác.

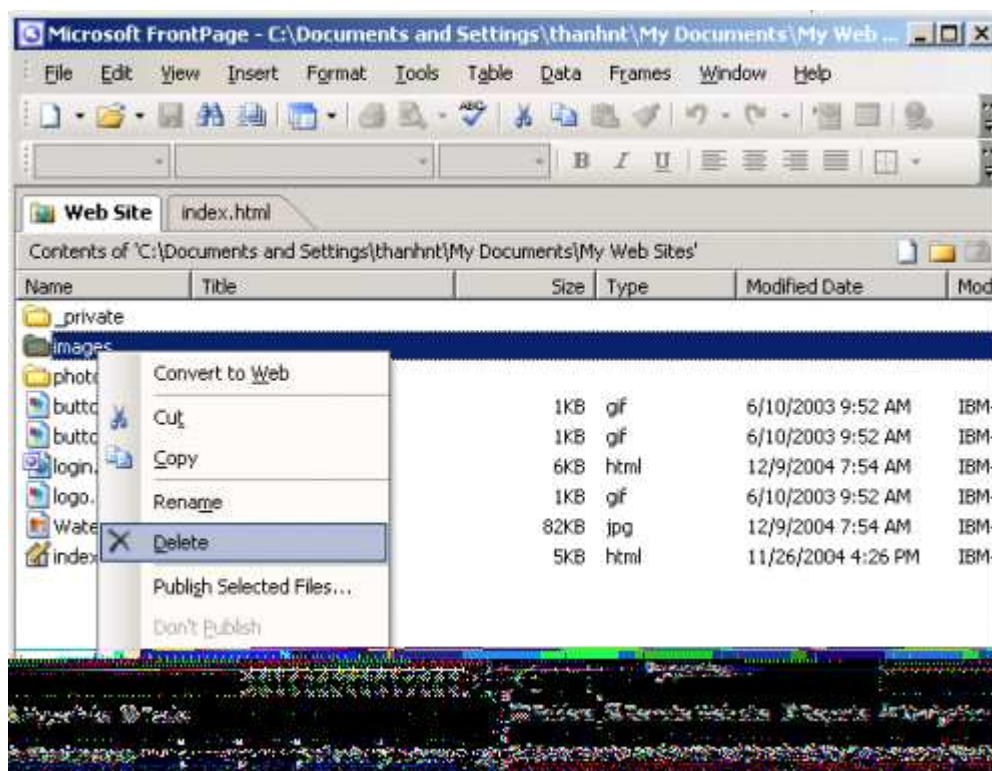
Khi ta làm việc trên một Web site bằng FrontPage, phần mềm tạo ra một số thư mục và file được sử dụng nhằm bảo trì site (ví dụ: FrontPage theo dõi khi nào từng file của site được xuất cuối cùng từ máy tính của ta lên World Wide Web).

Việc xóa thông tin FrontPage của site sẽ chỉ xóa các file và thư mục mà FrontPage sử dụng ở hậu cảnh để quản lý site. Mọi thứ khác – trang, hình ảnh và file khác – không bị loại bỏ.

Một site không có nội dung FrontPage vẫn được xem một cách bình thường bằng một trình duyệt Web nhưng ta sẽ không thể mở nó để hiệu chỉnh trong FrontPage. Ta có thể mở các trang riêng lẻ trong site để hiệu chỉnh, nhưng ta không thể áp dụng một theme vào một site, thêm các thư mục chia sẻ vào nhiều trang hoặc áp dụng bất kỳ tính năng khác có liên quan đến toàn bộ một site.

Để xóa một vài hoặc tất cả các file trong một site, hãy thực hiện các bước sau:

1. Nếu site đã không được mở, trên menu hệ thống chọn File → Open Site. Hộp thoại Open xuất hiện, hãy sử dụng nó để chọn thư mục chính của site, và click Open.
2. Nếu danh sách thư mục không được hiển thị, hãy mở nó: Click View → Folder List.
3. Click chuột phải vào tên của site – dòng trên cùng trong menu thư mục – và trong menu tắt vừa được mở, hãy chọn lệnh Delete (như minh họa trong hình dưới)



Hình VIII.26. Xóa một thư mục

Một hộp thoại mở ra cho phép ta xóa thông tin FrontPage hoặc xóa toàn bộ Web site.

Cả hai hoạt động này đều mang tính vĩnh viễn, vì thế ta nên cẩn thận khi sử dụng chúng.

4. Để xóa một site hoặc thông tin FrontPage của nó, chọn một tùy chọn và nhấp OK.
5. Để đóng hộp thoại mà không xóa bất cứ thông tin nào, click Cancel.

Một site hoàn toàn bị xóa sẽ không thể phục hồi trở lại.

Nếu ta loại bỏ thông tin FrontPage ra khỏi một site, thì nó có thể được tạo lại sau đó: trên menu hệ thống chọn File → Open, và sau đó sử dụng hộp thoại Open để tìm và chọn thư mục chính của site.

FrontPage hỏi ta có muốn chuyển đổi thư mục này thành một Web site hay không – click Yes. Sau đó FrontPage sẽ tạo thông tin nó cần để quản lý site.

### Các file và thư mục ẩn

Nếu ta mở một thư mục của một Web site bên ngoài FrontPage, ta sẽ nhìn thấy nó chứa các thư mục và file không thể hiển thị trong khung nhìn Folders hoặc khung Folders List – đáng chú ý nhất là toàn bộ một nhóm thư mục có tên `_vti_cnf`

đang được sử dụng bởi FrontPage làm một phần trong các tính năng quản lý site của nó. Mặc dù các tên của chúng làm cho chúng dường như là các trang Web, đồ họa và đại loại như thế, nhưng các file chứa thông tin về các file trong site của ta - chẳng hạn như thời điểm sau cùng mà một file cụ thể đã được chỉnh sửa. Các file trong thư mục \_vti\_cnf cũng được sử dụng cho việc quản lý site.

Những file này không nên xóa (mặc dù FrontPage có thể tạo lại chúng nếu cần thiết)

Khi ta xóa thông tin FrontPage được kết hợp với một site, những thư mục và file này là những gì được loại bỏ.

### **Sử dụng tài liệu World trên Web**

FrontPage được kết nhập chặt chẽ với các chương trình khác trong bộ Microsoft Office. Kết quả là ta có thể dễ dàng kết nhập dữ liệu được tạo bằng một chương trình vào một chương trình khác.

Ví dụ, ta có thể làm cho các tài liệu được tạo bằng Microsoft Word trở thành một phần của Web site. Ta có thể thực hiện điều này theo hai cách.

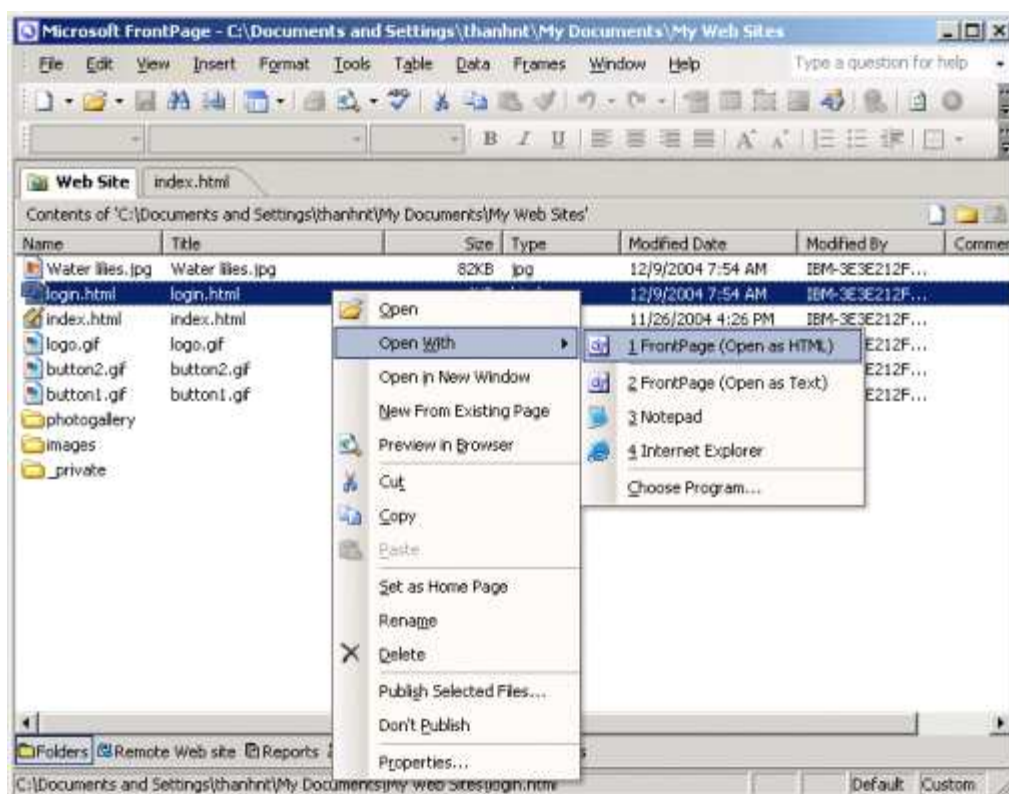
Cách dễ dàng là nhập (import) một tài liệu Word vào trong site của ta và để nó theo dạng nó. Những tham quan đến site của ta họ có thể xem các tài liệu Word sẽ có thể xem file bằng cách tải nó với Word.

Khi ta để một tài liệu theo dạng Word, ta không thể hiệu chỉnh nó với FrontPage. Thay vào đó, khi ta cố mở nó theo cách bình thường - bằng cách click đúp file trong khung nhìn Folders hoặc danh sách Folders - nó mở trong Microsoft Word.

Những người dùng sẽ cảm thấy thuận lợi hơn nhiều khi xem thông tin của ta nếu nó được chuyển đổi từ dạng tài liệu Word sang một hoặc nhiều trang Web.

Để chuyển đổi nội dung một tài liệu Word sang một trang Web:

1. Nhập tài liệu bằng cách trên menu hệ thống chọn File → Import, nhưng ta thường thực hiện đối với bất kỳ file.
2. Nếu ta không làm việc trong khung xem Folders, click nút Folder bên dưới cửa sổ hiện tại.
3. Click chuột phải vào tài liệu trong khung xem Folders, và sau đó chọn Open With → FrontPage (Open as HTML) từ menu tắt vừa xuất hiện (xem hình dưới)



Hình VIII.27. Hiệu chỉnh một tài liệu Word trong FrontPage

Lần đầu tiên ta thực hiện điều này với một tài liệu, có thể nó không có sẵn một chuyển đổi để hiển thị file một cách chính xác. Một hộp thoại sẽ xuất hiện hỏi ta có muốn cài một bộ chuyển đổi hay không. Bởi vì ta đang chuyển đổi tài liệu thành một trang Web, click No.

Khi ta sử dụng nó để hiệu chỉnh một tài liệu Word, một trang Web mới được tạo chứa các nội dung của tài liệu.

Tài liệu này có thể chứa nhiều định dạng – càng nhiều càng tốt, các lựa chọn font, kiểu căn chỉnh và style được sử dụng trong Microsoft Word sẽ được chuyển sang FrontPage.

Sau khi đã chuyển đổi, nó có thể được lưu một cách bình thường: Click nút Save trên thanh công cụ Standard hoặc chọn File → Save.

### VIII.3. Phát triển nhanh một site với các template

Khi bắt đầu tạo các Web site riêng của ta bằng FrontPage 2003, chương trình dường như khá gây ấn tượng. Việc biết nơi để bắt đầu và nơi để thực hiện có thể là một công việc gây bối rối khi ta đã chưa bao giờ sử dụng phần mềm này.

FrontPage làm cho công việc này dễ quản lý hơn thông qua việc sử dụng các template, các site được tạo sẵn mà phần mềm biết cách để tự tạo.

Các template cho phép ta phát triển một site hoàn chỉnh trong một vài phút và sau đó dành những nỗ lực của ta vào việc tạo tùy biến site đó thay vì ta tạo một site ngay từ đầu.

Các template tiết kiệm cho ta rất nhiều thời gian khi chúng phù hợp cho một dự án.

### **VIII.3.1. Chọn một template Web site**

FrontPage cung cấp một vài chục template Web site và trang nhằm giúp ta có một sự khởi đầu dễ dàng và nhanh chóng. Mỗi template là một khung sườn mà ta có thể tạo tùy biến bằng cách thêm và loại bỏ text và thực hiện các thay đổi khác. Phần mềm chứa các template dành cho một số cách phổ biến nhất mà các nhà xuất bản sử dụng Worl Wide Web.

Các template site được cài sẵn chứa một site cá nhân, một site công ty, một trang phản hồi khách hàng và một trang mục lục.

Tất cả các site FrontPage và các trang Web bắt đầu từ các template. Các template sau đây có thể được sử dụng để bắt đầu một site:

- **One Page Web Site** - Một site có một trang trống và không có gì khác.
- **Personal Web Site** - Một site có các trang nơi ta có thể mô tả chính mình, chia sẻ ảnh kỹ thuật số, và viết về những sở thích của ta và các Web site mà ta ưa thích nhất.
- **Customer Support Web Site** - Một site cho phép một công ty cung cấp sự hỗ trợ khách hàng thông qua các trang mô tả sản phẩm, một diễn đàn thảo luận, mẫu biểu phản hồi (feedback form), form tìm kiếm site, và một trang có các câu hỏi thường gặp.
- **Project Web Site** - Một site dành cho một dự án hợp tác, với các trang dành cho thông tin thành viên, tình hình dự án, một thời biểu, các cuộc thảo luận nhóm và một nơi lưu trữ.
- **SharePoint Team Site** - Một site dành cho việc cộng tác của nhóm có một lịch, một danh sách tác vụ, một thư viện file, và một danh sách contact.

Cũng có một template Empty Web Site vốn tạo một site không chứa các file. Nó khác với template One-Page Web Site ở điểm nó thiếu một trang index.htm.

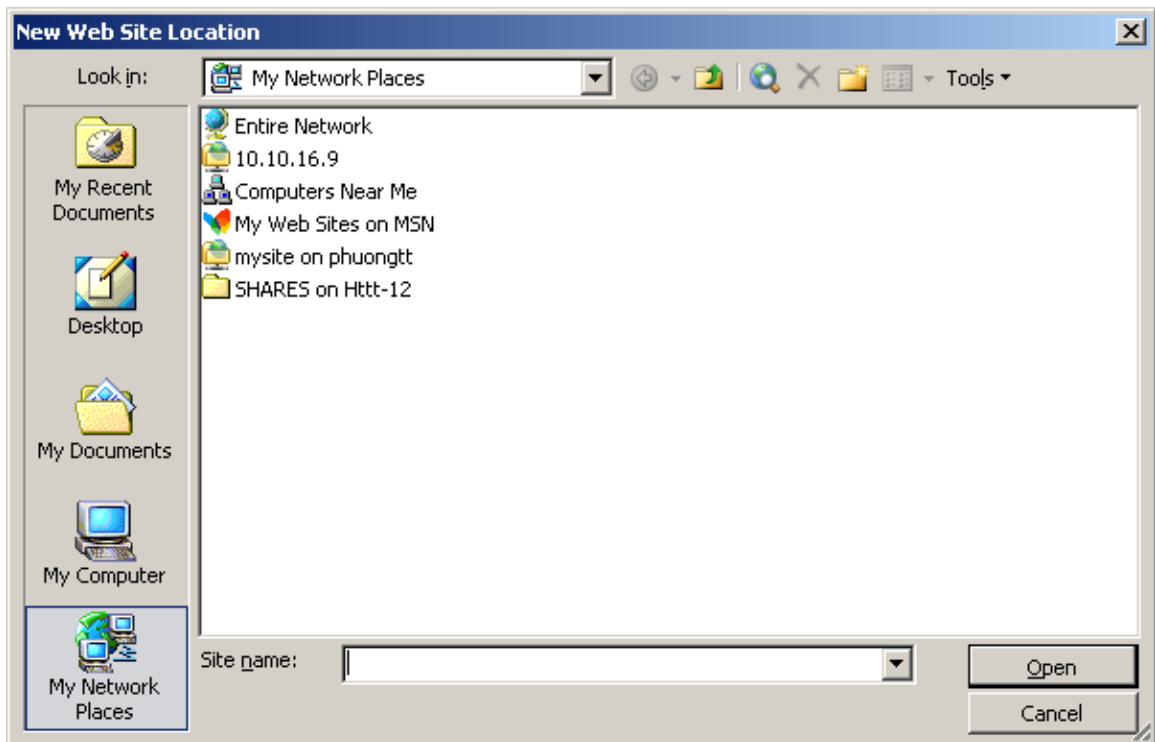


### VIII.3.2. Tạo một Web site mới

Trong bốn giờ đầu, ta đã làm việc trên các trang Web mà không đặt chúng vào các Web site riêng của chúng.

Ta có thể dễ dàng bắt đầu một dự án Web mới bằng cách tạo một site mới trước khi bất kỳ trang đã được tạo. Bởi vì tất cả các site bắt đầu từ một template, ta phải chọn một template để bắt đầu một dự án:

1. Trên menu hệ thống chọn File → New. Khung menu mở trong FrontPage nằm dọc theo mép phải của cửa sổ biên soạn.



*Hình VIII.28. Hộp thoại New Location được sử dụng để cho FrontPage biết nơi để lưu site của ta*

2. Click hyperlink More Web Site Templates. Hộp thoại Web Site Templates mở ra, liệt kê các template mà ta có thể sử dụng để tạo một site mới. Nó cũng liệt kê các wizard để ta làm việc dễ dàng hơn.
3. Chọn template để sử dụng cho site của ta – click biểu tượng của nó. Tiếp theo, ta nên tạo một thư mục mới nơi site sẽ được lưu trên máy tính của ta.
4. Click nút Browse. Hộp thoại này để tìm và mở thư mục nơi ta sẽ lưu các Web site của ta.
5. Sử dụng hộp thoại này để tìm và mở thư mục nơi ta sẽ lưu các Web site của ta.

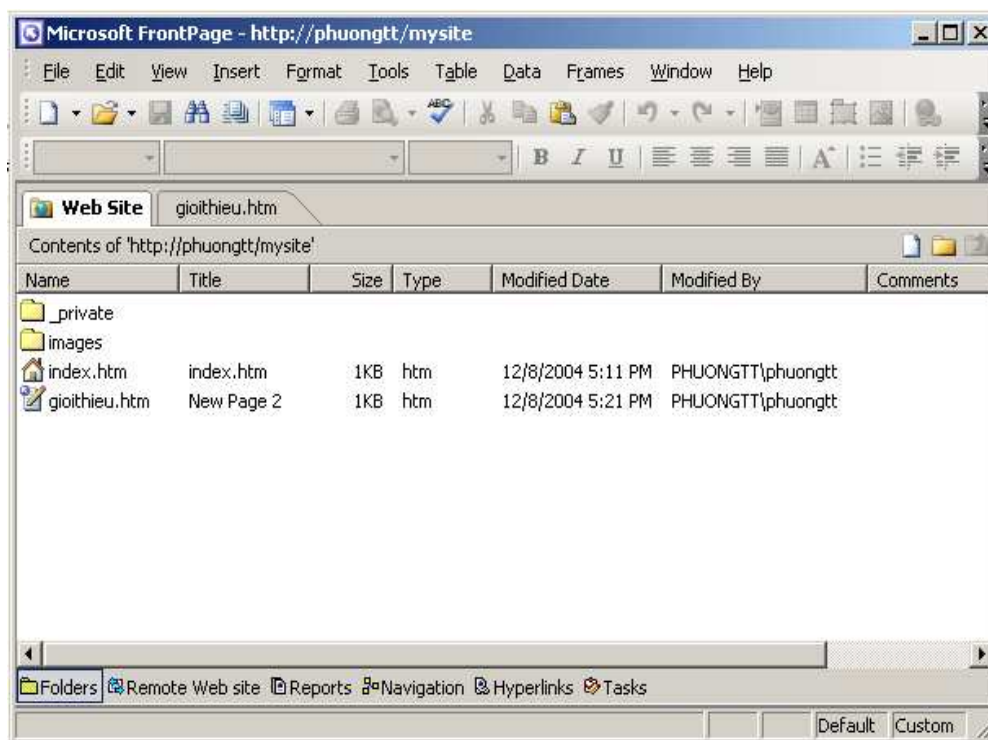
6. Mỗi site sẽ được cung cấp thư mục con riêng của nó: Để tạo một thư mục con, click nút Create New Folder, như được minh họa ở hình trên. Hộp thoại New Folder mở ra.
7. Nhập một tên cho thư mục trong trường Name, và sau đó click nút OK. Hộp thoại đóng lại, đưa ta trở về lại hộp thoại New Web Site Location.
8. Click nút Open. The mục mà ta đã chọn được hiển thị trong trường text Specify the Location of the New Web Site.
9. Click OK. FrontPage sẽ tạo site, lấp đầy thư mục các trang Web, hình ảnh, và các thư mục con.

### **VIII.3.3. Tạo tùy biến Web site mới của ta**

Sau khi một Web site mới được tạo từ một template, ta có thể bắt đầu thực hiện các thay đổi để tạo tùy biến site này.

FrontPage sẽ hiển thị một site vừa mới tạo trong khung xem Folders để ta có thể thấy tất cả các file mà nó chứa, như minh họa trong hình dưới.

Tất cả các trang Web trong site nằm trong thư mục chính của nó. Cũng có một số thư mục con được sử dụng bởi FrontPage và một vài thư mục con mà ta muốn sử dụng, chẳng hạn như images (dành cho hình ảnh), và \_private (dành cho các file mà các site khách tham quan sẽ không thể thấy được).



*Hình VIII.29. Khung nhìn Folders*

Một thư mục con `_private` của một Web site, thư mục con này ban đầu rỗng, có thể được sử dụng để làm vị trí cho các file mà đã được che giấu các khách tham quan đến Web site của ta. Nếu ta tạo một Web site nơi khách tham quan có thể cung cấp cho ta các địa chỉ thư tín của họ, thư mục này có thể được sử dụng làm một nơi để ngăn ngừa thông tin này khỏi những cặp mắt soi mói.

Nếu ta không sử dụng template Empty Web Site, thì thư mục chính thuộc site của ta sẽ chứa một file đặc biệt có tên `index.htm`. Đây là trang chủ của site – trang mà các khách tham quan sẽ thấy khi họ yêu cầu địa chỉ Web chính thuộc site của ta.

Một số template chứa các thư mục phục vụ một mục đích riêng cho template đó. Ví dụ, template Personal Web Site chứa một thư mục photogallery được sử dụng cho các trang vốn hiển thị các ảnh kỹ thuật số. Để bắt đầu làm việc trên Web site mới của ta, hãy mở một trong các trang dành để hiệu chỉnh: Click đúp vào tên của trang trong khung nhìn Folders.

Trang mở ra trong cửa sổ hiệu chỉnh, nơi ta có thể làm việc trên nó, như ta thường thực hiện với bất kỳ trang khác, hãy thực hiện các thay đổi và lưu nó với nút Save trên thanh công cụ Standard.

Để quay trở về khung nhìn Folders, click tab Web Site nằm ở phần trên cùng của cửa sổ biên soạn.

#### ***VIII.3.4. Thêm và loại bỏ các lời chú thích***

Trên mỗi trang trong các site được tạo từ một template FrontPage cung cấp thông tin hướng dẫn về một số trang theo dạng các lời chú thích (comment), text được hiển thị bằng text màu đỏ tía có từ “Comment”.

Các lời bình chú được hiển thị khi một trang đang hiệu chỉnh trong FrontPage nhưng được che giấu khi nó hiển thị trong một trình duyệt Web. Để thấy được điều này, hãy click nút Preview bên dưới cửa sổ biên soạn. Text chú thích không xuất hiện trên trang.

Để hiệu chỉnh một lời chú thích, click đúp nó. Một hộp thoại Comment mở ra với lời chú thích chúng trong một vùng text. Sau khi thực hiện các thay đổi, click OK để lưu chúng.

Một lời chú thích mới có thể được thêm vào một trang Web:

1. Với trang mở ra để hiệu chỉnh, click vào nơi mà lời chú thích được chèn vào đó.
2. Trên menu Standard chọn Insert → Comment để mở hộp thoại Comment.

3. Nhập text chú thích và click OK. Lời chú thích sẽ xuất hiện trên trang bằng text màu đỏ tía.

Ta có thể xóa các chú thích vào bất kỳ lúc nào, thậm chí nếu chúng là một phần của một template site hoặc template trang. Click lời chú thích mà ta muốn xóa và nhấn phím Delete.

### ***VIII.3.5. Khai thác template Personal Web Site***

Dù sớm hay muộn, thì hầu hết các nhà xuất bản Web đều sẽ tạo các trang chủ riêng cho mình, một nơi để làm cho họ trở nên nổi tiếng trên World Wide Web. Ta có thể dễ dàng tạo một file tự truyện cho sở thích chuyên môn hoặc cá nhân của ta bằng cách sử dụng template Personal Web Site.

Template này gồm các thành phần sau đây:

- Index.htm, trang chủ của site, cung cấp một nơi để chào đón các khách tham quan và tự giới thiệu về mình.
- Photo.htm, một trang để chia sẻ các ảnh kỹ thuật số hoặc các hình ảnh khác.
- Interest.htm, một nơi tốt để mô tả các sở thích của ta và cung cấp các hyperlink dẫn đến các Web site liên quan.
- Favorite.htm, một danh sách các hyperlink dẫn đến các Web site ưa thích nhất của ta.
- Aboutme.htm, một trang nơi ta có thể cung cấp một bản dài hơn về tiểu sử của ta.
- Feedback.htm, một nơi mà những khách tham quan đến site của ta có thể để lại cho ta một thông điệp.

Từng trang này chứa một thanh liên kết (link bar), một hàng gồm các nút đồ họa vốn có thể được click để tham quan phần còn lại của site.

Để xem cách hoạt động của các thanh liên kết trên site mới của ta, click nút Preview in Browser trên thanh công cụ Standard. Site mở ra trong trình duyệt Web chính của ta.

Trên các trang Web của site mới, ta có thể thay đổi hầu hết text, các tiêu đề chính và các hyperlink bằng cách hiệu chỉnh chúng một cách bình thường.

Các template cũng đặt một số thứ trên các trang Web vốn không thể được hiệu chỉnh theo cách mà ta mong đợi: các thanh liên kết, các biểu ngữ trang vốn

hiển thị tiêu đề của một trang, và các tem thời gian (timestamp) vốn hiển thị ngày tháng trên lịch.

Đây là các thành phần Web – các thành phần trang đặc biệt sẽ được hỗ trợ bởi FrontPage nhằm bổ sung chức năng vào một site.

### **VIII.3.6. Thêm một tem thời gian vào một trang Web**

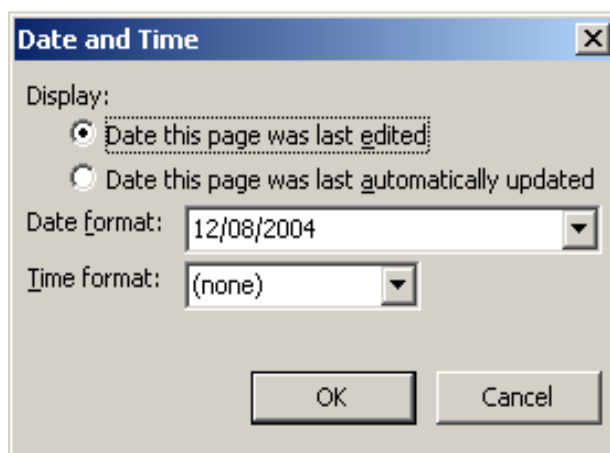
Trang chính trong template Personal Web Site, index.htm, chứa một thành phần tem thời gian nằm ở cuối trang, ngay sau text. “This page last updates” như trong “This page last updated 01/12/2004”.

Thành phần tem thời gian hiển thị ngày tháng mà trang Web đã được hiệu chỉnh lần sau cùng. Nó tự động thay đổi mỗi khi có bất kỳ cái gì ở trên trang bị thay đổi.

Bằng cách sử dụng thành phần này, ta có thể cho phép các khách tham quan biết được thông tin trên trang hiện hành như thế nào. Thay vì công việc mất nhiều thời gian của hoạt động click ngày tháng bằng tay khi ta hiệu chỉnh trang, ta có thể đặt một ngày thành phần tem thời gian lên trên nó.

Để thêm một tem thời gian vào một trang Web, hãy thực hiện các bước sau:

1. Nếu trang không được mở trong cửa sổ biên soạn, click đúp lên file của nó trong khung nhìn Folders.
2. Click một vị trí trên trang nơi một tem thời gian có thể xuất hiện.
3. Trên menu Standard chọn Insert → Date and Time. Hộp thoại Insert, Date and Time mở ra (xem hình dưới).



*Hình VIII.30. Hộp thoại Date and Time*

4. Chọn một trong các tùy chọn Display để xác định những gì làm cho tem thời gian thay đổi:
  - *Nếu nó thay đổi bất cứ khi nào thực hiện bằng tay một thay đổi cho trang, chọn Date This Page Was Last Editor.*
  - *Nếu nó cũng thay đổi khi FrontPage cập nhật trang vì một lý do nào đó, chọn Date This Page Was Last Automatically Updated.*
5. Trong menu Date Format thả xuống, chọn cách mà ngày tháng sẽ xuất hiện.
6. Nếu một thời gian cũng được hiển thị, trong menu Time format thả xuống, thay đổi giá trị từ (none) sang giá trị khác.
7. Click OK.

Tem thời gian xuất hiện trên trang, trông giống như chính ta đã nhập nó vào.

Sau khi một tem thời gian đã được thêm, để thực hiện một thay đổi, click đúp nó trong cửa sổ hiệu chỉnh, hộp thoại Date and Time mở lại. Tất cả các thành phần Web khác được hiệu chỉnh theo cùng một cách.

### **VIII.3.7. Lưu các thay đổi sang một site**

Nếu ta đã tạo một Web site cá nhân từ một template, ta có thể làm với nó bằng cách hiệu chỉnh text, tiêu đề chính và các hyperlink.

Để lưu một trang sau khi làm việc trên nó, click nút Save trên thanh công cụ Standard.

Để xóa một trang hoặc bất kỳ file khác, loại bỏ nó vĩnh viễn ra khỏi site, hãy thực hiện các bước sau đây:

1. Quay trở về khung nhìn Folders: click tab Web Site nằm trên cùng cửa sổ hiệu chỉnh.
2. Click phải tên file trong khung xem Folder.
3. Chọn Delete từ menu tắt
4. Một hộp thoại Confirm Delete hỏi file có nên thực sự bị xóa hay không. Click Yes để xác định việc xóa hoặc No nếu ta thay đổi ý định.

Khi ta sẵn sàng xem công việc mà ta đã thực hiện, hãy click nút Preview in Browser trên thanh công cụ Standard. Site của ta mở ra trong một trình duyệt Web, và ta có thể thưởng thức nó giống như những người khác sẽ tham quan nó – click các hyperlink hoặc các nút thanh liên kết để xem mỗi trang.

## **VIII.4. Tạo một site với sự trợ giúp của Wizard trong FrontPage 2003**

### **VIII.4.1. Mở một wizard tạo site**

Nhiều phần mềm sử dụng wizard, các hộp thoại vốn có làm cho một công việc khó khăn trở nên dễ dàng bằng cách phân chia nó thành một loạt câu hỏi.

FrontPage và các chương trình khác trong bộ Office được xác lập trên một máy tính thông qua wizard đã được cài đặt, do đó hình thức này sẽ quen thuộc với ta.

Trong FrontPage, các wizard có thể được xem là các template thông minh. Chúng có thể được sử dụng để tạo các site và các trang Web quá phức tạp đến nỗi không thể xử lý bằng một template.

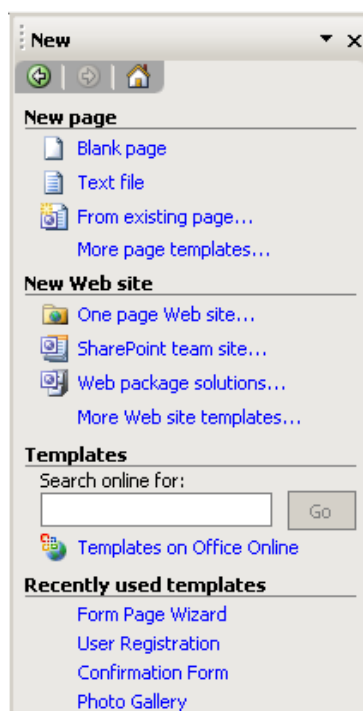
Bằng cách phân chia một tác vụ thành một loạt các bước đơn giản hơn, các wizard giúp ta có thể tạo các Web site phức tạp - chẳng hạn như một site công ty chuyên nghiệp 20 trang hoặc site hỗ trợ khách hàng - bằng cách trả lời vài câu hỏi đơn giản.

Hai loại wizard có sẵn trong FrontPage: các wizard Web site tạo toàn bộ các site và các wizard một trang.

Các wizard có thể được tạo khi ta đang tạo một site mới hoặc thêm một trang mới vào một site hiện hành.

Để gọi ra một site wizard, hãy thực hiện các bước sau:

1. Trên menu Standard chọn File → New. Khung New mở ra (ở phía bên phải cửa sổ - xem hình vẽ dưới)



Hình VIII.31. Khung New

2. Click hyperlink More Web Site Templates. Hộp thoại Web Site Templates mở ra với tab General nằm trên cùng. Ta có thể sử dụng một số wizard để tạo toàn bộ một site:
  - *Database Interface Wizard* sẽ tạo một site vốn có thể nối kết với một cơ sở dữ liệu Microsoft Access, hiển thị các record trên các trang Web và lưu thông tin được thu thập trên site vào sang cơ sở dữ liệu.
  - *Discussion Web Site Wizard* tạo một bảng thông báo nơi các khách tham quan đến site của ta có thể giao tiếp với nhau.
  - *Corporate Presence Wizard* sao chép một Web site hiện có sang máy tính của ta, ngay cả nếu nó không được tạo bằng FrontPage.
3. Để khởi động một wizard, chọn nó và click nút OK. Hộp thoại chính của wizard xuất hiện.
4. Để sử dụng wizard, trả lời từng câu hỏi mà nó đặt ra sau đó click Next để xem câu hỏi tiếp theo. Khi ta tiến đến câu hỏi sau cùng, một nút Finish hiển thị thay cho nút Next.
5. Khi ta sẵn sàng hoàn tất những tùy chọn mà ta đã chọn trong wizard, click Finish.



### **VIII.4.2. Nhập một site hiện có vào FrontPage**

Nếu ta đã tạo các Web site bằng phần khác trước khi sử dụng FrontPage, thì ta nên quan tâm đến Import Web Site Wizard, vốn có thể được sử dụng để đưa một site và tất cả file của nó vào trong FrontPage để hiệu chỉnh.

Với wizard này, ta có thể tận dụng các tính năng của FrontPage trên một site mà ban đầu không được tạo bằng phần mềm này.

Wizard có thể nhập một site từ một thư mục trên máy tính của ta, một thư mục trên mạng, hoặc một Web site trên Internet.

Để nhập một site, hãy thực hiện các bước sau:

1. Trên menu Standard chọn File → New để mở khung New.
2. Click hyperlink More Web site Templates. Hộp thoại Site Templates mở ra, liệt kê tất cả các site wizard mà ta có thể chọn.
3. Chọn biểu tượng Import Web Site Wizard. Trong hộp danh sách Specify the Location of the New Web Site, FrontPage chọn một tên cho một thư mục mới nơi các file của site sẽ được lưu trữ.
4. Để chọn một thư mục khác, click nút Browse. Hộp thoại New Web Site Location sẽ xuất hiện. Sử dụng hộp thoại này để tìm kiếm và mở thư mục nơi ta sẽ lưu trữ các Web site mà ta sẽ làm việc trên đó bằng FrontPage.
5. Mỗi site FrontPage phải được cung cấp thư mục riêng của nó. Để tạo một thư mục cho site được nhập vào này, click nút Create New Folder. Hộp thoại New Folder xuất hiện.
6. Cho thư mục một tên trong Name và click OK.
7. Trong hộp thoại New Web Site Location, click nút Open. Thư mục được liệt kê trong hộp thoại danh sách Specify the Location of the New Website trong Import Web Wizard.
8. Click OK. Import Web Wizard mở ra.

### **VIII.4.3. Chọn một phương pháp import**

Câu hỏi đầu tiên được hỏi bởi Import Web Site Wizard là cách để truy tìm site, như được trình bày ở hình dưới



Hình VIII.32. Import Web Site Wizard

Một site có thể được nhập theo năm cách, phụ thuộc vào vị trí của nó và kiểu truy cập nào mà ta có đối với vị trí đó:

- *FrontPage Server Extensions hoặc SharePoint Services, một kỹ thuật hiệu quả đòi hỏi một Web server được trang bị với một trong những tính năng này.*
- *FPT (File Transfer Protocol), cách phổ biến nhất mà các file được trao đổi trên Internet.*
- *HTTP (HyperTEtxt Transfer Protocol), một cách để truy tìm các trang Web và các file khác vốn có thể truy xuất trên một mạng cục bộ.*
- *WebDAV, một hệ thống tạo nội dung Web ít phổ biến hơn các phương pháp khác.*

Cách tốt nhất để nhập một site là sử dụng FrontPage Server Extensions hoặc SharePoint Services bởi vì điều này có thể truy tìm các trang và các file khác được đưa vào một Web site ngay cả nếu chúng không được trình bày trực tiếp cho các khách tham quan.

Khi ta sử dụng FrontPage để tạo một Web site, nó sẽ làm việc với nhiều file ở hậu cảnh trong suốt quá trình chuẩn bị một trang. Ví dụ, nếu site của ta sử dụng một theme, sẽ có hơn hai chục file đồ họa và các thư mục con của nó. Những file này không bao giờ được biểu diễn trực tiếp cho những người tham quan site của ta.

Thay vào đó, chúng được sử dụng bởi FrontPage để tạo các nút hover, đồ họa và định dạng các trang.

Bằng cách sử dụng các server extension hoặc SharePoint Services, wizard cũng có thể nhập các site con (subsite) – các Web site được chứa bên trong các thư mục của một site khác. (Ví dụ, một Web site trong thư mục C:\My Sites\Sports\baseball).

Nếu site để nhập nằm trên một phần mở rộng Web server hoặc SharePoint, ta có thể sử dụng chúng để nhập Web site này và toàn bộ site con của nó:

1. Chọn FrontPage Server Extensions hoặc SharePoint Services.
2. Nhập địa chỉ Web của trang chủ Web thuộc site vào trong trường text Web Site Location.

Để đưa vào các site con, chọn hộp kiểm Include Subsites.

Nếu site mà ta muốn nhập nằm trên máy tính của ta (hoặc trong một thư mục ở máy tính khác trên mạng của ta), ta có thể sao chép nó từ thư mục đó:

1. Chọn tùy chọn File System và click đúp Browse. Hộp thoại New Publish Location xuất hiện. Sử dụng hộp thoại này để tìm và chọn thư mục chính của Web site (thư mục nơi trang chủ của nó được lưu trữ).
2. Chọn thư mục và click Open. Thư mục được hiển thị trong trường text Web Site Location.
3. Để nhập các site con cùng một lúc (nếu có), chọn hộp kiểm Include Subsites.

Nếu site nằm trên một Web server mà ta có thể truy cập bằng cách sử dụng File Transfer Protocol (FTP), một tiêu chuẩn chung cho việc trao đổi các file với một Internet server, ta có thể sử dụng FTP account của ta để nhập site. Sau đây là cách thực hiện:

1. Chọn FTP.
2. Trong trường text Web Site Location, nhập địa chỉ của FTP server. Bởi vì nó đang sử dụng FTP, địa chỉ sẽ có ftp:// đứng trước thay vì http://
3. Trong trường Root Directory, gõ nhập vị trí và tên của thư mục mà ta chứa trang chủ của site.

4. Nếu server đòi hỏi Passive FTP, click hộp kiểm Use Passive FTP.

Nếu ta không thể tận dụng các phần mở rộng hoặc SharePoint, thì FTP là lựa chọn tốt nhất trong các tùy chọn còn lại.

Nếu site nằm trên một Web server mà không được trang bị các phần mở rộng hoặc SharePoint (hoặc nếu ta không biết), ta có thể lưu các file bằng việc truy tìm chúng từ Web bằng cách sử dụng HTTP.

Tùy chọn HTTP không hiệu quả bằng các phần mở rộng, SharePoint, hoặc FTP - chỉ lấy các trang được liên kết công cộng từ trang chủ. Nó cũng bỏ qua các file được sử dụng để tạo các trang đó, chẳng hạn như các theme FrontPage và những file khác vốn được sử dụng để tạo site.

Tuy nhiên, biện pháp sau cùng là HTTP có thể được sử dụng để nhập site:

1. Chọn HTTP.
2. Nhập địa chỉ Web của trang chủ thuộc site trong trường text Web Site Location. FrontPage tự mĩ về địa chỉ: Thay vì sử dụng một địa chỉ chính chẳng hạn như `www.example.com`, nó cụ thể hơn và cũng tham chiếu tên của trang chủ (nói cách khác, một địa chỉ nào như `www.example.com/index.html`).
3. Sau khi chọn một trong những phương pháp import, click Next để tiếp tục. Wizard được cập nhật để phản ánh sự lựa chọn của ta.
4. Nếu ta đang nhập một site từ một vị trí được bảo vệ bởi password, một hộp thoại Name and Password Required mở ra. Sử dụng các trường Name và Password để cung cấp thông tin đăng nhập của ta, và sau đó click OK.

Sau khi chọn một phương pháp nhập, click Next.

#### **VIII.4.4. Chọn vị trí để lưu site**

Câu hỏi kế tiếp mà wizard có thể đặt ra là nơi để lưu site. Bởi vì ta đã chọn một thư mục, nó xuất hiện trong trường text trường Local Copy Location, trường này được tắt để ta không thể thay đổi nó.

Tuy nhiên, có một cách để thay đổi một thực hiện một thay đổi nếu ta thay đổi ý định và chọn một thư mục khác:

1. Hủy chọn hộp kiểm Add to Current Web Site. Trường Location Copy Location và nút Browse trở nên được chọn.

2. Click nút Browse. Hộp thoại New Page Location xuất hiện. Sử dụng hộp thoại này để chọn (hoặc tạo) một thư mục mới nơi mà các file của site sẽ được lưu.
3. Chọn thư mục và click Open. Hộp thoại đóng lại, đưa ta trở về wizard.
4. Chọn hộp Add to Current Web Site. Click Next để đi đến câu hỏi kế tiếp.

#### **VIII.4.5. Thu thập thông tin phản hồi từ các khách tham quan Web site của ta**

Một trong những form dễ nhất để tạo là một trang phản hồi - một vị trí trong site của ta nơi những khách tham quan có thể gửi cho ta một lời nhận xét góp ý riêng tư. Điều này cung cấp một dịch vụ tiện lợi cho những khách tham quan của ta và có một ưu điểm thứ hai – nó cho phép ta bỏ qua địa chỉ email của ta từ site của ta.

Tại sao điều này lại hữu dụng? Khi ta bắt đầu xuất Web, bất kỳ địa chỉ email mà ta đặt trên một trang sẽ nhanh chóng được phát hiện bởi các spammer - những người tiếp thị email tạp nhạp lấp đầy các inbox bằng những lời đề nghị mang tính thương mại không cần thiết.

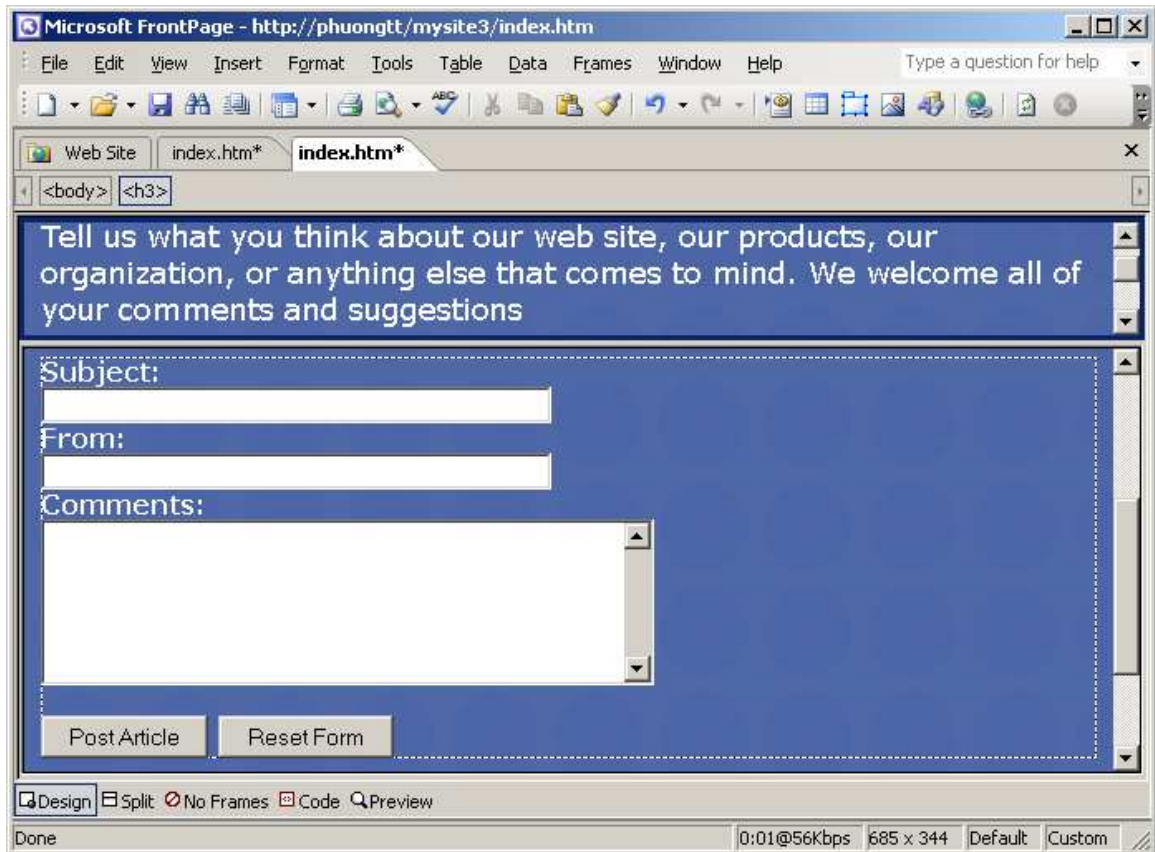
Để tìm các địa chỉ để thêm vào các danh sách thư tín của họ, cá spammer thường xuyên sử dụng phần mềm để truy tìm các trang trên World Wide Web. Điều mà ta chắc chắn không thể tránh khỏi là site của ta sẽ được tìm thấy nhanh chóng bởi những chương trình này - thường là trong một thời gian ngắn từ một đến hai tuần.

Bằng cách đặt một trang phản hồi thay vì địa chỉ của ta trên site của ta, ta sẽ giảm đi lượng spam mà ta nhận được.

Để thêm một trang phản hồi vào site của ta, hãy thực hiện các bước sau:

1. Mở site (nếu nó chưa được mở) và trên menu Standard chọn File → New. Khung New mở nằm dọc theo cửa sổ biên soạn.
2. Click hyperlink More Page Templates. Hộp thoại Page Templates mở ra, hiển thị các trang mà ta có thể tạo bằng cách sử dụng các template hoặc wizard.
3. Chọn biểu tượng Feedback Form và click OK. Một trang phản hồi tạo sử dụng các màu, text và nền giống với phần còn lại của site. Nó mở trong cửa sổ biên soạn. Một đường chấm chấm biểu thị các đường viền của form. Mỗi phần của form phải được đặt trong đường biên soạn này. Ở đầu trang,

một đoạn được tô màu đỏ tía mô tả form và cung cấp một số thông tin hướng dẫn về cách nó được sử dụng. Text này là một lời bình chú, text giải thích xuất hiện chỉ trong cửa sổ biên soạn của FrontPage chứ không phải khi một trang được xem trong một trình duyệt Web. (Để kiểm tra điều này, click nút Preview bên dưới cửa sổ biên soạn).



*Hình VIII.33. Hiệu chỉnh một trang phản hồi*

4. Để xóa lời chú thích, click nó một lần và nhấn phím Delete.
5. Nếu ta muốn thay đổi cách diễn đạt của một câu hỏi, hiệu chỉnh nó như ta thường thực hiện đối với bất kỳ text khác trong FrontPage: Nhấp con trỏ trên trường và sử dụng bàn phím để thêm hoặc loại bỏ các thông tin.
6. Việc định vị và định dạng các thành phần khác nhau của form cũng có thể được thay đổi giống như bất kỳ phần khác của một trang Web.
7. Để xóa bất kỳ phần của form, kéo chuột lên trên nó và nhấn phím Delete

Khi một trang phản hồi được tạo ra, FrontPage xác lập nó để lưu các câu trả lời sang file `feedback.txt` trong thư mục `_private` của Web site.

Để thực thi điều này, ta phải xuất site của ta lên một Web server được trang bị bằng FrontPage Server Extensions hoặc SharePoint Servers, những cải tiến nhằm giúp những người sử dụng FrontPage thêm các tính năng đặc biệt (giống như vậy) vào các site của họ.

Vào thời điểm này, ta có thể thay đổi cách thông tin phản hồi của khách tham quan được lưu như thế nào. Nó có thể được lưu sang một file theo nhiều dạng khác nhau hoặc gửi đến ta bằng một email.

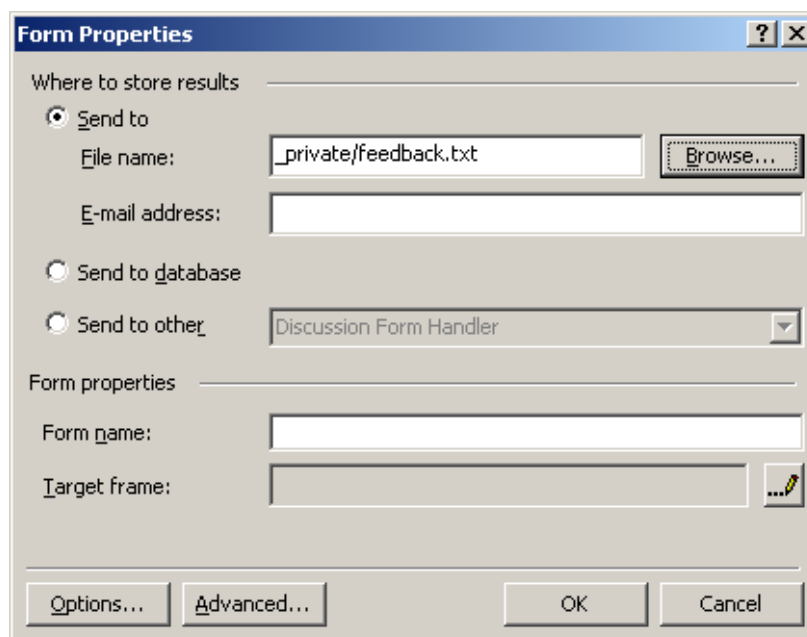
#### ***VIII.4.6. Lưu thông tin phản hồi của khách tham quan sang một file***

Những câu trả lời được thu thập trên một form có thể được lưu trữ trong một file trên Web server của ta theo ba dạng khác nhau:

- **Text** - Dạng này có thể được xem trong một bộ biên soạn text hoặc được tải trong một chương trình cơ sở dữ liệu và trang bảng tính, chẳng hạn như Microsoft Access và Microsoft Excel.
- **HTML** - Dạng này có thể được đọc bằng một trình duyệt.
- **XML** – Đây là một dạng phổ biến cho dữ liệu được tích hợp rộng rãi vào FrontPage 2003 và các chương trình còn lại của bộ Office.

Để lưu thông tin phản hồi sang một file, hãy thực hiện những bước sau đây:

1. Click phải ở bất cứ nơi nào trong các đường viền của form - đường chấm chấm được minh họa ban đầu ở hình trên.
2. Chọn Form Properties từ menu ngữ cảnh. Hộp thoại Properties xuất hiện như được minh họa trong hình dưới.



Hình VIII.34. Chọn cách lưu thông tin phản hồi của khách

3. Thông tin phản hồi của khách tham quan sẽ được lưu dưới dạng text. Để chọn một dạng khác, click nút Options. Hộp thoại Saving Results xuất hiện.
4. Sử dụng hộp File Format xổ xuống để chọn một dạng cho thông tin phản hồi:
  - Để giúp ta đọc dễ dàng với Microsoft Excel và những chương trình cơ sở dữ liệu khác, hãy chọn *Text Data-base Using Comma as a Separator*. Điều này tạo một file text với phần mở rộng tên file.csv.
  - Để đọc nó bằng một trình duyệt Web, một lựa chọn tốt là *HTML hoặc Formatted Text Within HTML*. Điều này tạo ra một file.html vốn sử dụng cùng một định dạng với các trang Web mà ta tạo bằng *ForntPage*.
  - Để lưu theo dạng XML, chọn *XML*. Điều này tạo ra một file.xml.
5. Click OK.
6. Để chọn thư mục nơi file sẽ được lưu trữ ở đó, click nút Browse. Hộp thoại Current Web site xuất hiện. Sử dụng hộp thoại này để chọn một trong những thư mục trên site của ta.
7. Click OK để đóng hộp thoại Form Properties.

Form sẽ được xác lập để lưu trữ các thông điệp phản hồi trong một file được chỉ định. Nếu file này không hiện hữu khi một người nào đó sử dụng form phản hồi sẽ được tạo ra.



### **VIII.4.7. Nhận thông tin phản hồi của khách tham quan bằng email**

Thông tin phản hồi cũng có thể được gửi đến ta bằng email thay vì được lưu trong một site của ta.

Khi nó được gửi đến, nó trông giống như sau:

```
Message Type:      Suggestion
Subject:           Web Site
Username:          Sam Snett
UserEmail:         snett@sampublishing.com
ContractRequested: ContacRequested
Date:              22 Aug 2003
Time:              21:19:30
Comments:
In this thing on?
```

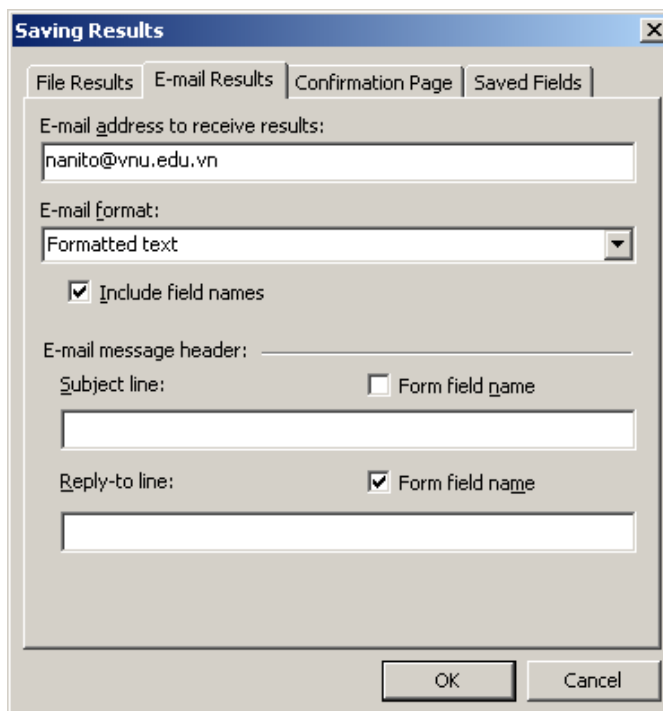
Trong email, các câu trả lời khác nhau trên form phản hồi được hiển thị trên các dòng riêng biệt. Trong ví dụ này, Sam Snett với địa chỉ email snett@sampublishing.com đã gửi một đề nghị với chủ đề “Web Site”. Mail hiển thị ở cuối. Sam đặt câu hỏi, “Is this thing on?”

Thư phản hồi bắt nguồn từ một địa chỉ tự động trả lời được thiết lập bởi Web host của ta - ta có thể thay đổi địa chỉ này thành địa chỉ của người đang liên lạc với ta như một mô tả ở phần kế tiếp.

Để gửi các trả lời phản hồi đến một địa chỉ email, hãy thực hiện các bước sau:

1. Click chuột phải ở bất cứ nơi nào trong form và chọn Form Properties từ menu tắt vừa xuất hiện. Hộp thoại Form Properties mở ra.
2. Nhập một địa chỉ email trong trường text E-mail Address. Địa chỉ này phải là địa chỉ email của ta hoặc địa chỉ của một người đang trả lời thư cho site của ta.
3. Xóa bất kỳ text trong hộp Filename. Thông tin phản hồi sẽ được gửi bằng thư bằng cách sử dụng dạng của ví dụ SamSnett. Để chọn một dạng khác, click nút Options. Hộp thoại Saving Result xuất hiện.
4. Click tab Email Result để đưa nó lên phía trước.
5. Chọn một dạng trong hộp Email Format thả xuống. Các tùy chọn có sẵn giống như cá tùy chọn dành cho các file. Email phản hồi được cung cấp dòng chủ đề “Data posted to form 1”, theo sau là địa chỉ Web của trang phản hồi.

- Để xác lập một dòng chủ đề khác và giúp dễ dàng tìm những email này, hãy gõ nhập nó trong trường text Subject Line.
- Để tận dụng địa chỉ email của người đã điền đầy đủ thông tin vào form, nhập text UserEmail trong trường text Reply-To Line và chọn hộp kiểm Form Field Name nằm ngay kế bên nó (xem hình dưới).



*Hình VIII.35. Hộp thoại Saving Result*

Click OK để đóng hộp thoại Saving Results, và sau đó click OK để đóng hộp thoại Form Properties.

Khi lưu kết quả của một form dưới dạng email, FrontPage có thể hiển thị một hộp thoại chỉ định rằng form không thể gửi kết quả bằng email và hỏi xem địa chỉ email có nên được loại bỏ hay không.

Nếu ta biết rằng Web server của ta cung cấp các server Extensions hoặc SharePoint, hãy click No để giữ lại địa chỉ email và bỏ qua lời cảnh báo. Sau khi ta xuất trang phản hồi, nó sẽ hoạt động một cách chính xác.

## **Bài tập**

- Thay vì tự viết các thẻ HTML, hãy dùng công cụ FrontPage để tạo tất cả các trang Web cho các bài tập ở những chương trước.

## BẢNG CÁC TỪ VIẾT TẮT

**Browser:** trình duyệt là chương trình đọc nội dung của file XML và hiển thị thông tin của file lên trên màn hình.

**Cascading Style Sheets:** một ngôn ngữ khai báo cách hiển thị các phần tử trong file XML.

**Style sheet:** là cách gọi tắt của Cascading Style Sheets nhưng nó ám chỉ đến một file chứa các khai báo style.<sup>7</sup>

**HTML:** Hyper Text Markup Language.

**WWW:** World Wide Web

**Markup:** Đánh dấu

**Language:** Ngôn ngữ

**Text:** Văn bản

**Rect:** Hình chữ nhật

**Circle:** Hình tròn

**Polygon:** Hình tam giác

**Transparency:** Độ trong suốt

## TÀI LIỆU THAM KHẢO

- [1] 24 bài tự học thiết kế Web với Front Page 2003, Thuận Thành, NXB Thanh niên.
- [2] HTML, DHTML and JavaScript, Aptech WorldWide's book.