

LỜI MỞ ĐẦU

Ngôn ngữ Siêu văn bản HTML (*Hyper Text Markup Language*) 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 (*World Wide Web*). 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 ``. 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 `` và `` là thẻ “**bold**” nghĩa là chữ béo. Toàn bộ đoạn văn bản giữa `` và `` 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 `` xin chào `` 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

LỜI MỞ ĐẦU	1
CHƯƠNG I. NHỮNG KHÁI NIỆM CƠ SỞ	8
I.1. World Wide Web là gì?	8
I.2. HTML là gì?	11
I.3. Các đặc điểm của siêu văn bản	12
I.3.1. Độc lập với phần cứng và phần mềm	12
I.3.2. Độc lập với khái niệm trang và thứ tự các trang	13
I.3.3. Website và trang chủ - homepage	13
I.4. Soạn thảo văn bản - những vấn đề chung	14
I.4.1. Trang mã nguồn HTML và trang Web	14
I.4.2. Các thẻ HTML	15
I.4.3. Các quy tắc chung	16
I.4.4. Cấu trúc của một tài liệu HTML	17
I.4.5. Các phần tử HTML (HTML element)	18
Bài tập	18
CHƯƠNG II. TRÌNH BÀY TRANG	19
II.1. Tạo tiêu đề	19
II.2. Thẻ trình bày trang	20
II.2.1. Một số thẻ chính	20
II.2.2. Các thuộc tính của thẻ trình bày trang	24
Bài tập	26
CHƯƠNG III. DANH SÁCH VÀ BẢNG TRONG HTML	27
III.1. Các kiểu danh sách	27
III.1.1. Danh sách không đánh số thứ tự	27
III.1.2. Danh sách đánh số thứ tự	28
III.1.3. Danh sách các định nghĩa	30
III.1.4. Danh sách phối hợp, lồng nhau	30

III.2. Bảng biểu.....	31
III.2.1. Khung cấu trúc	31
III.2.2. Một số lưu ý về bảng	32
III.2.3. Các ví dụ.....	36
Bài tập.....	39
CHƯƠNG IV. ĐƯA HÌNH ẢNH VÀO TÀI LIỆU HTML	40
IV.1. Hình ảnh tĩnh	40
IV.1.1. Tập ảnh.....	40
IV.1.2. Thẻ <IMG...>	40
IV.2. Các thuộc tính của thẻ chèn hình ảnh.....	40
IV.2.1. Thuộc tính ALT	40
IV.2.2. Thuộc tính WIDTH và HEIGHT	41
IV.2.3. Thuộc tính ALIGN	42
IV.2.4. Thuộc tính VSPACE và HSPACE	42
Bài tập.....	43
CHƯƠNG V. CÁC MỐI LIÊN KẾT SIÊU VĂN BẢN	44
V.1. Thẻ neo và mối liên kết	44
V.1.1. Thuộc tính HREF	44
V.1.2. Liên kết ra ngoài – External Links.....	44
V.1.3. Địa chỉ tuyệt đối.....	45
V.1.4. Địa chỉ tương đối	45
V.1.5. Liên kết nội tại – Internal Link	45
V.2. Dùng hình ảnh làm đầu mối liên kết	46
V.2.1. Thay chữ bằng hình	46
V.2.2. Image Map - thẻ AREA.....	46
V.3. Đưa âm thanh vào tài liệu	47
V.3.1. Liên kết đến tệp âm thanh	47
V.3.2. Tạo âm thanh nền.....	47

V.4. Đưa Video vào tài liệu	48
V.4.1. Chèn tệp Video.....	48
V.4.2. Nhúng tệp video	48
Bài tập.....	49
CHƯƠNG VI. BÀY TRÍ NỀN VÀ KHUNG.....	50
VI.1. Màu nền và văn bản	50
VI.1.1. Đặt màu nền	50
VI.1.2. Màu chữ của văn bản.....	50
VI.1.3. Màu của đầu mối liên kết - Thuộc tính LINK, VLINK và ALINK	50
VI.1.4. Thuộc tính và mã màu	51
VI.2. Nạp hình ảnh làm nền cho trang văn bản	52
VI.2.1. Thuộc tính BACKGROUND.....	52
VI.2.2. Water mark.....	52
VI.2.3. Hãy ký tên vào tài liệu của mình	53
VI.3. Khung – Frames	54
VI.3.1. Trang trí khung	54
VI.3.2. Thành phần FRAMESET	55
VI.4. Thiết lập Target, thẻ NOFRAME và IFRAME	58
VI.4.1. Thiết lập Target.....	58
VI.4.2. Thẻ NOFRAMES	59
VI.4.2. Nhúng frame - thẻ IFRAME	59
Bài tập.....	60
CHƯƠNG VII. BIỂU MẪU STYLE VÀ CASCADING STYLE SHEET	61
VII.1. FORM	61
VII.1.1. FORM là gì?	61
VII.1.2. Các thành phần trong FORM	62
VII.1.3. Thêm tính cấu trúc cho FORM.....	69
VII.2. Cascading style sheet	71

VII.2.1. Inline Style.....	71
VII.2.2. Giới thiệu Style Sheet	73
VII.2.3. Javascript Style Sheet	74
VII.2.4. Thuật ngữ Style Sheet.....	76
VII.2.5. Các chú thích trong Style Sheet	91
VII.2.6. Lợi ích của các Style Sheet.....	91
VII.2.7. Kết hợp Style Sheet với HTML	92
VII.2.8. Thứ tự ưu tiên của các style (Cascading)	96
CHƯƠNG VIII. CÔNG CỤ SOẠN THẢO TRỰC QUAN WEB (MICROSOFT FRONTPAGE 2003)	99
VIII.1. Tạo một trang Web.....	99
VIII.1.1. Bắt đầu sử dụng FrontPage2003	99
VIII.1.2. Tạo một trang từ một template.....	100
VIII.1.3. Tạo và lưu một trang mới.....	101
VIII.1.4. Tạo một đề mục	102
VIII.1.5. Chọn font và màu.....	103
VIII.2. Tổ chức một trang với các liên kết, danh sách và bảng	106
VIII.2.1. Thêm một hyperlink vào một trang Web	106
VIII.2.2. Tạo một danh sách	109
VIII.2.3. Tổ chức một trang với các bảng.....	111
VIII.2.4. Hiển thị hình ảnh trên một trang Web	119
VIII.2.5. Tạo một Web site mới	132
VIII.2.5.Khai thác site mới	134
VIII.3. Phát triển nhanh một site với các template	141
VIII.3.1. Chọn một template Web site	142
VIII.3.2. Tạo một Web site mới	143
VIII.3.3. Tạo tùy biến Web site mới của ta	144
VIII.3.4. Thêm và loại bỏ các lời chú thích.....	145

VIII.3.5. Khai thác template Personal Web Site	146
VIII.3.6. Thêm một tem thời gian vào một trang Web	147
VIII.3.7. Lưu các thay đổi sang một site.....	148
VIII.4. Tạo một site với sự trợ giúp của Wizard trong FrontPage 2003.....	149
VIII.4.1. Mở một wizard tạo site	149
VIII.4.2. Nhập một site hiện có vào FrontPage	151
VIII.4.3. Chọn một phương pháp import	151
VIII.4.4. Chọn vị trí để lưu site	154
VIII.4.5. Thu thập thông tin phản hồi từ các khách tham quan Web site của ta	155
VIII.4.6. Lưu thông tin phản hồi của khách tham quan sang một file	157
VIII.4.7. Nhận thông tin phản hồi của khách tham quan bằng email	159
Bài tập.....	160
BẢNG CÁC TỪ VIẾT TẮT	162
TÀI LIỆU THAM KHẢO.....	163

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

`Joe Cool`

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 mốc 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:

`Suppliers`

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

``

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 mắt.

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 sĩ), 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 diễn đạt yêu cầu cần phải làm gì chứ không cụ thể cần làm như thế nào.

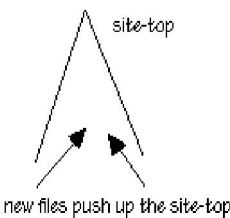
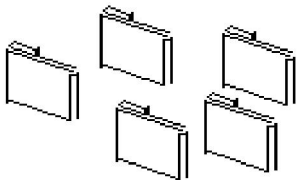
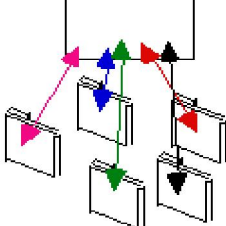
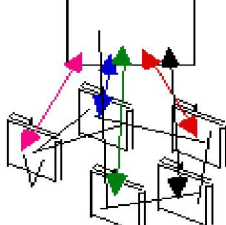
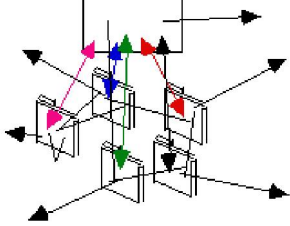
Cũng vì lẽ đó mà bạn không thể chắc chắn trang tài liệu siêu văn bản của bạn sẽ hiện lên màn hình chính xác là như thế nào vì còn tùy theo trình duyệt thể hiện các thành phần HTML ra sao.

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 (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 .

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 kí tự dấu lớn hơn ">", dấu nhỏ hơn "<" đã được dùng làm thẻ đánh dấu, do đó để hiển thị các kí tự này HTML quy định cách viết: > <

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

&tên_quy_định;

5 ký tự đặc biệt hay dùng

Ký tự	Cách viết
<	<
>	>
“	"
Ký tự trắng	
&	&

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 (
), 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 thể 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:

Nếu không muốn chèn một dòng trắng mà chỉ đơn thuần muốn xuống dòng mới thì cần sử dụng thẻ
 (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:

February 8, 1995, CERN

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 AThợ rèn
Trần Thị B   Thợ Sơn
</PRE>
```

Kết quả thu được:

Nguyễn Văn AThợ rèn Trần Thị B Thợ 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 lẻ
Một mình làm cả cuộc phân ly
</P>
```

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

<p>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>
--

b. Thuộc tính Clear của thẻ xuống dòng

Thẻ xuống dòng `
` 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:

Thẻ, thuộc tính	Miêu tả
<HR>	Chèn dòng kẻ ngang suốt chiều rộng của sổ màn hình
WIDTH = n%	Thay đổi độ dài của đường kẻ, chiếm n% độ rộng của 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 ... đánh dấu bắt đầu và kết thúc danh sách. Nó có tác dụng lùi lề trái danh sách vào sâu hơn.

Các thẻ 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:

- Mục thứ 1
 - Mục thứ 2
 - Mục thứ 3

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 `...` đánh dấu bắt đầu và kết thúc danh sách. Nó có tác dụng lùi lề trái danh sách vào sâu hơn.

Các thẻ 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">1. Mục thứ 12. Mục thứ 23. Mục thứ 3 |
|--|

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 |
| II. Mục thứ 2 |
| 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 |
| Nội dung mục thứ 1 |
| Mục thứ 2 |
| Nội dung mục thứ 2 |
| Mục thứ 3 |
| 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"> • Mức 1 <ul style="list-style-type: none"> • Mức 1.1 • Mức 1.2 • Mức 2 <ul style="list-style-type: none"> • Mức 2.1 • Mức 2.2 |
|--|

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 chông 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 lấp 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 bề 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:

Tổng kết năm học 2003 - 2004			
Thống kê	HS giỏi	HS khá	HS trung bình
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 “hiều” 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 xâu 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 nhay mat 	Day la hình nhay mat 
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

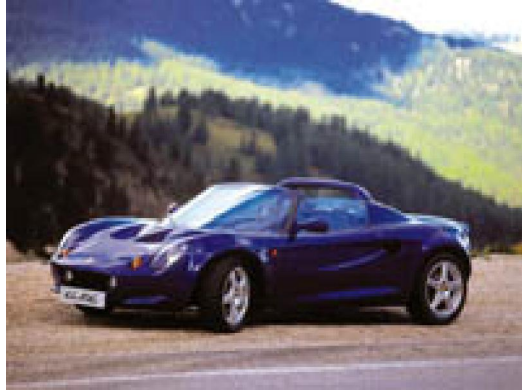
Ví dụ:

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

Ben trai  Ben phải

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.